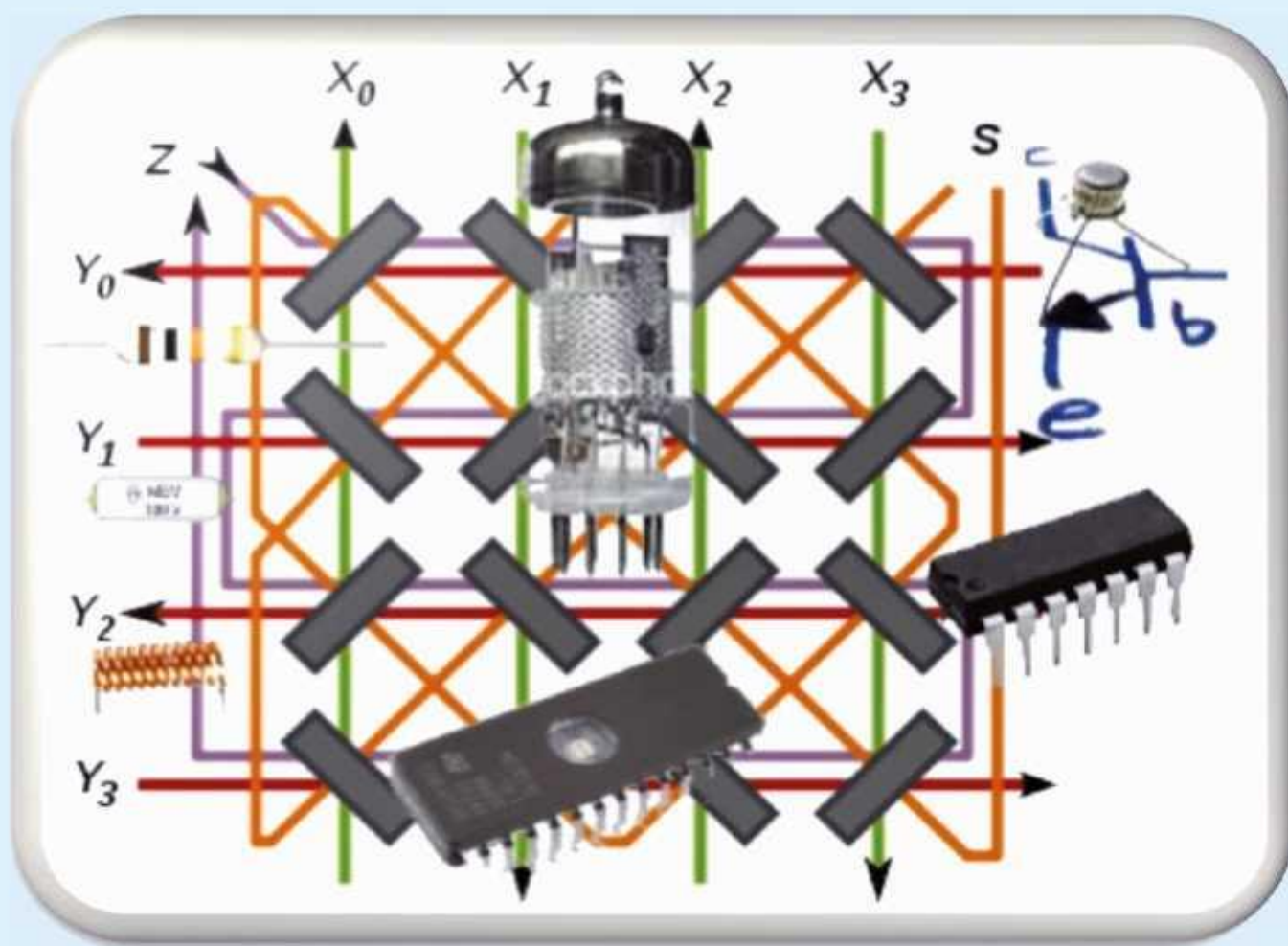


Дерягин А.В., Сабирова Ф.М.

# ОСНОВЫ АВТОМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Учебное пособие

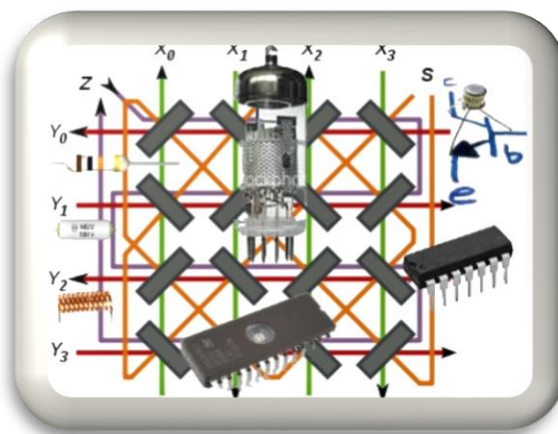


Казань – 2023

**Министерство науки и высшего образования РФ**  
**Казанский (Приволжский) федеральный университет**  
**Елабужский институт**

Дерягин А.В., Сабирова Ф.М.

**ОСНОВЫ АВТОМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ**



**Учебное пособие**

Казань – 2023

Печатается по решению Ученого совета Елабужского института Казанского (Приволжского) федерального университета (протокол № 1 от 27.01.2023)

УДК 681.5  
ББК 32.965:32.97  
Д36

**Кафедра физики**

**Рецензенты:**

**Шурыгин В.Ю.**, доцент кафедры физики, канд. физ.-мат. наук, доцент;

**Матвеев С.Н.**, доцент кафедры математики, физики и методики преподавания  
ФГБОУ ВО НГПУ, кандидат физико-математических наук

**Дерягин А. В., Сабирова Ф.М.**

**Д36** Основы автоматике и вычислительной техники: учебное пособие /  
А.В. Дерягин, Ф.М. Сабирова - Казань: Редакционно-издательский центр  
«Школа», 2023. - 124 с.

**ISBN 978-5-00162-786-9**

Учебное пособие предназначено для организации аудиторной и самостоятельной работы студентов по дисциплине «Основы автоматике и вычислительной техники» для обучающихся по направлению подготовки 44.03.05 Педагогическое образование (с двумя профилями подготовки), профили «Математика и физика», «Технология и информатика», дисциплине «Основы микроэлектроники» для обучающихся по направлению подготовки 44.03.04 Профессиональное обучение (по отраслям), профиль «Автоматизация энергетических систем», дисциплине «Схемотехника» для обучающихся по направлению подготовки 44.03.01 Педагогическое образование, профиль «Технология» и других дисциплин, включенных в учебные планы различных направлений УГС 44.00.00 Образование и педагогические науки. Оно также адресовано учителям и педагогам дополнительного образования детей.

УДК 681.5  
ББК 32.965:32.97

**ISBN 978-5-00162-786-9**

**А.В. Дерягин, Сабирова Ф.М., 2023 ©**

## Оглавление

Введение .....	5
1. ЛОГИЧЕСКИЕ ФУНКЦИИ И ЭЛЕМЕНТЫ.....	8
1.1. Формы представления чисел.....	8
1.2. Основы алгебры логики.....	12
1.3. Основные свойства и законы алгебры логики .....	17
1.4. Синтез комбинационных схем.....	19
1.5. Логические элементы с открытым коллектором и тремя состояниями выхода.....	21
2. УСТРОЙСТВА ПОСЛЕДОВАТЕЛЬНОГО ТИПА .....	24
2.1. Триггеры.....	25
2.2. Регистры .....	33
2.2.1 Регистры памяти .....	34
2.2.2 Регистры сдвига.....	35
2.2.3. Интегральные регистры.....	36
2.3. Счетчики.....	39
3. АРИФМЕТИЧЕСКИЕ УСТРОЙСТВА .....	44
3.1. Сумматоры параллельного действия .....	45
3.2. Последовательный сумматор .....	47
3.3. Арифметико-логические устройства.....	48
3.4. Мультиплексоры.....	53
3.5. Мультиплексоры TTL.....	59
3.6. Демультиплексоры и дешифраторы.....	61
3.7. Демультиплексоры-дешифраторы TTL.....	66
4. ЦАП и АЦП.....	68
4.1. Применение ЦАП .....	69
4.2. Применение АЦП .....	80
5. ОСНОВЫ АРХИТЕКТУРЫ ЭВМ .....	93
5.1. Понятие об архитектуре компьютера .....	96
5.2. Внешние устройства компьютера .....	101
5.3. Микросхемы памяти .....	109
5.4. Порты ввода/вывода.....	113
5.5. Синхронные последовательные порты .....	118
Литература.....	123

## Введение

Электронизация различных отраслей народного хозяйства, автоматизация и механизация производства, как и применение, принципиально новых технологий, направлены, в конечном счете, на качественное преобразование производительных сил, развитие которых должно сделать труд рабочих, колхозников, интеллигенции творческим и привлекательным.

Период становления электронной вычислительной техники занимает промежуток времени с момента появления в 1946 г. первой ЭВМ ЭНИАК и до 1955 г. Элементно-технологической базой первых ЭВМ были радиокомпоненты и методы монтажа радиоаппаратуры связи и экспериментальной ядерной физики. Для построения ЭВМ использовали десятки и сотни тысяч радиодеталей, а также большое количество электронных ламп. Малая надежность таких машин привела к отказу от традиционного принципа конструирования радиоаппаратуры в виде единой неразъемной конструкции. Вместо этого были предложены успешно используемые до сих пор конструкции ЭВМ в виде стоек с разъемами. Так как на платах ячеек монтировались практически все радиодетали и электронные лампы, а на стойках выполнялся лишь межъячеечный монтаж, то это позволило снизить стоимость изготовления ЭВМ, облегчить ее наладку и обеспечить восстановление работоспособности путем замены отказавших ячеек или электронных ламп.

Начиная с 1955 г. каждые последующие пять лет в вычислительной технике обновлялись конструктивно-технологические и схемно-логические принципы построения ЭВМ, что дало возможность говорить о поколениях ЭВМ.

ЭВМ первого поколения (с 1955 г.). Они строились на дискретных радиодеталях и электронных вакуумных лампах. В запоминающих устройствах этих ЭВМ в качестве накопителей информации использовались магнитные барабаны, ультразвуковые линии задержки.

Тактовая частота работы ЭВМ первого поколения находилась в пределах десятков и сотен килогерц, и применялись для решения научно-технических задач, характеризующихся малым количеством входной

информации и большим количеством вычислительных операций. Надежность ЭВМ обеспечивалась профилактическими работами, во время которых удалялись ненадежные элементы. Примером ЭВМ первого поколения являются микроЭВМ типа "Урал", "Днепр".

ЭВМ второго поколения (с 1960 г.). В качестве элементной базы в них использовались полупроводниковые приборы, миниатюрные дискретные радиодетали, а в качестве технологической базы – печатный монтаж. Носителями информации в оперативных запоминающих устройствах (ОЗУ) стали магнитные ферритовые кольцевые сердечники. В ЭВМ второго поколения, по сравнению с ЭВМ первого поколения, были снижены амплитуда информационных сигналов до десятков вольт, рабочие токи – до десятков миллиампер; значительно уменьшена мощность потребления из-за отсутствия потерь энергии в цепях накала и снижения мощности информационных сигналов; существенно возросла надежность ЭВМ за счет введения специальных электрических цепей, способных обнаруживать отказы и сбои.

ЭВМ второго поколения использовали не только для решения научно-технических задач, но и для автоматизации процесса производства. Возрастающая сложность программирования задач для этих ЭВМ вызвала необходимость автоматизации и создания автокодов и проблемно-ориентированных алгоритмических языков. Примером ЭВМ второго поколения является ЭВМ серии "Минск".

ЭВМ третьего поколения (с 1965 г.). В качестве элементной базы используется микроэлектроника. Эти ЭВМ имеют меньшие габаритные размеры и мощность потребления, значительно лучшую надежность и быстродействие по сравнению с машинами второго поколения. Постоянное улучшение параметров электронных интегральных схем позволяет совершенствовать ЭВМ и тем самым улучшать их характеристики.

В запоминающих устройствах ЭВМ третьего поколения в качестве носителей информации широко используются миниатюрные магнитные сердечники и интегральные запоминающие носители информации на основе полупроводниковых приборов. В ЭВМ этого

поколения тактовая частота работы составляет десятки мегагерц, амплитуда информационных сигналов – единицы вольт при токах в единицы миллиампер и менее. Для повышения надежности этих машин используются аппаратная и информационная избыточности, позволяющие обнаруживать и исправлять ошибки.

ЭВМ третьего поколения применяют во всех областях народного хозяйства. При их проектировании разрабатывают серии (ряды) вычислительных машин с учетом стандартизации систем элементов, блоков и устройств, а также устройств ввода – вывода. По функциональным возможностям и производительности машины одной серии существенно различаются между собой, но имеют программную совместимость.

Для более равномерной загрузки всех устройств и повышения производительности ЭВМ широко применяют мультипрограммный режим работы, т.е. решение в одной машине одновременно нескольких задач.

Примером ЭВМ третьего поколения являются ЭВМ серии ЕС ЭВМ, СМ ЭВМ.

ЭВМ четвертого поколения (с 1970 г.). К ним относятся микро-ЭВМ «Электроника 60» и др. В качестве элементной базы используются БИС с высокой степенью интеграции и интегральные носители информации в запоминающих устройствах. Во внутренней структуре высокопроизводительных ЭВМ и вычислительных системах (ВС) используют несколько процессоров. Многопроцессорные вычислительные машины позволяют значительно увеличить производительность обработки информации при решении сложных задач.

Развитие диалоговых режимов работы средств вычислительной техники дает возможность использовать ЭВМ для исследовательских разработок, автоматизации проектирования в различных, областях деятельности, решения сложных логических и экономических задач. Внедрение микроэлектронной технологии снизило стоимость электронных схем. Созданы персональные микро-ЭВМ с входным языком программирования высокого уровня, легкость

программирования и универсальность которых обеспечивает многочисленность их использования в разных областях.

В комплект подобных ЭВМ могут входить блоки сопряжения с объектом управления, специализированные пульты и индикаторные панели с обобщенными органами управления и индикации: устройства визуального отображения; регистрирующая аппаратура (графопостроители, самописцы, кассетные накопители на магнитной ленте, перфоленточное оборудование и др.).

## **1. ЛОГИЧЕСКИЕ ФУНКЦИИ И ЭЛЕМЕНТЫ**

### **1.1. Формы представления чисел**

В разные исторические периоды развития человечества для подсчетов и вычислений использовались те или иные системы счисления.

Системой счисления называют совокупность приемов и правил наименования и обозначения чисел, с помощью которых можно установить взаимно однозначное соответствие между любым числом и его представлением в виде совокупности конечного числа символов.

Например, довольно широко была распространена двенадцатеричная система. Многие предметы (ножи, вилки, тарелки, носовые платки и т.д.) и сейчас считают дюжинами. Число месяцев в году – двенадцать. Двенадцатеричная система счисления сохранилась в английской системе мер (например, 1 фут = 12 дюймов) и в денежной системе (1 шиллинг = 12 пенсов).

В древнем Вавилоне существовала весьма сложная шестидесятеричная система. Она, как и двенадцатеричная система, в какой-то степени сохранилась и до наших дней (например, в системе измерения времени: 1 ч = 60 мин, 1 мин = 60 с, аналогично в системе измерения углов:  $1^\circ = 60$  мин, 1 мин = 60 с).

У некоторых африканских племен была распространена пятеричная система счисления, у ацтеков и народов майя, (американского континента) – двадцатеричная система. У некоторых племен Австралии и Полинезии встречалась двоичная система счисления.



Десятичная система измерения возникла в Индии. Впоследствии ее стали называть арабской потому, что она была перенесена в Европу арабами. Цифры, которыми мы теперь пользуемся, – арабские.

В разное время существовали другие записи цифр, в настоящее время почти забытые. Однако до сих пор мы иногда встречаемся с записью чисел с помощью букв латинского алфавита, например на циферблатах часов, в книгах для обозначения глав или частей, на деловых бумагах для обозначения месяцев и т.д.

Имея число 12, посмотрим, каким оно получится в десятичной системе счисления: "один", "два", ..., "десять", "один-наддять", "две-наддять", т.е. после десяти вновь начинаем счет с единицы, предварительно умножив его на десять (дцать). Число десять (10) все время повторяется. Разложим число 12 по степеням числа 10, учитывая, что первое слагаемое (старшее) не должно превышать само число (12):

$$12=1\cdot 10^1+2\cdot 10^0=12,$$

где положение чисел 1 и 2 определяется степенью числа 10. Аналогично,

$$342=3\cdot 10^2+4\cdot 10^1+2\cdot 10^0=342$$

Таким образом, запишем десятичное число в общем виде:

$$A = a10^n + b10^{n-1} + \dots + p10^1 + q10^0 = ab\dots pq.$$

Величина числа  $A$  определяется коэффициентами при степенях числа 10. Отсюда видно, что число 10 является основанием системы счисления, которая в данном случае называется десятичной. В этой системе для записи чисел используются десять цифр – от 0 до 9. Истинное значение каждой цифры определяется местом ее в числе, т.е. степенью числа 10. В числе 12 цифра 2-это две единицы, а в числе 243 цифра 2-две сотни.

Система счисления, в которой величина цифры определяется ее местоположением (позицией), называется позиционной. Таким образом, десятичная система счисления является позиционной. Римская система счисления не является позиционной, т.е. положение цифр не меняет ее значения. Например, число 105 запишем как CV, а число 95 – как VC. При этом знак V в обоих случаях имеет одно и то

же значение – 5 единиц, только в одном случае он прибавляется к сотне (С), а в другом случае – вычитается.

В двоичной системе счисления основанием является число 2. В этом случае для записи чисел используют две цифры: 0 и 1.

Перевод числа из десятичной системы счисления в двоичную производится методом последовательного деления числа на 2 до тех пор, пока частное от деления не станет равным 1. Число в двоичной системе счисления записывается в виде остатков от деления, начиная с последнего частного, справа налево:

$$\begin{array}{r}
 123 \quad | 2 \\
 \hline
 122 \quad 61 \quad | 2 \\
 \hline
 1 \quad 60 \quad 30 \quad | 2 \\
 \hline
 \quad 1 \quad 30 \quad 15 \quad | 2 \\
 \hline
 \quad \quad 0 \quad 14 \quad 7 \quad | 2 \\
 \hline
 \quad \quad \quad 1 \quad 6 \quad 3 \quad | 2 \\
 \hline
 \quad \quad \quad \quad 1 \quad 2 \quad 1 \\
 \hline
 \quad \quad \quad \quad \quad 1
 \end{array}$$

Полученные остатки записываем в обратном порядке, таким образом:

$$123_{(10)} = 1111011_{(2)}, \text{ т.е.}$$

$$123 = 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 64 + 32 + 16 + 8 + 2 + 1.$$

Перевод десятичного дробного числа в двоичную систему осуществляется в два этапа: вначале переводится целая часть числа (см. выше), затем дробная. Для перевода десятичной дроби 0.41 в двоичную систему, необходимо выполнить последовательное умножение дроби на 2, до тех пор, пока дробная часть не станет равной 0 или пока не будет достигнута заданная точность вычисления. Получаем:

$$0.41 \cdot 2 = 0.82 (0)$$

$$0.82 \cdot 2 = 1.64 (1)$$

$$0.64 \cdot 2 = 1.28 (1)$$

$$0.28 \cdot 2 = 0.56 (0)$$

$$0.56 \cdot 2 = 1.12 (1)$$

$$0.12 \cdot 2 = 0.24 (0)$$

$$0.24 \cdot 2 = 0.48 (0)$$

$$0.48 \cdot 2 = 0.96 (0)$$

$$0.96 \cdot 2 = 1.92 (1)$$

$$0.92 \cdot 2 = 1.84 (1)$$

$$0.84 \cdot 2 = 1.68 (1)$$

По рассмотренным правилам числа можно переводить и в другие широко распространенные системы счисления – восьмеричную, шестнадцатеричную, двоично-десятичную. Во всех случаях умножение или деление переводимых чисел производится на основании новой системы счисления.

В табл. 1.1 приведены коды системы счисления, из которой видно, что двоично-десятичный код отличается от десятичного тем, что в нем каждое число десятичного разряда записывается в двоичном коде.

Таблица 1.1

Десятичный	Двоичный	Восьмеричный	Шестнадцатеричный	Двоично-десятичный
0	0	0	0	0000
1	1	1	1	0001
2	10	2	2	0010
3	11	3	3	0011
4	100	4	4	0100
5	101	5	5	0101
6	110	6	6	0110
7	111	7	7	0111
8	1000	10	8	1000
9	1001	11	9	1001
10	1010	12	A	0001 0000
11	1011	13	B	0001 0001
12	1100	14	C	0001 0010
13	1101	15	D	0001 0011
14	1110	16	E	0001 0100
15	1111	17	F	0001 0101
16	10000	20	10	0001 0110
17	10001	21	11	0001 0111
18	10010	22	12	0001 1000

Примеры перевод двоичного числа в восьмеричную, шестнадцатеричную, двоично-десятичную системы счисления:

$$364_{(10)} = \underline{101101100}_{(2)} = 554_{(8)}$$

5      5      4

$$364_{(10)} = \underline{101101100}_{(2)} = 16C_{(16)}$$

1    6      C

$$364_{(10)} = 001101100100_{(2/10)}$$

## 1.2. Основы алгебры логики

Все устройства ЭВМ состоят из элементарных логических схем. Работа этих схем основана на законах и правилах алгебры логики, которая оперирует двумя понятиями: истинности и ложности высказывания. В соответствии с такой двоичной природой высказываний условились называть их логическими двоичными переменными и обозначать 1 в случае истинности и 0 в случае ложности. Примерами логических переменных являются высказывания: А = "Земля плоская", В = "Автомобиль имеет двигатель".

На основании этих высказываний можно записать А = 0; В = 1, так как высказывание А ложно, а высказывание В истинно.

Высказывания могут быть простыми и сложными: простые содержат одно законченное утверждение, сложные образуются из двух или большего числа простых высказываний, связанных между собой некоторыми логическими связями. Формализация и преобразование связей между логическими переменными осуществляется в соответствии с правилами алгебры логики, называемой алгеброй Булл (в честь ее автора – английского математика Джорджа Буля).

Две логические переменные А и В, принимающие значения 0 или 1, могут образовывать логические функции. Из 16 возможных функций двух переменных наибольший практический интерес представляют функции отрицания, логического умножения и логического сложения.

Логическое отрицание НЕ переменной  $X$  есть логическая функция  $Y$ , которая истинна только тогда, когда ложно  $X$  и наоборот.

В алгебре логики любые функции удобно изображать в виде таблицы соответствия всех возможных комбинаций входных логических переменных и выходной логической функции, называемой таблицей истинности.

Таблица 1.2.

Таблица истинности для функции логического отрицания НЕ.

$Y$	$0$	$1$
$X$	$1$	$0$

где  $X$  – входная переменная,  $Y$  – выходная функция.

Функцию НЕ в символах алгебры логики записывают следующим образом:

$$Y = \bar{X} = \text{not } X$$

Графически эта функция обозначается кружком на входе или выходе логического символа (Рис.1.1).



Рис.1.1. Графическое изображение функции НЕ.

Пример реализации функции НЕ и ее временные диаграммы (Рис.1.2).

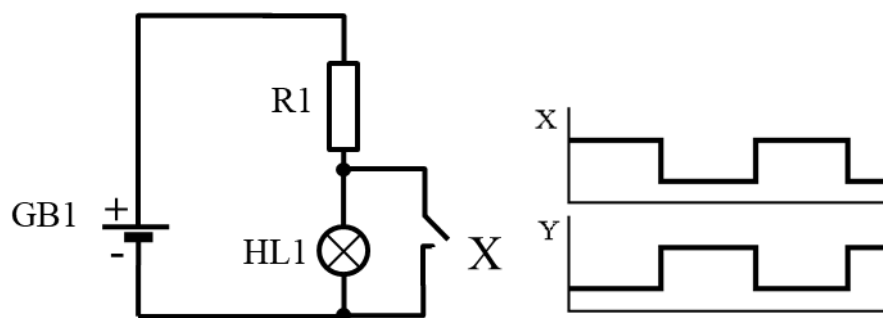


Рис.1.2. Реализация функции НЕ и ее временные диаграммы.

Примем положение переключателя за входную переменную и обозначим замкнутое состояние ключа логической единицей ( $X=1$ ), а разомкнутое – логическим нулем ( $X=0$ ).

Логическое умножение И двух переменных А и В есть логическая функция Х, которая истинна только тогда, когда одновременно истинны входные переменные. Для функции логического умножения таблица истинности имеет вид (Таб.1.3.)

Таблица.1.3.

Таблица истинности функции логического умножения

A	B	A&B
0	0	0
0	1	0
1	0	0
1	1	1

В алгебре логики логическое умножение «И» иногда называют конъюнкцией и записывают в виде  $X=AB$  или  $X=A \wedge B$  или  $X=A \& B$  или  $X= A \text{ and } B$ . Графически функция И обозначается в виде прямоугольника, внутри которого ставится символ & (Рис.1.3.).

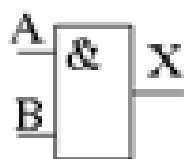


Рис.1.3. Графическое изображения функции И.

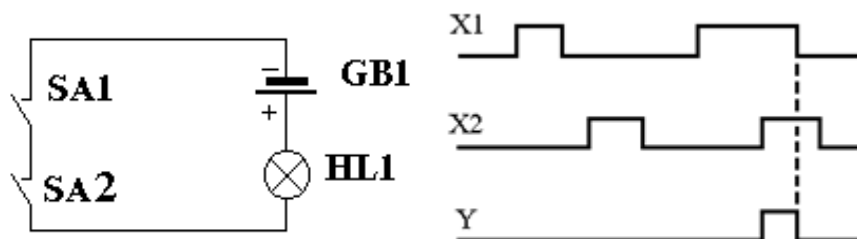


Рис.1.4. Пример реализации логической функции И, диаграммы входных и выходных сигналов.

Логическая сумма «ИЛИ» переменных А и В есть логическая функция Х, которая истинна, когда хотя бы одна из входных переменных истинна. Для логической суммы таблица истинности имеет вид

Логическая сумма в символах алгебры логики записывается так:  $X=A+B=A\vee B =A\text{or}B$ .

Таблица.1.4.

Таблица истинности функции логической суммы.

A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

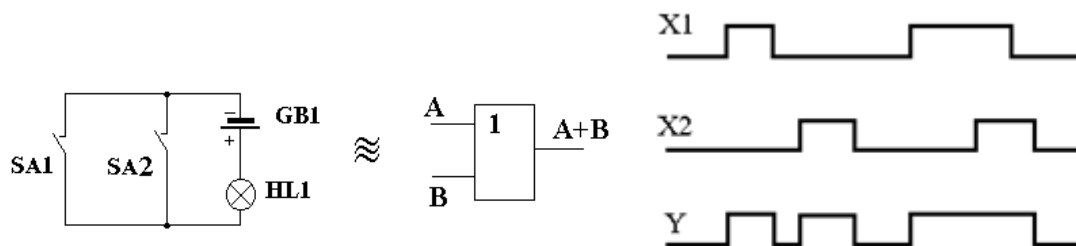


Рис.1.5. Пример реализации логической функции ИЛИ, диаграммы входных и выходных сигналов.

Любые сложные высказывания могут быть составлены из рассмотренных выше "НЕ", "И", "ИЛИ". Однако в практике чаще используются комбинации этих логических элементов, например, "И-НЕ" или "ИЛИ-НЕ". Выполнение логических операций и их комбинаций реализуется в специальных электронных устройствах – микросхемах. Например, в микросхемах серии К155 в качестве базового логического элемента реализуется элемент "И-НЕ". Дело в том, что с помощью этого элемента можно скомбинировать много других элементов, в том числе и элементарные "И", "НЕ", "ИЛИ".

Таблица истинности для элемента "И-НЕ" представлена в табл. 1.5, обозначение в электронных схемах – на рис.1.6.

Таблица.1.5.

Таблица истинности элемента "И-НЕ".

A	B	A+B
0	0	1
0	1	0
1	0	0
1	1	0

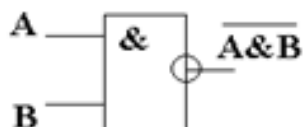


Рис.1.6. Графическое изображения функции "И-НЕ".

Приведенный на этом рисунке элемент "И-НЕ" чаще называют элементом "2И- НЕ", указывая при этом цифрой 2 на число входов элемента. Количество элементов в одном корпусе может быть больше одного, например, в микросхеме К155ЛА3 таких элементов 4. В этом случае записывают 4(2И-НЕ) – четыре двухвходовых элемента "И" с инверсией.

Функции, образованные логическими переменными, можно преобразовывать в соответствии с правилами или законами алгебры логики. При этом стремятся минимизировать логическое выражение, т.е. привести его к виду, удобному для практической реализации на логических элементах.



### 1.3. Основные свойства и законы алгебры логики

#### СВОЙСТВА АЛГЕБРЫ

Логическое произведение любого аргумента на 0 всегда равно 0:

$$X \wedge 0 = 0$$

Логическое произведение любого аргумента на 1 равно значению аргумента:

$$A \wedge 1 = A$$

Логическое произведение одних и тех же аргументов равно аргументу:

$$A \wedge A = A$$

Логическое произведение аргумента с его инверсией равно 0:

$$A \wedge \bar{A} = 0$$

Логическая сумма любого аргумента с нулем, равно значению аргумента:

$$A + 0 = A$$

Логическая сумма одних и тех же аргументов равно аргументу.

$$A + A = A$$

Логическая сумма любого аргумента с 1, равно 1.

$$A + 1 = 1$$

Логическое произведение аргумента с его инверсией равно 1:

$$A + \bar{A} = 1$$

Двойная инверсия аргумента дает его истинное значение:

$$\bar{\bar{A}} = A$$

#### ЗАКОНЫ:

Переместительный:

$$A + B = B + A$$

$$A \cdot B = B \cdot A$$

Сочетательный

$$(A + B) + C = A + (B + C)$$

$$(A \cdot B) \cdot C = A \cdot (B \cdot C)$$

Распределительный

$$A \cdot (B + C) = A \cdot B + A \cdot C$$

$$A + (B \cdot C) = (A + B) \cdot (A + C)$$

Инверсия (закон Де Моргана)

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

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

Склеивание

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

$$(A+B) \cdot (A+\overline{B}) = A$$

Поглощение

$$A + A \cdot B = A$$

$$A(A+B) = A$$

Введем понятия дизъюнктивной нормальной формы (ДНФ) и конъюнктивной нормальной формы (КНФ).

ДНФ – это функция, представляющая собой сумму, каждое слагаемое которой является произведением всех входных переменных или их дополнений:

$$Z = A \cdot B \cdot C + A \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot \overline{C} + \overline{A} \cdot \overline{B} \cdot \overline{C}$$

ДНФ является избыточной, если в ней имеется избыток числа членов и переменных по сравнению с необходимым для определения данной функции. В общем случае ДНФ является избыточной. Если удалить избыточные члены и переменные (т.е. упростить выражение), то получится так называемая минимальная сумма. Приведенная выше стандартная сумма упрощается следующим образом:

$$Z = (C + \overline{C})A \cdot B + (B + \overline{B})\overline{A} \cdot \overline{C},$$

и так как  $C + \overline{C} = B + \overline{B} = 1$ , то

$$Z = A \cdot B + \overline{A} \cdot \overline{C}$$

Это выражение является минимальной суммой.

Конъюнктивная нормальная форма (КНФ) – это функция, представляющая собой произведение членов, каждый из которых является суммой всех входных переменных или их дополнений:

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

Как и ДНФ, КНФ избыточна. Минимальное произведение является упрощенным не избыточным произведением.

Если мы упростим приведенное выше стандартное произведение Z, то получим

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

Так как  $C \cdot \bar{C} = B \cdot \bar{B} = 0$ , то

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

Правила алгебры логики применяют для преобразования исходных выражений к виду, удобному для их практической реализации и для анализа схем, построенных на ключах, релейных схемах и цифровых логических элементах.

