

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ**



ПОБЕДИТЕЛЬ КОНКУРСА ИННОВАЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ ПРОГРАММ ВУЗОВ

Ю.В. Китаев

ОСНОВЫ ЦИФРОВОЙ ТЕХНИКИ



Санкт-Петербург

2007

УДК 681.32

Китаев Ю.В. Основы цифровой техники. Учебное пособие: СПб: СПбГУ ИТМО, 2007, 87 с.

Рассмотрены основные функциональные узлы цифровой техники: логические элементы, комбинационные и последовательностные схемы, а также ЦАП, АЦП и запоминающие устройства.

Для студентов, обучающихся по направлениям “Приборостроение” и “ОпTOTехника”

Рекомендовано к печати Советом ИФФ от 02 октября 2006г., протокол №2.



В 2007 году СПбГУ ИТМО стал победителем конкурса инновационных образовательных программ вузов России на 2007–2008 годы. Реализация инновационной образовательной программы «Инновационная система подготовки специалистов нового поколения в области информационных и оптических технологий» позволит выйти на качественно новый уровень подготовки выпускников и удовлетворить возрастающий спрос на специалистов в информационной, оптической и других высокотехнологичных отраслях экономики.

© Санкт-Петербургский государственный университет информационных технологий, механики и оптики, 2007

© Ю.В. Китаев, 2007

ОГЛАВЛЕНИЕ

1. ЛОГИЧЕСКИЕ ФУНКЦИИ И ЭЛЕМЕНТЫ.....	5
1.1 ОСНОВНЫЕ ПОЛОЖЕНИЯ АЛГЕБРЫ ЛОГИКИ.....	5
1.2 ПЕРЕКЛЮЧАТЕЛЬНЫЕ ФУНКЦИИ.....	6
1.3 СПОСОБЫ ПРЕДСТАВЛЕНИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ.....	7
1.4 СХЕМНЫЕ ОСОБЕННОСТИ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ.....	9
1.4.1 БАЗОВЫЙ ЛОГИЧЕСКИЙ ЭЛЕМЕНТ.....	9
1.4.2 ЭЛЕМЕНТ С ОТКРЫТЫМ КОЛЛЕКТОРОМ.....	9
1.4.3 ТРИСТАБИЛЬНЫЕ ЭЛЕМЕНТЫ.....	10
1.5 УПРАВЛЯЮЩИЕ ВХОДЫ.....	12
1.6 ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ НА КМОП (CMOS) ТРАНЗИСТОРАХ.....	12
1.7 МИНИМИЗАЦИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ.....	13
1.8 ТАБЛИЦА КАРНО.....	14
1.9 ВРЕМЕННЫЕ ПАРАМЕТРЫ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ.....	15
1.10 ПЕРЕХОДНЫЕ ПРОЦЕССЫ В ЛОГИЧЕСКИХ СХЕМАХ.....	17
1.11 КОЭФФИЦИЕНТ РАЗВЕТВЛЕНИЯ ($K_{раз,N}$).....	18
1.12 ЗАДАЧИ И УПРАЖНЕНИЯ.....	19
2. КОМБИНАЦИОННЫЕ СХЕМЫ.....	20
2.1 НЕКОТОРЫЕ СИСТЕМЫ СЧИСЛЕНИЯ.....	21
2.2 ДЕШИФРАТОР.....	21
2.3 ДЕМУЛЬТИПЛЕКСОР.....	24
2.3.1 УВЕЛИЧЕНИЕ РАЗРЯДНОСТИ ДЕШИФРАТОРОВ И ДЕМУЛЬТИПЛЕКСОРОВ.....	24
2.4 МУЛЬТИПЛЕКСОР.....	25
2.5 ШИФРАТОР.....	27
2.6 ПРЕОБРАЗОВАТЕЛИ КОДА.....	29
2.7 СУММАТОРЫ.....	31
2.8 СХЕМА СРАВНЕНИЯ КОДОВ.....	34
2.9 СХЕМА КОНТРОЛЯ ЧЕТНОСТИ (НЕЧЕТНОСТИ).....	35
3. ПОСЛЕДОВАТЕЛЬНОСТНЫЕ СХЕМЫ.....	37
3.1 ТРИГГЕРЫ.....	37
3.1.1 АСИНХРОННЫЙ RS - ТРИГГЕР.....	37
3.1.2 СИНХРОННЫЙ RS - ТРИГГЕР.....	40
3.1.3 D - ТРИГГЕР СО СТАТИЧЕСКИМ УПРАВЛЕНИЕМ.....	41
3.1.4 D - ТРИГГЕР С ДИНАМИЧЕСКИМ УПРАВЛЕНИЕМ.....	44
3.1.5 УНИВЕРСАЛЬНЫЙ JK-ТРИГГЕР.....	45
3.1.6 T - ТРИГГЕР.....	46
3.1.7 ВЗАИМНЫЕ ПРЕОБРАЗОВАНИЯ ТРИГГЕРОВ.....	46
3.2 РЕГИСТРЫ.....	48
3.2.1 ПАРАЛЛЕЛЬНЫЕ И ПОСЛЕДОВАТЕЛЬНЫЕ РЕГИСТРЫ.....	48

3.2.2 РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА	50
3.2.3 СИНХРОННЫЙ И АСИНХРОННЫЙ СПОСОБЫ ЗАГРУЗКИ ПАРАЛЛЕЛЬНОГО КОДА	51
3.3 СЧЕТЧИКИ	52
3.3.1 АСИНХРОННЫЙ СЧЕТЧИК С ПОСЛЕДОВАТЕЛЬНЫМ ПЕРЕНОСОМ.....	53
3.3.2 СИНХРОННЫЙ СЧЕТЧИК С ПАРАЛЛЕЛЬНЫМ ПЕРЕНОСОМ	55
3.3.3 РЕВЕРСИВНЫЙ СЧЕТЧИК	56
3.3.4 КАСКАДНОЕ ВКЛЮЧЕНИЕ СЧЕТЧИКОВ	58
3.3.5 СЧЕТЧИК - ТАЙМЕР	59
3.3.6 ПРИМЕНЕНИЕ СЧЕТЧИКОВ В ИЗМЕРИТЕЛЬНОЙ ТЕХНИКЕ	60
3.3.7 ЗАДАЧИ И УПРАЖНЕНИЯ	61
4. ЦАП И АЦП	63
4.1 ЦАП С МАТРИЦЕЙ РЕЗИСТОРОВ R-2R.....	64
4.2 БИПОЛЯРНЫЙ ЦАП.....	66
4.3 ЧЕТЫРЕХКВАДРАНТНЫЙ ЦАП	67
4.4 АЦП ПОРАЗРЯДНОГО УРАВНОВЕШИВАНИЯ (ПОСЛЕДОВАТЕЛЬНЫХ ПРИБЛИЖЕНИЙ)	68
4.5 АЦП ПАРАЛЛЕЛЬНОГО ТИПА.....	69
5. ЗАПОМИНАЮЩИЕ УСТРОЙСТВА (ПАМЯТЬ).....	71
5.1 СТАТИЧЕСКОЕ ОЗУ (SRAM)	73
5.2 ДИНАМИЧЕСКОЕ ОЗУ (DRAM)	75
5.3 РЕПРОГРАММИРУЕМОЕ ПЗУ	76
5.4 ОДНОКРАТНО ПРОГРАММИРУЕМЫЕ ПЗУ ППЗУ (PROM,ОТР)	78
5.5 ЭНЕРГОНЕЗАВИСИМОЕ СТАТИЧЕСКОЕ ОЗУ (NVS RAM). FRAM и MRAM	79
5.6 УВЕЛИЧЕНИЕ РАЗРЯДНОСТИ ЯЧЕЙКИ ПАМЯТИ (СЛОВА) ..	79
5.7 УВЕЛИЧЕНИЕ КОЛИЧЕСТВА ЯЧЕЕК ПАМЯТИ.....	80
5.8 ПРОГРАММИРУЕМЫЕ ЛОГИЧЕСКИЕ ИНТЕГРАЛЬНЫЕ СХЕМЫ.....	81
5.9 ЗАДАЧИ И УПРАЖНЕНИЯ	82
6. ОПРЕДЕЛЕНИЯ НЕКОТОРЫХ ПАРАМЕТРОВ ИНТЕГРАЛЬНЫХ МИКРОСХЕМ	84
СПИСОК ЛИТЕРАТУРЫ	85

1. ЛОГИЧЕСКИЕ ФУНКЦИИ И ЭЛЕМЕНТЫ

1.1 ОСНОВНЫЕ ПОЛОЖЕНИЯ АЛГЕБРЫ ЛОГИКИ

В отличие от аналоговых электронных устройств, в цифровых устройствах (ЦУ) входные и выходные сигналы могут принимать ограниченное количество состояний. В соответствии с логическим соглашением (ГОСТ 2.743-82), в зависимости от конкретной физической реализации элементов ЦУ, более положительному значению физической величины, "Н" - уровень, соответствует состояние "логическая 1", а менее положительному значению, "L - уровень" - "логический 0". Такое соглашение называется положительной логикой. Обратное соотношение называется отрицательной логикой. В ГОСТ'е 19480 - 89 даны наименования, определения и условные обозначения основных параметров и характеристик цифровых микросхем.

Теоретической основой проектирования ЦУ является алгебра-логики или булева алгебра, оперирующая логическими переменными. Для логических переменных, принимающих только два значения, существуют 4 основных операции. Операция логическое "И" (AND) конъюнкция или логическое умножение, обозначается * или \wedge . Операция логическое "ИЛИ" (OR), дизъюнкция или логическое сложение, обозначается + или \vee . Операция логическое "НЕ" (NOT), изменение значения, инверсия или отрицание, обозначается чертой над логическим выражением. Инверсия иногда будет в тексте обозначаться знаком " ~ " или "#". Операция эквивалентности обозначается "=". Следующие соотношения являются аксиомами.

(1)	$0 + 0 = 0$	$1 * 1 = 1$	(1')
(2)	$1 + 1 = 1$	$0 * 0 = 0$	(2')
(3)	$1 + 0 = 0 + 1 = 1$	$0 * 1 = 1 * 0 = 0$	(3')
(4)	$\sim 1 = 0$	$\sim 0 = 1$	(4')

Из (1, 2) и (1',2') следует: $x + x = x$ и $x * x = x$. (5)

Из (1, 3) и (2',3') следует: $x + 0 = x$ и $0 * x = 0$. (6)

Из (2, 3) и (1',3') следует: $1 + x = 1$ и $x * 1 = x$. (7)

Из (3) и (3') следует: $x + \sim x = 1$ и $\sim x * x = 0$. (8)

Из (4) и (4') следует: $\sim(\sim x) = x$. (9)

И, наконец, из (1,1'), (2,2'), (3,3') и (4,4') следует:

$\sim(x_0 + x_1) = \sim x_0 * \sim x_1$ и $\sim(x_0 * x_1) = \sim x_0 + \sim x_1$. (10)

Последние выражения (10) называют принципом двойственности или теоремой Де Моргана (инверсия логической суммы равна логическому

произведению инверсий и наоборот). Соотношения двойственности для n переменных, часто записывают в виде:

$$\begin{aligned} \sim(x_1 + \dots + x_n) &= \sim x_1 * \dots * \sim x_n \text{ и} \\ \sim(x_1 * \dots * x_n) &= \sim x_1 + \dots + \sim x_n \end{aligned} \quad (11)$$

На функции И и ИЛИ распространяются обычные алгебраические законы - переместительный, сочетательный и распределительный, которые легко доказываются методом перебора: x_1 ор $x_0 = x_0$ ор x_1 - переместительный, x_2 ор x_1 ор $x_0 = (x_2$ ор $x_1)$ ор x_0 - сочетательный и $x_2*(x_1+x_0) = (x_2*x_1) + (x_2*x_0)$ и $x_2 + (x_1*x_0) = (x_2+x_1) * (x_2+x_0)$ - распределительный, где операция ор может быть, либо И, либо ИЛИ. Наряду с тремя основными логическими функциями, называемыми также переключательными, существуют и другие.

1.2 ПЕРЕКЛЮЧАТЕЛЬНЫЕ ФУНКЦИИ

Для n - логических переменных (аргументов) существует 2^n их комбинаций или двоичных наборов. На каждом таком наборе может быть определено значение функции 0 или 1. Если значения функции отличаются хотя бы на одном наборе, функции - разные. Общее число переключательных функций (ПФ) от n аргументов равно $N=2^{2^n}$. Для $n=2$, $N=16$. При $n=3$, $N=256$ и далее очень быстро растет. Практическое значение имеют 16 функций от 2-х переменных, т.к. любое сложное выражение можно рассматривать как композицию из простейших. В таблице 1 приведены некоторые из ПФ для $n=2$. i -номер набора входных переменных x_1 и x_0 .

Таблица 1

i	0	1	2	3	ОБОЗНАЧЕНИЕ ФУНКЦИИ
x_0	0	1	0	1	
x_1	0	0	1	1	
F1	0	0	0	1	$F1 = x_1 * x_0$, "И"
F6	0	1	1	0	$F6 = x_1 \oplus x_0$, "ИСКЛЮЧ.ИЛИ"
F7	0	1	1	1	$F7 = x_1 + x_0$, "ИЛИ"
F8	1	0	0	0	$F8 = \sim(x_1 + x_0)$, "ИЛИ - НЕ"
F9	1	0	0	1	$F9 = \sim(x_1 \oplus x_0)$, "ИСКЛ.ИЛИ - НЕ"
F12	1	1	0	0	$F12 = \sim x_1$, "НЕ"
F14	1	1	1	0	$F14 = \sim(x_1 * x_0)$, "И - НЕ"

ЗАПОМНИТЕ СЛЕДУЮЩИЕ ОПРЕДЕЛЕНИЯ. Функция "И" равна единице, если равны единице ВСЕ ее аргументы. Функция "ИЛИ" равна единице, если равен единице ХОТЯ БЫ один аргумент. Функция "ИСКЛЮЧАЮЩЕЕ ИЛИ" (XOR) равна единице, если равен единице ТОЛЬКО один ее аргумент. Знаком \oplus обозначается логическая функция

"СУММА ПО МОДУЛЮ ДВА", о которой пойдет речь в разделе "Сумматоры". Но для ДВУХ аргументов функции "СУММА ПО МОДУЛЮ ДВА" и "ИСКЛЮЧАЮЩЕЕ ИЛИ" тождественны. Поэтому "ИСКЛЮЧАЮЩЕЕ ИЛИ" в тексте также обозначается \oplus . В дальнейшем \oplus в тексте будет иногда представлено в виде (+).

Функции "И" и "ИЛИ" можно определить по другому: Функция "И" равна нулю, если равен нулю ХОТЯ БЫ один аргумент, функция "ИЛИ" равна нулю, если равны нулю ВСЕ ее аргументы. Отсюда следует, что для нулевых сигналов функция "И" действует, как функция "ИЛИ", а функция "ИЛИ" действует, как функция "И". Это же положение прямо вытекает из выражений (10) и (11) теоремы двойственности. Основным логическим функциям соответствуют одноименные логические элементы (рис.1):

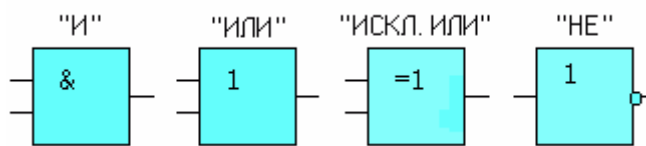


Рис.1. Условные обозначения

1.3 СПОСОБЫ ПРЕДСТАВЛЕНИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ

Целью проектирования цифрового устройства является получение его логической функции (ЛФ) и соответствующей ей схемной реализации. ЛФ могут иметь различные формы представления: 1) словесное, 2) графическое, 3) табличное, 4) алгебраическое, 5) на алгоритмическом языке (например VHDL, AHDL, Verilog, ...) и 6) схемное. В качестве примера, рассмотрим функцию Y от двух переменных x_1 и x_0 , заданную словесным описанием: $Y=1$, если переменные НЕ РАВНЫ и $Y=0$, если $x_1=x_0$. Такую ЛФ удобно назвать функцией НЕРАВНОЗНАЧНОСТИ. Переходим к табличному представлению Y (рис. 2).

i	0	1	2	3
x_0	0	1	0	1
x_1	0	0	1	1
Y	f_0	f_1	f_2	f_3
	0	1	1	0

$$Y = \sum_{i=0}^{2^n - 1} (f_i * m_i) \quad (12)$$

Рис.2. Таблица истинности и формула СДНФ

Табличное представление значений ЛФ для всех наборов входных переменных называется таблицей истинности. В общем виде переход от табличного представления к алгебраическому может осуществляться по формуле (12), одной из основных в алгебре логики. Выражение (12)

называется совершенной дизъюнктивной нормальной формой ЛФ (СДНФ). m_i - минтерм или логическое произведение всех переменных i -го двоичного набора, входящих в прямом виде, если значение этой переменной в наборе равно 1, и в инверсном виде, если ее значение равно 0. f_i - значение ЛФ на i -ом наборе.

Доказательство (12) базируется на теореме разложения, в соответствии с которой любую ЛФ $f(\dots)$ от n -переменных можно разложить по переменной x_i в виде: $f(x(n-1), \dots, x_i, \dots, x_0) = \sim x_i * f(x(n-1), \dots, 0, \dots, x_0) + x_i * f(x(n-1), \dots, 1, \dots, x_0)$. Это выражение для $x_i=0$ равно $\sim 0 * f(x(n-1), \dots, 0, \dots, x_0) + 0 * f(x(n-1), \dots, 1, \dots, x_0) = f(x(n-1), \dots, 0, \dots, x_0)$. При $x_i=1$ оно будет равно $\sim 1 * f(x(n-1), \dots, 1, \dots, x_0) + 1 * f(x(n-1), \dots, 1, \dots, x_0) = f(x(n-1), \dots, 1, \dots, x_0)$, т.е. при любых значениях x_i теорема разложения справедлива. Теорему разложения можно применить n раз и тогда ЛФ будет разложена по всем своим переменным.

В виде примера рассмотрим функцию $F=f(x_1, x_0)$ от двух переменных. Разложение этой функции по переменной x_1 даст: $F = \sim x_1 * f(0, x_0) + x_1 * f(1, x_0)$. Продолжая эту операцию для переменной x_0 , получим:

$$F = \sim x_1 * (\sim x_0 * f(0, 0) + x_0 * f(0, 1)) + x_1 * (\sim x_0 * f(1, 0) + x_0 * f(1, 1)) = \sim x_1 * \sim x_0 * f(0, 0) + \sim x_1 * x_0 * f(0, 1) + x_1 * \sim x_0 * f(1, 0) + x_1 * x_0 * f(1, 1). \quad (12.1)$$

Выражение (12.1) позволяет записать все переключательные функции от двух переменных, используя только три основных логических операции.

Рассмотрим разложение функций $F7$ - "ИЛИ" и $F1$ - "И", для чего необходимо обратиться к соответствующим строчкам таблицы 1. Функция И на двоичных наборах входных переменных x_1 и x_0 (00, 01, 10, 11) принимает значения 0, 0, 0, 1. Записывая выражение (12.1) для этих значений получим: $F1(x_1, x_0) = \sim x_1 * \sim x_0 * 0 + \sim x_1 * x_0 * 0 + x_1 * \sim x_0 * 0 + x_1 * x_0 * 1 = x_1 * x_0$, что соотгласуется с ее определением. Таким же образом, находим алгебраическое выражение функции $F7$ - "ИЛИ", которая, соответственно, на тех же входных наборах принимает значения: 0, 1, 1, 1. Тогда, в соответствии с (12.1), $F7(x_1, x_0) = \sim x_1 * \sim x_0 * 0 + \sim x_1 * x_0 * 1 + x_1 * \sim x_0 * 1 + x_1 * x_0 * 1$. Вынося за скобки в двух последних слагаемых x_1 , получим $F7 = \sim x_1 * x_0 * 1 + x_1 * (\sim x_0 * 1 + x_0 * 1)$. На основании аксиомы (8), выражение в скобке равно "1" и $F7 = \sim x_1 * x_0 * 1 + x_1$. Применяя распределительный закон, найдем $(\sim x_1 + x_1) * (x_0 + x_1) = x_1 + x_0$.

Возвращаясь к таблице 2, получим $Y = 0 * \sim x_1 * \sim x_0 + 1 * \sim x_1 * x_0 + 1 * x_1 * \sim x_0 + 0 * x_1 * x_0 = \sim x_1 * x_0 + x_1 * \sim x_0 = x_1 (+) x_0 = F6$ (т.е. функция неравнозначности для двух переменных совпадает с функцией "ИСКЛЮЧАЮЩЕЕ ИЛИ"). С помощью формулы (12) любую, сколь угодно сложную, логическую функцию можно представить в виде трех основных ЛФ: "И", "ИЛИ", "НЕ", представляющих собой логический базис.

1.4 СХЕМНЫЕ ОСОБЕННОСТИ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ

Приведенные выше логические элементы (ЛЭ) И, ИЛИ, НЕ, И-НЕ и другие могут иметь некоторые схемотехнические особенности.

1.4.1 БАЗОВЫЙ ЛОГИЧЕСКИЙ ЭЛЕМЕНТ

На рисунке 3 приведена упрощенная схема ЛЭ И-НЕ на биполярных n - p - n транзисторах и его условное обозначение.

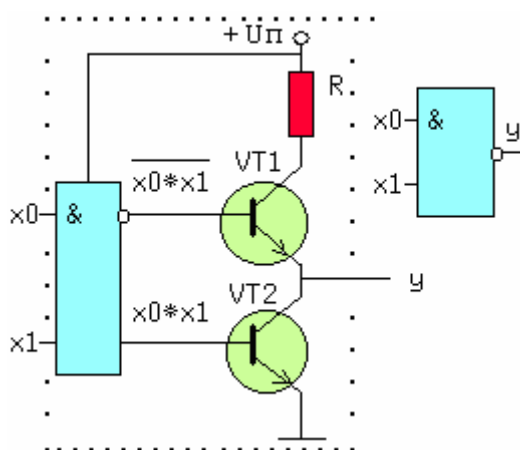


Рис.3. Базовый логический элемент

Напряжения на базах транзисторов $VT1$ и $VT2$ находятся в противофазе и, если $x0*x1=1$, то нижний транзистор открыт, а верхний закрыт, так как $\sim(x0*x1)=0$. Потенциал коллектора $VT2$ в этом случае примерно равен нулю и следовательно $y=0$. При других значениях $x0$ и $x1$ нижний транзистор закрыт, а верхний открыт и на выходе схемы - высокий уровень, т.е. схема работает как элемент И-НЕ. Выходы нескольких БЛЭ категорически нельзя соединять вместе, потому что, если $n-1$ элементов находятся в состоянии "1", а n -ый в состоянии "0", то $n-1$ транзисторов $VT1$ будут "сливать" (sink) токи в единственный транзистор $VT2$ n -го элемента. Суммарный ток может превысить допустимое значение и $VT2$ выйдет из строя.

1.4.2 ЭЛЕМЕНТ С ОТКРЫТЫМ КОЛЛЕКТОРОМ

Логический элемент И-НЕ с открытым коллектором, не имеет внутреннего резистора, подключенного к источнику питания и обозначается в поле элемента ромбом с чертой внизу (рис.4). На выходе такого элемента нетрудно получить логический ноль, для этого на базе транзистора д.б. высокий уровень ($x0=x1=1$). А вот логическую единицу на выходе можно получить, только подключив к коллектору транзистора внешний резистор, соединенный с источником питания (на базе транзистора в этом случае д.б. низкий потенциал). Такой резистор называют подтягивающим (pullup

resistor), т.к. выходное напряжение логической "1" подтягивается к напряжению источника питания $U_{п}$.

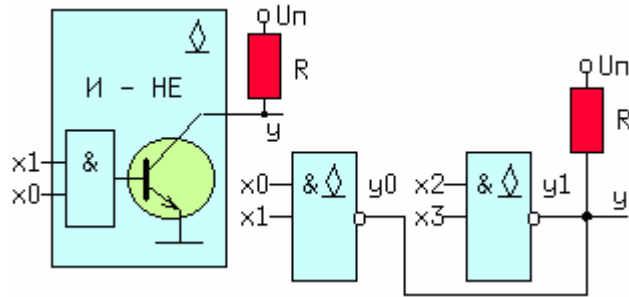


Рис.4. ЛЭ с открытым коллектором

К открытому коллектору снаружи могут подключаться помимо резистора, обмотки реле и двигателей, светодиоды и т.д. Открытые коллекторы нескольких элементов в отличие от базового логического элемента могут соединяться вместе, образуя "монтажное И" для прямых значений переменных т.к. $y = y_0 * y_1 = 1$ при $y_0 = y_1 = 1$. Иногда такую схему называют "монтажное ИЛИ", потому что $y = \sim(x_0 * x_1) * \sim(x_2 * x_3)$ в соответствии с соотношением двойственности равно $\sim(x_0 * x_1 + x_2 * x_3)$. Логический элемент И с открытым эмиттером, обозначается ромбом, но с чертой сверху. ЛЭ с открытым коллектором используются также в микросхемах памяти и шинных формирователях.

1.4.3 ТРИСТАБИЛЬНЫЕ ЭЛЕМЕНТЫ

Наряду с двумя логическими состояниями существует третье технологическое состояние, когда выход логического элемента отключается от внутренней схемы. При этом сопротивление между выходом и "землей" становится очень большим и выход микросхемы не оказывает никакого влияния на подключенные к нему выходы других микросхем.

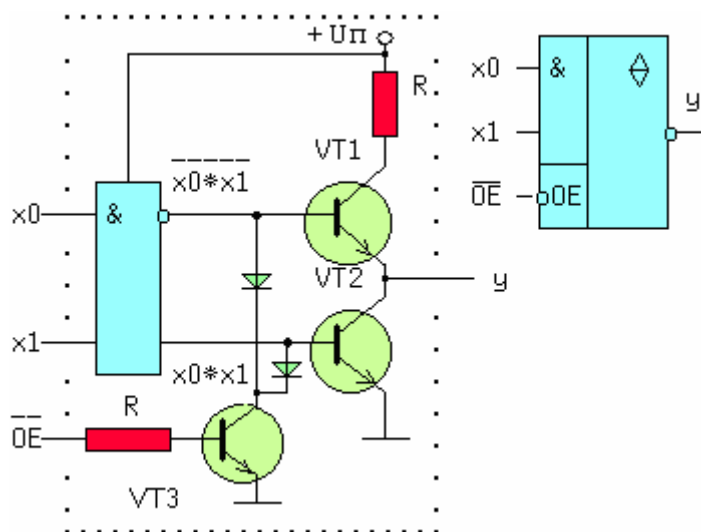


Рис.5. Тристабильный ЛЭ

Выходы нескольких таких элементов также могут соединяться вместе. Такое включение, разновидность "монтажного И", применяется там, где несколько источников сигналов по очереди подключаются к входам одного или нескольких приемников, не мешая друг другу. Третье состояние называют также высокоимпедансным или *Z* - состоянием. Схема И-НЕ с *Z*-состоянием выхода приведена на рисунке 5 слева, а ее условное обозначение - справа.

Если сигнал $\sim OE=0$, транзистор VT3 закрыт и включенные встречно диоды не оказывают влияния на логические выходы элемента И. Напряжения на базах транзисторов VT1 и VT2 находятся в противофазе и, если $x_0 \cdot x_1=1$, то верхний транзистор закрыт, а нижний открыт. Потенциал коллектора VT2 примерно равен нулю и следовательно $y=0$. При других значениях x_0 и x_1 нижний транзистор заперт, а верхний открыт и на выходе схемы - высокий уровень, т.е. при $\sim OE=0$ схема работает как обычный элемент И-НЕ. Картина существенно изменится при $\sim OE=1$. Транзистор VT3 откроется до насыщения и на базах транзисторов VT1 и VT2 потенциал опустится примерно до нуля, запирая их. Выход "у" окажется отключенным от внутренней логической схемы. На схемах тристабильные элементы обозначаются ромбом с поперечной чертой или буквой *Z*.

Как уже говорилось, такие элементы используются там, где необходима передача информации по одной линии от нескольких источников к одному или нескольким приемникам (например к микропроцессору - МП(CPU)). Причем, активным может быть только один источник (рис. 5-1). Так как линия данных одна, то для того, чтобы выходы пассивных источников не искажали информацию на выходе активного источника (или как известно из электротехники, не шунтировали выход "Yk") они должны переводиться в третье состояние, то есть отключаться от линии данных.

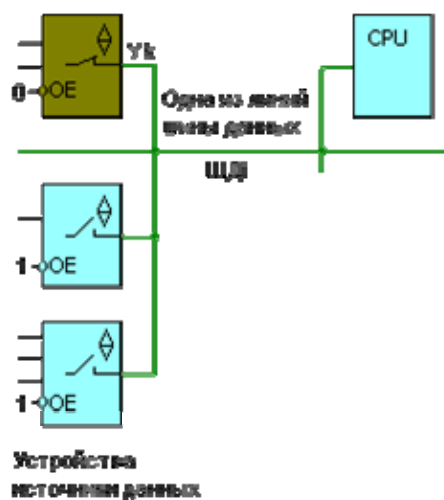


Рис.5-1. Подключение нескольких источников к одному приемнику

На приведенном рисунке верхнее устройство-источник передает данные в МП ($\sim OE=0$), а остальные находятся в третьем (т.е. *Z*) - состоянии ($\sim OE=1$).

Логические элементы с Z - состоянием используются по этой причине в микросхемах памяти, в шинных формирователях и в двунаправленных входах/выходах (портах) микроконтроллеров и ПЛИС. Последнее применение показано на рис. 6 (двунаправленные буферы - инвертирующие и неинвертирующие в отечественном и зарубежном обозначении). При OE=1 порт настроен на выход, а при OE=0 - на вход.

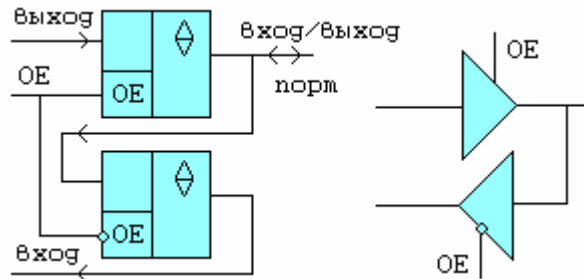


Рис.6. Схема двунаправленного порта

1.5 УПРАВЛЯЮЩИЕ ВХОДЫ

Помимо входов, участвующих в выполнении основной логической функции (входы x_0, x_1, \dots) существуют дополнительные управляющие входы. Например на схемах (рис.5, 5-1 и 6) дополнительный вход (\sim OE, OE) относится к категории управляющих.

