

Федеральное агентство по образованию

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Приоритетный национальный проект «Образование»

Инновационная образовательная программа
Санкт-Петербургского государственного
политехнического университета

Бунтов В.Д., Макаров С.Б.

Микропроцессорные системы
Часть I. Цифровые устройства

Учебное пособие

Санкт-Петербург
Издательство политехнического университета
2008

Федеральное агентство по образованию

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Приоритетный национальный проект «Образование»

Инновационная образовательная программа
Санкт-Петербургского государственного
политехнического университета

Бунтов В.Д., Макаров С.Б.

Микропроцессорные системы
Часть I. Цифровые устройства

Учебное пособие

Санкт-Петербург
Издательство политехнического университета
2008

УДК 621.3.049; 621.395.5; 681.327(075)

ББК 32.844-02: 32.73.26-02я73

Бунтов В.Д., Макаров С.Б. Микропроцессорные системы. Часть I. Цифровые устройства. Учебное пособие. – СПб.: Изд-во политехнического университета, 2008. – 199 с.

В предлагаемом учебном пособии рассмотрены вопросы, связанные с построением структурных и электрических схем различных цифровых устройств современных радиотехнических систем. Книга посвящена применению типовых цифровых микросхем и микропроцессоров в функциональных узлах радиотехнических устройств. Особое внимание уделено методам проектирования различных комбинационных и последовательностных устройств и их применению в цифровых радиотехнических устройствах формирования, генерирования и приема сигналов. Описаны различные методы проектирования, в том числе с использованием микропроцессоров. Рассмотрены особенности проектирования радиотехнических устройств, в которых использовано цифроаналоговое и аналого-цифровое преобразование сигналов. Излагаемый материал иллюстрируется электрическими схемами различных функциональных узлов радиотехнических устройств.

Учебное пособие составлено преподавателями кафедры “Радиоэлектронные средства защиты информации” Санкт-Петербургского государственного политехнического университета и предназначено для студентов, обучающихся по направлениям подготовки 230100 – Информатика и вычислительная техника, 210300 – Радиотехника, 210400 – Телекоммуникации, а также соответствующим инженерным специальностям.

Работа выполнена в рамках реализации Инновационной образовательной программы Санкт-Петербургского государственного политехнического университета «Развитие политехнической системы подготовки кадров в инновационной среде науки и высокотехнологичных производств Северо-Западного региона России».

Печатается по решению редакционно-издательского совета Санкт-Петербургского государственного политехнического университета.

© Бунтов В.Д., Макаров С.Б., 2008

© Санкт-Петербургский государственный политехнический университет, 2008

Оглавление

Введение	5
Глава 1. Комбинационные устройства	6
1.1. Позиционные системы счисления	6
1.2. Способы задания логических функций	8
1.3. Правила эквивалентных преобразований структурных формул	15
1.4. Минимизация структурных формул с помощью методов Квайна и карт Карно	17
1.5. Не полностью определённые логические функции и их минимизация	24
1.6. Минимизация системы логических функций	26
1.7. Универсальные логические элементы	31
1.8. Примеры построения комбинационных устройств на универсальных логических элементах	33
1.9. Общая задача синтеза комбинационных устройств	36
1.10. Типовые комбинационные устройства	39
1.10.1. Дешифраторы и шифраторы	39
1.10.2. Кодопреобразователи	44
1.10.3. Мультиплексоры и демультиплексоры	46
1.11. Проектирование комбинационных устройств, свободных от состязаний сигналов	58
Глава 2. Последовательностные устройства	70
2.1. Способ формального описания последовательностных устройств	70
2.2. Триггеры	79
Асинхронные RS-триггеры	
Синхронные RS-триггеры	
Т-триггеры и DV-триггеры	
2.3. Канонический метод проектирования последовательностных устройств	92
2.4. Синтез триггеров	96
2.5. Регистры	102
2.6. Счетчики	109
2.6.1. Двоичные счетчики	109
2.6.2. Счетчики с произвольным коэффициентом пересчета	114
2.6.3. Счетчики с переменным коэффициентом пересчета	118

2.6.4. Счетчики на основе регистров сдвига	120
Глава 3. Запоминающие устройства	125
3.1. Организация запоминающих устройств, их типы и характеристики	125
3.2. Оперативные запоминающие устройства	128
3.3. Постоянные запоминающие устройства	131
3.4. Программируемые логические матрицы	136
3.5. Базовые матричные кристаллы	141
3.6. Программируемые логические интегральные схемы	143
Глава 4. Цифроаналоговые преобразователи	150
4.1. Принципы цифроаналогового преобразования	150
4.2. Основные параметры цифроаналогового преобразователя	152
4.3. Реализация АЦП на интегральных схемах	155
Глава 5. Аналогово-цифровые преобразователи	162
5.1. Принципы аналого-цифрового преобразования и основные параметры АЦП	162
5.2. Методы построения АЦП	164
5.3. Интегральные схемы аналого-цифровых преобразователей	176
Глава 6. Радиотехнические устройства формирования, приема и передачи сигналов	183
6.1. Генераторы случайных последовательностей	183
6.2. Устройства формирования последовательностей Уолша	187
6.3. Генераторы сигналов	194
6.4. Цифровой коррелятор	208
6.5. Цифровые согласованные фильтры	212
6.6. Устройство выбора максимального напряжения	217
6.7. Устройство синхронизации радиомодемов	218
6.8. Дискретные фазовращатели	222
6.9. Цифровые фазовые и частотные дискриминаторы	224
6.10. Устройства контроля, кодирования и декодирования	232
6.10.1. Устройства контроля по модулю 2	232
6.10.2. Устройства кодирования и декодирования кодов Хэмминга	233
6.10.3. Устройство кодирования для кода Рида – Соломона	240

Введение

Цифровые и микропроцессорные радиотехнические устройства применяются для построения сетки опорных частот возбuditелей радиопередатчиков, в системах фазовой автоподстройки частоты радиоприемников и синтезаторах частот мобильных радиотелефонов. Кроме того, они используются для цифровой частотной селекции и детектирования, в устройствах кодирования и сжатия сигналов цифровых систем связи и телевидения, для сопряжения протоколов работы телекоммуникационных устройств, в устройствах преобразования аналоговых сигналов в цифровые и т.д. Интенсивное развитие теории и методов оптимального приема сигналов на фоне различного рода помех привело к необходимости создания достаточно сложных радиотехнических систем обнаружения и различения сигналов, таких как согласованные фильтры, устройства нелинейной фильтрации, реализация которых в ряде случаев возможна лишь на основе цифровых и микропроцессорных устройств.

В зависимости от назначения и параметров радиотехнической системы (РТС) приходится сталкиваться с различными методами проектирования цифровых устройств РТС и с различными типами применяемых в цифровых устройствах интегральных схем (ИС) и микропроцессоров. Во многих системах передачи и обработки изображений, радиолокации, в первую очередь решаются задачи повышения быстродействия РТУ, и часто рациональное проектирование такой аппаратуры связано с правильным выбором типа ИС и построением электрической схемы на минимальном числе корпусов ИС с минимальным числом их соединений при отсутствии состязаний сигналов в цифровом устройстве РТС.

При использовании в качестве ИС цифровых перепрограммируемых логических интегральных схем (ПЛИС), например фирмы ALTERA [2, 22, 27], необходимо обеспечить рациональное построение программы работы ПЛИС, позволяющее минимизировать объем памяти, энергопотребление и, в конечном итоге, стоимость цифрового устройства РТС. Для успешного решения подобных задач необходимо обладать методами расчета, учитывающими указанные выше особенности построения радиотехнических систем.

Цифровые устройства разделяются на два класса: *комбинационные* устройства, которые не обладают памятью, и их логическое состояние однозначно определяется входными сигналами, имеющимися в данный момент времени, и *последовательностные* устройства, которые обладают

памятью, и их логическое состояние определяется входными сигналами как в настоящий, так и в предыдущие моменты времени.

Комбинационные устройства реализуются, как правило, на отдельных ИС малой степени интеграции, либо изготавливаются в виде отдельной интегральной схемы средней степени интеграции. Основными узлами сложных последовательностных устройств являются *регистры*, выполняющие операции хранения и сдвига логического числа (логического уровня 1 или 0) на определенное число разрядов, *счетчики*, на выходах которых формируются числа, соответствующие количеству поступивших на входы сигналов, *генераторы чисел* (последовательностей), на выходах которых образуются заданные последовательности чисел. Последовательностные устройства (ПУ) могут быть синхронными, когда начало выполнения каждой логической операции четко фиксируется во времени (синхронизируется) поступлением синхронизирующего (тактового) сигнала, и асинхронными, не имеющими синхронизирующих сигналов. Как правило, более быстродействующими являются асинхронные устройства, однако правильная организация их работы оказывается сложнее, чем у синхронных устройств. В системах дальней радиосвязи, цифрового телевидения и мобильной сотовой связи, радиолокации обычно используются асинхронные ПУ, входящие в цифровые системы фазовой синхронизации, когерентные демодуляторы сигналов, частотные дискриминаторы. Синхронные ПУ нашли широкое применение в системах проводной и радиорелейной связи, устройствах сопряжения персональных компьютеров с периферийной аппаратурой, в радиотелеметрических системах, многоканальных телекоммуникационных системах передачи информации.

При проектировании цифровых устройств РТС, применяемых в информационных радиосистемах, т. е. связанных с передачей, приемом и преобразованием информации, необходимо решать задачу преобразования аналогового сигнала в цифровой код путем перехода к рассмотрению квантованных по уровню выборочных значений сигнала, и обратную задачу преобразования цифрового кода в аналоговое колебание. Эти процедуры выполняются с помощью аналого-цифровых (АЦП) и цифроаналоговых (ЦАП) преобразователей, в состав которых входят и ПУ, выполняющие функции устройств управления и сдвига, выборки и хранения, преобразования кодов.

Микропроцессорные устройства (микропроцессоры) – это обрабатывающие и управляющие цифровые однокристалльные или многокристалль-

ные структуры, обладающие способностью выполнять под программным управлением обработку входных сигналов, включая ввод и вывод данных, принятие решений, арифметические и логические операции.

Радиотехнические системы, построенные на основе микропроцессорных устройств, имеют значительно большую гибкость, чем системы с жесткой структурой. Классическим примером использования микропроцессоров можно считать их применение в мобильных телефонных устройствах подвижных средств телекоммуникаций. В этих устройствах как нельзя лучше проявилось основное преимущество микропроцессоров – их программная многофункциональность. В настоящее время практически ни одно сложное цифровое устройство радиотехнической системы не обходится без микропроцессора.

В книге рассмотрены вопросы, связанные с построением структурных и электрических схем различных цифровых устройств современных радиотехнических систем. Разумеется, используемые на практике методы построения подобных устройств могут отличаться от рассмотренных. Тем не менее, знание основных методов и правил построения цифровых и микропроцессорных устройств позволит быстро и рационально осуществить разработку новых радиотехнических систем с использованием быстроизменяющейся элементной базы цифровых интегральных схем и микропроцессоров.

Целью книги является попытка изложить с единых теоретических позиций методы проектирования цифровых и микропроцессорных устройств, не претендуя на подробное изложение достаточно полно представленных в литературе методов программирования специализированных микропроцессоров обработки сигналов [4, 8, 11, 13, 15, 16], описание цифровых интегральных схем, проектирование быстродействующих цифроаналоговых и аналого-цифровых преобразователей [14, 17, 18, 24, 25], проектирование узлов персональных компьютеров [8, 9, 11, 12]. Основное внимание в книге уделено:

- методам синтеза комбинационных устройств, в том числе устройств, свободных от состязаний сигналов;
- анализу способов формального описания последовательностных устройств, таких как триггеры, регистры, счетчики;
- принципам построения цифроаналоговых и аналого-цифровых преобразователей и их реализации на цифровых интегральных схемах;
- методам построения цифровых устройств формирования и обра-

ботки сигналов, в том числе цифровых корреляторов и согласованных фильтров;

- структурным и электрическим схемам построения радиотехнических систем на базе микропроцессоров;
- реализации радиотехнических устройств передачи и обработки сигналов в системах радиосвязи, радиолокации и радионавигации.

Глава 1. Комбинационные устройства

1.1. Позиционные системы счисления

Для описания логических функций комбинационных устройств обычно используются целые n -разрядные числа. При представлении чисел применяются определенные совокупности правил записи, называемые **системами счисления**. В позиционных системах счисления целое n -разрядное (положительное) число записывается в виде последовательности символов e_k ($k = n - 1 \dots 0$) следующим образом:

$$e_{n-1} \dots e_k \dots e_2 e_1 e_0.$$

Значение каждого символа e_k определяется его позицией в записи n -разрядного числа. Тогда любое целое число в системе счисления с основанием q можно записать в виде

$$e_{n-1}q^{n-1} + \dots + e_kq^k + \dots + e_2q^2 + e_1q^1 + e_0.$$

При этом символ e_{n-1} соответствует старшему разряду, символ e_0 – младшему разряду целого числа, а сложение выполняется по модулю, равному значению основания системы счисления.

Десятичная система счисления

Привычная для человеческой деятельности десятичная (decimal) система счисления имеет основание $q=10$. Значение каждого символа e_k в такой системе может изменяться от 0 до 9. Примером записи значений позиций чисел в десятичной системе при $q=10$ и $n=4$ является число 1327 или

$$e_3q^3 + e_2q^2 + e_1q^1 + e_0 = 1 \cdot 10^3 + 3 \cdot 10^2 + 2 \cdot 10^1 + 7$$

Двоичная система счисления

В двоичной (binary) системе счисления или системе с основанием $q=2$ значение каждого символа e_k может быть равно лишь 0 или 1. Например, 4-х разрядное ($n=4$) число в такой системе может быть записано в виде:

$$e_32^3 + e_22^2 + e_12^1 + e_0,$$

где сложение выполняется по модулю 2, а каждое слагаемое представляет собой n -разрядное двоичное число. Так, если $e_3 = 1, e_2 = 0, e_1 = 1, e_0 = 1$, то имеем двоичное число 1011.

Одно и тоже число может быть представлено в различных системах

счисления. Рассмотрим процедуру преобразования числа из десятичной системы счисления в двоичную. Для этого необходимо найти значения каждого символа e_k в двоичной системе, начиная с младшего разряда. Исходя из общей формы записи целого числа в системе счисления с основанием $q=2$, очевидно, получим значение e_0 , равное остатку от деления числа, записанного в десятичной системе, на основание $q=2$. Значение e_1 образуется, как остаток от деления, полученный на предыдущем шаге, десятичного числа целой части на основание $q=2$. Такое деление продолжается до тех пор, пока целая часть десятичного числа не окажется равной нулю. Например, числу 155 в десятичной системе будет соответствовать 8-разрядное число 10011011 в двоичной системе счисления.

Представим обратное преобразование числа из двоичной системы в десятичную систему счисления. Так, для предыдущего примера, имеем:

$$1 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 128 + 16 + 8 + 2 + 1 = 155$$

Шестнадцатеричная система счисления

В шестнадцатеричной (Hexadecimal) системе счисления или системе с основанием $q=16$, используются 16 значений каждого символа e_k : цифры от 0 до 9 и буквы A, B, C, D, E, F, которым соответствуют числа 10, 11, 12, 13, 14, 15 в десятичной системе счисления. Шестнадцатеричная система счисления широко используется на практике как экономичная форма записи чисел в большинстве микропроцессоров и микроконтроллеров.

Для преобразования числа из двоичной системы счисления в число шестнадцатеричной системы, его надо разделить на группы по 4 двоичных символа (тетрады). Затем каждой группе из 4-х символов поставить в соответствие шестнадцатеричный символ (цифру или букву).

Для обратного преобразования числа из шестнадцатеричной системы в число двоичной системы счисления каждый символ шестнадцатеричной системы должен быть заменен четырьмя двоичными символами. Так, шестнадцатеричное число 2С6Е может быть представлено в двоичной системе в виде 0010110001101110.

Шестнадцатеричные числа могут быть преобразованы в десятичные числа. Для предыдущего примера при $q=16$ и $n=4$ имеем:

$$e_3q^3 + e_2q^2 + e_1q^1 + e_0 = 2 \cdot 16^3 + C \cdot 16^2 + 6 \cdot 16 + E = 2 \cdot 16^3 + 12 \cdot 16^2 + 6 \cdot 16 + 14 = 11374$$

Рассмотрим процедуру преобразования числа из десятичной системы счисления в шестнадцатеричную. Для этого требуется найти значение сим-

волов e_k в шестнадцатеричной системе, начиная с младшего разряда. Значение e_0 будет равно остатку от деления десятичного числа на основание системы $q=16$. Значение e_k образуется, как остаток от деления, полученной на предыдущем шаге, целой части десятичного числа на $q=16$. Такое деление продолжается до тех пор, пока целая часть десятичного числа не окажется равной нулю. Например, числу 15797 в десятичной системе будет соответствовать 4-разрядное число 3DB5 в шестнадцатеричной системе счисления.

Восьмеричная система счисления

В этой системе счисления (octal) с основанием $q=8$ используется 8 значений каждого символа e_k от 0 до 7. Для преобразования двоичного числа в восьмеричное число его надо разделить на группы из 3 двоичных символов (триады). Далее каждая такая группа символов преобразуется в восьмеричный символ. Так двоичное число 011111000100 может быть представлено в восьмеричной системе как 3704.

При обратном преобразовании каждый символ восьмеричной системы заменяется тремя двоичными символами.

Преобразование чисел из восьмеричной системы в числа десятичной системы и обратно выполняется аналогично рассмотренным ранее преобразованиям чисел других систем счисления. Так, для восьмеричного числа 2357 эквивалентное десятичное число равно 1263. Десятичному числу 3336 соответствует восьмеричное число 6410.

1.2. Способы задания логических функций

В цифровых устройствах используются микроэлектронные элементы, на входах и выходах которых возможны два состояния, различающиеся значением какой-либо электрической величины – напряжения u или тока i . Одному из значений соответствует 0, другому – 1 (рис. 1.1).

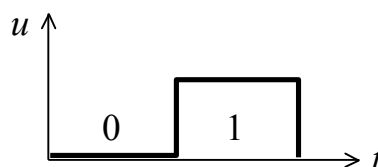


Рис. 1.1. Уровни напряжения, соответствующие 0 и 1

В комбинационных устройствах связь между выходными y_1, y_2, \dots, y_m (рис. 1.2) и входными x_1, x_2, \dots, x_n сигналами может быть за-

дана функциями вида:

$$\begin{aligned}
 y_1 &= f_1(x_1, x_2, \dots, x_n); \\
 y_2 &= f_2(x_1, x_2, \dots, x_n); \\
 &\dots \dots \dots \\
 y_m &= f_m(x_1, x_2, \dots, x_n).
 \end{aligned}$$

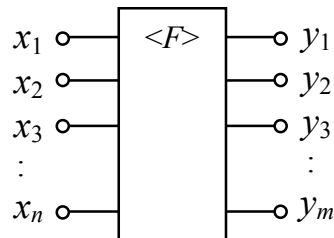


Рис. 1.2. Обозначение комбинационного устройства на функциональных схемах

Особенность входных (независимых переменных) и выходных сигналов (функций) заключается в том, что они могут принимать только два значения: 0 и 1. Такие функции называются **логическими**, а для их теоретического описания используется двоичная система счисления. Теоретической основой цифровых устройств, описываемых логическими функциями, является алгебра логики [1, 5, 6, 10, 18].

Существуют три способа задания логических функций (ЛФ) – словесное описание, таблица истинности и структурная формула.

Словесное описание должно однозначно определять, при каких значениях входных сигналов функция y_i принимает значение 1, а при каких – 0. Так, например, для $i = 1, n = 2$ имеем следующее словесное описание: если $x_1 = x_2$, то $y_1 = 0$, а если $x_1 \neq x_2$, то $y_1 = 1$.

Таблица истинности – перечисление всех возможных значений входных и соответствующих им выходных сигналов.

Таблица 1.1

x_2	x_1	y
0	0	0
0	1	1
1	0	1
1	1	0

При этом способе задания функция y_i представляется в виде таблицы,

в которой записываются все возможные (2^n) наборы входных сигналов в порядке возрастания их значений и для каждого набора устанавливается значение функции (0 или 1). Например, для функции, называемой функцией неравнозначности [6, 18], словесное описание которой дано выше, таблица истинности имеет вид, представленный в табл. 1.1.

Таблица истинности, по существу, представляет собой табличную форму записи словесного описания логической функции. Соответственно, имея словесное описание логической функции, легко составить её таблицу. Если таблицы истинности двух логических функций одинакового числа переменных совпадают друг с другом с точностью до перестановки строк, то они равны. Это позволяет производить сравнение логических функций.

Базовые логические функции двух переменных

Логическое отрицание (инверсия) – логический элемент НЕ, выполняющий операцию $y = \bar{x}$. Все возможные значения входных и выходных сигналов приведены в таблице истинности (табл. 1.2). Обозначение на схемах логического элемента НЕ приведено на рис.1.3.

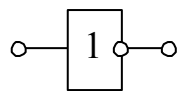


Рис.1.3. Обозначение логического элемента НЕ

Таблица 1.2.

x	y
0	1
1	0

Логическое умножение (конъюнкция) – логический элемент И, выполняющий операцию $y = x_1 \cdot x_2$. Из таблицы истинности (табл.1.3) можно установить все возможные значения выходных сигналов при заданных значениях входных сигналов. Обозначение логического элемента на схемах приведено на рис.1.4.

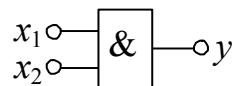


Рис. 1.4. Обозначение логического элемента И

Таблица 1.3.

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Логическое сложение (дизъюнкция) – логический элемент ИЛИ, выполняющий операцию $y = x_1 + x_2$. Все возможные значения входных и

выходных сигналов рассмотрены в таблице истинности (табл.1.4), а обозначение логического элемента ИЛИ на схемах приведено на рис.1.5.

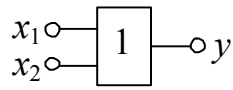


Рис. 1.5. Обозначение логического элемента ИЛИ

Таблица 1.4.

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	1

Поскольку все возможные логические функции могут быть образованы с помощью операций логического отрицания, логического сложения и логического умножения, то в общем случае логическую функцию y можно выразить с помощью структурной формулы, содержащей эти операции над независимыми переменными. Например, функция неравнозначности будет определяться формулой: $y_1 = \bar{x}_2 \cdot x_1 + x_2 \cdot \bar{x}_1$. Если определен порядок выполнения операций, то от формулы можно перейти к структурной схеме (рис. 1.6). Обычно операции по очередности их выполнения располагаются следующим образом: инверсия (НЕ), конъюнкция (И), дизъюнкция (ИЛИ). Для изменения порядка выполнения операций используют скобки. Задание логической функции с помощью структурной формулы позволяет построить схему устройства и оценить сложность её реализации. Кроме того, существующие правила эквивалентных преобразований структурных формул дают возможность минимизировать число элементов заданного типа.

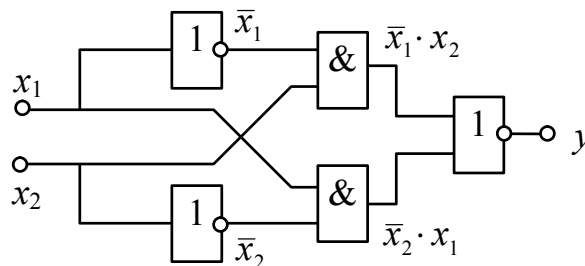


Рис.1.6. Структурная схема, соответствующая структурной формуле

$$y_1 = \bar{x}_2 x_1 + x_2 \bar{x}_1$$

Существуют две формы записи структурной формулы, дающие однозначное представление логической функции: в виде совершенной

дизъюнктивной нормальной формы (СДНФ) и в виде совершенной конъюнктивной нормальной формы (СКНФ).

Запись структурной формулы по таблице истинности в виде СДНФ

Рассмотрим на примере правило перехода от таблицы истинности логической функции к её записи в виде СДНФ. Для $y = f(x_1, x_2, x_3)$ введём в таблицу истинности три столбца вспомогательных функций (табл.1.5).

Число вспомогательных функций соответствует числу единиц в таблице истинности искомой функции y . Структура вспомогательных функций в этом случае будет следующая: каждому значению $y = 1$ соответствует значение 1 в одном столбце y_i , так что число нулей в данном столбце всегда равно семи (см. табл. 1.5). Функции такого вида называют полной элементарной конъюнкцией, минтермом или конститuentой единицы.

По табл. 1.5 можно составить формулы для $y_i = f_i(x_1, x_2, x_3)$, $y = f(y_1, y_2, y_3)$ и схему, реализующую логическую функцию y . Запишем:

$$y_1 = x_1 \cdot x_2 \cdot x_3;$$

$$y_2 = \bar{x}_1 \cdot x_2 \cdot x_3;$$

$$y_3 = \bar{x}_1 \cdot \bar{x}_2 \cdot x_3;$$

$$y = y_1 + y_2 + y_3 = x_1 \cdot x_2 \cdot x_3 + \bar{x}_1 \cdot x_2 \cdot x_3 + \bar{x}_1 \cdot \bar{x}_2 \cdot x_3.$$

Полученная структурная формула, соответствующая таблице истинности (см. табл.1.5), и есть СДНФ для функции y или запись в виде суммы членов, каждый из которых представляет собой произведение аргументов и их инверсий.

Нормальной называется функция (форма), представляемая суперпозицией специально вводимых вспомогательных функций – конститuent 1 (0).

Совершенной называют функцию (форму), в которой произведения (суммы) содержат все аргументы, причём каждый аргумент входит только один раз в прямом или инверсном виде. Следовательно, к свойствам СДНФ следует отнести следующие:

- в СДНФ нет двух одинаковых произведений;
- ни одно произведение не содержит двух одинаковых множителей;
- ни одно произведение не содержит вместе с независимой переменной её инверсию.

Таким образом, каждая логическая функция может иметь единственное представление в виде СДНФ. Для записи структурной формулы в виде СДНФ по таблице истинности необходимо:

Таблица 1.5.

x_3	x_2	x_1	y_1	y_2	y_3	y
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	0	0
1	0	0	0	0	1	1
1	0	1	0	0	0	0
1	1	0	0	1	0	1
1	1	1	1	0	0	1

- отметить строки, где $y = 1$;
- для каждой отмеченной строки составить логическое произведение всех независимых переменных, причём независимая переменная, равная 1 в выделенной строке, входит в соответствующее произведение без инверсии, а равная 0 – с инверсией;
- составить логическую сумму полученных произведений.

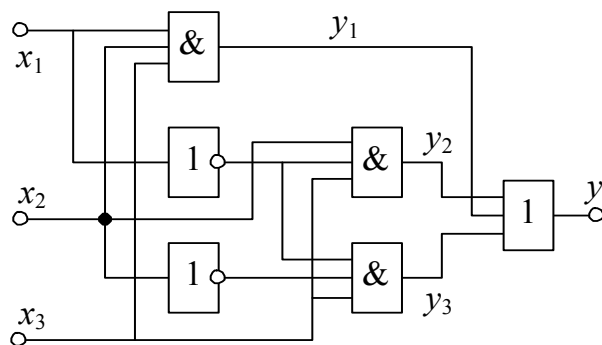


Рис.1.7. Структурная схема, реализующая СДНФ

Сформулированное правило позволяет записать структурную формулу без использования вспомогательных функций. Схема, реализующая структурную формулу, полученную по табл.1.5, приведена на рис.1.7.

Запись структурной формулы в виде СКНФ. Другой способ записи структурной формулы по таблице истинности может быть представлен в виде совершенной конъюнктивной нормальной формы (СКНФ). Так, в табл.1.6 число введённых вспомогательных функций соответствует числу нулей для функции y .

Вспомогательные функции y_i строятся таким образом, чтобы значению $y = 0$ соответствовало значение $y_i = 0$ при наличии в каждом столбце семи единиц. Такие функции называют макстермами или конституентами нуля.

Таблица 1.6.

x_3	x_2	x_1	y_1	y_2	y_3	y
0	0	0	0	1	1	0
0	0	1	1	0	1	0
0	1	0	1	1	1	1
0	1	1	1	1	0	0
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

Из табл. 1.6 получим:

$$y_1 = x_1 + x_2 + x_3;$$

$$y_2 = \bar{x}_1 + x_2 + x_3;$$

$$y_3 = \bar{x}_1 + \bar{x}_2 + x_3;$$

$$y = y_1 y_2 y_3 = (x_1 + x_2 + x_3)(\bar{x}_1 + x_2 + x_3)(\bar{x}_1 + \bar{x}_2 + x_3).$$

Для записи структурной формулы в виде СКНФ по таблице истинности необходимо:

- отметить все строки, где $y = 0$;
- для каждой отмеченной строки составить логическую сумму всех независимых переменных, причём независимая переменная, равная 0 в выделенной строке, входит в соответствующую сумму без инверсии, а равная 1 – с инверсией;
- составить логическое произведение полученных сумм.

Структурные формулы в виде СДНФ и СКНФ эквивалентны и могут быть преобразованы одна в другую на основе законов и тождеств алгебры логики [1, 6, 10, 18].

1.3. Правила эквивалентных преобразований структурных формул

Правила эквивалентных преобразований структурных формул логи-

ческих функций основаны на законах алгебры логики. Основные законы алгебры логики:

- переместительный (коммутативный):

$$x_1 + x_2 = x_2 + x_1; \quad x_1 \cdot x_2 = x_2 \cdot x_1;$$

- сочетательный (ассоциативный):

$$(x_1 + x_2) + x_3 = x_1 + (x_2 + x_3),$$

$$(x_1 \cdot x_2) \cdot x_3 = x_1 \cdot (x_2 \cdot x_3);$$

- распределительный (дистрибутивный):

$$x_1 \cdot (x_2 + x_3) = x_1 \cdot x_2 + x_1 \cdot x_3,$$

$$x_1 + (x_2 \cdot x_3) = (x_1 + x_2) \cdot (x_1 + x_3);$$

- отрицания или двойственности (правила де Моргана):

$$\overline{x_1 + x_2} = \bar{x}_1 \cdot \bar{x}_2, \quad \overline{x_1 \cdot x_2} = \bar{x}_1 + \bar{x}_2.$$

К основным тождествам алгебры логики можно отнести следующие:

- агрессивности и нейтральности:

$$x + 1 = 1, \quad x + 0 = x; \quad x \cdot 1 = x; \quad x \cdot 0 = 0;$$

- поглощения: $x + x = x; \quad x \cdot x = x;$

- исключения: $x + \bar{x} = 1; \quad x \cdot \bar{x} = 0;$

- двойного отрицания: $\bar{\bar{x}} = x.$

Применение законов и тождеств даёт возможность, производя эквивалентные преобразования, упрощать структурные формулы логических функций, т.е. находить для них более простые выражения. Примерами эквивалентных преобразований являются *операции склеивания и поглощения*.

Так, для СДНФ, выполнив эквивалентные преобразования структурных формул, получим:

$$x_1 \cdot x_2 + x_1 \cdot \bar{x}_2 = x_1 \cdot (x_2 + \bar{x}_2) = x_1 \cdot 1 = x_1;$$

$$x_1 \cdot x_2 \cdot x_3 + x_1 \cdot \overline{x_2 \cdot x_3} = x_1 \cdot (x_2 \cdot x_3 + \overline{x_2 \cdot x_3}) = x_1;$$

$$x_1 \cdot x_2 \cdot x_3 + x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 = x_1 \cdot (x_2 \cdot x_3 + \bar{x}_2 \cdot \bar{x}_3) \neq x_1$$

Операция склеивания суммы логических произведений выполняется, когда логические произведения имеют одинаковый множитель (это может

быть один аргумент или произведение нескольких аргументов), а второй сомножитель в одно произведение входит в инверсном виде, в другое – в прямом. Результатом операции склеивания является одинаковый для произведений множитель.

В случае, когда первое слагаемое входит в качестве сомножителя во второе слагаемое, выполняется операция поглощения, результат которой – первое слагаемое:

$$x_1 + x_1 \cdot x_2 = x_1 \cdot (1 + x_2) = x_1 \cdot 1 = x_1.$$

Для СКНФ после эквивалентных преобразований получим:

$$(x_1 + x_2) \cdot (x_1 + \bar{x}_2) = x_1 \cdot x_1 + x_2 \cdot x_1 + x_1 \cdot \bar{x}_2 + x_2 \cdot \bar{x}_2 = x_1 \cdot (1 + x_2 + \bar{x}_2) = x_1.$$

Если имеются два сомножителя, каждый из которых представляет собой логическую сумму, причём в этих суммах есть одинаковые слагаемые, а неодинаковые слагаемые являются инверсными друг другу, то осуществляется операция склеивания, результатом которой является общее слагаемое. Если один из двух сомножителей входит в другой в качестве слагаемого, то выполняется операция поглощения, в результате чего остаётся общая часть сомножителей:

$$x_1 \cdot (x_1 + x_2) = x_1 + x_1 \cdot x_2 = x_1.$$

Произведя аналогичные преобразования, можно получить результат поглощения отрицания в виде:

$$\begin{aligned} x_1 + \bar{x}_1 \cdot x_2 &= x_1 + x_2; & \bar{x}_1 + x_1 \cdot x_2 &= \bar{x}_1 + x_2; \\ x_1 \cdot (\bar{x}_1 + x_2) &= x_1 \cdot x_2; & \bar{x}_1 \cdot (x_1 + x_2) &= \bar{x}_1 \cdot x_2. \end{aligned}$$

Приведенные правила эквивалентных преобразований справедливы как для независимых переменных (x_i), так и для их функций (y_k) в сложных комбинационных устройствах.

1.4. Минимизация структурных формул с помощью метода Квайна и карт Карно

Целью минимизации является получение эквивалентной структурной формулы, соответствующей наименьшему числу элементов в цифровом устройстве. Достижение минимизированной формы ЛФ основывается на эквивалентных (тождественных) преобразованиях структурных формул для логических функций.

Таблица 1.7

x_3	x_2	x_1	y_1	y_2	y_3
0	0	0	1	1	0
0	0	1	1	1	1
0	1	0	0	1	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	0	1
1	1	0	0	0	1
1	1	1	0	1	1

При минимизации по методу Квайна используются операции склеивания и поглощения. В качестве примера рассмотрим три функции, заданные таблицей истинности (табл.1.7). Запишем их в СДНФ и упростим формулы, используя операции склеивания и поглощения.

Для первой функции получим:

$$y_1 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 = \bar{x}_2 \cdot \bar{x}_3 \cdot (\bar{x}_1 + x_1) = \bar{x}_2 \cdot \bar{x}_3.$$

Таким образом, вместо шести логических элементов – трёх НЕ, двух 3-х входных И, одного ИЛИ, требуемых для реализации СДНФ, достаточно иметь три логических элемента – два НЕ и одно И.

Функция y_2 в СДНФ имеет следующий вид:

$$y_2 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_1 \cdot x_2 \cdot \bar{x}_3 + x_1 \cdot x_2 \cdot x_3.$$

Выполняя возможные операции склеивания, представим y_2 в форме:

$$y_2 = \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_1 \cdot \bar{x}_3 + x_1 \cdot x_2 \cdot x_3.$$

Аналогично для y_3 имеем:

$$\begin{aligned} y_3 &= x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_2 \cdot x_3 + \bar{x}_1 \cdot x_2 \cdot x_3 + x_1 \cdot x_2 \cdot x_3 = \\ &= x_1 \cdot \bar{x}_2 + x_1 \cdot x_3 + x_2 \cdot x_3 = x_1 \cdot \bar{x}_2 + x_2 \cdot x_3. \end{aligned}$$

Здесь очевидны недостатки метода Квайна. Во-первых, в случае сложной функции возрастает число анализируемых вариантов склеиваний и поглощений, а во-вторых, могут появляться лишние операции склеивания.

При минимизации на основе карт Карно используются те же операции, что и при минимизации по методу Квайна, но формой представления исходных данных является карта (рис.1.8) с числом клеток 2^p , где p – число аргументов функции. Каждая карта соответствует одной функции, число

аргументов p которой определяет размеры карты. Клетка карты Карно соответствует строке таблицы истинности, а кодировка клеток определяется значениями аргументов. Признаком выполнения операции склеивания является расположение в соседних клетках единиц, поскольку кодировка соседних клеток карты такова, что минтермы (произведения переменных) отличаются значениями только одной переменной. Заметим, что по этой же причине соседними считаются и крайние по вертикали и горизонтали клетки, (как было бы, если бы карта была свёрнута в цилиндр).

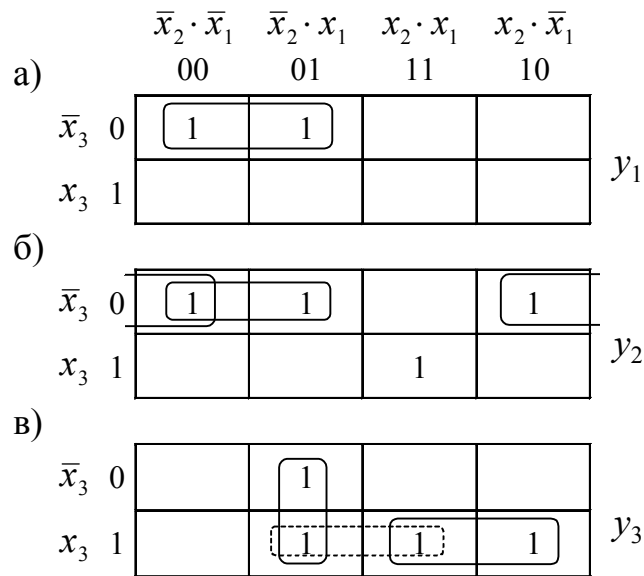


Рис.1.8. Карты Карно функций 3-х переменных

На рис.1.8 представлены карты для трёх функций:

$$y_1 = \bar{x}_3 \cdot \bar{x}_2,$$

$$y_2 = \bar{x}_3 \cdot \bar{x}_2 + \bar{x}_3 \cdot \bar{x}_1 + x_1 \cdot x_2 \cdot x_3,$$

$$y_3 = \bar{x}_2 \cdot x_1 + x_3 \cdot x_2,$$

где реализуемые операции склеивания показаны замкнутым контуром, охватывающим соседние единицы. Согласно карте, получим минимизированную формулу, эквивалентную исходной структурной. При этом учитываются операции склеивания и поглощения и осуществляется выбор нужного числа слагаемых. Для y_1 результатом минимизации являются общие значения аргументов (по горизонтали и по вертикали) единиц, охваченных контуром. Для y_2 имеем два контура, один из них расположен так же, как для функции y_1 , а второй охватывает клетки в крайних колонках (о склеи-

вании таких минтермов сказано выше). Для данного примера последнее слагаемое в y_2 соответствует единице в нижней части карты.

На карте Карно для y_3 (рис.1.8,в) штриховой линией показан дополнительный (лишний) контур склеивания, поскольку два других контура уже охватывают все единицы логической функции.

В минимизированной формуле при наличии на карте четырёх соседних клеток "выпадают" две переменные, а при наличии двух соседних клеток – одна переменная.

Примеры минимизации структурной формулы с помощью карт Карно для функций четырёх переменных

$$y_1 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 + x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 = \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4;$$

$$y_2 = \bar{x}_2 \cdot \bar{x}_3 \cdot \bar{x}_4 + \bar{x}_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_4;$$

$$y_3 = \bar{x}_2 \cdot \bar{x}_4;$$

$$y_4 = \bar{x}_2;$$

$$y_5 = x_3 \cdot \bar{x}_4 + \bar{x}_1 \cdot \bar{x}_4 + x_3 \cdot \bar{x}_2 \cdot x_1$$

приведены на рис.1.9.

Для функции y_1 имеется один контур склеивания, охватывающий единицы. Результатом минимизации являются общие значения аргументов клеток контура склеивания. Для y_2 в одной из клеток карты Карно расположена единица, не охваченная контурами склеивания. В этом случае, в выражении для минимизированной функции появляется логическое произведение значений аргументов, соответствующих единице в этой клетке. На рис.1.9,в-г контуры склеивания охватывают 4 и 8 единиц соответственно. При этом результатом минимизации являются общие, для клеток в охваченном контуре, значения аргументов. Карта Карно функции y_5 (рис.1.9,д) имеет такое расположение единиц, что можно реализовать три контура склеивания. Два из них охватывают по 4 единицы, а третий – две единицы. В результате, минимизированная функция содержит три слагаемых, каждое из которых представляет собой общие значения аргументов для клеток контуров склеивания.

На рис.1.10 представлена карта Карно для функций пяти переменных. Здесь условия склеивания выполняются также и для клеток, кодировка которых отличаются значениями одного из аргументов, но не расположенных рядом, а именно клеток, которые совпадают при совмещении половинок карт поворотом вокруг общего ребра (пунктирной линии). При

этом контур склеивания изображается, как показано на рис.1.10. Заметим, что на этом рисунке, в отличие от предыдущих, аргументы на карте Карно обозначены около левого верхнего угла карты.

Таким образом, для функций трёх или четырёх переменных следует использовать карты Карно, обеспечивающие простую и наглядную процедуру минимизации структурной формулы. В случае же большего числа переменных (пять и более) следует применять метод Квайна (или метод Квайна – Мак-Класки).

Процедуры минимизации структурных формул в совершенной конъюнктивной нормальной форме используют операции и приёмы, описанные выше для СДНФ. Проиллюстрируем их применение на следующих примерах. С помощью метода Квайна проведём минимизацию структурных формул для функций, заданных таблицей истинности (табл.1.8). Так как из правил эквивалентных преобразований следует $(x_1 + x_2) \cdot (x_1 + \bar{x}_2) = x_1$, то имеем:

$$y_1 = (x_1 + x_2 + x_3) \cdot (\bar{x}_1 + x_2 + x_3) = x_2 + x_3;$$

$$\begin{aligned} y_2 &= (\bar{x}_1 + x_2 + x_3) \cdot (\bar{x}_1 + x_2 + \bar{x}_3) \cdot (x_1 + \bar{x}_2 + \bar{x}_3) \cdot (\bar{x}_1 + \bar{x}_2 + \bar{x}_3) = \\ &= (\bar{x}_1 + x_2) \cdot (\bar{x}_1 + \bar{x}_3) \cdot (\bar{x}_2 + \bar{x}_3) = (\bar{x}_1 + x_2) \cdot (\bar{x}_2 + \bar{x}_3). \end{aligned}$$

Те же функции y_1 и y_2 можно минимизировать с помощью карт Карно (рис.1.11,а-б соответственно). Естественно при этом, будут получены те же выражения, что и методом Квайна.

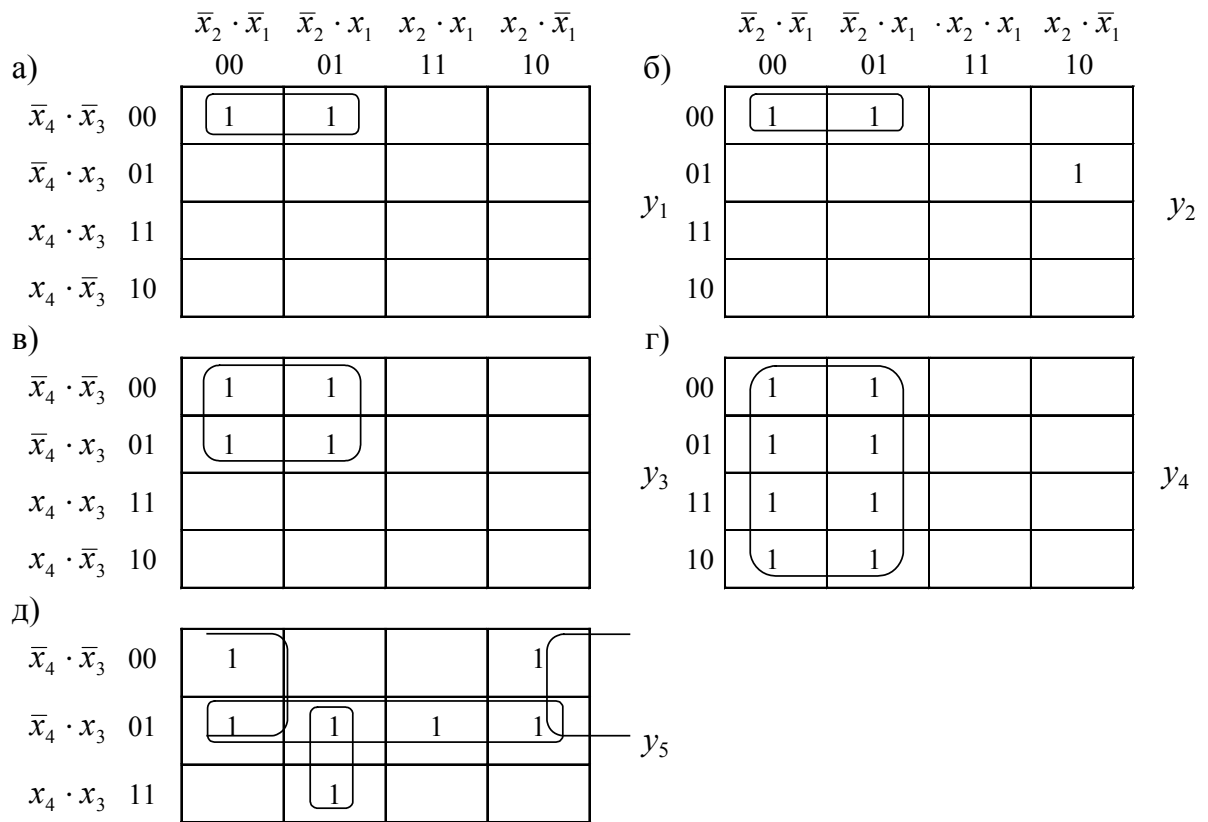


Рис.1.9. Карты Карно функций 4-х переменных.

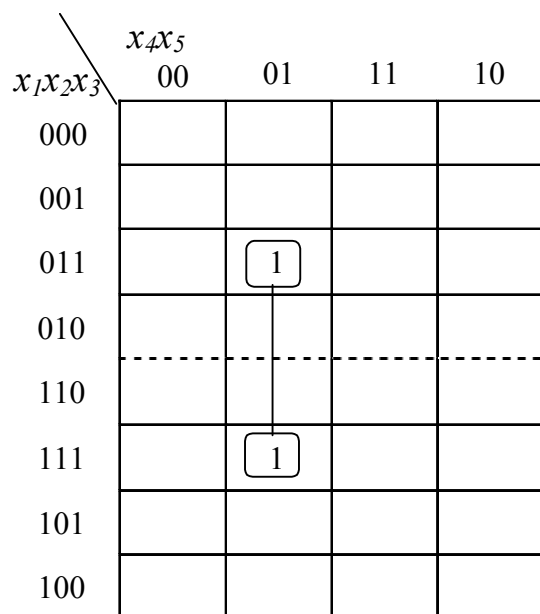


Таблица 1.8

x_3	x_2	x_1	y_1	y_2
0	0	0	0	1
0	0	1	0	0
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

Рис. 1.10. Карта Карно функции пяти переменных

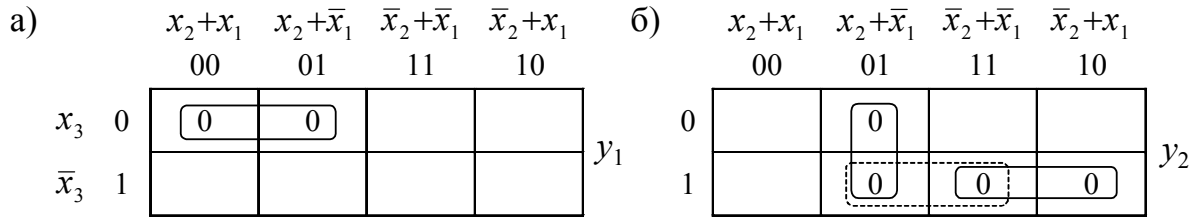


Рис. 1.11. Минимизация структурной формулы, заданной в СКНФ, для функции 3-х переменных

На карте Карно для совершенной конъюнктивной нормальной формы контуры склеивания охватывают клетки, в которых логическая функция имеет нулевые значения (единицы, при этом, на карте не изображаются). Правило получения результата склеивания и выявление лишних склеиваний те же, что и для совершенной дизъюнктивной нормальной формы.

Примеры использования карт Карно для минимизации функций четырёх переменных

$$y_1 = (\bar{x}_1 + x_4) \cdot (\bar{x}_1 + \bar{x}_2 + x_3),$$

$$y_2 = (x_2 + \bar{x}_3) \cdot (x_1 + \bar{x}_3) \cdot (\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + x_4)$$

приведены на рис.1.12,а-б соответственно. Для y_1 (рис.1.12,а) имеем два контура склеивания. Один контур охватывает четыре соседних клетки, в которых y_1 имеет значения 0, а другой – две клетки, расположенные в крайних строках. Минимизированное выражение для функции y_2 содержит результаты операции склеивания, соответствующие двум контурам, охватывающим по четыре 0 в соседних клетках (рис.1.12,б), и логическую сумму аргументов, соответствующую клетке с кодировкой 1011, в которой $y_2 = 0$.

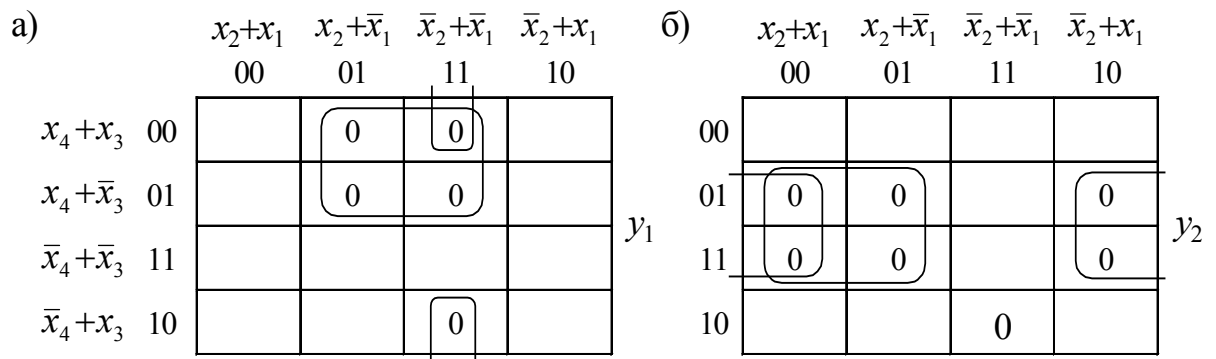


Рис. 1.12. Минимизация структурных формул, заданных в СКНФ для функций 4-х переменных

Дополнительные приёмы минимизации структурной формулы позволяют в ряде случаев упростить реализацию структурной формулы. Эти приёмы предполагают применение закона отрицания:

$$\underbrace{\bar{x}_1 \cdot \bar{x}_2}_{2 \text{ НЕ, 1 И}} = \overbrace{x_1 + x_2}^{1 \text{ НЕ, 1 ИЛИ}};$$

использование распределительного закона:

$$\underbrace{x_1 \cdot x_2 + x_1 \cdot x_3}_{2 \text{ И, 1 ИЛИ}} = \underbrace{x_1 \cdot (x_2 + x_3)}_{1 \text{ И, 1 ИЛИ}};$$

добавление в структурную формулу слагаемых, тождественно равных нулю:

$$\begin{aligned} y &= \underbrace{x_1 \cdot \bar{x}_2 + \bar{x}_1 \cdot x_2}_{2 \text{ НЕ, 2 И, 1 ИЛИ}} = x_1 \cdot \bar{x}_2 + \bar{x}_1 \cdot x_2 + x_1 \cdot \bar{x}_1 + x_2 \cdot \bar{x}_2 = \\ &= x_1 \cdot (\bar{x}_2 + \bar{x}_1) + x_2 \cdot (\bar{x}_2 + \bar{x}_1) = (x_1 + x_2) \cdot (\bar{x}_1 + \bar{x}_2) = \overbrace{x_1 \cdot x_2 \cdot (x_1 + x_2)}^{1 \text{ НЕ, 2 И, 1 ИЛИ}}. \end{aligned}$$

С левой стороны равенств под фигурными скобками приведено число и тип логических элементов, требуемых для реализации структурной формулы в исходной форме, а с правой стороны - после использования дополнительных приемов минимизации. Как следует из последнего выражения, после добавления слагаемых, тождественно равных нулю, применив дважды распределительный закон и закон отрицания, получим структурную формулу, содержащую на один логический элемент НЕ меньше, чем в исходной формуле. Видно, что при использовании дополнительных приемов минимизации имеет место уменьшение требуемого для реализации числа логических элементов. Применение таких приемов минимизации наиболее эффективным является при построении сложных комбинационных устройств, например, предназначенных для сопряжения протоколов работы различных телекоммуникационных систем.

1.4. Не полностью определённые логические функции и их минимизация

Не полностью (частично) определёнными логическими функциями называют функции, значения которых заданы лишь для части полного множества возможных наборов их аргументов [6, 18]. Рассмотрим ряд примеров.

Пример 1. Дано словесное описание устройства (рис.1.13): трёхрядный двоичный код (K), поступающий на вход, принимает значения K от 001 до 111, причём, если $K = 001$, то $y_1 = 1$, а если $K \neq 001$, то $y_1 = 0$. Заполним таблицу истинности для y_1 (табл.1.9). В первой строке можно поста-

вить значение y_1 как 0, так и 1, и это не будет противоречить словесному описанию. Следовательно, в первой строке имеет место неопределённость, что отмечается знаком Φ в таблице. Соответственно и y_1 не полностью определена при имеющимся в табл.1.9 наборе значений аргументов x_1, x_2, x_3 .

Пример 2. Для устройства, изображённого на рис.1.13, код K в двоичной системе счисления принимает значения от 000 до 101. При $K = 001$ и $K = 010$ значение функции $y_2 = 1$. При заполнении таблицы истинности (см. табл.1.9), неопределённость отмечена знаками Φ_1, Φ_2, Φ_3 .

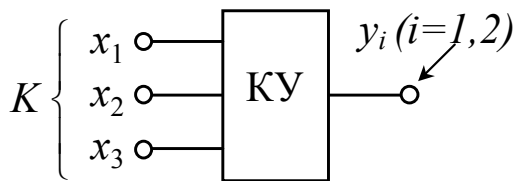


Рис.1.13. Функциональное обозначение комбинационного устройства в примерах

Таблица 1.9

x_3	x_2	x_1	y_1	y_2
0	0	0	Φ	Φ_1
0	0	1	1	1
0	1	0	0	1
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	Φ_2
1	1	1	0	Φ_3

Проведём минимизацию не полностью определённых логических функций, заданных в указанных выше примерах. При минимизации функции y_1 по методу Квайна положим значение $\Phi = 1$ и представим СДНФ в виде:

$$y_1 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 = \bar{x}_2 \cdot \bar{x}_3,$$

поскольку первое слагаемое является необязательным и может быть как 1, так и 0. Использование этого слагаемого позволяет упростить функцию y_1 .

Для функции y_2 второго примера предположим, что все значения

$\Phi_i = 1$, тогда получим:

$$y_2 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 + x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_1 \cdot x_2 \cdot \bar{x}_3 + \bar{x}_1 \cdot x_2 \cdot x_3 + x_1 \cdot x_2 \cdot x_3.$$

Здесь первое, четвёртое и пятое слагаемые являются необязательными. Используя операции склеивания применительно к первым двум, к первому и третьему, третьему и четвёртому, а также к четвёртому и пятому слагаемым, получаем:

$$y_2 = \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_1 \cdot \bar{x}_3 + \bar{x}_1 \cdot x_2 + x_2 \cdot x_3.$$

Если значение $\Phi_1 = 1$, а значения $\Phi_2 = \Phi_3 = 0$, то минимизированная функция y_2 имеет вид:

$$y_2 = \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_1 \cdot \bar{x}_3.$$

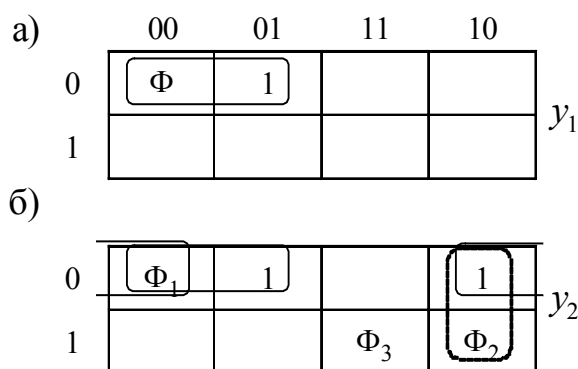


Рис. 1.14. Карты Карно для не полностью определённых логических функций 3-х переменных

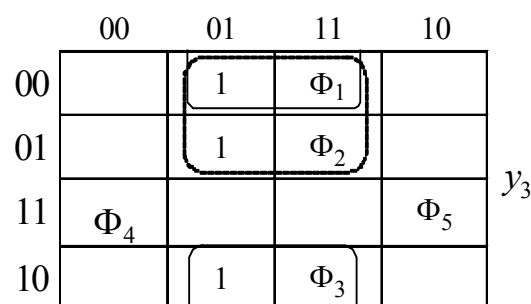


Рис.1.15.КартаКарно для не полностью определённых логических функций 4-х переменных

Процедуру минимизации рассмотренных выше функций y_1 и y_2 в примерах 1 и 2 можно провести и с использованием карт Карно (рис.1.14,а-б соответственно). Выбор значений Φ_i определяется расположением единиц в клетках карт. Пример минимизации с помощью карты Карно не полностью определённой функции y_3 четырёх переменных приведён на рис.1.15. Минимизированная функция y_3 в этом случае может быть записана в форме:

$$y_3 = x_1 \cdot \bar{x}_4 + x_1 \cdot \bar{x}_3.$$

1.6. Минимизация системы логических функций

Системой логических функций обычно описывается устройство с не-

сколькими выходами. Способ минимизации такой системы рассмотрим на примерах.

Пример 3. На рис.1.16,а приведено условное обозначение устройства, называемого полусумматором и предназначенного для сложения двух двоичных чисел с формированием суммы S и переноса в следующий разряд p . Процесс сложения двух двоичных чисел показан на рис.1.16,б, а таблица истинности выходных функций полусумматора, составленная в соответствии с правилами сложения, приведена на рис.1.16,в.

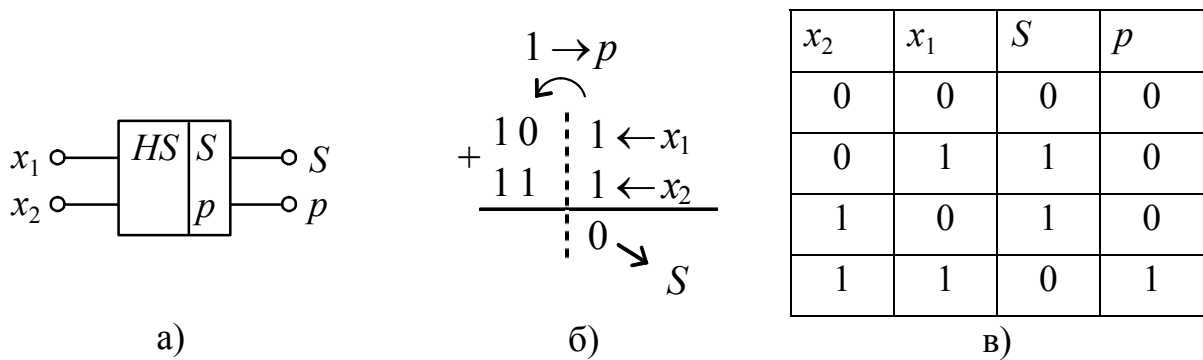


Рис. 1.16. Функциональное обозначение полусумматора (а), правило его работы (б) и таблица истинности (в)

Работа полусумматора описывается двумя логическими функциями:

$$S = f_1(x_1, x_2), p = f_2(x_1, x_2).$$

Структурные формулы для S и p имеют вид:

$$S = x_1 \cdot \bar{x}_2 + \bar{x}_1 \cdot x_2, p = x_1 \cdot x_2.$$

Непосредственное использование метода Квайна и карт Карно не позволяет провести дальнейшее упрощение функций. В этой связи можно попытаться использовать представление $S = S' = f_3(x_1, x_2, p)$.

Таблица 1.10.

p	x_2	x_1	S'
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	Φ_1
1	0	0	Φ_2
1	0	1	Φ_3

1	1	0	Φ_4
1	1	1	0

Составим таблицу истинности для S' как функции трёх переменных (табл.1.10), используя исходную таблицу истинности (рис.1.16,в) и правила работы устройства. В соответствии с этими правилами $p = 1$ в случае $x_1 = x_2 = 1$. Это означает, что не должна возникать комбинация входных сигналов $p x_1 x_2 - 011$. В таблице истинности такая комбинация отмечается знаком Φ_1 .

		$x_2 x_1$				
	p	00	01	11	10	
0			1	Φ_1	1	S'
1		Φ_2	Φ_3		Φ_4	

Рис.1.17. Карта Карно для логической функции S'

Аналогично знаками Φ_2, Φ_3, Φ_4 отмечаются комбинации входных сигналов 100, 101, 110. Карта Карно для полученной не полностью определённой логической функции $S' = f_3(x_1, x_2, p)$ приведена на рис.1.17. Сравнивая полученную структурную формулу $S' = x_1 \cdot \bar{p} + x_2 \cdot \bar{p} = (x_1 + x_2) \cdot \bar{p}$, где $p = x_1 x_2$, с исходной, видим, что при её реализации требуется на один элемент НЕ и на один элемент И меньше.

Пример 4. Построить одноразрядный комбинационный сумматор, т.е. устройство, предназначенное для сложения двух двоичных чисел, с учётом переноса из предыдущего разряда, с формированием значений суммы и переноса для следующего разряда. Условное обозначение, иллюстрация процесса суммирования и таблица истинности выходных функций сумматора даны на рис.1.18,а-в соответственно.

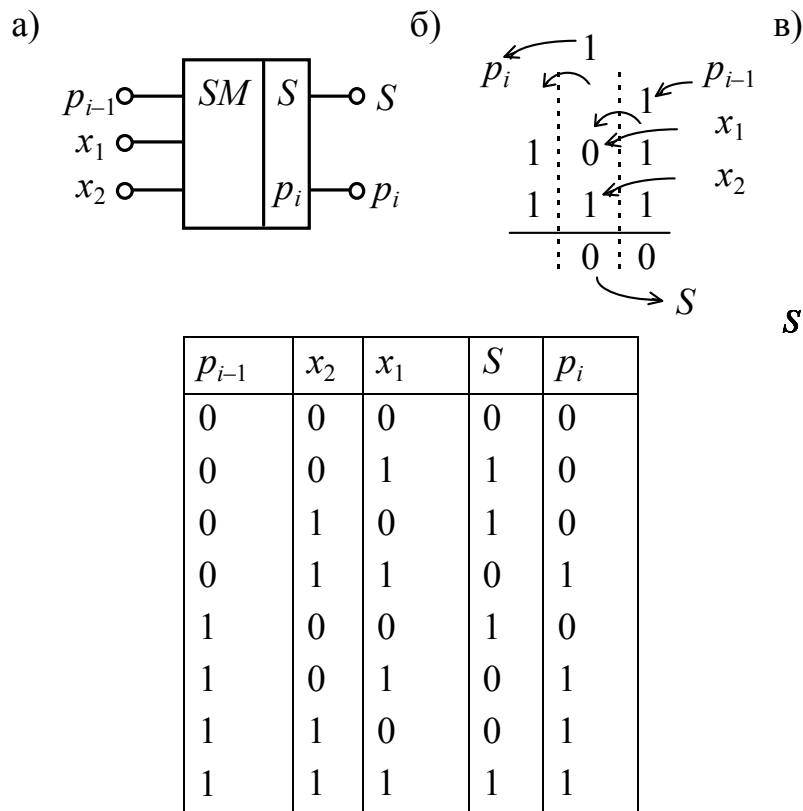


Рис. 1.18. Условное обозначение сумматора (а), иллюстрация процесса суммирования (б) и таблица истинности выходных функций (в)

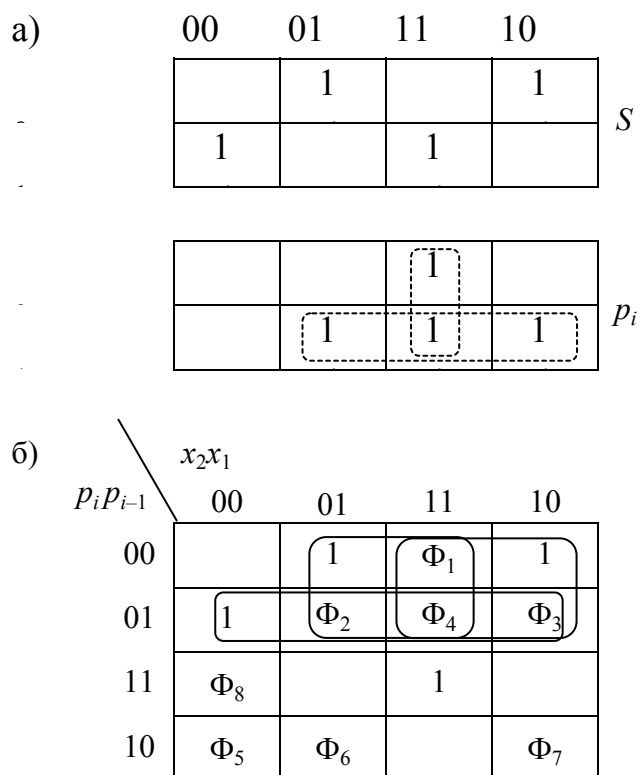


Рис. 1.19. Минимизация каждой выходной логической функции сумматора (а) и минимизация S' (б)

Проведём минимизацию каждой функции в отдельности с помощью

карт Карно для СДНФ (рис.1.19,а):

$$S = \bar{p}_{i-1} \cdot \bar{x}_2 \cdot x_1 + \bar{p}_{i-1} \cdot x_2 \cdot \bar{x}_1 + p_{i-1} \cdot \bar{x}_2 \cdot \bar{x}_1 + p_{i-1} \cdot x_2 \cdot x_1,$$

$$p_i = x_2 x_1 + p_{i-1} x_2 + p_{i-1} x_1$$

Таблица 1.11

p_i	p_{i-1}	x_2	x_1	S'
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	Φ_1
0	1	0	0	1
0	1	0	1	Φ_2
0	1	1	0	Φ_3
0	1	1	1	Φ_4
1	0	0	0	Φ_5
1	0	0	1	Φ_6
1	0	1	0	Φ_7
1	0	1	1	0
1	1	0	0	Φ_8
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Составим таблицу истинности для функции $S' = f(p_{i-1}, x_2, x_1, p_i)$, исходя из таблицы на рис.1.18,в и правил работы устройства, (табл.1.11). По карте Карно (рис.1.19,б) запишем формулу для не полностью определённой функции S' в виде:

$$S' = x_2 \cdot \bar{p}_i + x_1 \cdot \bar{p}_i + p_{i-1} \cdot \bar{p}_i + x_1 \cdot x_2 \cdot p_{i-1}.$$

Сравнивая между собой число логических элементов, необходимое для реализации S и S' , видим, что структурная формула S' реализуется меньшим числом элементов.

Пример 5. Построить цифровое устройство с двумя выходами, логические функции которых представлены с помощью карт Карно (рис.1.20,а-

б).

Проводя минимизацию по картам Карно каждой из функций, получим:

$$y_1 = x_4 \cdot \bar{x}_3 + x_4 \cdot x_2; \quad y_2 = \bar{x}_4 \cdot x_3 + x_3 \cdot x_2.$$

Реализация этих функций представлена на рис.1.20,в. Проведя совместную минимизацию логических функций и выделяя общую для обеих функций часть (на рис.1.20,а-б показано штриховой линией), получим следующие выражения:

$$y_1 = x_4 \cdot \bar{x}_3 + x_4 \cdot x_3 \cdot x_2; \quad y_2 = \bar{x}_4 \cdot x_3 + x_4 \cdot x_3 \cdot x_2.$$

При практической реализации этих логических функций (рис.1.20,г) требуется меньшее число элементов, чем при реализации устройства, в котором произведена независимая минимизация каждой функции в отдельности.

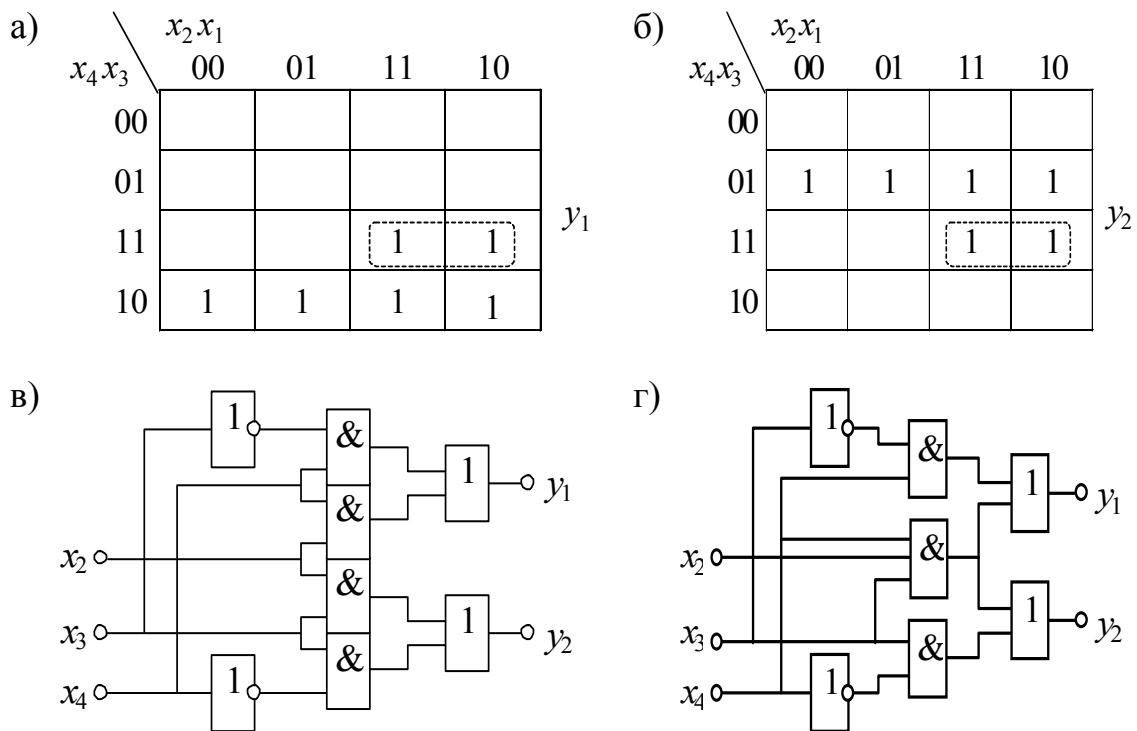


Рис.1.20. Минимизация системы двух логических функций: карты Карно (а,б), реализация минимизированных по отдельности логических функций (в), реализация системы логических функций с выделением общей части (г)

1.7. Универсальные логические элементы

Для удовлетворения требованиям однородности функциональной схемы и технологии необходимо для проектирования ИС применять универсальные логические элементы. К таким элементам относятся:

- Элемент Шеффера И-НЕ (рис.1.21,а). Реализуемая логическая функция может быть записана в виде:

$$y = \overline{x_1 \cdot x_2 \cdot \dots \cdot x_n}.$$

С помощью такого логического элемента можно построить логические функции НЕ, И, ИЛИ (рис.1.18,б-г соответственно).

- Элемент Пирса ИЛИ-НЕ. Логическая функция этого элемента может быть представлена в следующей форме:

$$y = \overline{x_1 + x_2 + \dots + x_n}.$$

Используя элемент ИЛИ-НЕ, также можно реализовать логические функции НЕ, И, ИЛИ.

Логические элементы И-НЕ или ИЛИ-НЕ обладают свойством двойственности. Действительно, заменяя в таблице истинности логического элемента И-НЕ (табл.1.12,а) символы 0 на 1 и, соответственно 1 на 0, получим табл.1.12,б. Сравнивая эту таблицу с таблицей истинности логического элемента ИЛИ-НЕ (табл.1.12,в), видим, что с точностью до перестановки строк они совпадают.

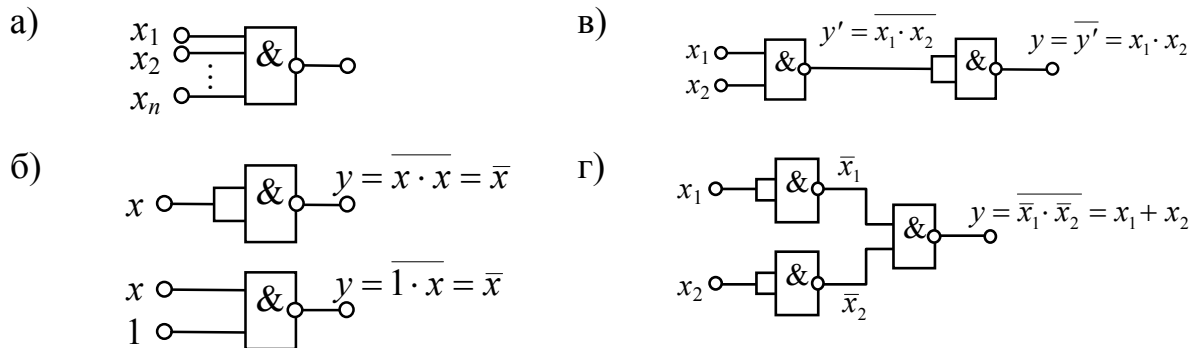


Рис.1.21. Условное обозначение элемента И-НЕ (а) и способы реализации с его помощью логических функций НЕ, И, ИЛИ (б-г)

Таблица 1.12

а)

x_2	x_1	y
0	0	1
0	1	1
1	0	1
1	1	0

б)

x_2	x_1	y
1	1	0
1	0	0
0	1	0
0	0	1

в)

x_2	x_1	y
0	0	1
0	1	0
1	0	0
1	1	0

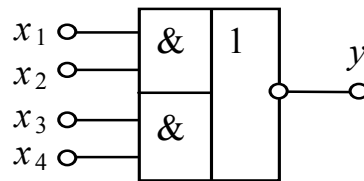


Рис.1.22. Условное обозначение элемента И-ИЛИ-НЕ

- Элемент И-ИЛИ-НЕ (рис.1.22). Реализуемая логическая функция имеет вид: $y = \overline{x_1 \cdot x_2 + x_3 \cdot x_4}$.

Так, например, если $x_1 = x_3 = 1$, то $y = \overline{x_2 + x_4}$. При $x_1 = 0, x_2 = 1$, имеем $y = \overline{x_3 \cdot x_4}$

1.8. Примеры построения комбинационных устройств на универсальных логических элементах

Рассмотрим прежде всего примеры построения комбинационных устройств на логических элементах И-НЕ.

Пример 6. Требуется построить устройство, реализующее логическую функцию вида:

$$y = x_1 + x_2 \cdot \bar{x}_3.$$

При использовании логических элементов И-НЕ получим схему, приведённую на рис.1.23,а. Как видно из рисунка, логические элементы И-НЕ, охваченные штриховой линией, являются избыточными. В этой связи, применяя законы отрицания, преобразуем логическую функцию, исключив из неё операции ИЛИ. Имеем:

$$y = x_1 + x_2 \cdot \bar{x}_3 = \overline{\overline{x_1 + x_2 \cdot \bar{x}_3}} = \overline{\bar{x}_1 \cdot x_2 \cdot \bar{x}_3}.$$

Схема, реализующая эту структурную формулу, приведена на рис.1.23,б.

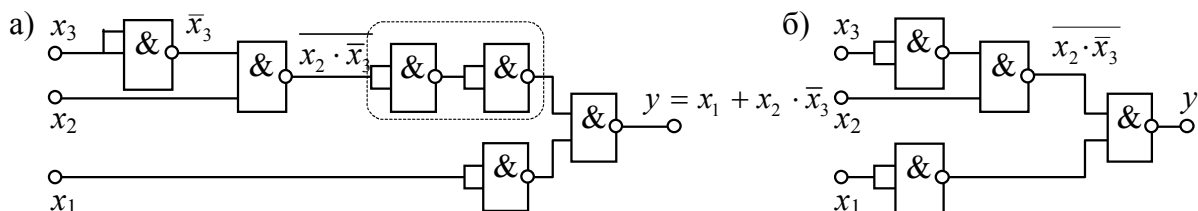


Рис. 1.23. Структурная схема устройства на элементах И-НЕ (а) и схема для структурной формулы, в которой исключена операция ИЛИ (б)

Из рассмотренного примера видно, что при реализации логической функции на логических элементах И-НЕ целесообразно использовать

представление этой функции в виде логической суммы (дизъюнктивной форме). Эта рекомендация подтверждается следующим примером.

Пример 7. Требуется построить устройство, реализующее на логических элементах И-НЕ логическую функцию вида:

$$y = x_1 \cdot x_2 + x_1 \cdot x_3$$

Для реализации этой логической функции преобразуем ее к виду:

$$y = x_1 \cdot x_2 + x_1 \cdot x_3 = \overline{\overline{x_1 \cdot x_2 + x_1 \cdot x_3}} = \overline{\overline{x_1 \cdot x_2} \cdot \overline{x_1 \cdot x_3}}$$

В этом случае требуется три логических элемента И-НЕ. Если же представить функцию в виде логического произведения (конъюнктивной форме), то получим:

$$y = x_1 \cdot (x_2 + x_3) = x_1 \cdot \overline{\overline{x_2 + x_3}} = x_1 \cdot \overline{\overline{x_2} \cdot \overline{x_3}}.$$

Для реализации этой логической функции потребуется уже пять элементов И-НЕ.

Пример 8. Синтезировать мажоритарный элемент (рис.1.24,а), представляющий собой устройство с нечётным числом входов, у которого $y = 1$ в случае, если на большинстве входов сигналы равны 1, и $y = 0$ при условии, что на большинстве входов сигналы равны 0.

Таблица 1.13

x_3	x_2	x_1	y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Составив в соответствии со словесным описанием таблицу истинности (табл.1.13) получим с помощью карты Карно (рис.1.24,б) минимизированную функцию

$$y = x_1 \cdot x_3 + x_1 \cdot x_2 + x_2 \cdot x_3.$$

Исключая операции ИЛИ, имеем:

$$y = \overline{\overline{x_1 \cdot x_3 + x_1 \cdot x_2 + x_2 \cdot x_3}} = \overline{\overline{x_1 \cdot x_3} \cdot \overline{x_1 \cdot x_2} \cdot \overline{x_2 \cdot x_3}} .$$

Реализация этого выражения для логической функции возможна на логических элементах И-НЕ.

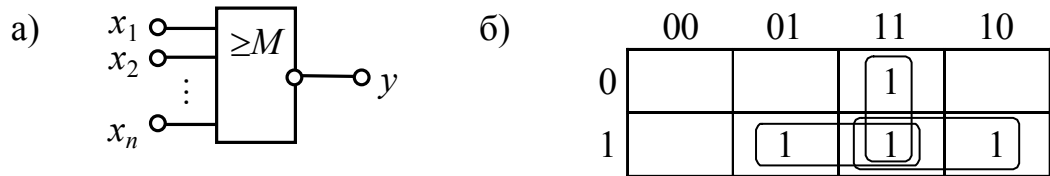


Рис .1.24. Условное обозначение мажоритарного элемента (а) и минимизация его логической функции (б)

Пример 9. Построить одноразрядный комбинационный сумматор на логических элементах И-НЕ.

Как следует из раздела 1.6, имеем:

$$p_i = x_2 \cdot x_1 + p_{i-1} \cdot x_1 + p_{i-1} \cdot x_2 ,$$

$$S = p_{i-1} \cdot \bar{p}_i + x_2 \cdot \bar{p}_i + x_1 \cdot \bar{p}_i + x_1 \cdot x_2 \cdot p_{i-1} .$$

После соответствующих преобразований, применив правило де Моргана, получим:

$$p_i = \overline{\overline{x_2 \cdot x_1} \cdot \overline{p_{i-1} \cdot x_1} \cdot \overline{p_{i-1} \cdot x_2}} ,$$

$$S = \overline{\overline{p_{i-1} \cdot \bar{p}_i} \cdot \overline{x_2 \cdot \bar{p}_i} \cdot \overline{x_1 \cdot \bar{p}_i} \cdot \overline{x_1 \cdot x_2 \cdot p_{i-1}}} .$$

При построении комбинационных устройств на элементах ИЛИ-НЕ логическую формулу необходимо преобразовать таким образом, чтобы исключить логическую операцию И.

Пример 10. Требуется построить устройство, реализующее на логических элементах ИЛИ-НЕ логическую функцию вида:

$$y = x_1 \cdot x_2 + x_1 \cdot x_3$$

Представим эту логическую функцию в форме:

$$y = x_1 \cdot x_2 + x_1 \cdot x_3 = \overline{\overline{x_1 \cdot x_2} \cdot \overline{x_1 \cdot x_3}} = \overline{\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_1} \cdot \overline{x_3}} .$$

Для её реализации требуется семь логических элемента ИЛИ-НЕ. Записывая функцию в виде логического произведения, имеем:

$$y = x_1 \cdot (x_2 + x_3) = \overline{\overline{x_1} \cdot \overline{(x_2 + x_3)}} = \overline{\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3}} = \overline{\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3}} .$$

Для реализации такой функции потребуется три логических элемента ИЛИ-НЕ.

Из рассмотренного примера видно, что при синтезе устройств на логических элементах ИЛИ-НЕ целесообразно использовать представление логической функции в виде логического произведения.

Пример 11. Провести синтез мажоритарного элемента (см. пример 8).

