



3. КОМБИНИРОВАННЫЕ УСТРОЙСТВА

В компьютере есть набор устройств, которые выполняют операции над данными, не сохраняя их.

Из таких операций можно отметить следующие:

- передача информации из нескольких каналов в один канал и наоборот;
- кодирование и декодирование информации;
- выполнение арифметических операций,
- выполнение логических операций.

Для выполнения этих операций используются цифровые электронные устройства, называемые комбинационными.

Определение. Комбинационное устройство - это электронное цифровое устройство, используемое для обработки информации без ее хранения.

Из комбинационных устройств можно назвать следующие:

- декодер;
- кодер;
- переводчик кода;
- мультиплексор;
- демультиплексор;
- компаратор;
- арифметико-логических единиц.

Комбинационные устройства отличаются от устройств, которые могут хранить информацию (бистабильные устройства, регистры, счетчики) следующими свойствами:

- они не могут удерживать информацию или у них отсутствует память;
- следующее состояние не зависит от предыдущего;
- состояние на выходах устройства однозначно зависит от состояния, поданного на его входы.

3.1. Декодер

Определение. Дешифратором называется комбинационное электронное устройство, которое имеет k входов и $m=2^k$ выходов и используется для преобразования двоичного кода, подаваемого на входы, в электрический сигнал (логическую 1 или 0) только на одном из выходов.

Мы различаем полный декодер, если $m=2^k$, и неполный декодер, если $m < 2^k$.

Пусть, в качестве примера, приведена таблица истинности полного декодера, для которого $k = 3$ и $m = 8$ (см. табл. 3.1). Таблица 3.1 заполняется в соответствии с определением декодера.

Таблица 3.1. Таблица истинности полного декодера, для которого $k = 3$ и $m = 8$

Н т д/о	Переменная			Функц ии							
	x_2	x_1	x_0	F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7
0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	0	0	0	0	0
3	0	1	1	0	0	0	1	0	0	0	0
4	1	0	0	0	0	0	0	1	0	0	0
5	1	0	1	0	0	0	0	0	1	0	0

6	1	1	0	0	0	0	0	0	0	1	0
7	1	1	1	0	0	0	0	0	0	0	1

3. КОМБИНИРОВАННЫЕ УСТРОЙСТВА

В компьютере есть набор устройств, которые выполняют операции над данными, не сохраняя их.

Из таких операций можно отметить следующие:

- а. передача информации из нескольких каналов в один канал и наоборот;
- б. кодирование и декодирование информации;
- с. выполнение арифметических операций,
- д. выполнение логических операций.

Для выполнения этих операций используются цифровые электронные устройства, называемые комбинационными.

Определение. Комбинационное устройство - это электронное цифровое устройство, используемое для обработки информации без ее хранения.

Из комбинационных устройств можно назвать следующие:

- декодер;
- кодер;
- переводчик кода;
- мультиплексор;
- демультиплексор;
- компаратор;
- арифметико-логических единиц.

Комбинационные устройства отличаются от устройств, которые могут хранить информацию (бистабильные устройства, регистры, счетчики) следующими свойствами:

- а) они не могут удерживать информацию или у них отсутствует память;
- б) следующее состояние не зависит от предыдущего;
- с) состояние на выходах устройства однозначно зависит от состояния, поданного на его входы.

3.1. Декодер

Определение. Дешифратором называется комбинационное электронное устройство, которое имеет k входов и $m=2^k$ выходов и используется для преобразования двоичного кода, подаваемого на входы, в электрический сигнал (логическую 1 или 0) только на одном из выходов.

Мы различаем полный декодер, если $m=2^k$, и неполный декодер, если $m < 2^k$.

Пусть, в качестве примера, приведена таблица истинности полного декодера, для которого $k = 3$ и $m = 8$ (см. табл. 3.1). Таблица 3.1 заполняется в соответствии с определением декодера.

Таблица 3.1. Таблица истинности полного декодера, для которого $k = 3$ и $m = 8$

Нет д/о	Переменная			Функции							
	x_2	x_1	x_0	F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7
0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	0	0	0	0	0
3	0	1	1	0	0	0	1	0	0	0	0
4	1	0	0	0	0	0	0	1	0	0	0
5	1	0	1	0	0	0	0	0	1	0	0
6	1	1	0	0	0	0	0	0	0	1	0
7	1	1	1	0	0	0	0	0	0	0	1

Из таблицы истинности 3.1 мы получаем FCND следующих логических функций:

$$_0 F(x_2, x_1, x_0) = \bar{x}_2 \times \bar{x}_1 \times \bar{x}_0; \quad (3.1)$$

$$_1 F(x_2, x_1, x_0) = \bar{x}_2 \times \bar{x}_1 \times x_0; \quad (3.2)$$

$$_2 F(x_2, x_1, x_0) = \bar{x}_2 \times x_1 \times \bar{x}_0; \quad (3.3)$$

$$_3 F(x_2, x_1, x_0) = \bar{x}_2 \times x_1 \times x_0; \quad (3.4)$$

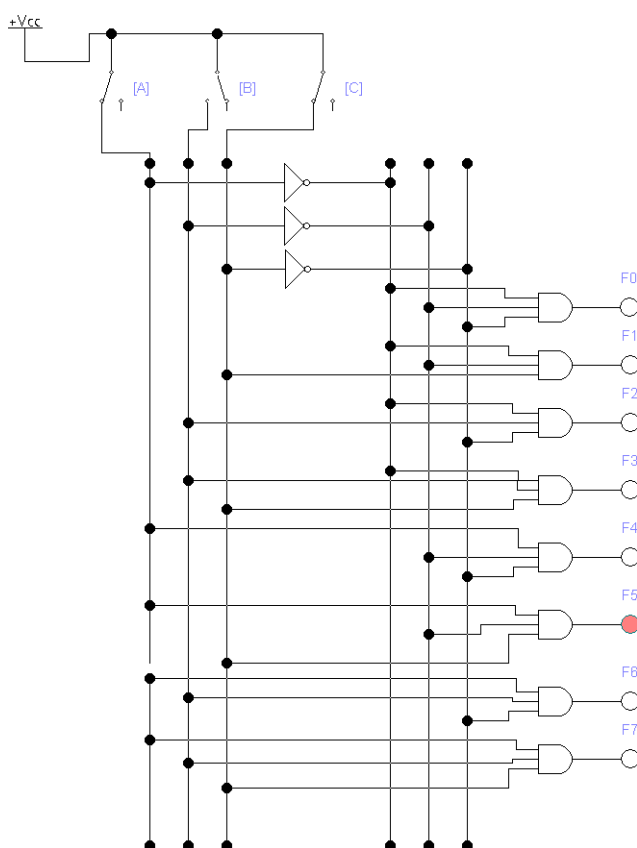
$$_4 F(x_2, x_1, x_0) = x_2 \times \bar{x}_1 \times \bar{x}_0; \quad (3.5)$$

$$_5 F(x_2, x_1, x_0) = x_2 \times \bar{x}_1 \times x_0; \quad (3.6)$$

$$_6 F(x_2, x_1, x_0) = x_2 \times x_1 \times \bar{x}_0; \quad (3.7)$$

$$_7 F(x_2, x_1, x_0) = x_2 \times x_1 \times x_0. \quad (3.8)$$

На рис. 3.1 показана электрическая схема декодера, построенная в соответствии с логическими функциями (3.1)...(3.8). На рис. 3.1 используются следующие обозначения: +Vcc - источник сигнала; A, B, C - переключатели шины данных; F7, ... , F0 - выходы



декодера. Из диаграммы видно, что на выходе F5 появляется электрический сигнал, если на входы подан код ABC = 1012.

Рис. 3.1. Схема подключения декодера.

Символ обычного декодера показан на рисунке 3.2.

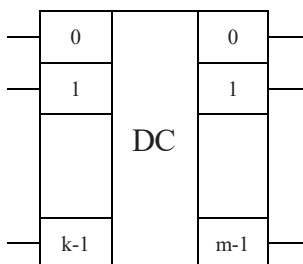


Рис. 3.2. Символ обычного декодера.

Декодер в основном используется для преобразования двоичного кода в командный импульс. Командные импульсы используются для управления рабочими состояниями компьютерных устройств. Декодер является частью микропроцессорного блока управления, постоянной памяти и цифро-аналогового преобразователя (ЦАП).

3.2. Кодер

Определение. Мы называем кодером комбинационное электронное устройство, которое имеет $k = 2^m$ входов и m выходов и используется для преобразования электрического сигнала (логической 1 или 0), подаваемого на один из входов, в двоичный код, представленный на выходах.

Мы различаем полный кодер, если $k=2^m$, и неполный кодер, если $k<2^m$.

Пусть, в качестве примера, дана таблица истинности неполного кодера, который имеет $k = 10$ входов и $m = 4$ выхода (см. табл. 3.2). Таблица 3.2 заполняется в соответствии с определением кодера. В таблице используется "отрицательная" логика.

