

5. Комбинационные схемы

5.1. Принципы построения комбинационных логических схем

Комбинационные схемы (КС) — это схемы, у которых выходные сигналы $Y=(y_1, y_2, \dots, y_m)$ в любой момент дискретного времени однозначно определяются совокупностью входных сигналов $X=(x_1, x_2, \dots, x_n)$, поступающих в тот же момент времени t . При изменении входных сигналов практически мгновенно изменяется код на выходе устройства. Реализуемый в КС способ обработки информации называется комбинационным потому, что результат обработки зависит только от комбинации входных сигналов и формируется сразу при поступлении входных сигналов. Поэтому одним из достоинств комбинационных схем является их высокое быстродействие. Преобразование информации однозначно описывается логическими функциями вида $Y=f(X)$.

Каждый из m выходных сигналов y_i комбинационной схемы описывается логической (булевой) функцией:

$$y_i = f_i(x_1, x_2, \dots, x_n), \quad (3.1)$$

где x_i — логический входной сигнал, $i = 1, 2, \dots, m$.

Запись логических функций осуществляется в различных формах. Наиболее часто используется **совершенная дизъюнктивная нормальная форма (СДНФ)**, при которой логическая функция записывается в виде дизъюнкции (логической суммы) слагаемых, каждое из которых есть конъюнкция всех аргументов (прямых и инверсных). Причем учитываются только значения логической функции, равной единице, а в произведениях записывают инверсии тех аргументов, значения которых в этом случае равны нулю.

Например, логическая функция элемента И-НЕ запишется в виде:

$$y = \bar{x}_1 x_2 + x_1 \bar{x}_2 + \bar{x}_1 \bar{x}_2. \quad (3.2)$$

Для преобразования логических функций используются законы алгебры логики:

- коммутативности: $x_1 + x_2 = x_2 + x_1$; $x_1 x_2 = x_2 x_1$;
- ассоциативности: $x_1 + (x_2 + x_3) = (x_1 + x_2) + x_3$;
- дистрибутивности: $(x_1 + x_2)x_3 = x_1 x_3 + x_2 x_3$;
- тавтологии: $xx = x$; $x + x = x$;
- поглощения: $x(x + y) = x$; $x + (xy) = x$;
- дополняемости: $x\bar{x} = 0$; $x + \bar{x} = 1$;
- Моргана: $\overline{xy} = \bar{x} + \bar{y}$, $\overline{x + y} = \bar{x} \bar{y}$; ;
- операций с константами: $x \cdot 0 = 0$; $x \cdot 1 = x$; $x + 0 = x$; $x + 1 = 1$.

Используя законы алгебры логики, логическую функцию (3.2) для элемента И-НЕ можно преобразовать путем применения двойной инверсии к виду:

$$y = \overline{\overline{x_1 x_2}}. \quad (3.3)$$

Для построения любой КС необходима таблица истинности ее функционирования (составляется или задается). Затем составляется функция зависимости каж-

дого выхода схемы от входа (в форме СДНФ, которую затем можно перевести в упрощенную форму) и производится построение схемы на определенных логических элементах (чаще всего на И-НЕ и ИЛИ-НЕ).

Для получения СДНФ достаточно рассмотреть только те строки таблицы истинности, которые содержат логические "1" в выходном сигнале. Строки, содержащие в выходном сигнале логический 0, в построении цифровой схемы не участвуют. Каждая строка, содержащая в выходном сигнале логическую "1", реализуется схемой логического элемента "И" с количеством входов, совпадающим с количеством входных сигналов в таблице истинности.

Входные сигналы, описанные в таблице истинности логической единицей, подаются на вход этого логического элемента непосредственно, а входные сигналы, описанные в таблице истинности логическим нулем, подаются на вход этого же логического элемента "И" через инверторы. Объединение сигналов с выходов логических элементов "И", реализующих отдельные строки таблицы истинности, производится при помощи логического элемента "ИЛИ". Количество входов в логическом элементе "ИЛИ" определяется количеством строк в таблице истинности, в которых в выходном сигнале присутствует логическая единица.