По табл.1.13 с помощью карт Карно получим минимизированную функцию и запишем её, используя операции ИЛИ-НЕ:

$$y = (x_2 + x_3) \cdot (x_1 + x_2) \cdot (x_1 + x_3) = \\ = \overline{\overline{(x_2 + x_3)} \cdot \overline{(x_1 + x_2)} \cdot \overline{(x_1 + x_3)}} = \overline{x_2 + x_3 + x_1 + x_2 + x_1 + x_3}.$$

Представляя функцию в виде логической суммы, имеем:

$$y = \overline{\overline{x_1 \cdot x_2} + \overline{x_1 \cdot x_3} + \overline{x_2 \cdot x_3}} = \overline{\overline{\overline{x_1 \cdot x_2}} + \overline{\overline{x_1 \cdot x_3}} + \overline{\overline{x_2 \cdot x_3}}} = \overline{\overline{\overline{x_1} \cdot \overline{\overline{x_2}}} + \overline{\overline{\overline{x_1} \cdot \overline{\overline{x_3}}} + \overline{\overline{\overline{x_2} \cdot \overline{\overline{x_3}}}}}$$

Очевидно, что во втором случае для реализации логической функции потребуется большее число логических элементов.

1.9. Общая задача синтеза комбинационных устройств

Исходными данными для проектирования комбинационных устройств являются их словесное описание и требования к основным электрическим параметрам. В задачу синтеза комбинационных устройств входит получение структурной схемы минимальной сложности, реализованной на заданном или выбранном типе логических элементов.

Первый этап - составление таблицы истинности по словесному описанию.

Назначение *второго этапа* – получение структурной формулы логической функции, реализуемой комбинационным устройством, причём на этом этапе стремятся получить наиболее простое (минимизированное) логическое выражение заданной функции.

В большинстве случаев процесс упрощения (минимизации) сводится к применению операций склеивания и поглощения. При использовании алгебраических методов минимизации – методов Квайна и Квайна–МакКласки – по таблице истинности записывается СДНФ (или СКНФ) и производятся склеивания имеющихся произведений (или сумм) [1, 6, 10, 18]. При графическом методе, использующем карты Карно [1, 9, 10, 18], по таблице истинности с помощью карты Карно выявляются все возможные процедуры склеивания и поглощения и записываются в минимизирован-

ную структурную формулу. После получения минимизированной функции с помощью указанных методов дальнейшее упрощение структурной формулы происходит путём использования дополнительных приёмов минимизации. К ним относятся: применение закона отрицания, использование распределительного закона и добавление слагаемых, тождественно равных нулю.

Третий этап: – запись полученной минимизированной структурной формулы в виде комбинации операций, выполняемых заданным (выбранным) типом логических элементов. Эта запись производится с помощью соответствующих приёмов. Так, при использовании логических элементов И-НЕ, над полученной функцией, представленной в совершенной дизъюнктивной нормальной форме, ставят два знака инверсии и с помощью закона отрицания преобразуют в инверсию функции, представленной в СКНФ. В результате получается логическое выражение, содержащее только операции И-НЕ.

Четвёртый этап: - составление структурной схемы. На этом этапе каждой логической операции преобразованной структурной формулы ставят в соответствие определённый логический элемент заданного типа. На основе структурной формулы осуществляют необходимые соединения между элементами.

В общем случае реализация полученной минимизированной функции имеет много вариантов, определяемых типом используемых элементов и предъявляемыми требованиями к устройству (быстродействие, простота реализации, стоимость и др.) Проиллюстрируем возможное многообразие решений задачи построения комбинационного устройства.

Пример 12. Построить схему, реализующую заданную минимизированную функцию $y = \bar{a} \cdot \bar{c} + \bar{b} \cdot \bar{c} + \bar{d}$ на различных логических элементах, при обеспечении минимальных аппаратных затрат и максимального быстродействия. Аппаратные затраты (W) оценить по числу применяемых корпусов микросхем, а быстродействие – по времени задержки (T) распространения сигнала в электрической схеме. При использовании логических элементов одной серии положить, что задержки сигналов в логических элементах различного вида одинаковы и равны некоторой усреднённой для данной серии микросхем величине τ .

В большинстве серий ИС в одном корпусе может быть размещено 6 инверторов (ЛН), либо 4 двухвходовых элемента одного из типов: И–НЕ (ЛА), ИЛИ–НЕ (ЛЕ), И (ЛИ), ИЛИ (ЛЛ). Также в одном корпусе может

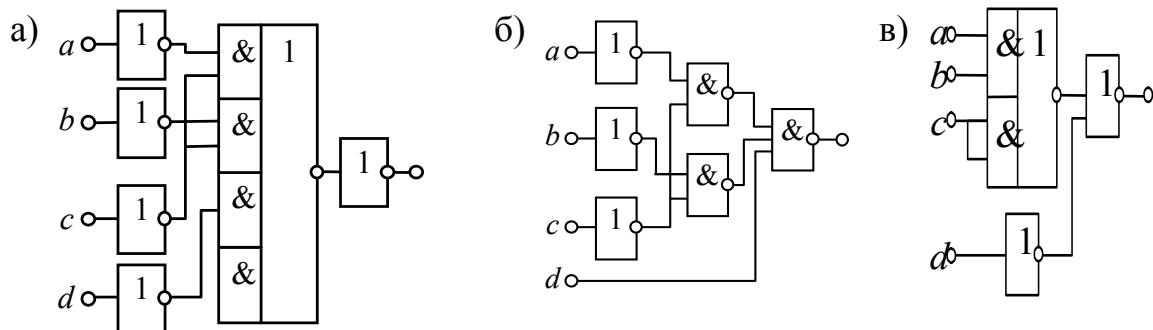
быть размещено 3 трехходовых элемента одного из указанных типов, либо 2 четырехходовых элемента или 1 восьмивходовой. В зависимости от числа входов логического элемента И-ИЛИ-НЕ (ЛР) в корпусе могут быть размещены один или два элемента такого типа.

Схемная реализация заданной формулы с использованием микросхем серии К155 или КР1533 [18.24.25] представлена на рис.1.25,а. Поскольку задержки распространения сигнала в инверторе и элементе И-ИЛИ-НЕ одинаковы, то задержка сигналов T во всей схеме равна 3τ . Схема состоит из пяти логических элементов НЕ, каждый из которых занимает 1/6 корпуса микросхемы К155ЛН1, и элемента И-ИЛИ-НЕ, который занимает корпус микросхемы К155ЛР3. Таким образом, аппаратные затраты W можно оценить следующим образом: $W = 5 \cdot 1/6 + 1 = 22/12$. Неиспользованные элементы частично занятого корпуса (в данном случае шестой логический элемент НЕ в корпусе микросхем ЛН1) не учитываются. Поскольку у микросхемы К155ЛР3 часть входов не используется, рациональнее реализовывать схему с помощью элементов И-НЕ, имеющих меньшее число входов. Применяя законы отрицания к заданной формуле, получим:

$$y = \overline{\overline{\overline{a \cdot c} + \overline{b \cdot c} + \overline{d}}} = \overline{\overline{a \cdot c} \cdot \overline{b \cdot c} \cdot \overline{d}}.$$

При реализации функции этого вида (рис.1.25,б) задержка сигнала $T = 3\tau$ и аппаратные затраты составляют $W = 3 \cdot 1/6 + 2 \cdot 1/4 + 1 \cdot 1/3 = 16/12$. Продолжая преобразование заданной логической функции, получим (рис.1.25,в):

$$y = \overline{a \cdot c} + \overline{b \cdot c} + \overline{d} = (\overline{a} + \overline{b}) \cdot \overline{c} + \overline{d} = \overline{\overline{\overline{a} + \overline{b}} + c} + \overline{d} = \overline{a \cdot b} + c + \overline{d}.$$



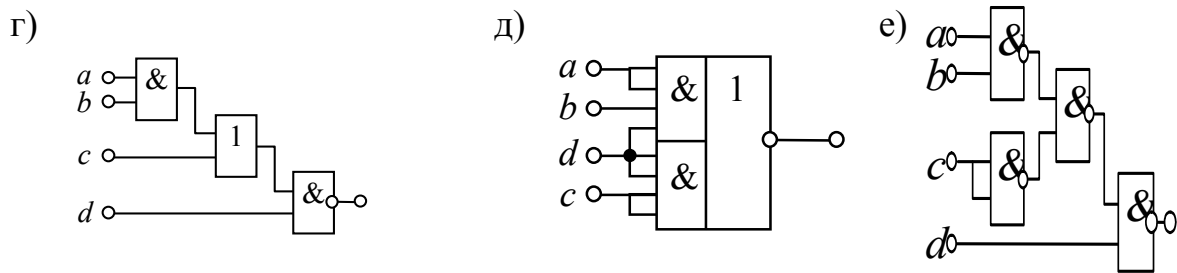


Рис. 1.25. Варианты реализации заданной логической функции

В этом случае $T = 2\tau$ и $W = 11/12$.

Также легко можно получить ещё один вид структурной формулы:

$$y = \overline{a \cdot b + c} + \bar{d} = \overline{(a \cdot b + c) \cdot d},$$

при реализации которой (рис.1.25,г) требуется меньше корпусов микросхем ($W = 9/12$) при $T = 3\tau$. При дальнейшем преобразовании формулы получим:

$$y = \overline{a \cdot b \cdot d + c \cdot d}.$$

Структурная схема представлена на рис.1.25,д. Здесь $W = 12/12$ и $T = \tau$. При стремлении реализовать заданную функцию с помощью одного типа логических элементов её можно записать в виде

$$y = \overline{\overline{a \cdot b \cdot c} + \bar{d}} = \overline{\overline{\overline{a \cdot b \cdot c} \cdot d}}.$$

В структурной схеме (рис.1.25,е) требуется один корпус двухвходовых элементов И-НЕ (т.е. $W = 12/12$) при $T = 3\tau$. Таким образом, как видно из рассмотренного выше примера, возможны различные схемные реализации одной и той же структурной формулы, полученной в результате использования процедуры минимизации.

В том случае, когда логическая функция является не полностью определённой, на втором этапе синтеза функцию доопределяют таким образом, чтобы максимально упростить структурную формулу. При синтезе комбинационного устройства с несколькими выходами требуется минимизировать систему логических функций, описывающих работу такого устройства. В этом случае стремятся так выделить в функциях общие составляющие, чтобы с соответствующих узлов устройства можно было бы осуществить разветвление сигнала по нескольким направлениям.

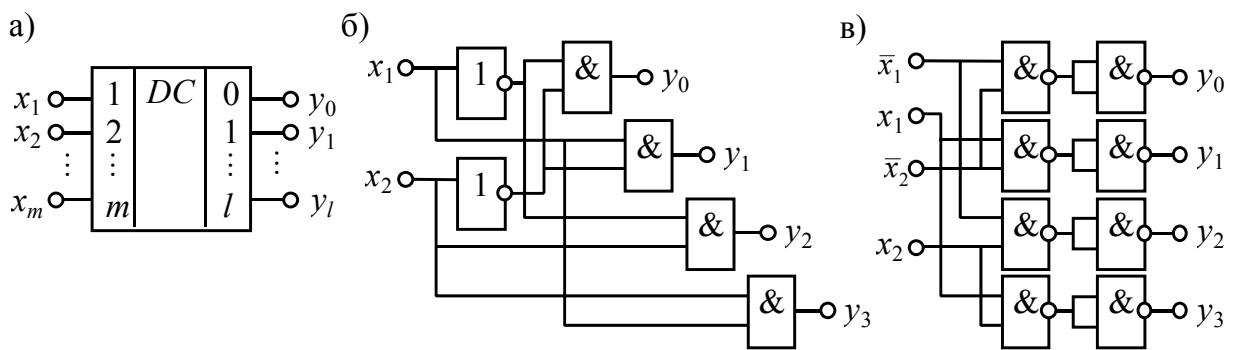
1.10 Типовые комбинационные устройства

1.10.1. Дешифраторы и шифраторы

Дешифратором называется устройство, содержащее m входов и $l+1 = 2^m$ выходов (рис.1.26,а). В дешифраторе в зависимости от значения входного m -разрядного двоичного кода появляется сигнал 1 на том выходе, номер которого в десятичной системе счисления соответствует номеру двоичного кода на входе. На всех остальных выходах дешифратора при этом присутствует сигнал 0. Такой выходной код называется унитарным.

Пример 13. Построить дешифратор на универсальных логических элементах для $m=2$.

В этом случае, очевидно, $l = 2^m - 1 = 3$. По таблице истинности (табл.1.14) получим:



$$y_0 = \bar{x}_1 \cdot \bar{x}_2, y_1 = x_1 \cdot \bar{x}_2, y_2 = \bar{x}_1 \cdot x_2, y_3 = x_1 \cdot x_2.$$

Рис.1.26. Условное обозначение дешифратора (а), его реализация на логических элементах И, НЕ, ИЛИ (б) и на элементах И-НЕ (в) для $m = 2$ и $l = 3$

Таблица 1.14

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

Построение дешифратора, реализующего указанные логические функции, приведено на рис.1.26,б.

Для реализации дешифратора на элементах И-НЕ представим систему логических функций в виде:

$$y_0 = \overline{x_1} \cdot \overline{x_2} = \overline{\overline{\overline{x_1} \cdot \overline{x_2}}},$$

$$y_1 = x_1 \cdot \overline{x_2} = \overline{\overline{x_1 \cdot \overline{x_2}}},$$

$$y_2 = \overline{x_1} \cdot x_2 = \overline{\overline{\overline{\overline{x_1} \cdot x_2}}},$$

$$y_3 = x_1 \cdot x_2 = \overline{\overline{x_1 \cdot x_2}}.$$

На рис.1.26,в приведено построение такого дешифратора.

Пример 14. Построить дешифратор, имеющий шесть входов ($m=6$) и 64 выхода. Таблица истинности такого дешифратора приведена в табл.1.15.

Таблица 1.15

x_6	x_5	x_4	x_3	x_2	x_1	y_0	y_1	\dots	y_{63}
0	0	0	0	0	0	1	0		0
0	0	0	0	0	1	0	1		0
—	—	—	—	—	—	—	—	—	—
—	—	—	—	—	—	—	—	—	—
1	1	1	1	1	1	0	0	\dots	1

Имеем:

$$y_0 = \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \cdot \overline{x_5} \cdot \overline{x_6} = \overline{\overline{\overline{\overline{\overline{\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \cdot \overline{x_5} \cdot \overline{x_6}}}}}}},$$

$$y_1 = x_1 \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \cdot \overline{x_5} \cdot \overline{x_6} = \overline{\overline{x_1 \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} \cdot \overline{x_5} \cdot \overline{x_6}}},$$

.....

$$y_{63} = x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_5 \cdot x_6 = \overline{\overline{x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_5 \cdot x_6}}.$$

Для реализации такого дешифратора требуется 64 корпуса интегральных схем, содержащих 8-ми входовые элементы И-НЕ и 16 корпусов ИС, содержащих по 4 двухвходовых элемента И-НЕ в каждом корпусе. Уменьшение числа корпусов интегральных микросхем можно осуществить путём перехода к двухступенчатой схеме, в которой на первой ступени используются два дешифратора с меньшим значением m , например с $m = 3$.

Таблица истинности для первого дешифратора дана в табл.1.16.

Таблица 1.16

x_3	x_2	x_1	y'_0	y'_1	\dots	y'_7
0	0	0	1	0		0

0	0	1	0	1		0
—	—	—	—	—	—	—
1	1	1	0	0	...	1

Структурные формулы, построенные по этой таблице, имеют следующий вид:

$$y'_0 = \bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3 = \overline{\overline{\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_3}},$$

$$y'_1 = x_1 \cdot \bar{x}_2 \cdot \bar{x}_3 = \overline{\overline{x_1 \cdot \bar{x}_2 \cdot \bar{x}_3}},$$

.....

$$y'_7 = x_1 \cdot x_2 \cdot x_3 = \overline{\overline{x_1 \cdot x_2 \cdot x_3}}.$$

Аналогично для второго дешифратора имеем:

$$y''_0 = \bar{x}_4 \cdot \bar{x}_5 \cdot \bar{x}_6 = \overline{\overline{\bar{x}_4 \cdot \bar{x}_5 \cdot \bar{x}_6}},$$

$$y''_1 = x_4 \cdot \bar{x}_5 \cdot \bar{x}_6 = \overline{\overline{x_4 \cdot \bar{x}_5 \cdot \bar{x}_6}},$$

$$\dots\dots\dots y''_7 = x_4 \cdot x_5 \cdot x_6 = \overline{\overline{x_4 \cdot x_5 \cdot x_6}} \dots$$

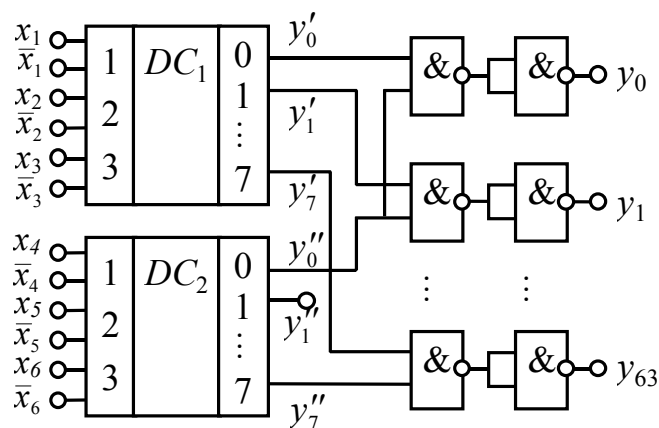


Рис.1.27. Структурная схема двухступенчатого дешифратора

Тогда:

$$y_0 = y'_0 \cdot y''_0 = \overline{\overline{y'_0 \cdot y''_0}},$$

$$y_1 = y'_1 \cdot y''_0 = \overline{\overline{y'_1 \cdot y''_0}},$$

.....

$$y_{63} = y'_7 \cdot y''_7 = \overline{\overline{y'_7 \cdot y''_7}}.$$

Построение такого двухступенчатого дешифратора приведено на рис.1.27. Для реализации дешифратора DC1 требуется 3 корпуса инте-

гральных схем, содержащих по 3 трёхвходовых элемента И-НЕ, и 2 корпуса интегральных схем с 4-мя двухвходовыми элементами И-НЕ. Аналогичное число микросхем требуется для реализации дешифратора DC₂. Таким образом, для реализации двухступенчатого дешифратора потребуется на 38 корпусов схем меньше, чем для реализации дешифратора, построенного по одноступенчатой схеме.

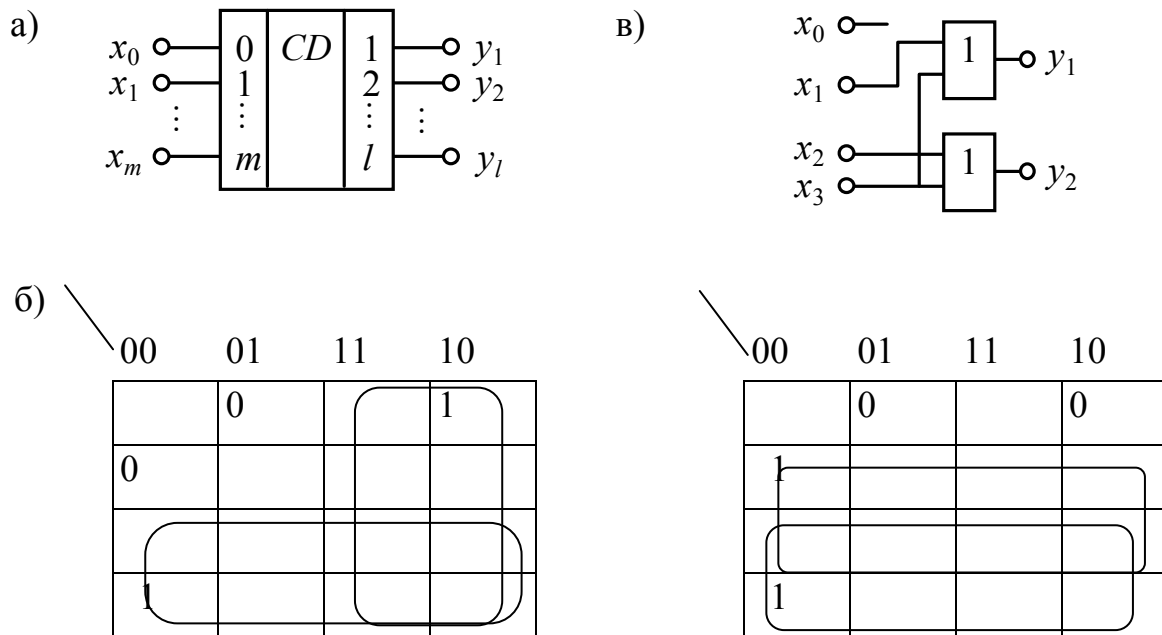


Рис.1.28. Условное обозначение шифратора (а), таблицы истинности выходных функций шифратора (б), структурная схема простейшего шифратора (в)

Таблица 1.17

x_3	x_2	x_1	x_0	y_2	y_1
0	0	0	0	Φ_1	Φ_2
0	0	0	1	0	0
0	0	1	0	0	1
0	0	1	1	Φ_3	Φ_4
0	1	0	0	1	0
0	1	0	1	Φ_5	Φ_6
0	1	1	0	Φ_7	Φ_8
0	1	1	1	Φ_9	Φ_{10}
1	0	0	0	1	1
1	0	0	1	Φ_{11}	Φ_{12}

1	0	1	0	Φ_{13}	Φ_{14}
1	0	1	1	Φ_{15}	Φ_{16}
1	1	0	0	Φ_{17}	Φ_{18}
1	1	0	1	Φ_{19}	Φ_{20}
1	1	1	0	Φ_{21}	Φ_{22}
1	1	1	1	Φ_{23}	Φ_{24}

Ш и ф р а т о р выполняет функцию, обратную дешифратору, т.е. преобразует унитарный код на входе в l -разрядный двоичный код на выходе. При наличии сигнала логической 1 на одном из входов, на l выходах шифратора появляется l -разрядный код, значение которого в двоичной системе счисления равно номеру входа, где присутствует сигнал 1 (рис.1.28,а). Число входов $m + 1$ и выходов l шифратора связано соотношением $2^l = m + 1$. Например, при $m = 3$ и $l = 2$ таблицу истинности можно записать в форме табл.1.17. Составляя для y_2 и y_1 карты Карно (рис.1.28,б), после выполнения процедуры минимизации структурных формул, имеем:

$$y_1 = x_1 + x_3, \quad y_2 = x_2 + x_3.$$

Реализация этих формул, представлена на рис.1.28,в. Для шифратора с $m = 7$ и $l = 3$ по таблице истинности (табл.1.18, где N – номер входа) запишем структурные формулы:

$$y_1 = x_1 + x_3 + x_5 + x_7,$$

$$y_2 = x_2 + x_3 + x_6 + x_7,$$

$$y_3 = x_4 + x_5 + x_6 + x_7.$$

Таблица 1.18

N	y_3	y_2	y_1
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Реализация такого дешифратора, так же как и в предыдущем примере,

осуществляется с помощью логических элементов ИЛИ.

1.10.2. Кодопреобразователи

В цифровых радиотехнических устройствах часто возникает необходимость в преобразовании в общем случае по произвольному правилу m -разрядных чисел одной двоичной системы в m -разрядные числа другой системы. Эта задача решается с помощью устройств, называемых кодопреобразователями. Например, правило преобразования заданных чисел с помощью кода Грея в двоичный код может быть представлено табл.1.19, где $x_2 x_1$ – код Грея.

При построении кодопреобразователей обычно используется последовательное соединение дешифратора (DC) и шифратора (CD), причём соединения входов шифратора с выходами дешифратора определяются правилом преобразования кодов. Так, для кодопреобразователя, работа которого задана табл.1.19, схема соединений выходов дешифратора и входов шифратора приведена на рис.1.29,а.

Если правило преобразования задано, например, табл.1.20, то схема соединений будет иметь вид, представленный на рис 1.29,б. При таком построении кодопреобразователей в случае изменения правил преобразования изменяются лишь соединения выходов дешифратора с входами шифратора.

Таблица 1.19

x_2	x_1	x'_2	x'_1
0	0	0	0
0	1	0	1
1	1	1	0
1	0	1	1

Таблица 1.20

x_3	x_2	x_1	x'_3	x'_2	x'_1
0	0	0	0	0	0
0	0	1	0	0	1
0	1	1	0	1	0
0	1	0	0	1	1
1	1	0	1	0	0
1	1	1	1	0	1
1	0	1	1	1	0
1	0	0	1	1	1

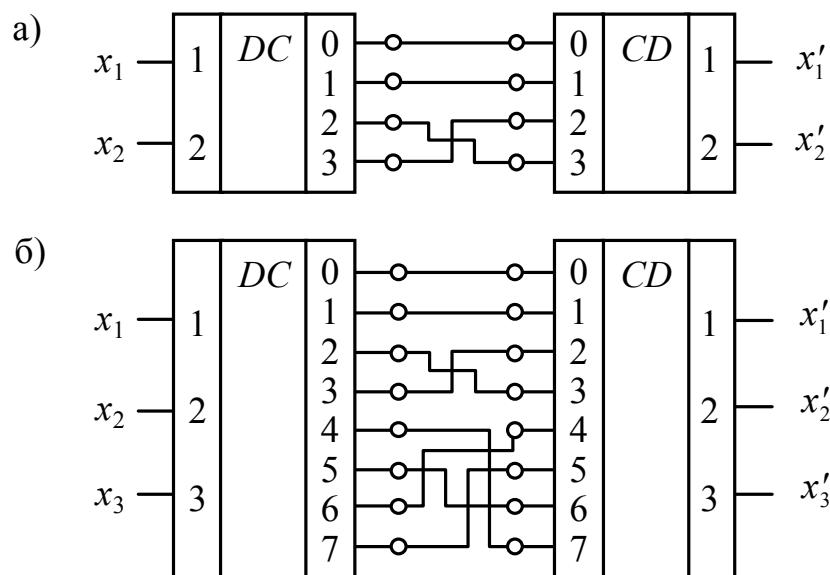


Рис.1.29. Структурные схемы кодопреобразователей

1.10.3. Мультиплексоры и демультиплексоры

Особую группу комбинационных устройств составляют логические коммутаторы. К ним относятся мультиплексоры, осуществляющие передачу одного из нескольких входных сигналов на выход (единственный), и демультиплексоры, реализующие передачу входного сигнала (единственного) на один из нескольких выходов. Кроме того, они используются для синтеза логических функций многих переменных. При этом число корпусов интегральных схем может быть значительно уменьшено.

Мультиплексоры. Мультиплексор является устройством, которое в зависимости от значения кода на адресных входах (A_0, A_1, \dots, A_M) осуществляет выбор одного из информационных входов (D_0, D_1, \dots, D_N) и направленную передачу сигнала на выход (Q). Мультиплексор можно представить как бесконтактный многопозиционный переключатель. На рис.1.30,а показано обозначение мультиплексора со стробированием по входу C ¹⁾ на функциональных схемах. Каждому информационному входу мультиплексора присваивается номер, называемый адресом. При разрешающем (стробирующем) сигнале на входе C мультиплексор подключает информационный вход, адрес которого соответствует двоичному коду на адресных входах, к выходу. Изменяя сигналы на адресных входах, можно

¹⁾ Заметим, что существуют интегральные схемы мультиплексоров без стробирования. Наличие стробирующего входа расширяет функциональные возможности мультиплексора и позволяет синхронизировать его работу с работой других цифровых устройств.

обеспечить передачу сигналов с различных информационных входов на выход. Число информационных входов $N + 1$ и число адресных входов $M + 1$ связаны соотношением: $N + 1 = 2^{M+1}$.

Работу мультиплексора, например, с 2-мя адресными и 4-мя информационными входами можно описать таблицей истинности (рис.1.30,б). При отсутствии стробующего сигнала ($C = 0$) связь между информационными входами и выходом отсутствует (при этом $Q = 0$). При наличии стробующего сигнала ($C = 1$) на выход передаётся логический уровень того из информационных входов D_i , номер которого i в двоичной системе счисления задан кодом на адресных входах. Так, если значения сигналов на входах A_1A_0 соответствуют значениям 10 в двоичной системе счисления, то на выход Q будет передаваться сигнал с входа D_2 .

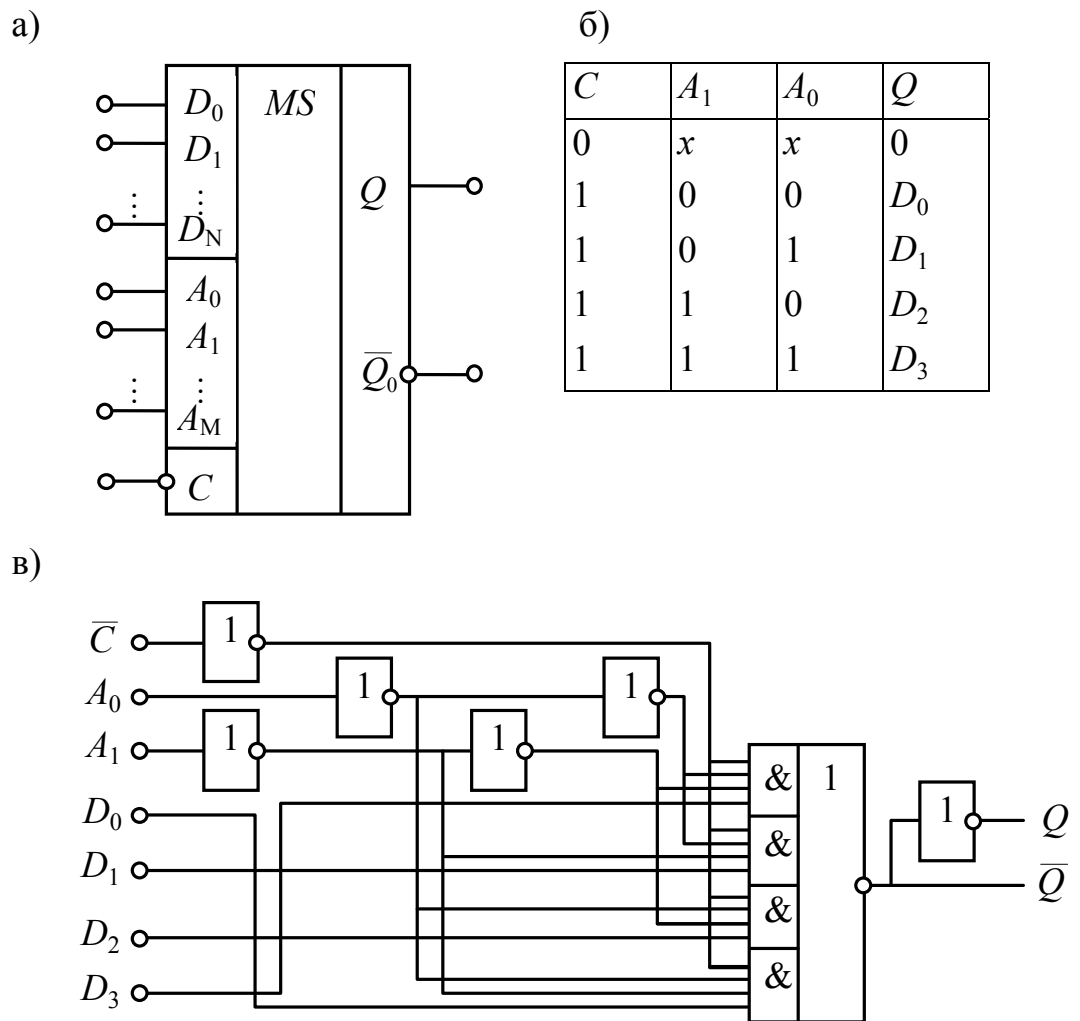


Рис.1.30. Обозначение мультиплексора на функциональных схемах (а), таблица истинности мультиплексора с 2-мя адресными входами (б) и его структурная схема (в)

По таблице истинности можно записать логическую функцию для вы-

ходного сигнала мультиплексора. Структурная схема такого мультиплексора приведена на рис.1.30,в.

Мультиплексоры могут быть использованы для синтеза комбинационных устройств, реализующих логические функции многих переменных. При этом мультиплексор работает как универсальный логический элемент, реализующий любую логическую функцию, содержащую до $M + 1$ переменных, где $M + 1$ – число адресных входов мультиплексора [1, 6, 18, 23]. Один мультиплексор в этом случае заменяет несколько корпусов интегральных схем с логическими элементами И, ИЛИ, НЕ.

При синтезе комбинационных устройств на мультиплексорах исходной является карта Карно заданной функции y . Число информационных входов мультиплексора выбирают равным числу клеток карты Карно. Если двоичные коды на адресных входах мультиплексора и коды клеток карт Карно совпадают, а информационные сигналы определяются значениями 0 или 1, записанными в клетках карты Карно, то такой мультиплексор полностью воспроизводит карту Карно, т.е. заданное комбинационное устройство.

Рассмотрим примеры применения мультиплексоров при синтезе комбинационных устройств и некоторые приёмы повышения эффективности использования мультиплексоров.

Функция неравнозначности, которая описывается выражением

$$y = x_0 \cdot \bar{x}_1 + \bar{x}_0 \cdot x_1$$

и картой Карно (рис.1.31,а), может быть реализована с помощью мультиплексора следующим образом. В правом нижнем углу каждой клетки карты Карно обозначается соответствующий информационный вход мультиплексора. Логические уровни 0 и 1, записанные в клетках Карно (рис.1.31,а), присваиваются значениям входных сигналов D_i ($i = 0, \dots, 3$) мультиплексора. Например, пусть на входы D_0, D_3 поступает логический ноль, а на входы D_1, D_2 – логическая единица (рис.1.31,б). В случае, если код клетки карты Карно $x_1 x_0$ равен 00 (рис.1.31,а), значение функции y будет равно нулю. При значении кода $x_1 x_0$ равного 01, значение функции $y = 1$. Если значения $x_1 x_0$ равны 10, то значение функции $y = 1$, а при значениях $x_1 x_0$ равных 11 соответственно $y = 0$.

При синтезе логической функции, например, вида

$$y = \bar{x}_2 \cdot \bar{x}_1 \cdot \bar{x}_0 + \bar{x}_2 \cdot x_1 \cdot \bar{x}_0 + x_2 \cdot \bar{x}_1 \cdot \bar{x}_0 + x_2 \cdot x_1 \cdot x_0.$$

карта Карно может быть представлена рис.1.32,а. Практическая реализация такой карты Карно с помощью 8-входового мультиплексора приведена на рис.1.32,б.

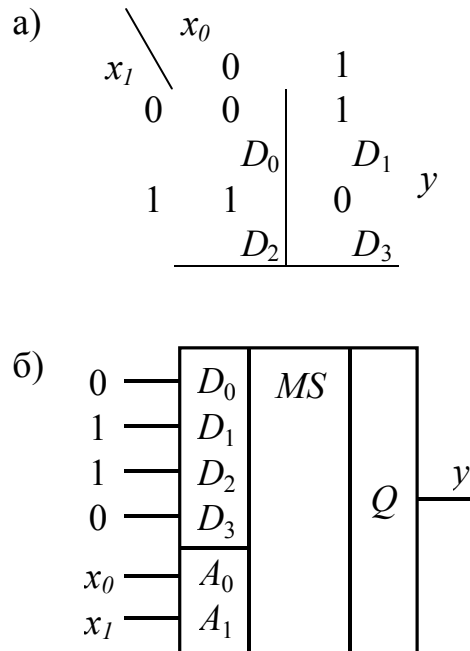
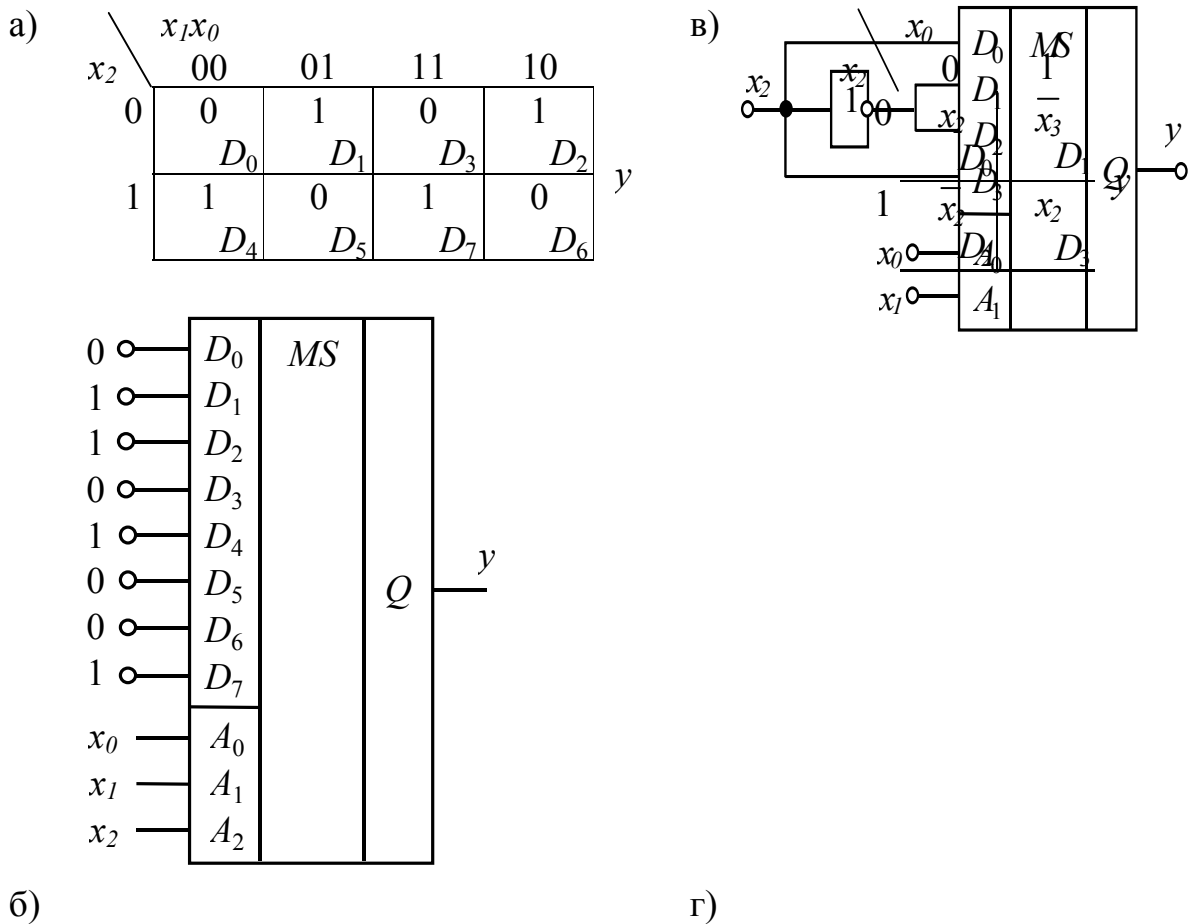


Рис.1.31. Карты Карно функции неравнозначности (а) и реализация этой функции с помощью мультиплексора (б)

Ту же функцию можно реализовать с помощью мультиплексора с меньшим числом входов. Пусть, например, входные сигналы x_1 и x_0 по-прежнему будут присутствовать на адресных входах мультиплексора, а входной сигнал x_2 используем для формирования сигналов на информационных входах. При этом каждой комбинации сигналов $x_1 x_2$ на карте Карно будет соответствовать не клетка, а столбец. Сопоставим каждому столбцу карты один из входов 4-входового мультиплексора, что эквивалентно переходу к 4-клеточной карте Карно (рис.1.32,в). В новой 4-клеточной карте Карно значения функции y выражены через входной сигнал x_2 с помощью первоначальной (8-клеточной) карты, где значения y в клетках первого столбца совпадают с x_2 ($y = x_2$), значения y в клетках второго столбца равны инверсным значениям ($y = \bar{x}_2$) и т.д. В общем случае возможны четыре варианта определения функции y по новой карте Карно (рис.1.32,в): $y = x_2$, $y = \bar{x}_2$, $y = 0$ и $y = 1$. Два последних варианта будут иметь место, когда в исходной карте (рис.1.32,а) в обеих клетках столбца $y = 1$ или $y = 0$. Практическая реализация функции y с помощью 4-входового мультиплексора

показана на рис.1.32,г.

Более сложным примером синтеза комбинационного устройства на основе мультиплексора является 4-входовый цифровой компаратор, имеющий следующее словесное описание: $y = 1$ в тех случаях, когда три и более входных сигналов равны 1. Соответствующая описанию карта Карно представлена на рис.1.33,а.



б)

г)

Рис.1.32. Карта Карно логической функции y (а), её реализация с помощью 8-входового мультиплексора (б), сокращённая форма карты Карно y (в) и реализация y на 4-входовом мультиплексоре (г).

Положим, что входные сигналы x_3 , x_1 и x_0 поступают на адресные входы, а сигналы x_2 используются для формирования сигналов на информационных входах. В общем случае заметим, целесообразно подавать на адресные (селекторные) входы мультиплексора переменные, которые входят в большее число слагаемых структурной формулы в совершенной дизъюнктивной нормальной форме.

Для осуществления перехода к новой карте Карно в рассматриваемом примере в каждом столбце исходной карты попарно объединим клет-

ки при $x_3=0$ и $x_3=1$, а значения выходной функции y для каждой пары клеток выразим через значения переменной x_2 и таким образом перейдём к 8-клеточной карте Карно (рис.1.33,б). Например, коду $x_3x_1x_0=000$ соответствует значение переменной на входе D_0 , а коду $x_3x_1x_0=111$ – на входе D_7 и т.д. Практическая реализация такого компаратора представлена на рис.1.33,в.

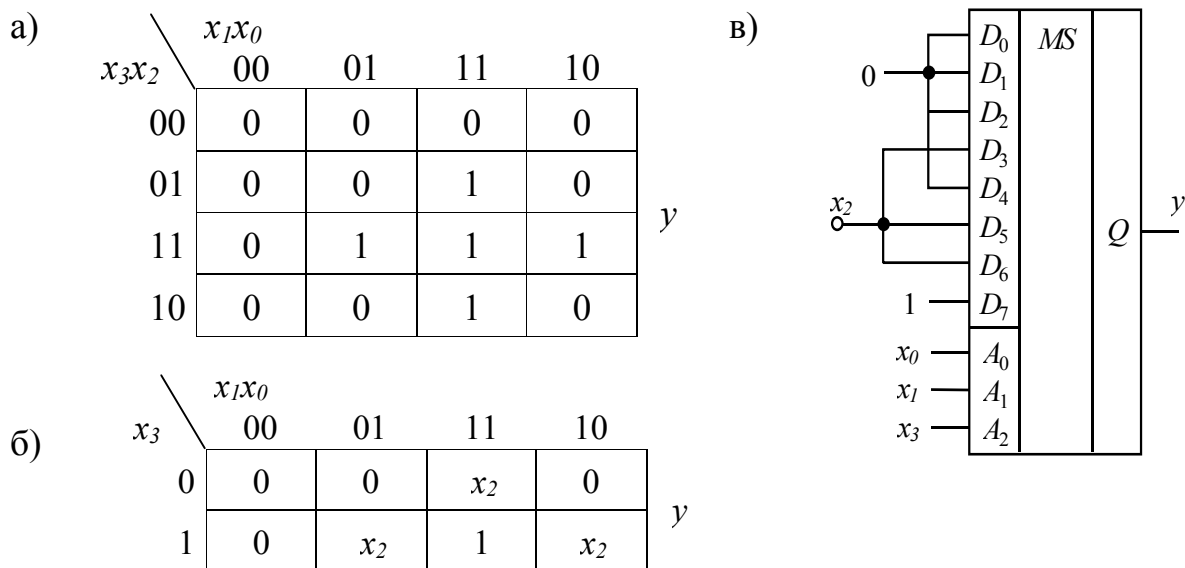


Рис.1.33. Карта Карно логической функции цифрового компаратора (а), его сокращённая карта Карно (б) и реализация на мультиплексоре (в)

При использовании логических элементов для формирования сигналов на информационных входах мультиплексора можно сократить число его входов. Пусть, требуется синтезировать логическую функцию четырёх переменных $y = f(x_1, x_2, x_3, x_4)$ с использованием 4-входового мультиплексора. Если адресными переменными выбрать x_1 и x_2 , то на информационные входы мультиплексора должны поступать переменные x_3 и x_4 , определяемые, как показано на рис.1.34,а, областями диаграммы Вейча²⁾ [6, 18, 23]. Внутри каждой очередной области диаграммы Вейча проводится процедура минимизации логической функции. Если логическая функция четырёх переменных задана картой Карно рис.1.34,б, то на информационные входы $D_0 \dots D_3$ мультиплексора должны поступать сигналы $1, 0, x_3 \cdot \bar{x}_4$ и \bar{x}_4 соответственно. Реализация заданной функции приведена на рис.1.34,в.

²⁾ Напомним, что диаграммы Вейча отличаются от карт Карно изменённой кодировкой клеток.

У мультиплексоров, выпускаемых в виде интегральных схем [18, 24, 25], число информационных входов обычно не превышает шестнадцати. Большое число входов можно обеспечить путём объединения нескольких мультиплексоров в пирамидальную (древовидную) структуру, либо путём использования последовательного соединения мультиплексоров с разрешающими входами и внешних логических элементов.

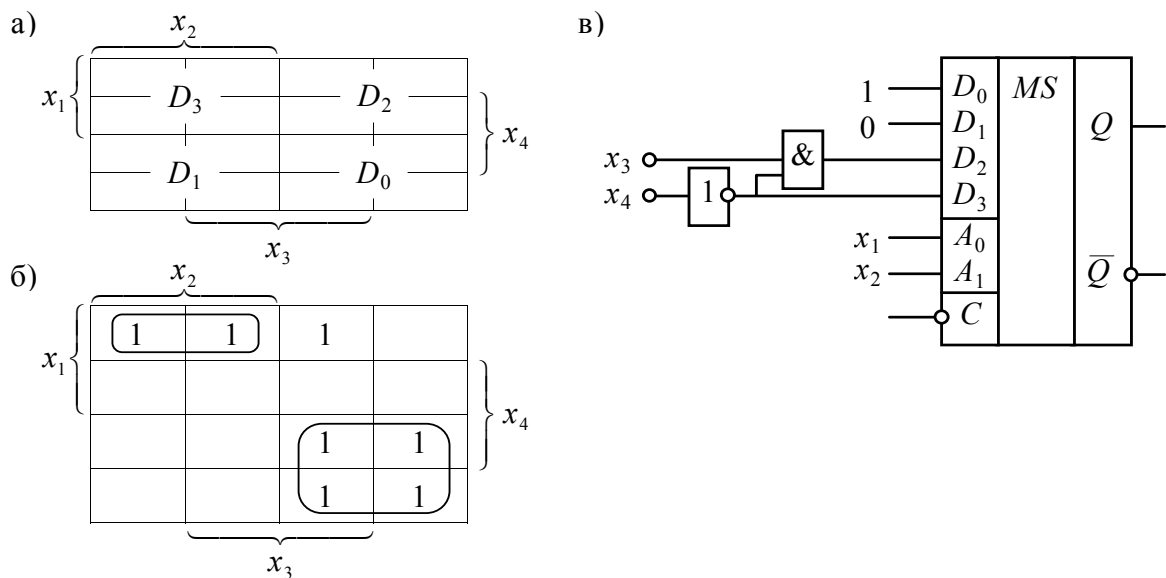


Рис.1.34. Диаграмма Вейча для функции 4-х переменных (а), диаграмма Вейча логической функции y (б) и её реализация (в)

При пирамидальном построении схемы мультиплексора каждая последующая ступень, начиная с первой, имеет меньше входов, чем предыдущая. Младшие разряды кода адреса подключены к адресным входам первой ступени. Ступеням более высокого ранга соответствуют старшие разряды кода.

На рис.1.35,а приведена схема мультиплексора с 32 информационными входами на основе четырёх 8-входовых и одного 4-входового мультиплексоров. Большое число интегральных схем является основным недостатком пирамидальной системы построения схемы мультиплексора с большим числом входов. Уменьшить количество интегральных схем можно путём построения таких мультиплексоров на основе использования в качестве адресных входов высших разрядов разрешающих входов интегральных схем (рис.1.35,б).

Мультиплексоры на основе ТТЛ, выполненные в виде отдельных интегральных схем, строятся по схеме рис.1.30,в и различаются числом информационных и адресных входов, наличием или отсутствием разрешающих

щего (стробирующего) входа. Кроме того, входные сигналы могут быть представлены в прямой и инверсной формах. Поскольку эти мультиплексоры обычно содержат дополнительные входные логические элементы, то в качестве входных цифровых сигналов могут использоваться лишь такие, логические уровни которых формируются с помощью интегральных схем на основе ТТЛ.

Мультиплексоры на логических элементах на основе КМОП логики строятся на основе дешифраторов и двунаправленных электронных ключей (рис.1.36) [6, 18]. Такие интегральные схемы содержат: преобразователь логических уровней, обеспечивающий согласование уровней (потенциалов) цифровых сигналов на адресных входах и внутренних потенциалов микросхемы; дешифратор, осуществляющий преобразование входного кода на адресных входах A_0, A_1, A_2 в сигнал на одном из его выходов и двунаправленные электронные ключи, управляемые выходными сигналами дешифратора. В отличие от мультиплексоров на основе ТТЛ, в данном случае сигнал от входа к выходу проходит без дополнительных преобразований в промежуточных интегральных схемах. Это даёт возможность осуществлять коммутацию как цифровых (импульсных), так и аналоговых сигналов. При этом неискажённая передача аналоговых сигналов обеспечивается соответствующим выбором величины питающих напряжений и схемой подключения двунаправленных электронных ключей. Такие мультиплексоры широко используются при построении многоканальных устройств уплотнения каналов связи с амплитудно-импульсной модуляцией. При этом на входы $x_0 \dots x_7$ поступают аналоговые сигналы с различных каналов связи, а на выходе у мультиплексора формируется групповой сигнал с амплитудно-импульсной модуляцией. Длительность выборочных значений аналоговых сигналов определяется формой последовательностей импульсов на выходах дешифратора (рис.1.36).

Демультимплексоры (рис.1.37,а) выполняют функции, противоположные функциям мультиплексора. Входной сигнал x поступает на один из N выходов в зависимости от значения кода $A_1 \dots A_M$. При M адресных входах демультимплексор имеет $N = 2^M$ выходов (прямых или инверсных). Таблица истинности для случая $N = 4$ представлена табл.1.21.

Таблица 1.21

a_2	a_1	y_1	y_2	y_3	y_4
0	0	D	0	0	0
0	1	0	D	0	0
1	0	0	0	D	0
1	1	0	0	0	D

Если на информационном входе присутствует логическая единица ($D = 1$), то на выбранном в соответствии с заданным адресом выходе формируется также логическая единица. При этом на остальных выходах формируется логический ноль. Выполняемой функцией демультиплексора в данном случае является дешифрование.

Одна и та же интегральная схема может выполнять функции либо дешифратора, если на входе D присутствует логическая единица, либо демультиплексора при $D \neq 1$. Такие интегральные схемы называются дешифраторами-демультиплексорами (рис.1.37,б), имеющими 2 разрешающих ($\overline{E_0}$ и $\overline{E_1}$), 4 адресных входа и 16 выходов.

Для демультиплексора на одном из разрешающих входов должен быть логический ноль, а другой вход используется как информационный. Если логический ноль присутствует на обоих разрешающих входах, интегральная схема работает как дешифратор на 4 входа и 16 выходов. При наличии логической единицы на любом из входов $\overline{E_0}$ и $\overline{E_1}$ на всех выходах устанавливается также логическая единица. Если требуемое число выходов превышает возможности одной интегральной схемы, демультиплексоры, как и мультиплексоры, могут наращиваться в систему путём построения демультиплексорного дерева (пирамидальной структуры) или с использованием разрешающих входов (рис.1.38,а,б).

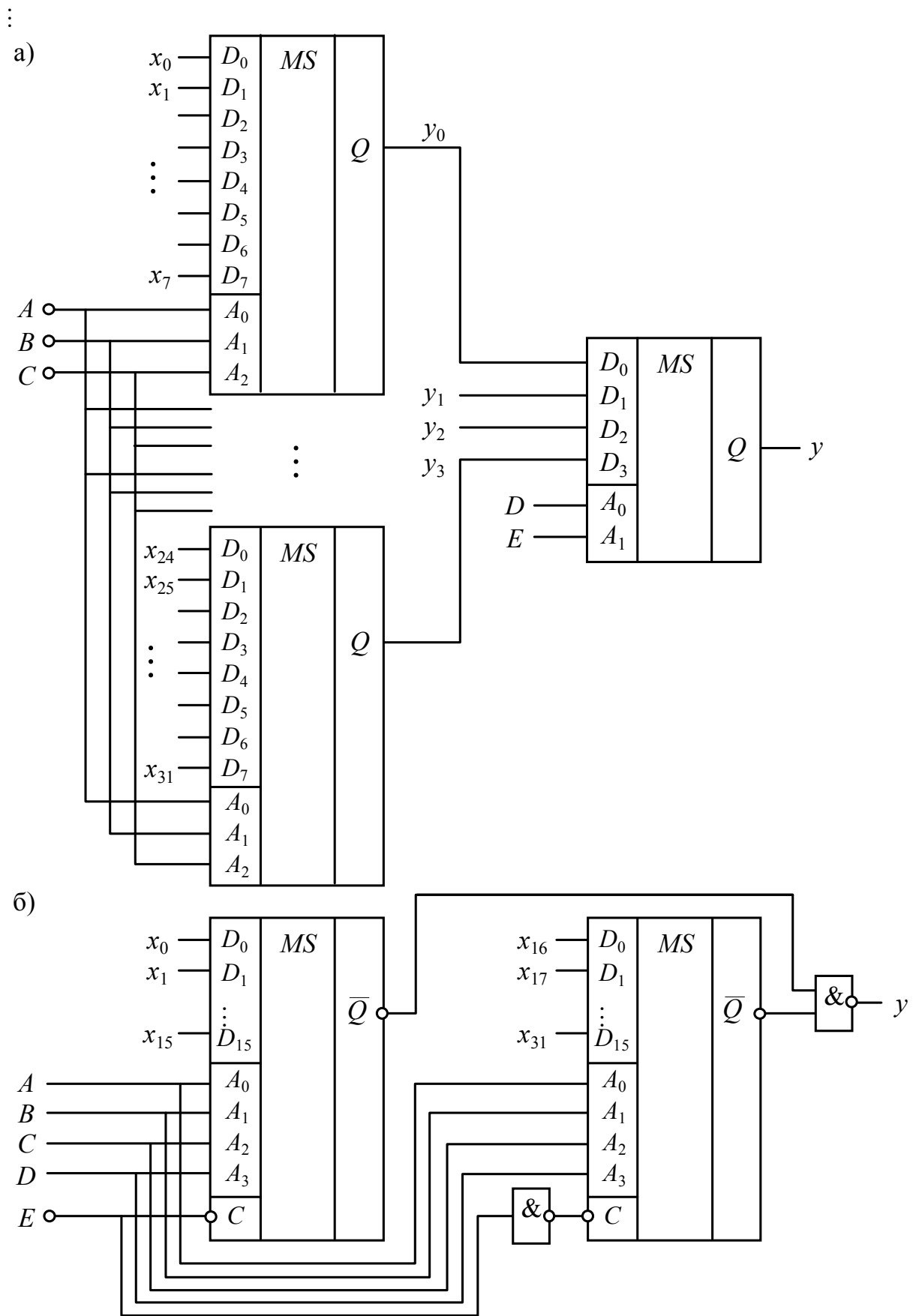


Рис.1.35. Схемы мультиплексоров с большим числом входов:
 а) пирамидальная структура; б) с использованием стробирующих входов

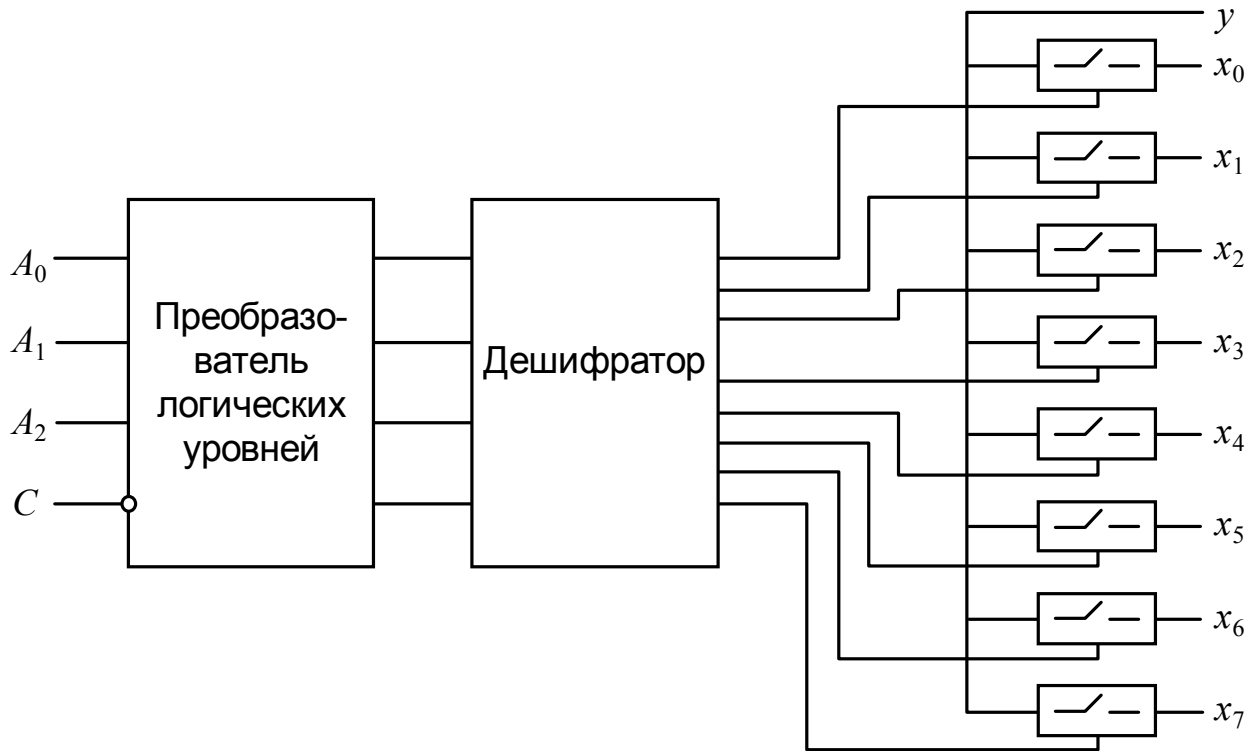


Рис.1.36. Структурная схема мультиплексора на основе КМОП логики

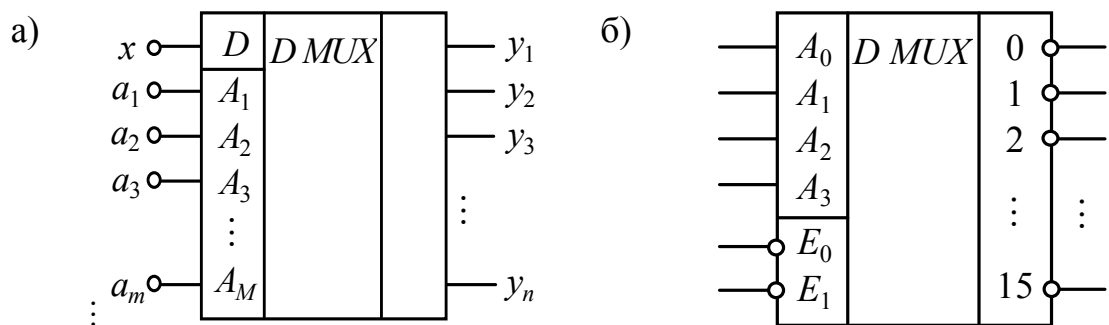


Рис.1.37. Условные обозначения демультиплексора (а) и демультиплексора-дешифратора (б)

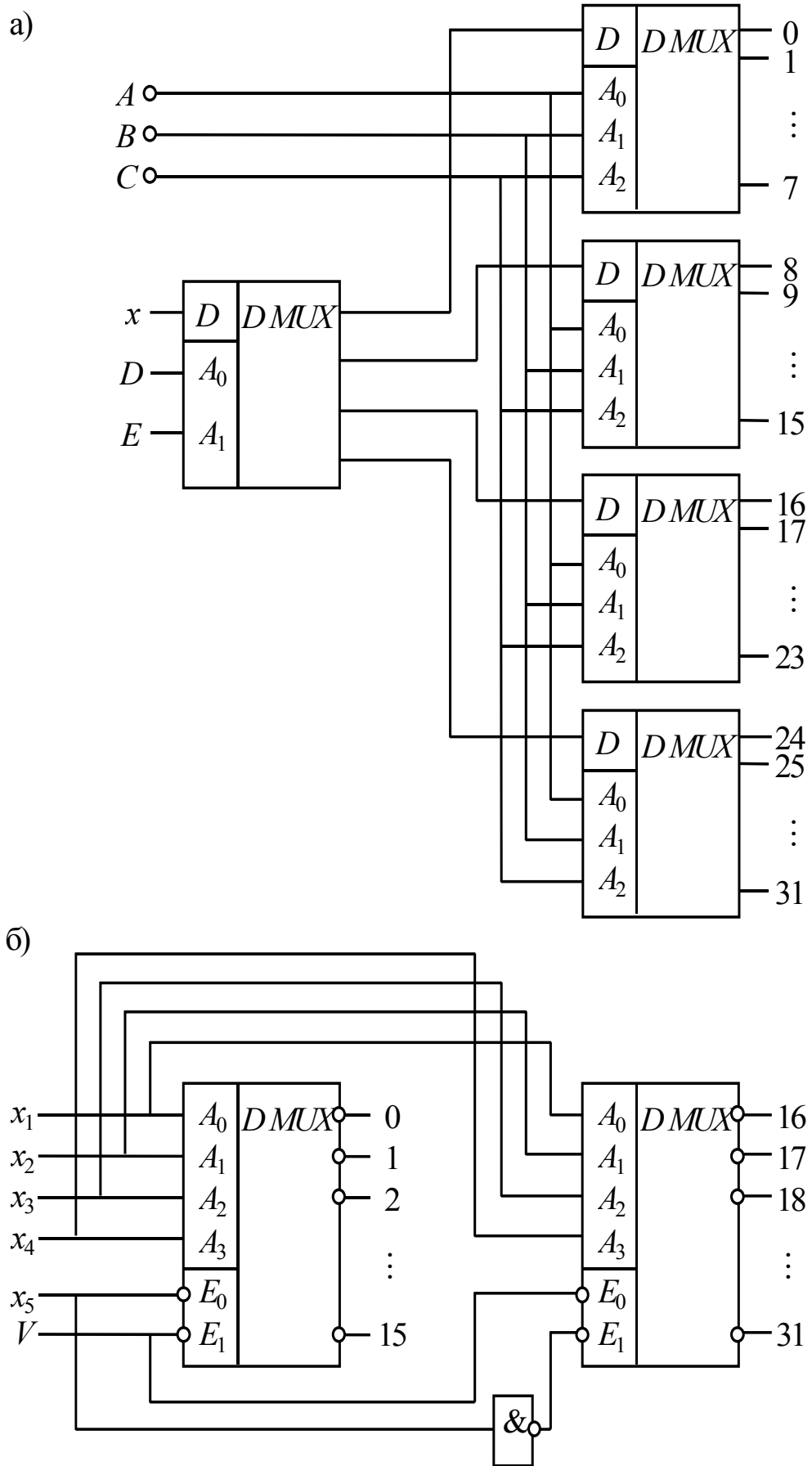


Рис.1.38. Демультимплексор с пирамидальной структурой (а) и объединение демультимплексоров путём использования разрешающих входов (б)

Таблица 1.22

x_3	x_2	x_1	y
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

С помощью демультиплексоров-дешифраторов, может осуществляться реализация логических функций, заданных таблицами истинности. Например, требуется реализовать функцию, описываемую табл.1.22. Искомая функция y может быть представлена в виде:

$$y = m_1 + m_2 + m_4 + m_5 + m_7,$$

где m_i – соответствующий минтерм или конъюнкция. Перепишем это выражение в инверсной форме:

$$\bar{y} = m_0 + m_3 + m_6.$$

Представим полученную функцию y следующим образом:

$$y = \overline{m_0 + m_3 + m_6} = \bar{m}_0 \cdot \bar{m}_3 \cdot \bar{m}_6.$$

Схемные реализации логической функции y в рассмотренных выше двух случаях представлены на рис.1.39,а,б соответственно.

Наиболее эффективным является использование демультиплексоров в комбинационных устройствах, имеющих значительное число выходных сигналов (распределители импульсных последовательностей [24], устройства формирования сигналов [10], устройства формирования нескольких логических функций одних и тех же переменных и др.). При совместном использовании мультиплексора и демультиплексора можно построить устройство, в котором по заданным адресам один из входов подключается к одному из выходов [10].

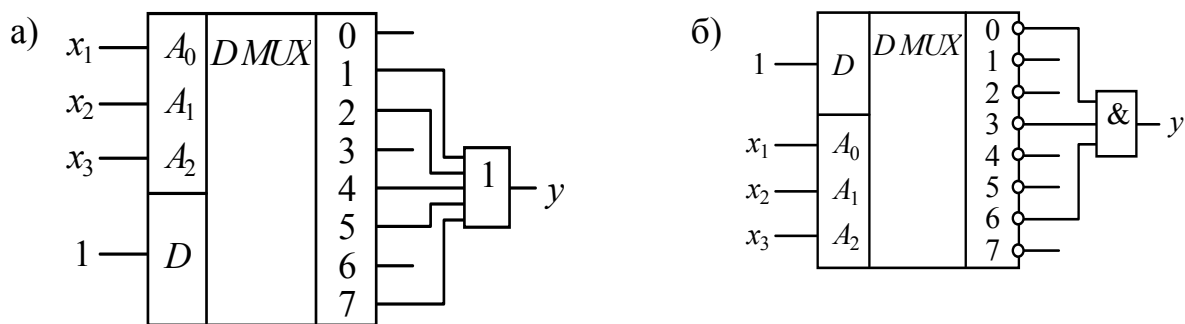


Рис.1.39. Реализация логической функции с использованием демультиплексоров

При реализации комбинационных устройств на мультиплексорах и демультиплексорах, представляющих собой интегральные схемы средней степени интеграции, сокращается требуемое число корпусов и число связей между ними (межсоединений) и, как следствие, повышается надёжность работы комбинационных устройств.

1.11. Проектирование комбинационных устройств, свободных от состязаний сигналов

При синтезе комбинационных устройств на ИС одной из основных задач является обеспечение их функциональной надёжности. Под функциональной надёжностью комбинационного устройства понимается его свойство точно реализовывать заданный алгоритм, т.е. значение сигнала на выходе комбинационного устройства должно точно соответствовать значению синтезируемой функции этого устройства.

Применение алгебры логики [1, 6, 10, 18], используемой для построения математических моделей комбинационных устройств, не позволяет рассматривать процессы перехода комбинационного устройства из одного состояния в другое, так как этот математический аппарат не учитывает временные (инерционные) параметры логических элементов (задержку сигналов в физических элементах реальных устройств) и структуру реальных сигналов (конечную длительность фронта и спада импульсов).

Наличие задержки сигналов в логических элементах, а также разброс величины этой задержки и конечные длительности перепадов уровней входных сигналов приводят к тому, что во время переходных процессов устройство функционирует не в соответствии с законами алгебры логики и в нём наблюдается **состязание сигналов**.

Состязания сигналов могут быть опасными и неопасными. Опасными являются состязания, приводящие к возникновению на выходе изменения уровня сигнала, не предусмотренного алгоритмом работы устройства и изменяющего работу последующих устройств.

Если работа последующих устройств даже при наличии неалгоритмического перехода в комбинационном устройстве не нарушается, то состязания становятся **н е о п а с н ы м и**. Опасные состязания различаются по месту их возникновения в условиях перехода комбинационного устройства из одного состояния (с определённым уровнем сигнала на выходе) в другое.

В зависимости от места возникновения различают состязания сигналов на входах устройства и состязания сигналов в логических элементах устройства.

Состязания сигналов на входах имеют место, если по условию работы комбинационного устройства одновременно изменяются несколько входных переменных. Так, например, если сигналы на входах логических элементов И и ИЛИ имеют вид, показанный на рис.1.40,а, то во время переходных процессов для случая $x \cdot \bar{x}$ могут быть моменты времени, когда сигналы на обоих входах эквивалентны единичному логическому уровню (рис.1.40,б), а для случая $x + \bar{x}$ – нулевому логическому уровню на обоих входах (рис.1.40,в). В результате сигналы на выходах не соответствуют логическим функциям элементов.

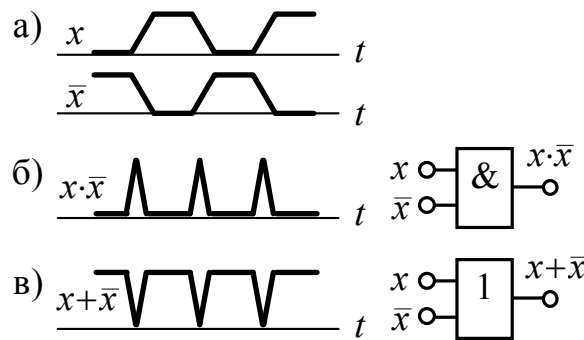


Рис. 1.40. Форма сигналов на входах (а) и выходах (б, в) логических элементов И-НЕ, ИЛИ-НЕ

Поясним на примере (рис.1.41) возникновение состязаний сигналов в логических элементах комбинационных устройств. Инерционность реальных логических элементов приводит к тому, что выходные сигналы появляются после изменения входных сигналов через некоторое время, определяемое длительностью переходных процессов [1]. Логический элемент в большинстве случаев представляется в виде двух частей: одна выполняет логическую функцию и является безынерционной, другая включена вслед

за первой и является элементом задержки на время, равное средней задержке сигналов в логическом элементе, т.е. на $t_{3\text{cp}}$ (рис.1.41,а).

Используя такое представление логических элементов, рассмотрим пример последовательного соединения нескольких логических элементов (рис.1.41,б), реализующих логическую функцию вида $y = \overline{\overline{x_1 + x_2 + x_3}}$.

Предположим, что при $t < t_0$ входные сигналы принимают значения: $x_1 = 0$, $x_2 = x_3 = 1$, а при $t = t_0$ – эти значения соответственно равны: $x_1 = 1$, $x_2 = x_3 = 0$.

Таблица 1.23

Время	x_1	x_2	x_3	A	B	C	E	F	y
$t < t_0$	0	1	1	1	1	0	0	0	0
t_0	1	0	0	0	1	0	0	1	0
$t_0 + t_{3\text{cp}}$	1	0	0	0	0	1	0	1	1
$t_0 + 2t_{3\text{cp}}$	1	0	0	0	0	1	1	0	1
$t_0 + 3t_{3\text{cp}}$	1	0	0	0	0	1	1	0	0
$t_0 + 4t_{3\text{cp}}$	1	0	0	0	0	1	1	0	0

Без учёта временных задержек для обеих комбинаций входных сигналов значения выходного сигнала $y = 0$. Проследим с помощью временных диаграмм работу устройства, используя для логического элемента модель, представленную на рис.1.41,а. Результаты анализа работы такого комбинационного устройства приведены в табл.1.23, а временные диаграммы входных и выходных сигналов – на рис.1.41,в.

Как видно, на выходе комбинационного устройства в момент времени $t_0 + t_{3\text{cp}}$ возникает помеха длительностью $2t_{3\text{cp}}$ (функция y на рис.1.41,в), обусловленная переходными процессами в реальных физических элементах.

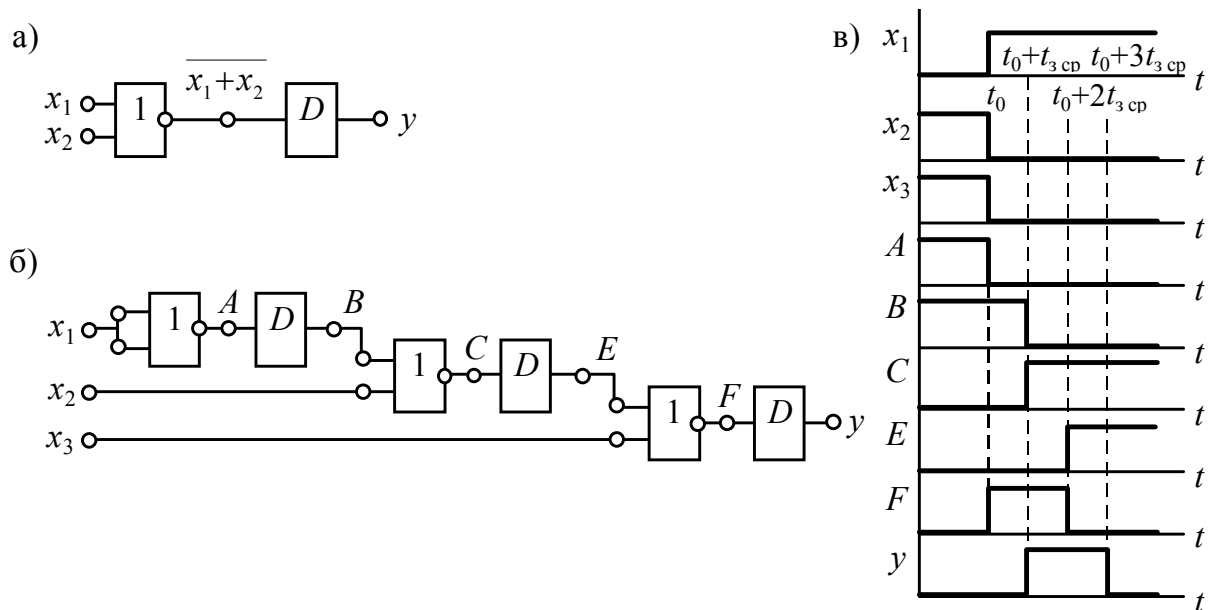


Рис.1.41. Модель логического элемента (а), модель устройства, реализующего заданную логическую функцию y (б), временные диаграммы в различных точках этого устройства (в)

В зависимости от условий перехода комбинационного устройства из одного состояния в другое различают **статические** и **динамические** состязания. Если для двух последовательных во времени соседних состояний сигналов на входах состояние выхода должно оставаться неизменным, то состязания в устройстве называют **статическими**. Если два последовательных во времени соседних состояния входов должны иметь переход состояния на выходе, то состязания в устройстве называются **динамическими**.

Статические состязания подразделяют на **единичные** (1-состязания) и **нулевые** (0-состязания). Единичные состязания появляются в случае, когда при изменении входных сигналов на выходе комбинационного устройства должно сохраниться значение логической 1, а в результате состязаний появляется логический 0. При нулевых состязаниях во время переходного процесса появляется логическая 1 при значениях сигналов на входах, сохраняющих логический 0 на выходе.

Приведём примеры образования неалгоритмических переходов при статических состязаниях.

Для иллюстрации явления единичных состязаний рассмотрим устройство (рис.1.42,а), которое описывается логической функцией вида:

$$y = \overline{\overline{x_1 \cdot x_2 \cdot x_3 \cdot \bar{x}_1}}.$$

Временные диаграммы, поясняющие возникновение состязаний сигналов, приведены на рис.1.42,б, причём у логического элемента НЕ задержка сигнала равна $t_{3\text{cp}1}$, а у логического элемента И–НЕ – $t_{3\text{cp}2}$. Из анализа временных диаграмм (рис.1.42,б) видно, что в комбинационном устройстве возникли два неалгоритмических перехода (единичные состязания).

Процесс образования нулевых состязаний покажем на примере комбинационного устройства (рис.1.43,а), реализующего логическую функцию вида:

$$y = \overline{\overline{x_1 \cdot x_2}} + \overline{\overline{x_1 \cdot x_3}}.$$

Как нетрудно убедиться (рис.1.43,б), в таком устройстве возможно возникновение нулевых состязаний.

Рассмотрим пример появления неалгоритмических переходов при динамических состязаниях. Пусть логическая функция, описывающая устройство, имеет вид:

$$y = \overline{\overline{x_1 \cdot x_2 \cdot \overline{x_1} \cdot x_3 \cdot \overline{x_1} \cdot x_4}}.$$

Структурная схема устройства, реализующего такую логическую функцию, изображена на рис.1.44,а, а временные диаграммы, поясняющие работу устройства, представлены на рис.1.44,б. В идеальном случае, если бы задержки сигналов в логических элементах отсутствовали, алгоритмический переход должен был бы произойти в момент времени $t = t_1$. Однако из-за наличия задержек этот переход сдвигается на время $t = t_{10}$. Кроме того, в комбинационном устройстве возникают два неалгоритмических перехода в моменты времени t_5 и t_8 (функция y на рис.1.44,б), причём первый неалгоритмический переход будет совпадать с алгоритмическим.

Анализ работы комбинационного устройства с целью выявления опасных состязаний по временным диаграммам при большом количестве логических элементов оказывается достаточно трудоёмким. Поэтому, как правило, используются формальные методы анализа.

Рассмотрим подробнее один из них, а именно аналитический метод, или метод Мак-Класки. Логическую функцию y , описывающую работу комбинационного устройства, преобразуют либо в дизъюнктивную нормальную форму [1] при анализе на статические нулевые состязания, либо в конъюнктивную нормальную форму при анализе на статические единичные состязания. При этом не допускаются такие логические преоб-

разования, при которых происходит потеря входных переменных x_i и \bar{x}_i . С этой целью x_i и \bar{x}_i рассматривают как независимые входные переменные, поскольку в динамическом режиме (т.е. во время переходных процессов) в некоторых точках цепи логических элементов x_i и \bar{x}_i могут оказаться не взаимно инверсными, а одновременно равными либо логическому 0, либо логической 1.

Статические нулевые состязания имеют место, если:

- логическая функция y в дизъюнктивной нормальной форме содержит хотя бы одно из слагаемых, в которое одна из переменных входит в прямом x_i и инверсном \bar{x}_i виде (например, $\dots + x_i \cdot \bar{x}_i + \dots$);
- выполняется следующее условие:

$$y \left| \begin{array}{l} x_1=x_2=\dots=1, \\ \text{кроме } x_i=1 \end{array} \right. + y \left| \begin{array}{l} x_1=x_2=\dots=1, \\ \text{кроме } x_i=0 \end{array} \right. \neq 1.$$

Первое слагаемое в этом выражении получается путём подстановки в логическую функцию y , записанную в дизъюнктивной нормальной форме, логической единицы вместо x_i , и логического нуля вместо \bar{x}_i . Второе слагаемое получается путём подстановки вместо x_i значения логического нуля, а вместо \bar{x}_i – логической единицы. Все другие входные переменные x_1, x_2, \dots при этом остаются записанными в общем виде. После соответствующих упрощений эти переменные заменяются на логические 1 для того, чтобы выяснить, выполняется ли записанное таким образом второе условие. Если и это условие имеет место, в комбинационном устройстве присутствуют нулевые состязания.