1. Действие управляющего входа зашифровано в его обозначении (например Output Enable - разрешение выхода (\sim OE) или OE).
2. Значение уровня на этом входе, при котором основная логическая функция выполняется (такой уровень называется активным) равно 1, если управляющий вход прямой, и равно 0, если управляющий вход инверсный.

1.6 ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ НА КМОП (CMOS) ТРАНЗИСТОРАХ

Ниже, на рисунке 7, приведена схема инвертора на комплементарной паре транзисторов и его передаточная характеристика.

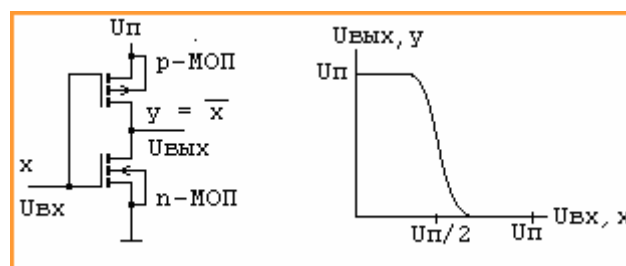


Рис.7. Инвертор на КМОП транзисторах

Оба транзистора работают в ключевом режиме, когда входной и выходной сигналы принимают значения близкие либо к U_p , либо к нулю. При $U_{вх}=0$ КМОП транзистор с n-каналом закрыт, а верхний с p-каналом, наоборот

открыт, поэтому $U_{вых} = U_{п}$. Если $U_{вх} = U_{п}$ то открыт только нижний транзистор и $U_{вых} = 0$.

На следующем рисунке 8 изображен элемент И-НЕ. Если хотя бы одна из переменных x_0, x_1 равна нулю, то хотя бы один p-МОП транзистор открыт и хотя бы один n-МОП транзистор закрыт. Тогда $y = 1$ ($U_{п}$). Если $x_1 = x_0 = 1$, то оба p-МОП транзистора закрыты, а оба n-МОП транзистора открыты и $y = 0$.

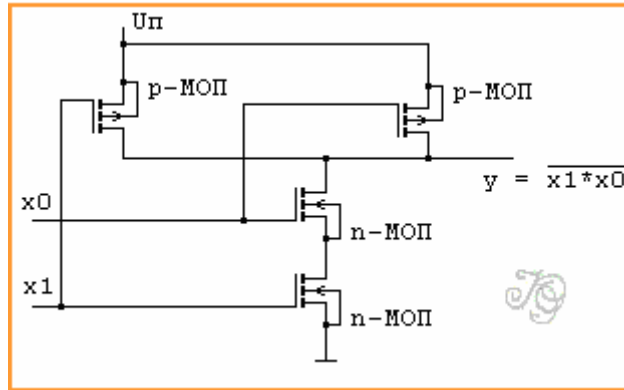


Рис.8. Логический элемент “И-НЕ”

Аналогично выглядит ЛЭ ИЛИ-НЕ (рис.9.). Его работу нетрудно проанализировать самостоятельно.

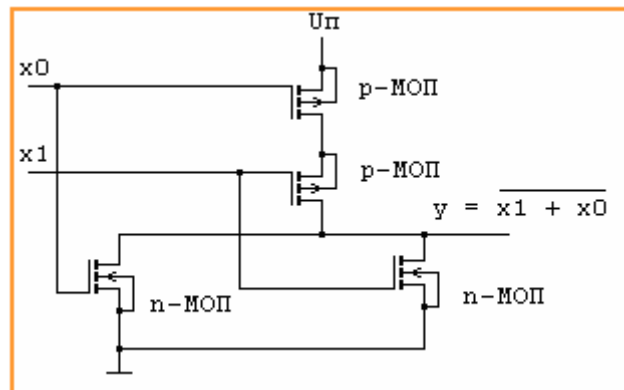


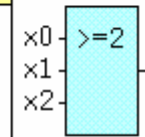
Рис.9. Логический элемент “ИЛИ-НЕ”

1.7 МИНИМИЗАЦИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ

Полученные по формуле СДНФ (12) выражение может быть преобразовано (не всегда) к виду, имеющему меньшее число переменных и операций по сравнению с исходным. Такое преобразование называется минимизацией. Рассмотрим пример. Имеется три двоичных датчика x_i . Необходимо реализовать ЛФ Y мажор принимающую значение 1, когда равны 1 значения двух и более датчиков. Такая функция называется мажоритарной. Ее таблица истинности имеет вид (таблица 2):

Таблица 2

Номер <i>i</i> входного набора	Сигналы от датчиков			ЛФ Yмажор
	x2	x1	x0	
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1



По формуле (12): $Y_{\text{мажор}} = \sim x_2 * x_1 * x_0 + x_2 * \sim x_1 * x_0 + x_2 * x_1 * \sim x_0 + x_2 * x_1 * x_0$.
(3,5,6,7 - строчки таблицы). Полученному выражению соответствует следующая схема (рис.10).

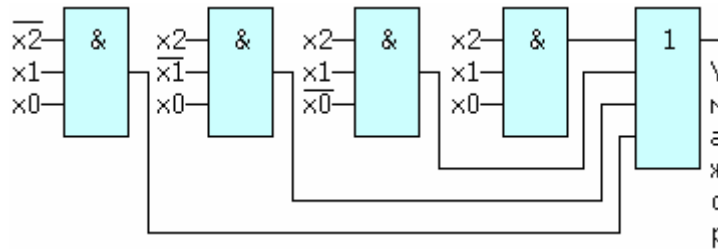


Рис.10. Схема мажоритарности

Схема содержит 4 трехвходовых элемента "И" и 1 четырехвходовый элемент "ИЛИ". Нахождение минимальной формы ЛФ производится методом алгебраических преобразований, с помощью таблиц Карно или машинными методами для больших проектов.

1.8 ТАБЛИЦА КАРНО

Таблица Карно (ТК) это видоизмененная запись таблицы истинности. Для функции мажоритарности из последнего примера (ТК) выглядит следующим образом (рис.11):

		"Yмажор"			
		x1x0	00	01	11
x2	0	0	0	1	0
	1	0	1	1	1

Diagram annotations: A blue box labeled 'A' covers the cells (1,01) and (1,11). An orange box labeled 'B' covers the cells (0,11) and (1,11). A green box labeled 'C' covers the cells (1,11) and (1,10).

Рис.11. Таблица Карно для ф-ии мажоритарности

Правила построения ТК следующие: 1) Количество клеток ТК равно количеству строк таблицы истинности. 2) Слева и сверху располагаются значения аргументов. Порядок размещения аргументов таков, что в двух соседних по горизонтали и вертикали клетках отличается значение только одного аргумента (поэтому соседними считаются и клетки, находящиеся на противоположных краях таблицы). 3) В клетки заносятся соответствующие значения ЛФ. 4) Единичные клетки объединяются в прямоугольники (импликанты) по 2^i клеток. 5) Для каждого прямоугольника записывается произведение тех аргументов, которые в соседних клетках не изменяют своего значения. 6) Переменные входят в произведение в прямом виде, если их значение в соседних клетках равно 1, в противном случае в инверсном. 7) Полученные произведения складываются по ИЛИ в искомую ЛФ.

В примере имеется 3 прямоугольника - A,B,C, причем $Y_a = x_2 * x_0$ (x_1 в соседних клетках меняет свое значение, поэтому в конъюнкции не входит). $Y_b = x_1 * x_0$ и $Y_c = x_2 * x_1$.

$$Y_{\text{мажор}} = Y_a + Y_b + Y_c = x_2 * x_0 + x_1 * x_0 + x_2 * x_1.$$

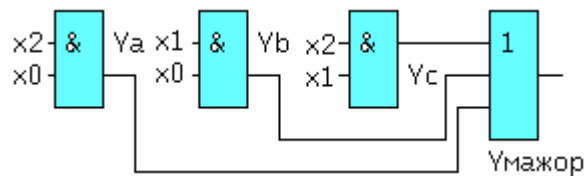


Рис.12. Минимизированная схема мажоритарности

Соответствующая схема (рис.12) проще, чем предыдущая.

1.9 ВРЕМЕННЫЕ ПАРАМЕТРЫ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ

Рассмотрим реакцию инвертора на изменение входного сигнала.

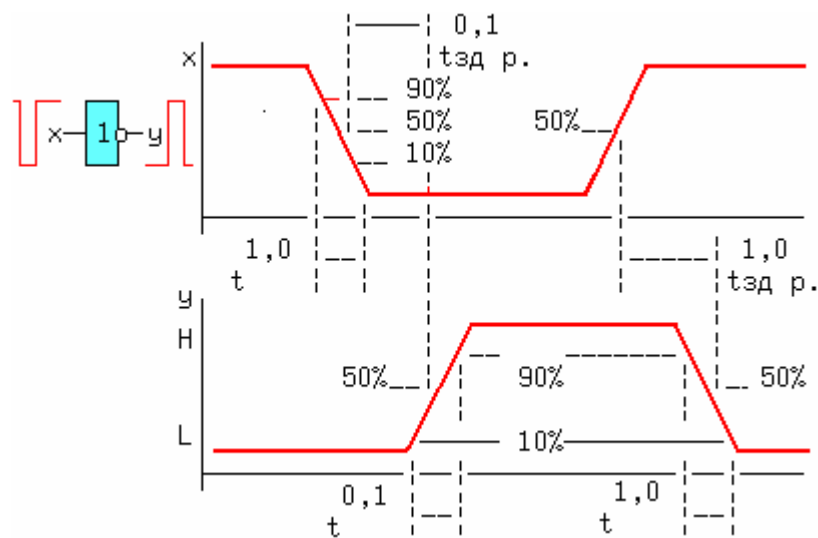


Рис.13. Временные параметры

Инерционные свойства инвертора приводят к задержке сигнала при его прохождении от входа к выходу (рис.13).

Процесс изменения напряжения от низкого уровня L к высокому H, называется фронтом сигнала (положительным перепадом, положительным фронтом), а обратный процесс - спадом (отрицательным перепадом, отрицательным фронтом). Длительность фронтов на рис.13 обозначена $t_{1,0}$ - отрицательный и $t_{0,1}$ - положительный. Величинами $t_{зд.р.0,1}$ и $t_{зд.р.1,0}$ обозначается время задержки распространения сигнала от входа до выхода при переходе из 0 в 1 и наоборот. Минимальная длительность импульса на входе элемента $t_{и.мин}$ пропорциональна среднему значению $t_{зд.р.ср.}$ равному полусумме $t_{зд.р.0,1}$ и $t_{зд.р.1,0}$. Максимальная частота входных импульсов $F_{макс}$ обратно пропорциональна $t_{зд.р.ср.}$. Из сказанного следует, что быстродействие элемента тем выше, чем меньше $t_{зд.р.ср.}$. В зарубежной литературе $t_{зд.р.}$ обозначается tpd (propagation delay).

В цифровой и микропроцессорной технике применяются еще два параметра:

$t_{уст}$ / t_{SU} - время установления входного сигнала (clock setup time) . Интервал времени между началом сигнала на одном заданном входе и активном переходом на другом заданном входе (обычно тактовом - clock).
 $t_{уд}$ / t_H - время удержания (hold). Время, в течение которого сигнал удерживается на заданном входе после активного перехода на другом заданном входе (рис.14).

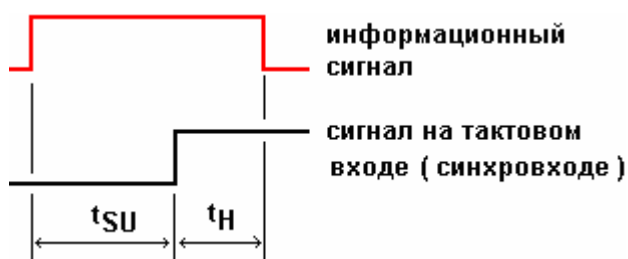


Рис.14. Время установления и время удержания

В последовательностных схемах используется еще один параметр: t_{co} (clock to output delay) - задержка появления выходного сигнала от момента появления активного фронта на тактовом входе. Определения вышеуказанных величин с их отечественными и международными обозначениями приведены в разделе обозначения некоторых параметров микросхем.

Быстродействие схемы зависит не только от перечисленных параметров, но и от алгебраической формы представления ЛФ (рис.15). Пусть $y = a*b + c*a + d = a*(b+c)+d$. Первой форме (ДНФ) соответствует схема (А), а второй - схема (В).

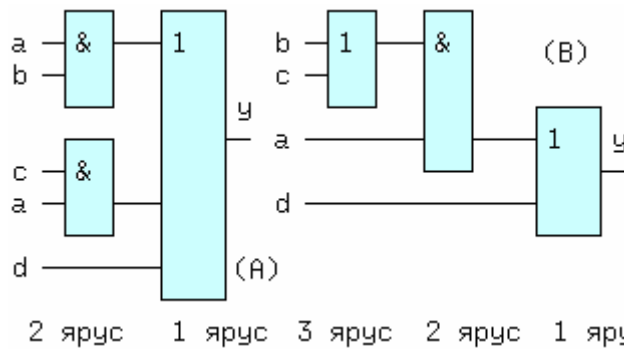


Рис.15. Два способа оптимизации логических схем

Если среднее время задержки сигнала в каждом элементе одинаково, то $2t_{зд.р.ср.} < 3t_{зд.р.ср.}$ и двухъярусные схемы (СДНФ) в общем случае быстрее. Правда в записи со скобками может уменьшиться количество элементов и/или проводников (в схеме (В) на один проводник меньше). Выбор схемы оптимальной по быстродействию или по минимальному числу связей/элементов остается за разработчиком. Большинство систем автоматизированного проектирования (САПР) имеют возможность оптимизации разрабатываемых цифровых устройств по указанным критериям, например "MAX+plus II", "Quartus II", "WebPack ICE" и другие.

1.10 ПЕРЕХОДНЫЕ ПРОЦЕССЫ В ЛОГИЧЕСКИХ СХЕМАХ

Отличие времени задержки $t_{зд.р.}$ от нуля при прохождении сигнала через логическую схему может приводить к возникновению помех в выходном сигнале. Эти помехи имеют вид коротких импульсов, и в некоторых случаях приводят к серьезным сбоям в работе схем. Рассмотрим устройство на рис.15-1. Если элементы схемы не вносят задержки сигнала, а x_0 и x_1 находятся в противофазе, т.е. $x_0 = \sim x_1$, то $y = \sim(x_1 * \sim x_1) = 1$. Если же каждый из четырех инверторов имеет задержку $t_{зд.р.}$, то x_0' запаздывает относительно x_0 на $4t_{зд.р.}$ и на выходе схемы возникает короткий незапланированный "отрицательный" импульс (интервал 1..2), сдвинутый на $t_{зд.р.}$ элемента И-НЕ (интервал 0..1). Процесс прохождения входных сигналов до общего выхода называется состязаниями или "гонками".

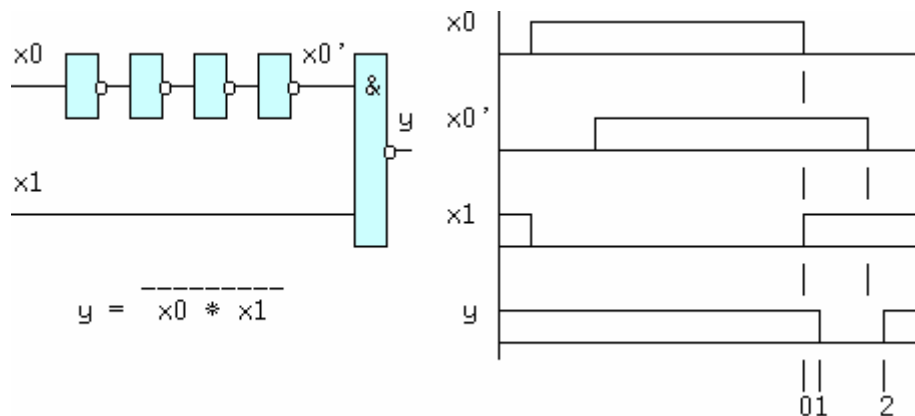


Рис.15-1. Помеха, вызванная гонками

Вредный эффект "гонок" может быть устранен несколькими способами, один из которых заключается в добавлении к ЛФ дополнительного слагаемого. Пусть некоторая ЛФ равна $F = x_1 * x_2 + \sim x_1 * x_0$, тогда при $x_2 = x_0 = 1$ может появиться помеха, вызванная тем, что сигнал $\sim x_1$ задержан относительно x_1 на величину задержки инвертора (см. рис.15-2).

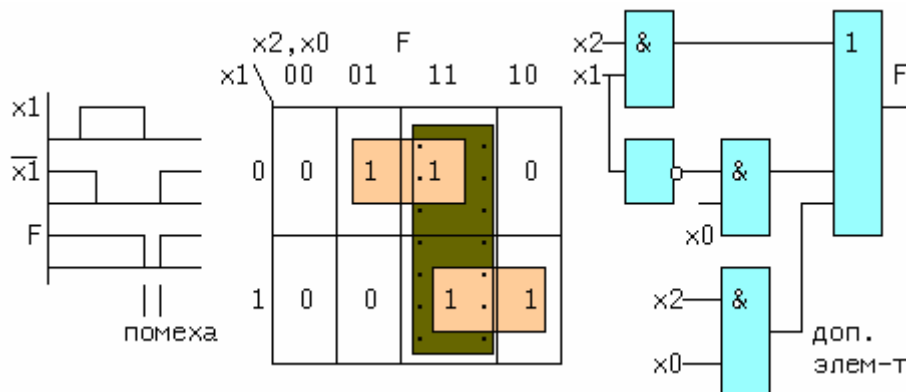


Рис. 15-2. Способ устранения помех

Добавление дополнительного импликанта (в таблице обведен точками) устраняет проблему, т.к. при критической ситуации, когда $x_2 = x_0 = 1$, дополнительная составляющая $x_0 * x_2 = 1$ и функция $F = x_1 * x_2 + \sim x_1 * x_0 + x_0 * x_2$ равна всегда 1 при $x_2 = x_0 = 1$.

В устройствах индикации такие короткие помехи можно игнорировать, так как они будут незаметны для глаз.

Эффект, вызванный состязаниями может иметь и положительное значение, например в удвоителе частоты (еще одно полезное применение "ИСКЛЮЧАЮЩЕГО ИЛИ"). Для наглядности задержка выходного сигнала на рисунке 15-3 не показана.

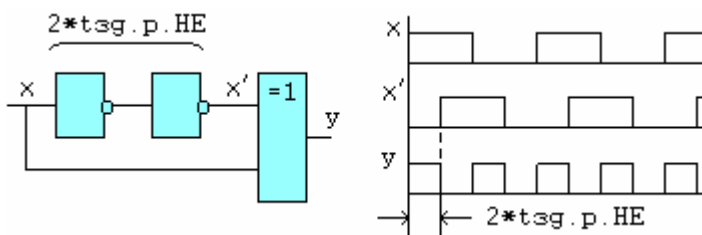


Рис. 15-3. Удвоитель частоты

1.11 КОЭФФИЦИЕНТ РАЗВЕТВЛЕНИЯ (Краз,N)

Коэффициент разветвления или нагрузочная способность - максимальное число входов которые можно подключить к отдельному выходу микросхемы (м/с). Причем сумма входных токов должна быть меньше выходного тока отдельного выхода м/с (рис. 15-4). Если м/с имеет несколько выходов, то суммарный ток всех выходов не должен превышать паспортного значения для данной микросхемы, даже если отдельные выходы микросхемы будут недогружены.

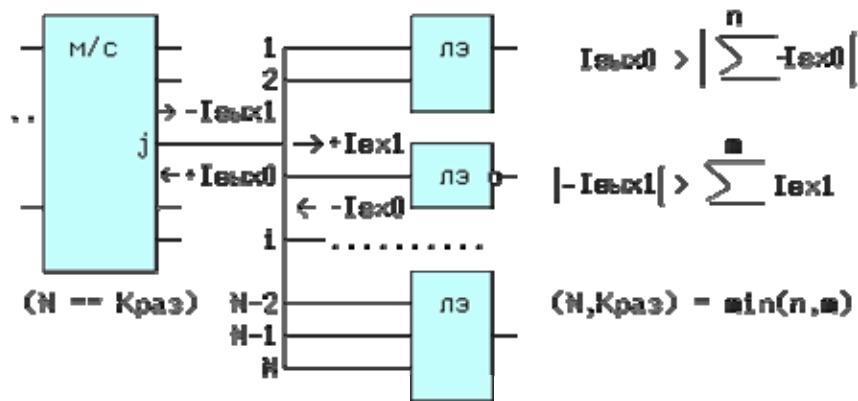


Рис.15-4. Коэффициент разветвления

Для определения N находят отдельно две суммы входных токов для логического 0 и 1 на j -выходе. Минимальное из двух значений и будет $\text{Краз} = \min\{m, n\}$. m, n - целые значения.

1.12 ЗАДАЧИ И УПРАЖНЕНИЯ

1) На входы схемы одновременно поступают несколько сигналов. Через какой промежуток времени на выходе гарантированно появится истинное значение логической функции? Задержки сигналов при прохождении через логические элементы приведены в таблице ($1\text{нсек}=10^{-9}$ сек) на рисунке 15-5.

тзд.р.ИЛИ	тзд.р.И	тзд.р.ИСКЛ.ИЛИ	тзд.р.НЕ
9	10	11	7

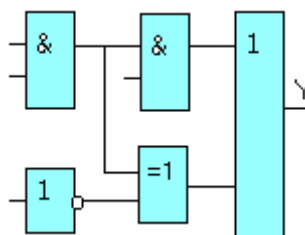


Рис.15-5. К задаче 1

Ответ: 30

2) Выберите правильный порядок следования логических элементов. Например: И, ИЛИ, ИСКЛЮЧАЮЩЕЕ ИЛИ (рис.15-6).

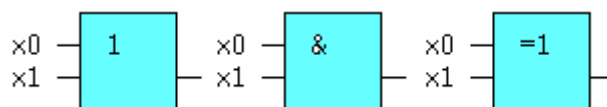


Рис.15-6. К задаче 2

- И, ИЛИ, ИСКЛЮЧАЮЩЕЕ ИЛИ.
- ИЛИ, И, ИСКЛЮЧАЮЩЕЕ ИЛИ.
- ИЛИ, ИСКЛЮЧАЮЩЕЕ ИЛИ, И.

3) Укажите правильное определение.

1. Функция И равна нулю, если равен нулю только один аргумент.
2. Функция ИЛИ равна нулю, если равен нулю только один аргумент.
3. Функция И равна единице, если равен единице хотя бы один аргумент.
4. Функция И равна нулю, если равен нулю хотя бы один аргумент.
5. Функция ИЛИ равна единице, если равны единице все аргументы.

- 1
- 2
- 3
- 4
- 5

4) Какому уравнению соответствует таблица Карно (рис.15-7)?

		x1, x0			
		00	01	11	10
x3, x2	00	1	1	0	1
	01	0	0	0	0
	11	1	1	0	0
	10	1	1	0	1

Рис.15-7. К задаче 4

- $x_3 * x_2 + \sim x_1 * x_0 + x_3 * \sim x_2$
- $x_1 * x_0 + \sim x_2 * x_1 + \sim x_0 * \sim x_3$
- $\sim x_0 * \sim x_2 + \sim x_1 * \sim x_2 + x_3 * \sim x_1$
- $\sim x_1 * \sim x_0 + x_2 * \sim x_1 + \sim x_3 * \sim x_0$
- $x_0 * x_1 + x_2 * x_3 + x_3 * \sim x_2$

2. КОМБИНАЦИОННЫЕ СХЕМЫ

В комбинационных схемах логическая функция зависит только от комбинации значений входных переменных. При описании многих цифровых устройств невозможно обойтись без упорядоченных двоичных наборов входных и выходных сигналов. Эти наборы удобно представлять в тех или иных системах счисления (СС).

2.1 НЕКОТОРЫЕ СИСТЕМЫ СЧИСЛЕНИЯ

В позиционных СС "вес" каждого разряда зависит от его позиции в числе. К числу непозиционных относится "римская" СС, например число - XVII. Любое целое неотрицательное n-разрядное целое число в позиционной системе счисления может быть представлено в виде:

$$D = C_{n-1} * b^{n-1} + C_{n-2} * b^{n-2} + \dots + C_1 * b^1 + C_0 * b^0$$

где D - десятичный эквивалент числа, C_i - значение i-го разряда, b - основание системы счисления, b в степени i - вес (весовой коэффициент) i-го разряда и n число разрядов числа. В цифровой и вычислительной технике наиболее распространены двоичная (BIN), десятичная (DEC), шестнадцатиричная (HEX) и непозиционная двоично-десятичная (BCD) системы счисления. В BCD системе вес каждого i-го десятичного разряда равен 10 в степени i, как в десятичной системе, а каждая цифра i-го разряда кодируется 4-мя двоичными цифрами. Восьмиричная СС(ОСТ) применяется реже. В 16-ной системе счисления цифры от 0 до 9 совпадают с десятичными, а для ЦИФР больше 10 используются буквы латинского алфавита : A(a) = цифра 10, B(b) = 11, C(c) = 12, D(d) = 13, E(e) = 14, F(f) = 15. Двоичное число преобразуется в десятичное беззнаковое число по формуле (15), например $10010011 = 1 * 2^7 + 1 * 2^4 + 1 * 2^1 + 1 * 2^0 = 147$ (DEC). Для перевода числа из двоичной системы в 16 - ную, его необходимо разбить начиная справа на группы по 4 двоичных цифры и в каждой четверке просуммировать веса (8,4,2,1) соответствующие единичным значениям C_i . Для обратного перевода каждая HEX цифра заменяется четверкой двоичных, незначащие нули слева, если они есть, отбрасываются.

Двоично-десятичное число также, как и шестнадцатиричное записывается четверками двоичных, но вес каждой четверки не 16^i , а 10^i . Двоично-десятичное число (BCD) можно записывать и десятичными цифрами, например 1998, и двоичными - $0001\ 1001\ 1001\ 1000 = 1 * 10^3 + 9 * 10^2 + 9 * 10^1 + 8 * 10^0$. Каждое десятичное число можно представить в виде BCD, например $19(DEC) = 19(BCD)$, но их двоичные представления не равны: $10011(19DEC)$ не равно $1\ 1001(19BCD)$. Не каждая запись из нулей и единиц является двоично-десятичным числом. Например, $11001001(BIN) = [C9(HEX), 201(DEC)] = ?9(BCD)$, т.к. десятичной цифры 1100=12 не существует.

2.2 ДЕШИФРАТОР

Дешифратор (ДШ) преобразует двоичный код на входах в активный сигнал на том выходе, номер которого равен десятичному эквиваленту двоичного кода на входах. В полном дешифраторе количество выходов $N = 2^n$, где n - число входов. В неполном ДШ $N < 2^n$. По другому определению, полный ДШ имеющий n входов должен реализовывать 2^n минтермов u_i ,

определенных на всех наборах из n - входных переменных. Большинство дешифраторов снабжено одним или несколькими дополнительными входами разрешения выходов (прямыми OE или инверсными \sim OE). Если на таком входе активный уровень, то ДШ работает в соответствии со своим определением, а если пассивный, то на всех выходах также будут пассивные значения (для прямых выходов - нули, а для инверсных - единицы).

Таблица 3

DEC число	Входы				Выходы				лф
	a1	a0	OE	\overline{OE}	y0	y1	y2	y3	
0	0	0			1	0	0	0	$y_0 = OE * \sim a_1 * \sim a_0$
1	0	1	1	0	0	1	0	0	$y_1 = OE * \sim a_1 * a_0$
2	1	0			0	0	1	0	$y_2 = OE * a_1 * \sim a_0$
3	1	1			0	0	0	1	$y_3 = OE * a_1 * a_0$
x	x	x	0	1	0	0	0	0	$y_i = 0$ ($\sim y_i = 1$)

Рассмотрим ДШ с $n=2$ и $N=4$, называемый также дешифратором "2 в 4" или "2 -> 4". Активным уровнем сигнала на прямых входах/выходах будет 1, а на инверсных - 0. По этому определению заполним таблицу истинности (табл. 3), где величина x может принимать любые значения. На рис.16 приведена таблица Карно (рис.16) для выхода y_0 и 3-х входных переменных.

		a1 a0			
	OE	00	01	11	10
0		0	0	0	0
1		1	0	0	0

Рис.16. Таблица Карно для выхода y_0 дешифратора

Прямоугольник, составленный из 1-ных клеток содержит только одну такую клетку, поэтому логическая функция выхода y_0 будет иметь вид: $y_0 = OE * \sim a_1 * \sim a_0$.

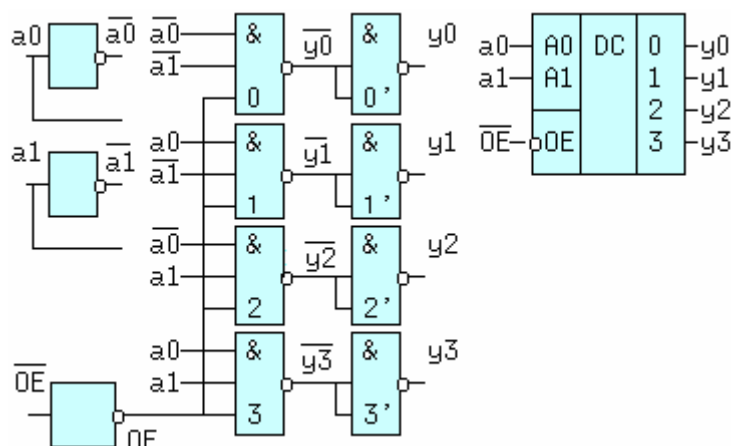


Рис.17. Дешифратор "2->4"

Аналогично получены остальные три уравнения. Преобразуем полученные для y_i уравнения с помощью аксиомы двойного отрицания к базису И-НЕ: $y_0 = \sim(\sim(OE * \sim a_1 * \sim a_0))$. Решению соответствует схема, приведенная на рисунке 17. Вместо инвертора OE, может применяться более сложная схема, показанная на рисунке 18. Здесь $OE=1$ в случае, когда $\sim OE_1 = \sim OE_2 = 0$ и $OE_3 = 1$.