Рассмотрим конкретный пример. Пусть необходимо реализовать таблицу истинности табл. 5.1.

Таблица 5.1- Таблица истинности произвольной логической функции

№	Входы				Выходы	
	X0	X1	X2	X3	Y0	Y1
0	0	0	0	0	0	0
1	0	0	0	1	1	0
2	0	0	1	0	0	0
3	0	0	1	1	0	0
4	0	1	0	0	0	0
5	0	1	0	1	0	1
6	0	1	1	0	1	0
7	0	1	1	1	0	0
8	1	0	0	0	0	0
9	1	0	0	1	0	1
10	1	0	1	0	0	0
11	1	0	1	1	0	0
12	1	1	0	0	1	0
13	1	1	0	1	0	0
14	1	1	1	0	0	0
15	1	1	1	1	0	1

Для синтеза цифровой схемы, реализующей сигнал Y0, достаточно рассмотреть строки 1,6 и 12. В таблице истинности три строки, содержащие единицу в выходном сигнале Y0, поэтому в формуле СДНФ будет содержаться три произведения комбинации входных сигналов (**термов**).

$$Y_0 = \overline{X_0} \overline{X_1} \overline{X_2} X_3 + \overline{X_0} X_1 X_2 \overline{X_3} + X_0 X_1 \overline{X_2} \overline{X_3}.$$

Полученное логическое выражение реализуется микросхемой D2 в схеме, приведенной на рисунке 5.1. Как и в формуле, каждая строка (терм) реализуется своим логическим элементом "И", затем выходы этих логических элементов объединяются при помощи логического элемента "ИЛИ". Количество входов логического элемента "И" (дизъюнкция) в СДНФ однозначно определяется количеством входных сигналов в таблице истинности. Количество этих элементов, а значит и количество входов в логическом элементе "ИЛИ" определяется количеством строк с единичным сигналом на реализуемом выходе цифровой схемы.

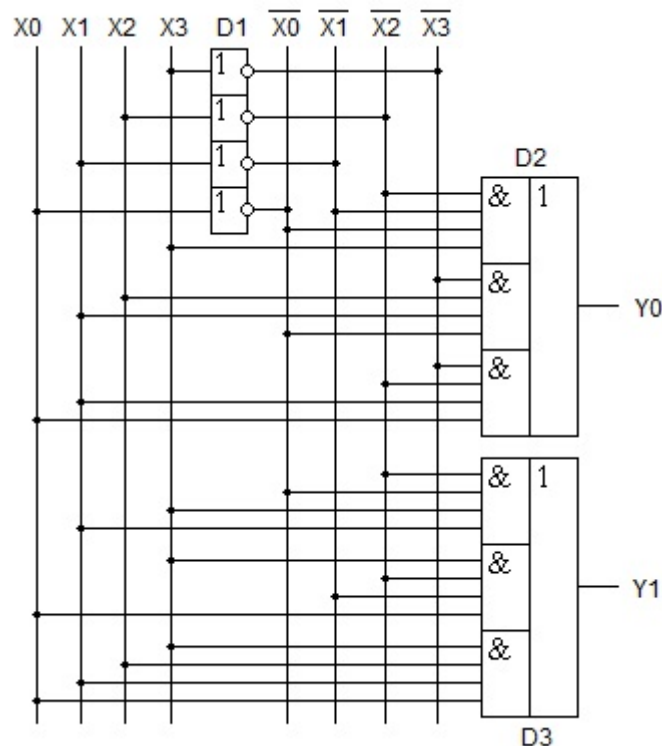


Рисунок 5.1 – Схема реализации логической функции

Для построения схемы, реализующей сигнал Y_1 , достаточно рассмотреть строки 5, 9 и 15. Эти строки реализуются микросхемой D3. Принцип построения этой схемы не отличается от рассмотренного выше. В таблице истинности присутствуют всего три строки, содержащие единицу в выходном сигнале Y_1 , поэтому в формуле СДНФ выхода Y_1 будет содержаться три произведения входных сигналов:

$$Y_1 = \overline{X_0} \overline{X_1} \overline{X_2} X_3 + \overline{X_0} \overline{X_1} X_2 \overline{X_3} + X_0 X_1 X_2 X_3.$$

Обычно при построении цифровых схем после реализации таблицы истинности производится минимизация схемы с целью ее упрощения.