Статические единичные состязания имеют место, если:

- логическая функция y в конъюнктивной нормальной форме содержит хотя бы один сомножитель, в который одна из переменных входит в прямом x_i и инверсном \bar{x}_i виде (например, $\dots (x_i + \bar{x}_i) \dots$);
- выполняется следующее условие:

$$y \left| \begin{array}{l} x_1=x_2=\dots=0, \\ \text{кроме } x_i=1 \end{array} \right. + y \left| \begin{array}{l} x_1=x_2=\dots=0, \\ \text{кроме } x_i=0 \end{array} \right. \neq 0.$$

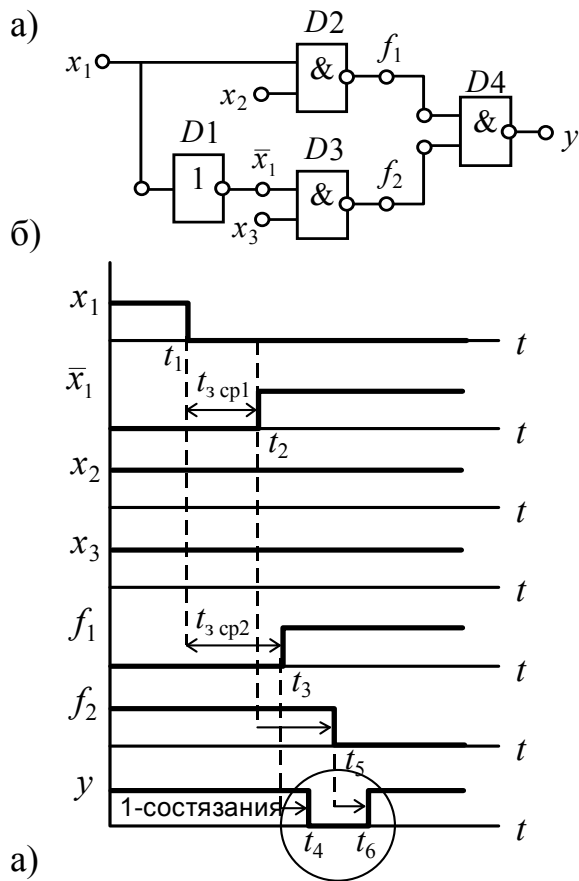


Рис.1.42. Единичные состоязания сигналов:
а) структурная схема устройства; б) временные диаграммы

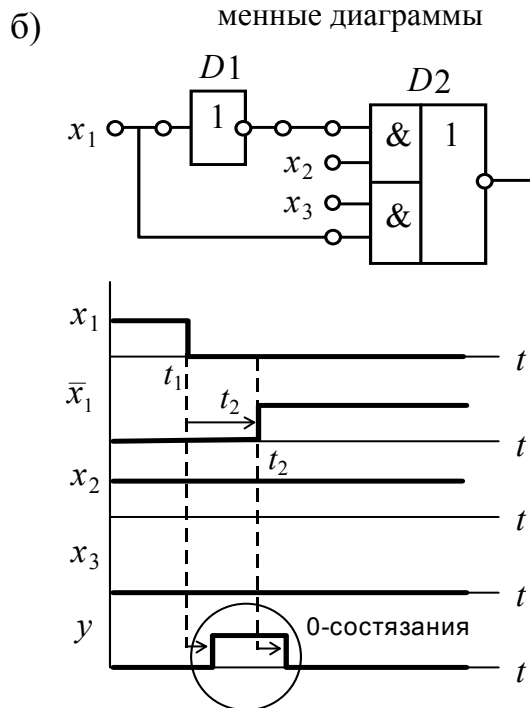


Рис.1.43. Нулевые состоязания сигналов: а) структурная схема устройства; б) временные диаграммы сигналов

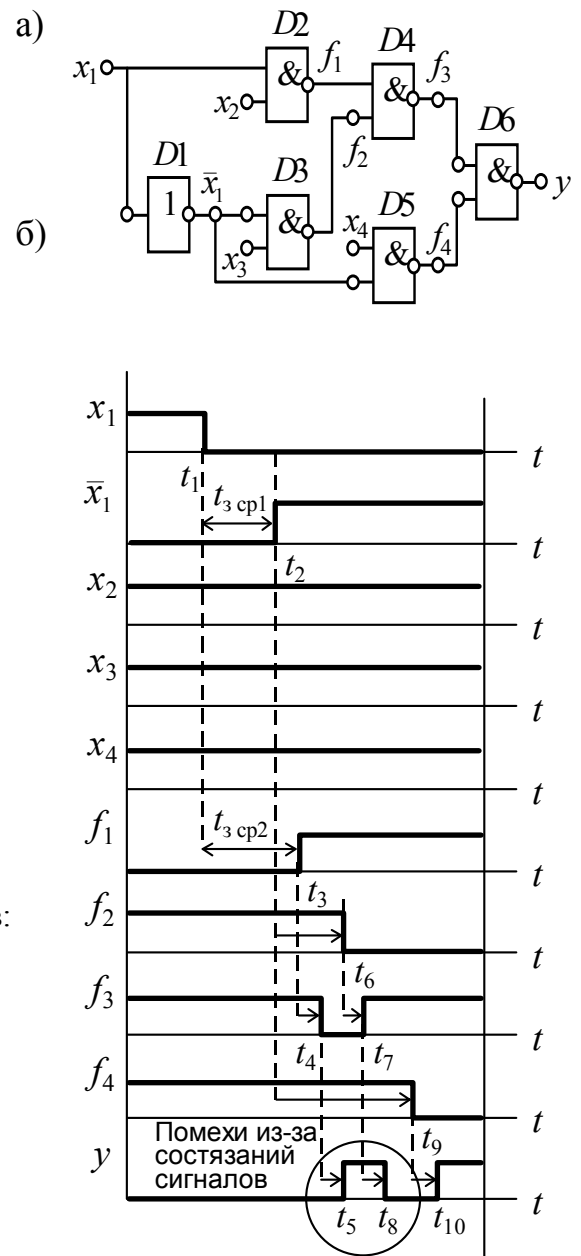


Рис.1.44. Динамические состоязания сигналов: а) структурная схема устройства; б) временные диаграммы

Рассмотрим практическое применение метода Мак-Класки. Пусть

$$y = \overline{\bar{x}_1 \cdot x_2 + x_1 \cdot x_3}.$$

Из структурной формулы видно, что состязания сигналов могут возникнуть в логическом элементе ИЛИ–НЕ. Предположим, что в таком комбинационном устройстве возможны статические нулевые состязания. Для проверки этого предположения представим выражение для y в дизъюнктивной нормальной форме:

$$y = \overline{\bar{x}_1 \cdot x_2 \cdot x_1 \cdot x_3} = (x_1 + \bar{x}_2) \cdot (\bar{x}_1 + \bar{x}_3) = x_1 \cdot \bar{x}_1 + \bar{x}_2 \cdot \bar{x}_1 + x_1 \cdot \bar{x}_3 + \bar{x}_2 \cdot x_3.$$

Видно, что одно из условий наличия 0-состязаний выполняется, так как имеется слагаемое вида $x_i \cdot \bar{x}_i$:

$$\begin{aligned} (1 \cdot 0 + \bar{x}_2 \cdot 0 + 1 \cdot \bar{x}_3 + \bar{x}_2 \cdot \bar{x}_3) + (0 \cdot 1 + \bar{x}_2 \cdot 1 + 0 \cdot x_3 + \bar{x}_2 \cdot \bar{x}_3) = \\ = \bar{x}_3 + \bar{x}_2 \cdot \bar{x}_3 + \bar{x}_2 + \bar{x}_2 \cdot \bar{x}_3 = \bar{x}_2 + \bar{x}_3 + \bar{x}_2 \cdot \bar{x}_3 = \bar{x}_2 + \bar{x}_3. \end{aligned}$$

После подстановки вместо x_2 и x_3 логической 1 получим: $0 + 0 \neq 1$. Следовательно, в таком комбинационном устройстве возникают статические нулевые состязания.

Проверим работу комбинационного устройства на наличие единичных состязаний. Для этого преобразуем выражение для y в конъюнктивной нормальной форме:

$$y = \overline{\bar{x}_1 \cdot x_2 \cdot x_1 \cdot x_3} = (x_1 + \bar{x}_2) \cdot (\bar{x}_1 + \bar{x}_3).$$

Таким образом, первое условие не выполняется. Следовательно, единичные состязания сигналов будут отсутствовать.

Другим методом анализа комбинационных устройств на наличие опасных состязаний сигналов является метод с использованием карт Карно. Применение этого метода рассмотрим на примере. Для этого проведём анализ комбинационного устройства, структурная схема которого представлена на рис. 1.43,а и логическая функция y определяется выражением:

$$y = x_1 \cdot \bar{x}_1 + \bar{x}_2 \cdot \bar{x}_1 + x_1 \cdot \bar{x}_3 + \bar{x}_2 \cdot \bar{x}_3.$$

Таблица 1.24

x_1	x_2	x_3	y	\bar{y}
0	0	0	1	0
0	0	1	1	0

0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	1	0
1	1	1	0	1

Карта Карно для y строится на основе её таблицы истинности (табл.1.24). В данном случае удобно взять вместо функции y инверсную функцию

$$\bar{y} = \bar{x}_1 \cdot x_2 + x_1 \cdot x_3,$$

а после составления для неё карты Карно изменить значения логического 0 на логическую 1 и, наоборот, в клетках карты Карно (рис.1.45).

		$x_2 x_3$			
		00	01	11	10
x_1	0	1	1	0	0
	1	1	0	0	1

Рис.1.45. Карта Карно, показывающая возможность возникновения нулевых состязаний

На картах Карно соседние единицы и соседние нули охватываются общим контуром (контуром склеивания). Если при изменении одной из входных переменных изменения функции y таковы, что на карте происходит переход из одного контура склеивания в другой, с ним не связанный, то в комбинационном устройстве имеют место состязания. Так, на рис.1.45 видно, что при переходе функции y от значения логической 1 к 1 в случае изменения одной из входных переменных x_1 или x_2 (соседние переходы) выход за пределы одного контура склеивания не будет происходить. С другой стороны, при переходе функции y от значения логического 0 к 0 осуществляется переход из одного контура в другой. Это означает наличие в комбинационном устройстве нулевых состязаний. Следовательно, условием отсутствия статических состязаний будет связанность всех контуров склеивания на карте Карно, что достигается введением в структурную формулу дополнительных слагаемых или сомножителей в инверсную функцию \bar{y} , обеспечивающих связан-

ность контуров склеивания.

При проектировании комбинационных устройств, свободных от опасных статических состязаний сигналов, как правило, используются следующие правила построения устройств.

Применение структурных методов. Наиболее эффективным из них является метод проектирования с использованием карт Карно, в соответствии с которым для каждой пары состояний сигналов на входах комбинационных устройств необходимо иметь на картах Карно для функций y и \bar{y} контуры склеивания, соответствующие одному из слагаемых этих функций.

Коррекция опасных состязаний. Представим комбинационное устройство в виде обобщённой структурной схемы (рис.1.46), где A_1 и A_2 – цепи, по которым происходит передача входных сигналов, создающих на выходе элемента $D7$ неалгоритмические переходы, распространяющиеся далее по цепи A_3 . Можно показать, что возникающий из-за этих переходов ложный сигнал при выполнении определённых условий будет постепенно уменьшаться по длительности в цепи A_3 . На этом и основывается метод коррекции опасных состязаний.

Разброс значений задержек сигналов в цепях A_1 и A_2 определяет длительность сложного сигнала, который образуется на выходе $D7$ (см. рис.1.46). Задача коррекции опасных состязаний сигналов в цепях A_1 и A_2 во всех известных случаях состоит в определении допустимой разницы задержек $\tau_{\text{доп}}$ в этих цепях, при которой появившийся ложный сигнал окажется подавленным цепью A_3 в силу того, что последовательная цепь логических элементов обладает формирующими свойствами [1,3]. Такое правило применяется в основном при разработке структур интегральных схем средней и большей степени интеграции (СИС и БИС).

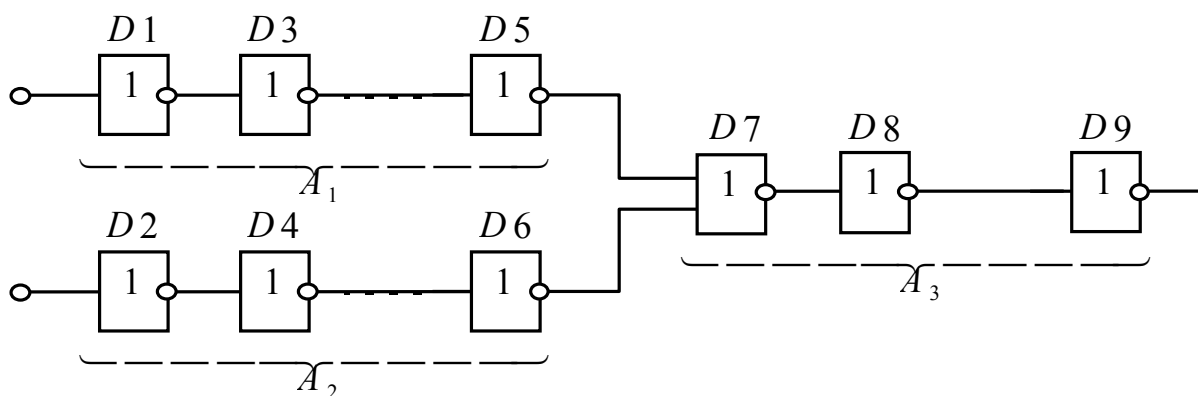


Рис.1.46. Обобщённая структурная схема комбинационного устройства при коррекции опасных состязаний

Введение синхронизации. Работой цифровых устройств в данном случае управляют тактовые (синхронизирующие) последовательности так, что запись и считывание информации осуществляются только в течение длительности тактовых импульсов. Переключение комбинационных устройств должно происходить за интервал времени Δt между тактовыми импульсами. При этом длительность Δt выбирается такой, чтобы в течение этого интервала времени все переходные процессы, связанные с переключением комбинационных устройств, закончились и на выходах установились бы стационарные значения сигналов. Заметим, что введение синхронизации существенно уменьшает быстродействие устройства.

Селекция импульсов по длительности. Если длительность ложного сигнала (помехи), обусловленного состязаниями, меньше, чем время между соседними изменениями уровней сигналов, то можно использовать для его подавления селекторы импульсов по длительности, не пропускающие кратковременные помехи.

Применение определённого порядка смены сигналов и состояний комбинационного устройства. Поскольку опасные состязания проявляются в виде помех только при смене определённых комбинаций входных сигналов, то иногда можно предусмотреть такой порядок чередования этих комбинаций, при котором опасные состязания будут отсутствовать.

Следует отметить, что при использовании в радиотехнических устройствах быстродействующих логических элементов (например, ЭСЛ) и в случаях передачи сигналов между устройствами на большие расстояния необходимо при анализе состязаний учитывать задержки распространения сигналов по цепям связи между элементами и от устройства к устройству.

В заключение необходимо отметить, что проблема состязаний в

цифровых устройствах является очень серьезной, так как большинство трудно обнаруживаемых и удивительно разнообразно проявляющихся ошибок в работе схем связано с наличием состязаний, возможности появления которых разработчик не заметил.

Глава 2. Последовательные устройства

2.1. Способ формального описания последовательных устройств

В последовательных устройствах значения выходных сигналов в каждый момент (интервал) времени зависят не только от значений входных сигналов в тот же момент времени, как это имело место в комбинационных устройствах, но и от их предыдущих значений. Последовательные устройства (ПУ) реализуют функциональную связь не между отдельными комбинациями (наборами) входных и выходных сигналов, а между их последовательностями. Поэтому ПУ должны обладать памятью, благодаря которой сохраняется информация о предыдущих воздействиях входных сигналов.

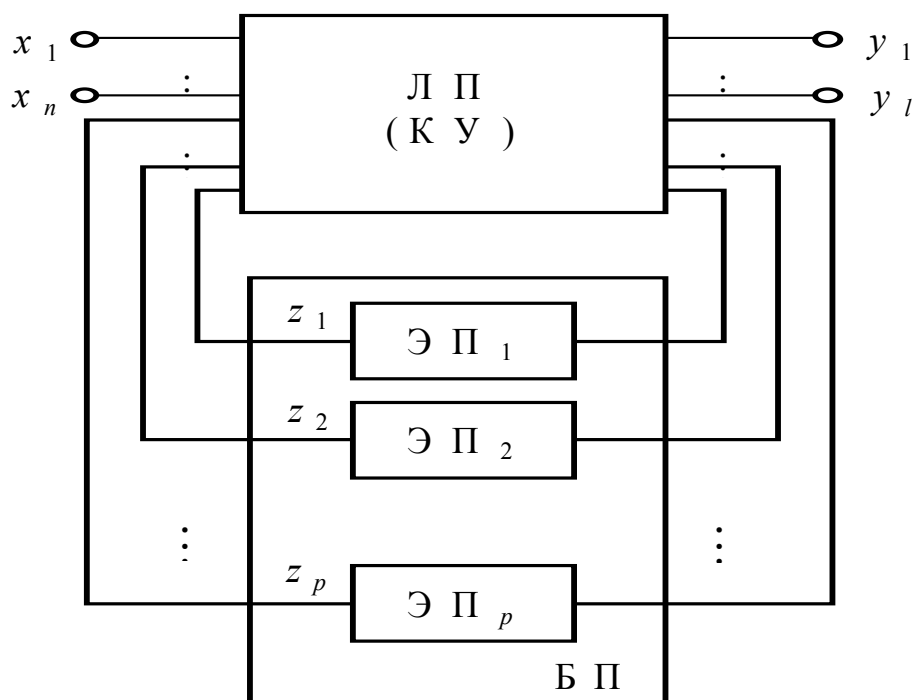


Рис.2.1. Структурная схема последовательного устройства

Поведение любого цифрового устройства определяется изменением во времени некоторых физических величин токов и напряжений на его входах, выходах и внутри самого устройства. В дальнейшем будем их называть входными, выходными и внутренними переменными.

В отличие от комбинационных устройств (КУ) выходные сигналы последовательных устройств определяются не только комбинацией входных сигналов, но зависят также от внутреннего состояния устройства (т.е. набора внутренних переменных) при приходе входных сигналов.

Внутреннее состояние ПУ отражает предысторию его работы, т. е. память устройства. Наличие памяти является характерным свойством любого последовательностного устройства, поэтому структурная схема ПУ (рис.2.1) может быть представлена состоящей из двух блоков: логического преобразователя (ЛП), выполненного на основе комбинационных устройств, и блока памяти (БП), т.е. совокупности элементов памяти ЭП₁, ЭП₂, ... , ЭП_p.

Состояния входа и выхода ЛП являются соответственно состояниями входа и выхода ПУ, а внутренние переменные – состояниями элементов памяти, образующих блок памяти.

Последовательностные устройства можно описать множеством состояний входа $X = \{x_1, x_2, \dots, x_n\}$, конечным множеством состояний выхода $Y = \{y_1, y_2, \dots, y_l\}$, конечным множеством внутренних состояний $Z = \{z_1, z_2, \dots, z_p\}$, где x_i, y_i, z_i – входные, выходные и внутренние переменные, и двумя функциями: функцией переходов, определяющей порядок смены внутренних состояний, и функцией выходов, задающей состояние выходов в зависимости от состояния входов и внутреннего состояния.

При работе цифровых устройств могут быть выделены временные интервалы, на которых все переменные (входные, выходные и внутренние) сохраняют постоянные значения. Эти интервалы времени Δt называют тактами работы цифрового устройства. В дальнейшем будем рассматривать работу последовательностных устройств лишь в моменты времени t_i ($i = 1, 2, \dots$) как равноотстоящие друг от друга на величину Δt , так и произвольно расположенные, считая, что именно в эти моменты происходит изменение той или иной входной переменной.

Обозначим x_i^n величину входной переменной x_i в момент времени t_n . Запись той же входной переменной в форме x_i^{n-1} и x_i^{n+1} будет означать, что эта переменная рассматривается в моменты времени t_{n-1} и t_{n+1} .

Различают два типа последовательностных устройств: асинхронные и синхронные (тактируемые). В асинхронных ПУ все входы равноправны, и изменение сигнала на любом входе может вызвать изменение сигналов на каких-либо выходах. В синхронных последовательностных устройствах имеется, по крайней мере, один выделенный вход синхронизации (С), и входные сигналы могут воздействовать на ПУ лишь при наличии определенного сигнала на этом входе.

Если имеется словесное описание работы последовательностного устрой-

ства то, для того, чтобы получить алгоритм его функционирования необходимо задать функции переходов и выходов. Для функции переходов это будет означать задание определенного перехода из одного внутреннего состояния Z_i в другое Z_j (при этом не исключается случай $i=j$) при состоянии входа X_k (либо при изменении последовательностей на входе ПУ). Задание функции выходов состоит в сопоставлении каждой паре X_i и Z_i состояния выхода $Y_i \in Y$. Обе функции можно представить в виде таблиц (соответственно переходов и выходов) или с помощью графа.

Таблицы переходов и состояний

Рассмотрим правила построения таблиц переходов. Строки таблицы переходов (табл.2.1) соответствуют текущим внутренним состояниям последовательностного устройства, столбцы – состояниям входов устройства. Элементы таблицы переходов соответствуют внутренним состояниям, в которые должно перейти ПУ под воздействием входных сигналов.

Таблица 2.2.

	X_1	X_2
Z_1	1	2
Z_2	3	–
Z_3	–	4
Z_4	1	–

Таблица 2.1.

	X_1	X_2	X_3	X_4
Z_1	Z_1	Z_2	Z_4	Z_1
Z_2	Z_2	Z_3	Z_1	Z_2
Z_3	Z_3	Z_4	Z_2	Z_3
Z_4	Z_4	Z_1	Z_3	Z_4

Таблица переходов обычно содержит 2^n столбцов, где n – число входных переменных X_i . Число строк равно числу состояний схемы. В табл.2.1 приведен пример таблицы переходов полного синхронного последовательностного устройства, функции переходов которого определены для всех наборов X и Z . Как видно, устройство имеет четыре состояния входа (X_1, X_2, X_3, X_4) и четыре внутренних состояния (Z_1, Z_2, Z_3, Z_4). В каждой клетке таблицы переходов указан номер внутреннего состояния, в которое устройство должно перейти в следующий момент времени. Например, при состояниях входа X_1 и X_4 последовательностное устройство не меняет своего внутреннего состояния, при X_2 – устройство переходит в следующее внутреннее состояние, при X_3 – в предыдущее.

В дальнейшем в целях упрощения вместо z_i в клетках таблиц будем указывать лишь номер состояния. Если в последовательностном устройстве какое-либо состояние не определено, т.е. оно является неиспользуемым или запрещенным, то в соответствующей клетке таблицы ставят прочерк

(табл.2.2).

В таблице выходов (табл.2.3) задается соответствие между состоянием выхода и внутренним состоянием. Если обе таблицы совместить, получим так называемую таблицу состояний, задающую одновременно как функцию переходов, так и функцию выходов. В каждой клетке такой таблицы (табл.2.4) записаны значения Z_i и Y_j .

Таблица 2.3.

	X_1	X_2	X_3	X_4
	Y	Y	Y	Y
	Y	Y	Y	Y
	Y	Y	Y	Y
	Y	Y	Y	Y

Таблица 2.4.

	X_1	X_2	X_3	X_4
	Z	Z	Z	Z
	Z	Z	Z	Z
	Z	Z	Z	Z
	Z	Z	Z	Z

Таблица 2.5.

	X_1	X_2	X_3	X_4
1	(1) ↑	2	3	—
2	1 ←	(2) →	(2)	3
3	4	—	(3)	(3)
4	(4)	2	(4)	—

Для асинхронных последовательностных устройств, если последующее внутреннее состояние при изменении X остается таким же, как и предыдущее, т. е. $f(X_i, Z_j) = Z_j$, оно является устойчивым, и в таблице переходов номер этого состояния заключают в скобки. В противном случае внутреннее состояние является неустойчивым, и его обозначают цифрой без скобок. Например, если последовательностное устройство находится в устойчивом состоянии (2) (табл.2.5) и состояние его входа изменяется с X_2 на X_3 , то устройство переходит из (2) в (2), т.е. внутреннее состояние его при этом не изменится. В случае, когда происходит изменение X_2 на X_1 , устройство первоначально примет неустойчивое состояние 1, а затем устойчивое (1).

Таким образом, изменение внутреннего состояния последовательностного устройства всегда связано с переходом его через неустойчивое состояние. Переход из одного внутреннего состояния в другое через устой-

чивое невозможно.

Минимизация числа состояний.

При построении таблицы переходов в рассмотрение может оказаться включенным больше состояний, чем необходимо для нормального функционирования последовательного устройства.

Таблица 2.6.

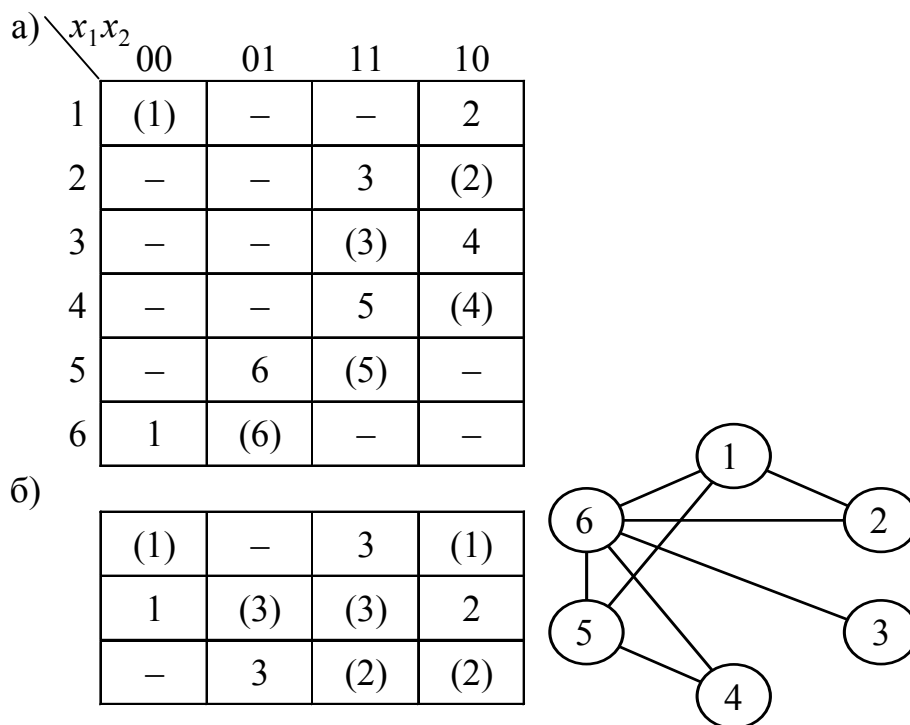


Рис.2.2. Диаграмма состояний последовательного устройства

При сокращении числа состояний избыточными состояниями считают такие, которым в таблице переходов соответствуют строки с непротиворечивым размещением цифр. Такие строки в таблице переходов имеют одинаковые цифры в каком-либо столбце, или в одной строке этого же столбца стоит цифра, а в другой имеется прочерк. Эти строки называют совместимыми. Группа строк является совместимой, если все строки, входящие в эту группу, попарно совместимы.

Объединение совместимых строк приводит к уменьшению числа внутренних переменных, необходимых для описания всех переходов. При объединении строк каждой группе совместимых строк присваивают новую цифру и производят новое обозначение состояний последовательного устройства. Заметим, что при выполнении операции объединения строк значения выходных функций во внимание не принимаются. Полученную

после объединения строк новую таблицу называют сокращенной таблицей переходов. Для оптимизации операции объединения строк строят соответствующую диаграмму. Построение такой диаграммы рассмотрим на примере таблицы переходов (табл.2.6,а).

Разместим номера строк по некоторой окружности (рис.2.2). Возможность объединения отдельных строк попарно условно отметим линиями между соответствующими номерами. Каждая строка должна быть сопоставлена со всеми другими для выявления возможности их объединения. Анализ диаграммы позволяет выявить оптимальную схему объединения строк, при которой минимизируется общее число строк в таблице. Например, объединяя строки 1 и 2, 3 и 6, а также 4 и 5, получим сокращенную таблицу переходов (табл.2.6,б). Поскольку p внутренних переменных могут образовать 2^p наборов их значений и в последовательностном устройстве с таким числом внутренних переменных можно реализовать 2^p состояний, возникает задача кодирования значений внутренних переменных.

Кодирование значений внутренних переменных

Кодирование заключается в присвоении каждому состоянию ПУ, т.е. каждой строке сокращенной таблицы переходов или каждому набору значений переменных z_i в данный момент времени, значения двоичного кода. Так, при реализации четырех состояний последовательностного устройства потребуются две внутренние переменные (обозначим их z_1 и z_2). Задачу кодирования внутренних переменных удобно решать с помощью специальной карты кодирования (табл.2.7,а), где код состояния совпадает с двоичным кодом соответствующей клетки. Располагая, например, состояние 1 последовательно в каждой клетке и всякий раз записывая последовательность состояний в карту кодирования, придерживаясь их размещения в направлении по часовой стрелке, получим четыре варианта кодирования.

При записи последовательности состояний в карту в направлении против часовой стрелки будем иметь еще четыре варианта кодирования. Дополняя таблицу переходов кодом состояний в соответствии с избранным вариантом кодирования, запишем кодированную таблицу переходов (табл.2.7,б).

Таблица 2.7.

$z_2 \backslash z_1$	0	1
0	2	3
1	1	4

$z \backslash x$	0	1	$z_1 z_2$
1	(1)	2	10
2	3	(2)	00
3	(3)	4	01
4	1	(4)	11

Диаграмма состояний

Последовательностные устройства могут быть представлены диаграммой состояний или графом, состоящим из узлов, соединенных ветвями. Обозначив состояние ПУ узлами, а переходы последовательностного устройства, получаемые под воздействиями входных сигналов X_i , ветвями, будем иметь диаграмму состояний. Так, диаграмма состояний, соответствующая таблице состояний (табл.2.8), приведена на рис.2.3. На ветвях графа указаны значения входных и выходных сигналов.

Диаграмма состояний последовательностного устройства может быть использована для определения вида выходных переменных при произвольных входных переменных для любого начального состояния устройства. Например, для ПУ, представленного графом на рис.2.3, если имеет место первое начальное состояние, а входная последовательность X равна 00101, то образуется последовательность состояний вида 22434 при соответствующей последовательности на выходе 11010. Та же входная последовательность, но воздействующая на ПУ при третьем начальном состоянии, приводит к выходной последовательности вида 01010.

Сравнивая табличный и графический способы задания состояний последовательностного устройства, можно отметить следующее:

- Табличный способ задания является наиболее полным, но труднообозримым для большого числа состояний. В то же время данный способ позволяет осуществить сравнительно простой переход к формульной записи функций Y и Z .

Таблица 2.8

$z \backslash x$	0	1
1	2,1	3,0
2	2,1	4,0
3	1,0	4,0
4	3,1	1,0

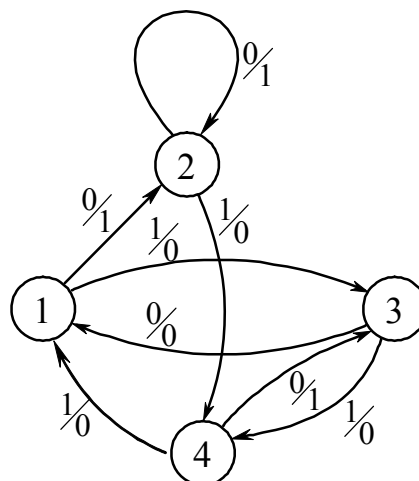


Рис.2.3. Граф последовательного устройства

- Графический способ более нагляден. Особенно его преимущества ощущаются при решении задач обнаружения возможных циклических режимов работы ПУ.

В процессе проектирования последовательных устройств может быть выполнен переход от графа переходов к таблице, и наоборот.

Пример формального описания последовательного устройства

Рассмотрим синтез последовательного двоичного сумматора, который должен производить сложение двух чисел, поступающих на его входы, начиная с младших разрядов. Числа вводятся в сумматор последовательно разряд за разрядом, синхронно с тактовым сигналом. Сумматор на каждом такте должен вычислять значение разряда суммы при поступлении на его входы соответствующих разрядов слагаемых с учетом единицы переноса из предыдущего младшего разряда. При этом в сумматоре должна запоминаться на один такт единица переноса в старший разряд.

В соответствии с этим словесным описанием последовательное устройство должно иметь два входа, на которые подаются сигналы (числа) x_1 и x_2 , один выход Y и один элемент памяти для запоминания сигнала переноса в старший разряд. Следовательно, последовательное устройство может иметь четыре состояния входа $X = (x_1, x_2)$: $X_1 = (0, 0)$, $X_2 = (0, 1)$, $X_3 = (1, 0)$, $X_4 = (1, 1)$ или $X = \{00, 01, 10, 11\}$, два состояния выхода $y_1 = 0, y_2 = 1$, т. е. $Y = \{0, 1\}$, два внутренних состояния $Z = Q$, т. е. $Z_1 = 0, Z_2 = 1$, или $Z = \{0, 1\}$.

Функции переходов и выходов данного ПУ имеют вид:

$$Q^n = f(x_1^n, x_2^n, Q^{n-1}),$$

$$Y^n = \varphi(x_1^n, x_2^n, Q^{n-1}).$$

В данном случае, поскольку сигнал переноса соответствует внутренней переменной, являющейся в то же время выходной для устройства, вместо Z используется обозначение Q .

Последовательностное устройство должно производить сложение чисел x_1 и x_2 с учетом переноса Q , поэтому $Q^n = 1$ в тех случаях, когда два или три сигнала из x_1^n, x_2^n, Q^{n-1} равны 1, а $y^n = 1$ только при нечетном числе сигналов x_1^n, x_2^n, Q^{n-1} , равных 1 (в остальных случаях $Q^n = 0$ и $y^n = 0$). Составим таблицу состояний ПУ (табл.2.9). Каждая строка таблицы соответствует одному из возможных состояний входа X на n -ом такте работы устройства. В клетках, соответствующих строкам Z_i и столбцам X_j , указывается состояние последовательностного устройства $Q^n = Z^n$, в которое он переходит, и состояние выхода y^n .

Таблица 2.9.

X	x_1^n	x_2^n	x_3^n	x_4^n
z_1^n	$z_1; y_1$	$z_1; y_2$	$z_1; y_2$	$z_2; y_1$
z_2^n	$z_1; y_2$	$z_2; y_1$	$z_2; y_1$	$z_2; y_2$

Таблица 2.10.

$x_1 x_2$	00	01	10	11	Q
1	1,0	1,1	1,1	2,0	0
2	1,1	2,0	2,0	2,1	1

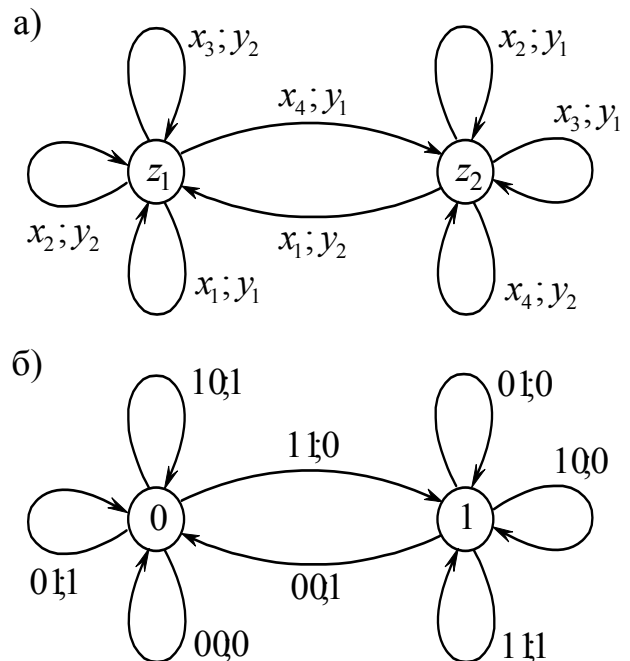


Рис.2.4. Варианты графов переходов сумматора

Таблица 2.11

x_1^n	x_2^n	Q^n	Q^{n+1}	y^n
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1

1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Кодирование состояний входа и выхода с помощью наборов значений входных и выходных сигналов часто определяется самой постановкой задачи проектирования, поэтому в таблице состояний часто удобно указывать наборы значений этих сигналов (табл.2.9). Дополняя табл.2.9 столбцом Q , в котором записывается принятое кодирование состояний ПУ, перейдем к табл.2.10. Графы переходов такого устройства, построенные по табл.2.9 и табл.2.10, приведены на рис.2.4,а и б соответственно. С помощью этих таблиц можно составить и таблицу истинности (табл.2.11).

2.2. Триггеры

Основным свойством триггеров, представляющих собой простейшие последовательностные устройства, является сохранение одного из двух устойчивых состояний после прекращения действия входных сигналов. Триггер имеет два выхода: прямой Q и инверсный \bar{Q} . Уровнями напряжения на этих выходах определяется состояние, в котором находится триггер: если напряжение на выходе соответствует уровню логического 0 ($Q = 0$), то принимается, что триггер находится в состоянии логического 0; при $Q = 1$ триггер находится в состоянии логической единицы.

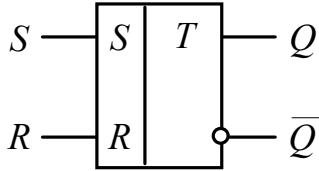
Рассмотрим свойства RS , T , D , DV и JK -триггеров, используемых в качестве запоминающих ячеек, электронных ключей, прерывателей и т. д., а также представляющих собой элементы, на которых строятся более сложные последовательностные устройства.

Асинхронные RS -триггеры

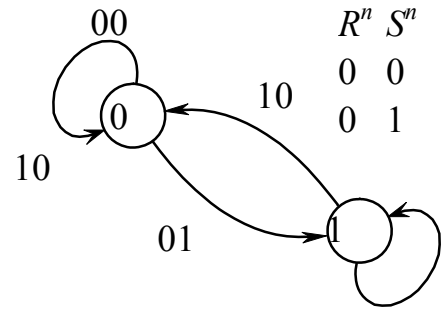
Словесное описание асинхронного RS -триггера с переключающими сигналами 1 (рис.2.5,а) имеет вид: если сигналы на входах R и S равны 0, т. е. $R^n = S^n = 0$, то триггер не изменяет своего состояния ($Q^n = Q^{n-1}$); если $R^n = 0$, $S^n = 1$, триггер переключается в состояние 1 ($Q^n = 1$) (название входа S происходит от английского слова «set» – установка), если $R^n = 1$, $S^n = 0$, триггер переключается в состояние 0 (название входа R происходит от слова «reset» – сброс); одновременное появление 1 на обоих входах, т. е. $R^n = S^n = 1$, должно быть исключено. Правила работы можно записать в виде таблицы переключений (табл.2.12), либо путем по-

строения по этой таблице или по словесному описанию графа переключений (рис.2.5,б). Следует отметить, что граф не содержит запрещенных комбинаций входных сигналов.

а)



б)

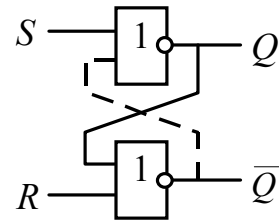


в) $S^n + Q^{n-1}S^n + \bar{Q}^{n-1}\bar{S}^n + \bar{Q}^{n-1}S^n + Q^{n-1}$

	00	01	11	10
R^n 0	0			
\bar{R}^n 1	0	0	Φ_2	Φ_1

Q^n

г)



Имея формальное описание триггера, можно синтезировать его схему так же, как схему комбинационного устройства. Для этого, ориентируясь на логические элементы ИЛИ-НЕ, заполняется таблица истинности для Q^n (табл.2.13), составляется карта Карно (рис.2.5,в), с помощью которой получается структурная формула для Q^n :

$$Q^n = \overline{\overline{R^n}} \cdot (S^n + Q^{n-1}) = \overline{\overline{R^n}} \cdot (S^n + Q^{n-1}) = \overline{\overline{R^n}} + \overline{\overline{S^n + Q^{n-1}}}$$

Таблица 2.13.

n	R	S	Q	Q
	n	n	$n-1$	n
0	0	0	0	0
0	0	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	1	1	0	Φ_1
1	1	1	1	Φ_2

Таблица 2.12.

R^n	S^n	Q^n
0	0	Q^{n-1}
0	1	1
1	0	0
1	1	—

Схема асинхронного RS -триггера на логических элементах ИЛИ-НЕ, работа которого осуществляется в соответствии с этой формулой, представлена на рис.2.5,г. При этом учтено, что Q^{n-1} и Q^n являются сигналами в одной точке схемы в разные моменты времени (штриховая линия на рис.2.5,г). Заметим, что в соответствии с правилами работы логических элементов ИЛИ-НЕ, при наличии на обоих входах RS -триггера переключающих сигналов $R^n = S^n = 1$ на прямом и инверсном выходах RS -триггера (рис.2.5,г) должны быть уровни логического нуля. Однако, из таблицы истинности (табл.2.13) следует, что в данном состоянии ($R^n = S^n = 1$) на выходах RS -триггера с равной вероятностью могут появиться как уровни логического нуля, так и единицы (состояния Φ_1 и Φ_2 - неопределенность). Иначе говоря, имеется противоречие между описанием работы отдельных логических элементов ИЛИ-НЕ и описанием работы RS -триггера. Действительно, как следует из принципов работы триггера, из-за наличия различных реальных задержек сигналов в логических элементах ИЛИ-НЕ, при $R^n = S^n = 1$ на его прямом Q и инверсном \bar{Q} выходах могут быть лишь сочетание сигналов: 0 и 1, либо 1 и 0. Поскольку рассмотренная неопределенность неприемлема для работы RS -триггера, то комбинация $R^n = S^n = 1$ считается запрещенной. Если входная комбинация сигналов сменится нейтральной ($R^n = S^n = 0$), то на выходах RS -триггера с равной вероятностью могут появиться как логические нули, так и единицы.

Переводя схему RS -триггера на логические элементы И-НЕ, получим схему, изображенную на рис.2.6,а. Исключив лишние элементы (перечеркнутые пунктирными линиями на рис.2.6,а), придем к RS -триггеру (рис.2.6,б) с той же таблицей переключения. Устранив из схемы на рис.2.6,б входные инверторы, получим RS -триггер с переключающими сигналами 0 (рис.2.6,в). Таблица переходов этого RS -триггера показана в табл.2.14.

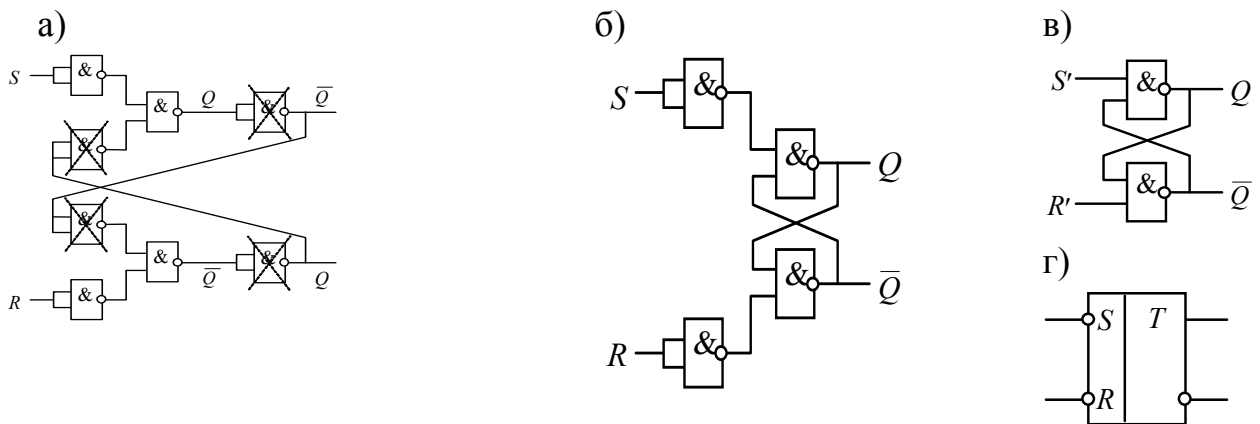


Рис.2.6. Варианты реализации RS -триггера на элементах И-НЕ (а-в)

После приведения табл.2.14 к нормальной форме, получим табл.2.15, соответствующую работе асинхронного RS -триггера с переключающими сигналами 0 (рис.2.6,г).

Таблица 2.14.

R'	S'	Q^n
1	1	Q^{n-1}
1	0	1
0	1	0
0	0	—

Таблица 2.15.

R'	S'	Q^n
0	0	—
0	1	0
1	0	1
1	1	Q^{n-1}

Проведенная процедура перевода RS -триггера на другие логические элементы дает возможность сформулировать следующее правило перехода. Для того, чтобы перейти от последовательностного устройства, выполненного на элементах ИЛИ-НЕ (И-НЕ), к функционально-подобному устройству, выполненному на элементах И-НЕ (ИЛИ-НЕ), необходимо каждый элемент ИЛИ-НЕ (И-НЕ) заменить элементом И-НЕ (ИЛИ-НЕ) с таким же количеством входов, оставив все соединения внутри схемы неизменными, и поменять местами прямые и инверсные выходы. Таблицу пе-

реключения нового последовательного устройства получают из таблицы переключений исходного устройства путем инвертирования значений входных сигналов.

Сформулированное правило отражает двойственность последовательного устройства на логических элементах И-НЕ и ИЛИ-НЕ.

Синхронные *RS*-триггеры

Синхронный *RS*-триггер изменяет свое состояние, определяемое сигналами на управляющих входах, лишь при наличии переключающего сигнала $C = 1$ на входе синхронизации C . При $C = 0$ состояние *CRS*-триггера не меняется. Функциональное обозначение *CRS*-триггера и граф его работы приведены на рис.2.7.

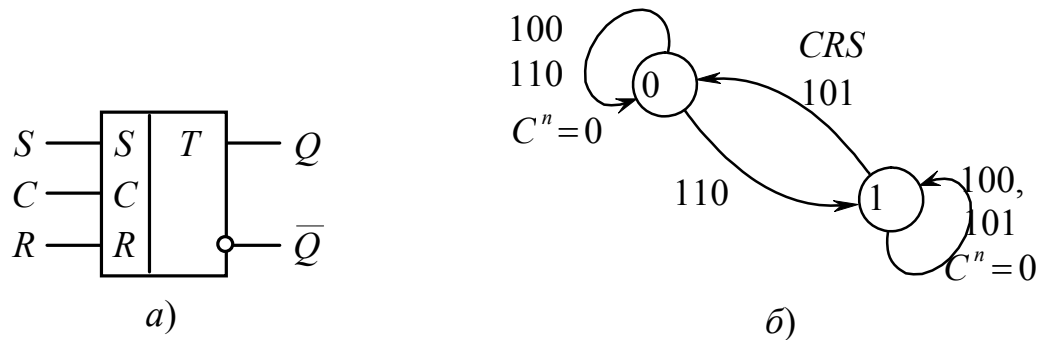


Рис. 2.7. Условное обозначение *CRS*-триггера (а) и граф его переходов (б)

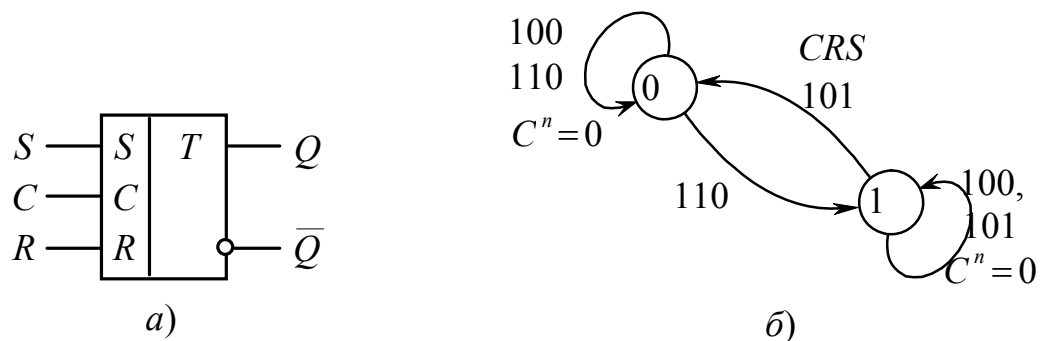


Рис. 2.8. Синхронный *CRS*-триггер: структурная схема (а) и её реализация (б)

По принципу работы этот *CRS*-триггер соответствует асинхронному *RS*- триггеру с переключающими сигналами 1 при $C^n = 1$. Учитывая это обстоятельство, можно представить структуру *CRS*-триггера в виде последовательно соединённых устройства управления и элемента памяти – *RS*-триггера (рис.2.8,а). Таблица истинности для входов элемента памяти R' и S' представлена в табл.2.16.

Таблица 2.16.

C^n	R^n	S^n	R'^n	S'^n
0	0	0	0	1
0	0	1	1	1
0	1	0	0	1
0	1	1	Φ_1	Φ_2
1	0	0	1	1
1	0	1	1	0
1	1	0	0	1
1	1	1	Φ_3	Φ_4

Таблица 2.17.

		$R^n S^n$				
		00	01	11	10	
\bar{C}^n	0			Φ_1		$\overline{S'^n} = S'^n \cdot C^n$
	1		1	Φ_3		
C^n	0			Φ_2		$\overline{R'^n} = R'^n \cdot C^n$
	1			Φ_4	1	

При ее составлении учитывалось словесное описание работы CRS -триггера и требуемые сигналы на входах RS -триггера. Поскольку таблица истинности содержит значительное число единиц для функций R' и S' , то карты Карно можно составить для обратных функций (табл.2.17). Полученная схема CRS -триггера приведена на рис.2.8,б.

T-триггеры

Счетный или T -триггер (рис.2.9,а) имеет лишь один информационный вход (название " T -триггер" происходит от слова «toggle» – переключатель). Этот триггер переключается в противоположное состояние с приходом каждого переключающего сигнала на вход T . Таблица переключений такого устройства (табл.2.18) содержит две строки.

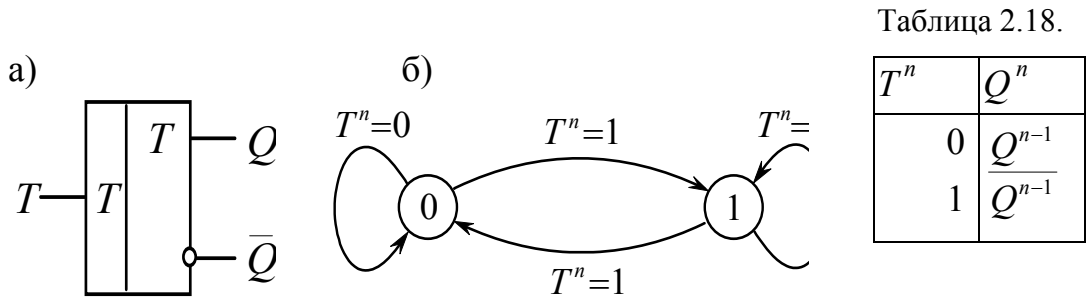


Рис. 2.9. T -триггер: условное обозначение (а), граф переходов (б)

Если изобразить граф переходов T -триггера, полагая, что в структуру его входит один элемент памяти (рис.2.9,б), то становится очевидным, что последовательностное устройство будет неустойчивым, так как при $T^n = 1$ будет происходить переключение триггера из одного состояния в другое и обратно (режим генерации).

Для обеспечения устойчивости работы такого последовательностного устройства необходимо ввести в структуру дополнительные элементы памяти. Реализация этого осуществляется: либо путем построения двухступенчатых триггеров, либо организацией динамического управления.

В двухступенчатом T -триггере (рис.2.10) один элемент памяти (Q) формирует выходные сигналы Q и \overline{Q} , а второй (Q_1) обеспечивает устойчивость работы. Эти триггеры часто называют триггерами типа MS (название такого триггера происходит от слов «master» – хозяин и «slave» – раб, или в иной интерпретации: «ведущий – ведомый», «главный – вспомогательный»).

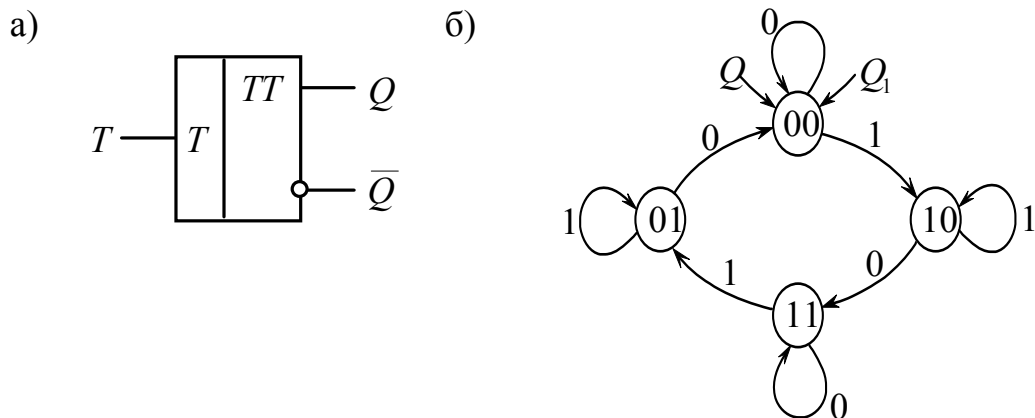


Рис. 2.10. Условное обозначение T -триггера по двухступенчатой схеме (а) и его граф (б)

Проведем синтез такой схемы T -триггера, считая, что в его структуру

входят два RS -триггера: основной Q и вспомогательный Q_1 . Задача синтеза сводится к определению функций управления элементов памяти (RS -триггеров). С помощью графа переключений составляется таблица истинности (табл.2.19) для входных сигналов RS -триггеров. При этом, где возможно, ставится знак Φ , чтобы полнее использовать возможности процедуры минимизации. Затем по картам Карно (табл.2.20), полученным для инверсных функций, определяются структурные формулы, которые позволяют построить схему двухступенчатого T -триггера (рис.2.11). Она включает два RS -триггера и дополнительные элементы, соответствующие полученным структурным формулам.

Таблица 2.19

T^n	Q^{n-1}	Q_1^{n-1}	R^n	S^n	R_1^n	S_1^n
0	0	0	Φ_1	1	Φ_2	1
0	0	1	Φ_3	1	0	1
0	1	0	1	Φ_4	1	0
0	1	1	1	Φ_5	1	Φ_6
1	0	0	1	0	Φ_7	1
1	0	1	Φ_8	1	1	Φ_9
1	1	0	1	Φ_{10}	Φ_{11}	1
1	1	1	0	1	1	Φ_{12}

Таблица 2.20.

Q^n	00	01	11	10
0			Φ_5	Φ_4
1	1			Φ_{10}

$$\overline{S^n} = T^n \cdot \overline{Q_1^{n-1}}$$

0	Φ_1	Φ_3		
1		Φ_8	1	

$$\overline{R^n} = T^n \cdot \overline{Q_1^{n-1}}$$

0			Φ_6	1
1		Φ_3	Φ_{12}	

$$\overline{S_1^n} = \overline{T^n} \cdot \overline{Q^{n-1}}$$

0	Φ_2	1		
1	Φ_1			Φ_{11}

$$\overline{R_1^n} = \overline{T^n} \cdot \overline{Q^{n-1}}$$

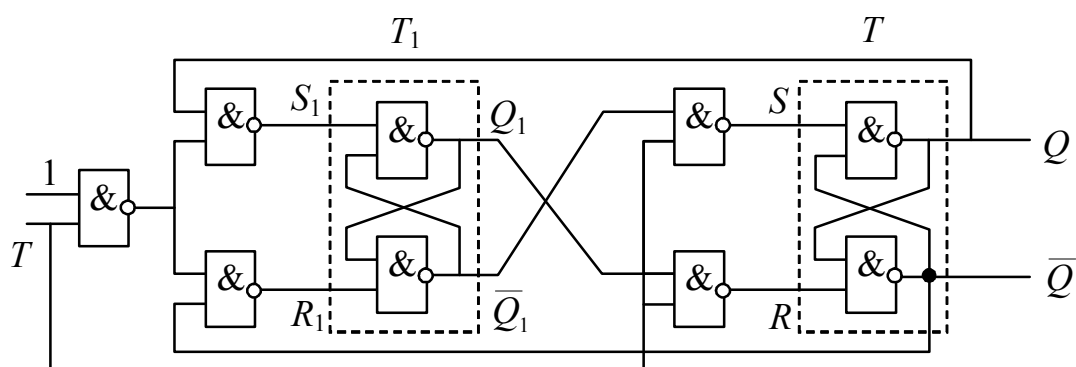


Рис.2.11. Структурная схема двухступенчатого T -триггера

В триггерах с динамическим управлением устойчивая работа обеспечивается тем, что элемент памяти, формирующий выходной сигнал, воспринимает входной сигнал только в течение малого интервала времени после изменения значения сигнала на входе T . Достигается это путем запоминания сигналов, существующих к моменту изменения значения сигнала на входе T , с помощью элементов памяти, включенных на входах RS -триггера, формирующего входной сигнал. Таким образом получается схема «трех триггеров», в которой к каждому входу основного RS -триггера, формирующего выходной сигнал Q , подключены дополнительные RS -триггеры, вырабатывающие сигналы R и S для основного триггера. Граф такого T -триггера и два варианта схемы – симметричная и несимметричная – приведены на рис.2.12,а-в соответственно. Полученные схемы содержат меньшее число логических элементов, чем схемы двухступенчатых триггеров.

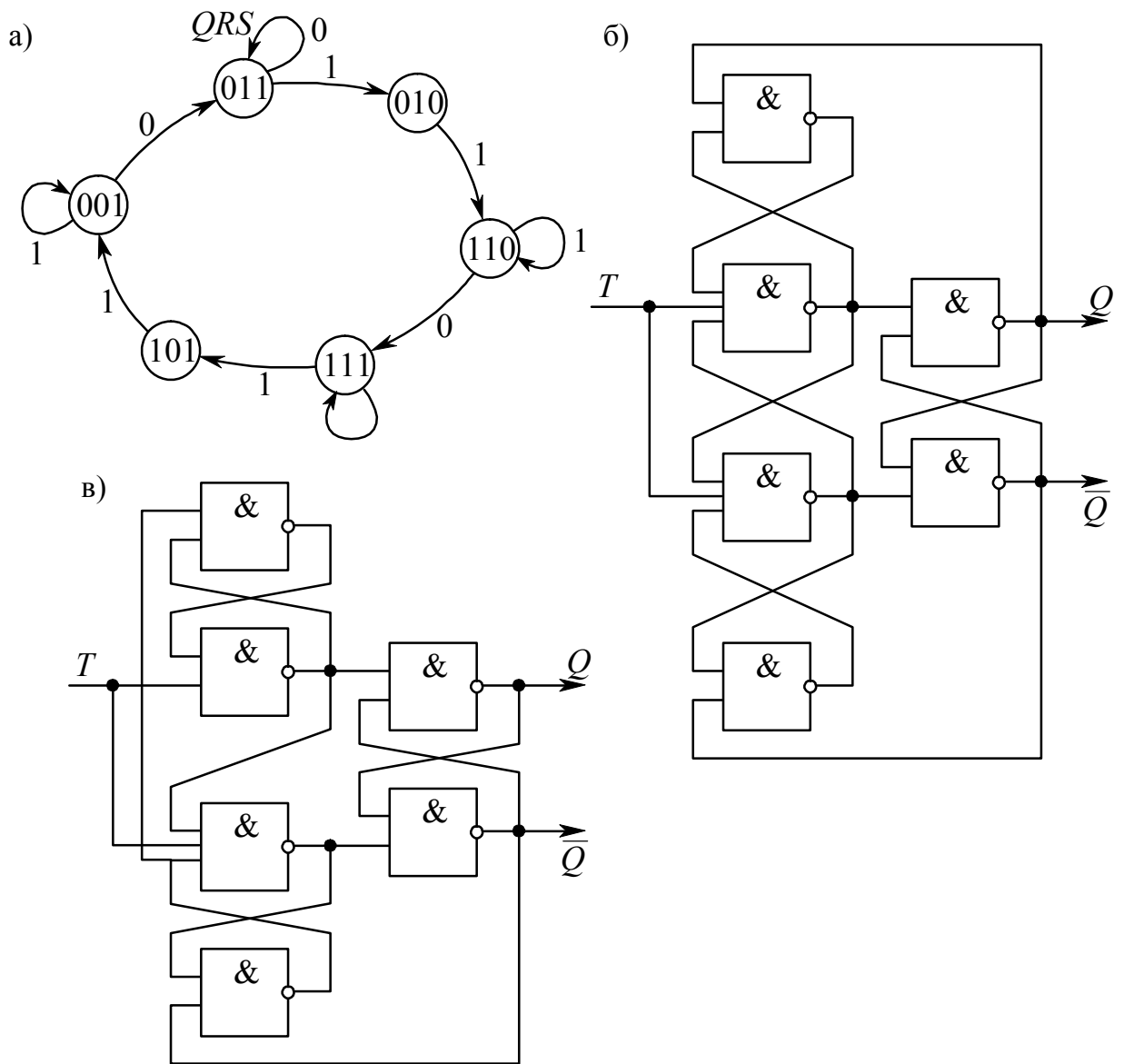


Рис. 2.12. T -триггер с динамическим управлением: граф (а), симметричная схема (б) и несимметричная схема (в)

D - и DV -триггеры

Синхронный D -триггер (рис.2.13) имеет два входа: вход данных (D) и вход синхронизации (C).

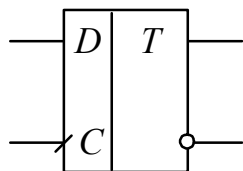


Рис. 2.13. Функциональное обозначение D -триггера

Таблица 2.21.

C^n	Q^n
0	Q^{n-1}
1	D^{n-1}

Такой триггер переходит в состояние, предписываемое сигналом на D -входе, только при появлении переключающего сигнала на входе C . Таблица переключений D -триггера приведена в табл.2.21. Такой триггер осуществляет задержку сигнала, поступающего на вход D , до появления сигнала на входе C , поскольку его выходной сигнал Q^n повторяет значение сигнала D на предыдущем такте (D^{n-1}) и сохраняет это состояние до следующего такта. Таким образом, максимальная задержка осуществляемая D -триггером один такт сигнала, поступающего на вход D . Такой D -триггер часто называют триггером задержки (название триггера происходит от слова «delay» – задержка).

Сравнивая таблицы переключений D - и T -триггеров, можно сделать вывод, что при $T = C$ и $D = \bar{Q}$ (рис.2.14,а) D -триггер будет функционировать как T -триггер.

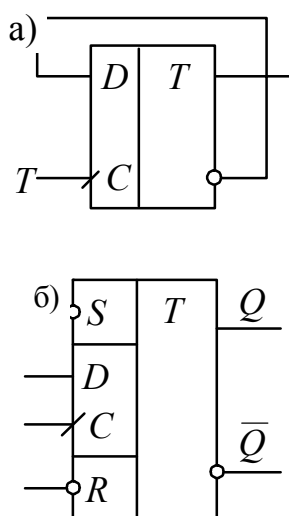


Рис.2.14. D -триггер и DRS -триггер

Отметим, что если в несимметричной схеме T -триггера (рис.2.12,в) исключить связь между выходом Q и входом и подавать на этот вход сигнал D , то такой T -триггер будет работать как D -триггер.

Дополненный установочными R - и S -входами D -триггер представляет собой DRS -триггер (рис.2.14,б), состояние которого определяется, в первую очередь, сигналами на установочных входах и при $R^n = S^n = 1$ зависит от значений сигналов на входах D и C . Комбинация $R^n = S^n = 0$, как и для RS -триггера, является запрещенной.

Модификацией D -триггеров являются DV -триггеры, имеющие дополнительный вход V (название происходит от слова «valve» – клапан), играющий роль разрешающего по отношению к входу D . При $V = 1$ триггер функционирует как D -триггер, а при $V = 0$ он сохраняет свое состояние при любых значениях сигналов на входах D и C . Тем самым наличие входа V позволяет в требуемые моменты времени сохранять информацию на выходах в течение заданного числа тактов, что расширяет функциональные возможности последовательностного устройства.

JK-триггеры

Эти триггеры (рис.2.15,а) являются универсальными синхронными триггерами, у которых входы J и K соответствуют входам S и R в RS -триггере с переключающими сигналами 1. Однако в отличие от RS -триггера состояние $J^n = K^n = 1$ считается допустимым, что приводит к изменению состояния триггера ($Q^n = \overline{Q^{n-1}}$).

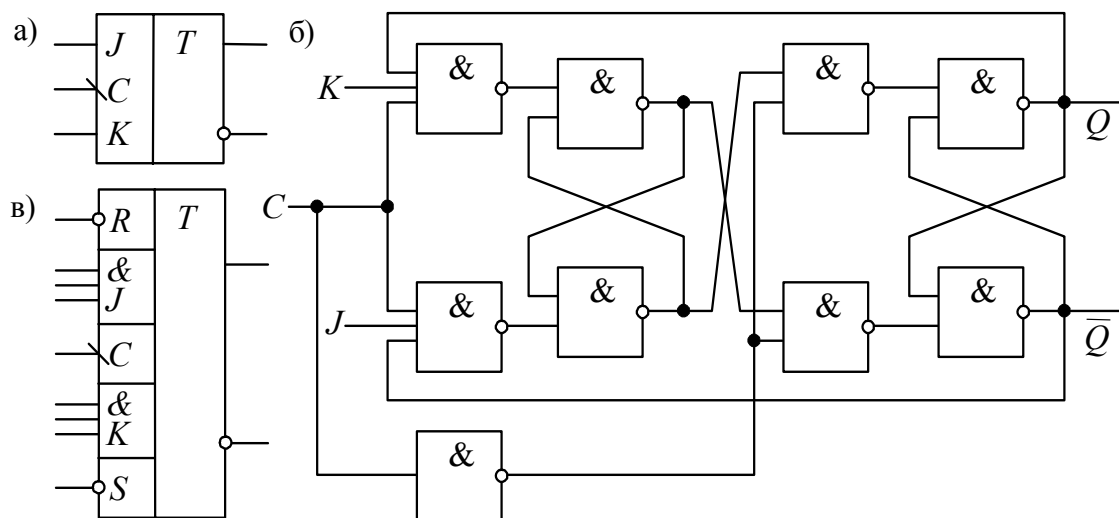


Рис.2.15. Условное обозначение JK-триггера (а), структура двухступенчатой схемы (б) и обозначение JKRS-триггера (в)

Поскольку JK -триггеры, реализуемые в виде интегральных схем, выполняются синхронными, то их выходные уровни устанавливаются только при поступлении на вход C переключающих сигналов¹. Таблица переключений JK -триггера дана в табл.2.22. Составляя карту Карно для Q^{n+1} (табл.2.23), получим с ее помощью структурную формулу для выходного сигнала триггера:

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

Таблица 2.22.

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

Таблица 2.23.

$Q^n \backslash KJ$	00	01	11	10
0		1	1	
1	1	1		

Это уравнение, называемое характеристическим уравнением триггера, полностью описывает его работу.

Пример схемы JK -триггера, имеющего структуру T -триггера, построенного по двухступенчатой схеме, приведен на рис.2.15,б. Путем определенного соединения входов и выходов JK -триггер может быть переведен в режим работы других типов триггеров. Так, при $J = K = 1$ и при наличии сигнала на входе C он выполняет функции T -триггера, а при $J = \overline{K}$ – D -триггера. Для расширения функциональных возможностей JK -триггеров вводятся асинхронные установочные входы (R, S) и схемы И по входам J и K . Обозначение такого $JKRS$ -триггера показано на рис.2.15,в.

¹ В практически реализуемых промышленностью JK -триггерах переключающим сигналом по входу C является перепад напряжений от логической 1 и логическому 0 (например, для ИС К155ТВ1), что и отражено в условном обозначении JK -триггера чертой с соответствующим наклоном, показывающим переход от 1 к 0 у входа C .

В заключение следует отметить, что различные схемы $RS-$ (TP^1), $D-$ (TM) и JK -триггеров (TB) имеются в составе большинства стандартных серий интегральных схем: ДТЛ, ТТЛ, ТТЛШ, ЭСЛ и КМОПТЛ. Как правило, в серии представлено несколько разновидностей каждого типа триггеров, отличающихся числом триггеров в корпусе и организацией входов управления.

2.3. Канонический метод проектирования последовательностных устройств

Построение первичного графа переходов, или составление первичной таблицы переходов на основании словесного описания является первым этапом проектирования последовательностных устройств.

При наличии словесной постановки задачи, т.е. словесного описания работы последовательностного устройства, первичный граф переходов проектируемого ПУ можно построить, используя следующий порядок действий.

- Определить входные и выходные сигналы проектируемого ПУ и построить его структурную схему, размещая входы слева, а выходы справа (проектируемые ПУ могут представлять собой специально выделенные модули-узлы), так что выходные сигналы одного модуля устройства будут входными сигналами другого и наоборот.
- Выявить причинно-следственные соотношения между входными и выходными сигналами.
- Определить состояние, с которого начинается последовательность событий. Для этого можно, например, выявить состояние, в которое желателен переход схемы по команде установки. На последующих этапах всем внутренним переменным этого состояния обычно присваиваются нулевые значения.
- Определить очередность событий (состояний) в схеме, начиная с подачи входных сигналов.
- Найти число переходов (линий) из каждого состояния (узла). Если при нахождении схемы в каком-либо состоянии на нее могут поступать (в зависимости от внешних условий) различные входные сигналы, то из соответствующего узла графа должны выходить несколько линий.

Анализ переходов из состояния в состояние следует продолжать до то-

¹ В скобках даны буквы, которые используются в отечественной системе обозначений класса и подкласса интегральных схем триггеров определённого типа.

го момента, пока последовательность состояний не образует цикл (замкнутый граф или подграф). Появление же запрещенного состояния, как правило, свидетельствует об ошибке при составлении графа переходов.

Если на первом этапе проектирования получен первичный граф переходов, то по этому графу составляется таблица переходов. Далее проводится минимизация числа состояний и строится сокращенная таблица переходов и сокращенный граф переходов. После этого осуществляется кодирование внутренних переменных и получается кодированная таблица и граф переходов.

На следующем этапе проектирования строят карты состояний и находят функции возбуждения внутренних переменных. Для этого, присвоив строкам кодированной таблицы переходов вместо десятичных цифр двоичные, равные коду состояний, и разместив строки в порядке, который предписывается картами Карно, преобразуют кодированную таблицу переходов в карты состояний (табл.2.24). Для наглядности в картах состояний следует строки и столбцы поменять местами.

В случае записи в клетках карты состояний двоичных цифр вместо десятичных цифр (табл.2.25), появляется возможность определить функции возбуждения внутренних переменных. Так, для табл.2. 25 имеем:

$$z_1^{n+1} = xz_2 + \bar{x}z_1;$$

$$z_2^{n+1} = xz_2 + \bar{x}z_1.$$

Таблица 2.24.

	z_1z_2	00	01	11	10
x	0	3	(3)	1	(1)
	1	(2)	4	(4)	2

Таблица 2.25.

	z_1z_2	00	01	11	10
x	0	01	01	10	10
	1	00	11	11	00

Составляя карты выходных переменных (если выходные переменные не совпадают с внутренними переменными), находят функции выходных

переменных.

Заключительным этапом проектирования последовательного устройства является построение структурной схемы, или структурный синтез ПУ.

Рассмотрим словарный метод структурного синтеза. Если в проектируемом последовательном устройстве в качестве элементов памяти предполагается использование триггеров, то на последнем этапе структурного синтеза можно применить словарный метод [1], суть которого заключается в следующем.

Карта состояний (табл.2.25) преобразуется в карту функций переходов F_Q внутренних переменных, которые обозначим Q . При этом $Q^n = Z^n$, $Q^{n+1} = Z^{n+1}$, а F_Q есть переход $Q^n \rightarrow Q^{n+1}$. Результат такого преобразования представлен в табл.2.26,а, где 0, 1, ∇ и Δ – условные обозначения перехода внутренних переменных соответственно из $0 \rightarrow 0$; $1 \rightarrow 1$; $1 \rightarrow 0$ и $0 \rightarrow 1$.

Затем составляют карту функций возбуждения [1] каждого триггера. Для этого символы функций переходов в карте функций переходов заменяют значениями входных сигналов триггера, например, \bar{R} и \bar{S} , которые необходимы для реализации данного перехода и которые определяются по словарю переходов избранного триггера (табл.2.26,б). Полученные таким образом карты представлены в табл.2.27.

$Q_1 Q_2$		$Q_1 Q_2$						$Q_1 Q_2$				
		00	01	11	10			00	01	11	10	
x	0	1	1	Φ	Φ	\bar{S}_1	0	0	Φ	1	1	\bar{S}_2
	1	1	0	Φ	1		1	1	Φ	Φ	1	
	0	Φ	Φ	1	1	\bar{R}_1	0	1	1	0	Φ	\bar{R}_2
	1	Φ	1	1	0		1	Φ	1	1	Φ	

Таблица 2.26.

а)

	Q_1Q_2			
x	00	01	11	10
0	0 Δ	0 1	1 ∇	1
1	0 0	Δ 1	1 1	∇

Таблица 2.27.

б)

F_Q	\bar{S}	\bar{R}
0	1	\times
1	\times	1
Δ	0	1
∇	1	0

Определив по картам функции возбуждения, преобразуем их для реализации в заданном базисе, например, в базисе И-НЕ. Имеем:

$$\begin{aligned} \bar{S}_1 &= \bar{x} + \bar{Q}_2 = \overline{xQ_2}; & \bar{R}_1 &= \bar{x} + Q_2 = \overline{x \cdot \bar{Q}_2}; \\ \bar{S}_2 &= \bar{x} + \bar{Q}_1 = \overline{xQ_1}; & \bar{R}_2 &= x + Q_1 = \overline{\bar{x} \cdot \bar{Q}_1}. \end{aligned}$$

Полученные формулы позволяют построить структурную схему последовательностного устройства (рис.2.16,а).

Поскольку карты функций возбуждения (табл.2.27) содержат неопределенные значения этих функций, имеется возможность представить ряд иных вариантов структурной реализации синтезируемого последовательностного устройства, провести их анализ и выбрать лучший с точки зрения функциональной надежности, вариант практической реализации. В этом заключается достоинство словарного метода структурного синтеза.

Проиллюстрируем сказанное одним из примеров. Доопределим карты функций \bar{S}_2 и \bar{R}_2 (табл.2.27), приняв значения функции \bar{S}_2 в клетках с двоичными номерами 001 и 001 и значения функции \bar{R}_2 в клетках 010 и 110 равным и нулю. Тогда имеем:

$$\begin{aligned} \bar{S}_2 &= Q_1 + x \cdot \bar{Q}_2 = Q_1 + R_1 = \overline{\bar{Q}_1 \cdot \bar{R}_1}; \\ \bar{R}_2 &= \bar{Q}_1 + x \cdot Q_2 = \bar{Q}_1 + S_1 = \overline{Q_1 \cdot \bar{S}_1}. \end{aligned}$$

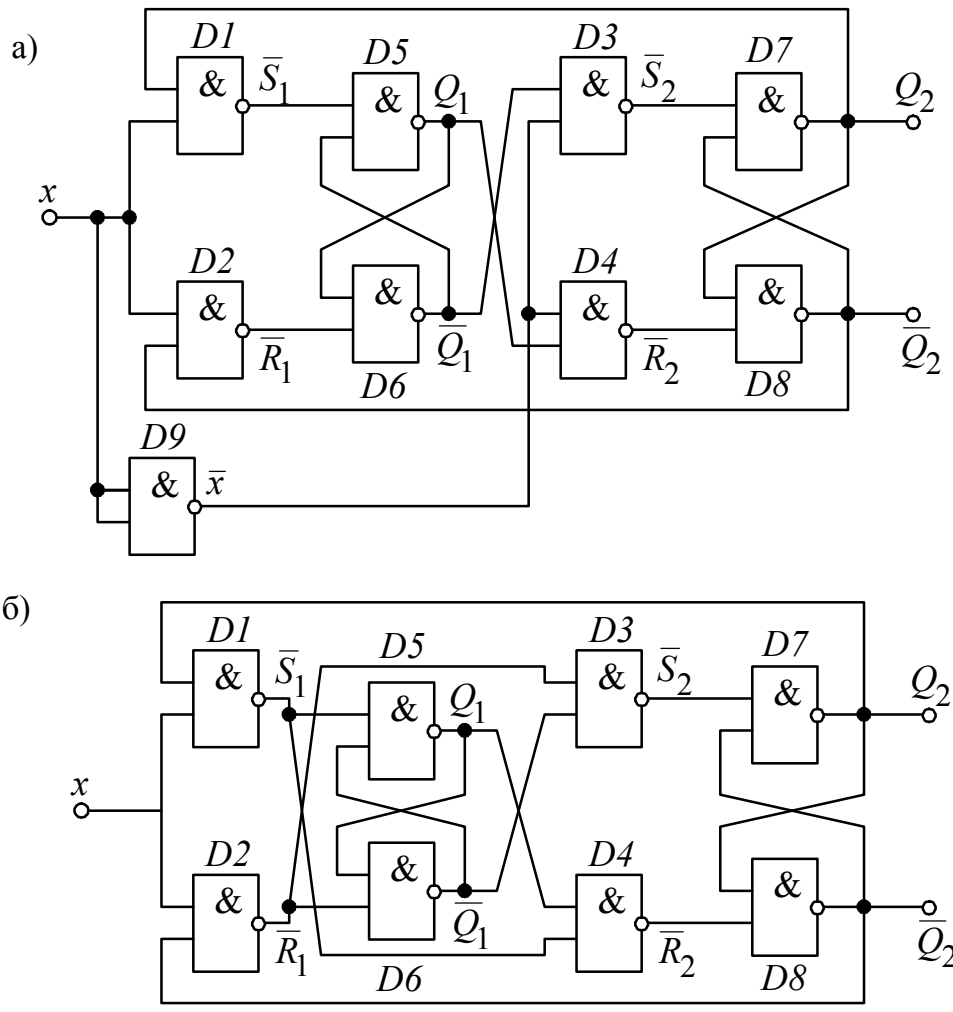


Рис.2.16. Варианты структурных схем T-триггеров

В последовательном устройстве, схема которого представлена на рис.2.16,а, могут возникнуть опасные состязания сигналов по следующим параллельным цепям: входы $D9$ – выход $D9$, $D1 - D5$ – выход Q_1 и $D2 - D6$ – выход \bar{Q}_1 . Так, например, в случае, когда входная переменная x изменяет свое значение с 0 на 1, а \bar{x} с 1 на 0, сигнал блокировки логических элементов $D3$ и $D4$ сигналом \bar{x} может задержаться относительно момента времени, соответствующего изменению сигналов Q_1 и \bar{Q}_1 на выходах $D5$ и $D6$. Это может вызвать неалгоритмический переход и изменение входных переменных Q_2 и \bar{Q}_2 .

Во втором варианте ПУ (рис.2.16,б) такие опасные состязания отсутствуют. Кроме того, ПУ содержит на один логический элемент меньше, чем аналогичное ПУ, показанное на рис.2.16,а.

2.4. Синтез триггеров

Простейшие триггеры, реализованные в виде интегральных схем, по-

зволяют проектировать цифровые устройства и их отдельные узлы. В частности, на их основе строятся счетчики и регистры.

Рассмотрим синтез триггера как простейшего последовательностного устройства на примере D -триггера. Исходными данными для синтеза является таблица переходов (зависимость выходных сигналов от входных) и требуемые свойства проектируемого триггера. Целью синтеза является поиск уравнения выхода и системы уравнений переходов.

Поскольку на первом этапе проектирования неизвестно число внутренних переменных (Q), то, следовательно, нет возможности записать различные предусмотренные состояния в окончательной таблице переходов. Поэтому вначале составляют так называемую первоначальную таблицу переходов и выходов, которая содержит максимально возможное число внутренних устойчивых состояний. Число столбцов первоначальной таблицы равно числу различных состояний входов триггеров. Для D -триггера оно составляет 4, так как D -триггер имеет два входа: C и D . Число строк первоначальной таблицы можно определить, если каждому возможному состоянию входов и выходов триггера сопоставить одно устойчивое внутреннее состояние. Для D -триггера, имеющего два входа и один выход, таблица будет содержать 8 строк.

Каждое устойчивое внутреннее состояние нумеруют и записывают в отдельной строке первоначальной таблицы переходов и в столбце, который соответствует значению входных переменных для этого внутреннего состояния. Значение выхода Z , соответствующее каждому устойчивому внутреннему состоянию D -триггера, записывают в отдельном столбце табл.2.28. Таким образом, каждый столбец табл.2.28 определяет состояние входных переменных, а каждая строка – состояние внутренних переменных. Полное устойчивое состояние D -триггера будет условно определяться пересечением строки и столбца таблицы. Это состояние записывается цифрой в круглых скобках. В каждой строке первоначальной таблицы переходов имеется лишь одно устойчивое состояние.

Следующим шагом заполнения первоначальной таблицы является запись переходов триггера из одного устойчивого состояния в другое. Ее производят на основании таблицы переходов и требуемых свойств триггера. Проектируемый D -триггер переключается при изменении состояния синхронизирующего входа из 1 в 0.

Переход из одного устойчивого состояния в другое возможен только при изменении значений входных переменных, что выражается сменой

столбца в таблице переходов. Для того, чтобы в последней отобразить переход триггера из одного устойчивого состояния в другое, необходимо на пересечении строки, означающей предыдущее состояние D -триггера, со столбцом таблицы, определяющим новое значение входных переменных, записать номер того внутреннего устойчивого состояния, в которое должен перейти D -триггер в соответствии с исходными данными. Этот номер в скобки не заключается, так как он соответствует неустойчивому состоянию D -триггера. В таблице переходов возможны только горизонтальное и вертикальное перемещения. Горизонтальное перемещение определяется изменением значений входных переменных схемы D -триггера. Вертикальное перемещение происходит самопроизвольно вследствие изменения значений внутренних переменных.

Таблица 2.28.

Внутренние состояния	Состояния входов CD				Состояние выхода Z
	00	01	11	10	
1	(1)				0
2		(2)			0
3			(3)		0
4				(4)	0
5	(5)				1
6		(6)			1
7			(7)		1
8				(8)	1

Через неустойчивое состояние D -триггер проходит в том случае, когда смена входных сигналов изменяет состояние входов, а выходы блоков памяти (из-за имеющихся в них задержек) еще не изменили своего состояния (см. рис.2.1). По истечении времени задержки устройство самопроизвольно переходит в новое устойчивое состояние.

Рассмотрим запись переходов. Предположим, что $C = D = 1$, $Z = 0$. Эта комбинация соответствует полному состоянию (3) в табл.2.28. Допустим, что входная переменная C изменила свое значение и стала равной 0, т.е. состояние входов CD равно 01. Поскольку переключаящим сигналом C является изменение значение сигнала 1 на 0, то в соответствии с таблицей внешних переходов D -триггер должен изменить свое выходное состояние. Новым устойчивым состоянием должно быть состояние (6), которое соответствует $C = 0$, $D = 1$, $Z = 1$. Чтобы обозначить этот переход

в табл.2.28, необходимо на пересечении третьей строки и столбца состояние входов CD равного 01 записать цифру 6 без скобок. Это означает, что соответствующее состояние неустойчиво и что триггер должен перейти к состоянию, обозначенному (6). Если же переход из (3) осуществляется изменением переменной D , т.е. состояние входов CD равно 10, то новым устойчивым состоянием должно быть состояние (4), поскольку изменения на логическом входе при том же состоянии синхронизирующего входа не должны переключать D -триггер. Таким образом на этапе заполнения первоначальной таблицы переходов учитывается асинхронный режим работы проектируемого синхронного D -триггера, т.е. его тип.

Переход из устойчивого состояния (3), обусловленный одновременным изменением переменных C и D , должен быть запрещен, поскольку в этом случае физически невозможно обеспечить однозначность перехода триггера из одного состояния в другое. Действительно, реально осуществить одновременное изменение значений C и D невозможно. Поэтому переход от состояния CD , равного 11, к CD , равному 00 будет проходить через промежуточные состояния входов CD , равные 01 или 10 в зависимости от порядка изменения значений входных переменных, т.е. возможны следующие переходы: состояния входов CD равны: $11 \rightarrow 01 \rightarrow 00$ или $11 \rightarrow 10 \rightarrow 00$. При первом изменении состояния входов D -триггер в соответствии с первоначальной таблицей переходов (табл.2.29) пройдет состояния $(3) \rightarrow 6 \rightarrow (6) \rightarrow 5 \rightarrow (5)$, и Z станет равным 1. При втором – $(3) \rightarrow 4 \rightarrow (4) \rightarrow 1 \rightarrow (1)$, и состояние выхода Z будет равно 0.

Таким образом, в зависимости от порядка изменения входных сигналов D -триггер может оказаться в состоянии 1 или 0. В таблице переходов запрещенные переходы будем обозначать символом (\times). Аналогичным образом заполним всю табл.2.28, переходя к табл.2.29. Очевидно, что при составлении таблицы все основные свойства проектируемого D -триггера, сформулированные словесно, выражаются в виде первоначальной таблицы переходов.

Таблица 2.29.

