

Министерство образования Республики Беларусь  
Учреждение образования  
«Белорусский государственный университет  
информатики и электроники»

Кафедра защиты информации

Л. М. Лыньков, В. П. Ширинский

***РАЗРАБОТКА ЛОГИЧЕСКОЙ СХЕМЫ  
ЦИФРОВОГО УСТРОЙСТВА***

Методическое пособие  
к курсовому проектированию  
по дисциплине «Цифровые и микропроцессорные устройства  
средств измерений»  
для студентов специальности  
I-54 01 01-02 «Метрология, стандартизация и сертификация»  
дневной формы обучения

Минск 2007

УДК 004.31 (076)  
ББК 32.973.26-018.2 я 73  
Л 88

**Рецензент**

доцент кафедры электроники, канд. техн. наук В. А. Мельников

**Лыньков, Л. М.**

Л 88      Разработка логической схемы цифрового устройства : метод. пособие к курсовому проектированию по дисц. «Цифровые и микропроцессорные устройства средств измерений» для студ. спец. I-54 01 01-02 «Метрология, стандартизация и сертификация» днев. формы обуч. / Л. М. Лыньков, В. П. Ширинский. – Минск : БГУИР, 2007. – 48 с. : ил.  
ISBN 978-985-488-198-0

Методическое пособие предназначено для выполнения практических занятий и курсового проектирования по дисциплине «Цифровые и микропроцессорные устройства средств измерений» для специальности «Метрология, стандартизация и сертификация». Пособие содержит методические указания по выполнению курсового проекта, тематику практических занятий и курсового проектирования комбинационных и последовательностных схем.

**УДК 004.31 (076)**  
**ББК 32.973.26-018.2 я 73**

**ISBN 978-985-488-198-0**

© Лыньков Л. М., Ширинский В. П., 2007  
© УО «Белорусский государственный университет информатики и радиоэлектроники», 2007

## Содержание

Введение .....	4
1. Этапы разработки логической схемы цифрового устройства (ЦУ) .....	5
2. Разработка и минимизация аналитической модели проектируемого цифрового устройства .....	5
2.1. Составление таблицы истинности цифрового устройства .....	5
2.2. Доопределение и минимизация аналитической модели .....	8
3. Построение и исследование различных схем триггеров .....	10
3.1. Асинхронный <i>RS</i> -триггер .....	10
3.2. Асинхронные триггеры с одним входом .....	13
3.3. Синхронные триггеры .....	16
3.4. Синтез триггерной структуры на универсальном <i>D</i> -триггере .....	21
4. Синтез счетчиков .....	23
4.1. Логические методы синтеза счетчиков .....	23
4.2. Синтез счетчика с фиксированным коэффициентом счета .....	23
5. Реализация логических схем в различных базисах .....	26
5.1. Логические схемы .....	26
5.2. Многоступенчатые реализации .....	28
5.3. Базисы И-НЕ и ИЛИ-НЕ .....	30
6. Обоснование выбора элементной базы .....	32
6.1. Серии логических элементов .....	32
6.2. Временные характеристики логических элементов .....	37
6.3. Оценка качества функциональных схем .....	38
7. Правила схемного включения элементов .....	42
7.1. Элемент с открытым коллектором .....	43
7.2. Расширение логических возможностей элементов .....	45
Литература .....	48

## Введение

Исходя из общих принципов проектирования микроэлектронной аппаратуры можно представить последовательность проектирования цифрового устройства:

1. Уточнение технического задания на цифровое устройство, входящее в систему.
2. Расчет цифрового устройства в целом, разработка его функциональной схемы.
3. Выбор степени интеграции системы.
4. Оценка возможности цифровой обработки сигнала.
5. Выбор элементной базы.
6. Разработка принципиальной схемы цифрового устройства.
7. Компьютерное моделирование отдельных узлов и цифрового устройства в целом.

*Уточнение технического задания на цифровое устройство, входящее в систему. Выбор структуры системы.* При системном подходе к проектированию, учете требований микроэлектроники еще на этапе проектирования системы удастся использовать методы обработки сигнала, реализуемые при микроэлектронном исполнении. Используя методы системотехники, можно выбрать более оптимальную структуру всей системы в целом.

*Разработка функциональной схемы.* На этом этапе выполняются основные расчеты с применением методов микросхемотехники. Функциональная схема цифрового устройства может разрабатываться в процессе проектирования. На цифровых интегральных схемах можно реализовать любой алгоритм обработки сигнала.

*Выбор степени интеграции системы* – сложная комплексная задача. Достоинством цифровых устройств обработки сигнала является возможность неограниченно долгого хранения запомненных сигналов, легкая возможность адаптации, высокая технологичность в производстве, отсутствие регулировок в процессе изготовления, большие перспективы дальнейшей микроминиатюризации.

*Выбор элементной базы.* Определяется требованиями к цифровому устройству, заданным уровнем интеграции, рабочим диапазоном частот.

*Разработка принципиальной схемы.* Предполагает расчет всех конкретных узлов и элементов цифрового устройства. Первым этапом разработки принципиальной схемы цифрового устройства является разработка его структурной схемы, где проектируемое устройство разбивается на узлы (структурные блоки более низкого уровня, чем проектируемое цифровое устройство) – шифраторы, мультиплексоры, сумматоры, счетчики, регистры и так далее. Синтез таких устройств проводится известными методами.

Для этого сначала составляется таблица истинности проектируемого узла и далее из соображений оптимизации доопределяется проектировщиком, если

такая возможность имеется. На основе такой доопределенной таблицы создается аналитическая модель узла, которая, как правило, нуждается в оптимизации. После минимизации вычисленные минимальные формы реализуются в различных базисах. Для полученных вариантов подбирается элементная база и результаты сравниваются по различным критериям: потребляемой мощности, размерам, быстродействию и т.д.

Заключительным этапом проектирования является компьютерное моделирование.

## **1. Этапы разработки логической схемы цифрового устройства (ЦУ)**

При наличии у разработчика небольшого опыта проектирования ЦУ можно рекомендовать следующую последовательность действий.

### **Этап 1. Составление таблицы истинности цифрового устройства**

Основная цель этапа – формализация задачи, в процессе которой нужно определить значения функции для каждой комбинации значений аргументов. Результат этапа – таблица истинности. Это задача, неоднозначное толкование которой невозможно. После этого можно строить аналитическую модель цифрового устройства.

### **Этап 2. Доопределение функции**

Если функция определена не при всех наборах аргументов, то нужно ликвидировать неоднозначность таблицы. При малом числе неопределенных значений лучше рассмотреть несколько вариантов. Если же число безразличных значений или самих аргументов большое, то, возможно, придется полностью определить функцию всеми нулями или всеми единицами – так, чтобы в результате уменьшить число членов аналитической модели.

### **Этап 3. Построение аналитической модели цифрового устройства**

По полностью определенной таблице составить аналитическое описание. Следует рассматривать несколько вариантов аналитического описания, чтобы в дальнейшем остановиться на лучшем варианте.

### **Этап 4. Минимизация аналитической модели цифрового устройства**

**Этап 5. Реализация полученных минимальных форм в различных базисах**

### **Этап 6. Оценка различных полученных вариантов**

### **Этап 7. Выбор наилучшего из полученных вариантов**

## **2. Разработка и минимизация аналитической модели проектируемого цифрового устройства**

### **2.1. Составление таблицы истинности цифрового устройства**

Рассмотрим реализацию первого этапа – составление таблицы истинности на примере преобразователя кода Грея в двоичный код.

Среди невзвешенных двоичных кодов специальное применение находят такие, у которых переход к соседнему числу сопровождается изменениями только в одном разряде (коды с обменной единицей). Так, в технике аналого-цифрового преобразования и пересчетных устройствах широко используется код Грея, называемый также циклическим или рефлексно-двоичным кодом. Он позволяет существенно сократить преобразования, упростить кодирующую логику, а также повысить эффективность защиты от нежелательных сбоев. Недостатком кода Грея является то, что в нем затруднено выполнение арифметических операций и цифроаналогового преобразования. Поэтому при необходимости код Грея преобразуется в обычный двоичный код.

Переход от двоичного кода к коду Грея осуществляется по правилу: старшие разряды совпадают, а любой следующий разряд  $X_k$  кода Грея равен сумме по модулю 2 соответствующего  $X_k$  и предыдущего  $X_{k-1}$  разрядов двоичного кода, т.е.

$$X_k = X_k \oplus X_{k-1}$$

(сумма по модулю 2 равна арифметической сумме без учета переноса в старший разряд). При единичном переходе старшие разряды также совпадают, но каждый следующий разряд получается в результате суммирования по модулю 2 полученного предыдущего разряда двоичного кода и соответствующего разряда кода Грея, т.е.

$$X_k = X_{k-1} \oplus \hat{X}_{k0}$$

Таким образом, получаем таблицу соответствия десятичных цифр и кода Грея.

Таблица 2.1

Дес.цифры	Код Грея
0	0 0 0 0
1	0 0 0 1
2	0 0 1 1
3	0 0 1 0
4	0 1 1 0
5	0 1 1 1
6	0 1 0 1
7	0 1 0 0
8	1 1 0 0
9	1 1 0 1

Один из способов цифровой индикации ориентирован на использование семисегментного индикатора (рис. 2.1).

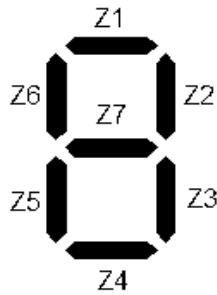


Рис. 2.1. Схема матрицы семисегментного индикатора

Для синтеза преобразователя, управляющего семисегментным индикатором, составим таблицу соответствия кодов на основании рис. 2.1. Десятичные цифры, которые необходимо высвечивать на индикаторе, задаются в 4-разрядном коде Грея. Через  $Y_1, Y_2, \dots, Y_7$  обозначаются управляющие потенциалы. Если элемент светится, то он находится в состоянии «1», если нет – «0». Однако управление цифровым индикатором осуществляется по другому принципу: уровень логической «1» на входе индикатора вызывает его погашение. Например, чтобы высветить цифру «6», необходимо погасить сегмент  $Z_2$ , т.е.  $Y_2$  должно быть равно «1». На основании указанных соображений построим таблицу соответствия (табл. 2.2).

Таблица 2.2

	Код Грея				Код семисегментного преобразователя						
	$X_4$	$X_3$	$X_2$	$X_1$	$\bar{Z}_1$	$\bar{Z}_2$	$\bar{Z}_3$	$\bar{Z}_4$	$\bar{Z}_5$	$\bar{Z}_6$	$\bar{Z}_7$
0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	1	1	0	0	1	1	1	1
2	0	0	1	1	0	0	1	0	0	1	0
3	0	0	1	0	0	0	0	0	1	1	0
4	0	1	1	0	1	0	0	1	1	0	0
5	0	1	1	1	0	1	0	0	1	0	0
6	0	1	0	1	0	1	0	0	0	0	0
7	0	1	0	0	0	0	0	1	1	1	1
8	1	1	0	0	0	0	0	0	0	0	0
9	1	1	0	1	0	0	0	0	1	0	0

На основании табл. 2.2 построим выражения для  $Y_1$ – $Y_7$ :

$$\begin{aligned}
Y_1 &= \overline{X_1}X_2X_3\overline{X_4} + X_1\overline{X_2}\overline{X_3}\overline{X_4}; \\
Y_2 &= X_1\overline{X_2}X_3\overline{X_4} + X_1X_2X_3\overline{X_4}; \\
Y_3 &= X_1X_2\overline{X_3}X_4; \\
Y_4 &= X_1\overline{X_2}X_3X_4 + \overline{X_1}X_2\overline{X_3}\overline{X_4} + \overline{X_1}X_2X_3\overline{X_4}; \\
Y_5 &= X_1\overline{X_2}X_3X_4 + \overline{X_1}X_2\overline{X_3}X_4 + \overline{X_1}X_2X_3\overline{X_4} + \\
&+ \overline{X_1}X_2X_3X_4 + X_1\overline{X_2}\overline{X_3}\overline{X_4} + X_1\overline{X_2}X_3X_4; \\
Y_6 &= X_1\overline{X_2}X_3X_4 + \overline{X_1}X_2\overline{X_3}X_4 + X_1\overline{X_2}X_3\overline{X_4} + \overline{X_1}X_2X_3\overline{X_4}; \\
Y_7 &= \overline{X_1}X_2\overline{X_3}X_4 + X_1\overline{X_2}X_3\overline{X_4} + \overline{X_1}X_2X_3\overline{X_4}.
\end{aligned}$$

## 2.2. Доопределение и минимизация аналитической модели

Для минимизации заполним карты Вейча–Карно для  $Y_1$ – $Y_7$ .

Таблица 2.3

$X_2X_1$ $X_4X_3$	00	0 1	1 1	1 0
0 0	0	1	0	0
0 1	0	0	0	1
1 1	0	0	*	*
1 0	*	*	*	*

Табл. 2.3 построена для функции  $Y_1$ . Звездочками показаны неопределенные состояния.

Эта функция имеет 2 единичных значения при значениях аргументов 0001 и 0110, при значениях аргументов 0000, 0011, 0010, 0100, 0101, 0111, 1100 и 1101 значения  $Y_1$  равны 0, в остальных случаях они не определены. Доопределяем их таким образом, чтобы получить самый минимальный результат. В результате получаем

$$Y_1 = X_1\overline{X_2}\overline{X_3} + \overline{X_1}X_2X_3.$$

Аналогично вычисляем  $Y_2$  (табл. 2.4).

