

1) Что такое схемотехника? Основные определения (элемент, устройство, цифровое устройство, система). Сигнал, типы сигналов. Классификация элементов.

Схемотехника — это научно-техническое направление, которое занимается анализом и синтезом интегральных схем в интересах выполнения ими оптимальных функций (операций).

Элемент — составная часть, выполняющая определённую функцию и неделимая на составные части.

Устройство — составная часть, выполняющая определённую функцию, представляет собой совокупность составных частей.

Цифровое устройство — устройство для обработки информации в цифровом виде.

Система — совокупность элементов и устройств, выполняющая весь набор требуемых функций.

Сигнал — изменение физической величины. Бывают: 1) аналоговые, 2) цифровые. Вторые по способу кодирования делятся на: потенциальные, динамические, импульсные, и другие.

Элементы классифицируются по функциональному назначению: логические, индикации, памяти, вспомогательные и другие. По ????: 1) комбинационные (выходной сигнал зависит только от входных значений) 2) последовательностные (выходной сигнал зависит от входного и от внутреннего состояния элемента). По типу логики: ДРЛ (диодно-резисторные), ДТЛ (диодно-транзисторные), МДП (металл-диэлектрик-полупроводник), КМОП (комплексный металл-оксид-полупров.).

2) Основные параметры и характеристики логических элементов, их классификация.

Статические параметры (хар-ки) логического элемента: 1) Амплитудно-передаточная, 2) Входная, 3) Выходная.

Параметры: 1) статические параметры — описывают работы л.э. когда сигнал входа не меняется. К ним относятся: выходные и входные напряжения уровня (U^1, U^0), пороговые напряжения. Логические переходы ($U_{л.п.} = U_1 - U_0$). Входные/выходные токи (I^1, I^0). Токи потребления ($I_{потр.}^0, I_{потр.}^1$), Мощность ($P^1, P^0, P_{стат.}$). Помехоустойчивость ($U_{п.мах}^+ = U_{пор.}^0 - U^0, U_{п.мах}^- = U_{пор.}^1 - U^1, U_{п.ср.} = (U_{п.мах}^+ + U_{п.мах}^-) / 2$).

2) динамические — параметры, зависящие от времени. К ним относятся: $t_{з.р.}^{1,0}, t_{з.р.}^{0,1}$ — время задержки распространения сигнала (реакции входного элемента на изменение сигнала). $t_{ф.}^1, t_{ф.}^0$ — длительность фронта (единичного и нулевого).

Классификацию написал выше

3) Логические элементы на биполярных транзисторах. Диодно-резисторная логика (ДРЛ).

4) Логические элементы на биполярных транзисторах. Диодно-транзисторная логика (ДТЛ).

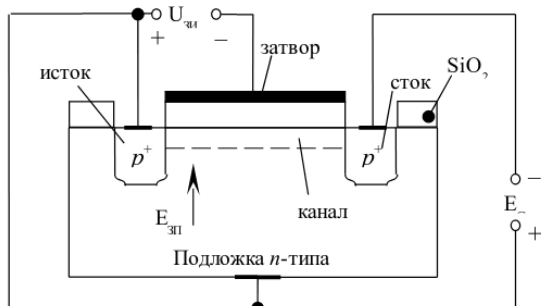
5) Логические элементы на биполярных транзисторах. Транзисторно-транзисторная логика (ТТЛ).

3-5 не проходили.

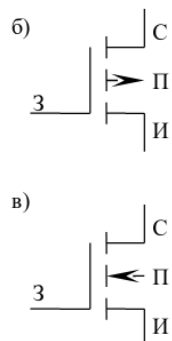
6) Логические элементы на полевых транзисторах. Общие сведения. Классификация полевых транзисторов и основные их параметры.

ПТ считаются лучше для интегральных схем.

Сам полевой транзистор выглядит так:

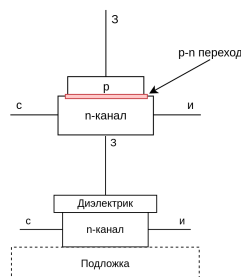


УГО: (б — р-канальный, в — н-канальный):



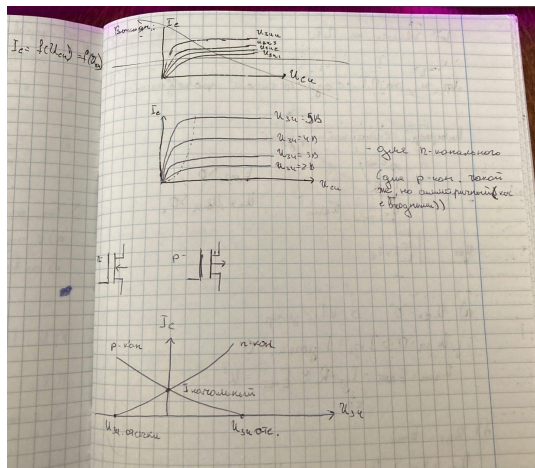
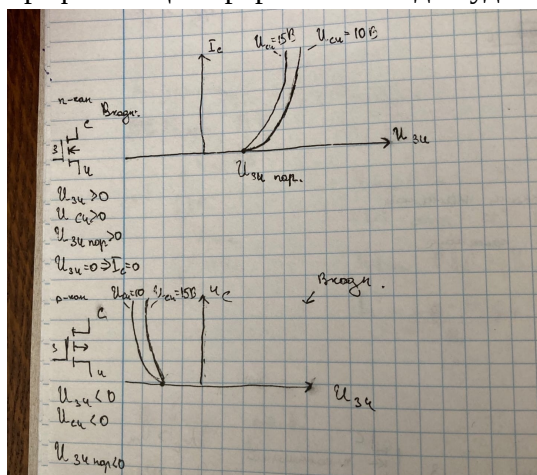
Классификация:

1. Тип проводимости: а) н-канальные
б) р-канальные
2. Тип затвора: а) С управляющим р-п переходом
б) С изолированным затвором
3. Тип канала: а) С индуцированным каналом
б) Со встроенным каналом



Х-ки ПТ: 1) Входная $I_c = f(U_{зи})$ (может быть $I_c = f(U_{зи}) = f(U_{си})$).
2) Выходная $I_c = f(U_{си})$ (может быть $I_c = f(U_{си}) = f(U_{зи})$).