Внутренние состояния	Состояния входов CD				Состояние выхода Z
	00	01	11	10	
1	(1)	2	×	4	0
2	1	(2)	3	×	0
3	×	6	(3)	4	0
4	1	×	3	(4)	0
5	(5)	6	×	8	1
6	5	(6)	7	×	1
7	×	6	(7)	8	1
8	1	×	7	(8)	1

Сложность логической структуры проектируемого триггера зависит от числа внутренних переменных, поэтому целесообразно минимизировать число внутренних состояний, равное числу строк первоначальной таблицы переходов. Под минимизацией числа внутренних состояний последовательностного устройства понимается процесс, целью которого является получение ПУ, имеющего минимальное число внутренних состояний среди всех последовательностных устройств, реализующих заданные условия его работы. Процесс совмещения строк таблицы переходов называют объединением и выполняют его в соответствии со следующими правилами [5]:

- две строки (или более) могут быть объединены, если в соответствующих столбцах номера состояний одинаковы или на одной из строк имеется запрещенное состояние;
- результирующее состояние заключают в скобки в том случае, если одни объединяемые номера в скобках, а другие объединяемые номера без скобок. Если же некоторые строки содержат цифры, а другие символ (×), то в объединенной строке должна стоять цифра.

Рассмотрим первую и четвертую строки табл.2.29. Их объединение в соответствии с данными правилами дает новую строку вида: (1); 2; 3; (4). Переход от состояния (4) к (1) в новой объединенной строке осуществляется изменением только входной переменной C без изменения внутреннего состояния.

Два варианта объединения строк табл.2.29 иллюстрируются сокращенными таблицами переходов (табл.2.30). Во втором варианте (табл.2.30,б), в отличие от первого (табл.2.30,а), переходы из одной строки

в другую осуществляется только при изменении значения переменной C . В первом варианте (табл.2.30,а) смена устойчивого состояния может быть вызвана изменением логической переменной D (без изменения значения состояния выхода Z).

При проектировании двухступенчатых D -триггеров целесообразно использовать объединение строк первоначальной таблицы, приводящее к табл.2.30,а, а при построении D -триггеров с динамическим управлением – к табл.2.30,б.

Таблица 2.30.

а)

Внутренние состояния	Состояния входов				Состояние выхода
	00	01	11	10	
1, 2, 4	(1)	(2)	3	(4)	0
3		6	(3)	4	0
5, 6, 7	(5)	(6)	(7)	8	1
8	1	×	7	(8)	1

б)

Внутренние состояния	Состояния входов				Состояние выхода
	00	01	11	10	
1, 2	(1)	(2)	3	4	0
3, 4	1	6	(3)	(4)	0
5, 6	(5)	(6)	7	8	1
7, 8	1	6	(7)	(8)	1

Следует отметить, что хотя число внутренних состояний последовательностного устройства определяет число элементов памяти, сокращение числа внутренних состояний ПУ не всегда приводит к уменьшению числа элементов памяти. Но даже и в этом случае целесообразно производить минимизацию числа внутренних состояний, так как при этом увеличивается число неиспользуемых состояний, что упрощает структуру логического преобразователя¹. Следующие этапы синтеза осуществляются в соответствии с каноническими методами проектирования последовательностных

¹ Иногда минимизация числа внутренних состояний может привести к возникновению состязаний сигналов в элементах памяти.

устройств, рассмотренными в разделе 2.3.

2.5. Регистры

Основная функция регистра заключается в хранении многоразрядного двоичного числа. Обычно регистры имеют регулярную структуру и состоят из одинаковых частей. Каждая из таких частей (разряд регистра) предназначена для хранения одного разряда двоичного числа. Разряд регистра содержит элемент памяти, выполненный на триггере того или иного типа. Регистр может работать в режимах записи, хранения и считывания чисел. В режиме записи в регистр вводится двоичное число. В режиме хранения записанное число остается без изменения, а при считывании – число передается из регистра в другие устройства.

По способам организации приема и считывания чисел различают следующие типы регистров:

- **параллельные**: запись и считывание во всех разрядах осуществляется одновременно;
- **последовательные (регистры сдвига)**: запись и считывание данных осуществляются последовательно;
- **параллельно-последовательные**: в регистрах этого типа запись осуществляется параллельно, а считывание последовательно;
- **последовательно-параллельные**: в отличие от предыдущего типа здесь запись производится последовательно, а считывание параллельно;
- **реверсивные**: направление сдвига чисел в них может быть изменено в зависимости от значений управляющих сигналов.

С помощью регистров осуществляются преобразования последовательного двоичного кода в параллельный и наоборот. Они используются для формирования временной задержки цифровых сигналов и выполнения арифметических и логических операций с многоразрядными числами. Рассмотрим основные принципы построения перечисленных выше типов регистров.

Структура параллельного регистра (рис.2.17,а) представляет собой совокупность однотипных ячеек (рис.2.17,б и в), содержащих триггер и схемы управления записью W_1 и считыванием W_2 . Каждая ячейка (рис.2.17,а) имеет информационный вход D_i . Входы управления записью (C_1) и считыванием (C_2) являются общими для всех ячеек. Правило работы для триггера в i -ой ячейке формулируется следующим образом. Если

$C_1^n = 1, C_2^n = 0$, то $Q_i^n = x_i^n; y_i^n = 0$.

При $C_1^n = C_2^n = 0$ $Q_i^n = Q_i^{n-1}; y_i^n = 0$, т. е. осуществляется режим хранения числа. В случае, когда $C_1^n = 0, C_2^n = 1$, триггер не переключается ($Q_i^n = Q_i^{n-1}$), и входной сигнал проходит на выход $y_i^n = Q_i^n$. Комбинация $C_1^n = C_2^n = 1$ считается запрещенной. Выбрав в качестве элемента памяти разряда регистра *RS*-триггер, можно составить таблицы истинности для выходной функции y_i^n и сигналов на входах *RS*-триггера (табл.2.31 и 2.32). Записав по таблицам структурные формулы, получим:

$$y_i^n = C_2^n \cdot Q_i^n = \overline{\overline{C_2^n \cdot Q_i^n}};$$

$$S = \overline{x_i \cdot C_1};$$

$$R = \overline{\overline{\overline{x_i} \cdot C_1 + C_1 \cdot \overline{C_1}}} = \overline{C_1 \cdot (\overline{x_i} + \overline{C_1})} = \overline{C_1 \cdot \overline{x_i C_1}} = \overline{C_1 \cdot S}$$

В соответствии с этими формулами схема одного разряда регистра имеет вид, показанный на рис.2.17,г.

Последовательный регистр или регистр сдвига (рис.2.18,а) имеет информационный вход и тактовый, обеспечивающий запись числа и сдвиг его по регистру. Структура регистра (рис.2.18,б) представляет

Таблица 2.31.

C_2	Q_i	y_i
0	0	0
0	1	0
1	0	0
1	1	1

Таблица 2.32.

C_1	x_i	R	S
0	0	1	1
0	1	1	1
1	0	0	1
1	1	1	0

последовательное соединение однотипных ячеек, и правило работы для i -го и $(i-1)$ -го разряда можно записать в следующем виде: если $C^n = 0$, то

$Q_{i-1}^n = Q_{i-1}^{n-1}$, а при $C^n = 1$ получим $Q_i^n = Q_{i-1}^{n-1}$. Такая запись совпадает с правилом работы D -триггера, на вход которого поступает сигнал Q_{i-1} . Следовательно, для построения регистра сдвига можно выбрать D -триггеры или JK -триггеры, работающие в режиме D -триггера. Соответствующие схемы регистров представлены на рис.2.18,в и г.

Параллельно-последовательный регистр (рис.2.19,а) работает по следующему правилу: при $C_1^n = 1$ и $C_2^n = 0$ производится запись во все разряды регистра сигналов со входов x_1, \dots, x_m , так же как в параллельном регистре; при $C_1^n = 0$ и $C_2^n = 1$ осуществляется сдвиг чисел, записанных в регистре, т.е. регистр работает как последовательный. В качестве элемента памяти в каждом разряде может использоваться DRS -триггер, у которого входы R и S предназначены для параллельной записи, а входы D и C – для организации последовательного сдвига чисел. Схема параллельно-последовательного регистра на DRS -триггерах представлена на рис.2.19,б. С помощью этого типа регистров легко осуществить преобразование параллельного кода в последовательный.

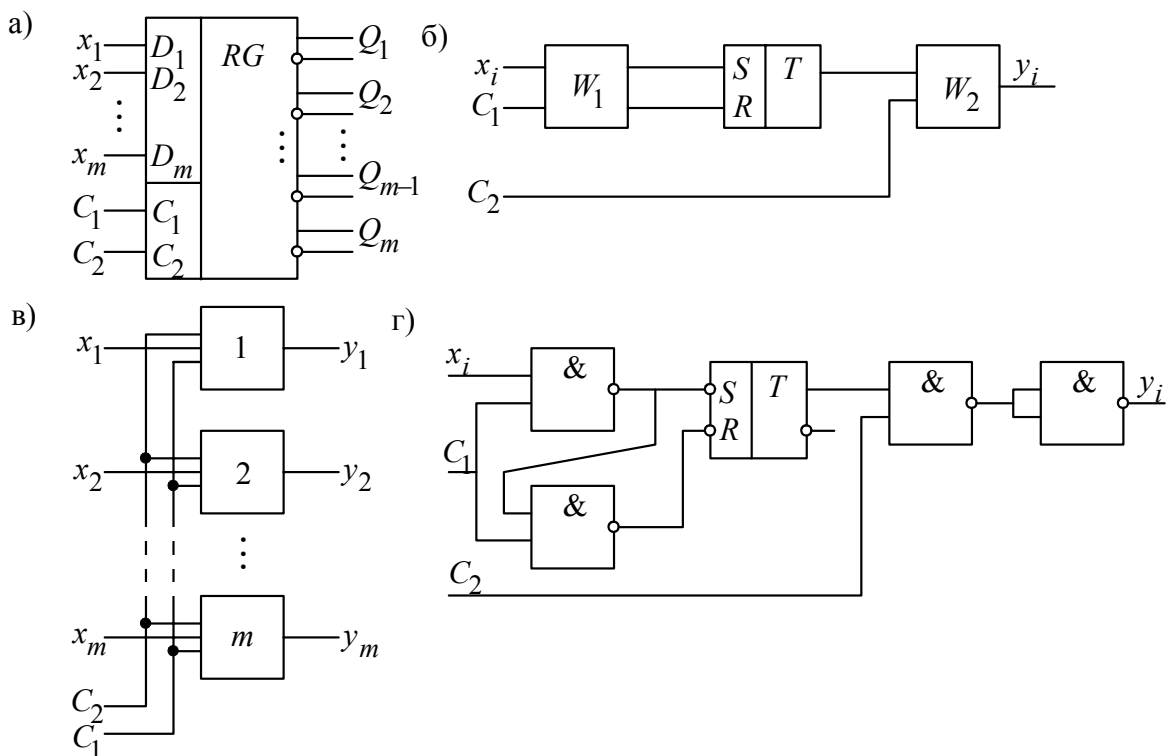


Рис. 2.17. Параллельный регистр: условное обозначение (а), схема разряда регистра (б), структурная схема регистра (в) и реализация разряда регистра (г)

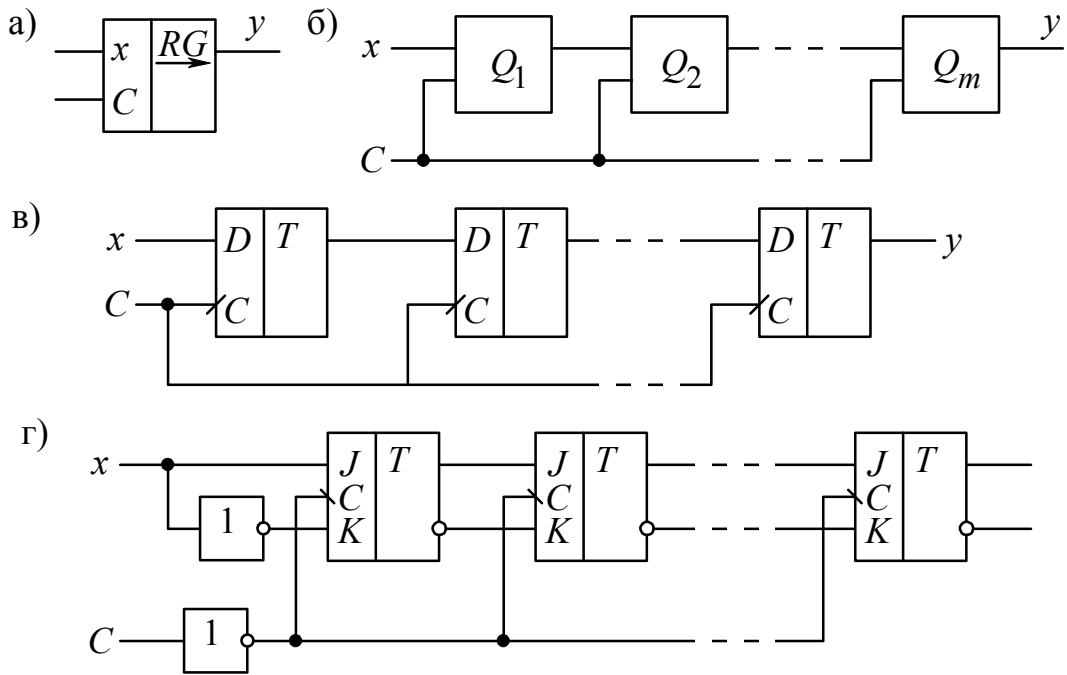


Рис. 2.18. Регистр сдвига: условное обозначение (а), структурная схема (б), реализация на D -триггерах (в) и JK -триггерах (г)

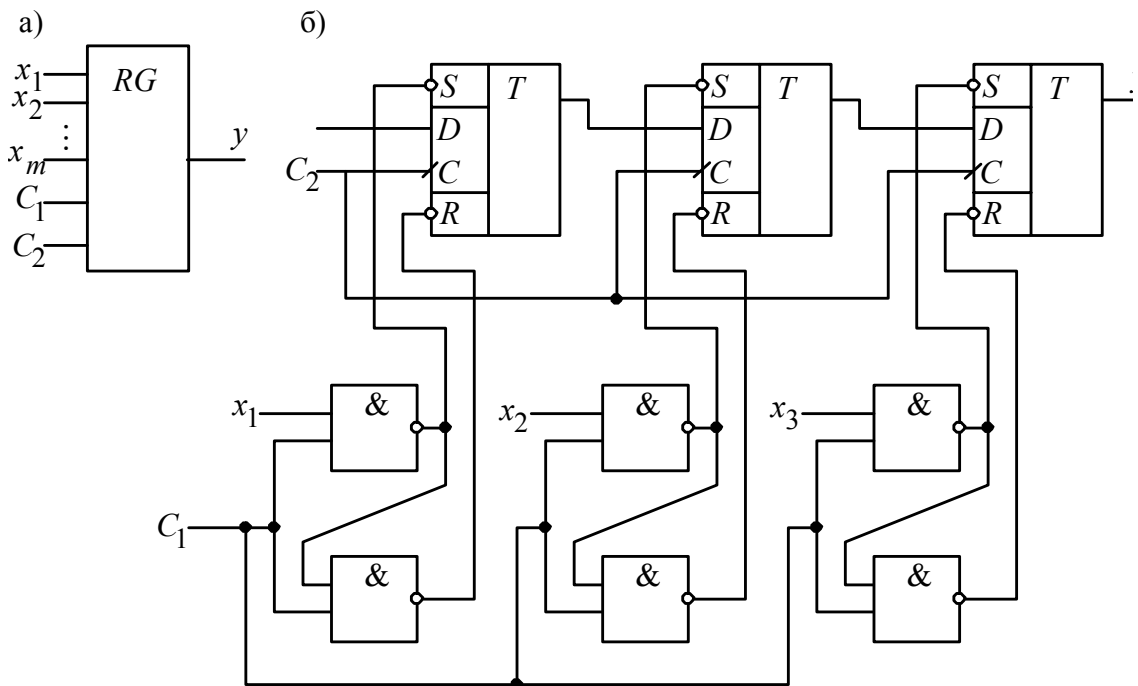


Рис. 2.19. Параллельно-последовательный регистр: условное обозначение (а), схема 3-х разрядного регистра (б)

Последовательно-параллельный регистр (рис.2.20,а) имеет вход для последовательного ввода данных в регистр при наличии переключающих сигналов на входе C_1 и m выходов (y_1, \dots, y_m), на которых появляются двоичные числа при переключающем сигнале на входе C_2 . При $C_1^n = 1$, $C_2^n = 0$ производится сдвиг чисел, как в последовательном регистре. При

$C_1^n = 0, C_2^n = 1$ считывание происходит одновременно из всех разрядов.

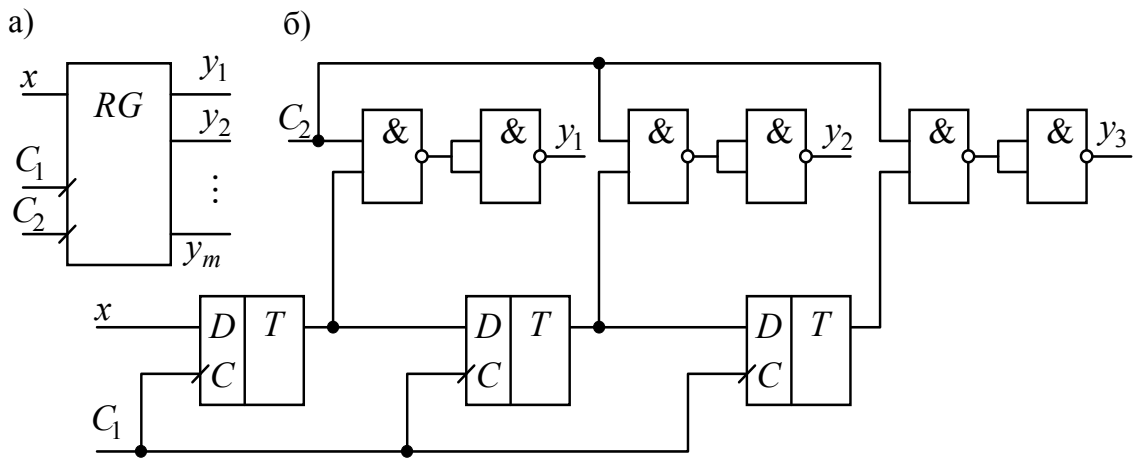


Рис. 2.20. Последовательно-параллельный регистр: условное обозначение (а) и схема (б)

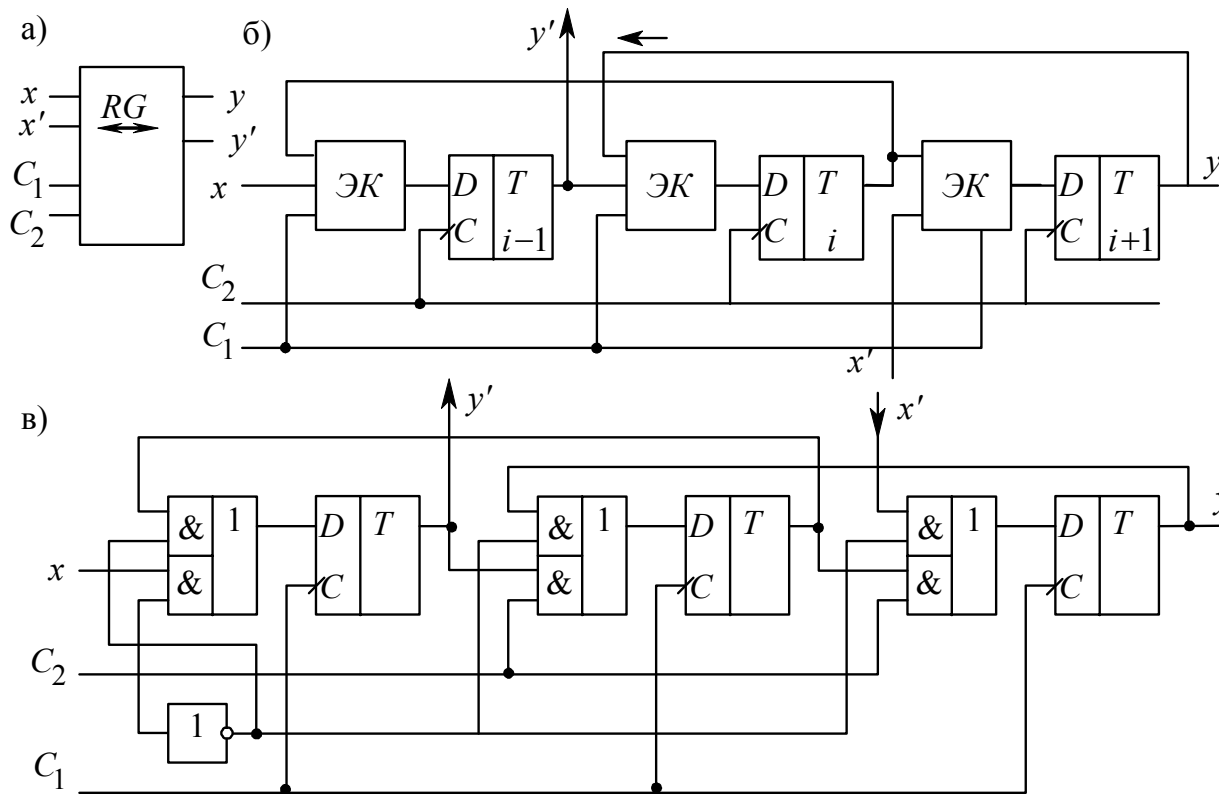


Рис. 2.21. Реверсивный регистр сдвига: условное обозначение (а), структурная схема (б) и реализация на D -триггерах (в)

Если к последовательному регистру на D -триггерах добавить схемы управления, то получим схему последовательно-параллельного регистра (рис.2.20,б), осуществляющего преобразование последовательного кода в параллельный.

Реверсивный последовательный регистр (рис.2.21,а) представляет собой регистр сдвига, в котором направление сдвига определяется управляющими сигналами на входе C_1 . Структура регистра (рис.2.21,б) содержит D -триггеры и элементы коммутации (ЭК). При реализации сдвига чисел в одном направлении элемент коммутации должен передавать сигнал с выхода предыдущего ($(i-1)$ -го) триггера на вход последующего (i -го), а при сдвиге в обратную сторону с выхода $(i+1)$ -го на вход i -го триггера. Следовательно, элемент коммутации представляет собой комбинационное устройство, подающее сигнал на вход i -го D -триггера. При $C_1 = 1$ имеем $D_i = Q_{i-1}$, а при $C_1 = 0$ $D_i = Q_{i+1}$. Построив по таблице истинности (табл.2.33) для выходного сигнала i -го элемент коммутации $D_i = f(C_1, Q_{i-1}, Q_{i+1})$ карту Карно (табл.2.34), можно получить структурную формулу:

Таблица 2.33.

C_1	Q_{i-1}	Q_{i+1}	D_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

$$D_i = \bar{C}_1 \cdot Q_{i+1} + C_1 \cdot Q_{i-1}.$$

Таблица 2.34.

		$\bar{Q}_{i-1}\bar{Q}_{i+1}$	$\bar{Q}_{i-1}Q_{i+1}$	$Q_{i-1}Q_{i+1}$	$Q_{i-1}\bar{Q}_{i+1}$	
		00	01	11	10	
\bar{C}_1	0		1	1		D_i
C_1	1			1	1	

Схема реверсивного регистра, содержащая элементы коммутации, полученные при синтезе, и D -триггеры, представлена на рис.2.21,в.

На интегральных схемах средней степени интеграции реализуются регистры сдвига различной разрядности (число разрядов от 4 до 64), причем в корпусе одной интегральной схемы может быть размещено несколько 4- или 5-разрядных регистров различных типов. Среди наиболее широко применяемых регистров можно выделить три типа:

Регистры сдвига с параллельной записью. Пример 4-разрядного универсального регистра приведен на рис.2.22,а.

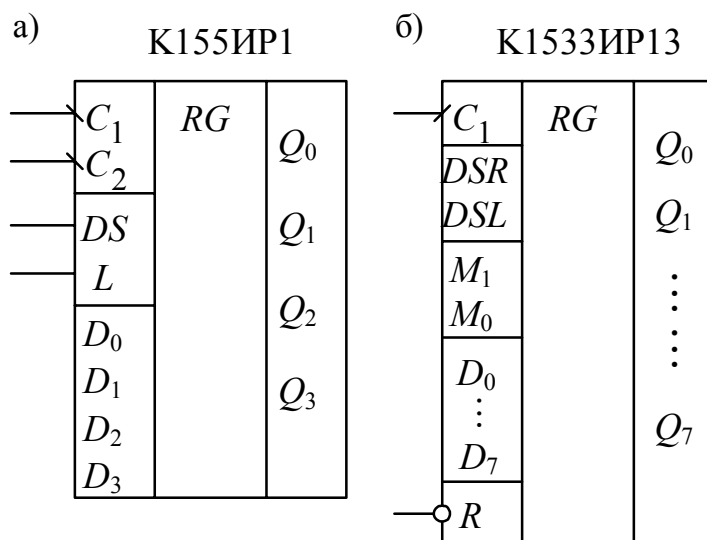


Рис.2.22. Универсальные регистры сдвига.

Он состоит из четырех тактируемых *JK*-триггеров с логическими элементами, осуществляющими коммутацию сигналов, управляющих состоянием триггеров. Триггеры регистра изменяют свое состояние при смене значений уровней сигнала с единицы до нуля на одном из тактовых входов C_1 или C_2 . Режим работы определяется значением сигнала на управляющем входе L . При $L = 1$ смена значений сигнала с 1 на 0 на входе C_2 приводит к установке триггеров в состояния, определяемые соответственно сигналами на входах $Q_0 \dots Q_3$, т. е. осуществляется параллельная запись. Когда $L = 0$, импульсы тактовой частоты на входе C_1 производят сдвиг чисел, поступающих на вход DS . При этом при считывании сигналов одновременно с выходов $Q_0 \dots Q_3$ осуществляется преобразование последовательного кода на входе DS в параллельный.

Регистры сдвига, в которых запись и считывание можно производить как параллельно, так и последовательно, а также осуществлять сдвиг чисел в том или ином направлении. Примером такого типа регистра является микросхема K1533IP13, представляющая собой 8-разрядный универсальный реверсивный регистр сдвига (рис.2.22,б), режим работы которого задается сигналами на управляющих входах M_1 , M_0 . В случае $M_1 = M_0 = 1$ осуществляется параллельная запись с входов $D_0 \dots D_7$. При $M_1 = 0$, $M_0 = 1$ происходит сдвиг чисел в одном направлении (при этом число вводится в регистр со входа DSR). Когда $M_1 = 1$, $M_0 = 0$, числа в регистре

сдвигаются в другом направлении. Входными сигналами для регистра при этом являются сигналы на входе *DSL*. Состояние входов управления режима $M_1 = M_0 = 0$ определяет режим хранения чисел в регистре. Этот тип регистров позволяет не только производить преобразования кодов (последовательного в параллельный, и наоборот), но и осуществлять операции поразрядного логического сложения и умножения.

Регистры третьего типа имеют управляющие входы, сигналы на которых изменяют характер режимов записи и считывания. Так, микросхема К564ИР9 (CD4035) может работать в следующих режимах: параллельная запись – параллельное считывание, параллельный ввод – последовательный вывод, последовательный ввод – параллельный вывод, последовательный ввод – последовательный вывод.

2.6. Счетчики

Счетчиком называется последовательностное устройство, имеющее несколько состояний, причем данное состояние счетчика определяется предыдущим его состоянием и значением логической переменной на входе. По мере поступления входных сигналов счетчик последовательно изменяет свои состояния в определенном порядке, возвращаясь к началу цикла после каждых K входных сигналов. Значение K , соответствующее числу состояний счетчика, является модулем счета или коэффициентом пересчета счетчика. Рассмотрим счетчики, построенные на основе триггеров. Код числа, записанного в счетчик, состоящий из m триггеров, отражает состояния триггеров и может быть представлен в следующем виде:

$$Q = Q_m \cdot Q_{m-1} \cdots Q_2 \cdot Q_1,$$

где Q_i – состояние i -го триггера.

Одно из возможных состояний счетчика принимается за начальное (исходное) – Q^0 . Запись Q^n соответствует состоянию счетчика после поступления на его вход n -го входного сигнала. Если порядок смены состояний триггеров соответствует последовательности двоичных чисел и число состояний счетчика соответственно равно $K = 2^m$, счетчик называется двоичным. В общем случае ($K = a^m$) устройство называют счетчиком по модулю числа a .

2.6.1. Двоичные счетчики

Рассмотрим двоичные счетчики с последовательным переносом. В та-

ких счетчиках входной сигнал воздействует только на первый триггер и каждый предыдущий триггер вырабатывает переключающий сигнал для последующего. Достоинствами этих счетчиков являются простота реализации и возможность наращивания их разрядности.

В суммирующем счетчике с последовательным переносом каждый входной импульс увеличивает значение двоичного числа, записанного в счетчик, на единицу. Работает такой счетчик согласно следующему правилу:

$$Q^n = Q^{n-1} + 1, \text{ если } Q^{n-1} \neq 2^m - 1;$$

$$Q^n = 0, \text{ если } Q^{n-1} = 2^m - 1.$$

Таблица 2.35.

Q^n			n
Q_3^n	Q_2^n	Q_1^n	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7
0	0	0	8

Правило работы суммирующего трехразрядного ($m = 3$) счетчика может быть представлено в виде таблицы переключений счетчика (табл.2.35). Используя таблицу переключений, можно определить требуемый тип триггера, ориентируясь на Q_1 , после чего выявить способ соединения триггеров. Очевидно, что первым (Q_1) должен быть T -триггер, так как он должен переключаться каждым входным сигналом. Поскольку из табл.2.35 следует, что второй триггер изменяет свое состояние, когда на выходе первого уровень 1 переходит в уровень 0, то необходимо сигнал на вход триггера подать с инверсного выхода первого. Аналогичные условия работы будут и у третьего триггера. Схема счетчика приведена на рис.2.23,а. Реализация суммирующих счетчиков на D - и JK -триггерах показана на рис.2.23,б и в.

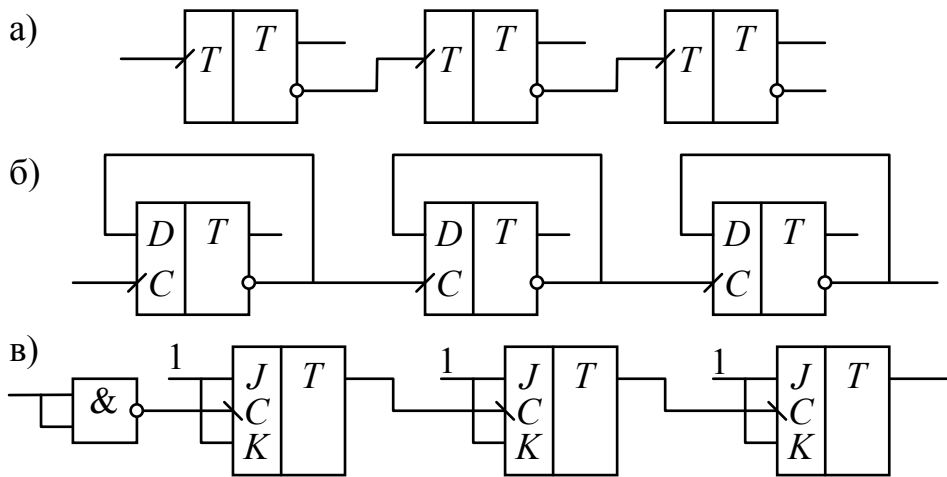


Рис. 2.23. Суммирующий двоичный счётчик: структурная схема (а), реализация на D -триггерах (б) и JK -триггерах (в)

В вычитающем счетчике с последовательным переносом с каждым входным импульсом уменьшается значение двоичного числа, записанного в счетчике. Правило его работы сводится к следующему:

$$Q^n = Q^{n-1} - 1, \text{ если } Q^{n-1} \neq 0;$$

$$Q^n = 2^m - 1, \text{ если } Q^{n-1} = 0.$$

Составив в соответствии с правилом работы таблицу переключений счетчика (табл.2.36), можно определить его схему. Полученные схемы вычитающего счетчика на T - и D -триггерах даны соответственно на рис.2.24,а и б.

Реверсивный счетчик может работать в качестве суммирующего и вычитающего. Выбор режима работы осуществляется с помощью управляющего сигнала, поступающего на вход F счетчика. Для построения реверсивного счетчика можно воспользоваться, как и ранее, последовательным соединением T -триггеров, дополнив его элементами коммутации. Словесное описание для выходного сигнала i -го элемента коммутации (T_i) можно получить, используя правило работы счетчика:

$$\text{при режиме суммирования имеем } T_i = \overline{Q_{i-1}},$$

$$\text{в режиме вычитания } T_i = Q_{i-1}.$$

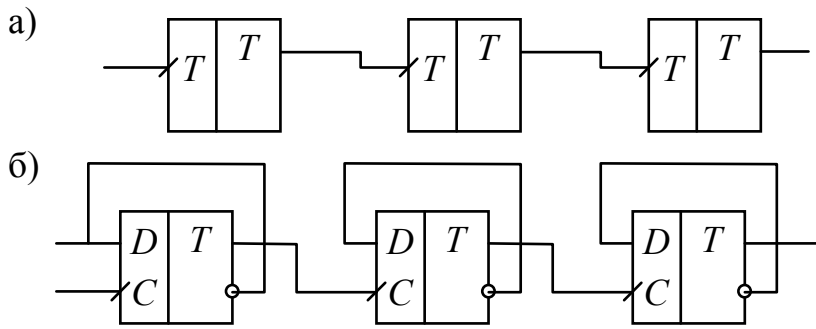


Рис. 2.24. Вычитающий двоичный счётчик: структурная схема на T -триггерах (а), реализация на D -триггерах (б)

Таблица 2.36.

Q^n			n
Q_3^n	Q_2^n	Q_1^n	
0	0	0	0
1	1	1	1
1	1	0	2
1	0	1	3
1	0	0	4
0	1	1	5
0	1	0	6
0	0	1	7
0	0	0	8

Таблица 2.37.

x	F	Q_1	T_2
0	0	0	0
0	0	1	0
0	0	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Следует учесть, что сигнал на входе F может изменяться только при сигнале на входе счетчика x , равном 0. Составленная по словесному описанию таблица истинности для входного сигнала i -го триггера (табл.2.37) позволяет записать для T_i структурную формулу

$$T_i = x\bar{F}Q_{i-1} + xFQ_{i-1} = x\bar{F}\overline{Q_{i-1}} + xFQ_{i-1}$$

и изобразить структурную схему элементы коммутации (рис.2.25)

Основным недостатком счетчиков с последовательным переносом является низкое быстродействие, что обусловлено последовательной работой триггеров. От этого недостатка свободны счетчики с параллельным переносом. В таких счетчиках входной сигнал может воздействовать на все триггеры счетчика одновременно и каждый предыдущий триггер вырабатывает управляющие сигналы для всех последующих. Рассмотрим построение суммирующего 3-разрядного ($m=3$) счетчика с параллельным переносом. Таблица переключений такого счетчика совпадает с таблицей переключений счетчика с последовательным переносом (табл.2.35). При

использовании JK -триггеров для первого триггера имеем $J_1 = K_1 = 1$.

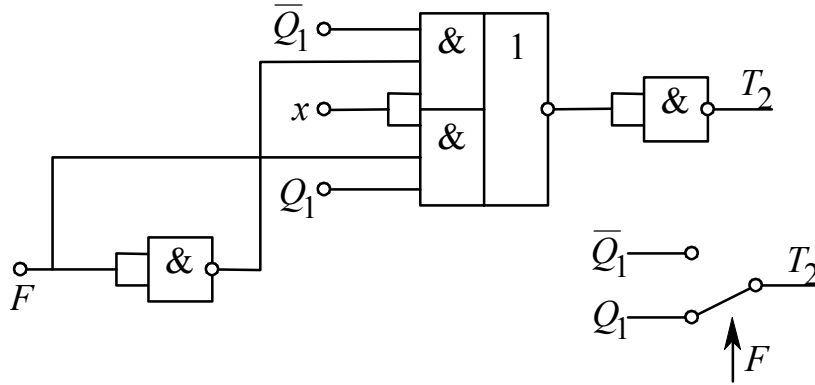


Рис. 2.25. Структурная схема элемента коммутации и его аналог

Управляющие сигналы для второго триггера описываются табл.2.38, а для третьего – табл.2.39. По таблицам получаем: $J_2 = K_2 = Q_1$, $J_3 = K_3 = Q_2Q_1$. Схема счетчика изображена на рис.2.26. Переключение триггеров в таком параллельном счетчике происходит одновременно и задержка переключения всего счетчика равна задержке переключения одного триггера. Повышение быстродействия такого счетчика достигается путем введения дополнительных логических элементов.

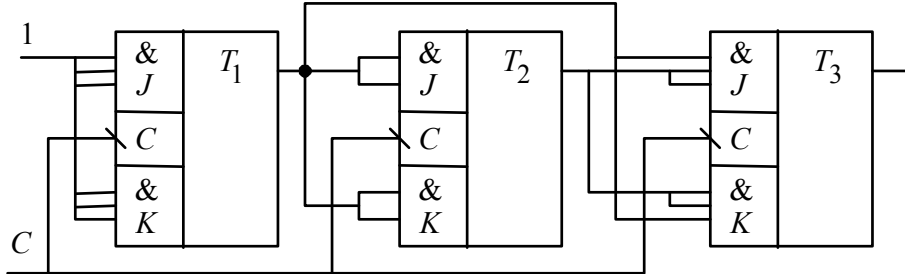


Рис. 2.26. Схема счётчика с параллельными переносами

Таблица 2.38.

Q_1	J_2	K_2
0	0	0
1	1	1

Таблица 2.39.

Q_2	Q_1	J_3	K_3
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

В счетчиках с параллельно-последовательным переносом триггеры, объединенные в группы, образуют счетчики с параллельным переносом, а группы соединяются как в счетчике с последовательным переносом.

2.6.2. Счетчики с произвольным коэффициентом пересчета

При построении счетчиков с произвольным коэффициентом пересчета K применяется метод исключения лишних состояний в двоичных счетчиках. Чтобы построить указанным методом счетчик с $K \neq 2^m$, используют двоичный счетчик с коэффициентом пересчета $2^m > K$. Определив число лишних состояний $l = 2^m - K$, осуществляют их исключение с помощью схем управления, дополняющих двоичный счетчик. В зависимости от выбранного способа счета и, соответственно, функций, возлагаемых на схемы управления, можно выделить четыре варианта реализации способа исключения «лишних» состояний:

1 Счетчик устанавливается в нулевое состояние дополнительной схемой, которая по состояниям выходов счетчика обнаруживает код окончания счета, равный $K - 1$ (рис.2.27,а).

2 Двоичный счетчик перед началом счета устанавливают в состояние, код которого является дополнением числа K до значения 2^m и, следовательно, равен числу исключаемых состояний l . Окончание счета определяется переполнением счетчика, т.е. появлением единиц во всех его разрядах и сигнала на выходе переноса (CP). Сигнал переноса должен снова установить начальное состояние счетчика, соответствующее значению l , что обычно осуществляется с помощью сигналов на входах параллельной записи (PL и $D_1 \dots D_M$).

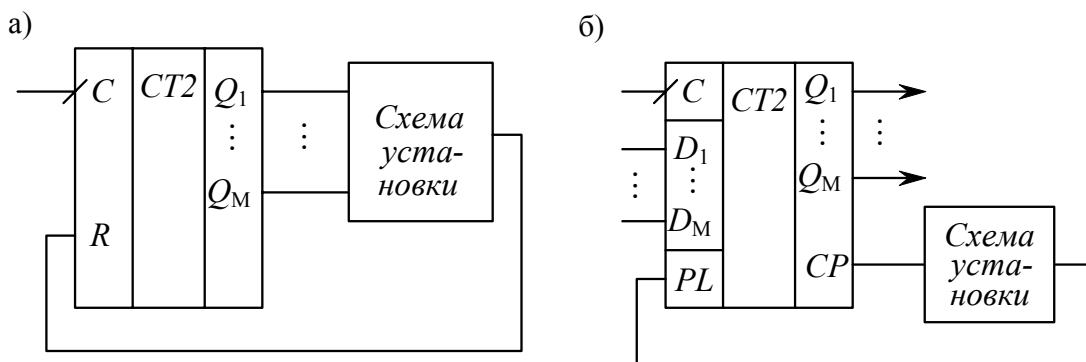


Рис.2.27. Структурные схемы счётчиков с произвольным коэффициентом пересчета

3. Предусматривается совместное использование предыдущих вариантов. Счетчик устанавливается в состояние, код которого $Q^0 \neq l$, и окончание счета определяется по состоянию, имеющему код, равный $Q^0 + (K - 1)$.

4. В двоичный счетчик вводят с помощью дополнительных логических элементов обратные связи между разрядами, ограничивающие число состояний счетчика.

В качестве примера рассмотрим использование второго варианта при построении счетчика с $K = 5$. По заданному значению K из соотношения $2^{m-1} < K < 2^m$ определяем минимальное число триггеров (разрядов) двоичного счетчика: $m = 3$. Построение таблицы переключений счетчика (табл.2.40) проведем с учетом использования части состояний триггеров последовательного счетчика при формировании сигнала начальной установки в случае перехода счетчика из состояния 111 в 000.

Кроме того, предусмотрим, чтобы сигнал установки формировался до начала появления следующего входного сигнала. Так как перед началом счета требуется прежде всего установить триггеры счетчика в начальное состояние, то используем *DRS*-триггеры (рис.2.28,а), причем первые три из них ($T_1 - T_3$) образуют счетчик с последовательным переносом, а четвертый — T_Φ — представляет собой триггер-формирователь импульса установки счетчика в исходное состояние (состояние Q^0 соответствует 011). Временные диаграммы работы такого триггера-формирователя даны на рис.2.28,б. Сигнал с выхода триггера-формирователя подается на установочные входы S тех триггеров счетчика, которым соответствует 1 в двоичном коде начального состояния счетчика. Очевидно, что при использовании первого варианта, схема двоичного счетчика может быть дополнена

дешифратором (цифровой схемой сравнения), выходной сигнал которого при возникновении на его входе кода конечного состояния счетчика должен сформировать сигнал управления на входы R всех триггеров счетчика.

Таблица 2.40.

Q^n			n	n
Q_3	Q_2	Q_1	дв.	$K = 5$
0	0	0	0	
0	0	1	1	
0	1	0	2	
0	1	1	3	0
1	0	0	4	1
1	0	1	5	2
1	1	0	6	3
1	1	1	7	4
0	0	0	8	5

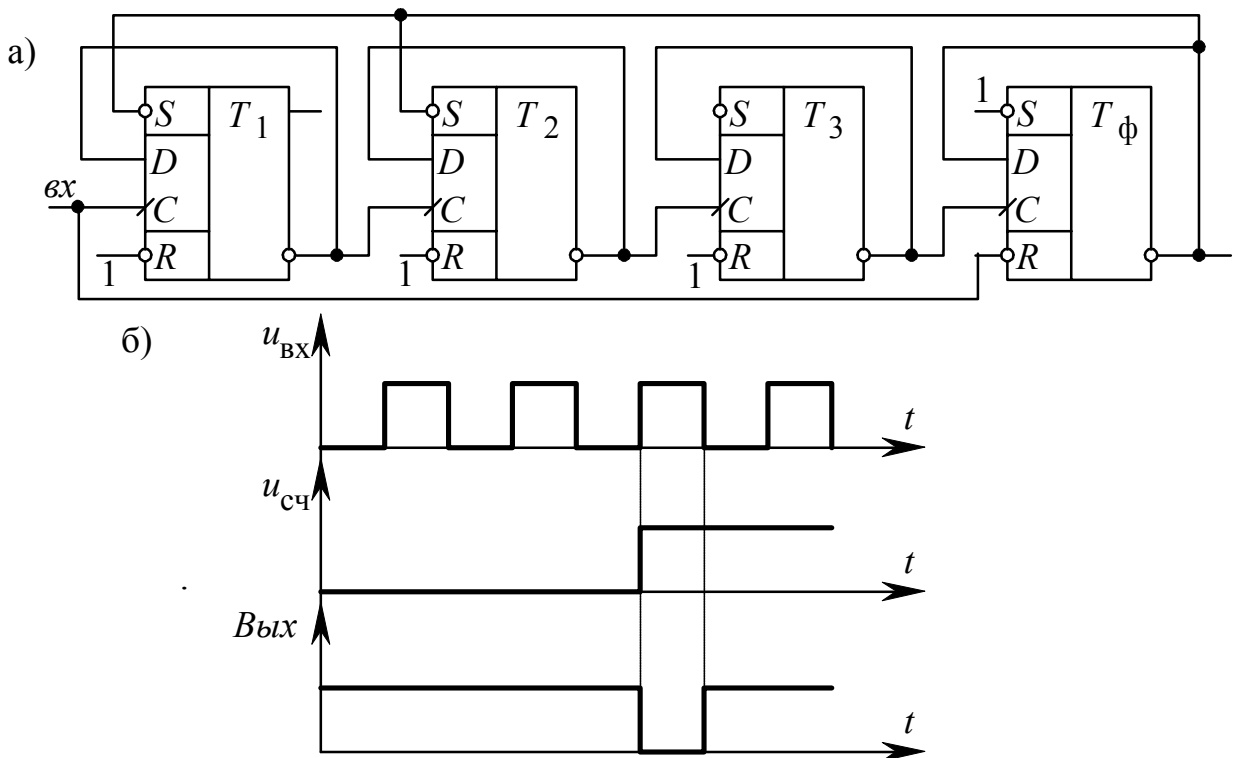


Рис. 2.28. Счётчик с $K = 5$ (а) и временные диаграммы сигналов триггера-формирователя (б)

Четвертый вариант построения счетчиков с произвольным коэффициентом пересчета проиллюстрируем на примере синтеза счетчика с парал-

лельным переносом на D -триггерах. Задаваясь коэффициентом пересчета $K = 5$, определим требуемое минимальное число триггеров двоичного счетчика: $m = 3$ из соотношения $2^m > K > 2^{m-1}$. Выбирая первые пять строк из таблицы переключений суммирующего 3-разрядного счетчика, получим таблицу переключений искомого счетчика (табл.2.41,а). По этой таблице с учетом правил работы D -триггера составим таблицу истинности для сигналов на D -входах триггеров (табл.2.41,б). С помощью карт Карно (табл.2.42) получим структурные формулы для D_i . Имеем:

$$D_1 = \overline{Q_1} \cdot \overline{Q_3}, \quad D_2 = Q_1 \cdot \overline{Q_2} \vee Q_2 \cdot \overline{Q_1}, \quad D_3 = Q_1 \cdot Q_2.$$

Структурная схема счетчика приведена на рис.2.29. Поскольку исключен ряд состояний двоичного счетчика, то необходимо при включении счетчика обеспечить его установку в одно из используемых состояний.

Таблица 2.41

Q^n			n
Q_3^n	Q_2^n	Q_1^n	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
0	0	0	5

Q_3	Q_2	Q_1	D_3	D_2	D_1
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	Φ_1	Φ_2	Φ_3
1	1	0	Φ_4	Φ_5	Φ_6
1	1	1	Φ_7	Φ_8	Φ_9

Таблица 2.42

$Q_2 Q_1$	00	01	11	10	
Q_3	0	1	Φ	Φ	Φ
	0	1	Φ	Φ	Φ
	1	Φ	Φ	Φ	D_1

0	Φ	Φ	1	Φ	
1	Φ	Φ	Φ	Φ	D_3

0	Φ	1	Φ	1	
1	Φ	Φ	Φ	Φ	D_2

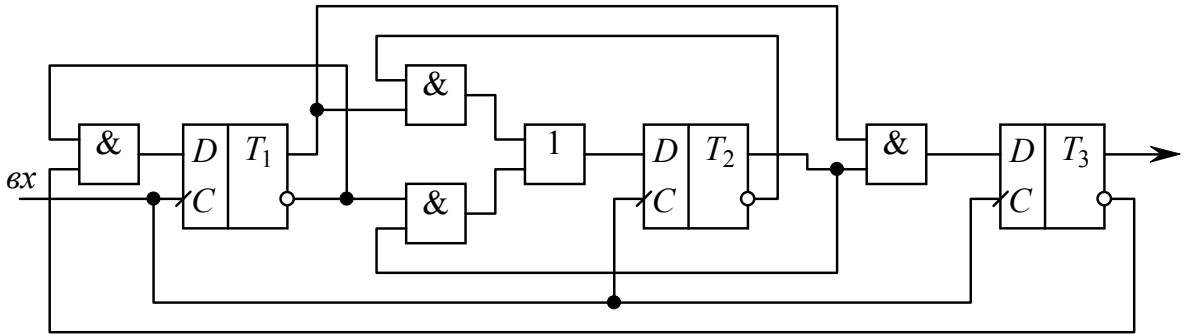


Рис. 2.29. Схема счётчика с использованием обратных связей между разрядами

2.6.3. Счетчики с переменным коэффициентом пересчета

Можно выделить три варианта построения счетчиков с переменным коэффициентом пересчета.

Первый предлагает использование реверсивных счетчиков, имеющих возможность предварительной установки (параллельной загрузки) и выходы сигналов переноса. Примером интегральной схемы такого счетчика является микросхема К155ИЕ7 (рис.2.30,а). Установка этого счетчика в исходное состояние осуществляется с помощью двоичного кода числа N , поступающего на входы $D_0...D_3$ при нулевом уровне на входе управления режимом (Z). Это позволяет устанавливать коэффициент пересчета в пределах от 1 до 16, причем $K = 15 - N$ в режиме суммирования и $K = N + 1$ – в режиме вычитания. Для возобновления записи числа N один из выходов $C_p^+ (\geq 15)$, $C_p^- (\leq 0)$ соединяется с входом управления Z .

При втором варианте построения счетчиков используется микросхема К155ИЕ8 (рис.2.30,б), являющаяся делителем частоты с переменным коэффициентом деления. Структура этой микросхемы представляет собой совокупность 6-разрядного двоичного счетчика и схемы сравнения двоичных кодов: кода, соответствующего состоянию двоичного счетчика, и кода на входах $A_0...A_5$. Микросхема имеет два выхода. На выходе y_2 частота следования импульсов в 64 раза ниже частоты следования входящих импульсов. На выходе y_1 , частота следования импульсов равна частному от деления числа в десятичной системе счисления, соответствующего двоичному числу на входах $A_0...A_5$, на 64. Таким образом, входы $A_0...A_5$ управляют числом импульсов на выходе y_1 во временном интервале между импульсами на выходе y_2 . Недостатком данной микросхемы является неравномерность расположения во времени импульсов последовательности на

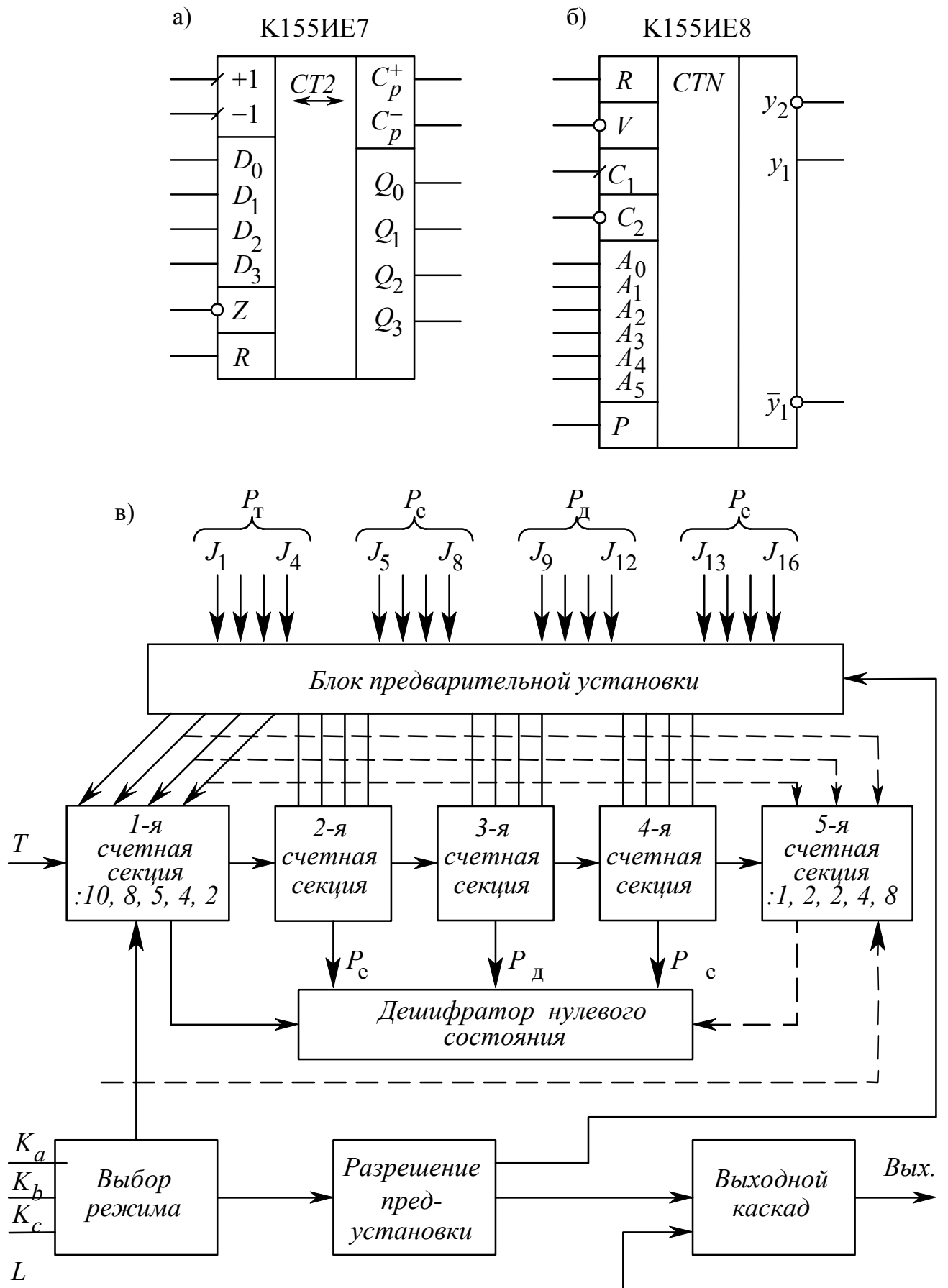


Рис. 2.30. Варианты счётчиков с переменным коэффициентом пересчёта

выходе y_1 . Третий вариант предполагает применение ИС программируемого делителя частоты К564ИЕ15 (CD4059) [2, 7, 24]. Структурная схема ИС (рис.2.30,в) содержит пять вычитающих счетчиков с предварительной установкой, дешифратор нулевого состояния, формирователи сигналов предварительной установки и выходного сигнала, а также ряд схем управления режимом работы. Коэффициент деления микросхемы можно представить в следующем виде:

$$N = M \cdot (1000p_T + 100p_C + 10p_D + p_E) + p_O,$$

где $M = 2; 4; 5; 8; 10$ – модуль числа; p_T – множитель тысяч; p_C, p_D, p_E – множители сотен, десятков, единиц; p_O – остаток. Коэффициент деления может быть целым числом в пределах от 3 до 21327.

2.6.4. Счетчики на основе регистров сдвига

При замыкании регистра сдвига в кольцо путем соединения выхода последнего триггера со входом первого (рис.2.31,а) и предварительной за

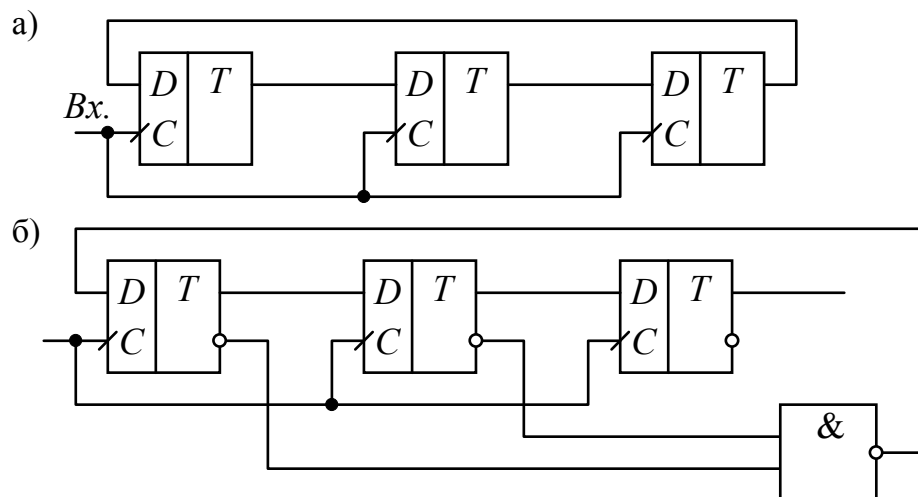


Рис. 2.31. Кольцевой счётчик с $K = 3$ (а), кольцевой счётчик со схемой защиты от сбоев (б)

писи единицы только в один (например, первый) триггер получим, что при сдвиге сигнал 1 будет появляться в каждом такте на выходе только одного триггера (разряда) регистра.

Если рассматривать такое устройство, как счетчик с таблицей переключений, записанной в виде табл.2.43, получим кольцевой счетчик, имеющий значение $K = 3$ состояния которого идентифицируются значениями унитарного кода. В общем случае при использовании m -разрядного регистра будем иметь значение $K = m$. Наличие унитарных кодов на выходах кольцевого счетчика позволяет использовать его в качестве распределителя импульсов, заменяя им совокупность двоичного счетчика и де-

шифратора. Недостатками схемы кольцевого счетчика являются необходимость его защиты от случайных импульсных помех, приводящих к появлению на выходах триггеров более, чем одной единицы. Один из вариантов схемы счетчика с защитой от таких сбоев показан на рис.2.31,б. Эта схема защиты позволяет после воздействия импульсных помех вернуть счетчик в исходное состояние.

Таблица 2.43.

Q^n			n
Q_3^n	Q_2^n	Q_1^n	
0	0	1	0
0	1	0	1
1	0	0	2
0	0	1	3

Другая разновидность счетчиков на основе регистров сдвига получила название счетчиков Джонсона [7, 10, 18, 24]. При проектировании схемы счетчиков Джонсона необходимо обеспечить, чтобы с приходом каждого импульса переключался только один триггер и переключения каждого триггера в счетчике происходили как можно реже (например, два раза за время работы счетчика). Таблица переключений трехразрядного счетчика Джонсона (табл.2.44) имеет шесть строк и, соответственно, $K = 6 = 2m$, где m – число разрядов счетчика. Составив таблицы истинности сигналов на входах D -триггеров (табл.2.45) и получив для них с помощью карт Карно (табл.2.46) структурные формулы, можно построить схему счетчика (рис.2.32), представляющую собой регистр сдвига D -вход которого соединен с инверсным выходом последнего триггера. При любом значении m структура схемы не изменяется, что позволяет на основе счетчиков Джонсона выпускать счетчики с переменным коэффициентом пересчета (например, ИС К561ИЕ19 [10, 24].)

Таблица 2.44.

Q^n			n
Q_3^n	Q_2^n	Q_1^n	
0	0	0	0
0	0	1	1
0	1	1	2
1	1	1	3
1	1	0	4
1	0	0	5
0	0	0	6

Полиномиальные счетчики строятся на основе регистров сдвига с использованием обратных связей через сумматоры по модулю 2. На рис.2.33,а показана одна из возможных схем такого счетчика, а в табл.2.47,а приведена последовательность смены его состояний. При использовании другой схемы реализации цепи обратной связи (рис.2.33,б) последовательность состояний счетчика (табл.2.47,б) также будет иной.

В общем случае длина и вид генерируемой счетчиком последовательности зависят от числа разрядов регистра сдвига и от того, между какими разрядами организованы обратные связи.

Таблица 2.45.

Q_3	Q_2	Q_1	D_3	D_2	D_1
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	Φ	Φ	Φ
0	1	1	1	1	1
1	0	0	0	0	0
1	0	1	Φ	Φ	Φ
1	1	0	1	0	0
1	1	1	1	1	0

Таблица 2.46.

$Q_2 Q_1$	00	01	11	10	
Q_3 0			1	Φ	$D_3 = Q_2$
1		Φ	1	1	
0		1	1	Φ	$D_2 = Q_1$
1		Φ	1		
0	1	1	1	Φ	$D_1 = \bar{Q}_3$
1		Φ			

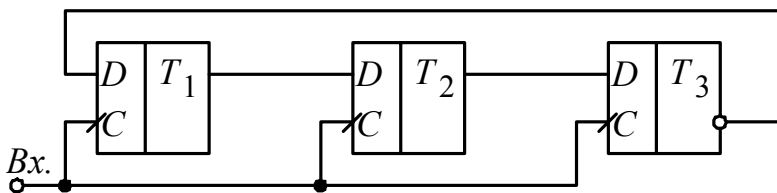


Рис. 2.32. Схема счётчика Джонсона

матором по модулю 2 в цепи обратной связи имеет вид, представленный на рис.2.34. Статистические характеристики последовательностей единиц и нулей, получаемых с выхода любого триггера при большом числе разрядов регистра близки к характеристикам случайной последовательности. В связи с этим такие счетчики используются в качестве генераторов (датчиков) псевдослучайных последовательностей чисел при цифровом моделировании радиотехнических устройств оптимальной обработки сигналов, радиотехнических и телекоммуникационных систем и во многих других случаях, когда наряду со «случайностью» появления чисел требуется иметь повторяемость результатов. Кроме того, полиномиальные счетчики используются для формирования циклических кодов, обеспечивающих обнаружение и исправление ошибок при передаче дискретных сообщений в системах проводной и радиосвязи.

Глава 3. Запоминающие устройства

3.1 Организация запоминающих устройств, их типы и характеристики

Запоминающие устройства (ЗУ) предназначены для записи, хранения и считывания сигналов. На основе ЗУ могут быть построены формирователи логических функций и импульсных последовательностей, преобразователи кодов и дешифраторы, синтезаторы сигналов в радиотехнических системах передачи сообщений, счётчики, устройства коррекции ошибок и диагностики в системах телекоммуникации и т. п. Запоминающее устройство содержит выполненные в одном полупроводниковом кристалле матрицу-накопитель, представляющую собой совокупность элементов памяти (ЭП)¹, и функциональные узлы, необходимые для управления матрицей-накопителем, а также для усиления сигналов при записи, считывании и для обеспечения режима синхронизации.

Функции элемента памяти обычно выполняют триггер (в статических ЗУ), или электрический конденсатор (в динамических ЗУ).

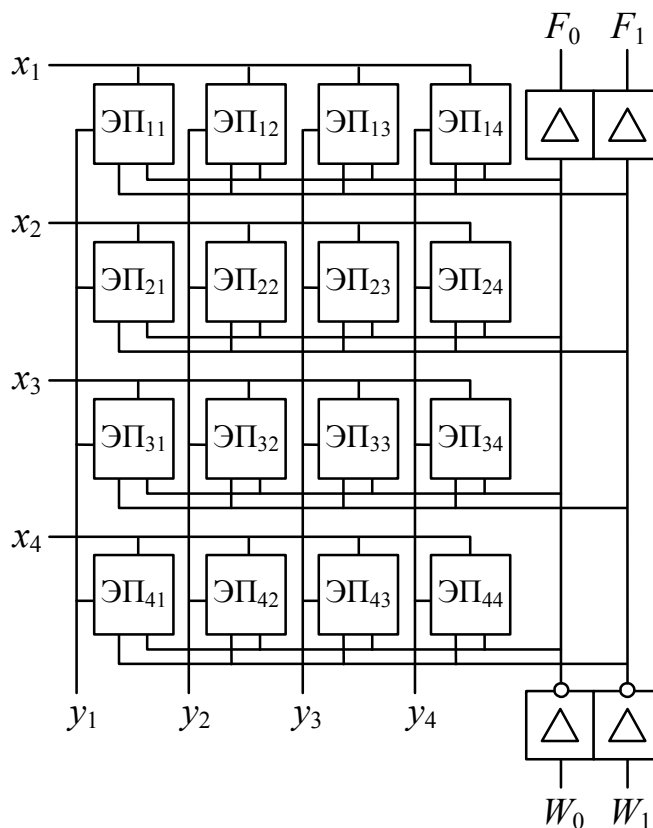


Рис.3.1. Структурная схема запоминающего устройства для $m = n = 4$

¹ Часто используется термин "запоминающая ячейка" (ЗЯ)

Элемент памяти может хранить один разряд двоичного числа (один бит информации). Элементы памяти при двухкоординатной системе адресации расположены на пересечениях m строк и n столбцов матрицы (рис.3.1). Общее число элементов памяти равно произведению $m \times n$. Обращение к требуемому элементу памяти осуществляется сигналами логической 1, которые поступают на те адресные шины строки и столбца, на пересечении которых находится данный элемент памяти. На всех остальных адресных шинах должны быть сигналы логического 0. Формирование сигналов считывания производится дешифратором адреса, который может быть внешним для запоминающего устройства (рис.3.1) или её внутренним функциональным узлом. Запись и считывание производятся по информационным (разрядным) шинам, которые обычно снабжаются усилителями записи и считывания. Часто используются две шины, что обуславливает парафазное представление записываемой информации. Разрядные шины соединены со всеми элементами матрицы-накопителя, но информацией они могут обмениваться только с выбранным элементом памяти. Операции записи и считывания разделяются во времени, поскольку они выполняются с использованием одних и тех же разрядных шин.

Построение запоминающего устройства для N -разрядных чисел осуществляется путём объединения интегральных схем таким образом, чтобы адресные входы ИС соединялись параллельно и подключались к выходам дешифратора. Так, если матрица-накопитель ИС содержит 16 элементов памяти (рис.3.1), то при параллельном соединении N таких ИС запоминающее устройство способно хранить 16 N -разрядных двоичных чисел, т.е. может иметь организацию $16 \times N$ бит. Для адресации любого из 16 двоичных чисел используется 16 различных комбинаций адресных сигналов при четырёхэлементном коде адреса. Например, при коде адреса 0000 появятся сигналы логической 1 на шинах x_1 и y_1 , подключённых к ЭП₁₁ (рис. 3.1). Следовательно, все разряды числа $a_4 a_3 a_2 a_1$, присутствующего на входах W_1 или W_0 , считываются или записываются в элемент памяти.

По назначению запоминающие устройства делят на две группы: оперативные запоминающие устройства (ОЗУ), предназначенные для хранения текущих результатов или данных, полученных в процессе работы, и постоянные запоминающие устройства (ПЗУ), предназначенные для хранения подпрограмм, микропрограмм, констант и т.п., работающие только в режиме многократного считывания.

По способу хранения информации ОЗУ разделяют на статические и динамические. Статические ОЗУ, элементами памяти в которых являются триггеры, способны хранить информацию неограниченное время, при условии, что имеется напряжение питания. Динамические ОЗУ, роль элементов памяти в которых выполняют электрические конденсаторы, для сохранения записанной информации нуждаются в её периодической перезаписи (регенерации).

Основными функциональными характеристиками запоминающих устройств являются: информационная ёмкость, разрядность, быстродействие, потребляемая мощность.

Информационная ёмкость определяется числом одновременно хранящихся в матрице-накопителе единиц информации.

Разрядность задаётся количеством разрядов в запоминаемом числе. Наибольшее распространение получила одноразрядная организация интегральных схем запоминающих устройств, при которой ИС обеспечивает одновременное хранение $m \times n$ одноразрядных чисел. Например, микросхема К155РУ1 (рис.3.1) имеет информационную ёмкость 16 бит, разрядность 1 и организацию матрицы-накопителя 16×1 бит.

Быстродействие количественно характеризуется несколькими временными параметрами, среди которых можно выделить в качестве обобщающего параметра время цикла записи (считывания) информации.

В статических ОЗУ время цикла считывания практически равно времени выборки адреса, которое определяется задержкой выходного сигнала относительно момента поступления кода адреса. В динамических ОЗУ время цикла считывания больше времени выборки адреса, так как после завершения считывания необходимо некоторое время на восстановление (регенерацию) исходных уровней выходных сигналов элементов памяти микросхем. В систему временных параметров входят длительность управляющих сигналов, их взаимный временной сдвиг, период повторения и длительность сигналов регенерации.

Потребляемая интегральной схемой памяти мощность обычно указывается, исходя из расчёта на 1 бит. Для ИС, у которых имеется существенное различие потребляемой мощности для разных режимов, приводятся значения этого параметра при режиме хранения и режиме обращения.

Быстродействие, потребляемая мощность, уровень интеграции и другие показатели запоминающих устройств в значительной степени зависят от технологии изготовления микросхемы. Микросхемы памяти выпускают

либо в составе широко известных серий общего промышленного и массового применения, например, серий К155, К500, К564 и др., либо отдельными сериями, например, К558, К565, К573, К1556, К1604 и др.

3.2. Оперативные запоминающие устройства

Интегральные схемы статических ОЗУ имеют, как правило, матричную структуру с двухкоординатной системой адресации (выборки). В быстродействующих ОЗУ используются биполярные транзисторные элементы ЭСЛ, ТТЛ (ТТЛШ), ИИЛ, а в ОЗУ среднего и низкого быстродействия p -МДП, n -МДП и КМДП транзисторные элементы.

На рис.3.2,а приведена схема элемента памяти на биполярных транзисторах. Основу составляет RS -триггер (T_1, T_2). По адресным шинам x_i, y_j , с которыми соединены эмиттеры 2–5 транзисторов T_1 и T_2 , поступают сигналы, определяющие режим работы элемента памяти: запись в триггер, считывание с его выходов или хранение информации.

Режим хранения обеспечивается при поступлении сигналов нулевого уровня на обе адресные шины или на одну из них. Разрядные шины соединены с эмиттерами 1 и 6 транзисторов T_1 и T_2 . Входные сигналы подаются через усилители записи и изменяют состояние транзисторов T_1 и T_2 только при условии, что оба адресных сигнала равны единичному уровню. Например, при записи логической 1 имеем: $W_1 = 1, W_0 = 0$. Поскольку усилители записи имеют инверсный выход, то на единичной разрядной шине будет уровень логического 0, а на нулевой – уровень логической 1.

Этими сигналами транзистор T_1 закрывается, а T_2 открывается. При записи нулевого уровня состояния транзисторов изменяются на обратные. В режиме считывания имеем: $W_1 = W_0 = 0$. На разрядных шинах устанавливаются уровни логической 1, чтобы сопротивления выходов усилителей записи не шунтировали входов усилителей считывания.

При выборке элемента памяти входы 2–5 транзисторов T_1 и T_2 закрываются и ток через транзистор T_2 , ранее протекавший в адресные шины, поступит в разрядную шину через эмиттерный переход 6 транзистора T_2 . Заметим, что переход 6 транзистора T_2 остаётся открытым при единичном уровне на разрядной шине, благодаря превышению напряжения на коллекторе T_2 над напряжением уровня логической 1 разрядной шины. В результате усилитель считывания формирует сигнал уровня логической 1, а на выходе другого усилителя в это время будет формироваться сигнал уровня логического нуля.

Пример принципиальной схемы элемента памяти на КМДП транзи-

сторях приведён на рис.3.2,б. Основу элемента памяти составляет триггер на двух комплементарных парах T_1, T_3 и T_2, T_4 . Транзистор T_5 выполняет функции ключа, управляемого сигналом на адресной шине строки x_i . Он соединяет триггер с j -й разрядной шиной (РШ $_j$), которая совмещает функции информационной и адресной шин столбца.

Выборка строки производится сигналом уровня логической 1 на адресной шине x_i . При этом открывается транзистор T_5 . В результате сигнал с разрядной шины поступает на вход пары транзисторов T_2, T_4 . Например, при записи уровня логической 1 транзистор T_2 открывается, а транзистор T_4 закрывается. С выхода транзистора T_2 напряжением низкого уровня (ниже порогового значения) транзистор T_1 переводится в закрытое, а транзистор T_3 – в открытое состояние.

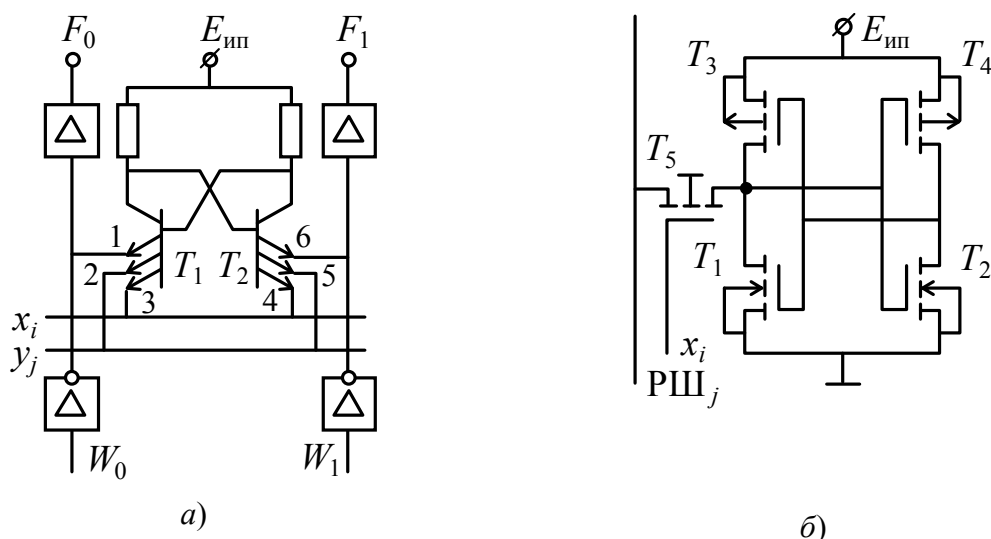


Рис. 3.2. Схема элемента памяти на биполярных транзисторах (а) и на КМДП-транзисторах (б)

Режим хранения информации обеспечивается при уровне логического 0 на адресной шине строки. При этом транзистор T_5 закрывается, что приводит к изоляции триггера от разрядной шины. При считывании в адресную шину x_i подаётся сигнал уровня логической 1. Тогда транзистор T_5 открывается и в разрядную шину поступает ток через открытый транзистор T_3 . Если в элементе памяти записано значение логического 0, транзистор T_3 закрыт, а транзистор T_1 открыт. Поэтому при обращении к элементу памяти ток в разрядную шину поступать не будет.

В качестве примера на рис.3.3 показана структурная схема статического оперативного запоминающего устройства (ОЗУ) К564РУ2, матрица-

накопитель которого состоит из 16×16 КМДП элементов памяти. Организация матрицы-накопителя позволяет хранить информацию объемом 256×1 бит. Для работы интегральной схемы необходимо к входам дешифраторов строк и столбцов подвести 8-разрядный двоичный код адреса, а также сигнал "выбор микросхемы" (BM), разрешающий обращение к матрице-накопителю по адресным и информационным входам а также вывод информации.

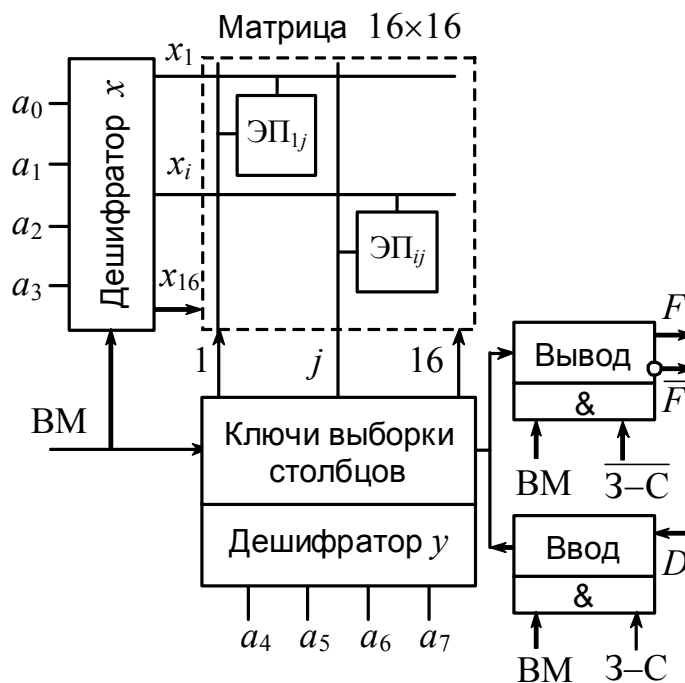


Рис. 3.3. Структурная схема статического ОЗУ K564PY2

При запрещающем значении сигнала «выбор микросхемы» матрица-накопитель изолирована от выходов дешифратора строк и шины ввода-вывода. Электронные ключи выборки столбцов (рис.3.3) управляются сигналами с выходов дешифратора y и предназначены для коммутации цепи между выбранными элементами памяти и шиной ввода-вывода. Режим работы микросхемы устанавливается сигналом "запись – считывание" (З-С).

При уровне логической 1 сигнала «запись-считывание» и наличии разрешающего сигнала «выбор микросхемы» открывается схема ввода и сигналы с входа через шину ввода-вывода и открытый электронный ключ выборки столбца поступают в выбранный элемент памяти. При считывании сигнал «запись-считывание» З-С имеет уровень логического 0, при котором открывается схема вывода информации и на выходе F появляются

ся последовательности информационных символов. Выходная цепь может принимать одно из трёх состояний: открытое ($F = 0$), закрытое ($F = 1$) и высокоомное (сигнал «выбор микросхемы» равен логическому 0), при котором выход отключается от внешней шины.

3.3 Постоянные запоминающие устройства

Интегральная схема ПЗУ содержит матрицу-накопитель, регистр и дешифратор адреса, усилители считывания. По способу записи информации ПЗУ подразделяются на масочные, программируемые (ППЗУ) и репрограммируемые (РПЗУ).

Масочные ПЗУ выполняются на биполярных или полевых транзисторах. Запись информации в ПЗУ осуществляется однократно на одной из завершающих технологических операций изготовления интегральной схемы путём кодирования схемы подключений транзисторов к шине строки (рис.3.4). Организация ПЗУ может быть как одноразрядной, так и много-разрядной. На рис.3.4 показана структура ПЗУ с организацией $m \times n$ бит. При кодировании принято следующее условие: логическому нулю соответствует наличие соединения базы транзистора с шиной строки, логической единице – отсутствие такого соединения.

При выборе строки открываются транзисторы, соединённые с адресной шиной, и на соответствующих им разрядных шинах фиксируется уровень логического 0. На остальных шинах будет уровень логической 1. Для сигнала разрешения считывания предусматривается вход "Выборка микросхемы" ВМ. Аналогичным образом строятся масочные ПЗУ на МДП транзисторах.

Программируемые ПЗУ в отличие от масочных ПЗУ позволяют записывать, но также однократно, нужную информацию самому пользователю. Для этого с помощью программатора устраняются (например, путем пережигания плавких перемычек) связи между адресными и разрядными шинами матрицы-накопителя.

Один из вариантов ППЗУ на основе многоэмиттерных транзисторов показан на рис.3.5. При выборке адресной шины на базу транзистора строки поступает сигнал логической 1. Транзистор открывается, и на разрядных шинах формируются уровни напряжения, соответствующие схеме соединения с этими шинами эмиттеров данного транзистора. Если эмиттер соединён с шиной, то в эту шину поступит ток от источника коллекторного напряжения. Когда перемычка разрушена, то тока в шине не будет. Различные состояния разрядных шин с помощью выходных усилителей пре-

образуются в код числа.

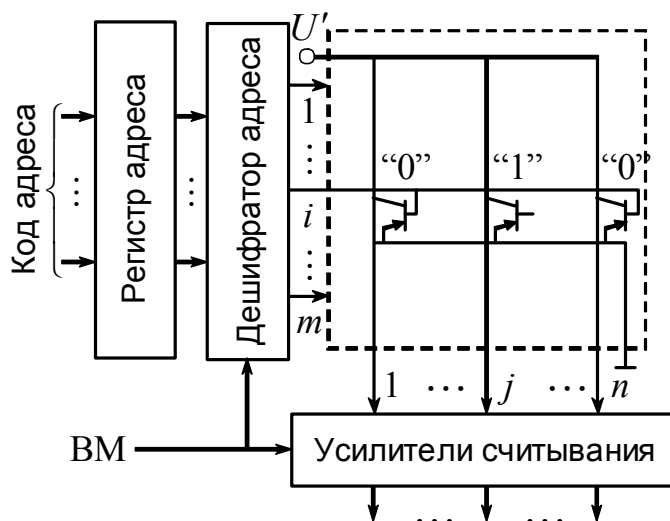


Рис. 3.4. Структурная схема масочного ПЗУ

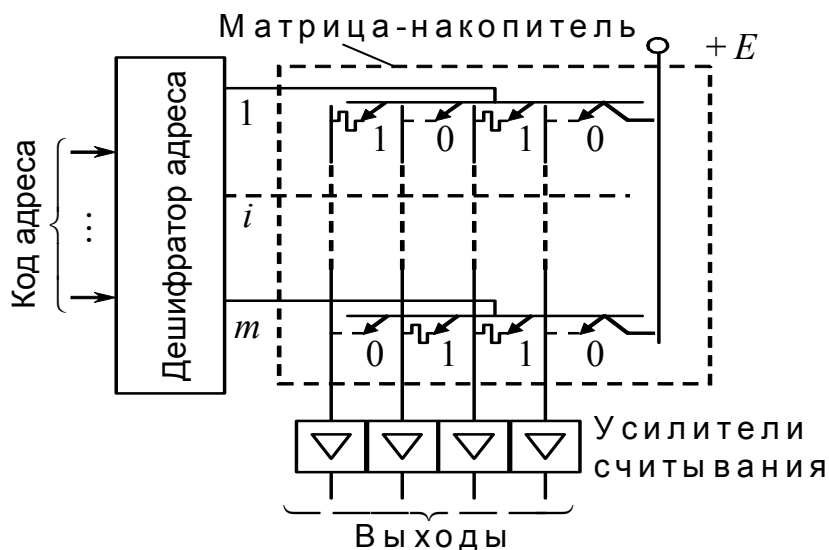


Рис. 3.5. Структура ПЗУ с плавкими перемычками

Репрограммируемые ПЗУ обычно строятся на основе структур МНОП (металл – нитрид кремния – окисел кремния – полупроводник) или МДП с плавающим затвором. Структура МНОП (рис.3.6,а) представляет собой МДП-транзистор с двухслойным диэлектриком под затвором. Здесь И – исток, С – сток, З – затвор и П – подложка транзистора. Нижний, при-
 мыкающий к полупроводнику, слой двуокиси кремния толщиной 3–4 нм "прозрачен" для электронов. Если к затвору относительно подложки при-
 ложить импульс напряжения положительной полярности, то под действием
 сильного электрического поля между затвором и подложкой электроны

приобретают достаточную энергию, чтобы пройти тонкий диэлектрический слой до границы раздела двух диэлектриков. Верхний слой нитрида кремния имеет значительную толщину, так что электроны не могут его преодолеть. Накопленный на границе раздела двух диэлектрических слоёв заряд электронов снижает пороговое напряжение и смещает передаточную характеристику транзистора, т.е. зависимость тока стока (I_c), от напряжения на затворе (U_z), влево (рис.3.6,б). Таким образом записывается уровень логической единицы.