Таблица 2.4

$X_2X_1$ $X_4X_3$	00	0 1	1 1	1 0
0 0	0	0	0	0



0 1	0	1	1	0
1 1	0	0	*	*
1 0	*	*	*	*

$$Y_2 = X_1 X_2 \overline{X_4}.$$

$Y_3$  представлен только одной импликантой, поэтому

$$Y_3 = X_1 X_2 \overline{X_3} \overline{X_4}.$$

Вычисленное на основе табл. 2.5 значение  $Y_4$

Таблица 2.5

$X_2 X_1$ $X_4 X_3$	0 0	0 1	1 1	1 0
0 0	0	1	0	0
0 1	1	0	0	1
1 1	0	0	*	*
1 0	*	*	*	*

$$Y_4 = X_1 \overline{X_2} \overline{X_3} + \overline{X_1} X_3 \overline{X_4}.$$

Вычисленное на основе табл. 2.6 значение  $Y_5$

Таблица 2.6

$X_2 X_1$ $X_4 X_3$	0 0	0 1	1 1	1 0
0 0	0	1	0	1
0 1	1	0	1	1
1 1	0	1	*	*
1 0	*	*	*	*

$$Y_5 = \overline{X_1} X_2 + X_2 X_3 + X_1 X_4 + \overline{X_1} X_3 \overline{X_4} + X_1 \overline{X_2} \overline{X_3}.$$

Вычисленное на основе табл. 2.7 значение  $Y_6$

Таблица 2.7

$X_2 X_1$ $X_4 X_3$	0 0	0 1	1 1	1 0
0 0	0	1	1	1

0 1	1	0	0	0
1 1	0	0	*	*
1 0	*	*	*	*

$$Y_6 = X_2 \overline{X_3} + X_1 \overline{X_2} \overline{X_3} + X_1 X_2 \overline{X_3} \overline{X_4}.$$

Вычисленное на основе табл. 2.8 значение  $Y_7$

Таблица 2.8

$X_2 X_1$ $X_4 X_3$	0 0	0 1	1 1	1 0
0 0	1	1	0	0
0 1	1	0	0	0
1 1	0	0	*	*
1 0	*	*	*	*

$$Y_7 = \overline{X_2} \overline{X_3} + \overline{X_1} X_2 X_4.$$

На основе полученных выражений можно построить схему семиразрядного преобразователя для цифровой индикации.

### 3. Построение и исследование различных схем триггеров

#### 3.1. Асинхронный RS-триггер

**RS-триггер** – двухвходовый триггер, который при подаче активного сигнала на  $S$ -вход и неактивного сигнала на  $R$ -вход устанавливается в единичное состояние, при подаче активного сигнала на  $R$ -вход и неактивного сигнала на  $S$ -вход устанавливается в нулевое состояние. Одновременная подача двух активных сигналов на  $S$ - и  $R$ -входы запрещена, если же такая ситуация возникнет, то состояние триггера считается неопределенным.

Описание функционирования RS-триггера можно представить в виде таблицы переходов (табл. 3.1). Если за активный сигнал на  $R$ - и  $S$ -входах принять уровень единицы, то получим табл. 3.1.

Таблица 3.1

Набор номера	$R$	$S$	$Q_t$	$Q_{t+1}$	$\overline{Q_{t+1}}$
0	0	0	0	(0)	1
1	0	0	1	(1)	0
2	0	1	0	(1)	0
3	0	1	1	(1)	0
4	1	0	0	(0)	1

5	1	0	1	0	1
6	1	1	0	x	x
7	1	1	1	x	x

Переменная  $R$ , стоящая в наборе  $RSQ_t$  по левую сторону, считается старшим разрядом двоичного кода. Из табл. 3.1 видно, что  $RS$ -триггер сохраняет одно из устойчивых состояний независимо от многократного изменения информационного сигнала на другом входе. Это свойство блокирования – основное функциональное свойство  $RS$ -триггера, и именно оно делает его элементарной запоминающей ячейкой.

В столбце  $Q_{t+1}$  записываются значения выходной переменной  $Q$ , которые она принимает в момент времени  $t + 1$ ... Если  $Q_{t+1} = Q_t$ , то такое состояние триггера устойчиво и в столбце  $Q_{t+1}$  записывается в скобках; если  $Q_{t+1} \neq Q_t$ , то состояние триггера неустойчиво и в столбце  $Q_{t+1}$  записывается без скобок. Для последней ситуации возможны два случая:

1. Триггер при том же наборе входных переменных переходит в устойчивое состояние (переход обозначен стрелкой).

2. Триггер при том же наборе входных переменных будет постоянно изменять свое состояние, т.е. будет находиться в автоколебательном режиме. Последнее говорит о том, что триггер данной структуры не может управляться потенциальными сигналами.

Из табл. 3.1 вытекает, что при всех наборах входных переменных  $RS$ -триггер имеет устойчивые состояния, причем предполагается, что набор переменных  $RS = 11$  при нормальной работе  $RS$ -триггера не возникает, поэтому значение  $Q_{t+1}$  при этом наборе не представляет интереса и обозначено знаком  $x$ . Отсутствие неустойчивых состояний у  $RS$ -триггера говорит о том, что его характеристическое уравнение целиком отражает структуру потенциально управляемого триггера.

Определим характеристическое уравнение  $RS$ -триггера, представив  $Q_{t+1} = f(x_t, Q_t)$  с помощью таблицы Вейча–Карно (табл. 3.2).

Таблица 3.2

$SQ_t$	0 0	0 1	1 1	1 0
$R$				
0	0	1	1	1
1	0	0	x	x

$Q_{t+1}$

Раскрыв карту Вейча–Карно, получим

$$Q_{t+1} = S + \bar{R}Q_t.$$

Наличие запрещенной комбинации информационных сигналов  $RS = 11$  запишем в виде следующего требования:

$$RS = 0.$$

Выбрав как элементную базу базис ИЛИ-НЕ, преобразуем

$$Q_{t+1} = S + \bar{R}Q_t,$$

используя закон отрицания и правило де Моргана, к виду

$$\bar{Q}_{t+1} = \overline{S + R + \bar{Q}_t}.$$

Из этого выражения видно, что  $RS$ -триггер представляет собой последовательное соединение двух элементов ИЛИ-НЕ, замкнутое само на себя (рис. 3.1, а).

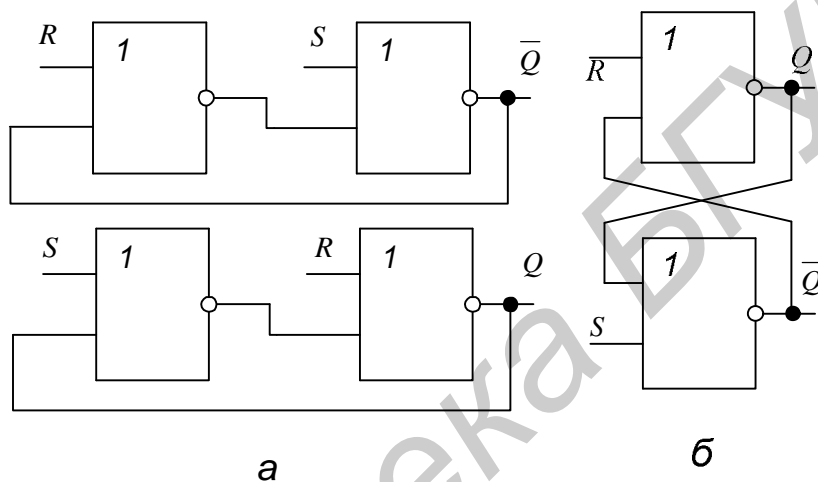


Рис. 3.1. Схема  $RS$ -триггера в базисе ИЛИ-НЕ (а, б)

Таблица 3.3

SQt	0 0	0 1	1 1	1 0
R	0	0	0	0
$\bar{Q}_{t+1}$	1	1	x	x

Из табл. 3.1 и 3.3 можно получить выражение для  $\bar{Q}_{t+1}$ :

$$\bar{Q}_{t+1} = R + \bar{S}\bar{Q}_t.$$

Выбрав как элементную базу базис И-НЕ, преобразуем

$$Q_{t+1} = S + \bar{R}Q_t \quad \text{и} \quad RS = 0,$$

используя правило де Моргана к виду

$$Q_{t+1} = \overline{\overline{S} \cdot \overline{R} \cdot Q_t} \quad \overline{R} + \overline{S} = 1.$$

Из этих выражений видно, что  $RS$ -триггер (точнее,  $\overline{RS}$ -триггер) представляет собой последовательное соединение двух элементов И-НЕ, замкнутое само на себя (рис. 3.2, а).

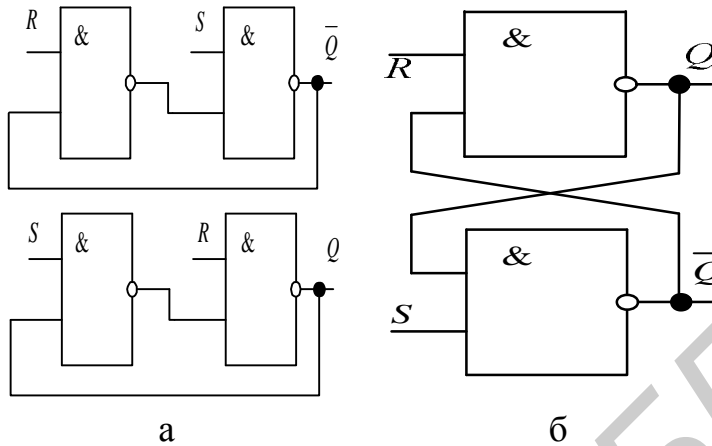


Рис. 3.2. Схема  $RS$ -триггера в базисе И-НЕ (а, б)

С помощью характеристических уравнений

$$Q_{t+1} = S + \overline{R}Q_t;$$

$$Q_{t+1} = \overline{R + \overline{S} + Q_t};$$

$$Q_{t+1} = \overline{\overline{S} \cdot \overline{R} \cdot Q_t}, \quad \overline{R} + \overline{S} = 1$$

можно определить состояние триггера  $Q_{t+1}$ , в которое он перейдет в момент времени  $t + 1$ , если известны комбинации входных сигналов и состояние триггера  $Q_t$  в предшествующий момент времени  $t$ .

### 3.2. Асинхронные триггеры с одним входом

Асинхронный  $D$ -триггер функционирует соответственно табл. 3.4 (базис ИЛИ-НЕ).

Таблица 3.4

Набор	$D$	$Q_t$	$Q_{t+1}$	$R^*$	$S^*$
0	0	0	(0)	×	0
1	0	1	0	1	0
2	1	0	1	0	1
3	1	1	(1)	0	×

В табл. 3.4 представлены карты Вейча–Карно для  $D$ -триггера (характеристическая таблица  $D$ -триггера).

Известно, что

$$Q_{t+1} = D;$$

$$R^* = \bar{D};$$

$$S^* = D.$$

Таблица 3.5

$Q$	0	1
$D$		
1	1	1
0	0	1

$Q_{t+1}$

$Q$	0	1
$D$		
1	0	0
0	1	1

$R^*$

$Q$	00	01
$D$		
1	1	1
0	0	0

$S^*$

Два последних выражения определяют структуру триггера, полученную на основе обобщенной схемы триггерного устройства. Однако

$$Q_{t+1} = D$$

говорит о том, что полученная схема (рис. 3.3) является тривиальной, вырожденной, т.к.  $D$ -триггер можно получить с одного повторителя или двух последовательно включенных инверторов, если необходимо парафазное представление выходных сигналов.

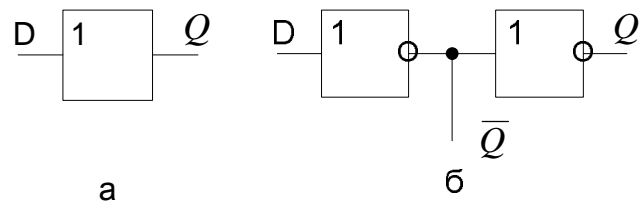


Рис. 3.3. Схема реализации  $D$ -триггера с однофазным (а) и парафазным (б) выходами

Схему, показанную на рис. 3.4, иногда называют  $RS$ -триггером с принудительной парафазной установкой.

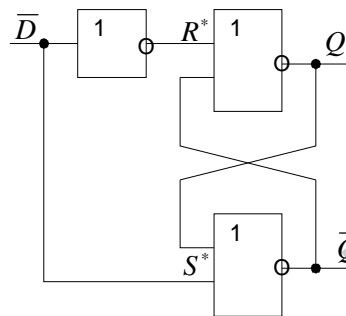


Рис. 3.4. Схема  $D$ -триггера в базисе ИЛИ-НЕ

В табл. 3.5 представлена характеристическая таблица асинхронного  $T$ -триггера.

Таблица 3.6

Набор	$T$	$Q_t$	$Q_{t+1}$	$R^*$	$S^*$
0	0	0	(0)	×	0
1	0	1	(1)	0	×
2	1	0	1	0	1
3	1	1	0	1	0

Из табл. 3.6 вытекает, что при  $T = 1$  триггер имеет неустойчивое состояние, т. е. находится в автоколебательном режиме. Это говорит о том, что асинхронный  $T$ -триггер с управлением потенциалом не может быть реализован на одной элементарной запоминающей ячейке. Анализ работы управляемого импульсом  $T$ -триггера представляет интерес, так как при  $T = 0$  триггер имеет устойчивые состояния.

В табл. 3.7 представлены карты Вейча–Карно для  $T$ -триггера с импульсным управлением.

Таблица 3.7

$Q_t$	1	0
$T$		
1	0	1
0	1	0

$Q_{t+1}$

$Q_t$	1	0
$T$		
1	1	0
0	0	x

$R^*$

$Q_t$	1	0
$T$		
1	0	1
0	x	0

$S^*$

Из табл. 3.7 вытекают следующие уравнения:

$$Q_{t+1} = T\bar{Q}_t + \bar{T}Q_t;$$

$$R^* = T \cdot Q_t = \overline{\bar{T} + \bar{Q}_t};$$

$$S^* = T \cdot \bar{Q}_t = \overline{\bar{T} + Q_t}.$$

При допущении, что схема выполняется в базисе ИЛИ-НЕ, а триггер управляется перепадами 10 на входе  $T$ , получим схему, показанную на рис. 3.5.

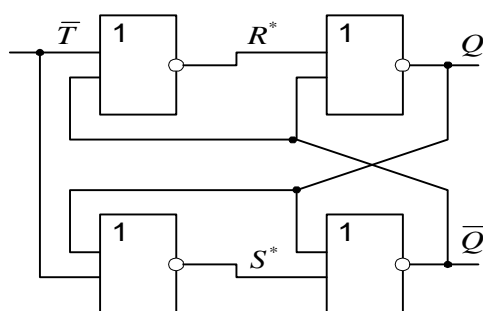


Рис. 3.5. Схема  $T$ -триггера с импульсным управлением в базисе ИЛИ-НЕ



### 3.3. Синхронные триггеры

Фактически синхронные (тактовые) триггеры можно рассматривать как особый тип асинхронных, в которых существуют определенные ограничения на возможность действия информационных сигналов, что позволяет существенно упростить их анализ и синтез.

В синхронных триггерах допускается изменять информационные сигналы в отсутствие действия тактовых импульсов, что предотвращает изменение состояний триггера (предполагается, что синхронный триггер изменяет состояние под действием тактового импульса). Важное преимущество синхронных триггеров – это маскировка эффектов задержек, обусловленных задержками распространения логических элементов и линий передачи информации. Последнее обстоятельство позволяет считать, что логические элементы и линии имеют нулевую задержку, а выходы и состояния триггера можно рассматривать только в фиксированные моменты времени. Эти допущения существенно упрощают анализ и синтез любых асинхронных устройств.

