

# Схемотехника ЭВМ.

## Конспект лекций. Рабочая версия.

Кустарев П.В.  
Кафедра Вычислительной Техники  
СПбГУ ИТМО  
Санкт-Петербург 2009г.

### ОГЛАВЛЕНИЕ.

Введение. Цифровая электроника.....	3
<b>I. ЦИФРОВАЯ СХЕМОТЕХНИКА .....</b>	<b>4</b>
<b>1 Аналоговые и цифровые сигналы. ....</b>	<b>4</b>
<b>2 Логические состояния. ....</b>	<b>5</b>
<b>3 Параметры цифровых сигналов. ....</b>	<b>6</b>
3.1 Диапазоны напряжений для логических «0» и «1». ....	6
3.2 Нагрузочная способность (коэффициент разветвления по выходу) .....	9
3.3 Длительность переключения состояния .....	10
3.4 Задержка распространения сигналов. ....	11
<b>4 Схемотехника цифровых микросхем. ....</b>	<b>11</b>
4.1 Введение. ....	11
4.2 Типов цифровых микросхем.....	11
4.3 Базовые элементы и каскады цифровых КМОП – микросхем. ....	12
4.4 Логические элементы на МОП-транзисторах. ....	13
4.5 Типы входных выходных каскадов цифровых схем. ....	17
<b>5 Цифровые элементы и схемы. ....</b>	<b>23</b>
5.1 Комбинационные последовательностные цифровые схемы. ....	23
5.2 Логические элементы и базовые операционные элементы цифровых схем. ....	23
<b>6 Базовые операционные элементы комбинационного типа.....</b>	<b>25</b>
6.1 Мультиплексор (multiplexer, MUX) .....	25
6.2 Демультимплексор (demultiplexer, DEMUX) .....	26
6.3 Дешифратор (decoder, DC).....	27
6.4 Шифратор (encoder).....	29
<b>7 Триггеры.....</b>	<b>31</b>
7.1 Классификация триггеров.....	31
7.2 Структура триггера.....	32
7.3 Синхронные управляемые фронтом (универсальные) триггеры. ....	38
<b>8 РЕГИСТРЫ (в процессе редактирования) .....</b>	<b>45</b>
8.1 Накопительные регистры.....	45
<b>9 Сдвигающие регистры.....</b>	<b>48</b>
9.1 Сдвигающие регистры на асинхронных RS-триггерах. ....	48
9.2 Реверсные сдвигающие регистры. ....	55
<b>10 СЧЕТЧИКИ (в процессе редактирования) .....</b>	<b>56</b>
10.1 Асинхронные двоичные счетчики .....	56
10.2 Счетчики с ускоренным распространением переноса.....	59
10.3 Синхронные двоичные счетчики.....	60
10.4 Счетчики с произвольным модулем доступа .....	62
10.5 Беззвентильные счетчики. ....	65
10.6 Десятичные счетчики. ....	66
10.7 Реверсивные счетчики.....	68

## II. СХЕМОТЕХНИКА МИКРОПРОЦЕССОРНЫХ СИСТЕМ. 69

<b>11 Основные понятия области микропроцессорных систем.....</b>	<b>69</b>
<b>12 Структура микропроцессорных систем.....</b>	<b>70</b>
12.1 Магистралы (шины).....	70
12.2 Шина адреса.....	71
12.3 Шина данных.....	72
12.4 Мультиплексированные шины адреса-данных.....	72
12.5 Шина управления.....	73
<b>13 Подключение системной памяти. ....</b>	<b>81</b>
13.1 Общие сведения.....	81
13.2 Стандартизация интерфейсов микросхем памяти.....	82

## ВВЕДЕНИЕ. ЦИФРОВАЯ ЭЛЕКТРОНИКА.

Схемотехника – область науки и техники изучающая принципы функционирования и технологии проектирования схем электронных устройств. Базисом схемотехнического проектирования являются дискретные электронные компоненты, усилительные, интегрирующие, генераторные и другие аналоговые элементы, цифровые логические элементы, элементы памяти и другие. Схемотехника является разделом электроники. Наиболее связанными со схемотехникой являются полупроводниковая электроника (базовые элементы, каскады), микроэлектроника (микросхемы), технология производства электронных компонентов и устройств.

В соответствии с типом электрических сигналов и функциями компонентов схем разделяют аналоговую и цифровую схемотехнику. *Данный курс лекций посвящен цифровой схемотехнике, как основе большинства современных вычислительных устройств.*

Цифровая схемотехника рассматривает схемы, в которых электрические сигналы, могут иметь лишь два устойчивых (дискретных) уровня напряжения: «высокий» и «низкий». Переход между этими уровнями происходит скачкообразно, в отличие от аналоговых сигналов, которые допускают произвольные уровни сигналов в некотором «рабочем» диапазоне и могут изменяться непрерывно. В цифровых схемах электронные каскады (обычно построенные на транзисторах) работают в режиме переключателя и находятся в одном из двух крайних состояний: пропускания (включены) или запирающего (выключены).

Главным достоинством цифровой электронной техники является схемотехническая простота каскадов при высокой помехоустойчивости. Это происходит потому что распознавание только двух сильно отличающихся уровней напряжения реализовать значительно проще, чем обработку аналогового сигнала. При этом обеспечивается высокая надежность работы схем даже при наличии больших искажений и помех. Следствие более простых схемотехнических решений – сравнительная простота методов проектирования и технологии производства.

С точки зрения функциональных возможностей цифровая электроника характеризуется удобством кодирования и обработки чисел, представленных в двоичном формате. За счет этого она стала физической основой вычислительной техники, что вывело ее на первое место по востребованности среди иных видов электроники. Применение микропроцессорных вычислительных устройств кардинально расширяет возможности обработки и хранения данных, возможности настройки и программирования самой системы.

С точки зрения конструктивных и эксплуатационных преимуществ использование цифровой электроники упрощает конструкцию, уменьшает габариты и массу устройств, снижает требования к качеству электропитания и в целом увеличивает надежность работы систем. Значительным преимуществом цифровых схем по сравнению с аналоговыми является простота их изготовления в виде интегральных микросхем, особенно БИС и СБИС.

Цифровая электронная техника находит широкое применение (на нынешнем этапе - является основой) в электронно-вычислительных устройствах различного назначения (персональные компьютеры, серверы, встроенные управляющие системы), в системах связи и передачи данных, в системах хранения медиаданных (CDROM, DVD и т.п.), в измерительных приборах, в бытовой технике и других областях.

## I. ЦИФРОВАЯ СХЕМОТЕХНИКА

### 1 АНАЛОГОВЫЕ И ЦИФРОВЫЕ СИГНАЛЫ.

*Аналоговые электрические сигналы* – сигналы, изменяющиеся во времени непрерывно и способные принимать любое значение в некотором диапазоне напряжений, тока, частоты или иных характеристик (метрик). Аналоговая природа естественна для многих физических процессов и сигналов – звука, перемещения, изменения температуры и т.п. Поэтому метрики данных физических процессов/сигналов удобно (и естественно) переводить в аналоговые электрические сигналы с целью дальнейшей их преобразования электронными схемами. Например, температура 25.256 градусов Цельсия может быть закодирована как напряжение 2.5256 В. Самыми большими проблемами использования аналоговых сигналов являются:

- их чувствительность к помехам, приводящая к искажению значений (например, в вышеприведенном примере помеха 0.1В приведет к ошибке температуры на 1 градус Цельсия);
- высокие погрешности обработки каскадами электронных схем (усиления, интегрирования и т.п.), связанные с сложностью/невозможностью изготовления электронных компонентов (резисторов, конденсаторов, транзисторов, микросхем) с параметрами (сопротивления, емкости, коэффициентами передачи и т.п.) высокой и сверхвысокой точности (до тысячных процента) и стабильности в диапазоне температур, давлений и т.д.

*Дискретные электрические сигналы* – сигналы, для которых допускаются лишь значения из заранее определенного ограниченного множества. Значения указываются с допустимой погрешностью. Например, дискретный электрический сигнал имеет три допустимых значения напряжений: 0В, 5В и 10В, с допуском  $\pm 1В$ . Дискретными могут быть физические процессы и сигналы. Например, состояние управляющей клавиши (вкл/выкл – 2 значения) или датчика установленной передачи в коробке передач автомобиля (количество дискретных значений равно числу передач) или импульсы в детекторе элементарных частиц (есть/нет). Использование дискретных сигналов имеет важное преимущество – допустимость установки значения с некоторой значительной погрешностью, что резко повышает помехоустойчивость и снижает требования к точности параметров электронных каскадов.

*Цифровые электрические сигналы* – так обычно называют те дискретные сигналы, которые имеют только два допустимых состояния. Данные состояния (например, уровни напряжения 0В и 5В) кодируют две цифры – «0» и «1». Данные цифры эквивалентны допустимым значениям разрядов двоичного представления чисел (**двоичный разряд – binary digits или bit**), а также допустимым значениям переменных в алгебре логики (булевой алгебре) – «Истина» (TRUE или «1») и «Ложь» (FALSE или «0»), что позволяет кодировать эти числа в виде цифровых электрических сигналов. С помощью простейших транзисторных каскадов, работающих в самом простом - ключевом режиме (включен/выключен), можно реализовать основные функции алгебры логики (логические (булевы) функции) и, их (логических функций) посредством, основные математические функции (сложение, вычитание, умножение, деление) для чисел в двоичном представлении. Существуют различные варианты схем хранения (памяти) для двухуровневых (цифровых) значений. Двухуровневый цифровой сигнал легко передавать на значительные расстояния при значительных помехах (например, «1» - напряжение =  $10\pm 5В$ , «0» - напряжение =  $1.5\pm 1.5В$ ), причем не только по электрическим проводам, но и по каналам других типов, например по оптоволоконному кабелю («свет» включен/выключен).

Все эти свойства позволили положить цифровые сигналы в основу современных вычислительных устройств, в частности, микропроцессоров, и в основу систем хранения и передачи данных.

## 2 ЛОГИЧЕСКИЕ СОСТОЯНИЯ.

Для кодирования значений логических переменных или двоичных разрядов (битов) обычно используется напряжение. Ток, частота и другие характеристики сигнала тоже применяются, но только в специальных случаях - в основном при передаче данных или как удобный вариант сопряжения электрических каскадов.

Допустимые уровни напряжения соответственно их значениям условно называют **ВЫСОКИМ (HIGH)** и **НИЗКИМ (LOW)**. Как говорилось выше, уровень соответствует не одному, а диапазону значений напряжений: например, 2,5..5В – **ВЫСОКИЙ** уровень, 0..1 В – **НИЗКИЙ** уровень, но для удобства указывают только «номинальный» (обычно крайний по значению) уровень, например, 5В и 0В.

Двум указанным уровням напряжения можно сопоставить пару логических значений (логических состояний, двоичных цифр).

Если **ВЫСОКИЙ** уровень напряжения цифрового сигнала соответствует значению «1» или «ИСТИНА», а **НИЗКИЙ** уровень напряжения соответствует значению «0» или «ЛОЖЬ», то такой способ кодирования логической переменной называется **ПОЗИТИВНОЙ (ПОЛОЖИТЕЛЬНОЙ) ЛОГИКОЙ**.

Если **ВЫСОКИЙ** уровень напряжения цифрового сигнала соответствует значению «0» или «ЛОЖЬ», а **НИЗКИЙ** уровень напряжения соответствует значению «1» или «ИСТИНА», то такой способ кодирования логической переменной называется **НЕГАТИВНОЙ (ОТРИЦАТЕЛЬНОЙ) ЛОГИКОЙ**.

Тип логики (**ПОЗИТИВНАЯ** или **НЕГАТИВНАЯ**) является не только характеристикой собственно цифрового сигнала, но также и характеристикой цифрового элемента (блока, схемы), который обрабатывает данный сигнал исходя именно из такого способа его кодирования. Например, элемент популярной логической микросхемы SN7408 в документации полностью именуется «двухвходовой элемент «И» с позитивным кодированием сигналов». Если же использовать негативное кодирование, то функция данного элемента изменится на «ИЛИ».

Современная элементная база и схемотехника в целом ориентирована на позитивную (положительную) логику. Однако в некоторых случаях негативная (отрицательная) логика может оказаться более удобным способом кодирования цифровых или логических значений. Например, схема определения нажатия кнопки на клавиатуре часто построена таким образом, что **ВЫСОКИЙ** уровень вырабатывается, если кнопка не нажата, и **НИЗКИЙ** – при нажатии кнопки. То есть, если кодировать факт нажатия кнопки как «ИСТИНА» и при этом вырабатывается **НИЗКИЙ** уровень сигнала, то получаем негативное (отрицательное) кодирование. Часто удобство негативной логики для сигналов цифровых элементов определяется особенностями внутренней схемотехники этих элементов.

Чтобы не путаться с тем, какие элементы в схеме используют позитивное кодирование, а какие негативное, принято соглашение всеми элементами в схеме используется один тип кодирования сигналов (например, позитивное), а если на входе или выходе какого-нибудь элемента должен формироваться сигнал с негативным кодированием, то он преобразуется из/в позитивный путем инвертирования. Такие инвертированные сигналы обозначаются на схемах чертой над названием сигнала (знак булевой операции «отрицание»), а вход или выход элемента, на котором выполняется инверсия сигнала (зачастую это мнимое инвертирование – схема использует внутри себя непосредственно негативно закодированный сигнал), обозначается кружочком.

*Примечания:*

- 1) В силу большей естественной воспринимаемости (принцип «большему соответствует большее») и распространенности положительной логики на схемотехническом сленге часто называют **ВЫСОКИЙ** уровень напряжения - «1», а **НИЗКИЙ** уровень напряжения – «0». Таким образом, в случае использования отрицательной логики может возникнуть путаница: говоря о «единице на сигнальной линии», подразумевают **ВЫСОКИЙ** уровень напряжения, который на самом деле соответствует логическому значению «0».
- 2) Термины «*позитивная*» логика и «*положительная*» логика, а также «*негативная*» и «*отрицательная*» логика эквивалентны и в различных комбинациях встречаются в литературе. Первоисточник – английские слова «*positive*» и «*negative*». Так же встречается вариант «*прямая*»-«*инверсная*» логика (подразумевается, что сигнал с негативной логикой («*инверсный*») может быть получен путем инверсии сигнала с позитивной логикой («*прямого*»).

### 3 ПАРАМЕТРЫ ЦИФРОВЫХ СИГНАЛОВ.

Параметрами *реальных* цифровых сигналов, наиболее важными для схемотехнического проектирования, являются:

- Диапазон напряжений для логических «0» и «1», для выходов логических элементов/схем и для входов цифровых элементов/схем;
- Нагрузочная способность (коэффициент разветвления по выходу) выходов цифровой схемы - fanout;
- Длительность переключения состояния – время измерения состояния сигнала с НИЗКОГО уровня на ВЫСОКИЙ и наоборот (перехода из логического «0» в «1» и наоборот) – transition time;
- Временная задержка цифрового сигнала при «прохождении» через логический элемент/схему – propagation delay.

#### 3.1 Диапазоны напряжений для логических «0» и «1».

Так как именно напряжение используется для кодирования значений «0» и «1», то диапазон напряжений для логических «0» и «1» являются основным параметром цифровых схем. При этом каждому из логических уровней «0» и «1» соответствуют не фиксированные значения напряжения, например, 0В или 5В, а некоторый диапазон напряжений. Например, для микросхем семейства ТТЛ логическому «0» будет соответствовать напряжение, попадающее в диапазон от 0В до +0.8В, а логической «1» будет соответствовать напряжение в диапазоне от +2В до +5В. Кодирование логических уровней диапазонами сделано потому что:

- 1) Позволяет использовать цифровые элементы/схемы с достаточно значительными, допусками параметров входных и выходных каскадов, что сильно удешевляет их производство.
- 2) Допускает колебание параметров элементов/схем и соответствующих цифровых сигналов за счет изменения температур, электрической нагрузки и напряжения питания схем и т.п.
- 3) Позволяет игнорировать влияние шумов – паразитных напряжений, которые добавляются/вычитаются из рабочего напряжения при «прохождении» его через схему. Шумы возникают за счет емкостных и индуктивных связей между сигналами в схеме, помех приходящих по подключенным внешним цепям и цепям питания, за счет электромагнитных наводок.

Диапазоны напряжений цифровых сигналов, генерируемые выходами цифровых схем и воспринимаемые входами схем, делают разными. Диапазон, воспринимаемый входами более широкий по сравнению с диапазоном выходных сигналов, и диапазон выходов целиком перекрывается диапазоном входов, оставляя запас по границе минимального и

максимального напряжений. Это гарантирует, что выходной сигнал вырабатываемый одной цифровой схемой и подаваемый на вход другой будет правильно восприниматься даже в условиях помех. Например, выход вырабатывает ВЫСОКИЙ уровень в диапазоне 4.5В – 5В, а вход будет воспринимать ВЫСОКИЙ уровень в диапазоне 3.5В-5.5В. Поэтому, если к выходному напряжению ВЫСОКОГО уровня равному 4.5В добавится помеха 1В, то суммарное напряжение будет 5.5В и будет воспринято входом верно - как ВЫСОКИЙ уровень.

Между диапазонами ВЫСОКОГО уровня и НИЗКОГО уровня располагается так называемая «мертвая зона». В пределах мертвой зоны производитель не гарантирует корректное восприятие уровня сигнала. Около середины мертвой зоны (но не точно) располагается пороговый уровень  $U_{вх.п}$  ( $V_{in.t}$ , threshold voltage), ниже которого уровень сигнала на входе воспринимается как НИЗКИЙ, а выше – как ВЫСОКИЙ. Номинальное значение  $U_{п}$  определяется документацией на электронный компонент (микросхему), но реальное значение может смещаться в рамках мертвой зоны в зависимости от особенностей конкретного образца (микросхемы), от температуры, от старения компонента, от напряжения питания и других параметров.

Итого: среди основных параметров цифровых схем должны быть заданы следующие напряжения цифровых сигналов:

- Для цифровых входов:
  - $U_{вх.0.мин.}$  ( $V_{IL.min}$ ) – минимальное напряжение, воспринимаемое как «0»;
  - $U_{вх.0.макс.}$  ( $V_{IL.max}$ ) – максимальное напряжение, воспринимаемое как «0»;
  - $U_{вх.1.мин.}$  ( $V_{IH.min}$ ) – минимальное напряжение, воспринимаемое как «1»;
  - $U_{вх.1.макс.}$  ( $V_{IH.max}$ ) – максимальное напряжение, воспринимаемое как «1»;
  - $U_{вх.п}$  ( $V_{IT}$ ) – напряжение переключения (threshold voltage), значения выше которого воспринимаются как «1», а ниже – как «0».
- Для цифровых выходов:
  - $U_{вых.0}$  ( $V_{OL.typ}$ ) – типовое напряжение, которое устанавливается при выводе «0»;
  - $U_{вых.0.мин.}$  ( $V_{OL.min}$ ) – минимальное напряжение, которое может быть установлено при выводе «0»;
  - $U_{вых.0.макс.}$  ( $V_{OL.max}$ ) – максимальное напряжение, которое может быть установлено при выводе «0»;
  - $U_{вых.1}$  ( $V_{OH.typ}$ ) – типовое напряжение, которое устанавливается при выводе «1»;
  - $U_{вых.1.мин.}$  ( $V_{OH.min}$ ) – минимальное напряжение, которое может быть установлено при выводе «1»;
  - $U_{вых.1.макс.}$  ( $V_{OH.max}$ ) – максимальное напряжение, которое может быть установлено при выводе «1».

Указанные напряжения зависят от схемотехники и параметров выходных и входных электрических каскадов цифровых схем. На рисунке представлены параметры наиболее распространенных семейств цифровых микросхем:

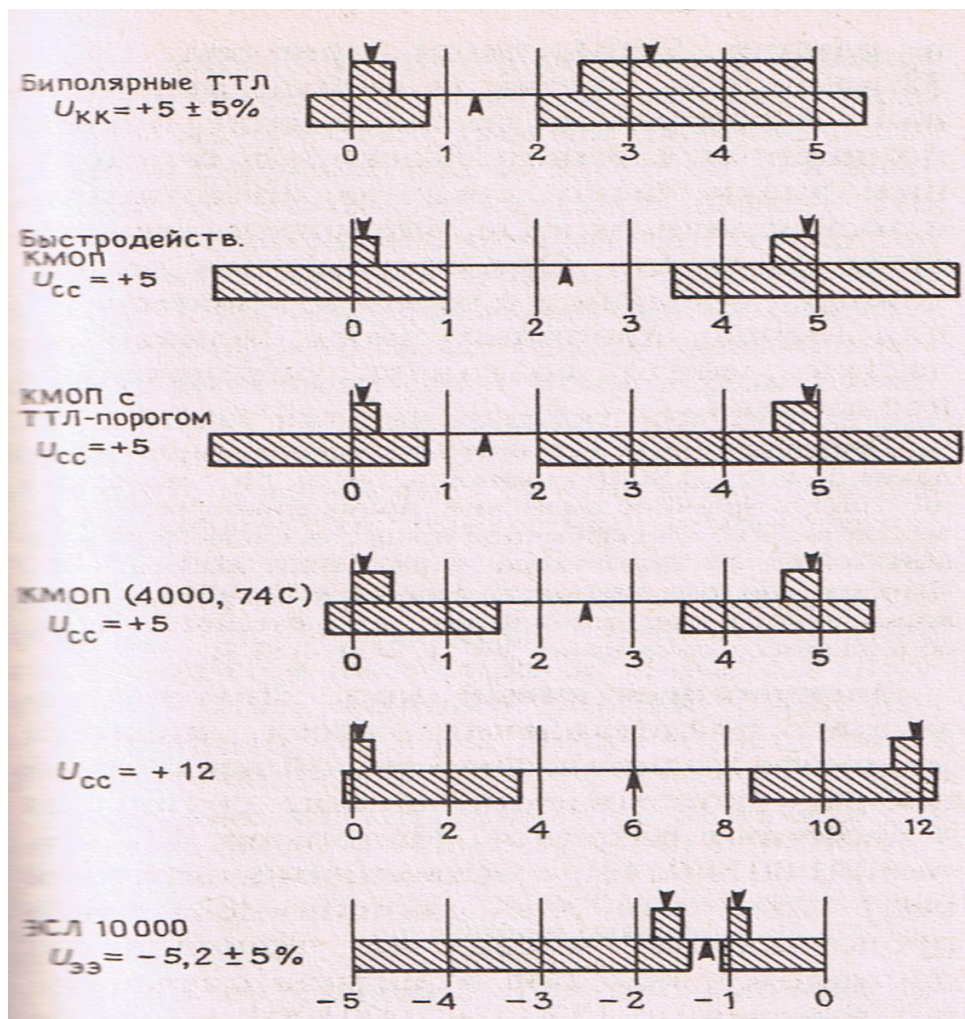
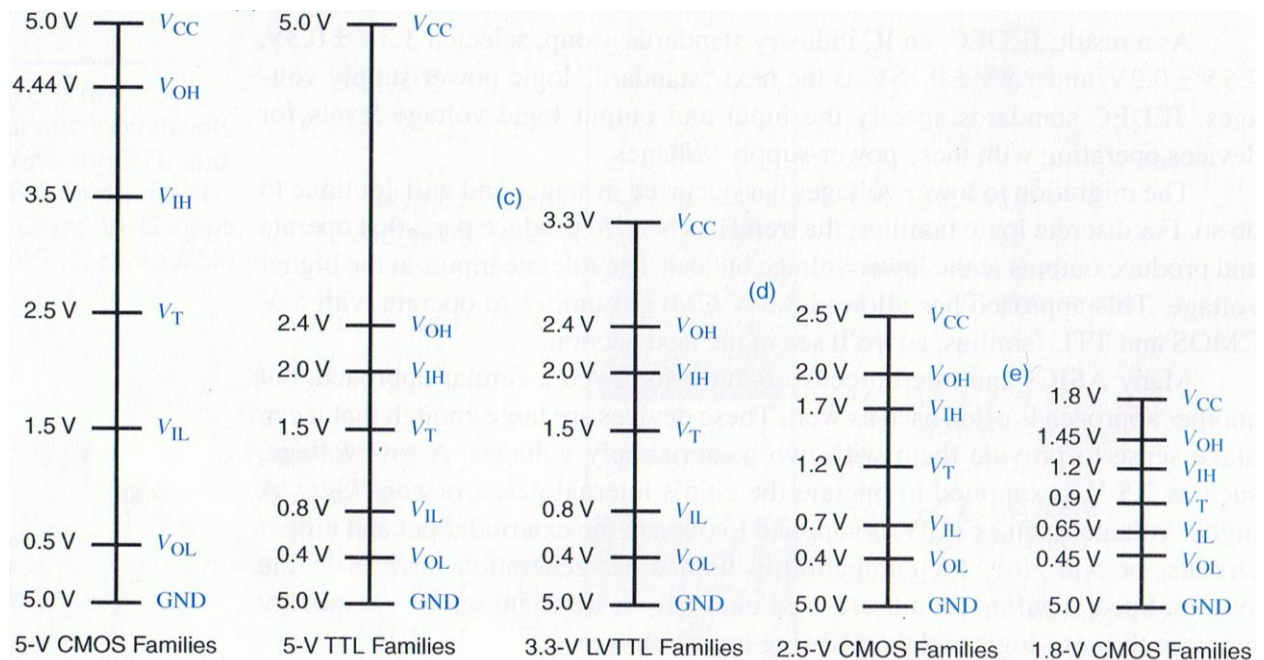


Рисунок 1 Уровни цифровых сигналов распространенных семейств цифровых микросхем

Из рисунка видно, что уровни напряжений для различных семейств микросхем не совпадают полностью. Например, это справедливо для распространенных семейств микросхем со схемотехникой ТТЛ (TTL) и КМОП (CMOS). ВЫСОКИЙ уровень на выходе ТТЛ может быть равен 2.5В, при этом для схем семейства КМОП это будет недопустимый уровень, который может считаться как ВЫСОКИЙ или как НИЗКИЙ. В результате этого схема будет работать неправильно и подключение микросхем одного семейства к микросхемам другого не допустимо. Чтобы преодолеть эту проблему, разрабатывают специальные семейства микросхем с диапазоном напряжений совместимым с другими семействами. Например, таким семейством является КМОП с ТТЛ-порогом. Однако в некоторых случаях совместимости добиться сложнее: например, у семейства микросхем ЭСЛ уровни напряжений вообще не пересекаются с другими семействами – они отрицательные. Единственный способ подключения их к ТТЛ- или КМОП-схемам, это использование специальных буферных каскадов.



Еще одна особенность/проблема – это использование цифровых микросхем с различными напряжениями питания. Дело в том, что при изменении напряжения питания микросхем, изменяются и уровни напряжения высокого и низкого уровня (см. рисунок ниже). На нынешний момент в цифровой технике наиболее распространенными являются напряжения питания 5В, 3.3В, 2.5В, 1.8В. Необходимость снижения напряжения питания вызвана многими причинами, основными из которых являются снижение потребляемой и выделяемой мощности, повышение быстродействия схем, уменьшение физических размеров транзисторов на кристалле интегральных микросхем.



**Рисунок 2** Уровни напряжения цифровых сигналов для микросхем с различным напряжением питания

Видно, что уровни схем с различным питанием не совместимы между собой. При этом их часто приходится использовать совместно в одной схеме. Например, электропитание микропроцессора может быть 1.8В, а питание подключенных к нему микросхем – 3.3В или 5В. И аналогов с иным питанием не производится! В таком случае добавляют специальные каскады/микросхемы преобразования уровней напряжения цифровых сигналов. Иногда эти каскады встроены в СБИС, например, в микропроцессоры. Иногда удается добиться частичной совместимости уровней, например, микросхема с питанием 3.3В допускает подключение к ней входных сигналов с напряжением до 5В с корректным распознаванием ВЫСОКОГО и НИЗКОГО уровней. Обратного подключения может не допускаться, например выходов «3.3В» ко входам КМОП «5В».

В заключении нужно отметить, что так как любое совместное использование схем с различными уровнями напряжений это потенциальный источник ошибок и часто причина усложнения схемы, то, без особой необходимости, стараются не делать смешанных схем.

### 3.2 Нагрузочная способность (коэффициент разветвления по выходу)

Нагрузочная способность выхода цифровой схемы показывает, какое количество входов цифровых схем может быть подключено к данному выходу без перегрузки выходных каскадов и без искажения уровней цифрового сигнала для входов. Нагрузочная способность зависит и устанавливается для пары типов «выход-вход». Например, для выхода типа X устанавливается количество подключаемых входов типа Y и количество подключаемых входов типа Z и т.п. Нагрузочная способность может различаться для уровней ВЫСОКИЙ и НИЗКИЙ, но обычно указывается только одно – меньшее значение.

Типовая нагрузочная способность – 20 входов того же типа, что и выход. Если к выходу одного типа подключены входы другого типа, то соотношение изменяется. Например, для наиболее популярных семейств КМОП и ТТЛ, к выходу ТТЛ можно подключить в 2-3 раза больше 20-ти входов типа КМОП (хотя это не рекомендуется). И наоборот, ко КМОП-выходу подключается 5-10 входов ТТЛ.

Ниже перечислены отрицательные последствия перегрузки выходов:

- Выходное напряжение НИЗКОГО уровня может превысить  $U_{вх.0.макс.}$  и НИЗКИЙ уровень будет определен как ВЫСОКИЙ;
- Выходное напряжение ВЫСОКОГО уровня может быть ниже  $U_{вх.1.мин.}$  и ВЫСОКИЙ уровень будет определен как НИЗКИЙ;
- Время изменения уровня с НИЗКОГО на ВЫСОКИЙ и обратно превышает значение, допустимое спецификацией данной схемы;
- Задержка распространения сигнала через схему превышает значение, допустимое спецификацией данной схемы;
- Перегрев элементов схемы из-за повышенного тепловыделения, возникающего из-за перегрузки. В результате может возникнуть изменение параметров схемы (уровней напряжения, нагрузочных способностей, параметров быстродействия) или физическая порча перегретых элементов.

### 3.3 Длительность переключения состояния

В идеальном случае ВСЕ выходы цифровой схемы или ее элемента изменяют свое состояние мгновенно и одновременно. Однако реальные выходы не могут моментально переключиться с ВЫСОКОГО на НИЗКИЙ уровень и наоборот: необходимо время на перезаряд паразитных емкостей элементов цифровой схемы или емкостей и индуктивностей проводников на плате. В итоге на рисунке идеальный сигнал (а) приобретает реальную форму (с). Условное изображение на временных диаграммах «постепенного перехода» выхода цифровой схемы из состояния в состояние показано на (b).

Время перехода с НИЗКОГО уровня в ВЫСОКИЙ ( $T_r$ ) называют «длительностью положительного фронта», иногда просто «длительность фронта», или rise time. Время перехода с ВЫСОКОГО уровня в НИЗКИЙ ( $T_f$ ) называют «длительностью отрицательного фронта», или «длительностью спада», или fall time. Эти времена обычно близкие по значению, но немного различаются у выходов цифровых схем. Для различных типов выходов (ТТЛ, КМОП и других) эти времена могут различаться в разы. Длительности переходов возрастают при подключении большего числа входов к выходу. Это объясняется, в основном, ростом значения емкости, подключенной к выходу за счет входных емкостей входов. Для наиболее распространенных на сегодня типа КМОП длительности переходов находятся в пределах 5-10 ns для типового числа подключенных входов. Для быстродействующих каскадов «внутри» СБИС процессоров, памяти и т.п. это время уменьшается до десятых – сотых наносекунды.

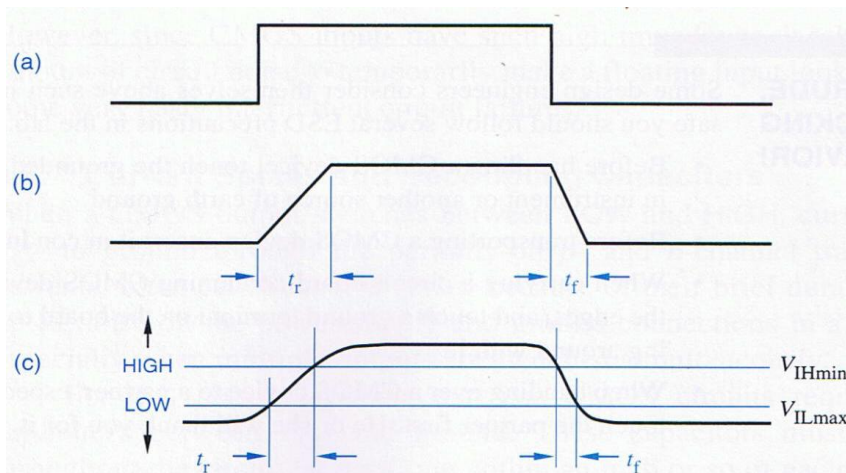


Рисунок 3 Идеальное (а) и реальное (b,c) переключение цифровых выходов

Задержка перехода является отрицательным фактором функционирования цифровых схем и, наряду с задержкой распространения сигнала, значительно усложняет их разработку. Основные причины этого:

- нахождение выхода в неопределенном состоянии приводит к возможности некорректного срабатывания входа, причем многократного;
- рассинхронизация в работе элементов/частей цифровых схем;
- повышенное энергопотребление во время нахождения в неопределенном состоянии.

#### 3.4 Задержка распространения сигналов.

Задержкой распространения сигнала через элемент (propagation delay,  $t_p$ ) называют время между фронтом (перепадом) цифрового сигнала на входе элемента и вызванным им (входным фронтом) перепадом сигнала на выходе элемента. Задержка распространения вызвана временем срабатывания транзисторных ключей внутри элемента. Она будет больше, чем больше количество таких ключей по пути распространения сигнала внутри элемента, т.е. количество последовательных каскадов. Задержка распространения может быть разной для перепада на выходе с НИЗКОГО на ВЫСОКИЙ уровень ( $t_{pLH}$ ) и для перепада с ВЫСОКОГО в НИЗКИЙ уровень ( $t_{pHL}$ ).

### 4 СХЕМОТЕХНИКА ЦИФРОВЫХ МИКРОСХЕМ.

#### 4.1 Введение.

В современной цифровой технике разработчик имеет дело почти исключительно с интегральными микросхемами: Ему достаточно знать функцию элементов, корректно подключить сигналы схемы к входам и выходам микросхемы, и обеспечивая электропитание микросхемы. При этом не требуется вникать в подробности внутреннего устройства микросхем.

Однако на практике одни и те же логические функции могут реализовать микросхемы, различающиеся напряжением питания, энергопотреблением, уровнями «1» и «0», быстродействием и нагрузочной способностью. Эти параметры, а также некоторые особенности подключения микросхем в схемы зависят от внутреннего устройства микросхем, которое необходимо знать хотя бы в общих чертах.

#### 4.2 Типов цифровых микросхем.

Существует несколько типов цифровых микросхем, различающихся типом используемых электронных элементов (диодов, транзисторов, резисторов), внутренней схемотехникой, электрическими, динамическими, тепловыми, конструктивными и другими параметрами и технологией производства. Основой каждого типа является

базовый элемент и базовый каскад (или каскады), реализующий логические функции. В остальном – происходит комплексирование таких базовых элементов и каскадов для получения схем с нарастающей сложной функциональностью. Наиболее распространенными на сегодня типами схемотехники являются:

1. Транзисторно-транзисторная логика (ТТЛ или TTL) и ее модификация, например «с диодом Шоттки» (ТТЛШ): базовый элемент – биполярный транзистор; базовый каскад – одноктактный или двухтактный транзисторный усилитель в ключевом режиме.
2. Логика на комплементарных полевых транзисторах (КМОП или CMOS): базовый элемент – полевой транзистор с изолированным затвором, базовый каскад – двухтактный ключ.
3. Эмиттерно-связанная логика (ЭСЛ или ECL): базовый элемент – биполярный транзистор, базовый каскад – дифференциальный усилитель.

Также существуют другие типы цифровых микросхем, менее популярные и/или устаревшие:

4. Диодная логика (ДЛ) и диодно-транзисторная логика (ДТЛ): базовый элемент – полупроводниковый диод и биполярный транзистор, базовый каскад – диодная матрица и транзисторный усилитель. Данная схемотехника является предшественником ТТЛ
5. Логика на полевых транзисторах (nМОП-логика или pМОП-логика): базовый элемент – полевой транзистор с изолированным затвором, базовый каскад – одноктактный ключ. Данная схемотехника – предшественник КМОП.