Логическому нулю соответствует состояние транзистора без заряда электронов в диэлектрике. Чтобы обеспечить это состояние, на затвор подаётся импульс напряжения отрицательной полярности. При этом электроны вытесняются в подложку. При отсутствии заряда электронов под затвором передаточная характеристика смещается в область высоких пороговых напряжений. Для считывания записанной информации на затвор необходимо подать напряжение, значение которого лежит между двумя пороговыми уровнями, соответствующими уровням логических 0 и 1. Тогда при записанном уровне логической единицы транзистор откроется, а при уровне логического нуля останется в закрытом состоянии. Число циклов перепрограммирования составляет несколько тысяч. Перепрограммирование осуществляется импульсами напряжения амплитудой 30...40 В, что обуславливает высокие требования к электрической прочности структуры МДП-транзистора, диэлектрических слоёв.

Другое направление создания РПЗУ основано на использовании свойств МДП структур с плавающим затвором (рис.3.7,а,б). Особенность устройства такого элемента памяти заключается в том, что затвор формируется внутри диэлектрика и не имеет наружного вывода (рис. 3.7,б, где С – сток, И – исток транзистора). Затвор отделён от подложки тонким, прозрачным для электронов, слоем диэлектрика. Для записи логической 1 между истоком или стоком и подложкой прикладывается обратное напряжение, достаточное для создания условий лавинного размножения электронов в электронно-дырочном переходе.

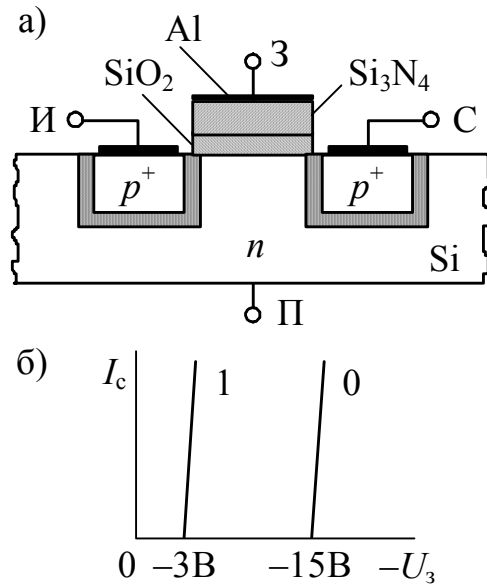


Рис. 3.6. Структура МДП-транзистора с двухслойным диэлектриком под затвором (а) и его передаточные характеристики (б)

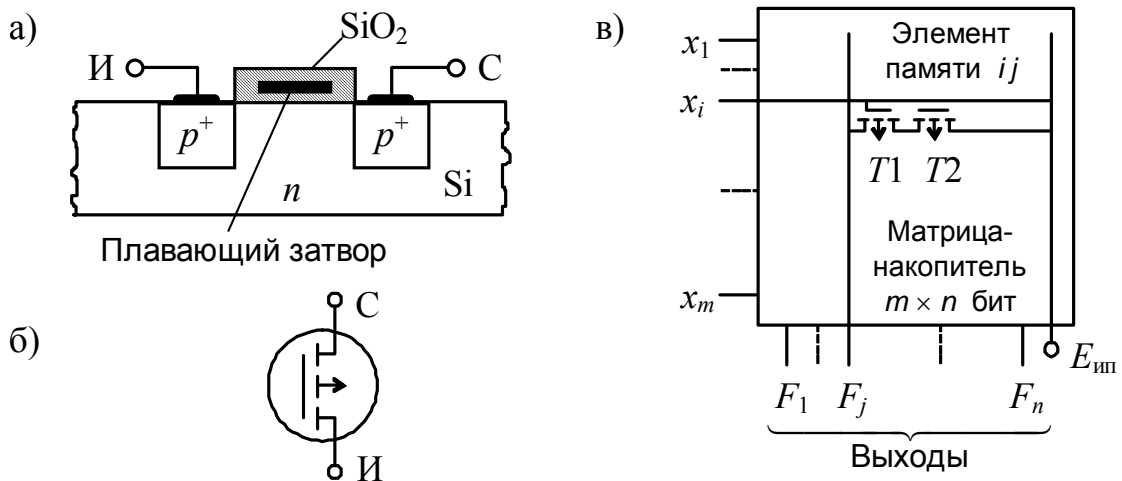


Рис. 3.7. Структура МДП-транзистора с плавающим затвором (а), его обозначение (б) и матрица-накопитель, использующая такие транзисторы

Эти электроны, имея большую кинетическую энергию, попадают на затвор, накапливаются на нём и создают потенциал, достаточный для наведения канала. Если на затворе заряд отсутствует, то канал не формируется. Это состояние транзистора соответствует уровню логического нуля.

В состав матрицы-накопителя МДП транзистор с плавающим затвором включают в паре с обычным МДП транзистором $T2$ (рис.3.7,в). Очевидно, что при проводящем состоянии транзистора $T2$, когда записан уровень единицы, через транзисторы $T1$ и $T2$ в выходную шину будет поступать ток считывания. Если же записан уровень логического нуля, транзи-

стор T_2 закрыт и тока в выходной шине не будет. Стирание информации в РПЗУ такого типа производится ультрафиолетовым облучением кристалла микросхемы через окно в крышке корпуса. Количество циклов перепрограммирования около 100. Репрограммируемые ПЗУ способны сохранять заряд при отключённом питании в течение 2–3 тысяч часов (некоторые современные ИС до 10 лет).

На основе объединения двух предыдущих структур РПЗУ возможно построение интегральных схем энергонезависимой памяти запоминающих устройств (flash). При создании таких ИС используется технология ЕТОХ III (Eprom Tunnel OXide), позволяющая реализовать программирование тем же способом, что и в ППЗУ с ультрафиолетовым стиранием, т. е. используя плавающий затвор и инжекцию "горячих" электронов, а стирание информации как у МНОП-структур с помощью двухслойного затвора и использования туннельного эффекта прохождения заряда носителей через диэлектрик.

На рис.3.8 показана структура одной подобной ячейки ЗУ и её подключение к источникам внешнего напряжения соответственно в режимах записи и стирания информации (на рис.3.8,а ,б соответственно, где $U_{\text{прг}}$ - напряжение программирования, U_c - напряжение на стоке, U_d - напряжение на истоке.). Как видно из рисунка, выбирающий затвор изолирован от плавающего затвора двухслойной структурой из оксида и нитрида кремния, а плавающий затвор изолирован от подложки очень тонким слоем (~100) окисла.

В состав интегральной схемы входит устройство управления записью и стиранием, осуществляющее коммутацию источников напряжения. Выпускаемые различными фирмами (Intel, Atmel и др.) микросхемы памяти, как правило, используют одно напряжение и для записи и для стирания, причём величина его снижена до 2,7 В для уменьшения энергопотребления. Времена программирования и стирания для запоминающего устройства ёмкостью 8 Мбайт (1 Мбит × 8) составляют 1...2 с (у запоминающих устройств с ультрафиолетовым стиранием время стирания информации не менее 15 минут). Время считывания равно 65...150 нс в зависимости от объёма памяти. Число циклов перепрограммирования до 100 000.

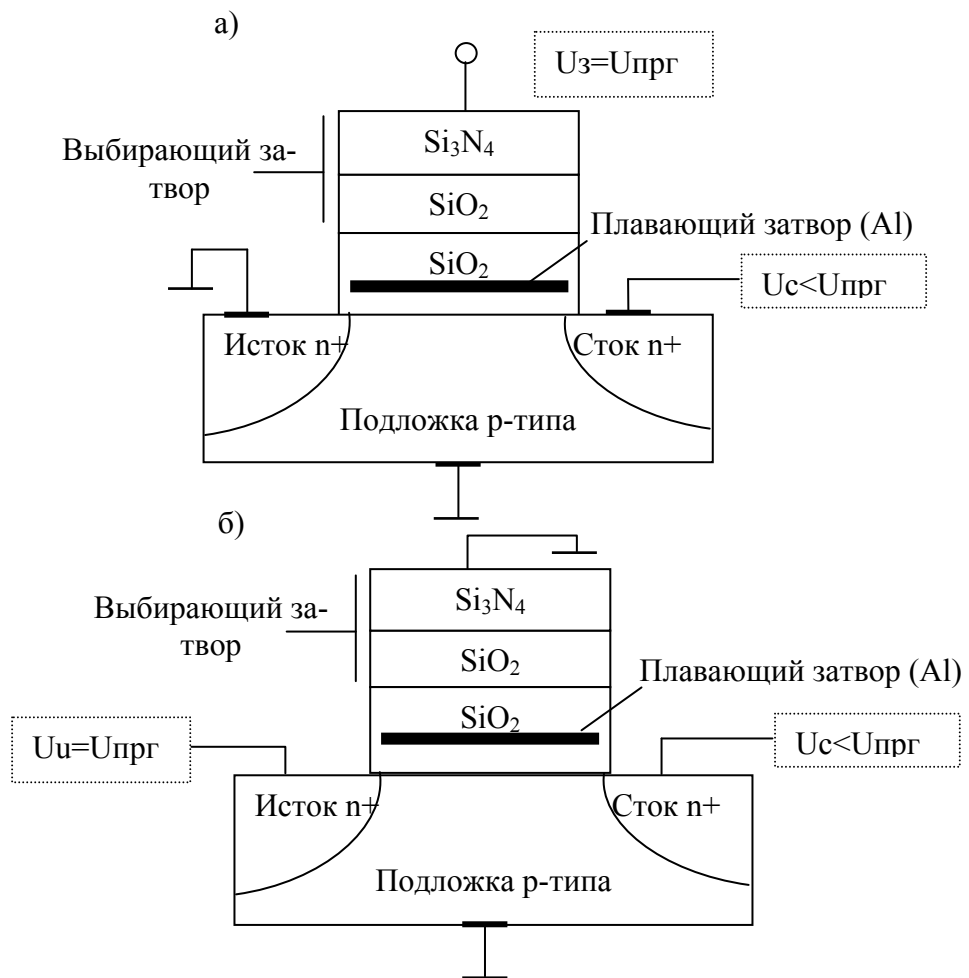


Рис. 3.8. Ячейка памяти (flash) при программировании (а) и при стирании (б) (поперечное сечение)

3.4 Программируемые логические матрицы

Структурная схема программируемой логической матрицы (ПЛМ) представляет собой последовательное соединение программируемой матрицы $M_{и}$ логических элементов И и матрицы $M_{или}$ логических элементов ИЛИ и буферные устройства на входах и выходах (БВх и БВых – рис.3.9,а). Внутренняя структура обеих матриц представлена на рис.3.9,б. Входные переменные x_1, x_2, \dots, x_6 и их инверсии $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_6$ поступают в горизонтальные шины матрицы M_1 , а вертикальные шины формируют логические произведения (конъюнкции) их. В узлах матрицы (точках пересечений горизонтальных и вертикальных шин) включены элементы связи (диоды, МОП или биполярные транзисторы), обозначенные на рисунке точками, посредством которых при программировании вводятся (или устраняются) связи горизонтальных и вертикальных шин. Таким образом могут формироваться любые требуемые логические произведения

(конъюнкции) входных переменных. Имеющиеся в узлах матрицы M_2 элементы позволяют формировать на выходных цепях (горизонтальных шинах M_2) любые требуемые логические суммы (дизъюнкции) функций, полученных на выходах матрицы M_1 . Программируя связи на отмеченных крестиками участках (рис.3.9,б), имеем возможность формирования прямых и инверсных форм выходных функций. Программируемая логическая матрица позволяет подать на элементы И матрицы M_1 комбинацию входных сигналов в прямой и инверсной формах, подключить к заданному элементу ИЛИ матрицы M_2 комбинацию сигналов на выходах элементов И, представить в инверсной форме сигналы на выходах элементов ИЛИ [6, 14, 17, 23].

Принцип реализации операции логического умножения (конъюнкции или операции И) и логического сложения (ИЛИ) с помощью МОП транзисторов, включённых в узлы матриц M_1 и M_2 , показан на рис.3.10.

При наличии во входной цепи логической 1 соответствующий транзистор оказывается открытым и в выходной цепи устанавливается значение логического нуля. Следовательно, для того чтобы в выходной цепи мог быть потенциал, соответствующий логической 1, все транзисторы, связанные с этой цепью, должны быть закрыты напряжением логического 0. Например, транзисторы на рис.3.10,а окажутся закрытыми при $x_1 = 0$ и $\bar{x}_2 = 0$. При этом в цепи z формируется логическое произведение $\bar{x}_1 \cdot x_2$.

Пусть в матрице M_2 включены транзисторы в узлах, образованных цепями z_1, z_3, z_6 с выходной цепью y_1 (рис.3.10,б). В данном случае на входе инвертора образуется логическое произведение $\bar{z}_1 \cdot \bar{z}_3 \cdot \bar{z}_6$, а на выходе – логическая сумма $y_1 = z_1 + z_3 + z_6$.

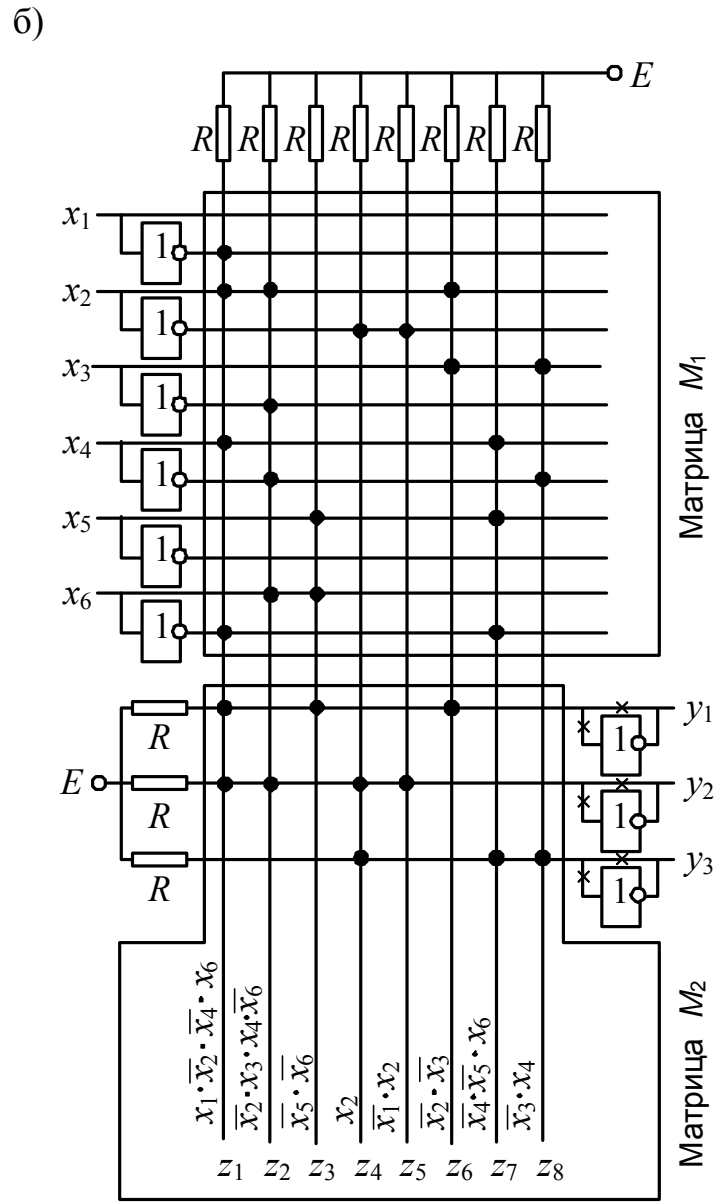
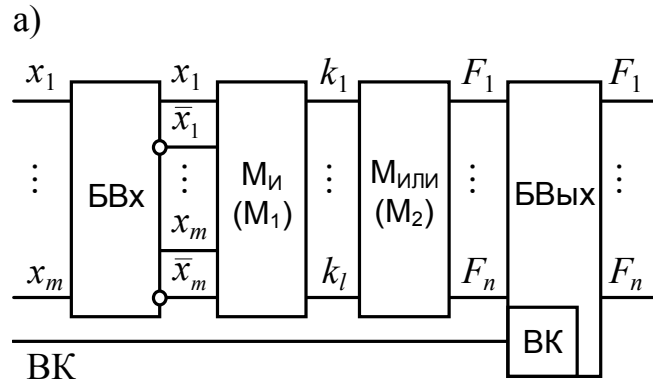


Рис. 3.9. Структурная схема программируемой логической матрицы (а) и вариант ее реализации (б)

Если транзисторы связи включены в местах, показанных на рис.3.9,б, точками, ПЛМ реализует логические функции следующего вида:

$$y_1 = z_1 + z_3 + z_6 = x_1 \cdot \bar{x}_2 \cdot \bar{x}_4 \cdot x_6 + \bar{x}_5 \cdot \bar{x}_6 + \bar{x}_2 \cdot \bar{x}_3,$$

$$y_2 = z_1 + z_2 + z_4 + z_5 = x_1 \cdot \bar{x}_2 \cdot \bar{x}_4 \cdot x_6 + \bar{x}_2 \cdot x_3 \cdot x_4 \cdot \bar{x}_6 + x_2 + \bar{x}_1 \cdot x_2,$$

$$y_3 = z_4 + z_7 + z_8 = x_2 + \bar{x}_4 \cdot \bar{x}_5 \cdot x_6 + \bar{x}_3 \cdot x_4.$$

В интегральной схеме ПЛМ (рис. 3.11,а, где Е – вход разрешения считывания) предусмотрена возможность размещения элементов связи в любых узлах M_1 и M_2 (рис.3.11,б). При программировании ненужные связи устраняются выплавлением перемычек. Возможна реализация требуемых связей с помощью маски на завершающих операциях изготовления ИС. В ПЛМ с МОП транзисторами при этом ненужные элементы связи остаются незавершёнными (затворы транзисторов не имеют выводов).

Сопоставление ППЗУ и ПЛМ показывает, что программирование в ПЛМ матрицы M_1 даёт возможность увеличить число входов без увеличения числа выходов и, соответственно, на той же площади кристалла расположить устройство, реализующее функции от большего числа переменных. В ПЛМ удаётся избежать свойственной ППЗУ большой избыточности, поскольку число выходов дешифратора в ППЗУ равно 2^m , где m – число адресных входов. Это связано с тем, что большинство встречающихся на практике логических функций содержит лишь малую долю от максимально возможного числа логических произведений входных переменных (термов).

Основными параметрами ПЛМ являются число входов m (8...16), число цепей конъюнкции l (24...96) и число выходов n (4...16).

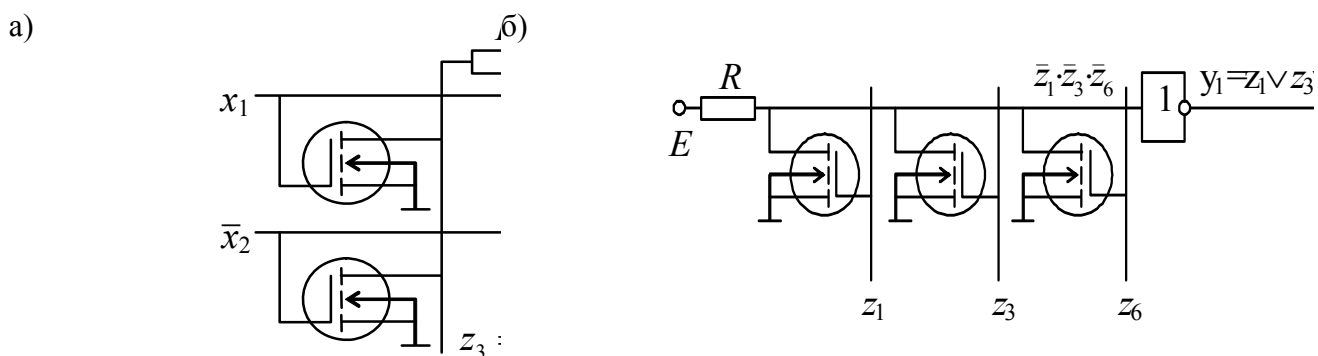


Рис. 3.10. Реализация операции И (а), ИЛИ (б) в программируемой логической матрице

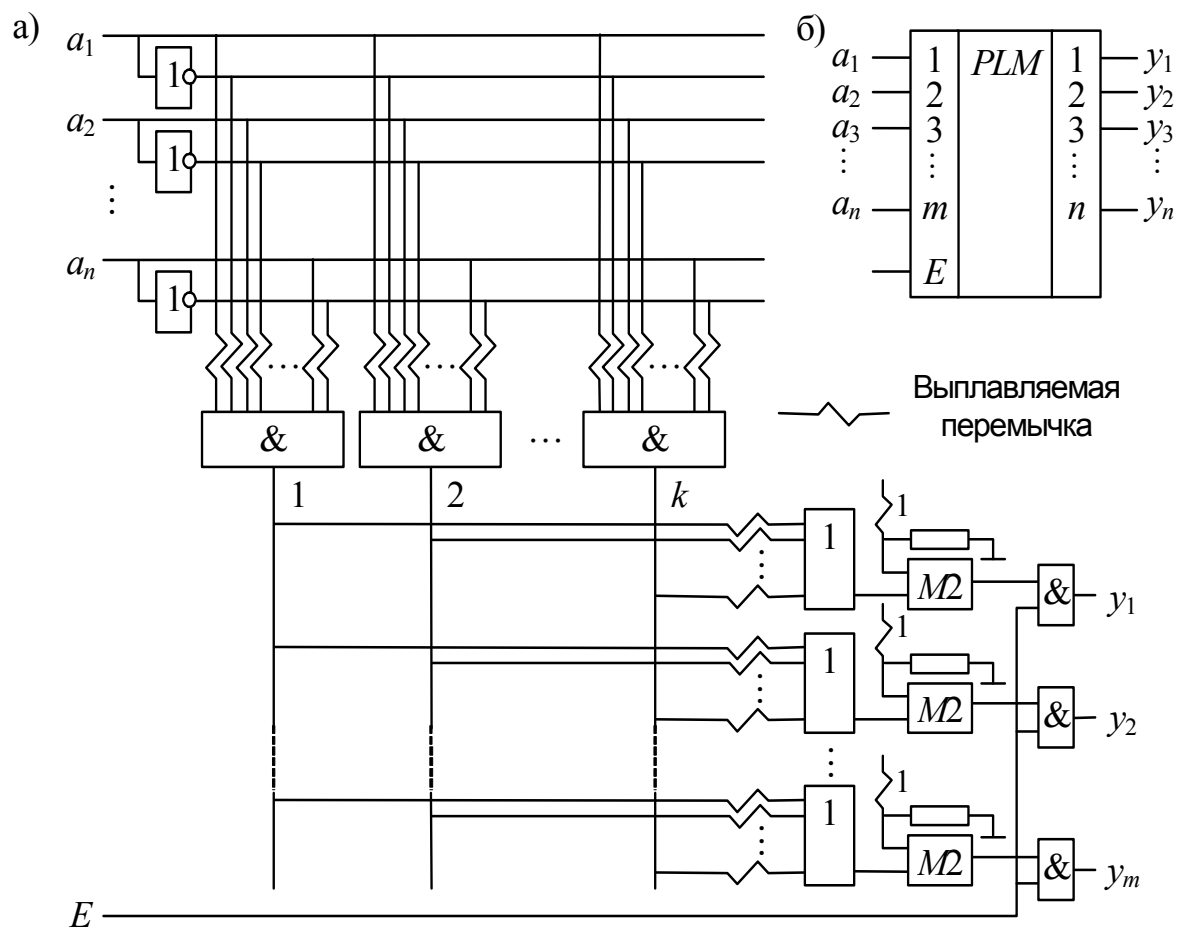


Рис. 3.11. Структура программируемой логической матрицы (а) и ее условное обозначение при использовании плавких перемычек (б)

Используя в программируемой логической матрице обратные связи с выходов на входы (рис.3.12), можно реализовать скобочную форму записи логических функций. Так, схема, представленная на рис.3.12, реализует функцию вида:

$$F = x_1 \cdot x_2 + \overline{(x_1 \cdot x_2 + \bar{x}_1 \cdot \bar{x}_2)} \cdot x_3.$$

Устройства на программируемых логических матрицах обладают существенными преимуществами перед микропроцессорами при реализации алгоритмов управления, когда необходимо обеспечить высокое быстродействие, но не требуется помнить большой объём промежуточной информации. На программируемых логических матрицах строятся радиотехнические устройства управления протоколами телекоммуникационных сетей, устройства кодирования и декодирования цифровых сообщений, скремблеры и узлы синхронизации модемов в спутниковых бортовых и наземных устройствах обработки сигналов.

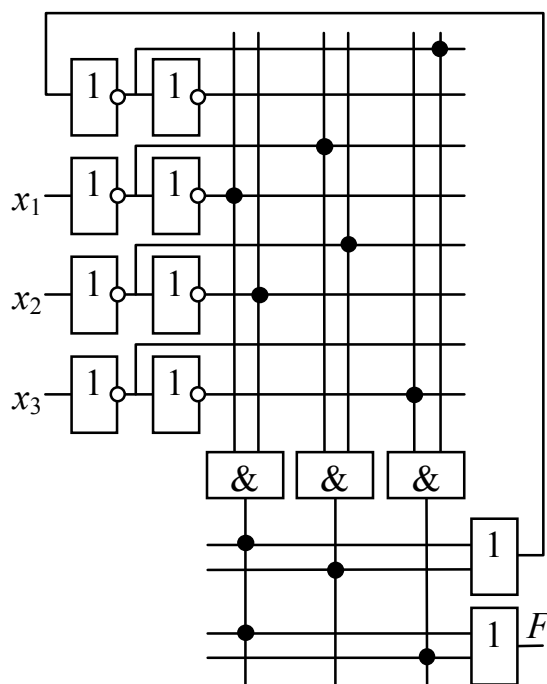


Рис.3.12. Фрагмент структуры программируемой логической матрицы с возможностью реализации обратных связей

3.5 Базовые матричные кристаллы

Базовый матричный кристалл (БМК) представляет собой совокупность регулярно расположенных на кристалле топологических фрагментов (ячеек), между которыми имеются свободные зоны для создания межэлементных соединений [4, 6, 17, 23]. Такой БМК представляет собой матрицу независимых схем, называемых ячейками, электрические связи между которыми формируются в соответствии с назначением микросхем на технологическом этапе реализации устройства. В состав ячеек входят группы элементов (транзисторов, резисторов, конденсаторов) и частей полупроводниковых шин для реализации пересекающихся электрических связей [23]. Из элементов ячеек с помощью электрических связей в виде металлических и полупроводниковых шин формируются различные функциональные узлы (логические элементы, триггеры, регистры, счётчики, буферные элементы) и соединения между ними. Применяются два способа организации ячеек матрицы БМК.

На основе элементов ячейки создается базовый логический элемент, реализующий одну из основных логических операций. Для воспроизведения логических функций производят соединения между собой нескольких ячеек матрицы. Такие БМК называют вентиляльными матрицами [23].

Из элементов ячейки может быть сформирован любой функциональный элемент. Набор реализуемых функциональных элементов составляет библиотеку функциональных элементов. В состав библиотеки могут входить как простые логические элементы, так и достаточно сложные узлы, например, триггеры, регистры, счётчики, дешифраторы, мультиплексоры, усилители и др. При разработке функционального элемента используются компоненты одной или нескольких соседних ячеек.

Базовые матричные кристаллы можно условно разделить на четыре группы матриц: КМДП цифровые матрицы; биполярные цифровые матрицы; аналого-цифровые; аналоговые. Особое место принадлежит КМДП цифровым БМК, сочетающим наибольшую степень интеграции и низкий уровень потребления энергии. Кроме того, благодаря уменьшению линейных размеров КМДП транзисторов такие БМК в ряде случаев достигают быстродействия ЭСЛ схем.

Проектирование БИС на основе БМК предусматривает использование библиотеки реализуемых на нём схем, т.е. библиотеки готовых решений по созданию функциональных элементов, узлов и их частей на основе данного БМК.

Дальнейшее развитие БМК – это направление, связанное с созданием БИС с программируемой архитектурой и реконструируемыми соединениями частей системы [17, 23]. Эта разновидность БМК представляет собой совокупность subsystemов, выполняемых на пластинах. Перестройка архитектуры subsystemы осуществляется с помощью встроенных элементов коммутации.

В качестве примера построения БИС на БМК рассмотрим таймер H1515XM1–109, содержащий счётчик с коэффициентом счёта 2^{24} , на выходах которого могут формироваться либо 24 последовательности сигналов прямоугольной формы, следующих со скважностью 2, либо 24 заданных импульсных последовательности. Здесь же имеются восемь вспомогательных счётчиков СТ1, СТ2, сопряжённых с дешифраторами, позволяющих сформировать восемь сложных двоичных сигналов, включая сигналы дискретной линейной частотной модуляции. Основные узлы дешифраторов и счётчиков этой БИС могут быть использованы в синтезаторах частот и в цифровых демодуляторах сигналов [20, 23].

На базе БИС H1515XM1–005 разработан свёрточный кодек, обеспечивающий скорости передачи 5, 17, 25 и 34 Мбит/с. Заметим, что при скорости 25 Мбит/с появляется возможность передачи в одном стволе цифро-

вой спутниковой системы связи двух высококачественных программ телевидения, либо совмещения передачи телевизионных программ и многоканальной телефонии путём временного уплотнения каналов.

Дальнейшее совершенствование технологии изготовления БИС позволит размещать на кристалле БИС ёмкостью десятки и сотни тысяч условных вентилях с временем задержки вентиля менее 1 нс, которые дадут возможность создать аппаратуру, осуществляющую передачу высококачественного цифрового телевидения на скорости до 12 Мбит/с. Такие кристаллы позволят создать высокоскоростные помехоустойчивые цифровые модемы с малыми энергетическими потерями и высококачественные кодеки цифрового радиовещания для скоростей передачи 2,4; 4,8; 9,6; 16 и 32 Кбит/с.

3.6 Программируемые логические интегральные схемы

Дальнейшее развитие идей программируемых логических матриц и базовых матричных кристаллов воплотилось в создании программируемых логических интегральных схем (ПЛИС). Программируемые логические интегральные схемы – это БИС, содержащие многосекционные или многоблочные матрицы (multiple array matrix – МАХ), позволяющие программно сконструировать в одном корпусе электронную схему, эквивалентную схеме, включающей от десятков до тысяч ИС малой и средней степени интеграции [8, 22, 23, 27, 29]. Это дает возможность на одной или нескольких программируемых логических интегральных схемах реализовать блоки модуляторов и демодуляторов для цифровых телевидения и радиовещания, устройства обработки информации (цифровые фильтры, умножители, и др.), системы автоматики и управления и т.д. Технология проектирования с использованием программируемых логических интегральных схем позволяет существенно сократить (до нескольких часов или дней) процесс проектирования устройства, уменьшая при этом и затраты на проектирование. Кроме того, такая технология дает возможность вносить изменения в структуру цифрового устройства на любой стадии проектирования и изготовления, в том числе после распайки больших интегральных схем на плате и даже в процессе работы. Фирмы, производящие программируемые логические интегральные схемы, ведущими среди которых являются ALTERA и XILINX, выпускают микросхемы EPLD – программируемые электрическим способом логические устройства и микросхемы FPGA (Field Programmable Gate Array) – многократно реконфигурируемые ПЛИС. Интегральная схема FPGA состоит из матрицы ячеек логических схем, соеди-

нённых между собой логическими ключами. Содержащаяся в микросхеме статическая память после заполнения определённой последовательностью двоичных символов осуществляет организацию логических ячеек (их настройку на реализацию определённых функций) и их соединения, что позволяет получать требуемые электрические схемы.

Фирмами, производящими программируемые логические интегральные схемы разработаны пакеты прикладных программ для автоматизированного проектирования различных цифровых устройств на выпускаемых ими ПЛИС.

Эти пакеты позволяют создавать с помощью компьютера цифровые устройства, реализующие заданные алгоритмы работы, проводить моделирование (симуляцию) с анализом функциональных и временных характеристик и формировать последовательность двоичных символов для конфигурирования FPGA или программирования EPLD БИС. Возможность перестройки программируемых логических интегральных схем на реализацию цифровых радиотехнических устройств позволяет им конкурировать с микропроцессорами.

Основными компонентами структуры программируемых логических интегральных схем (рис.3.13) фирмы ALTERA семейства FLEX 10К, являются следующие [2, 22, 23, 27]:

- логические блоки (ЛБ), содержащие 8 логических элементов (ЛЭ) и локальную программируемую матрицу соединений (ЛПМС). Структура логического блока дана на рис.3.14, а логического элемента – на рис.3.15;

- реконфигурируемые модули памяти (РМП), представленные на рис.3.16;

- глобальная программируемая матрица соединений (ГПМС), разделённая на строки и столбцы;

- межблочные цепи каскадирования и переноса;

- программируемые элементы ввода-вывода (ЭВВ), приведенные на рис.3.17.

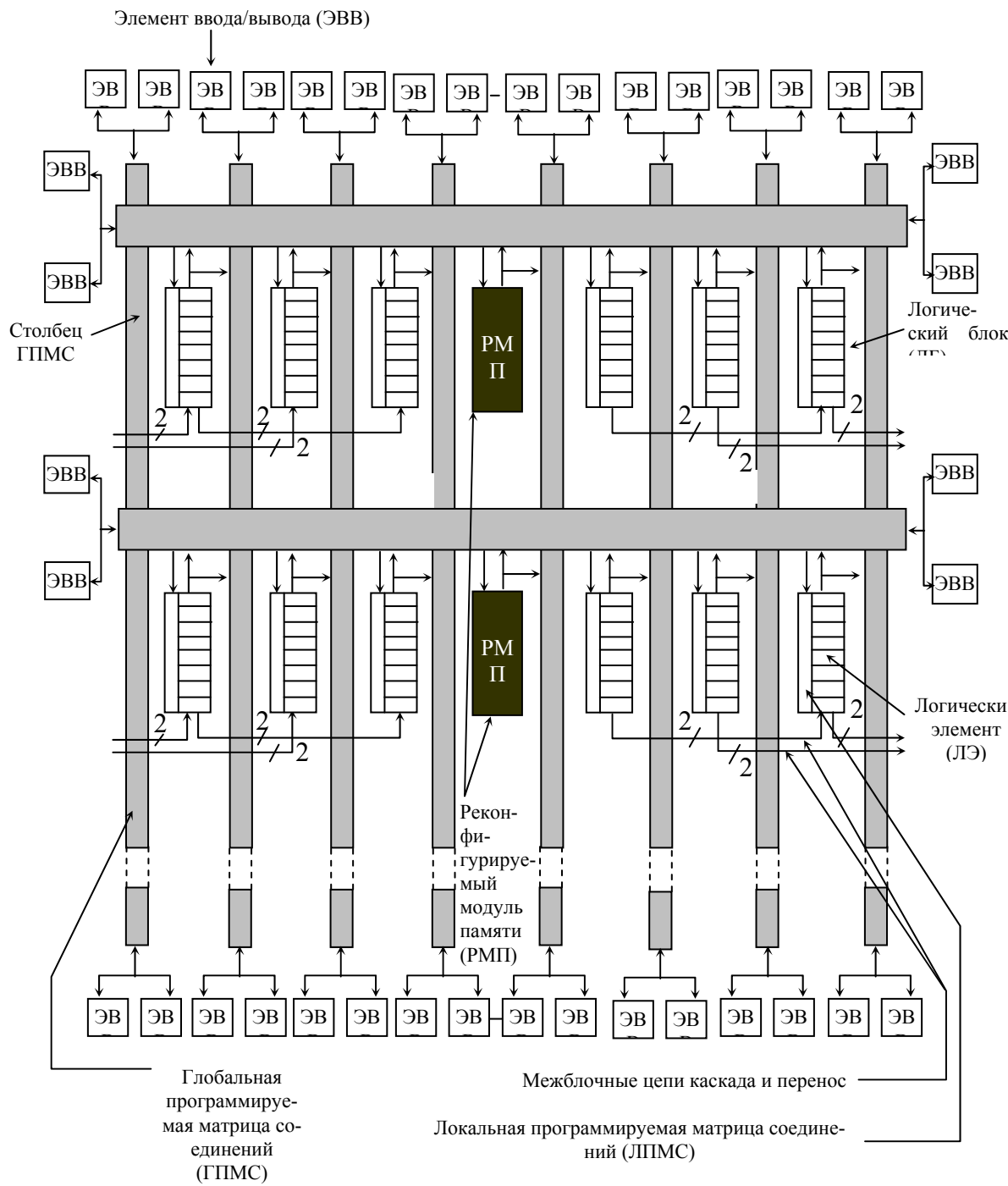


Рис. 3.13 Структура ПЛИС семейства FLEX10K

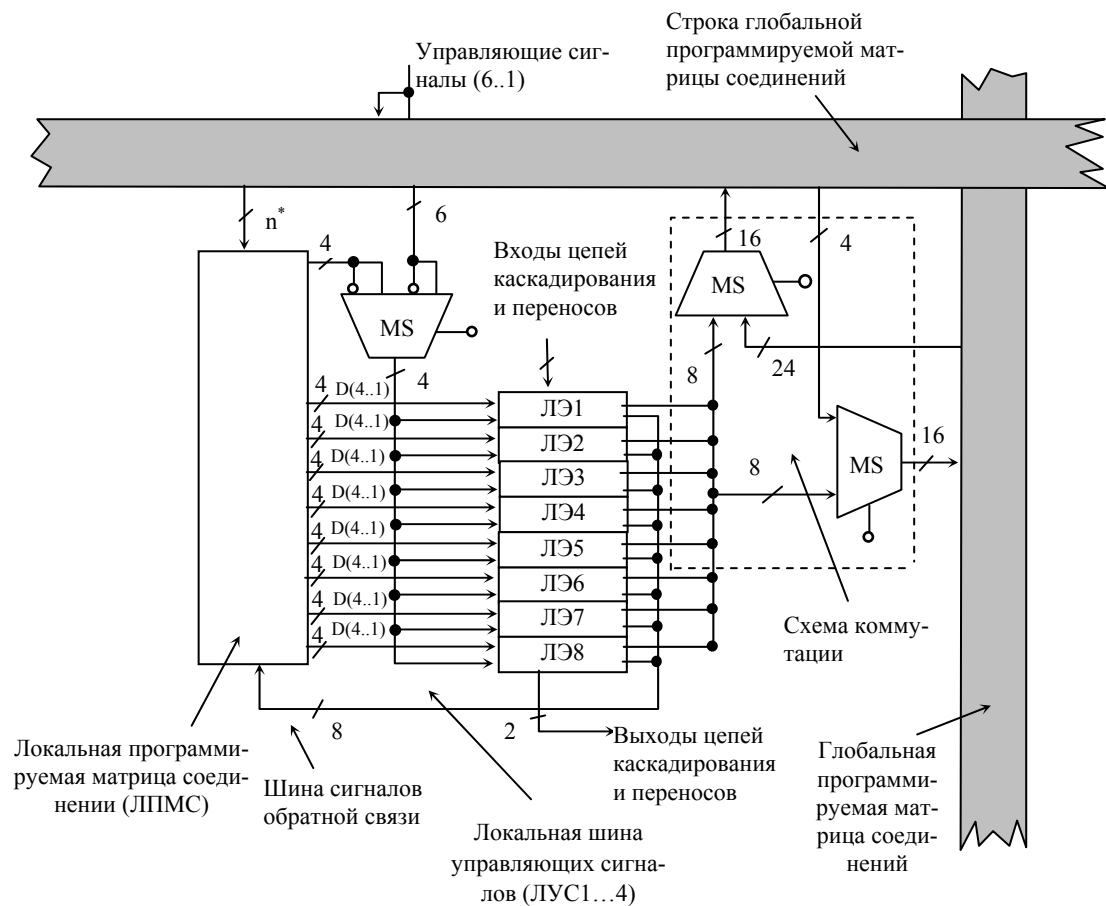


Рис. 3.14 Структура логического блока.

Логические блоки программируемых логических интегральных схем образуют матрицу (рис.3.13). В центре каждой строки матрицы расположен реконфигурируемый модуль памяти. Для каждого столбца и каждой строки матрицы встроена строка и столбец глобальной программируемой матрицы соединений. Они обеспечивают минимальный сдвиг фронтов тактовых сигналов и минимальную задержку распространения сигналов управления.

Структурная схема логического блока (рис.3.14) содержит 8 логических элементов (ЛЭ1, ..., ЛЭ8), локальную программируемую матрицу соединений (ЛПМС), шину локальных управляющих сигналов (ЛУС1...4), цепи каскадирования и переноса и мультиплексоры (MS), осуществляющие выбор одного из своих входных сигналов для передачи на выход.

На локально программируемую матрицу соединений логического блока поступает n (число n зависит от типа ПЛИС) входных сигналов со строки глобальной программируемой матрицы соединений и 8 сигналов обратной связи с логических элементов ЛЭ1...ЛЭ8 блока. Локальные управляющие сигналы (ЛУС1...4) представляют собой глобальные управляющие сигналы УС (6...1), либо сигналы из локально программируемой

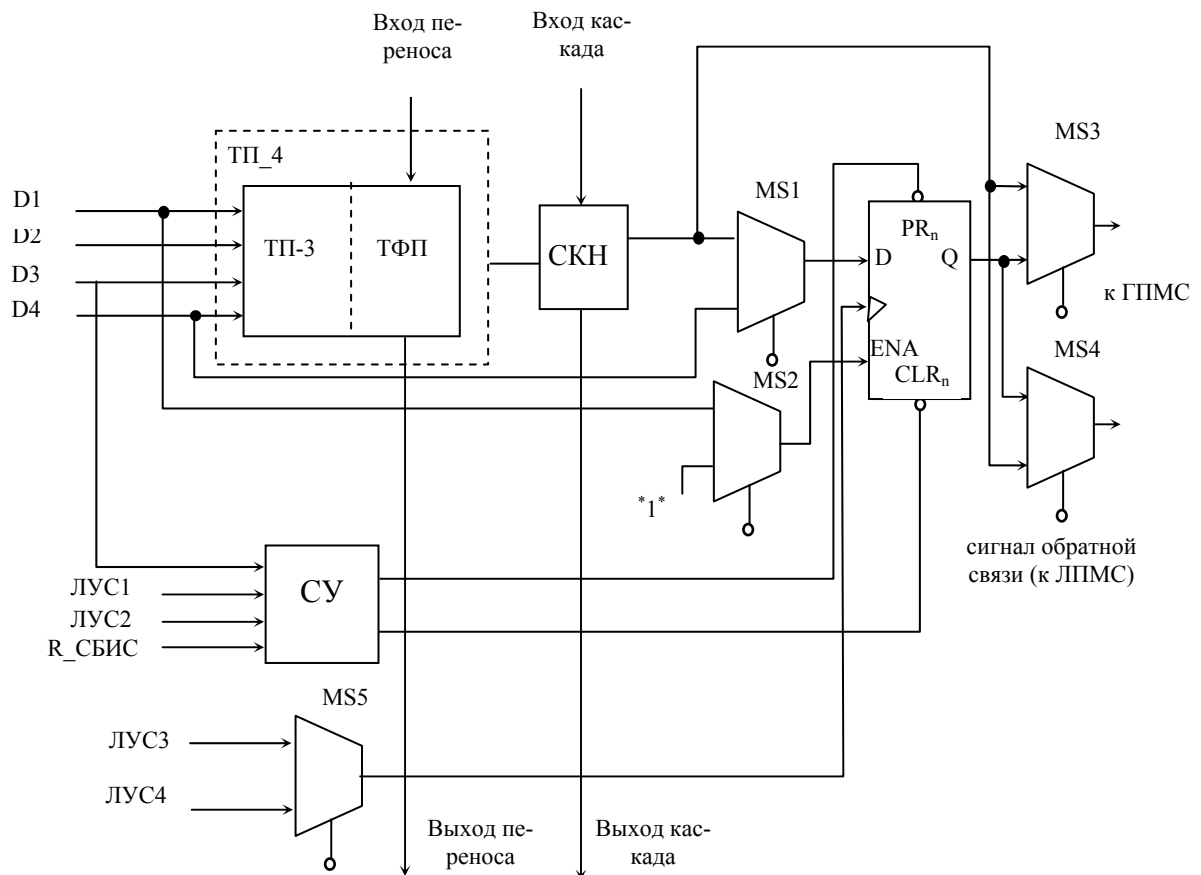


Рис.3.15 Структура логического элемента ПЛИС семейства FLEX 10 К

В состав логических элементов (рис.3.15) входят 4-входовая таблица перекодировок (ТП-4), схема каскадного наращивания (СКН), синхронный триггер, схема управления установкой/сбросом триггеров (СУ), набор программируемых мультиплексоров (MS 5...1), осуществляющих выбор одного из входных сигналов для передачи их на выход. Входная таблица перекодировок ТП-4 может быть преобразована в совокупность 3-х входовой таблицы перекодировки (ТП-3) и таблицы формирования переноса. На входы логических элементов поступают данные (D4...D1), локальные управляющие сигналы (ЛУС 4...1), сигналы переноса и каскадирования, сигнал (R_СБИС) асинхронного сброса триггеров ПЛИС. С помощью входной таблицы перекодировок ТП-4 можно реализовать любую логическую функцию 4-х переменных. Схема каскадного наращивания (СКН) позволяет организовать логические функции большего числа переменных. Схемы управления (СУ) позволяют реализовать различные асинхронные режимы работы триггера (сброс, установка, загрузка и др.).

Реконфигурируемый модуль памяти (РМП), структура которого дана на рис. 3.17, представляет собой совокупность элементов памяти, которую с помощью имеющихся в блоке синхронных буферных регистров (RG) и

программируемых мультиплексоров (MS), можно конфигурировать как ОЗУ или ПЗУ с организацией памяти 256*8, 512*4, 1024*2, 2056*1 бит. На основе реконфигурируемого модуля памяти можно реализовать не только блоки статической памяти (SRAM, FIFO, двухпортовой SRAM) и блоки постоянных запоминающих устройств (ROM), но и многоразрядные сумматоры, умножители, делители, кодеры/декодеры и конечные автоматы, используя табличный способ задания функций.

Такие устройства широко применяются в системах фазовой автоподстройки частоты мобильных радиотелефонов, радиотехнических малогабаритных приемо-передающих станций и т.п.

Элемент ввода-вывода, структура которого приведена на рис.3.17, соединяет канал строки или столбца глобальной программируемой матрицы

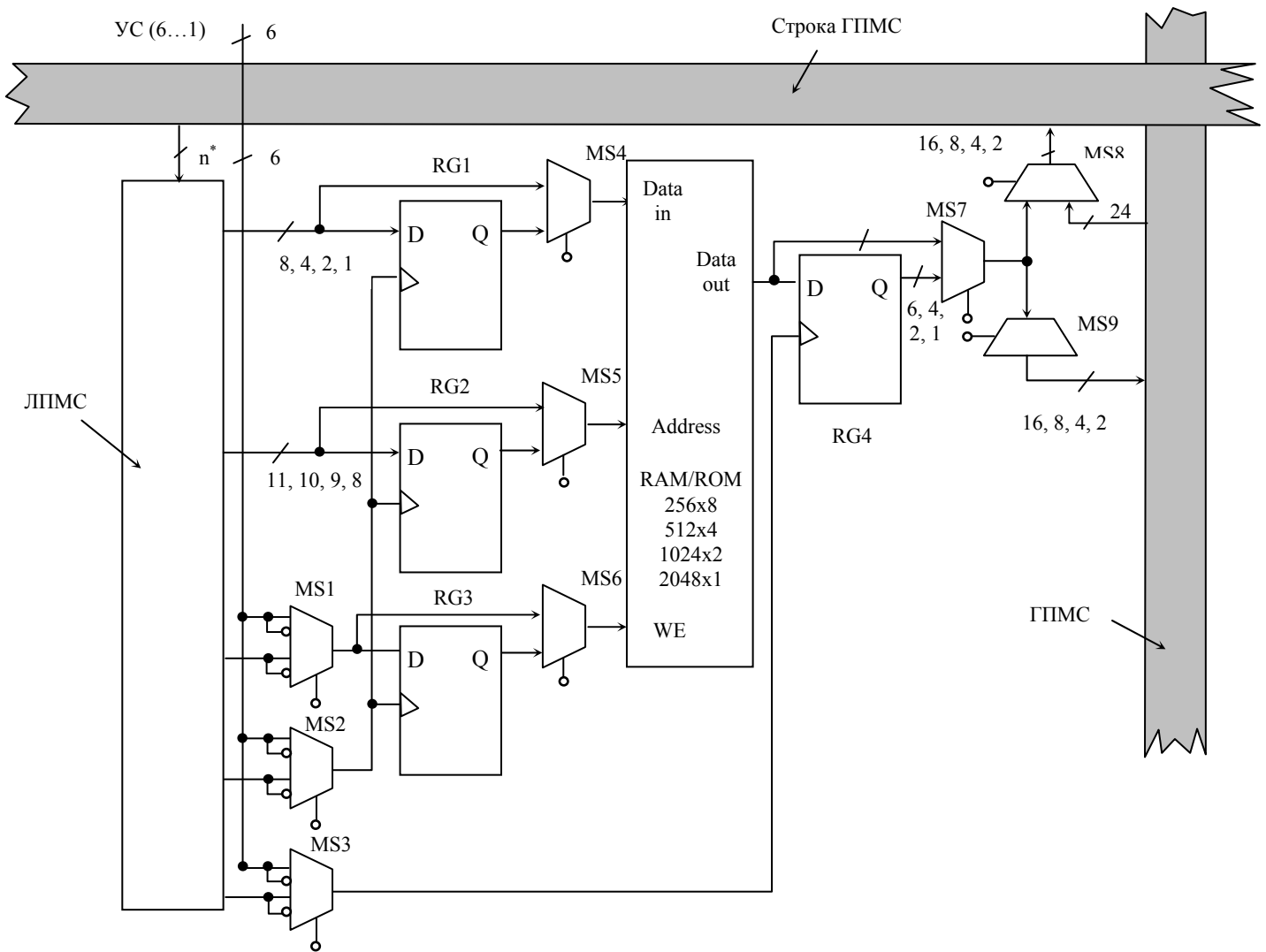


Рис. 3.16 Структура реконфигурируемого модуля памяти (РМП) ПЛИС семейства FLEX 10 K

соединений с физическим выводом микросхемы и осуществляет ввод, вы-

вод или временное хранение данных. Он может работать по выходу в режиме с открытым коллектором.

Выпущенные фирмой ALTERA программируемые логические инте-

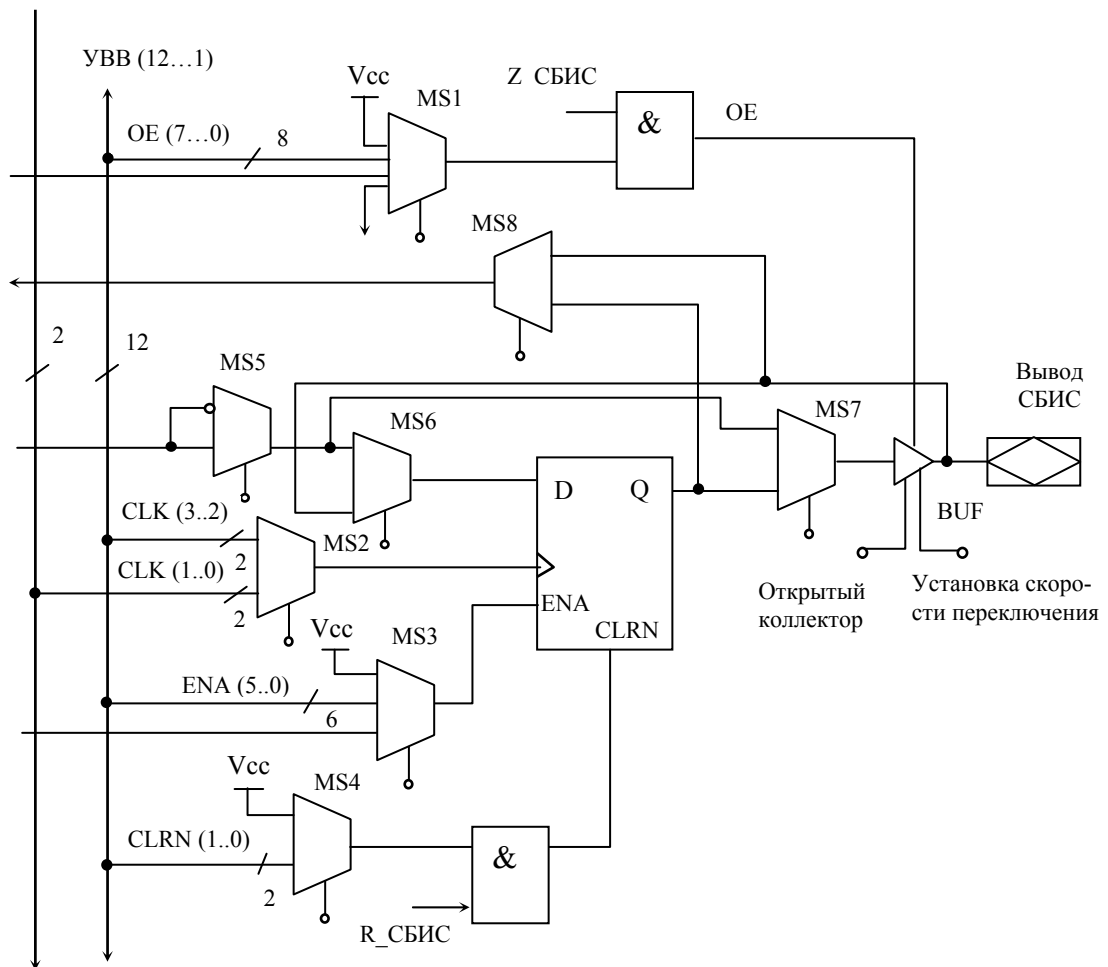


Рис. 3.17 Структура элемента ввода-вывода ПЛИС семейства FLEX 10 К

гральные схемы семейства FLEX 20К (APEX) объединяют достоинства FPGA ПЛИС (использование таблиц перекодировок и встроенных модулей памяти) и EPLD ПЛИС (логические блоки вычисления СДНФ). Они содержат до 2670000 эквивалентных логических элементов (FLEX 10К до 250 000), обеспечивают устойчивую работу на частотах до 600 МГц. Все программируемые логические интегральные схемы, использующие технологию FPGA, имеют неограниченное число циклов реконфигурирования.

Глава 4. Цифроаналоговые преобразователи

4.1 Принцип цифроаналогового преобразования

Цифроаналоговый преобразователь можно рассматривать как декодирующее устройство (рис.4.1), которое, воспринимая на входы цифровой сигнал (код) $X = \{x_1, x_2, \dots, x_N\}$ и эталонный (или опорный) $u_{\text{оп}}$, формирует на выходе аналоговый сигнал u , связанный с входными сигналами соотношением

$$u = u_{\text{оп}} \cdot D,$$

где $D = \frac{x_1}{2} + \frac{x_2}{2^2} + \dots + \frac{x_N}{2^N}$, $x_i = \begin{cases} 0 \\ 1 \end{cases}$.

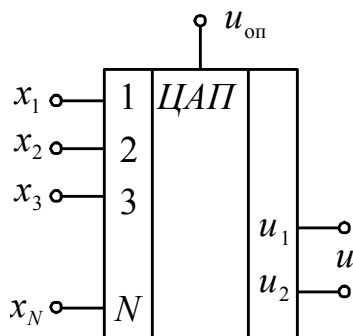


Рис.4.1. Условное обозначение цифроаналогового преобразователя.

В основу работы цифроаналогового преобразователя положено суммирование токов, соответствующих весам разрядов преобразователя. В состав ЦАП (два варианта реализации приведены на рис.4.2) входят: источник опорного напряжения $u_{\text{оп}}$, набор двухпозиционных электронных ключей (реализующих x_i), резистивная токозадающая матрица весовых резисторов и выходной сумматор на операционном усилителе [1, 6, 7, 8, 14].

Относительные веса разрядных токов I_1, I_2, \dots, I_N задаются при помощи матрицы весовых резисторов, величины сопротивлений которых удваиваются при переходе от старшего разряда к младшему. Инвертирующий операционный усилитель с большим входным сопротивлением и высоким коэффициентом усиления используется для суммирования отдельных разрядных токов и для формирования соответствующего аналогового напряжения. Суммарный аналоговый ток I_0 на инвертирующем входе операционного усилителя с эталонным напряжением $u_{\text{оп}}$ связан следующим соотношением (рис.4.2,б):

$$I_0 = \frac{2u_{\text{оп}}}{R} (x_1 \cdot 2^{-1} + x_2 \cdot 2^{-2} + \dots + x_N \cdot 2^{-N});$$

$$u = -I_0 R_0 = -u_{\text{оп}} (x_1 \cdot 2^{-1} + x_2 \cdot 2^{-2} + \dots + x_N \cdot 2^{-N}),$$

причем

$$u_{\text{max}} = -u_{\text{оп}} \frac{2^N - 1}{2^N}, \quad \Delta = \frac{u_{\text{оп}}}{2^N}.$$

При коммутации напряжений (рис.4.2,а) быстродействие цифроаналогового преобразователя ограничивается наличием паразитных емкостей и переходными процессами установления напряжений. При коммутации токов (рис.4.2, б) напряжения

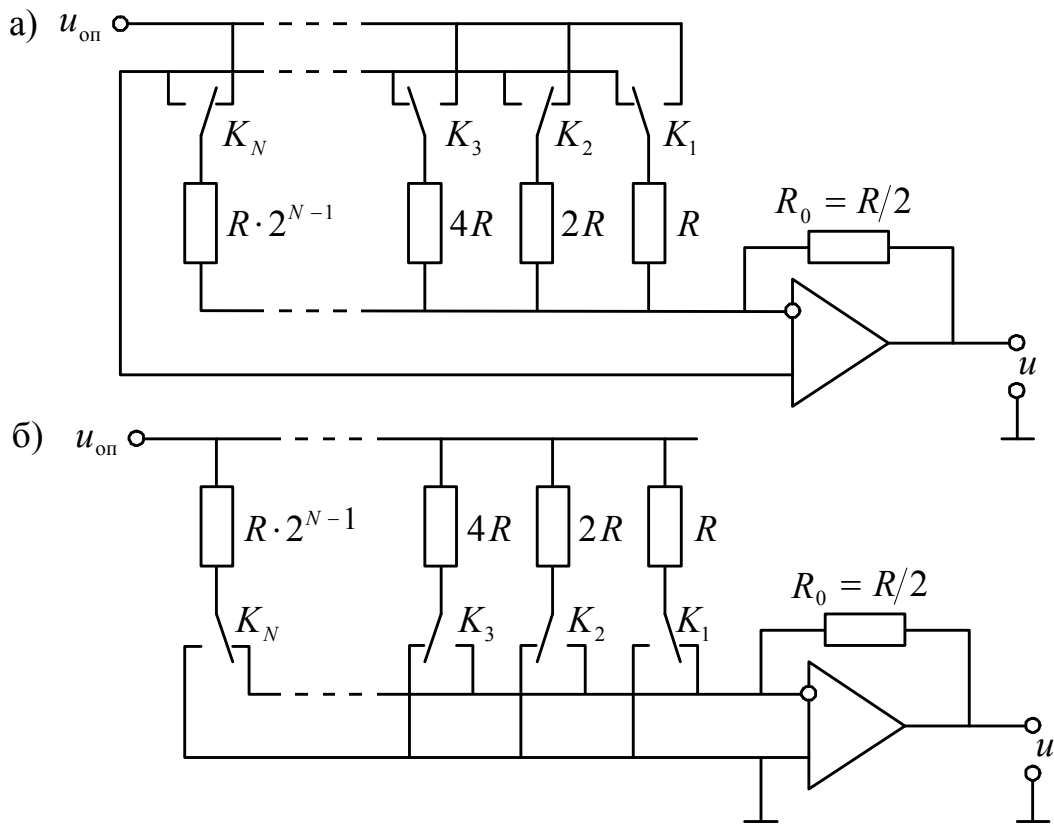


Рис. 4.2. Реализация цифроаналогового преобразователя с коммутацией напряжений (а) и токов (б)

во всех точках цифроаналогового преобразователя остаются неизменными. Это позволяет уменьшить длительности переходных процессов и времена установления напряжений, что приводит к увеличению быстродействия цифроаналогового преобразователя.

4.2 Основные параметры цифроаналогового преобразователя

При последовательном возрастании значений входного цифрового кода X выходной сигнал цифроаналогового преобразователя имеет вид ступенчатого пилообразного напряжения, форма которого представлена на рис.4.3, где q_i ($i = 0, 1, \dots, 2^N$) – значения N -разрядного цифрового кода X ; u_{\max} – максимальное значение выходного напряжения; Δ – шаг квантования по уровню напряжений. Такая зависимость называется характеристикой преобразования ЦАП.

Для описания цифроаналогового преобразователя используют следующие основные параметры:

Число разрядов N – число разрядов кода, связанного с аналоговой величиной, которое может воспринимать ЦАП, или двоичный логарифм максимального числа кодовых комбинаций на входе цифроаналогового преобразователя.

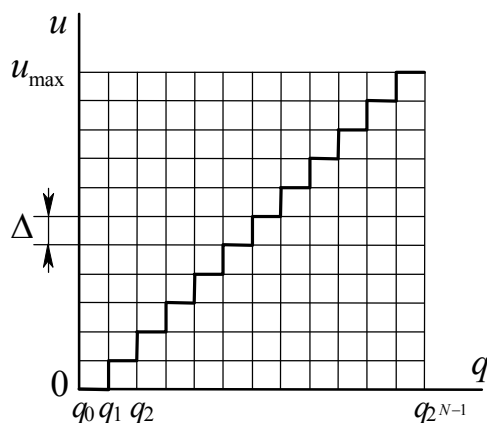


Рис.4.3. Характеристика преобразования ЦАП

Иногда пользуются термином «разрешающая способность». Она характеризует возможное число уровней аналогового сигнала и обычно выражается в виде количества двоичных разрядов входного цифрового сигнала. При разрешающей способности в N двоичных разрядов преобразователь должен обеспечивать формирование выходного аналогового сигнала, имеющего 2^N дискретных уровней.

Коэффициент преобразования $K_{пр}$ – отношение приращения выходного сигнала к приращению входного сигнала для линейной характеристики цифроаналогового преобразователя.

Погрешность (точность) преобразования представляет собой величину отклонения аналогового выходного сигнала от расчетного значения. Обычно ЦАП характеризуются абсолютной погрешностью преобразования в конечной точке шкалы ($\delta_{\text{пш}}$), являющейся отклонением значения выход-

ного напряжения цифроаналогового преобразователя от номинального значения, соответствующего конечной точке характеристики преобразования (рис.4.4,а). Погрешность может быть выражена в процентах от полного диапазона ($u_{дп}$) изменения выходного сигнала ($0 - u_{max}$) или в виде части (обычно половины) наименьшего значащего разряда, а также в виде числа значащих разрядов с гарантированной точностью. Если гарантированная точность – N двоичных разрядов, то максимальная возможная погрешность $\delta_{пш}$ может быть определена в виде: $\delta_{пш} \leq u_{max} \cdot 2^{-N}$.

Аналогично для цифроаналогового преобразователя, имеющего N двоичных разрядов и погрешность $1/2$ наименьшего значащего разряда (НМЗР), фактическая погрешность $\delta_{пш}$ должна удовлетворять неравенству

$$\delta_{пш} \leq u_{дп} \cdot \frac{1}{2} \cdot \frac{1}{2^N} = \frac{u_{дп}}{2^{N+1}}.$$

Необходимо отметить, что разрешающая способность и погрешность – не одно и то же. Например, при разрешающей способности в 12 двоичных разрядов может быть обеспечена требуемая точность в 10 двоичных разрядах и наоборот.

Время установления $T_{уст}$ характеризует быстроедействие ЦАП и представляет собой интервал времени от момента скачкообразного изменения входного цифрового сигнала до момента, когда выходной аналоговый сигнал достигнет нового установившегося значения с номинальной погрешностью (обычно $\pm 1/2$ НМЗР). Максимальным временем установления является время, требуемое для изменения выходного напряжения на полную шкалу дискретных значений.

Наибольший значащий разряд (НБЗР) является разрядом входного цифрового сигнала, имеющим наибольший вес. Он обеспечивает наибольшее приращение уровня аналогового сигнала. В цифроаналоговом преобразователе с двоичными разрядами наибольший значащий разряд вызывает изменение величины аналогового сигнала на половину его диапазона ($u_{max}/2$).

Наименьший значащий разряд (НМЗР) – это разряд входного цифрового сигнала, имеющий наименьший вес. Наименьший значащий разряд обуславливает наименьшее скачкообразное приращение аналогового сигнала, равное $u_{max}/2^N$.

Напряжение смещения нуля на выходе u_{00} – напряжение на выходе цифроаналогового преобразователя при входном коде, соответствующем

щем нулевому значению выходного напряжения (рис.4.4,а). Измеряется u_{00} в единицах наименьшего значащего разряда.

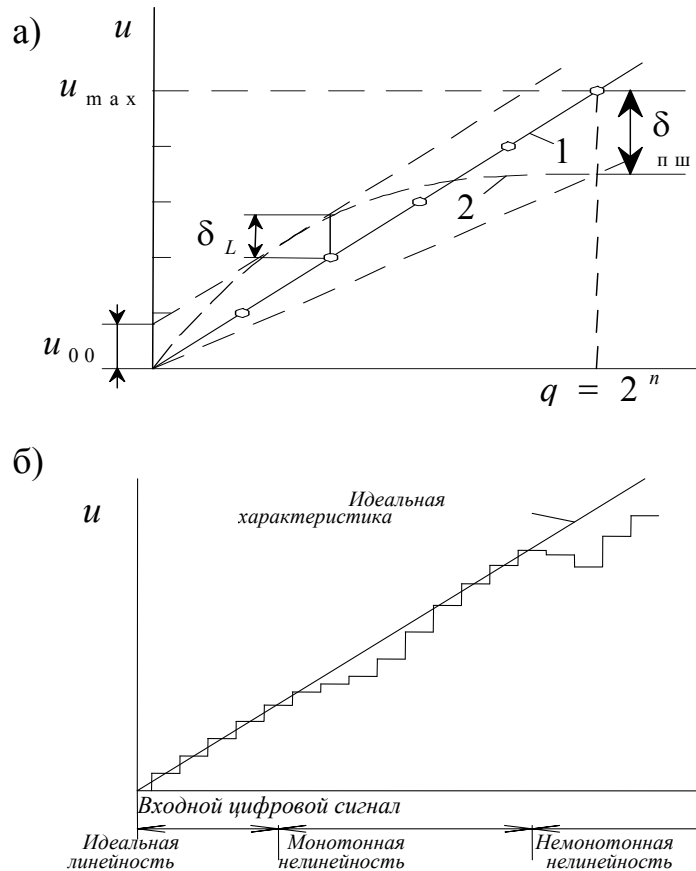


Рис. 4.4. Погрешности преобразования ЦАП

Линейность и монотонность. Если на цифровой вход цифроаналогового преобразователя подается нарастающая последовательность двоичных чисел, то выходной сигнал ЦАП должен представлять собой линейно-нарастающее ступенчатое напряжение с шагом, равным наименьшему значащему разряду. Постоянство ширины и высоты ступенек этого напряжения обуславливает линейность выходного сигнала.

Нелинейность ЦАП δ_L – отклонение характеристики преобразования от прямой линии (кривые 2 и 1 на рис.4.4,а). Величина нелинейности выражается в тех же единицах, что и погрешность.

Дифференциальная нелинейность δ_{LD} – отклонение разности двух аналоговых сигналов, соответствующих соседним значениям кода, от значения единицы наименьшего значащего разряда.

Немонотонность (рис.4.4,б) является наихудшей разновидностью нелинейности. Она имеет место в том случае, когда выходной сигнал изменяется немонотонно при изменении входного цифрового сигнала, т.е. имеет-

ся неидентичность знака приращения мгновенных значений входного и выходного сигналов преобразователя. Немонотонность может возникнуть вследствие накопления погрешностей разрядов. Менее характерными для преобразователей являются такие параметры, как ток потребления ($I_{\text{пот}}$) и диапазон изменения выходного напряжения.

4.3. Реализация ЦАП на интегральных схемах

Как уже отмечалось ранее, цифроаналоговый преобразователь содержит: резистивную матрицу, с помощью которой формируются выходные сигналы, пропорциональные входному коду; набор токовых ключей, реализующих коэффициенты двоичных разрядов; выходной усилитель, суммирующий токи разрядов, и источник опорного напряжения. Рассмотрим принципы построения основных узлов цифроаналогового преобразователя.

Резистивная матрица может иметь различную структуру. Один из ее вариантов (с весовыми резисторами) показан на рис.4.2,б. Здесь взвешенные разрядные токи формируются при помощи N параллельных независимых ветвей, имеющих сопротивление R , $2R$, $4R$ и т. д. В узле взвешивания такого типа значения токозадающих резисторов очень быстро увеличиваются по мере роста числа разрядов. Отношение сопротивления ветви, соответствующей наибольшему значащему разряду, к сопротивлению ветви, отвечающей за наименьший значащий разряд, в этом случае равно:

$$\frac{R_{\text{НБЗР}}}{R_{\text{НМЗР}}} = \frac{1}{2^{N-1}}.$$

Например, при разрешающей способности, равной восьми двоичным разрядам, требуется набор прецизионных резисторов, номинальные значения которых лежат в пределах от R до $128R$. В интегральных схемах трудно получить достаточно точные резисторы, номиналы которых находятся в таком широком диапазоне. Резистивная лестничная цепь $R - 2R$, показанная на рис.4.5, является другим вариантом резистивной схемы взвешивания. Эта цепь позволяет устранить недостаток, присущий схеме с весовыми коэффициентами и заключающийся в необходимости использования резисторов с большим диапазоном номиналов сопротивлений. В этой схеме деление токов, втекающих в узел, на две равные части достигается за счет поэтапного распределения каждого из токов между параллельной ($2R$) и последовательной (R) ветвями. Таким образом, соотношение между токами ветвей подчиняется закону: $I_1 = 2I_2 = 4I_3 = \dots = 2^{N-1}I_N$. При этом обеспечивается легко реализуемое соотношение ($2 : 1$) значений со-

противлений резисторов. Лестничная цепь содержит вдвое больше резисторов, чем цепь с весовыми коэффициентами. Кроме того, в нее в качестве нагрузки должно быть включено балластное сопротивление R_6 . Основным преимуществом лестничной цепи является возможность ее использования с источниками одинаковых токов в цифроаналоговых преобразователях, что позволяет обеспечить высокую степень идентичности параметров элементов, имеющих одинаковую конфигурацию и работающих при одинаковых токах.

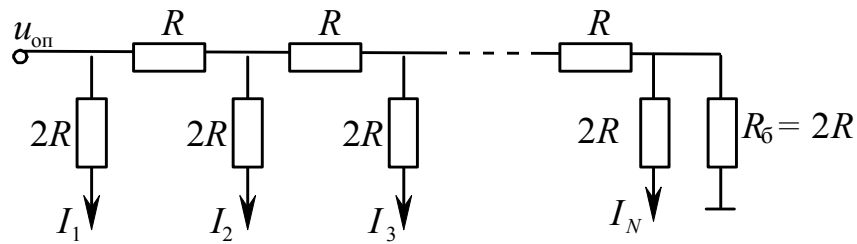


Рис. 4.5. Резистивная цепь $R-2R$

В цифроаналоговых преобразователях, имеющих небольшое число разрядов, применяются цепи с весовыми сопротивлениями. В цифроаналоговых преобразователях, содержащих шесть или более разрядов, преимущественно используются лестничные цепи.

В резистивных цепях должны быть обеспечены идентичность элементов и совпадение их температурных характеристик. Для этого резисторы изготавливаются из материалов с малыми температурными коэффициентами (никель-хром, тантал и др.) и располагаются рядом друг с другом.

Токовые электронные ключи, предназначенные для коммутации элементов резистивной матрицы, должны иметь высокое быстродействие и не вносить заметных погрешностей в разрядные токи, иметь хорошую развязку (большое сопротивление изоляции между управляющей цифровой цепью и выходной аналоговой частью), малые обратные токи утечки, т. е. токи через разомкнутые электронные ключи.

Электронные ключи для быстродействующих цифроаналоговых преобразователей строятся на биполярных транзисторах и диодах. Для цифроаналоговых преобразователей со средним и низким быстродействием

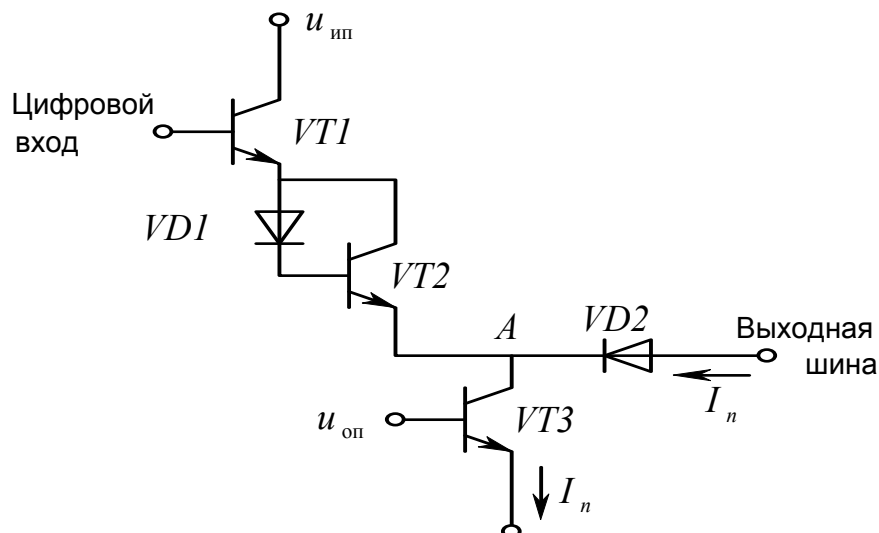


Рис. 4.6. Электронный ключ на биполярных транзисторах

применяются токовые электронные ключи на КМДП-транзисторах, которые характеризуются малым потреблением энергии.

Один из вариантов токового электронного ключа на биполярных приборах показан на рис.4.6, где $u_{\text{ип}}$ – напряжение источника питания. Если на цифровой вход подан сигнал логического 0, транзисторы $VT1$, $VT2$ и диод $VD1$ открываются, а диод $VD2$ закрывается и отключает выходную шину. Транзистор $VT3$ все время открыт, поэтому через резисторы матрицы течет постоянный ток. Этим достигается отсутствие влияния на быстродействие цифроаналогового преобразователя постоянных времени цепей эмиттеров и постоянных времени электронного ключа, зависящих от сопротивлений матрицы.

Выходным усилителем обычно служит операционный усилитель, который суммирует разрядные токи. Напряжение на выходе операционного усилителя пропорционально входному коду:

$$u_{\text{оу}} = I_0 R_0 = \frac{2R_0}{R} u_{\text{он}} \sum_{i=1}^N x_i \cdot 2^{-i},$$

где R_0 – сопротивление обратной связи операционного усилителя; x_i – разряды цифроаналогового преобразователя.

Источники опорного напряжения (ИОН) $u_{\text{он}}$, как правило, строятся на основе использования полупроводниковых стабилитронов. Схемы двух вариантов построения ИОН даны на рис.4.7. На рис.4.7,а приведена схема источника опорного напряжения, построенная на основе операционного усилителя с параллельной обратной связью, а на рис.4.7,б – с после-

довательной обратной связью. Отношение сопротивлений R_2/R_1 определяется требуемыми значениями $u_{оп}$ и напряжения стабилизации u_c . При использовании подобных схем необходимо выполнять условие $u_{оп} \leq (u_{нас} - \Delta u_{c\max}) \cdot 0,7$, где $u_{нас}$ – граница динамического диапазона операционного усилителя; $\Delta u_{c\max}$ – максимальное температурное и временное изменение напряжения стабилизации. Температурная стабильность опорного напряжения $u_{оп}$ в данных устройствах определяется стабильностью напряжения стабилизации u_c и выходного напряжения операционного усилителя.

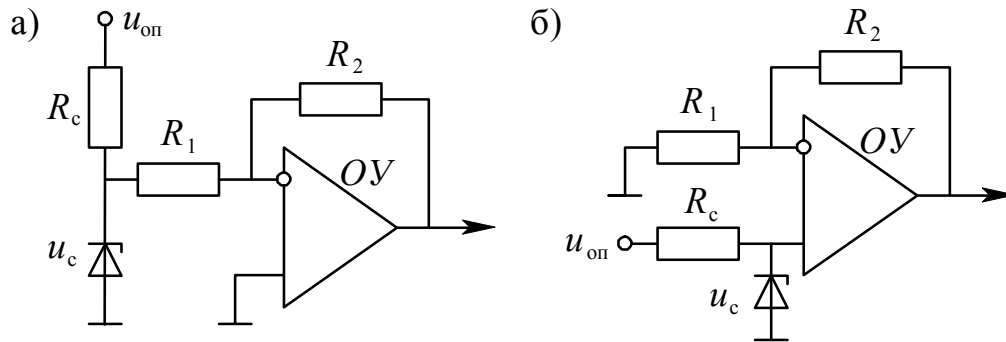


Рис. 4.7. Источники опорного напряжения

Построение высокоточных ($\delta \leq 0,1\%$) источников опорного напряжения связано с введением в схему реализации элементов, обеспечивающих температурную и временную стабилизацию тока опорного стабилитрона [7, 14].

Интегральные схемы цифроаналоговых преобразователей [7,14,15] содержат различные реализации резистивной матрицы, токовых электронных ключей и некоторые дополнительные элементы (регистры, преобразователи уровней, схемы стабилизации тока и др.) и не имеют суммирующего операционного усилителя и источника опорного напряжения.

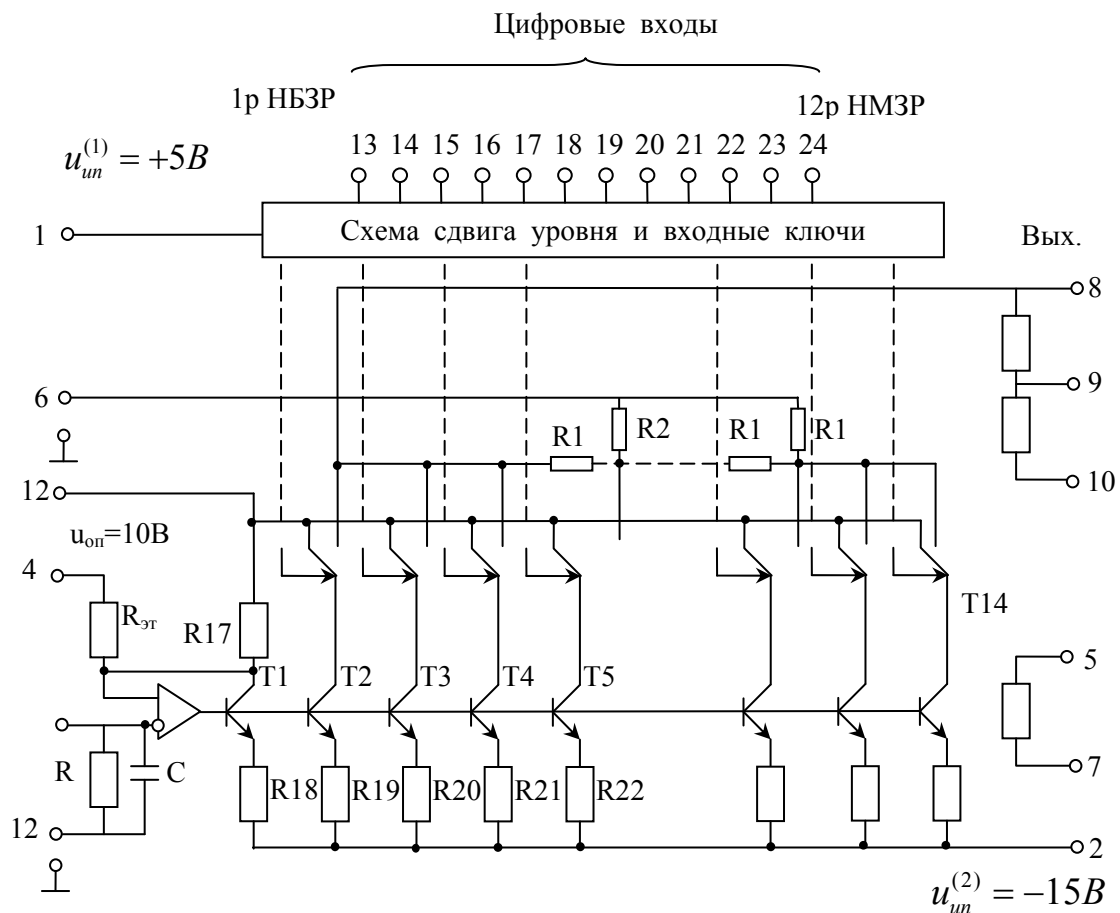


Рис. 4.8. Принципиальная схема цифроаналогового преобразователя К1108ПА1

Быстродействующим прецизионным цифроаналоговым преобразователем на основе биполярных структур с диэлектрической изоляцией является интегральная схема К1108ПА1. В состав интегральной схемы входят 12 идентичных токовых ключей, схема стабилизации тока ключей, резистивная матрица взвешенного типа и группа резисторов (рис.4.8.). Схемы сдвига и согласования логических уровней, состоящие из эмиттерных повторителей и стабилитронов на смещенных в обратном направлении р-п-переходах, управляют дифференциальными переключателями разрядных токов без значительной задержки фронтов цифровых сигналов.