Синхронный  $RS$ -триггер функционирует соответственно табл. 3.8 (характеристическая таблица синхронного  $RS$ -триггера).

Таблица 3.8

Набор	$C$	$R$	$S$	$Q_t$	$Q_{t+1}$	$R^*$	$S^*$
0	0	0	0	0	0	X	1
1	0	0	0	1	1	1	X
2	0	0	1	0	0	x	1
3	0	0	1	1	1	1	X
4	0	1	0	0	0	X	1
5	0	1	0	1	1	1	x
6	0	1	1	0	0	X	1
7	0	1	1	1	1	1	X
8	1	0	0	0	0	x	1
9	1	0	0	1	1	1	X
10	1	0	1	0	0	1	0
11	1	0	1	1	1	1	x
12	1	1	0	0	0	X	1
13	1	1	0	1	1	0	1
14	1	1	1	0	0	X	X
15	1	1	1	1	1	X	x

Предположим, что синхронный  $RS$ -триггер должен быть выполнен в базисе И-НЕ.

Из табл. 3.8 видно, что данный триггер сохраняет свое состояние при  $C = 0$  и работает как асинхронный  $RS$ -триггер при  $C = 1$ .

В табл. 3.9, а, б, в представлены карты Вейча–Карно для тактового триггера.

Таблица 3.9, а

$SQ_t$ $CR$	1 0	1 1	0 1	0 0
1 0	1	1	1	0
1 1	1	1	0	0
0 1	0	1	1	0
0 0	0	1	1	0

$Q_{t+1}$

а)  $Q_{t+1} = SC + \bar{R}Q_t + Q_t\bar{C}$ ,

откуда при  $C = 1$

$$Q_{t+1} = S + \bar{R}Q_t,$$

а при  $C = 0$

$$Q_{t+1} = Q_t.$$

Из карт Вейча–Карно (табл. 3.9, б, в):

Таблица 3.9, б

$SQ_t$ $CR$	1 0	1 1	0 1	0 0
1 0	1	1	1	X
1 1	x	x	0	X
0 1	x	1	1	X
0 0	X	1	1	x

$R^*$

б)  $R^* = \bar{R} + \bar{C} = \overline{RC}$ ;

Таблица 3.9, в

$SQ_t$ $CR$	1 0	1 1	0 1	0 0
1 0	0	x	x	1
1 1	x	x	1	1

01	1	x	x	1
00	1	x	X	1

$S^*$

в)  $S^* = \bar{S} + \bar{C} = \overline{SC}$ .

Выражения

$$R^* = \bar{R} + \bar{C} = \overline{RC}; \quad S^* = \bar{S} + \bar{C} = \overline{SC}$$

определяют структуру тактируемого уровнем «1» *RS*-триггера, приведенного на рис. 3.6.

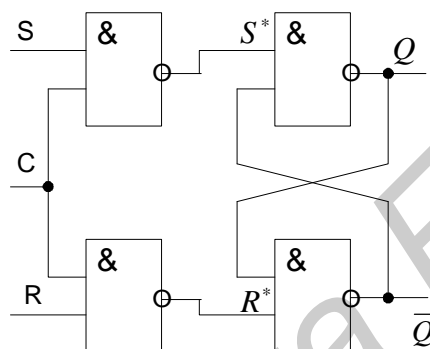


Рис. 3.6. Схема тактируемого триггера

Синхронный *RS*-триггер, равно как и асинхронный *RS*-триггер, имеет запрещенную комбинацию  $RS = 11$ . Его схема широко используется при построении регистров.

Синхронный *D*-триггер функционирует соответственно табл. 3.10, откуда видно, что при  $C = 0$  он сохраняет свое состояние, а при  $C = 1$  работает как асинхронный *D*-триггер.

Таблица 3.10

Набор	$C$	$D$	$Q_t$	$Q_{t+1}$	$R^*$	$S^*$
0	0	0	0	0	x	1
1	0	0	1	1	1	x
2	0	1	0	1	x	1
3	0	1	1	0	1	x
4	1	0	0	0	x	1
5	1	0	1	0	0	1
6	1	1	0	1	1	0
7	1	1	1	1	1	x

Табл. 3.10 представляет собой характеристическую таблицу синхронного *D*-триггера

Из карты Вейча–Карно (табл. 3.11, а)

$$Q_{t+1} = DC + Q_t \bar{C},$$

откуда при  $C = 1$  получим выражение

$$Q_{t+1} = D,$$

а при  $C = 0$

$$Q_{t+1} = Q_t.$$

Таблица 3.11, а

$DQ_t$	00	0 1	1 1	1 0
$C$				
0	1	1	0	0
1	0	1	1	0

$Q_{t+1}$

Таблица 3.11, б

$DQ_t$	00	0 1	1 1	1 0
$C$				
0	1	1	0	X
1	x	1	1	x

$R^*$

Таблица 3.11, в

$DQ_t$	00	0 1	1 1	1 0
$C$				
0	0	x	1	1
1	1	x	x	1

$S^*$

В табл. 3.11 представлены карты Вейча–Карно для тактируемого  $D$ -триггера

Из карт Вейча–Карно (табл. 3.11, б, в)

$$R^* = D + \bar{C} = \overline{DC};$$

$$S^* = \bar{D} + \bar{C} = \overline{DC}.$$

Чтобы устранить инвертор, необходимый для получения переменной  $\bar{D}$ , выражение для  $R^*$  запишем в виде

$$R^* = DC + \overline{C}.$$

Подставляя вместо  $DC$   $\overline{S^*}$  ( $\overline{S^*} = \overline{\overline{DC}} = DC$ ), получаем

$$R^* = \overline{S^*} + \overline{C} = \overline{S^* C}.$$

В результате получаем следующую схему триггера (рис. 3.7).

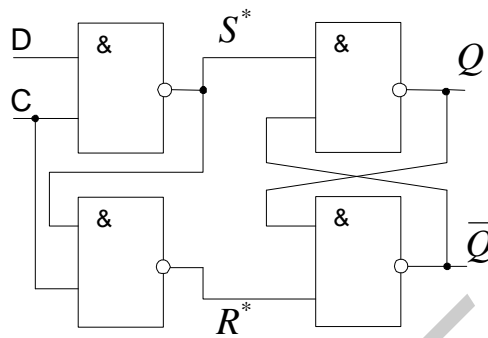


Рис. 3.7. Схема тактового  $D$ -триггера в базисе И-НЕ

Данная схема широко используется для построения регистров и в устройствах управления.

### 3.4. Синтез триггерной структуры на универсальном $D$ -триггере

Универсальные триггеры – триггеры, которые могут работать в разных режимах и как разные типы триггеров. Обозначения универсальных триггеров представлены на рис. 3.8.

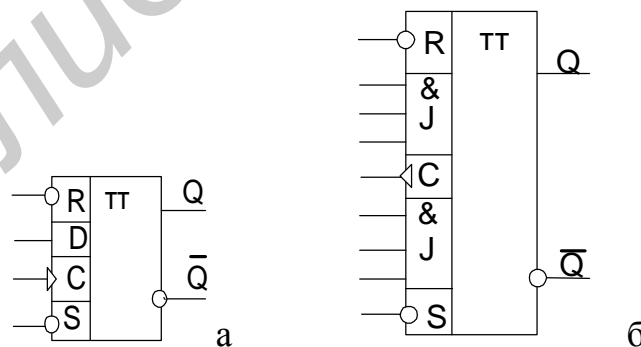


Рис. 3.8. Условное обозначение универсальных  $D$ -триггеров (а) и  $JK$ -триггеров (б)

Разработаем тактируемый переходом 01 триггер с двумя входами, который функционирует соответственно табл. 3.12 на базе универсального  $D$ -триггера .

Таблица 3.12

$X_1$	$X_2$	$Q_t$
0	0	0
0	1	$\overline{Q_t}$
1	0	$\overline{Q_t}$
1	1	1

Структуру синтезированного триггера в общем виде представим так, как показано на рис. 3.9.

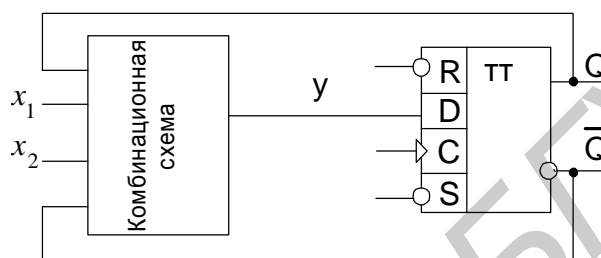


Рис. 3.9. Структура синтезируемого триггера на базе универсального  $D$ -триггера

Задача синтеза состоит в построении комбинационной схемы, которая вырабатывает сигнал  $y = D$ , что обеспечивает необходимый закон функционирования.

Таблица 3.13

Набор	$X_1$	$X_2$	$Q_t$	$Q_{t+1}$	$y = D$	Набор	$X_1$	$X_2$	$Q_t$	$Q_{t+1}$	$y = D$
0	0	0	0	0	0	4	1	0	0	1	1
1	0	0	1	0	0	5	1	0	1	0	0
2	0	1	0	0	0	6	1	1	0	1	1
3	0	1	1	1	1	7	1	1	1	1	1

Из таблицы переходов триггера (табл. 3.13) вытекает, что

$$y = D = Q_{t+1}.$$

Из карты Вейча–Карно (табл.3.14) для синтезируемого триггера

$$y = D = Q_t X_1 + X_2 \overline{Q_t}.$$

Таблица 3.14

$X_2 Q_t$	00	0 1	1 1	1 0
-----------	----	-----	-----	-----

$X_1$				
0	0	0	1	0
1	1	0	1	1

В результате получаем следующую схему (рис. 3.10).

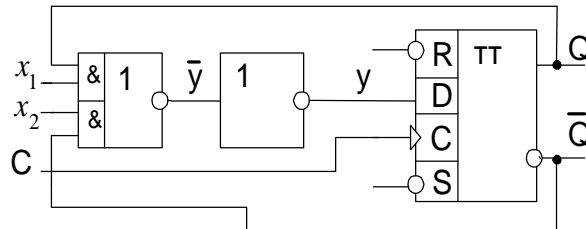


Рис. 3.10. Окончательная схема триггера

## 4. Синтез счетчиков

### 4.1. Логические методы синтеза счетчиков

Логические методы синтеза счетчиков основываются на применении теории конечных автоматов.

Наиболее широко известный логический метод синтеза счетчиков основан на совместном решении прикладных уравнений счетчика с характеристическими уравнениями триггеров. Здесь используются, как правило, синхронные триггеры  $RS$ ,  $T$ ,  $D$  и  $JK$ -типов, хотя принципиально возможно применение и других типов триггеров. В данном параграфе этот метод иллюстрируется на триггерах  $JK$ -типа по той основной причине, что они входят в состав практически всех отечественных и зарубежных серий ИМС. Синтез счетчика, как и триггеров, начинается с этапа абстрактного синтеза. В данном случае абстрактный синтез включает в себя следующие пункты:

- определение числа элементов памяти (объем памяти счетчика);
- кодирование внутренних и выходных состояний счетчика;
- установление связи между входными, внутренними и выходными состояниями (описывается закон функционирования счетчика).

Объем памяти автомата вычисляется по формуле (4.1).

$$N_{p.cч} = (\log_2 K_{cч})^+, \quad (4.1)$$

где  $K_{cч}$  – коэффициент счета счетчика (число фиксированных состояний счетчика).

Два других пункта включают в себя составление таблицы, в которую заносятся внутренние состояния счетчика и порядок их изменения, а также состояния, в которые должен устанавливаться счетчик под воздействием

входных сигналов. Процедуру синтеза рассмотрим на ряде конкретных примеров применительно к синхронным счетчикам.

#### 4.2. Синтез счетчика с фиксированным коэффициентом счета

Требуется синтезировать счетчик с  $K_{сч} = 5$ , считающий в такой последовательности: 0, 1, 2, 3, 4, 0, 1, ....

Счетчик имеет восемь состояний, следовательно, согласно выражению (4.1) для его построения потребуется три триггера, которые надо соединить так, чтобы выполнялась заданная последовательность переходов. Обозначим эти триггеры через  $A, B, C$ . Предположим, что до прихода входного сигнала счетчик находится, например, в состоянии 0 ( $Q_A^n = Q_B^n = Q_C^n = 0$ ). Тогда при поступлении входного импульса счетчик в соответствии с заданными состояниями должен перейти в состояние, соответствующее коду цифры 1 ( $Q_A^n = Q_C^n = 0, Q_B^n = 1$ ) и т.д. Словесное описание работы счетчика представим в виде табл. 4.1. Здесь индексами  $Q_A^n, Q_B^n, Q_C^n$  обозначены состояния счетчика до прихода входного сигнала (внутренние состояния автомата), а индексами  $Q_A^{n+1}, Q_B^{n+1}, Q_C^{n+1}$  – состояния счетчика после прихода входного сигнала (выходные состояния автомата). На этом этап абстрактного синтеза заканчивается.

Порядок переключения состояний счетчика приведен в табл. 4.1. В таком счетчике возможны три избыточных кода, которым соответствуют произвольные состояния триггеров, что отмечено знаком \* в соответствующих строках столбцов  $Q^{n+1}$ .

Таблица 4.1

Состояния счетчика		$Q_A^n$	$Q_B^n$	$Q_C^n$	$Q_A^{n+1}$	$Q_B^{n+1}$	$Q_C^{n+1}$
Рабочие	1	0	0	0	0	0	1
	2	0	0	1	0	1	0
	3	0	1	0	0	1	1
	4	0	1	1	1	0	0
	5	1	0	0	0	0	0
Избыточные	6	1	0	1	*	*	*
	7	1	1	0	*	*	*
	8	1	1	1	*	*	*

Этап структурного синтеза включает в себя составление прикладных уравнений триггеров, их минимизацию, последующее совместное решение характеристических и прикладных уравнений и собственно составление функциональной схемы счетчика.

