

## БАЗОВЫЕ ЭЛЕМЕНТЫ ЭВМ

### 3.1. ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ

**Логические элементы** — это электронные схемы с одним или несколькими входами и одним выходом, через которые проходят электрические сигналы, представляющие 0, 1.

Для характеристики логических элементов и микросхем, построенных на их основе, используются временные диаграммы и определенные параметры.

Далее приведены некоторые параметры микросхем, (ГОСТ 19480—89 «Микросхемы интегральные. Термины, определения и буквенные обозначения электрических параметров» и ГОСТ 2.743—91 «ЕСКД. Обозначения условные графические в схемах. Элементы цифровой техники»), имеющие следующие обозначения:

$L$  — напряжение низкого уровня  $U^0$ , которое интерпретируется как логический «0».

$H$  — напряжение высокого уровня  $U^1$ , которое интерпретируется как логическая «1».

Любое напряжение, значение которого, находится между  $L$  и  $H$ , называется уровнем нестабильности и выражается в относительных единицах или в процентах.

$t_{3A, P}^{0,1} (t_{PHL})$  — время задержки распространения сигнала при включении. Интервал времени, измеренный на заданном уровне, между входным и выходным импульсами, в течение которого выходной сигнал переходит от  $H$  к  $L$  уровню.

$t_{3A, P}^{1,0} (t_{PLH})$  — время задержки распространения сигнала при включении. Интервал времени, измеренный на заданном уровне, между входным и выходным импульсами, в течение которого выходной сигнал переходит от  $L$  к  $H$  уровню

$t_{3A, P. CP} (t_{PAV})$  — среднее время задержки распространения

$$(t_{3A, P. CP} = \frac{t_{3A, P}^{1,0} + t_{3A, P}^{0,1}}{2}).$$

$t^{1,0}(t_{THL})$  — время перехода при выключении. Время, в течение которого выходное напряжение переходит от  $H$  к  $L$  уровню, измеренное на заданном уровне.

$t^{0,1}(t_{TLH})$  — время перехода при включении. Время, в течение которого выходное напряжение переходит от  $L$  к  $H$  уровню, измеренное на заданном уровне.

$U_{\text{пор}}^1$  — пороговое напряжение высокого уровня.

$U_{\text{пор}}^0$  — пороговое напряжение низкого уровня.

Под пороговым напряжением понимают наименьшее ( $U_{\text{пор}}^1$ ) или наибольшее ( $U_{\text{пор}}^0$ ) значение соответствующих уровней, при котором начинается переход логического элемента в другое состояние. Эти параметры определяются с учетом разброса параметров соответствующей серии микросхем в рабочем диапазоне температур; в справочниках часто приводится одно усредненное значение  $U_{\text{пор}}$ .

Процесс изменения напряжения от низкого уровня  $L$  к высокому  $H$ , называется фронтом сигнала (положительным перепадом, положительным фронтом), а обратный процесс — спадом (отрицательным перепадом, отрицательным фронтом). Если существенно их взаимное расположение, то фронт может быть передним и задним.

**Схема элемента И.** Схема элемента И реализует конъюнкцию (логическое умножение) двух или более логических значений. Обозначается НИ, где  $N$  — количество входов (2И, 3И и т. д.).

Единица на выходе схемы НИ будет тогда и только тогда, когда на всех  $N$ -входах будут единицы. Когда хотя бы на одном входе будет ноль, на выходе также будет ноль.

Таблица истинности для логической функции 2И (рис. 3.1) приведена в табл. 3.1.

**Схема элемента ИЛИ.** Схема элемента ИЛИ реализует дизъюнкцию (логическое сложение) двух или более логических значений. Когда хотя бы на одном входе схемы ИЛИ будет единица, на ее выходе также будет единица. Обозначается НИЛИ, где  $N$  — количество входов (2ИЛИ, 3ИЛИ и т. д.).

Таблица 3.1

X1	X2	$Y = X1 \& X2$
0	0	0
0	1	0
1	0	0
1	1	1

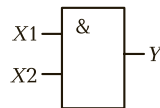


Рис. 3.1. Условное обозначение элемента 2И

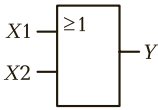


Рис. 3.2. Условное обозначение элемента 2ИЛИ

Таблица 3.2		
X1	X2	$Y = X1 X2$
0	0	0
0	1	1
1	0	1
1	1	1

Таблица истинности для логической функции 2ИЛИ (рис. 3.2) показана в табл. 3.2.

**Схема элемента НЕ.** Схема элемента НЕ (инвертор) реализует операцию отрицание двух или более логических значений.

Рассмотрим реакцию инвертора на изменение входного сигнала (рис. 3.3). Инерционные свойства инвертора приводят к задержке сигнала при его прохождении от входа к выходу.

Процесс изменения напряжения от низкого уровня  $L$  к высокому  $H$ , называется фронтом сигнала (положительным перепадом, положительным фронтом), а обратный процесс — спадом (отрицательным перепадом, отрицательным фронтом). Если существенно их взаимное расположение, то фронт может быть передним и задним.

Таблица истинности для логической функции НЕ показана в табл. 3.3.

В чистом виде логические элементы И, ИЛИ, НЕ используются очень редко. Наиболее часто в реализации логических элементов применяются комбинированные логические элементы, выполняющие две функции.

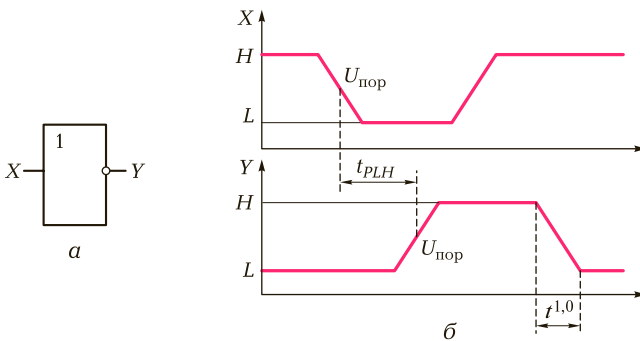


Рис. 3.3. Условное обозначение (а) и временная диаграмма (б) элемента НЕ

Для реализации сколь угодно сложных функций удобно использовать элементы, обладающие логической полнотой (способность функции выражать все логические функции). Это функции И—НЕ и ИЛИ—НЕ.

**Таблица 3.3**

$X$	$Y = \neg X$
0	1
1	0

**Схема элемента И—НЕ.** Схема элемента И—НЕ (операция Шеффера) состоит из элементов И и НЕ. Она осуществляет отрицание результата схемы И. Связь между выходом  $Y$  и входами  $X_1$  и  $X_2$  схемы записывают следующим образом:  $Y = \neg(X_1 \& X_2)$  или  $Y = \overline{X_1 \& X_2}$ .

Таблица истинности для логической функции 2И—НЕ (рис. 3.4) показана в табл. 3.4.

**Схема элемента ИЛИ—НЕ.** Схема элемента ИЛИ—НЕ (стрелка Пирса) состоит из элементов ИЛИ и НЕ. Она осуществляет отрицание результата схемы ИЛИ. Связь между выходом  $Y$  и входами  $X_1$  и  $X_2$  схемы записывают следующим образом:  $Y = \neg(X_1 | X_2)$  или  $Y = \overline{X_1 | X_2}$ .

Таблица истинности для логической функции 2ИЛИ—НЕ (рис. 3.5) показана в табл. 3.5.

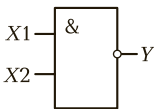


Рис. 3.4. Условное обозначение элемента 2И—НЕ

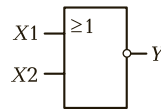


Рис. 3.5. Условное обозначение элемента 2ИЛИ—НЕ

**Таблица 3.4**

$X_1$	$X_2$	$Y = X_1 \& X_2$	$Y = \neg(X_1 \& X_2)$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

**Таблица 3.5**

$X_1$	$X_2$	$Y = X_1   X_2$	$Y = \neg(X_1   X_2)$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

## 3.2. БАЗОВЫЕ СХЕМЫ

Из логических элементов путем их комбинации строятся основные схемы компьютера:

- триггер — электронный прибор, имеющий два устойчивых состояния (0 и 1), является типичным запоминающим элементом;
- регистр — совокупность триггеров, предназначенных для хранения числа в двоичном коде;
- сумматор — устройство, обеспечивающее суммирование двоичных чисел с учетом переноса из предыдущего разряда.