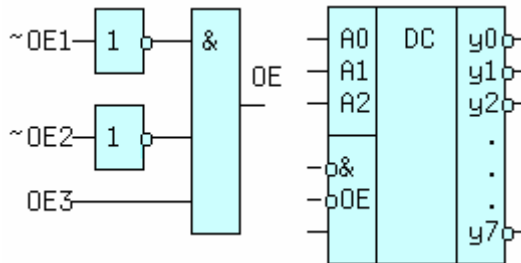


Рис.18. Дешифратор “3->8”

Такая схема применяется в дешифраторе “3 в 8” типа 1533ИД7 с инверсными выходами, условное обозначение которого приведено на рисунке.

Дешифраторы широко применяются в вычислительной технике, как часть больших интегральных схем, для выбора одного из нескольких внешних устройств (ВУ) при обмене данными между ним и микропроцессором (CPU) см. рис. 19. В этом случае на входы A_i дешифратора микропроцессор (МП) по шине адреса (ША) помещает адрес ВУ, поэтому такие входы называются адресными. Активный сигнал $\sim Y_i=0$ с одного из выходов ДШ поступает на инверсный вход $\sim CS_i$ “выбор микросхемы - Chip Select” адресуемого ВУ, подключая его к шине данных (ШД). Вслед за этим МП производит обмен данными с выбранным ВУ по шине данных. Все остальные ВУ пассивны, т.к. на их входах $\sim CS_j=1$ и их информационные выходы, подключенные к ШД, находятся в третьем состоянии и не препятствуют обмену данными между адресованным ВУ и МП. Шина - группа проводников, имеющих одинаковое функциональное назначение.

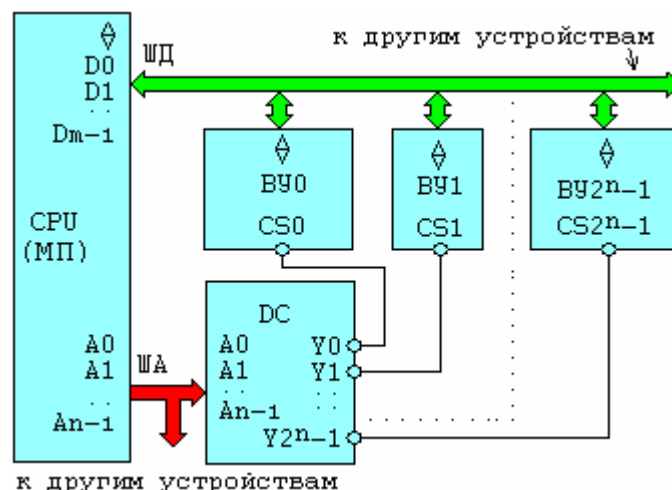


Рис.19. Применение дешифратора в МП технике

2.3 ДЕМУЛЬТИПЛЕКСОР

Устройство передающее сигнал с информационного входа "x" на один из выходов, причем номер этого выхода равен десятичному эквиваленту двоичного кода на адресных входах, называется демультиплексором (ДМ). В качестве ДМ используется дешифратор, у которого на вход OE подается информационный сигнал x. Например, если на адресные входы подать код $a_1a_0=10(\text{BIN})=2(\text{DEC})$, то сигнал x появится на выходе y_2 , а на остальных выходах $y_i=0$. В самом деле, если $x=0$ ($\text{OE}=0$), то на всех выходах ДМ также нули, в том числе $y_2=0$. А если $x=1$ ($\text{OE}=1$), то вследствие того, что $a_1a_0=2$ будет активизирован выход $y_2=1$. Следовательно при $a_1a_0=10(\text{BIN})=2(\text{DEC})$ сигнал на выходе $y_2=x$, что соответствует определению демультиплексора. На рисунке 20 даны условное обозначение ДМ "1 в 4" и его механический аналог.

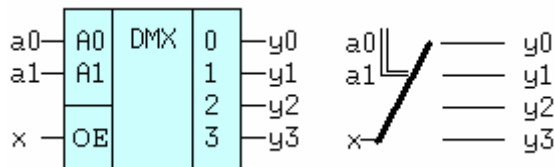


Рис.20. Демультиплексор "1->4"

2.3.1 УВЕЛИЧЕНИЕ РАЗРЯДНОСТИ ДЕШИФРАТОРОВ И ДЕМУЛЬТИПЛЕКСОРОВ

На рис.20-1 показано соединение двух ДШ "3 в 8" для получения одного ДШ "4 в 16", или двух демультиплексоров "1 в 8" для получения одного "1 в 16".

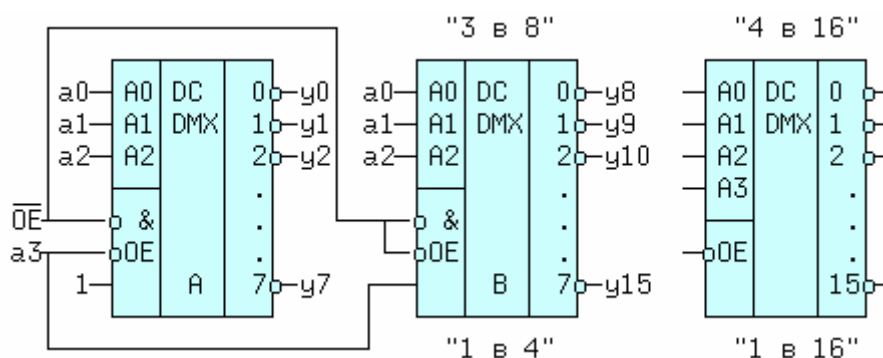


Рис.20-1. Увеличение числа разрядов дешифраторов

При пассивном значении $\sim\text{OE}=1$ сигнал $\text{OE}=0$ и на всех выходах y_i будет "1", независимо от значений сигналов a_i . Если $\sim\text{OE}=0$ (активный уровень), то какой из дешифраторов работает зависит только от сигнала a_3 . Так если $a_3=0$, то на всех разрешающих входах ДШ (A) будут активные уровни, а на прямом разрешающем входе OE дешифратора (B) сигнал a_3 равный нулю переведет все выходы в состояние 1 (см. табл. 3-1), т.е.

работоспособным будет ДШ (А). Когда $a_3=1$, наоборот работоспособным становится дешифратор (В), потому что для инверсного входа $\sim OE$ дешифратора (А) этот сигнал запрещает его выходы. Комбинации сигналов a_3 и $a_2..a_0$, это видно из первых колонок таблицы, образуют последовательность двоичного кода 0000 ... 0111 (0 ... 7) для дешифратора (А) и последовательность 1000 ... 1111 (8 ... 15) для ДШ (В). Поэтому нумерация выходов y_i , получившегося ДШ "4 в 16" сквозная от 0 до 15. На рис.20-1, справа приведено условное обозначение, получившегося дешифратора - демультиплексора (в справочниках они так часто и называются и помещаются в один раздел).

Таблица 3-1

DEC # Вых	a3 a2 a1 a0				y0 y1 y2 ... y7				y8 y9 y10 ... y15						
					A				B						
0	0	0	0	0	0	0	1	1	...	1	1	1	1	...	1
1		0	0	1	1	1	0	1	...	1	1	1	1	...	1
2		0	1	0	1	1	1	0	...	1	1	1	1	...	1
...	
7	1	1	1	1	1	1	1	...	0	1	1	1	...	1	
8	1	0	0	0	0	1	1	1	...	1	0	1	1	...	1
9		0	0	1	1	1	1	1	...	1	1	0	1	...	1
10		0	1	0	1	1	1	1	...	1	1	1	0	...	1
...	
15	1	1	1	1	1	1	1	...	1	1	1	1	...	0	

2.4 МУЛЬТИПЛЕКСОР

Мультиплексор (или мультиплексор-селектор) передает сигнал с одного из информационных входов x_i на единственный выход y , причем номер этого входа равен десятичному эквиваленту двоичного кода на адресных входах. Если имеется вход разрешения выхода $OE(\sim OE)$, то "0(1)" на этом входе должен перевести выход в пассивное состояние, если выход мультиплексора тристабильный, то на этом выходе установится Z – состояние. Рассмотрим мультиплексор "4 в 1", имеющий 4 информационных входа и $\log_4 = 2$ адресных входов. Его уравнение тогда будет иметь вид:

$$y = OE(x_0 \cdot \sim a_1 \cdot \sim a_0 + x_1 \cdot \sim a_1 \cdot a_0 + x_2 \cdot a_1 \cdot \sim a_0 + x_3 \cdot a_1 \cdot a_0).$$

Применяя аксиомы двойного отрицания и двойственности к правой части уравнения получим: $y = \sim(\sim(OE \cdot x_0 \cdot \sim a_1 \cdot \sim a_0) \cdot \dots \cdot \sim(OE \cdot x_0 \cdot a_1 \cdot a_0))$. Этому выражению соответствует схема, приведенная на рисунке 21,

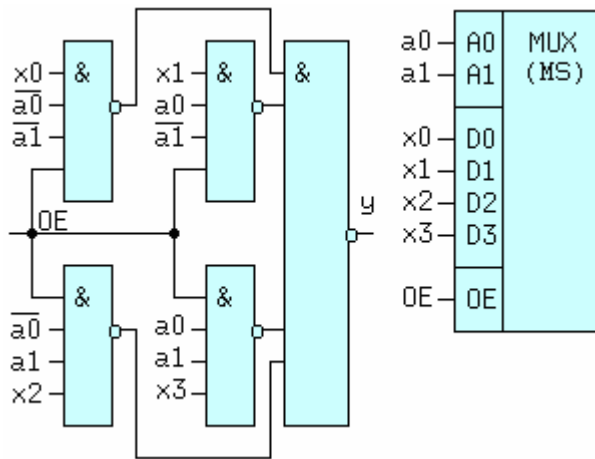


Рис.21. Мультиплексор "4->1"

Если на адресные входы подать комбинацию $a_1a_0 = 11(\text{BIN}) = 3(\text{DEC})$, то к выходу y будет подключен вход D_3 , при условии, что $OE=1$. Мультиплексор может иметь инверсный выход, а также третье состояние этого выхода, которое отмечается на схеме ромбом с поперечной чертой.

На рисунке 21-1 показан способ соединения 5-ти мультиплексоров "4->1" в один мультиплексор "16->1".

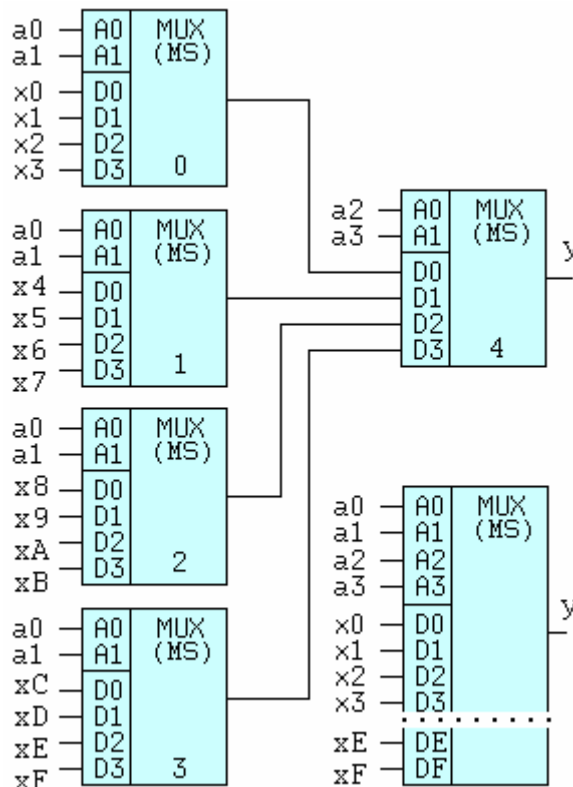


Рис.21-1. Увеличение числа разрядов мультиплексоров

Пример: на входах $A_3A_2A_1A_0$ действует комбинация $1011(\text{BIN}) = B(\text{HEX}) = 11(\text{DEC})$. Тогда на входы $D_3D_2D_1D_0$ мультиплексора №4 будут "стучаться" сигналы с третьих входов ($a_1a_0=11=3$) остальных мультиплексоров - x_3, x_7, x_B и x_F . Но на общий выход "y" пройдет только

сигнал с выхода мультиплексора №2, т.к. $a_3a_2=2$. В результате $y=xB$, что соответствует определению мультиплексора.

Мультиплексоры находят широкое применение в технике связи, а также в вычислительной технике, например многие выводы у микропроцессоров "мультиплексированы", т.е. к одному выводу подключается несколько внутренних источников различных сигналов. Это могут быть внутренние сигналы линий шины данных (D7..D0) и шины адреса (A7..A0), передаваемые поочередно на общие выводы AD7..AD0 совмещенной шины адрес/данные (ШАД), что позволяет сократить общее число выводов микропроцессора (на рисунке 22 в 2 раза). Используются 8 мультиплексоров "2->1"

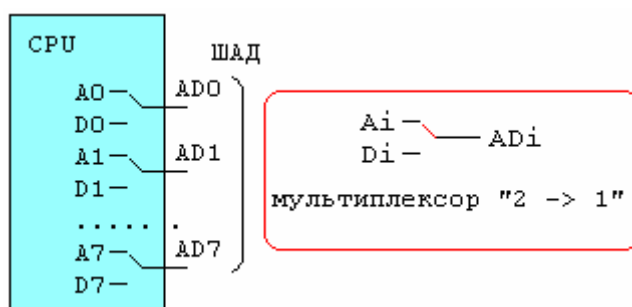


Рис.22. Мультиплексирование шин адреса и данных

Аналоговые мультиплексоры могут передавать сигналы как в прямом, так и в обратном направлении, то есть фактически являются мультиплексорами-демультиплексорами.

2.5 ШИФРАТОР

Шифратор (Ш) может быть неприоритетным, если допускается подача только одного активного сигнала и может быть приоритетным, если допускается подача одновременно нескольких активных сигналов на входы. Неприоритетный Ш осуществляет преобразование десятичного номера активного входа в двоичный эквивалент этого номера. Для неприоритетного шифратора "4 в 2" таблица истинности имеет вид (табл. 4):

Таблица 4

номер входа	входы				выходы	
	x0	x1	x2	x3	y1	y0
0	1	0	0	0	0	0
1	0	1	0	0	0	1
2	0	0	1	0	1	0
3	0	0	0	1	1	1

В приоритетном Ш производится преобразование максимального десятичного номера активного входа в двоичный эквивалент этого номера. Для такого Ш входные сигналы, лежащие снизу от единичной диагонали, по определению могут иметь любое значение ("х" может быть 0 или 1). Комбинация 0000 на входах не определена.

Для синтеза схемы неприоритетного Ш для каждого выхода составим таблицу Карно. Четыре входных переменных дают $2^4=16$ комбинаций из которых по определению заданы в таблице только 4. Остальные 12 неопределенных (запрещенных) комбинаций в таблицах Карно отметим символом Φ . Так как появление этих комбинаций на входах не предусмотрено (по определению), то в соответствующие клетки т.Карно можно подставлять любые значения, в том числе такие, которые позволяют наиболее полно минимизировать ЛФ. Ниже приведена таблица Карно (рис.23) для выхода y_0 . Величины Φ доопределены до 1.

		$x_2 \times x_3$			
		00	01	11	10
$x_0 \times x_1$	00	Φ	1	Φ	0
	01	1	Φ	Φ	Φ
	11	Φ	Φ	Φ	Φ
	10	0	Φ	Φ	Φ

Рис.23. Таблица Карно для выхода y_0 шифратора

Аналогично заполняется ТК для выхода y_1 . Из полученных таблиц находим y_1 и y_0 :

$$y_1 = \sim x_0 * \sim x_1 = \sim(x_0 + x_1) \quad \text{и} \quad y_0 = \sim x_0 * \sim x_2 = \sim(x_0 + x_2).$$

Реализация и условное обозначение неприоритетного Ш приведены на рисунке 24.

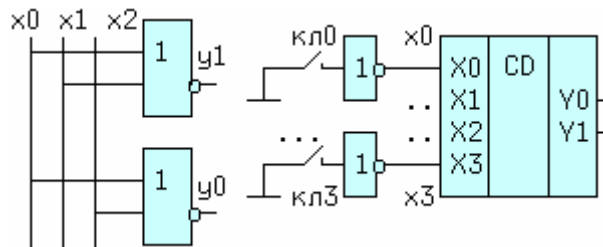


Рис.24. Неприоритетный шифратор "4->2"

Переменная x_3 оказалась "обделенной", но это произошло из-за того, что если нет сигнала ни на одном из первых трех входов, то он неизбежно должен присутствовать, по определению, на оставшемся, т.е. на третьем.

Переменная x_3 , вместе с остальными может быть использована для формирования функции $x_0 + x_1 + x_2 + x_3$ равной 0, когда не активен ни один из входов, что может сигнализировать, например о неисправности источников сигналов.

Шифраторы применяются в контроллерах прерываний работы микропроцессора внешними устройствами, в быстродействующем параллельном преобразователе напряжения в код (АЦП) и для кодирования номера клавиши. Последнее применение показано на рисунке. Если нажата клавиша кл3, то на выходе ей будет соответствовать код 11(BIN) = 3(DEC).

2.6 ПРЕОБРАЗОВАТЕЛИ КОДА

Преобразователи кодов (ПК) могут быть весовыми и невесовыми. Весовые ПК преобразуют информацию из одной системы счисления в другую. Основное назначение невесовых - преобразование информации для ее дальнейшего отображения. По ГОСТ'у преобразователь обозначается наклонной чертой, где в "числителе" стоит то что, а в "знаменателе" то во что преобразуется (в нашем случае некоторый код X преобразуется в некоторый код Y). В качестве примера рассмотрим преобразователь двоично-десятичного кода в код для семисегментных светодиодных индикаторов. На рисунке 25 также показан фрагмент подключения одного светодиода (сегмента) к выходу преобразователя с открытыми коллекторами и приведены начертания первых пяти цифр.

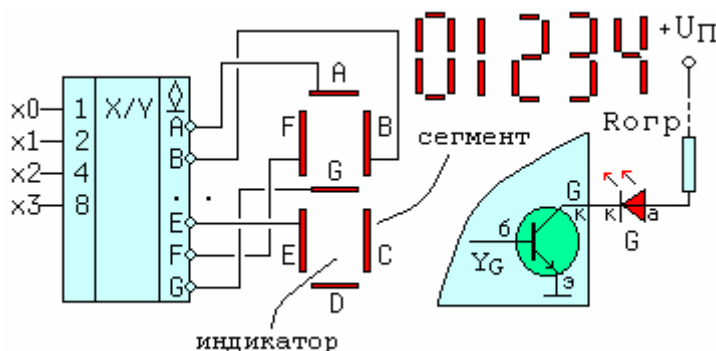


Рис.25. Преобразователь кода

Такой преобразователь должен иметь четыре входа, т.к. для кодирования десятичных цифр от 0 до 9 достаточно четырех двоичных, и семь выходов, по одному на каждый сегмент.

Сформулируем условия свечения/гашения светодиода: 1) Светодиод "горит", если напряжение на его аноде больше, чем на катоде (о конкретных значениях напряжения и тока пока речь не идет). Анод через ограничивающий ток резистор уже подключен к плюсу источника питания, поэтому на катоде должен быть потенциал близкий к нулю. Для этого n-p-n

транзистор, работающий в ключевом режиме, должен быть открыт. Тогда потенциал его коллектора близок к нулю. Транзистор открыт, если потенциал на его базе больше нуля, т.е. должно быть $Y_G = 1$ (Y_G - логическая переменная, соответствующая сегменту G).

2) Светодиод погашен, если потенциалы его анода и катода равны. Это достигается, если ключевой транзистор закрыт и через него не протекает ток. Потенциал базы в этом случае должен быть равен нулю, т.е. $Y_G = 0$.

Теперь в соответствии с полученными условиями заполним таблицу истинности преобразователя. Например в цифре 0 должны светиться все сегменты за исключением сегмента G. В цифре 1 светятся только два сегмента B и C и т.д. Весовые коэффициенты b^i двоично-десятичных разрядов равны 2^i (8,4,2 и 1). На рис.26 слева дана таблица истинности. В таблице заполнена только колонка для сегмента A. Нули в ней проставлены для тех цифр, в которых сегмент A не светится.

Десятичная цифра	веса				сегменты		
	x_3	x_2	x_1	x_0	Y_A	Y_B	Y_G
0	0	0	0	0	1	1	0
1	0	0	0	1	0	1	0
2	0	0	1	0	1	.	.
3	0	0	1	1	1	.	.
4	0	1	0	0	0	.	.
5	0	1	0	1	1	.	.
6	0	1	1	0	1	.	.
7	0	1	1	1	1	.	.
8	1	0	0	0	1	.	.
9	1	0	0	1	1	.	.

Рис.26. Таблица истинности и схема выхода Y_A

В общем случае для синтеза этого ПК требуется составить семь уравнений. Найдем одно, для сегмента A, заполнив сначала для него таблицу Карно. На рис.27 приведена ТК прямого значения функции сегмента A. Когда нулевых клеток в таблице значительно меньше и они компактно сгруппированы, целесообразно искать алгебраическое выражение инверсной логической функции, т.е. $\sim Y_A$. Логическая функция при этом может получиться значительно проще, т.е. содержать меньше переменных и слагаемых. Шесть значений ЛФ в таблице не определены (Ф) из-за отсутствия десятичных цифр больших девятки, поэтому для минимизации доопределяем некоторые из них единицами. Из таблицы найдем: $\sim Y_A = x_2 \cdot \sim x_1 \cdot \sim x_0 + \sim x_3 \cdot \sim x_2 \cdot \sim x_1 \cdot x_0$. Тогда искомое выражение будет иметь вид: $Y_A = \sim(x_2 \cdot \sim x_1 \cdot \sim x_0 + \sim x_3 \cdot \sim x_2 \cdot \sim x_1 \cdot x_0)$.

		x_1, x_0		$\sim y_a$	
		00	01	11	10
x_3, x_2	00	0	1	0	0
	01	1	0	0	0
	11	Φ	Φ	Φ	Φ
	10	0	0	Φ	Φ

Рис.27. Таблица Карно выхода Y_A

Соответствующая ему реализация на элементе И-ИЛИ-НЕ типа 1533ЛР4 будет одной из самых простых (рис. 26). Промышленность выпускает микросхемы преобразователей 514ИД2 с открытым коллектором и 514ИД1 с открытым эмиттером.

2.7 СУММАТОРЫ

Сумматор осуществляет арифметическое суммирование n -разрядных кодов.

$X = (x_{n-1}, \dots, x_0)$ - 1-е слагаемое

$Y = (y_{n-1}, \dots, y_0)$ - 2-е слагаемое

 $S = (s_{n-1}, \dots, s_0)$ - сумма

Правила сложения двух одноразрядных двоичных чисел ("+" - арифметическое сложение):

$$0 + 0 = 0$$

$$0 + 1 = 1 + 0 = 1$$

$$1 + 1 = 0 \text{ и перенос } 1 \text{ в старший разряд.}$$

Устройство реализующее эти правила называется одноразрядным полусумматором и имеет два входа и два выхода. Сложение трех одноразрядных чисел производится следующим образом ("+" - арифметическое сложение):

$$0 + 0 + 0 = 0$$

$$0 + 0 + 1 = 1$$

$$0 + 1 + 1 = 0 \text{ и перенос } 1 \text{ в старший разряд}$$

$$1 + 1 + 1 = 1 \text{ и перенос } 1 \text{ в старший разряд.}$$

Устройство реализующее эти правила называется одноразрядным полным сумматором (ОПС) и имеет три входа и два выхода. Таблица истинности (таблица 5) ОПС приведена на рисунке, слева.

Таблица 5

	x_i	y_i	c_i	s_i	$c(i+1)$
0	0	0	0	0	0
1	0	0	1	1	0
2	0	1	0	1	0
3	0	1	1	0	1
4	1	0	0	1	0
5	1	0	1	0	1
6	1	1	0	0	1
7	1	1	1	1	1

x_i, y_i - одноименные двоичные разряды чисел X и Y, c_i - перенос из предыдущего разряда, s_i - частичная сумма по модулю два и $c(i+1)$ - перенос в следующий разряд. Значения $c(i+1)$ совпадают со значениями функции мажоритарности, поэтому воспользуемся готовым решением:

$$c(i+1) = x_i * y_i + x_i * c_i + y_i * c_i.$$

Таблица Карно для s_i приведена на рис.26 справа. Из таблицы находим: $s_i = x_i * \sim y_i * \sim c_i + \sim x_i * \sim y_i * c_i + x_i * y_i * c_i + \sim x_i * y_i * \sim c_i = \sim y_i (x_i * \sim c_i + \sim x_i * c_i) + y_i (x_i * c_i + \sim x_i * \sim c_i) = \sim y_i (x_i (+) c_i) + y_i (x_i * c_i + \sim x_i * \sim c_i)$. Выражение в последней скобке необходимо преобразовать, используя соотношение двойственности:

$$x_i * c_i + \sim x_i * \sim c_i = \sim(\sim(x_i * c_i) * \sim(\sim x_i * \sim c_i)) = \sim((\sim x_i + \sim c_i) * (x_i + c_i)) = \sim(\sim x_i * x_i + \sim x_i * c_i + \sim c_i * x_i + \sim c_i * c_i) = \sim(\sim x_i * c_i + x_i * \sim c_i) = \sim(x_i (+) c_i) = \sim F6 = F9.$$

С учетом последнего выражения $s_i = \sim y_i (x_i (+) c_i) + y_i \sim(x_i (+) c_i) = y_i (+) (x_i (+) c_i) = y_i (+) x_i (+) c_i$.

Операция (+) называется - сумма по модулю два (переключательная функция F6 для двух аргументов). Схема одноразрядного полного сумматора и ее условное обозначение приведены на рисунке 28.

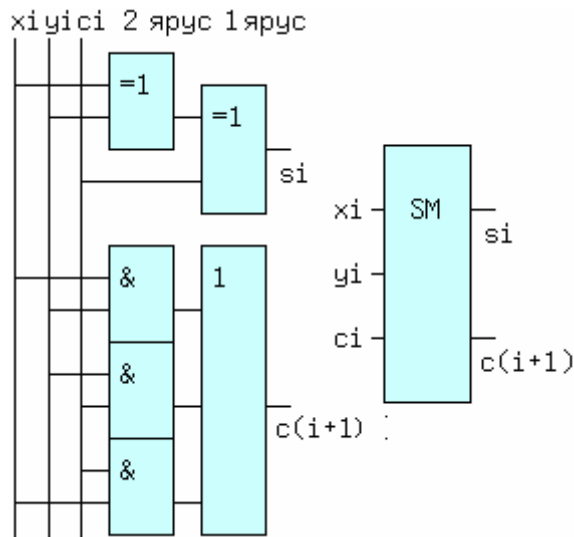


Рис.28. Схема ОПС

Сумматор с последовательным переносом для сложения n- разрядных двоичных чисел показан на рисунке 29. К его недостатку относится большое время задержки, в наихудшем случае, когда от сложения x_0, y_0 возникает сквозной перенос через все разряды до выхода $s(n-1)$. При двухъярусной схеме одноразрядного сумматора, задержка сигнала от входов до выходов составит $2t_{зд.р.}$, если считать задержку в каждом ярусе одинаковой. Суммарная величина задержки будет равна: $t_{зд.р. посл. сумматора} = n * 2t_{зд.р.}$

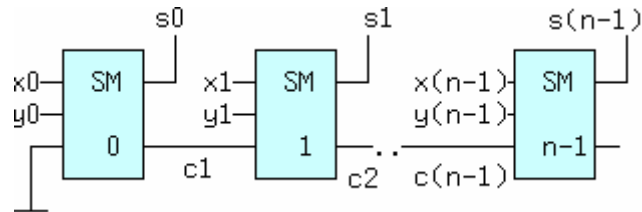


Рис.29. Схема n-разрядного сумматора

При сложении многоразрядных чисел задержка выходного сигнала на выходе последнего разряда становится недопустимо большой.

В ЭВМ сумматор является центральным узлом арифметико-логического устройства (АЛУ) и от его быстродействия зависит производительность компьютера. Поэтому применяются сумматоры с параллельной схемой переноса.

Выражение для младшего разряда можно преобразовать, используя тождество для функции ИЛИ: $x + y = \sim x * y + x * \sim y + xy$. В правой части равенства СДНФ ф-ии ИЛИ.

Тогда $c_1 = x_0 * y_0 + x_0 * c_0 + y_0 * c_0 = x_0 * y_0 + c_0(x_0 + y_0) = x_0 * y_0 + c_0(\sim x_0 * y_0 + x_0 * \sim y_0 + x_0 * y_0) = x_0 * y_0(c_0 + 1) + c_0(\sim x_0 * y_0 + x_0 * \sim y_0) = x_0 * y_0 + c_0(x_0 (+) y_0)$.

Уравнениям для s_0 и c_1 соответствует схема на рис.29-1.

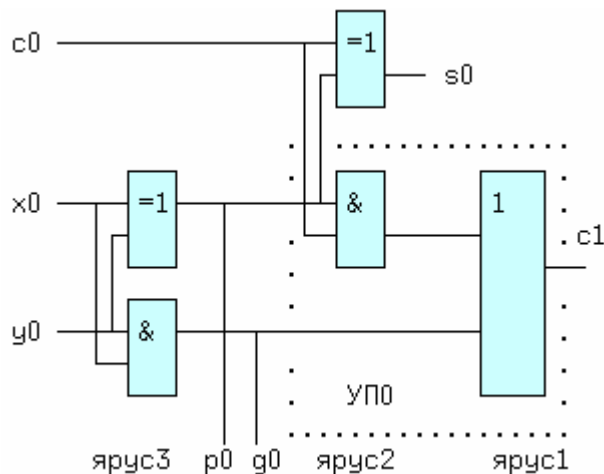


Рис.29-1. Схема младшего разряда сумматора

Если в каждом разряде сумматора использовать такой одноразрядный сумматор, то никакого выигрыша в скорости не будет. Узел обведенный точками называется узлом переноса (УП), а функции g_i и p_i называются функциями генерации переноса и распространения переноса. С учетом этого можно записать:

$$c_1 = g_0 + p_0 * c_0,$$

$$c_2 = g_1 + p_1 * c_1 = g_1 + p_1 * g_0 + p_1 * p_0 * c_0,$$

$$c_3 = g_2 + p_2 * c_2 = g_2 + p_2 * g_1 + p_2 * p_1 * g_0 + p_2 * p_1 * p_0 * c_0,$$

и так далее. Последние выражения соответствуют сумматору с параллельным переносом, т.к. величина c_i снимается с выхода предыдущего разряда, в котором она формируется параллельно из всех первичных переменных. Схемы узлов переноса УП1 и УП2 приведены на рис.29-2.