Таблица 3.2. Таблица истинности неполного кодера, для которого $k = 10$ и $m = 4$

Н е т д/о	Перемен ная										Функц ии			
	x_9	x_8	x_7	x_6	x_5	x_4	x_3	x_2	x_1	x_0	F_3	F_2	F_1	F_0
0	1	1	1	1	1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	1	0	0	0	1
2	1	1	1	1	1	1	1	0	1	1	0	0	1	0
3	1	1	1	1	1	1	0	1	1	1	0	0	1	1
4	1	1	1	1	1	0	1	1	1	1	0	1	0	0
5	1	1	1	1	0	1	1	1	1	1	0	1	0	1
6	1	1	1	0	1	1	1	1	1	1	0	1	1	0
7	1	1	0	1	1	1	1	1	1	1	0	1	1	1
8	1	0	1	1	1	1	1	1	1	1	1	0	0	0
9	0	1	1	1	1	1	1	1	1	1	1	0	0	1

Из таблицы истинности 3.2 получаются следующие логические функции:

$$_3 F(x_9, x_8, x_7, x_6, x_5, x_4, x_3, x_2, x_1, x_0) = \bar{x}_9 \times x_8 \times x_7 \times x_6 \times x_5 \times x_4 \times x_3 \times x_2 \times x_1 \times x_0 +$$

$$+ x_9 \times \bar{x}_8 \times x_7 \times x_6 \times x_5 \times x_4 \times x_3 \times x_2 \times x_1 \times x_0 = \bar{x}_9 + \bar{x}_8 = \bar{x}_9 \times \bar{x}_8 \quad (3.9)$$

$$_2 F(x_9, x_8, x_7, x_6, x_5, x_4, x_3, x_2, x_1, x_0) = \bar{x}_7 \times \bar{x}_6 \times \bar{x}_4 \quad (3.10)$$

$$_1 F(x_9, x_8, x_7, x_6, x_5, x_4, x_3, x_2, x_1, x_0) = \bar{x}_7 \times \bar{x}_6 \times \bar{x}_2 \quad (3.11)$$

$$_0 F(x_9, x_8, x_7, x_6, x_5, x_4, x_3, x_2, x_1, x_0) = \bar{x}_9 \times \bar{x}_7 \times \bar{x}_6 \times \bar{x}_3 \quad (3.12)$$

На рис. 3.3 показана электрическая схема энкодера, построенная в соответствии с логическими функциями (3.9),..., (3.12). На рис. 3.3 используются следующие обозначения: +Vcc - источник сигнала; 9, ..., 0 - переключатели каналов электрических сигналов; F_3, \dots, F_0 - выходы энкодера или шина данных. Из диаграммы видно, что на выходах получается код 01112, если только на вход под номером 7 подать ноль.

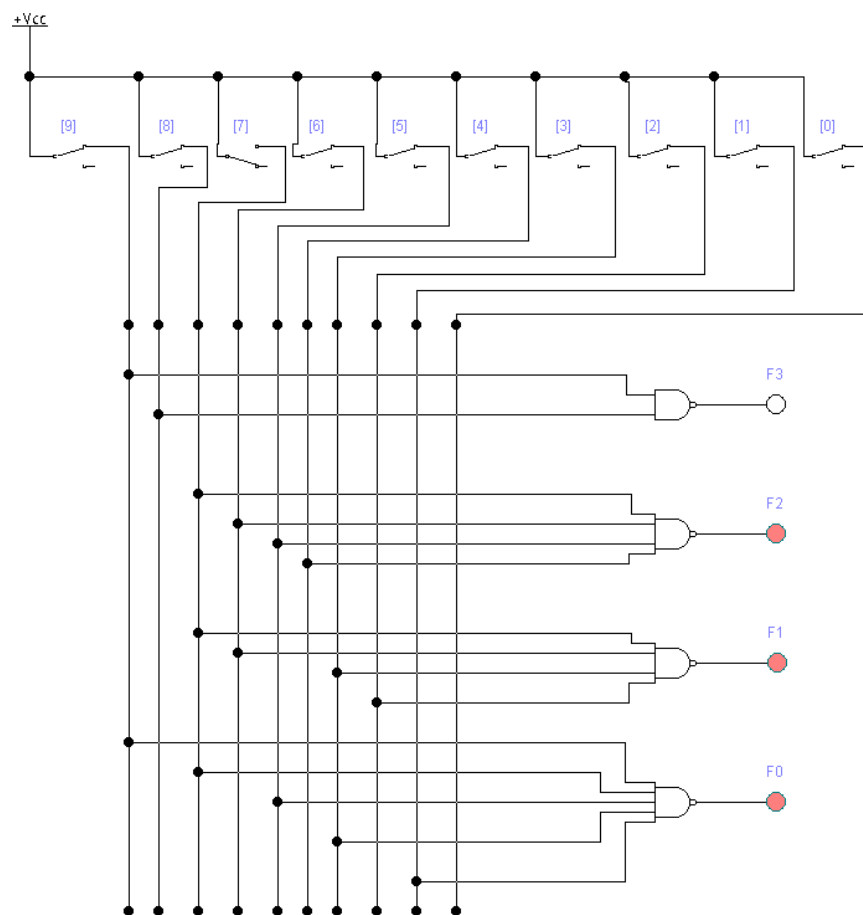


Рис. Электрическая схема энкодера.

Символ обычного кодера показан на рисунке 3.4.

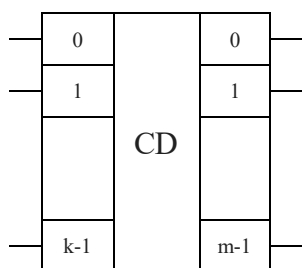


Рис. 3.4. Символ обычного кодера.

Кодер в основном используется для преобразования электрического сигнала в двоичный код. Энкодер является частью микропроцессорного блока управления, постоянной памяти и аналого-цифрового преобразователя (АЦП).

3.3. Переводчик кода

Транслятор кода строится из декодера и кодера (см. рис. 3.5) и используется для соединения устройств, имеющих шины данных разного порядка. Транслятор кода должен удовлетворять следующим условиям:

- количество входов меньше количества выходов или $k < n$;
- количество выходов декодера равно количеству входов кодера.

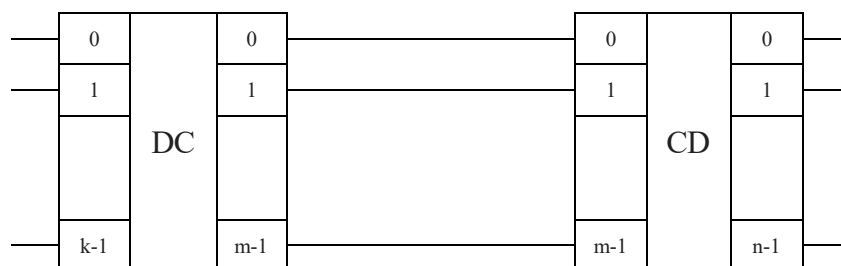


Рис. Блок-схема транслятора кода.

3.3. Мультиплексор

Определение. Мультиплексором мы называем комбинационное электронное устройство, используемое для передачи информации из n ($n > 1$) каналов по одному каналу.

Операция передачи информации из нескольких каналов в один канал может быть выполнена с помощью структурной модели, представленной на рис. 3.6.

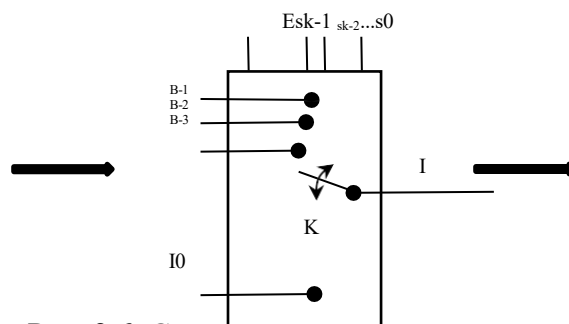


Рис. 3.6. Структурная модель мультиплексора.

На рис. 3.6 используются следующие обозначения: $I_{n-1}, I_{n-2}, I_{n-3}, \dots, I_0$ - шина данных на входе мультиплексора; E - канал управления (разрешает или запрещает передачу информации); $s_{k-1}, s_{k-2}, \dots, s_0$ - шина маршрутизации положения переключателя K ; K - переключатель; I - канал данных на выходе мультиплексора.

Структурная модель мультиплексора имеет следующие рабочие режимы:

- а) $E = 0$, мультиплексор простаивает (не передает информацию);
- б) $E = 1$, мультиплексор находится в рабочем режиме (передает информацию, поданную на один из входов, на выход в соответствии с соединением, выполненным переключателем K).