#### 4.3 Базовые элементы и каскады цифровых КМОП – микросхем.

На сегодня самой распространенной для цифровых микросхем является КМОП-схемотехника. Большинство современных БИС и СБИС для компьютерной техники, в том числе микропроцессоры, микросхемы памяти и т.п., имеют КМОП-схемотехнику или ее модификации.

Кроме того, работа КМОП-каскадов является наиболее простой и доступной для понимания, по сравнению с другими структурами, например, ТТЛ. Поэтому на ее примере легко понять базовые принципы функционирования цифровых микросхем, во многом схожие и для других схемотехнических структур.

Базовым элементом КМОП-структуры является полевой транзистор с изолированным затвором по принципу МЕТАЛЛ-ОКИСЕЛ-ПОЛУПРОВОДНИК (МОП или MOSFET/MOS).

МОП-транзистор является сопротивлением, управляемым напряжением. Вывод управления называется «затвор» / GATE; выводы «управляемого сопротивления» – «исток» / SOURCE и «сток» / DRAIN. При использовании его в ключевом режиме – ключом, имеющим два состояния: «выключен» (сопротивление больше 1 МОм) и «включен» (сопротивление меньше 10 Ом).

Входное сопротивление (между «затвором» и «стоком»/ «истоком») у МОП-транзисторов очень высокое (несколько МОм) и, соответственно, входной ток (ток «затвора») очень маленький (единицы микроампер и меньше). В связи с этим МОП-логика очень энергоэкономична, что является одним из ее преимуществ по сравнению с ТТЛ, особенно для микросхем с миллионами транзисторов.

В цифровой технике используют два типа транзисторов, отличающихся типом полупроводника (канала): с n-каналом (nМОП) и с p-каналом (pМОП). Они отличаются полярностью напряжения, подаваемого на выводы затвора и истока (полярность определяется относительно вывода «сток»): у nМОП она положительная, у pМОП – отрицательная.

МОП-транзисторы бывают с обедненным каналом («нормальнозамкнутые») и с обогащенным каналом (нормально разомкнутые). В цифровых микросхемах обычно используются последние.

Условные обозначения МОП-транзисторов показаны на рисунке. (Иногда, для простоты изображения, данные символы заменяют на более простые в начертании символы МОП-транзисторов с обедненным каналом: вместо прерывистого канала рисуется непрерывная черта. Для удобства также можно рМОП-транзистор обозначать со знаком инверсии (кружочек) на входе («затворе»), что указывает на то, что он открывается отрицательным напряжением.)



Рисунок 4 Условные обозначения МОП-транзисторов

В ключевом режиме МОП транзистор работает следующим образом:

1. Если на «затвор» (относительно «истока») подано напряжение (в микросхемах это напряжение ВЫСОКОГО уровня; полярность напряжения зависит от типа транзистора – nМОП или pМОП), то сопротивление канала между выводами «сток» и «исток» становится низким и по нему протекает ток (направление тока зависит от типа транзистора – nМОП или pМОП).
2. Если на «затвор» (относительно «истока») подано напряжение близкое к нулю (или полярности, противоположной для данного типа транзистора), то сопротивление канала между выводами «сток» и «исток» становится высоким и по нему не протекает ток.

Таким образом, эквивалент МОП-транзистора в ключевом режиме – обычный ключ (включатель).

#### 4.4 Логические элементы на МОП-транзисторах.

Комбинируя управляемые ключи на основе МОП-транзисторов можно реализовать все элементарные булевы (логические) функции, а также более сложные логические функции. Далее рассмотрены такие схемы для *положительной* КМОП-логики. Напряжение питания для примера выбрано +5В и соответствующие ему значения напряжений будут иметь логические сигналы ВЫСОКОГО и НИЗКОГО уровней.

##### 4.4.1 КМОП-инвертер.

Инвертер является простейшим логическим элементом. Схема инвертера включает только один каскад на 2-х комплементарных МОП-транзисторах: pМОП + nМОП. Такой каскад является базовым для КМОП-схемотехники и используется в составе схем многих других логических элементов. Условные обозначения инвертера по европейским и американским стандартам, а также схема инвертера приведены на рисунке ниже. Работает инвертер следующим образом:

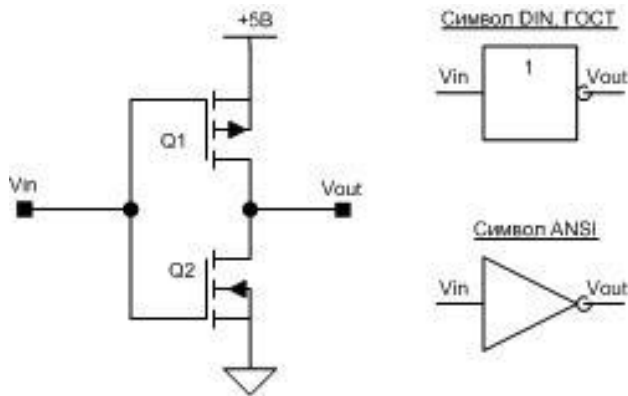


Рисунок 5 Обозначения и схема (КМОП) инвертора.

Если на вход  $V_{in}$  подано напряжение ВЫСОКОГО уровня – логическая «1» (например, 5В), то транзистор Q1 типа pМОП будет закрыт (напряжение затвор-исток будет  $(5В - 5В) = 0В$ , а для открытия pМОП нужно  $-1.5В$ ), а транзистор Q2 типа nМОП будет открыт (напряжение затвор исток будет равно  $(5В - 0В) = 5В$ ).  $V_{out}$  соединится с шиной 0В (минус источника питания) и на выходе будет 0В, то есть НИЗКИЙ логический «0». Таким образом произойдет инверсия уровня «1» в «0».

Если на вход  $V_{in}$  подать НИЗКИЙ уровень – логический «0», то закроется Q2 и откроется Q1. Напряжение питания +5В попадет через Q1 на выход  $V_{out}$  и там появится ВЫСОКИЙ уровень напряжения – логическая «1». То есть тоже произойдет инверсия входного сигнала.

Для удобства демонстрации работы логического элемента – инвертера – покажем его эквивалентную схему на двух механических ключах, выполняющих функцию транзисторных ключей и управляемых логическим сигналом: нижний замыкается при значении управляющего сигнала «1», а верхний – при значении «0». На рисунке показано состояние ключей для  $V_{in} = \langle 1 \rangle$  (слева) и  $V_{in} = \langle 0 \rangle$  (справа). Видно, что выход соединяется через ключ либо с шиной питания 0В, либо с шиной питания +5В.

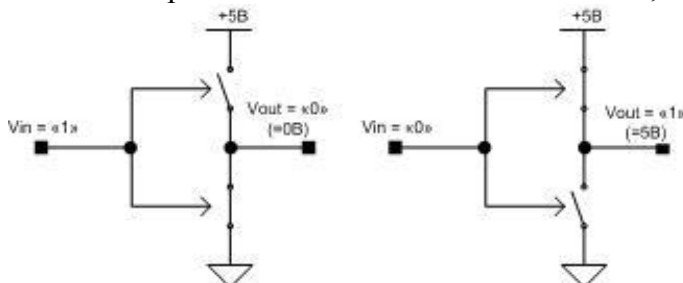


Рисунок 6 Эквивалентная схема КМОП-инвертора на механических ключах.

#### 4.4.2 КМОП-элемент И-НЕ (NAND).

Условные обозначения элемента И-НЕ, принципиальная схема КМОП-элемента И-НЕ, таблица истинности, показывающая все состояния элемента И-НЕ и эквивалентные схемы на ключах для каждого из этих состояний показаны на рисунке.

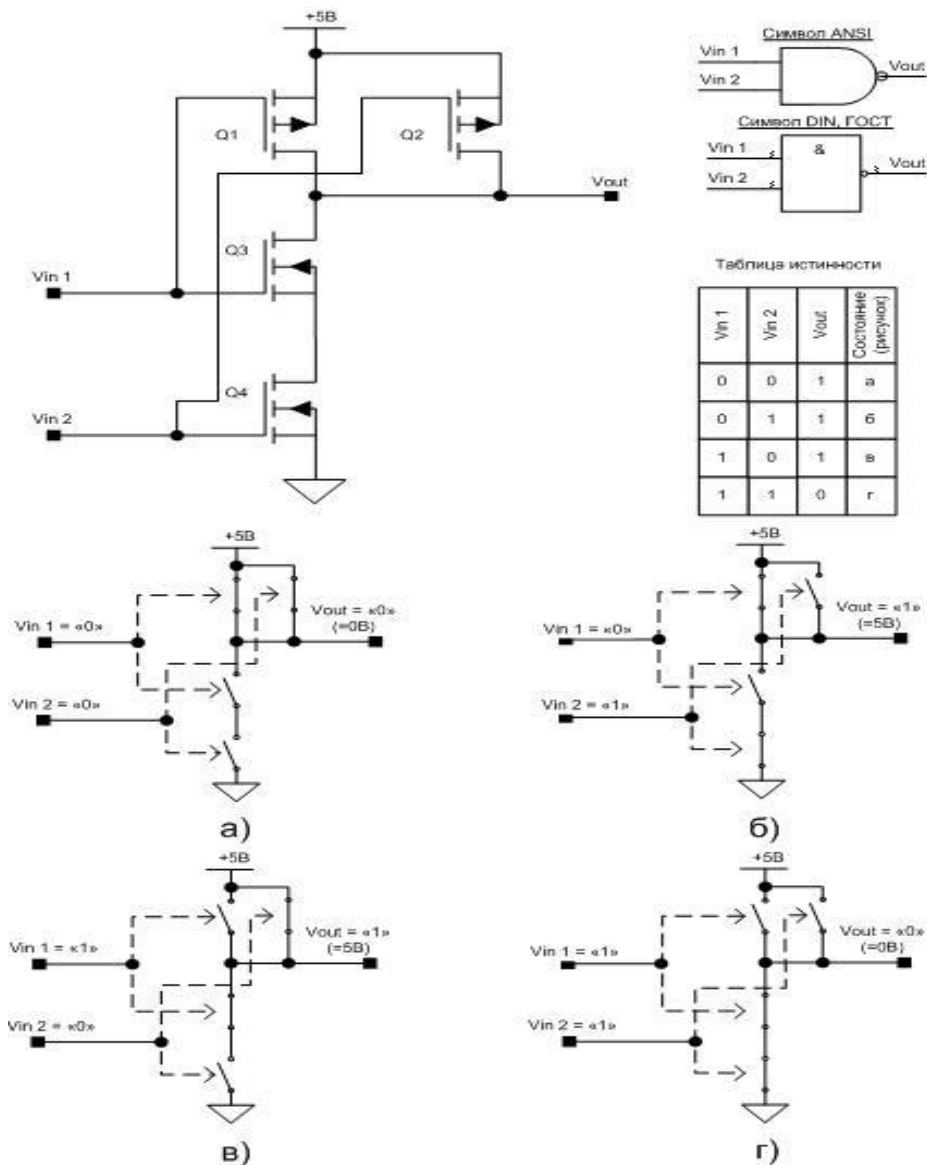


Рисунок 7 Обозначение и схема (КМОП) элемента И-НЕ.

Схема работает следующим образом: если хотя бы на одном из входов Vin1 или Vin2 появляется уровень логического «0» (НИЗКИЙ), то закрывается (размыкается) один из транзисторов Q3 или Q4 (или закрываются оба сразу, если на обоих входах установлен уровень «0»). Одновременно с этим открывается (замыкается) один из или оба транзистора Q1 и Q2. На выход попадает напряжение питания +5В, то есть ВЫСОКИЙ уровень – логическая «1». Единственное состояние схемы, когда на выходе НИЗКИЙ уровень – логический «0», это при наличии логических «1» на обоих входах Vin1 и Vin2. Оба транзистора Q1 и Q2 закрыты (разомкнуты), а транзисторы Q3 и Q4 – открыты (замкнуты), и выход Vout подключен через них к шине питания 0В. Таблица истинности элемента И-НЕ, приведенная на рисунке, соответствует работе данной схемы.

Многовходовые элементы И-НЕ строятся аналогичным образом, но с добавлением дополнительных nМОП-транзисторов, последовательно включенных с Q3, Q4, и pМОП транзисторов, параллельно включенных с Q1, Q2.

#### 4.4.3 КМОП-элемент ИЛИ-НЕ (NOR).

Схема элементов ИЛИ-НЕ построена зеркально относительно схемы И-НЕ: nМОП-транзисторы «нижнего» плеча включены параллельно, а pМОП-транзисторы «верхнего» плеча – последовательно.

Условные обозначения элемента ИЛИ-НЕ, принципиальная схема КМОП-элемента ИЛИ-НЕ, таблица истинности, показывающая все состояния элемента ИЛИ-НЕ и эквивалентные схемы на ключах для каждого из этих состояний показаны на рисунке.

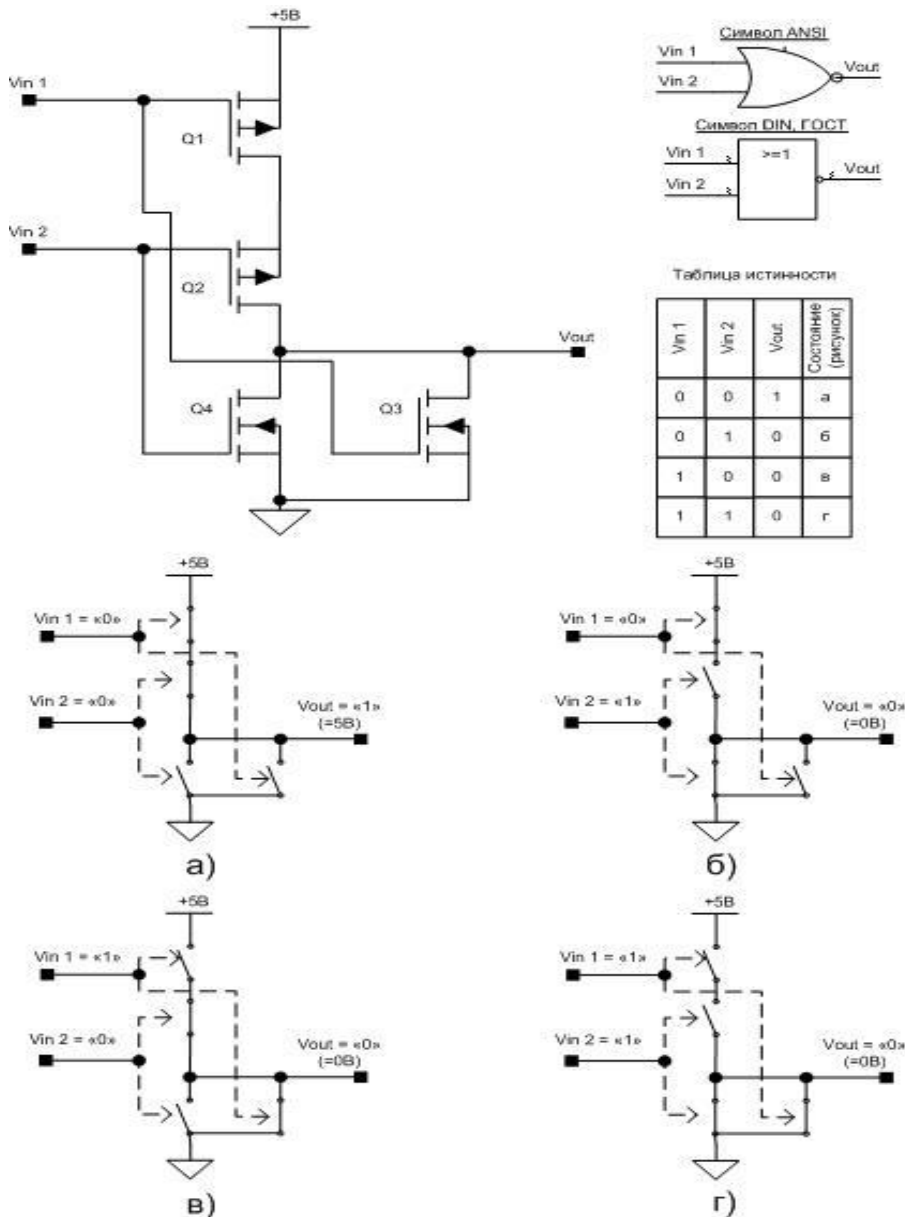


Рисунок 8 Обозначение и схема (КМОП) элемента ИЛИ-НЕ

Принцип работы данной схемы показан в виде состояний эквивалентной схемы на ключах. Единственное состояние, когда на выходе устанавливается ВЫСОКИЙ уровень напряжения – логическая «1», это подача на оба входа Vin 1 и Vin2 уровней «0».

Многовходовые схемы ИЛИ-НЕ строятся аналогично многовходовым схемам И-НЕ, описанным в предыдущем параграфе.

#### 4.4.4 Неинвертирующие КМОП-элементы.

Инвертер, элементы И-НЕ и ИЛИ-НЕ являются простейшими – базовыми – элементами КМОП-схемотехники, требующими минимальное число транзисторов для реализации. Если же появляется необходимость реализовать элементы без инверсии – повторители, И, ИЛИ, то их получают путем добавления инвертора на выходе базового элемента. Например, И-НЕ + НЕ = И; ИЛИ-НЕ + НЕ = ИЛИ; НЕ + НЕ = повторитель. Схема элемента ИЛИ приведена на рисунке, остальные строятся аналогичным образом из схем И-НЕ, НЕ.



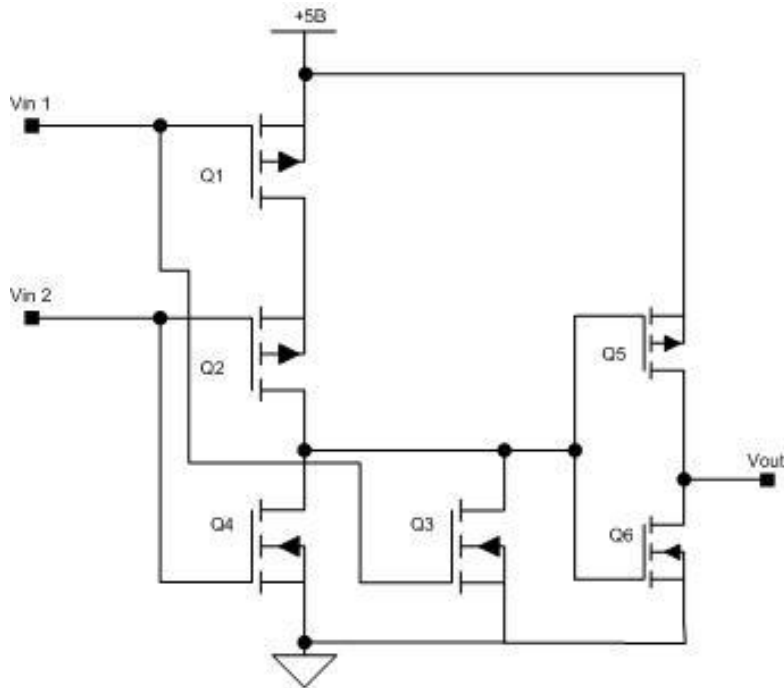


Рисунок 9 Схема КМОП-элемента ИЛИ

Так как схемы без инверсии имеют два последовательных каскада, то задержка распространения сигнала (propagation delay) у них будет больше. Аналогичное правило будет действовать для любых цифровых схем с несколькими последовательными каскадами.

#### 4.5 Типы входных выходных каскадов цифровых схем.

Цифровые элементы и схемы (в смысле электронные схемы, состоящие из множества элементов) могут иметь входы и выходы различных типов, обладающих различными электрическими параметрами. Некоторые типы входов/выходов реализуемы для обеих базовых технологий - КМОП и ТТЛ, а некоторые специфичны для одной из них.

##### 4.5.1 Управляемый ключ или передаточный вентиль (Transmission Gate, T-Gate).

Управляемый ключ – элемент, специфичный для КМОП-технологии, выполняющий функцию, аналогичную механическому ключу: цифровой сигнал передается с входа на выход в замкнутом состоянии ключа, и не передается в разомкнутом состоянии ключа, т.е. выход остается электрически не соединенным со входом. Состояние ключа определяется уровнем сигнала на специальном управляющем цифровом входе.

Управляемый ключ имеет специфическую схемотехнику входов и выходов, обеспечивающую минимальную задержку распространения сигнала - на уровне десятых долей наносекунд, позволяющую передавать сигналы в обе стороны, а также позволяющую передавать не только цифровые, но и аналоговые сигналы.

Схема управляемого ключа представлена на рисунке.

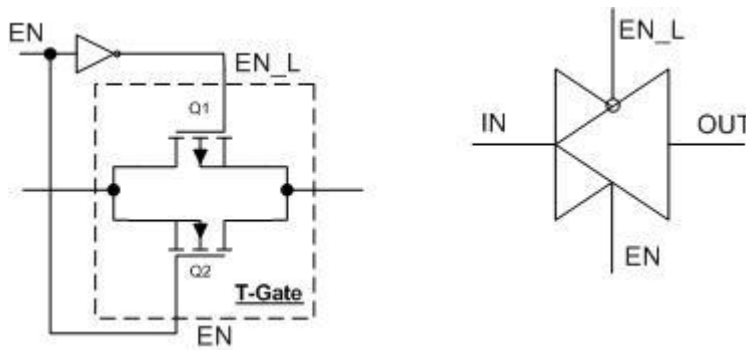


Рисунок 10 Управляемый ключ (передаточный вентиль, transmission gate)

Она включает два комплементарных транзистора: pМОП (Q1) и nМОП (Q2). Их истоки и стоки объединены, а на затворы подаются управляющие сигналы противоположной полярности (инверсия управляющего сигнала на затворе Q1 выполняется с помощью дополнительного элемента – инвертора, изображенного в левом верхнем углу рисунка). Если сигнал EN = ВЫСОКИЙ, а EN\_L = НИЗКИЙ, то оба транзистора находятся в открытом состоянии и сопротивление между входом IN и выходом OUT равно единицы или десятые доли Ома, т.е. вход замкнут на выход. Если EN = НИЗКИЙ, а EN\_L = ВЫСОКИЙ, то транзисторы закрыты и выход OUT переведен в так называемое «высокоомное состояние (Z-состояние)», т.е. электрически отключен от входа IN. Использование только одного транзистора недопустимо, так как каждый из них пропускает ток только в своем диапазоне входных напряжений: Q1 – ВЫСОКИЙ уровень (Уисток – Узатвор = -Увысокий), Q2 – НИЗКИЙ уровень (Уисток – Узатвор = Увысокий).

За счет того, что в открытом состоянии на пути прохождения сигнала через T-Gate нет переключающихся элементов - транзисторов, то задержка распространения сигнала будет минимальной, на уровне долей наносекунд.

В цифровой электронике передаточный вентиль используется в качестве выходного каскада для подключения цифровых элементов к шинам с несколькими источниками, а также в качестве цифровых коммутаторов – мультиплексоров.

Уникальной особенностью T-Gate является возможность передавать сигнал как с IN на OUT, так и в обратном направлении, а также способность пропускать аналоговый сигнал, аналогично обычному механическому ключу. Поэтому на базе T-Gate строят не только цифровые ключи, но и электронные коммутаторы аналоговых сигналов. Также по аналогичной схеме построены мощные электронные ключи и твердотельные реле, используемые, например, для включения и отключения электродвигателей и других электрических нагрузок.

#### 4.5.2 Вход с триггером Шмитта.

Как говорилось ранее (см.п.3.1), среди параметров цифрового входа имеется пороговый уровень напряжения  $U_{вх.п}$  ( $U_{in.t}$ , threshold voltage). Уровень ниже  $U_{вх.п}$  воспринимается как НИЗКИЙ, уровень выше  $U_{п}$  – как ВЫСОКИЙ. Если смена уровня входного сигнала происходит достаточно медленно (пологий, длительный фронт), то, из-за высокочастотных помех, накладывающихся на основной сигнал (см. Рисунок 11-а), может происходить многократное «переключение» входного сигнала между «1» и «0» и соответственное изменение выходного сигнала, как, например, для элемента-повторителя, показано на диаграмме Рисунок 11-б.

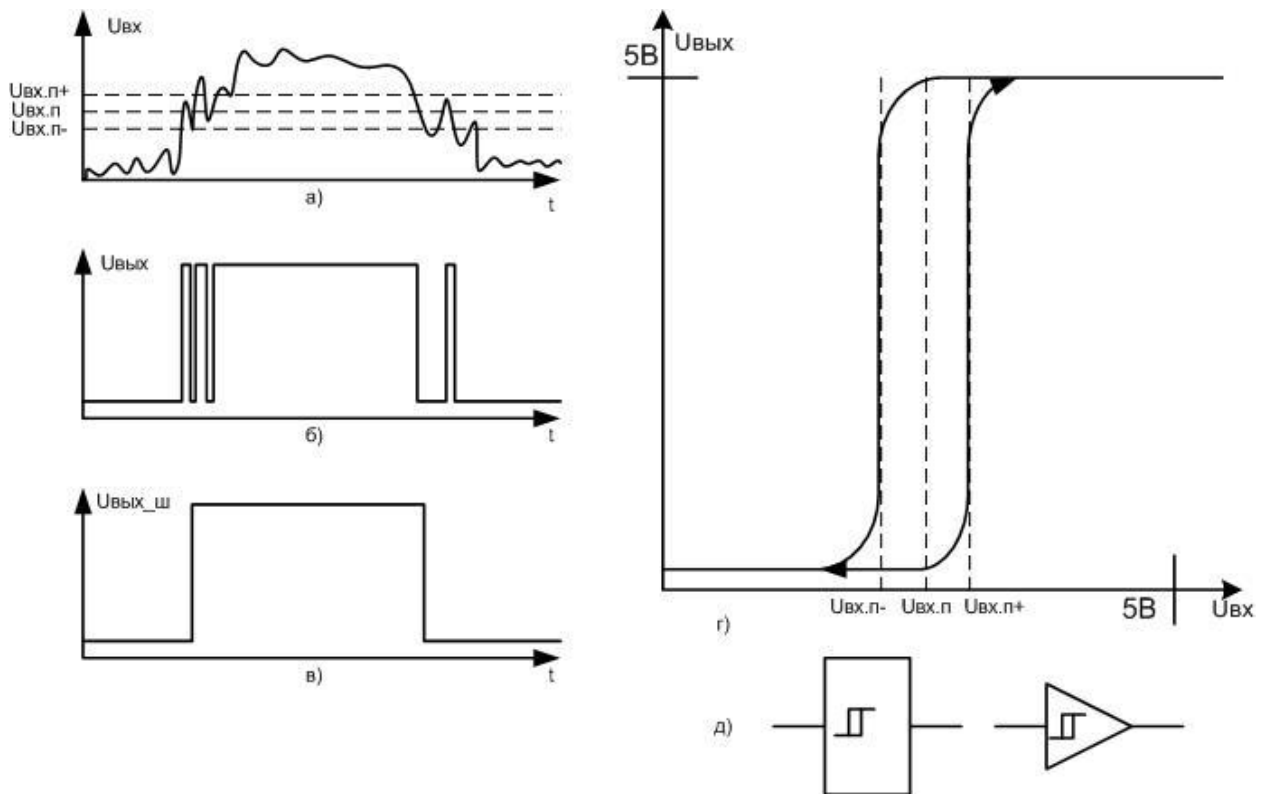


Рисунок 11 Вход с триггером Шмитта

Предотвратить такую ситуацию удастся с помощью положительной обратной связи, добавленной в схему входного каскада. Тогда, если на входе НИЗКИЙ уровень пороговый уровень смещается «вверх», до значения  $U_{вх.п+}$ , а если на входе ВЫСОКИЙ уровень - то вниз, до значения  $U_{вх.п-}$ . В этом случае один раз превысив порог  $U_{вх.п+}$ , сигнал должен опуститься ниже  $U_{вх.п-}$ , чтобы быть считанным как НИЗКИЙ уровень, а это не могут сделать помехи с малым размахом напряжения. То есть повышается помехоустойчивость входа. В случае отрицательного фронта ситуация аналогичная. Пример сигнала после триггера Шмитта показан на диаграмме Рисунок 11-в.

Вход такого типа называется входом с триггером Шмитта. Передаточная характеристика входа с триггером Шмитта показана на Рисунок 11-г, а условное обозначение элемента с соответствующим входом – на Рисунок 11-д. Основной характеристикой триггера Шмитта является напряжение гистерезиса (или просто гистерезис)  $U_{вх.г} = U_{вх.п+} - U_{вх.п-}$ . Чем больше гистерезис, тем выше помехоустойчивость такого входа. Однако чрезмерно увеличивать гистерезис так, чтобы он захватывал диапазоны ВЫСОКОГО или НИЗКОГО уровней нельзя. Типовые значения для логических микросхем с питанием +5В:  $U_{вх.п-} = 2.1В$ ,  $U_{вх.п+} = 2.9В$ .

Входы с триггером Шмитта широко применяются в цифровой технике: их имеют большинство современных микропроцессоров, периферийных контроллеров, памяти, ПЛИС и других СБИС.

#### 4.5.3 Логический выход двухтактный, с активной нагрузкой.

Логический выход – выход цифровой схемы, вырабатывающий цифровые сигналы с двумя уровнями - ВЫСОКИМ и НИЗКИМ, соответствующими значениям логических «0» и «1». Параметры цифрового сигнала - стандартные для данной технологии: КМОП, TTL или другой. Можно назвать логические выходы «обычными» или «стандартными» цифровыми выходами.

Наиболее распространенным типом логических выходов являются двухтактные каскады на комплементарных транзисторах, иначе называемые выходами с активной нагрузкой. Схема и функционирование таких логических выходов для технологии КМОП

рассмотрены в п.4.4.1 (для технологии ТТЛ(Ш) принципы работы аналогичные). Логические выходы построенные по двухтактной схеме за счет этого обеспечивают низкое выходное сопротивление и, соответственно, значительные выходные токи для состояний логических «0» и «1». Большие выходные токи обеспечиваемые открытыми выходными транзисторами обеспечивают быстрый перезаряд емкостей цифровых входов, подключенных к выходу, то есть высокое быстродействие схемы.

Логические выходы нельзя соединять параллельно друг другу: если на одном будет ВЫСОКИЙ уровень, а на другом НИЗКИЙ, то они вступят в противоборство, выходное напряжение будет иметь неопределенное значение, а выходные токи будут значительными (выходы через свои транзисторы фактически замкнут источник питания), что может привести к перегреву выходных транзисторов и выходу их из строя.

Однако короткие импульсы тока все же останутся в моменты переключения состояния выхода (например, из ВЫСОКОГО в НИЗКОЕ состояние), когда транзистор одного (в примере - верхнего) плеча еще не закрылся, а другого (в примере - нижнего) уже открылся. Это паразитные токи, повышающие энергопотребление и нагрев элементов, а также служащие источником электромагнитных помех. Для борьбы с ними в современной схемотехнике используют различные приемы.

#### 4.5.4 Логический выход одноклапный, с резистивной нагрузкой.

Одноклапный выход построен на одном транзисторе (МОП или биполярном), включенным по схеме с общим эмиттером (истоком) и имеющим резистивную нагрузку (*Примечание: иногда резистивную нагрузку называют пассивной, сравнивая с активной нагрузкой в каскадах КМОП-схем, но это не совсем корректно: с точки зрения электроники резистор имеет активное сопротивление, а конденсатор и катушка индуктивности - реактивное*). Схема инвертора с одноклапным выходным каскадом приведена на рисунке. Специального условного обозначения такой тип выхода не имеет.

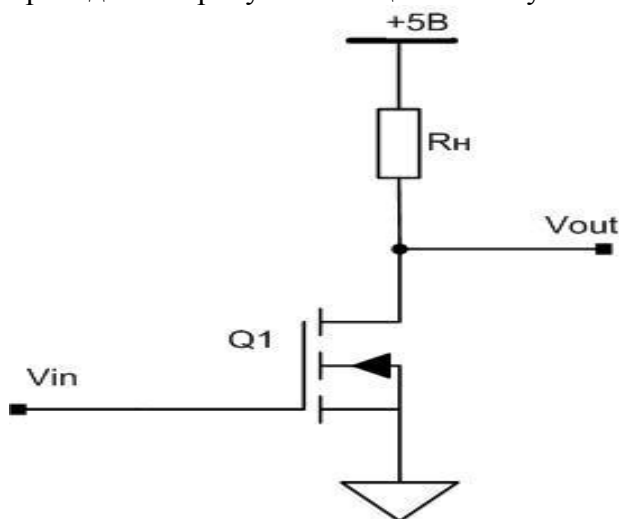


Рисунок 12 Одноклапный выход с резистивной нагрузкой

Когда на входе  $V_{in}$  установлен ВЫСОКИЙ уровень напряжения, то транзистор открывается и соединяет выход с общим проводом.  $V_{out} = 0V$ , т.е. на выходе установлен НИЗКИЙ уровень. Выходное сопротивление низкое, как и у КМОП-каскадов и равно сопротивлению открытого транзистора, т.е. единицы или десятые доли Ома. Нагрузочная способность также равна нагрузочной способности КМОП-каскадов. Нагрузочный резистор оказывается включенным между шинами питания  $0V$  и  $+5V$ , через него протекает ток, но, в силу достаточно высокого сопротивления в диапазоне  $5 - 50 \text{ КОм}$ , ток этот невелик ( $1\text{мА} - 100 \text{ мкА}$ ) и выделяемая на резисторе мощность не превышает  $1\text{мВт}$  - он не перегревается и не перегорает.

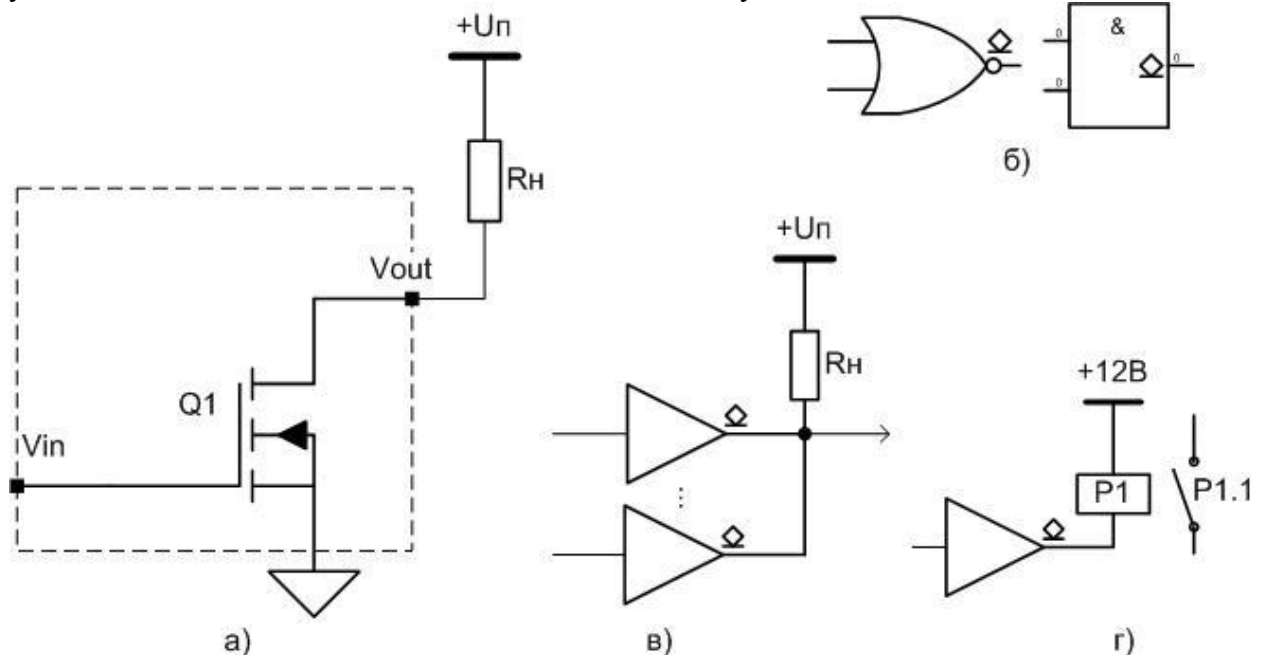
Если на вход подано напряжение низкого уровня, то транзистор закрывается и напряжение  $+5V$  с шины питания через резистор  $R_n$  попадает на выход схемы, т.е.