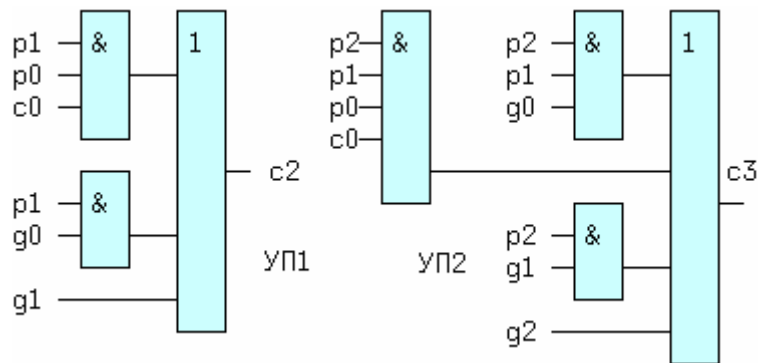


Рис.29-2. Схемы узлов переноса 1-го и 2-го разрядов

Из рис.29-1 и 29-2 видно, что узел сложения в каждом разряде остается неизменным, а изменяется только узел переноса, причем задержка сигнала от входов x_i, y_i до c_{i+1} остается неизменной и для 3-ярусной схемы равна $3t_{зд.р.}$. Суммарная задержка в каждом разряде увеличится на время прохождения сигнала от входа c_i до s_i , т.е. на величину $t_{зд.р.}$, и составит: $t_{зд.р.паралл.сумматора} = 4t_{зд.р.}$ независимо от количества разрядов. За это приходится платить усложнением узла переноса от разряда к разряду.

2.8 СХЕМА СРАВНЕНИЯ КОДОВ

Два кода X и Y считаются равными, если равны их одноименные разряды. Можно ввести функцию $F(X==Y)$, которая равна 1, если $x_i=y_i$ для всех i , иначе ее значение равно нулю. В качестве примера возьмем два двухбитовых числа $X=(x_1,x_0)$ и $Y=(y_1,y_0)$. Таблица Карно для этих чисел приведена на рисунке 30, справа.

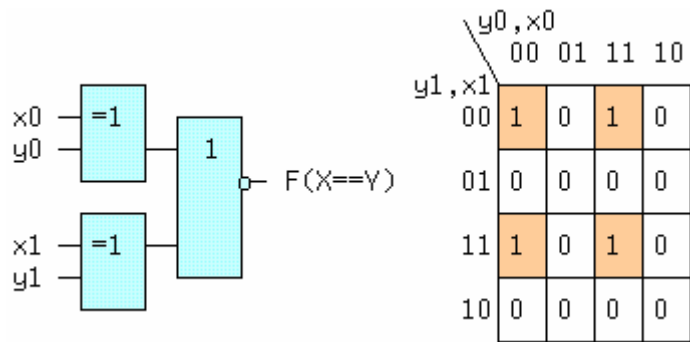


Рис.30. Таблица Карно и схема сравнения

$$F(X==Y) = \sim y_1 \cdot \sim x_1 \cdot \sim y_0 \cdot \sim x_0 + \sim y_1 \cdot \sim x_1 \cdot y_0 \cdot x_0 + y_1 \cdot x_1 \cdot \sim y_0 \cdot \sim x_0 + y_1 \cdot x_1 \cdot y_0 \cdot x_0 = \sim y_1 \cdot \sim x_1 (\sim y_0 \cdot \sim x_0 + y_0 \cdot x_0) + y_1 \cdot x_1 (\sim y_0 \cdot \sim x_0 + y_0 \cdot x_0) = \sim (x_0 (+) y_0) \cdot \sim (x_1 (+) y_1) = F_9(x_0, y_0) \cdot F_9(x_1, y_1) = \sim (F_6(x_0, y_0) + F_6(x_1, y_1)).$$
 Преобразования в последних двух строчках сделаны с учетом, того что $\sim F_6(x, y) = F_9(x, y)$ и наоборот (см. раздел сумматоры). Одна из возможных реализаций приведена на рисунке, слева. Практические схемы дополняются функциями "больше/меньше", как например в микросхеме 555СП1, которая сравнивает два четырехразрядных числа. На рисунке 31 показано соединение двух таких схем, для увеличения разрядности сравниваемых чисел до восьми.

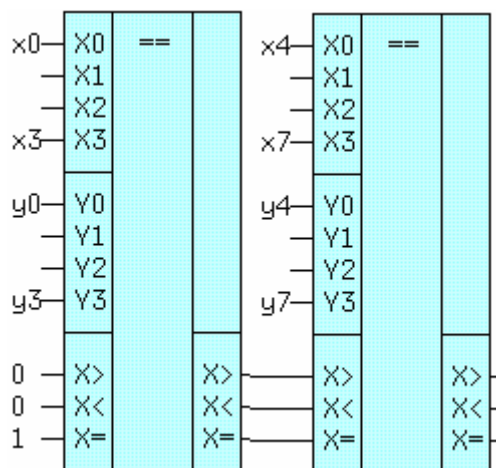


Рис.31. Каскадное соединение схем сравнения

Для правильного результата сравнения чисел $X = (x_7, x_6, \dots, x_0)$ и $Y = (y_7, y_6, \dots, y_0)$ на вход $X =$ необходимо подать 1. Схема сравнения так же, как и сумматор входит в состав АЛУ микропроцессора и часто называется цифровым компаратором.

2.9 СХЕМА КОНТРОЛЯ ЧЕТНОСТИ (НЕЧЕТНОСТИ)

Схема применяется для выявления ошибок, вызванных помехами в линии связи или в блоках памяти. Метод основан на подсчете числа единиц в передаваемой в линию или направляемой в память на хранение порции информации, причем если число единиц четное - функция четности $P(\text{arity})$

равна нулю. Для четырехразрядного двоичного числа таблица Карно, схемная реализация и условное обозначение приведены на рисунке 32.

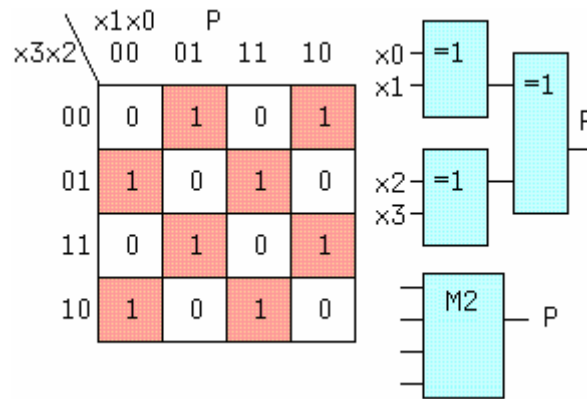


Рис.32. Схема контроля четности

Символом M2 обозначена операция - "сумма по модулю два". Четыре строки таблицы Карно дают 4 составляющих: $P = \sim x_3 \cdot \sim x_2 \cdot F_6(x_1, x_0) + \sim x_3 \cdot x_2 \cdot \sim F_6(x_1, x_0) + x_3 \cdot x_2 \cdot F_6(x_1, x_0) + x_3 \cdot \sim x_2 \cdot \sim F_6(x_1, x_0) = F_6(F_6(x_3, x_2), F_6(x_1, x_0)) = (x_3 (+) x_2) (+) (x_1 (+) x_0)$. Рассмотрим пример на рисунке 33.

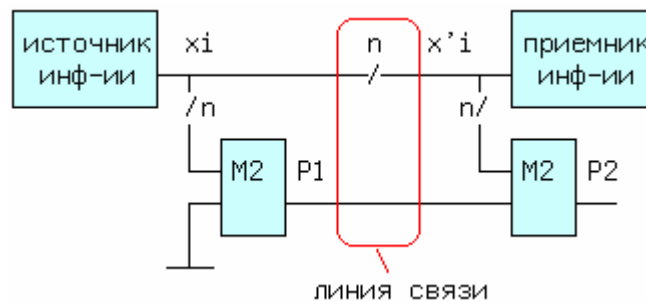


Рис.33. Передача данных по линии связи

Пусть по n-проводной линии связи передается параллельный двоичный код $x^{(n-1)}, x^{(n-2)}, \dots, x_1, x_0$, а принимается код $x'^{(n-1)}, x'^{(n-2)}, \dots, x'_1, x'_0$. Тогда величина $P_1 = 0 (+) x_0 (+) x_1 (+) \dots (+) x^{(n-1)}$. На приемном конце линии связи $P_2 = x'_0 (+) x'_1 (+) \dots (+) x'^{(n-1)} (+) P_1$. Подставляя в последнюю формулу выражение для P1 и группируя переменные в одноименные пары, получим: $P_2 = (x_0 (+) x'_0) (+) (x_1 (+) x'_1) (+) (x_2 (+) x'_2) (+) \dots$. Из последнего выражения следует, что если передача прошла без искажений, то $x_i = x'_i$ и $x_i (+) x'_i = 0$, а $P_2 = 0$! При искажении одного и в общем случае нечетного числа бит функция $P_2 = 1$. Аналогично протекает процесс контроля и при последовательной передаче по одной линии связи n-бит и одного бита четности, такая схема используется в коммуникационных СОМ-портах компьютеров.

3. ПОСЛЕДОВАТЕЛЬНОСТНЫЕ СХЕМЫ

В последовательностных схемах (ПС) выходные сигналы зависят не только от комбинаций входных, но и от значений самих выходных сигналов в предшествующий момент времени. Для работы ПС принципиальное значение имеет время задержки распространения тзд.р. Простейшей ПС является триггер.

3.1 ТРИГГЕРЫ

Триггером называют последовательностную схему с положительной обратной связью и двумя устойчивыми состояниями 0 и 1 (то есть триггер обладает свойством памяти). В общем случае триггер может иметь 1)асинхронные входы предварительной установки, 2)тактовый или синхронизирующий и 3)информационные входы. К основным типам триггеров (выпускаемых промышленно и использующихся в САПР'ах) относятся:

- RS-триггер с отдельной установкой состояний,
- D - триггер "защелка",
- JK - универсальный триггер,
- T - триггер со счетным входом.

По способу записи информации триггеры подразделяются на асинхронные и синхронные или тактируемые, а по способу управления - на триггеры со статическим управлением (обычно высоким уровнем тактового сигнала) и триггеры с динамическим управлением (положительным - из 0 в 1, или отрицательным - из 1 в 0 фронтом тактового сигнала). В последнем случае говорят о триггерах с прямым или инверсным динамическим входом управления.

3.1.1 АСИНХРОННЫЙ RS - ТРИГГЕР

Асинхронный триггер имеет два входа S(et) - установка и R(eset) - сброс и два выхода прямой - Q и инверсный - $\sim Q$. Триггер переходит из текущего состояния X на выходе к состоянию Q=0, при подаче на вход S нуля и на вход R единицы, а при поступлении на вход S единицы и на вход R нуля триггер переходит к состоянию Q=1. При нулевых значениях, когда S=R=0 триггер должен сохранять старое значение. Комбинация сигналов S=R=1 не определена. В соответствии с описанием составим таблицу состояний триггера (табл. 6).

Таблица 6

Текущее сост	Последующее состояние		Название режима	
	S R Q ^t	Q ^{t+dt}		$\overline{Q^{t+dt}}$
0 0 Q		Q	$\sim Q$	Хранение инф-ии (память)
1 0 X		1	0	Установка в "1" (установка)
0 1 X		0	1	Уст-ка в "0" (сброс, очистка)
1 1 X		Ф	Ф	Неопределенное состояние

Q и X - могут принимать любые значения, но Q в пределах одной строки, неизменно. Неопределенные, на этапе описания, значения "Ф" будут доопределены на этапе минимизации. Входных переменных три - S,R и текущее состояние выхода Q^t, поэтому всего должно быть восемь состояний при различных значениях Q и X. Последующее состояние выходов отделено от текущего, временем задержки сигнала (интервалом времени) dt.

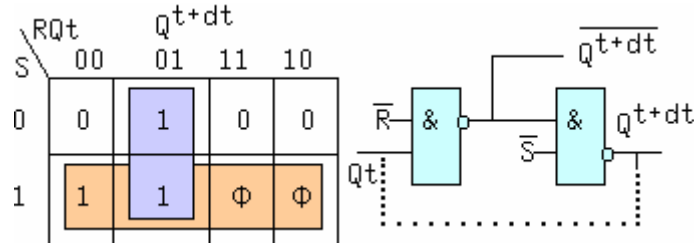


Рис.34. Асинхронный RS-триггер

Таблица Карно функции $Q(t+dt)$ с учетом всех возможных состояний приведена на рисунке 34, слева. Доопределяем значения "Ф" единицей и находим характеристическое уравнение RS - триггера:

$$Q(t+dt) = S + \sim R * Q^t = \sim(\sim S * \sim(\sim R * Q^t)).$$

Этому уравнению соответствует схема на рисунке, справа. Учитывая, что Q^t и Q(t+dt) сигналы на одном и том же выходе, но в разные моменты времени, свяжем их. Окончательно схема триггера и его условное обозначение будет выглядеть следующим образом (рис.35):

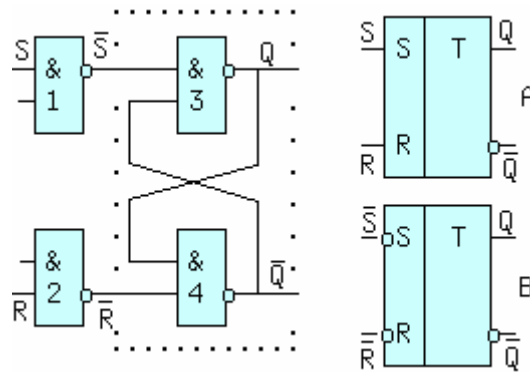


Рис.35. RS-триггер с прямыми и инверсными входами

Условное обозначение В соответствует части рисунка, обведенной точками, т.е. RS триггеру с инверсными входами, а обозначение А - всему рисунку, или RS триггеру с прямыми входами.

Если на триггер с прямыми_входами подать сигналы $R=S=1$ или на входы инверсного - нули, то оба выхода Q и $\sim Q$ будут установлены в 1, что противоречит аксиоме $Q \cdot \sim Q = 0$. Поэтому такой режим, иногда называют запрещенным. Однако ничто не мешает разработчику использовать его, например для сигнализации об одновременном и нежелательном поступлении единичных сигналов на RS входы, введением дополнительной схемы "И".

Временные диаграммы RS триггера с инверсными входами приведены на рис.36. Через время $t_{зд.р.}$,равное интервалу t_1-t_0 , от поступления сигнала $\sim S = 0$ на вход элемента И-НЕ с номером 3, выход Q переключится первым, а следом через такой же промежуток времени переключится и выход $\sim Q$. Спустя интервал времени $t_2 - t_0 = dt$ на выходах установятся новые значения. Также протекает процесс при $\sim R = 0$, но выход $\sim Q$ переключится первым. Отсюда вытекает, что изменение входных сигналов не должно происходить быстрее времени dt .

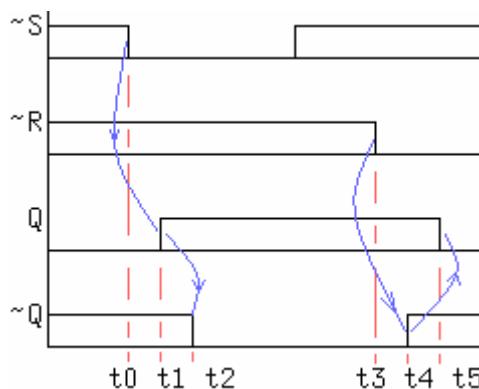


Рис.36. Временные диаграммы асинхронного RS-триггера

RS триггер используется как основа для создания остальных типов триггеров. Одним из применений RS триггера с инверсными входами служит схема подавления "дребезга" контактов клавиатуры. Процесс многократного

размыкания и замыкания контактов при их переключении называется дребезгом. Схема и диаграммы показаны на рисунке 37.

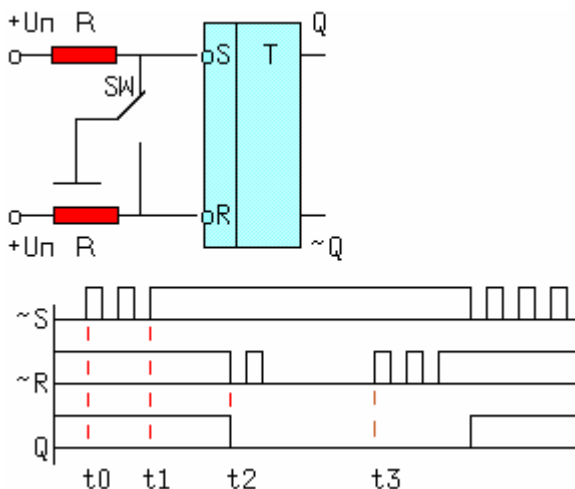


Рис.37. Схема подавления “дребезга” контактов

В момент t_0 нажатия на клавишу, начинаются соударения верхнего и среднего контактов. До момента t_1 сигналы $\sim S, \sim R$ поочередно принимают значения 1,1 и 0,1 что соответствует режимам памяти и установки в 1. При этом, естественно начальное значение $Q = 1$ не изменится, что и требуется. В интервале $t_1..t_2$ средний контакт находится в "свободном полете". Первое его касание нижнего контакта в момент t_2 сбросит триггер ($\sim S = 1, \sim R = 0$). До момента t_3 сигналы $\sim S, \sim R$ поочередно принимают значения 1,0 и 1,1 что соответствует режимам сброса и памяти, т.е. $Q = 0$. При отпускании клавиши (момент t_3) развивается обратный процесс. В результате действия схемы выходной сигнал чист от импульсных помех.

3.1.2 СИНХРОННЫЙ RS - ТРИГГЕР

Если незадействованные входы элементов И-НЕ 1 и 2 соединить вместе получится синхронный RS - триггер со статическим управлением (синхронизируемый уровнем). Схема и условное обозначение приведены на рисунке 38.

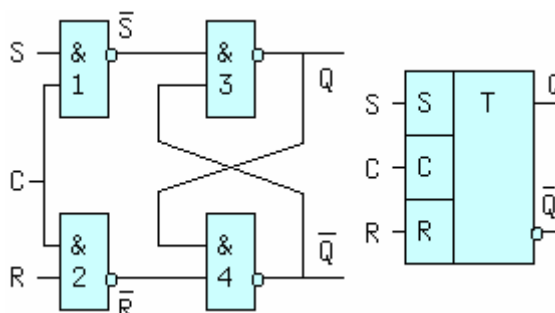


Рис.38. Синхронный RS-триггер

Нетрудно убедиться, что при $C=0$ сигнал $Q = \sim(\sim Q * 1) = Q$, а $\sim Q = \sim(Q * 1) = \sim Q$, т.е. независимо от значений S и R, выходы сохраняют старые значения и триггер

находится в режиме памяти. При $C=1$ он функционирует, как асинхронный RS-триггер. Триггеры со статическим управлением называют, также "прозрачными", т.к. при активном уровне синхросигнала C , информация с входов беспрепятственно проходит на выходы. Временные диаграммы приведены на рисунке 39.

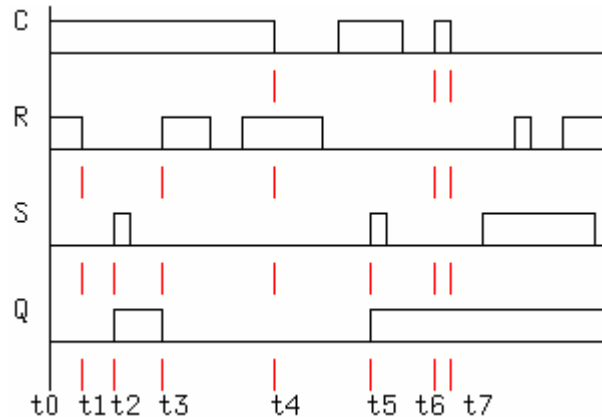


Рис.39. Временные диаграммы синхронного RS триггера

До момента времени t_4 сигнал $C = 1$ и выходное значение определяется комбинациями сигналов R и S . В течение интервала времени $t_0...t_1$ на входе R действует 1, а сигнал $S = 0$, поэтому Q тоже равно 0. Начиная с момента t_1 и до момента t_2 $R = S = 0$ и действует режим памяти (Q не изменяется). В момент t_2 $R = 0$, а $S = 1$ и триггер устанавливается ($Q = 1$). С момента окончания импульса S и до момента t_3 триггер хранит эту единицу, а в момент t_3 сбрасывается, т.к. $R = 1$, а $S = 0$. Аналогично можно проанализировать и все остальные состояния выхода.

3.1.3 D - ТРИГГЕР СО СТАТИЧЕСКИМ УПРАВЛЕНИЕМ

D - триггер имеет как минимум два входа: информационный вход $D(ata)$ и вход управления записью/запоминанием (защелкиванием) $L(oad)/L(atch)$ - отсюда его второе имя : "защелка". Последний вход часто обозначают символом $C(Clock)$. Выходной сигнал Q принимает значение равное входному $Q(t+dt) = D$ при $L = 1$ и сохраняет предыдущее значение $Q(t+dt) = Q_t$ при $L = 0$. Таблица состояний триггера имеет вид (табл. 7):

Таблица 7

Текущее состояние		Последующее состояние		Название режима
L	D	Q_t	$Q(t+dt)$ $\sim Q(t+dt)$	
0	X	Q	Q $\sim Q$	Хранение инф-ии(режим памяти)
1	0	X	0 1	Уст-ка в "0" (сброс, очистка)
1	1	X	1 0	Установка в "1" (установка)

Q и X - могут принимать любые значения, но Q в пределах одной строки, неизменно. Таблица Карно дана на рисунке. Связующий импликант добавлен для получения схемы свободной от "гонок" (см. раздел - переходные процессы) и от инверсии входных величин. $Q(t+dt) = \sim L * Q_t + L * D + D * Q_t = L * D + Q_t(\sim L + D) = L * D + Q_t((\sim L + D) * (\sim L + L)) = L * D + Q_t * ((\sim L + D * \sim L) + D * L) = L * D + Q_t * (\sim L + D * L) = \sim(\sim(L * D) * \sim(Q_t * (D * L + \sim L))) = \sim(\sim(L * D) * \sim(Q_t * \sim(D * L) * L))$.

Этой формуле соответствует схема и условное обозначение на рисунке 40, в центре и справа.

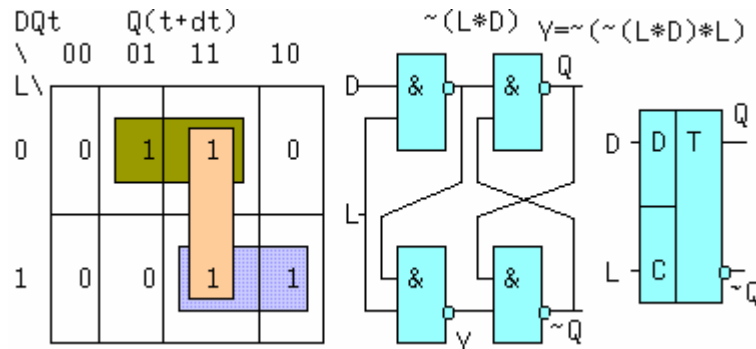


Рис.40. Статический D-триггер

Если в уравнение вместо $\sim(D * L) * L$ подставить $\sim(D * L) * L = (\sim D + \sim L) * L = \sim D * L$ получится реализация D-триггера с использованием RS-триггера, но появляется дополнительный инвертор. Ниже на рисунке 41 приведена схема такого триггера, дополненная асинхронными инверсными входами установки и сброса $\sim S$ и $\sim R$ (эти две перекрестные связи показаны двойными линиями). Если на вход $\sim S$ подать активный сигнал 0, а на вход $\sim R$ единицу, то $Q=1$ независимо от сигналов на остальных входах элемента 3. На выходе 2-го элемента по той же причине тоже единица. Три единицы встречаясь на входах элемента 4, дают на его выходе ноль, который попадая на вход 3-го элемента подтверждает его состояние.

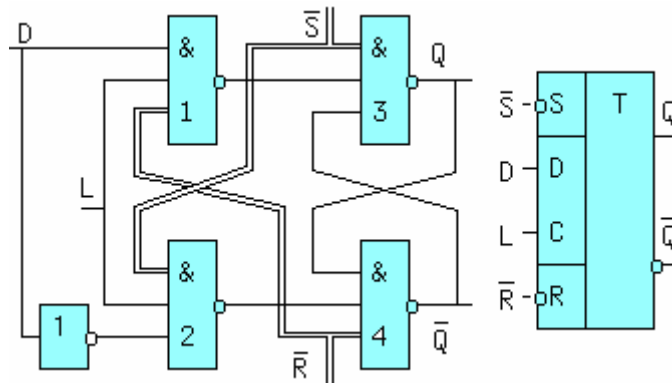


Рис.41. Статический D-триггер с асинхронными входами

Триггер устанавливается в единицу. Причем сигналы D и L не влияют на этот процесс. В силу этого, асинхронные входы ($\sim S$ и $\sim R$) имеют **наивысший**

приоритет. Вследствие симметричности асинхронных связей, аналогично протекает процесс при $\sim S=1$ и $\sim R=0$, но триггер, естественно сбрасывается ($Q = 0$). Уравнение синхронного D-триггера с асинхронными входами сброса/установки $\sim S$ и $\sim R$ записываются в следующем виде:

$$Q(t+dt) = S + \sim R * (\sim L * Q_t + L * D + D * Q_t).$$

В этом выражении до скобок записано уравнение асинхронного RS-триггера, а в скобках уравнение D-триггера. Нетрудно увидеть, что при $\sim S = 0$ ($S = 1$) и $\sim R = 1$ все выражение равно единице (установка триггера в "1"), а при $\sim S = \sim R = 1$, RS-триггер "отключается" и схема функционирует, как D-триггер. Временные диаграммы работы триггера приведены на рис. 42.

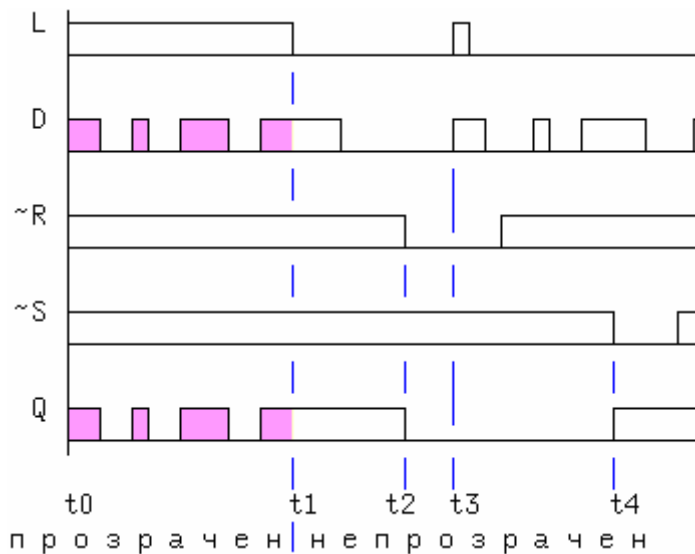


Рис.42. Временные диаграммы статического D-триггера

С момента времени t_0 до момента t_1 сигнал загрузки $L = 1$ (на входах $\sim R$ и $\sim S$ пассивный уровень) и данные с входа D беспрепятственно проходят на выход Q (свойство прозрачности D-триггера со статическим управлением видно особенно наглядно), поэтому статические D-триггеры еще называют **прозрачными**. В момент t_1 триггер становится непрозрачным, информация защелкивается и последнее значение на выходе будет храниться до прихода нулевого уровня на вход $\sim R$ в момент t_2 . Состояние $Q = 0$ не изменится даже при $L = D = 1$ в момент t_3 . Триггер установится только в момент t_4 по сигналу $\sim S = 0$. Если вернуться к рис.40 и убрать из условного обозначения триггера вход S , получится повторитель и инвертор, как на рис. 43, и эта схема не будет обладать свойствами памяти.

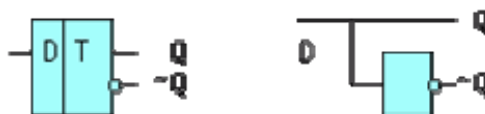


Рис.43. Несуществующий асинхронный D-триггер

Поэтому асинхронных D - триггеров в природе не существует и определение "синхронный" по отношению к D-триггеру является избыточным.

3.1.4 D - ТРИГГЕР С ДИНАМИЧЕСКИМ УПРАВЛЕНИЕМ

Отличается от статического D-триггера свойствами L(C) входа (см. таблицу 8). Запись информации происходит только в момент перехода тактового сигнала L из 0 в 1 ($Q(t+dt)=D$ при $L=\uparrow$). При постоянном значении $L=0$, $L=1$ или отрицательном перепаде триггер хранит предыдущую информацию ($Q(t+dt)=Q_t$). Такой триггер не обладает свойством прозрачности при $L=1$. Промышленно выпускаемые триггеры дополняются асинхронными инверсными входами установки и сброса $\sim S$ и $\sim R$.

Таблица 8

Текущее состояние		Последующее состояние		Название режима
\bar{S} \bar{R}	L D Qt	Q^{t+dt}	\bar{Q}^{t+dt}	
1 1	0, 1, \uparrow X Q	Q	$\sim Q$	Хранение инф-ии Установка в "0" Установка в "1"
	\downarrow 0 X	0	1	
	\downarrow 1 X	1	0	
0 1	X X X	1	0	Установка в "1" Установка в "0" "Неопред." сост
1 0	X X X	0	1	
0 0	X X X	1	1	

D-триггер
RS-триггер

Q и X - принимают любое значение, но Q в пределах одной строки, неизменно. На схемах (рис. 44) прямые динамические входы D- триггера обозначаются наклонной чертой "слева - снизу - вправо - вверх" или стрелкой внутрь. Из табл. 8 следует, что при активизации хотя-бы одного сигнала $\sim R = 0$ или $\sim S = 0$ схема работает, как RS - триггер с инверсными входами. И только при пассивных значениях $\sim R = \sim S = 1$ схема функционирует, как D - триггер.

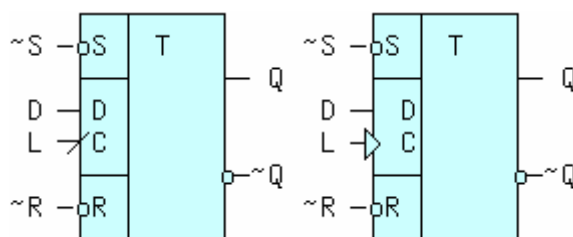


Рис.44. Условное обозначение динамического D-триггера

Временные диаграммы (рис. 45) триггеров с динамическим входом существенно изменяются. Действие асинхронных входов такое же, как в D-триггере со статическим управлением, поэтому на временных диаграммах они не указаны.