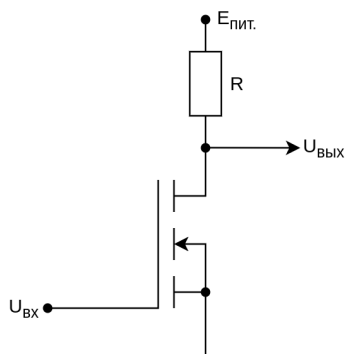
Графики ещё перерисовать надо будет с конспекта:



7) Логические элементы на полевых транзисторах. Инвертор с линейной, нелинейной, квазилинейной и токостабилизирующей нагрузками.

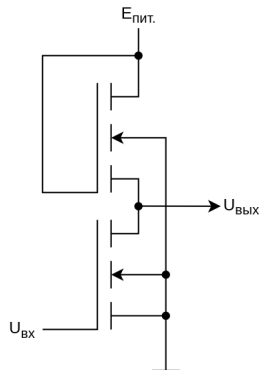
Понятия не имею, что про ЛЭ на ПТ сказать кроме того, что написал выше.

Инвертор с линейной нагрузкой:



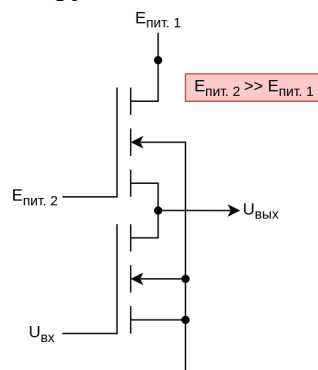
Проблема: резистор (большой, потребляет много мощности и т. д.)

Инвертор с нелинейной нагрузкой:



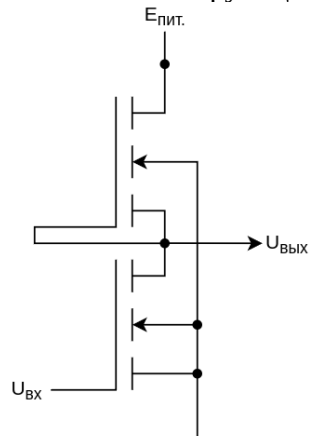
Проблема: график не линейный. Надо сделать $U_{с.и.} < U_{з.и.}$

Инвертор с квазилинейной нагрузкой:



Проблема: лишний источник питания (сложно неприятно)

Токостабилизирующий:



8) Логические элементы на полевых транзисторах. Логические элементы с одним типом проводимости.

Основными логическими элементами являются элементы «И-НЕ» или «ИЛИ-НЕ» для построения полного базиса.

Схема И-НЕ на n-канальном

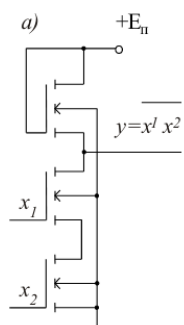


Схема И-НЕ на p-канальном

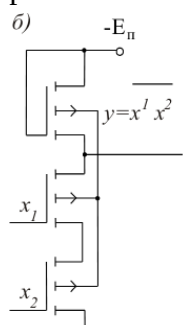


Схема ИЛИ-НЕ на n-канальном

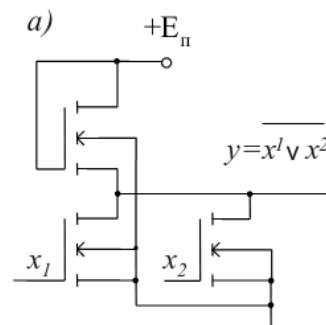
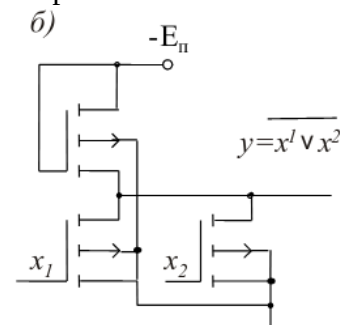


Схема ИЛИ-НЕ на p-канальном.



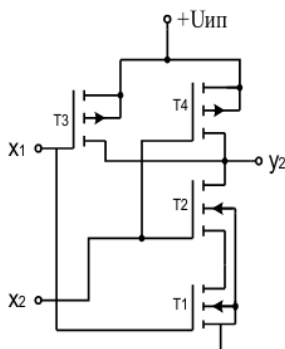
Л.Э. на ПТ имеют следующие преимущества:

1. Высокое входное сопротивление.
2. Малые размеры и высокая технологичность.
3. Меньшая потребляемая мощность, чем у ТТЛ.
4. Высокая помехоустойчивость.
5. Устойчивость к нейтронной составляющей радиационного воздействия.

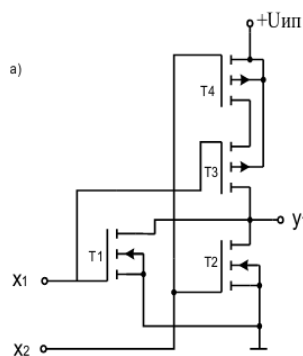
9) Логические элементы на полевых транзисторах. Инвертор на КМДП транзисторах.
Логические элементы на КМДП транзисторах.

Особенность КМДП является свойство не потреблять мощности в статическом режиме при любом сочетании сигналов. Отличием в схемах является последовательное соединение транзисторов с разным типом проводимости. Для полного базиса представлены схемы И-НЕ ИЛИ-НЕ

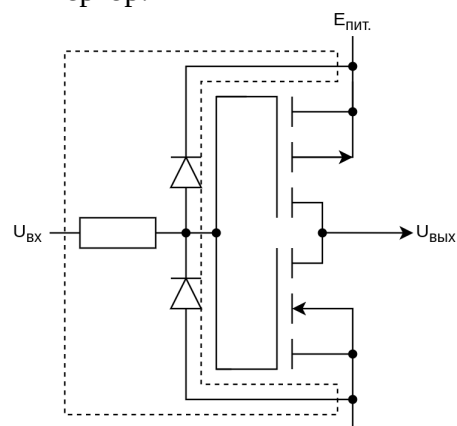
И-НЕ



ИЛИ-НЕ



Инвертор:



На инверторе схема с диодами (обведена пунктиром) нужна исключительно для устранения резкого перепада или подачи напряжения.

10) Функциональные узлы комбинационного типа. Дешифратор. Функциональная и реальная схемы. Время переключения.

