

Белорусско-Российский университет

Кафедра «Программное обеспечение
информационных технологий»

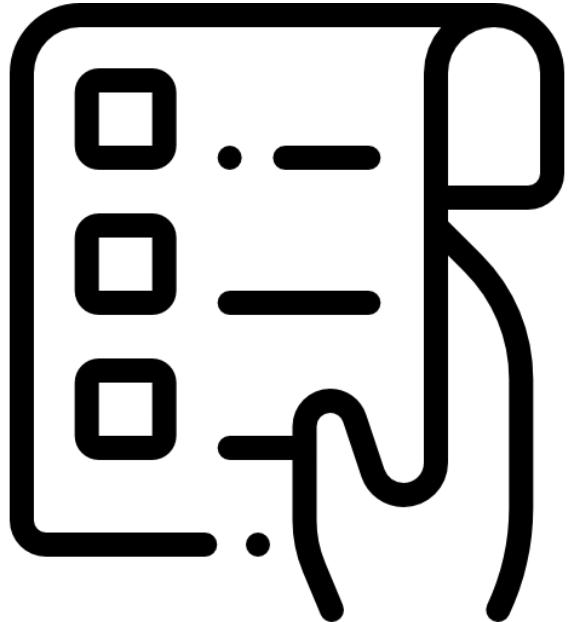
ЭВМ, периферийные устройства и контроллеры

Тема: Цифровая логика и
базовые компоненты ЭВМ

Кутузов Виктор Владимирович

Республика Беларусь, Могилев, 2025





Содержание лекции

Содержание лекции

Тема: Цифровая логика и базовые компоненты ЭВМ

1. Рекомендуемые материалы по теме
2. **Цифровая логика и базовые компоненты ЭВМ**
3. **Комбинационная логика**
4. Базовые комбинационные блоки
5. Мультиплексоры и демультиплексоры
6. Шифраторы и дешифраторы
7. Преобразователь кодов
8. Сумматоры (Вычитатели)
9. Компараторы
10. Арифметико-логические устройства (АЛУ)

Содержание лекции

Тема: Цифровая логика и базовые компоненты ЭВМ

10. Последовательностная логика

11. Триггеры

12. Регистры

13. Счетчики

14. Изучение работы последовательной и комбинационной логики

15. Современные методы проектирования цифровых устройств

16. Дополнительные материалы по теме на YouTube

Дополнительные материалы по теме на YouTube

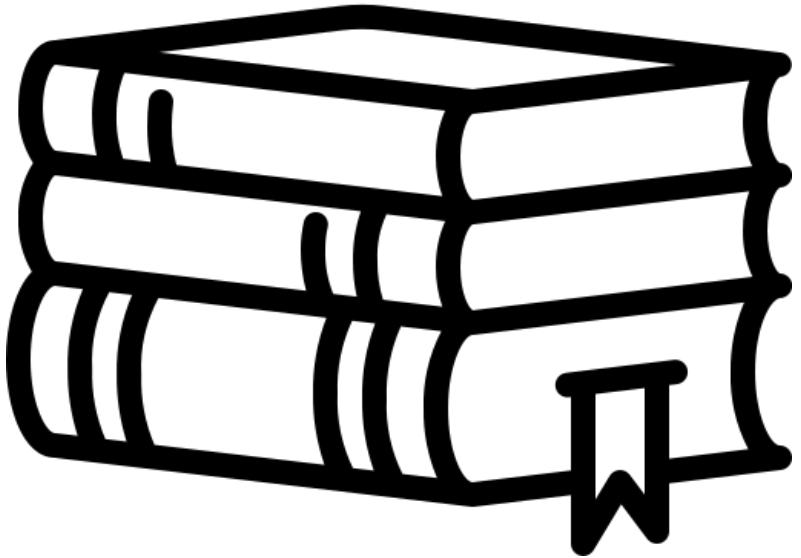
Тема: Цифровая логика и базовые компоненты ЭВМ

1. [Дополнительные материалы по теме на YouTube](#)
2. [Булева алгебра](#)
3. [Комбинационная логика](#)
4. [Последовательностная логика](#)
5. [Сумматоры, Полусумматоры](#)
6. [Мультиплексоры, Демультиплексоры](#)
7. [Шифраторы, Дешифраторы](#)
8. [Преобразователи кодов](#)
9. [Компараторы](#)
10. [Триггеры](#)

Дополнительные материалы по теме на YouTube

Тема: Цифровая логика и базовые компоненты ЭВМ

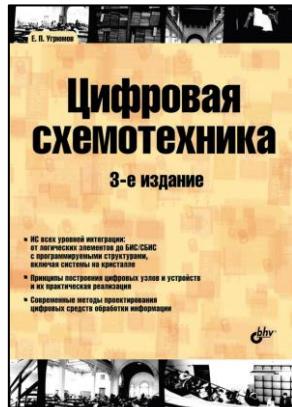
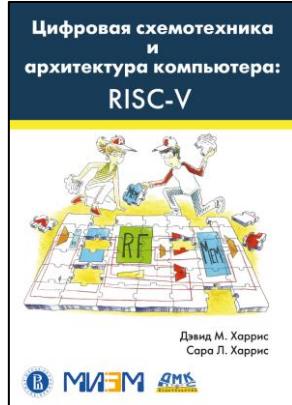
11. [RS-триггер](#)
12. [R-триггер](#)
13. [D-триггер](#)
14. [T-триггер](#)
15. [JK-триггер](#)
16. [MS-триггер](#)
17. [Регистры](#)
18. [Счетчики](#)
19. [Арифметико-логические устройства \(АЛУ\)](#)
20. [YouTube каналы и плейлисты](#)



Рекомендуемые материалы по теме



Литература

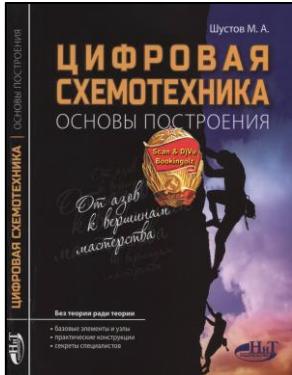


Сара Л. Харрис, Дэвид Харрис **Цифровая схемотехника и архитектура компьютера: RISC-V** / пер. с англ. В. С. Яценкова, А. Ю. Романова; под ред. А. Ю. Романова. – М.: ДМК Пресс, 2021. – 810 с. <https://rutracker.org/forum/viewtopic.php?t=6204850>

Угрюмов, Е. П. **Цифровая схемотехника**: учеб., пособие для вузов. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2010. – 816 с. <https://djvu.online/file/I5iaM7nsDIFK9>

Шустов, М. А. **Цифровая схемотехника. Практика применения**. – СПб.: Наука и Техника, 2018. – 432 с. <https://djvu.online/file/psKYETGSqWW7w>

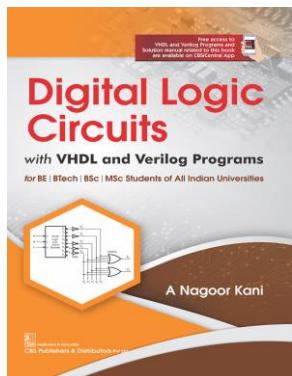
Литература



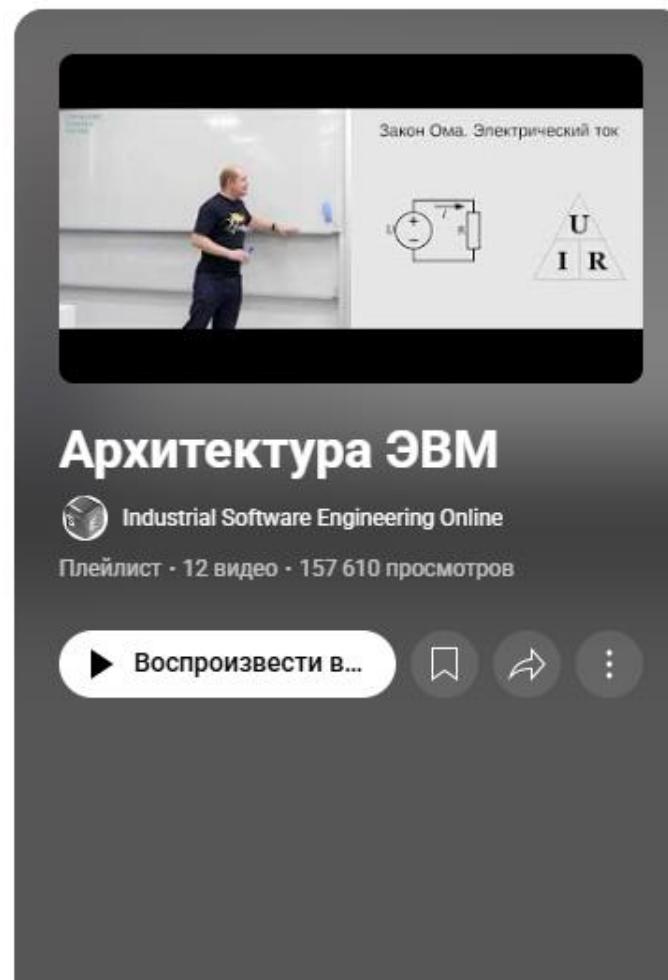
Шустов, М. А. **Цифровая схемотехника. Основы построения.** – СПб.: Наука и Техника, 2018. – 320 с.
<https://djvu.online/file/jleHgaxy8AOxE>



Булатов, В. Н. **Основы аналоговой и цифровой электроники. Цифровая электроника:** учебное пособие / В. Н. Булатов; Оренбургский гос. ун-т, 2020. – 206 с.
http://elib.osu.ru/bitstream/123456789/13104/1/20804_20200528.pdf



Kani A. N. (Ed.) **Digital Logic Circuits: With VHDL and Verilog Programs.** – CBS, 2025. – 1410 р.
<https://www.twirpx.com/file/4364828/>



Курс: Архитектура ЭВМ (понятно о том из чего состоит компьютер и как он работает, на уровне его архитектуры) <https://www.youtube.com/playlist?list=PLnseyzyGdZdfv8H7LkvyVVE33fbBZaSdH>

Архитектура ЭВМ. Лекция 0: Предварительные сведения

Industrial Software Engineering Online • 5 лет назад • 188 тыс. просмотров

Архитектура ЭВМ. Лекция 1: Типы архитектур. Комбинационная и последовательная логика.

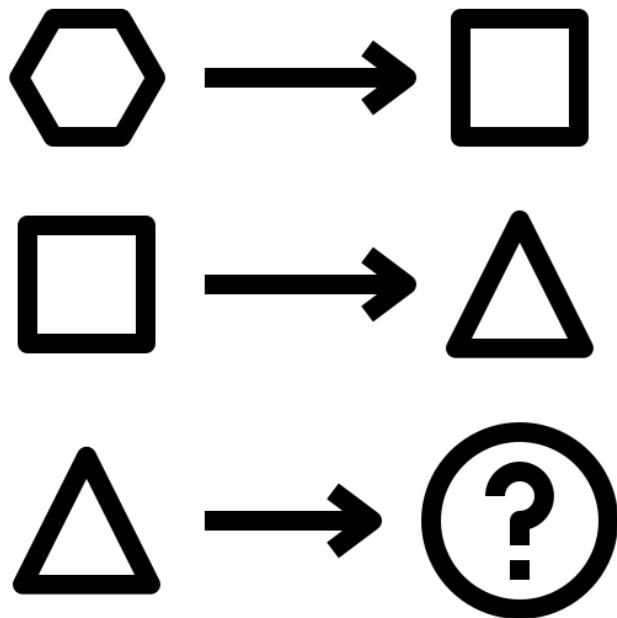
Industrial Software Engineering Online • 3 года назад • 79 тыс. просмотров

Архитектура ЭВМ. Лекция 2: АЛУ. Устройство памяти

Industrial Software Engineering Online • 3 года назад • 44 тыс. просмотров

Архитектура ЭВМ. Лекция 3: Кодирование и тип инструкций процессора.

Industrial Software Engineering Online • 3 года назад • 30 тыс. просмотров



Цифровая логика и базовые компоненты ЭВМ



Цифровые технологии

- В сегодняшнем мире термин **цифровой** стал частью нашего ежедневного общения благодаря повсеместному использованию цифровых систем и цифровой техники почти во всех областях жизни: в компьютерах, автоматике, робототехнике, в медицине и на производстве, транспорте, развлечениях, космических исследованиях и т.д.
- Начнем с представления базовых концепций, которые являются фундаментальной частью цифровых технологий.
 - Численные представления
 - **Существуют два основных способа представления численных значений величин: аналоговый и цифровой.**

Аналоговое представление

- **При аналоговом представлении** величина отображается напряжением, током или движением стрелки измерительного прибора пропорционально значению данной величины.
- **Например:**
 - **Автомобильный спидометр**, в котором отклонение стрелки прибора пропорционально скорости машины.
 - **Ртутный термометр**, в котором высота столбика ртути пропорциональна температуре в комнате.
 - **Микрофон**. В этом приборе выходное напряжение генерируется пропорционально амплитуде звуковых волн, попадающих в микрофон.
- Перечисленные выше **аналоговые величины имеют общую важную характеристику: они могут варьироваться в непрерывном диапазоне значений**. Скорость автомобиля может иметь любое значение между 0 и, скажем, 100 километрами в час. Точно так же и выходной сигнал микрофона может иметь любое значение в диапазоне от 0 до 10 мВ (например, 1 мВ; 2,3724 мВ; 9,9999 мВ).

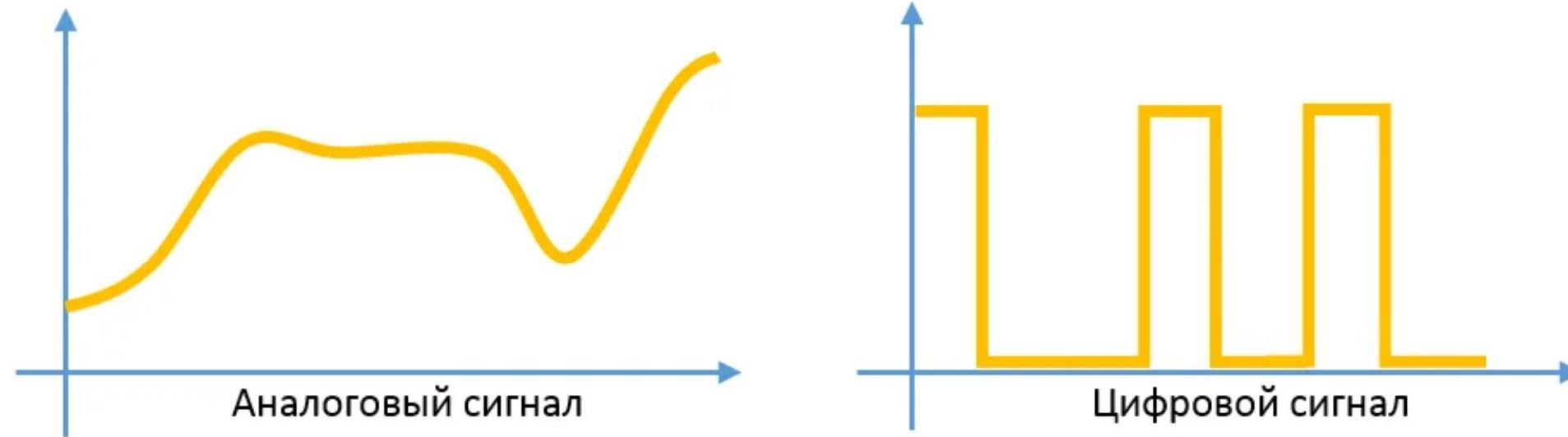
Цифровое представление

- **При цифровом представлении** величины отображаются не пропорциональными величинами, а символами, которые называются разрядами.
- В качестве примера возьмем электронные часы, которые показывают время суток в форме десятичных разрядов, соответствующих часам и минутам (иногда и секундам). Как известно, время суток изменяется постоянно, что нельзя сказать о показаниях электронных часов, — они меняются с шагом 1 за минуту (или за секунду).
- Другими словами, такое **цифровое представление времени суток изменяется дискретными шагами** по сравнению со временем на аналоговых часах, где показания циферблата меняются непрерывно.

Аналоговые и цифровые величины

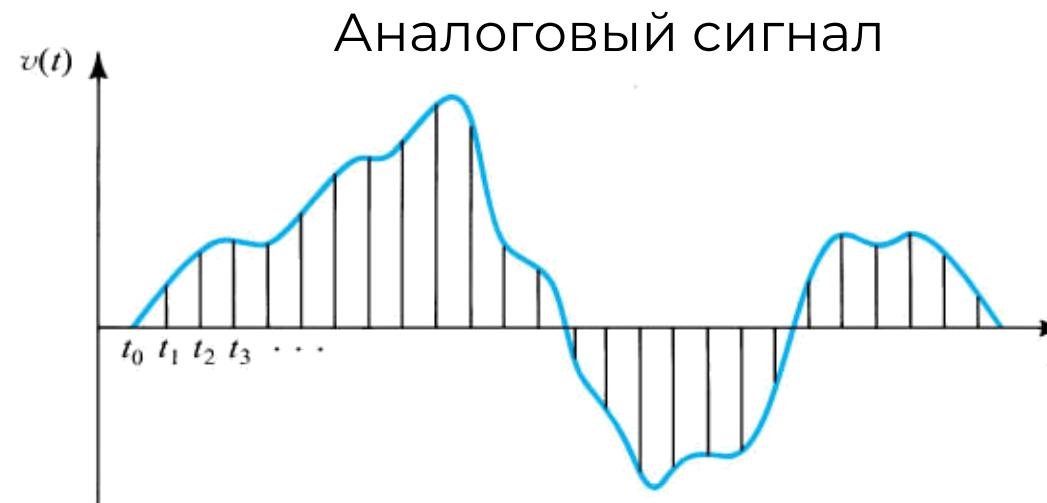
- Главную разницу между аналоговыми и цифровыми величинами можно записать так:

Аналоговый = непрерывный
Цифровой = дискретный (шаг за шагом)

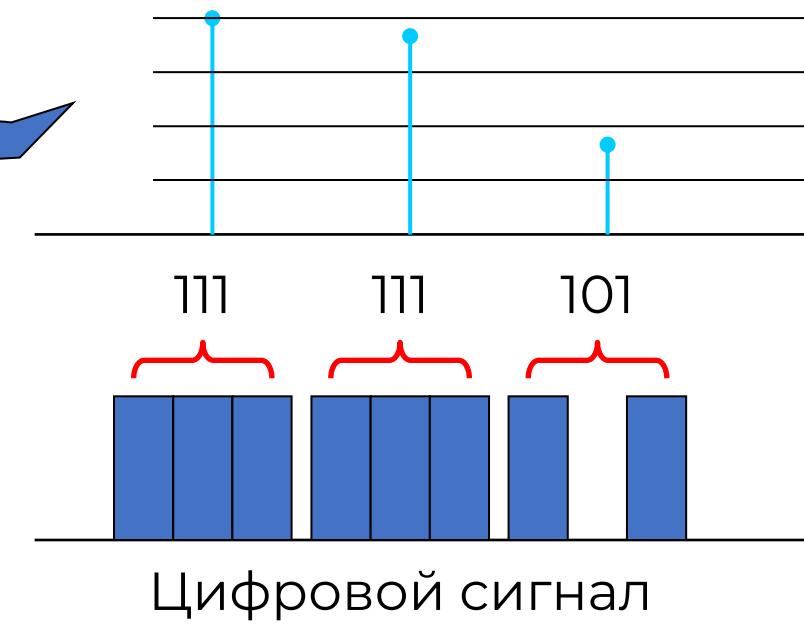
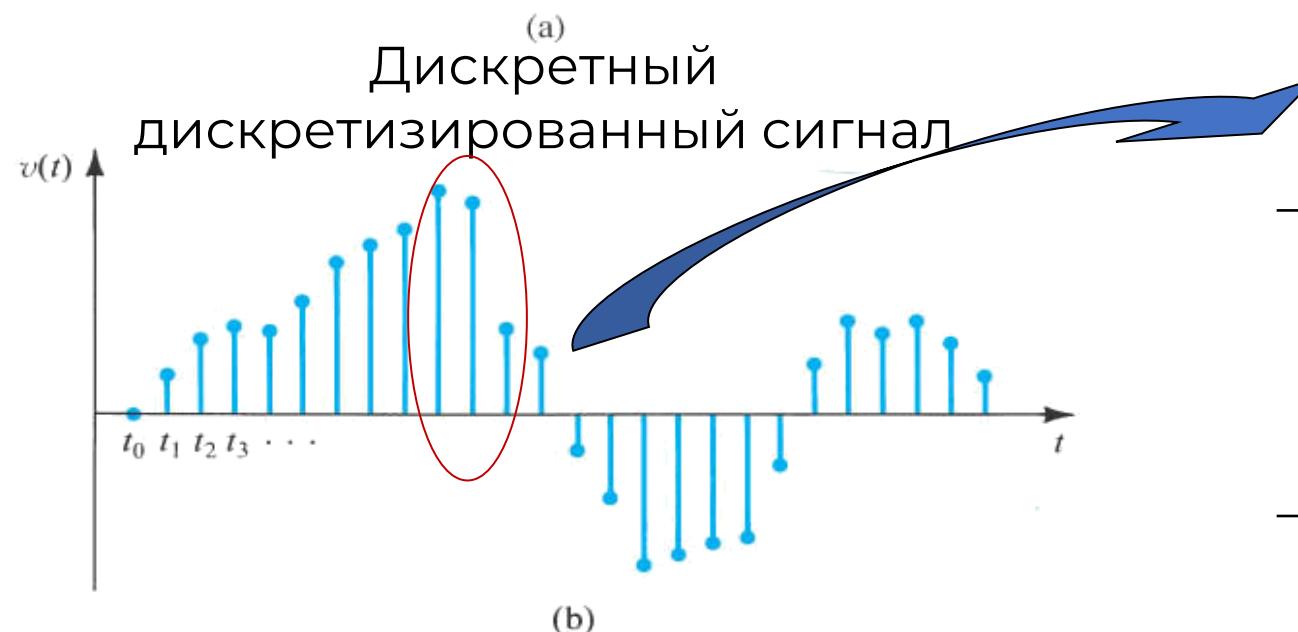
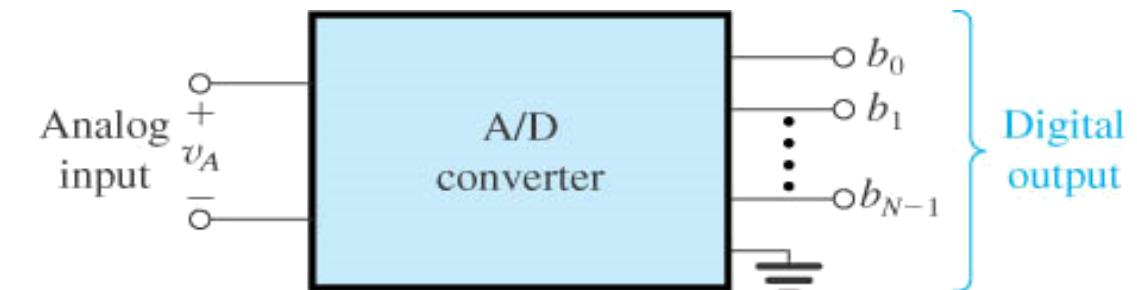


- Благодаря дискретной природе цифрового представления нет никакой двусмысленности при чтении значения цифровой величины, тогда как значение аналоговой величины часто может иметь различные интерпретации.

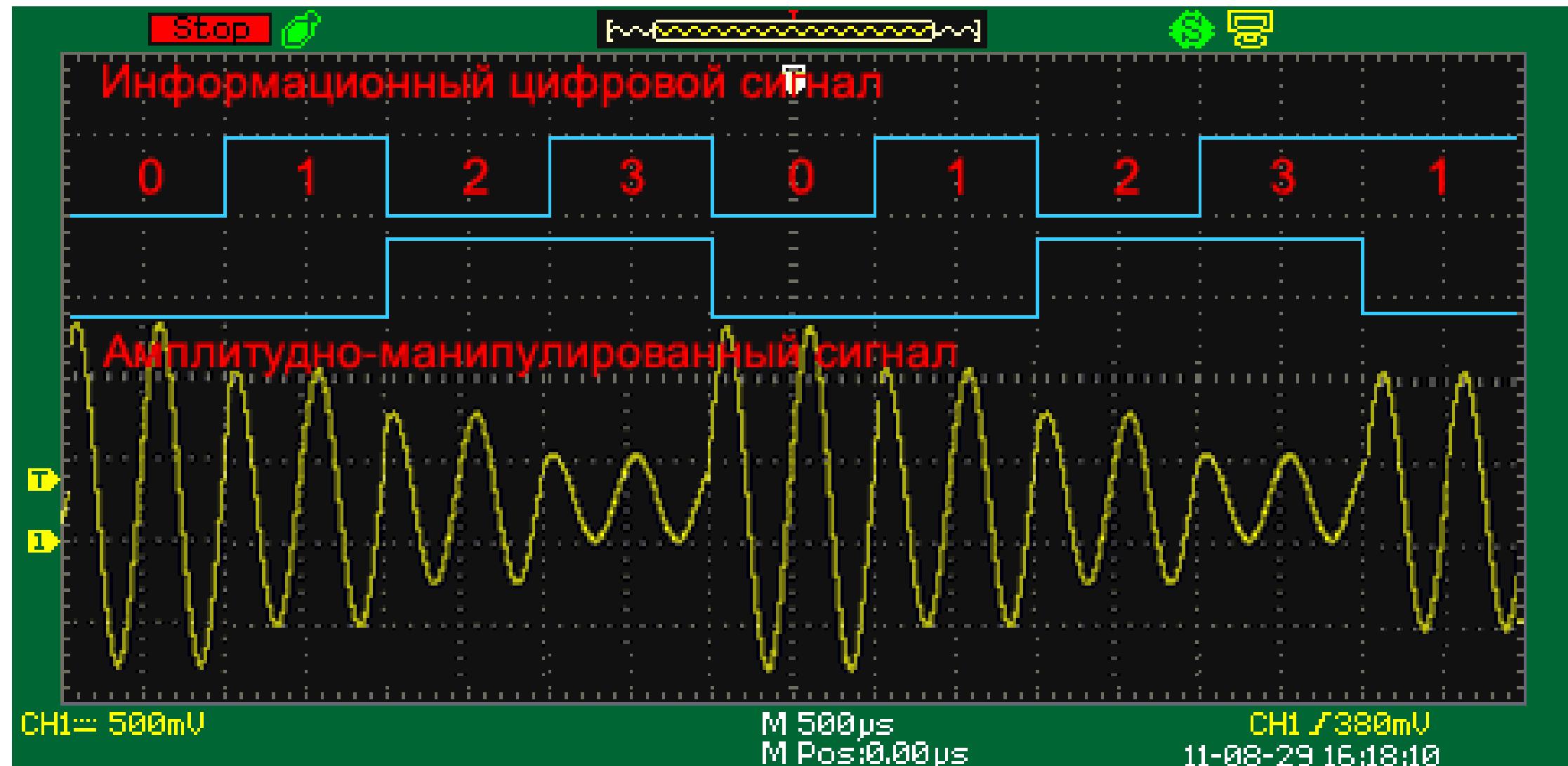
Аналоговые и цифровые сигналы



Аналого-цифровой преобразователь



Аналоговые и цифровые сигналы



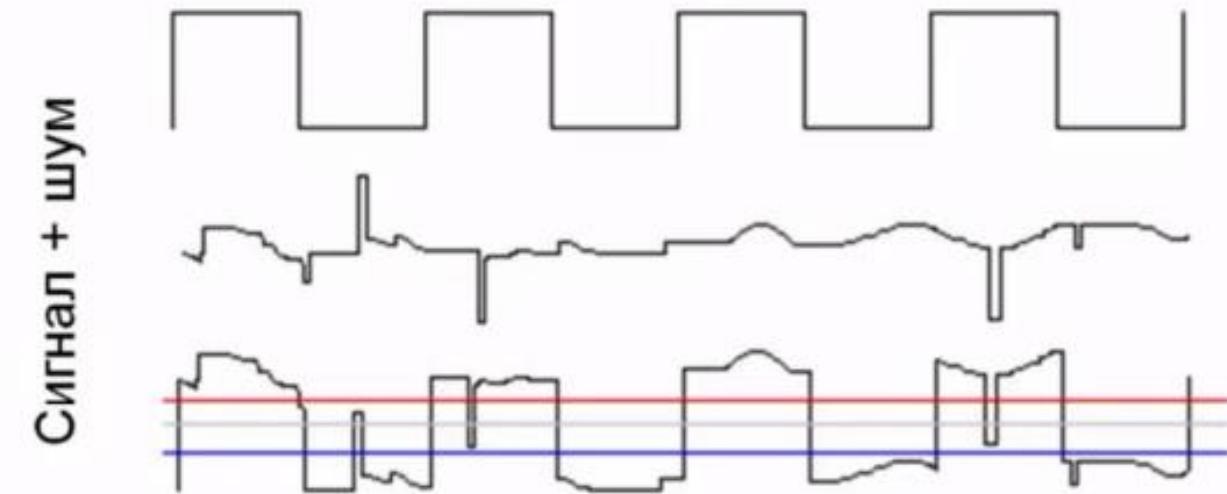
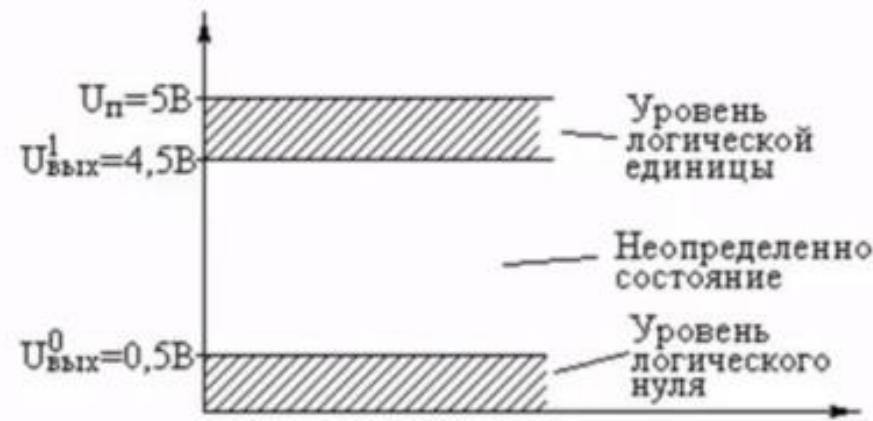
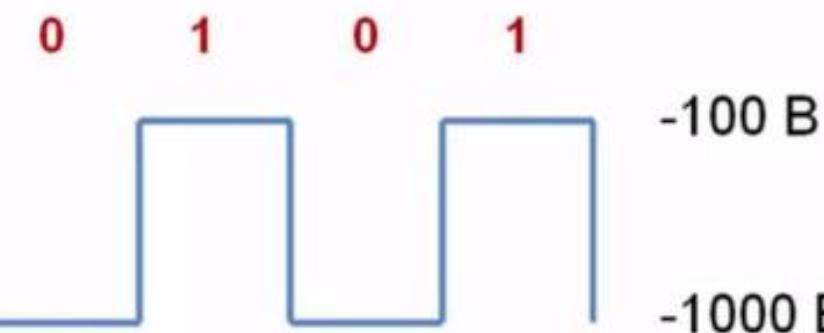
Аналоговые – изменяются плавно, принимая любые значения.

Цифровые – изменяются дискретно, принимая ограниченный набор значений.

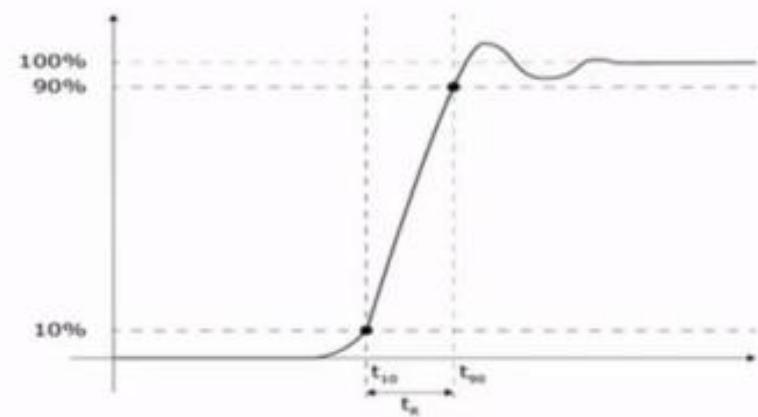
Численные представления

- **Цифровая система** — это комбинация устройств, разработанных для обработки логической информации или физических величин, которые представлены в цифровой форме. Эти устройства чаще всего электронные, но могут быть механическими, магнитными или пневматическими.
- **Наиболее распространенные цифровые системы:**
 - Цифровые компьютеры и калькуляторы;
 - Цифровое оборудование по обработке аудио- и видеоданных;
 - Телефонные системы.
 - И т.д.
- **Аналоговая система** содержит устройства, которые оперируют с физическими величинами, представленными в аналоговой форме. В аналоговой системе амплитуда выходного сигнала в колонке радиоприемника может иметь любое значение между нулем и максимальным пределом.
- **Другие обычные аналоговые системы — это:**
 - Усилители звука;
 - Устройства записи и воспроизведения на магнитной ленте;
 - Обычный плавный (реостатный) выключатель света
 - И т.д.

Цифровое представление информации

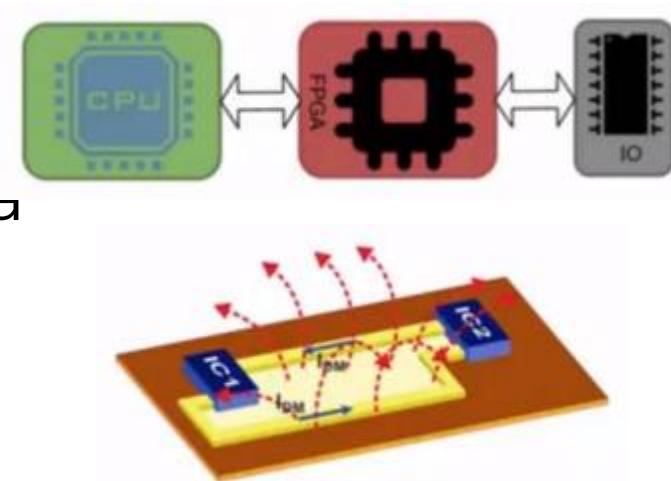
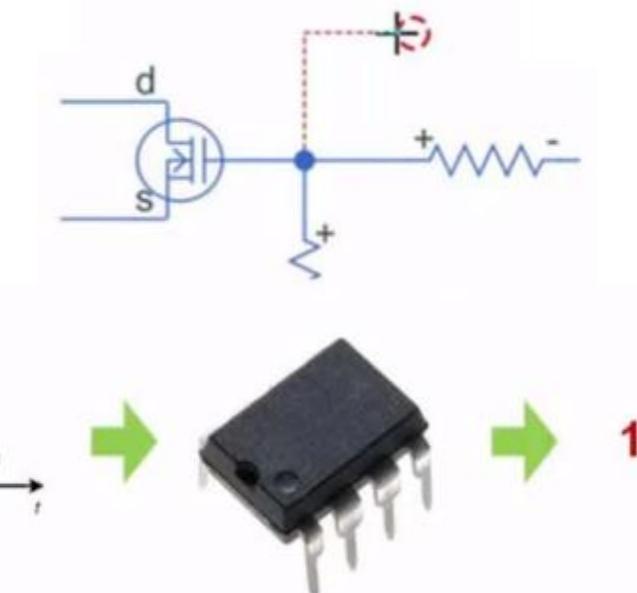
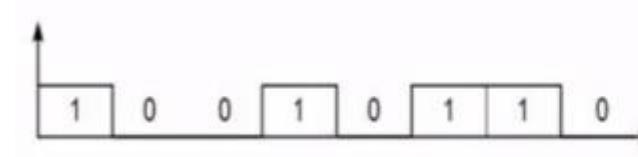
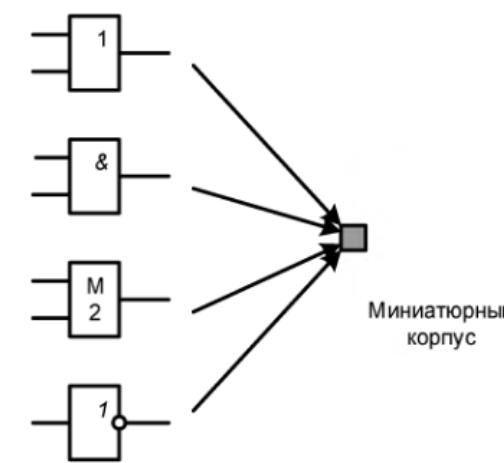
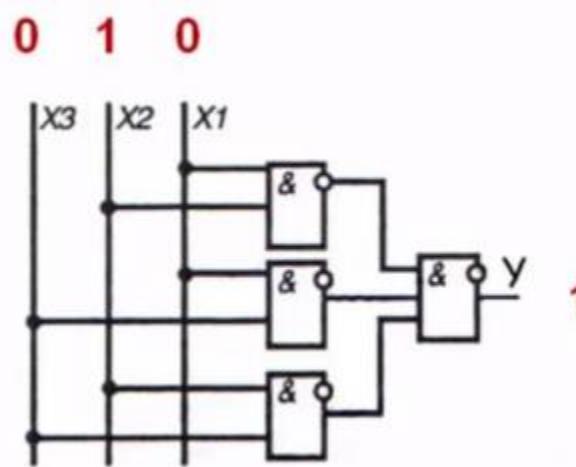


Переходный
процесс
(из 0 в 1):



Укрупненные уровни цифровой схемотехники

- **1. Системный уровень** - архитектура, интерфейсы, устройства
 - **2. Конструкторский уровень** - компоненты на плате, совместимость
 - **3. Электрическая схема** - сосредоточенные компоненты, токи и напряжения
 - **4. Логическая схема**

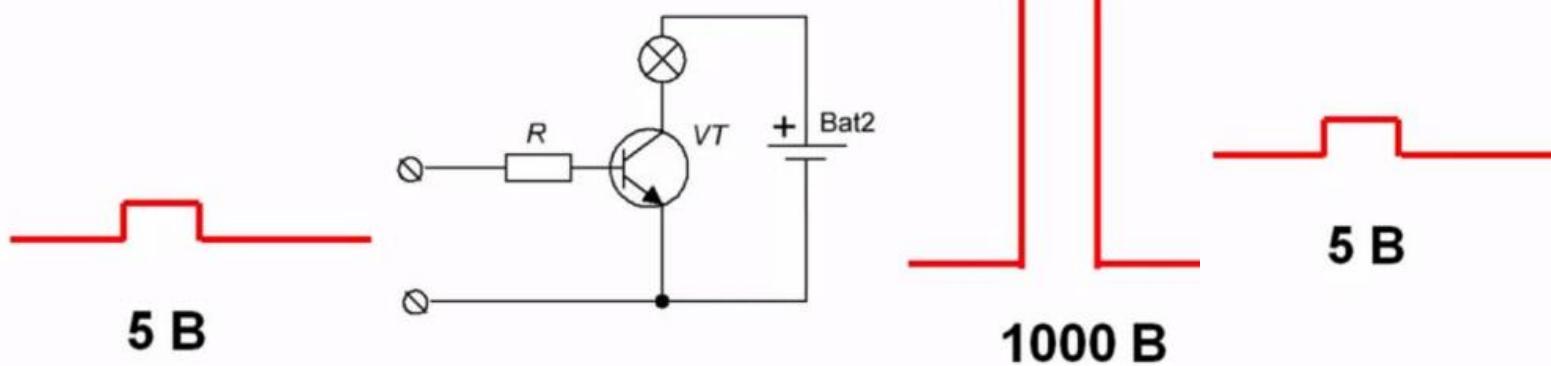


Электронные ключи

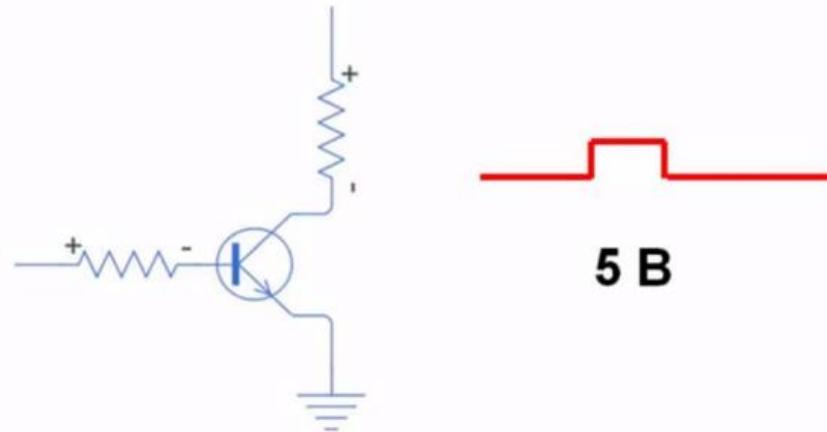
- **Ключ** — электрический коммутационный аппарат служащий для замыкания и размыкания электрической цепи.
- **Виды ключей:** Механический, Электромагнитный, Электронный



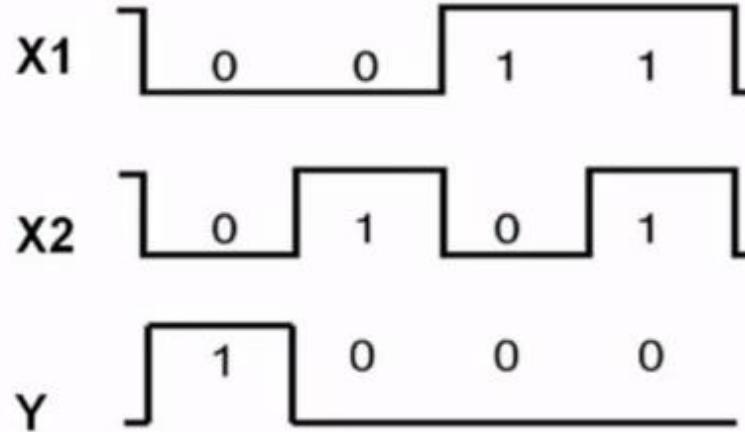
Транзисторный ключ:



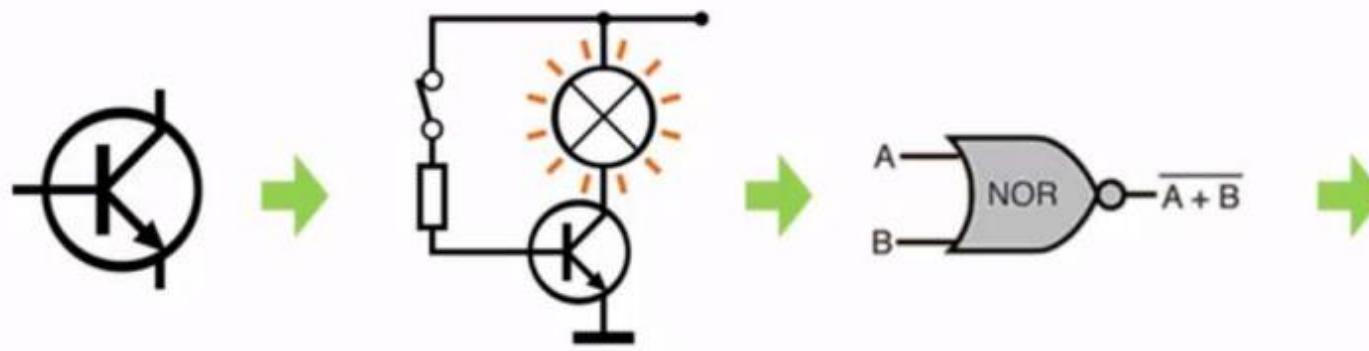
Транзисторный ключ:



Логические схемы

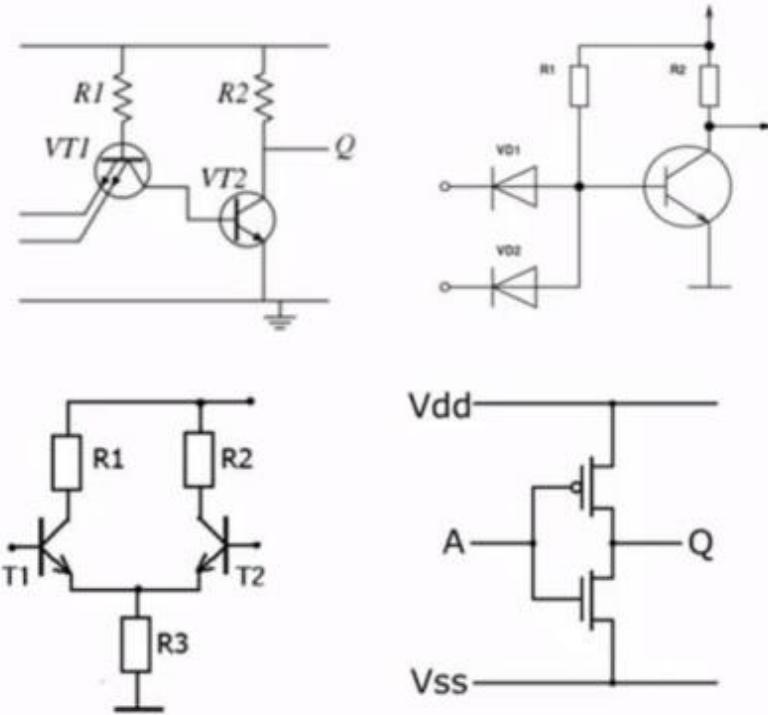


Вентиль ИЛИ-НЕ (NOR GATE)



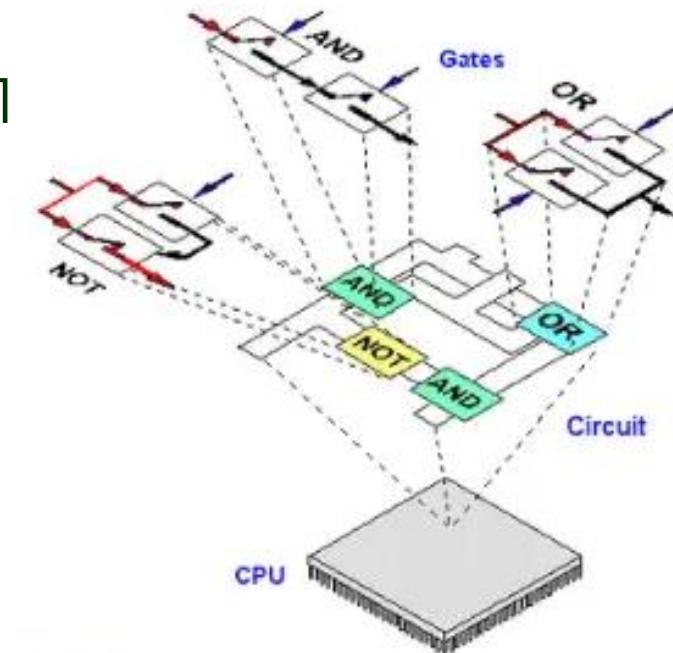
Виды логики:

- РТЛ
- ДТЛ
- ТТЛ (ТТЛШ)
- ЭСЛ
- КМОП
- проч

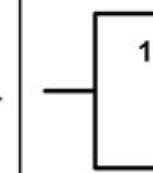
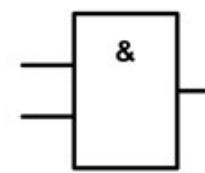
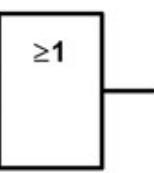
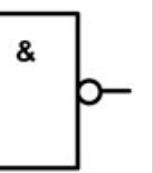
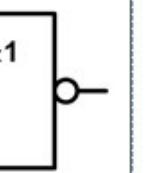
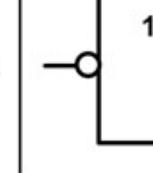
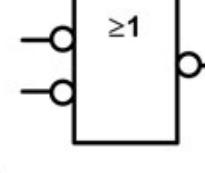
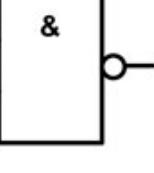
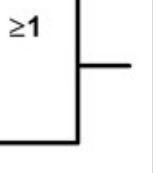
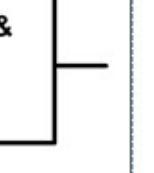
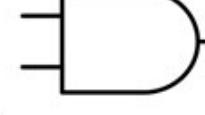
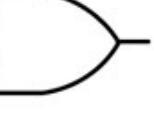
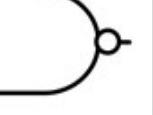
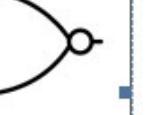
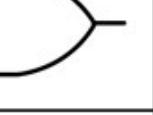
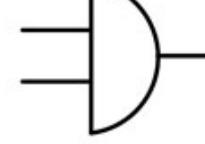
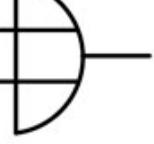
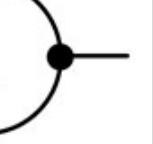
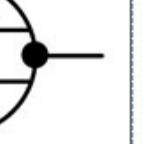
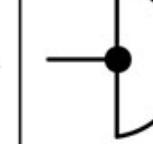
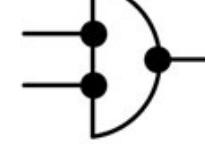
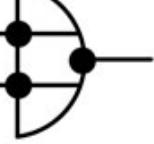
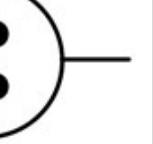
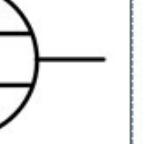


Базовые логические элементы

- **Логические элементы** — это базовые компоненты цифровых схем, реализующие элементарные логические операции.
- **Основные элементы:**
 - **И (AND)** : Выход равен 1, если все входы равны 1
 - **ИЛИ (OR)** : Выход равен 1, если хотя бы один вход равен 1.
 - **НЕ (NOT)** : Инвертирует входной сигнал.
 - **И-НЕ (NAND)** : Комбинация И и НЕ.
 - **ИЛИ-НЕ (NOR)** : Комбинация ИЛИ и НЕ.
 - **Исключающее ИЛИ (XOR)** : Выход равен 1, если входы различаются.
- Эти элементы строятся на транзисторах (например, в технологии CMOS) и используются для создания сложных схем.
- С помощью базовых логических схем (И, ИЛИ, НЕ) можно реализовать самые сложные логические функции. Использование отдельно взятого простого логического элемента оправдано лишь в том случае, если для завершения построения логики устройства не хватает буквально одного-двух логических элементов.



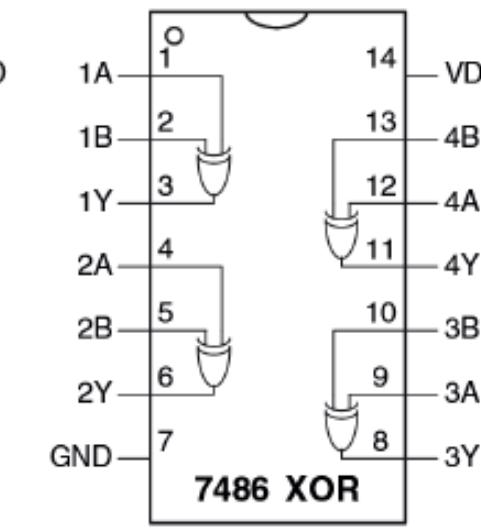
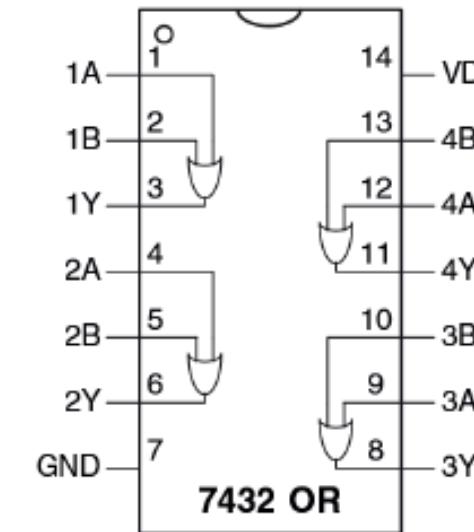
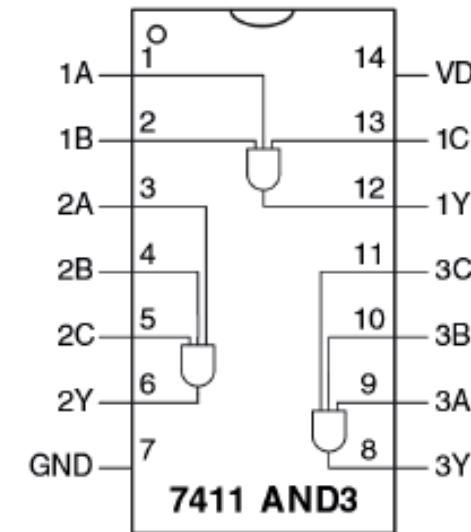
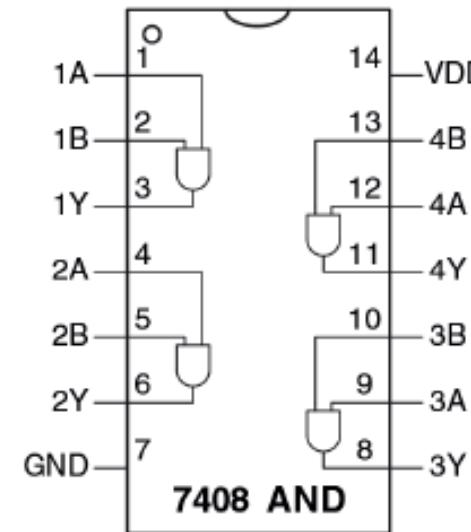
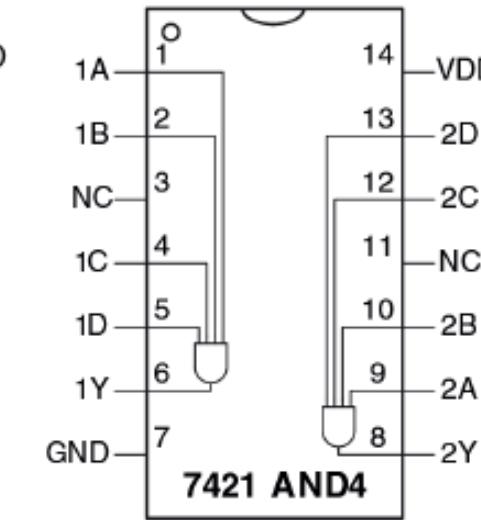
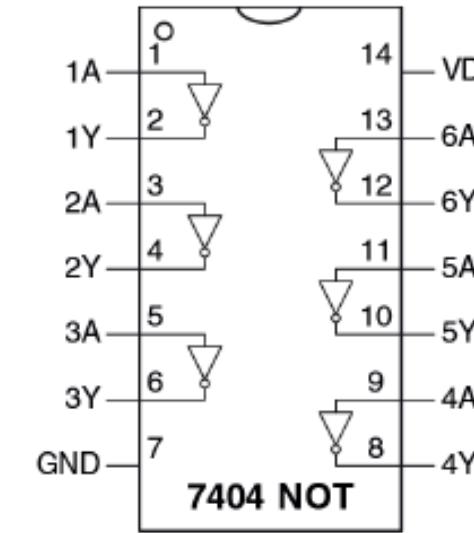
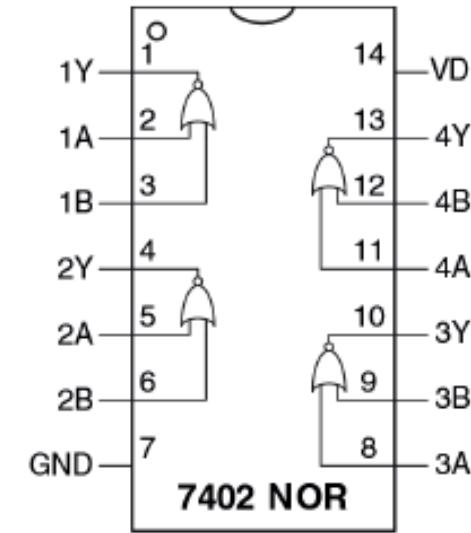
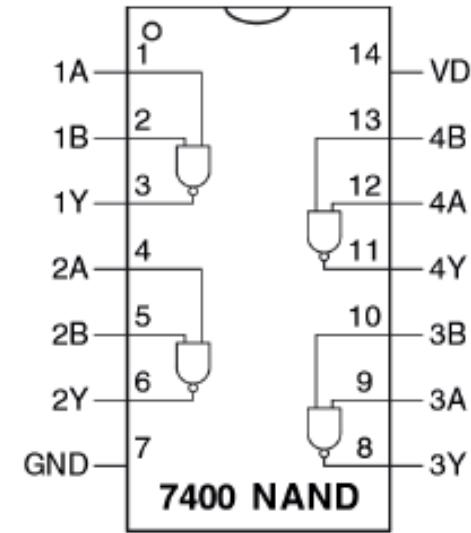
Базовые логические элементы

Логика	«НЕ»	«И»	«ИЛИ»	«И-НЕ»	«ИЛИ-НЕ»
ГОСТ и IEC	Полож. 				
	Отриц. 				
ANSI	Полож. 				
	Отриц. 				
DIN	Полож. 				
	Отриц. 				

Описание логических функций и им соответствующих логических элементов

Логическая функция	Формула	Системы обозначений логических элементов				Наименование и примеры логических элементов	Таблица истинности		
		Multisim	DIN	ANSI	ГОСТ и IEC		Входы	Выход	
		A	B	Y					
Тавтология, повторение	$Y = A$					Повторитель ТТЛ ≈7463 КМОП 4050	0 1	- -	0 1
Инверсия, отрицание	$Y = A'$					Элемент НЕ, инвертор ТТЛ 7404 КМОП 4049	0 1	- -	1 0
Дизъюнкция, логическое сложение	$Y = A + B$					Элемент 2ИЛИ ТТЛ 7432 КМОП 4081	0 0 1 1	0 1 0 1	0 1 1 1
Штрих Пирса (элемент, стрелка, функция Пирса)	$Y = A' + B'$					Элемент 2ИЛИ-НЕ ТТЛ 7402 КМОП 4001	0 0 1 1	0 1 0 1	1 0 0 0
Конъюнкция, логическое умножение	$Y = AB$					Элемент 2И, схема совпадений ТТЛ 7408 КМОП 4081	0 0 1 1	0 1 0 1	0 0 0 1
Штрих Шеффера (элемент, стрелка, функция Шеффера)	$Y = A'B'$					Элемент 2И-НЕ ТТЛ 7400 КМОП 4011	0 0 1 1	0 1 0 1	1 1 1 0
Исключающее ИЛИ (сумма по модулю два, отрицание эквивалентности)	$Y = A'B + AB'$					Элемент 2Исключающее ИЛИ ТТЛ 7486 КМОП 4070	0 0 1 1	0 1 0 1	0 1 1 0
Исключающее ИЛИ-НЕ (эквивалентность)	$Y = AB + A'B'$					Элемент 2Исключающее ИЛИ-НЕ ТТЛ 74266 КМОП 4077	0 0 1 1	0 1 0 1	1 0 0 1

Стандартные логические микросхемы серии 74xx



Стандартные логические микросхемы серии 74xx

AliExpress

Каталог

диктофон

Найти

Заказы

Корзина

Войти

Горячие товары

Топ-товары

Беларусь

RU

USD

AliExpress > ... > Детали для инструментов > Прочие детали для инструментов

В избранное

Поделиться



Набор логических микросхем 74HCxx и 74LSxx

★★★★★ 5,0 5 отзывов

15 купили

[Характеристики](#) [Описание](#)

Цвет: Индиго

Индиго

Номер модели ... Цифровой
Интегрированн...

Единица 100078581
измерения

Количество 1

[Все характеристики](#)

\$ 10.87
\$14.89 -27%
Цена за 1 штуку

[В корзину](#) [Купить сейчас](#)

Не доставляется в Минск

 LOVE HOUSE Store
82,08% рейтинг продавца - 328 подписчиков

Классификация цифровых устройств

ЦИФРОВЫЕ УСТРОЙСТВА



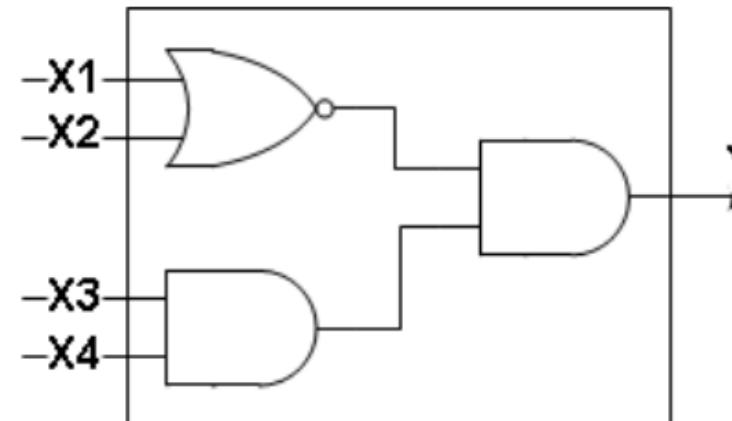
КОМБИНАЦИОННЫЕ

Выходные сигналы
определяются **только текущими
входными сигналами!**

ПОСЛЕДОВАТЕЛЬНОСТНЫЕ

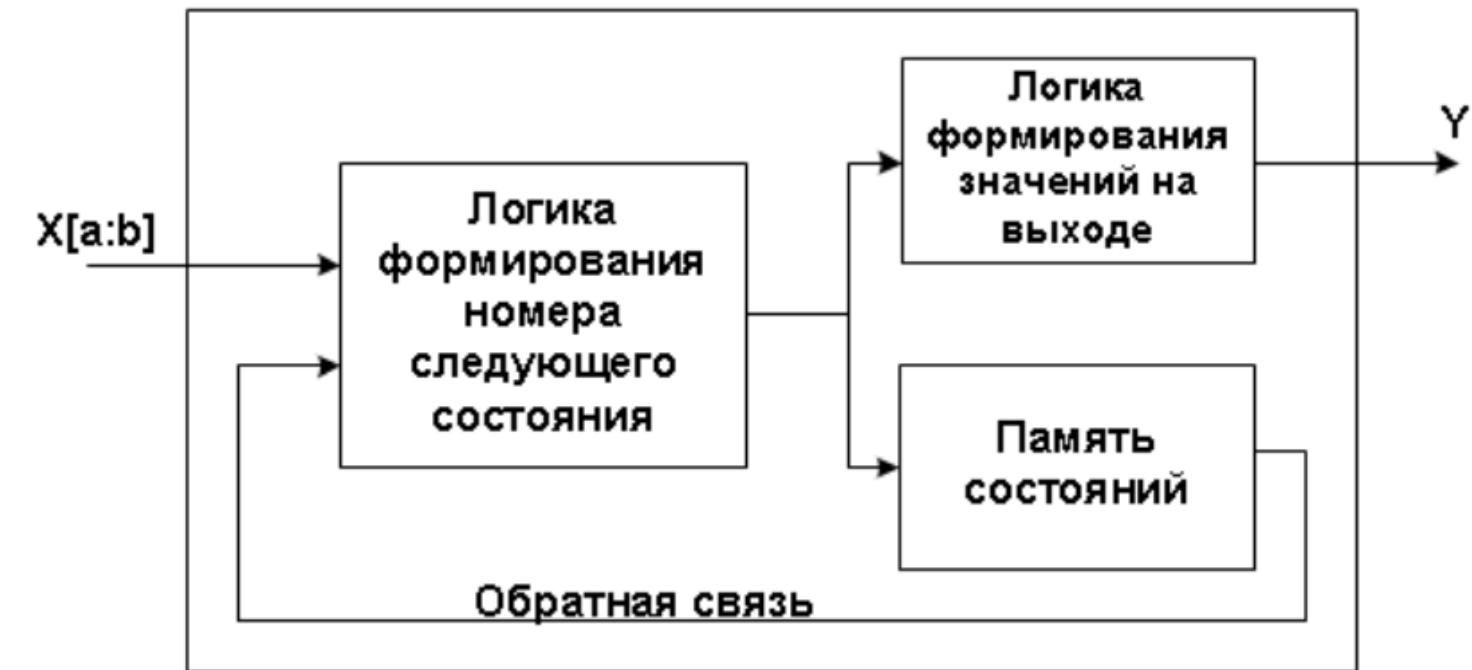
Выходные сигналы определяются не
только текущими входными сигналами, но
и входными **сигналами, действовавшими в
предыдущие моменты времени!**

Комбинационная и Последовательностная логика



$$Y = (\text{not } (X_1 \text{ or } X_2)) \text{ and } (X_3 \text{ and } X_4)$$

Пример комбинационной схемы
и ее описания



Структура последовательностной схемы –
автомата с памятью

Комбинационная логика

- **Комбинационная логика** — это тип цифровой логики, в которой состояние выходов однозначно определяется только текущим набором входных сигналов.
- **Комбинационные схемы не имеют памяти: выход зависит только от текущих входов.** Примеры:
 - **Сумматоры** :
 - **Полусумматор** : Складывает два бита, выдает сумму и перенос.
 - **Полный сумматор** : Учитывает перенос от предыдущего разряда.
 - **Мультиплексор** (MUX) : Выбирает один из нескольких входов по адресным сигналам.
 - Пример: 4-в-1 MUX использует 2 адресных линии для выбора входа.
 - **Демультиплексор** (DEMUX) : Перенаправляет один вход на один из нескольких выходов по адресу.
 - **Дешифратор** : Преобразует двоичный код в активный выход (например, 3-в-8 дешифратор для управления семисегментным индикатором).
 - **Шифратор** : Преобразует активный вход в двоичный код (например, 8-в-3 шифратор).
 - **Компаратор** : Сравнивает два числа на равенство или больше/меньше.

Последовательная (секвенциальная) логика

- В отличие от комбинационной, **последовательные схемы имеют память**: выход зависит от текущих входов и предыдущего состояния.
- **Триггеры** :
 - **SR-триггер** : Устанавливается (S) и сбрасывается (R).
 - **D-триггер** : Хранит бит данных, синхронизированный по фронту тактового сигнала.
 - **JK-триггер** : Усовершенствованный SR-триггер без запрещенного состояния.
- **Регистры** : Группа D-триггеров для хранения многоразрядных данных.
- **Счетчики** : Считывают такты (например, двоичный счетчик на JK-триггерах).
- **Конечные автоматы** : Управляют логикой на основе состояний и входов (например, светофор).

Последовательная (секвенциальная) логика

- **Последовательная логика может быть синхронной** (работающей по тактам) **или асинхронной** (без тактового сигнала).
- **Основной элемент последовательной логики – триггер**, который способен хранить один бит информации и изменять своё состояние в зависимости от входных сигналов и предыдущего состояния.

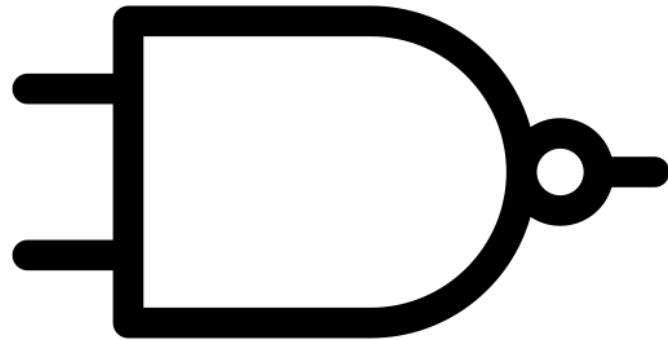
Комбинационная и Последовательная логика

- **Комбинационная логика**, применяется:
 - в **АЛУ процессора**: сложение чисел через сумматоры.
 - в **GPU**: параллельные вычисления цвета пикселей.
- **Последовательная логика**, применяется:
 - в **CPU**: регистровый файл хранит данные между тактами.
 - в **SSD**: контроллер использует конечные автоматы для управления NAND-памятью.
 - в **сетевых чипах**: буферы на триггерах для синхронизации пакетов.
- **Без комбинационной** логики невозможны вычисления, а **без последовательной** — хранение состояния и управление сложными процессами.
- Эти принципы лежат в основе всех цифровых устройств, от микроконтроллеров до суперкомпьютеров.

Отличие последовательной логики от комбинационной

- **Последовательная логика отличается от комбинационной логики тем, что ее выходные данные зависят как от текущих входных данных, так и от предыдущих состояний.**
- Это означает, что устройства последовательной логики имеют память, хранящую часть своей "истории", чтобы влиять на будущие выходные данные.

Характеристика	Комбинационная логика	Последовательная логика
Зависимость выхода	Только от текущих входов	От текущих входов и предыдущего состояния
Память	Нет	Есть
Основные элементы	Логические элементы (И, ИЛИ, НЕ и др.)	Триггеры, регистры, счетчики и др.
Примеры устройств	Сумматор, дешифратор, мультиплексор	Счетчик, регистр, конечный автомат



Комбинационная логика. Булева алгебра



Комбинационная логика

- **Комбинационная логика (комбинационная схема)** в теории цифровых устройств — двоичная логика функционирования устройств комбинационного типа.
- **У комбинационных устройств состояние выхода однозначно определяется набором входных сигналов**, что отличает комбинационную логику от секвенциальной логики (последовательная логика), в рамках которой выходное значение зависит не только от текущего входного воздействия, но и от предыстории функционирования цифрового устройства.
- Другими словами, секвенциальная логика (последовательная логика) предполагает наличие памяти, которая в комбинационной логике не предусмотрена.

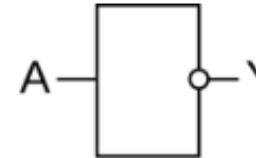
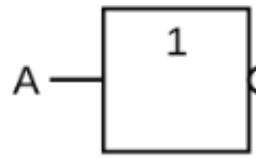
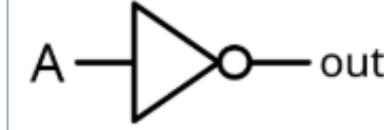
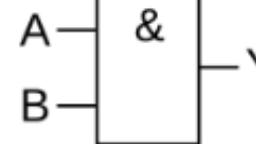
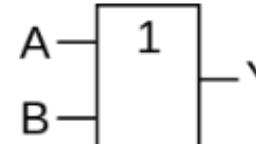
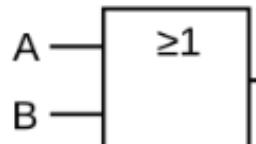
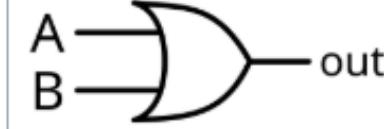
Комбинационная логика

- **Комбинационная логика используется в вычислительных цепях для формирования входных сигналов и для подготовки данных, которые в последующем подлежат сохранению.**
- На практике вычислительные устройства обычно сочетают комбинационную и секвенциальную логику (последовательную логику).
- Например, арифметическое логическое устройство (АЛУ) содержит комбинационные узлы.
- **Математику комбинационной логики обеспечивает булева алгебра.**
- Базовыми операциями являются: конъюнкция, дизъюнкция, отрицание (инверсия)

Комбинационная логика

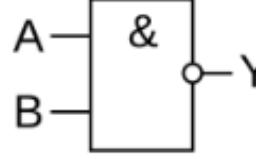
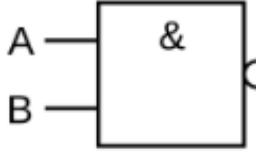
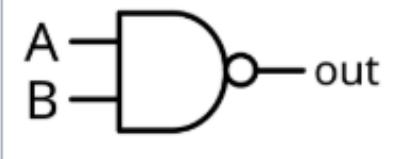
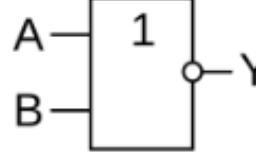
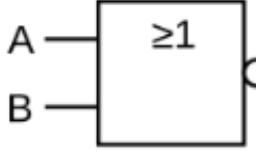
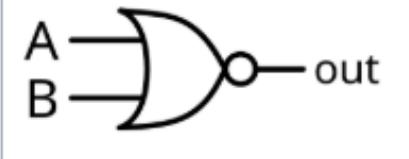
- В комбинационных схемах используются логические элементы (логические вентили):
 - конъюнктор (И);
 - дизъюнктор (ИЛИ);
 - инвертор (НЕ),
- а также производные элементы:
 - И-НЕ;
 - ИЛИ-НЕ;
 - Исключающее ИЛИ
 - Эквивалентность (исключающее ИЛИ-НЕ).
- Любой сколь угодно сложный элемент компьютера может быть сконструирован из элементарных логических элементов.
- Наиболее известными комбинационными устройствами являются сумматор, полусумматор, шифратор, дешифратор, мультиплексор и демультиплексор.

Логические элементы (логические вентили)

Логический вентиль	Условные графические обозначения			Функция, запись	Таблица истинности															
	ГОСТ 2.743-91	IEC 60617-12 : 1997	US ANSI 91-1984																	
НЕ (англ. NOT gate)				Отрицание $Y = \bar{A}$ $Y = \neg A$ $Y = \tilde{A}$	<table border="1" data-bbox="2086 338 2291 568"> <tr> <td>A</td> <td>Y</td> </tr> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </table>	A	Y	0	1	1	0									
A	Y																			
0	1																			
1	0																			
И (англ. AND gate)				Конъюнкция $Y = A \wedge B$ $Y = A \cdot B$ $Y = A \& B$ $Y = AB$	<table border="1" data-bbox="2060 597 2342 972"> <tr> <td>A</td> <td>B</td> <td>Y</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </table>	A	B	Y	0	0	0	0	1	0	1	0	0	1	1	1
A	B	Y																		
0	0	0																		
0	1	0																		
1	0	0																		
1	1	1																		
ИЛИ (англ. OR gate)				Дизъюнкция $Y = A \vee B$ $Y = A + B$	<table border="1" data-bbox="2060 986 2342 1360"> <tr> <td>A</td> <td>B</td> <td>Y</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	1
A	B	Y																		
0	0	0																		
0	1	1																		
1	0	1																		
1	1	1																		

Логический вентиль https://ru.wikipedia.org/wiki/Логический_вентиль

Логические элементы (логические вентили)

Логический вентиль	Условные графические обозначения			Функция, запись	Таблица истинности															
	ГОСТ 2.743-91	IEC 60617-12 : 1997	US ANSI 91-1984																	
НЕ И (И-НЕ) (англ. <i>NAND gate</i>) Элемент Шеффера				$Y = \overline{A \wedge B}$ $Y = A \bar{\wedge} B$ $Y = \overline{A \cdot B}$ $Y = \overline{AB}$ $Y = A B$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	1	1	0	1	1	1	0
A	B	Y																		
0	0	1																		
0	1	1																		
1	0	1																		
1	1	0																		
НЕ ИЛИ (ИЛИ-НЕ) (англ. <i>NOR gate</i>) Элемент Пирса				$Y = \overline{A \vee B}$ $Y = A \bar{\vee} B$ $Y = \overline{A + B}$ $Y = A - B$	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	0
A	B	Y																		
0	0	1																		
0	1	0																		
1	0	0																		
1	1	0																		

Логический вентиль https://ru.wikipedia.org/wiki/Логический_вентиль

Логические элементы (логические вентили)

Логический вентиль	Условные графические обозначения			Функция, запись	Таблица истинности															
	ГОСТ 2.743-91	IEC 60617-12 : 1997	US ANSI 91-1984																	
Исключающее ИЛИ (англ. <i>XOR gate</i>) сложение по модулю 2				Строгая дизъюнкция $Y = A \vee B$ $Y = A \oplus B$	<table border="1"><thead><tr><th>A</th><th>B</th><th>Y</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></tbody></table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	0
A	B	Y																		
0	0	0																		
0	1	1																		
1	0	1																		
1	1	0																		
Исключающее ИЛИ с инверсией (англ. <i>XNOR gate</i>) равнозначность				Эквиваленция $Y = \overline{A \vee B}$ $Y = A \overline{\vee} B$ $Y = \overline{A \oplus B}$ $Y = A \odot B$	<table border="1"><thead><tr><th>A</th><th>B</th><th>Y</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></tbody></table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	1
A	B	Y																		
0	0	1																		
0	1	0																		
1	0	0																		
1	1	1																		

Логический вентиль https://ru.wikipedia.org/wiki/Логический_вентиль

Булева алгебра

- **Булева алгебра** — это раздел математики, занимающийся изучением операций с логическими значениями истинности.
- Она основана на системе, определяемой значениями «**истина**» и «**ложь**», обычно обозначаемыми как **1** и **0** соответственно.
- **Основные операции булевой алгебры логики включают конъюнкцию (И), дизъюнкцию (ИЛИ) и отрицание (НЕ).**
- **Данные операции лежат в основе цифровой логики** и имеют широкое применение в областях, таких как компьютерные науки, цифровая электроника и теория множеств.

Логические операции и их обозначения

Операция	Обозначение	Речевой оборот
Отрицание (инверсия, логическое НЕ)	$\neg A$, \bar{A} , НЕ A , not A	«Не», «неверно, что»
Конъюнкция (логическое умножение, логическое И)	$A \wedge B$, $A \& B$, $A \cdot B$, AB , A И B , A and B	«И», «как ..., так И», «вместе с», «НО», «хотя», «а»
Дизъюнкция (логическое сложение, логическое ИЛИ)	$A \vee B$, $A + B$, $A B$, A ИЛИ B , A or B	«Или», «или ... или ... или оба вместе»
Строгая дизъюнкция (исключающая дизъюнкция, исключающее ИЛИ)	$A \oplus B$, A xor B	«Либо ..., либо», «только ... или только»
Импликация (логическое следование)	$A \rightarrow B$, $A \Rightarrow B$	«Если ..., то», «из ... следует», «влечёт»
Эквиваленция (эквивалентность, равнозначность)	$A \leftrightarrow B$, $A \Leftrightarrow B$, $A \equiv B$	«Эквивалентно», «равносильно», «необходимо и достаточно», «тогда и только тогда, когда»

Логические функции

A	B	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

$F_1(A, B) = 0$ — константа «ложь»;

$F_2(A, B) = A \& B$ — конъюнкция;

$F_3(A, B) = \overline{A \rightarrow B}$ — отрицание импликации;

$F_4(A, B) = A$ — функция, равная первому аргументу;

$F_5(A, B) = \overline{B \rightarrow A}$ — отрицание обратной импликации;

$F_6(A, B) = B$ — функция, равная второму аргументу;

$F_7(A, B) = A \oplus B$ — строгая дизъюнкция;

$F_8(A, B) = A \vee B$ — дизъюнкция;

Логические функции

A	B	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

$F_9(A, B) = A \downarrow B$ — стрелка Пирса (отрицание дизъюнкции, ИЛИ-НЕ);

$F_{10}(A, B) = A \leftrightarrow B$ — эквиваленция;

$F_{11}(A, B) = \overline{B}$ — отрицание второго аргумента;

$F_{12}(A, B) = B \rightarrow A$ — обратная импликация;

$F_{13}(A, B) = \overline{A}$ — отрицание первого аргумента;

$F_{14}(A, B) = A \rightarrow B$ — импликация;

$F_{15}(A, B) = A | B$ — штрих Шеффера (отрицание конъюнкции, И-НЕ);

$F_{16}(A, B) = 1$ — константа «истина».

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	$A \cdot 0 = 0, A \cdot 1 = A$	$A + 0 = A, A + 1 = 1$
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	$A + B \cdot C = (A + B) \cdot (A + C)$	$A \cdot (B + C) = A \cdot B + A \cdot C$
законы де Моргана	$\overline{A \cdot B} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A} \cdot \overline{B}$

Синтез логических схем по заданной функции

Представление логических функций (ЛФ)

3 способа представления логических функций:

1. графиком (в виде временной диаграммы напряжения);
2. аналитическим (булевым выражением);
3. таблицей истинности.

В аналитическом виде ЛФ может быть представлена различными сочетаниями операций сложения и умножения переменных. Однако наиболее удобно представлять ЛФ в двух формах записи:

- 1) как суммы произведений переменных:

$$Y + \overline{X}Y + X\overline{Y}Z + \overline{X}Y\overline{Z}.$$

Такая запись функции называется **дизъюнктивной нормальной формой (ДНФ)**.

- 2) как произведения сумм переменных:

$$Y(\overline{X} + Y)(\overline{X} + Y + Z)(X + \overline{Y} + Z)$$

Такая запись функции называется **конъюнктивной нормальной формой (КНФ)**.

Представление логических функций (ЛФ)

Переход от одной формы записи функции к другой осуществляется инверсией функции по теореме де Моргана. Например, логическая функция дана в ДНФ:

$$F = Y + \overline{X}Z + X\overline{Y}Z.$$

Инвертируем и получаем функцию в КНФ:

$$\overline{F} = \overline{Y + \overline{X}Z + X\overline{Y}Z} = (\overline{Y})(\overline{\overline{X}Z})(\overline{X\overline{Y}Z}) = \overline{Y}(X + \overline{Z})(\overline{X} + Y + \overline{Z}).$$

$$\overline{X + Y} = \overline{X} \cdot \overline{Y}$$

$$\overline{X \cdot Y} = \overline{X} + \overline{Y}$$

Пользуясь законами логики, можно любую ЛФ преобразовать к ДНФ и КНФ. Для одной и той же ЛФ может существовать несколько равносильных дизъюнктивных и конъюнктивных форм. Однако существует только один вид ДНФ и КНФ, в которых функция может быть записана единственным образом – это **совершенные нормальные формы**.

В **совершенной дизъюнктивной нормальной форме (СДНФ)** каждое слагаемое содержит произведение всех переменных и/или их отрицаний и нет одинаковых слагаемых.

В **совершенной конъюнктивной нормальной форме (СКНФ)** каждый сомножитель содержит суммы всех переменных и/или их отрицаний и нет одинаковых сомножителей.

Наиболее наглядно и полно логическая функция представляется таблицей истинности. Переход от аналитического выражения ЛФ к таблице истинности осуществляется определением значения функции для всех вариантов сочетания значений переменных функции (т.е. **методом перебора**).

Представление логических функций (ЛФ)

№ комб.	X	Y	Z	F
1	0	0	0	0
2	0	0	1	1
3	0	1	0	1
4	0	1	1	0
5	1	0	0	0
6	1	0	1	0
7	1	1	0	1
8	1	1	1	0

Рассмотрим на примере переход от табличной формы представления функции к аналитической записи ее в СДНФ или СКНФ.

Пусть функция задана таблицей истинности.

Функция $F=1$ (истинна) в комбинациях переменных 2,3,7.

$$\overline{X}\overline{Y}Z = 1, \quad \overline{X}Y\overline{Z} = 1, \quad XY\overline{Z} = 1$$

Комбинации переменных, при которых функция истинна называют **минтермами**.

Функция в СДНФ есть сумма комбинаций переменных, при которых функция истинна, т.е.

$$F = \overline{X}\overline{Y}Z + \overline{X}Y\overline{Z} + XY\overline{Z}$$

Функцию можно представить не только единичными, но и нулевыми значениями.

Функция $F = 0$ (ложна) или $\overline{F} = 1$, если

$$\overline{X}\overline{Y}\overline{Z} = 0, \quad \overline{X}YZ = 0, \quad X\overline{Y}\overline{Z} = 0, \quad X\overline{Y}Z = 0, \quad XYZ = 0.$$

$$\overline{F} = \overline{X}\overline{Y}\overline{Z} + \overline{X}YZ + X\overline{Y}\overline{Z} + X\overline{Y}Z + XYZ.$$

Воспользовавшись теоремой де Моргана, получаем функцию в СКНФ:

$$F = (X + Y + Z)(X + \overline{Y} + Z)(\overline{X} + Y + Z)(\overline{X} + Y + \overline{Z})(\overline{X} + \overline{Y} + \overline{Z}).$$

Комбинации переменных, при которых функция ложна называют **макстермами**.

Методы минимизации логических функций

- **Минимизация** – упрощение формы записи логической функции, направленное на устранение избыточности в записи функции.
- При синтезе логических схем минимизированная функция реализуется с наименьшим числом логических элементов.
- **Минимизация производится:**
 - **1. алгебраическими способами;**
 - **2. методом карт Карно.**

1. Использование законов булевой алгебры:

- добавление существующих слагаемых: $X + X + X = X$;
- умножение отдельных слагаемых на функцию вида: $X + \bar{X} = 1$;
- выделение слагаемых вида: $X + \bar{X} = 1$.

Пример.

$$\begin{aligned}\overline{XYZ} + X\overline{Y}Z + XY\overline{Z} + XYZ &= \underline{\overline{XYZ}} + \underline{X\overline{Y}Z} + \underline{XY\overline{Z}} + \underline{XYZ} + \underline{XYZ} + \underline{\underline{XYZ}} = \\ &= YZ(\overline{X}+X) + XZ(\overline{Y}+Y) + XY(\overline{Z}+Z) = YZ + XZ + XY.\end{aligned}$$

Преобразование алгебраическими способами требует громоздких математических выкладок и больших временных затрат. Существуют приемы, основанные на правилах алгебры логики, позволяющие минимизировать функцию более быстро и просто (и даже безошибочно!)

Метод упрощения логических функций

- **Метод карт Карно (диаграмм Карно) (или диаграммы Вейча)** — служит для наглядного представления и упрощения нормальной формы ИЛИ. Карты Карно представляют таблицы истинности для полных конъюнкций. Единица в поле карты соответствует наличию полной конъюнкции.
- Метод используется для минимизаций функций с числом переменных до 5-6.
- Карты Карно представляют собой таблицу, в которую заносятся значения всех возможных комбинаций переменных

Количество полей в таблице составляет 2^n , где n – количество переменных.

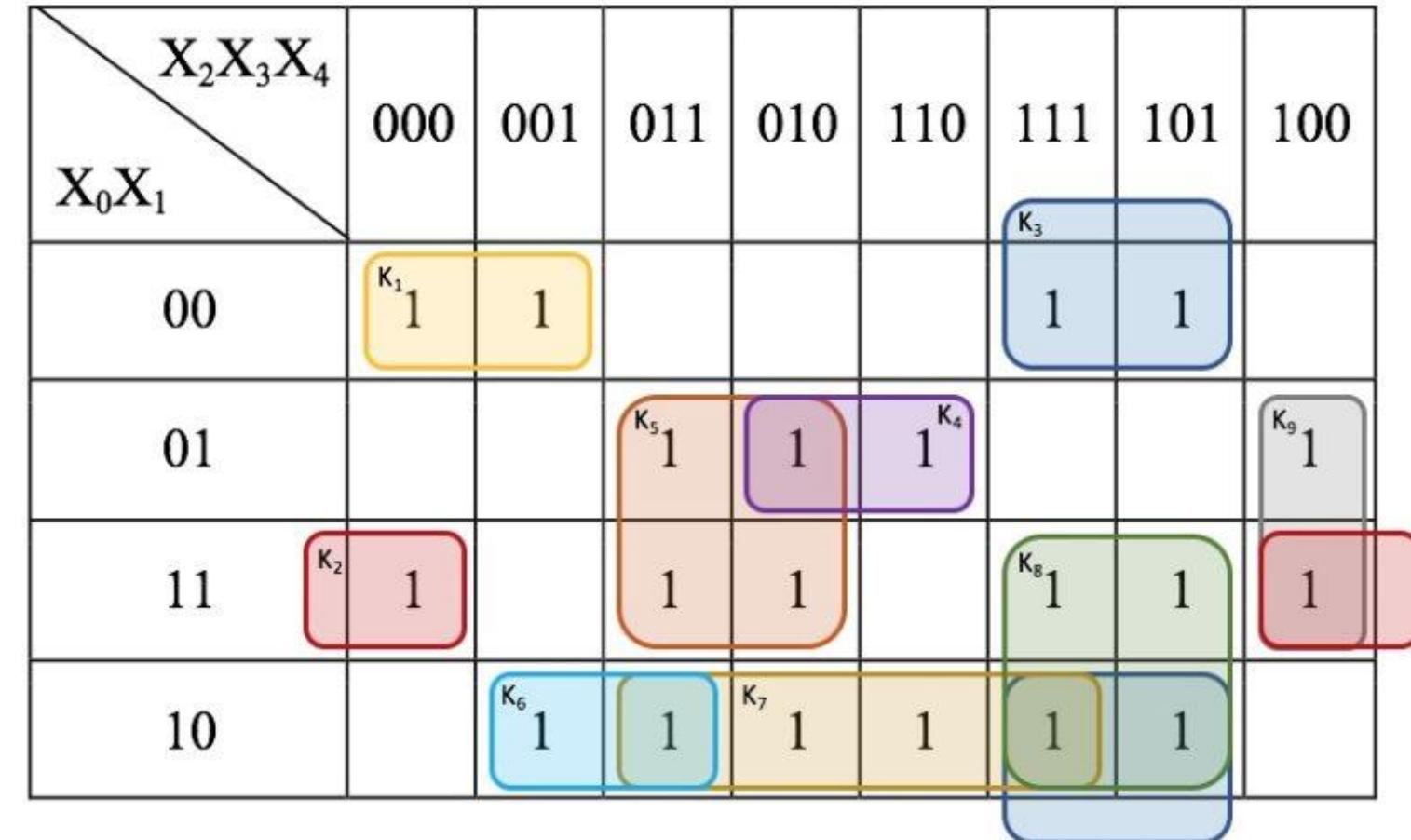
«1» - прямое значение переменной;
«0» - инверсное значение переменной.

		A	
		0	1
B 0	0	$\bar{A}\bar{B}$	$A\bar{B}$
	1	$\bar{A}B$	AB

Карта Карно из двух переменных

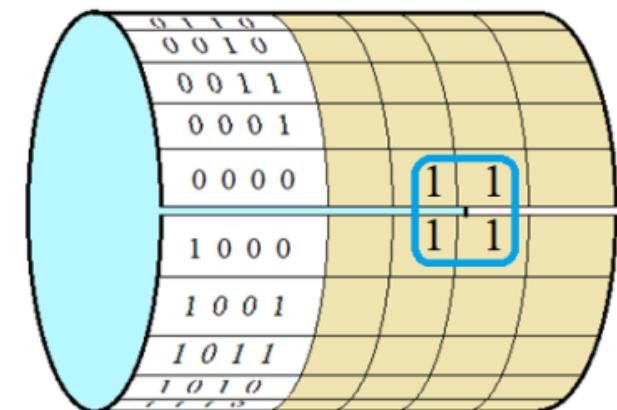
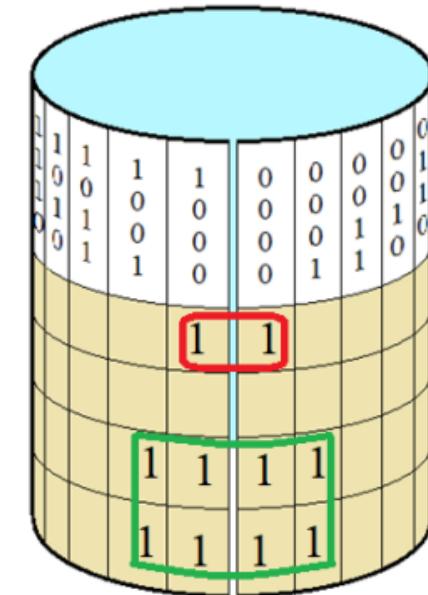
Карты Карно для минимизации логических функций

- **Карта Карно** — это графический метод минимизации логических функций, который упрощает процесс проектирования логики.
- Карта Карно представляет собой таблицу, в которой по осям указаны все возможные комбинации входных значений, а в ячейках таблицы — соответствующие им выходные значения.
- Каждая клетка карты представляет собой возможное состояние входных переменных.
- Группируя соседние клетки с одинаковыми значениями (например, с единицами), можно минимизировать количество логических операций.
- Чем меньше групп, тем более эффективной будет схема.



Правила записи карты Карно

- **1.** Разметка вертикальной оси не зависит от разметки горизонтальной.
- **2.** Разметку осей можно начинать с любого сочетания переменных.
- **3.** По каждой оси должны быть перечислены все сочетания переменных.
- **4.** Карта составляется таким образом, чтобы соседние клетки отличались только одной переменной. Соседними клетками также считаются крайние клетки каждого столбца или строки.



Фактическая модель карты Карно

Таблица истинности и карта Карно для функции двух переменных

a	b	f(a,b)
0	0	f(0,0)
0	1	f(0,1)
1	0	f(1,0)
1	1	f(1,1)

		b	
		1	0
a	0	f(0,1)	f(0,0)
	1	f(1,1)	f(1,0)

Для функции двух переменных карта Карно — это квадрат 2x2 клетки.
В этих клетках размещаются 4 значения функции из последнего столбца таблицы истинности.

Таблица истинности и карта Карно для функции трех переменных

a	b	c	f(a,b,c)
0	0	0	$f(0,0,0)$
0	0	1	$f(0,0,1)$
0	1	0	$f(0,1,0)$
0	1	1	$f(0,1,1)$
1	0	0	$f(1,0,0)$
1	0	1	$f(1,0,1)$
1	1	0	$f(1,1,0)$
1	1	1	$f(1,1,1)$

		c	
		1	0
ab	00	$f(0,0,1)$	$f(0,0,0)$
	01	$f(0,1,1)$	$f(0,1,0)$
	11	$f(1,1,1)$	$f(1,1,0)$
	10	$f(1,0,1)$	$f(1,0,0)$

		bc			
		01	00	11	10
a	0	$f(0,0,1)$	$f(0,0,0)$	$f(0,1,1)$	$f(0,1,0)$
	1	$f(1,0,1)$	$f(1,0,0)$	$f(1,1,1)$	$f(1,1,0)$

		c	
		1	0
ab	11	$f(1,1,1)$	$f(1,1,0)$
	01	$f(0,1,1)$	$f(0,1,0)$
	00	$f(0,0,1)$	$f(0,0,0)$
	10	$f(1,0,1)$	$f(1,0,0)$

		bc			
		10	00	11	01
a	0	$f(0,1,0)$	$f(0,0,0)$	$f(0,1,1)$	$f(0,0,1)$
	1	$f(1,1,0)$	$f(1,0,0)$	$f(1,1,1)$	$f(1,0,1)$

неправильное заполнение

Для функции трех переменных карта Карно - это прямоугольник 2×4 или 4×2 клетки.

В этих клетках размещаются 8 значений функции из последнего столбца таблицы истинности.

При разметке большей из осей нужно четко придерживаться последнего - четвертого – правила разметки и следить за тем, чтобы соседними не оказались сочетания 00 и 11 либо 01 и 10, в которых одновременно меняются обе переменные.

Таблица истинности и карта Карно для функции четырех переменных

a	b	c	d	f(a,b,c,d)
0	0	0	0	f(0,0,0,0)
0	0	0	1	f(0,0,0,1)
0	0	1	0	f(0,0,1,0)
0	0	1	1	f(0,0,1,1)
0	1	0	0	f(0,1,0,0)
0	1	0	1	f(0,1,0,1)
0	1	1	0	f(0,1,1,0)
0	1	1	1	f(0,1,1,1)
1	0	0	0	f(1,0,0,0)
1	0	0	1	f(1,0,0,1)
1	0	1	0	f(1,0,1,0)
1	0	1	1	f(1,0,1,1)
1	1	0	0	f(1,1,0,0)
1	1	0	1	f(1,1,0,1)
1	1	1	0	f(1,1,1,0)
1	1	1	1	f(1,1,1,1)

ab	cd			
	11	01	00	10
00	f(0,0,1,1)	f(0,0,0,1)	f(0,0,0,0)	f(0,0,1,0)
01	f(0,1,1,1)	f(0,1,0,1)	f(0,1,0,0)	f(0,1,1,0)
11	f(1,1,1,1)	f(1,1,0,1)	f(1,1,0,0)	f(1,1,1,0)
10	f(1,0,1,1)	f(1,0,0,1)	f(1,0,0,0)	f(1,0,1,0)

ab	cd			
	11	01	00	10
00	f(0,0,1,1)	f(0,0,0,1)	f(0,0,0,0)	f(0,0,1,0)
10	f(1,0,1,1)	f(1,0,0,1)	f(1,0,0,0)	f(1,0,1,0)
ab	f(1,1,1,1)	f(1,1,0,1)	f(1,1,0,0)	f(1,1,1,0)
01	f(0,1,1,1)	f(0,1,0,1)	f(0,1,0,0)	f(0,1,1,0)

Таблица истинности и карта Карно для функции четырех переменных

Для функции четырех переменных карта Карно - это квадрат 4x4 клетки.

В этих клетках размещаются 16 значений функции из последнего столбца таблицы истинности.

При разметке обеих осей нужно также четко придерживаться последнего - четвертого - правила разметки и следить за тем, чтобы по одной оси соседними не оказались сочетания 00 и 11 либо 01 и 10, в которых одновременно меняются обе переменные.

Методы минимизации логических функций

Пример 1. $F = AB + A\bar{B}$

	\bar{A}	A
\bar{B}	0	1
B	0	1

Для каждого сочетания переменных AB в соответствующую ячейку пишется «1». В незаполненные клетки – «0». Соседние единицы объединяем в один контур по 2 или 4 или 8 единиц.

Каждый контур – член упрощенного булева выражения.

В примере имеется 1 контур. Это означает, что новое минимизированное выражение будет состоять из одного члена.

В контуре встречается комбинация с B и \bar{B} , в соответствии с правилами булевой алгебры B и \bar{B} дополняют друг друга и их можно опустить, т.е. $(B + \bar{B})=1$. Т.о. $F=AB+A\bar{B} = A(B+\bar{B})=A$.

Ответ: $F = A$.

Методы минимизации логических функций

Пример 2. $F = AB\bar{C}D + \bar{A}B\bar{C}D + ABCD$

	$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$
$\bar{C}\bar{D}$	0	0	0	0
$\bar{C}D$	0	0	1	0
$C\bar{D}$	0	1	1	0
CD	0	0	0	0

Карта Карно из четырех переменных

На карте имеется два контура, следовательно новое минимизированное выражение будет состоять из двух членов, связанных функцией ИЛИ.

В горизонтальном контуре опускаем $(A + \bar{A})=1$;
в вертикальном контуре опускаем $(C + \bar{C})=1$.
Ответ: $F = BCD + ABD$.

В итоге, получили функцию, форма которой не подлежит дальнейшей минимизации и называется **тупиковой**.

Правила использования карт Карно

1. Нанести на карту Карно единицы в соответствии с заданной функцией (логическая функция должна быть представлена в СДНФ).
2. Объединяем соседние единицы контурами по 2, 4 или 8 клеток.
3. Проводим упрощения, исключая взаимодополняющие переменные внутри контура
4. Оставшиеся члены объединяем функцией ИЛИ. Полученное выражение записываем в ДНФ.

Методы минимизации логических функций

	$\bar{A}\bar{B}$	$\bar{A}B$	$A\bar{B}$	$A\bar{B}$
$\bar{C}\bar{D}$				1
$\bar{C}D$	1	1		1
$C\bar{D}$	1	1		
CD				

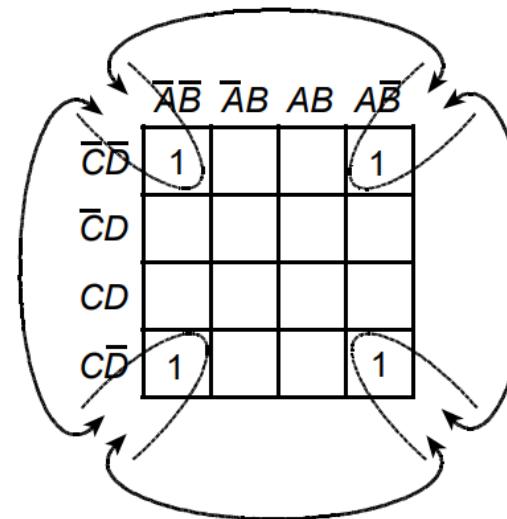
Пример 3.

$$F = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}CD + \bar{A}\bar{B}C\bar{D} + A\bar{B}\bar{C}\bar{D}$$

Объединяем в два контура по 2 и 4 единицы.

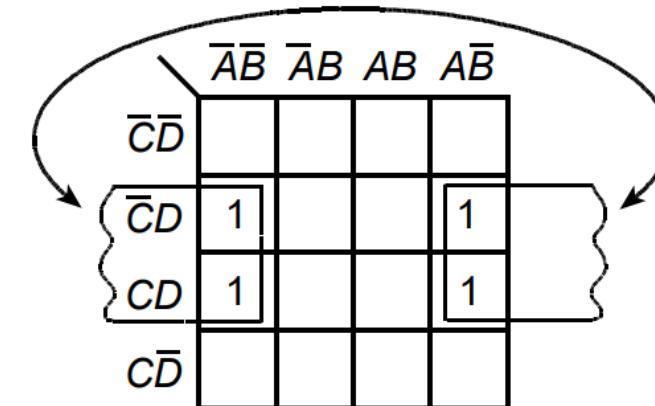
Результат минимизации: $F = \bar{A}D + A\bar{B}\bar{C}$

Существуют также нестандартные способы построения контуров.



$$F = BD.$$

Здесь опускаются A и \bar{A} и C и \bar{C} .



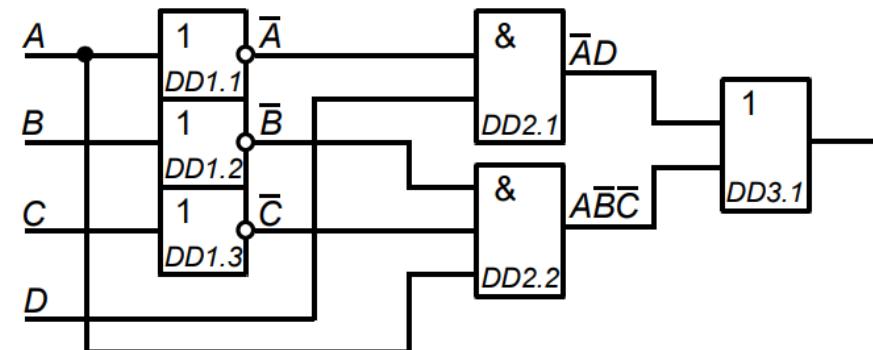
$$F = \bar{B}D.$$

Здесь опускаются C и \bar{C} и A и \bar{A} .

Синтез электронных схем по заданной функции

В результате минимизации: $F = \overline{AD} + A\overline{BC}$

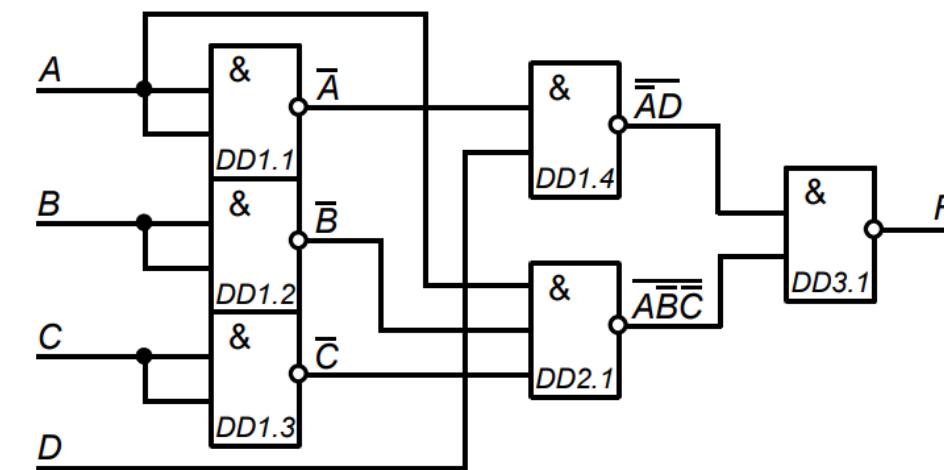
1. Реализация ЛФ в смешанном базисе.



2. Реализация ЛФ в базисе И-НЕ.

Преобразуем функцию к виду, в котором будет использоваться функция И-НЕ.

$$F = \overline{\overline{F}} = \overline{\overline{\overline{AD} + A\overline{BC}}} = \overline{\overline{\overline{AD}}} \cdot \overline{\overline{A\overline{BC}}} = \overline{AD} \cdot \overline{ABC}$$

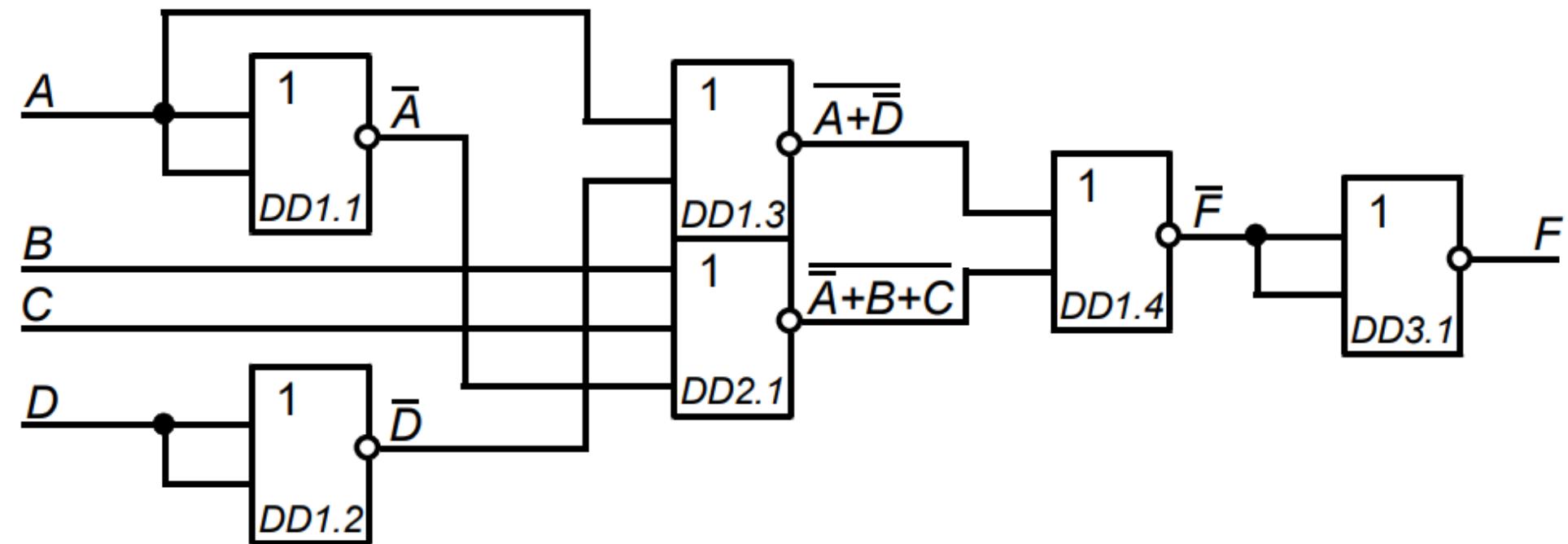


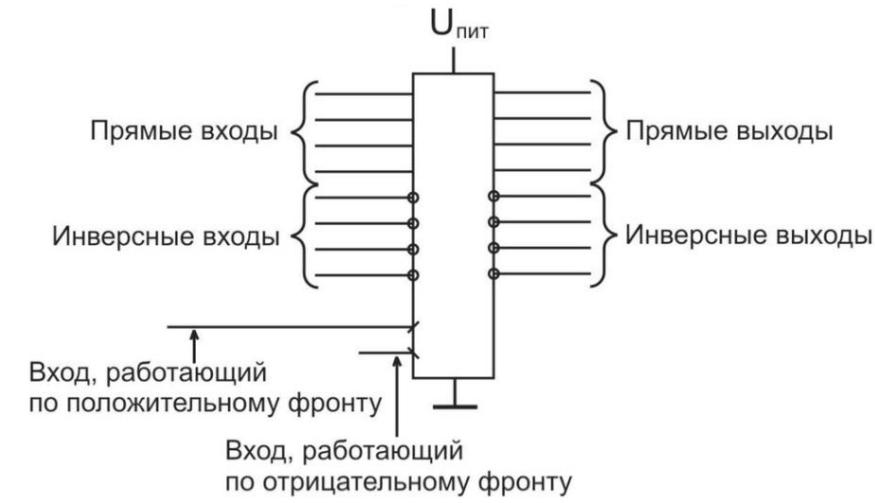
Синтез электронных схем по заданной функции

3. Реализация ЛФ в базисе ИЛИ-НЕ.

Преобразуем функцию к виду, где используется только логическая функция ИЛИ-НЕ.

$$F = \overline{AD} + A\overline{BC} = \overline{\overline{AD} \cdot \overline{ABC}} = \overline{(A + \overline{D})(\overline{A} + B + C)} = \overline{(A + \overline{D})} + \overline{(\overline{A} + B + C)}$$





Базовые комбинационные блоки

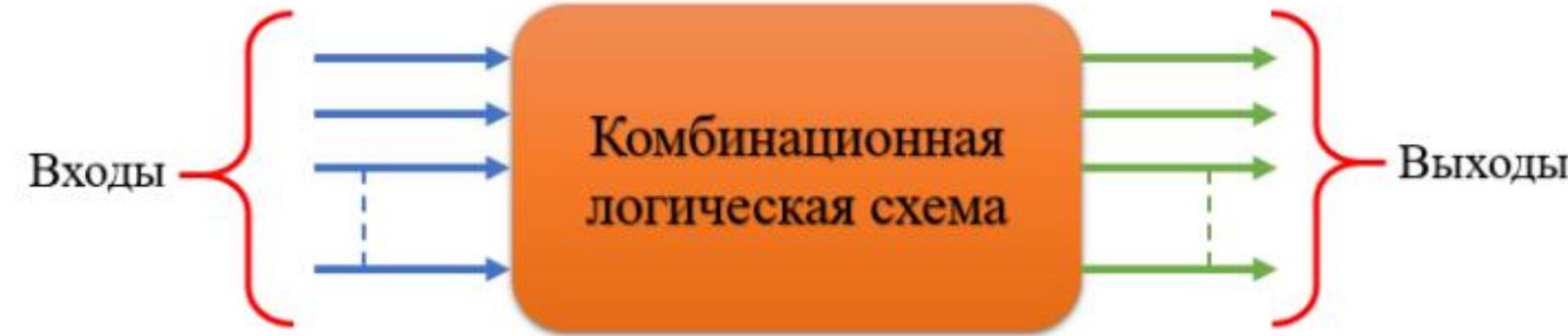


Базовые комбинационные блоки

- **Базовые комбинационные блоки** являются фундаментальными строительными элементами цифровых схем, в том числе и в электронно-вычислительных машинах (ЭВМ).
- Их ключевая особенность заключается в том, что выходное состояние схемы зависит исключительно от текущей комбинации входных сигналов, без учета предыдущих состояний (в отличие от последовательностных схем, имеющих память).
- **Основные базовые комбинационные блоки:**
 - **Логических функций, элементы:** И (AND), ИЛИ (OR), НЕ (NOT), И-НЕ (NAND), ИЛИ-НЕ (NOR), Исключающее ИЛИ (XOR).
 - **Сумматоры** (Adders) (Полусумматор, Полный сумматор)
 - **Шифраторы** (Encoders) и **Дешифраторы** (Decoders)
 - **Мультиплексоры** (Multiplexers, MUX) и **Демультиплексоры** (Demultiplexers, DEMUX)
 - **Компараторы** (Comparators)
 - **Преобразователи кодов** (Code Converters)

Комбинационные логические устройства

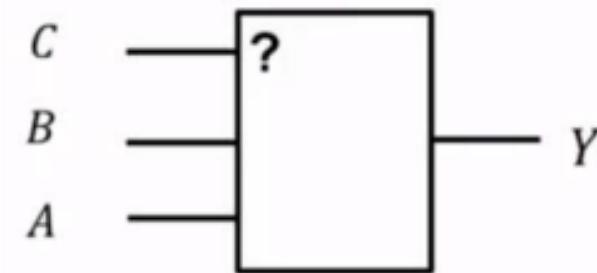
- Комбинационными называются логические устройства, выходные сигналы которых однозначно определяются комбинацией входных сигналов в тот же момент времени.



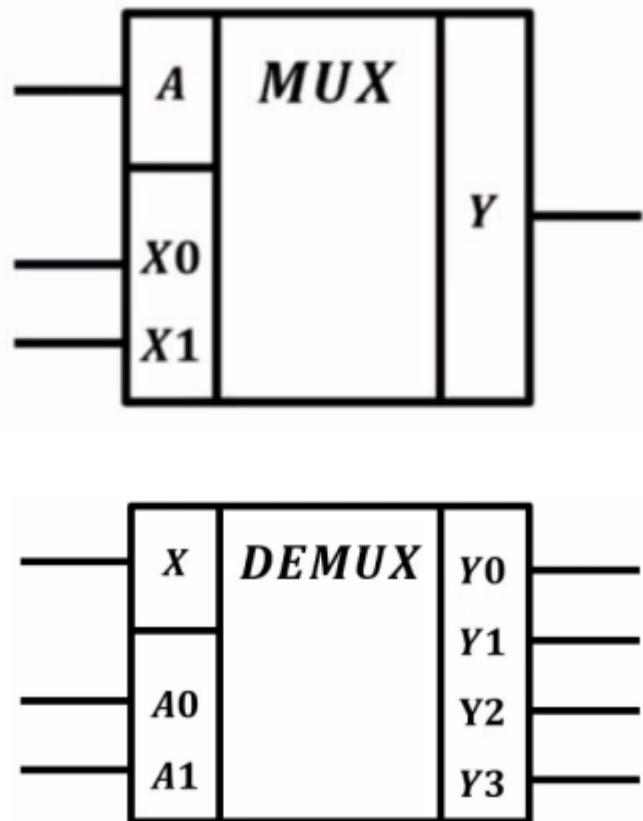
- **Логические элементы И, ИЛИ, НЕ, И-НЕ, ИЛИ-НЕ являются строительными блоками комбинационных логических схем.**
- Примером комбинационной схемы является дешифратор, который преобразует двоичные кодовые данные, присутствующие на его входе, в ряд различных выходных линий, по одной за раз производя эквивалентный десятичный код на своем выходе. Комбинационные логические схемы могут быть очень простыми или очень сложными, и любая комбинационная схема может быть реализована только с использованием вентилей И-НЕ и ИЛИ-НЕ, поскольку они классифицируются как «универсальные» вентили.
- **Три основных способа задания функции комбинационной логической схемы: Булева алгебра, таблица истинности и логическая схема.**
- **К распространенным комбинационным схемам, состоящим из отдельных логических вентилей, относятся** мультиплексоры, демультиплексоры, шифраторы, дешифраторы, сумматоры, компараторы и т. д.

Комбинационные цифровые устройства

- **Комбинационные цифровые устройства (КЦУ)** - устройства, в которых выходные сигналы определяются **только текущими** входными сигналами!
- **Комбинационные цифровые устройства - устройства без памяти**



A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



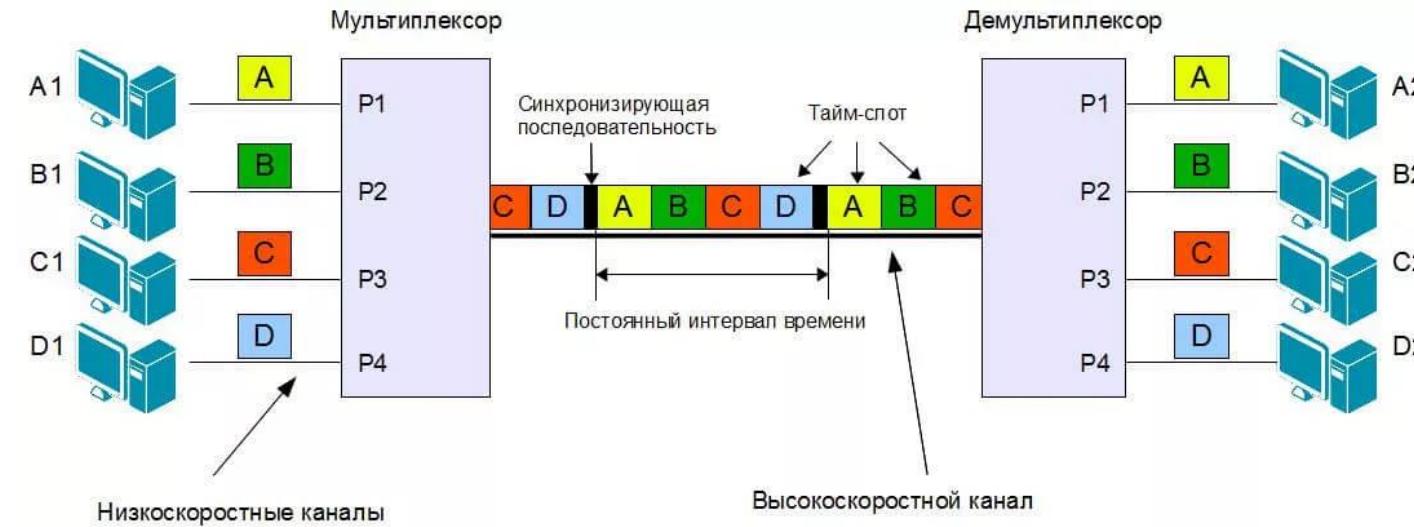
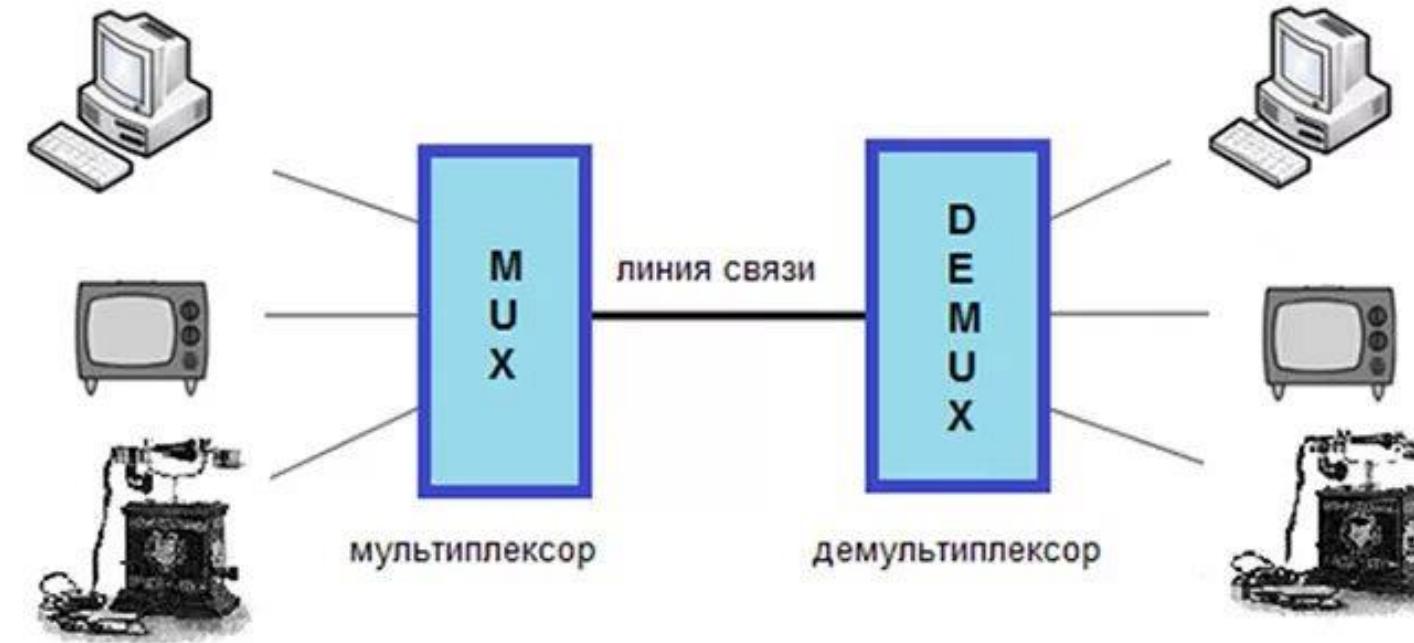
Базовые комбинационные блоки Мультиплексоры и Демультиплексоры



Мультиплексоры и Демультиплексоры

- **Мультиплексоры и демультиплексоры** — это важные устройства в области электроники и телекоммуникаций.
- Они применяются для эффективной передачи данных через ограниченные каналы связи, таких как электрические цепи, оптоволоконные линии и другие системы.
- **Мультиплексоры** позволяют передавать несколько сигналов через один канал, тем самым экономя ресурсы и повышая эффективность передачи информации.
- **Демультиплексоры** выполняют обратную задачу — разделяют один канал на несколько потоков данных.

Мультиплексор и Демультиплексор



Мультиплексор и Демультиплексор

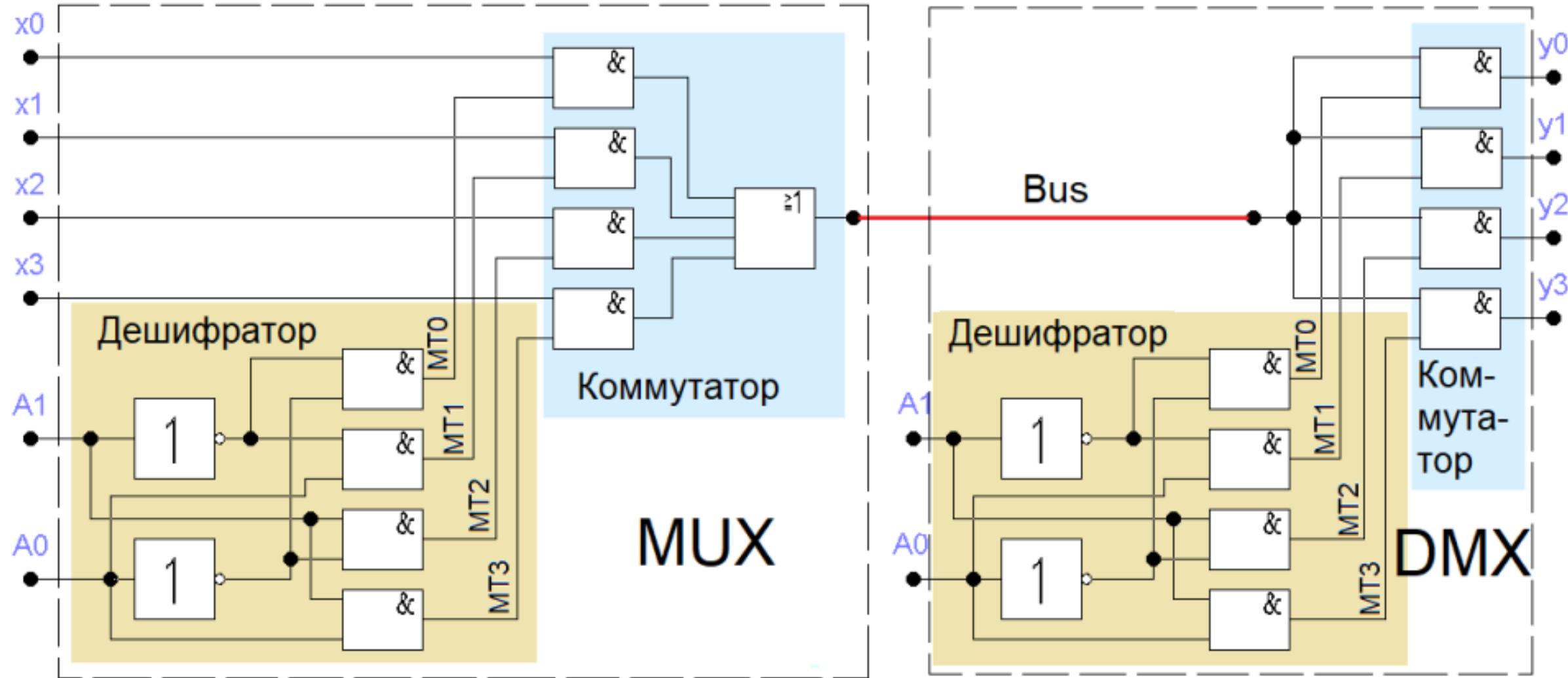
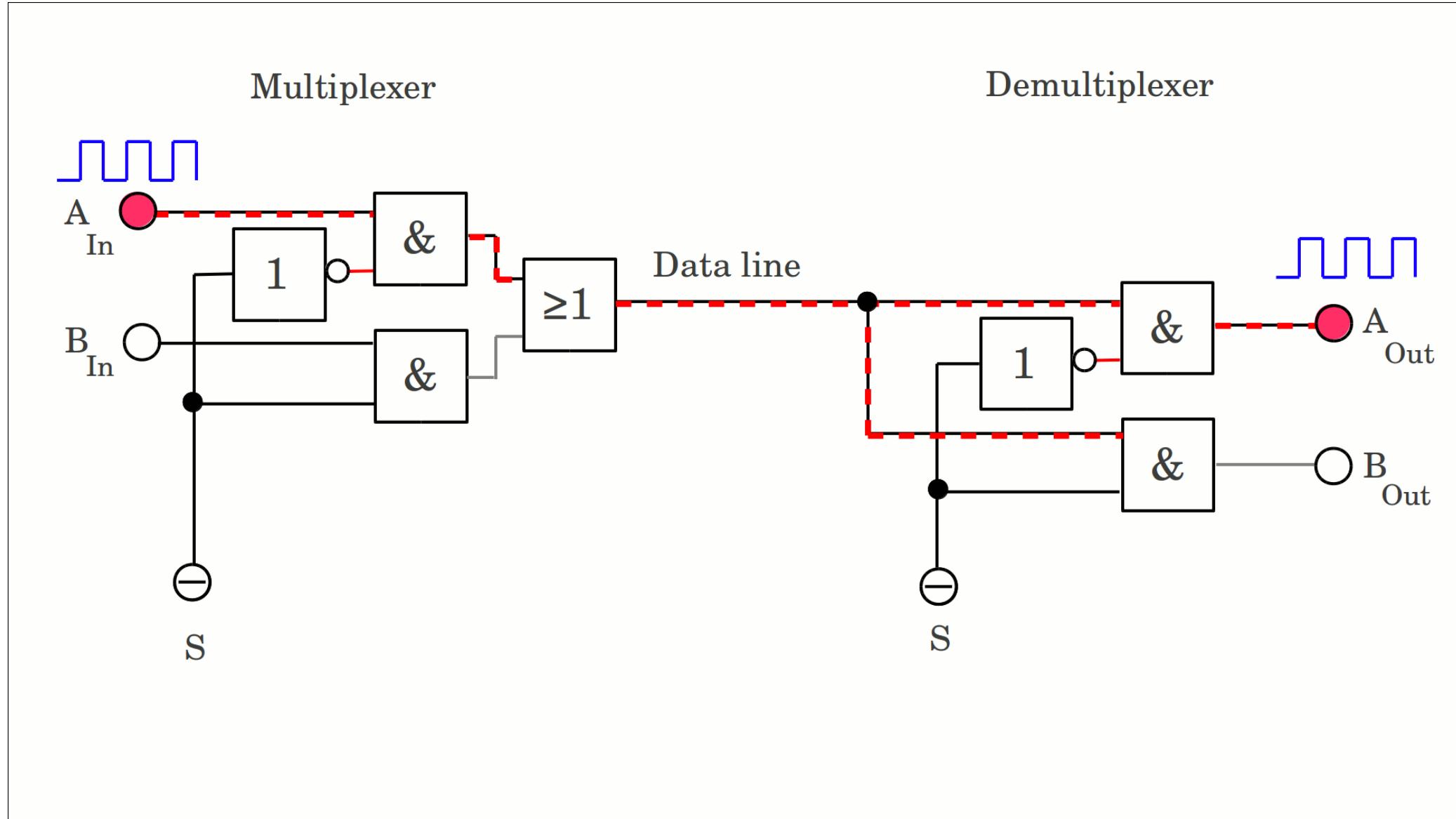


Схема 4-канальной системы передачи-приема

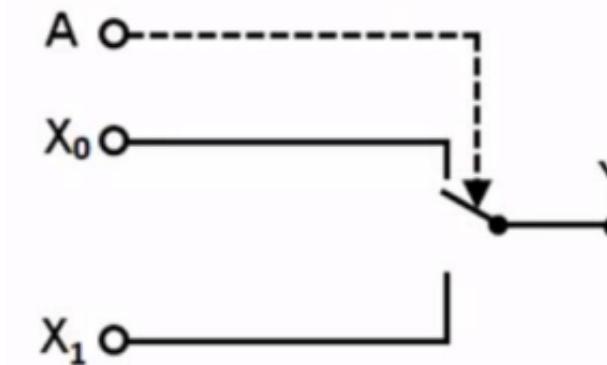
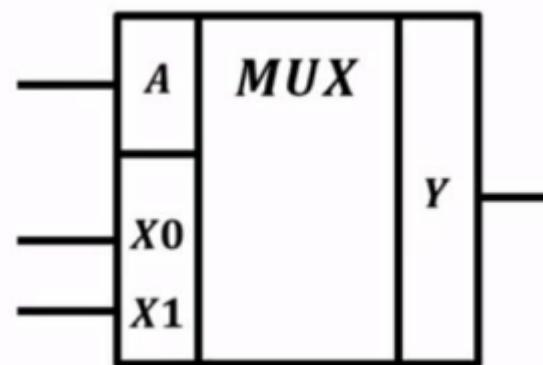
Мультиплексор и Демультиплексор



Мультиплексор

- **Мультиплексор** — устройство, имеющее несколько сигнальных входов, один или более управляющих входов и один выход.
- Мультиплексор позволяет передавать сигнал с одного из входов на выход; при этом выбор желаемого входа осуществляется подачей соответствующей комбинации управляющих сигналов.

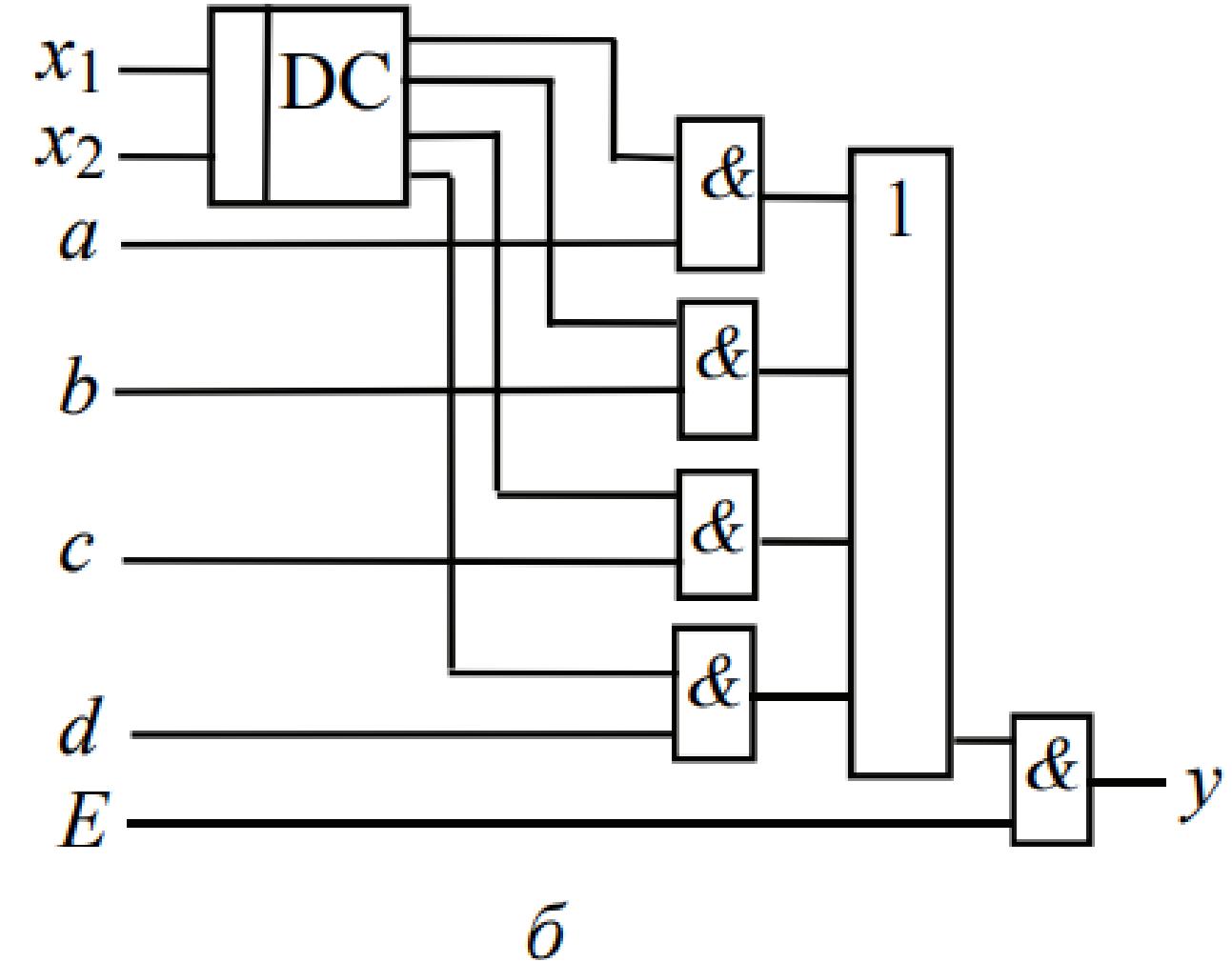
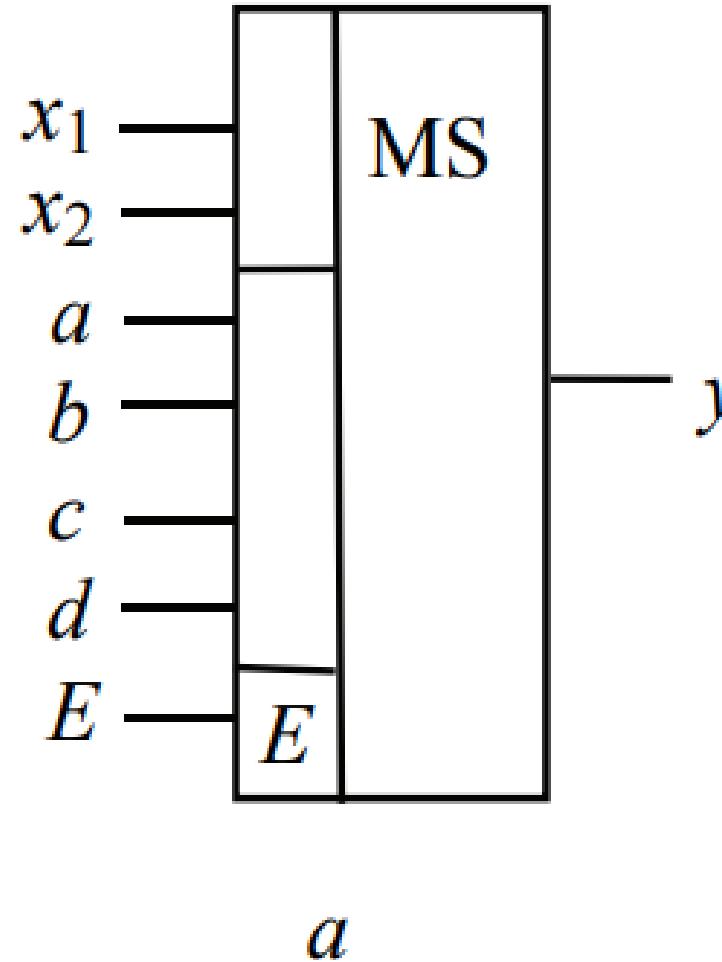
A	X1	X0	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



A	\bar{A}		
X_1	1	1	1
\bar{X}_1	0	0	1

$$Y = A \cdot X_1 + \bar{A} \cdot X_0$$

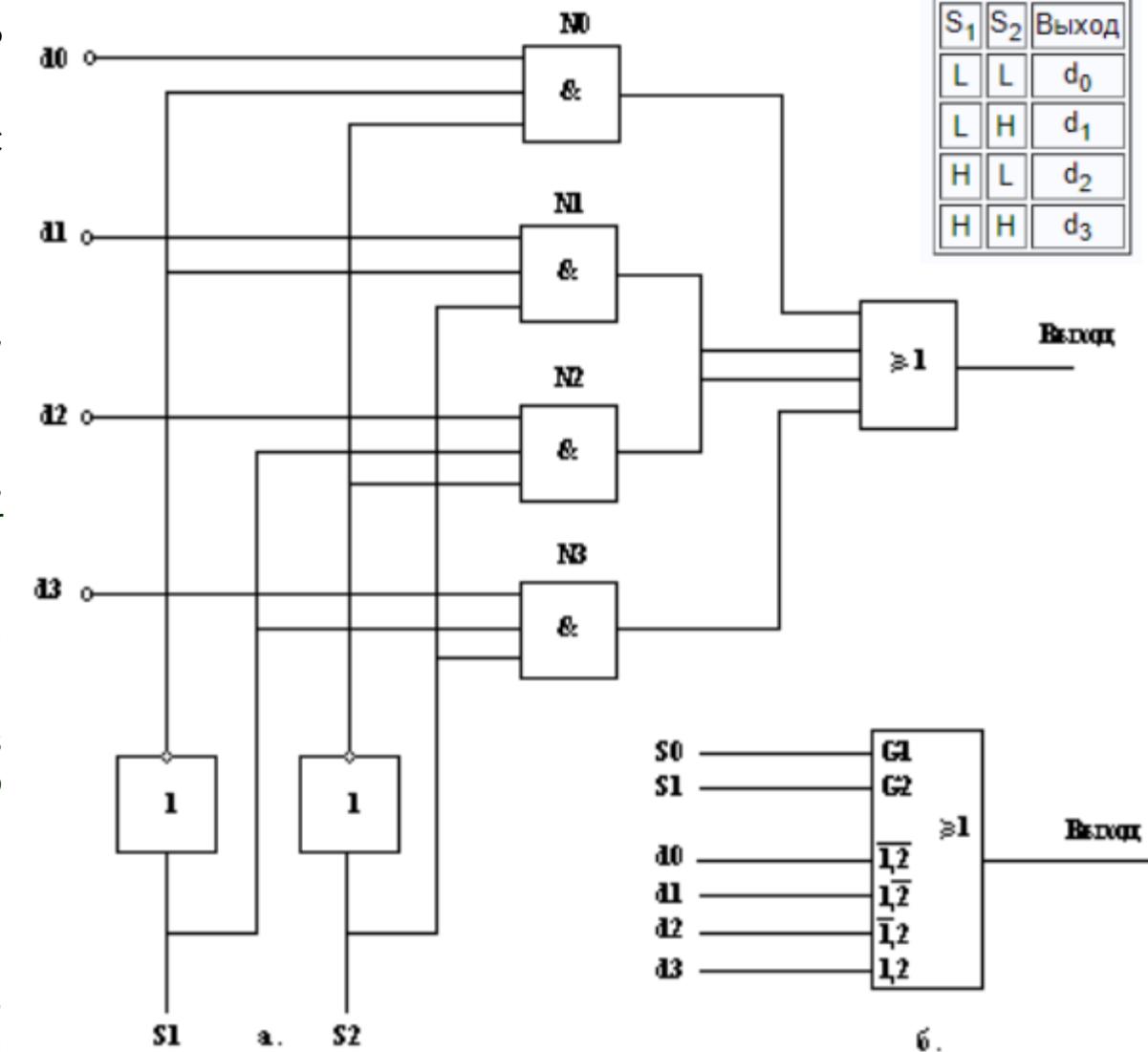
Мультиплексор



а – условное изображение; б – вариант структурной схемы

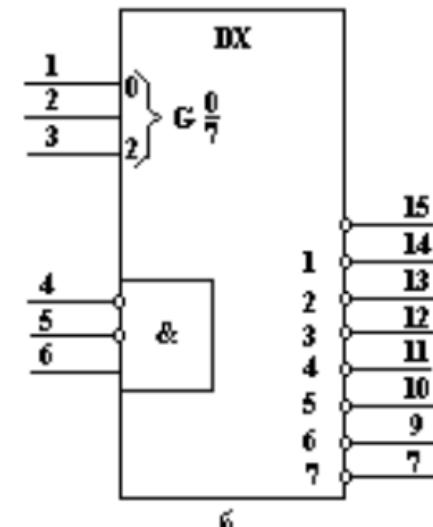
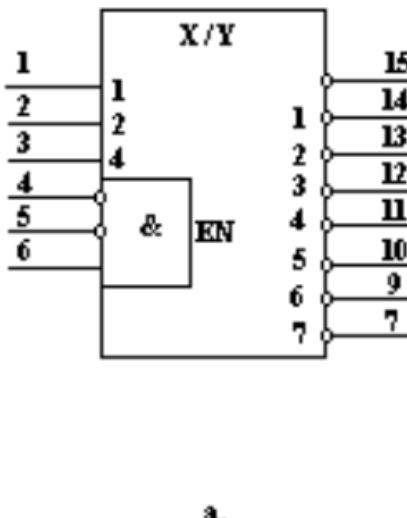
Мультиплексор (Селектор)

- **Мультиплексоры, они же Селекторы**, есть схемы выбора входов.
- Используются для связи приемного устройства с различными источниками данных.
- Определенный вход (источник данных) выбирается путем подачи на адресные линии двоичного числа (адреса), которое указывает, какой именно канал должен быть выбран.
- В большинстве случаев такая схема снабжается дополнительным стробирующим входом, который разрешает передачу данных в момент времени, когда происходит выборка.
- Механическим аналогом мультиплексора является многопозиционный переключатель.
- Представленный мультиплексор состоит из четырех схем И, выходы которых связаны со схемой ИЛИ.
- На один из входов схем И поступают данные. Два других входа используются для выборки.
- Определенный канал будет выбран в зависимости от того, какой уровень сигнала установлен на селектирующих линиях S_1 и S_2 .

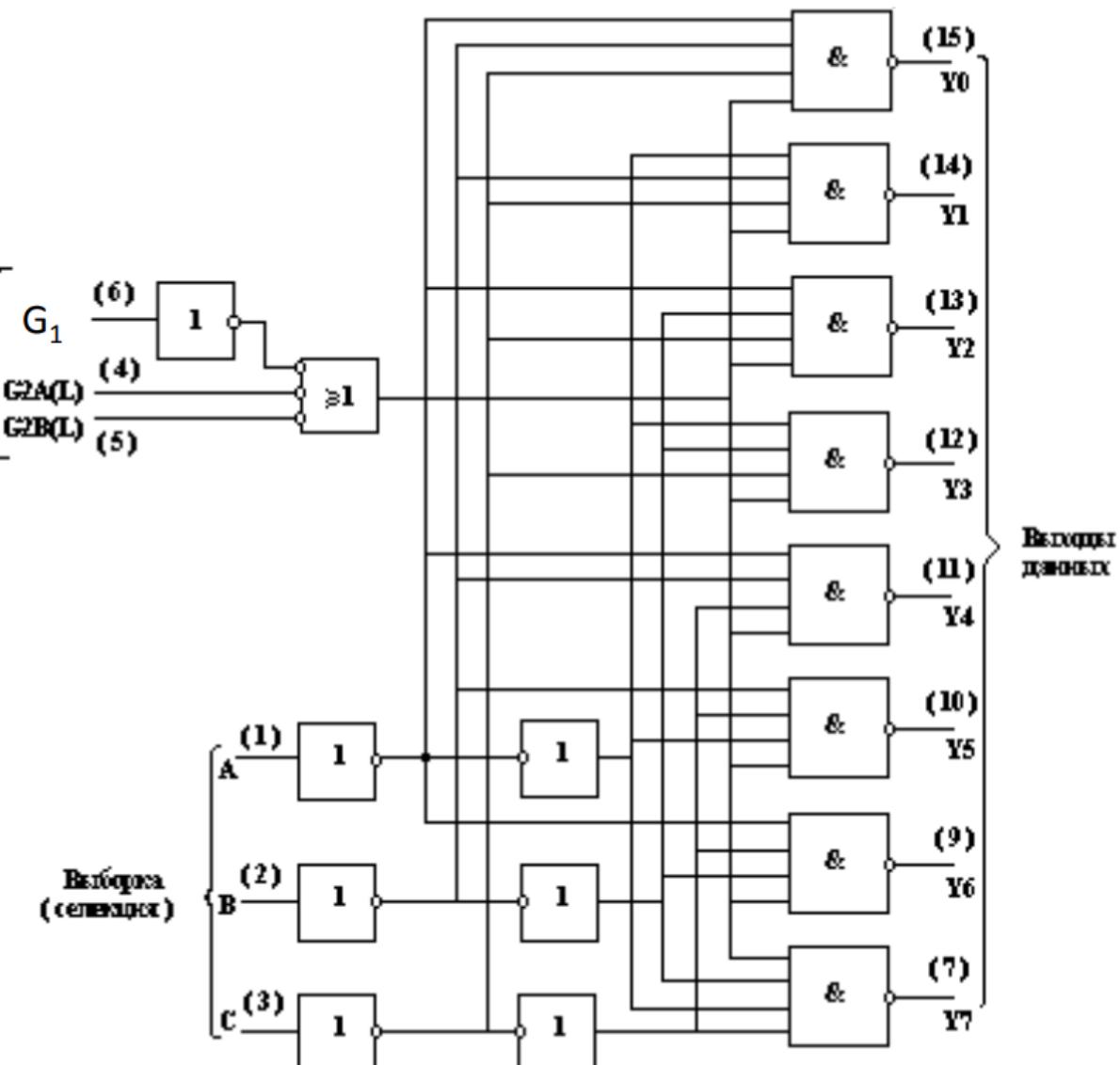


Демультиплексор

- **Демультиплексор** – логическое устройство, предназначенное для переключения сигнала с одного информационного входа на один из нескольких информационных выходов.
- Номер выхода, на который передается значение входного сигнала, определяется адресным кодом.
- Поэтому демультиплексор основан на схеме дешифратора и реализуется в одной микросхеме.

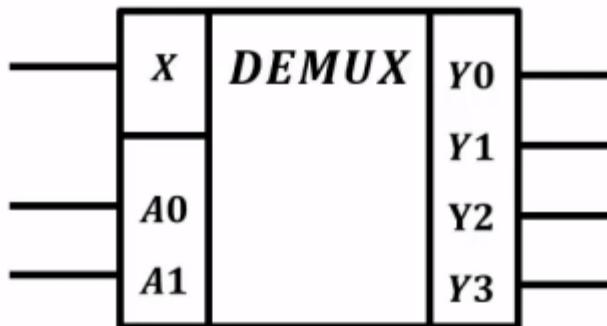


Представленная схема имеет в своем составе комбинированный вход "данные/строб". Вход G1 (данные) запускается H-сигналом, входы G2A и G2B – L-сигналами. Вариант (а) – дешифратор, (б) – демультиплексор.



Демультиплексор

- **Демультиплексор** — это логическое устройство, предназначенное для переключения сигнала с одного информационного входа на один из нескольких информационных выходов.



A1	A0	Y3	Y2	Y1	Y0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

A1	A0	X	Y3	Y2	Y1	Y0
0	0	0	0	0	0	0
0	0	1	0	0	0	1
0	1	0	0	0	0	0
0	1	1	0	0	1	0
1	0	0	0	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	0	0
1	1	1	1	0	0	0

$$Y3 = A1 \cdot A0 \cdot X$$

$$Y2 = A1 \cdot \overline{A0} \cdot X$$

$$Y1 = \overline{A1} \cdot A0 \cdot X$$

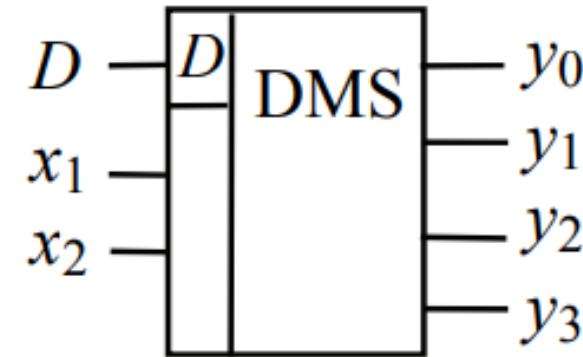
$$Y0 = \overline{A1} \cdot \overline{A0} \cdot X$$

Демультиплексор

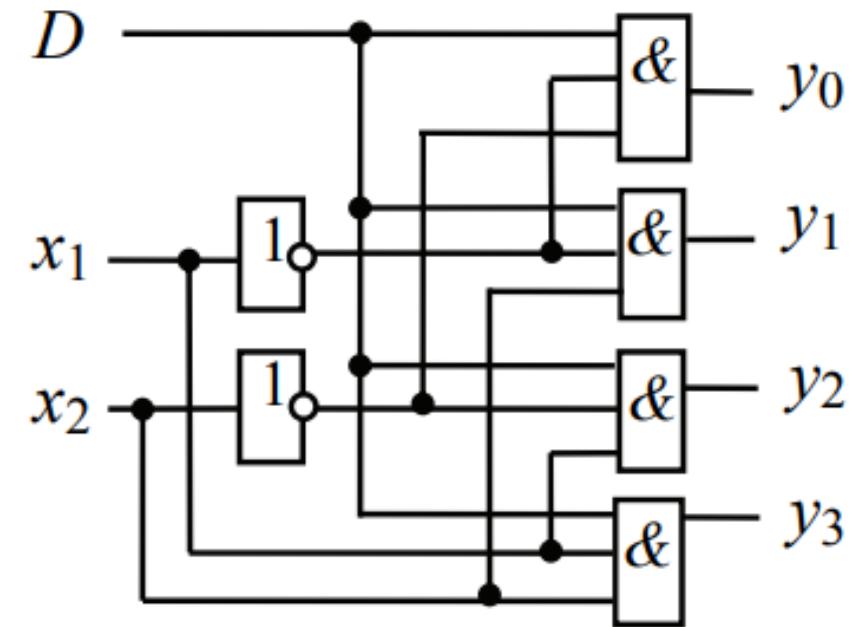
Закон функциональности

D	x_1	x_2	y_3	y_2	y_1	y_0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

a



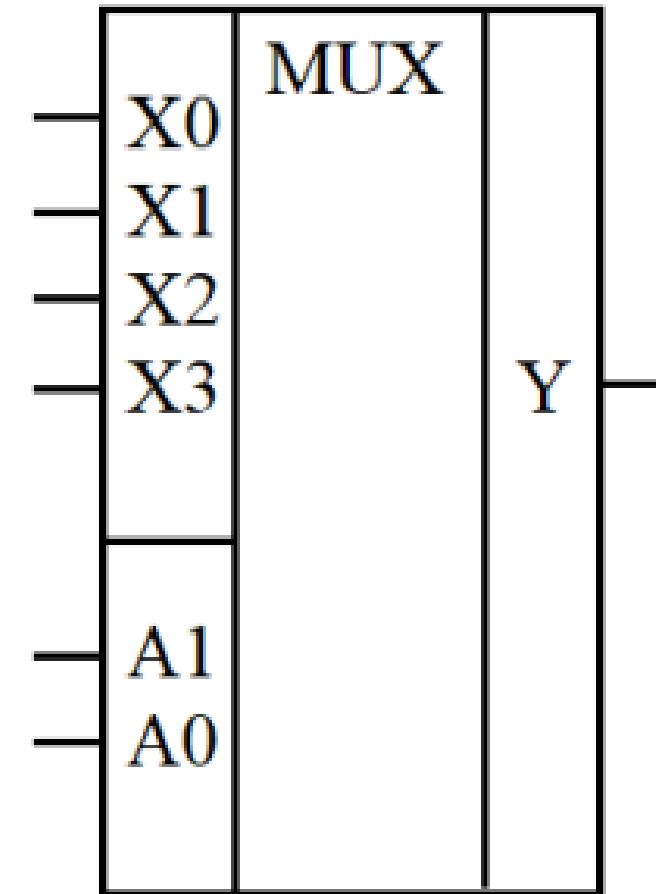
б



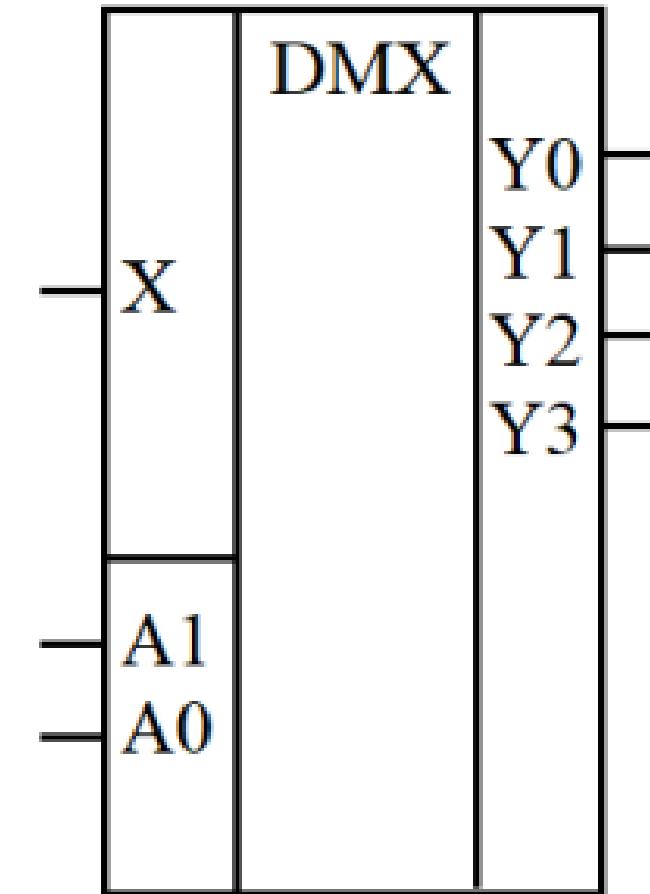
в

а – закон функциональности; б – условное графическое изображение;
в – функциональная схема демультиплексора

Мультиплексор и Демультиплексор

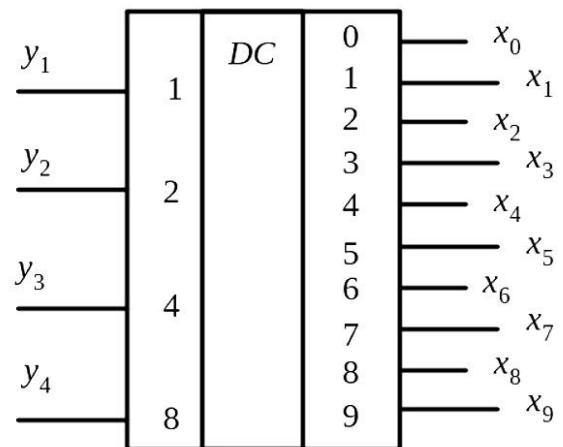
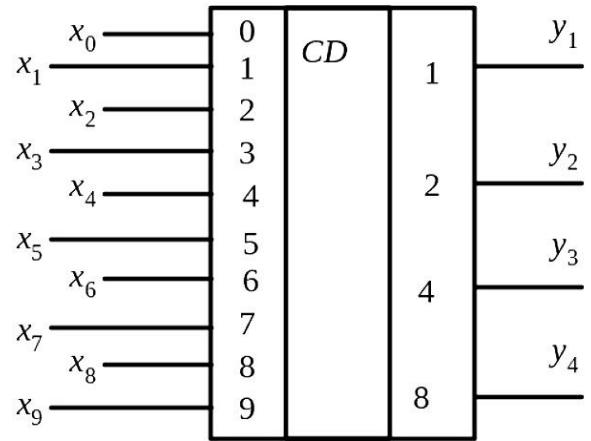


a)



б)

мультиплексор (а) и демультиплексор (б)



Базовые комбинационные блоки Шифраторы и Дешифраторы

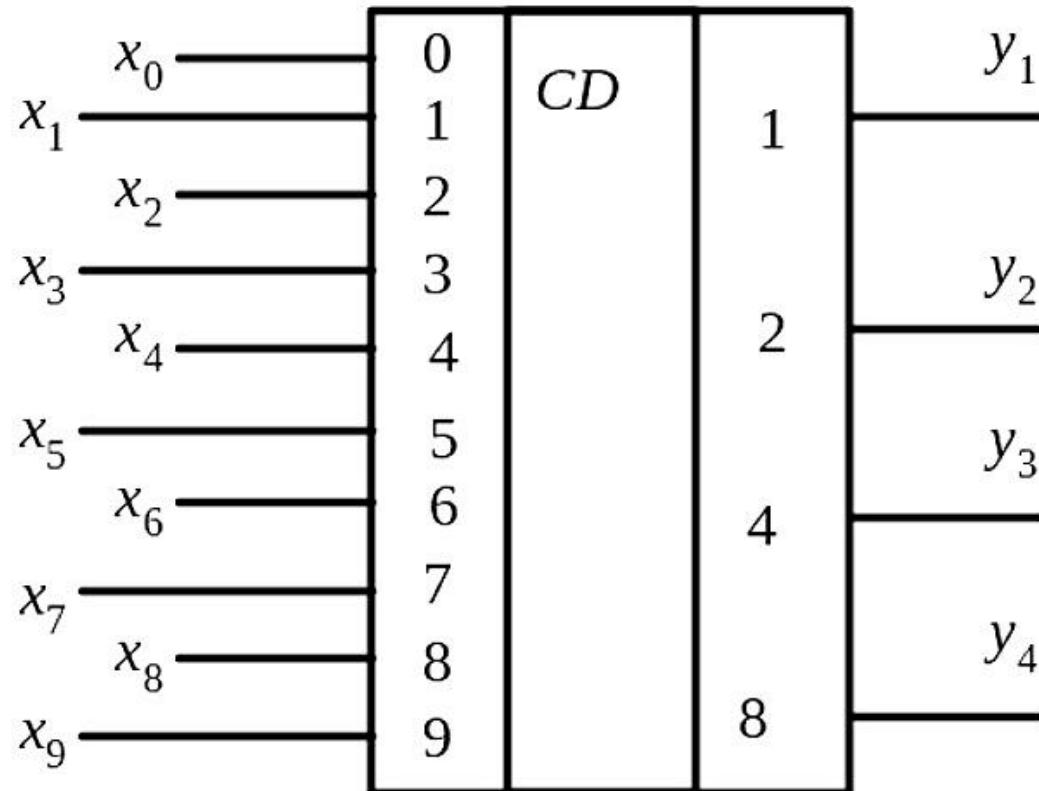


Шифратор и дешифратор

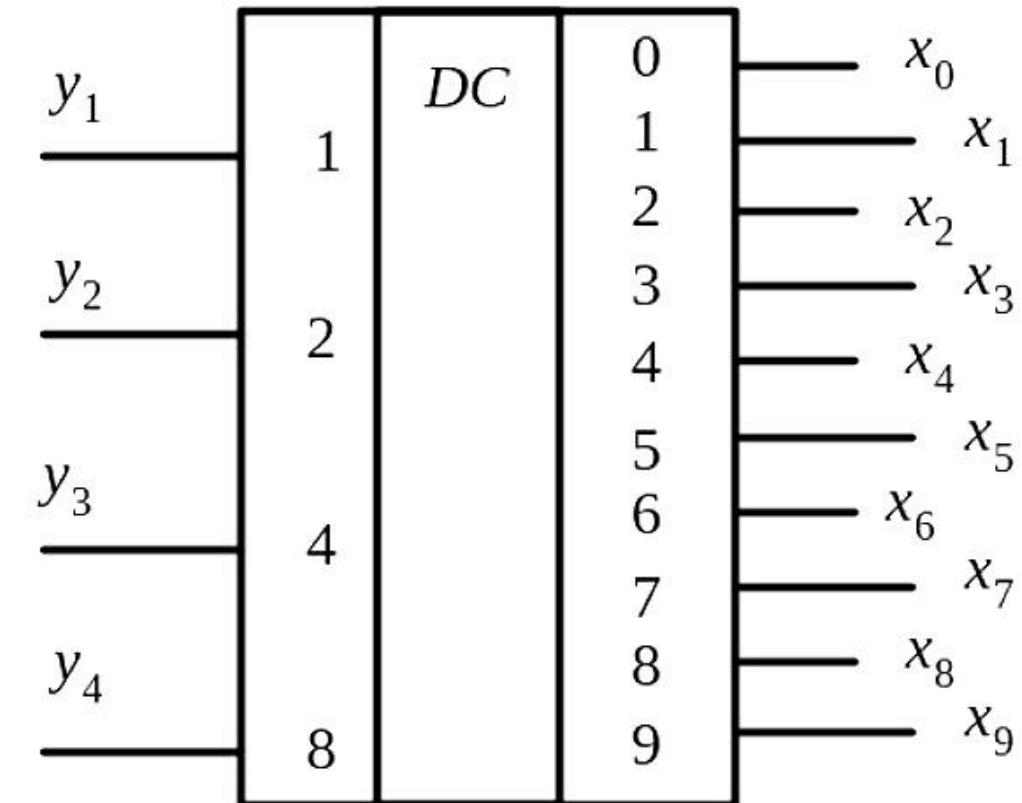
- Шифратор и дешифратор являются типовыми узлами ЭВМ.
- **Шифратор (кодер)** - это логическое устройство, которое преобразует единичный сигнал на одном из входов в n-разрядный двоичный код.
- Наибольшее применение он находит в устройствах ввода информации (например в клавиатуре), для преобразования десятичных чисел в двоичную систему счисления.
- **Дешифратор (декодер)** - это логическое устройство, преобразующее двоичный код, поступающий на его входы, в сигнал только на одном из его выходов.
- Дешифраторы широко применяются в устройствах управления, в системах цифровой индикации с газоразрядными индикаторами, для построения распределителей импульсов по различным цепям и т.д. Схема используется для перевода двоичных цифр в десятичные.

Шифратор и дешифратор

Шифратор



Дешифратор

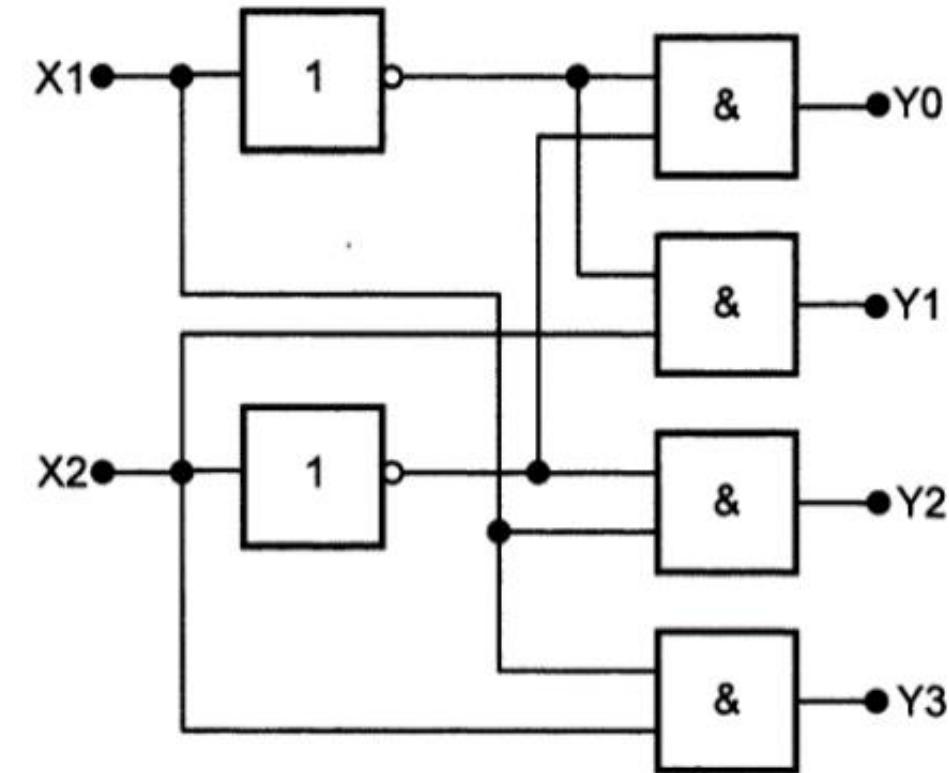


Дешифратор преобразует код в активную линию, а **шифратор** выполняет обратное действие — преобразует активную линию в код. Дешифратор имеет меньше входов и больше выходов.

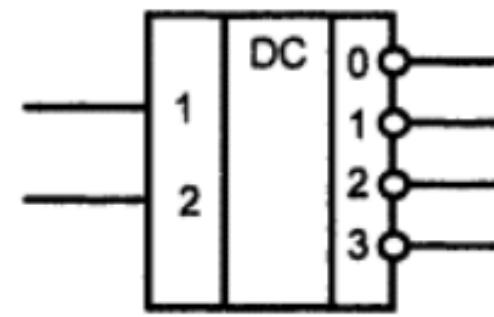
Шифратор наоборот имеет больше входов и меньше выходов. Эти устройства часто применяются в паре для организации различных логических схем выбора и управления.

Дешифратор

- **Функция дешифраторов** состоит в том, чтобы преобразовывать входной двоичный код в номера линий выходного сигнала.
- Дешифратор называют полным, если он имеет количество выходов m , связанных с количеством разрядов n входного двоичного числа соотношением $m = 2^n$.
- В стандартных сериях микросхем существуют дешифраторы на 4, 8 или 16 выходов, соответственно они имеют 2, 3 или 4 входа.



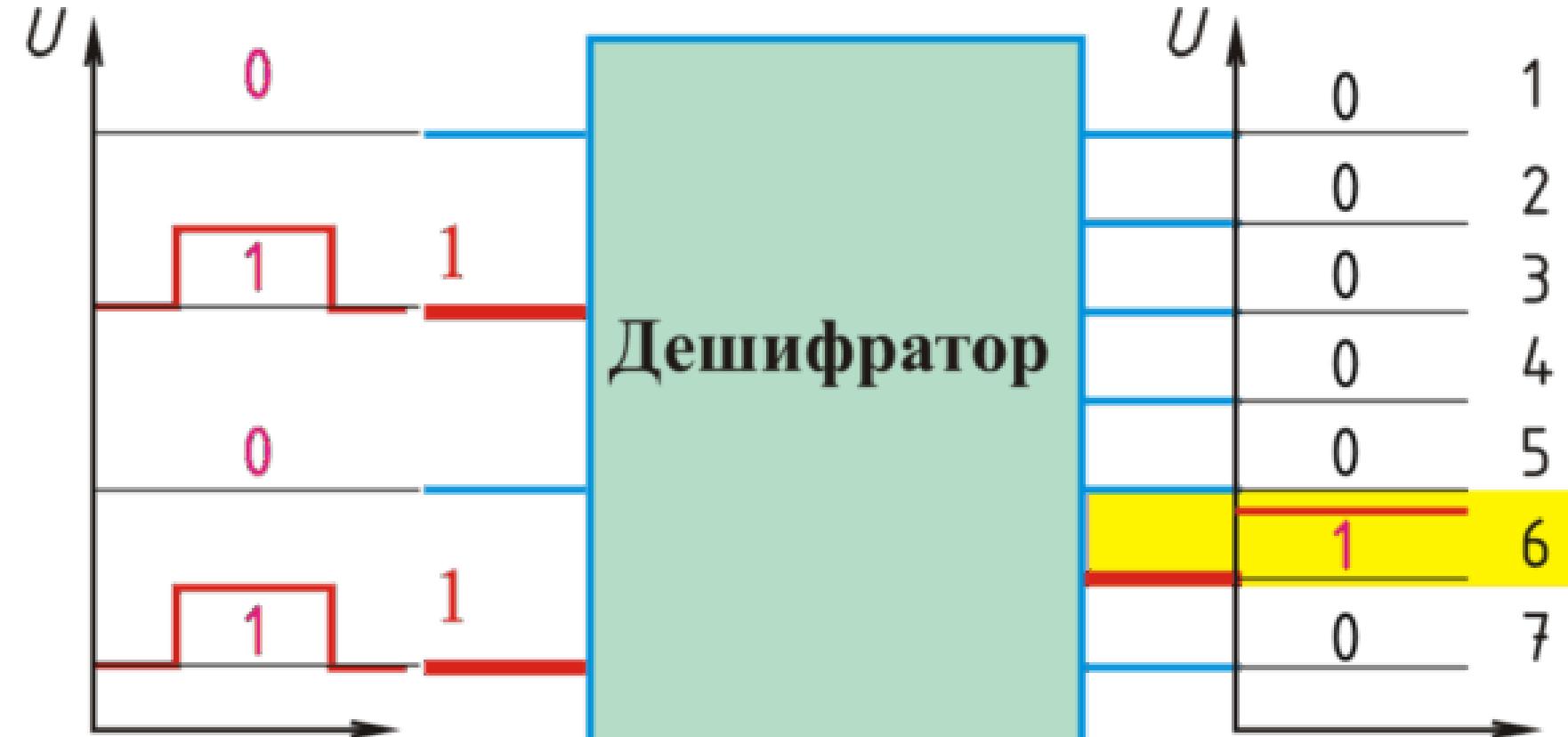
Простейший двухразрядный дешифратор



Номер входного сигнала	Вход		Выход			
	X1	X0	Y3	Y2	Y1	Y0
0	0	0	0	0	0	1
1	0	1	0	0	1	0
2	1	0	0	1	0	0
3	1	1	1	0	0	0

Дешифратор

x1	x2	x3	x4	у
0	0	0	0	1
0	0	0	1	2
0	0	1	0	3
0	0	1	1	4
0	1	0	0	5
0	1	0	1	6
0	1	1	0	7



Классификация дешифраторов

- **Дешифраторы различаются по ряду признаков.**
- Наиболее важными характеристиками являются количество входов и выходов, наличие или отсутствие входа разрешения работы, тип логики выходов.
- **Прямой дешифратор** представляет собой стандартный вид дешифратора, который при наличии активного входного кода активирует один из выходов. Остальные выходы при этом находятся в неактивном состоянии. Такой тип дешифратора используется для простых схем выбора.
- **Дешифратор с разрешением** (англ. enable input) имеет дополнительный вход, который служит для включения или отключения работы устройства. При неактивном сигнале разрешения все выходы находятся в неактивном состоянии, независимо от состояния входных линий. Такой вариант применяется в случаях, когда необходимо управлять работой дешифратора извне, например, в многоуровневых схемах выбора.
- **Частично декодированный** отличается тем, что не все возможные комбинации входов преобразуются в отдельные выходы. Они используются там, где полный набор выходов не требуется или где необходимо уменьшить число логических элементов в схеме.

Дешифратор

Дешифратор семисегментного индикатора

Нумерация сегментов:



0 1 2 3 4 5 6 7 8 9

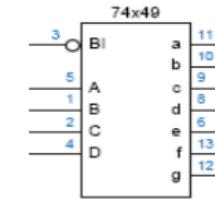
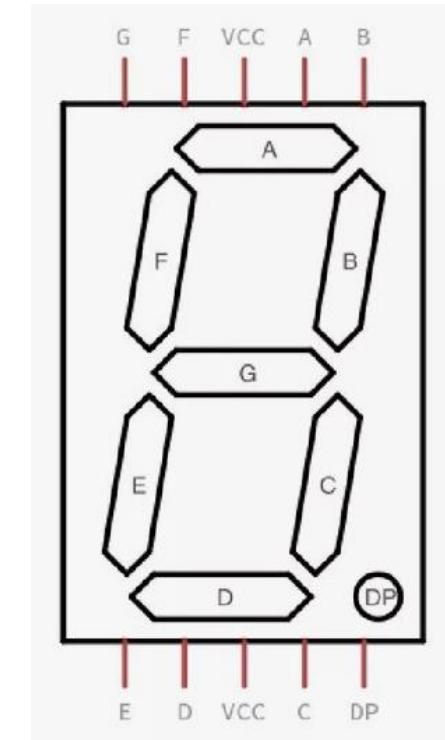
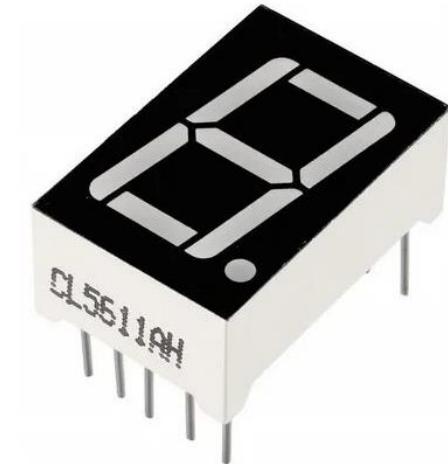
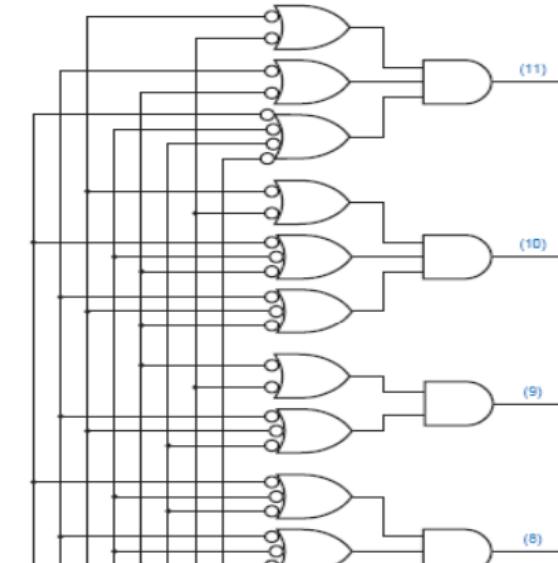
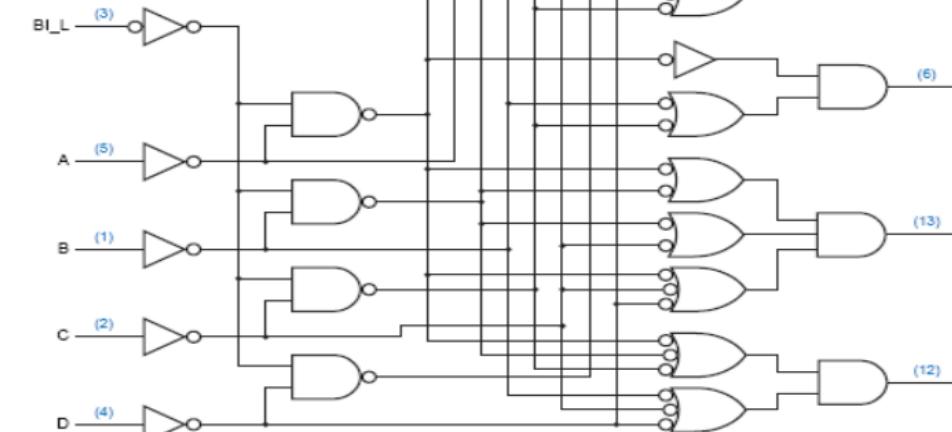
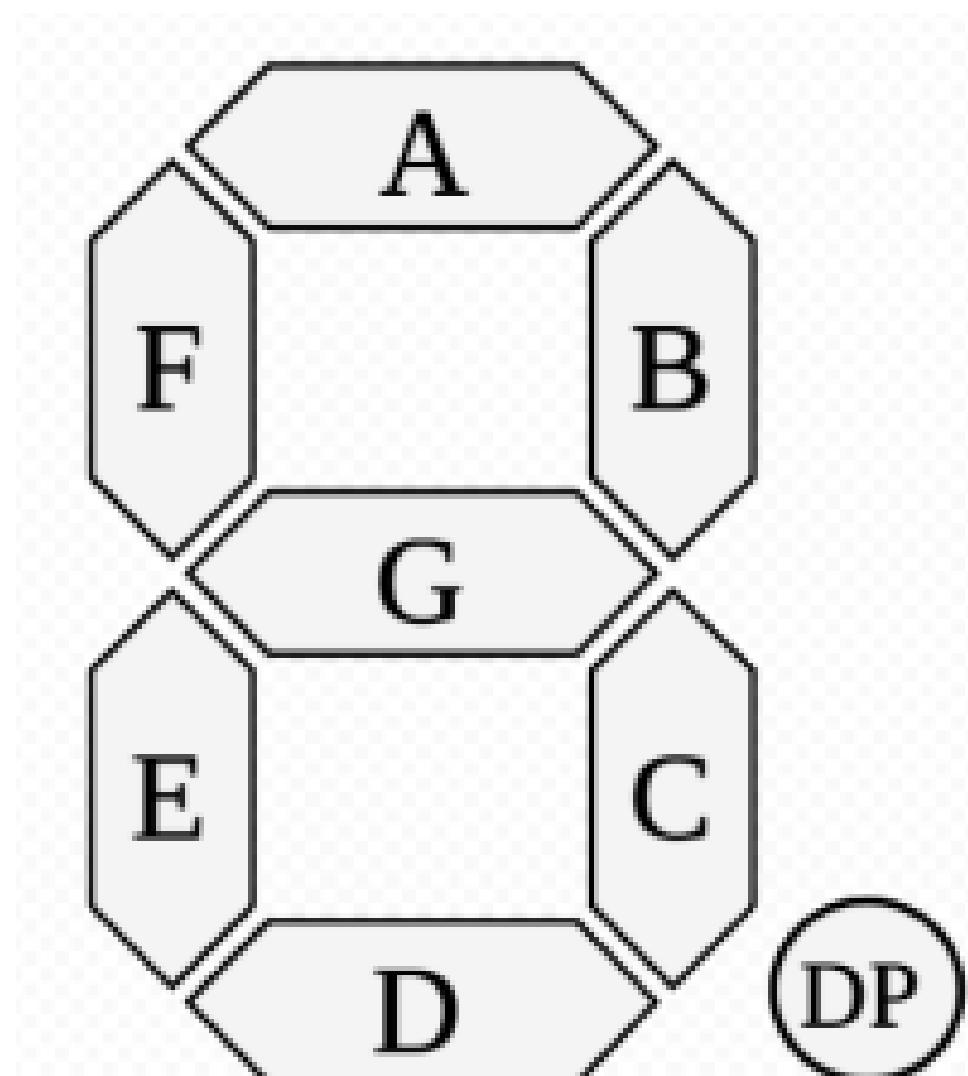


Схема дешифратора:

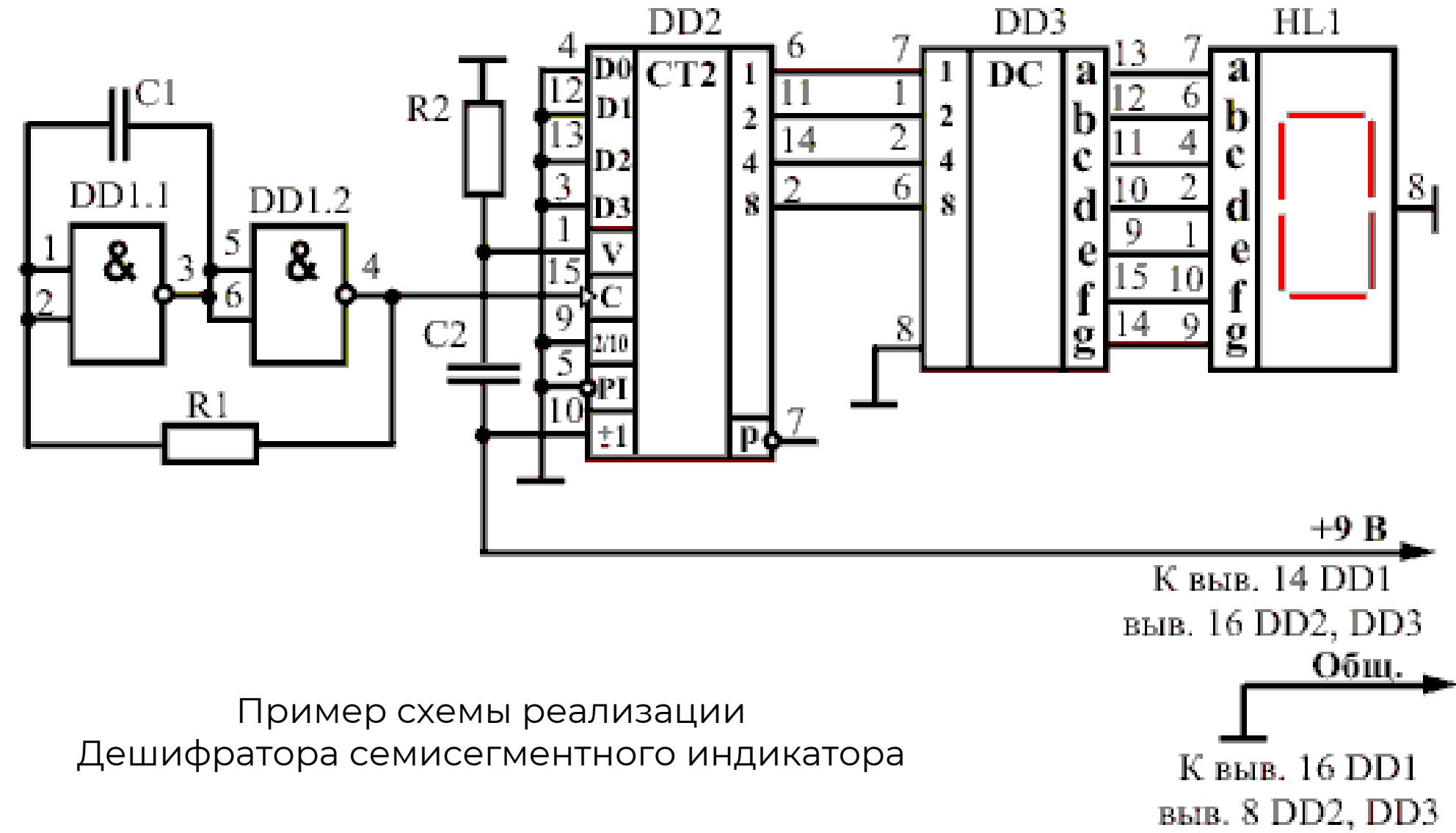


Семисегментный индикатор

Шестнадцатеричные числа	Двоичные числа	Состояние сегментов								УГО индикации
		A	B	C	D	E	F	G	DP	
0	0000	1	1	1	1	1	1	1	0	0
1	0001	0	1	1	0	0	0	0	0	0
2	0010	1	1	0	1	1	0	1	1	0
3	0011	1	1	1	1	0	0	1	1	0
4	0100	0	1	1	1	0	1	1	1	0
5	0101	1	0	1	1	0	1	1	1	0
6	0110	1	0	1	1	1	1	1	1	0
7	0111	1	1	1	0	0	0	0	0	0
8	1000	1	1	1	1	1	1	1	1	1
9	1001	1	1	1	1	0	1	1	1	1
A	1010	1	1	1	0	1	1	1	1	1
B	1011	0	0	1	1	1	1	1	1	1
C	1100	1	0	0	1	1	1	1	0	1
D	1101	0	1	1	1	1	1	1	0	0
E	1110	1	0	0	1	1	1	1	1	0
F	1111	1	0	0	0	1	1	1	1	1



Дешифратор

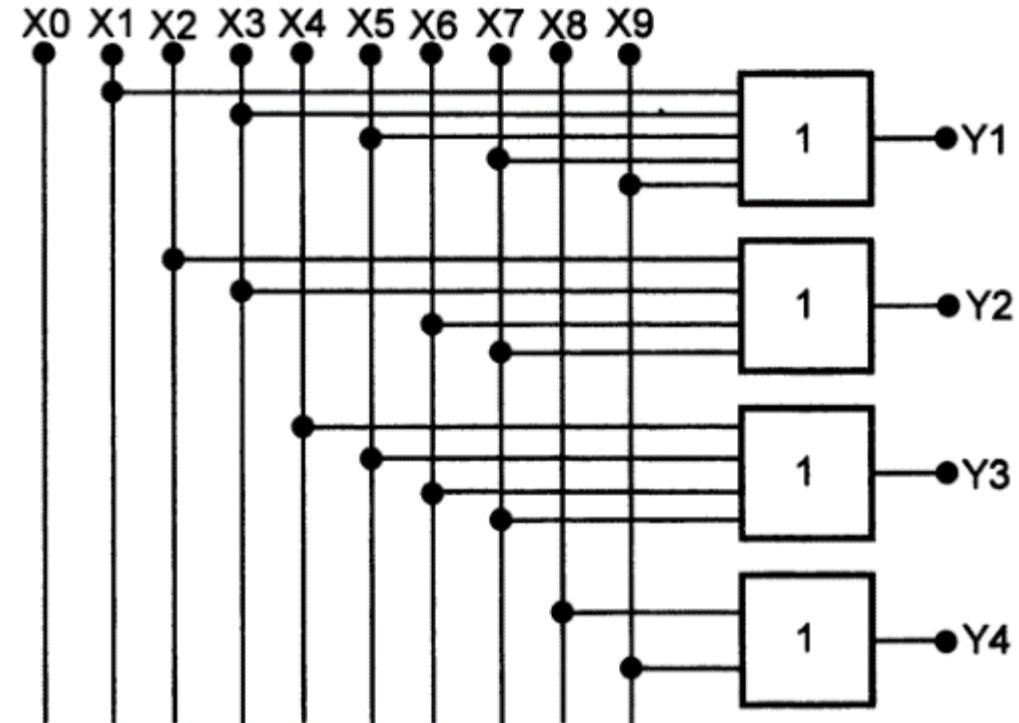


Пример схемы реализации
Дешифратора семисегментного индикатора

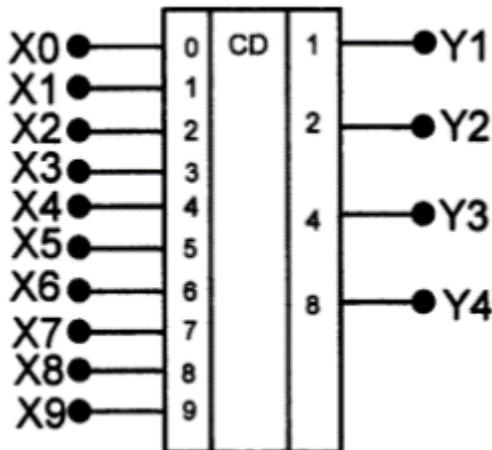
Шифратор

- Преобразователи кода решают задачу преобразования из одной системы кодирования информации в другую.
- Шифратор** (encoder, coder) – цифровое устройство, имеющее n входов и m выходов, преобразующее сигнал с n входных линий в m -разрядный (m -ичный) код.
- Шифратор называют полным, если выполняется условие $n = 2m$. Микросхемы шифраторов применяются значительно реже, чем дешифраторы.

Десятичное число	Двоичный код 8 – 4 – 2 – 1			
	Y4	Y3	Y2	Y1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1



Шифратор для преобразования десятичного кода (для чисел от 0 до 9) в двоичный код.



Классификация шифраторов

- Шифраторы подразделяются на несколько типов в зависимости от их функциональных особенностей и области применения.
- **Прямой шифратор** формирует на выходе двоичный код, соответствующий активному входу. В случае одновременной активации нескольких входов такой шифратор обычно выдаёт неопределённый результат, если не предусмотрена система приоритета.
- **Приоритетный шифратор** отличается от прямого тем, что при одновременной активации нескольких входов он формирует код входа с наибольшим приоритетом. Такой шифратор используется в более сложных схемах, где возможна ситуация наложения сигналов.

Приоритетный шифратор

A	B	C	D	Y1	Y0
0	0	0	0	0	0
0	0	0	1	0	0
0	0	1	0	0	1
0	0	1	1	0	1
0	1	0	0	1	0
0	1	0	1	1	0
0	1	1	0	1	0
0	1	1	1	1	0
1	0	0	0	1	1
1	0	0	1	1	1
1	0	1	0	1	1
1	0	1	1	1	1
1	1	0	0	1	1
1	1	0	1	1	1
1	1	1	0	1	1
1	1	1	1	1	1

	A	\bar{A}		
B	1	1	1	1
\bar{B}	1	1	1	1
\bar{C}	1	1	0	0
C	1	1	0	0
\bar{C}	1	1	0	0
	\bar{D}	D	\bar{D}	D

$$Y1 = A + B$$

	A	\bar{A}		
B	1	1	0	0
\bar{B}	1	1	0	0
\bar{C}	1	1	1	0
C	1	1	1	0
\bar{C}	1	1	1	0
	\bar{D}	D	\bar{D}	D

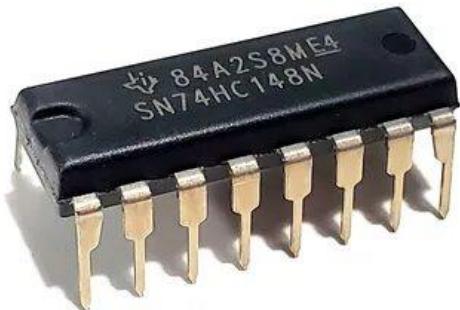
$$Y0 = A + \bar{B}C$$

Примеры микросхем

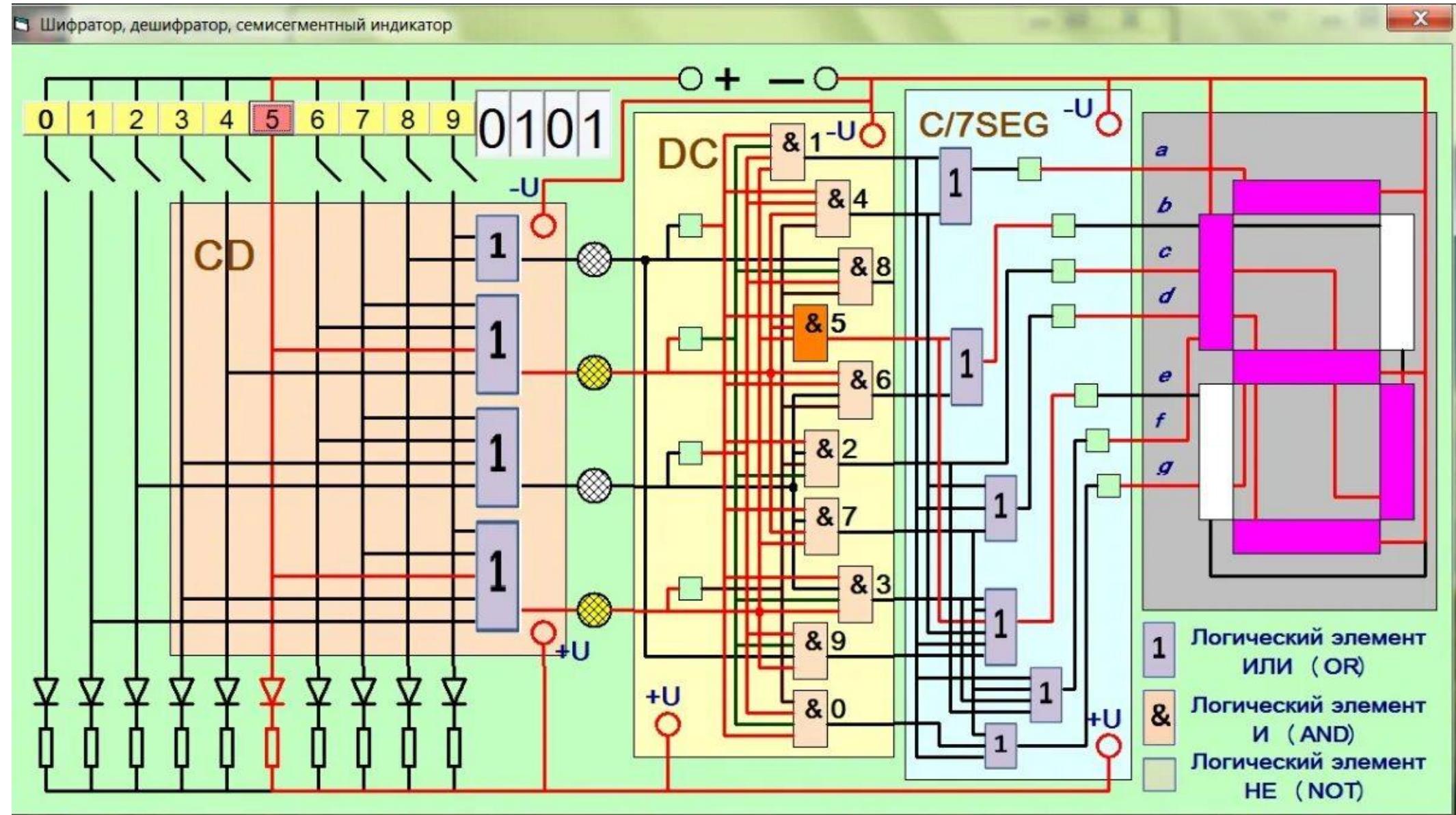
- **Наиболее распространённые микросхемы дешифраторов:**
- **74HC138** — 3 кода на входе, 8 выходов, входы разрешения.
- **74LS42** — 4 кода на входе, 10 выходов.
- **CD4514** — 4 кода на входе, 16 выходов.



- **Наиболее распространённые микросхемы шифраторов:**
- **74HC148** — 8 входов, 3 кода на выходе, приоритетный.
- **74LS147** — 10 входов, 4 кода на выходе.



Шифратор/Дешифратор/Семисегментный индикатор



Шифратор – Пример: Клавиатура

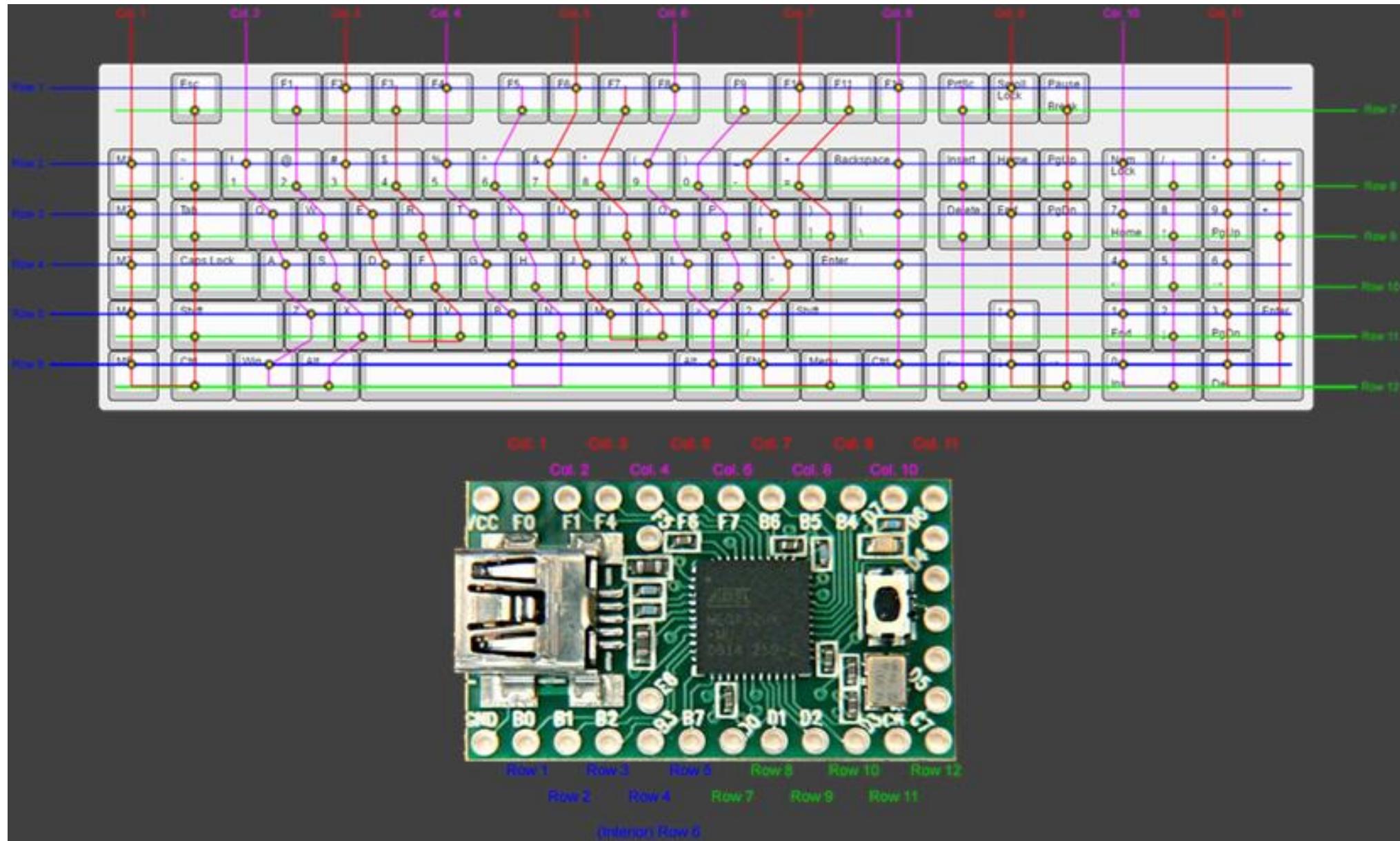


- В качестве примера **рассмотрим двоичное шифрование клавиш стандартной клавиатуры** IBM-PC-совместимых компьютеров в виде так называемых «сканированных кодов» (скан-кодов).
- На стандартной клавиатуре размещено **104 клавиши** ($N=104$).
- Чтобы их закодировать двоичным кодом, необходимо $n=7$ разрядов, так как должно выполняться соотношение $2^{n-1} < N < 2^n$.
- Но поскольку $N=104 < 2^7 = 128$, то шифратор клавиатуры не является полным, т. к. отдельные клавиши не задействованы.
- Во многих клавиатурах дополнительно используются еще набор специализированных клавиш и тогда шифратор можно назвать полным.

Скан-коды первых 64-х клавиш стандартной клавиатуры

Клавиша	Скан-код	Клавиша	Скан-код	Клавиша	Скан-код	Клавиша	Скан-код
(Отсутствует)	0000000	Q Й	0010000	D В	0100000	В И	0110000
ESC	0000001	W Ц	0010001	F А	0100001	Н Т	0110001
1 !	0000010	Е У	0010010	G П	0100010	М Ъ	0110010
2 @ «»	0000011	R К	0010011	Н Р	0100011	, < Б	0110011
3 # №	0000100	Т Е	0010100	J О	0100100	. > Ю	0110100
4 \$;	0000101	Y Н	0010101	К Л	0100101	/ ? . ,	0110101
5 %	0000110	U Г	0010110	L Д	0100110	Shift	0110110
6 ^ :	0000111	I Ш	0010111	; : Ж	0100111	Alt (левый)	0110111
7 & ?	0001000	O Щ	0011000	‘ “ Э	0101000	(Space bar)	0111000
8 *	0001001	P З	0011001	‘ ~ Ё	0101001	Caps Lock	0111001
9 (0001010	[{ X	0011010	Shift	0101010	F1	0111010
0)	0001011] } Ъ	0011011	\ /	0101011	F2	0111011
- _	0001100	Enter	0011100	Z Я	0101100	F3	0111100
= +	0001101	Ctrl (левый)	0011101	X Ч	0101101	F4	0111101
Backspace	0001110	А Ф	0011110	С С	0101110	F5	0111110
Tab	0001111	S Ы	0011111	V М	0101111	F6	0111111

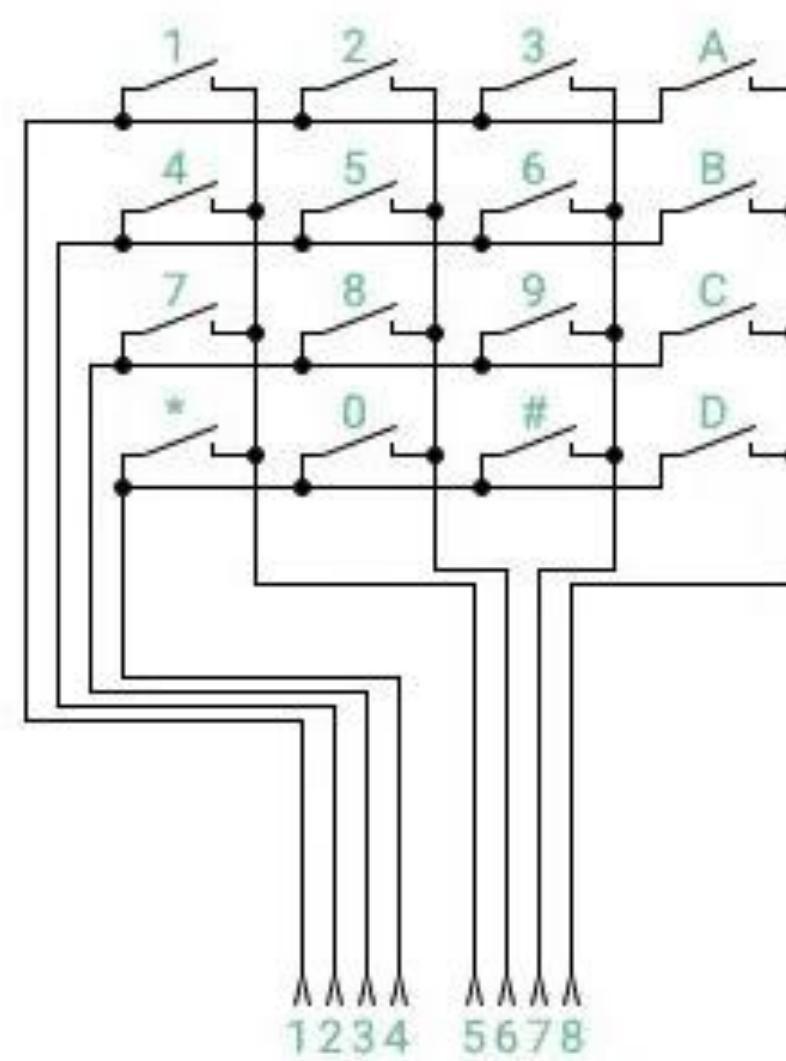
Принцип работы клавиатуры



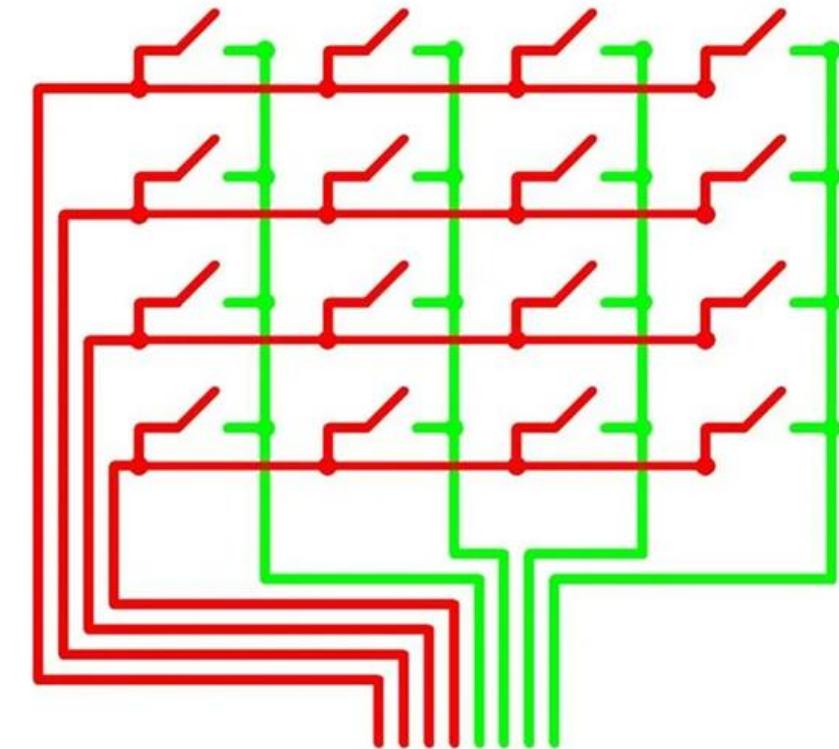
Мембранные клавиатура 4x4

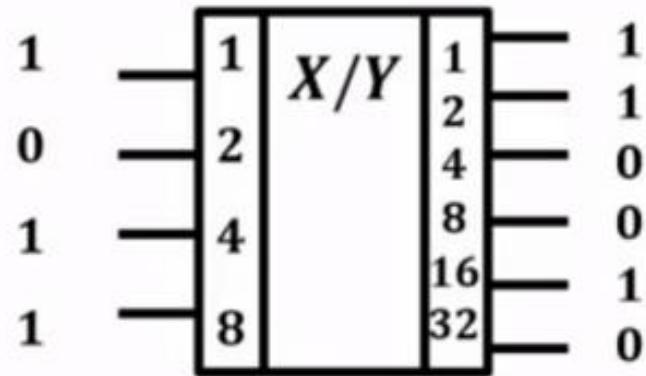


12345678



1234 5678





Базовые комбинационные блоки Преобразователь кодов



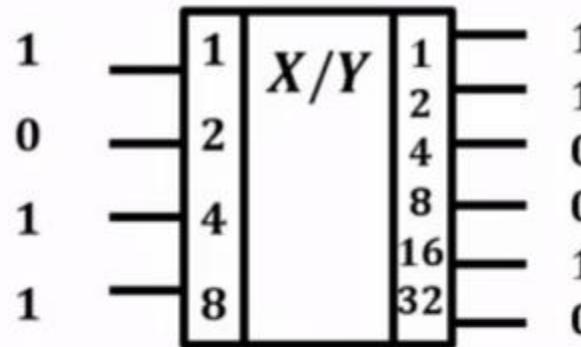
Преобразователь кодов

- **Преобразователь кодов** — это комбинационная схема, предназначенная для перевода данных из одного формата (кода) в другой.
- Он работает без памяти, то есть выход зависит только от текущих входных сигналов (как и все схемы комбинационной логики).
- **Преобразователь кодов** — общее название для всех схем, переводящих данные из одного формата в другой, по заданному правилу.
- **Примеры:**
 - **Дешифратор** (двоичный код → унитарный код).
 - **Шифратор** (унитарный код → двоичный код).
 - **BCD** → 7-сегментный код.
 - **Код Грея** → двоичный код.
 - **Параллельный код** → последовательный.

Преобразователь кодов

- Основные функции преобразователей кодов:
 - **Перекодирование**, то есть изменение кода числа из одного формата в другой;
 - **Поддержка аппаратного быстродействия** в цифровых устройствах и микроконтроллерах;
 - **Обеспечение удобства ввода и вывода данных** (например, с десятичных кнопок на двоичный формат для обработки компьютером);
 - **Выполнение дополнительных логических операций**, например, умножение на весовые коэффициенты.
 - **Частными случаями преобразователей кодов являются шифраторы и дешифраторы** — устройства, которые выполняют преобразование десятичного кода в двоичный и обратно.

Преобразователь кодов



A	B	C	Y2	Y1	Y0	DEC
0	0	0	0	1	0	2
0	0	1	1	1	0	6
0	1	0	0	0	1	1
0	1	1	1	0	1	5
1	0	0	0	0	1	1
1	0	1	1	0	0	4
1	1	0	0	1	1	3
1	1	1	1	1	0	6

	A	\bar{A}	
B	0	1	1
\bar{B}	0	1	1
\bar{C}	C	C	\bar{C}

$$Y2 = C$$

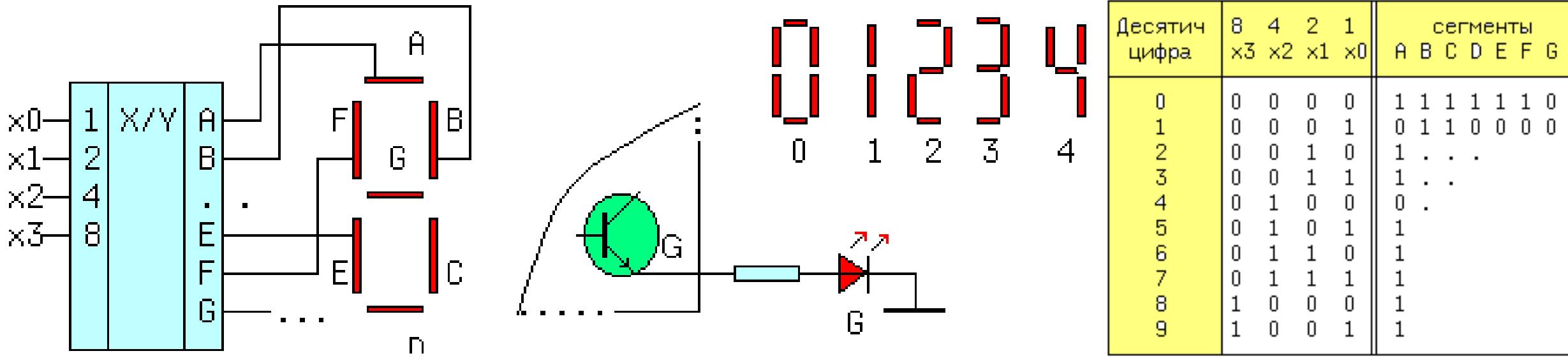
	A	\bar{A}	
B	1	1	0
\bar{B}	0	0	1
\bar{C}	C	C	\bar{C}

$$Y1 = AB + \overline{AB}$$

	A	\bar{A}	
B	1	0	1
\bar{B}	1	0	0
\bar{C}	C	C	\bar{C}

$$Y0 = \overline{AB} \cdot AC$$

Преобразователь кодов

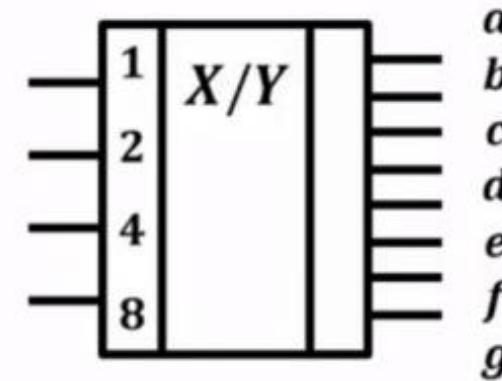


Преобразователи кодов бывают разных **типов**:

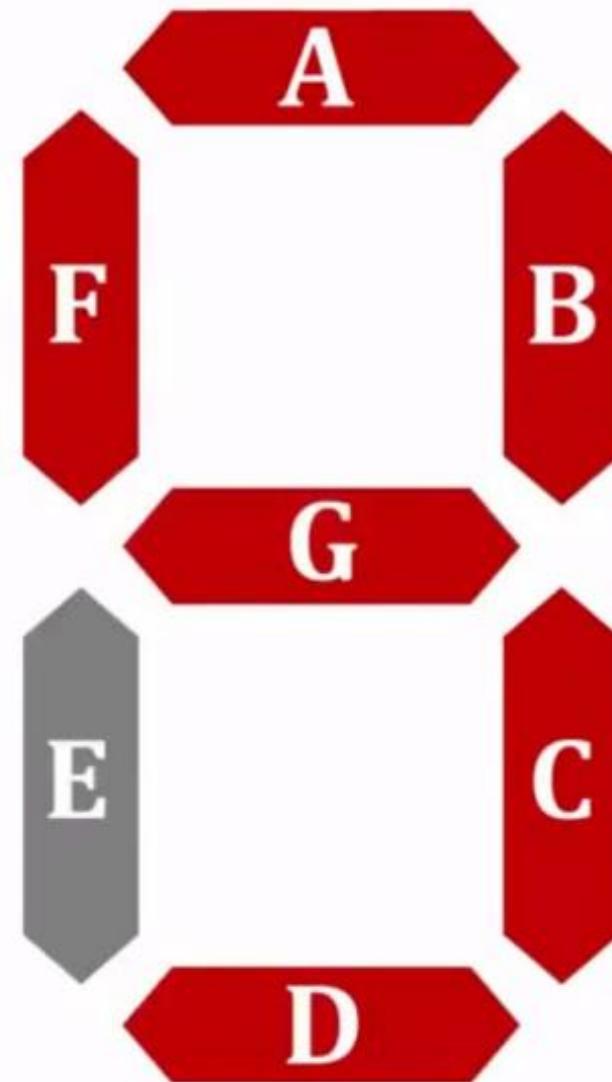
- **Весовые преобразователи** — преобразуют информацию из одной системы счисления в другую (например, двоичный в десятичный код и наоборот). В этом случае каждый разряд исходного кода соответствует определённому весу, и преобразование происходит с учётом этих весов. Весовые преобразователи используются, когда необходимо сохранить численное значение данных при смене формата.;
- **Невесовые преобразователи** — изменяют код для дальнейшего отображения или управления. Невесовые преобразователи кодов изменяют форму кода без сохранения весов разрядов, то есть они выполняют преобразование между кодами, которые не обязательно имеют прямое числовое соответствие. Примером является **преобразование двоично-десятичного кода в формат, управляющий семисегментным индикатором**.

Основное отличие в том, что весовые преобразователи сохраняют числовой вес и значение при перекодировании, а невесовые — выполняют логическую трансформацию кода для управления или отображения без сохранения весов разрядов.