Эти схемы входят в состав разнообразных цифровых устройств.

Все цифровые устройства строятся из логических микросхем, каждая из которых обязательно имеет следующие выводы (ножки):

- выводы питания: общий (или «земля») и напряжения питания (в большинстве случаев — +5 В или +3,3 В), которые на схемах обычно не показываются;
- выводы для входных сигналов (или «входы»), на которые поступают внешние цифровые сигналы;
- выводы для выходных сигналов (или «выходы»), на которые выдаются цифровые сигналы из самой микросхемы.

Каждая микросхема преобразует тем или иным способом последовательность входных сигналов в последовательность выходных сигналов. Способ преобразования чаще всего описывается или в виде таблицы (таблицы истинности), или в виде временных диаграмм, т. е. графиков зависимости от времени всех сигналов.

Все цифровые микросхемы работают с логическими сигналами, имеющими два разрешенных уровня напряжения. Один из этих уровней называется уровнем логической единицы (единичный уровень), а другой — уровнем логического нуля (нулевой уровень). Чаще всего логическому нулю соответствует низкий уровень напряжения, а логической единице — высокий уровень. Однако при передаче сигналов на большие расстояния и в системных шинах микропроцессорных систем порой используют и обратное представление, когда логическому нулю соответствует высокий уровень напряжения, а логической единице — низкий уровень.

Иногда логический нуль кодируется положительным уровнем напряжения (тока), а логическая единица — отрицательным уровнем

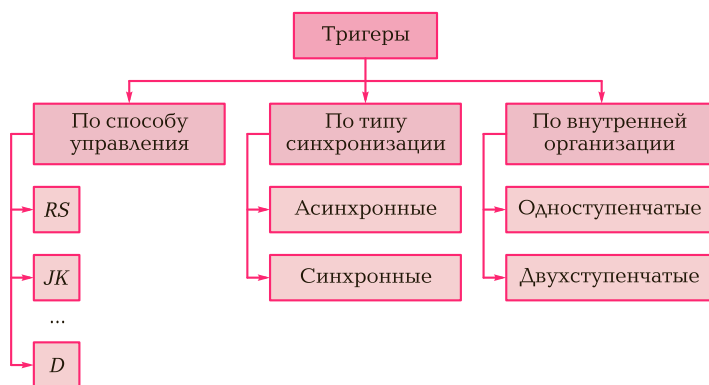


Рис. 3.6. Общая классификация триггеров

напряжения (тока), или наоборот. Есть и более сложные методы кодирования логических нулей и единиц.

### 3.3. ТРИГГЕРЫ

Устройство, в любой момент времени находящееся только в одном из двух устойчивых состояний (0 либо 1), называют **триггером**. Он строится на рассмотренных ранее логических элементах. Триггеры можно классифицировать по различным признакам (рис. 3.6).

По типу синхронизации (способу записи информации) триггеры подразделяются на асинхронные и синхронные. В асинхронных триггерах состояние на выходе изменяется сразу же после изменения сигнала на информационных входах. В синхронных триггерах для передачи сигнала с информационных входов на выходы требуется специальный синхронизирующий импульс на специальный вход *C* (см. рис. 3.10).

По способу управления синхронные триггеры подразделяются на триггеры со статическим управлением и триггеры с динамическим управлением. В триггерах с динамическим управлением передача сигнала с информационных входов на выходы осуществляется по фронту или по спаду синхронизирующего импульса.

По внутренней организации триггеры со статическим управлением подразделяются на одноступенчатые и двухступенчатые триггеры. Запись информации в одноступенчатые триггеры пред-

ставляет собой непрерывный во времени процесс установления состояния триггера под воздействием информационных сигналов. Запись информации в двухступенчатые триггеры осуществляется в два этапа: сначала информация записывается в первую ступень, а затем переписывается во вторую и появляется на выходе триггера.

Схема симметричного триггера была разработана в 1918 г. советским ученым М. А. Бонч-Бруевичем. В 1919 г. аналогичная схема была разработана также американскими учеными Икклзом и Джорданом.

В основе любого триггера лежит схема из двух логических элементов, которые охвачены положительными обратными связями (сигналы с выходов подаются на входы). В результате подобного включения схема может находиться в одном из двух устойчивых состояний, причем находиться сколь угодно долго, пока на нее подано напряжение питания.

Если мы хотим хранить какую-либо двоичную последовательность цифр, то для каждой цифры следует использовать отдельный триггер. Для того чтобы получить данные о состоянии триггера, у него должна быть, по крайней мере, одна выходная линия, представляющая логическое значение, которое соответствует состоянию триггера.

Когда на выходной линии логическая единица, говорят, что триггер установлен, в противном случае говорят, что триггер сброшен.

Триггер имеет несколько входных линий. Последующее состояние триггера определяется текущим состоянием триггера и сигналами на входных линиях.

Триггер обозначается прямоугольником, разделенным на основное и вспомогательное поля. В первом помещают букву *T* (триггер), во втором — условные обозначения (метки) входов и логических операций на входе. Метки имеют следующие обозначения:

- *S* (Set-установка) — вход триггера для установки в состояние «1»;
- *R* (Reset-сброс) — вход триггера для установки в состояние «0»;
- *J* (Jump) — вход триггера для установки в состояние «1» в *JK*-триггере;
- *K* (Kill) — вход триггера для установки в состояние «0» в *JK*-триггере;
- *D* (Data input) — информационный вход. На него подается информация, предназначенная для записи в триггер;

- $T$  (Toggle-переключатель) — счетный вход (если у триггера только счетный вход, букву не пишут);
- $C$  (Clock input) — вход синхронизации.

Рассмотрим простейшую схему, позволяющую запоминать двоичную информацию на основе простейших логических элементов И—НЕ (рис. 3.7) или ИЛИ—НЕ (рис. 3.8).

На примере рис. 3.7 мы видим два выхода:

- $Q$  — прямой выход;
- $\bar{Q}$  — инверсный выход.

Потенциалы на них взаимно инвертированы: логическая единица на одном выходе соответствует логическому нулю на другом. С приходом переключающих (запускающих) сигналов переход триггера из одного состояния в другое происходит лавинообразно, и потенциалы на выходах меняются на противоположные.

В интервале между переключающими сигналами состояние триггера не меняется, т. е. триггер «запоминает» поступление сигналов, отражая это величиной потенциала на выходе. Это дает возможность использовать его как элемент памяти.

У схемы есть два инверсных входа:

- вход  $S$  позволяет устанавливать выход триггера  $Q$  в единичное состояние при подаче на его вход логического нуля;
- вход  $R$  позволяет сбрасывать выход триггера  $Q$  в нулевое состояние при подаче на его вход логического нуля.

Эта схема является структурной схемой  $RS$ -триггера.

Точно так же можно построить  $RS$ -триггер и на логических элементах ИЛИ—НЕ. Схема  $RS$ -триггера, построенного на логических элементах ИЛИ—НЕ, приведена на рис. 3.8. Единственное отличие заключается в том, что сброс и установка триггера будут производиться единичными логическими уровнями.

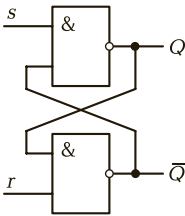


Рис. 3.7. Схема простейшего триггера на схемах 2И—НЕ

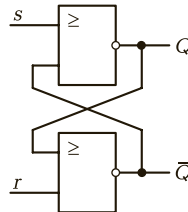


Рис. 3.8. Схема простейшего триггера на схемах 2ИЛИ—НЕ



В стандартные серии цифровых микросхем входит несколько типов микросхем триггеров, различающихся методами управления, а также входными и выходными сигналами. На схемах триггеры обозначаются буквой *T*.

В отечественных сериях микросхем триггеры имеют наименование ТВ, ТМ и ТР в зависимости от типа триггера. Наиболее распространены три типа:

- *RS*-триггер (обозначается ТР) — самый простой триггер, но редко используемый;
- *JK*-триггер (обозначается ТВ) имеет самое сложное управление, также используется довольно редко;
- *D*-триггер (обозначается ТМ) — наиболее распространенный тип триггера.

Простейшим из них является *асинхронный RS-триггер* (рис. 3.9).