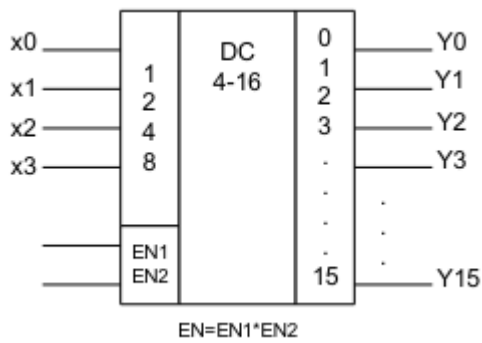
Функциональные узлы предназначаются для выполнения простейших операций над машинными словами или их частями: хранение, изменение и т. д. Строятся из логических элементов и (или) запоминающих элементов.

Функциональные узлы делятся на 1) комбинационные (шифраторы, дешифраторы, мультиплексоры), которые содержат только комбинационную цепь без запоминающих элементов и 2) последовательностные (счётчики, регистры и т. д.), в состав которых входит блок (элемент) памяти.

Дешифратор — устройство, предназначенное для преобразования двоичного кода в унарный комбинационный. В зависимости от двоичного кода на входе, на выходе возбуждается только одна из выходных цепей. Если часть входов не используется, то дешифратор называется неполным.

УГО:

Функционирование описывается формулами:



$$Y_0 = \bar{x}_0 \bar{x}_1 \bar{x}_2 \dots \bar{x}_{n-2} \bar{x}_{n-1} EN,$$

$$Y_1 = \bar{x}_0 \bar{x}_1 \bar{x}_2 \dots \bar{x}_{n-2} x_{n-1} EN,$$

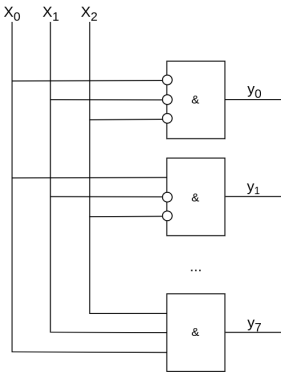
$$Y_2 = \bar{x}_0 \bar{x}_1 \bar{x}_2 \dots x_{n-2} \bar{x}_{n-1} EN,$$

$$\dots$$

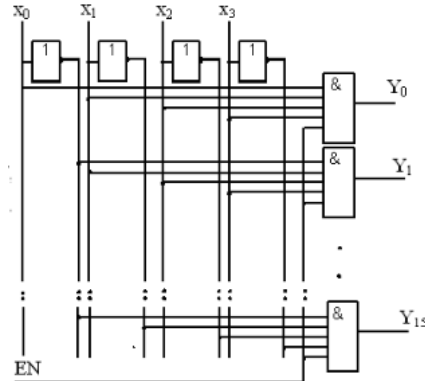
$$Y_{2^{n-2}} = x_0 x_1 x_2 \dots x_{n-2} \bar{x}_{n-1} EN,$$

$$Y_{2^{n-1}-1} = x_0 x_1 x_2 \dots x_{n-2} x_{n-1} EN.$$

Функциональная схема:



Реальная схема:



Время переключения: $t_{DC} = t_{з.инв.} + \max(t_{з.р.}^{1,0}, t_{з.р.}^{0,1})$ (Для реальной схемы).

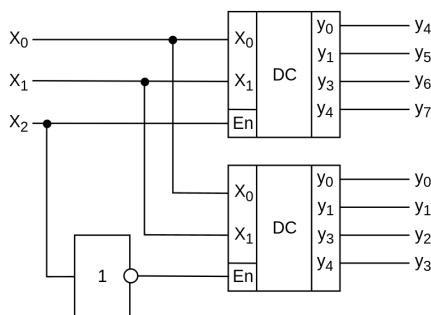
Реальная схема отличается от функциональной для того, чтобы увеличить коэффициент разветвления по выходу.

11) Функциональные узлы комбинационного типа. Дешифратор. Увеличение разрядности дешифратора. Реализация булевых функций на дешифраторе.

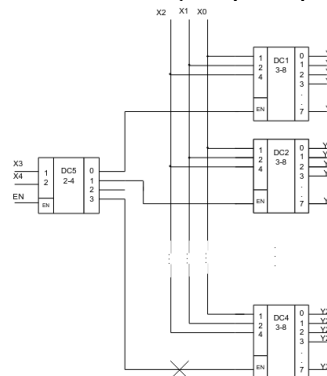
Всё вышеперечисленное, +:

Увеличение разрядности:

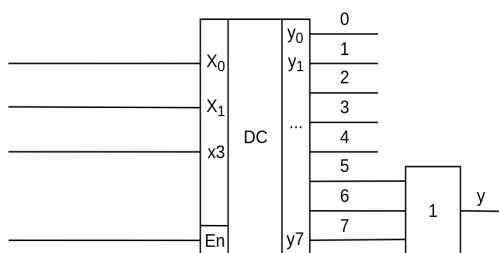
Если из DC 2 → 4 делается DC 3 → 8:



Если DC делается на больше выходов из меньших (например из 2 → 4 в 4 → 16)



На базе дешифратора можно реализовывать различные булевы функции. К примеру: $y = x_1 x_2 x_3 + x_1 x_3 + x_2 x_3$. Данная запись эквивалентна (в логике дешифратора) $y = 7 + 5 + 6$ (111 + 101 + 110. Напоминаю, что x_1 — младший бит, x_n — старший бит). На дешифраторе реализуется как:



12) Функциональные узлы комбинационного типа. Шифратор. Схема шифратора. Управляющие входы/выходы. Расширение разрядности шифратора

Функциональные узлы предназначены для выполнения простейших операций над машинными словами или их частями: хранение, изменение и т. д. Строятся из логических элементов и (или) запоминающих элементов.

Функциональные узлы делятся на 1) комбинационные (шифраторы, дешифраторы, мультиплексоры), которые содержат только комбинационную цепь без запоминающих элементов и 2) последовательностные (счётчики, регистры и т. д.), в состав которых входит блок (элемент) памяти.

Шифратор — это узел, выполняющий функцию, обратную декодированию. На вход подаётся унарный сигнал, а на выходе формируется двоичный код. Активизировать следует только один вход шифратора, но имеются приоритетные шифраторы, в которых при подаче активного уровня сигнала на несколько входов будет шифроваться более приоритетный вход.