5.2. Шифраторы

Шифратор (*кодер*) преобразует сигнал на одном из входов в n -разрядное двоичное число. Если шифратор имеет n выходов, число его входов должно быть не более чем 2^n . Шифратор, имеющий 2^n входов и n выходов, называется *полным*. Если число входов шифратора меньше 2^n , он называется *неполным*. Таблица истинности неполного шифратора на 9 входных линий приведена в таблице 5.2.

Таблица 5.2 - Таблица истинности неполного шифратора

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	y_3	y_2	y_1	y_0
1	0	0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	0	0	0	0	1	1
0	0	0	1	0	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	1	0	1
0	0	0	0	0	1	0	0	0	0	1	1	0
0	0	0	0	0	0	1	0	0	0	1	1	1
0	0	0	0	0	0	0	1	0	1	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	1

Функциональная схема шифратора, преобразующего десятичные цифры в 4-разрядное двоичное число, приведена на рисунке 5.2,а, а его условное обозначение – на рисунке 5.2,б. При появлении сигнала логической единицы на одном из десяти входов на четырех выходах шифратора будет присутствовать соответствующее двоичное число. Пусть сигнал логической единицы подан на вход X_7 . Тогда на выходах логических элементов DD1.1, DD1.2, DD1.3 будут сигналы логических единиц, а на выходе элемента DD1.4 – сигнал логического нуля. Таким образом, на выходах 8, 4, 2, 1 шифратора мы получим двоичное число 0111. Некоторые из шифраторов снабжаются входом стробирования. Наличие входа стробирования позволяет выделять сигнал в определенный момент времени. Шифраторы используются обычно в устройствах ввода цифровой информации (клавиатура) для преобразования символов алфавита или десятичных чисел в двоичные.

Одной из разновидностей шифраторов являются преобразователи кодов (ПК). Преобразователь кода – это комбинационное устройство, в котором каждой входной бинарной комбинации (входному слову) соответствует другая

комбинация (выходное слово), причем, разрядности входной и выходной комбинации могут различаться. Условное графическое обозначение преобразователя кода в схемах показано на рис.5.2в.