#### 1.4. Синтез комбинационных схем

Комбинационным устройством называют устройство, состояние выходов которого зависит от текущего состояния на входе и не зависит от предыдущего состояния устройства, т.е. для любой комбинации входных сигналов – состояние устройства однозначно определено.

Комбинационное устройство может быть задано уравнением, таблицей истинности и словесным описанием.

Пусть устройство задано уравнением;  $Y = X1 \& (\bar{X}1 \vee X2)$

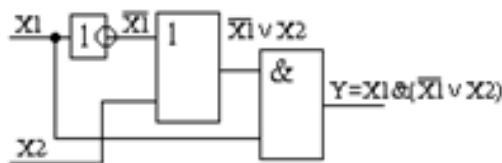


Рис.1.7. Схема исходного выражения.

Упрощая исходное выражение – упростим принципиальную схему устройства:  $X1 \& (\bar{X}1 \vee X2) =$  (раскрываем скобки)=

$$= X1 \& \bar{X}1 \vee X1 \& X2 = 0 \vee X1 \& X2 = X1 \& X2$$

В результате преобразований получаем следующую схему:

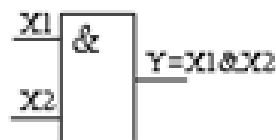


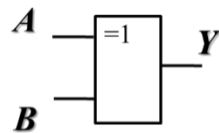
Рис.1.8. Схема выражения, после преобразования

По таблице истинности составьте булево выражение, упростите его и составьте схему.

X1	X2	Y
0	0	0
0	1	1
1	0	1
1	1	0

$$\left. \begin{array}{l} X1\overline{X2} \\ \overline{X1}X2 \end{array} \right\} Y = X1 \& \overline{X2} \vee \overline{X1}X2$$

$$X1 \cdot \overline{X2} + \overline{X1} \cdot X2 = \overline{\overline{X1 \cdot \overline{X2} + \overline{X1} \cdot X2}} = \overline{\overline{X1} + X2} \cdot \overline{X1 + \overline{X2}} = \overline{\overline{X1} + X2 + X1 + \overline{X2}} = \overline{X1 \cdot \overline{X1} + \overline{X1} \cdot \overline{X2} + X1 \cdot X2 + X2 \cdot \overline{X2}} = \overline{\overline{X1} \cdot \overline{X2} + X1 \cdot X2}$$



1/4 К155ЛП5

Рис.1.9. Конечная схема К155ЛП5 – 4("Исключающее ИЛИ")

Рассмотрим пример. В трехэтажном доме лестничная клетка освещается одной лампочкой. На каждом из этажей имеется выключатель. Необходимо спроектировать устройство включения и выключения освещения, переключением любого из выключателей. Единичный уровень на выходе устройства соответствует включенной лампочки, нулевой- выключенной.

Составим таблицу истинности.

Лестничная площадка освещается, если включен только один из выключателей или сразу все, и не освещается, если выключены все или включены любые два.

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

$$\begin{aligned}
Y &= A * \bar{B} * \bar{C} + \bar{A} * B * \bar{C} + \bar{A} * \bar{B} * C + A * B * C = \\
&= \bar{C} * (A * \bar{B} + \bar{A} * B) + C * (\bar{A} * \bar{B} + A * B) = \\
&\left[ \bar{A} * \bar{B} + A * B = \overline{\bar{A} * \bar{B} * A * B} = \overline{A + B * \bar{A} * \bar{B}} = \overline{(A + B) * (\bar{A} + \bar{B})} \right] \\
&= \\
&= \left[ \overline{(A + B) * (\bar{A} + \bar{B})} = \overline{A * \bar{A} + \bar{A} * B + A * \bar{B} + B * \bar{B}} \right. \\
&\quad \left. = \overline{\bar{A} * B + A * \bar{B}} \right] = \\
&= \bar{C} \&(A \oplus B) \vee C \&(\bar{A} \oplus \bar{B}) = A \oplus B \oplus C
\end{aligned}$$

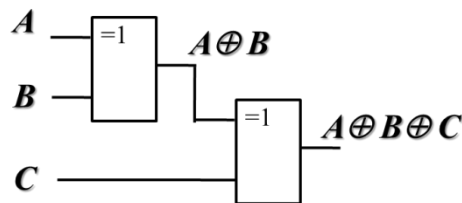


Рис.1.10. Конечная схема.

## 1.5. Логические элементы с открытым коллектором и тремя состояниями выхода

Физические параметры и функциональные возможности логических элементов (ЛЭ) зависят от выполнения выходного каскада. Наиболее часто используются три типа схем выходных каскадов.

1. На рис. 1.11 показан ЛЭ со стандартным выходом (с активной нагрузкой). Этот выходной каскад в различных сериях может иметь различные модификации (различные значения R; вместо верхнего транзистора и диода может использоваться составной транзистор).

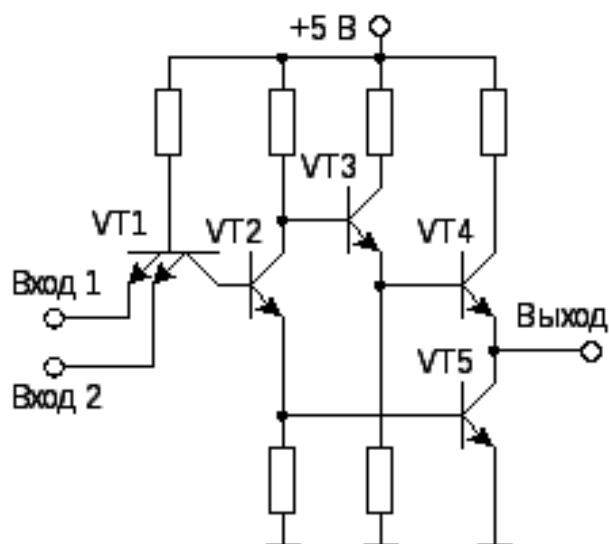


Рис.1.11. ЛЭ со стандартным выходом

ЛЭ со стандартным выходом имеет нагрузочную способность  $n=10$  ( $n = 20$  для 555 серии,  $n=33$  для 1531 серии); такой же выходной каскад имеют ЛЭ с повышенной нагрузочной способностью (для большинства серии  $n=30$ ).

2. В ЛЭ с открытым коллектором (рис. 1.12) в качестве выходного каскада используется транзистор, коллектор которого не подключен к нагрузке. Эти транзисторы изготавливаются на разное допустимое напряжение питания: +5, +15, +30, +35 В и др. Выходы таких ЛЭ должны быть подключены с помощью внешнего резистора к соответствующему источнику питания  $U_{и.и2}$  и  $U_{и.и.1}=5В$ . В таблице 2.10 приведены средние значения основных параметров ЛЭ с открытым коллектором ТТЛ-серий.

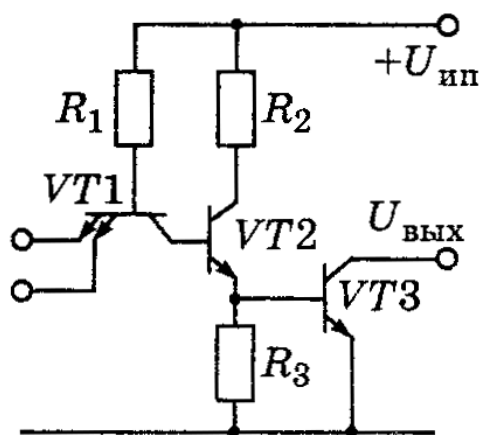


Рис.1.12. ЛЭ с открытым коллектором.

Логический элемент с открытым коллектором. Наиболее важным свойством ЛЭ с открытым коллектором является возможность реализации с их помощью логических функций, называемых «монтажное ИЛИ» и «монтажное И».

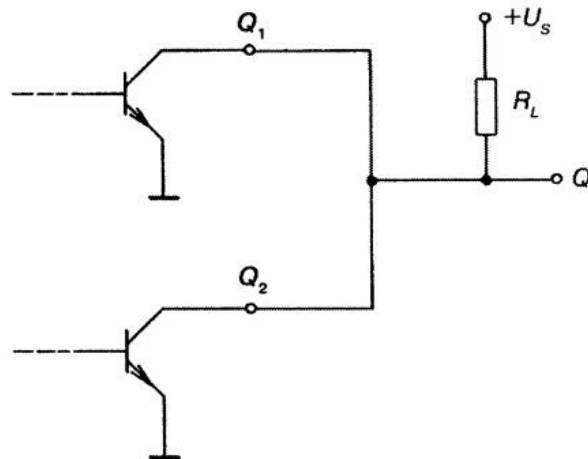


Рис. 1.13. Совместное подключение ТТЛ-элементов с открытым коллектором.

3. На рис. 1.14, показан ЛЭ с тремя состояниями выхода: ЛЭ имеет управляющий вход ОЕ (*Output Enable*-разрешение выхода), одно из значений сигнала на котором переводит оба выходных транзистора в закрытое состояние. При  $E=1$ , ЛЭ с тремя состояниями работает так же, как и ЛЭ со стандартным выходом. При  $E=0$  напряжение на базах транзисторов  $VT_3$  и  $VT_4$  принимает низкий уровень, что делает невозможным протекание выходных токов через транзисторы. Выходной вывод окажется отсоединенным от входных цепей и от обеих шин питания. В этом состоянии микросхема потребляет значительно меньшую мощность. “Z”-состоянием, или так называемым третьим состоянием, называют преднамеренный отказ в выдаче информации логическим элементом путем размыкания его выходной цепи при подаче соответствующего управляющего сигнала, при котором выходное сопротивление элемента стремится к бесконечности, а выходной ток — к нулю.

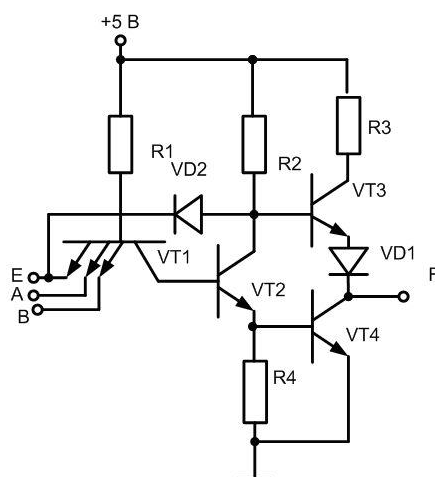


Рис.1.14. ЛЭ с тремя состояниями выхода.

## 2. УСТРОЙСТВА ПОСЛЕДОВАТЕЛЬНОГО ТИПА

Простейшими цифровыми автоматами с памятью являются триггеры. Триггер – это устройство последовательного типа с двумя устойчивыми состояниями равновесия, предназначенное для записи и хранения информации. Под действием входных сигналов триггер может переключаться из одного устойчивого состояния в другое. При этом напряжение на его выходе скачкообразно изменяется. Как правило, триггер имеет два выхода прямой и инверсный. Число входов зависит от структуры и функций, выполняемых триггером. По способу записи информации- триггеры делят на асинхронные и синхронизируемые (тактируемые). В асинхронных триггерах информация может записываться непрерывно и определяется информационными сигналами, действующими на входах в данный момент времени. Если информация заносится в триггер только в момент действия так называемого синхронизирующего сигнала, то такой триггер называют синхронизируемым или тактируемым. Помимо информационных входов синхронизируемые триггеры имеют тактовый вход (синхронизации). В цифровой технике приняты следующие обозначения входов триггеров.

S (Set установка) – отдельный вход установки в единичное состояние (напряжение высокого уровня на прямом выходе Q);

R (Reset сброс) – отдельный вход установки в нулевое состояние



(напряжение низкого уровня на прямом выходе;

Иногда входы R и S называют по другому: clear – очистка (сброс) и preset – предварительная установка соответственно.

D (Data данные) – информационный вход, вход данных (на него подается информация, предназначенная для занесения в триггер);

C (Clock часы) – тактовый вход, вход синхронизации;

T (Toggle переключатель)- счетный вход.

## 2.1. Триггеры

### 2.1.1. Статичные триггеры

Наибольшее распространение в цифровых устройствах получили RS-триггер, тактируемый D-триггер. Рассмотрим функциональные возможности каждого из них.

Асинхронный RS-триггер. В зависимости от логической структуры различают RS-триггеры с прямыми и инверсными входами. Их схемы и условные обозначения приведены на рис.2.1. Триггеры такого типа построены на двух логических элементах; 2ИЛИ-НЕ-триггер с прямыми (рис.2.1.в), 2И-НЕ-триггер с инверсными входами (рис.2.1.а).

Выход каждого из элементов подключен к одному из входов другого элемента, что обеспечивает триггеру два устойчивых состояния.

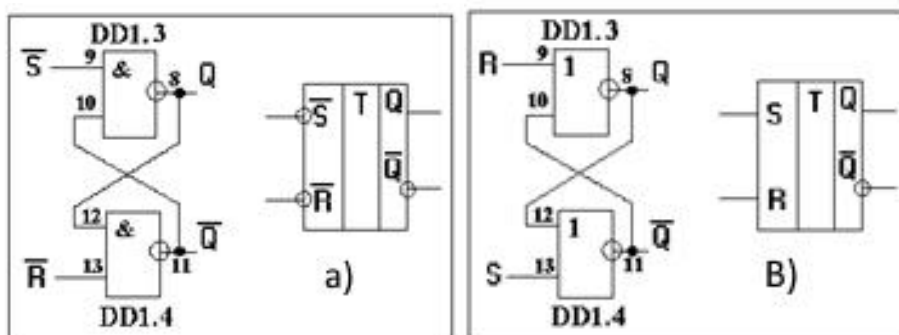


Рис.2.1. Асинхронный RS- триггер.

Таблица 2.1.

Таблица состояний триггера с прямыми и инверсными входами.

RS Триггеры с прямыми входами			RS Триггеры с инверсными входами		
S	R	Состояние	$\bar{S}$	$\bar{R}$	Состояние
0	0	Хранение	0	0	Запрещенное
1	0	Запись 1	1	0	Запись 0
0	1	Запись 0	0	1	Запись 1
1	1	Запрещенное	1	1	Хранение

Активным называют логический уровень, действующий на входе логического элемента и однозначно определяющий логический уровень выходного сигнала (независимо от логических уровней, действующих на остальных входах). Для элементов ИЛИ-НЕ за активный уровень принимают высокий уровень, а для элементов И-НЕ – низкий уровень. Уровни, подача которых на один из входов не приводит к изменению логического уровня на выходе элемента, называют – пассивными. Уровни  $\bar{Q}'$  и  $Q'$  обозначают логические уровни на выходах триггера после подачи информации на его входы.

Для триггера с прямыми входами  $Q' = 1$  при  $S=1$  и  $R=0$ ;  $Q'=0$  при  $S=0$  и  $R=1$ ;  $Q'=Q$  при  $S=0$  и  $R=0$ . При  $S=R=1$  состояние триггера будет неопределенным, так как во время действия информационных сигналов логические уровни на выходах триггера одинаковы ( $Q'=Q=0$ ), а после окончания их действия триггер может равновероятно принять любое из устойчивых состояний – поэтому такая комбинация является запрещенной.

Режим  $S=1$ ,  $R=0$  называют режимом записи 1 (так как  $Q'=1$ .); режим  $S=0$  и  $R=1$  режимом записи 0. Режим  $S=0$ ,  $R=0$  называется режимом хранения информации, так как информация на выходе остается неизменной. Для триггера с инверсными входами режим записи логической 1 реализуется при  $S=0$ ,  $R=1$ , режим записи логического 0 при  $S=1$ ,  $R=0$ . При  $S=R=1$  обеспечивается хранение информации. Комбинация  $S=R=0$  является запрещенной.

Синхронный (тактируемый) RS-триггер имеет управляющий сигнал  $C$  на входе, который при  $C=1$  разрешает переключение по закону RS -триггера, при  $C=0$  триггер сохраняет свое состояние. Схема и временные диаграммы тактируемого синхронного RS -триггера приведены на рис.2.2.

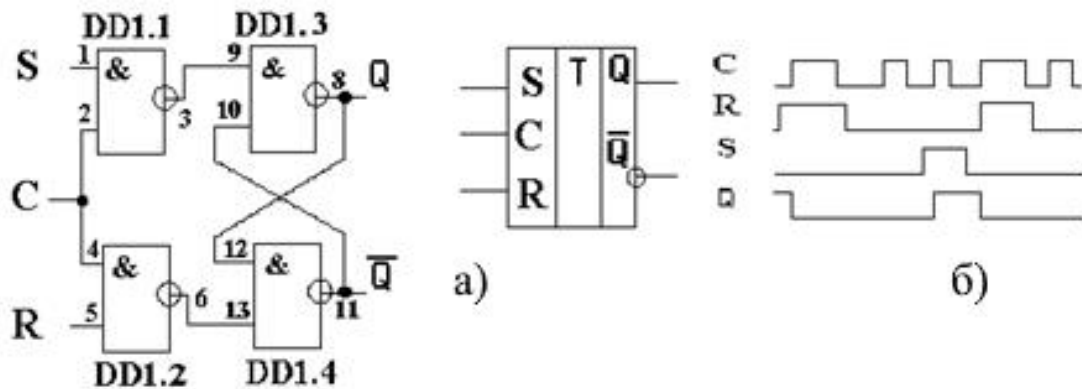


Рис.2.2. Схема (а)и временные диаграммы (б) тактируемого синхронного RS –триггера.

Тактируемый D-триггер. Он имеет информационный выход и вход синхронизации. Одна из возможных структурных схем одноканального D- триггера и его условное обозначение показаны на рис. 2.3.

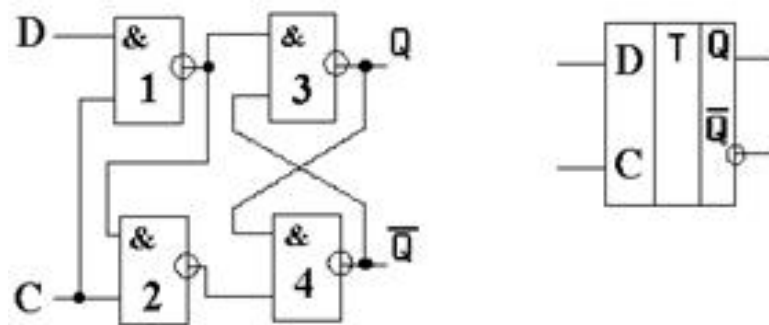


Рис. 2.3. Структурная схема D- триггера и его условное обозначение.

Если уровень сигнала на входе  $C = 0$ , состояние триггера устойчиво и не зависит от уровня сигнала на информационном входе. При этом на входы RS -триггера с инверсными входами (элементы 3 и 4) поступают пассивные уровни ( $R = S = 1$ ). При подаче на вход синхронизации уровня  $C = 1$  информация на прямом выходе будет

повторять информацию, подаваемую на вход D. Таким образом, при  $C = 0$   $Q'=Q=0$ , а при  $C = 1$   $Q' = D$ . Таблица истинности тактируемого D – триггера приведены в таблице 2.2.

Таблица 2.2.

Таблица истинности тактируемого D – триггера

D	Q	Q'
0	0	0
0	1	0
1	0	1
1	1	1

Здесь Q означает логический уровень на прямом выходе до подачи импульса синхронизации, а Q'- логический уровень на этом выходе после подачи импульса синхронизации.

На рис.2.4. изображены временные диаграммы тактируемого D - триггера. В таком триггере происходит задержка сигнала на выходе по отношению к сигналу, поданному на вход, на время паузы между синхросигналами. Для устойчивой работы триггера необходимо, чтобы в течение синхроимпульса информация на входе была неизменной.

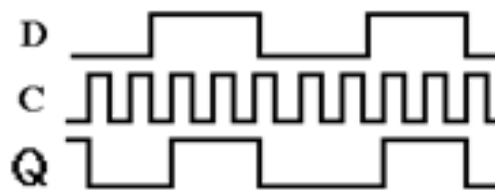


Рис. 2.4. Временные диаграммы D -триггера.

### 2.1.2 Динамичные триггеры

Простейшими цифровыми автоматами с памятью являются триггеры. Триггер – это устройство последовательного типа с двумя устойчивыми состояниями равновесия, предназначенное для записи и хранения информации. Под действием входных сигналов триггер может переключаться из одного устойчивого состояния в другое. При

этом напряжение на его выходе скачкообразно изменяется. Как правило, триггер имеет два выхода прямой и инверсный. Число входов зависит от структуры и функций, выполняемых триггером.

Триггер может управляться импульсом в течение всей его длительности (УРОВНЕМ) или фронтами импульса, его перепадами. Триггеры второго типа можно также разделить на две категории в зависимости от того, каким фронтом импульса они управляются нарастающим или спадающим. В условных обозначениях тип триггера указывается буквами S, R, D, которые проставляются у входов. Управляющий вход обозначается буквой C, командная зависимость его обозначается специальным символом (Рис.2.5.)

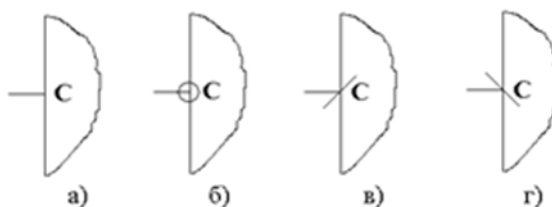


Рис.2.5. а) высокий уровень; б) низкий уровень; в) передний фронт; г) задний фронт

Для надежной и четкой работы триггерных ячеек предназначены двухступенчатые триггеры, называемые master-slave, что лучше всего переводится как "мастер-помощник" (слово master имеет еще одно значение: хозяин). Структурная схема такого триггера, состоящего из двух RSC-триггеров, показана на рис.2.6.

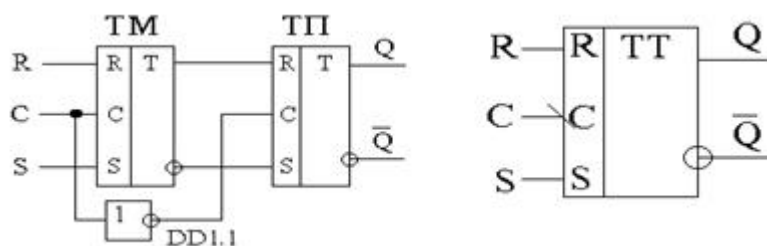


Рис. 2.6. Структурная схема триггера и его условное обозначение.

Входы C обоих триггеров TM (мастер) и TP (помощник) соединены между собой через инвертор DD1.1. Составным триггером TM – TP управляет полный (с фронтом и срезом) тактовый импульс C. Действительно, если каждый из триггеров имеет установку положительным перепадом, входная RS -комбинация будет записана в

ТМ в момент прихода положительного перепада тактового импульса С. В этот момент в ТП информация попасть не может. Когда придет отрицательный перепад входного импульса С, на выходе инвертора DD1.1 он появится как положительный. Следовательно, положительный перепад импульса С перепишет данные от выходов  $\bar{Q}$  и Q в ТП. Наиболее часто в цифровых интегральных микросхемах, а также в импульсных устройствах применяют триггеры с единственным входом данных D (data), так называемые D -триггеры. Схема D -триггера (рис.2.7.) отличается от схемы RSC -триггера (рис.2.6.) наличием инвертора DD1.2, добавленного между входами S и R.

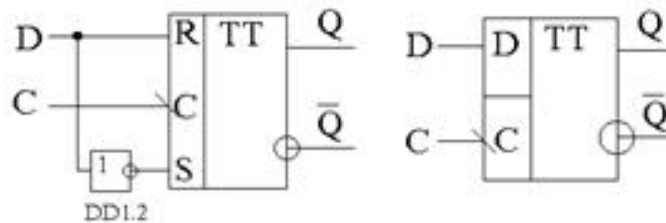


Рис. 2.7. Структурная схема триггера и его условное обозначение.

### МИКРОСХЕМА К155ТМ2

Микросхема ТМ2 содержит два независимых D -триггера, имеющих общую цепь питания (Рис. 2.8).

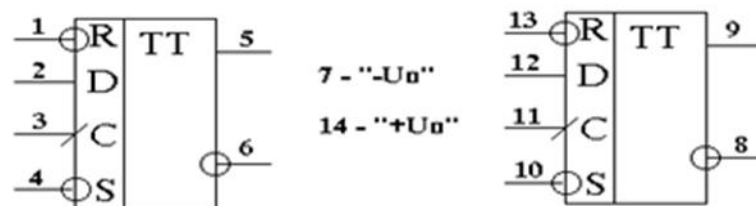


Рис. 2.8. Условное обозначение микросхемы К155ТМ2.

У каждого триггера есть входы D, S и R, а также комплементарные выходы  $\bar{Q}$  и Q. Входы S и R -асинхронные, работают независимо от сигнала на тактовом входе; активный уровень для них низкий. Сигналы от входа D передается на выходы  $\bar{Q}$  и Q по положительному перепаду импульса на тактовом входе С (от 0 к 1). Чтобы триггер

переключался правильно, уровень на входе D следует зафиксировать заранее, перед приходом тактового перепада. Если на входы S и R триггера ТМ2 одновременно подаются напряжения низкого уровня, состояние выходов  $\bar{Q}$  и Q окажется неопределенным. Загрузить в триггер входные уровни 1. или 0 можно, если на входе S и R подать напряжение высокого уровня.

Таблица 2.3.

Состояние триггера из микросхемы ТМ2

Режим работы	Вход				Выход	
	S	R	C	D	Q	$\bar{Q}$
Асинхронная установка	0	1	*	*	1	0
Асинхронный сброс	1	0	*	*	0	1
Неопределенность	0	0	*	*	1	1
Загрузка 1(установка)	1	1	↑	1	1	0
Загрузка 0(сброс)	1	1	↑	0	0	1

\* -безразличное состояние

Асинхронная установка нужного сочетания уровней на выходах получается, когда на входах R и S поданы взаимно противоположные логические сигналы. В это время входы C и D отключены.

### МИКРОСХЕМА К155ТВ15.

Таблица 2.4.

Состояние JK -триггера (К155ТВ15)

Режим работы	Вход					Выход	
	S	R	C	J	$\bar{K}$	Q	$\bar{Q}$
Асинхронная	0	1	*	*	*	1	0
Асинхронный	1	0	*	*	*	0	1
Неопределенность	0	0	*	*	*	1	1
Переключение	1	1	↑	1	0	$\bar{q}$	q
Загрузка 0(сброс)	1	1	↑	0	0	0	1
Загрузка	1	1	↑	1	1	1	0
Хранение	1	1	↑	0	1	q	$\bar{q}$

Микросхема К155ТВ15 состоит из двух независимых JK - триггеров, которые запускаются положительным перепадом тактового импульса.

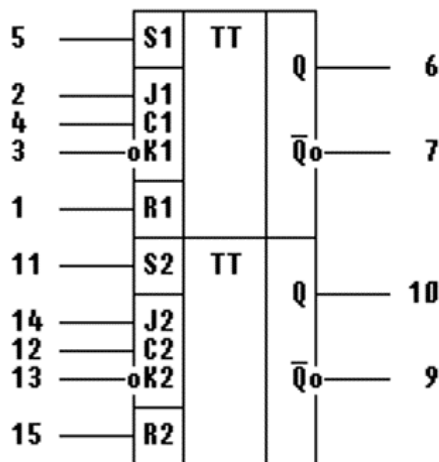


Рис. 2.9. Микросхема К155ТВ15.

Каждый триггер имеет независимые асинхронные входы установки S и сброса R. Как и у других триггеров ТТЛ, если на одном входе (или на обоих) R и S присутствует напряжение низкого уровня, то прием сигналов по входам C, J и K запрещается, а выходные сигналы  $\bar{Q}$  и Q устанавливаются на высокий или низкий уровни согласно табл.2. Структурная схема одного триггера из микросхемы К155ТВ15 показана на рис.5.

Выбранная полярность логических уровней для входов J и K позволяет превратить этот триггер в D, соединив входы J и K (рис.6). Сигналы на входах J и K следует зафиксировать перед приходом положительного перепада тактового импульса на вход C. Ток потребления для микросхемы К155ТВ15 30mA, максимальная частота переключения 25 МГц.

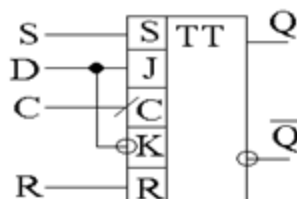


Рис. 2.10. D – триггер на основе JK – триггеров.



## 2.2. Регистры

Назначение регистров – хранение и преобразование много разрядных чисел. При простоте схемы регистры обладают большими функциональными возможностями. Они используются в качестве управляющих и запоминающих устройств, генераторов и преобразователей кодов, счетчиков, делителей частоты, узлов временной задержки.

Элементами структуры регистров являются синхронные триггеры D- типа либо RS(JK)- типа с динамическим или статическим управлением. Как отмечалось, одиночный триггер может запоминать (регистровать) один разряд (бит) двоичной информации. Такой триггер можно считать одноразрядным регистром. Цифровая техника, как правило, оперирует с много разрядными числами. Поэтому в качестве регистров в зависимости от назначения используются цепочки триггеров по количеству разрядов числа. В схемы регистров входят также комбинационные элементы, роль которых в данном случае вспомогательная.

Занесение информации в регистр называют операцией ввода или записи. Выдача информации к внешним устройствам характеризует операцию вывода или считывания. Запись информации в регистр не требует его предварительного обнуления.

Все регистры в зависимости от функциональных свойств подразделяются на две категории: накопительные (регистры памяти, хранения) и сдвигающие. В свою очередь, сдвигающие регистры делятся:

- по способу ввода и вывода информации на параллельные, последовательные и комбинированные (параллельно-последовательные и последовательно-параллельные);
- по направлению передачи (сдвига) информации на однонаправленные и реверсные.

## 2.2.1 Регистры памяти

Регистры памяти, простейший вид регистров. Их назначение, хранить двоичную информацию небольшого объема в течение короткого промежутка времени. Эти регистры представляют собой набор синхронных триггеров, каждый из которых хранит один разряд двоичного числа. Ввод (запись) и вывод (считывание) информации производится одновременно во всех разрядах параллельным кодом. Ввод обеспечивается тактовым командным импульсом. С приходом очередного тактового импульса происходит обновление записанной информации.

Сигнал на выходах триггеров характеризуют выходную информацию. Считывание может производиться, в прямом или обратном коде (в последнем случае с инверсных выходов).

Регистры хранения представляют собой, по существу, набор триггеров с независимыми информационными входами и общим тактовым входом. В качестве регистров подобного рода могут быть использованы без дополнительных элементов многие типы триггеров. Особенно пригодны микросхемы, содержащие в одном корпусе несколько самостоятельных триггеров, например К155ТМ5, К155ТМ7, К155ТМ8 и другие, которые можно рассматривать, как четырехразрядные регистры памяти.

Наращивание разрядности регистров памяти достигается добавлением нужного числа триггеров, тактовые входы которых присоединяются к общей шине синхронизации (рис.2.11).

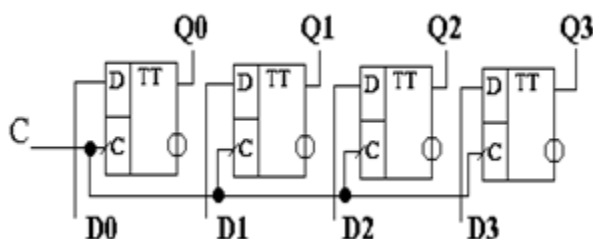


Рис. 2.11. Четырех разрядный параллельный регистр.

## 2.2.2 Регистры сдвига

Основную массу регистров, используемых на практике, представляют регистры сдвига. Это вид регистров отличаются большим разнообразием как в функциональном отношении, так и в отношении схемных решений, временных параметров, емкости и т. п.

Регистры сдвига, помимо операции хранения, осуществляют преобразование последовательного двоичного кода в параллельный, а параллельного в последовательный, выполняет арифметические и логические операции, служит в качестве цифровых элементов временной задержки.