*RS*-триггер имеет два входа и два выхода. Входы и выходы триггера имеют свои обозначения. Один из входов триггера называется установочным входом и обозначается буквой *S*, а другой — входом сброса и обозначается буквой *R*. Триггер имеет два симметричных выхода. Один выход — прямой (без отрицания) (выход *Q*), а другой — инверсный (с отрицанием) (выход  $\bar{Q}$ ).

*RS*-триггер может иметь прямые и инверсные входы (рис. 3.7—3.9).

Рассмотрим работу *RS*-триггера, используя временную диаграмму (рис. 3.9). Временная диаграмма может строиться без указания  $t^{0,1}$  и  $t^{1,0}$  (времени перехода от состояния *L* к состоянию *H* и наоборот)

Через время  $t_{3A,P}$  ( $t_2 - t_1$ ) от поступления сигнала  $\bar{S} = 0$ , выход *Q* переключится первым, а следом, через такой же промежуток времени ( $t_3 - t_2 = t_2 - t_1$ ), переключится и выход  $\bar{Q}$ . Спустя интервал времени  $t_3 - t_1 = \Delta t$  на выходах установятся новые значения. Так-

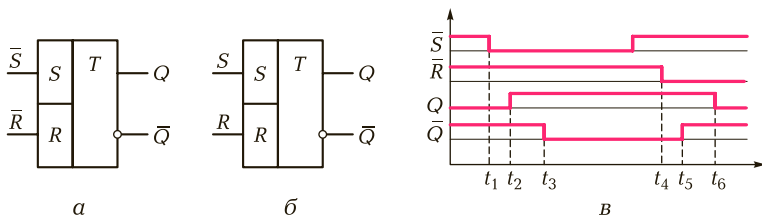


Рис. 3.9. Графическое обозначение асинхронного *RS*-триггера (а — с инверсными входами; б — с прямыми входами) и временная диаграмма (в)

же протекает процесс при  $\bar{R} = 0$ , но при этом, переключится первым выход  $\bar{Q}$ . Отсюда вытекает, что изменение входных сигналов не должно происходить быстрее времени  $\Delta t$ .

Несмотря на большое разнообразие триггеров, практически все триггеры строятся на базе  $RS$ -триггеров.

Важным методом, используемым для описания функционирования триггеров, является метод таблиц состояний (таблиц переходов). Таблица состояний  $RS$ -триггера в сокращенной форме (эту таблицу называют также управляющей таблицей, таблицей функционирования) содержит два входных сигнала (сигналы  $R$  и  $S$ ) и один выходной сигнал  $Q$  (функция). Хотя триггеры имеют два выхода — один прямой  $Q$ , а другой — инверсный  $\bar{Q}$ , в описании триггера и в таблице состояний указывают лишь состояние прямого выхода  $Q$ .

Сокращенная таблица состояний триггера отражает лишь динамику изменения состояния триггера и не учитывает свойство триггера запоминать единицу информации. Полная таблица состояний триггера должна учитывать влияние (на процесс управления) значения предыдущего состояния триггера. Причем предыдущее состояние представляется как входная переменная.

Рассмотрим сокращенную таблицу состояний  $RS$ -триггера.

Сокращенная таблица состояний  $RS$ -триггера показана в табл. 3.6.

Из сокращенной таблицы состояний триггера видно, что при подаче на вход  $R$  логической 1 триггер принимает состояние логического 0, а при подаче логической 1 на вход  $S$  — состояние логического 0. Следует отметить также, что если до подачи управляющего сигнала, например на вход  $R$ , триггер находился в состоянии логического 0, его состояние не изменится и после подачи логического 1 на вход  $R$ . Если на обоих входах триггера имеются уровни логического 0, это состояние соответствует режиму хранения и триггер сохраняет предыдущее состояние. При подаче на входы  $R$  и  $S$  одновременно логической 1 триггер будет находиться в неопределенном (или неправильном) состоянии, поэтому такое сочетание

**Таблица 3.6**

$S$	$R$	$Q$
0	0	Режим хранения информации без изменения
0	1	0
1	0	1
1	1	Режим неопределенности

Таблица 3.7

№ п/п	S	R	Состояние RS-триггера	
			до поступления сигнала	после поступления сигнала
1	0	0	0	0
2	0	0	1	1
3	0	1	0	0
4	0	1	1	0
5	1	0	0	1
6	1	0	1	1
7	1	1	0	Режим неопределенности
8	1	1	1	Режим неопределенности

сигналов  $R$  и  $S$  называется запрещенной комбинацией управляющих сигналов.

Теперь рассмотрим полную таблицу состояний RS-триггера, которая приведена в табл. 3.7. Таблицу состояний строят так же, как и таблицу истинности.

Анализ табл. 3.7 показывает, что только в ситуациях, описываемых строками 4 и 5, происходит изменение состояния триггера.

Строки 1 и 2: сигналы  $S = 0$  и  $R = 0$  и, следовательно, никаких изменений в состоянии триггера не происходит.

Строка 3: сигнал  $R = 1$ , и этот сигнал в нормальных условиях должен сбросить триггер, но так как триггер уже «сброшен» и  $Q = 0$ , то сигнал  $R = 1$  не изменяет его состояние.

Строка 4: после того, как подается сигнал на вход  $R$ , триггер сбрасывается, т. е. переходит из состояния «1» в состояние «0».

Строка 5: триггер устанавливается, т. е. переходит из состояния «0» в состояние «1», в результате подачи сигнала «1» на вход  $S$ .

Строка 6: сигнал  $S = 1$ , и этот сигнал в обычных условиях будет устанавливать триггер в «1», но  $Q = 1$  и, следовательно, состояние триггера останется без изменений до поступления следующего сигнала  $R$ .

Строки 7 и 8: при подаче одновременно на входы  $R$  и  $S$  сигнала, соответствующего логической 1, состояние триггера становится неопределенным: на обоих выходах  $Q$  и  $\bar{Q}$  установится уровень «1», а после снятия со входов управляющих сигналов, в силу случайных причин, триггер может установиться в состояние «0» либо «1». Очевидно, что для нормальной работы триггера необходимо исключить указанное сочетание входных сигналов, приводящее к неопределенному состоянию.

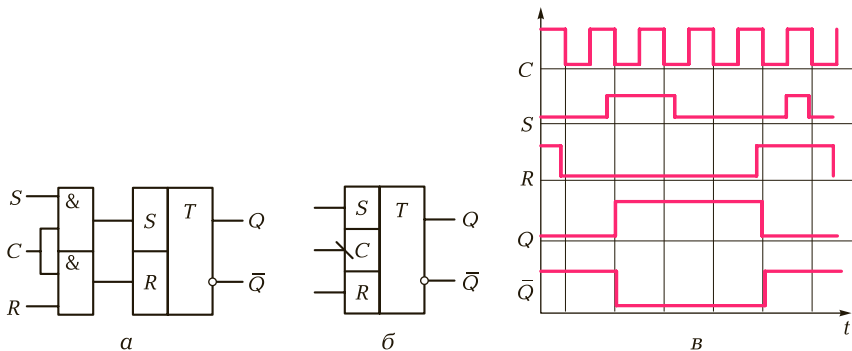


Рис. 3.10. Структурная схема (а), графическое обозначение (б) и временная диаграмма (в) синхронного  $RS$ -триггера

Мы рассмотрели асинхронный  $RS$ -триггер. Кроме него существует *синхронный  $RS$ -триггер* (рис. 3.10). Синхронные триггеры снабжаются дополнительным входом, по которому поступает синхронизирующий (тактирующий) сигнал. При этом изменение состояния триггера происходит (при наличии управляющего сигнала) только в те моменты времени, когда на специальный синхровход триггера поступает тактирующий импульс. Синхронизирующий вход обозначается буквой  $C$ .

Входные сигналы, по которым происходит переключение триггера:  $S$  и  $R$  — информационные и  $C$  — синхронизирующие.

Все триггеры с синхронизирующим входом, которые управляются уровнем синхронизирующего сигнала, являются *триггерами со статическим управлением*.

Эти триггеры могут изменять свое состояние не только при поступлении тактирующего импульса, но и в процессе действия, что создает неопределенность в работе логических устройств, приводящую к ошибкам в управлении.