устанавливается **ВЫСОКИЙ** уровень выходного напряжения. В данном состоянии выходное сопротивление высокое, до нескольких десятков КОм – оно равно сопротивлению нагрузочного резистора  $R_H$ . Такой выход не может питать большое число входов и не может быстро перезаряжать входные емкости подключенных к нему цифровых входов.

Именно малая нагрузочная способность (коэффициент разветвления по выходу) и невысокое быстродействие при перезарядке емкостной нагрузки (входов) являются основными недостатками одноктактных выходов. В настоящее время одноктактная схемотехника используется крайне редко, в основном в давно выпускающихся микросхемах или для поддержки определенных стандартов на типы выходов. Например, однокристальная микроЭВМ i8051 (Intel), разработанная в 80-х годах и использующая такие выходные схемы, стала стандартом де-факто, ее аналоги выпускаются многими фирмами до сих пор и также имеют одноктактные выходы - для совместимости.

#### 4.5.5 Выход с открытым коллектором (ОК) или открытым стоком (ОС).

Схемотехника и функционирование выхода ОК (для КМОП-схем правильно говорить об выходе с ОС, но часто данный тип выхода по аналогии с ТТЛ-схемами называют ОК) аналогична одноктактному выходу, рассмотренному в предыдущем параграфе, с той разницей, что внутренний (на кристалле микросхемы) нагрузочный резистор отсутствует, а подключается внешний резистор. Схема представлена на рисунке (см. Рисунок 13-а), условное обозначение выхода данного типа – см. Рисунок 13-б).



**Рисунок 13 Выход с «открытым коллектором» (открытым стоком)**

Особенностями выхода ОК являются:

- 1) Возможность подключать несколько выходов к одной шине (Рисунок 13-в). Эта схема включения может быть необходима для выполнения одной из двух функций:
  - 1) Реализация «монтажной логики». В частности операции «монтажное И» для положительной логики: если хотя бы на одном из выходов будет установлен **НИЗКИЙ** уровень, это значит, что общая выходная шина будет замкнута на 0В через открытый транзистор этого выхода и, вне зависимости от уровней на остальных выходах, на общей шине будет установлен **НИЗКИЙ** уровень, то есть логический «0». Таким образом будет реализована операция И без какого-либо специального элемента. Если использовать отрицательную логику, т.е. **ИСТИНОЙ** (логической «1») считать **НИЗКИЙ** уровень, то данная схема реализует функцию

монтажного ИЛИ. Именно в такой интерпретации чаще всего используется данная схема включения. Например, сигналы прерывания от нескольких источников с выходом ОК объединены и подключены к одному входу запроса прерывания у микропроцессора. Если хоть один из них перейдет в активное НИЗКОЕ состояние, то выработается запрос прерывания и далее процессор путем опроса найдет источник этого прерывания. Другой пример – сканирование матричной клавиатуры.

- 2) Подключение нескольких источников к общей шине. В данном случае активным является только один источник, а все неактивные обязаны установить на выходе ВЫСОКИЙ уровень. Далее, так как реализуется монтажное И (см. выше), то на шине присутствует только сигнал от одного – активного – источника. При этом, даже если одновременно активизируются два или более источников, выход не перегрузится и не «перегорит». Функция общей шины широко используется для периферийных интерфейсов микропроцессорных систем, например, в интерфейсах I<sup>2</sup>C, 1-WireBus.
- 2) Возможность подключать нагрузку к напряжению большему или меньшему, чем напряжение питания микросхемы. Например, на рисунке (Рисунок 13-г) показан способ подключения обмотки реле с напряжением питания 12В. Аналогично можно подключать другие нагрузки: лампочки, линейки светодиодов, звукоизлучатели, двигатели постоянного тока и т.п. Использовать как большее, так и меньшее напряжение питания также может быть удобно для адаптеров логических уровней схем с различным напряжением питания. Например, при конверсии сигналов с ВЫСОКИМ уровнем равным 5В в сигналы с ВЫСОКИМ уровнем равным 3.3В, и наоборот.

#### 4.5.6 Логические выходы с тремя состояниями (с высокоомным Z-состоянием).

Выход с тремя состояниями (3-State Output) предназначен для подключения нескольких выходов к общей шине. Использование для этого схемы с ОК, рассмотренной выше, ограничено для быстродействующих систем, например, для шин адреса и данных в компьютерах, из-за высокого выходного сопротивления при ВЫСОКОМ уровне выходного сигнала. В связи с этим был разработан выход с тремя состояниями.

Выход с тремя состояниями в активном режиме работает как обычный двухтактный логический выход и может вырабатывать два уровня напряжения – «ВЫСОКИЙ» и «НИЗКИЙ». Однако, при необходимости, можно отключать выход подобно схеме управляемого ключа (переводить в состояние с высоким выходным сопротивлением – высокоомное или Z-состояние или «третье»-состояние). Для отключения/подключения выхода используется специальный управляющий вход – вход разрешения (выхода). Если выход схемы переведен в третье состояние, то будет оставаться в нем независимо от уровней на входах (кроме входа разрешения) этой схемы и не будет конфликтовать с другими – активными – выходами, подключенными к общей шине.

Схемотехника выхода (см. Рисунок 14-а) с тремя состояниями реализована следующим образом: к затворам выходных транзисторов подключена простейшая логическая схема, которая переводит оба транзистора в закрытое состояние при неактивном уровне на входе ENB (Enable - разрешение). В «разрешенном» состоянии эта логика просто ретранслирует сигнал управления затворами транзисторов.

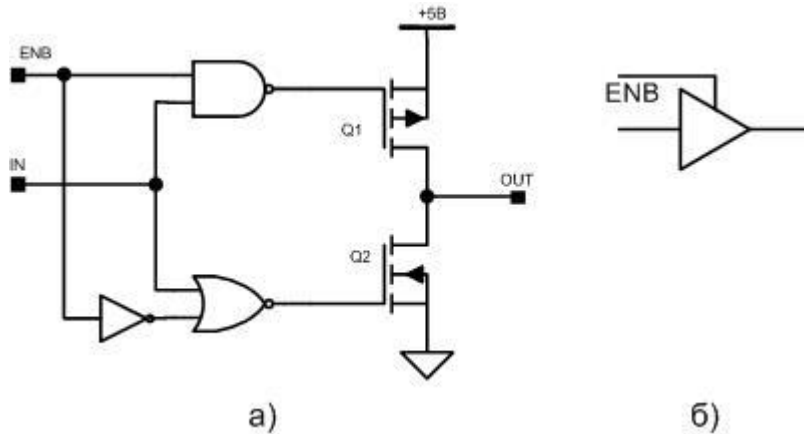


Рисунок 14 Выход с тремя состояниями

В принципе, для перевода схемы в третье состояние можно также использовать управляемый ключ (T-Gate), включенный на выходе схемы последовательно, но это часто неудобно с точки зрения технологии производства микросхем (из-за специфической структуры T-Gate).

Выход с тремя состояниями широко используется для выходов шин адреса и данных, а также для портов ввода-вывода у компонентов микропроцессорных систем. Также выпускаются микросхемы логических элементов, регистров, счетчиков с такими выходами. На рисунке (см. Рисунок 14-б) показано обозначение буфера с выходом с тремя состояниями.

## 5 ЦИФРОВЫЕ ЭЛЕМЕНТЫ И СХЕМЫ.

### 5.1 Комбинационные последовательностные цифровые схемы.

Основная задача любой цифровой схемы – выработка выходных цифровых сигналов в соответствии с входными сигналами. Зависимость между входными и выходными сигналами может быть двух типов:

- 1) Состояние выходов является predetermined функцией состояния входов, то есть текущее значение на выходах схемы зависит только от текущих значений на входах и не зависит от предыстории их изменения. Цифровые схемы такого типа называются схемами с комбинационной логикой - **комбинационными**. Так как комбинационным схемам не требуется хранить и учитывать свое состояние, то они не имеют элементов памяти и обратной связи между выходами и входами. Функция, реализуемая комбинационными схемами, называется булевой функцией и записывается в виде булева уравнения (выражения) или таблицы истинности. Значения на входах схемы являются аргументами функции. На выходе формируется значение функции. Законы составления и преобразования булевых выражений (булева алгебра) и методы синтеза комбинационных схем подробно рассматриваются в курсе «Дискретная математика».
- 2) Состояние выходов зависит от текущего состояния входов, а также от предыстории их изменения. Цифровая схема, реализующая подобную функцию, должна иметь запоминающие элементы (элементы памяти) и обратную связь от этих элементов ко входам схемы. Цифровые схемы, реализующие такую функцию называются схемами с последовательностной логикой – **последовательностными** - или **автоматами с памятью**. Методы описания и синтеза схем автоматов подробно рассматриваются в курсе «Теория цифровых автоматов».

### 5.2 Логические элементы и базовые операционные элементы цифровых схем.

И комбинационные и последовательностные цифровые схемы представляют собой совокупность соединенных между собой **цифровых элементов**. Цифровой элемент -

также электронная схема, реализующая некоторую типовую функцию и имеющая конструктивно законченную реализацию, например, в виде микросхемы или ячейки ПЛИС/БМК. Цифровой элемент – один из уровней иерархии электронного устройства. На уровне цифровой схемы цифровой элемент является функционально и конструктивно неделимым. Ниже цифрового элемента могут располагаться или уровни еще более простых цифровых элементов, или уровень дискретных элементов – транзисторов, резисторов, конденсаторов и т.д. Выше уровня цифровой схемы могут находиться уровни блочных схем (это схемы, состоящие из нескольких блоков), уровень устройства (устройство также состоит из нескольких блоков, конструктивно и функционально завершено).

Цифровые элементы делятся на несколько типов:

- 1) **Логические элементы** или **вентили** - реализуют некоторую элементарную логическую функцию: НЕ/NOT (отрицание), ИЛИ/OR (дизъюнкция), И/AND (конъюнкция), ИЛИ-НЕ/NOR (дизъюнкция с отрицанием), И-НЕ/NAND (конъюнкция с отрицанием), ИСКЛЮЧАЮЩЕЕ ИЛИ/XOR, ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ/XNOR. Вентили имеют два или большее число входов (исключение – вентиль НЕ, который имеет один вход) и один выход. Условные обозначения и таблицы истинности для перечисленных типов вентиляей приведены на рисунке ниже.

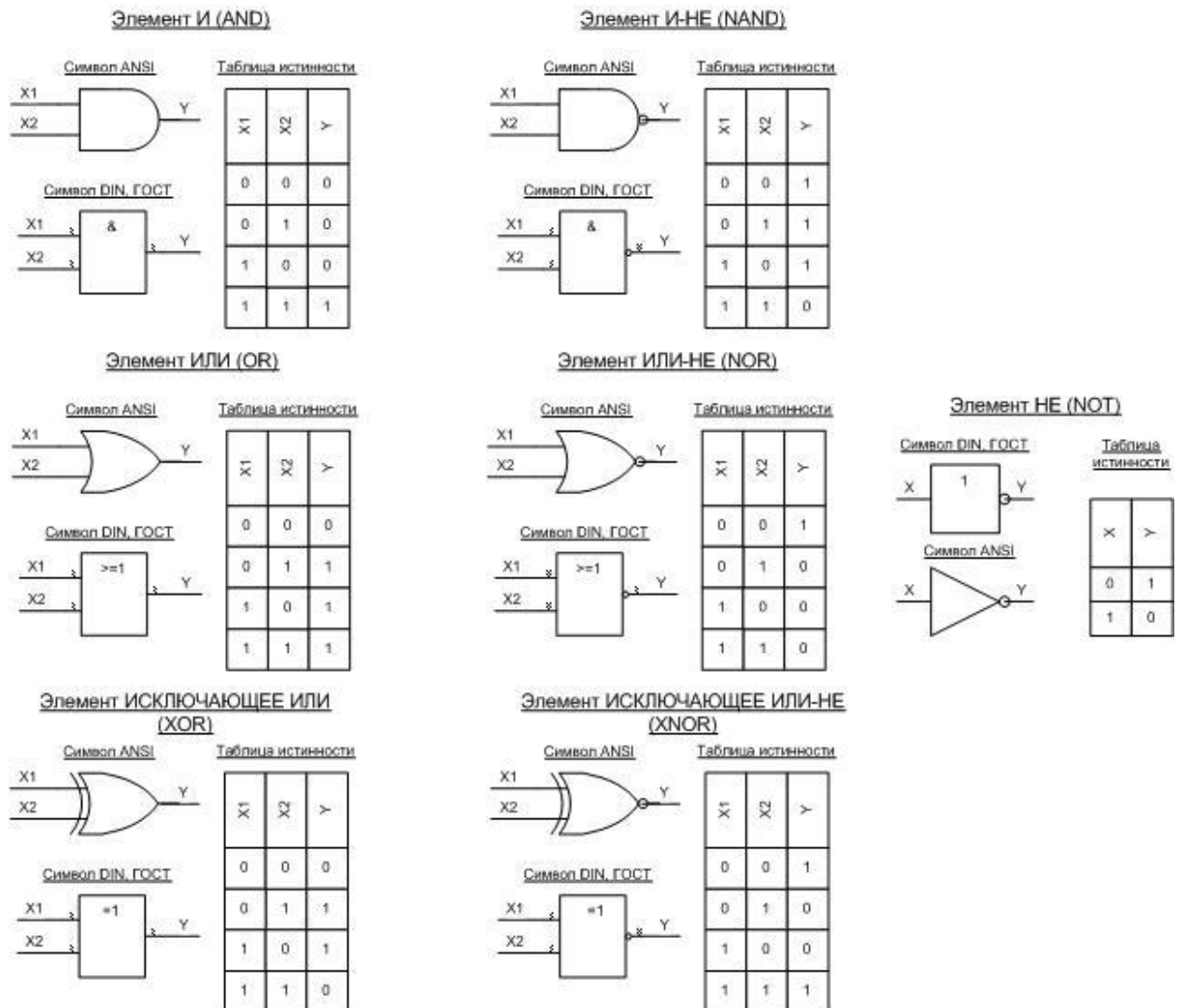


Рисунок 15 Логические элементы (вентили)

- 2) Триггеры – элементы с несколькими устойчивыми состояниями. Наиболее часто используются триггеры с двумя состояниями – бистабильные. Триггер может быть построен как на базе логических элементов, так и на дискретных компонентах.



Триггер – основной тип элемента памяти в последовательных схемах. Типы триггеров и способы их использования описаны ниже в разделе с описанием цифровых схем последовательного типа.

- 3) Базовые операционные элементы (БОЭ) - структурные блоки сложных схем, реализующие некоторые типовые операции обработки цифровых сигналов. БОЭ характеризуются структурной (выпускаются в качестве микросхем или являются ячейками ПЛИС/БМК) и функциональной завершенностью. Так как БОЭ сами являются цифровой схемой, то бывают:
  - а) БОЭ комбинаторного типа (мультиплексоры, шифраторы, дешифраторы, сумматоры, компараторы и т.п.)
  - б) БОЭ последовательного типа (регистры, счетчики, делители частоты, математические умножители/делители и т.п.).

## 6 БАЗОВЫЕ ОПЕРАЦИОННЫЕ ЭЛЕМЕНТЫ КОМБИНАЦИОННОГО ТИПА.

БОЭ комбинационного типа используются для преобразования входных сигналов и кодов в выходные, в том числе для взаимной коммутации входов и выходов БОЭ, для взаимного преобразования различных кодов, для выполнения математических и логических операций с числами, представленными в виде различных кодов.

Ниже рассмотрены наиболее часто применяемые БОЭ.

### 6.1 Мультиплексор (multiplexer, MUX)

Осуществляет коммутацию одного из нескольких цифровых входов на один выход. Номер подключенного входа задается двоичным кодом на специальных адресных входах мультиплексора. Если необходимо коммутировать многоуровневый код, то используют несколько одноразрядных мультиплексоров с параллельно соединенными адресными входами. Одно- или многоуровневые входы и выход называют соответственно входными и выходными каналами. Количество входных каналов  $N$  и количество разрядов адресного входа  $K$  связано соотношением:  $N=2^K$ .

На рисунке (см. Рисунок 16) для одноразрядного четырехвходового мультиплексора показаны условное обозначение на принципиальных электрических схемах (а), на структурно-структурных схемах (б), таблица истинности и вариант внутренней схемы (в).

Мультиплексоры применяются как в виде отдельных микросхем, так и в виде блоков в составе более сложных микросхем.

Возможные применения мультиплексоров в современной цифровой технике:

- а) подключение одного из нескольких функциональных блоков внутри микросхемы (микропроцессора, периферийного контроллера, ПЛИС и т.д.) к разделяемым выходным контактам этой микросхемы.
- б) Подключение к общей шине передачи данных одного из нескольких источников данных. Адрес источника подается на входы управления мультиплексора. Таким образом реализуются порты ввода-вывода для микропроцессоров.
- в) На базе мультиплексоров можно строить универсальные логические модули (УЛМ) – схемы, которые могут быть настроены на реализацию любой логической функции, задаваемой специальными конфигурационными сигналами. В простейшем варианте это реализуется следующим образом: на адресные входы подаются аргументы, а на входы данных – константы, соответствующие правильным результатам. Тогда при подаче некоторой комбинации аргументов к выходу подключится вход с «поданным на него» правильным результатом. Есть и более эффективные УЛМ структуры, подробно описанные в книге Е.П.Угрюмова «Цифровая схемотехника». УЛМ на базе мультиплексоров используются в составе некоторых ПЛИС, например, в ПЛИС фирмы АСТЕЛ.
- г) На базе мультиплексоров строятся комбинационные схемы сдвига (см. описание ниже).

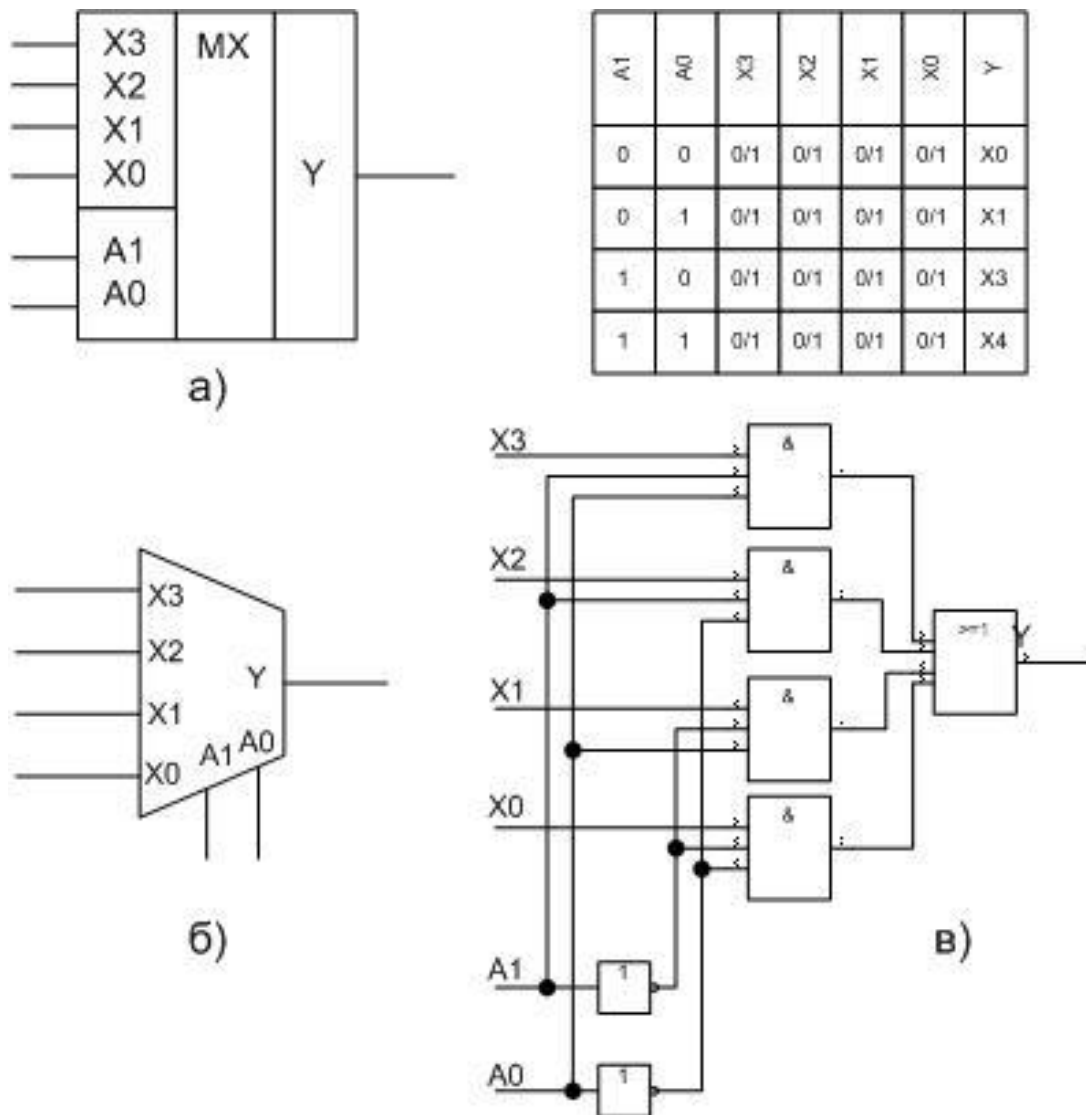


Рисунок 16 Одноразрядный мультиплексор «4 в 1»

## 6.2 Демультимплексор (demultiplexer, DEMUX)

Выполняет функцию обратную мультиплексору: коммутирует один входной канал на несколько выходных. Каналы могут быть одноразрядными или многоразрядными. Функция адресных входов аналогична мультиплексору с той разницей, что выбирается один из выходных каналов. Количество выходных каналов  $N$  и количество разрядов адресного входа  $K$  связано соотношением:  $N=2^K$ .

Так как демультимплексор имеет несколько выходов, то он описывается не одним, а системой логических выражений.

На рисунке (см. Рисунок 17) для одноразрядного четырехвходового демультимплексора показаны условное обозначение на принципиальных электрических схемах (а), на структурно-функциональных схемах (б), таблица истинности и вариант внутренней схемы (в).

Демультимплексоры применяются реже чем мультиплексоры, особенно в виде отдельных микросхем. Возможные варианты:

- подключение выходов одного функционального блока внутри микросхемы (микропроцессора, периферийного контроллера, ПЛИС и т.д.) к одному из выходных контактов микросхемы.
- В сетевых коммутаторах - подключение передатчика к одному из нескольких каналов связи, в зависимости от выбранного маршрута передачи данных по сети.

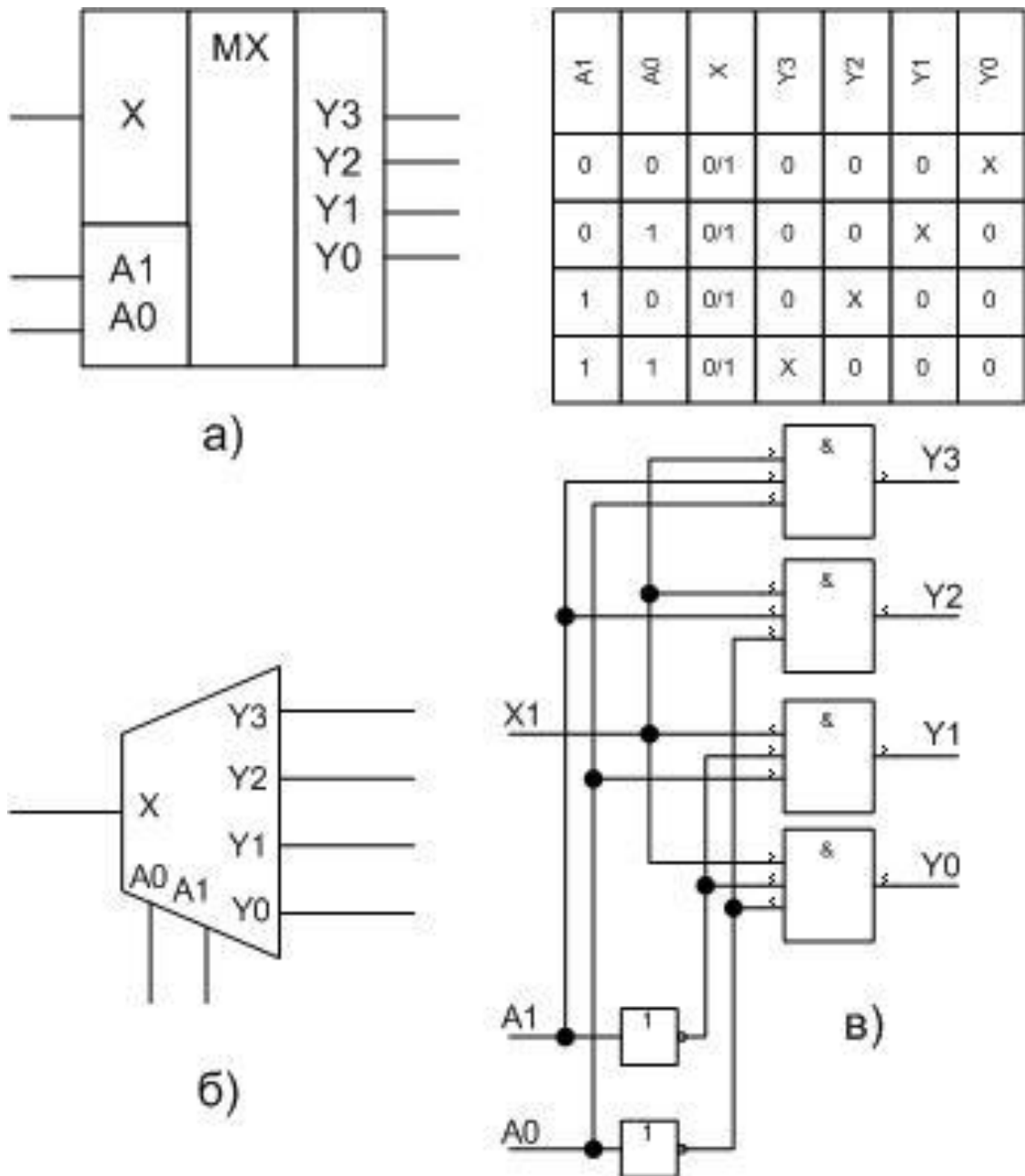


Рисунок 17 Одноразрядный демультиплексор «1 в 4»

### 6.3 Дешифратор (decoder, DC).

Дешифратор – комбинационная схема с несколькими входами и несколькими выходами, которая преобразует входной код в отличный от него выходной код с равным или большим количеством разрядов (цифровых сигналов, битов).

Наиболее распространены:

- двоичный дешифратор «К в  $2^K$ »;
- дешифратор для семисегментного индикатора.

*Двоичный дешифратор* преобразует входной двоичный К-разрядный код в выходной позиционный код «1 из N». В коде «1 из N» в состоянии «1» находится только один разряд, а остальные – в состоянии «0». Номер активного разряда выходного кода «1 из N» определяется двоичным К-разрядным кодом. Если  $N=2^K$ , то каждому варианту входного

кода соответствует свой выходной код и дешифратор называется *полным*. Если  $N < 2^K$ , то часть входных кодов не может быть преобразована и дешифратор называют *неполным*.

Кроме входов преобразуемого двоичного кода у дешифратора обычно есть вход EN (enable) – разрешение выхода. Если на входе EN активный уровень (при положительной полярности входа, активный уровень - «1»), то дешифратор преобразует код, как писано выше. Если на входе EN неактивный уровень (при положительной полярности входа, неактивный уровень - «0»), то все выходы находятся в неактивном состоянии вне зависимости от входного кода.

На рисунке (см. Рисунок 18) показано условное обозначение (а), таблица истинности и внутренняя схема (б) двоичного дешифратора.

Двоичные дешифраторы выпускаются в виде отдельных микросхем, а также используются как функциональные блоки в составе больших интегральных схем – микропроцессоров, ПЛИС и т.д. Типовые применения двоичных дешифраторов:

- а) Адресный селектор (декодер адреса) – при обращении процессора к памяти или устройствам ввода вывода на вход дешифратора подается адрес в двоичном коде, а на выходе формируется активный сигнал, соответствующий этому адресу, который активизирует («выбирает») только одну микросхему памяти или периферийный контроллер из нескольких, подключенных к общей шине данных. Далее именно эта микросхема (чей адрес выставлен на шине адреса) выполняет обмен данными с процессором, а остальные находятся в пассивном состоянии.
- б) Селектор строк матричной клавиатуры. Матричная клавиатура – «решетка» из проводов (строк и столбцов), в каждом пересечении находится контактная пара кнопки. На входе дешифратора в цикле перебираются номера всех строк клавиатурной матрицы. С выхода дешифратора в каждый момент времени только на одну строку подан активный сигнал, на остальные строки - пассивный. Сигналы столбцов «подтянуты» к пассивному уровню через резисторы 1КОм. Если клавиша нажата, то активный сигнал будет только на том столбце, который через нажатую кнопку соединен с активной строкой. Зная номера строки и столбца можно определить нажатую клавишу.

Так как период смены номера строки составляет несколько десятков миллисекунд, то в течение нескольких сотен миллисекунд может быть просканирована вся клавиатурная матрица.

*Дешифратор семисегментного кода* преобразует входной двоичный 4-х разрядный код в выходной восьмиразрядный код для управления типовым семисегментным индикатором (семь сегментов символа + десятичная точка). В зависимости от того, какое значение двоичного кода будет подано на вход, такая шестнадцатиричная цифра загорится на индикаторе.

Хотя типовой семисегментный дешифратор имеет четырехразрядный входной код и восьмиразрядный выходной, но можно построить дешифратор семиразрядного двоичного кода с выводом в два знакоместа, но такого обычно не делают.

Если на вход EN будет подан неактивный уровень, то на всех выходах будет неактивный уровень и все сегменты будут потушены.

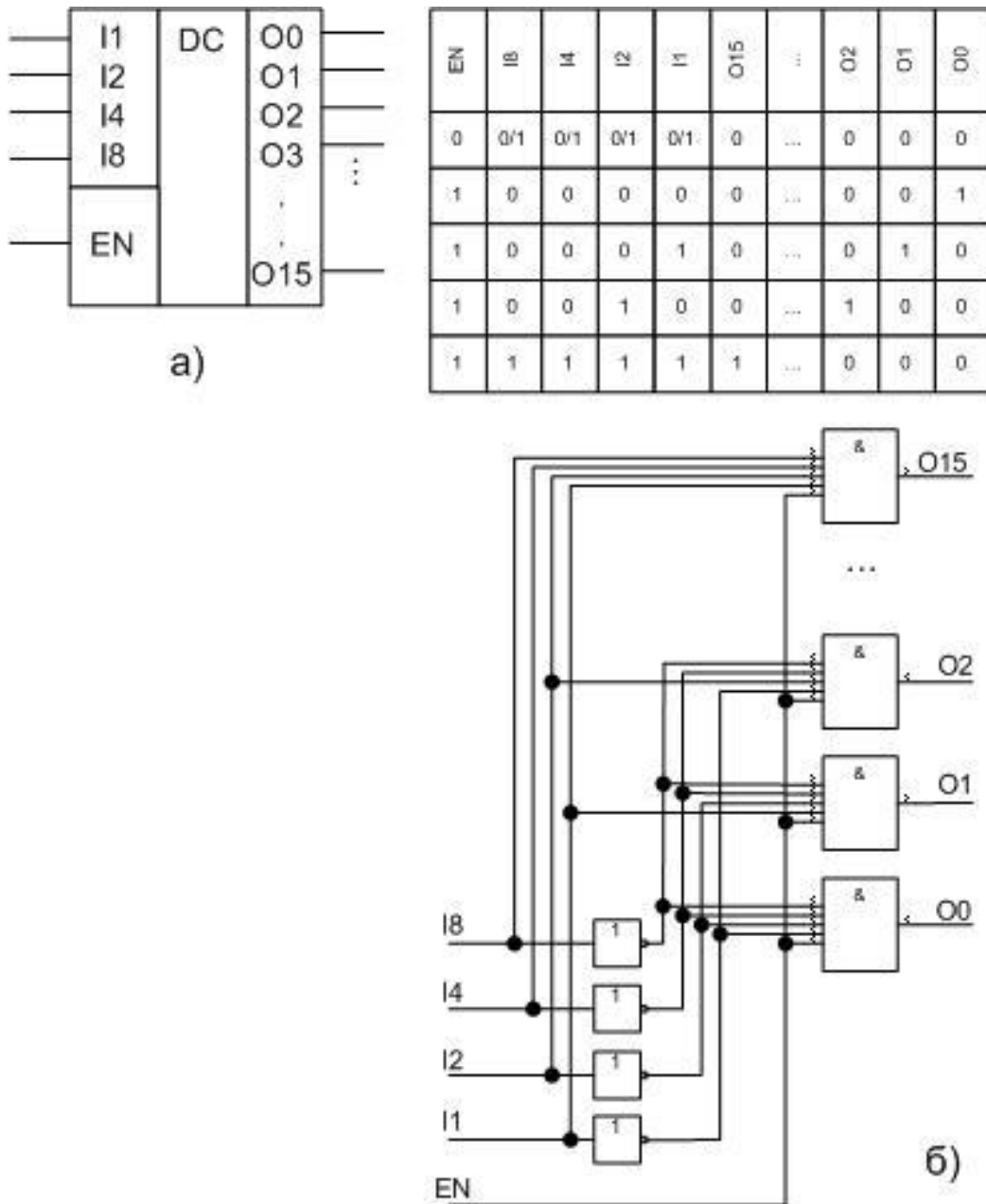


Рисунок 18 Двоичный дешифратор «4 в 16»

#### 6.4 Шифратор (encoder).

Шифратор – комбинационная схема с несколькими входами и несколькими выходами, которая преобразует входной код в отличный от него выходной код с меньшим количеством разрядов (цифровых сигналов, битов).

Наибольшее распространение получили двоичные и приоритетные шифраторы.

*Двоичный шифратор* выполняет функцию обратную двоичному дешифратору - преобразование позиционного кода «1 из N» в двоичный код, соответствующий номеру активного сигнала. При этом на входы может быть подан только один активный сигнал – код «1 из N».

*Приоритетный шифратор* функционально аналогичен двоичному шифратору, но допускает одновременную подачу активных сигналов сразу на несколько входов. Код на выходе приоритетного шифратора будет соответствовать входу с наибольшим порядковым номером, т.е. входу с наивысшим приоритетом.

Двоичный шифратор является частным случаем приоритетного шифратора, поэтому не выпускают отдельные микросхемы двоичных и приоритетных шифраторов – используются только приоритетные шифраторы.

Основное применение шифратора – кодирование номера и приоритетный арбитраж запросов на использование некоторого ресурса. Например, через приоритетный шифратор процессору подаются несколько запросов на прерывание, а на выходе формируется номер самого высокоприоритетного – того, который подключен ко входу шифратора с максимальным номером.

Компаратор (comparator)– схема сравнения, определяют соотношения между двумя числами, представленными в виде кодов, поданных на входы компаратора. Результаты - меньше, равно, больше – также закодированы в виде цифровых сигналов.

Сумматор (adder) – схема сложения, выполняет операцию сложения с числами, представленными в двоичном коде. Формирует признаки переноса между разрядами.

Комбинационные схемы сдвига – сдвигает входной код на N разрядов. Код со значением N подается на специальные управляющие входы. Комбинационные схемы сдвига строятся на базе мультиплексоров.

Схемы свертки - функционально аналогичны шифраторам. Выходной код получается путем выполнения логических операций между битами входного кода. Схемы свертки имеют однобитовый выход (например, схемы формирования бита четности).

Схемы вычисления и проверки специальных кодов (Грея, Хемминга) – это дешифраторы (вычисление кодов) или шифраторы (проверки кодов) со специальными типами вычисляемых кодов. Чаще всего используются для формирования/проверки защитных кодов при передаче или хранении информации.