Прикладными называют уравнения, которые описывают поведение каждого триггера в момент  $t^{n+1}$  как функцию переменных всех триггеров в момент  $t^n$ . Другими словами, это уравнение записывается в виде



$$Q_i^{n+1} = f(Q_A^n, Q_B^n, Q_C^n \dots Q_n^n).$$

После нанесения полученных уравнений на карты Вейча–Карно (табл. 4.2, а–в) проводится этап минимизации. Однако в данном случае минимизация будет выполняться таким образом, чтобы полученное выражение в каждом из логических произведений обязательно содержало буквы (в прямом или инверсном виде), имеющие тот же индекс, что и у минимизируемого выражения. Например, если минимизируется выражение для  $Q_A^{n+1}$ , то в каждом из его членов должна присутствовать либо буква  $Q_A$ , либо  $\overline{Q_A}$ . Смысл такой минимизации будет очевиден далее.

В табл. 4.2, а–в приведены карты Вейча–Карно для выражений  $Q_A^{n+1}$ ,  $Q_B^{n+1}$  и  $Q_C^{n+1}$ : двоичного счетчика с  $K_{сч}=5$ : а – для триггера А; б – для триггера В; в – для триггера С.

Таблица 4.2

а	$QaQ$	00	0 1	1 1	1 0
	$b$				
	$Qc$				
	0	0	0	*	0
	1	0	1	*	*
б	$QaQ$	00	0 1	1 1	1 0
	$b$				
	$Qc$				
	0	0	1	*	0
	1	1	0	*	*
в	$QaQ$	00	0 1	1 1	1 0
	$b$				
	$Qc$				
	0	1	1	*	0
	1	0	0	*	*

В результате минимизации получаем

$$Q_A^{n+1} = Q_B \cdot Q_C \cdot \overline{Q_A}; \quad Q_B^{n+1} = \overline{Q_B} \cdot Q_C + Q_B \cdot \overline{Q_C}; \quad Q_C^{n+1} = \overline{Q_A} \cdot \overline{Q_C}.$$

Записав систему уравнений

$$J_A \cdot \overline{Q_A} + \overline{K_A} \cdot Q_A = \overline{Q_B} \cdot Q_C \cdot \overline{Q_A} + Q_A \cdot 0;$$

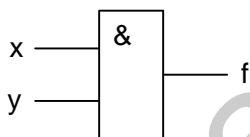
$$J_B \cdot \overline{Q_B} + \overline{K_B} \cdot Q_B = \overline{Q_B} \cdot Q_C + Q_B \cdot \overline{Q_C};$$

$$J_C \cdot \overline{Q_C} + \overline{K_C} \cdot Q_C = \overline{Q_C} \cdot \overline{Q_A} + Q_C \cdot 0$$

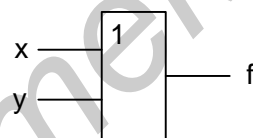


– вентильном, регистровом и т. д. При моделировании на вентильном уровне логические формулы выражаются в булевом базисе И, ИЛИ, НЕ, поэтому и логические схемы обычно строятся с помощью аналогичного набора вентилях. Между тем могут оказаться удобными и другие вентили, преобразование к которым обеспечивается надлежащей методикой. В частности, широко используются логические схемы на основе вентиля И-НЕ, реализующего штрих Шеффера, а также ИЛИ-НЕ, реализующего стрелку Пирса, что допустимо благодаря функциональной полноте каждой из этих операций (см. рис. 5.2).

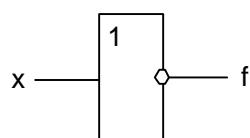
Логические схемы непосредственно не связаны с физической природой и конструкцией вентилях. Тем не менее при выборе функционального базиса для реализации логических функций учитываются технико-экономические характеристики реальных компонентов цифровых устройств с тем, чтобы с наибольшей эффективностью обеспечить техническую реализацию логических схем.



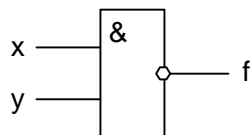
а) И (конъюнктор)



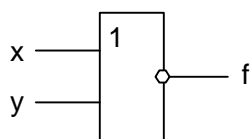
б) ИЛИ (дизъюнктор)



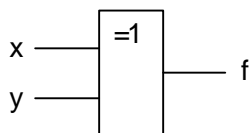
в) НЕ (инвертор)



г) И-НЕ (штрих Шеффера)



д) ИЛИ-НЕ (стрелка Пирса)



е) Исключающее ИЛИ

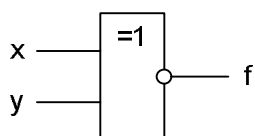
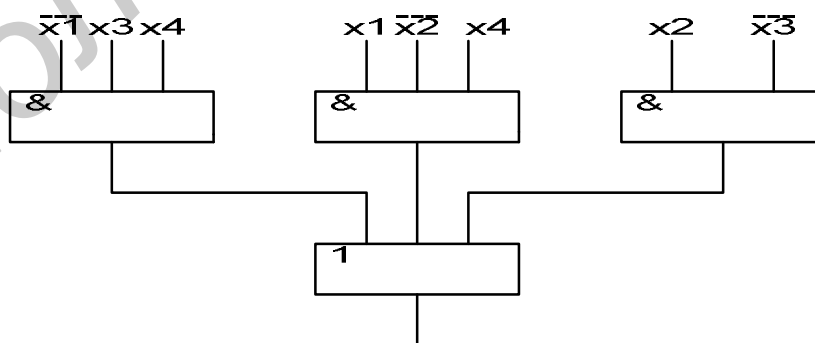


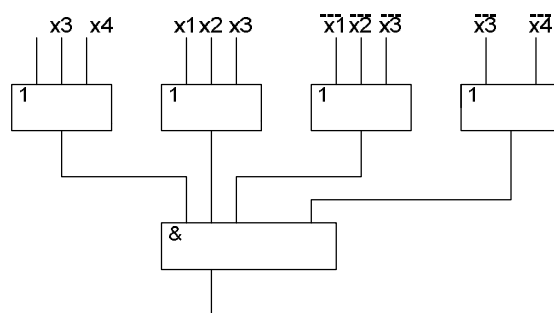
Рис. 5.2. Структурные модели основных логических элементов

Переход от булевой функции к логической схеме в булевом базисе очевиден: достаточно в соответствии с формулой обозначить входы вентиля и соединить их между собой надлежащим образом. Так, на рис. 5.3 показаны логические схемы, реализующие минимальные формы.

Функции, заданные в нормальной дизъюнктивной или конъюнктивной форме, реализуются двухступенчатыми схемами. Первая ступень реализует произведения или суммы переменных, а вторая – соответственно суммы минтермов или произведения макстермов. Двухступенчатые схемы предпочтительны по быстродействию, которое вследствие инерционности логических вентилях пропорционально числу ступеней. Однако в инженерной практике приходится по различным причинам обращаться к многоступенчатым схемам.



а) 
$$y = \bar{x}_1 x_3 x_4 + x_1 \bar{x}_2 x_4 + x_2 \bar{x}_3$$



$$\text{б) } y = (x_2 + x_3 + x_4)(x_1 + x_2 + x_3)(\bar{x}_1 + \bar{x}_2 + \bar{x}_3)(\bar{x}_3 + \bar{x}_4).$$

Рис. 5.3. Примеры реализации аналитических выражений

## 5.2. Многоступенчатые реализации

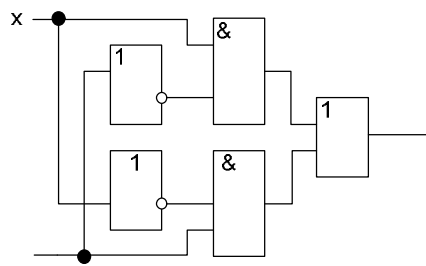
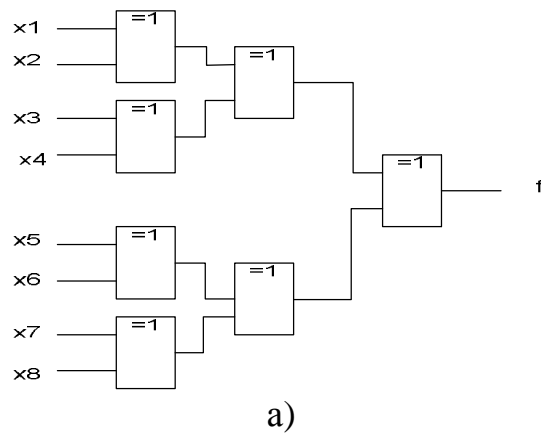
Характерным примером многоступенчатых схем может служить реализация функции, обеспечивающей проверку на четность набора из переменных. Она равна единице тогда и только тогда, когда число единичных значений входных переменных нечетное и выражается как сумма по модулю 2:

$$f(X_1, X_2, \dots, X_n) = X_1 \oplus X_2 \oplus \dots \oplus X_n.$$

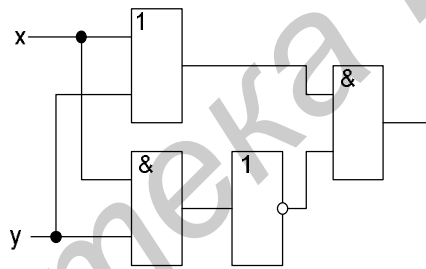
Реализация этой функции двухступенчатой схемой при значительном числе  $n$  нереальна, так как потребовалось бы  $2^{n-1}$  вентилях И на  $n$  входов каждый и один вентиль ИЛИ на  $2^{n-1}$  входов. Это следует из того, что среди  $2^n$  наборов значений переменных всегда половина нечетных и каноническая форма функции содержит  $2^{n-1}$  минтермов. К тому же она не минимизируется, так как не содержит ни одной пары склеивающихся минтермов (на карте Карно отмеченные ячейки располагаются в шахматном порядке). В то же время данная функция реализуется многоступенчатой схемой, состоящей из двухвходовых вентилях «исключающее ИЛИ», которые выполняют операцию «сумма по модулю 2», причем для функции  $n$  переменных таких вентилях требуется  $(n - 1)$ . Соответствующая схема при  $n = 8$  показана на рис. 5.4, а.

Реализация в булевом базисе требует замены вентилях «исключающее ИЛИ» соответствующими булевыми эквивалентами – дизъюнктивными (рис. 5.4, б) или конъюнктивными (рис. 5.4, в). В результате функция реализуется двухвходовыми вентилями И и ИЛИ совместно с вентилями НЕ, причем всего требуется

$3(n - 1)$  логических вентилях и  $2(n - 1)$  инверторов.



б)  $x \oplus y = x\bar{y} + \bar{x}y$



в)  $x \oplus y = (x + y)(\bar{x} + \bar{y}) = (x + y)\bar{x}y$

Рис. 5.4. Примеры многоступенчатых реализаций

Многоступенчатые реализации появляются также вследствие преобразований стандартных форм, вызванных практическими ограничениями по числу входов вентиля (*коэффициенту разветвления*) и числу выходов (*коэффициенту нагрузки*).

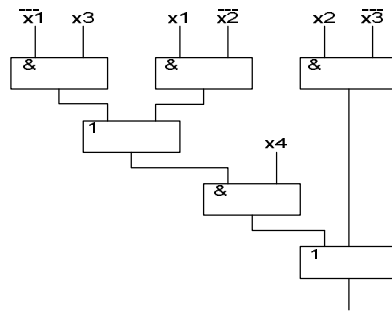
В одном из способов уменьшение числа входов достигается разложением булевой формулы на множители. Например, функция

$$y = \bar{x}_1\bar{x}_3x_4 + \bar{x}_1x_2x_4 + x_2x_3,$$

двухступенчатая реализация которой показана на рис. 5.3, а, преобразуется к виду

$$y = (\bar{x}_1\bar{x}_3 + \bar{x}_1x_2)x_4 + x_2\bar{x}_3.$$

Соответствующая многоступенчатая реализация, в которой используются только двухвходовые вентили, показана на рис. 5.5.



$$y = (\bar{x}_1 x_3 + x_1 \bar{x}_2) x_4 + x_2 \bar{x}_3$$

Рис. 5.5. Иллюстрация уменьшения числа входов путем разложения булевой формулы на множители

### 5.3. Базисы И-НЕ И ИЛИ-НЕ

Вследствие функциональной полноты функций Шеффера и стрелки Пирса (см. рис. 5.2) реализующие их вентили И-НЕ или ИЛИ-НЕ могут представлять любую булеву операцию И, ИЛИ, НЕ и таким образом самостоятельно образовать базис, в котором реализуется любая логическая функция. Это целесообразно с двух точек зрения. Во-первых, при проектировании логических схем можно обойтись одним единственным типом вентиля, что позволяет предельно унифицировать этот процесс. Во-вторых, для большинства серий ТТЛ- и КМОП-логик вентиль И-НЕ, как и ИЛИ-НЕ, является базисным и предпочтителен во многих отношениях. Вследствие этого реализация логических схем в базисах И-НЕ и ИЛИ-НЕ получила широкое распространение на практике.

Булевы операции И, ИЛИ, НЕ выражаются через штрих Шеффера.

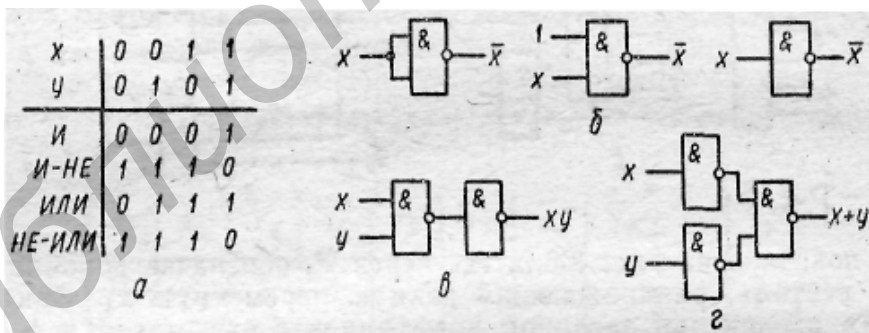


Рис. 5.6. Пример реализации логических операций инверсии дизъюнкции и конъюнкции в базисе И-НЕ

Отсюда следует, что вентиль И-НЕ, таблица соответствия которого дана на рис. 5.8, а, эквивалентен вентилю НЕ-ИЛИ (см. рис. 5.2) и позволяет реализовать булевы операции, как показано на рис. 5.8, б–г. Для реализации инверсии имеются два варианта: либо на все входы вентиля И-НЕ подается переменная  $x$ , либо на все входы, кроме входа  $x$ , подается единица. Обычно эти особенности на схемах отсутствуют и используется упрощенное представление с одним входом (рис. 5.6, б).