Преобразователь кодов



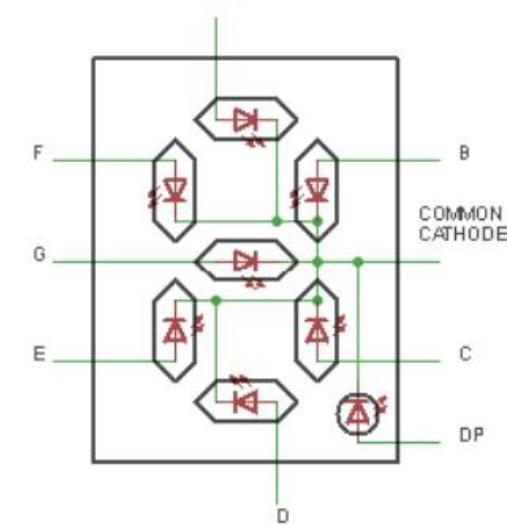
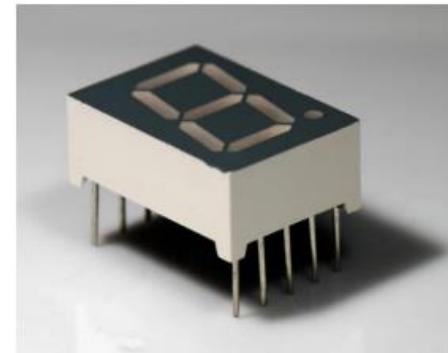
DEC	X3	X2	X1	X0	A	B	C	D	E	F	G
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1



Код для семисегментного индикатора

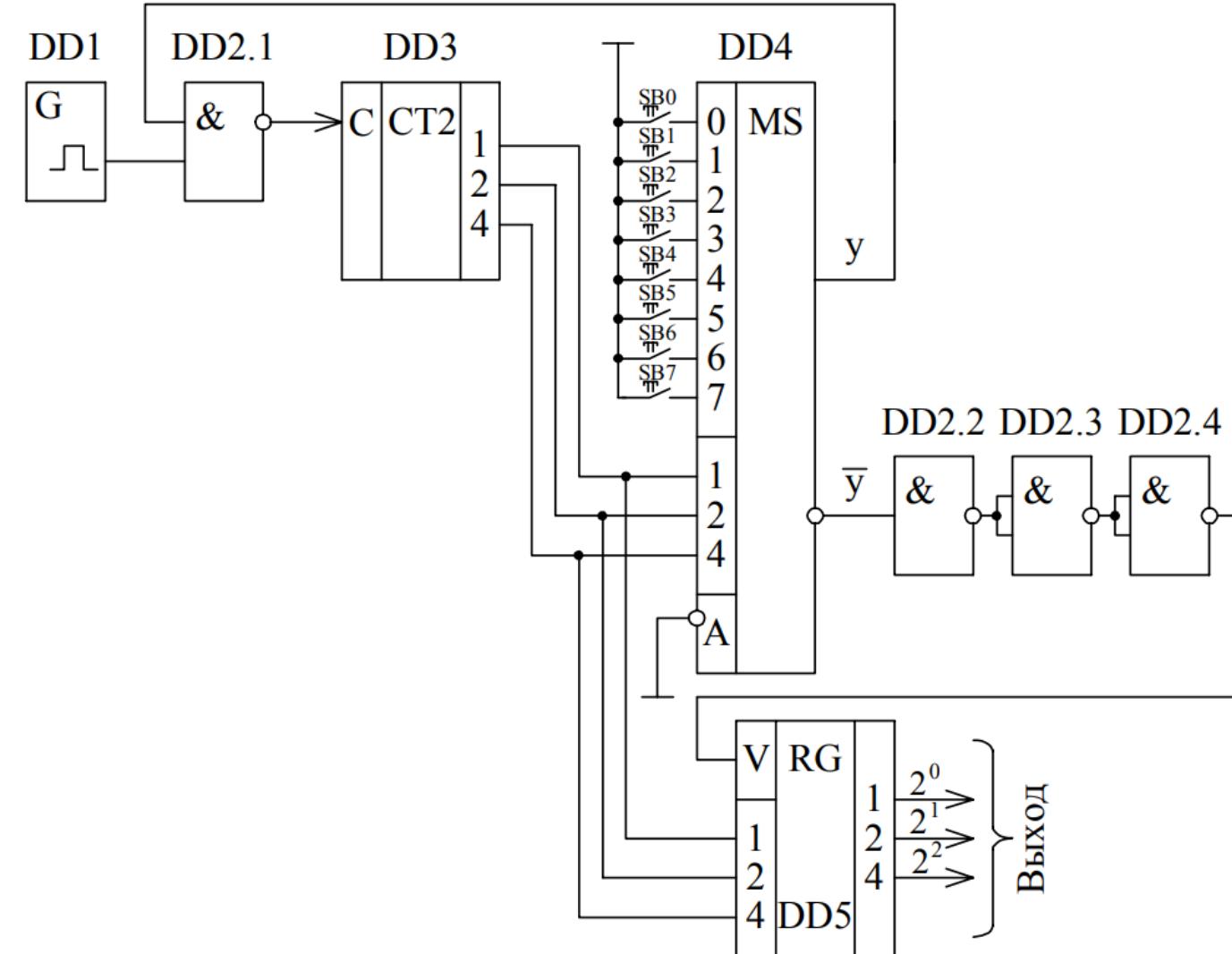
Дешифратор / Преобразователь кодов

- Существуют **дeшифраторы, преобразующие двоичный код в семисегментный**. Используется для управления светодиодными семисегментными индикаторами.
- В этом случае **их называют преобразователями кодов**.
- В соответствие с двоичным кодом на выходе дешифратора на индикаторе высвечиваются цифры или буквы.
- Пример такого дешифратора – микросхема К155ПП5.**



7

Преобразователь кодов



Функциональная схема преобразователя десятичного кода в двоичный код

Преобразователь кодов

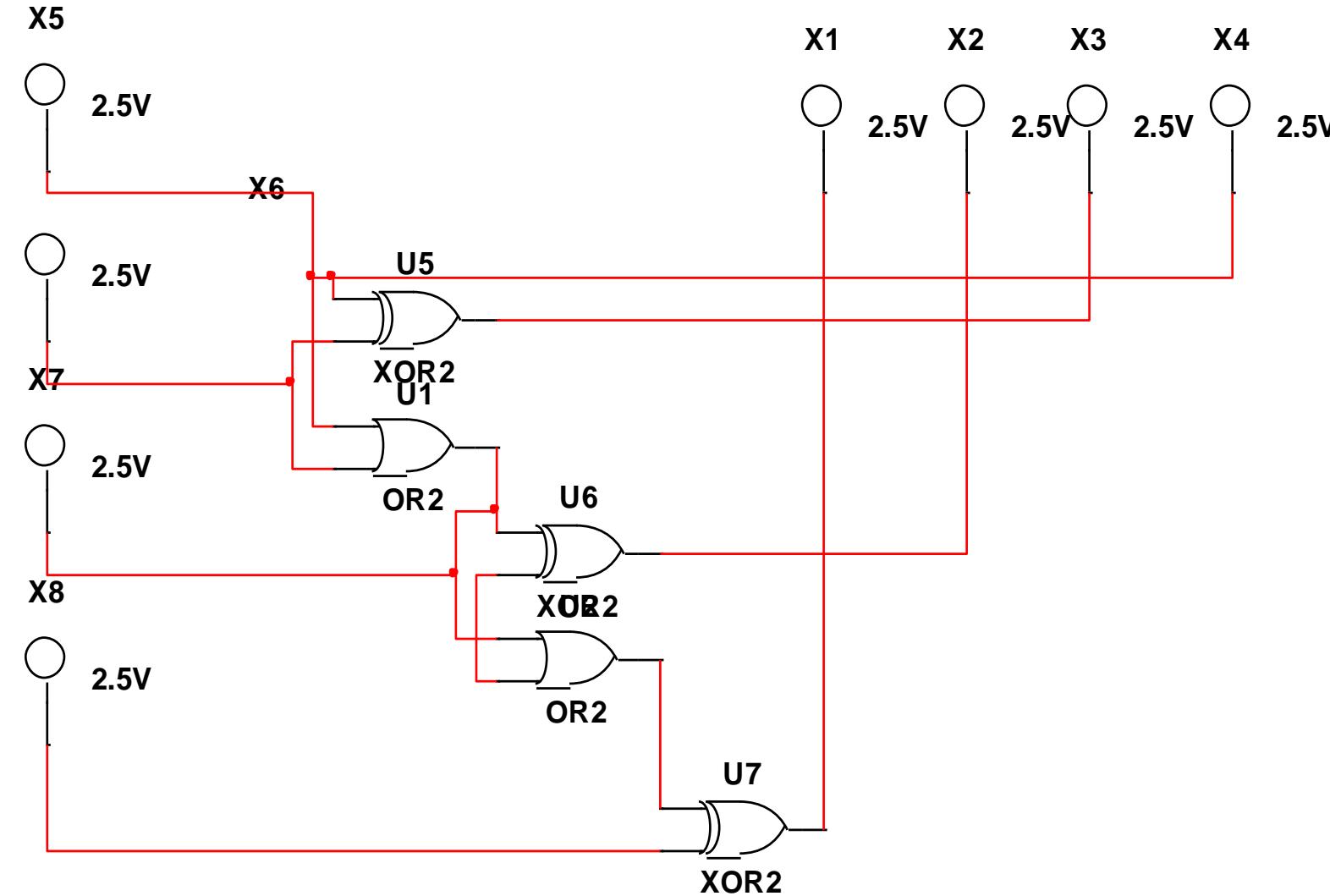
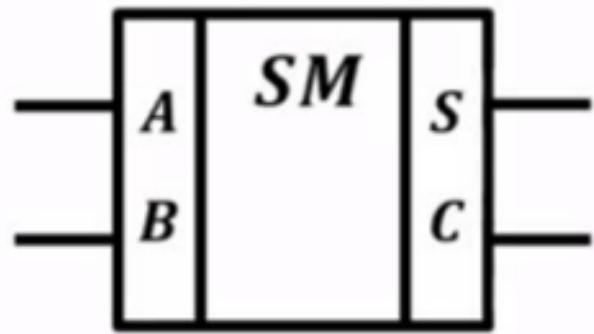


Схема преобразователя кодов



Базовые комбинационные блоки

Сумматоры (Вычитатели)



Сумматоры

- Сумматоры выполняют арифметическое сложение чисел.
- Они являются также ядром арифметико-логических устройств (АЛУ), входящих в состав процессоров.
- **Сумматор** — это электронное устройство или логическая схема, предназначенная для выполнения операции сложения чисел или сигналов.
- В зависимости от типа сумматор может работать как с аналоговыми, так и с цифровыми данными.
- **Сумматоры широко используются в различных областях электроники, включая:**
 - **Цифровую технику** (процессоры, микроконтроллеры, арифметико-логические устройства).
 - **Аналоговую электронику** (операционные усилители, схемы обработки сигналов).
 - **Телекоммуникации** (антенные сумматоры, схемы объединения сигналов).
 - **Автоматику и системы управления** (контрольные устройства, измерительные системы).
- В цифровой электронике сумматоры являются базовыми элементами процессоров и вычислительных схем. В аналоговой электронике они применяются для обработки сигналов, создания смесителей и усилителей.

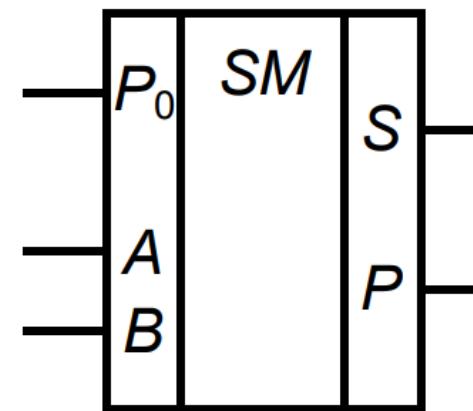
Полусумматор и полный сумматор

Сумматоры – комбинационные устройства, выполняющие функцию сложения чисел.

Задача. Сложить два числа.

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

A, B – одноразрядные числа;
 S – сумма;
 P - перенос.



УГО сумматора

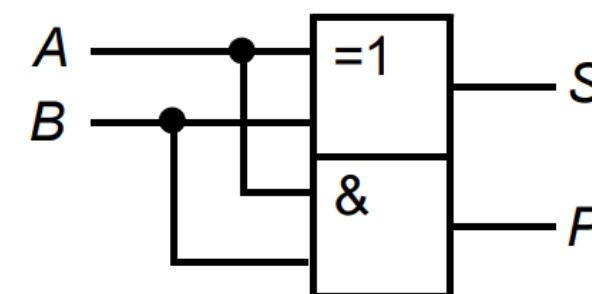
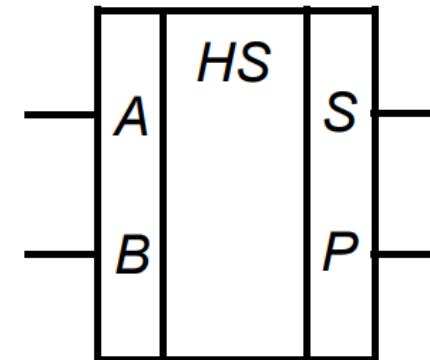


Схема одноразрядного полусумматора

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

$$P = A \cdot B$$



УГО полусумматора

Полный сумматор, в отличие от полусумматора, учитывает результат предыдущего сложения и для этого имеет вход переноса из предыдущего разряда.

Классификация сумматоров

• По назначению:

- **Сумматор чисел** - Используется для сложения числовых значений, представленных в двоичной, десятичной или другой системе счисления. Применяется в процессорах, микроконтроллерах и калькуляторах.
- **Сумматор сигналов** - Предназначен для суммирования электрических сигналов. Используется в аналоговой электронике, например, в схемах усилителей.
- **Антенный сумматор** - Применяется в радиотехнике для объединения сигналов от нескольких антенн в один общий выходной сигнал. Используется в телевидении, радиосвязи и беспроводных сетях.
- **Усилитель-сумматор** - Это особый тип операционного усилителя (ОУ), который суммирует входные сигналы и усиливает их. Используется в аудиоаппаратуре и системах обработки сигналов.

• По типу обработки данных:

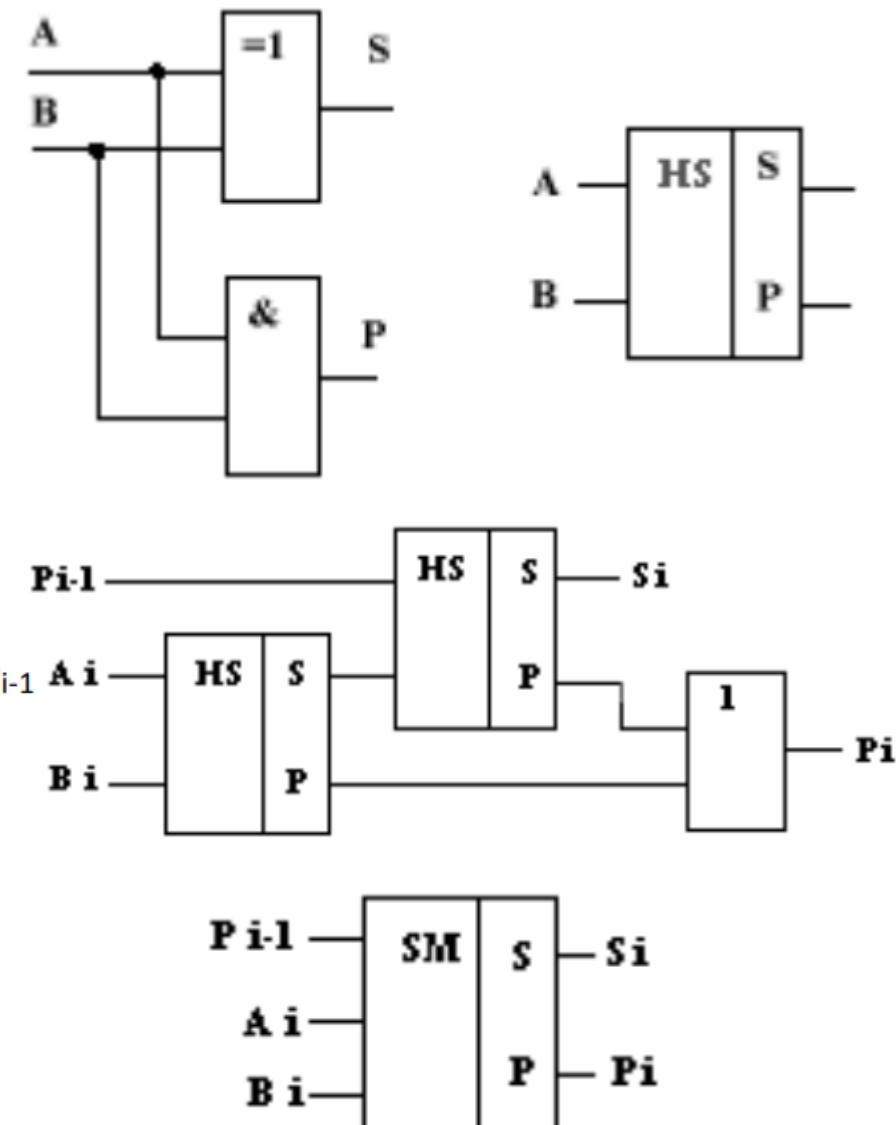
- **Двоичный сумматор** - Обрабатывает числа в двоичной системе счисления. Используется в процессорах и микроконтроллерах.
- **Логический сумматор** - Выполняет логическую операцию сложения (исключающее ИЛИ – XOR) и может быть частью арифметико-логического устройства.
- **Аналоговый сумматор** - Обрабатывает аналоговые сигналы и используется в схемах с операционными усилителями.

Классификация сумматоров

- **По структуре:**
 - **Полусумматор и полный сумматор**
 - **Полусумматор** выполняет сложение двух однобитных чисел, но не учитывает перенос.
 - **Полный сумматор** учитывает перенос, что делает его более функциональным в сложных вычислительных системах.
 - **Одноразрядный и многоразрядный сумматор**
 - **Одноразрядный сумматор** складывает только два бита.
 - **Многоразрядный сумматор** позволяет сложение многобитных чисел, используя несколько одноразрядных сумматоров, соединенных каскадом.
 - **Параллельный и последовательный сумматор**
 - **Параллельный сумматор** выполняет сложение всех разрядов одновременно, что делает его быстрым, но требующим больше аппаратных ресурсов.
 - **Последовательный сумматор** складывает числа разряд за разрядом, что уменьшает аппаратные затраты, но замедляет вычисления.
 - **Комбинационный сумматор** - Не содержит памяти и выполняет операцию сложения в один такт. Используется в арифметико-логических устройствах процессоров.

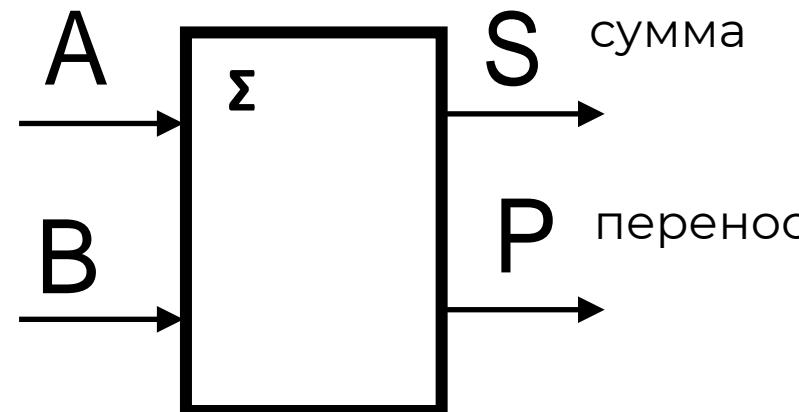
Сумматор и Полусумматор

- Простейшей арифметической операцией в интегральной логике является сложение двух одноразрядных чисел, принимающих два возможных значения: 0 и 1.
- Эта операция выполняется в устройстве, называемом **полусумматором**.
- На выходе S элемента "исключающее ИЛИ" получается "сумма по модулю 2", равная нулю, когда $A = B = 0$, а также когда $A = B = 1$.
- При $A = 1$ и $B = 0$ или $A = 0$ и $B = 1$ значение $S = 1$.
- Выход P называется "переносом в следующий разряд".
- Если $A = B = 1$, то перенос $P = 1$, во всех остальных случаях $P = 0$.
- При сложении двух многоразрядных двоичных чисел только в младшем разряде складываются два числа.
- В остальных разрядах складываются три числа: два слагаемых и перенос из суммы чисел предыдущего разряда.
- Эти функции реализует **полный сумматор**, состоящий из двух полусумматоров и элемента ИЛИ.

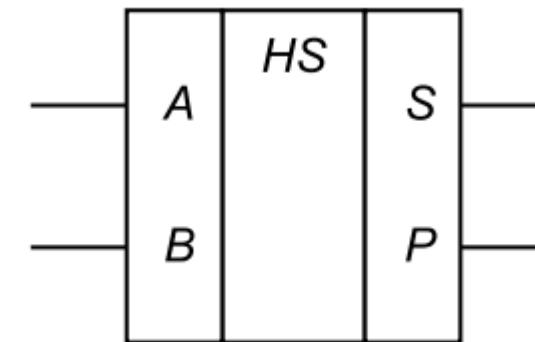


Полусумматор

- **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

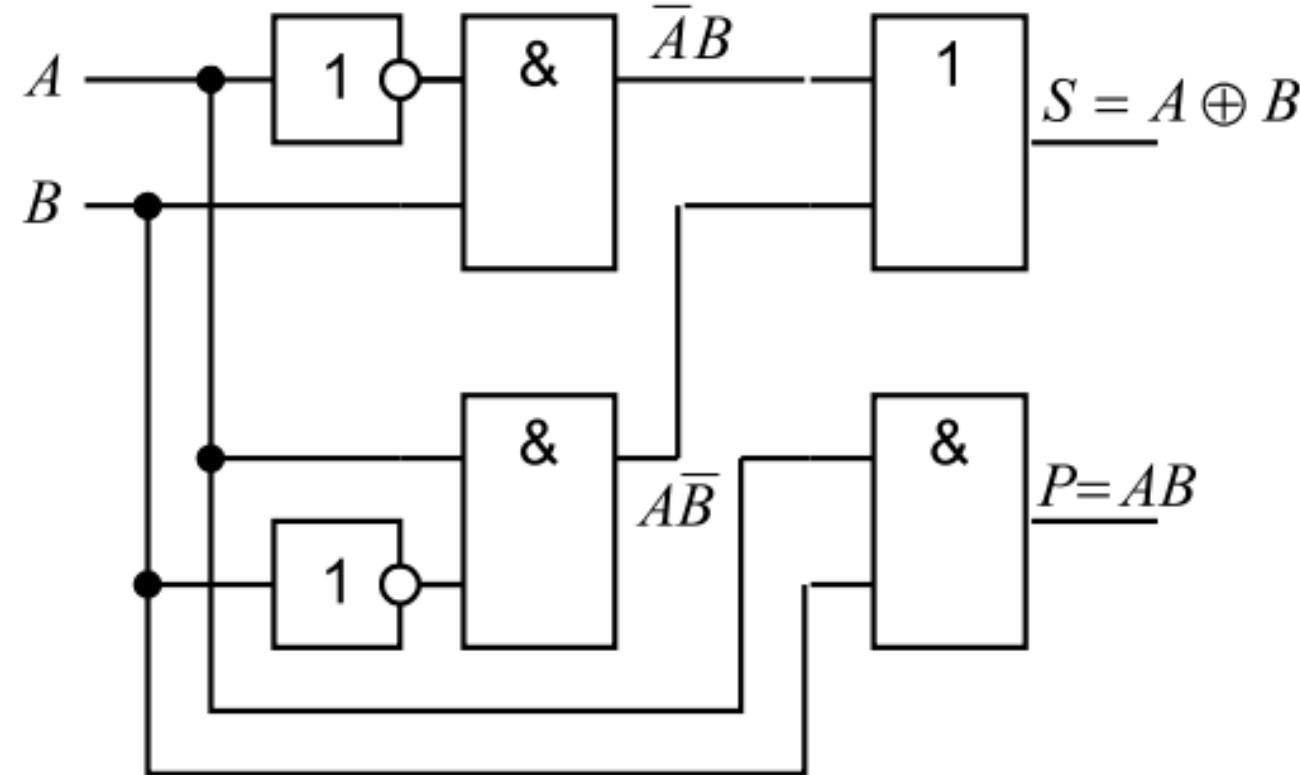
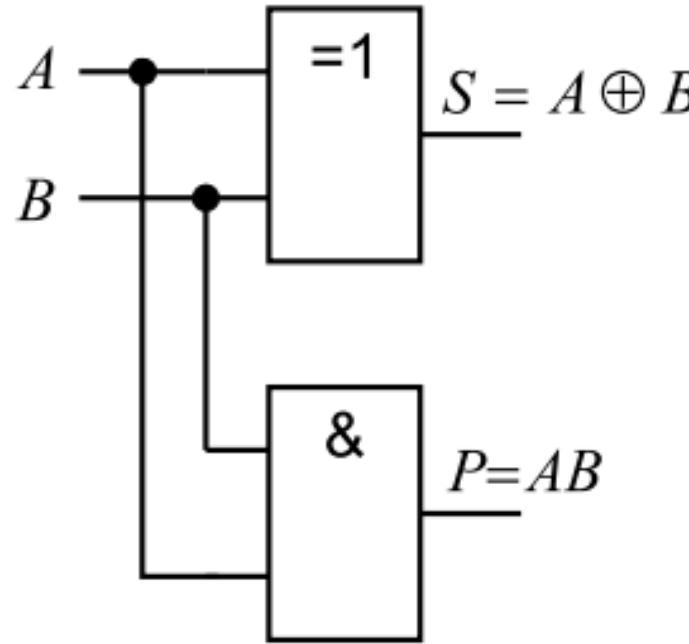


Булевы функции, описывающие работу полусумматора, имеют вид

$$S = \overline{A}B + A\overline{B} = A \oplus B \quad P = AB.$$

Полусумматор

- Логическая структура полусумматора в общем и развернутом видах



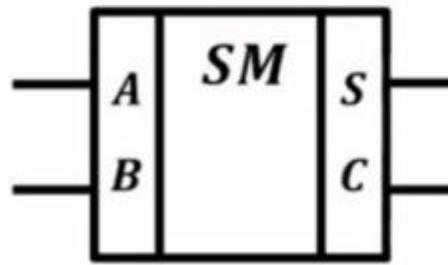
Полусумматор имеет два входа и поэтому пригоден для использования только в младшем разряде многоразрядных двоичных чисел. Начиная со второго разряда многоразрядных чисел, необходимо использовать полный одноразрядный сумматор, содержащий три входа, на один из которых подается сигнал переноса из предыдущего разряда.

Полусумматор

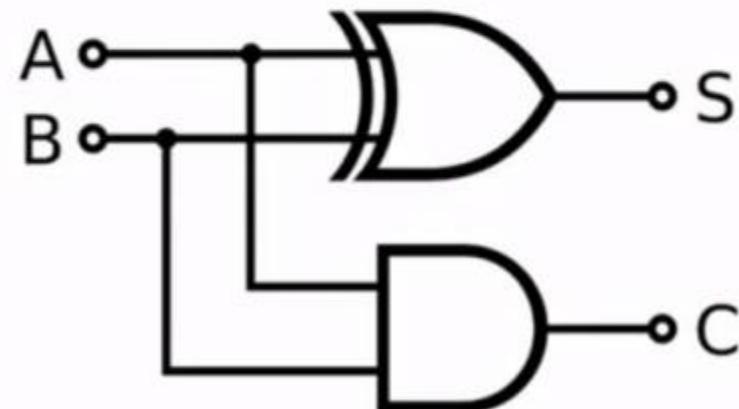
$$\begin{array}{r} 1 \ 1 \\ + 11 \\ \hline 110 \end{array}$$

$$\begin{array}{r} 1 \\ + 1 \\ \hline 10 \\ C \ S \end{array}$$

S — сумма
C — перенос (англ. carry)



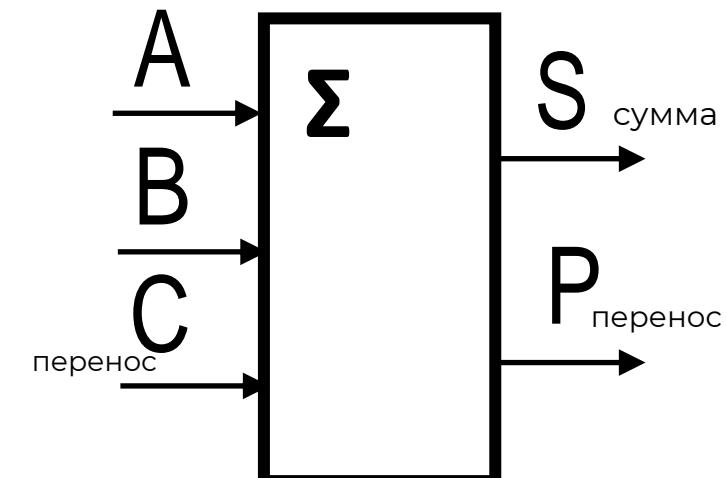
A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0



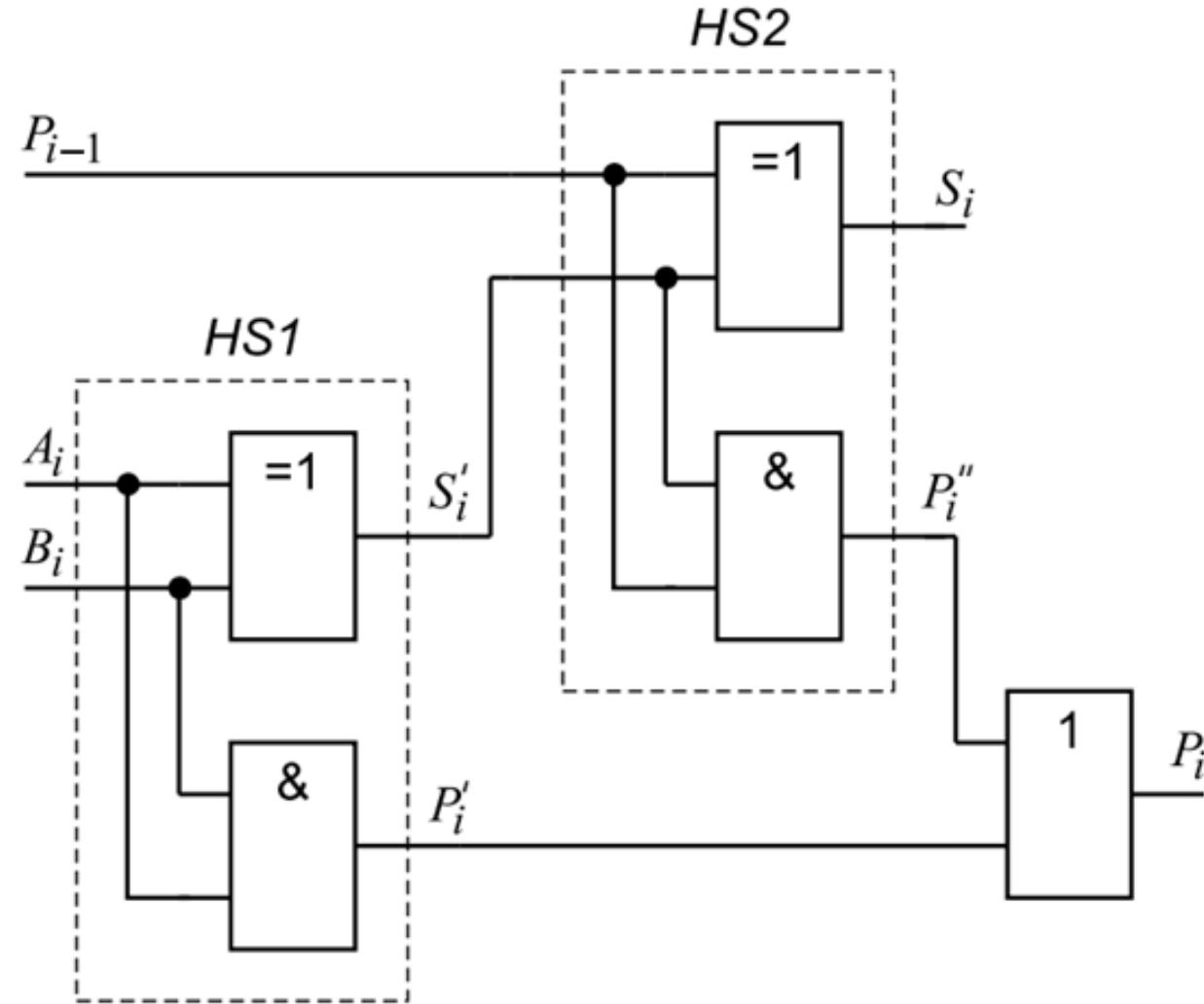
Сумматор

- Сумматор - это электронная логическая схема, выполняющая суммирование двоичных чисел поразрядным сложением.
- Сумматор – это логическая схема, способная складывать два одноразрядных двоичных числа с переносом из предыдущего разряда.
- Сумматор является центральным узлом арифметико-логического устройства процессора. Находит он применение и в других устройствах компьютера.
- Сумматор выполняет сложение многозначных двоичных чисел. Он представляет собой последовательное соединение одноразрядных двоичных сумматоров, каждый из которых осуществляет сложение в одном разряде. Если при этом возникает переполнение разряда, то перенос суммируется с содержимым старшего соседнего разряда.

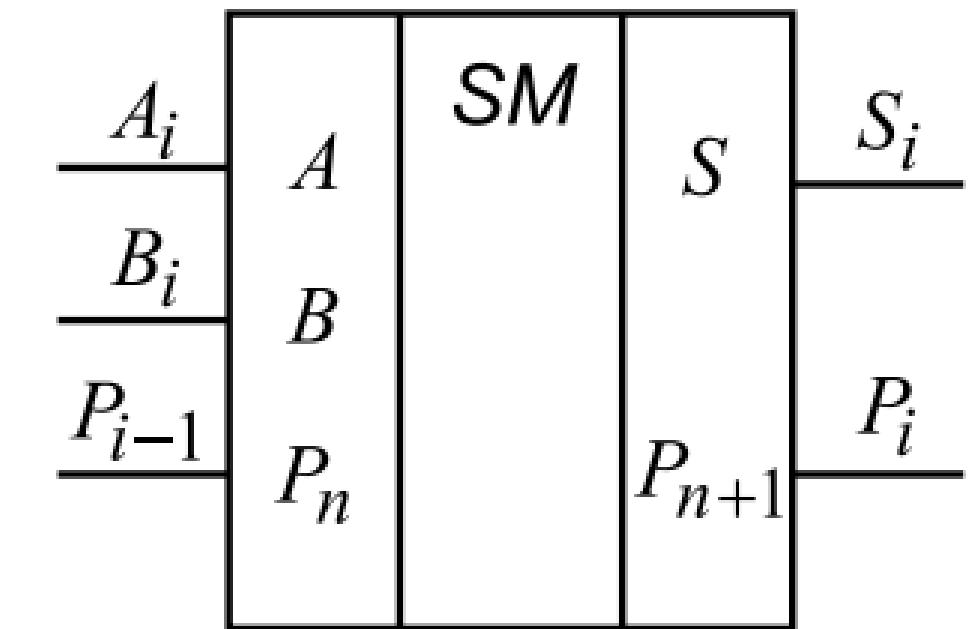
A	B	C	P	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



Полный одноразрядный сумматор

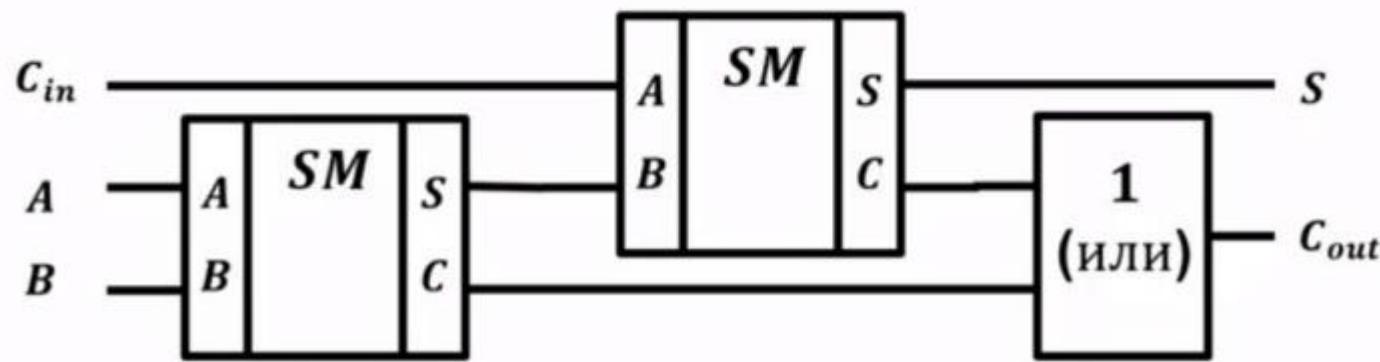
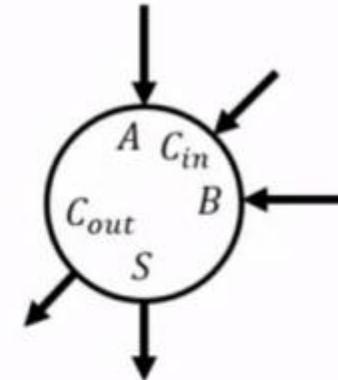
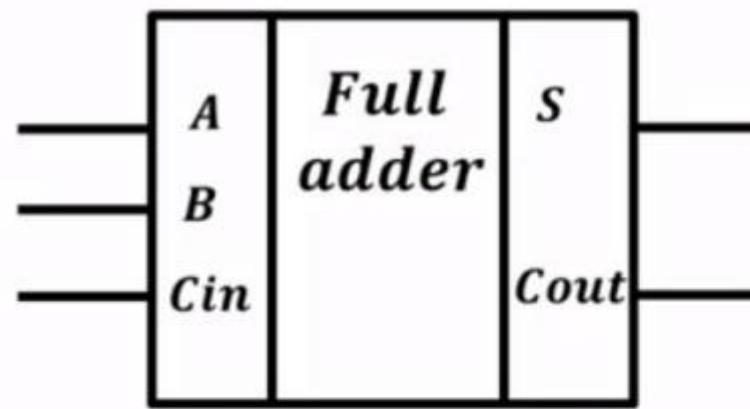


реализация на полусумматорах



условное графическое обозначение

Полный сумматор

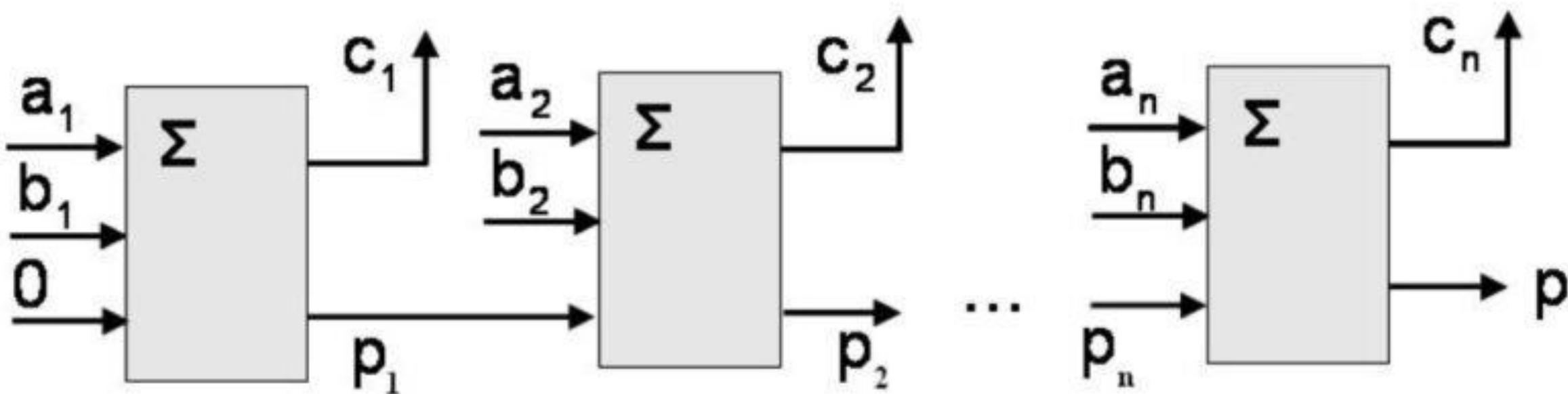


Cin	A	B	Cout	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

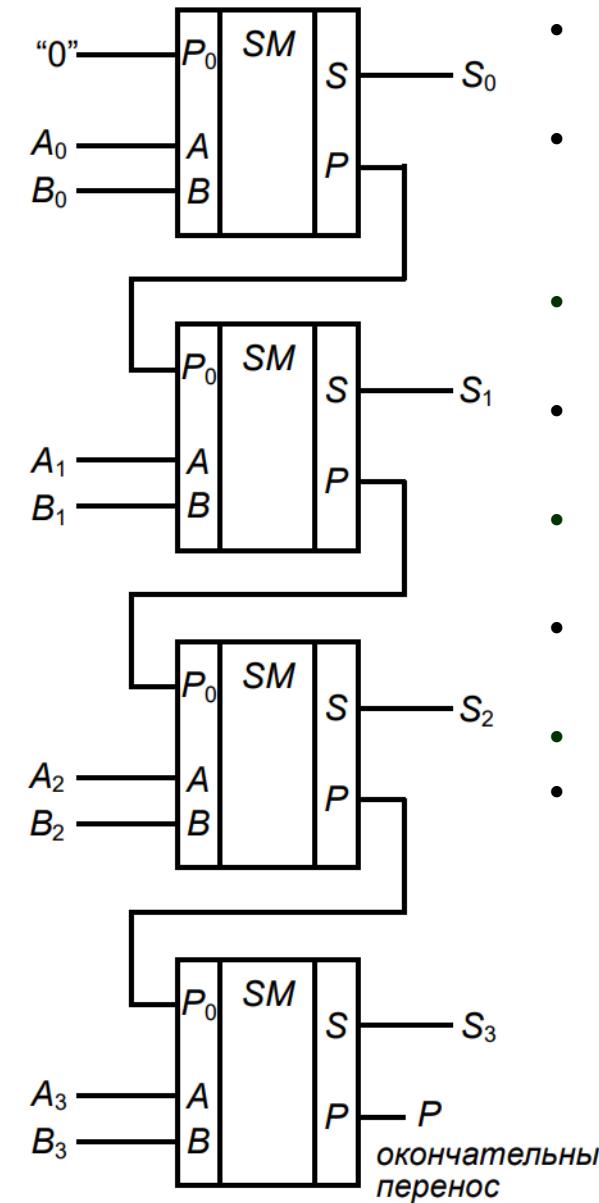
Полные сумматоры двух бит можно соединять в **каскады**, и таким образом получать сумматоры для чисел **любой битности!**

Многоразрядный сумматор

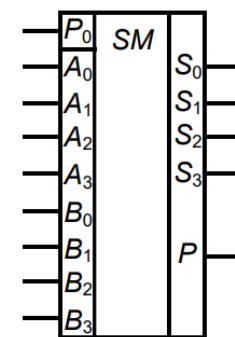
- Многоразрядные сумматоры позволяют складывать числа, содержащие несколько битов. Они строятся на основе нескольких одноразрядных полных сумматоров.



Многоразрядный сумматор



- На базе одноразрядного полного сумматора строятся многоразрядные сумматоры.
- Параллельный сумматор осуществляет сложение двух 4-х разрядных чисел.
- Окончательный перенос необходим для дальнейшего увеличения разрядности сумматора.
- В рассмотренной схеме перенос выполняется последовательно из разряда в разряд, что занимает некоторое время.
- При суммировании многоразрядных чисел это время **значительно!** и именно оно определяет время суммирования.
- С целью уменьшения времени суммирования вместо последовательного используют параллельный перенос.
- Такой перенос реализован в 4-х разрядном сумматоре К555ИМ6.
- В ИМС выпускаются одно-, двух- и 4-х разрядные двоичные сумматоры.



УГО 4-х разрядного сумматора К155ИМ3

Регистровый сумматор

- **Регистровый сумматор** — это устройство, в котором результаты сложения временно сохраняются в регистрах.
- **Взаимодействие с регистрами**
 - Регистры принимают данные, хранят их и передают на вход сумматора.
 - После сложения результат записывается обратно в регистр.
 - Используется в арифметико-логических устройствах (АЛУ).
- **Применение в цифровых схемах**
 - Микропроцессоры (для выполнения арифметических операций).
 - Контроллеры памяти.
 - Программируемые логические устройства.

Вычитатели

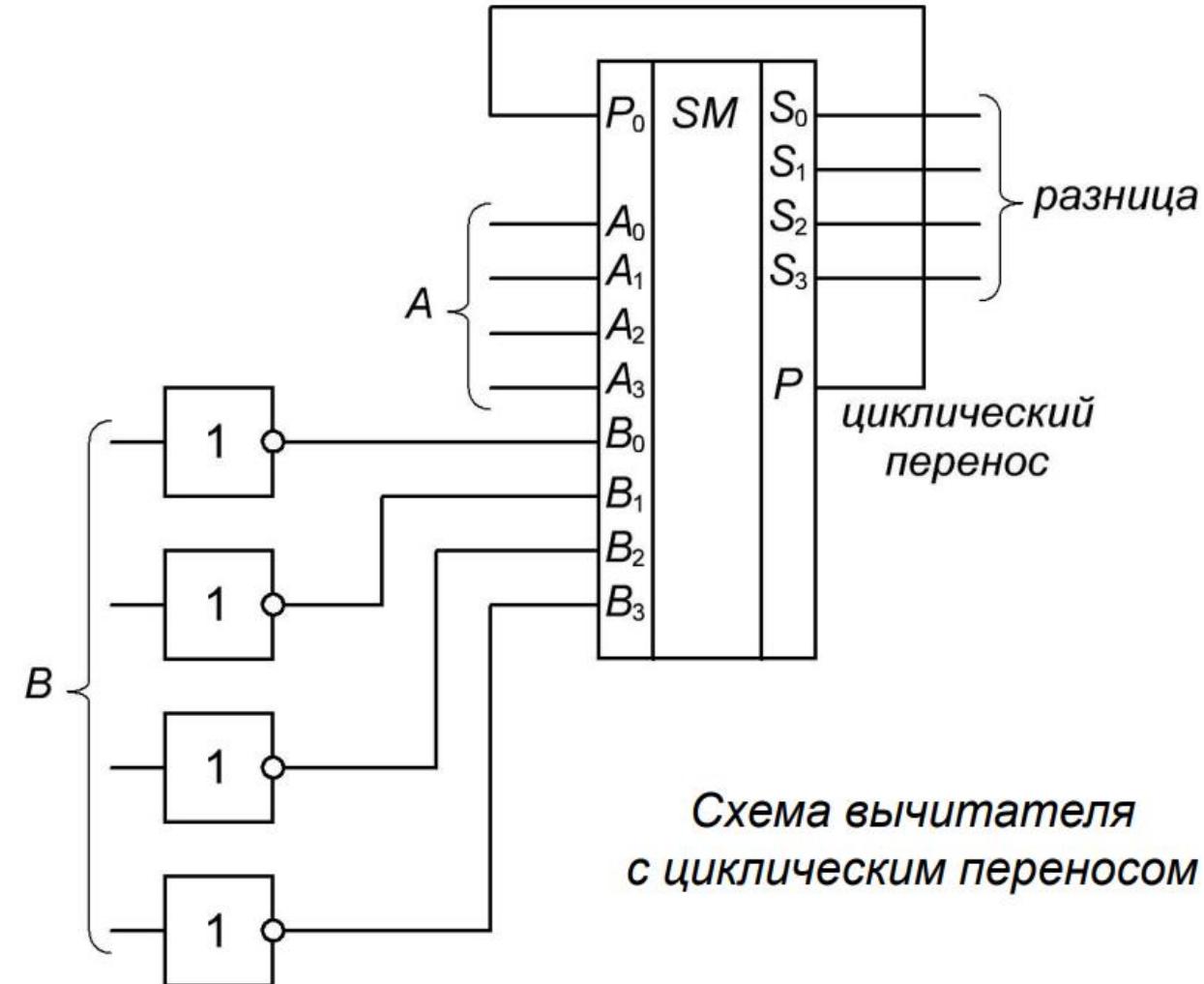
- Вычитатели строятся на базе сумматоров и в виде отдельных устройств не предусмотрены.
- Для выполнения операции вычитания на сумматоре необходимо вычитаемое представить в обратном коде, и к результату прибавить 1.

Пример.

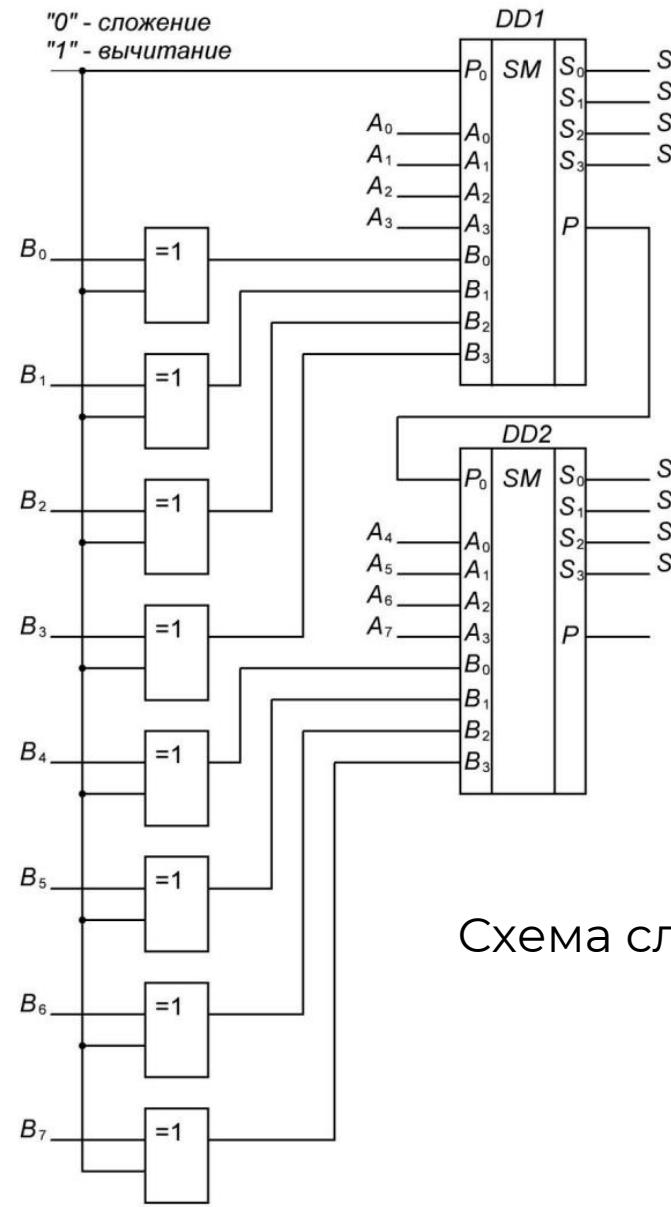
$$\begin{array}{r} -11_{10} \\ \hline -5_{10} \\ \hline 6_{10} \end{array} \quad \begin{array}{r} -1011_2 \\ -0101_2 \\ \hline 0110_2 \end{array} \quad \begin{array}{r} + 1011_2 \\ 1010_2 \\ \hline 10101_2 \\ +1_2 \\ \hline 0110_2 \end{array}$$

- промежуточный результат
перенос
- разность

- Данный метод вычитания реализуется схемой с циклическим переносом.
- Циклический перенос позволяет использовать «1» на выходе Р для сложения с промежуточным результатом вычитания.
- Возможно обойтись без циклического переноса, но в этом случае на вход $P_0 = 1$.
- Если $P = 1$, то число на выходе положительное, т.е. представлено в прямом коде.
- Если $P = 0$, то число на выходе отрицательное, т.е. представлено в обратном коде, результат вычитания нужно инвертировать.



Вычитатели



- Вычитатели строятся на базе сумматоров и в виде отдельных устройств не предусмотрены.
- Если в предыдущей схеме вычитателя с циклическим переносом вместо инверторов поставить элементы «исключающего ИЛИ», то на одной ИМС можно производить и сложение и вычитание.

Схема сложения-вычитания

Вычитатель / Сумматор

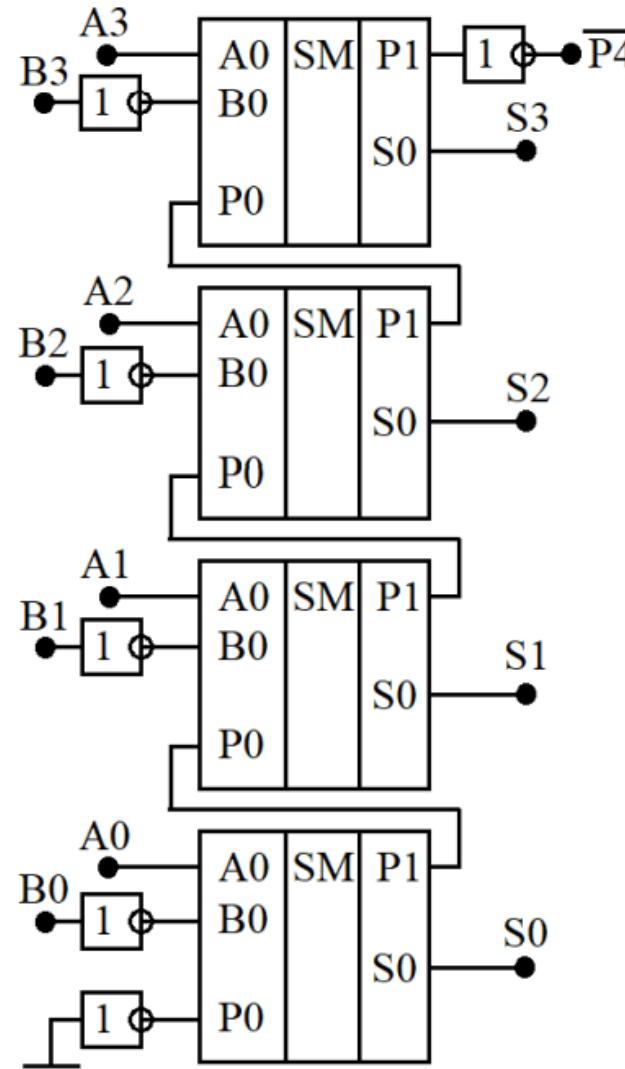


Схема 4-разрядного сумматора
для вычитания двоичных чисел

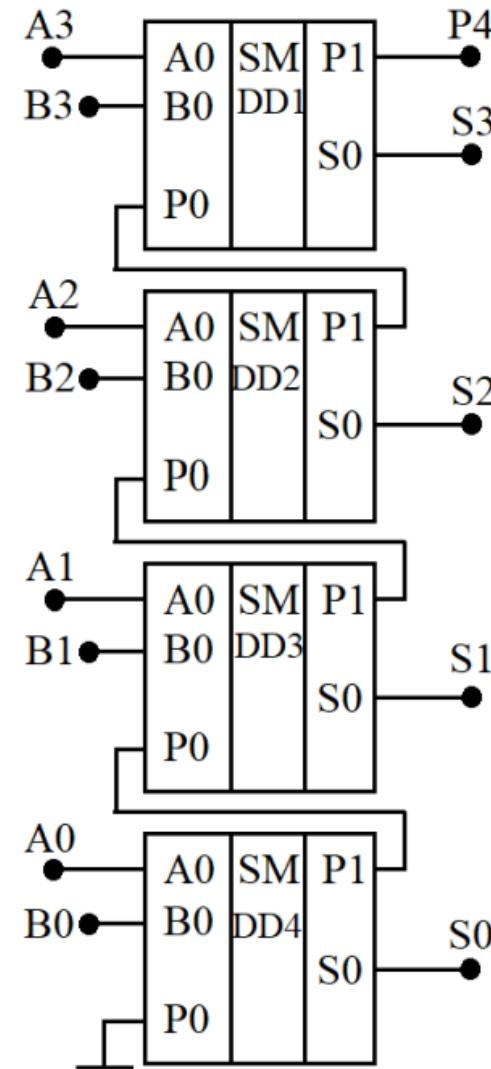


Схема 4-разрядного сумматора
для сложения двоичных чисел

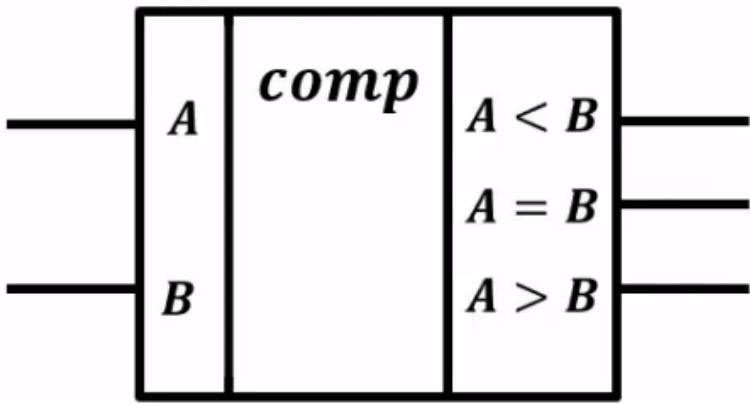
Применение Сумматоров в Электронике

- Сумматоры являются основными компонентами цифровых схем, выполняя ключевые арифметические и логические операции.
- Сумматоры используются в процессорах и микроконтроллерах**
- В процессорах и микроконтроллерах сумматоры выполняют сложение чисел в:**
 - Арифметико-логическом устройстве (АЛУ)** — центральной части процессора, где выполняются арифметические и логические операции.
 - Операциях сложения адресов** — например, при вычислении следующего адреса инструкции или данных в памяти.
 - Реализации счётчиков и таймеров**, где сумматор используется для увеличения значений.
- Пример:** В процессоре архитектуры x86 сложение выполняется инструкцией ADD, где используется многоразрядный сумматор.

Сумматор двоичных чисел

В целях максимального упрощения работы компьютера все многообразие математических операций в процессоре сводится к сложению двоичных чисел.

Поэтому главной частью процессора является сумматор, который как раз и обеспечивает такое сложение.



Базовые комбинационные блоки Компараторы



Цифровые компараторы

- Цифровые компараторы осуществляют сравнение двух чисел в двоичном коде.
- Цифровой компаратор имеет три выхода: $A > B$, $A = B$, $A < B$.
- Компаратор можно построить на сумматоре, если производить на нем вычитание двух сравниваемых величин.

Пусть $A > B$

$$\begin{aligned}A &= 1110_2 = 14_{10} \\B &= 1100_2 = 12_{10}\end{aligned}$$

$$\begin{array}{r} + \quad 1110 \text{ - число в прямом коде} \\ + \quad 0100 \text{ - число в дополнительном коде} \\ \hline P = 10010 \end{array}$$

↑
перенос

$$\begin{array}{|c|c|} \hline P = 1 & S \neq 0 \\ \hline \end{array}$$

$A > B$

Пусть $A = B$

$$\begin{aligned}A &= 1110_2 = 14_{10} \\B &= 1110_2 = 14_{10}\end{aligned}$$

$$\begin{array}{r} + \quad 1110 \\ + \quad 0010 \\ \hline P = 10000 \end{array}$$

$$\begin{array}{|c|c|} \hline P = 1 & S = 0 \\ \hline \end{array}$$

$A = B$

Пусть $A < B$

$$\begin{aligned}A &= 1100_2 = 12_{10} \\B &= 1110_2 = 14_{10}\end{aligned}$$

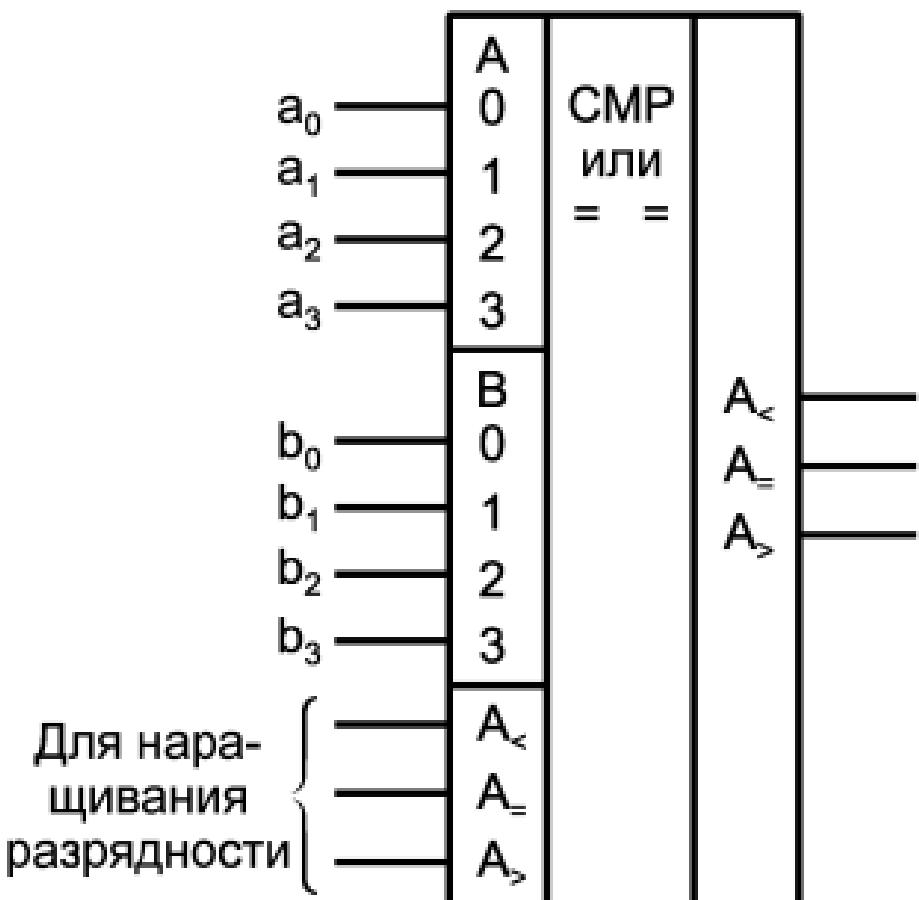
$$\begin{array}{r} + \quad 1100 \\ + \quad 0010 \\ \hline P = 01110 \end{array}$$

$$\begin{array}{|c|c|} \hline P = 0 & S \neq 0 \\ \hline \end{array}$$

$A < B$

Компаратор

- **Компараторы** (устройства сравнения) определяют отношения между двумя словами.
- Основными отношениями, через которые можно выразить остальные, можно считать два — "равно" и "больше".
- В сериях цифровых элементов обычно имеются компараторы с тремя выходами: "равно", "больше" и "меньше".
- Для краткости записей в индексе выходных функций указывается только слово A.

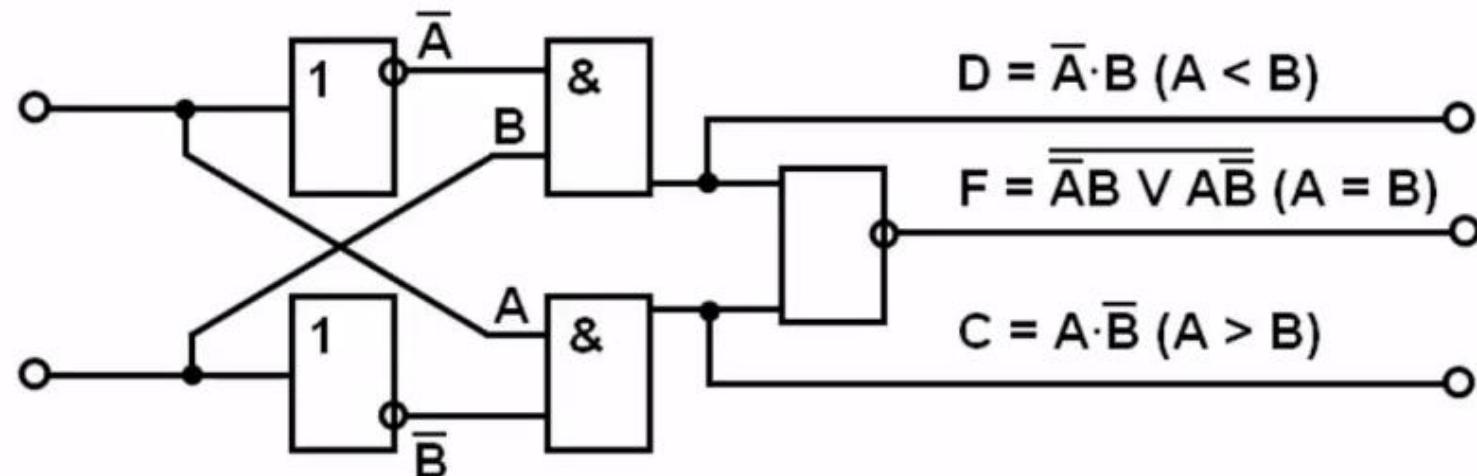
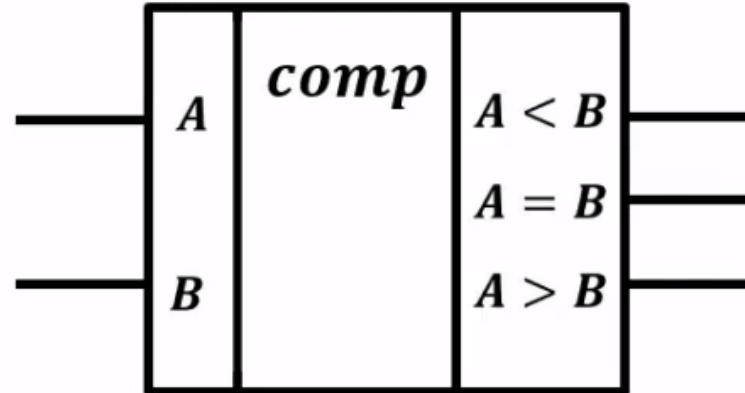


Условное обозначение компаратора с тремя выходами

Компаратор

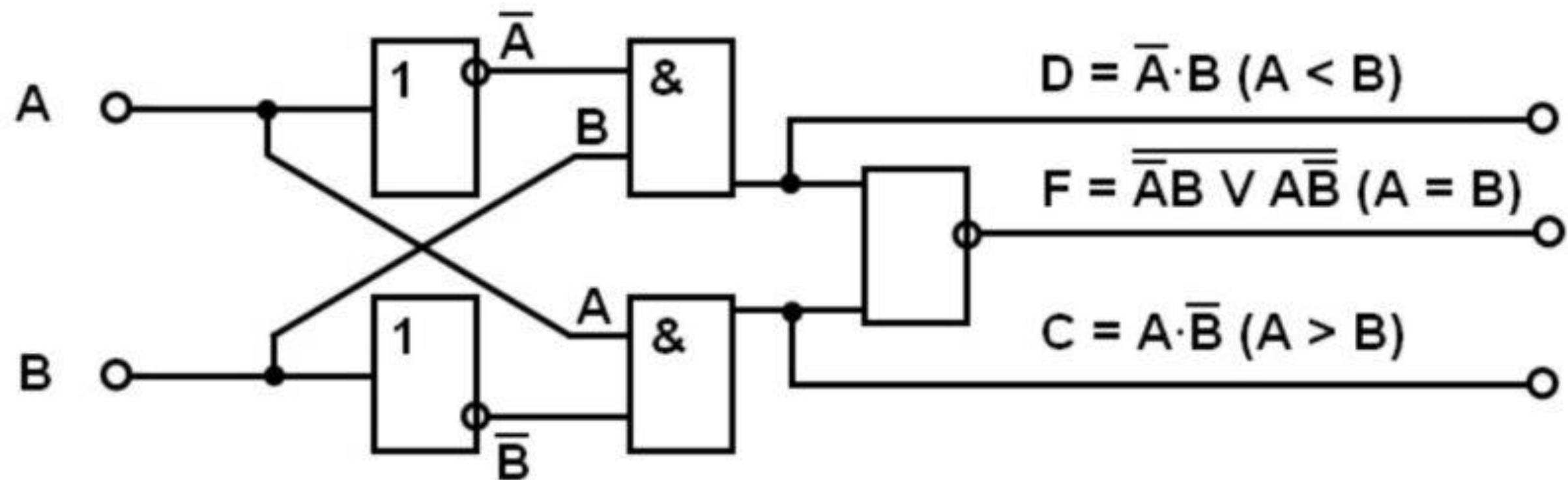
- Цифровой компаратор или компаратор кодов** - логическое устройство с двумя словарными входами, на которые подаются два разных двоичных слова равной в битах длины и обычно с тремя двоичными выходами, на которые выдаётся признак сравнения входных слов, — первое слово больше второго, меньше или слова равны.

A	B	A < B	A = B	A > B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0



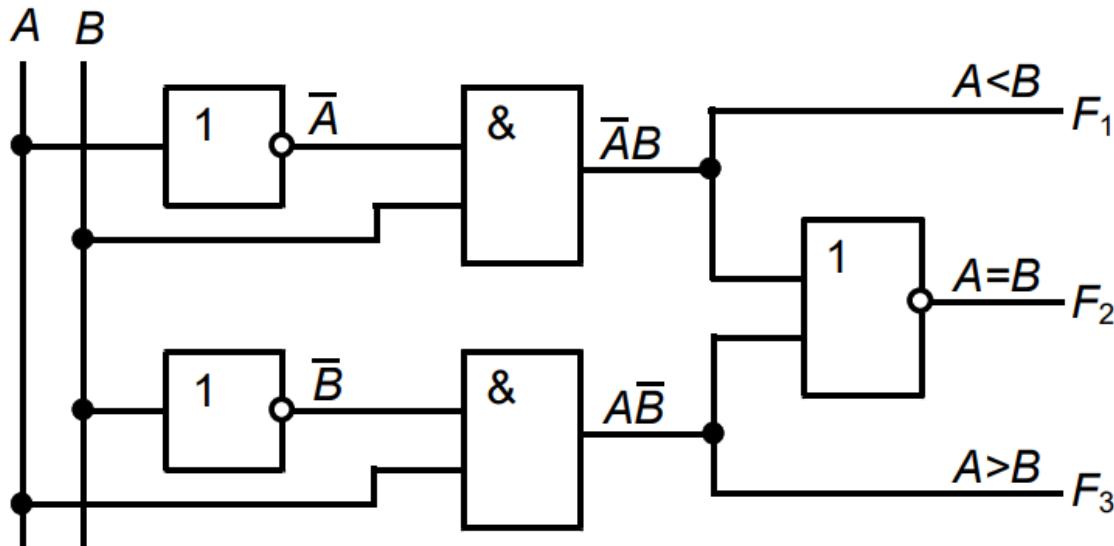
Простейшие компараторы формируют на выходе однобитовый сигнал равенства – “1” или неравенства – “0” двух чисел. Более сложные компараторы в случае неравенства определяют, которое из чисел больше.

Компаратор



Цифровые компараторы

Схема сравнения двух одноразрядных чисел.

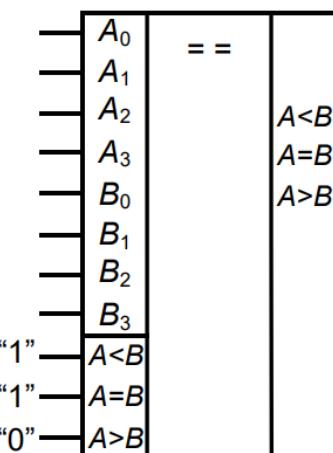


A	B	F_1 ($A < B$)	F_2 ($A = B$)	F_3 ($A > B$)
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

$$F_1 = \bar{A}B; \quad F_3 = A\bar{B};$$

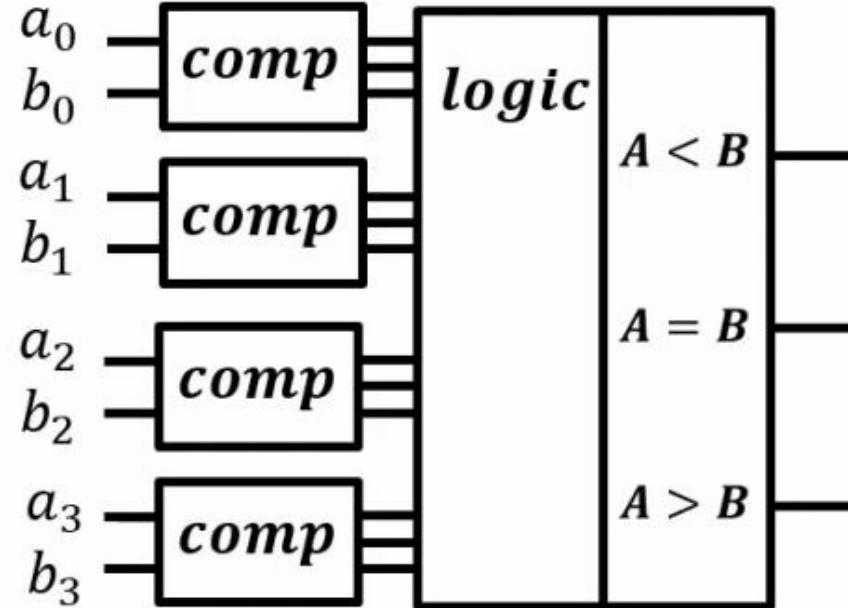
$$F_2 = \overline{\bar{A}B + A\bar{B}} - \text{исключающее ИЛИ-НЕ}$$

K564ИП2



На базе простейшей схемы строятся n-разрядные компараторы. При сравнении многоразрядных чисел А3А2А1А0 и В3В2В1В0 сравнение начинается со старших разрядов. Входы A<B, A=B, A>B – входы расширения - для наращивания разрядности компаратора. Наращивание разрядности компараторов осуществляется последовательно (каскадно) или параллельно (пирамидально). Если используется одна ИМС, то на расширяющие входы подаются 1,1,0. Примеры цифровых компараторов: K564ИП2, K561ИП2, 555СП1.

Наращивание разрядности компараторов



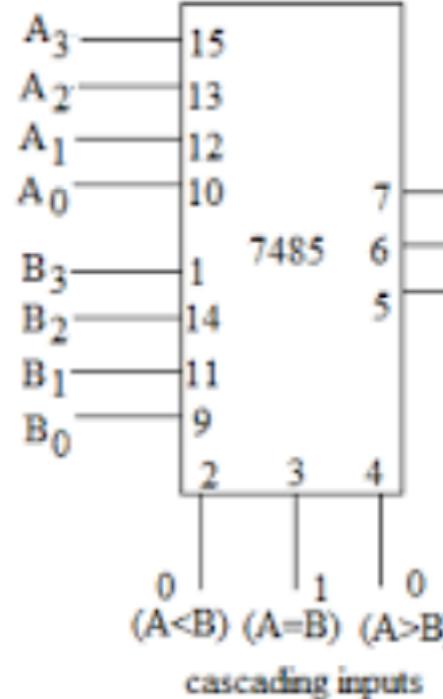
A **0011**
 $\| \quad \|$
 B **0001**

1101
|| || || ||
1101
 $A = B$

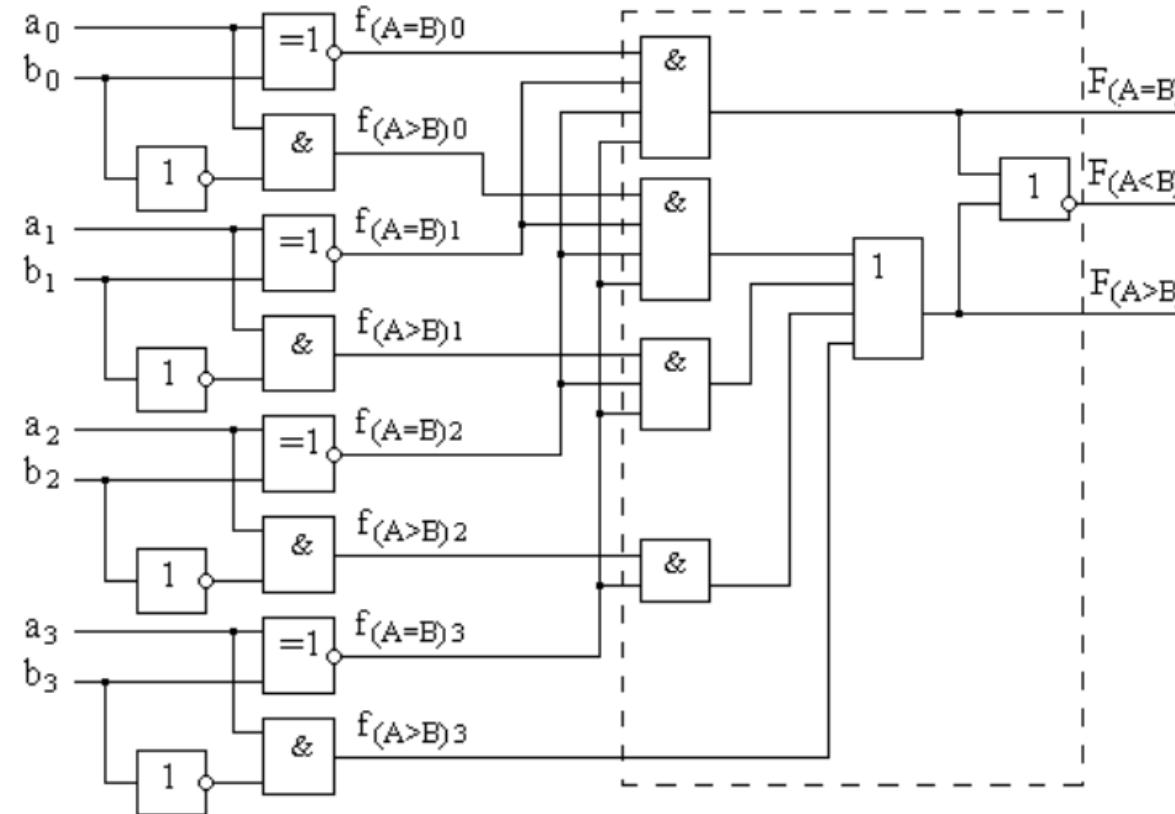
$A > B ?$

A1	A0	B1	B0	A < B	A = B	A > B
0	0	0	0	0	1	0
0	0	0	1	1	0	0
0	0	1	0	1	0	0
0	0	1	1	1	0	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	0	0
1	0	0	0	0	0	1
1	0	0	1	0	0	1
1	0	1	0	0	1	0
1	0	1	1	1	0	0
1	1	0	0	0	0	1
1	1	0	1	0	0	1
1	1	1	0	0	0	1
1	1	1	1	0	1	0

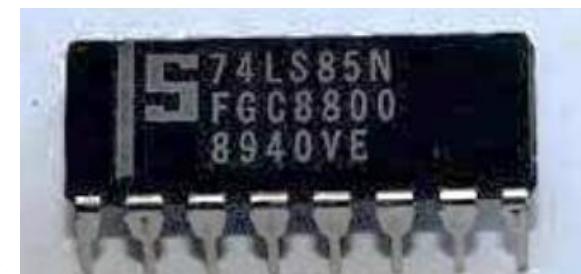
Компаратор



Если все биты в двух словах одинаковы, то на "A=B"- выходе появляется 1. При каскадировании выходы A>B, A=B и A<B схемы, сравнивающей младшие разряды, следует присоединить к одноименным входам последующего каскада. Этим способом с помощью двух таких компараторов можно сравнивать два восьмиразрядных слова.



На схеме представлен 4-битный компаратор. Для сравнения двух 4-разрядных слов требуется четыре схемы равнозначности. На их выходах появится 1 в том случае, когда на все входы поступают пары одинаковых сигналов



Применение цифровых компараторов

- Цифровые компараторы находят широкое применение в различных областях электроники и вычислительных систем, в том числе:

1. Цифровая логика и микропроцессорные системы: Цифровые компараторы часто используются для выполнения операций сравнения между числами в вычислительных устройствах, таких как микроконтроллеры и процессоры. Они могут быть частью арифметико-логических устройств (АЛУ) и использоваться для сравнения данных в алгоритмах обработки.

2. Системы управления: В системах автоматического управления компараторы могут использоваться для принятия решений на основе заданных пороговых значений. Например, цифровой компаратор может контролировать, превышает ли входной сигнал заданное цифровое значение.

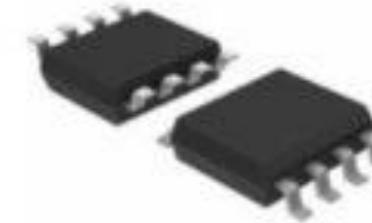
3. Преобразование аналоговых сигналов в цифровые: Цифровые компараторы могут быть частью системы аналого-цифрового преобразования (АЦП). Они используются для сравнения аналоговых сигналов с цифровыми уровнями и преобразования этих аналоговых данных в цифровую форму.

4. Цифровые фильтры и детекторы: В различных детекторных устройствах цифровые компараторы могут применяться для фильтрации сигнала, анализа его характеристик и принятия решений о его значимости.

Компараторы



Analog Devices
AD8561ARZ-REEL7,
Быстродействующий (7нс)
компаратор напряжения с...



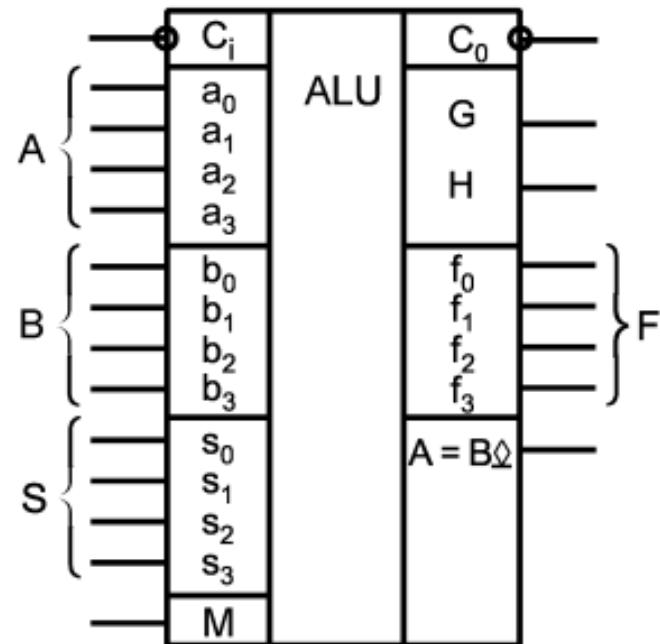
Analog Devices
AD8611ARZ-REEL7,
Быстродействующий (4нс)
компаратор напряжения с...



Analog Devices
AD8612ARUZ, Компаратор
двойной, ультрабыстрый, 4нс
[TSSOP-14]



UMW
LM139DR, Счетверенный
дифференциальный
компаратор [SOIC-14]

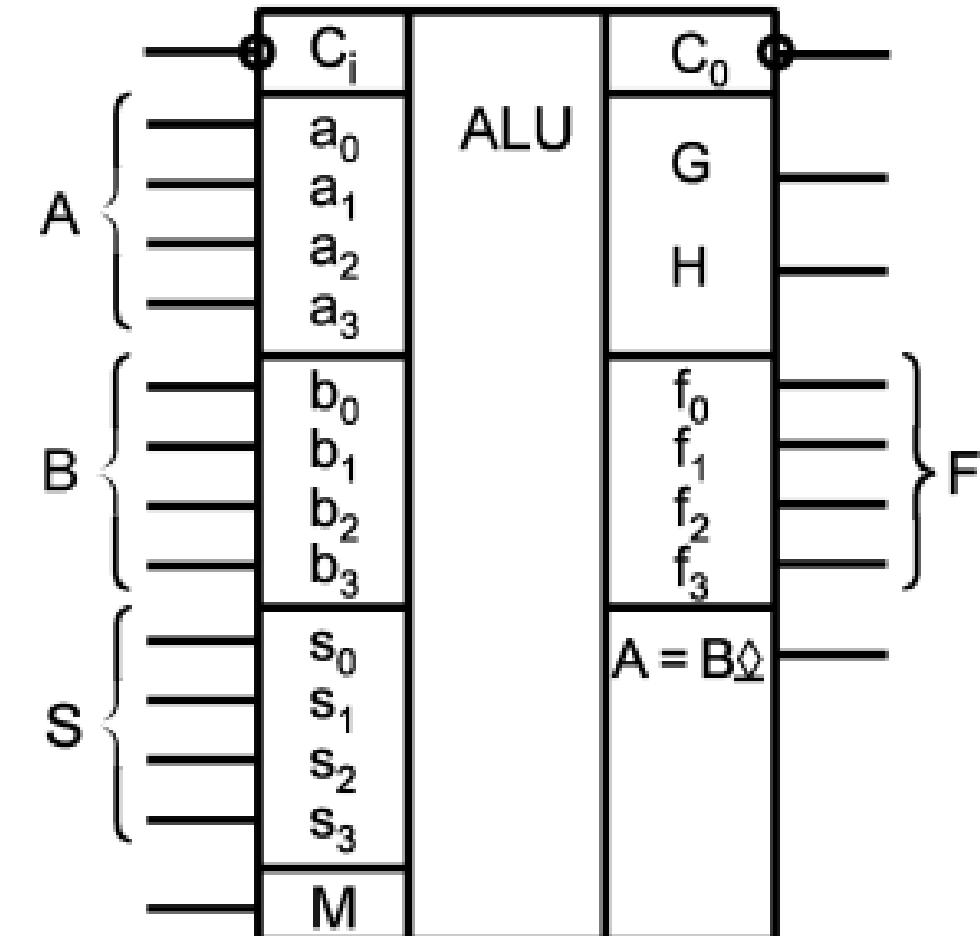


Базовые комбинационные блоки Арифметико- логическое устройство (Алу)



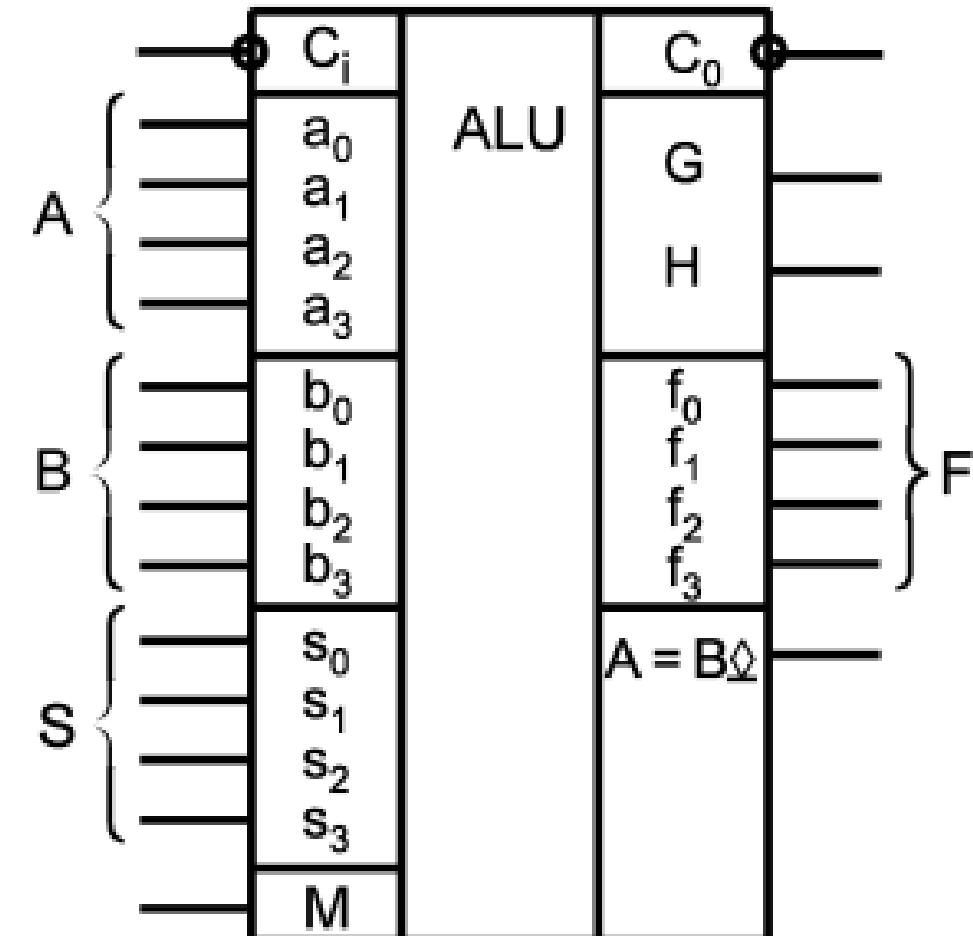
Арифметико-логические устройства

- Арифметико-логические устройства, или сокращенно АЛУ (ALU, Arithmetic-Logic Unit), выполняют над словами ряд действий.
- Основой АЛУ служит сумматор, схема которого дополнена логикой, расширяющей функциональные возможности АЛУ и обеспечивающей его перестройку с одной операции на другую.
- Для наращивания разрядности АЛУ объединяются в вариантах с последовательными или параллельными переносами.
- Логические возможности АЛУ разных технологий (ТТЛШ, КМОП) идентичны.
- В силу самодвойственноеTM выполняемых операций условное обозначение и таблица истинности АЛУ встречаются в двух вариантах, отличающихся взаимно инверсными значениями переменных.



Арифметико-логические устройства

- АЛУ имеет входы операндов A и B, входы выбора операций S, вход переноса Ci и вход M (Mode), сигнал которого задает тип выполняемых операций: логические ($M = 1$) или арифметико-логические ($M = 0$).
- Результат операции вырабатывается на выходах F, выходы G и H дают функции генерации и прозрачности, используемые для организаций параллельных переносов при наращивании размерности АЛУ.
- Сигнал C_0 — выходной перенос, а выход A = B есть выход сравнения на равенство с открытым коллектором.



Условное обозначение АЛУ

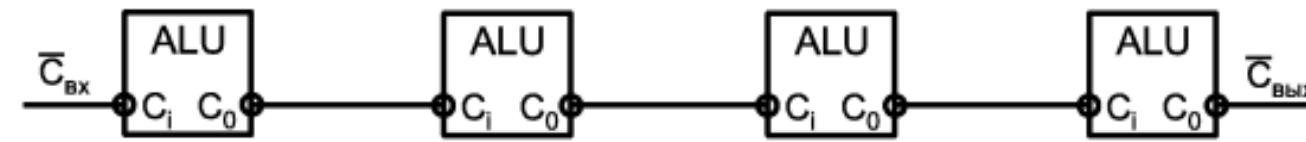
Перечень выполняемых АЛУ операций

S	Логические функции (M = 1)	Арифметико-логические функции (M = 0)
0	\bar{A}	$A + C_i$
1	$\overline{A \vee B}$	$A \vee B + C_i$
2	\overline{AB}	$A \vee \overline{B} + C_i$
3	0	$1 + C_i$
4	\overline{AB}	$A + A\bar{B} + C_i$
5	\bar{B}	$A \vee B + A\bar{B} + C_i$
6	$A \oplus B$	$A + \bar{B} + C_i$
7	\overline{AB}	$A\bar{B} + 1 + C_i$
8	$\overline{A} \vee B$	$A + AB + C_i$
9	$\overline{A \oplus B}$	$A + B + C_i$
10	B	$A \vee \bar{B} + AB + C_i$
11	AB	$AB + 1 + C_i$
12	1	$A + A + C_i$
13	$A \vee \bar{B}$	$A \vee B + A + C_i$
14	$A \vee B$	$A \vee \bar{B} + A + C_i$
15	A	$A + 1 + C_i$

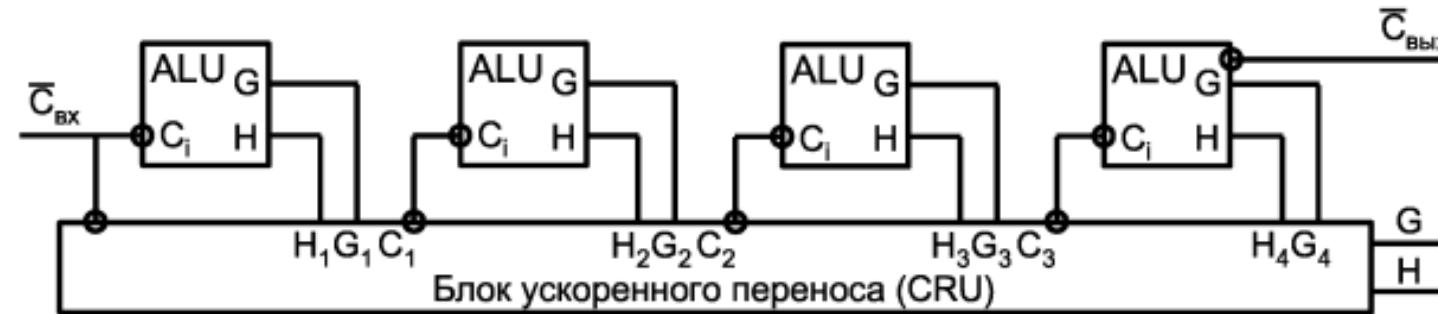
Шестнадцать логических операций позволяют воспроизводить все функции двух переменных.

В логико-арифметических операциях встречаются и логические и арифметические операции одновременно.

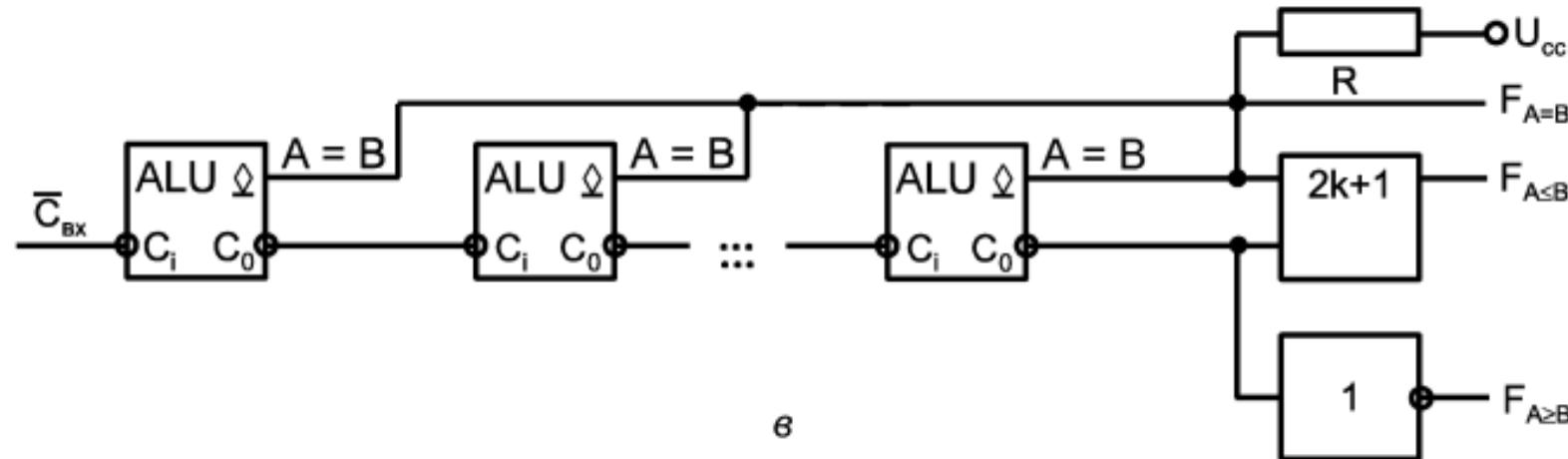
Схемы наращивания АЛУ



а

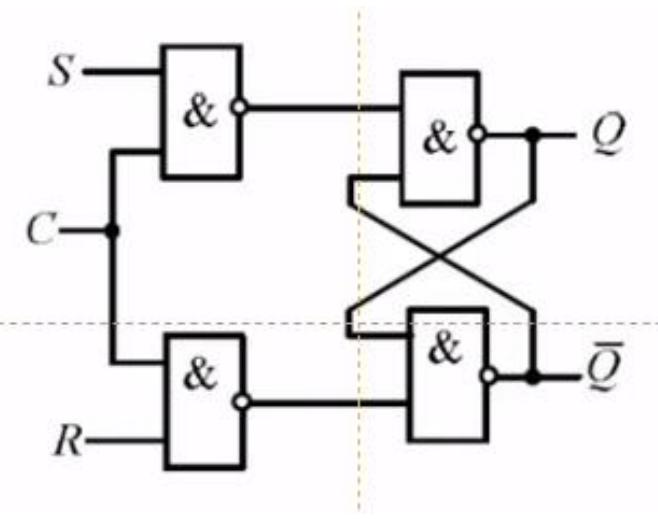


б



в

Схемы наращивания АЛУ при последовательном (а) и параллельном (б) переносах и реализация функций компаратора для группы АЛУ (в)



Последовательностная логика



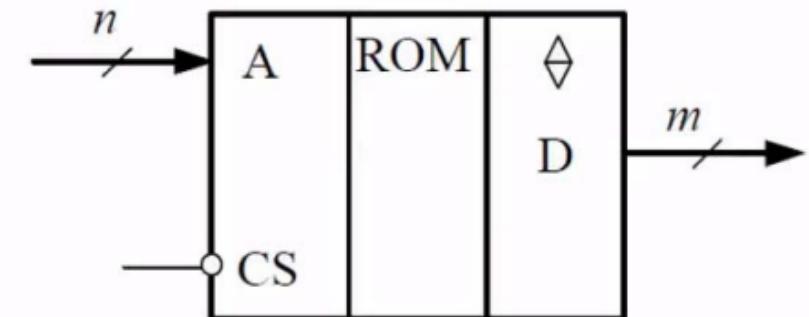
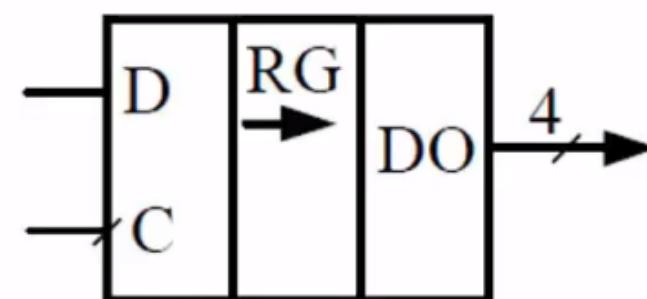
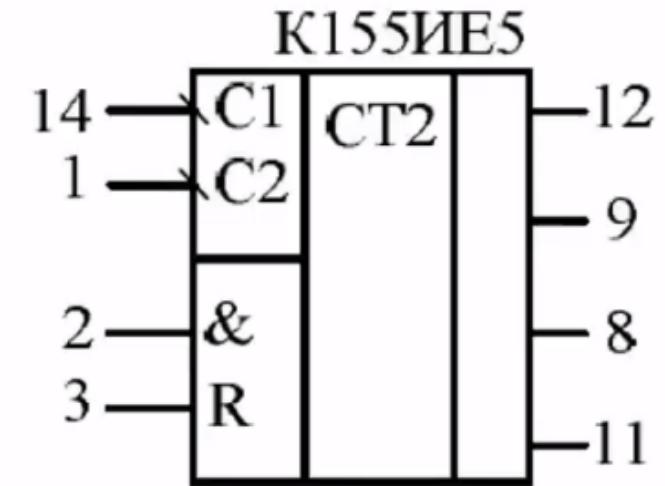
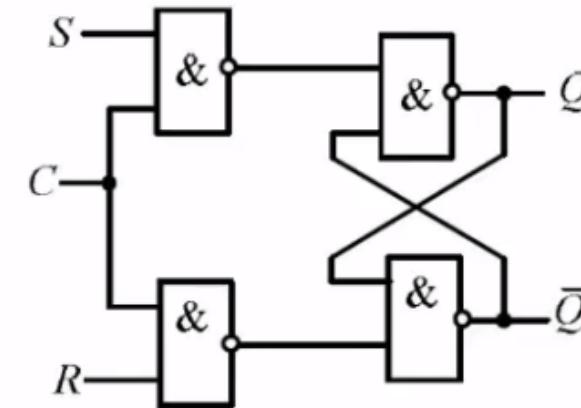
Последовательностные устройства

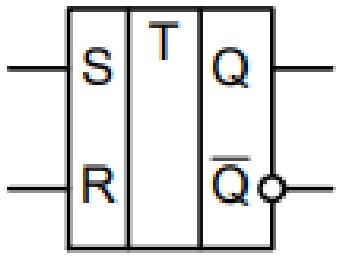
- **Особенностью последовательностных устройств** является зависимость выходного сигнала не только от действующих в настоящий момент на входе логических переменных, но от тех значений переменных, которые действовали на входе в предыдущие моменты времени.
- Для выполнения этого условия значения переменных должны быть запомнены логическим устройством. **Функцию запоминания значений логических переменных в цифровых схемах выполняют так называемые триггерные элементы.**
- **Последовательностные (секвенциальные) цифровые устройства (ПЦУ)** – цифровые устройства с памятью, в которых выходные сигналы зависят не только от текущих значений входных сигналов, но и от последовательности значений входных сигналов, поступивших на входы в предшествующие моменты времени

Последовательная логика

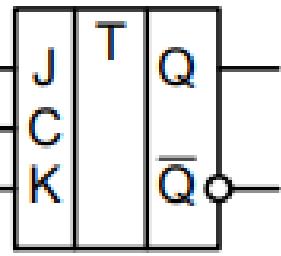
- Типичные Последовательностные (секвенциальные) цифровые устройства (П11У)

- Триггеры
- Регистры
- Счётчики
- Память
- и прочие

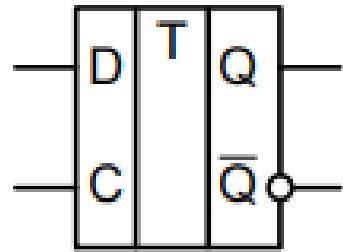




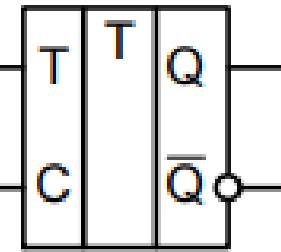
RS



JK



D



T

Последовательная логика Триггеры



Триггер (Trigger, Flip-Flop)

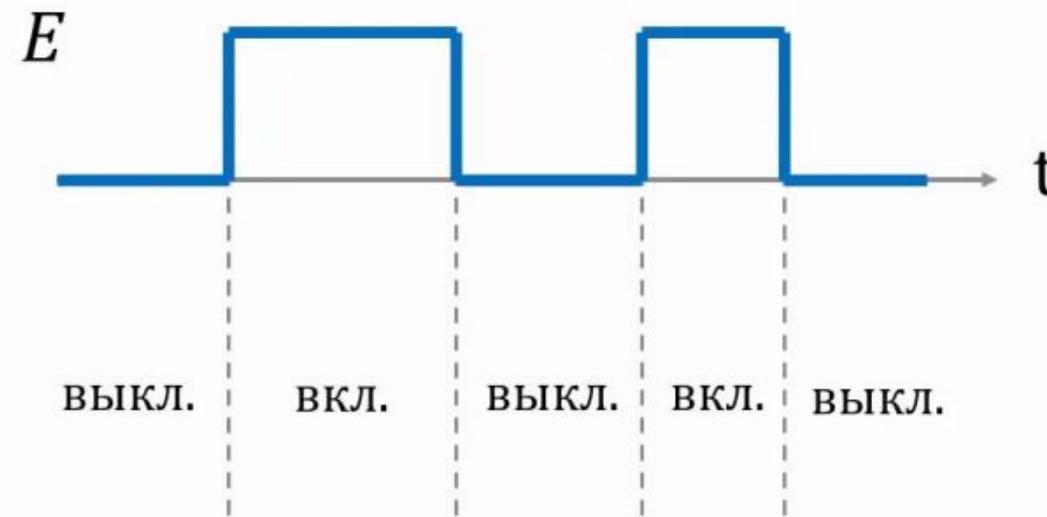
- **Триггером** называется устройство, способное формировать два устойчивых значения выходного сигнала и скачкообразно изменять эти значения под действием внешнего управляющего сигнала.
- **Это свойство позволяет применять триггер в качестве элемента памяти.**
- **Триггер** с английского переводится как «спусковой крючок».
- **Триггер** – это цифровой автомат, имеющий несколько входов и 2 выхода.
- **Триггер** – это устройство последовательностного типа с двумя устойчивыми состояниями равновесия, предназначенное для записи и хранения информации. Под действием входных сигналов триггер может переключаться из одного устойчивого состояния в другое. При этом напряжение на его выходе скачкообразно изменяется.
- **Триггер** – один из базовых (основных) элементов цифровой техники.
- **Триггер** – элементарная ячейка оперативной памяти, в которую можно записать 1 бит информации (0 или 1).
- **Триггер** – это последовательностное устройство с двумя устойчивыми состояниями равновесия, предназначенное для записи и хранения информации.

Триггеры

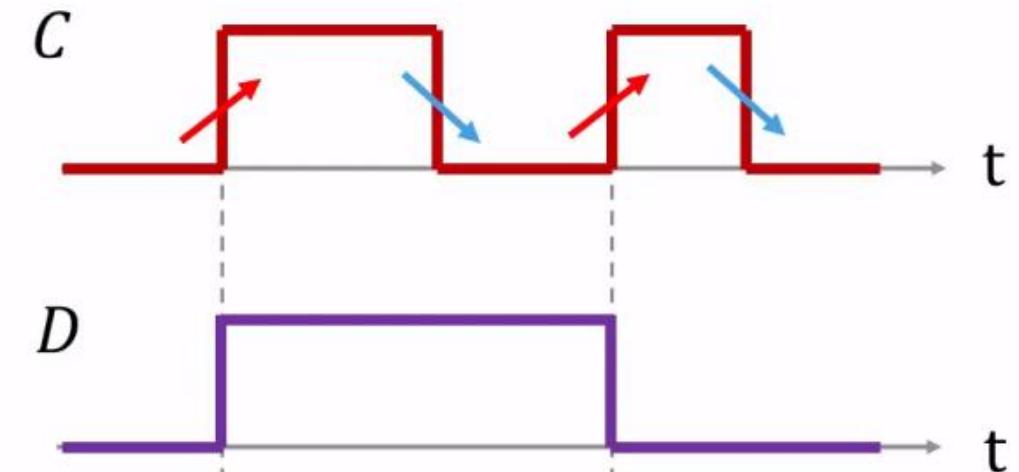
- По способу работы с сигналами различают асинхронные, синхронные и смешанные триггерные схемы, статические и динамические.
- Триггеры можно классифицировать
 - по способу приема информации,
 - по принципу построения,
 - по функциональным возможностям.
- По способу приема информации триггеры подразделяются на асинхронные и синхронные.
 - Асинхронные триггеры воспринимают информационные сигналы и реагируют на них в момент появления на входах триггера.
 - Синхронные (тактируемые) триггеры реагируют на информационные сигналы при наличии разрешающего сигнала на специальном управляющем входе С (от англ. Clock), называемом входом синхронизации (тактовым входом).

Статическое и динамическое управление

Статическое управление –
реакция на **уровень** сигнала



Динамическое управление –
реакция на **перепад** сигнала



* сигнал **D** переключается по
положительному перепаду сигнала **C**

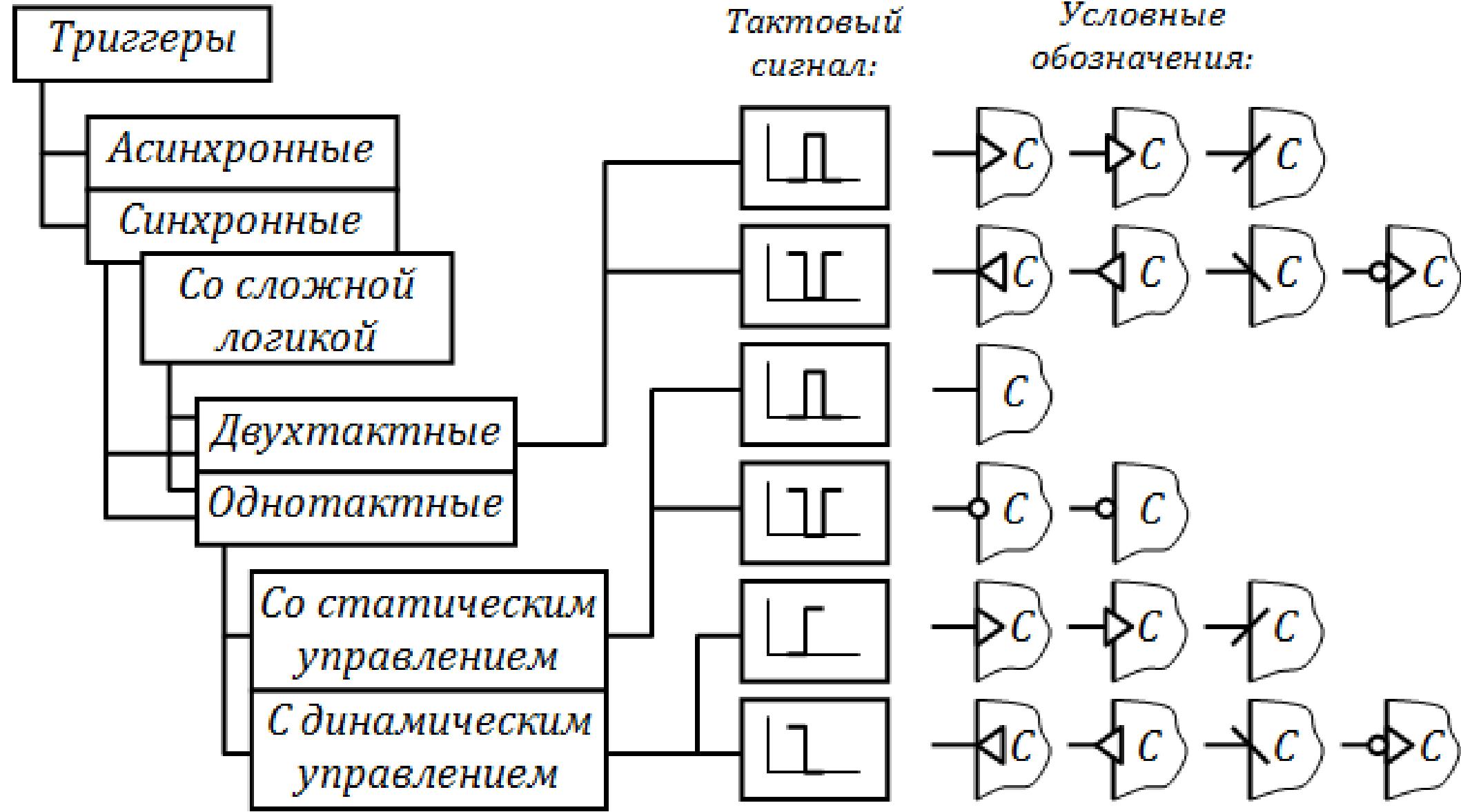
Статические триггеры воспринимают информационные сигналы при подаче на вход тактовый вход С логической единицы или логического нуля.

Динамические триггеры воспринимают информационные сигналы при изменении сигнала на тактовом входе С от 0 к 1 или от 1 к 0.

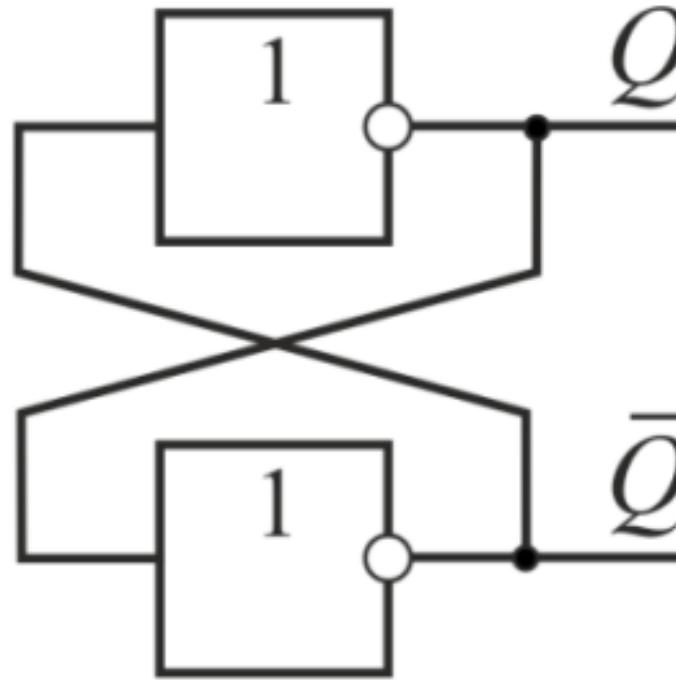
Триггеры

- **По принципу построения** триггеры со статическим управлением можно разделить на одноступенчатые триггеры и двухступенчатые триггеры.
 - **Одноступенчатые** (однотактные) триггеры характеризуются наличием одной ступени запоминания информации.
 - В **двухступенчатых** (двухтактные) триггерах имеются две ступени запоминания информации: в начале информация записывается в первую ступень, а затем переписывается во вторую и появляется на выходе. Двухступенчатый триггер обозначают ТТ.
- **По способу реакции на помехи различают прозрачные и непрозрачные.**
- **Триггеры используются** в основном в вычислительной технике для организации компонентов вычислительных систем: процессоров, регистров, счётчиков, ОЗУ

Триггеры



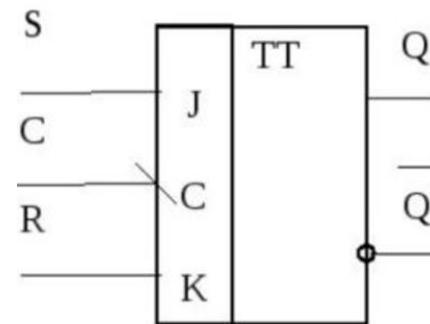
Триггеры



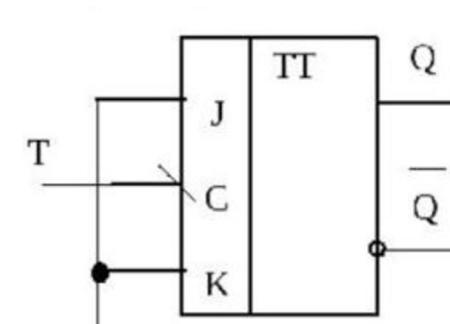
- В основе триггера лежит бистабильная ячейка, которая образуется объединением перекрёстными инвертирующих логических элементов.
- Такая ячейка имеет только два устойчивых состояния «0» и «1».
- Выход Q называют **прямым** выходом, выход \bar{Q} называют **инверсным** выходом.
- Говорят, что триггер **установлен** в единичное состояние, когда на его прямом выходе появляется напряжение высокого уровня. В противном случае триггер считается **сброшенным**.
- **1 триггер хранит 1 бит информации.**

Триггеры

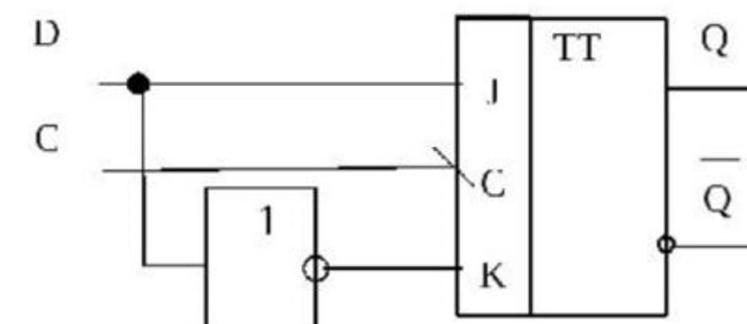
- По функциональным возможностям различаются:
 - триггер с раздельной установкой состояний 0 и 1 (**RS-триггер**);
 - триггер с приемом информации по одному входу D (**D-триггер** или триггер задержки);
 - триггер со счетным входом T (**T-триггер**);
 - универсальный триггер с информационными входами J и K (**JK-триггер**).



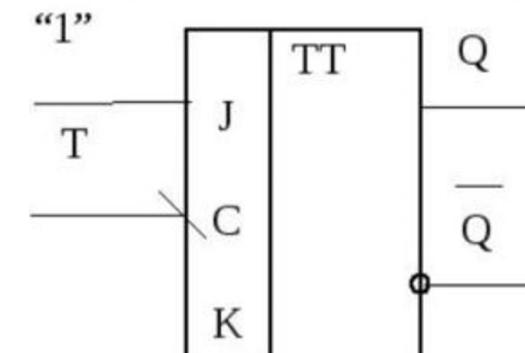
Синхронный
RS-триггер



T-триггер



D-триггер



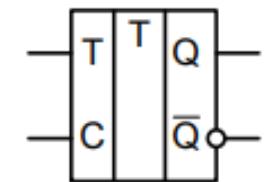
Асинхронный
T-триггер

Обозначения ВХОДов Триггеров разного назначения

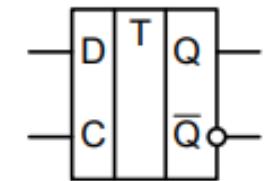
- **S** (Set) или **J** (Jump) – раздельный вход, устанавливающий триггер в единичное состояние на Q-выходе
- **R** (Reset) или **K** (Kill) – раздельный вход, устанавливающий триггер в нулевое состояние на Q-выходе
- **C** (Clock, CLK) – тактовый вход, вход управления/синхронизации.
- **D** (Data) – информационный вход (на него подается информация для занесения в триггер)
- **T** (Toggle) – счетный вход.

Кроме указанных основных входов некоторые триггеры могут иметь вход V. Вход V блокирует работу триггера и он сколь угодно долго может сохранять ранее записанную в него информацию.

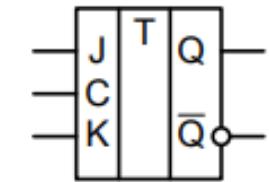
По виду активного логического сигнала, действующего на информационных входах, триггеры подразделяют на статические, управляемые уровнем, и динамические, управляемые перепадом входного сигнала. При этом сами входы могут быть **прямymi и инверсными**.



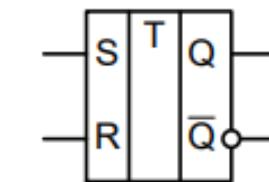
T



D



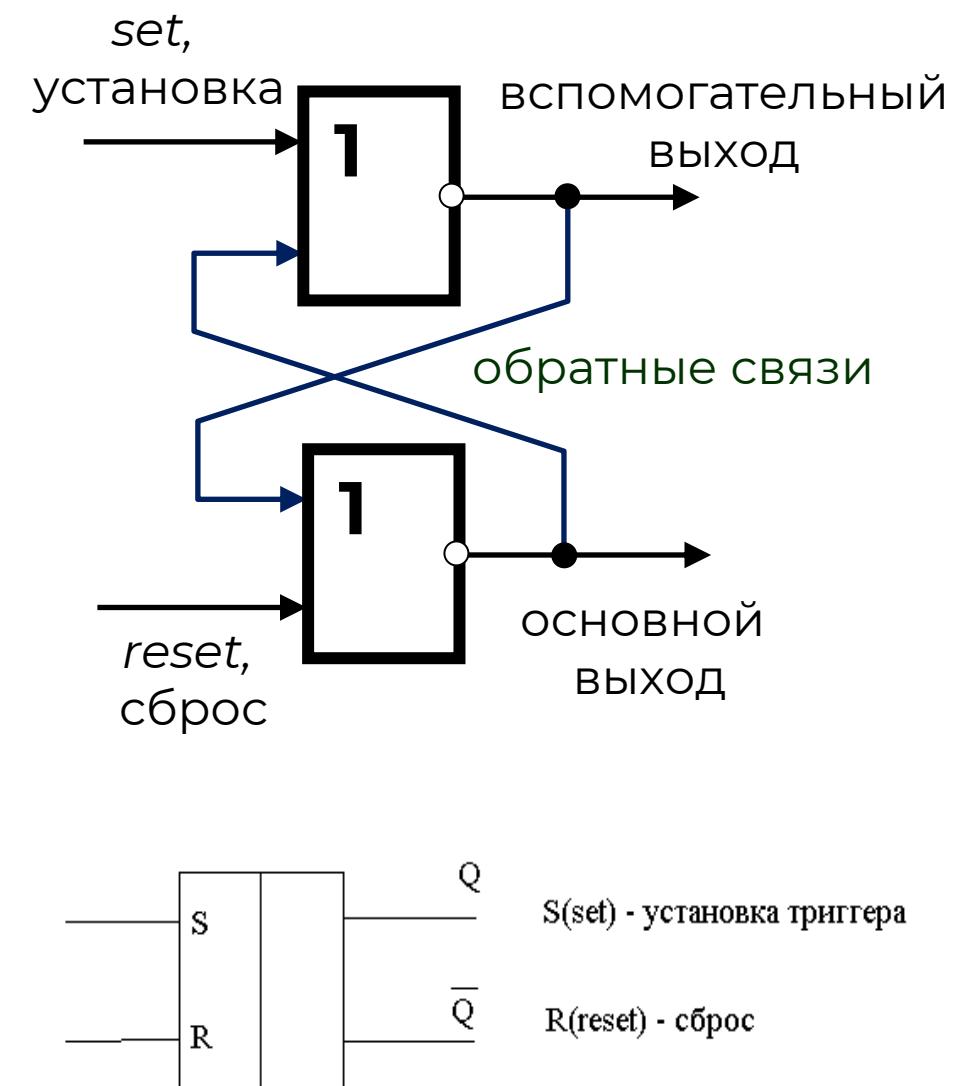
JK



RS

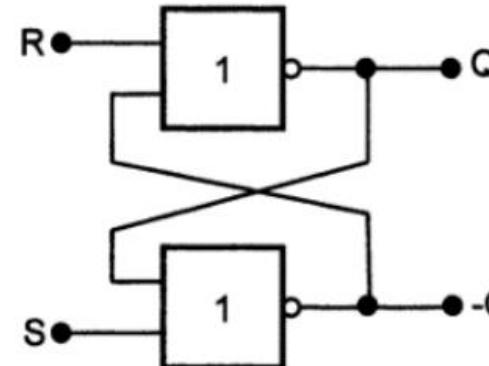
Триггер

- **Триггер** – это логическая **схема, способная хранить 1 бит информации (1 или 0)**.
- Воздействуя на входы триггера, его переводят в одно из двух возможных состояний (0 или 1).
- С поступлением **сигналов** на входы триггера в зависимости от его состояния либо происходит переключение, либо исходное состояние сохраняется.
- **При отсутствии входных сигналов** триггер сохраняет свое состояние сколь угодно долго.
- **Самый распространённый тип триггера – это RS-триггер** (S и R соответственно от английских set - установка, и reset - сброс).

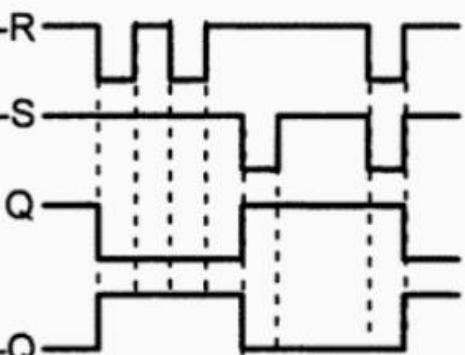
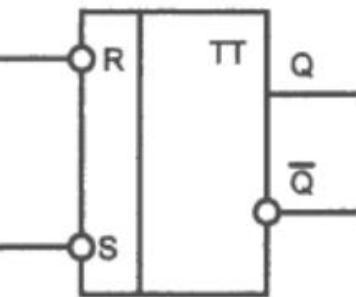
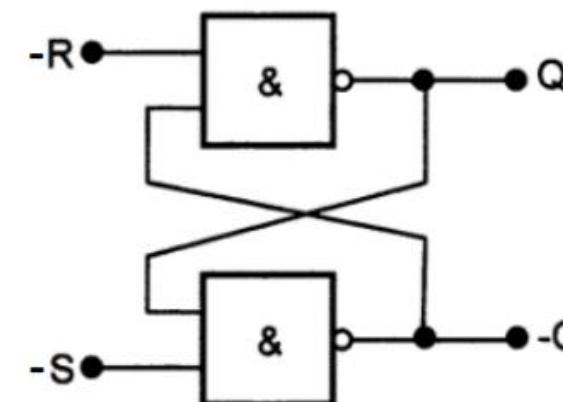
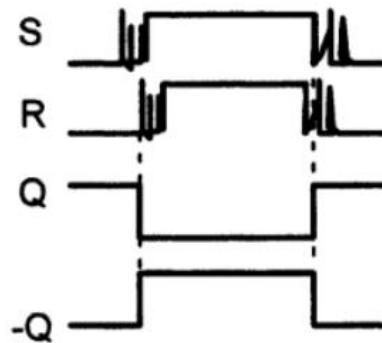


Асинхронный RS-триггер

- **Асинхронный триггер RS-типа** имеет два информационных входа R и S. При $S = 0$ и $R = 1$ на выходах триггера появляются сигналы: на прямом выходе $Q = 0$, на инверсном $\bar{Q} = 1$. При $S = 1$ и $R = 0$ выходные сигналы триггера принимают противоположные состояния ($Q = 1, \bar{Q} = 0$). Этот триггер не имеет тактового входа.
- **Простейший RS-триггер можно реализовать на логических элементах ИЛИ-НЕ или И-НЕ.**

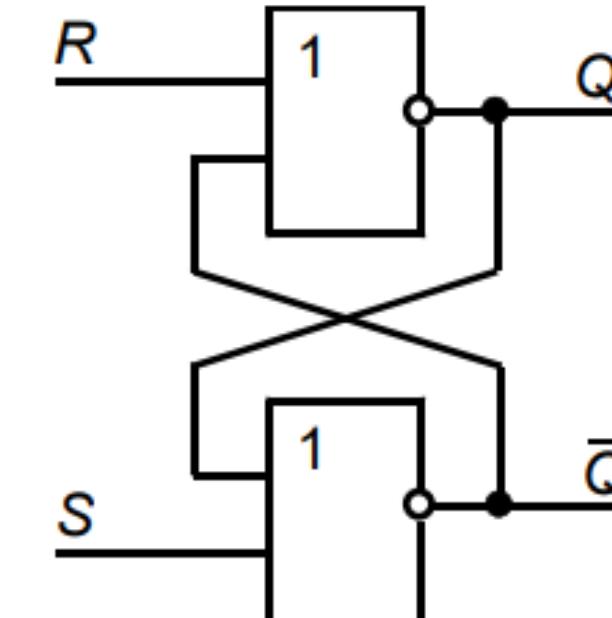


S	R	Q
0	1	0
0	0	0
1	0	1
0	0	1
1	1	X

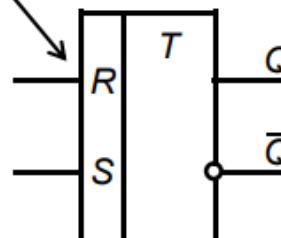


RS-триггеры используют в основном для устранения влияния дребезга контактов при переключении механических коммутаторов. Первый импульс, поступивший на вход R триггера при нажатии на некую кнопку, переключает его выход в состояние логического 0, а первый импульс на входе S переключает выход триггера в состояние 1. Все остальные сигналы, обусловленные дребезгом контактов, не влияют на триггер.

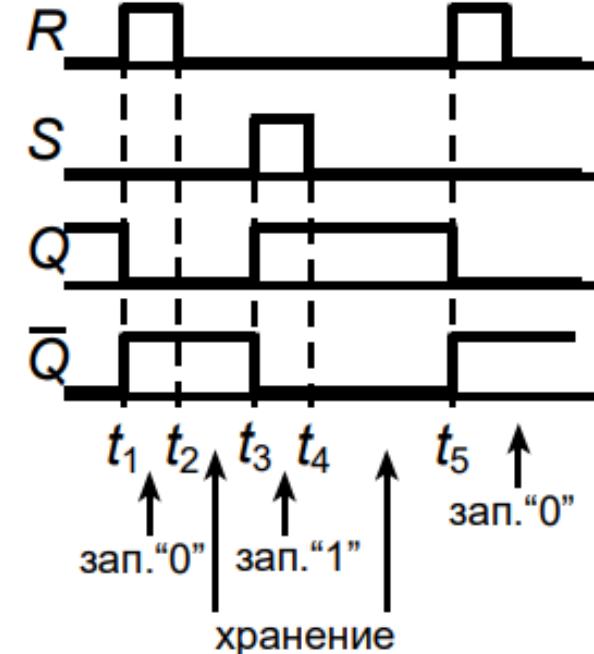
Асинхронный RS-триггер (схема на элементах ИЛИ-НЕ)



Активный
уровень «1»



УГО триггера



В запрещенном режиме:

1. инверсный выход утрачивает свою «инверсность»
2. работа неопасна для микросхемы

Режим называется запрещенным, поскольку очень сложно предсказать поведение микросхемы после снятия уровней лог. «1» с обоих входов. Такое включение микросхемы является нестандартным

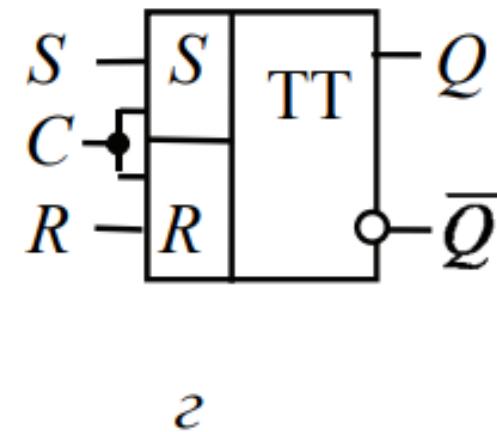
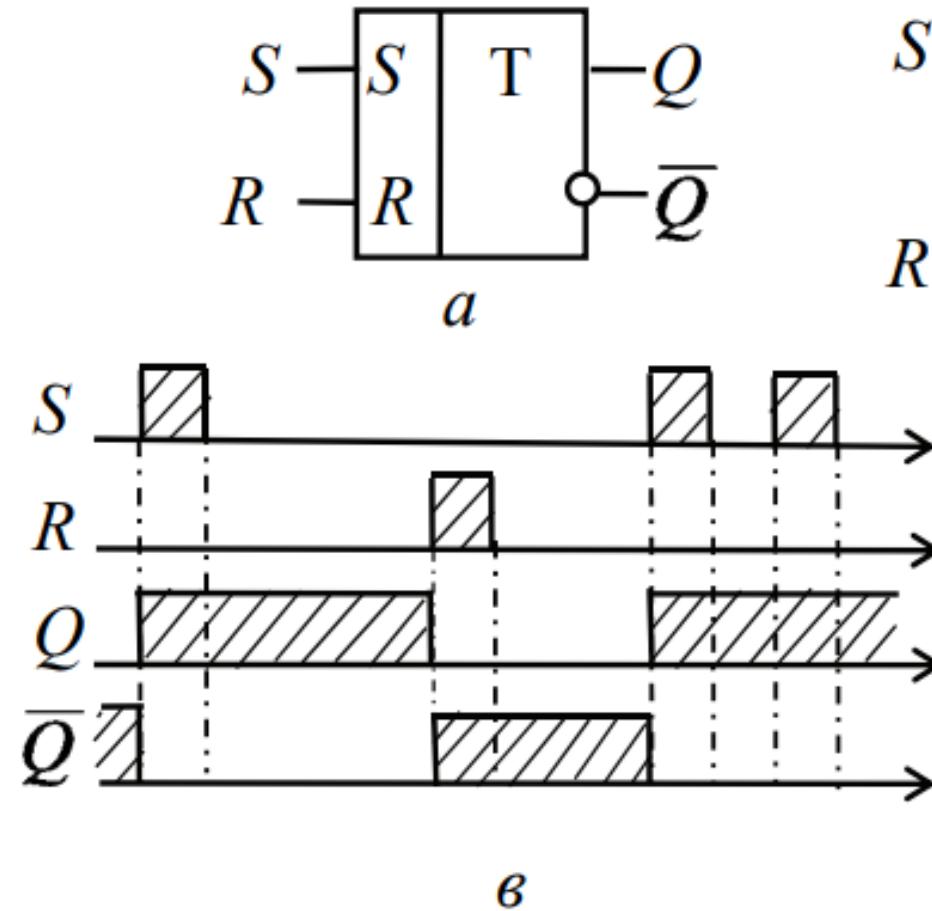
R	S	Q^n	Q^{n+1}	Режим
0	0	0	0	хранения (сост. не меняется)
0	0	1	1	
0	1	0	1	запись «1»
0	1	1	1	
1	0	1	0	запись «0»
1	0	0	0	
1	1	X		запрещенный!

Q^n - предыдущее состояние (состояние триггера до подачи сигналов R,S).

Q^{n+1} - последующее состояние (состояние триггера после подачи сигналов R,S).

X – безразличное состояние.

Асинхронный RS-триггер



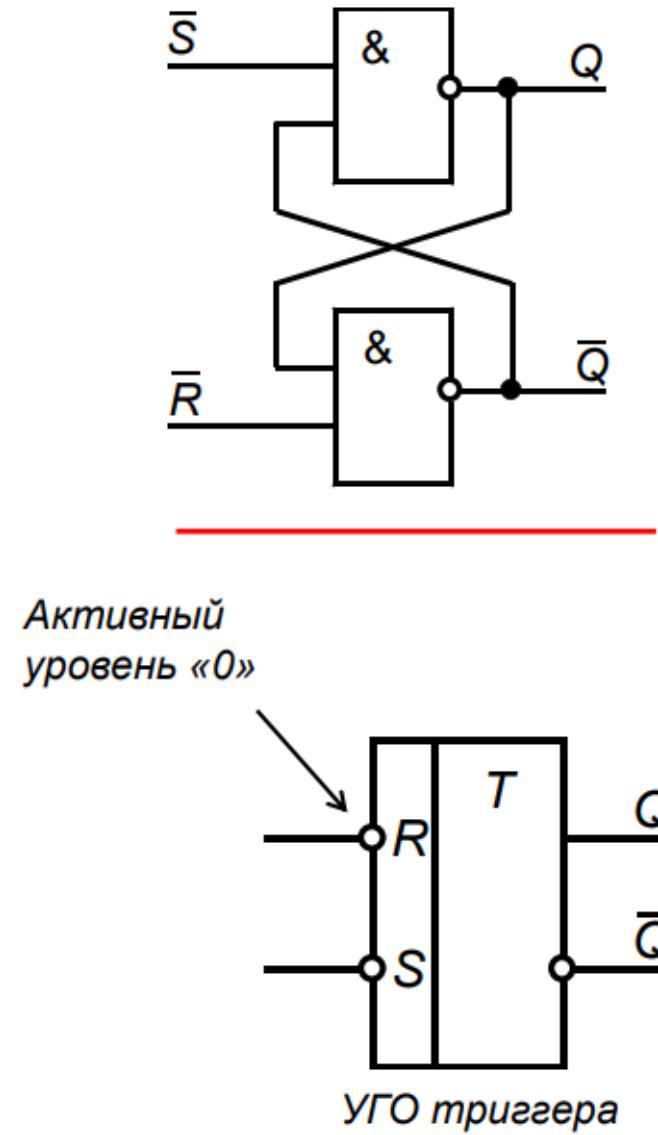
а – условное графическое обозначение; б – схема реализации на базовых элементах И-НЕ;
в – временная диаграмма; г – преобразование в синхронный; д – таблица переходов

Таблица переходов

S	R	Q^{t+1}
0	0	Q^t
0	1	0
1	0	1
1	1	–

д

Асинхронный RS-триггер инверсными входами (схема на элементах И-НЕ)



\bar{R}	\bar{S}	Q^n	Q^{n+1}	Режим
0	0	X		запрещенный!
0	1	X	0	запись «0»
1	0	X	1	запись «1»
1	1	X	Q^n	хранение

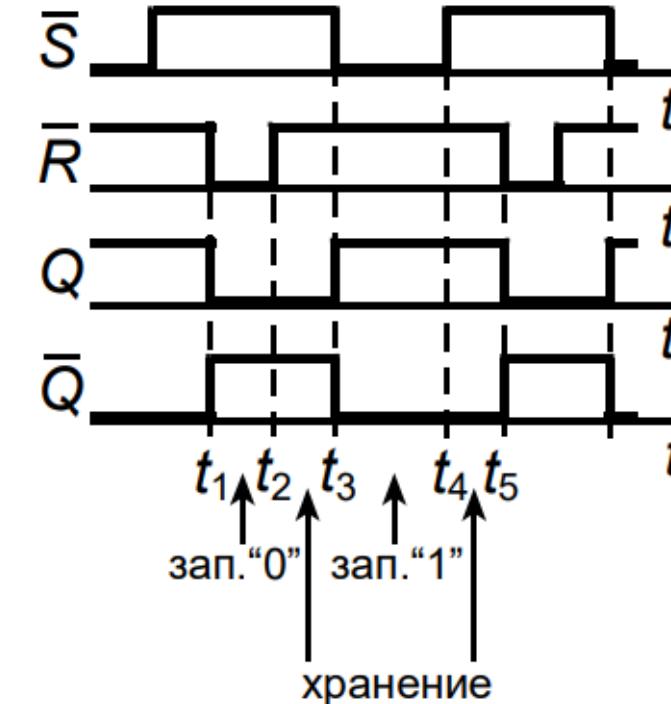


Схема RS-триггера на элементах 2ИЛИ-НЕ

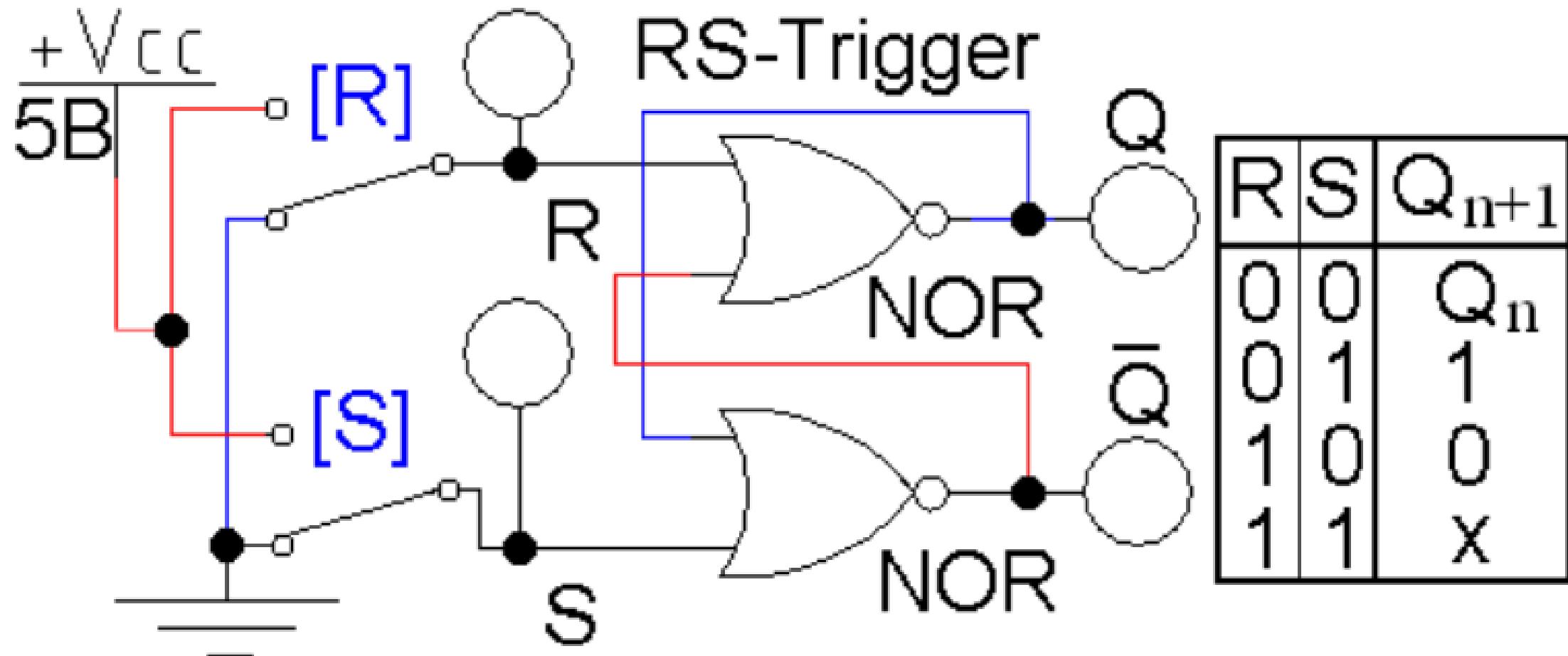
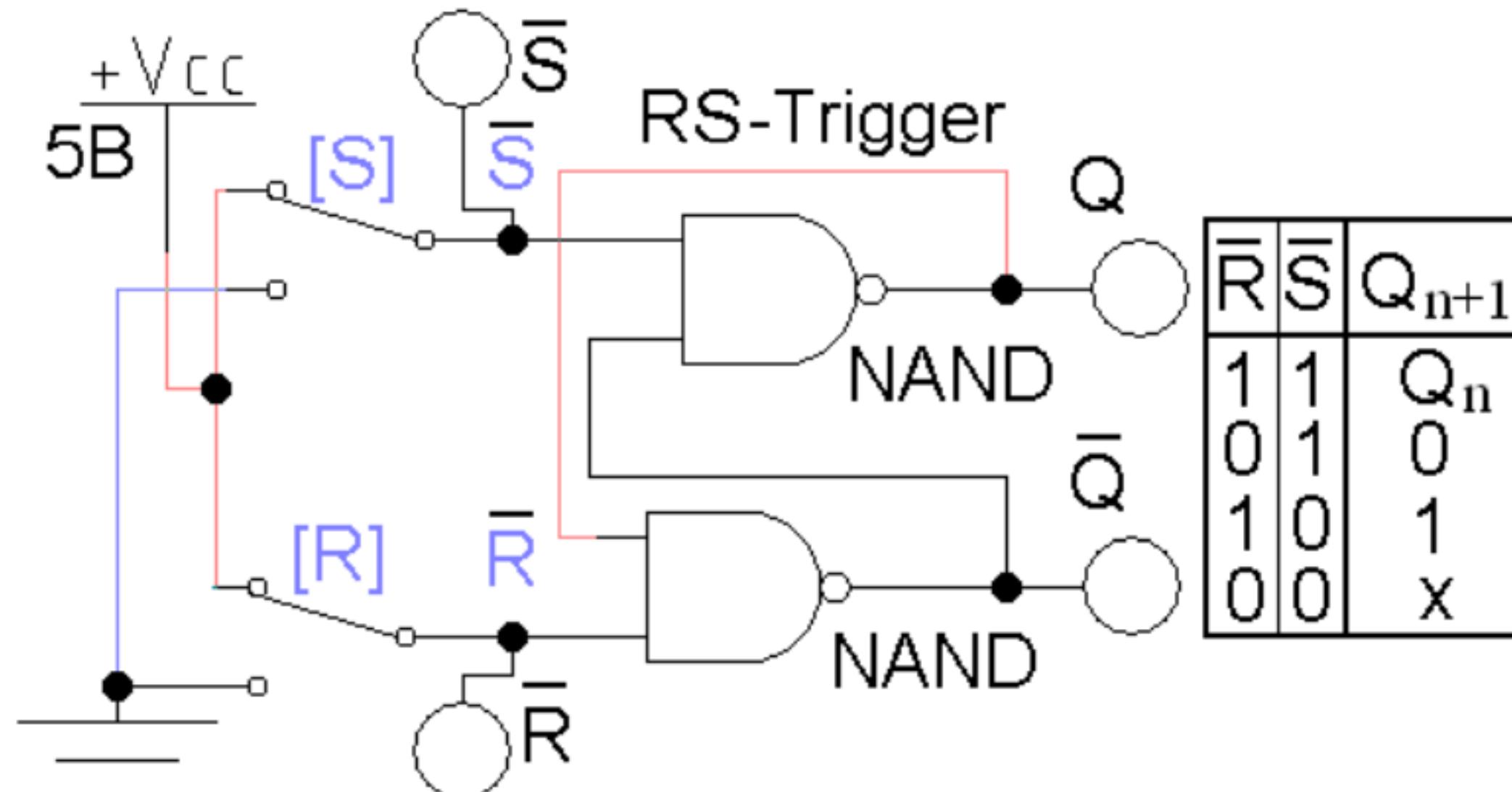
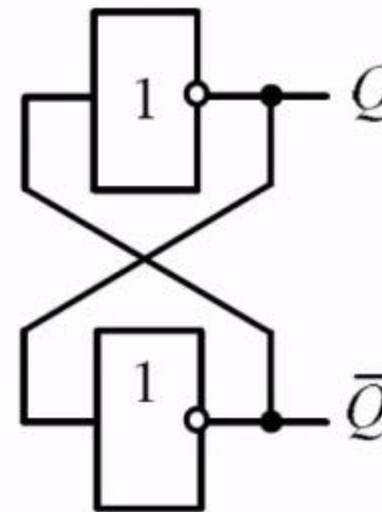


Схема RS-триггера на элементах 2И-НЕ

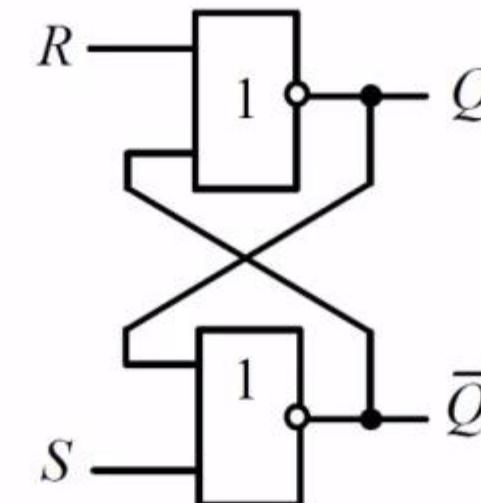


Асинхронный RS-триггер (защелка)

Бистабильная ячейка:

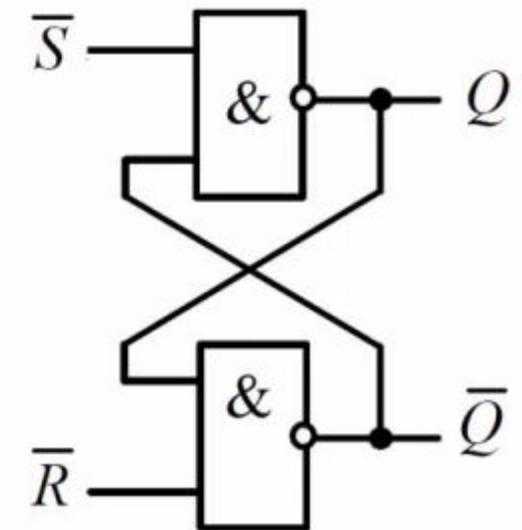


RS-защёлка на
элементах 2ИЛИ-НЕ:



S – **set**, установка
 R – **reset**, сброс

SR-защёлка на
элементах 2И-НЕ:



«активный» уровень – 0
«неактивный» – 1

Защёлка или триггер?

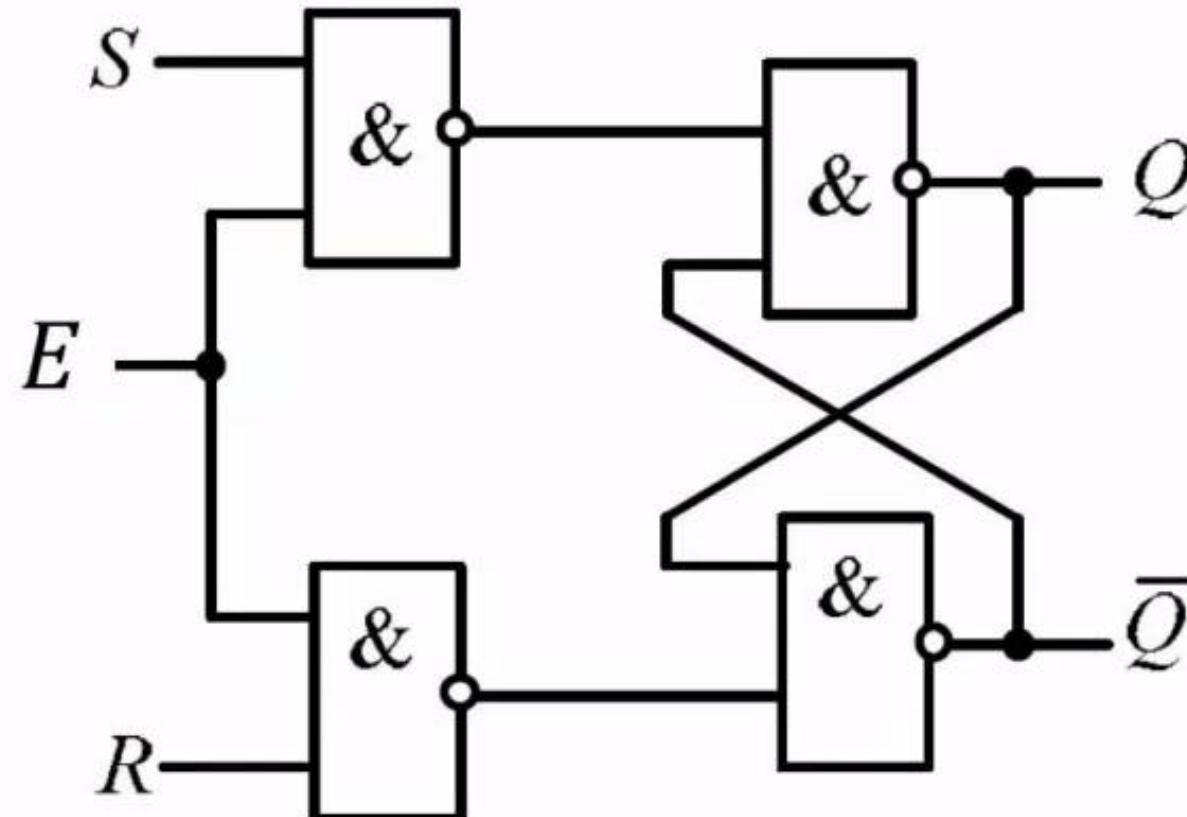
ЗАЩЁЛКА (LATCH)

- Асинхронна
- Управляется уровнем
- «Прозрачна» - изменения сигналов на входах могут приводить к моментальным изменениям выхода
- Может хранить несколько бит информации

ТРИГГЕР (FLIP-FLOP)

- Синхронен
- Управляется фронтом
- Изменения состояния могут происходить только в моменты перепада управляющего сигнала С
- Хранит 1 бит информации

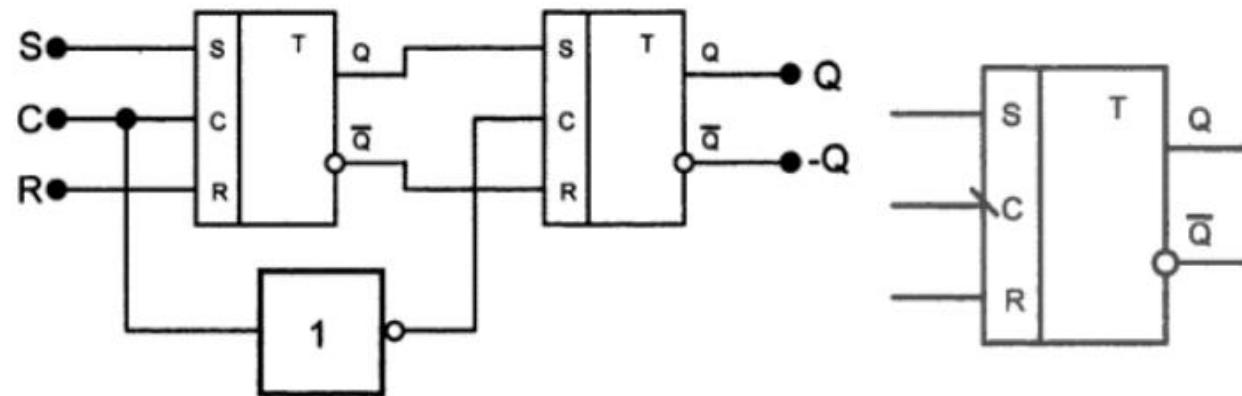
Управляемая RS-защёлка (gated latch)



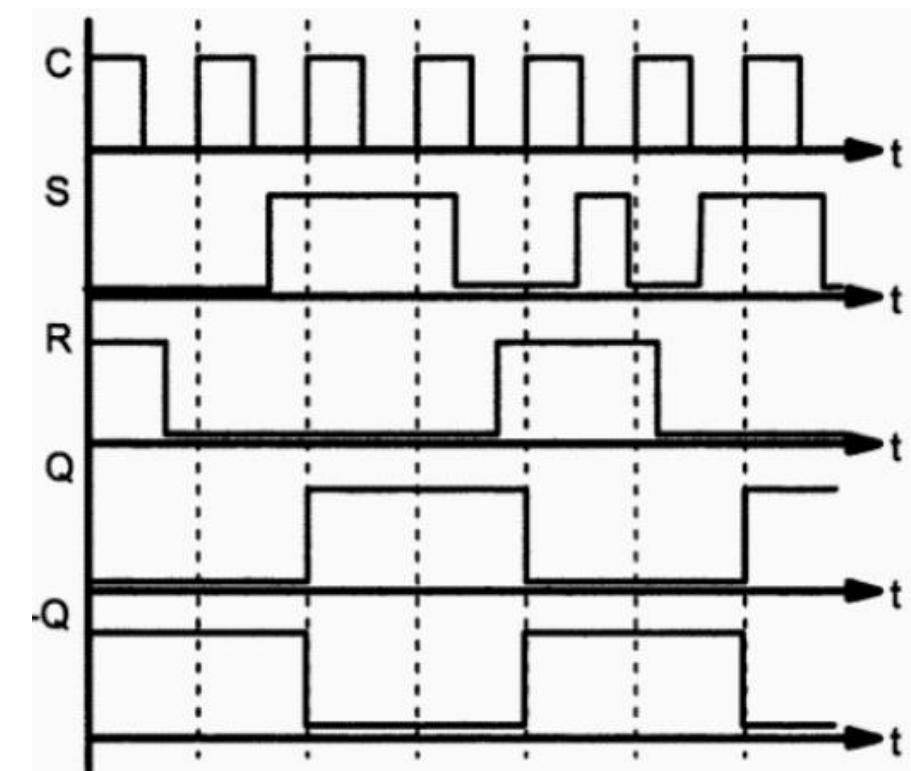
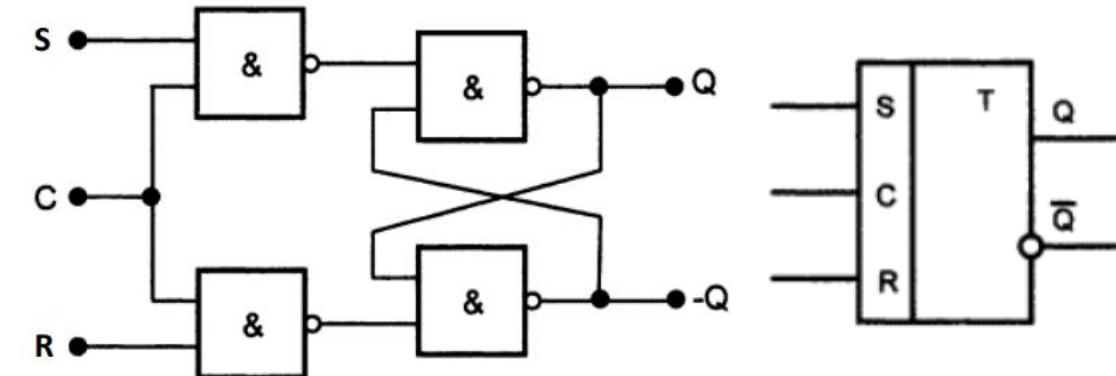
- Вход **E** (англ, enable) - это кнопка **ВКЛ/ВЫКЛ** для защёлки
- На вход **E** можно подать сигнал синхронизации (**тактовый** сигнал), но обычно на него поступают стробы чтения/записи
- Традиционно в литературе называется «синхронным RS-триггером», что не совсем верно по сути

Синхронный RS-триггер

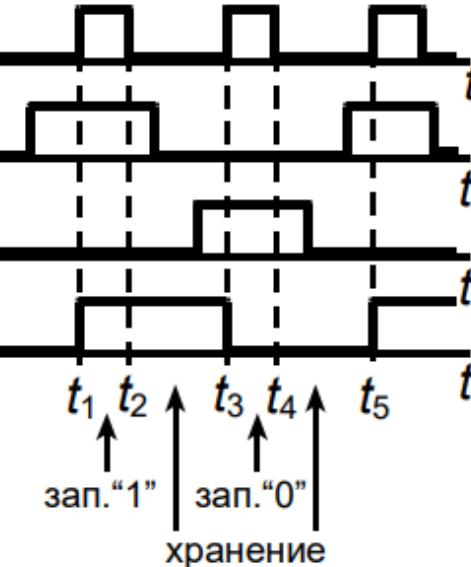
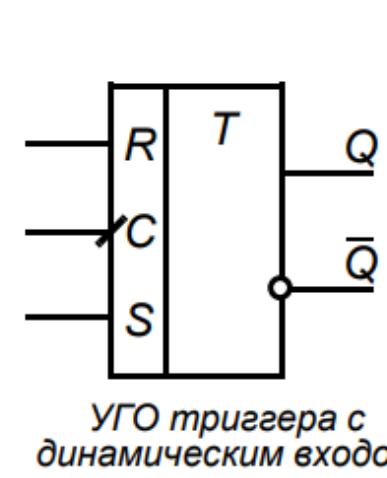
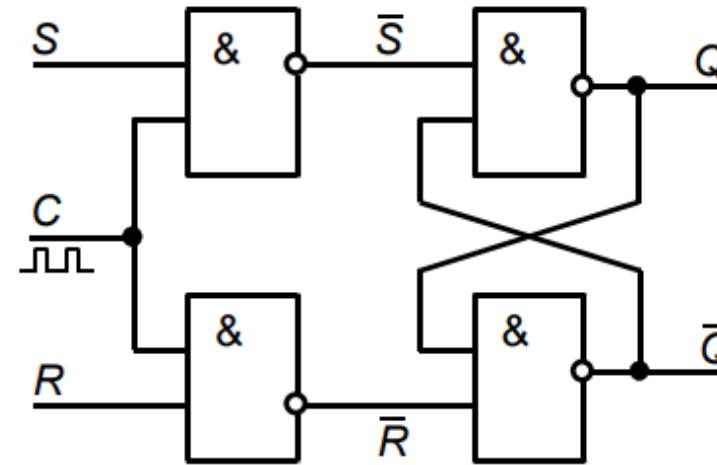
- **Синхронный одноступенчатый RS-триггер** отличается от асинхронного наличием С-входа для синхронизирующих тактовых импульсов.
- Синхронный триггер состоит из асинхронного RS-триггера и двух логических элементов ("схема управления") на его входе.
- Если на входе синхронизации С присутствует логический ноль, RS-триггер находится в режиме "хранение" и не реагирует на входные сигналы.
- При подаче на вход синхронизации С логической единицы триггер переходит в режим записи.



RS-триггер с синхронизацией по заднему фронту импульсов



Синхронный RS-триггер (RST-триггер)



S	R	C	Q^n	Q^{n+1}	
X	X	0	X	Q^n	хранение
1	0	1	X	1	запись «1»
0	1	1	X	0	запись «0»
1	1	1	X	неопределенность (запрещенный режим)	

Переключение триггера производится только при наличии тактового сигнала.

Вход С или CLK - тактовый вход, сокр. от англ. «**CLock**» - тактировать.

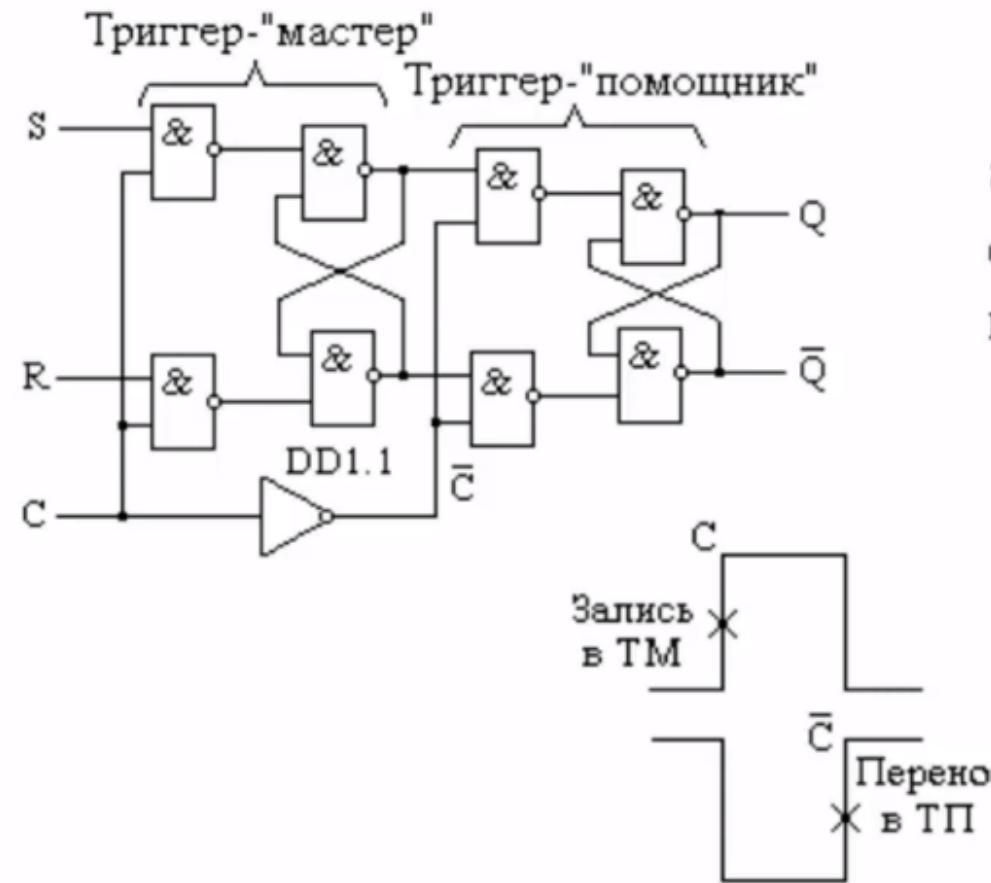
При $C = 1$ триггер становится прозрачным по входам S и R , т.е. любые изменения сигналов на входах S и R отражаются на выходах \bar{S} и \bar{R} , \Rightarrow триггер переключается в соответствии с таблицей истинности. Триггер становится **прозрачным**.

Для нормальной работы триггера во время действия синхроимпульса (сигнала по входу C) сигналы R и S **не должны изменяться!**

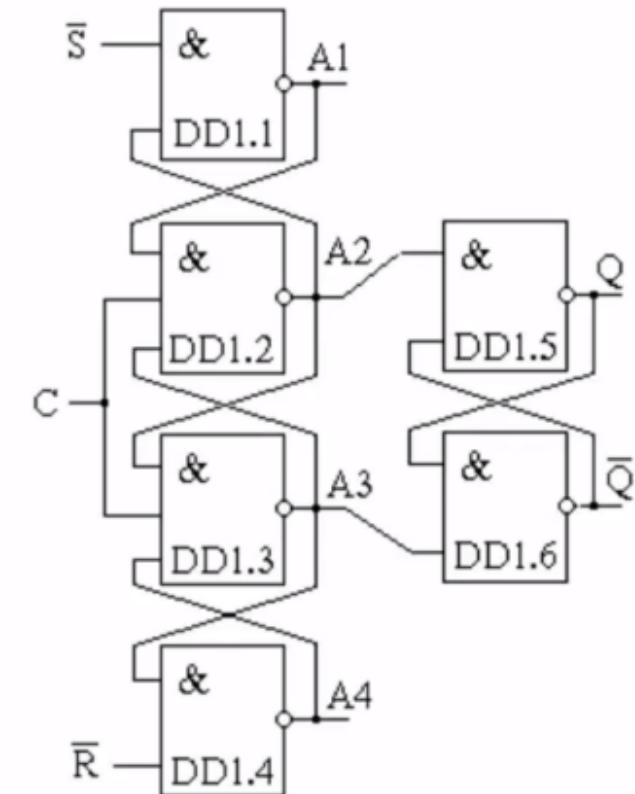
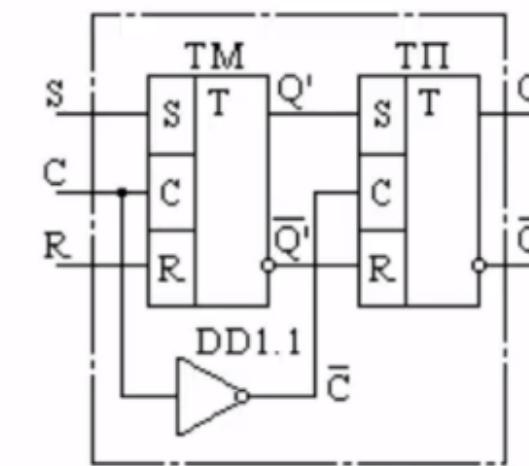
При $C = 0$ первые два элемента работают как инверторы, триггер находится в режиме хранения информации $Q^{n+1} = Q^n$.

Синхронные RS-триггеры (flip-flops)

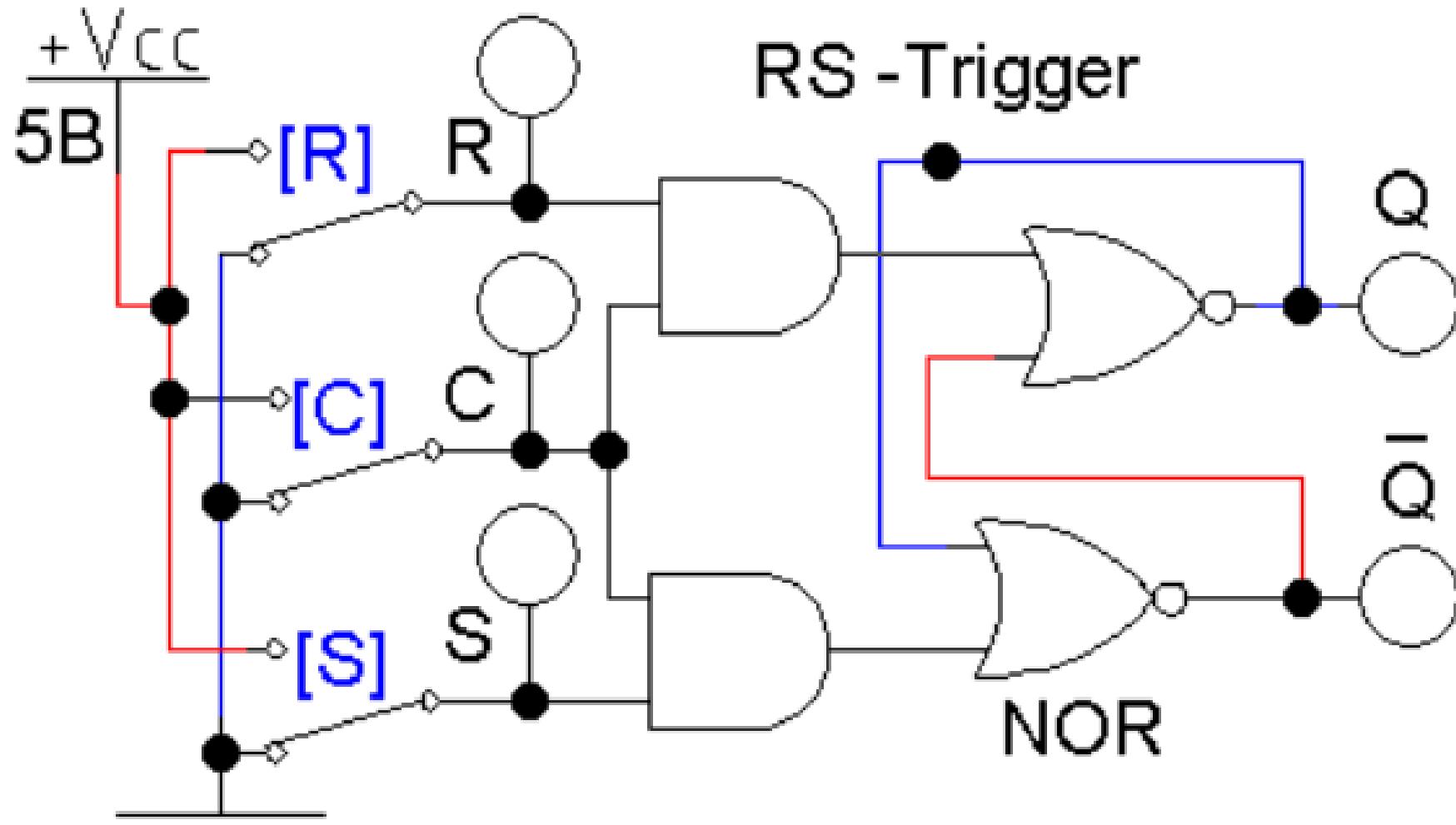
Двухступенчатый RS-триггер:



RS-триггер с
динамическим управлением:



Синхронный RS-триггер



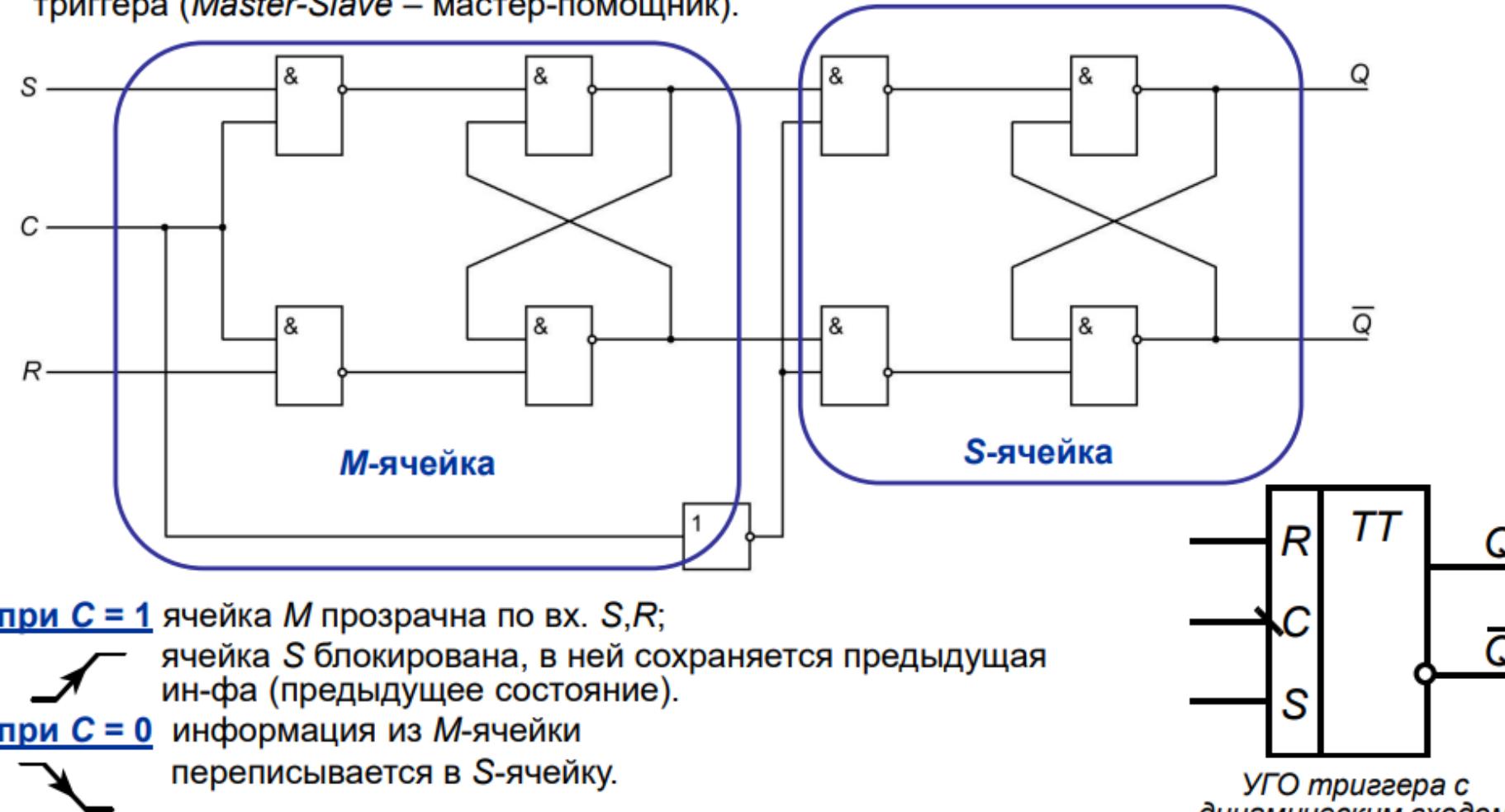
RS -Trigger

NOR

C	S	R	Q_{n+1}
1	0	0	Q_n
1	0	1	0
1	1	0	1
1	1	1	X
0	-	-	0

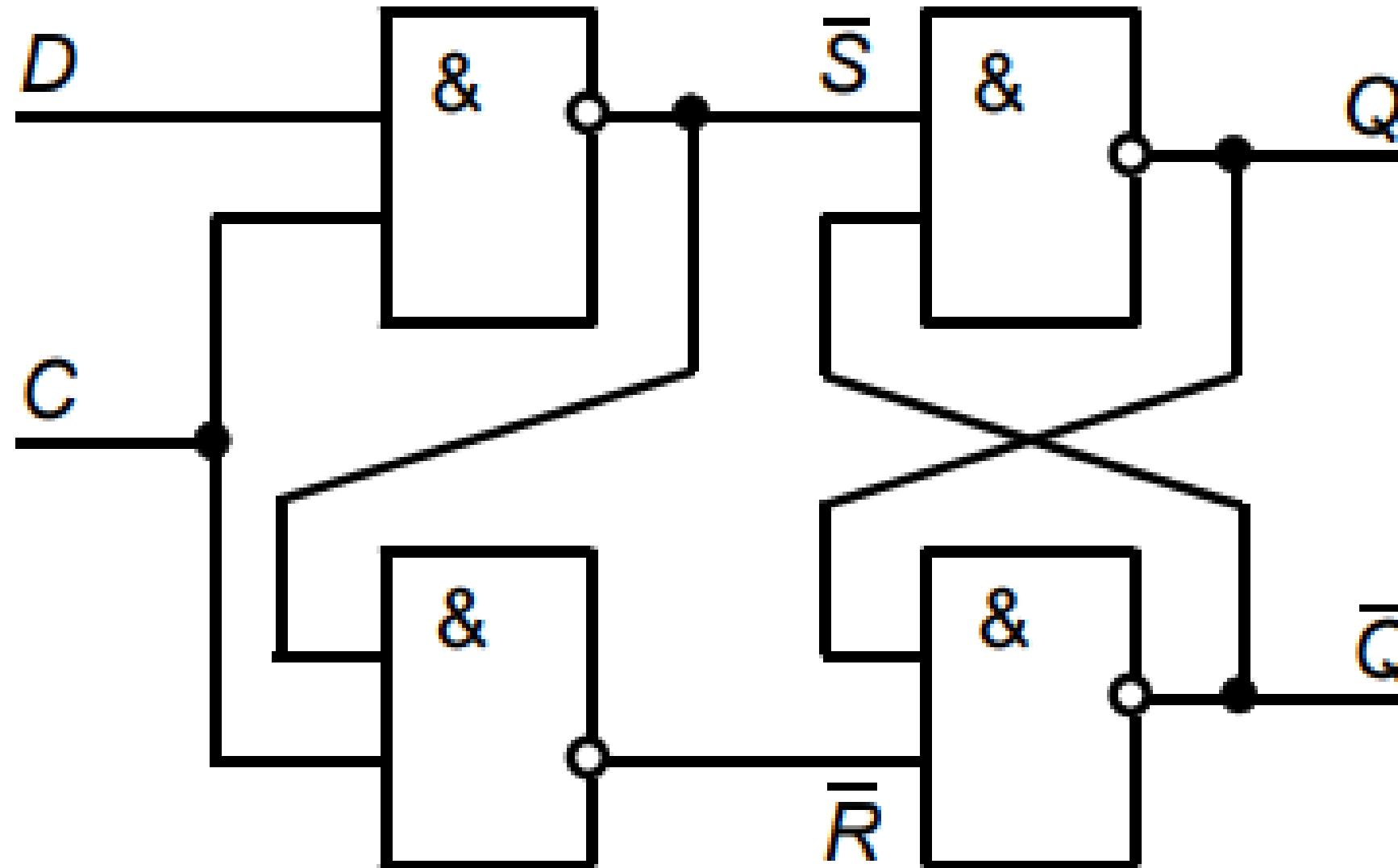
Двухступенчатый RS-триггер (MS-схема)

Для устранения прозрачности схемы триггера при $C = 1$ используется *MS*-схема триггера (*Master-Slave* – мастер-помощник).



Т.о. работа триггера осуществляется в 2 такта: в первом такте – установка 1 триггера (*M-ячейки*);
во втором такте информация из *M-ячейки* переписывается в *S-ячейку*.

D-триггер



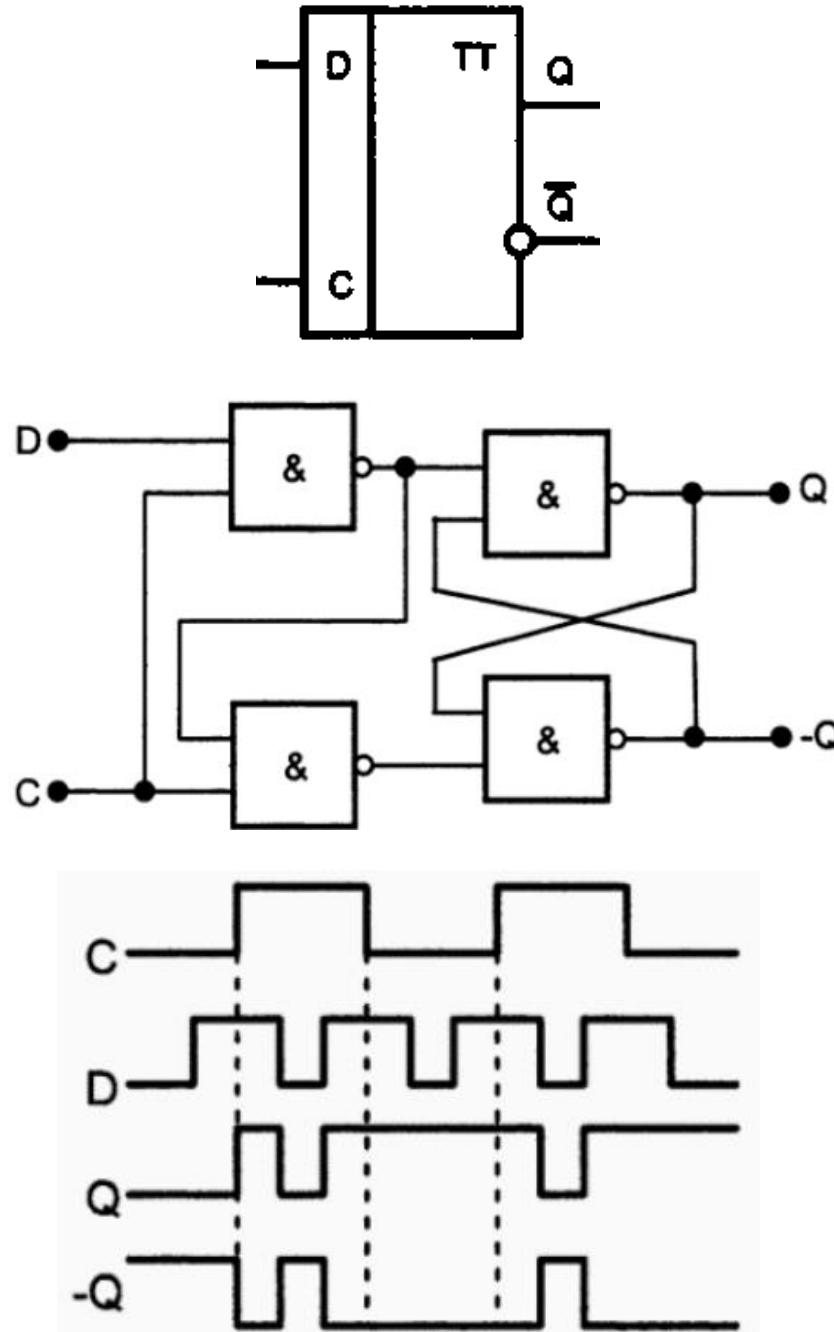
D-триггер. Другие названия: триггер задержки, фиксатор или прозрачная защелка.

D-триггер

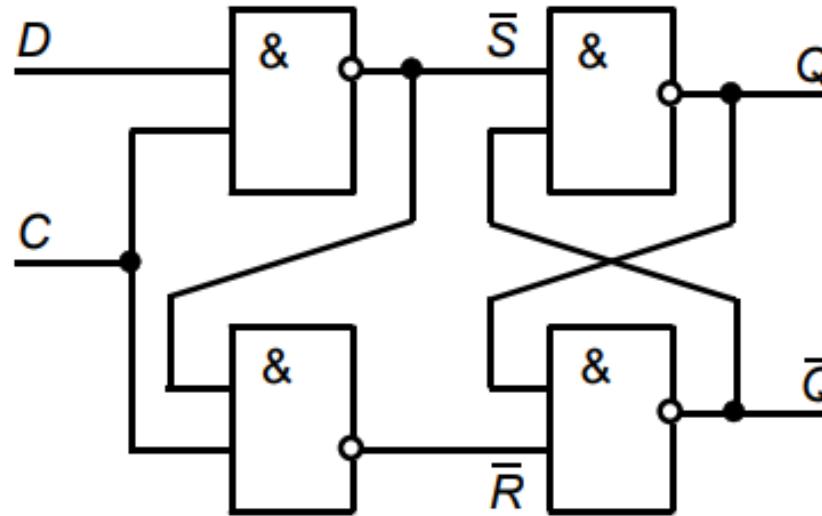
- D-триггер – **триггер задержки** или информационный триггер, при разрешающем сигнале на тактовом входе устанавливается в состояние, соответствующее значению на входе D.
- D-триггер имеет два входа: **C** – синхронизации (Clock) и **D** – информационный (Data).

Входы		Выходы	
C	D	Q	Q _{t+1}
0	0	0	0
	0	1	1
	1	0	0
	1	1	1
1	0	0	0
	0	1	0
	1	0	1
	1	1	1

- В тактируемом фронтом D-триггере изменение потенциала на входе D, синхронное с тактовыми импульсами, повторяется на выходе Q с задержкой на один период тактовых импульсов (отсюда и название-триггер задержки).



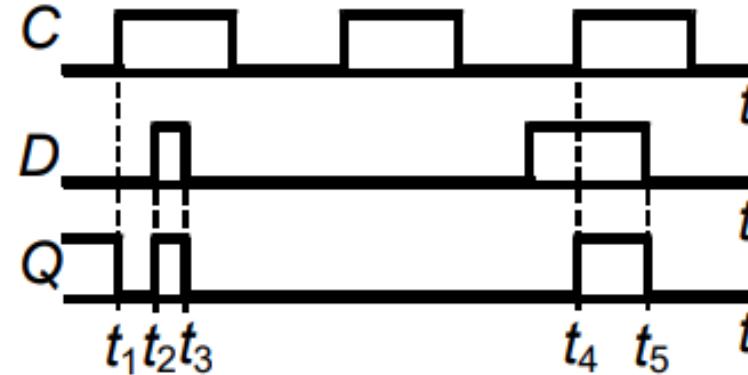
D-триггер



$$Q^{n+1} = D$$

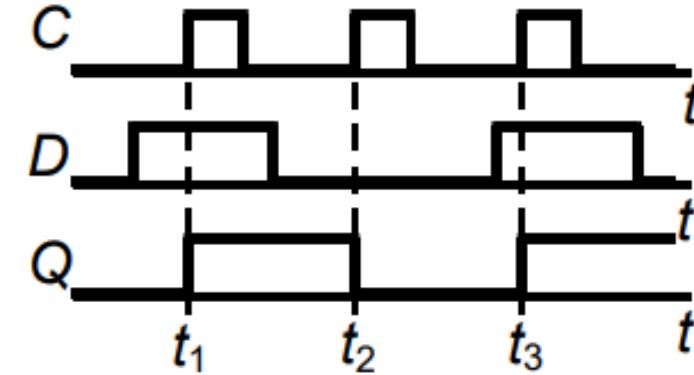
C	D	Q^n	Q^{n+1}	
0	X	Q^n	Q^n	хранение
1	1	X	1	запись «1»
1	0	X	0	запись «0»

Прозрачная защелка

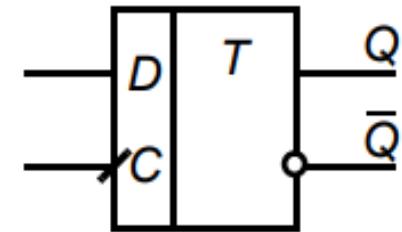


при $C = 1$ изменение сигнала на входе D вызывает изменение сигнала на выходе Q .

Триггер задержки



Триггер задерживает (хранит) информацию до следующего синхроимпульса.

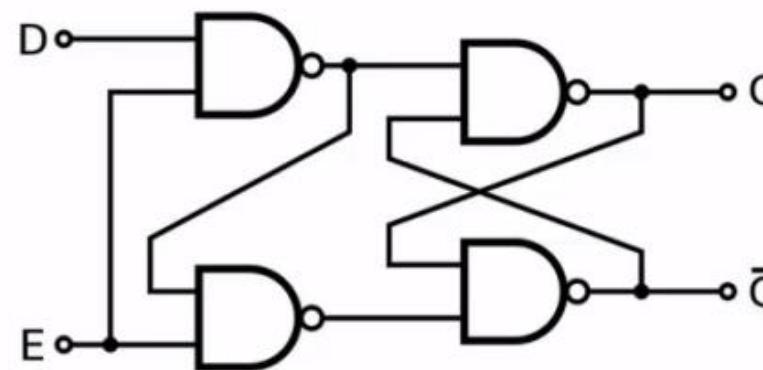


УГО D-триггера с динамическим входом

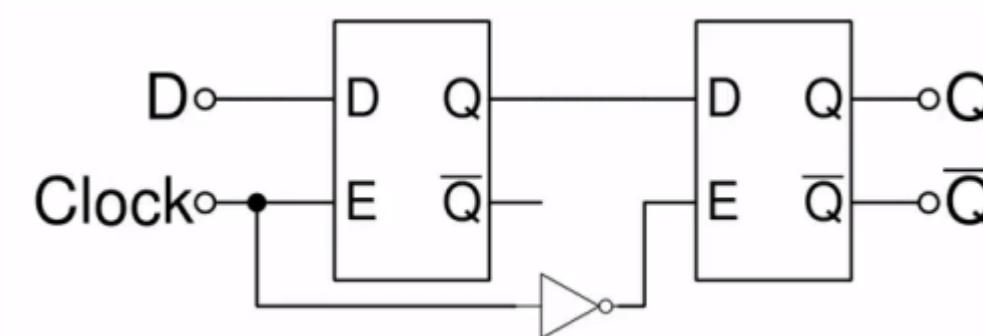
D-триггер

- **D-триггер** (**D** от англ, **delay** — задержка, либо от **data** — данные) — запоминает состояние входа и выдаёт его на выход.
- Имеет минимум два входа - данных и синхронизации.

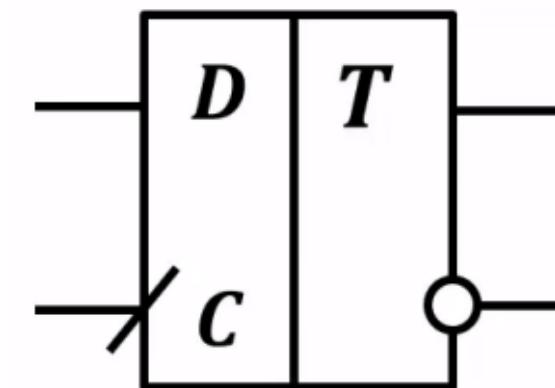
Управляемая D-защёлка



Двухступенчатый D-триггер

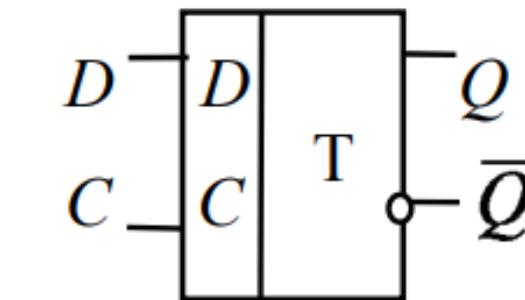


D-триггер с динамическим управлением по положительному перепаду

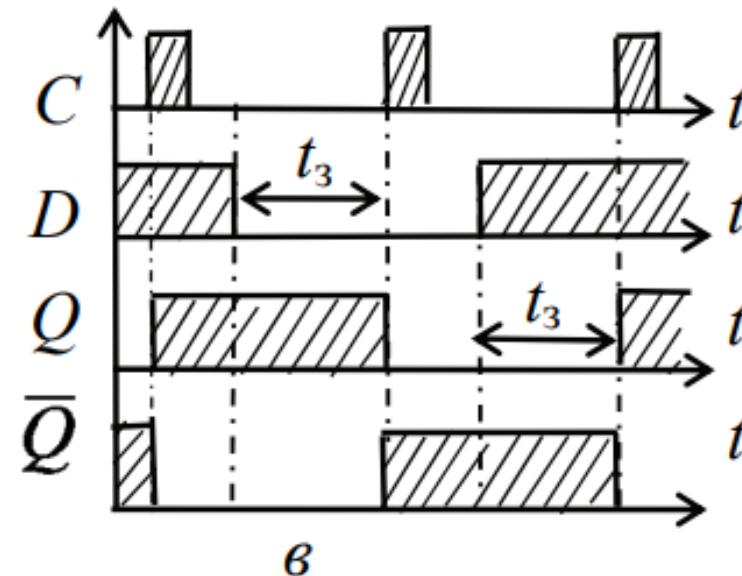


D-триггеры с динамическим управлением являются самыми распространёнными типами триггеров

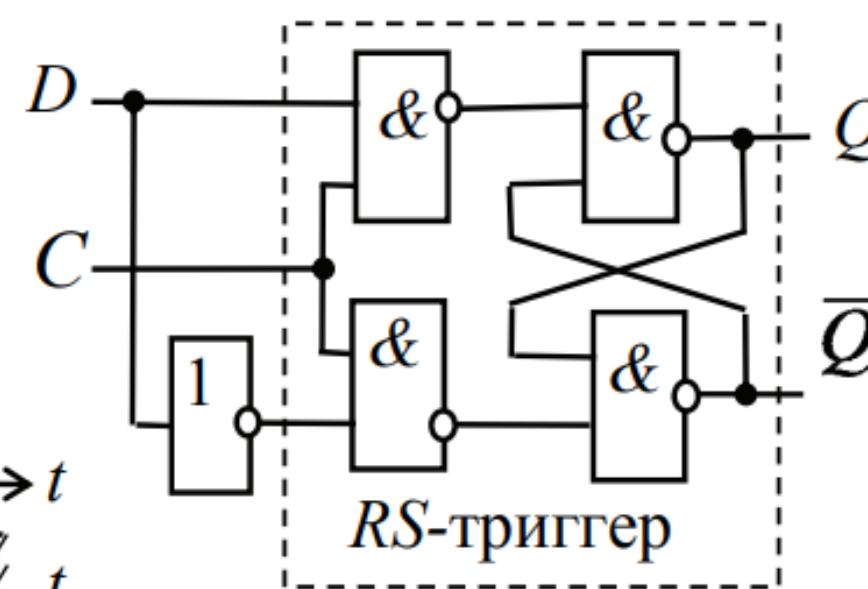
D-триггер



а



в



б

Таблица переходов

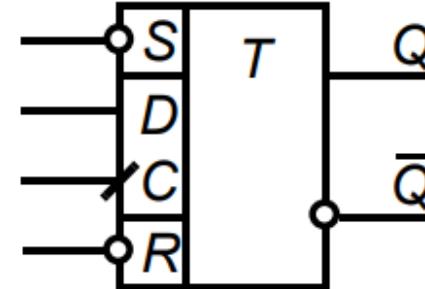
C^t	D^t	Q^t	Q^{t+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

г

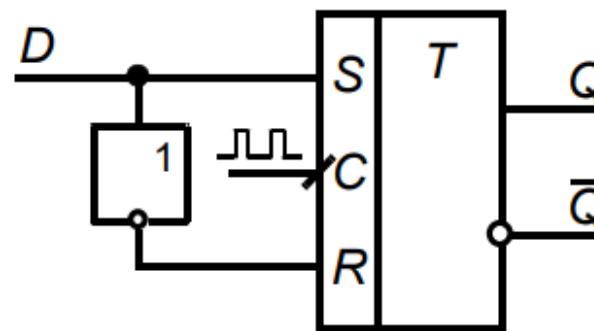
а – условное графическое обозначение; б – схема реализации;
в – временная диаграмма; г – таблица переходов

D-триггер

D-триггер с входами
предустановки R и S.



D-триггер можно построить на
основе синхронного RS-триггера.

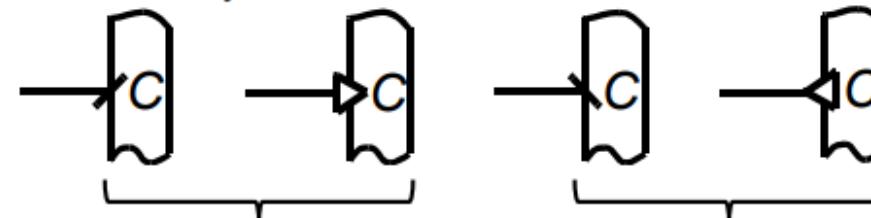


Переключение происходит при переходе
сигнала на тактовом входе из 0 в 1 (), т.е.
по переднему фронту синхроимпульса.

асинхронные входы

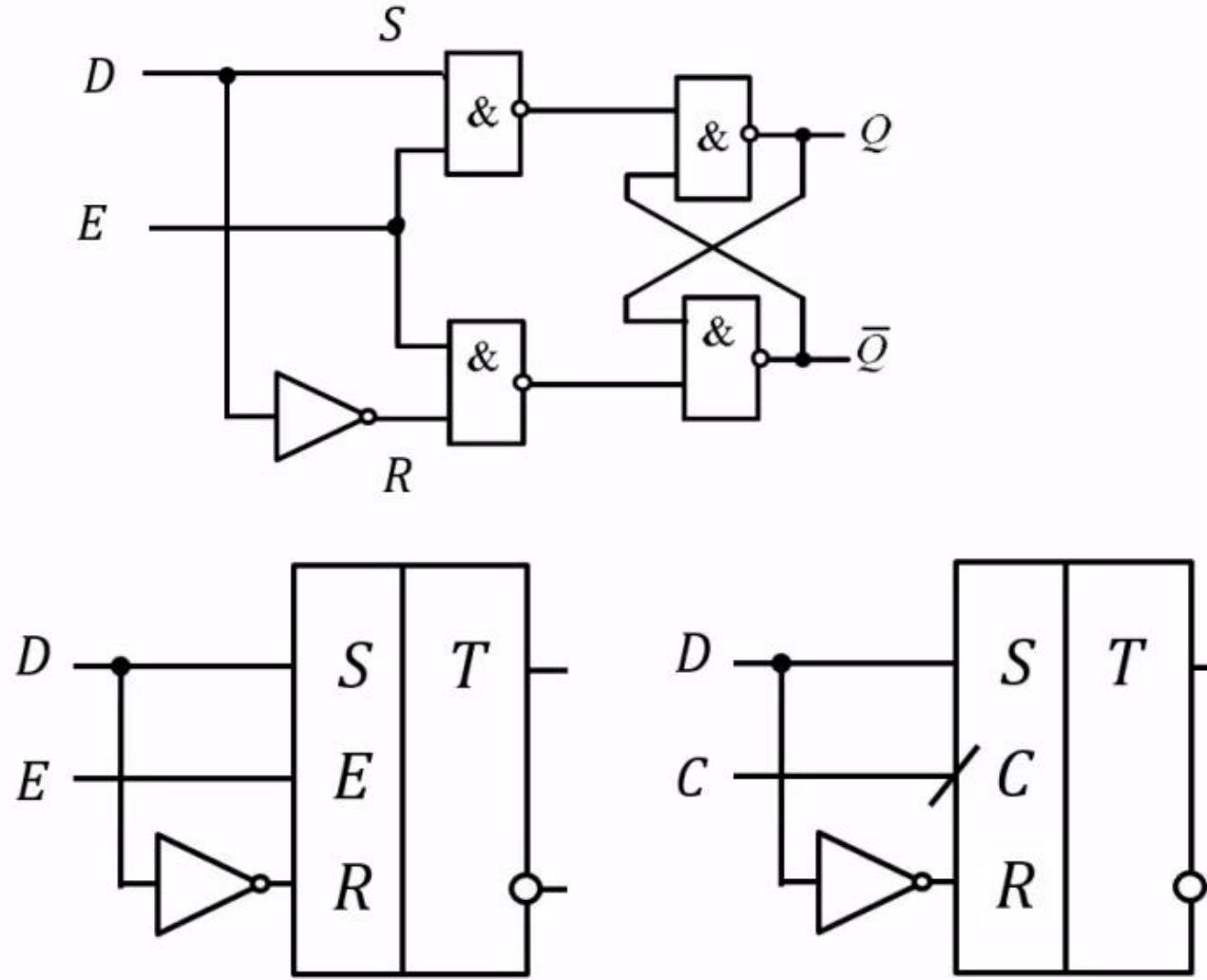
\bar{S}	\bar{R}	C	D	Q		
0	1	X	X	1	асинхр. уст-ка «1»	обычный асинхр. RS-триггер
1	0	X	X	0	асинхр. уст-ка «0»	
0	0	X	X		запрещенное состояние	
1	1		1	1	синхр. уст-ка «1»	обычный D-триггер
1	1		0	0	синхр. уст-ка «0»	

Используемые обозначения динамического входа

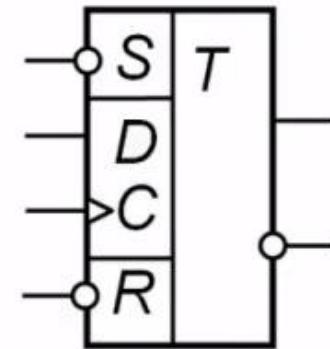
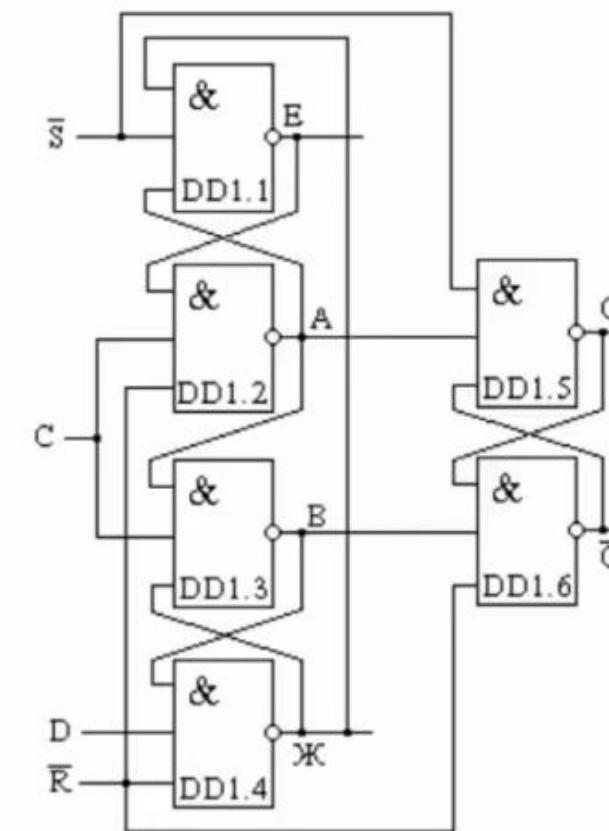


Срабатывание происходит при
переходе тактового сигнала из 1 в
0 (, т.е. по заднему фронту
(срезу) синхроимпульса.

D-триггер на основе SR-триггера

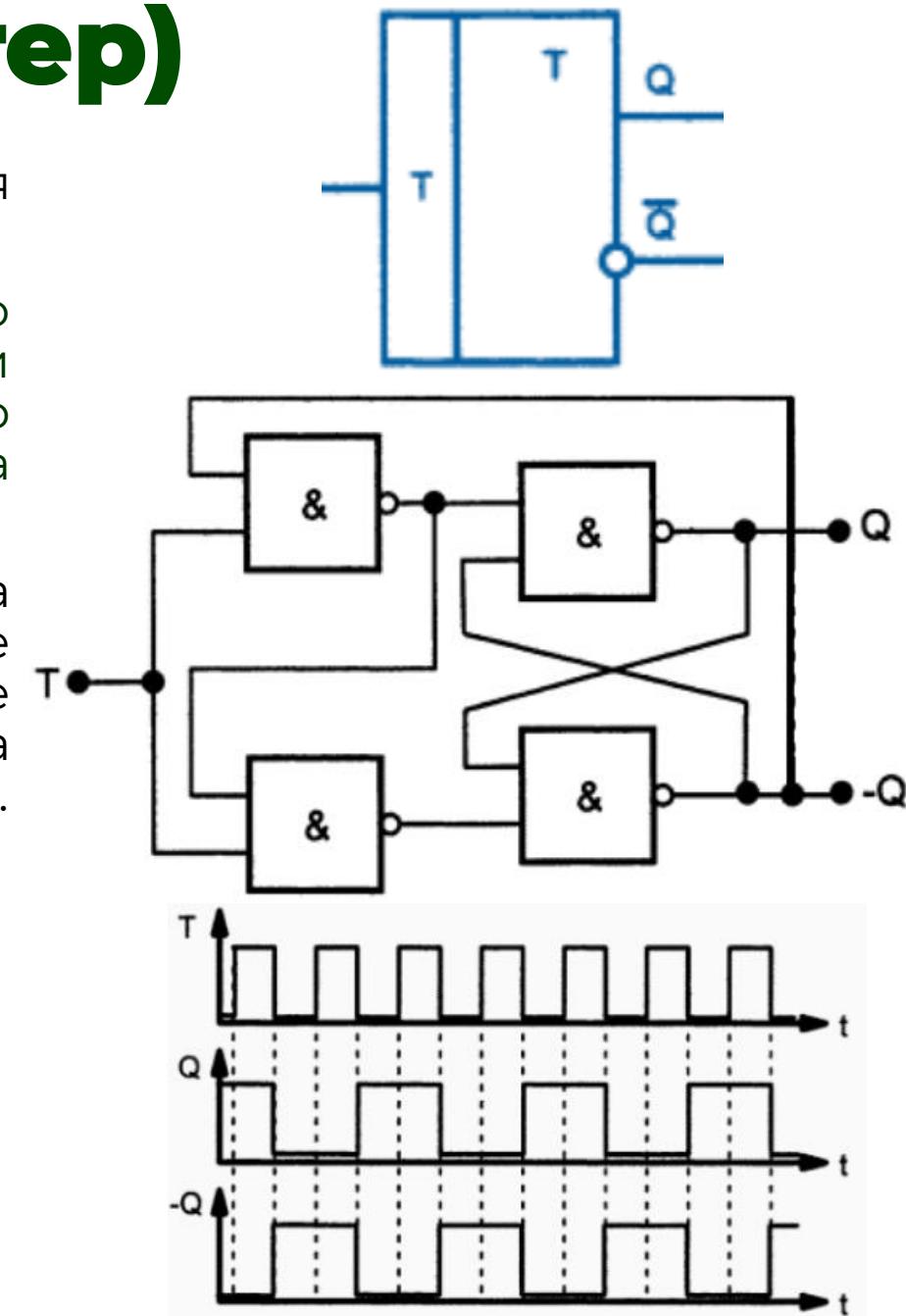
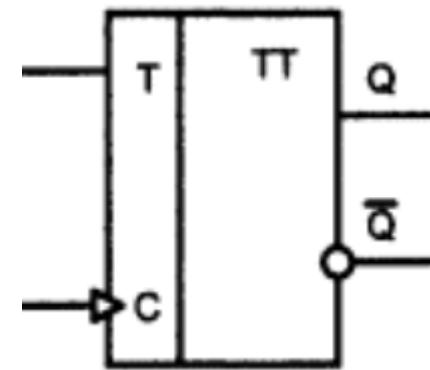
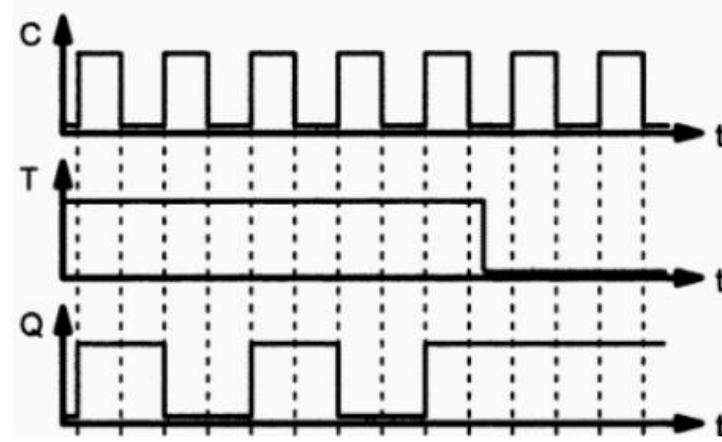


D-триггер с дополнительными асинхронными инверсными входами S и R:

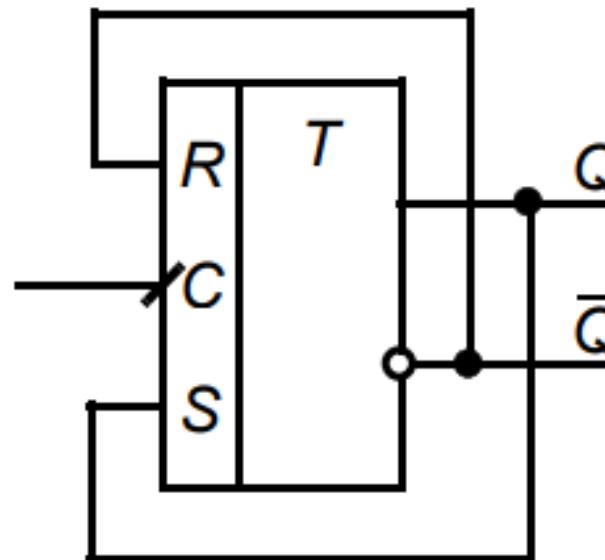


Т-триггер (счетный триггер)

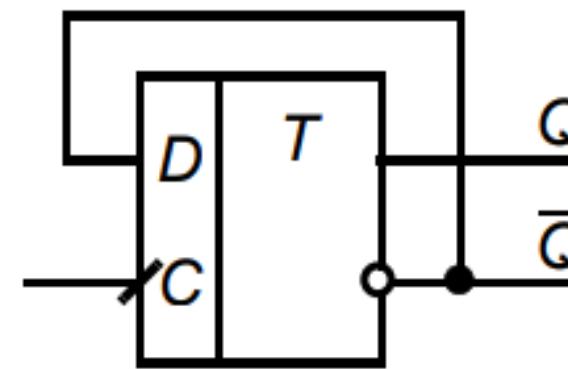
- **Т-триггеры** или счетные триггеры используют для деления частоты (в двоичных счетчиках).
- Бывают тактируемые (имеют входы Т и С) либо асинхронные (имеют единственный вход Т - Toggle). При поступлении импульса на этот вход Т-триггер переключается, при поступлении следующего импульса — возвращается в исходное состояние.
- **Синхронный Т-триггер** при логической единице на входе Т по каждому такту на входе С изменяет свое логическое состояние на противоположное и не изменяет выходное состояние при логическом нуле на входе Т. Т-триггеры бывают только тактируемые фронтом.