## 7 ТРИГГЕРЫ.

Триггер – электронная схема, имеющая несколько устойчивых состояний, сохраняющихся длительное время (пока включено электропитание схемы). Соответственно триггеры могут выполнять функцию элементов памяти. В цифровой технике используются бистабильные триггеры с двумя устойчивыми состояниями, которые кодируют значения логических «0» и «1». Триггеры могут быть построены на дискретных элементах (транзисторные каскады) и на логических элементах – цифровые триггеры.

### 7.1 Классификация триггеров.

Триггеры различаются по логике работы и по способу приема и запоминания информации. Ниже представлена классификация триггеров, предложенная в книге Е.Угрюмова «Цифровая схемотехника». В ней выделены наиболее популярные типы триггеров, однако количество их разновидностей гораздо больше.

#### а) По логике работы:

- 1) RS – с отдельными входами записи «1» - «установки» (вход S) и записи «0» - «сброса» (сигнал R).
- 2) D – триггер задержки информационного сигнала (Delay).
- 3) T – тактируемый (Tact) или счетный триггер.
- 4) JK – универсальный триггер.
- 5) комбинированный – например, RST – тактируемый со входами «установки» и «сброса».
- 6) со сложной логикой – с группами входов, связанных логическими зависимостями. Например, JK-триггер с тремя входами J и тремя K, связанными конъюнкцией:  $(J1*J2*J3)$  и  $(K1*K2*K3)$ .

#### б) По способу приема информации:

- 1) Асинхронные (нетактируемые) – переход в новое состояние происходит по изменению состояния управляющих (информационных) входов.
- 2) Синхронные (тактируемые) – переход в новое состояние под воздействием информационных входов происходит по только сигналу на специальном тактовом (синхронизации) входе (C / CLK / CLOCK).

*Примечание: у реальных триггеров наряду с синхронизируемыми информационными входами сохраняют асинхронные информационные входы, чтобы обеспечить асинхронную установку или сброс триггера при реинициализации системы («сброс» или включение питания).*

- Синхронные управляемые уровнем – при одном уровне тактового сигнала триггер воспринимает состояние управляющих входов, а при другом – не воспринимает и остается в одном состоянии, сохраняя записанную в него информацию. Типичный триггер управляемый уровнем – «триггер-защелка» (latch).

- Синхронные управляемые фронтом – При управлении фронтом переходы из состояния в состояние происходят по фронту тактового сигнала.

#### в) По внутренней структуре:

- 1) Одноступенчатые: в их внутренней структуре одна запоминающая схема, которая переключает свое состояние (запоминает данные) под влиянием управляющих сигналов. По одноступенчатой схеме строятся асинхронные триггеры и синхронные управляемые уровнем.
- 2) Двухступенчатые (или многоступенчатые) триггеры состоят из двух одноступенчатых триггеров. При переключении двухступенчатого триггера сначала переключается первая, а только следом – вторая. Это позволяет записывать

данные только по фронту синхроимпульса, не реагируя на изменения на информационных входах в течении синхроимпульса. Т.е. по многоступенчатой схеме строятся синхронные управляемые фронтом триггеры. Это наиболее применяемый сегодня тип триггеров.

## 7.2 Структура триггера.

Триггеры состоят из двух блоков: схемы памяти (фиксатора) и схемы управления.

### 7.2.1 Схема памяти.

Электрически элемент памяти любого триггера это схема с положительной обратной связью. В частности, фиксатор цифрового триггера представляет собой два соединенных «крест-накрест» элемента НЕ (см. Рисунок 19).

Если на выходе одного элемента НЕ установилась «1», то она попадет на вход второго элемента НЕ и на его выходе будет «0», который, в свою очередь попадет на вход первого элемента НЕ и удержит на его выходе «1». Таким образом схема находится в устойчивом состоянии, которое будет сохраняться бесконечно долго – пока на схему памяти подано электропитание. Если же на выходе первого элемента изначально установился «0», то на выходе второго элемента будет «1», т.е. схема памяти переключится в противоположное состояние.

Если принять выход одного из этих элементов (допустим первого по описанию, приведенному выше) как выход всей схемы, то в первом из описанных устойчивых состояний схема памяти запомнит «1», а во втором запомнит «0». Выход второго элемента НЕ всегда будет в противоположном состоянии относительно выхода – его называют инверсным выходом. Выход (или «прямой» выход) обозначают Q. «Инверсный» выход обозначают nQ.

Схема из двух инвертеров не позволяет переключать ее из одного состояния в другое. Чтобы выполнить это, элементы НЕ заменяют элементами И-НЕ или ИЛИ-НЕ. Соответственно у схемы памяти появляются два входа, один из которых (этот вход называют входом «установки» – SET или S) выполняет переключение схемы в состояние «1» на выходе, а другой (этот вход называют входом «сброса» – RESET или R) – в состояние «0» на выходе.

Рассмотрим работу схемы с элементами ИЛИ-НЕ. Входы SET и RESET будут иметь активный уровень «1» и неактивный «0». Когда на входы R и S подан неактивный уровень логического «0» значение на выходе будет непредсказуемым – оно установится случайным образом при включении электропитания. Если для вышерассмотренной схемы на вход первого элемента ИЛИ-НЕ подать «1», то на выходе Q установится «0», а на nQ – «1», т.е. триггер запомнит «0» и будет сохранять это значение, даже если вход перейдет в неактивное состояние «0». В соответствии с таким действием данный управляющий вход называют R (Reset или Сброс). Если же «1» подать на второй вход, то на выходе Q установится и будет «запомнена» логическая «1» (nQ = 0), которая сохранится при переходе входа S в неактивное состояние «0». Соответственно этот вход называют S (Set или Установка).

Есть две важных особенности использования элемента памяти:

1. Если на входы R и S *нельзя* одновременно подать активные сигналы, фиксатор перейдет в состояние, когда на обоих выходах будет «1». Такое состояние считается недопустимым, так как на выходах Q и nQ должно быть противоположное значение. Кроме того, при снятии активных уровней с обоих входов R и S одновременно, невозможно предсказать, в какое состояние переключится элемент памяти.
2. Переключения фиксатора - «запоминание» - происходит в момент перехода управляющего сигнала R или S в активный уровень, и после этого, элемент памяти



не изменяет своего состояния вне зависимости от того сохранился ли активный уровень на входе R или S или опять стал пассивным.

Аналогично описанной схеме работает и схема памяти на базе элементов И-НЕ. Разница только в полярности управляющих сигналов: в данном случае активным уровнем управляющих сигналов будет «0» и они будут соответственно называться  $\bar{n}R$  и  $\bar{n}S$ .

Описанная выше схема памяти является простейшим триггером – его называют триггером RS-типа или RS-триггером. RS-триггер достаточно неудобен в управлении, но может быть усовершенствован и преобразован в другие типы триггеров путем подключения дополнительной схемы управления. Таким образом RS-триггер основа большинства триггеров других типов.

### 7.2.2 Схема управления.

Описанная выше схема памяти имеет два независимых управляющих входа R и S, записывающих в нее «0» и «1» соответственно.

Однако использование только входов R и S не всегда удобно и сильно ограничивает возможности по запоминанию значений и по использованию триггеров в последовательностных схемах. Поэтому добавляют схему управления, которая преобразует более сложные комбинации управляющих сигналов, в том числе последовательности сигналов, в описанные сигналы R и S. Схема управления задает как логику (правила реакции на входные сигналы) работы триггера, так и способ приема информации триггером.

### 7.2.3 Асинхронные и синхронные управляемые уровнем триггеры.

#### 7.2.4 RS-триггеры.

Функционирование и внутренняя схема *асинхронного RS-триггера* соответствует описанной выше схеме памяти (см. п.7.2.1). Данный тип триггера фактически не имеет схемы управления на входе.

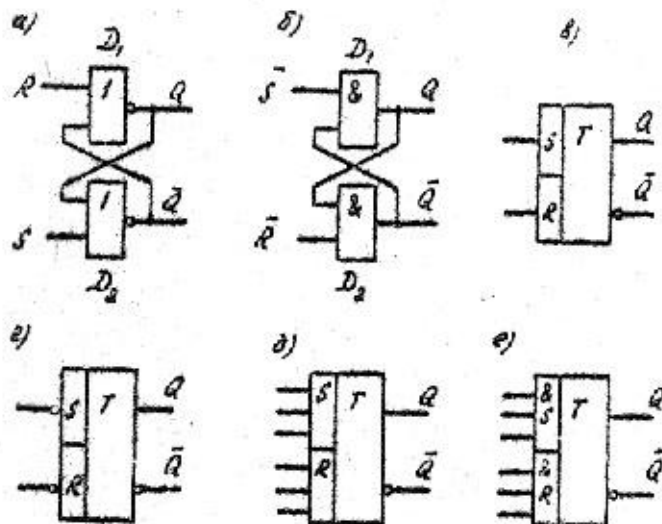


Рисунок 19 Асинхронные RS-триггеры на элементах И-НЕ (а), ИЛИ-НЕ (б) и их условные обозначения (в, г)

*Синхронные RS-триггеры с управлением уровнем* имеют вход синхронизации (C), сигнал на котором разрешает управление с информационных входов, пока находится в активном состоянии. Входные информационные сигналы «пропускаются» через логические вентили И (для схемы на базе элементов ИЛИ-НЕ, активный уровень (C) = 1) или через вентили ИЛИ (для схемы на базе элементов И-НЕ, активный уровень (C) = 0).

Характерной особенностью триггерных схем, управляемых уровнями синхроимпульса, является то, что изменение уровней информационных сигналов в течение действия синхроимпульса приводит к изменению состояния триггера. Если по фронту синхросигнала в триггер было занесено значение «1», и далее в течение удержания активного уровня синхросигнала  $S$  информационные сигналы приняли значения  $R=1$ ,  $S=0$ , то это приводит к переключению триггера в состояние «0». Следовательно, для обеспечения работы триггера сигналы  $R$  и  $S$  должны оставаться неизменными в течение активного уровня синхроимпульса, а изменяться в течение паузы между синхроимпульсами.

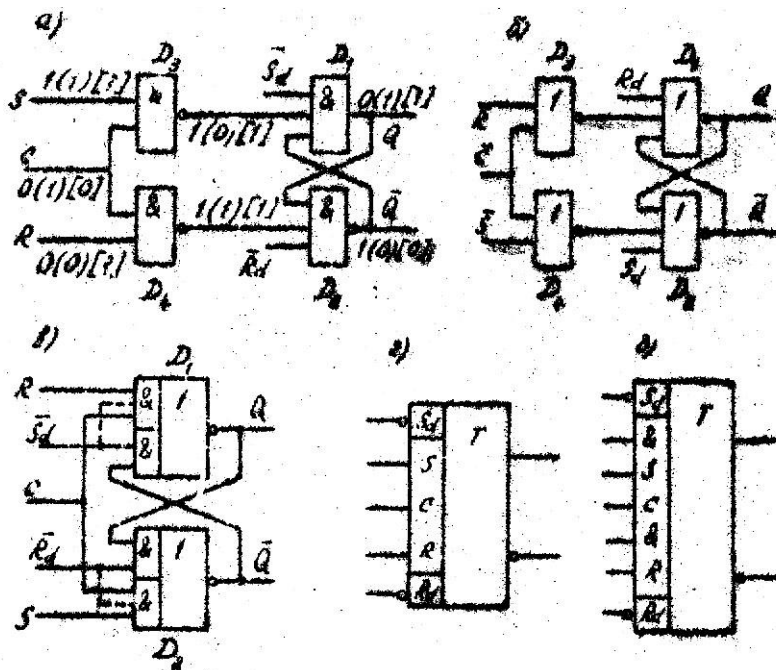


Рисунок 20 Синхронные RS-триггеры с управлением уровнем на элементах И-НЕ (а), ИЛИ-НЕ (б) и их условные обозначения (в, г)

Синхронный RS-триггер с управлением по фронту построен по многоступенчатой схеме, описанной ниже.

RS-триггеры редко используются как самостоятельные элементы, но, как говорилось выше, являются элементом памяти (фиксатором) в составе других типов триггеров.

### 7.2.5 D-триггеры.

*D-триггером* (или триггером задержки сигнала) называют триггер с одним информационным входом ( $D$ ) и входом синхронизации ( $C$ ). По импульсу синхронизации ( $C$ ) состояние на входе ( $D$ ) сохраняется в триггере и устанавливается на выходе ( $Q$ ). Состояние выхода  $Q$  останется неизменным до следующего импульса на входе синхронизации. Как видно из описания  $D$ -триггер является синхронным (обязательно используется сигнал синхронизации). Можно построить также схему асинхронного  $D$ -триггера (см. Рисунок 21 а)), но она будет иметь смысл только как повторитель с задержкой на переключение, а не как схема памяти. Асинхронные  $D$ -триггеры почти не используются.

Из широко применяемых первый - *D-триггер с управлением по уровню*, так же называемый «триггер-защелка» (LATCH) (см. Рисунок 21 г). Он прозрачен для входного информационного сигнала пока сигнал синхронизации ( $C$ ) в активном состоянии и защелкивает значение на выходе ( $Q$ ) в момент перехода ( $C$ ) в неактивное состояние. Такой  $D$ -триггер представляет собой RS-триггер с синхронизацией по уровню, у которого на

вход (S) подключен информационный сигнал (D), а на вход (R) – инвертированный информационный сигнал. То есть в схему управления добавляется один инвертор «по входу». Входы Rd и Sd используются для асинхронных сброса или установки триггера в определенное состояние и удержание его в этом состоянии, если требуется. Для устойчивой работы такого D-триггера необходимо, чтобы состояние входа D не изменялось в момент действия синхроимпульса на входе C.

Для расширения функциональности к D-триггеру можно добавить вход разрешения V. При активном уровне (в данном случае «1») на данном входе триггер работает в вышеописанном режиме, при  $V = \langle 0 \rangle$ , триггер не реагирует на входы C и D. Поведение DV-триггера описывается формулой  $Q^{n+1} = D^n V^n \vee Q^n \bar{V}^n$ .

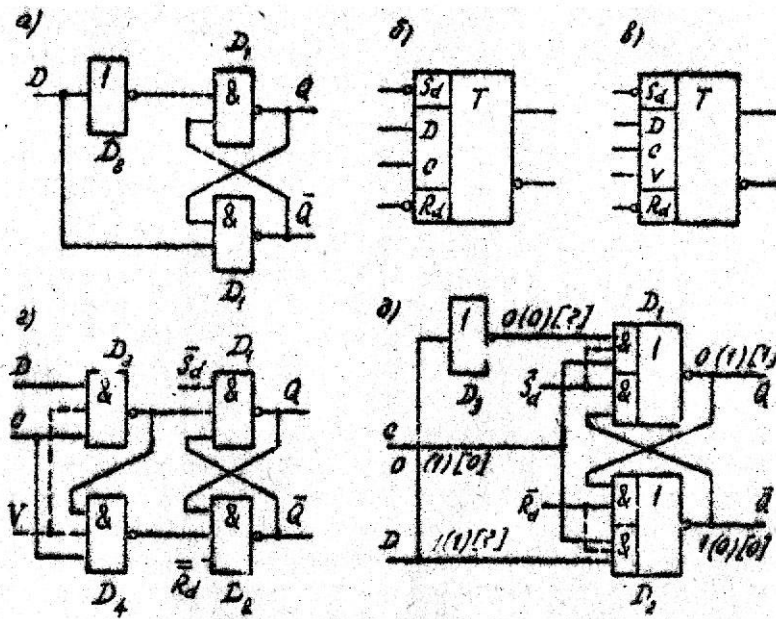


Рисунок 21 Асинхронный (а) и управляемые уровнем синхроимпульса (г,д) D-триггеры и их условные обозначения (б,в)

Второй тип, наиболее часто используемый - *D-триггер с управлением по фронту синхросигнала*, так называемый FLIP-FLOP. Данные записываются в него в момент перепада синхросигнала и сохраняются неизменными до следующего фронта. Такие триггеры строятся по двухступенчатой схеме и будучи описаны ниже.

D-триггер является одним из самых широко используемых типов. Такие триггеры выпускаются отдельные микросхемы, а также являются базовыми структурными элементами многих ПЛИС. Несколько D-триггеров, с объединенными тактирующими входами образуют многоразрядные регистры: синхронные регистры (управление по уровню) или регистры-защелки (управление по фронту).

### 7.2.6 T-триггеры.

T-триггер представляет собой схему с одним логическим входом T. Так как этот триггер работает в режиме учета импульсов на входе, его иногда называют счетным триггером (триггером со счетным входом). T-триггер изменяет свое состояние на противоположное после воздействия импульса, поступающего на вход T, т.е. его функционирование описывается формулой:  $Q^{n+1} = \bar{T}^n Q^n \vee T^n \bar{Q}^n$ .

Схема простейшего T-триггера с элементами задержки D5 и D6 в цепях обратной связи приведена ниже (см. Рисунок 22). Элементы задержки обеспечивают надежное

переключение триггера, причем время задержки  $t_3$  на этих элементах должно быть больше длительности синхроимпульса на входе Т.

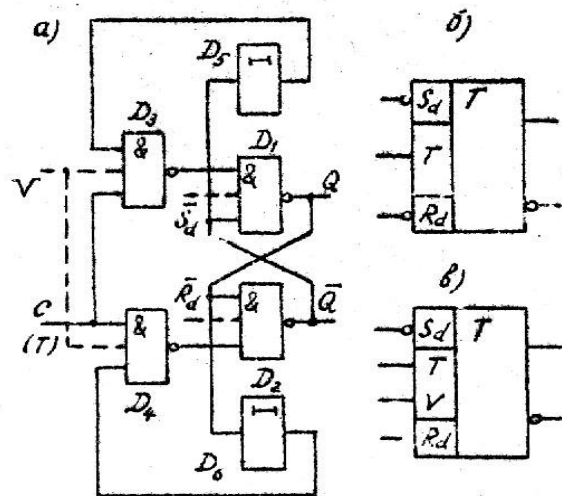


Рисунок 22 Т- и TV-триггеры, управляемые уровнем синхроимпульса

Пусть  $Q = 1$ ,  $\bar{Q} = 0$ . Поступающий на счетный вход импульс приводит к появлению нулевого сигнала на выходе элемента  $D_4$  (на его входах – две «1») и последовательной установке сигналов  $\bar{Q} = 1$ ,  $Q = 0$ , т.е. к опрокидыванию триггера в нулевое состояние. При этом сигнал на входе  $D_3$  не изменяется, так как на его входе в течение  $t_3$  действует нулевой сигнал, поступающий с выхода  $\bar{Q}$  через элемент задержки  $D_5$ .

После окончания действия счетного импульса сигналы на выходах элементов  $D_3$  и  $D_4$  принимают единичные значения, а на вход  $D_3$  через элемент задержки  $D_5$  поступает разрешающий уровень с выхода  $\bar{Q}$ . В результате следующий счетный импульс перебросит триггер в исходное состояние.

Элементы  $D_5$  и  $D_6$  обеспечивают задержку появления сигналов обратной связи с тем, чтобы в течение действия счетного импульса не происходило многократного переключения триггера. В интегральных триггерах роль элементов задержки могут выполнять либо логические элементы, либо специальные полупроводниковые приборы с накоплением заряда. Однако, в силу жестких ограничений на длительность импульса на счетном входе и сложности реализации элементов задержки, Т-триггеры на базе RS-триггера с управлением уровнем синхросигнала почти не используются, а применяются триггеры с управлением по фронту синхросигнала.

TV-триггер является разновидностью Т-триггера с дополнительным входом разрешения счета. Принцип его работы аналогичен DV-триггеру, описанному выше.

### 7.2.7 JK-триггеры.

JK-триггер похож на RS-триггер (вход  $J$  эквивалентен входу  $S$ , а вход  $K$  - входу  $R$ ), но не имеет запрещенного состояния  $K(R)=1$ ,  $J(S)=1$ . При условии  $J=K=1$  триггер осуществляет инверсию предыдущего состояния, а при остальных комбинациях входных сигналов функционирует в соответствии с таблицей истинности RS-триггера. Закон функционирования синхронного JK-триггера отражен логическом уравнении:

$$Q^{n+1} = \bar{K}^n Q^n \vee J^n \bar{Q}^n.$$

$JK$ -триггер можно получить из  $RS$ -триггера, связав выходы триггера с его входами обратной связью. При этом схема управления должна быть построена таким образом, чтобы на входах  $R$  и  $S$  единичные сигналы одновременно не появлялись. Тогда при  $J=K=1$  схема управления принуждает триггер работать в режиме переключения (счетный режим), так как, благодаря связи с выхода на вход, учитывается предшествующее состояние схемы, и сигнал направляется на соответствующий вход  $RS$ -триггера, вызывая его переключение.

Принципиально можно построить асинхронный  $JK$ -триггер (см. Рисунок 23 а), но они очень неустойчивы практического применения не находят: для нормальной работы асинхронного триггера в счетном режиме длительность управляющих сигналов не должна превышать времени переключения триггера, что очень сложно обеспечить на практике. В противном случае схема будет непрерывно переходить из одного состояния в другое, пока хотя бы один из сигналов не станет равным нулю.

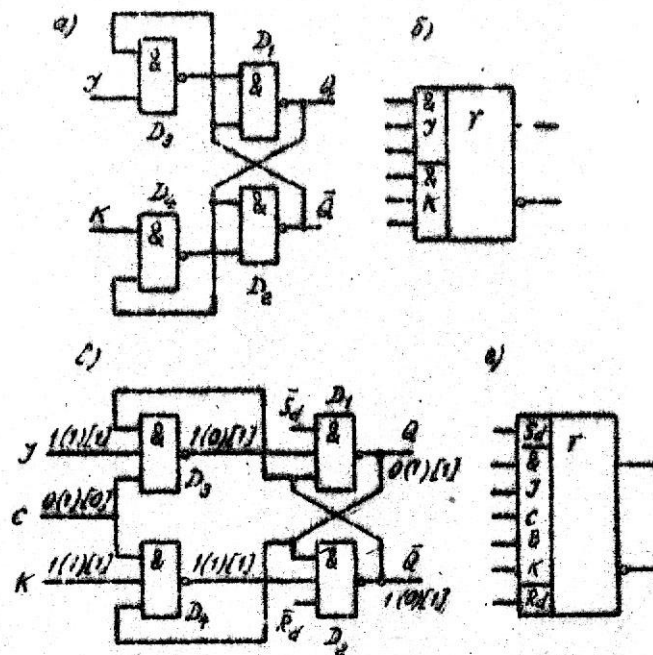


Рисунок 23 Асинхронный (а) и синхронный с управлением уровнем импульса (в)  $JK$ -триггеры и их условные обозначения (б, г)

Синхронный вариант  $JK$ -триггера с управлением уровнем синхроимпульса получается из асинхронного путем добавления входа для подачи синхроимпульсов (Рисунок 23.в). В этой схеме занесение информации в триггер происходит при поступлении синхроимпульса. На Рисунок 23.в сигналами «0» и «1» отражены значения сигналов на входах и выходах элементов до (без скобок) в течение (в круглых скобках) и по окончании (в квадратных скобках) действия синхроимпульсов при функционировании триггера в счетном режиме при  $J=K=1$ . Максимальная длительность синхроимпульса должна быть чуть меньше, чем задержка в цепочке  $D1(D2)+D4(D3)$ . Превышение этой величины вызывает появление ложного сигнала на выходе элемента  $D4(D3)$ , что может привести к повторному переключению триггера. С другой стороны, длительность синхроимпульса ограничена по минимуму задержкой цепочки  $D1+D2$ , чтобы успел переключиться триггер. Чтобы выполнить эти условия на практике потребуется подбирать задержки элементов (причем, у  $D3$  и  $D4$  они должны быть больше чем у  $D1$  и  $D2$ ) и длительность синхроимпульса. Учитывая большие разбросы временных параметров, выполнить все условия на практике не представляется возможным. Поэтому рассмотренные синхронные  $JK$ -триггера также как и асинхронные на практике фактически нереализуемы. Работоспособность  $JK$ -триггеров обеспечивается только в схемах, управляемых фронтом (срезом) синхроимпульса (см. далее).

### 7.3 Синхронные управляемые фронтом (универсальные) триггеры.

#### 7.3.1 Общие сведения об универсальных триггерах.

Асинхронные и управляемые уровнем синхроимпульса синхронные триггеры в современных устройствах вычислительной техники находят ограниченное применение. Из них главным образом используются асинхронные и синхронные  $RS$ -триггеры и  $D$ -триггеры управляемые уровнем синхроимпульса («защелка»), на основе которых можно реализовывать запоминающие регистры, сдвигающие регистры или счетчики.  $JK$ - и  $T$ -триггеры, управляемые уровнем синхроимпульса, не используются из-за фактически невыполнимых требований к длительности синхроимпульсов и к параметрам задержек для внутренних логических элементов.

Чтобы устранить жесткие требования к параметрам синхроимпульсов можно использовать многотактные схемы, где запись данных и формирование сигналов обратной связи будут выполняться в несколько этапов, связанных с отдельными импульсами синхронизации. Однако это замедлит работу схемы, потребует более сложного генератора синхроимпульсов, способного синтезировать сложные последовательности импульсов.

Поэтому в основном используют одноктактные («срабатывающие» по одному синхроимпульсу) триггерные схемы, в которых переключение состояния триггера происходит только в момент изменения (фронта или среза) уровня синхроимпульса. На базе таких триггеров строятся различные операционные элементы: запоминающие и сдвигающие регистры, схемы памяти для конечных автоматов и другие, поэтому такие триггеры часто называют универсальными.

Универсальные триггеры управляемые фронтом синхроимпульса бывают различных типов:  $RS$ -,  $D$ ,  $JK$ -. Но принцип их построения схож и заключается в том, что в их схему управления вводятся дополнительные элементы, обеспечивающие блокировку информационных сигналов и сигналов обратной связи сразу после фронта и в течение действия синхроимпульсов с целью исключения влияния изменения входных информационных сигналов и внутреннего состояния триггерной схемы на состояние триггера до следующего фронта синхроимпульса. В качестве таких блокирующих элементов используются либо дополнительные асинхронные  $RS$ -триггеры, либо аналоговые элементы памяти (диоды и транзисторы с накоплением заряда, конденсаторы или барьерные емкости переходов), либо их сочетание.

Можно выделить три основных типа универсальных триггеров:

- а) Триггеры  $MS$ -типа;
- б) Трехтриггерные ячейки ( $ТЯ$ ) на основе основного и двух коммутируемых триггеров;
- в) Триггеры с использованием аналоговых элементов внутренней памяти.

#### 7.3.2 $MS$ -триггеры.

$MS$ -триггеры содержат в своем составе основной/главный ( $MASTER$ ) и вспомогательный/подчиненный ( $SLAVE$ ) – триггеры. В зависимости от организации связей между основными и вспомогательными триггерами существующие  $MS$ -триггеры подразделяются на схемы с инвертором и схемы с запрещающими связями.

##### 7.3.2.1 $MS$ -триггер с инвертором.

$MS$ -триггер с инвертором фактически является двухтактным, так как его переключение выполняется в два шага: по переднему и заднему фронтам синхроимпульса. Однако, так как переключение триггера в сумме выполняется за *один* синхроимпульс, то такой триггер можно рассматривать как псевдо-одноктактный.

Схема  $MS$ -триггера с инвертором приведена на рисунке (см. Рисунок 24). Схема включает в себя два  $RS$ -триггера, управляемых уровнем синхроимпульса:  $M$  – триггер на элементах  $D1$ - $D4$  и  $S$  – триггер на элементах  $D5$ - $D8$ . Синхроимпульс на вторую пару

схем совпадения подается через инвертор  $D_9$ , чем и объясняется название триггера «MS-триггер с инвертором».

Информация заносится в M-триггер по переднему фронту синхроимпульса, а по заднему фронту (по спаду) синхроимпульса – переписывается из M-триггера в S-триггер. Для срабатывания M- и S-триггеров по переднему и заднему фронтам - каждого по своему, в цепи синхросигнала между M- и S-триггерами включается инвертор  $D_9$ , блокирующий перезапись информации в S – триггер, пока  $C=1$  и выполняется запись в M – триггер.

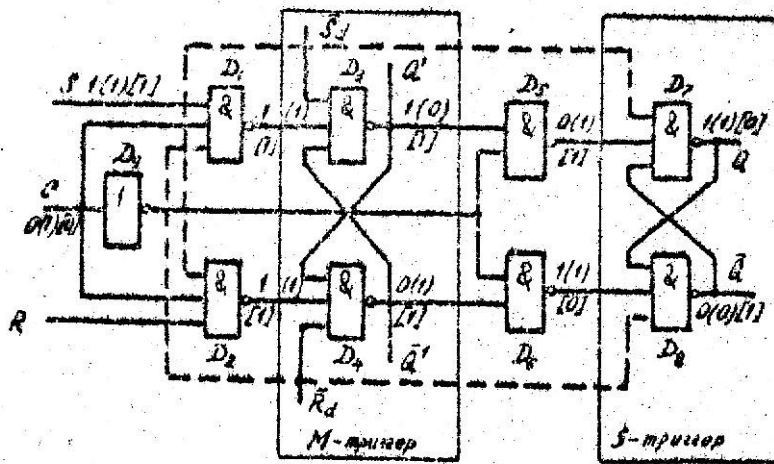


Рисунок 24 Универсальный RS-триггер с инвертором

В исходном состоянии ( $C=0$ ) оба триггера (M и S) находятся в одинаковых состояниях: в них сохраняется информация, записанная по предыдущему синхроимпульсу. При поступлении синхроимпульса ( $C=1$ ) новая входная информация записывается в M-триггер. При этом состояние S-триггера и, соответственно, значения на выходах схемы остаются неизменными, т.к. он заблокирован низким уровнем сигнала синхронизации S-триггера, поступающим с выхода  $D_9$  на входы  $D_5$  и  $D_6$ . По спаду синхроимпульса на входе  $C$  ( $C=0$ ) состояние M-триггера фиксируется и остается неизменным, пока  $C=0$ . Это обеспечивается блокирующими элементами  $D_1$  и  $D_2$ . Таким образом, в MS-триггер записывается информация, установленная на информационных входах непосредственно перед спадом синхроимпульса.

Одновременно с блокировкой M-триггера синхросигнал S-триггера, поступающий с выхода  $D_9$  на входы  $D_5$  и  $D_6$ , становится равным «1», что ведет к перезаписи информации с выходов M-триггера в S-триггер. При этом значения на информационных входах MS-триггера R и S могут изменяться как угодно – они не будут восприняты заблокированным M-триггером.

Таким образом, при записи в MS-триггер длительность синхроимпульса может быть произвольной – это не приведет к многократному переключению выходных сигналов, т.к. заблокирован S-триггер, а в процессе установки выходных значений на них не будут влиять изменяющиеся в этот момент информационные сигналы, т.к. заблокирован M-триггер.

На практике MS-триггеры дополняются асинхронными информационными входами  $S_d$  и  $R_d$ , для принудительной установки MS-триггера в единичное или нулевое состояние. Эти установочные сигналы подаются и на M- и на S-триггеры, а также их используют для блокировки входных элементов  $D_1$  и  $D_2$  для исключения влияния воздействия синхроимпульса в момент асинхронной установки (см. цепи, обозначенные на схеме пунктиром).

MS-триггер RS-типа легко преобразовать в T-триггер или в JK-триггер путем подключения выходов Q и  $\bar{Q}$  ко входам R и S соответственно. Так как схема MS-

триггера фактически работает в двухтактном режиме (по фронту и спаду синхроимпульса), то проблем с многократным переключением триггера из за обратных связей, подобных рассмотренным выше Т- и JK-триггерам управляемых уровнем синхросигнала, не будет. Схема приведена на рисунке ниже (Рисунок 25). В случае Т-триггера входы J- и K- не используются ( $D_1$  и  $D_2$  будут двухвходовыми элементами).

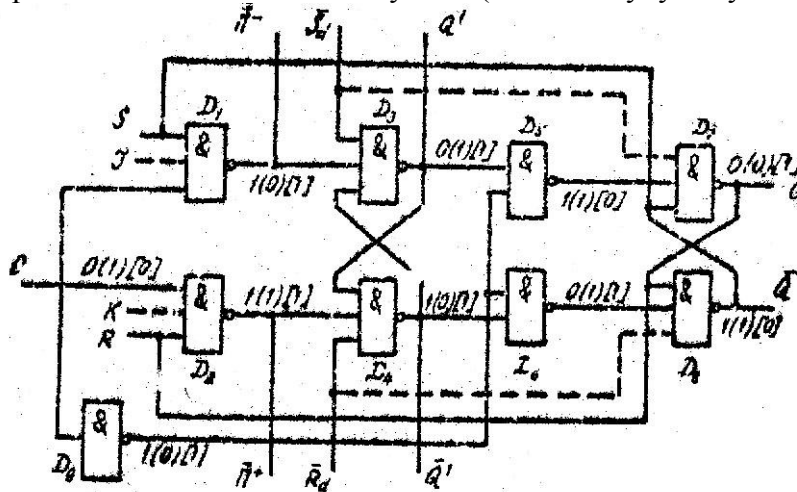


Рисунок 25 Т- и JK-триггер на основе универсального MS-триггера с инвертором.

Параметры, определяющие быстродействие MS-триггера - минимальная длительность ( $t_n$ ) управляющего импульса и максимально возможная частота ( $f$ ) переключения (при  $t_n = t_{n \min}$ ) соответственно равны  $t_{u \min} = 2 * t_{з \text{ ср max}}$ ,  $f_{\max} = 1/(6 * t_{з \text{ ср max}})$ , где  $t_{з \text{ ср max}}$  - наибольшая величина задержки распространения сигнала через логический элемент.

### 7.3.2.2 MS-триггер с запрещающими связями.

Существенным недостатком триггера с инвертором является возможность возникновения в нем опасных состязаний при больших разбросах времени задержки распространения сигнала логических элементов. Если в схемах, представленных выше,  $t_{31} + t_{33} > t_{34}$  ( $t_{32} + t_{34} > t_{39}$ ), где ( $t_{3i}$  - время задержки соответствующего элемента, то процесс занесения информации в M-триггер происходит быстрее, чем будут заблокированы элементы  $D_5$  и  $D_6$  нулевым сигналом, формируемым на выходе инвертора  $D_9$ . В этом случае возникает опасность того, что вновь поступившая информация будет также занесена в S-триггер во время действия синхроимпульса, а не по его окончании. В схемах Т-триггера и JK-триггера в счетном режиме это приводит к преждевременному изменению сигналов обратной связи и новому переключению M-триггера. При работе в составе сдвигающего регистра преждевременное изменение информационных сигналов  $Q$  и  $\bar{Q}$  вызывает внесение ложной информации в M-триггер последующего разряда.

От этого недостатка свободен MS-триггер с запрещающими связями (Рисунок 26), в котором функцию инвертора синхросигнала выполняют элементы  $D_1$  и  $D_2$ . При выработке любым из этих элементов управляющего нулевого сигнала (что означает, что M-триггер должен изменить свое состояние) до переключения M-триггера производится блокировка элементов  $D_5$  и  $D_6$ , чем обеспечивается сохранение ранее занесенной информации в S-триггер. Благодаря этому исключаются условия для возникновения опасных состязаний. В остальном принцип работы рассматриваемой схемы ничем не отличается от принципа работы схемы MS-триггера с инвертором.

Цепи асинхронной установки организуются аналогичным образом.

Схемы Т-триггера и JK-триггера также получаются аналогично описанным для MS-триггера с инвертором.