Для снижения вероятности ошибок применяют *триггеры с динамическим управлением*, переключаящим сигналом для которых является не наличие высокого ( $H$ ) или низкого ( $L$ ) уровня разрешающего сигнала, а переход этого сигнала с низкого на высокий или с высокого на низкий логический уровень.

В первом случае говорят, что переключение происходит по фронту (переднему фронту) (рис. 3.11) синхроимпульса, а вход называют прямым динамическим входом. Во втором случае говорят, что переключение происходит по срезу (заднему фронту) синхроимпульса, а вход тактирующего сигнала называют инверсным динамическим входом.

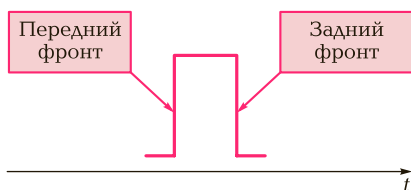


Рис. 3.11. Такт синхронизирующего сигнала

Следует отметить, что для надежной работы триггера необходимо, чтобы длительность переключающего сигнала (синхронизирующего сигнала) на входе  $C$  была не меньше времени переключения триггера. Временем переключения (срабатывания, установки) триггера называется время от момента изменения входных сигналов до соответствующего изменения состояния выходов. Это время определяется задержками распространения сигнала логическими элементами, входящими в состав триггера.

Рассмотренные схемы  $RS$ -триггеров являются одноступенчатыми. Применение одноступенчатых  $RS$ -триггеров в качестве самостоятельных запоминающих элементов ограничено.

Большой востребованностью пользуются *двухступенчатые триггеры (MS-триггеры)* (рис. 3.12).

Для построения двухступенчатых триггеров используются одноступенчатые  $RS$ -триггеры.

Двухступенчатый триггер состоит из двух каскадных секций (ступеней), причем каждая секция содержит по синхронному  $RS$ -триггеру. Первая секция, ведущая или  $M$ -секция (Master) принимает информацию с входных линий  $S$  и  $R$ . Состояние выходов ведущей секции подается на вторую секцию, ведомую, или  $S$ -секцию (от англ. Slave).

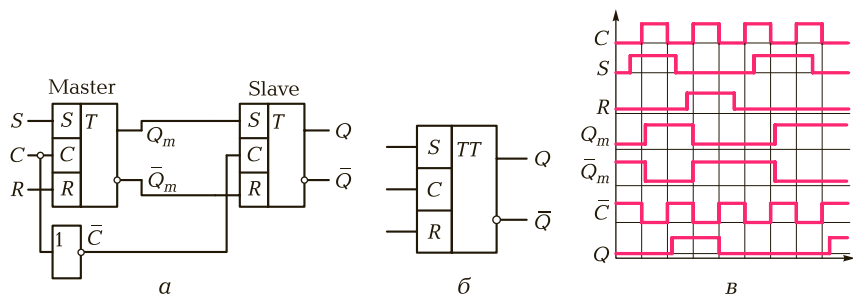


Рис. 3.12. Структурная схема [а], графическое обозначение [б] и временная диаграмма [в]  $MS$ -триггера

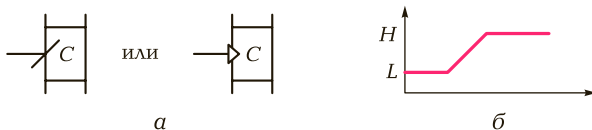


Рис. 3.13. Синхронизация передним фронтом:  
*а* — условные обозначения; *б* — временная диаграмма

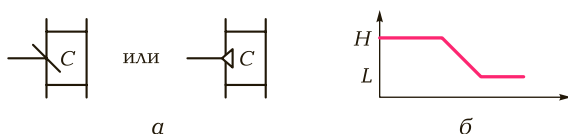


Рис. 3.14. Синхронизация задним фронтом:  
*а* — условные обозначения; *б* — временная диаграмма

Если в обозначении синхронного *RS*-триггера с динамическим управлением стрелка на входе *C* направлена к триггеру, то передача сигналов с информационных входов на выходы происходит по фронту импульса, а если стрелка направлена от обозначения триггера, то передача сигнала осуществляется по спаду импульса. Для этого существуют следующие обозначения, приведенные на рис. 3.13 и рис. 3.14 соответственно.

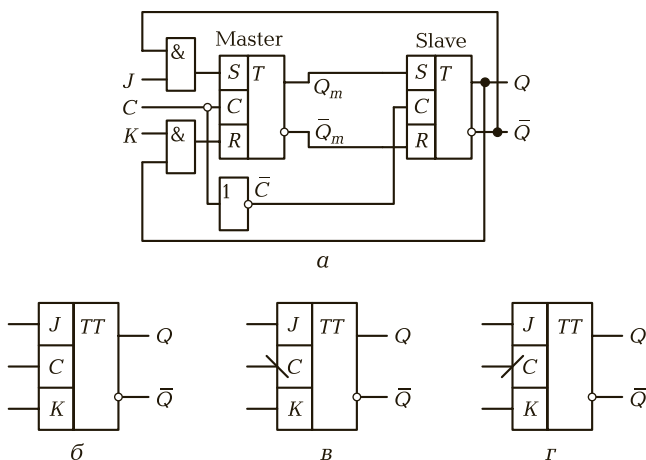


Рис. 3.15. *JK*-триггер со статическим управлением:  
*а* — структурная схема; *б* — графическое обозначение; графическое обозначение *JK*-триггера с динамическим управлением: *в* — по заднему фронту; *г* — по переднему фронту

Таблица 3.8

$J$	$K$	$Q$
0	0	Режим хранения информации без изменения
0	1	0
1	0	1
1	1	Инверсия предыдущего состояния

Для устранения недостатка  $RS$ -триггеров (режим неопределенности при  $R = S = 1$ ) применяются *динамические JK-триггеры* (рис. 3.15) с более широкими функциональными возможностями.

Сокращенная таблица состояний  $JK$ -триггера показана в табл. 3.8.

Из таблицы состояний видно, что для первых трех строк (наборов переменных) входы  $J$  и  $K$  играют роль входов  $S$  и  $R$   $RS$ -триггера. Однако для четвертого набора переменных, когда  $J = K = 1$ , состояние триггера сильно отличается от состояния  $RS$ -триггера. Для  $RS$ -триггера — это запрещенная комбинация входных переменных, а в  $JK$ -триггере меняется (инвертируется) предыдущее состояние.  $JK$ -триггер можно синтезировать (построить) на базе  $MS$ -триггера.

Рассмотрим следующий тип триггера:  $D$ -триггер (от англ. Delay). Схемное обозначение  $D$ -триггера приведено на рис. 3.16.

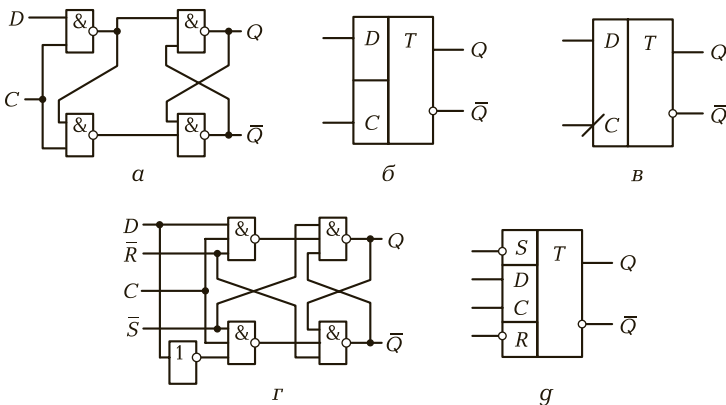


Рис. 3.16.  $D$ -триггер со статическим управлением (а — структурная схема; б — графическое обозначение); с динамическим управлением (в — графическое обозначение); с асинхронными инверсными входами установки и сброса  $\bar{S}$  и  $\bar{R}$  (г — структурная схема; д — графическое обозначение)

Он обладает двумя устойчивыми состояниями и имеет, как минимум, две входные линии: информационный вход  $D$  (Data) и вход управления записью (защелкиванием)  $L$  (Load или Latch) — отсюда его второе имя: «защелка». Последний вход часто обозначают символом  $C$  (lock).

$D$ -триггер называют *информационным триггером*, а также *триггером задержки*.  $D$ -триггер бывает только синхронным. Для триггера типа  $D$  состояние в интервале времени между сигналом на входной линии и следующим состоянием триггера формируется проще, чем для любого другого типа.