В качестве примера, будет построен мультиплексор с 4 входными каналами данных (также называемыми

"Состояния мультиплексора показаны в таблице истинности, где x может быть 0 или 1 (см. таблицу 3.3).

Таблица 3.3. Таблица истинности мультиплексора "четыре в одном

E	0	1	1	1	1
s_1	x	0	0	1	1
s_0	x	0	1	0	1
I	0	I_0	I_1	I_2	I_3

Из таблицы истинности мультиплексора получается следующая логическая функция

$$I = EI_0 s_1 \bar{s}_0 + EI_1 s_1 \bar{s}_0 + EI_2 s_1 + EI_3 s_1 s_0. \quad (3.13)$$

На рис. В 3.7 показана электрическая схема мультиплексора, построенная в соответствии с логической функцией (3.13). В электрической схеме используются следующие обозначения: I3, I2, I1, I0 - шина данных на входе мультиплексора; E - канал управления (разрешает ($E = 1$) или запрещает ($E = 0$) передачу информации); s1, s0 - каналы, отвечающие за коммутацию каналов шины данных I3, I2, I1, I0 на выходе I (при этом s1, s0 определяют адреса каналов шины данных); I - канал данных на выходе мультиплексора. В соответствии с электрической схемой на рис. 3.7 для $E = 1$, $s1s0 = 10$ только канал данных I2 соединен с выходом I.

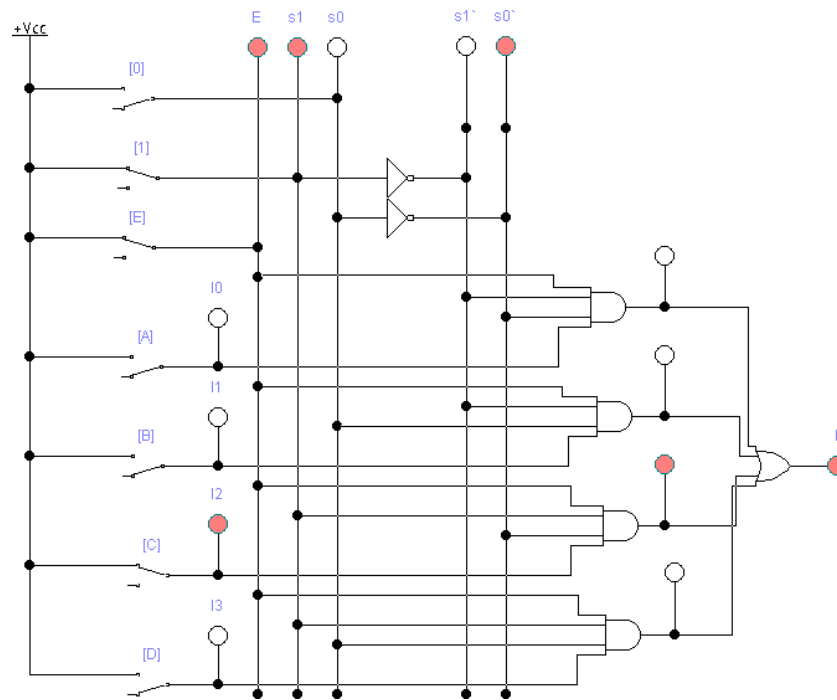


Рис. 3.7. Схема подключения мультиплексора.

Символ обычного мультиплексора показан на рисунке 3.8.

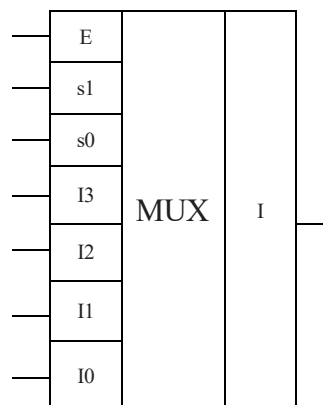


Рис. 3.8. Символ обычного мультиплексора.

Мультиплексор можно использовать для выполнения следующих операций:

- а) передача информации из нескольких каналов в один канал или мультиплексирование информации;

б) преобразование параллельного двоичного кода в последовательный двоичный код. В этом случае на входы подается параллельный двоичный код, который выдерживается длительное время и с помощью маршрутных кодов $sk-1, sk-2, \dots, s0$ и переключателя К входные каналы поочередно соединяются с выходным каналом (после завершения передачи кода на входы мультиплексора подается другой двоичный код);

с) для циклической генерации двоичных кодов - в этом случае на вход постоянно подается двоичный код, а положения переключателя К циклически изменяются под действием маршрутных сигналов $sk-1, sk-2, \dots, s0$.

3.4. Генератор мультиплексора

На рис. 3.9 показана электрическая схема мультиплексора, который может генерировать сигналы в цикле. Форма сигнала на выходе мультиплексора I определяется двоичным кодом, подаваемым на входы I3, I2, I1, I0. На электрической схеме имеются следующие обозначения: I3, I2, I1, I0 - шина данных на входе мультиплексора; E - канал управления (разрешает ($E = 1$) или запрещает ($E = 0$) передачу информации); s1, s0 - каналы, отвечающие за коммутацию каналов шины данных I3, I2, I1, I0 на выходе I (при этом s1, s0 определяют адреса каналов шины данных); G - переключатель, обеспечивающий переключение цикла формирования сигнала на выходе мультиплексора I; I - канал данных на выходе мультиплексора.

Для получения на выходе мультиплексора сигналов I в цикле необходимо переключить переключатели A, B, C, D в соответствии со значениями I0, I1, I2, I3, а затем соединить генератор и бистабильность с помощью переключателя G. Сигналы, формируемые бистабилями, меняются s1, s0 в цикле.

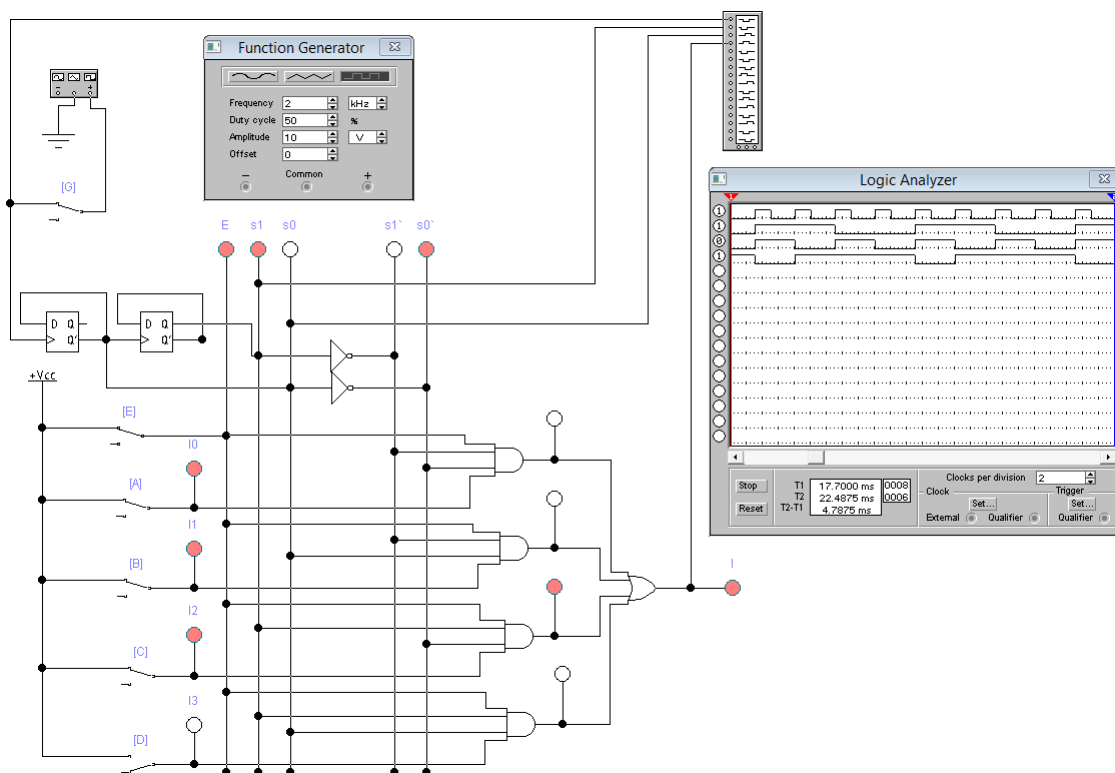


Рис. 3.9. Схема подключения генераторного мультиплексора.

В диаграмме на рис. 3.9 $E = 1$, а $I3 = 0$, $I2 = 1$, $I1 = 1$ и $I0 = 1$. Подстановка этих данных в формулу (3.13) дает логическую функцию сигналов на выходе мультиплексора I

$$I = s_1 \oplus s_0 + s_1 \bar{s}_0 + s_1 s_0 \quad (3.14)$$

Для других комбинаций I3, I2, I1, I0 на выходе мультиплексора получают другие функции I или другие сигналы.