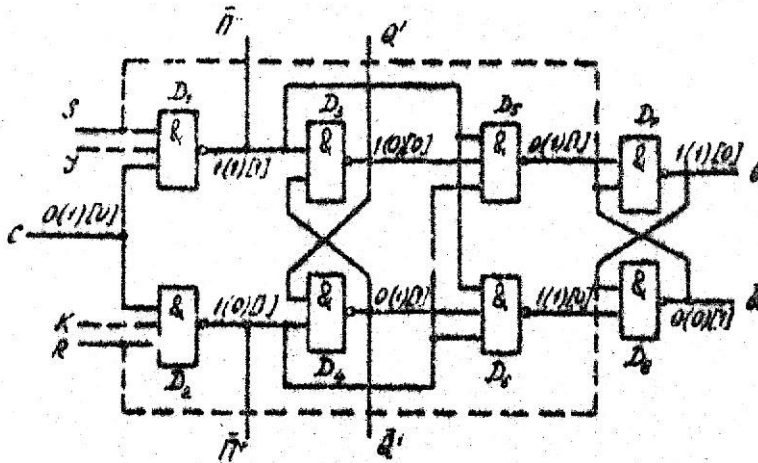


Рисунок 26 MS-триггер с запрещающими связями

## 7.3.2.3 MS-триггер D-типа.

D – триггер можно реализовывать либо по схеме с инвертором, либо по схеме с запрещающими связями. Так как эти устройства существенно не отличаются, принцип работы D – триггера рассматривается на примере схемы с запрещающими связями (Рисунок 27). D – триггер имеет один информационный вход (D), который играет роль входа S, а сигнал R вырабатывается путем инверсии сигнала S ( $R = \bar{S}$ ) на элементе D1. Кроме подключения входа R к выходу элемента D1 схема D-триггера ничем не отличается от описанных ранее.

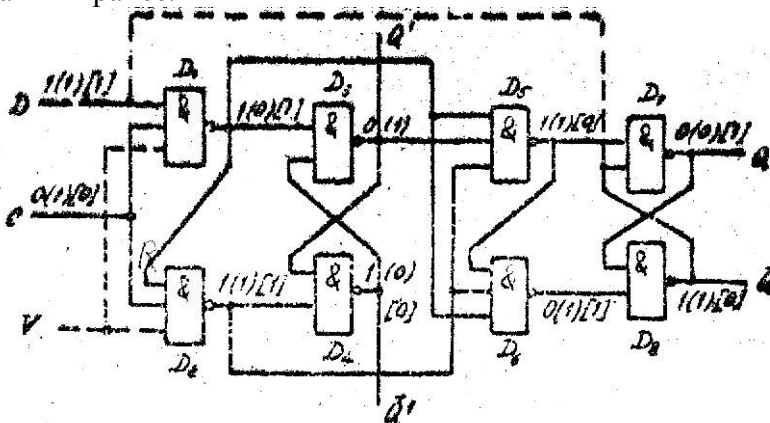


Рисунок 27 В-триггер на базе MS-триггера с запрещающими связями

## 7.3.2.4 Универсальная трехтриггерная ячейка.

В схеме трехтриггерной ячейки (Рисунок 28) используются три типовых асинхронных RS – триггера, один из которых является основным, и два других – коммутирующими. Коммутирующие триггеры реализованы на элементах D1, D2 и D3, D4, а главный триггер – на элементах D5 и D6. По переднему фронту синхроимпульса на входе С коммутирующие триггеры сохраняют значения со входов notR и notS соответственно. С выходов коммутирующих триггеров сигналы попадают на входы установки основного триггера (на элементах D5 и D6) и одновременно с этим на входы блокировки противоположного коммутирующего триггера. Соответственно, если активный уровень «0» подан на вход notR, это приведет к блокировке нижнего коммутирующего триггера и, соответственно, к невозможности изменения состояния триггера (в данном примере - его установки), до следующего синхроимпульса.

Если соединить выходы со входами, как это показано штриховой линией на рисунке, то получим Т-триггер. Для функциональности JK-триггера нужно использовать дополнительные входы элементов D2 и D3.

Схема D-триггера может быть получена, также, как описано для MS-триггера – путем подключения на вход R инвертированного сигнала  $S=D$ .

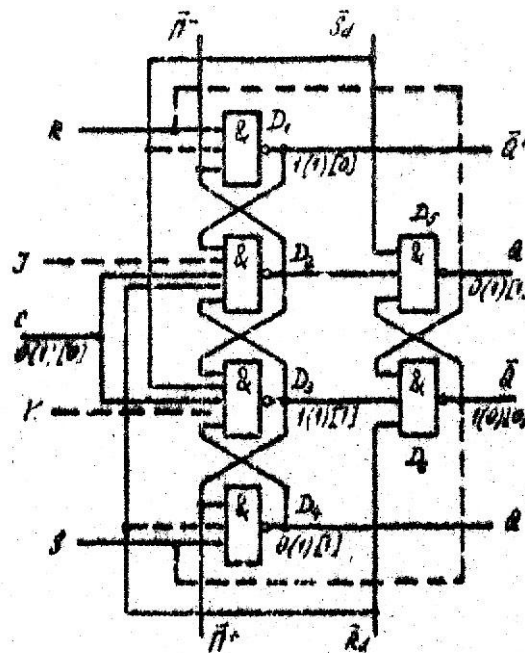


Рисунок 28 Триггерная ячейка на основе трех триггеров

### 7.3.2.5 Универсальные триггеры с аналоговыми элементами памяти.

Недостатком схем MS-триггеров и трехтриггерных ячеек является большое число логических элементов в их составе: 6-9 шт. Поэтому в настоящее время не менее широко распространены триггерные схемы, в которых в качестве схем памяти (фиксаторов) используются не  $RS$ -триггеры, а электронные элементы, способные в течение некоторого интервала времени сохранять электрический заряд. Такую функцию могут выполнять либо конденсаторы, либо барьерные ёмкости переходов, либо для этой цели используется эффект накопления заряда в диодах и транзисторах. Подобные компоненты называют аналоговыми элементами памяти, чем подчеркивается их принципиальное отличие от дискретных (триггерных) элементов памяти. Благодаря применению аналоговых элементов памяти, число радиокомпонентов, входящих в состав универсальных триггеров, приблизительно в 2-3 раза меньше, чем их число в дискретных триггерных схемах. Принцип построения триггеров с аналоговыми элементами памяти рассматривается на примерах построения универсальных  $JK$ -триггеров на основе ТТЛ схемы с барьерными запоминающими емкостями (Рисунок 29).

В данном триггере роль запоминающих элементов выполняют барьерные емкости диодов  $D_1$  и  $D'_1$ , шунтирующие транзисторы  $T_4$  и  $T'_4$  вспомогательного триггера с простейшими резистивными цепями обратной связи. Главный триггер на ТТЛ элементах со сложным инвертором (транзисторы  $T_5-T_9$  и  $T'_5-T'_9$  соответственно). Синхронизирующий и информационные сигналы  $J$  и  $K$ , а также сигналы обратной связи с выходов  $Q$  и  $\bar{Q}$  главного триггера подаются на входы управляющих логических схем (на транзисторах  $T_1-T_3$  и  $T'_3-T'_3$ ) с открытыми коллекторными выходами, которые подключены к базам промежуточных (фазорасщепительных) транзисторов  $T_6$  и  $T'_6$ . Синхроимпульс подается как на входы (эмиттеры транзисторов  $T_1$  и  $T'_1$ ), так и на эмиттеры выходных транзисторов  $T_3$  и  $T'_3$  управляющих логических схем.

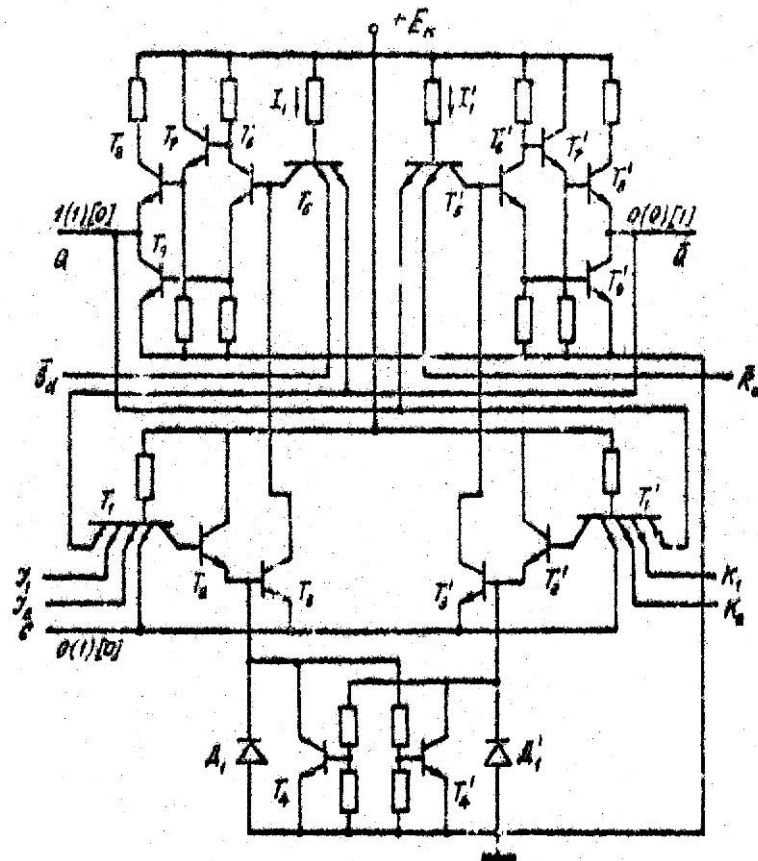


Рисунок 29 Универсальный JK-триггер с барьерными запоминающими емкостями

Для изучения принципа действия триггера достаточно рассмотреть его работу в счетном режиме, т.е. при  $J_1 = J_2 = K_1 = K_2 = 1$

Пусть до поступления сигнала на вход  $C$  ( $C=0$ , символы без скобок) триггер находится в единичном состоянии ( $Q=1, \bar{Q}=0$ ). Ток  $I_1$  протекает через коллекторный переход  $T_5$  и открытый транзистор  $T_9$ , а ток  $I_1'$  через коллекторный переход  $T_5$  (так как оба его эмиттерных перехода заперты) поступает в базу  $T_6$ , вследствие чего транзисторы  $T_6'$  и  $T_9$  поддерживаются в режиме насыщения ( $\bar{Q}=0$ ). Токи  $I_2$  и  $I_2'$  замыкаются через соответствующие эмиттерные переходы  $T_1'$  и  $T_1'$  на ширину синхроимпульсов (ток  $I_2$  частично протекает через открытый транзистор  $T_9$ ), поэтому транзисторы  $T_2, T_3, T_2', T_3'$ , а так же  $T_4'$  и  $T_4'$ , закрыты.

Во время действия единичного сигнала на входе  $C$  (символы в круглых скобках) транзисторы  $T_3$  и  $T_3'$  не открываются (так как на их эмиттерах устанавливается высокий уровень напряжения) и поэтому главный триггер своего состояния не изменит ( $Q=1, \bar{Q}=0$ ). В течении этого интервала времени осуществляется заряд барьерной емкости диода  $D_1'$  через транзистор  $T_2'$ , который открывается током  $I_2'$ . Этот ток протекает в базу  $T_2'$  через коллекторный переход  $T_1'$ , так как все его эмиттерные переходы в течении действия управляющего импульса заперты. При этом транзистор  $T_4'$  заперт (ток  $I_2$  замыкается через открытый транзистор  $T_9$ ) и не препятствует заряду емкости диода  $D_1'$ .

Эмиттерный ток  $T_2'$  наряду с осуществлением заряда этой емкости также обеспечивает введение транзистора  $T_4$  в режиме насыщения.

После окончания управляющего сигнала ( $C=0$ , символы в квадратных скобках) состояние вспомогательного триггера ( $T_4, T_4'$ ) в течение некоторого интервала времени поддерживается неизменным благодаря заряду, накопленному в барьерной емкости диода  $D_1'$ . В результате разряда этой емкости через эмиттерный переход  $T_3'$ , последний отпирается и ток  $I_1'$  будет замыкаться через  $T_3'$ . Это приводит к запиранию транзисторов  $T_6$  и  $T_9$ , т.е. к переключению триггера в нулевое состояние, после чего  $I_1$  будет протекать в базу транзистора  $T_6$  а ток  $I_1'$  - через открытый транзистор  $T_9$ . По окончании процесса переключения триггера накопленный заряд рассасывается.

Накопленный в диодах заряд защищает триггер от ложных срабатываний, если во время действия синхросигнала на информационные входы поступают короткие импульсы помех. Входы  $\bar{S}_d$  и  $\bar{R}_d$  предназначены для начальной установки триггера.

Триггерные схемы с аналоговыми элементами памяти накладывают определенные ограничения на максимальную длительность спада синхроимпульсов при большой длительности спада барьерная емкость диода успевает разрядиться до окончания процесса переключения главного триггера, что может привести к сбою в работе триггера.

## 8 РЕГИСТРЫ (В ПРОЦЕССЕ РЕДАКТИРОВАНИЯ)

Регистром называется устройство, предназначенное для запоминания  $N$ - разрядного двоичного числа (слова), а также выполнений над ним ряда логических преобразований. Регистр представляет собой совокупность  $N$  триггеров, число которых соответствует количеству разрядов в слове, и вспомогательных схем, обеспечивавших выполнение требуемых функций. По функциональному назначению регистры подразделяются на накопительные (регистры памяти) и сдвигающие. В общем случае накопительные регистры обеспечивают выполнение следующих операций:

- обнуление триггеров регистра (сброс);
- прием (занесение) слова из другого устройства (регистра, сумматора, запоминающего устройства и т.д.) и хранение его в течение требуемого интервала времени;
- передачу слова в другой регистр;
- преобразование прямого кода в обратный и наоборот.

Сдвигающие регистры наряду с отмеченными функциями накопительных регистров обеспечивают реализацию следующих дополнительных операций:

- сдвиг слова вправо или влево на требуемое количество разрядов;
- преобразование последовательного кода в параллельный и наоборот.

В сдвигающих регистрах обычно выполняются и поразрядные логические операции (логическое сложение, логическое умножение и поразрядное сложение).

Конкретные регистры допускают выполнение части указанных операций.

### 8.1 Накопительные регистры

#### 8.1.1 Накопительные регистры на асинхронных $RS$ – триггерах.

По способу занесения (приема) информации накопительные регистры на  $RS$  – триггерах подразделяются на схемы с однопроводной (с предварительным гашением) и парафазной (без предварительного гашения) передачей заносимых сигналов. Схемы регистров с однопроводной передачей информации, выполненных на логических элементах ИЛИ-НЕ и И-НЕ, показаны на рис. 25, а и б соответственно.

Занесение информации в эти регистры осуществляется в два этапа. На первом этапе при воздействии управляющего сигнала  $Y_2$ , поступающего на входы  $R$  (рис. 25, а), все триггеры регистра ( $RG$ ) устанавливаются в нулевое состояние, т. е. реализуется микрооперация обнуления

$$Y_2 : RG := 0$$

На втором этапе при поступлении управляющего сигнала  $Y_2$ , производится прием информации, поступающей по шинам  $X$  ( $X = x_1, x_2, \dots, x_n$ ), т.е.

$$Y_1 : RG := X$$

В связи с тем, что роль элементов совпадения ( $D_1, D_2, \dots, D_n$ ) в схеме рис. 25, а выполняют ЛЭ ИЛИ-НЕ, управляющий сигнал  $Y_1$  при приеме информации должен иметь нулевой уровень, а входные сигналы  $X_1, X_2, \dots, X_n$  – необходимо подавать в обратном коде. В этом случае сигнал, поступающий на вход  $S$   $i$ -го триггера, определяется логическим соотношением  $S_i = \overline{x_i \vee Y_2} = x_i y_i$ , т.е. если  $x_i = 1$ , то  $S_i = 1$  и при  $x_i = 0$ ,  $S_i = 0$ .

Если регистр реализуется на основе ЛЭ И-НЕ (рис. 25, б), то микрооперация обнуления должна осуществляться подачей нулевого уровня сигнала на шину  $Y_2$ , а прием информации происходит в момент, когда управляющий сигнал  $Y_1$ , принимая единичное значение. Входные сигналы подаются в прямом коде.

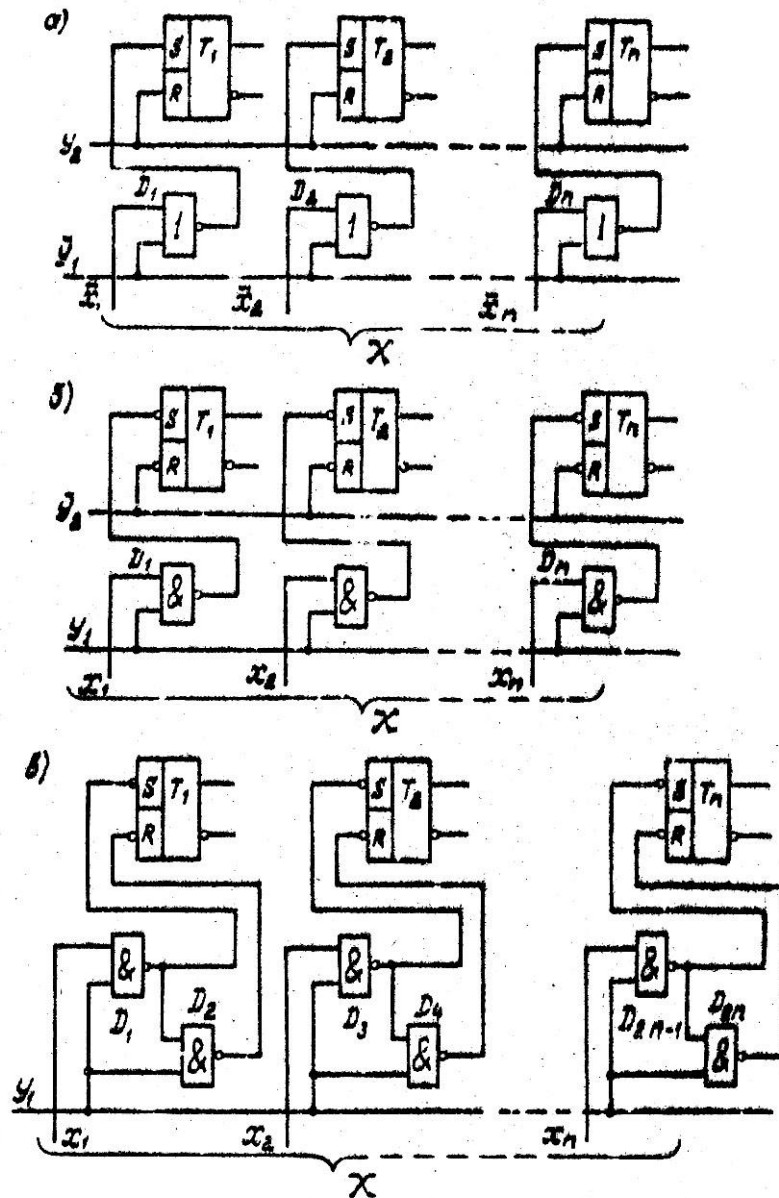


Рис. 1. Накопительные регистры на асинхронных RS-триггерах

Недостатком рассмотренных схем является необходимость предварительного обнуления триггеров регистра. От этого недостатка свободна схема, в которой занесение информации осуществляется в парафазном коде, т.е. сигналы, поступающие на установочные входы  $R$  и  $S$  триггеров при поступлении управляющего сигнала занесения, всегда взаимоинверсны (рис. 25, в). В этой схеме при  $Y_1=0$  сигналы на выходах всех логических элементов ( $D_1-D_{2n}$ ) имеют единичные значения, вследствие чего в триггерах регистра сохраняется ранее записанная информация. При занесении информации сигнал  $Y_1$  принимает единичное значение, в результате чего на входах  $S$  и  $R$   $i$ -го триггера  $\overline{S}_i = x_i Y_1$ ,  $\overline{R}_i = x_i \overline{Y_1}$ . Если, например,  $x_1=1$ , то уровень сигнала на выходе ЛЭ  $D_1$  (входе  $\overline{S}$  триггера  $T_1$ )  $\overline{S}_1 = 0$ , а на выходе ЛЭ  $D_2$ , так как на одном из его выходов действует нулевой сигнал ( $\overline{S}_1 = 0$ ),  $\overline{R}_1=1$ . Под воздействием этих сигналов триггер  $T_1$  устанавливается в единичное положение. Если в то же время на вход  $x_2$  подается нулевой сигнал ( $x_2=0$ ), то на выходе ЛЭ  $D_3$   $\overline{S}_2 = 1$ , а в результате совпадения единиц на обоих входах ЛЭ  $D_4$   $\overline{R}_2 = 0$ , и триггер  $T_2$  переводится в нулевое положение. Из рассмотренного следует, что при

парафазной организации целей занесения информации отпадает необходимость в предварительном обнулении триггеров регистра.

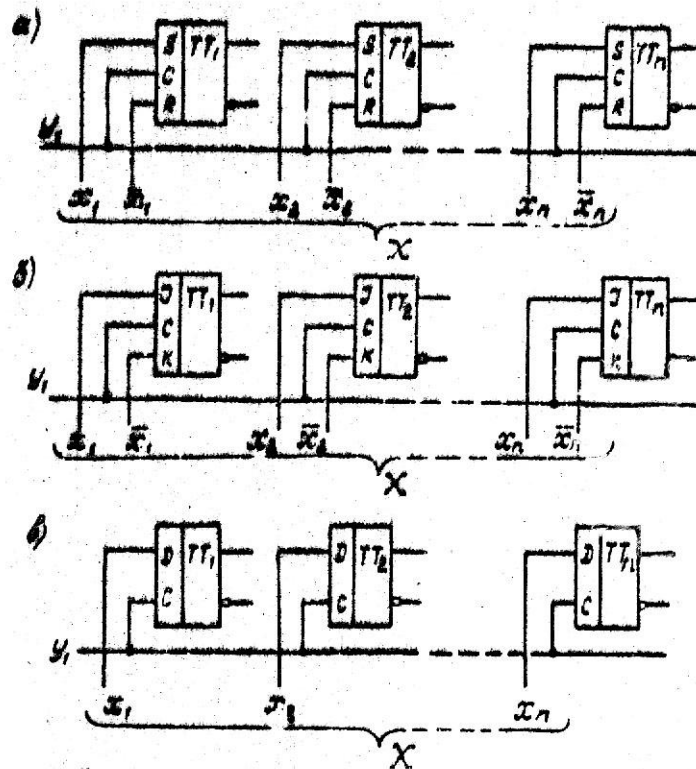


Рис. 2. Накопительные регистры на универсальных триггерах

### 8.1.2 Накопительные регистры на универсальных триггерах.

Наличие синхровхода у универсальных триггеров позволяет до предела упростить цепи занесения информации в регистры (рис. 2.б). В этих схемах управляющий сигнал  $Y_1$ , по которому осуществляется занесение информации (3), поступает на синхронизирующие входы (С) триггеров, причем предварительного обнуления регистра производить не требуется. Как видно из схем, приведенных на рис. 26, регистры, реализуемые на универсальных RS – и JK – триггерах, требуют подачи парафазных (как прямого так и инверсного  $x_i$  и  $\bar{x}_i$ ) сигналов (рис. 26, а,б). Наибольшей простотой обладают схемы на D – триггерах (рис. 26, в), причем в накопительных регистрах могут быть использованы простейшие D – триггеры типа «защелка» (см. рис. 4, д).

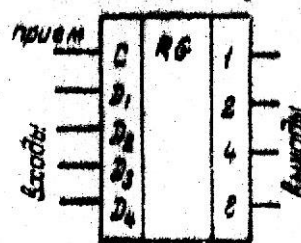


Рис. 3. Условные обозначение накопительного регистра

Условное обозначение четырехразрядного накопительного регистра, построенного на синхронных D – триггерах, показано на рис. 27. Выходы регистра обычно указываются в соответствии с весом соответствующего разряда ( $1, 2, \dots, 2^{n-1}$ ).

## 9 СДВИГАЮЩИЕ РЕГИСТРЫ

### 9.1 Сдвигающие регистры на асинхронных RS-триггерах.

Операция сдвига кода – это перемещение содержимого всех разрядов регистра на один (или в общем случае на несколько) разрядов вправо или влево. В этом случае содержимое разрядов слова, вышедшее из разрядной сетки вправо (или влево) теряется, а в освободившихся разрядах регистра фиксируются нули. По направлению сдвига регистра принято разделять на правосторонние (сдвигающие код слова вправо, т.е. в сторону младших разрядов), левосторонние (сдвигающие код слова влево, т.е. в сторону старших разрядов) и реверсивные, осуществляющие сдвиг информации вправо или влево в зависимости от управляющих сигналов.

Принцип работы сдвигающего регистра. Сдвигающие регистры на асинхронных RS – триггерах принципиально могут быть подразделены на одноктактные и двухтактные. Принцип работы одноктактного сдвигающего регистра, осуществляющего сдвиг информации на один разряд вправо, поясняется схемой, приведённой на Рис. 4. Здесь прямые и инверсные входы ( $Q_i$  и  $\bar{Q}_i$ ) предыдущего триггера ( $T_i$ ) регистра через схемы совпадений ( $D_1, D_2; D_3, D_4; \dots$ ) соответственно связаны с единичным и нулевым входом ( $S_i, R_i$ ) триггера последующего разряда ( $T_{i+1}$ ). На другие входы схем совпадений подаётся управляющий сигнал  $V_3$ , под действием которого и осуществляется сдвиг содержимого регистра на один разряд вправо, т.е.

$$V_3 : RG := R1(RG)$$

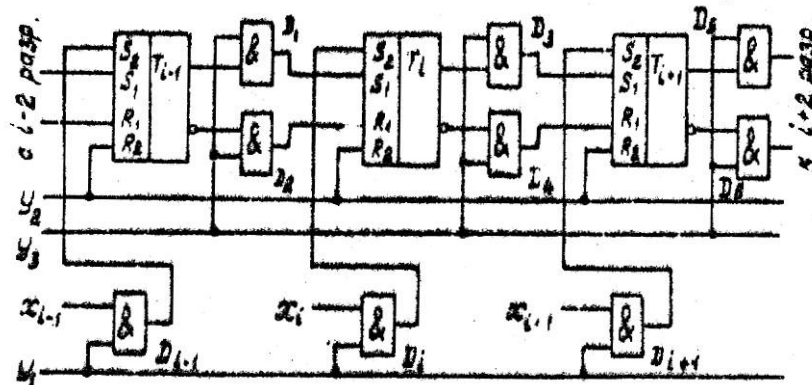


Рис. 4. Принцип построения одноктактного сдвигающего регистра на асинхронных RS-триггерах

Предположим, что в триггер  $T_{i-1}$  была занесена единица, а в триггер  $T_i$  – нуль. Тогда при подаче управляющего сигнала  $V_3$  в результате совпадения единичных уровней на входах ЛЭ  $D_1$  и  $D_4$  на выходах последних формируются сигналы, которые соответственно поступают на входы  $S_1$  триггера  $T_i$  и  $R_1$  триггера  $T_{i+1}$ . Под их действием в триггер  $T_i$  заносится единица, в триггер  $T_{i+1}$  – нуль. Таким образом, после каждого импульса, поступающего на шину  $V_3$ , код слова регистра сдвигается на один разряд вправо.

Для реализации левостороннего сдвигающего регистра, выходы  $Q_i$  и  $\bar{Q}_i$  триггера  $T_i$  необходимо через схемы совпадений соединить со входами  $S_i$  и  $R_i$  триггера  $T_{i-1}$  предыдущего разряда регистра. При построении регистра, осуществляющего сдвиг информации на два (три и т. д.) разряда, выходы  $Q_i$  и  $\bar{Q}_i$  через схемы И подключаются к входам  $S_i$  и  $R_i$  триггера  $T_{i+2}$  ( $T_{i+3}$  и т. д.). Принципы, положенные в основы построения левосторонних регистров сдвига и регистров, осуществляющих сдвиг информации на несколько разрядов, в полной мере справедливы для других схем сдвигающих регистров.



Сигналы  $V_2$  и  $V_1$  осуществляют обнуление (гашение) регистра (2) и занесение информации (3) в параллельном коде.

Принципиальным недостатком одноктного сдвигающего регистра является критичность к длительности управляющего импульса. При большей длительности этого импульса информация может сдвинуться не на один, а на несколько разрядов. С другой стороны, при малой длительности импульса, некоторые триггеры могут не успеть переключиться. Исходя из анализа работы схемы, можно записать следующие условия, обеспечивающие нормальное функционирование регистра

$$t_{u \min} \geq t_{з.ср.} + t_{перекл.}, \quad t_{u \max} \leq 2 \cdot t_{з.ср.} + t_{перекл.},$$

где  $t_{з.ср.}$  и  $t_{перекл.}$  – среднее время задержки распространения сигнала [2] и время переключения триггера соответственно. Так как выполнение этих условий практически неосуществимо, одноктные регистры распространения не получили.

От указанного недостатка свободны двухтактные сдвигающие регистры.

Такой регистра (Рис. 5) состоит из основных ( $T_i - T_n$ ) и вспомогательных ( $T'_i - T'_n$ ) триггеров, причём подразделение триггеров на основные и вспомогательные носит условный характер. Выходы  $Q$  и  $\bar{Q}$  основных триггеров через первую группу схем совпадений ( $D_1, D_2; D_3, D_4; \dots$ ) связаны с входами  $S$  и  $R$  вспомогательных триггеров одноимённых разрядов (т. е.  $T_i$  с  $T'_i$ ); выходы  $Q'$  и  $\bar{Q}'$  вспомогательных триггеров через вторую группу схем совпадений ( $D'_1, D'_2; D'_3, D'_4; \dots$ ) соединены установочными входами ( $S$  и  $R$ ) триггеров последующих разрядов регистра (т. е.  $T'_i$  с  $T_{i+1}$ ). Управляющий сигнал сдвига ( $V_3$ ) расщепляется на два импульса (два такта)  $V'_3$  и  $V''_3$  (Рис. 6). Шина  $V'_3$  подключена ко входам первой группы, а шина  $V''_3$  – второй группы схем совпадений.

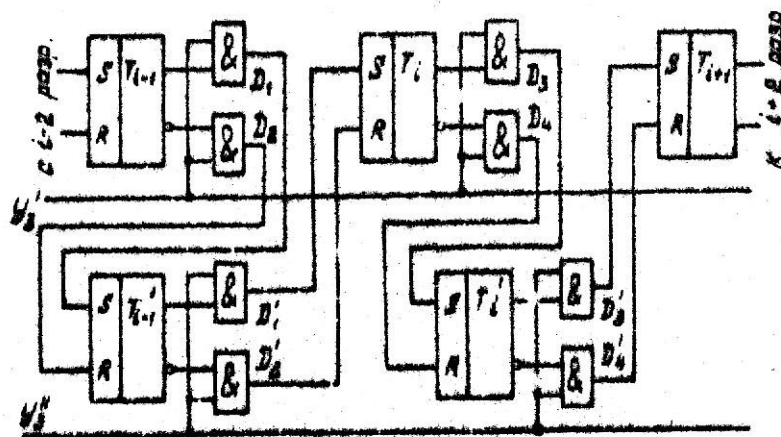


Рис. 5. Принцип построения двухтактного сдвигающего регистра на асинхронных RS-триггерах

Во время действия первого такта ( $V'_3$ ) (интервал  $0-t_1, t_2-t_1, \dots$ ) содержимое основных триггеров через первую группу схем совпадения переписывается во вспомогательные триггеры. При подаче второго такта  $V''_3$  (интервал  $t_1-t_2, t_3-t_2, \dots$ ) информация из вспомогательных триггеров через вторую группу схем совпадения заносится в основные триггеры со сдвигом на один разряд вправо. При этом никаких ограничений на длительность управляющих сигналов сверху не предъявляется. Это объясняется тем, что триггеры, из которых переписывается информация в триггеры другой группы, в течение этого интервала своего состояния не изменяют.

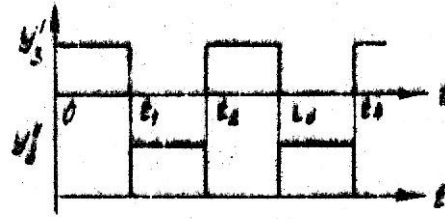


Рис. 6. Временная диаграмма следования управляющих импульсов в двухтактном регистре

Гашение регистра и приём информации в параллельном коде осуществляется также, как и в предыдущей схеме. При этом новую информацию достаточно заносить только в главные триггеры регистра.

При реализации сдвигающих регистров на универсальных логических элементах (ИЛИ–НЕ или И–НЕ) цепи межтриггерных связей принимают вид, показанный на Рис. 7. Если регистр (как триггеры, так и схемы совпадений) выполняется на ЛЭ ИЛИ–НЕ, то схемы связи между выходами и входами соответствующих триггеров регистра выполняются так, как это показано на Рис. 7.а. Здесь сдвиг осуществляется при подаче управляющего сигнала нулевого уровня ( $U_3$ ). Если, например, триггер  $T$  находится в нулевом состоянии, то  $Q_i = 0$ , и при поступлении сигнала  $\bar{U}_3 = 0$  на выходе ЛЭ  $D_1$  вырабатывается единичный сигнал, который, поступая на вход  $R$  триггера  $T_{i+1}$ , переводит последний в нулевое состояние. Если же триггер  $T_i$  находится в единичном состоянии ( $\bar{Q}_i = 0$ ), то при  $\bar{U}_3 = 0$  в результате совпадения нулевых сигналов на обоих входах ЛЭ  $D_2$ , на его выходе устанавливается единичный сигнал. Этот сигнал, поступая на вход  $S$  триггера  $T_{i+1}$ , обеспечивает его установку в единичное состояние.

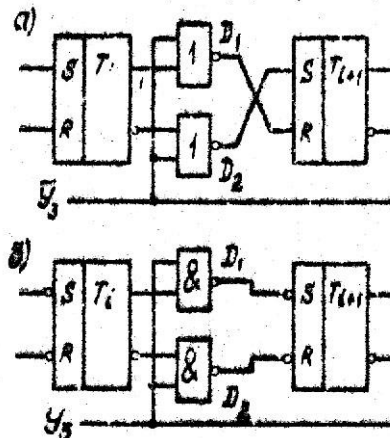


Рис. 7. Цепи межтриггерных связей регистров, построенных на универсальных логических элементах

При реализации регистра на элементах И–НЕ (Рис. 7.б) управляющий сигнал для переключения триггера  $T_{i+1}$  вырабатывается в зависимости от состояния триггера  $T_i$ . На выходе одного из ЛЭ  $D_1$  или  $D_2$  в результате совпадения на их входах единичных сигналов  $U_3$  и  $Q_i$  или  $\bar{Q}_i$ . Если, например,  $Q_i = 1$ , то при  $U_3 = 1$  сигнал на выходе ЛЭ  $D_1$  равен нулю ( $\bar{S} = 0$ ), и в результате его воздействия триггер  $T_{i+1}$  устанавливается в единичное состояние. При  $\bar{Q}_i = 1$ ,  $U_3 = 1$ ,  $\bar{R} = 0$ , и триггер  $T_{i+1}$  переключается в нулевое положение.

Аналогичным образом строятся двухтактные регистры на  $D$ -триггерах типа «защёлка» (Ошибка! Источник ссылки не найден..д), причём, благодаря наличию у них одного информационного входа, межтриггерные связи существенно упрощаются.

### 9.1.1 Сдвигающие регистры на универсальных триггерах.

На основе универсальных синхронных  $RS$ ,  $JK$  и  $D$ -триггеров строятся простые и надёжные одноклапные сдвигающие регистры (Рис. 8).

Схема сдвигающего регистра на универсальных  $RS$ -триггерах представлена на Рис. 8.а. Здесь по сигналу  $\bar{V}_2$  осуществляется обнуление триггеров регистра (2). Занесение информации в параллельном коде (3) производится подачей сигнала  $V_1$ , который поступает на ЛЭ  $D_1 - D_n$ , на другие входы которых подаются информационные сигналы с шины  $X$ . Выходы этих ЛЭ связаны с соответствующими входами  $\bar{S}_d$  непосредственной установке триггеров регистра. Управляющий сигнал  $V_3$ , по которому осуществляется сдвиг информации на один разряда вправо (4), подаётся на синхронизирующие входы триггеров. Если, например,  $Q_{i-1} = 1$ ,  $\bar{Q}_{i-1} = 0$ ,  $Q = 0$ ,  $\bar{Q} = 1$ , то при подаче сигнала  $V_3$  в триггер  $TT_i$  заносится единица, а в триггер  $TT_{i+1}$  – ноль.