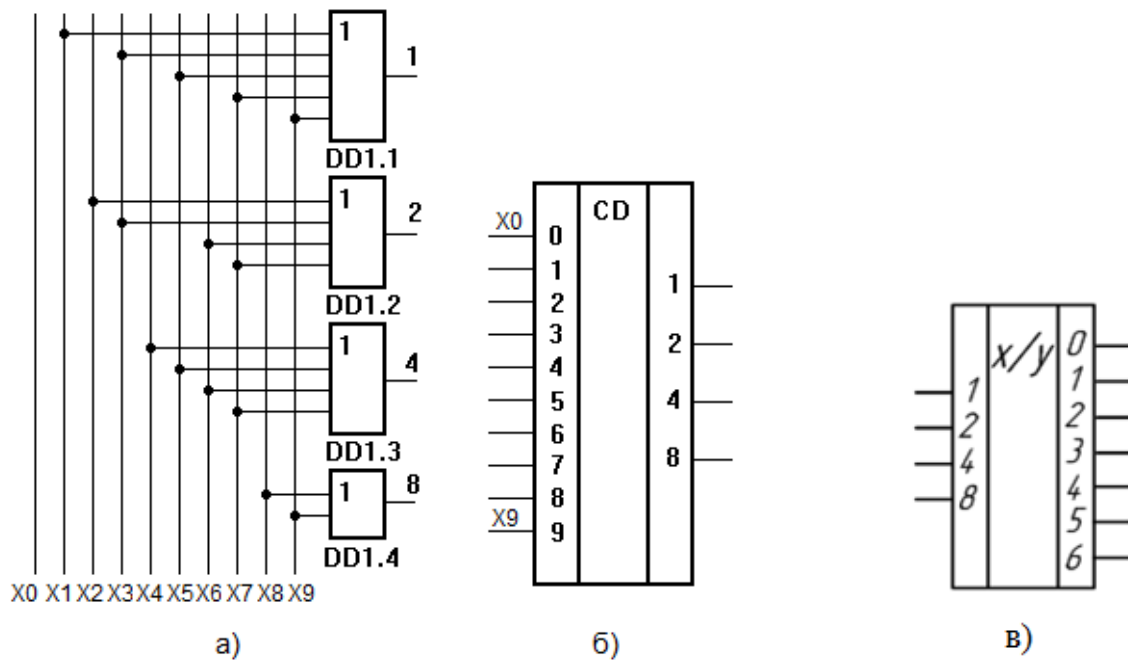


Рисунок 5.2 – Функциональная схема (а) и условное обозначение шифратора (б) и преобразователя кода (в)

На рис.5.3, в качестве примера, показано подключение преобразователя двоично-десятичного кода в семисегментный.

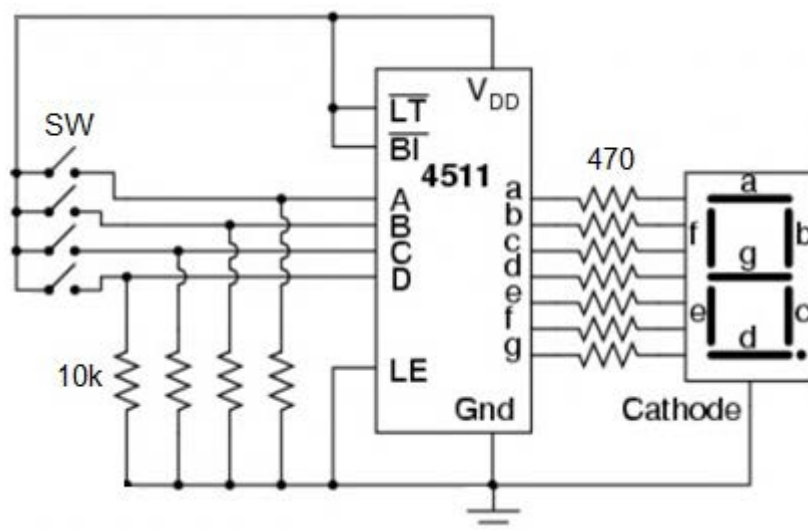


Рисунок 5.3 – Схема подключения ПК типа 4511 к индикатору

5.3. Дешифраторы

Дешифратор (декодер) – это комбинационная схема, у которой логическая единица на одном выходе при нулевых сигналах на остальных выходах соответствует определенному коду на входе (табл.5.3). Следовательно, дешифратор (ДШ) преобразует код, поступающий на его входы, в активный сигнал только на одном из его выходов. В дешифраторах с прямым выходом активным сигналом является высокий уровень напряжения (логическая 1), а в ДШ с инверсным выходом активным сигналом является низкий уровень напряжения, близкий к нулю (логический 0). Дешифратор n -разрядного двоичного числа имеет максимум 2^n выходов. Такой ДШ называется полным. Если количество выходов ДШ меньше максимального значения, то дешифратор называется неполным.

Таблица 5.3 – Таблица истинности двухразрядного дешифратора

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

На рисунке 5.4,а в качестве примера, изображена функциональная схема полного дешифратора на 16 инверсных выходов.