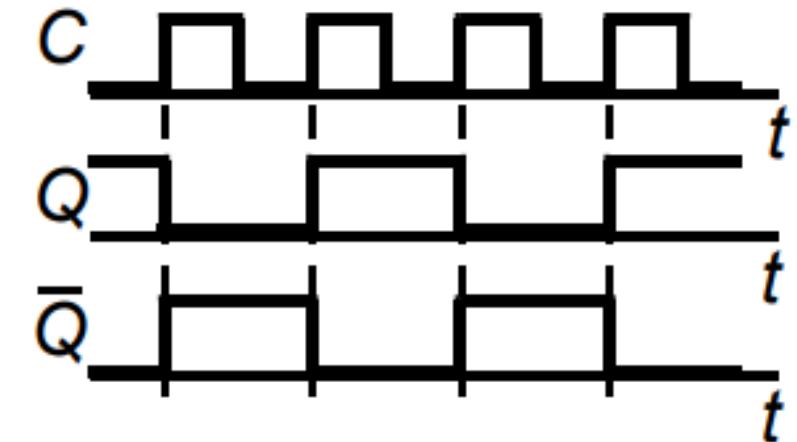
Т-триггер



Т-триггер на базе
RST-триггера



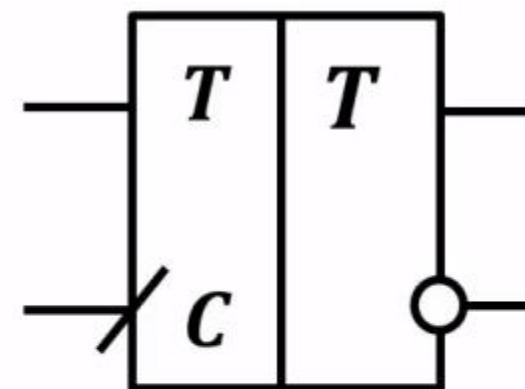
Т-триггер на базе
D-триггера



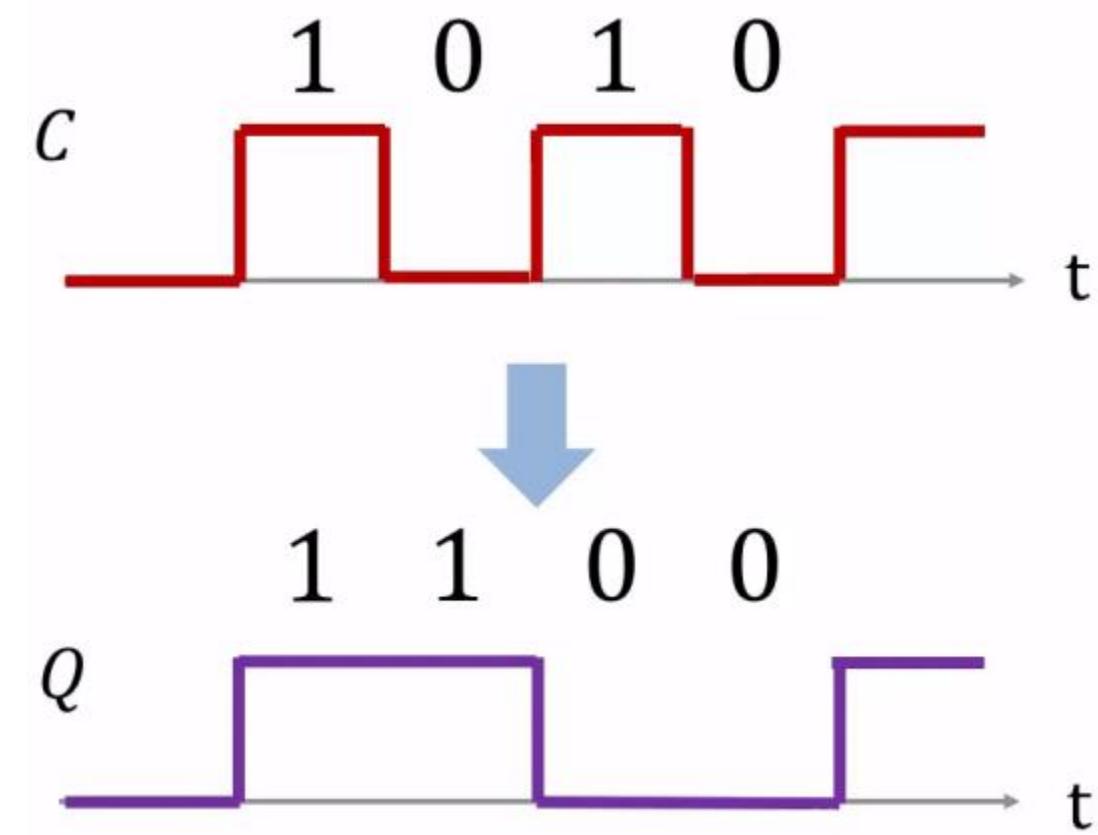
Диаграммы работы
Т-триггера

Т-триггер

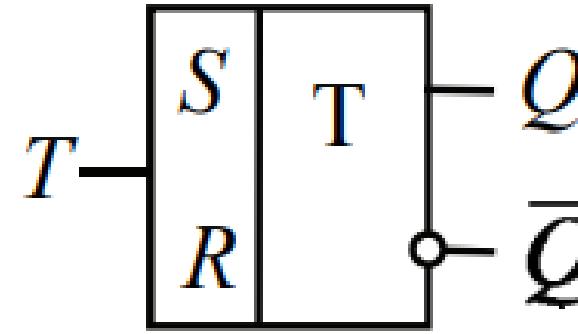
- **Т-триггер** (от англ, toggle – переключатель) – меняет своё состояние на каждом перепаде синхросигнала.
- Т-триггер часто называют счётным триггером, так как он является простейшим счётчиком по модулю



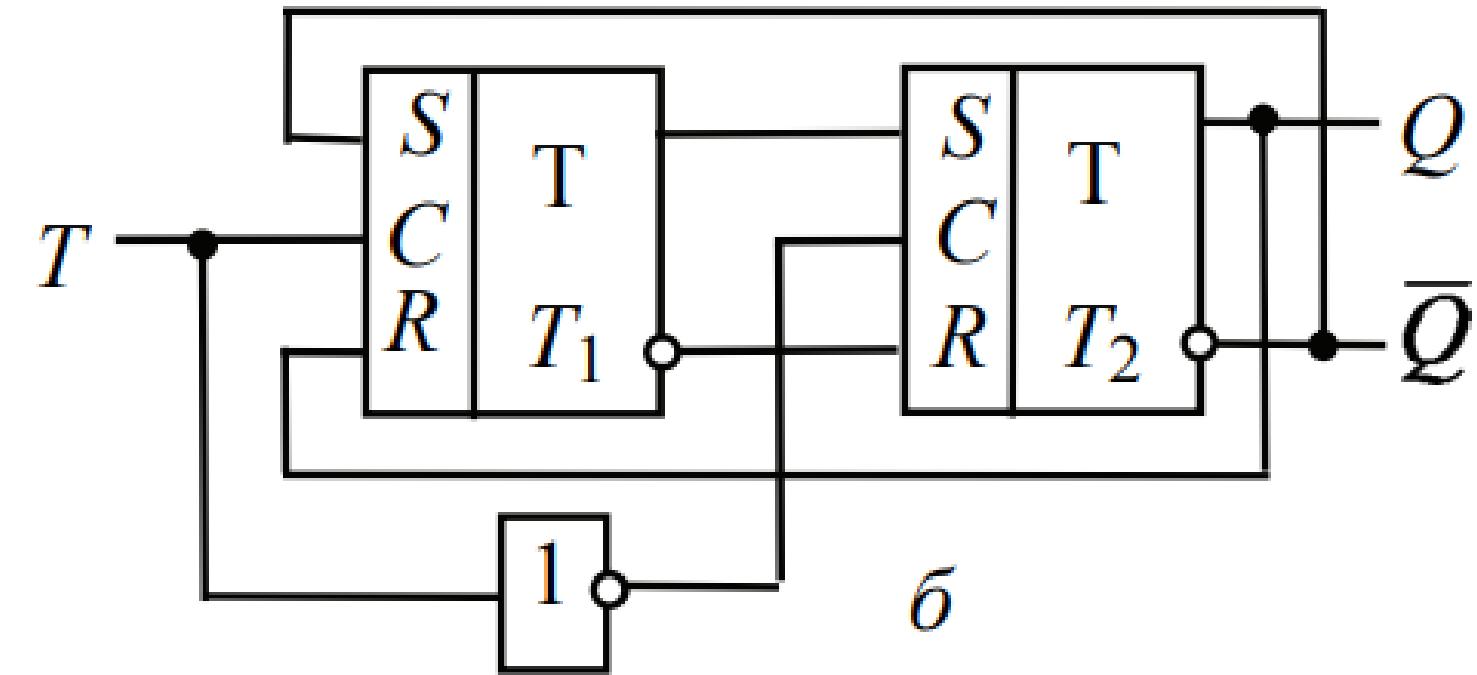
Вход **T** принимает разрешающий сигнал
счёта/переключения (статическое управление)



Т-триггер



a



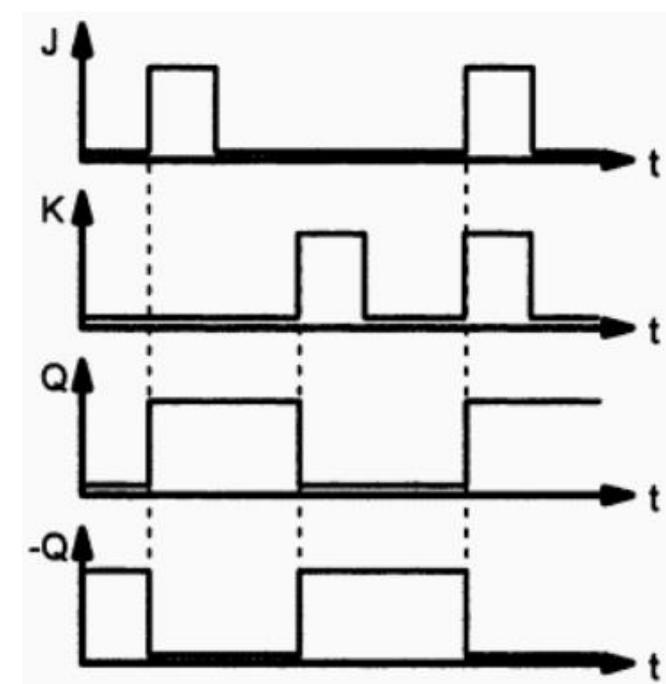
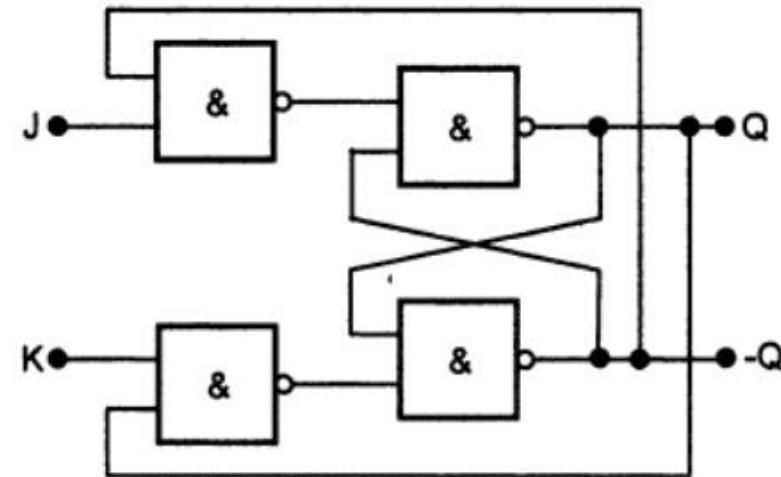
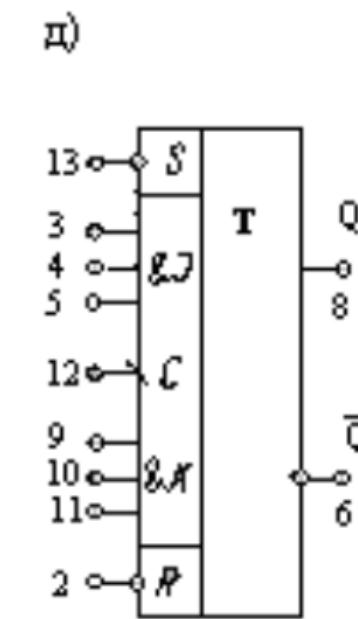
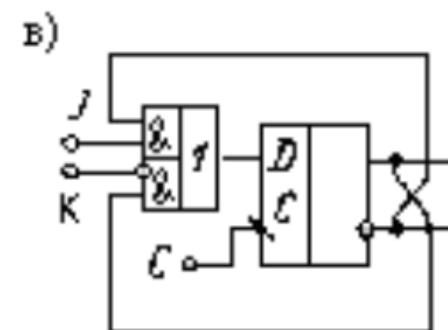
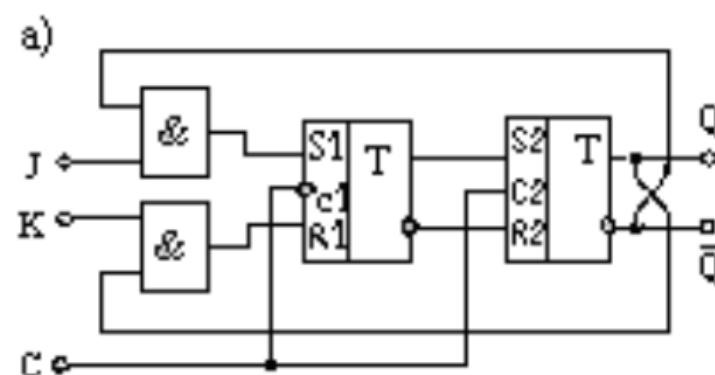
б

а – условное графическое обозначение;

б – схема реализации Т-триггера на базе двух синхронных RS-триггеров

JK-триггер

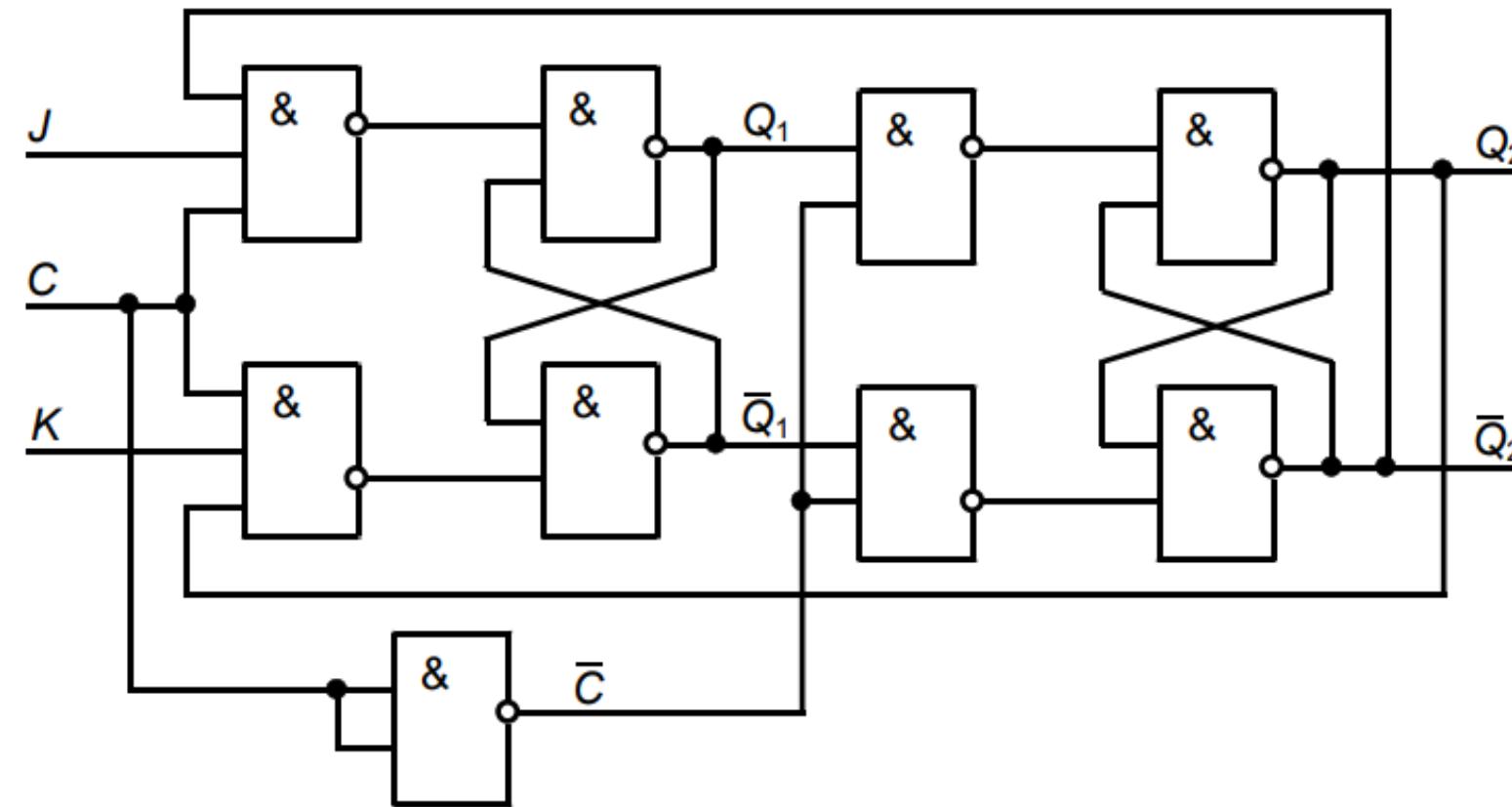
- **JK-триггер** имеет два управляющих входа **J** (jump) и **K** (kill) – это входы установки триггера в единицу и нуль, тактовый вход **C**.
- В отличие от RS-триггера, в JK-триггере наличие двух единичных управляющих сигналов ($J=K=1$) приводит к переходу триггера в противоположное состояние, т. е. JK-триггер работает как T-триггер.
- JK-триггеры тактируются только перепадом потенциала на тактовом входе.



JK-триггер

JK-триггер – наиболее широко используемый универсальный триггер.

На его основе могут быть построены другие типы триггеров. JK-триггер получается из Т-триггера введением дополнительных входов J и K. Триггер может быть одноступенчатым и двухступенчатым, со статическим и динамическим переключением.



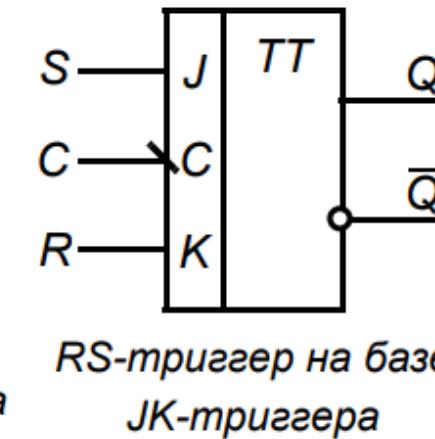
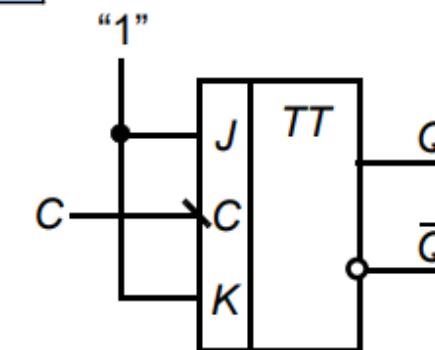
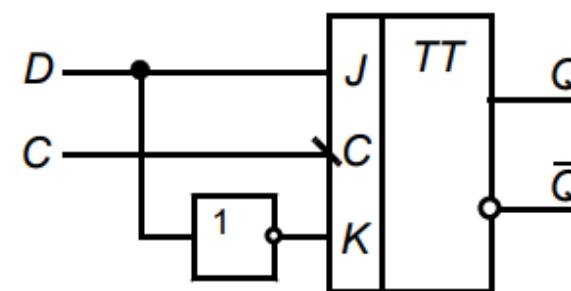
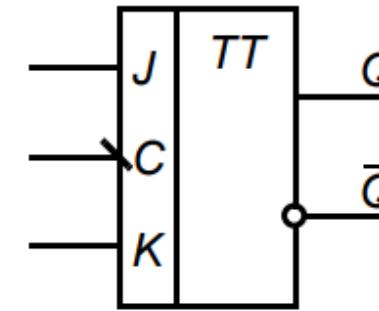
JK-триггер выполненный по MS-схеме

JK-триггер

Логическое выражение для асинхронного триггера: $Q^{n+1} = J \cdot \overline{Q^n} + \overline{K} \cdot Q^n$

Логическое выражение для синхронного триггера: $Q^{n+1} = (J \cdot \overline{Q^n} + \overline{K} \cdot Q^n) \cdot C + Q^n \cdot \overline{C}$

C	J	K	Q^n	Q^{n+1}	
0	0	0	X	Q^n	хранение
0	0	1	X	Q^n	
0	1	0	X	Q^n	
0	1	1	X	Q^n	
1	0	0	X	Q^n	хранение
1	0	1	1	0	
1	1	0	0	1	
1	1	1	Q^n	$\overline{Q^n}$	



JK-триггер

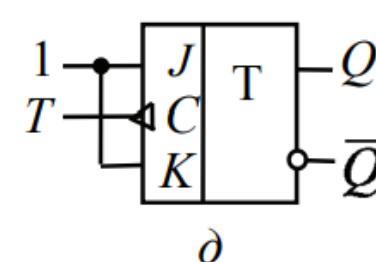
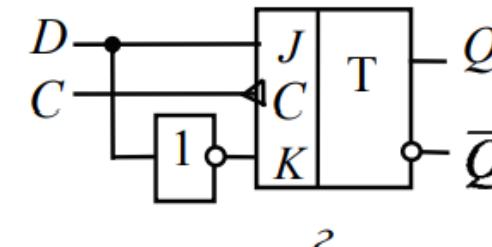
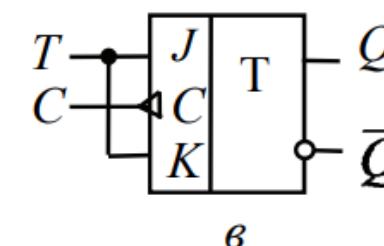
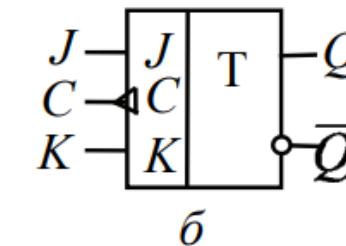
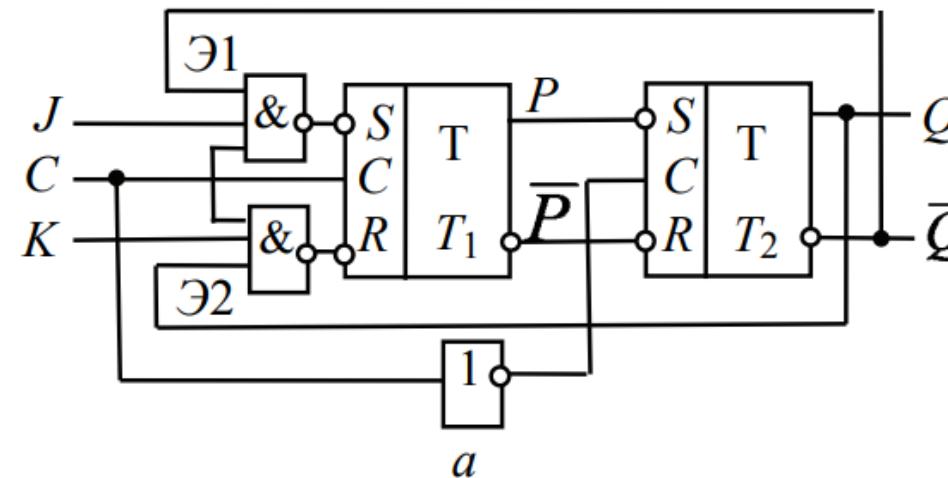
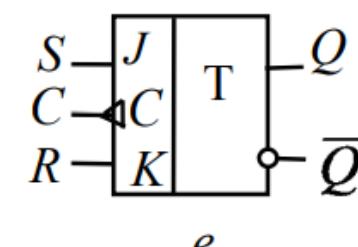


Таблица переходов

J^t	K^t	Q^t	Q^{t+1}
0	0	0	0
1	0	0	1
0	1	0	0
1	1	0	1
0	0	1	1
1	0	1	1
0	1	1	0
1	1	1	0

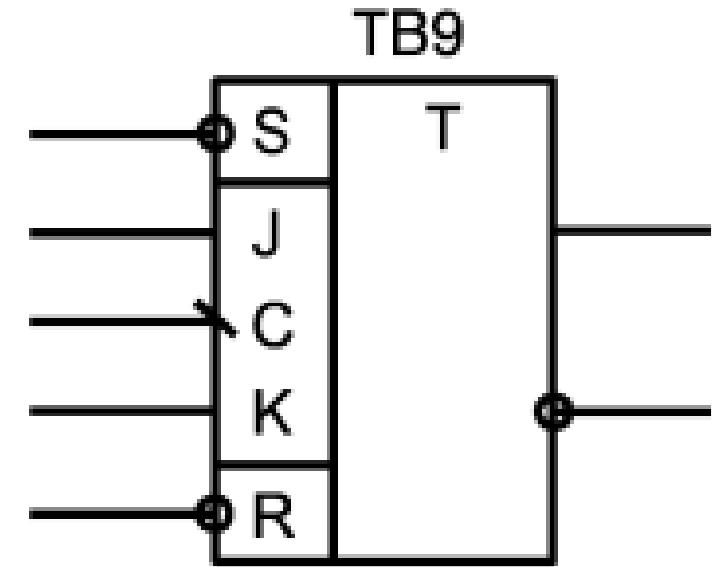
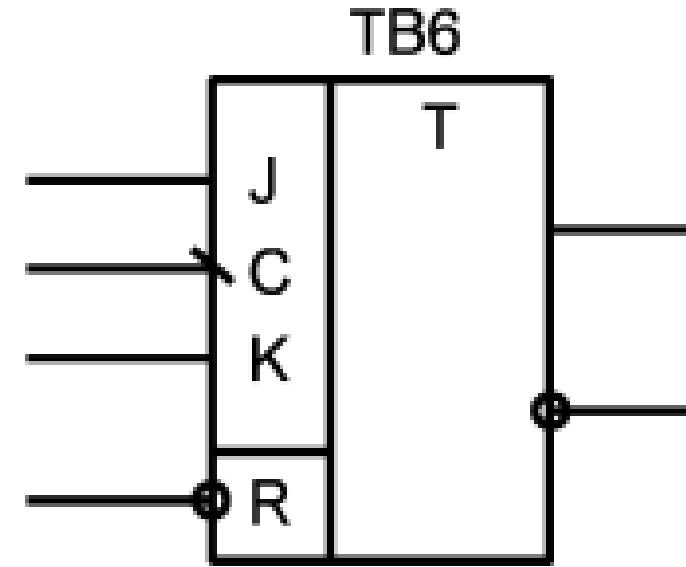
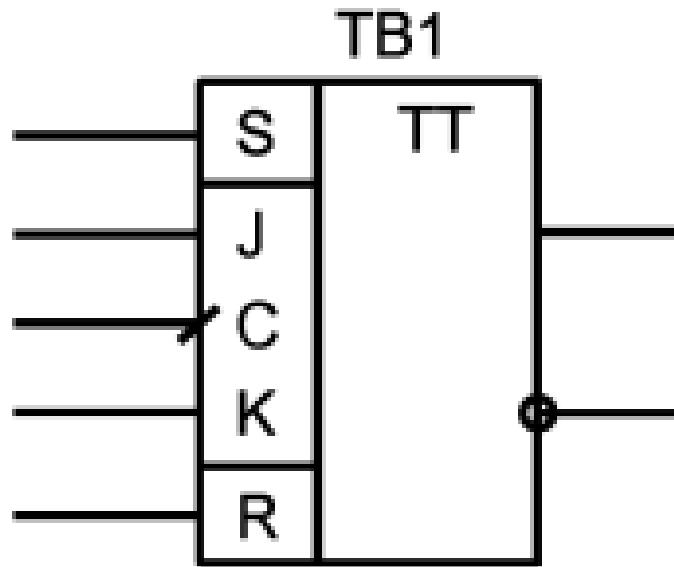
ж



а – функциональная схема; б – Т-триггер; в – синхронный JK-триггер;
г – асинхронный Т-триггер; д – D-триггер; е – синхронный RS-триггер; ж – таблица переходов

JK-триггер

Схемы стандартных триггеров типа JK



Триггер ТВ1 указанных серий — двухступенчатый, свободный от "захватов", в связи с чем трактуется как управляемый фронтом.
Триггеры ТВ6, ТВ9 — с внутренними задержками, одноступенчатые, переключаемые отрицательным фронтом синхросигнала.

JK-триггер, как базовый элемент

- Из JK-триггера, соединив определенные выводы, можно получить любой иной вид триггера.
- В отличие от RS-триггера, JK-триггер не имеет запрещенных комбинаций входных сигналов.

Входы					Выходы	
-S	-R	C	J	K	Q	-Q
0	1	X	X	X	1	0
1	0	X	X	X	0	1
0	0	X	X	X	H*	
1	1	1→0	1	0	1	0
1	1	1→0	0	1	0	1
1	1	1→0	0	0	Не изменяется	
1	1	1→0	1	1	Меняется на противоположное	
1	1	1	X	X	Не изменяется	
1	1	0	X	X	Не изменяется	
1	1	0→1	X	X	Не изменяется	

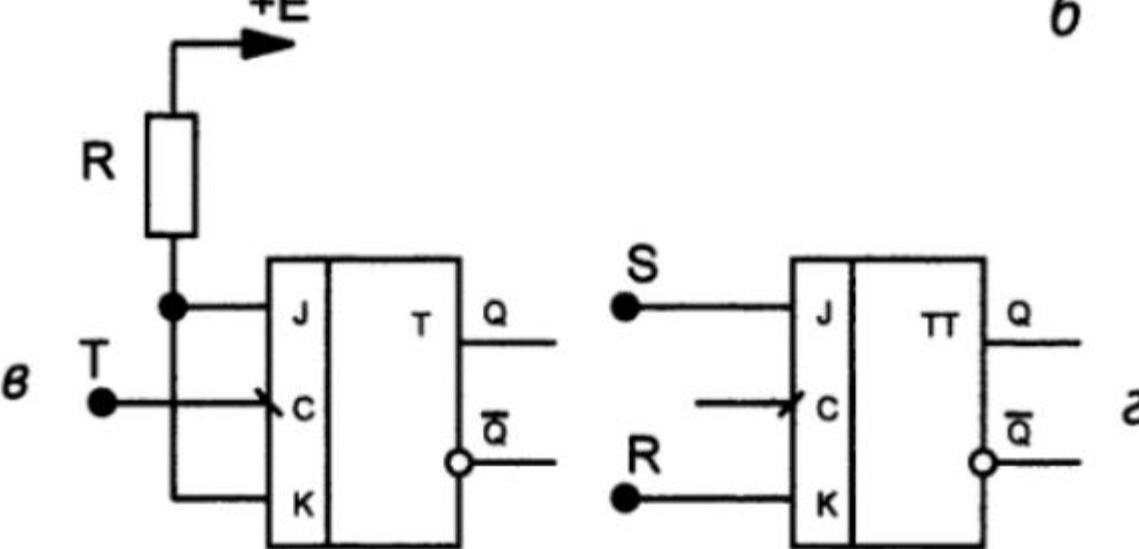
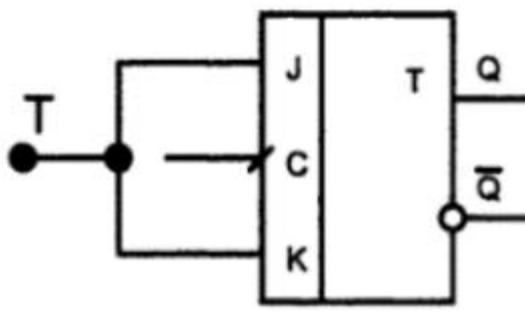
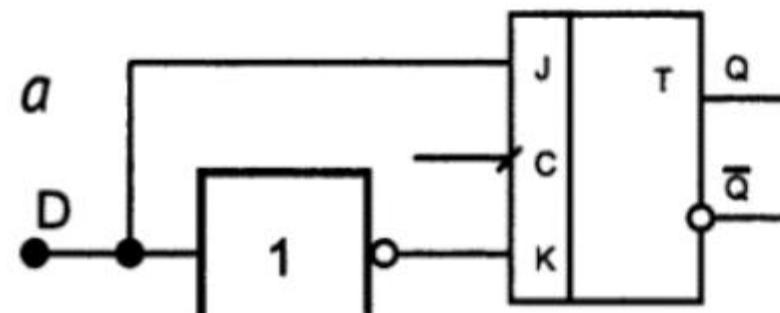
X - безразличное состояние; H* - неустойчивое состояние

- При наличии на J и K-входах уровня 1 по каждому тактовому сигналу, поступающему на вход C, триггер изменяет свое состояние, т. е. работает как счетный T-триггер.
- При любых других комбинациях на входах J и K он работает как RS-триггер (вход J соответствует входу S, а вход K — входу R), а при определенном постоянном сигнале на одном из этих входов — как D-триггер.

J	K	Q	Режим работы
0	0	Q	Хранение
1	0	1	Установка в 1
0	1	0	Сброс в 0
1	1	-Q	Счет

JK-триггер, как базовый элемент

- Из JK-триггера несложно получить другие виды триггеров:



а) D-триггер; б) синхронный Т-триггер;
в) асинхронный Т-триггер; г) RS-триггер.

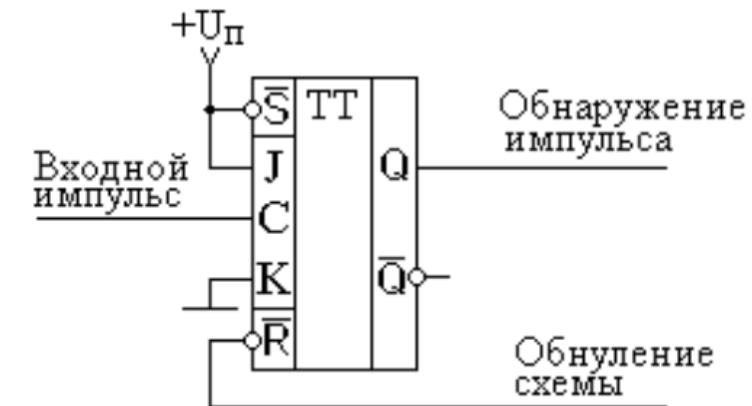


Схема обнаружения короткого импульса на JK-триггере

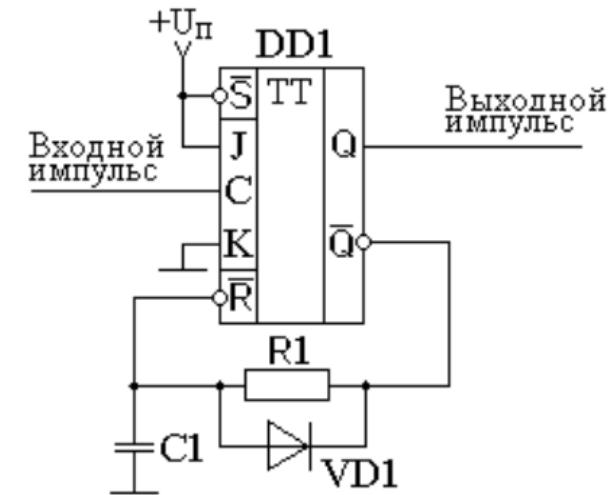
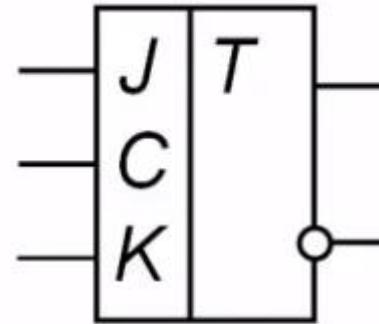
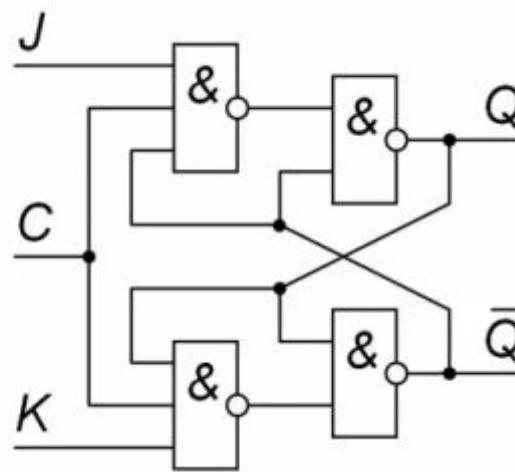


Схема ждущего мультивибратора

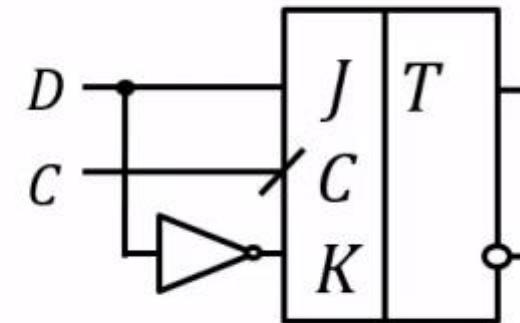
JK-триггер



JK-триггер на основе RS-триггера:



D-триггер на основе JK-триггера:



T-триггер на основе JK-триггера:

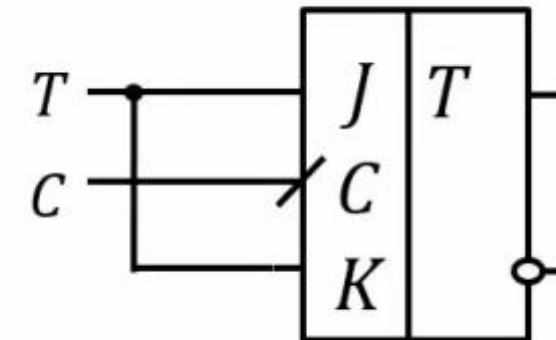


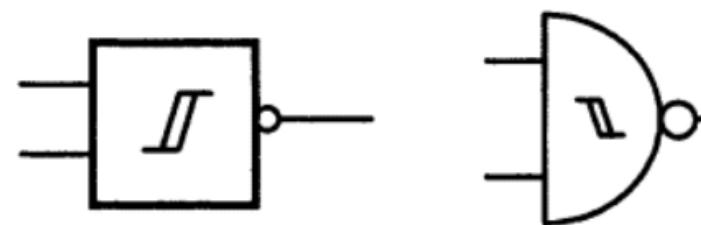
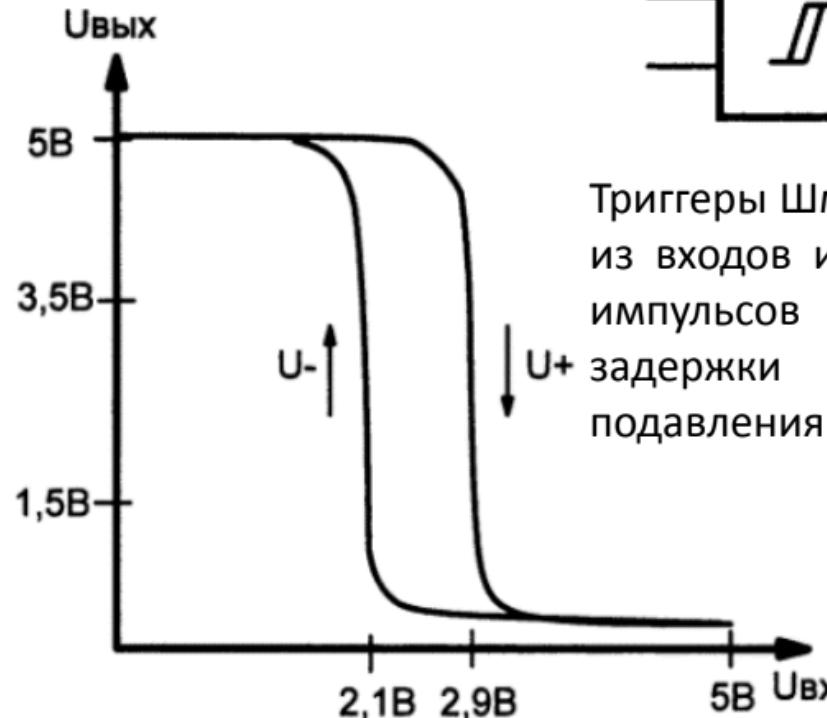
Таблица перехода JK-триггера:

J	K	Q(t)	Q(t+1)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

JK-триггер - наиболее универсальный из всех. На его основе можно построить D-триггер и T-триггер. Применяется только синхронный, с динамическим управлением.

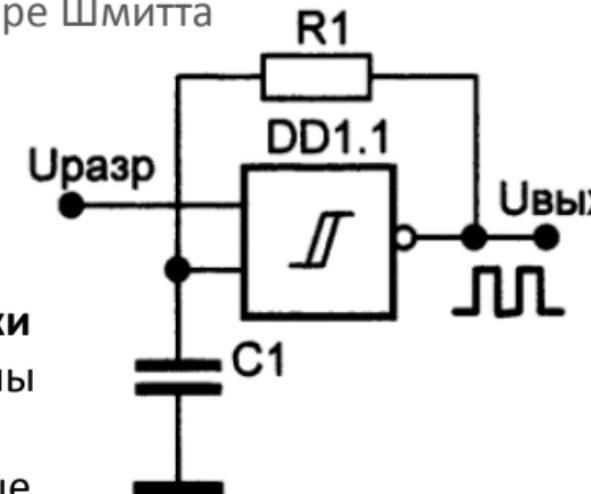
Триггер Шмитта

ТРИГГЕР ШМИТТА

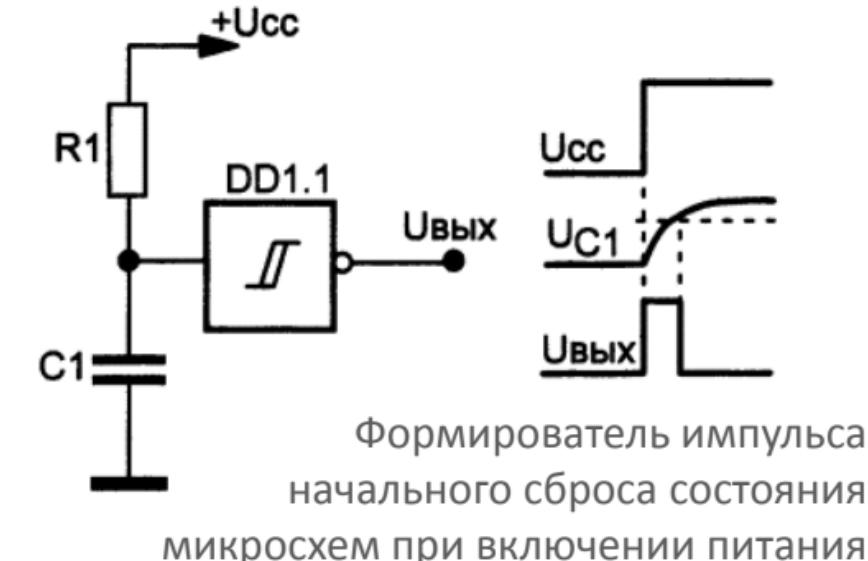


Триггеры Шмитта с RC-задержкой на одном из входов используют для формирования импульсов регулируемой длительности, задержки прохождения импульсов, подавления дребезга контактов и т. д.

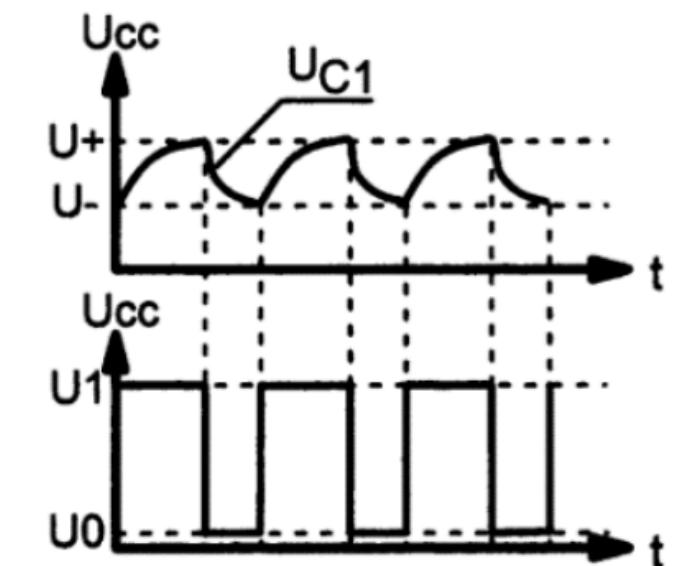
Генератор прямоугольных импульсов на триггере Шмитта



Наличие петли гистерезиса при переключении триггера Шмитта приводит к тому, что любые **помехи** цифрового сигнала с амплитудой, меньшей величины $U_{\text{вкл}} - U_{\text{откл}} = 800 \text{ мВ}$, отсекаются, а любые фронты и спады (даже самые пологие) преобразуются в крутые фронты и спады выходного сигнала.



Формирователь импульса начального сброса состояния микросхем при включении питания



Прозрачные и непрозрачные триггеры

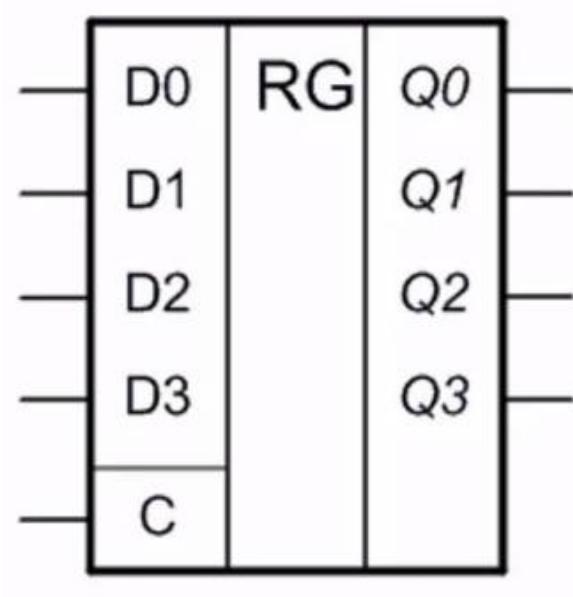
- Во время действия тактового импульса С состояния управляющих входов R, S, D или J, K не должны меняться.
- В противном случае может произойти ложное срабатывание.
- Если при $C = 1$ при смене состояния управляющих входов R, S, D или J, K сигнал на выходе триггера меняется, то триггер – прозрачный.

Особенности эксплуатации синхронных триггеров со статическим управлением

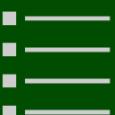
- Во время действия тактового импульса С состояния управляющих входов R, S, D или J, K не должны меняться.
- В противном случае может произойти ложное срабатывание.
- **1. Проскок помехи** – при неизменных состояниях на управляющих входах R, S, D, J, K состояние на выходе триггера изменяется сразу после прихода помехи при $C=1$ без сохранения этого состояния до следующего тактового импульса.
- **2. Захват помехи** – при неизменных состояниях на управляющих входах R, S, D или J, K состояние на выходе триггера изменяется по тактовому импульсу и сохраняется до прихода следующего тактового импульса.

Роль Триггеров в современных электросистемах

- Триггеры играют ключевую роль в цифровой электронике, так как обеспечивают хранение, обработку и управление цифровыми сигналами.
- **Они выполняют важные функции**, включая:
 - **Хранение информации** – используются в регистрах и оперативной памяти компьютеров, где каждый элемент может хранить один бит данных.
 - **Формирование тактовых импульсов** – применяются в генераторах импульсов и тактовых схемах для синхронизации работы цифровых устройств.
 - **Преобразование сигналов** – например, делители частоты на основе триггеров используются в системах связи, радиотехнике и цифровых часах.
 - **Создание автоматов управления** – лежат в основе конечных автоматов, которые управляют процессами в промышленных контроллерах, бытовой электронике и робототехнике.
 - **Обработка сигналов** – в схемах управления помогают фильтровать и стабилизировать сигналы, предотвращая ложные срабатывания и помехи.
- Благодаря этим возможностям они остаются незаменимыми компонентами в разработке современных цифровых систем.



Последовательная логика Регистры



Регистр

- **Регистр** — устройство для записи, хранения и считывания n-разрядных двоичных данных и выполнения других операций над ними.
- **Регистр** — это устройство хранения данных в цифровых схемах, которое представляет собой набор элементов памяти, способных сохранять и передавать двоичные значения.
- **Регистр** – последовательностное устройство, предназначенное для приема, хранения, передачи и преобразования информации.
- Обычно он состоит из нескольких триггеров (чаще всего D-триггеров), которые могут хранить одно или несколько двоичных значений (битов).
- **Регистры строятся на базе D-триггеров.**
- Регистры играют ключевую роль в различных электронных устройствах, начиная от простых систем управления до сложных вычислительных машин.

Регистры, Счетчики

- **Существует несколько типов регистров**, отличающихся видом выполняемых операций.
- **Некоторые важные регистры имеют свои названия**, например:
 - **сдвиговый регистр** - предназначен для выполнения операции сдвига;
 - **счетчики** - схемы, способные считать поступающие на вход импульсы. К ним относятся Т-триггеры (название от англ. tumble - опрокидываться). Этот триггер имеет один счетный вход и два выхода. Под действием сигналов триггер меняет свое состояние с нулевого на единичное и наоборот. Число перебрасываний соответствует числу поступивших сигналов;
 - **счетчик команд** - регистр устройства управления процессора (УУ), содержимое которого соответствует адресу очередной выполняемой команды; служит для автоматической выборки программы из последовательных ячеек памяти;
 - **регистр команд** - регистр УУ для хранения кода команды на период времени, необходимый для ее выполнения. Часть его разрядов используется для хранения кода операции, остальные - для хранения кодов адресов operandов.
- В ЭВМ применяются регистры 8, 16, 32, 48 и 64 разрядов.

Регистры

Регистры различаются:

- **По способу приёма и передачи данных:**
 - Параллельные (статические)
 - Последовательные (регистры сдвига)
 - Последовательно-параллельные
- **По способу преобразования данных:**
 - Сдвиговые
 - С преобразованием параллельного кода в последовательный и наоборот
 - С поразрядным логическим умножением (сложением)
 - Последовательного приближения
- **По управлению записью данных**
 - Асинхронные
 - Синхронные

Регистры

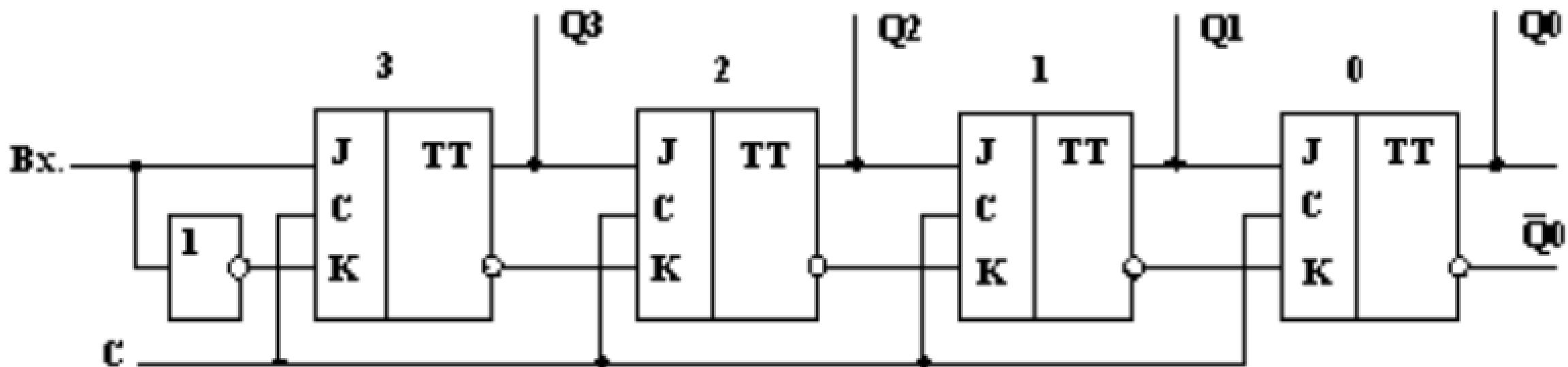
- Различают **регистры памяти** и **регистры сдвига**.
- **Регистры памяти** предназначены для временного хранения информации (данных).
- **Регистры сдвига** (сдвиговые регистры) предназначены как для временного хранения информации, так и для ее сдвига в сторону младших разрядов (вправо), так и в сторону старших разрядов (влево) без изменения информации. Кроме того, регистры сдвига могут преобразовывать информацию из параллельной формы в последовательную и наоборот.
- **Параллельный код (форма)** – информация (двоичное число), появляющаяся на всех разрядах регистра одновременно.
- **Последовательный код (форма)** – информация (двоичное число), появляющаяся на одном разряде регистра последовательно с каждым тактовым импульсом.

Задачи решаемые регистрами в цифровых схемах

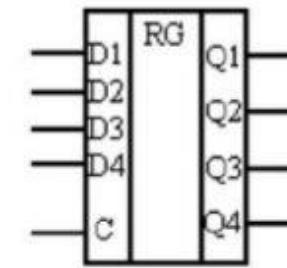
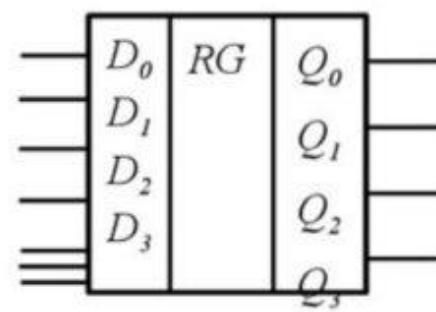
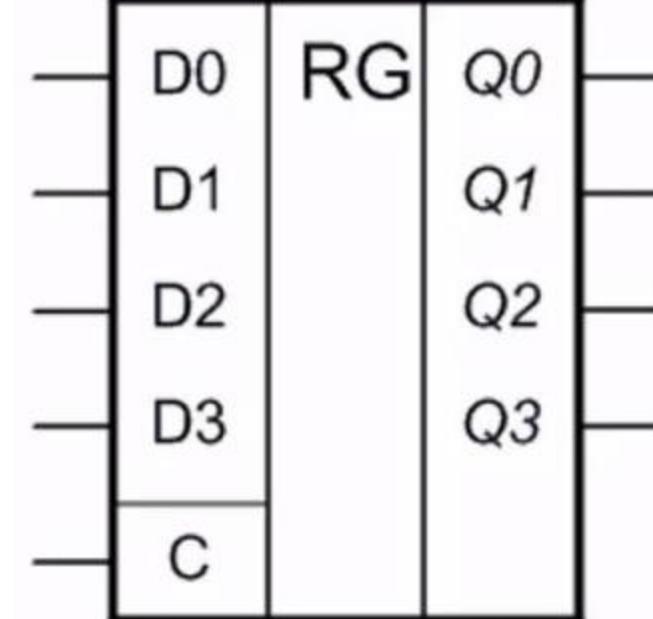
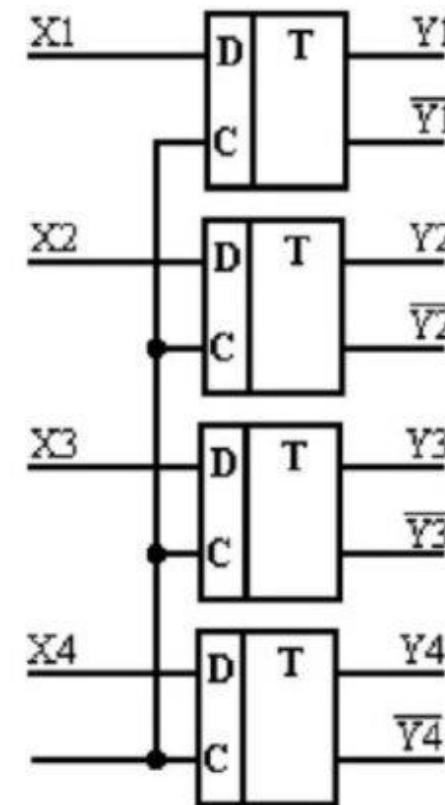
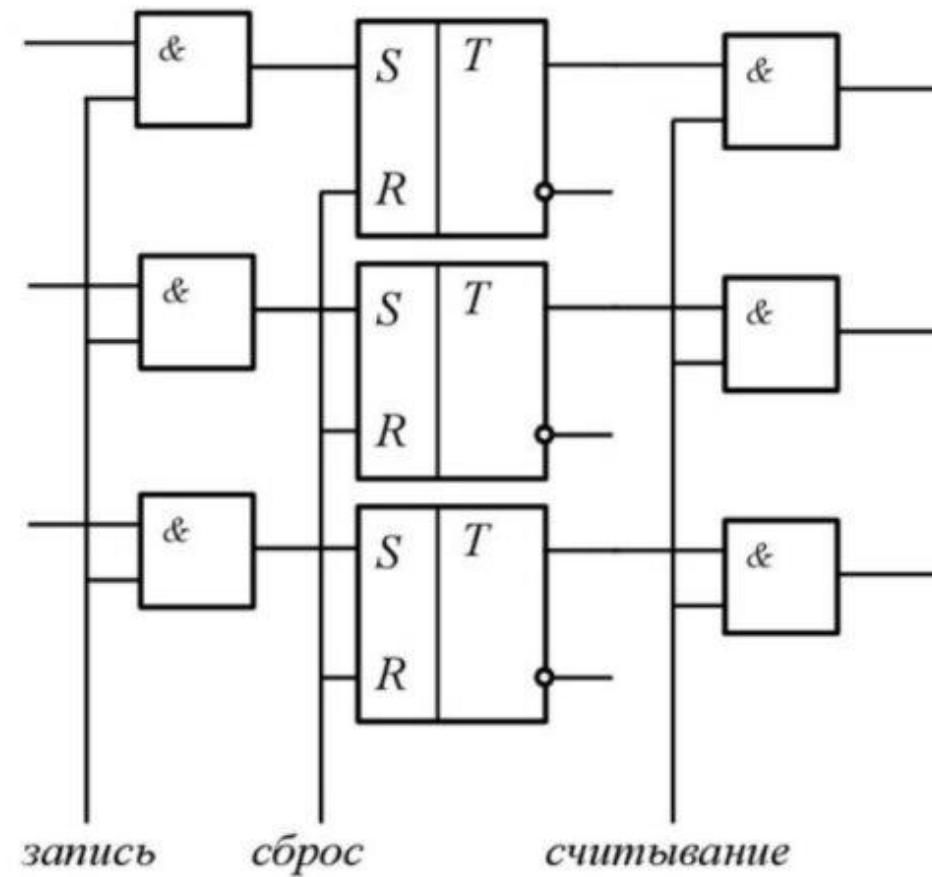
- **Основные функции регистров в цифровых схемах включают:**
- **Хранение данных:** Регистры используются для временного хранения данных в процессе обработки. Это особенно важно для систем с ограниченной памятью, таких как микроконтроллеры или процессоры.
- **Перемещение данных:** Регистры могут быть использованы для передачи данных между различными частями цифровой схемы. Они могут работать в качестве промежуточных звеньев, соединяя блоки, например, в процессорах.
- **Управление данными:** В некоторых системах регистры также могут использоваться для управления потоком данных или для реализации различных типов счетчиков и таймеров.
- **Сдвиг данных:** Специальные регистры, называемые сдвиговыми, могут выполнять операции сдвига, такие как сдвиг влево или вправо, что полезно для множества математических операций, включая умножение и деление на степени двойки.
- **Обработка команд:** В процессорах регистры часто хранят команды программы, а также различные флаги состояния, которые управляют выполнением инструкций.

Регистр

- Регистры сдвига широко применяются для преобразования последовательного кода в параллельный или параллельного в последовательный, а также при построении арифметико-логических устройств.
- Если 0 и 1 в регистре трактовать как двоичную запись числа, то сдвиг в одну сторону соответствует делению на 2, а в другую - умножению на 2.

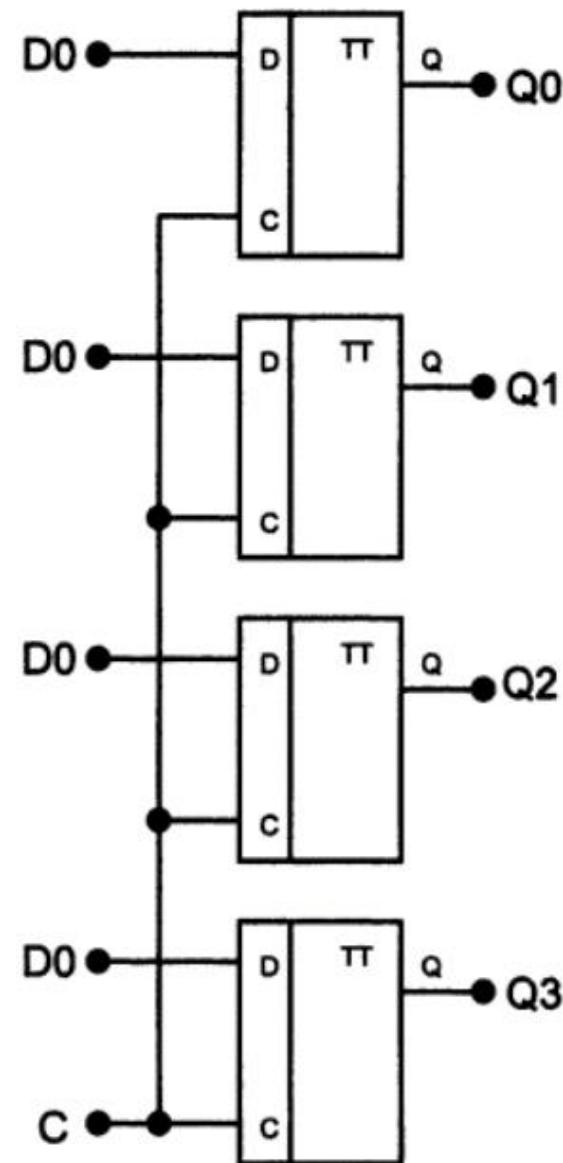
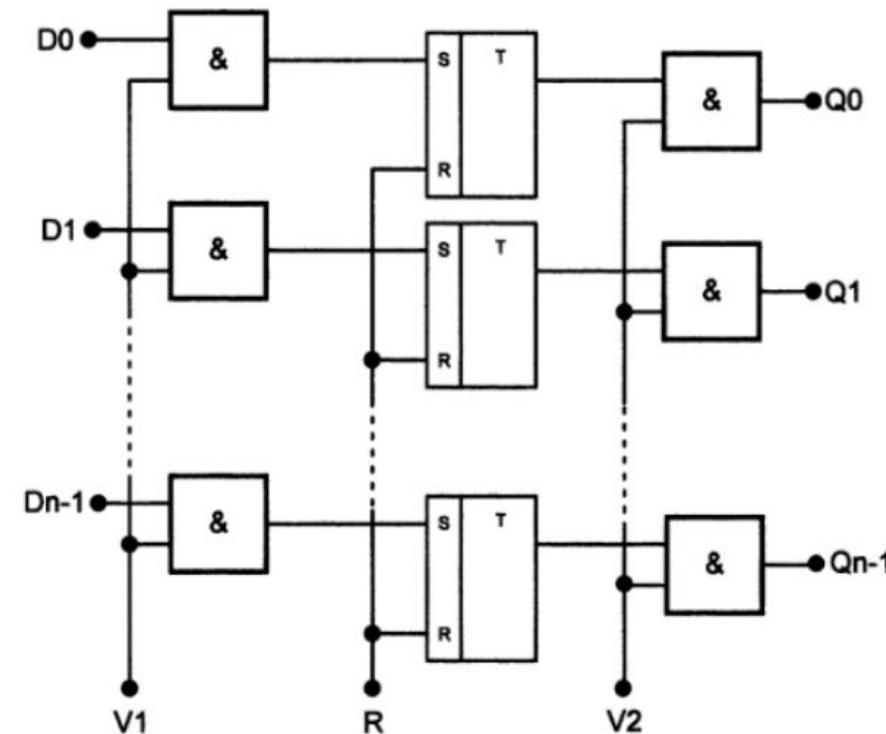
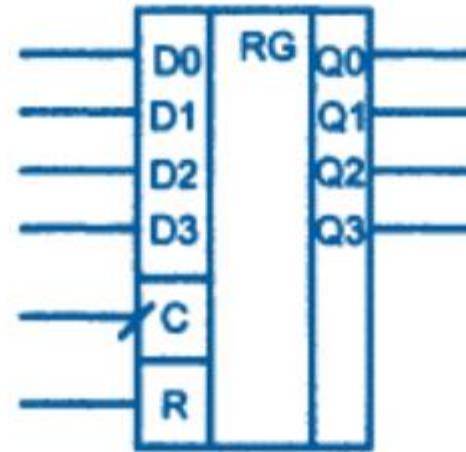


Регистры



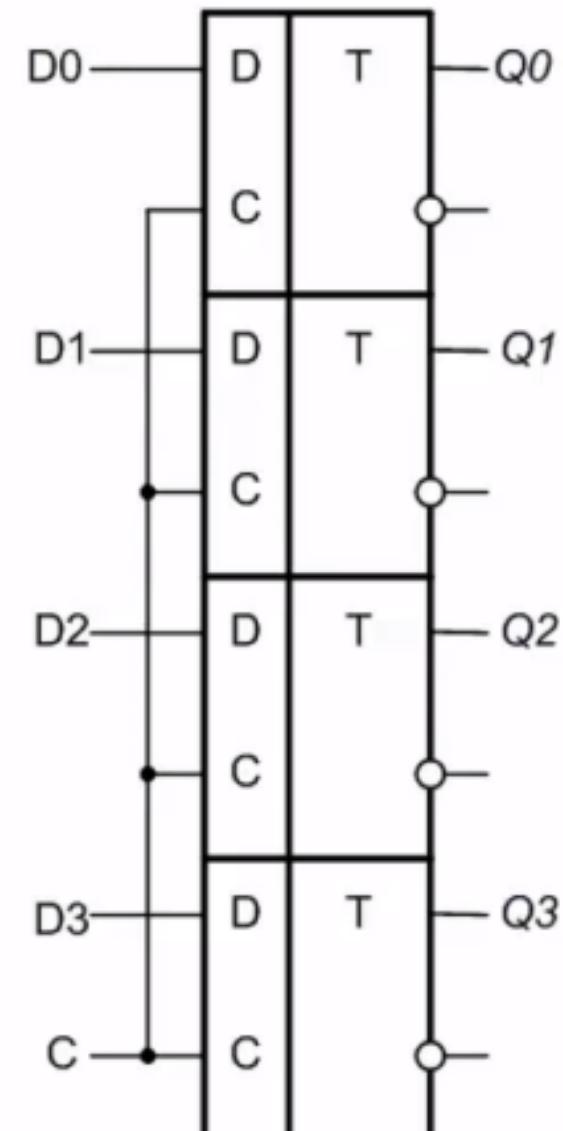
Регистры памяти (параллельные регистры)

- В параллельном регистре имеется общий тактовый вход С, причем все триггеры соединены параллельно и каждый из них имеет отдельный вход D и отдельный выход Q, которые независимы от других триггеров.
- Запись в асинхронный регистр производится подачей разрядов данных $D_0 - D_{n-1}$ на S-входы асинхронной установки триггеров по сигналу разрешения приема V_1 . Выдача данных на выходы Q производится по сигналу V_2 . Обнуление регистра производится по сигналу R.

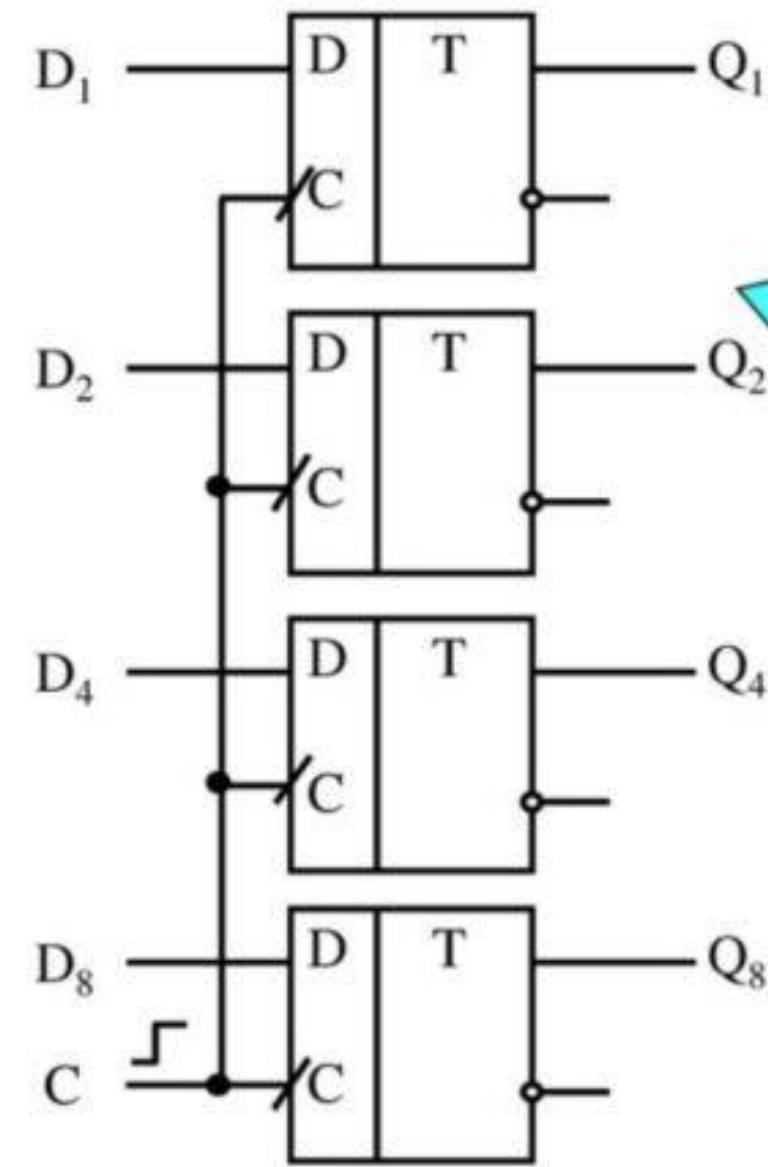
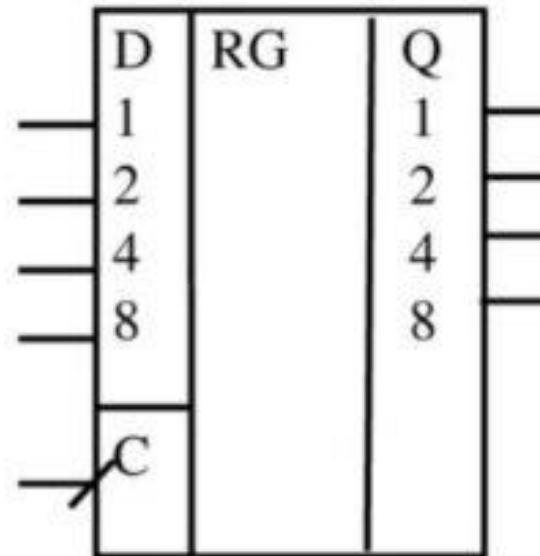


Регистры памяти (параллельные регистры)

- **Регистры памяти осуществляют:** прием (запись), хранение и передачу информации. Параллельными регистры называют из-за того, что запись двоичного числа осуществляется во все разряды регистра одновременно, т.е. в параллельном коде. Все операции выполняются также в параллельном коде, поэтому разряды регистра между собой не связаны.
- **Параллельный регистр,** точнее, регистр с параллельной загрузкой и чтением, строится из триггеров, не обменивающихся данными друг с другом, но имеющих общий сигнал синхронизации и другие сигналы управления *.
- * фактически выполняет роль D-триггера для многоразрядных слов!
- **Применение:** хранение двоичных данных; инициализация; синхронизация

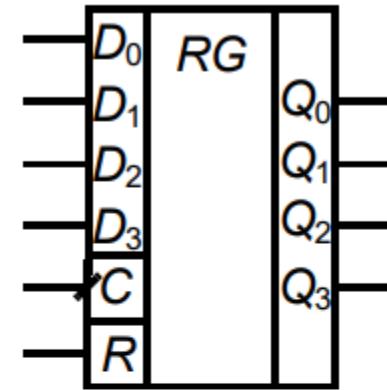
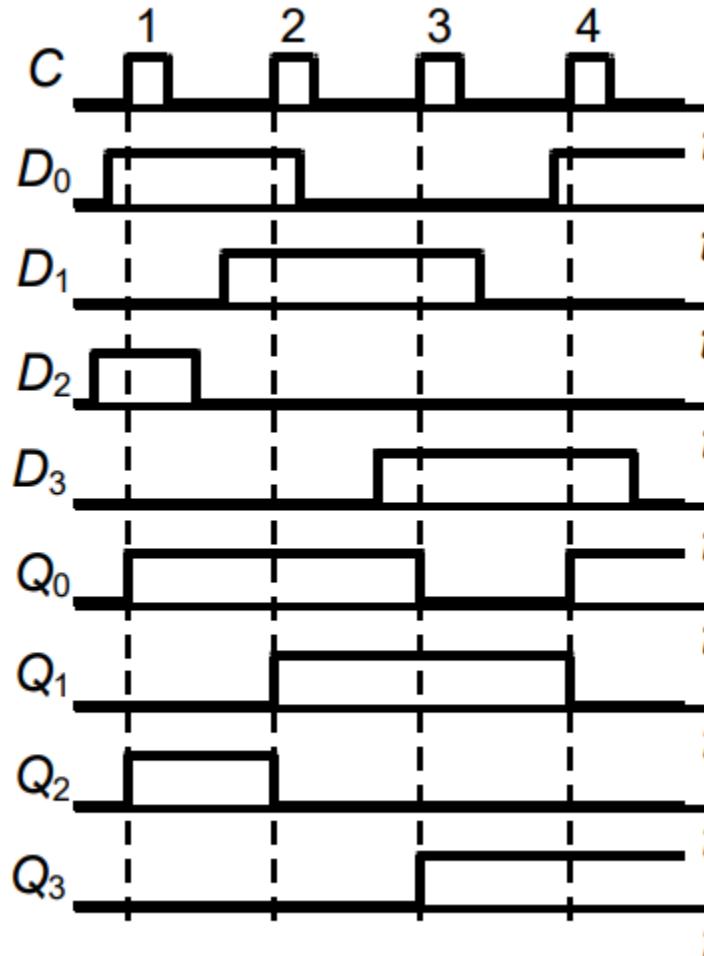
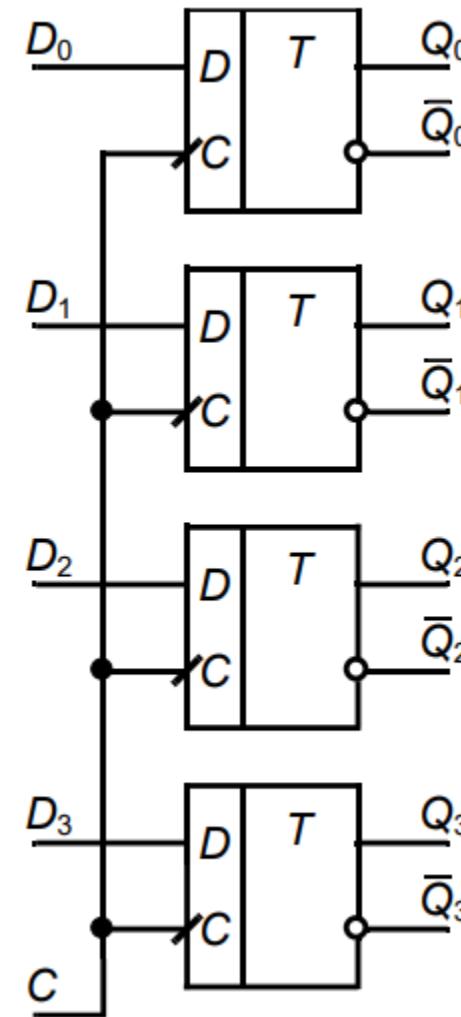


Регистры памяти (параллельные регистры)



все разряды
числа
подаются на
информацион-
ные входы
регистра
одновременно
- параллельно

Регистры памяти (параллельные регистры)



УГО регистра памяти

Запись информации в триггеры регистра может происходить по переднему и заднему фронту, а также по уровню.

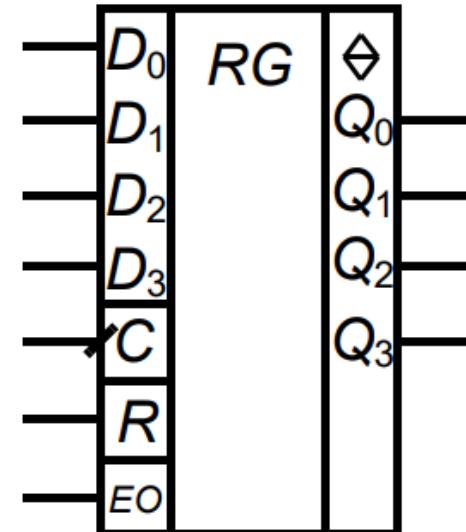
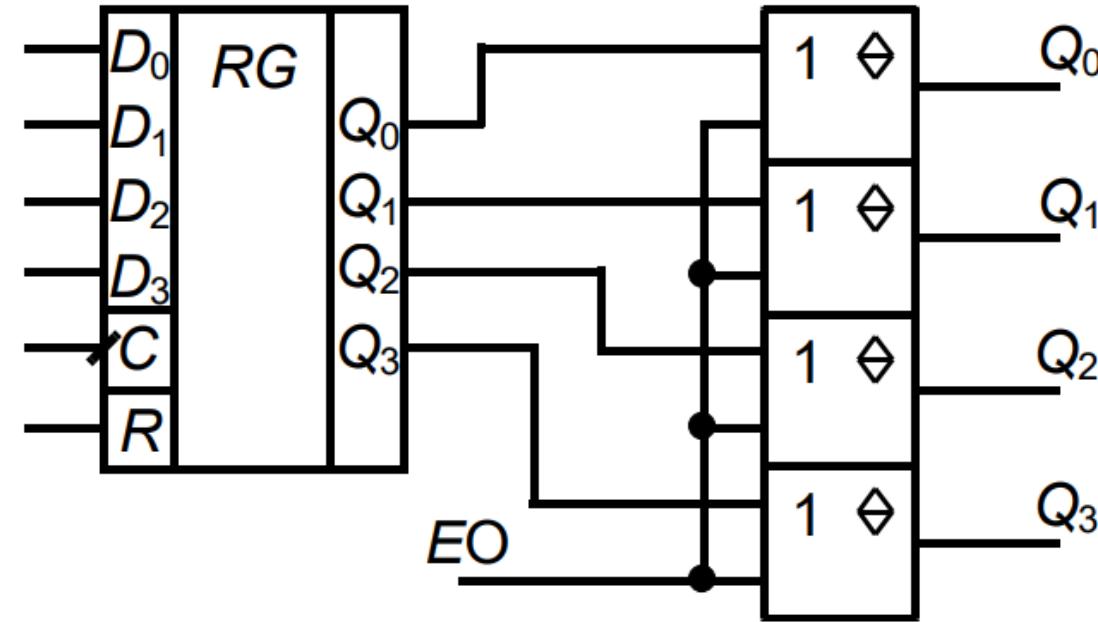
Часто регистры снабжаются входами асинхронного сброса R.

Регистры, работающие по фронту называются **регистрами с динамическим тактовым входом**.

Регистры работающие по уровню – **со статическим тактовым входом**.

Регистры памяти с тремя состояниями на выходе

Такие регистры образуются добавлением к регистру элементов с тремя состояниями

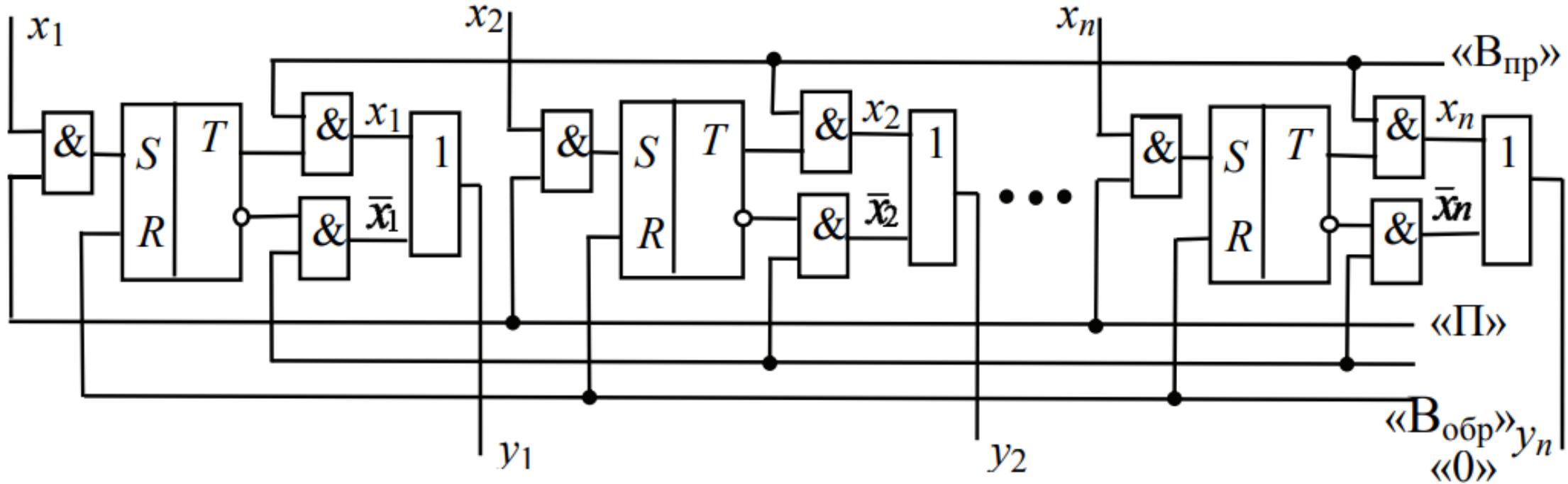


Регистры памяти выпускаются 4-х, 8-х и 16-ти разрядными.

УГО регистра памяти с тремя состояниями

Назначение: используются в микропроцессорных системах при работе на общую нагрузку (шина адреса, шина данных).

Параллельный регистр на RS-триггерах

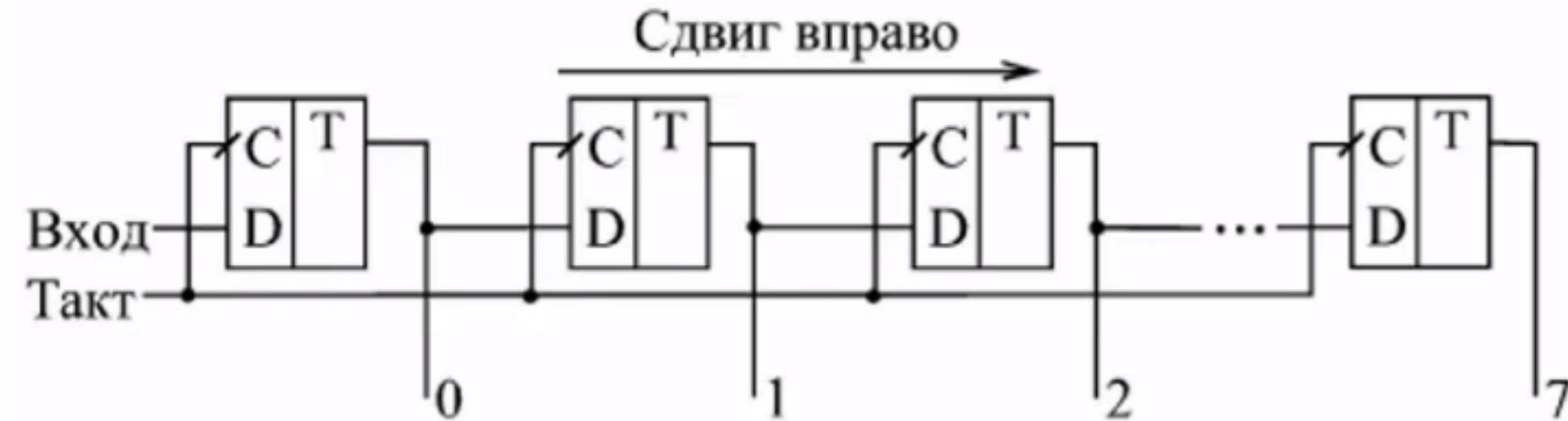


Работа параллельного регистра на RS-триггерах. Ввод (запись) числа осуществляется в два такта. Во избежание ошибочной записи числа $x_1x_2\dots x_n$ в первом такте все триггеры регистра обнуляются. Для этого на шину «0» подается логический 0.

Во втором такте по сигналу 1 на шине « Π » («Прием») через конъюнкторы одновременно записывается в соответствующие разряды регистра двоичное число $x_1x_2\dots x_n$. Вывод (считывание) числа $y_1y_2\dots y_n$ в прямом коде происходит по сигналу 1 на шине « $B_{пр}$ », а в обратном – по сигналу 1 на шине « $B_{обр}$ ».

Последовательный (сдвиговый) регистр

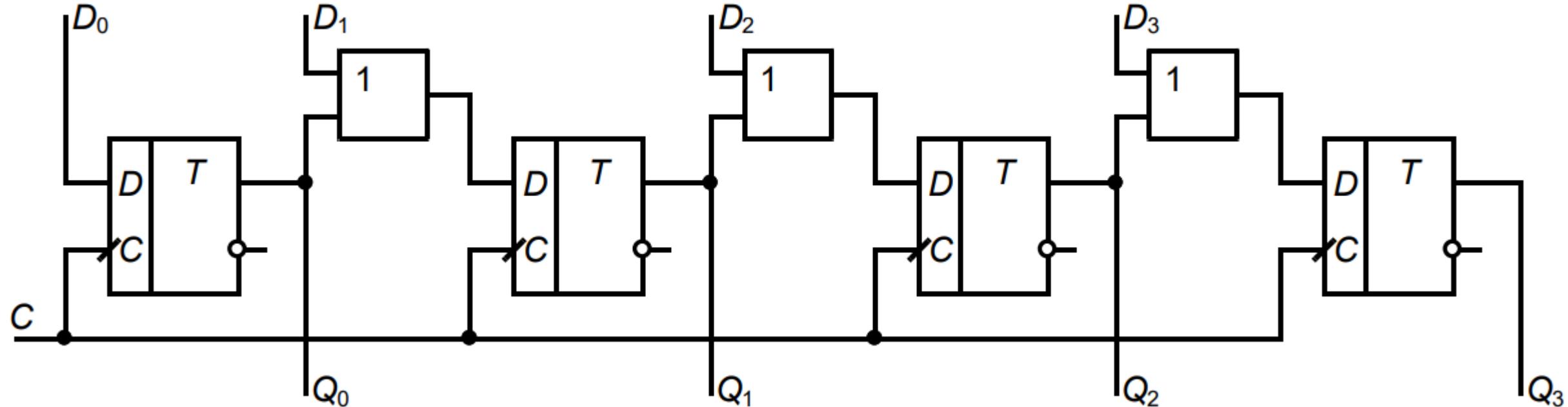
- **Последовательный (сдвиговый) регистр** представляет собой цепочку триггеров, связанных цепями переноса.
- На каждом такте синхросигнала выход одного триггера будет записываться в последующий триггер, и так далее.



- **Применение:**
 - ввод/вывод
 - арифметические операции
 - генерация сигналов

Регистры сдвига (последовательные регистры)

Для создания регистров триггеры соединяются последовательно

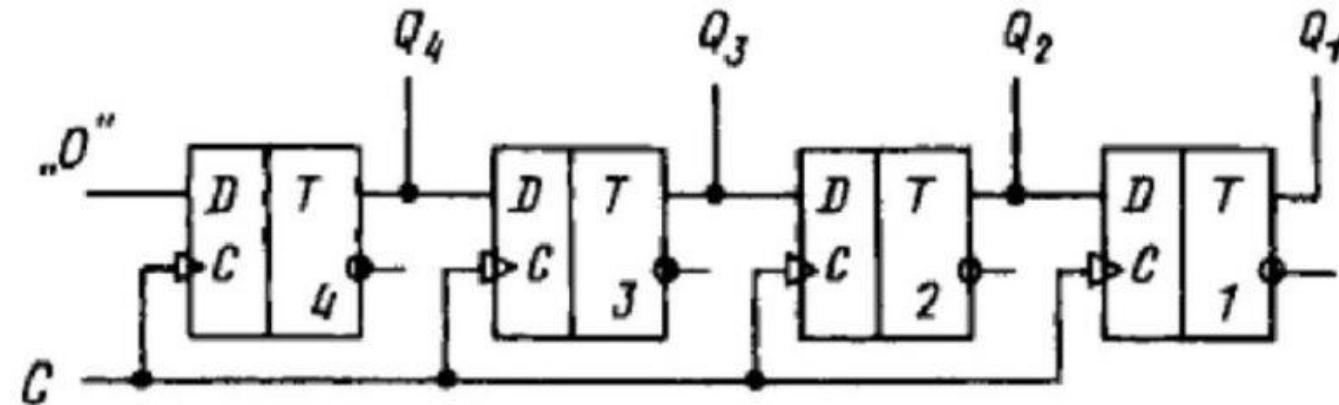


Для установки в исходное состояние (запись нулевого кода) триггеры снабжаются входом сброса R (на схеме не показан).

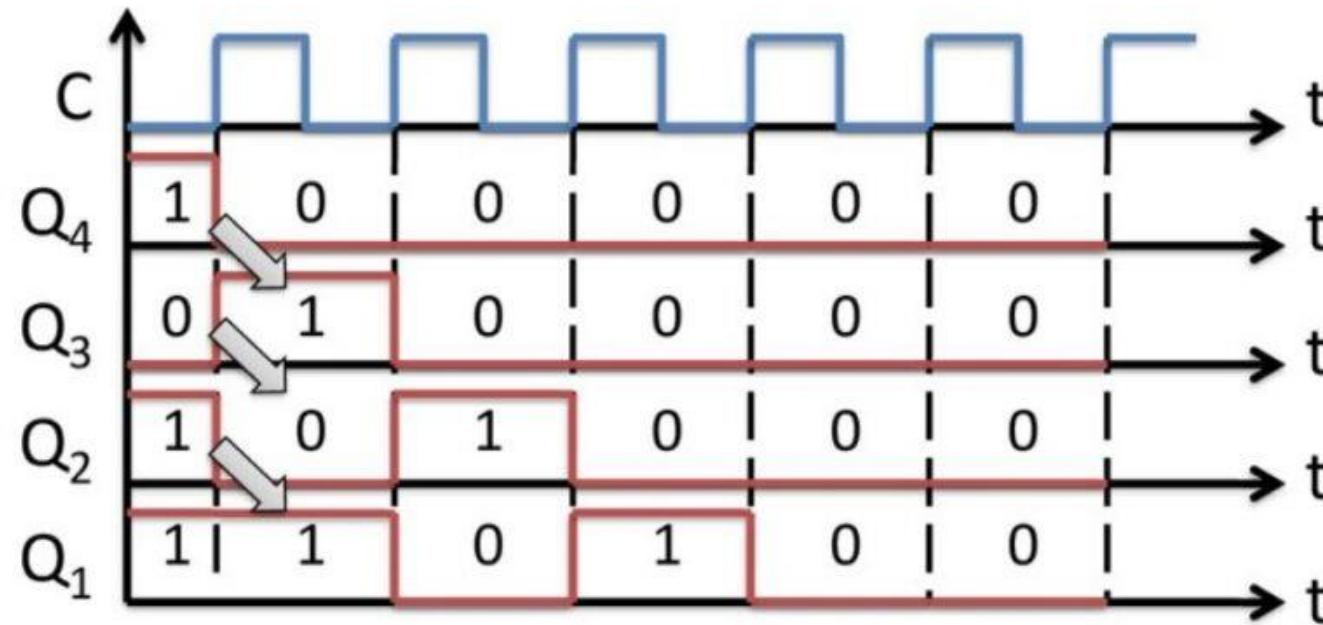
Данный регистр представляет собой последовательно соединенные двоичные ячейки памяти (триггеры). Под действием тактовых импульсов состояния ячеек памяти сдвигаются (передаются) на последующие ячейки.

Схема регистра позволяет преобразовать информацию из последовательного в параллельный код и наоборот.

Последовательный (сдвиговый) регистр



Временные диаграммы



Сдвиговый регистр — это тип цифрового регистра, в котором данные могут передаваться по цепочке триггеров, сдвигаясь на одну или несколько позиций за каждый такт.

Это устройство позволяет манипулировать данными с помощью сдвига битов влево или вправо.

Сдвиговые регистры часто используются в операциях умножения и деления на степени двойки, а также в различных алгоритмах обработки сигналов.

Принцип работы сдвигового регистра

Сдвиговый регистр состоит из последовательности триггеров, каждый из которых хранит один бит данных. Данные, поступающие в первый триггер, передаются по цепочке на следующие триггеры в зависимости от направления сдвига.

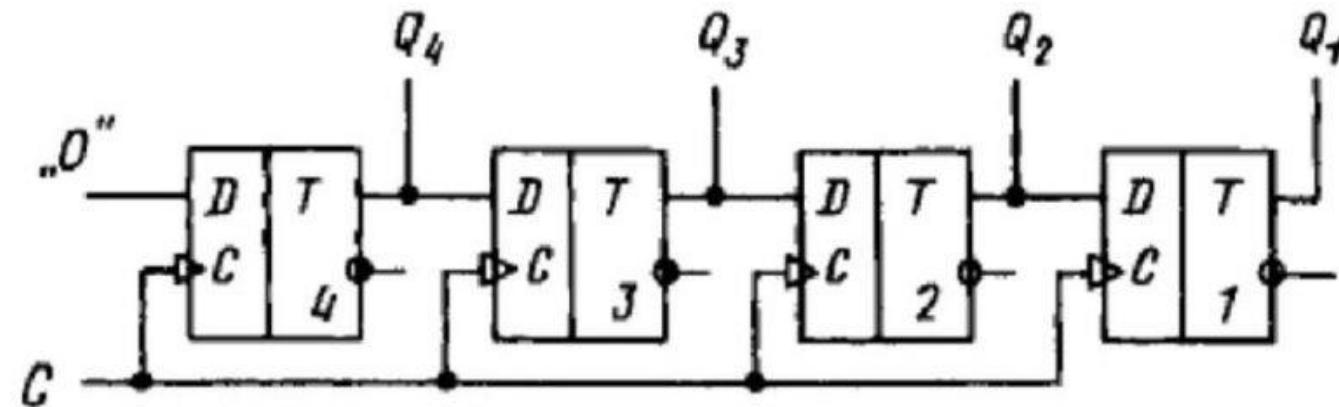
При каждом такте сдвигового регистра данные «сдвигаются» влево или вправо.

Сдвиг влево: Биты данных перемещаются влево, и пустое место на выходе заполняется значением ноль или другим значением, заданным схемой.

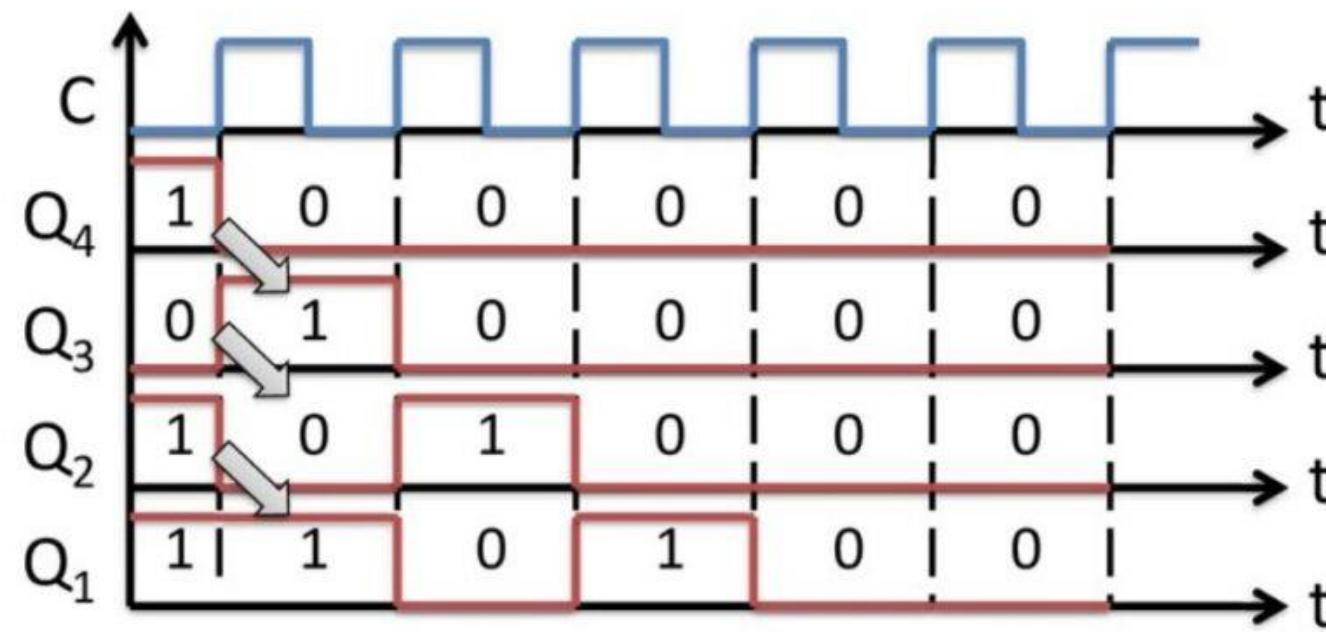
Сдвиг вправо: Биты данных перемещаются вправо, и пустое место на входе может быть заполнено нулем или другим значением.

Сдвиговый регистр может быть использован для последовательной передачи данных, а также для выполнения операций сдвига для различных алгоритмов.

Последовательный (сдвиговый) регистр



Временные диаграммы



Виды сдвиговых регистров: сдвиг влево и вправо

Сдвиговые регистры можно классифицировать по направлению сдвига:

Сдвиг влево: Каждый бит данных перемещается на одну позицию влево. Например, если регистр хранит 4 бита данных 1101, то после одного сдвига влево оно превратится в 1010.

Сдвиг вправо: Каждый бит данных перемещается на одну позицию вправо. Например, для данных 1101 сдвиг вправо приведет к 0110.

Кроме того, существуют сдвиговые регистры с возможностью сдвига в обе стороны — **двунаправленные регистры**.

Сдвиговые регистры применяются в различных цифровых устройствах:

- **Операции сдвига:** В математических операциях, например, при умножении или делении на степени двойки.

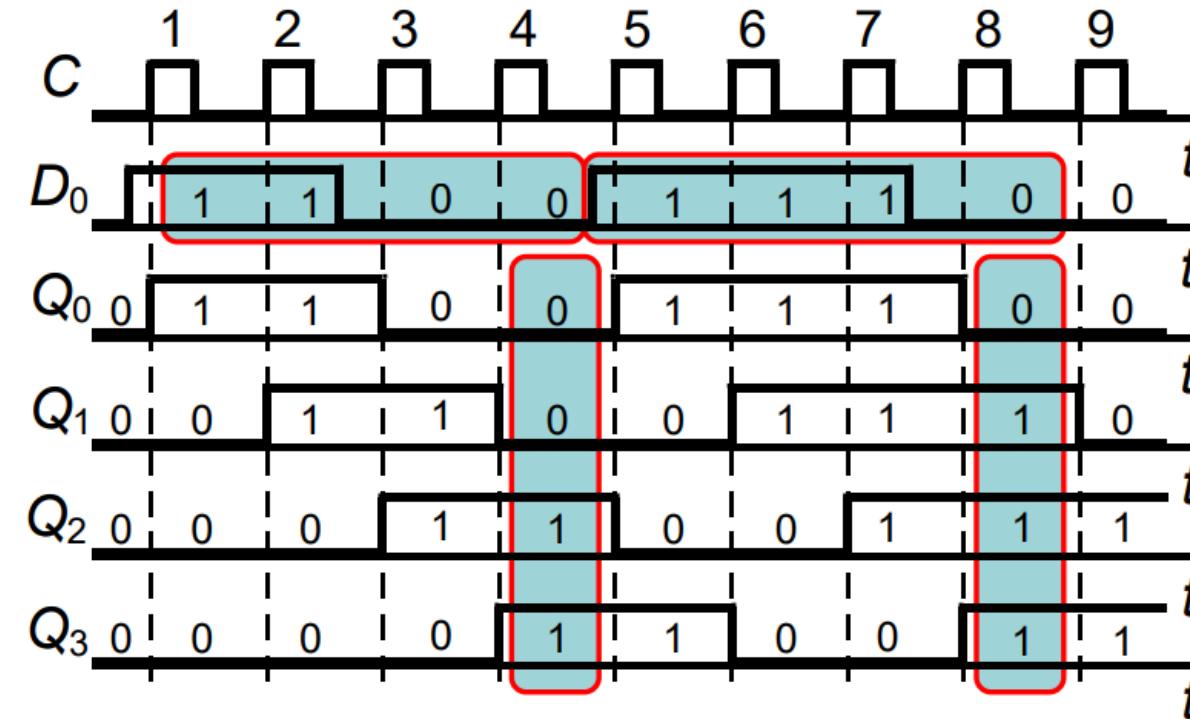
- **Обработка данных:** В цифровых сигнальных процессорах (DSP) сдвиговые регистры используются для обработки аудио и видеосигналов, фильтрации и других алгоритмов.

- **Цифровая передача:** Сдвиговые регистры используются для последовательной передачи данных через интерфейсы типа SPI или UART.

- **Реализация алгоритмов:** В некоторых криптографических алгоритмах, а также в генераторах случайных чисел.

Регистры сдвига

1. Преобразование информации из последовательного кода в параллельный

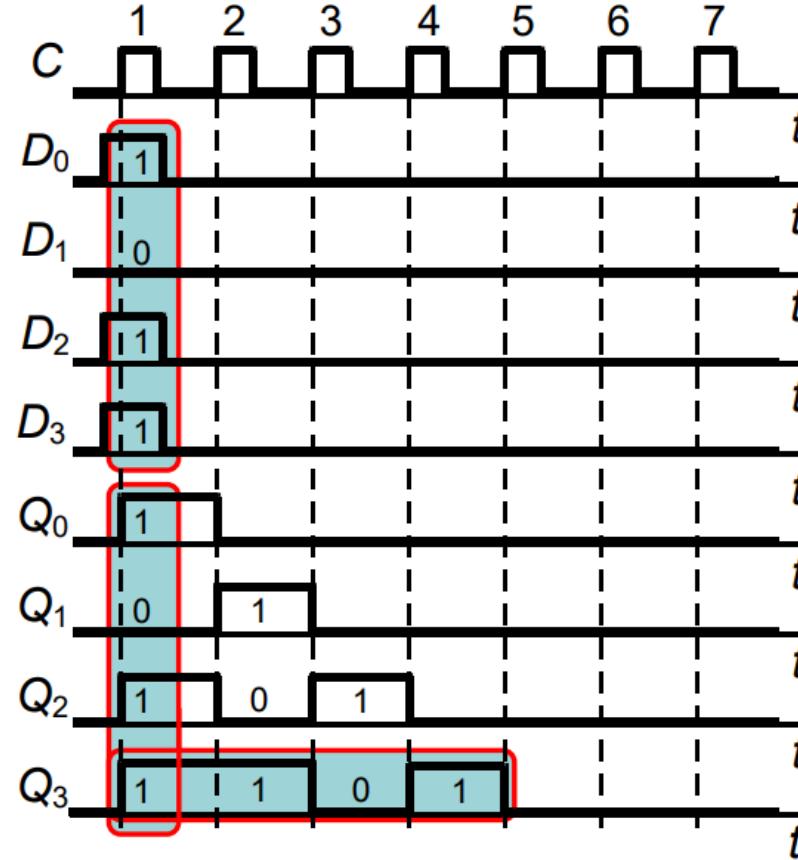


Вход D_0 используется в качестве информационного. На вход D_0 поступает информация в последовательном коде, т.е. данные вводятся в регистр последовательно (поразрядно). D -триггер – триггер с задержкой (передает информационный сигнал с D -входа на выход Q с задержкой в один такт).

Из диаграмм видно, что данные со входа D_0 последовательно передаются (сдвигаются) с каждым тактовым импульсом. За 4 такта двоичное число появляется на выходах Q_0 - Q_3 в параллельном коде.

Регистры сдвига

2. Преобразование информации из параллельного кода в последовательный



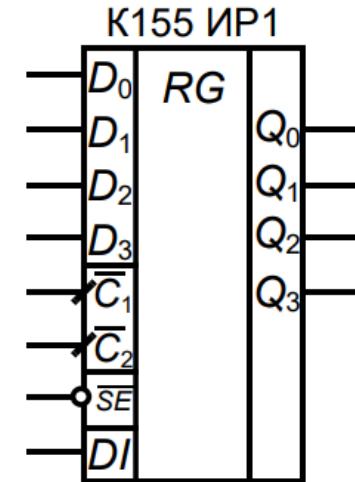
SE – выбор режима (память/сдвиг): параллельная или последовательная загрузка;

C1 – тактовый вход регистра сдвига;

C2 – тактовый вход регистра памяти.

В первом такте информация на D -входах записывается в триггеры и с каждым последующим тактовым импульсом сдвигается влево (в сторону старших разрядов). На выходе Q_3 через 4 такта получаем информацию в последовательном коде.

Существуют универсальные регистры, выполняющие все описанные ранее операции.



Регистры сдвига

Математические операции умножения и деления

С помощью регистров можно выполнять операции умножения и деления на 2.

Умножение хранящегося в регистре числа на 2 выполняется путем его сдвига влево и записью «0» в младший разряд.

Пример.

$00011_2 = 3_{10}$ - пусть задано данное число

$00110_2 = 6_{10}$ - сдвигаем влево (умножаем на 2)

$01100_2 = 12_{10}$ - сдвигаем влево (умножаем на 2)

Таким образом умножили число 3 на 4 и получили 12.

Деление на 2 осуществляется сдвигом хранящегося в регистре числа вправо и записью «0» в старший разряд, причем деление на 2 **целочисленное**.

Пример.

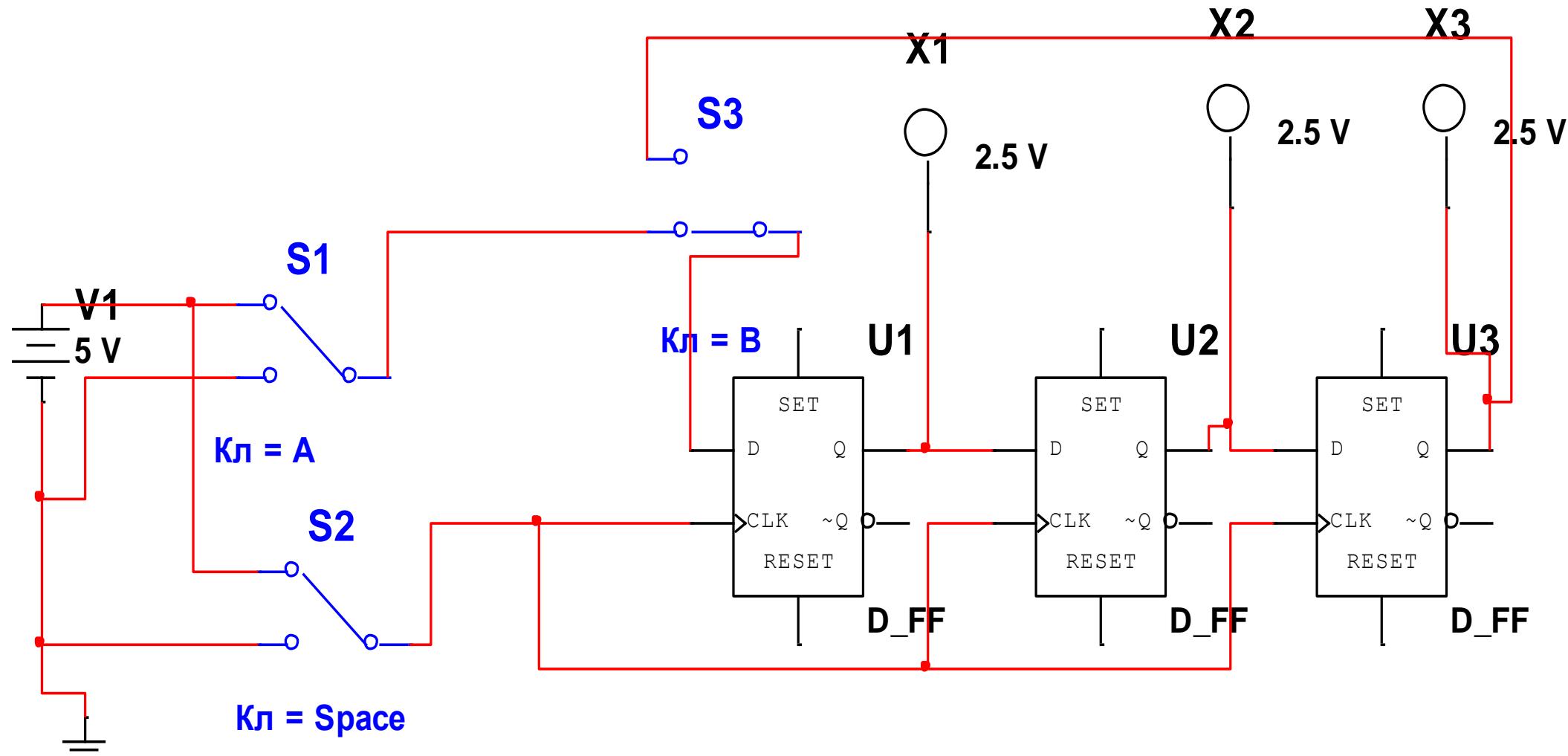
$01110_2 = 14_{10}$ - пусть задано число 14

$00111_2 = 7_{10}$ - сдвигаем вправо (делим на 2)

$00011_2 = 3_{10}$ - сдвигаем вправо (делим на 2)

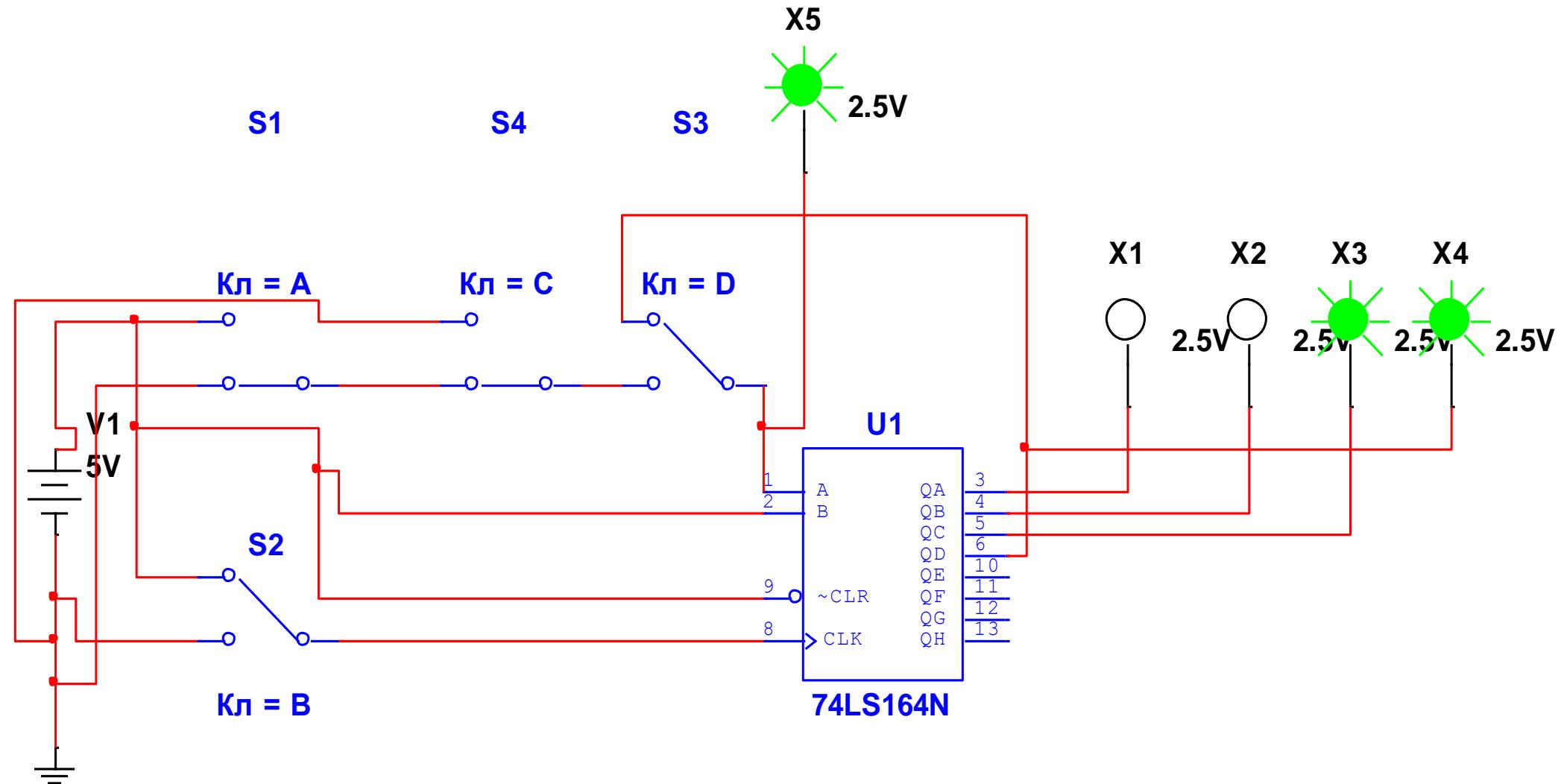
В итоге разделили число 14 на 4 и получили целое число 3.

Последовательный (сдвиговый) регистр



Сдвиговый последовательный регистр

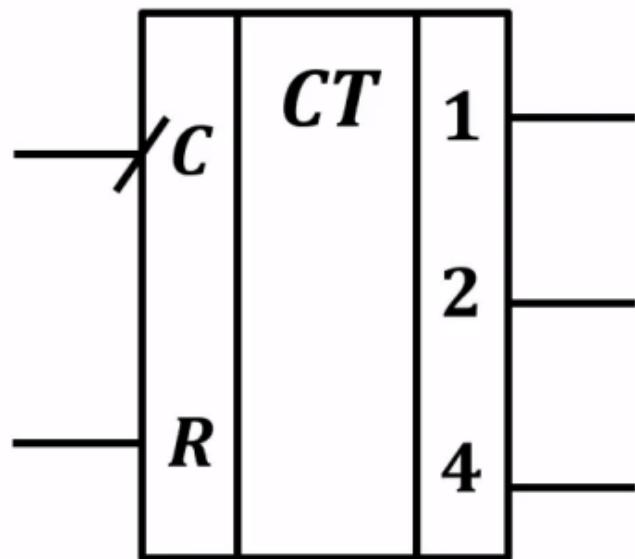
Последовательный (сдвиговый) регистр



Последовательный регистр на микросхеме

Применение регистров в вычислительной технике

- **Регистры являются ключевыми компонентами в архитектуре процессоров и микроконтроллеров.** Они выполняют роль быстрого хранения данных, которые необходимы для выполнения инструкций, и участвуют в обработке промежуточных результатов вычислений.
- **В процессорах регистры обеспечивают:**
 - **Хранение операндов** для выполнения арифметических и логических операций.
 - **Сохранение состояния** процессора, например, флагов состояния или указателей на инструкции.
 - **Передачу данных** между различными частями процессора, такими как арифметико-логическое устройство (АЛУ) и память.
- **В микроконтроллерах регистры** также выполняют важные функции, например, управление периферийными устройствами, настройку режимов работы и обработку данных с внешних устройств.



Последовательная логика Счетчики



Счетчик, Счетчик импульсов

- **Счетчик** – устройство, производящее подсчет числа импульсов, поступивших на его вход.
- **Счетчик** – функциональный узел для подсчёта импульсов, поданных на его вход.
- **Счетчики импульсов построены из триггеров.**
- Могут строиться на двухступенчатых D-триггерах, T-триггерах и JK-триггерах.
- **Основной параметр счётчика** – модуль счёта – максимальное число единичных сигналов, которое может быть сосчитано счётчиком. Счётчики обозначают через CT (от англ. counter).



Счетчик

- **Информационная емкость счетчика** - коэффициент счёта ($K_{сч}$) – это число в двоичном коде, до которого можно производить счёт (максимальное значение.)

$$K_{сч} = 2^n, \text{ где } n \text{ – число триггеров.}$$

- Если $K_{сч} \neq 2^n$, то счётчик имеет произвольный коэффициент счёта.
- Счетчики одновременно являются и делителями частоты в 2^n раз, где n – число разрядов (триггеров).
- **Максимальное количество импульсов** N , которое может зарегистрировать счетчик без его переполнения, для двоичных счетчиков равно $N = 2^n$, n – число разрядов счетчика. Это число называют модулем счета.
- **В случае переполнения счетчика** ($N > 2^n$) **счетчик обнуляется**, а счет циклически возобновляется.
- **Простейшие счетчики импульсов состоят из цепочки последовательно включенных Т-триггеров**, каждый из которых делит частоту входного сигнала на два. Каждый из триггеров этой цепочки называют разрядом счетчика.

Счетчики. Классификация

Счётчики классифицируют:

- **по числу устойчивых состояний триггеров**
 - на двоичных триггерах
 - на троичных триггерах
 - на n -ичных триггерах
- **по модулю счёта:**
 - двоично-десятичные;
 - двоичные;
 - с произвольным постоянным модулем счёта;
 - с переменным модулем счёта;
- **по выполнению специальных операций:**
 - двоично-десятичные счётчики
 - цифровые частотометры
 - кольцевые счётчики
 - счётчики Джонсона



Счетчики. Классификация

Счётчики классифицируют:

- **по способу формирования внутренних связей:**

- с последовательным переносом;
- с ускоренным переносом;
 - с параллельным ускоренным переносом;
 - со сквозным ускоренным переносом;
- с комбинированным переносом;
- кольцевые;

- **по принципу действия** (По способу переключения триггеров):

- **1. асинхронные (последовательные)** – каждый последующий триггер срабатывает от предыдущего, а сигналы счёта подаются на первый триггер.
 - **2. синхронные** – тактовый сигнал подается сразу на все разряды, но срабатывание последующего триггера происходит после того, как единицами заполнены все младшие разряды.
-
- Признаки классификации независимы и могут встречаться в различных сочетаниях, скажем, асинхронный двоично-десятичный суммирующий счетчик или бинарный синхронный реверсивный счетчик.

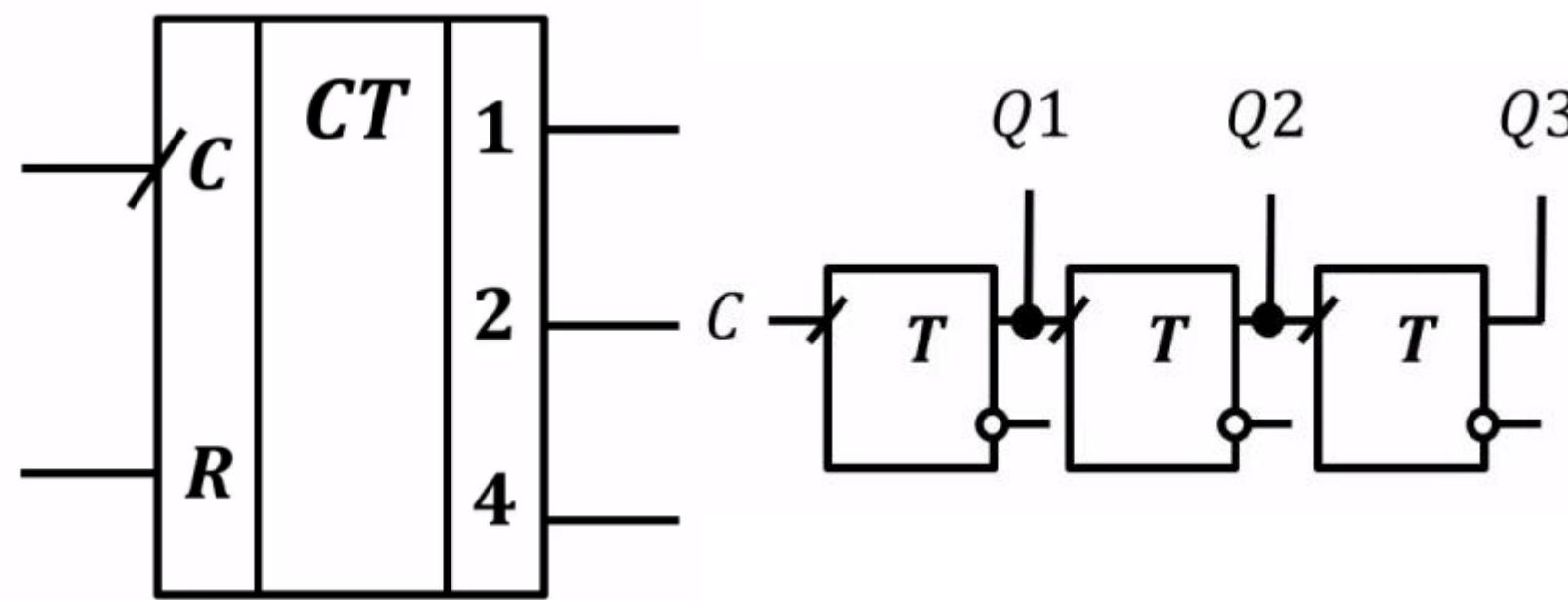


Счетчики. Классификация

- Классифицируют счетчики по следующим признакам: по коэффициенту пересчета K , по направлению счета, по способу управления счетом.
- По коэффициенту пересчета K различают:
 - бинарные счетчики, у которых $K = 2^n + 1$;
 - счетчики с произвольным коэффициентом пересчета ($K^1 \neq 2^n + 1$, $K = \text{const}$);
 - счетчики с произвольным $K^1 \neq 2^n + 1$, $K = \text{var.}$
- По направлению счета счетчики разделяют как:
 - суммирующие, у которых $S(0) < S(t)$, $t > 0$ (**суммирующий счетчик** - счетчик, который под действием счетного импульса переходит из состояния $s(t)$ в $(s(t) + 1) \bmod K$ состояние);
 - вычитающие, у которых $S(0) > S(t)$, $t > 0$ (**вычитающий счетчик** - счетчик, который под действием счетного импульса переходит из состояния $s(t)$ в $(s(t) - 1) \bmod K$ состояние);
 - реверсивные счетчики (цифровые устройства, способные работать как в режиме суммирующего, так и в режиме вычитающего счетчика).

Двоичный счетчик

- **Счётчик числа импульсов** — устройство, на выходах которого получается двоичный код, определяемый числом поступивших импульсов. Счётчики могут строиться на Т-триггерах, D-триггерах и JK-триггерах.
- **Основной параметр счётчика — модуль счёта** — максимальное число единичных сигналов, которое может быть сосчитано счётчиком. Счётчики обозначают через СТ (от англ, counter).

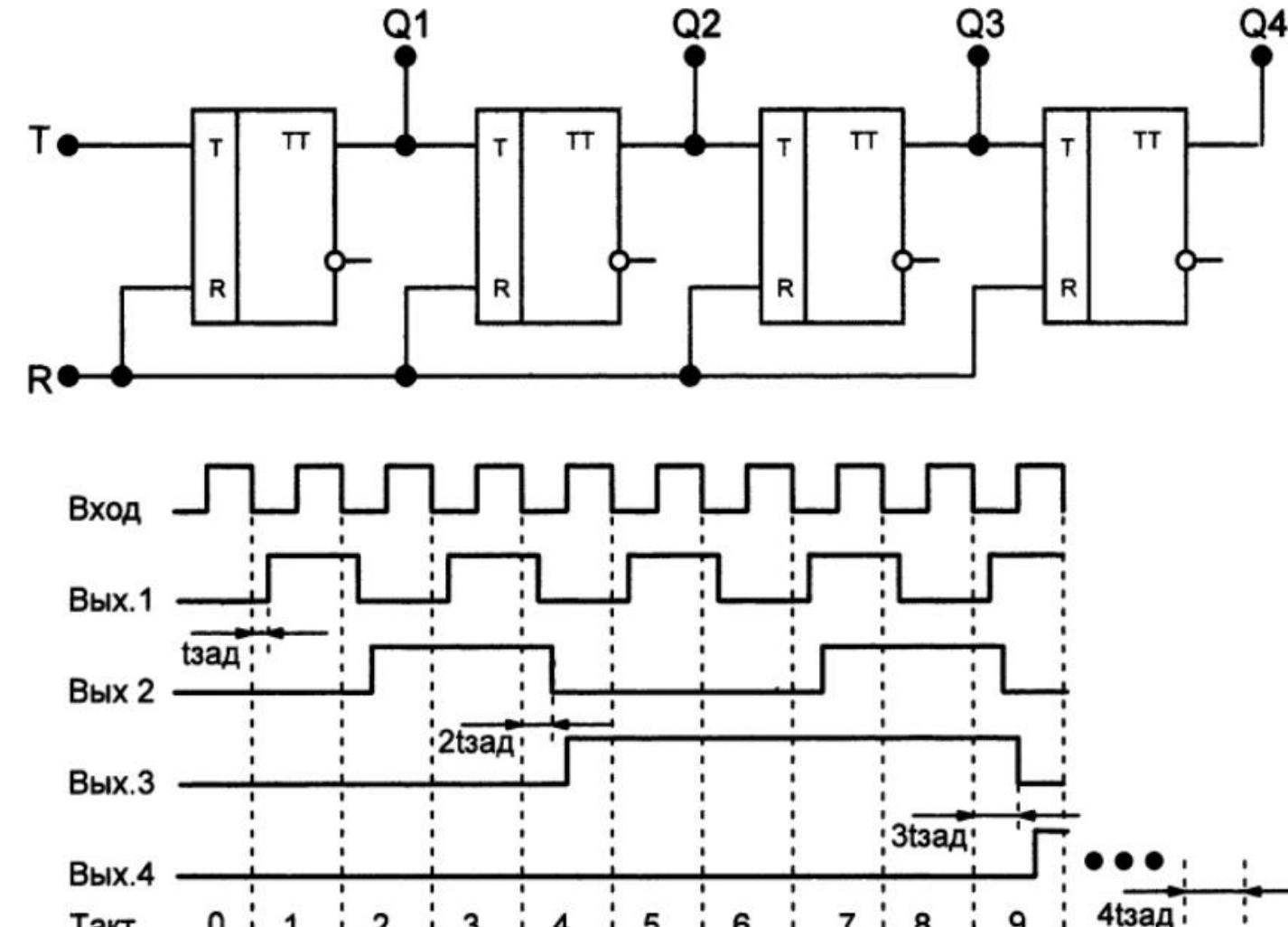
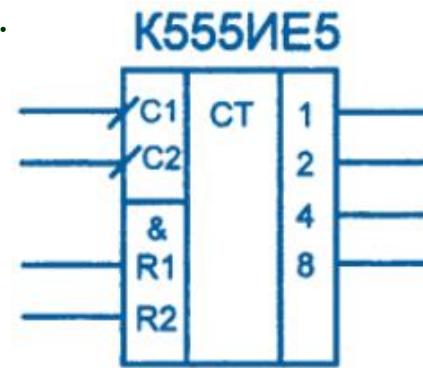


Применение:

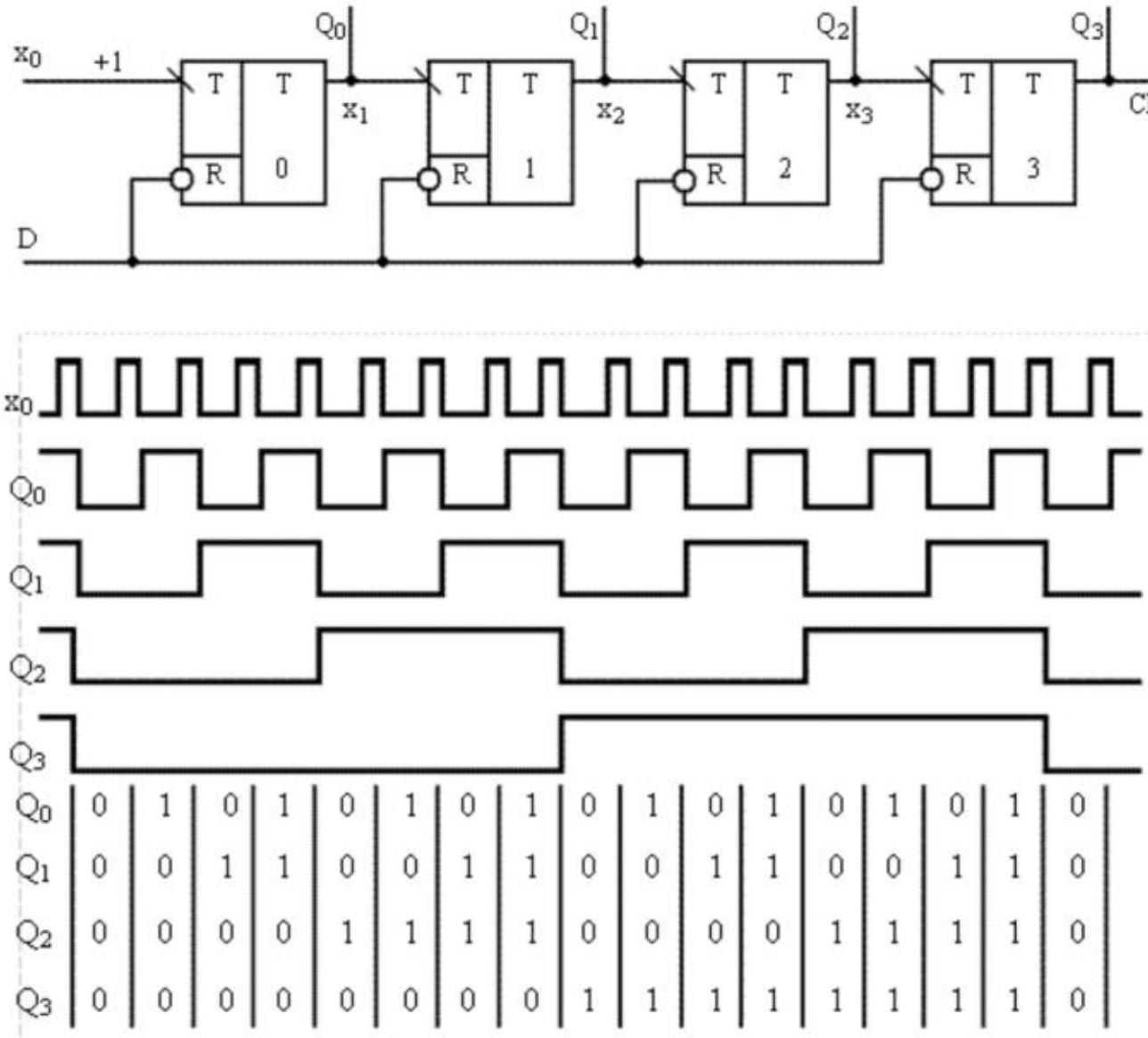
- деление частоты
- подсчёт импульсов

Асинхронный счётчик

- Все триггеры включаются последовательно, следовательно, и выходы счетчика также переключаются последовательно.
- Задержка переключения каждого разряда примерно равна задержке триггера.
- При периоде входного сигнала, меньшем полной задержки установления кода счетчика, правильный код на его выходе не успеет установиться.
- Это накладывает ограничения на максимальную частоту входного сигнала.



Счетчик



Суммирующий бинарный 4-разрядный счетчик с основанием $K = 2l+1$, временные диаграммы и таблицы состояний 4-разрядного счетчика

Асинхронный счётчик

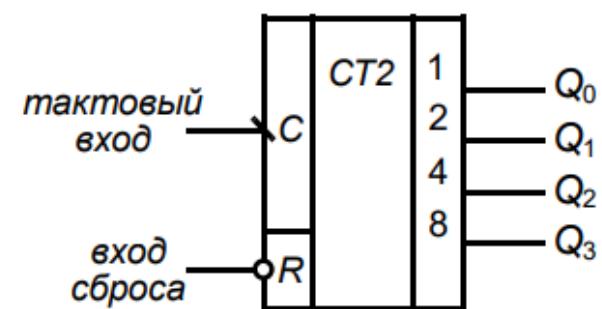
Дес. код	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

(или счётчик с последовательным или сквозным переносом)

Из таблицы:

- значение Q_i изменяется только тогда, когда в младшем разряде происходит переход сигнала с «1» на «0» (счётчики последовательного типа);
- значение Q_i изменяется, когда предыдущие младшие разряды заполнены единицами (счётчики параллельного типа).

Для организации счёта в обратном направлении необходимо все выходы Q соединить с тактовыми входами C . В этом случае с приходом тактового импульса информация будет убывать. Диаграммы в счётчике обратного счёта получаются инверсией диаграмм выходов Q (кроме C).

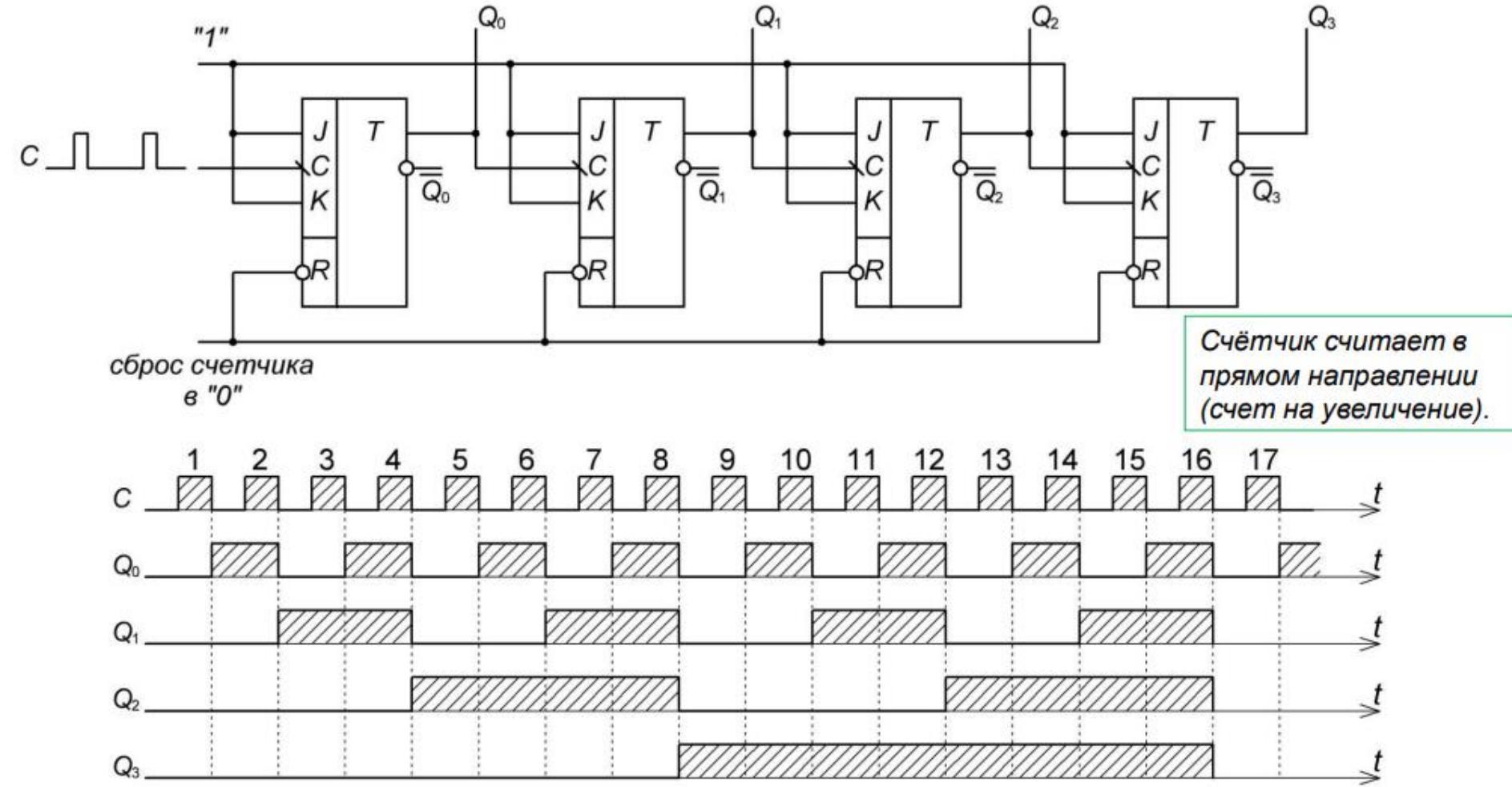


УГО счётчика импульсов

Обозначение:

СТ2 – двоичный счётчик;
СТ2/10, СТ10 – двоично-десятичный счётчик, считает в двоичном коде от 0 до 9, на десятом импульсе сбрасывается и считает заново.

Асинхронный счётчик

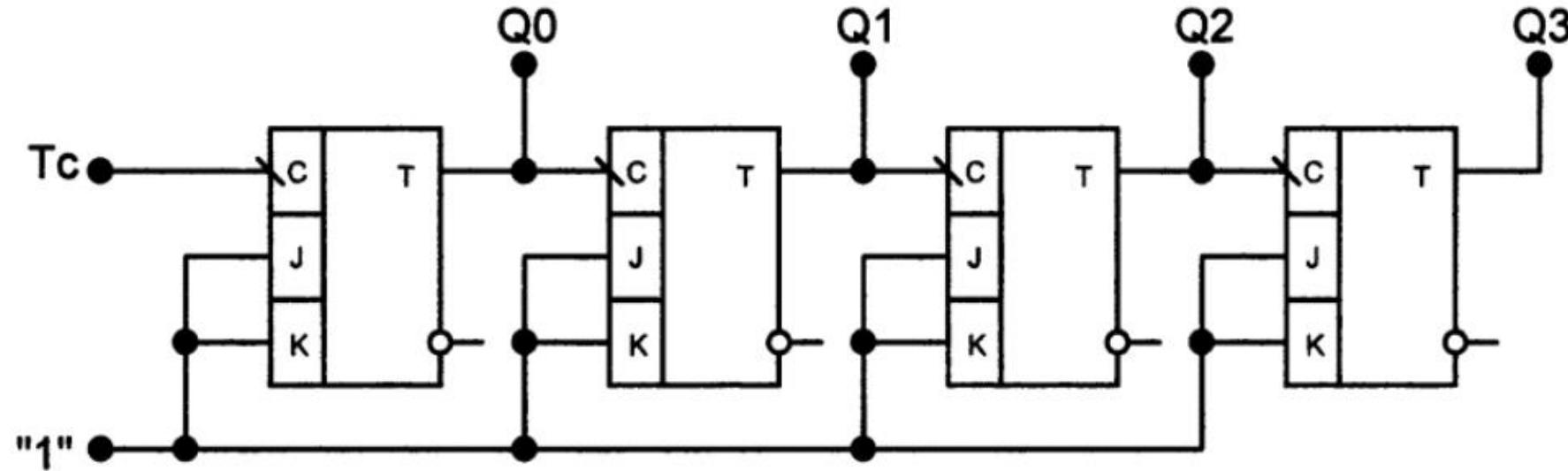


Счётчик считает в прямом направлении (счёт на увеличение).

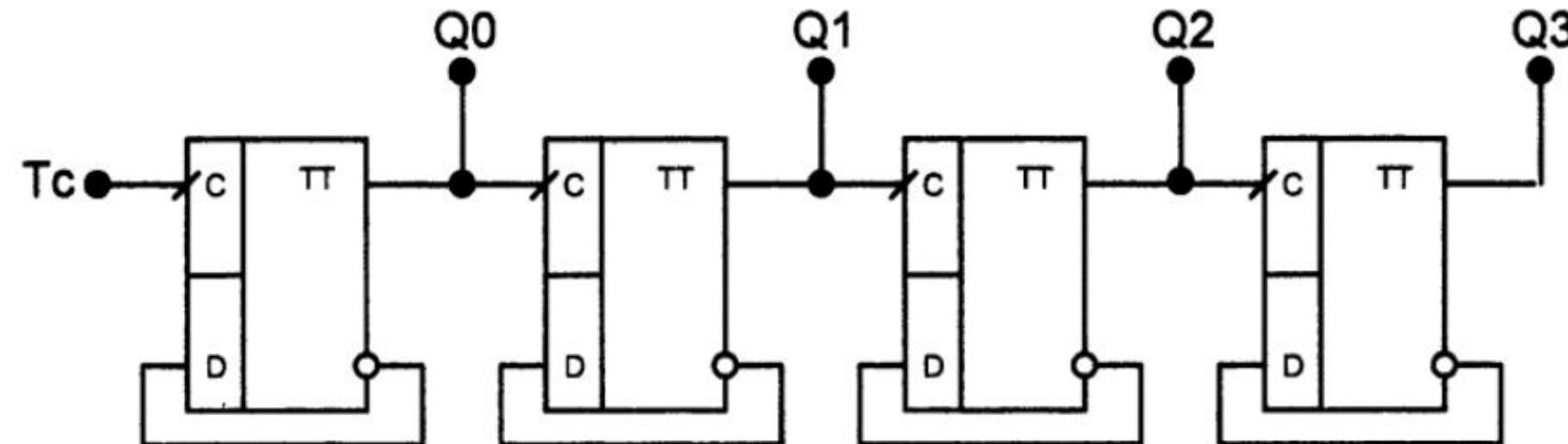
Счётчик является асинхронным, т.к. выходные сигналы появляются последовательно во времени (значение Q_i изменяется, когда в младшем разряде происходит переход сигнала с «1» на «0»).

Варианты асинхронных счётчиков

Асинхронный последовательный суммирующий счетчик на JK-триггерах.

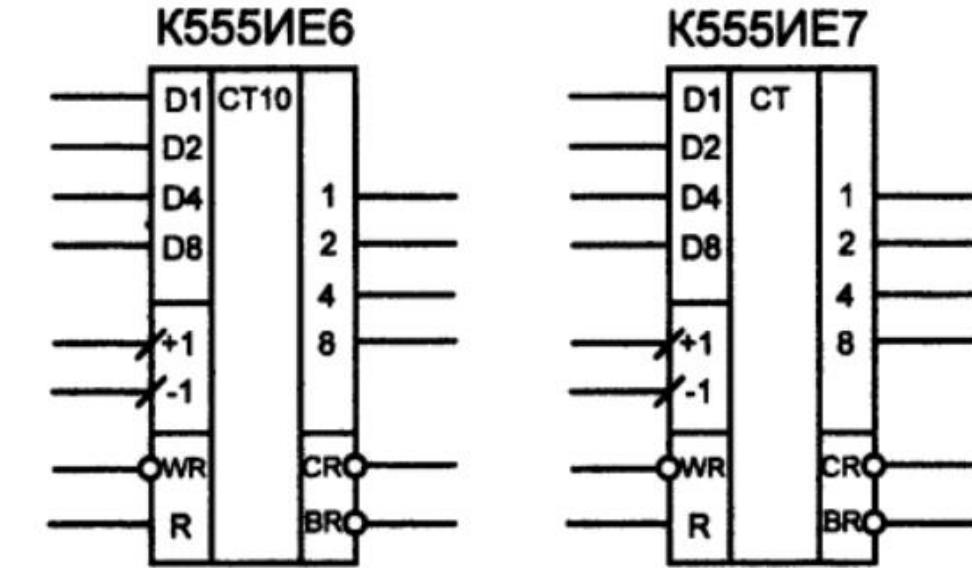


Асинхронный последовательный вычитающий счетчик на D-триггерах



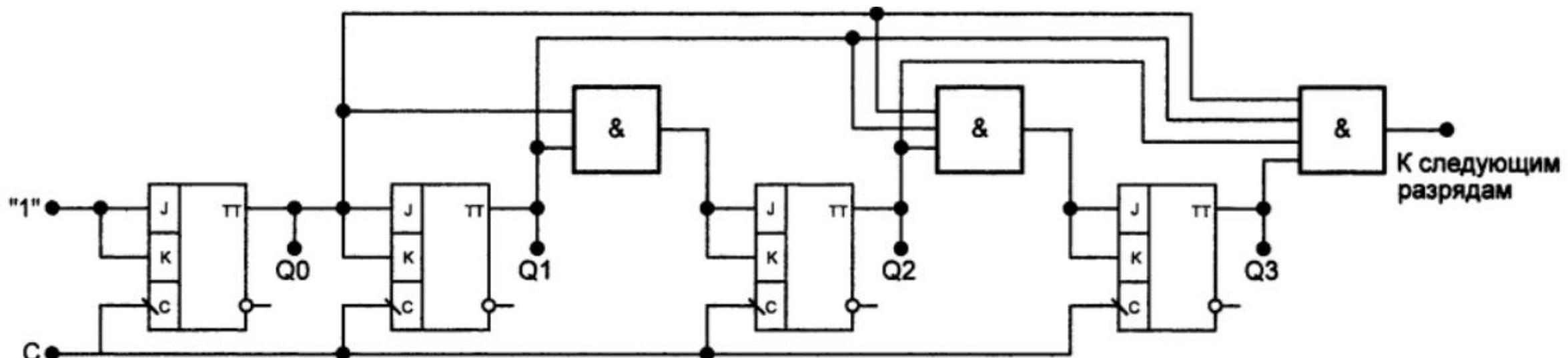
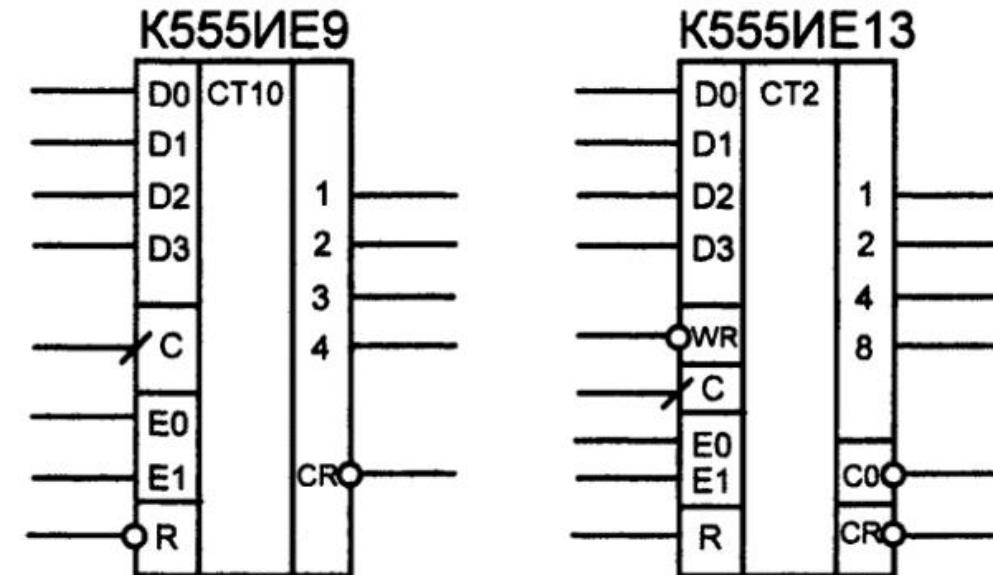
Синхронный счётчик с асинхронным переносом

- **Синхронные счетчики** с асинхронным переносом отличаются тем, что переключение разрядов осуществляется одновременно, а сигнал переноса вырабатывается с некоторой задержкой.
- Типичными представителями синхронных счетчиков с асинхронным переносом являются микросхема декадного реверсивного счетчика К555ИЕ6 (счет от 0 до 9) и двоичного четырёхразрядного реверсивного счётчика К555ИЕ7 (счет от 0 до 15).
- Декадный счётчик считает в двоично-десятичном коде от 0 до 10. Реверсивный счётчик - счётчик, работающий как в прямом, так и в обратном направлении счёта импульсов в зависимости от уровня управляющего сигнала (WR, он же V). +1 и -1 – тактовые входы C1 и C2 , R – сигнал обнуления, CR – сигнал переноса, BR – сигнал заёма, 8 нога – общий, 16 – напряжение питания.

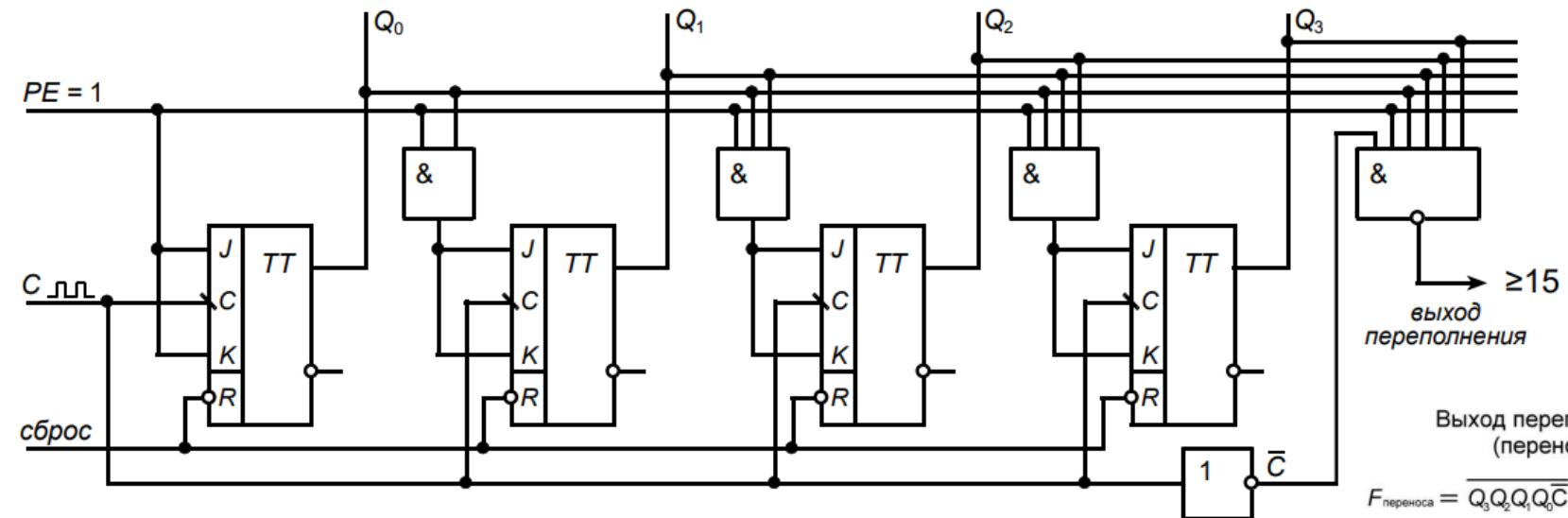


Синхронный счётчик

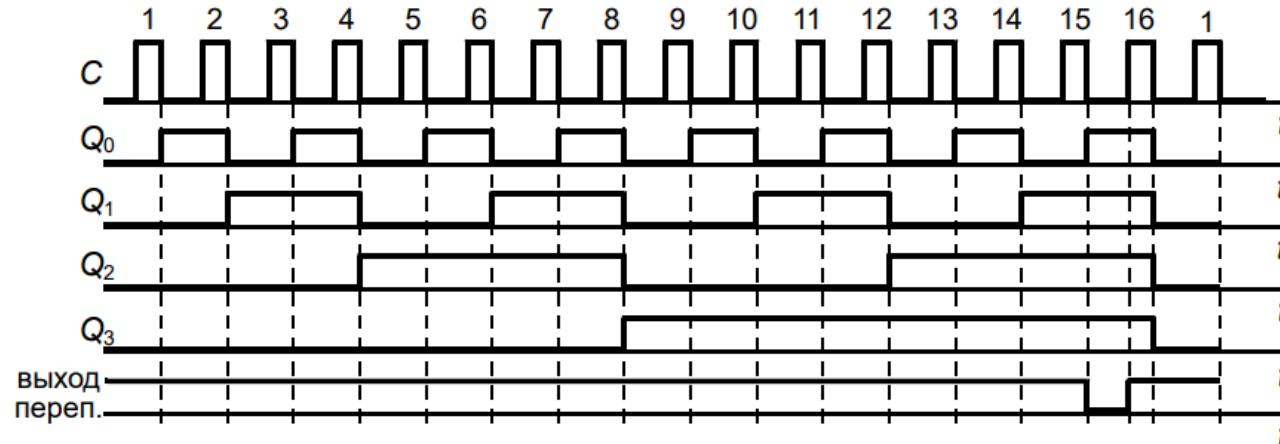
- В синхронных счётчиках переключение разрядов осуществляется одновременно.
- Сигнал переноса синхронно подается на вход разрешения счета каждого последующего счётчика.
- Синхронные счетчики могут полностью заменить функционал асинхронных и синхронных с асинхронным переносом счетчиков, имея самое высокое быстродействие.



Синхронный счётчик

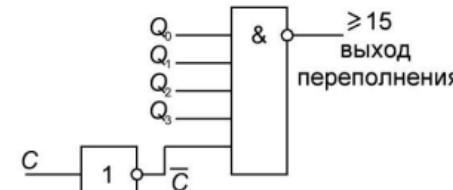


PE = 1 – разрешение счёта



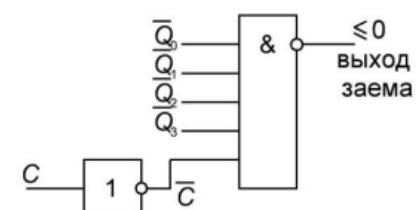
Синхронный счётчик, т.к. выходные сигналы появляются по каждому тактовому импульсу. Счётчик с параллельным переносом (или параллельного типа), т.е. значение Q_i меняется, когда младшие разряды заполнены единицами.

$$F_{\text{переноса}} = \overline{Q_3 Q_2 Q_1 Q_0} \bar{C}$$



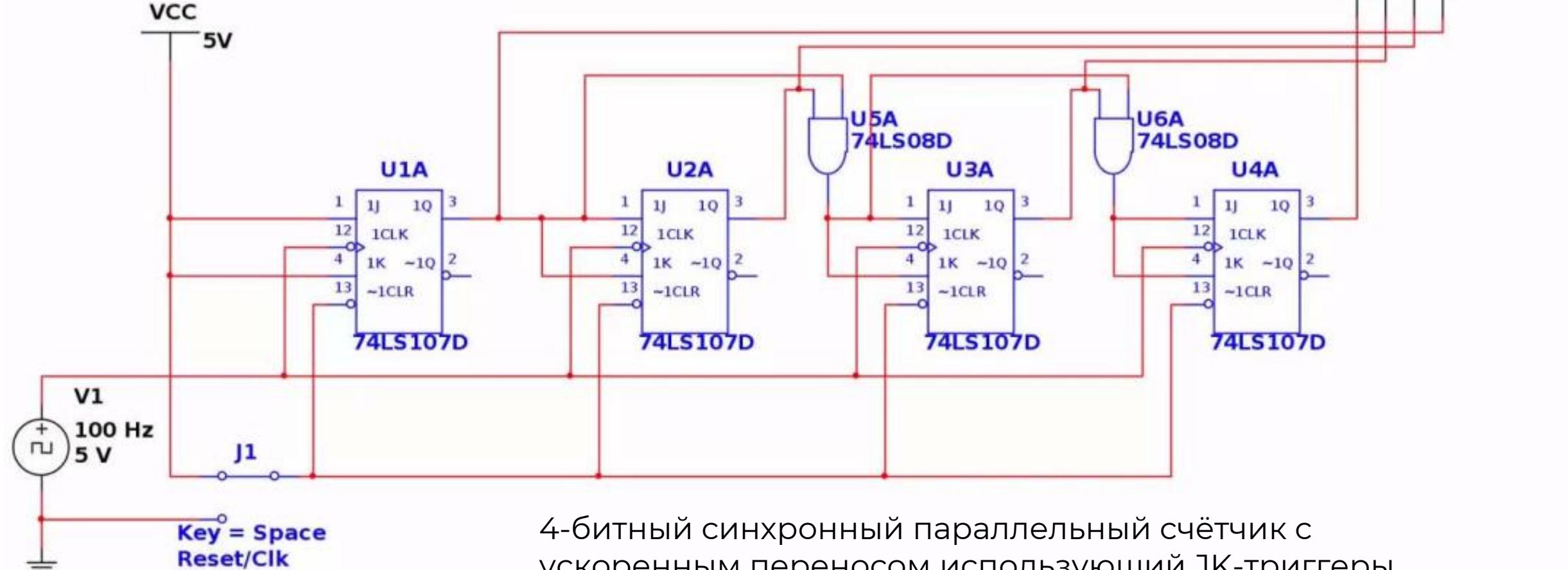
Выход заема

$$F_{\text{заема}} = \overline{\bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0} C$$

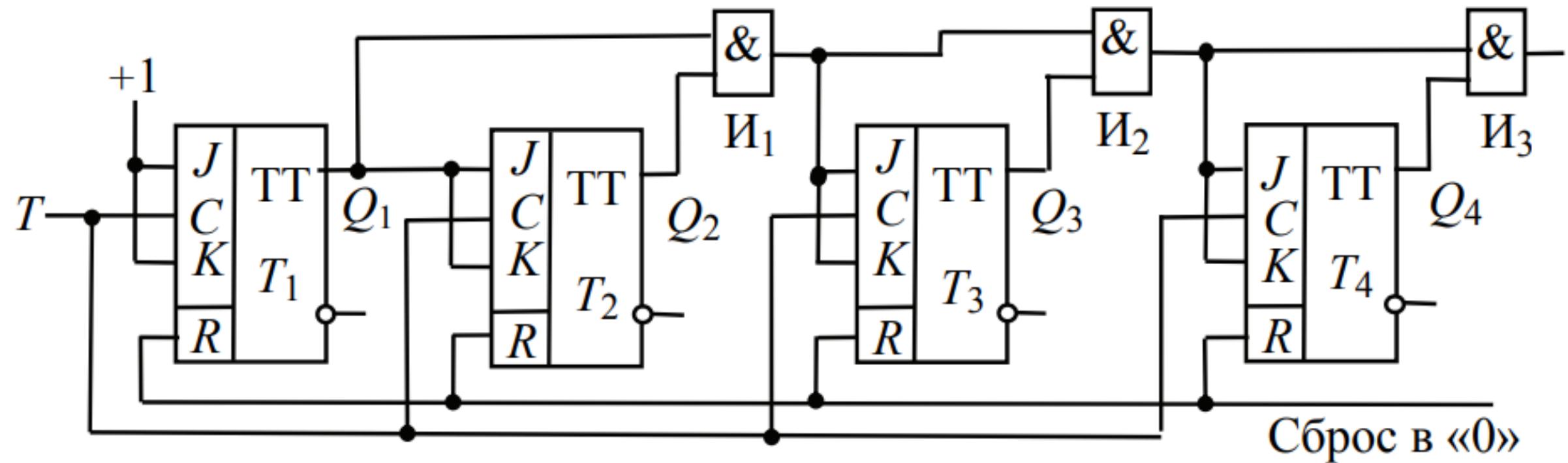


Синхронный счётчик

Перенос осуществляется последовательно, но на каждый триггер одновременно приходит тактовый сигнал синхронизации!

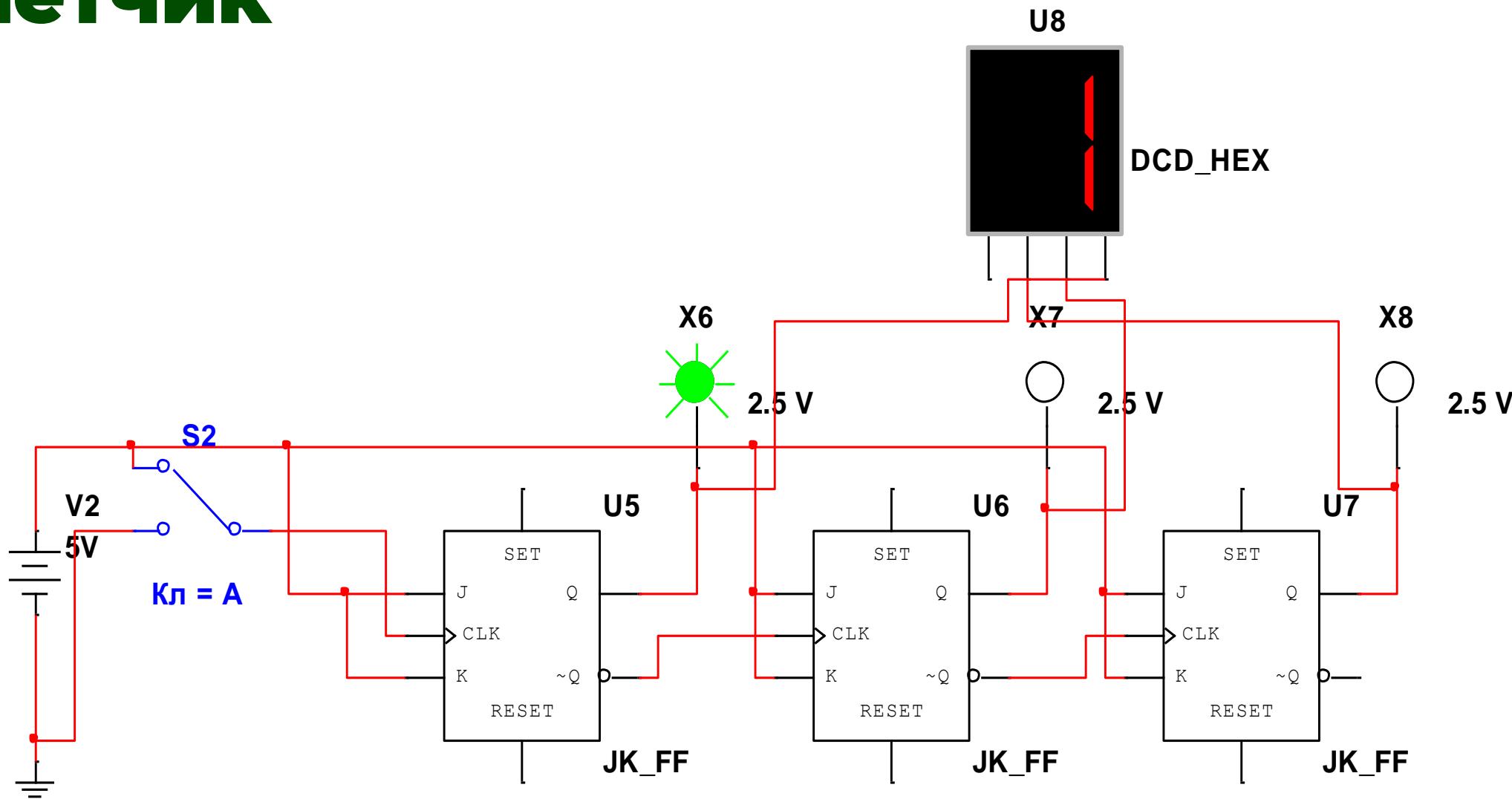


Счетчик



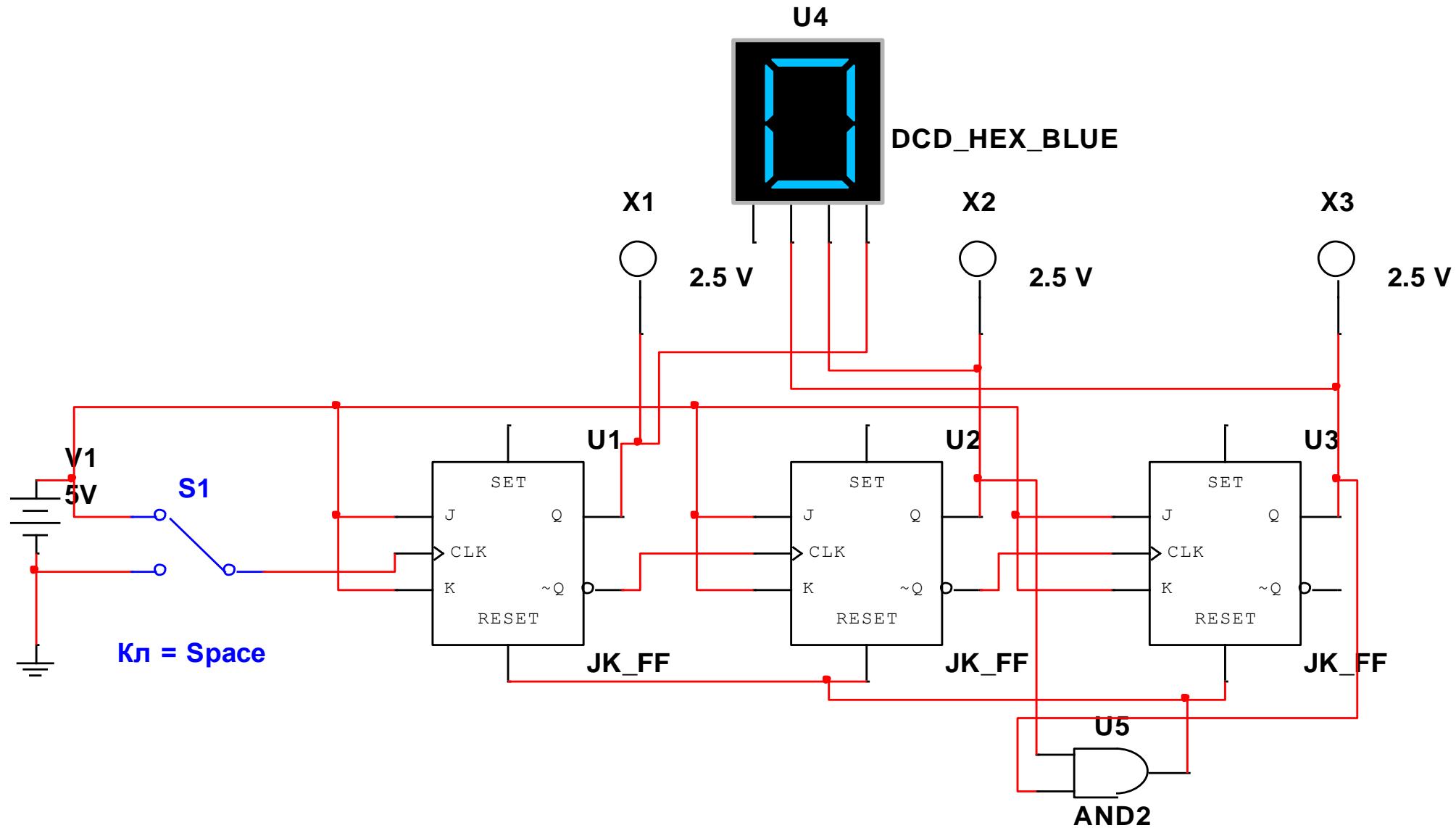
Четырехразрядный синхронный счетчик на JK-триггерах

Счетчик



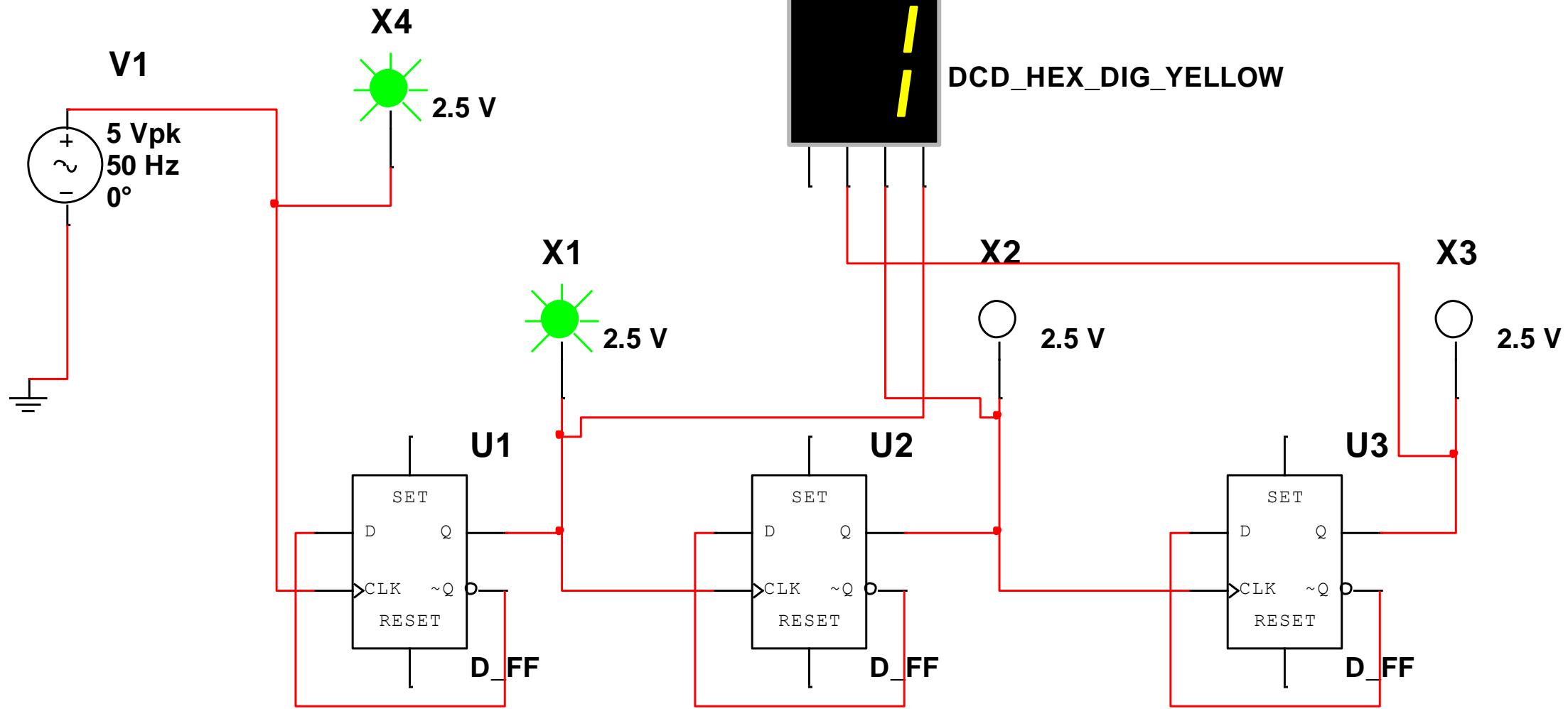
Суммирующий счетчик на JK-триггерах

Счетчик



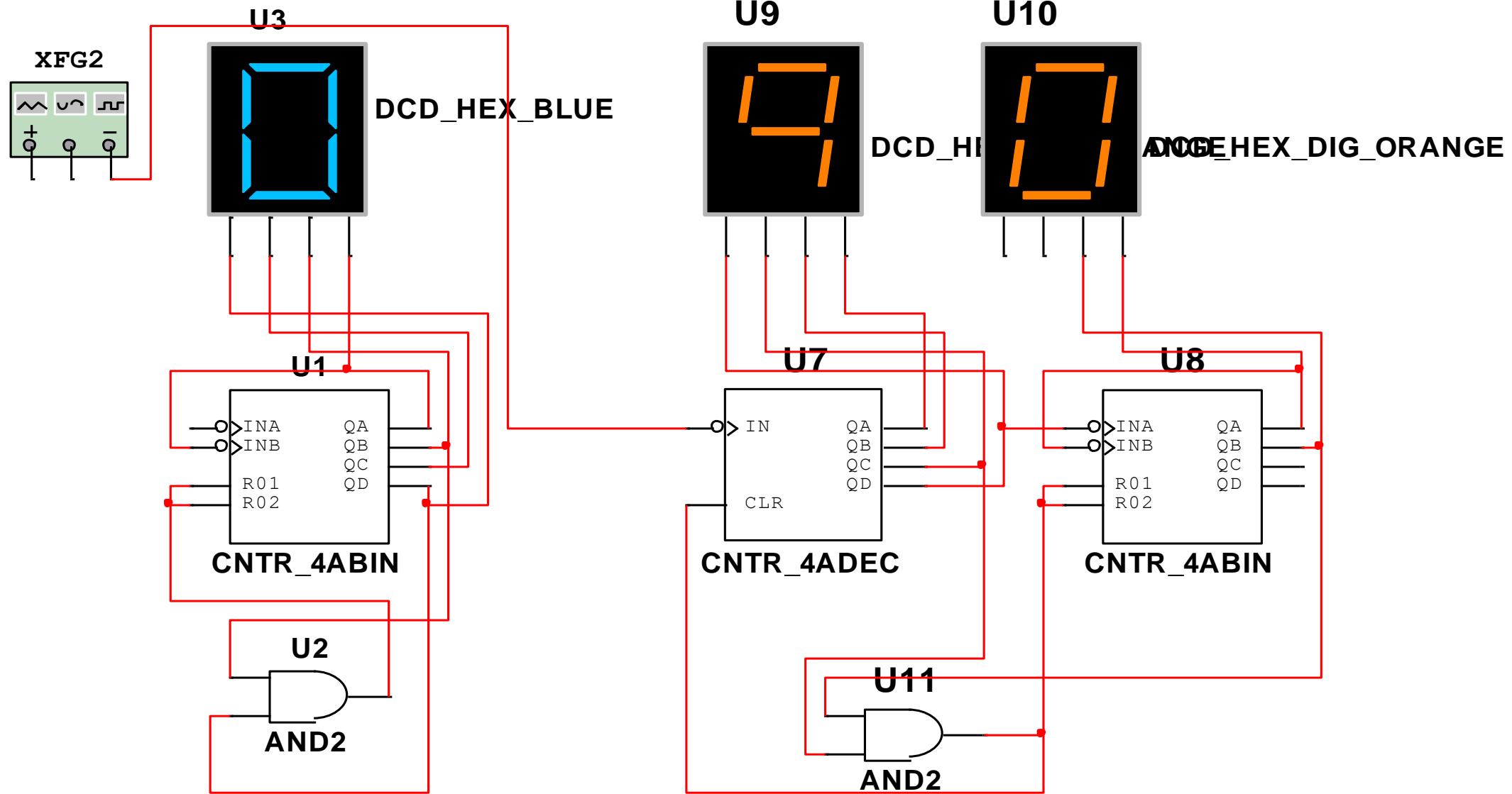
Суммирующий счетчик на JK-триггерах с произвольным коэффициентом счета

Счетчик



Вычитающий счетчик на D-триггерах

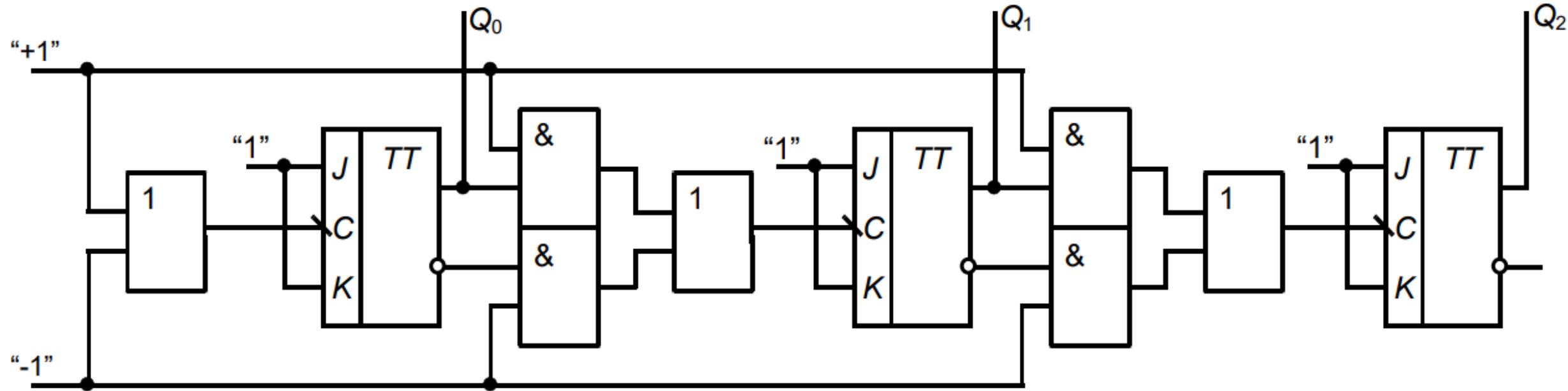
Счетчик



Счетчик на 10 и 24 на микросхемах

Реверсивный счётчик

- Реверсивный счётчик считает как в прямом, так и в обратном направлении.
- Как правило, имеет два тактовых входа: на увеличение (+1) и на уменьшение (-1).



В режиме вычитания входные импульсы подаются на вход «-1», при этом на вход «+1» подаётся лог. 0.
В режиме сложения входные импульсы подаются на вход «+1», а на вход «-1» следует подать лог. 0.

- В данной схеме – счётчик с последовательным переносом.
- Триггеры работают в как счётные триггеры.

Счётчики с предварительной установкой

Ряд счётчиков имеет информационные входы для предварительной записи (параллельной загрузки) информации. Сигналы, подаваемые на эти входы устанавливают триггеры счётчика в соответствующее состояние. В результате на выходах устанавливается двоичный код числа, с которого счётчик при следующем тактовом импульсе начнёт счёт.

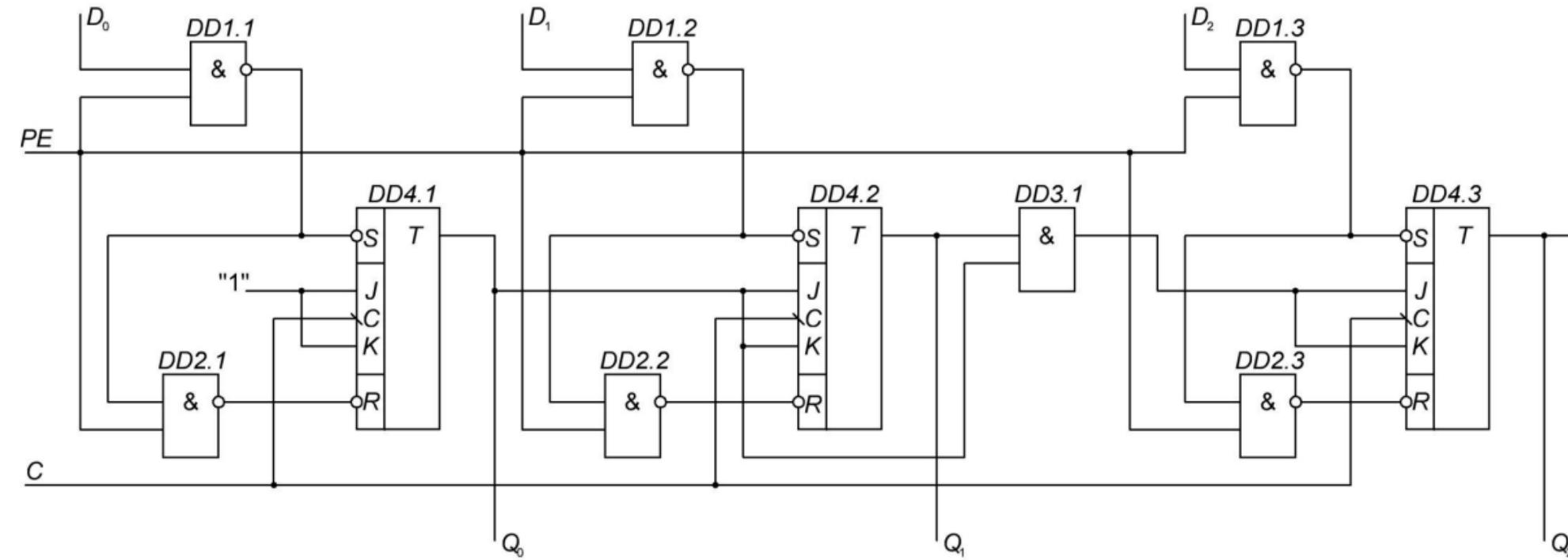
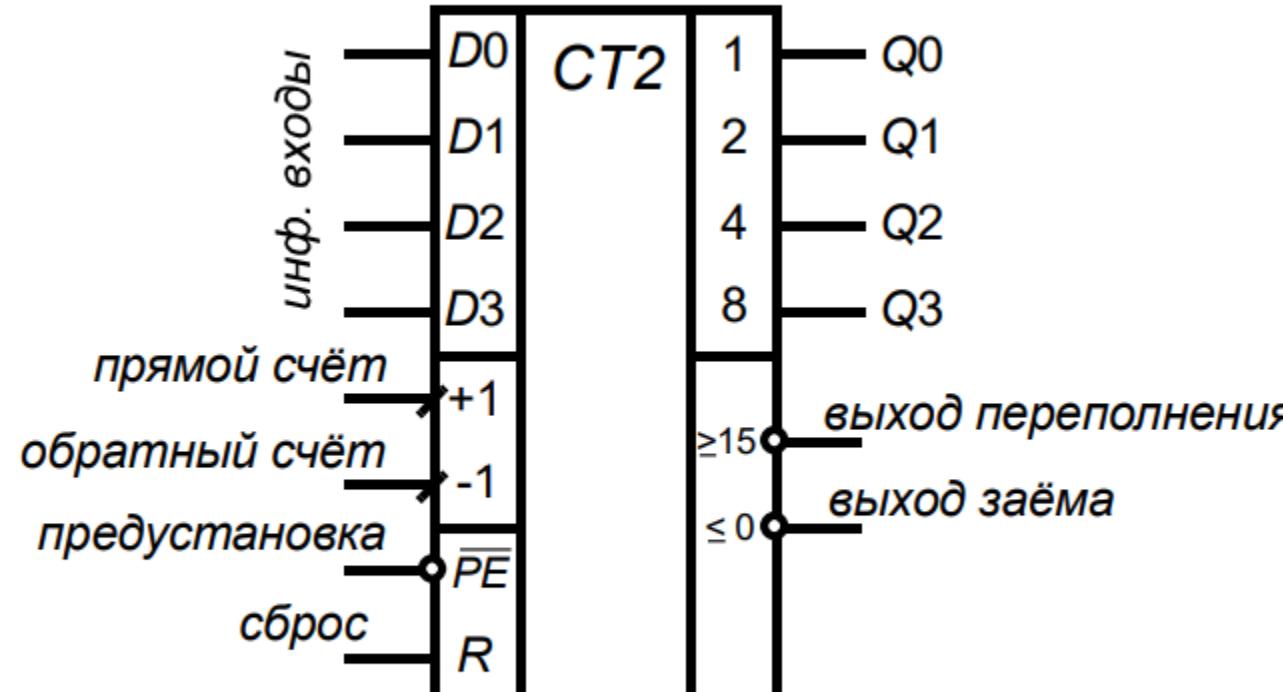


Схема трёхразрядного счётчика с предварительной установкой

На информационных входах $D_0 - D_2$ устанавливаем число в двоичном коде. Запись в счётчик (предустановка) осуществляется при $PE = 1$. В этом случае на выходах $Q_0 - Q_2$ появится двоичный код $Q_i = D_i$. С поступлением следующего тактового импульса счётчик начнёт счёт с установленного на выходах числа. Вход PE (preset enable) имеет приоритет над J, K и C .

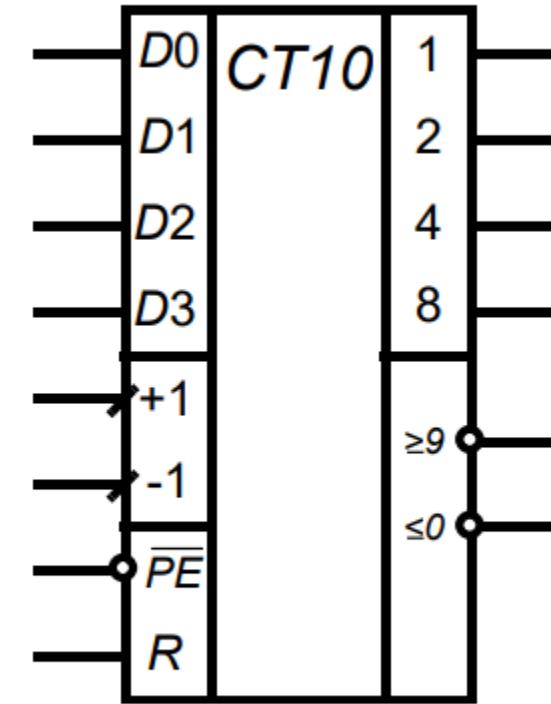
Счётчики с предварительной установкой

К1533ИЕ7



Двоичный счётчик с
предустановкой

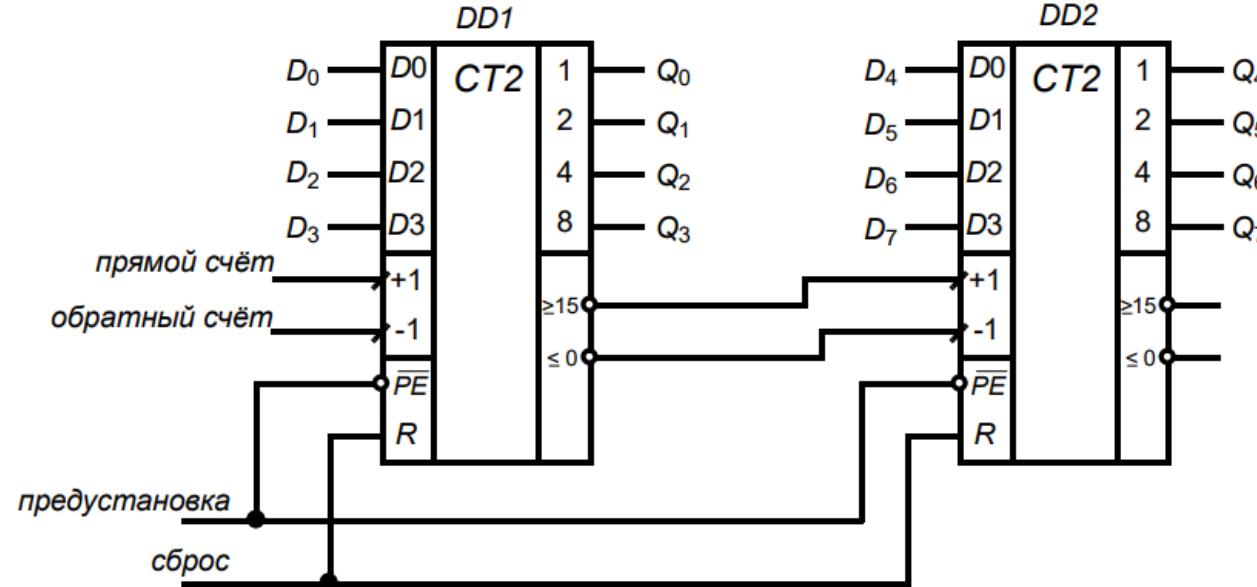
К1533ИЕ6



Двоично-десятичный
счётчик с предустановкой

Предустановка счётчика осуществляется подачей «0» на вход PE.

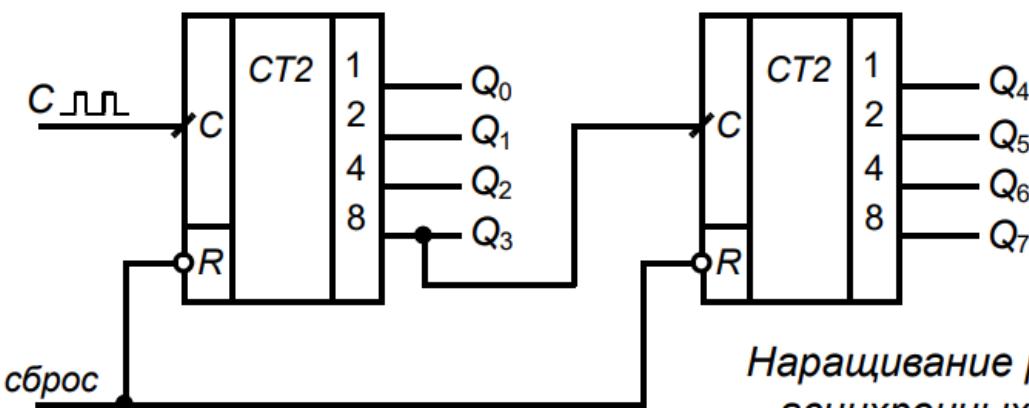
Способы увеличения разрядности счётчиков



Для наращивания
разрядности счётчиков
используются выходы
переполнения (переноса)
и заёма.

Наращивание разрядности синхронных
счётчиков с предустановкой

D₀ - *D₇* – информационные входы
Q₀ - *Q₇* – выходы



$2^8 = 256$
считает от 0 до 255

Наращивание разрядности
асинхронных счётчиков

Счётчики с произвольным коэффициентом счёта

- **Два способа** организации счётчиков с произвольным коэффициентом счёта:
 - 1. метод опознавания и сброса (метод управляемого сброса)
 - 2. метод предварительной установки
- **1. Метод опознавания и сброса** (метод управляемого сброса): сигнал сброса формируется путем логического умножения комбинации выходных сигналов $Q_0Q_1Q_2Q_3$ в зависимости от требуемого коэффициента счета.
- **2. Метод предварительной установки** заключается в том, что сигнал с выхода переполнения (заема) подается на вход разрешения предустановки PE, а на входах предустановки $D_0D_1D_2D_3$ устанавливается исходный двоичный код, с которого счётчик начнёт считать по приходу следующего синхроимпульса.

Счётчики с произвольным коэффициентом счёта

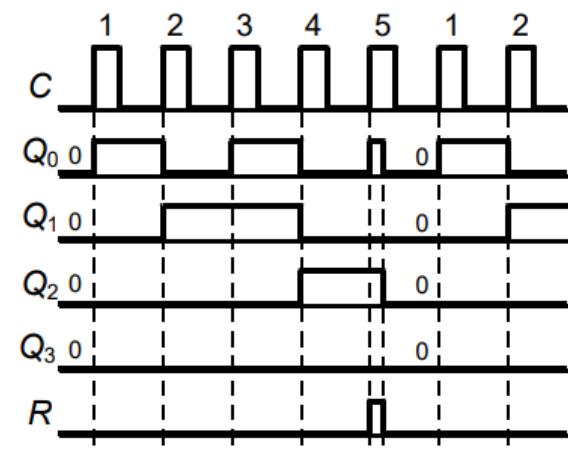
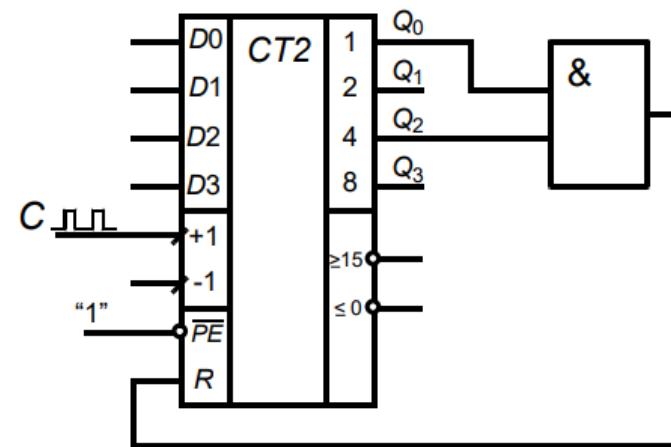
Два способа организации счётчиков с произвольным коэффициентом счёта:

1. метод опознавания и сброса (метод управляемого сброса)
2. метод предварительной установки

1. Метод опознавания и сброса

Реализовать схему совпадения и выдать результат на вход сброса. Схема совпадения выполняется на логике.

Пример. Реализовать счётчик с коэффициентом счёта 5 на суммирующем счётчике.



$$K_{\text{сч}} = 5$$

$$5_{10} = 0101_2$$

Счётчик считает начиная с 0000 на увеличение. Как только на выходах Q_0 и Q_2 появляются единицы (что будет соответствовать числу 0101) на выходе элемента И появится лог. 1, которая поступая на вход R , сбрасывает счётчик. Далее счёт начинается заново.

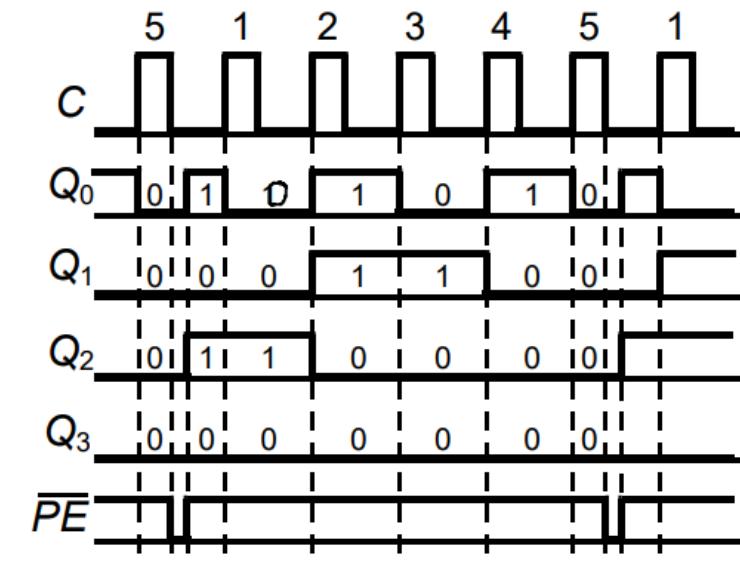
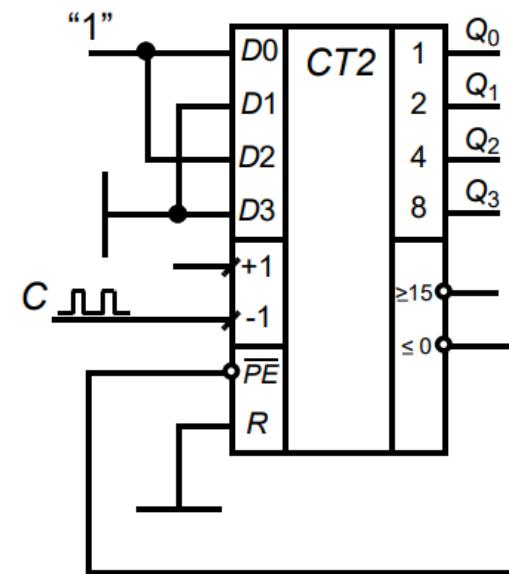
При организации счетчика на вычитание в данном методе необходимо считать первый импульс, устанавливающий на всех выходах 1111, следовательно число, до которого нужно считать равно $16 - 5 = 11$.

Счётчики с произвольным коэффициентом счёта

2. Метод предварительной установки

Пример 1. Реализовать счётчик с коэффициентом счёта 5. Счётчик работает на уменьшение.

Т.к. счётчик работает на уменьшение (вычитание), то необходимо использовать выход заёма (≤ 0) для загрузки кода в счётчик. Коэффициент счёта задаётся числом в двоичном коде, загружаемое через информационные входы.



$K_{\text{сч}} = 5$

$$5_{10} = 0101_2$$

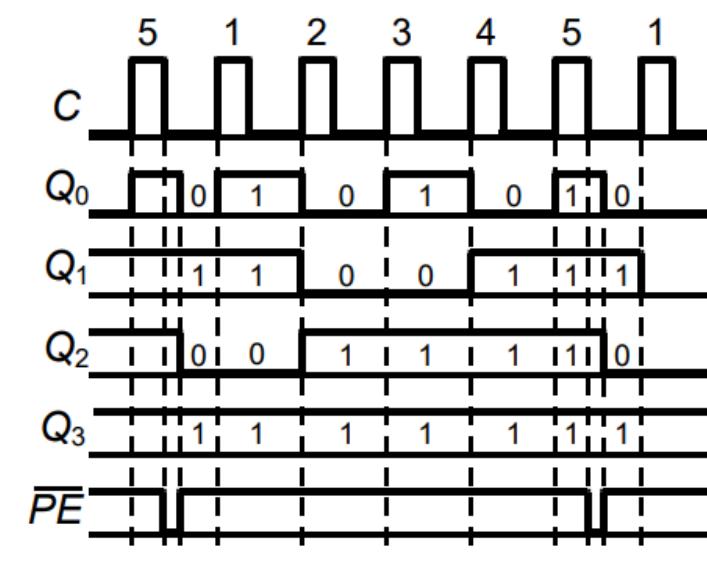
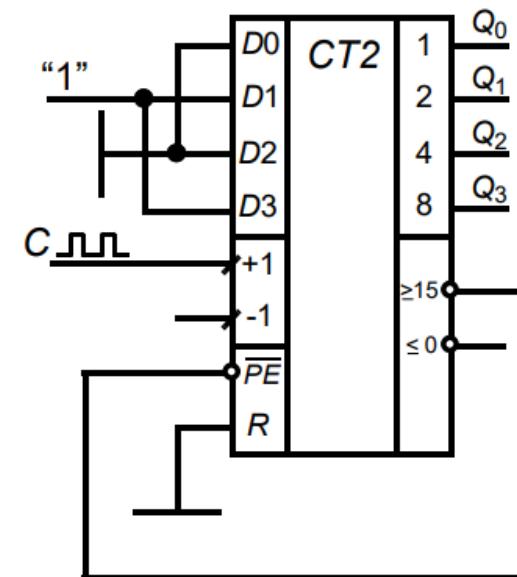
Для заданного коэффициента счёта на информационные входы необходимо подать число $5_{10} = 0101_2$. С каждым поступающим на вход импульсом содержимое счётчика уменьшается на 1. Когда счётчик обнулится (на выходах 0000), на выходе заёма появится лог. 0, который снова загрузит число $5_{10} = 0101_2$, и счёт начнётся заново.

Счётчики с произвольным коэффициентом счёта

2. Метод предварительной установки

Пример 2. Реализовать счётчик с коэффициентом счёта 5. Счётчик работает на увеличение.

Поскольку счётчик работает на увеличение, то необходимо использовать выход переполнения (переноса), а на информационные входы подать число равное ($15_{10} - K_{\text{сч}}$).



$$K_{\text{сч}} = 5$$

$$5_{10} = 0101_2$$

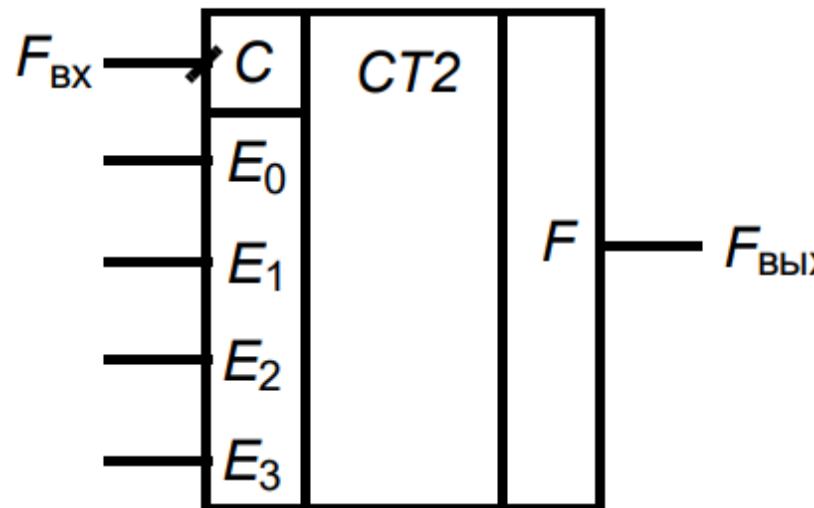
На информационные входы необходимо подать число:
 $15_{10} - 5_{10} = 10_{10} = 1010_2$.

Счётчик отсчитывает 5 импульсов. На выходе переполнения формируется сигнал (лог. 0), который вновь загружает число $10_{10} = 1010_2$ в счётчик, и счёт начинается заново. Перезагрузка счётчика происходит при его переполнении, когда $Q_0 = Q_1 = Q_2 = Q_3 = 1$.

Счётчики с произвольным коэффициентом счёта

Счётчики с установкой коэффициента счёта

Счётчики с установкой коэффициента счёта – специальные счётчики-делители частоты. Коэффициент деления (счёта) задаётся двоичным числом на информационных входах E_5-E_0 .



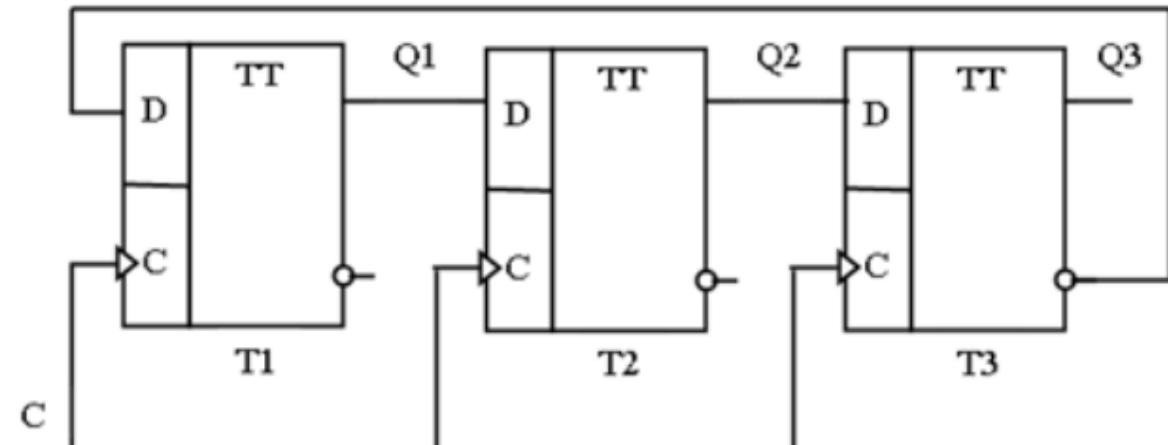
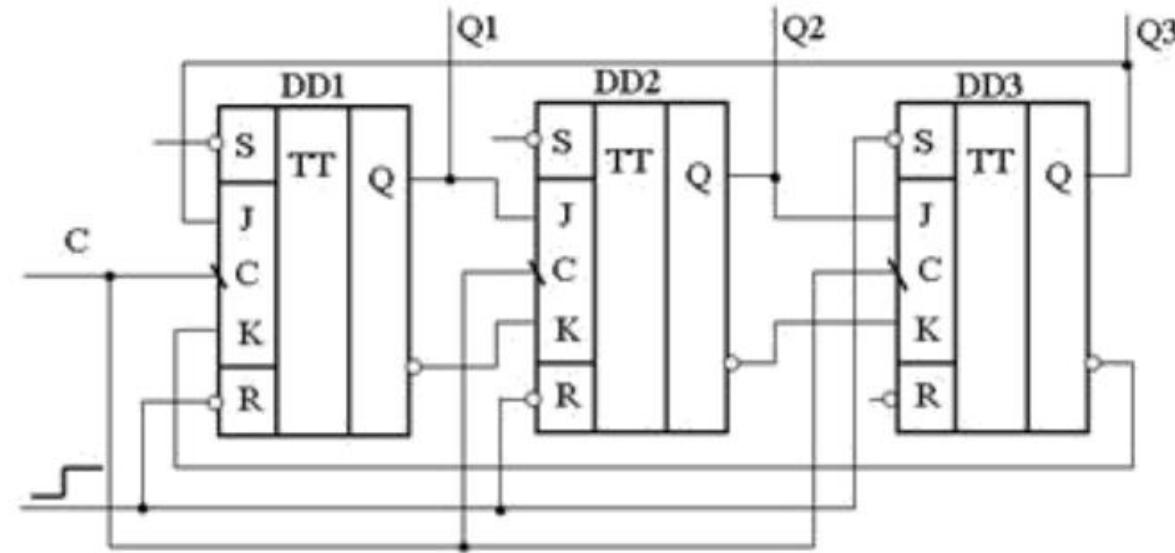
Частота выходных импульсов
определяется по формуле:

$$F_{\text{вых}} = \frac{F_{\text{вх}}}{K_{\text{дел}}} = \frac{F_{\text{вх}}}{E_0 \cdot 2^0 + E_1 \cdot 2^1 + E_2 \cdot 2^2 + E_3 \cdot 2^3}$$

Кольцевые счетчики

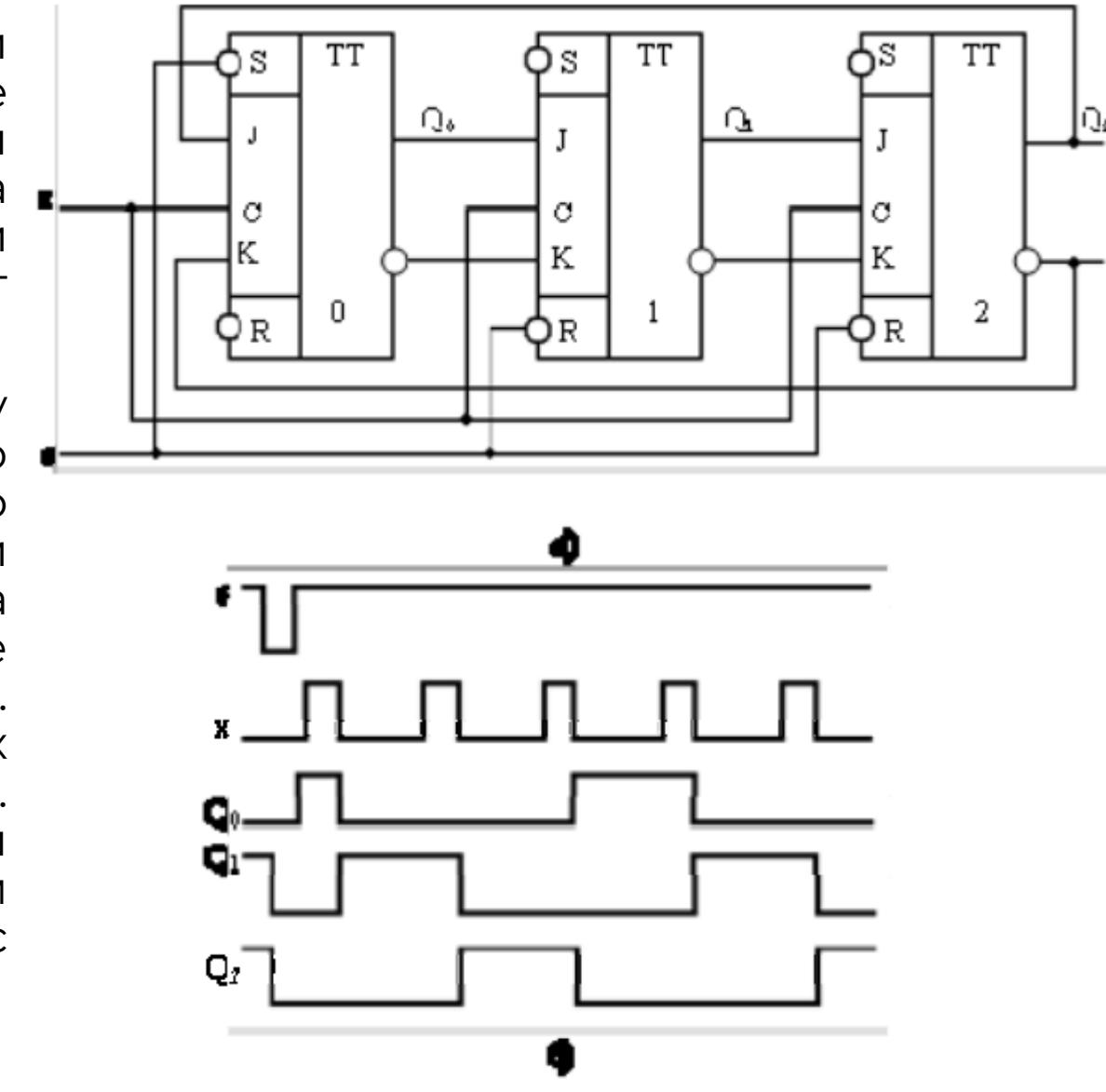
- **"Кольцевые" счетчики** – это, на самом деле, замкнутые в кольцо регистры сдвига, состояния триггеров в которых изменяются под воздействием входных сдвигающих импульсов.
- В простейшем случае по кольцу циркулирует одна кодовая единица, так что коэффициент пересчета счетчика равен числу входящих в него триггеров.

• **Счетчик Джонсона** является разновидностью кольцевого счётчика. Он имеет коэффициент пересчета, вдвое больший числа составляющих его триггеров. В частности, если счетчик состоит из трех триггеров, то он будет иметь шесть устойчивых состояний. Код, в котором работает счетчик Джонсона, называют кодом Либау - Крейга.



Кольцевые распределители, счетчики

- **Распределитель** - цифровой узел, который преобразует временное распределение импульсов в пространственное, когда каждый следующий импульс из входного потока отправляется на свой выход. Распределители строят на базе регистров сдвига, но называют их, тем не менее, счетчиками.
- **Кольцевой счетчик** - сдвиговый регистр, у которого выход триггера самого старшего разряда соединен со входом триггера самого младшего разряда, и в любой момент времени только в одном триггере регистра записана единица. Эта единица при каждом сдвиге перемещается из одного триггера в другой. Так поток сдвиговых импульсов x распределяется по триггерам (в пространстве). Из всех 2^n+1 возможных состояний разрешенными для регистра в кольцевом счетчике являются только $(n + 1)$ состояний с единицей лишь в одном из $(n + 1)$ разрядов.



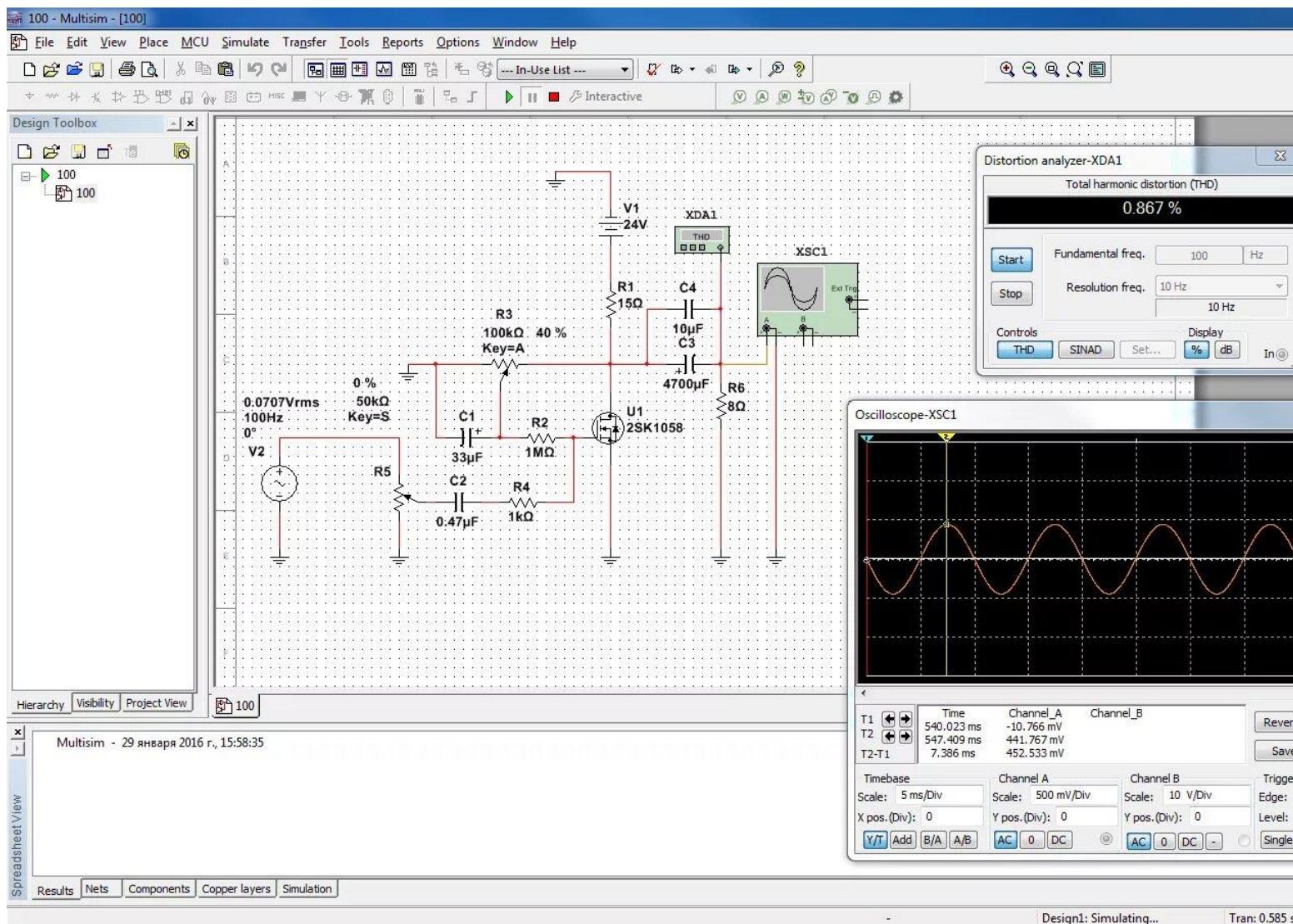


Multisim

Изучение работы
последовательной
и комбинационной
логики



Multisim



Multisim – приложение для создания и тестирования электрических схемотехники.

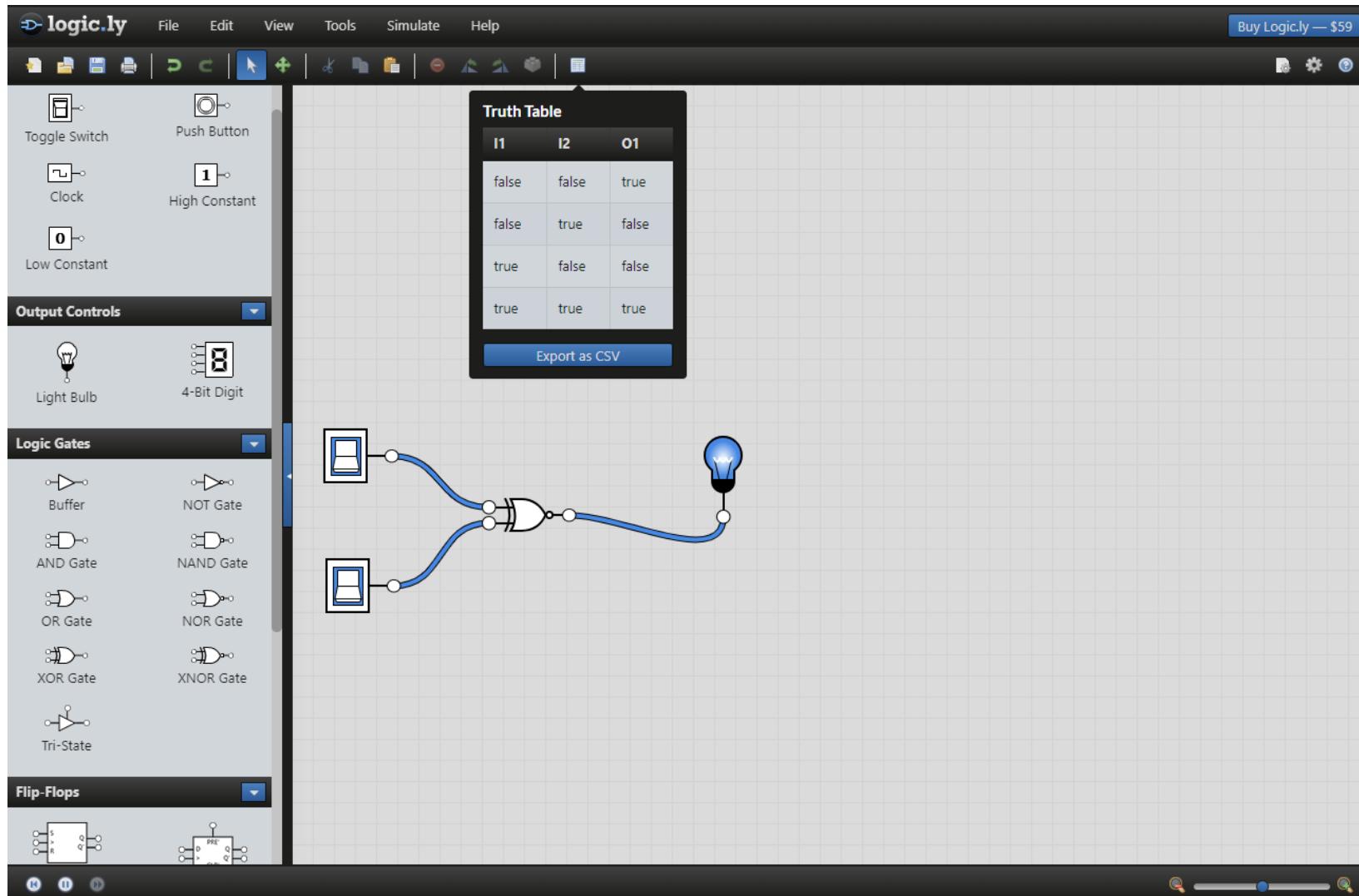
Оно включает сотни виртуальных электронных компонентов и измерительных устройств, умеет моделировать их работу и взаимодействие, что необходимо для проектирования, анализа и отладки электрических схем. Программа поддерживает имитацию различных режимов работы цепей и применяется на всех этапах их разработки.