Своим названием они обязаны характерной для этих устройств операции сдвига. Сущность сдвига состоит в том, что с приходом каждого тактового импульса происходит перезапись (сдвиг) содержимого триггера каждого разряда в соседний разряд без изменения порядка следования единиц и нулей.

При сдвиге информации вправо после каждого тактового импульса бит из более старшего разряда сдвигается в младший, а при сдвиге влево, наоборот.

На схемах символом регистра служат буквы RG. Для регистров сдвига указывается также направление сдвига; "→"- вправо; "←" – влево; "↔"- реверсивный (двунаправленный).

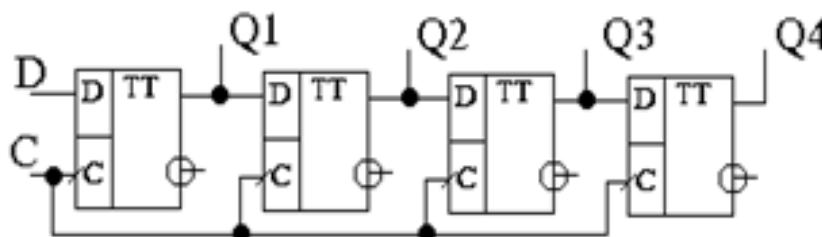


Рис. 2.12. Четырехразрядный последовательный регистр.

На рисунке 2.12. приведена схема четырехразрядного регистра с последовательным вводом входной информации и сдвигом ее вправо. Для наглядности показано, что в регистре применены RS(JK)-триггеры, а первый из них при помощи инвертора преобразован в D- триггер.

Допустим, что в регистр последовательно вводится, начиная с

младшего разряда, двоичный код 1101, который поступает от внешнего устройства синхронно с тактовыми импульсами. С первым тактовым импульсом в триггер DD1 будет записана единица младшего разряда. Со следующим тактовым импульсом эта единица будет сдвинута в триггер DD2 и окажется на его выходе. Одновременно в первый триггер поступит нуль (следующий разряд кода). Таким же образом будет происходить сдвиги с выхода Q2 на вход DD3 и с Q3 в DD4. После четырех тактовых импульсов код на выходах Q4-Q1 будет соответствовать входному коду и может быть считан внешним устройством. Таким образом, регистр преобразует последовательный код в параллельный. После очередного тактового импульса (в данном случае- пятого) информационный сигнал, бывший на выходе последнего триггера, выводится из регистра и пропадает. На выходе Q4 каждый сигнал появляется через четыре такта, считая с момента подачи его на вход. Это свойство регистра сдвига часто используется для задержки цифровой информации на заданное число тактовых периодов. Разрядность регистра сдвига, подобно показанному, может быть увеличена подключением дополнительных триггеров.

### 2.2.3. Интегральные регистры

Регистры сдвига как готовые изделия производятся в нескольких сериях микросхем ТТЛ. Мы рассмотрим наиболее распространенный тип: К155ИР1.

МИКРОСХЕМА К155ИР1 представляет собой четырехразрядный регистр сдвига с последовательным или параллельным вводом информации и параллельным выводом ее. Микросхема может использоваться в качестве буферной памяти, элемента задержки на несколько тактов, преобразования последовательного кода в параллельный и наоборот, делителя частоты, кольцевого распределителя импульсов, элемента арифметических устройств и т.п.

Регистр может выполнять следующие операции: ввод информации параллельным кодом; сдвиг информации вправо; ввод информации последовательным кодом; ввод последовательным кодом со сдвигом влево; хранение.

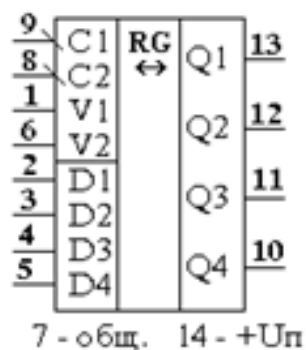


Рис. 2.13. Условное изображение регистра К155ИР1.

Регистр имеет два тактовых входа С1. и С2, управляющий вход выбора режима V2, пять информационных входов (V1 для ввода информации в последовательном коде и четыре входа D1-D4 для записи информации в параллельном коде, а также четыре выхода Q1-Q4 с каждого разряда регистра.

Наличие двух тактирующих входов допускают синхронизацию от различных генераторов при работе в режиме "сдвиг вправо" и "параллельный ввод". Если в обоих режимах синхронизация осуществляется от общего источника, тактовые импульсы можно подавать на оба тактовых входа С1 и С2 одновременно. На информационных входах триггеров сигналы должны обновляться до прихода фронта тактового импульса.

Триггеры, образующие регистр, – двухступенчатые, срабатывание их происходит по перепаду 1,0 входных импульсов, поступающих на один из тактовых входов С1 или С2. Рабочий режим регистра определяется уровнем сигнала на входе V2.

Ввод информации последовательным кодом, а также сдвиг ее вправо производится при V2=0. Входная информация подается на вход V1, а тактовые импульсы- на вход С1. Сдвиг вправо на один разряд происходит при каждом перепаде 1,0 тактовых импульсов. Информация в последовательном коде преобразуется в параллельный и после четырех тактовых импульсов может быть считана с выходов Q1-Q4.

Ввод информации параллельным кодом осуществляется при V2=0. Разрешающим входом служит С2. Запись в триггеры регистра информации со входов D1-D4 происходит по перепаду 1,0

разрешающего импульса. Входы V1 и C1 при этом заблокированы, и их состояние не играет роли.

В этом же режиме на входах V2 и C2 производится преобразование последовательного кода в параллельный со сдвигом влево. В этом случае поток информации имеет обратное направление: от четвертого триггера к третьему, от третьего ко второму и т.д., для чего необходимо произвести внешние соединения выходов Q4,Q3,Q2 с входами D3,D2,D1 соответственно. Информация в последовательном коде в регистр через вход D4. Сдвиг влево на один разряд происходит при каждом перепаде 1,0 тактовых импульсов, подаваемых на вход C2.

Состояние входов и выходов регистра K155ИР1 при работе в разных режимах приведены в таблице 2.5.

Таблица 2.5.

Состояние входов и выходов регистра K155ИР1.

Состояние входов			Режим
V1	C1	C2	
1	*	↓	Запись параллельным кодом, сдвиг влево
0	↓	*	Запись последовательным кодом, сдвиг вправо

Во избежание сбоев смена состояний входа V2 должна происходить только при C1=C2=0. Однако изменение V2 от 1 до 0, когда C2=0, и от 0 к 1, когда C1=0, не вызовут изменений на выходах регистра.

Последовательным соединением n микросхем можно получить 4n-разрядный регистр с преобразованием параллельного кода в последовательный и наоборот.

### 2.3. Счетчики

Счетчик – это устройство, которое служит для подсчета количества импульсов, поступающих на его вход.

Наибольшее распространение в устройствах ВТ получили двоичные счетчики, состоящие из триггеров со счетным входом.

Емкость счетчика характеризуется максимально возможным числом  $N$  двоичных сигналов, которое может быть зафиксировано им. Триггеры, входящие в счетчик, образуют его разряды. Между емкостью счетчика и числом разрядов, содержащихся в нем, существует соотношение  $N=2^n-1$ , где  $n$ - число разрядов счетчика.

По своему назначению счетчики подразделяют на суммирующие, вычитающие и реверсивные. В суммирующих счетчиках производится сложение поступающих на вход импульсов с тем числом, которое хранилось в счетчике. Вычитающие счетчики производят вычитание поступивших на вход импульсов из начального числа. Реверсивные счетчики могут производить как сложение, так и вычитание поступающих на вход импульсов в зависимости от управляющего сигнала, который переключает счетчик (с помощью специальной схемы переключения) либо в режим сложения, либо в режим вычитания.

В суммирующих счетчиках счетный вход каждого последующего триггера соединен с выходом предыдущего таким образом, что при переходе триггера младшего разряда из состояния 1 в состояние 0 в цепи переноса между триггерами появляется сигнал переноса, под действием которого триггер старшего разряда изменяет свое состояние на противоположное.

Действие четырехразрядного суммирующего двоичного счетчика на JK-триггерах (рис. 2.14а) поясняют временные диаграммы (рис. 2.14б), из которых следует, что до прихода первого импульса все триггеры находились в нулевом состоянии.

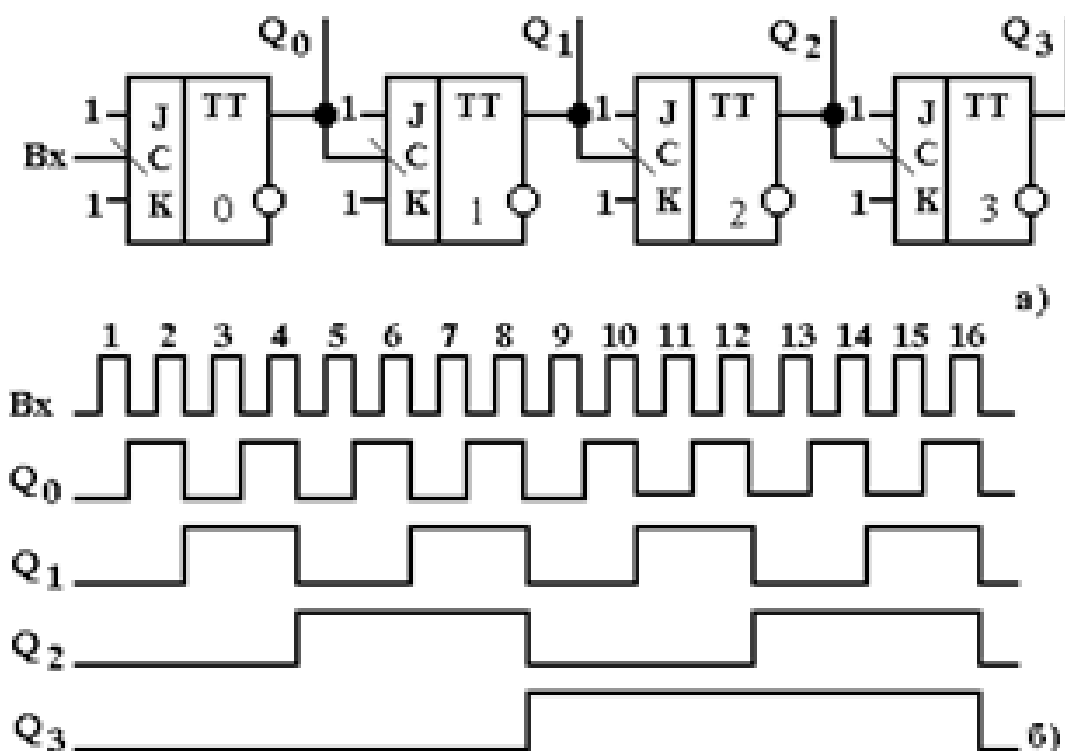


Рис. 2.14. Четырехразрядный суммирующий двоичный счетчик (а) на JK-триггерах и временные диаграммы (б) его работы.

Срез первого импульса переключает нулевой триггер в единичное состояние. Затем срез второго импульса переключает его в нулевое состояние и т. д. Триггер под номером 1 переключается срезом импульса с выхода триггера 0 и т. д. Из временных диаграмм видно, что частота следования импульсов каждым триггером делится на 2.

После прихода шестнадцати импульсов все четыре триггера находятся в таком же состоянии, как и до прихода первого импульса. Наблюдая состояние выходов  $Q_0$ ,  $Q_1$ ,  $Q_2$ ,  $Q_3$  можно судить о том, сколько пришло импульсов. Например, при нуле импульсов  $Q_0=Q_1=Q_2=Q_3=0$ . После прихода пятнадцати импульсов  $Q_0=Q_1=Q_2=Q_3=1$ . В первом случае это соответствует записи числа 0 в виде 0000, а во втором – записи числа 15 в виде 1111. После прихода восьми импульсов  $Q_0=Q_1=Q_2=0$  и  $Q_3=1$ . Следовательно, числа записываются в обратном порядке – последний разряд является высшим. В интегральном исполнении выпускаются 4-, 8- и 12-разрядные счетчики. Счетчики одновременно являются и делителями частоты в  $2^n$  раз, где  $n$  – число разрядов.



Описанный выше счетчик называется асинхронным или последовательным. В нем каждый последующий каскад считает после предыдущего.

В вычитающих счетчиках при переходе триггера младшего разряда из состояния 0 в состояние 1 в цепи переноса появляется сигнал займа, переводящий триггер старшего разряда в противоположное состояние.

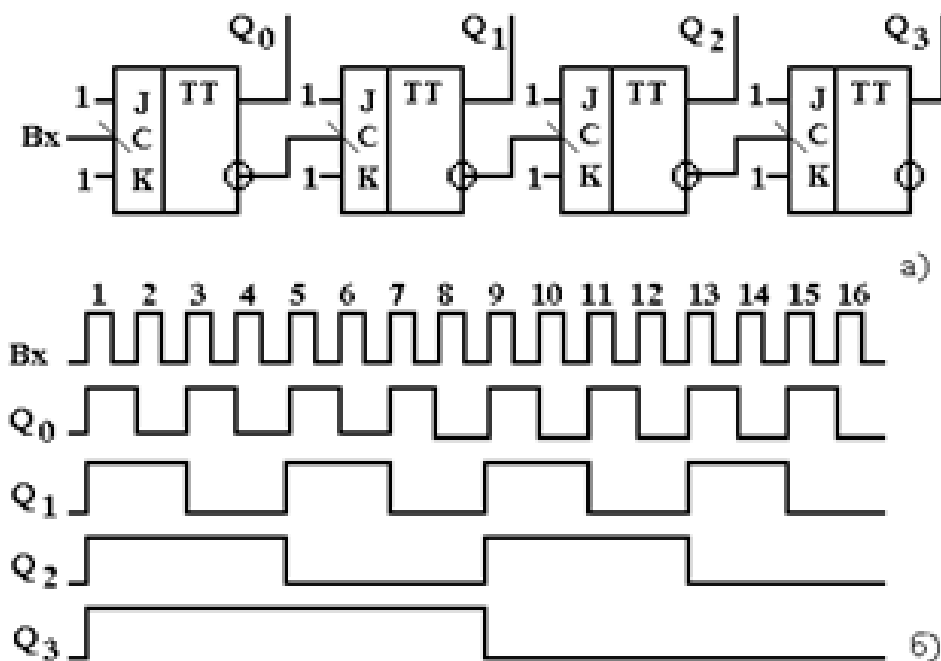


Рис. 2.15. Четырехразрядный вычитающий двоичный счетчик (а) на JK-триггерах и временные диаграммы (б) его работы.

В реверсивных счетчиках с помощью логических элементов осуществляется переключение связей между триггерами, поэтому результат в счетчике может как увеличиваться, так и уменьшаться на единицу с приходом входного импульса.

### 2.3.1. Счетчики в интегральном исполнении

Микросхемы К155ИЕ6 и К155ИЕ7 – четырехразрядные быстродействующие реверсивные счетчики-делители с параллельным переносом между разрядами. Различие между обоими типами- только в модуле счета: для К155ИЕ6 Ксч=10, а для К155ИЕ7 Ксч=16.

Условные изображения обоих счетчиков приведены на рис.2.16.

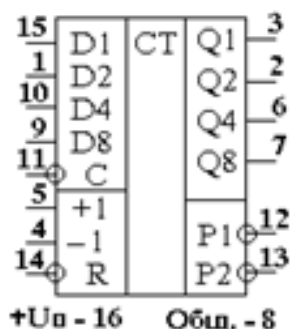


Рис. 2.16. Четырехразрядный двоичный реверсивный счетчик K155IE6(7)

Входы +1. и -1 -счетные. Последовательность входных импульсов подается на один из этих входов в зависимости от того, в каком направлении (прямом или обратном) требуется вести счет. Входы D1,D2,D4,D8 предназначены для ввода в счетчик исходного числа, с которым суммируются счетные импульсы (в режиме сложения) либо из которого они вычитаются (режим вычитания). Ввод данных происходит с момента появления импульса записи низкого уровня на входе С. Вход " сброс " служит для установки нуля на всех выходах. Этот выход обладает приоритетом над остальными.

Выходы Q1,Q2,Q4,Q8- прямые выходы разрядов счетчиков. Два других выхода- переноса (вывод 12) и займа (вывод 13)- используются при каскадировании микросхем, при работе в качестве делителя, а также при циклической записи в счетчик информации со D1,D2,D4,D8.

Таблица 2.6 характеризует поведение счетчиков в разных режимах. В режиме суммирования сигнал переноса возникает на выходе P1 во время перехода из состояния 1111H= 15D для K155IE7 (1001H= 9D для K155IE6) в состояние 0000, а в режиме вычитания сигнал займа возникает на выходе P2 при изменении состояния 0000 на 1111 (для K155IE7) либо на 1001 (для K155IE6). Сигнал переноса (займа) формируется срезом соответствующего счетного импульса и длится, пока на соответствующем входе существует уровень 0, т.е. состояние выхода переноса (займа) повторяет состояние входа переноса (займа) повторяет состояние входа (с задержкой сигнала во внутренних элементах).

Таблица 2.6.  
Состояние микросхем К155ИЕ6 и К155ИЕ7

Входы								Выходы						Режим работы
R	C	+1	-1	D 1	D 2	D 3	D 4	Q1	Q2	Q3	Q4	P1	P2	
1	×	×	0	×	×	×	×	0	0	0	0	1	0	Установка нуля
1	×	×	1	×	×	×	×	0	0	0	0	1	1	
0	0	×	0	0	0	0	0	0	0	0	0	1	0	Предварительная (параллельная) установка
0	0	×	1	0	0	0	0	0	0	0	0	1	1	
0	0	0	×	1	×	×	1	Q <sub>i</sub> =D <sub>i</sub>				0	1	
0	0	1	×	1	×	×	1					1	1	
0	1	↑	1	×	×	×	×	Прямой счет				1*	1	Суммирование
0	1	1	↑	×	×	×	×	Обратный счет				1	1 **	Вычитание

\*- на выходе P1 при выходной комбинации Q1,Q2,Q4,Q8=1001 (для К155ИЕ6) и Q1,Q2,Q4,Q8=1111 (для К155ИЕ7) в режиме суммирования Uвых соответствует состоянию выхода +1.

\*\* - на выходе P2 при выходной комбинации 0000 в режиме вычитания Uвых соответствует состоянию входа -1.

Импульсы переноса (займа) можно использовать для циклической записи в счетчик информации с входа D1, D2, D4, D8. Для этого достаточно соединить вход C с соответствующим выходом.

Если счетчик используется в качестве делителя частоты, то при соединении входа C с выходом переноса P1 коэффициент деления будет 15-M (или 9-M), где M – десятичный эквивалент двоичного кода на входах D1,D2,D4,D8. Максимальный модуль счета (Kсч=16 для К155ИЕ7 и Kсч=10 для К155ИЕ6) обеспечивается без указанной обратной связи.

При прямом счете на входе обратного счета должно быть напряжение высокого уровня, а при обратном счете высокое напряжение должно быть на входе прямого счета. В случае работы с инвертированием во избежание сбоев перед каждым переключением счета следует прервать путем установки на входах +1 и -1- уровня 1.

### 3. АРИФМЕТИЧЕСКИЕ УСТРОЙСТВА

Комбинационные устройства, которые рассматривались ранее, выполняют логические функции. Для описания их поведения используется аппарат алгебры логики. Входные и выходные сигналы оцениваются логическими 1 и 0.

Дискретная техника имеет дело и с другим классом приборов, назначение которых состоит в выполнении арифметических действий с двоичными числами: сложение, вычитание, умножение, деление. К арифметическим устройствам относят также узлы, выполняющие специальные арифметические операции: выявление четности чисел и сравнение двух чисел.

В этих устройствах сигналам приписываются не логические 0 и 1, а арифметические значения 0 и 1. Действия над ними подчиняются законам двоичной арифметики. Для описания работы арифметического устройства также удобно использовать таблицы истинности, как и для логических устройств. Арифметические устройства широко используются в ЭВМ и в аппаратуре информационно-измерительной техники.

Важнейшая из арифметических операций – сложение ( $\Sigma$ ). Помимо прямого назначения сложение используется и при других операциях: вычитание – это сложение, в котором вычитаемое вводится в обратном или дополнительном коде, а умножение и деление – это последовательное сложение или вычитание.

Сумматоры представляют собой функциональные цифровые устройства, выполняющие операцию сложения чисел. В цифровой технике суммирование осуществляется в двоичном или, реже, в двоично-десятичном коде.

По характеру действия сумматоры подразделяются на комбинационные и накопительные; по количеству одновременно обрабатываемых разрядов складываемых чисел сумматоры делятся на одноразрядные и многоразрядные.

По числу входов и выходов одноразрядные двоичные сумматоры подразделяются на:

– полусумматоры, характеризующиеся наличием двух входов, на которые подаются одноименные разряды двух чисел, и двух выходов: на одном реализуется арифметическая сумма в данном разряде, а на другом – перенос в следующий (более старший разряд);

– полные одноразрядные двоичные сумматоры, характеризующиеся наличием трех входов, на которые подаются одноименные разряды двух складываемых чисел и перенос из предыдущего (более младшего) разряда, и двумя выходами: на одном реализуется арифметическая сумма в данном разряде, а на другом – перенос в следующий (более старший разряд).

По способу представления и обработки складываемых чисел многоразрядные двоичные сумматоры подразделяются на:

– последовательные, в которых обработка разрядов чисел ведется поочередно, разряд за разрядом на одном и том же оборудовании;

– параллельные, в которых слагаемые складываются одновременно по всем разрядам и для каждого разряда имеется свое оборудование.

### 3.1. Сумматоры параллельного действия

**Полусумматор** (рис. 3.1.) имеет два входа  $a$  и  $b$  для двух слагаемых и два выхода:  $S$  — сумма,  $P$  — перенос. Обозначением полусумматора служат буквы HS (half sum — полусумма). Работу его отражает таблица истинности 2 (табл. 3.1), а соответствующие уравнения имеют вид:

$$\begin{cases} S = a \oplus b \\ P = ab \end{cases} \quad (1)$$

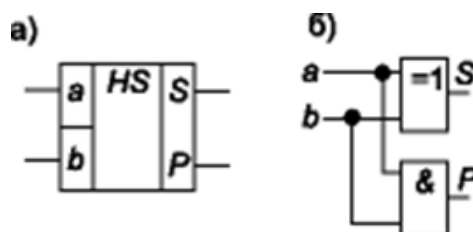


Рис. 3.1. Стандартное изображение двоичного полусумматора.

Полусумматор, реализованный на элементах “ИСКЛЮЧАЮЩЕЕ ИЛИ” и “И”.

Таблица 3.1.

Таблица истинности полусумматора.

a	b	P	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Из уравнений (1) следует, что для реализации полусумматора требуется один элемент “исключающее ИЛИ” и один двухвходовый вентиль И (рис. 3.1.б).

**Полный одноразрядный двоичный сумматор** (рис. 3.2) имеет три входа:  $a$ ,  $b$  — для двух слагаемых и  $p$  — для переноса из предыдущего (более младшего) разряда и два выхода:  $S$  — сумма,  $P$  — перенос в следующий (более старший) разряд. Обозначением полного двоичного сумматора служат буквы  $SM$ . Работу его отражает таблица истинности 3 (табл. 3.2).

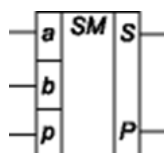


Рис. 3.2. Полный одноразрядный двоичный сумматор.

Таблица 3.2.

Таблица истинности полного сумматора.

<b>p</b>	<b>a</b>	<b>b</b>	<b>P</b>	<b>S</b>
0	0	0	0	0
1	0	0	0	1
0	0	1	0	1
1	0	1	1	0
0	1	0	0	1
1	1	0	1	0
0	1	1	1	0
1	1	1	1	1

Полный двоичный сумматор может быть реализован на двух полусумматорах и одном двухвходовом элементе ИЛИ. Соответствующая схема приведена на рис. 3.3.

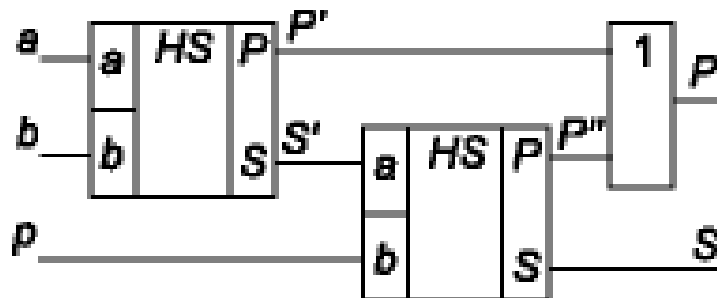


Рис. 3.3. Полный одноразрядный двоичный сумматор.

### 3.2. Последовательный сумматор

Сложение многоразрядных чисел можно реализовать последовательно либо параллельно. При последовательном суммировании требуется одноразрядный полный сумматор, на вход которого в течение тактового интервала последовательно, начиная с младшего разряда, подаются соответствующие разряды слагаемых и результат переноса от сложения на предыдущем такте (рис. 3.4.). Результат суммирования поразрядно с выхода сумматора передается в линию связи либо запоминается в буферном сдвигающем регистре суммы.

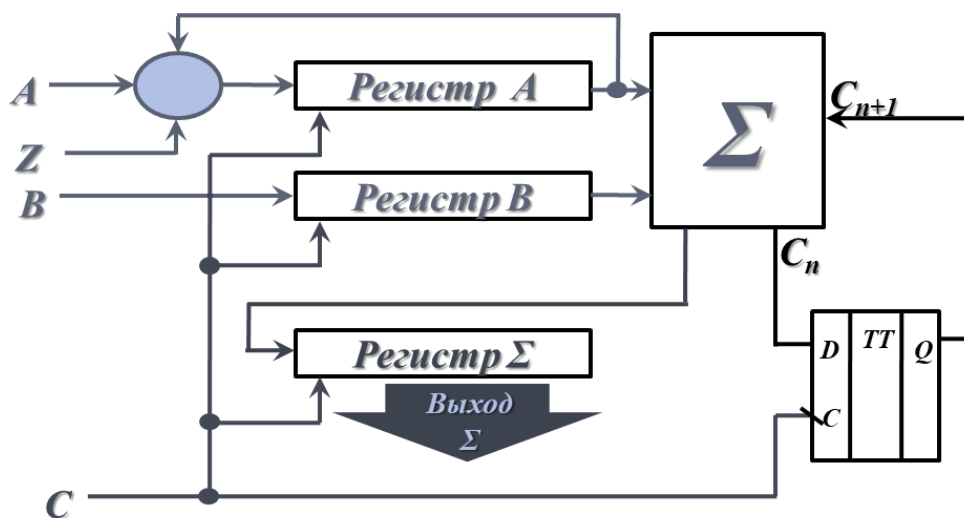


Рис. 3.4. Последовательный сумматор.

Операция суммирования заканчивается через количество тактов суммирования  $(N+1)$ , где  $N$  – количество разрядов слагаемых чисел  $A$  и  $B$ . Дополнительный такт необходим для реализации переноса от суммирования старших разрядов. При этом общее время суммирования будет составлять:  $T_{\text{общ}} = t_{z\Sigma} (N+1)$ , где  $t_{z\Sigma}$  – задержка сумматора  $\Sigma$ . Последовательный сумматор требует минимальных затрат оборудования, не зависящих от разрядности суммируемых чисел. Однако длительность операции суммирования пропорциональна разрядности операндов, поэтому последовательный сумматор может использоваться в относительно медленнодействующих цифровых устройствах.

### 3.3. Арифметико-логические устройства

Интересными свойствами обладают специализированные микросхемы, выполняющие в соответствии с программой на входах арифметические и логические преобразования двоичной информации. Эти микросхемы так и называют – арифметико-логическими устройствами (АЛУ). По сравнению с приборами, работающими по жесткой, наперед заданной программе, АЛУ представляют собой устройства более высокого класса. В микропроцессорной технике АЛУ являются базовыми элементами. Они используются в сочетании с регистрами сдвига, оперативными запоминающими устройствами и



другими узлами.

АЛУ дороже простых микросхем, однако благодаря универсальным свойствам, применение их в аппаратуре во многих случаях оказывается оправданным. Микросхемы АЛУ, принадлежащие к разным видам логики, в частности к ТТЛ – типа К155ИПЗ и КНОП – типа 564ИПЗ функционально во многом совпадают, в том числе и по разводке выводов, поэтому описание будет идти применительно типу – К155ИПЗ.

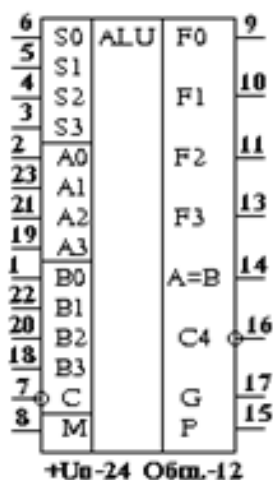


Рис. 3.5. Микросхема К155ИПЗ.

Микросхема К155ИПЗ предназначена для действий с двумя четырехразрядными двоичными словами:  $A=A_3A_2A_1A_0$  и  $B=B_3B_2B_1B_0$  (рис.3.5.). Конкретный вид операции, выполняемой микросхемой, задается 5-разрядным кодом на входах MS3S2S1S0. Всегда это АЛУ способно выполнить 32 операции; 16 логических (И, И-НЕ, ИЛИ, ИЛИ-НЕ и др.) и 16 арифметических и арифметико-логических (сложение, вычитание, удвоение, сравнение чисел и ряд иных). Операции сложения и вычитания проводятся с ускоренным переносом из разряда в разряд. Кроме того, имеется вход приема сигнала переноса С.

На выходах F0.F1,F2 и F3 формируется результаты логических преобразований и арифметических действий. На выходе переноса С4 образуется сигнал старшего (пятого) разряда при выполнении арифметических операций. Дополнительные выходы – образования

ускоренного переноса  $C$  и распространения ускоренного переноса  $P$  – используются только при организации много разрядного переноса К155ИП4.

Слова  $A$  и  $B$ , подлежащие обработке, могут быть представлены в положительной либо в отрицательной логике. Таблицы истинности для каждого варианта логики различны (табл.21). Во избежание путаницы уровни сигналов обозначены в них буквенными символами. Результаты арифметических операций выражены в дополнительном коде. Числа в дополнительном и обратном коде связаны простым соотношением  $N_{доп.} = N_{обр.} + 1$  или  $N_{обр.} = N_{доп.} - 1$ . Поэтому в тех строках таблицы -, где указана операция "минус 1-", результат арифметических действий представлен в обратном коде.

Старший разряд кода выбора операции (вход  $M$ ) определяет характер действий, выполняемых АЛУ. Когда на этом входе сигнал высокого уровня, АЛУ производит логические операции поразрядно над каждой парой бит слов  $A$  и  $B$ . Внутренний перенос в этом режиме бездействует.

Арифметические операции выполняются, когда на входе  $M$  установлен низкий потенциал, который является также разрешающим сигналом для переноса между разрядами. Выходной результат формируется с учетом состояния входа переноса. Оба сигнала переноса – входной  $C$  и выходной  $C_4$  – инверсные относительно сигналов на входах  $A$  и  $B$ , т.е. когда слова  $A$  и  $B$  – в положительной логике, сигналу переноса отвечает низкий уровень напряжения на соответствующем выводе, а в отрицательной логике – наоборот. Если АЛУ выполняет логико-арифметическую операцию, логическая функция реализуется поразрядно, а арифметическая с переносом. Например, входному коду  $MS_3S_2S_1S_0=01101$  отвечает операция  $AB$  плюс  $A$  ( третья снизу строка табл. 21, отрицательная логика ), где  $AB$  -логическое умножение двух слов. Если  $A=1010$  и  $B=0111$ , то первая операция дает  $AB=0010$  и, следовательно,  $0010$  плюс  $1010=1100$ .

При использовании АЛУ в качестве компаратора сигнал снимают с выхода  $A=B$  (вывод 14). Этот выход – с открытым коллектором, и к источнику питания его следует подключать через внешний резистор 1

ком.