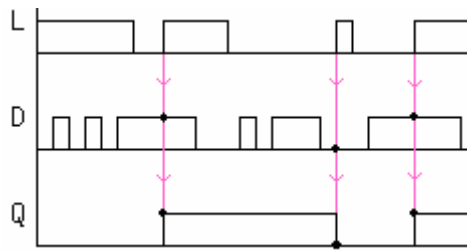


Рис.45. Временные диаграммы

Из диаграмм наглядно видно, что выходное значение может измениться только при положительном фронте тактового сигнала (если на асинхронных входах пассивные значения).

3.1.5 УНИВЕРСАЛЬНЫЙ JK-ТРИГГЕР

JK-триггер имеет два информационных входа J и K, тактовый динамический вход, чаще инверсный, и два асинхронных входа установки и сброса. Его таблица состояний (табл. 9) имеет вид:

Таблица 9

Текущее состояние		Последующее состояние		Название режима
\bar{S} \bar{R}	C J K Qt	Q^{t+dt}	\bar{Q}^{t+dt}	
1 1	0, 1, \downarrow X X Q	Q	\bar{Q}	JK триггер
	\downarrow 0 0 Q	Q	\bar{Q}	
	\downarrow 1 0 X	1	0	
	\downarrow 0 1 X	0	1	
	\downarrow 1 1 Q	\bar{Q}	Q	
0 1	X X X X	1	0	RS триггер
1 0	X X X X	0	1	
0 0	X X X X	1	1	

Q и X - принимают любые значения, но Q в пределах одной строки, неизменно. Запись информации, при пассивных уровнях сигналов сброса ($\sim R$) и установки ($\sim S$), осуществляется только в моменты перехода сигнала C из 1 в 0, за исключением триггера типа ТВ15, который переключается положительным фронтом, т.е. JK - триггеры являются непрозрачными. Уравнение JK-триггера с асинхронными входами $\sim R$ и $\sim S$:

$$Q(t+dt) = S + \sim R (J*\sim Q_t + \sim K*Q_t). \quad (27.1)$$

Три равноправных условных обозначения JK-триггера с инверсным динамическим входом приведены на рисунке 46. Наклонная черта "смотрит слева - направо - сверху - вниз", а стрелка повернута наружу. Универсальность JK - триггера будет продемонстрирована далее.

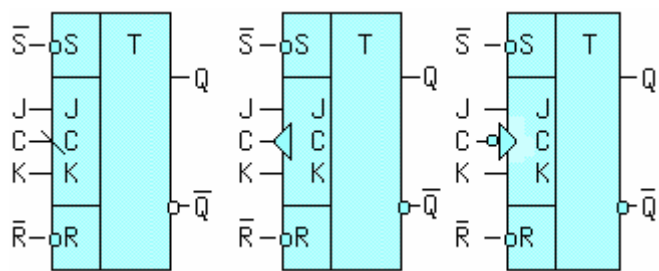


Рис.46. Равноправные условные обозначения JK-триггера

Вторая, третья и четвертая строчки таблицы состояний идентичны соответствующим строчкам RS-триггера, если вход J уподобить входу S, а вход K - входу R. Отличие в том, что комбинация $J = K = 1$ определена и триггер в этом режиме приобретает очень полезное свойство, при поступлении каждого отрицательного фронта на вход C, меняет значение сигнала на выходе, как показано на рисунке 47.

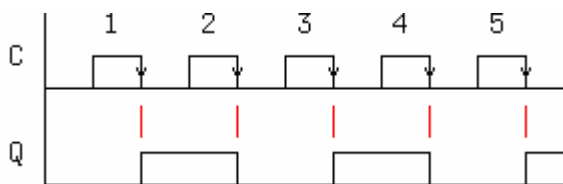


Рис.47. Временные диаграммы при $J=K=1$

3.1.6 Т - ТРИГГЕР

Триггер со счетным входом или Т - триггер промышленностью не выпускается, а реализуется с помощью динамического D, или JK - триггеров. Например для JK триггера анализ временной диаграммы (рис.47) при $J = K = 1$ позволяет сделать два важных вывода. Во-первых, период повторения выходных импульсов увеличился в два раза, значит триггер в этом режиме делит частоту входных импульсов на два. Во-вторых, с приходом четного импульса выходной сигнал равен 0, а с приходом нечетного равен 1, т.е. триггер является счетчиком по модулю два.

3.1.7 ВЗАИМНЫЕ ПРЕОБРАЗОВАНИЯ ТРИГГЕРОВ

JK триггер преобразуется в динамический D - триггер подключением инвертора к входу K, при этом из четырех комбинаций сигналов: $J=K=0$, $J=K=1$, $J=0 K=1$, $J=1 K=0$ осуществлены будут две последних, т.е. синхронные установка и сброс. Если необходим прямой синхровход, к входу C подключается еще один инвертор. На рис. 48 справа JK триггер включен по схеме Т - триггера со счетным входом ($J=K=1$). Счет можно прервать, подав на один из асинхронных входов 0.

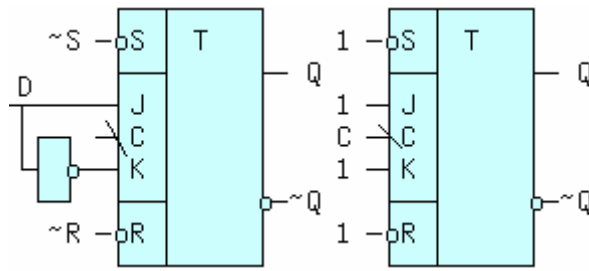


Рис.48. Преобразование JK-триггера в D и T триггеры

D-триггер с динамическим управлением (рис. 49) также преобразуется в T-триггер, путем введения обратной связи с инверсного выхода на вход D. Тогда $Q(t+dt) = D$, но D в свою очередь равно $D = \sim Q_t$ и, следовательно $Q(t+dt) = \sim Q_t$, т.е. новое значение на выходе триггера является инверсией старого с каждым поступлением положительного перепада тактового импульса C.

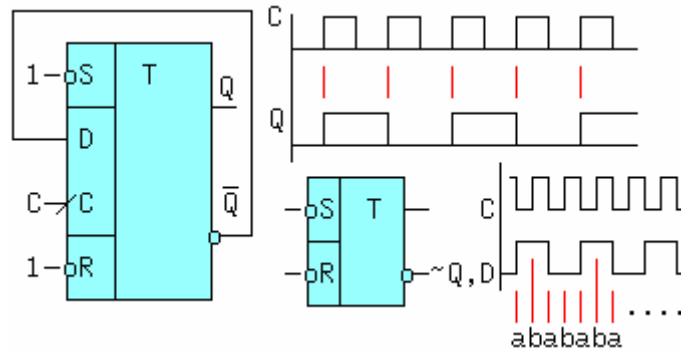


Рис.49. Преобразование D триггера в T триггер и любого из приведенных триггеров в RS триггер

И, наконец, любой из перечисленных триггеров может быть использован в качестве асинхронного RS-триггера с инверсными входами, невзирая на остальные сигналы, что объясняется наивысшим приоритетом входов $\sim S$ и $\sim R$.

Схема на рисунке 49 позволяет оценить максимальную частоту входных импульсов F_{max} на тактовом входе D-триггера в счетном режиме. В справочниках приводится задержка активного фронта сигнала от поступления на вход C до момента установления нового значения сигнала на выходе триггера $t_{зд.р.тр.} = t_{ба}$. Реже приводится время опережения установки $t_{уст} = t_{аб}$ информационным сигналом на входе D, активного фронта сигнала C. Новый активный фронт должен поступить не ранее окончания интервала времени равного сумме этих двух временных параметров. Отсюда следует, что максимальное значение $F_{макс.деления} < 1 / (t_{зд.р.тр.} + t_{уст})$. Существует еще один параметр t_H - время удержания (hold) информационного сигнала относительно тактового (синхронизирующего), однако этим временем можно пренебречь, т.к. t_H поглощается временем $t_{зд.р.тр.}$ ($t_H < t_{зд.р.тр.}$).

3.2 РЕГИСТРЫ

Последовательные схемы с различными комбинациями последовательного и параллельного способов записи и считывания информации. Выполняются на основе триггеров.

3.2.1 ПАРАЛЛЕЛЬНЫЕ И ПОСЛЕДОВАТЕЛЬНЫЕ РЕГИСТРЫ

Регистры с параллельной записью называются, также регистрами памяти. В них могут использоваться, как прозрачные "защелки", так и триггеры с динамическим управлением. На рисунке 50 приведена схема 8-ми разрядного регистра памяти с общим входом управления записью информацией и ее условное обозначение. Высокий уровень на входе С переписывает информацию с входа на выход ($Q_i = D_i$), а низкий уровень - защелкивает данные.

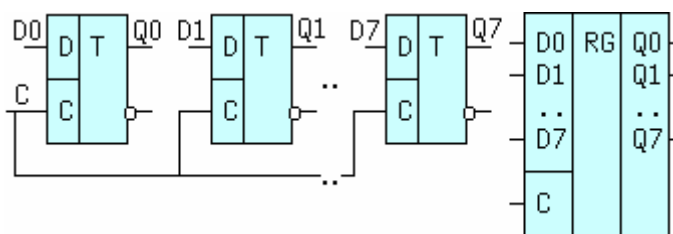


Рис.50. Параллельный регистр

Последовательные регистры (рис. 51) или, как их еще называют, регистры сдвига выполняются на основе триггеров с динамическим синхровходом, (справедливо для регистров с одним тактирующим сигналом). В двухтактных можно использовать и прозрачные регистры - "защелки". Функция записи в n-разрядном регистре сдвига на D-триггерах задается в виде условий: $D_0 = DS = x$, $D_i = Q_{(i-1)}$, где $i = 1, 2, \dots, n-1$. DS(Data Serial) - вход для последовательной записи. С приходом очередного положительного фронта синхроимпульса С, сигнал с входа i-го триггера через время $t_{зд.р.}$ окажется на его выходе и поступит на вход следующего (i+1)-го триггера.

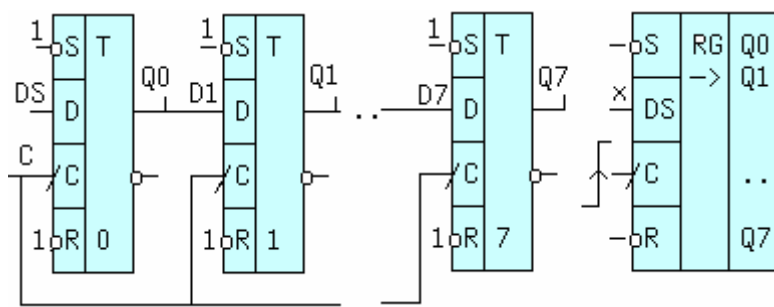


Рис.51. Последовательный регистр

Однако на его выход эта информация не переписется, т.к. длительность активного фронта $t_{0,1}$ меньше $t_{зд.р.}$. На этом процесс сдвига данных на один разряд закончится до прихода следующего положительного фронта тактового сигнала. Отсюда понятно, почему нельзя использовать триггеры со статическим управлением. Каждый раз при $C = 1$ вся цепочка окажется прозрачной от входа DS до выхода $Q7$ и значение $DS = x$ будет записано во все триггеры.

В обозначениях регистров сдвига направление стрелки, указывающей сдвиг, условно. В разных справочниках, ее направление различно. Условно принимается, что сдвиг производится от младшего разряда к старшему. Практические схемы регистров дополняются схемами, подключаемыми к каждому триггеру и имеющими вход параллельной записи D_i , общий вход разрешения записи L и общий асинхронный вход сброса $\sim R$ всех триггеров. Эти схемы подключаются к незадействованным входам $\sim R_i$, $\sim S_i$ триггеров. Данному описанию соответствует таблица истинности (табл. 10).

Таблица 10

Входы			Выходы		Название режима
L	D_i	$\sim R$	$\sim S_i$	$\sim R_i$	
X	X	0	1	0	Установка в "0" триггеров регистра
1	D_i	1	$\sim D_i$	D_i	Параллельная запись инф-ии в триггеры
0	X	1	1	1	Режим хранения или сдвига

D_i и X - могут принимать любые значения, но D_i в пределах одной строки, неизменно. Минимизируя логические функции $\sim R_i$ и $\sim S_i$ с помощью таблиц Карно получим: $\sim S_i = \sim(L * D_i * \sim R)$ и $\sim R_i = \sim R * (\sim L + D_i)$. Этой паре уравнений соответствует схема, показанная на рис. 52, где приведено также условное обозначение регистра сдвига выполненного по такой схеме. Вход $\sim R$ обладает наивысшим приоритетом, если $\sim R = 0$, то $\sim R_i = 0$, а $\sim S_i = 1$ и все триггеры обнуляются, независимо от сигналов L, D_i и C . Меньшим приоритетом обладают входы L и D_i . Если $\sim R = 1$, то при $L = 1$, производится параллельная запись информации и $Q_i = D_i$ независимо от сигнала C .

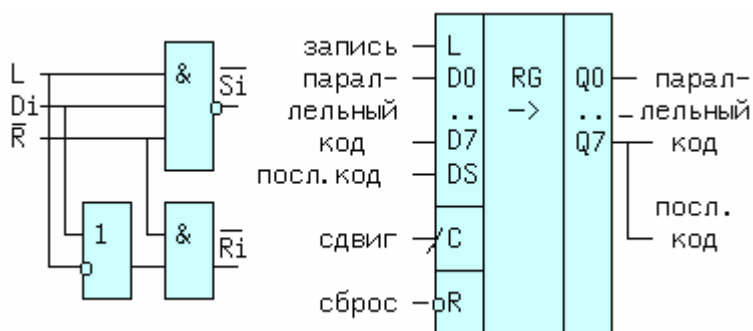


Рис.52. Обозначение регистра сдвига с входами параллельной записи

И, наконец, если на входах $\sim R$ и L пассивные уровни, то $\sim R_i = \sim S_i = 1$, тоже пассивный уровень и регистр хранит информацию, либо производит ее сдвиг.

Одно из применений регистров сдвига с параллельной загрузкой кода заключается в преобразовании параллельного формата данных в последовательный в передающем устройстве, передаче этих данных по однопроводной линии связи (вторая линия должна быть, как минимум "землей" или экраном) и обратном преобразовании последовательной информации в параллельную в приемном устройстве (рис. 53).

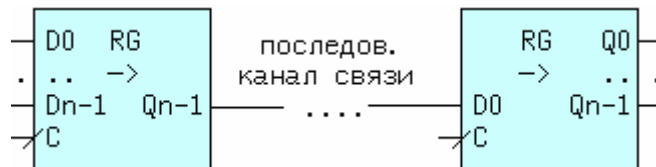


Рис.53. Применение регистров сдвига при передаче данных в последовательном формате

Такая схема применяется в коммуникационных портах ЭВМ (СОМ-порты), в микросхеме К580ИК51, а также в модемах.

3.2.2 РЕВЕРСИВНЫЙ РЕГИСТР СДВИГА

Название указывает, что сдвиг данных от разряда к разряду может производиться, как в одну сторону, так и в другую. Одна из возможных схем трехразрядного реверсивного регистра с двумя последовательными информационными входами, для сдвига информации влево DSL, и вправо DSR, с параллельными входами записи ($D_0..D_2$), синхровходом C , входом $\sim R$ установки в 0 всех триггеров и двумя входами выбора режима M_1, M_0 , приведена на рисунке 54.

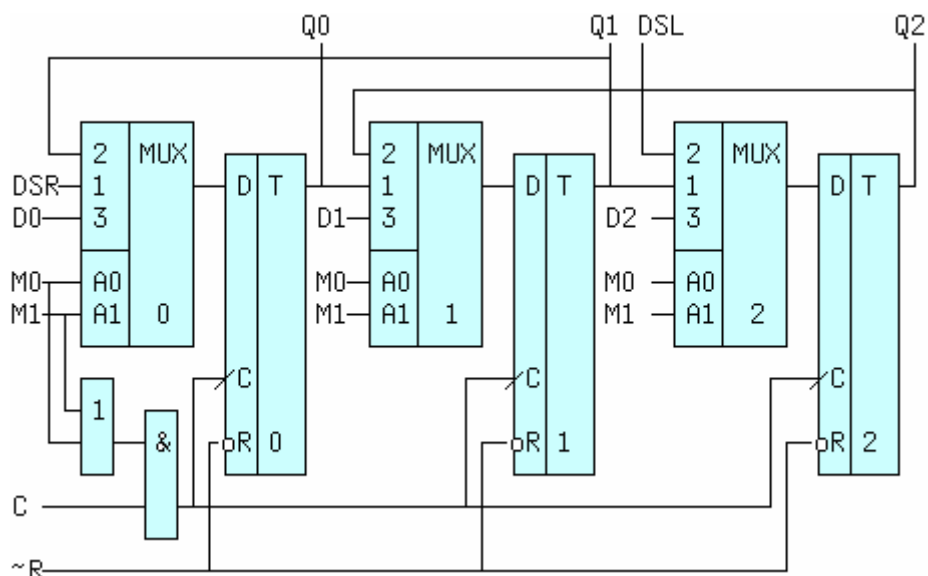


Рис.54. Реверсивный регистр сдвига

К D-входу любого разряда, за исключением крайних, подключены, через входы 1 и 2 мультиплексора, выходы и левого и правого соседних триггеров. Если $M1=0$, а $M0=1$, то к входам D подключены первые входы мультиплексоров и информация в каждый триггер, кроме нулевого записывается от левого соседа (происходит сдвиг вправо).

Вход DSR служит в этом режиме для последовательного ввода информации. Если $M1=1$, $M0=0$, то к входам D подключены вторые входы мультиплексоров и информация в каждый триггер, кроме последнего записывается от правого соседа (происходит сдвиг влево). Для последовательного ввода данных в этом режиме используется вход DSL. При $M1=M0=1$ происходит параллельная запись $Q_i=D_i$ положительным фронтом тактового сигнала. Для хранения информации необходимо подать комбинацию $M1=M0=0$. В этом случае к D-входам триггеров подключатся нулевые входы мультиплексоров (на схеме не показаны). Чтобы не произошло несанкционированной записи схема ИЛИ-И запрещает в этом режиме прохождение синхроимпульсов. По такой схеме выполнен 8-разрядный регистр 153ЗИР13. Реверсивный регистр может использоваться для быстрого (всего за n тактов) деления и умножения двоичных чисел на 2^n , где n -число сдвигов (рис.55). Например вместо числа 5, после сдвига влево на 2 разряда, стало число 20.

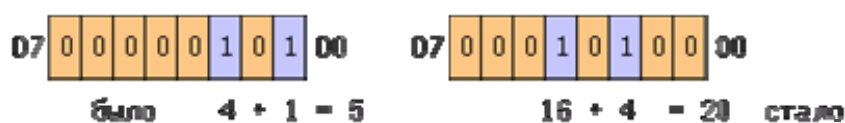


Рис.55. Применение регистра сдвига для деления и умножения

3.2.3 СИНХРОННЫЙ И АСИНХРОННЫЙ СПОСОБЫ ЗАГРУЗКИ ПАРАЛЛЕЛЬНОГО КОДА

Устройства (в том числе регистры), в которых для записи входного параллельного кода D_i используется сигнал разрешения записи L, а тактовый сигнал C не используется, называются устройствами с *асинхронной записью кода*.

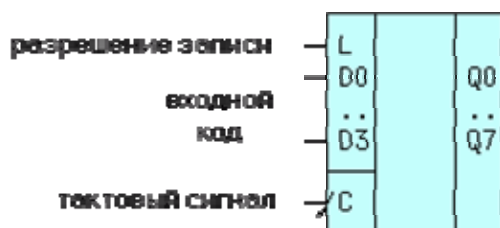


Рис.56. Способы загрузки параллельного кода

Устройства (в том числе регистры), в которых для записи входного параллельного кода D_i необходим, во-первых, сигнал разрешения записи L и, во-вторых, перепад синхросигнала на тактовом входе C - называются устройствами с *синхронной записью* кода.

Условные обозначения таких устройств обычно совпадают (рис. 56), поэтому для точной идентификации способа записи необходимо обращаться к справочникам (таблица состояний, описание или временные диаграммы).

3.3 СЧЕТЧИКИ

Счетчик (Сч) - последовательностная схема, преобразующая поступающие на вход импульсы в код Q , соответствующий их количеству. Большинство счетчиков снабжено выходом переноса CR . Для двоичного и двоично-десятичного кода, как впрочем и для других систем счисления, справедливы следующие соотношения:

$$Q = (D + \text{SUM}(C)) \bmod M$$
$$CR = (D + \text{SUM}(C)) / M$$

В этих формулах: Q - код на выходах счетчика, D - начальное значение записанное в счетчик, $\text{SUM}(C)$ - сумма импульсов поступивших на вход в процессе счета и M - модуль счета или число различных состояний счетчика (число импульсов поступивших на счетный вход, после которых счетчик возвращается в исходное состояние), CR - число импульсов переноса, возникающих на одноименном выходе при переполнениях счетчика, \bmod - операция нахождения остатка при делении на M , $/$ - операция целочисленного деления. Переполнением называется переход из максимального значения в минимальное или наоборот, при поступлении очередного импульса. Если $D = 0$ и $\text{SUM}(C) < M$, то очевидно, что $Q = \text{SUM}(C)$ и $CR = 0$. С приходом каждого M -ного импульса счетчик возвращается в исходное состояние. Пример: пусть $D = 17$, $\text{SUM}(C) = 9$, $M = 8$, тогда $Q = 26 \bmod 8 = 2$, а $CR = 3$.

Счетчики выполняются на триггерах со счетным входом (Т-триггерах). По способу счета Сч могут быть суммирующие, вычитающие и реверсивные, т.е. изменяющие направление счета. По способу переключения триггеров делятся на асинхронные и синхронные. В асинхронных счетчиках триггеры переключаются последовательно (асинхронно) от разряда к разряду, а в синхронных одновременно. Один Т-триггер обеспечивает модуль счета $M = 2$, а n триггеров дадут $M = 2^n$. При суммировании импульсов необходимо формировать перенос из i -го в $(i+1)$ -ый разряд по следующему правилу.

Правило 1: перенос CR_i из i -го в $(i+1)$ -ый разряд формируется с приходом очередного импульса, если во всех разрядах с i -го по 0-й записана

максимальная для данной системы счисления цифра, при этом разряды младше $(i+1)$ -го обнуляются. На прямых выходах триггеров этих разрядов Q_i формируется отрицательный перепад, а на инверсных - положительный.

Отсюда вытекает правило 2: Если в суммирующем счетчике используются триггеры с инверсным динамическим входом, то сигналы переноса берутся с прямых выходов предыдущих триггеров, а если триггеры с прямым динамическим входом, то сигналы переноса снимаются с инверсных выходов.

3.3.1 АСИНХРОННЫЙ СЧЕТЧИК С ПОСЛЕДОВАТЕЛЬНЫМ ПЕРЕНОСОМ

В качестве примера рассмотрим четырехразрядный счетчик (рис. 57). Четыре двоичных разряда счетчика обеспечивают $M = 16$ состояний. Ниже приведена схема и условное обозначение такого счетчика, выполненного на JK-триггерах с инверсными динамическими входами

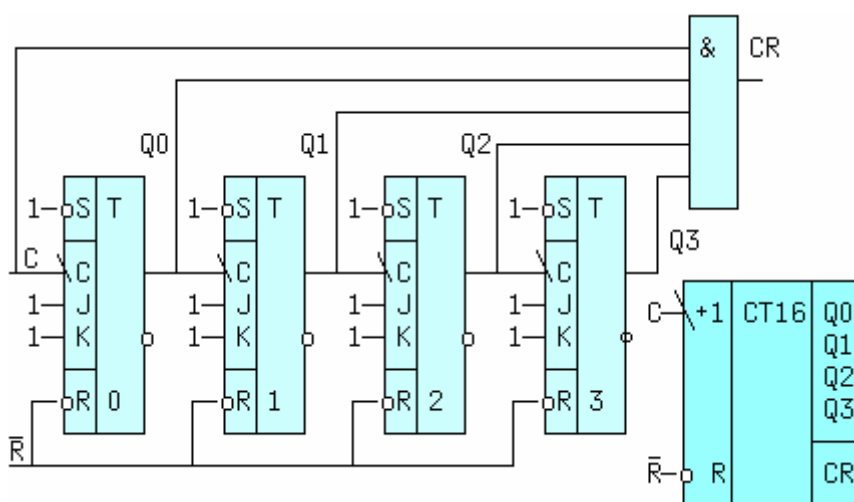


Рис.57. Асинхронный счетчик

По правилу 2 сигналы переносов должны сниматься с прямых выходов триггеров, которые переключаются последовательно друг за другом, т.е. асинхронно. Триггеры поставлены в режим счета ($J = K = 1$). Счетчик дополнен схемой формирования ускоренного переноса CR (Carry), выход которой может быть подключен к счетному входу C следующего такого же счетчика. Входы $\sim R$ всех триггеров объединены, а на входы $\sim S$ подана "лог.1", что позволяет "сбрасывать" счетчик сигналом $\sim R = 0$. Счетный вход суммирующего счетчика обозначается "+1". Идеальные временные диаграммы схемы (тзд.р.триггера = 0) построены на основе 5-ой строчки таблицы состояний JK-триггера (рис.58).

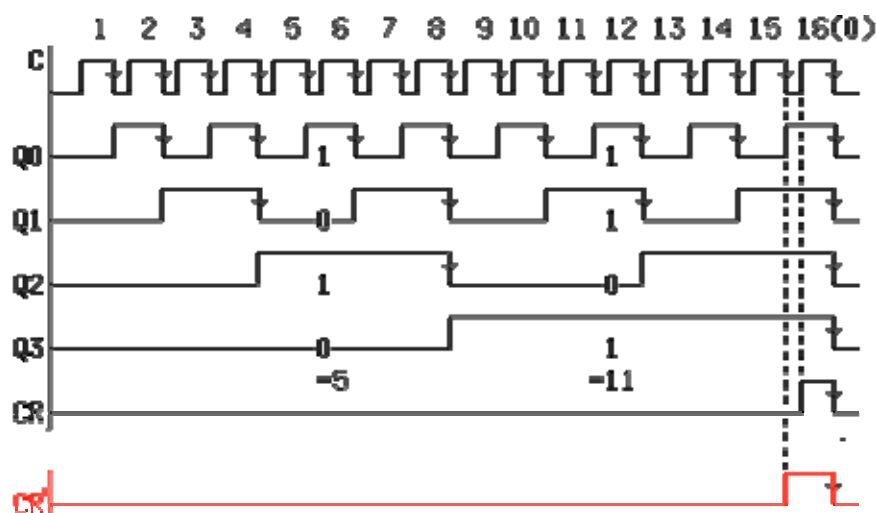


Рис.58. Идеальные временные диаграммы

Анализ временных диаграмм позволяет сделать ряд выводов:

- 1) После k -го по счету входного импульса код на выходах $Q = Q_3Q_2Q_1Q_0 = k$, например после 5-го, код $Q = 0101 = 5$, а после 11-го - $Q = 1011 = 11(DEC)$, т.е. схема действительно является счетчиком.
- 2) С приходом активного фронта 16-го импульса все триггеры обнуляются и далее процесс повторяется, т.е. модуль счета $M=16$.
- 3) Схема также является делителем частоты входных импульсов на 2 в степени $(i+1)$, где i - номер триггера, с которого снимается выходной сигнал. Причем частота сигнала на выходах CR и $Q_{n-1}(Q_3)$ совпадает и уменьшается в M - раз по отношению к входной.
- 4) Если снимать выходной код с инверсных выходов, то нетрудно заметить, что начальное значение $Q = Q_0Q_1Q_2Q_3 = 1111 = 15$, т.е. максимальному числу для четырех разрядов и далее, с приходом очередного импульса, код на выходах уменьшается на 1 .

1111	1110	1101	1100	1011
15	14	13	12	11

В этом случае счетчик называют вычитающим. Такого же результата можно добиться, если снимать переносы с противоположных выходов триггеров, а код по прежнему с прямых. Убедиться в этом нетрудно, самостоятельно построив временные диаграммы. Счетный вход вычитающего счетчика обозначается "-1".

- 5) Важно также отметить, что задний фронт импульса переноса совпадает с моментом перехода всех триггеров из 1 в 0 , для суммирующего счетчика и с моментом перехода из 0 в 1 - для вычитающего, т.е. с моментом

переполнения. В некоторых счетчиках импульс переноса имеет длительность равную периоду повторения входных импульсов (на временной диаграмме CR').

Скорость счета или максимальная частота входных импульсов определяется задержкой сигнала от момента прихода активного фронта счетного импульса до появления нового кода на выходе последнего триггера: $t_{зд.р.сч} = n * t_{зд.р.триггера}$, где n-число триггеров. Тогда $F_{макс.сч} < 1/t_{зд.р.сч}$. На рисунке 59 показаны реальные временные диаграммы ($t_{зд.р.триггера} > 0$) последовательного переключения отдельных триггеров счетчика с приходом восьмого импульса.

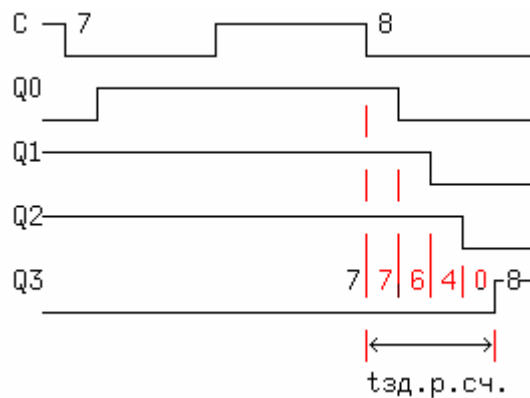


Рис.59. Фрагмент реальных временных диаграмм

Выходной код в течение интервала $t_{зд.р.сч}$ принимает недопустимые значения (6,4,0). Скорость деления частоты входного сигнала не зависит от количества триггеров и определяется максимальной частотой переключения входного триггера, т.е. $F_{макс.деления} < 1/(t_{зд.р.триггера} + t_{уст})$ и больше $F_{макс.сч}$.