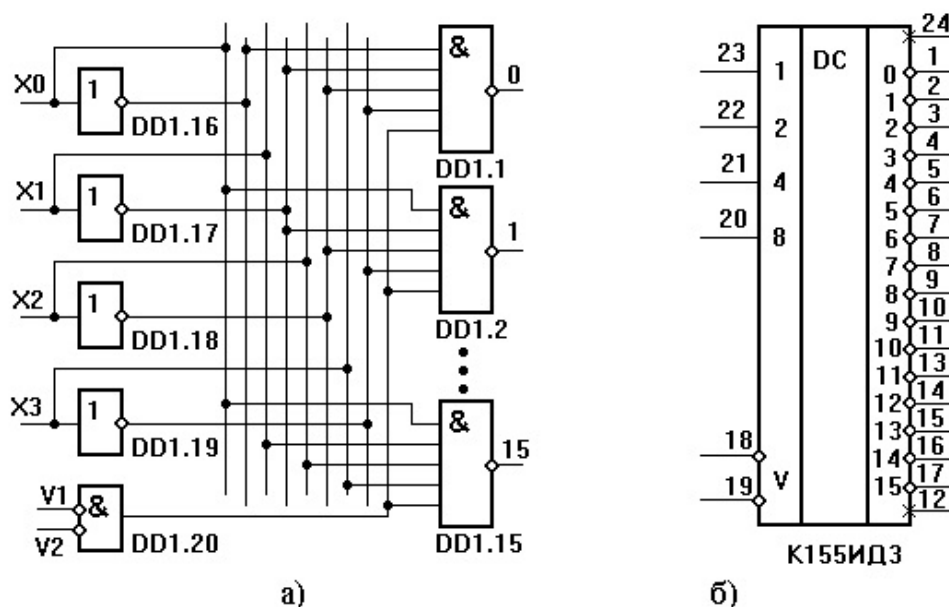


Рисунок 5.4 – Схема и условное обозначение дешифратора

По такой функциональной схеме построена, в частности, микросхема К155ИД3. Условное обозначение этой микросхемы на принципиальных схемах приведено на рисунке 5.4,б. Для дешифрации входных сигналов необходимо подать дополнительно разрешающие сигналы нулевого уровня на входы V1 и V2 микросхемы.

Пусть на входе дешифратора присутствует двоичное число 1111. В этом случае на всех пяти входах элемента DD1.15 будут сигналы логических единиц, а на выходе этого элемента будет логический нуль. На выходах всех остальных 15 элементов будут сигналы логических единиц. Если хотя бы на одном из входов V логическая единица, то единицы будут на всех 16 выходах.

Дешифраторы широко используются в устройствах вывода цифровой информации, для индикации двоичного числа в десятичном виде или семисегментном коде, для выбора порта в вычислительных системах и т.п.

5.4. Мультиплексоры и демультиплексоры

В цифровой технике широко применяются мультиплексоры и демультиплексоры. **Мультиплексор** это устройство, обеспечивающее соединение одного из множества информационных входов с единственным выходом. Номер информационного входа, который соединяется с выходом, задается в двоичном коде на адресных входах. Если мультиплексор имеет n адресных входов, то в нем может быть 2^n информационных входов.

Функциональная схема мультиплексора, имеющего четыре входа D0-D3 и один выход Q, приведена на рисунке 5.5а, а его условное обозначение на принципиальных схемах – на рисунке 5.5б. Мультиплексоры могут снабжаться дополнительным входом разрешения передачи информации с входов на выход.