Режим компаратора обеспечивается при  $M=0$  и  $S_3S_2S_1S_0=0110$ . Когда числа  $A$  и  $B$  равны, на выходе  $A=B$  формируется сигнал высокого уровня. Одновременно сигнал на выходе  $C_4$  (вывод 16) характеризует соотношение между числами  $A$  и  $B$  и в случае их неравенства согласно табл. 3.3.

Таблица 3.3.

Таблица истинности МС К155ИПЗ в режиме четырехразрядного компаратора.

Вид логики	Состояние входов		Состояние выхода $C_4$
	$C$	$A$ и $B$	
Положительная	1	$A \leq B$	1
	0	$A < B$	1
	1	$A > B$	0
	0	$A \geq B$	0
Отрицательная	0	$A \leq B$	0
	1	$A < B$	0
	0	$A > B$	1
	1	$A \geq B$	1

Таблица 3.4.

Функциональная зависимость выходов МС К155ИПЗ от состояния  
ВХОДОВ

Входы-выходы функции				Входы-выходы (отрицательная логика)		Входы-выходы (положительная логика)	
S3	S2	S1	S0	логическая функция (M=1)	Арифметическое действие (M=0;C=0)	логическая функция (M=1)	Арифметическое действие (M=0;C=1)
0	0	0	0	$\bar{A}$	$A - 1$	$\bar{A}$	$A$
0	0	0	1	$\bar{A}\bar{B}$	$AB - 1$	$\overline{A \vee B}$	$A \vee B$
0	0	1	0	$\bar{A} \vee B$	$A\bar{B} - 1$	$\bar{A}B$	$A \vee \bar{B}$
0	0	1	1	Логическая 1	Минус 1	Логический 0	Минус 1
0	1	0	0	$\overline{A \vee B}$	$A + (A \vee \bar{B})$	$\bar{A}\bar{B}$	$A + A\bar{B}$
0	1	0	1	$\bar{B}$	$AB + (A \vee \bar{B})$	$\bar{B}$	$A\bar{B} + (A \vee B)$
0	1	1	0	$\overline{A \oplus B}$	$A - B - 1$	$A \oplus B$	$A - B - 1$
0	1	1	1	$A \vee \bar{B}$	$A \vee \bar{B}$	$A\bar{B}$	$A\bar{B} - 1$
1	0	0	0	$\bar{A}\bar{B}$	$A + (A \vee B)$	$\bar{A} \vee B$	$AB + A$
1	0	0	1	$A \oplus B$	$A + B$	$\overline{A \oplus B}$	$A + B$
1	0	1	0	$B$	$A\bar{B} + (A \vee B)$	$B$	$AB + (A \vee \bar{B})$
1	0	1	1	$A \vee B$	$A \vee B$	$AB$	$AB - 1$
1	1	0	0	Логический 0	$(A + A)^*$	Логическая 1	$(A + A)^*$
1	1	0	1	$A\bar{B}$	$AB + A$	$A \vee \bar{B}$	$A + (A \vee B)$
1	1	1	0	$AB$	$A\bar{B} + A$	$A \vee B$	$A + (A \vee \bar{B})$
1	1	1	1	$A$	$A$	$A$	$A - 1$

\*- Равнозначно тому, что каждый разряд сдвинут в направлении более высокого разряда.

### 3.4. Мультиплексы

#### *Основные положения*

Назначение мультиплексов (от англ. multiplex- многократный)- коммутировать в желаемом порядке информацию, поступающую с нескольких входных шин на одну выходную. С помощью мультиплексора осуществляется временное разделение информации, поступающей по разным каналам. Мультиплексор можно уподобить бесконтактному многопозиционному переключателю.

Мультиплексы обладают двумя группами входов и одним, реже двумя- взаимодополняющими выходами. Одни входы информационные, а другие служат для управления. К ним относятся адресные и разрешающие (стробирующие) входы. Если мультиплексор имеет  $n$  адресных входов, то число информационных входов будет  $2^n$ . Набор сигналов на адресных входах определяет конкретный информационный вход, который будет соединен с выходным выводом.

Разрешающий (стробирующий) вход управляет одновременно всеми информационными входами независимо от состояния адресных входов. Запрещающий сигнал на этом входе блокирует действие всего устройства. Наличие разрешающего входа расширяет функциональные возможности мультиплексора, позволяя синхронизировать его работу с работой других узлов. Разрешающий вход употребляется также для наращивания разрядности мультиплексора.

#### *Схемотехнические решения*

На рисунке 3.6. показана принципиальная схема простейшего мультиплексора – селектора вида "две линии к одной" (2:1). Для переключения входных цепей используется один внешний сигнал. Когда  $A=1, F=x_1$ ; при  $A=0, P=x_2$ .

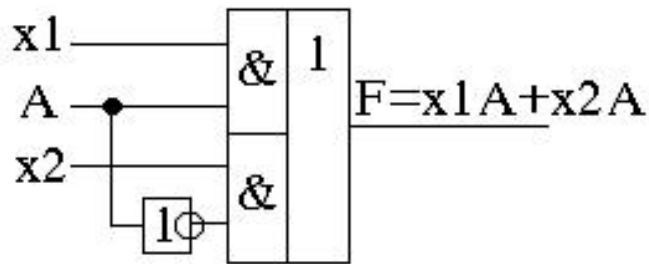


Рис. 3.6. Принципиальная схема мультиплексора – селектора вида 2:1.

Эти же принципы положены в основу построения и более сложных схем мультиплексоров. Рисунок 3.7. отражает логическую структуру реального мультиплексора "четыре линии к одной" (4:1) – половину микросхемы ТТЛ К155КП2.

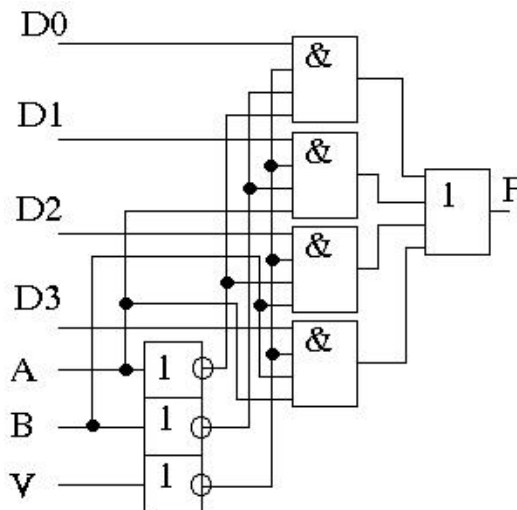


Рис. 3.7. Принципиальная схема мультиплексора – селектора вида 4:1.

Она содержит четыре информационных входа D0-D3, два адресных входа A и B и разрешающий вход V. Вход A принадлежит младшему разряду, B - старшему. Когда разрешающий вход находится под высоким потенциалом  $U_{вх}=1$  один из входов логических элементов 'И' будет под низким и, следовательно, на их выходах также будет нулевые уровни независимо от состояния остальных входов. Выходной сигнал в этом случае также будет  $F=0$ .

Схема управления выполнена так, что при разрешающем сигнале на входе V любые комбинации сигналов на адресных входах, A и B

создают условия, при которых на входах (а значит, и на выходах) трех логических элементов 'И' существуют потенциалы низкого уровня, неактивные для элемента 'ИЛИ'. Состояние четвертого элемента 'И' определяется сигналами на информационном входе, тот же сигнал будет и на выходе мультиплексора. Двоичные числа (00;01;10 и 11), характеризующие сигналы на входах В и А, эквивалентны индексу задействованного информационного входа (D0,D1,D2,D3). Так, например, двоичное число 10 на адресных входах обеспечит селекцию шины D2. Это следует и из таблицы истинности рассматриваемого мультиплексора (таблица 3.5.) и из формулы

Таблица 3.5.

Таблицы истинности мультиплексора 4:1.

Входы			Выход
V	A	B	F
0	0	0	D0
0	0	1	D1
0	1	0	D2
0	1	1	D3
1	*	*	0

Таблица 16

$$F = \bar{V} (\bar{B}\bar{A}D_0 + \bar{B}AD_1 + B\bar{A}D_2 + BAD_3)$$

### Способы наращивания

У мультиплексоров, выпускаемых в виде самостоятельных изделий, число информационных входов не превышает шестнадцати. Большее число входов обеспечивается путем наращивания. Наращивание можно выполнять двумя способами: объединением нескольких мультиплексоров в пирамидальную (древовидную) систему либо последовательным соединением разрешающих входов и внешних логических элементов. На практике применяют оба метода.

Пирамидальные мультиплексоры строятся по ступенчатому принципу, причем обычно применяются две, реже три и более ступени. Пирамидальный характер схемы состоит в том, что каждая ступень, начиная с первой, имеет больше входов, чем последующая. Младшие разряды кода адреса подаются на адресные входы первой

ступени, а ступеням более высокого ранга соответствуют старшие разряды адресного кода.

На рисунке 3.8. показан вариант мультиплексора 32:1 на основе приборов 8:1 и 4:1. Чередование сигналов в последовательности от 000 до 111 на адресных входах С, В, А микросхемы DD1.1-DD1.4 одновременно коммутируют входы  $x_0$ -  $x_7$  на шину  $y_0$ , входы  $x_8$ -  $x_{15}$  на шину  $y_1$ ;  $x_{16}$ -  $x_{23}$  на шину  $y_2$  и  $x_{24}$ -  $x_{31}$  на шину  $y_3$ . Адресу  $СВА=100$ , например, соответствует поступление сигналов с  $x_4$  на  $y_0$ ;  $x_{12}$  на  $y_1$ ; с  $x_{20}$  на  $y_2$  и с  $x_{28}$  на  $y_3$ .

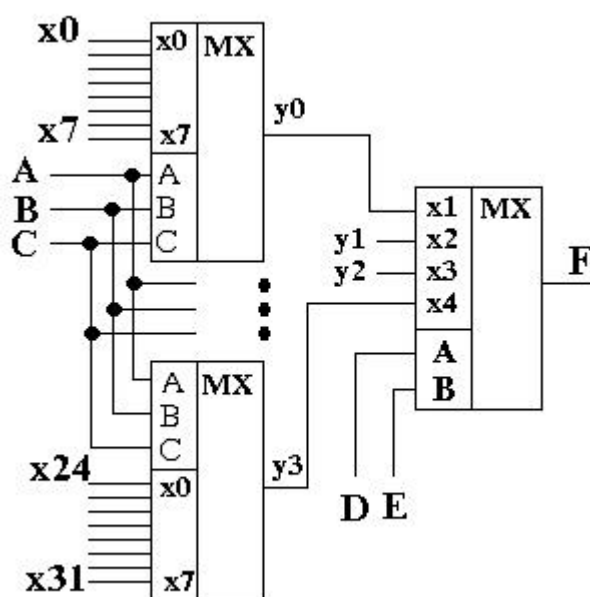


Рис. 3.8. Принципиальная схема мультиплексора – селектора вида 32:1.

В свою очередь, мультиплексирование шин  $y_0$ -  $y_3$  происходит при смене сигналов на адресных входах высших разрядов  $E, D$  (микросхема DD5) от 00 до 11. Если требуется, чтобы выход системы был связан с входом  $x_{20}$ , то должно быть  $ED=10$  (код шины  $y_2$ ). Следовательно, информационному входу  $x_{20}$  принадлежит адрес  $EDCBA=10100$ .

Общее число информационных входов при одноступенчатых мультиплексорах ранга равно произведению числа входов отдельных мультиплексоров и числа самих мультиплексоров:  $U_{вх.общ.} = nU_{вх.}$ . Для очень больших  $U_{вх.общ.}$  понадобится ряд мультиплексоров.



Порядок их подключения не требует пояснений. Общее число входов определяется по той же формуле.

Недостатками пирамидального наращивания следует считать повышенный расход микросхем, а также сравнительно невысокое быстродействие из-за суммирования задержек при последовательном прохождении сигналов по ступеням пирамиды.

На рисунке 3.9. показано, как получить мультиплексор 32:1 из двух 16:1 с использованием разрешающих входов микросхем в качестве адресных входов высшего разряда.

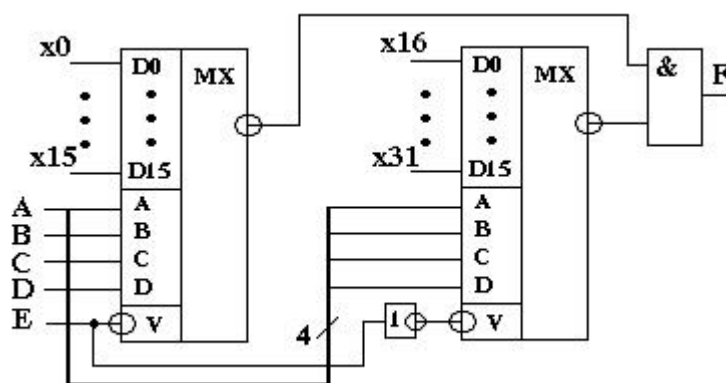


Рис. 3.9. Принципиальная схема мультиплексора – селектора вида 32:1.

По функциональным возможностям мультиплексоры являются очень гибкими устройствами и помимо прямого назначения могут выполнять и другие функции.

Мультиплексоры, в частности, используются для преобразования параллельного кода в последовательный.

*Мультиплексор как универсальный (функционально полный) логический элемент.*

Еще одно интересное свойство мультиплексоров – работа в качестве универсального логического элемента, реализующего любую логическую функцию, содержащую до  $n+1$  переменных, где  $n$  - число адресных входов мультиплексора. Применение этого свойства особенно оправданно, когда число переменных достаточно велико, 4-5 и более. Один мультиплексор в этом случае может заменить

несколько корпусов с логическими элементами вида И, ИЛИ, НЕ и др. Синтез таких схем довольно прост и осуществляется на основе словесного описания функции или по таблице истинности.

Использование мультиплексора в качестве универсального логического элемента основано на общем свойстве логических функций независимо от числа аргументов всегда равняется логической единице или нулю. Если на адресные входы мультиплексора подавать входные переменные, зная, какой выходной уровень должен отвечать каждому сочетанию этих сигналов, то предварительно установив на информационных входах потенциалы нуля и единицы согласно программе, получим устройство, реализующее требуемую функцию. На простом примере функции "исключающее ИЛИ" покажем, как с помощью мультиплексора 4:1, описанного выше, можно реализовать любую двоичную функцию двух переменных. Как следует из таблицы истинности для функции "исключающее ИЛИ" (рис.3.11.), сочетаниям  $x_2x_1=00$  и  $x_2x_1=11$  отвечает значение логического 0, а двум другим  $x_2x_1=01$  и  $x_2x_1=10$  – логической 1. Для выполнения этих условий достаточно подключить к адресным входам мультиплексора А и В шины сигналов  $x_1$  и  $x_2$  соответственно, на информационные входы D0 и D3 подать потенциал логического 0, а на D1 и D2 – логической 1. Разрешающий вход при этом должен быть в состоянии логического 0.

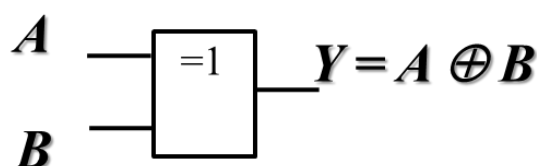


Рис. 3.10. Элемент, выполняющий функцию "исключающее ИЛИ".

Таблица истинности для функции "исключающее ИЛИ".

<i>B</i>	<i>A</i>	<i>Y</i>
0	0	0
0	1	1
1	0	1
1	1	0

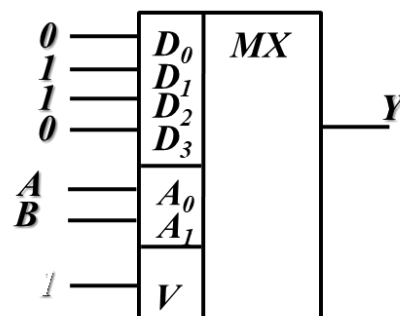


Рис. 3.11. Реализация функции "исключающее ИЛИ" с помощью мультиплексора 4:1.

### 3.5. Мультиплексоры ТТЛ

Мультиплексоры ТТЛ, выполненные в виде самостоятельных микросхем, строятся по образцу схемы, показанной на рис.30, и различаются главным образом числом информационных и адресных входов, наличием или отсутствием разрешающего входа, а также характером выходных сигналов (Относительно входных информационных), которые могут быть прямыми, инверсными или парными.

#### МИКРОСХЕМА К155КП2

Микросхема К155КП2 содержит в одном корпусе два мультиплексора вида 4:1. Каждая секция имеет четыре информационных входа  $D_0$ - $D_3$ , разрешающий вход  $V$ , а также прямой по отношению к информационным входам выход  $F$ . Два входа  $B$  и  $A$  - общие для обоих мультиплексоров (рис.3.12).

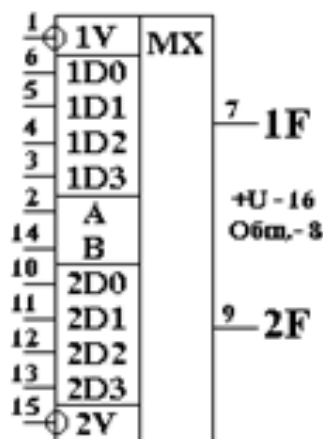


Рис. 3.12. Микросхема К155КП2

Логическая функция для каждой из секций микросхемы К155КП2 выглядит следующим образом:

$$F = \bar{V} (\bar{B} \bar{A} D 0 + \bar{B} A D 1 + B \bar{A} D 2 + B A D 3)$$

В зависимости от цифровой комбинации на адресных входах на выходы поступают сигналы от одноименных информационных входов. Нормальное функционирование происходит, когда потенциал разрешающего' входа  $V=0$ . При сигнале  $V=1$  на одном из разрешающих входов происходит блокирование соответствующей секции, на выходе которой устанавливается уровень  $U_{\text{вых}}=0$  независимо от состояния остальных входов. Работу обоих мультиплексоров характеризует табл.3.6. При желании микросхему можно использовать в качестве мультиплексора 8:1. Соединение обеих секций производится по тому же принципу (рис.3.13).

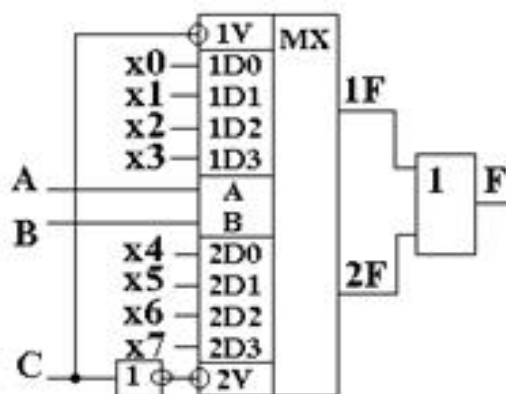


Рис. 3.13. Принципиальная схема мультиплексора – селектора вида 8:1.

Таблица 3.6.

Таблица истинности работы мультиплексора 4:1.

Входы							Выходы
V	B	A	D0	D1	D2	D3	
0	0	0	1/0	*	*	*	1/0
0	0	1	*	1/0	*	*	1/0
0	1	0	*	*	1/0	*	1/0
0	1	1	*	*	*	1/0	1/0
1	*	*	*	*	*	*	0

### 3.6. Демультимплексоры и дешифраторы

#### Основные положения

Демультимплексоры в функциональном отношении противоположны мультиплексорам. Здесь сигналы с одного информационного входа распределяются в желаемой последовательности по нескольким выходам. Выбор нужной выходной шины, как и в мультиплексоре, обеспечивается кодом на адресных входах. При  $m$  адресных входах де мультиплексор может иметь в зависимости от конструкции до  $2^m$  выходов.

Идею работы де мультиплексора поясняет рис.3.14. Вход X-информационный, вход A- адресный, потенциал на этом входе определяет, на каком но выходов будет формироваться сигналы, повторяющие X. Когда  $A=1$ , верхний элемент И заперт и на выходе его  $F0=0$ ; нижний элемент, напротив, открыт и работает как повторитель информационных сигналов. При  $A=0$  заперт нижний элемент, а верхний пропускает входную информацию.

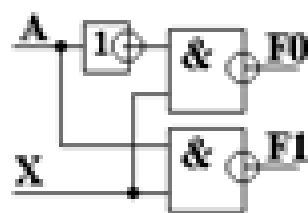


Рис. 3.14. Де мультиплексор 1:2.

Де мультиплексоры ТТЛ с большим числом выходов работают по тому же принципу, только имеют более сложную схему.

Логическая структура простого де мультиплексора вида 1:4 представлена на рис.3.15.

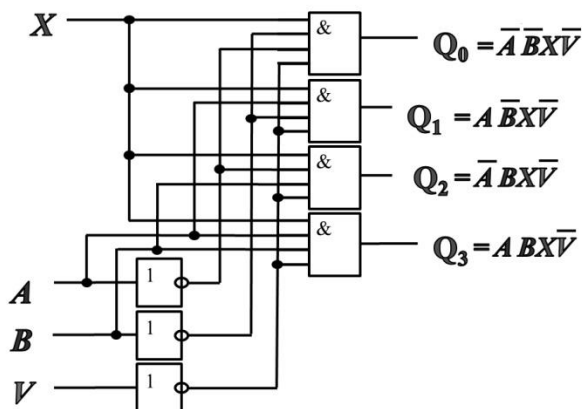


Рис. 3.15. Де мультиплексор 1:4.

Здесь В и А – адресные входы, X – информационный вход, V – разрешающий. Схема функционирует согласно табл. 3.7.

Таблица 3.7.

Таблица истинности работы де мультиплексора 1:4.

Входы				Выходы			
В	А	Х	V	F0	F1	F2	F3
0	0	0/1	0	0/1	1	1	1
0	1	0/1	0	1	0/1	1	1
1	0	0/1	0	1	1	0/1	1
1	1	0/1	0	1	1	1	0/1
0	0	*	1	0	1	1	1
0	1	*	1	1	0	1	1
1	0	*	1	1	1	0	1
1	1	*	1	1	1	1	0

Номера выходных выводов соответствуют двоичному коду на адресных входах (А- младший разряд). Работу устройства описывают

следующие булевы уравнения:

$$F0 = \overline{(\overline{X + V})BA}; F1 = \overline{(\overline{X + V})\overline{BA}}; F2 = \overline{(\overline{X + V})\overline{BA}}; F3 = \overline{(\overline{X + V})BA}$$

Дешифратором (декодером) называют устройство с несколькими входами и выходами, у которого определенным комбинациям входных сигналов соответствует активные состояние одного из выходов. Дешифратор, следовательно, можно рассматривать как обращенный по входам де мультиплексор, у которого адресные входы стали информационными, а бывший информационный вход, на который подается напряжение определенного уровня (0/1), поддерживает напряжение выходных выводов в активном состоянии.

Если у де мультиплексора 1:4 на информационном входе поддерживается потенциал 0 или на разрешающем входе 1, то прибор будет работать как дешифратор 2:4. Таким образом, между обоими типами рассматриваемых устройств нет принципиальной разницы, а различие сводится к виду сигналов на одиночном входе: если они меняются во времени, это де мультиплексор, если нет – дешифратор. У дешифраторов этот вход нередко отсутствует и выходные сигналы на активном выходе имеют одно, наперед известное значение. На условных графических обозначениях у де мультиплексоров в основном поле помещают символ DMX, а дешифраторы обозначают как DC (от англ. decoder).

Дешифраторы и демультимплексоры, оформленные как микросхемы средней степени интеграции, широко применяются в информационно-измерительной технике. Как и мультиплексоры, они часто используются в сочетании со счетчиками и регистрами. Они служат в качестве коммутаторов – распределителей информационных сигналов и синхроимпульсов, для демультимплексирования данных и организации адресной логики в оперативных и постоянных запоминающих устройств, а также для преобразования двоично-десятичного кода в десятичный с целью управления индикаторами и печатающими устройствами. Число выходов и распределение сигналов на них определяются характером предполагаемой нагрузки. Дешифраторы для работы с газоразрядными индикаторными лампами имеют на выходе высоковольтные транзисторы и организацию

выходов "один из десяти". Микросхемы, работающие с семи сегментными индикаторами (полупроводниковыми, накальными, вакуумными), имеют семь выходов и надлежащее распределение сигналов на них при каждом сочетании входных сигналов.

Демультимплексоры-дешифраторы как самостоятельные изделия имеют 4; 8 или 16 выходов. Если потребное число выходных шин превышает возможности МС, де мультимплексоры (дешифраторы) наращиваются в систему. В этом отношении нет принципиального различия с мультимплексорами. В пирамидальных схемах наращивание производится ступенями, причем каждая последующая ступень имеет больше элементов, чем предыдущая. Как пример, на рис.3.17. показан демультимплексор 1:24, составленный из демультимплексора 1:4 и трех - 1:8.

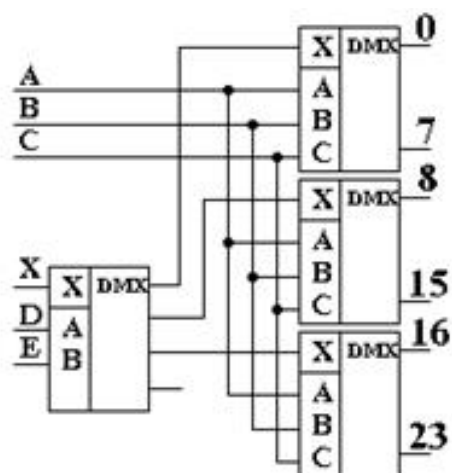


Рис. 3.16. Пирамидальная схема наращивания де мультимплексора 1:24.

Адрес единственного выхода, который находится в активном состоянии, составляется из кода младших разрядов от 000 до 111 (входы С, В, А – общие для всех выходных де мультимплексоров) и кода старших – от 00 до 11 ( входы Е и D). Адресом выхода 4 будет, таким образом, код EDCBA=00100,а выхода 12 – код EDCBA=01100. Они являются двоичными эквивалентами десятичных чисел 4 и 12.

Демультимплексоры-дешифраторы, подобно мультимплексорам, легко приспособить для генерации логических функций согласно заданной таблице истинности. Методику проектирования разберем на



простом примере. Допустим, нужно схемно реализовать функцию, которая описывается таблицей истинности 3.8.

Таблица 3.8.

Таблица истинности логических функций.

Входы			Выходы
X3	X2	X1	
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Непосредственно из таблицы следует, что реализуемая функция может быть представлена в прямом виде  $F = m_1 + m_2 + m_5 + m_7$  или в инверсном  $\overline{F} = m_0 + m_3 + m_6$ , откуда  $F = \overline{m_0 + m_3 + m_6} = \overline{m_0} \cdot \overline{m_3} \cdot \overline{m_6}$ . Схемные воплощения обоих вариантов решения показано на рис.3.18. Очевидно, что в данном конкретном случае второй вариант более экономичен.

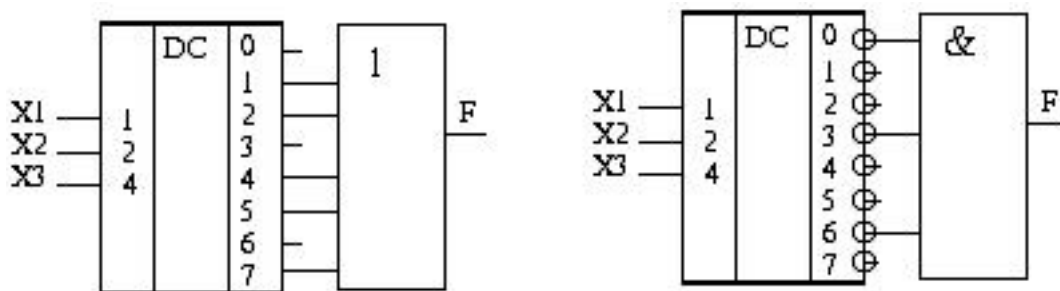


Рис. 3.17. Варианты реализации функций по таблице истинности.

### 3.7. Демультимплексоры-дешифраторы ТТЛ

Устройства подобного рода выпускаются в виде функциональных узлов в составе нескольких серий микросхем ТТЛ, в частности 133,155,531,555 и др. Каждый тип изделий предназначен для определенных задач и характеризуется числом входов, выходов, активным уровнем задействованных выходов, наличием или отсутствием дополнительных выводов, потребляемой мощности, быстродействием и т. п.

#### МИКРОСХЕМА К155ИД4

Микросхема К155ИД4 содержит в одном корпусе два демультимплексора -дешифратора. Каждая секция имеет один информационный и один разрешающий вход. Информационный вход одной секции (ввод 1)- прямой, а другой секции- инверсный (ввод 15). Два адресных входа А и В – общие для обеих секций. Каждая секция имеет по четыре выхода D0-D3 и E0 – E4. Активным уровнем выходных сигналов является 0 рис. 3.18.

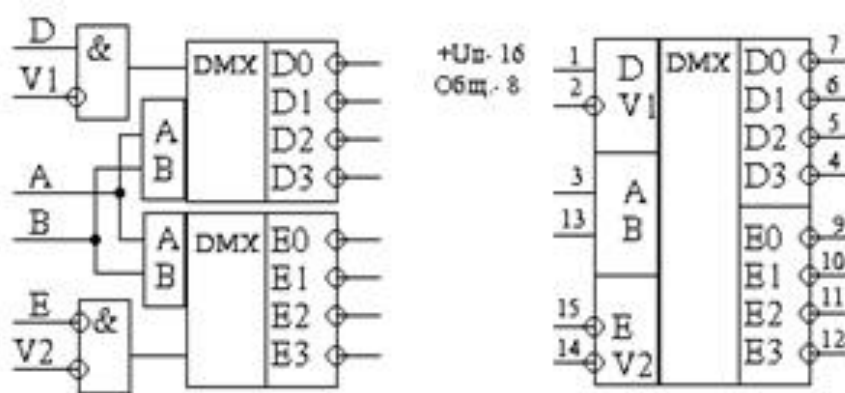


Рис. 3.18. Демультимплексор –дешифратор К155ИД4.

В зависимости от схемы включения МС может быть использована в следующих режимах: два де мультимплексора с 1 линии на 4; два дешифратора с 2 линий на 4; де мультимплексор с 1 линии на 8; дешифратор с 3 линий на 8.

В режиме сдвоенного де мультимплексора 1:4 входы D и E используются как информационные для первого и второго де мультимплексора соответственно, входы V1 и V2 – разрешающие, а

общие входы В и А – адресные. Таблицы истинности для обеих секций несколько различаются за счет различий в информационных входах (табл. 3.9).

Таблица 3.9.

Таблица истинности МС К155ИД4

Адресные входы		Секция D						Секция E					
		Входы		Выходы				Входы		Выходы			
В	А	D	V1	D0	D1	D2	D3	E	V2	E0	E1	E2	E3
0	0	1	0	0	1	1	1	0	0	0	1	1	1
1	0	1	0	1	0	1	1	0	0	1	0	1	1
0	1	1	0	1	1	0	1	0	0	1	1	0	1
1	1	1	0	1	1	1	0	0	0	1	1	1	0
*	*	0	*	1	1	1	1	1	*	1	1	1	1
*	*	*	1	1	1	1	1	*	1	1	1	1	1

Два дешифратора с двух шин на четыре (2:4) получается в том случае, когда входы А и В служат как информационные, а V1 и D (для одной секции) и V2 и E (для другой) играют роль разрешающих. Задавая разрешающие уровни потенциалов на паре входов V1 и D, следует учитывать, что в отличие от инверсного входа V1 вход D прямой.

Для работы в двух других режимах МС включается согласно рис.3.19.

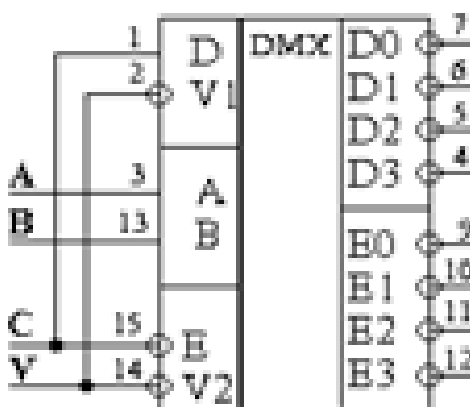


Рис. 3.19. Де мультиплексор с 1 линии на 8.