3.3.2 СИНХРОННЫЙ СЧЕТЧИК С ПАРАЛЛЕЛЬНЫМ ПЕРЕНОСОМ

В синхронном счетчике импульсы поступают на тактовые входы всех триггеров одновременно. На рисунке 60 приведен трехразрядный счетчик с модулем счета $M = 8$. Справа приведено условное обозначение промышленно выпускаемого счетчика, дополненного входами D_i для параллельной синхронной загрузки начального кода перепадом сигнала на входе $C(+1)$ при $\sim L = 0$, а также входом сброса $\sim R$. Перенос из i -го разряда в $(i+1)$ -ый производится по формуле $CR_i = Q_i * \dots * Q_1 * Q_0 * (CE * PE)$ с помощью элементов "И". Входы CE (разрешение счета) и PE (разрешение переноса) на данном этапе рассмотрения могут быть объединены вместе ($CE = PE = 1$). Входы J и K триггеров соединены, поэтому возможно только два режима их работы - память ($J = K = 0$) и счет ($J = K = 1$).

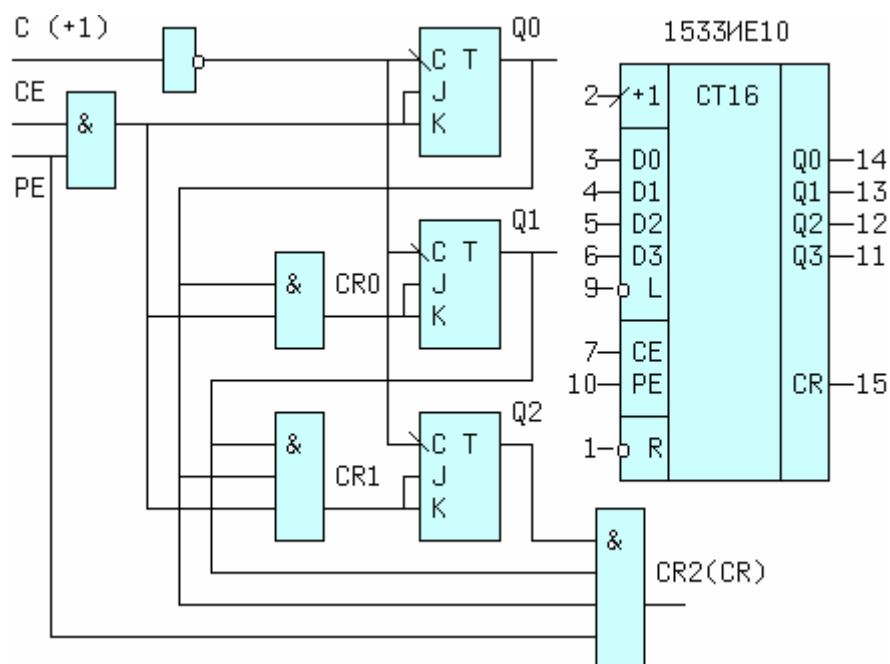


Рис.60. Счетчик с параллельным переносом

Триггер "i" изменит свое состояние только тогда, когда $CR(i-1) = J = K = 1$, т.е. при единичных значениях выходов всех предыдущих триггеров, что отвечает правилу 1. Во всех остальных случаях $CR(i-1) = J = K = 0$ и значение Q_i не изменится. Быстродействие синхронного счетчика значительно выше, из-за того, что триггеры переключаются одновременно и сигнал CR_i появится на JK входах через интервал $t_{зд.р.сч.} = t_{зд.р.т.} + t_{зд.р.э.}$, который не зависит от числа триггеров. Частота счета $F_{\max.сч.} < 1/(t_{зд.р.сч.} + t_{уст.})$. Максимальная частота деления $F_{\max.дел.} = F_{\max.сч.}$.

3.3.3 РЕВЕРСИВНЫЙ СЧЕТЧИК

Схема двухразрядного счетчика с общим входом сброса R, выходом переноса CR при суммировании и выходом переноса BR при вычитании приведена на рис. 61.

Предположим, что начальное значение кода равно 2 ($Q_1 = 1$ и $Q_0 = 0$). Рассмотрим режим суммирования, когда на входе "-1" - действует ноль, а на вход "+1" приходит импульс. Отрицательный фронт входного импульса не пройдет через верхний канал мультиплексора "2 в 1", выполненного на элементе И-ИЛИ-НЕ, этому препятствует старое значение $Q_0=0$. На входах нижнего канала действует конъюнкция единицы ($\sim Q_0 = 1$) и нуля. Поэтому сигнал на выходе мультиплексора равен нулю и триггер T1 сохраняет старое значение. Через время, равное задержке элемента ИЛИ отрицательный фронт поступит на счетный вход триггера T0, и еще через время задержки триггера на его выходах установится новое значение $Q_0 = 1$ ($\sim Q_0 = 0$). При этом верхний канал открывается для прохождения отрицательного фронта

счетного импульса, но т.к. длительность фронта $t_{1,0}$ значительно меньше суммарного времени задержки $t_{зд} = t_{зд.р.ИЛИ} + t_{зд.р.триггера}$, на входах "+1" и "-1" к этому времени действуют нули и выходной сигнал мультиплексора по прежнему равен 0. Триггер Т1 остается в прежнем состоянии. Новый код на выходах счетчика увеличился на 1 и стал равен $Q_1Q_0 = 11 = 3$, до следующего импульса.

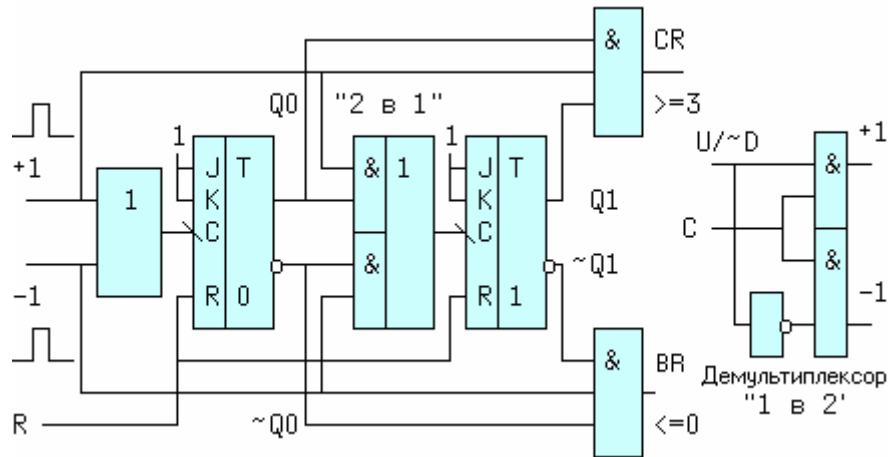


Рис.61. Реверсивный счетчик

Рассмотрим режим вычитания, когда на входе "+1" - действует ноль, а на вход "-1" приходит импульс. Отрицательный фронт входного импульса беспрепятственно пройдет через нижний канал мультиплексора, т.к. $\sim Q_0 = 1$ и переключит триггер Т1. Спустя время $t_{зд}$ новый код появится и на выходах триггера Т0, но отрицательный перепад с выхода $\sim Q_0$ не пройдет на счетный вход триггера Т1, по причине того, что сигнал на входе "-1" к этому времени уже равен нулю. Старый код 10 на выходах счетчика сменился кодом 01, т.е. уменьшился на 1.

Примерно так выполнены счетчики типа 1533ИЕ6 и 1533ИЕ7, с дополнительным асинхронным входом $\sim L$ параллельной записи начального кода. Их обозначения приведены на рисунке 62.

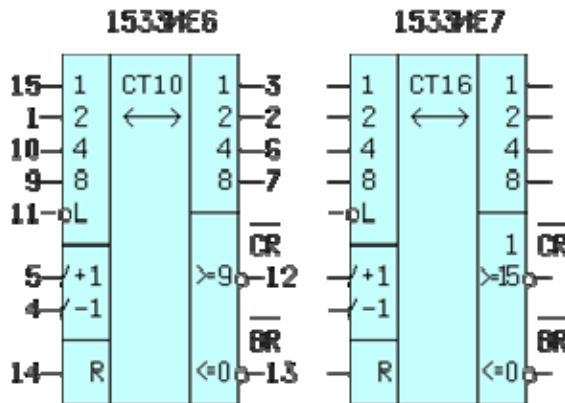


Рис.62. Условные обозначения реверсивных счетчиков

В некоторых счетчиках предпочтительней иметь один счетный вход С и вход управления направлением счета. Для этого к входам "+1" и "-1" подключают демультиплексор "1 в 2" (см. рис. 61). В этом случае, если управляющий сигнал $U/\sim D = (U_p/\sim Down) = 1$, счетные импульсы С поступают на вход суммирования, а если он равен нулю - на вход вычитания. Счетчик такого типа приведен на рисунке 63.

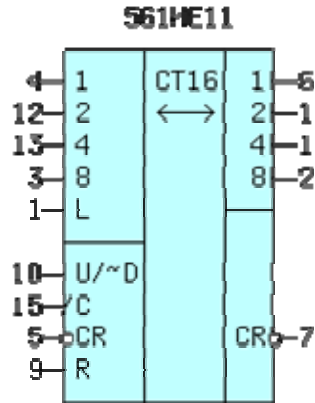


Рис. 63. Реверсивный счетчик с переключением направления счета

3.3.4 КАСКАДНОЕ ВКЛЮЧЕНИЕ СЧЕТЧИКОВ

Рассмотрим последовательное включение n - счетчиков с различными модулями счета M_i (рис.64). Возможна постановка трех задач.

- 1) необходимо определить частоту сигнала на выходе переноса $\sim CR_i$ или CR_i каждого счетчика при частоте входного сигнала $F_{вх}$.
- 2) сколько импульсов переноса появится на выходе $\sim CR_i$ или CR_i при поступлении $N_{вх}$ импульсов?
- 3) какой код будет зафиксирован на выходах QQ каждого счетчика при поступлении $N_{вх}$ импульсов?

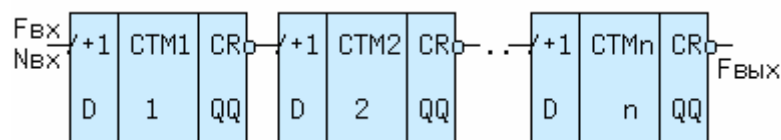


Рис.64. Каскадное включение счетчиков

Решение первой задачи основано на том, что каждый счетчик делит частоту сигнала на счетном входе в M - раз (см. временные диаграммы и вывод №3 к ним). Поэтому частота на выходе CR_1 : $F_{CR1} = F_{вх} / M_1$, на выходе CR_2 : $F_{CR2} = F_{CR1} / M_2 = F_{вх} / (M_1 * M_2)$ и т.д. до $F_{вых} = F_{вх} / (M_1 * M_2 * \dots * M_n)$.

Для решения остальных задач воспользуемся формулами счетчика.

Тогда во второй задаче на выход переноса первого счетчика пройдет $CR1 = (D1 + N_{вх}) / M1$ - импульсов, где все величины известны. Аналогично найдем, что на выход переноса второго счетчика пройдет $CR2 = (D2 + CR1) / M2$ импульсов ($CR1$ - число импульсов на одноименном выходе уже найдено, а остальные величины $D2$ и $M2$ также известны. Для последнего счетчика: $CRn = (Dn + CR(n-1)) / Mn$.

В третьей задаче код на выходе 1-го счетчика $QQ1 = (D1 + N_{вх}) \bmod M1$, код на выходе второго счетчика $QQ2 = (D2 + CR1) \bmod M2$, и так далее до $QQn = (Dn + CR(n-1)) \bmod Mn$, причем величины CRi вычислены во второй задаче.

3.3.5 СЧЕТЧИК - ТАЙМЕР

Служит для формирования временных отрезков заданной длительности. Таймер может выполняться, как на вычитающих так и на суммирующих синхронных счетчиках с синхронной загрузкой начального кода. Общим для этих схем является загрузка (перезагрузка) начального кода в момент переполнения счетчика. При этом интервал времени отсчитывается от активного фронта первого счетного импульса до заднего фронта сигнала переноса (рис.65).

Пусть требуется сформировать временной интервал или задержку длительностью $T = N * tclk$, причем $tclk$ известно, а N може принимать значения ($2 \leq N \leq M-1$). Тогда начальный код D , загружаемый в суммирующий счетчик должен равняться $D = M - N$. В вычитающий счетчик требуется записать $D = \sim(M - N) = N - 1$.

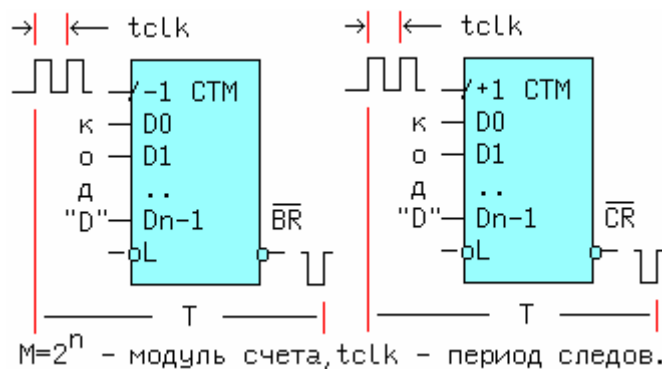


Рис.65. Формирование интервала заданной длительности

На рисунке 66 приведен пример реализации таймера, выполненного на суммирующем счетчике 1533ИЕ18 (74163). В примере $N = 10$. Когда $\sim L = \sim CR = 0$ (т.е. загрузка кода разрешена) очередной положительный фронт входного импульса опять загрузит $D=6$ и далее процесс повторится.

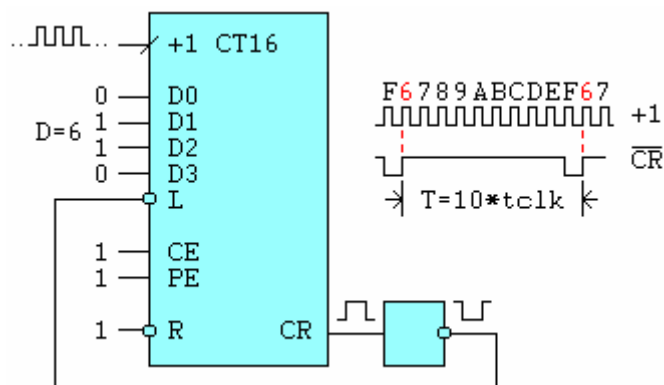


Рис.66. Схема таймера

Разумеется такой счетчик делит входную частоту $f_{clk} = 1/t_{clk}$ в N раз $F = f_{clk}/N$, где $F = 1/T$. Синхронные счетчики с синхронной загрузкой начального кода используются в таймерах, встраиваемых в микроконтроллеры (МК).

3.3.6 ПРИМЕНЕНИЕ СЧЕТЧИКОВ В ИЗМЕРИТЕЛЬНОЙ ТЕХНИКЕ

На рисунке 67 последовательно включены 6 счетчиков с модулем 10 (двоично - десятичные). Информационные выходы Q_i каждого каскада через преобразователь BCD кода подключены к семисегментному индикатору.

В целом устройство может решать три задачи: 1) считать количество импульсов, 2) измерять их частоту следования и 3) длительность.

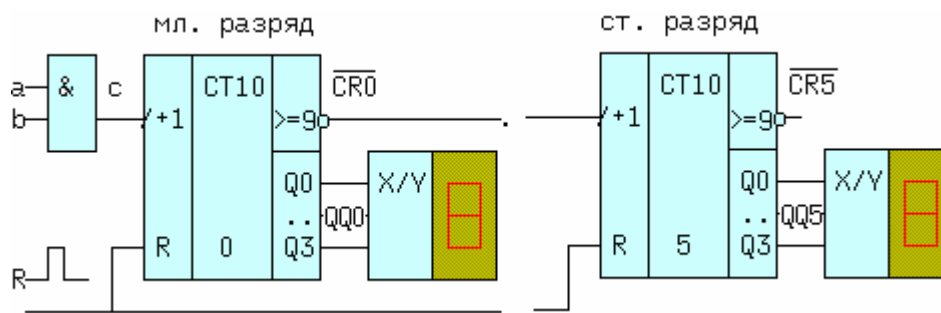


Рис.67. Измерительная схема

Перед любыми измерениями необходимо очистить счетчики, подав на вход R импульс сброса.

1) Для подсчета количества импульсов необходимо подать их на оба входа схемы И (или только на один, а на другой 1). Пусть на эти входы поступило $N = 1997$ импульсов. Тогда на выходах младшего счетчика останется код $QQ0 = 1997 \bmod 10 = 7$. На выход переноса пройдет $CR0 = 1997 / 10 = 199$ импульсов, и далее $QQ1 = (199 \bmod 10) = 9$ и $CR1 = 199 / 10 = 19$. В конце счета на выходах счетчиков будет зафиксирован BCD код = 0000 0000 0001 1001 1001 0111, а на индикаторах высветится число 001997. Внимание! У счетчиков на схемах младший разряд находится слева, а в числах - справа. В

общем виде десятичный эквивалент BCD кода на выходах будет иметь вид: $N = QQ(n-1) * 10^{(n-1)} + \dots + QQ1 * 10^1 + QQ0 * 10^0$. Из этого выражения следует, что N находится в диапазоне: $1 \leq N \leq 999999$.

2) При измерении частоты входных импульсов, их необходимо подвести к входу "a", а на вход "b" подать импульс калиброванной длительности T равной 10^i ($i = \dots -1, 0, 1, \dots$), например $T = 1$ сек, как на рисунке 68.

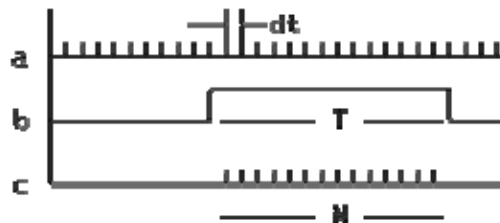


Рис.68. Измерение частоты и длительности импульсов

Частота повторения импульсов на входе "a" равна по определению: $F = 1/dt$. Из диаграмм "b" и "c" можно найти период повторения $dt = T / N$, с погрешностью dt / T . Тогда $F = N / T$ и при $T = 1$ с частота $F = N$. Диапазон измерения находится в пределах: $(1 \dots (10^6) - 1)$ Гц. Для измерения длительности импульса на вход "a" необходимо подать сигнал с частотой $F = 10^i$ ($i = 2, 3, \dots$), например $F = 10^6$ [Гц]. Длительность импульса $T = N * dt = N / F$ и диапазон ее измерения равен $(1 \dots 999999)$ мксек.

3.3.7 ЗАДАЧИ И УПРАЖНЕНИЯ

1) На входы реверсивного счетчика (рис.68-1) подаются приведенные внизу последовательности импульсов. Какая шестнадцатиричная цифра будет высвечена на семисегментном индикаторе в момент времени отмеченный знаком + ?

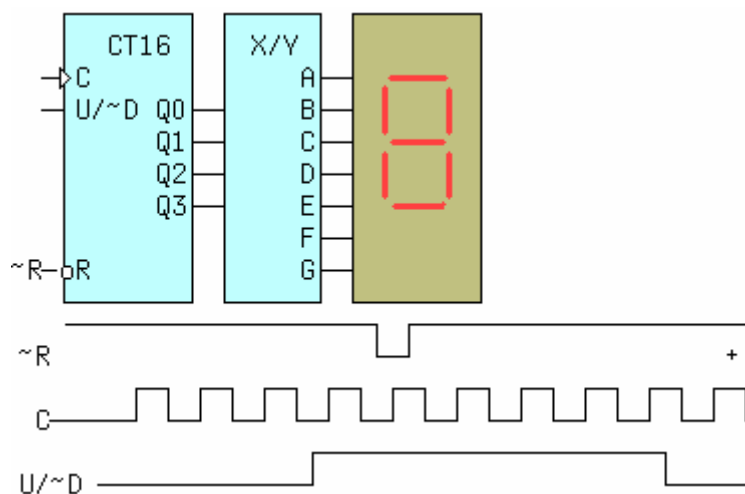


Рис.68-1. Задача 1

Пояснения : 1) Так как вход (R)eset имеет наивысший приоритет, то при $\sim R = 0$ счетчик обнуляется, независимо от состояния остальных входов (и все, что с ним происходило до этого не имеет значения). 2) При $U/\sim D = 0$ (Down - вниз) счетчик работает на вычитание, а при $U/\sim D = 1$ (Up - вверх) - на сложение. 3) Счетчик имеет прямой динамический вход (срабатывает по положительному перепаду, об этом свидетельствует стрелка внутрь у входа С), поэтому необходимо подсчитать отдельно количество положительных перепадов для режимов суммирования (4) и вычитания (1) и получить ответ : цифра 3.

2) Какой десятичный код будет зафиксирован на выходах Q3Q2Q1Q0 счетчика (рис.68-2) при поступлении на его вход 4 импульсов и начальных значениях на выходах Q3Q2Q1Q0 = 1100?

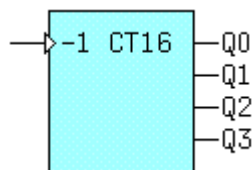


Рис.68-2. Задача 2

Пояснения : 1) Находим десятичный (DEC) эквивалент двоичного кода 1100(BIN) = 12(DEC) . 2) Обращаем внимание, что счетчик вычитающий (вход С "-1"). 3) Очевидно, что ответом будет код = 12 - 4 = 8.

3) Два счетчика включены последовательно (рис.68-3). Определить десятичный эквивалент выходного кода Q2Q1Q0 второго счетчика. Число импульсов, поступивших на вход С равно 94.

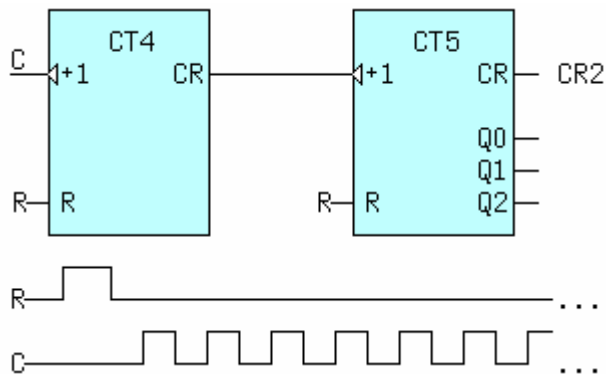


Рис.68-3. Задача 3

Пояснения : 1) Обращаем внимание, что оба счетчика - суммирующие и перед поступлением импульсов обнуляются импульсом R=1. 2) Число импульсов прошедших на выход переноса CR первого счетчика равно $CR = (0 + \text{SUM}(C)) \setminus M = 94 \setminus 4 = 23$. 3) Выходной код второго счетчика равен остатку от деления $Q = (0 + 23) \bmod 5 = 3$.

4) На входы реверсивного счетчика импульсов подаются приведенные внизу последовательности импульсов. Какая шестнадцатиричная цифра (0..9AbCdEF) будет высвечена на семисегментном индикаторе в момент времени отмеченный знаком + ? На рис.68-4 приведены содержимое памяти и рисунок схемы.

АДРЕС: 0 1 2 3 4 5 6 7 8 ...
 КОД : 07 39 4f 6f 7d 77 7c 06 07 ...

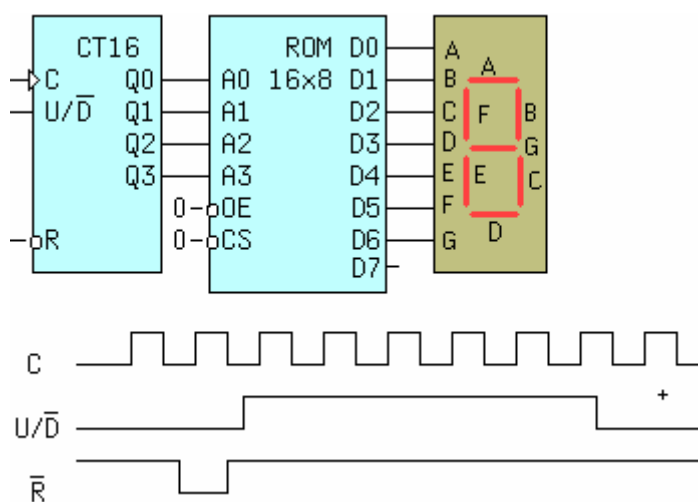


Рис.68-4. Задача 4

Пояснения : 1) Реверсивный счетчик с прямым динамическим входом С (стрелка --> направлена внутрь). 2) Импульсом $\sim R=0$ счетчик обнуляется 3) Подсчитайте число активных положительных перепадов импульсов С для режима сложения ($U/\sim D=1$) и вычитания ($U/\sim D=0$), 6 и 1 соответственно. 4) Найдите результирующий код на адресных входах микросхемы памяти: $6 - 1 = 5$. 5) Найдите в памяти HEX код, хранящийся по этому адресу: равен 77(HEX). 6) Распишите этот код в двоичном виде, $77 = 01110111$ (BIN) т.е. светиться будет цифра А(сегменты GFE и CBA). сегменты = HGFEDCBA. Сегмент "H" - десятичная точка не используется.

Ответ: цифра А.

4. ЦАП И АЦП

Цифроаналоговые преобразователи (ЦАП) численные данные преобразуют в аналоговый сигнал, чаще в напряжение или в ток и служат для связи цифровых и выходных устройств. Аналогоцифровые преобразователи (АЦП) осуществляют обратное преобразование и являются промежуточными звеньями между датчиками сигналов и цифровыми схемами их обработки.

4.1 ЦАП С МАТРИЦЕЙ РЕЗИСТОРОВ R-2R

В ЦАП используется метод суммирования токов, пропорциональных весовым коэффициентам двоичных разрядов. К входу матрицы (рис. 69) подключается прецизионный источник опорного напряжения U_0 с током потребления $I_{вх} = I_0 * 2^n$, где n - число звеньев матрицы (разрядов ЦАП).

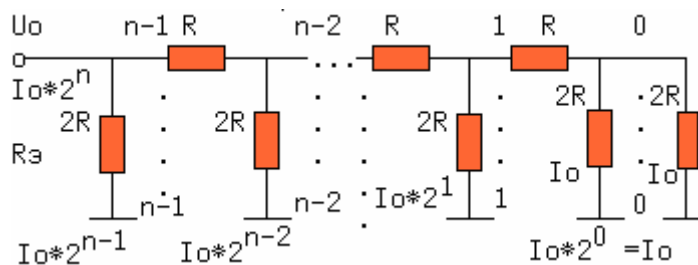


Рис. 69. Матрица резисторов R-2R

Эквивалентное сопротивление цепи справа от "сечения 0" равно $R_{э0} = 2R$. Сопротивление цепи справа от "сечения 1" равно $R_{э1} = R + 2R || R_{э0} = 2R$. Значком $||$ обозначено параллельное соединение двух сопротивлений. Рассуждая таким образом найдем, что $R_{э_{n-1}} = R + 2R || R_{э_{n-2}} = 2R$ и полное сопротивление цепи со стороны входа, на который подается напряжение U_0 , равно $R_{э} = 2R || R_{э_{n-1}} = R$, т.е. равно номиналу R независимо от числа звеньев матрицы. Теперь можно найти величину тока

$$I_0 * 2^n = U_0 / R \quad (I)$$

Нетрудно подсчитать, что ток в узле $n-1$ делится пополам, одна половина ответвляется в сопротивление $2R$, а другая в сопротивление $R_{э_{n-1}}$ тоже равное $2R$. Половина попадающая в узел $n-2$ также делится пополам и т.д. Следовательно величина тока в каждой "ветке" равна $I_0 * 2^i$, т.е. пропорциональна весовому коэффициенту 2^i i -го разряда. Суммирование токов осуществляется с помощью операционного усилителя (ОУ), включенного по схеме сумматора (см. Рис. 70). Часть схемы, слева от ОУ, выпускается промышленно в виде микросхем, например К572ПА1,2 и 1108ПА1.

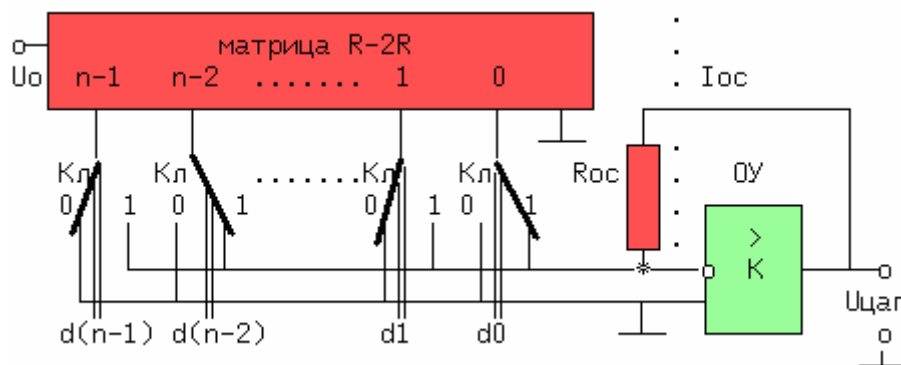


Рис.70. ЦАП с матрицей R-2R

Электронные ключи Кл управляются входными сигналами d_i цифрового кода. Левое положение ключа на рисунке эквивалентно нулевому значению i -го разряда, а правое - единичному. Относительно ОУ необходимо сделать известные допущения:

- 1) Коэффициент усиления ОУ без ОС стремится к бесконечности,
- 2) $R_{вх}$ обоих входов тоже бесконечно велико.

Отсюда следует, что во-первых разность потенциалов $dU_{вх}$ на входах ОУ близка к нулю, т.к. $dU_{вх} = U_{цап} / K$ ($K = \text{бесконечности}$) и следовательно потенциал инверсного входа ОУ тоже близок к нулю и, во-вторых, током каждого входа ОУ можно пренебречь. Сумма токов втекающих в узел (*) и вытекающих из него равна нулю, поэтому можно записать: $I_{ос} + \sum (I_o * 2^i * d_i) = 0$ или $\sum (I_o * 2^i * d_i) = -I_{ос}$. Подставляя в последнее выражение значение I_o из формулы (I) получим: $-I_{ос} = (U_o / (R * 2^n)) * (\sum (d_i * 2^i))$. В свою очередь, $U_{цап} = I_{ос} * R_{ос}$. Подставляя сюда значение $I_{ос}$ и учитывая, что $R_{ос} = R$ получим окончательную формулу напряжения на выходе ЦАП $U_{цап}$:

$$U_{цап} = -\frac{U_o}{2^n} \sum_{i=0}^{n-1} d_i * 2^i = -\frac{U_o * D}{2^n}$$

где $D = \sum (2^i * d_i)$ - десятичный эквивалент цифрового кода на входах ЦАП.