УГО:

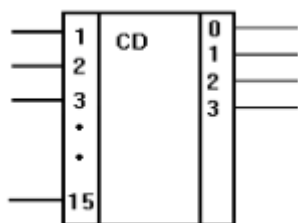


Схема шифратора:

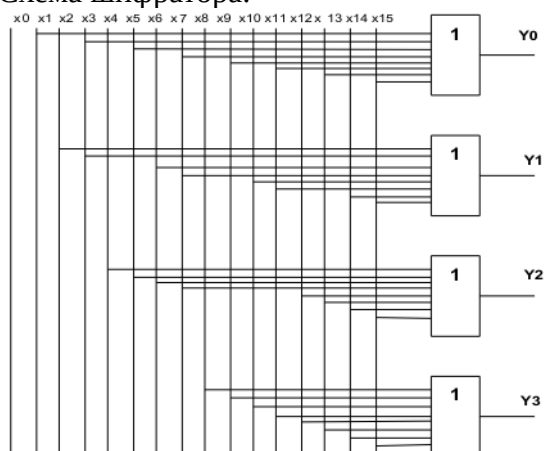


Таблица истинности обычного шифратора:

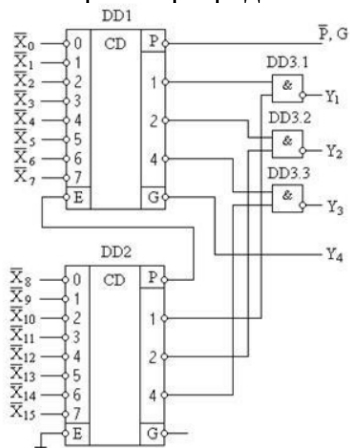
ВХОД				ВЫХОД	
x3	x2	x1	x0	y0	y1
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

Таблица истинности приоритетного шифратора:

ВХОД				ВЫХОД	
x3	x2	x1	x0	y0	y1
0	0	0	1	0	0
0	0	1	*	0	1
0	1	*	*	1	0
1	*	*	*	1	1

У шифратора имеется управляющие входы Ei (Enable input разрешение входа) и выходы Eo (Enable output отсутствие активного уровня на выходах) и G — выход активного уровня сигнала на одном из информационных выходов. Благодаря этим управляющим входам/выходам можно расширять разрядность шифраторов.

Расширение разрядности (Выход P = Eo):



13) Функциональные узлы комбинационного типа. Мультиплексор. Схема мультиплексора. Реализация мультиплексора на дешифраторе и логике.

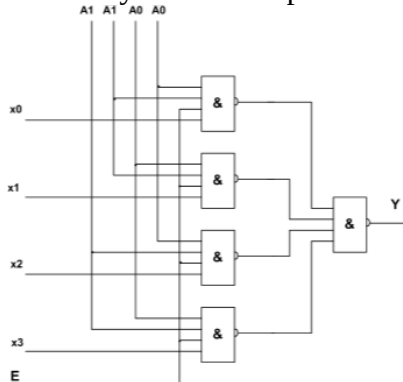
Функциональные узлы предназначены для выполнения простейших операций над машинными словами или их частями: хранение, изменение и т. д. Строятся из логических элементов и (или) запоминающих элементов.

Функциональные узлы делятся на 1) комбинационные (шифраторы, дешифраторы, мультиплексоры), которые содержат только комбинационную цепь без запоминающих элементов и 2) последовательностные (счётчики, регистры и т. д.), в состав которых входит блок (элемент) памяти.

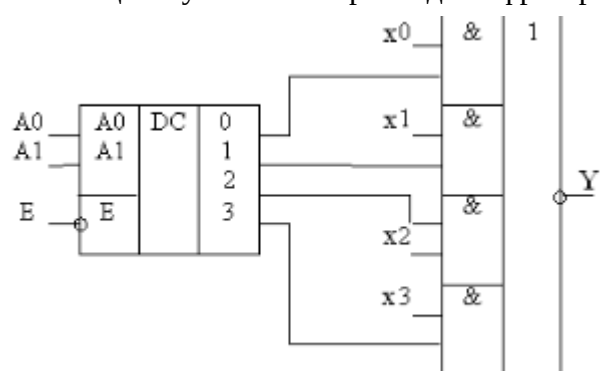
Мультиплексор — узел, реализующий операцию передачи сигнала с любого информационного входа. При помощи адресных входов на мультиплексоре выбирается один из $n=2^k$ информационных сигналов $x_{n-1}..x_0$. На адресные входы подаётся номер индекса в двоичном коде.

Реализуется следующей формулой: $y = x_0 + a_0x_1 + a_1x_2 + a_0a_1x_3$.

Схема мультиплексора:



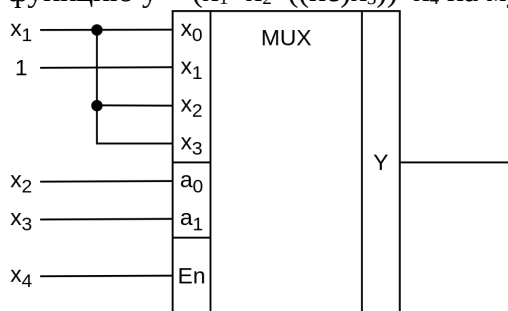
Реализация мультиплексора на дешифраторе:



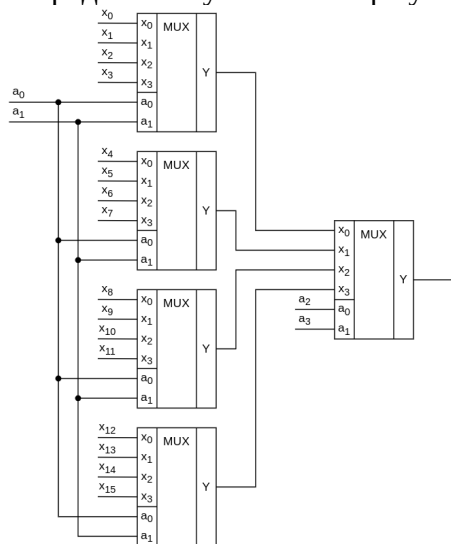
14) Функциональные узлы комбинационного типа. Мультиплексор. Мультиплексор со входом разрешения. Увеличение разрядности мультиплексора. Реализация произвольных булевых функций на базе мультиплексора

Всё вышеперечисленное, +:

Мультиплексор со входом разрешения — просто мультиплексор, в котором ещё есть вход E_n , на котором должен быть активный уровень сигнала, чтобы мультиплексор работал. Благодаря этому можно реализовывать булевы функции на мультиплексорах. К примеру: реализовать функцию $y = (x_1 + x_2 * ((\text{не})x_3)) * x_4$ на мультиплексоре 4 → 1



Разрядность мультиплексора увеличивается следующим образом:



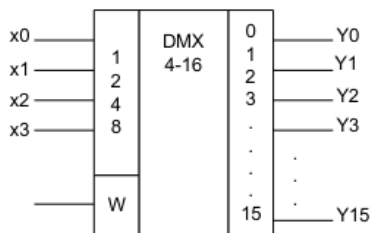
15) Функциональные узлы комбинационного типа. Демультимплексор.