$D$ -триггер можно дополнить асинхронными инверсными входами установки и сброса  $\bar{S}$  и  $\bar{R}$  (рис. 3.16,  $g$ ).

Если на вход  $\bar{S}$  подать 0, а на вход  $\bar{R}$  — сигнал 1, то  $Q = 1$  и  $\bar{Q} = 0$  независимо от сигналов на остальных входах. Триггер устанавливается в единицу. Сигналы  $D$  и  $C$  не влияют на этот процесс. В силу этого, асинхронные входы ( $\bar{S}$  и  $\bar{R}$ ) имеют наивысший приоритет. Так как асинхронные связи симметричны, при  $\bar{S} = 1$  и  $\bar{R} = 0$  триггер сбрасывается ( $Q = 0$ ).

Полная таблица состояний  $D$ -триггера приведена в табл. 3.9.

Из таблицы видно, что при отсутствии синхроимпульсов состояние  $D$ -триггера не меняется. Отметим, что  $D$ -триггеры могут переключаться как уровнем синхроимпульса, так и его фронтом. В технической литературе  $D$ -триггер, управляемый уровнем синхроимпульса, известен также как триггер-защелка.

Посмотрим на графическое обозначение  $D$ -триггера. Если соединить инверсный выход с входом  $D$  (рис. 3.17), то получим  $T$ -триггер

**Таблица 3.9**

№ п/п	$C$	$D$	Состояние $D$ -триггера	
			до поступления сигнала	после поступления сигнала
1	0	0	0	0
2	0	0	1	1
3	0	1	0	0
4	0	1	1	1
5	1	0	0	0
6	1	0	1	0
7	1	1	0	1
8	1	1	1	1



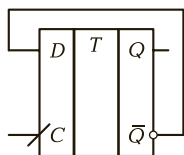


Рис. 3.17. Схема  $T$ -триггера, построенная на основе  $D$ -триггера

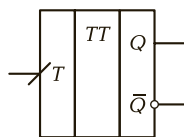


Рис. 3.18. Графическое обозначение  $T$ -триггера

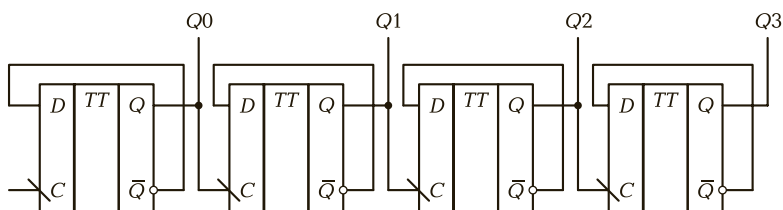


Рис. 3.19. Схема четырехразрядного счетчика

(рис. 3.18). При подаче сигнала на вход  $C$  происходит изменение состояния триггера, как мы говорили ранее.

$T$ -триггер часто называют *счетным триггером*. У  $T$ -триггера имеется только один вход. После поступления на этот вход импульса состояние  $T$ -триггера меняется на противоположное. Счетным он называется потому, что он как бы подсчитывает количество импульсов, поступивших на его вход. «Жаль» только, что считать этот триггер умеет только до одного. При поступлении второго импульса  $T$ -триггер снова сбрасывается в исходное состояние.

$T$ -триггер можно построить и на  $JK$ -триггерах.  $JK$ -триггер переходит в инверсное состояние каждый раз при одновременной подаче на входы  $J$  и  $K$  логической единицы. Если объединить входы  $J$  и  $K$ , то получим  $T$ -триггер, используя  $JK$ -триггер. В результате триггер при каждом счетном импульсе запоминает значение  $Q$ , т. е. будет переключаться в противоположное состояние.

$T$ -триггеры используются при построении схем различных счетчиков (рис. 3.19).

### 3.4. РЕГИСТРЫ

**Регистром** называется узел, предназначенный для приема, временного хранения и выдачи машинного слова. Регистры могут также использоваться для некоторых операций преобразования

данных. Эти дополнительные функции регистров обеспечиваются путем усложнения схем хранения, выбора более сложных триггеров и подключения дополнительных логических схем на их входах и выходах.

Таким образом, регистры представляют собой совокупность триггеров, число которых соответствует числу разрядов в слове, и вспомогательных схем, обеспечивающих выполнение различных операций над словом.

Регистры по своим функциональным обязанностям делятся на типы:

- регистры с параллельной записью (регистры памяти);
- последовательные регистры сдвига;
- реверсивные регистры сдвига.

Регистры памяти — простейший вид регистров. Их назначение — хранить двоичную информацию небольшого объема в течение короткого промежутка времени. Эти регистры представляют собой набор синхронных триггеров, каждый из которых хранит один разряд двоичного числа. Рассмотрим четырехразрядный регистр на базе  $D$ -триггера. На рис. 3.20 приведена функциональная схема

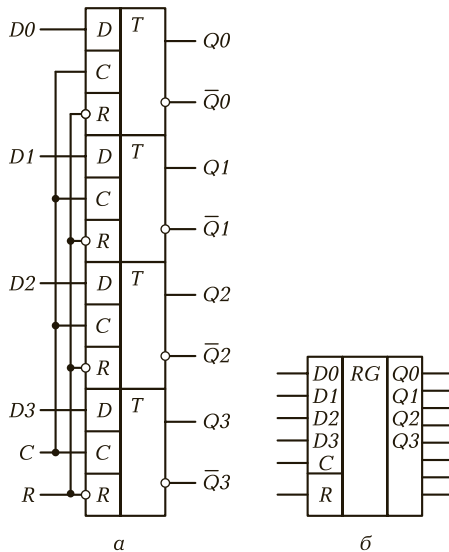


Рис. 3.20. Регистр памяти:

а — функциональная схема; б — условное обозначение

(рис. 3.20, а) и условное обозначение регистра (рис. 3.20, б), который содержит четыре  $D$ -триггера.

У него есть информационные входы  $D0, D1, D2, D3$ ; линия сброса  $R$ ; линия записи  $C$ ; прямые и инверсные выходы от каждого триггера.

Ввод и вывод информации производится одновременно во всех разрядах параллельным кодом. Запись данных, поступивших на информационные входы, происходит одновременно во всех триггерах по сигналу на входе  $C$ . С приходом очередного тактового импульса записанная информация обновляется.

Сигналы на выходах триггеров характеризуют выходную информацию. Она может быть в виде прямого или обратного кода (с инверсных выходов). Регистры памяти представляют собой, по существу, наборы триггеров с независимыми информационными входами и общим тактовым входом.

Последовательный регистр сдвига (рис. 3.21) — регистр, обеспечивающий помимо хранения информации, сдвиг влево или вправо всех разрядов одновременно на одинаковое число позиций. При этом выдвигаемые за пределы регистра разряды теряются, а в освобождающиеся разряды заносится информация, поступающая по отдельному внешнему входу регистра сдвига. Обычно эти регистры обеспечивают сдвиг кода на одну позицию влево или вправо. Но существуют и универсальные регистры сдвига, которые выполняют сдвиг как влево, так и вправо в зависимости от значения сигнала на специальном управляющем входе или при подаче синхросигналов на разные входы регистра. Регистр сдвига может быть спроектирован и таким образом, чтобы выполнять сдвиг одновременно не на одну, а на несколько позиций.

Регистры сдвига строятся на  $D$ -триггерах. В обозначениях регистров сдвига направление стрелки, указывающей сдвиг, условно. В разных справочниках ее направление различно. Условно принимается, что сдвиг производится от младшего разряда к старшему.

Реверсивный регистр сдвига может сдвигать данные от разряда к разряду, как в одну сторону, так и в другую. Чтобы обеспечить реверс сдвига, сразу на все  $D$ -входы триггеров подается сигнал «0»

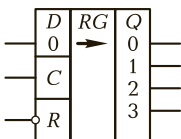


Рис. 3.21. Графическое обозначение четырехразрядного регистра сдвига

или «1». При этом происходит подключение входа  $i$ -го триггера к выходу младшего или старшего соседа.