Переход к базису И-НЕ осуществляется проще всего для двухуровневых схем И/ИЛИ или при задании функции в стандартной форме суммы минтермов, что видно на простом примере  $f = xy + zv$  (рис. 5.7). Исходная схема в булевом базисе (рис. 5.7, а) преобразуется к такому виду (рис. 5.7, б), что пары инверторов оказываются соединенными последовательно и выполняют двойную инверсию сигналов. После их удаления схема с вентилями И-НЕ существенно упрощается (рис. 5.7, в) и по своей структуре полностью совпадает с исходной схемой. Таким образом, для перехода от двухуровневой схемы И-ИЛИ к схеме в базисе И-НЕ достаточно заменить все вентили вентилями И-НЕ.

В случае произвольных многоступенчатых схем сначала преобразуются по изложенному выше правилу соседние уровни И-ИЛИ, а остальные вентили И и ИЛИ заменяются их эквивалентами в базисе И-НЕ (см. рис. 5.7). В качестве примера на рис. 5.7, а показана схема, полученная в результате такого преобразования из схемы на рис. 5.7, в.

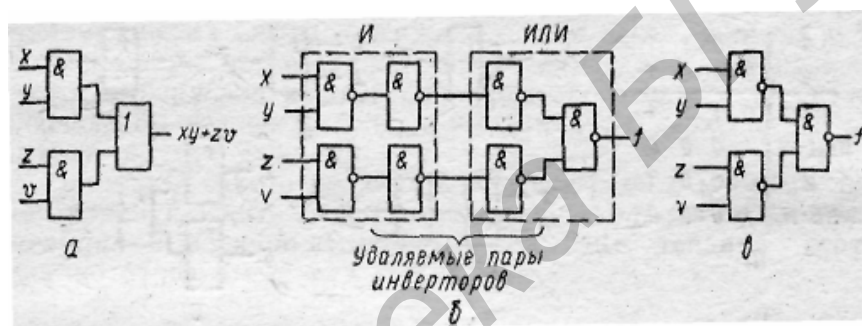


Рис. 5.7. Реализация логической схемы в базисе И-НЕ

Операция ИЛИ-НЕ, реализующая стрелку Пирса

$$x \downarrow y = \overline{x + y} = \overline{x} + \overline{y},$$

позволяет выразить булевы функции соотношениями

$$xy = \overline{\overline{x} + \overline{y}} = \overline{x \downarrow y}, \quad x + y = \overline{\overline{x} \downarrow \overline{y}} = \overline{x \downarrow y}, \quad \overline{x} = \overline{x + x} = x \downarrow x.$$

Отсюда следует, что вентиль ИЛИ-НЕ, таблица соответствия которого дана на рис. 5.8, а, позволяет реализовать булевы операции, как показано на рис. 5.8, б–г. Вследствие дуальности операций И-НЕ и ИЛИ-НЕ соответствующие реализации вместе с сопровождающими их комментариями можно получить на основе принципа дуальности.



x	0	0	1	1
y	0	1	0	1
ИЛИ	0	1	1	1
ИЛИ-НЕ	1	0	0	0
И	0	0	0	1
НЕ-И	1	0	0	0

а

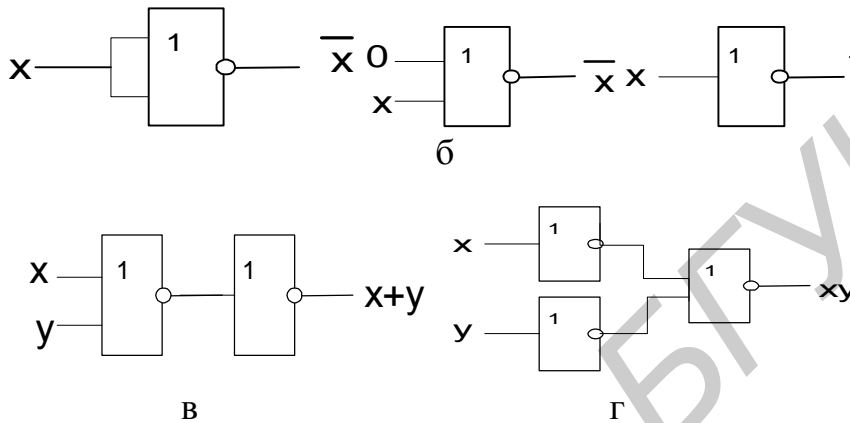


Рис. 5.8. Пример реализации основных логических операций в базисе ИЛИ-НЕ

На рис. 5.9, дуальном рис. 5.7, показан переход от двухуровневой схемы ИЛИ/И, которая соответствует конъюнктивной нормальной форме

$$f = (x + y)(z + v),$$

к реализации в базисе ИЛИ-НЕ. Здесь исходная схема (рис. 5.9, а) преобразуется так, что можно устранить последовательно включенные пары инверторов (рис. 5.9, б). После этого получаем схему (рис. 5.9, в), которая по своей структуре полностью совпадает с исходной.

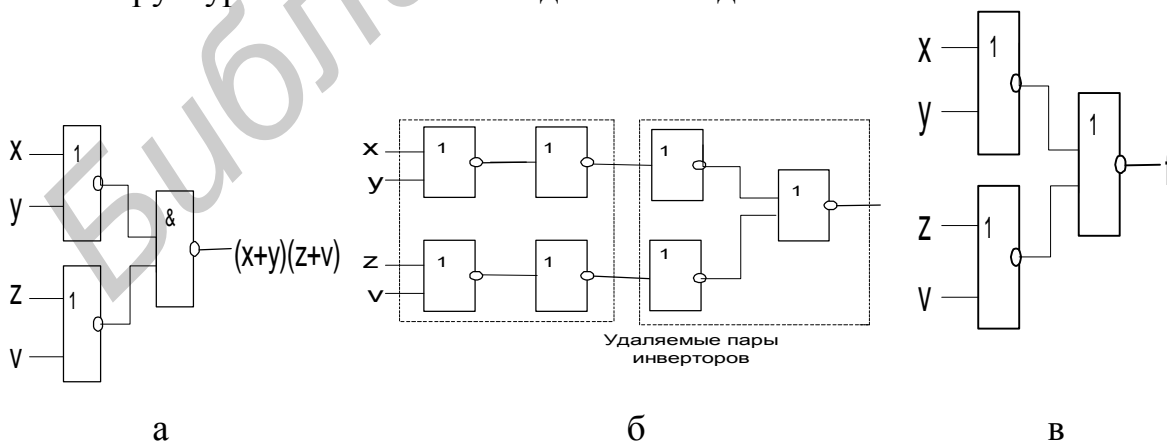


Рис. 5.9. Пример реализации логической схемы в базисе ИЛИ-НЕ

## 6. Обоснование выбора элементной базы

### 6.1. Серии логических элементов

*Серией* микросхем называют группу микросхем, выполненных по одинаковой или близкой технологии, имеющих сходные технические характеристики и предназначенных для совместной работы в составе цифровой аппаратуры.

Условное обозначение логической микросхемы состоит из следующих элементов:

1) *буквы*, в большой степени характеризующей стойкость микросхемы к воздействию окружающей среды и связанный с этим тип корпуса. Отсутствие буквы рассматривается как своего рода «нулевая буква»;

2) *трех или четырех цифр*, обозначающих номер серии. При этом предыдущую букву часто считают первым элементом наименования серии;

3) *двух букв*, характеризующих выполняемую функцию (функциональная группа);

4) *одной или двух цифр*, обозначающих тип микросхемы внутри функциональной группы, например, число входов элемента И, тип триггера и т.п.;

5) *буквы*, характеризующей возможные вариации значений некоторых параметров, например выходной мощности. Этой буквы чаще всего не бывает.

Пример: K155JA2 – микросхема серии K155, выполняющая функцию И-НЕ, второго типа (в серии K155 этот тип имеет 8 входов).

Микросхемы заключены в стандартные корпуса в основном с двумя типами выводов:

1) *перпендикулярными плоскостями корпуса*, с шагом 2,5 мм, которые вставляются в отверстия монтажной платы и распаиваются на стороне платы, противоположной корпусу. Такие корпуса иногда называют корпусами *типа ДИП* (*DIP – dual in-line package* – корпус с двумя рядами выводов). В корпуса ДИП чаще всего заключаются *микросхемы широкого применения*, имеющие перед номером серии буквы К или КМ. Обычный температурный диапазон микросхем с буквой К минус 10–плюс 70°C, а с буквами КМ минус 45–плюс 85°C. В микросхемах широкого применения сравнительно новых выпусков пластмассовый корпус часто обозначается буквами КР;

2) *плоскими (планарными)*, которые накладываются на плату и распаиваются на той же стороне, где находится и сам корпус; шаг выводов 1,25 мм. В таких корпусах обычно выпускаются серии без буквы перед номером. Типичный диапазон их рабочих температур минус 60 – плюс 125°C. Подробнее об обозначениях микросхем, составе различных серий, условиях их эксплуатации и т.п. см. [6–8] или отраслевые справочники.

Габариты микросхемы определяет не кристалл кремния, а выводы из корпуса. Поэтому если элементы простые, то, чтобы использовать по возможности все выводы стандартного корпуса, в нем размещают несколько

элементов. Простые логические элементы обычно размещаются в корпусах с 14 выводами, из которых один вывод – это питание и один вывод – общий провод всех логических входов, выходов и питания, кратко называемый *общим* или, менее строго, – *земля*. Оставшиеся 12 выводов – логические. Примеры состава корпусов: 4×(2И) – четыре двухвходовых элемента И (заняты все 12 логических выводов), 2×(4 И-НЕ) – два четырехвходовых И-НЕ (не использованы только два вывода). Более сложные логические узлы размещаются в корпусах с 16, 24 и большим числом выводов.

В настоящее время наиболее распространены три технологии изготовления логических элементов: ТТЛ (иТТЛШ), КМДП и ЭСЛ.

Для технологии ТТЛ (*транзисторно-транзисторной логики*) и ТТЛШ (ТТЛ с диодами Шотки) самыми удобными для изготовления являются элементы И-НЕ. Именно они шире всего используются для построения более сложных узлов. Второй особенностью является простота изготовления двухступенчатых логических элементов И-ИЛИ-НЕ. В текстовых документах и справочниках логические возможности таких элементов принято обозначать в сокращенной форме. Принцип обозначения понятен из следующих примеров:

2-2-2И-3ИЛИ-НЕ – функция  $Y = \overline{ab \vee cd \vee ef}$  ;

3-3И-2ИЛИ-НЕ – функция  $Y = \overline{abc \vee def}$  ;

2-2-4И-3ИЛИ-НЕ – функция  $Y = \overline{ab \vee cd \vee efgh}$ .

В серии ТТЛ есть *расширяемые* элементы И-ИЛИ-НЕ, к которым можно подключать *расширители*, увеличивая тем самым число входов ИЛИ (рис. 6.1). Обозначение метки *EX* происходит от *expand* – расширять. Сумма входов ИЛИ, имеющихя внутри корпуса и добавляемых за счет расширителей, обычно ограничена числом 8. Как правило, выпускаются расширители двух типов – с числом входов И, равным 4 и 8. На рис. 6.1 показаны различные по степени подробности способы изображения расширяемого элемента И-ИЛИ-НЕ с подключенными расширителями.

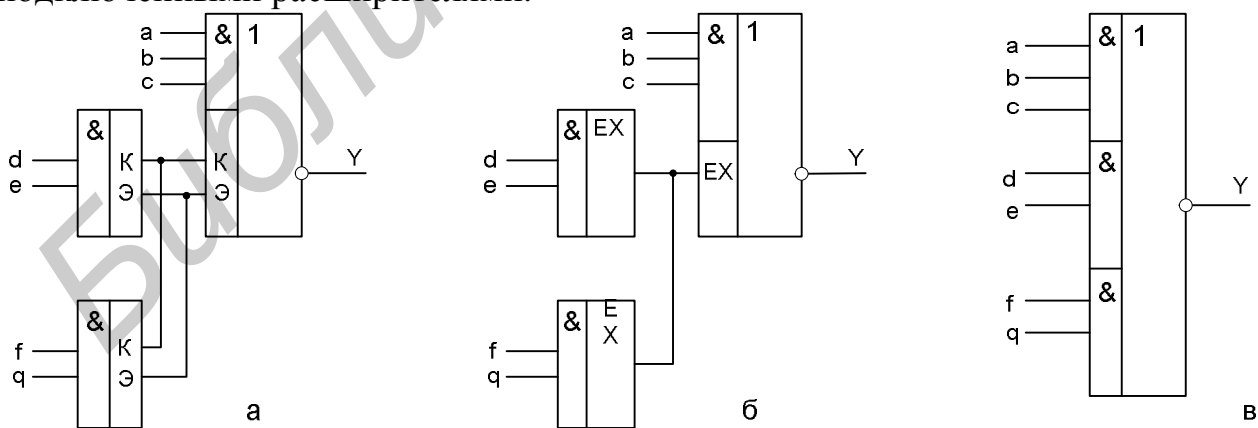


Рис. 6.1. Способы изображения расширяемого элемента И-ИЛИ-НЕ с расширителями:

а – для электрических схем; б – для функциональных подробных схем (отмечен факт использования расширителей); в – для функциональных

схем общих (отмечена лишь результирующая функция комплекса

$$Y = abc \vee de \vee fg)$$

Элементы ТТЛ, а тем более их усовершенствованная модификация – ТТЛШ, имеют хорошее быстродействие, удовлетворительные электрические и эксплуатационные характеристики, хорошо отработаны технологически, следствием чего является достаточно высокая надежность. Им не свойственны труднообъяснимые эффекты при неисправностях, как у элементов КМДП. В одном устройстве допустимо использование элементов ТТЛ и ТТЛШ различных серий. Большинство микропроцессорных БИС и БИС памяти согласовано по питанию и уровням сигналов с элементами ТТЛ. Серии ТТЛ и ТТЛШ – наиболее распространены и популярны у разработчиков цифровой автоматики. Подробнее о них см. [6, 8, 12].

*Комплементарные* (взаимно дополняющие) МДП-структуры (КМДП), построенные на основе МДП-транзисторов с различным типом проводимости. В КМДП-схемотехнике операции И и ИЛИ реализуются комбинированием последовательного и параллельного включения МДП-транзисторов. Серьезных технологических преимуществ у операции И перед операцией ИЛИ нет, поэтому в КМДП-сериях одинаково широко используются и элементы И-НЕ и ИЛИ-НЕ. Готовых двухступенчатых функций типа И-ИЛИ в КМДП-сериях в отличие от ТТЛ пользователю обычно не предоставляется, однако в состав серии, как правило, входят транзисторные сборки, позволяющие самостоятельно набирать любые функции непосредственно из МДП-транзисторов. Возможность строить функции из элементов И, ИЛИ и т. п. при этом также остается.