Если входы А, В, С использовать как адресные, а информацию подавать на вход V, данная схема работает как де мультиплексор 1:8. При заземленном входе V она действует как дешифратор состояний трех входов А, В, С на восемь выходов (от D3 до E3). Для обеспечения очередности действия входов старших разрядов инвертор в данном случае не требуется, так как инверсия сигналов, поступающих на вход E, обеспечивается самой микросхемой.

#### 4. ЦАП и АЦП

Цифро-аналоговые преобразователи (ЦАП, DAC — "Digital-to-Analog Converter") и аналого-цифровые преобразователи (АЦП, ADC — "Analog-to-Digital Converter") главным образом применяются для сопряжения цифровых устройств и систем с внешними аналоговыми сигналами, с реальным миром. При этом АЦП преобразует аналоговые сигналы во входные цифровые сигналы, поступающие на цифровые устройства для дальнейшей обработки или хранения, а ЦАП преобразует выходные цифровые сигналы цифровых устройств в аналоговые сигналы.

ЦАП и АЦП применяются в измерительной технике (цифровые осциллографы, вольтметры, генераторы сигналов и т.д.), в бытовой аппаратуре (телевизоры, музыкальные центры, автомобильная электроника и т.д.), в компьютерной технике (ввод и вывод звука в компьютерах, видеомониторы, принтеры и т.д.), в медицинской технике, в радиолокационных устройствах, в телефонии и во многих других областях. Применение ЦАП и АЦП постоянно расширяется по мере перехода от аналоговых к цифровым устройствам.

В качестве ЦАП и АЦП обычно применяются специализированные микросхемы, выпускаемые многими отечественными и зарубежными фирмами.

Сразу же надо отметить, что для грамотного и профессионального использования микросхем ЦАП и АЦП совершенно не достаточно знания цифровой схемотехники. Эти микросхемы относятся к аналого-цифровым, поэтому они требуют также знания аналоговой схемотехники, существенно отличающейся от цифровой.

Практическое применение ЦАП и АЦП требует расчета аналоговых цепей, учета многочисленных погрешностей преобразования (как статических, так и динамических), знания характеристик и особенностей аналоговых микросхем (в первую очередь, операционных усилителей) и многого другого, что далеко выходит за рамки этой книги. Существует обширная литература, специально посвященная именно вопросам применения ЦАП и АЦП. Поэтому в данной лекции мы не будем говорить о специфике выбора и принципах включения конкретных микросхем ЦАП и АЦП мы будем рассматривать только основные особенности методов соединения ЦАП и АЦП с цифровыми узлами. Нас будет в первую очередь интересовать организация цифровых узлов, предназначенных для соединения с ЦАП и АЦП.

#### 4.1. Применение ЦАП

В общем случае микросхему ЦАП можно представить в виде блока (рис. 4.1), имеющего несколько цифровых входов и один аналоговый вход, а также аналоговый выход.

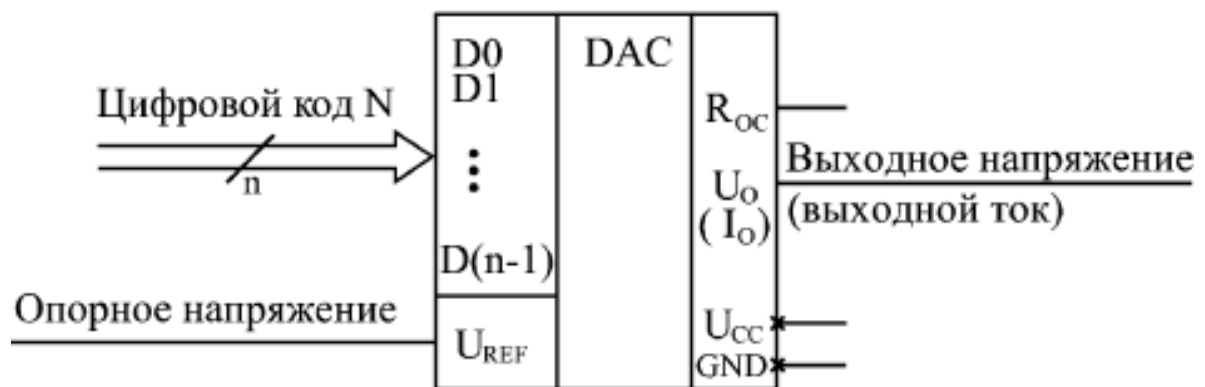


Рис. 4.1. Микросхема ЦАП

На цифровые входы ЦАП подается  $n$ -разрядный код  $N$ , на аналоговый вход – опорное напряжение  $U_{оп}$  (другое распространенное обозначение –  $U_{REF}$ ). Выходным сигналом является напряжение  $U_{вых}$  (другое обозначение —  $U_O$ ) или ток  $I_{вых}$  (другое обозначение —  $I_O$ ). При этом выходной ток или выходное напряжение пропорциональны

входному коду и опорному напряжению. Для некоторых микросхем опорное напряжение должно иметь строго заданный уровень, для других допускается менять его значение в широких пределах, в том числе и изменять его полярность (положительную на отрицательную и наоборот). ЦАП с большим диапазоном изменения опорного напряжения называется умножающим ЦАП, так как его можно легко использовать для умножения входного кода на любое опорное напряжение.

Кроме информационных сигналов, микросхемы ЦАП требуют также подключения одного или двух источников питания и общего провода. Обычно цифровые входы ЦАП обеспечивают совместимость со стандартными выходами микросхем ТТЛ.

Чаще всего в случае, если ЦАП имеет токовый выход, его выходной ток преобразуется в выходное напряжение с помощью внешнего операционного усилителя и встроенного в ЦАП резистора  $R_{oc}$ , один из выводов которого выведен на внешний вывод микросхемы (рис. 4.2). Поэтому, если не оговорено иное, мы будем в дальнейшем считать, что выходной сигнал ЦАП — напряжение  $U_o$ .

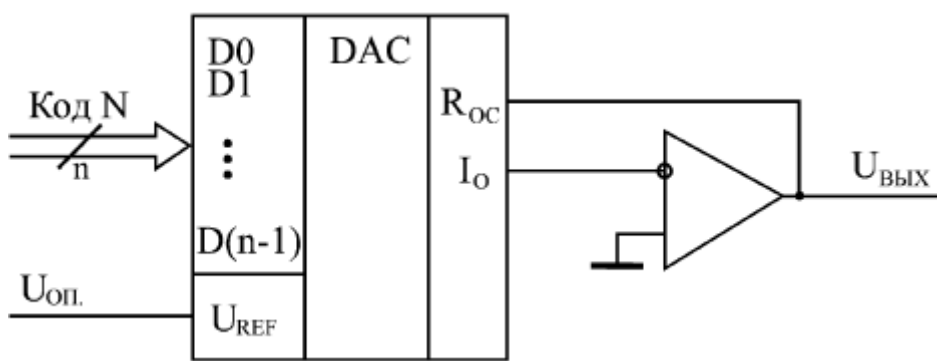


Рис. 4.2. Преобразование выходного тока ЦАП в выходное напряжение.

Суть преобразования входного цифрового кода в выходной аналоговый сигнал довольно проста. Она состоит в суммировании нескольких токов (по числу разрядов входного кода), каждый последующий из которых вдвое больше предыдущего. Для получения этих токов используются или транзисторные источники тока, или резистивные матрицы, коммутируемые транзисторными ключами.

В качестве примера на рис. 4.3 показано 4-разрядное ( $n = 4$ ) цифро-аналоговое преобразование на основе резистивной матрицы  $R-2R$  и ключей (в реальности используются ключи на основе транзисторов). Правому положению ключа соответствует единица в данном разряде входного кода  $N$  (разряды  $D_0 \dots D_3$ ). Операционный усилитель может быть как встроенным (в случае ЦАП с выходом по напряжению), так и внешним (в случае ЦАП с выходом по току).

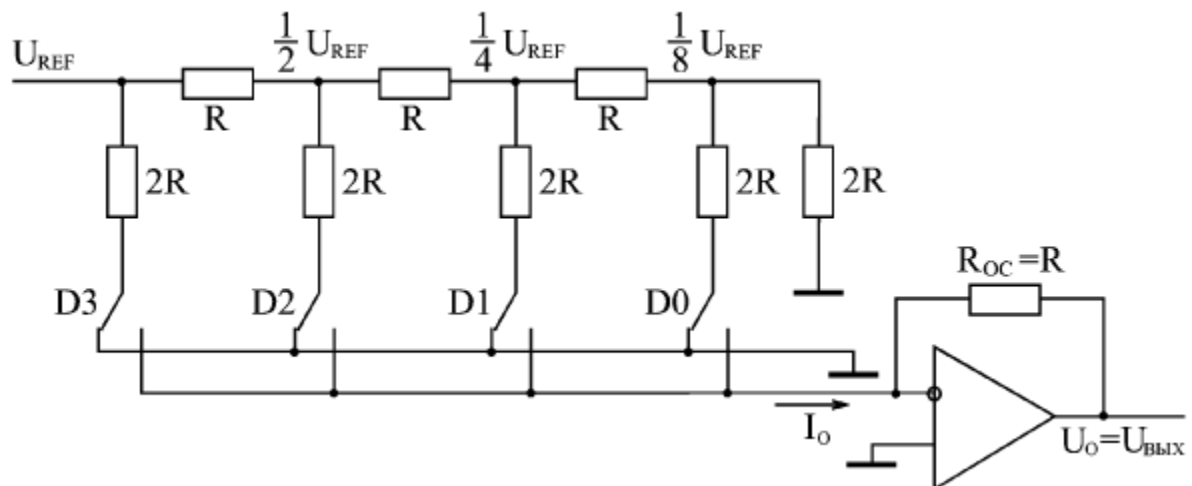


Рис. 4.3. 4-разрядное цифро-аналоговое преобразование.

Первым (левым по рисунку) ключом коммутируется ток величиной  $U_{REF}/2R$ , вторым ключом – ток  $U_{REF}/4R$ , третьим – ток  $U_{REF}/8R$ , четвертым — ток  $U_{REF}/16R$ . То есть токи, коммутируемые соседними ключами, различаются вдвое, как и веса разрядов двоичного кода. Токи, коммутируемые всеми ключами, суммируются и преобразуются в выходное напряжение с помощью операционного усилителя с сопротивлением  $R_{OC}=R$  в цепи отрицательной обратной связи.

При правом положении каждого ключа (единица в соответствующем разряде входного кода ЦАП) ток, коммутируемый этим ключом, поступает на суммирование. При левом положении ключа (ноль в соответствующем разряде входного кода ЦАП) ток, коммутируемый этим ключом, на суммирование не поступает.

Суммарный ток  $I_o$  от всех ключей создает на выходе операционного усилителя напряжение  $U_o = I_o R_{OC} = I_o R$ . То есть вклад первого ключа (старшего разряда кода) в выходное напряжение

составляет  $U_{REF}/2$ , второго —  $U_{REF}/4$ , третьего —  $U_{REF}/8$ , четвертого —  $U_{REF}/16$ . Таким образом, при входном коде  $N = 0000$  выходное напряжение схемы будет нулевым, а при входном коде  $N = 1111$  оно будет равно  $-15U_{REF}/16$ .

В общем случае выходное напряжение ЦАП при  $R_{OC} = R$  будет связано со входным кодом  $N$  и опорным напряжением  $U_{REF}$  простой формулой:

$$U_{ВЫХ} = -N \cdot U_{REF} 2^{-n}$$

где  $N$  — количество разрядов входного кода. Знак минус получается из-за инверсии сигнала операционным усилителем. Эту связь можно проиллюстрировать также табл. 4.1.

Таблица 4.1.

Преобразование ЦАП в однополярном режиме

Входной код $N$	Выходное напряжение $U_{ВЫХ}$
000 . . . 000	0
000 . . . 001	$-2^{-n} U_{REF}$
. . . . .	. . . . .
100 . . . 000	$-2^{-1} U_{REF}$
. . . . .	. . . . .
111 . . . 111	$-(1-2^{-n}) U_{REF}$

Некоторые микросхемы ЦАП предусматривают возможность работы в биполярном режиме, при котором выходное напряжение изменяется не от нуля до  $U_{REF}$ , а от  $-U_{REF}$  до  $+U_{REF}$ . При этом выходной сигнал ЦАП  $U_{ВЫХ}$  умножается на 2 и сдвигается на величину  $U_{REF}$ . Связь между входным кодом  $N$  и выходным напряжением  $U_{ВЫХ}$  будет следующей:

$$U_{ВЫХ} = U_{REF}(1 - N \cdot 2^{1-n})$$

Это можно проиллюстрировать табл. 4.2. Такое биполярное преобразование при возможности смены знака опорного напряжения называется также четырехквadrантным умножением (То есть и опорное, и выходное напряжения могут быть в данном случае как положительными, так и отрицательными).



Таблица 4.2.

## Преобразование ЦАП в биполярном режиме

Входной код N	Выходное напряжение $U_{\text{ВЫХ}}$
000 . . . 000	$U_{\text{REF}}$
. . .	. . .
011 . . . 111	$2^{-n} U_{\text{REF}}$
100 . . . 000	0
. . .	. . .
111 . . . 111	$-(1-2^{1-n}) U_{\text{REF}}$

Микросхемы ЦАП, имеющиеся на рынке, различаются количеством разрядов (от 8 до 24), величиной задержки преобразования (от единиц наносекунд до единиц микросекунд), допустимой величиной опорного напряжения (обычно — единицы вольт), величинами погрешностей преобразования и другими параметрами. Различаются они также технологией изготовления и особенностями внутренней структуры, что нередко накладывает ограничения на их использование. Поэтому выбирать микросхему ЦАП для конкретного применения необходимо с использованием подробной справочной информации, предоставляемой фирмами-изготовителями. Мы же будем говорить только об общих принципах включения ЦАП в цифровые схемы без учета их частных особенностей.

Иногда бывает необходимо уменьшить количество разрядов ЦАП. Для этого нужно подать сигналы логического нуля на нужное число младших разрядов ЦАП (но никак не старших разрядов). На рис. 4.4 показано, как из 10-разрядного ЦАП можно сделать 8-разрядный, подав нули на два младших разряда. Увеличение количества разрядов ЦАП представляет собой гораздо более сложную задачу, требующую построения сложных аналоговых схем, поэтому оно встречается довольно редко. Значительно проще подобрать микросхему с нужным или с большим, чем нужно, количеством разрядов.

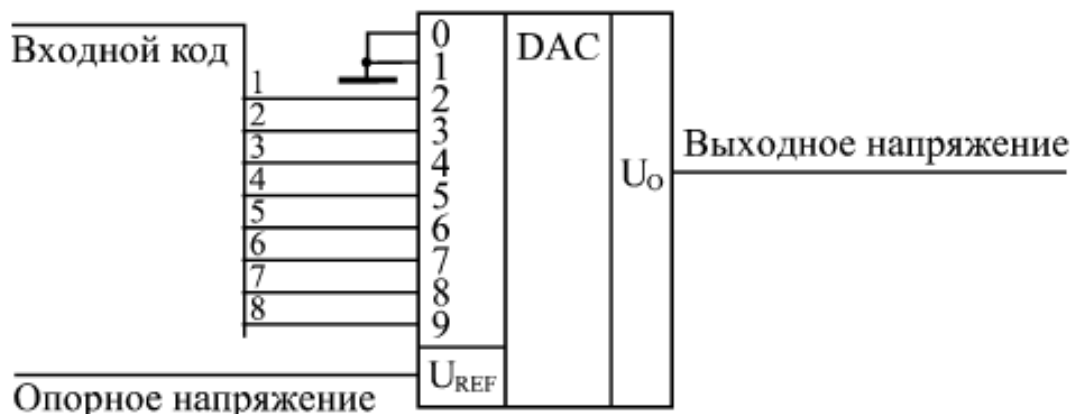


Рис. 4.4. Уменьшение разрядности ЦАП

Основное применение микросхем ЦАП состоит в получении аналогового сигнала из последовательности цифровых кодов (рис. 4.5). Как правило, коды подаются на входы ЦАП через параллельный регистр, что позволяет обеспечить одновременность изменения всех разрядов входного кода ЦАП.

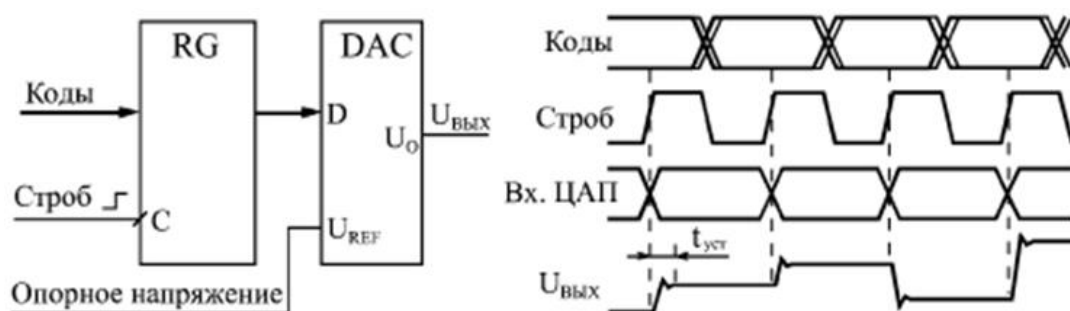


Рис. 4.5. Преобразование последовательности кодов в выходное напряжение

При неодновременном изменении разрядов входного кода на выходе ЦАП появляются большие короткие импульсы напряжения, уровни которых не соответствуют ни одному из кодов.

Однако, даже при одновременном изменении всех разрядов входного кода ЦАП, уровень напряжения, соответствующий поданному коду, устанавливается не сразу, а за время установления ЦАП  $t_{уст}$ , что связано с неидеальностью внутренних элементов ЦАП. Выходной ток ЦАП, как правило, устанавливается значительно

быстрее выходного напряжения, так как он не зависит от инерционности операционного усилителя. Понятно, что условие правильной работы ЦАП состоит в том, чтобы длительность сохранения входного кода была больше, чем время установления ЦАП  $t_{уст}$ , иначе выходной сигнал не успеет принять значение, соответствующее входному коду.

Если подавать коды на вход ЦАП редко, то приведенная на рис. 4.5 схема может использоваться, например, в управляемом источнике питания, выходное напряжение которого задается входным кодом. Правда, при этом необходимо еще обеспечить большой выходной ток источника питания, применив внешний усилитель тока.

Если же подавать коды на вход ЦАП с высокой частотой, то можно получить генератор (он же синтезатор) аналоговых сигналов произвольной формы. В этом случае коды, поступающие на ЦАП, называют кодами выборок (то есть мгновенных значений) генерируемого аналогового сигнала.

В простейшем случае в качестве источника входных кодов ЦАП можно использовать обычный двоичный счетчик (рис. 4.6). Выходное напряжение ЦАП будет нарастать при этом на величину  $2^{-n}U_{REF}$  с каждым тактовым импульсом, формируя пилообразные выходные сигналы амплитудой  $U_{REF}$ . Длительность каждой ступеньки равна периоду тактового генератора  $T$ , а период всего выходного сигнала равен  $2^n T$ . Количество ступенек в периоде выходного сигнала равно  $2^n$ . Если в данной схеме использовать синхронные счетчики с синхронным переносом, то входной регистр ЦАП не нужен, так как все разряды счетчика переключаются одновременно. Если же используются асинхронные счетчики или синхронные счетчики с асинхронным переносом, то входной регистр ЦАП необходим.

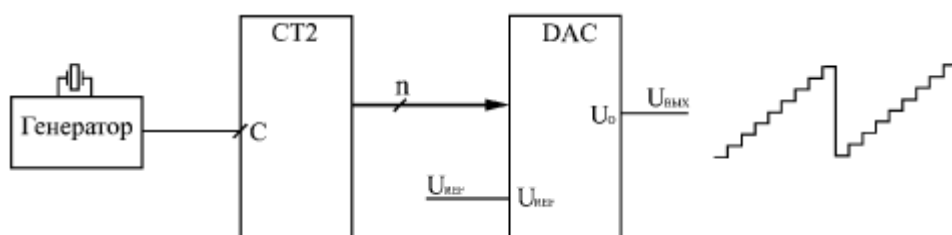


Рис. 4.6. Генератор пилообразного аналогового сигнала.

В случае, когда нужно формировать аналоговые сигналы произвольной формы (синусоидальные, колоколообразные, шумовые, треугольные, импульсные и т.д.), в качестве источника кодов, поступающих на ЦАП, необходимо использовать память, работающую в режиме чтения (рис. 4.7).

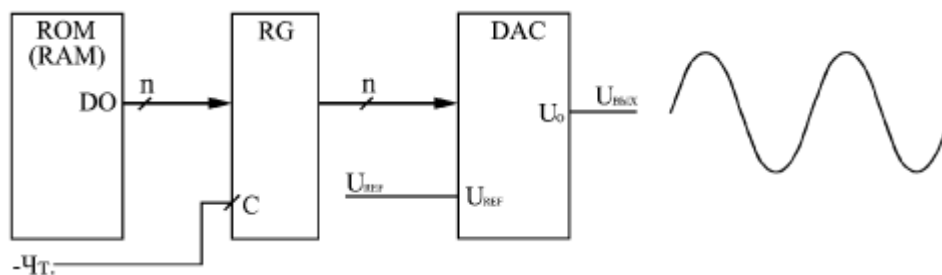


Рис. 4.7. Генерация сигналов произвольной формы.

Если память постоянная, то набор форм генерируемых сигналов задается раз и навсегда. Если же память оперативная, то строится однонаправленный информационный буфер с периодическим режимом работы, что позволит записывать в память коды для генерации самых разных сигналов. В обоих случаях входной регистр ЦАП необходим, информация в него записывается стробом чтения из памяти.

Как и в предыдущем случае, выходной сигнал ЦАП будет состоять из ступенек, высота которых кратна  $2^{-n}U_{REF}$ . Амплитуда выходного сигнала не превышает  $U_{REF}$ . Если адреса памяти перебираются счетчиком, то период выходного аналогового сигнала равен  $2^m T$ , где  $T$  — период тактового сигнала чтения из памяти "– Чт.", а  $m$  — количество адресных разрядов памяти.

Если надо вычислить коды выборок для генерации какого-то периодического сигнала, то необходимо его период разделить на  $2^m$  частей и вычислить соответствующие  $2^m$  значений этого сигнала  $U^i$ . Затем надо пересчитать значения сигнала в коды по формуле  $N_i = 2^n U_i / A$  где  $A$  — амплитуда сигнала, и взять ближайшее целое значение кода. Нулевое значение сигнала даст при этом нулевой код 000 ... 000, максимальное значение сигнала (равное амплитуде  $A$ ) даст максимальный код 111 ... 111. В результате подачи этих кодов на ЦАП с периодом  $T$  будет генерироваться аналоговый сигнал

требуемой формы с амплитудой, равной  $U_{REF}$  и с периодом  $T_{ВЫХ}=2^m T$ . Пример такого вычисления проиллюстрирован рис. 4.8.

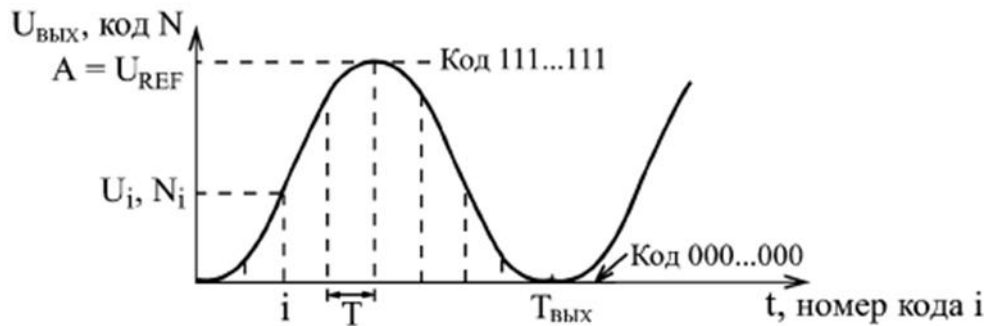


Рис. 4.8. Вычисление кодов выборок периодического сигнала.

Подробнее задача проектирования генератора аналоговых сигналов произвольной формы будет рассмотрена в следующей лекции.

Преобразование цифровых кодов в аналоговый сигнал — это не единственное применение микросхем ЦАП. Они могут также использоваться для управляемой обработки аналоговых сигналов, например, для усиления и ослабления аналоговых сигналов в заданное число раз. Для этого лучше всего подходят умножающие ЦАП, которые допускают изменение уровня опорного напряжения в широких пределах, в том числе и с изменением его знака. Таких микросхем ЦАП выпускается сейчас достаточно много, с различным быстродействием и с различным количеством разрядов входного кода.

Самая простейшая схема — это цифровой аттенюатор (ослабитель) аналогового сигнала (рис. 4.9), применяемый часто для регулировки амплитуды выходного сигнала генератора на основе ЦАП.

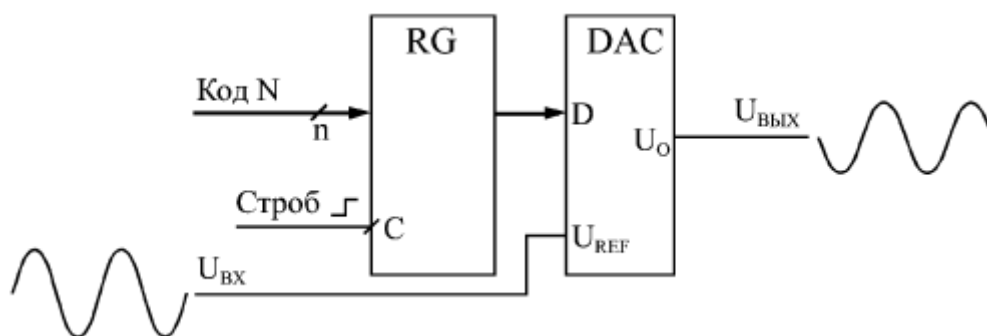


Рис. 4.9. Аттенюатор аналогового сигнала на ЦАП

Схема практически ничем не отличается от схемы на рис. 4.5. Но два важных отличия все же имеются: вместо постоянного опорного напряжения подается переменный аналоговый сигнал, а ЦАП должен быть обязательно умножающим. Выходной сигнал связан со входным по простой формуле

$$U_{\text{ВЫХ}} = - U_{\text{ВХ}} \cdot N 2^{-n},$$

то есть выходной сигнал пропорционален входному (с инверсией), а коэффициент пропорциональности определяется входным цифровым кодом  $N$ . Коэффициент пропорциональности изменяется в данном случае от нуля и почти до единицы с шагом, равным  $2^{-n}$ .

Входной регистр ЦАП в данном случае также необходим, поскольку при неодновременном переключении разрядов входного кода на выходной сигнал ЦАП могут накладываться короткие импульсы значительной амплитуды. Требования к быстродействию ЦАП (к величине его времени установления) в данном включении не слишком высоки, так как амплитуду выходного сигнала обычно требуется менять нечасто. А частота входного аналогового сигнала может быть довольно большой, она никак не связана с временем установления ЦАП.

Существует также схема включения ЦАП, которую можно использовать как управляемый усилитель аналогового сигнала с коэффициентом усиления, задаваемым входным кодом  $N$  (рис. 4.10).

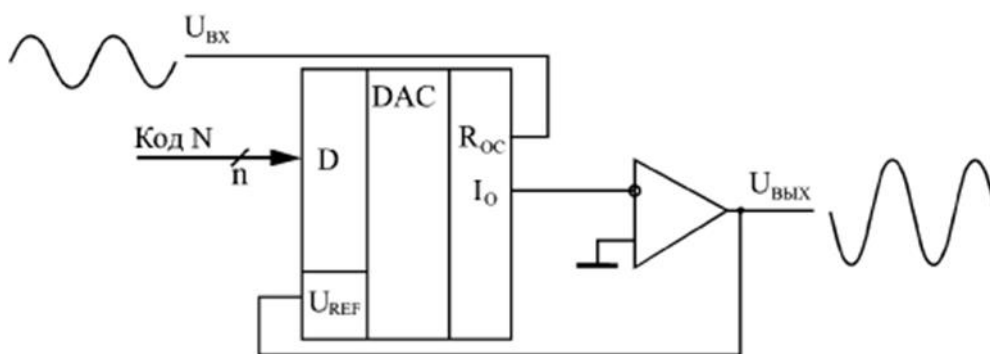


Рис. 4.10. Управляемый усилитель входного сигнала.

В этом случае выходной ток ЦАП равен величине  $U_{\text{ВХ}}/R_{\text{ОС}}$ , а так как в качестве опорного напряжения используется выходное

напряжение, то получается, что выходное напряжение связано со входным по формуле

$$U_{\text{ВЫХ}} = - U_{\text{ВХ}} \cdot N2^{-n}/N.$$

То есть коэффициент пропорциональности между выходным и входным напряжениями обратно пропорционален коду  $N$ . Код  $N$  может меняться в этом случае от 1 до  $(2^n - 1)$ , что соответствует коэффициенту усиления от примерно единицы до  $2^n$ . Например, при 10-разрядном ЦАП коэффициент усиления схемы может достигать 1024.

Как и в предыдущем случае, скорость переключения ЦАП не очень важна, так как коэффициент усиления обычно не требуется переключать слишком часто. На схеме для простоты не показан входной регистр ЦАП, который опять же необходим, чтобы обеспечить одновременность переключения всех разрядов входного кода.

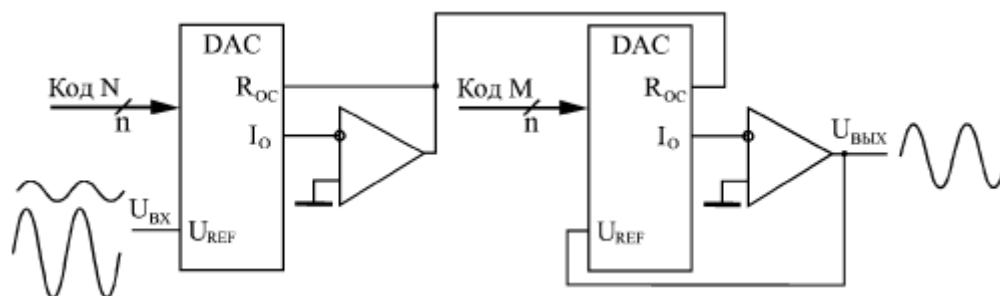


Рис. 4.11. Последовательное включение аттенюатора и усилителя.

Используя последовательное включение схем рис. 4.9 и рис. 4.10, можно обеспечить приведение к стандартному уровню входного напряжения, изменяемого в очень широких пределах (рис. 4.11). Такая задача часто встречается в аналого-цифровых системах. Коэффициент передачи всей схемы будет равен отношению входных кодов обоих ЦАП  $N/M$  и может быть установлен с высокой точностью как в диапазоне от 0 до 1 (аттенюатор), так и в диапазоне от 1 до  $2^n$  (усилитель). На схеме не показаны входные регистры обоих ЦАП, но они также нужны.

Наконец, последняя схема с применением ЦАП, которую мы рассмотрим, — это схема сдвига аналогового сигнала на величину, задаваемую входным цифровым кодом. Сдвиг представляет собой, по

сути, сложение аналогового сигнала с постоянным напряжением. Такая задача довольно часто встречается в аналого-цифровых системах.