Функциональные узлы предназначены для выполнения простейших операций над машинными словами или их частями: хранение, изменение и т. д. Строятся из логических элементов и (или) запоминающих элементов.

Функциональные узлы делятся на 1) комбинационные (шифраторы, дешифраторы, мультиплексоры), которые содержат только комбинационную цепь без запоминающих элементов и 2) последовательностные (счётчики, регистры и т. д.), в состав которых входит блок (элемент) памяти.

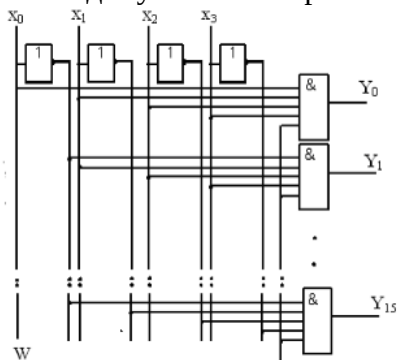
Демультимплексор — узел, выполняющий функцию распределения информации, находящейся на его входе, по своим выходам. Т. е. демультимплексор передаёт значение информации от единственного входа на один из выходов. Активизируемый вход определяется кодом, подаваемым на адресные входы демультимплексора.

УГО:



На самом деле демультиплексор является дешифратором со входом Еп.

Схема демультиплексора = схема дешифратора со входом Еп:



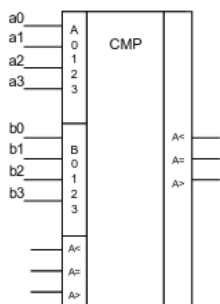
16) Функциональные узлы комбинационного типа. Компараторы. Признаки равенства, больше и меньше битов. Таблицы истинности и булевы выражения. Нарращивание разрядности. Реализация компаратора на равенство с использованием демультиплексора и мультиплексора.

Функциональные узлы предназначены для выполнения простейших операций над машинными словами или их частями: хранение, изменение и т. д. Строятся из логических элементов и (или) запоминающих элементов.

Функциональные узлы делятся на 1) комбинационные (шифраторы, дешифраторы, мультиплексоры), которые содержат только комбинационную цепь без запоминающих элементов и 2) последовательностные (счётчики, регистры и т. д.), в состав которых входит блок (элемент) памяти.

Компаратор — устройство сравнения. Определяет отношения между двумя словами. Основными отношениями, через которые можно выразить остальные являются «равно» и «больше».

УГО:



Признак равенства РАЗРЯДОВ:

$$r_i = a_i b_i \vee \overline{a_i} \overline{b_i} = \overline{a_i b_i \vee \overline{a_i} \overline{b_i}} = \overline{a_i b_i} * \overline{\overline{a_i} \overline{b_i}} = \overline{a_i} \oplus \overline{b_i}$$

Признак неравенства:

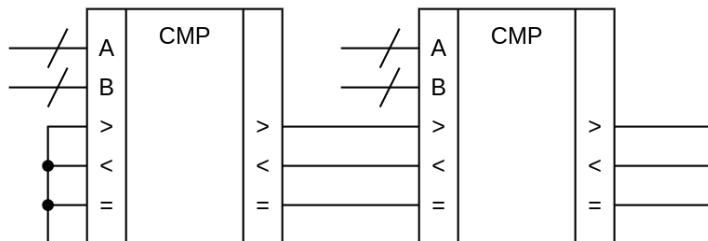
$$\overline{r_i} = \overline{a_i b_i \vee \overline{a_i} \overline{b_i}} = \overline{a_i b_i} \vee \overline{\overline{a_i} \overline{b_i}} = \overline{a_i} \overline{b_i} * \overline{\overline{a_i} \overline{b_i}} = a_i \oplus b_i$$

Признак сравнения чисел: (a_i, b_i)

$F_{a>b} = a_{n-1}(\text{не})b_{n-1} + r_{n-1} \bar{a}_{n-1}(\text{не})b_{n-1} + \dots + r_1 \bar{a}_0(\text{не})b_0$, где r_x – признак равенства предыдущего разряда.

$F_{a<b} = ((\text{не})a_{n-1})b_{n-1} + r_{n-1}((\text{не})a_{n-1})b_{n-1} + \dots + r_1((\text{не})a_0)b_0$

Наращивание разрядности происходит с помощью прямого каскадирования.



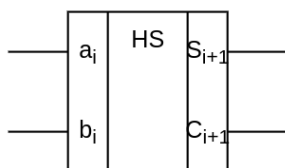
17) Функциональные узлы комбинационного типа. Сумматоры. Определение, классификация. Одноразрядные сумматоры (полусумматор и полный сумматор).

Функциональные узлы предназначены для выполнения простейших операций над машинными словами или их частями: хранение, изменение и т. д. Строятся из логических элементов и (или) запоминающих элементов.

Функциональные узлы делятся на 1) комбинационные (шифраторы, дешифраторы, мультиплексоры), которые содержат только комбинационную цепь без запоминающих элементов и 2) последовательностные (счётчики, регистры и т. д.), в состав которых входит блок (элемент) памяти.

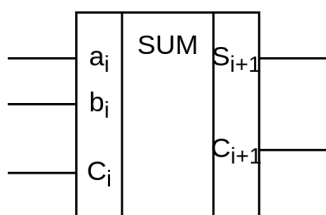
Сумматоры — элементы, выполняющие арифметическое сложение и вычитание чисел. Сумматоры классифицируют как: 1) По С.С.: двоичные, двоично-десятичные и др.. 2) По способу суммирования: комбинационные и накапливающие. 3) По способу организации межразрядных переносов: последовательные, параллельные, смешанные (групповой архитектуры). 4) По способу суммы всех чисел: последовательные и параллельные.

Одноразрядный полусумматор имеет два входа и два выхода. Входы: два числа. Выходы: число и перенос в следующий разряд. Организует работу по формуле: $S_i = a_i \oplus b_i$, $C_{i+1} = a_i b_i$. УГО:

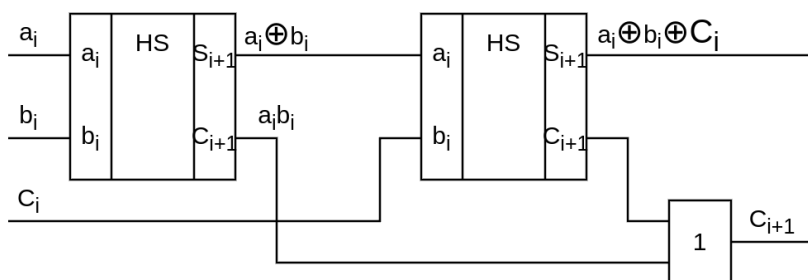


Сумматор – это элемент, который учитывает перенос с прошлого разряда и все так же имеет два выхода.