NI Circuit Design Suite 14.3 x64 [Multisim & Ultiboard, 2022, ENG] <https://rutracker.org/forum/viewtopic.php?t=6225014>

NI Circuit Design Suite 14.2 [Multisim & Ultiboard, EN/RU] <https://rutracker.org/forum/viewtopic.php?t=4994297>

Multisim & Ultiboard Power Pro 14.1.31 + Rus 14.1.31 PRO x86 x64 [2017, ENG + RUS] <https://rutracker.org/forum/viewtopic.php?t=5870078>

<https://logic.ly/demo/>



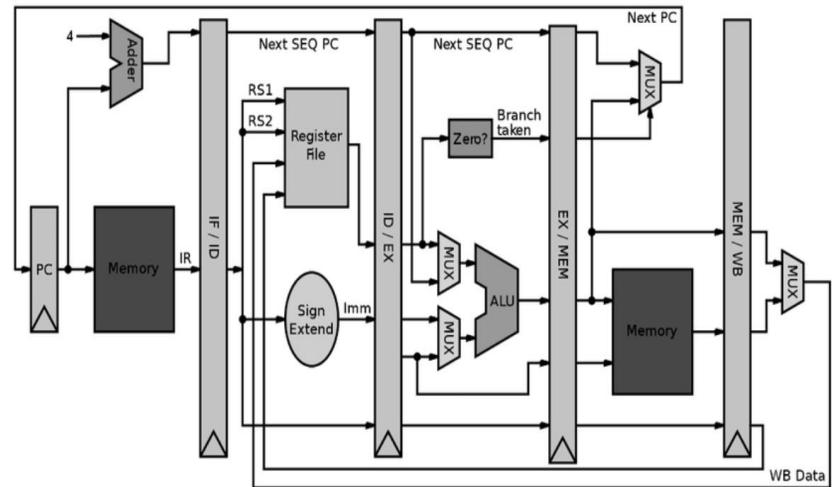
Сервис logic.ly позволяет строить логические схемы, используя входные (переключатели) и выходные сигналы (лампочку), а также логические элементы (NOT, OR, AND, XOR). Пользователь может построить схему любой сложности.

Логические схемы в Logic.ly позволяют:

- научиться использовать логические элементы, входные и выходные сигналы;
- получить навык построения логических схем;
- переключать входные данные в режиме реального времени;
- выполнять проверку заполненных таблиц истинности;
- понимать как работают логические схемы.

Изучение работы логических элементов, последовательной и комбинационной логики будем осуществлять на лабораторных работах и во время выполнения курсовой работы.

Современные методы проектирования цифровых устройств



Высокий уровень абстракции

- Процесс поиска наилучшего набора логических элементов для выполнения заданной логической функции трудоемок и может приводить к ошибкам, так как требует упрощения логических таблиц или выражений и перевода конечных автоматов в представление на уровне логических элементов вручную.
- **В 1990-е годы разработчики обнаружили, что их производительность труда резко возрастала, если они работали на более высоком уровне абстракции**, определяя только логическую функцию и предоставляя создание оптимизированных логических схем системе автоматического проектирования (САПР).
- Два основных языка описания аппаратуры (Hardware Description Language, HDL) – SystemVerilog и VHDL.
- Большинство коммерческих систем сейчас строятся с использованием языков описания цифровой аппаратуры, а не на уровне схемотехники.

Управление сложностью в микроэлектронике

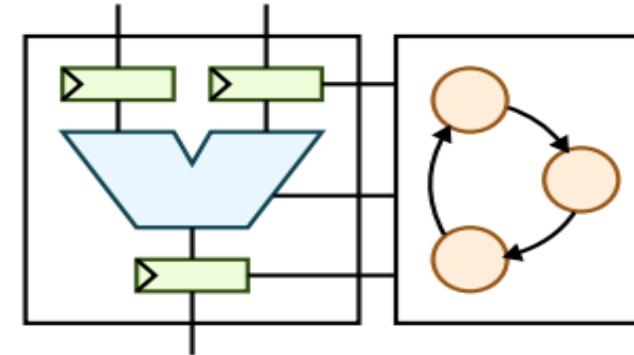
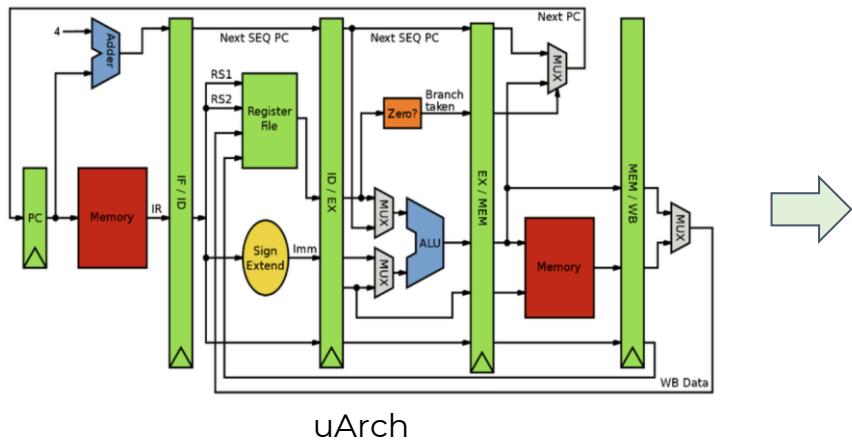


Уровень RTL в проектировании цифровых вычислительных систем

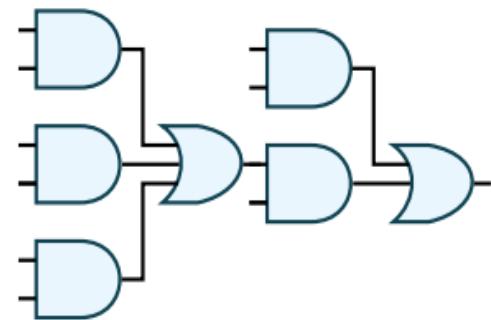
Уровни абстракции в проектировании вычислительных систем:

- **Прикладное ПО:** пользовательские программы и библиотеки
- **Системное ПО:** ОС, драйверы, загрузчики
- **Компиляторы:** переменные, граф управления, ...
- **ISA:** инструкции, регистры, память, режимы адресации, ...
- **Микроархитектура:** АЛУ, кеш, конвейер, ОоО, микрооперации, ...
- **Цифровая электроника:** логические элементы, счетчики, мультиплексоры, ...
- **Аналоговая электроника:** ФНЧ, RC-цепь, полевой транзистор, ...
- **Физика:** p/n-полупроводник, изолятор, электромагнитное поле, ...

Уровень регистровых передач (англ. register transfer level, RTL) — способ разработки синхронных цифровых интегральных схем, при применении которого работа схемы описывается в виде последовательностей логических операций, применяемых к цифровым сигналам (данным) при их передаче от одного регистра к другому (то есть не описывается, из каких электронных компонентов или из каких логических вентилей состоит схема).



RTL

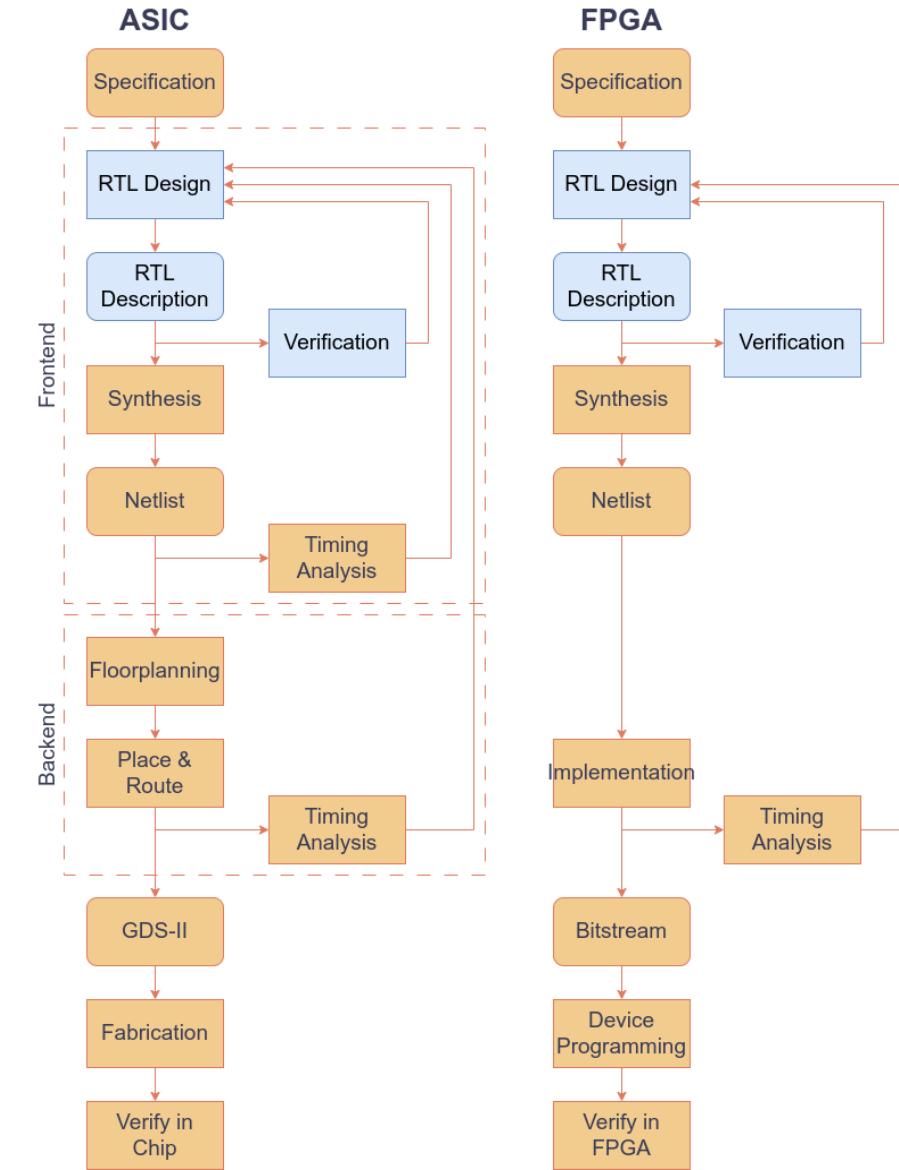


Gate-level

- Можно описывать микроархитектурные объекты набором примитивов цифровой схемотехники, но **удобнее использовать RTL-описание**

HDL в проектировании ASIC и разработки под FPGA

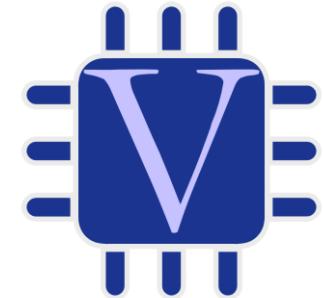
- **Hardware Description Language (HDL)**
используются для создания RTL-описания аппаратуры
 - Verilog/SystemVerilog
 - VHDL
- **Application Specific Integrated Circuit (ASIC)**
 - проектируется с помощью HDL
- **Field Programmable Gate Array (FPGA)** –
конфигурируется с помощью HDL
- **Общие этапы** – спецификация, RTL-описание, верификация и др.
- **Верификационное окружение** также может создаваться на HDL
- **RTL-описание может генерироваться** из более высокоуровневого описания (MATLAB/Simulink, Xilinx Vitis HLS, Intel HLS, SystemC, Chisel, SpinalHDL, Migen, PyMTL, Bluespec HDL и др.)



Verilog, VHDL и SystemVerilog

Verilog

- Разработан в 1984 году как язык **для моделирования** логических схем
- Позже стал применяться для логического синтеза
- Стал стандартом IEEE в 1995 году
- Файлы: .v, .vh



VHDL

- Разработан в 1981 году на основе языка ADA **для моделирования и формального описания** логических схем
- Позже стал применяться для логического синтеза
- Стал стандартом IEEE в 1987 году
- Файлы: .vhd

VHDL

Very High Speed Integrated Circuit
Hardware Description Language

SystemVerilog

- В 2005 году Verilog был расширен для лучшей поддержки моделирования и верификации
- **Verilog и расширения были объединены** в стандарт IEEE 1800-2009
- Файлы: .sv, .svh

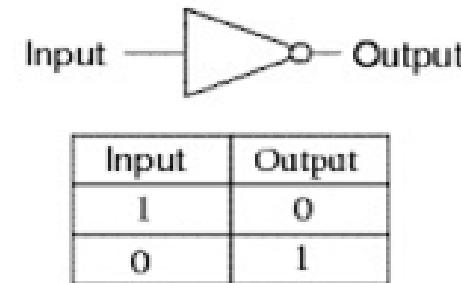
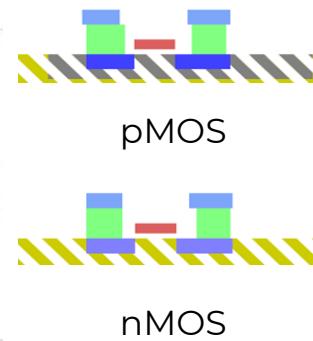
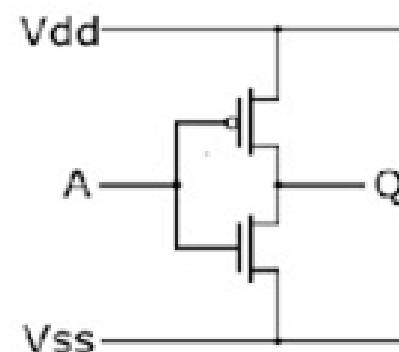
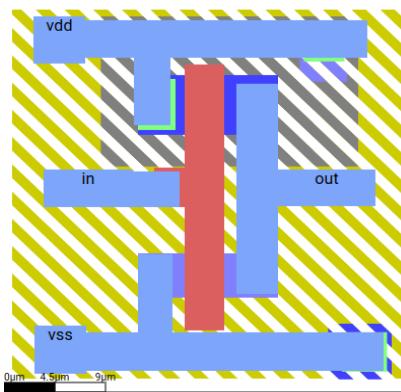


SystemVerilog и VHDL

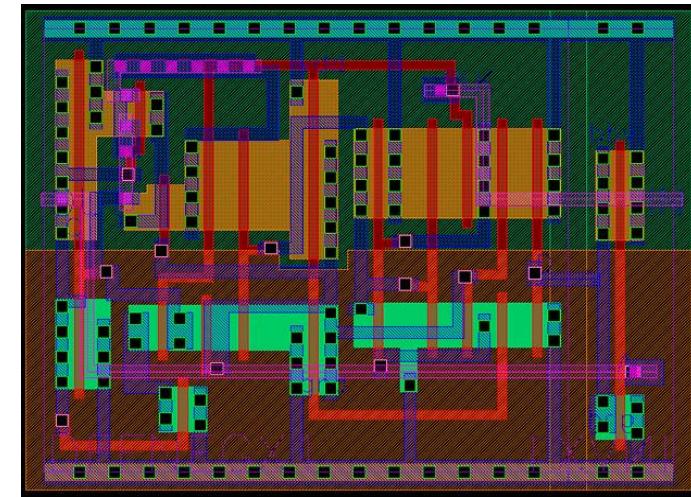
- **SystemVerilog**
- Verilog был разработан компанией Gateway Design Automation в 1984 году как проприетарный язык для моделирования логических схем.
- В 1989 году Gateway приобрела компания Cadence, и Verilog стал открытым стандартом в 1990 году под управлением сообщества Open Verilog International.
- Язык стал стандартом IEEE в 1995 году.
- В 2005 году язык был расширен для устранения противоречий в языке и лучшей поддержки моделирования и верификации систем.
- Эти расширения были объединены в единый стандарт, который сейчас называется SystemVerilog (стандарт IEEE 1800-2009).
- Файлы языка SystemVerilog обычно имеют расширение .sv.
- **VHDL**
- Аббревиатура VHDL расшифровывается как VHSIC Hardware Description Language.
- VHSIC, в свою очередь, происходит от сокращения Very High Speed Integrated Circuits – названия программы министерства обороны США.
- Разработка VHDL была начата в 1981 году министерством обороны для описания структуры и функциональности электронных схем. За основу для разработки был взят язык программирования ADA.
- Изначальной целью языка была документация, но затем он был быстро адаптирован для моделирования и синтеза. IEEE стандартизировал его в 1987 году, и после этого язык обновлялся несколько раз.
- Файл языка VHDL имеет расширение .vhd.

ASIC

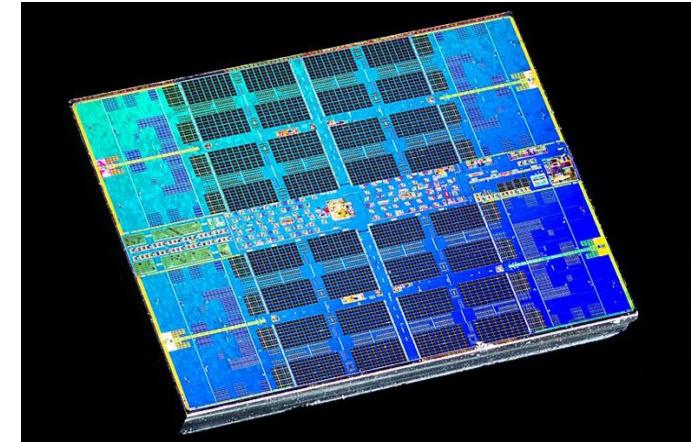
- **ASIC** – Application Specific Integrated Circuit
- **ИМС** – интегральная микросхема
- **VLSI** – Very-Large-Scale Integration
- **Производятся по технологии CMOS** (КМОП – комплементарная структура металл-оксид-полупроводник) с помощью фотолитографии на пластине из поликристаллического кремния
- **Стоимость производства** набора фотошаблонов – порядка \$100K
- Могут иметь в составе как **цифровые**, так и чисто **аналоговые** блоки
- **В уже произведенной микросхеме ничего изменить нельзя**



Инвертор, выполненный по КМОП-технологии



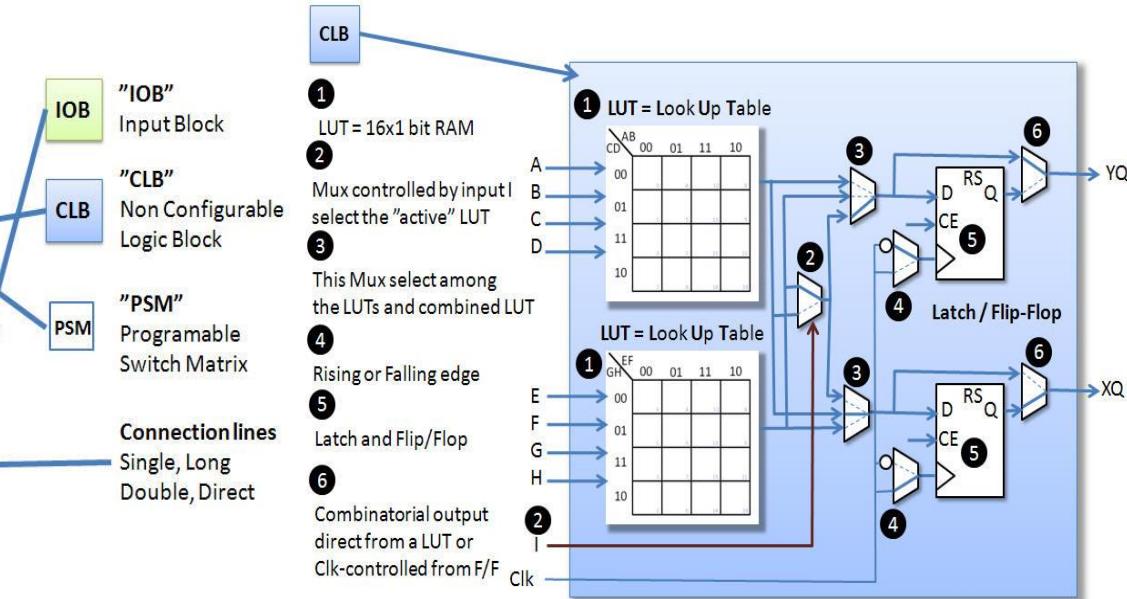
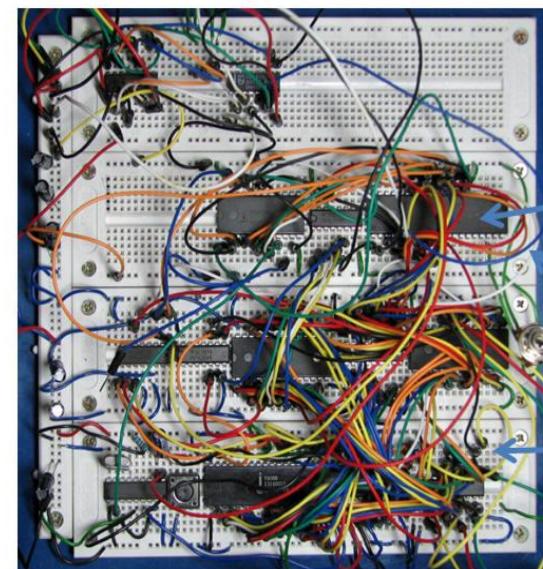
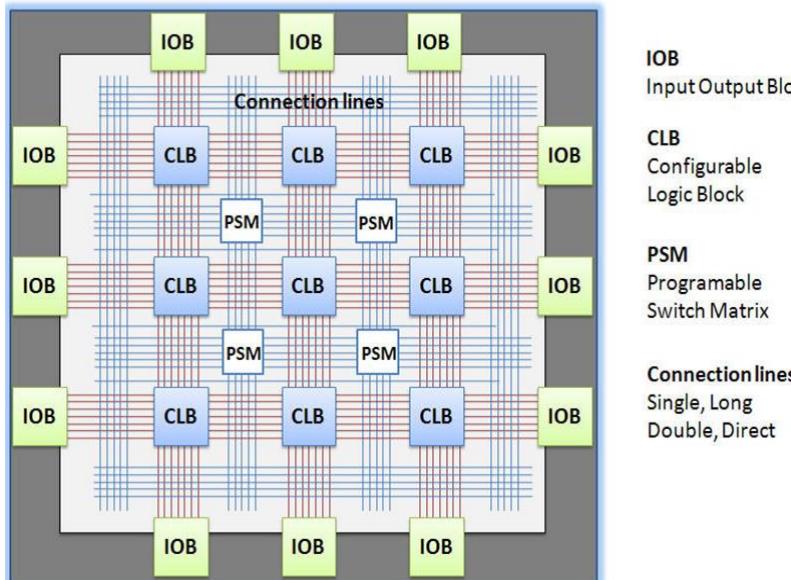
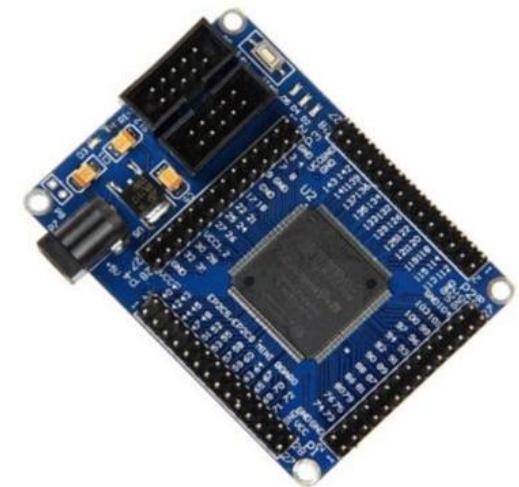
GDS-II (Graphic Database System)



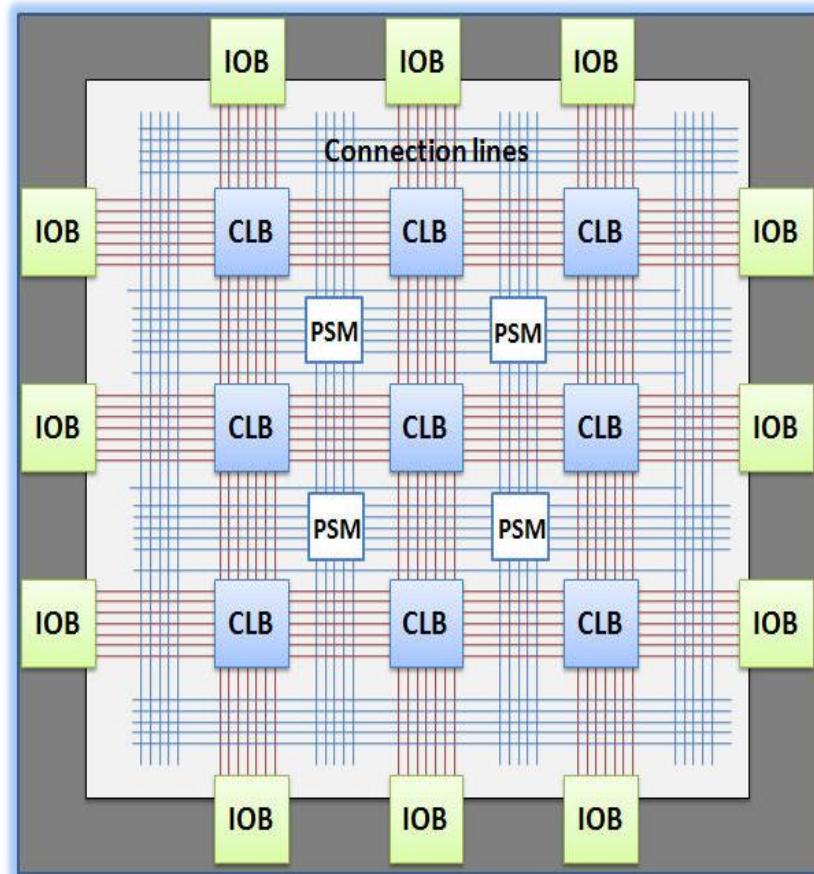
AMD Ryzen Zen 2

ПЛИС (FPGA)

- **FPGA** – Field-Programmable Gate Array
- **ПЛИС** – программируемая логическая интегральная схема
- **Набор ресурсов фиксирован**, но есть возможность реконфигурации цифровых ячеек
- Используются в обработке радиосигналов, изображений, сетевых пакетов и др.
- **Подходят для прототипирования перед производством ASIC**



ПЛИС (FPGA)



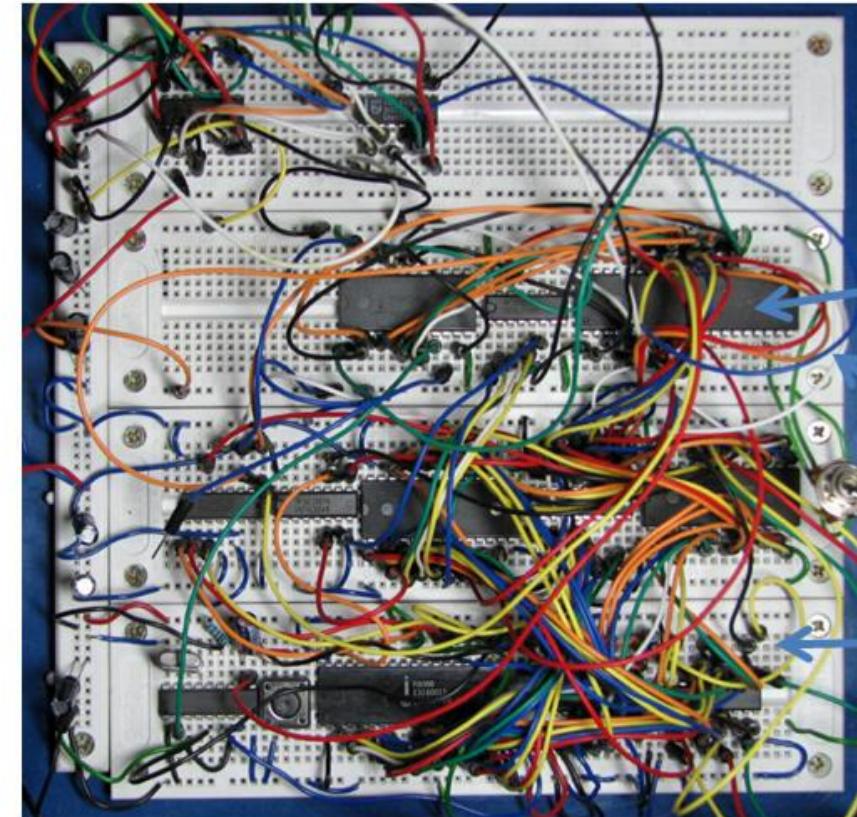
Внутри ПЛИС (FPGA)

IOB
Input Output Block

CLB
Configurable
Logic Block

PSM
Programable
Switch Matrix

Connection line
Single, Long
Double, Direct

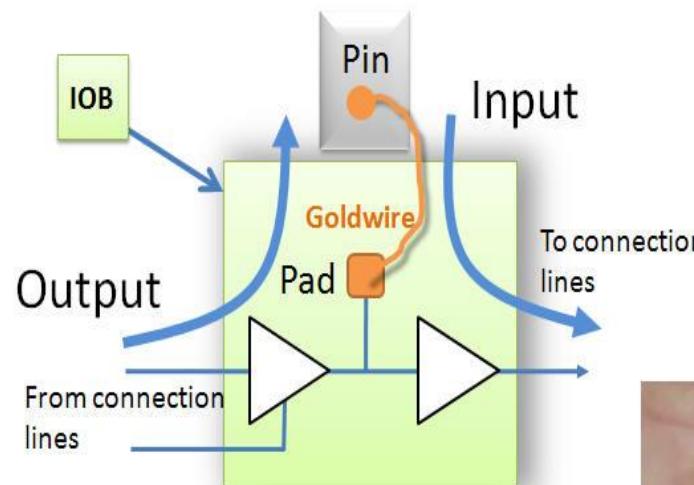


Старая «альтернатива» ПЛИС (FPGA)

"IOB"	Input Block
"CLB"	Non Configurable Logic Block
"PSM"	Programmable Switch Matrix
Connection lines	Single, Long Double, Direct

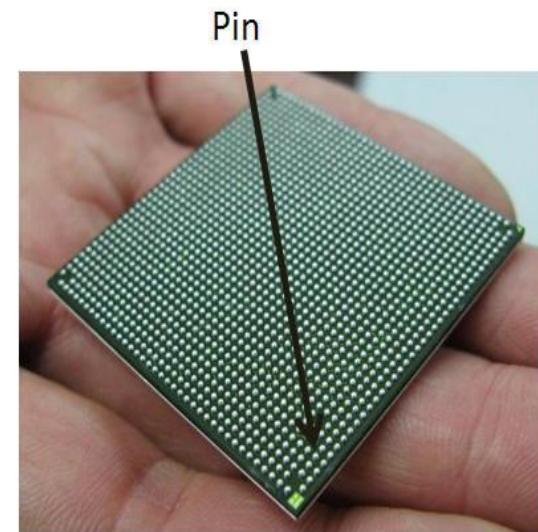
- **FPGA** – Field-Programmable Gate Array
- **ПЛИС** – программируемая логическая интегральная схема

ПЛИС (FPGA)

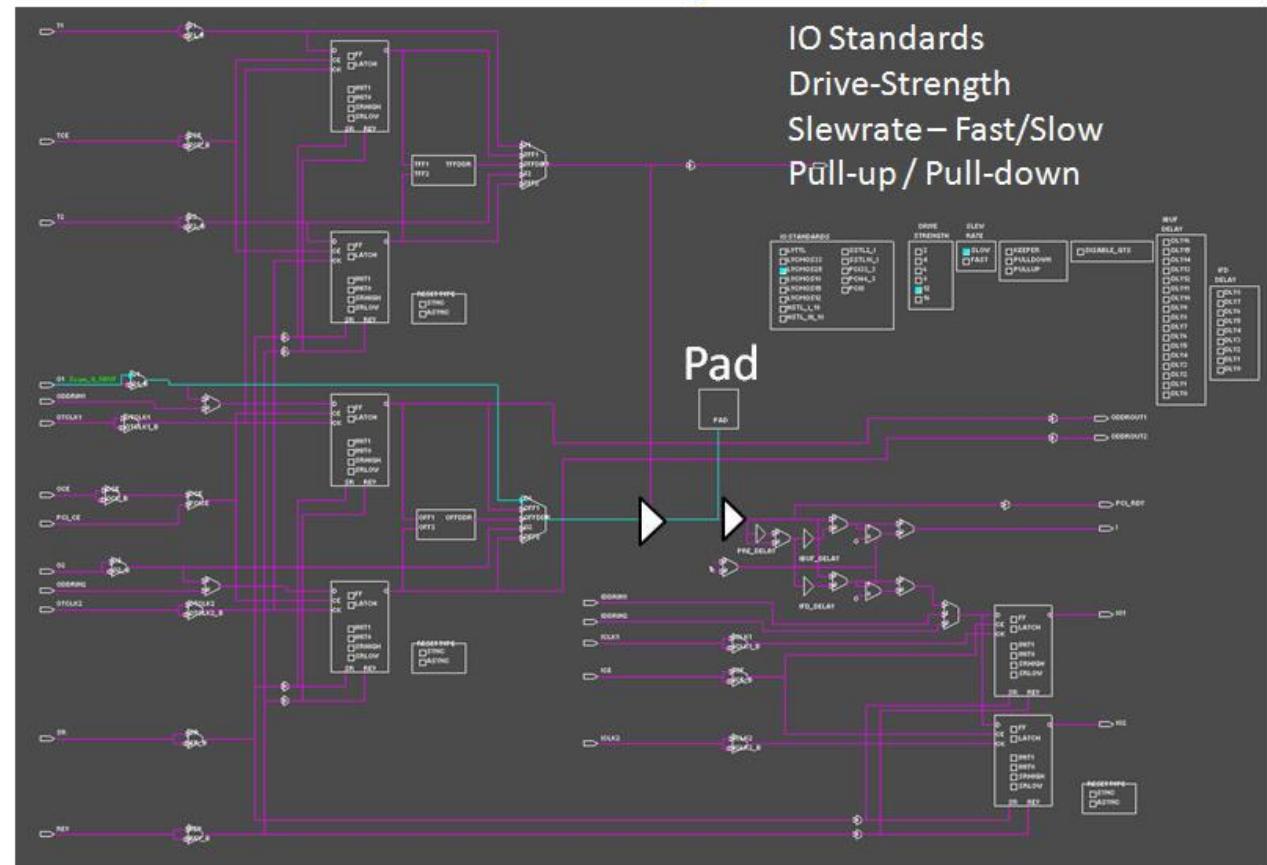


FPGAs comes in different housings
The most compact called PBGA.

PBGA = Pin Ball Grid Array

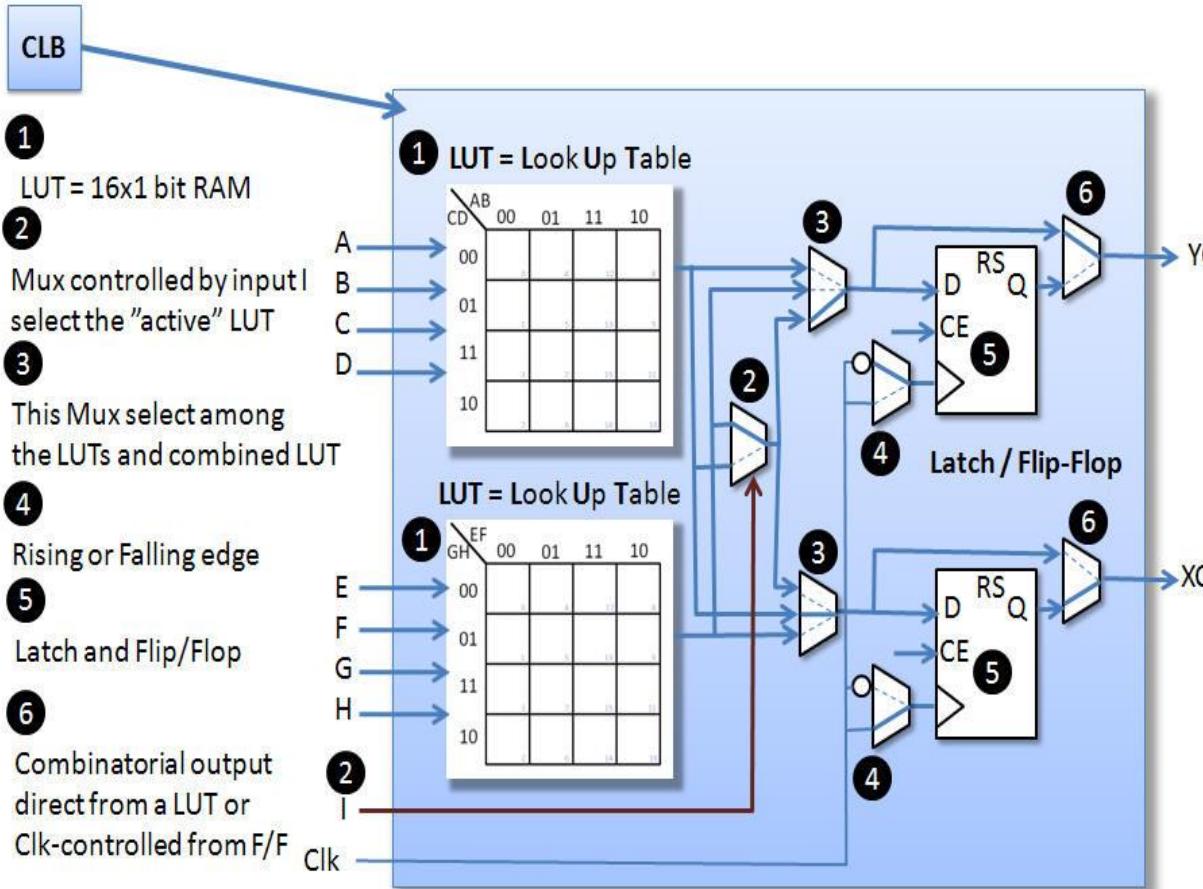


Упрощенный вид
Блока ввода-вывода (I/O-Block)

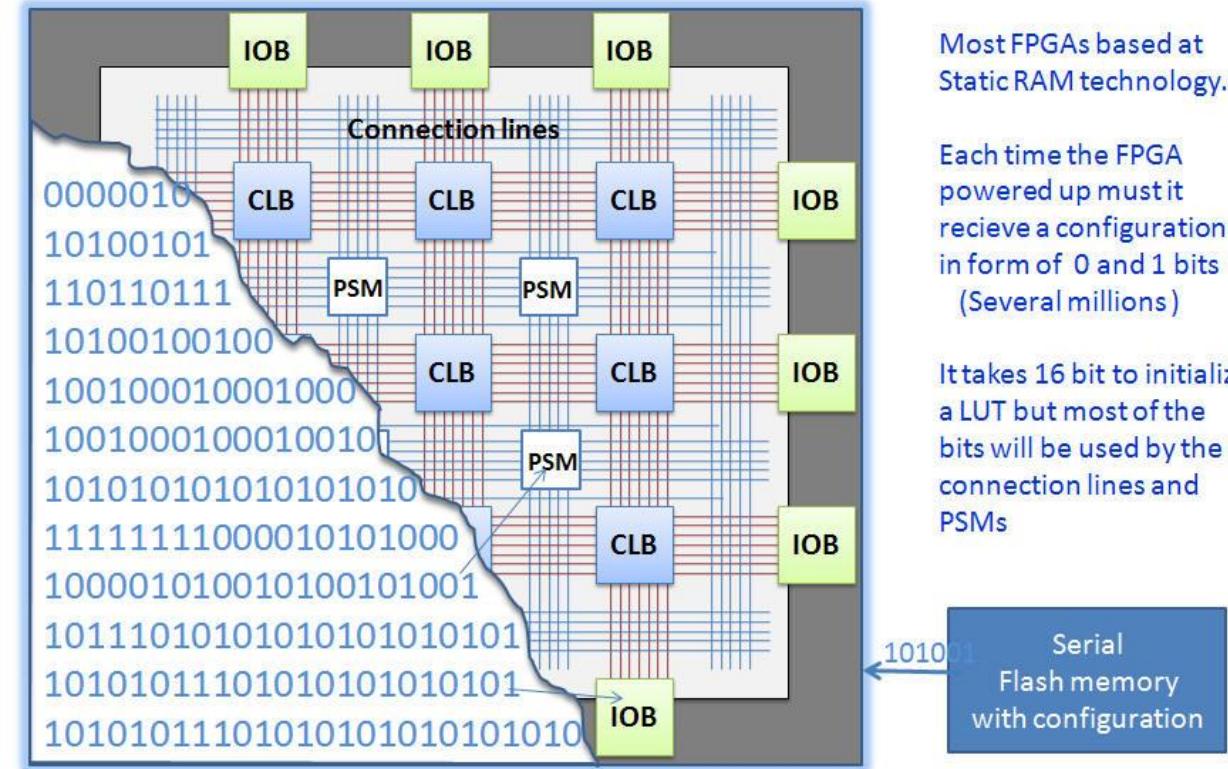


Реальный вид
Блока ввода-вывода (I/O-Block)

ПЛИС (FPGA)

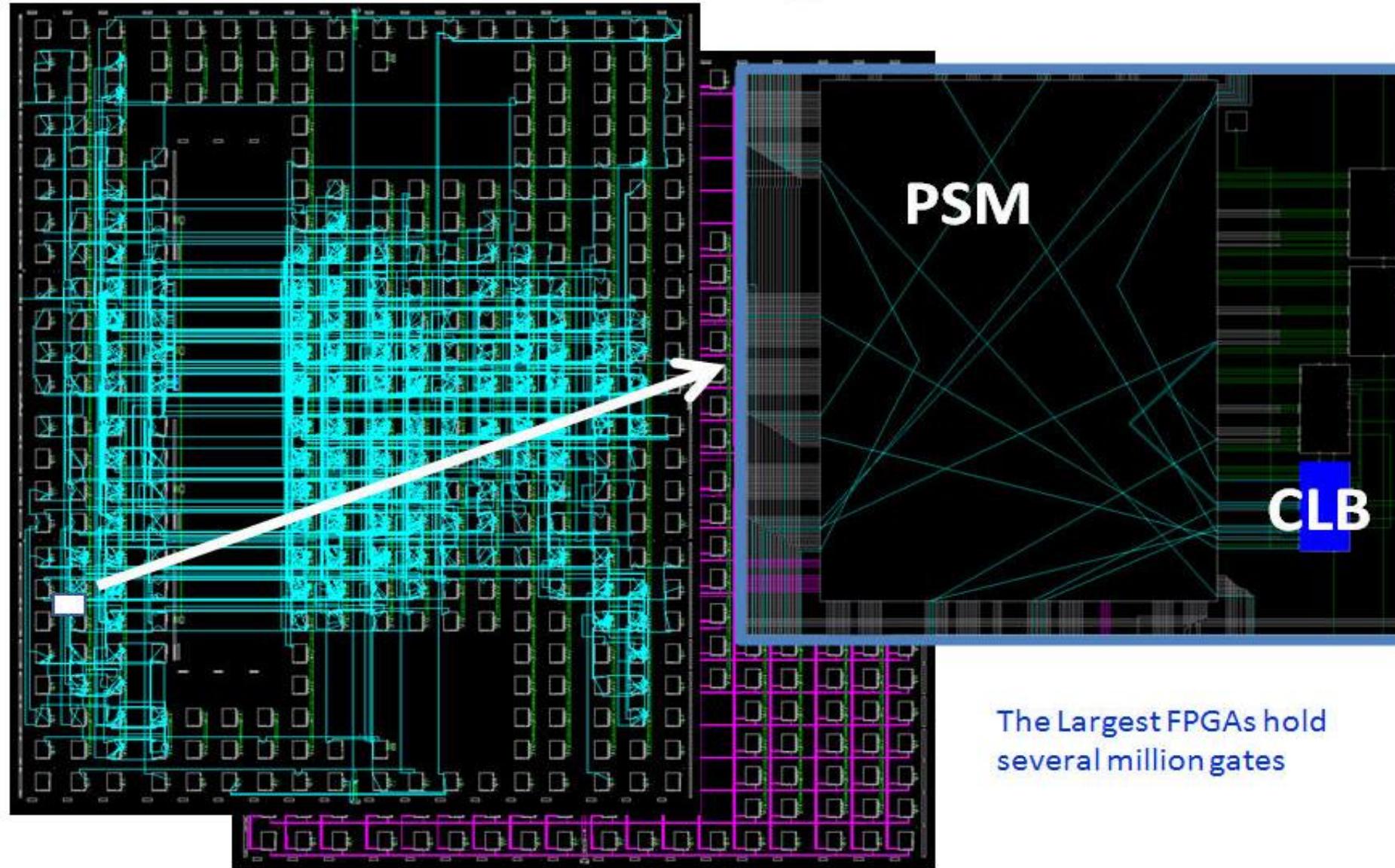


Настраиваемый логический блок



Что скрывается за ПЛИС

ПЛИС (FPGA)



The Largest FPGAs hold
several million gates

Внутри ПЛИС — 100 000 вентилей — используется 36 %

Языки программирования vs. языки описания аппаратуры: CPU

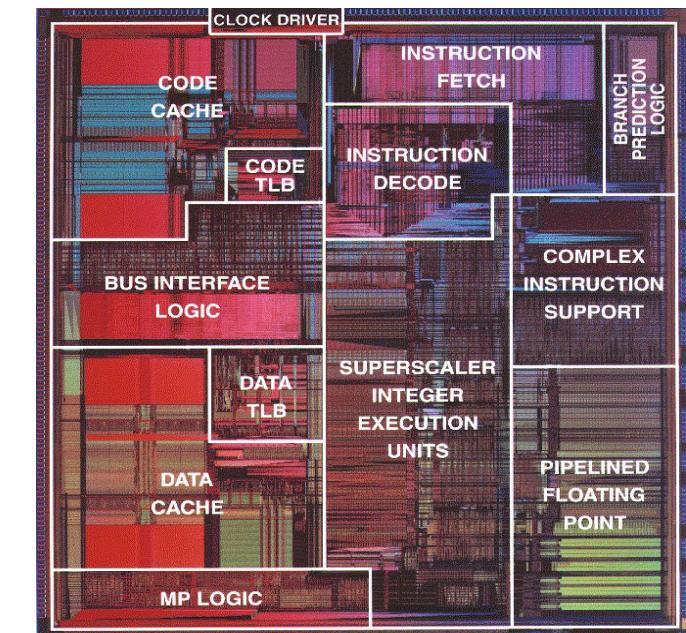
```
void add_func(const int *a,  
             const int *b, int *c)  
{  
    for (int i = 0; i < 100; i++)  
    {  
        c[i] = a[i] + b[i];  
    }  
}
```



```
add_func:  
    xor eax, eax  
.L2:  
    mov ecx, DWORD PTR  
[rsi+rax*4]  
    add ecx, DWORD PTR  
[rdi+rax*4]  
    mov DWORD PTR [rdx+rax*4],  
ecx  
    add rax, 1  
    cmp rax, 100  
    jne .L2  
    ret
```



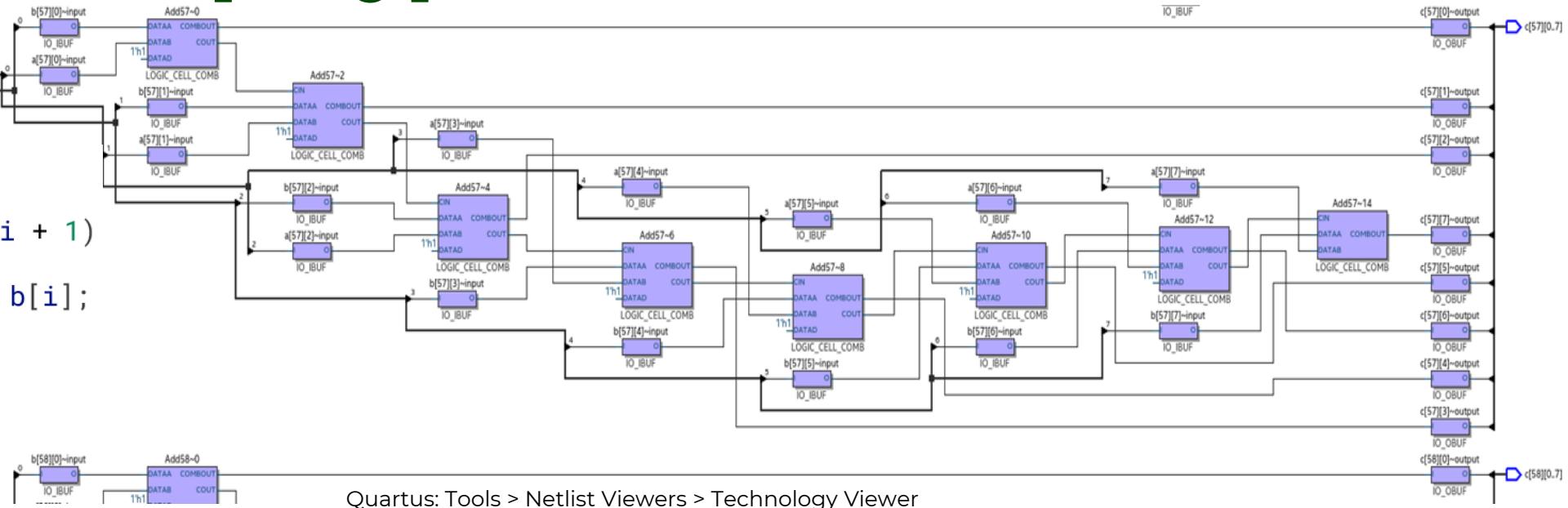
31	c0
8b	0c 86
03	0c 87
89	0c 82
48	83 c0 01
48	83 f8 64
75	ed
	c3



- Пример: сложение векторов из 100 элементов на CPU
- Код на С описывает последовательность действий
- Код на С компилируется в бинарный **код для x86 CPU**
- **Бинарный код** загружается в память CPU и исполняется
- Количество сложений в единицу времени определяется **количеством целочисленных сумматоров** в ядре CPU и **тактовой частотой**

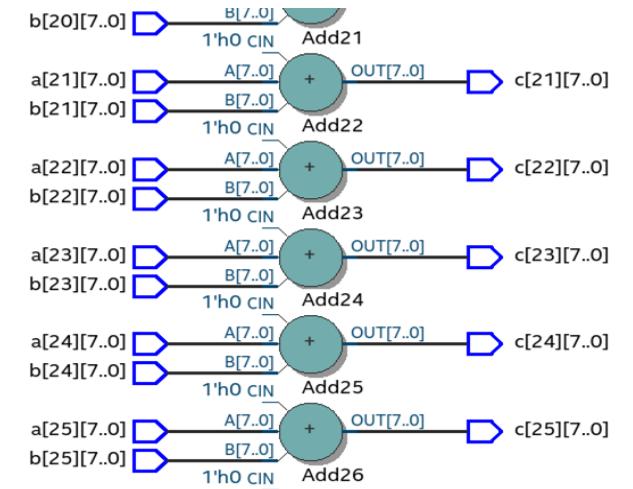
Языки программирования vs. языки описания аппаратуры: FPGA

```
generate
genvar i;
for (i = 0; i < 100; i = i + 1)
begin : gen_vec_add
    assign c[i] = a[i] + b[i];
end
endgenerate
```



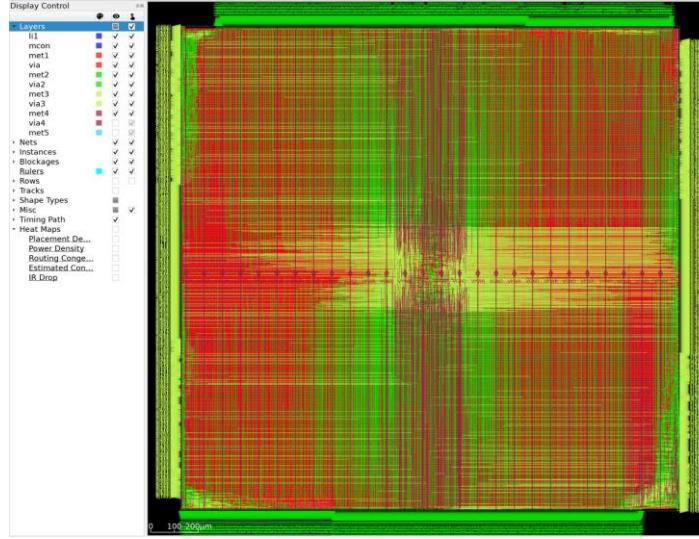
Quartus: Tools > Netlist Viewers > Technology Viewer

- **Пример: сложение векторов из 100 элементов на FPGA**
- **Код на Verilog** описывает цифровую схему
- Код на Verilog компилируется в **Netlist из логических блоков FPGA**
- Netlist компилируется в **Bitstream** для FPGA
- **Bitstream прошивается в FPGA**
- Количество сложений в единицу времени определяется **количеством использованных логических блоков FPGA и тактовой частотой**
(макс. частота зависит от свойств FPGA и свойств схемы)

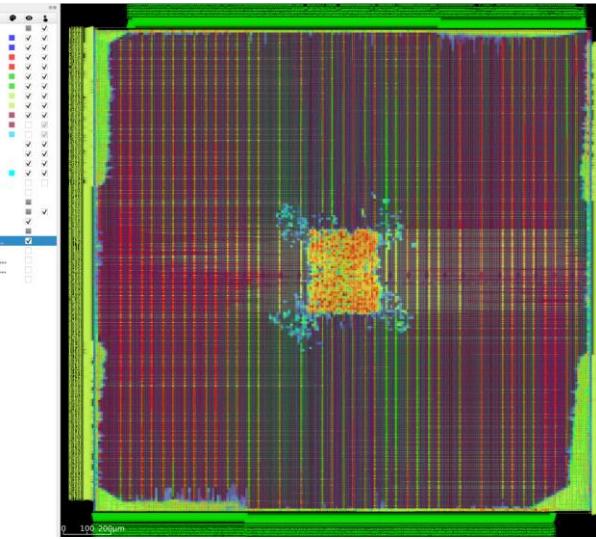


Quartus: Tools > Netlist Viewers > RTL Viewer

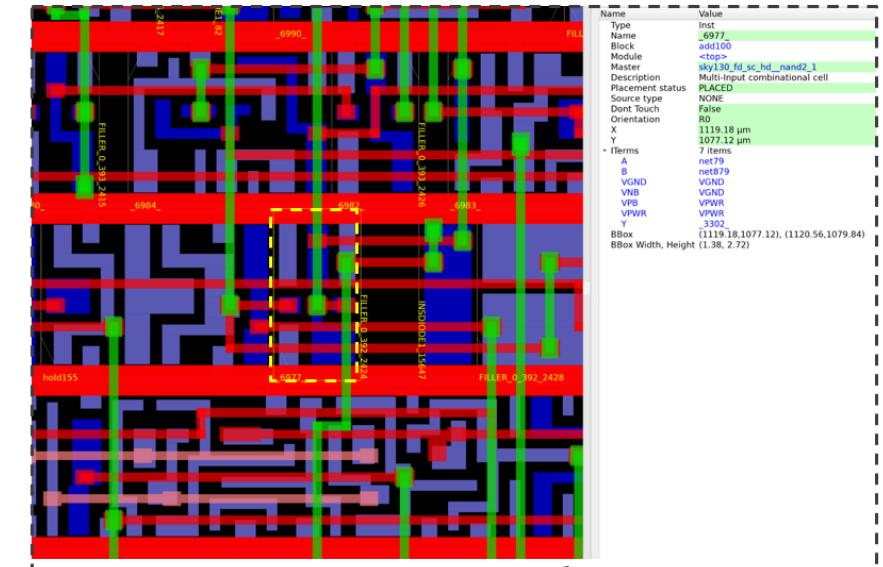
Языки программирования vs. языки описания аппаратуры: ASIC



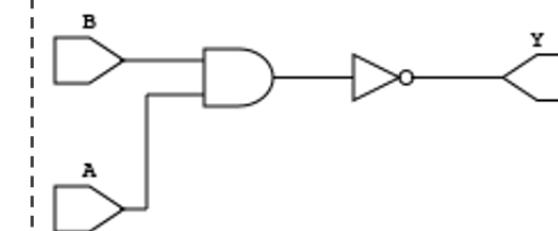
OpenLANE, Skywater 130nm PDK: ~4.8 mm²



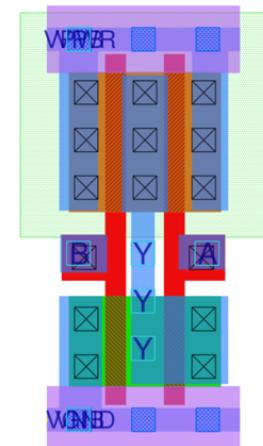
Placement Density Heat Map



Ячейка внутри блока



Принципиальная схема ячейки



Ячейка

Skywater 130nm PDK: High-density NAND2_1 cell

Средства проектирования ASIC и FPGA

• Симуляция:

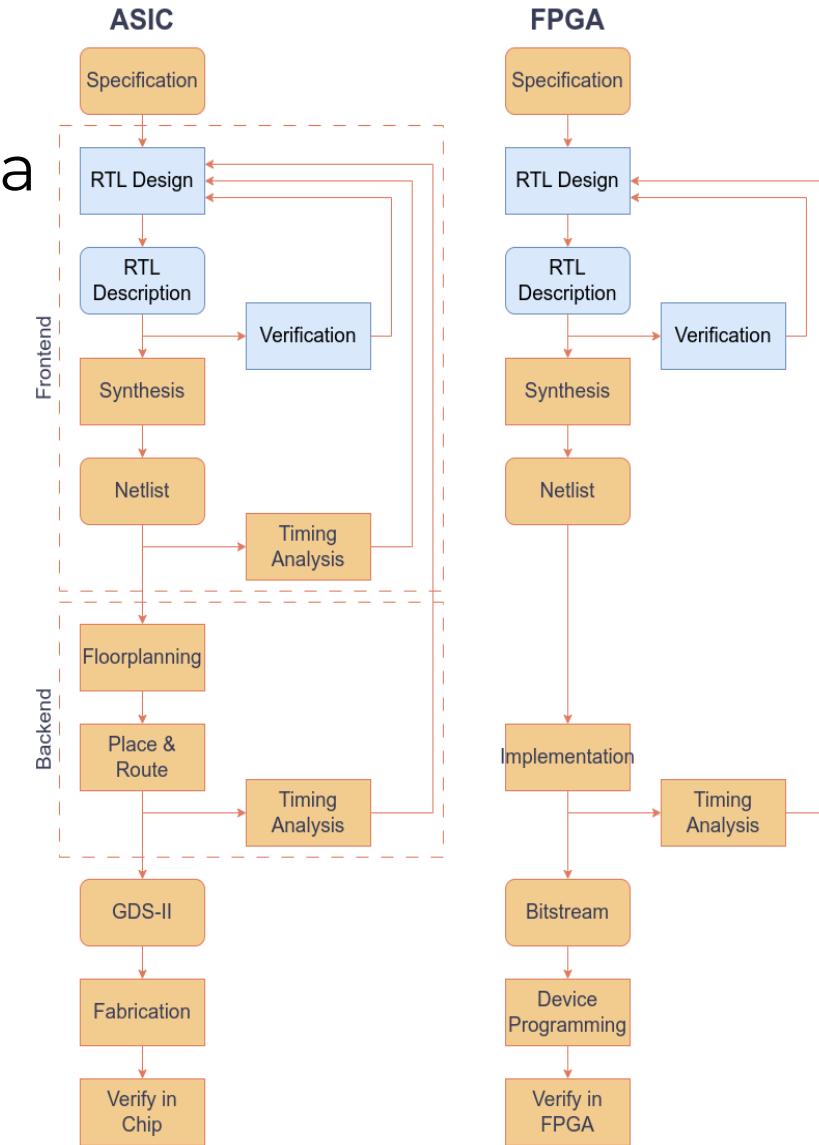
- **ASIC**: Cadence Xcelium, Synopsys VCS
- **FPGA**: Xilinx Vivado, Siemens ModelSim/Questa
- **Open Source**: Icarus Verilog, Verilator

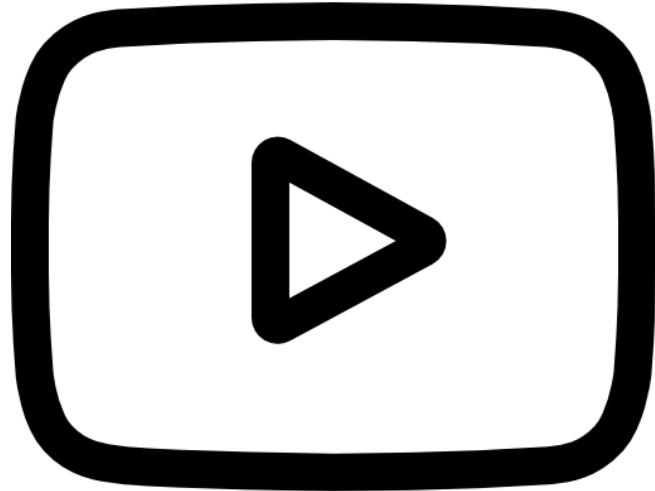
• Синтез:

- **ASIC**: Cadence Genus, Synopsys DC Compiler
- **FPGA**: Xilinx Vivado, Intel Quartus
- **Open Source**: Yosys + ABC

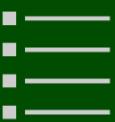
• Имплементация:

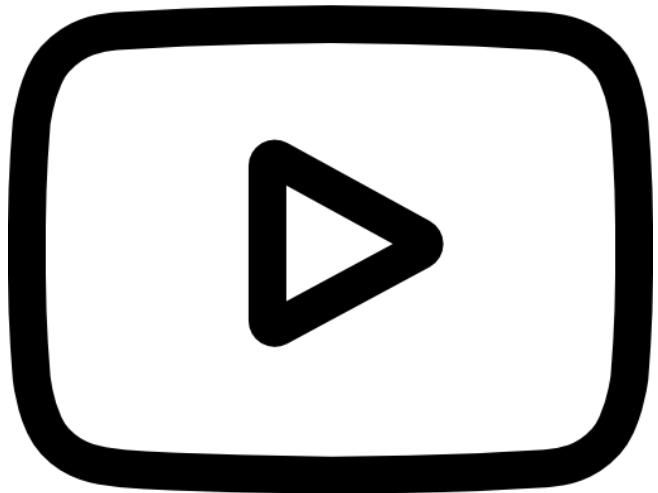
- **ASIC**: Cadence Innovus, Synopsys IC Compiler
- **FPGA**: Xilinx Vivado, Intel Quartus
- **Open Source**: OpenROAD





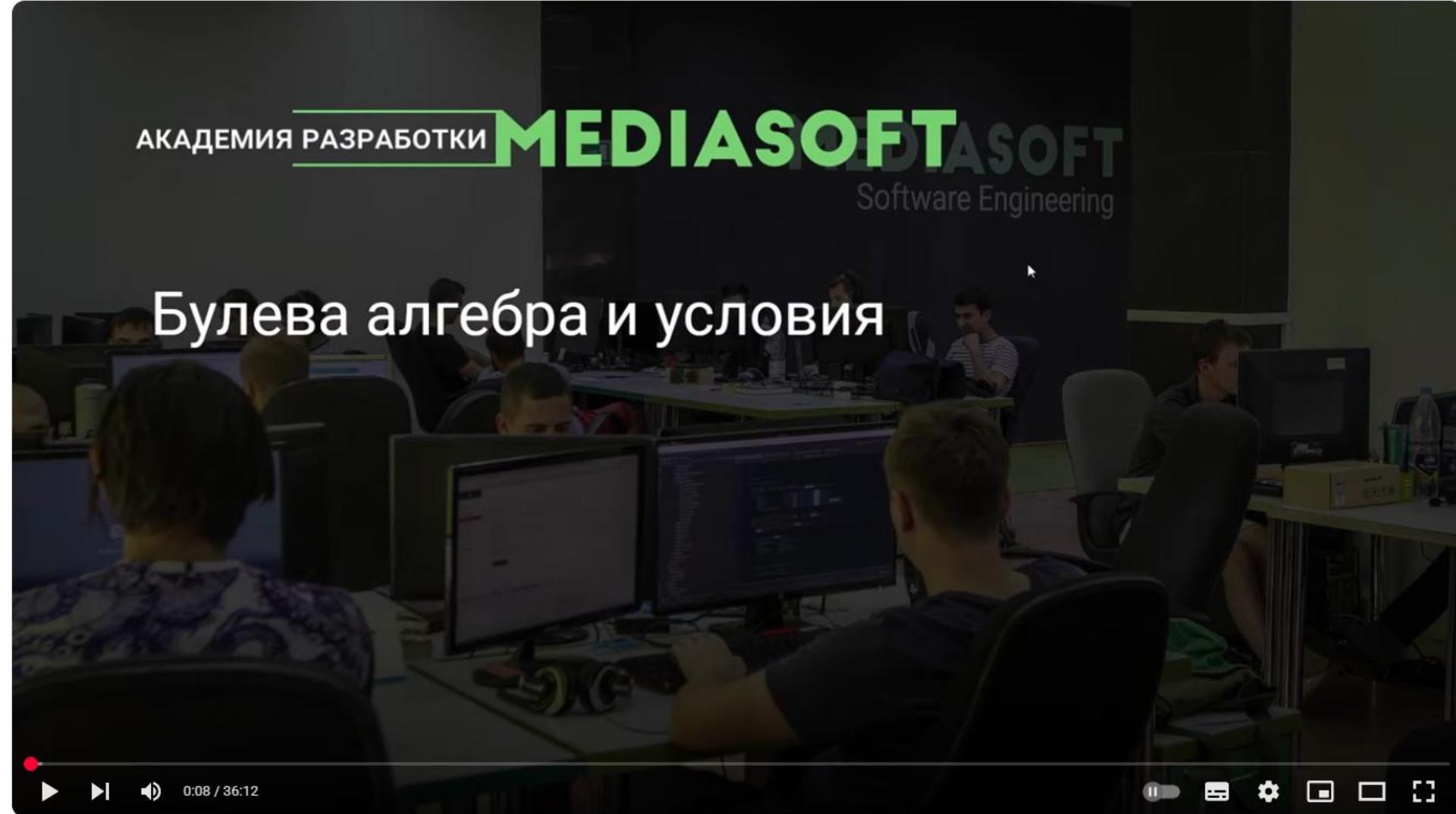
Дополнительные
материалы по
теме на YouTube





Булева алгебра





Лекция №6 / Основы программирования / Булева алгебра и условия (2021)
<https://www.youtube.com/watch?v=EDEIPk2Y5Zc>

\Rightarrow - „следовательно“ $A \Rightarrow B$ - „из утверждения A следует утверждение B“

Если из A следует B, то это не значит, что из B следует A !

\neg - „не“ (логическое „не“) $\neg A$ - „не A“ (отрицание утверждения A)

Если утверждение A истинно, то утверждение $\neg A$ (отрицание A) ложно.

Если A истинно и $A \Rightarrow B$, то B тоже истинно.

Закон тождества: в рассуждении каждое понятие и каждое суждение должны употребляться в одном и том же смысле.

The image shows a man with a beard and short brown hair, wearing a light blue button-down shirt, standing in front of a green chalkboard. He is positioned centrally, looking directly at the camera. Behind him is a chalkboard with handwritten text and several truth tables.

Handwritten text on the chalkboard includes:

- Таблицы истинности
- НЕ
- И
- ИЛИ
- Импликация
- эквив

Truth tables displayed on the chalkboard:

- NOT (НЕ):

A	$\neg A$
0	1
1	0

- AND (И):

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

- IMPLIcation (Импликация):

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

- EQUIV (эквив):

A	B	$A \Leftrightarrow B$
0	0	1
0	1	0
1	0	0
1	1	1

- OR (ИЛИ):

A	B	$A \vee B$
0	0	0
0	1	1

- XOR ($A \oplus B$):

A	B	$A \oplus B$
0	0	0
0	1	1

Информатика. Алгебра логики: Таблицы истинности (2014)
<https://www.youtube.com/watch?v=qri6Ekwqr-c>

Законы алгебры логики

Простые:

$$\begin{array}{lll} A \cdot A = A & A + A = A & A \cdot (A + B) = A \\ A \cdot A^{\perp} = 0 & A + A^{\perp} = 1 & A + A \cdot B = A \end{array}$$

Свойства операндов:

$$\begin{array}{lll} A \cdot B = B \cdot A & B = B + A & \\ A \cdot (B \cdot C) = (A \cdot B) \cdot C & A + (B + C) = (A + B) + C & \\ A \cdot (B + C) = A \cdot B + A \cdot C & B \cdot C = (A + B) \cdot (A + C) & \end{array}$$

Законы отрицания:

$$\begin{array}{lll} \overline{\overline{A}} = A & \overline{A + B} = \overline{A} \cdot \overline{B} & \overline{A \cdot B} = \overline{A} + \overline{B} \\ A + \overline{A} = 1 & & \end{array}$$

Представление

$$\begin{array}{ll} A \rightarrow B = \overline{A} + B & A \rightarrow B^{\perp} = A \cdot B^{\perp} \\ \text{XOR}, \rightarrow, \Leftrightarrow: & A \cdot \overline{B} + \overline{A} \cdot B \end{array}$$

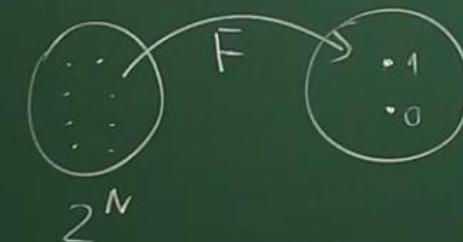
Алгебра логики: Законы алгебры логики (2014)
<https://www.youtube.com/watch?v=7XA77xNVBv4>



Информатика. Алгебра логики: Теория множеств (2014)
<https://www.youtube.com/watch?v=LVDZaTmaLWw>

Логические функции

$$F(x, y, z) = x \rightarrow y \vee \neg z$$



$$F(x, y, z) = \neg x \vee y \vee \neg z$$

x	y	z	F(x, y, z)
0	0	0	1
0	1	0	1
0	1	1	1
1	0	0	1
1	1	1	0
1	0	1	1
1	1	0	1



▶ ▶ ⏪ 6:18 / 8:18 ⏩ ⏹ ⏷ ⏸ ⏹ ⏺ ⏻

Алгебра логики: Логические переменные и логические функции (2014)
<https://www.youtube.com/watch?v=Ys8b60jCZUQ>

Белорусско-Российский университет, Кафедра «Программное обеспечение информационных технологий»

Законы алгебры логики

3 группы равносильностей:

- 1. $x \leftrightarrow y = (x \rightarrow y) \wedge (y \rightarrow x)$
- 2. $x \rightarrow y = \bar{x} \vee y$
- 3. $\bar{x \wedge y} = \bar{x} \vee \bar{y}$
- 4. $\bar{x \vee y} = \bar{x} \wedge \bar{y}$
- 5. $x \wedge y = \bar{\bar{x}} \wedge \bar{\bar{y}}$
- 6. $x \vee y = \bar{\bar{x}} \vee \bar{\bar{y}}$

II чп Равносильности, выраж. одни лог. операции через другие

III чп Равносильности, выраж. син. закончи алгебры логики.

1. $x \vee y = y \vee x$ Коммутативность дизьюнкции

2. $x \wedge y = y \wedge x$ Коммутативность конъюнкции (переместительный закон)

3. $x \vee (y \vee z) = (x \vee y) \vee z$ Ассоциативность диз

4. $x \wedge (y \wedge z) = (x \wedge y) \wedge z$ Ассоциативность кон (согласительный закон)

5. $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ - дистрибутивность кон

6. $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ - дистрибутивность кон (распределительный закон)

$B \vee \bar{B} = A \wedge 1 = A$

$x \rightarrow (y \rightarrow x) = \bar{x} \vee (\bar{y} \vee x) = \bar{y} \vee 1 = 1$

$\neg y \wedge \bar{y} = \bar{x} \wedge \bar{\bar{x}} \vee y \wedge \bar{y} = \bar{x} \wedge \bar{x} \vee 0 = \bar{x} \wedge 0 = 0$

Законы алгебры логики (2020)
<https://www.youtube.com/watch?v=sByAeOXgMPI>

Построить таблицу истинности: 3 2 1

$$A \& (B \vee \bar{B})$$

A	B	\bar{B}	$B \vee \bar{B}$	$A \& B \vee \bar{B}$
0	0	1	1	0
0	1	0	1	0
1	0	1	1	1
1	1	0	1	1



Построение таблиц истинности (2020)
<https://www.youtube.com/watch?v=n944UR8ENz0>

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



$$\begin{aligned} & (\overline{A+B}) + (\overline{A+B}) + A \cdot B = \bar{A} \cdot \bar{B} + \bar{A} \cdot \bar{B} + AB = \\ & \underline{\bar{A}B} + \underline{\bar{A} \cdot \bar{B}} + \underline{AB} = \bar{A} (\underline{B + \bar{B}}) + AB = \bar{A} + AB = \\ & \underline{(\bar{A} + A)} \underline{(\bar{A} + B)} = \bar{A} + B \end{aligned}$$

Закон	Для «И»	Для операции «ИЛИ»
двойного отрицания	$\bar{\bar{A}} = A$	
исключённого третьего	$A \cdot \bar{A} = 0$	$A + \bar{A} = 1$
операции с константами	$A \cdot 1 = A, A \cdot 0 = 0$	$A + 1 = 1, A + 0 = A$
повторения	$A \cdot A = A$	$A + A = A$
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	$A + B \cdot C = (A + B) \cdot (A + C)$	$A \cdot (B + C) = A \cdot B + A \cdot C$
поглощения	$A + A \cdot B = A$	$A \cdot (A + B) = A$
де Моргана	$\overline{A \cdot B} = \bar{A} + \bar{B}$	$\overline{A + B} = \bar{A} \cdot \bar{B}$



Упрощение логических выражений. К.Ю.Поляков, 10 класс, параграф 18 (2023)
<https://www.youtube.com/watch?v=LgcPbKD500w>

Булева алгебра

Юрий Григорьев

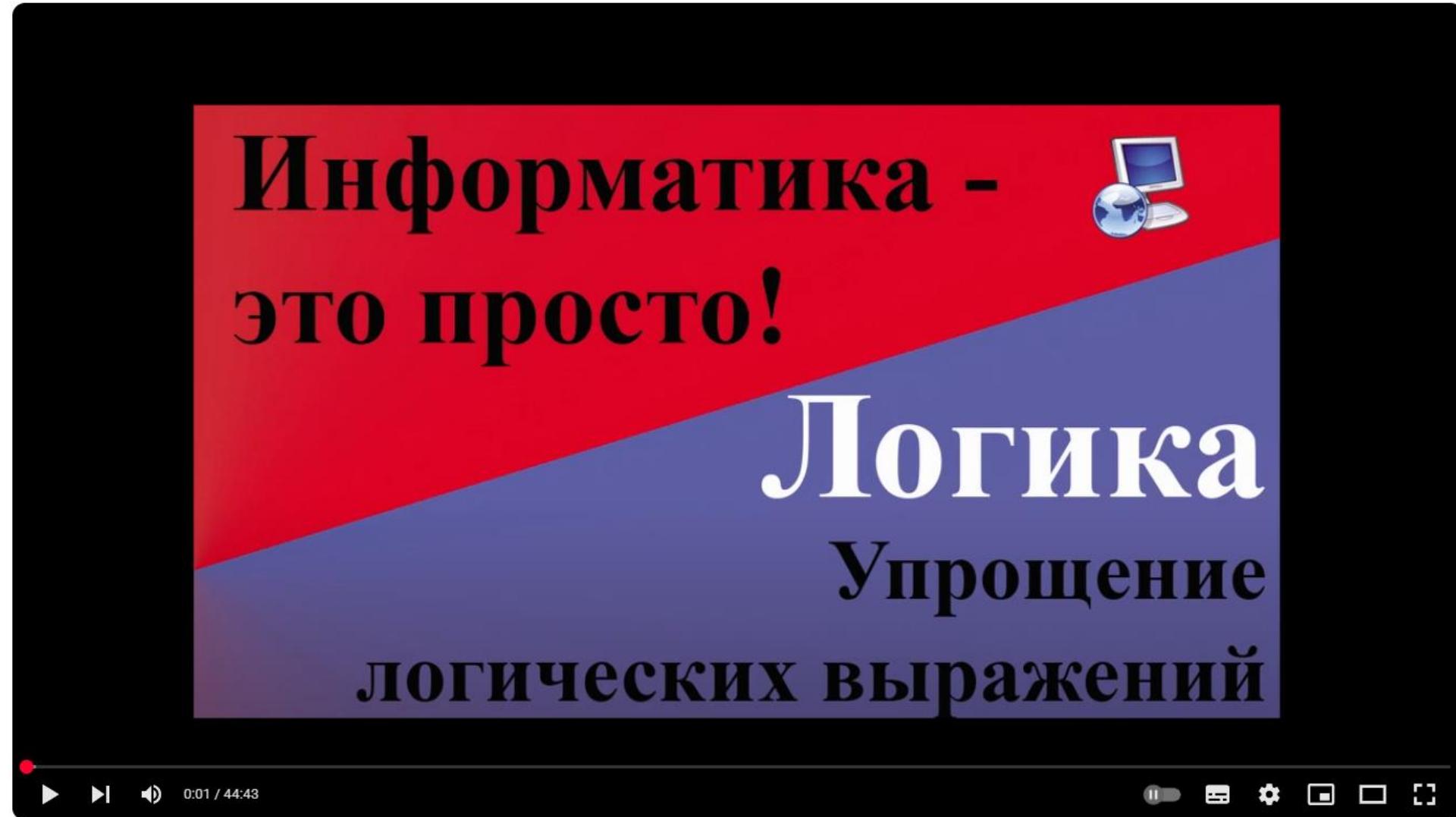
Плейлист - 18 видео · 33 355 просмотров

▶ Воспроизвести в...

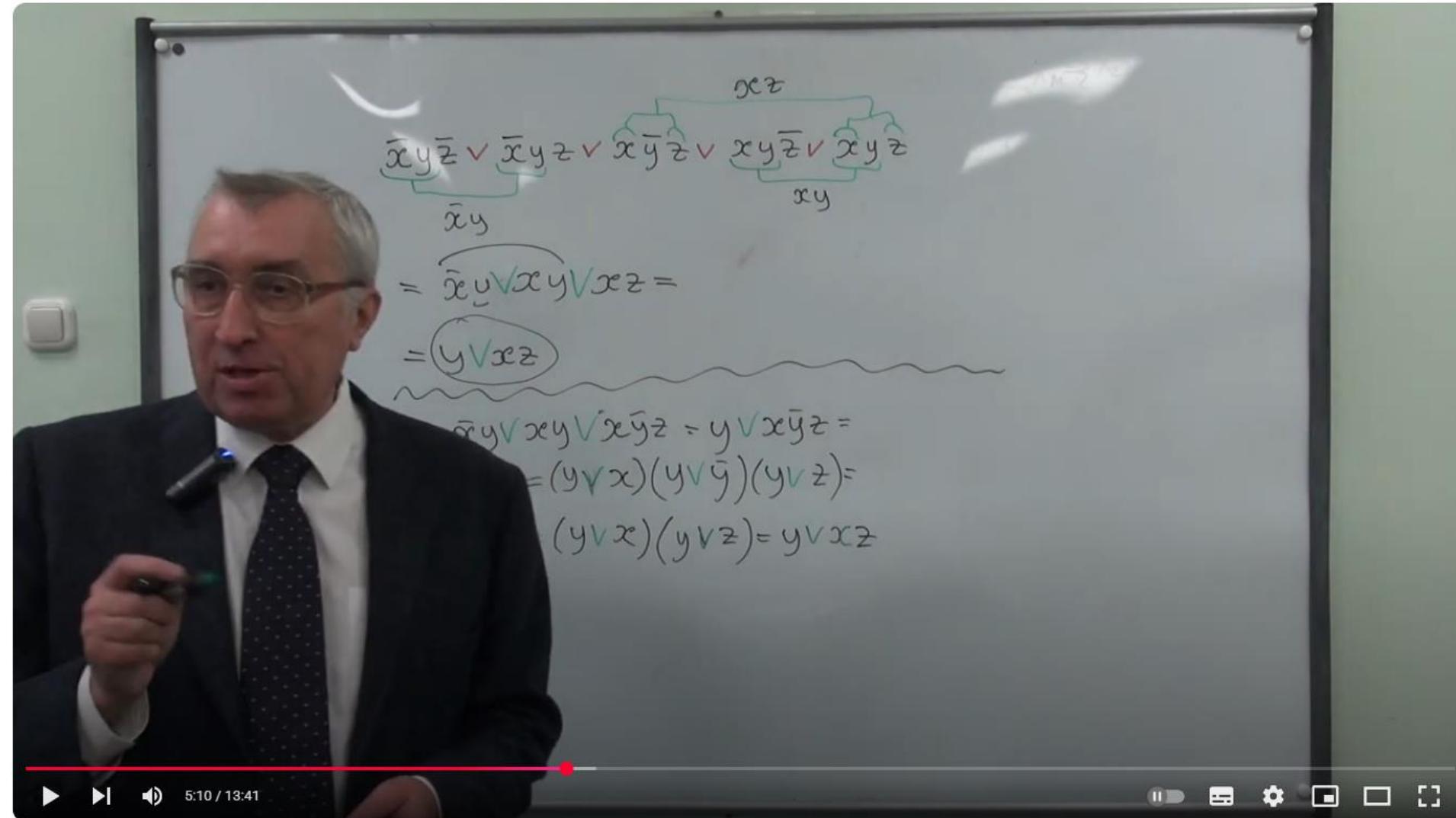
- 1 **Булевые функции**
1:14:14
- 2 **Совершенные формы булевых функций**
Юрий Григорьев · 7 лет назад · 10 тыс. просмотров
- 3 **СДНФ булевых функций**
Юрий Григорьев · 7 лет назад · 8 тыс. просмотров
- 4 **Совершенная конъюнктивная нормальная форма**
Юрий Григорьев · 7 лет назад · 6,4 тыс. просмотров

Плейлист: Булева алгебра

<https://www.youtube.com/playlist?list=PLzlnMoWvt7wPYCiGGb9hV7sooSBcY9ytd>



Логика - Упрощение логических выражений. Законы алгебры логики (2018)
<https://www.youtube.com/watch?v=sNI5dB8I1qc>



Три способа упрощения логической функции (2014)
<https://www.youtube.com/watch?v=5U4P56A2ePY>



Минимизация логических функций. Карты Карно (2022)
<https://www.youtube.com/watch?v=YinX3OltpPE>

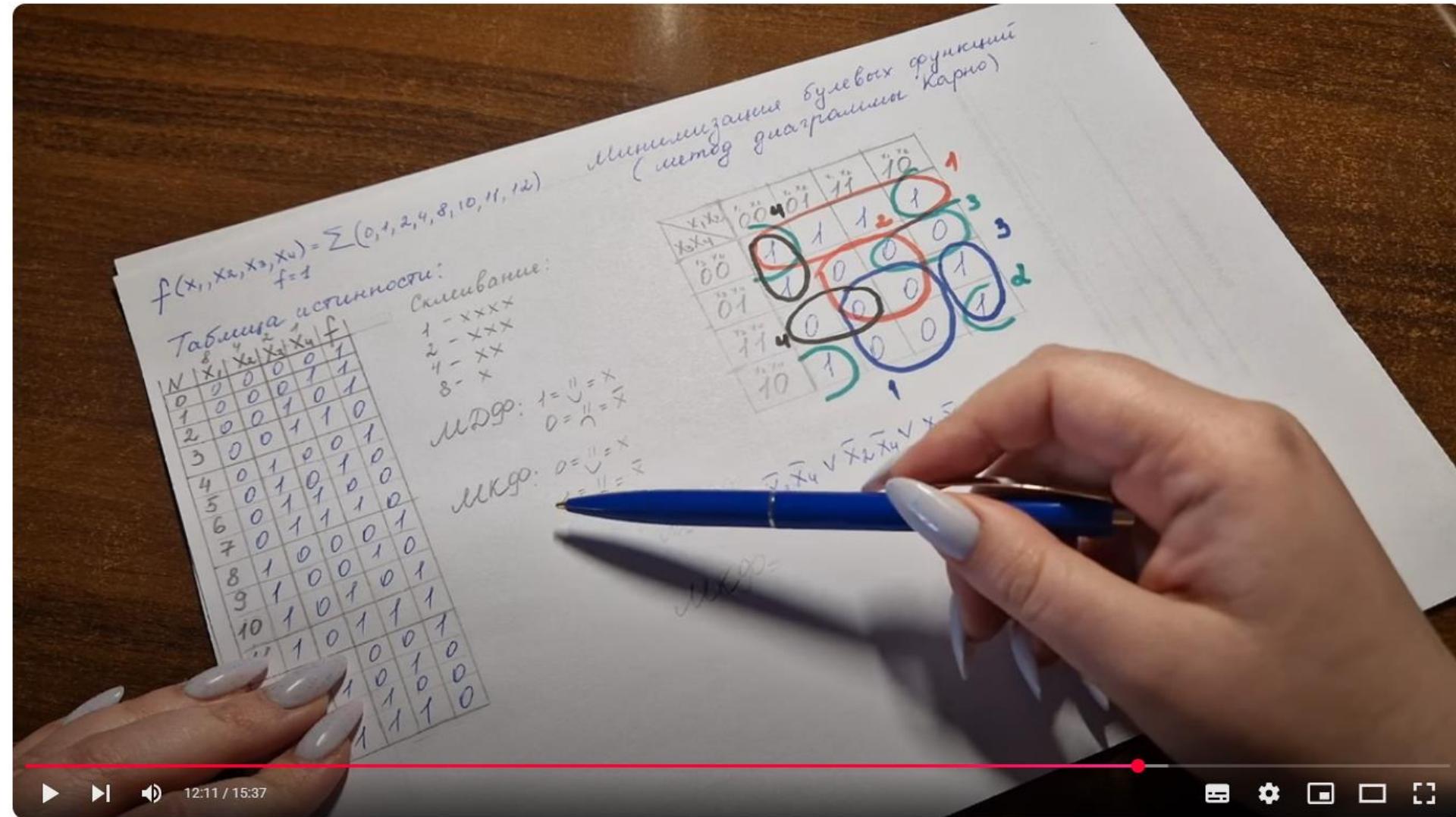
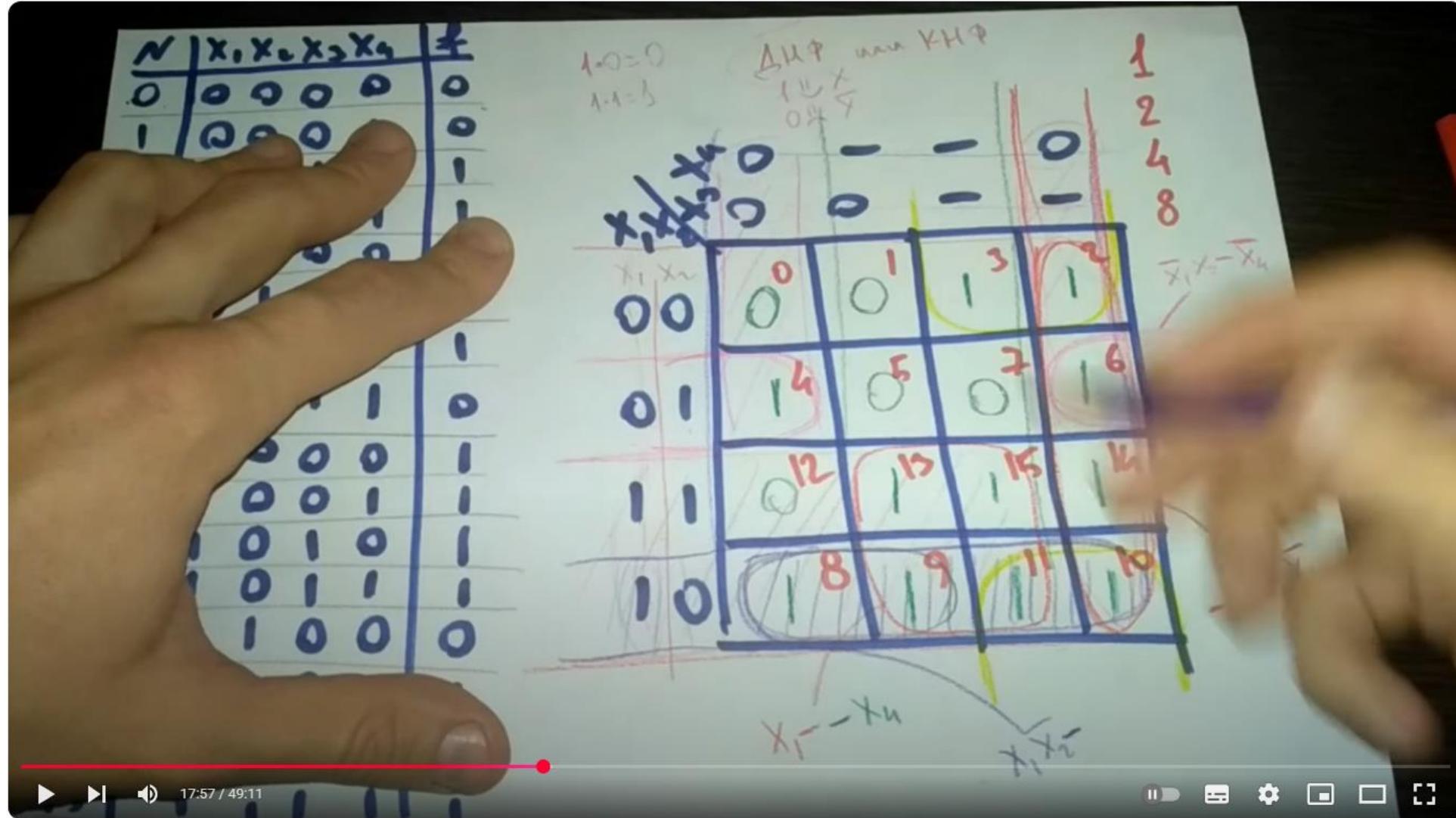
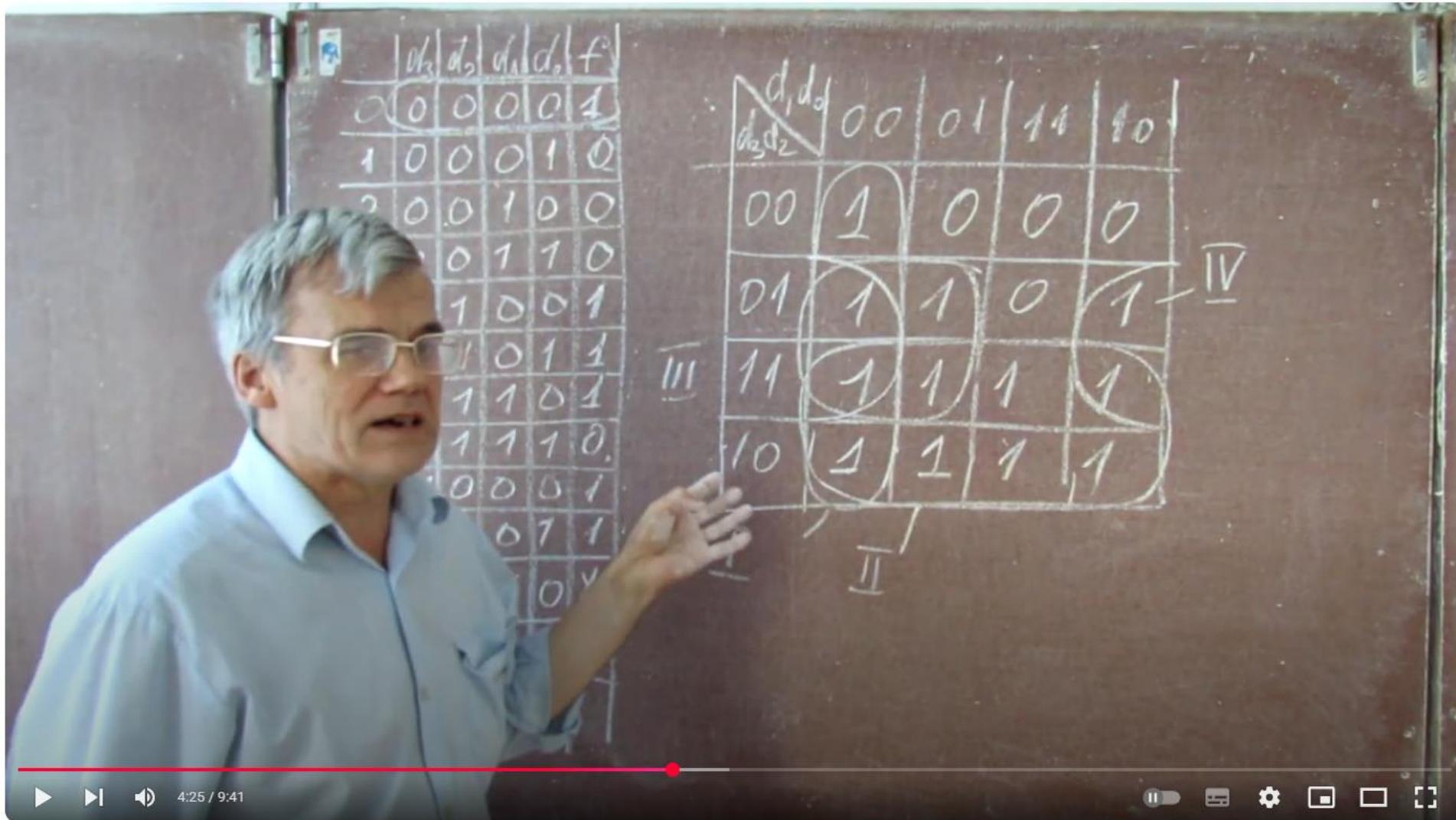


Диаграмма Карно. Минимизация булевых (логических) функций (2023)
<https://www.youtube.com/watch?v=dsNfRQ-aONU>



Карты Карно. Как они работают. Большой выпуск (2019)
<https://www.youtube.com/watch?v=wIEiX9R0SoE>



Лекция 80. Карта Карно (2013)
<https://www.youtube.com/watch?v=a37anDvo0bs>

Что такое она нужна

а чем

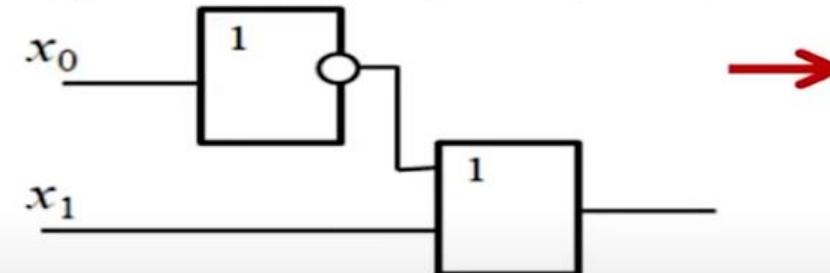
За счёт тождественных преобразований исходную функцию

$$y = \overline{x_1} \cdot \overline{x_0} \vee x_1 \cdot \overline{x_0} \vee x_1 \cdot x_0$$

преобразовали к виду

$$y = x_1 \vee \overline{x_0}$$

По данной функции строим схему ЦУ в базисе «И», «ИЛИ», «НЕ»



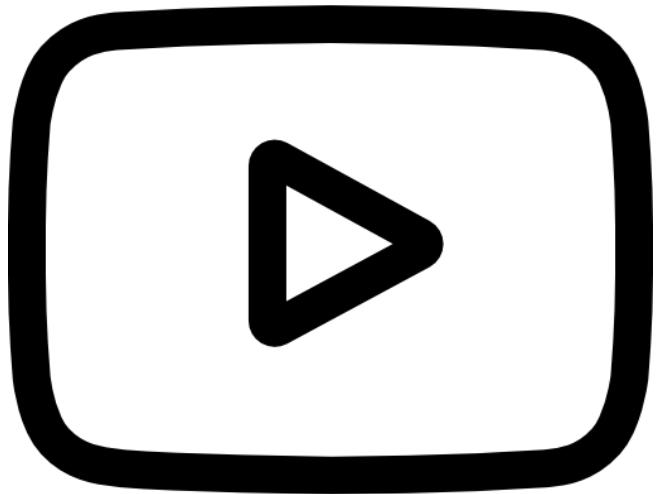
**2 логических элемента
вместо 6!**

Минимизация – процесс сокращения количества используемых логических функций для описания работы ЦУ

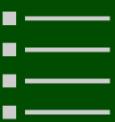
Каноническая задача минимизации – нахождение минимальной ДНФ, (ДНФ, в которую входит минимальное число аргументов)



Цифровые устройства и программируемые логические интегральные схемы.
Лекция №5 (22.02.2022) (2022) https://www.youtube.com/watch?v=dw2bt-Qm_wg



Комбинационная логика

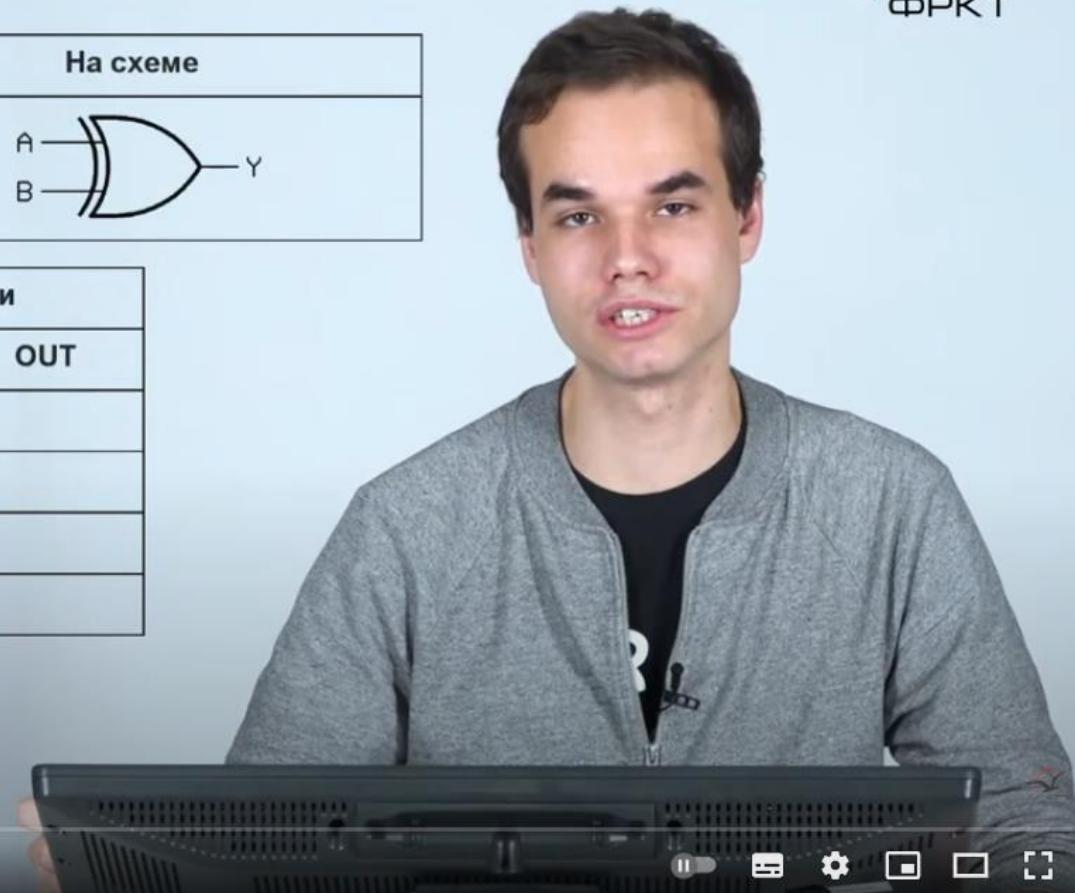


Логический элемент XOR

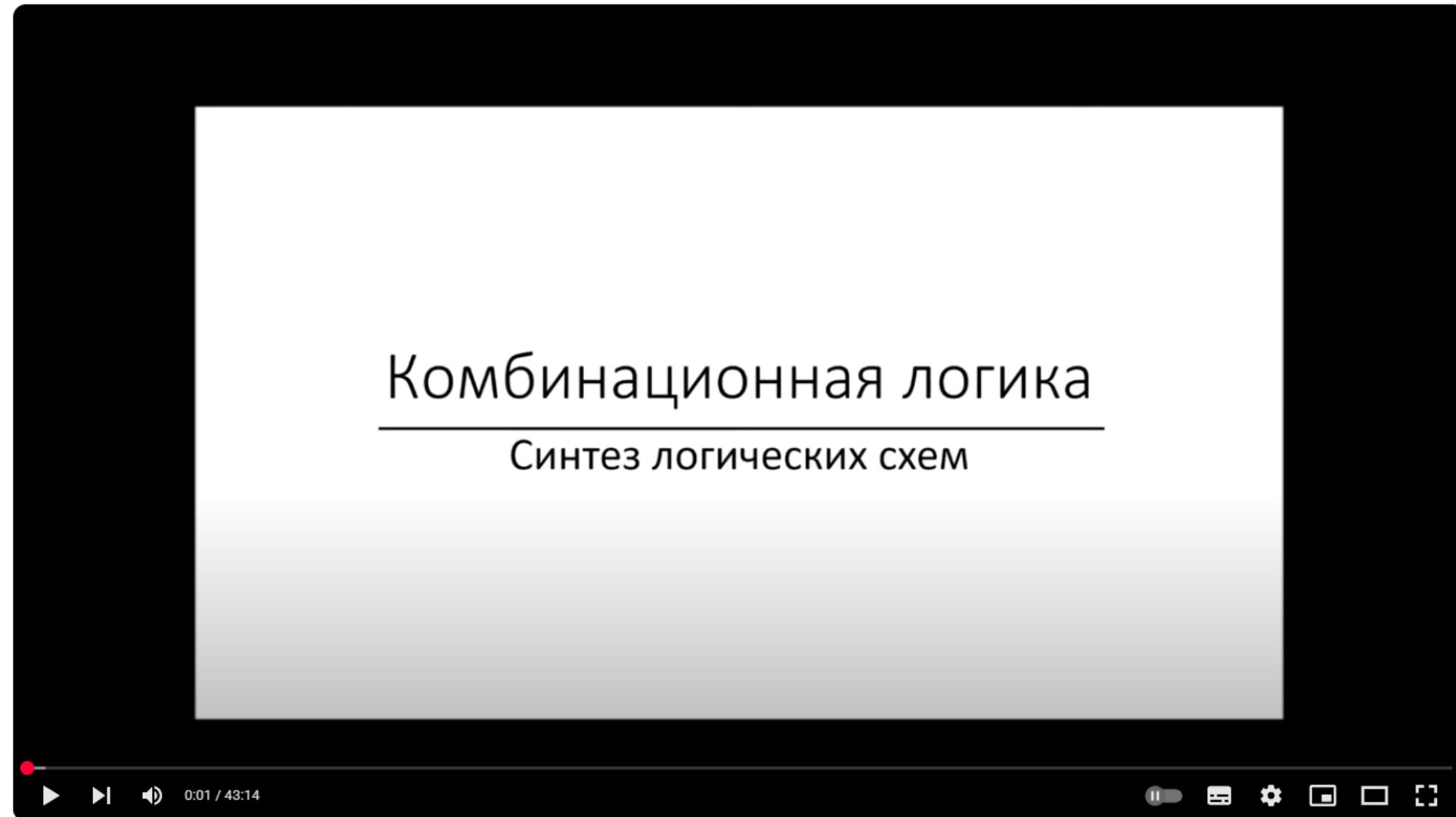


В языке Verilog	На схеме
$OUT = A \wedge B$	

Таблица истинности		
A	B	OUT
0	0	
0	1	
1	0	
1	1	



Verilog. Комбинационная логика (2019)
<https://www.youtube.com/watch?v=pMrJkHk10wU>

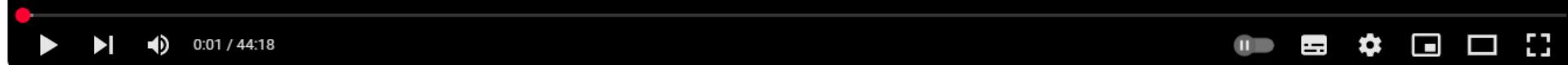


Комбинационная логика (2021)
<https://www.youtube.com/watch?v=DExYPdT2Bkk>

Комбинационная логика

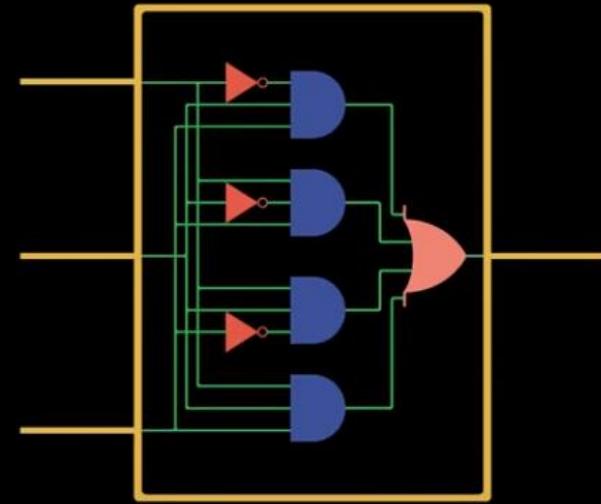
Синтез логических схем

Часть 2

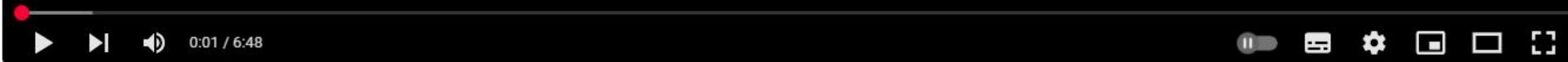


Комбинационная логика 2 (2021)
<https://www.youtube.com/watch?v=AAzkPNxO4l8>

ОСНОВЫ ЦИФРОВОЙ ЭЛЕКТРОНИКИ



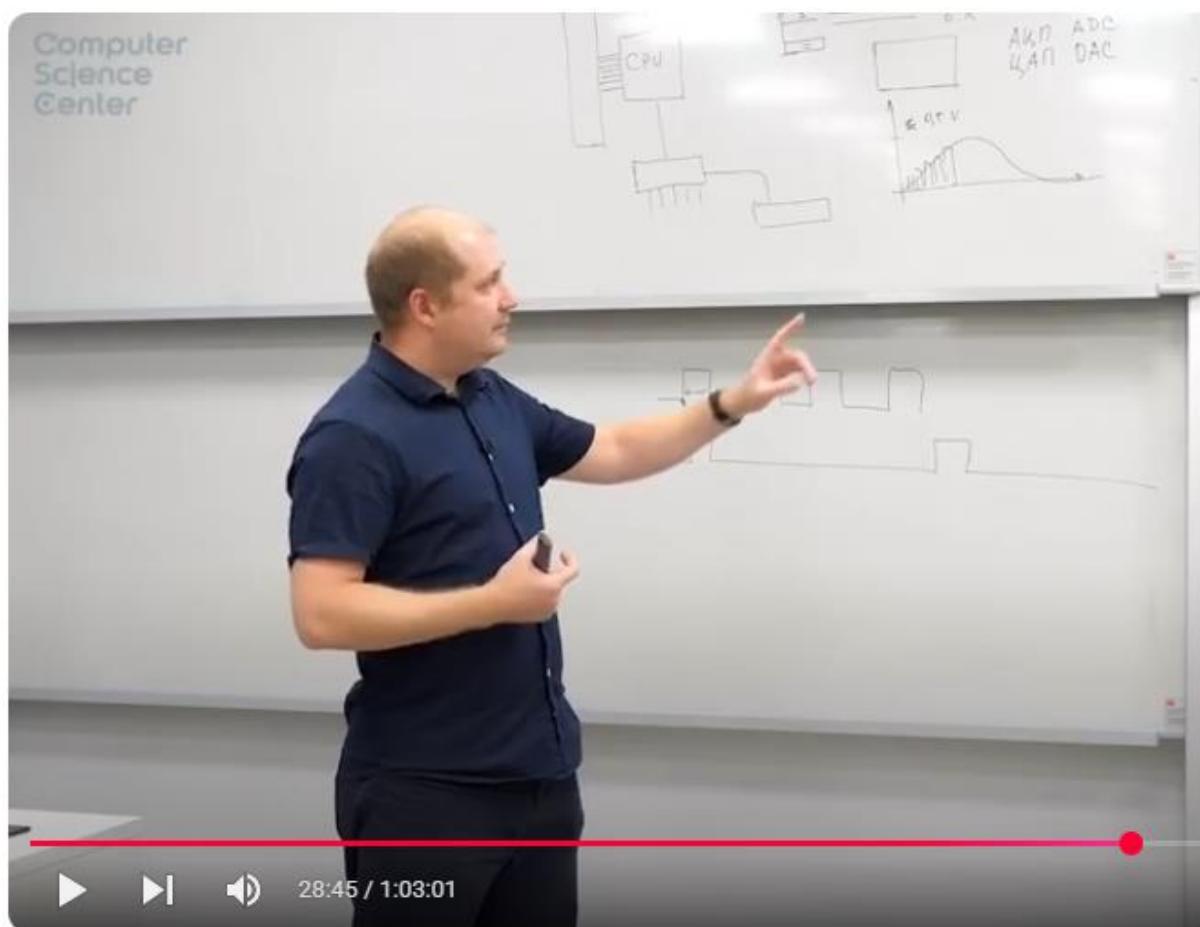
4. Комбинационная схема



Основы Цифровой Электроники - 4. Комбинационная схема (2025)
<https://www.youtube.com/watch?v=CVx91ZGp4tc>



Лекция Александра Силантьева “Комбинационная логика и её описание на языке Verilog” (ChipEXPO-2021) (2021) <https://www.youtube.com/watch?v=hkZT8-uXxpw>



Базовые блоки комбинационной и
последовательной логики
(H&H 1.5, 2.8, 3.2)

Цифровая схемотехника
и архитектура компьютера
второе издание
Дэвид М. Харрис и Сара Л. Харрис

Архитектура ЭВМ. Лекция 1: Типы архитектур. Комбинационная и последовательная логика (2021) <https://www.youtube.com/watch?v=R-twnbnaueE>

The screenshot shows a YouTube video player. On the left, there is a video frame showing a man with glasses and a beard, wearing a blue polo shirt, standing behind a white podium and speaking. On the right, the video title is displayed in large black text: "ВВЕДЕНИЕ В МАРШРУТ ПРОЕКТИРОВАНИЯ И УПРАЖНЕНИЯ С КОМБИНАЦИОННОЙ ЛОГИКОЙ". Below the title, the speaker's name is given: "Александр Силантьев" and his position: "Руководитель лаборатории НИЛ ЭСК Университета МИЭТ". At the bottom of the video frame, there is a red banner with the text "школа синтеза цифровых схем" and a logo. The YouTube player interface includes a progress bar at 9:50 / 1:35:39, a control bar with play/pause, volume, and settings icons, and a large watermark for "школа синтеза цифровых схем" in the background.

ВВЕДЕНИЕ В МАРШРУТ
ПРОЕКТИРОВАНИЯ
И УПРАЖНЕНИЯ
С КОМБИНАЦИОННОЙ
ЛОГИКОЙ

Александр Силантьев
Руководитель лаборатории НИЛ ЭСК
Университета МИЭТ

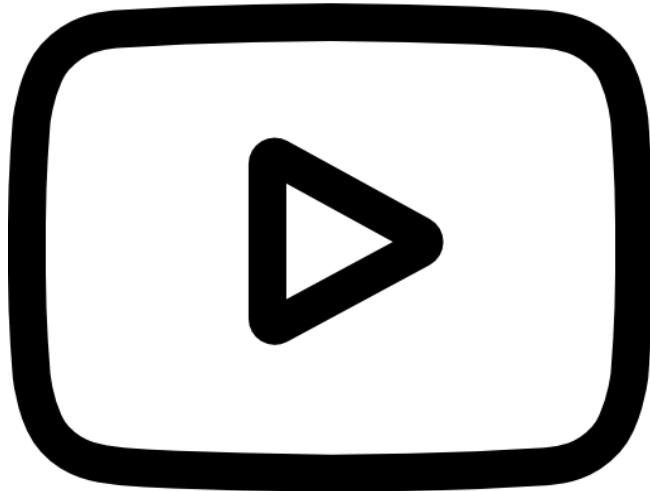
школа синтеза
цифровых схем

9:50 / 1:35:39

ПРИ ПАРТНЕРСТВЕ

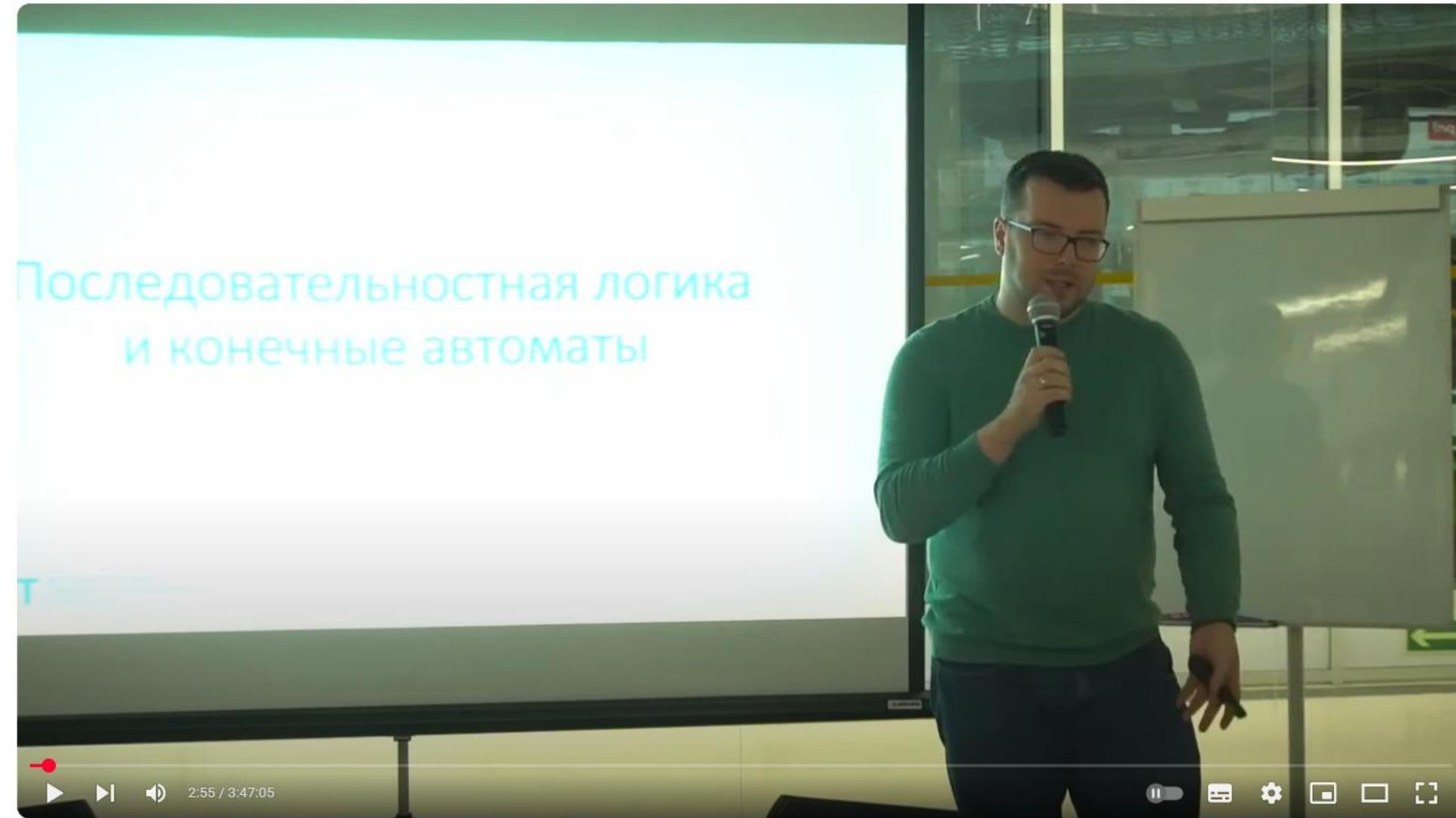
школа синтеза
цифровых схем

Занятие 1 (2023-24): Введение в маршрут проектирования и упражнения с комбинационной логикой (2023) <https://www.youtube.com/watch?v=DFcvEO-gP0c>



Последовательностная логика

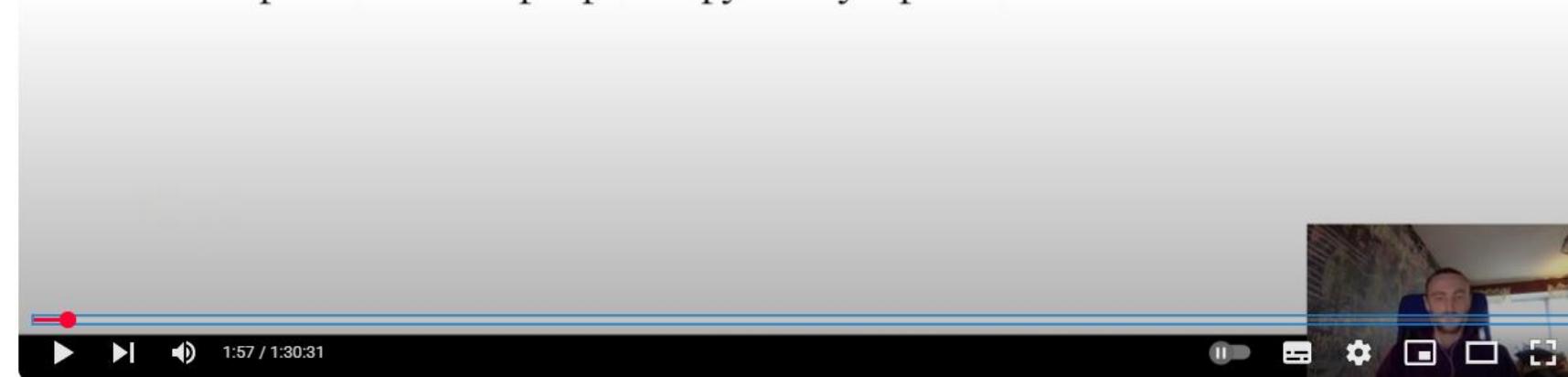




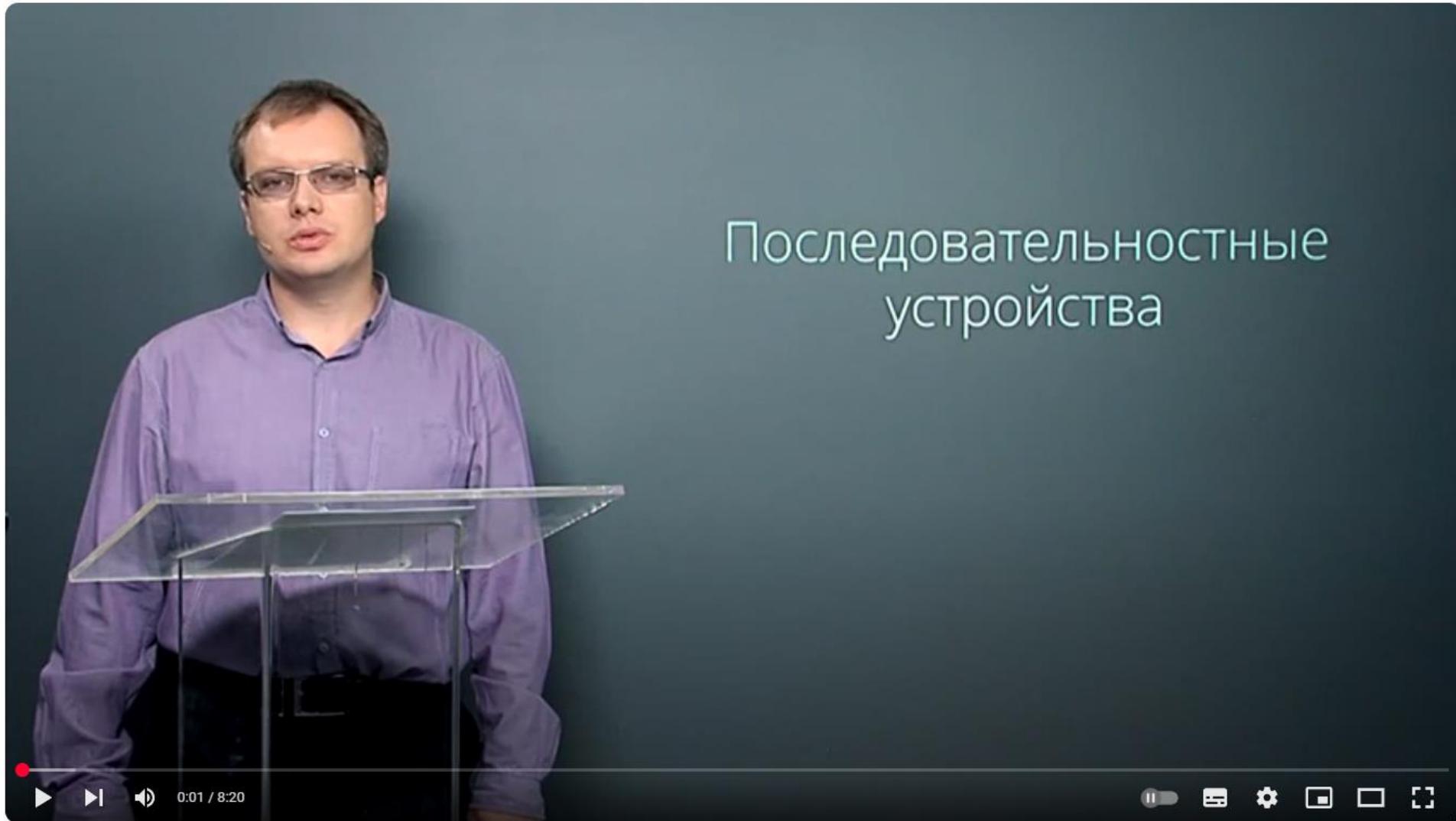
Последовательная логика (2021)
<https://www.youtube.com/watch?v=iB7cCzNEbAo>

План лекции

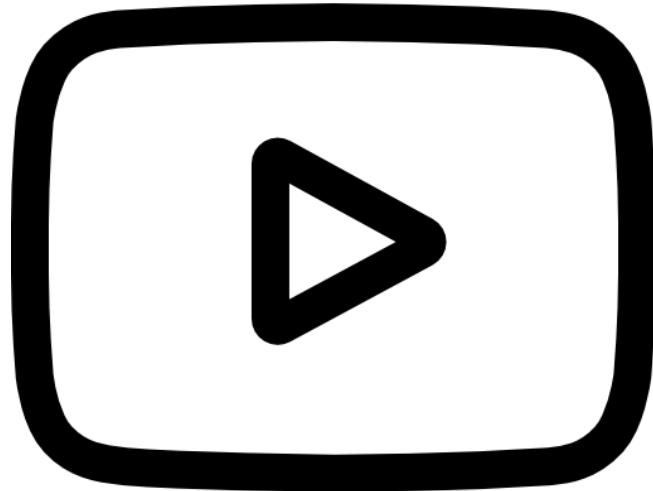
- Зашелка, триггер, регистр
- Регистровый файл
- Автомат состояний (конечный автомат)
- Временные характеристики цифровых устройств
- Примитивное программируемое устройство



ИВТ-23М. Л11. Последовательностная логика. Память (2020)
<https://www.youtube.com/watch?v=CZy5IDflgUU>



4 4 11 Последовательностные устройства Триггеры (2017)
<https://www.youtube.com/watch?v=hgswGqUKKmU>

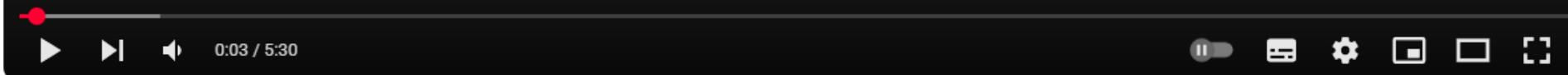
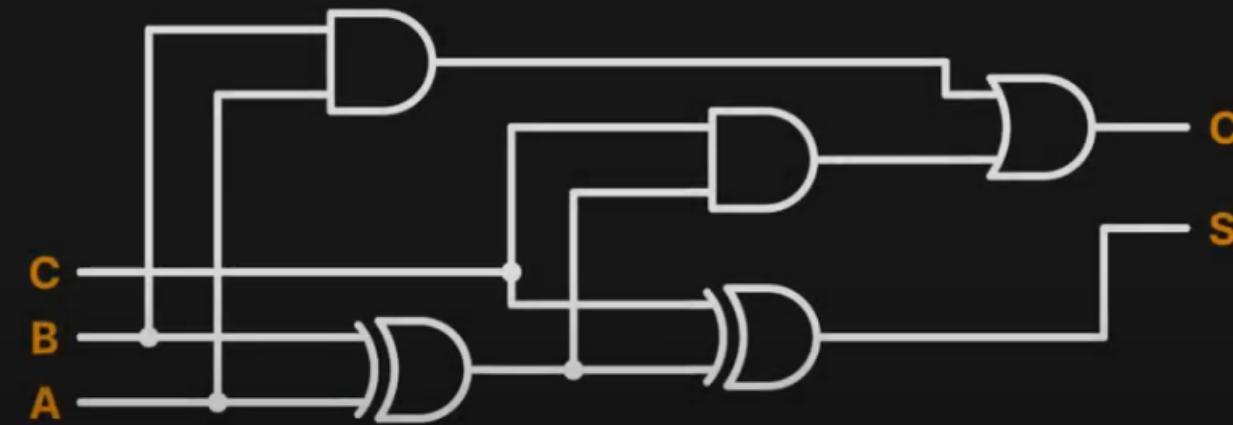


**Сумматоры,
Полусумматоры**



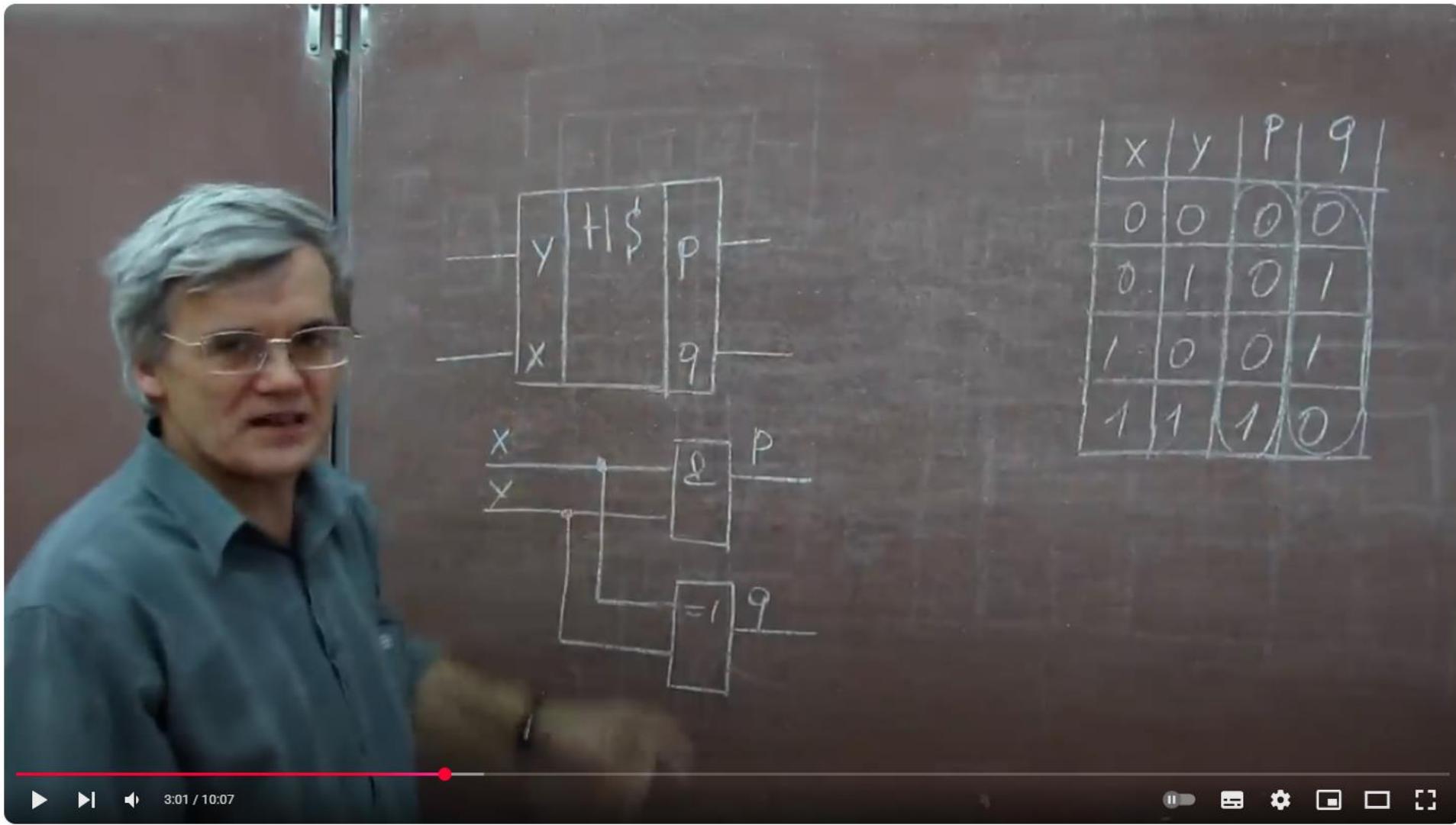
Сумматор

Компонент, способный сложить два разряда складываемых чисел, с учетом переноса из предыдущего разряда, и выдать значение текущего разряда и перенос в следующий разряд



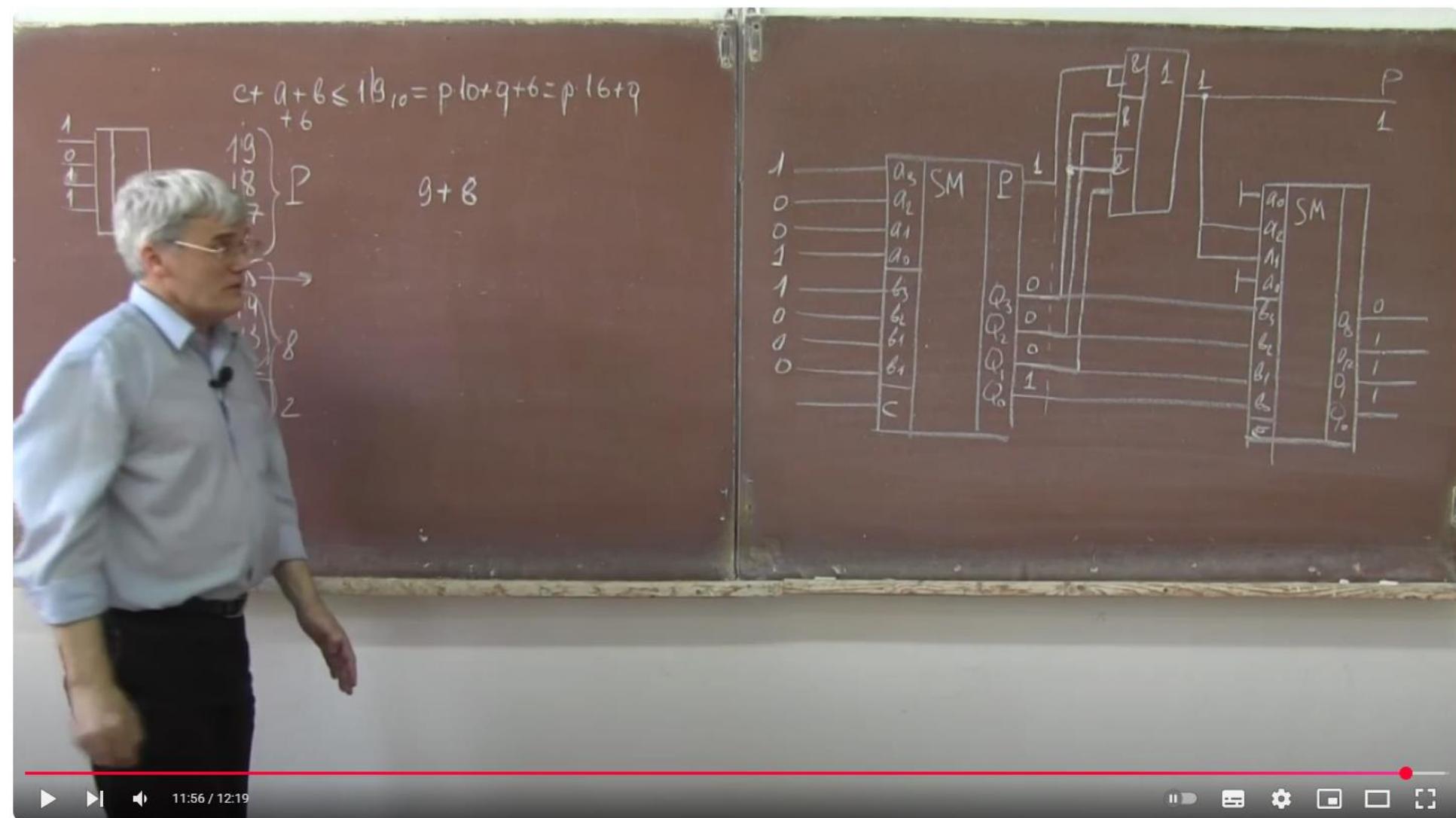
Сумматор (2025)

<https://www.youtube.com/watch?v=sUApzhh3HBO>



Лекция 105. Однобитный сумматор (2013)

<https://www.youtube.com/watch?v=3b-slaOPfGI>



Лекция 295. Двоично-десятичный сумматор (2016)

<https://www.youtube.com/watch?v=ALngFJzGfhU>

Binary addition:

$$\begin{array}{r} & 1 & 1 \\ & 100 & 1_2 \\ + & 11 & 2 \\ \hline 1100 & 2 \end{array}$$

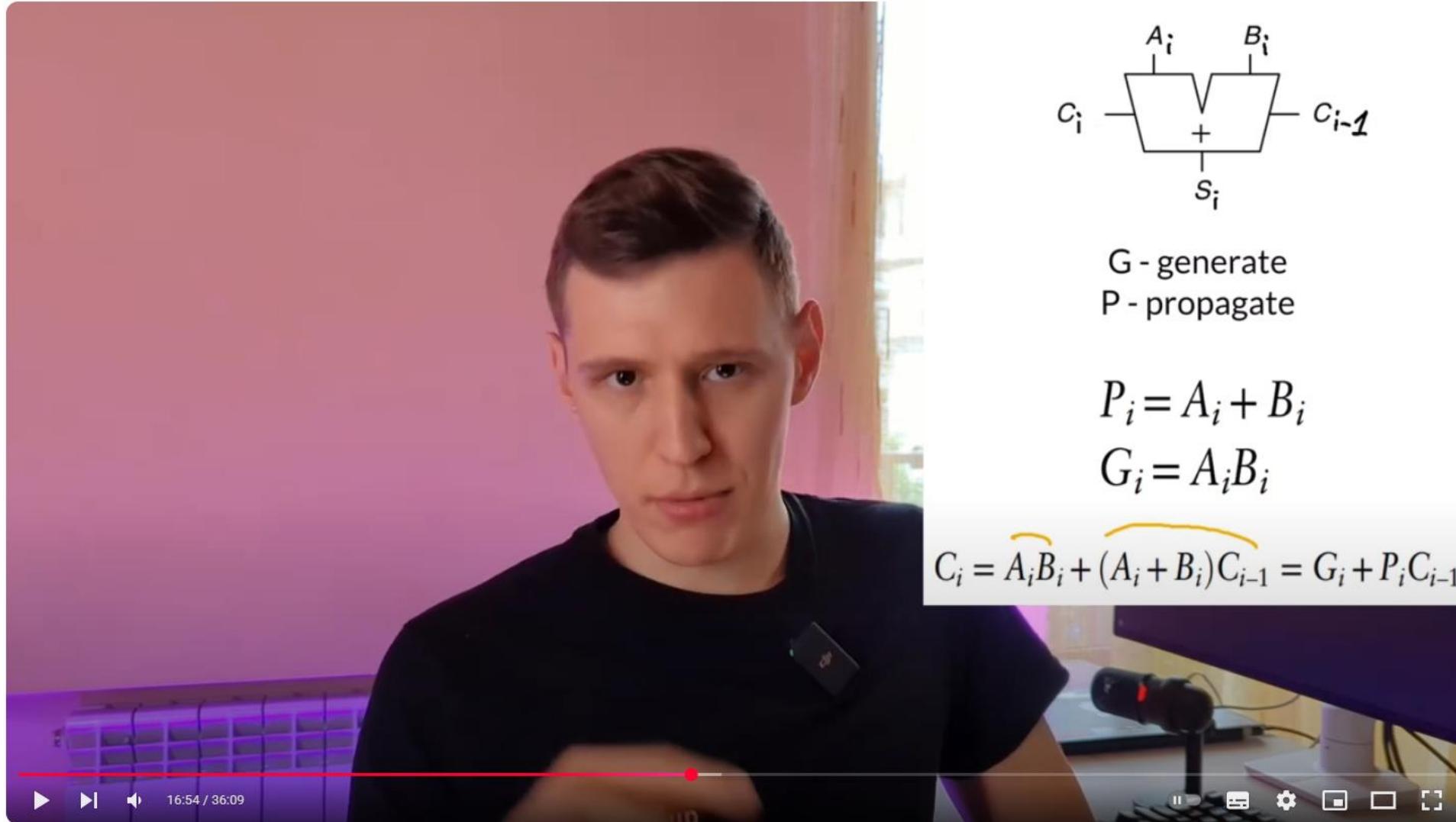
Half-Sum

Logic Circuits:

- Left: A vertical column of four boxes labeled A, H, S, and P from top to bottom. Inputs A and B enter the first two boxes. The third box is labeled "Half-Sum". The fourth box is labeled "Sum".
- Right: A logic diagram showing inputs A and B entering AND gates. The outputs of these AND gates enter an OR gate, which produces output S. The output of the OR gate also enters an inverter, which produces output P.

YouTube player controls at the bottom: play/pause, volume, progress bar (3:19 / 17:15), and other standard video controls.

Полусумматоры, сумматоры. Назначение, классификация, принцип работы, типовые схемы. (2021) <https://www.youtube.com/watch?v=e5TmYrVjZl0>



The video shows a young man in a black t-shirt sitting in front of a computer monitor. He is looking directly at the camera. The background is a pink wall. On the desk in front of him is a purple keyboard and a microphone. The video player interface at the bottom shows a play button, volume, and a progress bar at 16:54 / 36:09.

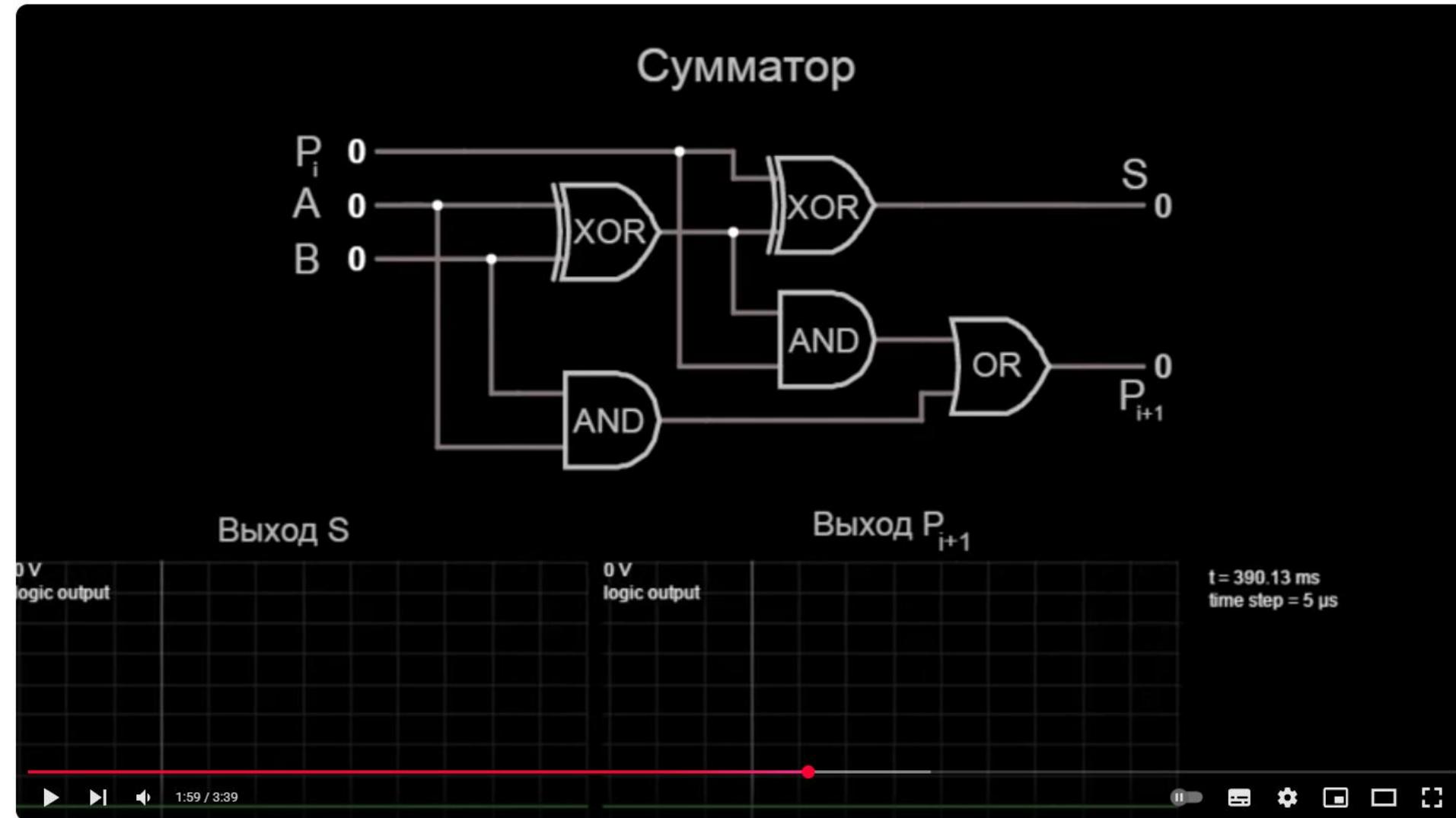
$C_i = \overbrace{A_i B_i}^G + \overbrace{(A_i + B_i) C_{i-1}}^P = G_i + P_i C_{i-1}$

$P_i = A_i + B_i$
 $G_i = A_i B_i$

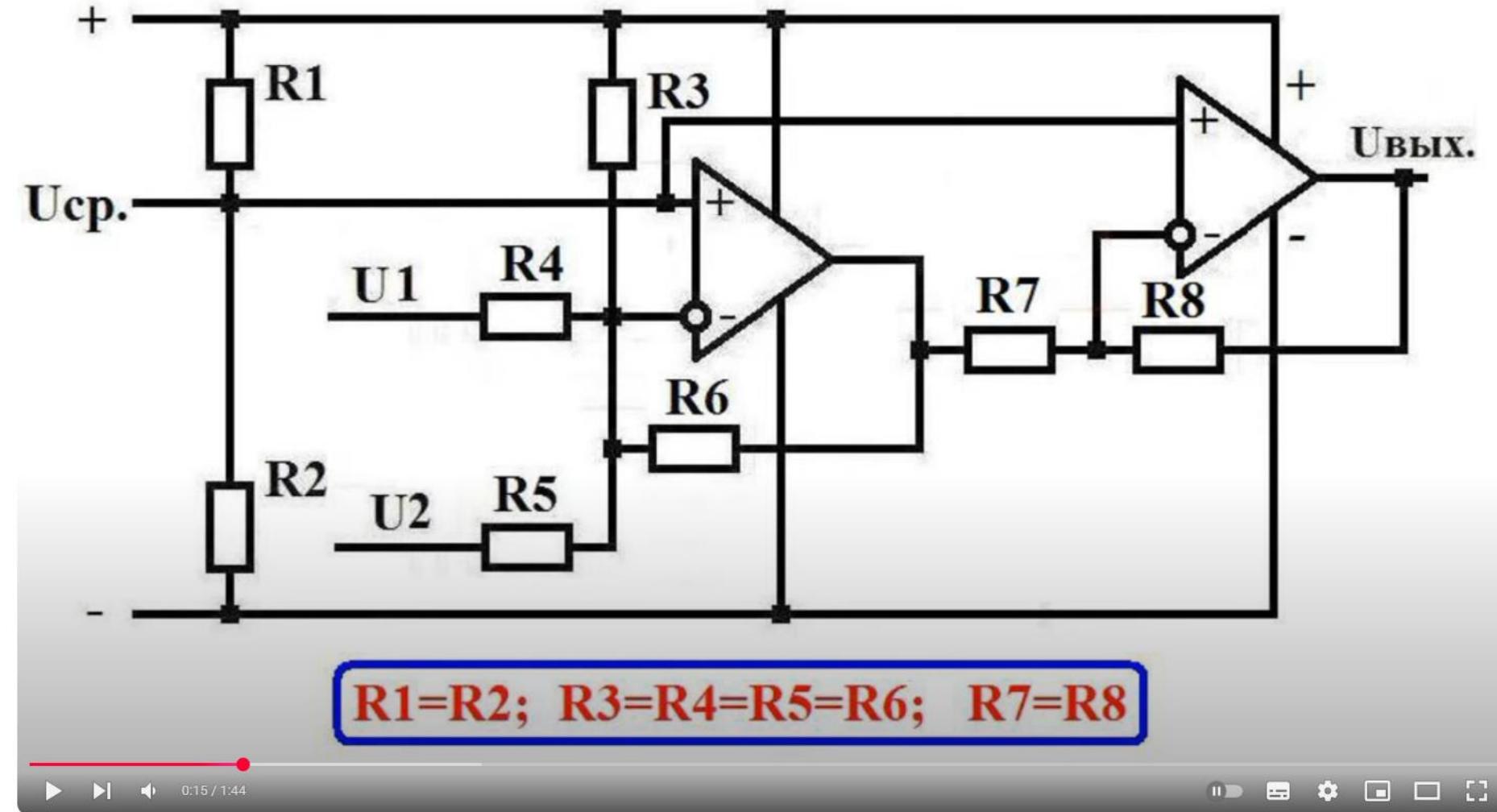
A_i B_i c_i c_{i-1} s_i

Diagram of a full adder circuit. Two inputs, A_i and B_i , enter two inputs of a V-shaped summing junction. The output of this junction is labeled s_i . The other input of the junction is the carry-in from the previous stage, c_{i-1} . The output of the junction is the sum bit, c_i .

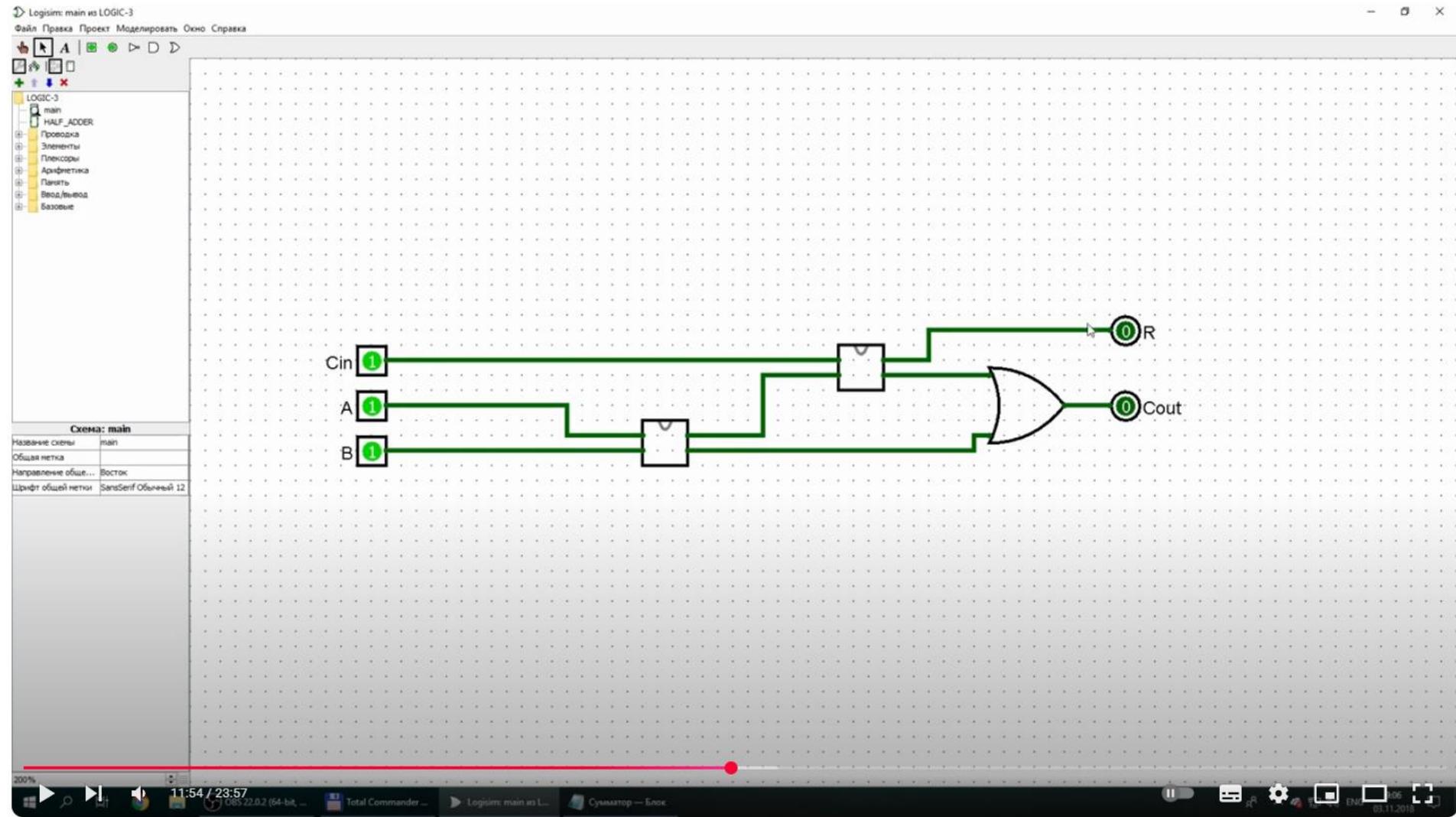
Префиксный сумматор: все о сумматорах по шагам (2023)
<https://www.youtube.com/watch?v=G1GnAhp5WHc>



Полусумматор и сумматор (2018)
<https://www.youtube.com/watch?v=adITv8WwMk4>

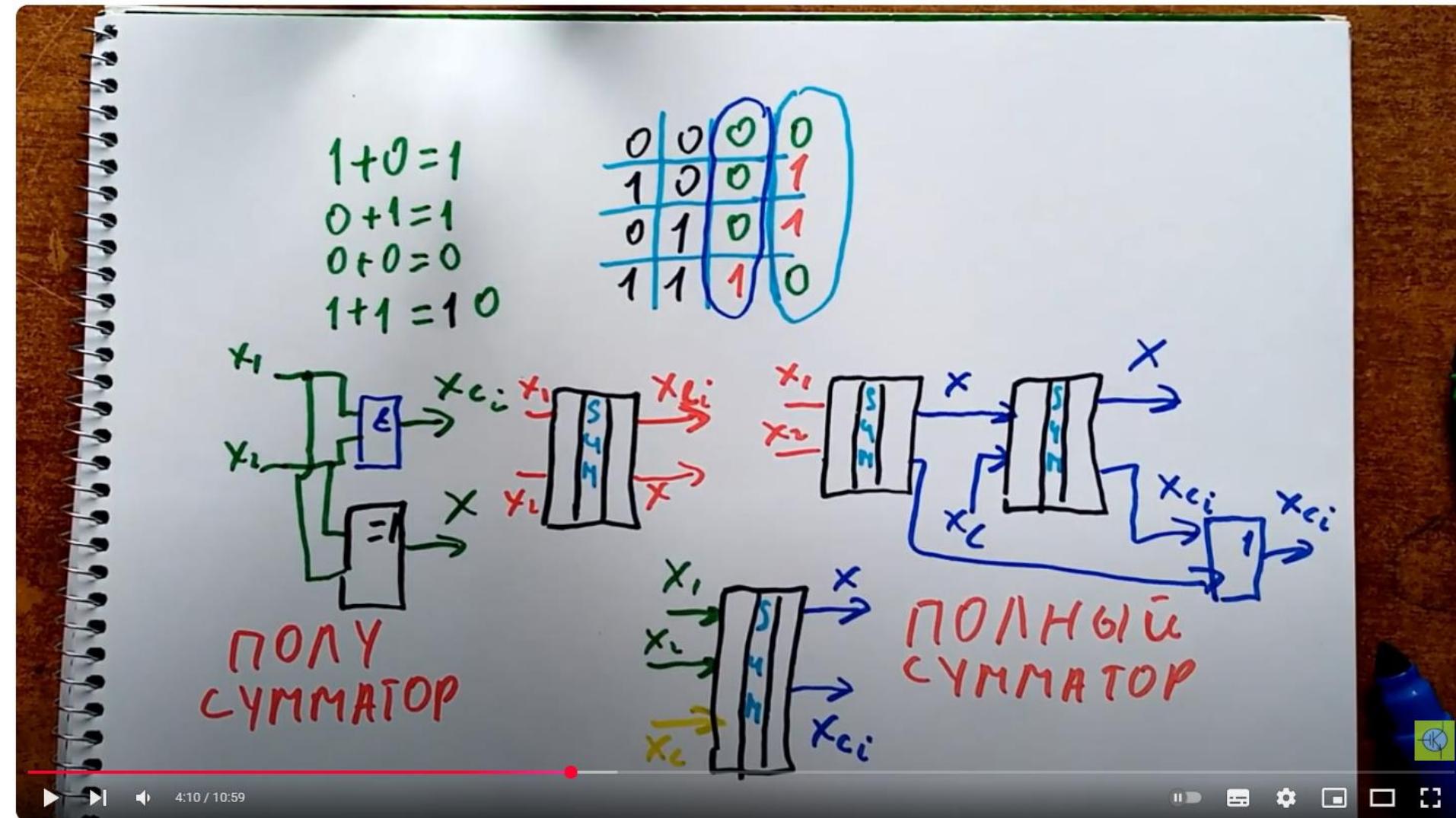


Сумматор на ОУ: Просто и Гениально! (2025)
<https://www.youtube.com/watch?v=WAQevRd02r4>



04 Сумматор (2024)

<https://www.youtube.com/watch?v=1xs0zsgbsRY>



Компьютер на транзисторах своими руками №6 Сложение чисел. Сумматор (2020)
<https://www.youtube.com/watch?v=7IRN33i016M>

The image shows a YouTube video player interface. In the top left corner, there is a logo for 'e^x ЭКСПОНЕНТА' with the text 'ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И МОДЕЛИРОВАНИЯ'. In the top right corner, there is a link 'Engee - переходите на сайт' with an info icon. Below the video title, there is a 3D surface plot of a bell-shaped curve. The video title is 'Основы цифровой схемотехники' (Fundamentals of Digital Circuit Design) followed by '08. Сумматоры' (8. Adders). On the left side of the video frame, there is a small image of a blue microchip. Below the video frame, there is a control bar with a play button, a progress bar showing '0:02 / 4:35', and other standard video controls.

Основы цифровой схемотехники

08. Сумматоры

A logic gate symbol for an SR flip-flop. It has two inputs, S and R, and two outputs, Q and Q̄. The symbol is divided vertically by a line, with S on the left and R on the bottom-left, and Q and Q̄ on the right.

A graph showing a sigmoidal curve (logistic function) starting at a low value, passing through zero, and approaching a high value. This represents the transfer characteristic of a digital logic element.

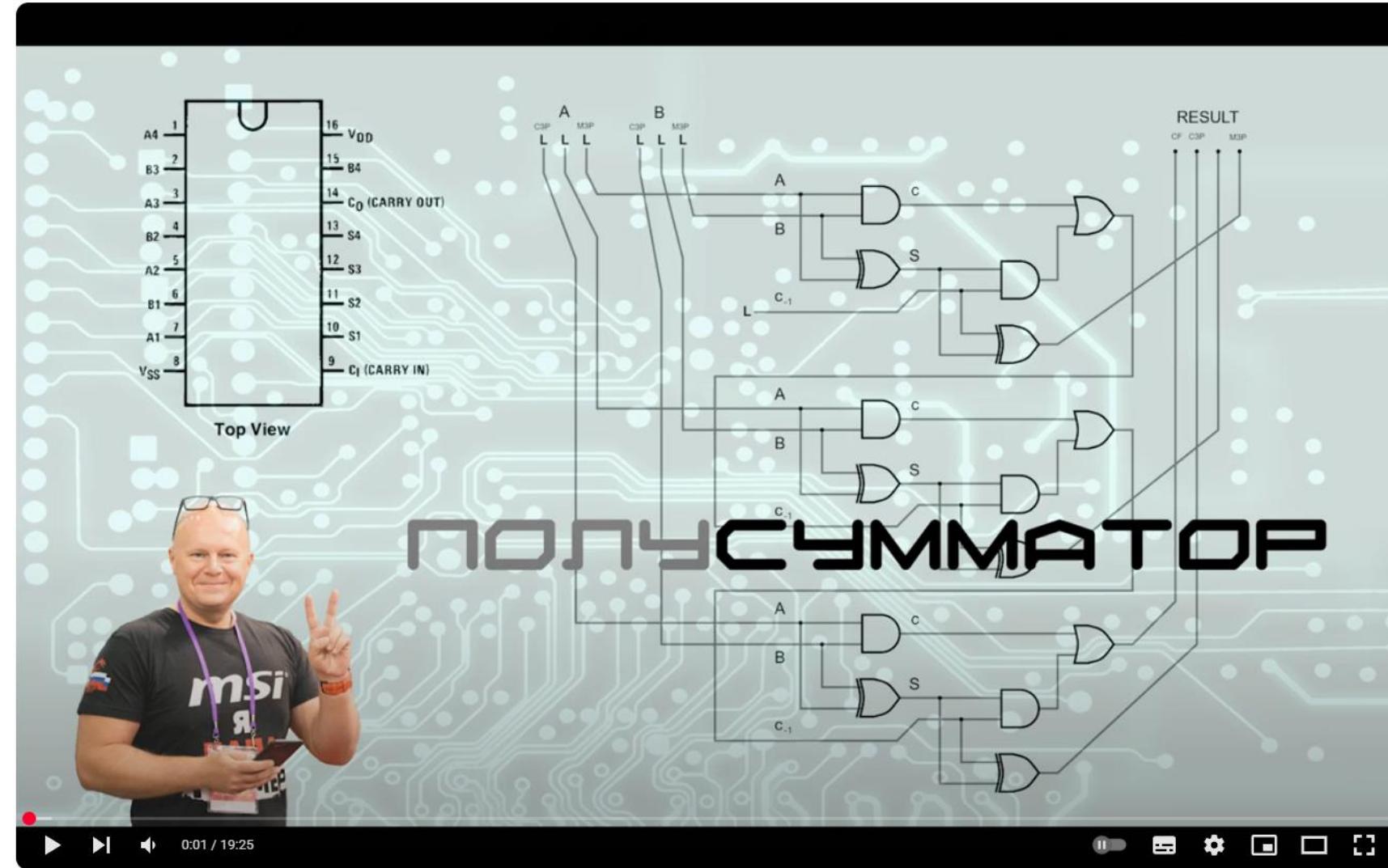
Марат Усс

Основы цифровой схемотехники. 08. Сумматоры (2019)
https://www.youtube.com/watch?v=_ISGmCyzAN4

The screenshot shows a YouTube video player with a dark theme. The video content is a presentation slide with a light gray background. At the top center, the title Сумматоры is displayed. Below it, the text defines an Сумматор (SM) as a node designed for performing the microoperation of word addition. It notes that depending on the системы счисления, there are binary, binary-decimal, and other adders. The slide then distinguishes between комбинационные and накапливающие adders. Комбинационные adders do not have memory elements and implement the addition operation in the form of $C := A+B$. Накапливающие adders have a register where the addend is stored before the start of the operation, and the result is formed at the end of the addition. They implement the operation $C := C+A$. The slide also mentions that adders are categorized into одноразрядные and многоразрядные. One-bit adders are synthesized using full adder blocks. Multi-bit adders are divided into последовательные and параллельные. Sequential adders process digits sequentially. Parallel adders handle digits simultaneously across all bits. Finally, the slide discusses the способу организации межразрядных переносов (method of organizing inter-bit carries), which classifies parallel adders into those with sequential, parallel, and group carries. The video player interface includes a progress bar at 0:01 / 7:20, a play button, volume controls, and a settings icon.

РК6. Схемотехника. Сумматоры (2020)

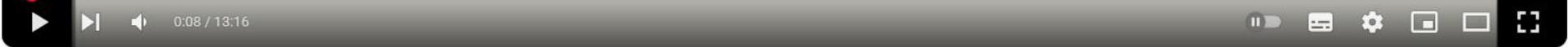
<https://www.youtube.com/watch?v=kHPzByBmn98>



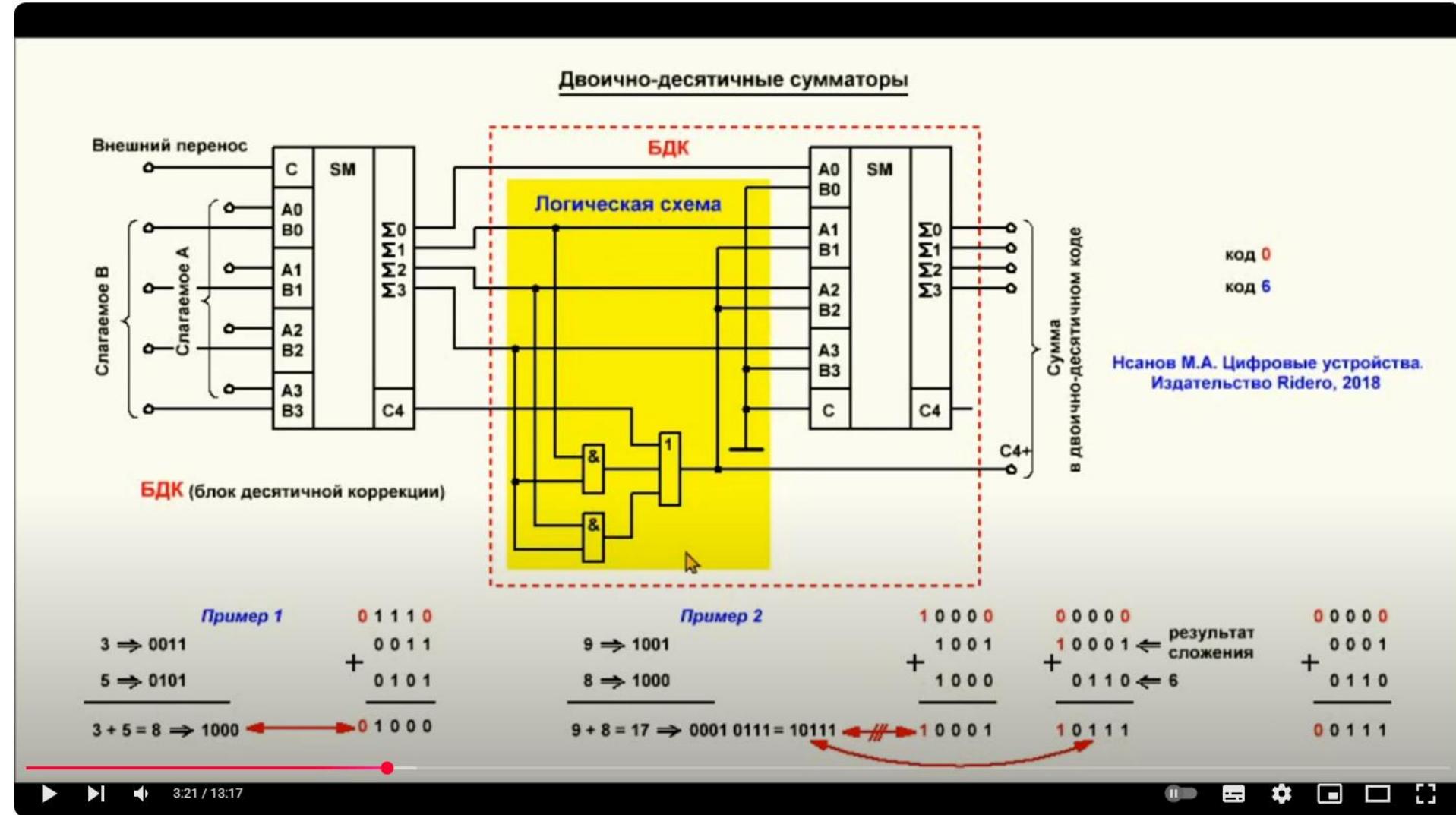
Полусумматор, Сумматор, Многоразрядный сумматор (2024)
<https://www.youtube.com/watch?v=Ik2un9jTmOo>

ДВОИЧНО-ДЕСЯТИЧНЫЕ СУММАТОРЫ ЧАСТЬ 1

Комбинационные ЦУ



Видеоурок 3.34. Двоично-десятичные сумматоры. Часть 1 (2023)
<https://www.youtube.com/watch?v=fsliyGX-b7s>

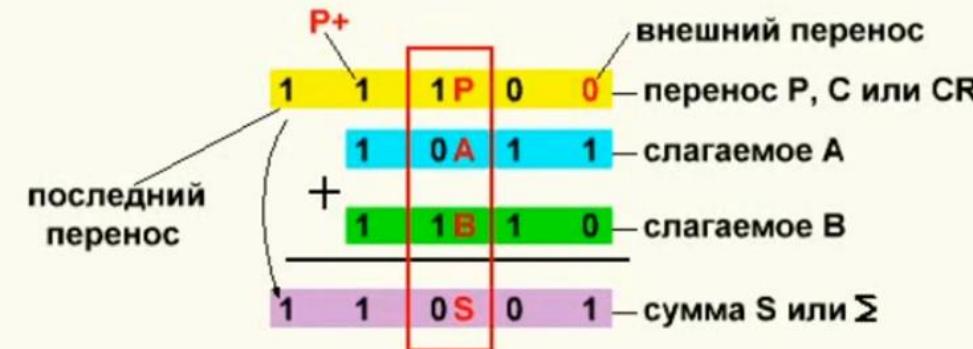


Видеоурок 3.35. Двоично-десятичные сумматоры. Часть 2 (2023)
<https://www.youtube.com/watch?v=wqGq1Ht9SV0>



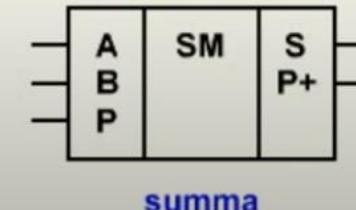
Вideoурок 3.30. Двоичные сумматоры. Часть 1 (2023)
<https://www.youtube.com/watch?v=cawnk7w8b84>

Двоичные сумматоры



Входы			Выходы	
A	B	P	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

одноразрядный сумматор



2 - в двоичном коде 10

3 - в двоичном коде 11

СДНФ:

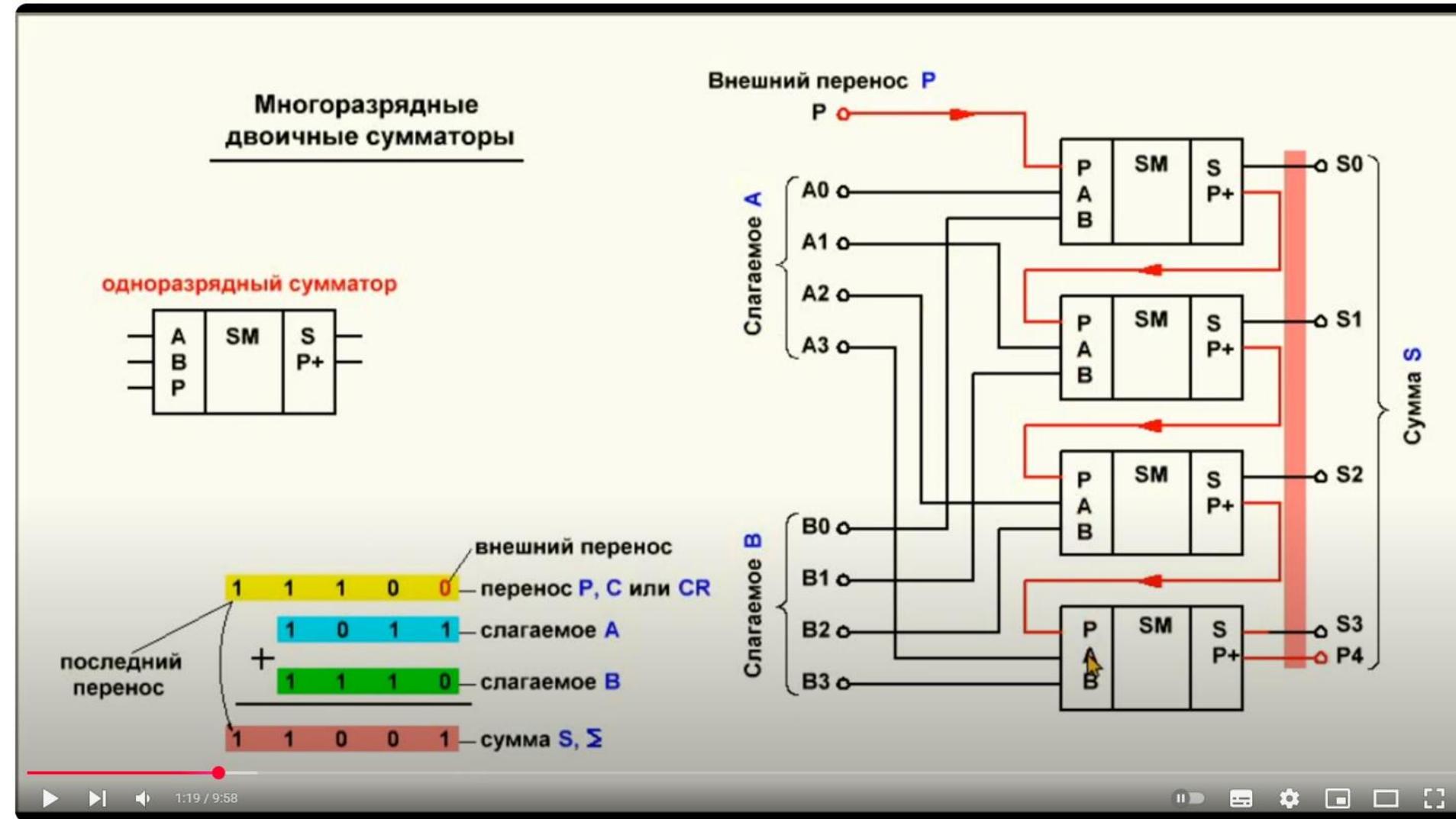
$$S = \overline{A} \cdot \overline{B} \cdot P + A \cdot B \cdot P + A \cdot \overline{B} \cdot P + A \cdot B \cdot \overline{P}$$

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

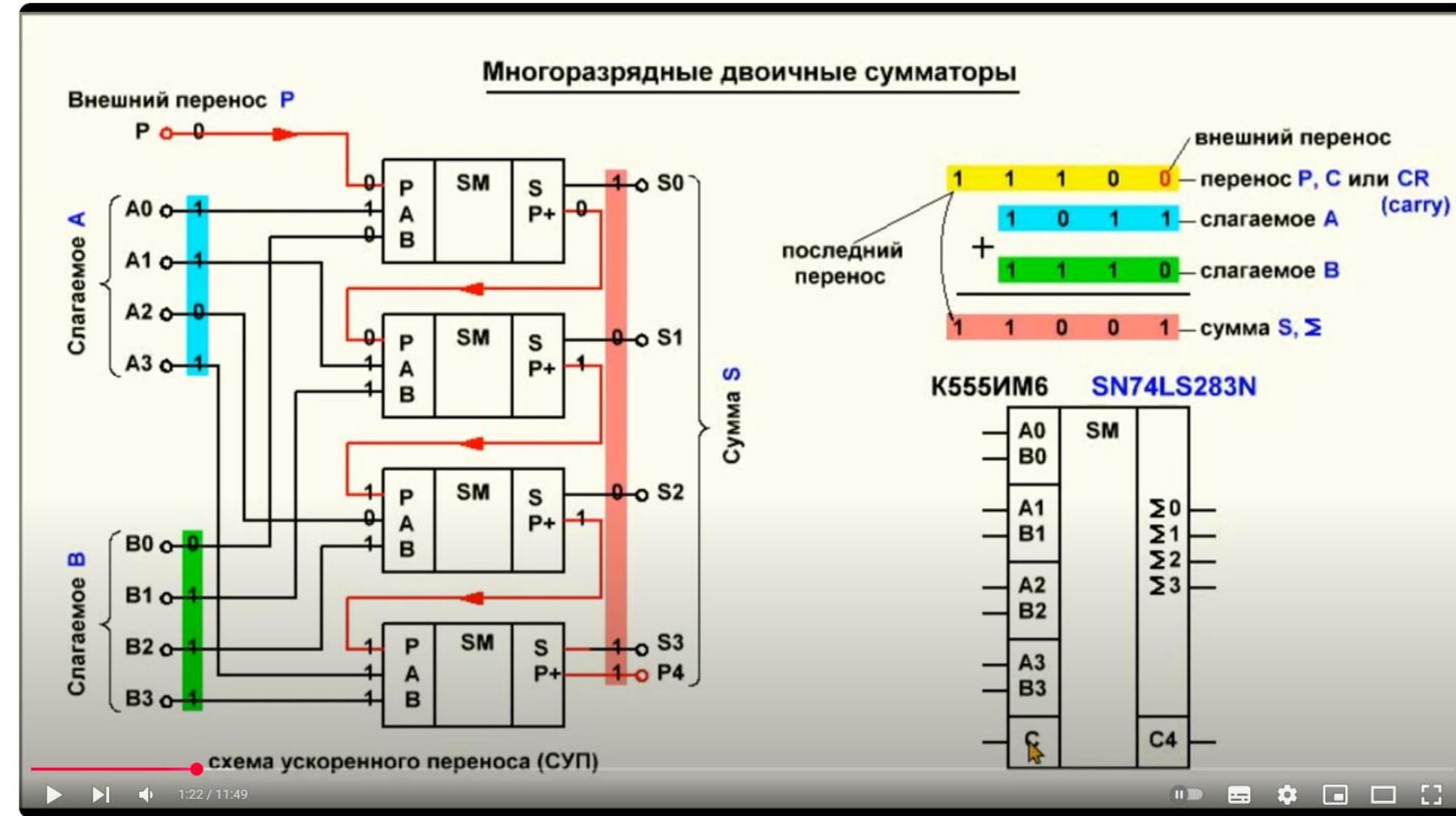
▶ ⏸ 🔍 6:32 / 13:19



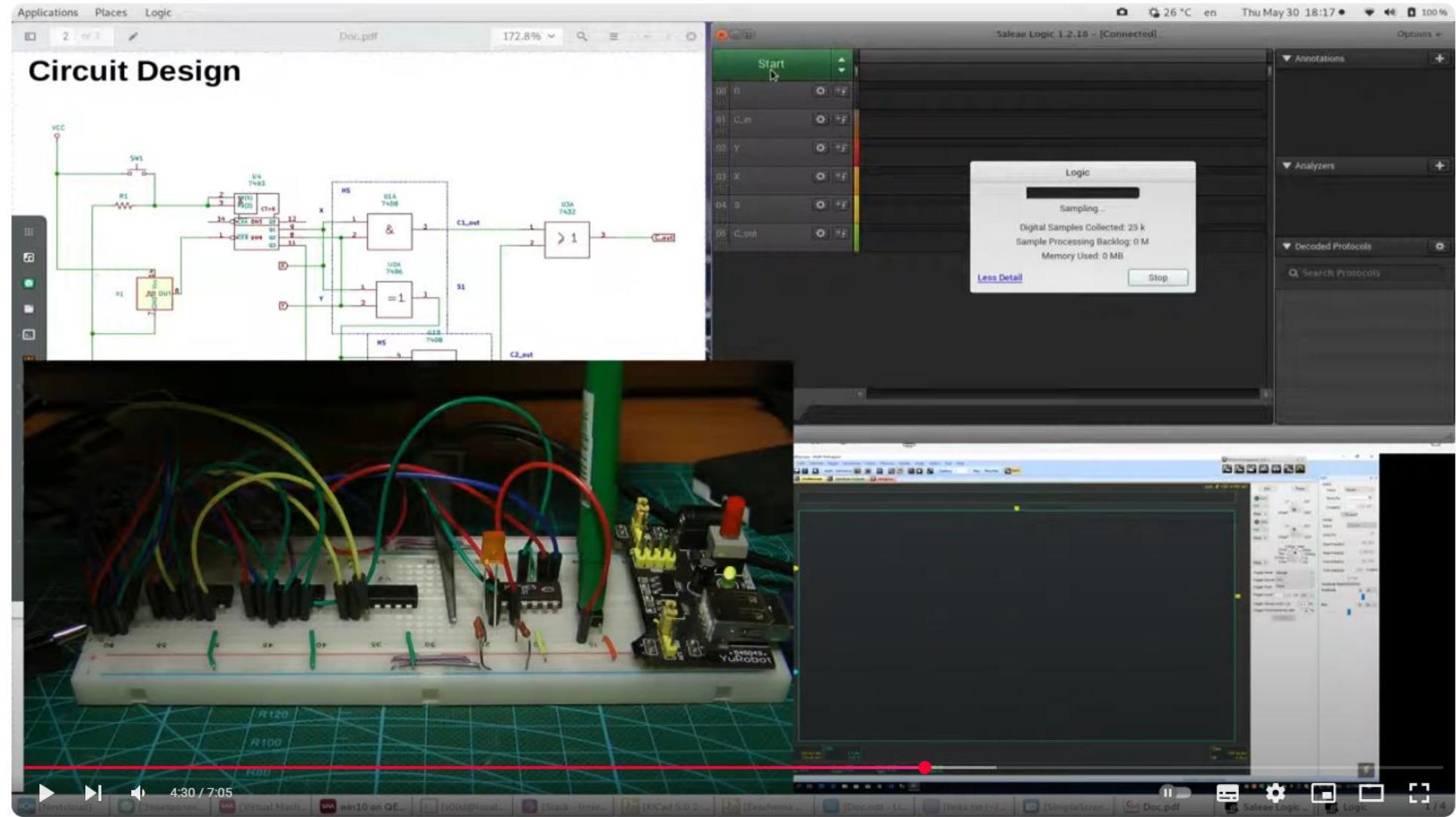
Видеоурок 3.31. Двоичные сумматоры. Часть 2 (2023)
<https://www.youtube.com/watch?v=vWcj6fvXdnA>



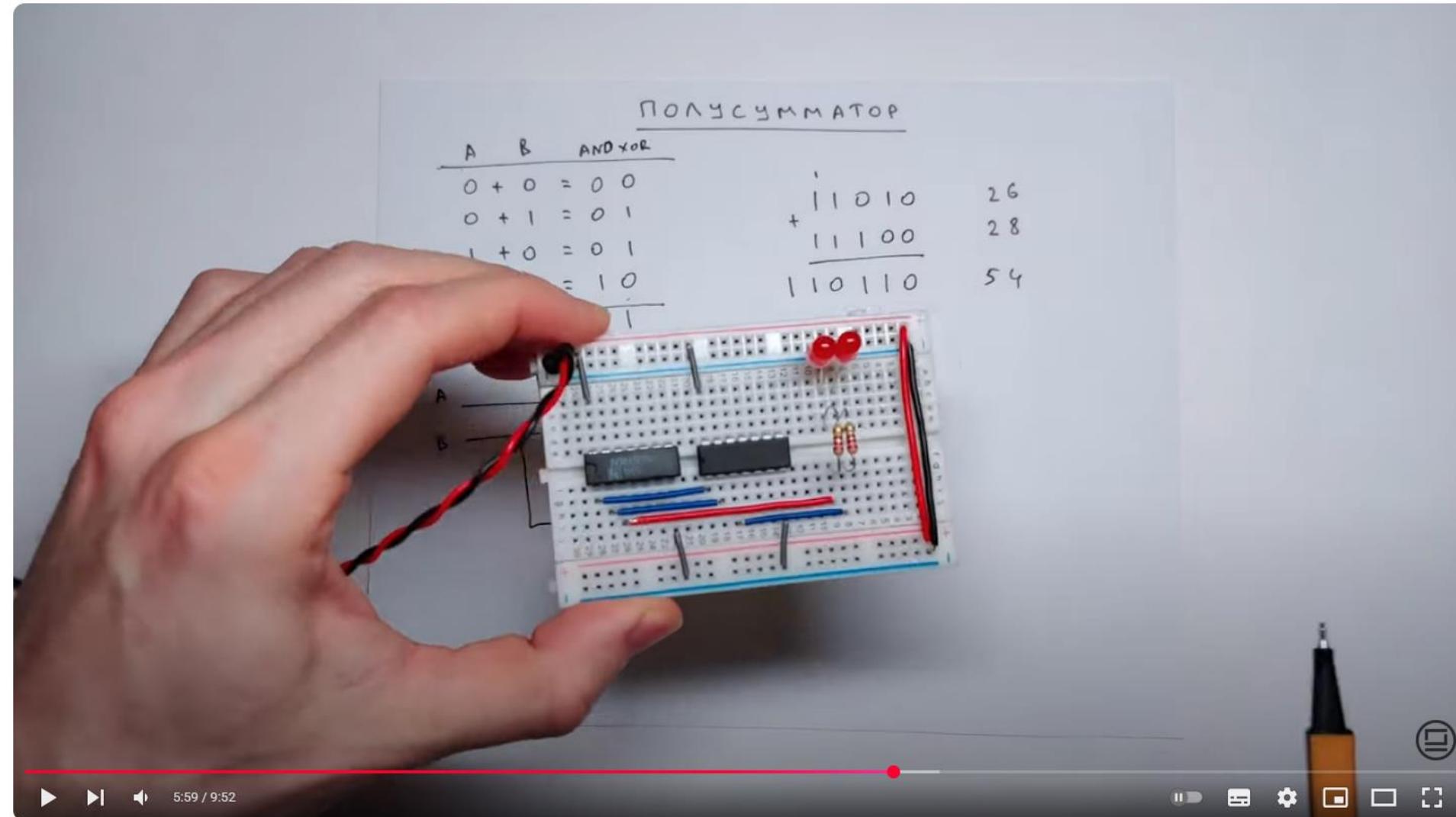
Видеоурок 3.32. Двоичные сумматоры. Часть 3 (2023)
<https://www.youtube.com/watch?v=fxWw1ZpQoQ>



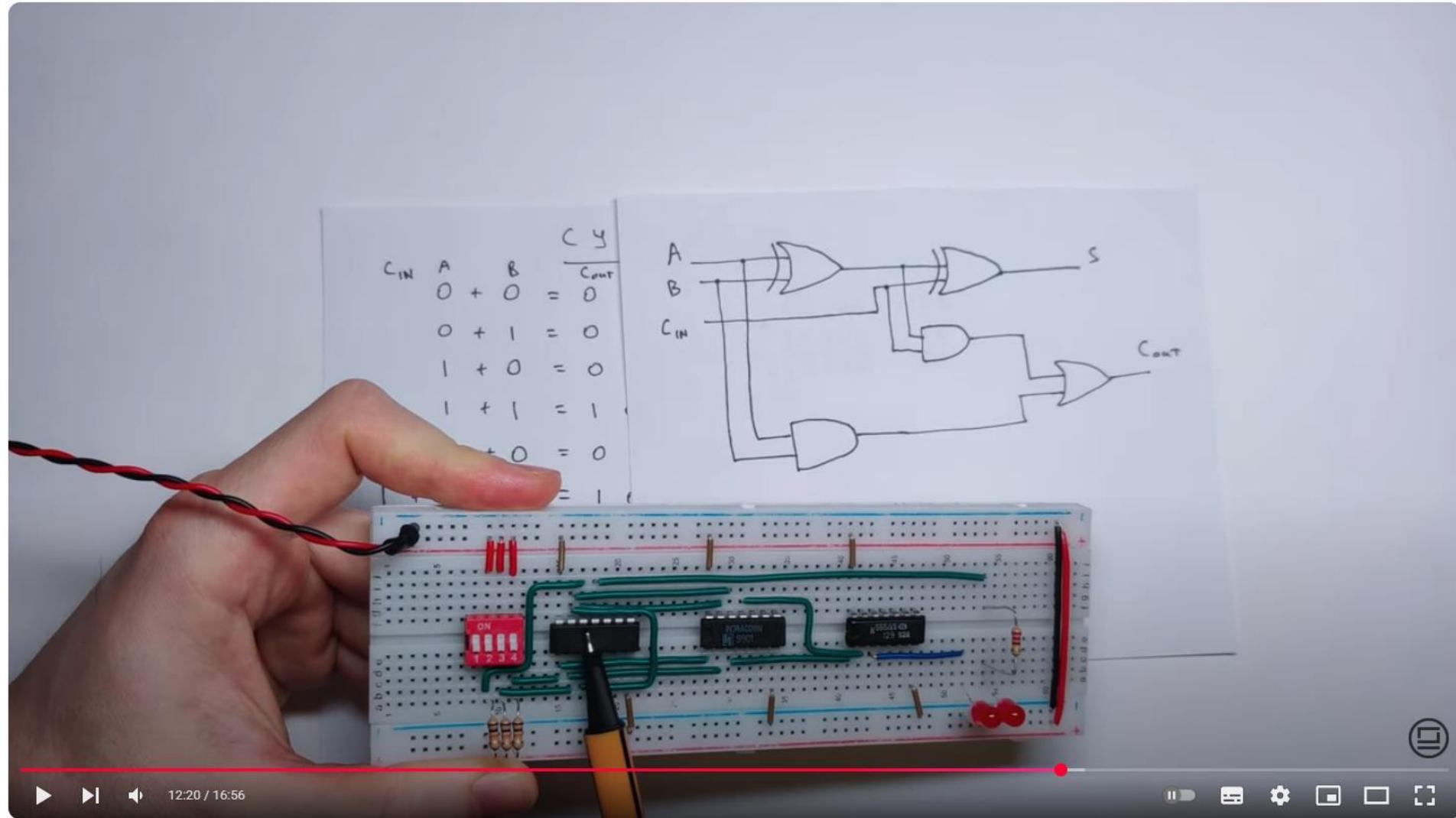
Видеоурок 3.33. Двоичные сумматоры. Часть 4 (2023)
<https://www.youtube.com/watch?v=Zq81TKZt8xU>



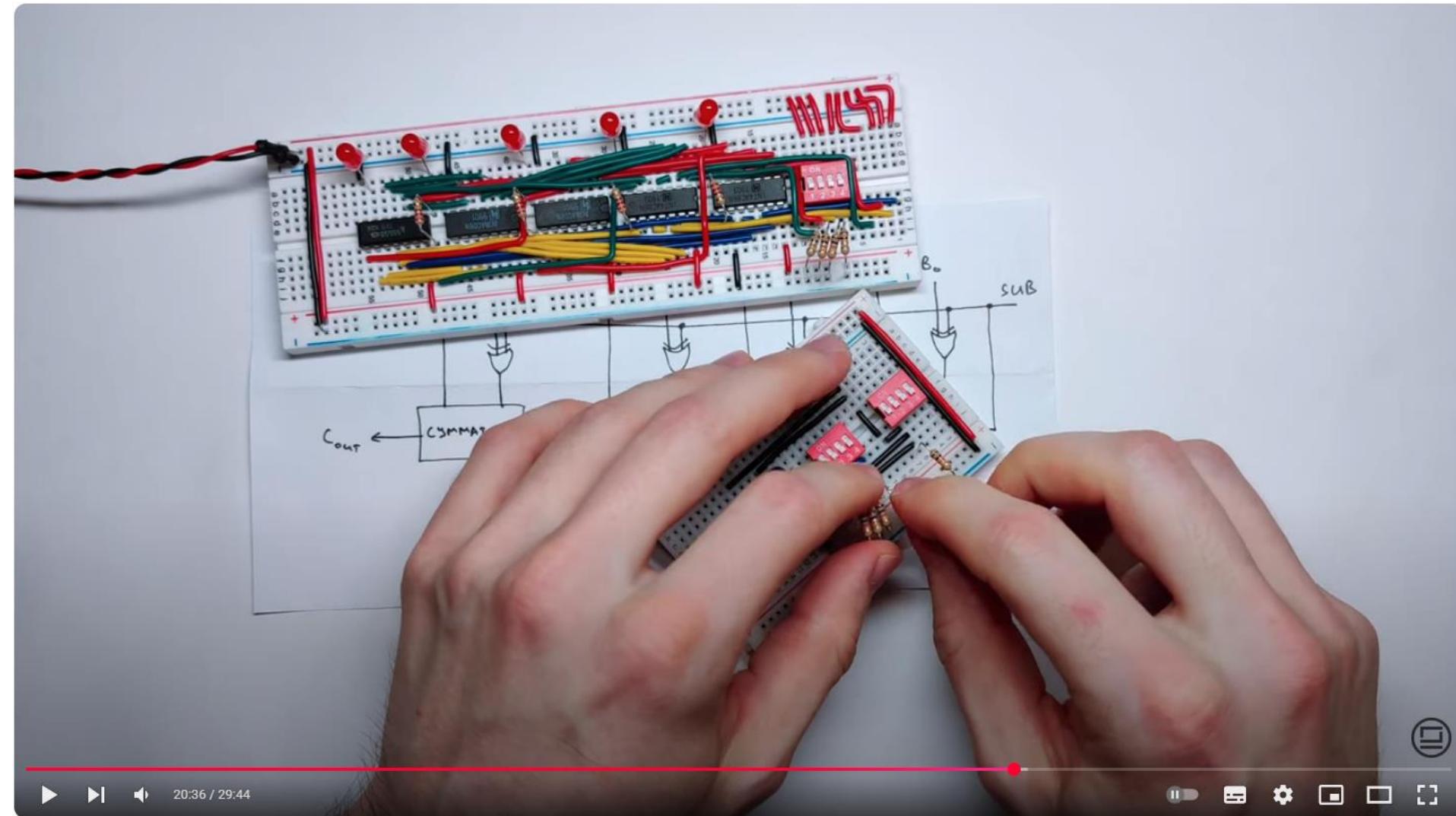
0105. Однобитный сумматор (2019)
<https://www.youtube.com/watch?v=vynyNsA1b0Y>



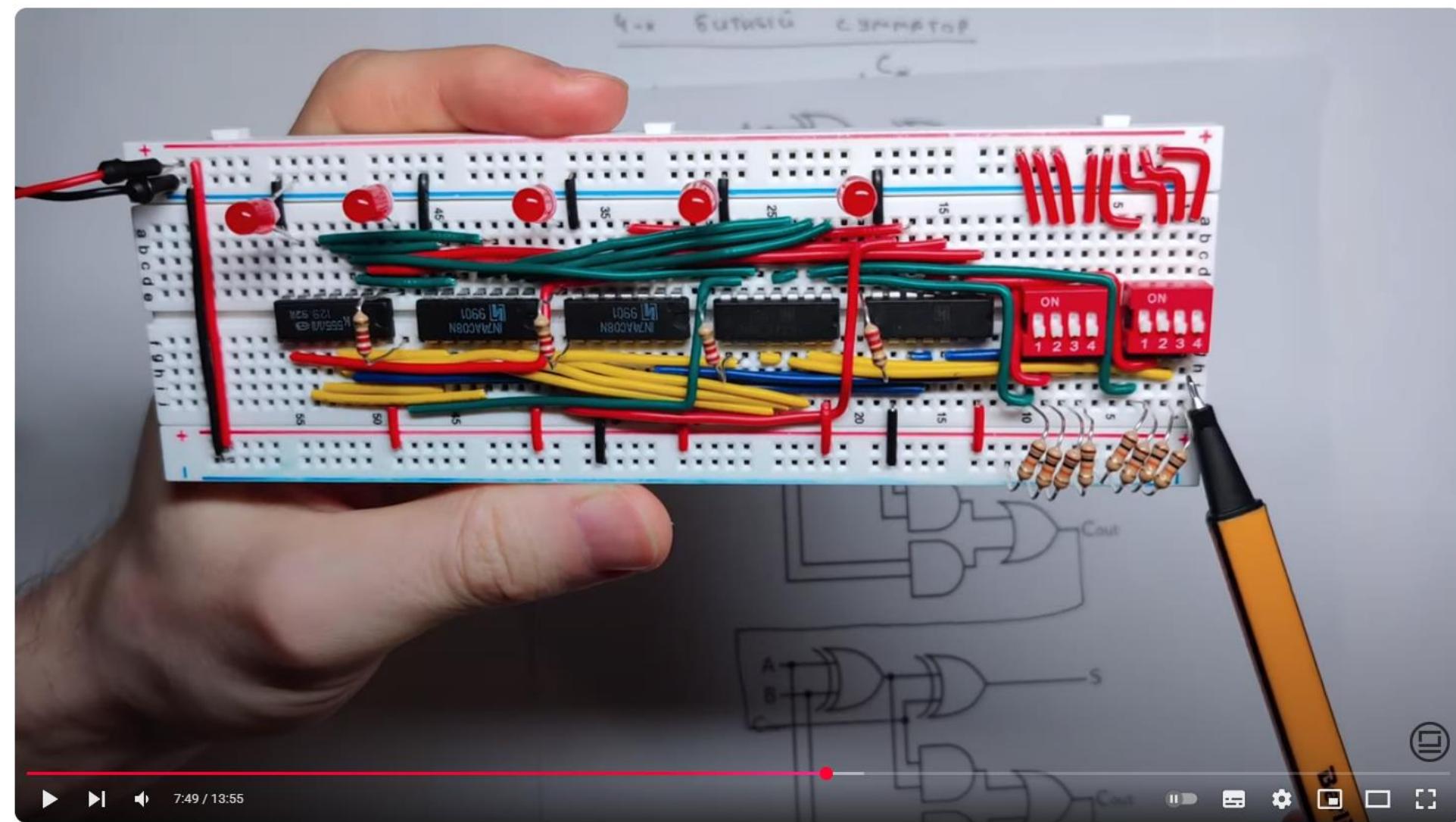
BitFlip. Полусумматор - складываем двоичные цифры (2020)
<https://www.youtube.com/watch?v=tbqtt7cvjxM>



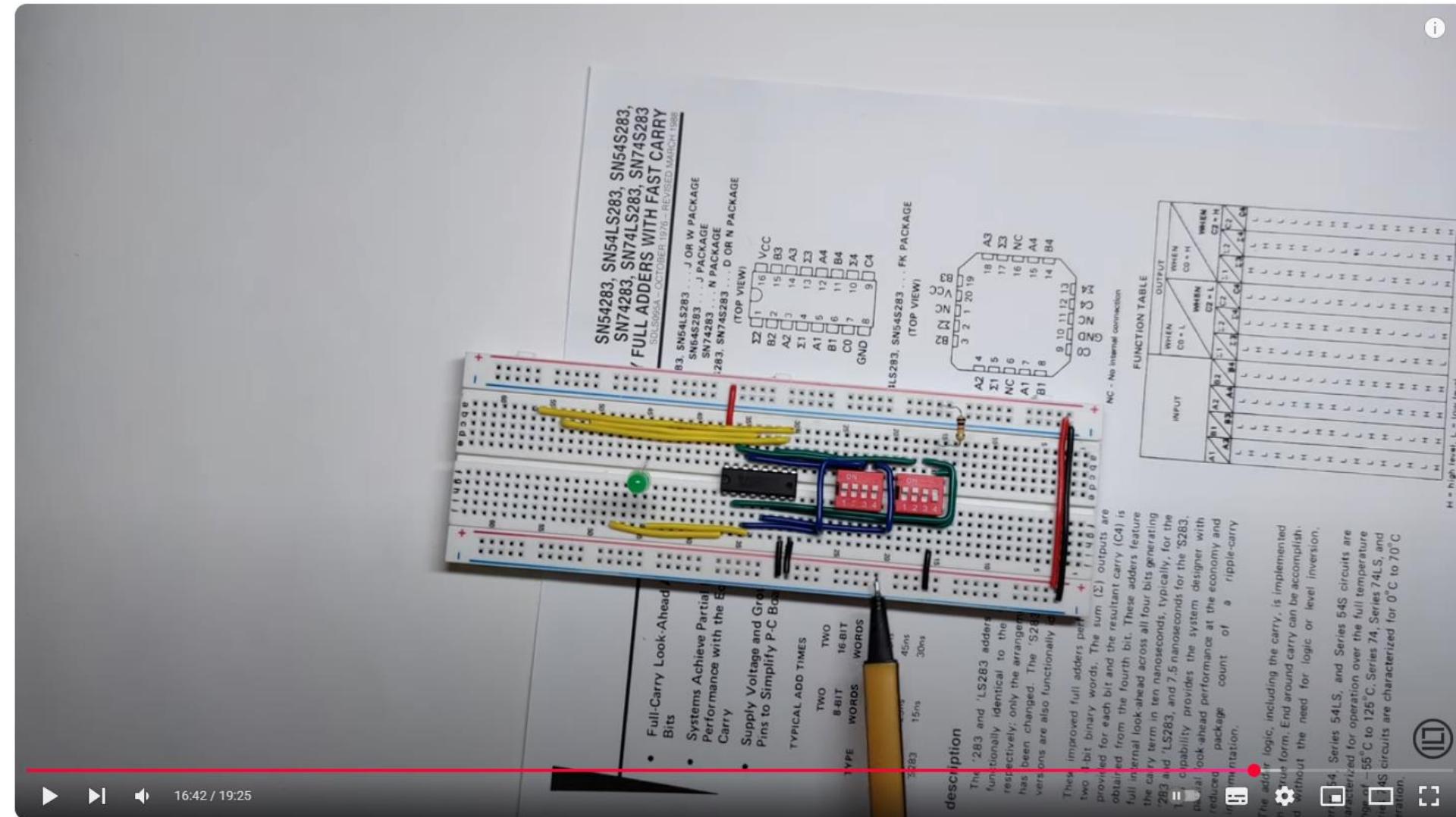
BitFlip. Сумматор (2020)
https://www.youtube.com/watch?v=5pK5m0d_RNk



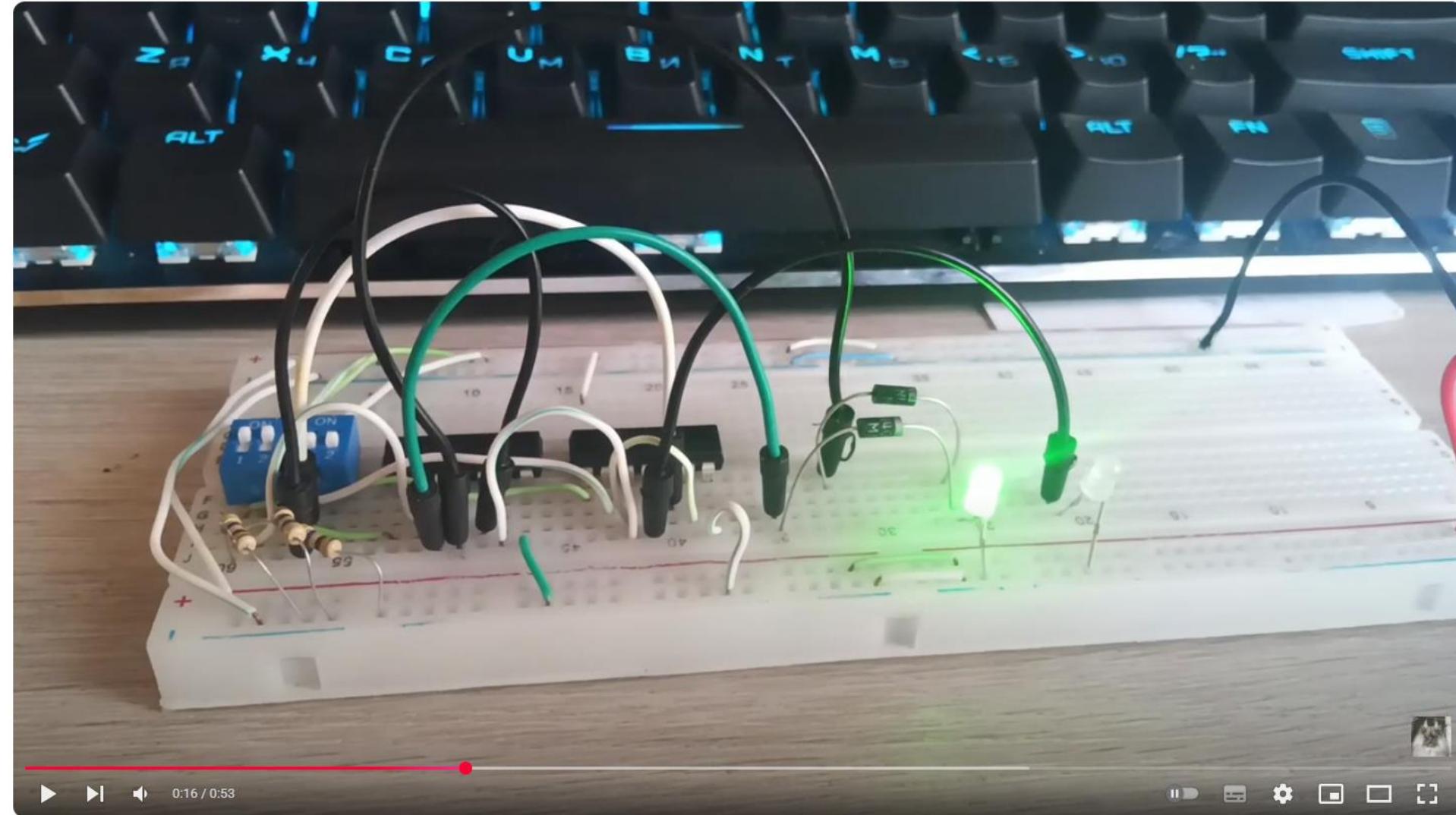
BitFlip. Сумматор с функцией вычитания (2020)
https://www.youtube.com/watch?v=K2_RWCCpsS8



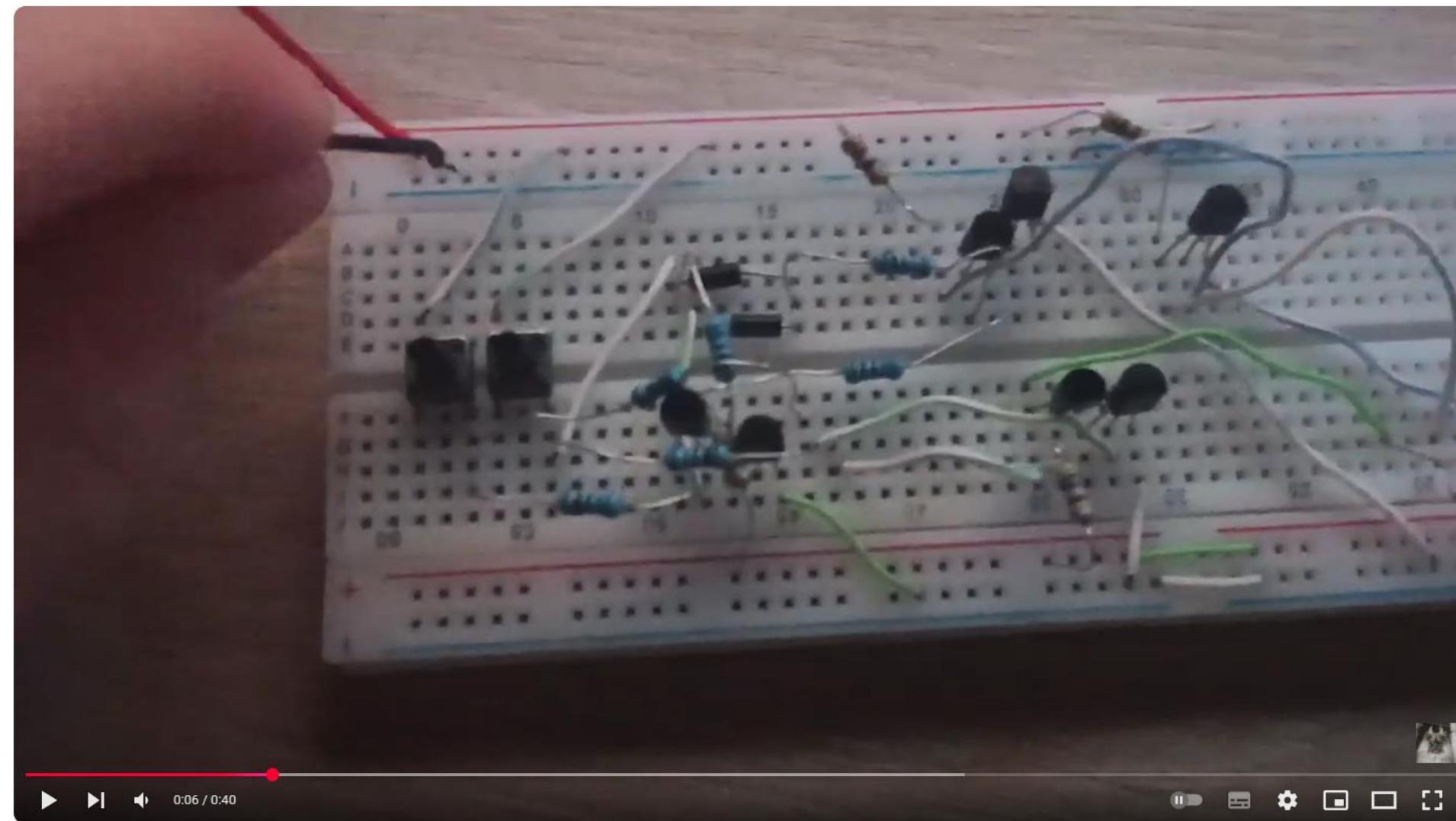
BitFlip. Четырехбитный сумматор (2020)
<https://www.youtube.com/watch?v=ONEMWuStDI8>



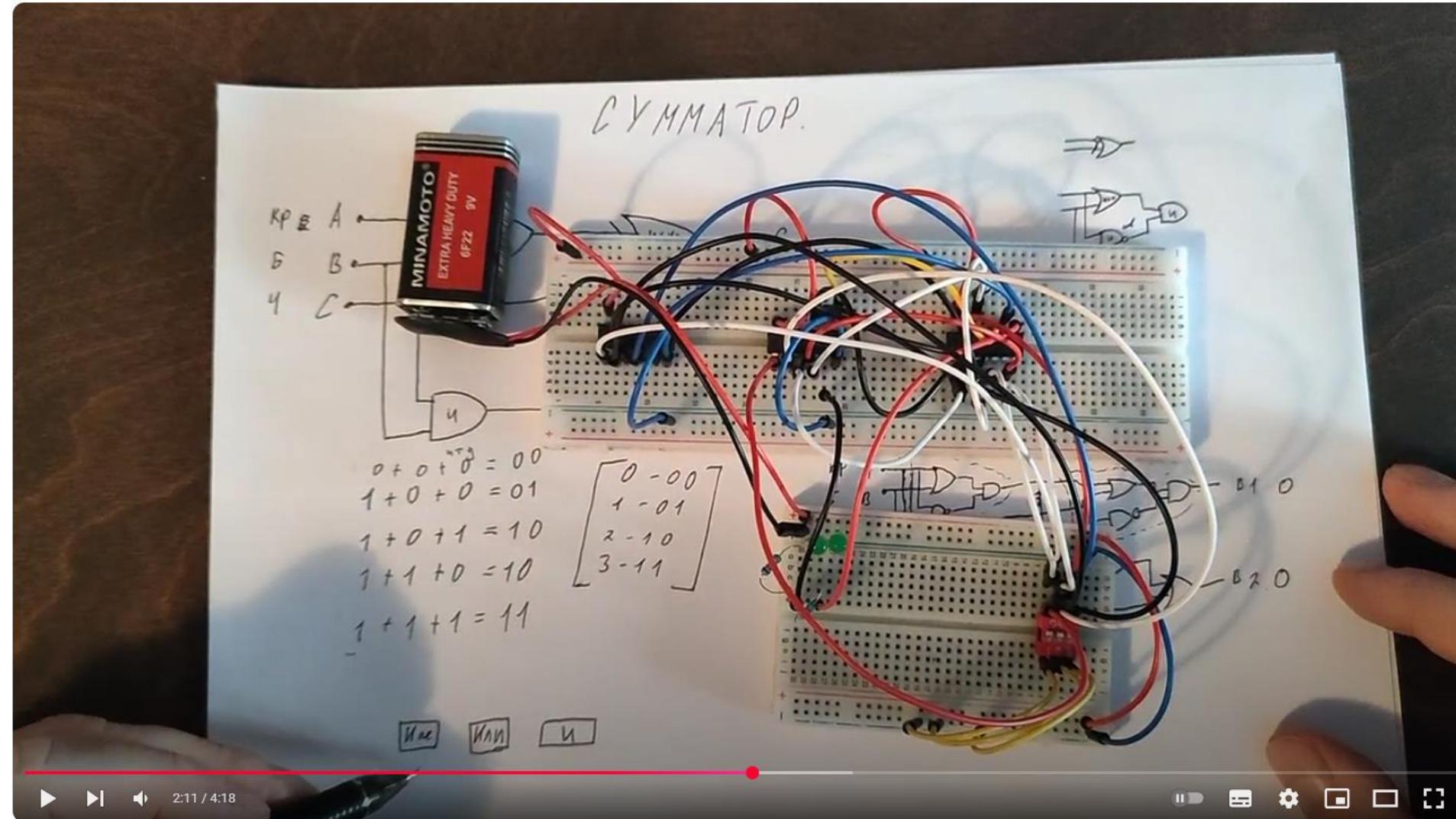
BitFlip. Четырехбитный сумматор в одной микросхеме, стягивающие резисторы и микросхемы ТТЛ логики (2020) https://www.youtube.com/watch?v=I4eomM5v_2I



Полный Однобитный Сумматор (2024)
<https://www.youtube.com/watch?v=fqYqFOTrkro>



Полусумматор на npn транзисторах (2024)
<https://www.youtube.com/watch?v=2vXxgUxvpoM>



Сумматор на логических элементах (2024)
<https://www.youtube.com/watch?v=1vfeMJhGQJU>

Курс: Вычислительные сис... X Вычислительные системы... X Google Календарь - Недели X Сибирский государственный... X Meet – bsr-evjo-cqj X

https://portal.sibadi.org/mod/assign/view.php?id=97802

Бречка Денис Михайлович

Лабораторная работа 4. Сумматоры

Задание:

Задание 1. Провести моделирование полусумматора (рис. 2.16 б).
Задание 2. Провести моделирование полного одноразрядного сумматора, выполненного на основе двух полусумматоров (рис. 2.17).
Задание 3. Провести моделирование трехразрядного сумматора с последовательным переносом (см. разд. 2.4).
Задание 4. Провести моделирование сумматора на 4 разряда из библиотеки LogiSim.
Задание 5. Провести моделирование вычитателя на 4 разряда из библиотеки LogiSim.

Проект Logisim.

Теоретические сведения по моделированию простейших комбинационных устройств (пт. 2.3).

В качестве ответа на задание загрузите отчет о проделанной работе.