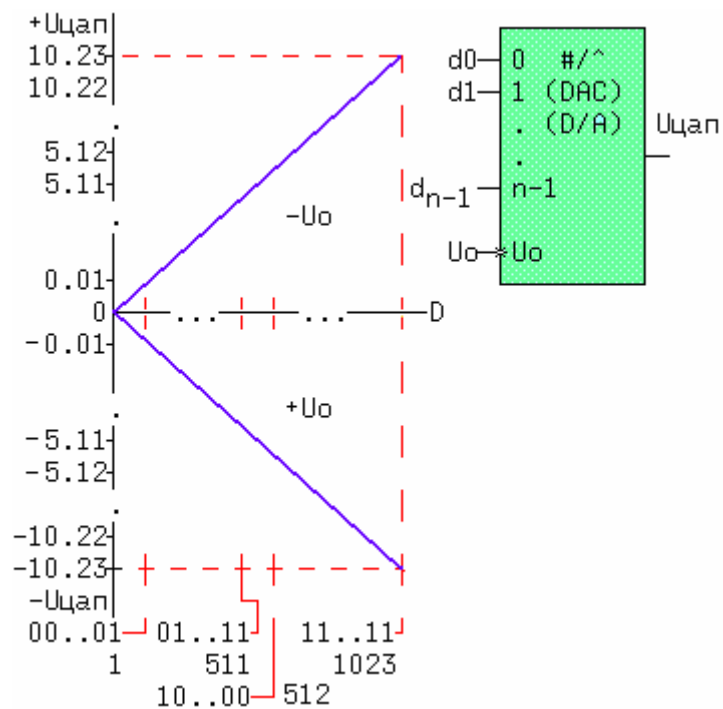


Рис.71. Униполярный ЦАП и его передаточная характеристика

Пусть число разрядов $n = 10$ и $U_0 = -10.24\text{В}$, тогда $U_{\text{цап}} = (10.24/1024)*D$ и находится в пределах (0, 0.01, 0.02 ... 10.22, 10.23)В. Передаточная характеристика показана на рисунке 71. Такой ЦАП называется: 1) униполярным, т.к. $U_{\text{цап}}$ в зависимости от полярности U_0 , либо отрицательно, либо положительно, 2) двухквадрантным, потому, что передаточная характеристика располагается в двух квадрантах, 3) умножающим, т.к. $U_{\text{цап}}$ пропорционально U_0*D .

Верхний квадрант относится к $-U_0$, а нижний к $+U_0$. Вообще-то знак в формуле (2) может быть как (-), так и плюс (+), в зависимости от способа включения операционного усилителя, но в конспекте везде (-).

На рисунке 71-1 показано применение униполярного ЦАП для формирования пилообразного напряжения.

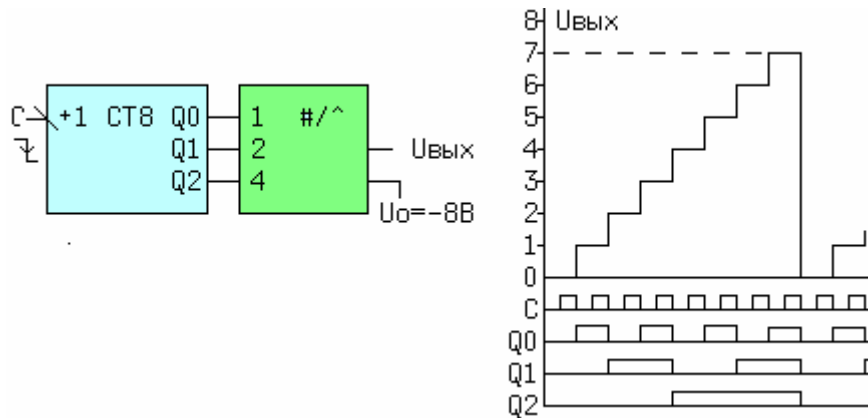


Рис.71-1. Пример использования ЦАП

4.2 БИПОЛЯРНЫЙ ЦАП

Если необходимо менять знак выходного напряжения в зависимости от значений входного кода, а знак U_0 изменить невозможно, применяется биполярный ЦАП, в котором характеристика смещается на половину своего диапазона. К выходу униполярного ЦАП (см. рис. 72) подключается еще один сумматор токов на ОУ. Применяя полученные ранее выражения для $U_{\text{цап}}$ и учитывая, что $I_{\text{см}} + I_{\text{цап}} + I_{\text{ос}} = 0$, а также, что $I_{\text{см}} = U_0/2R$, $I_{\text{цап}} = U_{\text{цап}}/R$ и $I_{\text{ос}} = U_{\text{цап}}/R$, нетрудно получить: $U_{\text{цап}} = I_{\text{ос}}*R_{\text{ос}} = U_0*D/2^n - U_0*D$. Обе характеристики пересекаются в точке $U_{\text{цап}} = 0$ и $D=512$.

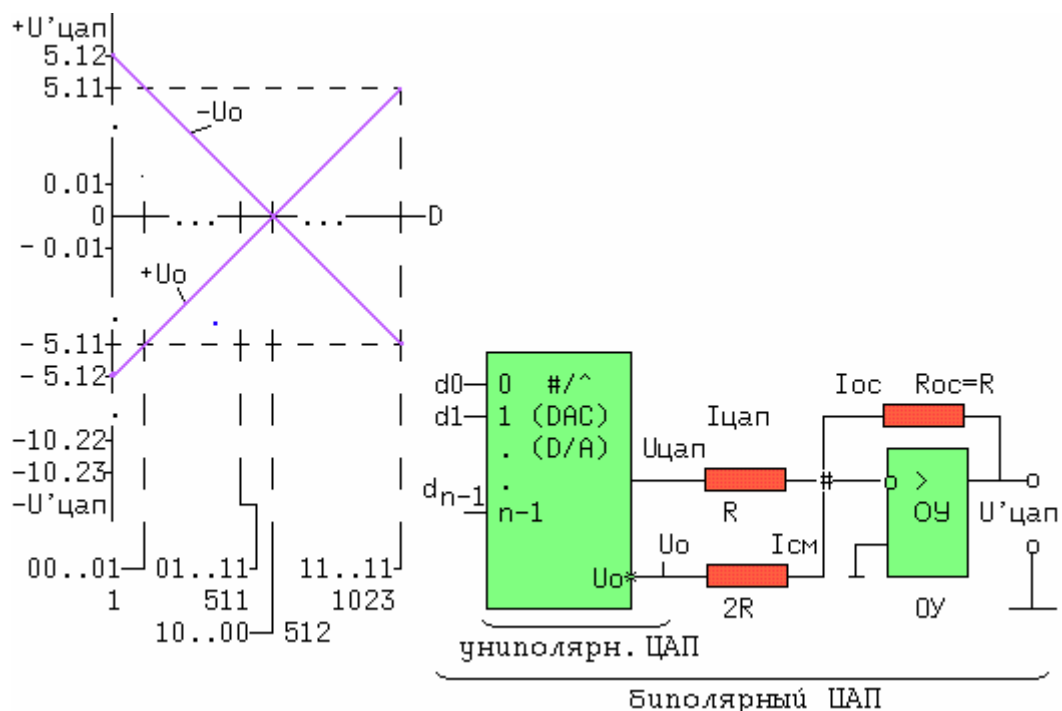


Рис.72. Биполярный ЦАП и его характеристика

4.3 ЧЕТЫРЕХКВАДРАНТНЫЙ ЦАП

Недостаток биполярного ЦАП - ненулевое значение входного кода при нулевом выходном напряжении. Преодолеть этот недостаток можно, если договориться о том, что одни двоичные числа считать положительными, а другие - отрицательными. Одним из двоичных кодов для чисел со знаком является "дополнительный до двух" код, в котором число с противоположным знаком находится инверсией исходного числа и увеличением результата на единицу.

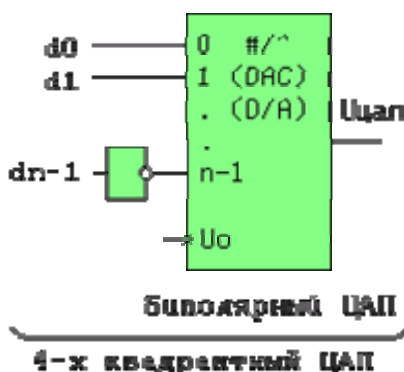


Рис.73. Четырехквadrантный ЦАП

Причем старший бит обозначает знак числа. Если он равен единице, число отрицательное, если равен нулю - положительное. Минимальное отрицательное 10-ти разрядное число $10..00(\text{BIN}) = -512(\text{DEC})$. Максимальное положительное $01..11(\text{BIN}) = 511$. Максимальное отрицательное 10-ти разрядное число $11..11(\text{BIN}) = -1(\text{DEC})$. Минимальное

положительное $00..01(\text{BIN}) = 1(\text{DEC})$. Проинвертировав старший бит, получим передаточную характеристику расположенную в зависимости от знака U_0 в четырех квадрантах, с нулем посередине. Такой ЦАП называется четырехквadrантным (рис. 73).

4.4 АЦП ПОРАЗРЯДНОГО УРАВНОВЕШИВАНИЯ (ПОСЛЕДОВАТЕЛЬНЫХ ПРИБЛИЖЕНИЙ)

Метод аналогичен взвешиванию груза. В старший разряд регистра записывается 1 (на весы кладется самая большая гиря). Если входное напряжение больше или равно напряжению соответствующего этой единице (груз тяжелее гири), единица в старшем разряде сохраняется (гиря остается на весах), если наоборот - единица заменяется нулем (гиря снимается). Затем единица записывается в предпоследний разряд (ставится следующая гиря) и процесс уравнивания повторяется. Количество приближений равно числу разрядов преобразователя.

Пусть число разрядов АЦП $n = 10$ (см. рисунок 74). АЦП начинает цикл преобразования с приходом положительного импульса на вход R. Высоким уровнем этого сигнала регистр RG1 обнуляется, а переход сигнала R к нулю запускает внутренний генератор и схему управления.

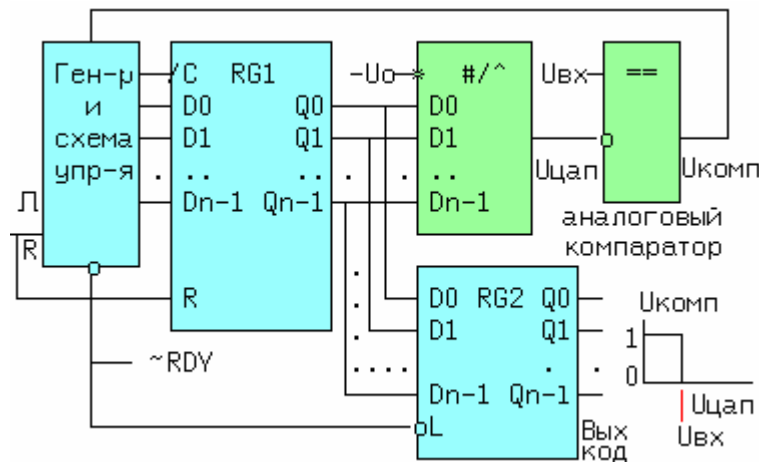


Рис.74. АЦП последовательных приближений

В момент t_0 в старший разряд регистра RG1 записывается лог."1" (гиря с наибольшим весом). На выходах регистра RG1 появляется код $10..00(\text{BIN}) = 2^{(n-1)} = 512$. ЦАП генерирует напряжение $U_{\text{цап}(n-1)} = U_0 * 2^{(n-1)} / 2^n = U_0 / 2$. Аналоговый компаратор сравнивает напряжения, и если $U_{\text{вх}} \geq U_{\text{цап}(n-1)}$, то единица в старшем разряде сохраняется, если $U_{\text{вх}} < U_{\text{цап}(n-1)}$, в старший разряд записывается ноль. В этот момент код на выходах RG1 равен $Q_{n-1}, 0, 0, \dots, 0$.

В момент t_1 единица записывается в предпоследний разряд (n-2) и на выходах регистра RG1 появится код: $Q_{n-1}, 1, 0, 0, \dots, 0$ равный сумме двух кодов:

$(Q_{n-1}) * 2^{(n-1)}$ - значения старшего бита, полученного в предыдущем такте, и $2^{(n-2)} = 0100..0(\text{BIN}) = 256$, т.е. цифрового кода с уменьшенным в два раза "весом". Выходное напряжение $U_{\text{цпн}}(n-2) = U_{\text{цпн}}(n-1) * Q(n-1) + U_0 * 2^{(n-2)} / 2^n = (U_0 / 2) * Q(n-1) + U_0 / 4$ опять сравнивается с входным. Если $U_{\text{вх}} \geq U_{\text{цпн}}(n-2)$, то единица в предпоследнем разряде сохраняется, если $U_{\text{вх}} < U_{\text{цпн}}(n-2)$, в $(n-2)$ -ой разряд записывается ноль. В этот момент код на выходах RG1 равен $Q_{n-1}, Q_{n-2}, 0, ..0$.

Описанный процесс производится в общей сложности n -раз. В конце цикла преобразования схема управления вырабатывает импульс готовности $\sim \text{RDY}$, который записывает полученный код в выходной регистр RG2. Этот же импульс выводится наружу, для оповещения. Время преобразования $T_{\text{преобр}} = t_{\text{clk}} * n$, где $t_{\text{clk}} = t_{i+1} - t_i$ период повторения тактовых импульсов внутреннего генератора. Временные диаграммы и условное обозначение АЦП приведены рисунке 75.

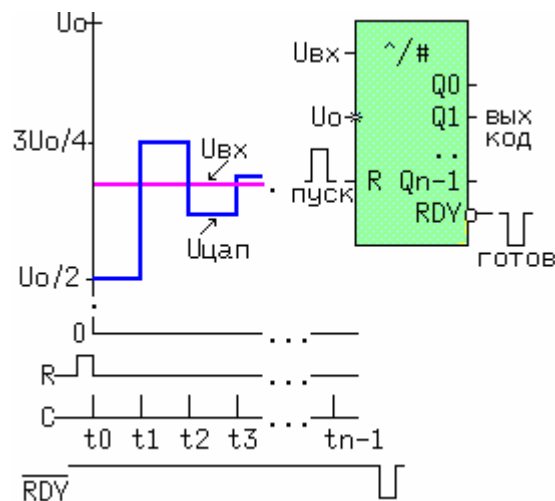


Рис.75. Временные диаграммы и условное обозначение АЦП

По этому принципу действия выпускаются преобразователи типов 572ПВ1 и 1113ПВ1(А..В). Зависимость прямого двоичного кода D от входного напряжения получается путем решения формулы ЦАП относительно двоичного кода: $D = |(U_{\text{вх}} * 2^n) / U_0|$, где D десятичный эквивалент кода на входе АЦП.

4.5 АЦП ПАРАЛЛЕЛЬНОГО ТИПА

В таком АЦП весь диапазон входного напряжения разбивается на 2^n интервалов (рис.76). Каждому интервалу соответствует опорное напряжение $U_0(i)$, снимаемое с делителя напряжения, и свой аналоговый компаратор, сравнивающий $U_{\text{вх}}$ с $U_0(i)$.

Для любого входного напряжения в диапазоне $(0 .. |U_0|)В$ найдется такой i -ый компаратор, входное напряжение на котором будет больше или равно опорному $U_0(i)$.

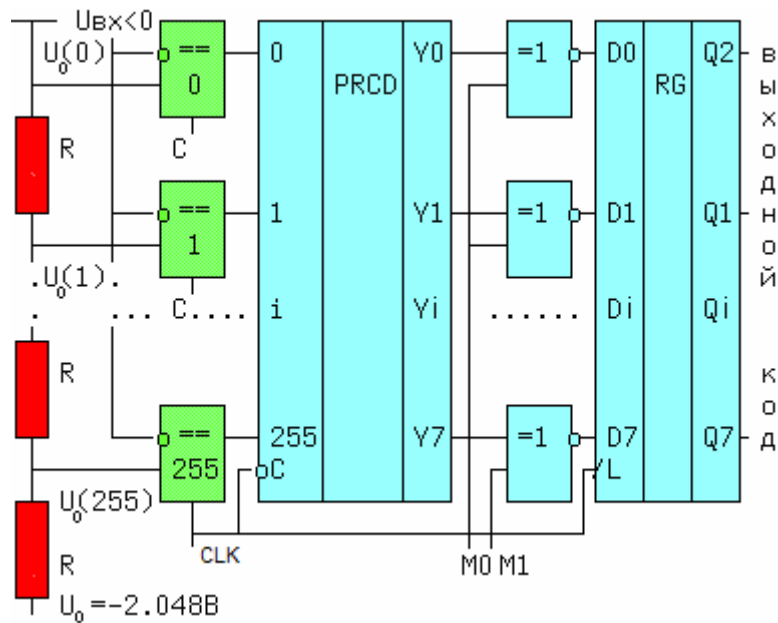


Рис.76. АЦП параллельного типа

В этом случае на выходе этого компаратора напряжений и на выходах всех компараторов с номерами меньшими i появится "1", а на выходах остальных "0". Приоритетный шифратор сформирует двоичный код, равный наивысшему номеру входа на котором еще присутствует единица. Полученный код через управляемые инверторы/повторители, выполненные на элементах равнозначности, защелкивается в выходном регистре. Учитывая логику работы шифратора, заключающуюся в том, что если на его входах с 1-го по $(n-1)$ -ый одни нули, то единица обязательно должна быть на нулевом входе, можно сэкономить на нулевом компараторе. В этом случае нулевой вход шифратора постоянно подключен к "1", и если входное напряжение находится в пределах $0 \leq |U_{вх}| < |U_0(1)|$, то нулевой код на выходе генерируется автоматически. Элементы ИСКЛЮЧАЮЩЕЕ-ИЛИ-НЕ (рис.77) при значении $M_i = 0$ инвертируют выходной сигнал шифратора, а при $M_i = 1$ повторяют его (еще одно полезное применение ИСКЛЮЧАЮЩЕГО-ИЛИ(-НЕ)). Зависимость типа выходного кода от режима дана ниже:

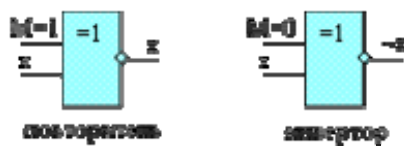


Рис.77. Управляемый инвертор/повторитель

M1	M0	Выходной код
1	1	прямой
0	0	обратный
0	1	дополнительный прямой
1	0	дополнительный обратный.

Быстродействие достигается, как за счет параллельного принципа работы, так и за счет конвейерного передвижения информации внутри АЦП (см. рисунок 78).

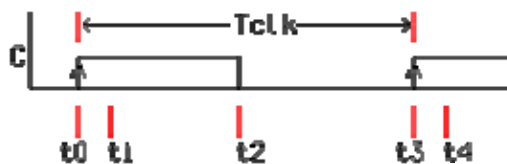


Рис.78. Временная диаграмма

В момент t_0 положительным фронтом сигнала C происходит защелкивание кода от предыдущего цикла преобразования. В момент t_1 компараторы открываются и начинается сравнение входного напряжения с опорным в текущем цикле. Когда тактовый сигнал C станет равным нулю (момент t_2), шифратор отпирается по входу C и начинает преобразование текущего значения $U_{вх}$ в код. Следующий цикл преобразования начинается в момент t_3 , когда производится запоминание в регистре текущего кода и его появление на выходах в момент t_4 , одновременно с началом сравнения на входах компараторов следующего значения напряжения. Время преобразования $T_{пр} = T_{clk}$, т.е. примерно на порядок меньше, чем у АЦП последовательных приближений. По такой схеме выполняются преобразователи типа 1107ПВ1,...5.

5. ЗАПОМИНАЮЩИЕ УСТРОЙСТВА (ПАМЯТЬ)

Память может быть внутренней и внешней. Внешней называют память на магнитных, оптических дисках, лентах и т.п. Внутренняя память выполняется, чаще всего, на микросхемах. Внутренняя или основная память может быть двух типов: оперативное запоминающее устройство (**ОЗУ**) или ЗУ с произвольной выборкой (**ЗУПВ**) и постоянное ЗУ (**ПЗУ**). ОЗУ, кроме того, обозначается - (**RAM**, Random Access Memory), а ПЗУ - (**ROM**, Read Only Memory). В ОЗУ коды в соответствии с решаемыми задачами постоянно изменяются и полностью пропадают при выключении питания. В ПЗУ хранятся управляющие работой ЭВМ стандартные программы, константы, таблицы символов и другая информация, которая сохраняется и при выключении компьютера, т.е. ПЗУ является энергонезависимой памятью. ОЗУ подразделяются на статическую память (**SRAM**), динамическую (**DRAM**), регистровую (**RG**). ПЗУ могут быть: масочными - запрограммированными на заводе изготовителе (**ROM**), однократно-программируемыми пользователем **ППЗУ (PROM или OTP)**, многократно-программируемыми (репрограммируемыми) пользователем **РПЗУ** с ультрафиолетовым стиранием (**EPROM или СПЗУ**) или с электрическим стиранием (**EEPROM (E²PROM) или ЭСПЗУ, Flash**). Широкое распространение имеют также программируемые логические интегральные

схемы - ПЛИС (CPLD, FPGA и т.д.) с большим выбором логических элементов и устройств на одном кристалле.

Сочетает в себе свойства ОЗУ и ПЗУ энергонезависимое ОЗУ (NonVolatile SRAM или NVSRAM). Активно пробивают дорогу еще два типа энергонезависимой памяти, имеющие особенности и ОЗУ и ПЗУ. Это ферроэлектрическая память - FRAM и магниторезистивная память - MRAM. В стадии разработки находится энергонезависимая память на аморфных полупроводниках - OUM (Ovonic Unified Memory). Число циклов стирания/записи у нее больше 10 трлн, что на несколько порядков больше, чем у флэш-памяти.

В зависимости от типа ЗУ элементом памяти (ЭП) может быть: триггер, миниатюрный конденсатор, КМОП (CMOS) транзистор с дополнительным "плавающим затвором", плавкая перемычка (или ее отсутствие). Упорядоченный набор ЭП образует ячейку памяти (ЯП). Количество элементов памяти в ячейке (число разрядов) обычно кратно 2^n (1,4,8,16, 32,64..), причем величины свыше 8-ми достигаются, обычно, группировкой микросхем с меньшим количеством ЭП. Количество ЭП в ЯП иногда называется длиной слова. Основными характеристиками микросхем памяти являются: информационная емкость, быстродействие и энергопотребление.

Емкость ЗУ (рис.79) обычно выражается в единицах кратных числу $2^{10} = 1024 = 1К$ (не путать с $1к = 1000$). Для длины слова равной биту (одному двоичному разряду) или байту (набору из восьми бит) эта единица называется килобит или килобайт и обозначается Кб или КВ.

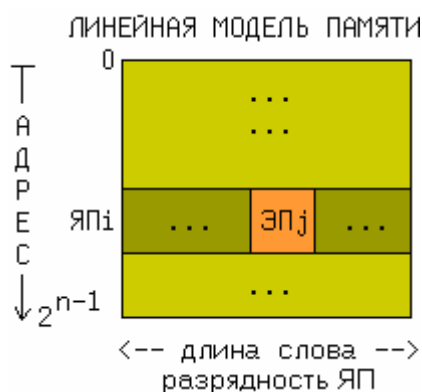


Рис.79. Емкость памяти

Каждой из двух в степени "n" ячеек памяти однозначно соответствует "n"- разрядное двоичное число, называемое адресом ЯП. Например, адресом 511-ой ячейки будет число $1\ 1111\ 1111(\text{BIN}) = 511(\text{DEC}) = 1\text{FF}(\text{HEX})$. В программах адреса употребляются в 16-ном формате.

Емкость ЗУ часто записывается в виде произведения двух чисел $2^n * m$, где 2^n - число ячеек памяти, а m - длина слова ячейки, например $8К * 8$ (м/с 537РУ17), т.е. 8192 ячейки размером в один байт. В некоторых справочниках для этой же микросхемы приводится обозначение емкости одной цифрой

64Кбит, что никак не отражает внутреннюю организацию этой микросхемы, такую же емкость могут иметь м/с с организацией 16К * 4, 64К * 1 и т.д.

5.1 СТАТИЧЕСКОЕ ОЗУ (SRAM)

В качестве элемента памяти используется простейший D-триггер защелка. В микросхеме 537PY10 каждая ЯП состоит из восьми триггеров и располагаются ячейки на кристалле в виде прямоугольной матрицы (рис.80).

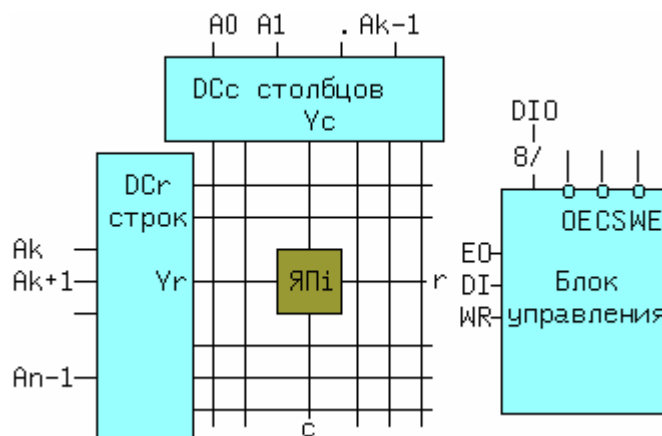


Рис.80. Типовая блок-схема ОЗУ

На рисунке 80 приведены обозначения: n-адресных входов (A₀ .. A_{n-1}), DIO - двунаправленная восьмиразрядная шина данных, вход разрешения выходов - \sim OE, вход выбора микросхемы - \sim CS и вход разрешения записи - \sim WE, который часто обозначают по другому - \sim WR/RD, подчеркивая этим, что при низком уровне сигнала на этом входе производится запись байта, а при высоком уровне может выполняться чтение при \sim OE = 0. EO, DI, WR - внутренние сигналы вырабатываемые блоком управления чтением/записью/хранением. Доступ к произвольной ЯП_i производится с помощью прямоугольного дешифратора, состоящего из двух обычных дешифраторов, причем k-адресных линий заводится на дешифратор столбцов (DC_c), а оставшиеся n-k линий подключены к дешифратору строк (DC_r). Количество строк и столбцов будет соответственно равно 2^{n-k} и 2^k , т.е. общее количество, обслуживаемых ЯП, равно $2^k * 2^{n-k} = 2^n$.

На рисунке 81 показан фрагмент внутренней структуры микросхемы, по которому можно проследить основные режимы ее работы. Здесь же дано условное обозначение микросхемы.

Элементы и схемы с открытым коллектором и третьим состоянием обозначены, как обычно \square и \square - соответственно, \square - повторитель с высокой нагрузочной способностью.

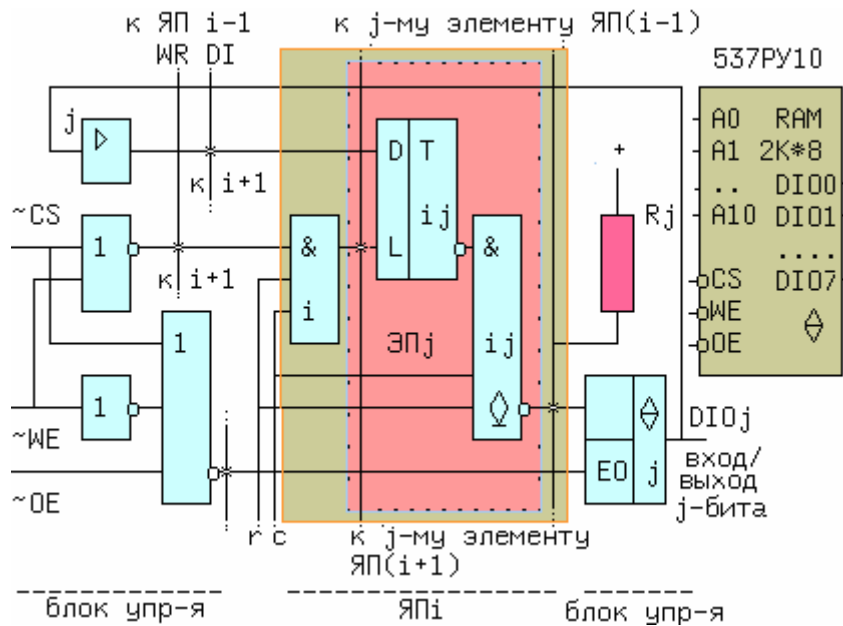


Рис.81. Типовая схема управления записью/чтением/хранением данных в ОЗУ

Точками выделен один (j -ый) из восьми элементов i -ой ячейки памяти. Элемент И с номером $i = (r * 2^k + c)$ является одним из 2^n выходных узлов прямоугольного дешифратора, где r и c - номера строк и столбцов матрицы. Инверсный вход (C)hip (S)elect - $\sim CS$, во всех микросхемах, где он встречается, служит для приведения схемы в рабочее состояние низким уровнем сигнала на этом входе.

Если $\sim CS = 1$ (пассивный уровень), микросхема - не выбрана и операции с ней производить невозможно. Из рис. видно, что в этом случае на L-входе D-триггера - ноль, запись невозможна и триггер хранит ранее записанный бит. Прочитать выходной код - Q тоже нельзя, т.к. на прямом входе EO разрешения выхода запрещающий нулевой сигнал и вход/выход DIO_i находится в третьем состоянии.

С поступлением $\sim CS = 0$, схемы ИЛИ-НЕ разблокируются и дальше все зависит от значений сигналов на входах $\sim WE$ и $\sim OE$.

В режиме записи на вход $\sim WE$ подается импульсный сигнал (строб записи) $\sim WE = 0$. Поэтому независимо от значения сигнала на входе $\sim OE$, внутренний сигнал EO, тоже равен 0, и чтение данных во время записи невозможно. На верхнем входе элемента I_i - единица и, если на адресных входах код $A_{n-1}, A_{n-2}, \dots, A_1, A_0 (BIN) = i (DEC)$, то сигналы на линиях Y_r и Y_c тоже равны 1 и триггер ij прозрачен для записи входной информации DIO_j .

В режиме чтения на входе $\sim WE$ пассивный уровень $\sim WE = 1$, а на вход $\sim OE$ подается импульсный сигнал (строб чтения) $\sim OE = 0$ и при $Y_r = Y_c = 1$, выходной сигнал $\sim Q$ после инверсии элементом I_{ij} с открытым коллектором проходит на выход DIO_i .

Следует обратить внимание на то, что выходы всех 2^n j -ых элементов памяти должны подключаться к общему j -му выводу микросхемы - DIO_j . Такое объединение выходов возможно с помощью схемного либо монтажного И(ИЛИ). Монтажное И(ИЛИ) не требует дополнительных схем и может выполняться на элементах с открытым коллектором или с третьим состоянием. Внутри рассматриваемой схемы j -е выходы ЭП объединены на общем резисторе R_j , служащем нагрузкой элементов И-НЕ $_{ij}$ с открытым коллектором.