3.5. Демультимплексор

Определение. Демультимплексор - это электронное комбинационное устройство, используемое для передачи информации из одного канала в несколько каналов.

Операция передачи информации из одного канала в несколько каналов может быть выполнена с помощью структурной модели, представленной на рис. 3.10.

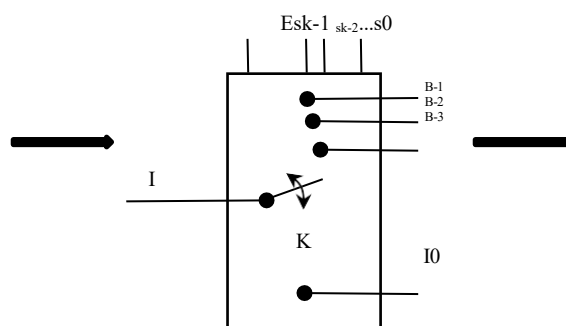


Рис. 3.10. Структурная модель демультимплексора.

На рис. 3.10 используются следующие обозначения: I - канал данных на входе демультимплексора;; E - канал управления (разрешает или запрещает передачу информации); $s_{k-1}, s_{k-2}, \dots, s_0$ - шина маршрутизации положения переключателя K; K - переключатель; $I_{n-1}, I_{n-2}, I_{n-3}, \dots, I_0$ - шина данных на выходе демультимплексора.

Структурная модель демультимплексора имеет следующие режимы работы:

- а) $E = 0$, демультимплексор находится в режиме ожидания (не передает информацию);
- б) $E = 1$, демультимплексор находится в рабочем режиме (передает информацию, поданную на вход, на один из выходов в соответствии с соединением, установленным переключателем K).

В качестве примера будет построен демультимплексор с 4 каналами данных на выходе (также называемый "один на четыре"). Состояния демультимплексора показаны в таблице истинности, где x может быть 0 или 1 (см. таблицу 3.4).

Таблица 3.4. Таблица истинности демультимплексора "один в четыре"

E	0	1	1	1	1
s1	x	0	0	1	1
s0	x	0	1	0	1
I3	0	0	0	0	1
I2	0	0	0	1	0
I1	0	0	1	0	0
I0	0	1	0	0	0

Из таблицы истинности мультиплексора получают следующие логические функции

$$I_3 = E I s_1 s_0 ; \quad (3.15)$$

$$I_2 = E I s_1 \bar{s}_0 ; \quad (3.16)$$

$$I_1 = E I \bar{s}_1 s_0 ; \quad (3.17)$$

$$I_0 = E I \bar{s}_1 \bar{s}_0 . \quad (3.18)$$

На рис. 3.11 показана электрическая схема демультимплексора, построенная в соответствии с логическими функциями (3.15), ..., (3.18). В электрической схеме используются следующие обозначения: I - шина данных, последовательная на вход мультиплексора; E - канал управления (разрешает ($E = 1$) или нет).

разрешает ($E = 0$) передачу информации); s_1, s_0 - каналы, отвечающие за переключение канала шины данных I на один из выходов I_3, I_2, I_1, I_0 (при этом s_1, s_0 определяют адреса каналов I_3, I_2, I_1, I_0). В соответствии с электрической схемой на рис. 3.11 для $E = 1$ и $s_1s_0 = 01$ канал данных I соединяется с выходом I_1 .

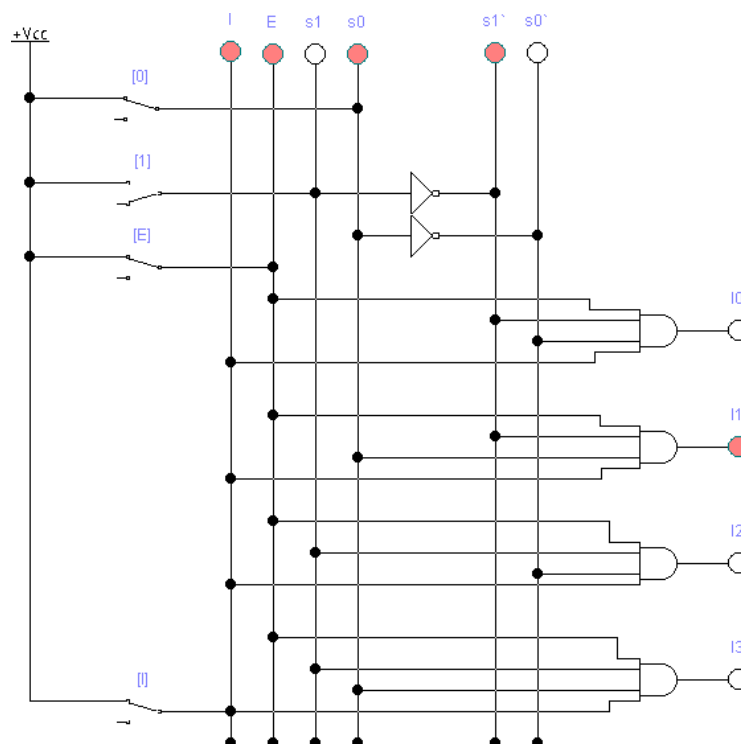


Рис. 3.11. Схема подключения демультиплексора.

Символ обычного демультиплексора показан на рисунке 3.12.

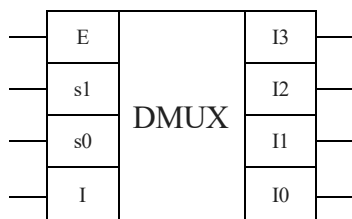


Рис. 3.12. Символ обычного демультиплексора.

Демультиплексор можно использовать для выполнения следующих операций:

- а) передача информации из одного канала в несколько каналов или демультиплексирование (фрагментация) информации;
- б) преобразование последовательного двоичного кода в псевдопараллельный двоичный код. В этом случае с помощью маршрутных кодов $s_{k-1}, s_{k-2}, \dots, s_0$ и коммутатора K выходные каналы поочередно присоединяются к входному каналу (после завершения передачи кода на вход демультиплексора подается другой двоичный код).

3.6. Компаратор

Определение. Мы называем компаратором электронное комбинационное устройство, используемое для сравнения двоичных кодов.

Цифровой компаратор или компаратор кодов - это логическое устройство с входами, на которые могут быть поданы два двоичных кода - $A(a_{n-1}, a_{n-2}, \dots, a_0)$ и $B(b_{n-1}, b_{n-2}, \dots, b_0)$. Обычно микросхемы компараторов имеют три двоичных выхода, на одном из которых выводится сигнал, полученный в результате сравнения кодов, поданных на входы:

- a) $A > B$;
- b) $A = B$;
- c) $A < B$.

Некоторые микросхемы компараторов имеют только выход $A = B$.

Компаратор сравнивает биты кода. В качестве примера в таблице 3.5 показаны состояния компаратора первого порядка, сравнивающего два бита $A = a_0$ и $B = b_0$.

Таблица 3.5. Таблица истинности компаратора первого порядка

Нес т д/о	Запис и		Вых оды		
	A	B	$A < B$	$A = B$	$A > B$
0	0	0	0	1	0
1	0	1	1	0	0
2	1	0	0	0	1
3	1	1	0	1	0

Из таблицы 3.5 мы получаем следующие логические функции:

$$F(A < B) = \bar{A} \times B; \quad (3.19)$$

$$F(A = B) = \bar{A} \times \bar{B} + A \times B; \quad (3.20)$$

$$F(A > B) = A \times \bar{B}. \quad (3.21)$$

По формулам (3.19), ..., (3.21) получаются следующие состояния:

- a) $F(A < B) = 1$, если $A = 0$ и $B = 1$;
- b) $F(A = B) = 1$, если $A = B$, и $F(A = B) = 0$, если $A \neq B$;
- c) $F(A > B) = 1$, если $A = 1$ и $B = 0$.

Зависимость между функциями $F(A = B)$, $F(A < B)$ и $F(A > B)$ можно получить через

преобразование формулы (3.20), используя аксиому $\bar{A} \times A = 0$ или $B \times B = 0$:

$$\begin{aligned} F(A = B) &= \bar{A} \times \bar{B} + A \times B = \bar{A} \times A + \bar{A} \times B + B \times B + A \times B = \\ &= \bar{A} \times A + A \times \bar{B} + \bar{A} \times B + B \times B = A \times (\bar{A} + B) + B \times (\bar{A} + B) = (\bar{A} + B) \times (A + B) = \\ &= A + B \times (A + B) = A + B + (A + B) = A \times B + A \times B \end{aligned}$$

или

$$F(A = B) = \bar{A} \times \bar{B} + A \times B = \bar{A} \times B + A \times \bar{B}. \quad (3.22)$$

На рис. В 3.13 показана электрическая схема компаратора первого порядка, построенная в соответствии с логической функцией (3.22).