Специальные возможности

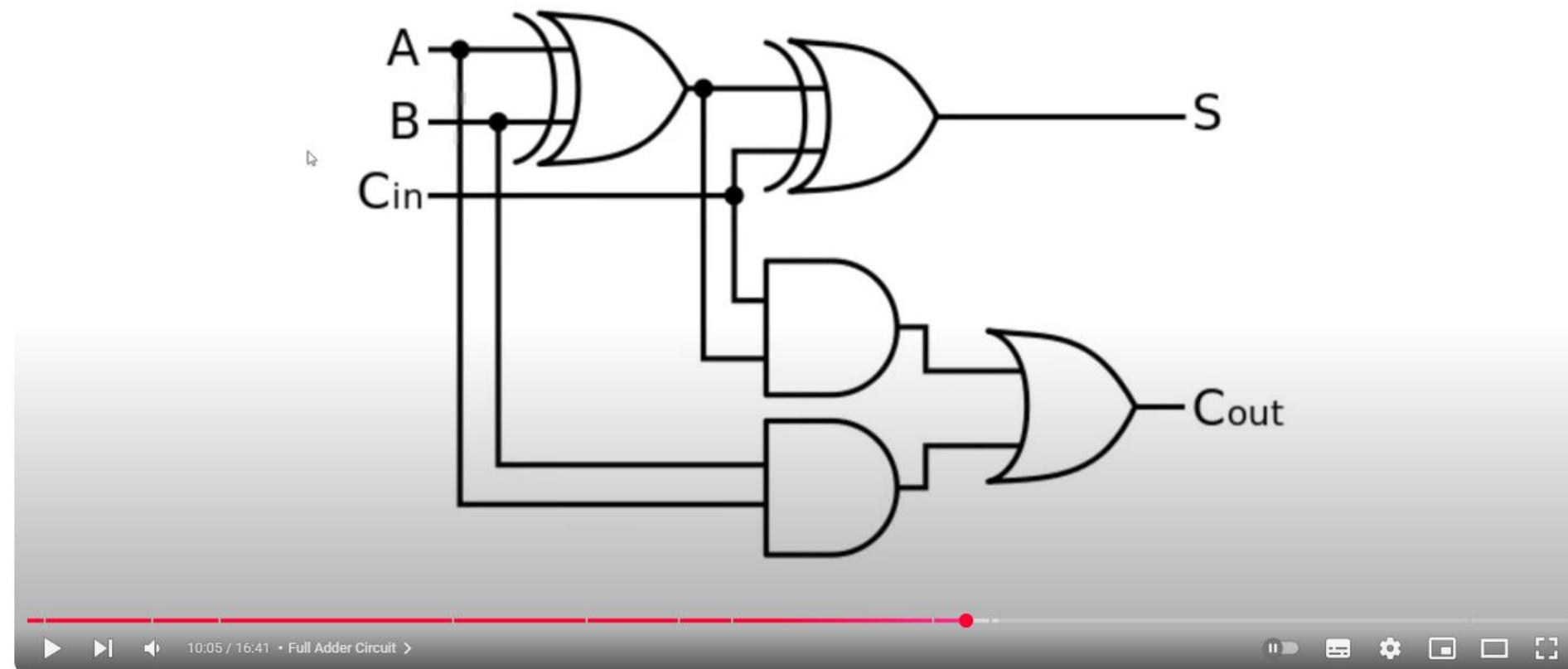
ЗАПУСТИТЬ ATBAR

Статистика

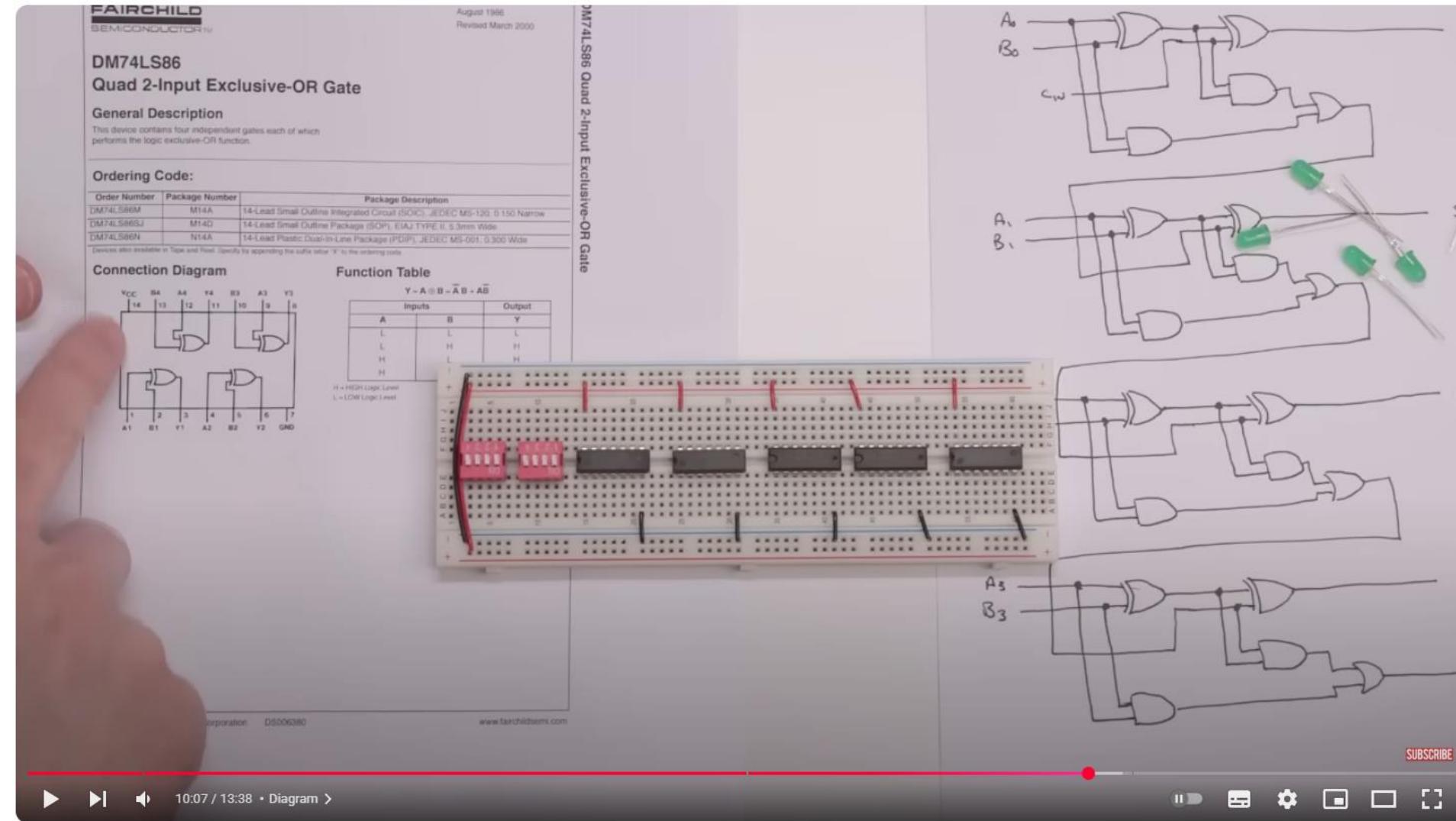
1

Лабораторная работа №4. Сумматоры (2021)
https://www.youtube.com/watch?v=uOWPh8_YKvQ

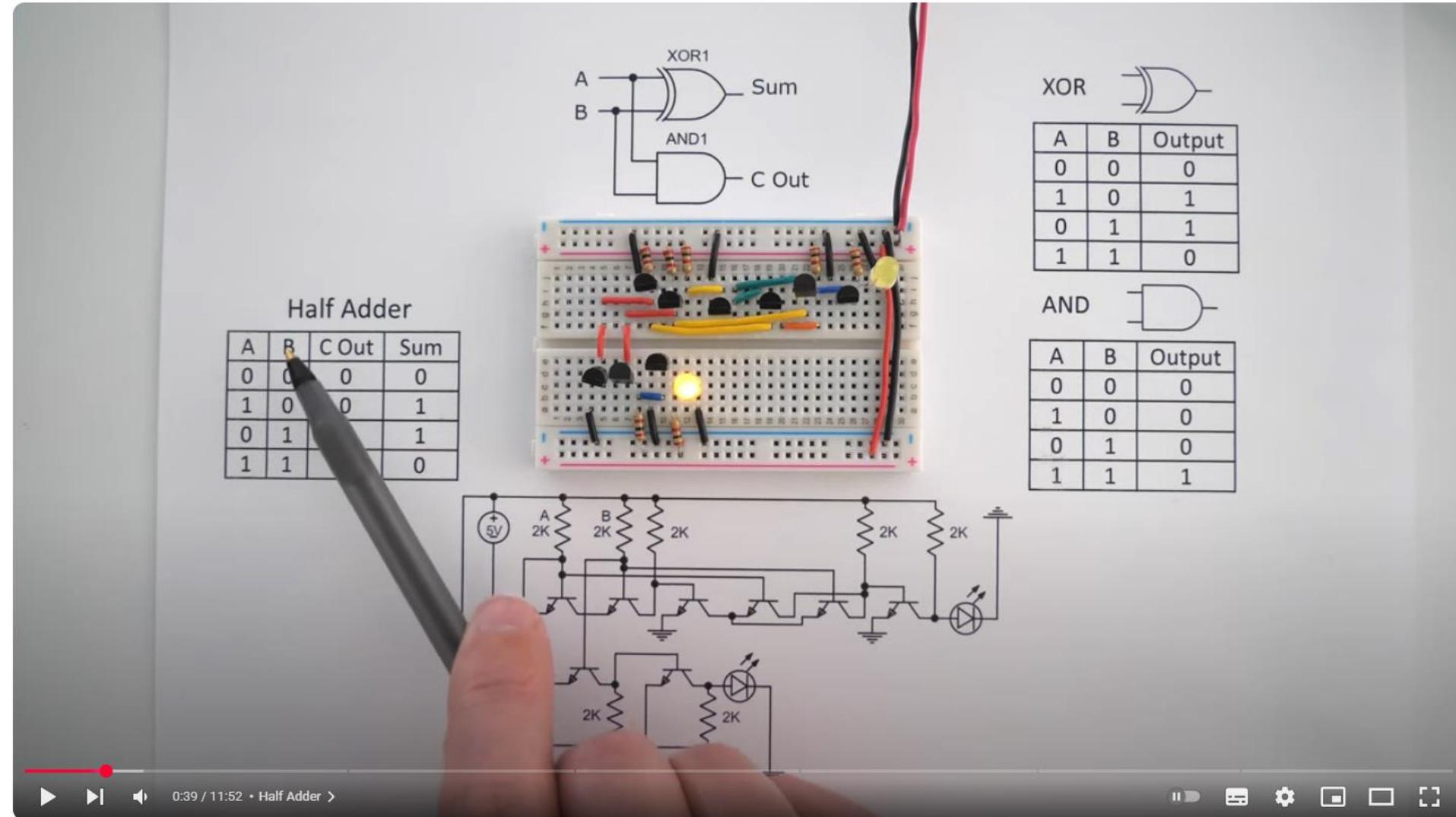
Full Adder (One implementation)



Half Adders and Full Adders Beginner's Tutorial (2020)
Полусумматоры и сумматоры для начинающих (2020)
<https://www.youtube.com/watch?v=ecn-8iGDRSo>



Learn how computers add numbers and build a 4 bit adder circuit (2015)
 Узнайте, как компьютеры складывают числа и создают схему 4-разрядного сумматора (2015)
<https://www.youtube.com/watch?v=wvJc9CZcvBc>



Full Adder (2022)
Полный сумматор (2022)
https://www.youtube.com/watch?v=mZ_2406nCrE

The screenshot shows a YouTube video player. On the left is a video frame of a man with a beard and grey hair, wearing a light purple shirt. To his right is a whiteboard with handwritten notes. At the top left of the whiteboard is a truth table for a half-adder:

C_i	A	B	Σ	C_o
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

Below the truth table is a logic circuit diagram for a half-adder, consisting of an AND gate (A and B) and an OR gate (the sum output Σ). A carry output C_o is also shown.

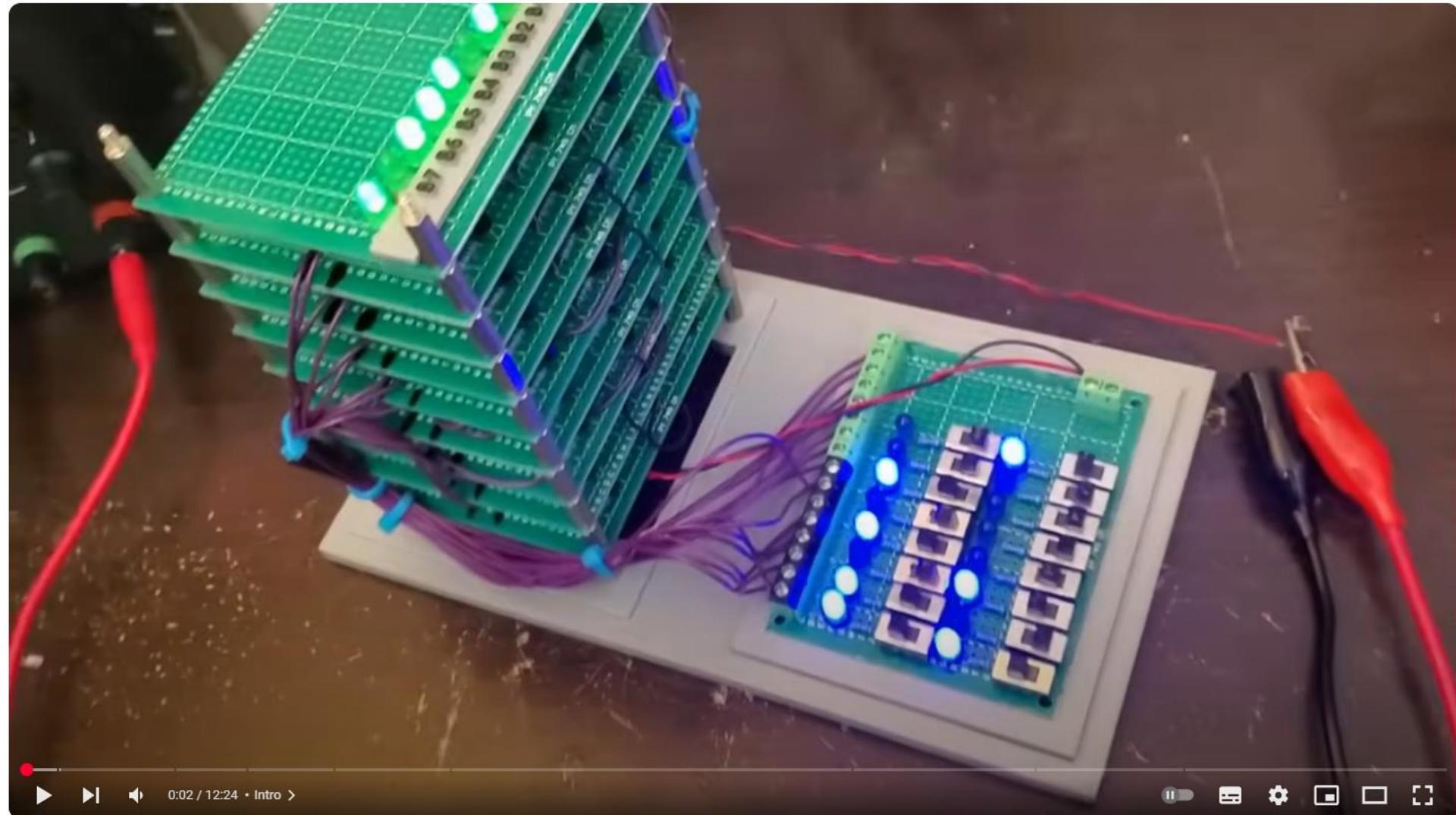
Handwritten text below the circuit diagram reads "Half-adder".

Below the half-adder section is another truth table for a full-adder, with columns for C_o , A , B , and C_i . The sum column Σ is identical to the one above. The carry output C_o is highlighted with colored circles and arrows, showing its logic expression: $C_o = AB + AC_i + BC_i$.

At the bottom right of the whiteboard is a handwritten conversion of binary numbers: $10111 = 11001 + 1100$.

The YouTube player interface at the bottom includes a progress bar, a timestamp of 8:19 / 19:52, and standard control buttons.

Combinational Devices 1: Half Adder and Full Adder (2020)
 Комбинационные устройства 1: Половинный сумматор и полный сумматор (2020)
<https://www.youtube.com/watch?v=dNDo4wAJBio>



8-Bit Adder built from 152 Transistors (2021)
8-разрядный сумматор, построенный на 152 транзисторах (2021)
<https://www.youtube.com/watch?v=X3IB1pVowlo>

Half Adder



A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

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

▶ ▶ 🔍 1:31 / 14:19 • Half Adder Circuit >

ALL ABOUT ELECTRONICS



Half Adder and Full Adder Explained | The Full Adder using Half Adder (2022)
 Объяснены половинный и полный сумматоры | Полный сумматор с использованием половинного сумматора (2022)
<https://www.youtube.com/watch?v=5XbRIVWFRlw>

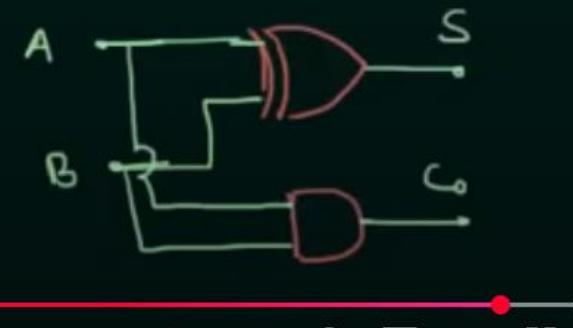
Half Adder [comb. cts]

- * Single bit no's
 - * it does not take carry from previous sum



A	B	C	D
0	0	0	0
0	1	1	0
1	0	1	0

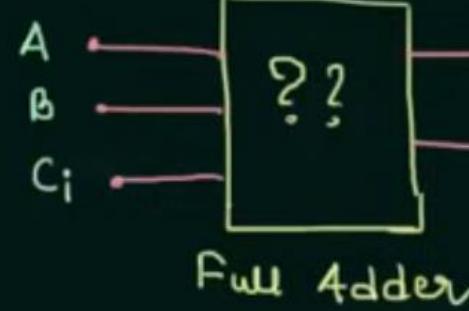
$$S = A \oplus B$$



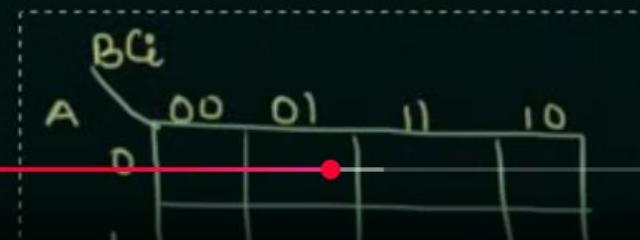
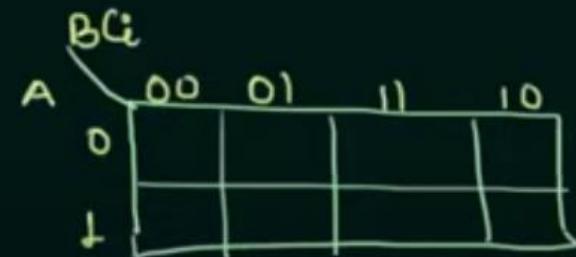
Half Adder (2014)
Полу сумматор (2014)
<https://www.youtube.com/watch?v=aLUY-s7LSns>

Full Adder [Part 2]

A	B	C _i	S	C _o
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	+ 1 = 0	0	1
1	+ 1 = 0	0	0	1
1	+ 1 = 0	1	1	1



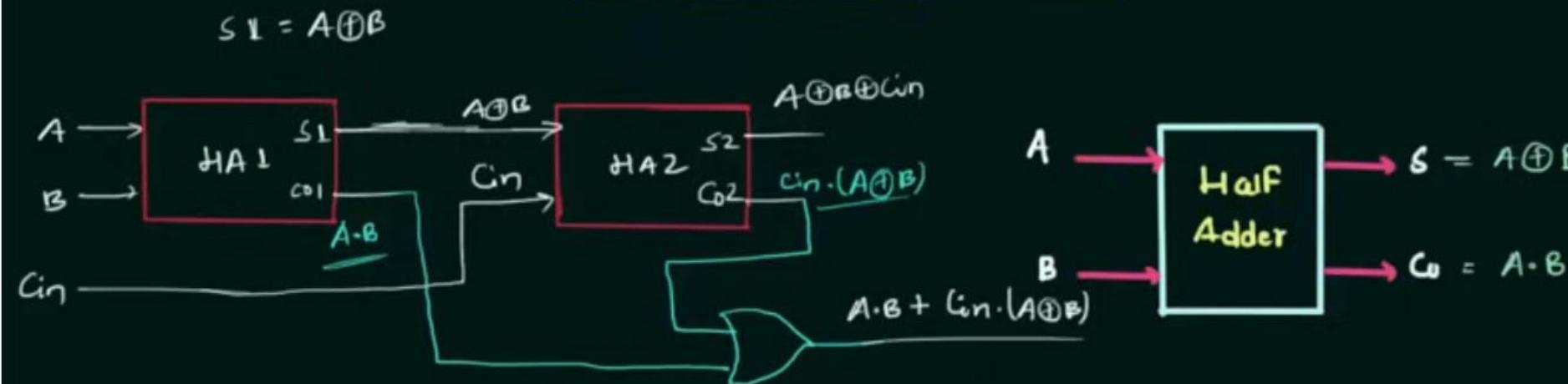
$$\begin{array}{r} \gamma = 2 \\ 2 = (2)_{10} \\ \text{Full Adder} \\ \begin{array}{r} A \\ B \\ C_i \end{array} \quad \begin{array}{r} S \\ C_o \end{array} \\ \hline \begin{array}{r} S = 0 \\ C_o = 1 \end{array} \end{array}$$



$$\begin{array}{r} \text{decimal } (\gamma = 10) \\ 13 + 6 \\ \hline 19 \end{array}$$

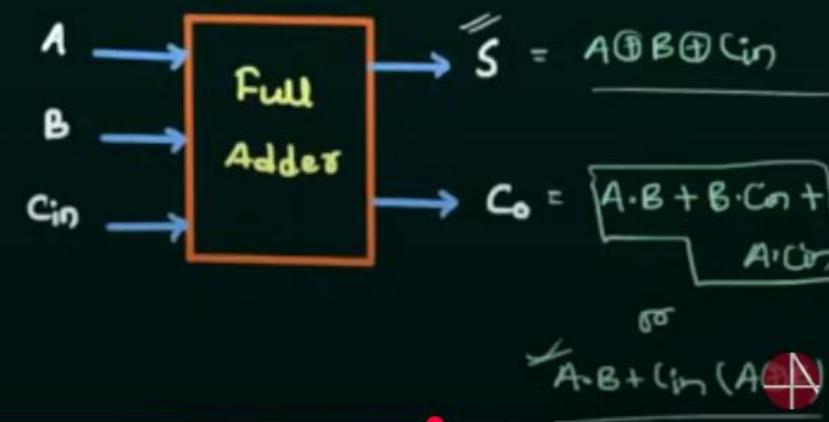
4

Full Adder using Half Adder

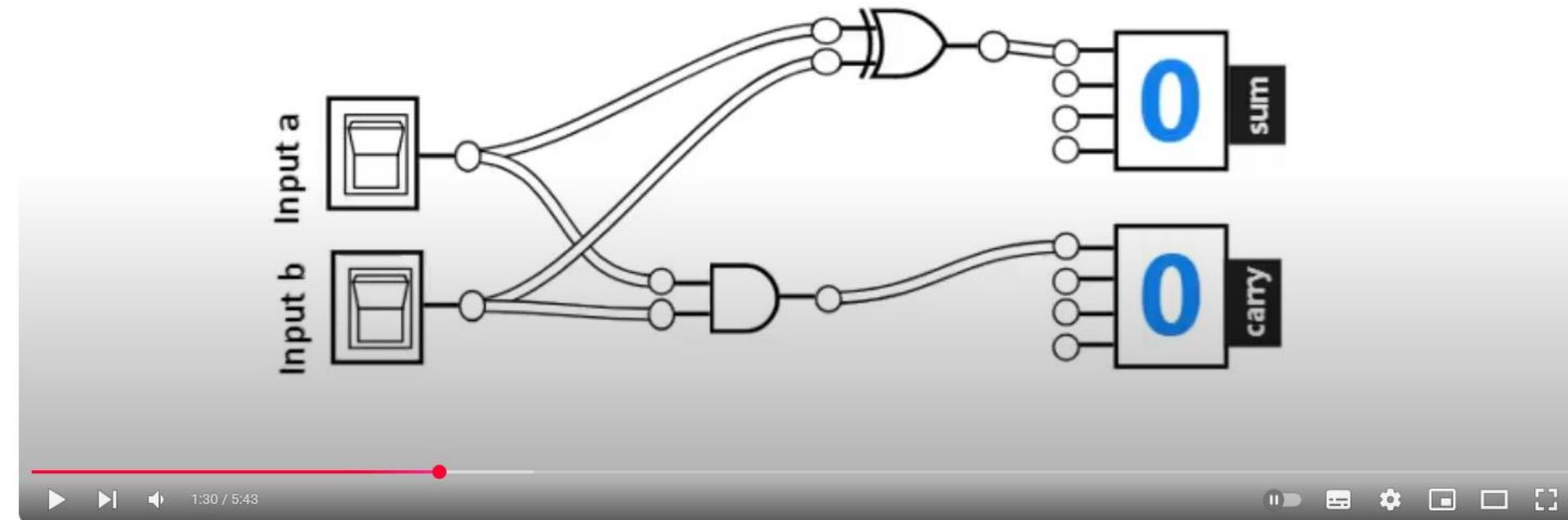


$$A \cdot B + C_{in} \cdot (A \oplus B) = A \cdot B + C_{in} \cdot (AB' + A'B)$$

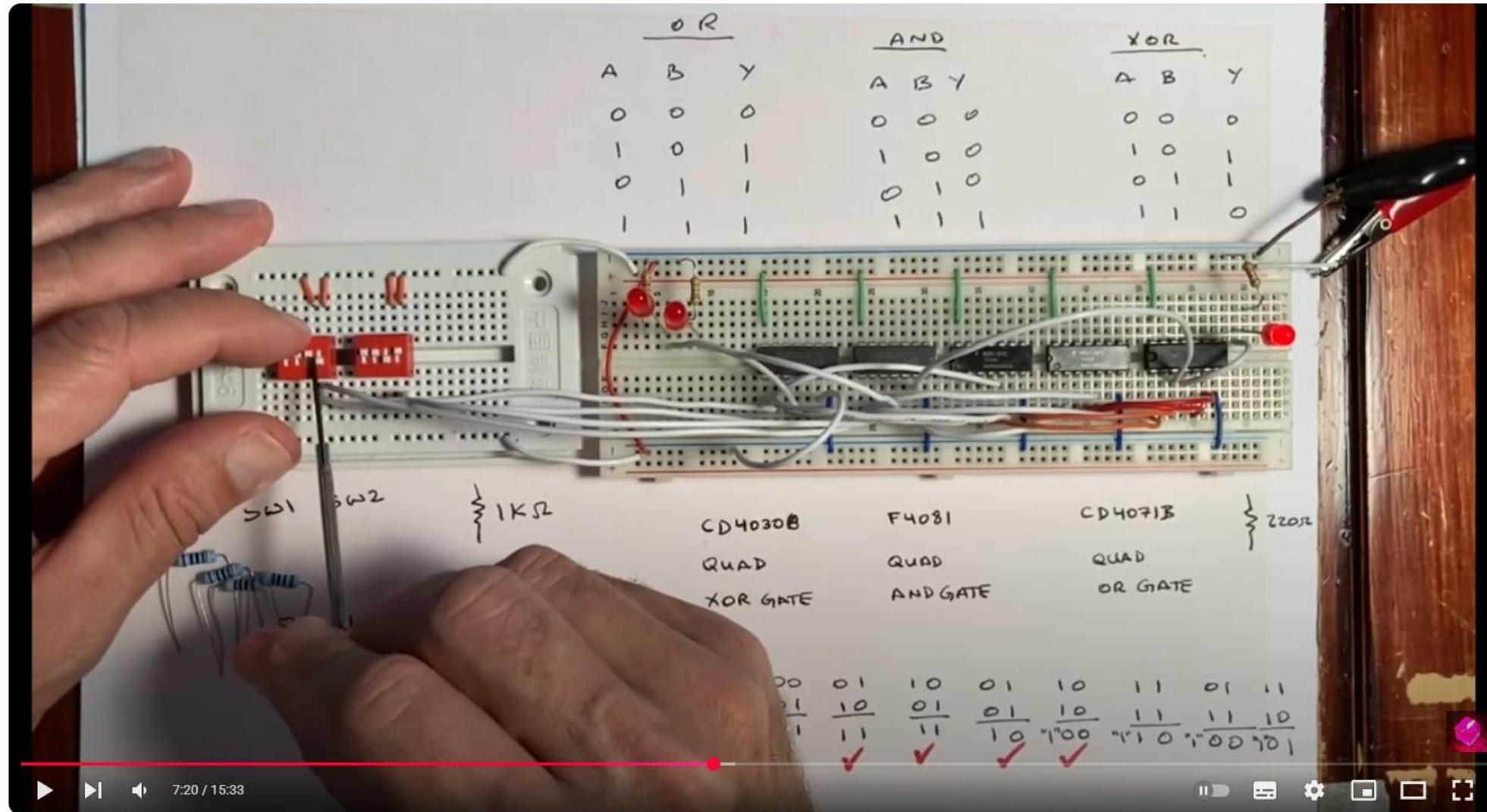
$$\begin{aligned} X + \cancel{(1)}X + X^T &= \cancel{A}B + \cancel{A}B' C_{in} + A'B C_{in} \\ &= A(B + \cancel{B}C_{in}) + A'B C_{in} \\ &= A(B + C_{in}) + A'B \end{aligned}$$



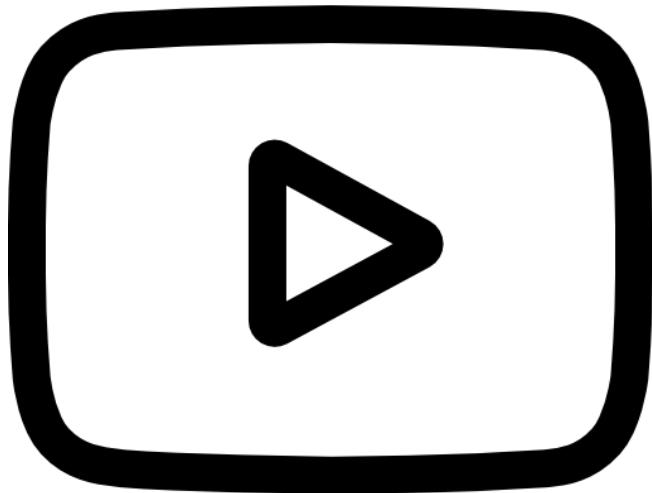
- Note that there are two inputs. There are also two outputs; sum and carry. If the sum of the numbers being added is greater than 1, the number gets carried over into the 'carry' output.



Binary Addition, Half-Adder, Full-Adder Logic Gate Circuits CLEARLY EXPLAINED (2018)
Двоичное сложение, полусумматор, сумматор. Логические схемы с понятным объяснением (2018)
https://www.youtube.com/watch?v=hExfiB_UFzA

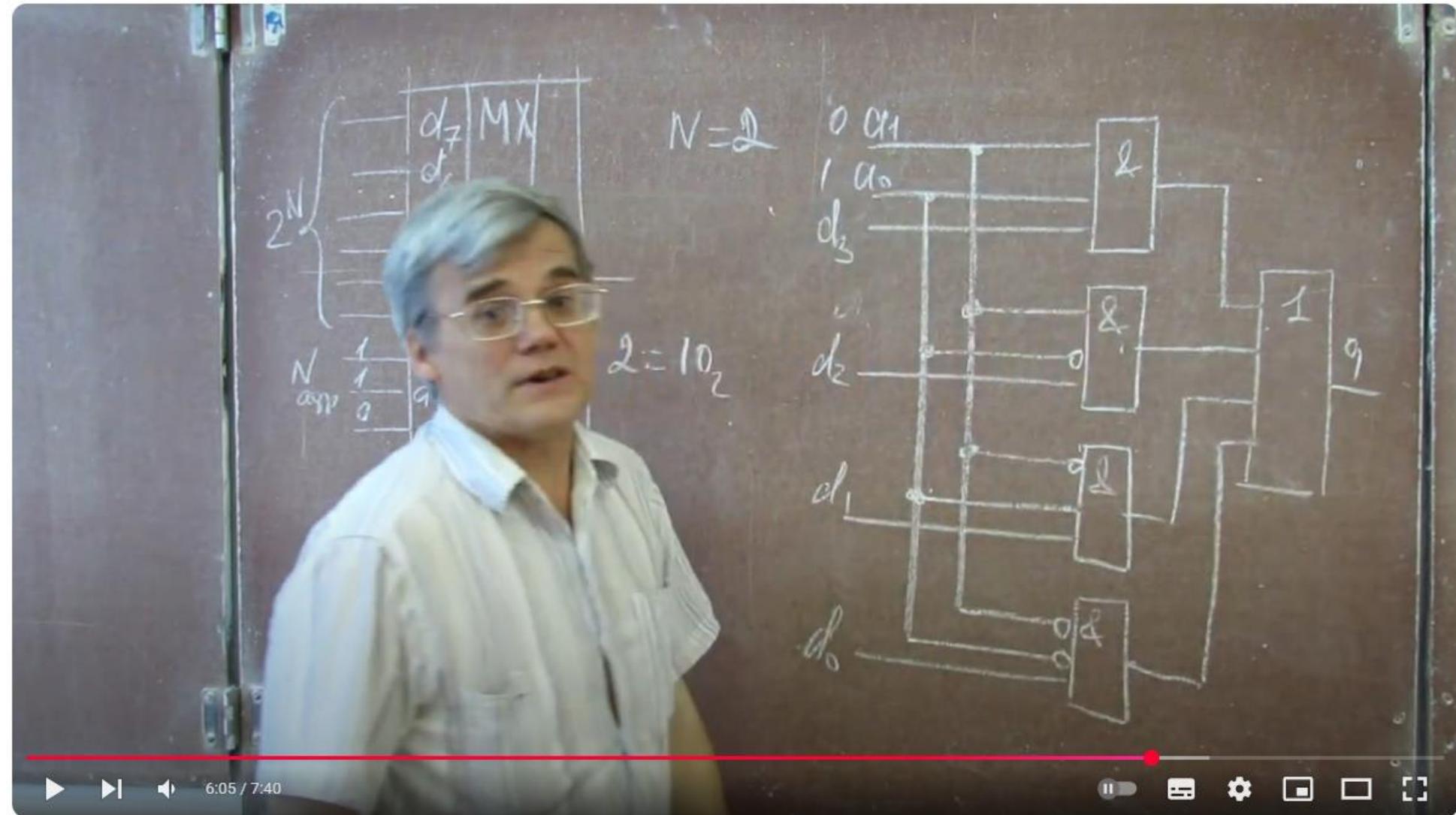


4-Bit Adder built from logic gates (2021)
4-битный сумматор, построенный на логических вентилях (2021)
<https://www.youtube.com/watch?v=QDjNPL6aGp0>

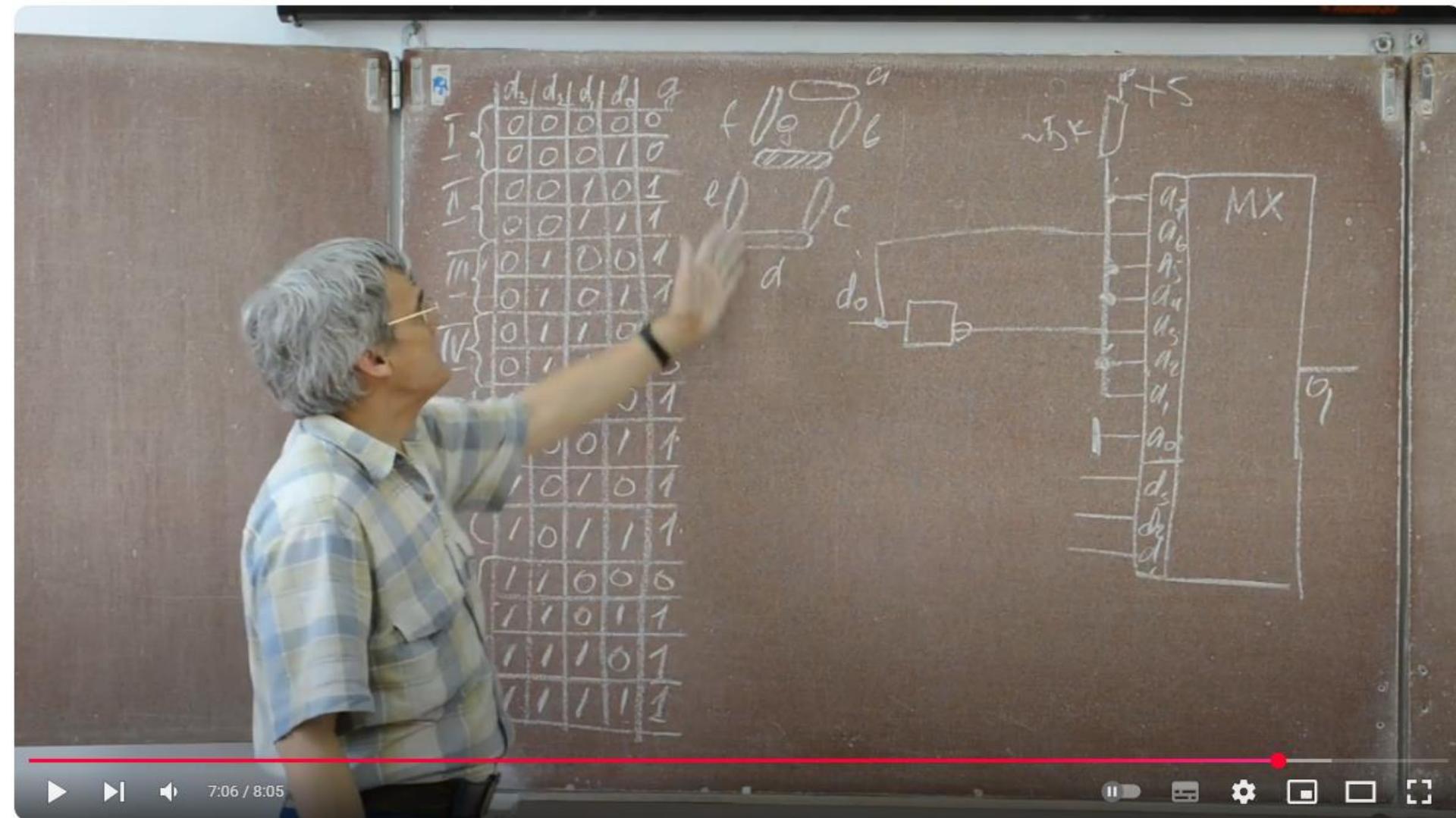


**Мультиплексоры,
Демультиплексоры**





Лекция 91. Мультиплексор (2013)
<https://www.youtube.com/watch?v=LX5osXWp5il>



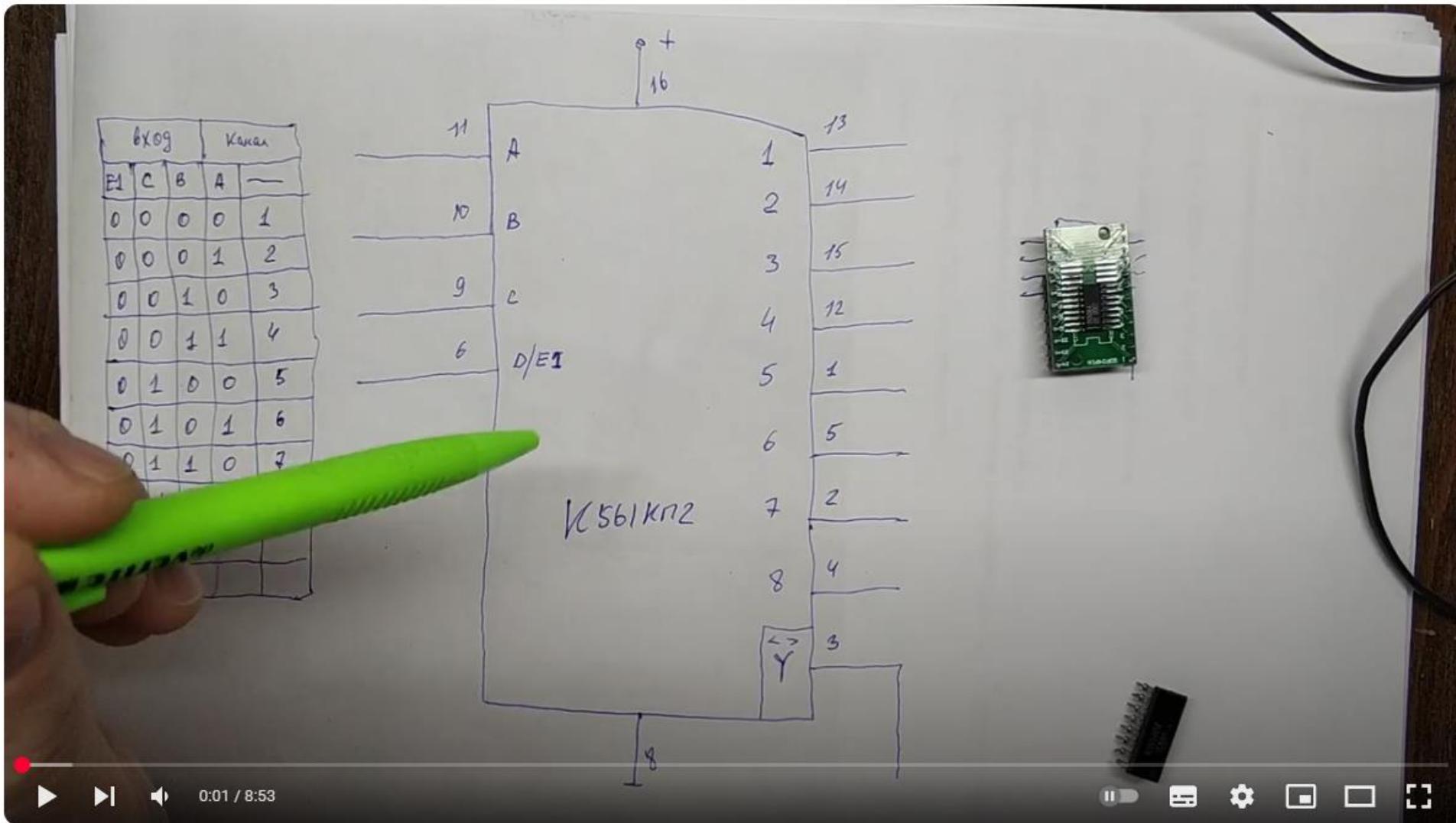
Лекция 94. Применение мультиплексора (2013)

<https://www.youtube.com/watch?v=Wj171iJQGYs>

YouTube 3:00



Мультиплексоры принцип действия (2012)
<https://www.youtube.com/watch?v=jd2U83In190>



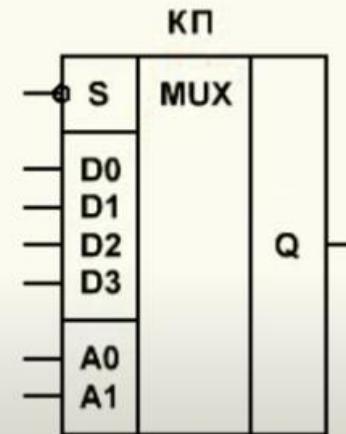
Что такое мультиплексор и как он работает (2022)
<https://www.youtube.com/watch?v=RhhO7jedMi8>

Мультиплексоры

мультиплексор (multiplexer)

селектор данных (data selector)

коммутирующие



D (data - данные, информация)- информационные входы

A (address) - адресные входы

S - стробирующий (разрешающий) вход

strobe - разрешение принимать и обрабатывать сигналы,
поступающие на другие входы

Q (quit) - выход

селектор-мультиплексор (или мультиплексор-селектор)



Вideoурок 3.22. Мультиплексоры. Часть 1 (2023)
<https://www.youtube.com/watch?v=tgJ9nYP55c8>

Мультиплексоры

активный разрешающий

0 → S
0 → D0
1 → D1
0 → D2
0 → D3
0 → A0
1 → A1

код цифры 2 (адрес входа D2)

1. Мультиплексоры 4x1

Десятичные цифры	Двоичные коды
0	00
1	01
2	10
3	11

D (data - данные, информация)- информационные входы

A (address) - адресные входы

S - стробирующий (разрешающий) вход
strobe - разрешение принимать и обрабатывать сигналы, поступающие на другие входы

Q (quit) - выход

одноразрядные

мультиплексор-селектор KP1533КП2

0 1G
1C0
1C1
1C2
1C3
A
B
2C0
2C1
2C2
2C3
0 2G

1Y
2Y

SN74ALS153

▶
▶
◀
2:57 / 10:33
||
▢
⚙️
▢
▢

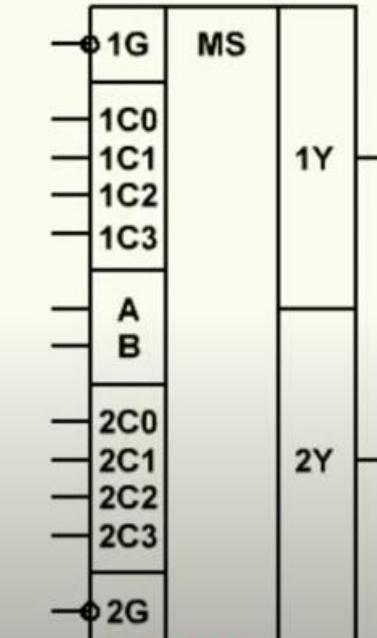
Вideoурок 3.23. Мультиплексоры. Часть 2 (2023)
<https://www.youtube.com/watch?v=jKy6o1gAk0l>

Мультиплексоры

1. Мультиплексоры 4x1

мультиплексор-селектор

KP1533КП2

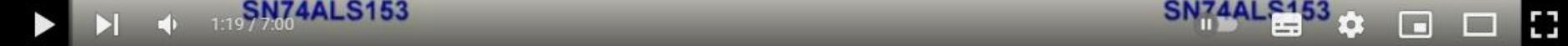
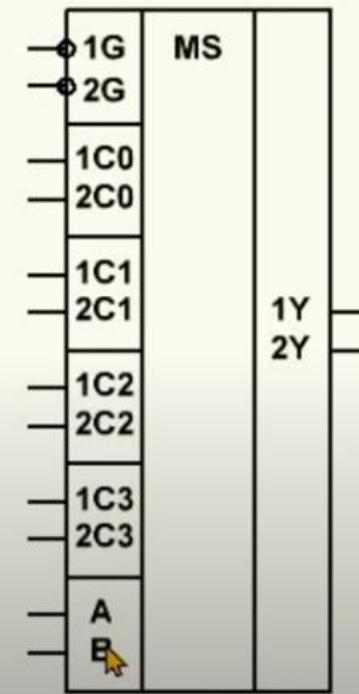


два
одноразрядных

Десятичные цифры	Двоичные коды
0	00
1	01
2	10
3	11

один 2-разрядный

KP1533КП2



Вideoурок 3.24. Мультиплексоры. Часть 3 (2023)
<https://www.youtube.com/watch?v=bN1C1kSIPGk>

Мультиплексоры

2. Мультиплексоры 8x1

мультиплексор-селектор KP1533КП7

SN74ALS151

3. Мультиплексоры 16x1

Десятичные цифры	3-разрядный двоичный код
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

комплементарные
complementary - добавочный, дополнительный

Q KP1533KP5 Z - состояние

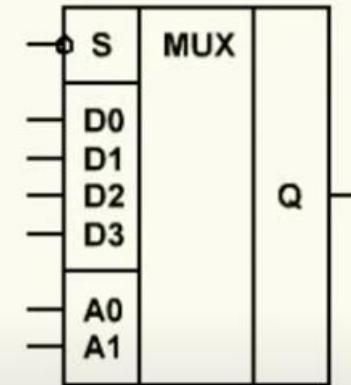
K155КП1 SN74150N

SN74150N

Вideoурок 3.25. Мультиплексоры. Часть 4 (2023)
<https://www.youtube.com/watch?v=bUpFV3TJXjA>

Структура мультиплексоров

Мультиплексор 4x1



Адресные входы		Стробирующий вход	Выход
A1	A0	\bar{S}	Q
x	x	1	0
0	0	0	D0
0	1	0	D1
1	0	0	D2
1	1	0	D3

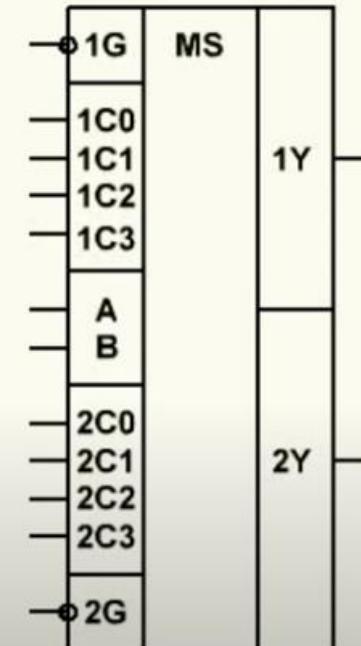
3 элемента НЕ

4 элемента 4И

1 элемент 4ИЛИ

$$Q = \bar{A}_1 \cdot \bar{A}_0 \cdot (\bar{S}) \cdot D_0 \vee \bar{A}_1 \cdot A_0 \cdot (\bar{S}) \cdot D_1 \vee A_1 \cdot \bar{A}_0 \cdot (\bar{S}) \cdot D_2 \vee A_1 \cdot A_0 \cdot (\bar{S}) \cdot D_3$$

**мультиплексор-селектор
KP1533КП2**



▶ ▶ ⏴ 6:43 / 13:57

⏸ ⏴ ⏵ ⏷ ⏸ ⏹

Видеоурок 3.26. Структура мультиплексоров (2023)
<https://www.youtube.com/watch?v=BMNQku6S4Dc>

Определение

Мультиплексор представляет собой комбинированное цифровое устройство, обеспечивающее поочередную передачу на один выход нескольких входных сигналов. Он позволяет передавать (коммутировать) сигнал с желаемого входа на выход.

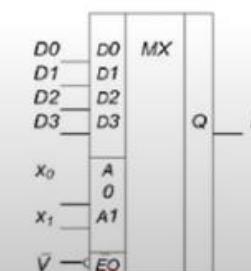
Мультиплексор

Мультиплексор – это комбинационная многовходовая схема с одним выходом. Входы мультиплексора подразделяются на информационные X_0, X_1, \dots, X_{n-1} и управляющие (адресные) A_0, A_1, \dots, A_{k-1} . Обычно $2^k = n$, где k и n – число адресных и информационных входов соответственно. Двоичный код, поступающий на адресные входы, определяет (выбирает) один из информационных входов, значение переменной с которого передается на выход y , т.е. мультиплексор реализует функцию:

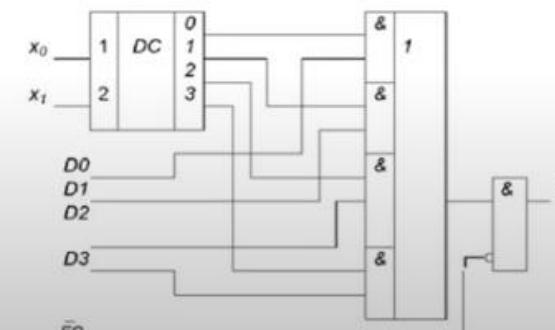
$$y = X_i, \text{ если } \sum_{i=0}^{k-1} A_i \cdot 2^i = i$$

V	x_1	x_0	D_3	D_2	D_1	D_0	f
0	0	0	x	x	x	0/1	0/1
0	0	1	x	x	0/1	0	0/1
0	1	0	0/1	x	x	x	0/1
0	1	1	0/1	x	x	x	0/1
1	x	x	x	x	x	x	0

таблица истинности



условное обозначение



логическая схема

A1	A0	Y
0	0	D0
0	1	D1
1	0	D2
1	1	D3

Таблица истинности для MS 4-1



Мультиплексоры принцип работы, наращивание (2022)
<https://www.youtube.com/watch?v=rJaJsRQ1eUs>

Applications Places Activities LibreOffice Writer en Thu May 16 02:19 ● 100 %

File Edit View Insert Format Styles Table Form Tools Window Help

Table Contents Liberation Serif 12 Header (Default Style) +

Results

Inputs		Data					Enabled	Expected Outputs	Experimental Outputs
Address		C_{13}	C_{12}	C_{11}	C_{10}	V	Y_1	Y_2	
B	A								
*	*	*	*	*	*				
0	0	*	*	*	*				
0	0	0	0	0	0				
0	1	*	*	*	*				
0	1	0	0	0	0				
1	0	*	*	0	0				
1	0	0	0	1	1				
1	1	0	0	*	*				
1	1	1	1	0	0				

I

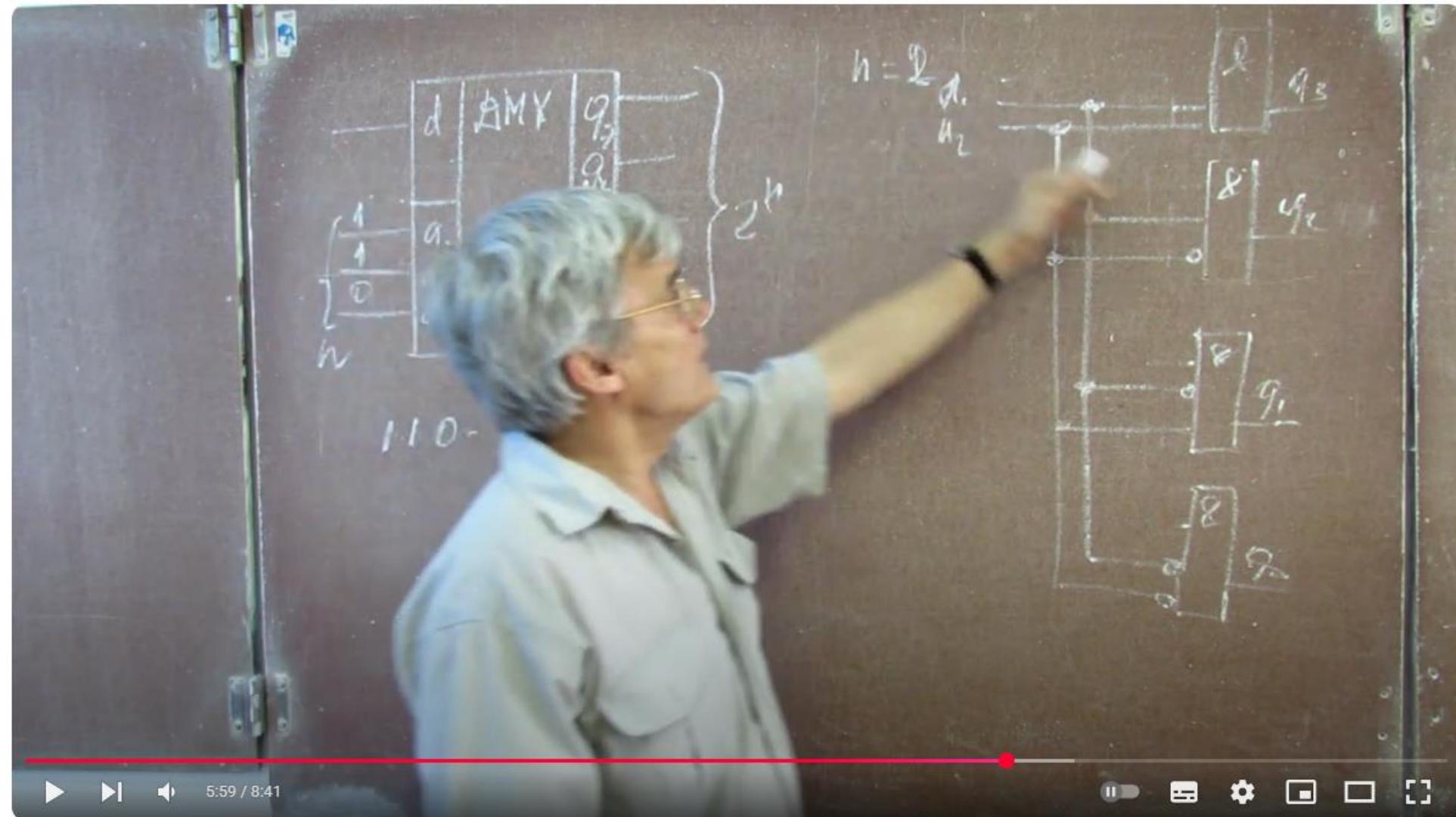
0091. Мультиплексор (2019)

<https://www.youtube.com/watch?v=EodHGKQAWBg>



Мультиплексор. Декодер. Светодиодный индикатор

Verilog. Мультиплексор. Декодер. Семисегментный индикатор (2019)
<https://www.youtube.com/watch?v=aLgDQIL-e2Y>



Лекция 88. Демультиплексор (2013)
<https://www.youtube.com/watch?v=FBiRItM8uFY>



Демультиплексоры, принцип действия (2014)
<https://www.youtube.com/watch?v=tzPfUaqZcgs>

Тема: Демультиплексоры**План :**

- 1 Назначение демультиплексора.
- 2 Принцип работы демультиплексора.
- 3 Пример функциональной схемы, графическое изображение демультиплексора.

Демультиплексорами называются устройства, которые позволяют подключать один вход к нескольким выходам. Демультиплексор можно построить на основе точно таких же схем логического "И", как и при построении мультиплексора. Существенным отличием от мультиплексора является возможность объединения нескольких входов в один без дополнительных схем. [1]

Функциональная схема и графическое изображение демультиплексора приведены на рисунке 9.1 (пример). В этой схеме для выбора выхода(ов) демультиплексора, как и в мультиплексоре, используется двоичный дешифратор.



Рисунок 9.1 мультиплексор а) функциональная схема б) графическое изображение

**Тема: Демультиплексор****Программа: Logo!Soft Comfort**

Теоретические сведения приведены в лекционных материалах, рис. DMS.1.

ЗАДАНИЯ.

Задание 1. Составить в Logo!Soft Comfort логическую схему демультиплексора, рис. DMS.1.

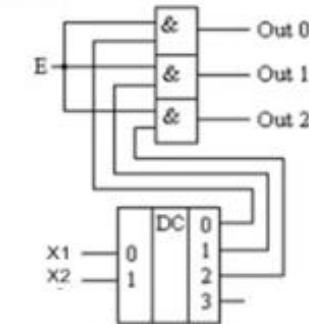


Рисунок DMS.1 – Функциональная схема демультиплексора

Описание процесса.

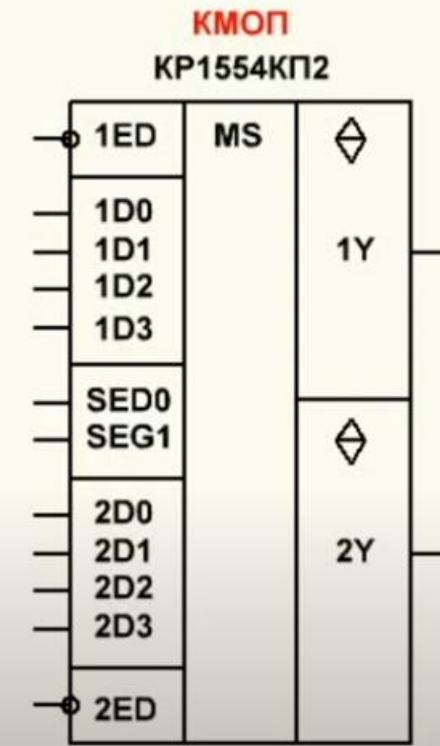
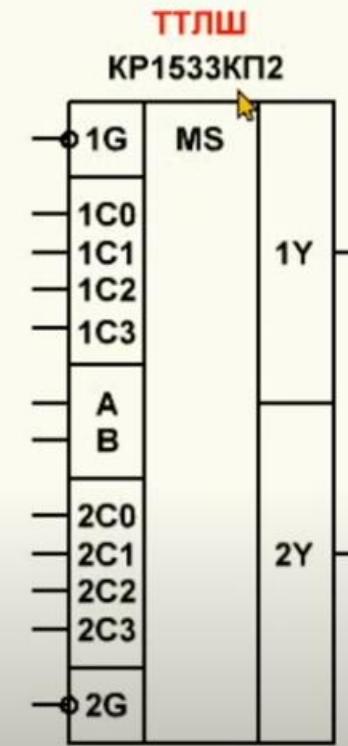
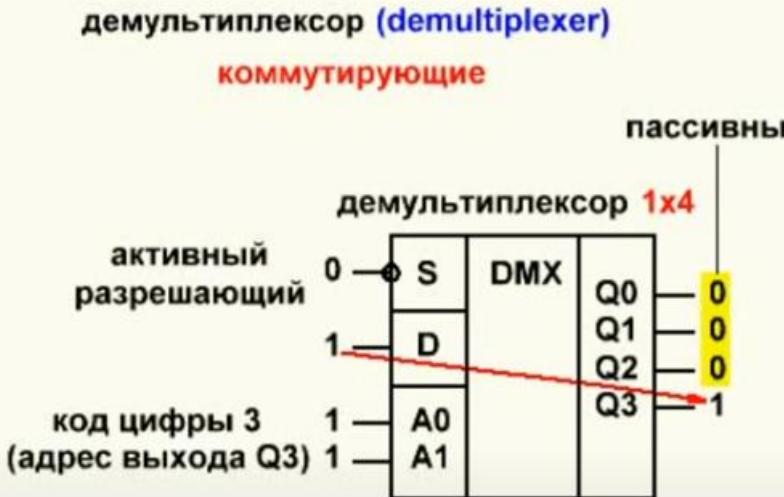
На адресных входах X1, X2 демультиплексора устанавливаются (выбираются) адреса коммутируемой линии в соответствии с таблицей работы дешифратора.

Таблица работы дешифратора

X1	X2	Выходы дешифратора		
		«Q0» (Out0)	«1» (Out1)	«2» (Out2)
0	0	1	0	1
0	1	0	1	1
1	0	1	1	0
1	1	1	1	1

Демультиплексор, простейшая схема (2020)
<https://www.youtube.com/watch?v=jlaG1QmMqmQ>

Демультиплексоры

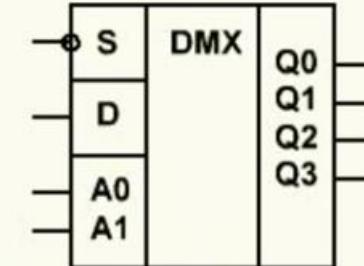


▶ ▶ ⏪ 5:40 / 8:35

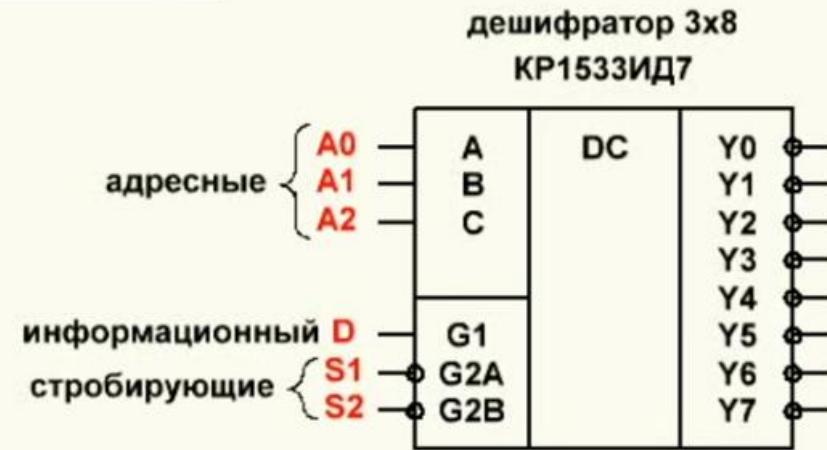
Вideoурок 3.27. Демультиплексоры. Часть 1 (2023)
https://www.youtube.com/watch?v=2y_cAgTZPH0

Демультиплексоры

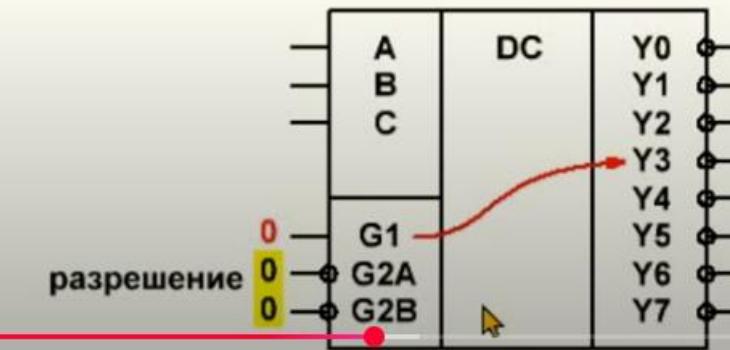
2. Микросхемы **любой** структуры



дешифраторы, имеющие хотя бы 1 разрешающий вход



адресные
информационный
стробирующие



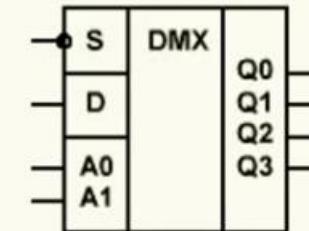
Десятичные цифры	Двоичные коды
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

▶ ▶ ⏪ 3:51 / 13:23

Вideoурок 3.28. Демультиплексоры. Часть 2 (2023)
<https://www.youtube.com/watch?v=YXxCfzU9uyo>

Структура демультиплексоров

демультиплексор 1x4



Адресные входы		Стробирующий вход	Выходы			
A1	A0	\bar{S}	Q3	Q2	Q1	Q0
x	x	1	0	0	0	0
0	0	0	0	0	0	D
0	1	0	0	0	D	0
1	0	0	0	D	0	0
1	1	0	D	0	0	0

$$Q_0 = \bar{A}_1 \cdot \bar{A}_0 \cdot (\bar{S}) \cdot D$$

$$Q_1 = \bar{A}_1 \cdot A_0 \cdot (\bar{S}) \cdot D$$

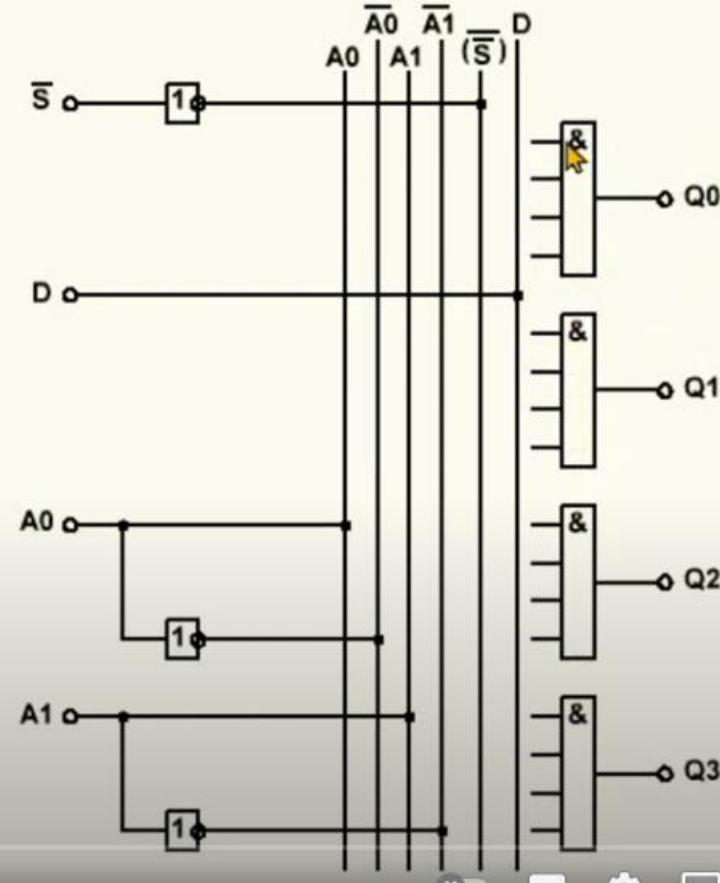
$$Q_2 = A_1 \cdot \bar{A}_0 \cdot (\bar{S}) \cdot D$$

3 элемента НЕ

4 элемента 4И

$$Q_3 = A_1 \cdot A_0 \cdot (\bar{S}) \cdot D$$

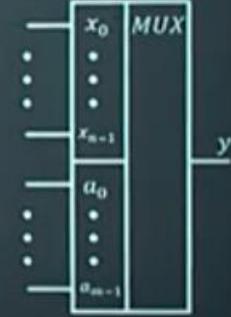
4:53 / 10:58



Видеоурок 3.29. Структура демультиплексоров (2023)
<https://www.youtube.com/watch?v=Rv5kCzSzmoA>

Мультплексор

Мультплексор (MUX, MS) – комбинационное устройство с n информационными входами (x_i), m адресными входами (a_i) и одним выходом (y), передающее информацию на выход от одного из информационных входов, адрес которого указывается на адресном входе.

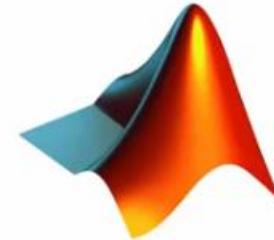


УГО мультиплексора

▶ ⏸ 0:06 / 2:56

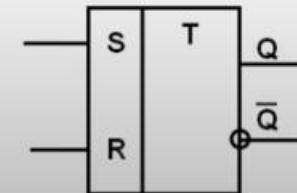
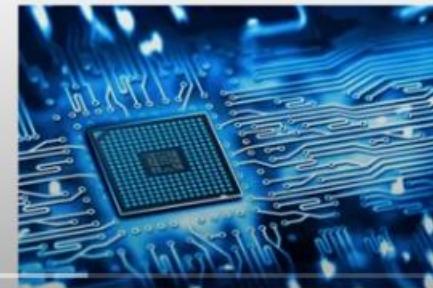
II ⏹ ⚙ □ □ □

4.3.2 Мультиплексоры и демультиплексоры (2017)
https://www.youtube.com/watch?v=x_sM-Fkg4h4



Основы цифровой схемотехники

07. Мультиплексоры и демультиплексоры



0:02 / 4:41

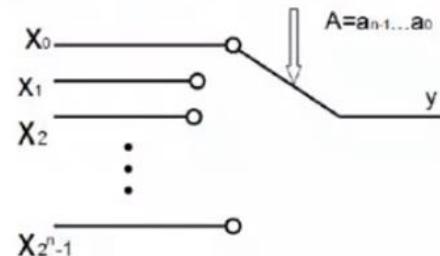


Основы цифровой схемотехники. 07. Мультиплексоры и демультиплексоры (2019)
<https://www.youtube.com/watch?v=XfQnEpJ0jLU>

Мультиплексоры

Мультиплексор (MUX) – коммутатор логических сигналов, обеспечивающий передачу информации, поступающей по нескольким линиям связи, на одну выходную линию. Выбор входной линии x_i осуществляется в соответствии с поступающим адресным входом. При наличии n адресных входов можно реализовать $N = 2^n$ комбинаций адресных сигналов, каждая из которых обеспечивает выбор одной из N входных линий.

Работу мультиплексора можно проиллюстрировать электромеханической схемой замещения:



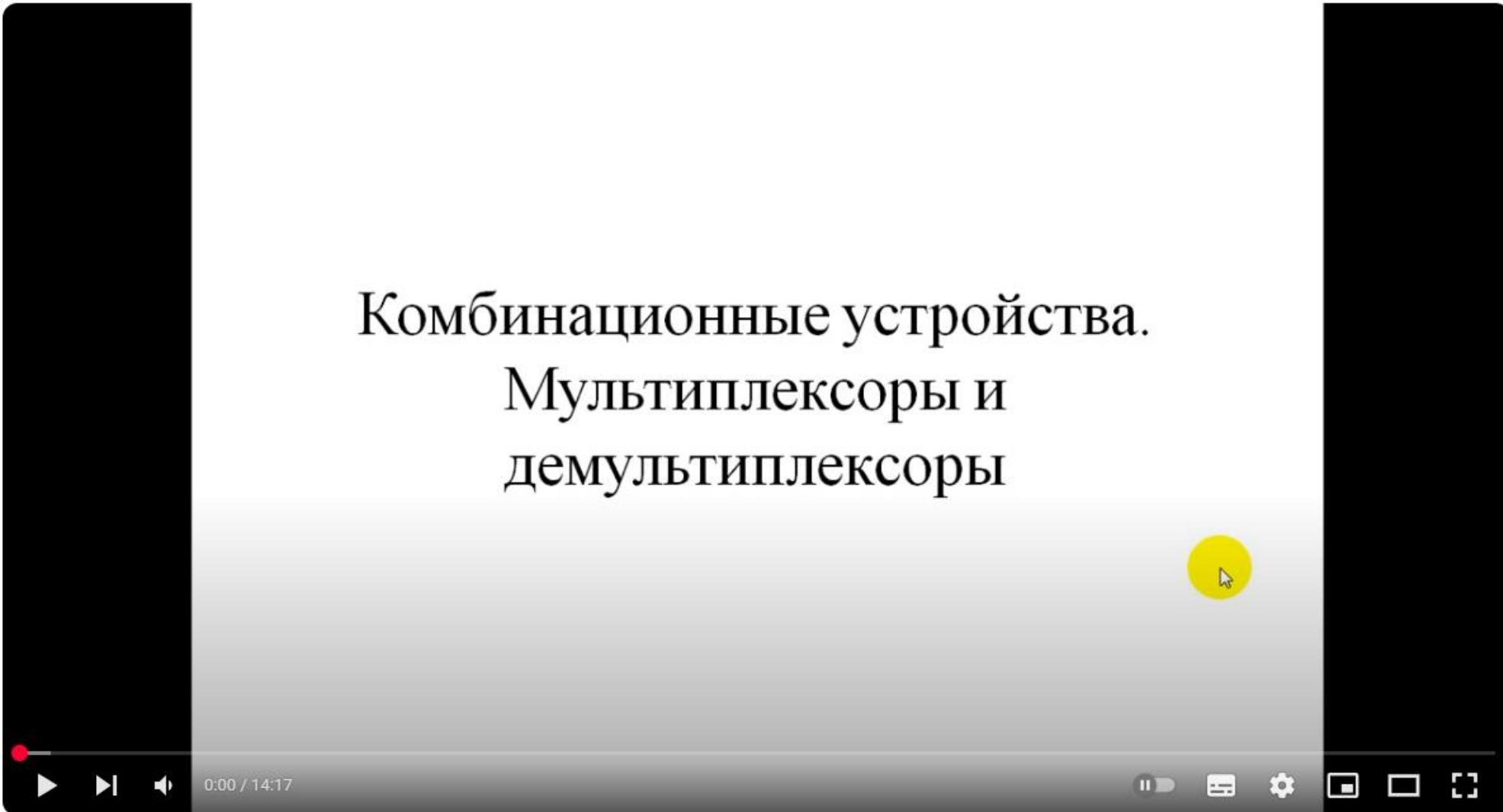
Работа мультиплексора описывается логической функцией:

$$y = x_0 \bar{a}_{n-1} \dots \bar{a}_1 \bar{a}_0 + x_1 \bar{a}_{n-1} \dots \bar{a}_1 a_0 + x_2 \bar{a}_{n-1} \dots a_1 \bar{a}_0 + \dots + x_{2^n-1} \bar{a}_{n-1} \dots a_1 a_0 = \bigvee_{i=0}^{2^n-1} x_i m_i ,$$

где m_i – минтерм « n » адресных переменных.

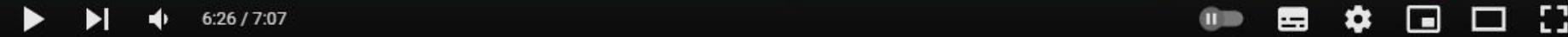
Адресный код A указывает номер информационного сигнала, передаваемого на выход.

The image shows a YouTube player interface. At the top left is the YouTube logo and the time '16:04'. Below the video frame, there is a progress bar with a red playhead at the beginning, indicating the video has just started. The video content itself is a slide from a presentation. It contains text, a diagram of a multiplexer circuit, and mathematical formulas. On the right side of the slide, there is a small icon of a lightbulb. At the bottom of the YouTube player, there are standard control buttons: back, forward, volume, and a settings gear icon. To the right of the player, there is a dark sidebar with a hand cursor icon and a small blue icon representing a brain or idea.

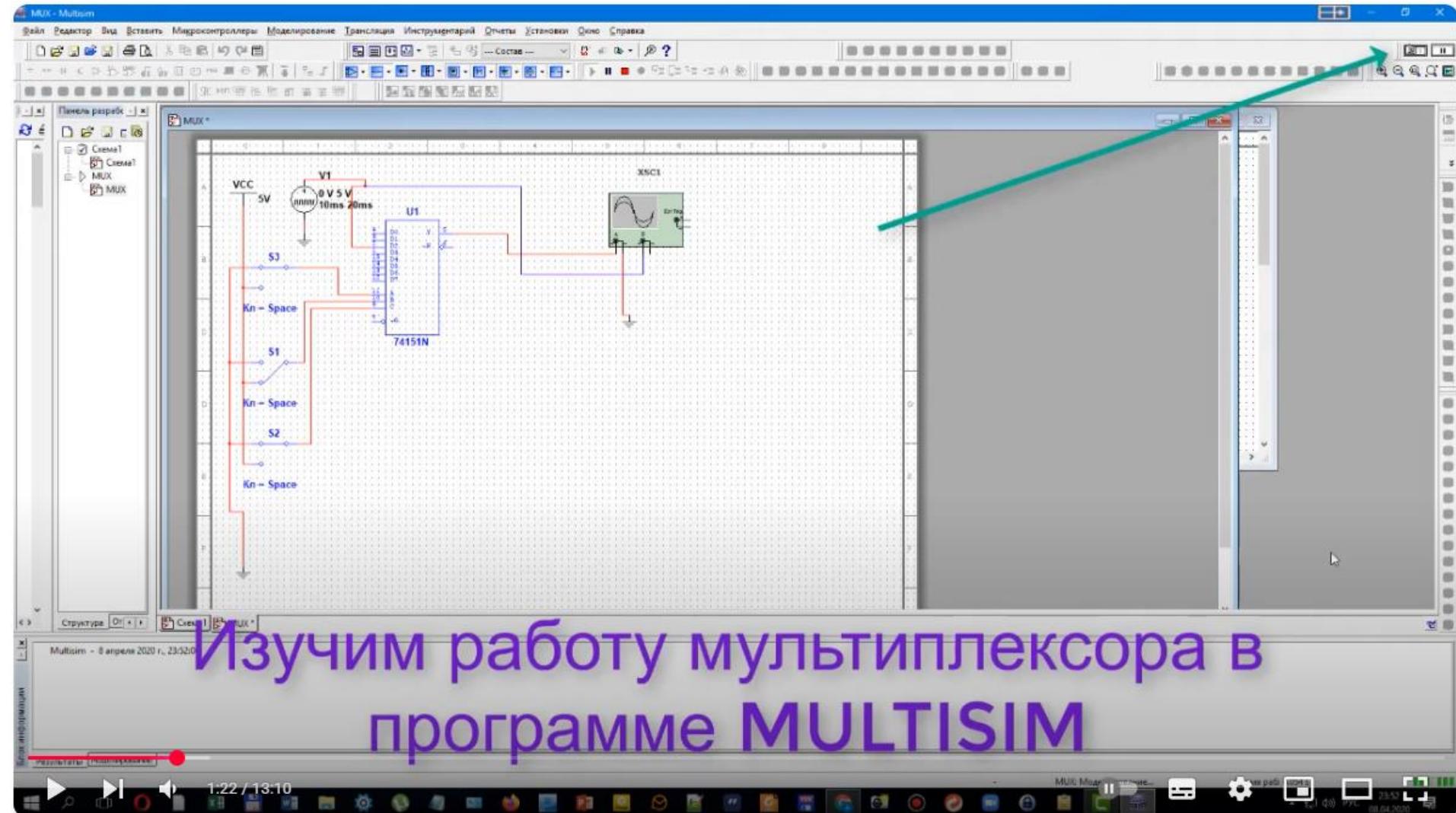


Мультиплексоры и демультиплексоры (2022)
<https://www.youtube.com/watch?v=-4XdhGPx5MY>

Мультиплексор и демультиплексор

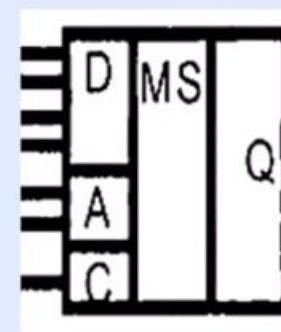


Мультиплексор и демультиплексор (2025)
<https://www.youtube.com/watch?v=3gOL8EqMs78>



Изучим работу мультиплексора в программе MULTISIM

Вideoурок на тему: "Мультиплексор и демультиплексор" (2020)
<https://www.youtube.com/watch?v=7o5tKpvVS2Q>



Мультиплексор – это функциональный узел, имеющий n адресных входов и $N=2^n$ информационных входов и выполняющий коммутацию на выход того информационного сигнала, адрес (т.е. номер) которого установлен на адресных входах.

Иначе мультиплексор – это адресный коммутатор. Мультиплексор обозначается MUX $N - 1$ или MS $N - 1$, т.е. коммутатор, имеющий N информационных входов и один выход. Мультиплексор переключает сигнал с одной из N входных линий на выходную линию.

Раздел 4 30 Мультиплексоры и демультиплексоры (2020)
<https://www.youtube.com/watch?v=IDUVVjxZntg>

Белорусско-Российский университет, Кафедра «Программное обеспечение информационных технологий»

Мультиплексоры и демультиплексоры.ppt [Режим совместимости] - Microsoft PowerPoint

Главная Вставка Дизайн Анимация Показ слайдов Рецензирование Вид

С текущего начала слайда Приватный показ Настройка демонстрации Скрыть слайд Использовать записанные времена Настройка Использовать текущее разрешение Показать презентацию на: Мониторы

Начать показ слайдов

Лабораторное занятие №5 Исследование функциональных схем мультиплексоров и демультиплексоров.

Цель работы: теоретически исследовать функциональные схемы мультиплексоров и демультиплексоров.

Оборудование: программа EWB 5.12 (Electronics Workbench), инструкционные карты.

Лабораторное занятие №5 Исследование функциональных схем мультиплексоров и демультиплексоров.

Цель работы: теоретически исследовать функциональные схемы мультиплексоров и демультиплексоров.

Оборудование: программа EWB 5.12 (Electronics Workbench), инструкционные карты.

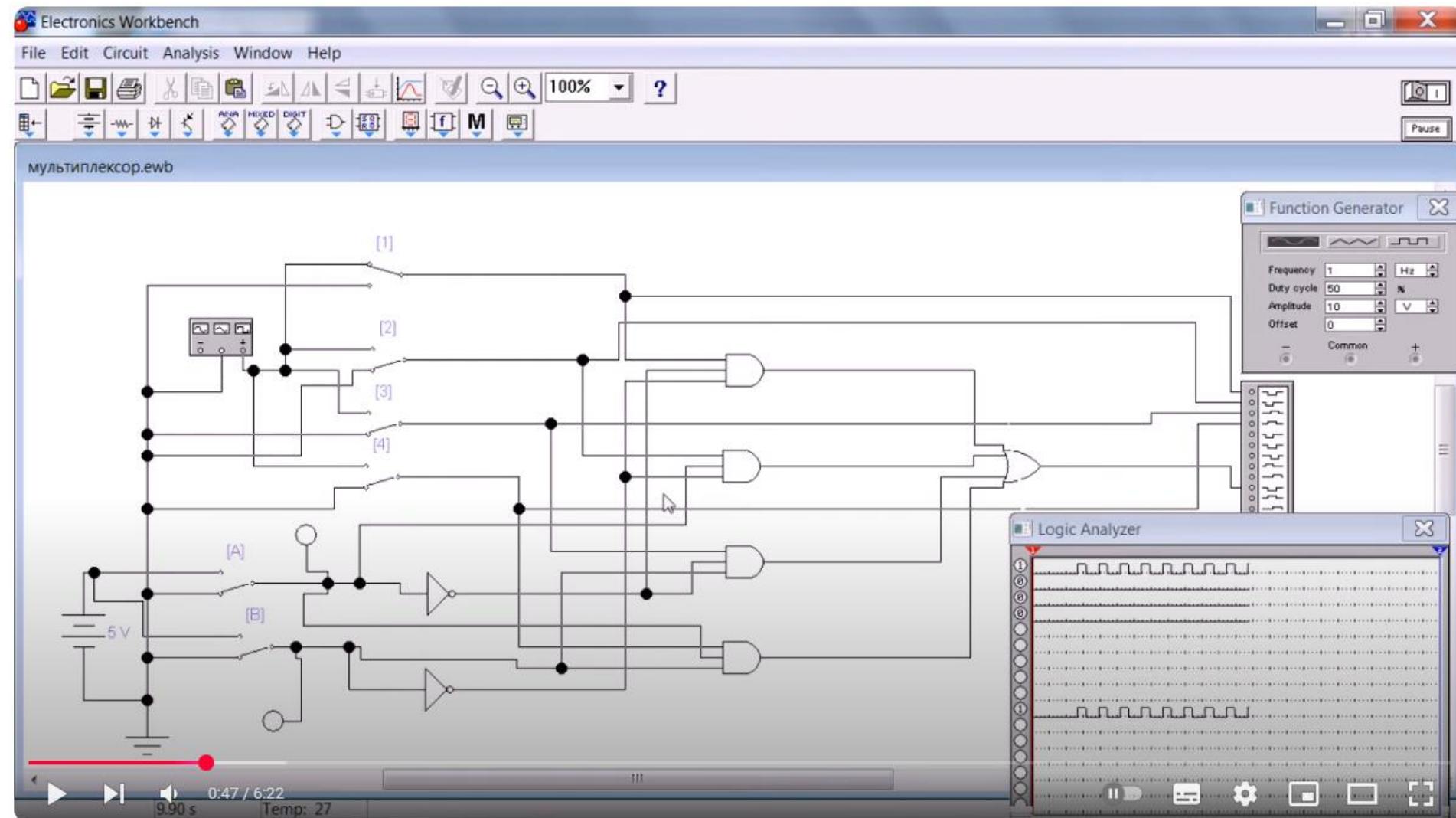
Заметки к слайду

Слайд 15 из 19 Тема Office 0:01 / 7:26 Русский (Россия)

Лабораторное занятие №5
Исследование функциональных схем
мультиплексоров и демультиплексоров.

- Цель работы: теоретически исследовать функциональные схемы мультиплексоров и демультиплексоров
- Оборудование: программа EWB 5.12 (Electronics Workbench), инструкционные карты.

Мультиплексоры и демультиплексоры часть 2 (2020)
https://www.youtube.com/watch?v=uaY9ij_yScQ



Мультиплексоры и демультиплексоры часть 3 (2020)
<https://www.youtube.com/watch?v=hEAUtVailvQ>

Вычислительные системы, сети и телекоммуникации. 2 семестр (ПИБ-18И1) (Бречка Д.М.)

Участники

Компетенции

Оценки

В начало

Личный кабинет

Календарь

Личные файлы

YouTube

Skype

Профессия интерн...

Web service API fun...

Другие закладки

Лабораторная работа 3. Мультиплексоры, демультиплексоры и компараторы кодов

Задание:

1. Исследовать мультиплексор с 4-мя информационными входами.
2. Исследовать демультиплексор с 4-мя выходами.
3. Исследовать компаратор четырехразрядных чисел.
4. Составить и исследовать схему мультиплексора с 4-мя информационными входами.
5. Составить и исследовать схему демультиплексора с 4-мя выходами.
6. Составить и исследовать схему компаратора четырехразрядных чисел.

Проект Logisim.

Теоретические сведения по моделированию простейших комбинационных устройств (пп. 2.2.3, 2.2.4, 2.2.5).

В качестве ответа на задание загрузите отчет о проделанной работе.

Специальные возможности

A- A A+ A
R A A A

ЗАПУСТИТЬ ATBAR

(всегда?)

Статистика

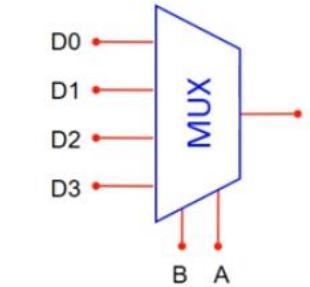
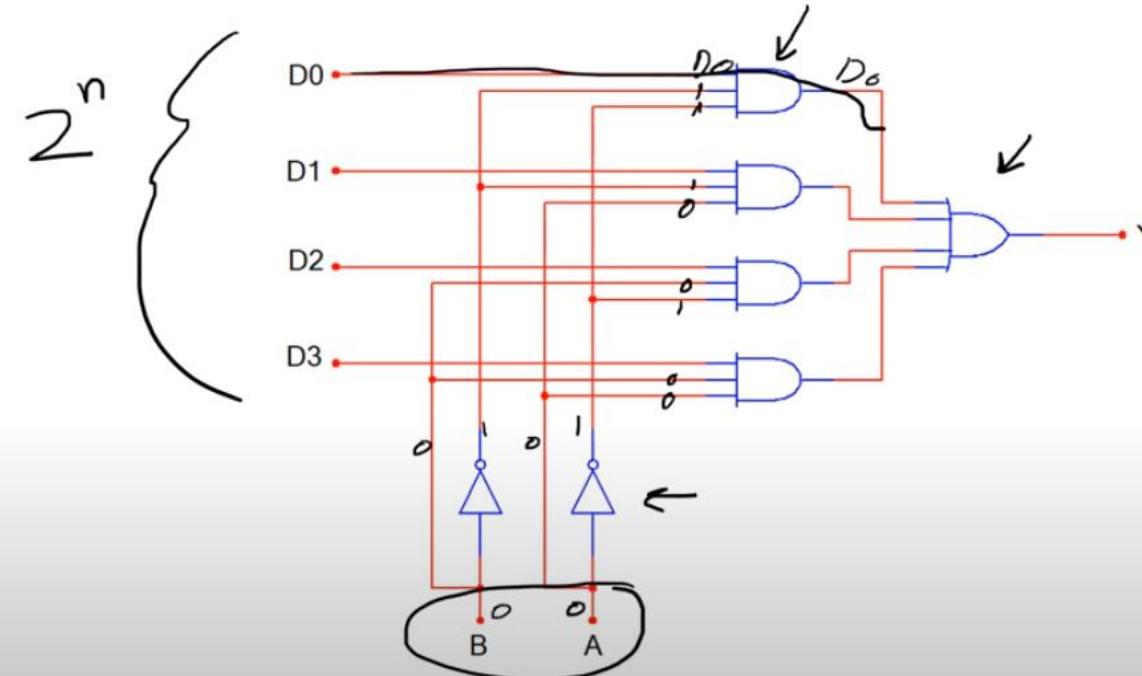
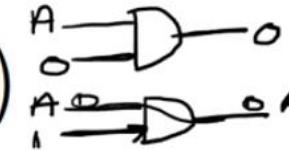
1

15
10
10:30

ENG недоступен 22.02.2021

Лабораторная работа № 3. Мультиплексоры, демультиплексоры и компараторы кодов (2021)
<https://www.youtube.com/watch?v=-iltmeKmqq0>

4-to-1 Multiplexer (MUX)



→

B	A	Y
0	0	D0
0	1	D1
1	0	D2
1	1	D3

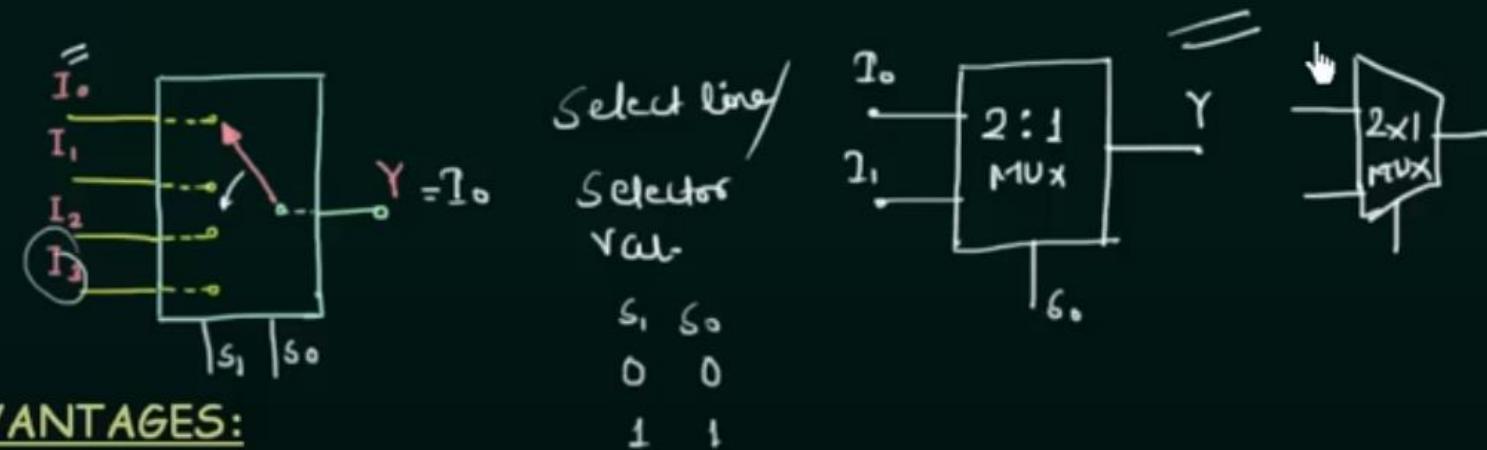


Multiplexers and DeMultiplexers (2017)
Мультиплексоры и демультиплексоры (2017)
<https://www.youtube.com/watch?v=HleQhZ9Gq5s>

Introduction to Multiplexers

MUX

- » It is combinational circuit that selects binary information from one of many input lines and directs it to o/p line.
- » It is simply a DATA SELECTOR



ADVANTAGES:

1. Reduces no. of wires.
2. Reduces circuit complexity & cost.
3. Implementation of various circuit using MUX.



4X1 Multiplexer

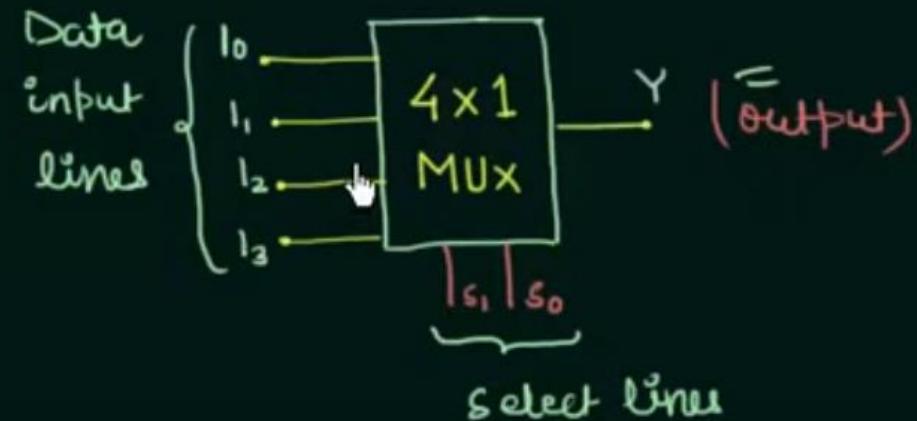
$$\underline{n} = 4$$

$$\underline{m} = \log_2 \underline{n}$$

$$m = \log_2 4$$

$$= \log_2 2^2 \quad \log_2 a = 2$$

$$m = 2 \log_2 2 \cdot 1 = 2 \cdot 1 = 2$$



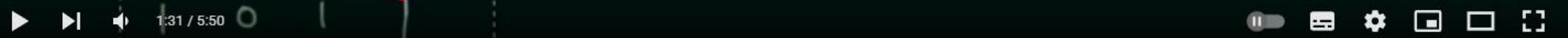
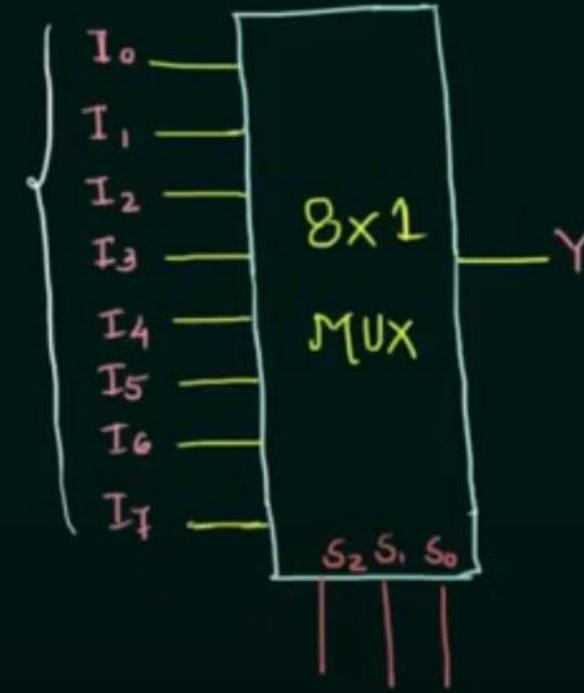
(8X1 Multiplexer)

$$n = 8 = 2^3$$

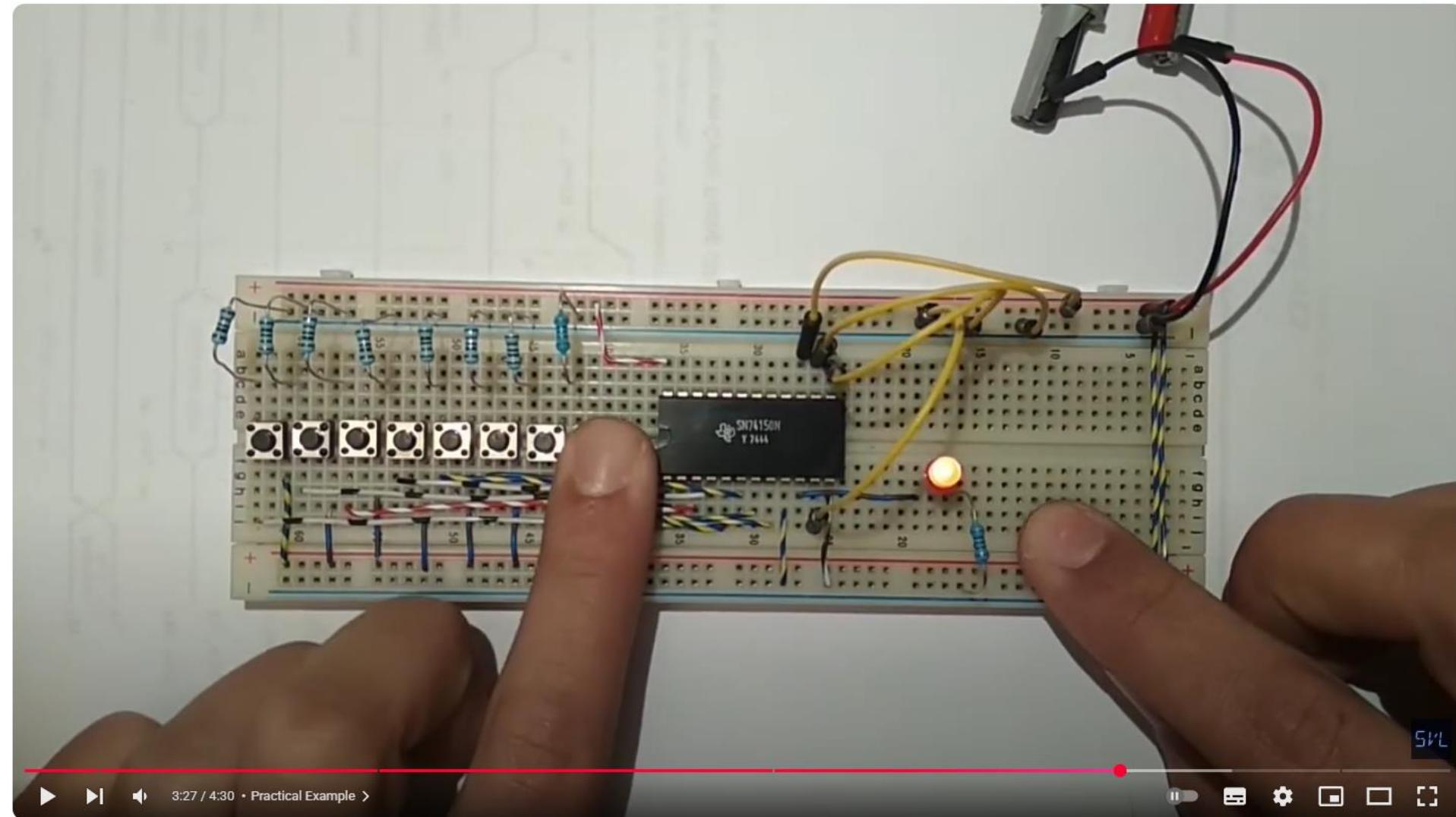
$$m = \log_2 n$$

$$m = \log_2 2^3 = 3$$

s_2	s_1	s_0	Y
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	



8X1 Multiplexer (2014)
Мультиплексор 8X1 (2014)
<https://www.youtube.com/watch?v=b0z7YKKCCyY>

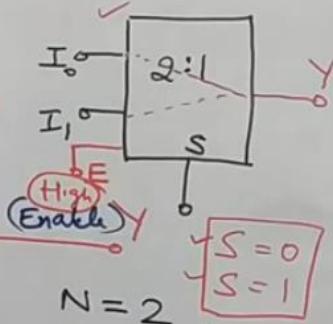
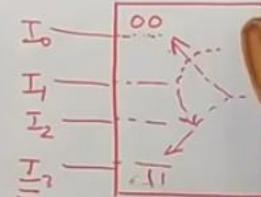
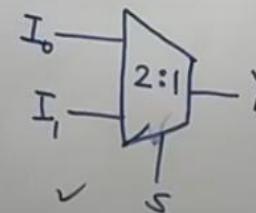


Multiplexer Tutorial | Basic Theory With Practical Example (2021)
Учебное пособие по мультиплексорам | Базовая теория с практическим примером (2021)
<https://www.youtube.com/watch?v=qUa29m-c-5o>

INTRODUCTION To MULTIPLEXERS

Multiplexer is a combinational circuit that select binary of information from one of many input lines and direct it to the output.

- It is simply a data selector
- It reduces the no. of wires
- Reduces ckt complexity & Cost
- Implementation of various cks using MUX
- Medium scale Integration (MSI) ckt. available in the form of Ics



$$\begin{aligned} N &= 2 \\ m &= \log_2 2 \\ m &= 1 \end{aligned}$$

Truth Table		
E	S	Y
0	X	0
1	0	I ₀
1	1	I ₁



1. Introduction to Multiplexers in Hindi | MUX Basics | 2 X 1 MUX Design (2019)

1. Введение в мультиплексоры на хинди | Основы MUX | Проектирование 2 X 1 MUX (2019)

<https://www.youtube.com/watch?v=2aZBMNj8fns>

4x1 MULTIPLEXER

The diagram illustrates a 4x1 Multiplexer (MUX) with the following components and connections:

- Inputs:** Four data inputs labeled I_0, I_1, I_2, I_3 are connected to the MUX.
- Select Lines:** Two select lines, S_1 (MSB) and S_0 (LSB), are connected to the MUX. A third select line, S_2 , is shown but not connected to the MUX.
- Enable Line:** An enable line labeled $E = \text{High}$ is connected to the MUX.
- Output:** The output Y is determined by the selected input based on the select lines.

Truth Table:

S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

Equation:

$$Y = \bar{S}_1 \bar{S}_0 I_0 + \bar{S}_1 S_0 I_1 + S_1 \bar{S}_0 I_2 + S_1 S_0 I_3$$

Annotations:

- $N = 4$
- $N = 2^m \rightarrow m = 2$
- $m = \log_2 N$
- $= \log_2 4 = \log_2 2^2$
- $= 2$
- $m = 2$
- S_1 (MSB)
- S_0 (LSB)
- Enable
- Multiplexer Tree

Speaker: A man wearing glasses and a black Adidas t-shirt is speaking and gesturing towards the whiteboard.

2. 4x1 Multiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
 2. Мультиплексор 4x1 на хинди | Tech Gurukul от Динеша Ары (2019)
<https://www.youtube.com/watch?v=06VyGspXpMo>

8 X 1 MULTIPLEXER

Input Databus
Select Line

$$Y = \bar{S}_2 \bar{S}_1 \bar{S}_0 I_0 + \bar{S}_2 \bar{S}_1 S_0 I_1 + \bar{S}_2 S_1 \bar{S}_0 I_2 + \bar{S}_2 S_1 S_0 I_3 + S_2 \bar{S}_1 \bar{S}_0 I_4 + S_2 \bar{S}_1 S_0 I_5 + S_2 S_1 \bar{S}_0 I_6 + S_2 S_1 S_0 I_7$$

S ₂	S ₁	S ₀	Y
0	0	0	I ₀
0	0	1	I ₁
0	1	0	I ₂
0	1	1	I ₃
1	0	0	I ₄
1	0	1	I ₅
1	1	0	I ₆
1	1	1	I ₇

adidas THE BRAND WITH THE THREE STRIPES

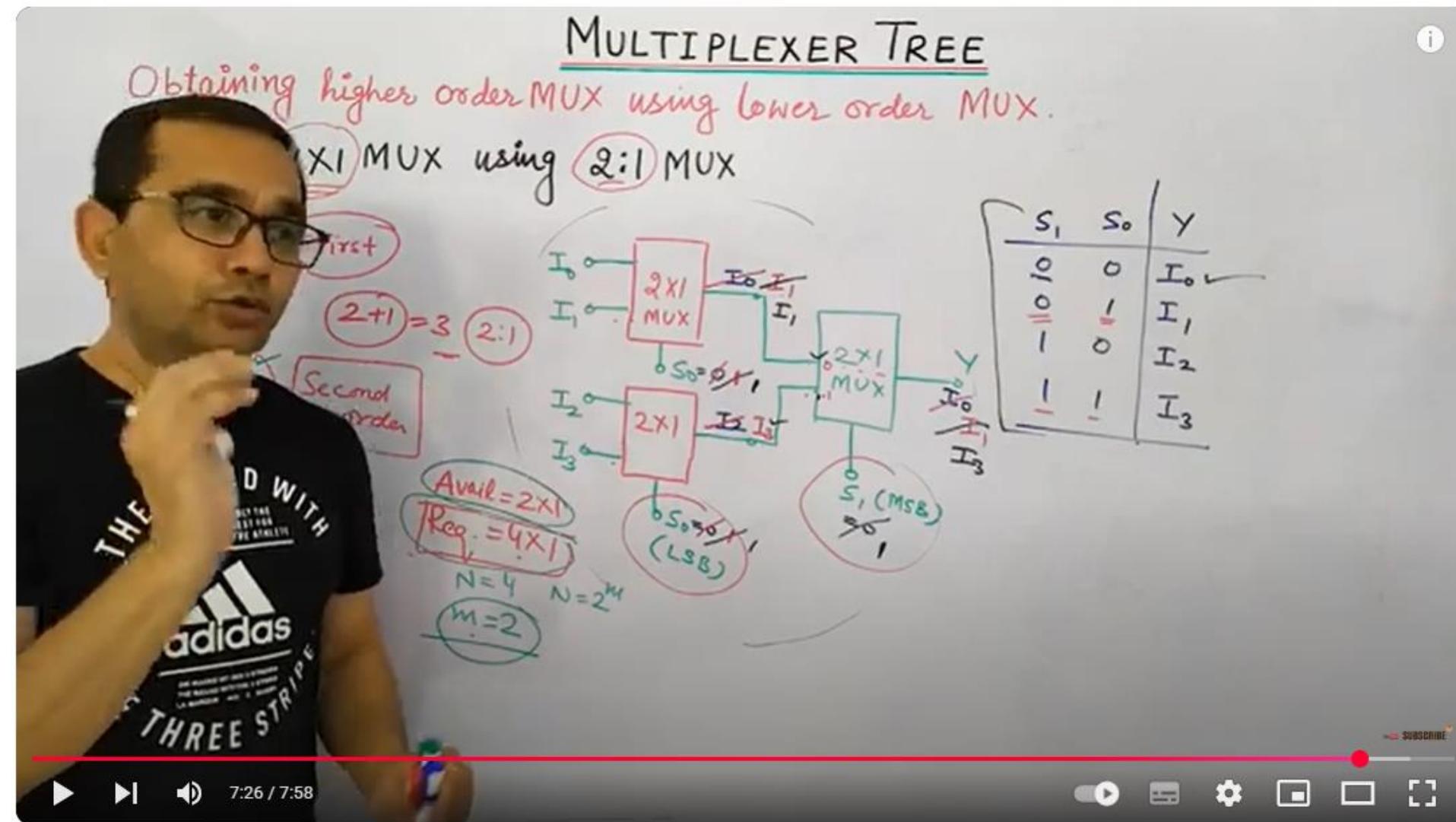
BRAND WITH 03 ONLY THE BEST FROM THE ATHLETIC

THE THREE STRIPES

SUBSCRIBE

10:43 / 11:21

3. 8X1 Multiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
 3. Мультиплексор 8X1 на хинди | Tech Gurukul от Динеша Ари (2019)
https://www.youtube.com/watch?v=IZRR-_2TmxU



4. 4X1 MUX using 2X1 MUX in Hindi | Multiplexer Tree | Tech Gurukul by Dinesh Arya(2019)
 4. Мультиплексор 4X1 с использованием мультиплексора 2X1 на хинди | Дерево мультиплексоров | Tech Gurukul от Динеша Арыи(2019)
<https://www.youtube.com/watch?v=78UHqvhDaSc>

MULTIPLEXER TREE

Obtaining higher order MUX using lower order MUX.

Implement 8x1 MUX using 2x1 MUX

Required IPs $n_1 = 8$

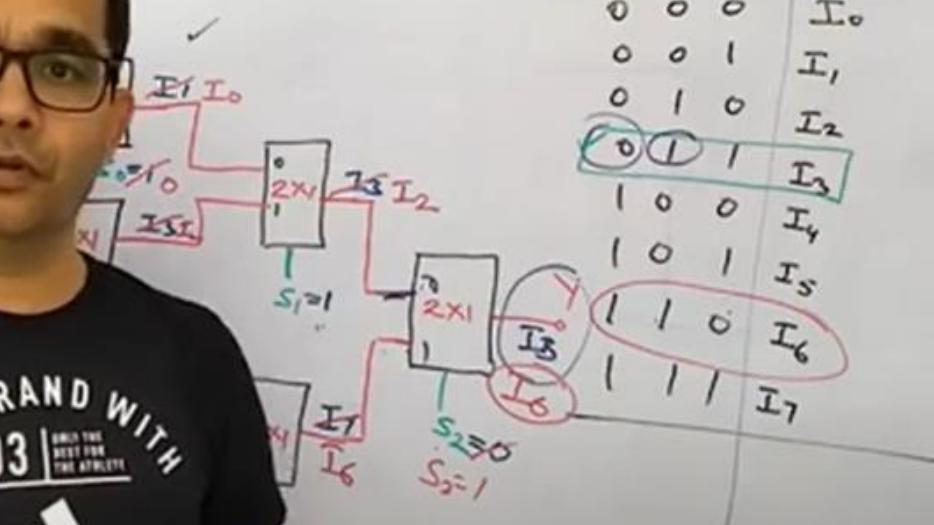
Available IP $n_2 = 2$

$$\frac{n_1}{n_2} = \frac{8}{2} = 4 \rightarrow \text{I}^{\text{st}} \text{ Order}$$

$$\frac{4}{2} = 2 \rightarrow \text{I}^{\text{nd}} \text{ Order}$$

$$\frac{2}{2} = 1 \rightarrow \text{I}^{\text{rd}} \text{ Order}$$

Enable



5. 8x1 MUX using 2x1 MUX in Hindi | Multiplexer Tree | Tech Gurukul by Dinesh Arya (2019)
 5. Мультиплексор 8x1 с использованием мультиплексора 2x1 на хинди | Дерево мультиплексоров | Tech Gurukul от Динеша Арии (2019)
<https://www.youtube.com/watch?v=QaL75vZt8-Q>

MULTIPLEXER TREE

Obtaining higher order MUX using lower order MUX.

Implement 8x1 MUX using 4x1 MUX (Specified Multiplexer Tree)

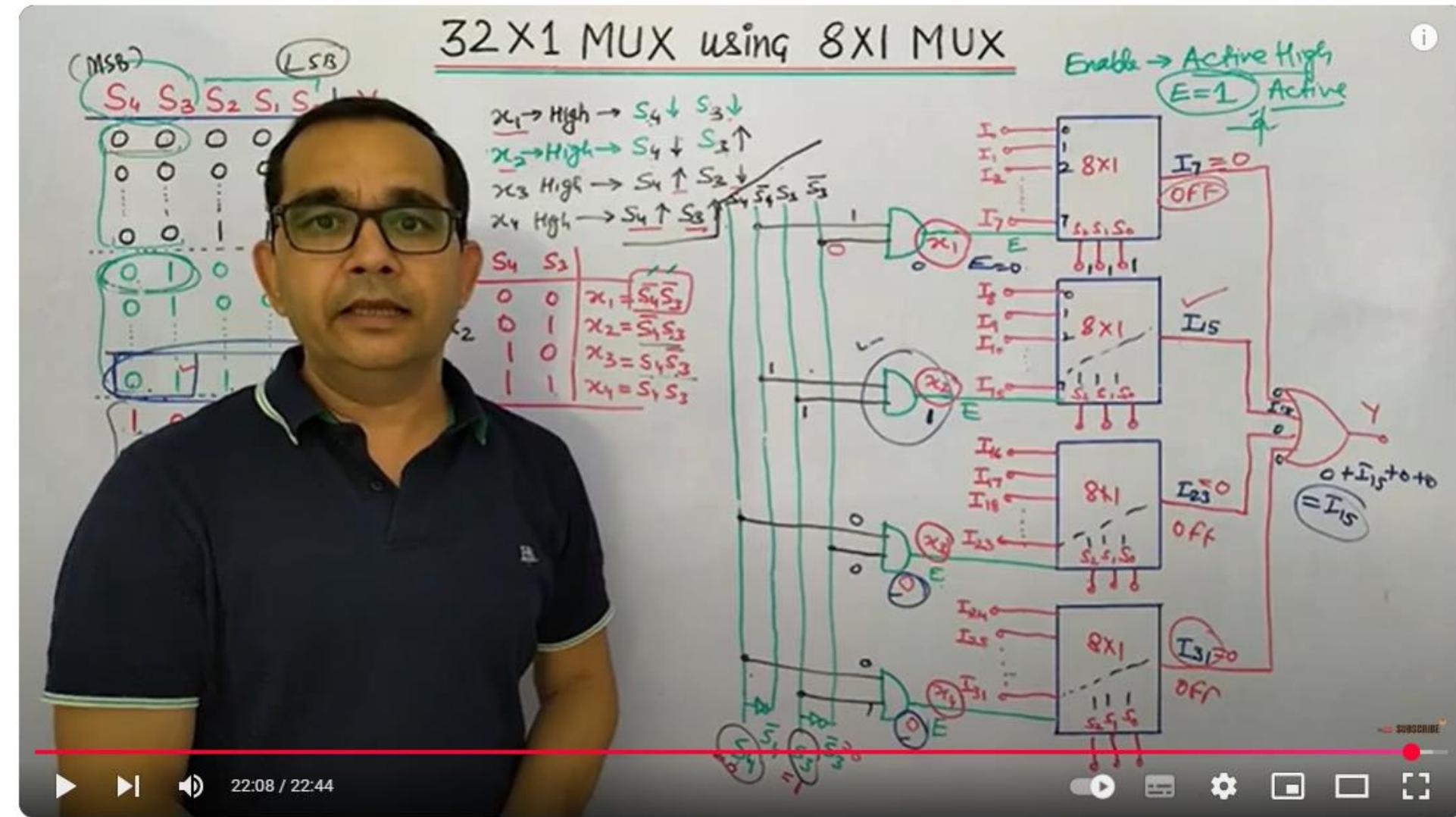
Req MUX $n_1 = 8$
Avail MUX $n_2 = 4$

$$\frac{n_1}{n_2} = \frac{8}{4} = 2$$

$\frac{2}{4} = 0.5$ X

The video player interface includes a play button, volume control, timestamp (8:50 / 9:26), and other standard video controls.

6. 8x1 MUX using 4x1 MUX in Hindi | Multiplexer Tree | Tech Gurukul by Dinesh Arya (2019)
 6. Мультиплексор 8x1 с использованием мультиплексора 4x1 на хинди | Дерево мультиплексоров | Tech Gurukul от Динеша Арии (2019)
https://www.youtube.com/watch?v=x4h7OAJq_N4



7. 32X1 MUX using 8X1 MUX in Hindi | Very Important | Tech Gurukul By Dinesh Arya (2019)
 7. МУЛЬТИПЛЕКСОР 32X1 с использованием мультиплексора 8X1 на хинди | Очень важно | Tech Gurukul от Динеша Арыи (2019)
https://www.youtube.com/watch?v=iRR8_fISIw

$F(A, B, C) = \sum(0, 2, 3, 4, 6)$ using 4×1 MUX

Implement given logic function.

Truth Table:

A	B	C	$\bar{B} \bar{C}$	$\bar{B} C$	$B \bar{C}$	BC	F
0	0	0	1	0	0	1	0
0	0	1	1	1	0	1	0
0	1	0	0	1	1	0	1
0	1	1	0	0	0	1	1
1	0	0	1	0	0	1	1
1	0	1	1	1	0	1	1
1	1	0	0	1	1	0	1
1	1	1	0	0	0	1	0

Karnaugh Map:

$\bar{A} \bar{B}$	$\bar{A} B$	$A \bar{B}$	AB
01	10	11	11
I_1	I_2	I_3	I_0

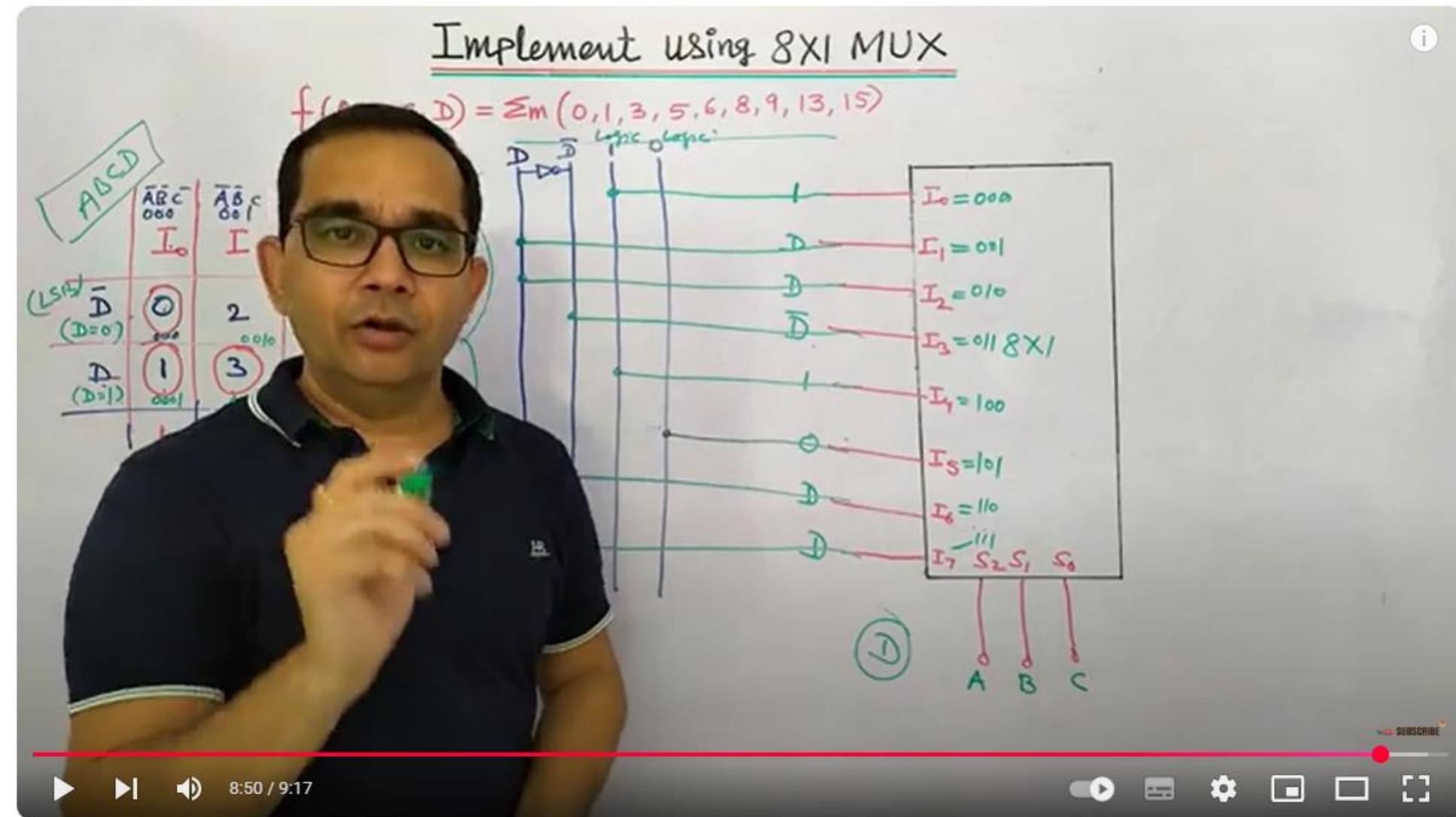
MUX Implementation:

Inputs: S_1, S_0 (select lines), A, B, C (data inputs).

Outputs: $I_0 = \bar{A} \bar{B} \bar{C}$, $I_1 = \bar{A} B \bar{C}$, $I_2 = A \bar{B} C$, $I_3 = A B C$.

Y = $\bar{B} \bar{C}$ (selected by $S_1, S_0 = 01$)

8. Implementation of Boolean Function using Multiplexers in Hindi - 1 | Very Easy (2019)
 8. Реализация логической функции с использованием мультиплексоров на хинди - 1 | Очень просто (2019)
<https://www.youtube.com/watch?v=tStPccdLLk>



9. Implementation of Boolean Function using Multiplexers in Hindi - 2 | Very Easy (2019)
 9. Реализация логической функции с использованием мультиплексоров на хинди - 2 | Очень просто (2019)
https://www.youtube.com/watch?v=_V_Md8-dSj8

INTRODUCTION TO DEMULTIPLEXERS

- One I/P and many O/P
 - Reverse operation of MUX
 - Also called as Data distributor
 - Demux is a combination of MUX and AND gate

$$n = 2^m$$

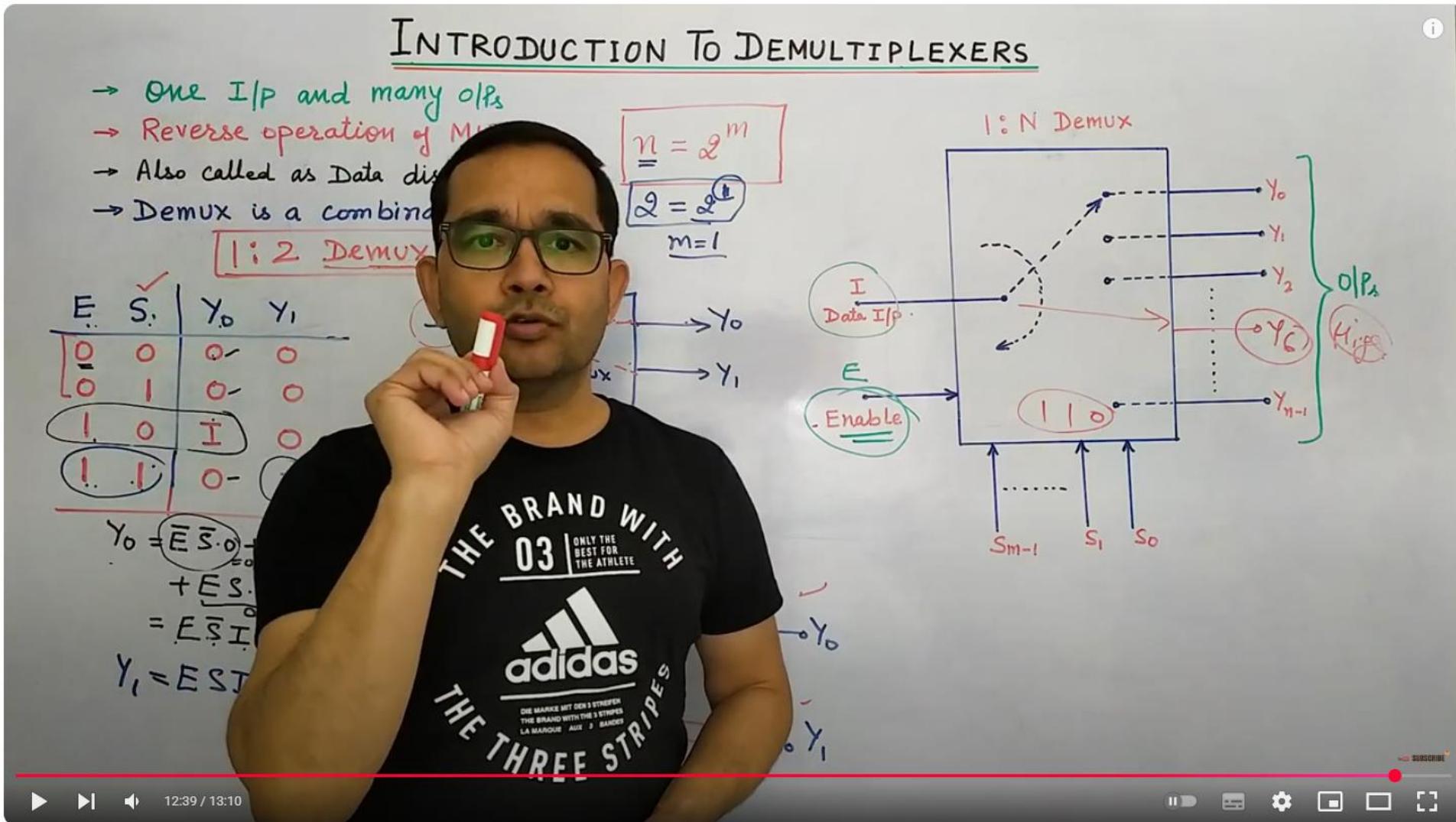
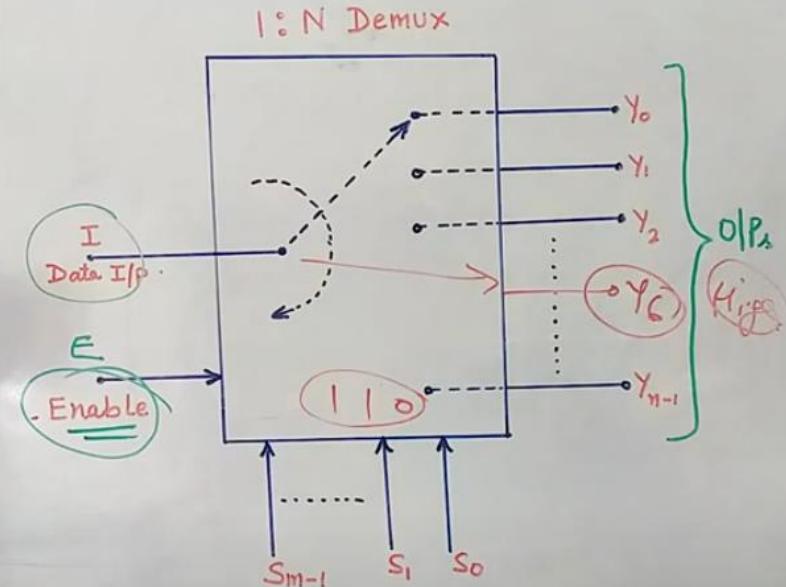
$$2 = \underline{2}^{\textcircled{1}}$$

11 - 2

E	S.	y_0	y_1
0	0	0-	0
0	1	0-	0
1	0	I	0
1	1	0-	0

$$Y_0 = \bar{E} \bar{S} \cdot 0 -$$

$$= E\bar{S}I$$



11. Introduction to Demultiplexers in Hindi | Tech Gurukul by Dinesh Arya (2019)
11. Введение в демультиплексоры на хинди | Tech Gurukul Динеша Арии (2019)
<https://www.youtube.com/watch?v=FarRWppOF-E>

1 : 4 DEMULTIPLEXER

→ One I/P and many O/Ps
 → Reverse operation of MUX
 → Also called as Data distributor
 → Demux is a combination of AND gate and OR gate

$4 = 2^2 \rightarrow m=2$
 $n = n.o.f. O/Ps$
 $m = \text{Select lines}$

Truth Table:

E	S ₁	S ₀	Y ₀	Y ₁	Y ₂	Y ₃
0	X	X	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

Logic Diagram:

Subtitles:

$Y_0 = E \bar{S}_1 \bar{S}_0$
 $Y_1 = E \bar{S}_1 S_0$
 $Y_2 = E S_1 \bar{S}_0$
 $Y_3 = E S_1 S_0$

YouTube Player Controls:

▶ ▶ ⏴ 8:58 / 9:28 🔍 SUBSCRIBE

12. 1:4 Demultiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
 12. Демультиплексор в формате 1:4 на хинди | Tech Gurukul от Динеша Ары (2019)
<https://www.youtube.com/watch?v=y-JOsmHVcKA>

Белорусско-Российский университет, Кафедра «Программное обеспечение информационных технологий»

1 : 8 DEMULTIPLEXER

$m = 2^3$
 $8 = 2^3$
 $m = \log_2 8$
 $= 3 \log_2 2$
 $= 3$

Block Diagram:

```

    graph LR
        A[Data I/p] --> B[1 : 8]
        E[Enable] --> B
        S2[S2 (MSB)] --> B
        S1[S1] --> B
        S0[S0 (LSB)] --> B
        B --> Y0["Y0"]
        B --> Y1["Y1"]
        B --> Y2["Y2"]
        B --> Y3["Y3"]
        B --> Y4["Y4"]
        B --> Y5["Y5"]
        B --> Y6["Y6"]
        B --> Y7["Y7"]
    
```

Truth Table:

E	S ₂	S ₁	S ₀	Y ₀	Y ₁	Y ₂	Y ₃	Y ₄	Y ₅	Y ₆	Y ₇
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0

Equations:

$$\begin{aligned} Y_0 &= E \bar{S}_2 \bar{S}_1 \bar{S}_0 \bar{I} \\ Y_1 &= E \bar{S}_2 \bar{S}_1 S_0 \bar{I} \\ Y_2 &= E \bar{S}_2 S_1 \bar{S}_0 \bar{I} \\ Y_3 &= E \bar{S}_2 S_1 S_0 \bar{I} \\ Y_4 &= E S_2 \bar{S}_1 \bar{S}_0 \bar{I} \\ Y_5 &= E S_2 \bar{S}_1 S_0 \bar{I} \\ Y_6 &= E S_2 S_1 \bar{S}_0 \bar{I} \\ Y_7 &= E S_2 S_1 S_0 \bar{I} \end{aligned}$$

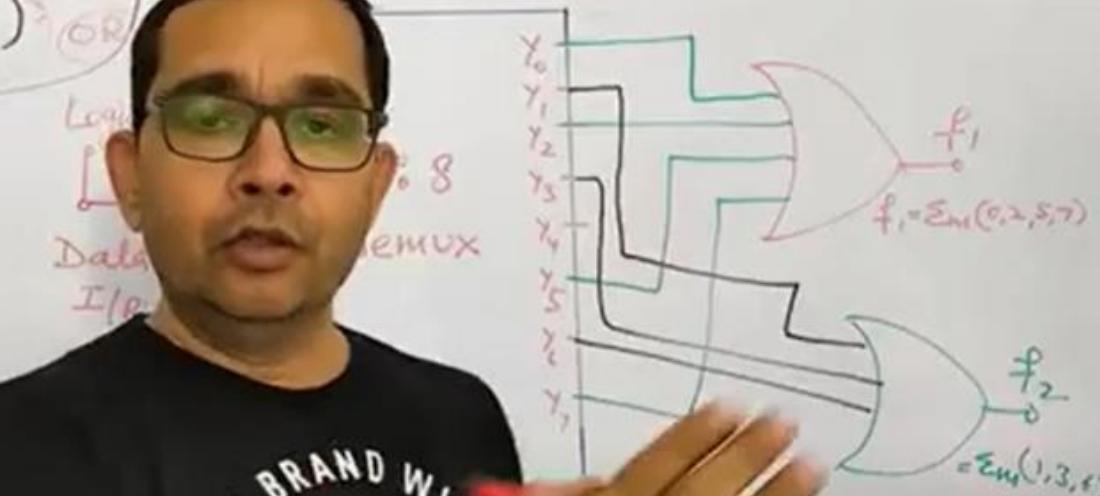
13. 1:8 Demultiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
 13. Демультиплексор в формате 1:8 на хинди | Tech Gurukul от Динеша Ары (2019)
<https://www.youtube.com/watch?v=epBoqrhAPq8>

Q: Implement the logic expression using demultiplexer.

$$\times f_1(A, B, C) = \sum_m (0, 2, 5, 7)$$

$$f_2(A, B, C) = \sum_m (1, 3, 6)$$

A	B	C	f_1	f_2
0	0	0	1	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	0
1	0	1	0	0
1	1	0	0	0
1	1	1	0	0

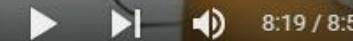
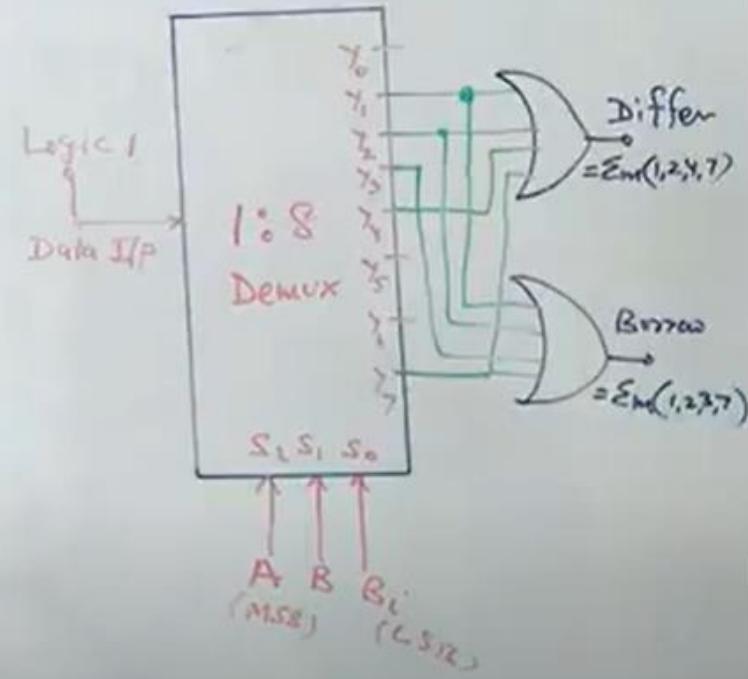
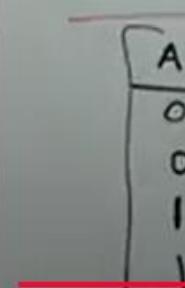


14. Implement Logic Expression or Truth Table using Demultiplexer in Hindi | Very Important (2019)
 14. Реализуйте логическое выражение или таблицу истинности с помощью демультиплексора на хинди | Очень важно (2019)
<https://www.youtube.com/watch?v=Sum3ZTULE3U>

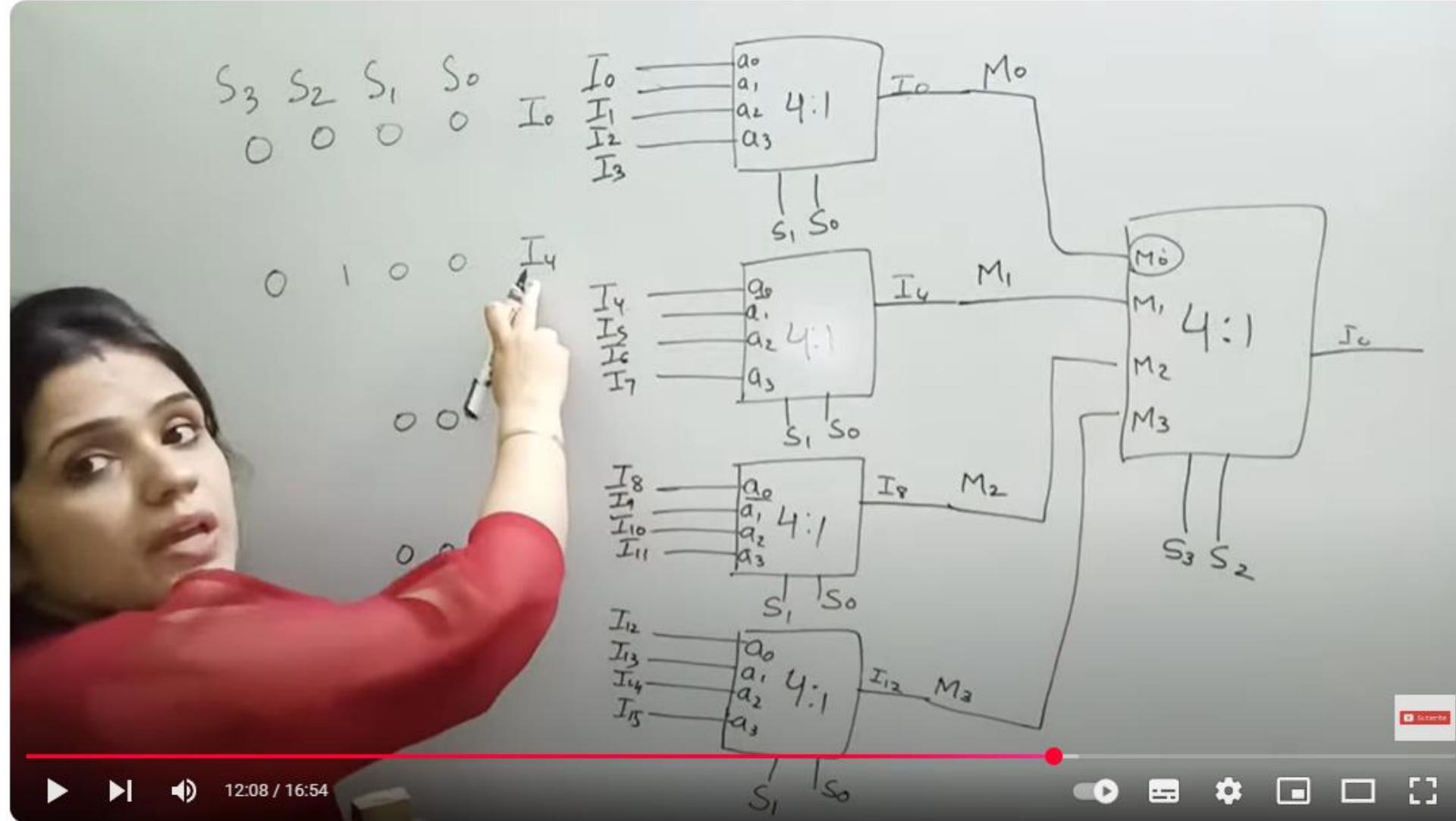
Full Subtractor using 1:8 Demux

Truth table for Full Subtractor

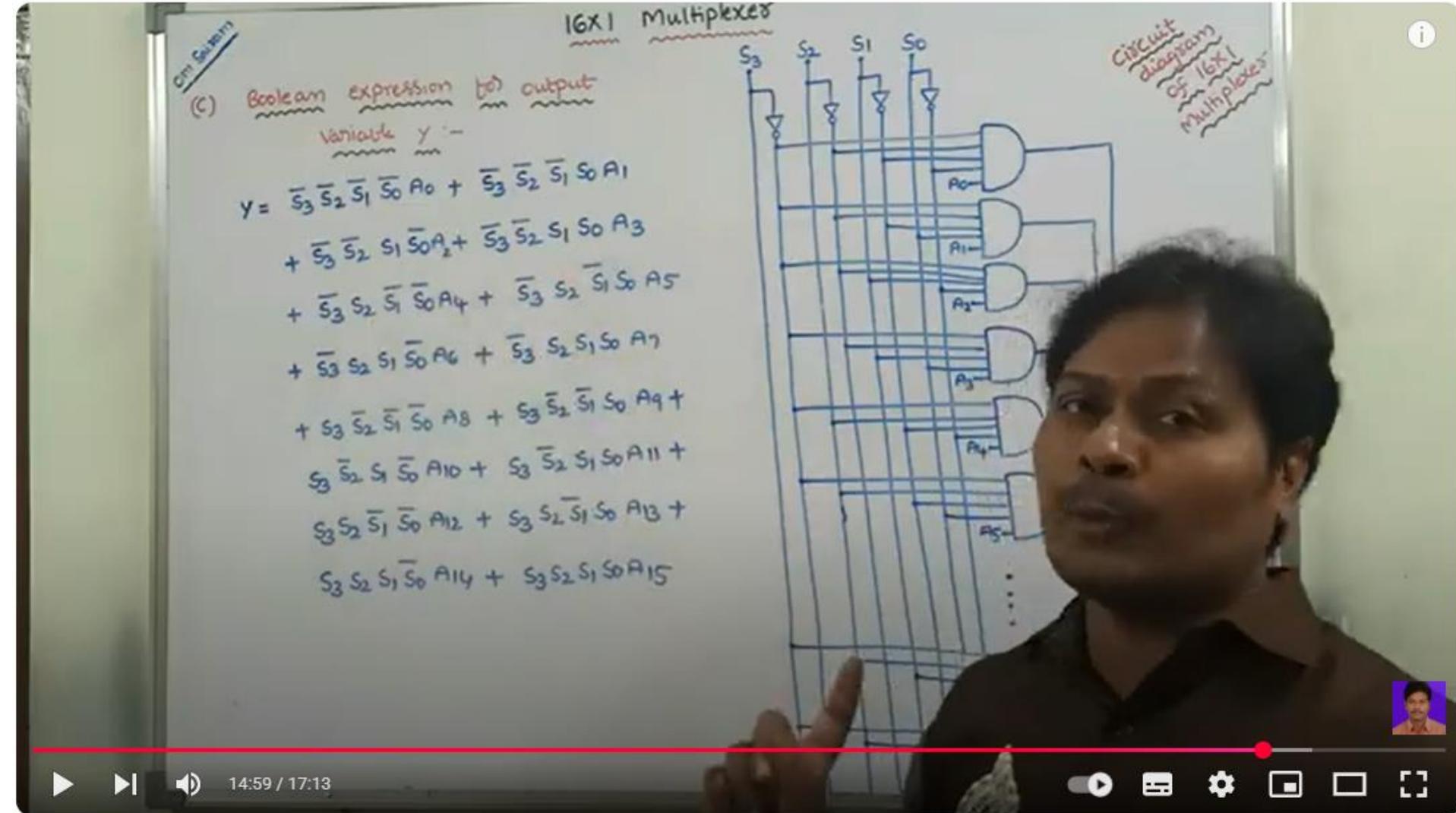
	A	B	B_i	Diff	Borrow
m_0	0 - 0	0	0	0	0
m_1	0 - 0	- 1	1	1	(1, 2, 3, 7)
m_2	0 - 1	1	1	1	(1, 2, 4, 7)
m_3	0 - 1	- 1	0	0	
m_4	1 - 0	1	1	0	
m_5	1 - 0	- 1	0	1	
m_6	1 - 1	0	0	0	
m_7	1 - 1	- 0	0	1	



15. Full Subtractor using 1:8 Demultiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
 15. Полное вычитание с использованием демультиплексора 1:8 на хинди | Tech Gurukul от Динеша Арии (2019)
<https://www.youtube.com/watch?v=hDtV9MkL7IU>



16 to 1 mux using 4 to 1 mux | 16 X 1 MULTIPLEXER USING 4 TO 1 MULTIPLEXER (2021)
 16 к 1 мультиплексору с использованием 4 к 1 мультиплексора | 16 X 1 МУЛЬТИПЛЕКСОР С ИСПОЛЬЗОВАНИЕМ 4 К 1 МУЛЬТИПЛЕКСОРА (2021)
<https://www.youtube.com/watch?v=3IUHu4AYqWk>



16X1 Multiplexer || 16 to 1 Multiplexer || Truth Table of 16X1 Multiplexer | Circuit of 16X1 MUX (2022)
 Мультиплексор 16X1 || Таблица истинности мультиплексора 16X1 | Схема мультиплексора 16X1 (2022)
<https://www.youtube.com/watch?v=9BnLejOS4YA>

Implement 4x1 Multiplexer using 2x1 Multiplexers

On Selection

Required Multiplexer : 4x1

Available Multiplexer : 2x1

Number of input lines in multiplexer (n_1) : 4

Number of selection lines in multiplexer (n_2) : 2

Multiplexers required at first level = $\frac{n_1}{n_2} = \frac{4}{2} = 2$

Multiplexers required at second level = $\frac{2}{2} = 1$

4x1 MUX = $2 \times 1 \times 2 = 4$

Truth Table :-

Inputs	Outputs
$S_1\ S_0$	Y
0 0	I_0
0 1	I_1
1 0	I_2
1 1	I_3

Circuit diagram :-

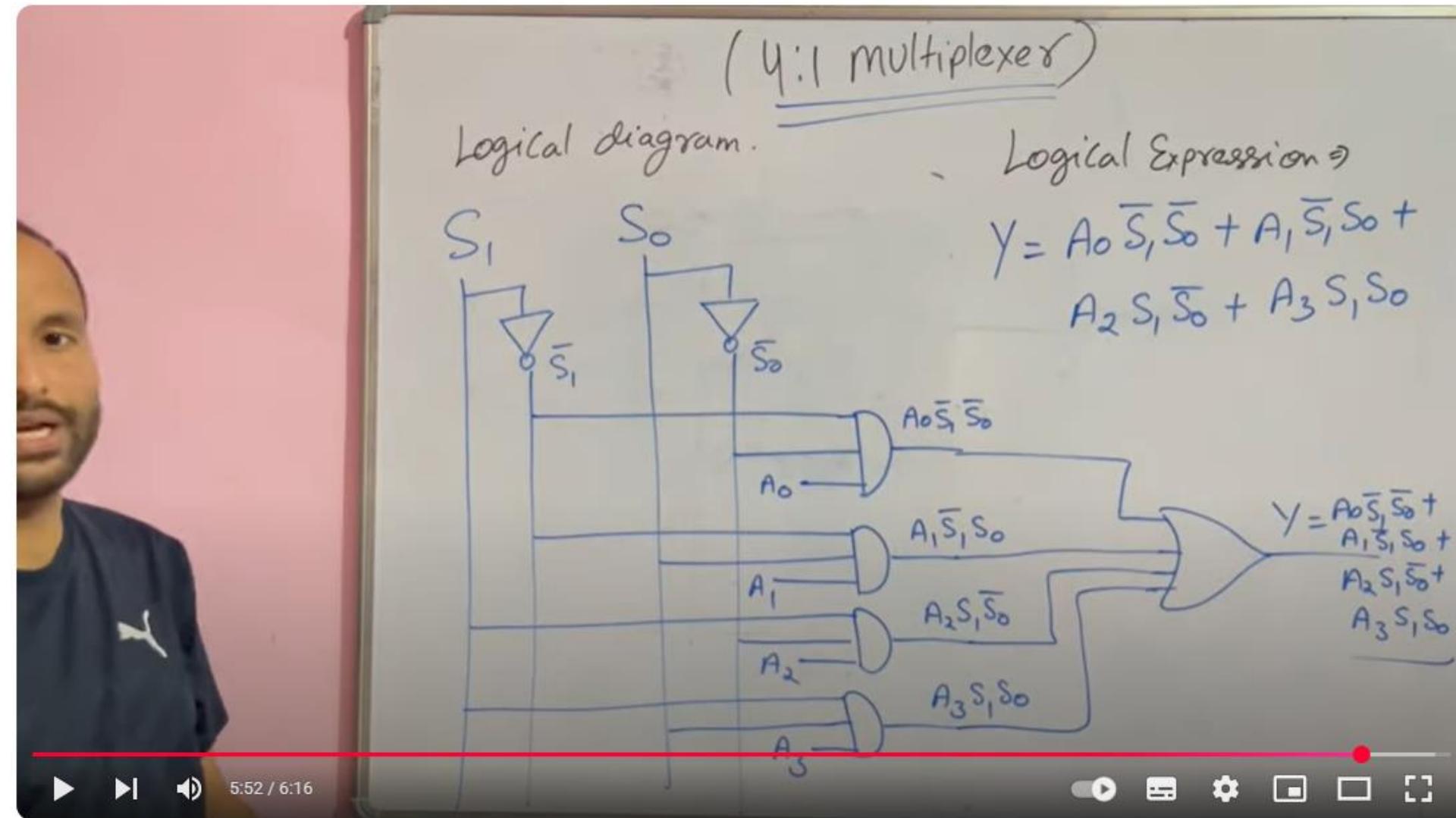
4x1
2x1
 $\therefore n=2$
Total number of Selection lines = 2

2x1
2x1
2x1
 $n=1$

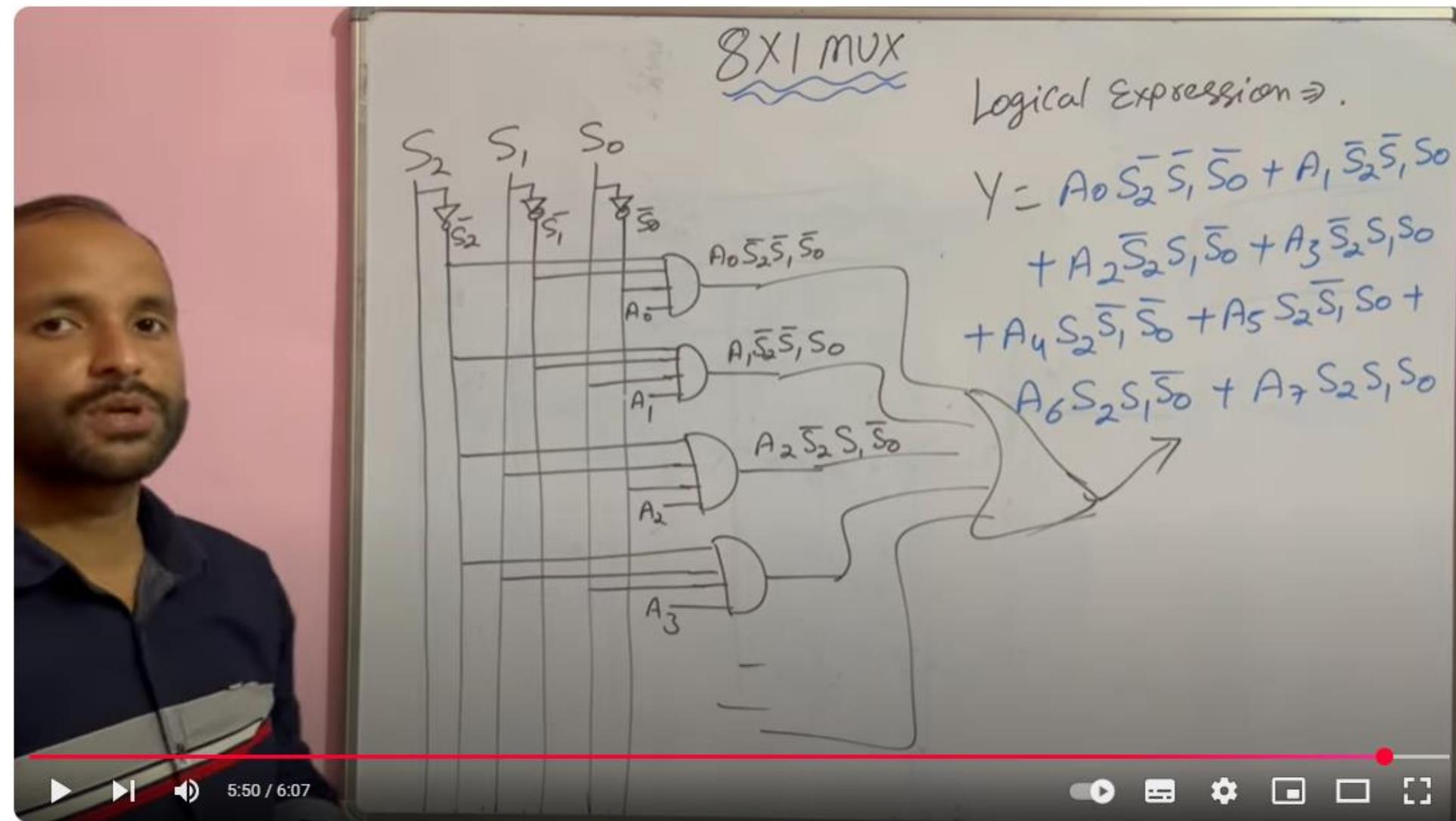
16:42 / 19:21

4X1 MUX using 2X1 MUX || 4*1 Multiplexer using 2*1 Multiplexer | 4x1 mux using 2x1 mux | DLD || STLD (2022)
 4X1 мультиплексор с использованием 2X1 мультиплексора | DLD || STLD (2022)

<https://www.youtube.com/watch?v=OYpdHCjPlcE>



4 to 1 Multiplexer (MUX) Block diagram, Truth Table, Logical expression and Diagram (2023)
 Структурная схема мультиплексора (MUX) 4 к 1, таблица истинности, логическое выражение и диаграмма (2023)
<https://www.youtube.com/watch?v=u7LVduhiYI8>

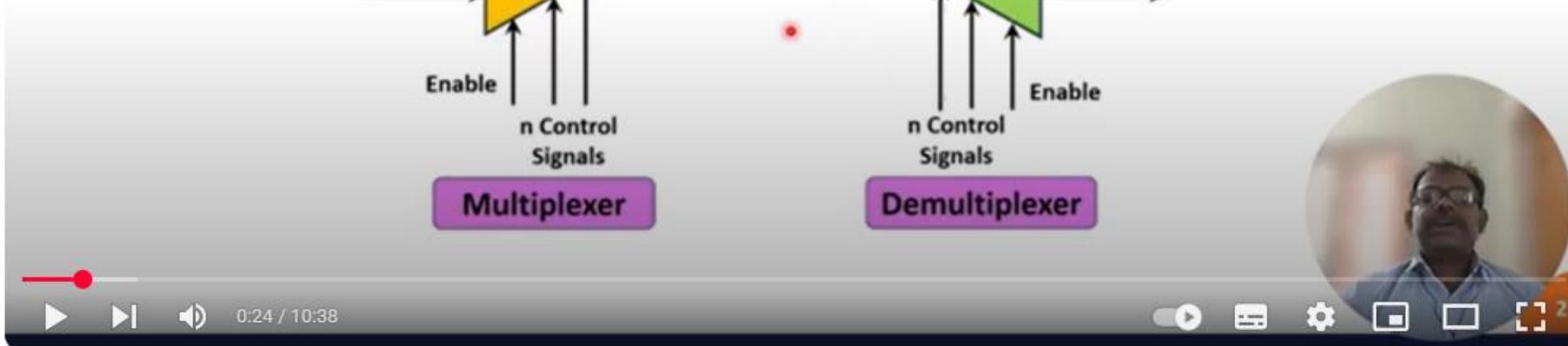
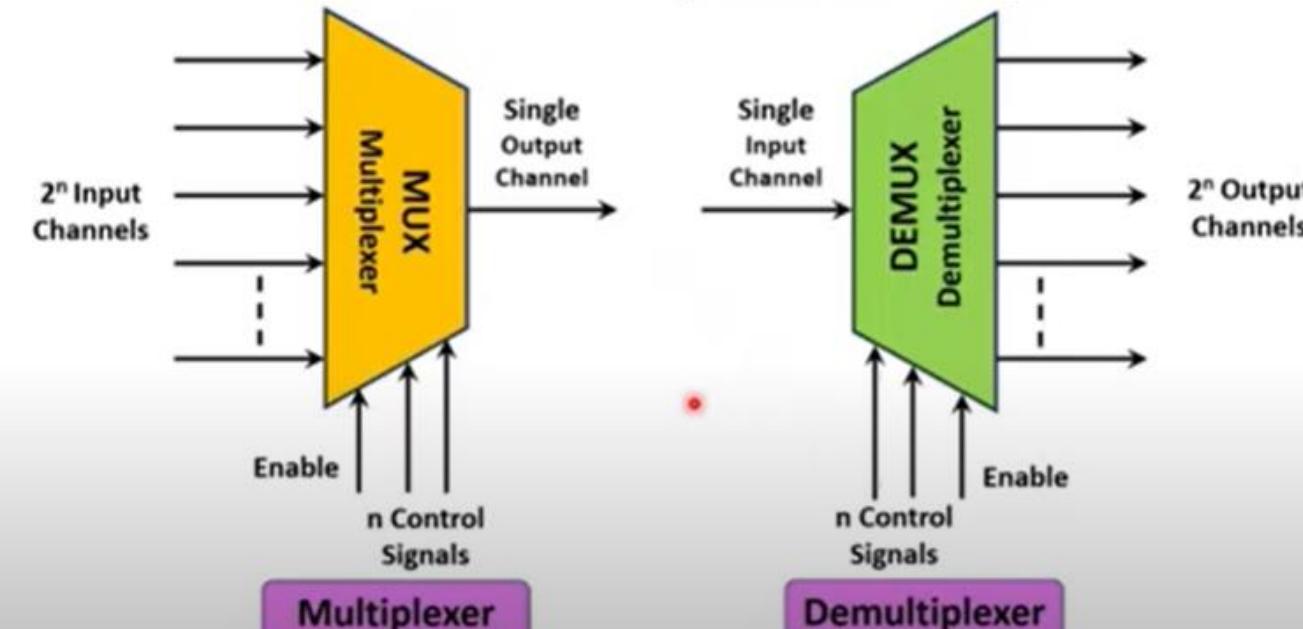


8 to 1 Multiplexer (MUX) Block diagram, Truth Table, Logical expression and Diagram (2023)
 Структурная схема мультиплексора (MUX) от 8 до 1, таблица истинности, логическое выражение и диаграмма (2023)
<https://www.youtube.com/watch?v=0-BsImNhPVs>

Multiplexer



Difference Between Multiplexer and Demultiplexer



MUX/DEMUX (2025)

<https://www.youtube.com/watch?v=wTTDfsMVxsQ>

Q. 1] $F(A, B, C, D) = \sum (1, 3, 4, 11, 12, 13, 14, 15)$ using 16×1 mux

A	B	C	D	F
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	0
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

16×1 mux
 $\therefore 2^4 = 16$
4 select
Here,
No. of select
lines =
No. of variables

Implementation of boolean function using multiplexers | Hindi | One question with three types of mux (2020)
Реализация логической функции с использованием мультиплексоров | Хинди | Один вопрос с тремя типами мультиплексоров (2020)
<https://www.youtube.com/watch?v=gTFx9FNIA2w>

You're sharing this screen.

Texas Instruments Multiplexers & Signal Switches

Making Connections. Switching Things Around.

January 2020

0:22 / 19:32 • Introduction >

TEXAS INSTRUMENTS

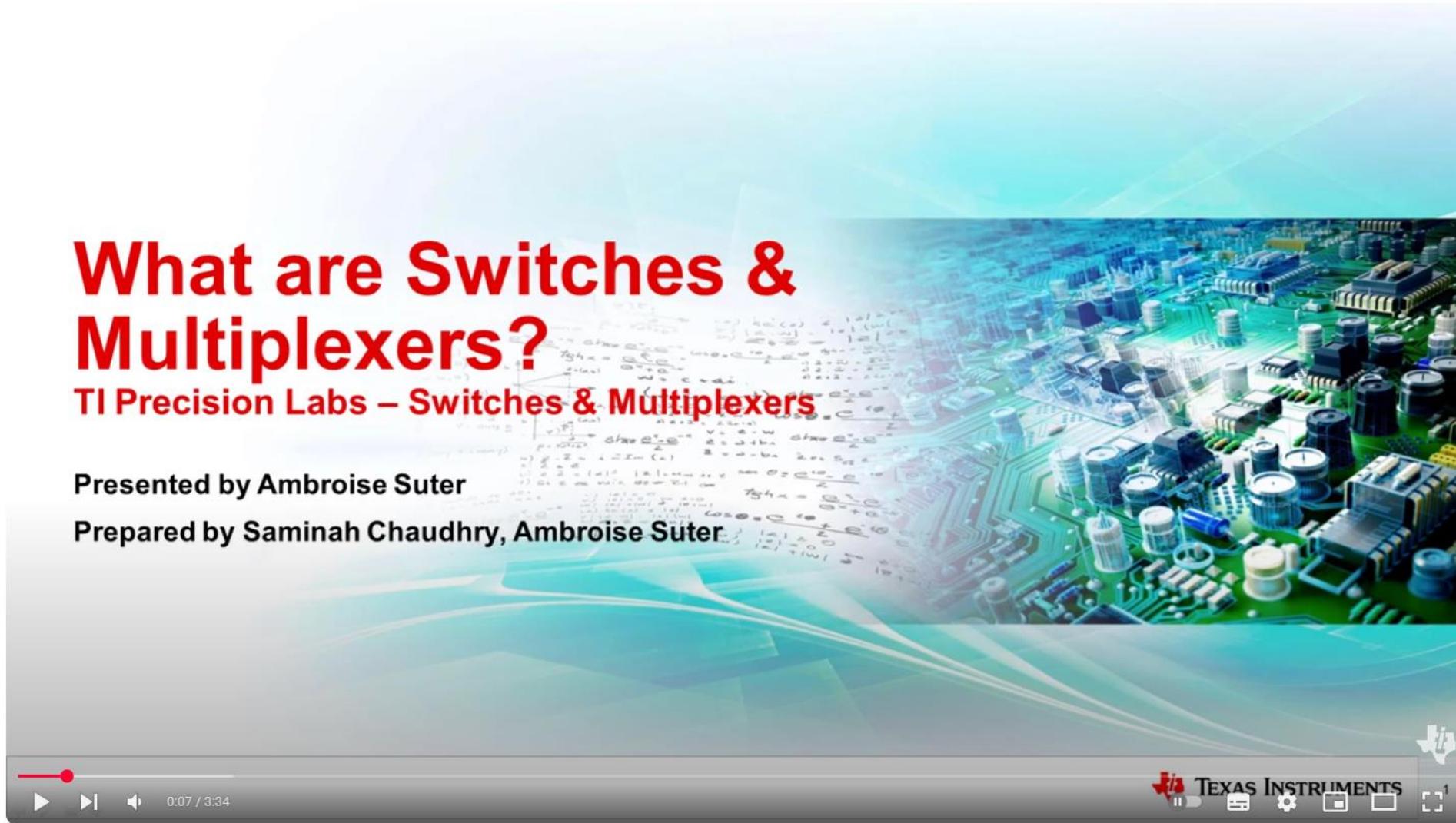
Multiplexers & Signal Switches (2020)
Мультиплексоры и коммутаторы сигналов (2020)
<https://www.youtube.com/watch?v=I2K8oQyS5z8>

What are Switches & Multiplexers?

TI Precision Labs – Switches & Multiplexers

Presented by Ambroise Suter

Prepared by Saminah Chaudhry, Ambroise Suter



What are switches & multiplexers? (2019)
Что такое коммутаторы и мультиплексоры? (2019)
<https://www.youtube.com/watch?v=n9EklrCrgH8>

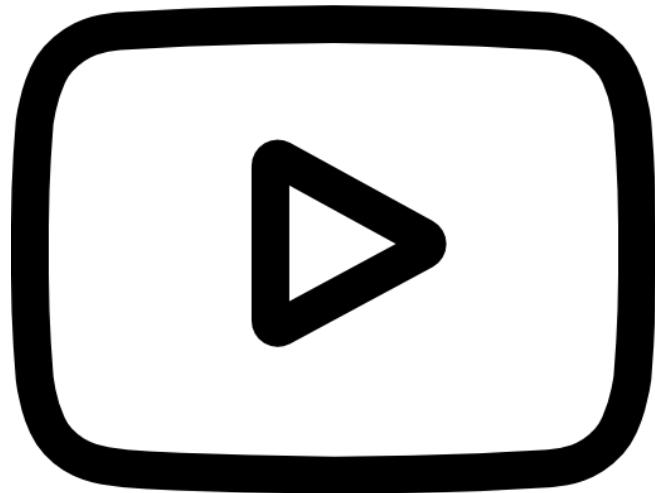
Are Switches & Multiplexers Bidirectional?

TI Precision Labs – Switches & Multiplexers

Presented and Prepared by Kate Dickson



Are switches & multiplexers bidirectional? (2019)
Являются ли коммутаторы и мультиплексоры двунаправленными? (2019)
<https://www.youtube.com/watch?v=z9GRiYPq7LM>



**Шифраторы,
Дешифраторы**





Шифраторы, дешифраторы. Назначение, принцип работы, типовые схемы (2021)
<https://www.youtube.com/watch?v=vsX4xMseF14>

Дешифратор (Decoder - DC) – комбинационное устройство с m входами (x_i) и n выходами (y_j), преобразующее m -разрядный двоичный код в n -разрядный унитарный код.

	x_0	x_1	x_2	x_{m-1}	DC	y_0	y_1	y_2	y_{n-1}
1	1					0			
2		2				1			
4			4			2			
\vdots				\vdots			\vdots	\vdots	\vdots
					2^{m-1}		$N-1$		

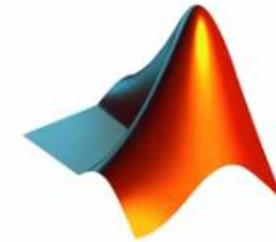
УГО дешифратора

4.3.1 Комбинационные устройства Дешифраторы и шифраторы (2017)
<https://www.youtube.com/watch?v=kR4m-BT3UwY>

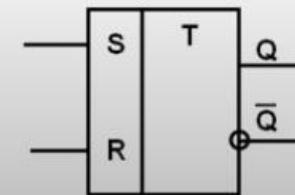
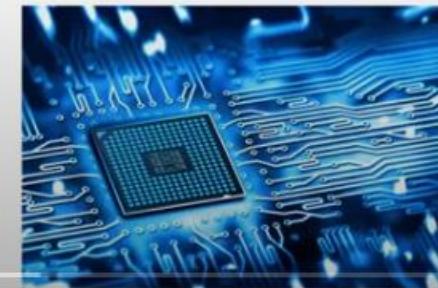


ЭКСПОНЕНТА
ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ

Engee - переходите на сайт [i](#)



Основы цифровой схемотехники 06. Шифраторы и дешифраторы



Марат Усс

▶ ▶ ⏪ 0:01 / 6:52

⏸ ⏴ ⏵ ⏷ ⏸ ⏹

Основы цифровой схемотехники. 06. Шифраторы и дешифраторы (2019)
<https://www.youtube.com/watch?v=szsxdj3gbnc>

SunRav BookReader - Д. М. Бречка. Моделирование цифровых устройств в системе Multisim

Файл Раздел Вид Панели Избранное Справка

Содержание

- Сод...
- Инд...
- По...
- Изб...
- Жу...
- Титульный лист
- Предисловие
- 1. Основы работы с Multisim
- 2. Цифровая схемотехника
 - 2.1. Моделирование простейших комбинационных схем
 - 2.2. Сложные комбинационные схемы**
 - 2.3. Сумматоры
 - 2.4. Триггеры
 - 2.5. Регистры и счетчики
- 3. Задания для самостоятельной работы
- Список использованной литературы
- Приложение А. Пример выполнения задания 3 ре
- Приложение Б. Пример выполнения заданий раз
- Приложение В. Пример выполнения заданий раз
- Приложение Г. Пример выполнения задания 3 ре

сложные комбинационные схемы. Рассмотрим дешифраторы, шифраторы, мультиплексоры и демультиплексоры как примеры сложных комбинационных устройств.

2.2.1. Дешифраторы

Материалы данного пункта приводятся на основе лабораторной работы «Преобразователи кодов» [3]. Дешифратор (DC), или декодер, – комбинационная схема с n входами и $m = 2^n$ выходами ($m > n$), преобразующая двоичный входной n -код (кодовое слово) в унитарный. На одном из m выходов дешифратора появляется логическая 1, а именно на том, номер которого соответствует поданному на вход двоичному коду. На всех остальных выходах дешифратора выходные сигналы равны нулю.

Rис. 2.5. Дешифратор

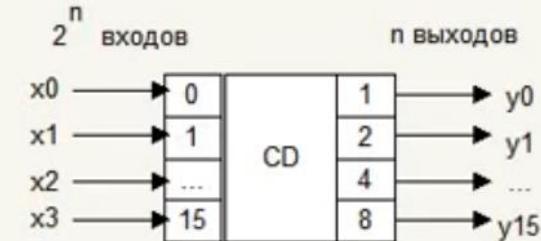
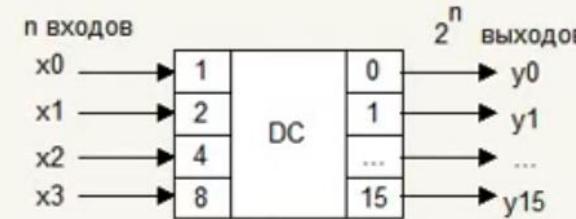
Дешифратор используют, когда нужно обращаться к различным цифровым устройствам по адресу, представленному двоичным кодом.

Шифраторы и дешифраторы (2021)
<https://www.youtube.com/watch?v=7WLaRHY108Y>

ТЕМА: ШИФРАТОРЫ/ДЕШИФРАТОРЫ

Дешифратор/DC (декодер) - это преобразователь 2ичного кода в 10ичный.

Шифратор/CD (кодер) - это преобразователь 10ичного кода в 2ичный.

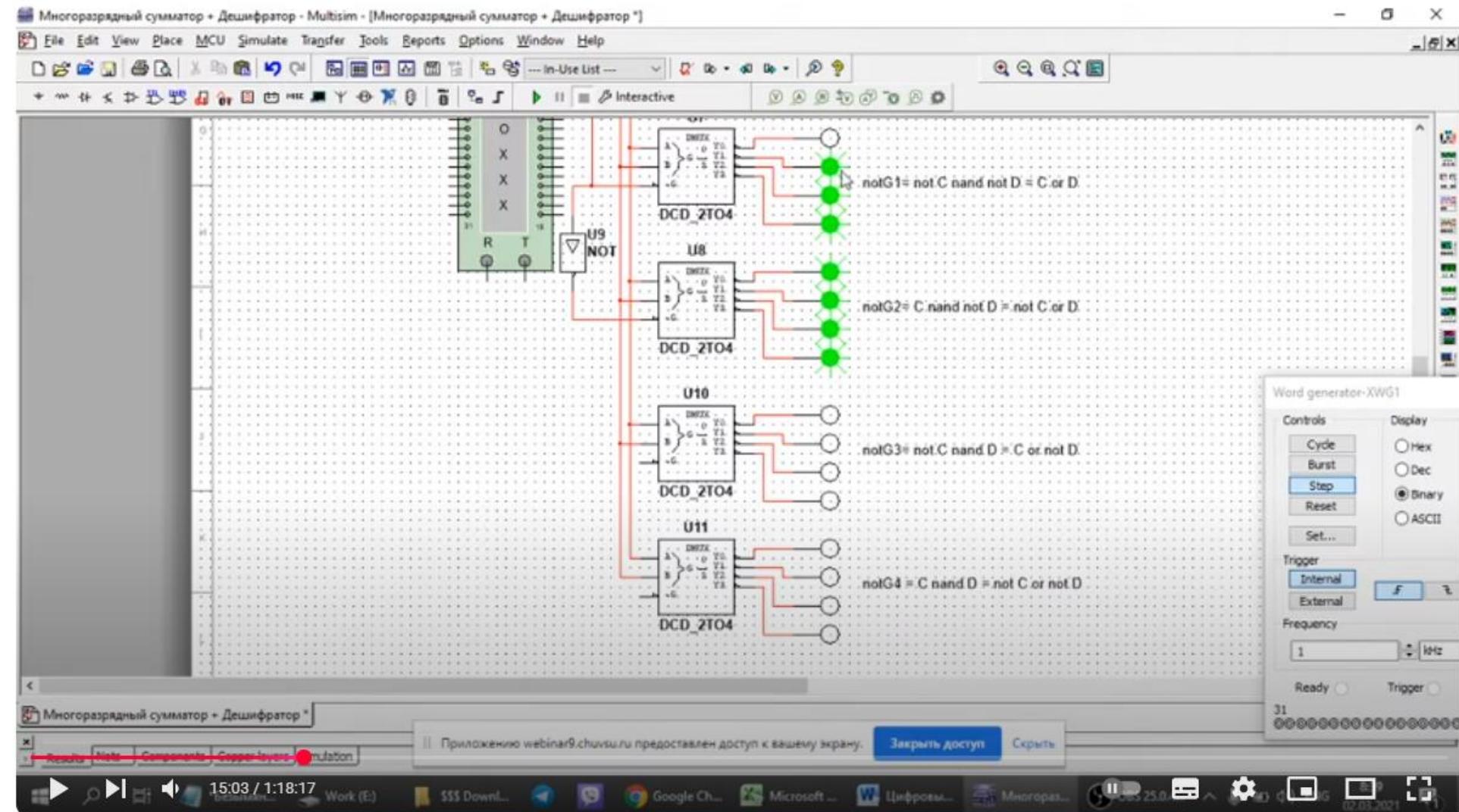


п входов			2^n выходов							
X	X	X	Y	Y	Y	Y	Y	Y	Y	Y
1	2	3	0	1	2	3	4	5	6	7
			1							
				1						
					1					
						1				
							1			
								1		
									1	
										1

2^n								п выходов		
X0	X1	X2	X3	X4	X5	X6	X7	Y3	Y2	Y1
1										
	1									
		1								
			1							
				1						
					1					
						1				
							1			
								1		
									1	

Дешифраторы и шифраторы (2017)

<https://www.youtube.com/watch?v=pBnAN5pjCDg>



2021.03.02 Лекция № 3 Дешифратор, Шифратор, Мультиплексор (2021)
<https://www.youtube.com/watch?v=0tbwCZ19aOs>

Applications Places Activities LibreOffice Writer

Doc.odt - LibreOffice Writer

File Edit View Insert Format Styles Table Form Tools Window Help

Table Contents Find & Replace (Ctrl+H)

Built on logic gates 4 → 2 coder

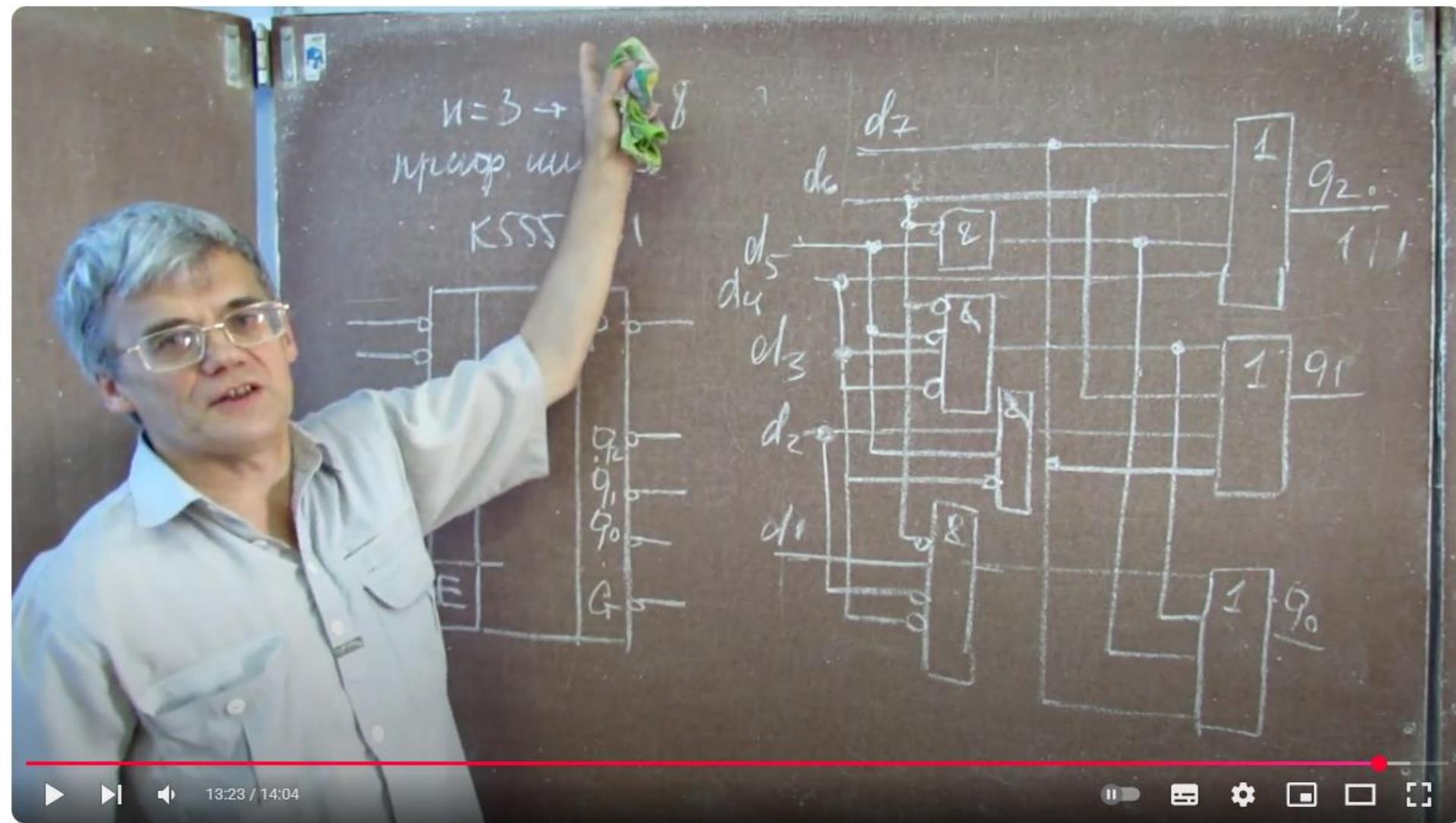
Inputs				Outputs					
x_3	x_2	x_1	x_0	Predefined			Experimental		
				y_1	y_0	GS	y_1	y_0	GS
0	0	0	0	0	0	0			
0	0	0	1	0	0	1			
0	0	1	0	0	1	1			
0	1	0	0	1	0	1			
1	0	0	0	1	1	1			

Page 8 of 9 | 371 words, 1,336 characters | Default Style | English (USA) | Table3

8:24 / 17:21

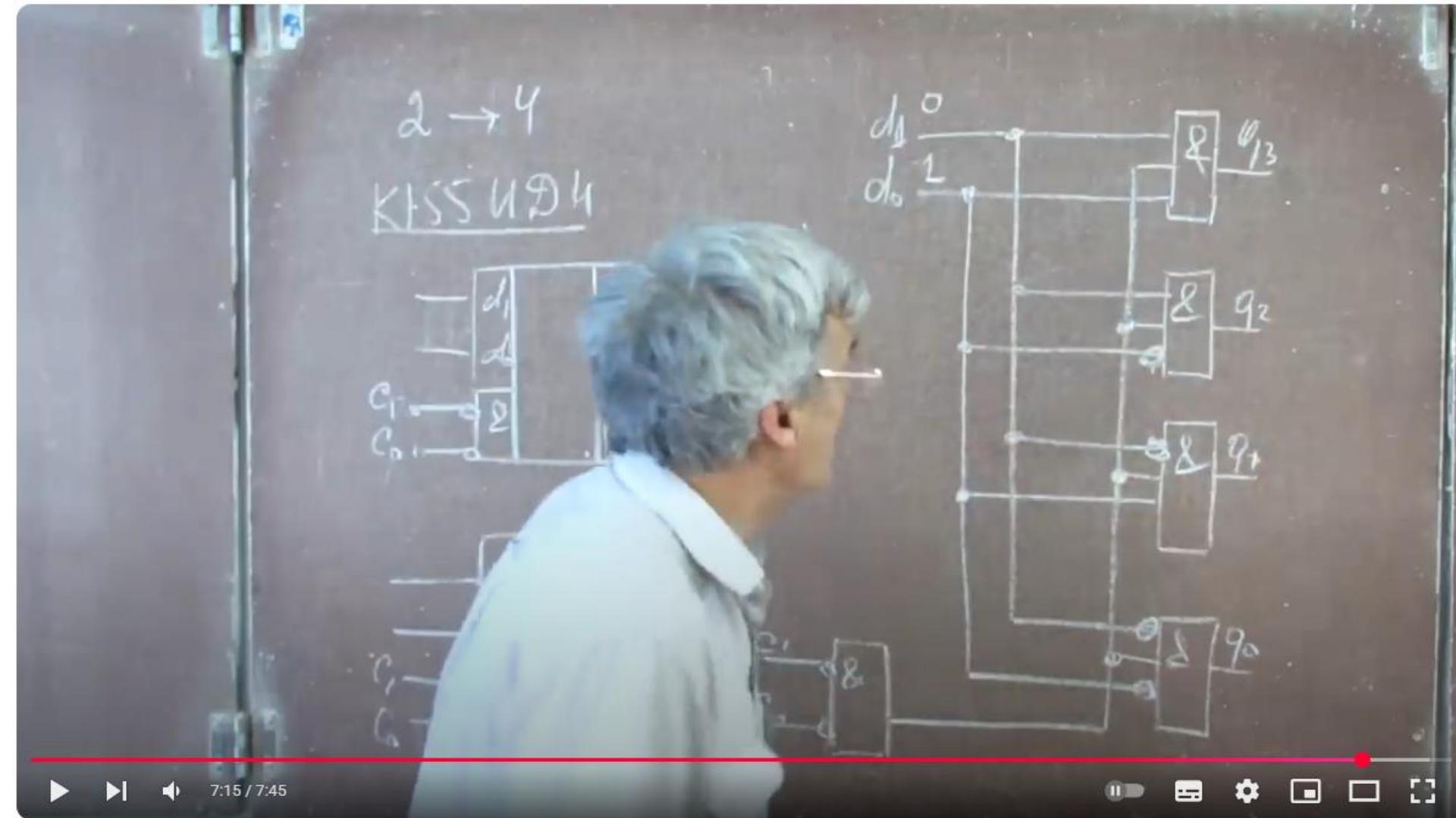
0087. Шифратор (2019)

https://www.youtube.com/watch?v=JoG_dpqyScs

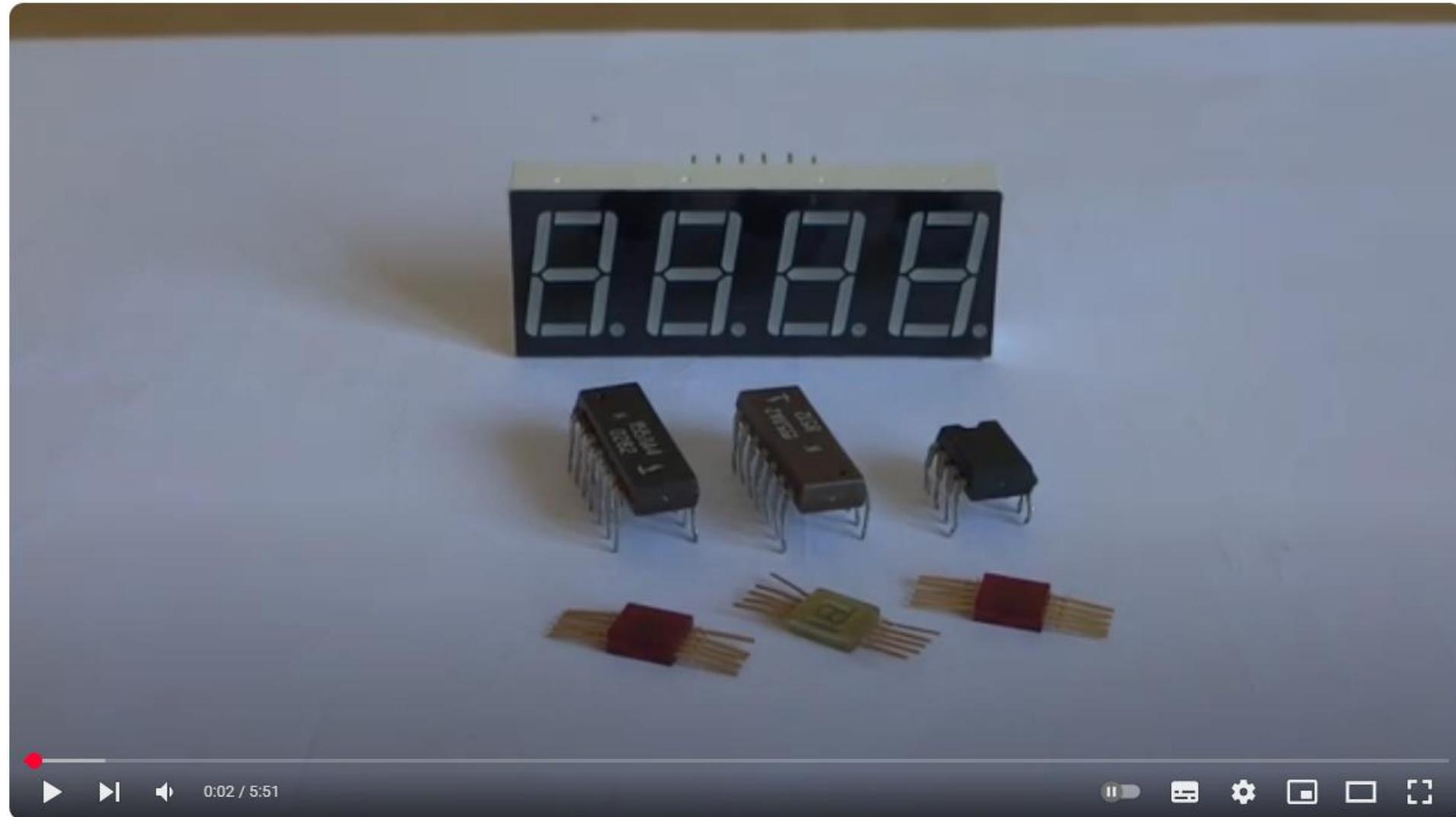


Лекция 87. Шифратор (2013)

<https://www.youtube.com/watch?v=mDhHgKF835E>



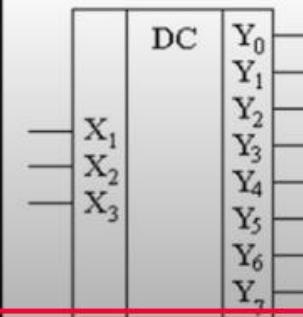
Лекция 85. Дешифратор (2013)
<https://www.youtube.com/watch?v=1Qr-ENKLgIA>



ДЕШИФРАТОР из СВЕТОДИОДОВ для семисегментного индикатора (2020)
<https://www.youtube.com/watch?v=-VoBka9gTmo>

Дешифраторы

Дешифратором, или декодером называется комбинационное логическое устройство для преобразования чисел из двоичной системы счисления в десятичную. Дешифратор выполняет операцию, обратную шифратору. Если число адресных входов дешифратора n связано с числом его выходов m соотношением $m=2^n$, то дешифратор называют **полным**. В противном случае, т. е. если $m < 2^n$, дешифратор называют **неполным**.



X_3	X_2	X_1	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

$$Y_0 = \overline{X_3} \cdot \overline{X_2} \cdot \overline{X_1}$$

$$Y_1 = \overline{X_3} \cdot \overline{X_2} \cdot X_1$$

$$Y_2 = \overline{X_3} \cdot X_2 \cdot \overline{X_1}$$

...

$$Y_7 = X_3 \cdot X_2 \cdot X_1$$



3:51 / 11:32

Шифраторы и дешифраторы (2022)
<https://www.youtube.com/watch?v=iEReAZ1BJYA>

Функциональные узлы комбинационного типа

Дешифраторы

Дешифратор (DC) – функциональный узел, преобразующий каждую комбинацию входных сигналов в информационный сигнал на выходе, номер которого соответствует десятичному эквиваленту входного кода.

Число выходов дешифратора равно числу разрешенных наборов входных сигналов. Дешифратор, имеющий n входов, может иметь $K \leq 2^n$ выходов.

Если $K = 2^n$ – полный дешифратор, $K < 2^n$ – неполный дешифратор.

Если $X = x_n x_{n-1} \dots x_1$ – слово, поступающее на вход дешифратора.

$Y = y_{m-1} y_{m-2} \dots y_0$ – слово, формируемое на выходе дешифратора ($m = 2^n$), то функция полного дешифратора описывается системой булевых функций

$$\left\{ \begin{array}{l} y_0 = \bar{x}_n \bar{x}_{n-1} \dots \bar{x}_3 \bar{x}_2 \bar{x}_1 \\ y_1 = \bar{x}_n \bar{x}_{n-1} \dots \bar{x}_3 \bar{x}_2 x_1 \\ y_2 = \bar{x}_n \bar{x}_{n-1} \dots \bar{x}_3 x_2 \bar{x}_1 \\ y_3 = \bar{x}_n \bar{x}_{n-1} \dots \bar{x}_3 x_2 x_1 \\ \dots \\ y_{m-2} = x_n x_{n-1} \dots x_3 x_2 \bar{x}_1 \\ y_{m-1} = x_n x_{n-1} \dots x_3 x_2 x_1 \end{array} \right. . \quad (1)$$

Основные параметры дешифратора

- 1) Число входов n , число выходов K .
- 2) Входные и выходные токи и напряжения логических 0 и 1, напряжение статической помехи.
- 3) Допустимый ток или мощность, потребляемые схемой дешифратора.
- 4) Динамический параметр: $t_{здр}^{0/1}$ и $t_{здр}^{1/0}$ – время задержки распространения сигнала в дешифраторе при переключении из 0 в 1 и из 1 в 0.

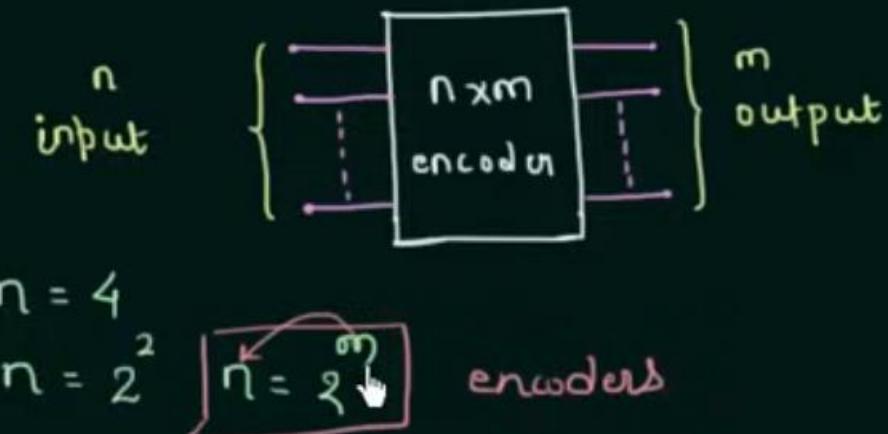
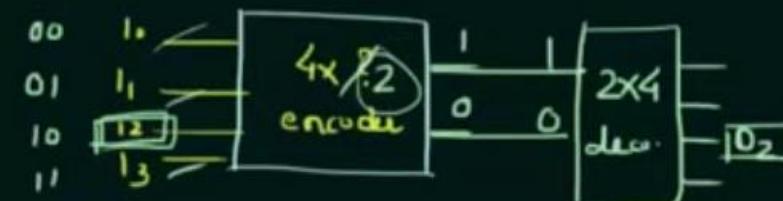
Существуют три основных способа построения дешифраторов: линейный, пирамидальный и ступенчатый.

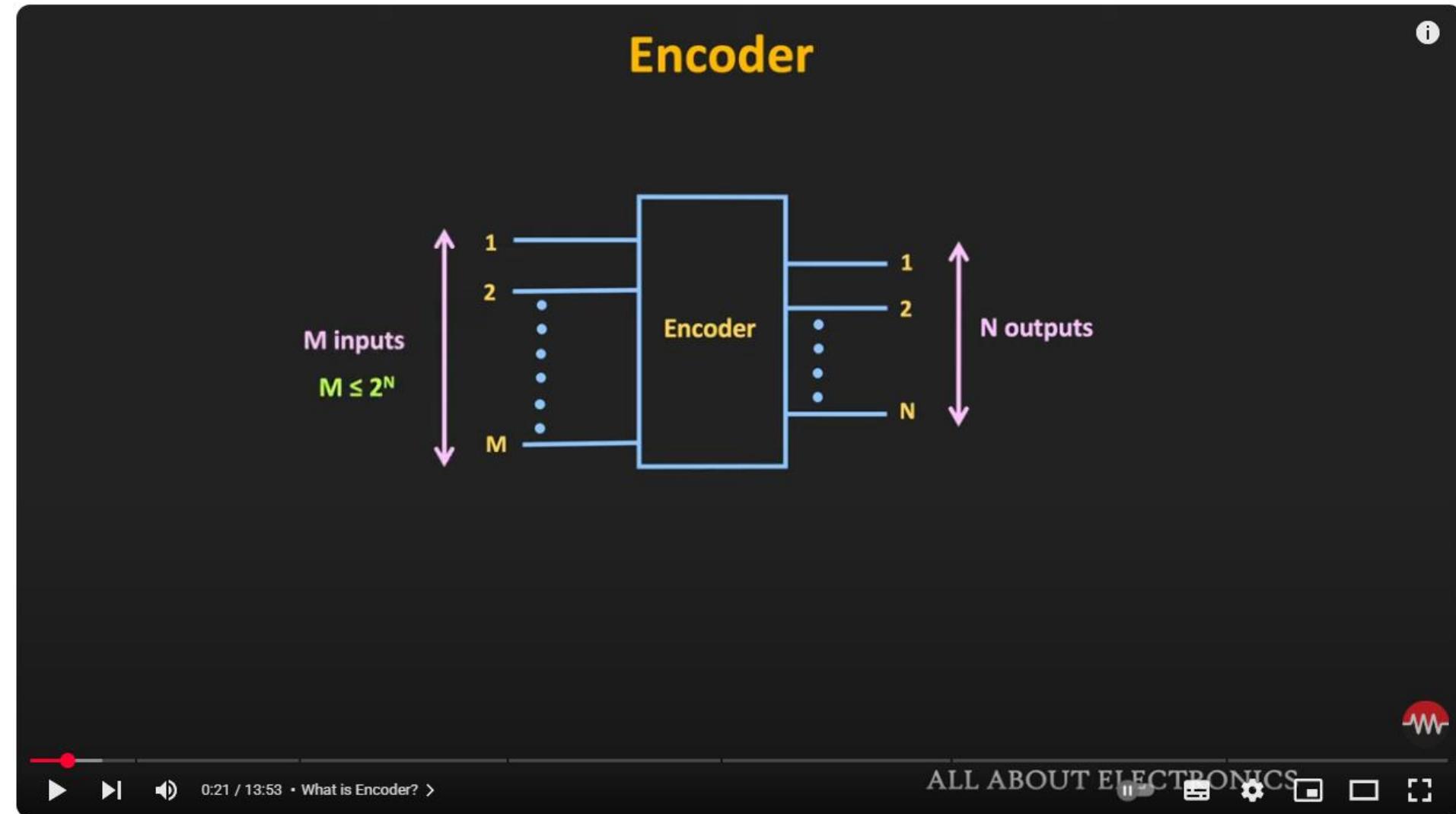
РК6. Схемотехника. Дешифраторы (2020)
<https://www.youtube.com/watch?v=ynofPq-3FEI>

Introduction to Encoders & Decoders

- » They are combinational circuits.
- » Encoders have "n" input and "m" output.
- » Function of decoder is opposite to encoder.

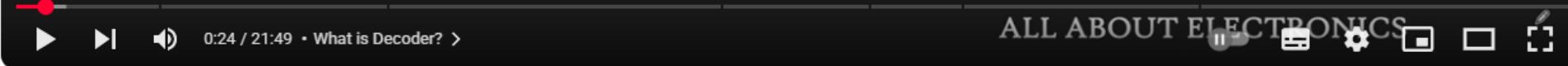
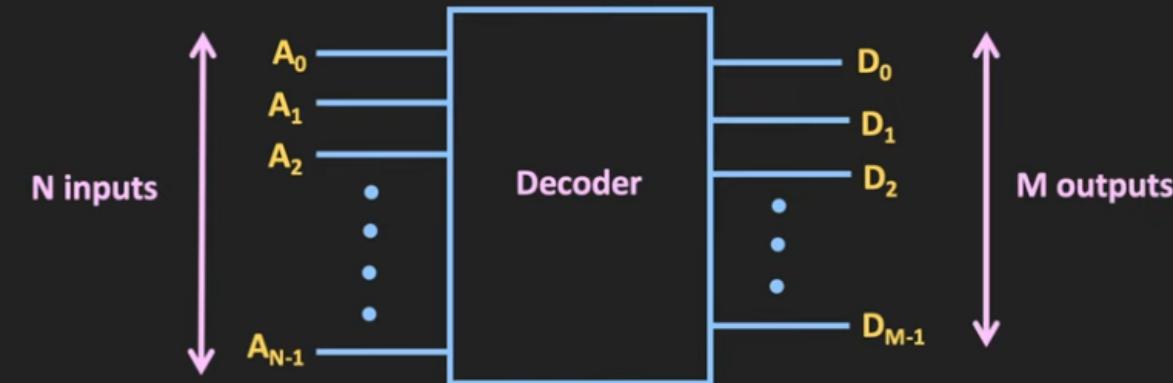
M151



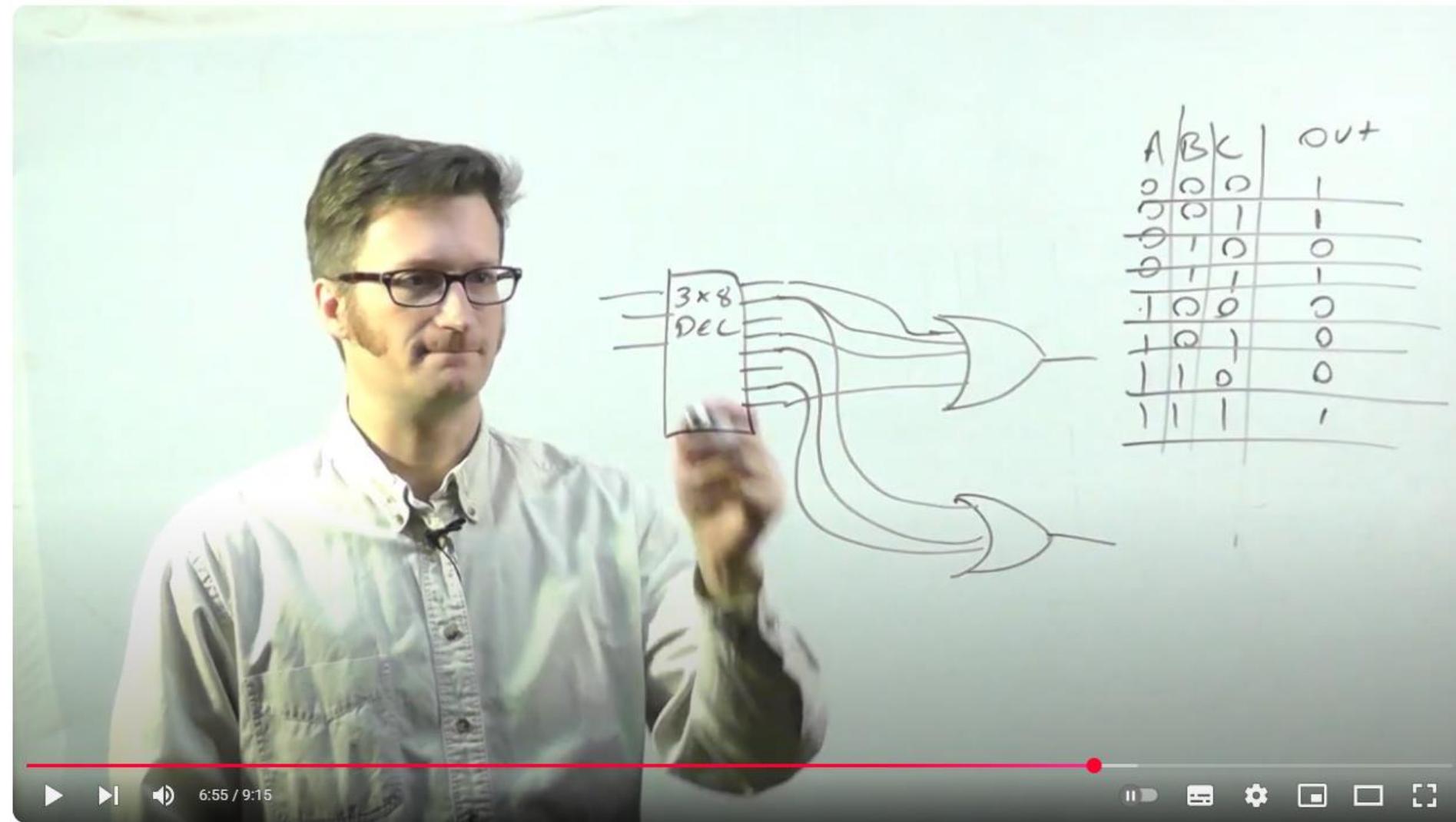


Encoder in Digital Electronics | Working, Application and Logic Circuit of Encoder(2022)
Кодировщик в цифровой электронике | Принцип работы, применение и логическая схема Шифратора (2022)
<https://www.youtube.com/watch?v=NWiPVMDh7GE>

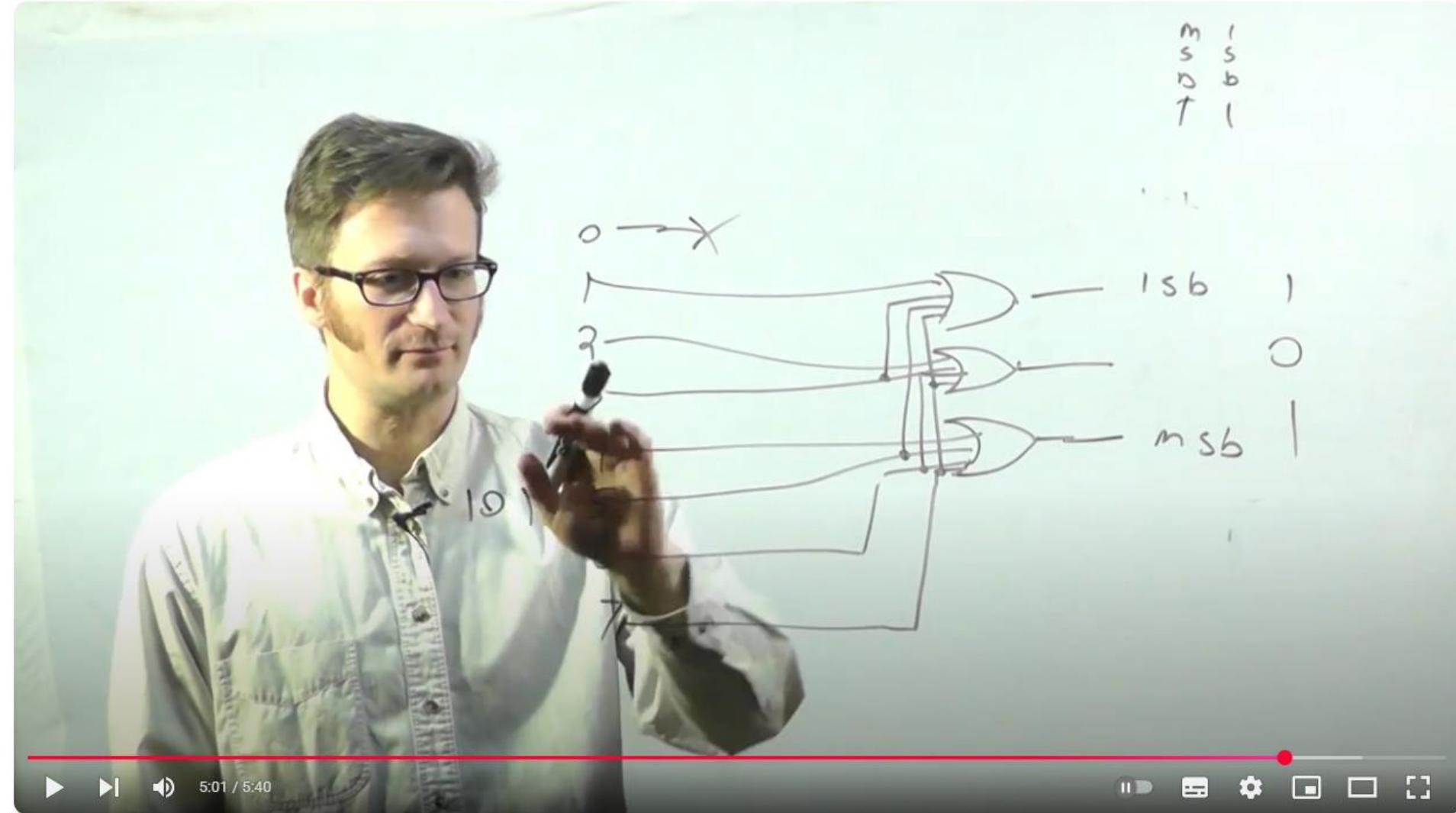
Decoder



Decoder Explained | What is Decoder? Applications of Decoder | 5 to 32 Decoder using 3 to 8 Decoders (2022)
Декодер: объяснение | Что такое Дешифратор? Применение дешифратора | Дешифратор 5-32 с использованием 3-8 дешифратора (2022)
<https://www.youtube.com/watch?v=a3wh7xV5PBU>



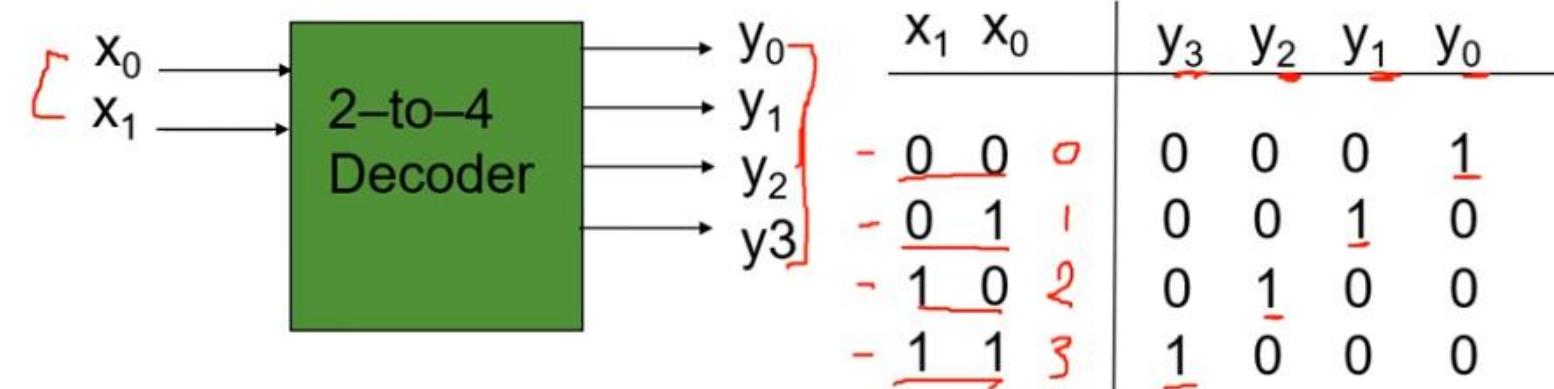
Digital Logic – Decoders (2013)
Цифровая логика – Дешифраторы (2013)
<https://www.youtube.com/watch?v=1prCy3kdyIU>



▶ ▶ 🔍 5:01 / 5:40

Digital Logic - Encoders (2013)
Цифровая логика - Шифраторы (2013)
<https://www.youtube.com/watch?v=s6V4cHFsQBE>

2-4 decoder



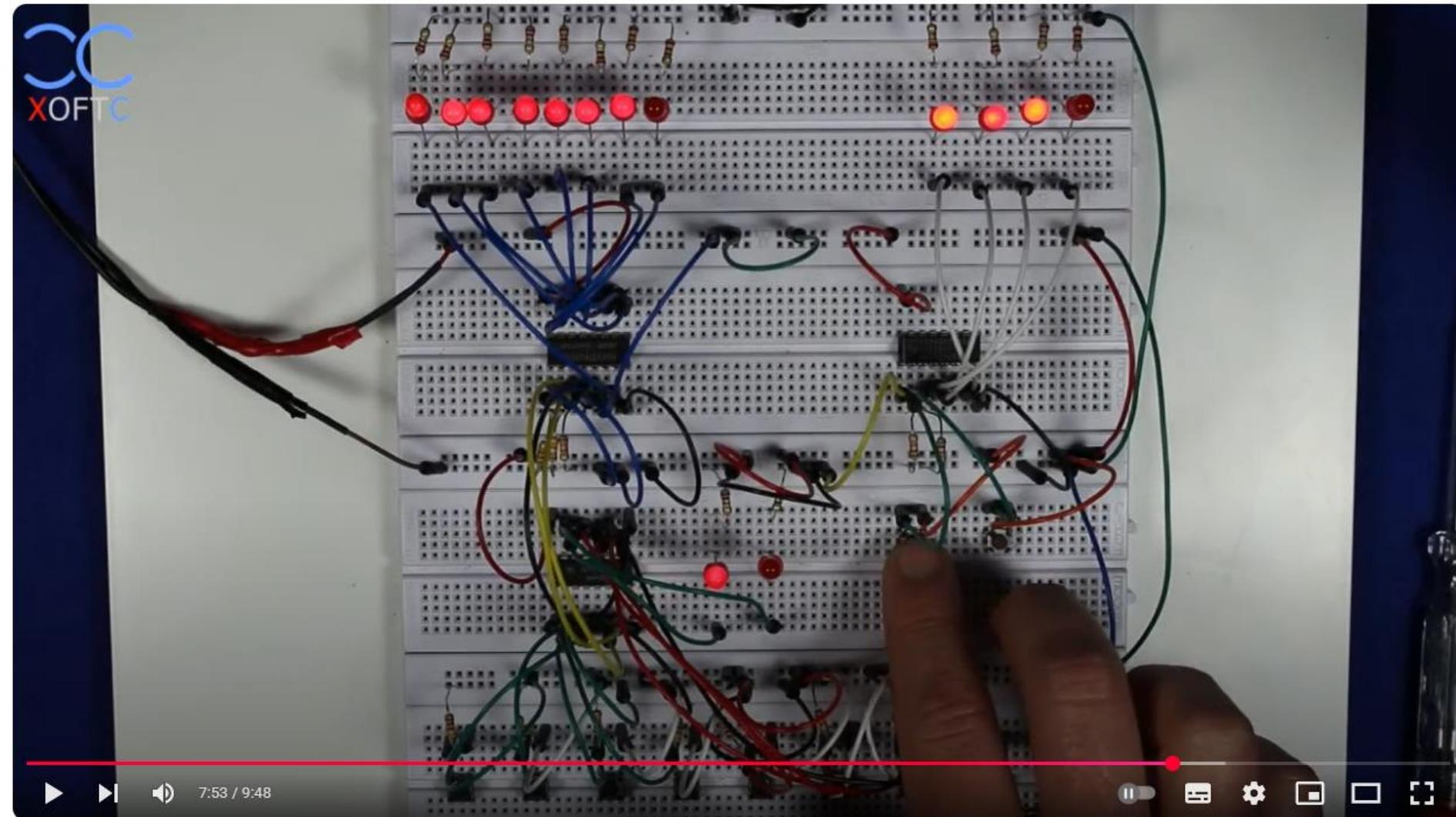
$$y_0 = m_0 = \overline{x_1} \overline{x_0}$$

$$y_1 = m_1 = \overline{x_1} x_0$$

$$y_2 = m_2 = x_1 \overline{x_0}$$

$$y_3 = m_3 = x_1 x_0$$





How To Use Line Decoders/Encoders(2016)
Как использовать линейные Дешифраторы/Шифраторы (2016)
<https://www.youtube.com/watch?v=DXIDgVKGg5c>

Encoder and Decoder Combinational Circuit

1) Encoders and Decoders are used to convert the data from one form to another form.
 2) Encoders have N inputs and M outputs
 3) Decoders have M inputs and N outputs
 4) Function of Decoder is exactly opposite to Encoder

$N > M$

$N = 2^M$

$4 = 2^2$ $\sqrt{4} = 2$

$8 = 2^3$ $\sqrt[3]{8} = 2$

4×2 Encoder

2×4 Dec

D_0 D_1 D_2 D_3

D_0 D_1 D_2 D_3

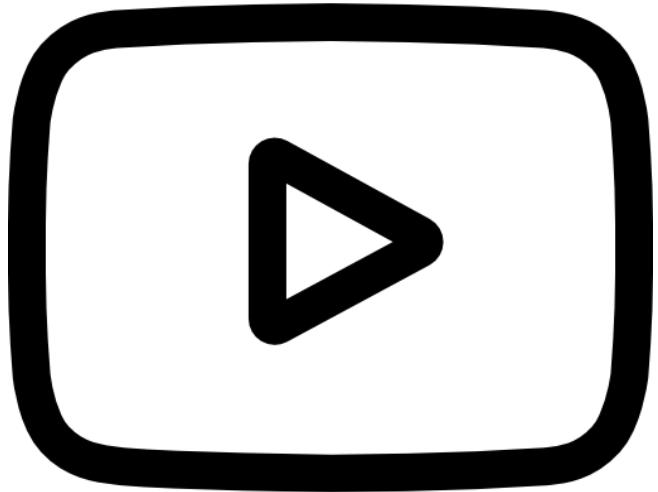
Y_0 Y_1

Y_0 Y_1 Y_2 Y_3

$Y_0 = 0$
 $Y_1 = 1:0$
 $Y_2 = 0:1$
 $Y_3 = 0$

6:03 / 9:49 • Theory >

Encoder and Decoder Combinational Circuits | 4 x 2 Encoder & 2 x 4 Decoder with Truth Table (2018)
 Комбинированные дешифраторы и шифраторы | 4 x 2 шифратор и 2 x 4 дешифратор с таблицей истинности (2018)
<https://www.youtube.com/watch?v=rEwJZAPqmGs>



Преобразователи кодов



СТРУКТУРА ПРЕОБРАЗОВАТЕЛЕЙ КОДОВ ЧАСТЬ 1

Комбинационные ЦУ



Видеоурок 3.18. Структура преобразователей кодов. Часть 1 (2023)
<https://www.youtube.com/watch?v=DeT-Y-ommIM>

видеоурок 2.7

Структура преобразователей кодов

16, 8, 4, 2

The diagram illustrates the structure of code converters for a 16-to-4 bit converter. It shows six Karnaugh maps (K-maps) for Y_0 , Y_1 , Y_2 , Y_3 , Y_4 , and a diamond-shaped summary map. The K-maps have four columns (X_2 , \bar{X}_2 , X_1 , \bar{X}_1) and three rows (X_3 , \bar{X}_3 , X_0). The summary map has four columns (X_2 , \bar{X}_2 , X_1 , \bar{X}_1) and four rows (Y_4 , Y_3 , Y_2 , Y_1). Colored lines and boxes highlight specific minterms in each map.