Аналогичным образом реализуется сдвигающий регистр на универсальных  $JK$ -триггерах (Рис. 8.б). Принцип работы этого регистра не отличается от принципа работы предыдущей схемы. В сдвигающем регистре на универсальных  $D$ -триггерах (Рис. 8.в) существенно уменьшается количество соединительных шин.

Использование в качестве элементов  $DV$ -триггеров даёт возможность развязать шину управляющего сигнала  $V_3$  от шины синхроимпульсов. Схема сдвигающего регистра на универсальных  $DV$ -триггерах представлена на Рис. 8.г. Здесь при  $V = 0$ , поступление синхроимпульса к изменению состояния триггеров не приводит. Если же  $V = 1$ , то при подаче каждого синхроимпульса происходит сдвиг содержимого регистра на один разряд вправо.

Цепи обнуления и занесения информации для трёх последних схем строятся так же, как и в схеме на  $RS$ -триггерах (Рис. 8.а).

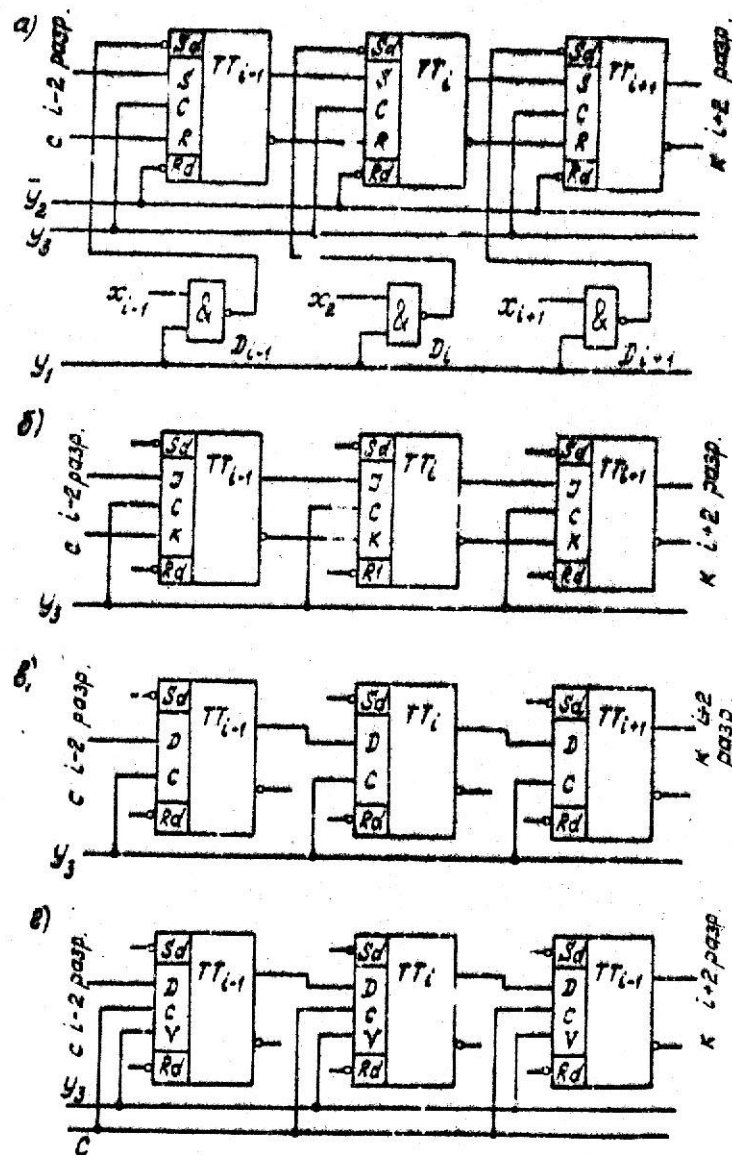


Рис. 8. Сдвигающие регистры на универсальных триггерах

На Рис. 9 приведены схемы сдвигающих регистров, в которых как сдвиг, так и занесение информации осуществляется по синхросигналу. В регистрах на основе универсальных  $RS$ -триггерах (Рис. 9.а, б) сигналы на информационные входы  $R$  и  $S$ -триггеров подаются в парафазном коде, что исключает необходимость в предварительном обнулении регистра. При занесении информации ( $V_1 = 1$ ), если  $x_i = 1$ , сигнал на выходе ЛЭ  $D_1$  (см. Рис. 9.а) имеет нулевое значение и следовательно, на вхожа триггера  $TT_i$   $S = 1$ ,  $R = 0$ . В этом случае при поступлении синхроимпульса ( $C = 1$ ) триггер  $TT_i$  переключается в единичное состояние. Если же  $x_i = 0$ , то на выходе  $D_1$  формируется уровень единицы, а на выходе  $D_3$  - нуля, т.е.  $S = 0$ ,  $R = 1$ . Подачи синхроимпульса приводит к установке триггера  $T$  в нулевое положение.

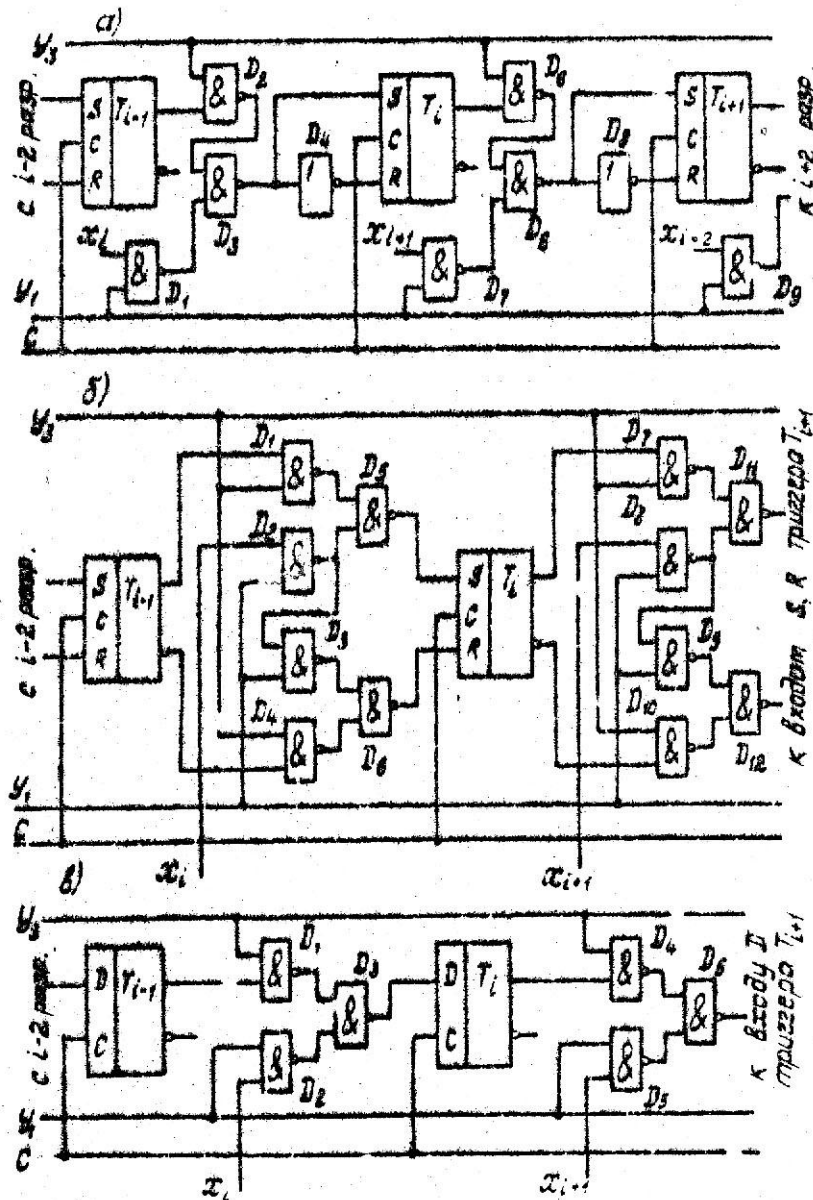


Рис. 9. Сдвигающие регистры на универсальных триггерах с приемом и сдвигом информации по синхросигналу

Сдвиг информации происходит при поступлении сигнала  $V_3 = 1$  **Ошибка! Источник ссылки не найден.** В зависимости от состояния триггера  $TT_{i-1}$  (нулевого или единичного) на информационных входах последующего триггера  $TT_i$  формируется комбинация сигналов  $S = 0, R = 1$  или наоборот  $S = 1, R = 0$  соответственно. В результате поступления синхросигнала в триггер  $TT$  заносится содержимое триггера  $TT_{i-1}$  предшествующего разряда регистра. Разумеется, что, как и для других схем регистров, комбинация сигналов  $V_1 = V_3 = 1$  является недопустимой.

Недостатком рассмотренной схемы является то, что при  $V_1 = V_3 = 0$  под воздействием синхроимпульса все триггера регистра устанавливаются в нулевое положение. Действительно, при  $V_1 = V_3 = 0$  на особых входах ЛЭ  $D_3, D_6$  и т.д. будут действовать единичные сигналы, что приводит к формированию комбинаций  $S = 0, R = 1$  на входах всех триггеров. Поэтому при  $C = 1$  регистр обнуляется.

От этого недостатка свободна схема, приведённая на Рис. 9.б. Здесь для формирования информационных сигналов  $S$  и  $R$  используются сигналы, снимаемые как с единичных, так и нулевых выходов триггеров. Не представляет труда убедиться в том, что при сдвиге ( $V_3 = 1$ ) и при  $Q_{i-1} = 1, \bar{Q}_{i-1} = 0$  на входах  $TT_i$   $S = 1, R = 0$ , а в противоположном случае ( $Q_{i-1} = 0, \bar{Q}_{i-1} = 1$ )  $S = 0, R = 1$ . Поэтому при  $C = 1$  в триггер  $TT_i$  заносится содержимое  $TT_{i-1}$  предшествующего разряда. Таким образом, можно показать, что занесение информации в параллельном коде ( $V_1 = 1$ ), если  $x_i = 1, S = 1, R = 0$  (если  $x_i = 0, S = 0, R = 1$ ), и, когда  $C = 1$ , триггер  $TT_i$  устанавливается в положение, определяемое значением входного сигнала  $x_i$ .

В отличие от предыдущей схемы комбинация  $V_1 = V_3 = 0$  не приводит к обнулению регистра. При такой комбинации управляющих на выходах ЛЭ  $D_1 - D_4$  ( $D_8 - D_{10}$  и т.п.) вырабатываются единичные сигналы, что вызывает формирование нулевых сигналов на выходах ЛЭ  $D_5$  и  $D_6$  ( $D_{11}, D_{12}$  и т.п.), т.е. на выходах всех триггеров  $S = R = 0$ . В этом случае поступление синхроимпульса к изменению состояния триггеров не приводит.

Схемы на  $JK$ -триггерах реализуется таким же образом, что и схемы, приведённые на Рис. 9.а, б. Аналогичная по принципу действия схема сдвигающего регистра на универсальных  $D$ -триггерах показана на Рис. 9.в. При сдвиге информации ( $V_3 = 1$ ) сигнал на выходе  $D$  триггера  $TT_i$  имеет то же значение, что и сигнал на выходе  $Q$  триггера  $TT_{i-1}$ . Когда происходит процесс занесения информации ( $V_1 = 1$ ) сигнал на входе  $D$  соответствует значению сигнала  $x_i$ . Как и в схеме на Рис. 9.а при  $V_1 = V_3 = 0$  поступление синхроимпульса приводит к обнулению регистра. Для установления этого в качестве элементов регистра можно использовать  $DV$ -триггеры, на входы  $V$  которых следует подавать сигнал запрета, вырабатываемые в соответствии с соотношением  $V = V_1 \vee V_3$ . В этом случае при  $V_1 = V_3 = 0$  и  $V = 0$ , поступление синхроимпульса не приводит к изменению состояний триггеров. Условное обозначение четырёхразрядного сдвигающего регистра, реализованного на основе  $D$ -триггеров показано на Рис. 10. Здесь  $V_2$ -функциональный вход младшего разряда регистра (иногда этот вход называют входом приёма последовательного кода), на вход  $V_1$  подаётся управляющий сигнал либо сдвига ( $V_1 = 0$ ) либо приёма ( $V_1 = 1$ ) информации в параллельном коде. Непосредственно сдвиг осуществляется подачей управляющего сигнала на вход  $C_1$ . Запись информации в параллельном коде производится путём подачи информационного слова на входе  $D_1 - D_4$  и управляющего сигнала на вход  $C_2$  (причём). Выходы регистра обозначены в соответствии с весами разрядов регистра.

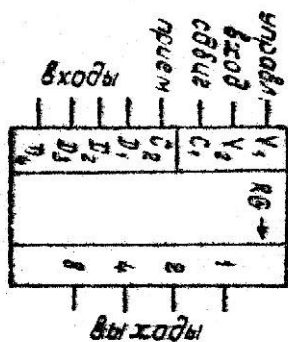


Рис. 10. Условное обозначение сдвигающего регистра

## 9.2 Реверсные сдвигающие регистры.

Равновесный режим работы регистров достигается за счёт использования логических устройств, обеспечивающих формирование информационных сигналов на входах триггера  $TT_i$  в соответствии со значениями выходных сигналов предшествующего триггера  $TT_{i-1}$  (при правостороннем сдвиге), либо последующего триггера  $TT_{i+1}$  (при левостороннем сдвиге) регистра. Так как на структура цепей обмена информации не зависит от типа универсальных триггеров ( $RS$ -,  $J_n$  - или  $D$ -типа), принцип построения реверсных триггеров сдвига рассматривается на примере схемы, реализованной на основе  $D$ -триггеров (Рис. 11). В этой схеме при  $V_3 = 1$  сигнал на вход  $D$  триггера  $TT_i$  поступает с выхода  $Q_{i-1}$  триггера  $TT_{i-1}$  через ЛЭ  $D_1$  и  $D_3$ , вследствие чего при поступлении синхронимпульса ( $C = 1$ ) производится сдвиг слова вправо (4). Если  $V_4 = 1$ , то на вход  $D$  триггера  $TT_i$  через ЛЭ  $D_2$  и  $D_3$  подаётся сигнал с выхода  $Q_{i+1}$  триггера  $TT_{i+1}$ .

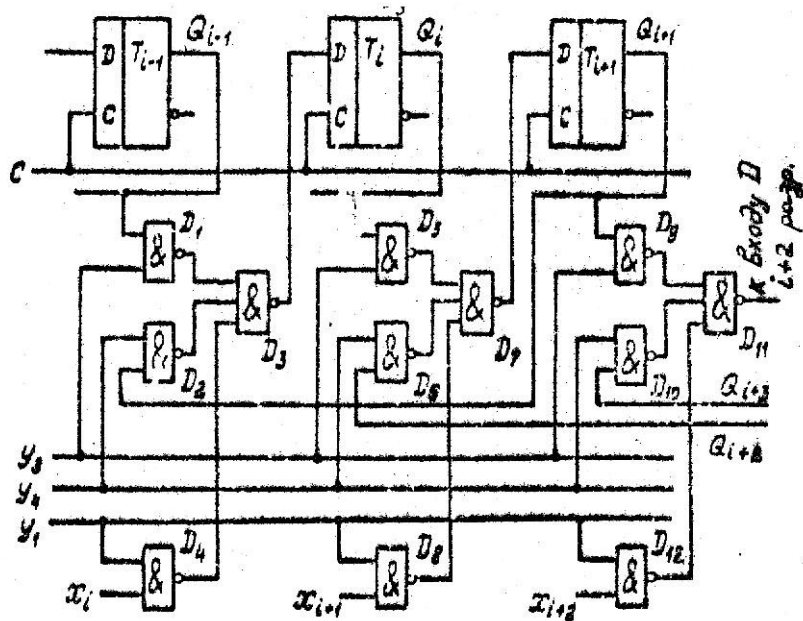


Рис. 11. Реверсивный регистр на основе  $D$ -триггеров

Благодаря этому при  $V_4 = 1$  и  $c = 1$  осуществляется логическая операция сдвига слова влево, иначе говоря  $V_4 : RG := L1(RG)$ . Занесение (приём) информации производится при  $V_1 = 1$  (3). Комбинация  $V_1 = V_3 = V_4 = 0$  приводит к обнулению регистра при  $C = 1$ .

## 10 СЧЕТЧИКИ (В ПРОЦЕССЕ РЕДАКТИРОВАНИЯ)

Счётчиком называется типовой узел ЦВМ, предназначенный для подсчёта числа входных импульсов. Счётчики используются для образования последовательностей адресов команд, для подсчёта количества циклов выполнения операций и т.п. Счётчики могут также выполнять микрооперации приёма и передачи кодов.

Схемы счётчиков можно классифицировать по следующим признакам.

По основанию системы счисления они подразделяются на двоичные и счётчики с произвольным основанием или модулем счёта  $N$ . В двоичных счётчиках модуль счёта равен  $2^n$ , где  $n$  – количество разрядов счётчика, вследствие чего они также называются счётчиком по модулю 2 (mod 2). В счётчиках с произвольным основанием модель счёта, иногда называемый коэффициентом пересчёта ( $K_{сч}$ ), представляет собой число, не кратное степени двух. К этой же группе схем относятся и широко распространенные десятичные (двоично-десятичные) счетчики с  $M = 10$ .

По целевому назначению счетчики принято подразделять на суммирующие, вычитающие и реверсивные. Суммирующие счетчики предназначены для выполнения счета в прямом направлении, т.е. для сложения. С приходом очередного счетного импульса содержимое счетчика увеличивается на единицу. В вычитающих счетчиках при поступлении очередного импульса содержимое уменьшается на единицу. Реверсивными называются такие счетчики, которые предназначены для выполнения операций счета как в режиме сложения, так и в режиме вычитания.

Особую группу составляют счетчики, функционирующие по принципу циклического сдвигающего регистра (так называемые сдвигающие счетчики).

По способу организации операции счета различают синхронные и асинхронные счетчики. В синхронных счетчиках счетные импульсы одновременно воздействуют на синхровходы всех триггеров. В асинхронных счетчиках каждый последующий триггер управляется сигналами, формируемыми в схеме предыдущего триггера, а входные сигналы поступают на счетный вход первого триггера.

В зависимости от способа построения межразрядных связей различают счетчики с последовательным, сквозным, параллельным и комбинированным (групповым) переносом.

### 10.1 Асинхронные двоичные счетчики

Элементами асинхронных счетчиков являются  $T$ -триггеры, реализуемые на основе универсальных RS-, JK- или В-триггеров (рис.37). Для перевода RS-триггера в счетный режим (рис. 37,а) необходимо соединить между собой входы  $S$  и  $R$  с выходами  $\bar{Q}$  и  $Q$  соответственно (см. рис. 13, 15, 16, 18). Счетный триггер на основе JK-триггера (табл.7) реализуется при выполнении условия  $J = K = 1$  (рис. 37,б). Для реализации  $T$ -триггера на основе D-триггера (см. рис. 17, 20) вход  $D$  необходимо соединить с выходом  $\bar{Q}$  (рис. 37,в). Условное обозначение  $T$ -триггера приведено на рис. 12,б.

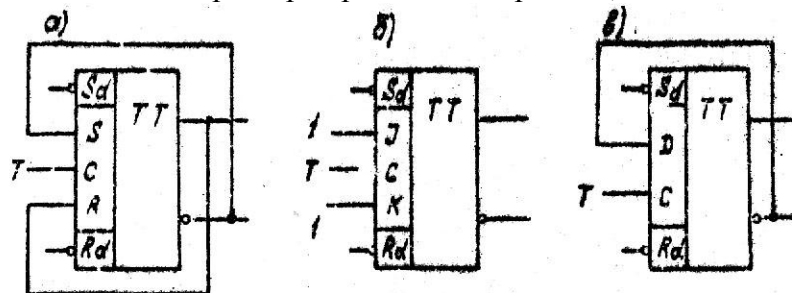


Рис. 12. Организация T-триггеров из универсальных триггеров

В зависимости от вида межразрядных связей различают счетчики с внутренним переносом и счетчики с непосредственными связями.



## 10.1.1 Счетчики с непосредственными связями.

На практике большее распространение получили счетчики, в которых сигналы переноса снимаются с информационных выходов триггеров  $Q$  или  $\bar{Q}$  (так называемые счетчики с непосредственными связями). Функциональная схема такого счетчика представлена на рис. 38,в. При этом предполагается, что она выполняется на универсальных триггерах, реализованных на ЛЭ И-НЕ или И-ИЛИ-НЕ.

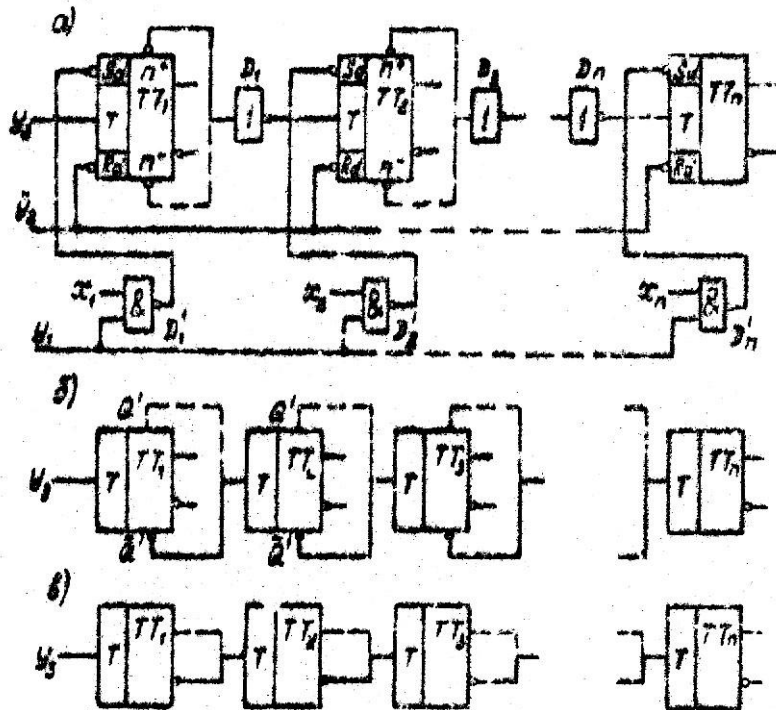


Рис. 13. Асинхронные двоичные счетчики с последовательным переносом

## 10.1.2 Реализация счетчиков на базе MS-триггеров, трехтриггерных ячеек и схем с аналоговыми элементами памяти.

Несмотря на то, что внешний вид схемы не зависит от типа используемых триггеров, принцип работы счетчика, построенного на  $MS$ -триггерах (рис. 13, 15, 16) существенно отличается от принципа работы схемы, реализуемой на основе триггерных ячеек (рис. 18).

Если счетчик реализуется на  $MS$ -триггерах, и при этом сигналы переноса снимаются с информационных выходов вспомогательных триггеров, то при организации суммирующего счетчика счетные входы триггеров  $T$  должны быть связаны с единичными выходами  $Q$  триггеров предшествующих разрядов (пунктир на рис. 38,в). Вычитающий счетчик получается при соединении выходов  $\bar{Q}$  со входами  $T$ , как это показано сплошными линиями.

Справедливость сказанного поясняется временными диаграммами, приведенными на рис. 40,а. Предположим что до момента поступления первого счетного импульса ( $t_1$ ) все триггеры счетчика были обнулены ( $Q_1 = Q_2 = \dots = Q_n = 0$ ). Воздействие первого импульса (интервал  $t_1 - t_2$ ) приводит к переключению триггера первого разряда  $TT_1$  в единичное состояние. При этом сигнал  $Q_1=1$  в соответствии с принципом работы  $MS$ -триггера устанавливается после окончания входного импульса (момент  $t_2$ , см. символы в квадратных скобках на рис. 13, 15, 16). После поступления второго импульса этот триггер возвращается в исходное состояние в момент  $t_4$ .

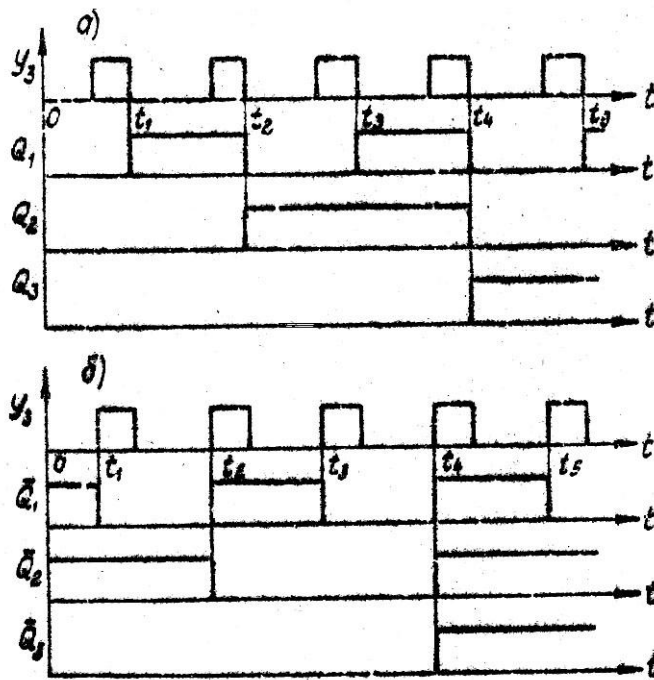


Рис. 14. Временные диаграммы работы счетчиков с непосредственными связями

Из временной диаграммы видно, что импульс, снимаемый с выхода  $Q_1$  триггера  $ТТ_1$ , выполняет функцию счетного сигнала для триггера  $ТТ_2$ . В момент  $t_2$  главный триггер  $ТТ_2$  устанавливается в единичное состояние (символы в круглых скобках), а затем в момент  $t_4$  эта единица переписывается во вспомогательный триггер ( $Q_2=1$ ). Остальные триггеры счетчика ( $ТТ_3 - ТТ_n$ ) функционируют аналогичным образом. Не представляет труда убедиться в том, что для организации режима вычитания сигналы на счетные входы триггеров должны сниматься с инверсных выходов.

Аналогичным образом работают и счетчики, реализуемые на УК- триггерах с аналоговыми элементами памяти (рис. 21, 22). Для устранения неопределенности при предварительной установке сигналы  $S_d$  и  $R_d$  должны подаваться как на главные, так и на вспомогательные триггеры (пунктир на рис. 13 и 15).

В случае реализации счетчика на трехтриггерных ячейках при организации режима сложения сигналы переноса должны сниматься с инверсных выходов триггеров (сплошные линии на рис. 38, в). Пусть до поступления первого счетного импульса все триггеры счетчика обнулены (т.е.  $\overline{Q_1} = \overline{Q_2} = \dots = \overline{Q_n} = 1$ , см. рис. 40,б). В момент подачи импульса ( $t_1$ ) триггер  $ТТ_1$  устанавливается в единичное состояние (см. символы в круглых скобках на рис. 18), уровень сигнала на выходе  $\overline{Q_1}$  становится равным нулю. При этом состояние триггера  $ТТ_2$  не изменится.

Когда на вход  $y_3$  поступает второй счетный импульс (момент  $t_3$ ), первый триггер возвращается в исходное состояние, сигнал на выходе  $\overline{Q_1}$  принимает единичное значение. Этот сигнал, поступая на счетный вход триггера  $ТТ_2$ , переводит его в единичное состояние ( $\overline{Q_2}=0$ ). Триггеры остальных разрядов функционируют аналогичным образом. Достоинством второго счетчика является то, что сигналы на выходах триггеров принимают новое значение по переднему фронту счетного импульса. Установочные сигналы ( $S_d$  и  $R_d$ ) должны подаваться не только на ЛЭ  $D_5$  и  $D_6$  главного триггера, но обеспечивать блокировку ЛЭ  $D_2$  и  $D_3$  (см. пунктир на рис. 18).

Интересно заметить, что режим, подобный режиму работы счетчика на трехтриггерных ячейках (рис. 40, б), можно получить и при реализации счетчика на MS-триггерах. Для этого необходимо связать выходы  $\overline{Q}$  главных триггеров с входами Т, а также изменить цели предварительной установки таким образом, чтобы главный и вспомогательный триггеры устанавливались в противоположные состояния (например, при обнулении главный триггер необходимо устанавливать в нулевое, а вспомогательный - в единичное состояния).

## 10.2 Счетчики с ускоренным распространением переноса.

Рассмотренный выше счетчик построен по классической схеме с последовательным переносом. Недостатком подобных счетчиков является большее время распространения переноса при одновременном переключении нескольких триггеров, причем наибольшее время переноса соответствует тому случаю, когда все триггеры переключаются из единичного состояния в нулевое. Уменьшение времени распространения при использовании схем сквозного или параллельного переноса.

В счетчике со сквозным переносом (рис. 41,а) управляющий сигнал  $y_3$  подается одновременно на счетный вход первого триггера и ЛЭ  $D_1$ , на другой вход которого поступает сигнал с выхода  $Q_1$ . Этот ЛЭ совместно с инвертором  $D_1^1$  осуществляет операцию совпадения, и если  $Q_1=1$ , то сигнал на входе  $ТТ_2$  устанавливается до переключения триггера  $ТТ_1$ . В свою очередь выход ЛЭ  $D_1^1$  связан со входом ЛЭ  $D_2$ , другой вход которого подключен к выходу  $Q_2$  триггера второго разряда и т.д. Эффективность данного метода ускорения распространения переноса зависит от того, насколько суммарная длительность задержки на двух ЛЭ ( $D_1, D_1^1; D_2, D_2^1$  и т.д.) меньше времени переключения триггера. Раньше в целях сквозного переноса использовались диодные схемы совпадения с малым временем задержки, подобные схемы использовались довольно часто. Для современных интегральных схем, где для реализации операции совпадения применяются два последовательно включенных ЛЭ, задержка на которых соизмерима с временем переключения триггера, эффективность этого метода весьма незначительна. Некоторые авторы даже называют подобные схемы счетчиками с последовательным переносом.

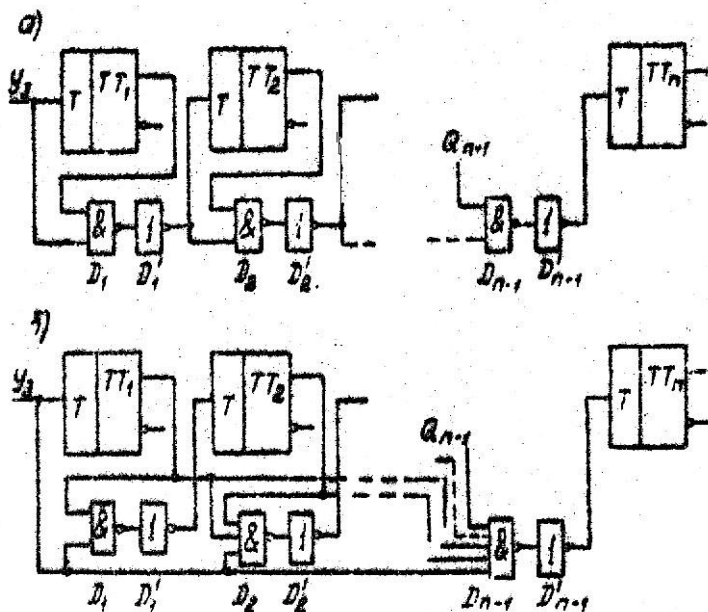


Рис. 15. Асинхронные счетчики со сквозным (а) и параллельным (б) переносом

Следует отметить, что схема, приведенная на рис. 41,а, является классической схемой счетчика со сквозным переносом.

Существенно больший эффект достигается в счетчике с параллельным переносом, схема которого приведена на рис.41,б. Здесь управляющий сигнал подается на все группы схем совпадений ( $D_1, D_1^1; D_2, D_2^1$  и т.д.), а другие их входы связаны с выходами  $Q$  предшествующих триггеров. Поэтому сигналы на счетных входах всех триггеров вырабатываются с задержкой, не превышающей (относительно  $y_3$ ) суммарной задержки двух ЛЭ. Если, например  $Q_1 = Q_2 = 1$ , то при поступлении  $y_3 = 1$  формируются сигналы на выходах инверторов  $D_1^1$  и  $D_2^1$ . Благодаря этому триггеры  $ТТ_2$  и  $ТТ_3$  одновременно с триггером  $ТТ_1$  переключаются в противоположные состояния (в данном случае  $Q_1 = Q_2 = 0$ ). Счетчики с параллельным переносом являются наиболее быстродействующими.

Из схемы счетчика с параллельным переносом видно, что число входов логического элемента совпадения увеличивается с возрастанием порядкового номера триггера. А так как число входов и нагрузкоспособность триггера ограничена, то и разрядность счетчика невелика. Поэтому при большом числе разрядов счетчик разбивают на группы и внутри каждой группы строят цепи параллельного переноса. Перенос между группами реализуется либо методом сквозного, либо последовательного переноса. комбинированным.

В рассмотренных схемах асинхронных счетчиков со сквозным и параллельным переносом из-за разброса длительностей переходных процессов в триггерах возможно возникновение опасных состояний (гонок), когда сигнал переноса, распространяясь по параллельным цепям (через триггеры и схемы совпадений), может вызывать появление ложных сигналов на выходах триггеров. Поэтому наиболее полно достоинства сквозного и параллельного переносов проявляются в счетчиках.

Условные обозначения четырехразрядных двоичных счетчиков показаны на рис.42.

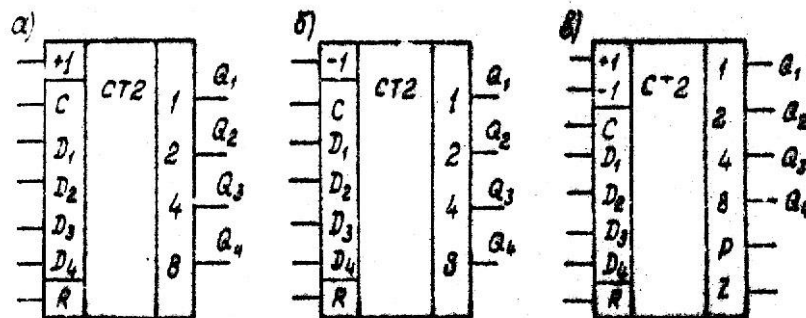


Рис. 16. Условные обозначения суммирующего (а), вычитающего (б) и реверсивного двоичных счетчиков

Здесь символами +I и -I обозначаются входы, на которые подаются счетные сигналы при работе счетчика соответственно в режиме сложения и вычитания; выходы  $Q_1 - Q_4$  указываются в соответствии с весом соответствующего разряда; R - вход обнуления; начальная установка триггеров осуществляется подачей сигналов на входы  $D_1 - D_4$  (или S) по синхроимпульсу, поступающему на вход C. С выходов P и Z (см. рис. 42, в) снимаются сигналы переноса и заема ( $\Pi^+$  и  $\Pi^-$ ). принципы построения реверсивных счетчиков будут изложены ниже.

### 10.3 Синхронные двоичные счетчики.

Преимущества параллельного переноса наиболее полно проявляются в синхронных счетчиках, в которых счетные импульсы ( $y_3$ ) одновременно поступают на синхровходы (C) всех триггеров. Выходы триггеров счетчиков связаны не со счетными входами последующих триггеров, а через логические цепи с их информационными входами. Синхронные счетчики обычно реализуются на основе универсальных синхронных

триггеров  $RS$  - или  $JK$  - типа. Значительно сложнее осуществить организацию синхронных счетчиков на основе  $D$  - триггеров.

Схема синхронного счетчика на  $RS$  - триггерах с параллельным переносом показана на рис. 43,а. Цепи межразрядной связи реализуются по ЛЭ  $D_1 - D_4$ ;  $D_5 - D_8$  и т.д. в соответствии со следующими зависимостями:  $S_1 = \bar{Q}_1$ ;  $R_1 = Q_1$ ;  $S_2 = Q_1 \bar{Q}_2$ ;  $R_2 = Q_1 Q_2$ ; .....  $S_n = Q_1 Q_2 \dots \bar{Q}_n$ ;  $R_n = Q_1 Q_2 \dots Q_n$ ;

Как видно из схемы, первый триггер реагирует на воздействие каждого входного импульса  $U_3$ , второй триггер переключается одновременно с переключением ТТ<sub>1</sub> только при наличии на входах ЛЭ  $D_1$  и  $D_2$  сигнала  $Q_1 = 1$  и т.п. Если, например,  $Q_1 = Q_2 = 1$ ,  $Q_3 = 0$ , то  $S_1 = S_2 = 0$ ,  $R_1 = R_2 = 1$ ,  $S_3 = 1$ ,  $R_3 = 0$ , и при поступлении  $J_3 = 1$  эти триггеры одновременно переключаются в противоположные состояния ( $Q_1 = Q_2 = 0$ ,  $Q_3 = 1$ ).