Элементы КМДП исключительно экономичны по потребляемой мощности, что на сегодня является их основным достоинством. Кроме того, они способны работать в широком диапазоне напряжений питания, например от 3 до 15 В, имеют хороший порог помехоустойчивости. Недостатком их является пока еще заметно меньшее, чем у ТТЛ, быстродействие.

Основы схемотехники на МДП-транзисторах даны в [13]. Подробно микросхемы КМДП описаны в [10] и особенно в [9], справочные данные по КМДП-сериям можно найти в [6, 7].

Эмиттерно-связанная логика (ЭСЛ) использует принцип переключения тока из одного транзистора в другой в паре транзисторов, работающих на общее эмиттерное сопротивление. Этот принцип обеспечивает работу транзисторов без насыщения, поэтому ЭСЛ-серии являются на сегодня наиболее быстродействующими. Однако реализация быстродействия ЭСЛ-элементов возможна лишь при высокой грамотности проектирования и технологической культуре изготовления монтажа. Поэтому на элементах ЭСЛ строят наиболее быстродействующие большие ЭВМ, а в цифровых устройствах автоматики ЭСЛ-серии практически не используют. Применению элементов ЭСЛ в небольших автономных устройствах автоматики мешает также

несогласованность уровней их сигналов с сериями ТТЛ, КМДП, большинством серий выпускаемых микросхем памяти, микропроцессоров, а также недостаточно широкий диапазон рабочих температур.

Принципы построения логических схем на элементах ЭСЛ те же, что и на элементах других технологий. Подробно методы работы с ЭСЛ-элементами изложены в [11], справочные сведения можно найти в [6,7].

В табл. 6.1 приведены наборы микросхем отдельных логических элементов, выпускаемых в рамках некоторых широко распространенных серий ТТЛ, ТТЛШ, КМДП. Такие микросхемы называют микросхемами *малой интеграции*, а на арго схемотехников – *россыпью*.

Таблица 6.1

Основные параметры и выполняемая функция	Технология					
	ТТЛ		ТТЛШ		КМДП	
	Серия					
	134 К134	133 К155	530 К531	533 К555	164 К176	564 К561
Типовая средняя задержка, нс	80	19	5	20	200	100
Типовая средняя потребляемая мощность одним элементом в статике	2 мВт	20 мВт	35 мВт	4 мВт	1,5 мкВт	0,7 мкВт
6НЕ		ЛН1	ЛН1	ЛН1		ЛН1
4 (2И-НЕ)	ЛБ1	ЛА3	ЛА3	ЛА3	ЛА7	ЛА7
3 (3И-НЕ)		ЛА4	ЛА4	ЛА4	ЛА9	ЛА9
2 (4И-НЕ)	ЛБ2	ЛА1	ЛА1	ЛА1	ЛА8	ЛА8
8И-НЕ	ЛА2	ЛА2	ЛА2	ЛА2		
4 (2ИЛИ-НЕ)		ЛЕ1	ЛЕ1	ЛЕ1	ЛЕ5	ЛЕ5
3 (3ИЛИ-НЕ)					ЛЕ10	ЛЕ10
2 (4ИЛИ-НЕ)		ЛЕ2			ЛЕ6	ЛЕ6
4 (2И)		ЛИ1		ЛИ1		
3 (3И)			ЛИ3	ЛИ3		
2 (4И)				ЛИ6		
4 (2ИЛИ)		ЛЛ1	ЛЛ1	ЛЛ1		
4 (2М2)		ЛП5	ЛП5	ЛП5	ЛП2	ЛП2
2-2И-2ИЛИ-НЕ+2-4И-2ИЛИ-НЕ	ЛР1					
2-2И-2ИЛИ-НЕ+3-3И-2ИЛИ-НЕ			ЛР11	ЛР11		
2-2-3-4И-4ИЛИ-НЕ	ЛР2		ЛР9			
2 (2-2И-2ИЛИ-НЕ) один с расширением по ИЛИ	ЛР4	ЛР1				
4-4И-2ИЛИ-НЕ с расширением по ИЛИ		ЛР4				
		ЛР3				
		ЛД1				
2-2-2-3И-4ИЛИ-НЕ с расширением по ИЛИ		ЛД3				

2 (4И – расширитель по ИЛИ) 8И – расширитель по ИЛИ						
--	--	--	--	--	--	--

Серии цифровых микросхем в большой мере унифицированы в международном масштабе. Данные по зарубежным аналогам выпускаемых в нашей стране микросхем можно найти в [8].

## 6.2. Временные характеристики логических элементов

Процесс изменения напряжения от низкого  $L$  уровня к высокому  $H$  называют *фронтом* сигнала, а обратный процесс – *срезом*. Иногда говорят *положительный фронт* и *отрицательный фронт*, а если приходится говорить о фронтах и срезах одновременно, то говорят *фронты*. Длительность перехода элемента из 0 в 1 и наоборот измеряется между специально оговоренными в ТУ измерительными уровнями, отличающимися от номинальных уровней 0 и 1 на 0,1–0,3 амплитуды сигнала. *Время перехода из 0 в 1* обозначается  $t^{01}$ , *время перехода из 1 в 0* –  $t^{10}$ . Как правило, эти времена не одинаковы и оба приводятся в справочниках. Задержка между фронтами сигналов на входе и выходе логического элемента называется *задержкой распространения сигнала* и измеряется на оговоренном в ТУ уровне, близком к половине единичного уровня. Различают задержку распространения при переключении выхода элемента из 0 в 1 –  $t_{зд.р}^{01}$  и задержку распространения при переключении выхода из 1 в 0 –  $t_{зд.р}^{10}$ . Как и фронты, эти задержки также обычно не равны друг другу.

При прохождении перепада сигнала по достаточно длинной цепочке инвертирующих элементов число элементов, переключающихся в 1 и 0, будет практически одинаково и равно половине общего числа элементов в цепочке. Поэтому для оценки задержки распространения всей цепочки можно пользоваться *средним временем задержки распространения сигнала*  $t_{зд.р.ср}$ , равным полусумме  $t_{зд.р}^{10}$  и  $t_{зд.р}^{01}$ , умножая его на число элементов цепочки. В силу распространенности инвертирующих элементов такой способ оценки задержки широко используется.

На рис. 6.2 показан процесс распространения фронта сигнала по цепочке элементов. Общее время  $T$  переходного процесса складывается из суммы задержек распространения всех элементов цепочки плюс половина фронта входного сигнала плюс половина фронта сигнала на выходе последнего элемента. Поскольку и задержки, и длительности фронтов элемента – величины одного порядка, то при достаточно длинной цепочке слагаемым в одну длительность фронта можно пренебречь и оценивать время переходного процесса лишь суммой задержек распространения элементов. На практике так обычно и поступают, кроме случаев анализа сигналов обратной связи, если они проходят по очень короткой цепочке.

На рис. 6.3, в показан в общем виде некоторый функциональный узел  $A$  с двумя входами и двумя выходами. При постоянном уровне сигнала на входе  $a$  и изменении сигнала на входе  $b$  узел  $A$  придется характеризовать двумя значениями задержек распространения – по тракту  $bY$  и тракту  $bZ$ . Более того, при  $a = 0$  и  $a = 1$  число переключающихся элементов, т.е. *глубина трактов*  $bY$  и  $bZ$  в общем случае не одинакова, и тогда переходный процесс в узле  $A$ , вызванный изменением сигнала на входе  $b$ , придется характеризовать уже четырьмя значениями задержки:

- от  $b$  до  $Y$  при  $a=0$ ;
- от  $b$  до  $Z$  при  $a = 0$ ;
- от  $b$  до  $Y$  при  $a = 1$ ;
- от  $b$  до  $Z$  при  $a = 1$ .

Аналогичной четверкой будет характеризоваться и задержка сигнала от входа  $a$  до выходов  $Y$  и  $Z$ . Схему с большим, чем на рис. 6.2, в, числом входов и выходов нужно в общем случае характеризовать довольно сложной матрицей. К счастью, в реальных схемах число временных параметров так быстро не растет. Выручают схемная идентичность многих входов, позволяющая объединять их в группы с равной задержкой, и отсутствие влияния большинства входов на глубину чужих трактов. Поэтому редкие узлы характеризуются более чем шестью значениями задержки.

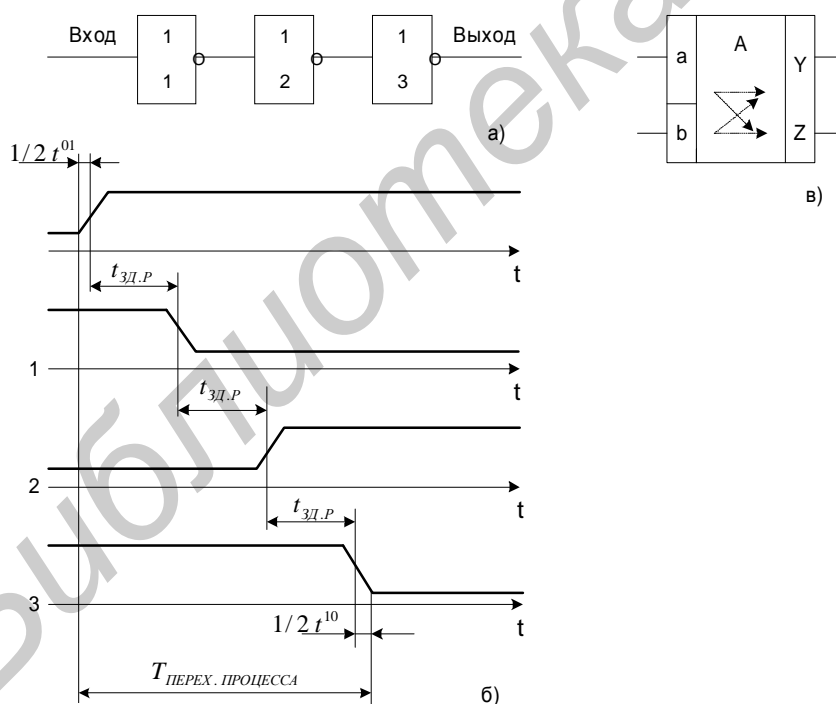


Рис. 6.2. Переходный процесс в группе элементов:

а – цепочка элементов; б – временная диаграмма переходного процесса в цепочке; в – тракты распространения сигналов в функциональном узле

### 6.3. Оценка качества функциональных схем

Для сравнения между собой различных вариантов схем, реализующих одну и ту же функцию, нужно уметь как-то оценивать их качество. При этом оценка должна, с одной стороны, возможно лучше отражать окончательные показатели качества того реального блока, который будет построен на основе данной схемы, а с другой, – выполняться только на базе той информации, которую несет функциональная схема, иначе выбор одной из конкурирующих схем на этапе логического проектирования будет невозможен.

Наиболее распространена оценка схемы по двум параметрам – задержке  $T$  и аппаратным затратам  $W$ , поскольку значения ряда других важных параметров цифрового блока – потребляемой мощности, частоты отказов, стоимости – при заданной элементной базе и априорно невысокой точности оценок в первом приближении допустимо считать пропорциональными аппаратным затратам. В дальнейшем методика оценки качества схем иллюстрируется на примере именно этой пары параметров –  $T$  и  $W$ . Если же проектирование блока специально ориентировано на достижение еще каких-то целей (уменьшение потребляемой мощности, повышение надежности, улучшение контролепригодности и т.п.), то можно вместо (или вместе с)  $T$  и  $W$  в процедуру оценки качества схемы включить любые актуальные для разработчика параметры. Каких-либо принципиальных изменений в излагаемом подходе это не вызовет.

При работе на микросхемах задержка  $T$  схемы достаточно объективно оценивается значением среднего времени задержки распространения  $t_{зд.р.ср}$  входящих в нее элементов. В рамках одной серии обычно целесообразно полагать, что задержки всех логических элементов россыпи – И-НЕ, И, И-ИЛИ-НЕ, М2 – одинаковы и равны некоторой усредненной для данной серии величине  $T$ . Это близко к истине, поскольку в микросхемах не только малой, но иногда и средней интеграции основная доля задержки приходится на мощный выходной каскад, перезаряжающий большую емкость печатного монтажа. Задержки внутренней, размещенной на кристалле части схемы намного меньше, поскольку там меньше размеры линий связи и, следовательно, их паразитные емкости. Для серий К155 и К555, например, значение  $T$  можно принять равным 20 нс. Задержку более сложных микросхем средней интеграции целесообразно округлять до значения, кратного целому  $T$  или его половине.

Аппаратурные затраты  $W$  функционального узла оценивают различными способами. Хорошие результаты дает оценка величины  $W$  площадью, занимаемой узлом на плате или кристалле. При использовании микросхем площадь платы при прочих равных условиях приблизительно пропорциональна числу корпусов. Размеры корпусов различны, поэтому их приходится приводить к какому-то единому, принятому за единицу. В качестве масштаба можно использовать отношение площадей корпусов или



чисел их выводов. Можно оценивать величину  $W$  схемы и непосредственно суммарным числом выводов всех корпусов.

Рассмотрим на примере использование вышеописанных методик.

На логических микросхемах серии К155 (см. табл. 6.1) построить несколько вариантов схем, реализующих заданную минимальную ДНФ (6.1). Сравнить полученные результаты.

$$Y_a = \overline{ac} \vee \overline{bc} \vee \overline{d}. \quad (6.1)$$

Схемная реализация этой формулы представлена на рис. 6.3, а. Поскольку на основании данных справочника задержки распространения элементов ЛРЗ и ЛН1 серии К155 одинаковы, задержка всей схемы равна  $3t$ . Аппаратурные затраты состоят из пяти инверторов ЛН1, каждый из которых занимает 1/6 корпуса, и элемента И-ИЛИ-НЕ-ЛРЗ, занимающего целый корпус. Итого для схемы по рис. 6.5, а

$$W_a = 5 \cdot 1/6 + 1 = 22/12 \text{ корпуса.}$$

Неиспользованные элементы частично занятого корпуса (в данном случае шестой инвертор) не учитываются, поскольку они могут быть использованы в других узлах. Подсчеты удобно производить в двенадцатых долях корпуса: 12 – это число логических выводов корпуса наименьшего размера. Оценка оборудования в двенадцатых долях корпуса численно близка к оценке оборудования по сумме входов и выходов всех элементов.