Триггеры и регистры являются простейшими представителями цифровых микросхем, имеющих внутреннюю память. Если выходные сигналы логических элементов и комбинационных микросхем однозначно определяются их текущими входными сигналами, то выходные сигналы микросхем с внутренней памятью зависят также еще и от того, какие входные сигналы и в какой последовательности поступали на них в прошлом, т. е. они помнят предысторию поведения схемы. Именно поэтому их применение позволяет строить гораздо более сложные и интеллектуальные цифровые устройства, чем в случае простейших микросхем без памяти.

Триггеры и регистры сохраняют свою память только до тех пор, пока на них подается напряжение питания. Иначе говоря, их память относится к типу оперативной памяти. После выключения питания и его последующего включения триггеры и регистры переходят в случайное состояние, т. е. их выходные сигналы могут устанавливаться как в уровень логической единицы, так и в уровень логического нуля. Это необходимо учитывать при проектировании схем.

Большим преимуществом триггеров и регистров перед другими типами микросхем с памятью является их максимально высокое быстродействие (минимальное время задержек срабатывания и максимально высокая допустимая рабочая частота). Именно поэтому триггеры и регистры иногда называют также *сверхоперативной памятью*. Однако недостаток триггеров и регистров в том, что объем их внутренней памяти очень мал, они могут хранить только отдельные сигналы, биты (триггеры) или машинные слова (регистры).

### 3.5. СЧЕТЧИКИ

**Счетчик** позволяет осуществлять подсчет поступающих на его вход сигналов и фиксацию результата в виде многоразрядного двоичного числа. Счетчик, состоящий из  $n$ -триггеров, дает возможность подсчитывать до  $N$  сигналов, связанных зависимостью:  $N = 2^n$ .

Счетчики используются для таких операций, как подсчет импульсов, сдвигов адресов, формирование адресов и т. д.

По своим функциям счетчики разделяются на суммирующие, вычитающие, реверсивные. Они также различаются логикой работы дополнительных логических элементов, подключаемых к тригграм.

В основу построения любого счетчика положено свойство  $T$ -триггеров изменять свое состояние при подаче очередного сигнала на счетный вход  $T$ .

Обычно требуется посчитать большее количество сигналов. В этом случае можно использовать выходной сигнал первого счетного триггера как входной сигнал для следующего триггера, т.е. соединить триггеры последовательно. Так можно построить любой счетчик, считающий до максимального числа, кратного степени два. Схема счетчика, позволяющего посчитать любое количество импульсов, меньшее шестнадцати, приведена на рис. 3.19. Количество поступивших на вход импульсов можно узнать, подключившись к выходам счетчика  $Q0...Q3$ . Это число будет представлено в двоичном коде.

При поступлении на счетный вход счетчика очередного импульса содержимое его триггеров увеличивается на 1. Поэтому такие счетчики получили название суммирующих двоичных счетчиков.

Представим работу этого счетчика в табл. 3.10.

**Таблица 3.10**

Номер входного импульса	$Q3$	$Q2$	$Q1$	$Q0$
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

**Таблица 3.11**

Вход $X$	$Q2$	$Q1$	$Q0$
0	0	0	0
1	0	0	1
0	0	0	1
1	0	1	0
0	0	1	0
1	0	1	1
0	0	1	1
1	1	0	0
0	1	0	0
1	1	0	1
0	1	0	1
1	1	1	0
0	1	1	0
1	1	1	1
0	1	1	1
1	0	0	0

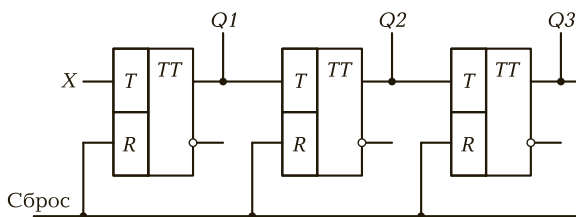


Рис. 3.22. Функциональная схема трехразрядного счетчика

Схема трехразрядного суммирующего счетчика, построенного на  $T$ -триггерах, показана на рис. 3.22. Логика его работы приведена в табл. 3.11.

В табл. 3.11 показано, что на входе может быть либо 0 (режим хранения), либо 1 (режим счета).

Счетчик может работать на увеличение выходного кода по каждому входному импульсу; это основной режим, имеющийся во всех счетчиках, он называется *режимом прямого счета*. Счетчик может также работать на уменьшение выходного кода по каждому входному импульсу; это режим обратного или инверсного счета, предусмотренный в счетчиках, называемых реверсивными. Инверсный счет бывает довольно удобен в схемах, где необходимо отсчитывать заданное количество входных импульсов.

Точно так же, как и в случае регистров, внутренняя память счетчиков оперативная, т.е. ее содержимое сохраняется только до тех пор, пока включено питание схемы. С выключением питания память стирается, а при новом включении питания схемы содержимое памяти будет произвольным, случайным, зависящим только от конкретной микросхемы, т.е. выходные сигналы счетчиков будут произвольными.

В стандартные серии микросхем входят несколько разновидностей счетчиков (рис. 3.23). Они различаются способом счета и управляющими сигналами (наличием или отсутствием сигнала сброса):

- ИЕ9 — двоично-десятичный;
- ИЕ10 — двоичный;
- ИЕ12 — реверсивный двоично-десятичный счетчик с отсутствием сигнала сброса в нуль;
- ИЕ13 — реверсивный двоичный счетчик с отсутствием сигнала сброса в нуль;
- ИЕ16 — двоично-десятичный счетчик с синхронной параллельной записью по фронту тактового сигнала  $C$ , возможностью

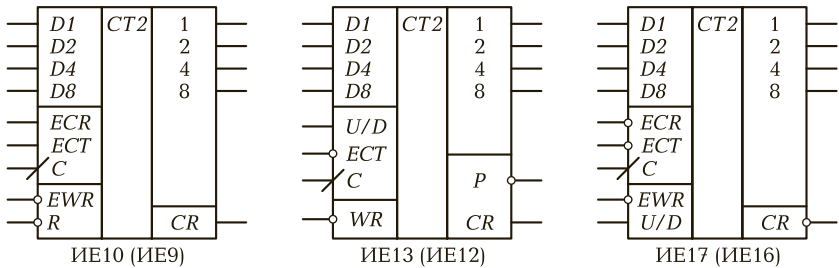


Рис. 3.23. Примеры графических обозначений счетчиков

прямого и обратного счета и отсутствием сигнала сброса в нуль;

- ИЕ17 — двоичный счетчик с синхронной параллельной записью по фронту тактового сигнала  $C$ , возможностью прямого и обратного счета и отсутствием сигнала сброса в нуль.

Все счетчики считают по положительному фронту тактового сигнала, имеют выход переноса  $CR$  и входы расширения для каскадирования, а также обладают возможностью параллельной записи информации.

На схемах использованы следующие обозначения:

- $R$  — сброс;
- $EWR$  — сигнал разрешения записи;
- $ECR$  — сигнал разрешения переноса;
- $ECT$  — сигнал разрешения счета;
- $CR$  — сигнал переноса;
- $C$  — тактовый сигнал;
- $U/D$  — 0 — прямой счет; 1 — обратный счет;
- $WR$  — сигнал параллельной записи;
- $P$  — выход последовательного переноса.

### 3.6. СУММАТОРЫ

**Сумматор** — устройство для суммирования кодов. Как правило, любой сумматор представляет собой комбинацию одноразрядных сумматоров.

Сумматоры различают по принципам построения на накапливающие (последовательные, асинхронные) и комбинационные (параллельные, синхронные).

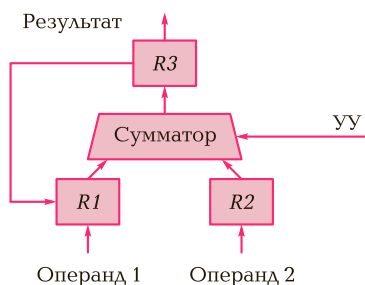


Рис. 3.24. Упрощенная схема работы сумматора

*Накапливающие сумматоры* строят на сложных *JK*- и *RS*-триггерах, дополняя их выходы достаточно сложными схемами формирования и распространения переносов. Процесс сложения при этом осуществляется поэтапно. Сначала на триггерах сумматора фиксируется первый код, затем подается второй код. В итоге на каждом триггере формируются одноразрядные суммы и значения переносов между разрядами. Учет возникающих переносов задерживает формирование окончательного результата суммы и может требовать дополнительных тактов сложения. Из-за этого многоразрядные схемы сумматора накапливающего типа используются достаточно редко.