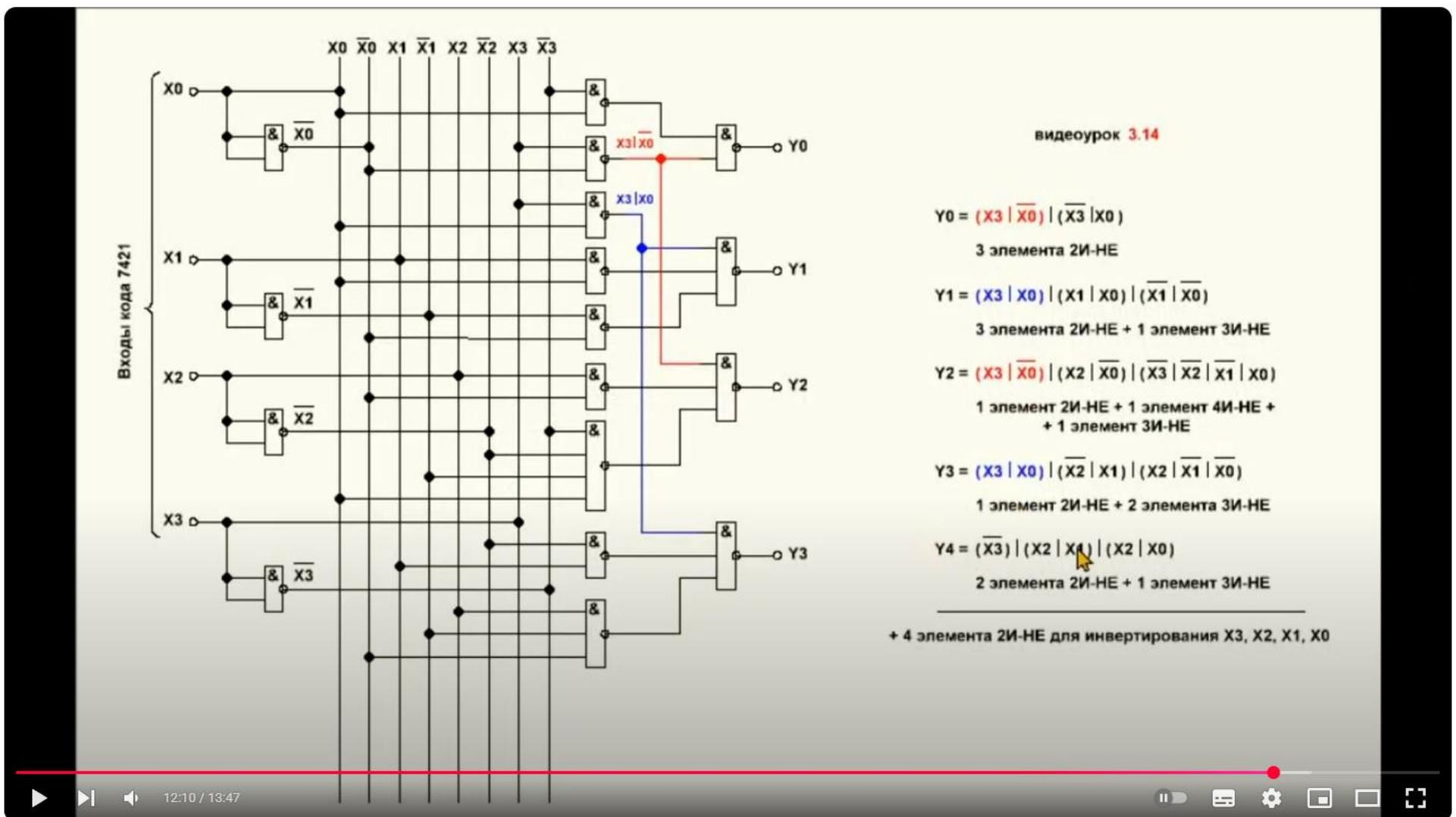
Вideoурок 3.19. Структура преобразователей кодов. Часть 2 (2023)
<https://www.youtube.com/watch?v=gQfoQ8BOF60>

склеивание видеоурок 2.4
видеоуроки 2.5 - 2.8

Структура преобразователей кодов

МДНФ	базис И,ИЛИ,НЕ
$Y_0 = X_3 \cdot \bar{X}_0 \vee \bar{X}_3 \cdot X_0$	$Y_0 = X_3 \cdot \bar{X}_0 \vee X_1 \cdot X_0 \vee \bar{X}_1 \cdot \bar{X}_0$
$Y_1 = X_3 \cdot X_0 \vee X_1 \cdot \bar{X}_0 \vee \bar{X}_1 \cdot \bar{X}_0$	$Y_1 = X_3 \cdot \bar{X}_0 \vee X_2 \cdot \bar{X}_0 \vee \bar{X}_3 \cdot \bar{X}_2 \cdot \bar{X}_1 \cdot X_0$
$Y_2 = X_3 \cdot \bar{X}_0 \vee X_2 \cdot \bar{X}_0 \vee \bar{X}_3 \cdot \bar{X}_2 \cdot \bar{X}_1 \cdot X_0$	$Y_2 = X_3 \cdot X_0 \vee \bar{X}_2 \cdot X_1 \vee X_2 \cdot \bar{X}_1 \cdot \bar{X}_0$
$Y_3 = X_3 \cdot X_0 \vee \bar{X}_2 \cdot X_1 \vee X_2 \cdot \bar{X}_1 \cdot \bar{X}_0$	$Y_3 = X_3 \vee X_2 \cdot X_1 \vee X_2 \cdot X_0$
$Y_4 = X_3 \vee X_2 \cdot X_1 \vee X_2 \cdot X_0$	

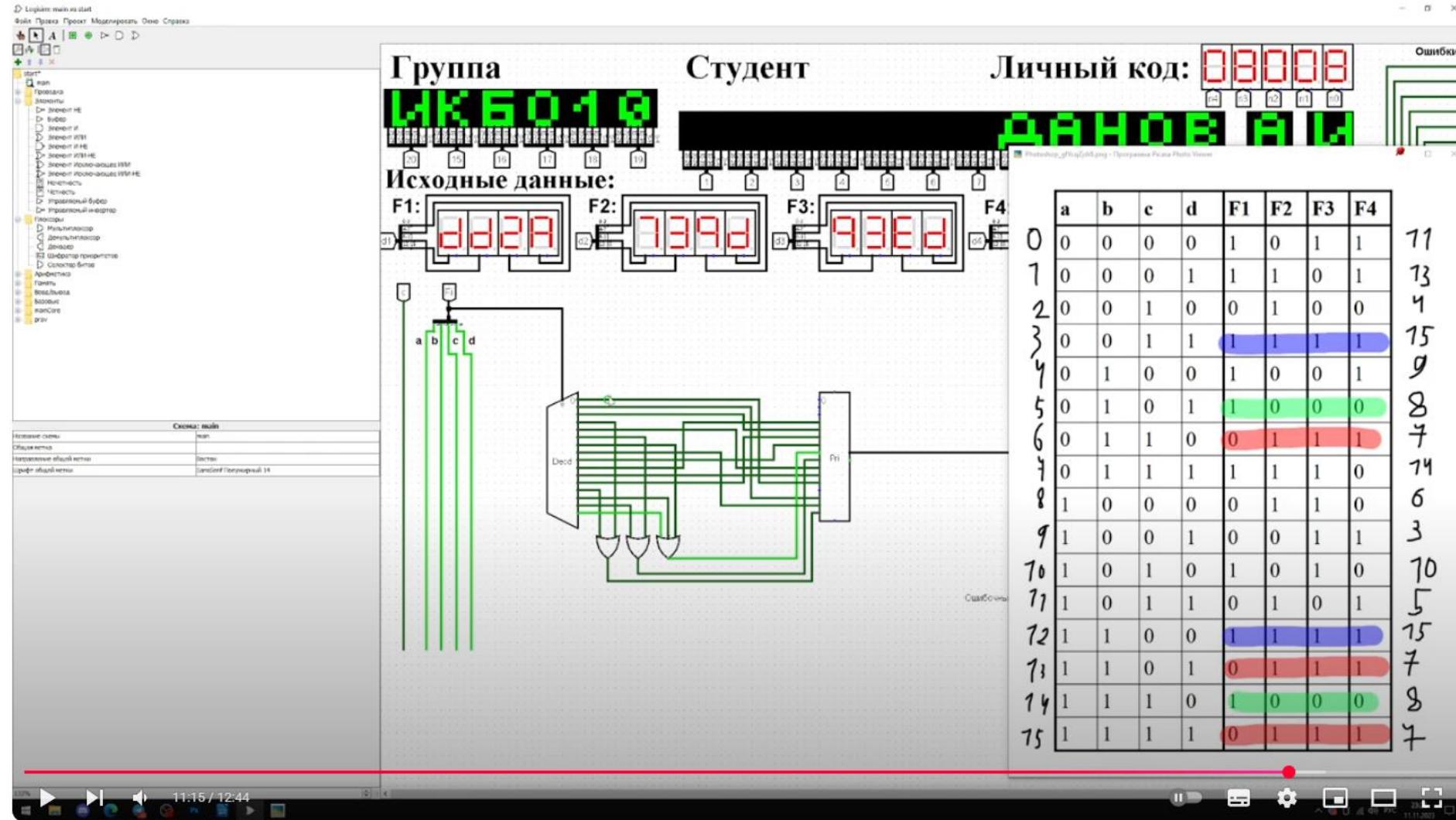
Вideoурок 3.20. Структура преобразователей кодов. Часть 3 (2023)
<https://www.youtube.com/watch?v=nTzc-C2GUOw>



Видеоурок 3.21. Структура преобразователей кодов. Часть 4 (2023)

<https://www.youtube.com/watch?v=Vg7fEPPragI>

<https://www.youtube.com/watch?v=Vg7fEPPragI>



Информатика, 9 практическая работа - Преобразователь кодов(2023)
<https://www.youtube.com/watch?v=PvQXcFluhNA>

Solⁿ:

Decimal Digit	8 4 -2 -1	BCD(8421)
	A B C D	80x4z
0	0 0 0 0	0 0 0 0
1	0 1 1 1	0 0 0 1
2	0 1 1 0	0 0 1 0
3	0 1 0 1	0 0 1 1
4	0 1 0 0	0 1 0 0
5	1 0 1 1	0 1 0 1
6	1 0 1 0	0 1 1 0
7	1 0 0 1	0 1 1 1
8	1 0 0 0	1 0 0 0
9	1 0 0 1	1 0 0 1

map for w

AB \ CD

00	0	X	X	X
01	0	0	0	0
11	X			
10	1	0	0	0

A { 11 } B { 10 }

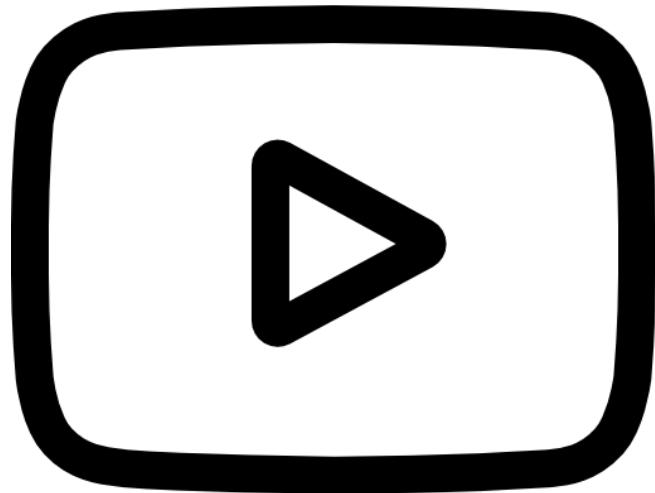
w =



5:09 / 12:19 • Problem statement >

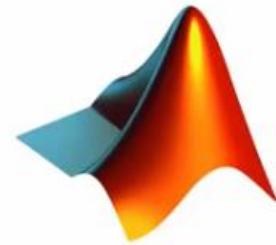


Q. 4.8: Design a code converter that converts a decimal digit from the 8, 4, -2, -1 code to BCD (2020)
 Вопрос 4.8: Разработайте преобразователь кодов, который преобразует десятичную цифру из кода 8, 4, -2, -1 в BCD (2020)
<https://www.youtube.com/watch?v=yqK9XSBGRqU>



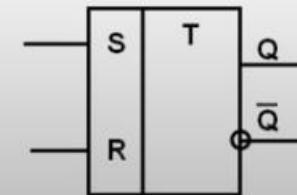
Компараторы





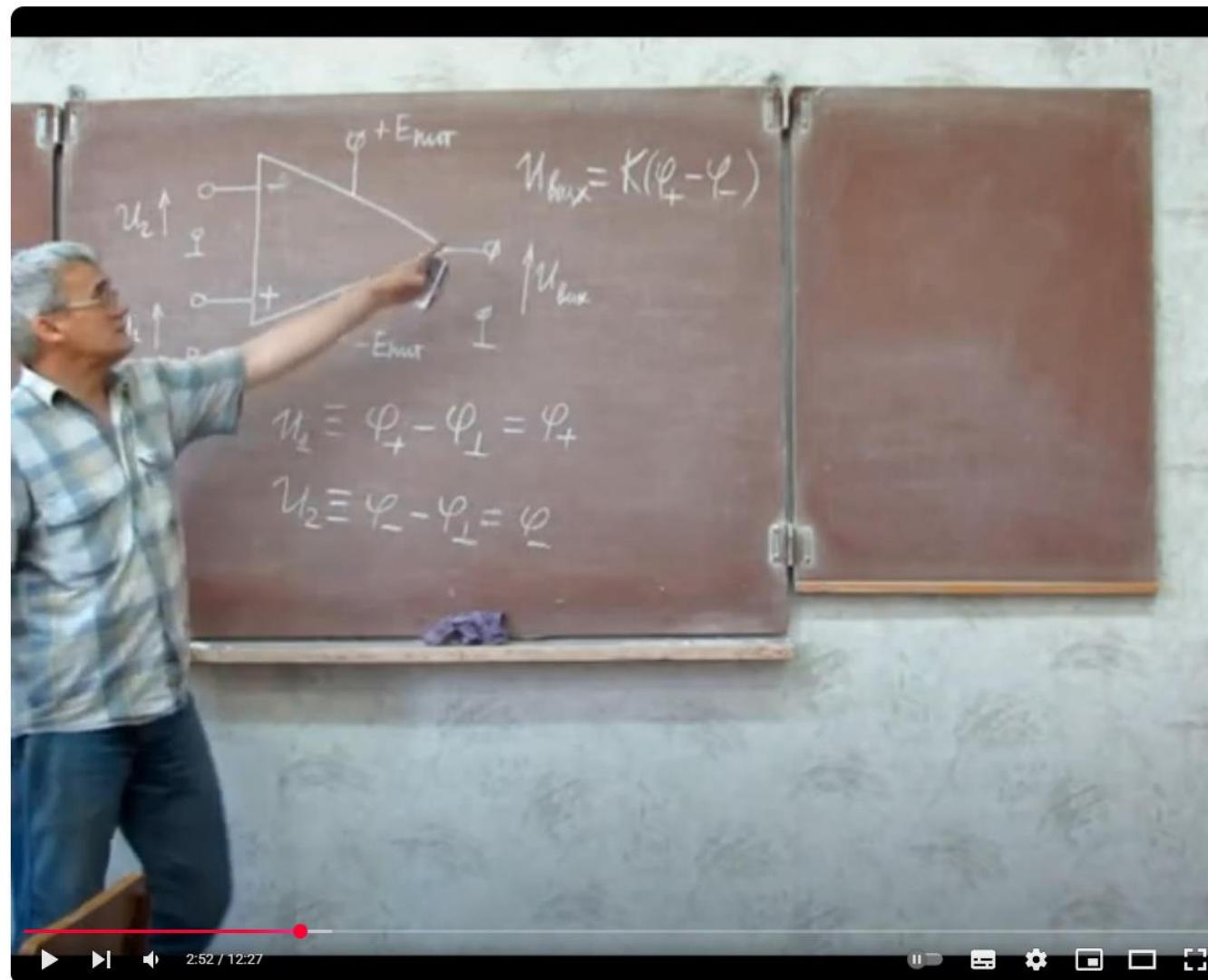
Основы цифровой схемотехники

10. Компараторы

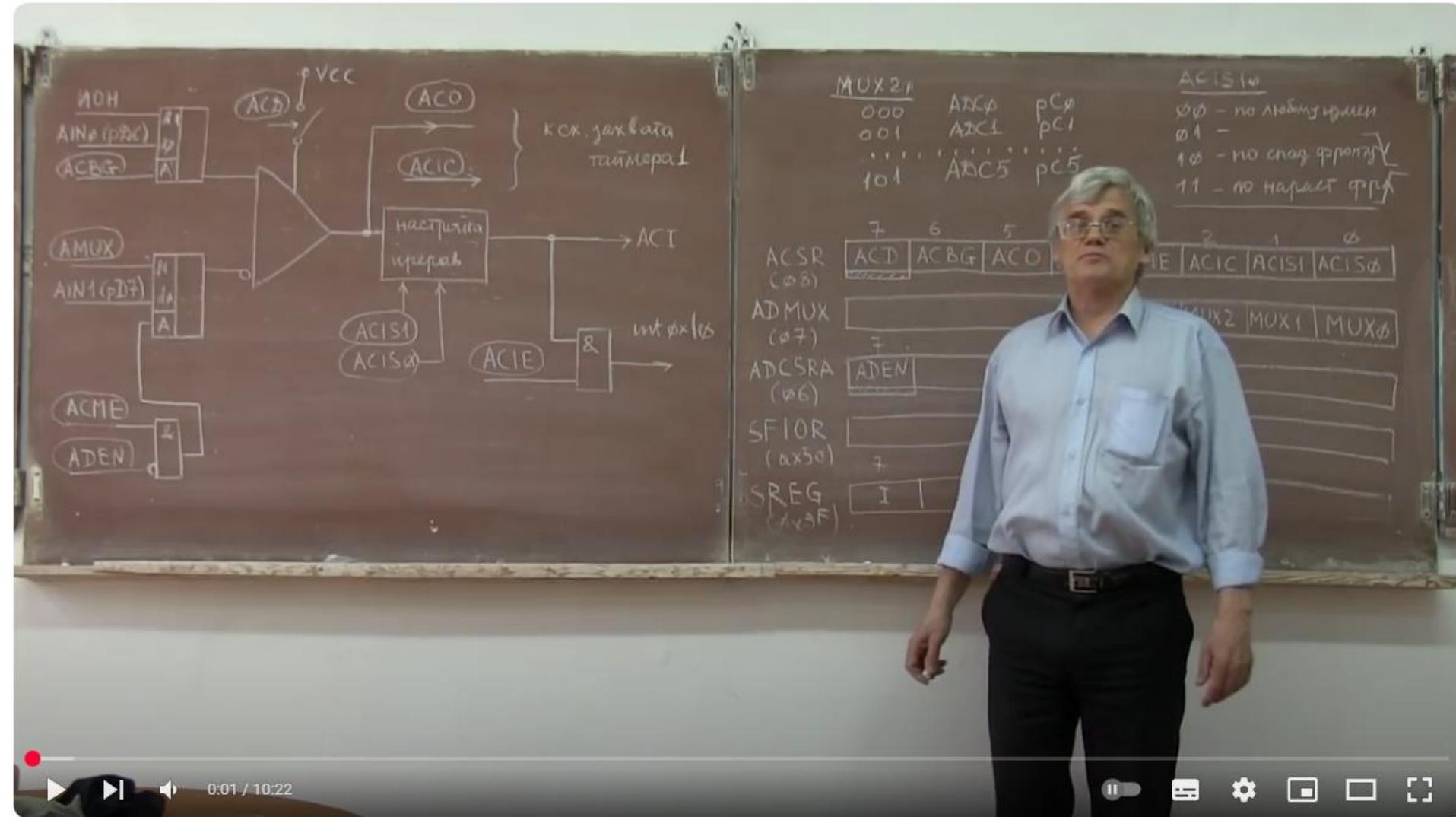


Марат Усс

Основы цифровой схемотехники. 10. Компараторы (2019)
<https://www.youtube.com/watch?v=xQ-HUUDEpng>



Лекция 18 Операционный усилитель,. Компаратор. (2012)
<https://www.youtube.com/watch?v=Gw5D-pLF54w>



лекция 371 компаратор в Atmega8 (2018)
<https://www.youtube.com/watch?v=p3fhNmWGJbc>

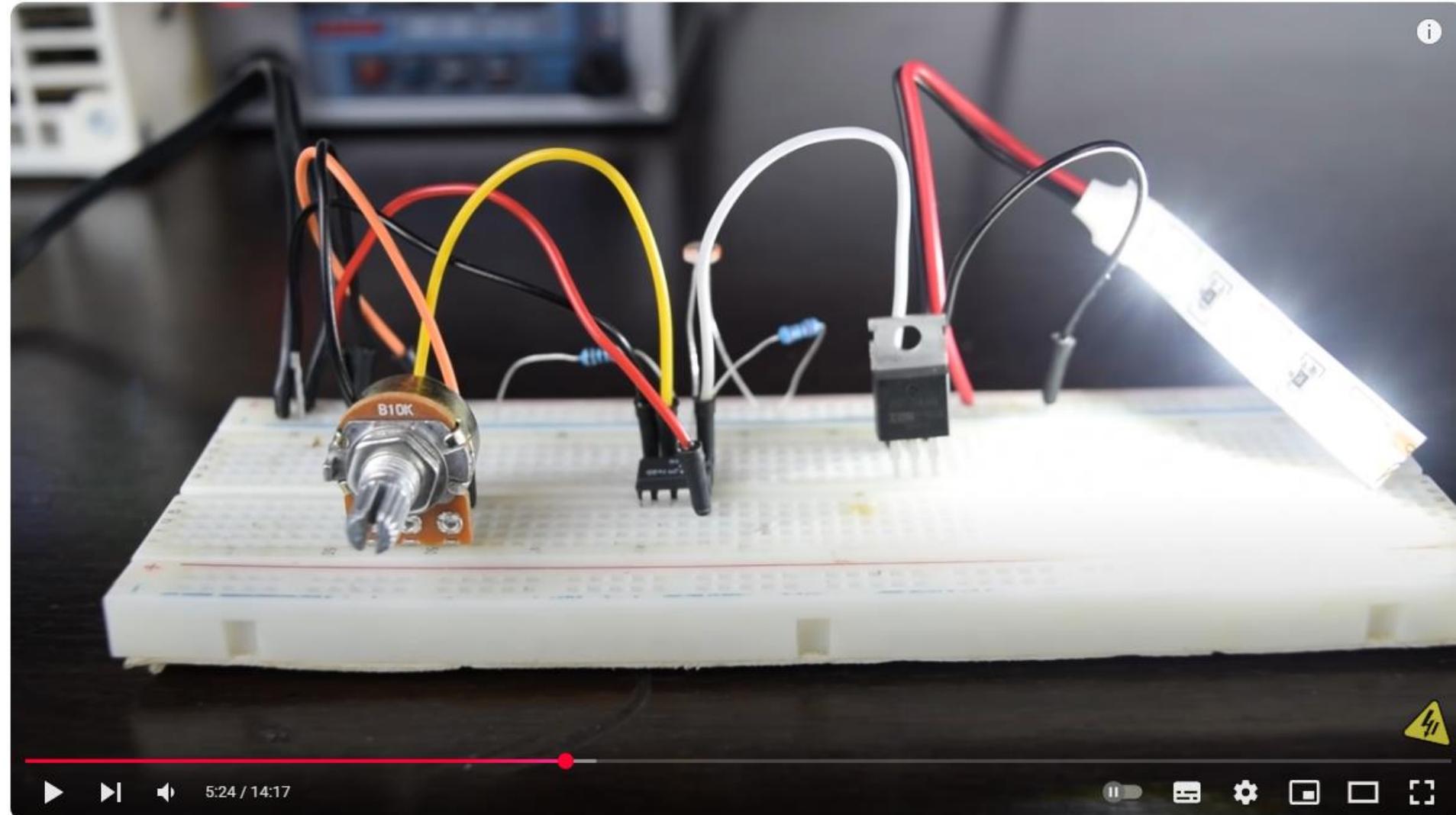
Цифровые компараторы

Цифровой компаратор – это комбинационное устройство, предназначенное для сравнения двух двоичных кодов.

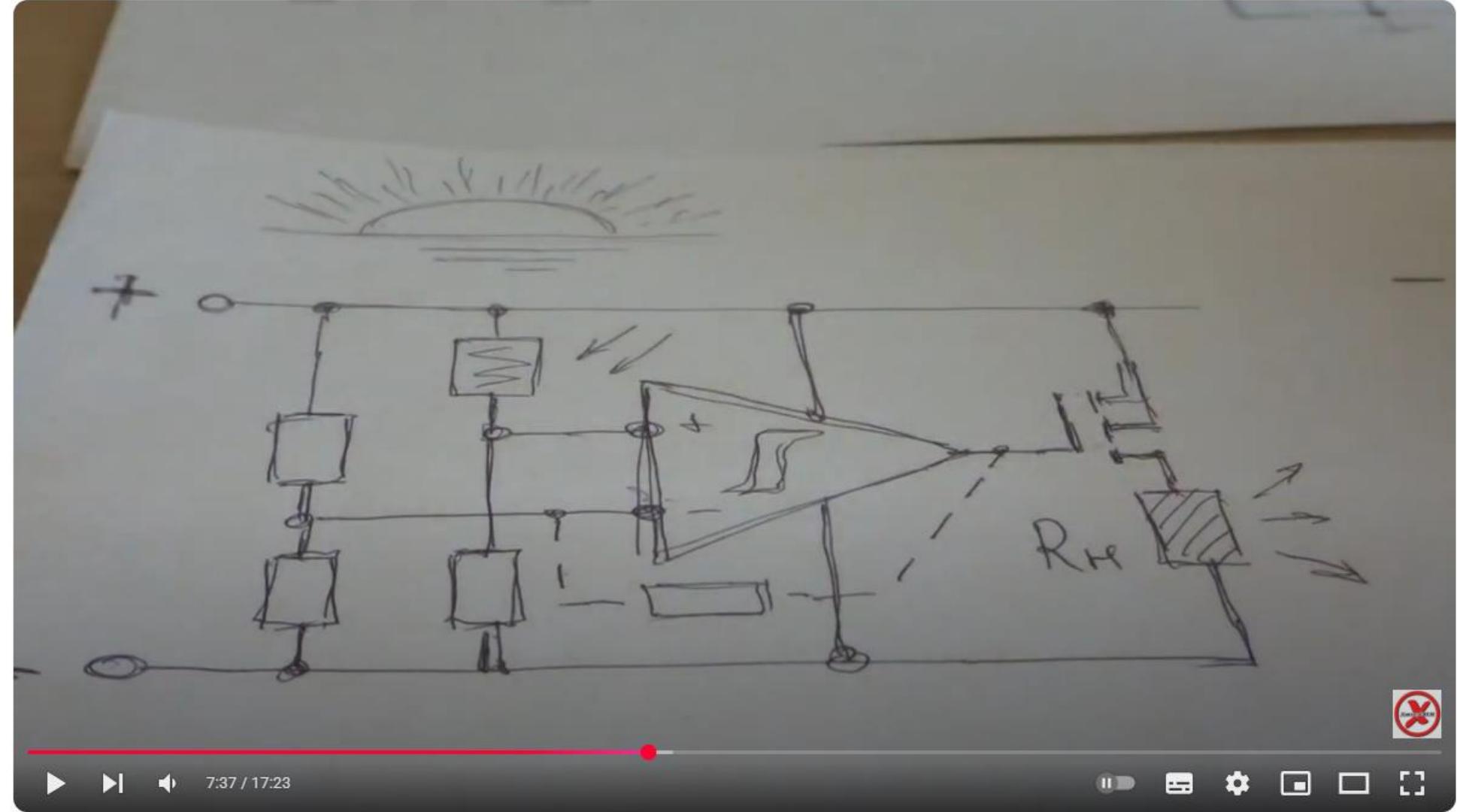
a_0	a_1	a_2	a_3	$==$	
$I(a > b)$	$I(a = b)$	$I(a < b)$		$a > b$	
b_0	b_1	b_2	b_3	$a = b$	
				$a < b$	

УГО цифрового компаратора:

4 3 4 Цифровые компараторы (2017)
<https://www.youtube.com/watch?v=tDRodUFaLao>



КОМПАРАТОР в электронике. Самое понятное объяснение! (2021)
<https://www.youtube.com/watch?v=bwNZcf9K5IE>

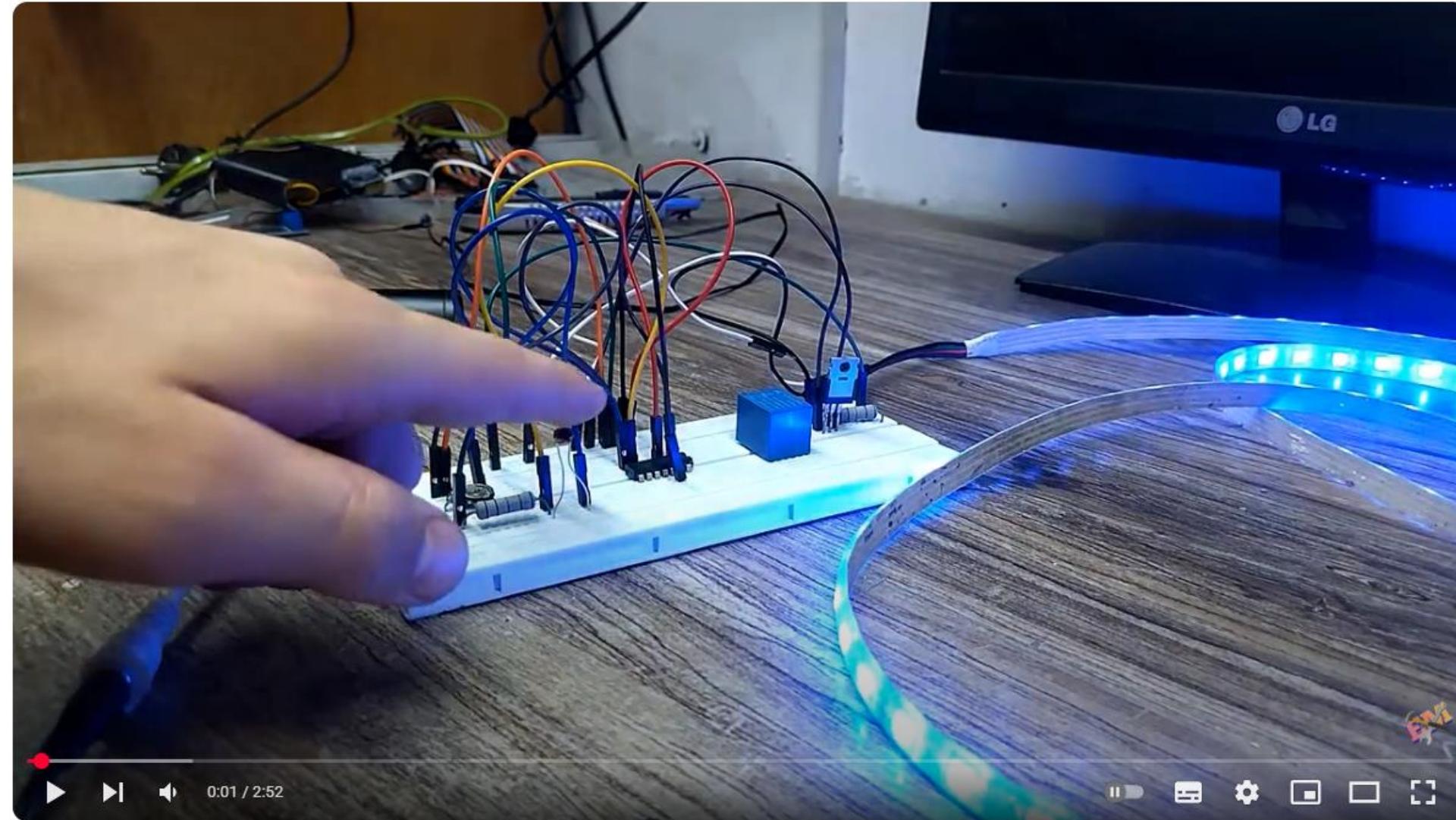


КОМПАРАТОР что за МИКРОСХЕМА и зачем он нужен И чем его просто заменить (2021)
<https://www.youtube.com/watch?v=a-XB6GZFfpA>

The image shows a breadboard circuit and its corresponding schematic diagram. The breadboard circuit is built on a white breadboard with various components like resistors, capacitors, and a microcontroller. The schematic diagram on the right is a detailed circuit diagram for a comparator using an LM393N integrated circuit. The circuit has two inputs: one connected to ground through a 10k potentiometer (PH1) and another connected to +5V through a 50k resistor (R5). The output of the IC1A (LM393N) is connected to an LED1 through a 4.7k resistor (R4) and a 2.7k resistor (R2). A third resistor (R3) is connected between the +5V supply and the non-inverting input of the IC1A. The inverting input of the IC1A is connected to ground through a 10k resistor (R6).

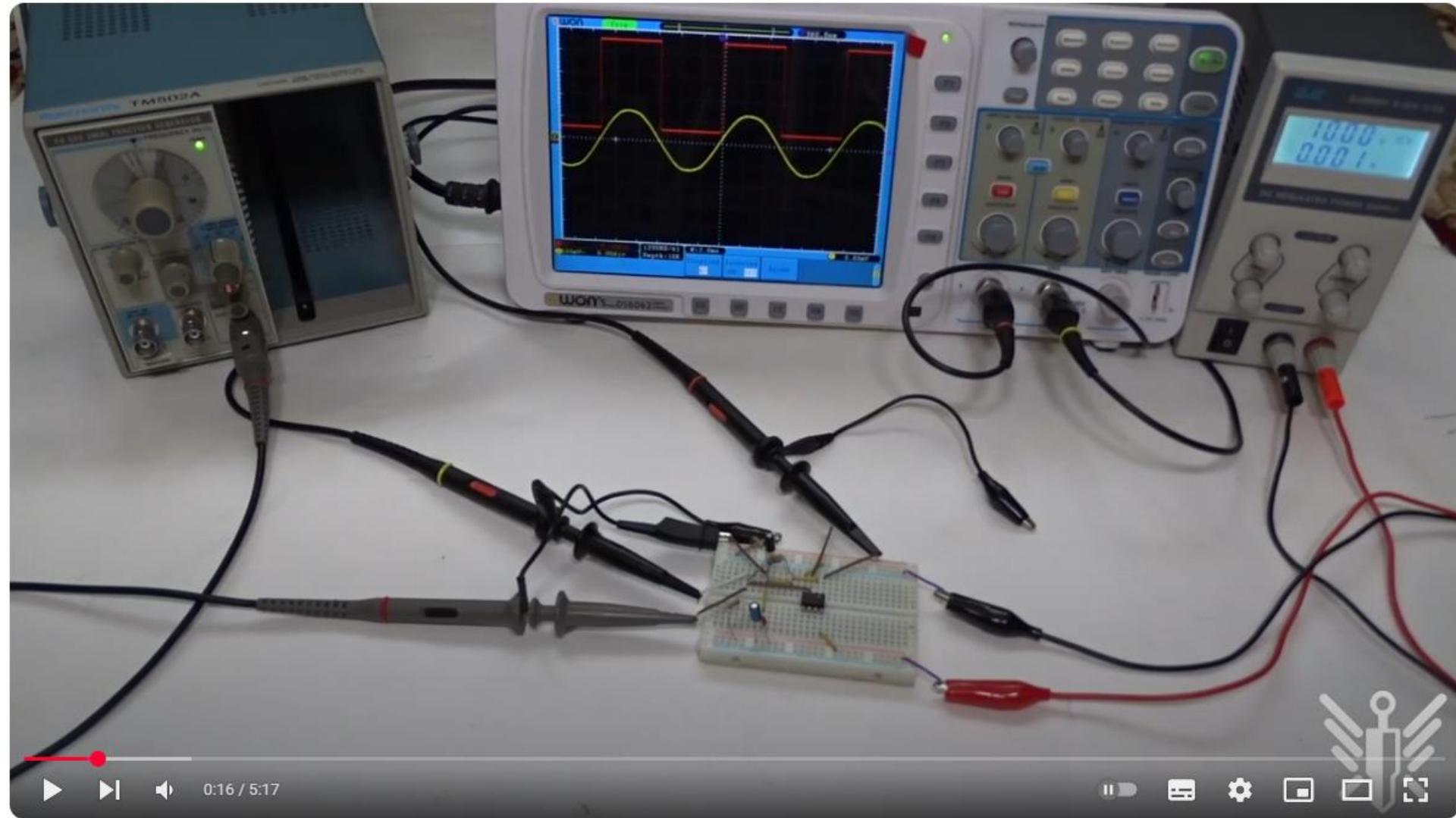
Компаратор (2018)

<https://www.youtube.com/watch?v=7-2fe7tHAMQ>

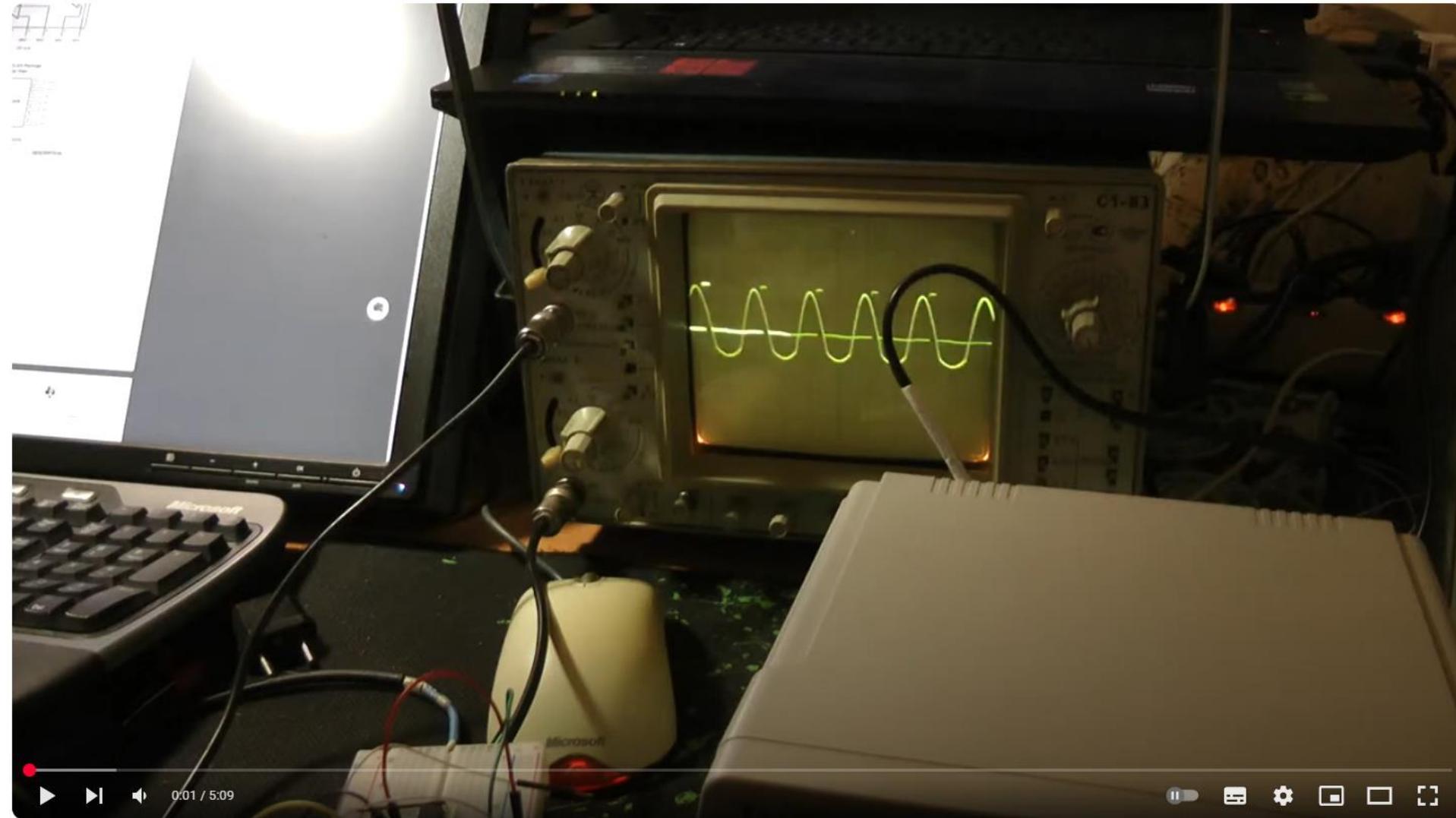


Чудо электротехники - Компаратор (2022)
<https://www.youtube.com/watch?v=xorxp8MaFew>

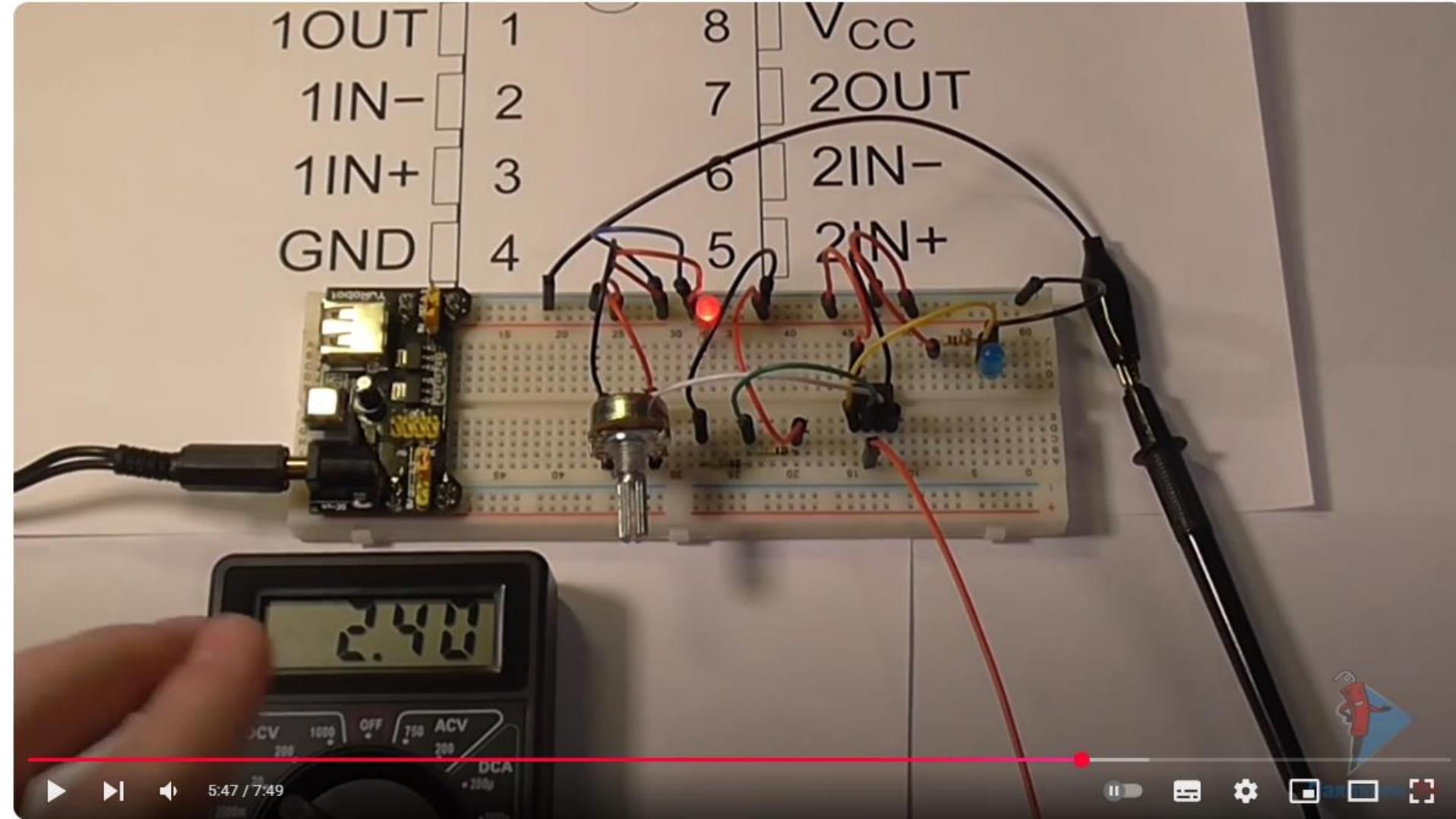
YouTube 5:17



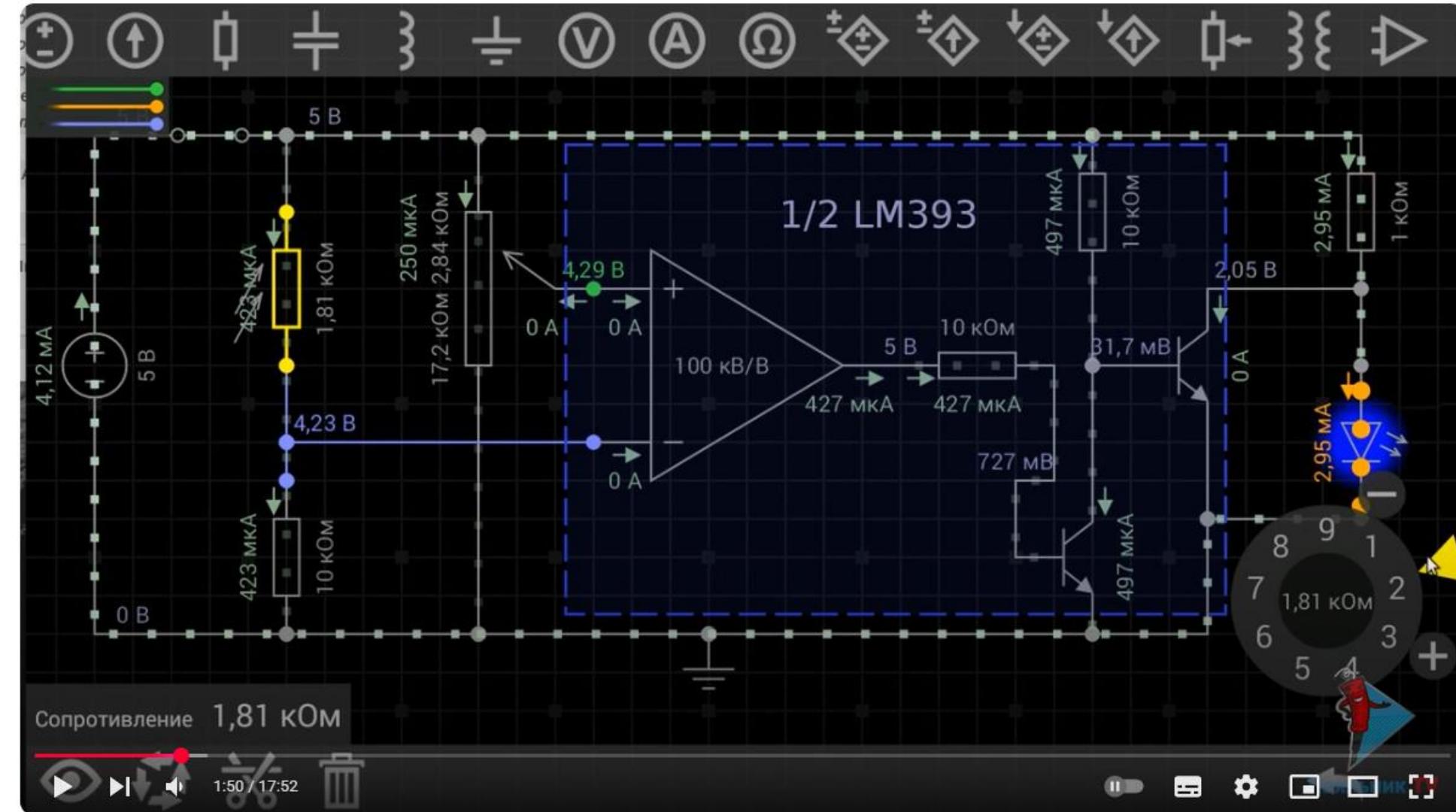
Компаратор на операционном усилителе (2017)
<https://www.youtube.com/watch?v=Gi2IxPJl2iw>



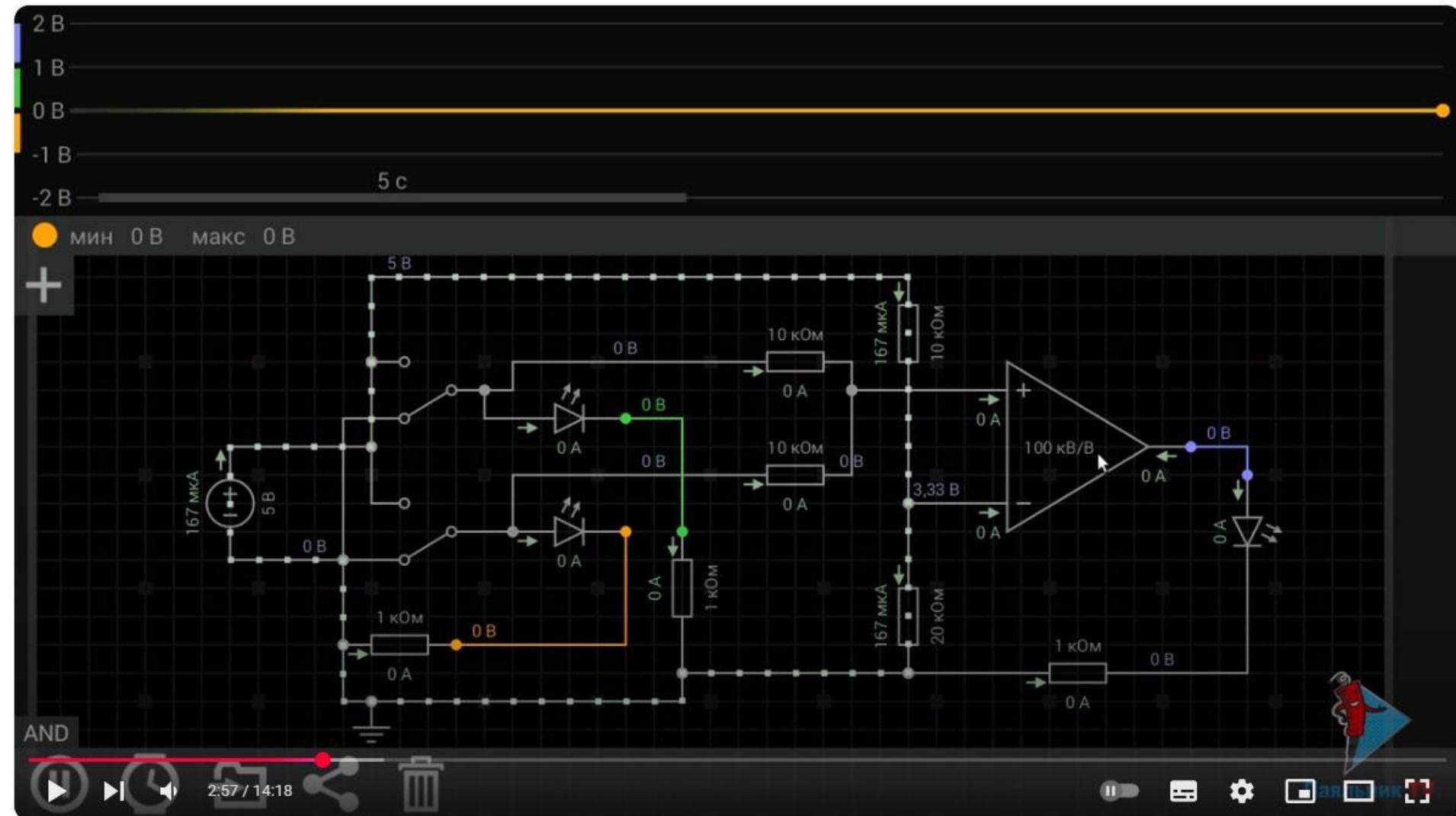
0018. Операционный усилитель. Компаратор. (2018)
<https://www.youtube.com/watch?v=MLd5r6RFe7o>



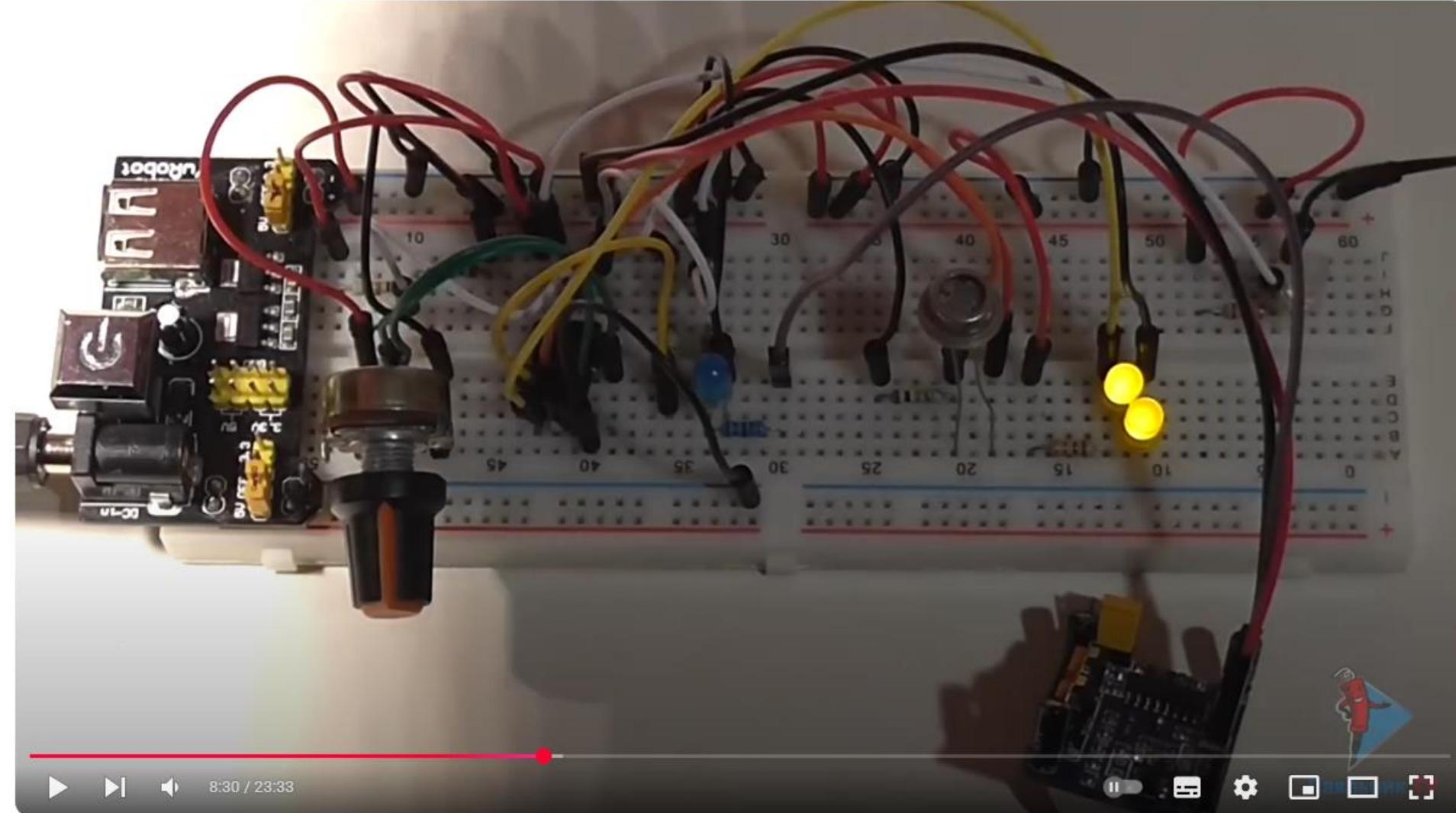
Компараторы. Часть 1 - введение (2015)
<https://www.youtube.com/watch?v=Nof9YFDaaJw>



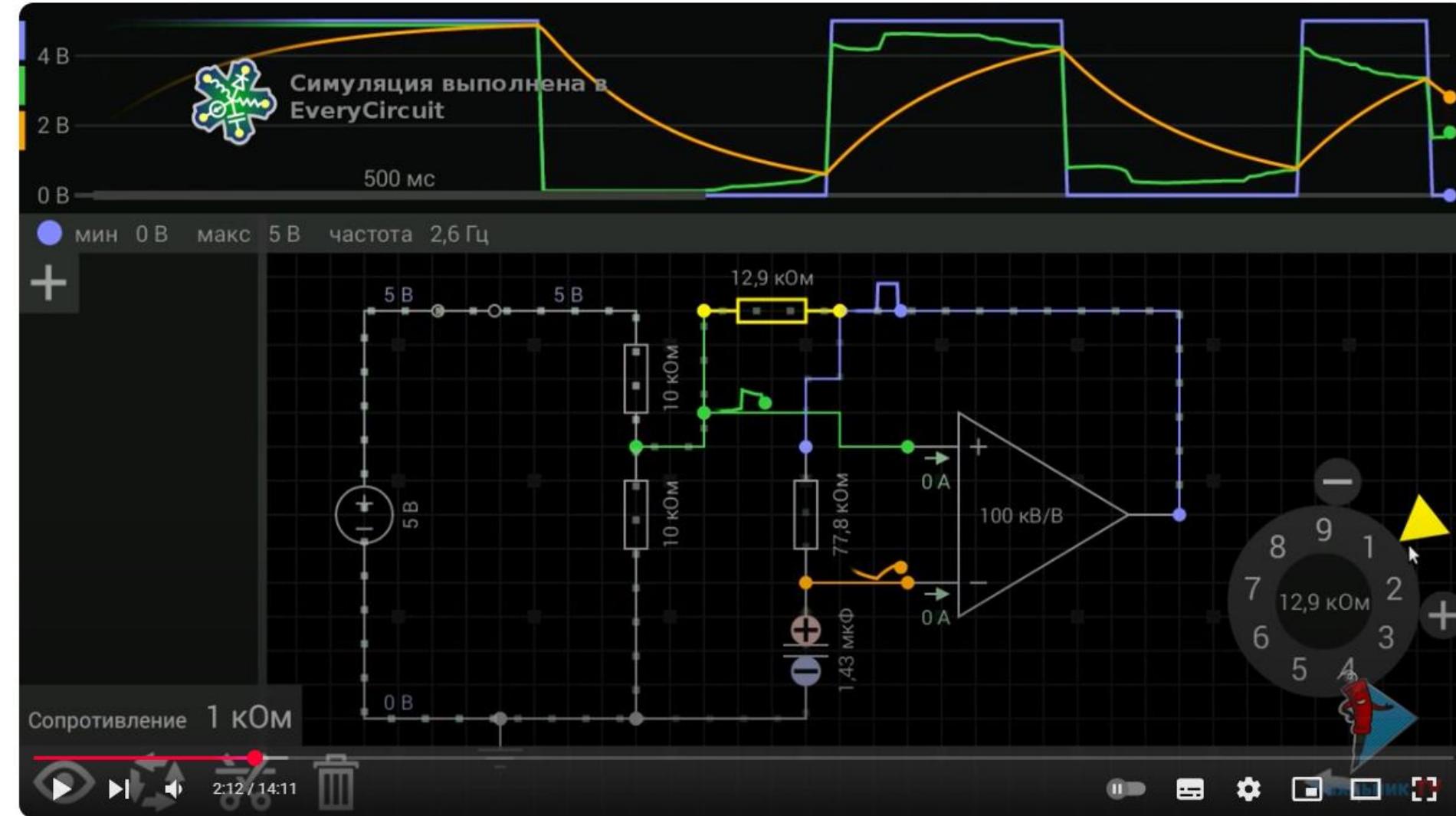
Компараторы. Часть 2 - фонарик, реагирующий на свет (2015)
<https://www.youtube.com/watch?v=nKljycWq2Xs>



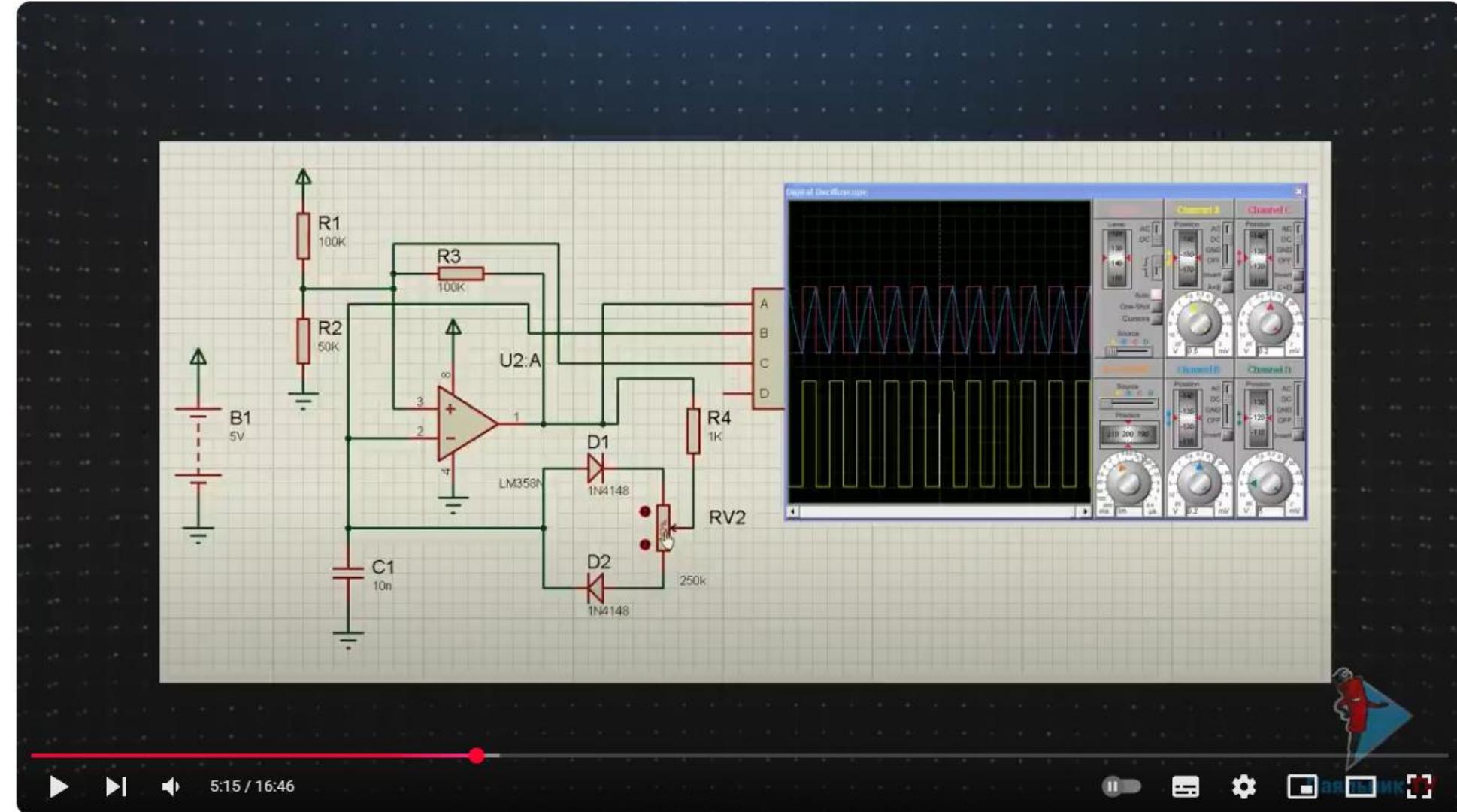
Компараторы. Часть 3 - логические элементы на компараторах (2015)
<https://www.youtube.com/watch?v=juuRsZPKrpw>



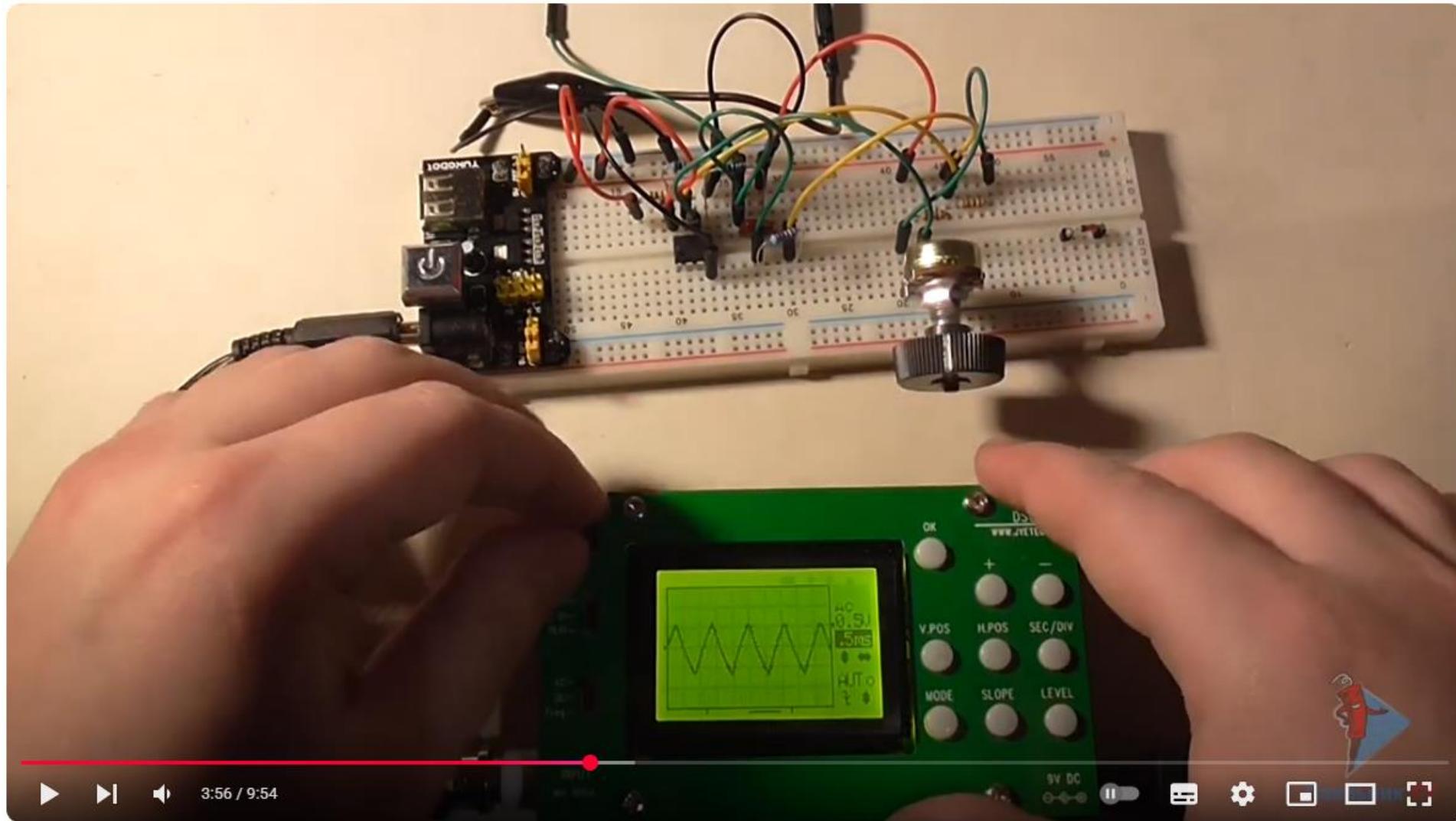
Компараторы. Часть 4 - Ночник на логике (2015)
<https://www.youtube.com/watch?v=8Fea9LZiMAg>



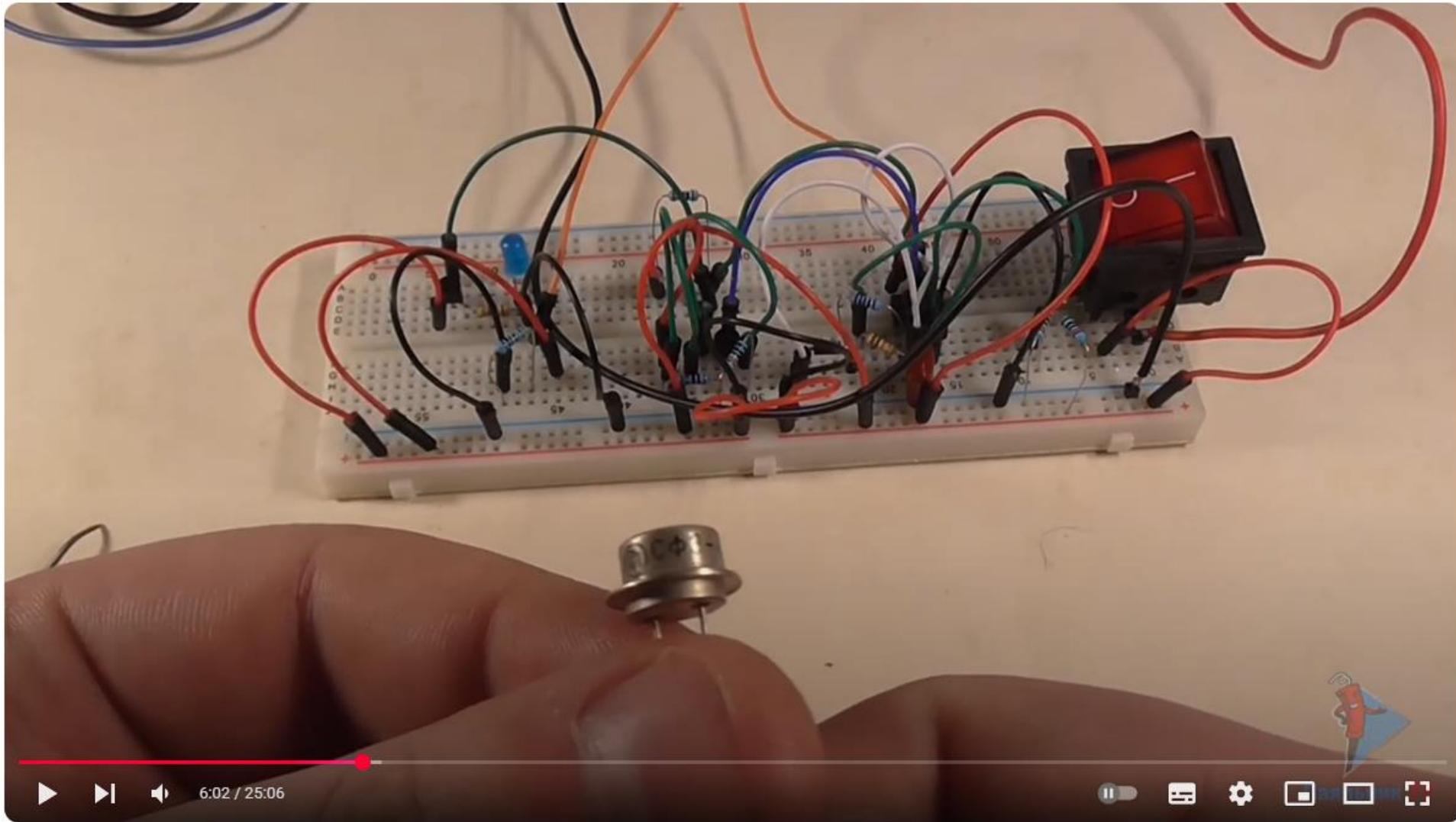
Компараторы. Часть 5 - Генератор прямоугольных импульсов (2016)
<https://www.youtube.com/watch?v=UlIpKrp8G3Dk>



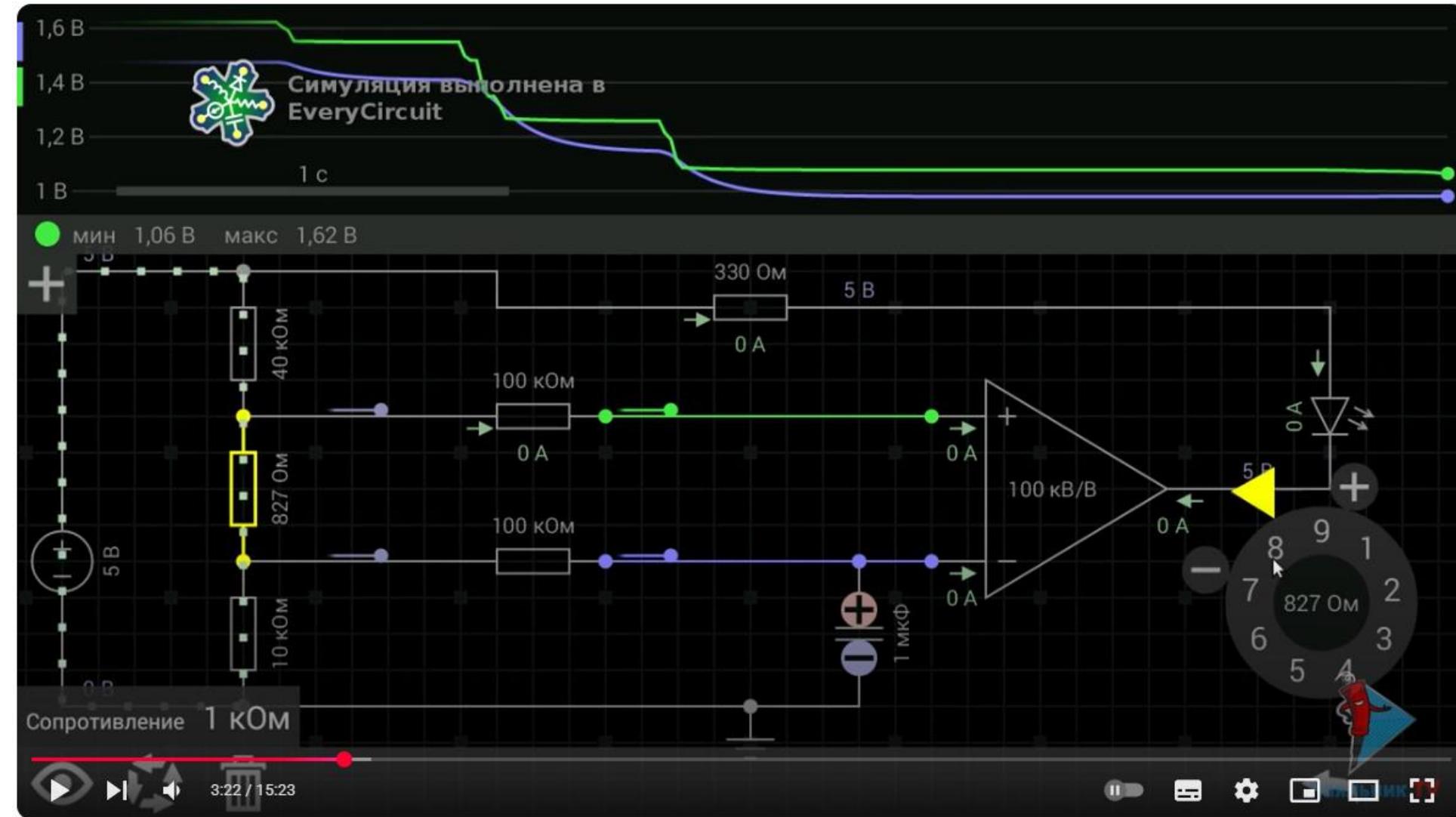
Компараторы. Часть 6 - ШИМ генератор из одного компаратора (2016)
<https://www.youtube.com/watch?v=D972Rxh50C4>



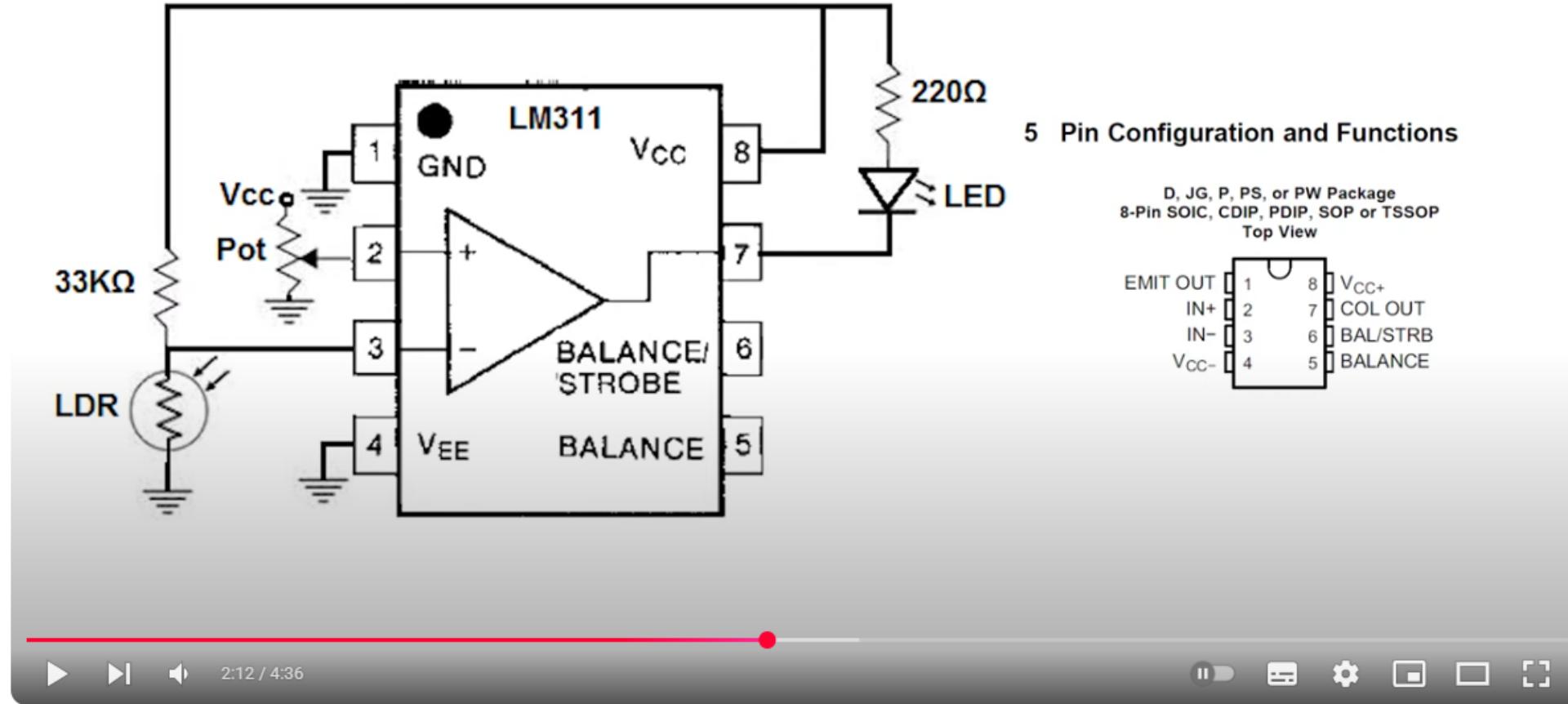
Компараторы. Часть 7 - Генератор треугольных импульсов (2016)
<https://www.youtube.com/watch?v=0x-or3q8bBU>



Компараторы. Часть 8 - ШИМ генератор, управляемый напряжением (2016)
<https://www.youtube.com/watch?v=RaGtsLdYbNM>



Компараторы. Часть 9 - Световой датчик движения (2016)
<https://www.youtube.com/watch?v=ndNhPk-a80s>



Включение света в темноте или как работает компаратор (2015)
<https://www.youtube.com/watch?v=qNskSCWQRzE>

ЦИФРОВЫЕ КОМПАРАТОРЫ

ЧАСТЬ 1

Комбинационные ЦУ

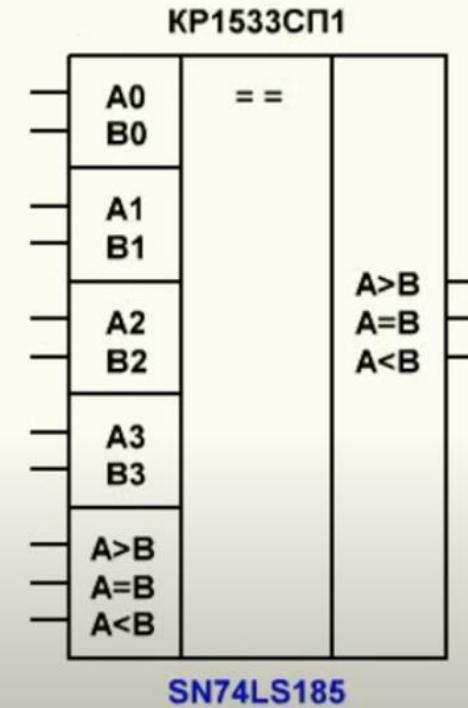
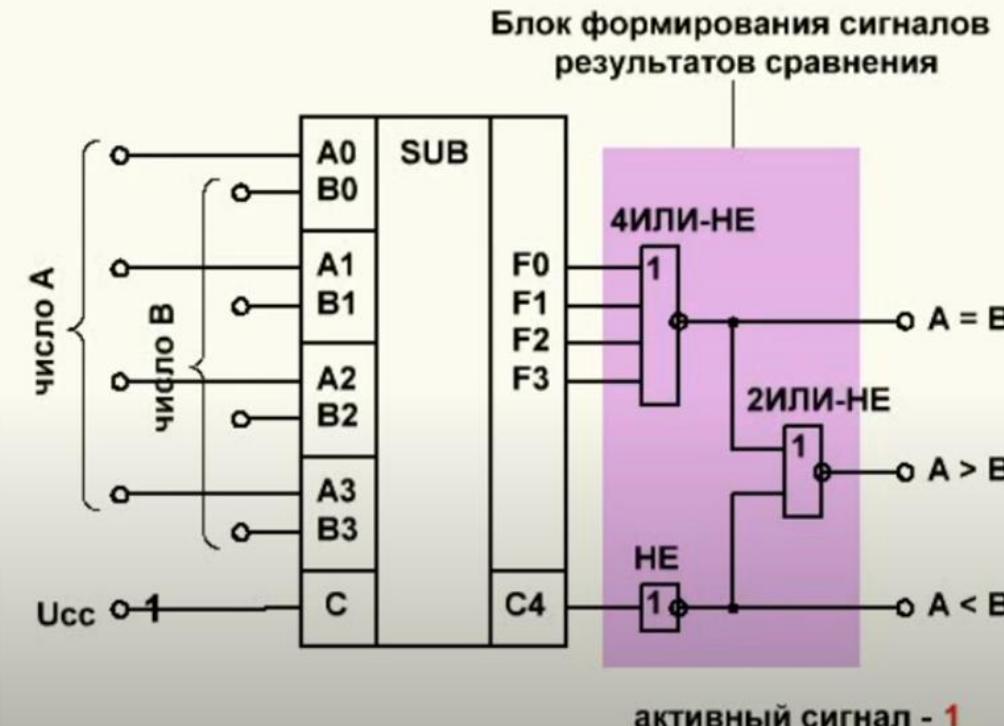
Белорусско-Российский университет, Кафедра «Программное обеспечение информационных технологий»

Вideoурок 3.37. Цифровые компараторы. Часть 1 (2023)
<https://www.youtube.com/watch?v=LUDm2UlwBJQ>

Компараторы

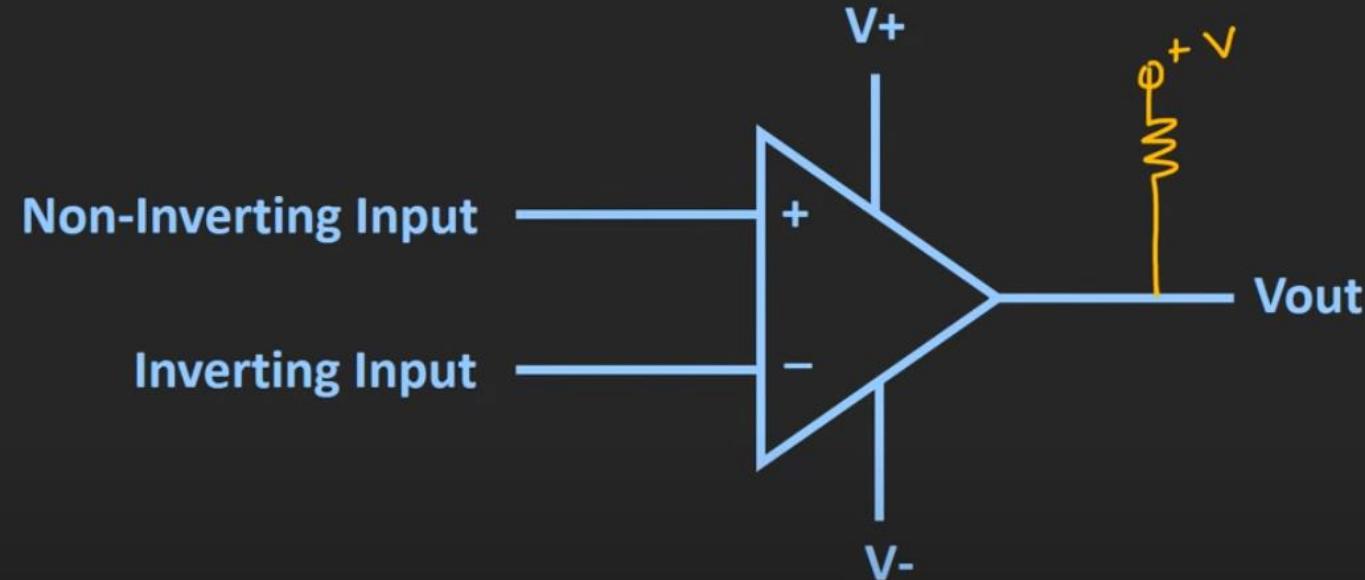
compare - сравнивать

4-разрядный компаратор



Вideoурок 3.38. Цифровые компараторы. Часть 2 (2023)
<https://www.youtube.com/watch?v=iuUMv1z8isE>

Comparator



▶ ▶ 🔍 3:40 / 12:37

ALL ABOUT ELECTRONICS



Comparitor Explained (Inverting Comparitor, Non-Inverting Comparitor and Window Comparitor) (2018)
Описание компаратора (Инвертирующий компаратор, неинвертирующий компаратор и оконный компаратор) (2018)
<https://www.youtube.com/watch?v=k9zQjEaktfk>



Comparator - Operational Amplifier | Basic Circuits #16 | Electronics Tutorials (2022)
Компаратор — операционный усилитель | Базовые схемы №16 | Учебники по электронике (2022)
<https://www.youtube.com/watch?v=zNAbcUSptWE>

Comparators: Basic Overview and Common Applications

TI Precision Labs – Op Amps

Presented by Chi Nguyen

Prepared by Thomas Kuehl, Paul Grohe, Chi Nguyen



An Overview on Comparators (2022)
Обзор компараторов (2022)

<https://www.youtube.com/watch?v=OHK9Hm3KMcA>

Comparator Applications 1

TIPL 2101
TI Precision Labs – Op Amps

Presented by Ian Williams

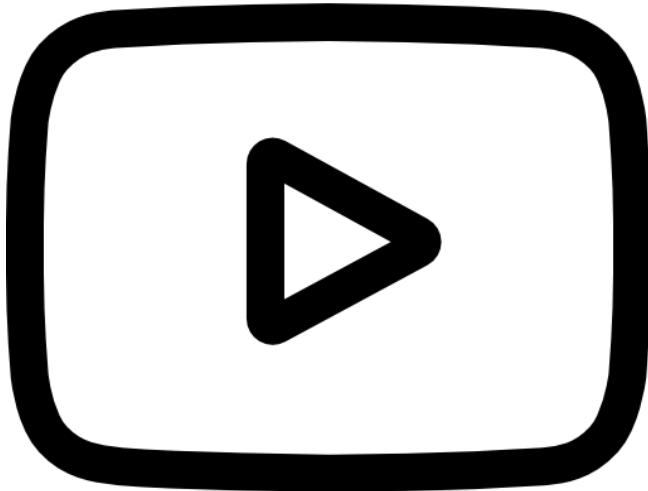
Prepared by Thomas Kuehl and Ian Williams

Смотреть (k)

0:01 / 17:29 • Intro >

TEXAS INSTRUMENTS

Introduction to comparator functions (2019)
Введение в функции компаратора (2019)
<https://www.youtube.com/watch?v=JmHyIoQYMVE>

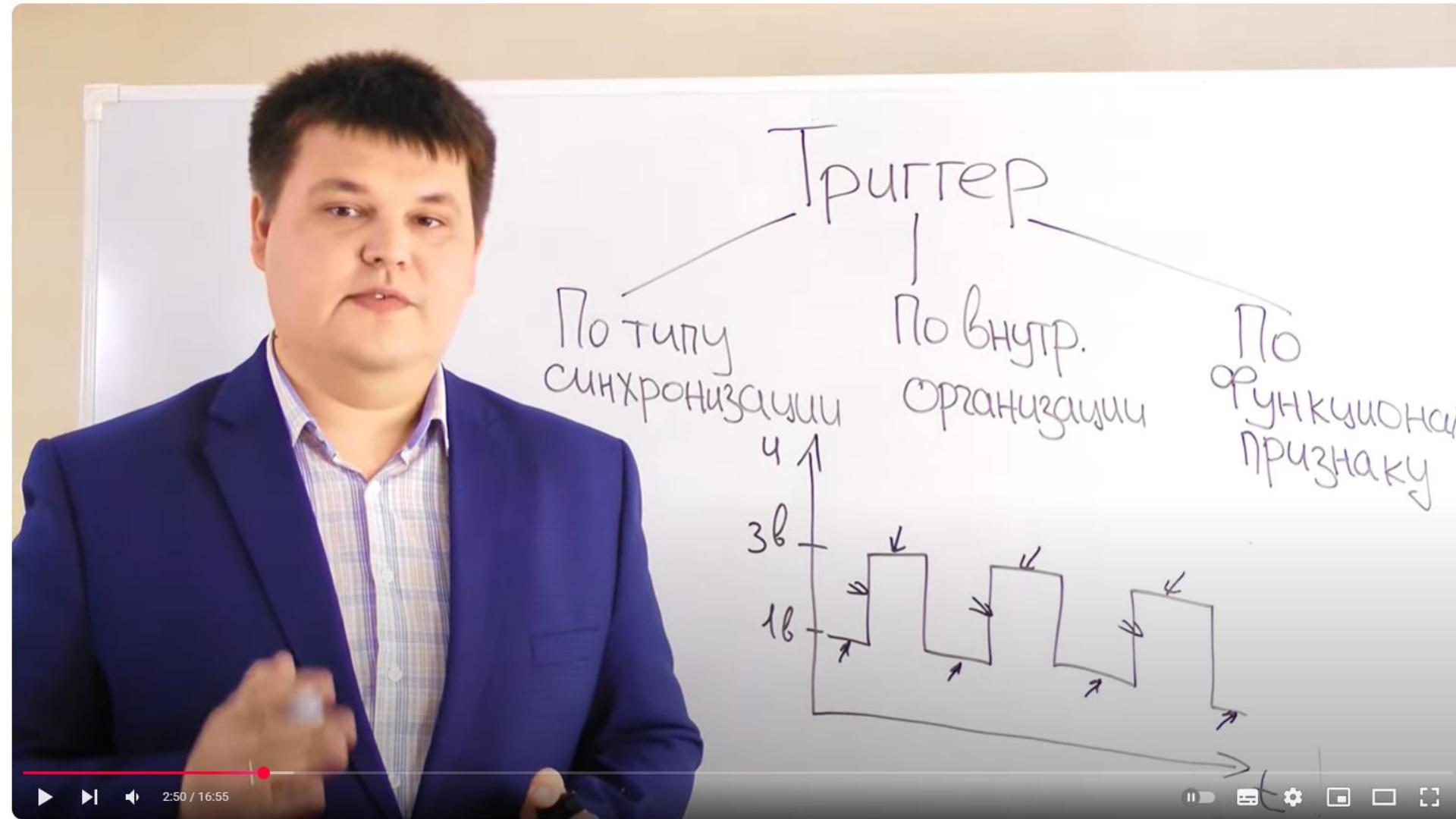


Триггеры.

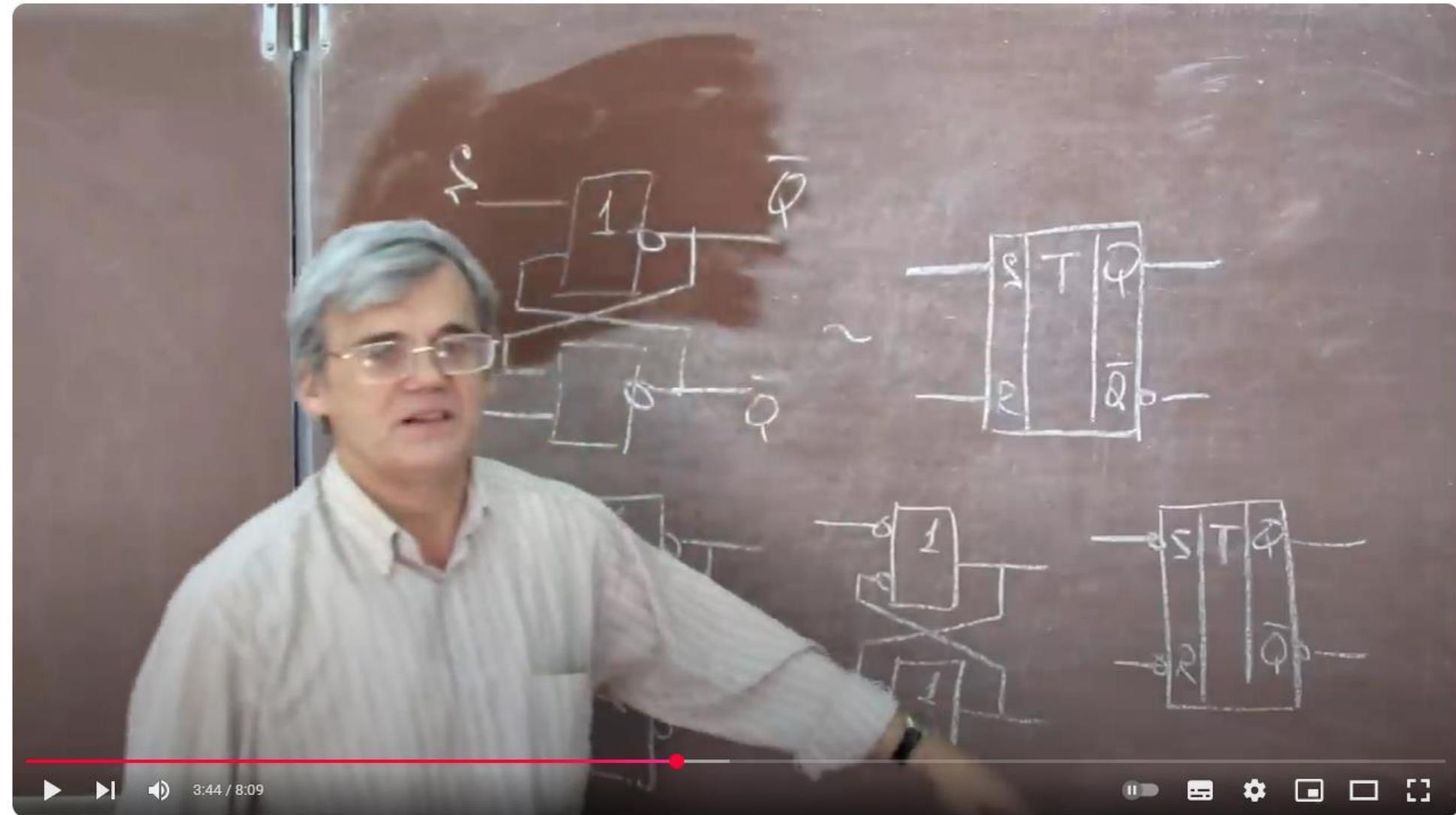
Асинхронные триггеры (на элементах И-НЕ и ИЛИ-НЕ, RS-триггеры)

Синхронные триггеры (на элементах И-НЕ и ИЛИ-НЕ, D-триггеры, JK-триггеры, T-триггеры)



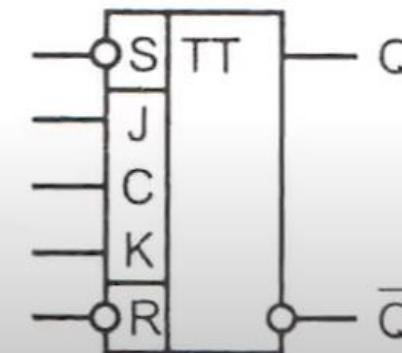
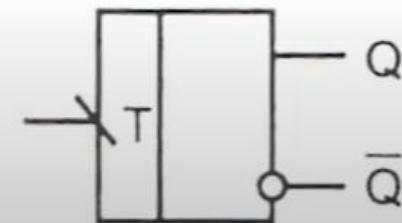
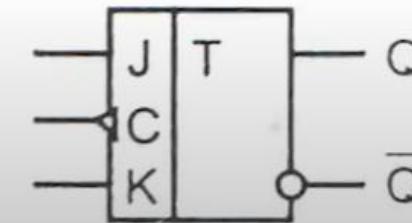
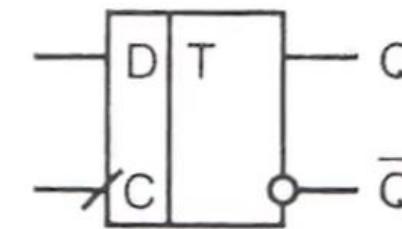
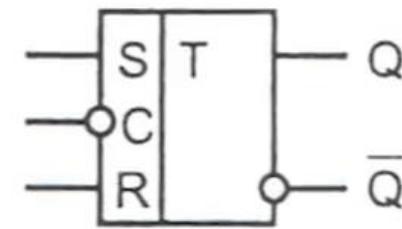
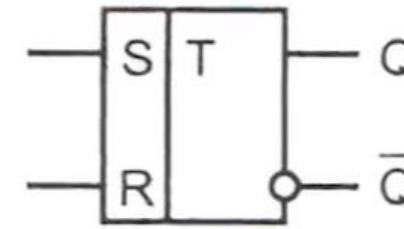


Последовательные логические устройства. Триггеры (RS, D, JK, T). Принцип работы, типовые схемы (2021) <https://www.youtube.com/watch?v=7QLQplw5EKE>

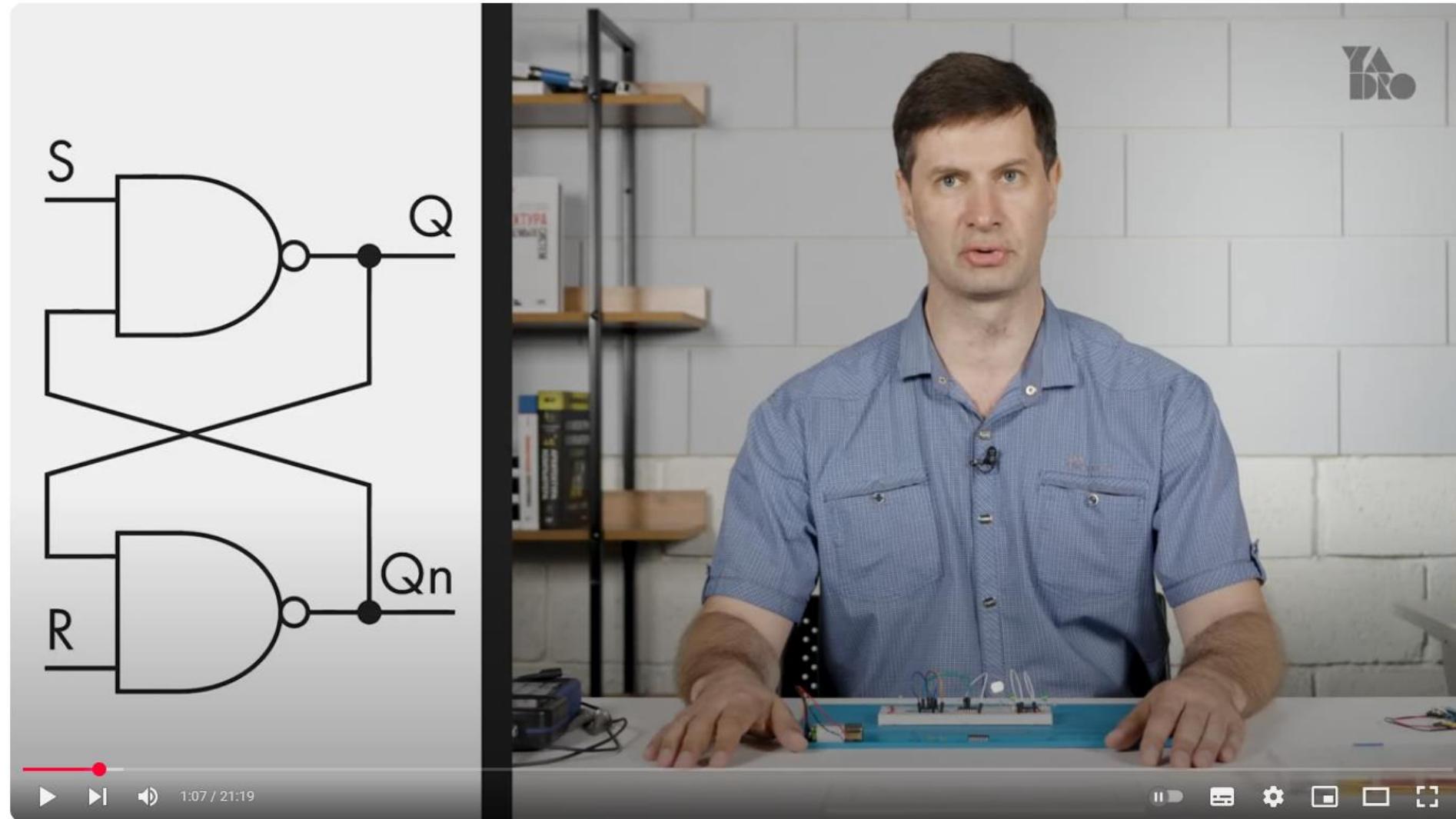


Лекция 101. Триггеры в терминах сигналов (2013)

<https://www.youtube.com/watch?v=w7qLb1AcfZQ>



Что такое триггер в электронике и не только (2020)
<https://www.youtube.com/watch?v=RKgfCaxeQwU>



Учимся хранить информацию: триггеры на макетной плате | Схемотехника для начинающих №8 (2024) <https://www.youtube.com/watch?v=jrz5Ssvzebs>

Элементы систем автоматики

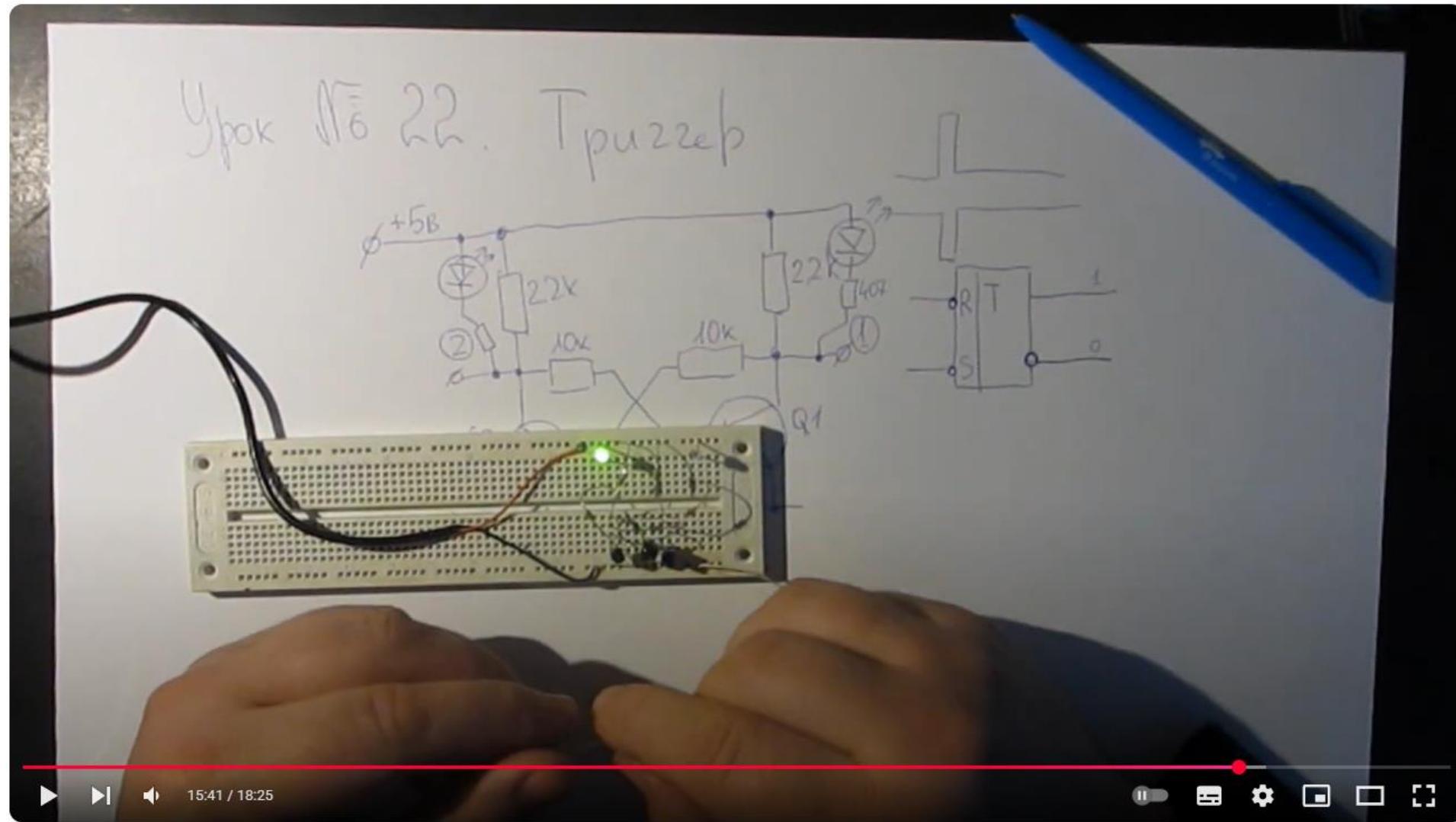
Триггеры.

1. Классификация

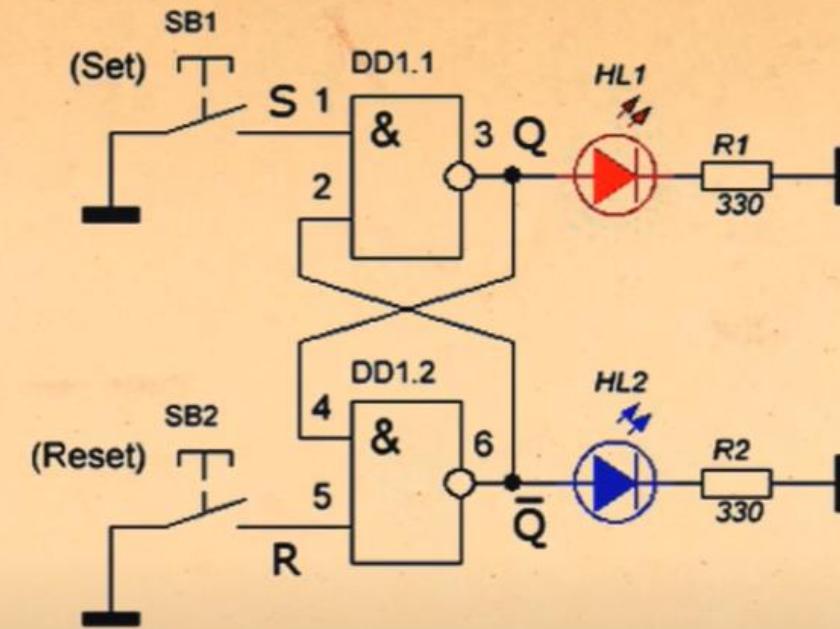
И.В. Музылева
К.т.н., доцент



Триггеры. 1. Классификация (2023)
https://www.youtube.com/watch?v=tOsdWWH_kmo



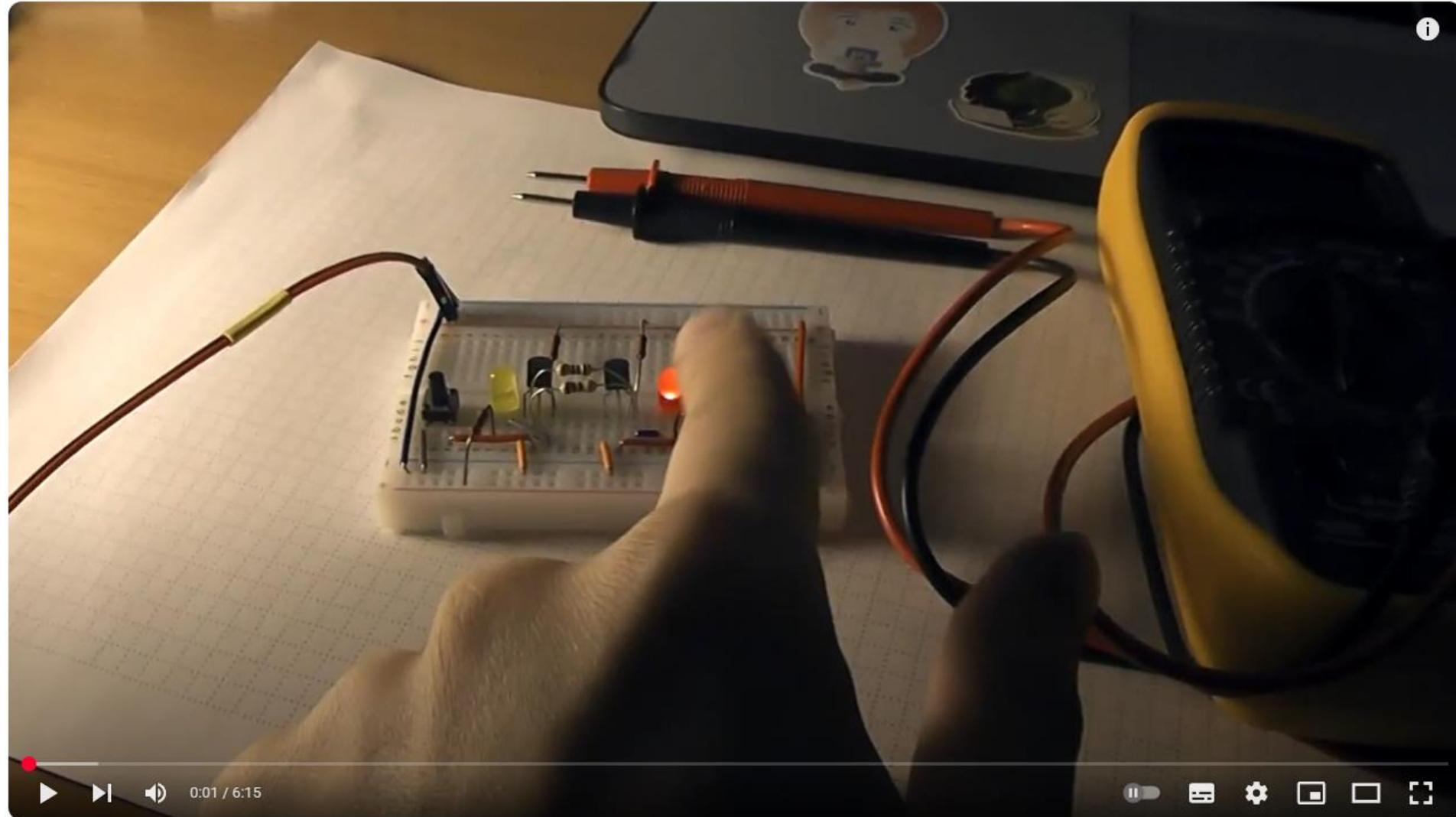
Урок №22. Триггер. (2016)
<https://www.youtube.com/watch?v=5Tce5yRX0pQ>



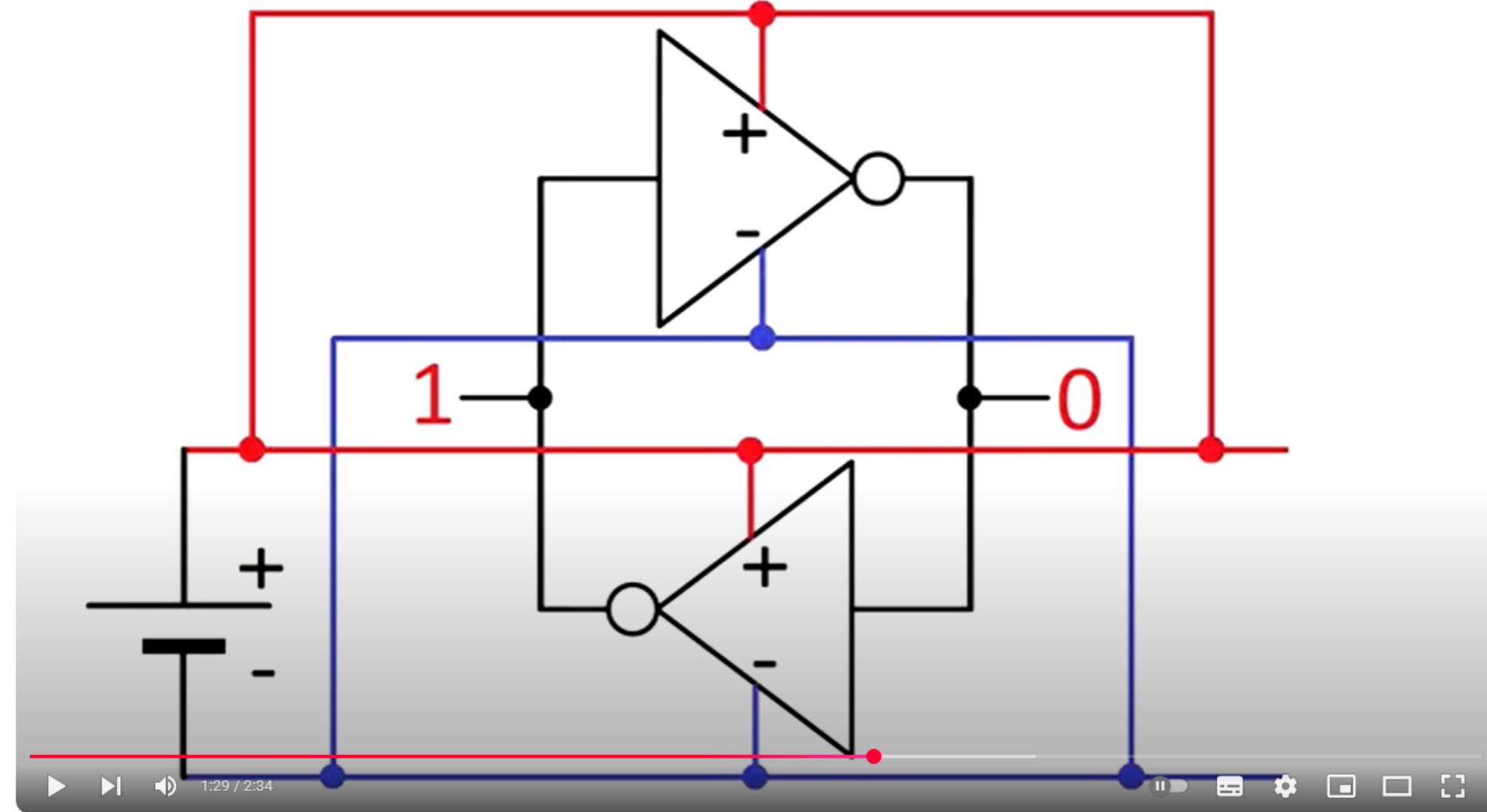
Триггер (англ. trigger) - защёлка, спусковой крючок



Урок №51. Триггер осциллографа (2019)
<https://www.youtube.com/watch?v=uPGDtRs2PQc>



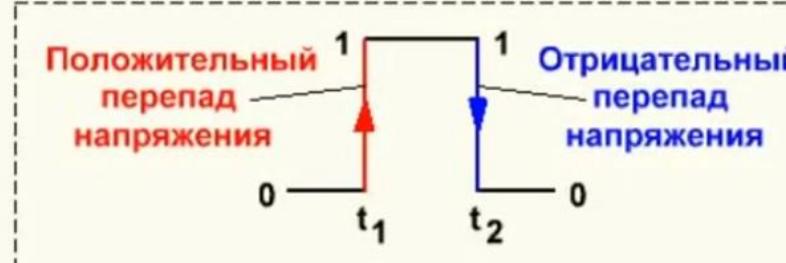
[электроника для начинающих] Триггер на транзисторах! Принцип работы и сборка. (2018) <https://www.youtube.com/watch?v=raWjVYSIg50>



Простейший триггер. Видео №1 к 6 главе книги. (2021)
<https://www.youtube.com/watch?v=Ous6cCk-mrg>

Управление интегральными триггерами

2. Динамические входы \Rightarrow синхронизирующие входы C (CLK)



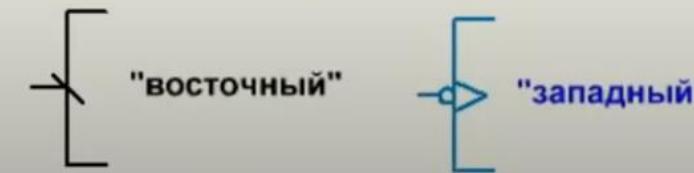
1. Прямые динамические.



Активный сигнал - положительный перепад напряжения

\nearrow , \uparrow , P или POS (positive-going edge)

2. Инверсные динамические.



Активный сигнал - отрицательный перепад напряжения

\nwarrow , \uparrow , N или NEG (negative-going edge)



Вideoурок 4.3. Способы управления интегральными триггерами. (2024)
<https://www.youtube.com/watch?v=iZB7g0UdwcE>

Методика Информатики ИРН | +

Файл С:/Users/Danov/Documents/Информатика/ИИИ%20для%20МРИ/Методика%20Информатики%20ИРН%20для%20ИИИ.pdf

Нарисовать Прочесть вслух

68

Практическая работа № 9

Построение логических схем триггеров и анализ их работы

1. Постановка задачи

Используя программу построения и моделирования логических схем Logisim [2] построить следующие схемы и научиться управлять ими:

1. Синхронный фронтовой RS-триггер с асинхронным управлением.
2. Синхронный фронтовой D-триггер с асинхронным управлением (на основе схемы 1).
3. Синхронный фронтовой T-триггер с асинхронным управлением (на основе схемы 1).
4. Синхронный фронтовой JK-триггер с асинхронным управлением.

Все схемы демонстрируются преподавателю в аудитории, с проверкой понимания их функционирования студентом.

После защиты какой-либо её части работы в тетради ставится соответствующая пометка. После защиты всех частей выставляется отметка о сдаче работы.

2. Последовательность выполнения работы

1. Построить в программе схему синхронного фронтового RS-триггера с асинхронным управлением.
2. Скопировав схему 1, построить на её основе в программе схему синхронного фронтового D-триггера с асинхронным управлением.
3. Скопировав схему 1, построить на её основе в программе схему синхронного фронтового T-триггера с асинхронным управлением.
4. Построить в программе схему синхронного фронтового JK-триггера с асинхронным управлением.
5. Защитить выполненные работы преподавателю, ответив на вопросы (или решив задачи) по функционированию схем.

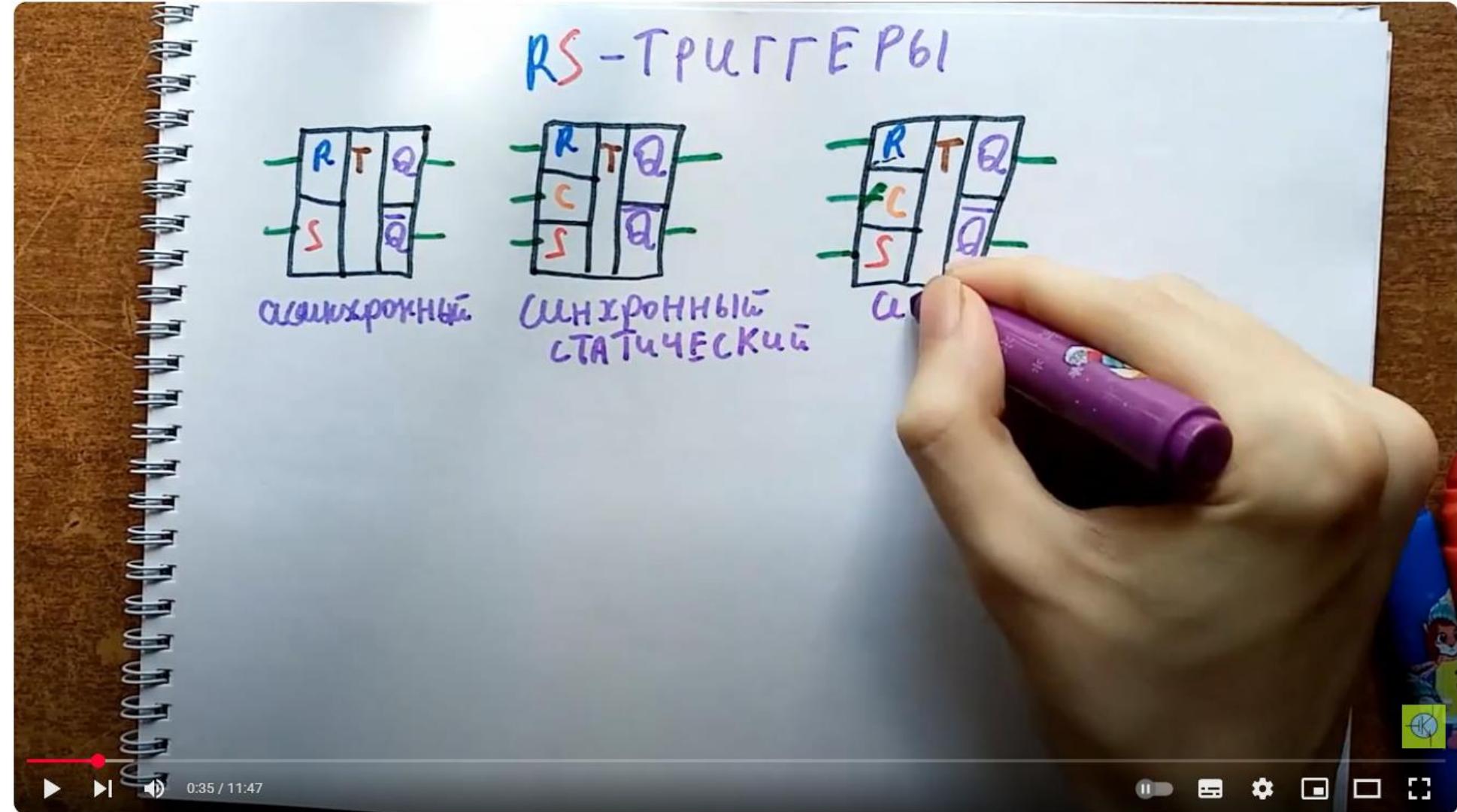
Схемы триггеров, способы их построения и описание режимов функционирования

0:11 / 7:36

https://www.youtube.com/watch?v=U2b5EAcFtFc

Информатика - Построение триггеров (2023)

<https://www.youtube.com/watch?v=U2b5EAcFtFc>



Компьютер на транзисторах своими руками №12 Т,D,JK - триггеры (2020)
<https://www.youtube.com/watch?v=i6y3iFP5K10>

Вычислительные системы, сети и телекоммуникации. 2 семестр (ПИБ-18И1) (Бречка Д.М.) Лабораторная работа 5. Триггеры — Mozilla Firefox

Portfolio | Портфолио | MeisterTask | Мой диск - Google Диск | ТЭн1 - Google Документы | Курс: Вычислительные | Вычислительные системы | Logisim: загрузка | Modeling of digital device | СиБАДИ | Trello | + | ... | Другие закладки

Job Home Shop

https://portal.sibadi.org/mod/assign/view.php?id=97803

(Бречка Д.М.)

В начало / Мои курсы / Вычислительные системы, сети и телекоммуникации. 2 семестр (ПИБ-18И1) (Бречка Д.М.) / Тема 2. Цифровая схемотехника / Лабораторная работа 5. Триггеры

Лабораторная работа 5. Триггеры

Задание:

Задание 1. Провести функциональное моделирование синхронного SR-триггера на базе элементов И-НЕ.
Задание 2. Провести функциональное моделирование SR-триггера из библиотеки LogiSim.
Задание 3. Провести функциональное моделирование D-триггера из библиотеки LogiSim.
Задание 4. Провести функциональное моделирование JK-триггера из библиотеки LogiSim.

Проект Logisim.

Теоретические сведения по моделированию простейших комбинационных устройств (пп. 2.4).

В качестве ответа на задание загрузите отчет о проделанной работе.

Изолированные группы: Все участники

Резюме оценивания

Скрыто от студентов	Нет
Участники	27
Черновик	0
Ответы	0
Требуют оценки	0

Специальные возможности

Специальные возможности

ЗАПУСТИТЬ АТВАР
(сегодня?)

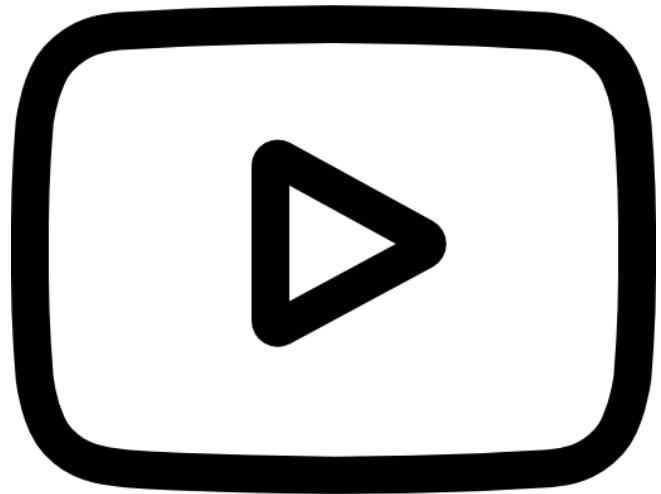
Статистика

Подробнее
Посетителей сегодня : 3

0:01 / 1:49:22

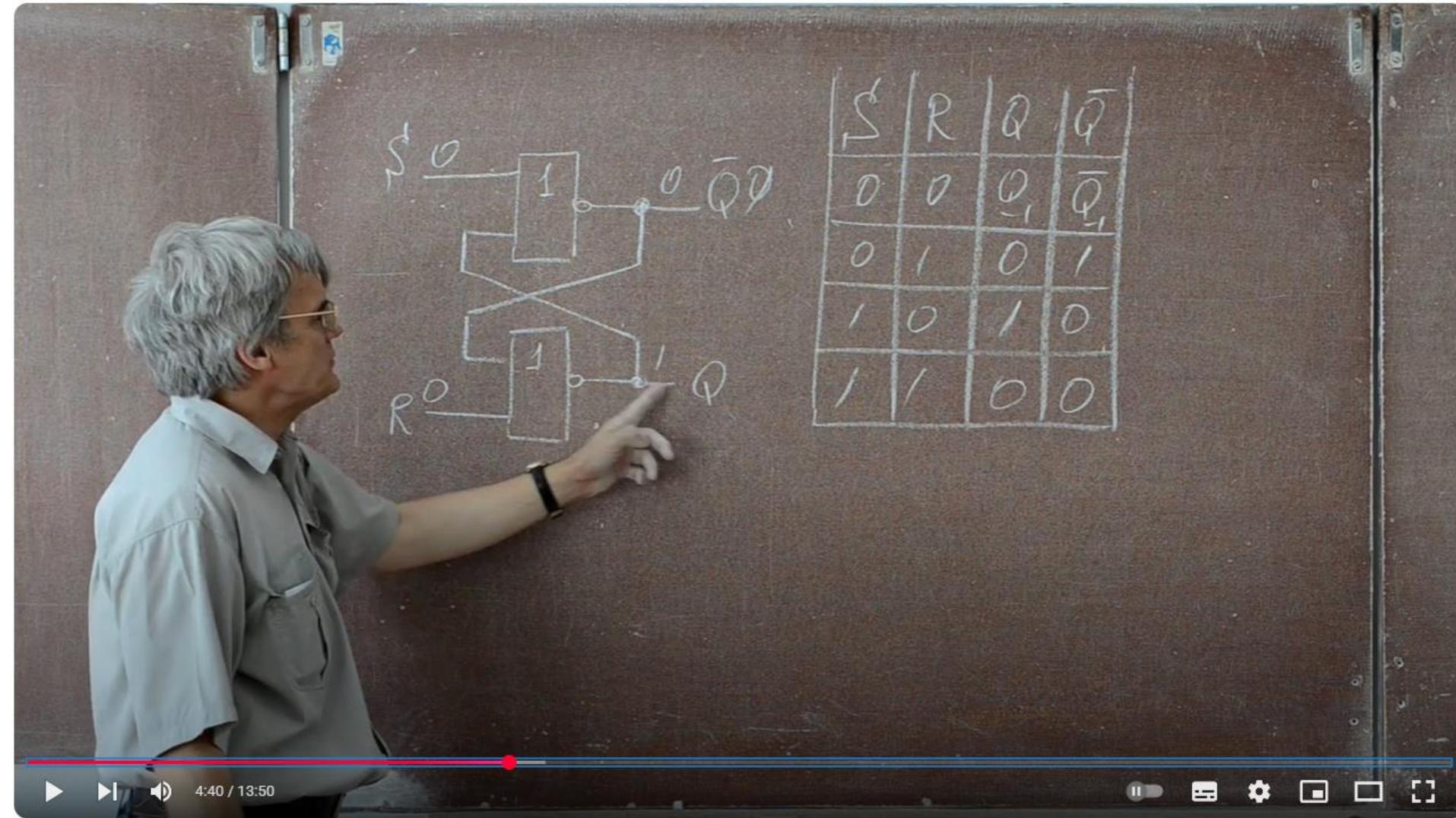
Лабораторная работа № 5. Триггеры (2021)

<https://www.youtube.com/watch?v=Sr6sYGo7nJc>



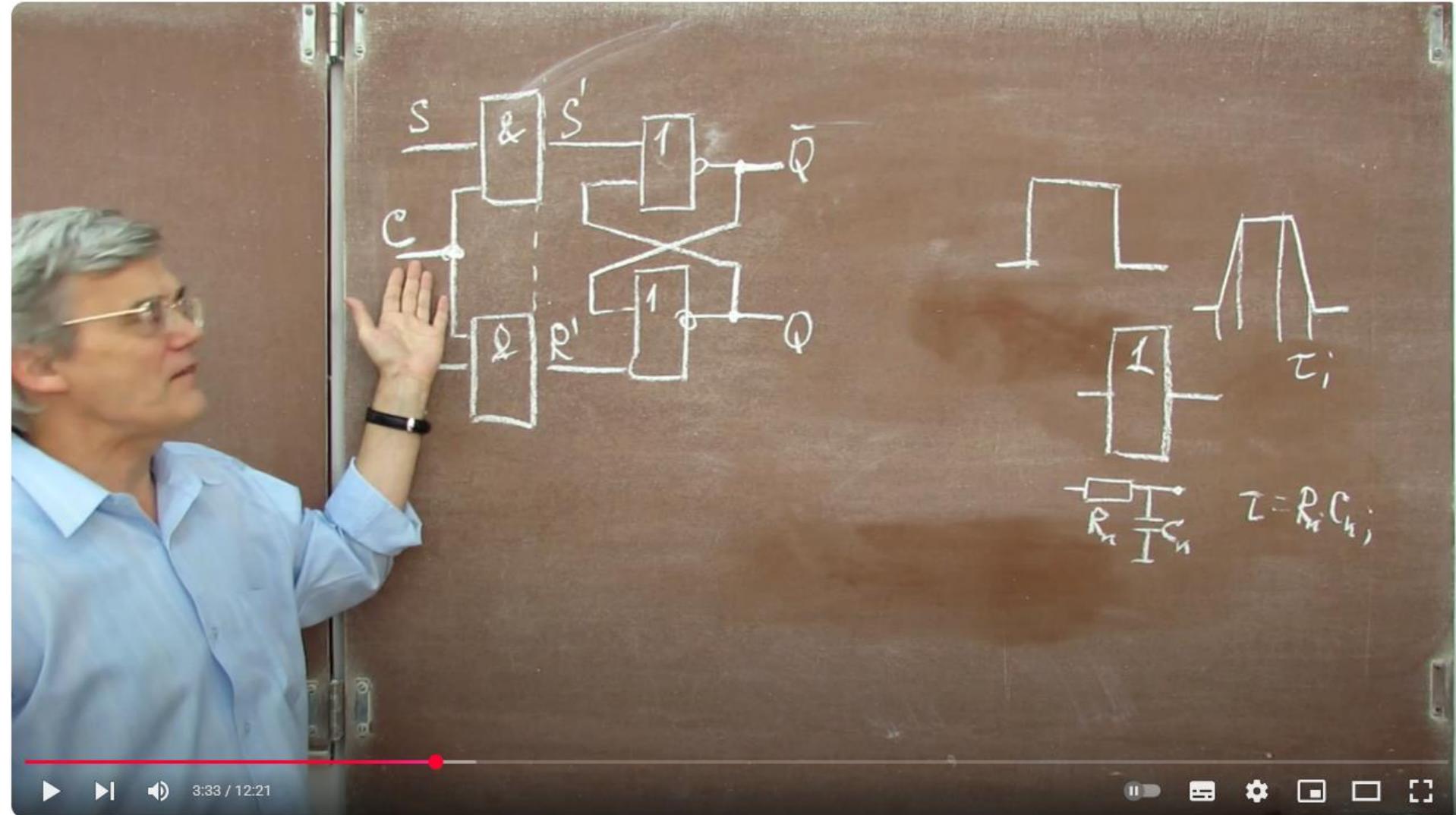
RS-триггер





Лекция 97. RS-Триггер (2013)

<https://www.youtube.com/watch?v=g1PHEXU5HeY>



Лекция 99. Синхронный RS-триггер (2013)

<https://www.youtube.com/watch?v=vNvYWVht6pM>



Логическая реализация ячейки памяти. RS триггер (2018)
<https://www.youtube.com/watch?v=C9hF9YmSGUg>

63

10. Практическая работа №10: изучение работы триггеров

1. Постановка задачи
Изучить на практике работу триггеров, показанных на рисунках ниже (рис. 37-45).

1.1. Одноступенчатый асинхронный RS-триггер на элементах И-НЕ
Таблица переходов триггера (табл. 6) и его функциональная схема.

Таблица 6

\bar{S}	\bar{R}	$Q(t + 1)$	$\bar{Q}(t + 1)$	Режим
0	0	1	1	Запрещенная комбинация
0	1	1	0	Установка 1
1	0	0	1	Установка 0
1	1	$Q(t)$	$\bar{Q}(t)$	Хранение

not S \bar{S} \bar{R} Q \bar{Q}

not R \bar{R}

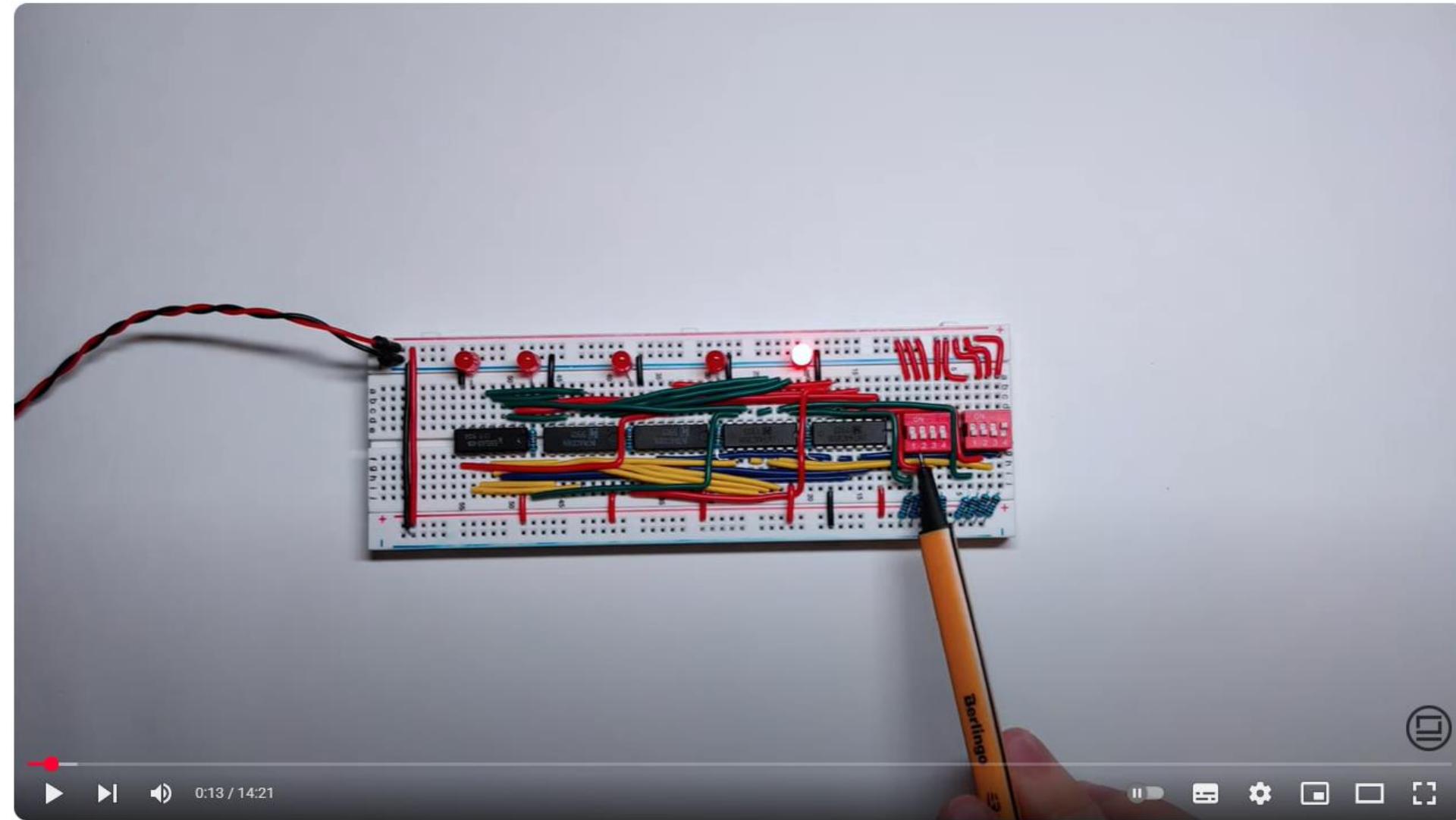
Рис.37 Одноступенчатый асинхронный RS-триггер на элементах И-НЕ

1.2. Одноступенчатый асинхронный RS-триггер на элементах ИЛИ-НЕ
Таблица переходов триггера (табл. 7) и его функциональная схема.

Таблица 7

S	R	$Q(t + 1)$	$\bar{Q}(t + 1)$	Режим
0	0	$Q(t)$	$\bar{Q}(t)$	Хранение
0	1	0	1	Установка 0
1	0	1	0	Установка 1
1	1	0	0	Запрещенная комбинация

Информатика, 10 практическая работа - Триггеры (2023)
<https://www.youtube.com/watch?v=kvpCofdRov0>



RS-триггер (2020)

<https://www.youtube.com/watch?v=SX2dVGgCWLk>

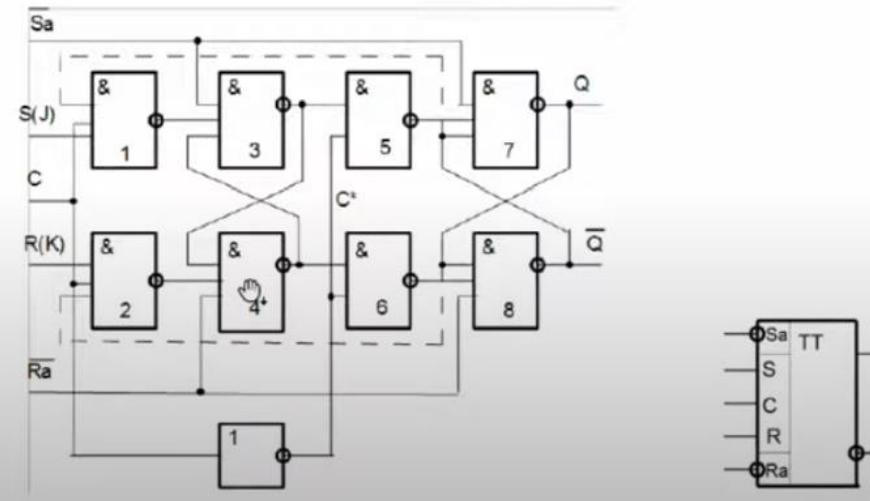
Двухступенчатый RS-триггер с запрещающим инвертором

При $C=1$ информация записывается в первую ступень. Для второй ступени $C^* = \bar{C} = 0$, перезапись информации из первой ступени во вторую ступень не происходит. При $C=0$, для второй ступени $C^*=1$, информация перезаписывается во вспомогательный триггер.

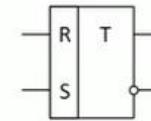
ЛЭ 1...4 – M-триггер (1-я ступень)

ЛЭ 5...8 – S-триггер (2-я ступень)

Инверсные входы \bar{S}_a и \bar{R}_a – это входы асинхронной установки триггера в 1 и 0 соответственно (независимо от сигнала синхронизации C). Эти входы напрямую управляют запоминающими ячейками первой (ЛЭ3 и ЛЭ4) и второй(ЛЭ7 и ЛЭ8) ступеней триггера.



Асинхронный статический
RS-триггер в базисе ИЛИ-НЕ



R	S	Q_{t+1}	\bar{Q}_{t+1}	сост.
0	0	Q_t	\bar{Q}_t	хр
0	1	1	0	уст
1	0	0	1	сбр
1	1	*	*	з.к.

Q_t	R	S	Q_{t+1}	\bar{Q}_{t+1}	сост.
0	0	0	0	1	хр 0
0	0	1	1	0	уст
0	1	0	0	1	сбр
0	1	1	*	*	з.к.
1	0	0	1	0	хр 1
1	0	1	1	0	уст
1	1	0	0	1	сбр
1	1	1	*	*	з.к.

Q_{t+1}	\bar{Q}_{t+1}	R	S
0	*	*	0
0	1	0	1
1	0	1	0

Q_t	00	01	11	10
Q_{t+1}	0	1	*	0
\bar{Q}_t	1	1	*	0

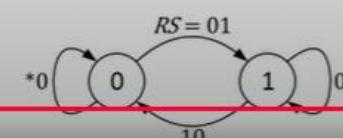
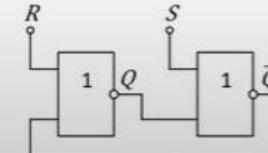
$$Q_{t+1} = Q_t \cdot \bar{R} \vee S$$

$$\bar{Q}_{t+1} = \overline{\bar{Q}_t \vee R} \vee S$$

Q_t	00	01	11	10
Q_{t+1}	0	1	*	0
\bar{Q}_t	1	1	*	0

$$Q_{t+1} = (Q_t \vee S) \cdot \bar{R}$$

$$Q_{t+1} = \overline{\bar{Q}_t \vee S \vee R}$$



Q_t	00	01	11	10
\bar{Q}_{t+1}	1	0	*	1
Q_t	0	0	*	1

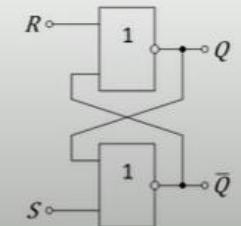
$$\bar{Q}_{t+1} = \bar{Q}_t \cdot \bar{S} \vee R$$

$$Q_{t+1} = \overline{\bar{Q}_t \vee S \vee R}$$

Q_t	00	01	11	10
Q_{t+1}	1	0	*	1
\bar{Q}_t	0	0	*	1

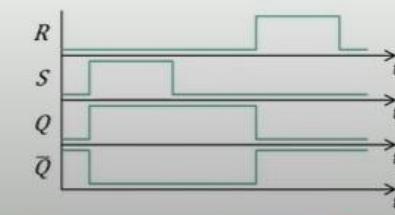
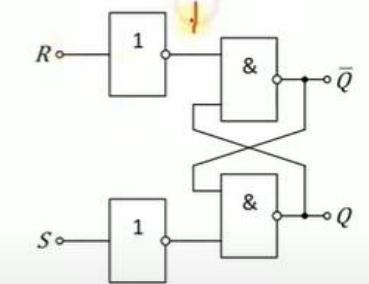
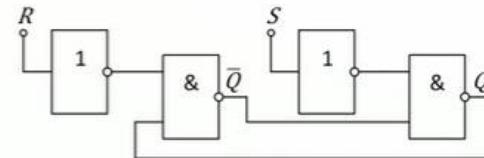
$$\bar{Q}_{t+1} = (\bar{Q}_t \vee R) \cdot \bar{S}$$

$$\bar{Q}_{t+1} = \overline{\bar{Q}_t \vee R \vee S}$$



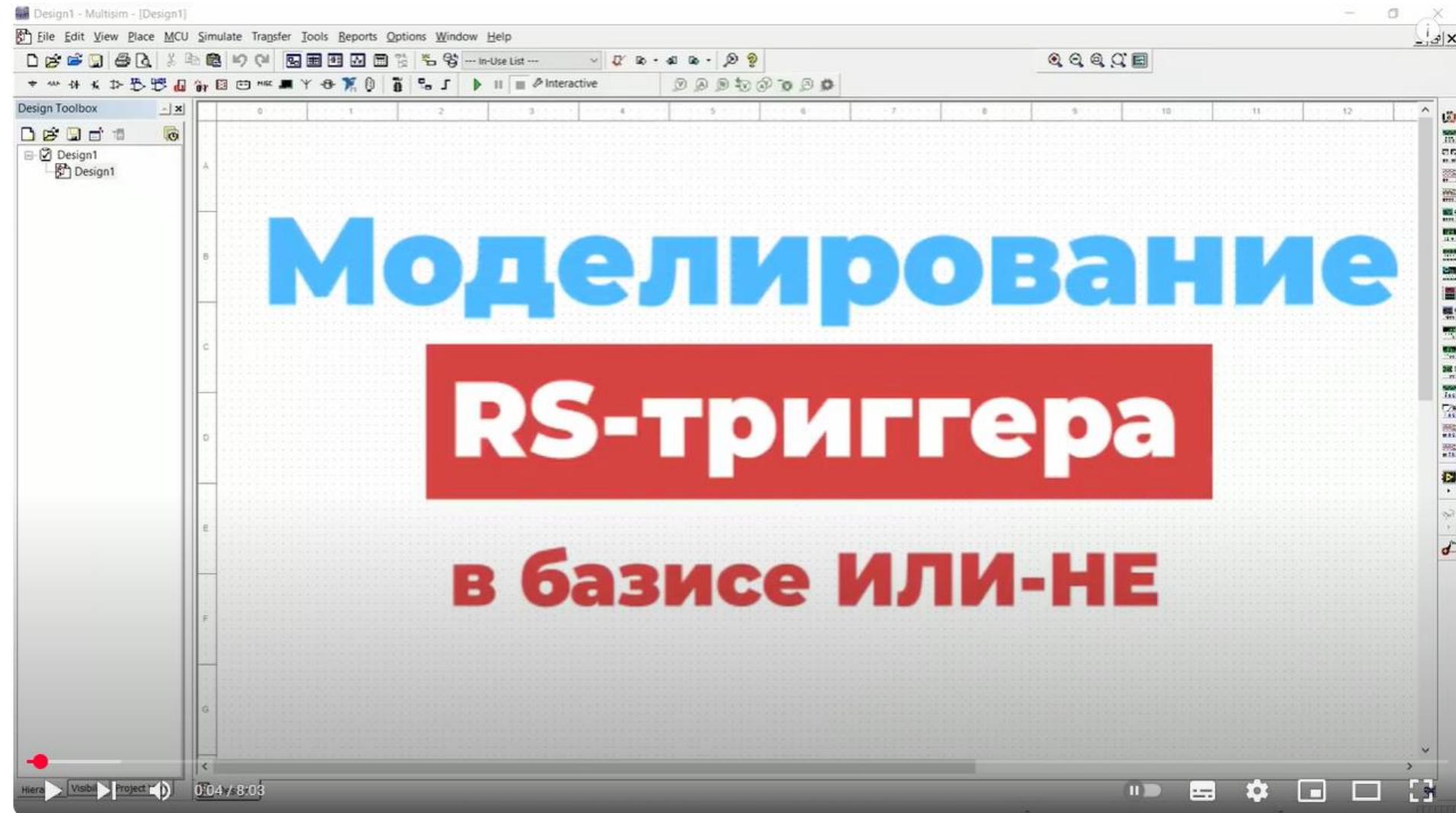
$$Q_{t+1} = Q_t \cdot \bar{R} \vee S$$

$$Q_{t+1} = \overline{\bar{Q}_t \cdot \bar{R}} \cdot S$$



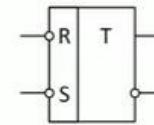
1.1. Асинхронный статический RS-триггер в базисе ИЛИ-НЕ (2022)

<https://www.youtube.com/watch?v=cyrdkJlOrBo>



1.2. Моделирование RS-триггера в Multisim (2022)
<https://www.youtube.com/watch?v=9SsWioqmp8Y>

Асинхронный статический
RS'-триггер в базисе И-НЕ



R'	S'	Q_{t+1}	\bar{Q}_{t+1}	сос
0	0	*	*	з. к.
0	1	0	1	сбр
1	0	1	0	уст
1	1	Q_t	\bar{Q}_t	хр

Q_t	R'	S'	Q_{t+1}	\bar{Q}_{t+1}	сос
0	0	0	*	*	з. к.
0	0	1	0	1	сбр
0	1	0	1	0	уст
0	1	1	0	1	хр
1	0	0	*	*	з. к.
1	0	1	0	1	сбр
1	1	0	1	0	уст
1	1	1	1	0	хр

Q_t	Q_{t+1}	R'	S'
0	0	*	1
0	1	1	0
1	0	0	1

▶ ▶ 🔊 7:26 / 8:02

Q_{t+1}	00	01	11	10	$R'S'$
Q_t	*	0	0	1	
	*	0	1	1	

$$Q_{t+1} = Q_t \cdot R' \vee S'$$

$$Q_{t+1} = \overline{\bar{Q}_t \cdot R'} \cdot S'$$

\bar{Q}_{t+1}	00	01	11	10	$R'S'$
Q_t	*	1	1	0	
	*	1	0	0	

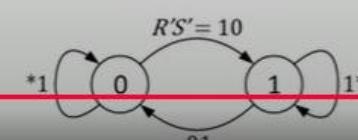
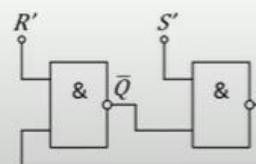
$$\bar{Q}_{t+1} = \bar{Q}_t \cdot S' \vee \bar{R}'$$

$$\bar{Q}_{t+1} = \overline{\bar{Q}_t \cdot S'} \cdot R'$$

Q_{t+1}	00	01	11	10	$R'S'$
Q_t	*	0	0	1	
	*	0	1	1	

$$Q_{t+1} = (Q_t \vee \bar{S}') \cdot R'$$

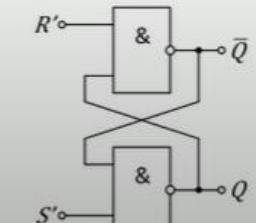
$$\bar{Q}_{t+1} = \overline{\bar{Q}_t \cdot S'} \cdot R'$$



\bar{Q}_{t+1}	00	01	11	10	$R'S'$
Q_t	*	1	1	0	
	*	1	0	0	

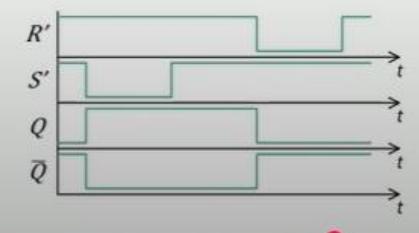
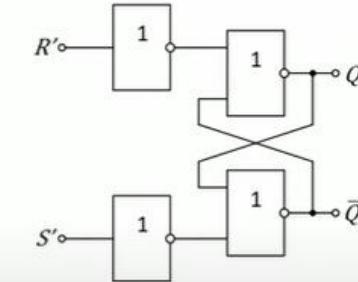
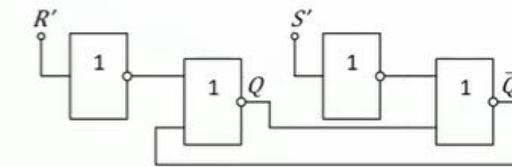
$$\bar{Q}_{t+1} = (\bar{Q}_t \vee \bar{R}') \cdot S'$$

$$Q_{t+1} = \overline{\bar{Q}_t \cdot R'} \cdot S'$$



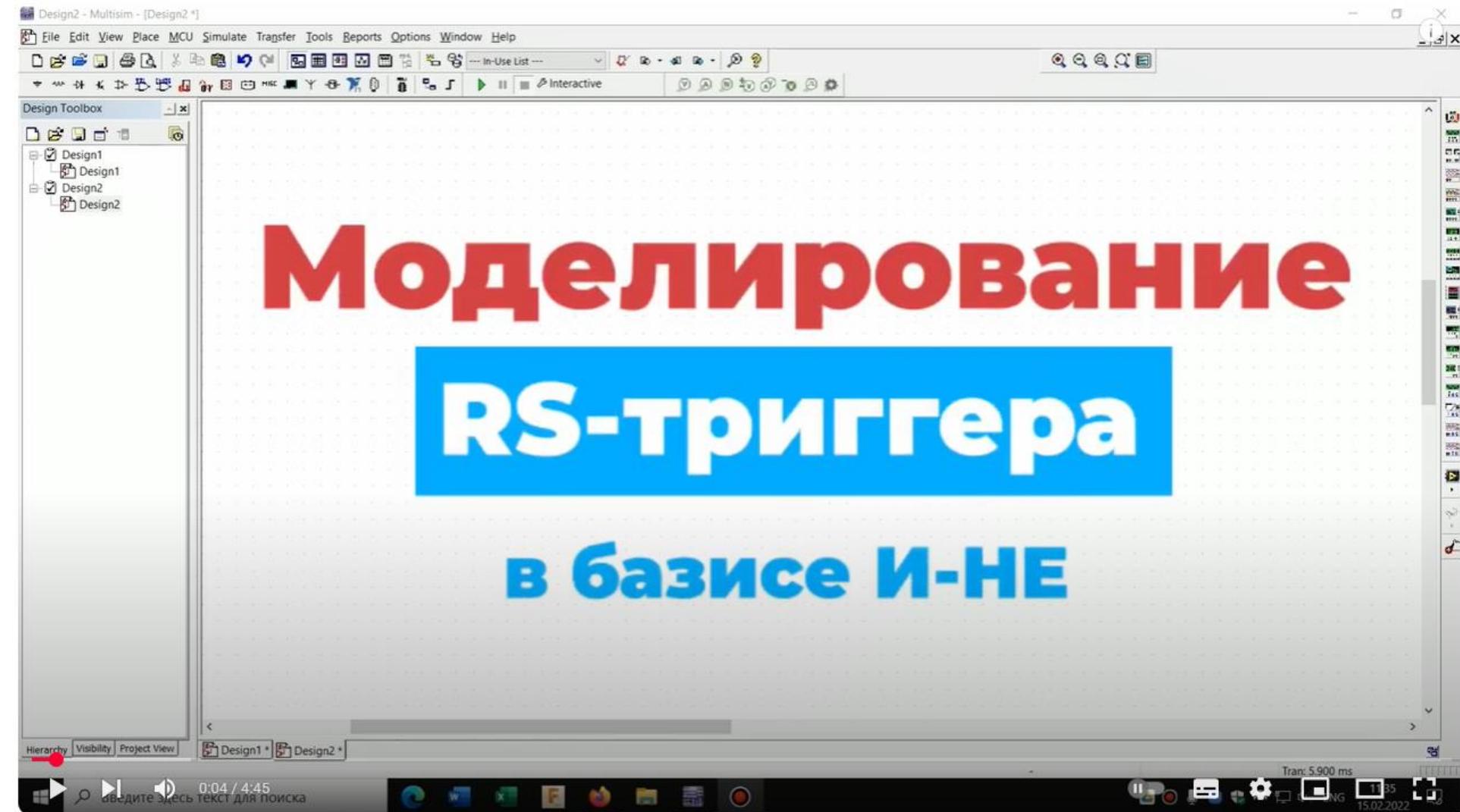
$$Q_{t+1} = Q_t \cdot \bar{R}' \vee S'$$

$$\bar{Q}_{t+1} = \overline{\bar{Q}_t \vee \bar{R}' \vee S'}$$



1.3. Асинхронный статический RS-триггер в базисе И-НЕ (2022)

https://www.youtube.com/watch?v=aVWAJy_0gFs



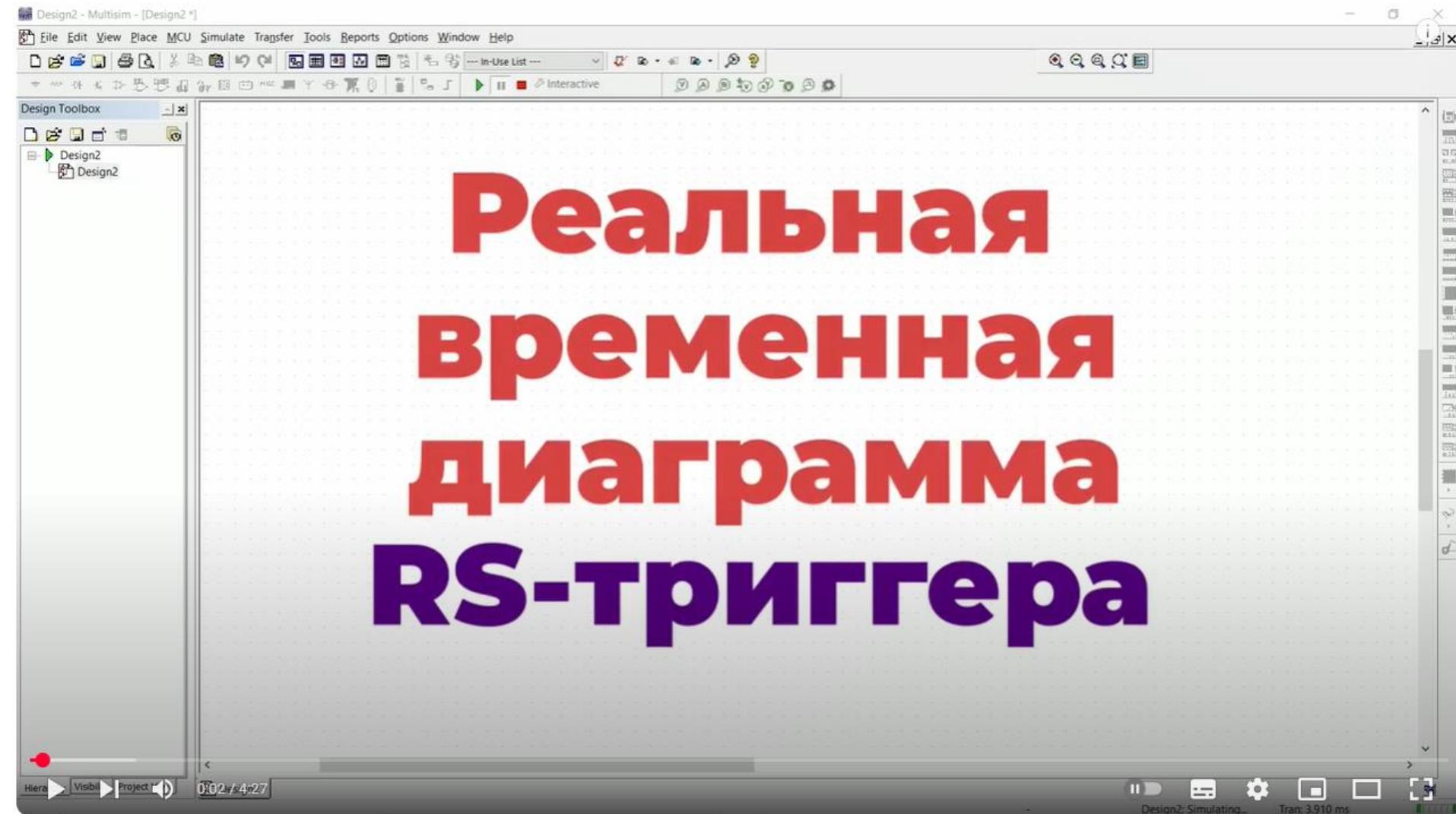
1.4. Моделирование RS-триггера в Multisim (2022)

<https://www.youtube.com/watch?v=IfEOA8w4HJA>

Временная диаграмма **RS-триггера** в базисе ИЛИ-НЕ



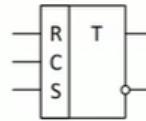
1.5. Временная диаграмма RS-триггера в базисе ИЛИ-НЕ (2022)
<https://www.youtube.com/watch?v=qoCyZ4ygK-Y>



1.6. Построение временной диаграммы RS-триггера с учетом задержек (2022)
https://www.youtube.com/watch?v=gA_x2hTK9-4

Синхронный статический RS-триггер

C	R	S	Q_{t+1}	\bar{Q}_{t+1}	сос
0	0	0	Q_t	\bar{Q}_t	хр
0	0	1	Q_t	\bar{Q}_t	хр
0	1	0	Q_t	\bar{Q}_t	хр
0	1	1	Q_t	\bar{Q}_t	хр
1	0	0	Q_t	\bar{Q}_t	хр
1	0	1	1	0	уст
1	1	0	0	1	сбр
1	1	1	*	*	з. к.



		RS			
		00	01	11	10
$Q_t C$	00	0	0	0	0
	01	0	1	*	0
$Q_t C$	11	1	1	*	0
	10	1	1	1	1

$$Q_{t+1} = Q_t \cdot \bar{C} \vee Q_t \cdot C \cdot \bar{R} \vee C \cdot S$$

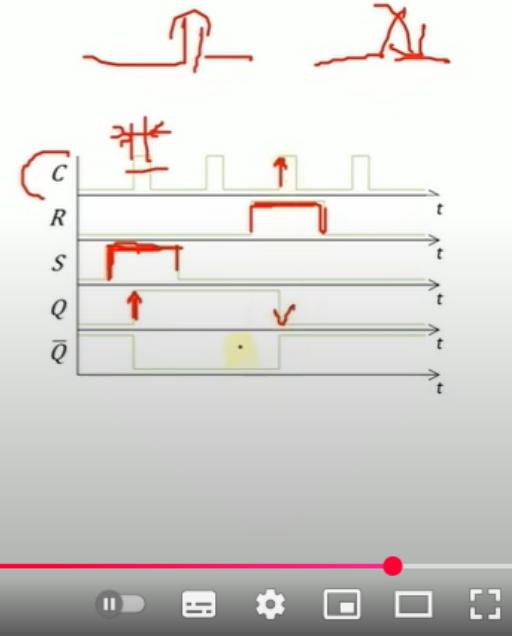
$$Q_{t+1} = Q_t \cdot \bar{C} \vee C(Q_t \cdot \bar{R} \vee S)$$

		RS			
		00	01	11	10
$Q_t C$	00	*	*	*	*
	01	*	0	*	*
$Q_t C$	11	0	0	*	1
	10	0	0	0	0

$$R_y = C \cdot R$$

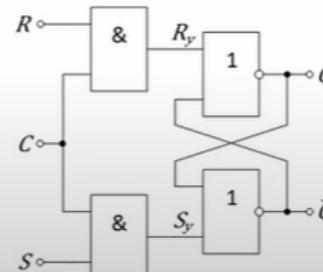
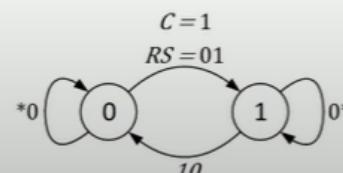
		RS			
		00	01	11	10
$Q_t C$	00	0	1	*	0
	01	0	*	*	*
$Q_t C$	11	*	*	*	*
	10	*	*	*	*

$$S_y = C \cdot S$$



Q_t	C	R	S	Q_{t+1}	R_y	S_y	сос
0	0	0	0	0	*	0	хр0
0	0	0	1	0	*	0	хр0
0	0	1	0	0	*	0	хр0
0	0	1	1	0	*	0	хр0
0	1	0	0	0	*	0	хр0
0	1	0	1	1	0	1	уст
0	1	1	0	0	*	0	сбр
0	1	1	1	*	*	*	з. к.
1	0	0	0	1	0	*	хр1
1	0	0	1	1	0	*	хр1
1	0	1	0	1	0	*	хр1
1	0	1	1	1	0	*	хр1
1	1	0	0	1	0	*	хр1
1	1	0	1	1	0	*	уст
1	1	1	0	0	1	*	сбр
1	1	1	1	*	*	*	з. к.

Q_{t+1}	\bar{Q}_{t+1}	R	S
0	0	*	0
0	1	0	1
1	0	1	0
1	1	0	*



2.5. Синхронный статический RS-триггер (2022)
<https://www.youtube.com/watch?v=O75Jjl2wnmE>

Элементы систем автоматики

Триггеры.

2. Асинхронные RS-триггеры в
базисе И-НЕ

И.В. Музылева
К.т.н., доцент



Асинхронный RS-триггер на элементах И-НЕ (2023)
<https://www.youtube.com/watch?v=1cYj8lxgec>

Элементы систем автоматики

Триггеры.

2. Асинхронные RS-триггеры в
базисе ИЛИ-НЕ

И.В. Музылева
К.т.н., доцент



Асинхронный RS-триггер в базисе ИЛИ-НЕ (2023)
<https://www.youtube.com/watch?v=IGzvWcWgHTM>

Элементы систем автоматики

Триггеры.

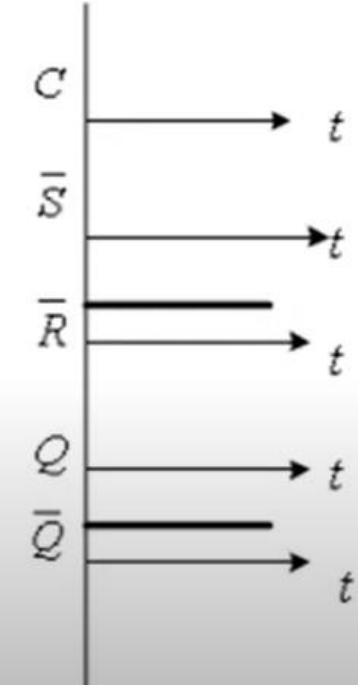
3. Синхронные RS-триггеры
со статическим управлением

И.В. Музылева
К.т.н., доцент

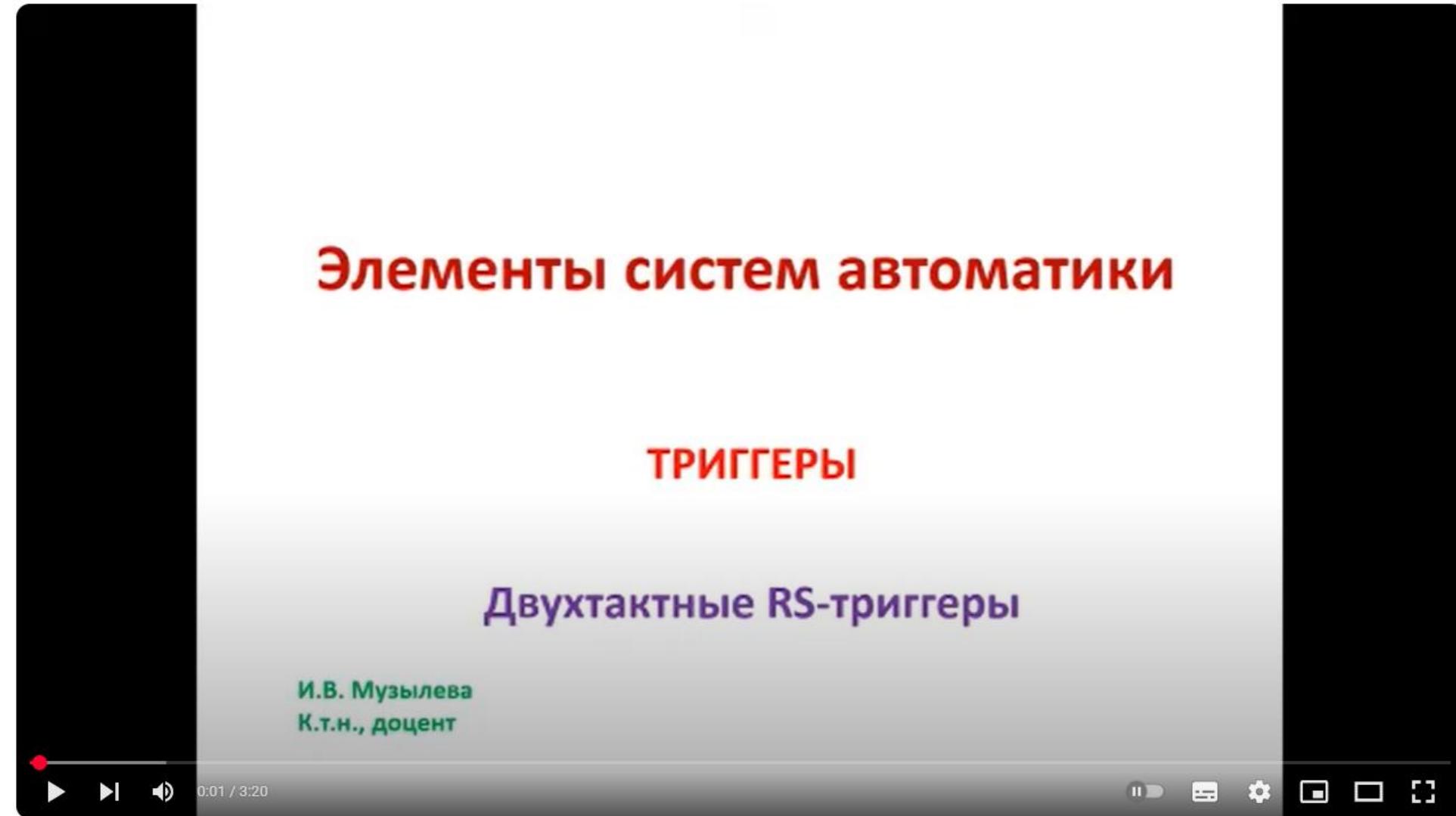


Синхронные RS-триггеры со статическим управлением на основе асинхронных в базисе И-НЕ и в базисе ИЛИ-НЕ (2023) <https://www.youtube.com/watch?v=t9sAoVSu3MY>

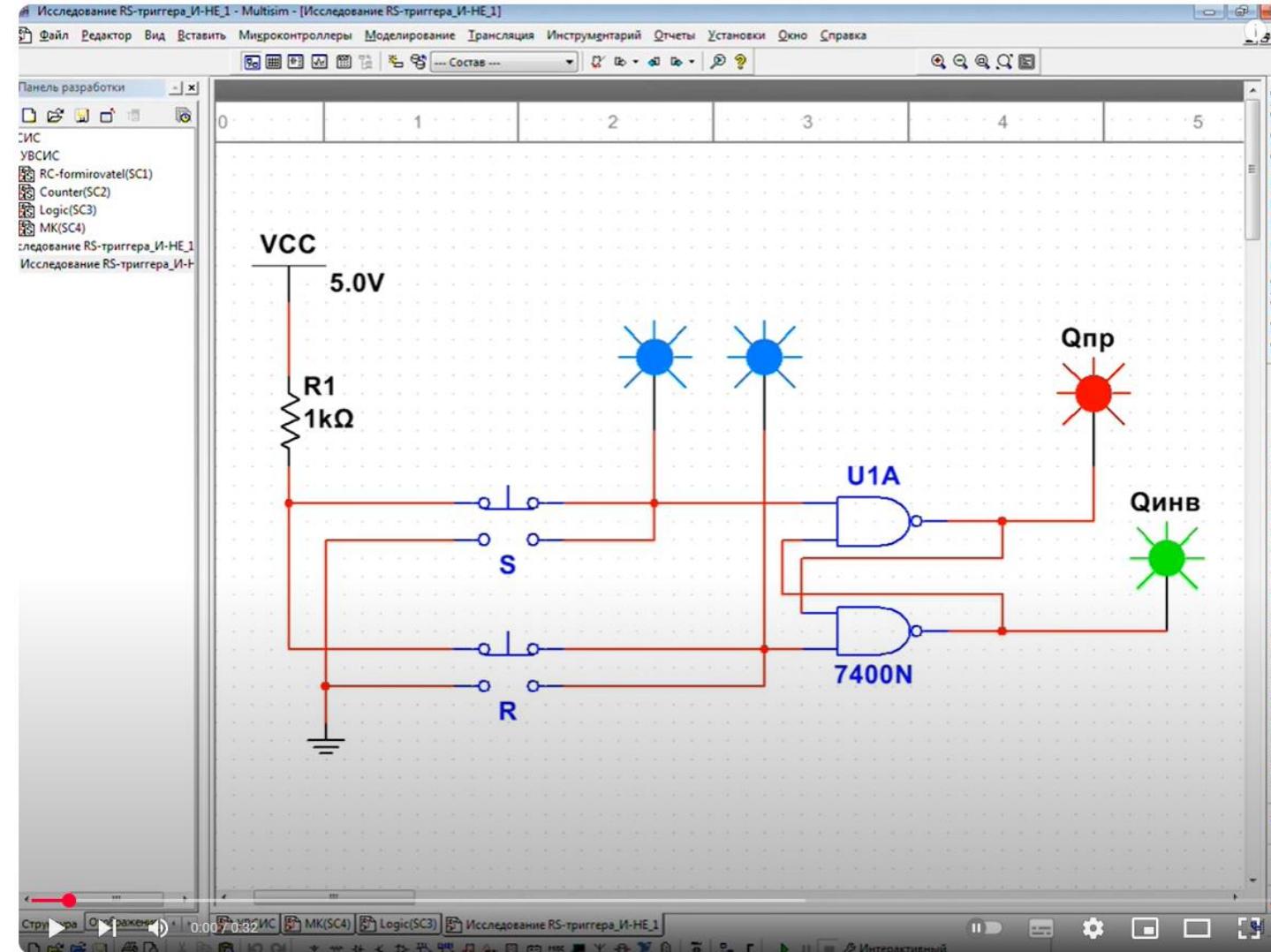
Синхронный RS-триггер с динамическим управлением по положительному фронту синхросигнала



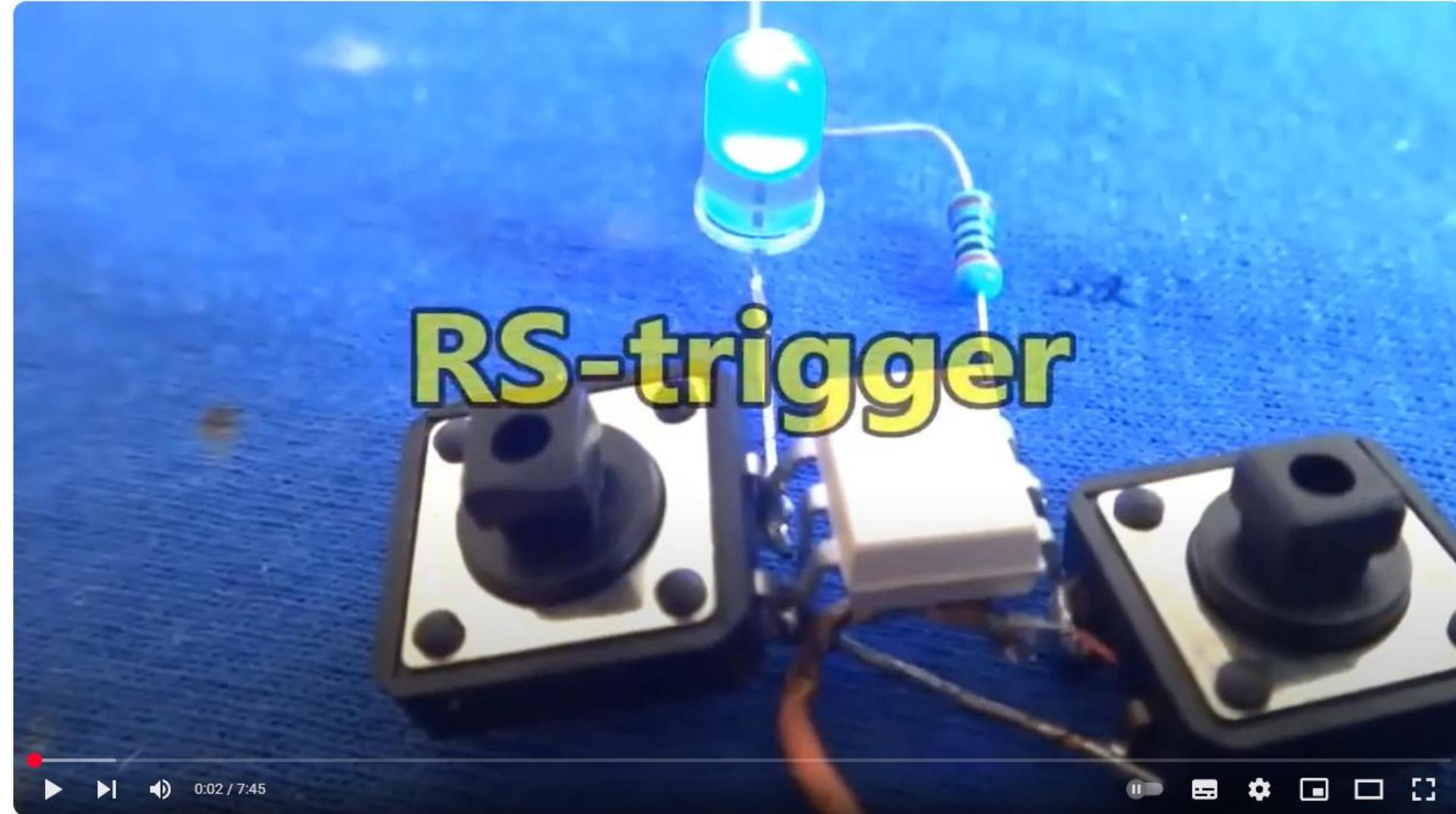
Синхронные RS-триггеры с динамическим управление (2023)
<https://www.youtube.com/watch?v=VTUDAO-8Opc>



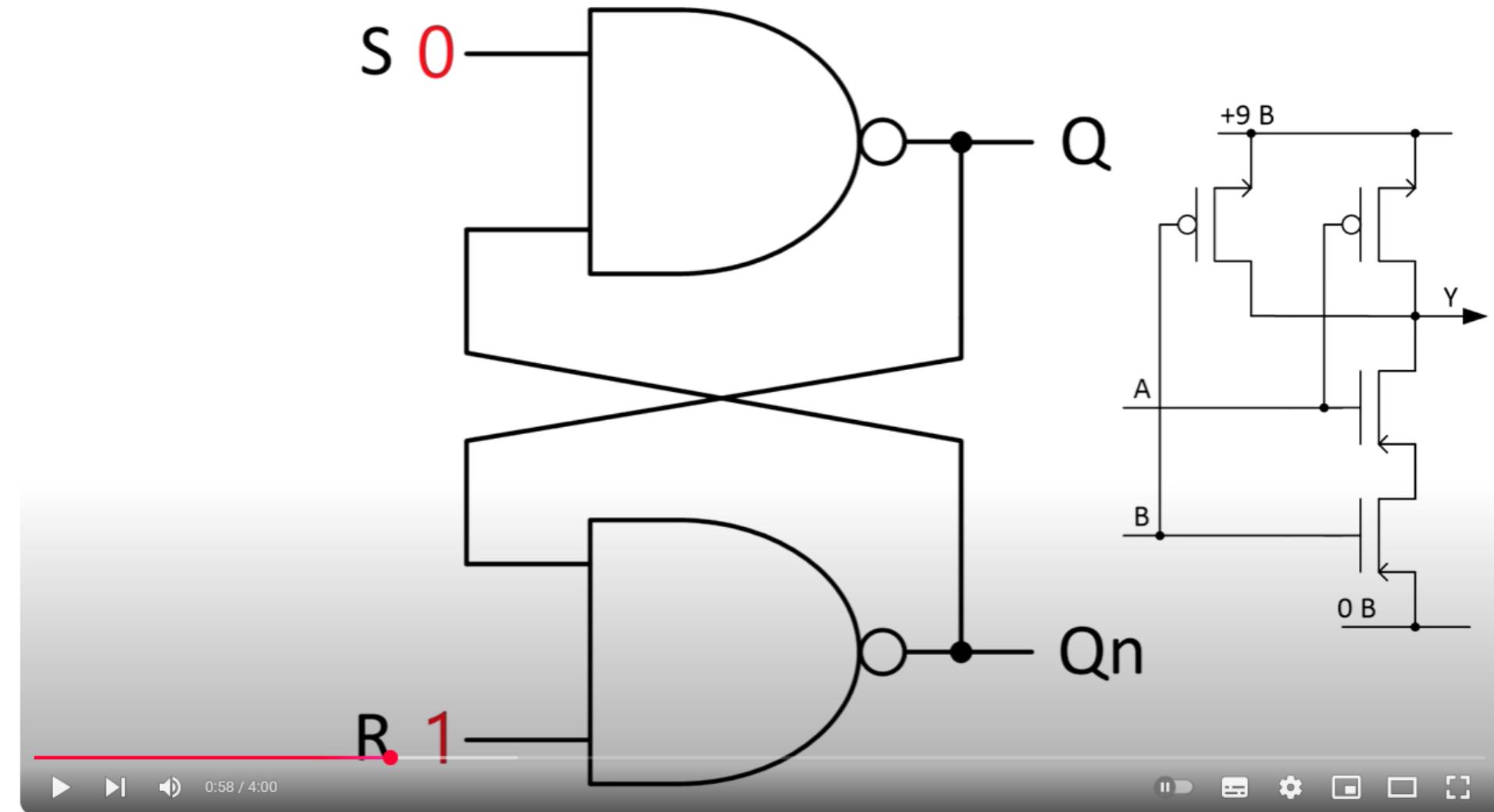
Триггеры. Двухтактные RS-триггеры (2023)
<https://www.youtube.com/watch?v=AgOB2vY4MRY>



Исследование работы RS-триггера на элементах И-НЕ в Multisim 14.1 (2021)
<https://www.youtube.com/watch?v=E3H6zO6U7yg>



RS-Trigger делаем: просто, сложно, мощно и ещё проще и веселей ! (2022)
<https://www.youtube.com/watch?v=Y9-wJEd5R5k>



RS-триггер. Видео №2 к 6 главе книги. (2021)
<https://www.youtube.com/watch?v=Aefa66TOY78>

Applications Places Document Viewer 18 °C en Thu May 23 20:40 • 100%

5 of 2 Doc.pdf 172.8% ▾

Introduction 3
Triggers clas... 4
Synchronous ... 5
Theory 5
Descrip... 5
Schema... 5
Truth ta... 5
Function... 5
Practice 6
Circuit ... 6
Results 6
Conclusion 7

Synchronous Static RS-Trigger

Theory

Description

RS-Trigger that change state only if exists signal on at synchronization input.

Schematic diagram

0099. Синхронный RS-Триггер (2019)

<https://www.youtube.com/watch?v=SRIA6tOz9n0>

Applications Places Document Viewer

15 °C en Wed May 22 20:43 100%
172.8% Doc.pdf

Introduction 3
Triggers clas... 4
Asynchronous... 5
Theory 5
Descrip... 5
Schema... 5
Truth T... 5
Function... 5
Practice 6
Circuit ... 6
Results 6
Conclusion 7

Asynchronous Static RS-Trigger

Theory

Description

Bi-stable relaxation schema.

- Bi-stable means that RS-Trigger can be in one of two possible states
- Relaxation schema means that it builds on positive feedback loop and sharply changes state

Schematic diagram



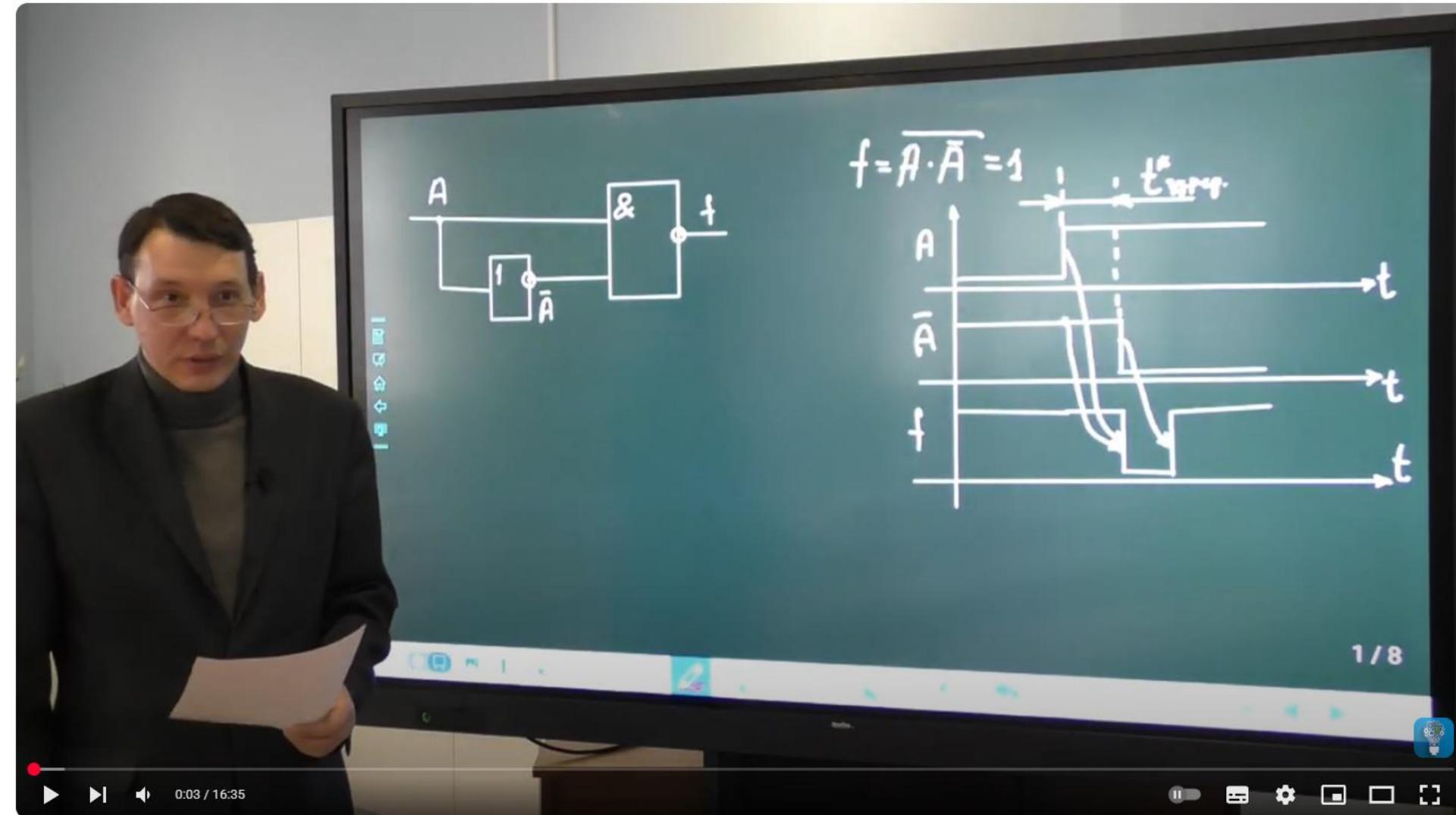
Truth Table

S	R	Q	\bar{Q}
0	0	Q^{-1}	\bar{Q}^{-1}
0	1	0	1
1	0	1	0

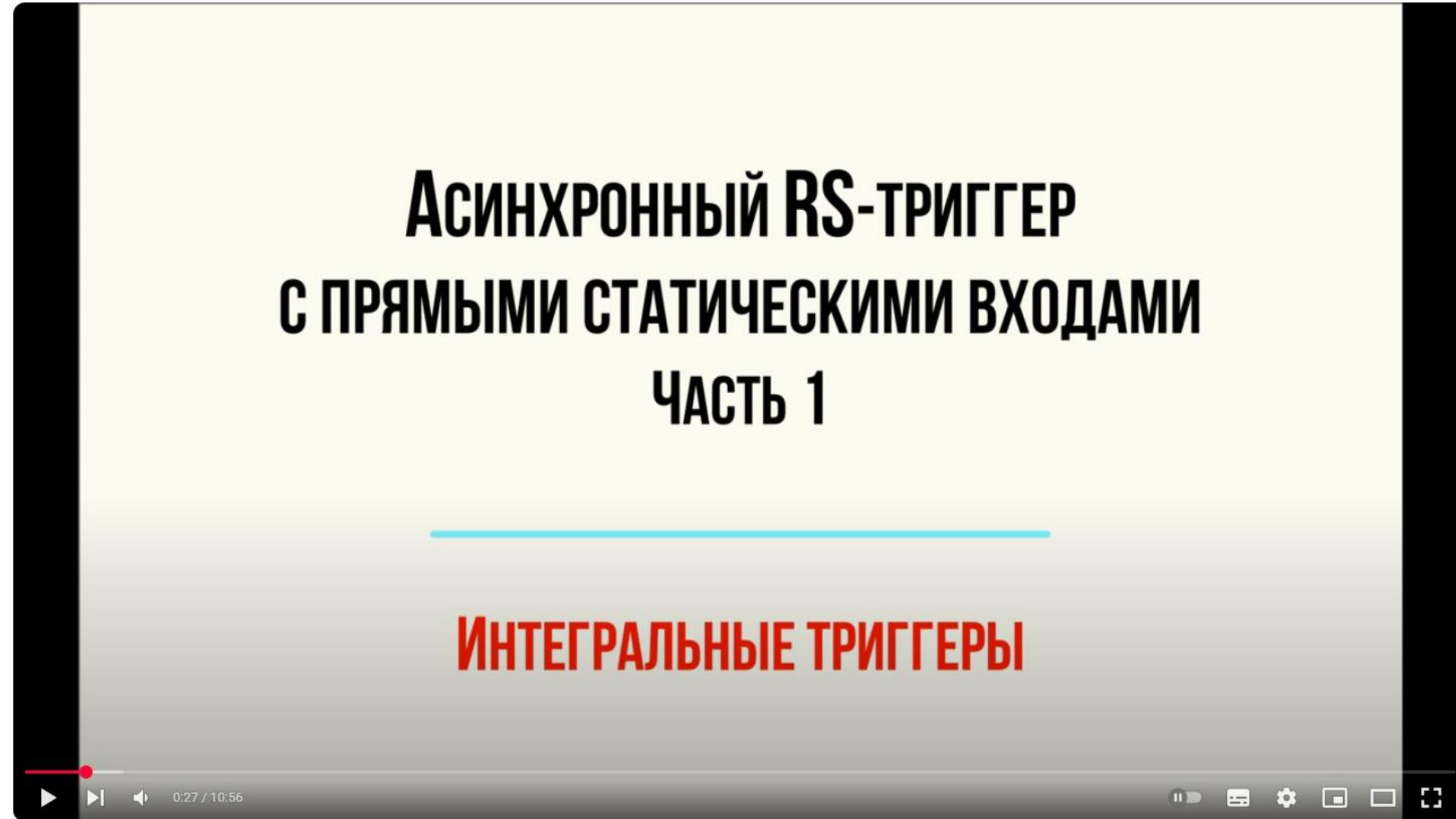
6:16 / 15:35 Doc.pdf SimpleScreenRecorder 1/4

0097. RS-триггер (2019)

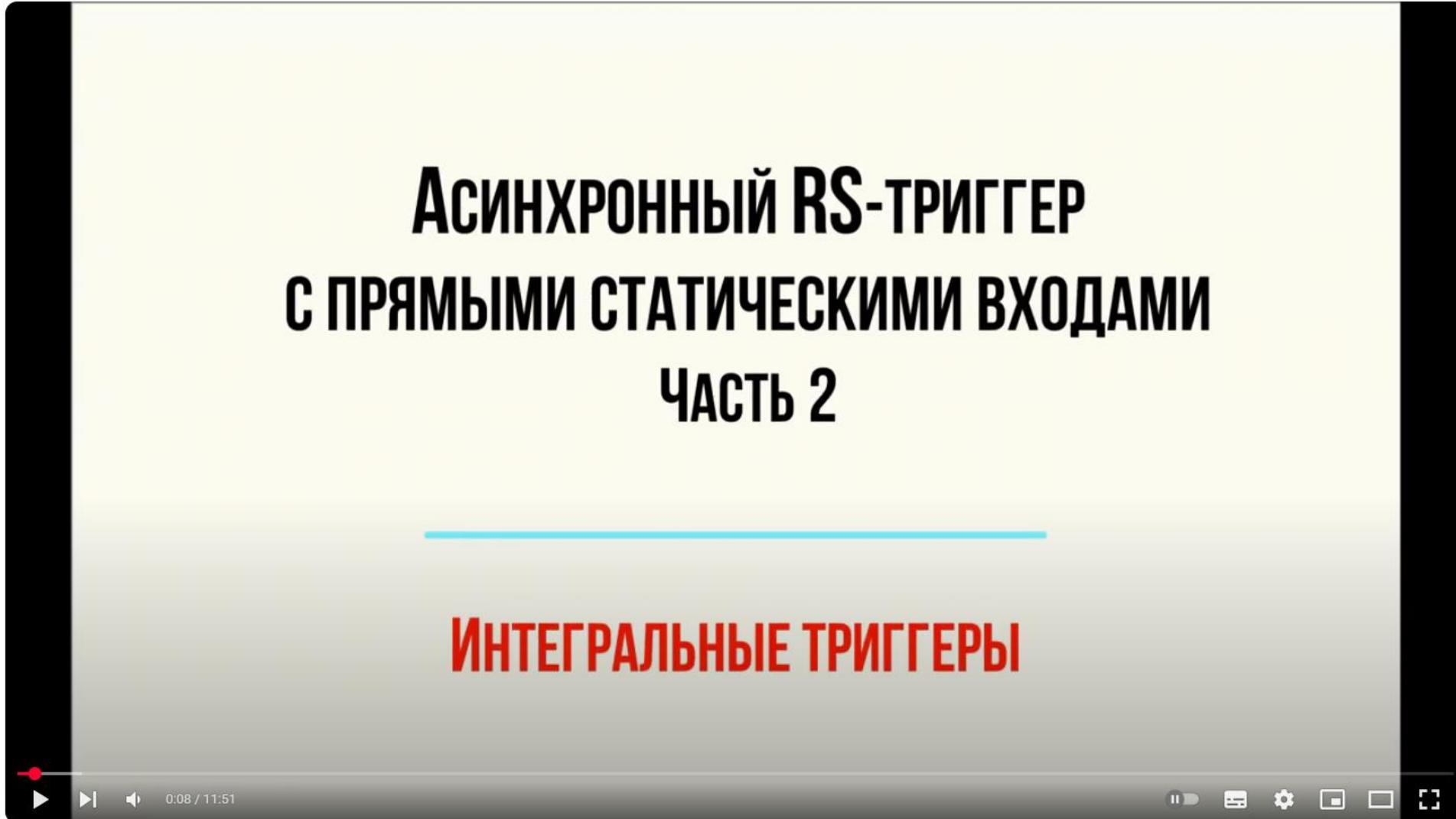
<https://www.youtube.com/watch?v=qJ52QGwdVIs>



РК6. Схемотехника. Синхронный RS-триггер (2020)
<https://www.youtube.com/watch?v=AqcPlfQSgiw>



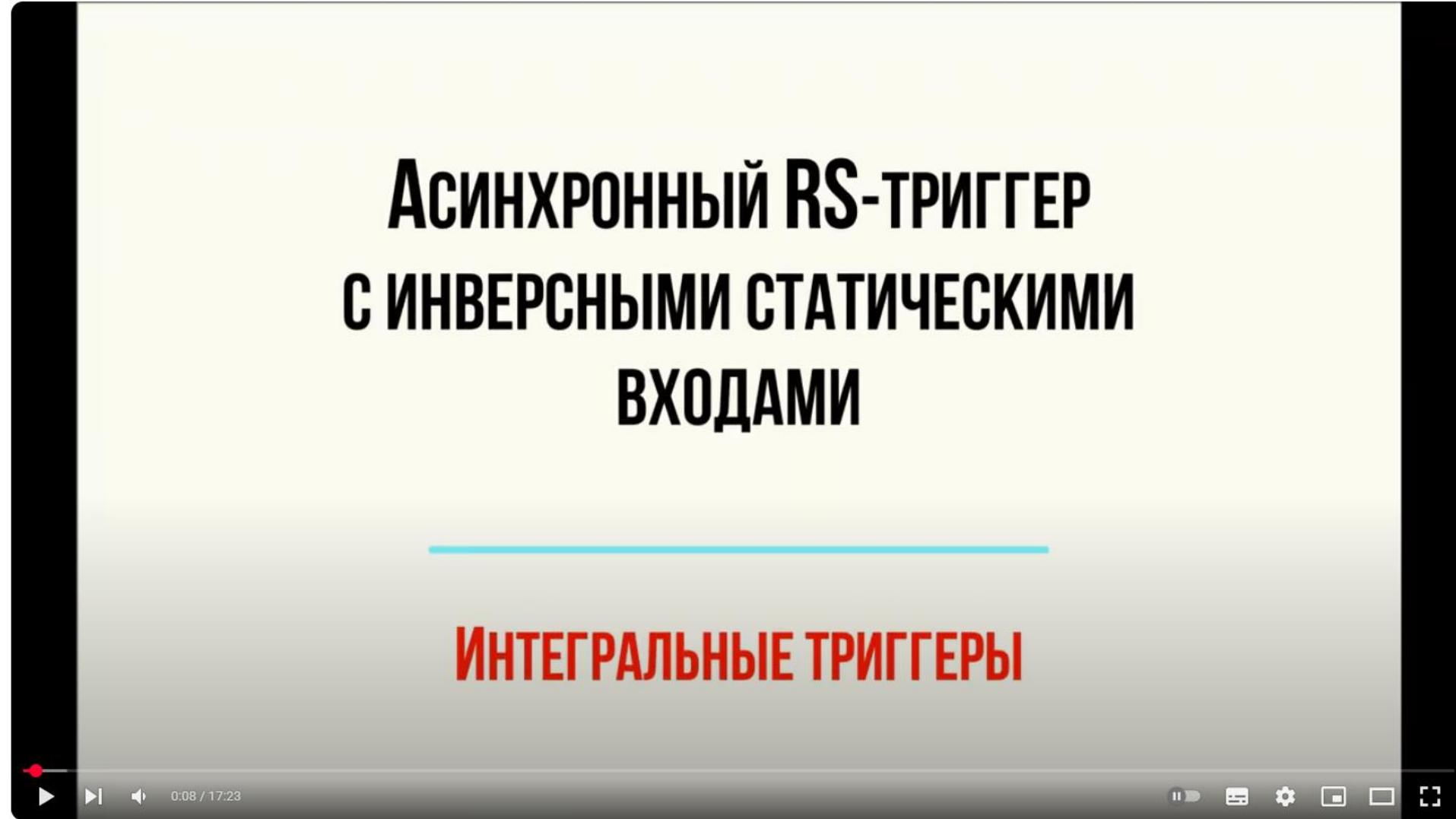
Вideoурок 4.4. Асинхронный RS-триггер с прямыми статическими входами. Часть 1 (2024) https://www.youtube.com/watch?v=S3sVe_Ogk8w



Вideoурок 4.5. Асинхронный RS-триггер с прямыми статическими входами.
Часть 2 (2024) <https://www.youtube.com/watch?v=uRLMJ2DmACK>

Асинхронный RS-триггер с инверсными статическими входами

ИНТЕГРАЛЬНЫЕ ТРИГГЕРЫ



Вideoурок 4.6. Асинхронный RS-триггер с инверсными статическими
входами (2024) <https://www.youtube.com/watch?v=OB-YG8ykWB8>

Интегральные триггеры

**асинхронный RS-триггер
с прямыми статическими входами**



J, S (set - установка) - вход установки (установки 1)

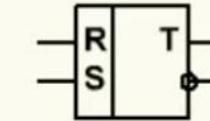
K, R (reset - сброс) - вход сброса (сброса в 0)

D (data - данные, информация) - информационный вход

C или CLK (clock - тактовый) - тактовый вход или синхронизирующий вход

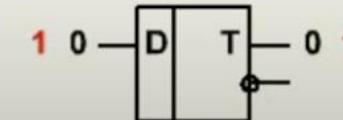
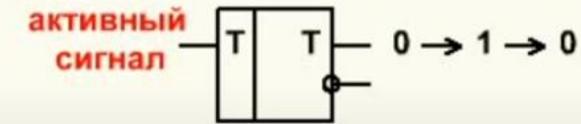
РАЗРЕШЕНИЕ

асинхронные

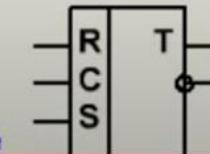


Графическое изображение

T (toggle - переключатель) · счетный вход



синхронные



SR FLIP FLOP

$S' = \overline{S \cdot CLK}$
 $= \overline{\overline{S} + \overline{CLK}} \quad \text{--- 1}$

$R' = \overline{R \cdot CLK}$
 $= \overline{R} + \overline{CLK} \quad \text{--- 2}$

$\text{CLK} = 0$

$S' = \overline{\overline{S} + 1} = 1$

$R' = \overline{\overline{R} + 1} = 1$

Now Qs

A	B	$\overline{A} \cdot \overline{B}$
0	0	1
0	1	0
1	0	0
1	1	0

$\text{II } \text{CLK} = 1$

(i) $S=0, R=0$ (ii) $S=0, R=1$
 $S'=1, R'=1$ $S'=1, R'=0$

(iii) $S=1, R=0$ (iv) $S=1, R=1$
 $S'=0, R'=1$ $S'=0, R'=0$

Truth Table for SR Latch with NAND Gate

S'	R'	Q	\bar{Q}
0	0	NOT Used	
0	1	1	0
1	0	0	1
1	1	Prev. State (Memory)	

S SR FF Q
 CLK R \bar{Q}

5. SR Flip Flop in Hindi | Tech Gurukul by Dinesh Arya (2018)
 5. SR Триггер на хинди | Tech Gurukul от Динеша Арыи (2018)
<https://www.youtube.com/watch?v=UIRPUaOfvco>

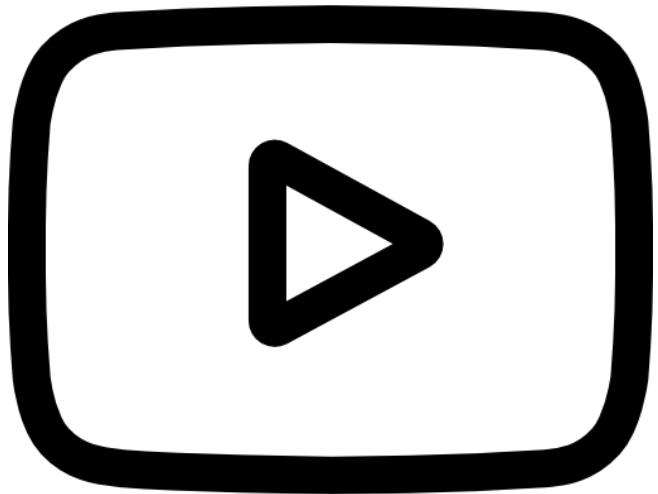
SR Latch 4 (Gated SR Latch)

NAND

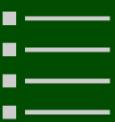
A	B	Output
0	0	1
1	0	1
0	1	1
1	1	0

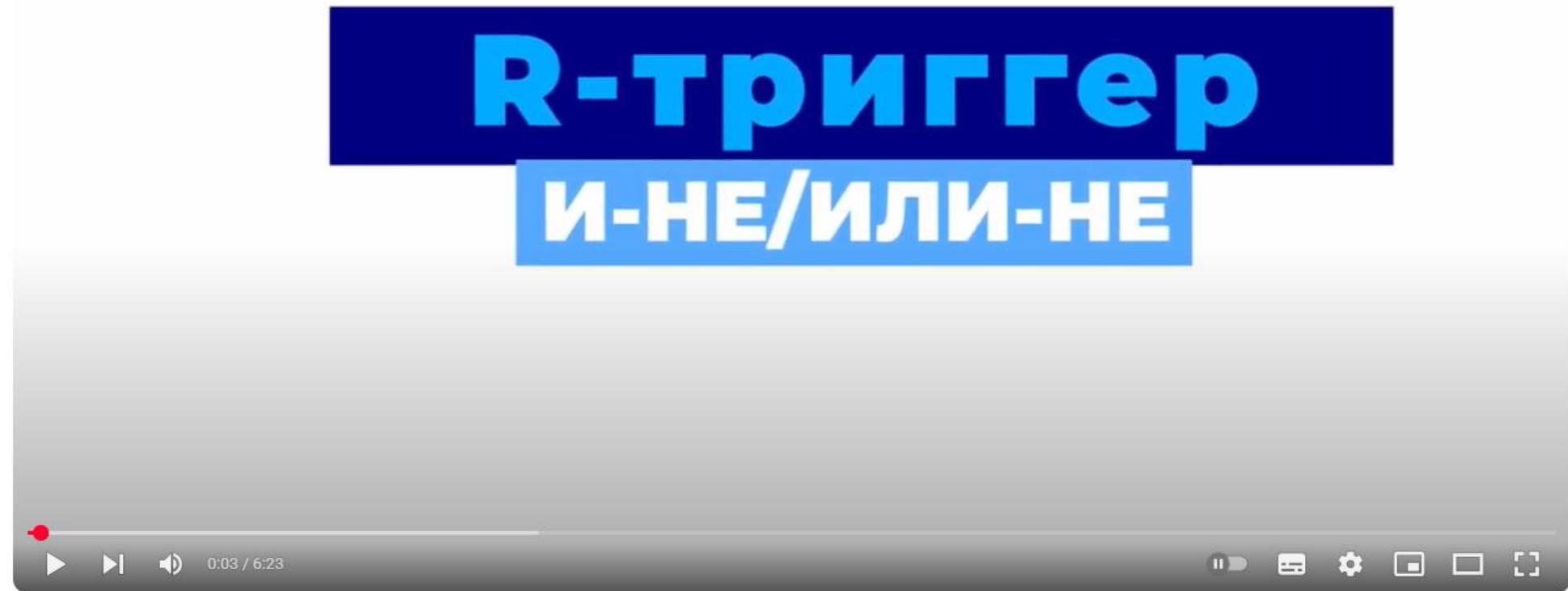
0:15 / 11:55 • Intro >

SR Latch, Gated SR Latch, and Data Latch (2023)
 Sr-защелка, закрытая SR-защелка и защелка для передачи данных (2023)
<https://www.youtube.com/watch?v=NBihtSRjA>

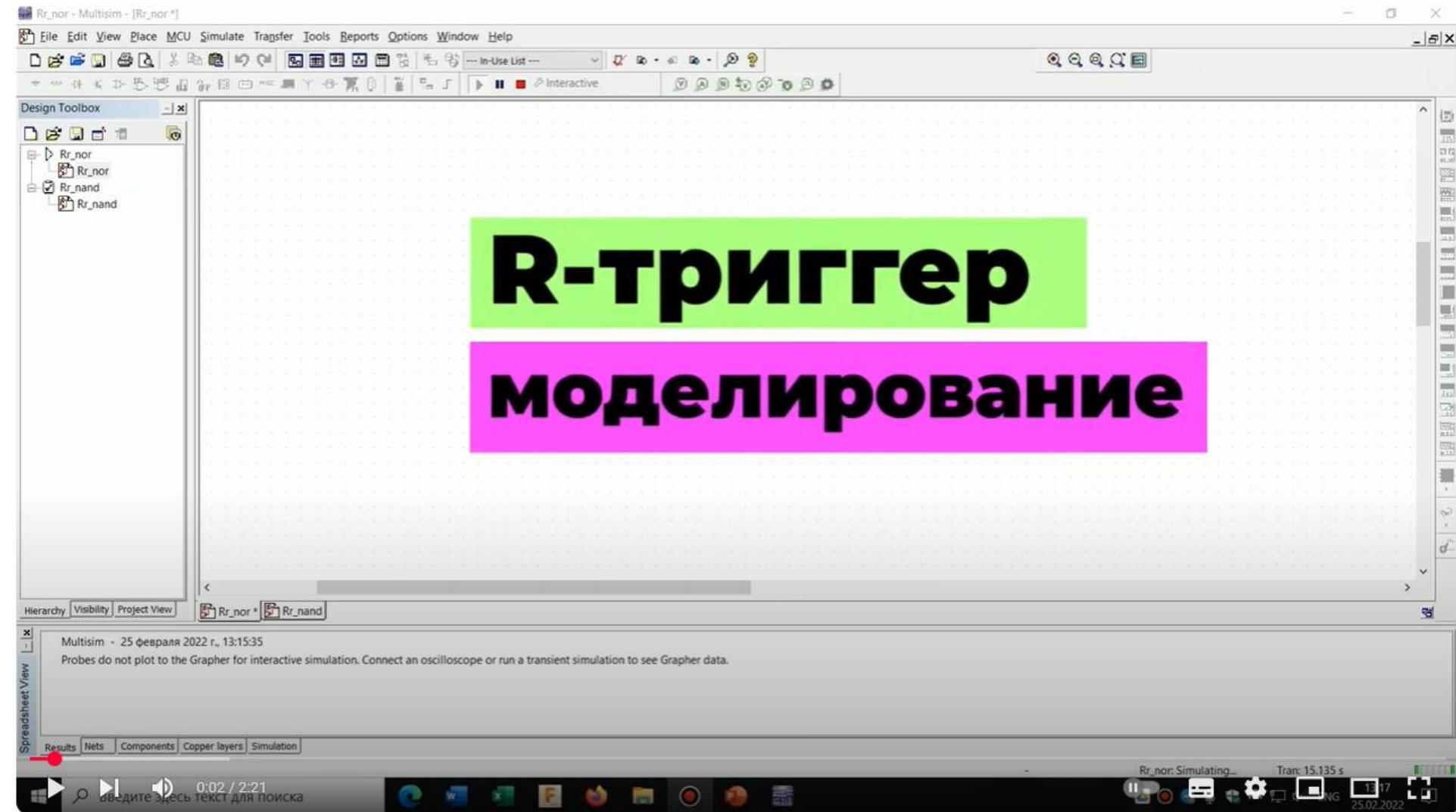


R-триггер

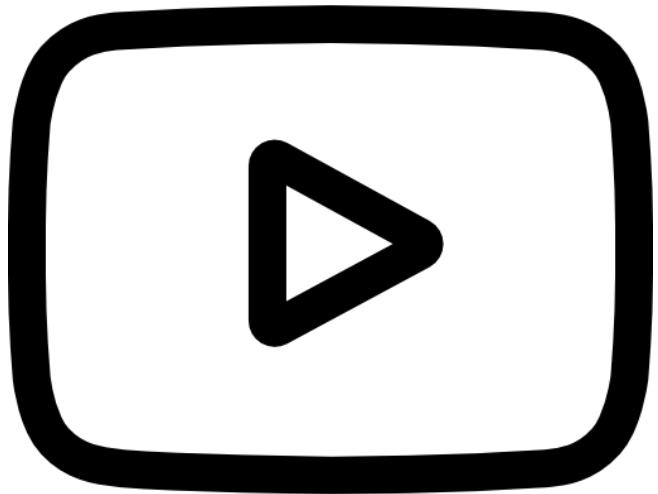




2.3. Асинхронный статический R-Триггер (2022)
<https://www.youtube.com/watch?v=nexJwj4mzLk>

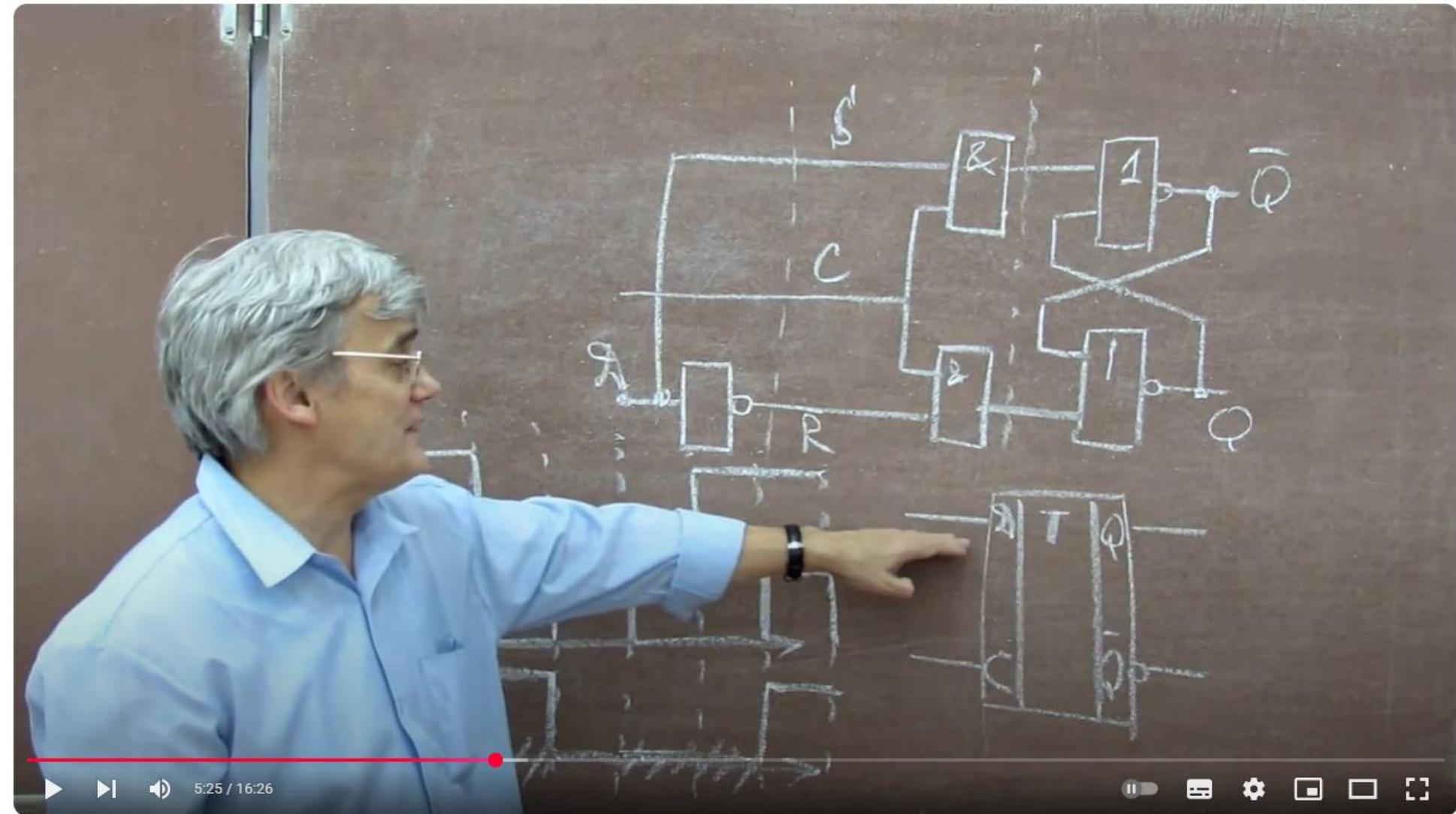


2.4. Моделирование R-триггера (2022)
<https://www.youtube.com/watch?v=pUXRbxnVys8>



D-триггер





Лекция 100. D-триггер (2013)
<https://www.youtube.com/watch?v=tKZkAx9Q3Po>



D-Триггер, подробный обзор и тестирование! [Электроника] Для начинающих! Не DIY KIT, Микросхема (2018) <https://www.youtube.com/watch?v=VX6Rz2VVxP4>

Белорусско-Российский университет, Кафедра «Программное обеспечение информационных технологий»

Характеристическая функция

$$Q_{n+1} = \bar{C}_n \cdot Q_n + C_n D_n$$

При $C=1$ $Q_{n+1} = D_n$

$$D_n \xrightarrow{\quad} Q_n \Rightarrow D_n \xrightarrow{\quad} \bar{Q}_n \xrightarrow{\quad} Q_n$$

$$R^* = C_n \bar{D}_n; R^* = \bar{D}_n \cdot C_n$$

$$R^* = \overline{\bar{D}_n \cdot C_n} = \bar{D}_n + \bar{C}_n = \bar{D}_n(C_n + \bar{C}_n) + \bar{C}_n = \bar{D}_n C_n + \bar{D}_n \bar{C}_n + \bar{C}_n =$$

$$= \bar{D}_n C_n + (\bar{D}_n + 1) \cdot \bar{C}_n = \underline{\bar{C}_n} + \bar{C}_n = \bar{S}^* + \bar{C}_n = S^* \cdot C_n$$

6 / 8

РК6. Схемотехника. Синхронный D-триггер (2020)
https://www.youtube.com/watch?v=MAgzsUO-z_I

Applications Places Document Viewer Doc.pdf 24 °C en Tue May 28 20:21 100% 207.4% 100%

Introduction 3
Synchronous 4
Theory 4
Descrip... 4
Schema... 4
States t... 4
On chip... 4
Function... 4
Practice 5
Circuit... 5
Results 6
Conclusion 7

Synchronous / Asynchronous Dynamic D-Trigger

Theory

Description

Synchronous dynamic D[RS]-Trigger change its state on clock change from negative to positive, in case signal on D input was changed to opposite.