Для сравнения двоичных кодов более высокого порядка компараторы могут быть построены из последовательно соединенных компараторов первого порядка или в соответствии с логическими функциями в таблицах истинности. Для иллюстрации в таблице 3.6 приведены состояния компаратора второго порядка, используемого для сравнения кодов $A(a_1, a_0)$ и $B(b_1, b_0)$.

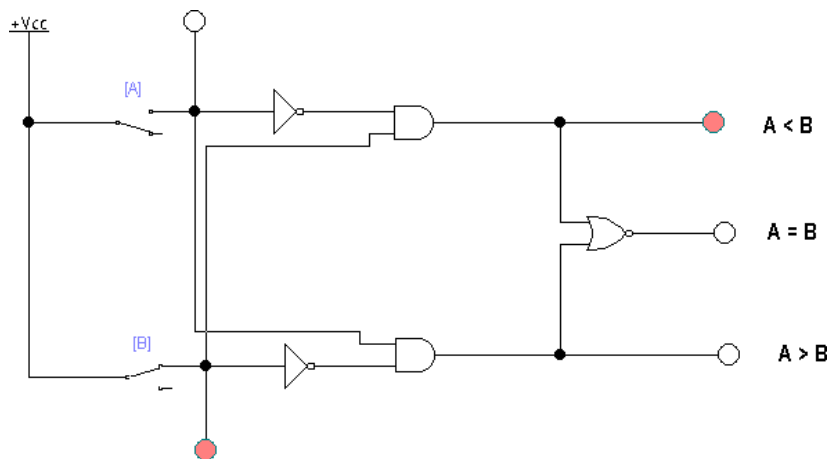


Рис. 3.13. Электрическая схема компаратора первого порядка.

Таблица 3.6. Состояния компаратора второго порядка

№ т д/о	Записи				Выходы		
	a_1	a_0	b_1	b_0	$A < B$	$A = B$	$A > B$
0	0	0	0	0	0	1	0
1	0	0	0	1	1	0	0
2	0	0	1	0	1	0	0
3	0	0	1	1	1	0	0
4	0	1	0	0	0	0	1
5	0	1	0	1	0	1	0
6	0	1	1	0	1	0	0
7	0	1	1	1	1	0	0
8	1	0	0	0	0	0	1
9	1	0	0	1	0	0	1
10	1	0	1	0	0	1	0
11	1	0	1	1	1	0	0
12	1	1	0	0	0	0	1
13	1	1	0	1	0	0	1
14	1	1	1	0	0	0	1
15	1	1	1	1	0	1	0

Из таблицы 3.6 мы получаем следующие логические функции:

$$F(A < B) = \sum(1, 2, 3, 6, 7, 11); \quad (3.23)$$

$$F(A = B) = \sum(0, 5, 10, 15); \quad (3.24)$$

$$F(A > B) = \sum(4, 8, 9, 12, 13, 14). \quad (3.25)$$

Функцию $F(A = B)$ можно упростить:

$$\begin{aligned}
 F(A = B) &= \bar{a}_1 \bar{a}_0 \bar{b}_1 \bar{b}_0 + \bar{a}_1 a_0 \bar{b}_1 b_0 + a_1 \bar{a}_0 b_1 \bar{b}_0 + a_1 a_0 b_1 b_0 = \\
 &= \bar{b}_1 (\bar{a}_1 \bar{b}_0 + a_1 b_0) + a_1 b_1 (\bar{a}_0 \bar{b}_0 + a_0 b_0) = \bar{b}_1 (\bar{a}_1 \bar{b}_0 + a_1 b_0) + a_1 b_1 (\bar{a}_0 \bar{b}_0 + a_0 b_0) = \\
 &= \bar{a}_1 \bar{b}_1 \oplus a_1 b_1 \oplus \bar{a}_0 \bar{b}_0 \oplus a_0 b_0 \\
 \text{или} \quad F(A = B) &= \bar{a}_1 \oplus \bar{b}_1 \oplus \bar{a}_0 \oplus \bar{b}_0 \quad (3.26)
 \end{aligned}$$

Функции $F(A < B)$ и $F(A > B)$ можно минимизировать методом Вейтча-Карнауха (см. рис. 3.14 и рис. 3.15).

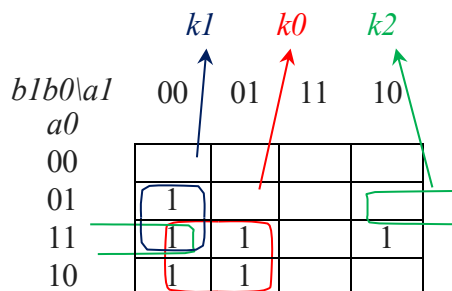


Рис. 3.14. Минимизация функции $F(A < B)$.

Из рис. 3.14 для $F(A < B)$ получен:

$$F(A < B) = k_2 + k_1 + k_0 = \bar{a}_1 b_1 + \bar{a}_1 \bar{a}_0 b_0 + \bar{a}_0 b_1 b_0. \quad (3.27)$$

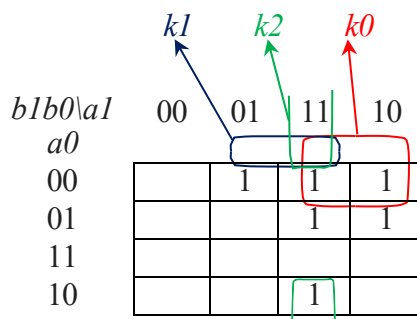


Рис. 3.15. Минимизация функции $F(A > B)$.

Из рис. 3.15 для $F(A > B)$ получен:

$$F(A > B) = k_2 + k_1 + k_0 = a_1 \bar{b}_1 + a_0 \bar{b}_1 \bar{b}_0 + a_1 a_0 \bar{b}_0. \quad (3.28)$$

На рис. 3.16 показана электрическая схема компаратора второго порядка, построенного в соответствии с логическими функциями (3.26), (3.27) и (3.28).

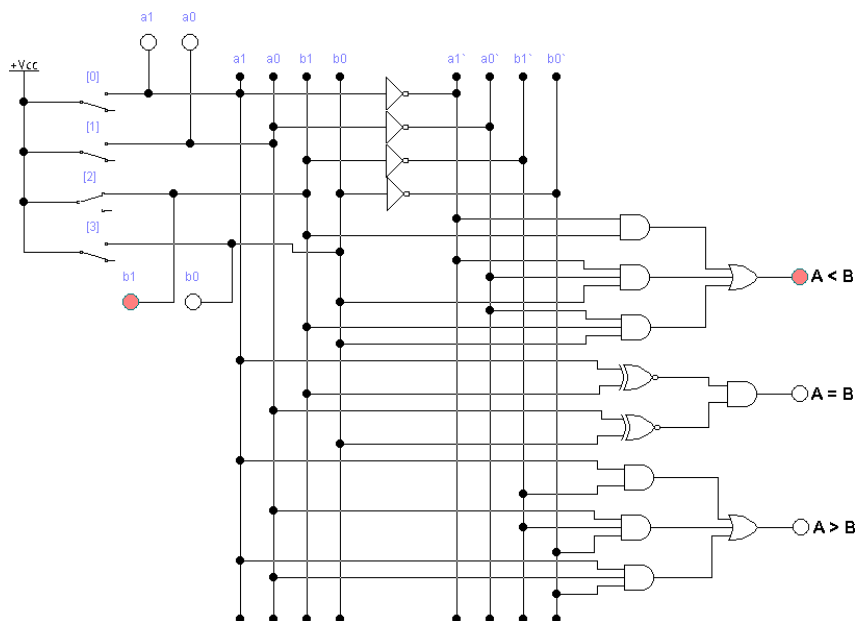


Рис. 3.16. Схема подключения компаратора второго порядка.

На рис. 3.16 показано состояние $A < B$, где $A(a_1, a_0) = 00$ и $B(b_1, b_0) = 10$.

Условное обозначение порядкового компаратора 4 показано на рисунке 3.17.

Входы и выходы $A < B$, $A = B$, $A > B$ используются для соединения компараторов между собой с целью формирования компараторных схем 8-го, 16-го и т.д. порядка.

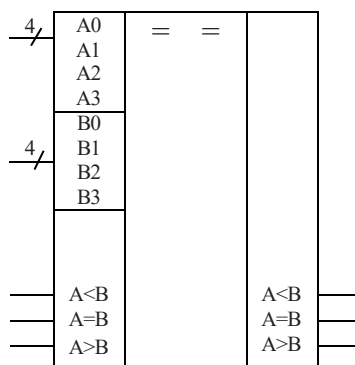


Рис. 3.17. Условное обозначение порядкового четверного компаратора.