Интегральная схема К1118ПА2 представляет собой 10-разрядный быстродействующий цифроаналоговый преобразователь, предназначенный для обработки сигналов в телевизионной, связной и измерительной радиотехнической аппаратуре.

Структурная схема такой интегральной схемы (рис. 4.9) содержит преобразователи входных уровней (ПУ), входной регистр данных с цепями управления (RG), генераторы разрядных токов (ГТ), токовые ключи (Кл.), $R-2R$ матрицу ($R-2R$) и операционный усилитель (ОУ) для обеспечения

температурной стабильности выходного напряжения ЦАП (вход U_{REF} представляет собой вход опорного напряжения). Преобразователи входных уровней сигналов осуществляют согласование уровней входного регистра ЦАП, реализованного на синхронных D -триггерах эмиттерно-связанной логики, с логическими уровнями источников цифрового сигнала. Входные сигналы могут иметь уровни напряжений, соответствующие ТТЛ и ЭСЛ-уровням. В этой связи напряжения питания интегральной схемы имеют значения $U_{CC1} = +5B$, $U_{CC2} = -5B$.

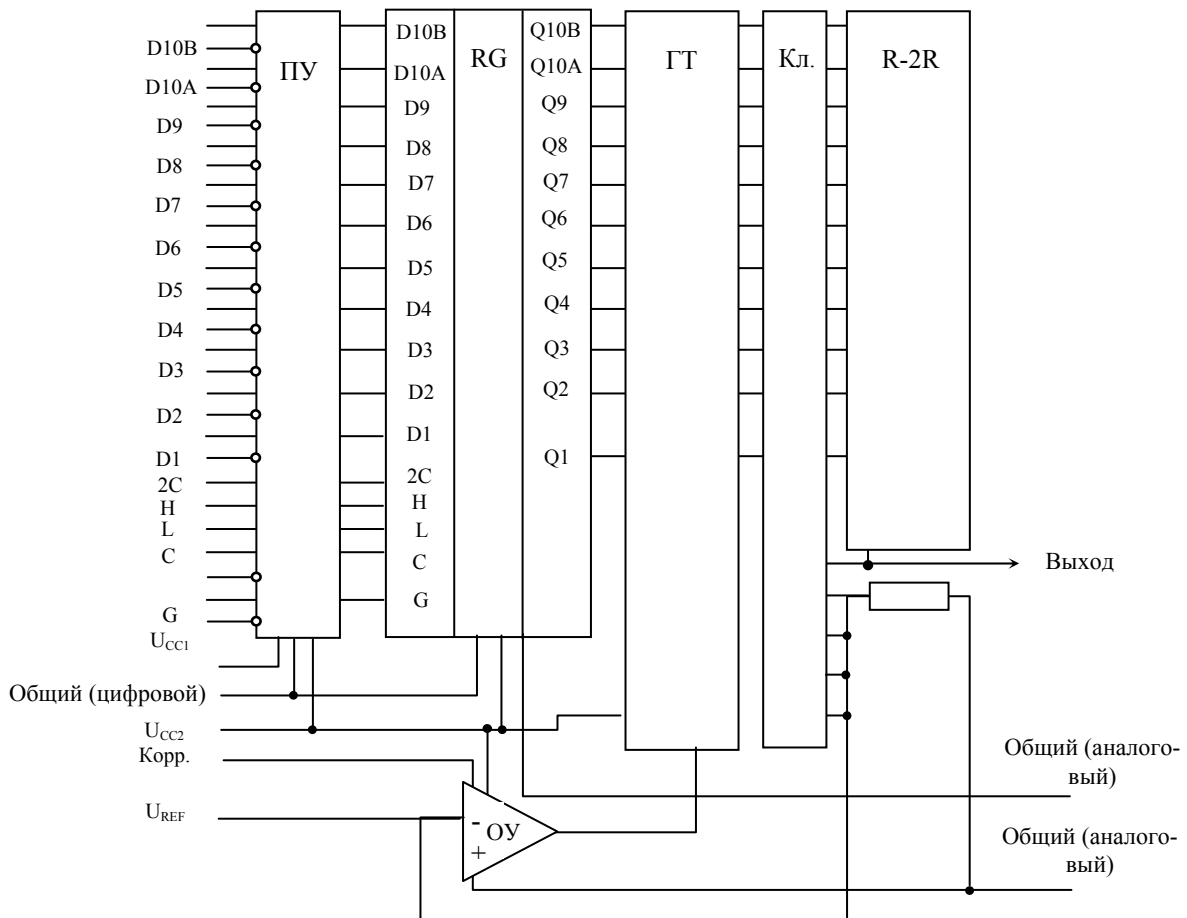


Рис. 4.9. Структурная схема 10-разрядного ЦАП серии K1118ПА2

Входной регистр данных имеет цепи управления с входами C , G , H , L , $2C$, позволяющие осуществлять преобразования прямых и обратных двоичных кодов, преобразования прямых и обратных дополнительных двоичных кодов. Кроме того, входной регистр данных дает возможность установить значения напряжения начальной и конечной точек характеристики преобразования.

Взвешивание разрядных токов осуществляется в идентичных структурных звеньях, содержащих генераторы токов и токовые ключи. Причем токи 8 – ми младших разрядов формируются с использованием $R-2R$ мат-

рицы, а токи двух старших разрядов поступают на выход преобразователя непосредственно.

Описание и структура выпускаемых промышленностью интегральных схем цифроаналоговых преобразователей, их характеристики и параметры, а также типовые схемы включения подробно изложены в [7,10,14]. Зарубежные фирмы (Analog Devices и др.) выпускают интегральные схемы одно-, двух- и четырёхканальных цифроаналоговых преобразователей с числом разрядов 8–14, а также высокоточные цифроаналоговые преобразователи с разрядностью 16.

Основные электрические параметры некоторых цифроаналоговых преобразователей, выполненных в виде отдельных микросхем, приведены в табл. 4.1.

Таблица 4.1.

Тип ИС	Число разрядов	$t_{уст}$, мкс	δ_L , %	$\delta_{ПШ}$, % (ед.НМЗР)	$u_{ин}$, В	$u_{оп}$, В	$I_{пот}$, мА	Технология				
К572ПА1(А-Г)	10	5	А $\pm 0,1$	(± 3)	5...17	± 17	2	КМОП, ТТЛ				
КР572ПА2(А-В)	12	15	Б $\pm 0,2$	$\pm 0,05$	5;15	± 15	2;2	КМОП, ТТЛ				
			В $\pm 0,4$									
			Г $\pm 0,8$									
К594ПА1	12	3,5	А $\pm 0,025$	$\pm 0,03$	5;15	9...11	25; 35	КМОП, ТТЛ				
К1108ПА1(А-Б)	12	0,4	Б $\pm 0,05$						5;-15	2,2...10, 5	15; 46	ТТЛ
			В $\pm 0,01$									
К1118ПА1	8	0,02	$\pm 0,024$		-15; -5,2		130	ЭСЛ				

Глава 5. Аналого-цифровые преобразователи

5.1. Принципы аналого-цифрового преобразования и основные параметры АЦП

Аналого-цифровые преобразователи используются для преобразования выборочных значений входного непрерывного сигнала в цифровой код с определенным числом разрядов. Процедура аналого-цифрового преобразования включает в себя квантование выборочного значения аналогового сигнала по амплитуде и формирование цифрового кода, значение которого определяется числом уровней квантования.

При квантовании по уровню текущее значение входного сигнала заменяется ближайшим дискретным значением, выбранным по шкале разрешенных уровней. Таким образом, входной сигнал при квантовании по уровню как бы накладывается на сетку горизонтальных линий, разнесенных на величину $\Delta_{\text{кв}}$. Если одновременно производится квантование по времени, то входная величина накладывается на сетку, имеющую по горизонтальной оси (оси времени) деления через интервал Δt , а по вертикальной (оси уровней) – через интервал $\Delta_{\text{кв}}$. Величина $\Delta_{\text{кв}}$ характеризует потенциальную разрешающую способность преобразователя по уровню (амплитуде входного сигнала), т. е. наименьшее различимое приращение входного уровня.

Следовательно, при квантовании значение входного напряжения u_a приближенно представляется в виде доли эталонного (опорного) напряжения $u_{\text{оп}}$. При этом выходным сигналом его является цифровой сигнал (код):

$$D = u_a / u_{\text{оп}} = b_1 \cdot 2^{-1} + b_2 \cdot 2^{-2} + \dots + b_N \cdot 2^{-N},$$

где N – длина двоичного цифрового кода (число двоичных разрядов); b_1, b_2, \dots, b_N – разрядные коэффициенты, принимающие значения ноль или единица.

Разрядные коэффициенты выходного сигнала АЦП могут быть получены либо одновременно на N параллельных выходах, либо могут поочередно появляться на одном и том же выходе. При последовательном формировании разрядных коэффициентов, сигнал b_1 , соответствующий наибольшему значащему разряду, обычно определяется первым, затем на выходе появляются разрядные коэффициенты последующих разрядов в порядке убывания их веса. В процессе преобразования непрерывного сигнала $s(t)$ в цифровой сигнал $s^{(n)}(k\Delta t)$ происходит «квантование» – разбиение

выборочных значений $s^{(n)}(k\Delta t)$ на конечное число n дискретных уровней, отличающихся между собой на величину наименьшего значащего разряда. Этим объясняется возникновение конечной разрешающей способности или погрешности квантования процесса аналого-цифрового преобразования, величина которой $\Delta_{\text{кв}}$ находится в пределах $\pm 1/2$ наименьшего значащего разряда. Погрешность квантования может быть выражена в единицах аналогового напряжения следующим образом: $0 \leq \Delta_{\text{кв}} \leq u_{\text{оп}}/2^{N+1}$.

В процессе аналого-цифрового преобразования используются мгновенные значения $s^{(n)}(k\Delta t)$ в дискретные моменты времени, причем вследствие ограниченного быстродействия АЦП, соответствующий этой величине цифровой сигнал появляется на выходе некоторое время спустя. Время, необходимое для завершения преобразования аналогового сигнала в цифровой, называется временем преобразования.

Для описания АЦП используют следующие основные параметры.

Число разрядов (N) – число разрядов кода, связанного с аналоговой величиной, которое может вырабатывать аналого-цифровой преобразователь.

Разрешающая способность – способность АЦП различать два ближайших значения входного сигнала. Определяется обычно числом различных кодовых комбинаций на выходе преобразователя и выражается либо в битах, либо в децибелах.

Абсолютная погрешность преобразования в конечной точке шкалы ($\delta_{\text{пш}}$) – отклонение значения входного напряжения (тока) от номинального значения, соответствующего конечной точке характеристики преобразования. Измеряется в единицах младшего разряда (МР) преобразования или процентах.

Дифференциальная нелинейность (δ_{LD}) – отклонение разности двух аналоговых сигналов, соответствующих соседним значениям кодов, от значения единицы МР. Измеряется в процентах от величины диапазона входных сигналов или в единицах МР.

Напряжение смещения нуля на входе (U_{10}) – приведенное ко входу напряжение, характеризующее отклонение начала характеристики преобразования от заданного значения. Измеряется в единицах МР.

Время преобразования – интервал времени от момента заданного изменения сигнала на входе до появления на его выходе соответствующего устойчивого кода.

Максимальная частота преобразования – наибольшая частота дис-

кретизации, при которой заданные параметры соответствуют установленным нормам.

Кроме того, важными параметрами аналого-цифрового преобразователя, характеризующими его энергетические показатели, являются диапазон входного напряжения, ток потребления и напряжения источников питания.

5.2 Методы построения АЦП

Наибольшее распространение на практике нашли три метода преобразования: метод последовательного счета, метод поразрядного кодирования и метод считывания [7,10].

Метод последовательного счета

При использовании метода последовательного счета входное аналоговое напряжение сравнивается с эталонным напряжением, представляющим собой сумму значений $\Delta_{\text{кв}}$. Момент равенства входной величины и суммы квантов определяется с помощью сравнивающего устройства. Результат преобразования характеризуется числом квантов, используемых при преобразовании. Это число квантов представляется с помощью счетчика в виде двоичного кода.

Аналого-цифровые преобразователи, построенные по данному методу, подразделяются на АЦП косвенного преобразования и АЦП на основе дискретного сравнения напряжений. К последним, в частности, относится следящий АЦП (рис.5.1), который содержит реверсивный двоичный счетчик (СЧР), выходной цифровой сигнал которого при помощи ЦАП вновь преобразуется в аналоговый сигнал, подаваемый на вход компаратора (К). Счетчик отсчитывает тактовые импульсы, поступающие с генератора тактовых импульсов (ГТИ). Получаемое в результате суммирования импульсов двоичное число при помощи ЦАП преобразуется в напряжение u_0 и сравнивается с выборочным значением входного напряжения $s(k\Delta t)$, полученным при дискретизации $s(t)$ в устройстве выборки и хранения (УВХ).

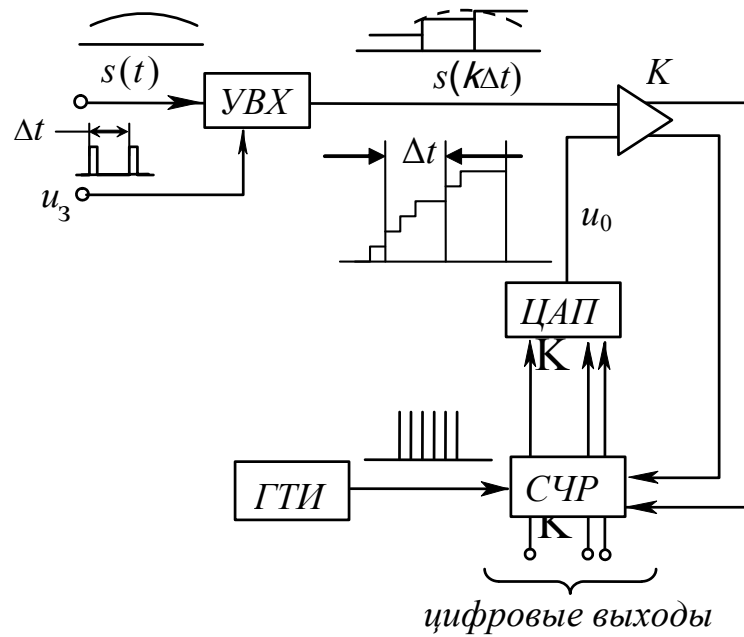


Рис. 5.1. Структурная схема следящего АЦП

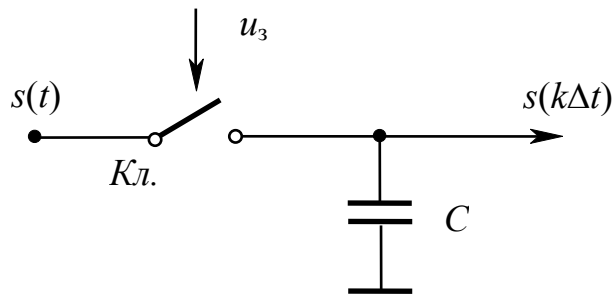


Рис. 5.2. Устройство выборки и хранения

Работа устройства выборки и хранения основана на фиксации мгновенного значения сигнала $s(t)$ при значениях $t = k\Delta t$ ($k = 0, 1, 2, \dots$) на время, необходимое для преобразования этого значения в N -разрядное число. Элементом памяти устройства выборки и хранения (УВХ) является конденсатор C (рис.5.2), подключаемый на время выборки мгновенного значения входного напряжения с помощью электронного ключа $Кл$ к источнику сигнала $s(t)$. Работой электронного ключа $Кл$ управляет генератор задающих импульсов, вырабатывающий последовательность u_3 импульсов прямоугольной формы с тактовой частотой f_T . Время заряда определяется скоростью изменения входного сигнала и постоянной времени заряда $\tau_3 = R_3 C$, где сопротивление цепи заряда R_3 представляет собой сумму выходного сопротивления источника сигнала и сопротивления открытого электронного ключа. Время хранения определяется временем разряда кон-

денсатора C . Когда напряжение на выходе ЦАП достигнет значения, равного $s(k\Delta t)$, компаратор срабатывает и подсчет импульсов прекращается. Цифровой выходной сигнал счетчика, при равенстве $u_0 = s(k\Delta t)$, является выходным сигналом АЦП, соответствующим входному аналоговому напряжению. Необходимо отметить, что в этом случае источник эталонного напряжения входит в ЦАП как его составная часть.

Время преобразования такого аналого-цифрового преобразователя (рис.5.1) является переменным. Его максимальное значение $t_{\text{пр.мах}}$ равно:

$$t_{\text{пр.мах}} = (2^N - 1)/f_T,$$

где f_T – частота следования импульсов генератора тактовых импульсов ($f_T \gg \frac{1}{\Delta t}$). При этом величина $t_{\text{пр.мах}}$ будет определять минимальное значение интервала дискретизации $\Delta t = \Delta t_{\text{мин}} = t_{\text{пр.мах}}$. Величина $f_T = 1/\tau$ выбирается, исходя из условия установления переходных процессов в схеме:

$$\tau \geq \tau_{\text{сч}} + \tau_{\text{цап}} + \tau_{\text{к}} + \tau_{\text{гги}},$$

где $\tau_{\text{сч}}$ – время задержки распространения сигнала в реверсивном двоичном счетчике; $\tau_{\text{цап}}$ – максимальное время установления ЦАП; $\tau_{\text{к}}$ – время задержки включения компаратора; $\tau_{\text{гги}}$ – время переходного процесса в генераторе тактовых импульсов.

При входном сигнале, соответствующем верхней границе диапазона, аналого-цифровые преобразователи этого типа являются довольно инерционными, так как в этом случае необходимо отсчитывать $(2^N - 1)$ импульсов. Однако при малых приращениях такие АЦП весьма быстродействующие. Например, в случае изменения входного напряжения на небольшую величину аналого-цифровой преобразователь может отследить его всего за несколько тактовых импульсов. Поэтому АЦП этого типа и называется «следящим».

Аналого-цифровой преобразователь с единичными приближениями представляет собой упрощенный вариант следящего АЦП. Вместо реверсивного счетчика в нем используется обычный двоичный счетчик. В каждом цикле преобразования счетчик сбрасывается в нулевое состояние и снова начинает отсчитывать число тактовых импульсов. При этом на выходе ЦАП образуется ступенчатое линейно-нарастающее напряжение. Когда выходной сигнал цифроаналогового преобразователя становится равным входному аналоговому напряжению, счетчик блокируется, причем по-

следнее двоичное число счетчика является цифровым эквивалентом входного аналогового сигнала. В отличие от следящего АЦП, данный аналого-цифровой преобразователь возвращается в исходное, нулевое состояние в каждом цикле преобразования.

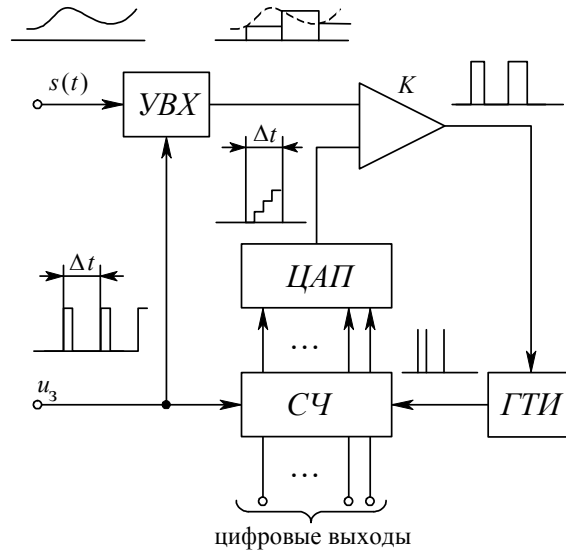


Рис. 5.3. АЦП с единичными приближениями

Особенностью АЦП с единичными приближениями (рис.5.3) является сравнительно небольшие значения частоты дискретизации $f_d = 1/\Delta t$. Например, для АЦП, построенных на основе КМОП-технологии, при $N = 10 \dots 12$, величина $f_d \leq 10$ кГц.

В аналого-цифровых преобразователях косвенного преобразования входное аналоговое напряжение предварительно преобразуется в частоту или длительность импульса и затем в цифровой сигнал. Обычно АЦП косвенного преобразования значительно проще аналого-цифровых преобразователей других видов. Широко распространенной разновидностью такого АЦП является аналого-цифровой преобразователь с генератором пилообразного напряжения или с широтно-импульсным модулятором (рис.5.4).

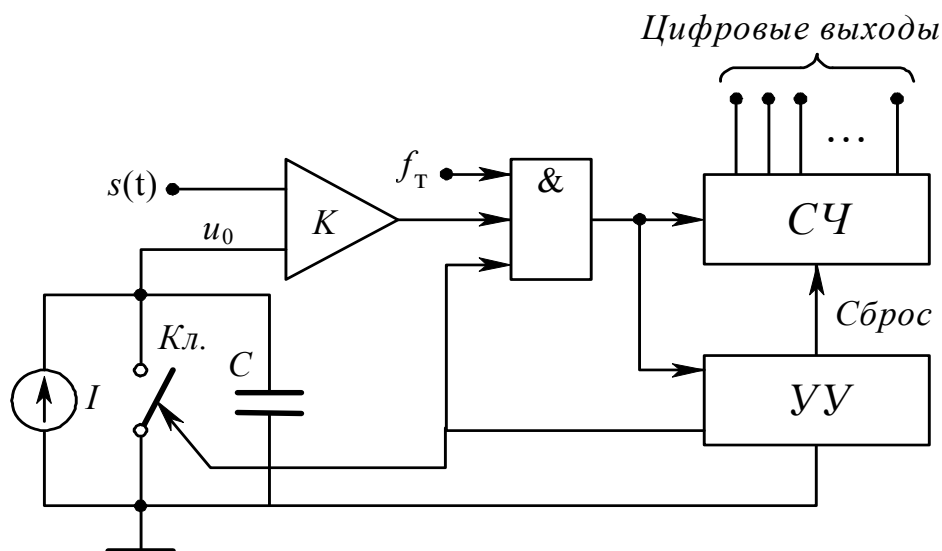


Рис. 5.4. Структурная схема АЦП косвенного преобразования

Аналого-цифровой преобразователь работает следующим образом. В начале цикла преобразования электронный ключ $Kл$ размыкается и, предварительно разряженный конденсатор, начинает заряжаться от источника тока I . В результате на конденсаторе возникает линейно-нарастающее напряжение вида:

$$U = \frac{I}{C} \cdot t.$$

В тот момент, когда электронный ключ $Kл$ размыкается, на счетчик начинают поступать тактовые импульсы с частотой f_T . Отсчет импульсов продолжается до тех пор, пока напряжение u_0 не достигнет величины уровня входного аналогового сигнала $s(t)$. В этот момент на выходе компаратора формируется импульс, блокирующий работу двоичного счетчика СЧ. При этом число, зафиксированное счетчиком, является выходным цифровым сигналом АЦП. После завершения цикла преобразования устройство управления УУ формирует импульс сброса счетчика. С помощью этого же импульса происходит замыкание электронного ключа $Kл$ и подготовка АЦП к новому циклу преобразования. Такие аналого-цифровые преобразователи называют также интегрирующими.

Аналого-цифровой преобразователь косвенного преобразования имеет малое быстродействие, ибо преобразования входного сигнала при разрешающей способности N двоичных разрядов осуществляется в течение $(2^N - 1)$ тактов. Кроме того, недостатком АЦП этого типа является использование в них конденсаторов, параметры которых зависят от внешних факторов (температура, влажность и пр.), что сказывается на точности преоб-

разования.

Метод поразрядного кодирования

Этот метод основан на том, что входное аналоговое напряжение последовательно сравнивается с суммой эталонных напряжений, имеющих значение $\Delta_{\text{кв}}2^i$, где $i = 0, 1, 2, \dots, N-1$. Таким образом, два соседних значения эталонного напряжения отличаются по величине в два раза. На первом такте входное напряжение сравнивается с максимальным эталонным напряжением.

Если входное напряжение меньше эталонного, то старший разряд выходного кода равен нулю. На следующем такте производится сравнение входного напряжения с суммой максимального эталонного напряжения и эталонного, но имеющего в два раза меньшее значение.

Если входное напряжение больше или равно эталонному, то старший разряд кода равен единице. На следующем такте производится сравнение входного напряжения с суммой максимального и эталонного напряжения и эталонного, но имеющего в два раза меньшее значение.

Процесс подобного сравнения продолжается до тех пор, пока входное напряжение не будет равно сумме эталонных напряжений, которым соответствуют единицы в разрядах выходного двоичного кода.

В аналого-цифровых преобразователях поразрядного сравнения (взвешивания) или последовательных приближений (рис. 5.5, где РПП – регистр последовательных приближений) на первом такте преобразования единица устанавливается в старшем разряде. Если $s(k\Delta t) < u_0$, выходной сигнал компаратора не изменяется и единица в старшем разряде сохраняется. В противном случае единица в старшем разряде заменяется на ноль и начинается следующий такт преобразования, в котором единица предварительно устанавливается в следующем разряде. Таким образом процесс последовательного приближения продолжается до тех пор, пока не будут определены значащие цифры всех разрядов. Схема, которая определяет, оставить ли в данном разряде единицу или заменить ее на ноль, содержит компаратор и узел формирования последовательности опроса (счетчик и регистр). В аналого-цифровых преобразователях этого типа в течение каждого такта определяется значащая цифра одного разряда, начиная со старшего. Сигналы с выхода компаратора на установку (хранение) или сброс единицы на каждом этапе преобразования определяют значащую цифру соответствующего двоичного разряда. Следовательно, его выход представ-

ляет собой цифровой эквивалент входного напряжения в последовательном коде (рис. 5.5).

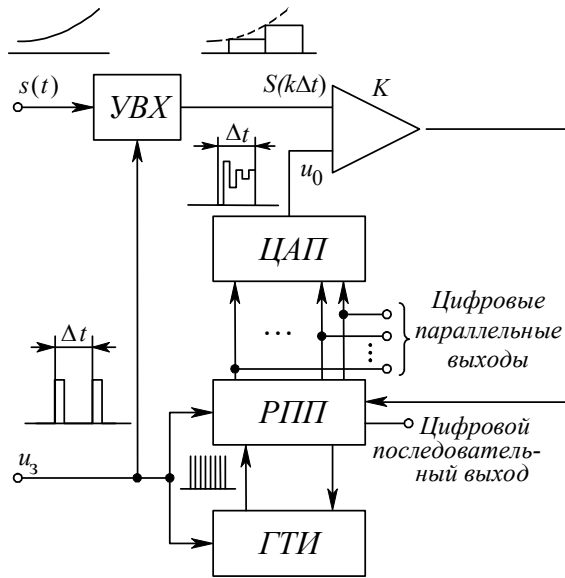


Рис. 5.5. АЦП последовательных приближений

Можно показать (табл.5.1), что для обеспечения разрешающей способности N разрядов требуется провести не более N последовательных приближений. Поскольку каждое приближение выполняется в течение одного такта, длительность цикла преобразования такого преобразователя довольно мала.

Таблица 5.1.

№ такта	Значение кода	Результат сравнения ($u_a \geq u_0$)	Десятичный эквивалент кода	Выходной код
1	1000	да	8	1000
2	1100	нет	12	1000
3	1010	да	10	1010
4	1011	нет	11	1010

Время преобразования аналого-цифрового преобразователя поразрядного взвешивания равно:

$$t_{\text{пр}} = \Delta t_{\text{мин}} = (N + 1)/f_T,$$

а величина $f_T = 1/\tau$ выбирается из условия

$$\tau \geq \tau_{\text{рпп}} + \tau_{\text{цап}} + \tau_{\text{к}} + \tau_{\text{гти}},$$

где $\tau_{\text{рпп}}$ – время установления регистра последовательных приближений.

Выигрыш в быстродействии АЦП рассмотренного типа по отношению к аналого-цифровым преобразователям, использующим метод последовательного счета, например, при значении $N = 10$ для АЦП, построенных на основе КМОП-технологии, достигает одного порядка, что дает возможность увеличить частоту дискретизации f_d до 100 кГц . Однако это увеличение быстродействия достигается за счет усложнения схемы. Кроме того, поскольку в начале каждого цикла преобразования регистр последовательных приближений сбрасывается в ноль и весь цикл преобразования повторяется, то, в отличие от следящего аналого-цифрового преобразователя, при медленных изменениях уровня входного сигнала не будет наблюдаться повышения быстродействия.

Аналого-цифровой преобразователь поразрядного взвешивания является самым универсальным из рассмотренных типов АЦП и обеспечивает представление сигнала как в параллельном, так и в последовательном коде.

Метод считывания

Метод основан на сравнении входного аналогового напряжения со всеми $(2^N - 1)$ значениями эталонного напряжения, равными $\Delta_{\text{кв}}i$, где $i = 0, 1, \dots, 2^N$. Для осуществления такого сравнения необходимо иметь $2^N - 1$ сравнивающих устройств. Результатом преобразования является параллельный код, образующийся на выходе сравнивающих устройств, который с помощью кодопреобразователя преобразует в выходной двоичный код.

Аналого-цифровые преобразователи, использующие этот метод, называются АЦП непосредственного сравнения (параллельными АЦП) и являются наиболее быстродействующими. В таких аналого-цифровых преобразователях для каждого уровня квантования используются отдельные источник опорного напряжения и компаратор. Выходы компараторов соответствующим образом соединяются с кодирующим устройством, которое формирует выходной двоичный параллельный код.

На рис. 5.6 приведена структурная схема аналого-цифрового преобразователя непосредственного сравнения, где обозначено: K – компараторы напряжения; KL – кодирующая логика; $ЦУВХ$ – цифровое устройство выборки и хранения. Для достижения разрешающей способности в N двоичных разрядах требуется $(2^N - 1)$ компараторов. Поэтому сложность аналого-цифрового преобразователя непосредственного сравнения возрастает по мере увеличения количества разрядов, но зато все преобразование может быть выполнено за один такт, так как все разрядные коэффициенты опре-

деляются одновременно. Обычно для аналого-цифрового преобразователя непосредственного сравнения величина N выбирается не более $N \leq 6 \dots 8$.

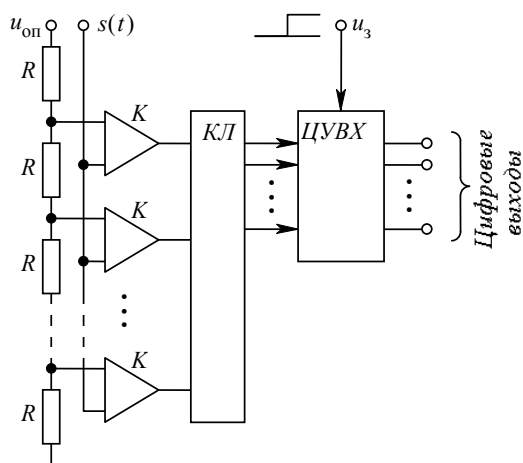


Рис. 5.6. Аналого-цифровой преобразователь непосредственного сравнения

Последовательно-параллельные АЦП строятся по последовательной двухкаскадной схеме и содержат в каждом каскаде N_1 -разрядный параллельный аналого-цифровой преобразователь ($N_1 = N/2$), причем в первом каскаде формируются старшие разряды N -разрядного цифрового кода, а во втором – младшие разряды. Выходной цифровой код первого каскада преобразуется с помощью цифроаналогового преобразователя в аналоговое напряжение и поступает на устройство вычитания, на другой вход которого подается входной сигнал $s(t)$. Напряжение разности этих двух колебаний преобразуется во втором каскаде в N_1 младших разрядов выходного цифрового кода. Последовательно-параллельные АЦП, построенные на основе ТТЛ и ТТЛШ-структур, обеспечивают частоту дискретизации $f_d = 1 \dots 2 \text{ МГц}$ при числе разрядов $N = 8 \dots 12$.

Одним из основных факторов, снижающих эффективность цифровых устройств обработки сигналов, является систематическая погрешность аналогово-цифрового преобразования входного сигнала $s(t)$, состоящая из погрешности, связанной с дискретизацией непрерывного колебания, и погрешности квантования выборочных значений по уровню. Остановимся подробнее на анализе погрешности квантования сигнала $s(t)$ по уровню. При этом необходимо рассмотреть два основных случая:

- квантование процесса, предварительно дискретизированного во времени и преобразованного с помощью устройства выборки и хранения в

последовательность импульсов прямоугольной формы длительности Δt , амплитуды которых соответствуют выборочным значениям $s(t)$ (последовательные АЦП);

- выбор значений (в моменты времени, кратные Δt) ранее квантованного сигнала $s(t)$ (параллельные АЦП);

В первом случае шум квантования $\varepsilon(t)$ может быть определен по формуле

$$\varepsilon(t) = f_s(t) - f_s^{(n)}(t),$$

где $f_s^{(n)}(t)$ – ступенчато изменяющаяся функция, которая, в отличие от $f_s(t)$, может принимать только конечное число $l = 2^N - 1$ различных значений (уровней квантования), причем переход от одного значения к другому происходит в моменты времени $t = k\Delta t$ (рис.5.7). Будем считать, что значения $\varepsilon(k\Delta t)$ представляют собой независимые случайные величины, равномерно распределенные на интервале $\pm\Delta_{\text{кв}}/2$ для каждого значения k ($k = 0, 1, \dots$). Тогда плотность распределения $\varepsilon(k\Delta t)$ имеет вид:

$$W(x) = 1/\Delta_{\text{кв}} \quad (x \leq \Delta_{\text{кв}}/2),$$

где $\Delta_{\text{кв}} = u_{\text{оп}}/(2^N - 1)$; $u_{\text{оп}}$ – максимальное значение опорного напряжения ЦАП. Можно показать, что энергетический спектр $G(\omega)$ импульсного случайного процесса $\varepsilon(t)$, описывается выражением:

$$G(\omega) = 2(\Delta t) \frac{\sin^2\left(\frac{\omega\Delta t}{2}\right)}{\left(\frac{\omega\Delta t}{2}\right)^2} \sigma^2,$$

где σ^2 – дисперсия импульсного случайного процесса $\varepsilon(t)$, которая может быть представлена в виде

$$\sigma^2 = \frac{1}{\Delta_{\text{кв}}} \int_{-\Delta_{\text{кв}}/2}^{+\Delta_{\text{кв}}/2} x^2 dx = \frac{\Delta_{\text{кв}}^2}{12}.$$

Дисперсия σ^2 определяет среднюю мощность шума квантования последовательности импульсов $f_s(t)$, амплитуды которых соответствуют выборочным значениям входного сигнала $s(t)$.

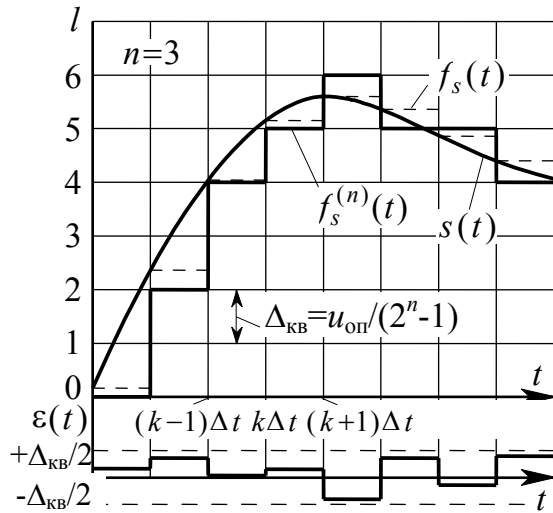


Рис. 5.7. Шум квантования при последовательном аналого-цифровом преобразовании

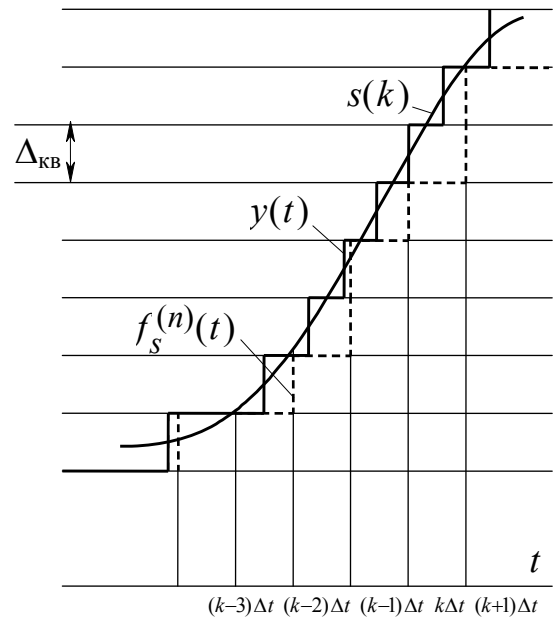


Рис. 5.8. Погрешность квантования при параллельном аналого-цифровом преобразовании

Во втором случае, при параллельном аналого-цифровом преобразовании, ступенчато изменяющаяся функция $y(t)$ (рис.5.8) представляет собой аналоговый эквивалент последовательности N -разрядных двоичных чисел, которые формируются на выходах цифрового устройства выборки и хранения параллельного АЦП (рис.5.6). В моменты времени $k\Delta t$ на N выходах ЦУВХ появляются цифровые коды (в параллельном коде), представляющие собой выборочные значения из последовательности N -разрядных двоичных чисел и соответствующие выборочным значениям сигнала. При этом шум квантования $\varepsilon(t)$, так же как и в предыдущем случае, будет импульсным случайным процессом, энергетический спектр $G(\omega)$ и σ^2 которого определяются полученными ранее выражениями.

Если предположить, что шум квантования представляет собой случайный процесс, некоррелированный с входным сигналом и канальным шумом (атмосферные помехи, тепловые шумы входных каскадов приема и обработки сигналов), то все АЦП можно рассматривать как второй источник шума, обусловленного квантованием по уровню. Пусть максимальное значение u_{\max} входного сигнала, которое может преобразовываться в аналого-цифровом преобразователе, удовлетворяет неравенству:

$$u_{\max} \geq 3\sqrt{N_0 F} + A_{\max},$$

где $N_0/2$ – спектральная плотность мощности аддитивного канального шума $n(t)$, в качестве которого рассматривается нормальный белый шум;

F – полоса частот, занимаемая случайным процессом на входе АЦП; A_{\max} – максимальное значение амплитуды сигнала $s(t)$. Тогда, очевидно, наибольшее влияние на величину погрешности аналого-цифрового преобразования сигналов будут оказывать погрешности, связанные с конечным значением шага квантования:

$$\Delta_{\text{кв}} = u_{\text{max}} / (2^N - 1).$$

Заметим, что в реальных аналого-цифровых преобразователях $u_{\text{max}} \approx u_{\text{оп}}$.

Выбор числа уровней квантования АЦП в рассматриваемых условиях зависит от соотношения между уровнем канального шума и шагом квантования, определяемого величиной v^2

$$v^2 = \Delta_{\text{кв}}^2 / (N_0 F).$$

Дисперсия шума на выходе аналого-цифрового преобразователя σ_0^2 для значений $v^2 \leq 1$ будет вычисляться по формуле:

$$\sigma_0^2 = N_0 F + \Delta_{\text{кв}}^2 / 12.$$

Из этого выражения можно найти относительное увеличение дисперсии шума на выходе АЦП за счет эффектов квантования.

Таким образом, разрядность аналого-цифрового преобразователя определяется, исходя из допустимого уровня шума квантования и требуемого динамического диапазона обрабатываемого входного сигнала $s(t)$.

Пример 15. При заданном отношении сигнал/шум на выходе АЦП, определить необходимое число разрядов при соотношении уровня канального шума и шага квантования, равного единице.

Пусть отношение сигнал/шум на входе АЦП равно $D_{\text{вх}} = 40 \text{ дБ}$, а величина $v^2 = 1$. Тогда отношение сигнал/шум на выходе аналого-цифрового преобразователя

$$D_{\text{вых}} = 10 \lg \frac{A_{\text{max}}^2}{\Delta^2 + \Delta^2 / 12} = 10 \lg \frac{A_{\text{max}}^2}{\Delta^2} - 10 \lg \left(1 + \frac{1}{12} \right) = (D_{\text{вх}} - 0,35) \text{ дБ}.$$

Как видно из этого выражения, уменьшение отношения сигнал/шум на выходе АЦП не будет превышать $0,35 \text{ дБ}$ при $v^2 = 1$. Величина $D_{\text{вх}}$ может быть представлена в виде:

$$D_{\text{вх}} = 10 \lg \frac{A_{\text{max}}^2}{\Delta^2} = 10 \lg (2^N - 1)^2 + \lg \frac{A_{\text{max}}^2}{u_{\text{max}}^2}.$$

При значении отношения сигнал/шум $D_{\text{вх}} = 40 \text{ дБ}$ число разрядов в АЦП должно быть не менее 7. Обычно с учетом удобства реализации аналого-

цифрового преобразователя на интегральных схемах, выбирают число разрядов, равное восьми.

5.3 Интегральные схемы аналого-цифровых преобразователей

Рассмотрим основные характеристики интегральных схем аналого-цифровых преобразователей последовательных приближений, непосредственного сравнения (параллельные) и интегрирующих (косвенного преобразования) [1, 10, 14, 24].

Аналого-цифровыми преобразователями последовательных приближений являются микросхемы К572ПВ1, К1107ПВ1, К1108ПВ1, К1108ПВ2 и К1113ПВ1. При подключении внешних компаратора, генератора тактовых импульсов и источника опорного напряжения МОП-микросхема К572ПВ1 может выполнять функции 12-разрядного АЦП последовательных приближений с временем преобразования 110 мкс. В состав ИС входят устройства побайтового обмена данными с 8-разрядной шиной данных микропроцессора.

Микросхема К1113ПВ1 представляет собой 10-разрядный аналого-цифровой преобразователь и содержит все, кроме устройства выборки и хранения, функциональные узлы АЦП последовательных приближений, включая источник опорного напряжения, компаратор и тактовый генератор. Для практического применения интегральной схемы необходимы два источника питания и несколько резисторов. Поскольку интегральная схема специально разрабатывалась для микропроцессорных систем, то она снабжена выходными каскадами с тремя состояниями, позволяющими считать результаты преобразования непосредственно на шину данных микропроцессоров. Время преобразования для этой микросхемы составляет 30 мкс.

Более быстродействующими функционально полными, аналого-цифровыми преобразователями являются микросхемы серии К1108: десятиразрядный АЦП К1108ПВ1 с временем преобразования 0,8 мкс (при работе в 8-разрядном режиме менее 0,5 мкс) и двенадцатиразрядный

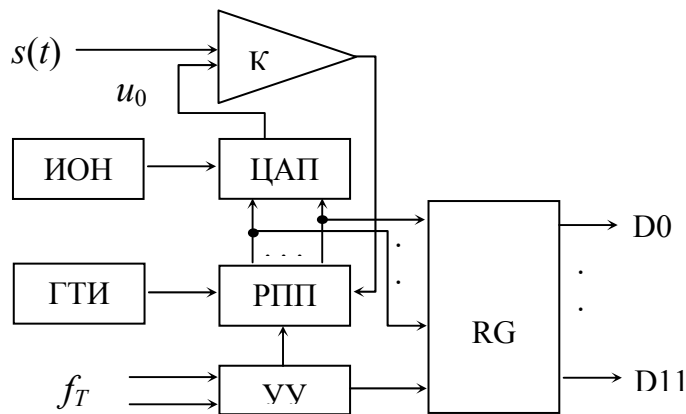


Рис. 5.9. Структурная схема аналого-цифрового преобразователя 1108PB2

АЦП К1108PB2 с временем преобразования 2,0 мкс. Интегральная схема К1108PB2, структурная схема которой дана на рис.5.9, представляет собой функционально полный аналого-цифровой преобразователь. В состав интегральной схемы входят: компаратор (К), цифроаналоговый преобразователь (ЦАП) и регистр последовательных приближений (РПП), источник опорного напряжения (ИОН), генератор тактовых импульсов (ГТИ). Кроме того, в состав ИС входит выходной буферный регистр (RG) с тремя состояниями на выходе, схемы управления (УУ) работой АЦП и выходного регистра. Наличие регистра с тремя состояниями позволяет применять АЦП в устройствах ввода-вывода микропроцессорных систем. Интегральные схемы параллельных АЦП выпускаются в составе серии ЭСЛ К1107. Микросхемы К1107PB1...PB5 отличаются друг от друга разрядностью ($N = 6...8$) и временем преобразования (20...100 нс) [10, 14, 24]. Основными областями применения этих интегральных схем являются устройства радиолокации, цифрового телевидения, телекоммуникационные устройства передачи цифровых сообщений и др.

Аналого-цифровой 6-разрядный параллельный преобразователь К1107PB1 предназначен для преобразования сигналов, амплитуда которых изменяется от 0 до -2,0 В. Он имеет максимальную частоту преобразования 20 МГц. Структурная схема АЦП приведена на рис. 5.10, обозначение выводов интегральной схемы в таблице 5.2, а принципиальная схема включения – на рис. 5.11. выходного регистра. Наличие регистра с тремя состояниями позволяет применять АЦП в устройствах ввода-вывода микропроцессорных систем.

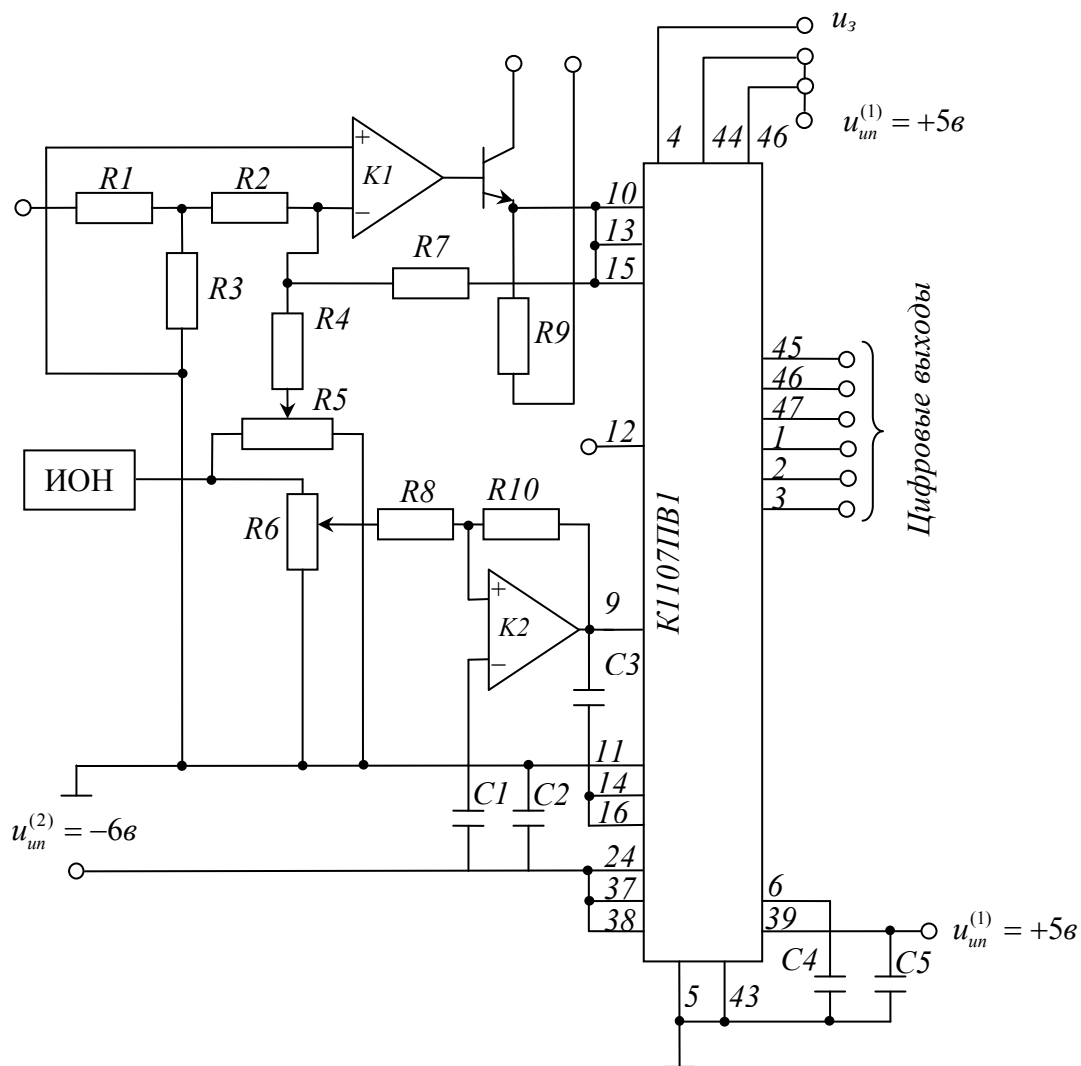


Рис. 5.11. Принципиальная схема включения аналого-цифрового преобразователя 1107PB1

В состав интегральной схемы (рис.5.10) входят:

- резисторный делитель эталонных напряжений по схеме последовательного делителя напряжений. На входы делителя подаются опорные напряжения, которые компенсируют напряжение смещения нуля характеристики преобразования и погрешность преобразования в конечной точке шкалы ($\delta_{\text{шт}}$);
- 64 стробируемых компаратора;
- преобразователь кода с выходов компараторов в шестиразрядный двоичный код;
- совокупность логических элементов, выполняющих функцию «исключающее ИЛИ», с помощью которых выбирается тип выходного ко-

да АЦП (двоичный, обратный двоичный, дополнительный, обратный дополнительный);

- выходной буферный регистр.

Кроме того, в аналого-цифровой преобразователь встроены преобразователи уровней входных напряжений транзисторно-транзисторной логики в уровни эмиттерно-связанной логики и уровней выходных напряжений эмиттерно-связанной логики в уровни транзисторно-транзисторной логики.

Последовательность тактовых импульсов имеет частоту следования импульсов $f_T = 20 \text{ МГц} \pm 1\%$ при уровне напряжения логической единицы равном $+2,4 \dots 5 \text{ В}$ и уровне напряжения логического нуля равном $0 \dots 0,4 \text{ В}$. Длительность импульса необходимо иметь $15 \text{ нс} \pm 20\%$ при длительности фронта импульса $\tau_\phi < 6 \text{ нс}$ и спада – $\tau_C < 6 \text{ нс}$.

Таблица 5.2.

Обозначение вывода	Назначение вывода
1	Выход 4
2	Выход 5
3	Выход 6 (MP)
4	Тактовый сигнал
5,43	Общий (цифровая «земля»)
8,39	Напряжение питания $u_{ин}^{(1)} = +5,0 \text{ в}$
9	Опорное напряжение $u_{он} = -2,0 \text{ в}$
10,13,15	Вход (аналоговый сигнал)
11,14	Общий (аналоговая «земля»)
12	Вывод корректировки нелинейности
16	Опорное напряжение $u_{он} = 0 \text{ в}$
24,37,38	Напряжение питания $u_{ин}^{(2)} = -6,0 \text{ в}$
44	Управление выходным кодом. Вход 1
45	Выход 1 (CP)
46	Выход 2
47	Выход 3
48	Управление выходным кодом. Вход 2

Интегрирующий аналого-цифровой преобразователь (разновидность АЦП косвенного преобразования) может быть выполнен по КМОП технологии (микросхема 572ПВ2). Аналого-цифровой преобразователь дополня-

ется источником опорного напряжения, несколькими резисторами и конденсаторами. Особенностью этой микросхемы является представление цифровых данных на выходе в специальном коде, предназначенном для непосредственного управления 3,5-декадными цифровыми дисплеями с 7-сегментными полупроводниковыми индикаторами. Для работы с жидкокристаллическими индикаторами разработана интегральная схема К572ПВ5, являющаяся по функциональному назначению, схемотехническому и технологическому исполнению аналогом ИС К572ПВ2.

Широкое распространение 8-разрядных микропроцессоров вызвало появление таких интегральных схем аналого-цифровых преобразователей, как К572ПВ3 и К572ПВ4. Первый из них имеет два аналоговых входа, соединенных резисторами с входом компаратора, что позволяет осуществить преобразование суммы напряжений, подаваемых на входы. Эта микросхема является АЦП последовательных приближений с время преобразования 15 мкс, сопрягаемая с микропроцессором.

Микросхема К572ПВ4 представляет собой аналого-цифровое устройство сбора данных и предназначена для преобразования аналоговых сигналов, поступающих по восьми входам, в цифровой код с последующим его хранением в оперативном запоминающем устройстве и считыванием во внешний микропроцессор в режим прямого доступа к памяти. С помощью аналоговых электронных ключей встроенного коммутатора, восемь входных аналоговых сигналов поочередно подаются на вход АЦП последовательных приближений с временем преобразования 32 мкс. Результаты преобразования записываются во встроенный ОЗУ объемом 8 однобайтовых слов. Каждому из входных сигналов выделяется своя область памяти, так что после одного цикла работы коммутатора, в ОЗУ оказываются записанными выборочные значения всех сигналов. Если одной входной аналоговый сигнал подать, непосредственно в ОЗУ, то в память будут записываться коды, соответствующие восьми последовательным значениям входного сигнала. Считывание данных из оперативного запоминающего устройства можно осуществлять в произвольном порядке.

Перспективным направлением создания интегральных схем является построение микроэлектронных систем обработки сигналов на базе микропроцессоров или микроконтроллеров со встроенными АЦП и ЦАП. Примером такой системы является микросхема КМ1813ВЕ1.

Тенденции развития и создания аналого-цифровых преобразователей являются следующие:

- разработка многоканальных (2, 4, 16, 32 канала) АЦП с разрядностью $N=10\dots 16$, малым потреблением тока и высокими частотами преобразования 250–400 кГц;
- разработка АЦП с высокой разрядностью ($N=14$) и высокой частотой преобразования (до 400 кГц), встраиваемых в персональные компьютеры.
- улучшение основных характеристик АЦП: уменьшение времени преобразования, повышение их точность и др. В частности, построение 8-разрядных АЦП с тактовыми частотами до 350–1000 МГц и шириной полосы частот входных сигналов до 100 МГц, 250 МГц, а также АЦП на основе GaAs с тактовыми частотами до 3 ГГц.

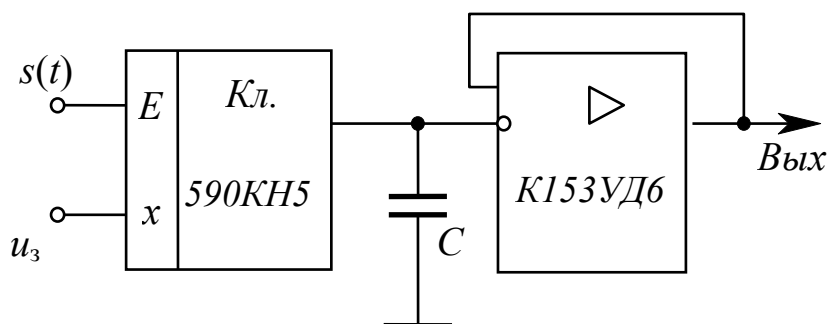


Рис. 5.12. Устройство выборки и хранения КР1100СК3

Важным узлом интегральных схем АЦП, существенным образом влияющим на быстродействие и точность преобразователя, является устройство выборки и хранения. В устройстве выборки и хранения КР1100СК3 (рис.5.12), состоящей из ключа, конденсатора C и операционного усилителя, выходным элементом является операционный усилитель. Время разряда конденсатора C определяется входным током операционного усилителя, дрейфом нуля ОУ, величиной тока утечки электронного ключа и заданной точностью аналого-цифрового преобразования.

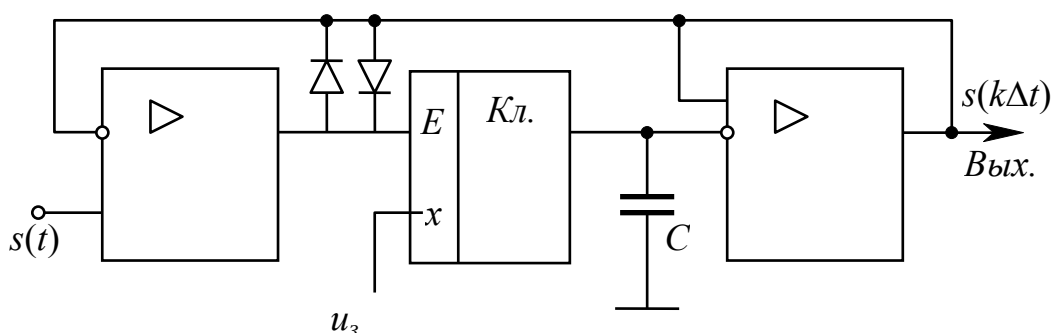


Рис. 5.13. Устройство выборки и хранения КР1100СК2

Микросхема КР1100СК2 (рис. 5.13) содержит два операционного усилителя с высоким входным сопротивлением (более 10 Мом), ключевую схему управления, обеспечивающую токовое управление электронными ключами, и конденсатор C емкостью $20 - 1000\text{ нФ}$, подключаемый к соответствующим выводам микросхемы. Представленные на рис. 5.12 и 5.13 устройства выборки и хранения выполняются в интегральном исполнении, либо входят в состав интегральных схем аналого-цифровых преобразователей.

Глава 6. Радиотехнические устройства формирования, приема и передачи сигналов

Рассмотренные принципы построения комбинационных и последовательностных устройств, цифроаналоговых и аналого-цифровых преобразователей являются основой для разработки более сложных узлов радиотехнических устройств, выполняющих разнообразные функции обработки сигналов в системах передачи информации, связи, радиолокации, телекоммуникаций, радионавигации. Цифровые радиотехнические устройства используются для формирования и преобразования различных сигналов на малом уровне мощности радиопередатчиков, в корреляторах и согласованных фильтрах, в системах фазовой автоподстройки частоты, в радиомодемах цифровых систем мобильной связи, в устройстве сбора данных и пространственного мониторинга, устройствах охранных и пожарных сигнализаций и др.

6.1. Генераторы случайных последовательностей

Такие генераторы используются для формирования псевдослучайных двоичных последовательностей, применяемых, например, в качестве тестовых цифровых последовательностей в устройствах формирования сигналов систем мобильной связи. Генераторы могут быть построены на основе n -разрядных сдвигающих регистров на D -триггерах (рис.6.1). С помощью такого регистра можно генерировать циклические псевдослучайные последовательности длиной $L \leq 2^n$. Как видно из рис.6.1, имеется несколько различных вариантов подключения обратных связей с выходов Q_0 , Q_1 и Q_2 к входам логических элементов. Каждый вариант подключения обеспечивает свою форму последовательности трех разрядных чисел. В представленном на рисунке варианте, последовательности длиной $L = 8$ имеют вид:

- на выходе Q_0 формируется последовательность 00011101;
- на выходе Q_1 формируется последовательность 00111011;
- на выходе Q_2 формируется последовательность 01110100.

На вход C такого генератора случайных последовательностей (рис. 6.1) поступают тактовые импульсы с частотой следования f_T .

Для построения помехозащищенных систем передачи дискретных сообщений, например, с использованием сигналов с перестройкой несущей частоты или сигналов на основе частотно-временных матриц, используются генераторы случайных последовательностей, имеющих большой период повторения и хорошие свойства автокорреляционных и взаимокорреляционных функций [18]. Такие двоичные последовательности максимальной длины или

псевдослучайные M -последовательности формируются с помощью n -разрядных регистров сдвига, охваченных линейной обратной связью. Подобная M -последовательность имеет наибольший период следования $L = 2^n - 1$ благодаря тому, что используются все, кроме нулевого, состояния регистра.

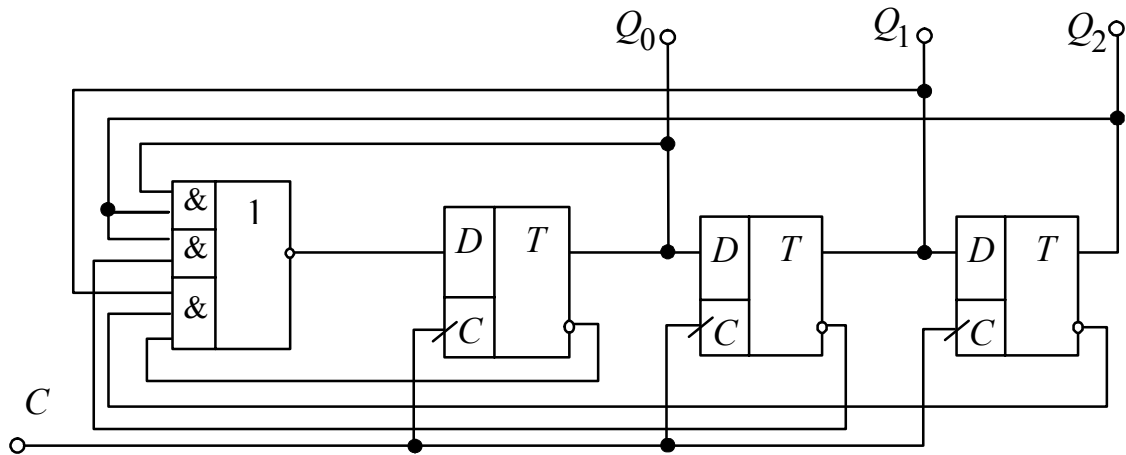


Рис. 6.1. Генератор случайных последовательностей на D -триггерах ($n = 3$)

Задача построения структуры логической обратной связи решается с помощью полинома вида:

$$f(x) = \alpha_0 x^n + \alpha_1 x^{n-1} + \dots + \alpha_{n-1} x + \alpha_n. \quad (6.1)$$

Двоичное $(n+1)$ -разрядное число $\alpha_0 \alpha_1 \dots \alpha_n$, полученное из значений коэффициентов $\{\alpha_k\}_{k=0,1,\dots,n} = 0$ или 1 полинома (6.1), расположенных с учетом веса соответствующего им члена полинома, определяет вид $f(x)$ и форму двоичной M -последовательности. В табл.6.1 в качестве примера приведены полиномы в восьмеричном коде (M_{103} , M_{147} , ..., M_{133}) для значения $n = 6$ [30]. Полиному вида (6.1) соответствует рекуррентная формула.

$$\alpha_0 d_j = \alpha_n d_{j-n} \oplus \alpha_{n-1} d_{j-(n-1)} \oplus \dots \oplus \alpha_2 d_{j-2} \oplus \alpha_1 d_{j-1}, \quad (6.2)$$

где d_j – двоичный j -й элемент, образуемый сложением по модулю 2 некоторого числа предшествующих элементов, хранящихся в регистре, коэффициенты при которых равны единице [18]. Эта формула определяет алгоритм формирования M -последовательности.

Таблица 6.1.

Восьмеричный код M	Двоичный код $\alpha_0\alpha_1\dots\alpha_6$	Уравнение $f(x)$
103	1 000 011	$x^6 + x + 1$
147	1 100 111	$x^6 + x^5 + x^2 + x + 1$
155	1 101 101	$x^6 + x^5 + x^3 + x^2 + 1$
141	1 100 001	$x^6 + x^5 + 1$
163	1 110 011	$x^6 + x^5 + x^4 + x + 1$
133	1 011 011	$x^6 + x^4 + x^3 + x + 1$

Генератор таких псевдослучайных M -последовательностей ($n=6$) состоит из 6-разрядного регистра сдвига, блока 2-входовых сумматоров по модулю 2 и мультиплексора (рис. 6.2). На вход C регистра сдвига поступают тактовые импульсы с частотой следования f_T . Входы $D_1\dots D_6$ регистра сдвига служат для записи начальной двоичной комбинации символов в регистр сдвига при условии, что на входе V присутствует логическая единица. Когда на входе V присутствует логический ноль, входы $D_1\dots D_6$ блокируются, а регистр сдвига переходит в режим сдвига записанной двоичной комбинации символов. На каждом такте комбинация сдвигается на один разряд в сторону старшего разряда, а новые символы последовательности, образованные в блоке сумматоров, через мультиплексор MS поступают на вход D_0 первого разряда регистра сдвига [18].

Структура блока сумматоров построена таким образом, что реализуется рекуррентная формула (6.2) с набором коэффициентов $\{\alpha_k\}$, соответствующих полиномам степени $n=6$ (Табл.6.1). Например, восьмеричному коду 163 соответствует рекуррентная формула вида:

$$d_j = d_{j-6} \oplus d_{j-5} \oplus d_{j-4} \oplus d_{j-1}, \quad (6.3)$$

Из (6.3) следует, что j -й символ последовательности M_{163} формируется сигналами с выходов Q_6, Q_5, Q_4, Q_1 регистра сдвига RG . На выходе M_{163} блока сумматоров формируются символы псевдослучайной M -последовательности. Применяв свойство ассоциативности операции сложения по модулю 2, получим:

$$M_{163} = (Q_6 \oplus Q_5) \oplus (Q_4 \oplus Q_1). \quad (6.4)$$

Для реализации (6.4) требуется три 2-входовых сумматора по модулю 2,

объединенных в 2-ступенчатую структуру.

Число выходов блока сумматоров равно числу формируемых псевдослучайных M -последовательностей.

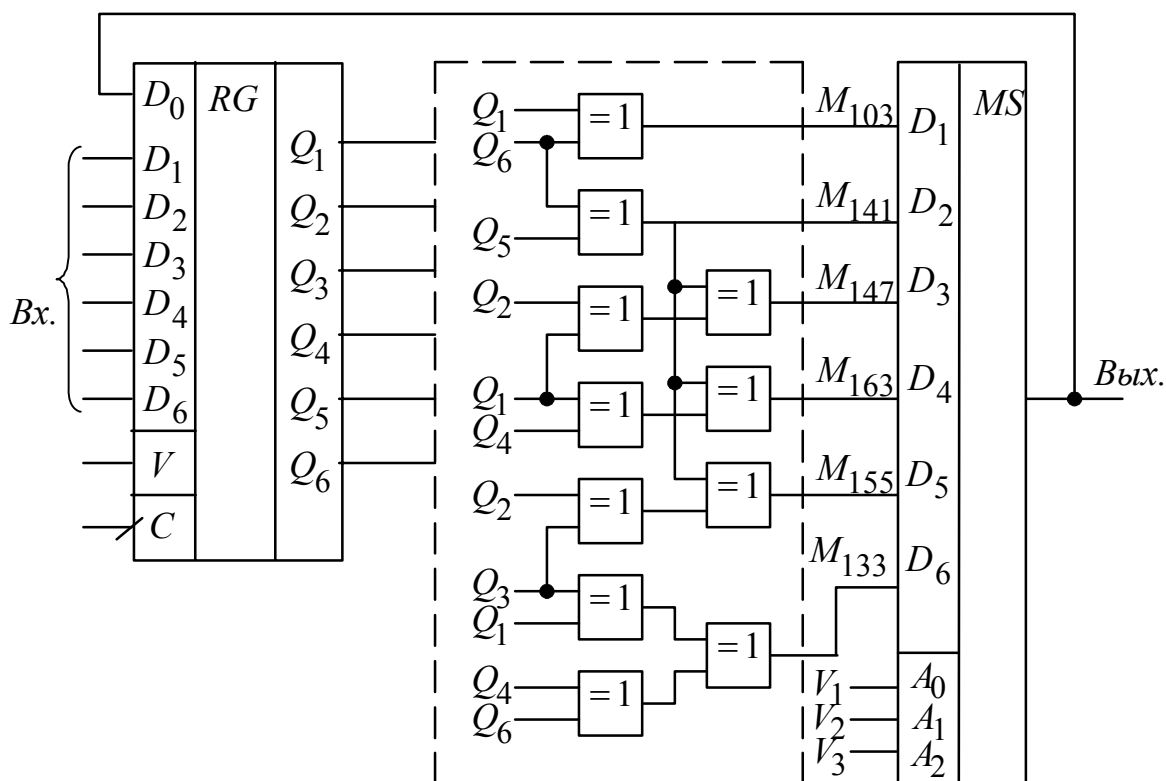


Рис. 6.2 Генератор M -последовательностей

Мультиплексор (рис.6.2) служит для подключения одного из выходов блока сумматоров к входу D_0 регистра сдвига, а именно того выхода, на котором формируется M -последовательность. Таким образом с помощью мультиплексора происходит коммутация логической обратной связи регистра сдвига. Управление процедурой коммутации осуществляется трехэлементным двоичным кодом $V_3V_2V_1$, каждому варианту которого, кроме нулевого и равного семи, соответствует один из шести вариантов коммутации:

- последовательности M_{103} соответствует двоичный код $\bar{V}_3\bar{V}_2V_1$
- последовательности M_{133} соответствует двоичный код $V_3V_2\bar{V}_1$
- последовательности M_{141} соответствует двоичный код $\bar{V}_3V_2\bar{V}_1$ (6.5)
- последовательности M_{147} соответствует двоичный код $\bar{V}_3V_2V_1$
- последовательности M_{155} соответствует двоичный код $V_3\bar{V}_2V_1$
- последовательности M_{163} соответствует двоичный код $V_3\bar{V}_2\bar{V}_1$

Например, при наличии на входах $V_3V_2V_1$ двоичного кода 001, на выход

мультиплексора коммутируется выход блока сумматоров M_{103} и генератор формирует M -последовательность, соответствующую уравнению $x^6 + x + 1$.

6.2 Устройства формирования последовательностей Уолша

Последовательности Уолша применяются в системах формирования и обработки ортогональных и широкополосных сигналов, а также в устройствах цифрового спектрального анализа в качестве опорных частот цифровых генераторов.

В основе принципов формирования последовательностей лежат алгоритмы синтеза функций Уолша. Свойства этих функций и возможные области их практических приложений изложены, например, в [10].

Функции Уолша определяются на интервале $[0,1]$ или $[-0,5, 0,5]$ значений переменной $\Theta=t/T$, которую можно рассматривать как время, нормированное на заданном интервале ортогональности T . Этот интервал равен, например, длительности ортогональных и широкополосных сигналов.

Функции Уолша $wal_j(\Theta)$, где j - номер функции, обычно задаются через систему базисных функций Радемахера:

$$r_k(\Theta) = \text{sign}[\sin(2^k \pi \Theta)], \quad r_0(\Theta) = 1, \quad (6.6)$$

где k - номер функции, целое вещественное число, принимающее значения от 1 до n .

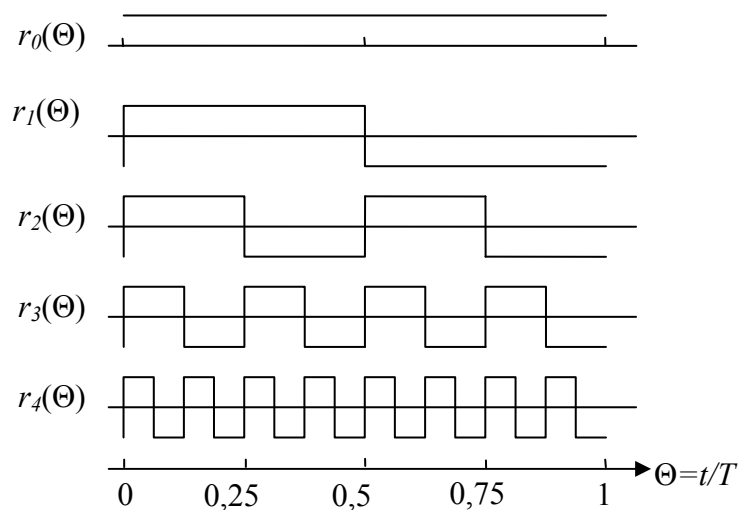


Рис. 6.3. Вид функций Радемахера

Вид функций Радемахера на интервале $[0,1]$ для $k=0, 1, 2, 3, 4$ приведен на рис.6.3. Как следует из (6.6), функции существуют на 2^k равных подинтервалах, каждый из которых формально соответствует полупериоду гармонического колебания. Это дает возможность применять понятия частоты f_r и периода T_r последовательностей Радемахера, определив их через интервал T :

$$wal_{\alpha}(\Theta) wal_{\beta}(\Theta) = wal_{\gamma}(\Theta),$$

где γ определяется сложением по модулю двух значений α и β , т.е. $\gamma = \alpha \oplus \beta$.

Номера функций Уолша представляются обычно десятичными цифрами, но для выполнения операции сложения по модулю 2 они преобразуются в двоичный код. Свойство мультипликативности позволяет дополнить систему базисных функций (6.8) их произведениями и образовать таким образом полную систему функций Уолша. Существует определенная закономерность в выборе базисных функций Радемахера для формирования заданных функции Уолша.

Рассмотрим прежде всего систему функций Уолша, полученную из четырех базисных функций Радемахера: $r_4(\Theta)$, $r_3(\Theta)$, $r_2(\Theta)$, $r_1(\Theta)$. Вид соответствующих этим функциям последовательностей приведен в табл.6.2 [10]. Операция перемножения двух последовательностей выполняется путем их поэлементного сложения по модулю 2. Так, путем перемножения двух базисных последовательностей $r_1(\Theta)$ и $r_2(\Theta)$ получается функция Уолша с номером $j=2$. Номер формируемой последовательности можно получить, сложив поразрядно по модулю 2 двоичные коды $[j]_2$, соответствующие номерам функции Уолша. Действительно, сложив поразрядно коды 0001 и 0011 для функций $r_1(\Theta)$ и $r_2(\Theta)$ получим $Wal_2(\Theta)$. Двоичные коды $[j]_2$ приведены в табл.6.2.

Алгоритм формирования функций Уолша, полученных из 4-х базисных функций Радемахера – $r_4(\Theta)$, $r_3(\Theta)$, $r_2(\Theta)$, $r_1(\Theta)$, имеет вид:

$$wal_j(\Theta) = g_4 r_4(\Theta) \oplus g_3 r_3(\Theta) \oplus g_2 r_2(\Theta) \oplus g_1 r_1(\Theta) \quad (6.10)$$

где g_4, g_3, g_2, g_1 – коэффициенты при соответствующих функциях Радемахера $\{r_k(\Theta)\}$, а $k=1, 2, \dots, 4$, коэффициенты принимающие значения 0 или 1 и составляют код, определяющий номер образуемой функции Уолша. Характер соответствия между кодами $[j]_2$ и коэффициентами $\{g_k\}$, приведенными в табл.6.2, указывает на то, что коэффициенты $\{g_k\}$ образуют код Грея $g[j]$ чисел j .

Представив номер получаемой функции Уолша двоичным кодом и преобразовав его в код Грея, из (6.10) получим базисные последовательности функций Радемахера, необходимые для формирования данной функции Уолша. Преобразование двоичного кода $\{j_n, j_{n-1}, \dots, j_k, \dots, j_2, j_1\}$ в код Грея $\{g_n, g_{n-1}, \dots, g_k, \dots, g_2, g_1\}$ производится поразрядно, начиная со старшего разряда в соответствии с равенствами:

$$g_n = j_n; g_{n-1} = j_n \oplus j_{n-1}; \dots; g_k = j_{k+1} \oplus j_k; \dots; g_1 = j_2 \oplus j_1.$$

В общем случае формирование функций Уолша осуществляется по формуле:

$$wal_j(\Theta) = g_n r_n(\Theta) \oplus g_{n-1} r_{n-1}(\Theta) \oplus \dots \oplus g_k r_k(\Theta) \oplus \dots \oplus g_2 r_2(\Theta) \oplus g_1 r_1(\Theta). \quad (6.11)$$

Структурная схема (рис. 6.4), реализующая (6.11), включает формирователь базисных последовательностей Радемахера z^{-i} , взвешивающие устройства g_i и сумматоры по модулю 2.

Таблица 6.2

j	$wal_j(\Theta)$	Функции Уолша	Код $[j]_2$			Код Грея $g(j)$			
			j_4	j_3	j_2	g_4	g_3		
			j_1		g_2	g_1			
0	0	0 1/4 1/2 3/4 1	0	0	0	0	0	0	0
1	$r_1(\Theta)$		0	0	0	1	0	0	0
3	$r_2(\Theta)$		0	0	1	1	0	0	1
2	$r_1(\Theta) \oplus r_2(\Theta)$		0	0	1	0	0	1	1
7	$r_3(\Theta)$		0	1	1	1	0	1	0
6	$r_1(\Theta) \oplus r_3(\Theta)$		0	1	1	0	0	1	0
4	$r_2(\Theta) \oplus r_3(\Theta)$		0	1	0	0	0	1	1
5	$r_1(\Theta) \oplus r_2(\Theta) \oplus r_3(\Theta)$		0	1	0	1	0	1	1
15	$r_4(\Theta)$		1	1	1	1	1	0	0
14	$r_1(\Theta) \oplus r_4(\Theta)$		1	1	1	0	1	0	0
12	$r_2(\Theta) \oplus r_4(\Theta)$		1	1	0	0	1	0	1
13	$r_1(\Theta) \oplus r_2(\Theta) \oplus r_4(\Theta)$		1	1	0	1	1	0	1
8	$r_3(\Theta) \oplus r_4(\Theta)$		1	0	0	0	1	1	0
9	$r_1(\Theta) \oplus r_3(\Theta) \oplus r_4(\Theta)$		1	0	0	1	1	1	0
11	$r_2(\Theta) \oplus r_3(\Theta) \oplus r_4(\Theta)$		1	0	1	1	1	1	0
10	$r_1(\Theta) \oplus r_2(\Theta) \oplus r_3(\Theta) \oplus r_4(\Theta)$		1	0	1	0	1	1	1

Формирователь базисных функций $\{r_k(\Theta)\}$ представляет собой цифровой делитель частоты, на вход которого подаются тактовые импульсы с частотой f_T . На выходе первого разряда делителя (триггера) формируется последова-

тельность $r_1(\Theta) = wal_1(\Theta)$, с частотой следования импульсов $f_1 = f_T/2$; на выходе второго разряда – последовательность $r_2(\Theta) = wal_3(\Theta)$, с частотой следования импульсов $f_2 = f_T/4$; и т.д. На выходе последнего n -го разряда цифрового делителя частоты формируется последовательность $r_n(\Theta) = wal_j(\Theta)$, где $j = 2^n - 1$, с частотой $f_n = f_T/2^n$.

Разрядность цифрового делителя частоты равна числу базисных последовательностей Радемахера и определяет общее число N_{wal} функций Уолша, равное $N_{wal} = 2^n - 1$. С учетом этого соотношения и выражения (6.8), получим:

$$f_T = (N_{wal} + 1)/T. \quad (6.12)$$

Формула (6.12) связывает число функций Уолша, длительность T и частоту следования тактовых импульсов. Максимальная частота тактовых импульсов ограничивается быстродействием цифрового делителя частоты (счетчика) $f_T \leq f_{сч \max}$, где $f_{сч \max}$ – максимальная частота работы двоичного счетчика, так что:

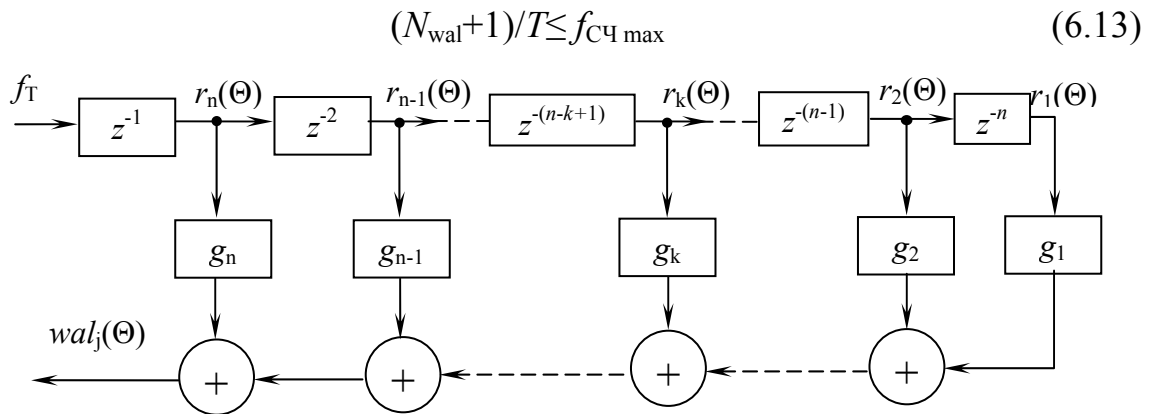


Рис. 6.4 Структурная схема формирования функций Уолша

При построении цифрового делителя частоты по схеме с параллельным переносом быстродействие устройства формирования функций Уолша определяется в основном максимальной частотой переключения триггеров. Так например, при использовании ТТЛ структур для скорости передачи двоичных сообщений $V = 1/T = 2400$ Бод, возможно формирование ансамбля из $N_{wal} = 8 \cdot 10^4$ функций Уолша. Для реализации такого устройства необходимы 16-разрядный двоичный счетчик и 15 двоичных сумматоров по модулю 2. Выбор функций Уолша определяется 16-разрядным кодом. При уменьшении длительности T , уменьшается и число формируемых функций Уолша.

Изменением кода $g[j]$ от 00...1 до 11...1 можно получить любую из 2^{n-1} функций Уолша. В устройстве, построенном по схеме, приведенной на рис.6.4, формирование функций Уолша происходит с различными времен-

ными задержками. Худшими по быстродействию являются варианты формирования функций Уолша с использованием $r_1(\Theta)$, когда сигнал проходит через все сумматоры по модулю 2. Тогда результирующая задержка появления сигнала на выходе устройства будет максимальной:

$$\tau_{\max} = \tau_{\text{СЧ}} + \tau_{\text{ЛОГ}} + \tau_{\text{М2}}(n-1) \quad (6.14)$$

где $\tau_{\text{СЧ}} = 1/f_{\text{СЧmax}}$, $\tau_{\text{ЛОГ}}$, $\tau_{\text{М2}}$ – временные задержки в двоичном счетчике, логическом элементе И и сумматоре по модулю 2 соответственно. Наименьшая задержка при формировании функций Уолша образуется для $wal_n(\Theta)$:

$$\tau_{\min} = \tau_{\text{СЧ}} + \tau_{\text{ЛОГ}} + \tau_{\text{М2}} \quad (6.15)$$

Последовательная схема соединения сумматоров приводит к снижению тактовой частоты

$$f_T \leq 1/[\tau_{\text{СЧ}} + \tau_{\text{ЛОГ}} + (n-1)\tau_{\text{М2}}] \quad (6.16)$$

На рис. 6.5,*a* приведена реализация устройства формирования функций Уолша с использованием многоразрядного двоичного счетчика и преобразователя двоичного кода в код Грея для управления работой мультиплексора. Достоинством такой структуры устройства является ее наращиваемость при необходимости изменения (увеличения) числа формируемых функций Уолша.