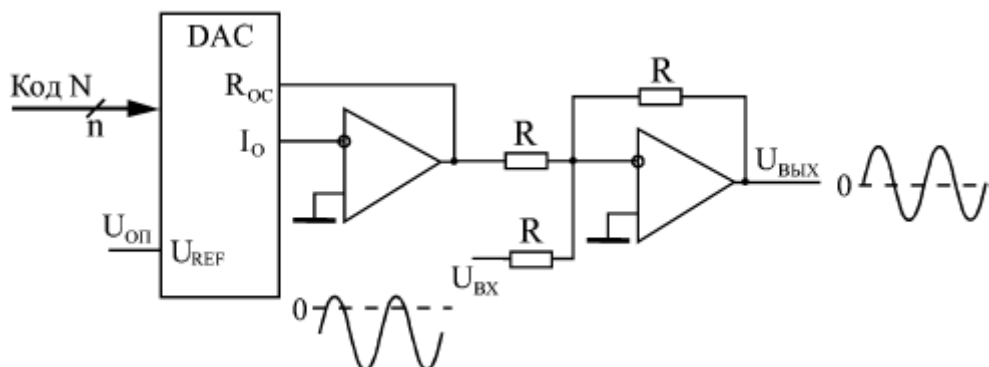


Рис. 4.12. Схема управляемого сдвига аналогового сигнала.

Схема сдвига (рис. 4.12) включает в себя преобразователь цифрового кода в выходное напряжение и аналоговый сумматор на операционном усилителе. Величина напряжения сдвига входного сигнала будет равна  $U_{REF} \cdot 2^{-n} N$ . Поскольку применяются два инвертирующих операционных усилителя, инверсии входного сигнала на выходе в данном случае не будет. Если нужен как положительный, так и отрицательный сдвиг, то необходимо применять ЦАП с биполярным выходным сигналом.

## 4.2. Применение АЦП

Микросхемы АЦП выполняют функцию, прямо противоположную функции ЦАП, — преобразуют входной аналоговый сигнал в последовательность цифровых кодов. В общем случае микросхему АЦП можно представить в виде блока, имеющего один аналоговый вход, один или два входа для подачи опорного (образцового) напряжения, а также цифровые выходы для выдачи кода, соответствующего текущему значению аналогового сигнала (рис. 4.13).

Часто микросхема АЦП имеет также вход для подачи тактового сигнала CLK, сигнал разрешения работы CS и сигнал, говорящий о готовности выходного цифрового кода RDY. На микросхему подается



одно или два питающих напряжения и общий провод. В целом микросхемы АЦП сложнее, чем микросхемы ЦАП, их разнообразие заметно больше, и поэтому сформулировать для них общие принципы применения сложнее.

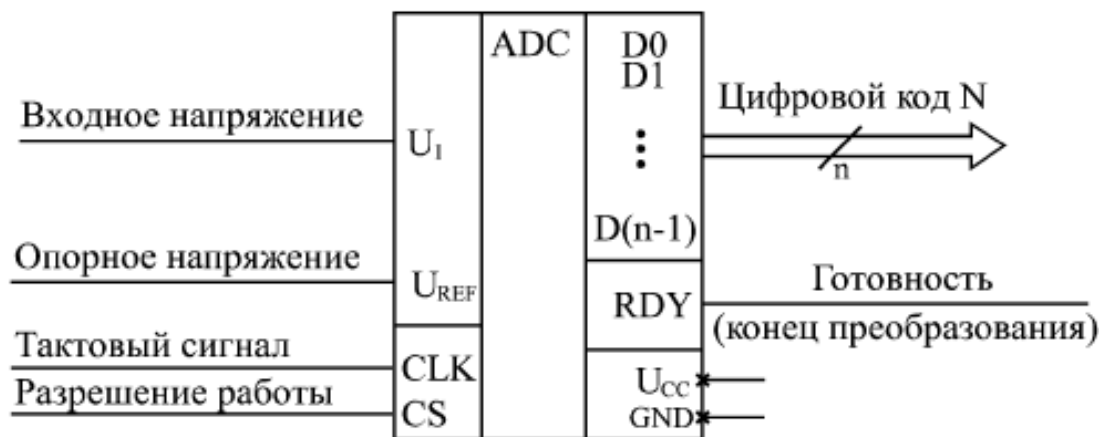


Рис. 4.13. Микросхема АЦП.

Опорное напряжение АЦП задает диапазон входного напряжения, в котором производится преобразование. Оно может быть постоянным или же допускать изменение в некоторых пределах. Иногда предусматривается подача на АЦП двух опорных напряжений с разными знаками, тогда АЦП способен работать как с положительными, так и с отрицательными входными напряжениями.

Выходной цифровой код  $N$  ( $n$ -разрядный) однозначно соответствует уровню входного напряжения. Код может принимать  $2^n$  значений, то есть АЦП может различать  $2^n$  уровней входного напряжения. Количество разрядов выходного кода  $n$  представляет собой важнейшую характеристику АЦП. В момент готовности выходного кода выдается сигнал окончания преобразования  $RDY$ , по которому внешнее устройство может читать код  $N$ .

Управляется работа АЦП тактовым сигналом  $CLK$ , который задает частоту преобразования, то есть частоту выдачи выходных кодов. Предельная тактовая частота — второй важнейший параметр АЦП. В некоторых микросхемах имеется встроенный генератор тактовых сигналов, поэтому к их выводам подключается кварцевый генератор

или конденсатор, задающий частоту преобразования. Сигнал CS разрешает работу микросхемы.

Выпускается множество самых разнообразных микросхем АЦП, различающихся скоростью работы (частота преобразования от сотен килогерц до сотен мегагерц), разрядностью (от 6 до 24), допустимыми диапазонами входного сигнала, величинами погрешностей, уровнями питающих напряжений, методами выдачи выходного кода (параллельный или последовательный), другими параметрами. Обычно микросхемы с большим количеством разрядов имеют невысокое быстродействие, а наиболее быстродействующие микросхемы имеют небольшое число разрядов. Область применения любой микросхемы АЦП во многом определяется использованным в ней принципом преобразования, поэтому необходимо знать особенности этих принципов. Для выбора и использования АЦП необходимо пользоваться подробными справочными данными от фирмы-производителя.

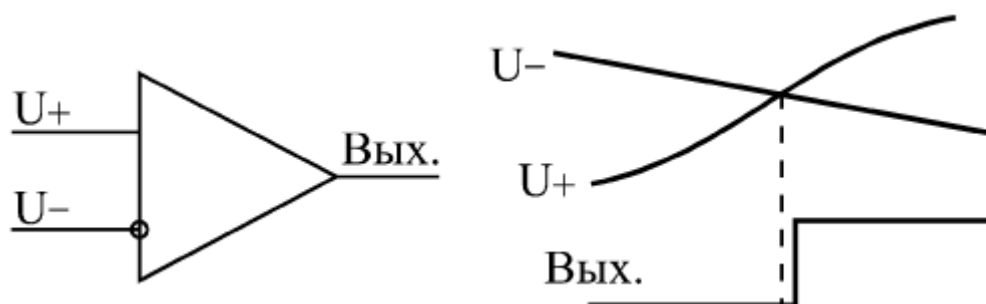


Рис. 4.14. Компаратор напряжения.

В качестве базового элемента любого АЦП используется компаратор напряжения (рис. 4.14), который сравнивает два входных аналоговых напряжения и, в зависимости от результата сравнения, выдает выходной цифровой сигнал — нуль или единицу. Компаратор работает с большим диапазоном входных напряжений и имеет высокое быстродействие (задержка порядка единиц наносекунд).

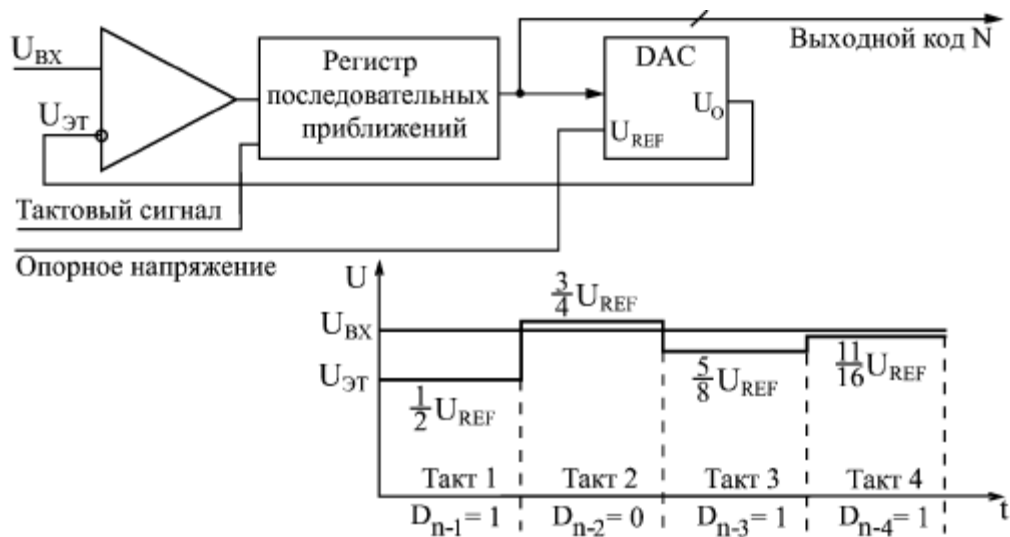


Рис. 4.15. АЦП последовательного типа.

Существует два основных принципа построения АЦП: последовательный и параллельный.

В последовательном АЦП входное напряжение последовательно сравнивается одним единственным компаратором с несколькими эталонными уровнями напряжения, и в зависимости от результатов этого сравнения формируется выходной код. Наибольшее распространение получили АЦП на основе так называемого регистра последовательных приближений (рис. 4.15).

Входное напряжение подается на вход компаратора, на другой вход которого подается эталонное напряжение, ступенчато изменяющееся во времени. Выходной сигнал компаратора подается на вход регистра последовательных приближений, тактируемого внешним тактовым сигналом. Выходной код регистра последовательных приближений поступает на ЦАП, которое из опорного напряжения формирует меняющееся эталонное напряжение.

Регистр последовательных приближений работает так, что в зависимости от результата предыдущего сравнения выбирается следующий уровень эталонного напряжения по следующему алгоритму:

- В первом такте входной сигнал сравнивается с половиной опорного напряжения.

- Если входной сигнал меньше половины опорного напряжения, то на следующем такте он сравнивается с четвертью опорного напряжения (то есть половина опорного напряжения уменьшается на четверть). Одновременно в регистр последовательных приближений записывается старший разряд выходного кода, равный нулю.

- Если же входной сигнал больше половины опорного напряжения, то на втором такте он сравнивается с  $3/4$  опорного напряжения (то есть половина увеличивается на четверть). Одновременно в регистр последовательных приближений записывается старший разряд выходного кода, равный единице.

- Затем эта последовательность сравнений повторяется нужное число раз с уменьшением на каждом такте вдвое ступени изменения эталонного напряжения (на третьем такте —  $1/8$  опорного напряжения, на четвертом —  $1/16$  и т.д.). В результате опорное напряжение в каждом такте приближается к входному напряжению. Всего преобразование занимает  $n$  тактов. В последнем такте вычисляется младший разряд.

Понятно, что процесс этот довольно медленный, требует нескольких тактов, причем в течение каждого такта должны успеть сработать компаратор, регистр последовательных приближений и ЦАП с выходом по напряжению. Поэтому последовательные АЦП довольно медленные, имеют сравнительно большое время преобразования и малую частоту преобразования.

Второй тип АЦП — АЦП параллельного типа — работает по более простому принципу. Все разряды выходного кода вычисляются в них одновременно (параллельно), поэтому они гораздо быстрее, чем последовательные АЦП. Правда, они требуют применения большого количества компараторов ( $2^n - 1$ ), что вызывает чисто технологические трудности при большом количестве разрядов (например, при 12-разрядном АЦП требуется 4095 компараторов).

Схема такого АЦП (рис. 4.16) включает в себя резистивный делитель из  $2^n$  одинаковых резисторов, который делит опорное напряжение на  $(2^n - 1)$  уровней.

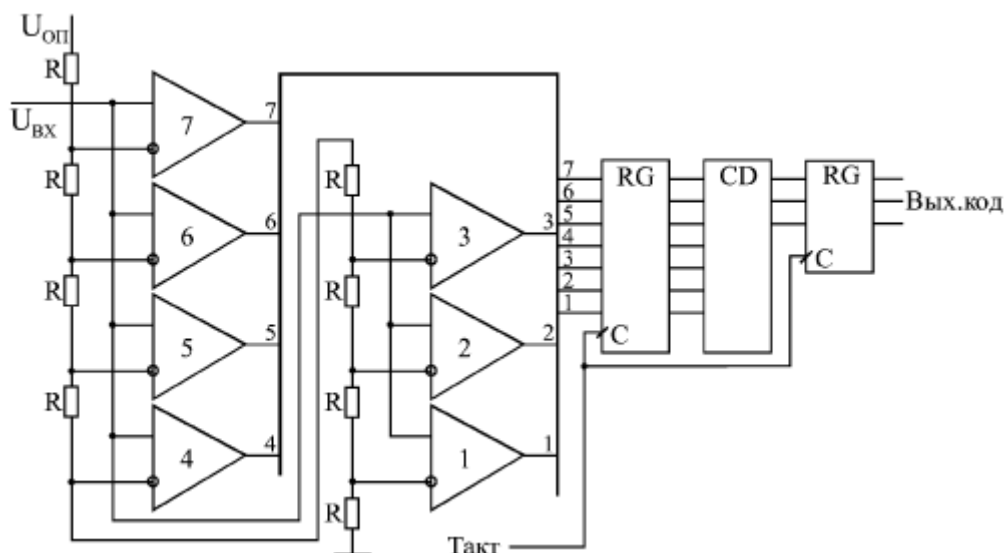


Рис. 4.16. 3-разрядный АЦП параллельного типа.

Входное напряжение сравнивается с помощью компараторов с уровнями, формируемыми делителем напряжения. Выходные сигналы компараторов с помощью шифратора преобразуются в  $n$ -разрядный двоичный код. Шифратор выдает на выход номер последнего из сработавших (то есть выдавших сигнал логической единицы) компараторов. Например, в случае 3-разрядного АЦП (на рисунке) при величине входного напряжения от 0 до  $1/8$  опорного напряжения выходной код будет 000, при входном напряжении от  $1/8$  до  $2/8$  опорного напряжения сработает первый компаратор, что даст выходной код 001, при входном напряжении от  $2/8$  до  $3/8$  опорного напряжения сработают компараторы 1 и 2, что даст выходной код 010, и т.д. Процесс преобразования происходит в параллельном АЦП очень быстро, поэтому частота преобразования может достигать сотен мегагерц.

Для повышения быстродействия в параллельном АЦП иногда применяется конвейерный принцип: выходной код компараторов записывается в  $(2^n - 1)$ -разрядный параллельный регистр, показанный на рис. 4.16. Выходной код шифратора также записывается в  $n$ -разрядный параллельный регистр. Оба регистра в этом случае тактируются одним и тем же тактовым сигналом. Это снижает требования к быстродействию компараторов и шифратора. Правда,

выходной код АЦП задерживается из-за таких регистров на два периода таковой частоты.

Громоздкость структуры параллельного АЦП приводит к тому, что в некоторых АЦП применяется смешанный параллельно-последовательный принцип. Это несколько снижает быстродействие подобного АЦП по сравнению с обычным параллельным АЦП, но зато позволяет получить большое число разрядов, не увеличивая количество компараторов до  $(2^n - 1)$ .

Для того чтобы АЦП любого типа работал с использованием всех своих возможностей, необходимо обеспечить согласование диапазона изменения входного аналогового сигнала с допустимым диапазоном (динамическим диапазоном) входного напряжения АЦП.

На рис. 4.17 показано четыре возможных случая соотношения динамического диапазона АЦП (от 0 до  $U_{REF}$  или от  $U_{REF1}$  до  $U_{REF2}$ ) и входного сигнала. В случаях а и б входной сигнал меньше динамического диапазона, поэтому АЦП будет работать правильно, но не будет использовать всех своих возможностей. В случае в входной сигнал слишком большой, поэтому часть его значений не будет преобразована. Только в случае г АЦП действительно будет работать как  $n$ -разрядный и будет преобразовывать все значения входного сигнала. Для согласования входного сигнала с динамическим диапазоном АЦП можно применять усилители, аттенюаторы, схемы сдвига. В некоторых случаях согласование может быть достигнуто простым выбором величин опорных напряжений.

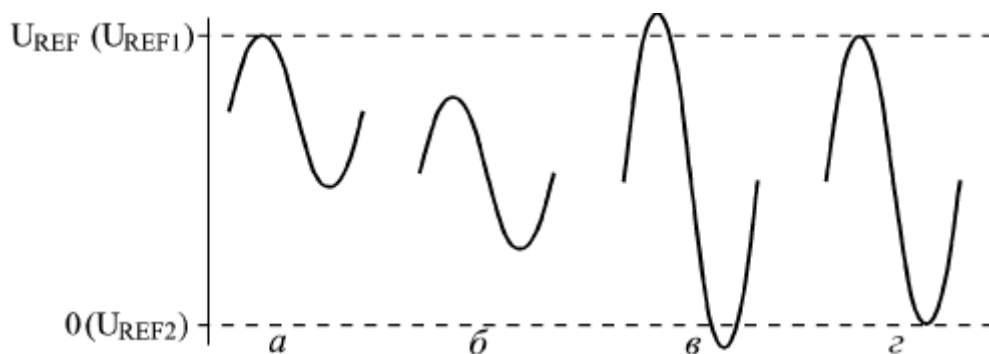


Рис. 4.17. Соотношение входного сигнала и динамического диапазона АЦП

Иногда бывает необходимо уменьшить количество разрядов АЦП. В этом случае нужное количество младших разрядов выходного кода микросхемы просто не используется. На рис. 4.18 показано использование 10-разрядного АЦП в качестве 8-разрядного.

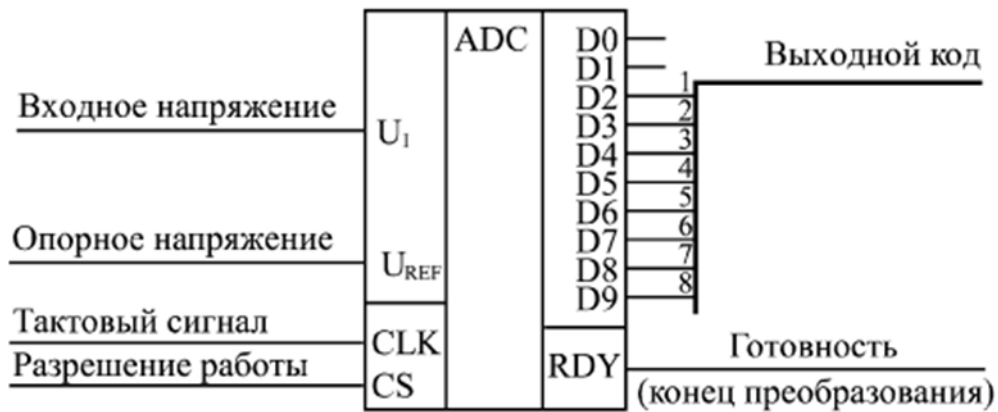


Рис. 4.18. Уменьшение количества разрядов выходного кода АЦП

Обратная задача — увеличение разрядности АЦП — встречается чаще. Существует ряд типичных схемотехнических решений по объединению нескольких микросхем АЦП для увеличения количества разрядов выходного кода, но большинство этих решений требует сложных расчетов результирующих погрешностей преобразования и применения аналоговых узлов. Мы не будем их здесь рассматривать. Отметим только, что при возникновении задачи увеличения разрядности надо, прежде всего, попытаться найти микросхему с нужным количеством разрядов, и только потом рассматривать возможности объединения нескольких микросхем АЦП.

Рассмотрим несколько типичных схем включения АЦП, используемых в аналого-цифровых системах.

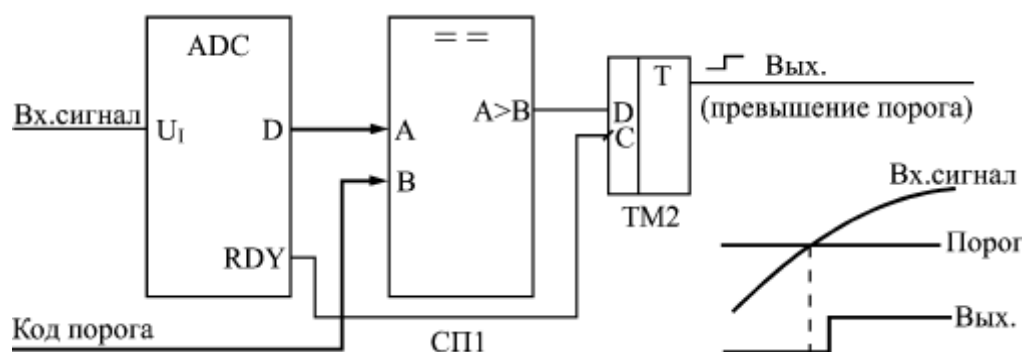


Рис. 4.19. Фиксатор превышения входным сигналом установленного порога.

Первая схема (рис. 4.19) предназначена для фиксации момента превышения входным аналоговым сигналом заданного порогового напряжения. Схема вырабатывает выходной сигнал (положительный фронт) тогда, когда входной аналоговый сигнал становится больше установленного уровня, причем уровень этот задается цифровым кодом порога. Код порога сравнивается с выходными кодами АЦП с помощью микросхемы компаратора кодов. Выходной сигнал компаратора кодов записывается в триггер по сигналу RDY с АЦП, что позволяет исключить влияние коротких импульсов, возникающих на выходе компаратора в момент изменения входных кодов. Применение этого триггера задерживает выходной сигнал на один такт.

Может показаться, что применение АЦП в данном случае не оправданно, избыточно. Но надо учитывать, что в аналого-цифровых системах АЦП, преобразующий входной сигнал в последовательность кодов, как правило, уже есть, поэтому дополнительного АЦП не требуется, достаточно только включить компаратор кодов и триггер.

АЦП также применяется в схемах вычисления амплитуды входного аналогового сигнала. Для такого вычисления можно использовать уже рассмотренную схему вычислителя экстремального значения входного кода. В качестве источника последовательности входных кодов в данном случае выступает АЦП (рис. 4.20).



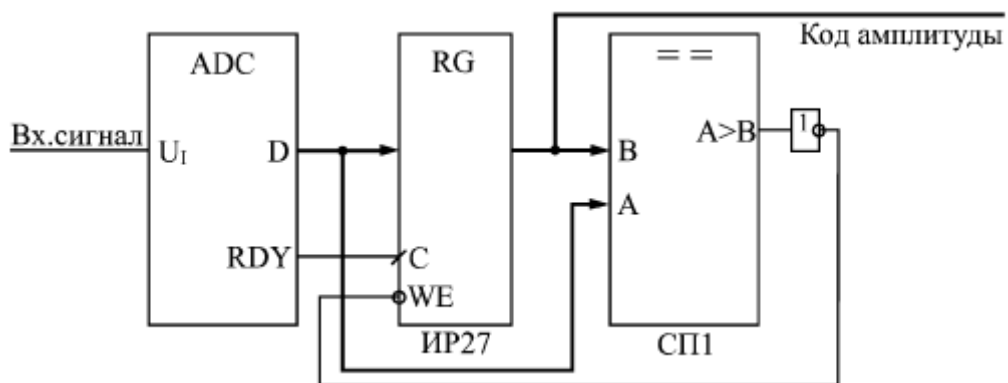


Рис. 4.20. Вычислитель амплитуды аналогового сигнала.

В регистр со входом разрешения записи записывается код с выхода АЦП по сигналу RDY в том случае, если текущее значение кода больше значения кода, записанного ранее в регистр. В результате уже после одного периода входного сигнала в регистре будет код амплитуды входного сигнала. За период преобразования АЦП должны успеть сработать компаратор кодов и регистр.

Если такой вычислитель амплитуды входного сигнала используется в составе сложной аналого-цифровой системы, в которой уже присутствует АЦП, непрерывно преобразующий входной сигнал в коды, то дополнительно требуются только цифровые микросхемы: компаратор кодов и регистр.

Наиболее часто встречающееся использование АЦП — это преобразование входного сигнала в поток кодов, причем коды эти обычно записываются в буферную память. В данном случае наиболее подходящим является однонаправленный буфер с периодическим режимом работы. То есть сначала в буферную память заносится массив кодов выборок входного сигнала, а затем этот массив читается для дальнейшей обработки. Именно так, например, строится цифровой осциллограф, предназначенный для наблюдения аналоговых сигналов на экране.

Схема включения АЦП в этом случае показана на рис. 4.21. В качестве строба записи в буферную память используется сигнал RDY с АЦП.

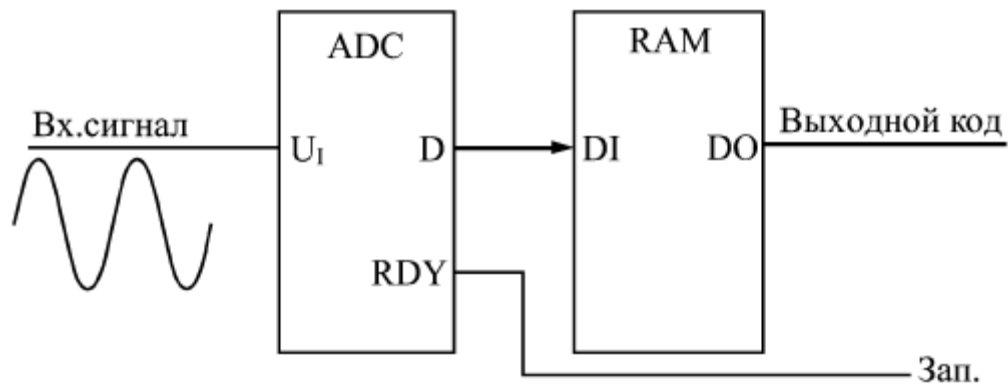


Рис. 4.21. Включение буферной памяти для запоминания кодов с выходов АЦП

Конечно, в реальных аналого-цифровых устройствах все гораздо сложнее, в них требуются схемы синхронизации процесса записи со входным сигналом, схемы предварительной обработки аналогового сигнала, но суть остается той же — буферная память, записывающая последовательность кодов с выхода АЦП. Чем больше объем памяти, тем больший фрагмент входного аналогового сигнала она может запомнить. Например, если память имеет организацию 64Кх8 и работает с 8-разрядным АЦП, то при частоте преобразования АЦП 10 МГц буфер сможет хранить в себе фрагмент аналогового сигнала длительностью 6,5536 мс.

Наконец, последняя схема, которую мы рассмотрим (рис. 4.22), позволяет вдвое повысить быстродействие АЦП, точнее, поднять вдвое частоту записи кодов выборок входного сигнала в буферную память.

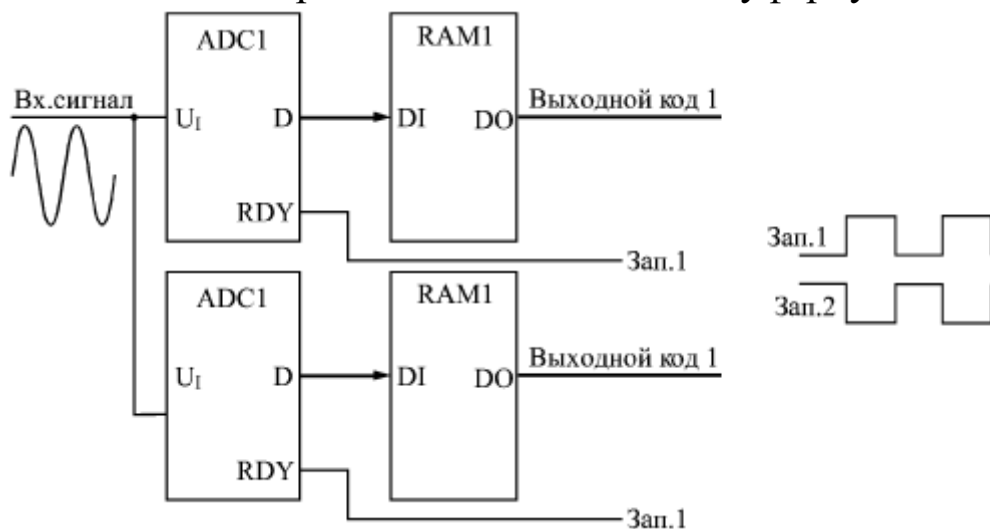


Рис. 4.22. Увеличение вдвое частоты преобразования входного сигнала с помощью двух АЦПс буферами.

Идея схемы очень проста: используется два АЦП и два буфера, которые работают по очереди, например, четные выборки входного сигнала обрабатывает один АЦП со своим буфером, а нечетные — другой АЦП со своим буфером. В результате запоминание кодов входного сигнала осуществляется с частотой вдвое больше частоты преобразования каждого из АЦП. Например, если каждый АЦП и каждый буфер работают с частотой 10 МГц, то результирующая частота преобразования составит 20 МГц.

Тактовые сигналы АЦП и сигналы RDY на выходах АЦП должны быть сдвинуты один относительно другого на половину периода тактового сигнала. Чтение зарегистрированных кодов из обоих буферов также должно быть организовано по очереди: первый код читается из первого буфера, второй — из второго, третий — опять из первого, четвертый — из второго и т.д. Объем обоих буферов в данном случае складывается. Например, при организации каждого буфера 64Кх8 результирующий буфер будет иметь организацию 128Кх8.

Пользуясь этим же принципом, можно повысить частоту обработки входного сигнала с помощью АЦП не только вдвое, но и втрое, в четыре раза и т.д. Необходимо только согласовать во времени работу соответственно трех, четырех и т.д. АЦП, у каждого из которых должна быть своя буферная память.

Помимо упомянутых здесь АЦП последовательно и параллельного типов существуют еще и АЦП с промежуточным преобразованием. В них входной аналоговый сигнал с помощью аналогового интегратора преобразуется во временной интервал между цифровыми импульсами или в частоту следования цифровых импульсов. Выходной цифровой код, соответствующий входному аналоговому сигналу формируется в результате измерения длительности временного интервала или частоты следования импульсов (рис. 4.23). Если используется выходная частота, то такой АЦП называется "преобразователем напряжение—частота" (ПНЧ).

Такой подход позволяет с помощью сравнительно простых аппаратных средств получить высокую точность преобразования, не зависящую от многих параметров используемых компонентов и от

характеристик окружающей среды. Измерение временных интервалов и частоты следования импульсов осуществляется простейшими цифровыми схемами, примеры которых приведены в "Асинхронные и синхронно-асинхронные счетчики" , "Синхронные счетчики". Измерения эти могут осуществляться с высокой точностью вследствие того, что существует очень хороший временной эталон — кварцевый генератор. Отметим, что достоинством ПНЧ является также возможность простой передачи его выходного цифрового сигнала на большие расстояния.



Рис. 4.23. АЦП с промежуточным преобразованием.

В конце лекции надо еще раз отметить, что приведенные здесь схемы сильно упрощены. Для их практической реализации необходимо знание не только цифровой схемотехники, но и аналоговой и аналого-цифровой схемотехники, а также знание особенностей конкретных микросхем ЦАП и АЦП, что не является предметом данной книги. Однако рассмотренные ключевые принципы использования ЦАП и АЦП и их совместного включения с цифровыми схемами будут полезны любому разработчику.

## 5. ОСНОВЫ АРХИТЕКТУРЫ ЭВМ

Современные ЭВМ бывают самыми разными: большие, занимающие целые залы, маленькие, помещающиеся на столе, в портфеле и даже в кармане. Сегодня самым массовым видом ЭВМ являются персональные компьютеры.

Создание персонального компьютера (ПК) можно отнести к одному из самых значительных изобретений 20 века. ПК существенно изменил роль и значение вычислительной техники в жизни человека.

Определение «персональный» возникло потому, что человек получил возможность общаться с ЭВМ самостоятельно (персонально) без посредничества профессионала – программиста.

Персональные компьютеры используются сейчас повсеместно. Их основное назначение – выполнение рутинной работы: поиск информации, составление типовых форм документации, фиксация результатов исследования, подготовка текстов разного рода от простейших документов до издательской верстки и так далее.