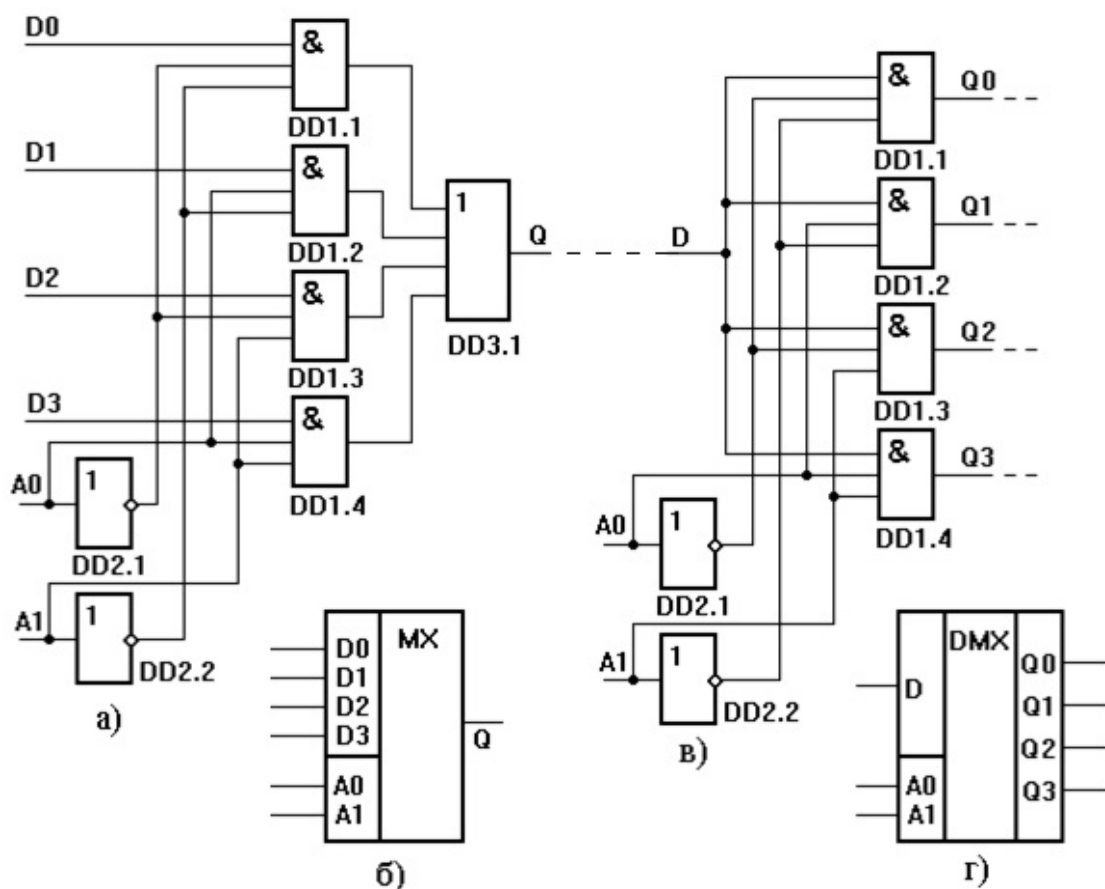


Рисунок 5.5 – Функциональные схемы мультиплексора (а), демультиплексора (в) и их обозначения (б) и (г)

Демультимплексор это устройство, обеспечивающее соединение одного из множества информационных выходов с единственным входом. Номер информационного выхода, который соединяется с входом, задается в двоичном коде на адресных входах. Если демультимплексор имеет n адресных входов, то в нем может быть 2^n информационных выходов.

Функциональная схема демультимплексора, имеющего четыре выхода Q0-Q3 и один вход D, показана на рисунке 5.5в, а его условное обозначение на принципиальных схемах – на рисунке 5.5г.

5.5. Двоичные сумматоры

Основным блоком любого процессора, осуществляющим вычислительные операции, является арифметико-логическое устройство (АЛУ). Четыре арифметические операции технически реализуются на основе двоичного сумматора, т.к. элементарные цифровые устройства могут выполнять только операцию суммирования двух чисел.

Не зависимо от разрядности используемых сумматоров, все они строятся на основе одноразрядного двоичного сумматора. Существуют два вида двоичных сумматоров: неполный и полный. Условное обозначение этих сумматоров показано на рис.5.6.



Рисунок 5.6 –Условные графические обозначения неполного (а) и полного (б) сумматоров

Неполный сумматор формирует сумму двух операндов А и В и сигнал переноса РО в старший разряд. Однако он не учитывает перенос из предыдущего разряда, поэтому он и называется полусумматором. В полном сумматоре в операции суммирования операндов текущего разряда учитывает и значение переноса PI из предыдущего разряда.