УГО:



Для получения полного сумматора из полусумматора следует сделать следующее:



18) Функциональные узлы комбинационного типа. Сумматоры. Многоразрядные сумматоры. Последовательный сумматор. Параллельный сумматор с последовательным переносом.

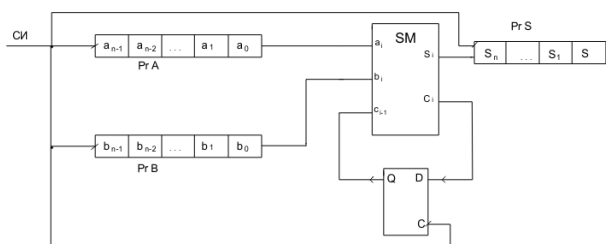
Функциональные узлы предназначены для выполнения простейших операций над машинными словами или их частями: хранение, изменение и т. д. Строятся из логических элементов и (или) запоминающих элементов.

Функциональные узлы делятся на 1) комбинационные (шифраторы, дешифраторы, мультиплексоры), которые содержат только комбинационную цепь без запоминающих элементов и 2) последовательностные (счётчики, регистры и т. д.), в состав которых входит блок (элемент) памяти.

Сумматоры — элементы, выполняющие арифметическое сложение и вычитание чисел. Сумматоры классифицируют как: 1) По С.С.: двоичные, двоично-десятичные и др.. 2) По способу суммирования: комбинационные и накапливающие. 3) По способу организации межразрядных переносов: последовательные, параллельные, смешанные (групповой

Многоразрядные сумматоры отличаются от одnorазрядных тем, что на входы подаются вектора значений. Эти вектора могут складываться последовательно, параллельно и последовательно-параллельно (по типу переноса).

Схема последовательного сумматора:



Плюс такого сумматора — экономит аппаратные ресурсы.

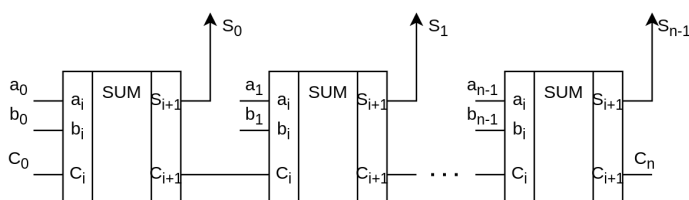
Минусы — время задержки $t_{з.п.} = (t_{\text{триггера}} + t_{\text{SUM}}) * n$, где n — количество разрядов.

Схема работы: На входы регистров a и b подаются вектора чисел в обратном порядке. Т. е. a_0 b_0 — младшие разряды числа. После их сложения вырабатывается перенос, который хранится в триггере и который подаётся на вход переноса предыдущего разряда через такт. Сумма записывается в регистр S .

Параллельный сумматор с последовательным переносом

Строится как цепочка одnorазрядных сумматоров, соединённых последовательно по цепям переноса.

Схема выглядит так:



Плюсы такого сумматора — простая схема.

Минусы: время задержки, и равно: $t_{зд} = t_{SUM} * n$

19) Функциональные узлы комбинационного типа. Сумматоры. Многоразрядные сумматоры. Параллельный сумматор с параллельным переносом.

Функциональные узлы предназначены для выполнения простейших операций над машинными словами или их частями: хранение, изменение и т. д. Строятся из логических элементов и (или) запоминающих элементов.

Функциональные узлы делятся на 1) комбинационные (шифраторы, дешифраторы, мультиплексоры), которые содержат только комбинационную цепь без запоминающих элементов и 2) последовательностные (счётчики, регистры и т. д.), в состав которых входит блок (элемент) памяти.

Сумматоры — элементы, выполняющие арифметическое сложение и вычитание чисел. Сумматоры классифицируют как: 1) По С.С.: двоичные, двоично-десятичные и др.. 2) По способу суммирования: комбинационные и накапливающие. 3) По способу организации межразрядных переносов: последовательные, параллельные, смешанные (групповой

Многоразрядные сумматоры отличаются от одnorазрядных тем, что на входы подаются вектора значений. Эти вектора могут складываться последовательно, параллельно и последовательно-параллельно (по типу переноса).

Параллельный сумматор с параллельным переносом решает проблему времени задержки, но добавляет громоздкость на схему, добавлением блока формирования переносов. Выглядит схема такого сумматора так:

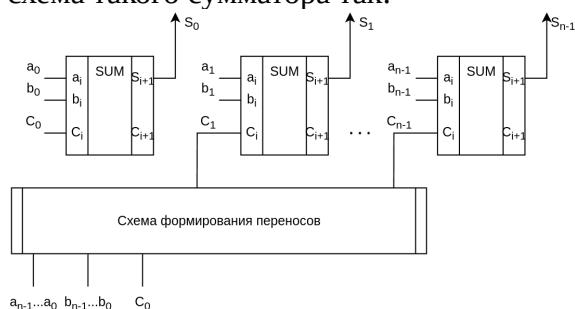
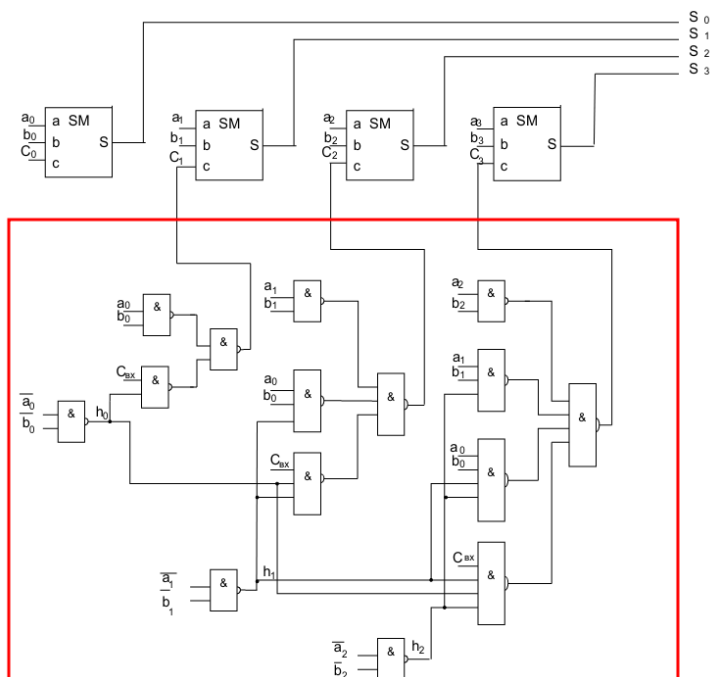