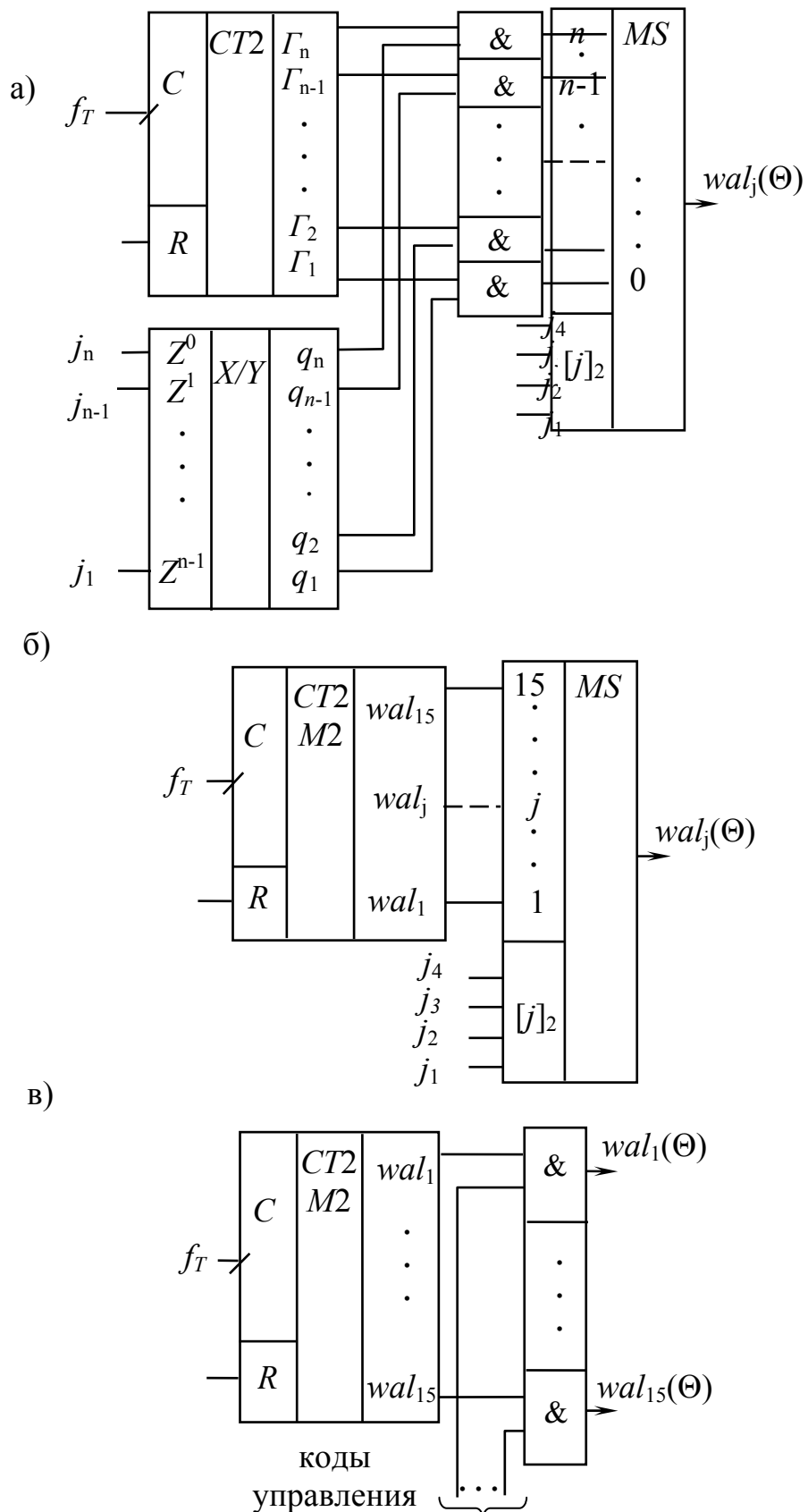


Рис. 6.5. Принципиальные схемы устройств формирования функций Уолша

Для распараллеливания процедур формирования функций Уолша, можно использовать свойство ассоциативности операции сложения по модулю 2. Например (см.табл.6.2) представим:

$$\begin{aligned} wal_{10}(\Theta) &= r_1(\Theta) \oplus r_2(\Theta) \oplus r_3(\Theta) \oplus r_4(\Theta) = \\ &= (r_1(\Theta) \oplus r_2(\Theta)) \oplus (r_3(\Theta) \oplus r_4(\Theta)). \end{aligned}$$

Операция сложения по модулю 2 выполняется в два этапа: вначале последовательности суммируются попарно, а затем та же операция выполняется над полученными результатами. Подобным образом строятся многоступенчатые сумматоры по модулю 2. Вариант реализации устройства, формирующего по указанному принципу функции Уолша, приведен на рис.6.5,б, где используется двоичный счетчик в сочетании с многоступенчатыми сумматорами по модулю 2. Для формирования нужной последовательности может быть использован мультиплексор, управляемый кодом $[j]_2$. При необходимости формирования определенного набора функций Уолша, мультиплексор заменяется коммутатором, состоящим из $2^n - 1$ логических элементов И (рис.6.5,в).

6.3 Генераторы сигналов

Значительная часть задач, связанных с созданием телекоммуникационных систем мобильной связи, относится к разработке генераторов сигналов входящих в состав модуляторов радиопередатчиков, радиомодемов, корреляционных устройств и т. п. Такие генераторы используются для формирования спектрально-эффективных сигналов или их огибающих [20], имеющих плавные законы изменения амплитуды, частоты или фазы колебания, которые широко применяются, например, в мобильных системах связи стандарта GSM; цифровых телевизионных системах DVB-C, T, S и ATSC; радиорелейных, спутниковых и проводных системах связи (Bell Northern Research, TRW, Mobile Satellite, North America Digital Cellular, POCSAG, Fujitsu) и др. Форма огибающей данных сигналов отличается от прямоугольной и может быть трапецеидального, косинусоидального, гауссовского вида, вида «приподнятого косинуса», полиномиального вида. При этом длительность формируемого сигнала может совпадать с длительностью символа канального алфавита, либо быть больше этой длительности. В последнем случае сигналы передаются при наличии так называемой управляемой межсимвольной интерференции.

Генераторы (формирователи) сигналов могут быть реализованы с помощью последовательностных устройств, запоминающих устройств, цифроана-

логовых преобразователей (Главы 2, 3, 4). Рассмотрим некоторые виды таких генераторов сигналов.

Формирование сигналов, спектр которых занимает полосу частот $[0, \Delta F]$, выполняется путем генерирования последовательностей импульсов методом амплитудно-импульсной модуляции вида:

$$u(t) = \sum_{k=0}^{p-1} s(k\Delta t) \cdot \psi(t - k\Delta t), \quad (6.17)$$

где

$$\psi(t) = \begin{cases} 1, & \text{при } 0 < t \leq \Delta t, \\ 0, & \text{при } t < 0, t > \Delta t, \end{cases}$$

p – число выборочных значений сигнала $s(t)$ на интервале формирования T_Φ ; интервал дискретизации $\Delta t \leq 1/2\Delta F$.

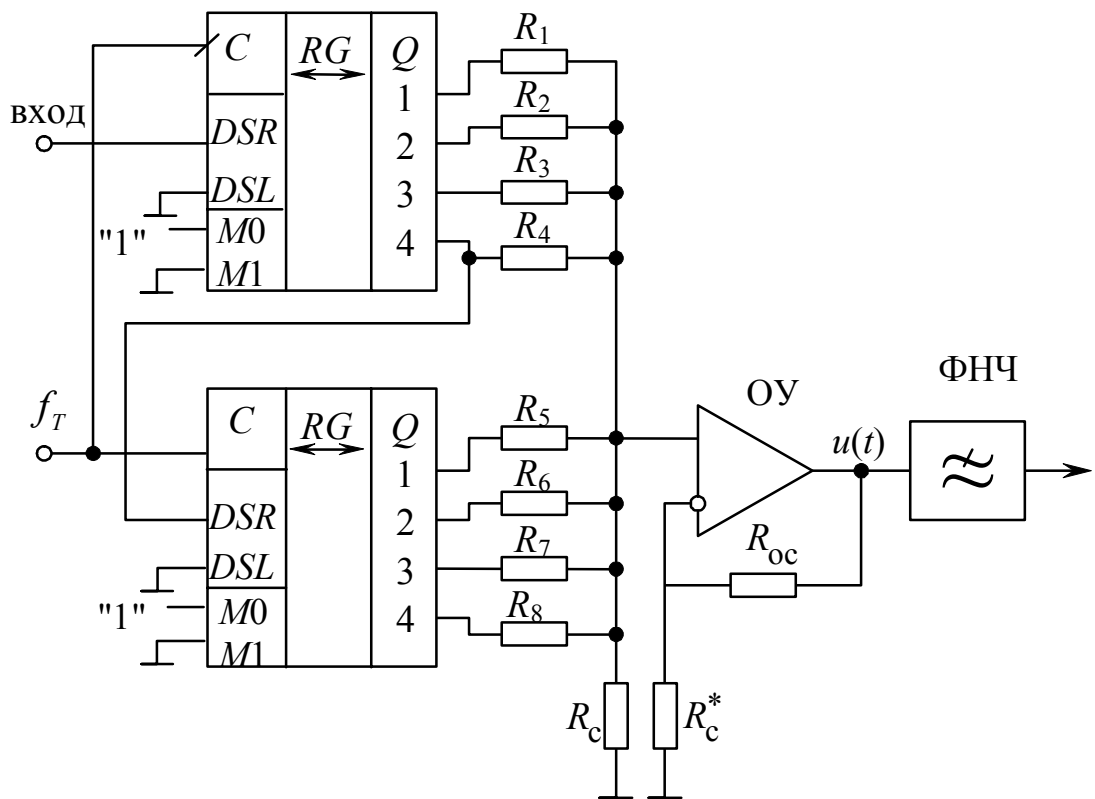


Рис. 6.6. Генератор сигналов на регистрах сдвига

Переход от ступенчато изменяющейся функции $u(t)$ к непрерывной функции осуществляется фильтром низкой частоты, полоса прозрачности $[0; \Delta F_{\text{ФНЧ}}]$ которого и форма амплитудно-частотной характеристики определяются требуемой точностью представления выходного аналогового напря-

жения. На рис.6.6 для $p = 8$ приведена схема генератора сигналов, выполненного на 4-разрядных регистрах сдвига. На тактовый вход (С) поступает тактовая последовательность с частотой следования импульсов $f_T = 1/\Delta t = 8/T_\phi$. Эта последовательность производит сдвиг логической единицы, поступающей на входы DSR регистров. В этом случае на выходах Q последовательно во времени со сдвигом на величину Δt будут формироваться уровни логической единицы. К выходам подключен комплект резисторов $R_1 \dots R_8$. Напряжение u_k на суммирующем резисторе R_c для k -го интервала дискретизации ($k = 0, \dots, 7$) определяется выражением:

$$u_k = \frac{u^{(1)}}{R_k/R_c + 1} \cong u^{(1)} \frac{R_c}{R_k}, \quad (6.18)$$

где $u^{(1)}$ – напряжение логической единицы на выходах Q регистров; $R_c \leq 0,1R_{k \min}$; $R_{k \min}$ – минимальное значение сопротивления резистора R_k . Суммирующий резистор R_c подключен к неинвертирующему входу операционного усилителя ОУ. Напряжение на выходе операционного усилителя определяется отношением сопротивлений резисторов R_{oc}/R_c , и выравнивание уровня напряжения осуществляется подстройкой резистора R_c^* . При таком соединении R_k и R_c с инвертирующим входом усилителя на его выходе будет формироваться однополярное напряжение. В случае, когда функция $u(t)$ изменяет свой знак на интервале формирования, необходимо подключить группы резисторов, участвующих в формировании напряжения другого знака, к инвертирующему входу операционного усилителя (суммирующий резистор R_c). Быстродействие всего устройства, изображенного на рис.6.6, ограничивается частотными свойствами операционного усилителя.

В ряде случаев огибающая сигнала удовлетворяет условию симметрии на интервале формирования, например, огибающая вида «приподнятого косинуса» или трапецеидального вида, поэтому можно уменьшить число взвешивающих элементов (резисторов) и несколько снизить погрешности, связанные с разбросом их сопротивлений. На рис. 6.7,а приведена схема генератора сигналов на основе цифрового реверсивного регистра сдвига. Символы канального алфавита в виде последовательности u_d логических 0 и 1 (рис.6.7,б) поступают на входы логических элементов, формирующих последовательности $f_{и}^{(I)}(t)$ и $f_{и}^{(II)}(t)$ импульсов запуска регистра по входам D_R и D_L , так и на входы логических элементов, формирующих импульсные последо-

вательности $f_{S_0}(t)$ и $f_{S_1}(t)$ управления работой регистра по входам S_0 и S_1 . На тактовый вход с регистра поступает импульсная последовательность с частотой следования импульсов $f_T=1/\Delta t=8/T$. К выходам Q_0, \dots, Q_3 регистра подключены резисторы, участвующие в формировании сигнала положительной полярности, а к выходам Q_4, \dots, Q_7 – резисторы, формирующие сигнал отрицательной полярности.

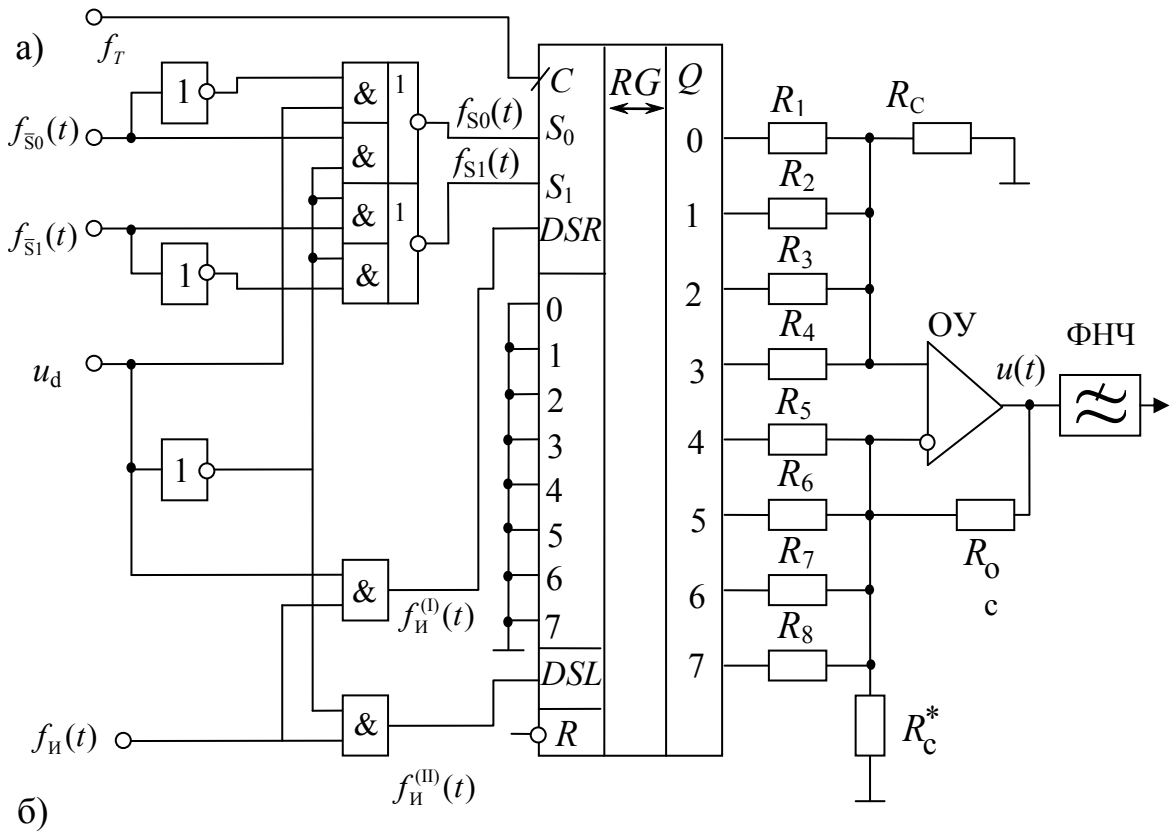


Рис. 6.7. Генератор сигналов на основе рекурсивного регистра сдвига

Рассмотрим более подробно работу реверсивного регистра сдвига. Он имеет четыре режима работы, управление которыми осуществляется по входам S_0 и S_1 в соответствии с табл. 6.3.

Таблица 6.3.

S_0	S_1	Режим работы
1	1	Параллельная запись 8-разрядного числа
1	0	Сдвиг вправо (от выхода Q_0 к Q_7)
0	1	Сдвиг влево (от выхода Q_7 к Q_0)
0	0	Хранение информации

На интервале времени $[0, T]$ при появлении на входе импульса запуска последовательности $f_{in}^{(I)}(t)$ (рис. 6.7,б) осуществляется последовательный сдвиг логической 1 вправо на четыре тактовых интервала Δt и затем влево также на четыре тактовых интервала, что приводит к появлению на выходах Q_0, \dots, Q_3 последовательностей $f_{Q_0}(t), \dots, f_{Q_3}(t)$. На интервале времени $[T, 2T]$ при появлении на входе D_L импульса запуска последовательностей $f_{in}^{(II)}(t)$ (рис. 6.7,б) происходит аналогичное формирование последовательностей $f_{Q_4}(t), \dots, f_{Q_7}(t)$. Суммирующие резисторы R_c и R_c^* подключены к входам ОУ, регулировка уровня напряжения на выходе ОУ осуществляется резистором R_c^* .

Функции цифрового регистра сдвига может выполнять совокупность счетчика и дешифратора на основе которых также можно реализовать генератор сигналов. На рис. 6.8 приведена схема такого генератора. Последовательность u_d (рис. 6.7,б) поступает на вход 1 старшего разряда дешифратора-демультиплектора. Напряжение уровня логического 0 будет формироваться либо на выходах 00...07 (на входе 8 присутствует логический 0), либо на выходах 08...15 (на входе 8 присутствует логическая 1). Уровень логического 0 с помощью логических инверторов преобразуется в уровень логической 1. Таким образом, на выходах логических инверторов будут появляться последовательно во времени со сдвигом на Δt уровни логической единицы. Как и в схеме на рис. 6.7,а, взвешивающие резисторы подключены к двум входам ОУ, на выходе которого и формируется напряжение $u(t)$.

Преимуществом этого устройства по сравнению с приведенным на рис. 6.7,а является более простая аппаратурная реализация, поскольку не требуется логических преобразований управляющих последовательностей, необходимых для обеспечения работы реверсивного регистра сдвига.

Формирование сигналов на несущей частоте ω_0 возможно путем непосредственного формирования отрезков колебаний частоты ω_0 , сдвинутых

относительно друг друга на интервалы, равные Δt . При этом амплитуды и фазы таких отрезков должны соответствовать значениям огибающей $A(t)$ и фазы $\varphi(t)$ сигнала $s(t)$ в моменты времени, совпадающие с началом отрезков (моментами дискретизации).

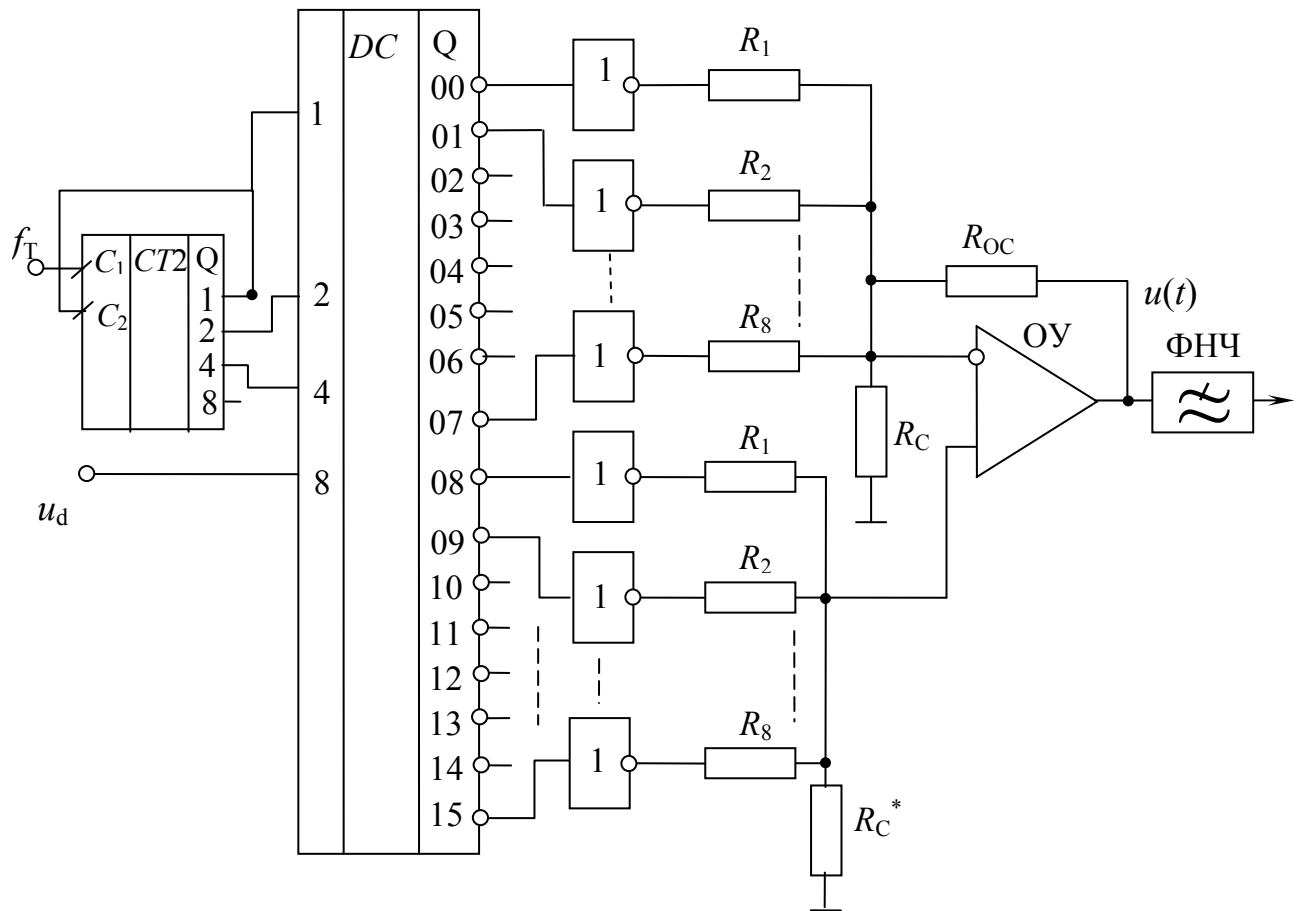


Рис. 6.8. Генератор сигналов на основе дешифратора – демультиплектора

В общем случае вид сигнала на интервале формирования T_Φ будет определяться выражением:

$$y(t) = \sum_{k=0}^{p-1} A(k\Delta t) \cdot \psi(t - k\Delta t) \cos[\omega_0 t - \varphi(k\Delta t)]. \quad (6.18)$$

Рассмотрим практическую реализацию такого генератора сигналов с амплитудной манипуляций. Вид сигнала $y(t)$ будет представлять собой последовательность модулированных по амплитуде отрезков колебаний с несущей частотой ω_0 . Достоинством такого метода формирования является отсутствие дополнительного преобразования несущей частоты, как это имело место в генераторах сигналов, схемы которых приведены на рис.6.6 – 6.8. На

рис. 6.9 приведена схема, где функции ключевых элементов выполняют аналоговые четырехканальные мультиплексоры, обладающие достаточно высоким быстродействием (время включения $\tau_{\text{вкл}} < 150 \text{ нс}$) и небольшим сопротивлением открытого канала ($R_{\text{откр}} < 70 \text{ Ом}$).

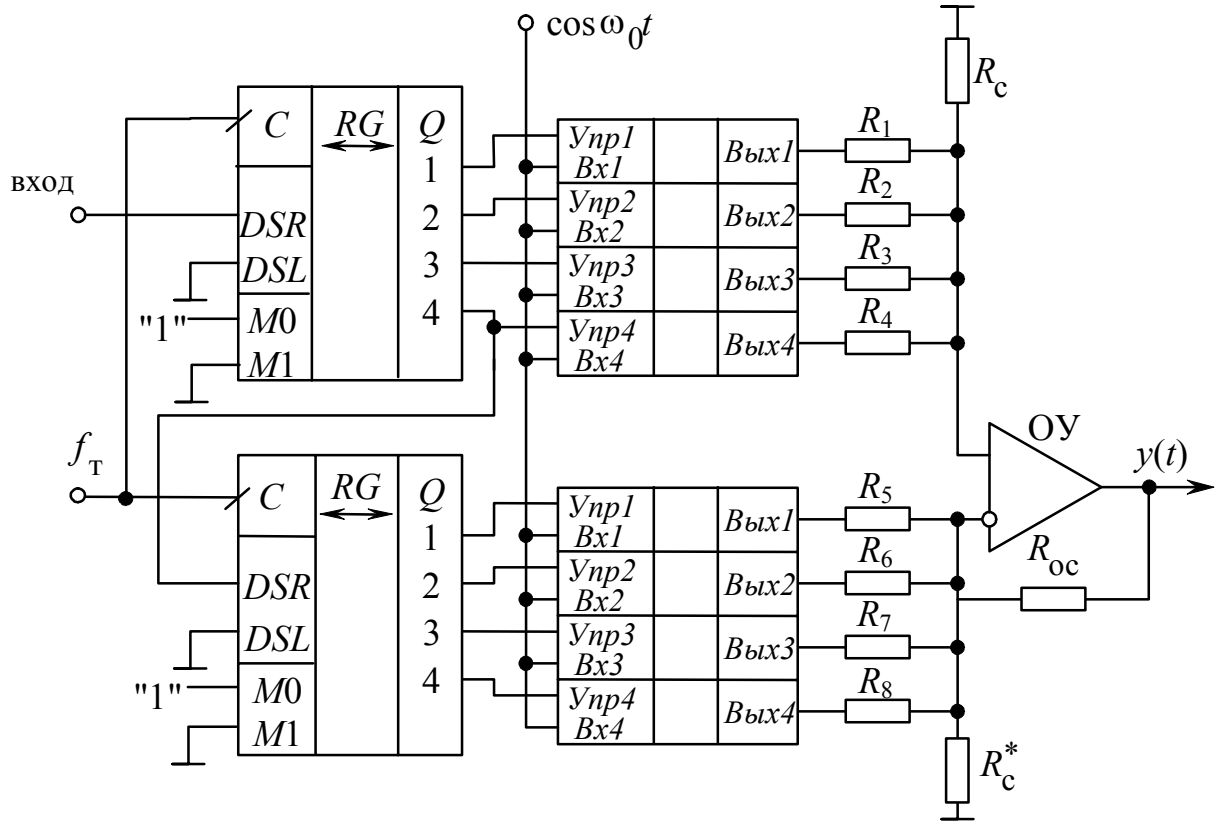


Рис. 6.9. Генератор сигналов на основе аналоговых 4-канальных мультиплексоров

Управляющие сигналы этих мультиплексоров имеют уровни ТТЛ, поэтому входы управления Упр1...Упр2 могут подключаться непосредственно к выходам Q регистров сдвига. При появлении уровня логической 1 на управляющих входах мультиплексора на его выходах Вых1...Вых4 будут формироваться отрезки гармонических колебаний длительностью Δt . На выходе операционного усилителя генерируется последовательность отрезков колебаний вида:

$$y(t) = \sum_{k=0}^{p-1} A(k\Delta t)\psi_0(t - k\Delta t)\cos(\omega_0 t), \quad (6.19)$$

причем подключение групп резисторов ($R_1...R_4$) и ($R_5...R_8$) к входам усилителя соответствует случаю, когда функция $A(t)$ изменяет свой знак на интервале формирования.

На рис. 6.10 приведена схема генератора сигналов, где функции ключе-

вых элементов выполняет аналоговый 8-канальный демультиплексор. Управление его работой осуществляется с помощью двоичного счетчика. В зависимости от вида управляющих последовательностей на входах 1...3 демультиплексора происходит подключение аналогового входа U к выходам демультиплексора. На этих выходах последовательно во времени с интервалом Δt будут появляться отрезки гармонического колебания, которые с помощью резисторов $R_1...R_8$ преобразуются в рассмотренную выше последовательность колебаний. На рис.6.10 подключение резисторов соответствует случаю, когда функция $A(t)$ не изменяет свой знак на всем интервале формирования.

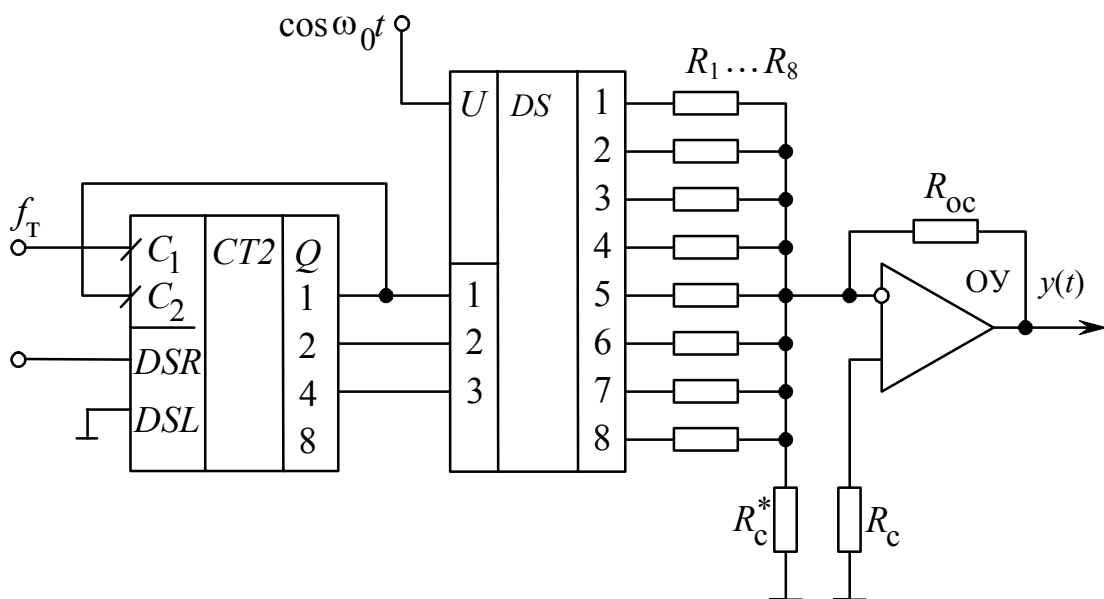


Рис. 6.10. Генератор сигналов на основе аналогового 8-ми канального демультиплексора

Аппаратурные погрешности этих генераторов сигналов связаны в основном с разбросом значений сопротивлений открытых каналов аналоговых мультиплексоров, что приводит к появлению паразитной амплитудной модуляции последовательности отрезков колебаний с частотой ω_0 .

Генераторы сигналов могут быть выполнены на основе запоминающих устройств и цифроаналоговых преобразователей. Действительно, поскольку работа запоминающего устройства описывается M -разрядной (M – число входов запоминающего устройства) логической функцией от N входных переменных, то с его помощью можно производить таблично заданные функциональные преобразования, в частности формирование сложных функций. Воспроизведение требуемой функции, выборочные значения которой записаны в ячейках ПЗУ, производится путём формирования номеров адресов ячеек ПЗУ с помощью, например, двоичных счётчиков. Включение в этих

условиях на выходе ПЗУ цифроаналогового преобразователя и фильтра нижних частот позволяет построить на основе ПЗУ функциональный генератор (формирователь аналоговых сигналов) или синтезатор сигналов.

Рассмотрим пример реализации цифрового генератора сигналов на основе ПЗУ. На рис. 6.11 приведена схема генератора сигналов при использовании ПЗУ с объёмом памяти 256×4 бит. В ПЗУ записываются 12-разрядные кодовые слова, соответствующие выборочным значениям функции, причём первые четыре разряда записываются в первую микросхему, вторые четыре разряда – во вторую и последние четыре – в третью микросхему ПЗУ. С выходов ПЗУ 12-разрядные кодовые слова, представленные в дополнительном коде, поступают на входы ЦАП. Считыванием информации из ПЗУ управляет двоичный счётчик и входная последовательность символов u_d , поступающая непосредственно на вход управления CS ПЗУ. К выходам ПЗУ подключены нагрузочные резисторы R_n , соединенные с источником питания $u_{\text{ин}}$. Требуемый объём памяти ПЗУ, необходимый для формирования числа выборочных значений сигнала $s(t)$ при $N = 16$ составляет 16×12 бит. При использовании ПЗУ с общим объёмом памяти $256 \times 4 \times 3$ бит появляется возможность записи кодовых слов, соответствующих различным сигналам, что существенно увеличивает функциональные возможности устройства. Управление сменой форм генерируемых сигналов будет осуществляться путём изменения кода адреса $A_1A_2A_3A_4$, поступающего на старшие разряды адресных входов управления ПЗУ.

На вход опорного напряжения ЦАП поступает гармоничное колебание. Показанное на рис. 6.11 включение ЦАП соответствует 4-квadrантному перемножению опорного колебания и 12-разрядных кодовых слов, представленных в дополнительном коде.

На выходе ОУ формируется гармоническое колебание, модулированное ступенчато-изменяющимся напряжением. Амплитуда выходного напряжения определяется отношением сопротивлений резисторов R_{oc}/R_1 , а регулировка уровня нуля на выходе ОУ осуществляется резистором R_2^* .

Быстродействие рассмотренного генератора сигналов ограничивается цифро-аналоговым преобразователем и операционным усилителем. Бóльшим быстродействием обладают ЦАП на основе биполярных структур, однако включение таких преобразователей в умножающем режиме требует значительно более сложных схемных решений.

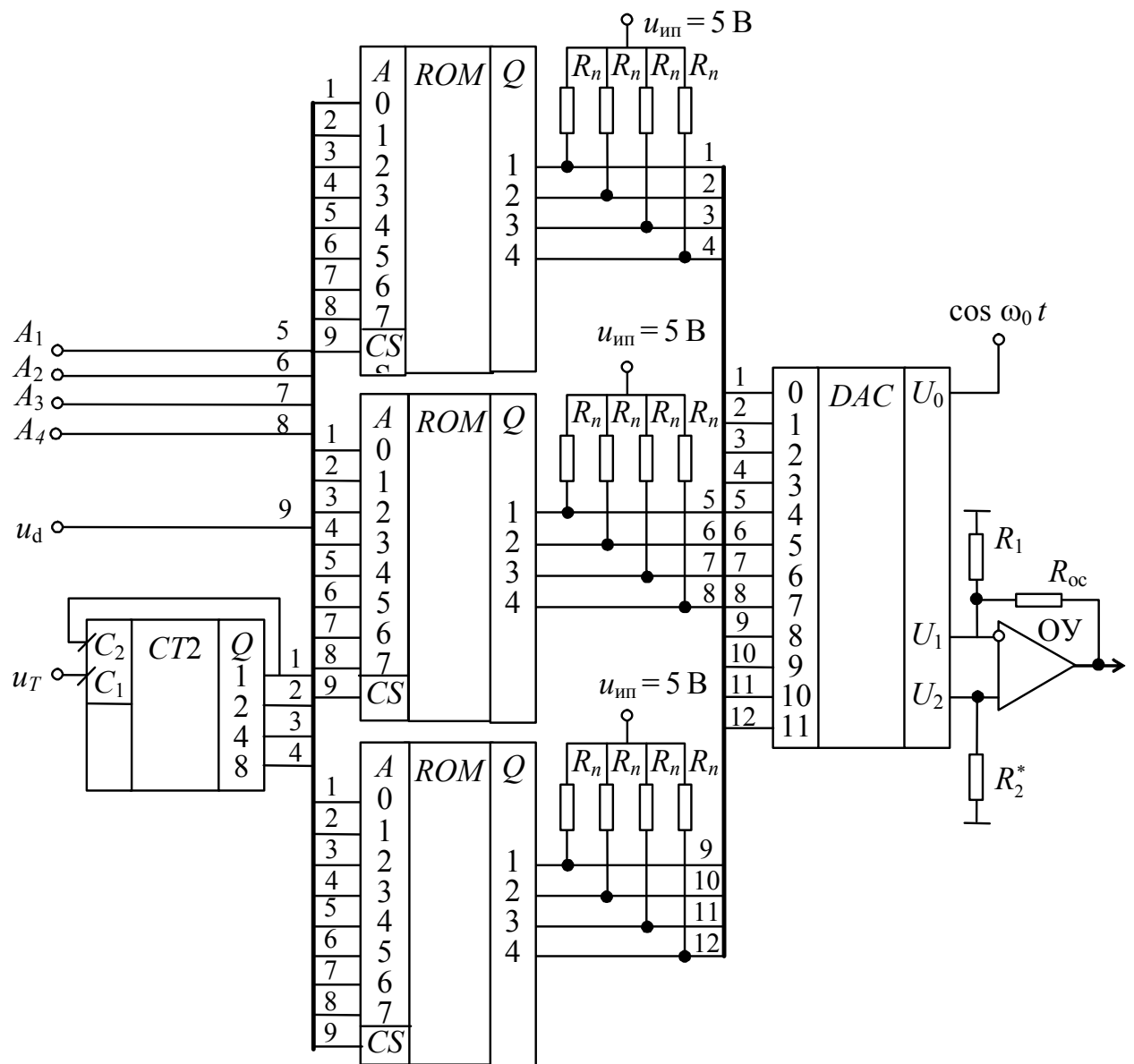


Рис. 6.11. Генератор сигналов на основе ПЗУ

Генераторы сигналов могут быть построены и на основе логических элементов. На рис.6.12,*а* приведена схема устройства, а на рис.6.12,*б* – временные диаграммы его работы, где $u_1 \dots u_4$ – импульсные последовательности на выходах блока формирования импульсных последовательностей БИП; u_5 – форма напряжения на суммирующем резисторе R_c .

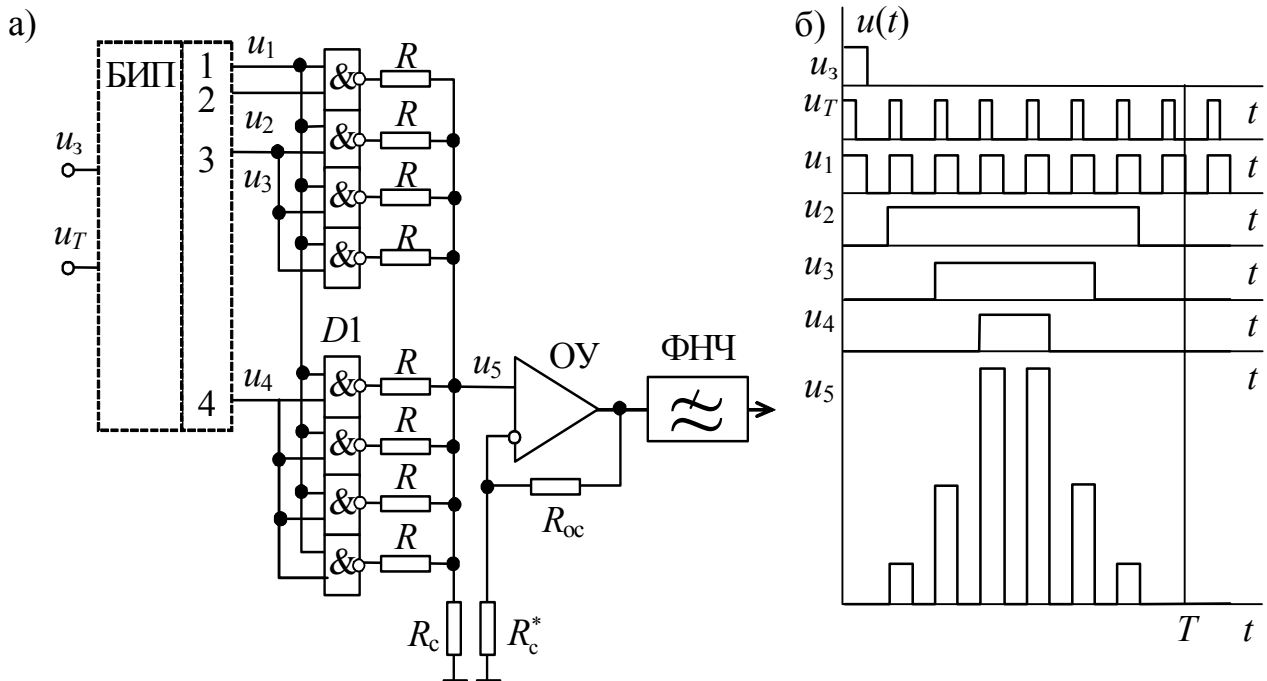


Рис. 6.12. Генератор сигналов на основе логических элементов

Подключение входов логических элементов И-НЕ к выходам БИП определяется формой напряжения, которое должно быть сформировано на выходе генератора (напряжение u_5 на рис.6.12,б). К выходам логических элементов подключён комплект резисторов. Сложение токов с каждой ветви осуществляется на суммирующем резисторе R_c , величина которого выбирается исходя из условия: $R_c \leq 0,1R$. Суммирующий резистор подключён к инвертирующему входу операционного усилителя и величина напряжения на его выходе будет определяться отношением сопротивлений R_{oc}/R_c .

Приведённая на рис.6.12,б форма выходного напряжения имеет вид АИМ последовательности, в которой длительность импульсов равна $\Delta t/2$, а амплитудные значения соответствуют значению сигнала в моменты времени $t = k \cdot \Delta t$ ($k = 0, 1, \dots, N-1$).

Аппаратурные погрешности данного устройства связаны в основном с разбросом уровней напряжения логической 1 на выходах логических элементов. Это вызвано тем, что при сложении уровней напряжений с различных выходов логических элементов погрешности, связанные с изменением уровня логической 1 относительно некоторого среднего значения, могут быть однополярными, что приведёт к их зависимости от количества одновременно складываемых импульсов и, в конечном итоге, к возрастанию паразитной амплитудной и фазовой модуляции. Быстродействие генератора сигналов ограничивается частотными свойствами операционного усилителя.

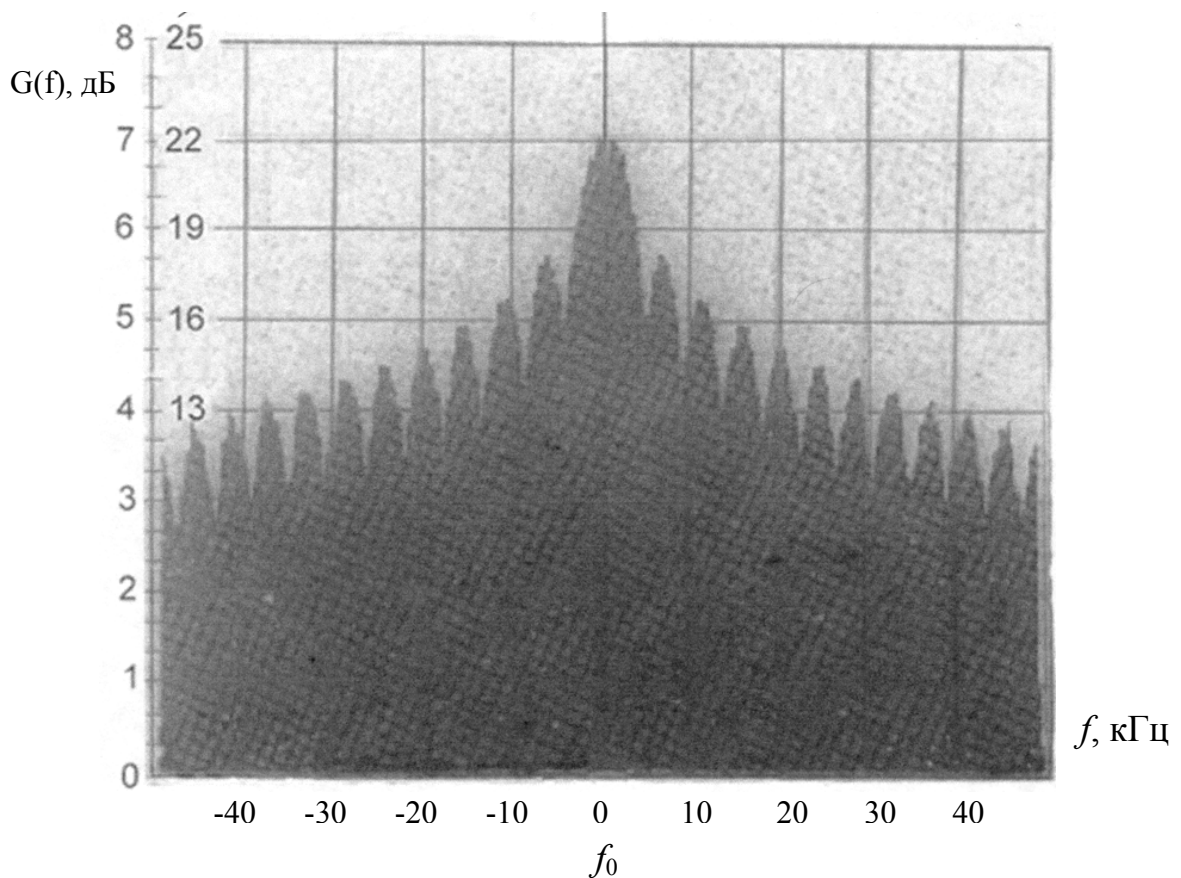


Рис. 6.13. Энергетический спектр случайной последовательности классических двоичных сигналов с фазовой манипуляцией: масштаб по оси ординат 10 дБ/дел

Возможность применения рассмотренных генераторов сигналов иллюстрируется спектрограммами последовательностей сигналов, полученных с помощью генератора, схема которого приведена на рис. 6.11. Формирование сигналов производилось на частоте $f_0 = 500$ кГц. Число выборочных значений на интервале формирования было выбрано 16 при числе уровня квантования $L = 511$. В качестве восстанавливающего использовался ФНЧ второго порядка с значением $\Delta F_{\text{ФНЧ}}/2 = 20$ кГц. Последовательностью символов канального алфавита являлась псевдослучайная последовательность двоичных символов с частотой следования $1/T = 3,9$ кБод. Для оценки влияния систематических и аппаратных погрешностей рассматриваемого генератора сигналов на рис. 6.13 приведена спектрограмма энергетического спектра $G(f)$ случайной последовательности классических ФМ сигналов со скачками фазы колебания на 180° . Из сравнения полученного энергетического спектра с теоретическим [20] видно, что их отличие, например в области 4-го побочного максимума ($f_0 + 4/T \leq f \leq f_0 + 5/T$) составляет не более 2 дБ, что позволяет считать аппаратные погрешности формирования незначительными.

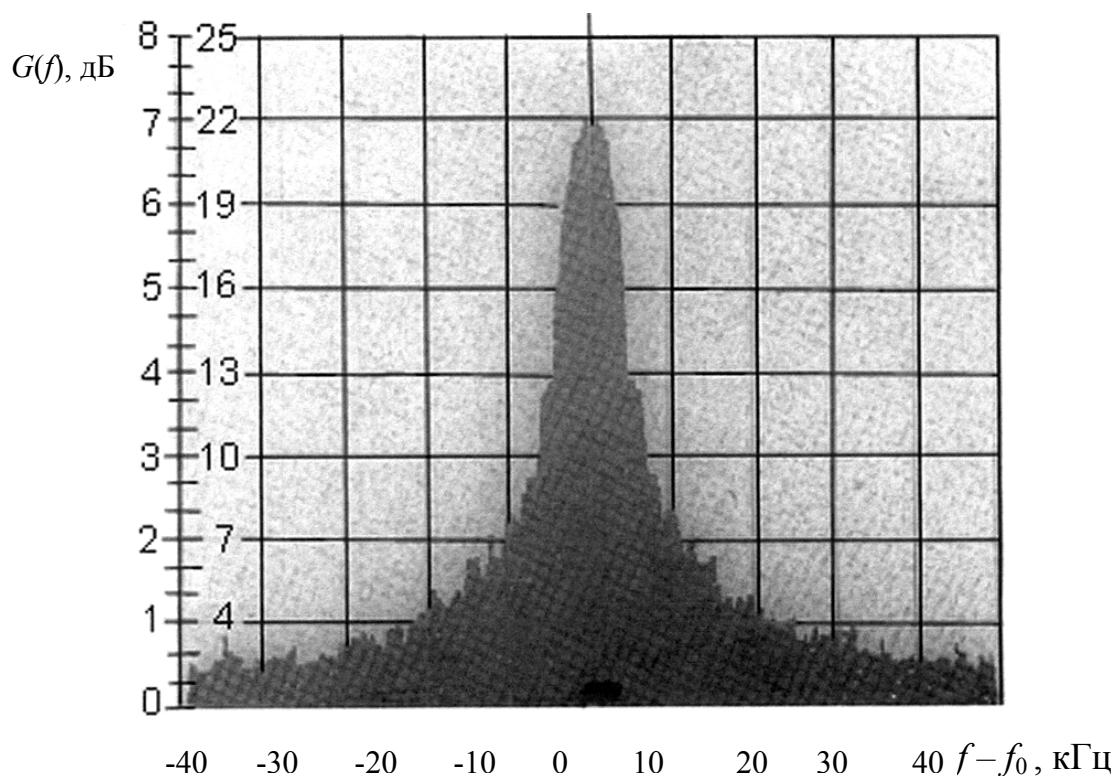


Рис. 6.14 Энергетический спектр случайной последовательности спектрально-эффективных двоичных сигналов с амплитудно-фазовой манипуляцией и огибающей вида «приподнятого косинуса» удвоенной длительности: масштаб по оси ординат 10дБ/дел

Рассмотрим на примере формирования спектрально-эффективных двоичных сигналов с амплитудно-фазовой манипуляцией и огибающей вида $\sin^2 x$ удвоенной длительности влияния аппаратных погрешностей на искажения энергетического спектра последовательности в занимаемой полосе частот ΔF . Заметим, что в данном случае в последовательности сигналов присутствует управляемая межсимвольная интерференция. Как видно из спектрограммы рис. 6.14, уровень внеполосных излучений, например, при расстройке от средней частоты f_0 на величину $2/T$ уменьшается от -26 дБ до -44 дБ (рис. 6.13 и 6.14) по отношению к классическим ФМ сигналам. Аппаратурные погрешности при формировании таких сигналов также не превосходит $1-2$ дБ для уровней энергетического спектра $-50 \dots -60$ дБ.

6.4 Цифровой коррелятор

В разнообразных радиотехнических устройствах используются оптимальные алгоритмы когерентного и некогерентного приема дискретных (цифровых) сообщений [20]. Практическая реализация таких алгоритмов предполагает использование либо корреляционных устройств, либо согласо-

ванных фильтров. Рассмотрим прежде всего реализацию корреляционных устройств. Такие корреляторы реализуют алгоритм вычисления выражения:

$$U_0 = \int_0^{T_a} x(t)s(t)dt, \quad (6.20)$$

где $x(t)$ – входное принимаемое колебание, представляющее собой сумму полезного сигнала и помех; $s(t)$ – ожидаемый сигнал; T_a – интервал анализа.

Реализация алгоритма когерентной обработки сигналов, может производиться с помощью цифровых корреляторов, в соответствии с процедурой вычисления U_0 в форме:

$$U_0 \cong U_0^{(N)} = \frac{\Delta t}{2} \cdot \sum_{k=0}^{N-1} A_x^{(n)}[k]A^{(p)}[k], \quad (6.21)$$

где $A_x^{(n)}[k]$ и $A^{(p)}[k]$ – последовательности чисел, являющиеся цифровыми эквивалентами принимаемого колебания $x(t)$ и ожидаемого сигнала $s(t)$; n, p – разрядность рассматриваемых чисел; q – количество разрядов числа на выходе цифрового коррелятора. Структурная схема цифрового коррелятора приведена на рис.6.15, где ЦП – цифровой перемножитель n -разрядных и p -разрядных чисел в некоторые фиксированные моменты времени $k \cdot \Delta t$; ЦУ – цифровое устройство формирования последовательности чисел $A^{(p)}[k]$; СН – цифровой сумматор-накопитель $(n+p)$ -разрядных чисел, количество которых равно N .

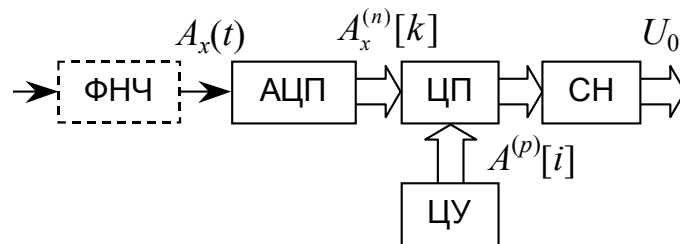


Рис. 6.15. Структурная схема цифрового коррелятора

Остановимся подробнее на принципах построения цифрового перемножителя. Наиболее быстродействующими цифровыми перемножителями, которые применяются в радиотехнических устройствах, являются ЦП параллельного типа. Такие цифровые перемножители разделяются на матричные перемножители и перемножители на основе ПЗУ.

Матричные перемножители (рис. 6.16,а) осуществляют операцию перемножения n -разрядных и p -разрядных чисел $X^{(n)}$ и $Y^{(p)}$ в соответствии со следующим алгоритмом работы:

$$Z^{(n+p)} = X^{(n)}Y^{(p)} = \sum_{i=0}^{n-1} x_i 2^i \sum_{j=0}^{p-1} y_j 2^j, \quad (6.22)$$

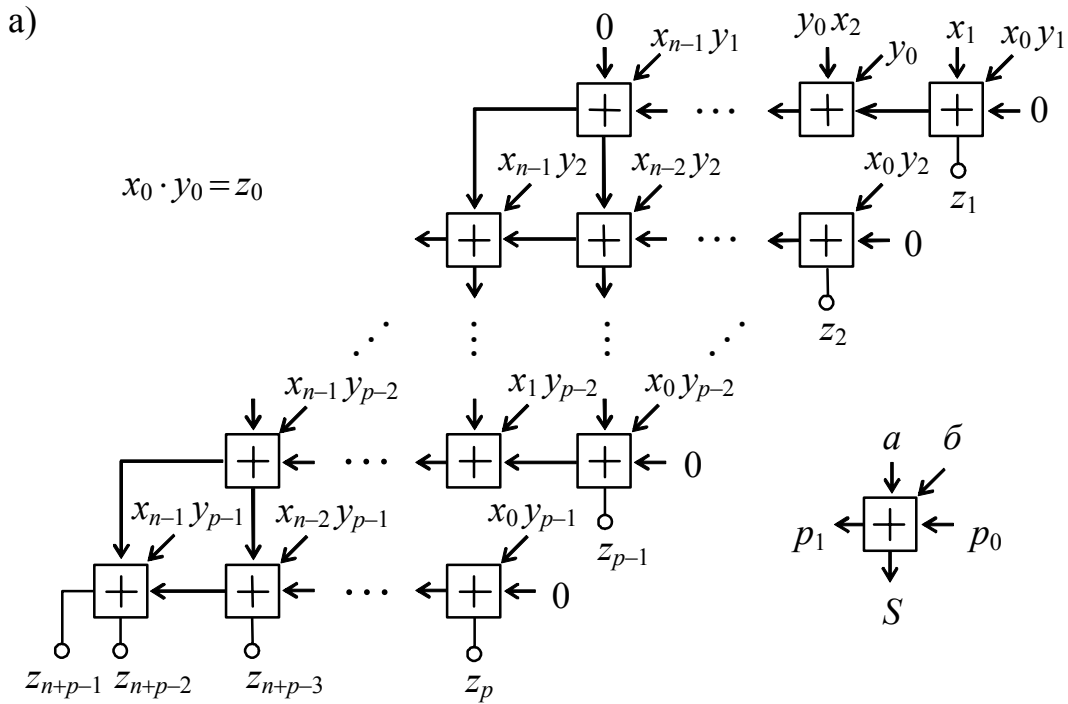
где x_i , y_j и z_k – значения i -го, j -го и k -го разрядов чисел $X^{(n)}$, $Y^{(p)}$ и $Z^{(n+p)}$, записанных в двоичном коде. При этом каждое число $X^{(n)}$ представляется в виде n -разрядного числа $(x_{n-1}, x_{n-2}, \dots, x_0)$; число $Y^{(p)}$ – в виде p -разрядного числа $(y_{p-1}, y_{p-2}, \dots, y_0)$; и число $Z^{(n+p)}$ – в виде $(n+p)$ -разрядного числа $(z_{n+p-1}, z_{n+p-2}, \dots, z_0)$, причём старшими являются разряды x_{n-1} , y_{p-1} и z_{n+p-1} , а младшими – x_0 , y_0 и z_0 .

На входы двоичных сумматоров поступают результаты логического перемножения разрядов положительных чисел $X^{(n)}$ и $Y^{(p)}$. Двоичный сумматор имеет входы $A_1 \dots A_4$ и $B_1 \dots B_4$, вход переноса числа $P0$, выход суммы чисел S и выход переноса числа p . Каждая строка сумматоров (рис.6.16,а) формирует частичную сумму и передаёт её следующей строке. Младший разряд числа $Z^{(n+p)}$ образуется отдельно как результат логического перемножения разрядов x_0 и y_0 .

Принципиальная схема матричного перемножителя, предназначенная для перемножения 4-разрядных и 3-разрядных чисел $X^{(4)}$ и $Y^{(3)}$, в которой для логического перемножения разрядов используются логические элементы И–НЕ, а в качестве сумматоров применяются 4-разрядные полные сумматоры, приведена на рис.6.16,б. Разряды результирующего числа $Z^{(7)}$ формируются в инвертированном виде. На входы сумматоров, обозначенные буквой a , подаётся напряжение логической единицы.

Быстродействие такого матричного перемножителя определяется быстродействием элементов, входящих в устройство, и для его оценки необходимо знать путь прохождения сигнала по перемножителю.

Перемножители на основе постоянных запоминающих устройств содержат ПЗУ, в которых хранятся таблицы произведений различных вариантов сомножителей, причём общее число произведений равно 2^{n+p} $(n+p)$ -разрядных чисел. Основу ПЗУ составляют матрицы ячеек памяти с записанными в них $(n+p)$ -разрядными числами, например, путём прожигания соответствующих перемычек в матрице. Обращение к хранимой информации производится через адресные входы путём подачи на них соответствующего кода адреса, а съём $(n+p)$ -разрядных чисел происходит, как правило, в параллельном коде с $(n+p)$ выходов ПЗУ.



б)

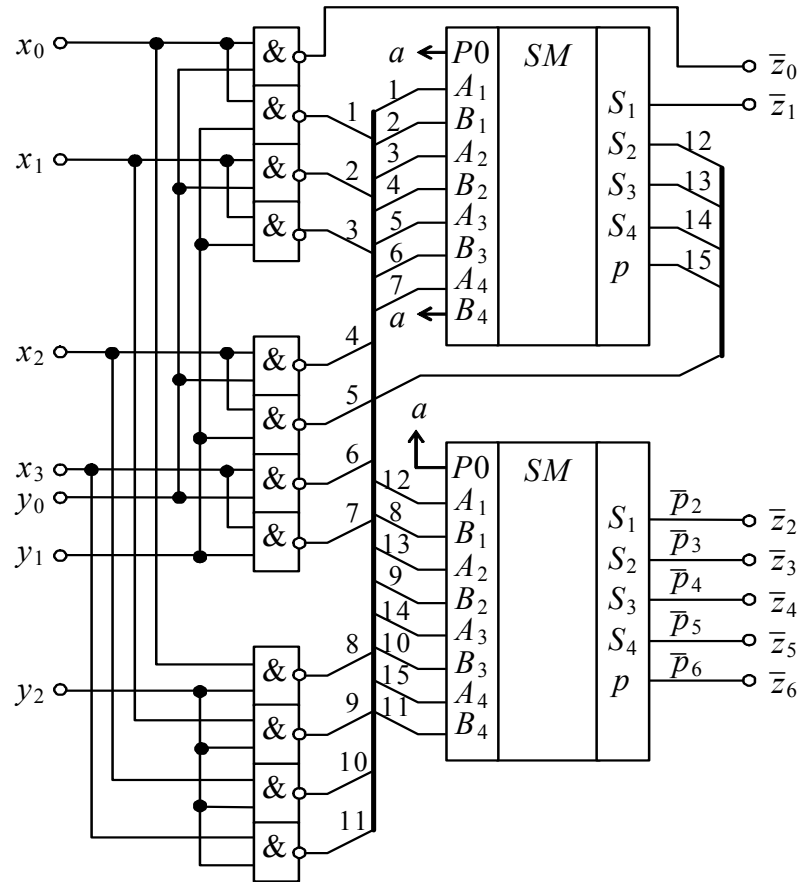


Рис. 6.16. Схема матричного перемножителя

Быстродействие таких перемножителей определяется минимальным временем считывания информации из ПЗУ и, как правило, оно меньше, чем у матричных перемножителей.

Одним из основных узлов структурной схемы на рис. 6.15 является цифровой сумматор-накопитель (СН) N полученных $(n+p)$ -разрядных чисел, осуществляющий операцию суммирования в соответствии с выражением (6.12). Такой сумматор-накопитель (рис. 6.17) состоит из 4-разрядного полного сумматора и регистра-накопителя. В сумматоре SM происходит сложение чисел, поступающих на входы $A_1...A_4$ с выходов ЦП и чисел, которые формируются на выходах регистра-накопителя и поступают на вторые входы $B_1...B_4$. Результат суммирования записывается в регистр-накопитель.

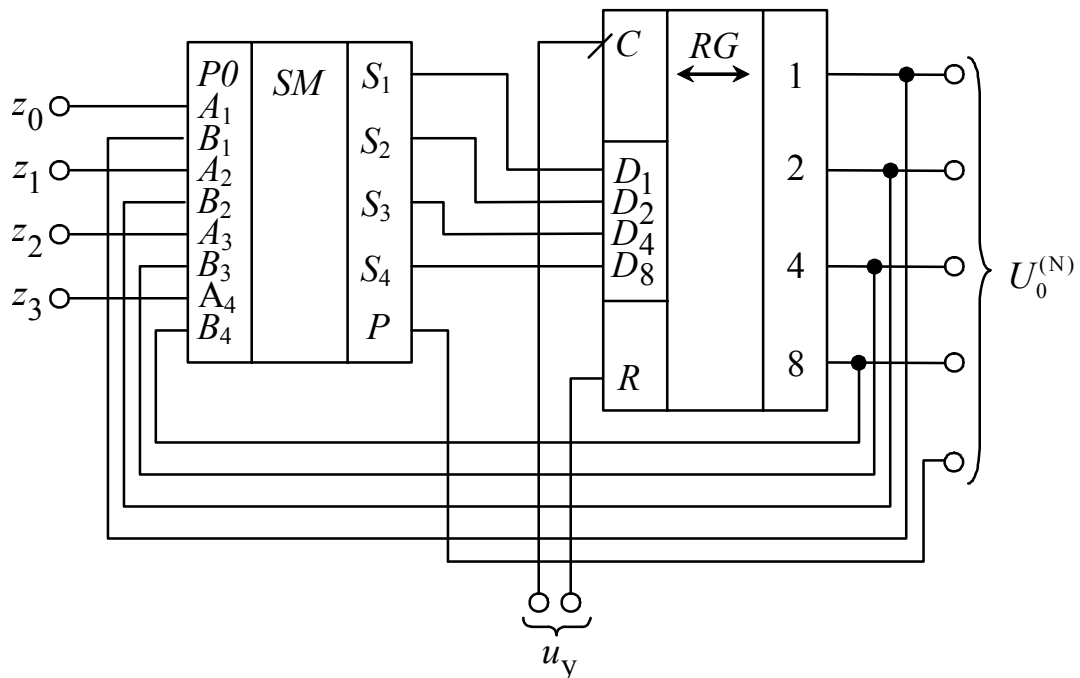


Рис. 6.17 Схема цифрового сумматора-накопителя

Организация процесса суммирования чисел с записью результата в регистр-накопитель осуществляется при помощи управляющих последовательностей u_y . На первом шаге суммирования вычисляется сумма первого числа, считываемого с выходов ЦП, с двоичным числом, записанным в регистре-накопителе, все разряды которого равны нулю. Эта первая сумма записывается в регистр-накопитель. На втором шаге вычисляется сумма второго числа, считываемого с выходов ЦП, с первой суммой, хранящейся в регистре-накопителе, и результат сложения записывается в этот же регистр-накопитель. Процесс суммирования и накопления чисел продолжается N шагов и, таким образом, в конце цикла суммирования в регистре-накопителе

будет записано число, соответствующее значению:

$$\sum_{k=0}^{N-1} A_x^{(n)}[k] \cdot A^{(p)}[k] \quad (6.23)$$

6.5 Цифровые согласованные фильтры

С помощью согласованных фильтров происходит вычисление выражения вида:

$$s_{\text{ВЫХ}}(t) = \int_0^t x(t-\tau)K(\tau)d\tau,$$

где $x(t)$ и $K(t)$ – принимаемое колебание и импульсный отклик фильтра соответственно.

Реализация алгоритма согласованной фильтрации может быть выполнена путём непосредственного вычисления выражения дискретной свёртки. Рассмотрим один из возможных вариантов построения цифрового согласованного фильтра. Алгоритм дискретной свёртки в этом случае может быть записан в виде [20]:

$$s_{\text{ВЫХ}}^{(q)}[k] = \frac{\Delta t}{2} \cdot \sum_{i=0}^{N-1} A_x^{(n)}[k-i]K^{(p)}[i], \quad (6.24)$$

где $s_{\text{ВЫХ}}^{(q)}[k]$ - последовательности q -разрядных чисел, соответствующих выборочным значениям из выходного сигнала $s_{\text{ВЫХ}}(t)$; $K^{(p)}[i]$ – последовательности p -разрядных чисел, являющихся цифровыми эквивалентами выборочных значений из импульсного отклика $K(t)$ согласованного фильтра.

Структурная схема цифрового согласованного фильтра приведена на рис. 6.18, где ФО – устройство формирования импульсного отклика (последовательности p -разрядных чисел $K^{(p)}[i]$). Дискретная свёртка осуществляется в два этапа. Первый этап заключается в организации перемножения каждого n -разрядного числа $A_x^{(n)}[k-i]$ на все N значений p -разрядных чисел $K^{(p)}[i]$ и записи полученных N произведений p -разрядных чисел в ОЗУ, представляющее собой куб памяти размерами $N \times N \times (n+p)$, в который построчно (по N чисел) записываются произведения $A_x^{(n)}[k-1]K^{(p)}[i]$ (рис.6.19). Второй этап состоит в считывании информации из ОЗУ и суммировании с помощью цифрового сумматора-накопителя (СН) N $(n+p)$ -разрядных чисел для каждого значения k функции $s_{\text{ВЫХ}}^{(q)}[k]$.

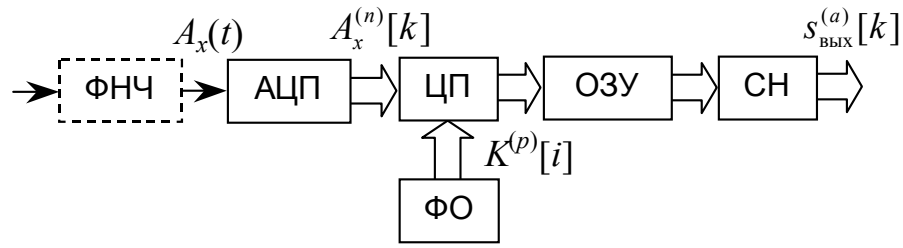


Рис. 6.18 Структурная схема цифрового согласованного фильтра

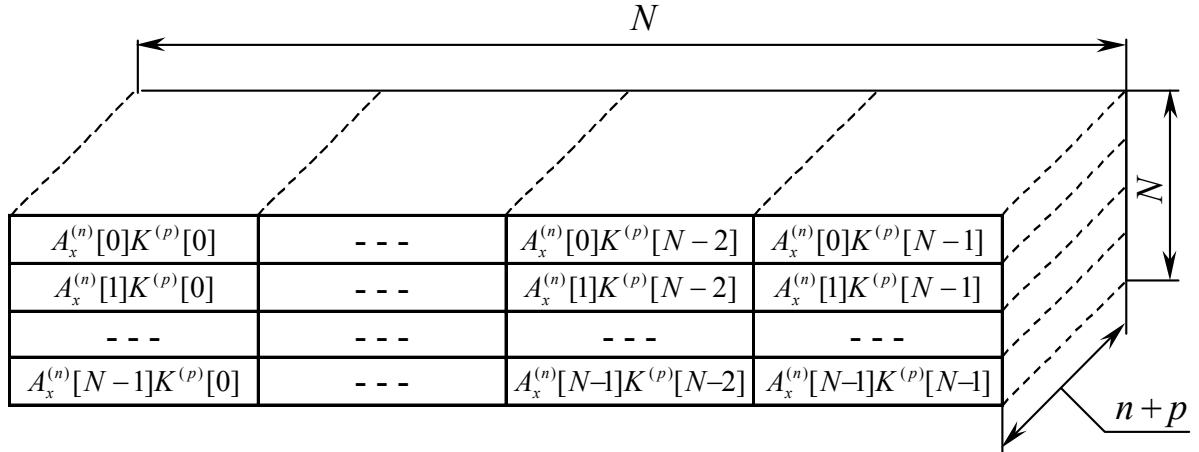


Рис. 6.19 Структура ОЗУ цифрового согласованного фильтра.

Все элементы куба представляют собой слагаемые, входящие в рассмотренное выше выражение свёртки, и для получения $s_{\text{ВЫХ}}^{(q)}[k]$ необходимо организовать диагональный режим считывания чисел из ОЗУ. Действительно, при $k = 0$

$$s_{\% \text{о}}^{(q)}[0] = A_x^{(n)}[0]K^{(p)}[0] + A_x^{(n)}[-1]K^{(p)}[1] + \dots + A_x^{(n)}[-(N-1)]K^{(p)}[N-1];$$

при $k = 1$

$$s_{\% \text{о}}^{(q)}[1] = A_x^{(n)}[1]K^{(p)}[0] + A_x^{(n)}[0]K^{(p)}[1] + \dots + A_x^{(n)}[-(N-2)]K^{(p)}[N-1];$$

при $k = N - 2$

$$s_{\% \text{о}}^{(q)}[N-2] = A_x^{(n)}[N-2]K^{(p)}[0] + A_x^{(n)}[N-3]K^{(p)}[1] + \dots + A_x^{(n)}[-1]K^{(p)}[N-1];$$

при $k = N - 1$

$$s_{\% \text{о}}^{(q)}[N-1] = A_x^{(n)}[N-1]K^{(p)}[0] + A_x^{(n)}[N-2]K^{(p)}[1] + \dots + A_x^{(n)}[0]K^{(p)}[N-1];$$

Значения чисел $A_x^{(n)}[-1]$, $A_x^{(n)}[-2]$, ..., $A_x^{(n)}[-(N-1)]$ в случае существования принимаемого колебания $x(t)$ на интервале анализа T_a равны нулю. В то же время, если принимаемое колебание $x(t)$ представляет собой непрерывный процесс на интервале $-\infty < t < +\infty$, то значения чисел $A_x^{(n)}[-1]$, $A_x^{(n)}[-2]$, ..., $A_x^{(n)}[-(N-1)]$ соответствуют значениям функций $A_x[-1]$, $A_x[2]$, ..., $A_x[-(N-1)]$, выборочных значений $x(-\Delta t)$, $x(-2\Delta t)$, ..., $x(-(N-1)\Delta t)$, на интервале анализа, так что имеем:

$$A_x^{(n)}[k] = A_x^{(n)}[k + N], \quad k = 1, 2, 3, \dots, N - 1. \quad (6.25)$$

С учетом (6.25), получим диагональный закон считывания и суммирования чисел из оперативного запоминающего устройства, причём очевидно, что для организации обработки $x(t)$ на интервале $-\infty < t < +\infty$ необходимо чередовать этапы считывания и суммирования с этапами записи чисел. При этом, после k -го диагонального считывания и суммирования N чисел из оперативно запоминающего устройства необходимо произвести запись N чисел следующих произведений в k -ю строку куба памяти. Процесс повторяется циклически через N тактов (считывание, суммирование)/(запись). Время одного такта равно интервалу дискретизации Δt . Такой метод построения цифровых согласованных фильтров позволяет создавать "наращиваемую" структуру устройства.

Методы построения и практическая реализация устройств формирования импульсного отклика (последовательности p -разрядных чисел $K^{(p)}[i]$) не отличаются от применяемых в генераторах сигналов (см. разд. 6.3).

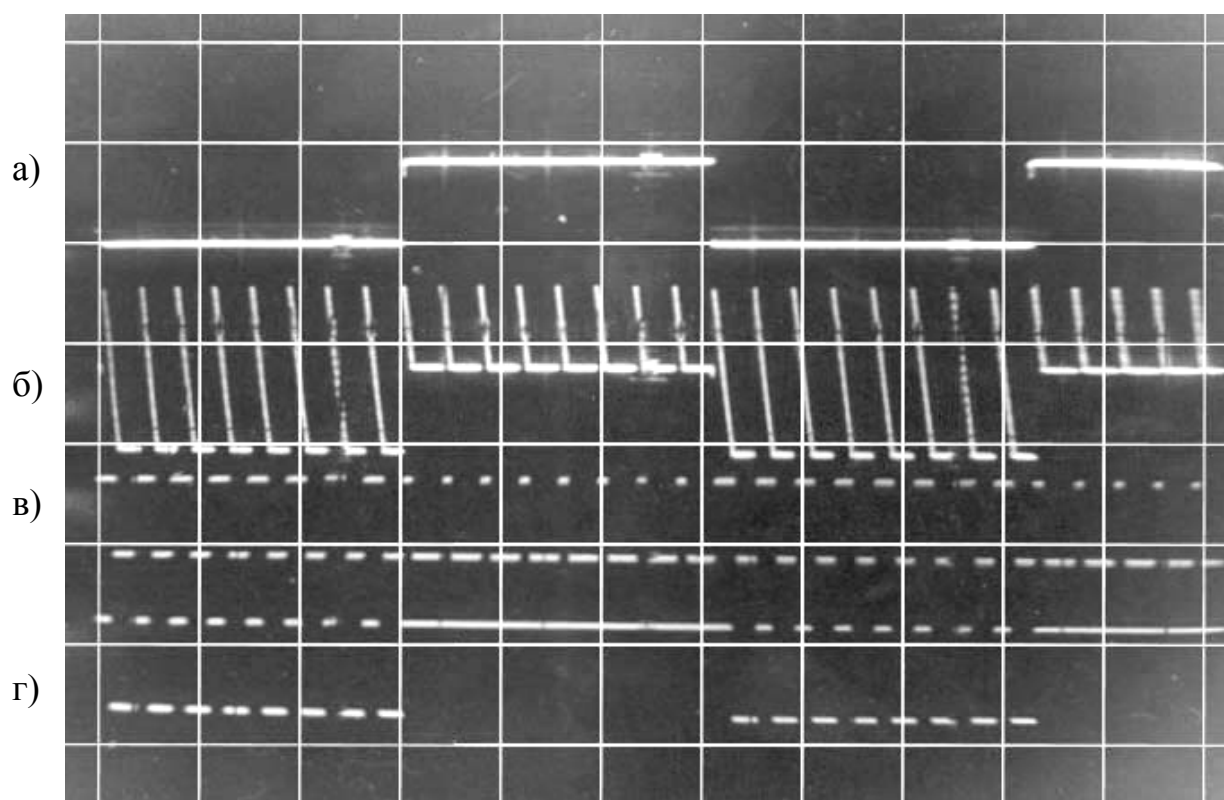


Рис. 6.20. Временные диаграммы работы АЦП цифрового согласованного фильтра

В качестве примера на рис. 6.20 приведены временные диаграммы работы АЦП цифрового согласованного фильтра. Принимаемый входной сигнал $s(t)$ представляет собой периодически повторяющуюся комбинацию сиг-

налов прямоугольной формы различной полярности, соответствующую чередованию символов канального алфавита 0101... (рис.6.20,а). В цифровом согласованном фильтре использован АЦП с единичными приближениями. На рис.6.20,б и в показаны временные диаграммы напряжений на выходах цифроаналогового преобразователя и компаратора, входящих в АЦП. Изменение уровня старшего разряда числа на выходе АЦП приведено на временной диаграмме рис.6.20,г.

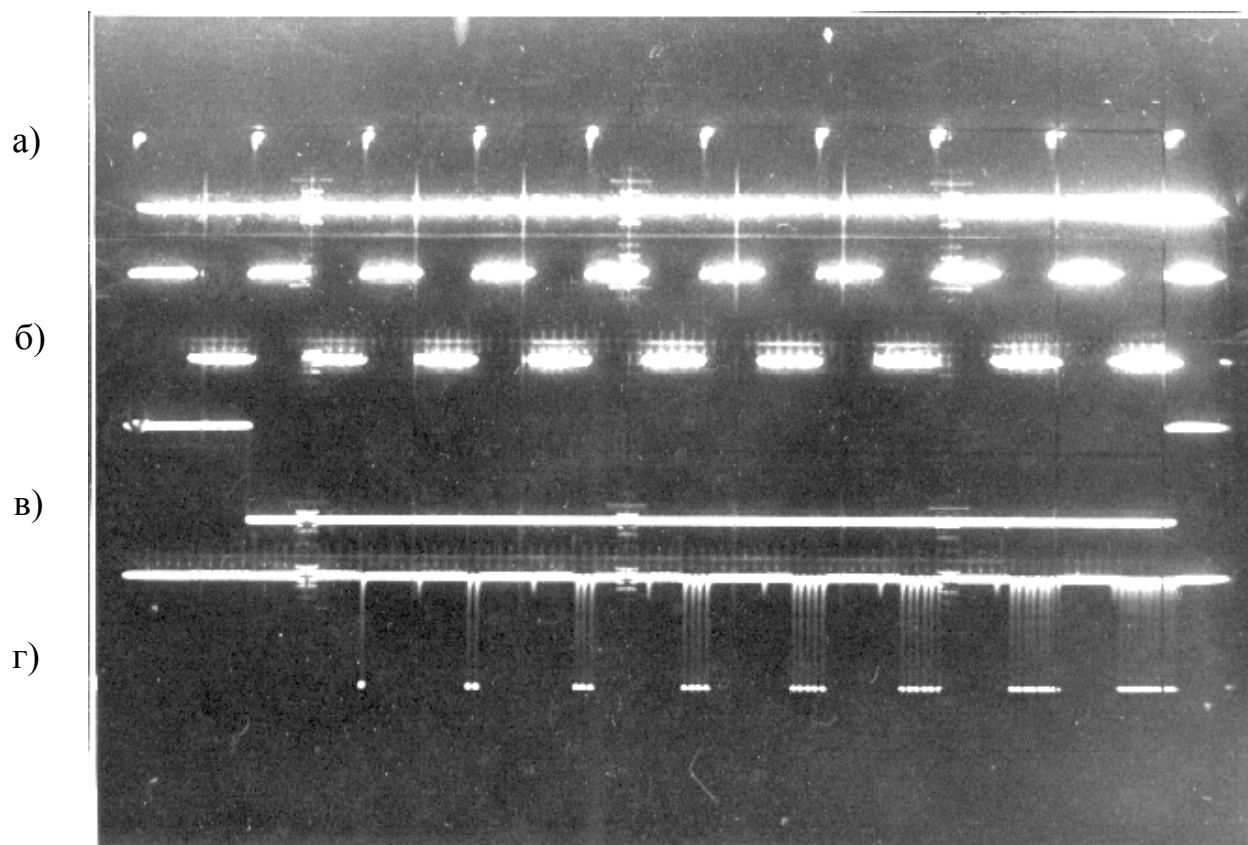


Рис.6.21 Временные диаграммы работы ОЗУ цифрового согласованного фильтра

Как отмечалось ранее, на каждом интервале дискретизации Δt происходит считывание, суммирование и запись чисел в оперативное запоминающее устройство. На рис.6.21,а-в приведены следующие временные диаграммы: последовательности импульсов, следующих с частотой дискретизации для $N=8$; последовательности импульсов записи и считывание чисел из ОЗУ; последовательности сигналов прямоугольной формы на входе АЦП соответственно.

Операция считывания чисел из ОЗУ иллюстрируется рис.6.21,г, где показана последовательность импульсов, получающаяся при считывании старшего разряда $(n+p)$ – разрядного числа. Как следует из рис.6.21,г, частота считывания чисел и, соответственно, частота тактовых импульсов суммато-

ра-накопителя для $N=8$ должна быть в 16 раз выше частоты дискретизации. Именно быстроедействие оперативного запоминающего устройства и сумматора-накопителя является ограничивающим фактором в практическом использовании цифрового согласованного фильтра.