Таблицы истинности двоичных неполного (табл. 5.4) и полного (табл.5.5) одноразрядных сумматоров можно получить на основании правил суммирования двоичных чисел.

Таблица 5.4

A	B	S	PO
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Таблица 5.5

PI	A	B	S	PO
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Для описания любого сумматора достаточно иметь два логических уравнения. Для полусумматора на основе таблицы истинности получим систему уравнений следующего вида:

$$S = \bar{A}B + A\bar{B} = A \oplus B ; \quad PO = AB . \quad (5.1)$$

Для полного сумматора система логических уравнений принимает вид:

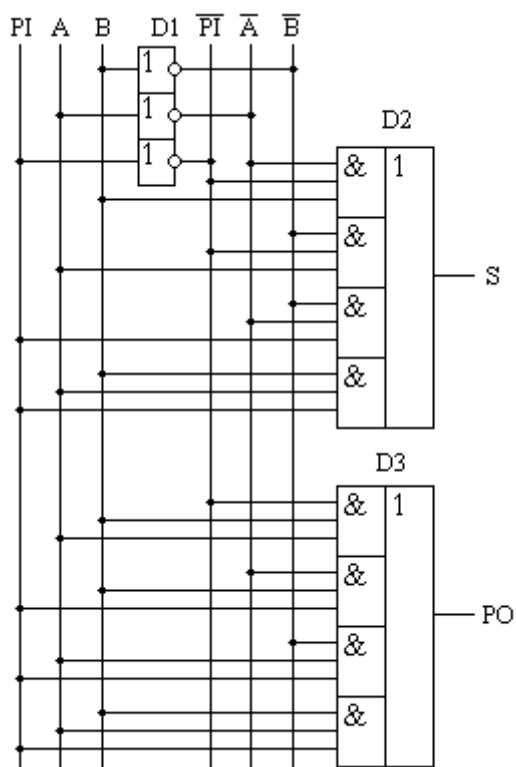
$$S = \overline{PI}(\bar{A}B + A\bar{B}) + PI(\bar{A}\bar{B} + AB) = PI \oplus (A \oplus B) . \quad (5.2)$$

$$PO = AB + PI(\bar{A}B + A\bar{B}) = AB + PI(A \oplus B) . \quad (5.3)$$

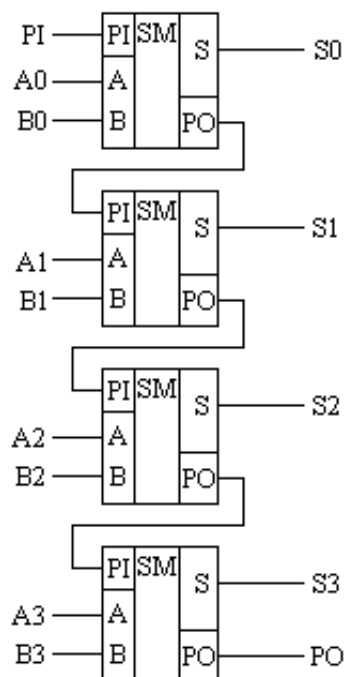
Как видно из уравнений, основной логической операцией в двоичных сумматорах является суммирование по модулю 2, которую реализует логический элемент исключающего ИЛИ типа 2И-ИЛИ-НЕ.

Схема полного сумматора, реализованная в соответствии с уравнениями (5.2) и (5.3), показана на рис.5.7,а.

Для построения многоразрядного сумматора, необходимо соединить входы и выходы переносов соответствующих двоичных разрядов. Схема соединения одноразрядных сумматоров для реализации четырехразрядного сумматора приведена на рисунке 5.7,б.



а)



б)

Рисунок 5.7 – Схема полного сумматора на логических элементах (а) и схема многоразрядного сумматора (б)

Одноразрядные сумматоры практически никогда не использовались, так как почти сразу же были выпущены микросхемы многоразрядных сумматоров. Примером четырехразрядных сумматоров является микросхема K561ИМ1 (зарубежный аналог 4008).