Схема формирования переносов — просто схема, в которой формируется переносы на каждый следующий разряд. Туда сразу же дублируются все значения на входах сумматора. Выглядит она примерно так (запоминать вроде не обязательно):



Такой сумматор работает в двух функциях: прозрачности и генерации. Функция генерации — генерация переноса для следующего разряда НЕ ЗАВИСИМО ОТ ПРЕДЫДУЩЕГО РАЗРЯДА — $g_i = a_i b_i$.

Функция прозрачности — суммирует значения $a_i b_i$ без переноса (сумма по модулю). $h_i = a_i \oplus b_i$

В конечном итоге перенос в следующий разряд формируется как $C_0 = g_0 + C_{BX} h_0$;

$C_1 = g_1 + C_0 h_1 = g_1 + (g_0 + C_{BX} h_0) h_1$;

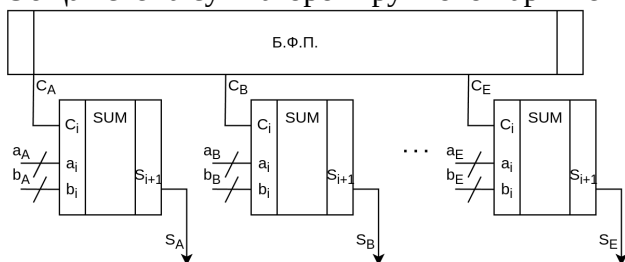
$C_i = g_i + h_i g_{i-1} + h_i h_{i-1} g_{i-2} + \dots + h_i \dots h_1 g_0 + h_i \dots h_0 C_{BX}$.

Время задержки такого сумматора: $t_{зд} = t_{SUM} + t_{Б.Ф.П.}$. (Примерно равно $t_{SUM} * 2$).

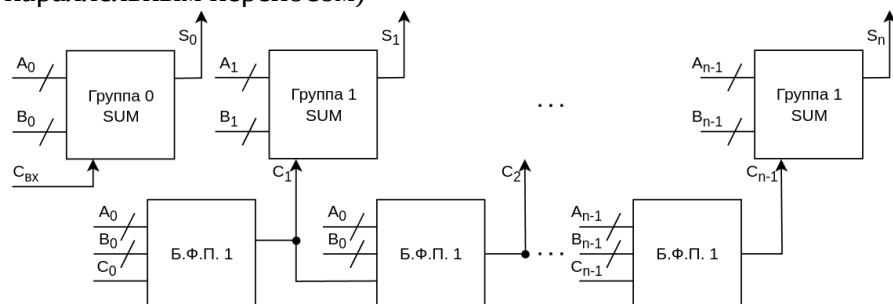
20) Функциональные узлы комбинационного типа. Сумматоры. Многоразрядные сумматоры. Сумматоры групповой архитектуры (с цепным и параллельным переносом).

Всё то же самое, что по сумматорам, +:

Общая схема сумматоров групповой архитектуры выглядит примерно так:

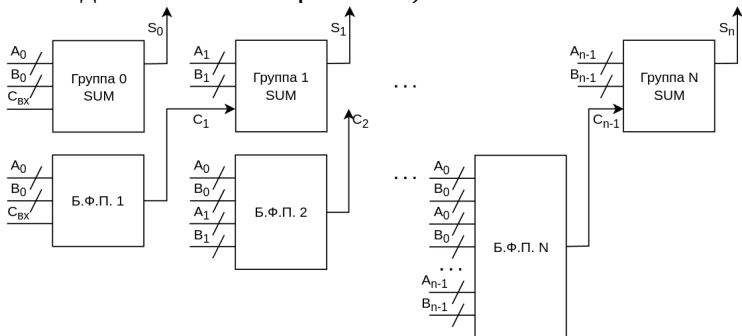


Сумматор групповой архитектуры с цепным переносом (на параллельных сумматорах с параллельным переносом)



Время задержки становится $t_{зд} = t_{Б.Ф.П.} * n$

Сумматор групповой архитектуры с параллельным переносом (на параллельных сумматора с последовательным переносом):

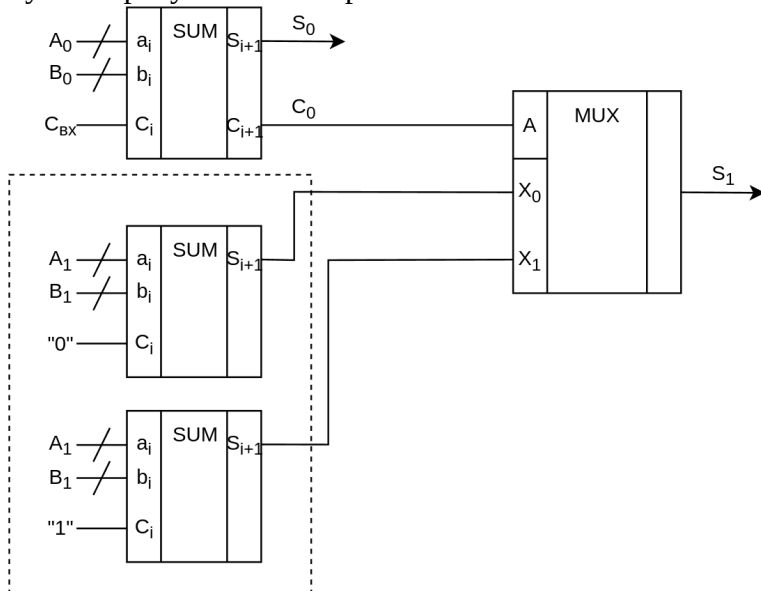


Время задержки уменьшается, и становится $t_{зд} = t_{Б.Ф.П.} + t_{групп.}$

21) Функциональные узлы комбинационного типа. Сумматоры. Многоразрядные сумматоры. Сумматор с условным переносом и накапливающий сумматор.