Общедоступность и универсальность персонального компьютера обеспечивается за счет наличия следующих характеристик:

- «дружественность» интерфейса взаимодействия человека с компьютером, что позволяет работать на нем без специальной подготовки в компьютерной области

- малая стоимость

- небольшие габариты и отсутствие специальных требований к условиям окружающей среды

- открытость архитектуры

- большое количество программных средств для различных областей применения

- совместимость на программном и физическом уровне новых версий и моделей

- высокая надежность работы

Составные части, из которых состоит компьютер, называют модулями. Среди всех модулей выделяют основные модули, без которых работа компьютера невозможна, и остальные модули,

которые используются для решения различных задач: ввода и вывода графической информации, подключения к компьютерной сети и т.д.

Классические принципы построения ЭВМ были предложены Джон фон Нейманом (Г. Голдстейгб А. Беркс) в 1946 году и известны как "принципы фон Неймана":

1. Принцип использование двоичной системы представления данных

2. Принцип хранимой программы

3. Принцип последовательного выполнения операций

4. Принцип произвольного доступа к ячейкам оперативной памяти ЭВМ, построенные на этих принципах, имеют классическую архитектуру (архитектуру фон Неймана).

Архитектура ЭВМ – это её логическая организация, структура и ресурсы. Архитектура определяет принцип действия, информационные связи и взаимное соединение основных логических узлов ЭВМ:

- процессора;
- оперативного ЗУ (запоминающего устройства);
- внешних ЗУ;
- периферийных устройств.

Функции памяти:

- приём информации от других устройств;
- запоминание информации;
- передача информации по запросу в другие устройства машины.

Память делят на:

- основную:
- ОЗУ (оперативно запоминающее устройство);
- ПЗУ (постоянное запоминающее устройство);
- внешнюю (устройства внешней памяти позволяют длительно хранить информацию).

Носители внешней памяти:

- жесткие и гибкие магнитные диски
- лазерные диски (CD)

Прежде, чем использовать, диски форматируют на дорожки и секторы.

К функциям периферийных устройств относятся ввод и вывод информации.

Каждое устройство имеет набор характеристик, которые позволяют подобрать такую конфигурацию устройств, которая наилучшим образом подходит для решения определенного круга задач с помощью компьютера.

Функции процессора:

- обработка данных по заданной программе (выполнение над ними арифметических и логических операций)– функция АЛУ (арифметико-логического устройства);

- программное управление работой устройств ЭВМ – функция УУ (устройства управления).

В состав процессора входят также регистры (процессорная память) – ряд специальных запоминающих ячеек.

Регистры выполняют две функции:

- кратковременное хранение числа или команды;
- выполнение над ними некоторых операций.

Важнейшие регистры:

- счетчик команд (служит для автоматической выборки команд программы из последовательных ячеек памяти, в нем хранится адрес выполняемой команды);

- регистр команд и состояний (служит для хранения кода команды).

- Команда – это элементарная операция, которую должна выполнить ЭВМ.

Команда содержит:

- код выполняемой операции
- адреса операндов
- адрес размещения результата

Выполнение команды разбивается на следующие этапы:

- из ячейки памяти, адрес которой хранится в счетчике команд, выбирается команда (при этом содержимое счётчика команд увеличивается)

- команда передаётся в устройство управления (в регистр команд)
- устройство управления расшифровывает адресное поле команды
- по сигналам устройства управления операнды выбираются из памяти в АЛУ (в регистры операндов)

- УУ расшифровывает код операции и выдаёт сигнал АЛУ выполнить операцию

- результат операции остаётся в процессоре, либо возвращается в ОЗУ

## Структура ПК

Персональные компьютеры обычно состоят из следующих основных модулей

- мышь
- монитор

В системном блоке находятся все основные узлы компьютера:

- материнская плата;
- электронные схемы (процессор, контроллеры устройств и т.д.);
- блок питания;
- дисководы (накопители).

### 5.1. Понятие об архитектуре компьютера

Под архитектурой ЭВМ надо понимать ту совокупность характеристик, которая необходима пользователю. Это, прежде всего, основные устройства и блоки ЭВМ, а также структура связей между ними. «Архитектура – это наиболее общие принципы построения ЭВМ, реализующие программное управление работой и взаимодействием основных ее функциональных узлов».

Фон Нейман показал преимущества двоичной системы для технической реализации, удобство и простоту выполнения в ней арифметических и логических операций. Двоичное кодирование



данных по-прежнему составляет информационную основу любого современного компьютера.

Фон Нейман выдвинул основополагающие принципы логического устройства ЭВМ и предложил ее структуру, которая воспроизводилась в течение первых двух поколений ЭВМ (рис.5.1).

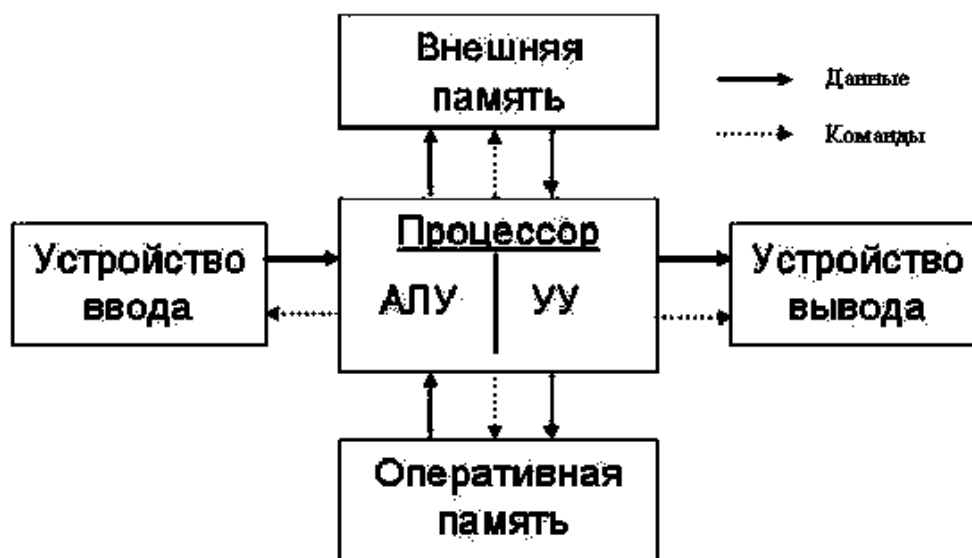


Рис.5.1 Архитектура машины фон Неймана.

Всю историю развития электронно-вычислительной техники принято делить на поколения. Смены поколений чаще всего были связаны со сменой элементной базы ЭВМ, с прогрессом электронной техники. Это всегда приводило к росту быстродействия и увеличению объема памяти. Кроме этого, как правило, происходили изменения в архитектуре ЭВМ, расширялся круг задач, решаемых на ЭВМ, менялся способ взаимодействия между пользователем и компьютером.

### *Поколения ЭВМ*

По настоящее время, сменилось 4 поколения ЭВМ:

1. 1946 г. создание машины ЭНИАК на электронных лампах. Запоминающие устройства (ЗУ) были построены на электронных лампах и линиях задержки.

2. 1960 год. ЭВМ построены на транзисторах, ЗУ на транзисторах, линиях задержки и ферритовых сердечниках.

3. 1965 год. ЭВМ построены на интегральных микросхемах (ИМС). ЗУ на ИМС.

4. 1970 г. Построены на основе больших интегральных схем (БИС) и сверх БИС (СБИС).

Устройства на элементах, изготовленных методами интегральной технологии, оказываются в десятки и сотни раз дешевле эквивалентных им устройств, выполненных на дискретных элементах. Резко улучшаются также практически все эксплуатационные и другие характеристики этих устройств (в том числе надежность, энергопотребление, массогабаритные параметры).

Естественным направлением снижения затрат на изготовление является применение микросхем повышенных степеней интеграции (или микросхем большей степени функциональной сложности). Однако со сложностью микросхем увеличивается их специализация, сужается область применения и увеличивая их стоимость. Поэтому применение специализированных БИС становится выгодным только при большой серийности изделий — например, БИС для электронных часов, телефонных аппаратов и калькуляторов и др.

Разрешение противоречия между высокой степенью интеграции и узкой специализацией БИС было найдено с созданием универсальных микросхем, логика, работы которых определяется не только внутренними связями, но и подаваемыми извне командами, т.е. программным управлением. Такая БИС получила название микропроцессора (МП БИС), так как она создана методом микроэлектронной технологии и способна, как и основное устройство ЭВМ — процессор, работать по изменяемой программе.

В настоящее время многие новые системы и средства, в том числе вычислительные, разрабатывают на базе МП.

Определим МП как обрабатывающее и управляющее устройство, выполненное в виде БИС и способное выполнять под программным управлением обработку информации, включая ввод и вывод информации, принятие решений, арифметические и логические операции. МП является центральной частью любой системы управления и обработки информации, но не самой системой.

Более подробное рассмотрение структуры МП проведем по функциональной схеме одного из распространенных отечественных МП КР580ВМ80А.

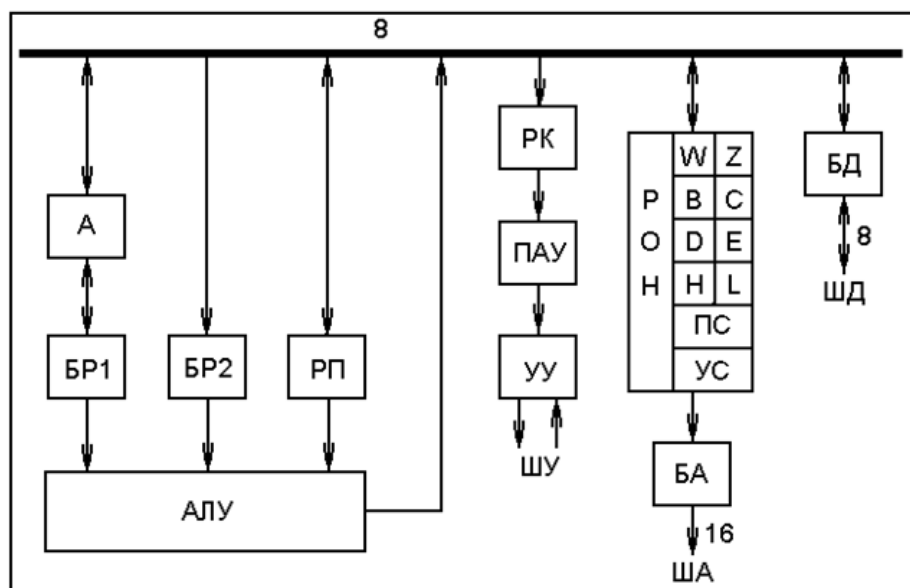


Рис.5.2. Структура микропроцессора КР580ВМ80А.

Микропроцессор имеет четыре шины: внутренняя информационная восьмиразрядная шина данных; внешняя шина данных для ввода-вывода информации и ввода рабочих программ; шина адреса — для адресации ЗУ. Для связи УУ с внешними устройствами введена шина управления. Внешняя информационная и адресная шины связаны с основными узлами МП через буферные регистры — буфер данных (БД) и буфер адреса (БА). К внутренней шине данных подключены регистры РОН, буферные регистры BR1, BR2, накопительный регистр — аккумулятор А, регистр признаков РП и АЛУ. Команды через БД поступают на регистр команд РК и далее на первичный автомат управления ПАУ, который расшифровывает их и включает УУ, выдающее соответствующую серию управляющих сигналов для всех узлов МП. Выборку команд при выполнении программы, размещенной в ЗУ, обеспечивает программный счетчик ПС (его состояние соответствует текущему адресу ЗУ, по которому хранится очередная команда программы). Элементом для выборки команд из ЗУ служит так называемый указатель стека УС. Указатель стека и ПС входят в состав РОН, их выходы в зависимости от

выполняемого режима подключаются к БА и адресом поступают в ЗУ. Регистры УС, ПС обеспечивают МП работу в дополнительных режимах, например в режиме обработки прерываний (останов основной программы по требованию внешних устройств) и в режиме обращения к подпрограммам. Такие режимы позволяют взаимодействовать МП с множеством подключаемых внешних устройств и использовать стандартные подпрограммы без их повторения при разработке основной программы.

Для пояснения работы МП рассмотрим, как выполняется операция сложения чисел А и В. Слагаемые последовательно через УВВ или ЗУ вводятся в регистры БР1 и БР2. Затем АЛУ выполняет сложение с размещением результата в одном из БР. В соответствии с программой этот результат используется в следующем шаге обработки, переводится в ЗУ, в один из регистров РОН, как информация или команда для исполнения. Все операции выполняются по командам от УУ в соответствии с заданной программой.

Для каждого типа МП предусматривается набор команд, которые может реализовать УУ. Число команд может быть от нескольких десятков до сотен. Помимо выполнения команд, возможности конкретного МП определяются его структурой, разрядностью данных (8, 16 и 32), объемом подключаемой памяти, числом подключаемых устройств (до нескольких сотен).

Приведенное описание данного МП носит чисто иллюстративный характер. Некоторые вопросы функционирования и структуры не были затронуты вообще. Однако перечисленного вполне достаточно для понимания сложности устройства МП.

## 5.2. Внешние устройства компьютера

Первые персональные ЭВМ имели одну внешнюю так называемую системную шину, к которой через слоты подключались адаптеры внешних устройств. Шины – это группы проводников, обеспечивающие взаимодействие различных устройств компьютера между собой. Отличают внутренние и внешние (по отношению к процессору) шины. Внутренние шины обеспечивают информационный обмен между узлами процессора. Внешние устройства связывают процессор с памятью и устройствами ввода-вывода (рис.5.3).

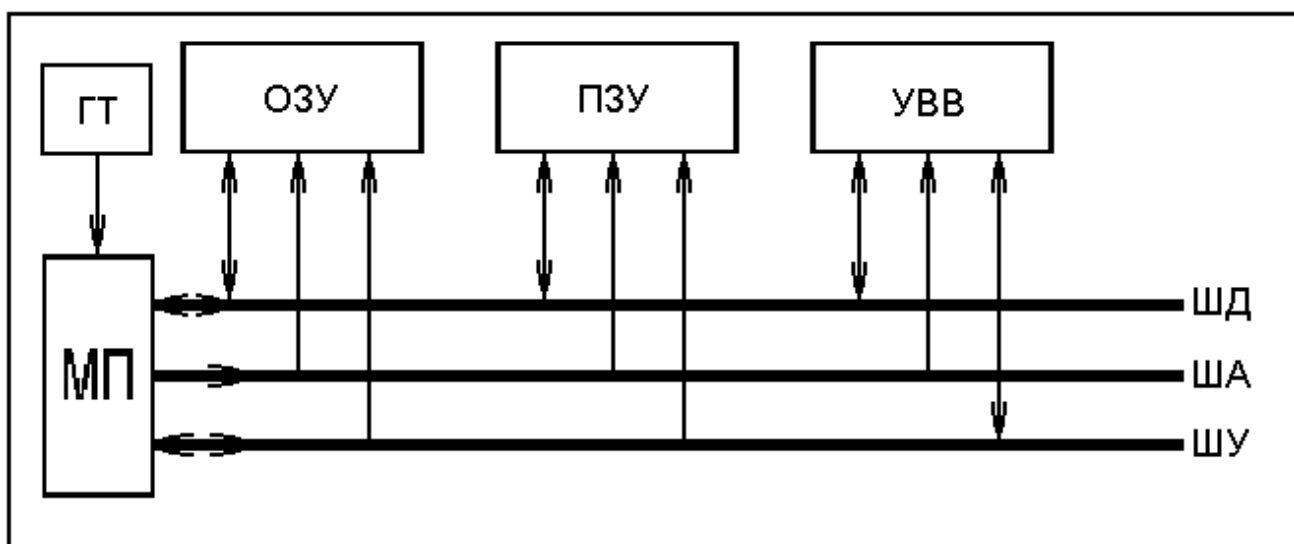


Рис.5.3. Архитектура микропроцессорных систем

Одна из основных задач ПЗУ обеспечить процедуру старта. В ПЗУ хранятся базовая система ввода/вывода BIOS, некоторые служебные программы и таблицы, например, начальный загрузчик, программы тестирования и т.п.

ОЗУ используется для хранения программы и необходимых ей областей данных. При выключении электропитания содержимое ОЗУ утрачивается.

Рост числа устройств и, как следствие, неспособность единой шины обслуживать их в приемлемое время привели к концепции так

называемых локальных шин. Существует несколько типов шин, что вызывает некоторые трудности, при переходе на новый тип процессора и материнской платы может сложиться ситуация, когда необходимая для вашего периферийного оборудования шина будет отсутствовать на новой модели материнской платы, что повлечет дополнительные расходы на модернизацию вполне работоспособного и удовлетворяющего вас устройства. В итоге архитектура современной МПС на базе Pentium II представлена на рис.5.4.

ISA (Industry Standard Architecture – архитектура промышленного стандарта) – основная шина на компьютерах типа PC AT (другое название – AT-Bus). Является расширением XT-Bus, разрядность – 16/24 (16 Мб), тактовая частота – 8 МГц, предельная пропускная способность – 5.55 Мб/с. Конструктив – 62-контактный разъем XT-Bus с прилегающим к нему 36-контактным разъемом расширения.

EISA (Enhanced ISA – расширенная ISA) – функциональное и конструктивное расширение ISA. Внешне разъемы имеют такой же вид. Разрядность – 32/32 (адресное пространство – 4 Гб), работает также на частоте 8 МГц. Предельная пропускная способность – 32 Мб/с. Поддерживает режим управления шиной со стороны любого из устройств на шине, имеет систему арбитража для управления доступом устройств к шине, позволяет автоматически настраивать параметры устройств, возможно разделение каналов IRQ и DMA.

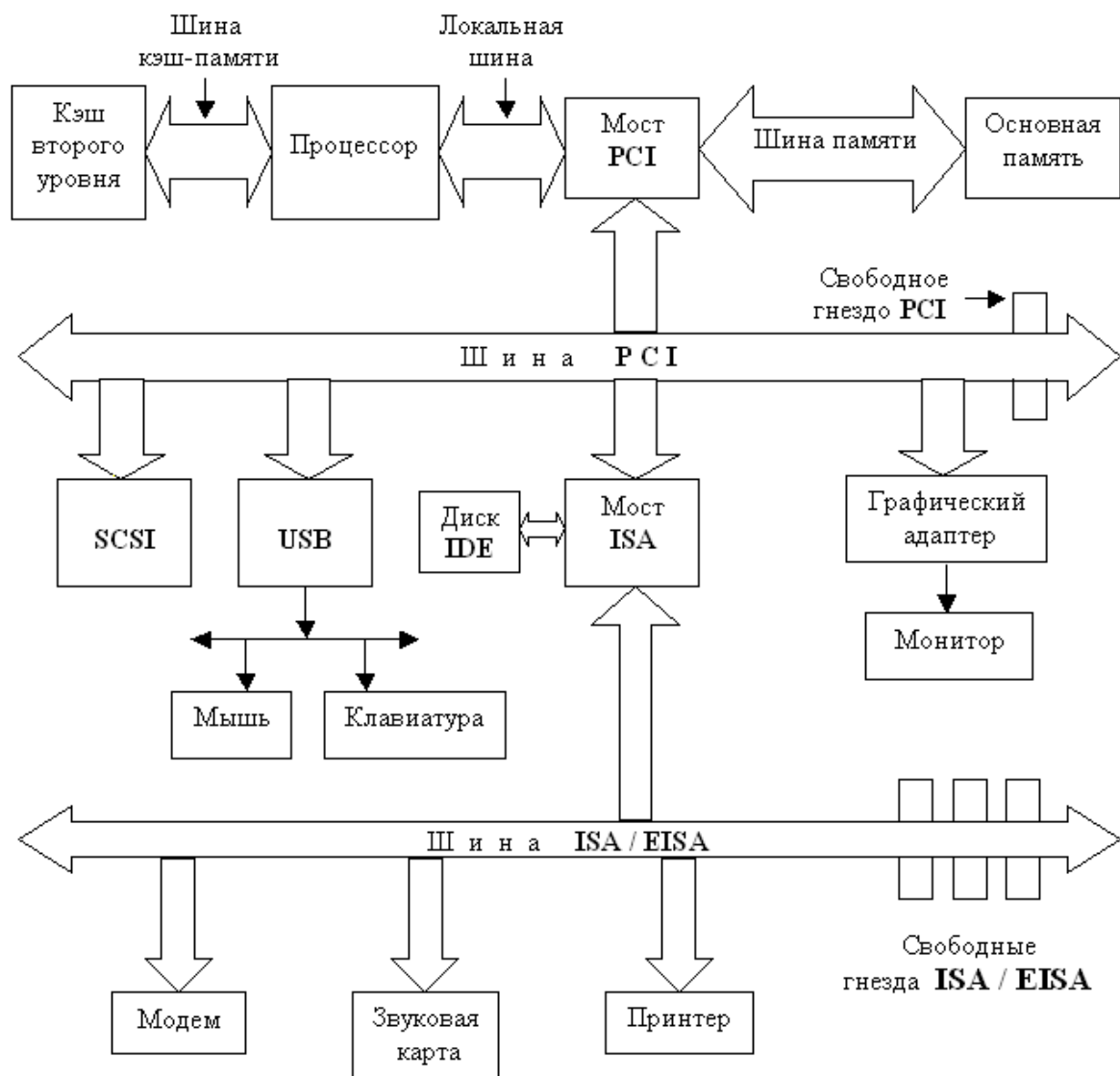


Рис.5.4. Архитектура современной МПС

IRQ (Interrupt ReQuest – запрос прерывания) – сигнал от одного из узлов компьютера, требующий внимания процессора к этому узлу. Возникает при наступлении какого-либо события (например, нажатии клавиши, завершении операции чтения/записи на диске и т.п.). На PC AT предусмотрено 15 (на XT – 8) линий IRQ, часть которых используется внутренними контроллерами системной платы, а остальные заняты стандартными адаптерами либо не используются:

- 0 – системный таймер
- 1 – контроллер клавиатуры
- 2 – сигнал возврата по кадру (EGA/VGA), на AT соединен с IRQ 9
- 3 – обычно COM2/COM4
- 4 – обычно COM1/COM3

- 5 – контроллер HDD (XT), обычно свободен на AT
- 6 – контроллер FDD
- 7 – LPT1, многими LPT-контроллерами не используется
- 8 – часы реального времени с автономным питанием (RTC)
- 9 – параллельна IRQ 2
- 10 – не используется
- 11 – не используется
- 12 – обычно контроллер мыши типа PS/2
- 13 – математический сопроцессор
- 14 – обычно контроллер IDE HDD (первый канал)
- 15 – обычно контроллер IDE HDD (второй канал)

DMA (Direct Memory Access – прямой доступ к памяти) – способ обмена данными между внешним устройством и памятью без участия процессора, что может заметно снизить нагрузку на процессор и повысить общую производительность системы. Режим DMA позволяет освободить процессор от рутинной пересылки данных между внешними устройствами и памятью, отдав эту работу контроллеру DMA; процессор в это время может обрабатывать другие данные или другую задачу в многозадачной системе. На PC AT есть 7 (на XT – 4) независимых каналов контроллера DMA:

- 0 – регенерация памяти на некоторых платах
- 1 – не используется
- 2 – контроллер FDD
- 3 – контроллер HDD на XT, на AT не используется
- 5 – не используется
- 6 – не используется
- 7 – не используется

Каналы 0-3 – восьмиразрядные, каналы 5-7 – шестнадцатиразрядные.

С учетом этого, новые адаптеры следует настраивать прежде всего на полностью свободные каналы IRQ (10, 11) и DMA (1, 5-7), а затем – на свободные в конкретной системе. Возможность использования одного IRQ несколькими адаптерами зависит от типа шины и требует поддержки со стороны драйверов этих адаптеров. Использование



разными адаптерами одного канала DMA в принципе возможно, но связано со множеством проблем и потому не рекомендуется.

В 1990 году появилась локальная шина PCI (Peripheral Component Interconnect- соединение внешних компонент). Она была разработана в основном фирмой Intel для процессора Pentium. Шина PCI является промежуточным звеном между локальной шиной процессора и шиной ISA/EISA. Соединение ее с шиной центрального процессора осуществляется через мост PCI (Host Bridge). Мост согласует шину центрального процессора с шиной PCI, обеспечивая в дальнейшем возможность ее стыковки с процессорами следующих поколений. Разрядность – 32/32 (расширенный вариант – 64/64), тактовая частота – до 33 МГц (PCI 2.1 – до 66 МГц), пропускная способность – до 528 Мб/с.

Предусмотрено автоматическое конфигурирование съемных плат шины PCI. При этом плата расширения хранит информацию о периферийном устройстве. По ней процессор определяет, какое устройство подключилось к системе.

Технологию, которая сводит к минимуму усилия по подключению новой аппаратуры называют PnP.

Plug And Play – "вставь и играйся". PnP-карты не имеют переключателей конфигурации или особых программ настройки; вместо этого общий для компьютера PnP-диспетчер (отдельная программа либо часть BIOS или ОС) сам находит каждую из них и настраивает на соответствующие адреса, линии IRQ, DMA, области памяти, предотвращая совпадения и конфликты.

Система SCSI (Small Computer System Interface) (интерфейс, универсальный контроллер, шина расширения). Это высокопроизводительный интерфейс, допускающий подключение до 8 устройств, одно из которых – контроллер SCSI (Host Adapter), а 7 других предназначены для периферии (винчестеры, CD-ROM, сканеры и другое оборудование специализированных вычислительных систем). Скорость обмена – до 40 Мбайт/с.

В этой системе использованы аппаратно-независимые стандарты, которые освобождают изготовителей периферийных устройств от необходимости разработки собственных контроллеров.

Шина USB (Universal Serial Bus – универсальная последовательная магистраль) – новый интерфейс для подключения к шинам ISA и PCI низкоскоростных периферийных устройств. Предусматривает подключение до 127 внешних устройств к одному USB-каналу. Обмен по интерфейсу – пакетный, скорость обмена – 12 Мбит/с.

Магистральный интерфейс AGP (Accelerated Graphic Port – ускоренный графический порт)- интерфейс для подключения видеоадаптера к отдельной магистрали AGP, имеющей выход непосредственно на системную память. В системной памяти размещаются преимущественно параметры трехмерных объектов требующие быстрого доступа со стороны как процессора, так и видеоадаптера. Наряду с повышением пропускной способности шины применяются меры по уменьшению потока данных, передаваемых по шине при графических построениях. Для этого графические адаптеры снабжаются акселераторами, а также увеличивается объем их буферной памяти (видеопамяти). При этом высокоэффективный поток данных в основном циркулирует внутри графической карты и только в самых сложных построениях выходит на шину PCI.

AGP – это новый стандарт подключения графических адаптеров, по составу сигналов напоминающий PCI. Его основные особенности:

- конвейеризация обращений к памяти (может ставить в очередь до 256 запросов);

- двоенная передача данных при частоте 66 МГц обеспечивает пропускную способность до 532 Мбайт/с.

На что следует обратить внимание при покупке системной платы?

Прежде всего – на ее внешний вид. Детали должны быть установлены ровно и аккуратно, пайка – блестящей, ровной и однородной. Криво установленные детали, "пузыри" припоя и не пропаянные выводы обычно встречаются на платах китайского производства и говорят об общем качестве работы. Если плата заметно выгнута в одну сторону – есть вероятность наличия микротрещин в

дорожках или кристаллах микросхем. Также могут быть неровно впаяны разъемы для SIMM, что грозит плохим контактом или вообще невозможностью вставить некоторые модули.

Желательно, чтобы на микросхемах Chipset'a были собственные обозначения (OPTi895, SiS496, UMC8881 и т.п.). Надписи типа "PC Chips" обычно наносятся на немаркированные микросхемы, полученные окольными путями – здесь высока вероятность брака. Вообще, чем больше технических обозначений – тем лучше. Не приветствуются наклейки, особенно с надписями типа "Write Back" вместо названий (Под термином Write Back в основном понимается отложенная запись, однако это может означать и буферизованную сквозную). При сомнениях можно снять наклейку, чтобы посмотреть настоящую маркировку чипа.

На качество платы может косвенно указывать ее упаковка и документация. Хорошие платы обычно имеют названия, поставляются в коробках и снабжаются подробной документацией в хорошо оформленной книжке. Однако бывает и так, что безродная плата с невзрачной книжечкой по совокупности характеристик оказывается лучше, чем фирменная – последнее слово должно быть за тестированием.

Chip Set – Это одна или несколько микросхем, специально разработанных для "обвязки" микропроцессора. Они содержат в себе контроллеры прерываний, прямого доступа к памяти, памятью и шиной – все те компоненты, которые в оригинальной IBM PC были собраны на отдельных микросхемах. В последних разработках в состав микросхем наборов для интегрированных плат стали включаться и контроллеры внешних устройств.

Внешне микросхемы Chipset'a выглядят, как самые большие после процессора. Название набора обычно происходит от маркировки основной микросхемы – OPTi495SLC, SiS471, UMC491, i82C437VX и т.п. При этом используется только код микросхемы внутри серии: например, полное наименование SiS471 – SiS85C471.

Тип набора в основном определяет функциональные возможности платы: типы поддерживаемых процессоров, структура/объем кэша,

возможные сочетания типов и объемов модулей памяти, поддержка режимов энергосбережения, возможность программной настройки параметров и т.п. На одном и том же наборе может выпускаться несколько моделей системных плат, от простейших до довольно сложных с интегрированными контроллерами портов, дисков, видео и т.п.

На материнской плате расположены разъемы процессора, оперативной памяти, шинные разъемы (слоты).

Что такое IR Connector?

Infrared Connector – разъем для инфракрасного излучателя/приемника. Подключен к одному из встроенных СОМ-портов (обычно – СОМ2) и позволяет установить беспроводную связь с любым устройством, снабженным подобным излучателем и приемником. Работает по тому же принципу, что и пульты управления бытовой радиоаппаратурой.

Что такое BIOS и зачем он нужен?

Это Basic Input/Output System – основная система ввода/вывода, зашитая в ПЗУ (отсюда название ROM BIOS). Она представляет собой набор программ проверки и обслуживания аппаратуры компьютера, и выполняет роль посредника между DOS и аппаратурой. BIOS получает управление при включении и сбросе системной платы, тестирует саму плату и основные блоки компьютера – видеоадаптер, клавиатуру, контроллеры дисков и портов ввода/вывода, настраивает Chipset платы и загружает внешнюю операционную систему. При работе под DOS/Windows BIOS управляет основными устройствами, при работе под WinNT BIOS практически не используется, выполняя лишь начальную проверку и настройку.

Раньше BIOS зашивался в однократно программируемые ПЗУ либо в ПЗУ с ультрафиолетовым стиранием; сейчас в основном выпускаются платы с электрически перепрограммируемыми ПЗУ (Flash ROM), которые допускают перешивку BIOS средствами самой платы. Это позволяет исправлять заводские ошибки в BIOS, изменять заводские умолчания, программировать собственные экранные заставки и т.п.

### 5.3. Микросхемы памяти

Из микросхем памяти (RAM – Random Access Memory, память с произвольным доступом) используется два основных типа: статическая (SRAM – Static RAM) и динамическая (DRAM – Dynamic RAM).

В статической памяти элементы (ячейки) построены на различных вариантах триггеров – схем с двумя устойчивыми состояниями. После записи бита в такую ячейку она может пребывать в этом состоянии столь угодно долго – необходимо только наличие питания. Ячейки статической памяти имеют малое время срабатывания, однако микросхемы на их основе имеют низкую удельную плотность данных (порядка единиц Мбит на корпус) и высокое энергопотребление. Поэтому статическая память используется в основном в качестве буферной (кэш-память).

В динамической памяти ячейки построены на основе областей с накоплением зарядов, занимающих гораздо меньшую площадь, нежели триггеры, и практически не потребляющих энергии при хранении. При записи бита в такую ячейку в ней формируется электрический заряд, который сохраняется в течение нескольких миллисекунд; для постоянного сохранения заряда ячейки необходимо регенерировать – перезаписывать содержимое для восстановления зарядов. Ячейки динамической памяти имеют большее время срабатывания, но большую удельную плотность (порядка десятков Мбит на корпус) и меньшее энергопотребление. Динамическая память используется в качестве основной.