Компараторы широко используются в компьютерах, цифровых измерительных приборах, радио- и кабельных сетях связи, бытовой технике. В качестве примера, цифровой будильник содержит цифровой компаратор, который инициирует звуковой сигнал, когда текущее время совпадает с заданным временем.

3.7. Определение и классификация лета

Определение. Мы называем суммирующим устройством электронное комбинационное устройство, используемое для суммирования кодов.

Резюме можно классифицировать по следующим критериям:

- а) в соответствии с архитектурными/конструктивными особенностями;
- б) в зависимости от того, как осуществляется процесс сборки кода;
- с) в соответствии с методом передачи перерасхода;
- д) в соответствии с используемой системой нумерации.

А. По архитектурным/конструктивным особенностям они бывают:

- **сумма по режиму** или логический элемент SAU-EX с двумя входами и одним выходом (на входы подается один двоичный разряд, а выход является результатом суммы по режиму);

- **полусуммирование** с двумя входами и двумя выходами (на входы подается один двоичный разряд, один выход дает результат суммирования по режиму, а два выхода - переполнение);

- **полный сумматор (ячейка сумматора)** с тремя входами и двумя выходами (на два входа подается двоичный разряд соответствующего порядка складываемых чисел, на третий вход подается переполнение младшего порядка, на одном выходе получается результат сложения в соответствии с режимом, а на втором выходе получается переполнение, которое передается в следующий старший порядок);

- **аккумуляторный отстойник**, имеющий собственную память.

В. В соответствии с тем, как осуществляется процесс суммирования кодов суммы

называются:

- **последовательный (серийный) сумматор**, в котором суммирование цифр производится последовательно, бит за битом

бит в каждом порядке чисел с помощью одного полного сумматора;

- **Параллельно-последовательный сумматор**, в котором биты пары чисел суммируются параллельно, а превышения переносятся последовательно;

- **параллельный сумматор**, в котором все пары цифр одинакового порядка и превышения складываются вместе.

С. В соответствии с **методом передачи перерасхода** существуют:

- [Ripple-carry](#) сумматор;

- Сумматоры с опережением переноса с параллельной или ускоренной групповой передачей превышений;
- Сумматор с пропуском переноса;
- Сумматор условных сумм;
- Сумматор с отбором по переносу;
- Сумматор с сохранением переноса.

D. В соответствии с используемой системой счисления сумматоры делятся на двоичные, троичные, четвертичные, восьмеричные, десятичные и шестнадцатеричные.

3.8. Полумассажер

Первый двоичный полусумматор "Model K Adder" на двух электромеханических реле (полусумматор Стибица) был разработан Джорджем Робертом Стибицем (1904 - 1995). Он является одним из всемирно признанных основателей современных цифровых компьютеров. Он работал исследователем в Bell Labs, известен своей работой в 1930-х и 1940-х годах по реализации булевой логики в цифровых схемах с использованием электромеханических реле в качестве переключающих элементов.

Определение. Мы называем полусумматором электронное комбинационное устройство, которое суммирует две двоичные цифры.

Результат арифметического суммирования для двух двоичных цифр показан в таблице 3.7, где:
 a, b - двоичные цифры, которые суммируются; C - переполнение; S - сумма по способу.

Таблица 3.7. Результат арифметического суммирования для двух двоичных цифр

Не т д/о	Запис и		Вых оды	
	a	b	C	S
0	0	0	0	0
1	0	1	0	1
2	1	0	0	1
3	1	1	1	0

Из таблицы 3.7 мы получаем следующие логические функции:

$$C = a \times b = \bar{a} \times \bar{b} = \bar{a} \bar{b}, \quad (3.29)$$

$$S = \bar{a} \times b + a \times \bar{b} = \bar{a} b + a \bar{b} = \bar{a} b + a \bar{b}. \quad (3.30)$$

Электрическая схема полусумматора, построенная по формулам (3.29) и (3.30), показана на рис. 3.18.

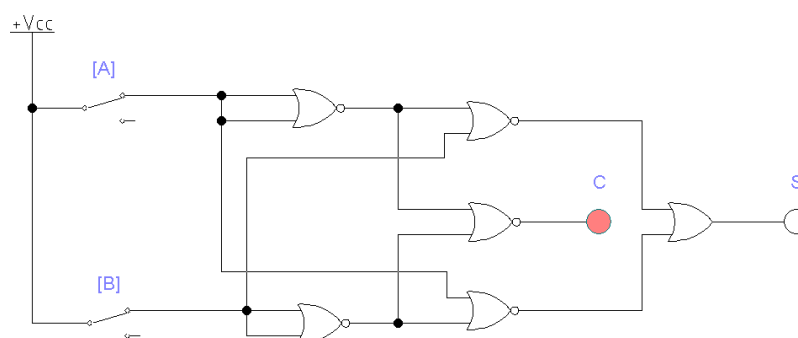


Рис. 3.18. Схема подключения полусумматора.

Хорошо известно, что в процессе сложения двух чисел в каждом порядке добавляются по две цифры, а переполненная цифра переносится из предыдущего более низкого порядка. Полусумматор нельзя использовать для построения сумматоров, так как он имеет только два входа и не может учитывать переполнение.

Обычный символ полусумматора показан на рисунке 3.19.

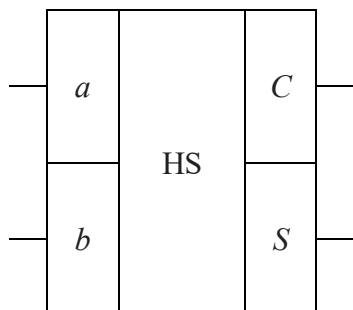


Рис. 3.19. Обычный символ полусумматора.

3.9. Полный сумматор

Определение. Мы называем полным сумматором электронное комбинационное устройство, которое складывает три двоичных разряда.

Из того, что представлено на странице 3.8, следует, что для построения сумм порядка $n \geq 2$ требуется элементарному суммирующему устройству с тремя входами. Состояния этого устройства, называемого полным сумматором, который складывает две двоичные цифры a_i , b_i и переполнение C_i , переданное из предыдущего более низкого порядка, приведены в таблице 3.8. В таблице 3.8 записаны C_{i+1} - переполнение для передачи в следующий более высокий порядок; S_i - сумма по способу; i - порядок.

Таблица 3.8. Полные состояния суммирующего устройства

Нес- т д/о	Запис и			Вых оды	
	C_i	a_i	b_i	C_{i+1}	S_i
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	1

Из таблицы 3.8 мы получаем следующие логические функции:

$$C_{i+1} = \bar{C}_i \bar{a}_i b_i + C_i \bar{a}_i b_i + C_i a_i \bar{b}_i + C_i a_i b_i; \quad (3.31)$$

$$S_i = \bar{C}_i \bar{a}_i b_i + C_i \bar{a}_i b_i + C_i a_i \bar{b}_i + C_i a_i b_i. \quad (3.32)$$

Логические функции (3.31) и (3.32) можно также представить следующим образом:

$$C_{i+1} = a_i b_i + C_i (a_i \oplus b_i); \quad (3.33)$$

$$S_i = C_i \oplus a_i \oplus b_i. \quad (3.34)$$

Логические функции (3.31) и (3.32) полного сумматора (суммирующей ячейки) выполняются по электрической схеме на рис. 3.20, построенный из двух полусумм.

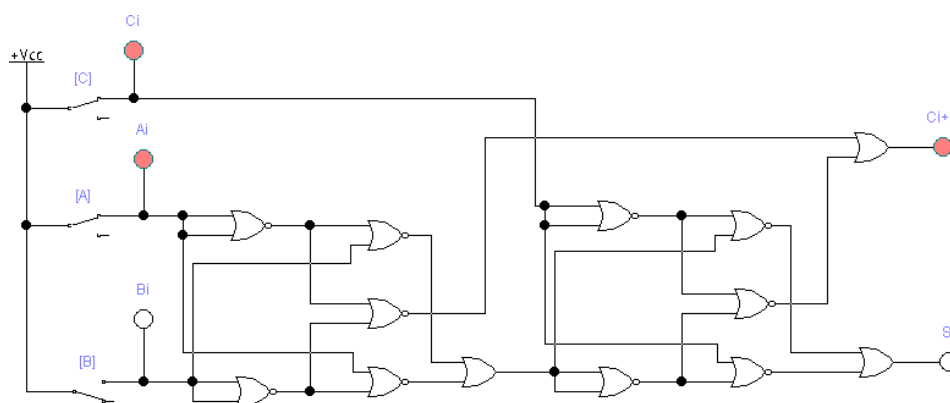


Рис. 3.20. Электрическая схема полного сумматора.

Условное обозначение полного сумматора показано на рисунке 3.21.