Чаще для построения сумматоров используются сумматоры комбинационного типа. Обычно у такого сумматора на входе и выходе имеются регистры для хранения и преобразования складываемых кодов и результата (рис. 3.24).

Регистр *R1* предназначен для хранения первого операнда, регистр *R2* — для хранения второго операнда. Сумматор по сигналам из устройства управления настраивается на выполнение определенной машинной операции, соответствующей коду операции, находящемуся в коде команды. Результат выполняемой операции фиксируется в регистре *R3*. При необходимости этот результат может использоваться для продолжения вычислений. Для этого предусматривается возможность перезаписи содержимого регистра *R3* на *R1* в качестве значения одного из операндов при выполнении очередной операции.

*Комбинационные сумматоры* представляют собой наиболее быстродействующую разновидность сумматоров. Нарастивание их разрядности при соблюдении определенных условий не приводит к увеличению полной задержки срабатывания. Можно считать, что именно комбинационные сумматоры работают как идеальные сумматоры, все разряды которых срабатывают одновременно, парал-



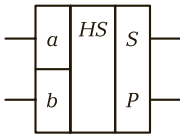


Рис. 3.25. Графическое обозначение полусумматора

тельно. Задержка срабатывания сумматора в этом случае примерно равна задержке срабатывания одного триггера. Достигается такое быстродействие существенным усложнением внутренней структуры микросхемы.

Вместе с тем недостатком комбинационных сумматоров является более сложное управление их работой по сравнению с накапливающими. Поэтому комбинационные счетчики целесообразно применять только в тех случаях, когда действительно требуется очень высокое быстродействие, очень высокая скорость переключения разрядов. В противном случае усложнение схемы управления может быть не оправдано.

Если на одноразрядной суммирующей схеме имеются два входа, то данная схема называется *полусумматором*, а если три, то — *полным одноразрядным сумматором*.

Рассмотрим полусумматор с двумя входами  $a$  и  $b$  (рис. 3.25). Выходными сигналами такого устройства являются сумма  $S$  и перенос  $P$ , которые вырабатываются согласно табл. 3.12 функционирования данного устройства.

В табл. 3.12 выходные сигналы  $P$  и  $S$  не случайно расположены именно в такой последовательности. Это подчеркивает, что  $PS$  рассматривается как двухразрядное двоичное число, например:

$$1 + 1 = 10^2, \text{ т.е. } P = 1, \text{ а } S = 0.$$

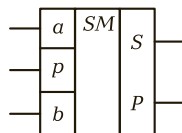
Обозначением полусумматора служат буквы  $HS$ .

Рассмотрим полный одноразрядный сумматор с тремя входами  $a$ ,  $b$  и  $p$  (рис. 3.26). Выходными сигналами такого устройства являются сумма  $S$  и перенос  $P$ , которые вырабатываются согласно табл. 3.13 функционирования данного устройства.

Таблица 3.12

Вход		Выход	
$a$	$b$	$P$	$S$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Рис. 3.26. Графическое обозначение полного одноразрядного сумматора



В табл. 3.13 выходные сигналы  $P$  и  $S$  не случайно расположены именно в такой последовательности. Это подчеркивает, что  $PS$  рассматривается как двухразрядное двоичное число, например:

$$1 + 1 + 0 = 10_2, \text{ т. е. } P = 1, \text{ а } S = 0.$$

Обозначением полного одноразрядного сумматора служат буквы  $SM$ .

Таблица 3.13

Вход			Выход	
$a$	$b$	$p$	$P$	$S$
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

## 3.7. ШИФРАТОРЫ И ДЕШИФРАТОРЫ

Рассмотрим рис. 3.27. Поставим задачу: построить такое устройство, которое позволяет нам определить, по какому номеру входа пришел сигнал.

Один из способов — это получить на выходе двоичный код, который соответствует номеру входа. Построим таблицу соответствий для этого устройства (табл. 3.14).

Итак, нам потребуется два разряда, чтобы знать, по какому входу пришел сигнал. Такой прибор называется шифратором.

**Шифратор** — устройство, преобразующее номер входного сигнала в выходной двоичный код (шифрует номер входного сигнала). Микросхемы шифраторов обозначаются на схемах буквами  $CD$ .

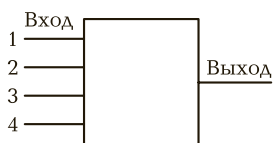


Рис. 3.27. Пример устройства для определения номера входа, по которому пришел сигнал

Таблица 3.14	
Номер входа	Двоичный код на выходе
1	00
2	01
3	10
4	11

Если на выходе мы можем определить только  $n$  разрядов, т. е. построить двоичный код из  $n$  разрядов, то мы можем определить только  $2^n$  входов. Другими словами, количество входных сигналов шифратора равно количеству возможных состояний двоичного кода на выходе, т. е.  $2^n$ , где  $n$  — разрядность двоичного кода (рис. 3.28).

Выходной код шифратора однозначно определяется номером входного сигнала. В отечественных сериях шифраторы имеют в названии буквы *ИВ*.

Если сигнал на входе приходит только по одной линии, то на выходе формируется только один определенный код.

Если на входе одновременно или почти одновременно приходят сигналы по нескольким линиям, то возможны следующие варианты:

- данный случай запрещен;
- формируется выходной код, соответствующий входу с наибольшим номером, т. е. старшие входы имеют приоритет перед младшими. Такой шифратор называется *приоритетным*;
- синхронизация сигналов.

Одновременное или почти одновременное изменение сигналов на входе шифратора приводит к появлению периодов неопределенности на выходах. Выходной код может на короткое время принимать значение, не соответствующее ни одному из входных сигналов. Поэтому в тех случаях, когда входные сигналы могут приходиться

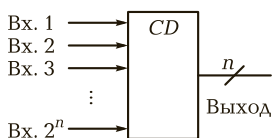


Рис. 3.28. Количество возможных состояний на выходе шифратора

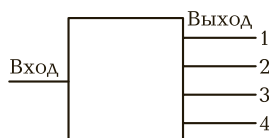


Рис. 3.29. Пример устройства для определения номера линии для включения в зависимости от входного сигнала

одновременно, необходима синхронизация выходного кода, например с помощью какого-либо разрешающего сигнала, который должен приходиться только тогда, когда состояние неопределенности уже закончилось.

Для шифратора задержка кода от входа до выхода примерно в полтора раза превышает задержку логического элемента. Точные величины задержек микросхем надо смотреть в справочниках.

Рассмотрим рис. 3.29. Поставим задачу обратную шифрованию: построить такое устройство, которое позволяет нам включить определенный номер линии в зависимости от входного сигнала.

На входе имеем двоичный код, каждому возможному коду поставим в соответствие номер выхода. Теперь имеется возможность включить номер выходной линии, который соответствует входному коду. Построим таблицу соответствий для этого устройства (табл. 3.15).

Итак, нам потребуется четыре номера выхода, чтобы поставить их в соответствие двум разрядам входного двоичного кода. Такой прибор называется дешифратором.

**Дешифратор** — это устройство, преобразующее входной двоичный код в унитарный код на выходах, который определяет номер выхода. Унитарным называется двоичный код, содержащий одну и только одну единицу, например 0000100, или только один нуль, например 11110111. Количество выходных сигналов дешифратора равно количеству возможных состояний входного двоичного кода, т. е.  $2^n$ , где  $n$  — разрядность двоичного кода (рис. 3.30). Микросхемы дешифраторов обозначаются на схемах буквами *DC*.

Двоичный код на входе	Номер выхода
00	1
01	2
10	3
11	4

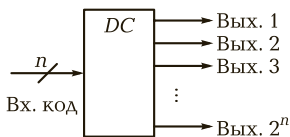


Рис. 3.30. Соотношение количеств выходных линий и входных двоичных кодов в дешифраторе

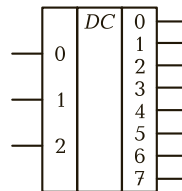


Рис. 3.31. Условно-графическое обозначение дешифратора на три входа

Таблица 3.16

Входы			Выходы							
2	1	0	0	1	2	3	4	5	6	7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

На выходе дешифратора всегда присутствует только один унитарный код сигнала, который однозначно определяется входным кодом.