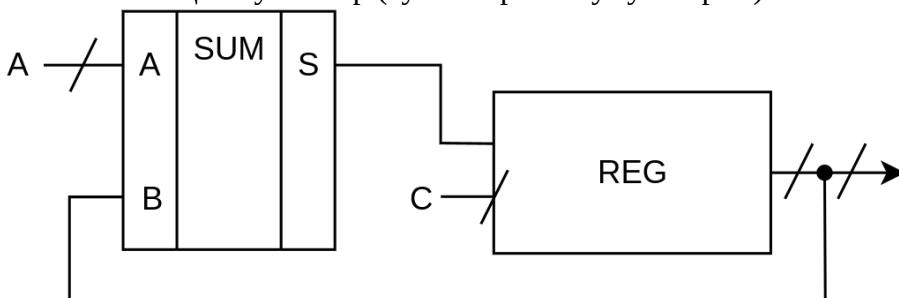
Всё то же самое, что по сумматорам, +:

Сумматор с условным переносом:



В обведённой пунктиром части верхний сумматор предполагает, что переноса нету, а нижний — что он есть.

Накапливающий сумматор (сумматор с аккумулятором)

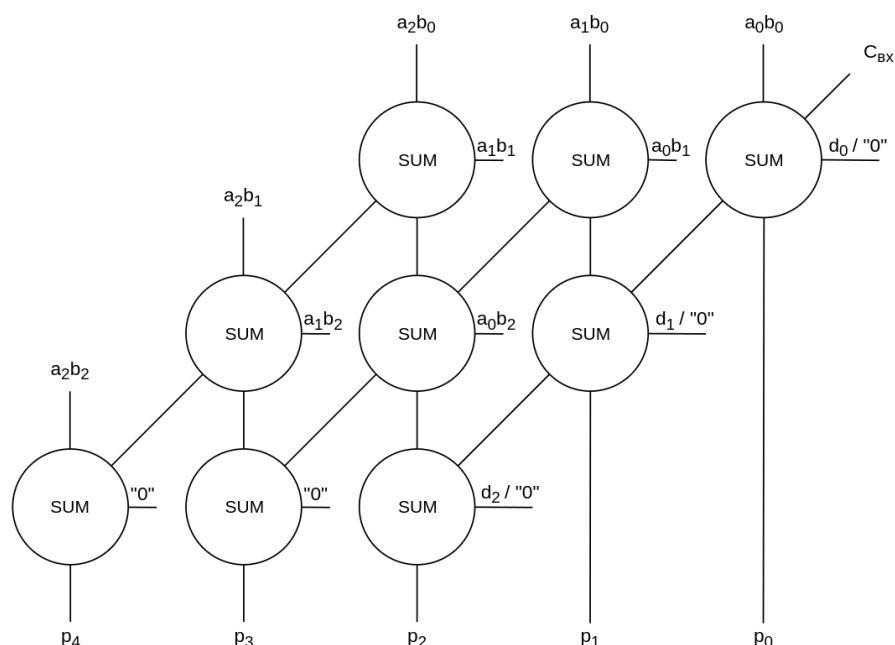


C — синхроимпульс. К содержимому сумматора (которое хранится в регистре) прибавляется очередное слагаемое и результат замещает старое значение суммы. Из регистра выходит целый вектор.

22) Функциональные узлы комбинационного типа. Умножитель.

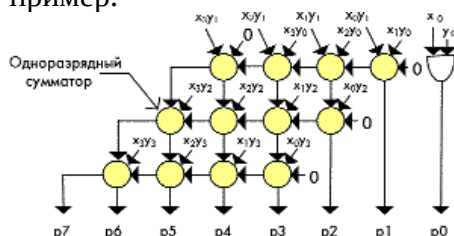
Всё по узлам, +:

Умножитель состоит из сумматоров, которые зачастую связаны в матричном виде. Матричный трёхразрядный умножитель выглядит так:



d — какое-то число, если мы захотим умножить не два, а три числа. Основные разряды — a и b .

Можно собрать умножитель на сколько угодно разрядов благодаря такой системе. Вот пример.



23) Функциональные узлы последовательного типа. Триггеры. Определение, классификация, основные параметры.

Последовательные элементы — это такие элементы, выход которых зависит не только от поступающих сигналов, но и от внутреннего состояния.

Триггеры — это функциональные последовательные узлы, которые могут находиться продолжительное время в устойчивом состоянии.

Классифицируются:

- 1) По функциональному назначению: RS, JK, D, T и другие (смешанные)
- 2) По способу переключения. (асинхронные, синхронные).

Синхронные триггеры:

- 1) Статические (которые переключаются уровнем синхросигнала (или нулём или единицей)).
- 2) Динамические (изменение синхросигнала (когда происходит переключение или $0 \rightarrow 1$ или $1 \rightarrow 0$)).

Стоит ещё рассказать про это переключение по фронту по спаду, как происходит.

Способы описания переключения триггера:

- 1) Таблица переходов
- 2) Графы
- 3) Диаграмма работы (с графиками переключения).
- 4) Характеристическими уравнениями.

К основным параметрам относится время предустановки и время постудержания (на графиках выглядят как время между приходом синхросигнала и переключения триггера. Триггер не может переключаться в тот же момент, как приходит синхросигнал, а происходит это через какое-то время.).

24) Функциональные узлы последовательного типа. Триггеры. RS триггер. Схема, принцип работы. Описание работы табличным способом, характеристическими уравнениями, графом, диаграммами работы.

Всё что выше по триггерам, +:

Схема RS триггера на элементе «И».

(на вход приходят НЕ S и НЕ R).

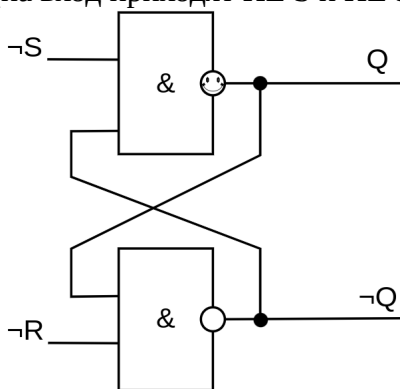
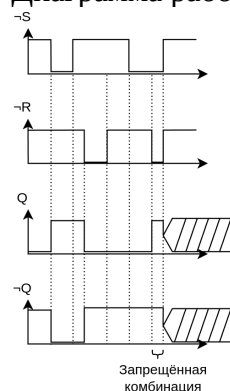


Диаграмма работы:



После прихода запрещённой комбинации триггер будет работать, но как — неизвестно. Может быть быстрее переключится единица, а может быть ноль. Так как на Q и ¬Q стоит единица.