Schematic diagram

Figure 2: Synchronous Dynamic D-Trigger (Positive Edge)

Figure 1: Asynchronous Dynamic D-Trigger / Frequency Divider (Positive Edge)

States table (Synchronous Dynamic D-Trigger (Positive Edge))

C / D	D = 0	D = 1
C: 0 → 1	0	1
C: 1 → 0	NOT CHANGED	NOT CHANGED

On chip implementation

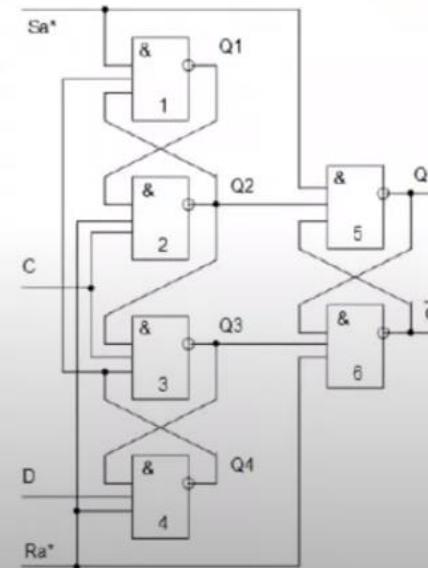
0100. D-Триггер / 0104. Схема деления частоты (2019)
<https://www.youtube.com/watch?v=dthinytMTEk>

Синхронные триггеры с динамическим управлением записью

В этих триггерах вход синхронизации является динамическим. Это означает, что информация записывается в триггер с информационного входа по перепаду 0/1 (по переднему фронту) или 1/0 (по заднему фронту) синхросигнала. В первом случае динамический вход называется прямым, во втором случае – инверсным. Информация в триггер записывается только в момент действия перепада сигнала. Рассмотрим схему D-триггера.

Синхронный D-триггер с динамическим управлением записью

Триггер управляется передним фронтом синхросигнала. Схема состоит из трех триггеров: основного (ЛЭ5,6) и двух вспомогательных (ЛЭ1,2 и ЛЭ3,4). Вспомогательные триггеры служат соответственно для записи 1 и 0 в основной триггер. Выходные сигналы основного триггера – это выходные сигналы триггера в целом. S_a^* и R_a^* – инверсные асинхронные входы установки триггера в 1 и 0 соответственно. Проанализируем работу схемы при $S_a^*=R_a^*=1$ (сигналы не активны).



При $C=0$:

$Q_2=Q_3=1$ и основной триггер находится в режиме хранения информации.

$Q_4=\overline{R_a^* \cdot D} \cdot Q_3=1 \cdot \overline{D} \cdot 1=\overline{D}$; $Q_1=\overline{S_a^*} \cdot Q_4 \cdot \overline{Q_2}=1 \cdot \overline{Q_4} \cdot 1=\overline{Q_4}=D$. Следовательно, в одном из вспомогательных триггеров оба выходных сигнала равны 1.

При $C=1$ это неправильное состояние исчезает. Установку выходного сигнала определяет тот вспомогательный триггер, который находился в неправильном состоянии. Т.о. если $C=1$:

- При $D=0$: $Q_4=1$; $Q_1=0$, значит $Q_2=\overline{R_a^* \cdot C \cdot Q_1}=\overline{R_a^* \cdot C \cdot 0}=1$ и $Q_3=\overline{Q_4 \cdot C \cdot Q_2}=1 \cdot 1 \cdot 1=0$ устанавливает выходной триггер в состояние 0.
- При $D=1$: $Q_4=0$; $Q_1=1$, значит $Q_2=\overline{Q_4 \cdot C \cdot Q_2}=0 \cdot C \cdot 1=1$ и $Q_3=\overline{R_a^* \cdot C \cdot Q_1}=1 \cdot 1 \cdot 1=0$ устанавливает выходной триггер в состояние 1.

Заметим, что пока $Q_2=0$, то $Q_3=1$ и $Q_1=1$, т.е. ЛЭ1 и ЛЭ3 заблокированы. При изменении сигнала D меняется Q4, но Q3 и Q1 не меняются. Т.е. никакие изменения сигнала D не изменят состояния ЛЭ1 и ЛЭ3, а значит и состояние триггера.

Если $Q_2 \neq 0$, то $Q_4=1$, т.е. заблокирован ЛЭ4. Любые изменения сигнала D не изменят состояния ЛЭ4, а значит и состояние триггера.

Чтобы триггер начал реагировать на вход D нужно сбросить сигнал C в состояние 0.



РК6. Схемотехника. D-триггер с динамическим управлением записью (2020)
https://www.youtube.com/watch?v=OJK_WxAxmCA

Быстродействие триггера определяется теми же формулами, что и для синхронного двухступенчатого RS-триггера.

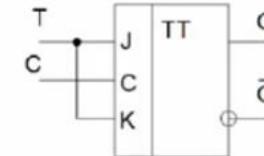
$$t_H = 3t^*$$

t^* - средняя задержка распространения сигнала на логическом элементе

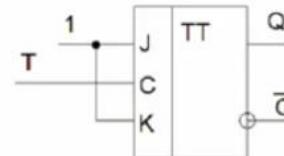
$$t_p = 7t^*$$

$$f_{max} = \frac{1}{t_p} = \frac{1}{7t^*}$$

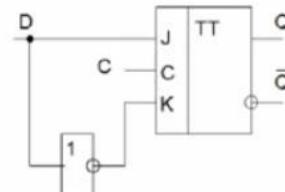
JK-триггер универсальный триггер. Он может использоваться как RS-, T- и D- триггеры. Варианты использования триггера



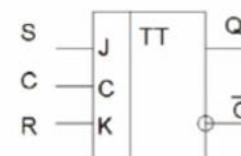
Синхронный
T-триггер



Асинхронный
T-триггер



Синхронный
D-триггер



Синхронный
RS-триггер.



Синхронный двухступенчатый D-триггер

Если в первой ступени применить синхронный D-триггер, со статическим управлением записью, а во второй -- синхронный RS-триггер, то получится синхронный двухступенчатый D-триггер.

Схема синхронного двухступенчатого D-триггера с запрещающими связями



РК6. Схемотехника. Двухступенчатый D-триггер с запрещающими связями (2020)
<https://www.youtube.com/watch?v=EUzLoQM67SI>

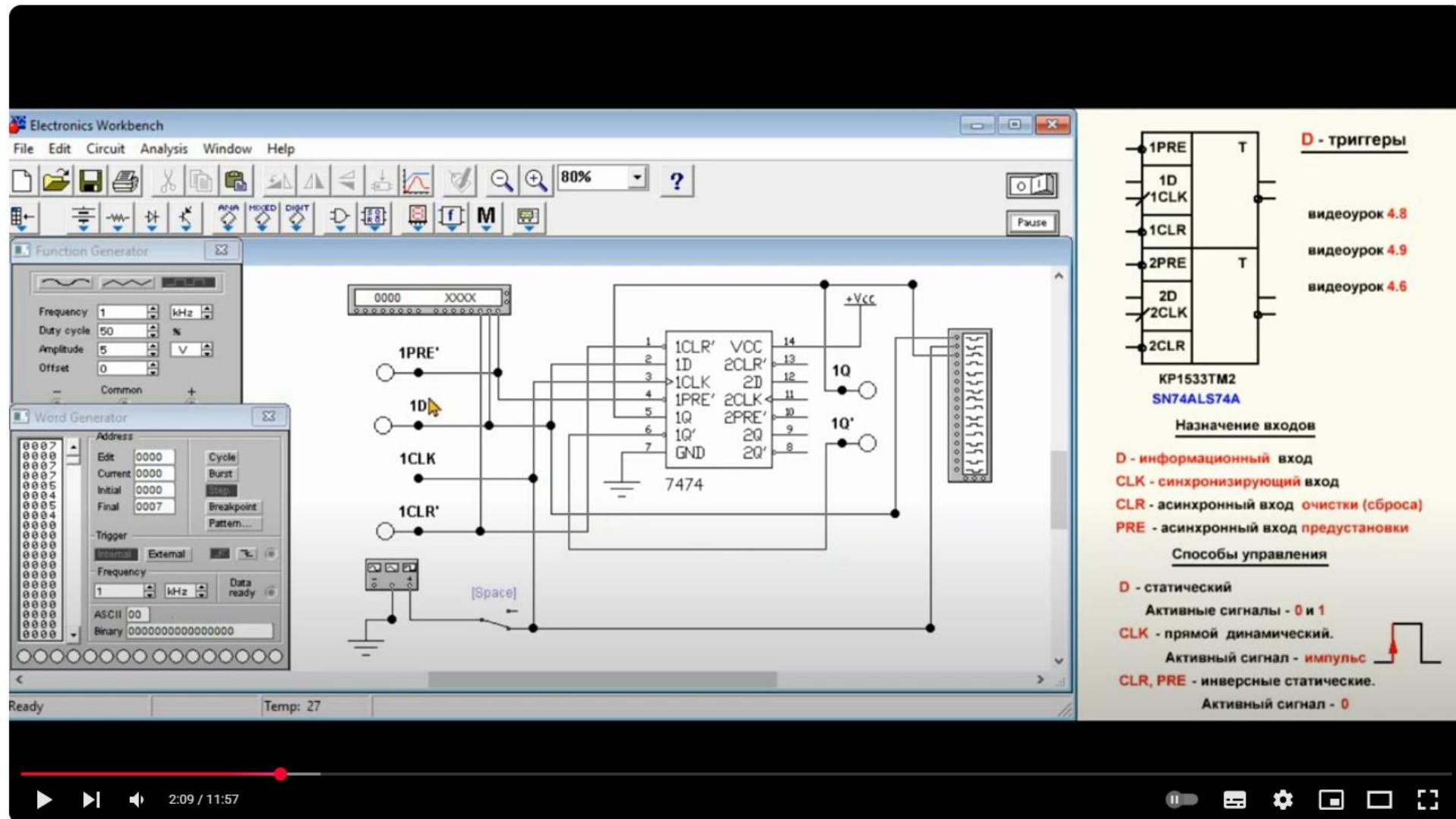
D-ТРИГГЕРЫ

ЧАСТЬ 1

ИНТЕГРАЛЬНЫЕ ТРИГГЕРЫ

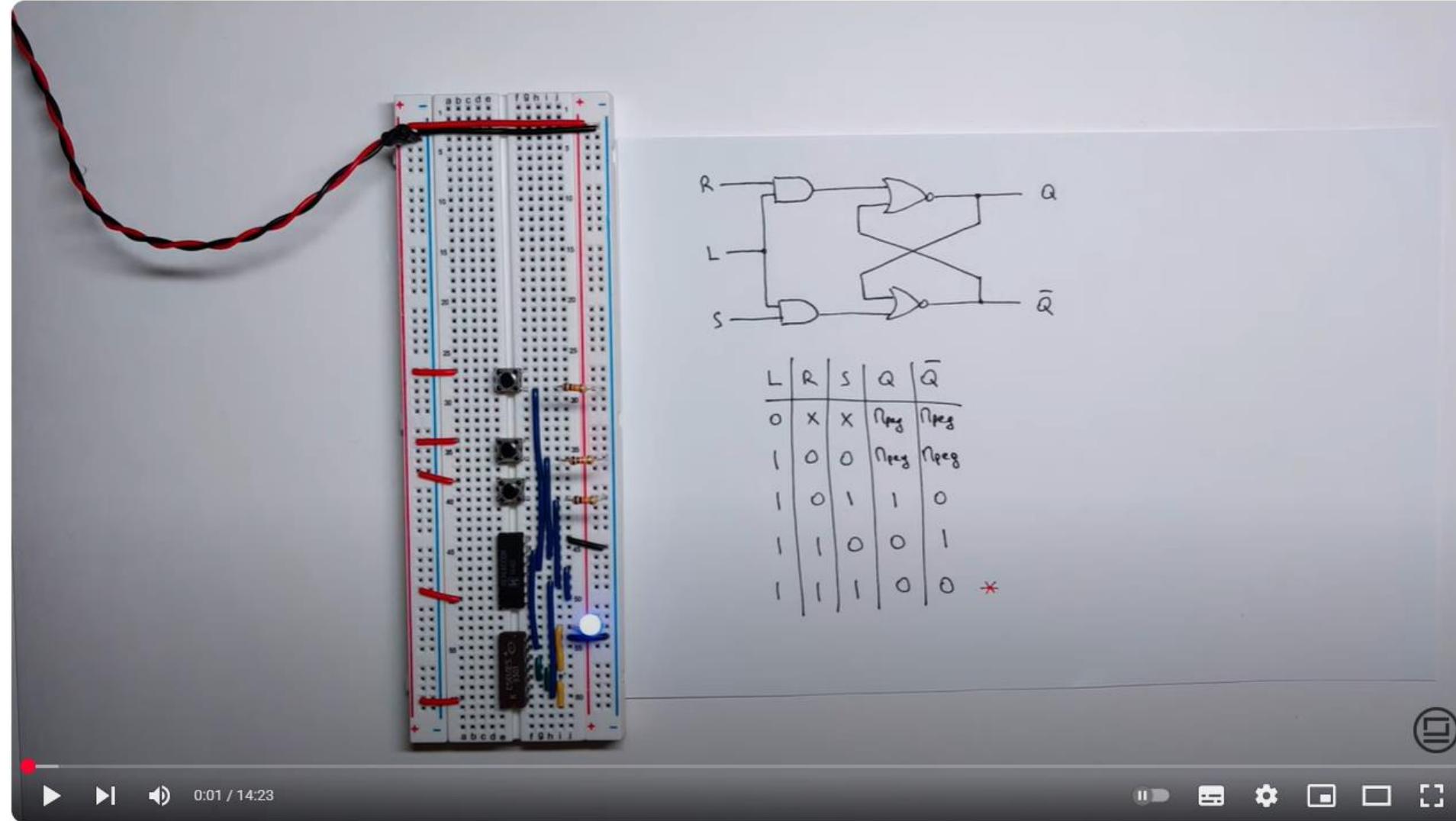


Вideoурок 4.10. D-триггеры. Часть 1. (2024)
<https://www.youtube.com/watch?v=ffRJ428Yw2o>

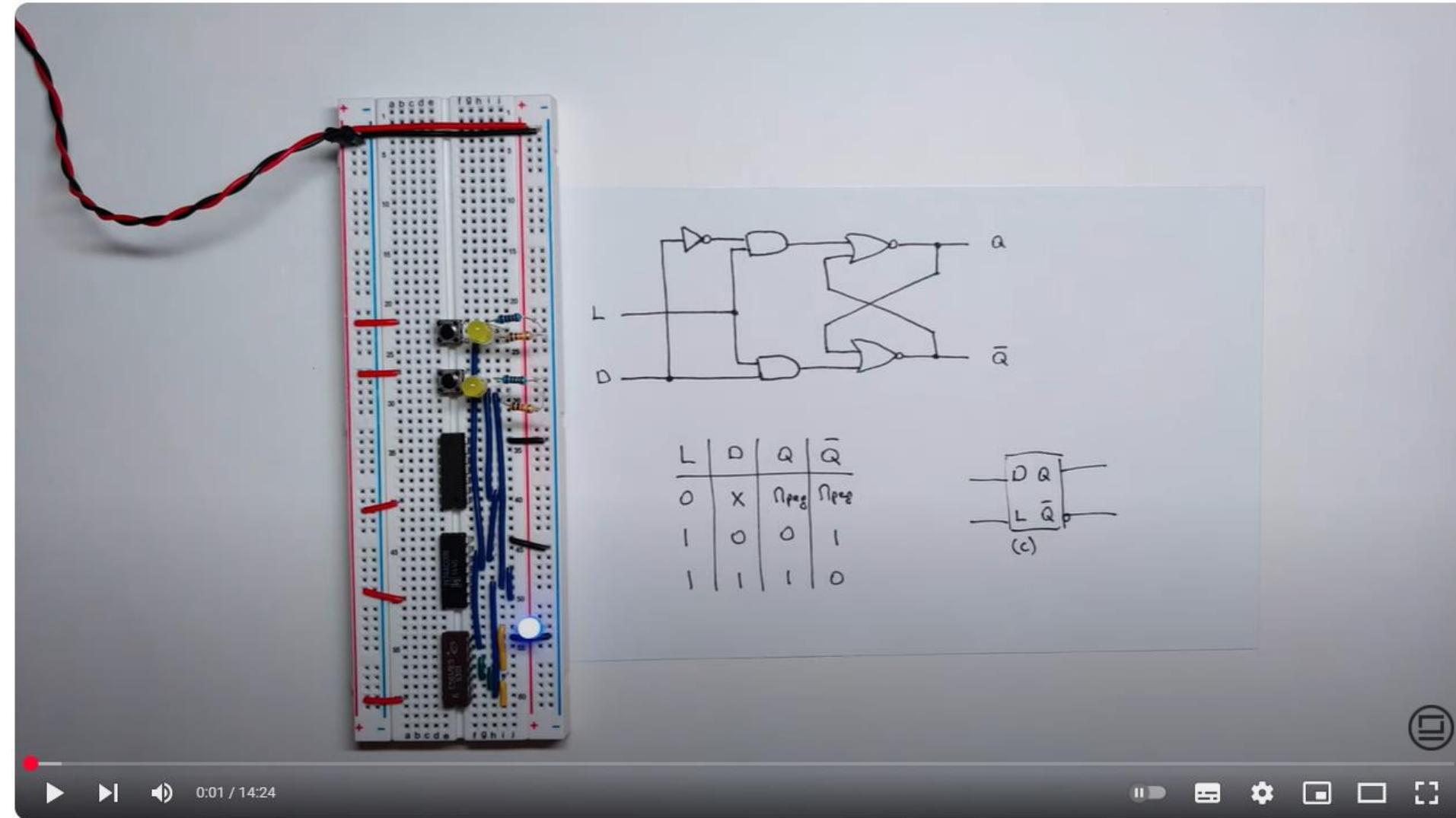


Видеокурс 4.11. D-триггеры. Часть 2. (2024)

<https://www.youtube.com/watch?v=KXZ4CWaXhLs>



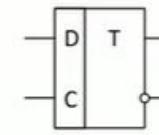
D-триггер: устранием состояния гонки в RS-триггере (2020)
https://www.youtube.com/watch?v=_A8-oLOkza0



D-триггер с динамическим управлением (2020)
<https://www.youtube.com/watch?v=8SeIYLY5fkl>

Синхронный статический D-триггер

D	Q_{t+1}	сос
0	0	сбр
1	1	уст

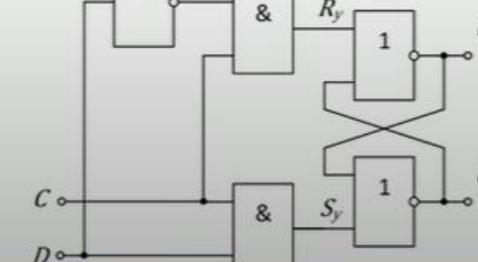
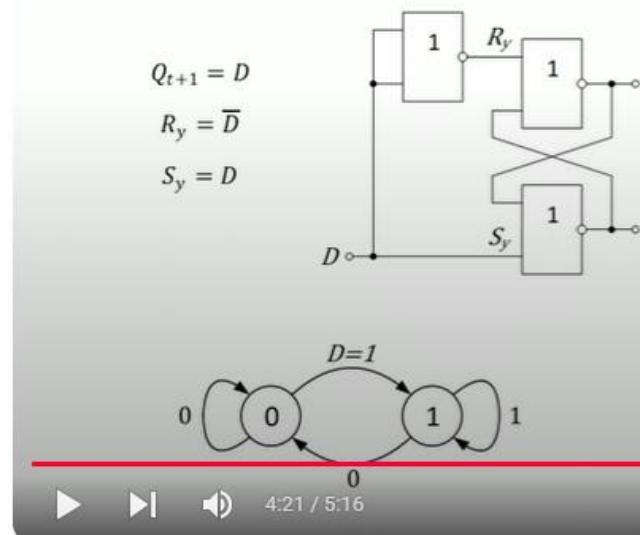


Q_t	D	Q_{t+1}	R_y	S_y	сос
0	0	0	1	0	сбр
0	1	1	0	1	уст
1	0	0	1	0	сбр
1	1	1	0	1	уст

Q_t	Q_{t+1}	R_y	S_y
0	0	*	0
0	1	0	1
1	0	1	0
1	1	0	*

C	D	Q_{t+1}	сос
0	0	Q_t	хр
0	1	Q_t	хр
1	0	0	сбр
1	1	1	уст

Q_t	C	D	Q_{t+1}	R_y	S_y	сос
0	0	0	0	*	0	хр 0
0	0	1	0	*	0	хр 0
0	1	0	0	*	0	сбр
0	1	1	1	0	1	уст
1	0	0	1	0	*	хр 1
1	0	1	1	0	*	хр 1
1	1	0	0	1	0	сбр
1	1	1	1	0	*	уст



Q_{t+1}	CD
0	00 01 11 10
1	11 10 00 01

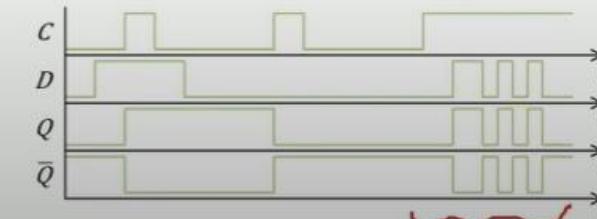
$$Q_{t+1} = Q_{t+1} \cdot \bar{C} \vee C \cdot D$$

R_y	CD
*	00 01 11 10
0	00 01 11 10

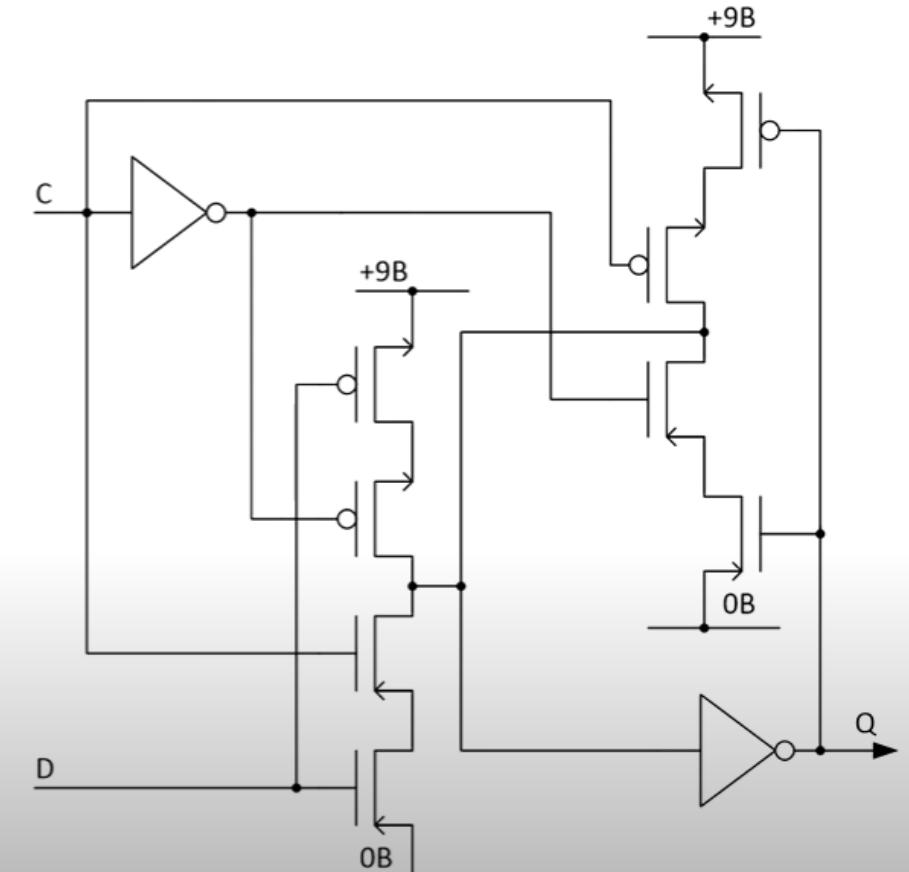
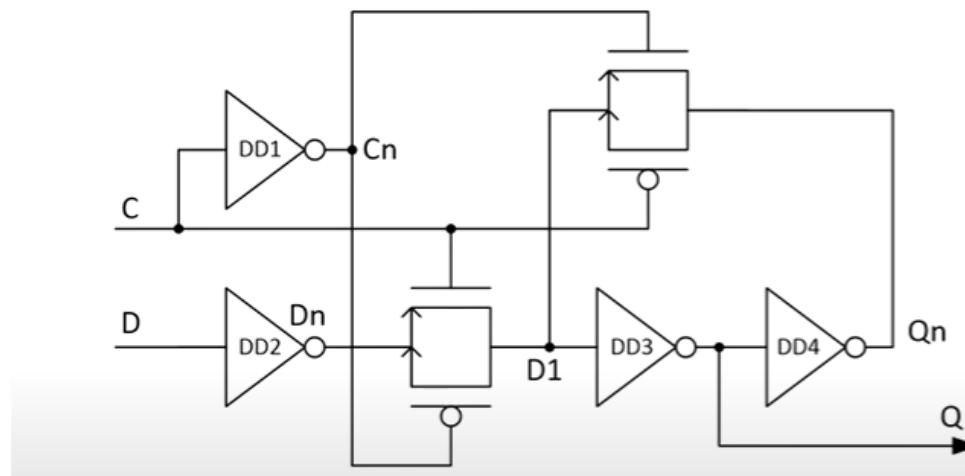
$$R_y = C \cdot \bar{D}$$

S_y	CD
0	00 01 11 10
*	00 01 11 10

$$S_y = C \cdot D$$



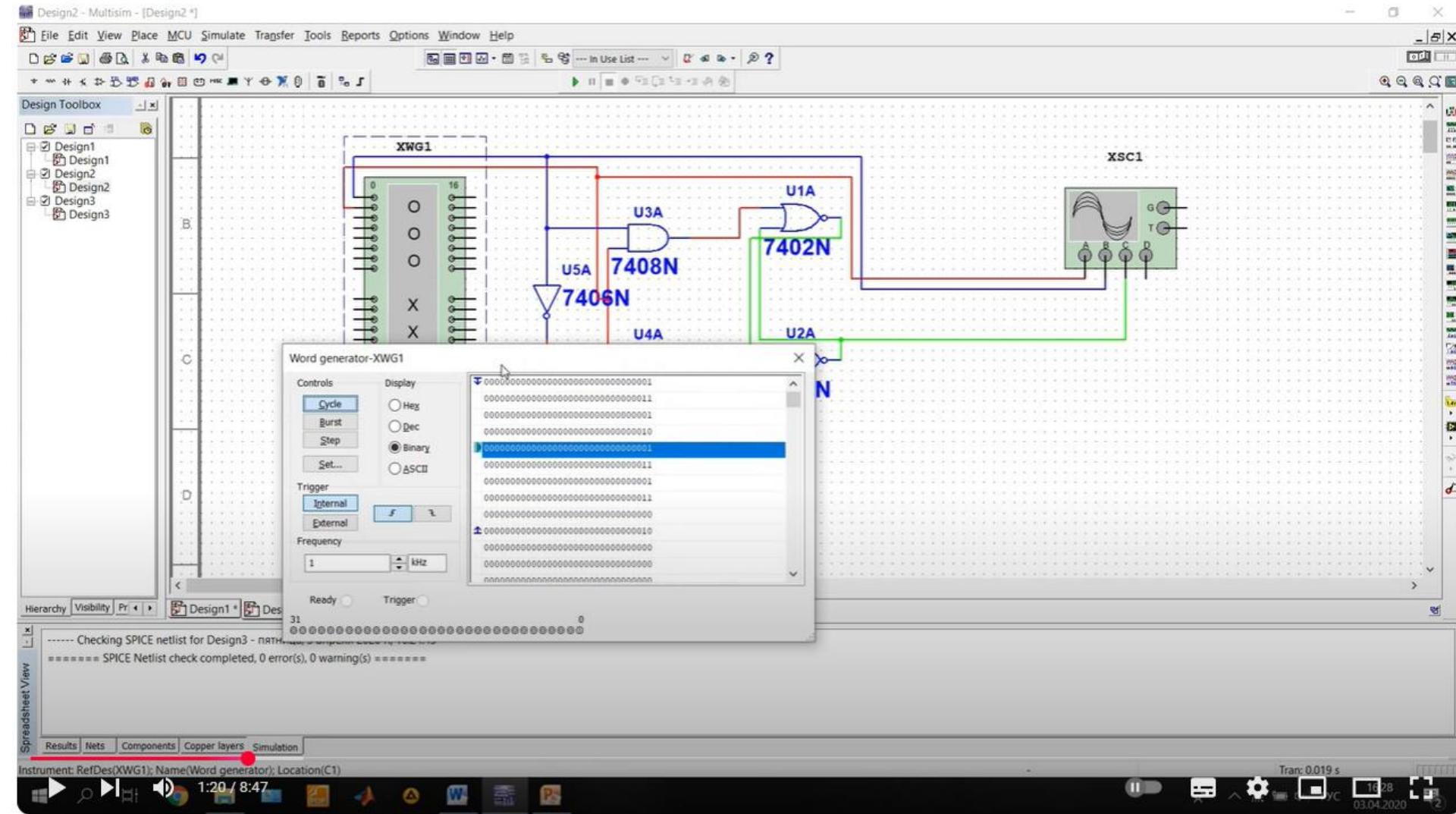
3.1. Синхронный статический D-триггер (2022)
<https://www.youtube.com/watch?v=1sGKOfNeFR4>



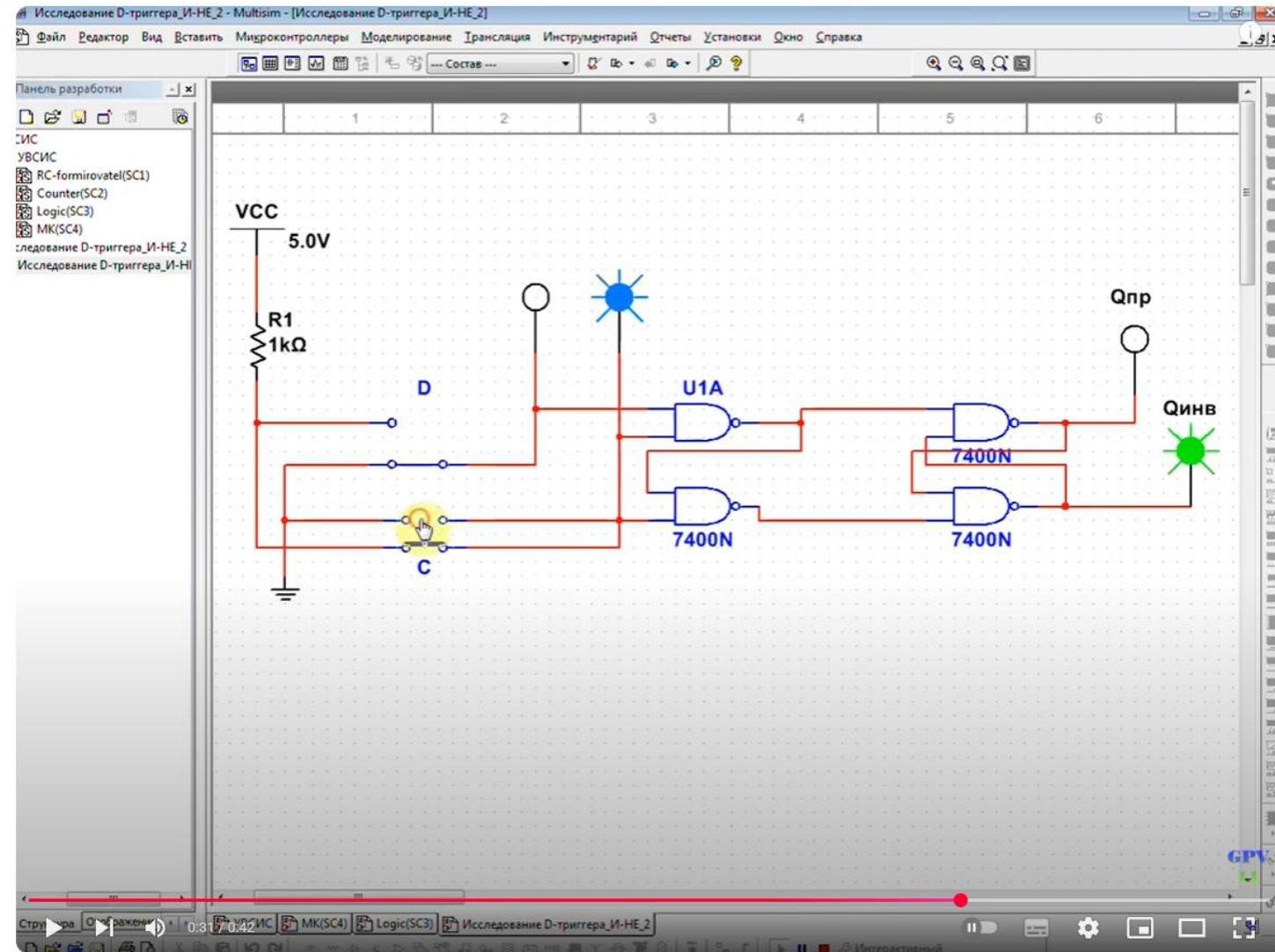
D-триггер на транзисторном уровне. Видео №4 к 6 главе книги.(2021)
<https://www.youtube.com/watch?v=kPP9ghnKgZE>



be 8:47



Лабораторная работа №10 "D-триггер. T-триггер" (2020)
<https://www.youtube.com/watch?v=Xpdm9ehJbmk>



Исследование работы D-триггера на элементах И-НЕ в Multisim 14.1 (2021)
<https://www.youtube.com/watch?v=SgPYI5LrTk8>

Applications Places Document Viewer 20 °C en Fri May 24 22:18 • 100 %

5 of 7 Doc.pdf 172.8% ▾

Introduction 3
Triggers clas... 4
Synchronous ... 5
Theory 5
Descrip... 5
Schema... 5
Truth ta... 5
Function... 5
Practice 6
Circuit ... 6
Results 6
Conclusion 7

Synchronous Static D-Trigger

Theory

Description

The extended synchronous static RS-Trigger. Usually it contains only two external inputs: D (data) and C (clock). But in practice it can contain also external R and S inputs. Its D input connected to the internal R and S inputs in a straight and inverse modes respectively.

Schematic diagram



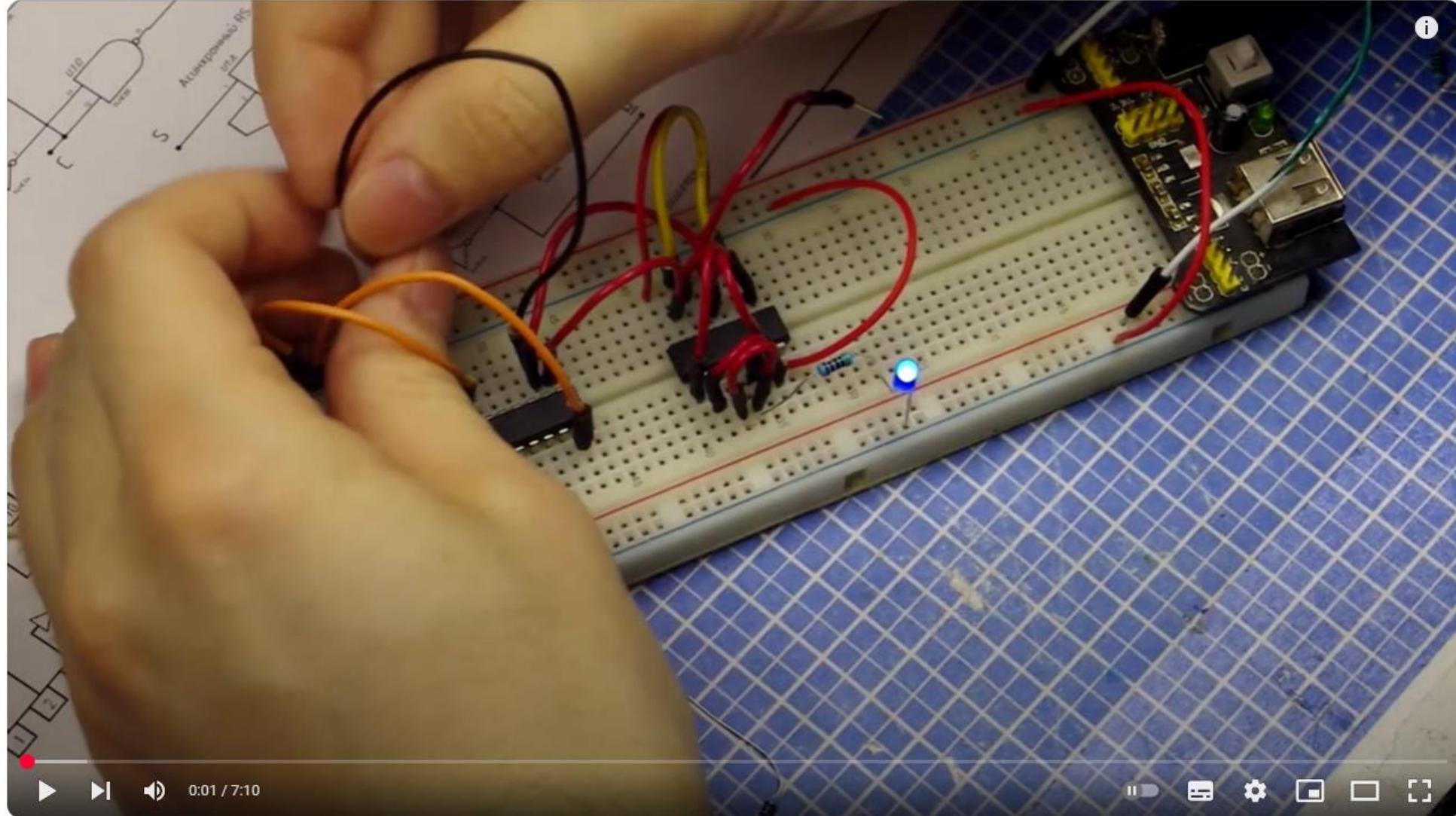
Truth table

C	D	Q	\bar{Q}
0	0	Q^{-1}	\bar{Q}^{-1}
0	1	Q^{-1}	\bar{Q}^{-1}
1	0	Q	\bar{Q}
1	1	Q	\bar{Q}

3:54 / 9:23

0100. D-Триггер (2019)

<https://www.youtube.com/watch?v=rWD2j9wtGmY>



Вся правда о RS-триггере, D-триггер, практика применения RS-триггера (2019)
<https://www.youtube.com/watch?v=Pi4PUSHxGC4>

D Flip Flop

Truth Table for SR FF

CLK	S	R	
0	X	X	
1	0	0	
1	0	1	$Q_n \rightarrow 1$
1	1	1	

$\begin{cases} S=1, R=0 \\ S=0, R=1 \end{cases} \Rightarrow Q_n = 1 \text{ Set}$

$\begin{cases} S=1, R=0 \\ S=0, R=1 \end{cases} \Rightarrow Q_n = 0 \text{ Reset}$

DFF

CLK	D	Q_{n+1}
0	X	Q_n
1	0	0
1	1	1

5:54 / 6:38

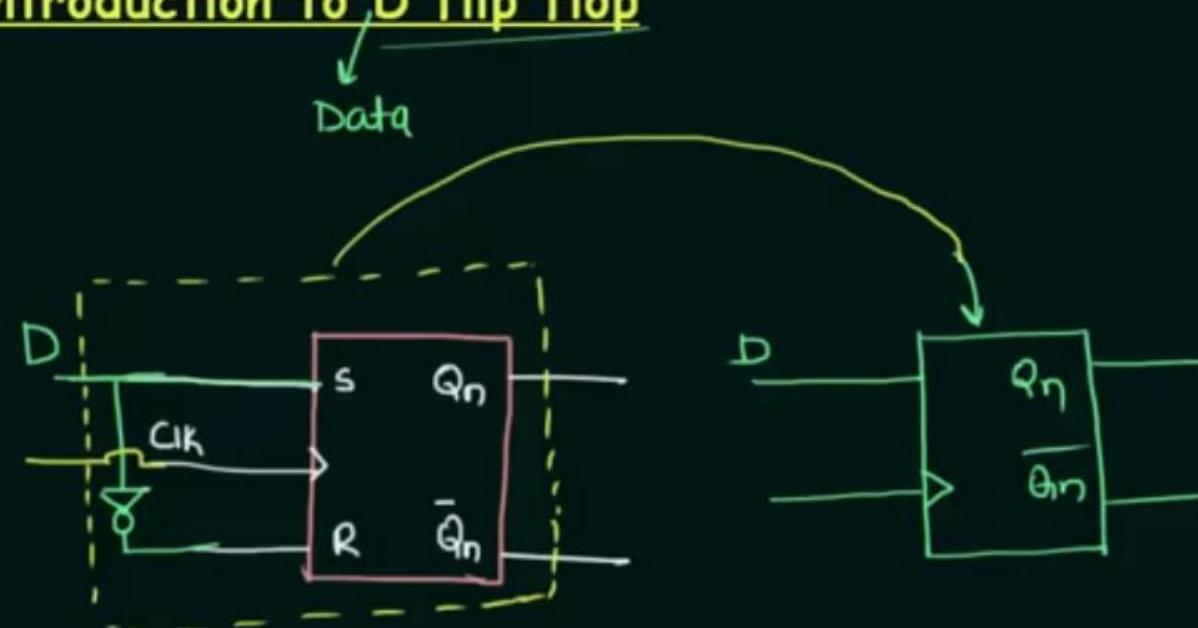
9. D Flip Flop | Sequential Circuits | Tech Gurukul by Dinesh Arya (2018)
 9. D-триггер | Последовательные схемы | Tech Gurukul от Динеша Ары (2018)
https://www.youtube.com/watch?v=_SwesRcW79s

Introduction to D flip flop

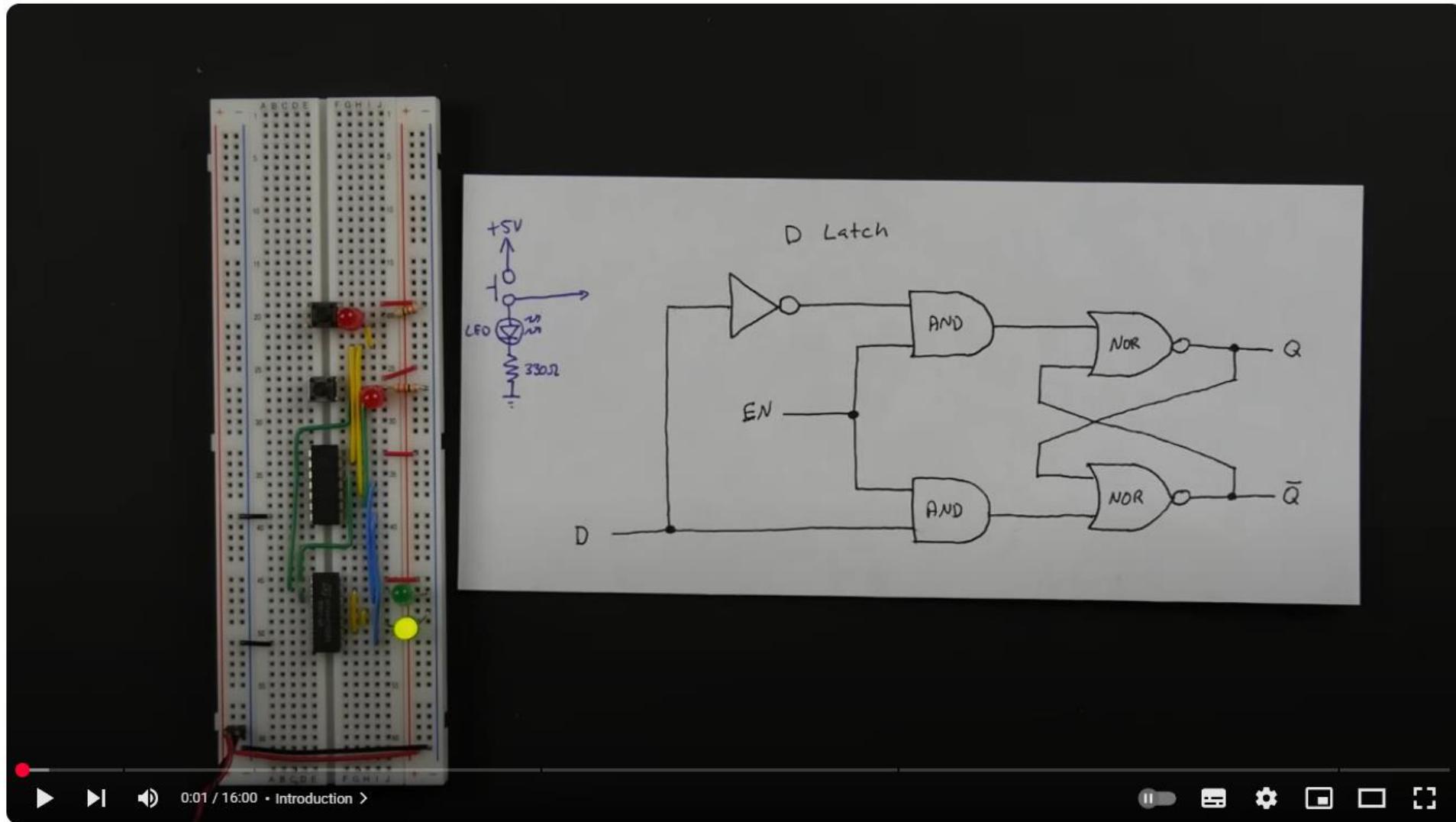
T.T for SR Flip Flop :-

Clk	S	R	Q_{n+1}
0	x	x	Q_n
1	0	0	Q_n
1	0	1	0
1	1	0	1
1	1	1	invalid

Memory

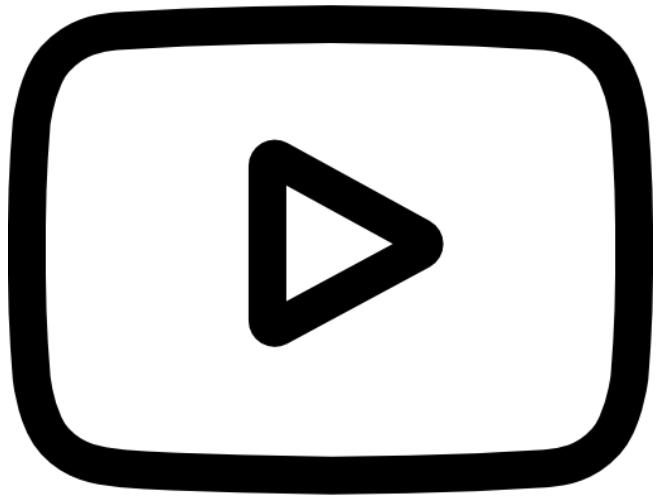


T.T. for D ff :-



D flip-flop (2016)
D триггер (2016)

https://www.youtube.com/watch?v=YW_GkUguMM



Т-триггер



Асинхронный T-триггер

Асинхронный T-триггер (счетный триггер) функционирует в соответствии таблицей переходов:

T_n	Q_n	Q_{n+1}	R	S
0	0	0	x	0
0	1	1	0	x
1	0	1	0	1
1	1	0	1	0

Каждым входным сигналом триггер переключается в противоположное состояние. (Входной сигнал – $T=1$).

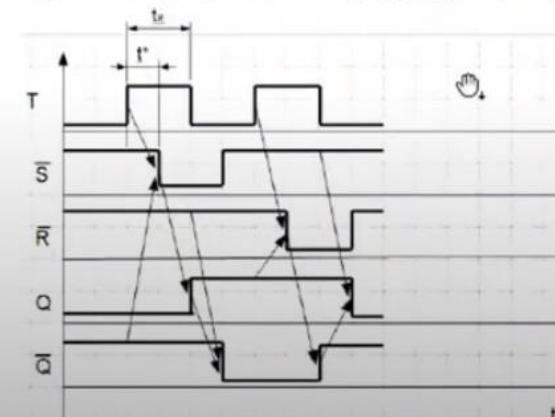
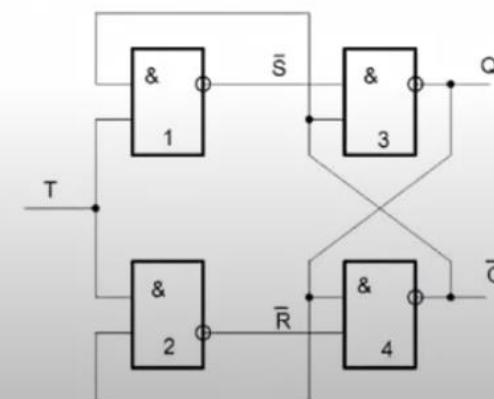
$$Q_{n+1} = \bar{T}_n Q_n + T_n \bar{Q}_n = T_n \oplus Q_n$$

Элемент сложение по модулю 2 не обладает памятью. Будем синтезировать схему на основе асинхронного RS-триггера.

Функции для управляющих сигналов $R = T_n Q_n$, $S = T_n \bar{Q}_n$

Для реализации в базисе И=НЕ: $\bar{R} = \overline{T_n Q_n}$, $\bar{S} = \overline{T_n \bar{Q}_n}$

Временная диаграмма иллюстрирует работу T-триггера.



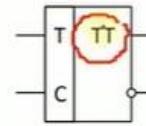
t_u – минимальная длительность импульса.

t^* – средняя задержка распространения сигнала на логическом элементе

РК6. Схемотехника. Асинхронный T-триггер (2020)
<https://www.youtube.com/watch?v=fsEDUgDMYJE>

Асинхронный/синхронный
статический T-триггер

T	Q_{t+1}	соч
0	Q_t	хр
1	\bar{Q}_t	счет

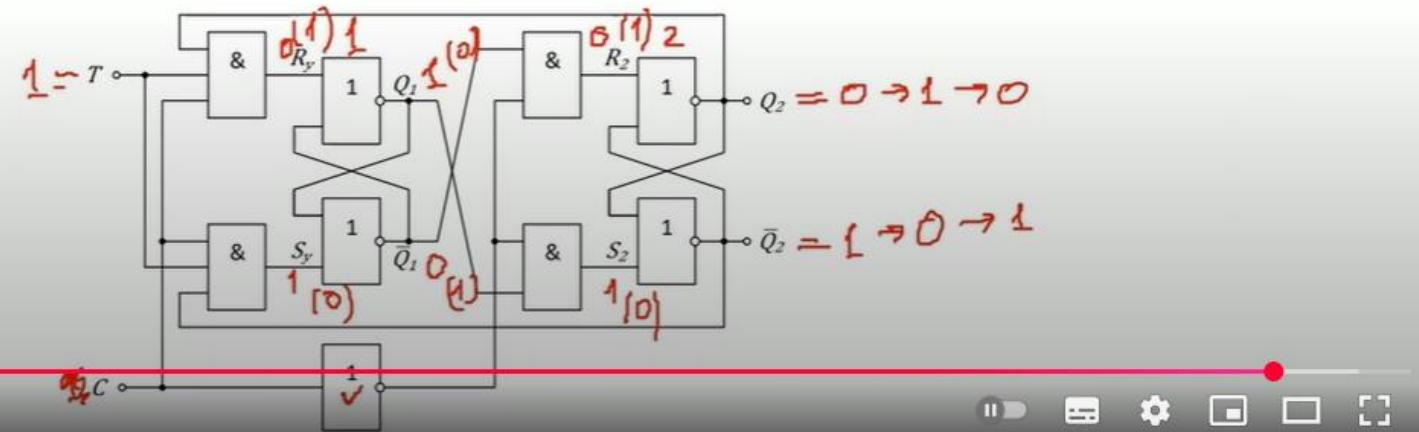
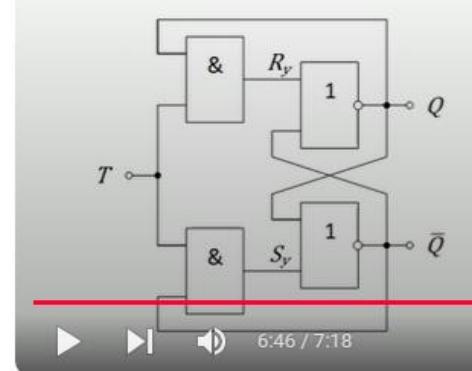


C	T	Q_{t+1}	соч
0	0	Q_t	хр
0	1	Q_t	хр
1	0	Q_t	хр
1	1	\bar{Q}_t	счет

Q_t	T	Q_{t+1}	R_y	S_y	соч
0	0	0	*	0	хр 0
0	1	1	0	1	счет
1	0	1	0	*	хр 1
1	1	0	1	0	счет

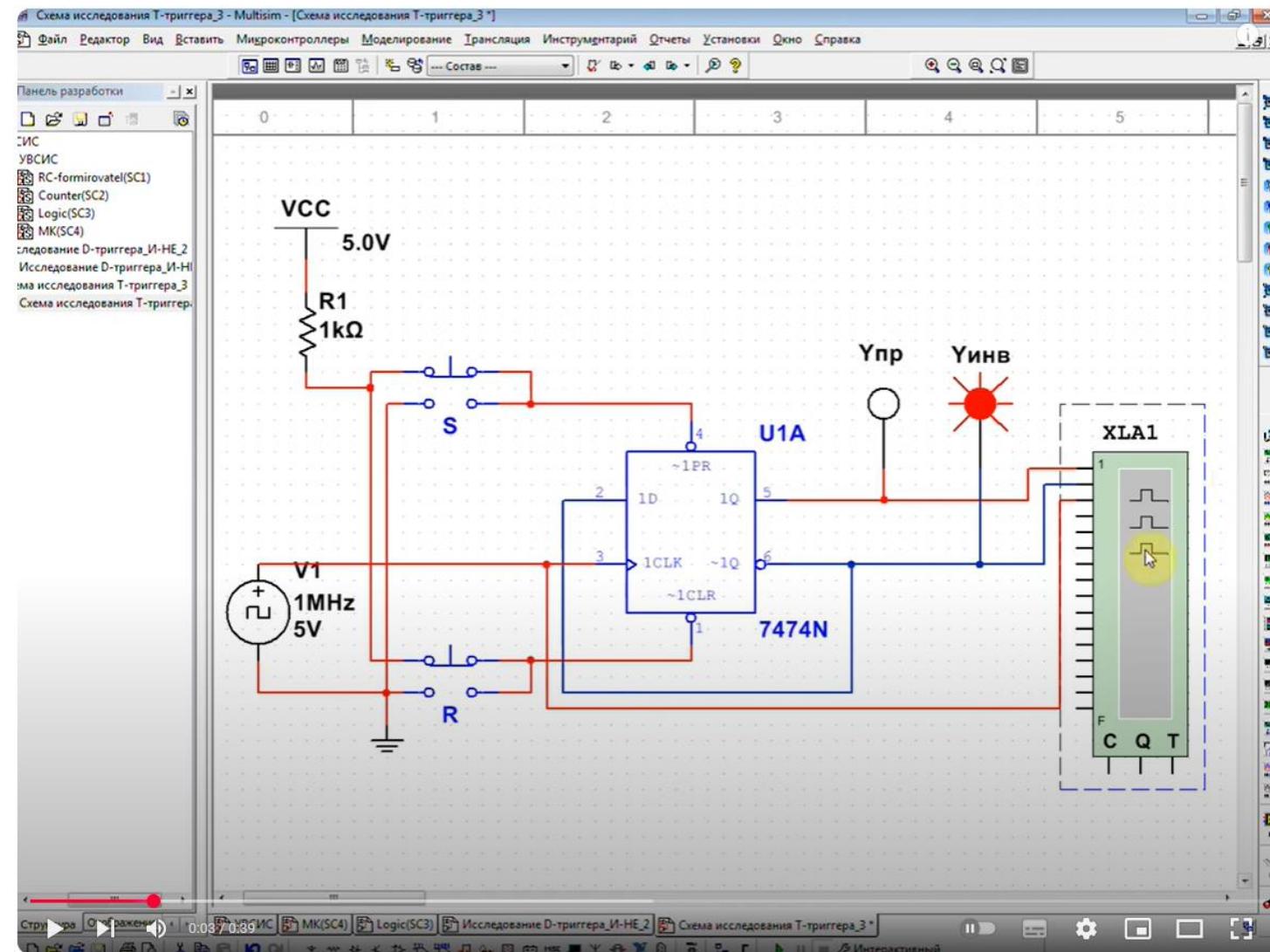
$$Q_{t+1} = \bar{Q}_t \cdot T \vee Q_t \cdot \bar{T} \quad R_y = Q_t \cdot T \\ S_y = \bar{Q}_t \cdot T$$

Q_t	C	T	Q_{t+1}	R_y	S_y	соч
0	0	0	0	*	0	хр 0
0	0	1	0	*	0	хр 0
0	1	0	0	*	0	хр 0
0	1	1	1	0	1	счет
1	0	0	1	0	*	хр 1
1	0	1	1	0	*	хр 1
1	1	0	1	0	*	хр 1
1	1	1	0	1	0	счет



4.1. Синхронный статический T-триггер (2022)

<https://www.youtube.com/watch?v=nuiDQ29UiY>

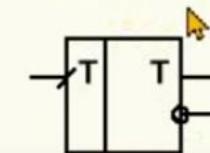
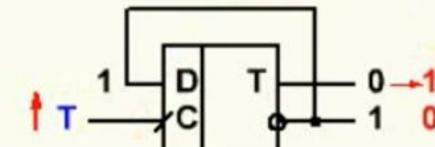


Исследование работы Т-триггера в Multisim 14.1 (2021)
<https://www.youtube.com/watch?v=kTYQRYdZFrE>

Т-триггеры

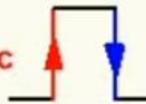
T (toggle - переключатель) - **счетный** вход

1. Синхронные D-триггеры



D - **информационный** вход
активные сигналы - 0 и 1

C или CLK - **синхронизирующий (разрешающий)** вход
динамический
активный сигнал - **импульс**



Вideoурок 4.12. Т-триггеры. (2024)
<https://www.youtube.com/watch?v=lr3HOA5b2aw>

Синхронный Т-триггер



4.1. Синхронный статический Т-триггер (2022)
<https://www.youtube.com/watch?v=nuiDQ29UiY>

T Flip Flop

JK FF

$J = K$

$T - FF$

$J = K$

$T - FF$

CLK	J	K	Q_{n+1}
0	X	X	Q_n
1	0	0	\bar{Q}_n
1	0	1	0
1	1	0	1
1	1	1	\bar{Q}_n

T

Q_n

\bar{Q}_n

$J = \bar{K}$

$K = 0$

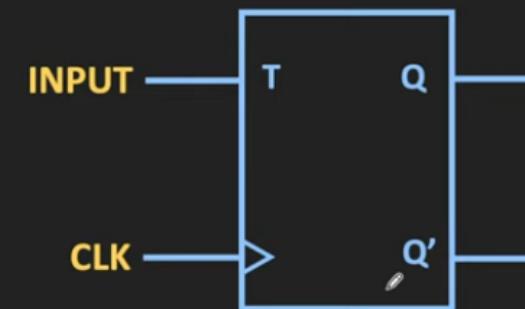
$J = 1$

$K = 1$

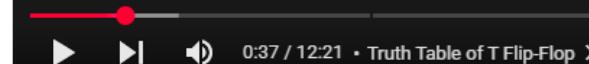
15. T Flip Flop | Sequential Circuits | Tech Gurukul by Dinesh Arya (2019)
 15. Триггер | Последовательные схемы | Tech Gurukul от Динеша Ары (2019)
https://www.youtube.com/watch?v=1wld_qMB1Yw

T Flip-Flop

i

Symbol**Truth Table**

CLK	T	Q_{n+1}
↑	0	Q_n
↑	1	Q_n'



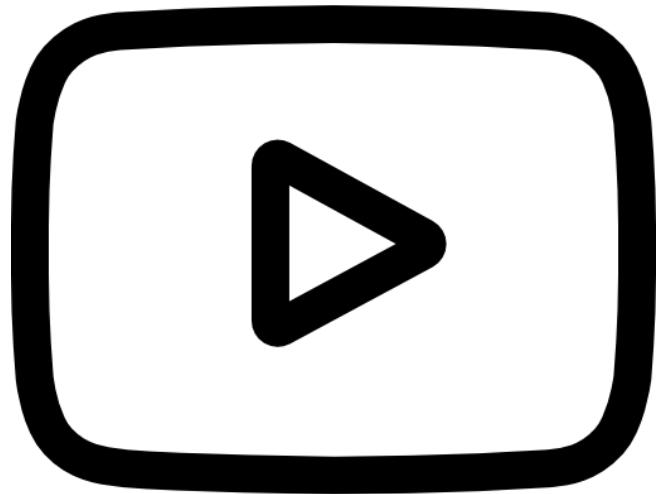
Субтитры (с)



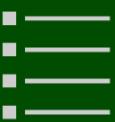
ALL ABOUT ELECTRONICS

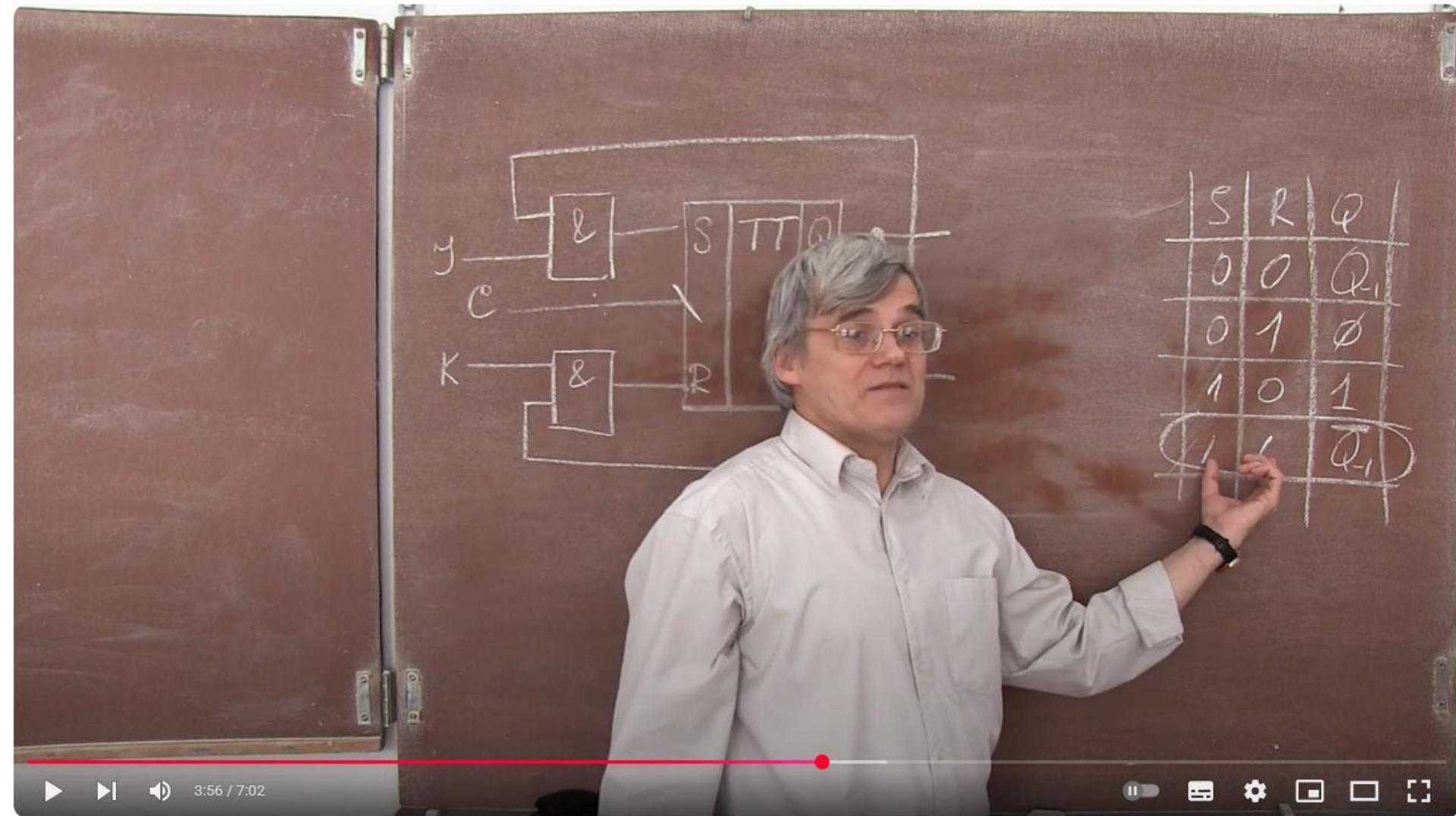


T Flip-Flop Explained | Circuit Diagram, Excitation Table and Characteristic Equation (2022)
Описание Т-триггера | Принципиальная схема, таблица возбуждения и характеристическое уравнение (2022)
<https://www.youtube.com/watch?v=hw2rWOlg9x4>



JK-триггер





Лекция 134. JK-Триггер (2014)
<https://www.youtube.com/watch?v=8ETkClVC0fg>

Синхронные двухступенчатые триггеры с запрещающими связями

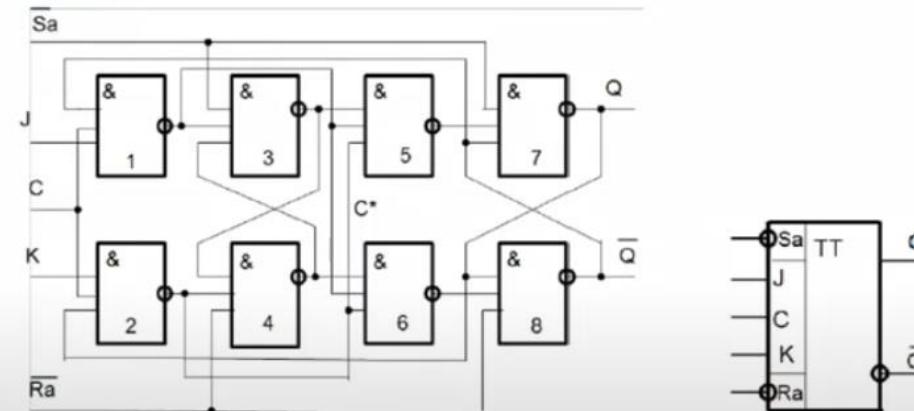
Синхронный двухступенчатый JK-триггер

В JK-триггере входы J и K соответствуют входам S и R RS-триггера. Отличие JK-триггера от RS-триггера состоит в том, что комбинация сигналов $S=R=C=1$ запрещена для RS-триггера, а JK-триггер при комбинации сигналов $J=K=C=1$ переключается в противоположное состояние, т.е. работает как T-триггер.

Схема синхронного двухступенчатого JK-триггера с запрещающими связями.

ЛЭ 1...4 – M-триггер (1-я ступень)

ЛЭ 5...8 – S-триггер (2-я ступень)



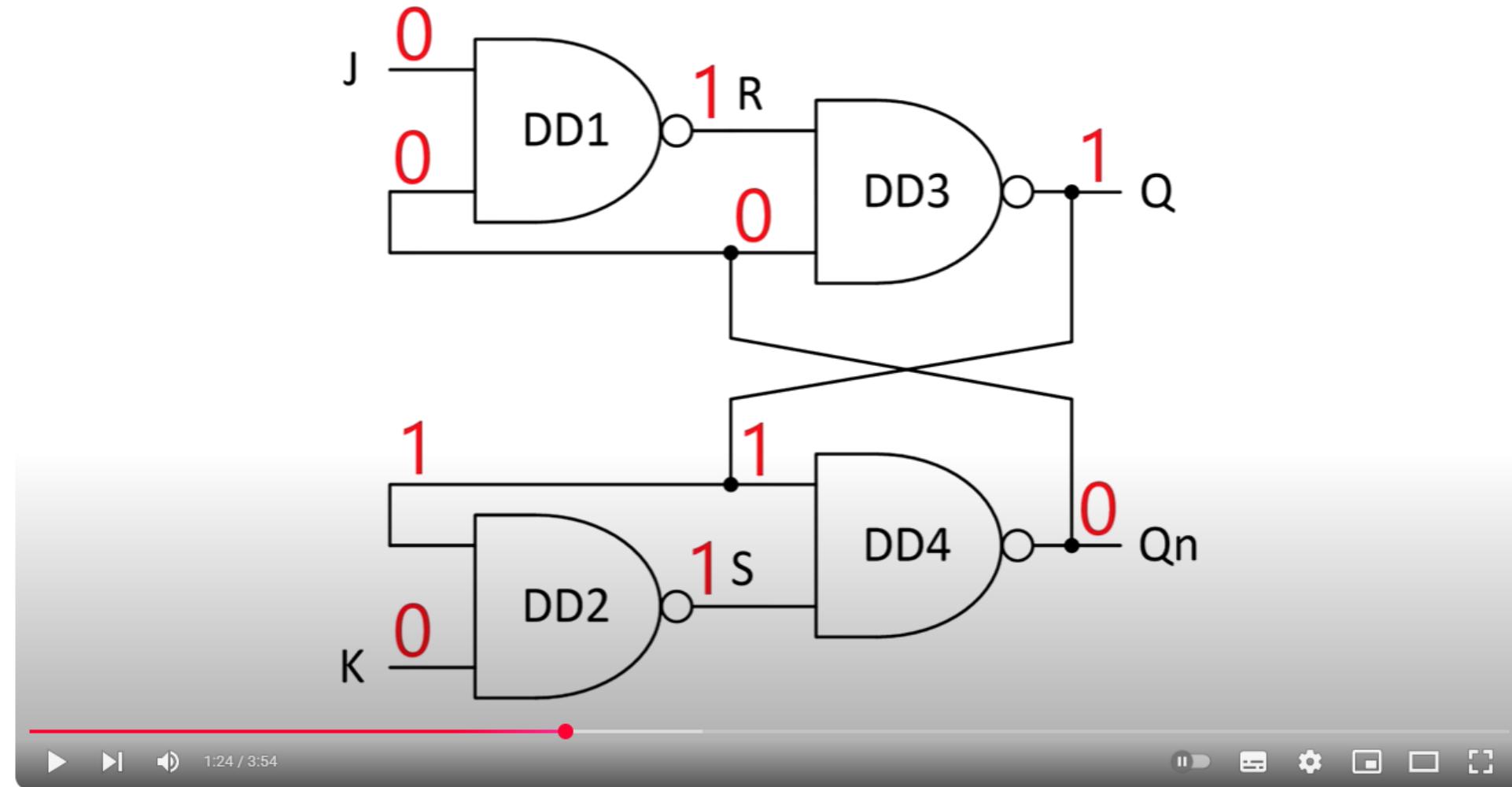
Проанализируем работу триггера.

При $C=1$ и любых сочетаниях сигналов J и K (кроме $J=K=0$) в состоянии логического нуля находится либо ЛЭ1 либо ЛЭ2. Ноль с выхода ЛЭ1 или ЛЭ2 осуществляет запись информации со входа J или K в основной триггер и запрещает перезапись информации во вспомогательный триггер через ЛЭ5 и ЛЭ6. Поэтому связи ЛЭ1 и ЛЭ2 с элементами ЛЭ5 и ЛЭ6 называют запрещающими.

После окончания синхроимпульса ($C=0$) на выходах ЛЭ1 и ЛЭ2 появятся сигналы логической единицы разрешающие перезапись информации во вспомогательный триггер через ЛЭ5 и ЛЭ6.



РК6. Схемотехника. Двухступенчатый JK-триггер с запрещающими связями (2020) <https://www.youtube.com/watch?v=oHzAUWCXed4>

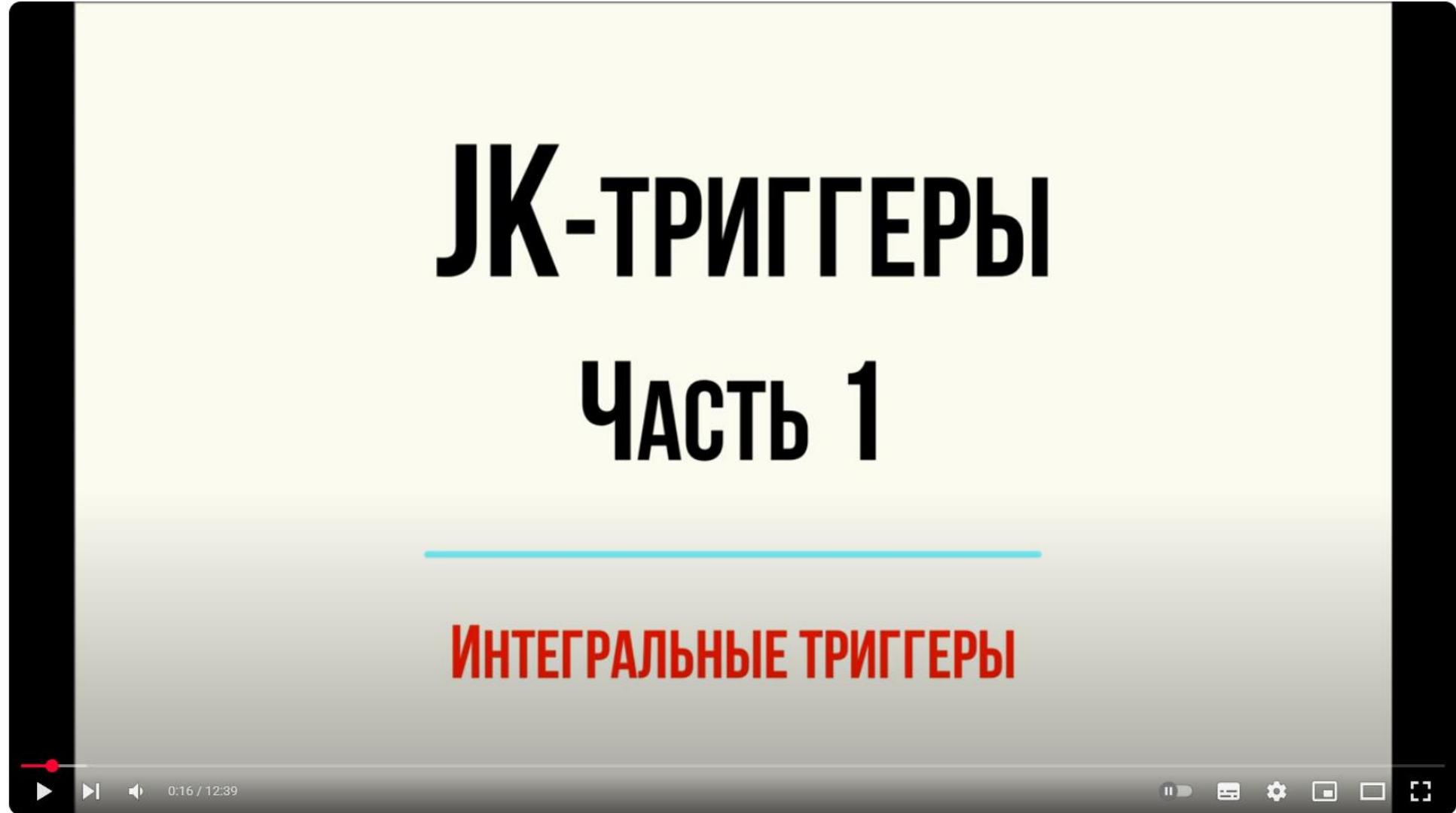


JK-триггер. Видео №3 к 6 главе книги. Ошибочные и рабочие схемы. (2021)
<https://www.youtube.com/watch?v=jsu5xLEc-uU>

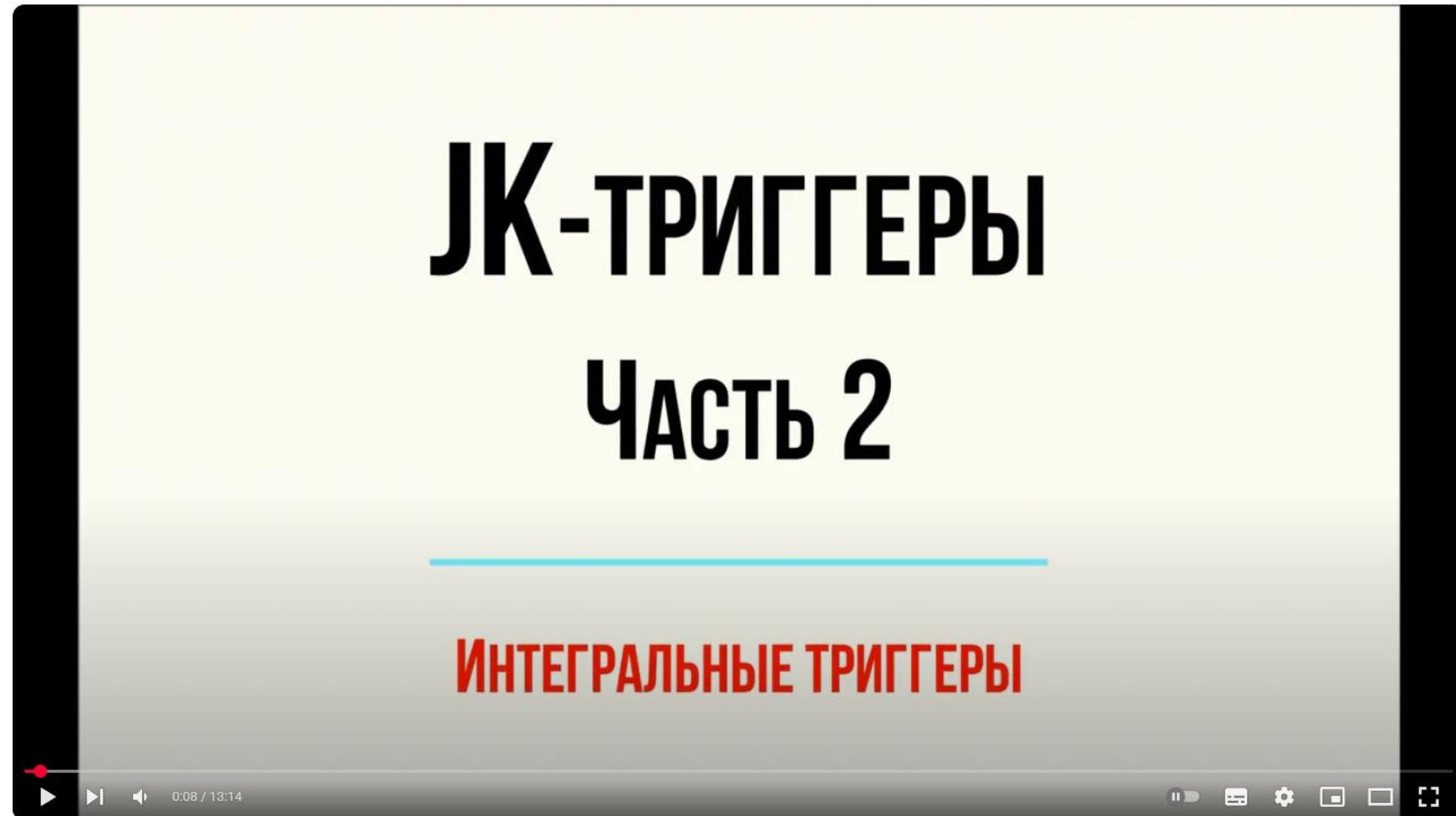
JK-ТРИГГЕРЫ

ЧАСТЬ 1

ИНТЕГРАЛЬНЫЕ ТРИГГЕРЫ



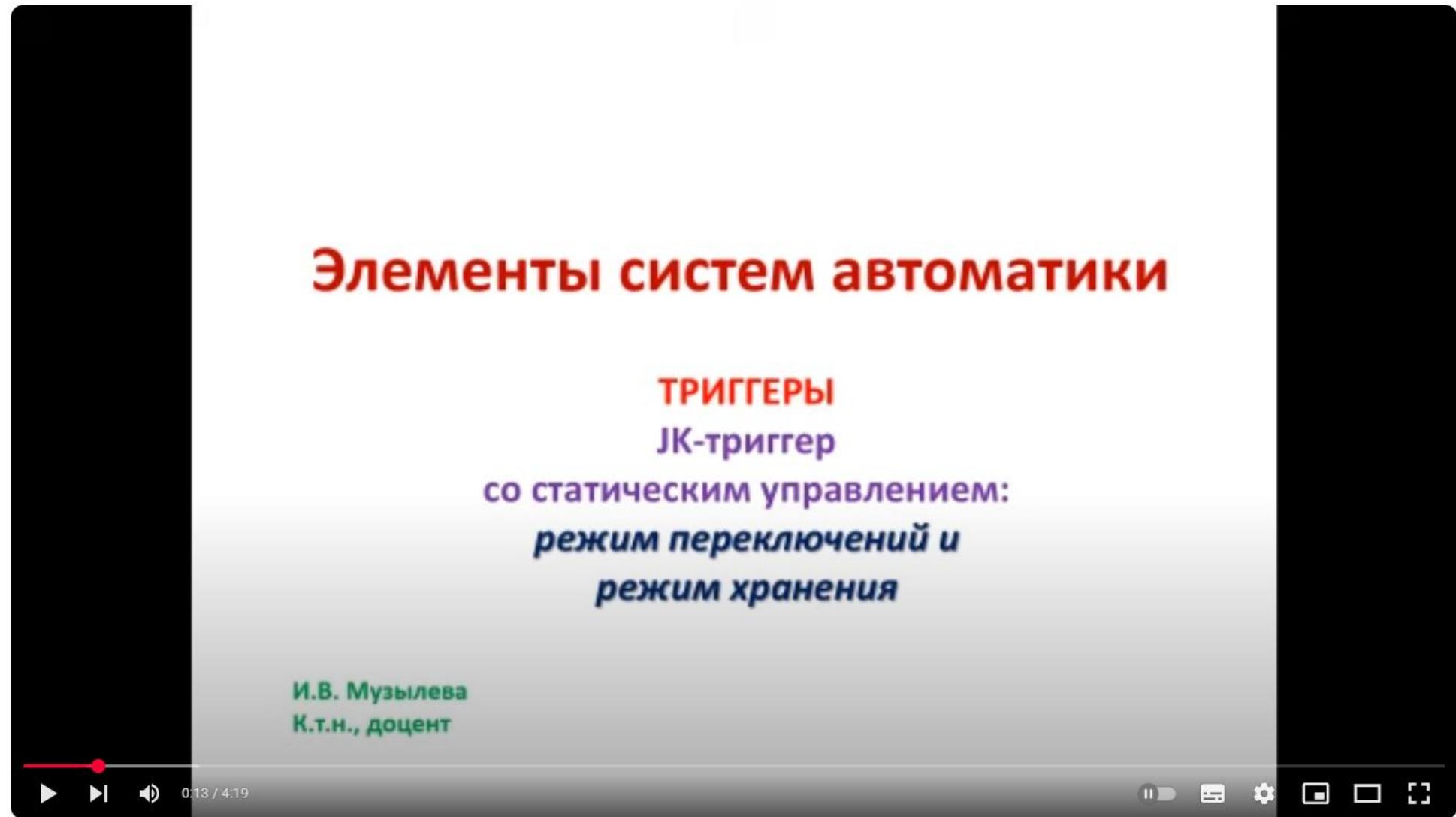
Вideoурок 4.7. JK-триггеры. Часть 1. (2024)
https://www.youtube.com/watch?v=_xRYo4a3MeM



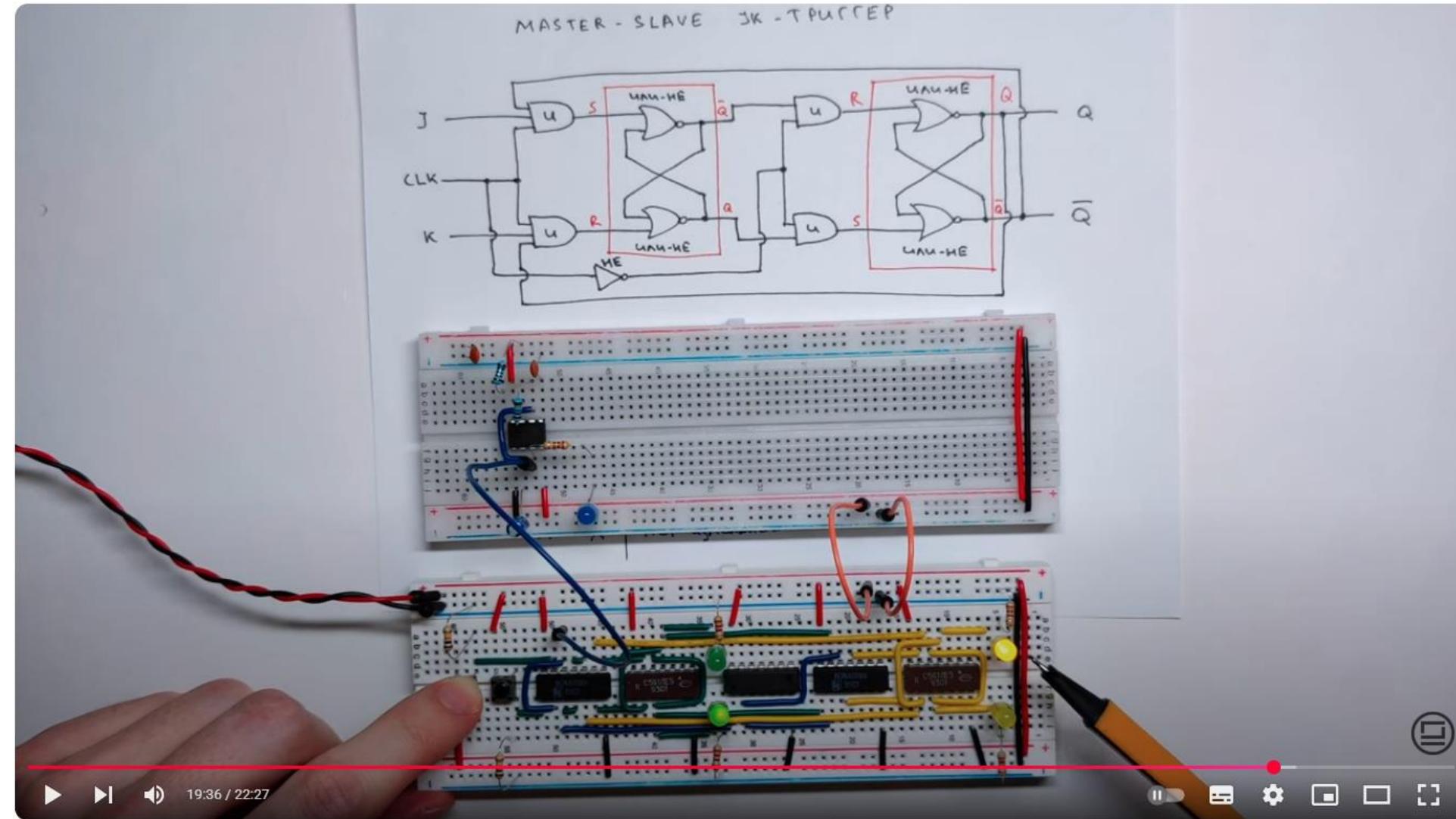
Видеоурок 4.8. JK-триггеры. Часть 2. (2024)
<https://www.youtube.com/watch?v=x8RzoohUZBc>



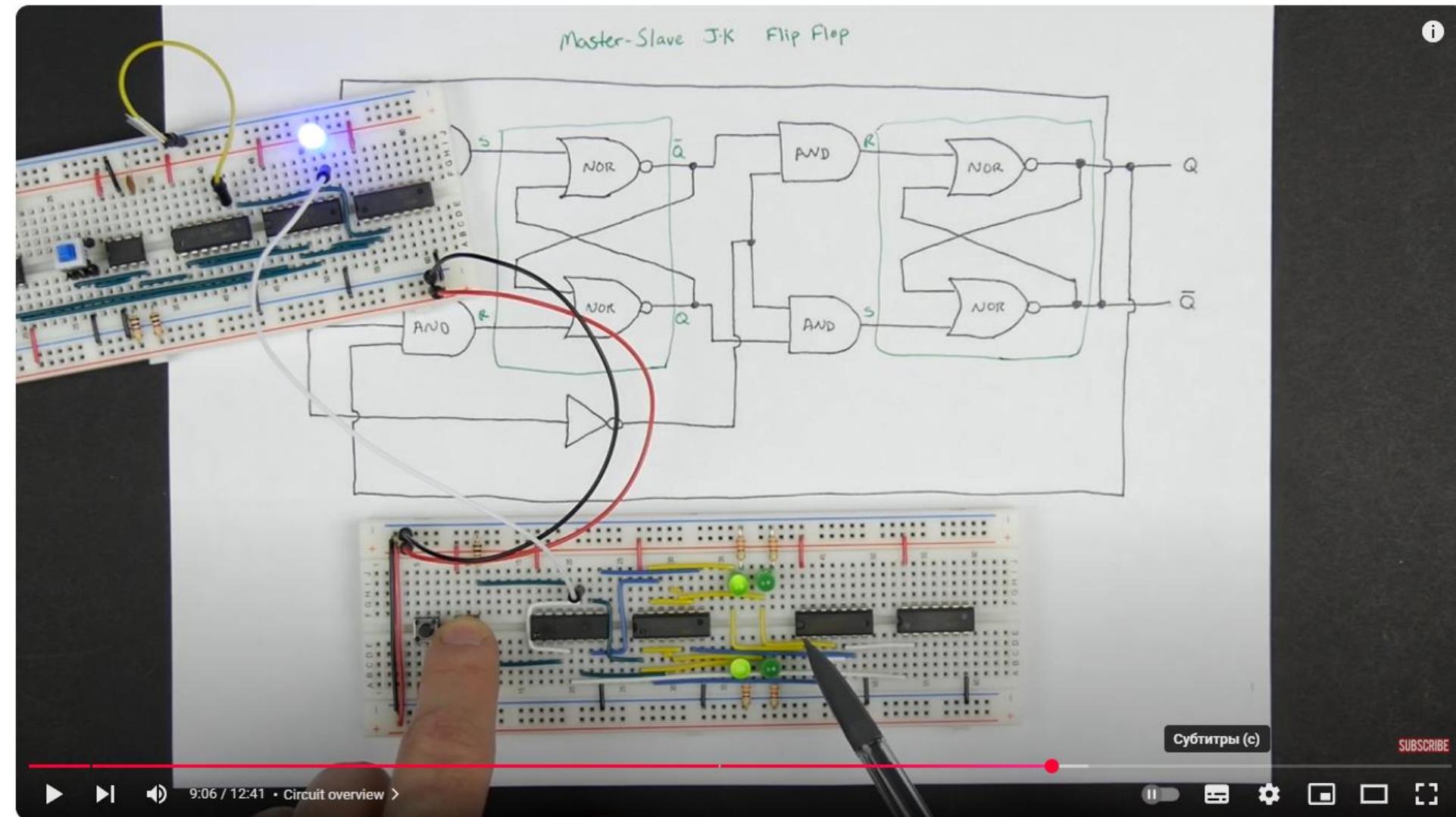
Видеоурок 4.9. JK-триггеры. Часть 3. (2024)
<https://www.youtube.com/watch?v=gtegCM-LRKQ>



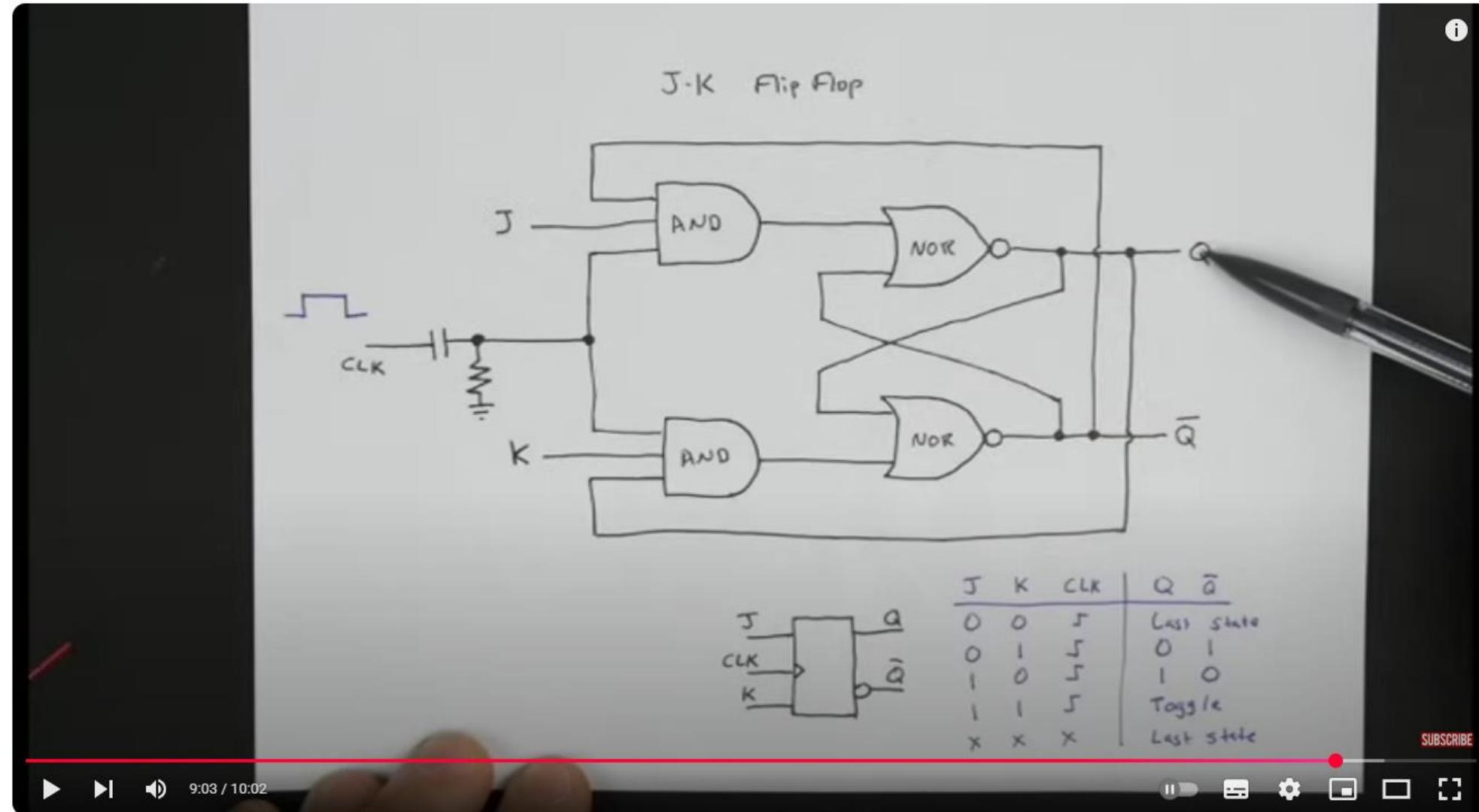
JK статический переключение и хранение (2023)
<https://www.youtube.com/watch?v=xtykTYEMuiQ>



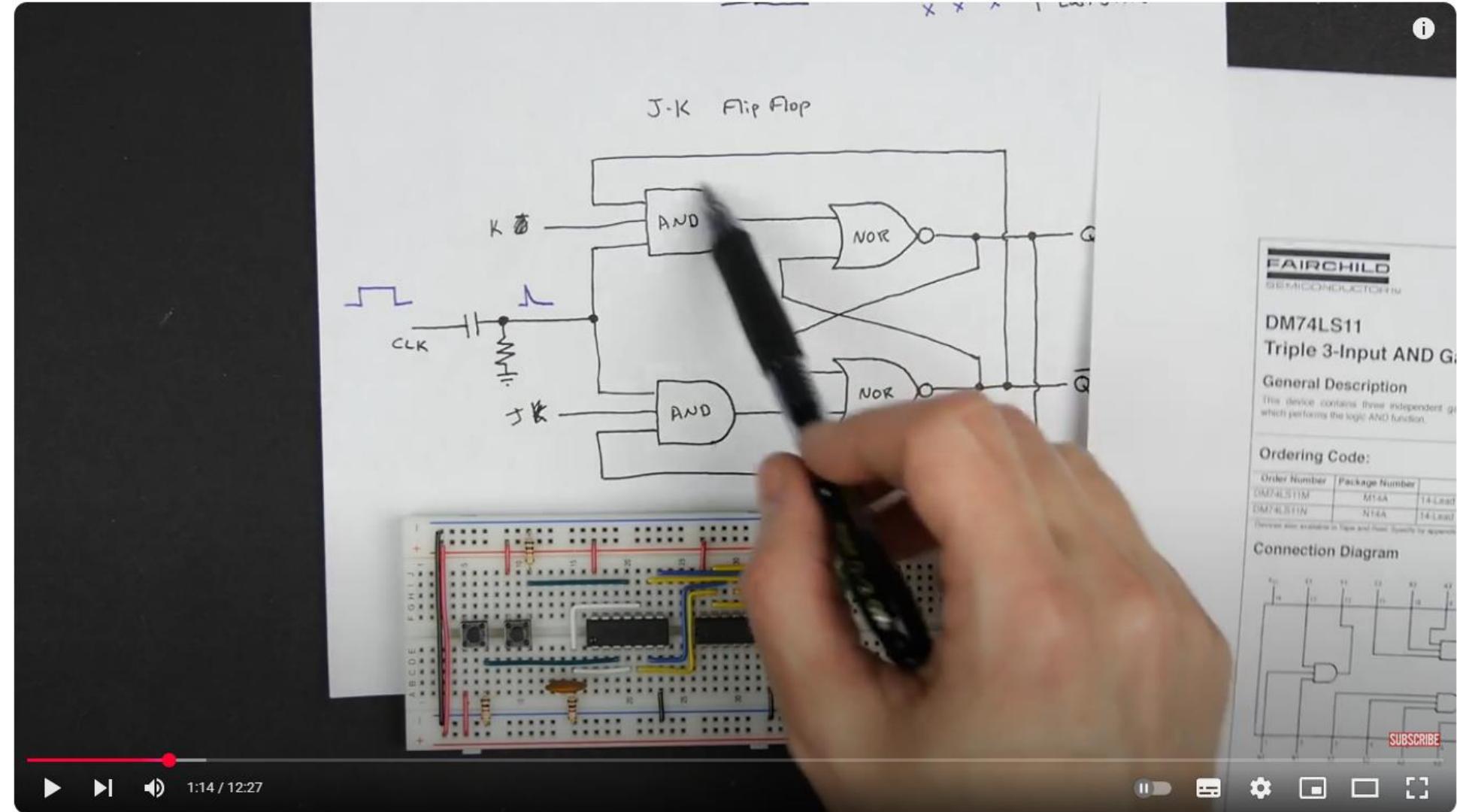
Разбираем устройство JK-триггера, его отличия от RS-триггера и собираем на макетной плате (2020) <https://www.youtube.com/watch?v=4Y123et2xRM>



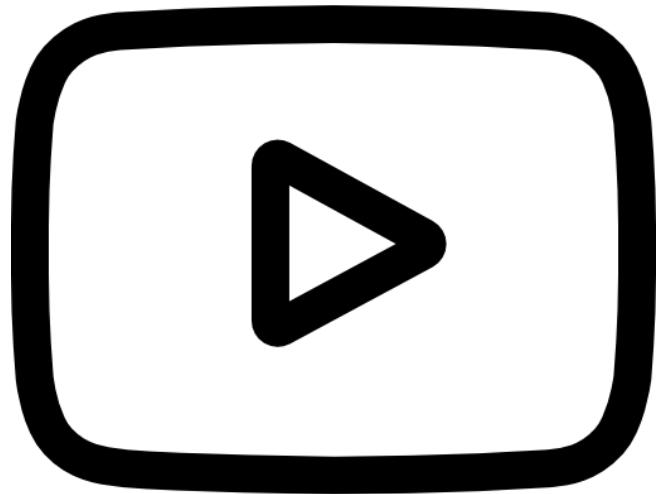
Master-slave JK flip-flop (2017)
Ведущий-ведомый JK-триггер (2017)
<https://www.youtube.com/watch?v=rXHSB5w7CyE>



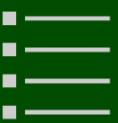
JK flip-flop (2016)
JK-Триггер (2016)
https://www.youtube.com/watch?v=F1OC5e7Tn_o

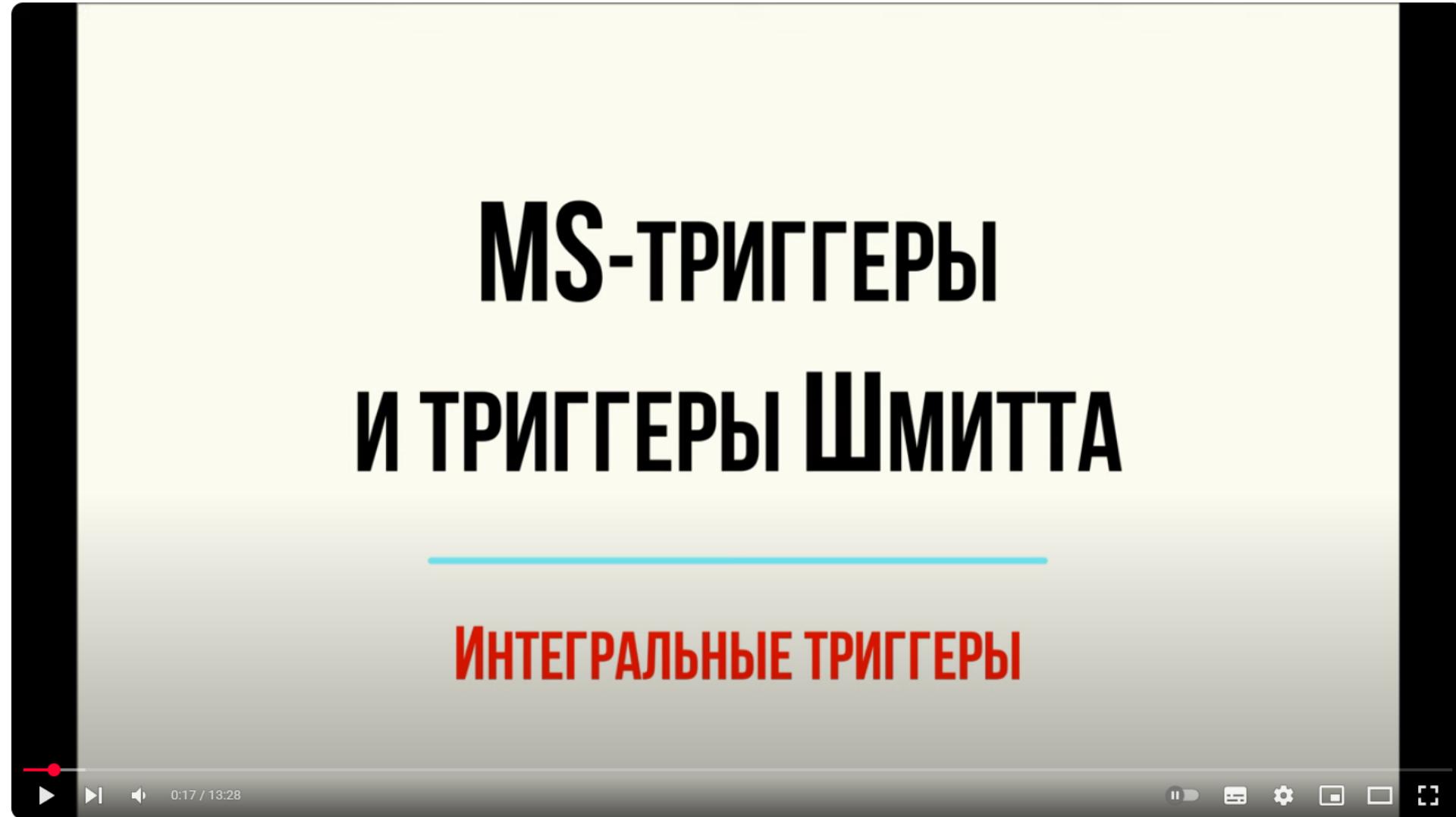


JK flip-flop racing (2017)
<https://www.youtube.com/watch?v=st3mUEub99E>

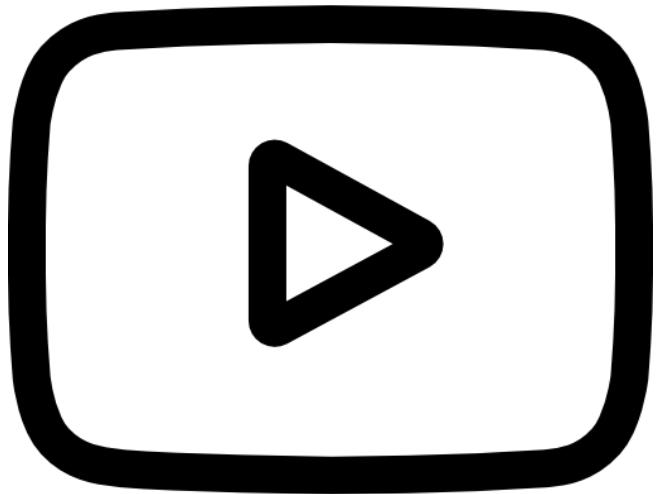


MS-триггер

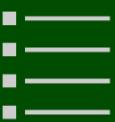




Вideoурок 4.13. MS-триггеры и триггеры Шмитта (2024)
<https://www.youtube.com/watch?v=UYBmGWVhVZw>



Регистры



Функциональные узлы ЭВМ

Узел – функциональная часть ЭВМ, предназначенная для выполнения операций над словом или частью слова.

Классификация узлов

1. По назначению: дешифраторы, мультиплексоры, демультиплексоры, шифраторы, схемы сравнения кодов, преобразователи кодов, схемы определения четности кодов, регистры, счетчики, сумматоры и др.
2. По принципу логического функционирования
 - комбинационного типа;
 - накапливающего типа.

Регистры

Регистр – функциональный узел ЭВМ, предназначенный для ввода, хранения, преобразования и вывода двоичного слова или его части.

Регистр состоит из запоминающих элементов (триггеров) и логических элементов.

Регистры выполняют ряд микроопераций над словами, которые можно разделить на четыре группы.

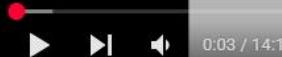
1 группа – микрооперации, связанные с приемом слова (установка в нулевое или единичное состояние, прием слова в прямом или обратном коде);

2 группа – выдача слова в прямом или обратном коде;

3 группа – выполнение поразрядных логических операций (сложение по модулю 2, логическое сложение, логическое умножение);

4 группа – сдвиг слов (влево или вправо на то или иное число разрядов), преобразование последовательного кода в параллельный и обратно.

Регистры классифицируются по способам приема и выдачи информации и способам представления информации на входе и выходе.

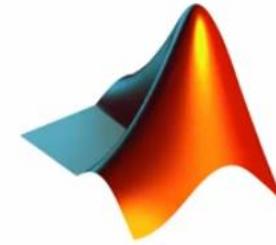


PK6. Схемотехника. Регистры (2020)
<https://www.youtube.com/watch?v=fq9pOH6R0rM>



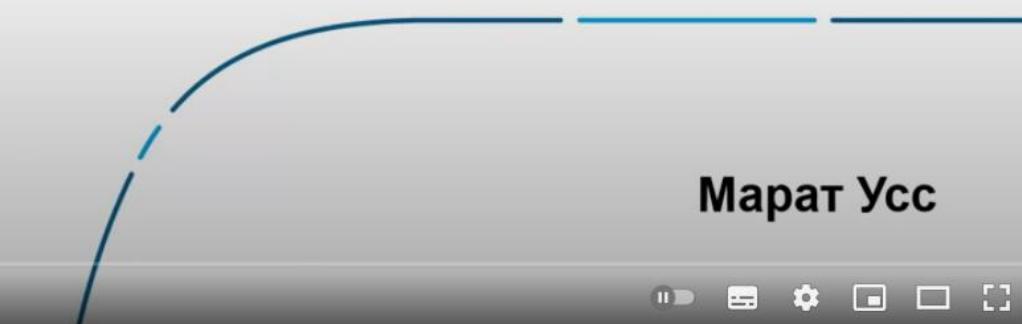
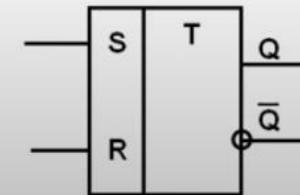
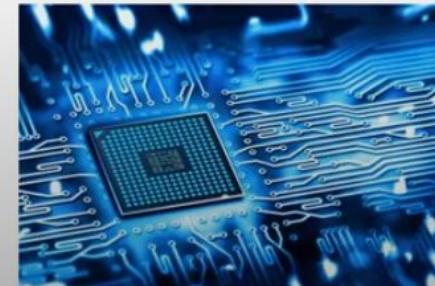
ЭКСПОНЕНТА
ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ

Engee - переходите на сайт [i](#)



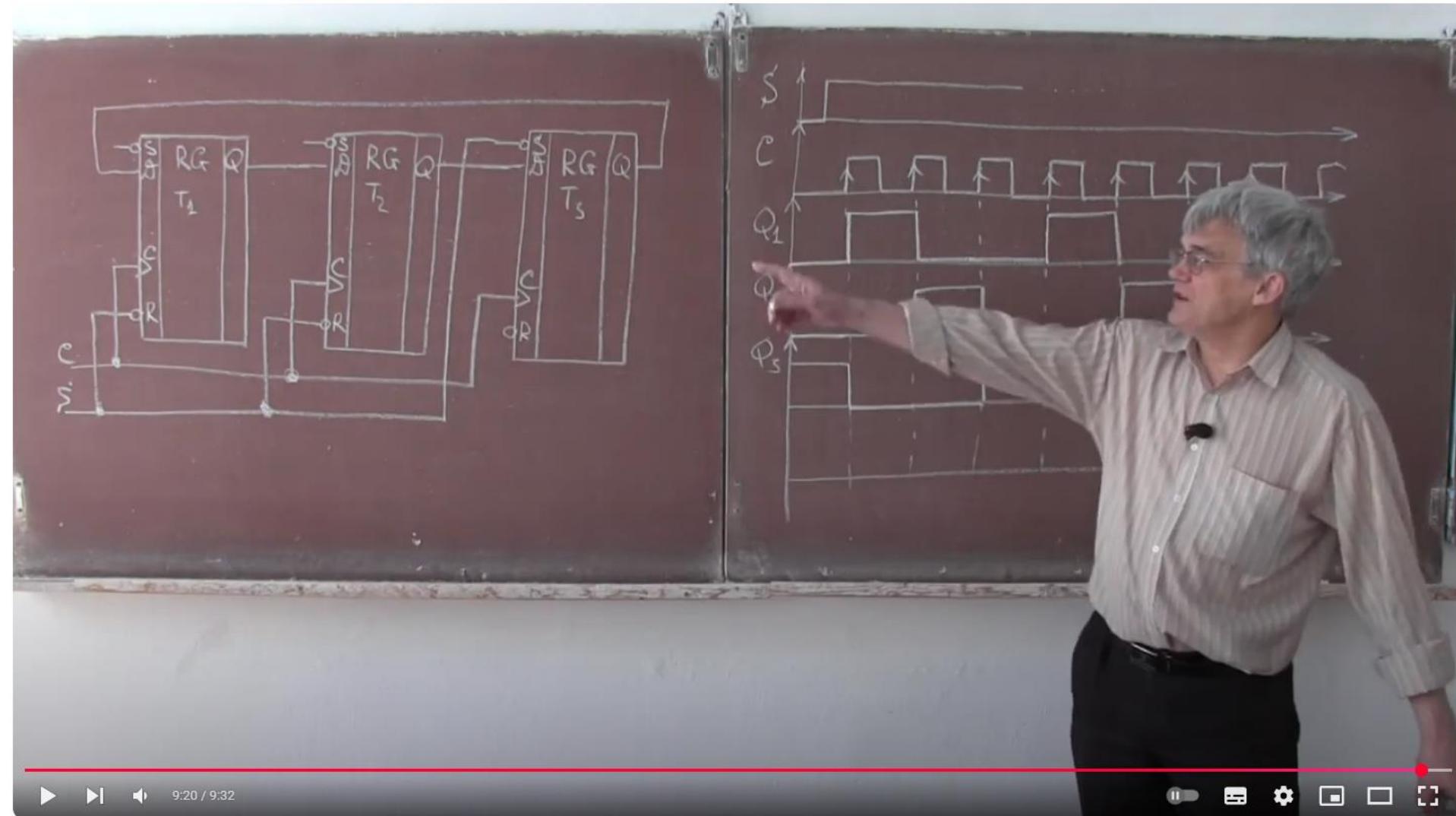
Основы цифровой схемотехники

14. Регистры

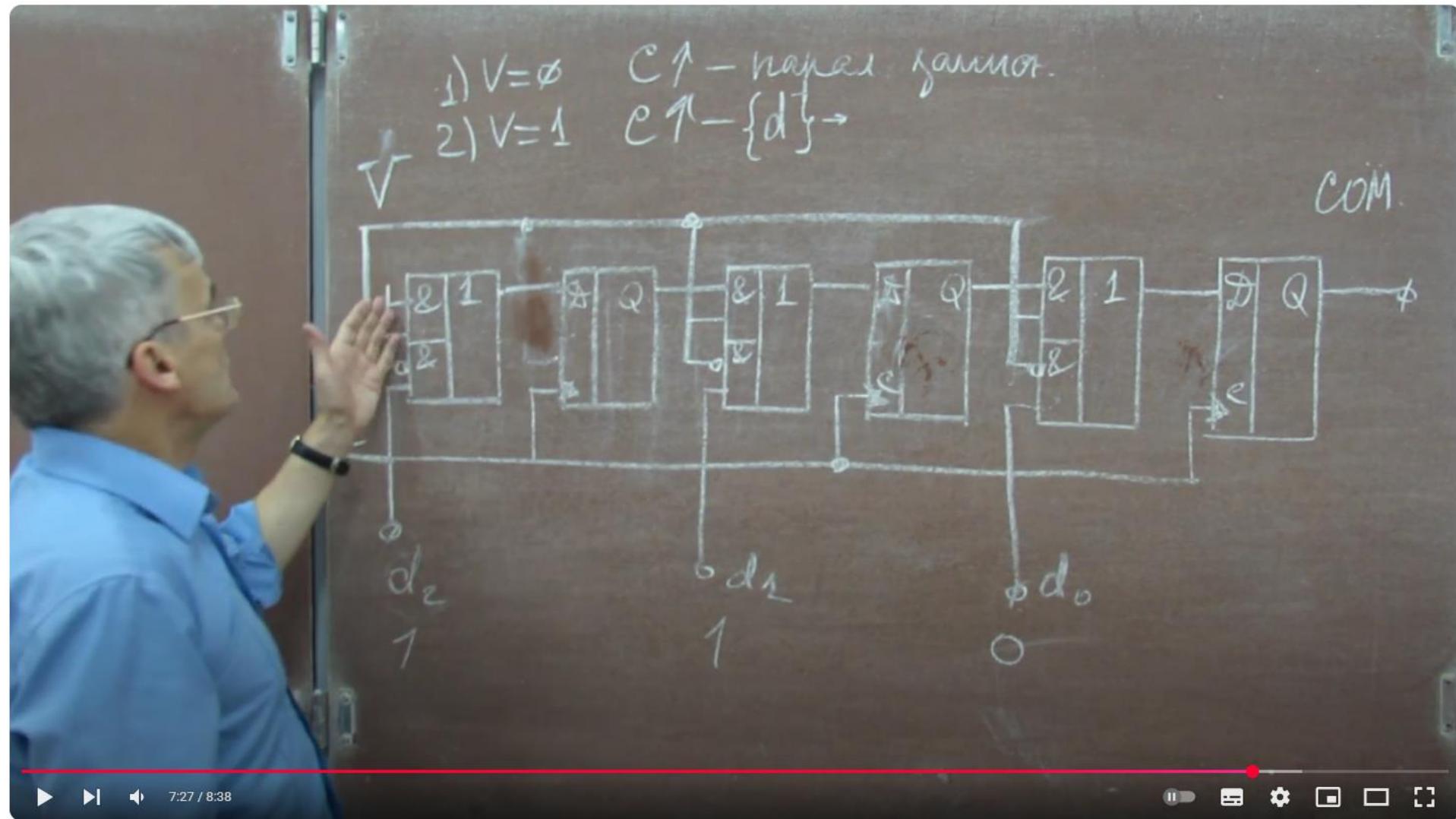


Марат Усс

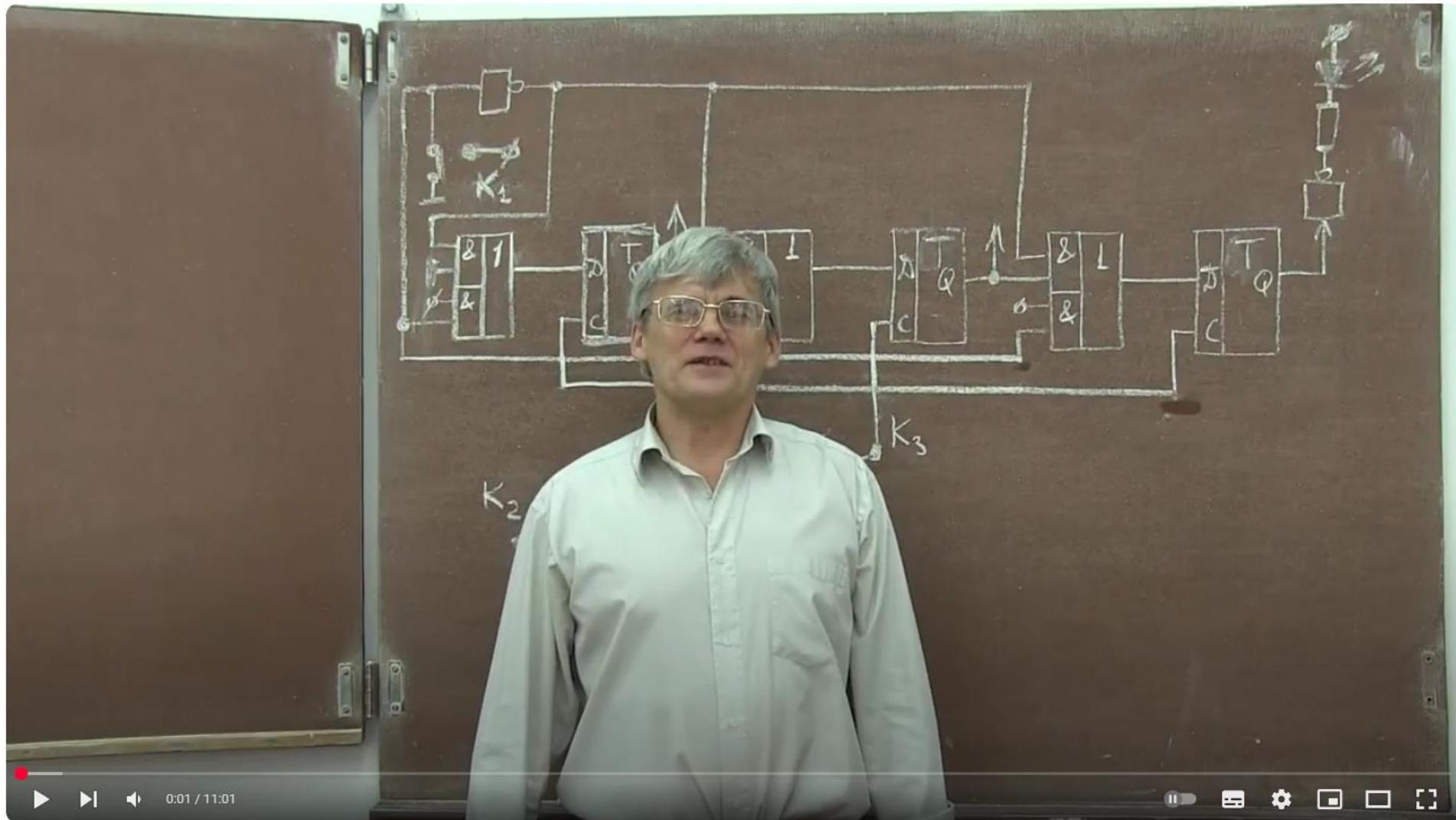
Основы цифровой схемотехники. 14. Регистры (2020)
<https://www.youtube.com/watch?v=Yxh7y6aHvXI>



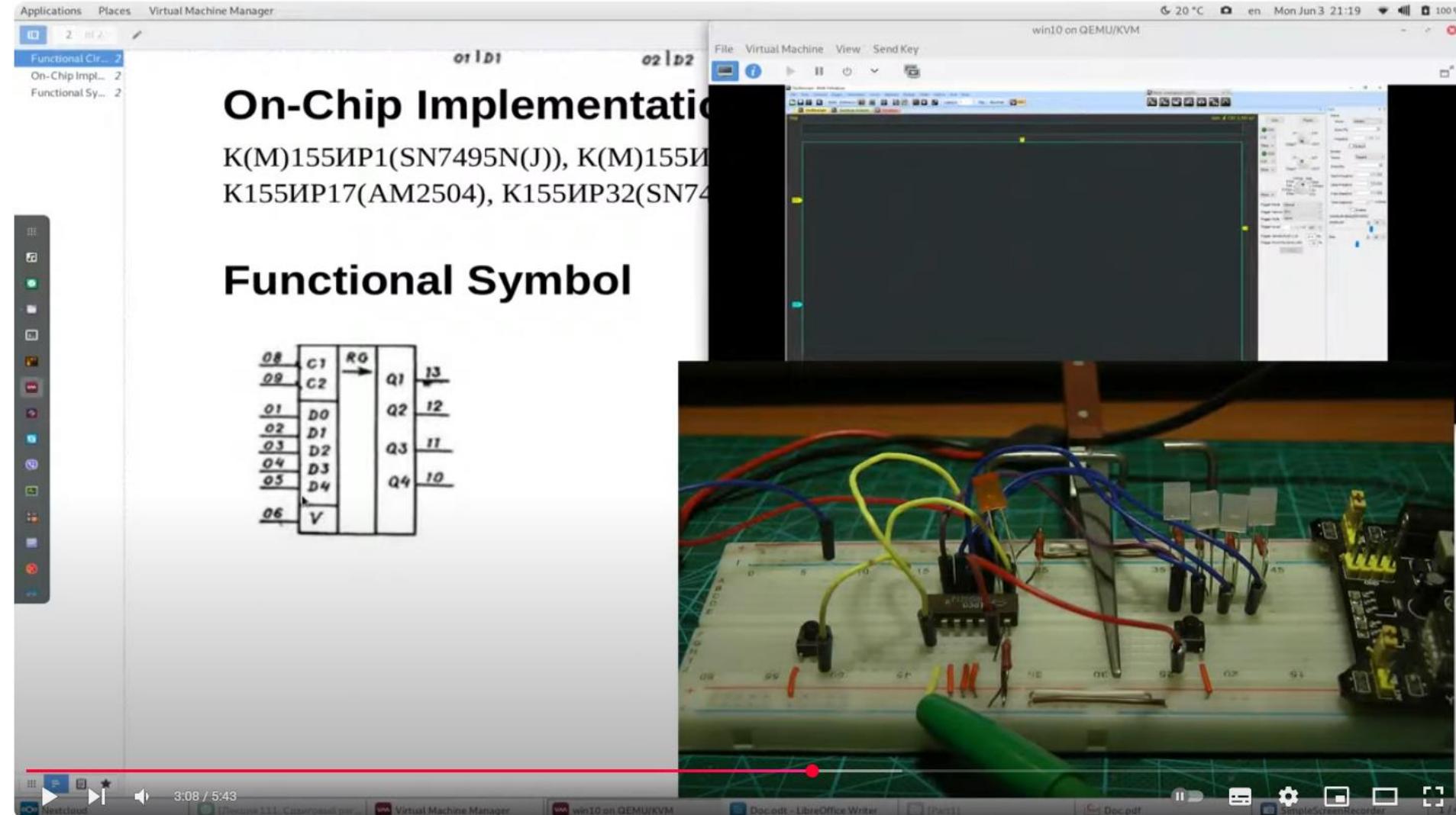
Лекция 296. Кольцевой регистр (2016)
<https://www.youtube.com/watch?v=YrQZHD2bE78>



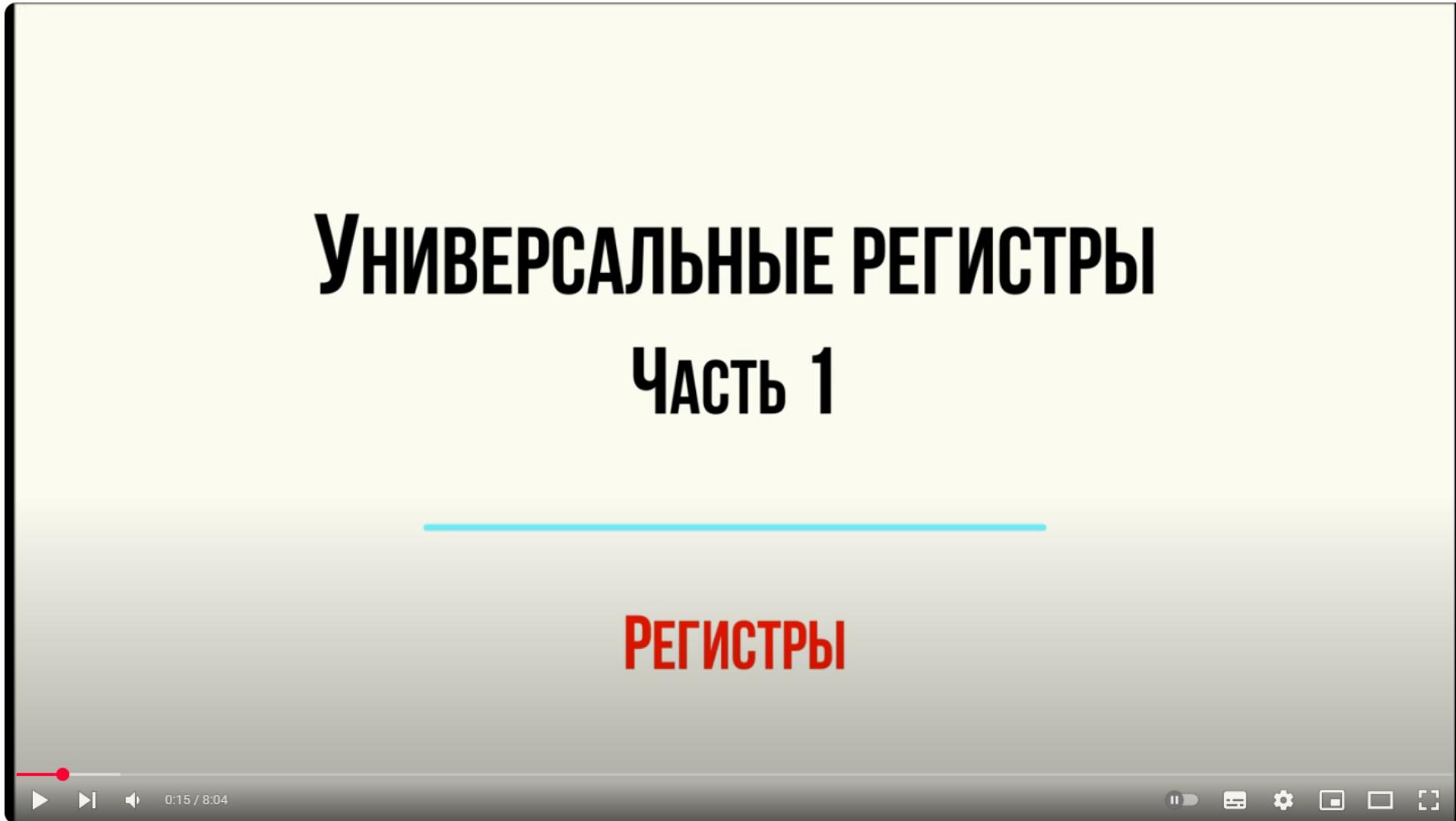
Лекция 111. Сдвиговый регистр (2013)
<https://www.youtube.com/watch?v=aq4JHS9eFO>



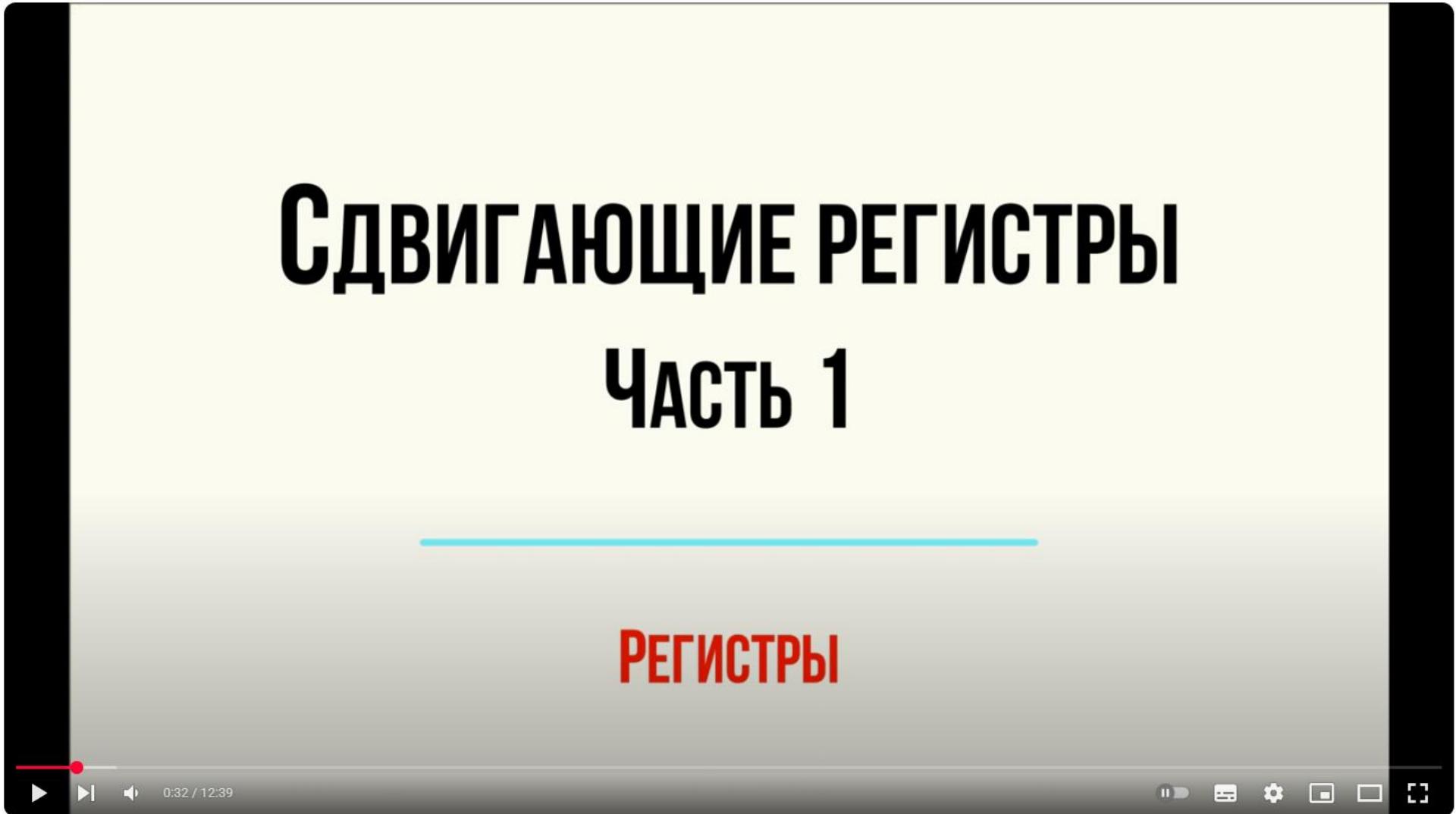
Лекция 148. Сдвиговый регистр на статических D триггерах (2014)
<https://www.youtube.com/watch?v=8ViAnNbM5m4>



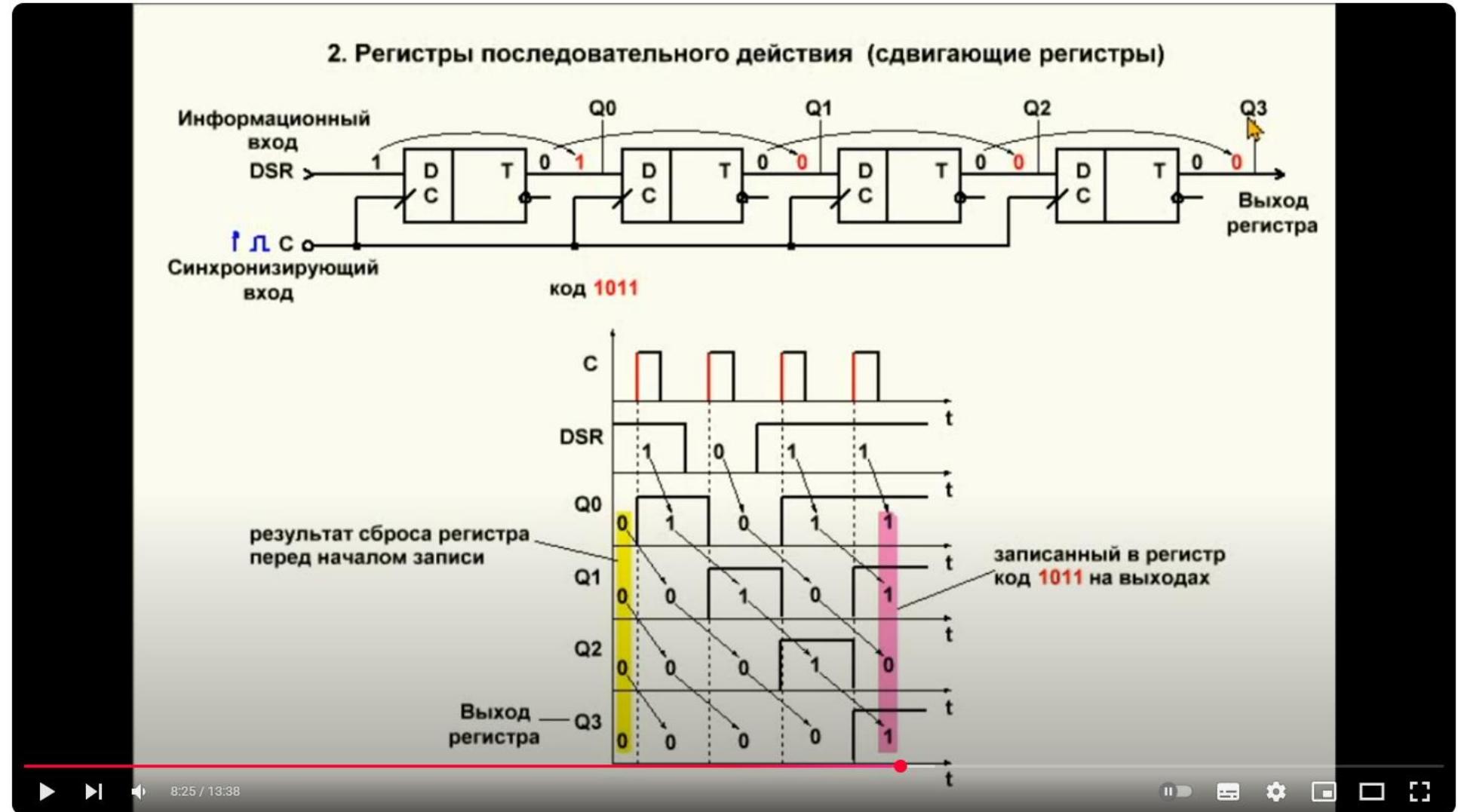
0111. Сдвиговый регистр (2019)
https://www.youtube.com/watch?v=vOFC7_r_JPE



Видеоурок 4.19. Универсальные регистры. Часть 1. (2024)
<https://www.youtube.com/watch?v=MapdzKWoBpg>



Видеоурок 4.17. Сдвигающие регистры. Часть 1. (2024)
<https://www.youtube.com/watch?v=rCpExW8D7gQ>



Видеоурок 4.18. Сдвигающие регистры. Часть 2. (2024)
<https://www.youtube.com/watch?v=x74L0RHhvM>

РЕГИСТРЫ ПАРАЛЛЕЛЬНОГО ДЕЙСТВИЯ (РЕГИСТРЫ D-ТИПА)

ЧАСТЬ 1

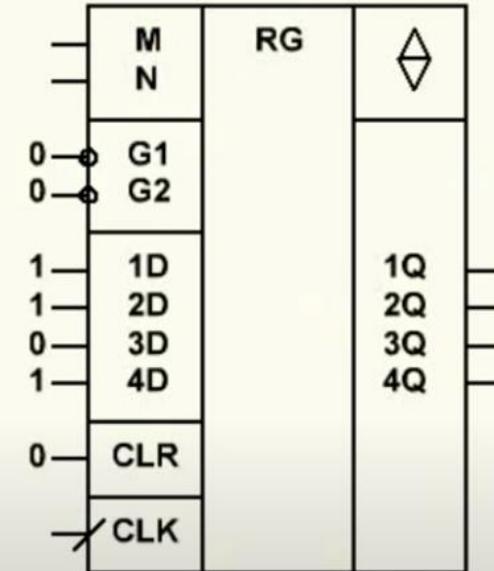
РЕГИСТРЫ



Видеоурок 4.14. Регистры D-типа. Часть 1. (2024)
<https://www.youtube.com/watch?v=ofZ11fa7AzU>

Регистры

KP1533ИР15 SN74LS173A



Режимы работы:

1. Запись
2. Хранение
3. Считывание

1D...4D - информационные входы

активные сигналы - 0 и 1

1Q...4Q - выходы

CLK - синхронизирующий вход

прямой динамический, активный сигнал - ↑

CLR - асинхронный вход сброса

прямой статический, активный сигнал - 1

G1,G2 - входы, разрешающие запись

инверсные статические, активные сигналы - 0

Z - состояние (запись, хранение)

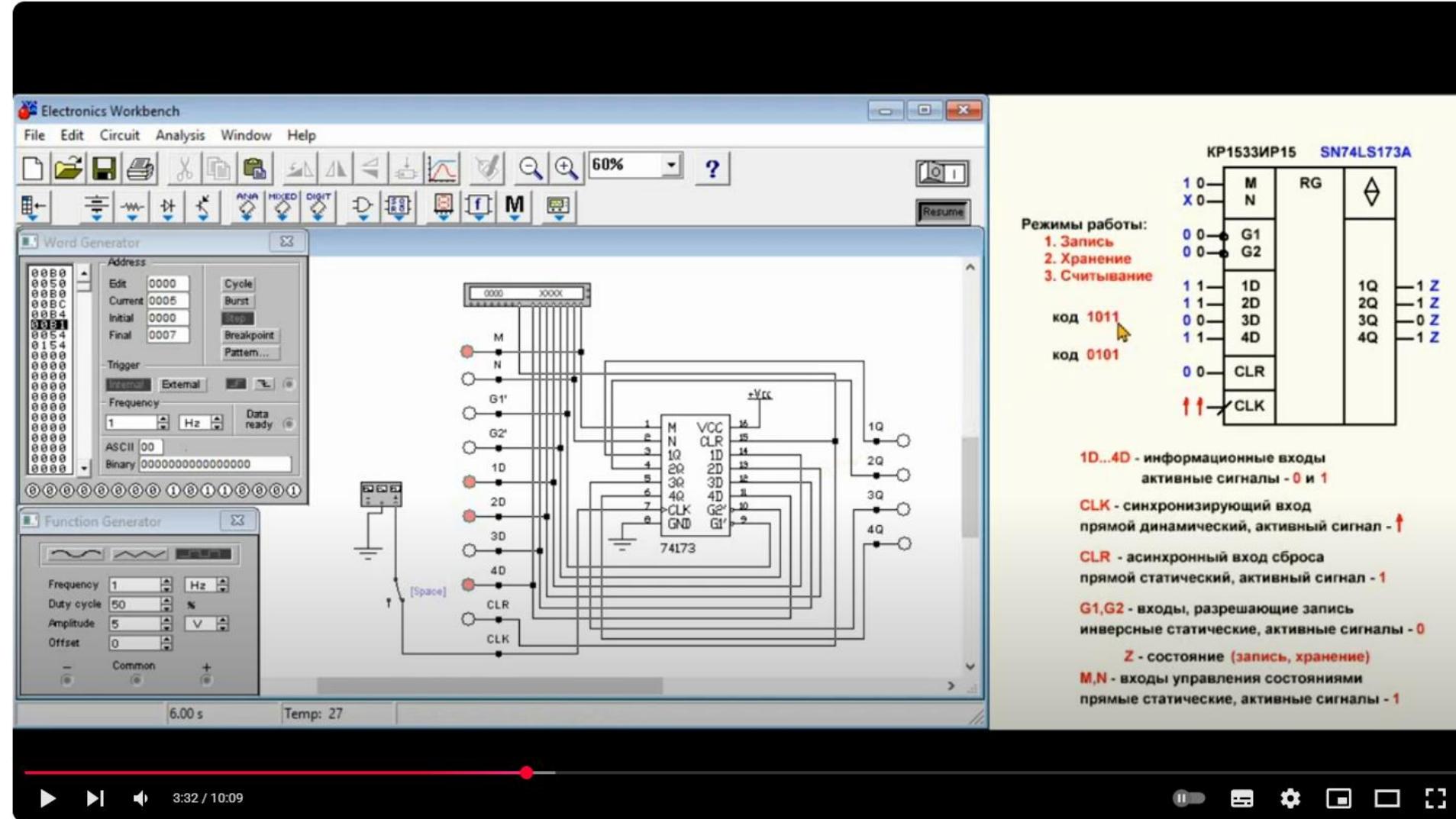
M,N - входы управления состояниями

прямые статические, активные сигналы - 1

код 1011



Видеоурок 4.15. Регистры D-типа. Часть 2. (2024)
<https://www.youtube.com/watch?v=UPOK9SRF9ns>



Видеоурок 4.16. Регистры D-типа. Часть 3. (2024)
<https://www.youtube.com/watch?v=XydeHPVr7EE>

Содержание

1. Назначение регистров.
2. Структура регистра.
3. Классификация регистров.
4. Параллельный регистр.
5. Сдвигающий (последовательный) регистр.
6. Моделирование регистров.
7. Контрольные вопросы.
8. Ссылки.



Регистры (2022)

<https://www.youtube.com/watch?v=1L00rrk-Cgc>

6.1.Регистры. Определение

Регистры – цифровые устройства последовательного типа, осуществляющие хранение и преобразование многоразрядных двоичных чисел.

Регистры реализуются на триггерах.

Регистры хранят информацию небольшой промежуток времени.

Регистры являются оперативными запоминающими устройствами (ОЗУ).

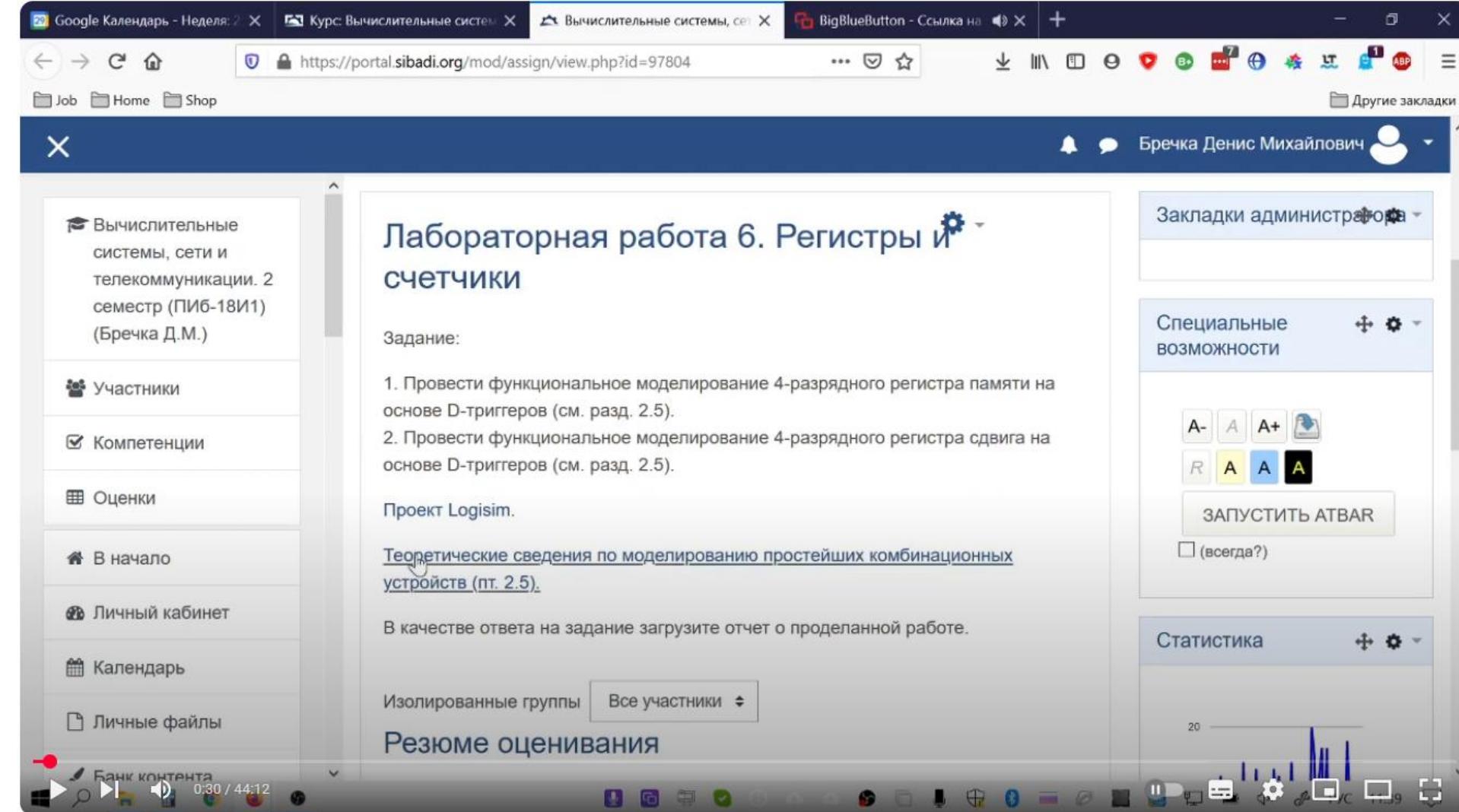
Регистры – важный узел всех цифровых радиотехнических систем.

Управляющие и запоминающие схемы, счётчики, генераторы кодов, кодопреобразователи, арифметические устройства выполняются на регистрах.



2

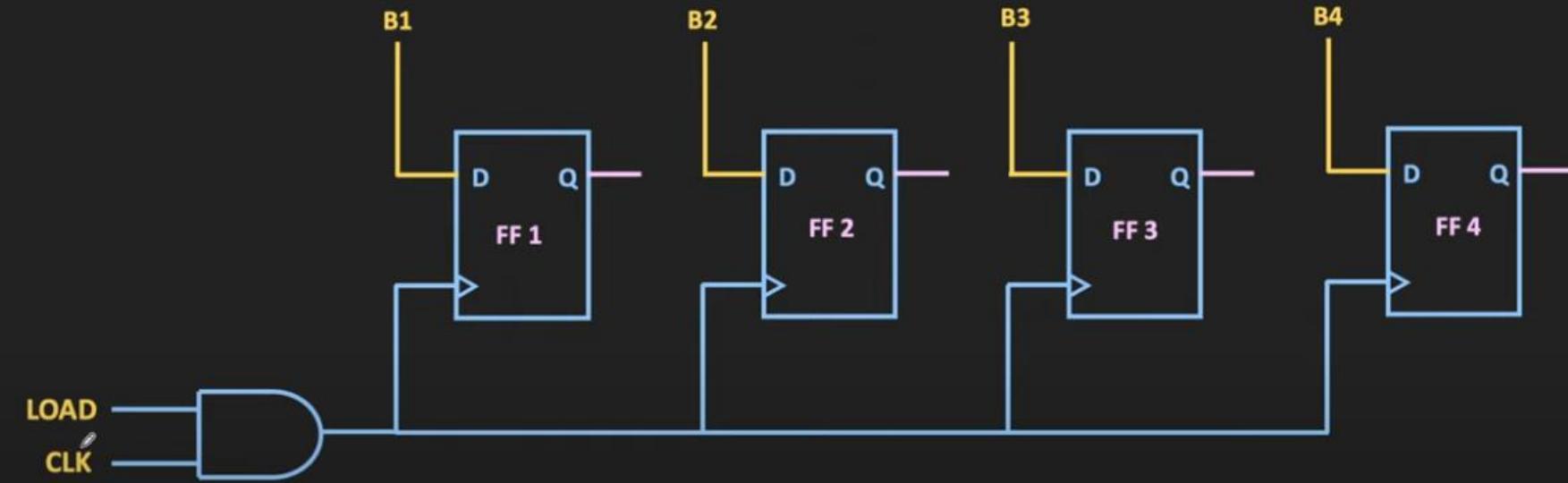
Цифровые устройства и программируемые логические интегральные схемы.
Лекция №11 (12.04.2022) (2022) <https://www.youtube.com/watch?v=w0BeTnjnEEw>



Лабораторная работа № 6. Регистры и счетчики (2021)

<https://www.youtube.com/watch?v=13YKegt4L4Q>

4-bit Register



▶ ▶ 🔍 4:14 / 10:53 • Basic Design of 4 bit Register (with LOAD and CLEAR inputs) >

ALL ABOUT ELECTRONICS

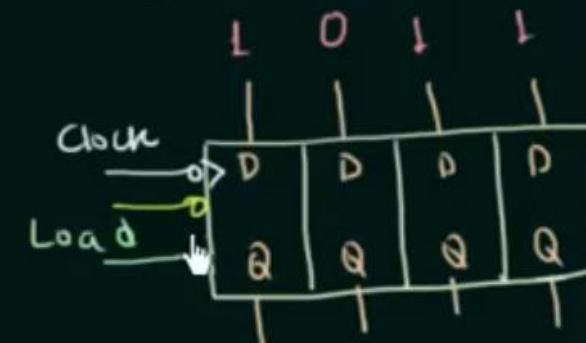
Introduction to Registers | What is Shift Register? Types of Shift Registers (2022)
Введение в регистры | Что такое сдвиговый регистр? Типы сдвиговых регистров (2022)
<https://www.youtube.com/watch?v=bAQfPQqKCHs>

i

>> Flip Flop is 1-bit memory cell

1015

- » To increase the storage capacity, we have to use group of flip-flop. This group of ff is known as REGISTER.
- » The n-bit register consist of "n" number of flip-flops and is capable of storing "n-bit" word.



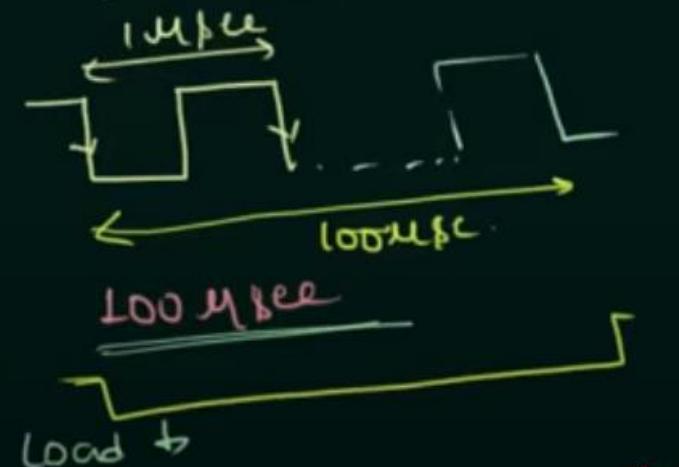
Synch :- Clock↑ and Load'

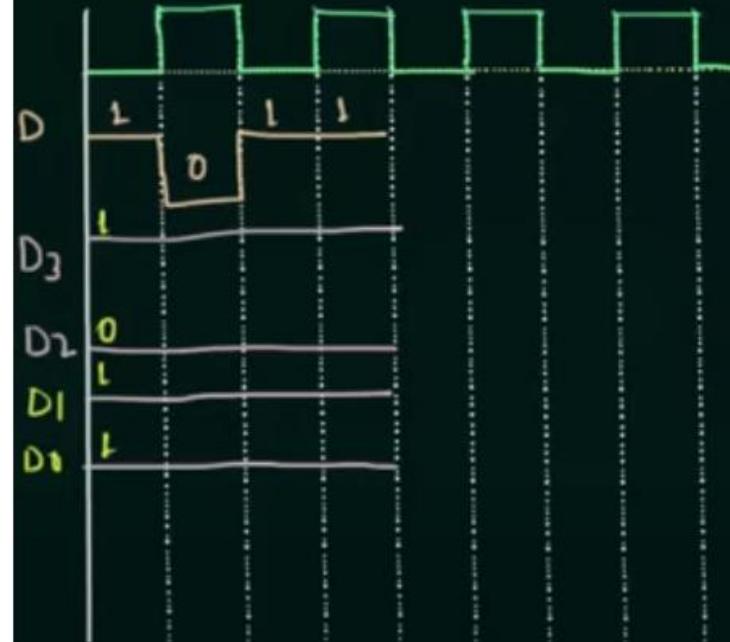
A synch :- only load

We are bound to follow the clock.

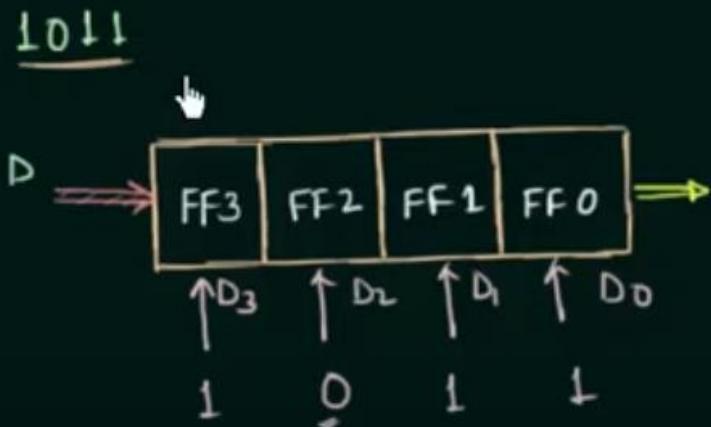
$$f = 1 \text{ MHz}$$

$$T = 144 \text{ sec.}$$



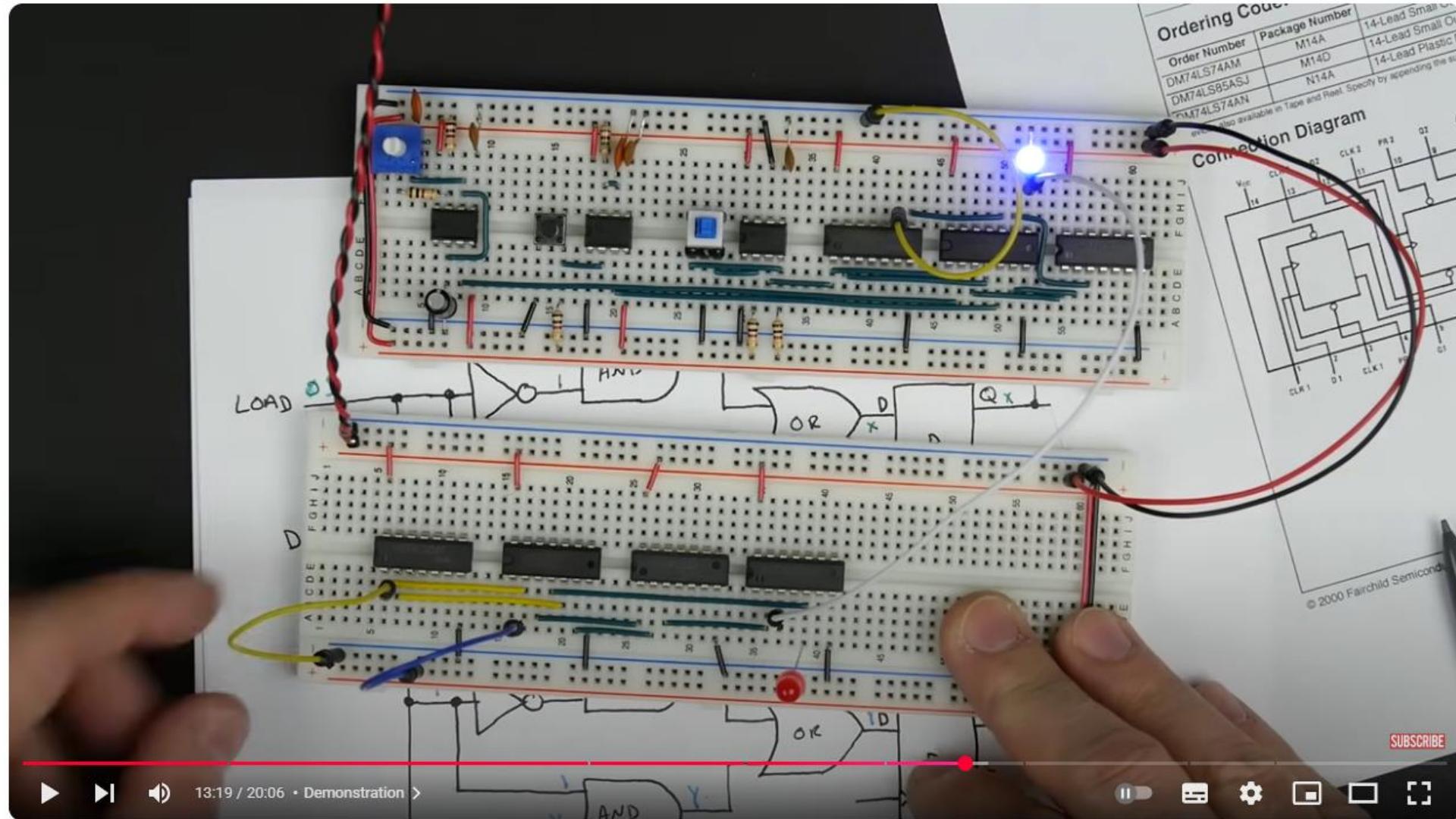
Data Formats & Classification of Registers

» Data can be entered in serial or in parallel form.

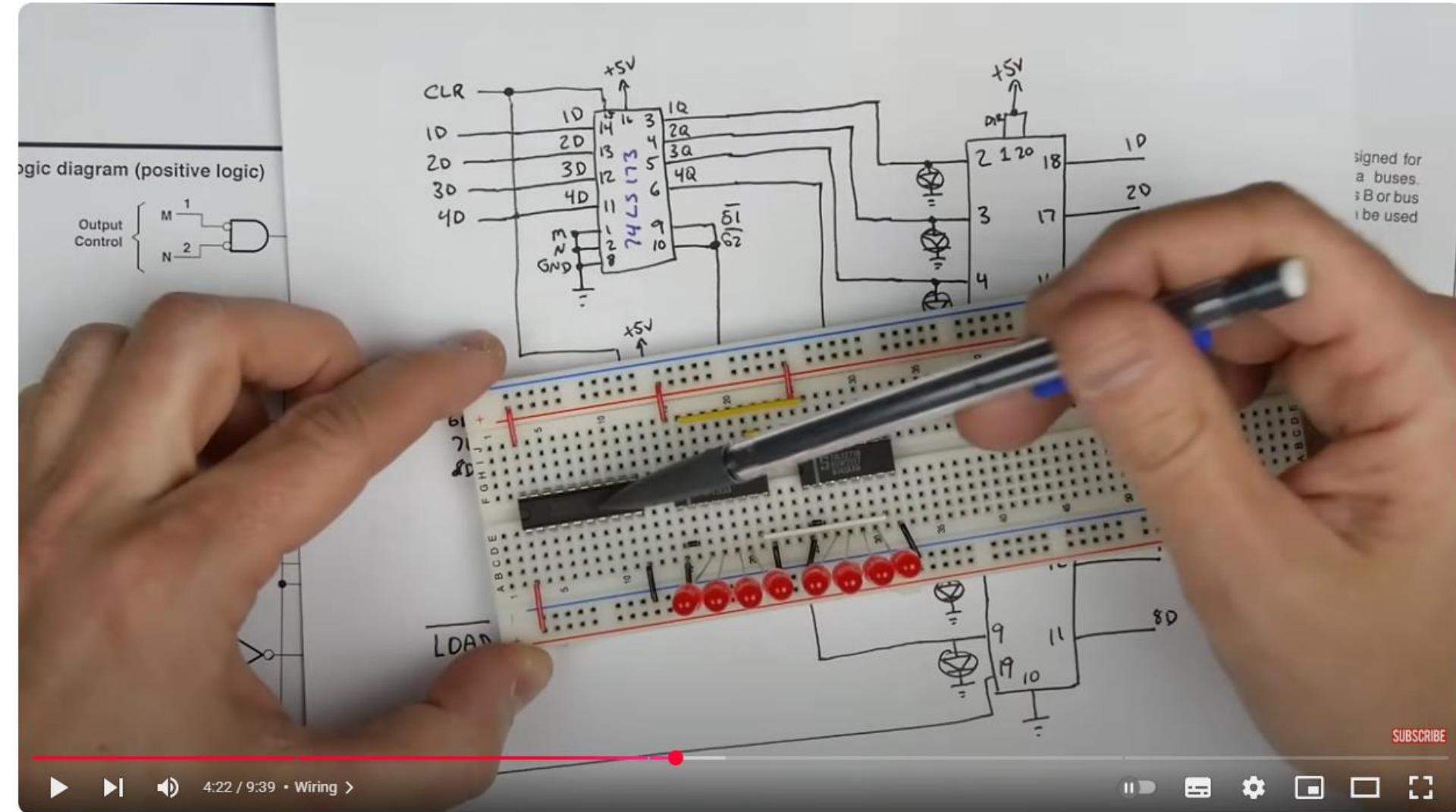


one bit at a time
all bits at a time

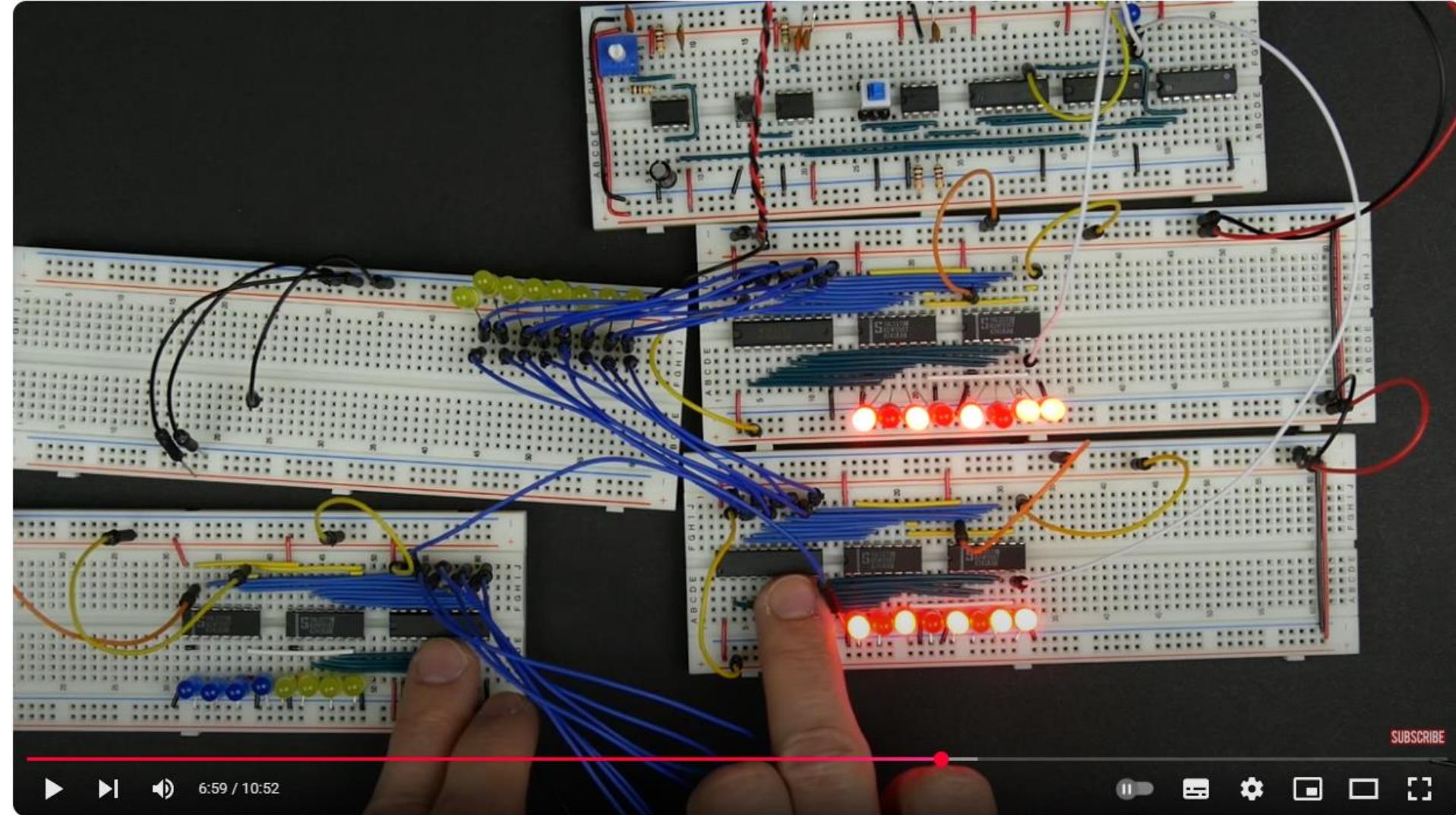




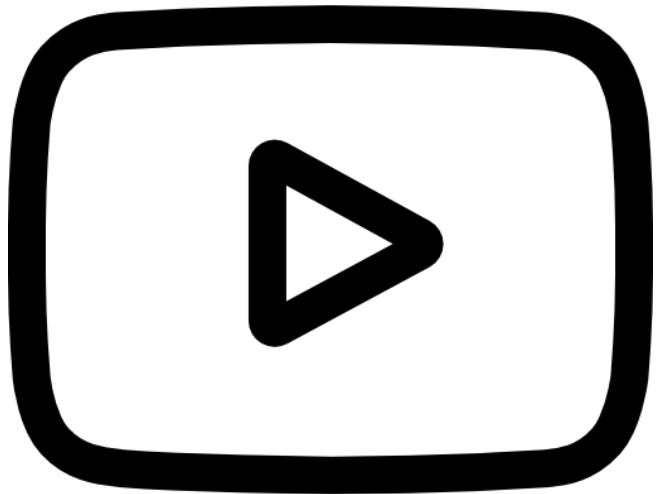
Designing and building a 1-bit register - 8 bit register - Part 3 (2016)
Проектирование и создание 1-разрядного регистра - 8-разрядный регистр - Часть 3 (2016)
https://www.youtube.com/watch?v=-arYx_oVlj8



Building an 8-bit register - 8-bit register - Part 4 (2016)
 Создание 8-разрядного регистра - 8-разрядный регистр - Часть 4 (2016)
https://www.youtube.com/watch?v=CiMaWbz_6E8

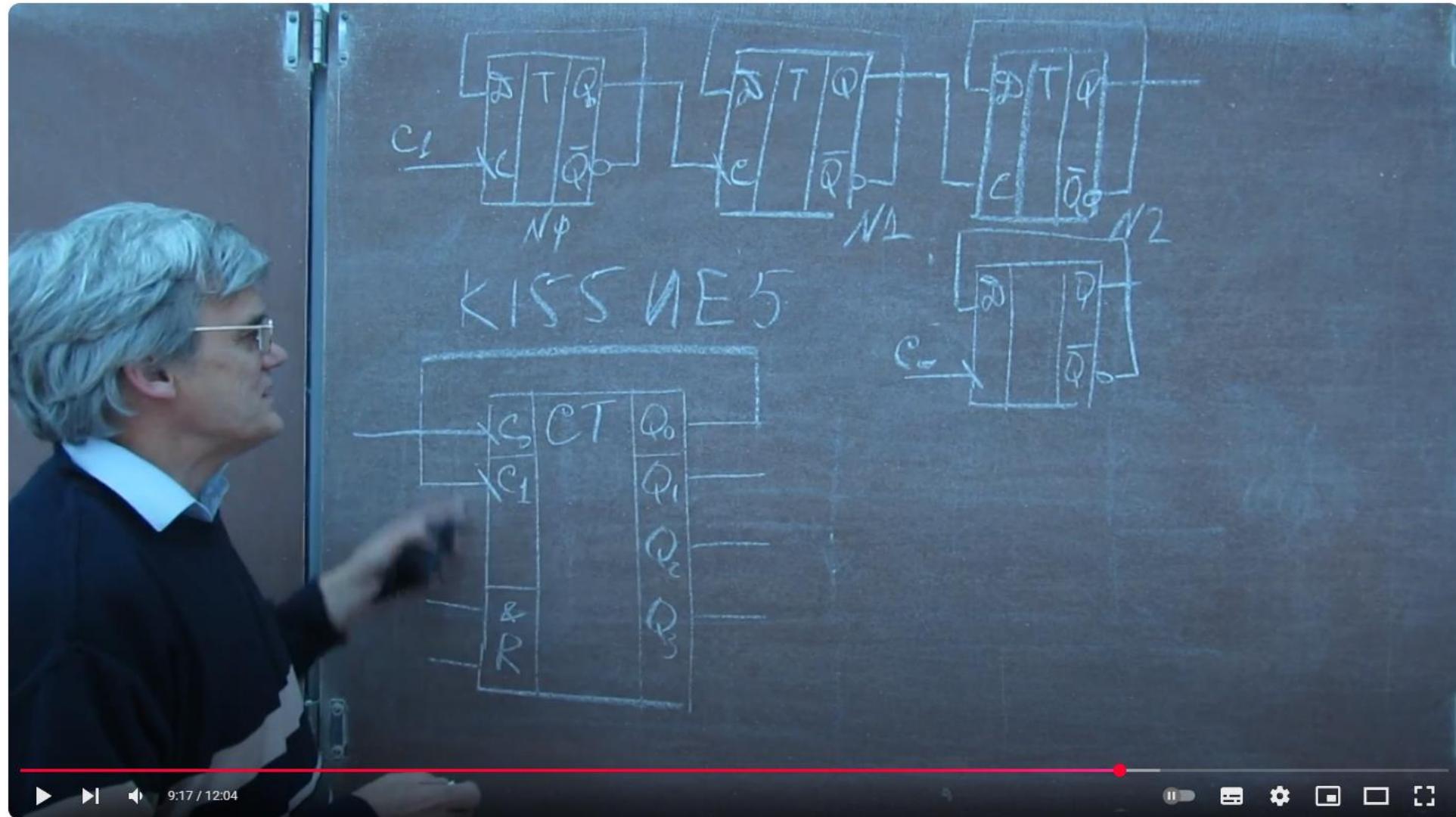


Testing our computer's registers - 8-bit register - Part 5 (2016)
Тестирование регистров нашего компьютера - 8-разрядный регистр - Часть 5 (2016)
<https://www.youtube.com/watch?v=9WE3Obdjtv0>



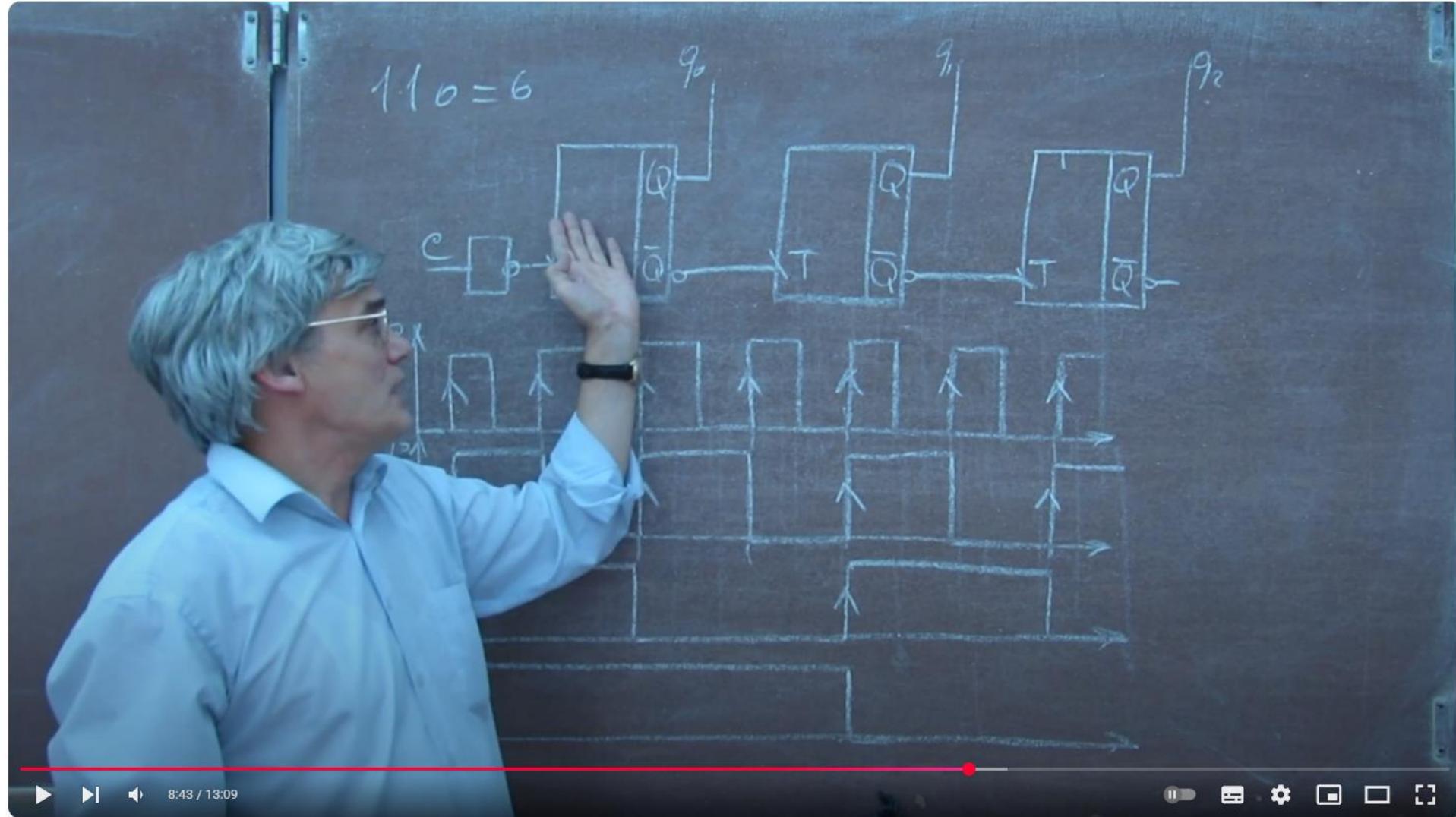
Счетчики



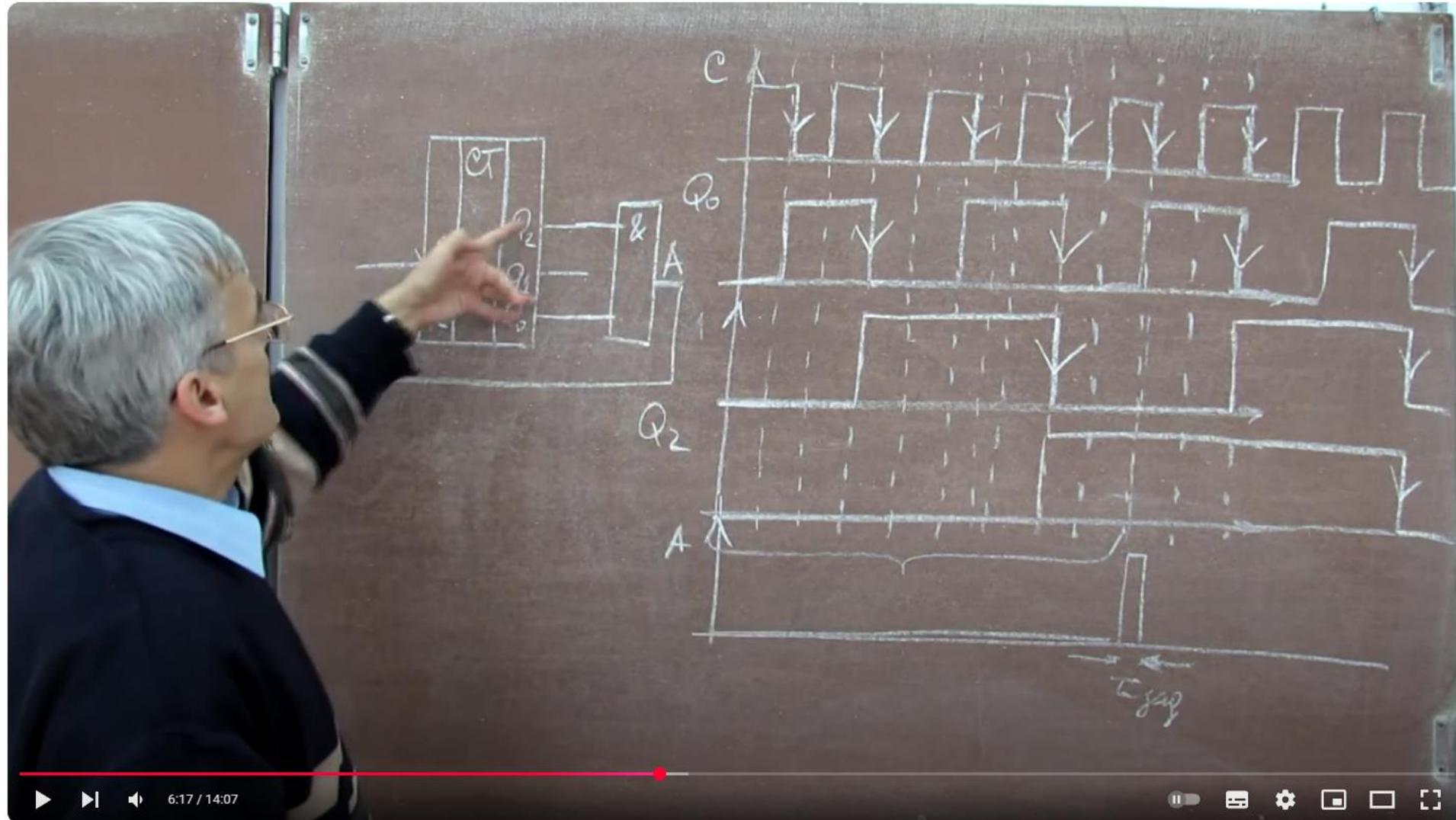


Лекция 106. Двоичные счетчики (2013)

<https://www.youtube.com/watch?v=xNO3UPW3QB4>



Лекция 107. Реверсивный счетчик (2013)
<https://www.youtube.com/watch?v=n0XvpFcRvHc>



Лекция 114. Счетчики с произвольным коэффициентом пересчета (2013)

https://www.youtube.com/watch?v=kl94_00tSJc



Как работает двоичный счётчик? Объясняем на пальцах! (2022)
https://www.youtube.com/watch?v=v_shMnmFnns

Счетчики

Счетчик – функциональный узел ЭВМ, выполняющий функцию хранения информации и микрооперацию счета.

Классификация счетчиков

Счетчики подразделяются на:

1. По назначению:

- суммирующие $CЧ := CЧ + 1$;
- вычитающие $CЧ := CЧ - 1$
- реверсивные (выполняющие суммирование и вычитание).

2. По системе счисления:

- двоичные, десятичные, другие модули счета.

3. По организации цепей переноса:

- с последовательным,
- сквозным,
- параллельным,
- групповым переносом.

4. По порядку изменения состояния:

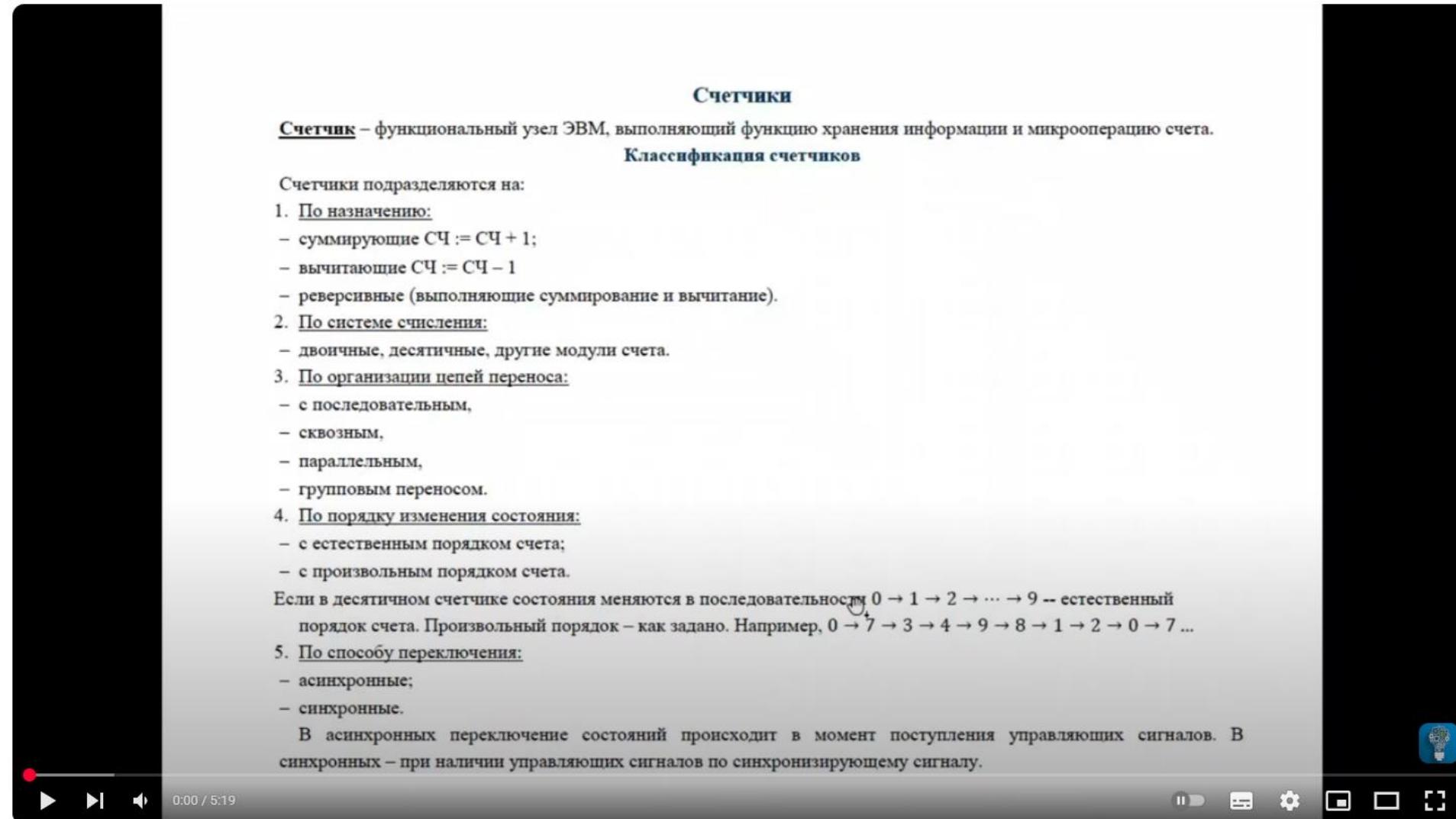
- с естественным порядком счета;
- с произвольным порядком счета.

Если в десятичном счетчике состояния меняются в последовательности $0 \rightarrow 1 \rightarrow 2 \rightarrow \dots \rightarrow 9$ – естественный порядок счета. Произвольный порядок – как задано. Например, $0 \rightarrow 7 \rightarrow 3 \rightarrow 4 \rightarrow 9 \rightarrow 8 \rightarrow 1 \rightarrow 2 \rightarrow 0 \rightarrow 7 \dots$

5. По способу переключения:

- асинхронные;
- синхронные.

В асинхронных переключение состояний происходит в момент поступления управляющих сигналов. В синхронных – при наличии управляющих сигналов по синхронизирующему сигналу.



РК6. Схемотехника. Счётчики: введение (2020)
<https://www.youtube.com/watch?v=lnvpaVga87A>

Для счетчиков с последовательным переносом

$$t_{\text{уст}} = n \cdot t_{\text{здр.тг}},$$

где n – число разрядов счетчика,

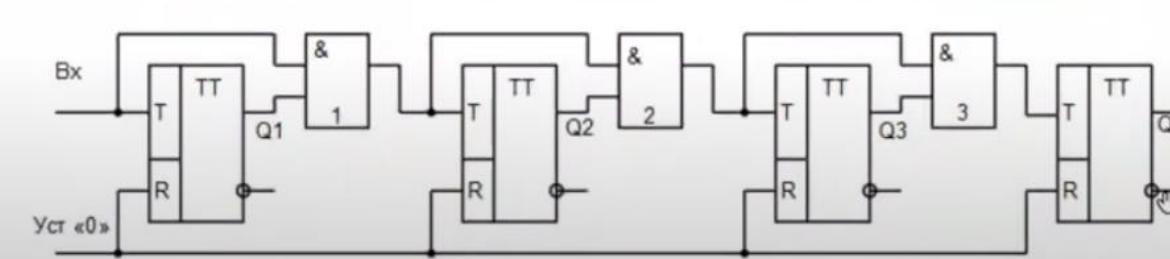
$t_{\text{здр.тг}}$ -- время задержки распространения сигнала в триггере.

По временным диаграммам можно определить, что частота сигнала на выходе Q_1 в два раза меньше, чем частота входных сигналов. Частота сигнала на выходе Q_2 в четыре раза меньше, чем частота входных сигналов, а на выходе Q_3 – в восемь раз меньше. Т.е. счетчик может выполнять функцию делителя частоты.

Счетчик со сквозным переносом

В счетчиках с последовательным переносом сигнал переноса проходит от входа до старшего n -ого разряда через цепочку из $n-1$ триггеров. Быстродействие таких счетчиков низкое.

В счетчиках со сквозным переносом сигнал переноса проходит через цепочку логических элементов (ЛЭ), что ускоряет работу счетчика.



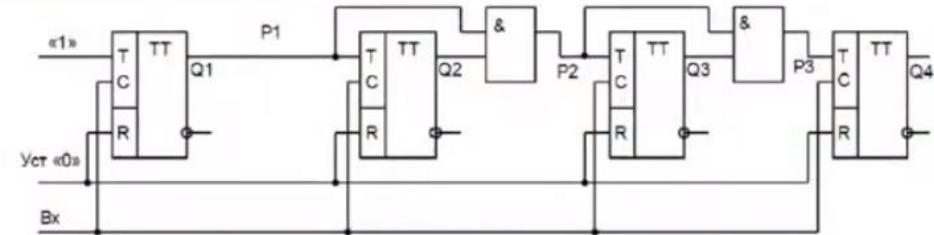
Временные диаграммы иллюстрируют принцип работы счетчика.



РК6. Схемотехника. Счётчики: асинхронные с ускоренным переносом (2020)
<https://www.youtube.com/watch?v=7LSXNoMp99U>

Синхронные счетчики с ускоренным переносом

1. Синхронный счетчик со сквозным переносом



Счетный сигнал поступает одновременно на синхровходы всех триггеров. Переносы проходят по цепи сквозного переноса, составленной из элементов И. Перенос из i -го разряда вырабатывается в соответствии с выражением $P_i = P_{i-1} \cdot Q_i$, при этом принимается $P_0 = 1$.

Итак, $P_0 = 1$; $P_1 = P_0 \cdot Q_1 = Q_1$; $P_2 = P_1 \cdot Q_2$; $P_3 = P_2 \cdot Q_3$.

После окончания переходных процессов в цепи переносов на синхровходы триггеров подается счетный сигнал. Если $P_{i-1} = 1$, то i -й триггер переключается в противоположное состояние, если $P_{i-1} = 0$ -- сохраняет прежнее состояние. Т.о., переключение счетчика происходит одновременно по всем разрядам в зависимости от сигналов на управляющих входах. После перехода счетчика в новое состояние в цепи переносов вырабатываются новые управляющие сигналы. Задержка распространения переносов отсчитывается от момента переключения счетчика и равна суммарной задержке переключения элементов И. Время установки кода в счетчике, измеряемое с момента окончания счетного импульса, равно времени переключения второй ступени триггеров

$$t_{уст} = t_{здр.тт}$$

Период работы счетчика

$$T \geq t_u + t_{здр.тт} + (n - 2) \cdot t_{здр.лэ},$$

где t_u -- минимальная длительность входного импульса, равная времени переключения первой ступени триггера,

$t_{здр.тт}$ -- задержки распространения сигнала в триггере (в его второй ступени),

$t_{здр.лэ}$ -- время задержки распространения сигнала на одном ЛЭ в цепи сквозного переноса,

n -- число разрядов счетчика.



РК6. Схемотехника. Счётчики: синхронные с ускоренным переносом (2020)
https://www.youtube.com/watch?v=pLbeK_KBdc8

Двоично-кодированные счетчики с произвольным модулем счета

При построении цифровых устройств часто требуются счетчики с модулем счета, не выражаемым целой степенью двойки, т.е. произвольным. Например, широкое применение находят двоично-десятичные счетчики с модулем счета $M=10$.

Синтез синхронных счетчиков

Рассмотрим алгоритм на примере синтеза суммирующего счетчика на JK-триггерах с $M=10$.

1) В синхронных счетчиках входной сигнал подается на входы С всех триггеров.

2) Определение числа разрядов счетчика

$$n = \lceil \log_2 M \rceil,$$

где $\lceil \rceil$ – округление до ближайшего справа целого числа,

M – коэффициент пересчета.

При $M=10$.

$$n = \lceil \log_2 10 \rceil = \lceil 3,32 \dots \rceil = 4$$

3) Выбирается тип триггеров.

Мы выбираем JK-триггеры.

4) Составляется характеристическая таблица возможных переходов триггера из текущего состояния Q^t в последующее Q^{t+1} .