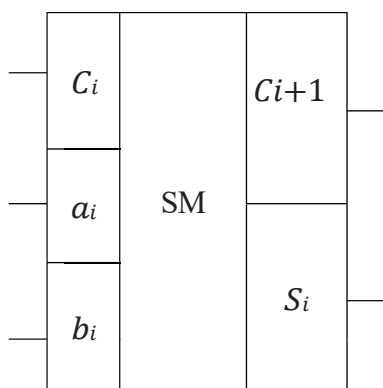


Рис. 3.21. Условное обозначение полного сумматора.

Полный сумматор используется для построения последовательных (серийных) сумматоров, параллельно-последовательных сумматоров, параллельных сумматоров и других устройств, выполняющих арифметические и/или логические операции.

3.10. Последовательный сумматор

Электрическая схема последовательного сумматора четвертого порядка показана на рис. 3.22.

Последовательный сумматор состоит из следующих устройств:

- три последовательных регистра прямого сдвига четвертого порядка для хранения чисел $A(a_3, a_2, a_1, a_0)$, $B(b_3, b_2, b_1, b_0)$ и результата их суммирования $S(s_3, s_2, s_1, s_0)$;
- полный сумматор SM, выполняющий операцию суммирования над числами $A(a_3, a_2, a_1, a_0)$ и $B(b_3, b_2, b_1, b_0)$;
- бистабильный D, необходимый для хранения превышений. Рабочий процесс последовательного сумматора включает в себя следующие шаги:
 - выключатель S отключен** - шаг записи чисел $A(a_3, a_2, a_1, a_0)$ и $B(b_3, b_2, b_1, b_0)$ в регистрах;
 - подключен переключатель S** - в этом случае процесс суммирования чисел и сохранения результата может быть выполнен путем подачи на входы набора командных

сигналов C

все регистры (суммирование, выполняемое одним полным SM сумматором, происходит в последовательном режиме, начиная с младшей битовой пары a_0, b_0 и превышая C_0).

Основным недостатком последовательного сумматора является то, что временной интервал, необходимый для выполнения операции суммирования, включает набор тактовых циклов, количество которых прямо пропорционально порядку суммируемых чисел. Время, необходимое для суммирования чисел порядка n , можно рассчитать по формуле:

$$T_{sum}(n) = n \times (T_s + T_d), \quad (3.35)$$

где T_{sum} - интервал времени, необходимый для выполнения операции суммирования чисел, n - порядок суммируемых чисел, T_s - интервал времени, необходимый для суммирования пары битов a_i и b_i , и T_d - интервал времени, необходимый для передачи переполнения с выхода C_0 на вход C_i полного сумматора бистабильным D (здесь C_0 и C_i показаны так, как отмечено в программе Electronics Workbench).

Параллельные сумматоры используются для сокращения времени, необходимого для выполнения операции суммирования.

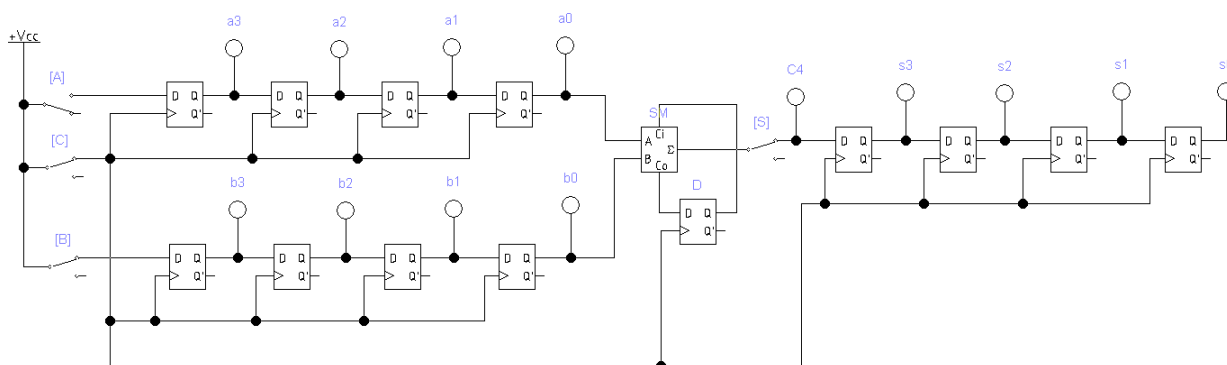


Рис. 3.22. Электрическая схема последовательного сумматора.

Условное обозначение сумматора четвертого порядка показано на рис. 3.23.

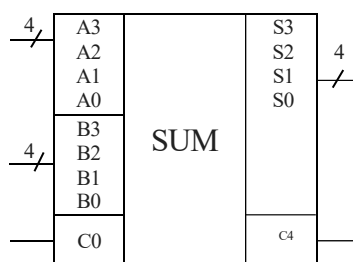


Рис. 3.23. Условное обозначение сумматора четвертого порядка.

3.11. Параллельный сумматор с последовательной передачей переполнений

Параллельный сумматор с последовательной передачей переборов строится из полных сумматоров, соединенных последовательно. Например, на рис. 3.22 показана электрическая схема последовательного сумматора четвертого порядка. На электрической схеме приведены следующие обозначения:

0 - переключатель канала обгона C_0 ; 2, 4, 6 и 8 - переключатели каналов номеров

$A(a_3, a_2, a_1, a_0)$; 1, 3, 5 и 7 - количество переключателей каналов $B(b_3, b_2, b_1, b_0)$;

4 C, C_3, C_2, C_1, C_0 - индикаторы переполнения; s_3, s_2, s_1, s_0 - индикаторы результата суммирования по режимам.

В случае параллельного сумматора с последовательной передачей переполнений пары цифр a_i, b_i , где $i = 0, 1, \dots, n-1$, подаются на входы сумматора одновременно, и суммирование этих цифр также происходит одновременно, но правильный результат суммирования в каждом порядке получается только после прихода

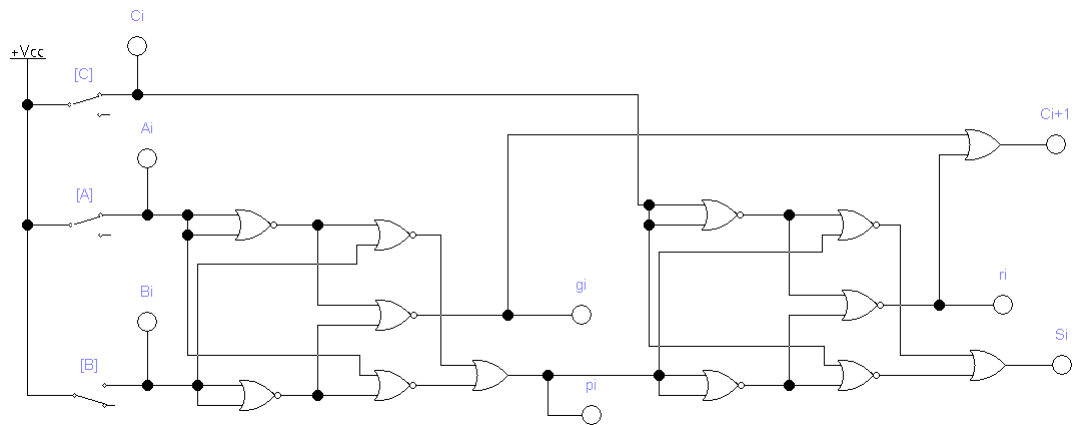


Рис. 3.23. Электрическая схема полного сумматора порядка "i".

В электрической схеме, показанной на рис. 3.23, являются следующие обозначения: +Vcc - источник напряжения; C, A, B - переключатели; a_i , b_i и C_i - каналы подачи пары однопорядковых цифр числа и перебора предыдущего младшего порядка; g_i , p_i - перебор и суммирование по режиму цифр a_i , b_i , полученных полусуммой первой ступени полного сумматора; r_i - перебор, полученный полусуммой второй ступени полного сумматора в процессе суммирования C_i и p_i ; C_{i+1} s_i - перебор и сумма по режиму разрядов a_i , b_i и C_i .

Состояния, реализованные в схеме полного сумматора порядка "i", показаны в таблице 3.9.