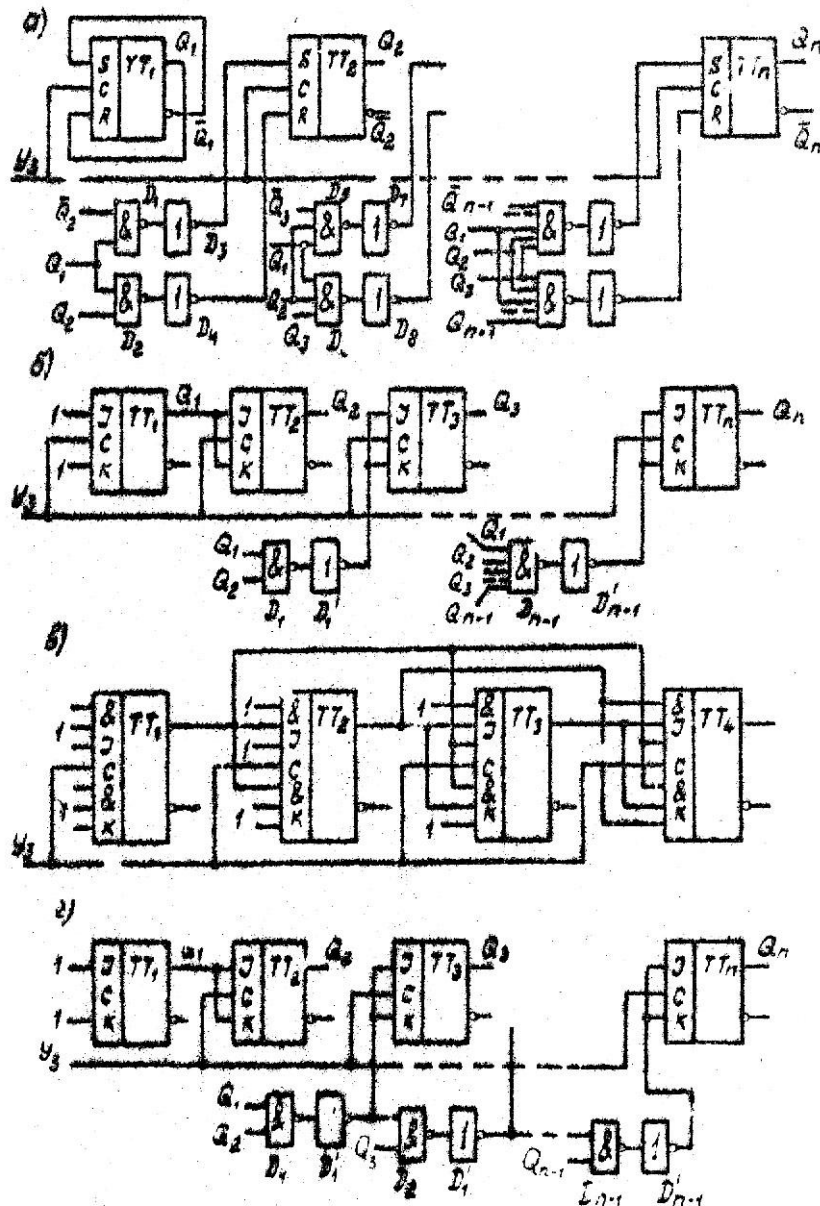


Рис. 17. Двоичные синхронные счетчики

Более удобны для построения синхронных счетчиков JK-триггеры. Учитывая, что для перевода JK-триггера в режим счетного запуска необходимо выполнить условие  $J = K = 1$  (см. табл.7), цепи межразрядной связи счетчика должны обеспечивать выполнение следующих условий:  $J_1 = K_1 = Q_1$ ;  $J_2 = K_2 = Q_1 Q_2$ ; .....  $J_n = K_n = Q_1 Q_2 \dots Q_n$ . Схема

синхронного счетчика на триггерах с параллельным переносом приведена на рис. 43.б. Указанные логические условия реализуются на ЛЭ  $D_1 - D_{n-1}$  и  $D'_1 - D'_{n-1}$ . Если предположить, что триггеры ТТ<sub>1</sub> и ТТ<sub>2</sub> находятся в единичных состояниях ( $Q_1 = Q_2 = 1$ ), то сигналы на входах второго и третьего триггеров  $J_2 = K_2 = J_3 = K_3 = 1$ . В этом случае проступание счетного импульса ( $J_3 = 1$ ) приводит к одновременному переключению всех триггеров в противоположное состояние.

Еще более удобны для построения синхронных счетчиков с параллельным переносом JK-триггеры с несколькими объединенными по И информационными входами J и K. Использование таких триггеров позволяет при ограниченной разрядности счетчика исключить из схемы элементы, осуществляющие операции совпадения. Схема четырехразрядного синхронного счетчика на трехвходовых JK-триггерах представлена на рис 43,в. На неиспользуемые входы J и K должны быть поданы единичные уровни сигналов. Принцип ее работы аналогичен принципу работы предыдущей схемы.

Как видно из приведенных схем повышение разрядности счетчика требует увеличения числа входов схем совпадения (или числа входов J и K). Поэтому в тех случаях, когда быстродействие не имеет решающего значения, находят применение синхронные счетчики с последовательным переносом (рис. 43.г.). Однако более перспективные представления пути организации и многоразрядных счетчиков на основе метода группового переноса.

#### 10.4 Счетчики с произвольным модулем доступа

Для многих устройств ЦВМ необходимы счетчики с модулем счета  $M \pmod{M}$ , отличным от степени двух. В этих счетчиках не полностью используется число возможных состояний, определенное числом каскадов счетчика. Иначе говоря, эти устройства представляют собой n-каскадные с модулем счета (коэффициентом пересчета) от  $M = 2^{n-1} + 1$  до  $M = 2^n - 1$  при  $n > 2$ . Например, при  $n=4$  можно реализовать счетчики с модулем счета от 9 до 15.

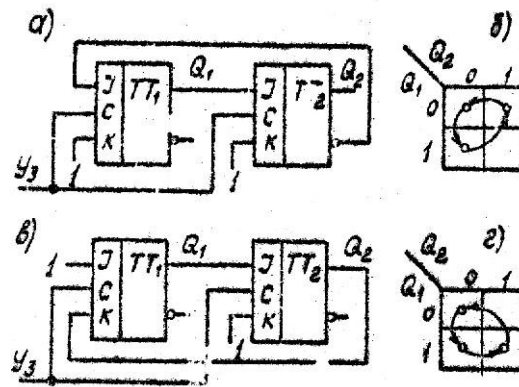
Принцип построения таких схем заключается в исключении “лишних” устойчивых состояний у счетчика с  $M = 2^n$ , т.е. в организации схем, запрещающие некоторые состояния счетчика. Число запрещенных состояний определяется соотношением  $N_{запр} = 2^n - M$ .

В зависимости от того, какие состояния счетчика выбираются в качестве рабочих, все счетчики с модулем счета  $M \neq 2^n$  подразделяются на счетчики с естественным и произвольным порядком счета. В дальнейшем будут рассмотрены только счетчики с естественным порядком счета, как нашедшие наибольшее распространение. Из счетчиков с произвольным порядком счета будут рассмотрены только безвентильные счетчики.

##### 10.4.1 Счетчики с естественным порядком счета.

В схему обычного двоичного счетчика вводятся цепи обратной связи, обеспечивающие переход счетчика в нулевое состояние из состояния  $M - 1$ . Например, счетчик с модулем 13 с приходом каждого счетного импульса последовательно регистрирует состояния от 0 до 12, после чего цикл счета повторяется.

На рис. 44, показана схема синхронного счетчика с модулем счета, равного трем (М3). Схема, в основу которой положен изложенный принцип, реализуется на универсальных JK-триггерах.



**Рис. 18.** Синхронные троичные счетчики

В исходном состоянии схемы ( $Q_1 = Q_2 = 0$ ) сигналы на информационных входах триггеров  $J_1 = 1, K_1 = 1, J_2 = 0, K_2 = 0$ . Поэтому при поступлении первого счетного импульса первый триггер переключается в единичное ( $Q_1 = 1$ ) состояние, а состояние второго триггера остается прежним. В результате сигналы на информационных входах принимают значения  $J_1 = 1, K_1 = 1, J_2 = 1, K_2 = 1$ , и подача второго импульса приводит к переключению обоих триггеров ( $Q_1 = 0, Q_2 = 1$ ), после чего  $J_1 = 0, K_1 = 1, J_2 = 0, K_2 = 1$ . Поступление третьего импульса приводит к переключению триггера ТТ<sub>2</sub> в нулевое состояние ( $Q_2 = 0$ ), не изменяя при этом состояние триггера ТТ<sub>1</sub>, т.е. осуществляется переход счетчика из состояния 10 в состояние 00 ( $2 \rightarrow 0$ ). Не менее наглядно происходящие в схеме процессы отражаются в карте переходов счетчика (рис. 44б).

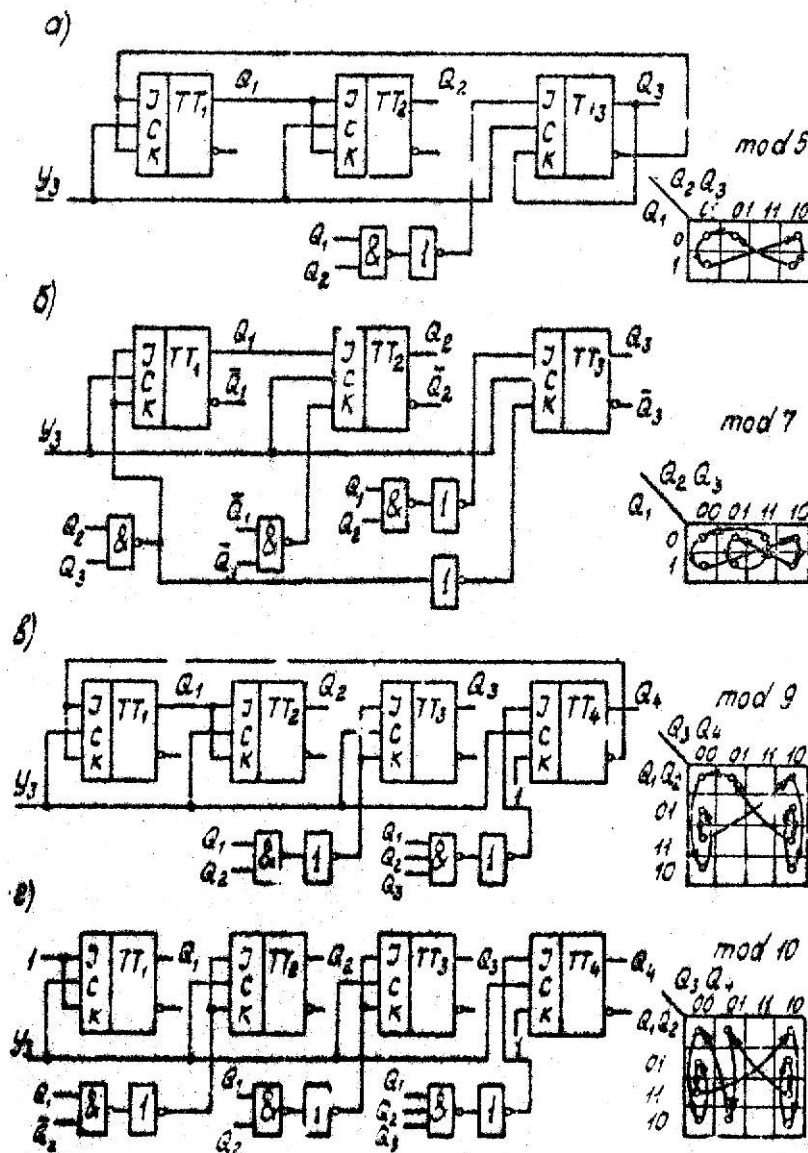


Рис. 19. Синхронные счетчики с модулями счета М(а), М(б), М(в), М(г)

Изложенный метод введения обратных связей, обеспечивающий переключение счетчика из состояния  $M-1$  в нуль, является далеко не единственным. На рис. 44, в продемонстрирована схема троичного счетчика, переключение триггеров которого происходит в следующее последовательности:  $00 \rightarrow 01 \rightarrow 11 \rightarrow 00$  (первыми символами обозначаются состояния триггера ТТ<sub>2</sub>). Карта переходов этого счетчика приведена на рис. 44, г.

На рис. 45 в качестве примеров представлены схемы счетчиков с модулями счета М5, М7, М9 и М10. Рядом с каждой схемой приведена карта переходов соответствующего счетчика.

Иногда желательно иметь счетчики, коэффициент пересчета которых можно изменять путем подачи внешних сигналов. Одна из возможных схем счетчиков (с максимальным числом состояний равным восьми) показана на рис. 46. Если на управляющих входах имеется комбинация сигналов  $X_1 = X_2 = 1$ , то счетчик работает в режиме вычитания с модулем М6, причем его состояния меняются в последовательности 0,7,6,5,4,3,2,1,0,7 и т.д. При комбинации  $X_1 = 1, X_2 = 0$  счетчик осуществляет счет модулю М6, а его состояния меняются так: 0,5,4,3,2,1,0,5 и т.д. Если  $X_1 = 0, X_2 = 1$ , то  $M=4$ , а последовательность состояний 0,3,2,1,0. Последняя комбинация  $X_1 = X_2 = 0$  дает модуль счета М2, причем поочередно меняются состояния 0 и 1.



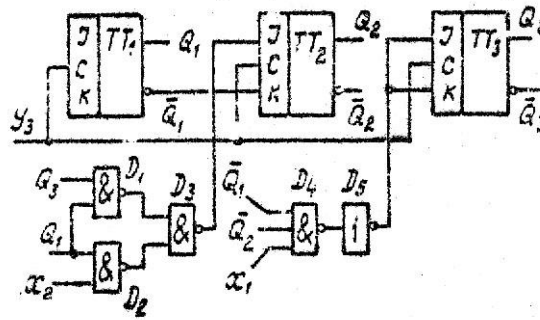


Рис. 20. Счетчик с управляемым модулем счета

Рассмотренные счетчики с произвольным модулем счета были реализованы на основе JK – триггеров, работающих в синхронном режиме. Принципиально счетчики с произвольным модулем могут строиться на основе асинхронных триггеров (рис. 37).

### 10.5 Безвентильные счетчики.

Особую группу схем с произвольным модулем счета представляют так называемые безвентильные счетчики [3,4]. В основе построения таких счетчиков лежит принцип организации счета по модулю  $2^n + 1$ , т.е. на счетчиках, позволяющих увеличивать модуль счета на единицу.

Принцип увеличения модуля счета на единицу можно рассмотреть на примере троичного счетчика (рис. 44,а). Здесь после подачи двух импульсов  $J_1 = J_2 = 0, K_1 = K_2 = 1$ , в результате чего поступление третьего импульса приводит к обнулению обоих триггеров. Таким образом, преобразование одноразрядного двоичного счетчика (триггер ТТ<sub>1</sub>) в троичный осуществляется путем введения дополнительного “единичного” триггера ТТ.

Аналогичный прием используется для увеличения на единицу модуля счета (коэффициента пересчета счетчика СТН с исходным модулем N (рис. 47,а)). Для этой цели необходимо вход J триггера младшего разряда счетчика СТН соединить с выходом  $\bar{Q}$  “единичного” триггера и выход старшего разряда счетчика СТН, со входом J дополнительного триггера. Остальные связи осуществляются также, как и в троичном счетчике (рис. 44,а).

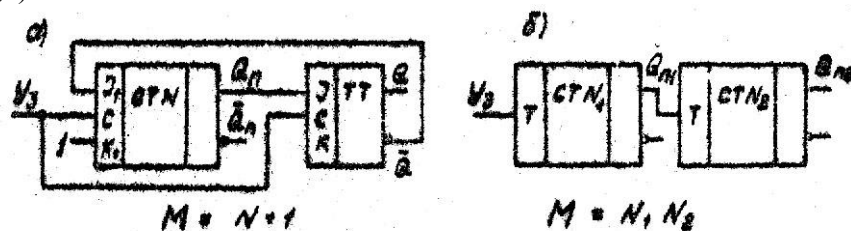


Рис. 21. Схемы, поясняющие принцип построения безвентильных счетчиков

После подачи N-1 входных импульсов на вход J триггера ТТ будет подан единичный сигнал с выхода СТН. С приходом N-го импульса счетчик СТН обнуляется, а триггер ТТ устанавливается в единичное состояние. Следовательно, сигналы на входах J принимают нулевые значения. Поступление (N+1)-го импульса возвращает “единичный” триггер в нулевое состояние, не изменяя состояния счетчика СТН (он был обнулен заранее).

Прием, позволяющий увеличить модуль счета на единицу, позволяет реализовать весьма экономичные счетчики без использования дополнительных логических схем. На рис. 47, демонстрируется принцип построения счетчика с произвольным модулем  $M = N_1 N_2$ .

На рис. 48 приведено несколько схем безвентильных счетчиков. Десятичный счетчик (рис. 48,а) организован на двух последовательно включенных счетчиков с модулем счета

M2 на триггере ТТ и M5 ( $5 = 4+1$ ) на триггерах ТТ<sub>2</sub> – ТТ<sub>4</sub>. Функцию “единичного” триггера выполняет триггер ТТ<sub>4</sub>. Счетчик с M11 (рис. 48,б) образован путем подключения у предшествующей схеме еще одного единичного триггера (ТТ<sub>5</sub>). Счетчик с M12 (рис. 48,в) разбит на две группы: первая – на триггерах ТТ<sub>1</sub> и ТТ<sub>2</sub> (“единичный” триггер ТТ<sub>2</sub>) осуществляет деление на три, вторая – на триггерах ТТ<sub>3</sub> и ТТ<sub>4</sub> – деление на четыре. Счетчик по модулю 13 реализуется путем подключения к счетчику с M12 дополнительного “единичного” триггера (рис. 48,г).

### 10.6 Десятичные счетчики.

Вследствие широкого распространения и возможности использования в логических подсистемах, эти счетчики обычно рассматриваются как специальные. Они получаются на основе обычных двоичных счетчиков с 16 состояниями, шесть из которых должны быть исключены. В десятичных счетчиках чаще всего используется натуральный двоичный код 8421, хотя иногда строятся счетчики в коде 4221, 2421, в коде с избытком 3 и т.д.

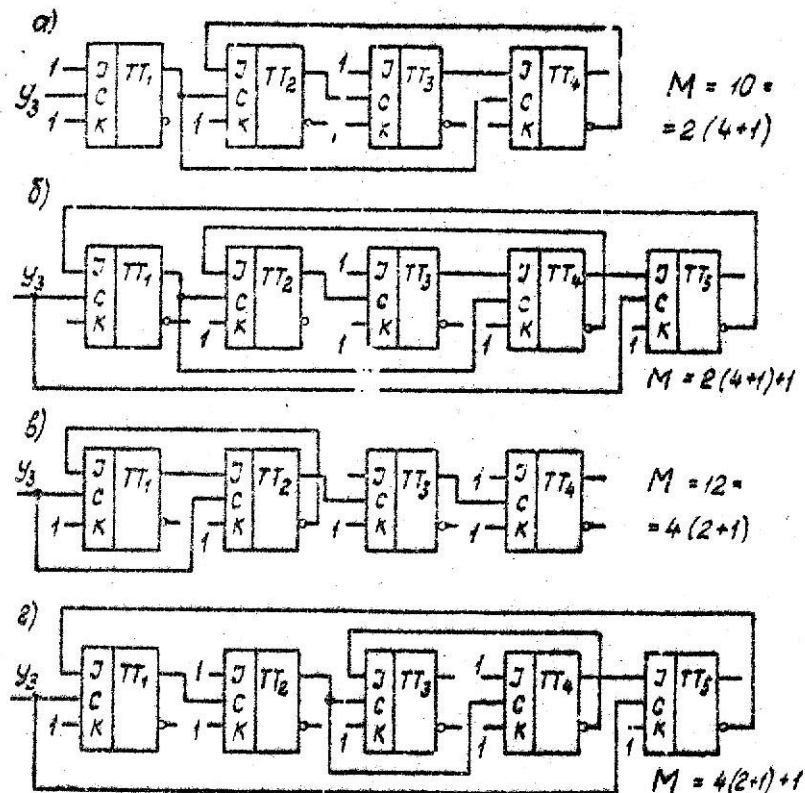


Рис. 22. Безвентильные счетчики с модулями M10(а), M11(б), M12(в), M13(г)

Основная схема суммирующего десятичного счетчика была приведена на рис. 45,г вместе с соответствующей картой переходов.

При наличии JK-триггеров с большим числом входов требуемые логические соотношения реализуются непосредственно на входах триггеров (рис.49,б). На том же рисунке приведена схема десятичного синхронного счетчика с последовательным переносом (рис.49,а).

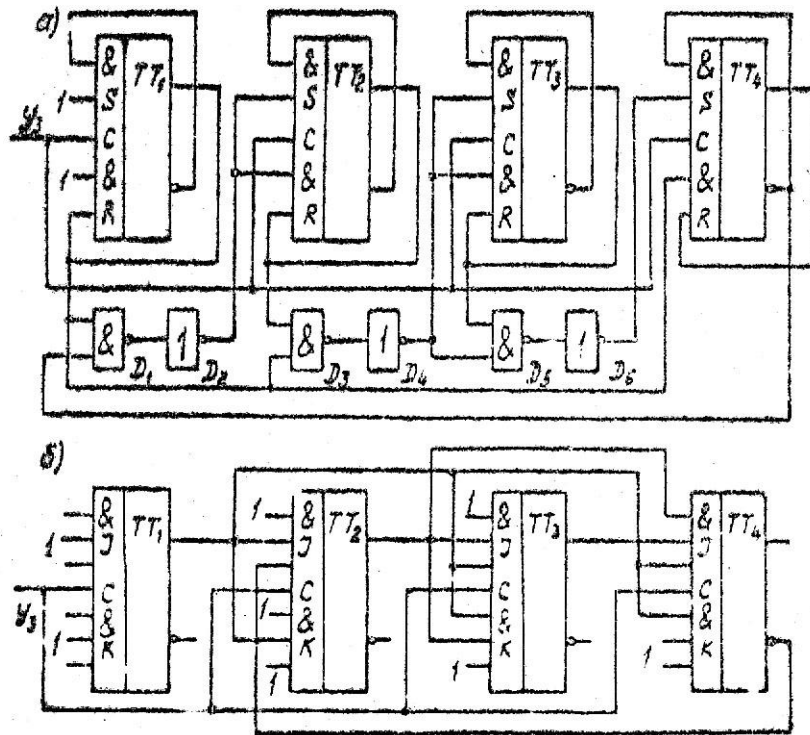


Рис. 23. Синхронные десятичные счетчики с последовательным (а) и параллельным (б) переносом

При жестких требованиях к быстродействию десятичные счетчики могут выполняться асинхронными. Такие счетчики, реализованные на основе JK и D – триггеров, показаны на рис. 50 а ,б соответственно. Логические цепи обеих схем выполнены таким образом, что счетчики до девяти осуществляют счет в натуральном двоичном коде, а с приходом десятого импульса все триггеры оказываются в нулевом состоянии.

Схема безвентильного десятичного счетчика была приведена на рис. 48, а.

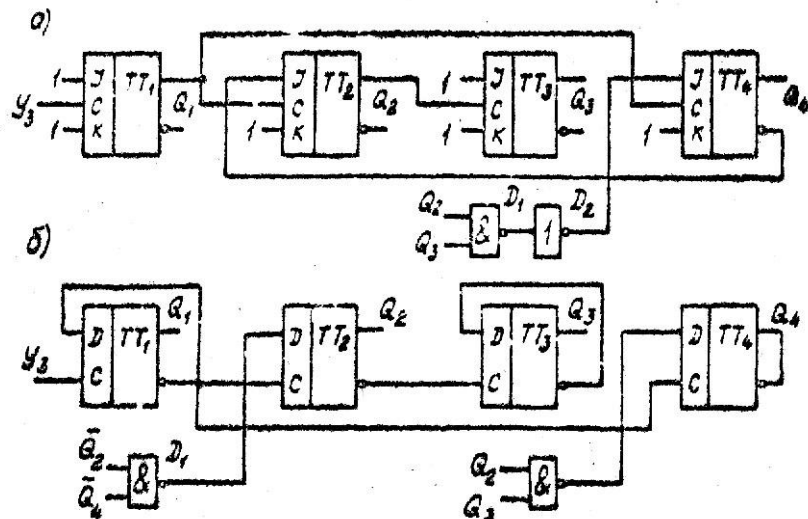


Рис. 24. Асинхронные десятичные счетчики

Условное изображение десятичного счетчика такое же, как к двоичного, только в главном поле (рис.42) указываются символы СТ10 вместо СТ2.

## 10.7 Реверсивные счетчики

Реализация реверсивного режима работы счетчиков заключается в том, что в зависимости от сигналов управления очетные импульсы либо свдадываются с содержимым счетчика, либо из него вычитаются.

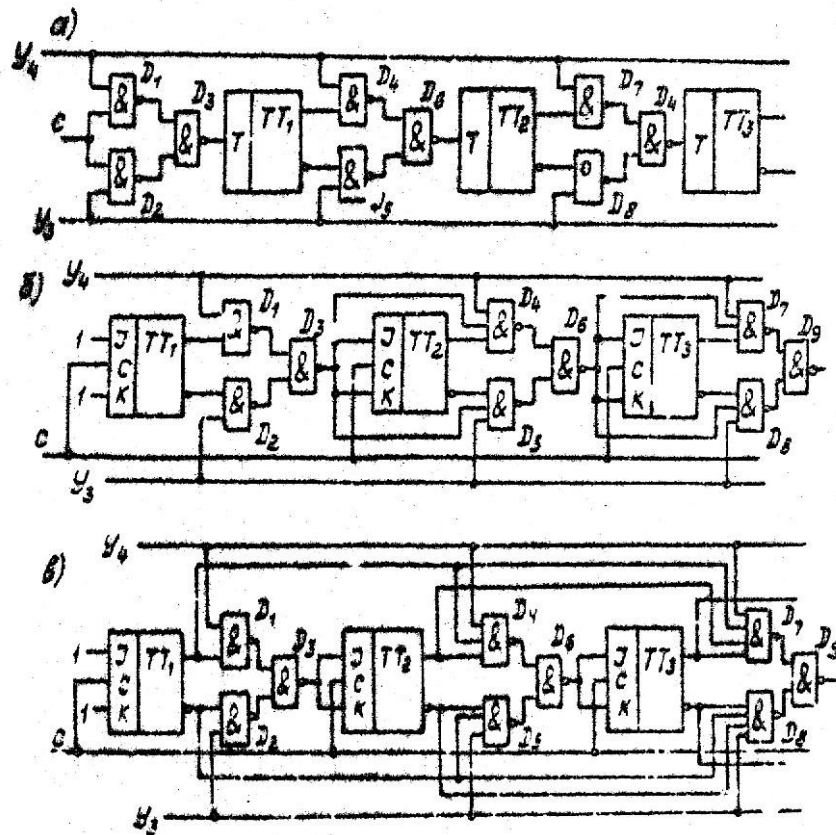


Рис. 25. Реверсивные счетчики

Схема асинхронного реверсивного счетчика с последовательным переносом представлена на рис.51, а. Здесь при  $Y_3 = 1$  и  $Y_4 = 0$  входы Т-триггеров подключаются к инверсным выходам триггеров предшествующих разрядов, т.е. осуществляется режим суммирования синхроимпульсов:  $y_3: CT = CT + 1$ . В противном случае, когда  $Y_4 = 1$  и  $Y_3 = 0$  счетчик работает в режиме вычитания:  $Y_4: CT = CT - 1$ . При  $y_3 = y_4 = 0$  содержимое счетчика не изменяется. Очевидно, что в реверсивном счетчике цепь последовательного переноса увеличивается на два ЛЭ в каждом разряде, поэтому максимальная частота следования счетных импульсов снижается.

На рис.51,б приведена схема синхронного реверсивного счетчика с последовательным переносом. Логические элементы  $D_1, D_2, D_3$  в первом разряде и аналогичные им в других разрядах образуют схему реверса. Здесь при  $Y_3 = 1$  входы  $J_i$  и  $K_i$  подключаются к выходам  $Q_{i-1}$  (режим сложения), а при  $Y_4 = 1$  - к выходам  $\bar{Q}_{i-1}$  (режим вычитания) триггеров предшествующих разрядов. Принципиально так же работает реверсивный счетчик с параллельным переносом (рис.51,в).

Условное обозначение реверсивного счетчика с последовательным переносом и отдельными импульсными и входами для работы на суммирование (+1) и вычитание (-1) показано на рис.42, в. Символы P и Z обозначают выходы сигналов переноса и заема.

## II. СХЕМОТЕХНИКА МИКРОПРОЦЕССОРНЫХ СИСТЕМ.

### 11 ОСНОВНЫЕ ПОНЯТИЯ ОБЛАСТИ МИКРОПРОЦЕССОРНЫХ СИСТЕМ

Вычислительная система (ВС) в широком смысле – техническая система, осуществляющая хранение, обработку и передачу данных с целью реализации различных прикладных задач: расчетных, управления, коммуникационных и других. Ключевым элементом (часто единственным) вычислительной системы является ЭВМ.

Более узкое понятие ВС относится к классификации «вычислительные машины – комплексы – системы – сети».

Центральный процессор (часто, просто «процессор») – центральный блок ЭВМ, осуществляющий реализацию алгоритмов обработки данных. В настоящее время основным типом процессоров являются цифровые программно управляемые процессоры. Такие процессоры построены на базе цифровых электронных схем и используют для обработки данных методы булевой алгебры и двоичной арифметики, реализуемые встроенными двоичными арифметическо-логическими устройствами (АЛУ). Процессоры данного типа имеют сложную, иерархическую классификацию по *архитектуре*, специализации, конструктивно-технологическим параметрам и по иным признакам.

Универсальный процессор – процессор, имеющий архитектуру, набор структурных блоков, систему команд и конструктивно-технологическое исполнение, позволяющее одинаково эффективно применять его для решения достаточно широкого круга разнотипных задач и использовать в различных условиях.

Специализированный процессор – процессор, у которого особенности архитектуры, набора структурных блоков, системы команд или конструктивно-технологического исполнения, позволяют значительно повысить эффективность решения достаточно узкого круга специальных задач по сравнению с иными применениями.

Микропроцессор – процессор, реализованный в виде одной (сверх)большой интегральной схемы (БИС или СБИС), включающий подавляющее большинство или все блоки и подсистемы процессора. Кроме того, существуют блочно-секционные процессоры, собираемые из нескольких одинаковых микросхем - секций. За счет этого достигается произвольная разрядность процессора. К блочно-секционным процессорам относится, например, микропроцессорный комплект К1804.

Микропроцессорная техника – совокупность микропроцессорных элементов: микропроцессоров, ОКМЭВМ и вспомогательных микросхем и блоков, ориентированных на использование в совокупности с микропроцессорами, а также множества методов и средств проектирования и реализации систем на базе микропроцессорных элементов.

Микропроцессорная система – вычислительная система, реализованная на базе элементов и методов микропроцессорной техники.

микроЭВМ – ЭВМ, центральный процессор которой относится к классу микропроцессоров, то есть, собран на одной (реже, на нескольких) СБИС.

Однокристалльная микроЭВМ (ОКМЭВМ) – СБИС, которая содержит все функциональные узлы, необходимые для обеспечения автономной работы в качестве ЭВМ. На кристалле ОКМЭВМ располагаются: процессор, блоки постоянной и

оперативной памяти (ПЗУ и ОЗУ), блоки ввода-вывода, блоки управления и синхронизации и, возможно, некоторые другие блоки.

Микроконтроллер – ОКМЭВМ, ориентированная на функции управления (контроля) некоторой физической системой или объектом. Соответственно микроконтроллер будет являться ядром *управляющей (вычислительной) системы*, часто встраиваемой в конструкцию управляемого ей объекта и, в этом случае, называемой *встроенной вычислительной системой (ВВС)*. Управляющие системы и, соответственно, микроконтроллеры в основном ориентированы на исполнение к конечной системе фиксированного набора функций, не изменяемого в течении существования этой системы. Система на кристалле (СнК или SoC (System on Chip)) – ОКМЭВМ, ориентированная на решение широкого круга задач различного типа и, в связи с этим, обладающая значительными вычислительными ресурсами. СнК является однокристалльным вариантом универсальной ЭВМ и ориентирована на использование схожих программных средств и методов проектирования ВС.

Архитектура вычислительной системы – это совокупность ключевых решений по реализуемым системой алгоритмам и методам исполнения алгоритмов, обработки и хранения данных, по организации взаимодействия со смежными компьютерными системами, физическим миром и человеком, по структуре и используемым технологиям производства аппаратных средств, а также по иным вопросам, ключевым образом влияющим на качество проектирования и использования вычислительных систем.

Схемотехника – область науки и техники изучающая принципы, методы функционирования, а также технологии проектирования схем электронных устройств на основе типовых (базовых) функциональных блоков: от дискретных электронных компонентов до активных и пассивных усилительных, интегрирующих, генераторных и других аналоговых каскадов, логических элементов, элементов памяти и других. Схемотехника является разделом электроники.

Схемотехника микропроцессорных систем или микропроцессорная схемотехника изучает функционирование и проектирование электронных схем и устройств для микропроцессорных систем.

## 12 СТРУКТУРА МИКРОПРОЦЕССОРНЫХ СИСТЕМ.

Ниже представлена структурная схема простой микроЭВМ и описано назначение основных блоков.

### 12.1 Магистрالی (шины)

#### 12.1.1 Магистрально-модульная организация микропроцессорных систем.

Структура большинства современных микропроцессорных систем является *магистрально-модульной*: различные функционально-завершенные блоки (микропроцессор, блоки памяти различного типа, контроллеры периферийных устройств и коммуникационные и т.п.) подключены к общим магистралам (шинам), посредством которых взаимодействуют между собой, образуя единую вычислительную систему. Передача информации и синхронизирующих сигналов между различными группами функциональных блоков выполняется в режиме разделения времени, то есть поочередно.

*Магистраль (или шина)* – совокупность сигналов, соединяющих несколько функциональных блоков ЭВМ. Это понятие охватывает как электрические линии (собственно «шина»), так и протоколы взаимодействия блоков по ним. Т.е. под понятием «шина» понимается более широкое понятие «шинный интерфейс».

Шины бывают *параллельными* – управляющие сигналы и/или биты информации передаются по отдельным проводам одновременно, и *последовательные* – сигналы и биты передаются по одним и тем же проводам, но последовательно друг за другом во времени.

Наиболее типичной является трехшинная архитектура микропроцессорных систем с *шинами адреса (ША или AB (Address Bus)), данных (ШД или DB (Data Bus)) и шиной управления (IE или CB (Control Bus))*.

В простых системах присутствуют только эти три основные шины. В сложных системах обычно существует иерархия шин: есть несколько шин, различных по функциональному назначению (шина памяти, шина внешних устройств, системные шины, периферийные шины и т.п.). Каждая из этих шин в свою очередь подразделяется на шины адреса, данных и управления.

Наряду с «трехшинными» интерфейсами в ЭВМ используются и другие типы. Например, двухшинные интерфейсы, где шины адреса и данных заменены на одну мультиплексированную шину адреса-данных. Или на последовательную шину, где управляющие команды, адреса и данные передаются по одним и тем же проводам.

Далее будет рассмотрена «классическая» трехшинная организация.