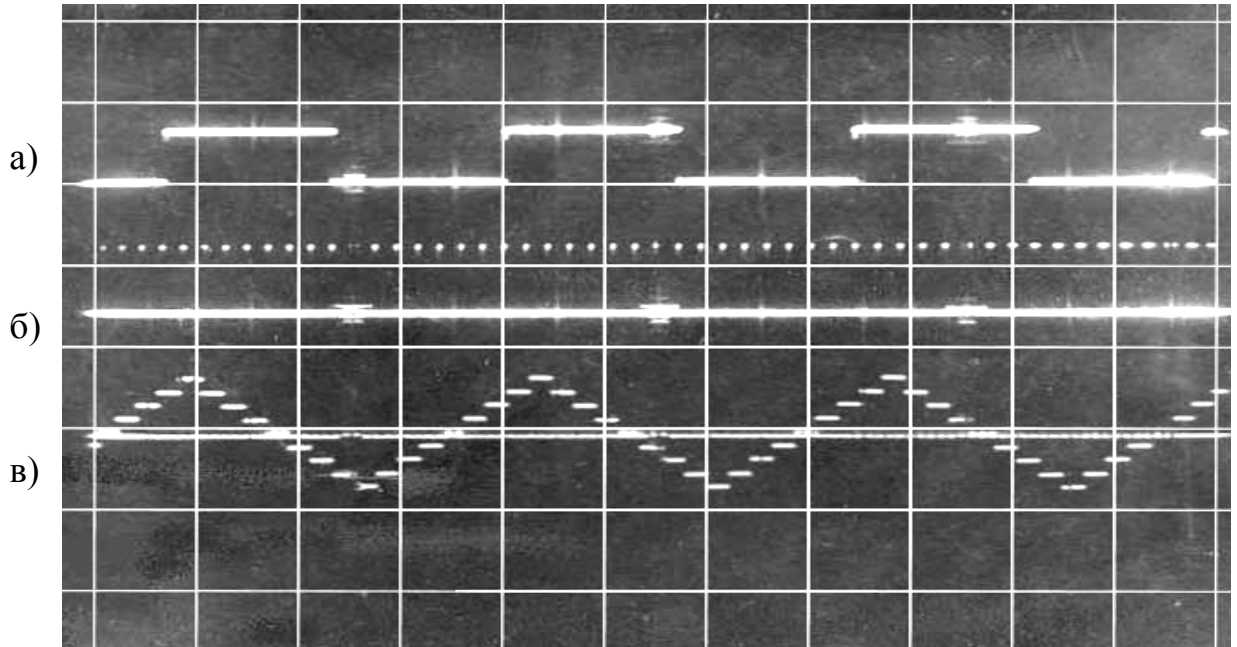


Рис. 6.22 Осциллограммы, иллюстрирующие работу сумматора-накопителя цифрового согласованного фильтра при приеме сигналов с прямоугольной формой огибающей

Пример работы сумматора-накопителя цифрового согласованного фильтра иллюстрируется осциллограммами на рис.6.22, где показаны: входное принимаемое колебание, представляющее собой последовательность сигналов прямоугольной формы (рис.6.22,а); последовательность таковых импульсов, следующих с частотой $f_T = 8/T$ (рис.6.22,б); форма напряжения на выходе цифрового сумматора-накопителя (рис.6.22,в). На входе цифрового согласованного фильтра последовательность сигналов представляет собой периодическую последовательность чередующихся сигналов длительностью T , имеющих различную полярность. Выход цифрового сумматора-накопителя для наглядности представлен в форме аналогового напряжения на выходе цифроаналогового преобразователя, подключенного к выходам регистра-накопителя (рис.6.18).

6.6 Устройство выбора максимального напряжения

Такие устройства широко применяются для построения демодуляторов, реализующих оптимальные правила приема сигналов, в соответствии с критерием максимального правдоподобия. В общем случае двоичного алфавита имеем [20]:

$$\max_i \left\{ \int_0^T A_x(t) A_i(t) dt \right\} > \delta(\mu),$$

где $\delta(\mu)$ - пороговое значение, зависящее от разности энергий сигналов $s_1(t)$ и $s_2(t)$ и коэффициента μ передачи по каналу связи; $A_x(t)$ и $A_i(t)$ для $i=1,2$ – входное колебание и ожидаемые сигналы $s_1(t)$ и $s_2(t)$.

Выражение, стоящее в фигурных скобках реализуется с помощью цифровых корреляторов или цифровых согласованных фильтров. Выбор максимального значения напряжения и сравнение его с пороговым значением может осуществляться с помощью устройства, схема которого приведена на рис.6.23.

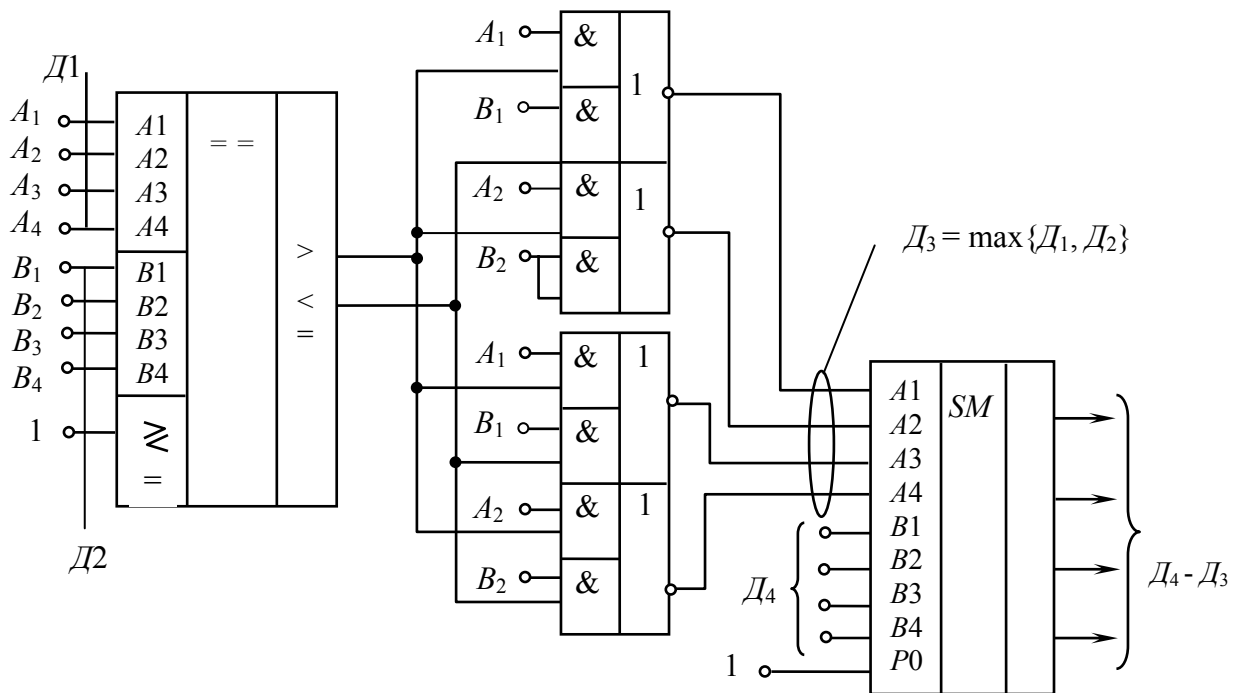


Рис. 6.23 Схема устройства выбора минимального значения напряжения

Устройство, осуществляющее выбор максимального из двух чисел D_1 и D_2 , состоит из схемы сравнения и мультиплексора, реализованного на логических элементах И-ИЛИ-НЕ. Для сравнения с величиной $\delta(\mu)$ применяется 4-разрядный полный сумматор, причем для реализации операции вычитания чисел $D_4 - D_3$, используется представление чисел в дополнительном двоичном коде.

Числа D_1 и D_2 , соответствуют значениям:

$$\int_0^T A_x(t) A_1(t) dt \quad \text{и} \quad \int_0^T A_x(t) A_2(t) dt,$$

а число D_4 – пороговому значению $\delta(\mu)$.

Такие устройства выбора максимального значения напряжения могут использоваться не только для сравнения двух чисел, но в более сложных случаях сравнения m чисел. Это может быть необходимо при реализации алгоритмов оптимального приема m -позиционных сигналов, где m – основание канального алфавита ($m > 2$). Выбор максимального из m чисел может быть осуществлен путем наращивания схемы, представленной на рис.6.23.

6.7 Устройство синхронизации радиомодемов

Демодуляторы радиомодемов дискретных сигналов содержат устройства синхронизации, служащие для формирования импульсов опроса при последетекторной обработке, временное положение которых определяется длительностью передаваемых символов канального алфавита. Взаимное расположение импульсов опроса, формируемых в демодуляторе местным генератором, и символов канального алфавита должно поддерживаться с точностью до нескольких процентов от длительности символа в течение весьма продолжительной работы демодулятора. Устройства синхронизации (УС) поддерживают фазовое расположение импульсов опроса и импульсов, получаемых из входного принимаемого колебания, представляющего собой сумму полезного сигнала и помех.

Наибольшее применение в телекоммуникационных системах нашли синхронные УС с обратной связью (замкнутые) без непосредственного воздействия на частоту местного генератора и использующие в качестве промежуточного преобразования делитель частоты, управляя которым, можно изменять фазу выходных тактовых импульсов. Из всех способов воздействия на делитель частоты широкое применение получил способ, при котором корректирование расхождения по фазе достигается изменением числа (добавлением или вычитанием) импульсов, подаваемых на вход промежуточного делителя частоты.

Замкнутые устройства синхронизации независимо от способа поддержания необходимого фазового расположения тактовых импульсов представляет собой разновидность устройств фазовой автоподстройки частоты (ФАПЧ). Чтобы частота снимаемых с выхода делителя тактовых импульсов f_T равнялась заданной $f_T = (f_T)_3$, частота задающего генератора f_G уменьшается делителем частоты в k раз:

$$f_T = (f_T)_3 = f_G / k = 1/k\tau,$$

где τ – период колебаний местного генератора. Если на вход делителя, кроме импульсов генератора, подать дополнительный импульс, то фаза выходных тактовых импульсов сместится в сторону опережения на величину, рав-

ную периоду τ колебаний местного генератора. При исключении одного из импульсов, подаваемых на делитель, фаза выходных импульсов сместится в сторону отставания также на τ . Структурная схема устройства синхронизации с дискретным управлением приведена на рис.6.24.а. Она содержит генератор Г, фазовый детектор, У – усредняющее устройство; Д – делитель частоты; УУ – устройство управления добавлением и вычитанием импульсов; ВУ – входное устройство, а на рис.6.24,б и в – временные диаграммы работы.

Входное устройство формирует импульсы, совпадающие по времени с моментами перехода одного значения входного сигнала к другому, и определяется в основном структурой (видом модуляции) принимаемых сигналов. Так, при передаче сообщений с помощью фазоманипулированных сигналов в качестве входных сигналов входного устройства используют сигналы на выходе детектора, и входное устройство регистрирует моменты перехода огибающей последовательности сигналов через нулевой уровень.

На входы «е» и «ж» фазового детектора поступают импульсные последовательности с выходов делителя, сдвинутые относительно друг друга на угол π . На вход «и» фазового детектора поступает последовательность коротких (длительностью $\tau_{и} \ll \tau$) импульсов с выхода входного устройства. На выходах фазового детектора в случае опережения или «отставания» этих импульсов относительно импульсов с выходов делителя формируются последовательности «в1» и «г1». Эти последовательности поступают на входы реверсивных счетчиков усредняющего устройства (рис.6.24,а), количество разрядов которых равно l . Если входные импульсы «опережают» по фазе импульсы опорной последовательности, то после усреднения на выходе «в2» появляется импульс управления, с помощью которого в устройстве управления происходит добавление импульса в последовательность «а». В случае «отставания» входных импульсов сигнал управления появляется на выходе «г2», и в устройстве управления происходит вычитание импульса из любой последовательности «а». Такая процедура будет повторяться до тех пор, пока импульсы входных и опорных последовательностей не совпадут по времени.

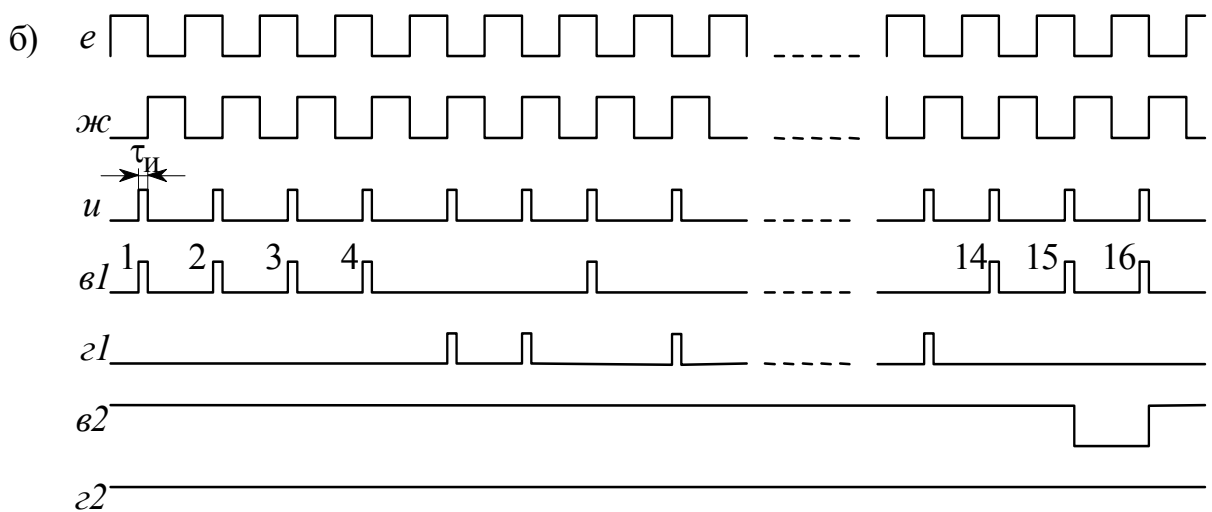
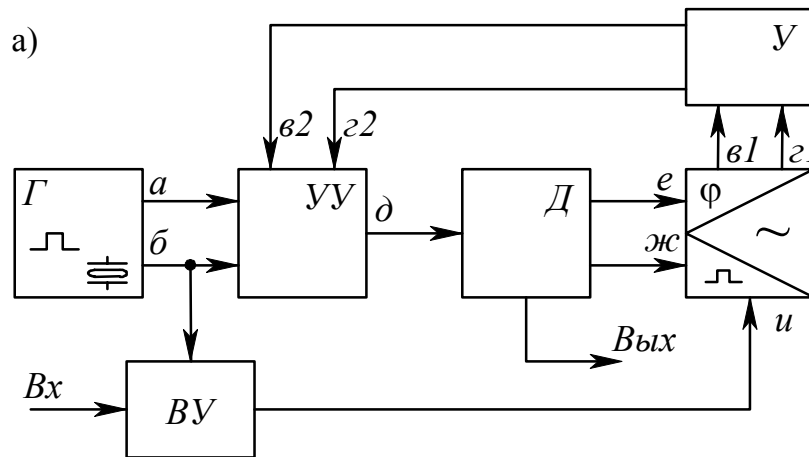
На рис. 6.25 приведена принципиальная схема устройства синхронизации, выполненного на цифровых последовательностных устройствах. Временные диаграммы работы устройства синхронизации представлены на рис.6. 24,в. Устройство управления осуществляет добавление или вычитание импульсов в последовательностях «а» и «б». На выходах «а» и «б» (рис. 6.24,в) местного кварцевого генератора формируются импульсные последовательности, сдвинутые относительно друг друга на π (рис. 6.24,в). На

этом же рисунке изображена вспомогательная инвертированная импульсная последовательность «а» (на рис.6.24,*а* не показана). Обе последовательности с выходов «а» и «б» поступают на входы устройства управления, которое выполняет следующие логические операции:

- пропускает импульсную последовательность «а» без изменения на выход «д» (рис. 6.24,*а*), если на входах «в» и «г» (рис.6.24,*в* и 6.25) присутствует напряжение логической 1;

- вычитает один импульс из импульсной последовательности «а» и пропускает ее на выход «д», если на входе «в» присутствует напряжение логического 0, а на выходе «г» – логической 1;

- добавляет один импульс к импульсной последовательности «а» и пропускает ее на выход «д», если на входе «в» присутствует напряжение логической 1, а на выходе «г» – логического 0.



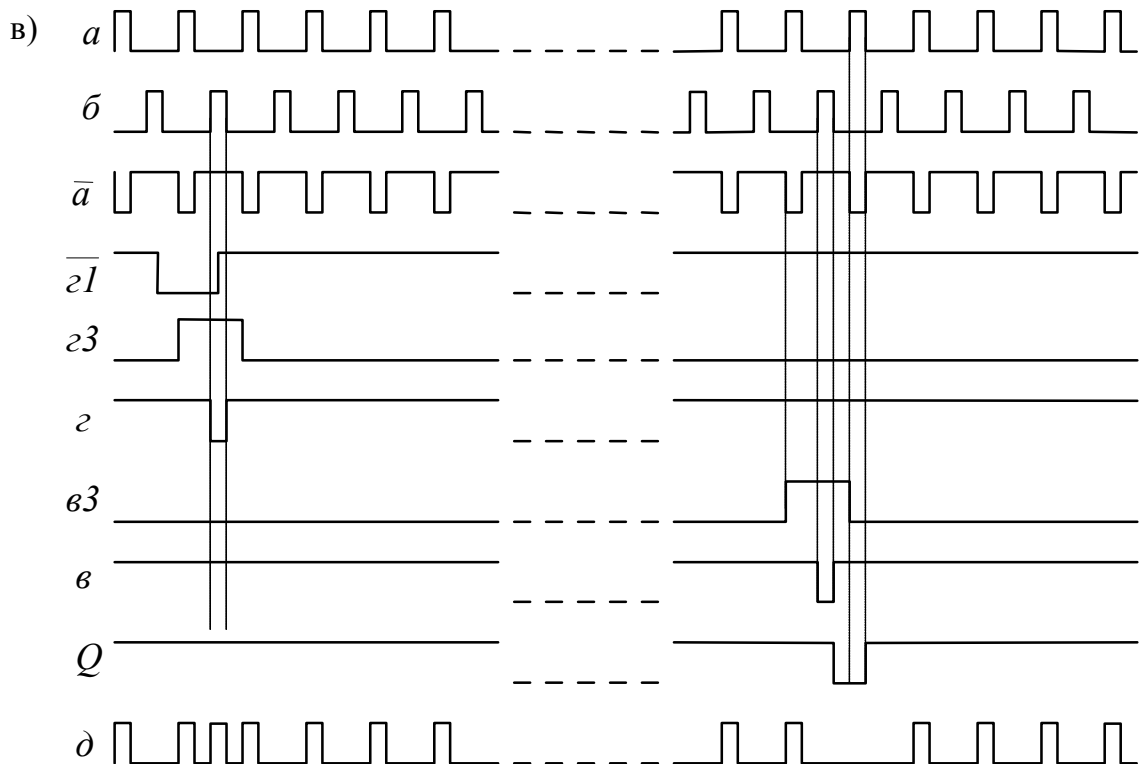


Рис. 6.24. Структурная схема устройства синхронизации и временные диаграммы работы

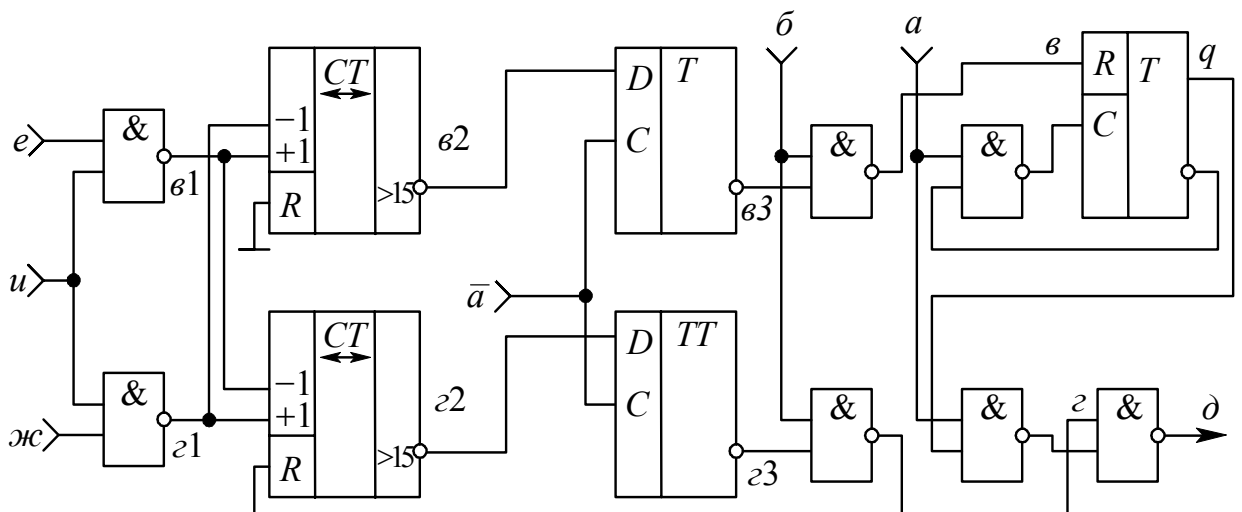


Рис. 6.25. Принципиальная схема устройства синхронизации

Усреднение импульсов осуществляется с помощью двух реверсивных счетчиков с коэффициентом пересчета, равным 16. Каждый из счетчиков расположен в устройстве управления перед схемами добавления или вычитания. Выбор коэффициента пересчета зависит от требований к времени вхождения в синхронный режим работы демодулятора сигналов, что, в свою очередь, определяется характеристиками системы передачи сообщений.

Если на выходах «г2» и «в2» (рис.6.25) появляются сигналы управления на вычитание или добавление импульсов (рис.6.24,в и 6.25), то при помощи логических элементов И и триггеров, входящих в устройство управления, производится формирование импульсов добавления или вычитания (рис. 6.25).

6.8 Дискретные фазовращатели

Для систем фазовой автоподстройки частоты используются дискретные фазовращатели, функциональные схемы которых представлены на рис. 6.26 [19].

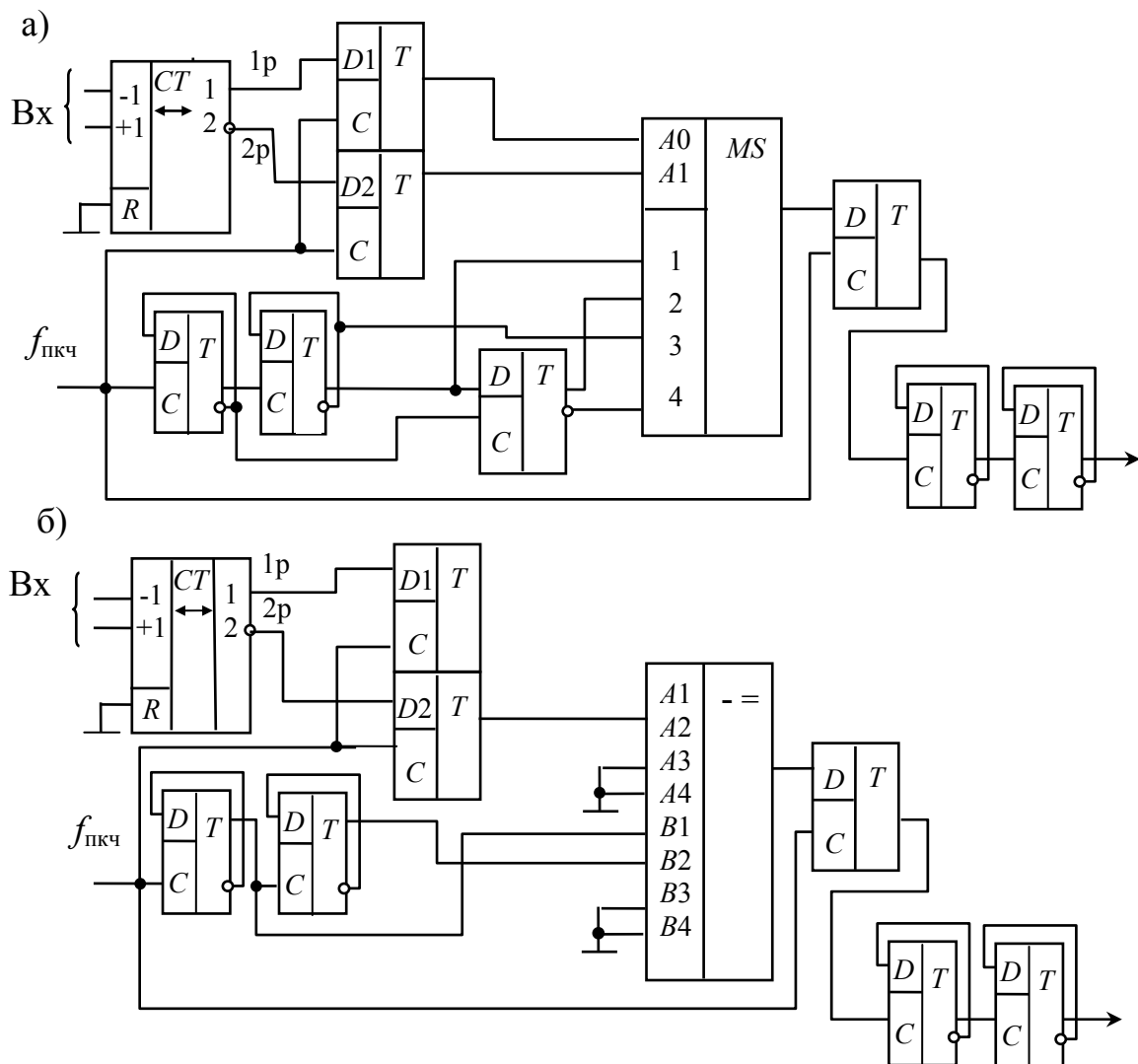


Рис. 6.26. Схемы дискретных фазовращателей на основе реверсивных счетчиков

Такой дискретный фазовращатель, изображенный на рис. 6.26,а, работает следующим образом. Сигнал с преобразователя код-частота $f_{пкч}$ подается на вход делителя на четыре. На выходе этого делителя формируются четыре одинаковые последовательности импульсов с частотой $f_{пкч}/4$, сдвинутых по

фазе относительно друг друга на $\pi/2$. Эти последовательности поступают на мультиплексор, где из четырех возможных выбирается одна последовательность в соответствии с кодом на адресном входе мультиплексора. Этот код снимается с реверсивного счетчика, на вход которого поступают импульсы, управляющие работой фазовращателя.

Импульсные последовательности на мультиплексор подаются таким образом, чтобы увеличение (уменьшение) на единицу числа в реверсивном счетчике приводило к появлению на выходе мультиплексора последовательности, сдвинутого по фазе на угол $\pi/2$ ($-\pi/2$) относительно последовательности на этом выходе в предшествующий момент времени.

Для того, чтобы избежать появления ложных импульсов, код управления мультиплексором и импульсную последовательность с его выхода необходимо синхронизировать сигналом с выхода преобразователя код-частота с помощью D -триггеров.

Для уменьшения шага по фазе на выходе дискретного фазовращателя используется делитель с коэффициентом деления 4.

В дискретном фазовращателе, изображенном на рис. 6.26,б, как и в предыдущей схеме, управляющие импульсы поступают на реверсивный счетчик, с выхода которого двухразрядный код поступает на схему сравнения. На другой вход схемы сравнения поступает текущий двухразрядный код с выхода счетчика, осуществляющего деление на 4 частоты импульсной последовательности с выхода преобразователя код-частота. При равенстве этих кодов на выходе схемы сравнения появляется импульс. Частота следования этих импульсов равна $f_{\text{пкч}}/4$. При изменении на единицу младшего разряда кода реверсивного счетчика происходит сдвиг последовательности по фазе на угол $+\pi/2$ ($-\pi/2$) относительно последовательности на выходе схемы сравнения. Для того чтобы избежать появления ложных импульсов, код, поступающий с реверсивного счетчика, и последовательность на выходе схемы сравнения необходимо синхронизировать с помощью D -триггера с последовательностью с выхода преобразователя код-частота. Как и в предыдущей схеме, для уменьшения шага по фазе на выходе дискретного фазовращателя используется делитель частоты с коэффициентом деления на четыре.

На рис. 6.27,а показана схема [22] цифрового фазовращателя со схемой временной привязки, выполненной на D -триггерах. Работу схемы поясняют временные диаграммы (рис. 6.27,б). Выходная последовательность фазовращателя Q_3 подается на тактовый вход счетчика по модулю K . При значении $Q_5 = 0$ период выходного сигнала счетчика по модулю K будет равен

$T_0 = 5KT_H$ где T_H . Если за K циклов сигнал y изменится m раз с 0 на 1, то период выходного сигнала счетчика $T = (5K \pm m)T_H = T_0 \pm mT_H$, т.е. положение момента изменения с 1 на 0 выходного сигнала счетчика по модулю K изменится на величину $\pm mT_H$. Чем выше частота тактовой последовательности H , тем выше точность цифрового фазовращателя. Если сигнал y изменяется с 0 на 1 с частотой f_C , то скорость перемещения фронтов выходного сигнала счетчика по модулю K относительно сигнала с периодом T_0 равна $v_C = \pm T_H \cdot f_C$.

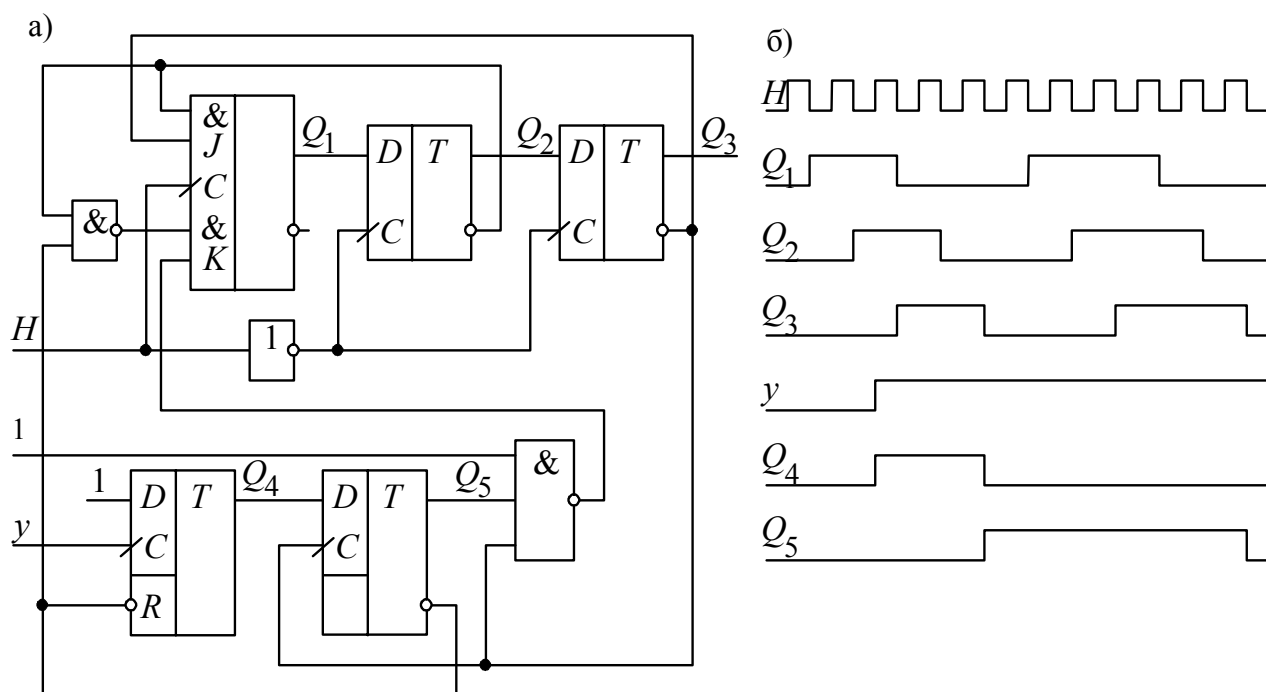


Рис. 6.27. Схема дискретного фазовращателя на основе D -триггеров и временные диаграммы работы

6.9 Цифровые фазовые и частотные дискриминаторы

Фазовые и частотные дискриминаторы применяются в радиолокационных и радионавигационных устройствах для измерения разностей фаз колебаний, временных положений радиолокационных импульсных сигналов, при фазовых методах приема сигналов, в дифференциально-дальномерных системах.

Цифровой фазовый дискриминатор [19] состоит из амплитудных ограничителей, цифрового фазового детектора и выходных усилителей. На рис. 6.28 приведена функциональная схема цифрового фазового детектора с дискриминационной характеристикой прямоугольного вида.

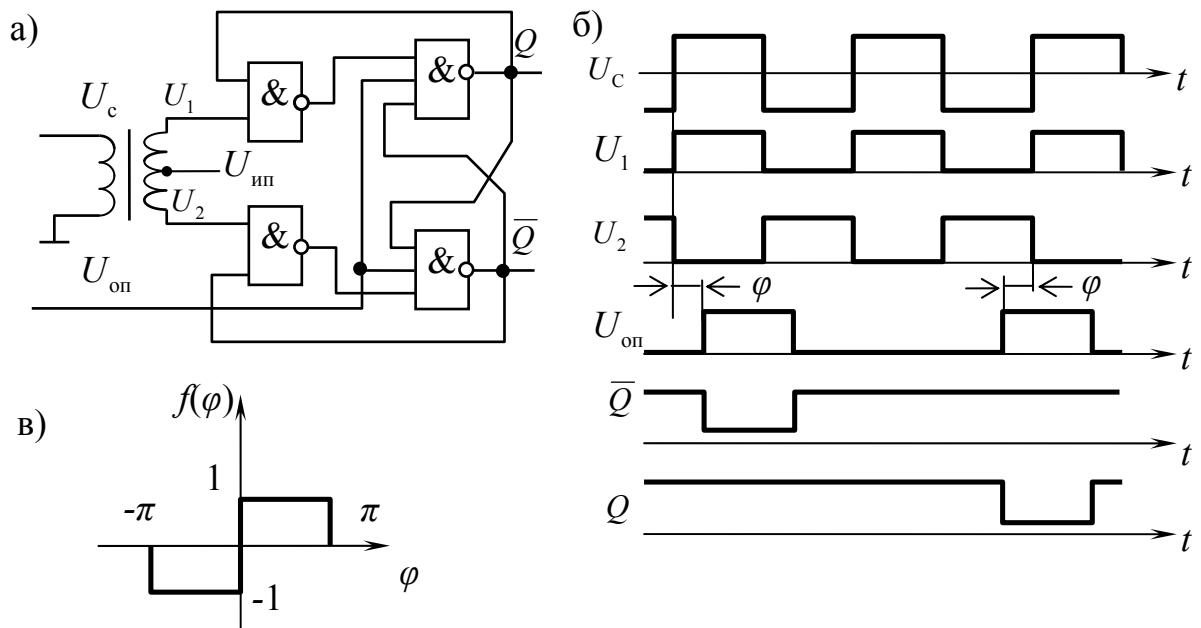


Рис. 6.28. Цифровой фазовый детектор

На вход трансформатора подается входной сигнал U_c , представляющий собой периодическую последовательность импульсов прямоугольной формы (рис.6.28,б). Средняя точка вторичной обмотки трансформатора подключена к источнику питания $U_{инп}$ с напряжением, равным уровню напряжения логической единицы. Амплитуда импульсов сигнала во вторичной обмотке на парафазных выходах трансформатора U_1, U_2 подбирается равной уровню логической единицы.

Опорный сигнал $U_{оп}$ (рис.6.28,б) может иметь форму входного сигнала U_c или представлять собой периодическую последовательность стробирующих импульсов ($U_{оп}$ на рис.6.28,б).

В интервалах времени между стробирующими импульсами при любых уровнях напряжений входного сигнала U_c , напряжения на выходах цифрового фазового детектора равны уровню логической единицы, т.е. $\bar{Q} = Q = 1$. На входах 2-х входных логических элементов И-НЕ также присутствует напряжение логического нуля. На выходах этих элементов напряжение также соответствует логическому нулю. Цифровой фазовый детектор позволяет проводить автоматический контроль пропадания уровня входного сигнала.

Так, если в момент появления первого стробирующего импульса (рис.6.28,б) фазовый сдвиг φ между последовательностями входного U_c и опорного сигнала $U_{оп}$ оказывается положительным, сигнал на выходе \bar{Q}

принимает значение логического нуля, которое сохраняется до окончания воздействия стробирующего импульса независимо от значения фазы $[0 < \varphi < \pi]$. Если фазовый сдвиг оказывается отрицательным (второй стробирующий импульс на рис. 6.28, б), сигнал логического нуля появляется на выходе Q логического элемента И-НЕ и сохраняет свое значение в течение времени, равного длительности стробирующего импульса, независимо от абсолютного значения фазы φ в пределах $-\pi < \varphi < 0$.

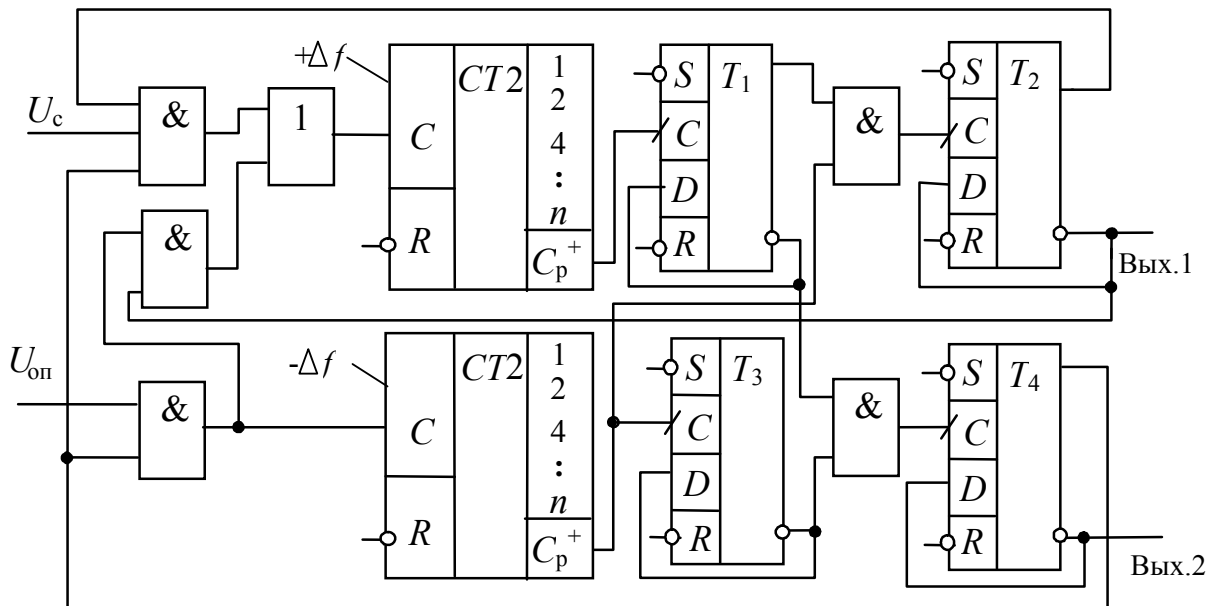


Рис. 6.29. Цифровой частотный дискриминатор

Частотные дискриминаторы используются для определения стабильности опорных генераторов радиопередающих устройств, для настройки радионавигационной аппаратуры, а также в бортовых устройствах дальномерных систем. Одна из схем цифрового частотного дискриминатора приведена на рис. 6.29.

Перед началом измерений на входы R счетчиков и входы S триггеров в течении короткого интервала времени поступает сигнал логического нуля. Счетчики обнуляются, а триггеры устанавливаются в состояние логической единицы на прямых (не инверсных) выходах. После этого начинается подсчет периодов исследуемого U_c и опорного $U_{оп}$ сигналов для определения разности их частот $\Delta f = f_c - f_{оп}$.

В случае $f_c > f_{оп}$ при переполнении первого счетчика $CT2$ триггер T_1 переводится в состояние логического нуля. При этом состояние триггер T_2 не из-

менится и первый счетчик $CT2$ начнет снова счет периодов U_c . После переполнения второго счетчика $CT2$ триггер T_3 переходит в состояние логического нуля, что приведет к появлению сигнала логической единицы на входе C триггера T_4 , который переключится в состояние логического нуля. Этим уровнем логического нуля на выходе T_4 через входные логические элементы И запрещается поступление импульсов на входы счетчиков и счет прекратится.

Появление уровня логической единицы на «Вых.2» разрешает считывание информации. Напряжение логического 0 на «Вых.1» указывает на то, что разность частот положительна и информацию следует снимать с выхода первого двоичного счетчика. Если в нем записано число ΔC , а емкость счетчиков составляет $C=2^n$, то $\Delta f = f_{оп} \Delta C / 2^n$.

Пусть теперь $f_c < f_{оп}$. Первым заканчивает счет второй двоичный счетчик, сигнал с выхода которого переводит триггер T_3 , а через логический элемент И, триггер T_2 , в состояние логического нуля. При этом входной логический элемент И прекращает пропускать входной сигнал U_c , а логический элемент И, включенный между триггерами T_3 и T_4 , а также логический элемент И, подключенный ко второму входу элемента ИЛИ, будут иметь на выходах напряжение логической единицы. Второй двоичный счетчик заново начинает счет периодов $U_{оп}$. Первый счетчик прекращает счет периодов входного сигнала U_c и переходит на счет периодов опорного сигнала $U_{оп}$ до переполнения. При переполнении первого счетчика триггер T_1 «опрокидывается», а вслед за ним через логический элемент «опрокидывается» и триггер T_4 . Работа частотного дискриминатора прекращается, так как на входные логические элементы И поступает напряжение логического нуля с выхода триггера T_4 и закрывает их. В данном случае на «Вых.1» и «Вых.2» присутствует напряжение логической единицы. Это значит, что разность частот отрицательна и съем информации разрешен со второго двоичного счетчика. В этом случае $\Delta f = -f_{оп} \Delta C / 2^n$.

Действительно, второй двоичный счетчик переполнится за время $T = 2^n / f_{оп}$ и затем еще работает в течение времени $\Delta t = \Delta C / f_{оп}$. За время T первый счетчик сосчитает число $C_1 = T f_c$; за время Δt число $C_2 = \Delta t f_c$, так как в это время он считает периоды опорного сигнала. Поскольку первый счетчик переполняется, то

$$C_1 + C_2 = 2^n = T f_c + \Delta t f_{оп} = 2^n (f_{оп} + \Delta f) / f_{оп} + \Delta C f_{оп} / f_{оп},$$

Из этих выражений следует, что погрешность измерения разности частот минимальна при $\Delta C = 1$ и составляет $\pm(f_{оп} / 2^n)$.

6.10. Устройства контроля, кодирования и декодирования

6.10.1 Устройства контроля по модулю 2

Устройства контроля по модулю 2 используются в различных радиотехнических системах передачи цифровых пакетных сообщений по проводным и радиоканалам, в системах сбора данных при пространственном мониторинге, в мобильных телекоммуникационных системах и пр. В устройствах контроля по модулю 2 каждый передаваемый в канале пакет сообщений дополняется контрольным разрядом, значение которого подбирается так, чтобы сделать четным число единиц в пакете. При одиночной ошибке в пакете число единиц и нулей меняется, что и обнаруживается устройством контроля. При двойной ошибке число единиц и нулей также меняется, однако суммарное число единиц остается четным и, соответственно, ошибка не выявляется. Таким образом устройство контроля по модулю 2 оказывается эффективным в каналах передачи пакетных сообщений, где вероятность единичной ошибки передаваемого символа много больше, чем вероятность двойной или групповой ошибки.

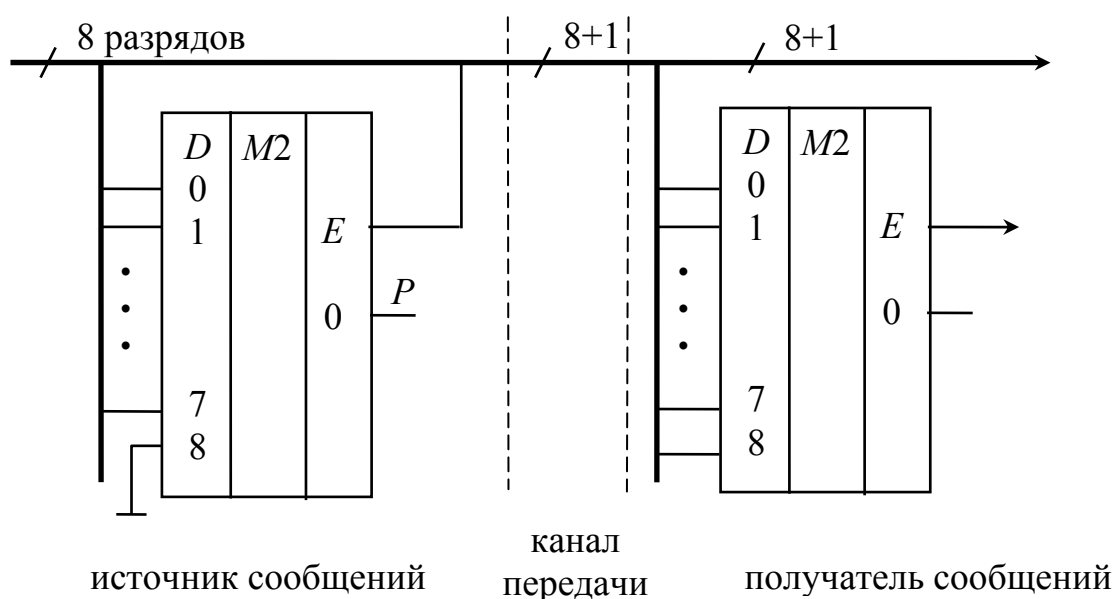


Рис. 6.30. Устройство контроля по модулю 2

Устройство контроля по модулю 2 может быть реализовано с помощью совокупности сумматоров по модулю 2, построенных на логических элементах «Исключающее ИЛИ» или интегральных схем, называемых схемами свертки, в состав которых входят такие сумматоры.

Типичная схема устройства контроля по модулю 2, применяемая в проводных каналах, приведена на рис. 6.30. В этом устройстве используются ин-

тегральные схемы КР1533ИП5, имеющие 9 входов (0 ...8). На выходах E (Even) формируется уровень логической 1, если на входах D присутствует четное число единиц. На выходах O (Odd) уровень логической единицы появляется при наличии на входах D нечетного числа единиц. В источнике сообщений реализуется свертка передаваемого байта сообщений с 9-ым контрольным разрядом. Таким образом по каналу передаются 9-ти разрядные пакеты двоичных сообщений с дополнительным контрольным разрядом. Получатель сообщений производит с помощью интегральной схемы контроль числа единиц в пакете сообщений. Если при приеме на выходе E присутствует логический 0, что соответствует присутствию на входах D нечетного числа единиц, такой пакет сообщений считается переданным с ошибкой и исключается из дальнейшей обработки. Таким образом устройство контроля по модулю 2 позволяет лишь обнаруживать одиночные ошибки в пакете сообщений.

6.10.2 Устройства кодирования и декодирования кодов Хэмминга

Коды Хэмминга позволяют не только обнаруживать, но и исправлять одиночные ошибки. Такие коды относятся к линейным систематическим кодам, в которых проверочные разряды формируются линейным преобразованием информационных разрядов. Коды Хэмминга (n, k) , где n – общее число разрядов кода, k – число информационных разрядов a_i ($i = 1, 2, \dots, k$), имеют $r = n - k$ число проверочных разрядов b_i ($i = 1, 2, \dots, r$). Рассмотрим код Хэмминга $(15, 11)$ с поэлементным формированием проверочных разрядов [22].

Для определения b_1, b_2, \dots, b_r в источнике сообщений пользуются операторами R_1, \dots, R_r , так что $b_i = R_i\{a_i\}$, где $\{a_i\}$ – множество информационных разрядов кодовой комбинации. Получатель сообщений по принятой совокупности информационных разрядов $\{a_i^*\}$ вычисляет с помощью тех же операторов R_i новые проверочные $b'_i = R_i\{a_i^*\}$ разряды (знак * означает принятые разряды кодовой комбинации, которые из-за ошибок в канале могут отличаться от переданных). Далее принятые проверочные разряды $b^*_1, b^*_2, \dots, b^*_r$ складываются по модулю 2 с вычисленными b'_1, b'_2, \dots, b'_r :

$$\oplus \frac{\begin{matrix} b_1^* & b_2^* & \dots & b_r^* \\ b'_1 & b'_2 & \dots & b'_r \end{matrix}}{\begin{matrix} C_1 & C_2 & \dots & C_r \end{matrix}} \quad (6.26)$$

В результате сложения получается вектор ошибки $C_1 C_2 \dots C_r$. Допустим, что все a_i приняты верно, тогда $\{a_i\} = \{a_i^*\}$ и, $\{b_i^*\} = \{b_i\}$, т.е. вычисленные

проверочные разряды такие же, как и в источнике сообщений. Если при приеме проверочных разрядов также не произошло ошибок, то

$$\{b_i'\} = \{b_i^*\} = \{b_i\}.$$

В этом случае все разряды вектора ошибки (синдрома) будут представлены нулями:

$$C_1 C_2 \dots C_r \Rightarrow \underbrace{00 \dots 0}_r. \quad (6.27)$$

Если при приеме кодовой комбинации происходят ошибки, то в разрядах синдрома (6.27) появятся единицы. Запишем по порядку в виде таблицы числа, представляющие синдром (табл.6.4).

Таблица 6.4

Элементы синдрома				Элементы кодовой комбинации с ошибкой
C_1	C_2	C_3	C_4	
0	0	0	1	b_4
0	0	1	0	b_3
0	0	1	1	a_1
0	1	0	0	b_2
0	1	0	1	a_2
0	1	1	0	a_3
0	1	1	1	a_4
1	0	0	0	b_1
1	0	0	1	a_5
1	0	1	0	a_6
1	0	1	1	a_7
1	1	0	0	a_8
1	1	0	1	a_9
1	1	1	0	a_{10}
1	1	1	1	a_{11}

Построим алгоритм \mathcal{R} так, чтобы он включал в себя информационные разряды, ошибка в которых приводила бы к появлению единицы в младшем (C_4) разряде синдрома. Наличие единицы в младшем разряде соответствует проверочному элементу b_4 . Кроме того, эта единица есть в синдромах, соответствующих информационным разрядам $a_1, a_2, a_4, a_5, a_7, a_9, a_{11}$. Таким образом,

$$\mathcal{R}_1 : b_4 = a_1 \oplus a_2 \oplus a_4 \oplus a_5 \oplus a_7 \oplus a_9 \oplus a_{11}.$$

Если в одном из этих разрядов произойдет ошибка, то это неизбежно приведет к изменению b_4 с нуля на единицу или с единицы на нуль. Следовательно, когда на приемной стороне определим b'_4 , то окажется, что $b'_4 \neq b_4$. Это приведет к наличию «1» в первом младшем разряде синдрома.

Для формирования проверочного разряда b_3 , в синдроме которого единица стоит на второй позиции (второй разряд), отбираем информационные разряды, имеющие единицу во втором разряде своих синдромов, т.е. $a_1, a_3, a_4, a_6, a_7, a_{10}, a_{11}$:

$$\mathcal{R}_2 : b_3 = a_1 \oplus a_3 \oplus a_4 \oplus a_6 \oplus a_7 \oplus a_{10} \oplus a_{11}.$$

$$\mathcal{R}_3 : b_2 = a_2 \oplus a_3 \oplus a_4 \oplus a_8 \oplus a_9 \oplus a_{10} \oplus a_{11}.$$

$$\mathcal{R}_4 : b_1 = a_5 \oplus a_6 \oplus a_7 \oplus a_8 \oplus a_9 \oplus a_{10} \oplus a_{11}.$$

Процесс формирования кода Хэмминга (15, 11) приведен на рис. 6.31.

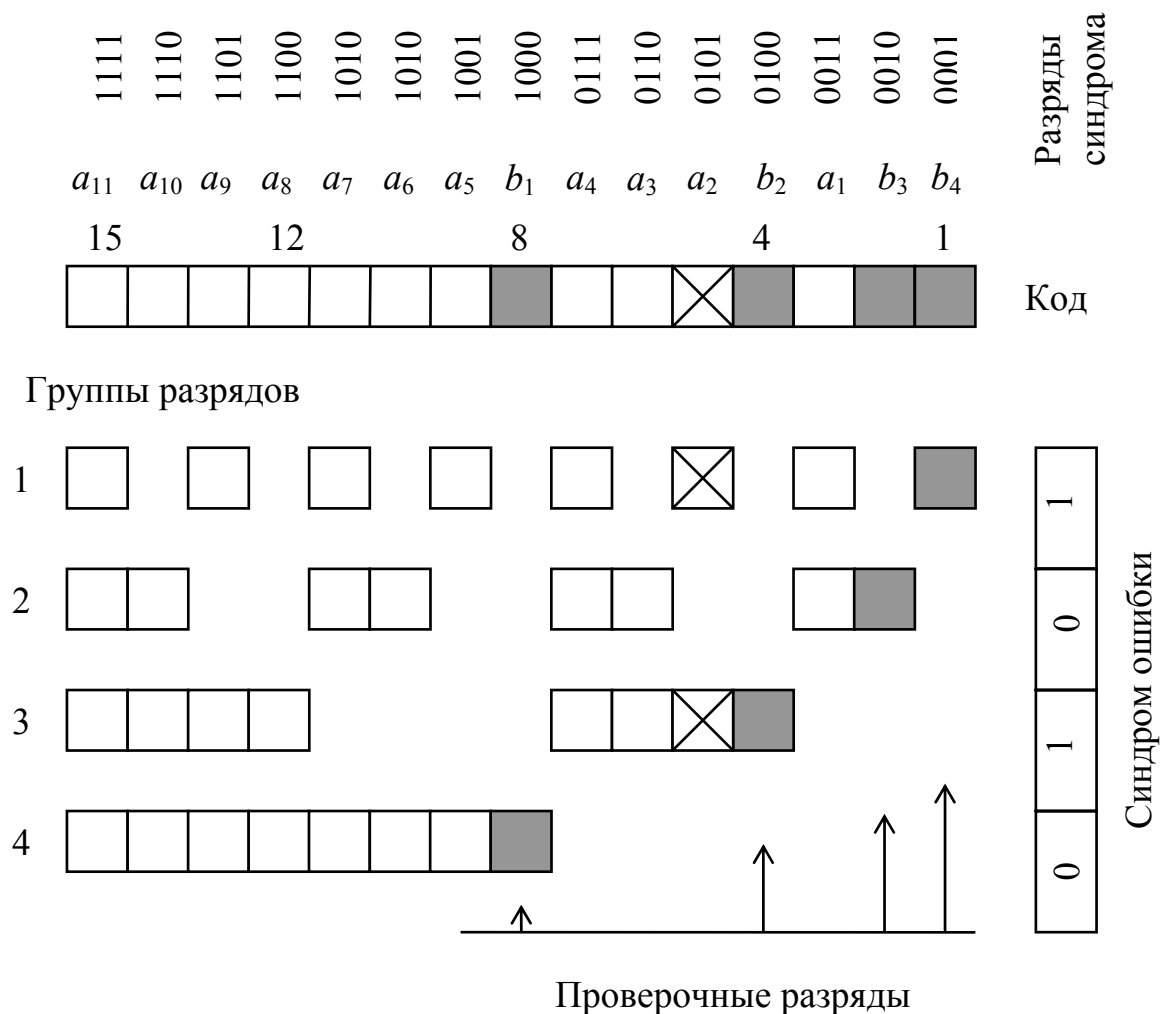


Рис. 6.31. Структура кода Хэмминга (15, 11)

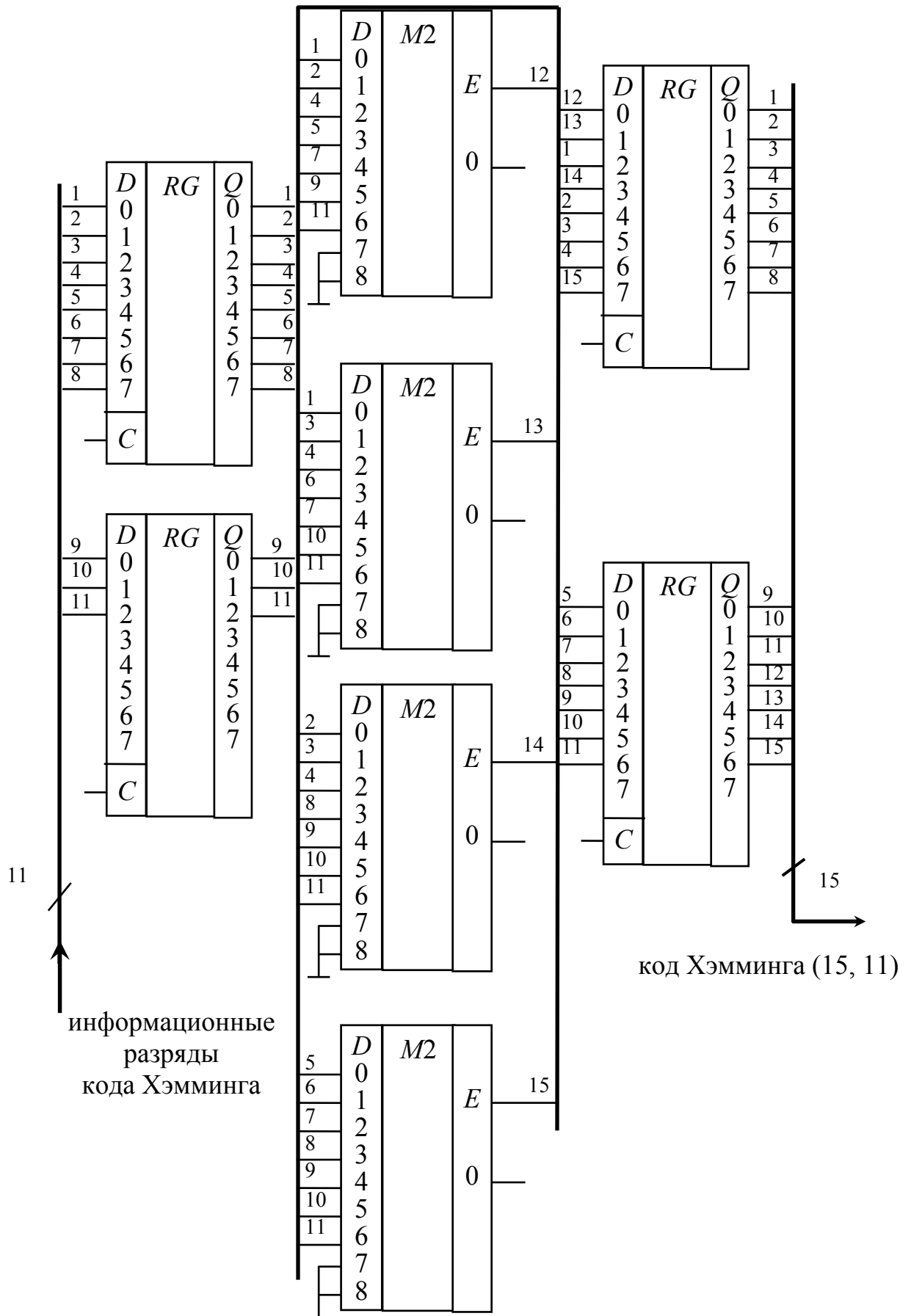


Рис. 6.32. Устройство формирования кода Хэмминга (15,11)

Как отмечалось ранее, код Хэмминга (15, 11) состоит из одиннадцати информационных разрядов a_1, a_2, \dots, a_{11} и четырех проверочных разрядов b_1, b_2, b_3, b_4 . Проверочные разряды показаны размещенными в коде Хэмминга не компактным массивом, а вперемежку с информационными разрядами. На рис.6.31 проверочные разряды зачернены.

На рис.6.32 показано устройство кодирования, формирующее код Хэмминга (15, 11) в источнике сообщений. Семиразрядные группы информационных разрядов, скомпонованные в соответствии с рис.6.31, поступают на семивходовые устройства контроля по модулю 2. Результаты свертки каждой группы с контрольным разрядом записываются в проверочные разряды кода Хэмминга. Проверочные разряды b_1, b_2, b_3, b_4 формируются на выходах E сумматоров по модулю 2 и поступают в выходной параллельный регистр.

Функциональная схема устройства декодирования кода Хэмминга (15, 11) представлена на рис. 6.33. Разряды принятого кода Хэмминга через входные регистры сдвига поступают на 4 устройства контроля по модулю 2. В отличие от устройства кодирования (рис.6.31) в устройстве декодирования в каждой свертке обрабатывается вся группа вместе с ее контрольным разрядом. На выходах E устройств контроля по модулю 2 формируются проверочные разряды b'_1, b'_2, b'_3, b'_4 . Если при передаче кода Хэмминга в одном из его разрядов произошла ошибка (на рис.6.31 в качестве примера крестиком отмечен поврежденный пятый разряд), то будет зафиксировано нарушение четности именно в тех контрольных группах, в состав которых входит поврежденный разряд. В результате код синдрома ошибки укажет номер поврежденного разряда принятого кода Хэмминга. Так, код синдрома 0101 указывает на поврежденный 5-ый разряд. Восстановление передаваемых информационных разрядов происходит в выходном блоке (рис.6.33) устройств контроля по модулю 2.

Физическое расположение разрядов в коде Хэмминга при его передаче или хранении может отличаться от показанного на рис.6.31. Например, информационные разряды могут образовывать один компактный массив, проверочные – другой.

Для исправления одиночной ошибки код Хэмминга имеет минимально возможную избыточность [21, 22]. Чтобы закодировать r разрядами номер любого из n возможных поврежденных разрядов кода Хэмминга, оставив при этом один из кодов синдрома – код «все нули» для идентификации отсутствия ошибки, должны выполняться соотношения:

$$n = k + r; n = 2^r - 1 \quad (6.28)$$

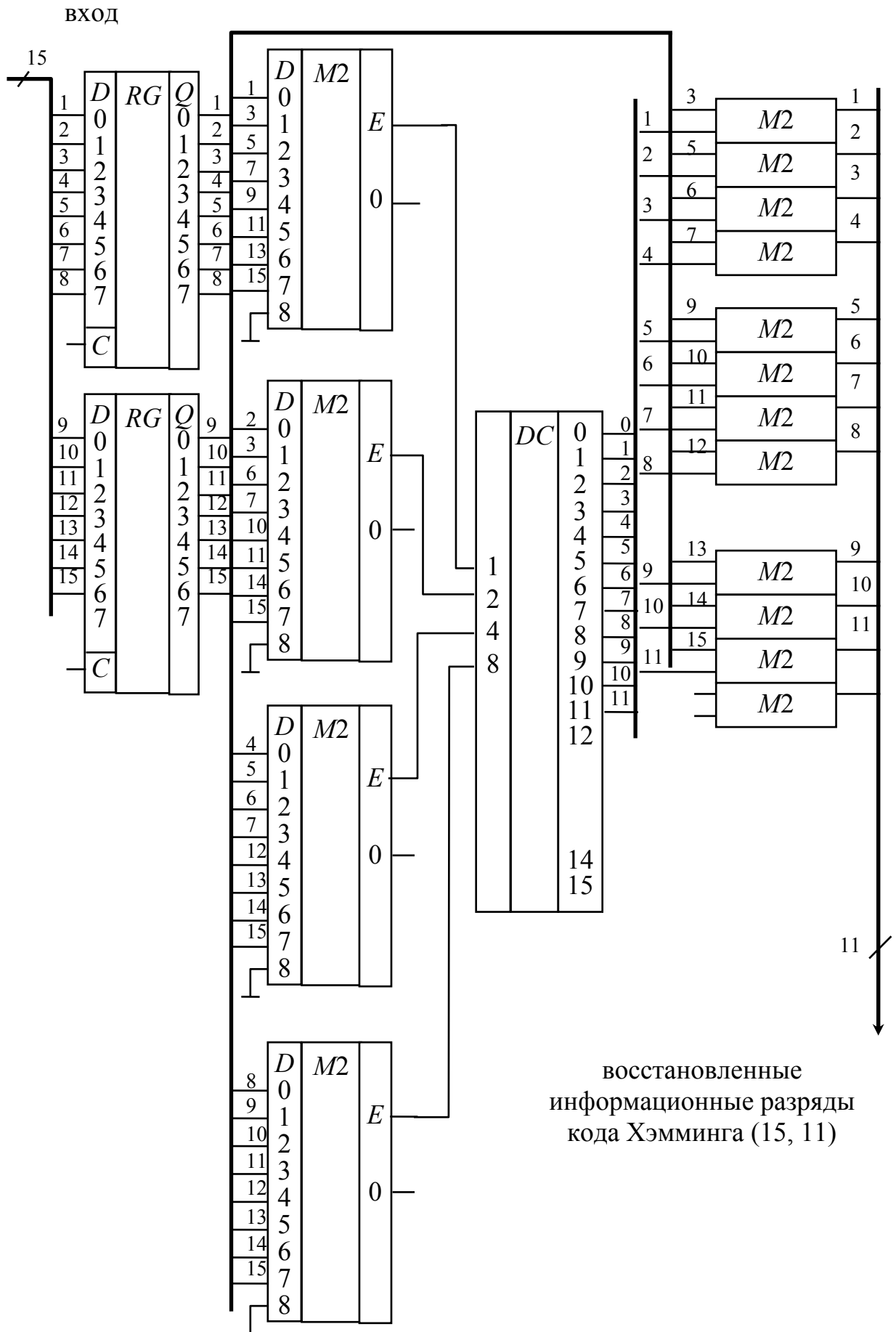


Рис. 6.33 Устройство декодирования кода Хэмминга (15, 11)

Если требуемое число информационных разрядов k' меньше того максимального значения, которое может обеспечить некоторое r , т.е.:

$$n' = (k' + r) < 2^r - 1,$$

то число входов устройств контроля по модулю 2, естественно, уменьшается. При этом неиспользуемыми не обязательно должны быть именно старшие разряды наибольшего, допускаемого числом r , кода Хэмминга. Могут быть использованы любые k' его разрядов. При этом изменяется закон дешифрации синдрома ошибки, но, может улучшиться способность кода обнаруживать какую-либо группу кратных ошибок.

Избыточность рассмотренного варианта кода Хэмминга достаточна лишь для исправления одиночной ошибки. Если ошибок, например, две, то устройство декодирования проведет коррекцию по указанным формальным правилам, но результат этой коррекции будет неверный: одна ошибка превратится в другую. Существует модифицированный код Хэмминга способный исправлять одиночную и обнаруживать двойную ошибки. Код строится на основе уже рассмотренного варианта кода Хэмминга путем добавления к нему еще одного, $(n+1)$ -го разряда контроля четности всего n -разрядного кода. Декодирование модифицированного кода Хэмминга осуществляется по следующим правилам.

- Число единиц в коде четное: синдром ошибки равен нулю – поврежденных разрядов нет; синдром ошибки не равен нулю – имеет место двойная ошибка (или групповая). Исправить ее невозможно.
- Число единиц в коде нечетное: синдром ошибки не равен нулю – имеет место одиночная ошибка, которая подлежит исправлению; синдром ошибки равен нулю – присутствует групповая ошибка. Исправить ее невозможно.

Отметим, что хотя модифицированный код Хэмминга и обнаруживает некоторые кратные ошибки, но среди ошибок, кратность которых больше двух, найдутся такие, которые этот код не обнаруживает т.е. просто пропускает их, или выполняет неверную коррекцию. Поэтому коды Хэмминга применяются там, где требования к достоверности данных достаточно высоки, а с учетом свойств канала передачи наиболее вероятны именно одиночные и уже совсем редко – двойные ошибки. Типичным примером применением таких кодов может быть полупроводниковая оперативная память, построенная из одноразрядных микросхем, вероятность сбоев при записи и считывании данных в которых довольно высока. Если же используются 4-разрядные микросхемы памяти, где вероятен 4-разрядный пакет ошибок, то применение ко-

да Хэмминга становится неэффективным. Для уверенного обнаружения пакетов ошибок, а тем более для их исправления, требуются коды с еще большей корректирующей способностью, а следовательно, и с большей избыточностью, такие как код БЧХ (Боуза – Чоудхури - Хоквингема), код Рида – Соломона и др.[13].

6.10.3. Устройство кодирования для кода Рида-Соломона

К числу кодов с высокой корректирующей способностью относятся коды Рида-Соломона, которые применяются в системах спутниковой связи, радиосвязи при наличии замираний, цифровом телевидении и радиовещании. Такие коды используются в европейском стандарте цифрового телевидения (DVB-T). Например, код Рида-Соломона (204, 188), применяемый в стандарте DVB-T позволяет при декодировании исправлять до восьми ошибок в пределах каждого кодового слова (рис. 6.34). Каждый элемент такого кодового слова имеет длину одного байта и, таким образом, код Рида-Соломона содержит $A_1, A_2 \dots A_{188}$ информационных элементов и $B_1, B_2 \dots B_{16}$ - проверочных элементов.

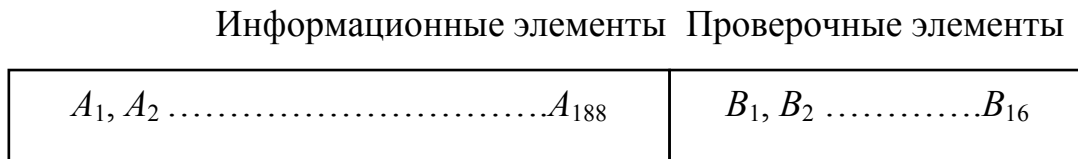


Рис. 6.34 Код Рида-Соломона (204, 188)

Структурная схема устройства кодирования для кода Рида-Соломона (204, 188), изображена на рис. 6.35.

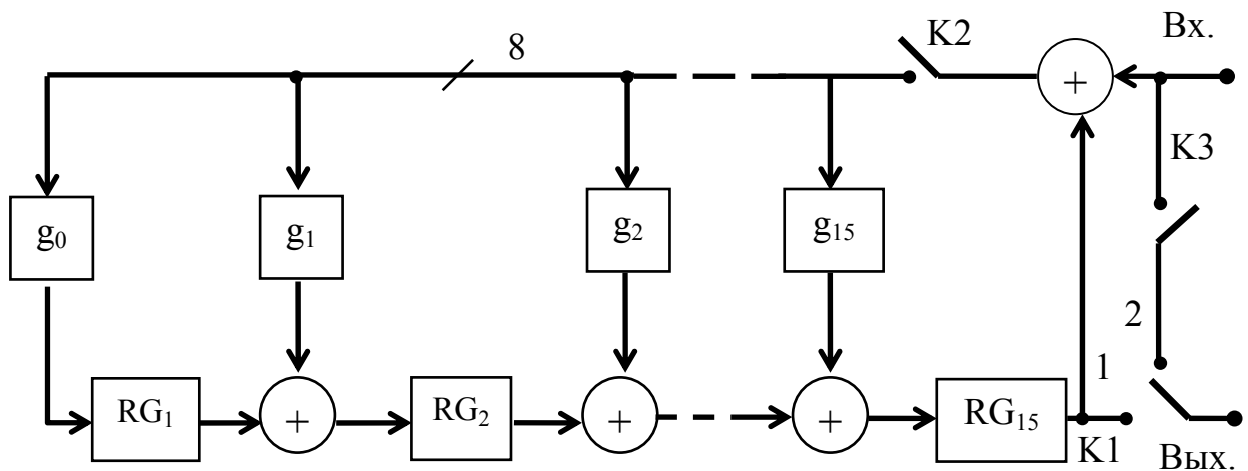


Рис. 6.35. Структурная схема устройства кодирования для кода Рида – Соломона

На вход устройства кодирования информационные байты поступают, начиная со старших разрядов $A_1, A_2 \dots, A_{188}$. В самом устройстве происходит формирование проверочных байт $B_1, B_2 \dots B_{16}$. Устройство кодирования реализуется с помощью регистра сдвига с 16-ю байтовыми ячейками памяти, сумматоров по модулю 2 и цифровых перемножителей, а также трех ключевых элементов К1, К2 и К3. Коэффициенты умножения для кода Рида-Соломона (204, 188) равны:

$$g_0 = 59, \quad g_1 = 36, \quad g_2 = 50, \quad g_3 = 98, \quad g_4 = 229, \quad g_5 = 41, \quad g_6 = 65, \quad g_7 = 163, \quad g_8 = 8, \quad g_9 = 30, \quad g_{10} = 209, \quad g_{11} = 68, \quad g_{12} = 189, \quad g_{13} = 104, \quad g_{14} = 13, \quad g_{15} = 59.$$

Процедура кодирования состоит из следующих этапов:

- Регистры сдвига обнуляются.
- Ключевой элемент К1 находится в положении 2, ключевые элементы К2 и К3 замкнуты. Информационные байты $A_1, A_2 \dots A_{188}$ непосредственно поступают на выход кодирующего устройства и одновременно участвуют в формировании проверочных байт.

- Ключевой элемент К1 переходит в положение 1, соединяя регистр сдвига с выходом. Ключевой элемент К2 в цепи обратной связи размыкается. Регистр сдвига в течение 16 тактов передает на выход проверочные байты $B_1, B_2 \dots B_{16}$. Таким образом, на выходе кодирующего устройства формируется код Рида – Соломона (рис.6.34).

На рис.6.36 изображена принципиальная схема кодирующего устройства, формирующего код Рида – Соломона, где RG_i - байтовые регистры сдвига, SM_i – сумматоры по модулю два, ROM_i – постоянные запоминающие устройства (ПЗУ) с объемом памяти $256 \cdot 8$, реализующие цифровые перемножители, MS-мультиплексор, логические элементы И и ключевые элементы К1, К2 и К3.

После обнуления регистров сдвига на втором этапе процедуры кодирования информационные байты подаются на Вх.[7..0] и одновременно на Вых.[7..0] и сумматор по модулю 2 (SM). Сумматор складывает информационные байты с байтами выходного регистра сдвига. Результат сложения поступает на 16 ветвей, содержащие блоки ПЗУ (цифровые перемножители) и сумматоры. Далее получившиеся байты записываются в регистры сдвига с тактовой частотой $f_{\text{синх}}$. С приходом управляющего сигнала K_1 ключевой элемент К1, состоящий из восьми ячеек логических элементов И цепи обратной связи (ПЗУ → сумматор → регистр → ПЗУ), запрещает прохождение информационных байт в кодирующее устройство. В течение 16 тактов на выход устройства кодирования поступают проверочные байты. Перед приходом следующего информационного пакета с помощью сигнала "Сброс" обнуляются все регистры сдвига.

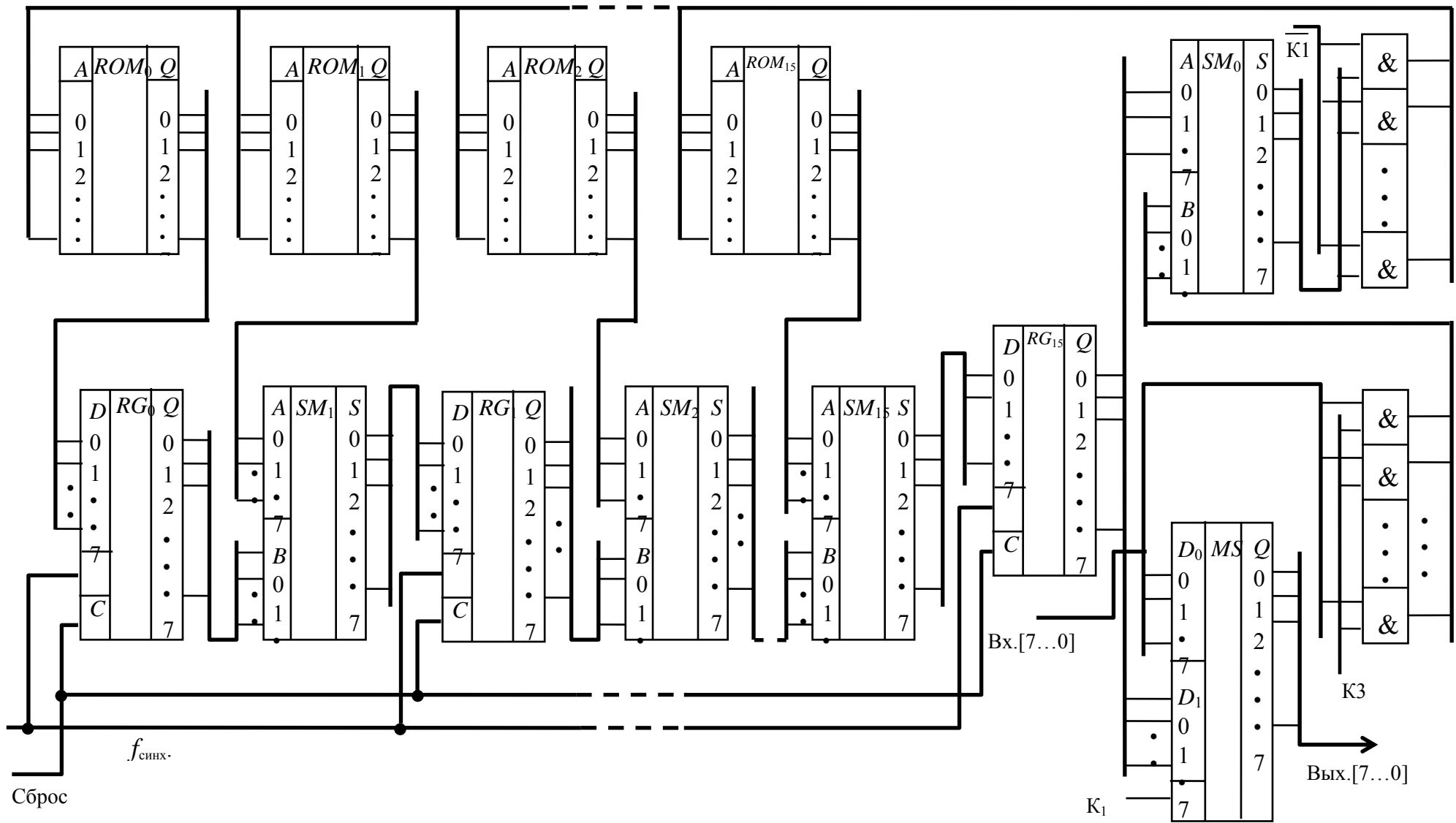


Рис. 6.36. Устройство кодирования для кода Рида-Соломона (204, 188)

Литература

1. Алексенко А.Г., Шагурин И.И. Микросхемотехника: учебное пособие для вузов. – М.: Радио и связь, 1990. – 496 с.
2. Антонов А.П., Мелехин В.Ф., Филиппов А.С. Обзор элементной базы фирмы ALTERA. – СПб.: ЭФО, 1997. – 142 с.
3. Бунтов В.Д., Емельянов А.Ф., Макаров С.Б. Проектирование цифровых радиотехнических устройств на интегральных схемах: учебное пособие – Л.: ЛПИ, 1985. 72 с.
4. Бунтов В.Д., Макаров С.Б. Цифровые радиотехнические устройства на интегральных схемах: учебное пособие. – Л.: ЛГТУ, 1991. – 81 с.
5. Бунтов В.Д., Макаров С.Б. Цифровые последовательностные устройства на интегральных схемах: учебное пособие. – СПб.: СПбГТУ, 1992. – 93 с.
6. Бойко В.И. и др. Схемотехника электронных систем. Цифровые устройства / В.И. Бойко, А.Н. Гуржий, В.Я. Жуйков, А.А. Зори, В.М. Спивак, В.В. Багрий. – СПб.: БХВ-Петербург, 2004. – 512 с.
7. Микросхемы памяти, ЦАП и АЦП: справочник / О.Н.Лебедев, А.Й.К. Марцинкявичус и др. – М.: КУБК-а, 1996. – 384 с.
8. Нарышкин А.К. Цифровые устройства и микропроцессоры: Учеб. пособие для студ. высш. учеб. заведений. – М.: Издательский центр «Академия», 2006. – 320 с.
9. Применение интегральных микросхем памяти: справочник /А.А. Дерюгин, В.В. Цыркин, Е.В.Красовский и др.; под редакцией А.Ю. Гордонова, А.А. Дерюгина. – М.: Радио и связь, 1994. – 232 с.
10. Пухальский Г.И., Новосельцева Т.Я. Цифровые устройства: учебное пособие для втузов. – СПб.: Политехника, 1996. – 885 с.
11. Соловьев А.А. Автогенераторы гармонических колебаний и синтезаторы частоты (основы теории и расчета): учебное пособие. – СПб.: СПбГТУ «ЛЭТИ», 2000. – 93 с.
12. Стешенко В.Б. ПЛИС фирмы ALTERA: проектирование устройств обработки сигналов. – М.: ДОДЕКА, 2000. – 128 с.
13. Угрюмов Е.П. Цифровая схемотехника: учебное пособие. – СПб.: БХВ – Петербург, 2005. – 528с.
14. Цифровые и аналоговые интегральные микросхемы: Справочник /С.В. Якубовский, Л.И. Ниссельсон, В.И. Кулешова и др.; Под редакцией С.В. Якубовского. – М.: Радио и связь, 1989. – 496 с.

15. Цифровые интегральные схемы: справочник / П.П. Мальцев, Н.С. Долидзе, М.И. Критенко и др. – М.: Радио и связь, 1994. – 240 с.
16. ALTERA 1998 Data Book, January 1998.
17. The Programmable Logic Data Book. Xilinx. 1998.
18. Сикарев А.А., Лебедев О.Н. Микроэлектронные устройства формирования и обработки сложных сигналов. – М.: Радио и связь, 1983. – 216 с.
19. Цифровые радионавигационные устройства. /В.В. Барашенков, А.Е. Лутченко, Е.М.Скорыходов и др.; Под редакцией В.Б. Смолова. – М.: Сов. радио, 1980, 288 с.
20. С.Б.Макаров, И.А. Цикин. Передача дискретных сообщений по радиолиниям с ограниченной полосой пропускания. – М.: Радио и связь, 1988. – 304 с.
21. И.С. Потемкин Функциональные узлы цифровой автоматики. – М.: Энергоатомиздат, 1988. –320 с.
22. Проектирование импульсных и цифровых устройств радиотехнических систем: Учебное пособие для радиотехнических спец. вузов./ Ю.П. Гришин, Ю.М. Казаринов, В.М. Катиков и др.; Под ред. Ю.М. Казаринова. – М.: Высш. школа, 1985. – 319 с.