Таблица 3.9. Реализованные состояния в схеме полного сумматора порядка "i"

№ д/о	Состояние входа			Промежуточные состояния			Условия на выходах	
	C_i	a_i	b_i	g_i	p_i	r_i	C_{i+1}	s_i
0	0	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	1
2	0	1	0	0	1	0	0	1
3	0	1	1	1	0	0	1	0
4	1	0	0	0	0	0	0	1
5	1	0	1	0	1	1	1	0
6	1	1	0	0	1	1	1	0
7	1	1	1	1	0	0	1	1

Для полного сумматора (см. параграф 3.9) была получена следующая логическая функция:

$$C_{i+1} = a_i b_i + C_i (a_i \oplus b_i). \quad (3.37)$$

Из таблицы 3.9 видно, что

$$g_i = a_i b_i, \quad (3.38)$$

$$p_i = a_i \oplus b_i \quad (3.39)$$

или для C_{i+1} , выраженного через g_i и p_i , получается:

$$C_{i+1} = g_i + p_i C_i. \quad (3.40)$$

Формула (3.40) дает следующий алгоритм расчета превышений C_1, C_2, C_3, \dots для $i = 0, 1, 2, 3, \dots$:

$$\begin{aligned}
i = 0 &\rightarrow C_1 = g_0 + p_0 C_0; \\
i = 1 &\rightarrow C_2 = g_1 + p_1 C_1 = g_1 + p_1 (g_0 + p_0 C_0) = g_1 + p_1 g_0 + p_1 p_0 C_0; \\
i = 2 &\rightarrow C_3 = g_2 + p_2 C_2 = g_2 + p_2 (g_1 + p_1 C_1) = g_2 + p_2 g_1 + p_2 p_1 C_1 = \\
&= g_2 + p_2 g_1 + p_2 p_1 (g_0 + p_0 C_0) = g_2 + p_2 g_1 + p_2 p_1 g_0 + p_2 p_1 p_0 C_0; \\
i = 3 &\rightarrow C_4 = g_3 + p_3 C_3 = g_3 + p_3 (g_2 + p_2 C_2) = g_3 + p_3 g_2 + p_3 p_2 C_2 = \\
&= g_3 + p_3 g_2 + p_3 p_2 (g_1 + p_1 C_1) = g_3 + p_3 g_2 + p_3 p_2 g_1 + p_3 p_2 p_1 C_1 = \\
&= g_3 + p_3 g_2 + p_3 p_2 g_1 + p_3 p_2 p_1 (g_0 + p_0 C_0) = \\
&= g_3 + p_3 g_2 + p_3 p_2 g_1 + p_3 p_2 p_1 g_0 + p_3 p_2 p_1 p_0 C_0.
\end{aligned}$$

Алгоритм расчета перерегулирования можно продолжить и для $i = 4, 5, 6, 7, \dots$ Из формул (3.38), (3.39) и рис. 3.23 полного сумматора мы видим, что промежуточные размеры

ig , pi определяются первой полусуммой и их значения зависят только от a_i и b_i . Отсюда следует, что все проскакивания $C_1, C_2, C_3, C_4, \dots$ могут быть вычислены одновременно, если известны промежуточные величины g_i, p_i , где $i = 0, 1, 2, 3, \dots$

В соответствии с представленным выше алгоритмом построена электрическая схема параллельного суммирующего автомата с параллельной (групповой ускоренной) передачей перебора (см. рис. 3.24).

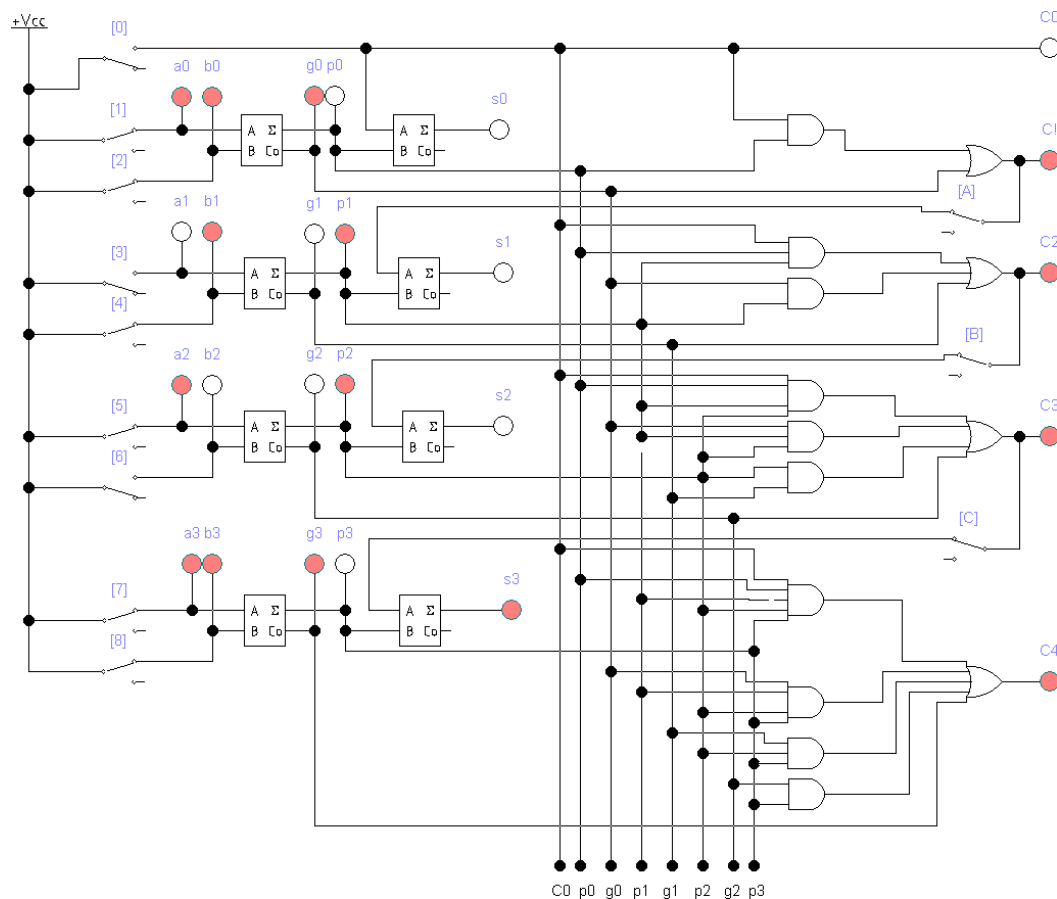


Рис. 3.24. Схема подключения параллельного суммирующего автомата с параллельным переносом перегрузки.

В электрической схеме, показанной на рис. 3.24, приведены следующие обозначения: $+V_{dc}$ - источник напряжения; 0 - переключатель канала перенапряжения C_0 ; 1, 3, 5, 7 - переключатели номерной шины.

$A(a_3, a_2, a_1, a_0)$; 2, 4, 6, 8 - количество выключателей шины $B(b_3, b_2, b_1, b_0)$; A, B, C - выключатели.

Суммирование чисел $A(a_3, a_2, a_1, a_0)$ и $B(b_3, b_2, b_1, b_0)$ происходит в три этапа:

1) $A(a_3, a_2, a_1, a_0)$, $B(b_3, b_2, b_1, b_0)$ и полусумматоры на первой ступени сумматора одновременно определяют все промежуточные величины g_i, p_i , где $i = 0, 1, 2, 3$, подаются на входы сумматора;

2) схема, построенная из элементов И, ИЛИ, называемая схемой параллельного переноса, используя только промежуточные величины g_i, p_i , параллельно определяет все переборы C_4, C_3, C_2, C_1 ;

3) превышения C_4, C_3, C_2, C_1 применяются к записям А полусумм второго шага, определяющих суммы в соответствии с s_3, s_2, s_1, s_0 .

Результат суммирования чисел $A(a_3, a_2, a_1, a_0)$ и $B(b_3, b_2, b_1, b_0)$ определяется C_4, s_3, s_2, s_1, s_0 или

$$A(a_3, a_2, a_1, a_0) + B(b_3, b_2, b_1, b_0) = S(C_4, s_3, s_2, s_1, s_0). \quad (3.41)$$

В электрической схеме, показанной на рис. 3.24 показывает случай суммирования чисел

$A(a_3, a_2, a_1, a_0) = 10012$ и $B(b_3, b_2, b_1, b_0) = 11012$, в результате что это получает $S(C_4, s_3, s_2, s_1, s_0) = 101102$.

Время, необходимое для сложения чисел, не зависит от их порядка и может быть рассчитано по формуле:

$$T_{sum} = T_1 + T_2 + T_3, \quad (3.42)$$

где T_{sum} - интервал времени, необходимый для выполнения операции суммирования двух чисел, T_1 - интервал времени, необходимый для одновременного определения промежуточных величин g_i, p_i , где $i = 0, 1, 2, 3$ полусумматорами первой ступени сумматора, T_2 - интервал времени, необходимый для определения переборов по схеме параллельного переноса, и

3 T - интервал времени, необходимый для вычисления конечного результата с использованием полусуммы второй ступени сумматора.

Параллельный сумматор с параллельным переносом переборов получает суммы чисел за гораздо меньший промежуток времени по сравнению с промежутком времени, необходимым для суммирования в случае последовательного сумматора или параллельного сумматора с последовательным переносом переборов.

Основным недостатком параллельного сумматора с параллельным переносом переборов является увеличение сложности электрической схемы по мере увеличения порядка суммируемых чисел.

Условное обозначение сумматора четвертого порядка показано на рис. 3.23.

Для суммирования чисел более высокого порядка параллельные сумматоры с параллельной передачей переполнения подключаются последовательно или параллельно.