Поскольку у ЛРЗ часть входов осталась неиспользованной, можно попытаться реализовать  $Y$  с помощью элементов И-НЕ, набор которых имеет более мелкую градацию по числу входов. Применив к

$$Y_a = \overline{ac} \vee \overline{bc} \vee \overline{d}$$

правило де-Моргана, получим

$$Y_a = \overline{\overline{ac} \vee \overline{bc} \vee \overline{d}} = \overline{\overline{\overline{ac} \cdot \overline{bc} \cdot d}}. \quad (6.2)$$

Реализация (6.2) представлена на рис. 6.3, б. Параметры схемы:  $T_b = 3t$ ;  $W_b = 3 \cdot 1/6 + 2 \cdot 1/4 + 1 \cdot 1/3 = 16/12$  корпуса. Схема оказалась заметно экономичнее.

Можно попытаться использовать формулы де-Моргана для уменьшения числа инверторов на входе. Тогда

$$\begin{aligned} Y_b &= \overline{ac} \vee \overline{bc} \vee \overline{d} = (\overline{a \vee b})\overline{c} \vee \overline{d} = \\ &= \overline{\overline{\overline{a \vee b} \vee c} \vee \overline{d}} = \overline{ab \vee c} \vee \overline{d}. \end{aligned} \quad (6.3)$$

Схема, реализующая (6.3), показана на рис. 6.3, в. Для нее  $T_b = 2t$ ;  $W_b = 11/12$  корпуса. Удалось выиграть и во времени, и в оборудовании. Еще одно применение формулы де-Моргана дает

$$Y_r = \overline{ab \vee c} \vee \overline{d} = \overline{(ab \vee c)} \vee \overline{d}. \quad (6.4)$$

Схема (6.4) представлена на рис. 6.3, г:  $T_r = 3t$ ;  $W_r = 9/12$  корпуса. Схема оказалась очень экономичной, хотя и довольно медленной.

Если в (6.4) раскрыть скобки, то можно получить еще один вариант схемы (рис. 6.5, д):

$$\begin{aligned} Y_d &= \overline{abd} \vee cd; \\ T_d &= 1t; \\ W_d &= 12/12 \text{ корпуса.} \end{aligned} \quad (6.5)$$

Задержка этой схемы оказалась наименьшей из всех рассмотренных.

На рис. 6.3 показаны не все возможные схемы. Продолжая преобразования логического выражения, можно самостоятельно построить еще ряд схем и сравнить с уже полученными.

**Примечательно, что даже весьма простые логические выражения типа (6.1) допускают далеко не одну схемную интерпретацию.**

Примечательно также то, что хотя за основу для построения схем была взята минимальная ДНФ, ее схемная реализация оказалась, тем не менее, самой неэкономичной из всех. Противоречия здесь нет. Минимальная ДНФ минимальна лишь в определенном узком смысле: это выражение, которое обязательно принадлежит классу ДНФ и имеет минимальное суммарное число букв.

Переводя на язык аппаратуры, можно сказать: это соответствует минимуму суммы входов всех конъюнкторов, реализующих элементарные конъюнкции ДНФ в трехъярусной схеме НЕ-И-ИЛИ типа показанной на рис. 6.3, а. Затраты инверторов и дизъюнкторов этот критерий минимальности игнорирует. Базис россыпи серии К155 намного богаче булевского базиса НЕ, И, ИЛИ, оставаться в жестких рамках ДНФ никто от схемотехника не требует, а выйдя за эти рамки, возможно, удастся найти и другие, более компактные представления или декомпозиции функции.

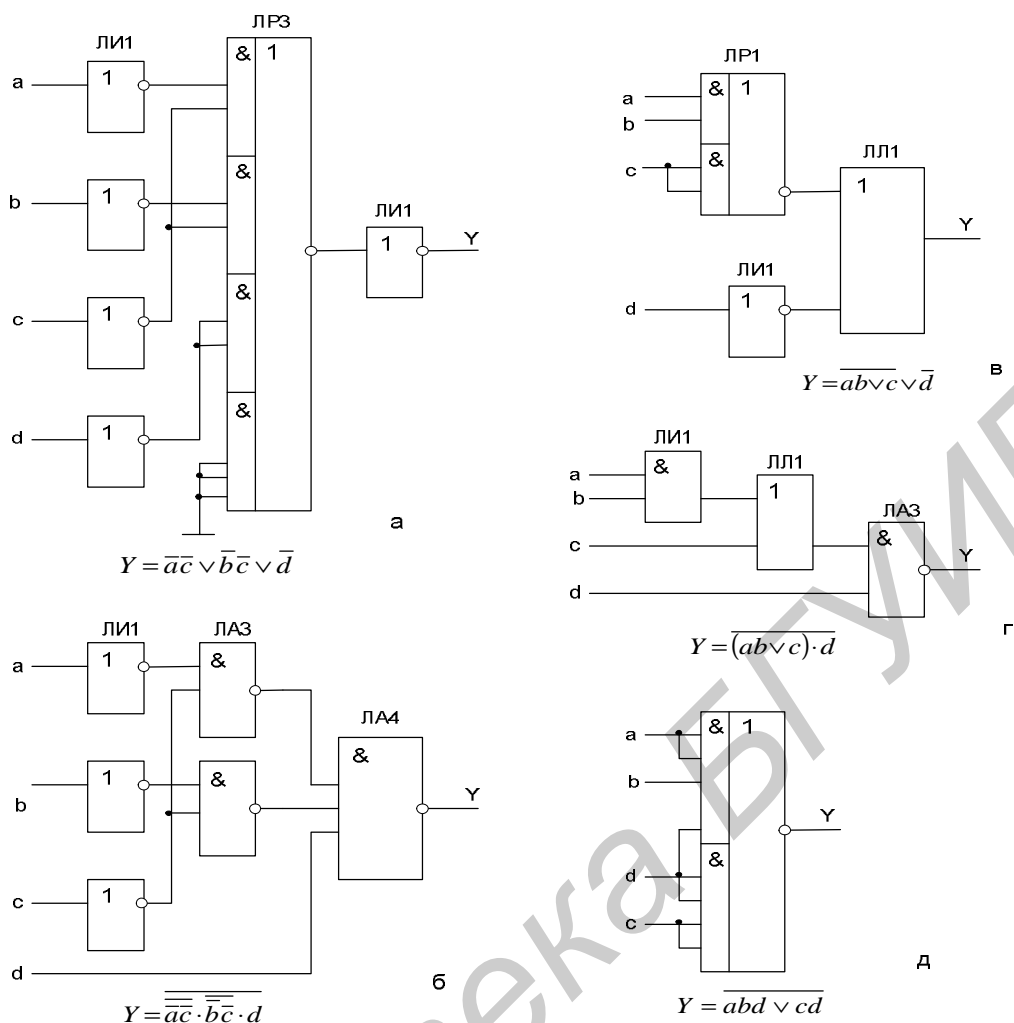


Рис. 6.3. Различные варианты схемной реализации выражения (6.1)

Поэтому практические оценки в единицах корпуса имеют не очень много общего с теоретическими оценками сложности, принятыми в булевой алгебре. Вид выражения, наиболее экономичного в каком-то техническом базисе, может существенно отличаться от вида минимальной ДНФ, и полученную в результате минимизации ДНФ схемотехник часто склонен рассматривать не как окончательный результат, а лишь как полуфабрикат, с которым можно еще поработать. Отсюда не следует, что минимизация не нужна вообще: чем компактнее выражение, тем легче обрабатывать его дальше.

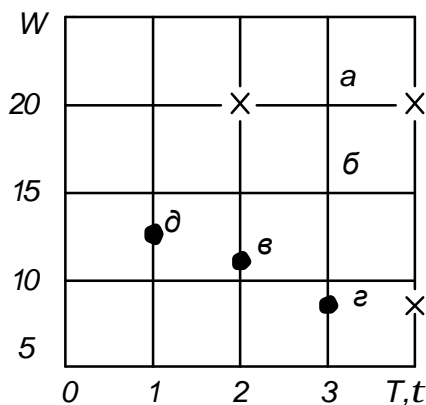


Рис. 6.4. Соотношение величин задержек  $T$  и аппаратурных затрат  $W$  схем, изображенных на рис. 6.3

На рис. 6.4 пять схем, которые были показаны на рис. 6.3, изображены в виде точек на плоскости  $TW$  ( $W$  дано в двенадцатых долях корпуса). Такой чертеж делает процесс сравнения различных вариантов существенно нагляднее. Нетрудно видеть, что схемы  $a$  и  $b$  (рис. 6.3) хуже, чем схема  $z$ , поскольку при той же задержке их аппаратурные затраты больше. Допустим, что преобразовав (6.1) еще несколькими способами, удалось получить схемы, помеченные на рис. 6.4 крестиками. Все эти схемы также оказываются плохими, поскольку для каждой из них существует хотя бы одна схема, которая лучше нее или по  $T$ , или по  $W$ , или по обоим показателям сразу. Этого нельзя сказать только о группе схем, образующей «лево-нижнюю» границу всего множества схем, т.е. о схемах  $d$ ,  $v$ ,  $z$ . При переходе от  $d$  к  $v$  и от  $v$  к  $z$  на каждом шаге получается выигрыш в аппаратуре, но проигрыш в скорости. Эти объекты лучше любых других объектов, расположенных выше них и правее.

## 7. Правила схемного включения элементов

Ограничение по нагрузочной способности элементов задается коэффициентом разветвления по выходу  $K_{раз}$ . Коэффициент равен максимальному числу входов той же серии, которые можно подключить к выходу данного элемента. При превышении нагрузки сверх допустимой значения выходных параметров элемента не гарантируются. Различные элементы различных серий имеют  $K_{раз} = 5-20$ , типовое значение – 10. Мощные элементы со специальным выходным каскадом имеют  $K_{раз} \geq 30$ . Часто кроме  $K_{раз}$  оговариваются максимально допустимый выходной ток и максимально допустимая емкость нагрузки, что необходимо знать при стыковке элементов различных серий.

Неиспользованные входы И в большинстве серий не должны оставаться ни к чему не подключенными (свободными). В ТТЛ- и ТТЛШ-сериях сигнал от свободного входа воспринимается элементом как логическая единица, и у элементов И-НЕ его можно было бы так и оставить свободным, однако возникающие при этом дополнительные заряды в базе замедляют переключение элемента по другим, работающим входам. В сериях ТТЛ и ТТЛШ неиспользованные И-входы либо объединяют с другими (см. рис. 6.3, д), если при этом не превышена допустимая нагрузка источника сигнала, либо подключают к источнику логической единицы. В качестве последнего используют или элемент И-НЕ, вход которого заземлен, или резистор с сопротивлением 1 кОм, подключенный к источнику питания +5 В. К одному такому резистору обычно разрешается подключать до 20 неиспользованных входов И.

В КМДП-элементах ни в коем случае не должно быть неподключенных входов. На них может оказаться наведенным любой потенциал, соответственно ложный потенциал окажется и на выходе. В сериях КМДП неиспользованные входы можно подключать к источнику питания непосредственно, без резистора. Как и в ТТЛ-сериях, неиспользованные входы можно объединять с рабочими.

Неиспользованные входы ИЛИ в любых сериях должны быть соединены с логическим нулем, т.е. с общим проводом. В неиспользуемой секции ИЛИ элемента И-ИЛИ-НЕ все или хотя бы один вход И должен быть подключен к общему проводу (см. рис. 6.3, а).

Если некоторые элементы, входящие в состав корпуса, не используются, то на входы неиспользуемых элементов ТТЛ-серий нужно подать такие сигналы, чтобы на их выходах была единица: в таком состоянии элемент потребляет меньше мощности и его можно использовать как источник логической единицы. Неиспользованные элементы КМДП-серий можно фиксировать в любом состоянии, не оставляя их незадействованными.

### 7.1. Элемент с открытым коллектором

Кроме логических элементов, имеющих обычный выходной каскад с двумя транзисторами, выпускаются элементы с *открытым (свободным) коллектором (open collector)*. Упрощенная его схема показана на рис. 7.1, а. Верхний транзистор обычной выходной пары отсутствует, и из корпуса выведен лишь коллектор нижнего транзистора  $V_T$ . Этот транзистор открывается логической частью элемента при совпадении на входе всех единиц. Если хотя бы один из входных сигналов равен 0, транзистор закрыт. Такой выходной каскад не способен сам по себе сформировать на выходе высокий уровень напряжения. Для этого к выводу корпуса *внешним монтажом* подключается резистор  $R$ , который в некоторой степени играет роль верхнего транзистора обычного выходного каскада. Этот резистор называют *подтягивающим (pull-up resistor)*. В таком виде элемент выполняет функцию И-НЕ. Выход с открытым коллектором помечают специальным значком – ромбиком, показанным на рис. 7.1, а. Одно из назначений элементов с открытым коллектором – быть переходным звеном от логической части устройства к элементам вывода информации. Вместо резистора  $R$  (или наряду с ним) в коллекторную цепь выходного транзистора могут быть включены светодиод, индикаторная лампочка, обмотка реле, коаксиальный кабель, вход усилителя мощности. Конкретные схемы подключения некоторых из этих элементов приведены в [6, 18, 9]. Выходной транзистор многих элементов с открытым коллектором имеет повышенный допустимый ток, а иногда и повышенное коллекторное напряжение. Примеры элементов с открытым коллектором: К155ЛА7; К155ЛАП. Открытый коллекторный выход иногда имеют и микросхемы более сложных функциональных узлов, например, дешифратор К155ИД1, многие микросхемы памяти.