Условно-графическое обозначение дешифратора на три входа приведено на рис. 3.31.

Рассмотрим таблицу истинности дешифратора с тремя входами и восемью выходами (табл. 3.16).

В стандартные серии входят следующие дешифраторы:

- на 4 выхода (2 разряда входного кода), обозначаются как «2—4»;
- на 8 выходов (3 разряда входного кода), обозначаются как «3—8»;
- на 16 выходов (4 разряда входного кода), обозначаются как «4—16».

Существуют дешифраторы «4—10», которые обрабатывают не все возможные 16 состояний входного кода, а только первые 10 из них.

Различаются микросхемы дешифраторов входами управления (разрешения/запрета выходных сигналов), а также типом выхода.

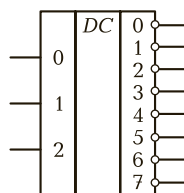


Рис. 3.32. Условно-графическое обозначение дешифратора с инверсными выходами

Таблица 3.17

Входы			Выходы							
2	1	0	0	1	2	3	4	5	6	7
0	0	0	0	1	1	1	1	1	1	1
0	0	1	1	0	1	1	1	1	1	1
0	1	0	1	1	0	1	1	1	1	1
0	1	1	1	1	1	0	1	1	1	1
1	0	0	1	1	1	1	0	1	1	1
1	0	1	1	1	1	1	1	0	1	1
1	1	0	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	0

Выходные сигналы всех дешифраторов имеют отрицательную полярность. Входы, на которые поступает входной код, называют часто *адресными входами*. Обозначают эти входы 1, 2, 4, 8, где число соответствует разряду двоичного кода ( $2^0 = 1$  — младший разряд,  $2^1 = 2$ ;  $2^2 = 4$ ;  $2^3 = 8$  — следующие разряды), или  $A_0$ ,  $A_1$ ,  $A_2$ ,  $A_3$ . В отечественных сериях микросхемы дешифраторов обозначаются буквами ИД.

Если построить дешифратор, у которого значения выходных сигналов имеют инверсный вид по отношению к представленному в табл. 3.16, то получим дешифратор с инверсными выходами (рис. 3.32). Таблица истинности дешифратора с инверсными выходами показана в табл. 3.17.

Наиболее типичное применение дешифраторов состоит именно в дешифрировании входных кодов, при этом входы используются как стробирующие управляющие сигналы. Номер активного (нулевого) выходного сигнала показывает, какой входной код поступил. Если нужно дешифровать код с большим числом разрядов, то можно объединить несколько микросхем дешифраторов. Старшие разряды кода подаются на основной дешифратор, выходы которого разрешают работу нескольких дополнительных дешифраторов. На объединенные входы этих дополнительных дешифраторов подаются младшие разряды входного кода, хотя лучше, конечно, взять готовую микросхему.

Шифраторы используются реже, чем дешифраторы. Это связано с более специфической областью их применения. Значительно меньше и выбор микросхем шифраторов в стандартных сериях.

## 3.8. МУЛЬТИПЛЕКСОРЫ

Если шифраторы и дешифраторы обеспечивали связь между входом и выходом при помощи одной (входной или выходной) линии  $n$  (выходных или входных) линиями, то на практике часто стоит задача связать  $m$  входных линий и  $n$  выходных.

**Мультиплексоры** (Multiplexer) предназначены для поочередной передачи на один выход одного из нескольких входных сигналов, т.е. для их мультиплексирования.

Линии входа называются *каналами*, а линии выхода — *разрядами*. Количество мультиплексируемых входов называется *количеством каналов* мультиплексора. Количество выходов называется *числом разрядов* мультиплексора. Например:

- 2-канальный 4-разрядный мультиплексор имеет 4 выхода, на каждый из которых может передаваться один из двух входных сигналов;
- 4-канальный 2-разрядный мультиплексор имеет 2 выхода, на каждый из которых может передаваться один из четырех входных сигналов.

Число каналов мультиплексоров, входящих в стандартные серии, составляет от 2 до 16, а число разрядов — от 1 до 4. Чем больше каналов имеет мультиплексор, тем меньше у него разрядов.

Управление работой мультиплексора (выбор номера канала) осуществляется с помощью входного двоичного кода адреса. Например, для 4-канального мультиплексора необходим 2-разрядный управляющий (адресный) код, а для 16-канального — 4-разрядный код. Разряды кода обозначаются 1, 2, 4, 8 или  $A_0, A_1, A_2, A_3$ . Выходы мультиплексоров бывают прямыми и инверсными. Некоторые микросхемы мультиплексоров имеют вход разрешения/запрета —  $C$  (другое обозначение —  $S$ ), который при запрете устанавливает прямой выход в нулевой уровень.

На рис. 3.33. показан мультиплексор, у которого обозначения информационных входов  $A, B, C, D$  и выхода  $Out$  в зарубежной литературе заменены на  $X_0, X_1, X_2, X_3$  и  $Y$ . Такие обозначения

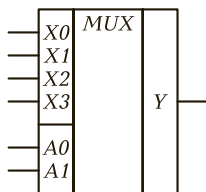


Рис. 3.33. Графическое обозначение мультиплексора на четыре входа

входов и выходов более распространены в отечественной литературе. Адресные входы обозначены как  $A0$  и  $A1$ .

Мультиплексоры обозначают через  $MUX$  или  $MS$ . В отечественных микросхемах мультиплексоры обозначаются буквами КП, следующими непосредственно за номером серии микросхем, например, К1533КП1. Микросхемы мультиплексоров можно объединять для увеличения количества каналов.

Задержки выходного сигнала мультиплексора по входам управляющего кода примерно в два раза превышают задержки логических элементов, а по информационным входам — примерно в полтора раза.

### 3.9. КОМПАРАТОРЫ КОДОВ

Микросхемы компараторов кодов (Comparator) применяются для сравнения двух входных кодов и выдачи на выходы сигналов о результатах этого сравнения (о равенстве или неравенстве кодов). На схемах компараторы кодов обозначаются двумя символами равенства: « $=$ » и « $\neq$ ». Код типа микросхемы компаратора кода в отечественных сериях — СП (рис. 3.34). В данном компараторе кодов имеется:

- 8 входов для двух 4-разрядных кодов  $A$  и  $B$ ;
- 3 управляющих входа для наращивания разрядности ( $A > B$ ,  $A < B$ ,  $A = B$ );
- 3 выхода результирующих сигналов ( $A > B$ ,  $A < B$ ,  $A = B$ ). Для удобства на схемах управляющие входы и выходы иногда обозначают просто « $>$ », « $<$ » и « $=$ ».

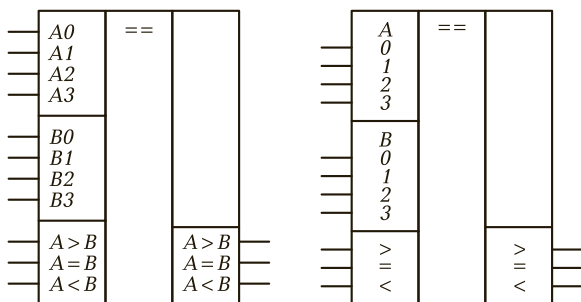


Рис. 3.34. Четырехразрядный компаратор кодов СП1 (два варианта обозначения)



Нулевые разряды кодов ( $A0$  и  $B0$ ) — младшие, третьи разряды ( $A3$  и  $B3$ ) — старшие.

Задержки компараторов кодов по входам разрядов кодов примерно вчетверо больше задержек логических элементов, а по входам расширения — примерно втрое. Так что эти микросхемы довольно медленные по сравнению с другими комбинационными микросхемами. Точные значения задержек надо смотреть в справочниках.

### КОНТРОЛЬНЫЕ ВОПРОСЫ

---

1. Как логические элементы отображаются на схемах?
2. Какие логические элементы называются триггерами и как они обозначаются на схемах?
3. Каково назначение триггеров?
4. Какие логические элементы называются регистрами и как они обозначаются на схемах?
5. Каково назначение регистров?
6. Какие логические элементы называются счетчиками и как они обозначаются на схемах?
7. Для чего нужны сумматоры?
8. Чем отличаются шифраторы от дешифраторов?
9. Каково назначение мультиплексоров?
10. Каково назначение компараторов?