Для JK-триггера

$Q^t \rightarrow Q^{t+1}$		JK-триггер	
Q^t	Q^{t+1}	J	K
0 → 0	0	α	
0 → 1	1	α	
1 → 0	α	1	
1 → 1	α	0	

Для RS-, D-, T- триггеров

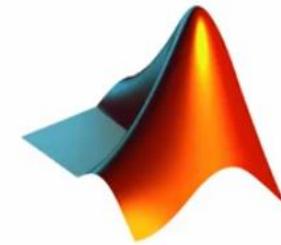
$Q^t \rightarrow Q^{t+1}$		RS-триггер	D-триггер	T-триггер	
Q^t	Q^{t+1}	S	R	D	T
0 → 0	0	α		0	0
0 → 1	1	0		1	1
1 → 0	0	1		0	1
1 → 1	α	0		1	0



РК6. Схемотехника. Счётчики: синтез синхронных двоичных счётчиков с заданным модулем счёта (2020) <https://www.youtube.com/watch?v=PMRUCmxs2B4>

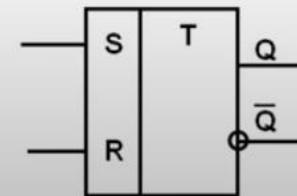
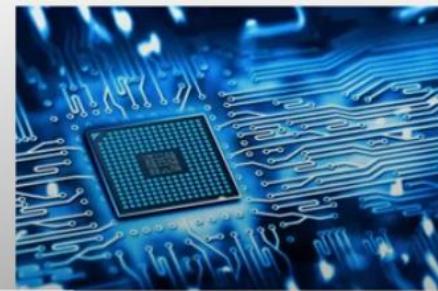


экспонента
ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ



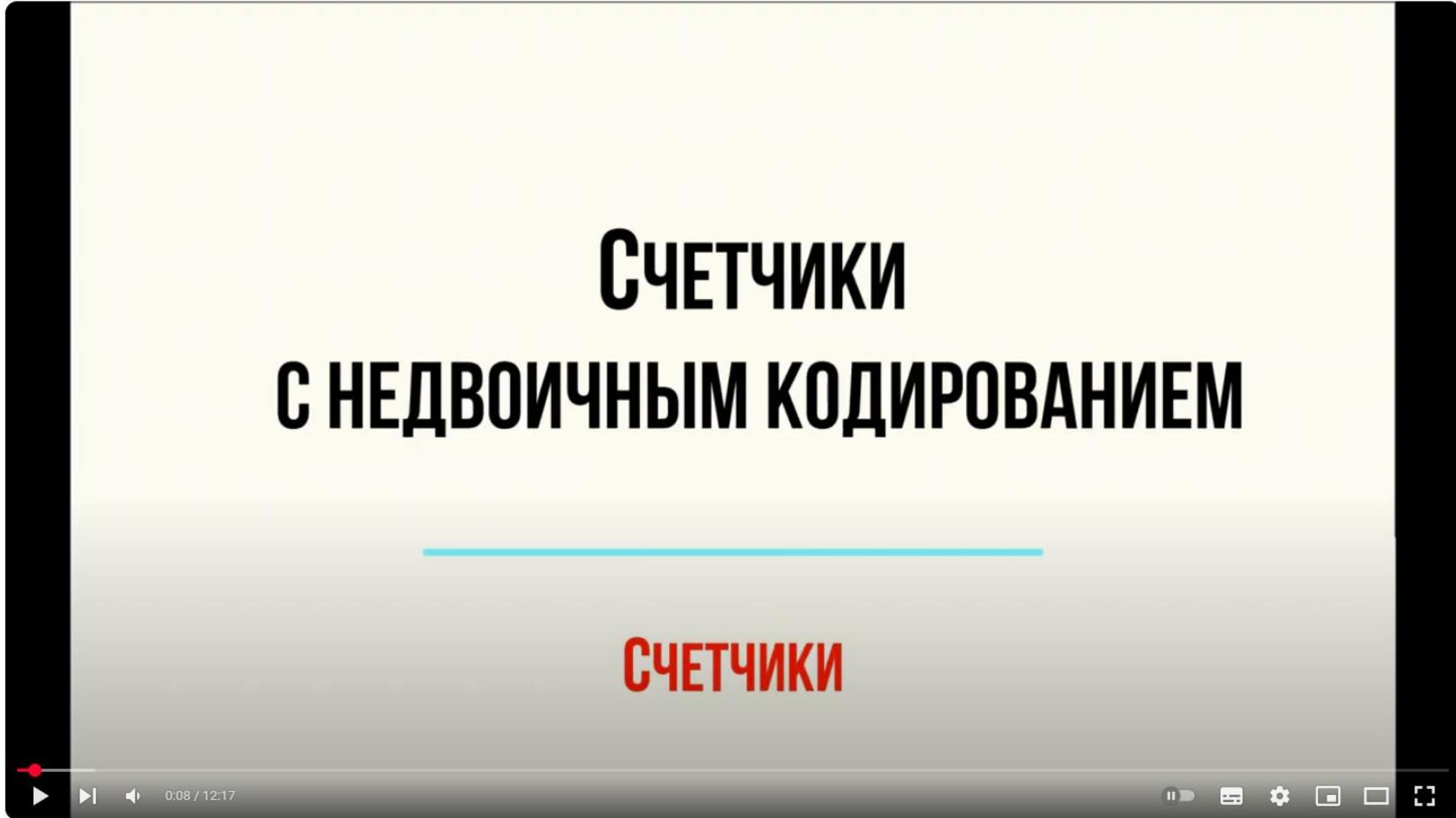
Основы цифровой схемотехники

15. Счётчики



Основы цифровой схемотехники. 15. Счётчики (2020)
<https://www.youtube.com/watch?v=2LVVOpRNAT4>

0106. Двоичные счётчики (2019)



Видеоурок 4.39. Счетчики с недвоичным кодированием (2025)
<https://www.youtube.com/watch?v=AjGMztivnjY>

**Счетчики
с переменным управляемым коэффициентом деления**

S1	СТ	
S2		
S3		
S4		
S5		
S6		
S7		
S8		
S9		
S10		
S11		
S12		
S13		
S14		
S15		
S16		
K1		
K2		
K3		
K4		
CLK		

K564ИЕ15
CD4059

$N = 21327 \quad N = M \cdot (1000 \cdot P1 + 100 \cdot P2 + 10 \cdot P3 + P4) + P5$

M - модуль K4...K1

P4 - множитель единиц S8...S5

P3 - множитель десятков S12...S9

P2 - множитель сотен S16...S13

P1 - множитель тысяч } S4...S1

P5 - остаток

Таблица 1

Модуль	Входы установки P5	Входы установки P1
2	S1	S4...S2
4	S2,S1	S4,S3
5	S3...S1	S4
8	S3...S1	S4
10	S4...S1	—

Таблица 2

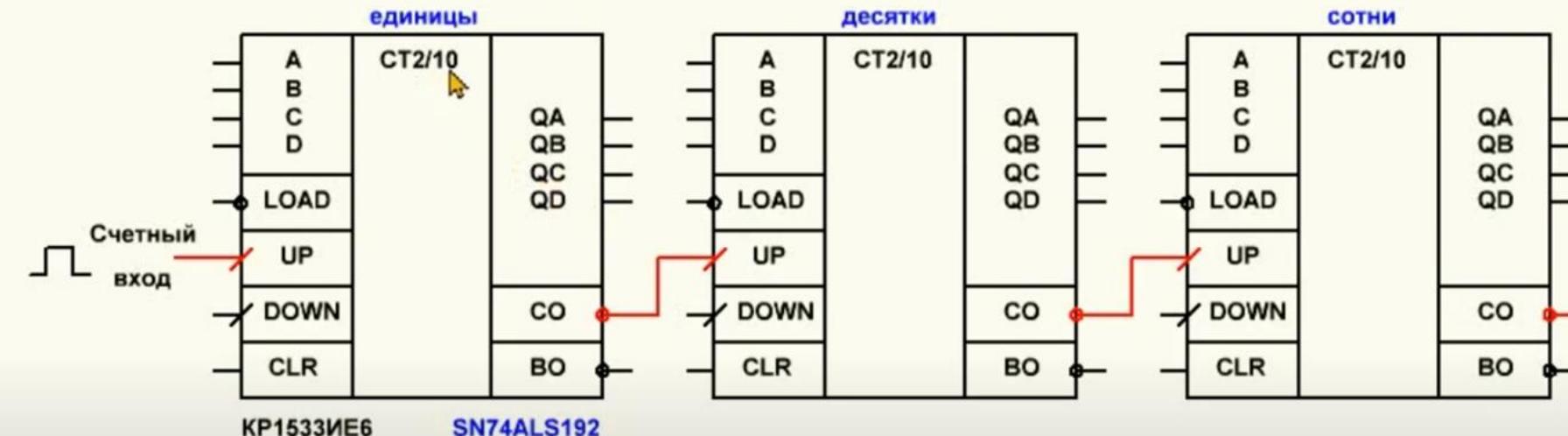
K4	K3	K2	K1	Модуль	Режим
0	0	0	0	—	D
0	0	0	1	—	D
0	0	1	0	—	A
0	0	1	1	10	B
0	1	0	0	8	B
0	1	0	1	5	B
0	1	1	0	4	B
0	1	1	1	2	B
1	0	0	0	—	D
1	0	0	1	—	D
1	0	1	0	10	B
1	0	1	1	10	C
1	1	0	0	8	C
1	1	0	1	5	C
1	1	1	0	4	C
1	1	1	1	2	C

1. Режим A. $N = 10$
2. Режим B. Программируемый N
3. Режим C. Сигнализация окончания счета
4. Режим D. Настроочный

Видеоурок 4.38. Счетчики с переменным управляемым коэффициентом счета (2025)
<https://www.youtube.com/watch?v=51kTdYk4Lx4>

Каскадное соединение счетчиков

1. С последовательным переносом



UP - счетный вход в режиме суммирования

DOWN - счетный вход в режиме вычитания

CLR - асинхронный вход сброса

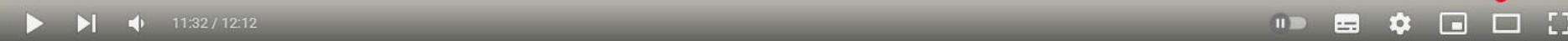
QA, QB, QC, QD - выходы результатов счета

ABCD - информационные входы
для предварительной загрузки

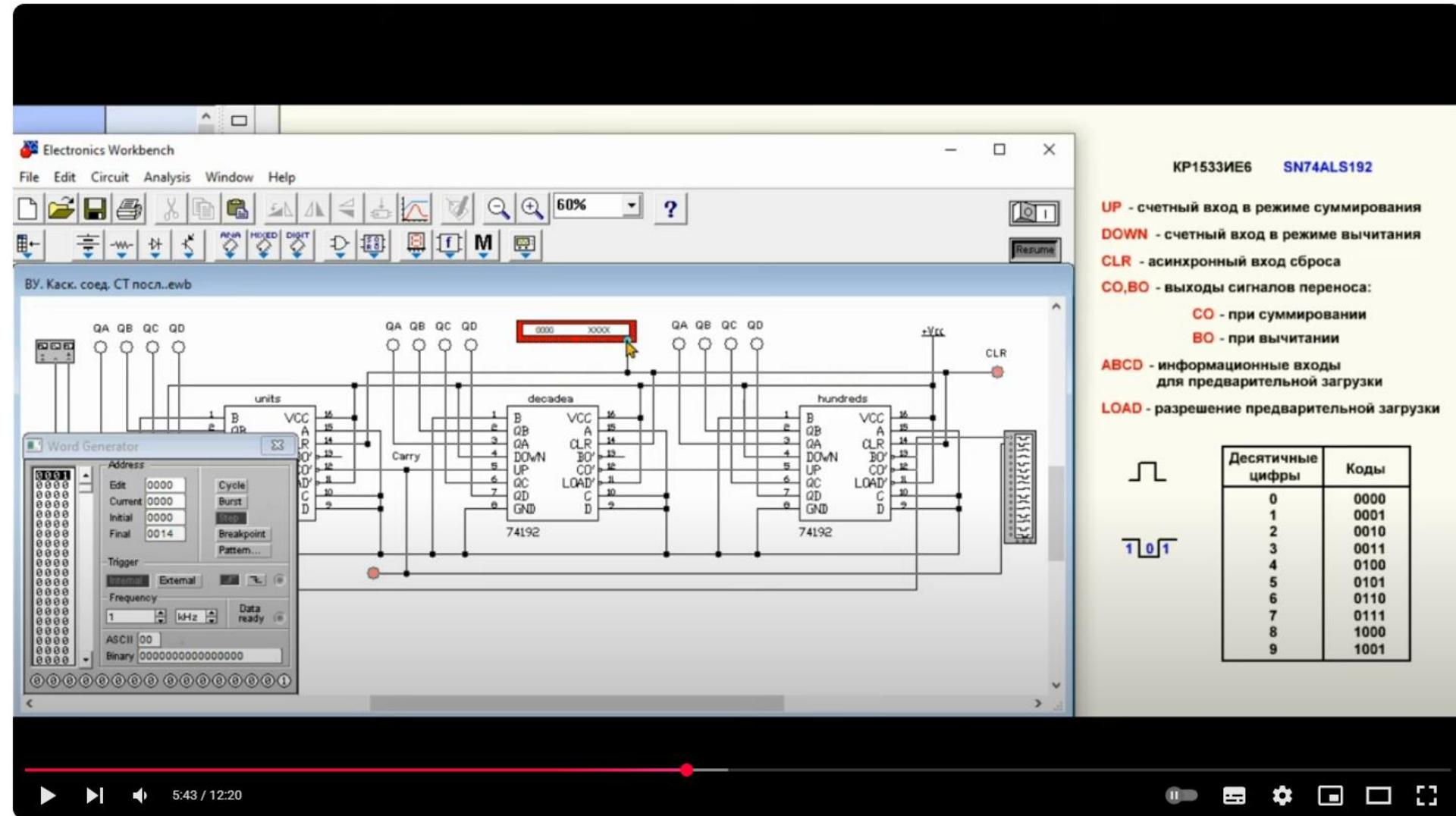
LOAD - разрешение предварительной загрузки

CO, BO - выходы сигналов переноса: **CO** - при суммировании

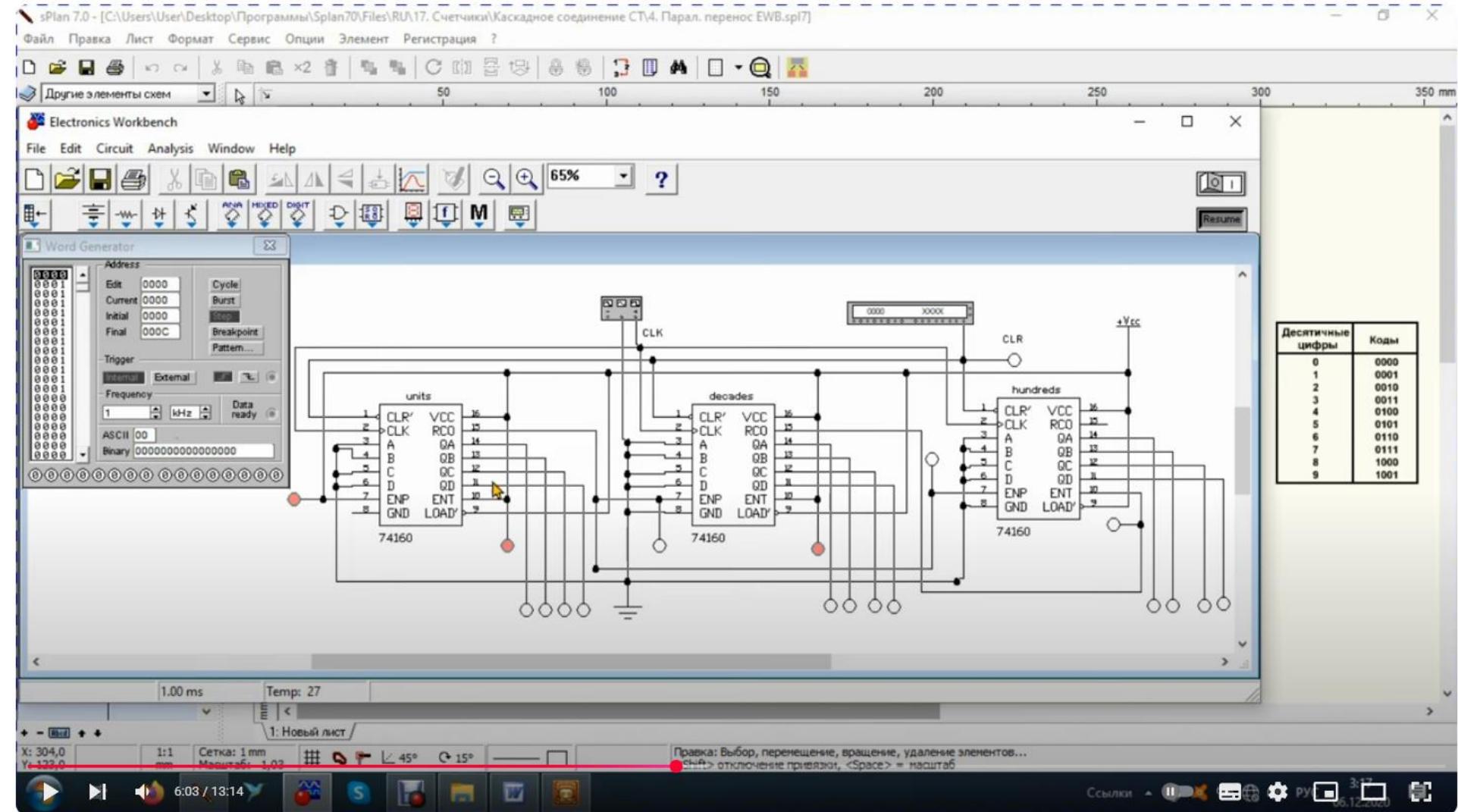
BO - при вычитании



Видеоурок 4.35. Каскадное соединение счетчиков. Часть 1. (2025)
<https://www.youtube.com/watch?v=ZWoDK6fQ8AE>



Вideoурок 4.36. Каскадное соединение счетчиков. Часть 2. (2025)
<https://www.youtube.com/watch?v=k3tMPUtfB3M>



Вideoурок 4.37. Каскадное соединение счетчиков. Часть 3. (2025)
https://www.youtube.com/watch?v=8XB_SKjmCkk

Счетчики с параллельным переносом

T, D, JK, RS

счетный вход - **C (CLK)**вычитающий **N = 14**начальное состояние - **0**результат подсчета - **двоичный код**последний импульс - **возврат в 0**

1. Определяем назначение и количество основных выводов счетчика, количество и вид триггеров в счетчике

один счетный вход **C (CLK)**

вход сброса **R**

из $2^n \geq N$ количество триггеров $n = 4 \Rightarrow 4$ выхода **Q0, Q1, Q2, Q3**

JK-триггеры

2. Составляем таблицу истинности счетчика

Таблица переходов

Переходы	J	K
$0 \rightarrow 0$	0	x
$0 \rightarrow 1$	1	x
$1 \rightarrow 0$	x	1
$1 \rightarrow 1$	x	0

J - вход установки

K - вход сброса

J, K - прямые статические. Активный сигнал - **1**



Вideoурок 4.29. Счетчики с параллельным переносом. Часть 1. (2025)
https://www.youtube.com/watch?v=ogSICHiI_MA

Счетчики с параллельным переносом

3. Определяем порядок заполнения карт Вейча

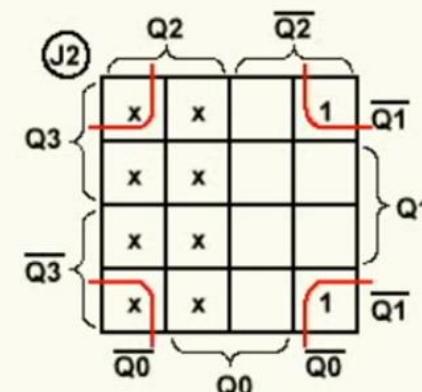
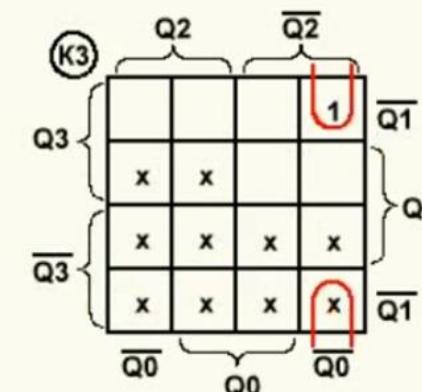
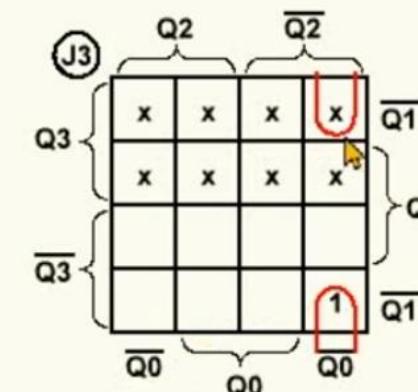
видеоурок 3.20

Номер входного импульса	Результат подсчета				Сигналы на входах триггеров									
	Десятичными числами		В двоичном коде на выходах		J3	K3	J2	K2	J1	K1	J0	K0		
			Q3	Q2										
1	0	1	0	0	1	x	1	x	0	x	1	x		
2	13	1	1	0	1	x	0	x	0	x	x	1	1	
3	12	1	1	0	0	x	0	x	1	x	1	x		
4	11	1	0	1	1	x	0	0	x	x	0	x	1	
5	10	1	0	1	0	x	0	0	x	x	1	1	x	
6	9	1	0	0	1	x	0	0	x	0	x	x	1	
7	8	1	0	0	0	x	1	1	x	1	x	1	x	
8	7	0	1	1	1	0	x	x	0	x	0	x	1	
9	6	0	1	1	0	0	x	x	0	x	1	1	x	
10	5	0	1	0	1	0	x	x	0	0	x	x	1	
11	4	0	1	0	0	0	x	x	1	1	x	1	x	
12	3	0	0	1	1	0	x	0	x	x	0	x	1	
13	2	0	0	1	0	0	x	0	x	x	1	1	x	
14	1	0	0	0	1	0	x	0	x	0	x	x	1	

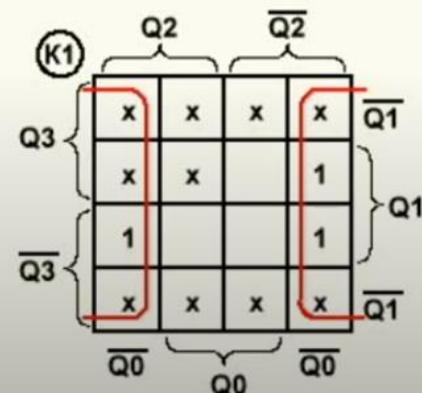
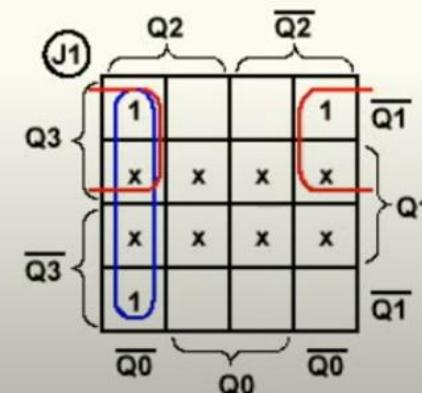
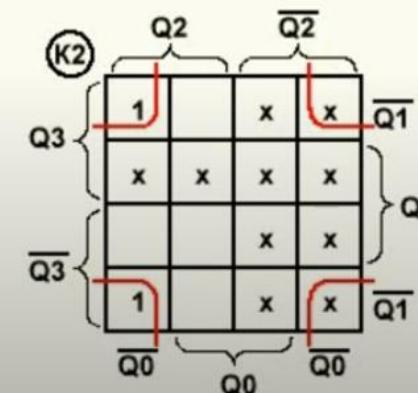


Видеоурок 4.30. Счетчики с параллельным переносом. Часть 2. (2025)
<https://www.youtube.com/watch?v=mCyNDChkJaA>

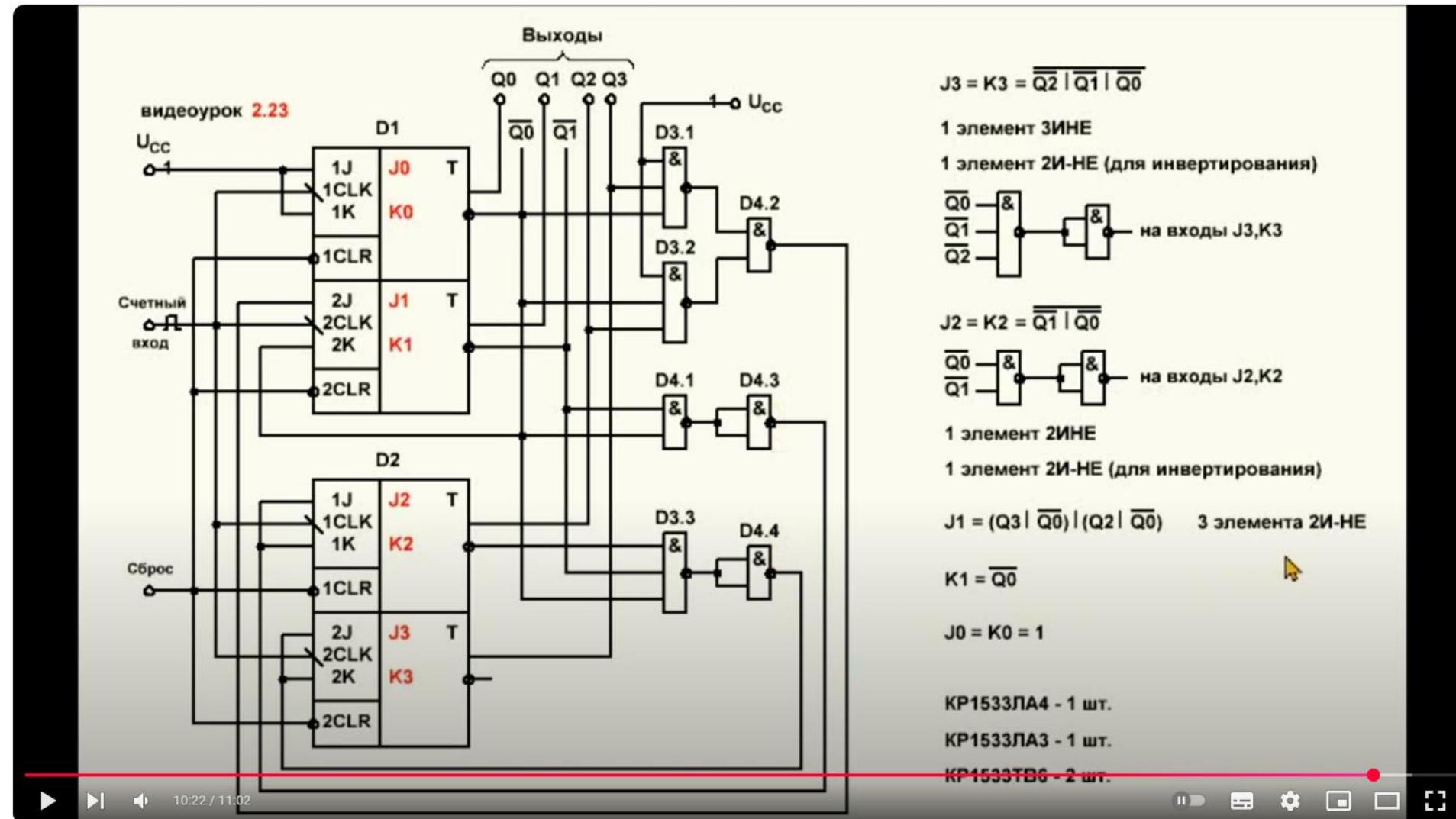
6. Выполняем операции "склеивания"



МДНФ:



Видеоурок 4.31. Счетчики с параллельным переносом. Часть 3. (2025)
<https://www.youtube.com/watch?v=IkPJPzYJBRM>



Видеоурок 4.32. Счетчики с параллельным переносом. Часть 4. (2025)
<https://www.youtube.com/watch?v=Lw76fyoNMpc>

**Вычитающий счетчик
с коэффициентом счета $N = 14$**

Номер входного импульса	Результат подсчета			
	В двоичном коде на выходах			
Десятичными числами	Q3	Q2	Q1	Q0
1	0	0	0	0
2	13	1	1	0
3	12	1	1	0
4	11	1	0	1
5	10	1	0	0
6	9	1	0	1
7	8	1	0	0
8	7	0	1	1
9	6	0	1	0
10	5	0	1	0
11	4	0	1	0
12	3	0	0	1
13	2	0	0	1
14	1	0	0	1

Видеоурок 4.33. Счетчики с параллельным переносом. Часть 5. (2025)
https://www.youtube.com/watch?v=_MYmYTKgi0g

**Счетчики
с последовательным переносом**

Суммирующий
двоичный счетчик

KP1533ИЕ5 SN74LS93

1. Счетчик-делитель на 2
2. Счетчик-делитель на 8
3. Счетчик-делитель на 16

R01,R02 - входы сброса

**Счетчики
с параллельным переносом**

Суммирующий
декадный (двоично-десятичный)
счетчик

KP1533ИЕ2 SN74LS90

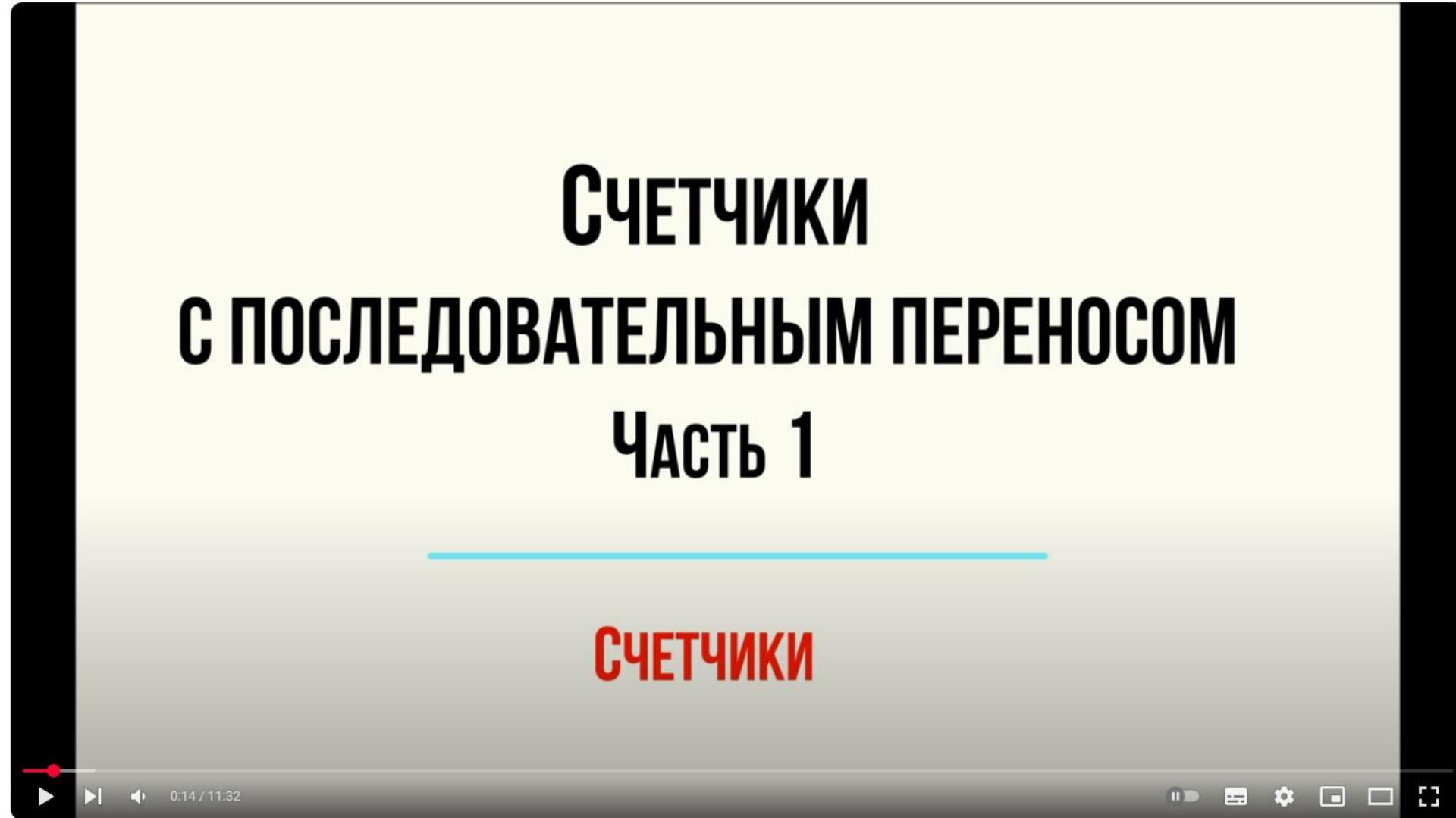
видеоурок 4.25 САПР

1. Счетчик-делитель на 2
2. Счетчик-делитель на 5
3. Счетчик-делитель на 10

R91,R92 - входы установки 9 = код 1001

▶ ▶ 音量 5:11 / 12:30
⏸ ⏹ ⚙️ ▢ ▢

Вideoурок 4.34. Счетчики с параллельным переносом. Часть 6. (2025)
<https://www.youtube.com/watch?v=B8BWV1nW3Gk>

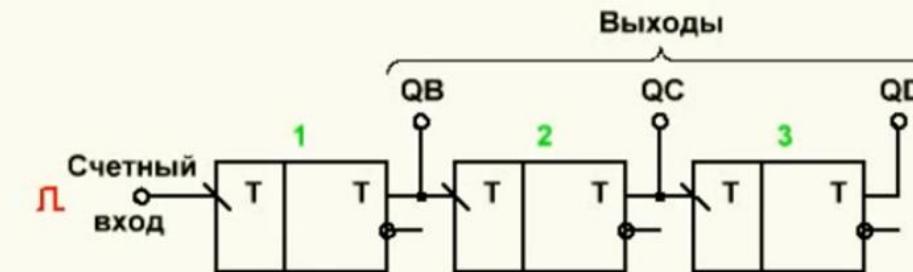


Вideoурок 4.25. Счетчики с последовательным переносом. Часть 1. (2024)
<https://www.youtube.com/watch?v=yPpqYIZpEbg>

Счетчики с последовательным переносом

Двоичные счетчики. Т-триггеры с динамическим входом Т

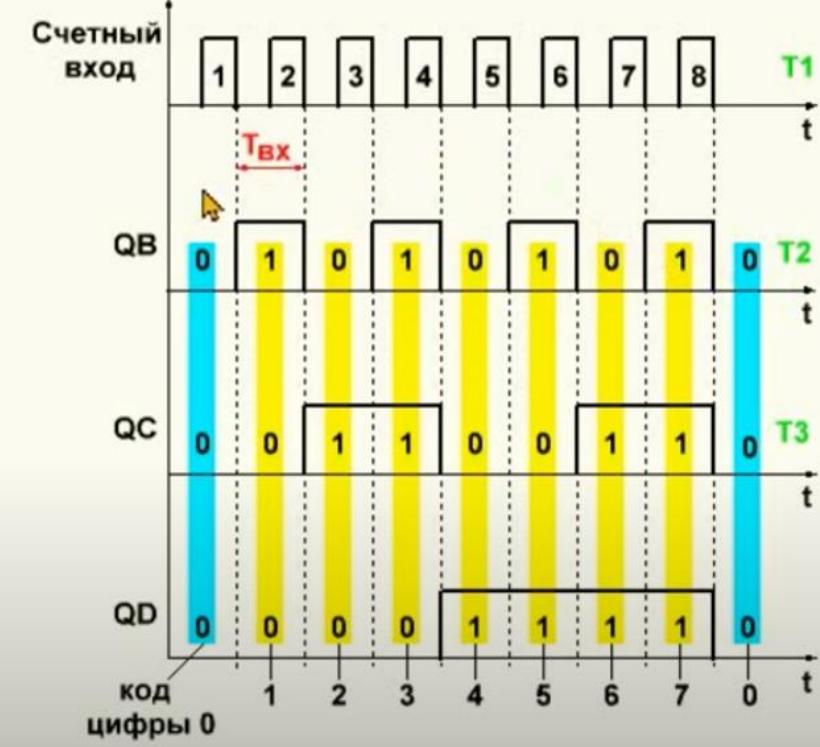
**Суммирующий 3-разрядный двоичный счетчик
с коэффициентом счета $N = 2^3 = 8$**



Десятичные цифры	Двоичные коды
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

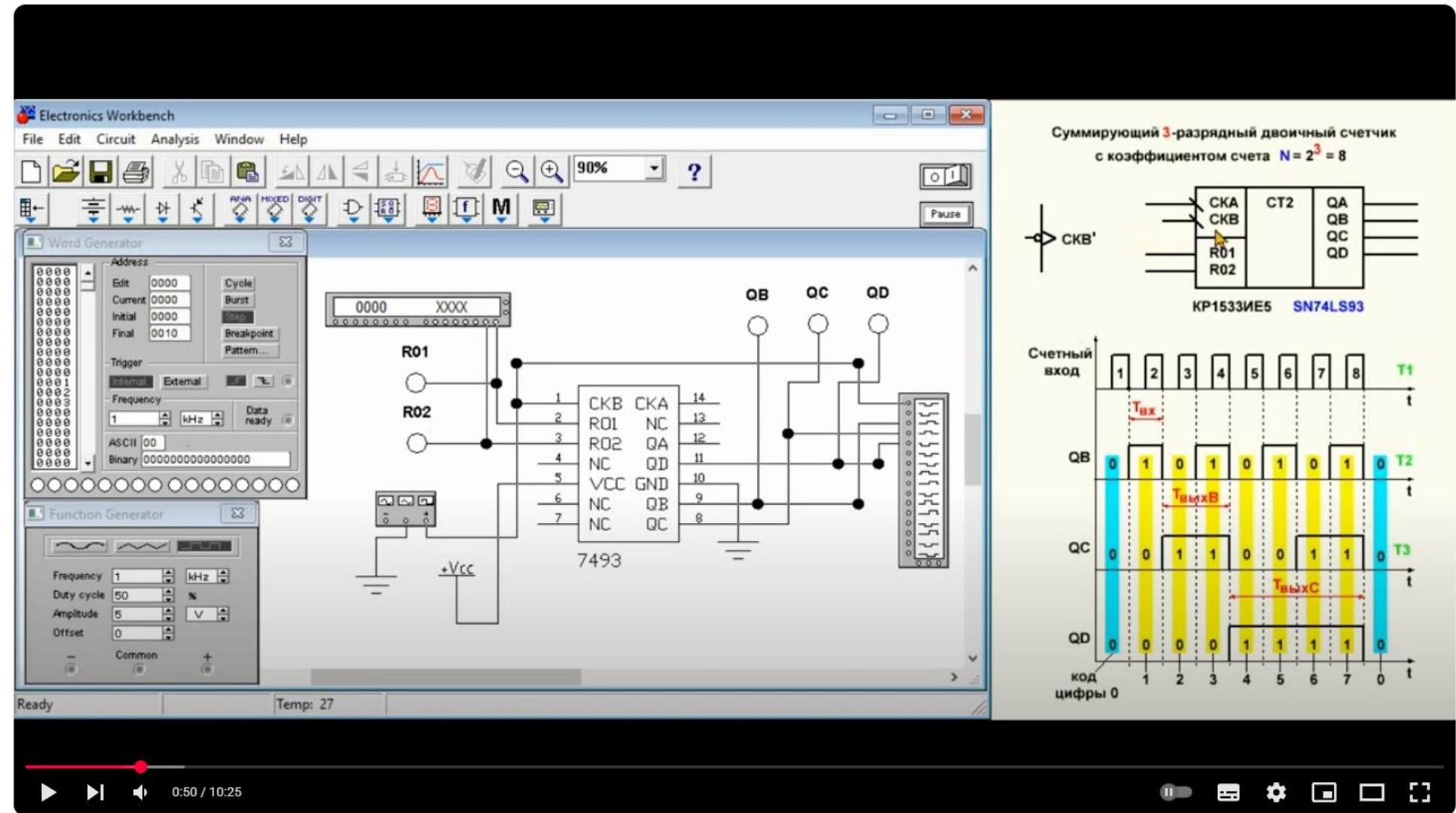
T (toggle - переключатель) –
счетный вход

KP1533ИЕ5



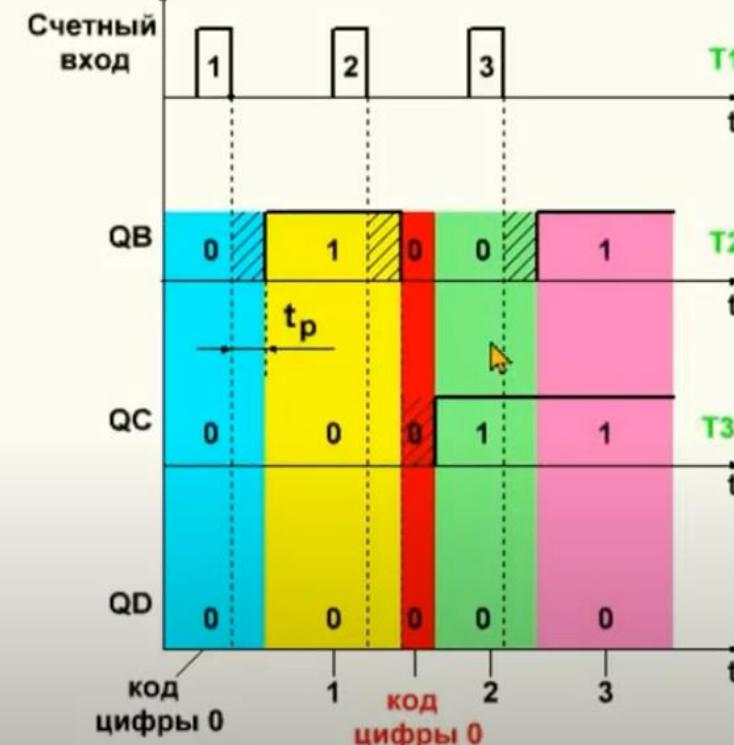
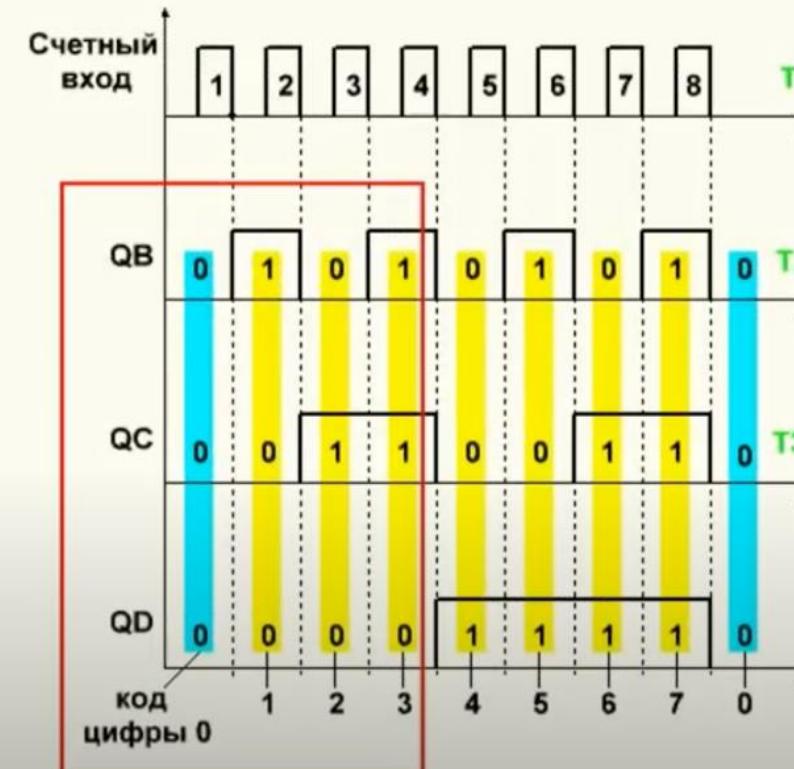
▶ ▶ ⏴ 0:47 / 7:28

Видеоурок 4.26. Счетчики с последовательным переносом. Часть 2. (2024)
<https://www.youtube.com/watch?v=xkqmy9d7tyM>



Вideoурок 4.27. Счетчики с последовательным переносом. Часть 3. (2025)
<https://www.youtube.com/watch?v=IBczil0TzzE>

Счетчики с последовательным переносом

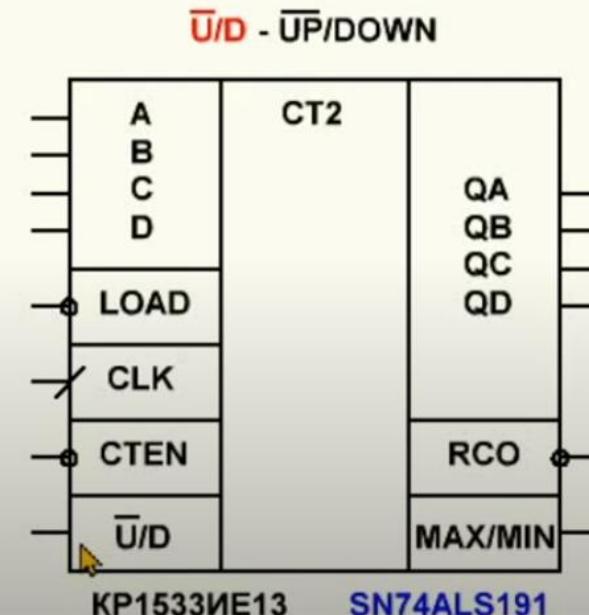
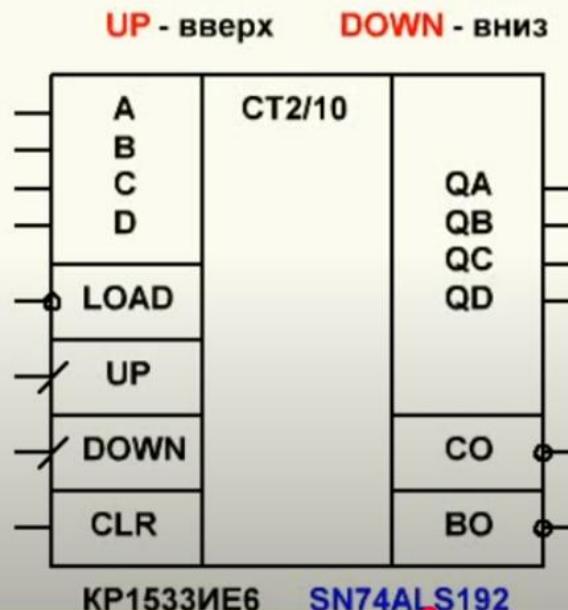
 t_p 

Вideoурок 4.28. Счетчики с последовательным переносом. Часть 4. (2025)
<https://www.youtube.com/watch?v=lqbYER53LxE>

Классификация счетчиков

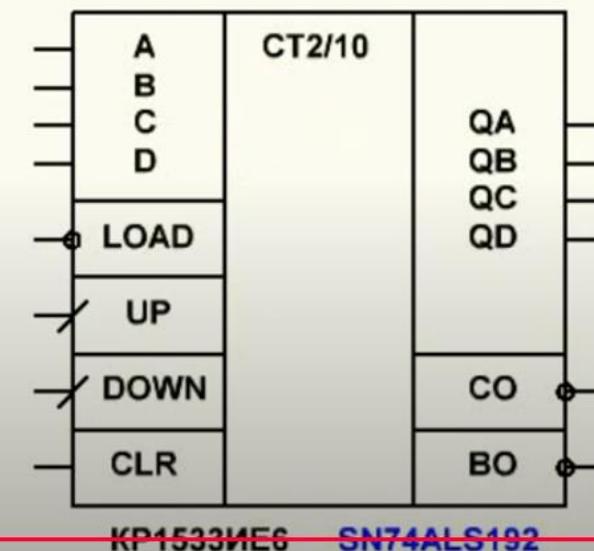
По направлению счета:

1. Суммирующие (прямого счета)
2. Вычитающие (обратного счета)
3. Реверсивные



Видеоурок 4.24. Классификация счетчиков (2024)
https://www.youtube.com/watch?v=_W3XySn43Ow

Счетчики. Основные понятия.



считывают импульсы

коэффициент счета
(цикл счета, модуль счета,
коэффициент деления, емкость)

N (или M, или P)

основные элементы - интегральные триггеры
n - количество триггеров (разрядность счетчика)

$$2^n \geq N$$

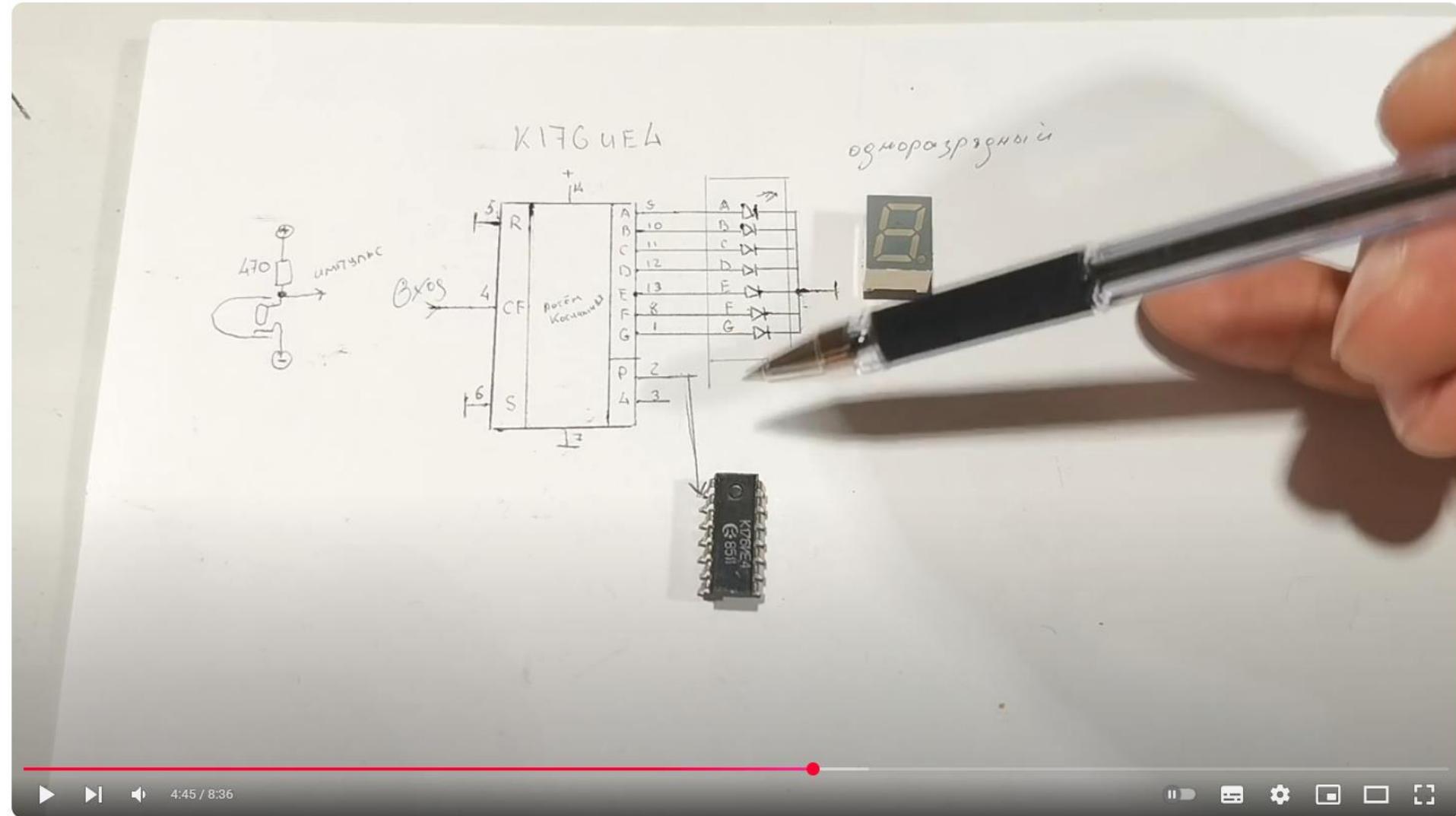
T- или JK-триггеры

в маркировке - ИЕ

в изображении - СТ



Видеоурок 4.23. Цифровые счетчики. Основные понятия. (2024)
https://www.youtube.com/watch?v=dm_mZIRZ3xA



Микросхема-счетчик. Как это работает на примере микросхемы к176иe4 (2020)
<https://www.youtube.com/watch?v=qpSgM7yvFtg>

Методика ПоИТ.pdf

70

11. Практическая работа №11:
синтез четырехразрядного счетчика с параллельным переносом между разрядами двумя способами

1. Постановка задачи

Разработать счетчик с параллельным переносом на D-триггерах двумя способами:

- с оптимальной схемой управления, выполненной на логических элементах общего базиса;
- со схемой управления, реализованной на преобразователе кодов (быстрая реализация, но не оптимальная схема).

В качестве исходных данных использовать индикатор CNT лабораторного комплекса, на котором слева направо отображены:

- направление счета (0 — сложение, 1 — вычитание);
- максимальное значение счетчика (не путать с модулем счета);
- шаг счета.

Протестировать работу схемы и убедиться в ее правильности. Подготовить отчет о проделанной работе и защитить ее.

2. Последовательность выполнения работы

1. Перед выполнением работы ознакомиться с приложениями 1-3.
2. Запустить лабораторный комплекс и получить персональные исходные данные для этой работы. Для этого требуется установить правильное название группы и ФИО студента. Рекомендуется зафиксировать личный код. В качестве исходных данных использовать показания индикатора CNT.

Информатика, 11 практическая работа - Счётчики (2023)
<https://www.youtube.com/watch?v=ILAj0qytUWo>

70

Практическая работа № 10

Построение счетчика с произвольным модулем счета

1. Постановка задачи

На основе заданного варианта построить схему счетчика с произвольным модулем счета на Т-триггерах.

Необходимо, исходя из задания, определить последовательность счета. Затем построить таблицу переходов состояний счетчика. На основе таблицы переходов сформировать таблицу управления Т-триггерами.

Все незадействованные состояния счетчика в процессе его проектирования должны быть проинициализированы такими значениями, которые позволяют сократить элементы в схеме управления.

Используя навыки, полученные в работе 1, необходимо сформировать логические функции управления Т-триггерами с учетом незадействованных состояний.

Используя навыки, полученные при выполнении работ №№ 6–9 построить схему счетчика с индикацией его значений.

Готовые схема и отчет демонстрируются преподавателю в аудитории, с проверкой правильности их оформления и понимания функционирования схемы студентом. После чего соответствующая отметка заносится в тетрадь.

2. Последовательность выполнения работы

- Сформировать последовательность счета счетчика.
- Написать таблицу переходов для сформированной последовательности.
- Написать таблицу управления Т-триггерами по таблице переходов.
- Отметив все незадействованные значения символом * провести минимизацию функций управления каждым триггером, аналогично решению работы № 1. При этом надо учитывать следующие особенности:
 - При склейке двух строк, отмеченных символом *, этот символ остается. При склейке строки, отмеченной символом *, со строкой без отметки —

ИИИ - ИРИ: Информатика, 10 практическая работа - Счётчики (2023)
<https://www.youtube.com/watch?v=DvOf3erQpvA>

Содержание

1. Понятие счетчика.
2. Классификация счетчиков.
3. Двоичные счетчики с последовательным переносом.
4. Счетчики с параллельным переносом.
5. Метод управляемого сброса.
6. Разбор задания №3 курсовой работы.
7. Домашнее задание.
8. Контрольные вопросы.



Счетчики (2022)

<https://www.youtube.com/watch?v=mJ3LFgeVlVE>

Introduction to Counters

ff as divide by 2 clk:

$$f_B = \frac{f_C}{2}$$

$$T_B = 2 T_A$$

$$\frac{1}{f_B} = \frac{2}{f_A}$$

$$f_B = \frac{f_A}{2} = \frac{f_C}{4}$$

$$2^4 = 16$$

P of ff $P=2$ $2^2 = 4$

Sq clk

$T_A = 2 T_C \Rightarrow \frac{1}{f_A} = \frac{2}{f_C} \Rightarrow f_A = \frac{f_C}{2}$

Types of Counters

Ripple / Asynchronous Counters

synchronous Counters

COUNTERS (As./S.)

UP counters Down cont. UP|Down cont.

0 - 1 - 2 - 3 - ... 7 - 6 - 5 - ... 0

Asynchronous/Ripple Counter

1. Flip flops are connected in such a way that the o/p of first flip flop drives the clock of next flip flop.

Synchronous Counter

1. There is no connection between o/p of first flip flop and clock of next flip flop.

▶ ▶ ⏴ 6:44 / 7:06

|| ⏴ ⏵ ⏵ ⏵ ⏵

Types of Counters | Comparison between Ripple and Synchronous counters (2015)
 Типы счетчиков | Сравнение пульсационных и синхронных счетчиков (2015)
<https://www.youtube.com/watch?v=yqglsqhZG3M>

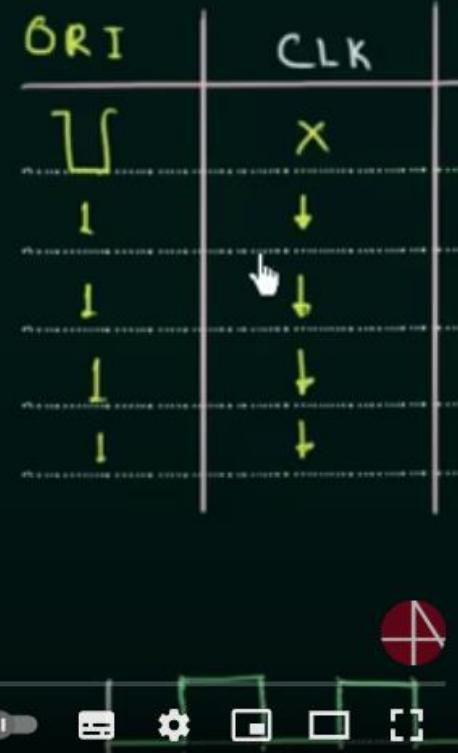
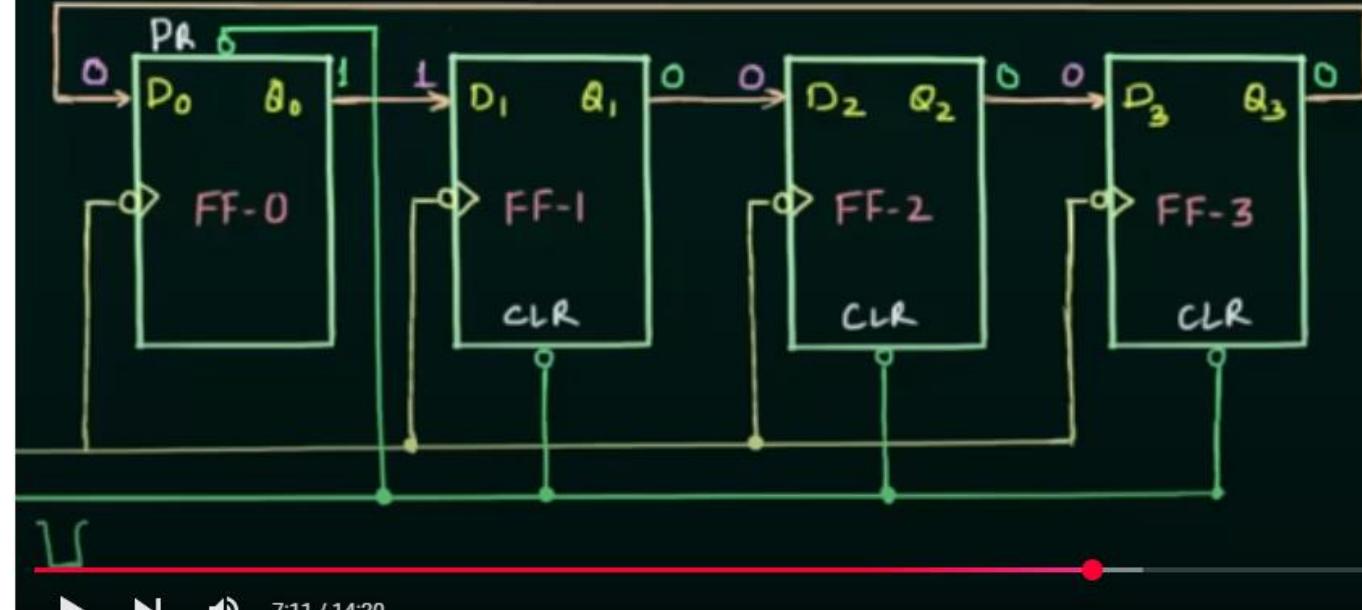
= Ring Counter

$$= 2^4 : 16 \quad * \text{no of states} = \text{no. of bits used}$$

counter is a typical application of shift register

only change is the output of last ff is connected to the input of first ff.

$$\begin{array}{ll} PR = 0 & Q = 1 \\ CLR = 0 & Q = 0 \end{array}$$



Ring Counter (2015)
Счетчик звонков (2015)
<https://www.youtube.com/watch?v=yOW-JsJL1Ks>

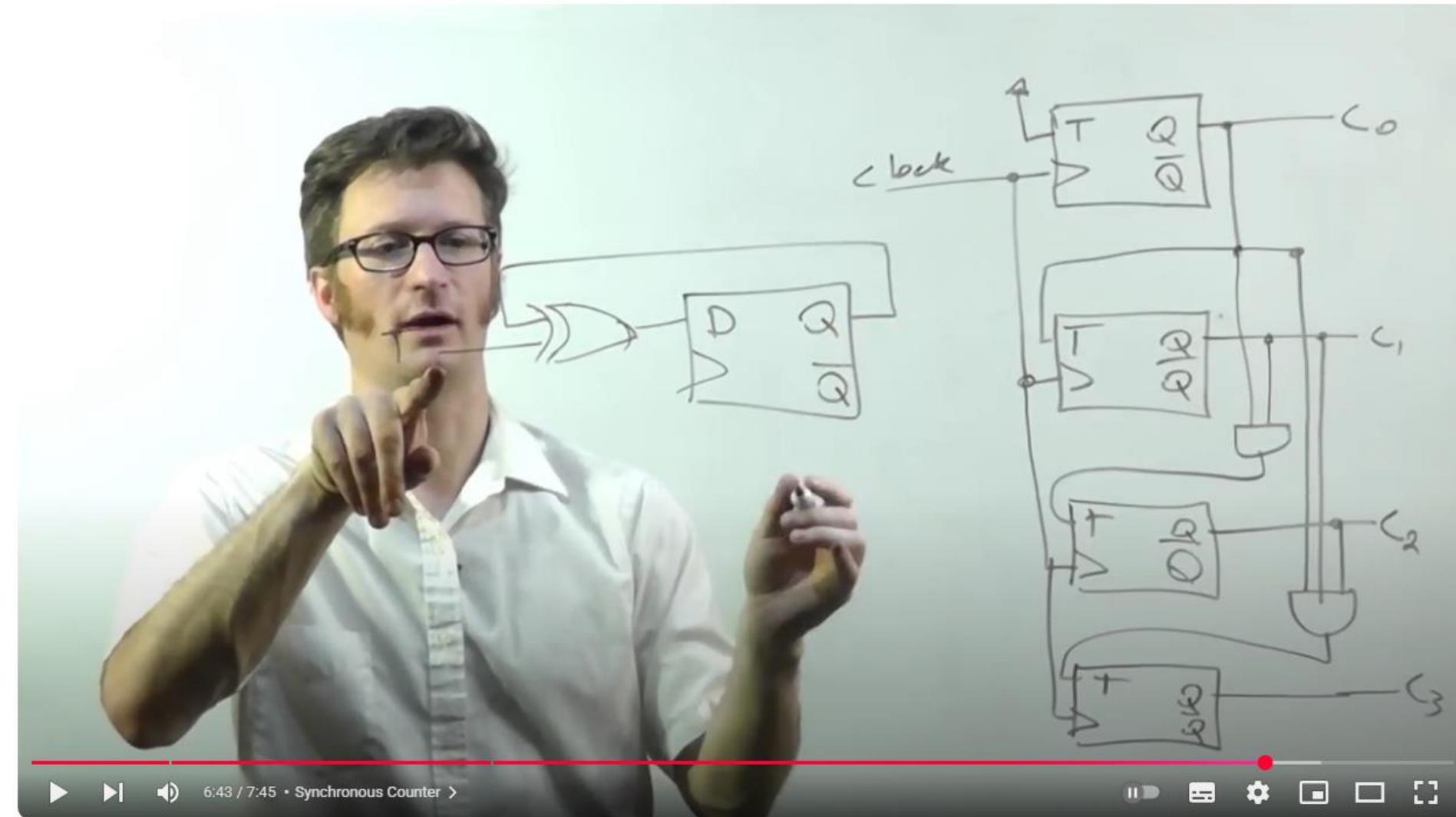


Digital Electronics

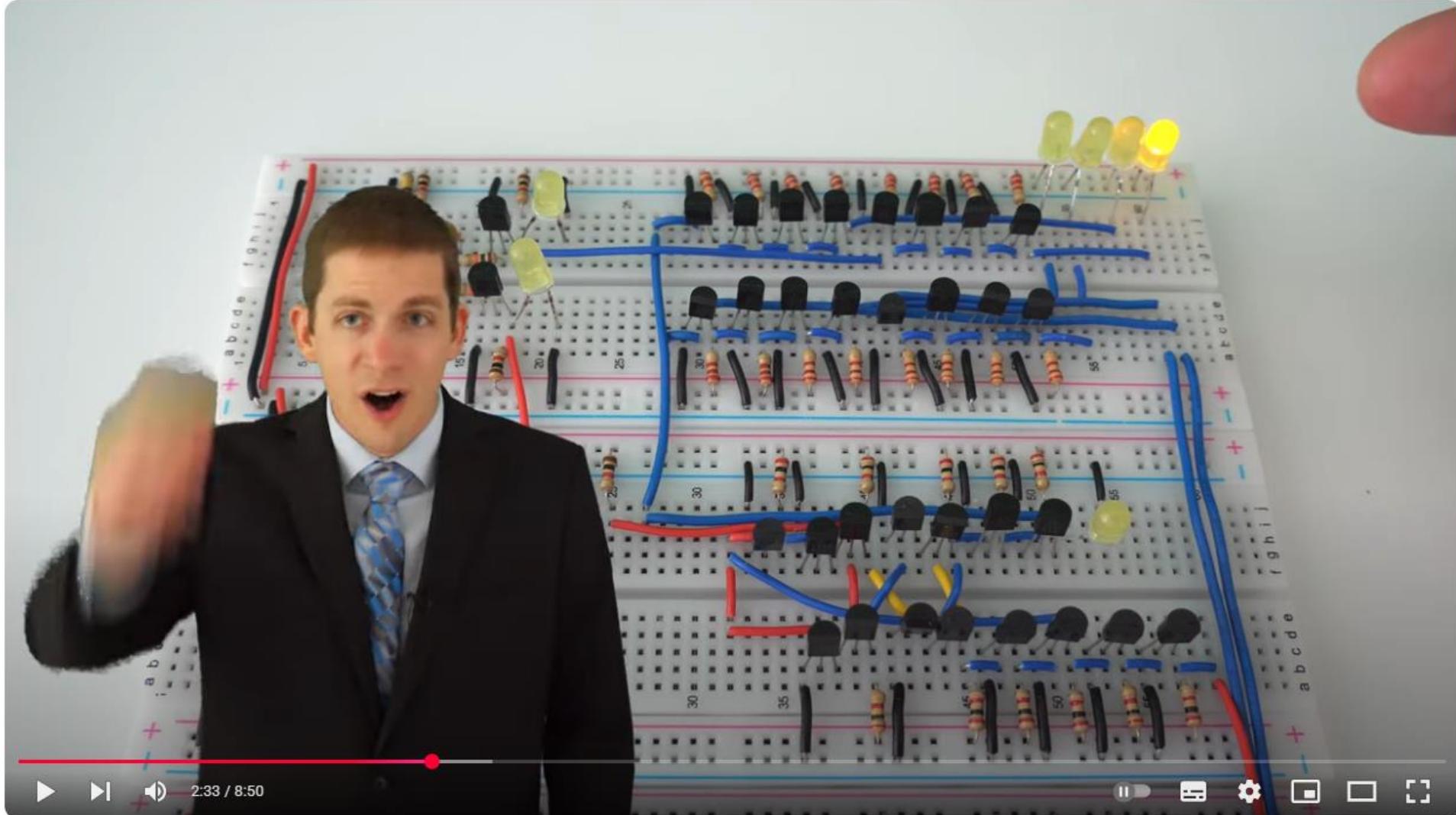
Synchronous Counter

The image shows a YouTube video player interface. At the top left is the YouTube logo and the time '25:58'. The main title 'Digital Electronics' is in light blue at the top center. A large yellow box contains the subtitle 'Synchronous Counter'. The video progress bar at the bottom shows a red playhead at 0:11 of a 25:58 minute video titled 'What is Synchronous Counter'. To the right of the playbar are video controls: a volume icon, a 'next' button, and a 'repeat' button. Below the playbar is the channel name 'ALL ABOUT ELECTRONICS' in a semi-transparent box, along with other channel icons. The background of the video player is dark.

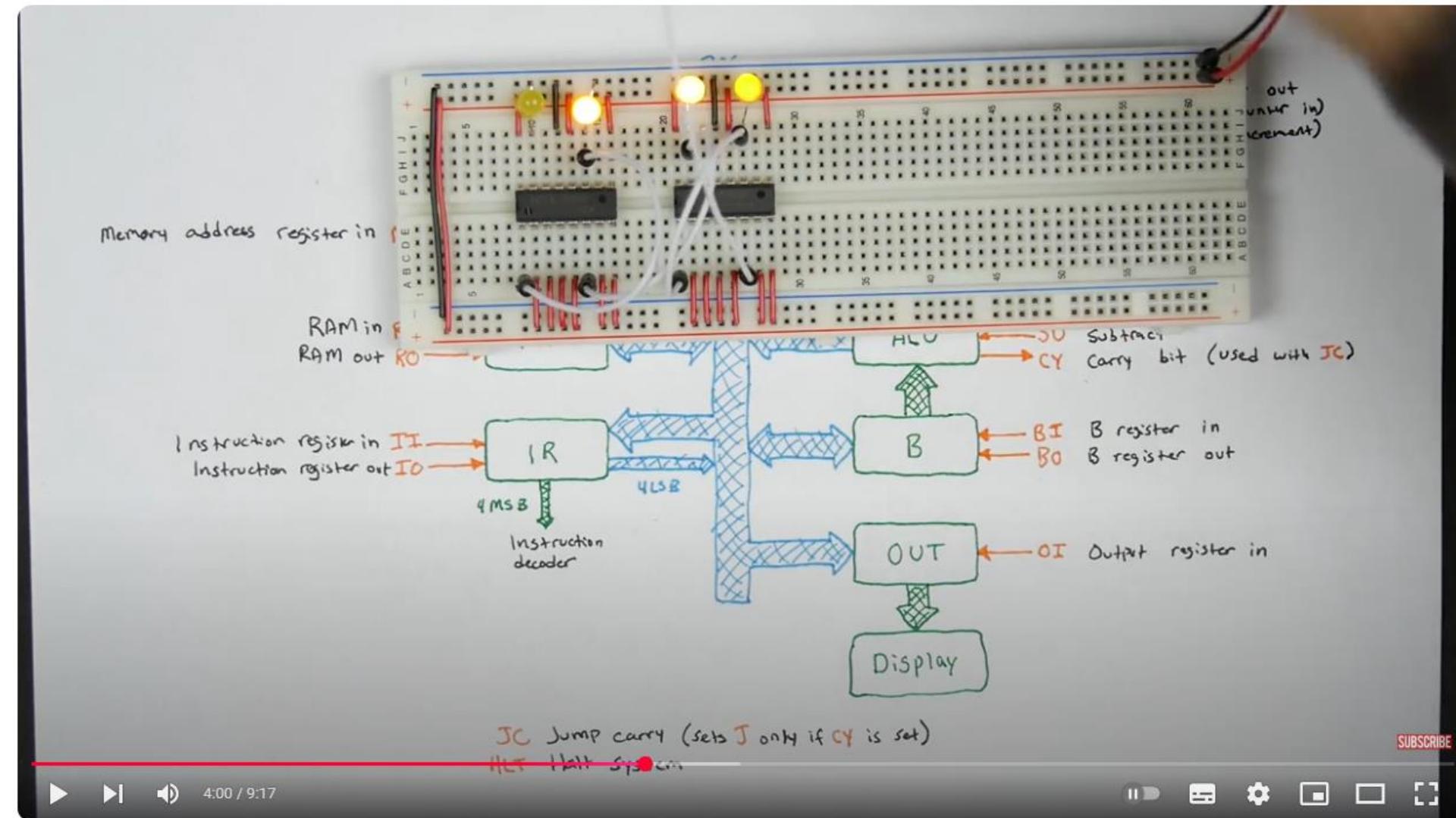
Synchronous Counters Explained (Part-1) (2022)
Объяснение работы синхронных счетчиков (Часть 1) (2022)
<https://www.youtube.com/watch?v=dlianWwz-Fw>



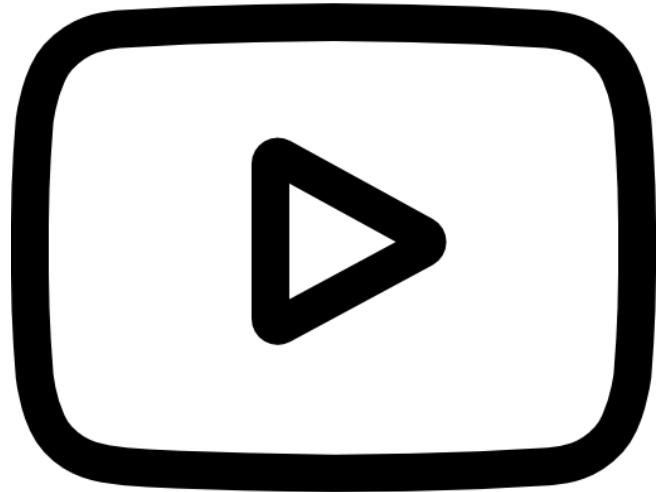
Digital Logic – Counters (2013)
Цифровые логические счетчики (2013)
<https://www.youtube.com/watch?v=ZiAbLltaz4A>



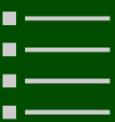
Binary Counter (2023)
Двоичный счетчик (2023)
<https://www.youtube.com/watch?v=UKIUIaojVqI>

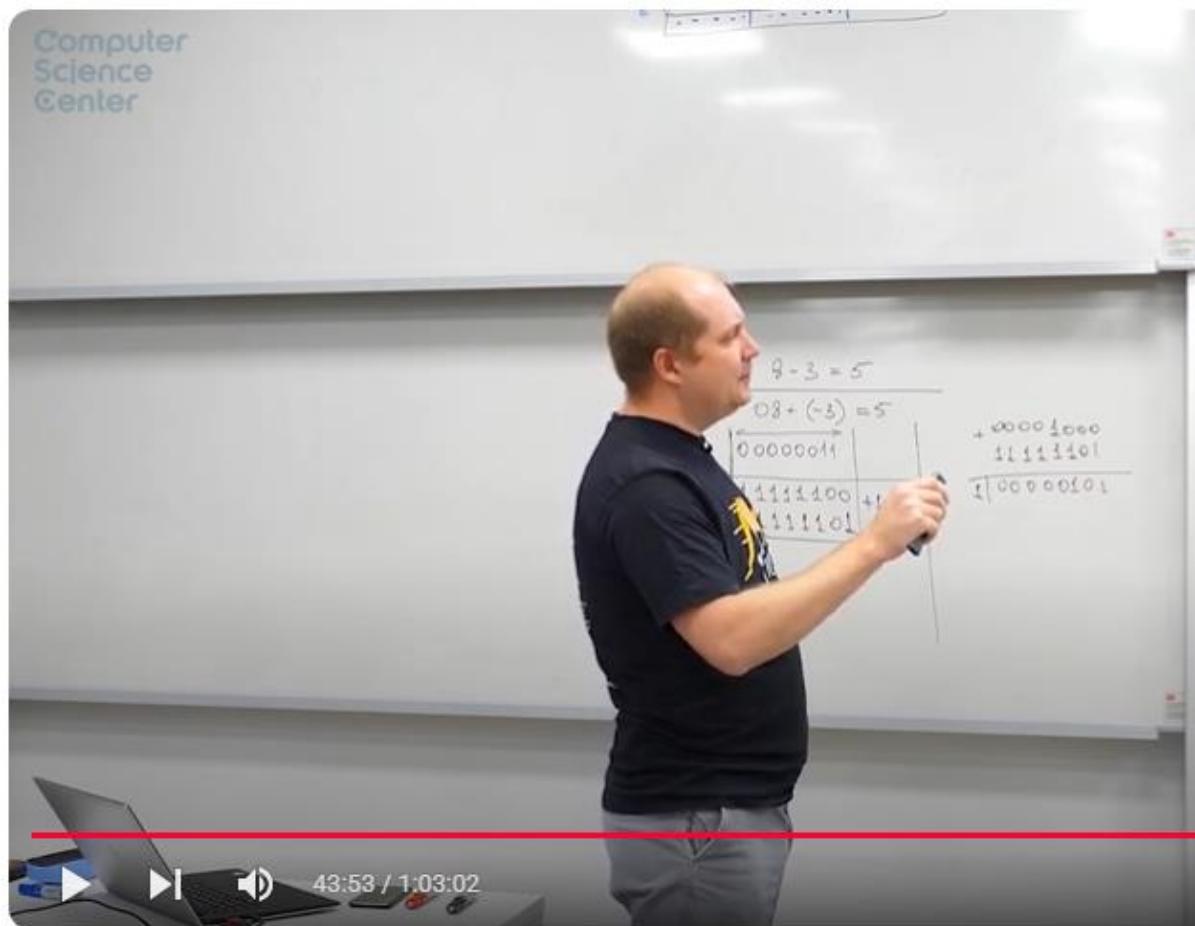


Program counter design (2017)
Разработка программного счетчика (2017)
https://www.youtube.com/watch?v=g_1HyxBzjI0

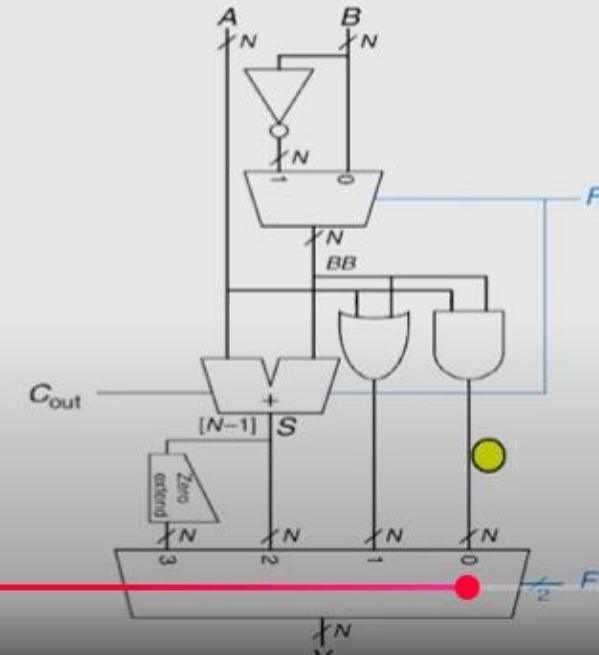


Арифметико- логические устройства (АЛУ)





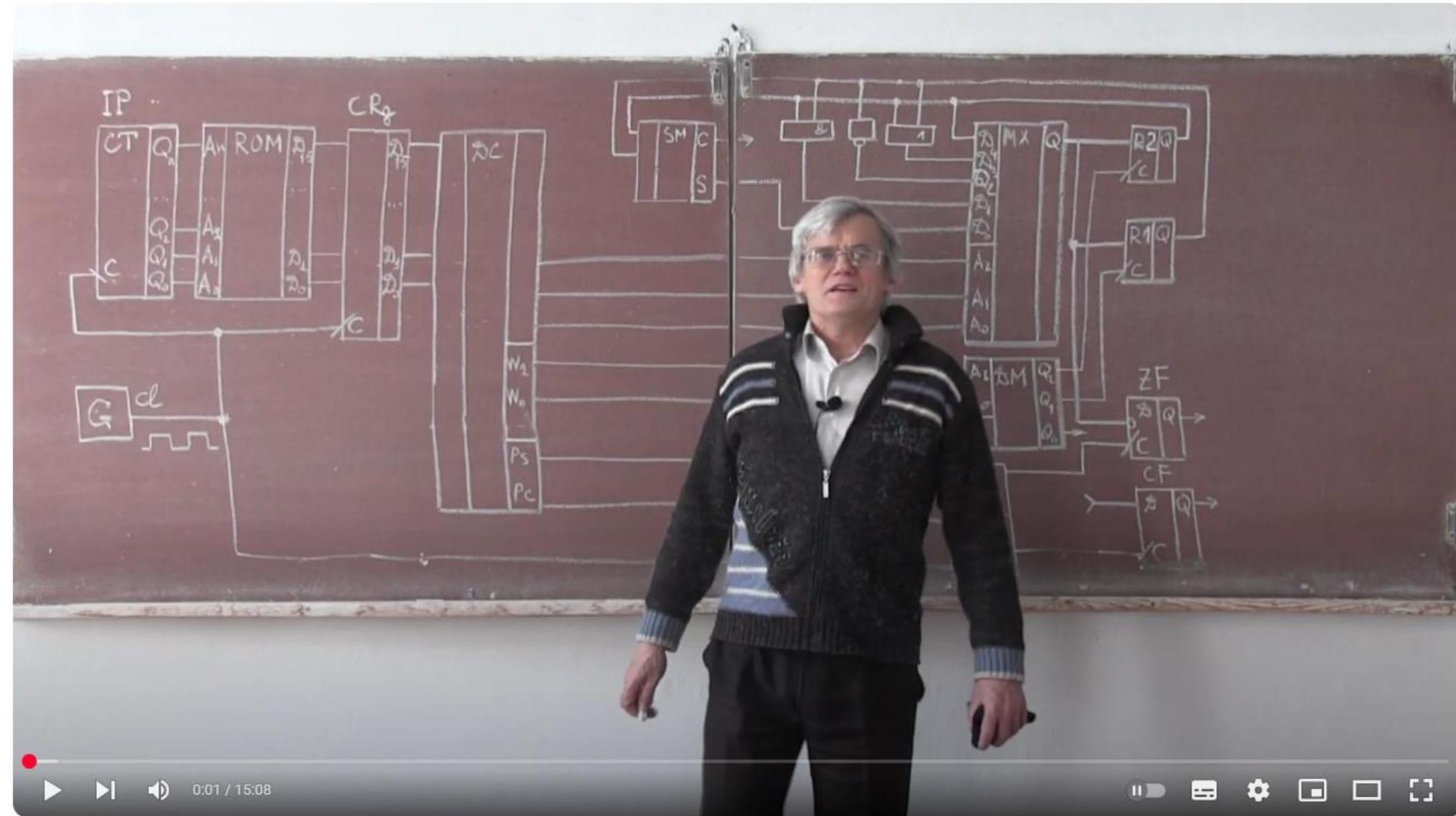
Арифметико-логическое устройство (ALU)



$F_{2:0}$	Function
000	A AND B
001	A OR B
010	A + B
011	not used
100	A AND \bar{B}
101	A OR \bar{B}
110	A - B
111	SLT



Архитектура ЭВМ. Лекция 2: АЛУ. Устройство памяти (2021)
<https://www.youtube.com/watch?v=HPGw4qFARTg>



Лекция 323. Atmega 8: Принцип работы АЛУ (2016)
<https://www.youtube.com/watch?v=EcksJCdIQog>

АРИФМЕТИКО-ЛОГИЧЕСКИЕ УСТРОЙСТВА (АЛУ)

ЧАСТЬ 1

Комбинационные ЦУ



Видеоурок 3.39. Арифметико-логические устройства (АЛУ). Часть 1 (2023)
<https://www.youtube.com/watch?v=O-90dCQKTrY>

Арифметико-логические устройства (АЛУ)

The diagram shows the SN74LS181 ALU with its pin connections. The control inputs are M, S0, S1, S2, S3, and C_N. The data inputs are A and B. The outputs are F0, F1, F2, F3, C_{N+4}, G, and P. The ALU symbol is shown with a diamond output.

1. $\bar{A} \vee B$

$A = 0001$
 $B = 0110$

$$\begin{array}{r} \bar{A} = 1110 \\ \bar{A} \vee B = \begin{array}{r} 1110 \\ \vee 0110 \\ \hline 1110 \end{array} \end{array}$$

2. $A \cdot \bar{B} + A$

$A = 1001$
 $B = 0100$

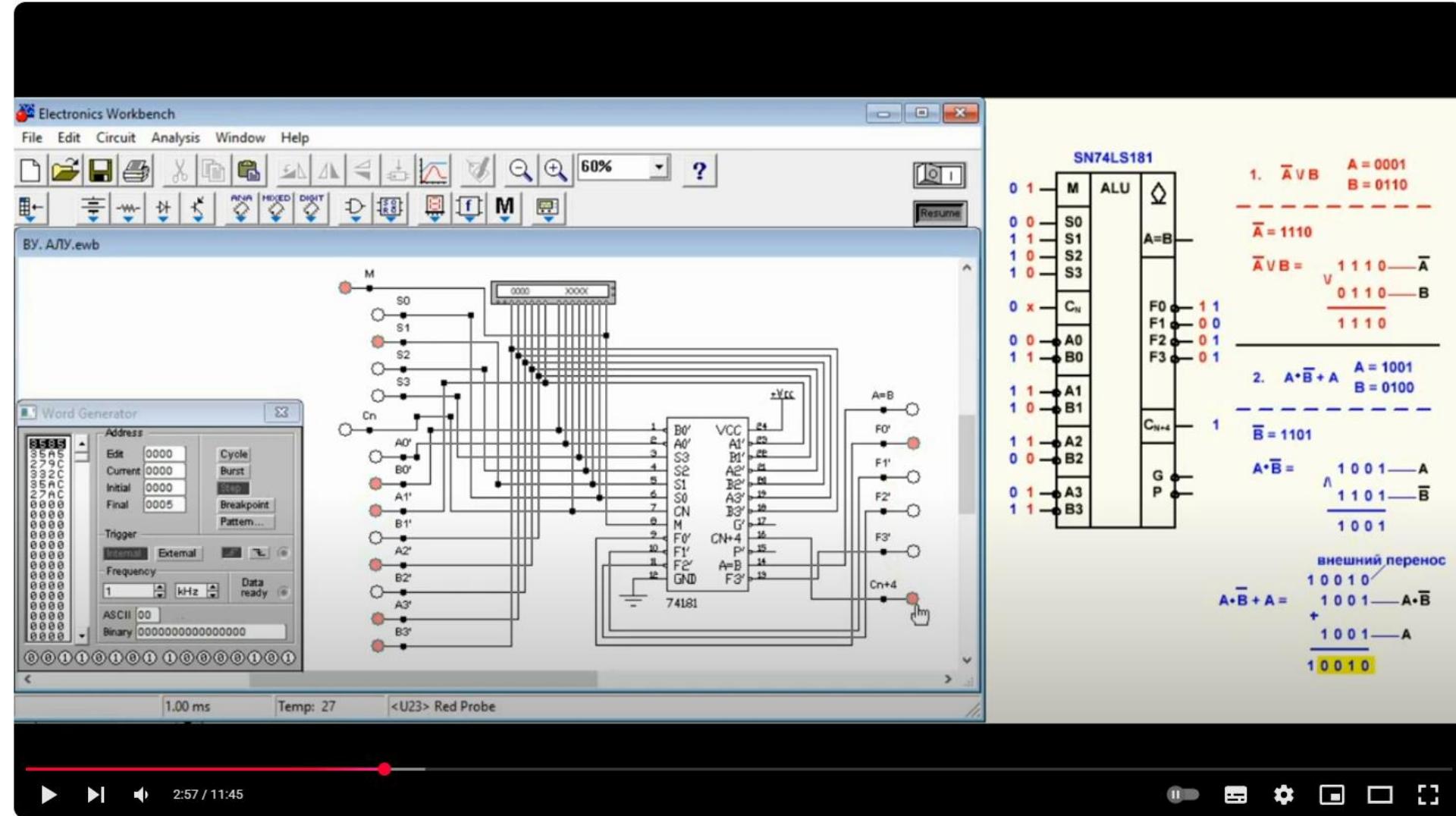
$$\begin{array}{r} \bar{B} = 1101 \\ A \cdot \bar{B} = \begin{array}{r} 1001 \\ \wedge 1101 \\ \hline 1001 \end{array} \end{array}$$

$$\begin{array}{r} \text{внешний перенос} \\ A \cdot \bar{B} + A = \begin{array}{r} 1001 \\ + 1001 \\ \hline 10010 \end{array} \end{array}$$

Выбор функции				Выполняемая функция		
S3	S2	S1	S0	Логические операции при M = 1	Арифметические операции при M = 0	
0	0	0	0	\bar{A}	$A - 1$	A
0	0	0	1	$\bar{A} \cdot B$	$A \cdot B - 1$	$A \cdot B$
0	0	1	0	$\bar{A} \vee B$	$A \cdot \bar{B} - 1$	$A \cdot \bar{B}$
0	0	1	1	1	-1	0
0	1	0	0	$\bar{A} \vee B$	$A + (A \vee \bar{B})$	$A + (A \vee \bar{B}) + 1$
0	1	0	1	\bar{B}	$A \cdot B + (A \vee \bar{B})$	$A \cdot B + (A \vee \bar{B}) + 1$
0	1	1	0	$\bar{A} \oplus B$	$A - B - 1$	$A - B$
0	1	1	1	$\bar{A} \vee \bar{B}$	$A \vee \bar{B}$	$(A \vee \bar{B}) + 1$
1	0	0	0	$\bar{A} \cdot B$	$A + (A \vee B)$	$A + (A \vee B) + 1$
1	0	0	1	$\bar{A} \oplus B$	$A + B$	$A + B + 1$
1	0	1	0	B	$A \cdot \bar{B} + (A \vee B)$	$A \cdot \bar{B} + (A \vee B) + 1$
1	0	1	1	$\bar{A} \vee B$	$A \vee B$	$(A \vee B) + 1$
1	1	0	0	0	$A + A$	$A + A + 1$
1	1	0	1	$\bar{A} \cdot \bar{B}$	$A \cdot B + A$	$A \cdot B + A + 1$
1	1	1	0	$\bar{A} \cdot B$	$A \cdot \bar{B} + A$	$A \cdot \bar{B} + A + 1$
1	1	1	1	A	A	A + 1

7:22 / 9:00

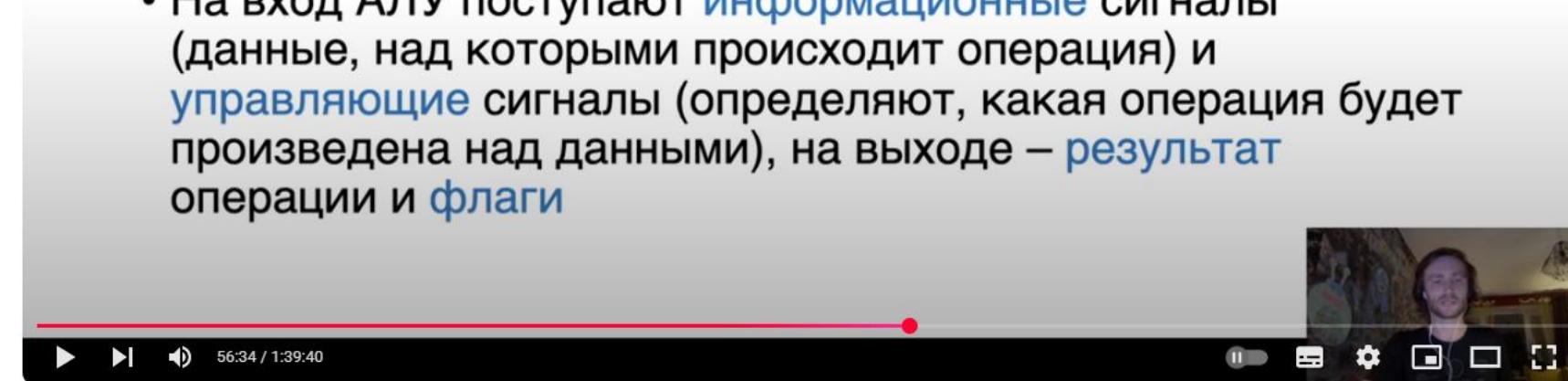
Видеоурок 3.40. Арифметико-логические устройства (АЛУ). Часть 2 (2023)
<https://www.youtube.com/watch?v=rEsbmJTqM7Y>



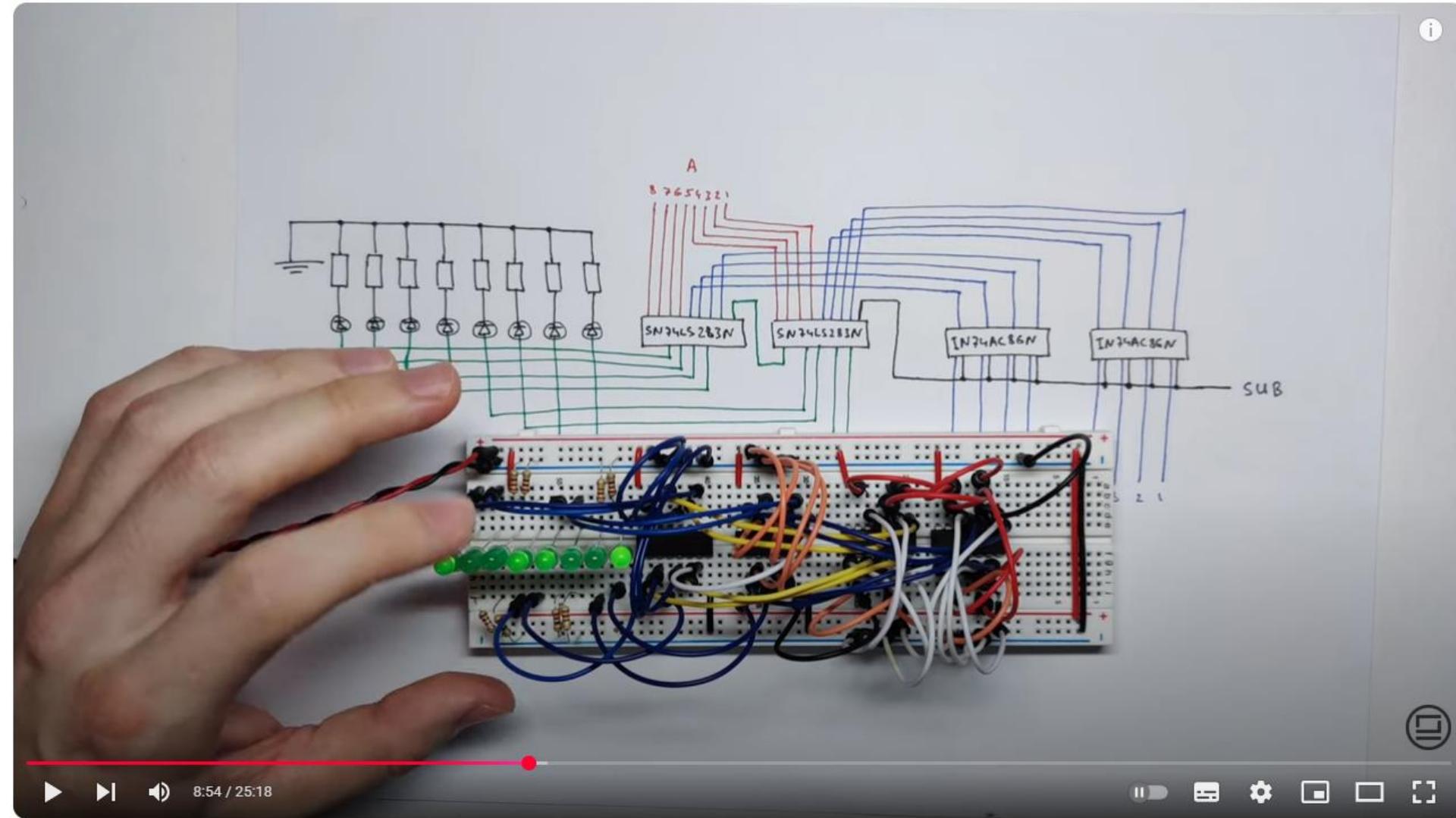
Видеоурок 3.41. Арифметико-логические устройства (АЛУ). Часть 3 (2023)
<https://www.youtube.com/watch?v=Pl785qcbKmk>

Арифметико-логическое устройство

- АЛУ – блок процессора, выполняющий арифметические и поразрядно логические операции
 - Арифметические операции имеют перенос
 - Логические операции без переноса
- АЛУ – комбинационная схема
- На вход АЛУ поступают **информационные сигналы** (данные, над которыми происходит операция) и **управляющие сигналы** (определяют, какая операция будет произведена над данными), на выходе – **результат операции и флаги**



АПС Л3. Цифровая арифметика. АЛУ (2020)
<https://www.youtube.com/watch?v=eXW6Q4jPkmQ>



BitFlip. Собираем 8-битное арифметико-логическое устройство (АЛУ)(2020)
https://www.youtube.com/watch?v=6P_-kfUIr8c

Nandgame Solve Level Levels Custom Components Playground Check solution Reset state Clear canvas Clear all levels Skip

ALU

The ALU (Arithmetic Logic Unit) combines the logic and arithmetic operations:

Input	Output
u op1 op0	
0 0 0	X and Y
0 0 1	X or Y
0 1 0	X xor Y
0 1 1	invert X
1 0 0	X + Y
1 1 0	X - Y
1 0 1	X + 1
1 1 1	X - 1

Two additional flags modify the operands:

- When the **sw** flag is 1, the **X** and **Y** inputs are swapped.
- When the **zx** flag is 1, the left operand is replaced with 0.

This affects the operands as shown here for the example of $X - Y$:

zx	sw	Effective operation
0	0	$X - Y$
0	1	$Y - X$
1	0	$0 - Y$
1	1	$0 - X$

Toolbox:

- nand
- logic unit
- arithmetic unit
- and 16
- add 16
- select 16
- inv 16
- 0
- decoder 1-2

Output: 0000 0000 0000 0000 Hex 0000 Dec 0

Input: u op1 op0 zx sw X Y

Output: 11 (000b) ✓ 3 ✓ -3 (ffffd) ✓ -4 (ffffc) ✓ -7 (ffff9) ✓ 0 ✓ 0 ✓ 0 ✓ 0 ✓ 0 ✓ 0 ✓ 177 (00b1) 1 ✓

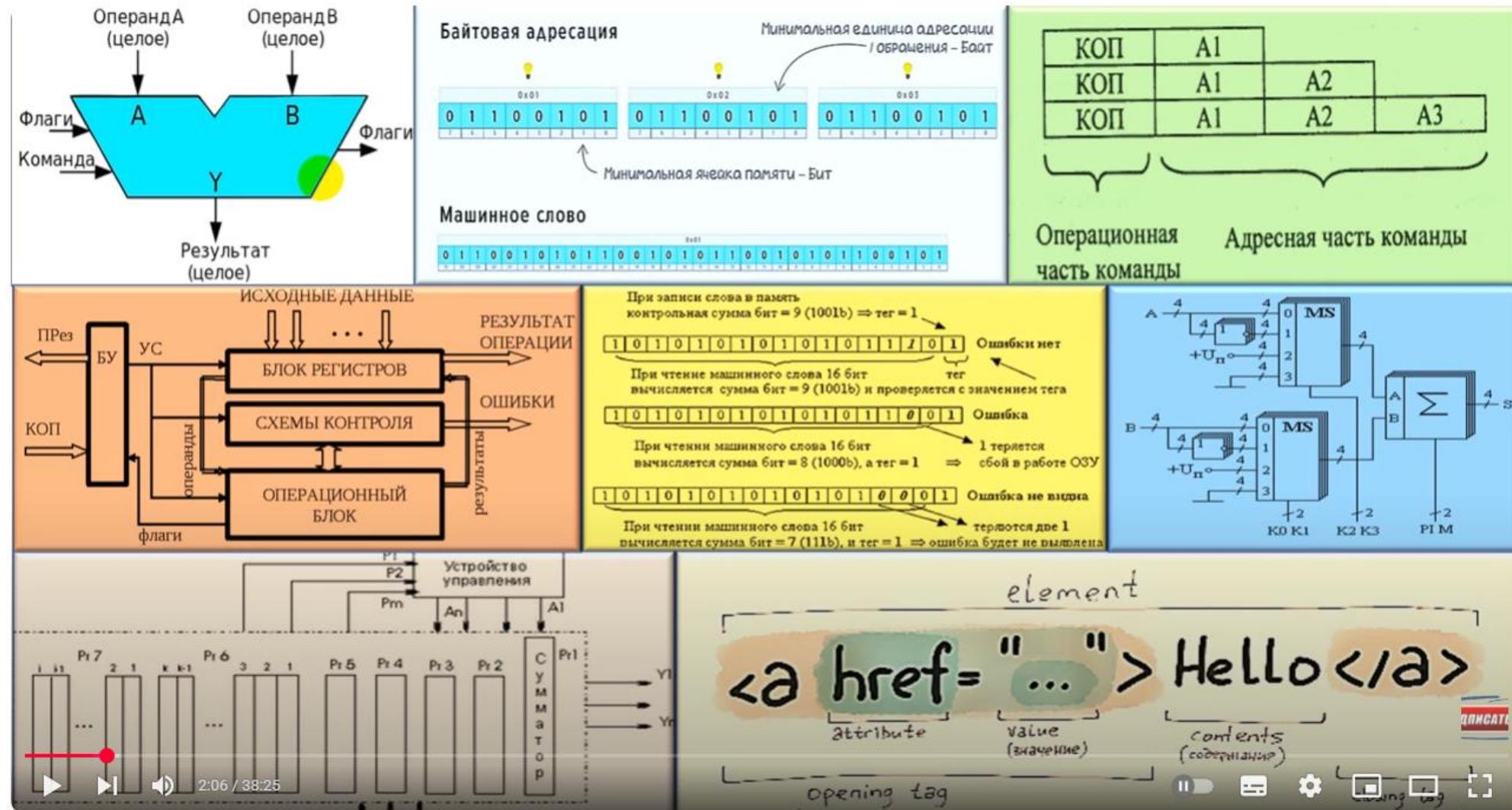
✓ Level successfully completed!

Levels in Arithmetic Logic Unit:

- ✓ Logic Unit
- ✓ Arithmetic Unit
- ✓ ALU
- Condition

 3 / 4 levels completed.

Арифметико-логическое устройство (2025)
<https://www.youtube.com/watch?v=X-P6Znxh-zQ>



29 АЛУ процессора (2024)

<https://www.youtube.com/watch?v=HNuraUZhcz4>

Донецкий национальный технический университет
Факультет компьютерных наук и технологий

Кафедра компьютерной инженерии

«АРИФМЕТИКО-ЛОГИЧЕСКИЕ ОСНОВЫ ЦИФРОВЫХ АВТОМАТОВ»

ЛЕКЦИЯ 3

Суммирующие и вычитающие схемы.
АЛУ для сложения и вычитания чисел
с фиксированной запятой. Сдвиговые операции



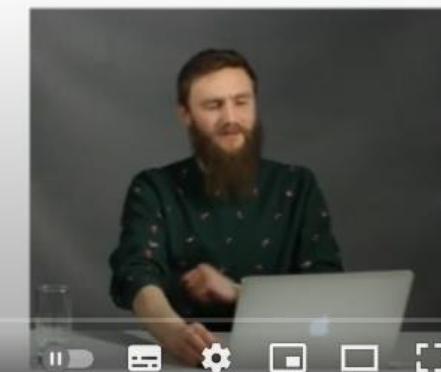
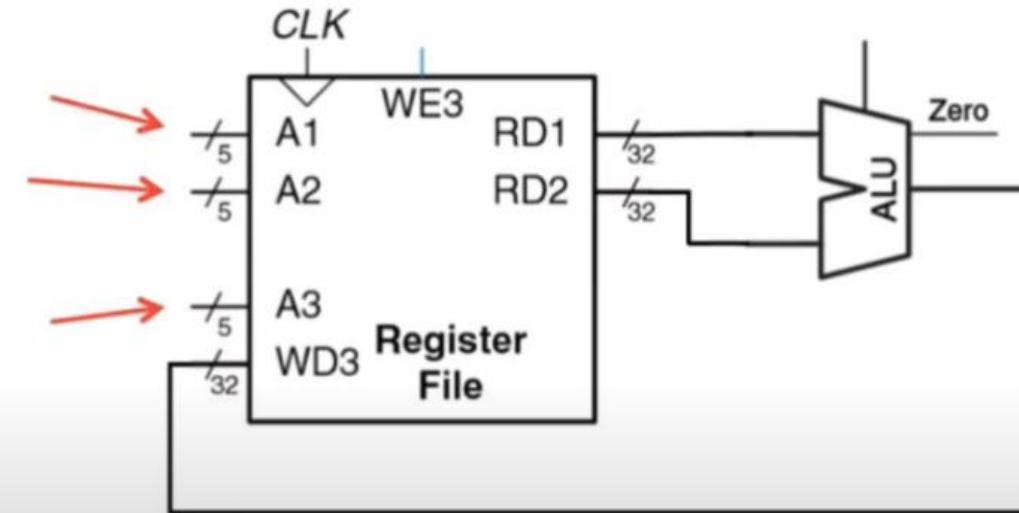
Иваница Сергей Васильевич,

доцент кафедры компьютерной инженерии

<http://masters.donntu.org/ivanitsa>

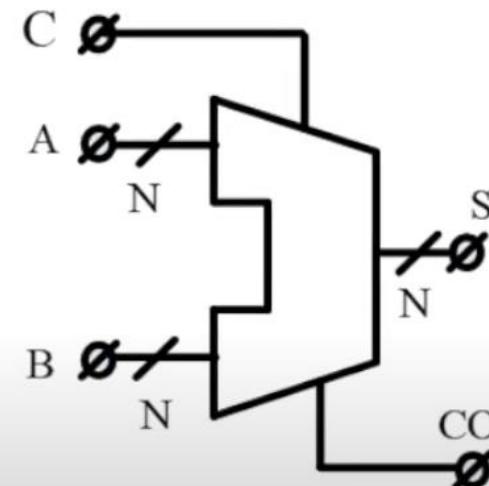


Видеолекция. Одноразрядные и многоразрядные суммирующие схемы (2021)
<https://www.youtube.com/watch?v=D6IzEOcOGy4>



Электронные мысли полупроводниковых машин или как работают процессоры (2019)
<https://www.youtube.com/watch?v=10We8RO8-xo>

АРИФМЕТИКО-ЛОГИЧЕСКОЕ УСТРОЙСТВО



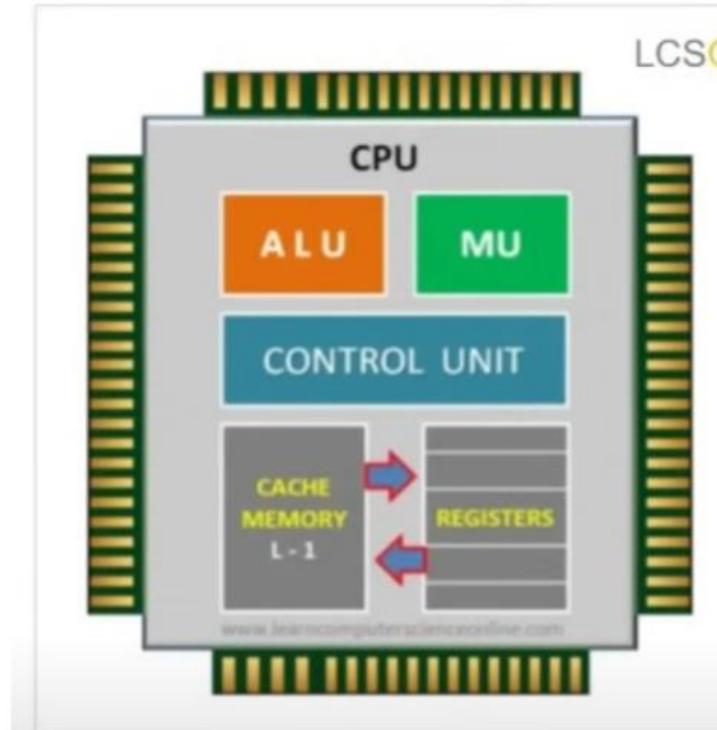
N – количество бит в двоичном числе

C-управление

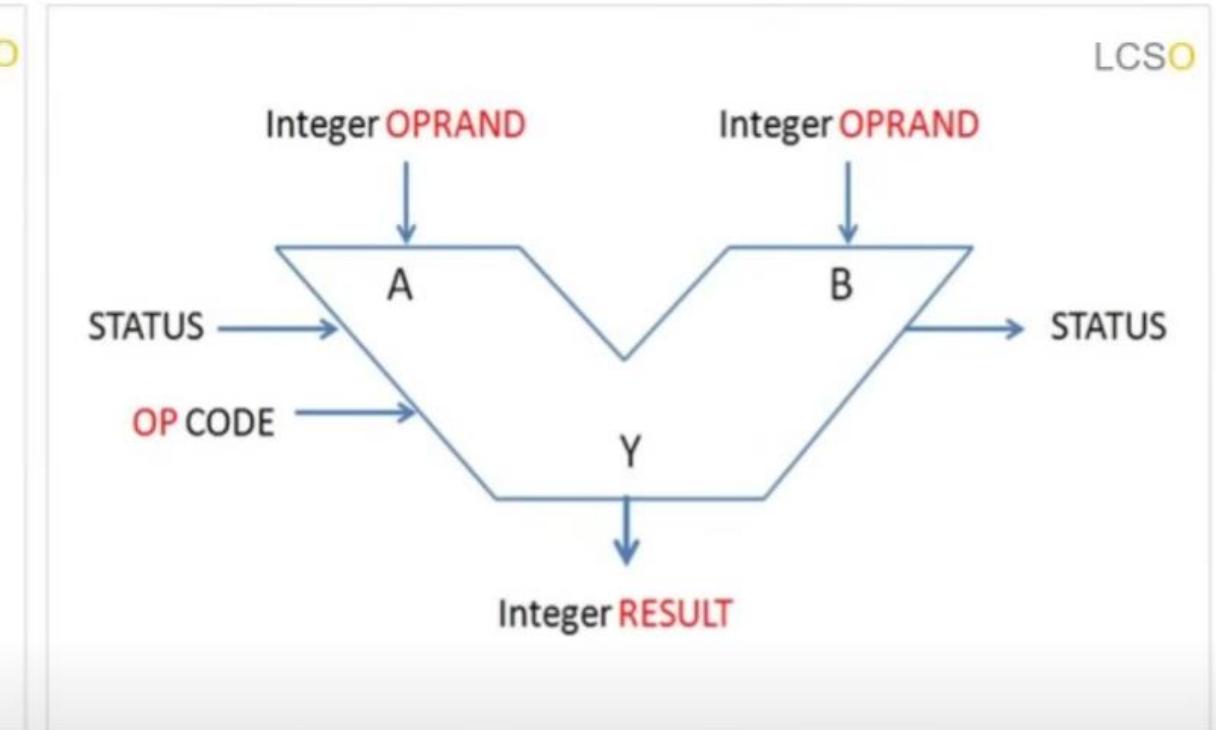
$$\begin{aligned} C=0, \quad S=A+B \\ C=1, \quad S=A-B \end{aligned}$$



Арифметико-логическое устройство. Сложение двоичных чисел. (2019)
<https://www.youtube.com/watch?v=ded0jOVZhSA>



The diagram illustrates a CPU architecture. It features a central grey block labeled 'CPU' containing an 'ALU' (Arithmetic Logic Unit) in orange, a 'MU' (Memory Unit) in green, and a 'CONTROL UNIT' in blue. Below the CPU are 'CACHE MEMORY L-1' and 'REGISTERS'. A red double-headed arrow connects the Cache and Registers. The entire CPU is mounted on a green printed circuit board (PCB) with gold-colored pins at the top and bottom.



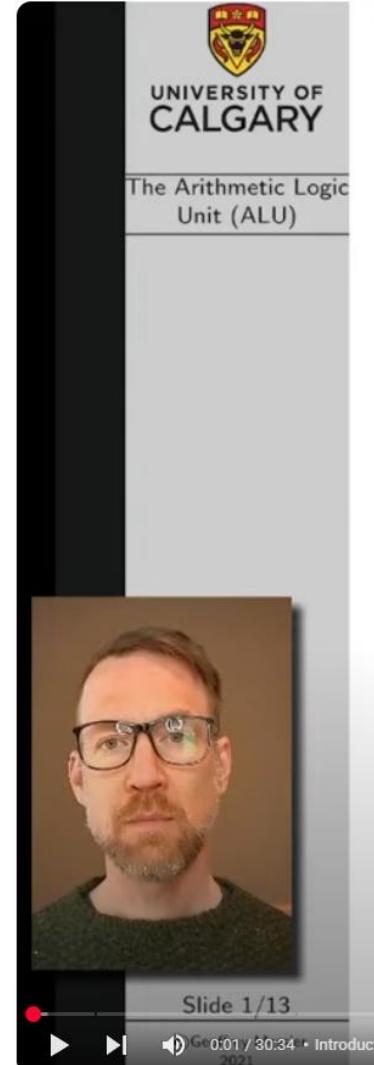
The diagram shows the internal operation of an ALU. Two 'Integer OPRAND' inputs, labeled 'A' and 'B', enter the ALU from the top. A 'STATUS' signal enters from the left. An 'OP CODE' signal enters from the bottom. The ALU processes these inputs to produce an 'Integer RESULT' at the bottom and a 'STATUS' signal at the right. The output 'RESULT' is also labeled with 'Y'.

What Is Arithmetic Logic Unit - ALU ?

www.learncomputerscienceonline.com

0:02 / 11:26

What Is Arithmetic Logic Unit ? | ALU In Computer Architecture Explained (2022)
 Что такое арифметико-логическая единица измерения? | Объяснен ALU в области компьютерной архитектуры (2022)
https://www.youtube.com/watch?v=H_aoaQYgKT8

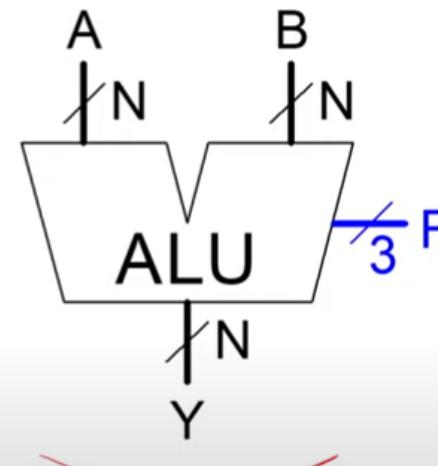


The Arithmetic Logic Unit (ALU)

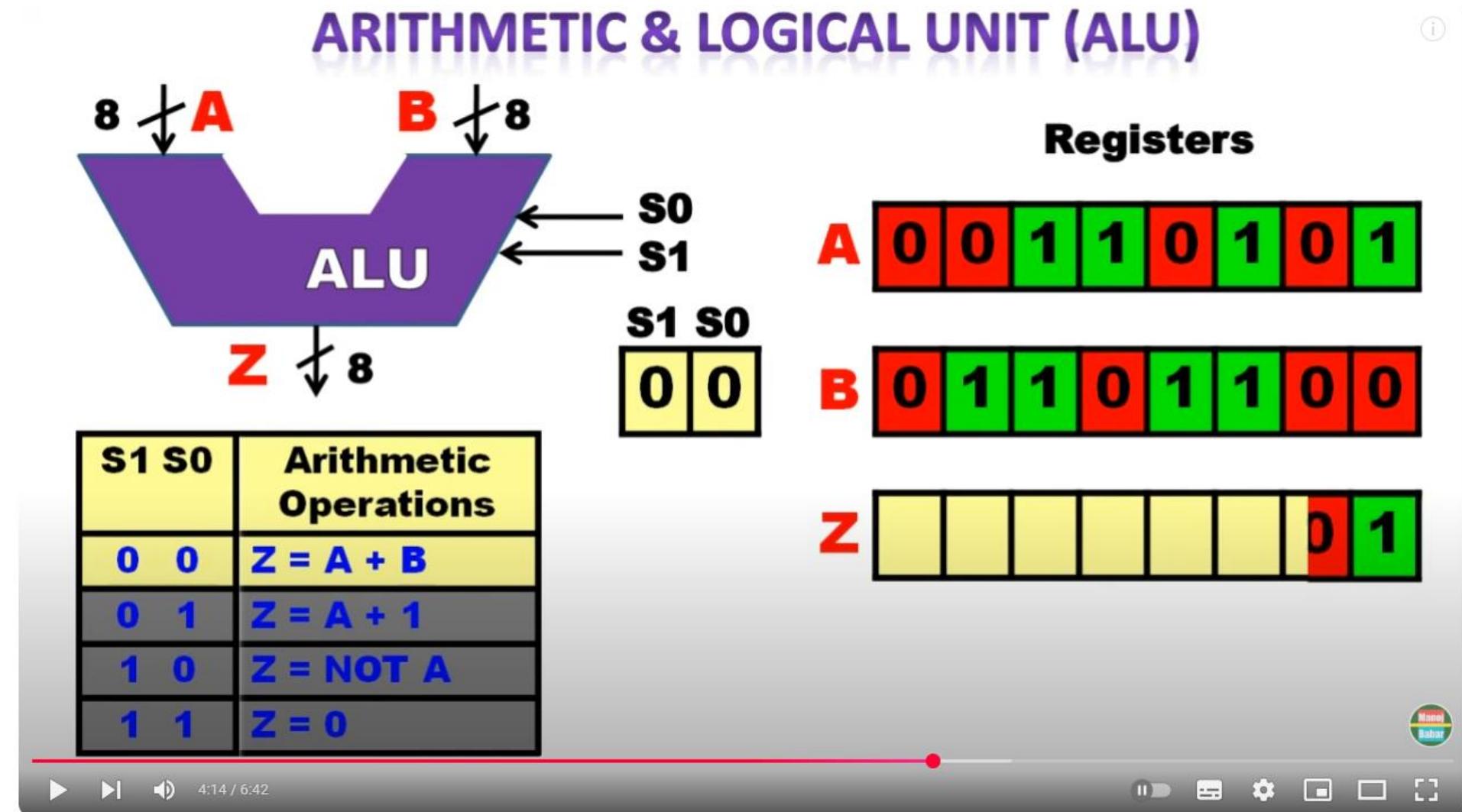
Geoffrey Messier
Professor of Electrical & Computer Engineering
Schulich School of Engineering

Computer Architecture Lecture 8: The Arithmetic Logic Unit (ALU) (2022)
Лекция 8 по компьютерной архитектуре: Арифметико-логический модуль (ALU) (2022)
<https://www.youtube.com/watch?v=VGHk1tU9xZ0>

Arithmetic Logic Unit (ALU)



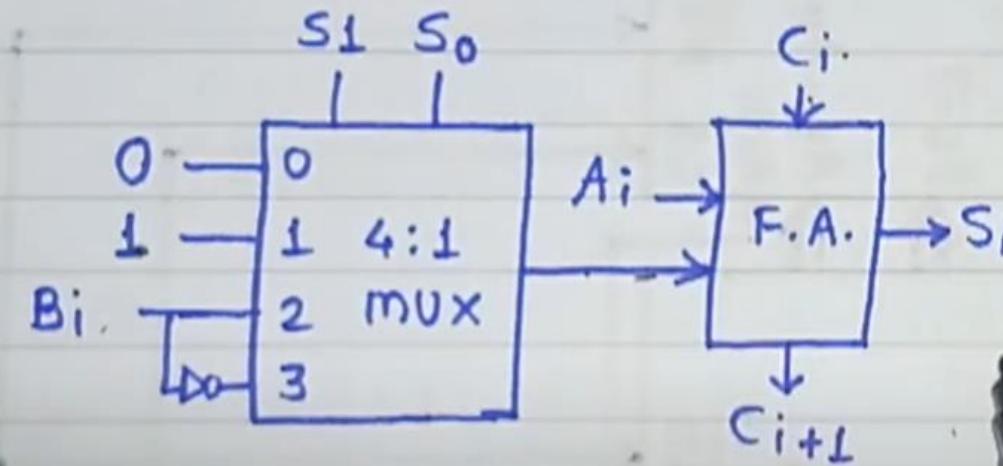
$F_{2:0}$	Function
000	A & B
001	A B
010	A + B
011	not used
100	A & \sim B
101	A \sim B
110	A - B
111	SLT



23.1) Arithmetic and Logic Unit (ALU) (2020)
 23.1) Арифметико-логический модуль (АЛУ) (2020)
<https://www.youtube.com/watch?v=dcsYdaO6txI>

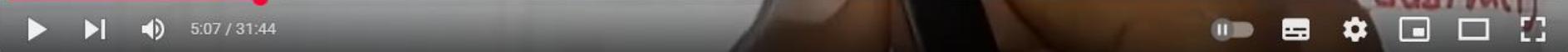
* Elementary ALU Design *

ALU = Arithmetic Unit + Logical Unit + Shift Unit



* Arithmetic Circuit *

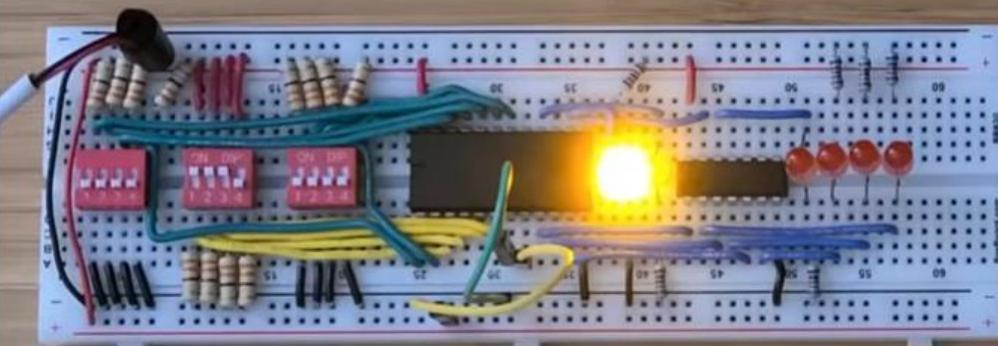
S_1	S_0	C_i	operation
0	0	0	A_i (Transfer)
0	0	1	$A_i + 1$ (Inc)
0	1	0	$A_i - 1$ (Dec)
0	1	1	A_i (Transfer)
1	0	0	$A + B$ (Add)
1	1	1	$A + B + 1$ (Add with carry) $A - B$ (Subtract with borrow) (Subtract)



23.2) Elementary ALU Design (2020)

23.2) Элементарное проектирование АЛУ (2020)

<https://www.youtube.com/watch?v=IPnYBgCmTCA>



FUNCTION TABLE

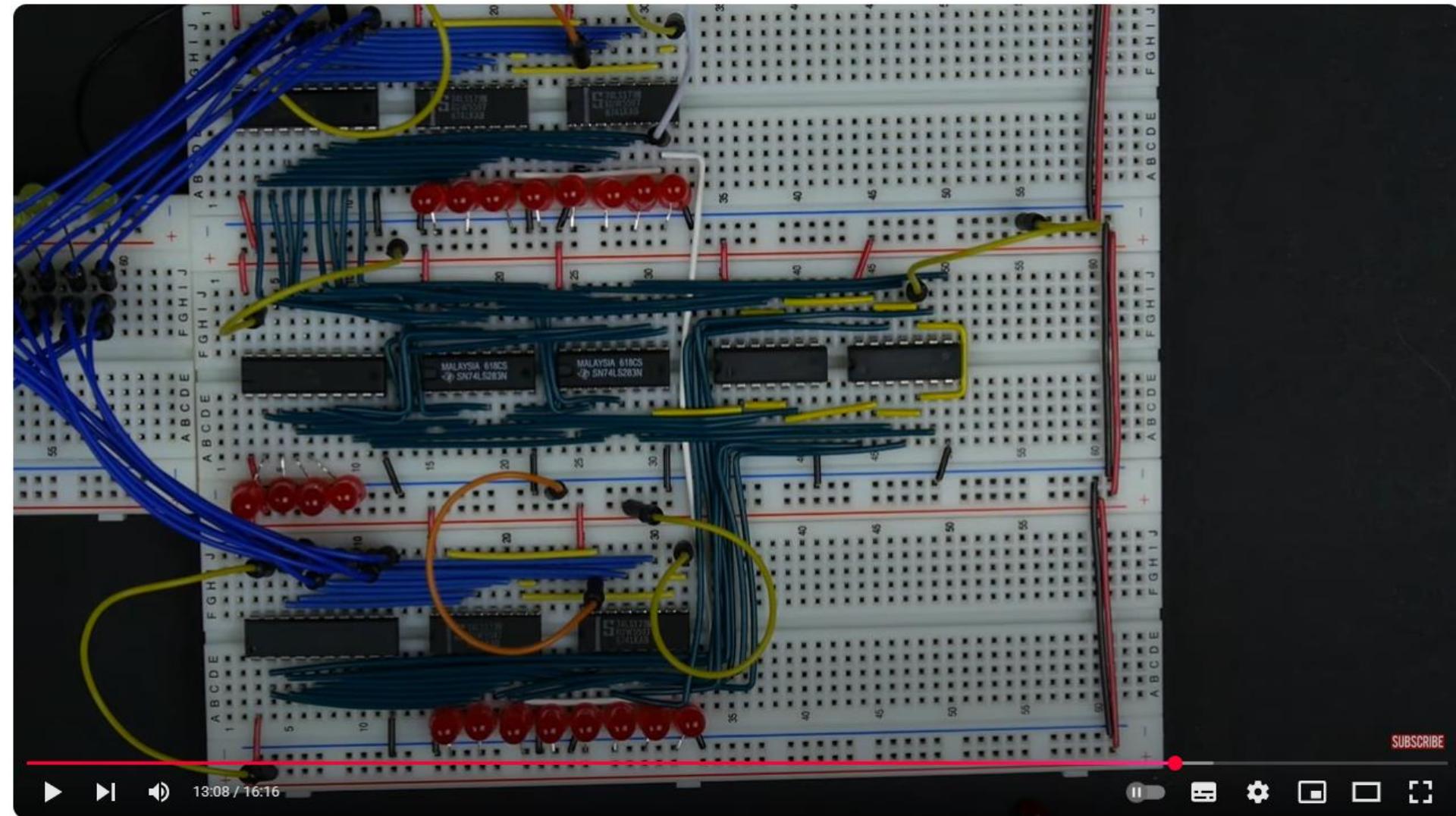
MODE SELECT INPUTS				ACTIVE LOW INPUTS & OUTPUTS		ACTIVE HIGH INPUTS & OUTPUTS	
S ₃	S ₂	S ₁	S ₀	LOGIC (M = H)	ARITHMETIC** (M = L) (C _n = L)	LOGIC (M = H)	ARITHMETIC** (M = L) (C _n = H)
L	L	L	L	A	A minus 1	A	A
L	L	L	H	AB	AB minus 1	A + B	A + B
L	L	H	L	A + B	AB minus 1	AB	A + B
L	L	H	H	Logical 1 minus 1	—	Logical 0 minus 1	—
L	H	L	L	A + B	A plus (A + B)	AB	A plus AB
L	H	L	H	B	AB plus (A + B)	(A + B)	(A + B) plus AB
L	H	H	L	A ⊕ B	A minus B minus 1	A ⊕ B	A minus B minus 1
L	H	H	H	A + B	A + B	AB	AB minus 1
H	L	L	L	AB	A plus (A + B)	A + B	A plus AB
H	L	L	H	A ⊕ B	A plus B	A ⊕ B	A plus B
H	L	H	L	B	AB plus (A + B)	B	(A + B) plus AB
H	L	H	H	A + B	A + B	AB	AB minus 1
H	H	L	L	Logical 0 A plus A*	—	Logical 1 A plus A*	—
H	H	L	H	AB	AB plus A	A + B	(A + B) plus A
H	H	H	L	AB	AB plus A	A + B	(A + B) Plus A
H	H	H	H	A	A	A	A minus 1

L = LOW Voltage Level
H = HIGH Voltage Level
*Each bit is shifted to the next more significant position
**Arithmetic operations expressed in 2s complement notation

▶ ▶ ⏴ 6:29 / 24:09 • Logic Operations >

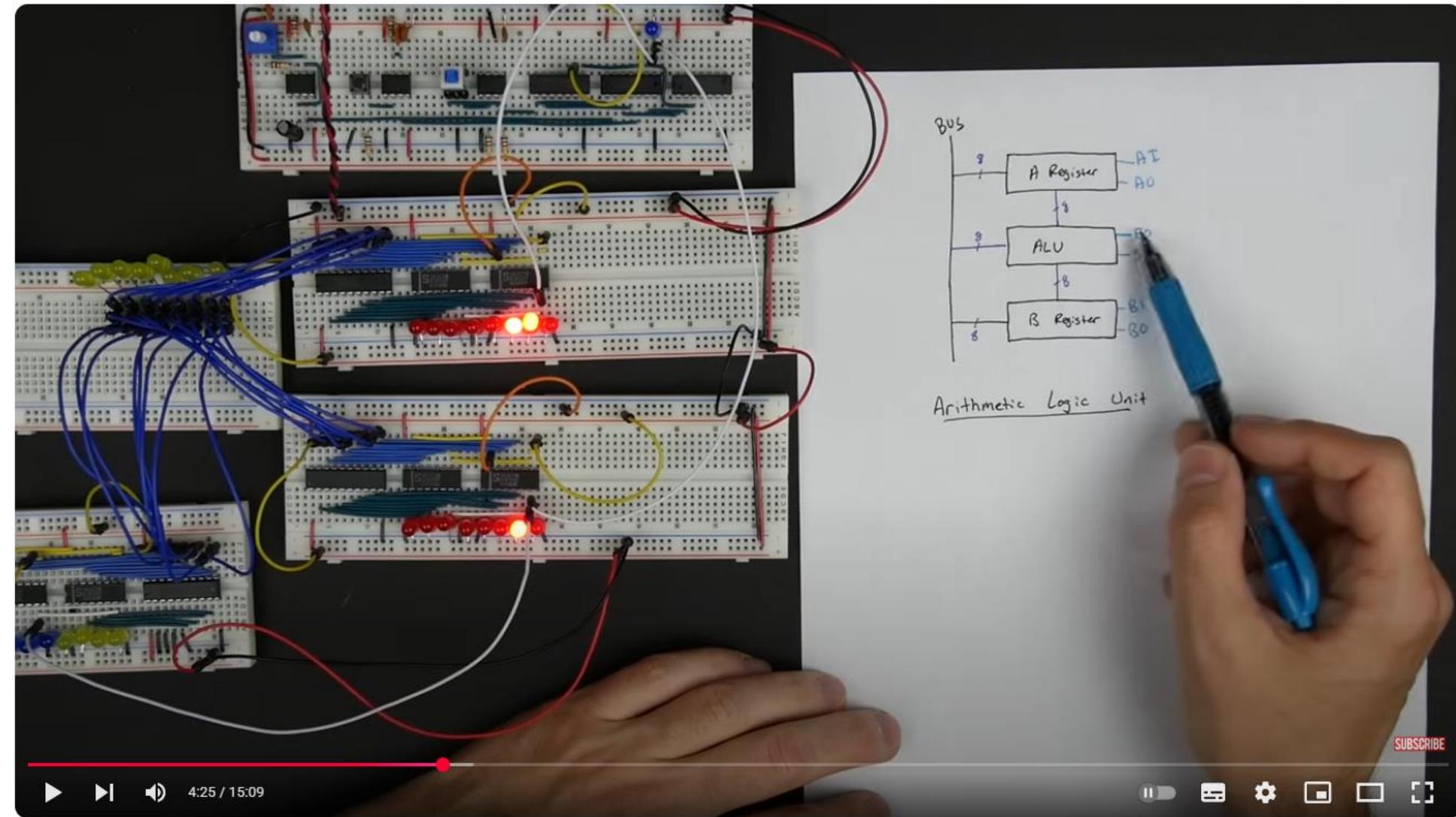
|| ⏷ ⏸ ⏹ ⏺ ⏻ ⏻ ⏻

The 74LS181 (4-bit Arithmetic Logic Unit) (2023)
74LS181 (4-разрядный арифметико-логический модуль) (2023)
<https://www.youtube.com/watch?v=rFILT31nVdw>

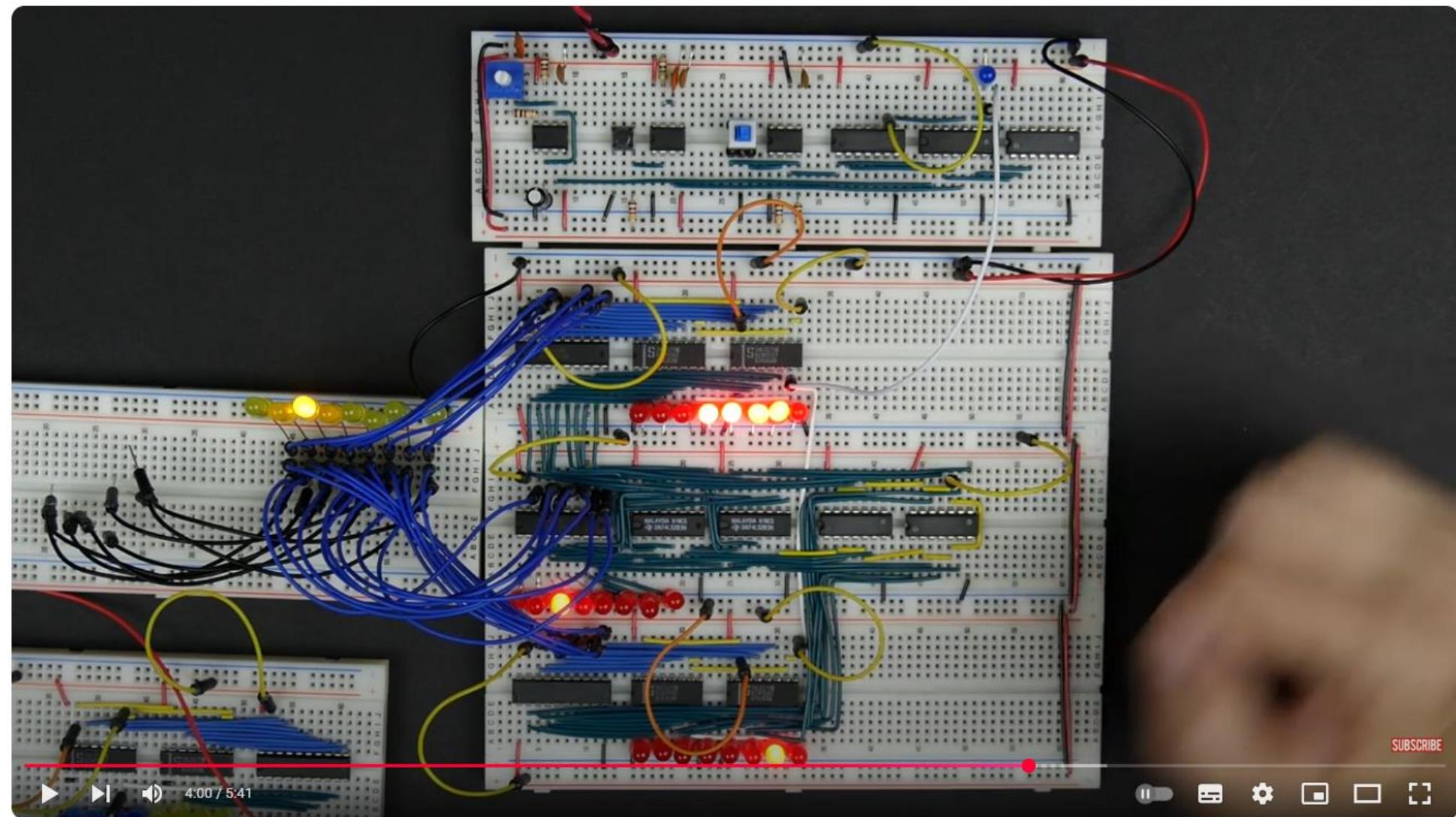


Building the ALU (2016)
Создание АЛУ (2016)

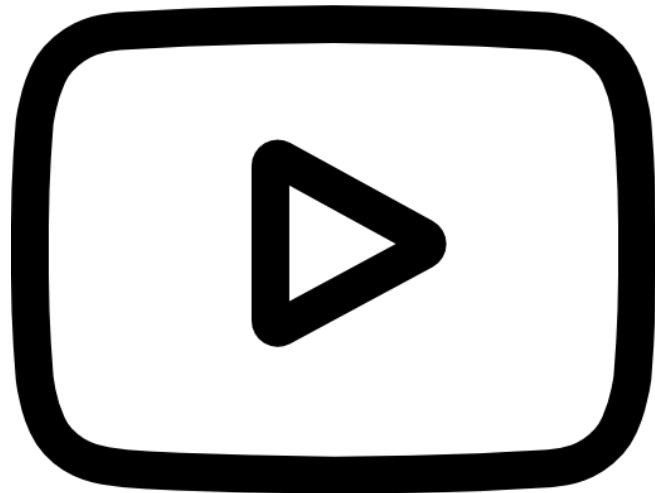
<https://www.youtube.com/watch?v=S-3fXU3FZQc>



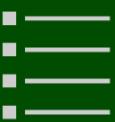
ALU Design (2016)
Проектирование АЛУ (2016)
<https://www.youtube.com/watch?v=mOVOS9AjqFs>



Testing the computer's ALU (2016)
Тестирование АЛУ компьютера (2016)
<https://www.youtube.com/watch?v=4nCMDvnR2Fg>



YouTube каналы и
плейлисты





Электротехника и электроника для программистов

@Zefar91 · 161 тыс. подписчиков · 430 видео

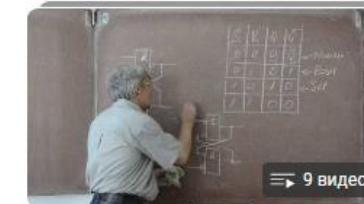
Рыбницкий филиал Приднестровского государственного университета им. Т.Г. Шевченко · ...ещё

abglazov.rfgu.ru

Вы подписаны

Главная Видео Плейлисты

триггер



Триггеры и счетчики

Электротехника и электроника для программистов · Плейлист

Лекция 97. RS-Триггер · 13:51

Лекция 99. Синхронный RS-триггер · 12:22

[ПОСМОТРЕТЬ ВЕСЬ ПЛЕЙЛИСТ](#)



Лекция 50. Триггер Шмитта на операционном усилителе

Электротехника и электроника для программистов · 40 тыс. просмотров · 12 лет назад

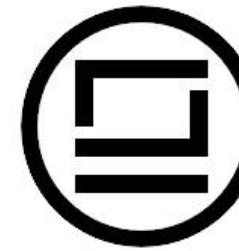
Очень часто возникает необходимость подсчета числа импульсов с плавными фронтами или при наличии помех....



Лекция 134. JK-Триггер

Электротехника и электроника для программистов · 70 тыс. просмотров · 11 лет назад

Электротехника и электроника для программистов
<https://www.youtube.com/@Zefar91/featured>



BitFlip

@BitFlipChannel · 5,43 тыс. подписчиков · 56 видео

Канал об электронике для начинающих. ...ещё

Вы подписаны

Главная

Видео

Плейлисты



Программируем компьютер и выполняем инструкции в ручном режиме

BitFlip · 4,9 тыс. просмотров · 3 года назад

В прошлом выпуске мы познакомились с инструкциями и разобрались, как компьютер должен их выполнять. В этот...

Для вас



Арифметико-логическое устройство



Регистры и АЛУ



Сумматор в одной микросхеме

Собираем 8-битное арифметико-логическое устройство (АЛУ)

10 тыс. просмотров · 4 года назад

Как получить сумму значений из двух регистров: соединяя регистры и арифметико-логическое...

3,9 тыс. просмотров · 4 года назад

Четырехбитный сумматор в одной микросхеме, стягивающие резисторы и микросхемы ТТЛ...

4 тыс. просмотров · 4 года назад

BitFlip - Канал об электронике для начинающих
<https://www.youtube.com/@BitFlipChannel>

Алексей Аббясов

@АлексейАббясов-д7y · 137 подписчиков · 12 видео

Подробнее о канале ...ещё

Подписаться

Главная Видео Плейлисты

Новые Популярные Старые

Синхронный T-триггер

Синхронный D-триггер

Синхронный статический RS-триггер

R-триггер И-НЕ/ИЛИ-НЕ

4.1. Синхронный статический T-триггер

3,5 тыс. просмотров · 3 года назад

3.1. Синхронный статический D-триггер

2,7 тыс. просмотров · 3 года назад

2.5. Синхронный статический RS-триггер

879 просмотров · 3 года назад

2.3. Асинхронный статический R-Триггер

239 просмотров · 3 года назад

2.4. Моделирование R-триггера

211 просмотров · 3 года назад

2.1. Асинхронный статический S-триггер

346 просмотров · 3 года назад

1.6. Построение временной диаграммы RS-триггера с учетом

750 просмотров · 3 года назад

1.5. Временная диаграмма RS-триггера в базисе ИЛИ-НЕ

1,1 тыс. просмотров · 3 года назад

Алексей Аббясов

<https://www.youtube.com/@АлексейАббясов-д7y/videos>



Инна Музылева

@ИннаМузылева · 44 подписчика · 60 видео

Подробнее о канале ...ещё

Подписаться

Главная

Видео

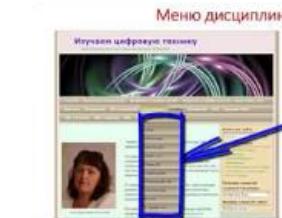
Плейлисты



Новые

Популярные

Старые



Музылева препод сайт

31 просмотр · 2 года назад



лог выраж и законы логики

136 просмотров · 2 года назад



Триггеры 1 классификация

388 просмотров · 2 года назад



Асинхр ИЛИ НЕ

396 просмотров · 2 года назад



Асинхр И НЕ

549 просмотров · 2 года назад



Триггеры 3 синхронные RS

201 просмотр · 2 года назад



Синхр RS триггеры с дин упр

536 просмотров · 2 года назад



Триггеры 5 D со стат упр

324 просмотра · 2 года назад

Инна Музылева

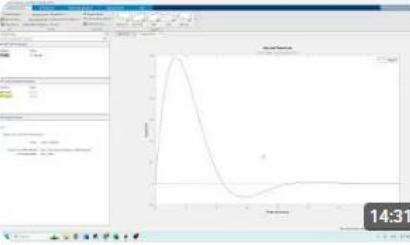
<https://www.youtube.com/@ИннаМузылева/videos>

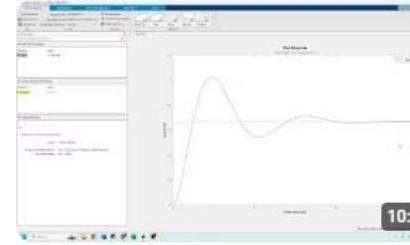
 Madiyar Nurgaliyev
@madiyarnurgaliyev6421 · 325 подписчиков · 98 видео
Подробнее о канале ...ещё

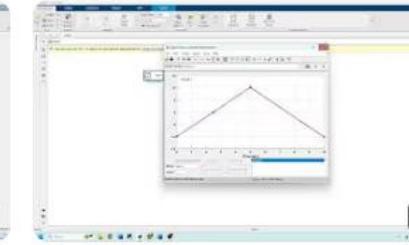
Подписаться

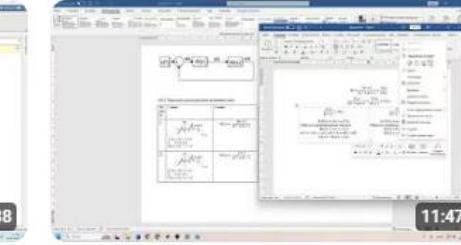
Главная Видео Плейлисты

Новые Популярные Старые

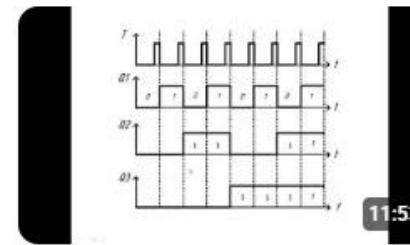
 14:31
Временные и частотные характеристики САУ в Simulink каз
198 просмотров · 5 месяцев назад

 10:42
Временные и частотные характеристики САУ в Simulink рус
117 просмотров · 5 месяцев назад

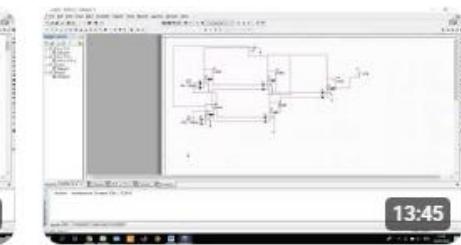
 8:38
Нелинейные САУ Нелинейные звенья
82 просмотра · 5 месяцев назад

 11:47
Нелинейные САУ. Фазовый портрет в Simulink
366 просмотров · 1 год назад

 22:55
Лабораторная работа №12 "АЦП и ЦАП"
Лабораторная работа №11 "Счетчики"

 11:53
Лабораторная работа №10 "D-триггер. T-триггер"
Лабораторная работа №9 "ДТЛ и ТТЛ" часть 3

 8:48

 13:45

Madiyar Nurgaliyev

<https://www.youtube.com/@madiyarnurgaliyev6421/videos>



Нсанов М.А. Цифровая микроэлектроника



Мурат Нсанов

@МуратНсанов · 3 тыс. подписчиков · 107 видео

Вашему вниманию представляется цикл видеоуроков по предмету «Цифровая

...ещё

Подписаться

Главная

Видео

Плейлисты



Новые

Популярные

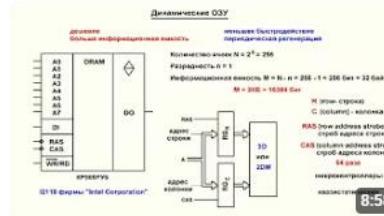
Старые



Видеоурок 4.53. Динамические ОЗУ.

Часть 2.

58 просмотров · 6 дней назад



Видеоурок 4.52. Динамические ОЗУ.

Часть 1.

64 просмотра · 7 дней назад



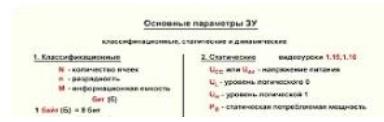
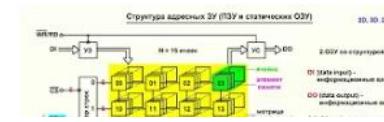
Видеоурок 4.51 Микросхемы постоянных ЗУ

178 просмотров · 10 дней назад



Видеоурок 4.50 Микросхемы адресных статических оперативных ЗУ

346 просмотров · 13 дней назад



Мурат Нсанов - Цикл видеоуроков по предмету «Цифровая микроэлектроника»
<https://www.youtube.com/@МуратНсанов/featured>



Дмитрий Гайворонский

@ДмитрийГайворонский-ю2г · 1,14 тыс. подписчиков · 121 видео

Подробнее о канале ...ещё

Подписаться

Главная

Видео

Плейлисты



Для вас

сигнала, входной сигнал воздействует на Q_0 , а далее входной сигнал каждого последующего триггера является выходным сигналом предыдущего.

$$Q_0^+ = Q_0 \oplus dH \quad Q_r^+ = Q_r \oplus dQ_{r-1}, r=1,2,\dots$$

41:59

Цифровая электроника Ответы на вопросы
(часть 2) 15 мая 2021

525 просмотров · 4 года назад

ЛУЧШИЕ	9 - 70	4,75 - 5,25
5592	-25 - 35	4,75 - 5,25

Особенности ТТЛ схем

1. существует входной ток высокого и низкого уровня $I_{H_L} = 2 \text{ мА}$
 $I_{H_H} = 20 \text{ нА}$
2. Выходные токи $I_{OL} = 8,20 \text{ мА}$ $I_{OH} = 0,2,0,4 \text{ мА}$
3. Уровни 0 и 1 $U_{OHL} = 3,5..4,0 \text{ В}$ $U_{OL} = 0,3 \text{ В}$
4. Ток потребления I_{cc} зависит от тока в нагрузке
5. Ток потребления I_{cc} зависит от t_{pd} . Чем выше t_{pd} , тем ниже I_{cc}

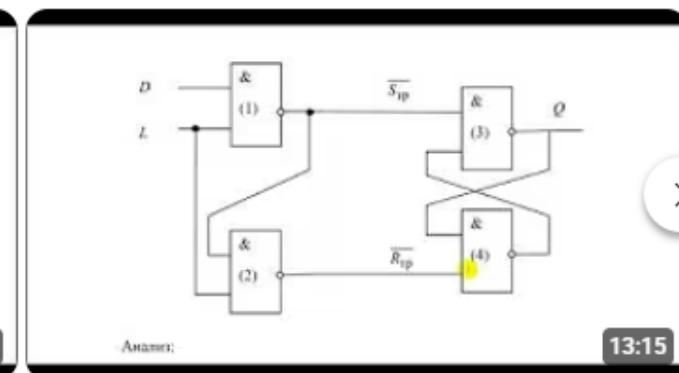
Качество серии определяется произведением t_{pd} и P (время задержки – мощность потребления)

Серии:

24:47

3 21 Стандартные интегральные схемы

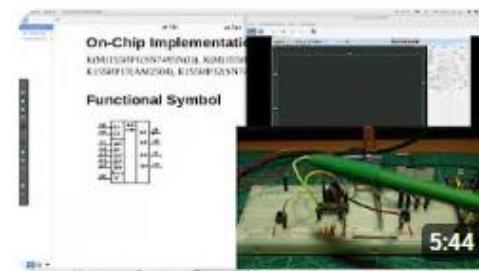
648 просмотров · 4 года назад



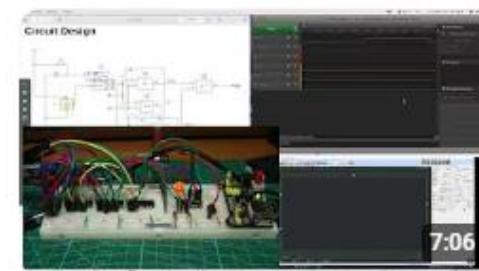
2 5 Асинхронный потенциальный D-L триггер на R-S триггере. Осциллограмма работы.

2,6 тыс. просмотров · 4 года назад

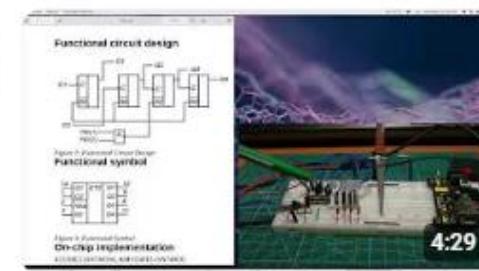
Дмитрий Гайворонский
<https://www.youtube.com/@ДмитрийГайворонский-ю2г>



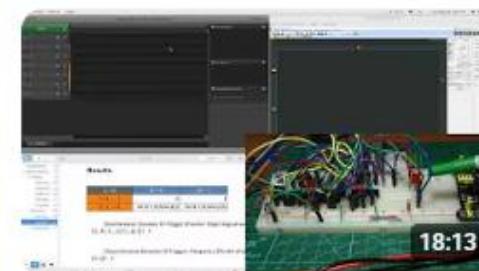
0111. Сдвиговый регистр



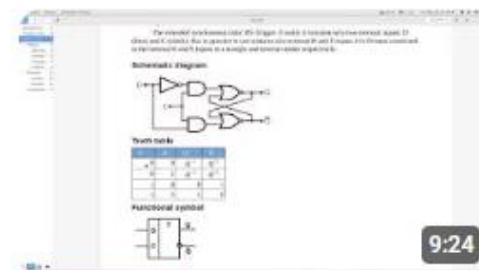
0105. Однобитный сумматор



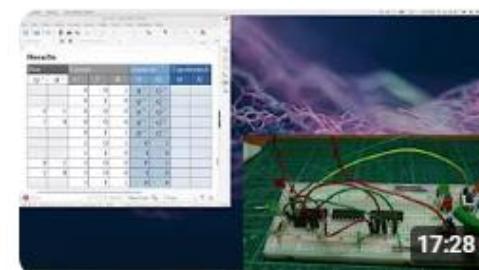
0106. Двоичные счётчики



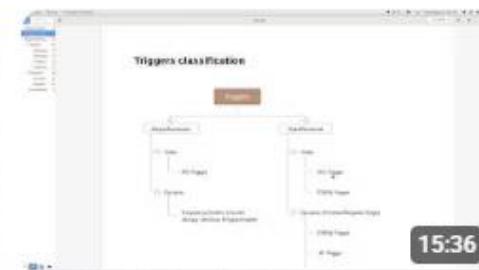
0100. D-Триггер / 0104. Схема деления частоты



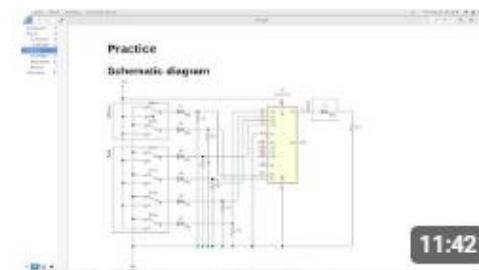
0100. D-Триггер



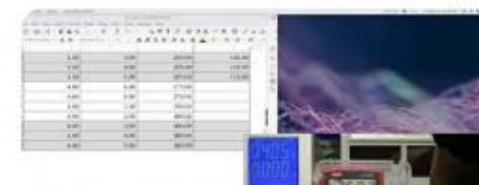
0099. Синхронный RS-Триггер



0097. RS-триггер



0091. Мультиплексор



Электротехника для программистов :: Практика

<https://www.youtube.com/@Электротехникадляпрограммистов/videos>

Электротехника

A Алексей Мак

Плейлист - 59 видео - 44 347 просмотров

▶ Воспроизвести в...

Схема заме
Схема замещения изображение мод элементов, замеш

Резист

- 36
Методом истинности таблица логической функции $f = x_1 \cdot x_2 + x_3 \cdot x_4 + x_5 \cdot x_6$
3:25
- 37
10:05
- 4 1 4 Составление логических выражений по таблице истинности
Алексей Мак • 7 лет назад • 4,3 тыс. просмотров
- 38
Составление таблиц истинности по логической функции
 $y = x_1 \cdot x_2 \cdot x_3 + x_4 \cdot x_5 \cdot x_6 + x_7 \cdot x_8 \cdot x_9$
3:02
- 4 1 5 Составление таблиц истинности по логической функции
Алексей Мак • 7 лет назад • 896 просмотров
- 39
Основные параметры импульсных сигналов
Периодичные
Непериодичные
4:19
- 4 1 6 Параметры импульсных сигналов
Алексей Мак • 7 лет назад • 7,6 тыс. просмотров
- 40
Способы электрического представления логических 0 и 1
Применяются для представления логических 0 и 1 в виде напряжения. Для записи информации на магнитную ленту, магнитную пленку, дискету, магнитную краску и т.д. используется логический элемент ИМК.
3:20
- 4 1 7 Электрические способы представления двоичной информации
Алексей Мак • 7 лет назад • 820 просмотров
- 41
Логические элементы
Алексей Мак • 7 лет назад • 1,4 тыс. просмотров

Плейлист - Электротехника

https://www.youtube.com/playlist?list=PLCKtVYi1kU04xx_6n2Uwl gj4KMG8NY1_X

Раздел 4.
Последовательностные цифровые устройства



Мурат Нсанов

Плейлист · 55 видео · 18 230 просмотров

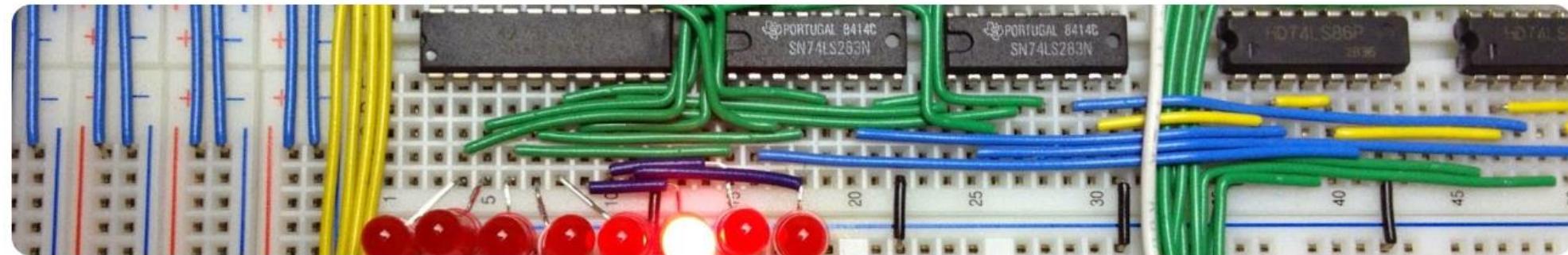
Поддержите автора:
<https://www.donationalerts.com/r/muratnsanov> ...ещё

▶ Воспроизвести в...

2 недоступных видео скрыто. 

 1  2  3  4	<p>Вideoурок 4.1. Введение в 4 раздел. Общие сведения об интегральных... Мурат Нсанов · 1 год назад · 1 тыс. просмотров</p> <p>Видеоурок 4.2. Входы интегральных триггеров. Мурат Нсанов · 1 год назад · 448 просмотров</p> <p>Видеоурок 4.3. Способы управления интегральными триггерами. Мурат Нсанов · 1 год назад · 412 просмотров</p> <p>Видеоурок 4.4. Асинхронный RS-триггер с прямыми статическими входами... Мурат Нсанов · 1 год назад · 617 просмотров</p>
---	---

Плейлист - Раздел 4. Последовательностные цифровые устройства / Мурат Нсанов
<https://www.youtube.com/playlist?list=PLN-wZHgzUk55cot8-IX6zs3XzNqUF4YfV>



Ben Eater •

@BenEater · 1,32 млн подписчиков · 126 видео

Subscribe to see tutorial-style videos about electronics, computer architecture, networking ...ещё

[patreon.com/beneater](https://www.patreon.com/beneater) и ещё 2 ссылки

[Подписаться](#)

Главная

Видео

Курсы

Плейлисты

Записи

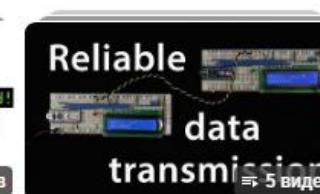


Все плейлисты

Упорядочить



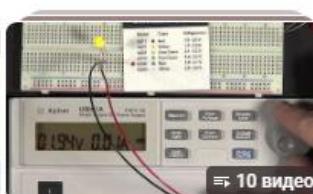
Build a 65c02-based computer from scratch
Весь курс



Error detection
Посмотреть весь плейлист



Building an 8-bit breadboard computer!
Весь курс



Digital electronics tutorial
Посмотреть весь плейлист



Networking tutorial
Весь курс

Ben Eater

<https://www.youtube.com/@BenEater/featured>



ЭВМ, периферийные устройства и контроллеры
Тема: Цифровая логика и базовые компоненты
ЭВМ

**Благодарю
за внимание**

КУТУЗОВ Виктор Владимирович

Белорусско-Российский университет, Кафедра «Программное обеспечение информационных технологий»
Республика Беларусь, Могилев, 2025

Список использованных источников

1. Рабочая программа дисциплины «ЭВМ, периферийные устройства и контроллеры» для студентов направлений подготовки 09.03.01 «Информатика и вычислительная техника» и 09.03.04 «Программная инженерия» / Кутузов В. В. – Могилев : Белорусско-Российский университет, 2025
2. Фотографии и картинки взяты с сайтов Яндекс.Картинки, Гугл.Картинки, Pinterest, иконки с flaticon.com
3. Отдельная информация генерировалась при помощи больших языковых моделей (LLM, Large Language Model). Прорабатывались идеи и структура отдельных разделов, уточнялась и перепроверялась отдельная информация, выполнялся поиск. Использовались: Qwen3-235B-A22B, Qwen3-235B-A22B-2507, Gemini-2.5-Flash-Lite-Preview, perplexity.ai. Вся сгенерированная информация многократно перепроверялась и дополнялась с интернет ресурсов.
4. Сервис перевода текстов DeepL Translate
<https://www.deepl.com/>
5. Сервис автоматического перевода текста от Яндекса встроенного в Яндекс.Браузер
<https://browser.yandex.ru>
6. Сара Л. Харрис, Дэвид Харрис Цифровая схемотехника и архитектура компьютера: RISC-V / пер. с англ. В. С. Яценкова, А. Ю. Романова; под ред. А. Ю. Романова. – М.: ДМК Пресс, 2021. – 810 с.: ил. ISBN 978-5-97060-961-3 <https://rutracker.org/forum/viewtopic.php?t=6204850>
7. Угрюмов, Е. П. Цифровая схемотехника: учеб, пособие для вузов. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2010. – 816 с. <https://djvu.online/file/l5iaM7nsDIFK9>
8. Шустов, М. А. Цифровая схемотехника. Практика применения. – СПб.: Наука и Техника, 2018. – 432 с. <https://djvu.online/file/psKYETGSqWW7w>

Список использованных источников

9. Забродин Ю.С. Промышленная электроника: Учебник для вузов. – М.: Альянс, 2008. – 496 с.
10. Лачин В.И., Савёлов Н.С. Электроника: Учебное пособие. – Ростов н/Д: Феникс, 2007. – 703 с.
11. Гусев В.Г., Гусев Ю.М. Электроника и микропроцессорная техника: Учеб. для вузов. – М.: Высшая школа, 2008. – 798 с.
12. Алексенко А.В., Шагуров И.И. Микросхемотехника. – М.: Радио и связь, 1990 (1982).
13. Потемкин И.С. Функциональные узлы цифровой автоматики. – М.: Энергоатомиздат, 1988.
14. Зельдин Е.А. Цифровые интегральные схемы в информационно-измерительной аппаратуре. – Л.: Энергоатомиздат, 1986.
15. Токхейм Р. Основы цифровой электроники. Пер. с англ. - М.: Мир, 1988. – 390 с.
16. Шустов, М. А. Цифровая схемотехника. Основы построения. – СПб.: Наука и Техника, 2018. – 320 с.
<https://djvu.online/file/jleHgaxy8AOxE>
17. Булатов, В. Н. Основы аналоговой и цифровой электроники. Цифровая электроника: учебное пособие / В. Н. Булатов; Оренбургский гос. ун-т, 2020. – 206 с.
http://elib.osu.ru/bitstream/123456789/13104/1/120804_20200528.pdf
18. Kani A. N. (Ed.) Digital Logic Circuits: With VHDL and Verilog Programs. – CBS, 2025. – 1410 р.
<https://www.twirpx.com/file/4364828/>
19. Введение в цифровые системы
https://portal.tpu.ru/SHARED/g/GREBENNIKOVVV/students/Tab4/Tab/011_Введение%20в%20цифровые%20системы.pdf

Список использованных источников

20. Электронные схемы 1 Лекция 1 Кафедра электронной инженерии Университета им. Ханджо Аджу
<https://slideplayer.com/slide/4983413/>
21. Аналоговые и Цифровые сигналы
<https://radioskot.ru/publ/teoria/analogovyj-i-czifrovoj-signal>
22. Цифровая схемотехника. Логические элементы
https://razumdom.ru/lectures/Digital_Logic.pdf
23. ФИЗИКА И СХЕМОТЕХНИКА ИНТЕГРАЛЬНЫХ СХЕМ. ТЕМА 4. СХЕМОТЕХНИКА ЦИФРОВЫХ ИНТЕГРАЛЬНЫХ СХЕМ (ЧАСТЬ 1) <https://edu2u.ru/sandbox/data/FSIC/L4.pdf>
24. Физика двоичной логики
<https://habr.com/ru/companies/timeweb/articles/653159/>
25. Набор логических микросхем 74HCxx и 74LSxx
https://aliexpress.ru/item/1005006441034028.html?sku_id=12000037176092542
26. Функциональная схемотехника/ Кустарев П.В. // Кафедра ВТ 2010-18, Университет ИТМО
https://picloud.pw/media/resources/posts/2018/03/15/lec_schemo_slide.pdf
27. Combinational and Sequential Circuits
<https://www.geeksforgeeks.org/digital-logic/combinational-and-sequential-circuits/>
28. Комбинационная логика
https://ru.wikipedia.org/wiki/Комбинационная_логика
29. Логический вентиль
https://ru.wikipedia.org/wiki/Логический_вентиль

Список использованных источников

30. Булева алгебра
<https://blog.skillfactory.ru/glossary/buleva-algebra/>
31. Босова Л. Л. Информатика. 10 класс : учебник / Л. Л. Босова, А. Ю. Босова. — М. : БИНОМ. Лаборатория знаний, 2016. — 288 с. : Глава 4 Элементы теории множеств и алгебры логики
<https://files.lbz.ru/authors/informatika/3/bosova-10-gl4.pdf>
32. Электроника. Синтез логических схем по заданной функции / В.В. Гребенников / Томский политехнический университет
https://portal.tpu.ru/SHARED/g/GREBENNIKOVVV/students/Tab4/Tab/06_Electronics_22_Design_Logic.pdf
33. Практикум по радиоэлектронике: цифровые схемы. Издание второе, переработанное и дополненное. Учебно-методическое пособие для студентов физического факультета МГУ и преподавателей практикума по радиоэлектронике. М.: Физический факультет МГУ им. М.В. Ломоносова, 2016. — 96 с
<http://photonics.phys.msu.ru/book-c.pdf>
34. Логические элементы
<https://radioskot.ru/publ/teoria/logicheskie-elementy>
35. Булатов, В.Н. Основы аналоговой и цифровой электроники. Цифровая электроника: учебное пособие / В.Н.Булатов; Оренбургский гос. ун-т, 2020. – 206 с.
http://elib.osu.ru/bitstream/123456789/13104/1/120804_20200528.pdf
36. Лекция Цифровые устройства
<https://portal.tpu.ru/SHARED/g/GFADDTPU/Ucheba/Ucheba/Лекции%20ЦУ%20-%20Ч.1%20Базовые%20логическ.pdf>

Список использованных источников

37. Лекция 8. Проектирование комбинационных логических микросхем
https://farabi.university/storage/files/21578641966759703851d0652117303_Лекция%208%20-%20Проектирование%20комбинационных%20логических%20микросхем%20+.pdf
38. Мультиплексоры и Демультиплексоры
<https://radioskot.ru/publ/teoria/multipleksory-i-demultipleksory>
39. Булатов, В.Н. Основы аналоговой и цифровой электроники. Цифровая электроника: учебное пособие / В.Н.Булатов; Оренбургский гос. ун-т, 2020. – 206 с.
http://elib.osu.ru/bitstream/123456789/13104/1/120804_20200528.pdf
40. Кобайло, А. С. Схемотехника цифровых вычислительных машин. Лабораторный практикум : учеб.-метод. пособие по учебной дисциплине «Арифметико-логические основы цифровых вычислительных машин и архитектура компьютеров» для студентов специальности 1-98 01 03 «Программное обеспечение информационной безопасности мобильных систем» / А. С. Кобайло, Н. А. Жиляк, И. А. Миронов. – Минск : БГТУ, 2018. – 97 с. https://elib.belstu.by/bitstream/123456789/26700/1/Kobajlo_Shemotehnika_2018.pdf
41. Основы логики и логические основы компьютера
<https://ppt-online.org/73297>
42. Дешифраторы и шифраторы
<https://radioskot.ru/publ/teoria/deshifratory-i-shifratory>
43. О работе ПК на примере Windows 10 и клавиатуры ч.2
<https://habr.com/ru/articles/489166/>
44. Цифровые электроизмерительные устройства
<https://ppt-online.org/687304>

Список использованных источников

45. Электроника. Комбинационные устройства

https://portal.tpu.ru/SHARED/g/GREBENNIKOVVV/students/Tab4/Tab/07_Electronics_22_Combin_Logic_2016.pdf

46. Сорока, Н. И. Кривинченко, Г. А. ТЕЛЕМЕХАНИКА Конспект лекций для студентов специальности 1-53 01 07 «Информационные технологии и управление в технических системах» всех форм обучения Часть 2 Коды и кодирование, Минск, 164 с. https://library.bsuir.by/m/12_100229_1_62250.pdf

47. Угрюмов Е. П. Цифровая схемотехника: учеб, пособие для вузов. — 3-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2010. — 816 с.

48. Сумматоры

<https://radioskot.ru/publ/teoria/summatory>

49. Легостаев Н. С. Микроэлектроника : учебное пособие / Н. С. Легостаев, К. В. Четвергов — Томск : Эль Контент, 2013. — 172 с. <https://edu.tusur.ru/publications/4280/download>

50. Угринович Н. Д. Информатика и ИКТ. Профильный уровень : учебник для 10 класса / Н. Д. Угринович. — 3-е изд., испр. — М. : БИНОМ. Лаборатория знаний, 2008. — 387 с. : ил.

51. Компараторы <https://radioskot.ru/publ/teoria/komparatory>

52. Физика и схемотехника интегральных схем. Тема 4. Схемотехника цифровых интегральных схем (Часть 1) <https://edu2u.ru/sandbox/data/FSIC/L4.pdf>

53. Электроника. Последовательностные логические устройства

https://stud.lms.tpu.ru/pluginfile.php/2701163/mod_resource/content/25/09_Основы%20ЦУ_ЛЕКЦИИ_Последоват-у-ва_2024.pdf

Список использованных источников

54. Кузин А. П. Учебно-методическое пособие по дисциплине «ОСНОВЫ МИКРОПРОЦЕССОРНОЙ ТЕХНИКИ» Ростов-на-Дону, 2019 – 99 с.
https://de.donstu.ru/CDOCourses/structure/_new_/39199/2971/5249.pdf
55. Триггеры
<https://radioskot.ru/publ/teoria/triggery>
56. Регистры
<https://radioskot.ru/publ/teoria/registry>
57. Счётчик (электроника)
[https://ru.wikipedia.org/wiki/Счётчик_\(электроника\)](https://ru.wikipedia.org/wiki/Счётчик_(электроника))
58. Кобайло, А. С. Схемотехника цифровых вычислительных машин. Лабораторный практикум : учеб.-метод. пособие по учебной дисциплине «Арифметико-логические основы цифровых вычислительных машин и архитектура компьютеров» для студентов специальности 1-98 01 03 «Программное обеспечение информационной безопасности мобильных систем» / А. С. Кобайло, Н. А. Жиляк, И. А. Миронов. – Минск : БГТУ, 2018. – 97 с. https://elib.belstu.by/bitstream/123456789/26700/1/Kobajlo_Shemotehnika_2018.pdf
59. Комбинационная логика и ее описание на языке Verilog
<https://ppt-online.org/989626>
60. Введение в проектирование на языке Verilog. Вводная лекция
https://docs.google.com/presentation/d/1TqIFGqrBzD166VZrZCWKF1KIriF4yPpPZ9JolRr6bsQ/edit?slide=id.g2a43841bf6b_0_125#slide=id.g2a43841bf6b_0_125
61. Введение в проектирование на языке Verilog
<https://viktor-prutyanov.github.io/drec-fpga-intro/>

Список использованных источников

62. Уровень регистровых передач
https://ru.wikipedia.org/wiki/Уровень_регистровых_передач
63. KLayout Documentation (Qt 4)
https://www.klayout.org/downloads/master/doc-qt4/manual/main_window.html
64. Inside a FPGA – Reduced and simplified
<http://jmk.dk/MMMI/PLDs/FPGA/fpga.htm>

Список использованных источников (YouTube)

Булева алгебра

1. Лекция №6 / Основы программирования / Булева алгебра и условия (2021)
<https://www.youtube.com/watch?v=EDEIPk2Y5Zc>
2. Логика - Булева алгебра и предикаты. Основы (2023)
<https://www.youtube.com/watch?v=dpHOT8dRS-A>
3. Информатика. Алгебра логики: Таблицы истинности (2014)
<https://www.youtube.com/watch?v=qrij6Ekwqr-c>
4. Алгебра логики: Законы алгебры логики (2014)
<https://www.youtube.com/watch?v=7XA77xNVBv4>
5. Информатика. Алгебра логики: Теория множеств (2014)
<https://www.youtube.com/watch?v=LVDZaTmaLWw>
6. Алгебра логики: Логические переменные и логические функции (2014)
<https://www.youtube.com/watch?v=Ys8b60jCZUQ>
7. Законы алгебры логики (2020)
<https://www.youtube.com/watch?v=sByAeOXgMPI>
8. Построение таблиц истинности (2020)
<https://www.youtube.com/watch?v=n944UR8ENz0>
9. Упрощение логических выражений. К.Ю.Поляков, 10 класс, параграф 18 (2023)
<https://www.youtube.com/watch?v=LgcPbKD5O0w>

Список использованных источников (YouTube)

Булева алгебра

10. Плейлист: Булева алгебра
<https://www.youtube.com/playlist?list=PLzlnMoWvt7wPYCiGGB9hV7sooSBcY9ytd>
11. Логика - Упрощение логических выражений. Законы алгебры логики (2018)
<https://www.youtube.com/watch?v=sNI5dB8l1qc>
12. Три способа упрощения логической функции (2014)
<https://www.youtube.com/watch?v=5U4P56A2ePY>
13. Минимизация логических функций. Карты Карно (2022)
<https://www.youtube.com/watch?v=YinX3OltpPE>
14. Диаграмма Карно. Минимизация булевых (логических) функций (2023)
<https://www.youtube.com/watch?v=dsNfRQ-aONU>
15. Карты Карно. Как они работают. Большой выпуск (2019)
<https://www.youtube.com/watch?v=wIEiX9ROSoE>
16. Лекция 80. Карта Карно (2013)
<https://www.youtube.com/watch?v=a37anDvo0bs>
17. Цифровые устройства и программируемые логические интегральные схемы. Лекция №5 (22.02.2022) (2022) https://www.youtube.com/watch?v=dw2bt-Qm_wq

Список использованных источников (YouTube)

Комбинационная логика

1. Verilog. Комбинационная логика (2019)
<https://www.youtube.com/watch?v=pMrJkHk10wU>
2. Комбинационная логика (2021)
<https://www.youtube.com/watch?v=DExYPdT2Bkk>
3. Комбинационная логика 2 (2021)
<https://www.youtube.com/watch?v=AAzkPNxO4l8>
4. Основы Цифровой Электроники - 4. Комбинационная схема (2025)
<https://www.youtube.com/watch?v=CVx91ZGp4tc>
5. Лекция Александра Силантьева “Комбинационная логика и её описание на языке Verilog” (ChipEXPO-2021) (2021) <https://www.youtube.com/watch?v=hkZT8-uXxpw>
6. Архитектура ЭВМ. Лекция 1: Типы архитектур. Комбинационная и последовательная логика (2021)
<https://www.youtube.com/watch?v=R-twnbnaueE>
7. Занятие 1 (2023-24): Введение в маршрут проектирования и упражнения с комбинационной логикой (2023) <https://www.youtube.com/watch?v=DFcvEO-gP0c>

Список использованных источников (YouTube)

Последовательностная логика

1. Последовательная логика (2021)
<https://www.youtube.com/watch?v=iB7cCzNEbAo>
2. ИВТ-23М. Л11. Последовательностная логика. Память (2020)
<https://www.youtube.com/watch?v=CZy5IDflgUU>
3. 4 4 11 Последовательностные устройства Триггеры (2017)
<https://www.youtube.com/watch?v=hgswGqUKKmU>

Список использованных источников (YouTube)

Сумматоры, Полусумматоры

1. Сумматор (2025)
<https://www.youtube.com/watch?v=sUApzhh3HBo>
2. Лекция 105. Однобитный сумматор (2013)
<https://www.youtube.com/watch?v=3b-slaOPfGI>
3. Лекция 295. Двоично-десятичный сумматор (2016)
<https://www.youtube.com/watch?v=ALNgFJzGfhU>
4. Полусумматоры, сумматоры. Назначение, классификация, принцип работы, типовые схемы. (2021)
<https://www.youtube.com/watch?v=e5TmYrVjZl0>
5. Префиксный сумматор: все о сумматорах по шагам (2023)
<https://www.youtube.com/watch?v=G1GnAhp5WHc>
6. Полусумматор и сумматор (2018)
<https://www.youtube.com/watch?v=adITv8WwMk4>
7. Сумматор на ОУ: Просто и Гениально! (2025)
<https://www.youtube.com/watch?v=WAQevRd02r4>
8. 04 Сумматор (2024)
<https://www.youtube.com/watch?v=1xs0zsgbsRY>
9. Компьютер на транзисторах своими руками №6 Сложение чисел. Сумматор (2020)
<https://www.youtube.com/watch?v=7IRN33i016M>

Список использованных источников (YouTube)

Сумматоры, Полусумматоры

10. Основы цифровой схемотехники. 08. Сумматоры (2019)
https://www.youtube.com/watch?v=_ISGmCyzAN4
11. РК6. Схемотехника. Сумматоры (2020)
<https://www.youtube.com/watch?v=kHPzByBmn98>
12. Полусумматор, Сумматор, Многоразрядный сумматор (2024)
<https://www.youtube.com/watch?v=Ik2un9jTmOo>
13. Видеоурок 3.34. Двоично-десятичные сумматоры. Часть 1 (2023)
<https://www.youtube.com/watch?v=fsliyGX-b7s>
14. Видеоурок 3.35. Двоично-десятичные сумматоры. Часть 2 (2023)
<https://www.youtube.com/watch?v=wqGq1Ht9SV0>
15. Видеоурок 3.30. Двоичные сумматоры. Часть 1 (2023)
<https://www.youtube.com/watch?v=cawnk7w8b84>
16. Видеоурок 3.31. Двоичные сумматоры. Часть 2 (2023)
<https://www.youtube.com/watch?v=vWcj6fvXdnA>
17. Видеоурок 3.32. Двоичные сумматоры. Часть 3 (2023)
<https://www.youtube.com/watch?v=fxWw1ZpQoQ>
18. Видеоурок 3.33. Двоичные сумматоры. Часть 4 (2023)
<https://www.youtube.com/watch?v=Zq81TKZt8xU>

Список использованных источников (YouTube)

Сумматоры, Полусумматоры

19. 0105. Однобитный сумматор (2019)
<https://www.youtube.com/watch?v=vynyNsA1b0Y>
20. BitFlip. Полусумматор - складываем двоичные цифры (2020)
<https://www.youtube.com/watch?v=tbqtt7cvjxM>
21. BitFlip. Сумматор (2020)
https://www.youtube.com/watch?v=5pK5m0d_RNk
22. BitFlip. Сумматор с функцией вычитания (2020)
https://www.youtube.com/watch?v=K2_RWCCpsS8
23. BitFlip. Четырехбитный сумматор (2020)
<https://www.youtube.com/watch?v=ONEMWuStDI8>
24. BitFlip. Четырехбитный сумматор в одной микросхеме, стягивающие резисторы и микросхемы ТТЛ логики (2020) https://www.youtube.com/watch?v=l4eomM5v_2I
25. Полный Однобитный Сумматор (2024)
<https://www.youtube.com/watch?v=fqYqF0Trkro>
26. Полусумматор на npn транзисторах (2024)
<https://www.youtube.com/watch?v=2vXxgUxvpoM>
27. Сумматор на логических элементах (2024)
<https://www.youtube.com/watch?v=1vfeMJhGQJU>

Список использованных источников (YouTube)

Сумматоры, Полусумматоры

28. Лабораторная работа №4. Сумматоры (2021)
https://www.youtube.com/watch?v=uOWPh8_YKvQ
29. Half Adders and Full Adders Beginner's Tutorial (2020)
Полусумматоры и сумматоры для начинающих (2020)
<https://www.youtube.com/watch?v=ecn-8iGDRSo>
30. Learn how computers add numbers and build a 4 bit adder circuit (2015)
Узнайте, как компьютеры складывают числа и создают схему 4-разрядного сумматора (2015)
<https://www.youtube.com/watch?v=wvJc9CZcvBc>
31. Full Adder (2022)
Полный сумматор (2022)
https://www.youtube.com/watch?v=mZ_2406nCrE
32. Combinational Devices 1: Half Adder and Full Adder (2020)
Комбинационные устройства 1: Половинный сумматор и полный сумматор (2020)
<https://www.youtube.com/watch?v=dNDo4wAJBio>
33. 8-Bit Adder built from 152 Transistors (2021)
8-разрядный сумматор, построенный на 152 транзисторах (2021)
<https://www.youtube.com/watch?v=X3lB1pVowlo>
34. Half Adder and Full Adder Explained | The Full Adder using Half Adder (2022)
Объяснены половинный и полный сумматоры | Полный сумматор с использованием половинного сумматора (2022) <https://www.youtube.com/watch?v=5XbRIVWFRIw>

Список использованных источников (YouTube)

Сумматоры, Полусумматоры

35. Half Adder (2014)
Полу сумматор (2014)
<https://www.youtube.com/watch?v=aLUY-s7LSns>
36. Full Adder (2014)
Сумматор (2014)
<https://www.youtube.com/watch?v=RK3P9L2ZXk4>
37. Full Adder using Half Adder(2015)
Полный сумматор с использованием полусумматора (2015)
https://www.youtube.com/watch?v=Z_DYRgtAXfw
38. Binary Addition, Half-Adder, Full-Adder Logic Gate Circuits CLEARLY EXPLAINED (2018)
Двоичное сложение, полусумматор, сумматор. Логические схемы с понятным объяснением (2018)
https://www.youtube.com/watch?v=hExfiB_UFzA
39. 4-Bit Adder built from logic gates (2021)
4-битный сумматор, построенный на логических вентилях (2021)
<https://www.youtube.com/watch?v=QDiNPL6aGp0>

Список использованных источников (YouTube)

Мультиплексоры, Демультиплексоры

1. Лекция 91. Мультиплексор (2013)
<https://www.youtube.com/watch?v=LX5osXWp5il>
2. Лекция 94. Применение мультиплексора (2013)
<https://www.youtube.com/watch?v=Wj171iJQGYs>
3. Мультиплексоры принцип действия (2012)
<https://www.youtube.com/watch?v=jd2U831n190>
4. Что такое мультиплексор и как он работает (2022)
<https://www.youtube.com/watch?v=RhhO7jedMi8>
5. Видеоурок 3.22. Мультиплексоры. Часть 1 (2023)
<https://www.youtube.com/watch?v=tgJ9nYP55c8>
6. Видеоурок 3.23. Мультиплексоры. Часть 2 (2023)
<https://www.youtube.com/watch?v=jKy6o1gAk0l>
7. Видеоурок 3.24. Мультиплексоры. Часть 3 (2023)
<https://www.youtube.com/watch?v=bN1C1kSIPGk>
8. Видеоурок 3.25. Мультиплексоры. Часть 4 (2023)
<https://www.youtube.com/watch?v=bUpFV3TJXjA>
9. Видеоурок 3.26. Структура мультиплексоров (2023)
<https://www.youtube.com/watch?v=BMNQku6S4Dc>

Список использованных источников (YouTube)

Мультиплексоры, Демультиплексоры

10. Мультиплексоры принцип работы, наращивание (2022)
<https://www.youtube.com/watch?v=rJaJsRQleUs>
11. 0091. Мультиплексор (2019)
<https://www.youtube.com/watch?v=EodHGKQAWBg>
12. Verilog. Мультиплексор. Декодер. Семисегментный индикатор (2019)
<https://www.youtube.com/watch?v=aLgDQIL-e2Y>
13. Лекция 88. Демультиплексор (2013)
<https://www.youtube.com/watch?v=FBiRItM8uFY>
14. Демультиплексоры, принцип действия (2014)
<https://www.youtube.com/watch?v=tzPfUaqZcgs>
15. Демультиплексор, простейшая схема (2020)
<https://www.youtube.com/watch?v=jlaG1QmMqmQ>
16. Видеоурок 3.27. Демультиплексоры. Часть 1 (2023)
https://www.youtube.com/watch?v=2y_cAgTZPH0
17. Видеоурок 3.28. Демультиплексоры. Часть 2 (2023)
<https://www.youtube.com/watch?v=YXxCfzU9uyo>
18. Видеоурок 3.29. Структура демультиплексоров (2023)
<https://www.youtube.com/watch?v=Rv5kCzSzmOA>

Список использованных источников (YouTube)

Мультиплексоры, Демультиплексоры

19. 4 3 2 Мультиплексоры и демультиплексоры (2017)
https://www.youtube.com/watch?v=x_sM-Fkg4h4
20. Основы цифровой схемотехники. 07. Мультиплексоры и демультиплексоры (2019)
<https://www.youtube.com/watch?v=XfQnEpJOjLU>
21. РК6. Схемотехника. Мультиплексоры (2020)
<https://www.youtube.com/watch?v=40FwfpZdnha>
22. Мультиплексоры и демультиплексоры (2022)
<https://www.youtube.com/watch?v=-4XdhGPx5MY>
23. Мультиплексор и демультиплексор (2025)
<https://www.youtube.com/watch?v=3gOL8EqMs78>
24. Видеоурок на тему: "Мультиплексор и демультиплексор" (2020)
<https://www.youtube.com/watch?v=7o5tKpvVS2Q>
25. Раздел 4 30 Мультиплексоры и демультиплексоры (2020)
<https://www.youtube.com/watch?v=IDUVVjxZntg>
26. Мультиплексоры и демультиплексоры часть 2 (2020)
https://www.youtube.com/watch?v=uaY9ij_yScQ
27. Мультиплексоры и демультиплексоры часть 3 (2020)
<https://www.youtube.com/watch?v=hEAUtvailvQ>

Список использованных источников (YouTube)

Мультиплексоры, Демультиплексоры

28. Лабораторная работа № 3. Мультиплексоры, демультиплексоры и компараторы кодов (2021)
<https://www.youtube.com/watch?v=-iltmeKmqq0>
29. Multiplexers and DeMultiplexers (2017)
Мультиплексоры и демультиплексоры (2017)
<https://www.youtube.com/watch?v=HleQhZ9Gq5s>
30. Introduction to Multiplexers | MUX Basic (2014)
Введение в мультиплексоры | MUX Basic (2014)
<https://www.youtube.com/watch?v=FKvnmxte98A>
31. 4X1 Multiplexer (2014)
Мультиплексор 4X1 (2014)
<https://www.youtube.com/watch?v=g1Lfz1XgrH8>
32. 8X1 Multiplexer (2014)
Мультиплексор 8X1 (2014)
<https://www.youtube.com/watch?v=b0z7YKKCCyY>
33. Multiplexer Tutorial | Basic Theory With Practical Example (2021)
Учебное пособие по мультиплексорам | Базовая теория с практическим примером (2021)
<https://www.youtube.com/watch?v=qUa29m-c-5o>
34. 1. Introduction to Multiplexers in Hindi | MUX Basics | 2 X 1 MUX Design (2019)
1. Введение в мультиплексоры на хинди | Основы MUX | Проектирование 2 X 1 MUX (2019)
<https://www.youtube.com/watch?v=2aZBMNj8fns>

Список использованных источников (YouTube)

Мультиплексоры, Демультиплексоры

35. 11. Introduction to Demultiplexers in Hindi | Tech Gurukul by Dinesh Arya (2019)
11. Введение в демультиплексоры на хинди | Tech Gurukul Динеша Арии (2019)
<https://www.youtube.com/watch?v=FarRWppOF-E>
36. Multiplexers & Signal Switches (2020)
Мультиплексоры и коммутаторы сигналов (2020)
<https://www.youtube.com/watch?v=l2K8oQyS5z8>
37. What are switches & multiplexers? (2019)
Что такое коммутаторы и мультиплексоры? (2019)
<https://www.youtube.com/watch?v=n9EkIrCrgH8>
38. Are switches & multiplexers bidirectional? (2019)
Являются ли коммутаторы и мультиплексоры двунаправленными? (2019)
<https://www.youtube.com/watch?v=z9GRiYPq7LM>
39. 2. 4x1 Multiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
2. Мультиплексор 4x1 на хинди | Tech Gurukul от Динеша Арии (2019)
<https://www.youtube.com/watch?v=06VyGspXpMo>
40. 3. 8X1 Multiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
3. Мультиплексор 8X1 на хинди | Tech Gurukul от Динеша Арии (2019)
https://www.youtube.com/watch?v=IZRR-_2TmxU

Список использованных источников (YouTube)

Мультиплексоры, Демультиплексоры

41. 4. 4X1 MUX using 2X1 MUX in Hindi | Multiplexer Tree | Tech Gurukul by Dinesh Arya(2019)
4. Мультиплексор 4X1 с использованием мультиплексора 2X1 на хинди | Дерево мультиплексоров | Tech Gurukul от Динеша Арыи(2019)
<https://www.youtube.com/watch?v=78UHqvhDaSc>
42. 5. 8x1 MUX using 2x1 MUX in Hindi | Multiplexer Tree | Tech Gurukul by Dinesh Arya (2019)
5. Мультиплексор 8x1 с использованием мультиплексора 2x1 на хинди | Дерево мультиплексоров | Tech Gurukul от Динеша Арыи (2019)
<https://www.youtube.com/watch?v=QaL75vZt8-Q>
43. 6. 8x1 MUX using 4x1 MUX in Hindi | Multiplexer Tree | Tech Gurukul by Dinesh Arya (2019)
6. Мультиплексор 8x1 с использованием мультиплексора 4x1 на хинди | Дерево мультиплексоров | Tech Gurukul от Динеша Арыи (2019)
https://www.youtube.com/watch?v=x4h7OAJq_N4
44. 7. 32X1 MUX using 8X1 MUX in Hindi | Very Important | Tech Gurukul By Dinesh Arya (2019)
7. МУЛЬТИПЛЕКСОР 32X1 с использованием мультиплексора 8X1 на хинди | Очень важно | Tech Gurukul от Динеша Арыи (2019)
https://www.youtube.com/watch?v=iRR8_f1SIw
45. 8. Implementation of Boolean Function using Multiplexers in Hindi - 1 | Very Easy (2019)
8. Реализация логической функции с использованием мультиплексоров на хинди - 1 | Очень просто (2019)
<https://www.youtube.com/watch?v=tStPccdLLkk>

Список использованных источников (YouTube)

Мультиплексоры, Демультиплексоры

46. 9. Implementation of Boolean Function using Multiplexers in Hindi - 2 | Very Easy (2019)
9. Реализация логической функции с использованием мультиплексоров на хинди - 2 | Очень просто (2019) https://www.youtube.com/watch?v=_V_Md8-dSj8
47. 12. 1:4 Demultiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
12. Демультиплексор в формате 1:4 на хинди | Tech Gurukul от Динеша Ари (2019) <https://www.youtube.com/watch?v=y-JOsmHVcKA>
48. 13. 1:8 Demultiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
13. Демультиплексор в формате 1:8 на хинди | Tech Gurukul от Динеша Ари (2019) <https://www.youtube.com/watch?v=epBoqrhAPq8>
49. 14. Implement Logic Expression or Truth Table using Demultiplexer in Hindi | Very Important (2019)
14. Реализуйте логическое выражение или таблицу истинности с помощью демультиплексора на хинди | Очень важно (2019) <https://www.youtube.com/watch?v=Sum3ZTULe3U>
50. 15. Full Subtractor using 1:8 Demultiplexer in Hindi | Tech Gurukul by Dinesh Arya (2019)
15. Полное вычитание с использованием демультиплексора 1:8 на хинди | Tech Gurukul от Динеша Ари (2019) <https://www.youtube.com/watch?v=hDtv9MkL7IU>
51. 16 to 1 mux using 4 to 1 mux | 16 X 1 MULTIPLEXER USING 4 TO 1 MULTIPLEXER (2021)
16 к 1 мультиплексору с использованием 4 к 1 мультиплексора | 16 X 1 МУЛЬТИПЛЕКСОР С ИСПОЛЬЗОВАНИЕМ 4 К 1 МУЛЬТИПЛЕКСОРА (2021) <https://www.youtube.com/watch?v=3IUHu4AYqWk>

Список использованных источников (YouTube)

Мультиплексоры, Демультиплексоры

52. 16X1 Multiplexer || 16 to 1 Multiplexer || Truth Table of 16X1 Multiplexer | Circuit of 16X1 MUX (2022)
Мультиплексор 16X1 || Таблица истинности мультиплексора 16X1 | Схема мультиплексора 16X1 (2022)
<https://www.youtube.com/watch?v=9BnLejOS4YA>
53. 4X1 MUX using 2X1 MUX || 4*1 Multiplexer using 2*1 Multiplexer | 4x1 mux using 2x1 mux | DLD || STLD (2022)
4X1 мультиплексор с использованием 2X1 мультиплексора | DLD || STLD (2022)
<https://www.youtube.com/watch?v=OYpdHCjP1cE>
54. 4 to 1 Multiplexer (MUX) Block diagram, Truth Table, Logical expression and Diagram (2023)
Структурная схема мультиплексора (MUX) 4 к 1, таблица истинности, логическое выражение и диаграмма (2023)
<https://www.youtube.com/watch?v=u7LVduhiY18>
55. 8 to 1 Multiplexer (MUX) Block diagram, Truth Table, Logical expression and Diagram (2023)
Структурная схема мультиплексора (MUX) от 8 до 1, таблица истинности, логическое выражение и диаграмма (2023)
<https://www.youtube.com/watch?v=0-BsImNhPVs>
56. MUX/DEMUX (2025)
<https://www.youtube.com/watch?v=wTTDfsMVxsQ>
57. Implementation of boolean function using multiplexers | Hindi | One question with three types of mux (2020)
Реализация логической функции с использованием мультиплексоров | Хинди | Один вопрос с тремя типами мультиплексоров (2020)
<https://www.youtube.com/watch?v=gTFx9FNIA2w>

Список использованных источников (YouTube)

Шифраторы, Дешифраторы

1. Шифраторы, дешифраторы. Назначение, принцип работы, типовые схемы (2021)
<https://www.youtube.com/watch?v=vsX4xMseF14>
2. 4 3 1 Комбинационные устройства Дешифраторы и шифраторы (2017)
<https://www.youtube.com/watch?v=kR4m-BT3UwY>
3. Основы цифровой схемотехники. 06. Шифраторы и дешифраторы (2019)
<https://www.youtube.com/watch?v=szsxdj3gbnc>
4. Шифраторы и дешифраторы (2021)
<https://www.youtube.com/watch?v=7WLaRHY108Y>
5. Дешифраторы и шифраторы (2017)
<https://www.youtube.com/watch?v=pBnAN5pjCDg>
6. 2021.03.02 Лекция № 3 Дешифратор, Шифратор, Мультиплексор (2021)
<https://www.youtube.com/watch?v=0tbwCZl9a0s>
7. 0087. Шифратор (2019)
https://www.youtube.com/watch?v=JoG_dpqyScs
8. Лекция 87. Шифратор (2013)
<https://www.youtube.com/watch?v=mDhHgKF835E>
9. Лекция 85. Дешифратор (2013)
<https://www.youtube.com/watch?v=1Qr-ENKLg1A>

Список использованных источников (YouTube)

Шифраторы, Дешифраторы

10. ДЕШИФРАТОР из СВЕТОДИОДОВ для семисегментного индикатора (2020)
<https://www.youtube.com/watch?v=-VoBka9gTmo>
11. Шифраторы и дешифраторы (2022)
<https://www.youtube.com/watch?v=iEReAZ1BJYA>
12. РК6. Схемотехника. Дешифраторы (2020)
<https://www.youtube.com/watch?v=ynofPq-3FEI>
13. Introduction to Encoders and Decoders (2015)
Введение в Шифраторы и Дешифраторы (2015)
<https://www.youtube.com/watch?v=feBvhLFQEDk>
14. Encoder in Digital Electronics | Working, Application and Logic Circuit of Encoder(2022)
Кодировщик в цифровой электронике | Принцип работы, применение и логическая схема Шифратора (2022) <https://www.youtube.com/watch?v=NWiPVMDh7GE>
15. Decoder Explained | What is Decoder? Applications of Decoder | 5 to 32 Decoder using 3 to 8 Decoders (2022)
Декодер: объяснение | Что такое Дешифратор? Применение дешифратора | Дешифратор 5–32 с использованием 3–8 дешифратора (2022)
<https://www.youtube.com/watch?v=a3wh7xV5PBU>
16. Digital Logic – Decoders (2013)
Цифровая логика – Дешифраторы (2013)
<https://www.youtube.com/watch?v=1prCy3kdy1U>

Список использованных источников (YouTube)

Шифраторы, Дешифраторы

17. Digital Logic - Encoders (2013)
Цифровая логика - Шифраторы (2013)
<https://www.youtube.com/watch?v=s6V4cHFsQBE>
18. Decoders/Encoders (2020)
Дешифраторы /Шифраторы (2020)
<https://www.youtube.com/watch?v=7IzEw-LUQiM>
19. How To Use Line Decoders/Encoders(2016)
Как использовать линейные Дешифраторы/Шифраторы (2016)
<https://www.youtube.com/watch?v=DXIDgVKGg5c>
20. Encoder and Decoder Combinational Circuits | 4 x 2 Encoder & 2 x 4 Decoder with Truth Table (2018)
Комбинированные дешифраторы и шифраторы | 4 x 2 шифратор и 2 x 4 дешифратор с таблицей истинности (2018)
<https://www.youtube.com/watch?v=rEwJZAPqmGs>

Список использованных источников (YouTube)

Преобразователи кодов

1. Видеоурок 3.18. Структура преобразователей кодов. Часть 1 (2023)
<https://www.youtube.com/watch?v=DeT-Y-ommIM>
2. Видеоурок 3.19. Структура преобразователей кодов. Часть 2 (2023)
<https://www.youtube.com/watch?v=gQfoQ8BOF60>
3. Видеоурок 3.20. Структура преобразователей кодов. Часть 3 (2023)
<https://www.youtube.com/watch?v=nTzc-C2GUow>
4. Видеоурок 3.21. Структура преобразователей кодов. Часть 4 (2023)
<https://www.youtube.com/watch?v=Vg7fEPPragI>
5. Информатика, 9 практическая работа - Преобразователь кодов(2023)
<https://www.youtube.com/watch?v=PvQXcFluhHA>
6. Q. 4.8:Design a code converter that converts a decimal digit from the 8, 4, -2, -1 code to BCD (2020)
Вопрос 4.8: Разработайте преобразователь кодов, который преобразует десятичную цифру из кода 8, 4, -2, -1 в BCD (2020)
<https://www.youtube.com/watch?v=yqK9XSBRqU>

Список использованных источников (YouTube)

Компараторы

1. Основы цифровой схемотехники. 10. Компараторы (2019)
<https://www.youtube.com/watch?v=xQ-HUUEpng>
2. Лекция 18 Операционный усилитель,. Компаратор. (2012)
<https://www.youtube.com/watch?v=Gw5D-pLF54w>
3. лекция 371 компаратор в Atmega8 (2018)
<https://www.youtube.com/watch?v=p3fhNmWGJbc>
4. 4 3 4 Цифровые компараторы (2017)
<https://www.youtube.com/watch?v=tDRodUFaLao>
5. КОМПАРАТОР в электронике. Самое понятное объяснение! (2021)
<https://www.youtube.com/watch?v=bwNZcf9K5IE>
6. КОМПАРАТОР что за МИКРОСХЕМА и зачем он нужен И чем его просто заменить (2021)
<https://www.youtube.com/watch?v=a-XB6GZFfpA>
7. Компаратор (2018)
<https://www.youtube.com/watch?v=7-2fe7tHAMQ>
8. Чудо электротехники - Компаратор (2022)
<https://www.youtube.com/watch?v=xorxp8MaFew>
9. Компаратор на операционном усилителе (2017)
<https://www.youtube.com/watch?v=Gj2IxPJ12iw>

Список использованных источников (YouTube)

Компараторы

10. 0018. Операционный усилитель. Компаратор. (2018)
<https://www.youtube.com/watch?v=MLd5r6RFe7o>
11. Компараторы. Часть 1 - введение (2015)
<https://www.youtube.com/watch?v=Nof9YFDaaJw>
12. Компараторы. Часть 2 - фонарик, реагирующий на свет (2015)
<https://www.youtube.com/watch?v=nKljycWq2Xs>
13. Компараторы. Часть 3 - логические элементы на компараторах (2015)
<https://www.youtube.com/watch?v=juuRsZPKrpw>
14. Компараторы. Часть 4 - Ночник на логике (2015)
<https://www.youtube.com/watch?v=8Fea9LZiMAg>
15. Компараторы. Часть 5 - Генератор прямоугольных импульсов (2016)
<https://www.youtube.com/watch?v=UlpkRp8G3Dk>
16. Компараторы. Часть 6 - ШИМ генератор из одного компаратора (2016)
<https://www.youtube.com/watch?v=D972Rxh50C4>
17. Компараторы. Часть 7 - Генератор треугольных импульсов (2016)
<https://www.youtube.com/watch?v=0x-or3q8bBU>
18. Компараторы. Часть 8 - ШИМ генератор, управляемый напряжением (2016)
<https://www.youtube.com/watch?v=RaGtsLdYbNM>

Список использованных источников (YouTube)

Компараторы

19. Компараторы. Часть 9 - Световой датчик движения (2016)
<https://www.youtube.com/watch?v=ndNhPk-a80s>
20. Включение света в темноте или как работает компаратор (2015)
<https://www.youtube.com/watch?v=qNskSCWQRzE>
21. Видеоурок 3.37. Цифровые компараторы. Часть 1 (2023)
<https://www.youtube.com/watch?v=LUDm2UlwBJ0>
22. Видеоурок 3.38. Цифровые компараторы. Часть 2 (2023)
<https://www.youtube.com/watch?v=iuUMv1z8isE>
23. Comparator Explained (Inverting Comparator, Non-Inverting Comparator and Window Comparator) (2018)
Описание компаратора (Инвертирующий компаратор, неинвертирующий компаратор и оконный компаратор) (2018) <https://www.youtube.com/watch?v=k9zQjEaKtfk>
24. Comparator - Operational Amplifier | Basic Circuits #16 | Electronics Tutorials (2022)
Компаратор — операционный усилитель | Базовые схемы №16 | Учебники по электронике (2022)
<https://www.youtube.com/watch?v=zNAbcUSptWE>
25. An Overview on Comparators (2022) / Обзор компараторов (2022)
<https://www.youtube.com/watch?v=ОНК9Нм3KMсA>
26. Introduction to comparator functions (2019)
Введение в функции компаратора (2019)
<https://www.youtube.com/watch?v=JmHyloQYMVE>

Список использованных источников (YouTube)

Триггеры

1. Последовательные логические устройства. Триггеры (RS, D, JK, T). Принцип работы, типовые схемы (2021) <https://www.youtube.com/watch?v=7QLQplw5EKE>
2. Лекция 101. Триггеры в терминах сигналов (2013) <https://www.youtube.com/watch?v=w7qLb1AcfZQ>
3. Что такое триггер в электронике и не только (2020) <https://www.youtube.com/watch?v=RKgfcAxeQwU>
4. Учимся хранить информацию: триггеры на макетной плате | Схемотехника для начинающих №8 (2024) <https://www.youtube.com/watch?v=jrz5Ssvzebs>
5. Триггеры. 1. Классификация (2023) https://www.youtube.com/watch?v=tOsdWWH_kmo
6. Урок №22. Триггер. (2016) <https://www.youtube.com/watch?v=5Tce5yRX0pQ>
7. Урок №51. Триггер осциллографа (2019) <https://www.youtube.com/watch?v=uPGDtRs2PQc>
8. [электроника для начинающих] Триггер на транзисторах! Принцип работы и сборка. (2018) <https://www.youtube.com/watch?v=raWjVYSIg50>
9. Простейший триггер. Видео №1 к 6 главе книги. (2021) <https://www.youtube.com/watch?v=0us6cCk-mrg>

Список использованных источников (YouTube)

Триггеры

10. Видеоурок 4.3. Способы управления интегральными триггерами. (2024)
<https://www.youtube.com/watch?v=iZB7g0UdwCE>
11. Информатика - Построение триггеров (2023)
<https://www.youtube.com/watch?v=U2b5EAcFtFc>
12. Компьютер на транзисторах своими руками №12 T,D,JK - триггеры (2020)
<https://www.youtube.com/watch?v=i6y3iFP5KIO>
13. Лабораторная работа № 5. Триггеры (2021)
<https://www.youtube.com/watch?v=Sr6sYGo7nJc>

Список использованных источников (YouTube)

Триггеры - RS-триггер

1. Лекция 97. RS-Триггер (2013)
<https://www.youtube.com/watch?v=g1PHEXU5HeY>
2. Лекция 99. Синхронный RS-триггер (2013)
<https://www.youtube.com/watch?v=vNvYWVht6pM>
3. Логическая реализация ячейки памяти. RS триггер (2018)
<https://www.youtube.com/watch?v=C9hF9YmSGUg>
4. Информатика, 10 практическая работа - Триггеры (2023)
<https://www.youtube.com/watch?v=kvpCofdRov0>
5. RS-триггер (2020)
<https://www.youtube.com/watch?v=SX2dVGgCWLk>
6. РК6. Схемотехника. Двухступенчатый RS-триггер с запрещающим инвертором (2020)
https://www.youtube.com/watch?v=eqSA7Ja_Mck
7. 1.1. Асинхронный статический RS-триггер в базисе ИЛИ-НЕ (2022)
<https://www.youtube.com/watch?v=cyrdkJlOrB0>
8. 1.2. Моделирование RS-триггера в Multisim (2022)
<https://www.youtube.com/watch?v=9SsWioqmp8Y>
9. 1.3. Асинхронный статический RS-триггер в базисе И-НЕ (2022)
https://www.youtube.com/watch?v=aVWAJy_0gFs

Список использованных источников (YouTube)

Триггеры - RS-триггер

10. 1.4. Моделирование RS-триггера в Multisim (2022)
<https://www.youtube.com/watch?v=IfEOA8w4HJA>
11. 1.5. Временная диаграмма RS-триггера в базисе ИЛИ-НЕ (2022)
<https://www.youtube.com/watch?v=qoCyZ4ygK-Y>
12. 1.6. Построение временной диаграммы RS-триггера с учетом задержек (2022)
https://www.youtube.com/watch?v=gA_x2hTK9-4
13. 2.5. Синхронный статический RS-триггер (2022)
<https://www.youtube.com/watch?v=O75Jjl2wnmE>
14. Асинхронный RS-триггер на элементах И-НЕ (2023)
<https://www.youtube.com/watch?v=1cYj8ltxgesc>
15. Асинхронный RS-триггер в базисе ИЛИ-НЕ (2023)
<https://www.youtube.com/watch?v=IGzvWcWgHTM>
16. Синхронные RS-триггеры со статическим управлением на основе асинхронных в базисе И-НЕ и в базисе ИЛИ-НЕ (2023) <https://www.youtube.com/watch?v=t9sAoVSu3MY>
17. Синхронные RS-триггеры с динамическим управление (2023)
<https://www.youtube.com/watch?v=VTUDAO-8Opc>
18. Триггеры. Двухтактные RS-триггеры (2023)
<https://www.youtube.com/watch?v=Ag0B2vY4MRY>

Список использованных источников (YouTube)

Триггеры - RS-триггер

19. Исследование работы RS-триггера на элементах И-НЕ в Multisim 14.1 (2021)
<https://www.youtube.com/watch?v=E3H6zO6U7ya>
20. RS-Trigger делаем: просто, сложно, мощно и ещё проще и веселей ! (2022)
<https://www.youtube.com/watch?v=Y9-wJEd5R5k>
21. RS-триггер. Видео №2 к 6 главе книги. (2021)
<https://www.youtube.com/watch?v=Aefa66TOY78>
22. 0099. Синхронный RS-Триггер (2019)
<https://www.youtube.com/watch?v=SRIA6tOz9n0>
23. 0097. RS-триггер (2019)
<https://www.youtube.com/watch?v=qJ52QGwdVIs>
24. РК6. Схемотехника. Синхронный RS-триггер (2020)
<https://www.youtube.com/watch?v=AqcPlfQSgiw>
25. Видеоурок 4.4. Асинхронный RS-триггер с прямыми статическими входами. Часть 1 (2024)
https://www.youtube.com/watch?v=S3sVe_Ogk8w
26. Видеоурок 4.5. Асинхронный RS-триггер с прямыми статическими входами. Часть 2 (2024)
<https://www.youtube.com/watch?v=uRLMJ2DmACK>
27. Видеоурок 4.6. Асинхронный RS-триггер с инверсными статическими входами (2024)
<https://www.youtube.com/watch?v=OB-YG8ykWB8>

Список использованных источников (YouTube)

Триггеры - RS-триггер

28. Видеоурок 4.2. Входы интегральных триггеров. (2024)
<https://www.youtube.com/watch?v=rbeYtYQ6wxo>
29. 5. SR Flip Flop in Hindi | Tech Gurukul by Dinesh Arya (2018)
5. SR Триггер на хинди | Tech Gurukul от Динеша Ары (2018)
<https://www.youtube.com/watch?v=UIRPUa0fvco>
30. SR Latch, Gated SR Latch, and Data Latch (2023)
Sr-защелка, закрытая SR-защелка и защелка для передачи данных (2023)
<https://www.youtube.com/watch?v=NBihtSRjaZA>

R-триггер

28. 2.3. Асинхронный статический R-Триггер (2022)
<https://www.youtube.com/watch?v=nexJwj4mzLk>
29. 2.4. Моделирование R-триггера (2022)
<https://www.youtube.com/watch?v=pUXRbxnVys8>

Список использованных источников (YouTube)

Триггеры - D-триггер

1. Лекция 100. D-триггер (2013)
<https://www.youtube.com/watch?v=tKZkAx9Q3Po>
2. D-Триггер, подробный обзор и тестирование! [Электроника] Для начинающих! Не DIY KIT, Микросхема (2018) <https://www.youtube.com/watch?v=VX6Rz2VVxP4>
3. РК6. Схемотехника. Синхронный D-триггер (2020)
https://www.youtube.com/watch?v=MAgzsUO-z_l
4. 0100. D-Триггер / 0104. Схема деления частоты (2019)
<https://www.youtube.com/watch?v=dthinytMTFk>
5. РК6. Схемотехника. D-триггер с динамическим управлением записью (2020)
https://www.youtube.com/watch?v=OJK_WxAxmCA
6. РК6. Схемотехника. Двухступенчатый D-триггер с запрещающими связями (2020)
<https://www.youtube.com/watch?v=EUzLoQM67SI>
7. Видеоурок 4.10. D-триггеры. Часть 1. (2024)
<https://www.youtube.com/watch?v=ffRJ428Yw2o>
8. Видеоурок 4.11. D-триггеры. Часть 2. (2024)
<https://www.youtube.com/watch?v=KXZ4CWaXhLs>
9. D-триггер: устранием состояния гонки в RS-триггере (2020)
https://www.youtube.com/watch?v=_A8-oLOkza0

Список использованных источников (YouTube)

Триггеры - D-триггер

10. D-триггер с динамическим управлением (2020)
<https://www.youtube.com/watch?v=8SeIYLY5fkl>
11. 3.1. Синхронный статический D-триггер (2022)
<https://www.youtube.com/watch?v=1sGKOfNeFR4>
12. D-триггер на транзисторном уровне. Видео №4 к 6 главе книги.(2021)
<https://www.youtube.com/watch?v=kPP9ghnKgZE>
13. Лабораторная работа №10 "D-триггер. Т-триггер" (2020)
<https://www.youtube.com/watch?v=Xpdm9ehJbmk>
14. Исследование работы D-триггера на элементах И-НЕ в Multisim 14.1 (2021)
<https://www.youtube.com/watch?v=SgPYI5LrTk8>
15. 0100. D-Триггер (2019)
<https://www.youtube.com/watch?v=rWD2j9wtGmY>
16. Вся правда о RS-триггере, D-триггер, практика применения RS-триггера (2019)
<https://www.youtube.com/watch?v=Pi4PUSHxGC4>
17. 9. D Flip Flop | Sequential Circuits | Tech Gurukul by Dinesh Arya (2018)
9. D-триггер | Последовательные схемы | Tech Gurukul от Динеша Арыи (2018)
https://www.youtube.com/watch?v=_SwesRcW79s
18. Introduction to D flip flop (2015) / Введение в D-триггеры (2015)
<https://www.youtube.com/watch?v=dnfXXpW7tlw>

Список использованных источников (YouTube)

Триггеры - D-триггер

19. D flip-flop (2016)
D триггер (2016)
https://www.youtube.com/watch?v=YW-_GkUguMM

Список использованных источников (YouTube)

Триггеры - Т-триггер

1. РК6. Схемотехника. Асинхронный Т-триггер (2020)
<https://www.youtube.com/watch?v=fsEDUgDMYJE>
2. 4.1. Синхронный статический Т-триггер (2022)
<https://www.youtube.com/watch?v=nuiDQ29UiY>
3. Исследование работы Т-триггера в Multisim 14.1 (2021)
<https://www.youtube.com/watch?v=kTYQRYdZFrE>
4. Видеоурок 4.12. Т-триггеры. (2024)
<https://www.youtube.com/watch?v=lr3HOA5b2aw>
5. 4.1. Синхронный статический Т-триггер (2022)
<https://www.youtube.com/watch?v=nuiDQ29UiY>
6. 15. T Flip Flop | Sequential Circuits | Tech Gurukul by Dinesh Arya (2019)
15. Т Триггер | Последовательные схемы | Tech Gurukul от Динеша Ары (2019)
https://www.youtube.com/watch?v=1wld_qMB1Yw
7. T Flip-Flop Explained | Circuit Diagram, Excitation Table and Characteristic Equation (2022)
Описание Т-триггера | Принципиальная схема, таблица возбуждения и характеристическое уравнение (2022)
<https://www.youtube.com/watch?v=hw2rWOlg9x4>

Список использованных источников (YouTube)

Триггеры - JK-триггер

1. Лекция 134. JK-Триггер (2014)
<https://www.youtube.com/watch?v=8ETkClVC0fg>
2. РК6. Схемотехника. Двухступенчатый JK-триггер с запрещающими связями (2020)
<https://www.youtube.com/watch?v=oHzAUWCXed4>
3. JK-триггер. Видео №3 к 6 главе книги. Ошибочные и рабочие схемы. (2021)
<https://www.youtube.com/watch?v=jsu5xLEc-uU>
4. Видеоурок 4.7. JK-триггеры. Часть 1. (2024)
https://www.youtube.com/watch?v=_xRYo4a3MeM
5. Видеоурок 4.8. JK-триггеры. Часть 2. (2024)
<https://www.youtube.com/watch?v=x8RzoohUZBc>
6. Видеоурок 4.9. JK-триггеры. Часть 3. (2024)
<https://www.youtube.com/watch?v=gtegCM-LRKQ>
7. JK статический переключение и хранение (2023)
<https://www.youtube.com/watch?v=xtykTYEMuiQ>
8. Разбираем устройство JK-триггера, его отличия от RS-триггера и собираем на макетной плате (2020)
<https://www.youtube.com/watch?v=4Yl23et2xRM>
9. Master-slave JK flip-flop (2017)
Ведущий-ведомый JK-триггер (2017)
<https://www.youtube.com/watch?v=rXHSB5w7CyE>

Список использованных источников (YouTube)

Триггеры - JK-триггер

10. JK flip-flop (2016)
JK-Триггер (2016)
https://www.youtube.com/watch?v=F1OC5e7Tn_o
11. JK flip-flop racing (2017)
<https://www.youtube.com/watch?v=st3mUEub99E>

MS-триггер

1. Видеоурок 4.13. MS-триггеры и триггеры Шмитта (2024)
<https://www.youtube.com/watch?v=UYBmGWVhVZw>

Список использованных источников (YouTube)

Регистры

1. РК6. Схемотехника. Регистры (2020)
<https://www.youtube.com/watch?v=fq9pOH6R0rM>
2. Основы цифровой схемотехники. 14. Регистры (2020)
<https://www.youtube.com/watch?v=Yxh7y6aHvXI>
3. Лекция 296. Кольцевой регистр (2016)
<https://www.youtube.com/watch?v=YrQZHD2bE78>
4. Лекция 111. Сдвиговый регистр (2013)
<https://www.youtube.com/watch?v=aq4JIHS9eF0>
5. Лекция 148. Сдвиговый регистр на статических D триггерах (2014)
<https://www.youtube.com/watch?v=8ViAnNbM5m4>
6. 0111. Сдвиговый регистр (2019)
https://www.youtube.com/watch?v=vOFC7_r_JPE
7. Видеоурок 4.19. Универсальные регистры. Часть 1. (2024)
<https://www.youtube.com/watch?v=MapdzKWoBpg>
8. Видеоурок 4.17. Сдвигающие регистры. Часть 1. (2024)
<https://www.youtube.com/watch?v=rCpExW8D7gQ>
9. Видеоурок 4.18. Сдвигающие регистры. Часть 2. (2024)
<https://www.youtube.com/watch?v=x74L0RHhvpm>

Список использованных источников (YouTube)

Регистры

10. Видеоурок 4.14. Регистры D-типа. Часть 1. (2024)
<https://www.youtube.com/watch?v=ofZ11fa7AzU>
11. Видеоурок 4.15. Регистры D-типа. Часть 2. (2024)
<https://www.youtube.com/watch?v=UP0K9SRF9ns>
12. Видеоурок 4.16. Регистры D-типа. Часть 3. (2024)
<https://www.youtube.com/watch?v=XydeHPVr7EE>
13. Регистры (2022)
<https://www.youtube.com/watch?v=1LO0rrk-Cgc>
14. Цифровые устройства и программируемые логические интегральные схемы. Лекция №11 (12.04.2022) (2022) <https://www.youtube.com/watch?v=w0BeTnjnEEw>
15. Лабораторная работа № 6. Регистры и счетчики (2021)
<https://www.youtube.com/watch?v=13YKegt4L4Q>
16. Introduction to Registers | What is Shift Register? Types of Shift Registers (2022)
Введение в регистры | Что такое сдвиговый регистр? Типы сдвиговых регистров (2022)
<https://www.youtube.com/watch?v=bAQfPQqKCHs>
17. Introduction to Registers (2015)
Введение в регистры (2015)
<https://www.youtube.com/watch?v=-paFaxtTCkl>

Список использованных источников (YouTube)

Регистры

18. Data Formats and Classification of Registers (2015)
Форматы данных и классификация регистров (2015)
https://www.youtube.com/watch?v=b43_l4r1R2c
19. Designing and building a 1-bit register - 8 bit register - Part 3 (2016)
Проектирование и создание 1-разрядного регистра - 8-разрядный регистр - Часть 3 (2016)
https://www.youtube.com/watch?v=-arYx_oVlj8
20. Building an 8-bit register - 8-bit register - Part 4 (2016)
Создание 8-разрядного регистра - 8-разрядный регистр - Часть 4 (2016)
https://www.youtube.com/watch?v=CiMaWbz_6E8
21. Testing our computer's registers - 8-bit register - Part 5 (2016)
Тестирование регистров нашего компьютера - 8-разрядный регистр - Часть 5 (2016)
<https://www.youtube.com/watch?v=9WE3Obdjtv0>

Список использованных источников (YouTube)

Счетчики

1. Лекция 106. Двоичные счетчики (2013)
<https://www.youtube.com/watch?v=xNO3UPW3QB4>
2. Лекция 107. Реверсивный счетчик (2013)
<https://www.youtube.com/watch?v=nOXvpFcRvHc>
3. Лекция 114. Счетчики с произвольным коэффициентом пересчета (2013)
https://www.youtube.com/watch?v=kl94_00tSJc
4. Как работает двоичный счётчик? Объясняем на пальцах! (2022)
https://www.youtube.com/watch?v=v_shMnmFnns
5. РК6. Схемотехника. Счётчики: введение (2020)
<https://www.youtube.com/watch?v=lnvpaVga87A>
6. РК6. Схемотехника. Счётчики: асинхронные с ускоренным переносом (2020)
<https://www.youtube.com/watch?v=7LSXNoMp99U>
7. РК6. Схемотехника. Счётчики: синхронные с ускоренным переносом (2020)
https://www.youtube.com/watch?v=pLbeK_KBdc8
8. РК6. Схемотехника. Счётчики: синтез синхронных двоичных счётчиков с заданным модулем счёта (2020) <https://www.youtube.com/watch?v=PMPUCmxs2B4>
9. Основы цифровой схемотехники. 15. Счётчики (2020)
<https://www.youtube.com/watch?v=2LVVOpRNAT4>

Список использованных источников (YouTube)

Счетчики

10. 0106. Двоичные счётчики (2019)
<https://www.youtube.com/watch?v=obuMbM3ruEI>
11. Видеоурок 4.39. Счетчики с недвоичным кодированием (2025)
<https://www.youtube.com/watch?v=AjGMztivnjY>
12. Видеоурок 4.38. Счетчики с переменным управляемым коэффициентом счета (2025)
<https://www.youtube.com/watch?v=51kTdYk4Lx4>
13. Видеоурок 4.35. Каскадное соединение счетчиков. Часть 1. (2025)
<https://www.youtube.com/watch?v=ZWoDK6fQ8AE>
14. Видеоурок 4.36. Каскадное соединение счетчиков. Часть 2. (2025)
<https://www.youtube.com/watch?v=k3tMPUtfB3M>
15. Видеоурок 4.37. Каскадное соединение счетчиков. Часть 3. (2025)
https://www.youtube.com/watch?v=8XB_SKjmCkk
16. Видеоурок 4.29. Счетчики с параллельным переносом. Часть 1. (2025)
https://www.youtube.com/watch?v=ogSICHi1_MA
17. Видеоурок 4.30. Счетчики с параллельным переносом. Часть 2. (2025)
<https://www.youtube.com/watch?v=mCyNDCnKJaA>
18. Видеоурок 4.31. Счетчики с параллельным переносом. Часть 3. (2025)
<https://www.youtube.com/watch?v=IkpJPzYJBRM>

Список использованных источников (YouTube)

Счетчики

19. Видеоурок 4.32. Счетчики с параллельным переносом. Часть 4. (2025)
<https://www.youtube.com/watch?v=Lw76fyoNMpc>
20. Видеоурок 4.33. Счетчики с параллельным переносом. Часть 5. (2025)
https://www.youtube.com/watch?v=_MYmYTKgi0g
21. Видеоурок 4.34. Счетчики с параллельным переносом. Часть 6. (2025)
<https://www.youtube.com/watch?v=B8BWW1nW3Gk>
22. Видеоурок 4.25. Счетчики с последовательным переносом. Часть 1. (2024)
<https://www.youtube.com/watch?v=yPpqYIZpEbg>
23. Видеоурок 4.26. Счетчики с последовательным переносом. Часть 2. (2024)
<https://www.youtube.com/watch?v=xkqmy9d7tyM>
24. Видеоурок 4.27. Счетчики с последовательным переносом. Часть 3. (2025)
<https://www.youtube.com/watch?v=IBczil0TzzE>
25. Видеоурок 4.28. Счетчики с последовательным переносом. Часть 4. (2025)
<https://www.youtube.com/watch?v=lqbYER53LxE>
26. Видеоурок 4.24. Классификация счетчиков (2024)
https://www.youtube.com/watch?v=_W3XySn43Ow
27. Видеоурок 4.23. Цифровые счетчики. Основные понятия. (2024)
https://www.youtube.com/watch?v=dm_mZIRZ3xA

Список использованных источников (YouTube)

Счетчики

28. Микросхема-счетчик. Как это работает на примере микросхемы к17бие4 (2020)
<https://www.youtube.com/watch?v=qpSgM7yvFtg>
29. Информатика, 11 практическая работа - Счётчики (2023)
<https://www.youtube.com/watch?v=ILAj0qytUWo>
30. ИИИ - ИРИ: Информатика, 10 практическая работа - Счётчики (2023)
<https://www.youtube.com/watch?v=DvOf3erQpvA>
31. Счетчики (2022)
<https://www.youtube.com/watch?v=mJ3LFgeVlVE>
32. Introduction to Counters | Important (2015)
Введение в счетчики | Важно (2015)
<https://www.youtube.com/watch?v=ialu5SYmWVM>
33. Types of Counters | Comparison between Ripple and Synchronous counters (2015)
Типы счетчиков | Сравнение пульсационных и синхронных счетчиков (2015)
<https://www.youtube.com/watch?v=yqglsqhZG3M>
34. Ring Counter (2015) / Счетчик звонков (2015)
<https://www.youtube.com/watch?v=yOW-JsJL1Ks>
35. Synchronous Counters Explained (Part-1) (2022)
Объяснение работы синхронных счетчиков (Часть 1) (2022)
<https://www.youtube.com/watch?v=dlianWwz-Fw>

Список использованных источников (YouTube)

Счетчики

36. Digital Logic – Counters (2013)
Цифровые логические счетчики (2013)
<https://www.youtube.com/watch?v=ZiAbLltaz4A>
37. Binary Counter (2023)
Двоичный счетчик (2023)
<https://www.youtube.com/watch?v=UKlUIaojVql>
38. Program counter design (2017)
Разработка программного счетчика (2017)
https://www.youtube.com/watch?v=g_1HyxBzjl0

Список использованных источников (YouTube)

Арифметико-логические устройства (АЛУ)

1. Архитектура ЭВМ. Лекция 2: АЛУ. Устройство памяти (2021)
<https://www.youtube.com/watch?v=HPGw4qFARTg>
2. Лекция 323. Atmega 8: Принцип работы АЛУ (2016)
<https://www.youtube.com/watch?v=EcksJCdlQog>
3. Видеоурок 3.39. Арифметико-логические устройства (АЛУ). Часть 1 (2023)
<https://www.youtube.com/watch?v=O-90dCQKTrY>
4. Видеоурок 3.40. Арифметико-логические устройства (АЛУ). Часть 2 (2023)
<https://www.youtube.com/watch?v=rEsbmJTqM7Y>
5. Видеоурок 3.41. Арифметико-логические устройства (АЛУ). Часть 3 (2023)
<https://www.youtube.com/watch?v=Pl785qcbKmk>
6. АПС Л3. Цифровая арифметика. АЛУ (2020)
<https://www.youtube.com/watch?v=eXW6Q4jPkmQ>
7. BitFlip. Собираем 8-битное арифметико-логическое устройство (АЛУ)(2020)
https://www.youtube.com/watch?v=6P_-kfUl8c
8. Арифметико-логическое устройство (2025)
<https://www.youtube.com/watch?v=X-P6Znxh-zQ>
9. 29 АЛУ процессора (2024)
<https://www.youtube.com/watch?v=HNuraUZhcz4>

Список использованных источников (YouTube)

Арифметико-логические устройства (АЛУ)

10. Видеолекция. Одноразрядные и многоразрядные суммирующие схемы (2021)
<https://www.youtube.com/watch?v=D6IzEOc0Gy4>
11. Электронные мысли полупроводниковых машин или как работают процессоры (2019)
<https://www.youtube.com/watch?v=IOWe8RO8-xo>
12. Арифметико-логическое устройство. Сложение двоичных чисел. (2019)
<https://www.youtube.com/watch?v=ded0iOVZhSA>
13. What Is Arithmetic Logic Unit ? | ALU In Computer Architecture Explained (2022)
Что такое арифметико-логическая единица измерения? | Объяснен ALU в области компьютерной архитектуры (2022)
https://www.youtube.com/watch?v=H_aoaQYgKT8
14. Computer Architecture Lecture 8: The Arithmetic Logic Unit (ALU) (2022)
Лекция 8 по компьютерной архитектуре: Арифметико-логический модуль (ALU) (2022)
<https://www.youtube.com/watch?v=VGHk1tU9xZ0>
15. 39 Arithmetic Logic Unit (ALU) (2024)
39 Арифметико-логический блок (ALU) (2024)
<https://www.youtube.com/watch?v=TUNmWdrGxX4>
16. 23.1) Arithmetic and Logic Unit (ALU) (2020)
23.1) Арифметико-логический модуль (АЛУ) (2020)
<https://www.youtube.com/watch?v=dcsYdaO6txI>

Список использованных источников (YouTube)

Арифметико-логические устройства (АЛУ)

17. 23.2) Elementary ALU Design (2020)
23.2) Элементарное проектирование АЛУ (2020)
<https://www.youtube.com/watch?v=IPnYBgCmTCA>
18. The 74LS181 (4-bit Arithmetic Logic Unit) (2023)
74LS181 (4-разрядный арифметико-логический модуль) (2023)
<https://www.youtube.com/watch?v=rFILT3lnVdw>
19. Building the ALU (2016)
Создание АЛУ (2016)
<https://www.youtube.com/watch?v=S-3fXU3FZQc>
20. ALU Design (2016)
Проектирование АЛУ (2016)
<https://www.youtube.com/watch?v=mOVOS9AjqFs>
21. Testing the computer's ALU (2016)
Тестирование АЛУ компьютера (2016)
<https://www.youtube.com/watch?v=4nCMDvnR2Fg>

Список использованных источников (YouTube)

YouTube каналы и плейлисты

1. Электротехника и электроника для программистов
<https://www.youtube.com/@Zefar91/featured>
2. BitFlip - Канал об электронике для начинающих
<https://www.youtube.com/@BitFlipChannel>
3. Алексей Аббясов
<https://www.youtube.com/@АлексейАббясов-д7y/videos>
4. Инна Музылева
<https://www.youtube.com/@ИннаМузылева/videos>
5. Madiyar Nurgaliyev
<https://www.youtube.com/@madiyarnurgaliyev6421/videos>
6. Мурат Нсанов - Цикл видеоуроков по предмету «Цифровая микроэлектроника»
<https://www.youtube.com/@МуратНсанов/featured>
7. Дмитрий Гайворонский
<https://www.youtube.com/@ДмитрийГайворонский-ю2г>
8. Электротехника для программистов :: Практика
<https://www.youtube.com/@Электротехникадляпрограммистов/videos>
9. Плейлист - Электротехника
https://www.youtube.com/playlist?list=PLCKtVYi1kU04xx_6n2Uw1gj4KMG8NY1_X

Список использованных источников (YouTube)

YouTube каналы и плейлисты

10. Плейлист - Раздел 4. Последовательностные цифровые устройства / Мурат Нсанов
<https://www.youtube.com/playlist?list=PLN-wZHgzUk55cot8-IX6zs3XzNqUF4YfV>
11. Ben Eater
<https://www.youtube.com/@BenEater/featured>