Для увеличения информационной емкости отдельные микросхемы группируются в банки и их одноименные выходы должны объединяться. По этой причине выходы всех микросхем памяти также выполняются с открытым коллектором либо с третьим состоянием.

В микропроцессорных системах и ЭВМ быстродействующее статическое ОЗУ широко используется для оперативного хранения информации в том числе в Cash-памяти. Кстати, самым быстрым ОЗУ являются внутренние регистры (регистровая память) МП или МК. С какой частотой работают МП? С такой же частотой 2..3 ГГц производится обращение к регистрам, например по команде "mov eax,ecx".

5.2 ДИНАМИЧЕСКОЕ ОЗУ (DRAM)

В качестве элемента памяти используется микроконденсатор в интегральном исполнении, размеры которого значительно меньше D-триггера статической памяти. По этой причине, при одинаковых размерах кристалла, информационная емкость DRAM выше, чем у SRAM. Количество адресных входов и соответственно габариты должны увеличиться. Чтобы не допустить этого, адресные линии внутри микросхемы разбиваются на две группы, например старшая и младшая половина. Две одноименные k -линии каждой группы подключаются к двум выходам внутреннего k -го демультиплексора "1 в 2", а его вход соединяется с k -ым адресным входом микросхемы. Количество адресных входов, при этом уменьшается в два раза, но зато передача адреса в микросхему должна производиться, во-первых в два приема, что несколько уменьшает быстродействие, и во-вторых потребуется дополнительный внешний мультиплексор адреса. В процессе хранения бита конденсатор разряжается. Чтобы этого не допустить заряд необходимо поддерживать (регенерировать).

Суммируя, можно перечислить чем отличается динамическое ОЗУ от статического: 1)мультиплексированием адресных входов, 2)необходимостью регенерации хранимой информации, 3)повышенной емкостью (до нескольких сотен Мбит в одном корпусе), 4)более сложной схемой управления. На рисунке внизу приведено условное обозначение м/с 565РУ7 емкостью

256К*1 (2^{18} К) и способ подключения 18-ти линий адреса к девяти адресным входам с помощью 9-ти мультиплексоров "2 в 1", например трех счетверенных селекторов-мультиплексоров типа 1533КП16.

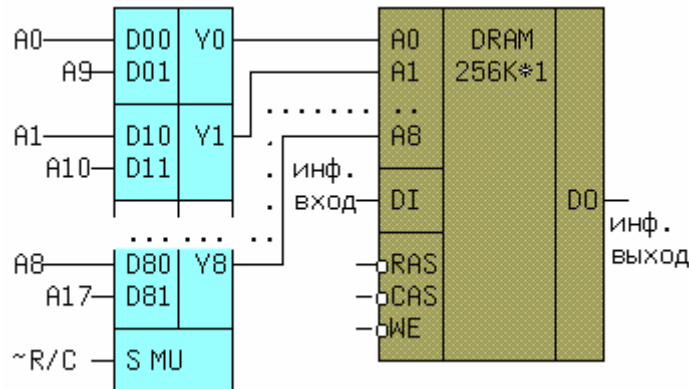


Рис.82. Типовое условное обозначение DRAM

Элементы памяти расположены на кристалле в виде матрицы $512 * 512 = 2^9 * 2^9$, управляемой двумя линейными дешифраторами строк и столбцов, каждый с 9-ю адресными входами. Если сигнал строка/столбец $\sim R/C$ на входе выбора S мультиплексора, равен нулю, то $A(0..8) = Y(0..8)$ и в микросхеме передается адрес строки. Этот адрес фиксируется отрицательным фронтом строка адреса строк $\sim RAS$. При $\sim R/C = 1$ на выходы мультиплексора передается адрес столбцов $A(9..17)$, который защелкивается отрицательным перепадом строка адреса столбцов $\sim CAS$. Вход $\sim WE$ управляет записью/чтением. Оперативная память персональных ЭВМ - (EDO, SDRAM, DDR, DDR2 ...) является динамической памятью. Время обращения к ней меньше 10нс, а емкость достигает 1024Мб в одном корпусе.

5.3 РЕПРОГРАММИРУЕМОЕ ПЗУ

Микросхемы РПЗУ допускают многократное, более 100000 циклов перепрограммирования (стирания и записи) пользователем. Это свойство обеспечивается применением ЭП на КМОП (CMOS) транзисторах с дополнительным "плавающим затвором". Толщина изоляции "плавающего затвора" порядка 200 ангстрем. Информация считается стертой, если на выходах всех ЭП высокий уровень сигнала. В режиме программирования, на выбранный по адресной шине ЭП, куда необходимо записать ноль, подается импульс. Этот импульс наводит заряд на "плавающем затворе". Чтение данных производится с помощью управляющего затвора. Стирание осуществляется УФ-излучением (EPROM), либо электрически (EEPROM(E^2 PROM), Flash). При этом все ячейки переводятся в состояние "1". Одной из м/с этого типа является EPROM 573PФ2 с организацией ($2K * 8$) и тристабильными выходами. Скорость записи в (E)EPROM меньше, чем во флэш-память. Перед записью должна быть стерта предыдущая информация.

Во флэш-памяти толщина изоляции "плавающего затвора" меньше чем у EEPROM, поэтому при перепрограммировании используется туннельный эффект. Недостатком флэш-памяти является относительно невысокая скорость произвольного доступа вследствие поблочной организации данных, однако этот недостаток компенсируется высокой информационной емкостью и высокой скоростью последовательного доступа при чтении.

Записанная в РПЗУ информация сохраняется в течение 20-ти и более лет.

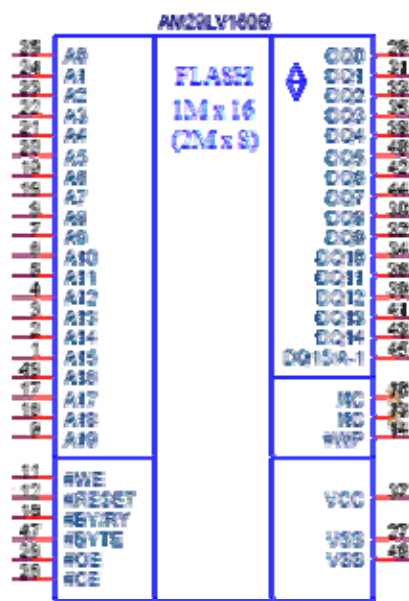


Рис.83. Типовое обозначение микросхемы флэш памяти

На рис. 83 показана типовая микросхема флэш-памяти емкостью $2^{21} = 2$ миллиона байт (1 миллион слов). #WE - вход разрешения записи, #RESET - начальная инициализация. #BY/R#Y - выход готовности #BY/R#Y = 1 или занятости #BY/R#Y = 0. Вход #BYTE = 0 - запись/чтение производится байтами (данные DQ7..DQ0, адрес A19,A18..A1,A0,A-1), #BYTE = 1 - запись/чтение производится словами (данные DQ15..DQ0, адрес A19,A18..A1,A0). #OE - вход разрешения выходов (#OE=0 при чтении данных). #CE (он же ~CS) - вход выбора микросхемы (#CE=0 - микросхема готова к обмену данными). VCC - напряжение питания, VSS - общий провод (земля). #WP - вспомогательные импульсы записи, NC - Not Connected. Вывод DQ15/A-1 - используется, как младший бит адреса (A-1) при байтовой организации чтения/записи и, как 16-ый вывод данных DQ15 при обмене данными словами.

Срок хранения данных не менее 20 лет, число циклов перезаписи одного сектора не менее 1.000.000.

5.4 ОДНОКРАТНО ПРОГРАММИРУЕМЫЕ ПЗУ ПЗУ (PROM, ОТР)

В качестве элементов памяти ПЗУ имеют набор плавких перемычек, которые в процессе программирования пережигаются импульсами тока. На рисунке 84 приведена схема типового ПЗУ.

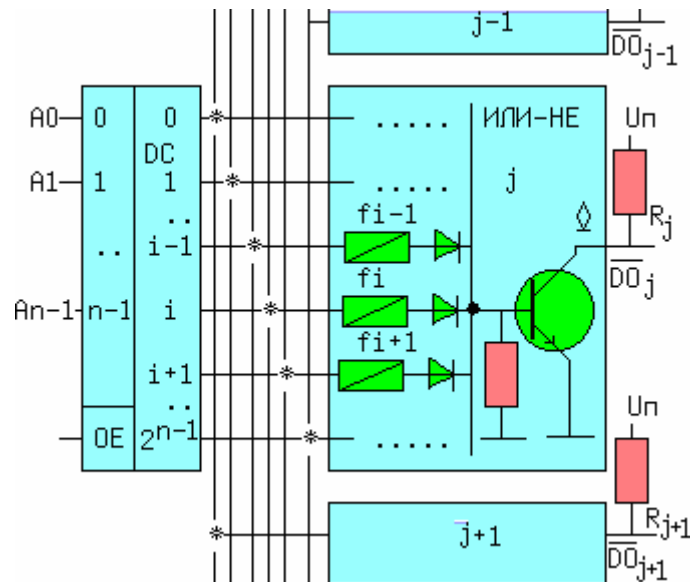


Рис.84. Типовая схема организации ПЗУ

Для любого значения адресных сигналов $A_{n-1}, A_{n-2}, \dots, A_1, A_0$ (BIN) = i (DEC), найдется единственный выход дешифратора "i" на котором сигнал $Y_i = 1$, на остальных выходах будут нули. Потенциал базы j-транзистора будет зависеть в этом случае только от наличия или отсутствия перемычки f_i . Если перемычка есть ($f_i=1$), то на базе высокий уровень сигнала, транзистор открыт и выходной сигнал $DO_j = 0$. Если перемычки нет ($f_i=0$), то $DO_j=1$. Пережиганием перемычек в соответствующих j-битах всех адресов, в микросхему записывается программа и/или данные. Выходной сигнал дешифратора $Y_i = m_i$, где m_i -минтерм входных переменных $A_0..A_{n-1}$. Диоды с резистором выполняют роль ИЛИ, а транзистор инвертирует выходной сигнал DO_j .

$$\overline{DO_j} = \sum_{i=0}^{2^n-1} f_i * m_i, \text{ где } m_i = \bigwedge_{k=0}^{n-1} /A_k$$

причем $/A_k = \sim A_k$, если A_k во входном наборе равна 0 и $/A_k = A_k$, если $A_k = 1$. Эти формулы соответствуют формулам СДНФ (12), поэтому с помощью ПЗУ с n-адресными входами и m-выходами можно реализовать любые m-логических функций с n-переменными (учитывая инверсию сигнала выходным каскадом).

5.5 ЭНЕРГОНЕЗАВИСИМОЕ СТАТИЧЕСКОЕ ОЗУ (NVSRAM). FRAM и MRAM

Для сохранения данных в статическом ОЗУ после отключения питания используют:

- встроенную в микросхему литиевую батарейку большой емкости или
- дополнительную EEPROM на том же кристалле, причем обмен данными между SRAM и EEPROM производится, либо программно, либо автоматически при падении/восстановлении напряжения

Записанная в NVSRAM информация сохраняется в течение 20-ти и более лет.

Существуют еще два типа оперативной энергонезависимой памяти FRAM и MRAM, обладающие свойствами, как ОЗУ так и ПЗУ, которые в недалеком будущем могут заменить и ОЗУ и ПЗУ. Соотношение емкость/стоимость у этих типов памяти пока невелико. Более подробно об энергонезависимой памяти и ее перспективах можно прочитать в статьях, например: "Компоненты и технологии" - №4, 2004г., стр.66 и №9, 2003г., стр.62.

5.6 УВЕЛИЧЕНИЕ РАЗРЯДНОСТИ ЯЧЕЙКИ ПАМЯТИ (СЛОВА)

Если требуется хранить данные размером в n -бит, а длина слова ячейки памяти m -бит ($n > m$), то прибегают к наращиванию длины слова. Делается это путем объединения n/m - микросхем в группы, причем все одноименные входы, кроме информационных, соединяются между собой. Например, если требуется динамическая память емкостью 256К с длиной слова равной байту, то необходимо объединить $8 / 1 = 8$ микросхем типа 565PY7, как это показано на рис. 85.

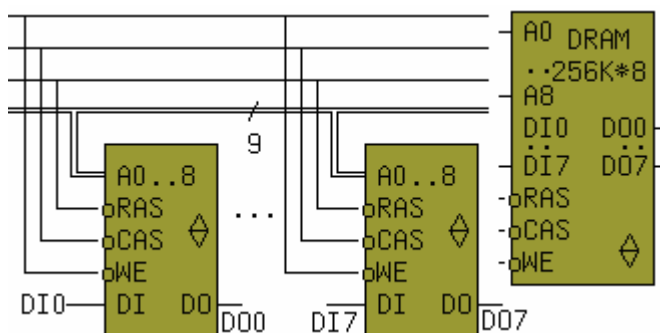


Рис.85. Увеличение числа разрядов ЯП

На рисунке девять линий адреса показаны в виде шины - т.е. группы проводников, объединенных по функциональному признаку.

5.7 УВЕЛИЧЕНИЕ КОЛИЧЕСТВА ЯЧЕЕК ПАМЯТИ

Увеличение адресного пространства ЗУ в 2^k раз требует столько же микросхем памяти и "k" дополнительных линий адреса, к уже имеющимся "n" линиям $A_{n+k-1}, \dots, A_{n+0}, A_{n-1}, A_{n-2}, \dots, A_1, A_0$. Дополнительные адресные линии $A_{n+k-1} \dots A_{n+0}$ должны разбивать требуемое адресное поле на 2^k неперекрывающихся интервалов. Каждый такой интервал равен емкости памяти отдельной микросхемы. Для решения этой задачи требуется дополнительный дешифратор "k в 2^k ". Например, если нужен блок ПЗУ емкостью $2K \times 4$, то потребуется 8 микросхем 256×4 типа 541РТ1 и один дешифратор "3 в 8", как показано на рисунке 86.

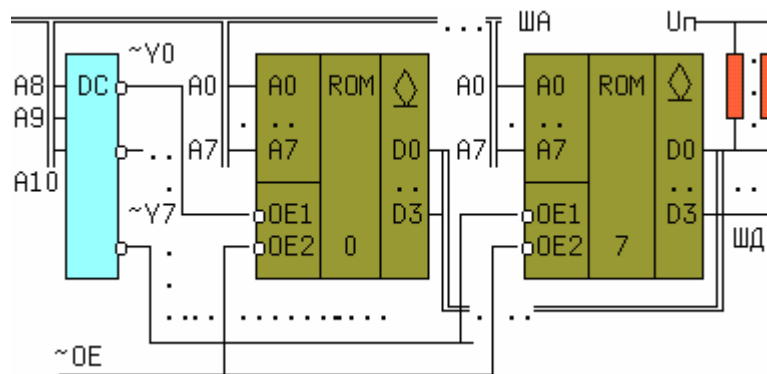


Рис.86. Увеличение числа ячеек памяти

Одноименные j-е выходы микросхем с открытым коллектором соединены с общим нагрузочным резистором R_j . Три старших дополнительных бита адреса A_{10}, A_9, A_8 выбирают одну из восьми микросхем, а восемь младших бит адреса выводят содержимое одной из 256-ти ячеек памяти на шину данных (ШД). Пусть на шину адреса (ША) поступил код $A_{10}..A_0 = 11000011010 = 61A$. На всех выходах дешифратора, кроме шестого ($A_{10}..A_8 = 110 = 6$) будет высокий уровень. Нулевой сигнал $\sim Y_6 = 0$ на входе $\sim OE_1$ шестой микросхемы разрешит прохождение записанной информации на выходы, а код $11010 = 1A(HEX) = 26(DEC)$ на адресных входах $A_7..A_0$ извлечет содержимое 26-ой ЯП и поместит его на четыре линии шины данных (ШД).

Особенностью метода является необходимость объединения по ИЛИ(И) одноименных выходов микросхем. Это можно выполнить или подключением одноименных выходов к 2^k - входным схемам ИЛИ(И) для каждого разряда, или выполнять выходные структуры микросхем памяти по схеме допускающей монтажное ИЛИ(И) с открытым коллектором или с третьим состоянием, что целесообразней. По этой причине все микросхемы памяти выпускаются с такими выходами.

5.8 ПРОГРАММИРУЕМЫЕ ЛОГИЧЕСКИЕ ИНТЕГРАЛЬНЫЕ СХЕМЫ

Проектирование современных цифровых устройств невозможно без использования программируемых логических интегральных схем (ПЛИС). Они обладают следующими преимуществами перед дискретными элементами: 1) уменьшение габаритов устройства, 2) увеличение быстродействия, 3) повышение надежности, 4) защита от копирования результатов разработки, 5) беспрецедентная скорость разработки и модификации проекта, 6) проектирование, отладку и модификацию устройства может вести разработчик на своем рабочем столе.

Схемные характеристики большинства ПЛИС (CPLD - Complex Programmable Logic Devices или FPGA - Field Programmable Gate Array) основаны на реализации дизъюнктивной нормальной формы (ДНФ) с помощью элементов И, ИЛИ, ИСКЛЮЧАЮЩЕЕ ИЛИ и инверторов. В состав ПЛИС входят также триггеры, мультиплексоры конфигурации, буферы (прямые, инверсные, тристабильные) и программируемые перемычки. Потребительские свойства разрабатываемой схемы складываются, т.о. из жесткой топологии, закладываемой на заводе и из схемных особенностей проекта, программируемых разработчиком (потребителем). Основой ПЛИС является набор матриц И и ИЛИ и макроячеек, включающих триггеры, буферы входов/выходов, элементы управления и обратные связи. Укрупненная структурная схема ПЛИС показана на рисунке 87.

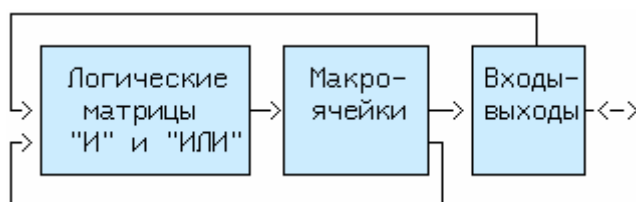


Рис.87. Типовая блок-схема ПЛИС

Входы элементов И матрицы И принято обозначать на схемах, как на рисунке 88 слева.

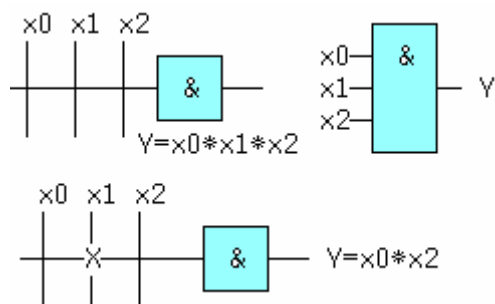


Рис.88. Обозначение перемычек

В незапрограммированном виде перемычки находятся на месте. Удаляя (значок X) с помощью программатора перемычки можно получить любое

логическое произведение (терм) входных переменных. На рисунке 88 перемычка от линии x_1 до соответствующего входа И удалена. Обычно на входы матриц И заводятся не только прямые, но и инверсные значения переменных. Выходы матриц И, как и положено в ДНФ подключаются к элементам ИЛИ.

На рис.89 показана типовая макроячейка ПЛИС. Выход элемента ИЛИ подключен через управляемый инвертор/повторитель, выполненный на ИСКЛ.ИЛИ ($=1$), во-первых к входу триггера, во-вторых через мультиплексор MUX1 и управляемый тристабильный буфер к выходу.

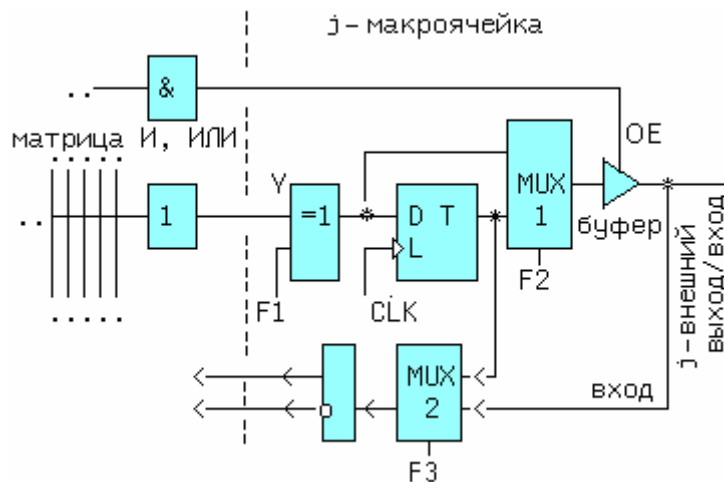


Рис.89. Схема типовой макроячейки ПЛИС

Если буфер переведен в третье состояние сигналом OE, внешний вывод микросхемы "выход/вход" может служить входом. Через MUX2, также может заводиться сигнал обратной связи с выхода триггера. Сигналы от переключателей $F(\text{use})2$ и $F(\text{use})3$ поступают на управляющие входы мультиплексоров "2->1" обеспечивая подключение одного из двух входов к единственному выходу, каждого MUX. Программирование ПЛИС под конкретную задачу может многократно производиться конфигурированием переключателей с помощью специальных программаторов под управлением САПР. ПЛИС выпускают многие фирмы: Altera, Xilinx, Atmel, Actel и др. Наиболее распространены САПР: Quartus и MAX+plus II фирмы Altera и WebPack ISE фирмы Xilinx.

5.9 ЗАДАЧИ И УПРАЖНЕНИЯ

1) Для приведенной на рисунке 89-1 микросхемы памяти дайте ответ на три вопроса: а) какой тип памяти изображен на рисунке - ram или rom? б) укажите емкость памяти (количество ячеек) в) чему равна длина слова у этой микросхемы в битах?

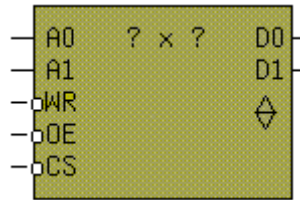


Рис.89-1. Задача 1

Пояснения : 1)Микросхема имеет три управляющих входа: выбор микросхемы $\sim CS$, разрешение выходов $\sim OE$ и разрешение записи $\sim WR$, последний вход и определяет принадлежность микросхемы к ОЗУ (ram).
 2)Наличие двух адресных входов свидетельствует о 2^2 ячейках памяти.
 3)Две линии данных соответствуют длине слова = 2

2) Найдите логическую функцию на выходе Y матрицы И,ИЛИ ПЛИС (рис.89-2).

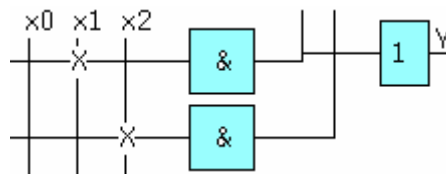


Рис.89-2. Задача 2

Пояснения : По принятым в зарубежной литературе обозначениям перед нами две трехвходовых схемы И выходами, подключенные к двум входам ИЛИ. X - обозначает отсутствие перемычки в данном месте. Поэтому ответ: $x_0 \cdot x_2 + x_0 \cdot x_1$.

3) Найдите логическую функцию на выходе Y (рис.89-3).

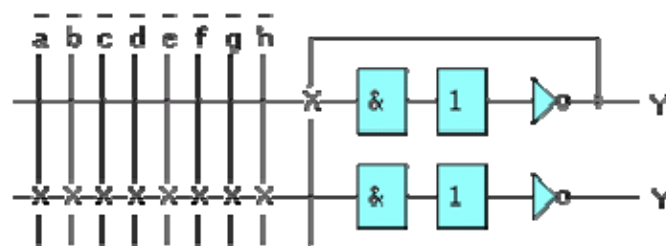


Рис.89-3. Задача 3

Пояснения: На рисунке показаны только задействованные элементы И в каждой макроячейке. Выходной буфер у каждой макроячейки инвертирующий. Поэтому верхняя матрица И,ИЛИ реализует функцию $Y_1 = \sim(\sim b * \sim c * \sim d * \sim e * \sim f * \sim g * \sim h)$. Сигнал Y_1 является входным для нижней схемы И, но не поступает на вход верхней, так как соответствующая перемычка разомкнута. Далее продолжите решение самостоятельно. Не забывайте про соотношение двойственности (Де Моргана)!

Ответ: $\sim(b + c + d + e + f + g + h)$

6. ОПРЕДЕЛЕНИЯ НЕКОТОРЫХ ПАРАМЕТРОВ ИНТЕГРАЛЬНЫХ МИКРОСХЕМ

Ниже приведены некоторые параметры в отечественном по ГОСТ 19480-89 и международном обозначении.

tзд.р.1,0 / tPHL - время задержки распространения при включении. Интервал времени между входным и выходным импульсами, в течение которого выходной сигнал интегральной микросхемы (ИМС) переходит от Н к L уровню, измеренный на уровне 0,5 или на других заданных значениях.

tзд.р.0,1 / tPLH - то же - от L к H уровню.

tзд.р.ср / tPAV - среднее время задержки распространения.

t1,0 / tTHL - время перехода при включении ИМС. Время, в течение которого выходное напряжение ИМС переходит от H к L уровню, измеренное на уровне 0,1 и 0,9 или на других заданных значениях.

t0,1 / tTLH - то же - от L к H уровню.

туст / tSU - время установления входного сигнала. Интервал времени между началом сигнала на одном заданном входе и активном переходе на другом заданном входе.

ty / tH - время удержания. Время, в течение которого сигнал удерживается на заданном входе после активного перехода на другом заданном входе.

Краз / N - коэффициент разветвления по выходу. Число единичных нагрузок, которые можно подключить к выходу ИМС.

Коб / Ni - коэффициент объединения по входу. Число входов ИМС, по которым реализуется ЛФ.

Uп / Uсс - напряжение источника питания ИМС.

СПИСОК ЛИТЕРАТУРЫ

1. Китаев Ю.В. Цифровые и микропроцессорные устройства. Учебник и задачник. WWW-адрес (<http://faculty.ifmo.ru/electron>), 2003.
2. Китаев Ю.В. Дистанционные лабораторные и практические работы. WWW-адрес (<http://faculty.ifmo.ru/electron>), 2003.
3. Угрюмов Е. П. Цифровая схемотехника. СПб., БХВ-Петербург, 2004.
4. Соловьев В. В. Проектирование цифровых систем на основе программируемых логических интегральных схем. М., Горячая линия-Телеком, 2001.
5. Бродин В.Б., Калинин А.В. Системы на микроконтроллерах и БИС программируемой логики. Изд-во "ЭКОМ", М., 2002г.
6. Пухальский Г.И., Новосельцева Т.Я. Цифровые устройства. СПб, Политехника, 1996.
7. Потемкин И.С. Функциональные узлы цифровой автоматики. Изд-во "Энергоатомиздат", М., 1988.
8. Опачий Ю.Ф. и др. Аналоговая и цифровая электроника. Учебник для вузов. М.: Горячая линия-Телеком, 1999.
9. Антонов А.П. Язык описания цифровых устройств. Изд-во "РадиоСофт", М., 2002.
10. Поляков А.К. Языки VHDL и Verilog в проектировании цифровой аппаратуры. Изд-во "СОЛОН-пресс", М., 2003.
11. Новиков Ю.В. Основы цифровой схемотехники. Базовые элементы и схемы. Методы проектирования. М., Мир, 2001.
12. Новожилов О.П. Основы цифровой техники. М., РадиоСофт, 2004.



В 2007 году СПбГУ ИТМО стал победителем конкурса инновационных образовательных программ вузов России на 2007–2008 годы. Реализация инновационной образовательной программы «Инновационная система подготовки специалистов нового поколения в области информационных и оптических технологий» позволит выйти на качественно новый уровень подготовки выпускников и удовлетворить возрастающий спрос на специалистов в информационной, оптической и других высокотехнологичных отраслях экономики.

КАФЕДРА ЭЛЕКТРОНИКИ

Заведующий кафедрой: д.т.н., проф. Г.Н. Лукьянов.

Кафедра Электроники (первоначальное название “Радиотехники”) была основана в 1945 году. Первым руководителем кафедры был С.И. Зилитинкевич известный в стране и за рубежом ученый в области физической электроники и радиотехники, активный работник высшей школы, заслуженный деятель науки и техники РСФСР, доктор технических наук, профессор ЛИТМО с 1938г., инициатор создания в ЛИТМО инженерно-физического и радиотехнического факультетов (1946г.). С.И. Зилитинкевич заведовал кафедрой с 1945 до 1978 года. Под его научным руководством аспирантами и соискателями выполнено более 50 кандидатских диссертаций, многие его ученики стали докторами наук.

В дальнейшем, с 1978г. по 1985г. кафедру возглавил к.т.н., доцент Е.К. Алахов, один из учеников С.И. Зилитинкевича.

С 1985г. по 2006г. руководителем кафедры стал д.т.н., профессор В.В. Тогатов, известный специалист в области силовой электроники и приборов для измерения параметров полупроводниковых структур.

Начиная с 2006г. кафедрой заведует д.т.н., профессор Г.Н. Лукьянов.

Основные направления кафедры связаны с разработкой приборов для лазерной и медицинской техники, приборов для измерения параметров полупроводниковых структур, а также встраиваемых цифровых и микропроцессорных устройств.

Под руководством В.В. Тогатова было разработано и изготовлено большое число приборов различного назначения:

- Измеритель параметров ультрабыстрых диодов;

- Универсальное устройство для исследования переходных процессов в силовых полупроводниковых структурах;
- Измеритель времени жизни заряда в слаболегированных областях диодных, тиристорных и транзисторных структур;
- Универсальный разрядный модуль для накачки твердотельных лазеров;
- Импульсный источник токов для накачки лазерных линеек;
- Высокочастотный разрядный модуль для систем накачки твердотельных лазеров и импульсных источников света;
- Программируемый источник света для питания галогенных ламп;
- Блок управления затвором с нарушением полного внутреннего отражения;
- и много других.

На кафедре написаны и размещены на сайте ЦДО следующие материалы для дистанционного обучения (автор Ю.В. Китаев):

- Конспект лекций по дисциплине “Электроника и микропроцессорная техника”;
- свыше 600 вопросов к обучающим и аттестующим тестам;
- 18 дистанционных лабораторных и практических работ

На кафедре имеются следующие компьютеризированные учебные лаборатории:

- АРМС – полупроводниковые приборы;
- Устройства на полупроводниковых приборах;
- Цифровая техника;
- Микропроцессорная техника
- Моделирование электронных устройств.