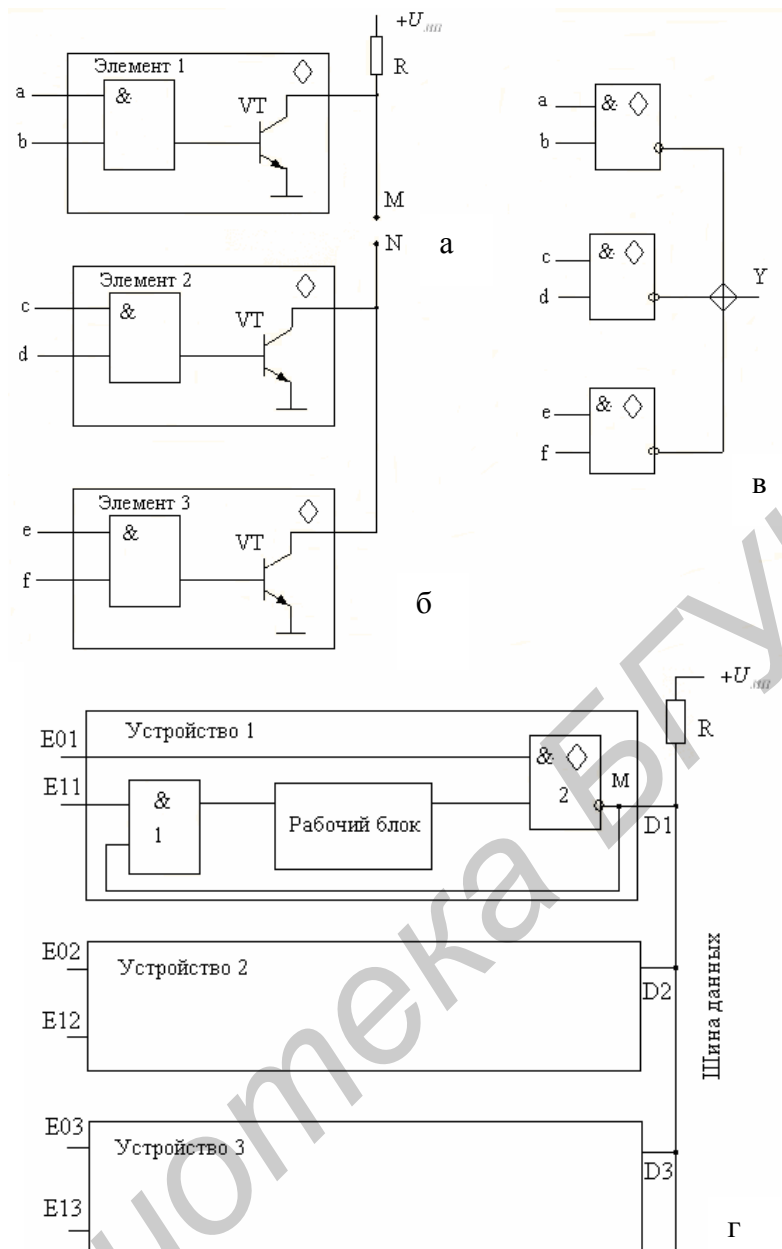


Рис. 7.1. Элементы с открытым коллектором: а – логический элемент;

б – организация вместе с элементом монтажного ИЛИ по рис. а;

в – обозначение монтажного ИЛИ на схемах; г – организация шины данных

Элементы с открытым коллектором в отличие от обычных логических элементов приспособлены для объединения по выходу. Если точку М на рис. 7.1, а, объединить с точкой N пары элементов на рис. 7.1, б, то уровень выхода будет равен 0, когда открыт выходной транзистор любого из объединенных элементов. Все три элемента совместно будут обрабатывать функцию И-ИЛИ-НЕ:

$$Y = \overline{ab \vee cd \vee ef}.$$

Такую схему называют **монтажное ИЛИ, проводное ИЛИ, wired OR**. Впрочем, на логику работы полученной схемы можно посмотреть и несколько иначе: на объединенном выходе будет 1 тогда и только тогда, когда каждый

элемент будет стремиться установить на своем выходе 1. Если же хотя бы один элемент установит на своем выходе 0, этот же 0 установится и на объединенном выходе. Такое рассуждение позволяет рассматривать объединение открытых коллекторов на общем подтягивающем резисторе как схему, реализующую функцию И: единица при всех единицах и нуль при любом нуле. Это дало повод для еще одного названия рассматриваемой схемы: **монтажное И**.

Схему монтажного ИЛИ при работе на россыпи в ТТЛ- и КМДП-сериях с чисто логическими целями используют не часто в силу худших ее частотных свойств по сравнению с элементами, имеющими обычный выходной каскад. Однако принцип монтажного ИЛИ широко используется в БИС памяти и ПЛМ, а также для построения двунаправленных **числовых шин (числовых магистралей – data bus)** (рис. 7.1, г).

## 7.2 Расширение логических возможностей элементов

При проектировании схем иногда возникает необходимость в логических элементах с числом входов или коэффициентом разветвления по выходу, превышающими паспортные значения реальных элементов. В этих случаях требуемые большие элементы строят из обычных.

### Увеличение нагрузочной способности

Требуется обеспечить разводку сигнала на число элементов, превышающее коэффициент разветвления по выходу  $K_{раз}$  элемента, служащего источником сигнала. Необходимо также дать решение как для небольшой кратности превышения, так и для превышения в десятки раз.

В качестве решения на рис. 7.2 показаны возможные варианты схем. Простое запараллеливание элементов (рис. 7.2, а) в ТТЛ-сериях допускается в виде исключения не более чем для двух элементов, при этом считается, что результирующий коэффициент разветвления по выходу увеличивается лишь в 1,9 раза. Оба элемента должны быть размещены в одном корпусе, в противном случае возникающие уравнивательные токи могут вызвать помехи в других цепях. В МДП-сериях ограничения на число запараллеливаемых элементов, как **правило**, не вводятся.

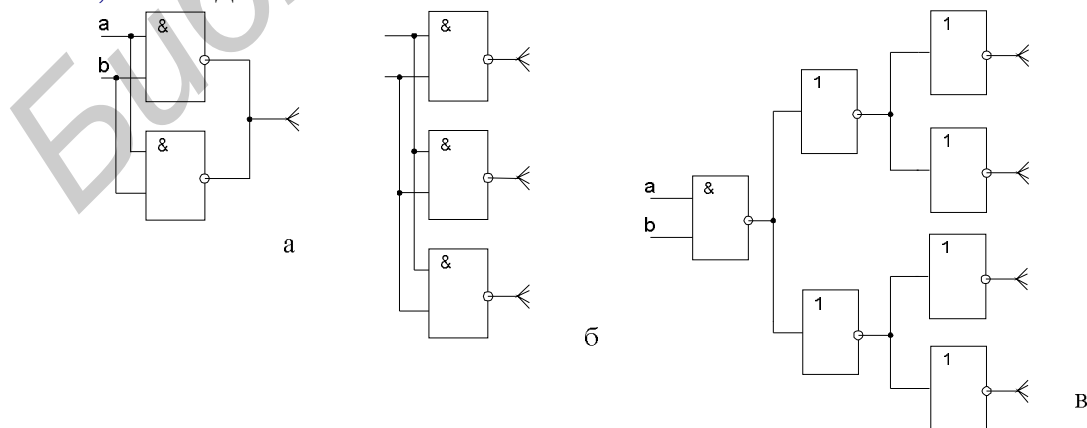


Рис. 7.2. Способы размножения сигнала

В схеме по рис. 7.2, б число буферных элементов ограничено допустимой нагрузкой источников сигналов  $a$  и  $b$ . Схема каскадного размножения по рис. 7.2, б используется при разводке по цифровому устройству цепей синхронизации.

### Построение многовходовых И

Требуется на элементах серии К155 построить И на 16 входов.

Подсказка. **Типовые** ошибки **начинающих** схемотехников: а) включение двухвходовых И *ЛИ1* в четыре каскада; б) **построение** шестнадцативходовой И из восьмивходовых И, **которые, в свою очередь, строятся** из восьмивходовых И-НЕ *ЛА7* с подключенным на выходе инвертором *ЛН1*. Эти решения расточительны и по количеству оборудования, и по быстродействию.

Лучшее решение дает разбиение функции с помощью формулы де-Моргана на такие группы, размер которых соответствует числу входов имеющихся элементов:

$$\begin{aligned}
 Y &= x_0x_1x_2\dots x_{14}x_{15} = \\
 &= \overline{\overline{x_0x_1x_2x_3x_4x_5x_6x_7} \vee \overline{x_8x_9x_{10}x_{11}x_{12}x_{13}x_{14}x_{15}}} = \\
 &= \overline{x_0x_1x_2x_3} \vee \overline{x_4x_5x_6x_7} \vee \overline{x_8x_9x_{10}x_{11}} \vee \overline{x_{12}x_{13}x_{14}x_{15}}.
 \end{aligned}
 \tag{7.1}$$

Схема, реализующая среднюю строку (7.1), показана на рис. 7.3, а. Если используется серия, в которой отсутствуют элементы ИЛИ-НЕ, их можно заменить элементами И-ИЛИ-НЕ, как показано на рис. 7.3, б. Для серий, в которых отсутствуют восьмивходовые И-НЕ, массив из 16 входов нужно разбить на четырехвходовые части [нижняя строка (7.1)]. Схема для этого случая показана на рис. 7.3, в. Ее параметры:  $T = 2t$ ,  $W \approx 2,5$  корпуса. Варианты по рис. 7.3, б и в требуют несколько больших аппаратных затрат, чем схема по рис. 7.3, а. Материал данного примера иллюстрирует еще одно эвристическое правило: по возможности выбирать разбиение, использующее больше многовходовых элементов. Это уменьшает общую сумму выводов корпусов, а следовательно, и количество использованного оборудования.

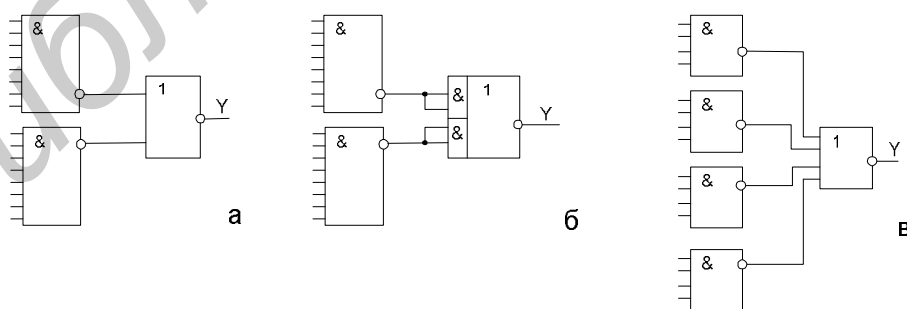


Рис. 7.3. Варианты построения многовходового И

Многовходовые И используются, например, для обнаружения нулевого значения содержимого регистра. При этом на схему И подаются инверсные выходы всех триггеров регистра.



### Построение многовходовых ИЛИ

Требуется построить схему ИЛИ на 16 входов на элементах серии К155.

Решение. После рассмотрения предыдущего примера задача становится типовой:

$$Y = x_0 \vee x_1 \vee x_2 \dots x_{14} \vee x_{15} =$$

---

$$= x_0 \vee x_1 \vee x_2 \vee x_3 \cdot x_4 \vee x_5 \vee x_6 \vee x_7 \cdot x_8 \vee x_9 \vee x_{10} \vee x_{11} \cdot x_{12} \vee x_{13} \vee x_{14} \vee x_{15}. \quad (7.2)$$

Схема, реализующая (7.2), по начертанию аналогична схеме, показанной на рис. 7.3, а, но вместо элементов И-НЕ стоят ИЛИ-НЕ и наоборот. Как и в варианте на рис. 7.3, б, роль элементов ИЛИ-НЕ могут играть элементы И-ИЛИ-НЕ.

Многовходовое ИЛИ используется, например, когда нужно обнаружить присутствие хотя бы одной единицы в регистре.

Библиотека БГУИР

## Литература

1. Точки, Р. Д. Цифровые системы : теория и практика / Р. Д. Точки, Н. С. Уидмер. – 8-е изд. – М. : Вильямс, 2004. – 1024 с.
2. Схемотехника электронных систем. Цифровые устройства / В. И. Бойко [и др.]. – СПб. : БХВ-Петербург, 2004. – 512 с.
3. Угрюмов, Е. П. Цифровая схемотехника : учеб. пособие / Е. П. Угрюмов. – 2-е изд., перераб. и доп. – СПб. : БХВ-Петербург, 2004. – 782 с.
4. Опадчий, Ю. Ф. Аналоговая и цифровая электроника / Ю. Ф. Опадчий, О. П. Глудкин, А. И. Гуров. – М. : Горячая линия–Телеком, 2005. – 767 с.
5. Пухальский, Г. И. Цифровые устройства : учеб. пособие / Г. И. Пухальский, Т. Я. Новосельцева. – СПб. : Политехника, 1996. – 885 с.
6. Справочник по интегральным микросхемам / Б. В. Тарабрин [и др.]; под ред. Б. В. Тарабрина. – М. : Энергия, 1980. – 816 с.
7. Интегральные микросхемы : справочник / Б. В. Тарабрин [и др.] ; под ред. Б. В. Тарабрина. – М. : Энергоатомиздат, 1985. – 528 с.
8. Аналоговые и цифровые интегральные микросхемы : справочное пособие / С. В. Якубовский [и др.]; под ред. С. В. Якубовского. – М. : Радио и связь, 1985. – 432 с.
9. Ланцов, А. Л. Цифровые устройства на комплементарных МДП интегральных микросхемах / А. Л. Ланцов, Л. Н. Зворыкин, И. Ф. Осипов. – М. : Радио и связь, 1983. – 272 с.
10. Кармазинский, А. Н. Синтез принципиальных схем цифровых элементов на МДП-транзисторах / А. Н. Кармазинский. – М. : Радио и связь, 1983.
11. Основы построения технических средств ЕС ЭВМ на интегральных микросхемах / В. В. Саморуков [и др.]; под ред. Б. Н. Файзулаева. – М. : Радио и связь, 1981.
12. Зельдин, Е. А. Цифровые интегральные микросхемы в информационно-измерительной аппаратуре / Е. А. Зельдин. – Л. : Энергоатомиздат. Ленинградское отделение, 1986.
13. Угрюмов, Е. П. Проектирование узлов и элементов ЭВМ / Е. П. Угрюмов. – М. : Высшая школа, 1987.
14. Шило, В. Л. Популярные цифровые микросхемы : справочник / В. Л. Шило. – 2-е изд., исправл. – М. : Радио и связь. 1989. – 352 с.

Учебное издание

**Лыньков** Леонид Михайлович  
**Ширинский** Валерий Павлович

**РАЗРАБОТКА ЛОГИЧЕСКОЙ СХЕМЫ  
ЦИФРОВОГО УСТРОЙСТВА**

Методическое пособие  
к курсовому проектированию  
по дисциплине «Цифровые и микропроцессорные устройства  
средств измерений»  
для студентов специальности  
«Метрология, стандартизация и сертификация»  
дневной формы обучения

Редактор Т. Н. Крюкова  
Корректор М. В. Тезина

---

Подписано в печать 06.08.2007.  
Гарнитура «Таймс».  
Уч.-изд. л. 2,8.

Формат 60x84 1/16.  
Печать ризографическая.  
Тираж 100 экз.

Бумага офсетная.  
Усл. печ. л. 3,02.  
Заказ 237.

---

Издатель и полиграфическое исполнение: Учреждение образования  
«Белорусский государственный университет информатики и радиоэлектроники»  
ЛИ №02330/0056964 от 01.04.2004. ЛП №02330/0131666 от 30.04.2004.  
220013, Минск, П. Бровки, 6