Обозначение корпусов микросхем и типов модулей памяти.

DIP (Dual In line Package – корпус с двумя рядами выводов) – классические микросхемы, применявшиеся в блоках основной памяти ХТ и ранних АТ, а сейчас – в блоках кэш-памяти.

SIP (Single In line Package – корпус с одним рядом выводов) – микросхема с одним рядом выводов, устанавливаемая вертикально.

SIPP (Single In line Pinned Package – модуль с одним рядом проволочных выводов) – модуль памяти, вставляемый в панель наподобие микросхем DIP/SIP; применялся в ранних АТ.

SIMM (Single In line Memory Module – модуль памяти с одним рядом контактов) – модуль памяти, вставляемый в зажимающий разъем; применяется во всех современных платах, а также во многих адаптерах, принтерах и прочих устройствах. SIMM имеет контакты с двух сторон модуля, но все они соединены между собой, образуя как бы один ряд контактов.

DIMM (Dual In line Memory Module – модуль памяти с двумя рядами контактов) – модуль памяти, похожий на SIMM, но с отдельными контактами (обычно 2 x 84), за счет чего увеличивается разрядность или число банков памяти в модуле.

Микросхемы памяти имеют четыре основные характеристики – тип, объем, структуру и время доступа. Тип обозначает статическую или динамическую память, объем показывает общую емкость микросхемы, а структура – количество ячеек памяти и разрядность каждой ячейки.

Например, 28/32-выводные DIP-микросхемы SRAM имеют восьмиразрядную структуру (8к\*8, 16к\*8, 32к\*8, 64к\*8, 128к\*8), и кэш для 486 объемом 256 кб будет состоять из восьми микросхем 32к\*8 или четырех микросхем 64к\*8. Две микросхемы по 128к\*8 поставить уже нельзя, так как нужна 32-разрядная шина данных, что могут дать только четыре параллельных микросхемы.

72-контактные SIMM имеют 32-разрядную структуру и могут ставиться с 486 по одному, а с Pentium – по два. 168-контактные DIMM имеют 64-разрядную структуры и ставятся в Pentium по одному. Установка модулей памяти или микросхем кэша в количестве больше минимального позволяет некоторым платам ускорить работу с ними, используя принцип расслоения (Interleave – чередование).

Время доступа характеризует скорость работы микросхемы и обычно указывается в наносекундах через тире в конце наименования. На более медленных динамических микросхемах могут указываться только первые цифры (-7 вместо -70, -15 вместо -150), на более

быстрых статических "-15" или "-20" обозначают реальное время доступа к ячейке. Часто на микросхемах указывается минимальное из всех возможных времен доступа – например, распространена маркировка 70 нс EDO DRAM, как 50, или 60 нс – как 45, хотя такой цикл достижим только в блочном режиме, а в одиночном режиме микросхема по-прежнему срабатывает за 70 или 60 нс. Аналогичная ситуация имеет место в маркировке PB SRAM: 6 нс вместо 12, и 7 – вместо 15. Микросхемы SDRAM обычно маркируются временем доступа в блочном режиме (10 или 12 нс).

Микросхемы EDO часто (но далеко не всегда) имеют в обозначении "некруглые" числа: например, 53C400 – обычная DRAM, 53C408 – EDO DRAM.

EDO (Extended Data Out – расширенное время удержания данных на выходе) фактически представляют собой обычные микросхемы, на выходе которых установлены регистры-защелки данных. При страничном обмене такие микросхемы работают в режиме простого конвейера: удерживают на выходах данных содержимое последней выбранной ячейки, в то время как на их входы уже подается адрес следующей выбираемой ячейки. Это позволяет примерно на 15% ускорить процесс считывания последовательных массивов данных. При случайной адресации такая память ничем не отличается от обычной.

SDRAM (Synchronous DRAM – синхронная динамическая память) – память с синхронным доступом, работающая быстрее обычной асинхронной. Помимо синхронного метода доступа, SDRAM использует внутреннее разделение массива памяти на два независимых банка, что позволяет совмещать выборку из одного банка с установкой адреса в другом банке. Основная выгода от использования SDRAM состоит в поддержке последовательного доступа в синхронном режиме, где не требуется дополнительных тактов ожидания. При случайном доступе SDRAM работает практически с той же скоростью, что и EDO.

Наиболее распространены сегодня модули DDR SDRAM. Аббревиатура DDR расшифровывается как double data rate — «двойная

скорость передачи данных»: память этого типа, как и современные процессоры, способна «удваивать» оригинальную частоту шины памяти. Например, память DDR-333 работает на частоте шины всего в 166 МГц.

Увы, даже этой скорости сегодня оказывается недостаточно: напомним, что последние версии чипсетов под процессоры Pentium 4 поддерживают частоту системной шины в 533 МГц, — в перспективе же ожидается ее увеличение еще, как минимум, вдвое! Память сегодня становится тем самым «узким местом», которое может свести на нет все преимущества мощного процессора.

Что такое кэш и зачем он нужен?

Cache (запас) обозначает быстродействующую буферную память между процессором и основной памятью. Кэш служит для частичной компенсации разницы в скорости процессора и основной памяти – туда попадают наиболее часто используемые данные. Когда процессор первый раз обращается к ячейке памяти, ее содержимое параллельно копируется в кэш, и в случае повторного обращения в скором времени может быть с гораздо большей скоростью выбрано из кэша.

Как лучше выбрать частоту платы и внутренний множитель процессора?

Если одну и ту же внутреннюю частоту процессора можно задать несколькими способами, то на более высокой входной частоте (на которой работает сама системная плата) обычно достигается более высокая производительность. Чаще всего это делается на недокументированных частотах – 75 или 83 МГц. Например, при работе программ, интенсивно пересылающих данные между памятью и шиной (анимация, игры, обработка больших баз данных и т.п.) конфигурация  $75 \times 2.5 = 187$  превосходит конфигурацию  $66 \times 3 = 200$ , а  $83 \times 2.5 = 208$  превосходит  $75 \times 3 = 225$ . Однако выигрыш будет только в том случае, если системная плата и PCI- устройства стабильно работают на повышенной частоте; если, например, на ней не успевает память или внешний кэш, то придется вводить дополнительные такты ожидания, которые могут свести на нет преимущество высокой частоты. Кроме этого, может потребоваться понижение на ступень



скорости РЮ в связи с тем, что временные параметры РЮ вычисляются из системной частоты и при ее завышении могут выйти за допустимые пределы.

Что происходит при замыкании контактов разъема Turbo?

В компьютерах Turbo XT и ранних AT-286 кнопка Turbo была предназначена для повышения тактовой частоты процессора сверх номинальной с целью ускорения его работы; при этом устойчивая работа на этой частоте не гарантировалась. На более поздних и быстрых AT-286 и ранних 386 она, наоборот, снижала частоту, чтобы приблизить быстродействие к PC XT – многие старые программы пользовались для измерения времени скоростными параметрами XT, отчего на AT начинали работать с ошибками.

В начале 90-х годов, на последних AT-286 и 386/486 был введен другой способ управления скоростью: частота системного генератора была постоянной, а при замыкании контактов Turbo принудительно замедлялась работа с внешним кэшем и памятью. Для большинства программ это не давало заметного эффекта, поскольку сам процессор и его внутренний кэш продолжали работать с обычной скоростью.

На многих современных платах для Pentium контакты Turbo выполняют функцию Suspend – приостановки работы платы и внешних устройств путем перехода в режим энергосбережения (Green Mode). Suspend обычно может быть запрещен опцией в Setup – тогда кнопка Turbo не влияет на работу системы. На некоторых новых платах замыкание контактов снова понижает частоту системного генератора.

#### **5.4. Порты ввода/вывода**

##### *Параллельный порт*

Параллельные порты предназначены для обмена информацией микропроцессора с внешними устройствами, при этом в качестве внешнего устройства может использоваться другой компьютер. Параллельные порты позволяют согласовывать низкую скорость работы внешнего устройства и высокую скорость работы системной шины микропроцессора. С точки зрения внешнего устройства порт

представляет собой обычный источник или приемник информации со стандартными цифровыми логическими уровнями (обычно ТТЛ), а с точки зрения микропроцессора — это ячейка памяти, в которую можно записывать данные или в которой сама собой появляется информация.

В качестве внешнего устройства может служить любой объект управления или источник информации (различные кнопки, датчики, микросхемы приемников, синтезаторов частот, дополнительной памяти, исполнительные механизмы, двигатели, реле и т.д.).

В зависимости от направления передачи данных параллельные порты называются портами ввода, вывода или портами ввода вывода.

Структурная схема порта ввода приведена на рис 5.5.

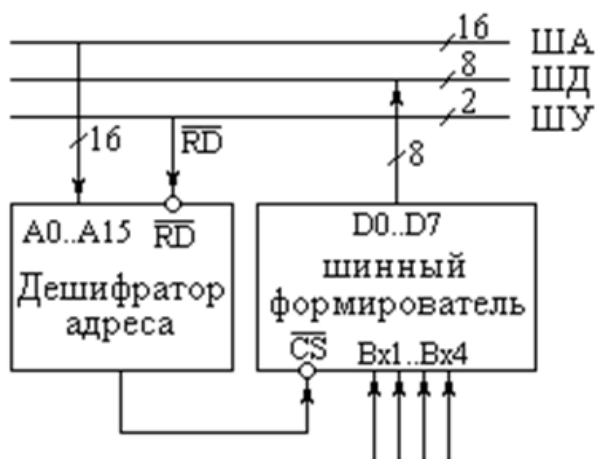


Рис.5.5. Структурная схема порта ввода.

В качестве порта ввода может быть использована схема с открытым коллектором или с третьим (Z) состоянием. В настоящее время обычно используются схемы с третьим состоянием. Параллельное соединение таких схем называется шинным формирователем. Из порта ввода возможно только чтение информации.

Выход шинного формирователя подключается к внутренней шине. Значение сигнала с внешнего вывода порта считывается по сигналу "RD".

Для отображения этого шинного формирователя только в одну ячейку памяти адресного пространства микропроцессорного

устройства в составе порта ввода-вывода всегда присутствует дешифратор адреса.

Так как с точки зрения программиста эта ячейка памяти ничем не отличается от регистра данных порта вывода, то по аналогии она называется регистром данных порта ввода.

Параллельные порты, предназначенные для обмена данными между компьютерами, или компьютером и принтером, устроены несколько иначе. Основным отличием обмена данными между компьютерами или контроллерами от обмена данными между компьютером и простым внешним устройством является большой объём передаваемых данных. В этом случае недостаточно выдачи на выход порта одного или даже нескольких байт информации, поэтому приходится передавать данные последовательно байт за байтом через один и тот же параллельный порт. Байты необходимо каким-либо образом отличать друг от друга, поэтому вводится специальный сигнал синхронизации CLK, который позволяет отличать один байт от другого. Для формирования такого сигнала можно воспользоваться вторым параллельным портом, и получить его программным способом, но обычно этот сигнал формируется аппаратно из сигнала WR# при записи очередного байта в параллельный порт вывода. Временная диаграмма обмена данными через параллельный порт приведена на рисунке 5.6.

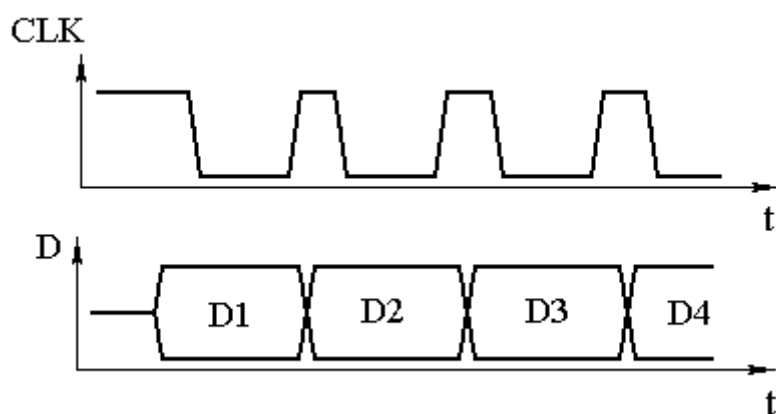


Рис. 5.6. Временная диаграмма работы параллельного порта

Параллельные интерфейсы характеризуются тем, что в них для передачи бит в слове используются отдельные сигнальные линии, и

биты передаются одновременно. Параллельные интерфейсы используют логические уровни ТТЛ (транзисторно-транзисторной логики), что ограничивает длину кабеля из-за невысокой помехозащищенности ТТЛ-интерфейса. Гальваническая развязка отсутствует. Параллельные интерфейсы используют для подключения принтеров. Передача данных может быть как однонаправленной (Centronics), так и двунаправленной (Bitronics). Иногда параллельный интерфейс используют для связи между двумя компьютерами – получается сеть, "сделанная на коленке" (LapLink). Ниже будут рассмотрены протоколы интерфейсов Centronics, стандарт IEEE 1284, а также реализующие их порты PC.

### *Интерфейс Centronics и LPT-порт*

Для подключения принтера по интерфейсу Centronics в PC был введен порт параллельного интерфейса – так возникло название LPT-порт (Line PrinTer – построчный принтер). Хотя сейчас через этот порт подключаются не только построчные принтеры, название "LPT" осталось.

### *Интерфейс Centronics*

Интерфейс Centronics поддерживается принтерами с параллельным интерфейсом. Его отечественным аналогом является интерфейс ИРПР-М.

Традиционный, он же стандартный, LPT-порт SPP (Standard Parallel Port) является однонаправленным портом, через который программно реализуется протокол обмена Centronics. Порт вырабатывает аппаратное прерывание по импульсу на входе Ack#. Сигналы порта выводятся на разъем DB-25S (розетка), установленный непосредственно на плате адаптера (или системной плате) или соединяемый с ней плоским шлейфом. Название и назначение сигналов разъема порта (табл. 5.1) соответствуют интерфейсу Centronics.

Таблица 5.1.

## Название и назначение сигналов разъема порта Centronics.

Сигнал	I/O*	Контакт	Назначение
Strobe#	I	1	Строб данных. Данные фиксируются по низкому уровню сигнала
Data [0:7]	I	2-9	Линии данных. Data 0 (контакт 2) – младший бит
Ack#	O	10	Acknowledge – импульс подтверждения приема байта (запрос на прием следующего). Может использоваться для формирования запроса прерывания
Busy	O	11	Занято. Прием данных возможен только при низком уровне сигнала
PaperEnd	O	12	Высокий уровень сигнализирует о конце бумаги
Select	O	13	Сигнализирует о включении принтера (обычно в принтере соединяется резистором с цепью +5 В)
AutoLF#	I	14	Автоматический перевод строки. При низком уровне принтер, получив символ CR (Carriage Return – возврат каретки), автоматически выполняет и функцию Lf(Line Feed – перевод строки)
Error#	O	32	Ошибка: конец бумаги, состояние OFF-Line или внутренняя ошибка принтера
Init#	I	31	Инициализация (сброс в режим параметров умолчания, возврат к началу строки)
Select In#	I	36	Выбор принтера (низким уровнем). При высоком уровне принтер не воспринимает остальные сигналы интерфейса
GND	-	19-30, 33	Общий провод интерфейса



Рис. 5.7. Передача данных по протоколу Centronics.

Для того, чтобы вывести символ на принтер, программа вначале должна убедиться, что уровень сигнала на линии BUSY (бит 7 порта 379h) равен 0, а уровень сигнала на линии ACK (бит 6 порта 379h) – единице. После этого следует установить код выводимого символа на линиях DATA (порт 378h).

Затем не ранее, чем через 0,5 мкс линию STROBE (бит 0 порта 37Ah) необходимо перевести в состояние логического 0. При этом выводимый символ запишется во внутренний буфер принтера. Уровень логического нуля необходимо удерживать в течение как минимум 0,5 мкс. Это время нужно для того, чтобы символ записался в буфер принтера. После истечения интервала времени линию STROBE нужно опять перевести в состояние логической единицы.

После того, как программа установит линию STROBE в состояние логического нуля, выходная линия принтера BUSY устанавливается в единицу, сигнализируя о том, что принтер занят обработкой полученного символа и временно не может принимать другие символы.

Когда принтер полностью обработает выведенный символ, линия ACK перейдет в состояние 0. Приблизительно через 5 мкс после этого линия BUSY также перейдет в состояние 0.

Еще через 5 мкс линия ACK примет состояние 1. Теперь принтер готов принят следующий символ распечатываемых данных

Последовательные порты предназначены для обмена информацией микропроцессоров между собой, а также для связи с устройствами в которых критично количество соединительных проводов. В настоящее время широко используются два вида последовательных портов:

1. синхронные последовательные порты;
2. асинхронные последовательные порты.

### 5.5. Синхронные последовательные порты

При рассмотрении работы параллельного порта в режиме обмена данными с другим компьютером или принтером уже рассматривался режим последовательной передачи байтов. В последовательном порту

режим последовательной передачи применяется не только к байтам, но и к отдельным битам внутри байта. В этом случае для передачи данных достаточно только одного провода. Передаваемая и принимаемая информация обычно представляется в виде однобайтовых или многобайтовых слов. Вес каждого бита в слове различен, поэтому кроме битовой синхронизации, аналогичной байтовой синхронизации для параллельного порта, требуется кадровая синхронизация. Кадровая синхронизация позволяет однозначно определять номер каждого бита в передаваемом слове. Временная диаграмма передачи кадра по синхронному последовательному порту приведена на рисунке 5.8.

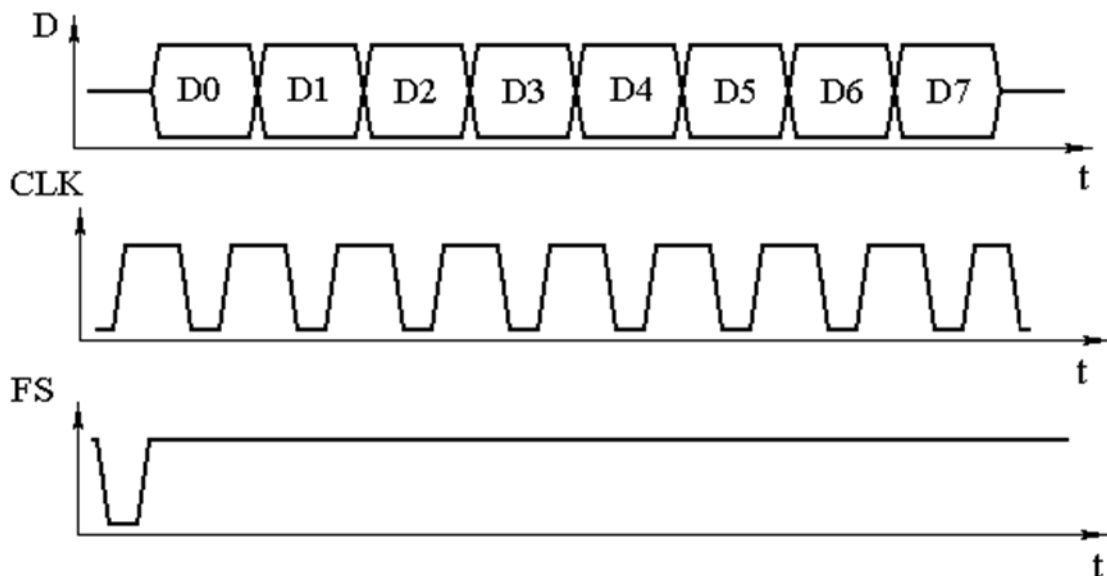


Рис. 5.8. Временная диаграмма передачи одного кадра двоичной информации по последовательному порту.

Временная диаграмма, приведённая на рисунке 5.8, применяется в синхронных последовательных портах, которые используются чаще всего в сигнальных процессорах для обмена информацией с кодеками речи, аналого-цифровыми и цифро-аналоговыми преобразователями. На приведённой временной диаграмме показаны два сигнала синхронизации: сигнал тактовой синхронизации CLK и сигнал кадровой синхронизации FS. Сигнал кадровой синхронизации формируется аппаратно из сигнала WR# при записи очередного байта в параллельный порт вывода. Полярность сигналов синхронизации

зависит от конкретного типа применяемых микросхем, поэтому в большинстве сигнальных процессоров возможна настройка полярности сигналов синхронизации.

Упрощённая схема синхронного последовательного порта приведена на рис. 5.9. На этой схеме видно, что в состав последовательного порта входит параллельный порт, который позволяет подключаться к системной шине микропроцессора. Для преобразования параллельного кода, который поступает из системной шины в последовательный, используется сдвиговый регистр. При обращении центрального процессора к последовательному порту вырабатывается сигнал записи в последовательный порт, который подаётся на вход параллельной записи  $V$  универсального регистра. Этот же сигнал используется в качестве сигнала кадровой синхронизации  $FS$ . Сигнал тактовой синхронизации  $CLK$ , вырабатываемый отдельным генератором, подаётся на вход последовательного сдвига  $C$  универсального регистра порта.

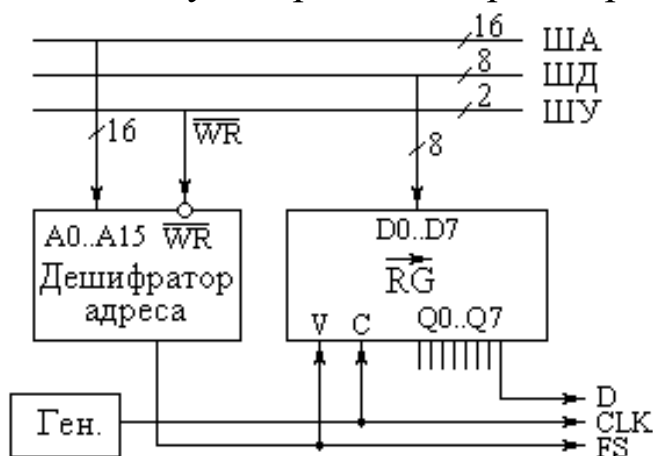


Рис. 5.9. Упрощённая схема синхронного последовательного порта

Количество передаваемых в одном кадре бит может меняться от восьми до тридцатидвух.

В рассмотренной схеме синхронного последовательного порта на приёмном конце необходимо подсчитывать количество тактовых импульсов, прошедшее после импульса кадровой синхронизации. Кроме того, в таком синхронном последовательном порту информация передаётся непрерывно, что, конечно, удобно для устройств с



непрерывным потоком информации, как, например, в кодеках речи. Но существуют устройства, к которым необходимо обращаться только периодически, как, например, синтезаторы частоты, микросхемы приёмников, блоков цветности телевизоров, микросхем памяти данных и многие другие устройства. В этих случаях используются другие виды синхронных последовательных портов такие как SPI порт и I2C шина. Временная диаграмма SPI интерфейса приведена на рис.5.10.

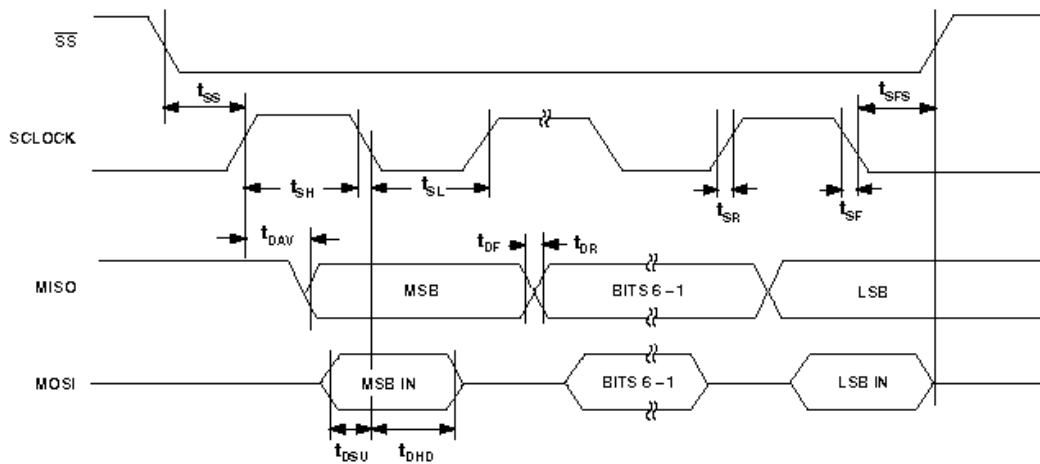


Рис. 5.10. Временная диаграмма SPI интерфейса

Основное отличие этого интерфейса от приведённого выше, заключается в том, что сигнал тактовой синхронизации передаётся только в момент действия импульса кадровой синхронизации. Активный уровень сигнала кадровой синхронизации длится до окончания передачи последнего бита в передаваемом кадре. По одним и тем же линиям передачи данных MISO (вход для главного, выход для ведомого) и MOSI (выход для главного, вход для ведомого) может передаваться информация к совершенно различным микросхемам. Выбор, для какой из микросхем предназначена информация, производится сигналом SS (выбор ведомого). В SPI интерфейсе в приёмнике не требуется счётчик тактовых импульсов. Запись принятой информации производится по окончанию кадрового импульса.

Если в устройстве используется несколько микросхем, то количество линий выбора ведомого становится значительным,

поэтому в таких случаях используется ещё один вид синхронного последовательного интерфейса: I2C шина. Временная диаграмма этого интерфейса приведена на рис. 5.11. В I2C шине приём и передача данных, а также передача адреса микросхемы и адреса регистра внутри микросхемы, к которому осуществляется обращение, производится по одному и тому же проводу. Для подключения к этому проводу используются микросхемы с открытым коллектором. Нагрузкой для всех микросхем, подключенных к линии SDA служит внешний резистор. Естественно, что скорость передачи данных по такому порту будет ниже, по сравнению с SPI портом. Тактовая синхронизация в I2C шине передаётся по линии SCL. Начало работы с микросхемой обозначается особой комбинацией сигналов SDA и SCL, которая называется условием старта. Эта же комбинация одновременно осуществляет кадровую синхронизацию. Завершение работы с микросхемой обозначается ещё одной комбинацией сигналов SDA и SCL. В качестве примера микросхем, использующих интерфейс I2C можно назвать микросхемы EEPROM серии 24сXX.

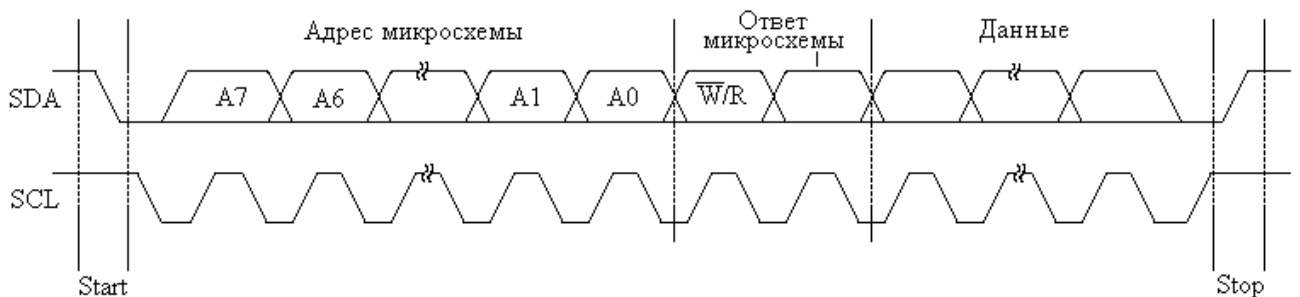


Рис. 5.11. Временная диаграмма I2C интерфейса.

## Литература

1. Бабичев, Ю. Е. Электротехника, электроника и схемотехника ЭВМ. Линейные электрические цепи: учебное пособие / Ю. Е. Бабичев. — Москва : МИСИС, 2017. — 70 с.
2. Бойт К. Цифровая электроника (пер. с нем. Ташлицкого М.М.), Серия Мир электроники Издательство Техносфера 2007. 472с.
3. Электроника : учебно-методическое пособие / Л. Г. Далгатова, И. И. Кузнецова, Г. Р. Гаджибабаев, И. Б. Магарамов. — Махачкала : ДагГАУ имени М.М.Джамбулатова, 2021. — 56 с
4. Дерягин А. В. Лабораторный практикум по основам автоматике и вычислительной техники. Учебно-методическое пособие для студентов физико-математического факультета педвуза специальность 032200.00 " физика с дополнительной специальностью. Допущено УМО по направлениям пед. образования МО РФ, Санкт-Петербург, 2004 52 с.3,3 п.л.
5. Дерягин А. В. Лабораторный практикум. Радиоэлектроника / Учебно-методическое пособие для студентов физико-математического и инженерно-технологического факультетов. – Елабуга: изд-во ЕИ КФУ, 2013. – 50с
6. Дерягин А. В., Насыбуллин Р. А. Конструирование комбинационных схем. //Методическое пособие для студентов физико-математического факультета педвуза. Елабуга, 1997г. 2 п.л.
7. Дерягин А.В. Изучение микроконтроллера Stellaris LM4F120XL / А.В. Дерягин / Учебное пособие. – Елабуга: ЕИ КФУ, 2018.– 49 с.
8. Джонс М.Х. Электроника – практический курс (пер. с англ. Воронова Е.В., Ларина А.Л.) Изд. 2-е, испр. Серия Мир электроники Издательство Техносфера 2006. 512с.
9. Ефимов, И. Е. Основы микроэлектроники : учебник / И. Е. Ефимов, И. Я. Козырь. — 3-е изд. — Санкт-Петербург : Лань, 2022. — 384 с.
10. Зорин, О. А. Основы электротехники и цифровой электроники : учебное пособие / О. А. Зорин. — Пермь : ПГАТУ, 2021. — 214 с.
11. Кушнер, Д. А. Основы автоматике и микропроцессорной техники : учебное пособие / Д. А. Кушнер, А. В. Дробов, Ю. Л. Петроченко. — Минск : РИПО, 2019. — 245 с.
12. Маркарян, Л. В. Схемотехника цифровой электроники : учебное пособие / Л. В. Маркарян. — Москва : МИСИС, 2018. — 74 с.
13. Маркарян, Л. В. Схемотехника цифровой электроники : учебное пособие / Л. В. Маркарян. — Москва : МИСИС, 2018. — 74 с.

14. Опадчий Ю., Глудкин О. П., Гуров А. И. Аналоговая и цифровая электроника (полный курс) Учеб. для вузов. Под ред. О. П. Глудкина. – М.: Горячая линия – Телеком. 2005. -768с.: ил.
15. Расторгуев, А. Н. Основы цифровой электроники : учебное пособие / А. Н. Расторгуев. — Санкт-Петербург : СПбГЛТУ, 2013. — 52 с.
16. Самедов М.Н., Шибанов В.М. Электроника: учебно-методическое пособие. – Елабуга: изд-во ЕИ КФУ, 2013. – 160 с.
17. Смирнов, Ю. А. Основы микроэлектроники и микропроцессорной техники : учебное пособие / Ю. А. Смирнов, С. В. Соколов, Е. В. Титов. — 2-е изд., испр. — Санкт-Петербург : Лань, 2022. — 496 с.
18. Соколов, О. А. Электроника : учебное пособие / О. А. Соколов, П. С. Назаров, Д. О. Соколов. — Санкт-Петербург : СПбГУ ГА им. А.А. Новикова, 2022. — 178 с.
19. Устройства цифровой электроники : учебно-методическое пособие / составитель В. И. Парфенов. — Воронеж : ВГУ, 2017. — 42 с. —

Учебное издание

Дерягин Александр Владимирович

Сабирова Файруза Мусовна

**Основы автоматики и вычислительной техники**

Учебное пособие

Техническое редактирование и компьютерная верстка

*Ф.М. Сабировой*

Подписано к печати 25.01.2023.

Формат 60x84<sub>1/16</sub>. Бумага офсетная.

Гарнитура «Times». Печать цифровая.

Усл. печ. 7,21 л. Печ. 7,75 л. Тираж 100 экз. Заказ № 14

420111, Казань, Дзержинского, 9/1. Тел. 8(917)264-84-83  
Отпечатано в редакционно-издательском центре «Школа»  
E-mail: ric-school@yandex.ru

-----  
ISBN 978-5-00162-786-9