## 12.2 Шина адреса.

По шине адреса процессор указывает, к какой ячейке памяти или регистру он хочет обратиться, чтобы выполнить операции записи или чтения. Сигналы шины адреса (адресные сигналы) формирует микропроцессор на основе адресов, указанных в исполняемых машинных командах. Адресуемыми элементами будут являться ячейки системной памяти или регистры (данных, управления) различных контроллеров (системных, периферийных), подключенных к шине. Соответственно для процессора адресные сигналы и порты микропроцессора будут являться *выходными*, а для адресуемого устройства (памяти, периферийных контроллеров и т.п.) – адресные сигналы и порты будут *входными*. Таким образом, для шины адреса характерна *однонаправленность*.

В сложных системах источниками адреса, кроме процессора, могут быть некоторые системные контроллеры, например, контроллер прямого доступа к памяти (ПДП или DMA (Direct Memory Access)). Для многопроцессорных систем шина адреса может быть разделяемой между несколькими процессорами. Если рассматривается периферийная шина, то в роли адресуемого блока будет выступать контроллер периферийной шины, а подчиненными устройствами будут выступать периферийные устройства. Во всех подобных случаях разделение шины адреса выполняется по времени (сначала адресует память один процессор, потом другой процессор или системный контроллер).

Шина адреса на схемах обозначается **A[n:0]**. Отдельные сигналы (провода) шины адреса обозначаются **A<sub>x</sub>** или **A[x]** ( $x=0..n$ ). Число **x** обозначает номер двоичного разряда адреса, передаваемого по шине. Разрядностью (или шириной) шины адреса называется количество сигналов (разрядов) шины равное **(n+1)**. При такой разрядности обеспечивается доступ к ячейкам памяти по адресам от 0 до  $(2^n-1)$ . Множество всех адресов, которые можно передать по шине называется *адресным пространством шины*. У параллельной шины с  $(n+1)$  линиями адреса адресное пространство составит множество от 0 до  $(2^n-1)$ .

Каждая ячейка памяти может иметь различную разрядность ячейки  $N_b$ : ячейка может быть или 8-разрядной (один байт) или иметь большую разрядность, например, 4 байта = 32 бита. Это не изменяет адресного пространства, но увеличивает объем адресуемой памяти, который в байтах рассчитывается как  $2^n * N_b$ . Когда описывают память ЭВМ часто указывают именно данное выражение –  $(2^n * N_b)$ , которое отражает объем и организацию памяти. Например,  $(2K * 16)$  – это 2048 слов по 16 бит = 4 Кбайт;  $(8M * 32)$  – это  $(8 * 2^{20})$  слов по 4 байта = 32 Мбайт.

### 12.3 Шина данных.

По шине данных передаются данные между всеми устройствами, подключенными к этой шине. От процессора в память передаются сохраняемые результаты выполнения операций. Из памяти в процессор передаются коды команд исполняемой программы и считанные из памяти данные. Между процессором и контроллерами (ввода-вывода, коммуникационными) передаются выводимые и конфигурационные данные в одном направлении - от процессора, а также принятые данные и коды состояния от контроллеров к процессору. Таким образом, шина данных - двунаправленная.

На схемах шина данных обозначается  $D[k:0]$ , а отдельные разряды –  $Dy$  или  $D[y]$ , где  $y$  – номер разряда данных,  $k$  – ширина (разрядность) шины данных.

Так как шина данных двунаправленная, то каждое устройство, подключенное к ней должно переводить порты шины данных на ввод или вывод в зависимости от операции (чтение, запись), выполняемой этим устройством в данный момент времени. Для активных устройств – процессора – такое переключение выполняется по собственной инициативе, в зависимости от типа исполняемой команды (чтение, запись) и этапа выполнения команды (чтение кода команды, исполнение команды). Для пассивных (подчиненных) устройств на шине, например, для микросхем памяти, перевод шины данных на ввод или вывод выполняется под контролем специальных управляющих сигналов из состава шины управления.

Из-за того, что (в отличие от шины адреса) к шине данных подключены несколько источников (процессор, микросхемы памяти, контроллеры ввода-вывода и т.п.), то на время вывода данных одним из них, остальные должны быть синхронно переведены в режим считывания данных с шины или вообще отключены (переведены в режим высокого выходного сопротивления (так называемое Z-состояние или третье состояние). Если переключение направления будет выполняться не одновременно (не синхронно), то в какие-то моменты времени на шине окажется два источника работающих одновременно, что приведет к электрическому замыканию их портов шины данных и возникнут сбои или даже физическое «перегорание» портов данных. Чтобы обеспечить синхронность переключения, должен быть один «центр управления» шиной. В однопроцессорных системах таким центром является процессор, который формирует управляющие сигналы для всех подчиненных устройств. В сложных многопроцессорных системах или системах с интеллектуальными контроллерами, самостоятельно работающими с памятью по шине данных, для управления шиной данных (и вообще всеми шинами в вычислительной системе) используют выделенный блок – контроллер шины.

### 12.4 Мультиплексированные шины адреса-данных.

С целью экономии ножек микропроцессора и числа сигналов шин адреса и данных на электронной плате ЭВМ, иногда используют мультиплексированную шину адреса/данных. Это означает, что по одним и тем же проводам последовательно (с разделением во времени) передаются и адреса, и данные. Например, для доступа к микросхеме памяти 64Кслов \* 16 бит требуется 16-ти разрядная шина адреса  $A[15:0]$  и 16-ти разрядная шина данных  $D[15:0]$  – всего 32 линии. Если выполнить мультиплексирование адреса и данных, то необходима только 16-ти разрядная мультиплексированная шина  $A/D[15..0]$ . При этом доступ к памяти будет медленнее до 2-х раз из-за появления двух этапов: 1) запись адреса; 2) запись/чтение данных.

Чтобы организовать доступ со стороны микропроцессора с мультиплексированной шиной адреса-данных к микросхеме памяти с отдельными шинами (большинство микросхем памяти выпускаются с отдельными шинами) необходимо использовать дополнительный регистр-защелку адреса, который сохраняет адрес на время, пока шина используется для передачи данных.

Кроме регистра используется дополнительный управляющий сигнал AS (Adress Select) (или ALE (Adress Latch Enable)), который указывает подключенным к шине схемам, в какие моменты времени на шине передается адрес, а в какие - данные.



Цикл доступа к памяти с мультиплексированием адреса/данных на шине выполняется в следующем порядке:

1. В начале сигнал AS находится в неактивном состоянии. На шину A/D выставляется адрес ячейки памяти, к которой происходит обращение;
2. После некоторой задержки, необходимой, чтобы установленный адрес успел записаться в подключенный к шине регистр-защелку адреса, формируется активный уровень сигнала AS. По переключению состояния AS регистр-защелка прекращает считывать значение с мультиплексированной шины A/D, а на выходах, которые подключены к адресным входам микросхем памяти, удерживается ранее сохраненный адрес.
3. Сразу после перехода AS в активное состояние на шину адреса-данных выставляются данные (для записи в память), или эта шина переводится процессором на ввод (режим чтения из памяти). После этого выполняется запись или чтение данных в/из памяти по адресу, удерживаемому на выходе регистра-защелки.

Мультиплексирование может охватывать не всю шину адреса, а только ее часть. Например, у простейшего 8-ми разрядного микроконтроллера i8051 мультиплексируется только 8-мь разрядов данных [D7:0] с 8-ю младшими разрядами шины адреса A[7:0], а для старших разрядов адреса, начиная с A8, предусмотрены выделенные сигналы.

## 12.5 Шина управления.

Шина управления – совокупность управляющих и синхронизирующих сигналов различного назначения. Можно различать принципы управления и соответственно шины управления сигнальные и командные.

Сигнальные шины – это набор групп электрических сигналов, каждая из которых поддерживает определенную функцию: управление записью – чтением в память, управление режимами и процессом прямого доступа к памяти (ПДП), сигналы прерывания и т.д. У сигнальных шин управления нет единой функции, подобно шине адреса и данных.

Командные шины (например, PCI) подразумевают передачу кодов команд общей магистрали (по основной шине данных или по выделенной шине команд). Выделенные сигналы управления отдельными функциями используются в ограниченном объеме – только для управления самой шиной управления (например, выделенные синхросигналы шины или сигналы сброса).

Далее будет рассмотрена сигнальная шина и отдельные группы сигналов шины управления.

### 12.5.1 Линии выбора кристаллов – «ChipSelect».

Как было сказано выше, при разработке схем следует внимательно относиться к подключению и работе устройств на шине данных, чтобы не допустить одновременной работы нескольких устройств «на вывод». За синхронное переключение ведомых устройств на шине (микросхем памяти, контроллеров) в режим записи/чтения отвечает процессор – он соответствующим образом формирует управляющие сигналы команд записи, чтения, ожидания и т.п. Однако необходимо еще разделить во времени доступ к нескольким устройствам, подключенным на шину параллельно. Например, на шину может быть подключено несколько микросхем памяти (для получения требуемого объема памяти) и несколько контроллеров ввода-вывода. Логически они различаются диапазоном адресов, закрепленным за каждым из этих устройств. Однако физически на каждое из устройств (памяти, контроллер) подключена одна и та же шина адреса, и каждое устройство воспринимает адреса начиная с 0. Если попытаться считать значение из ячейки с адресом 0, то все микросхемы выставят на шину данных различные – «свои» - значения, в результате чего произойдет конфликт на шине.

Чтобы избежать описанной ситуации каждая микросхема имеет специальный управляющий вход «выбора кристалла» (Chip Select, CS или Chip Enable, CE) или сигнал «разрешение вывода (на шину данных)» (Output Enable, OE). Только в том случае, если сигнал CS/CE/OE активный микросхема вводит/выводит на шине данных в соответствии с поданными на нее сигналами управления. В противном случае она переводит выходы шины данных в состояние высокого сопротивления и не мешает работе других устройств на шине.

В случае CS/CE микросхема в неактивном режиме отключает не только вывод на шину данных, но и остальные внутренние функции, в том числе запись, а также переходит в режим пониженного энергопотребления. Сигнал OE только отключает вывод на шину, а остальные функции работают без изменения.

Источником сигналов CS/CE/OE будет являться процессор или контроллер шины. Специальный функциональный блок (в процессоре, или в контроллере шины) называемый «Селектор адреса» вырабатывает сигналы CS/CE/OE для каждого устройства на шине из адреса, сформированного процессором.

### 12.5.2 Селектор адреса.

Селектор адреса – обязательная подсистема любой микропроцессорной системы, которая обеспечивает:

- 1) Формирование сигналов «ChipSelect / Выбор кристалла» для выбора одного из нескольких устройств на шине данных с отображением их (устройств) в различные диапазоны адресов.
- 2) Отображение физической памяти (микросхем памяти, контроллеров) в один или несколько заданных диапазонов адресного пространства процессора.
- 3) Управление подсистемой перенастройки параметров шины адреса/данных под конкретное устройство (микросхему), подключенную к шине и доступную в данном диапазоне адресов. К настраиваемым параметрам относятся разрядность шины данных, количество тактов ожидания готовности данных, использование или неиспользование сигнала готовности устройства на шине RDY и т.п.

Обычно селектор адреса – комбинационная схема дешифратора двоичного кода адреса, поступающего от процессора по шине адреса. На выходе получаем сигналы «выбора кристалла». При обращении к ячейке памяти по некоторому адресу, селектор адреса определяет, попадает ли установленный адрес в заданный диапазон адресов. Если попадает, то вырабатывается активный сигнал CS/CE/OE разрешения только этой микросхемы памяти. Остальные микросхемы, также подключенные к шине, но для которых сигналы разрешения остались в неактивном состоянии, не реагируют на команды чтения/записи.

#### **НАПРИМЕР:**

В самых примитивных случаях, если к системной шине подключены всего два адресуемых устройства (микросхемы памяти или контроллеры ввода-вывода), селекция адреса выполняется подключением старшего разряда адреса к сигналу выборки /CS одного устройства и инверсии старшего разряда адреса к сигналу выборки другого устройства.

#### **НАПРИМЕР:**

В более сложных случаях сектор адреса, используя для декодирования кроме сигналов адреса управляющие сигналы, может реализовывать дополнительные функции управления доступом к микросхемам памяти:

- 1) Доступа в различные микросхемы памяти при выполнении различных операций (чтения и записи) по одному и тому же адресу. Для этого селектор адреса должен обрабатывать сигналы выбора операции записи или чтения.
- 2) Функции переключения страниц памяти с целью расширения общего объема памяти без увеличения размера поля адреса в кодах команд процессора. Для этого на вход

селектора адреса подается код страницы памяти (PAGE). Код станицы памяти устанавливается процессором либо через специальные выходы либо через универсальные порты ввода-вывода.

- 3) Функции доступа и переключения активной и теневой страниц памяти. Теневой режим используется для отладочных целей. Механизм аналогичен выбору страницы памяти. Разница в использовании: сигнал выбора теневой страницы формируется не под управлением программы, выполняемой процессором, а внешним устройством, например, от кнопки перевода в теневой режим.

**НАПРИМЕР:**

### 12.5.3 Сигналы управления транзакциями на асинхронной шине.

Управляющие сигналы данной группы используются для асинхронной передачи команд записи-чтения устройствам, подключенным к шине (микросхемам памяти, порт ввода-вывода).

#### 12.5.3.1 Синхронные и асинхронные операции.

Асинхронность исполнения команд подразумевает, что выполнение начинается сразу же после перехода управляющих сигналов в активный уровень, без дополнительных импульсов по линиям синхронизации. Синхронное исполнение операций подразумевает, что их исполнение начинается и завершается строго по фронтам импульсов на специальной (выделенной) линии синхронизации.

Наибольшее распространение в микропроцессорной технике нашли два варианта сигналов управления операциями чтения и записи.

#### 12.5.3.2 Раздельные сигналы управления операциями чтения и записи («вариант Intel»).

В данном варианте, впервые предложенном и реализованном в системах фирмы Intel, команды чтения и записи стробируются раздельными сигналами. Такой вариант в настоящее время наиболее популярен, реализован во многих микропроцессорах и почти во всех микросхемах асинхронной памяти (ОЗУ, ПЗУ).

**WR (Write) или WE (Write Enable)** – сигнал стробирования операции «запись» в устройство на шине. Сигнал WR также формируется процессором – для него это выходной сигнал. При установке активного уровня данного сигнала (обычно это НИЗКИЙ уровень) начинается запись данных установленных процессором на шине данных, в ячейку по адресу, установленному на шине адреса процессора в устройство на шине (в память или в порт ввода-вывода). Данные и адрес должны быть сформированы процессором и выставлены на шины данных и адреса заранее (обычно – за несколько наносекунд). После «снятия» сигнала WR – перевода его в неактивный уровень – запись завершается, можно «убирать» с шины данные и адрес и начинать выполнение следующей шинной операции. Длительность удержания WR в активном состоянии должна быть такой, чтобы устройство на шине – память или контроллер ввода-вывода – успело выполнить данную операцию. Данное время определяется параметрами устройства на шине: для ОЗУ это 3-70 нс., для контроллеров ввода-вывода - 20-200 нс. Если длительность выполнения операции записи значительная и может изменяться в широких пределах (например, длительность записи в репрограммируемую постоянную память РППЗУ может меняться в пределах 0.1-5 мс), то следует использовать механизм контроля готовности (см. ниже описание сигнала RDY).

Следует понимать, что цикл записи на шине не всегда одно и то же, что цикл записи в память или регистр. Алгоритм записи может отличаться для различных устройств на шине: для ОЗУ запись в ячейку памяти происходит непосредственно по активному уровню данного сигнала. Для РППЗУ сначала выполняется запись данных и адреса во внутренние буферные регистры, а потом записывается специальная команда управления записью, запускающая внутренний автомат перезаписи ячеек памяти. Таким

образом цикл записи информации в физическую память разбивается на несколько последовательных циклов записи на шине.

**/RD (Read) или /OE (Output Enable)** – сигнал стробирования операции «чтение» из устройства на шине в процессор. По активному уровню данного сигнала (обычно это НИЗКИЙ уровень, т.е. сигнал «инверсный»), на шину данных выставляются данные из ячейки памяти по адресу, заданному значением на шине адреса.

#### 12.5.3.3 «Объединенные» сигналы управления операциями чтения и записи («вариант Motorola»).

В данном варианте, изначально предложенном и реализованном в системах фирмы Motorola, есть отдельный сигнал выбора типа команды (чтение или запись) и сигнал стробирования исполнения выбранной команды.

**R/nW (Read / not Write)** – сигнал выбора операции «запись» или «чтение» на шине. Если он равен «1» - будет выполняться чтение. Если он равен «0» - будет выполняться запись. Данный сигнал должен быть установлен в требуемое состояние за некоторое время (для современных микросхем памяти – за единицы или доли наносекунд) тем как будет подан импульс стробирования операции (см. ниже).

**E (operation Enable)** – сигнал стробирования (разрешения выполнения) операции «запись» или «чтение» на шине. По его переднему фронту выполняется чтение данных или запись данных из процессора по шине, в зависимости от того, какая операция выбрана при помощи сигнала R/nW (см. выше). Сигнал E должен удерживаться в активном состоянии в течение всего времени выполнения операции (от единиц до десятков наносекунд).

#### 12.5.3.4 Преобразование сигналов /RD и /WR в сигналы R/nW и E.

Так как большинство микросхем асинхронной памяти и многие периферийные контроллеры имеют сигналы /RD и /WR, а некоторые процессоры оснащены сигналами R/nW и E, то может потребоваться преобразование первых во вторые. Эта делается путем следующих булевых операций (реализованных аппаратно на логических элементах):

$$/WR = (\text{not } (E)) \text{ or } (R/nW); \quad /RD = (\text{not } (E)) \text{ or } (\text{not } (R/nW)).$$

#### 12.5.3.5 Управление длительностью выполнения операций на шине (цикла шины).

К процессору могут быть подключены устройства памяти или периферийные контроллеры с различным быстродействием. Например, «быстрая» асинхронная статическая память может выполнять операцию записи или чтения за 1..10 нс, а ПЗУ или периферийный контроллер – за десятки и сотни наносекунд. Иногда с операцией записи в периферийный контроллер может быть связано не только помещение данных во внутренние ячейки памяти, но и исполнение контроллером сложного алгоритма. Например, запись в контроллер ЖКИ команды очищение экрана подразумевает очистку (запись нулей) во все ячейки видеопамяти, что может занимать десятки микросекунд.

Процессор должен подстраиваться под быстродействие памяти или периферийных контроллеров на шине. Это можно выполнить двумя способами:

1. *Введение циклов ожидания (Wait State)*. В данном случае процессор затягивает длительность сигналов управления операцией на шине (/RD и /WR или R/nW и E), чтобы устройство на шине успело ее выполнить. «Затягивание» выполняется на целое число периодов тактового сигнала процессора, которые называются *циклами ожидания*. Количество циклов ожидания устанавливается путем записи их числа в специальный конфигурационный регистр контроллера памяти. Чтобы память работала при старте системы, до инициализации числа циклов ожидания, по умолчанию устанавливается максимально возможное число.

2. *Использование сигналов готовности RDY (Ready).* Сигнал RDY (Ready) – готовность устройства на шине – формируется устройством на шине для процессора (соответственно для процессора это вход) и указывает, что устройство завершило выполнение операции и готово к приему следующей команды. После установки активного уровня сигнала, например,  $\overline{WR}$  (записи), устройство (память, периферийный контроллер), переводит выход RDY в неактивное состояние – «0» и удерживает в этом состоянии до окончания выполнения операции. Процессор, со своей стороны, удерживает сигнал выполнения операции (в данном примере -  $\overline{WR}$ ) в активном состоянии пока  $RDY = 0$  (неактивен). Как только  $RDY = 1$  (исполнение операции завершилось), процессор «снимает» сигнал  $\overline{WR}$  и тем самым завершает цикл шины.

Используя режим с ожиданием готовности можно совмещать на одной шине устройства с различным быстродействием без простоя процессора: каждое устройство будет само указывать, когда окончилось исполнение команды.

Так как у процессоров обычно один вход RDY, а устройств на шине может быть несколько, то выходы RDY у устройств выполняются с открытым коллектором и подключаются к одному процессорному входу по схеме «монтажное И».

#### 12.5.3.6 Сигналы выбора байтов на шине.

**LB (Lower Byte) и UB (Upper Byte) (другое наименование BLE (Byte Low Enable) и BHE (Byte High Enable))** – два сигнала выбора байта на шине: выбора младшего байта  $\overline{LB}$  и старшего байта  $\overline{UB}$  на 16-ти разрядной шине данных для выполнения текущей операции записи. Это необходимо если по 16-разрядной шине нужно записать только один младший (старший) байт, не изменяя второго – старшего (младшего) байта. Например, нужно записать байт 0x55 по адресу 0x0000. Так как шина данных 16-ти разрядная, то она «покрывает» 2-ве байтовых ячейки памяти по соседним адресам: четному и нечетному. Если на шину адреса выставлен 0x0000, то разряды данных [D7..D0] попадают в байтовую ячейку памяти с адресом 0x0000, а разряды [D15..D7] – в байтовую ячейку памяти с адресом 0x0001. Соответственно на младших разрядах шины данных [D7..D0] процессор устанавливает записываемое значение 0x55, а на старших разрядах [D15..D8] значение не определено и может быть установлено процессором в любое состояние, например, 0x00. Если теперь выполнить операцию записи в микросхему ОЗУ с 16-ти разрядным доступом, то ячейка [0x0000] = 0x55 (это правильно), но ячейка [0x0001] = 0x00 (ЭТО НЕВЕРНО. «ИСПОРЧЕНЫ» ДАННЫЕ ЗАПИСАННЫЕ В ЭТУ ЯЧЕЙКУ РАНЕЕ). Чтобы избежать такой ситуации, используется дополнительный активный сигнал  $\overline{LB}$ , разрешающий запись младшего байта с шины данных. При этом сигнал  $\overline{UB}$  должен оставаться неактивным. Если нужно выполнить запись старшего байта с шины данных (запись байта по нечетному адресу), то наоборот: будет активный  $\overline{UB}$  и неактивный  $\overline{LB}$ . Если нужно записать 16-ти разрядное слово целиком, то активные оба сигнала: и  $\overline{UB}$  и  $\overline{LB}$ .

При операциях чтения сигналы выбора байтов работают аналогично, но в большинстве случаев в этом нет необходимости, так как процессор сам может считать только один из байтов, выставленных на шину данных, или, если нужно, оба байта.

Сигналы  $\overline{UB}$  и  $\overline{LB}$  являются входными для микросхем памяти. Набор сигналов для управления этими входами может несколько различаться у микропроцессоров различных типов, например:

- $\overline{UB}$ ,  $\overline{LB}$  – как описаны выше;
- $\overline{WR0}$ ,  $\overline{WR1}$  – сигналы записи одного байта на 16-ти разрядной шине (см. ниже).

Если микропроцессор имеет 32-х разрядную шину данных, то таких сигналов 4 шт. - каждый управляет доступом к памяти по 8-ми разрядам шины.

**BYTE (Byte)** – входной для микросхемы памяти сигнал, выбирающий используемый разрядность шины данных: 16-ть или 8-мь разрядов. Если на данный вход подан активный

уровень (обычно это НИЗКИЙ уровень, т.е. сигнал «инверсный»), то микросхема переключается в режим использования 8-ми разрядной шины. Если подан неактивный (обычно ВЫСОКИЙ) уровень, то микросхема переключается в режим 16-ти разрядной шины.

Объем памяти и в первом и во втором режиме остается одинаковым, но различается размер адресного пространства: в 8-ми разрядном режиме адресное пространство в 2 раза больше, чем в 16-ти разрядном режиме. Соответственно в 8-ми разрядном режиме используется дополнительный младший разряд адреса. Данный разряд выбирает (адресует) байты в рамках 16-ти разрядного слова, то есть по весу он меньше А0. Поэтому его называют А-1 (вес равен  $2^{-1}$ , т.е. адресуется полслова).

Переключение разрядности шины у микросхемы памяти с помощью сигнала BYTE позволяет использовать один и тот же тип микросхем в ЭВМ с различной (по разрядности) системной шиной. В данном случае сигнал BYTE должен подключаться к «0» или к «1» фиксировано, на этапе проектирования электрической схемы. Следует отметить, что в настоящее время для микросхем ОЗУ такой режим используется достаточно редко: сейчас выпускаются в основном микросхемы ОЗУ с фиксированным размером шины.

Несколько шире применимость сигнала BYTE для микросхем ПЗУ. У ПЗУ (например, у FLASH-памяти) может потребоваться оперативное изменение разрядности шины. Обычно на этапе производства системы в ПЗУ записывается код/данные («прошивка») с помощью внешнего программатора. Для удобного подключения программатора нужно минимизировать число соединительных проводов с программатором, что достигается путем сокращения разрядности шины при переводе микросхемы памяти в байтовый режим. А после этого - в рабочем режиме ЭВМ - ПЗУ может использоваться уже в 16-ти разрядном режиме.

#### 12.5.3.7 Прямой и обратный порядки байт в слове.

**BigEndian и LittleEndian** – это не специальные сигналы управления байтами на шине, а варианты порядка следования байтов в многобайтовом слове данных (это относится не только к организации шин, а вообще к организации многобайтовых слов к памяти).

Существует два варианта. При прямом порядке байт - LittleEndian (вариант Intel) - по младшему адресу находится младший байт. При обратном порядке байт - BigEndian (IBM / Motorola) - по младшему адресу расположен старший байт слова.

Как это влияет на схемотехнику шин?

Так как "выравнивание" слов в разрядной сетке выполняется всегда в сторону младших значащих разрядов, то при LittleEndian-варианте организации байтов на многоразрядной шине в разрядах (D7-D0) всегда будет младший байт слова. Следующий по старшинству байт слова (для 16-ти разрядного слова это будет старший байт) будет располагаться в разрядах (D15-D8) и так далее. То есть, получаем прямой порядок байт, когда меньшие номера разрядов шины данных соответствуют меньшим номерам байтов в слове. Такая организация характерна для многих выпускаемых сегодня микропроцессоров и почти для всех микросхем памяти и микросхем контроллеров ввода-вывода.

Второй вариант – BigEndian. Опять же, так как "выравнивание" слов в разрядной сетке выполняется всегда в сторону младших значащих разрядов, то при BigEndian, в старших разрядах (D31-D24) всегда располагается МЛАДШИЙ байт и по нему идет выравнивание байтов в слове. В разрядах (D23-D16) будет следующий (2-й) байт и т.д. Если по 32-х разрядной шине передается байт – будут использованы разряды (D31-D24); если 16-ти разрядное слово - будут использованы разряды (D31-D16); если 32-х разрядное слово – используются все разряды шины данных.

Вариант BigEndian распространен гораздо меньше, но все-же его поддерживают многие производители, например, фирмы Fujitsu и Freescale - в своих микроконтроллерах, Freescale и IBM - в процессорах PowerPC. При этом появляется необходимость

подключения микросхем памяти Little-Endian-шиной к процессорам с Big-Endian-шиной. В этом случае - при подключении микросхемы памяти с 16-ти разрядной шиной данных - разряды (D7-D0) памяти должны быть подключены к разрядам (D31-D24) процессора, разряды (D15-D8) памяти должны быть подключены к разрядам (D23-D16) процессора, и соответствующим образом должны быть подключены сигналы управления доступом к отдельным байтам (UB, LB). При этом "внутри" байта порядок бит не перевернут: D7 к D31, D6 к D30 и т.д.

#### 12.5.4 Другие сигналы шины управления.

Кроме описанных выше сигналов управления транзакциями на шине, используется еще ряд групп сигналов, кратко описанных ниже:

##### 12.5.4.1 Сигналы передачи информации о состоянии (статуса)

По данным линиям одно устройство может сообщить другому информацию о своем состоянии. Чаще всего ведомое устройство на шине передает ведущему код своего текущего состояния, например, готовность к началу транзакции, или код ошибки по результатам выполнения шинной операции, например, ошибка в принятом по шине слове данных. В определенном смысле к этой группе может быть отнесен сигнал готовности RDY. Сигналы статуса слабо унифицированы для различных шин, для несложных систем используются минимально.

##### 12.5.4.2 Сигналы арбитража.

Арбитраж необходим для выбора одного из нескольких ведущих устройств, одновременно претендующих на доступ к шине. Например, это могут быть несколько процессоров или, более простой вариант, процессор и периферийный контроллер, запрашивающий прямой доступ к памяти. Встречаются различные схемы арбитража: централизованный арбитраж, когда в системе имеется выделенный «центральный арбитр» или «центральный контроллер шины» и децентрализованный (распределенный) арбитраж, когда функции арбитража реализуются путем взаимодействия контроллеров шины в каждом устройстве на шине.

Широко представленная и применяемая в микропроцессорной технике система прямого доступа к памяти (ПДП или DMA) является случаем централизованного арбитража, когда в качестве «центрального арбитра» выступает контроллер ПДП, выделенный или встроенный в микропроцессор. Основными управляющими сигналами такой схемы являются DRQ - «запрос/удержание прямого доступа», передаваемый от устройства к контроллеру ПДП, и DACK – разрешение (подтверждение) прямого доступа, передаваемый от контроллера ПДП к устройству.

##### 12.5.4.3 Сигналы запроса прерывания.

По этим линиям передаются запросы на обслуживание от ведомых устройств (обычно это периферийные контроллеры или устройства) к ведущему (обычно - процессор). Запросы могут быть связаны с транзакциями по шине, например, запрос от сетевого контроллера на передачу принятых данных процессору, или не связаны с шинными транзакциями.

Для передачи запросов используются линии запроса прерываний INT n (n-номер источника запроса), которые являются выходами для ведомого и входами для ведущего устройства (процессора). ВАЖНО! Если линия запроса прерывания в схеме не используется – ее по возможности следует «подтянуть» резистором в неактивное состояние («0» или «1» в зависимости от типа процессора), чтобы не допустить ложного срабатывания из-за помех на входе.

##### 12.5.4.4 Сигналы тактирования.

Это, прежде всего сигнал основной частоты тактирования для синхронных шин. Также могут использоваться вспомогательные тактовые сигналы.

#### 12.5.4.5 Другие сигналы.

К данной группе относятся сигналы конфигурирования устройств на шине, каналы передачи служебных данных и другие линии.

Примером сигналов конфигурирования могут служить сигналы установки адреса (идентификатора) устройства на шине. Адрес устанавливается путем установки соответствующего кода на конфигурационные линии при помощи переключателей на электронной плате ЭВМ.

Если используются заменяемые платы периферийных контроллеров, то при этом могут использоваться два варианта конфигурирования адреса периферийного контроллера: конфигурационные переключатели устанавливаются на «материнской» плате, тогда адрес задается для разъема на материнской плате, а к нему можно подключить любую плату контроллера. Это удобно для быстрой замены платы контроллера, но не удобно если к данному разъему могут подключаться контроллеры различных типов.

Второй вариант – установка конфигурационных переключателей на плате самого контроллера. Это удобно, если конфигурационный код идентифицирует не только адрес, но и тип контроллера, и не удобно при замене контроллера.

Примером каналов передачи служебных данных может служить последовательный канал I<sup>2</sup>S в составе шины доступа к модулям динамической памяти DDR. По этому каналу считываются данные об установленных модулях памяти: их объем, организация, параметры, данные о производителе и т.д.



## 12.5.5

### 13 ПОДКЛЮЧЕНИЕ СИСТЕМНОЙ ПАМЯТИ.

#### 13.1 Общие сведения

Системная память предназначена для хранения кода исполняемых программ и оперативных данных: переменных и параметров.

Микропроцессор и память совместно образуют работоспособное ядро вычислительной системы.

Системная память в целом разделяется на энергозависимую оперативную память, построенную на микросхемах ОЗУ, и на энергонезависимую постоянную память на микросхемах ПЗУ. Кроме прочего данные типы памяти различаются временем выполнения операций чтения/записи (время доступа к памяти; у микросхем ОЗУ оно меньше в разы, чем у микросхем ПЗУ), объемом адресуемых данных (размером адресного пространства) и интерфейсом (составом сигналов, протоколом записи/чтения).

Системная память подключается к системному интерфейсу/шине микропроцессора либо непосредственно, либо через контроллер памяти, который выполняет функции адаптации интерфейса микросхем памяти к системной шине микропроцессора. Во всех случаях основное требование – минимальная скорость доступа при операциях записи или чтения из памяти.

- 4) Непосредственное подключение системной памяти реализуется в простых 8/16/32-разрядных системах<sup>1</sup>, где основным устройством на системной шине является системная память и, соответственно, микропроцессоры имеют системный интерфейс аналогичный или легко адаптируемый к интерфейсу типовых микросхем памяти – и ОЗУ и ПЗУ - и подключается к ним «напрямую», без дополнительных контроллеров памяти. Данный вариант подключения памяти характерен для большинства систем на базе однокристалльных микроЭВМ, так как периферийные контроллеры встроены в однокристалльные микроЭВМ и никаких иных блоков кроме памяти (встроенной памяти «хватает» только для систем нижнего уровня – на 8/16 разрядных микроконтроллерах) к системной шине подключать не нужно.

В некоторых случаях непосредственного подключения системной памяти все же может потребоваться простейшая вспомогательная схема (комбинационный преобразователь сигналов управления, буферные регистры).

- 5) Выделенные контроллеры памяти требуются в системах на базе производительных микропроцессоров<sup>2</sup>, требующих значительных объемов системной памяти (свыше 10Мбайт) и допускающих подключение, в том числе и одновременное, нескольких микросхем и модулей памяти одного или различных типов, в том числе с различными интерфейсами (например, SRAM, FLASH, DRAM, SDRAM).

Контроллеры памяти:

- а) Преобразуют системный интерфейс процессора в интерфейс микросхем памяти используемого типа;
- б) Обеспечивают раздельное подключение различных по быстродействию микросхем ОЗУ и для ПЗУ;
- в) Обеспечивают специальные операции управления памятью вне циклов доступа процессора и без его участия, в том числе, регенерацию динамического ОЗУ, циклы записи FLASH-памяти и т.п.;

<sup>1</sup> В настоящее время к таким системам относятся в основном встраиваемые контроллеры

<sup>2</sup> Такие микропроцессоры широко применяются в персональных компьютерах, серверах, сетевом оборудовании, приборных контроллерах, промышленных компьютерах и в других системах.

- г) Либо встраиваются в микропроцессор, либо реализуются в дополнительных микросхемах системной логики (chipset).

### 13.2 Стандартизация интерфейсов микросхем памяти.

Для микросхем памяти каждого типа (статическое ОЗУ (SRAM), перезаписываемое ПЗУ (EEPROM, FLASH-ROM) и др.) определены международные стандарты на интерфейс, в том числе на состав и функции сигналов, на электрические параметры сигналов, на конструкцию корпуса и расположение выводов. Благодаря этому обеспечивается взаимозаменяемость микросхем различных производителей. Аналогичными стандартами описываются модули и карты памяти, например, SDRAM, DDR I..DDR III, MMC.

Основной организацией, занимающейся стандартизацией в области полупроводниковых приборов, в том числе микросхем, является JEDEC ( Joint Electronic Device Engineering Council - Объединенный инженерный совет по электронным устройствам). В направлении памяти различных типов работают комитеты JEDEC JC-42, JC-45, JC-64 и другие. Есть (особенно в области модулей памяти) и другие организации по стандартизации. Кроме того, есть так называемые «стандарты де-факто»: общепризнанные, но не закрепленные стандартом сертифицированной на это организации.

#### 13.2.1 Электрические аспекты организации шин.

(по учебнику Цилькер, Орлов «Организация ЭВМ и систем»)

## ЛИТЕРАТУРА

1. Азов А.К., Джалиашвили З.О. Интегральные логические схемы в устройствах цифровой вычислительной техники. Учебное пособие. – СПб.: СПбГУИТМО, 2005.
2. Азов А.К. Интегральные логические элементы на биполярных транзисторах. Учебное пособие. – СПб.: СПбГУИТМО, 2005.
3. Титце У., Шенк К. *Полупроводниковая схемотехника* /Пер. с нем. - М.:Мир, 1983.- 512с.
4. Угрюмов Е.П. Цифровая схемотехника. Уч пособие для ВУЗов. 2-е изд. – СПб.: БХВ-Петербург, 2007.-800с.
5. Хоровиц П., Хилл У. Искусство схемотехники /Пер. с англ. 6-е изд. – М.: Мир, 2003.- 704с.
6. Wakerly, John F. Digital Design: principles and practices, third edition. – Prentice Hall, 2000.