



# Цифровые устройства

## *II. Комбинационные логические устройства*

**Комбинационная схема** – логическая схема, сигнал на выходе которой определяется только уровнями сигналов на ее входах. Такой подход построения схем называется *комбинационной логикой*.

*ИВ*

$I_0$	<b>C</b>	
$I_1$		
$I_2$		
$I_3$		
$I_4$		
$I_5$		$F_2$
$I_6$		$F_1$
$I_7$		$F_0$

**1. Шифратор** – устройство, которое преобразует входной позиционный код в выходной двоичный.

*C* – coder



## Таблица истинности шифратора

Номер позиции								Выходной код		
$I_7$	$I_6$	$I_5$	$I_4$	$I_3$	$I_2$	$I_1$	$I_0$	$F_2$	$F_1$	$F_0$
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

$$F_2 = \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0}$$

$$F_1 = \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0}$$

$$F_0 = \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0} + \overline{I_7 I_6 I_5 I_4 I_3 I_2 I_1 I_0}$$



## Приоритетный шифратор

Вход				Выходной код	
$I_3$	$I_2$	$I_1$	$I_0$	$F_1$	$F_0$
0	0	0	1	0	0
0	0	1	X	0	1
0	1	X	X	1	0
1	X	X	X	1	1

$I_3$	$I_2$	$I_1$	$F_1$	$F_0$
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

$$F_0 = \overline{I_3} \overline{I_2} I_1 + I_3 \overline{I_2} \overline{I_1} + I_3 \overline{I_2} I_1 + I_3 I_2 \overline{I_1} + I_3 I_2 I_1$$

$$F_1 = \overline{I_3} I_2 \overline{I_1} + \overline{I_3} I_2 I_1 + I_3 \overline{I_2} \overline{I_1} + I_3 \overline{I_2} I_1 + I_3 I_2 \overline{I_1} + I_3 I_2 I_1$$

$F_0$	$\overline{I_2} \overline{I_1}$	$I_2 \overline{I_1}$	$I_2 I_1$	$\overline{I_2} I_1$	$I_3$
	1	1	1	1	
	0	0	0	1	$\overline{I_3}$

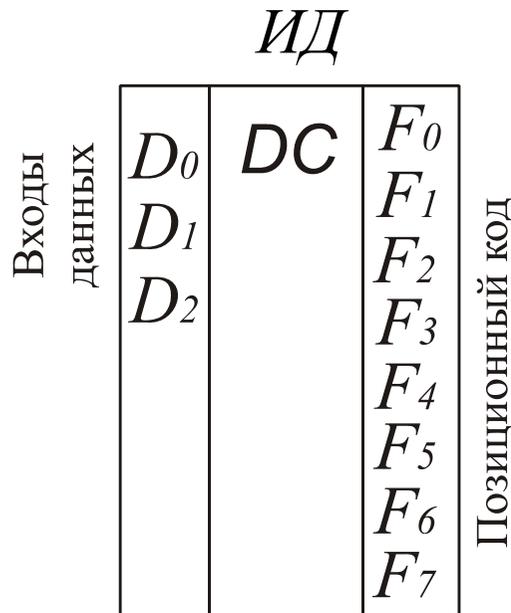
$F_1$	$\overline{I_2} \overline{I_1}$	$I_2 \overline{I_1}$	$I_2 I_1$	$\overline{I_2} I_1$	$I_3$
	1	1	1	1	
	0	1	1	0	$\overline{I_3}$

После минимизации:  $F_0 = I_3 + I_1 \overline{I_2}$

$F_1 = I_3 + I_2$



## 2. Дешифратор – устройство, которое преобразует ВХОДНОЙ ДВОИЧНЫЙ В ВЫХОДНОЙ ПОЗИЦИОННЫЙ КОД.



DC – decoder

$$n = 2^i$$

Входной код			Состояния на выходах							
$x_2$	$x_1$	$x_0$	$F_0$	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$	$F_7$
0	0	0	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

В любой момент времени дешифратор активизирует только *один* из  $n$  выходов. Выбор осуществляется при помощи управляющих сигналов, поступающих на входы  $D_0$ - $D_i$ .



## 2. Дешифратор – устройство, которое преобразует ВХОДНОЙ ДВОИЧНЫЙ В ВЫХОДНОЙ ПОЗИЦИОННЫЙ КОД.

$$F_0 = \overline{x_2 x_1 x_0}$$

$$F_1 = \overline{x_2} x_1 x_0$$

$$F_3 = x_2 \overline{x_1} x_0$$

$$F_3 = x_2 x_1 \overline{x_0}$$

$$F_4 = x_2 \overline{x_1} \overline{x_0}$$

$$F_5 = x_2 x_1 \overline{x_0}$$

$$F_6 = x_2 x_1 x_0$$

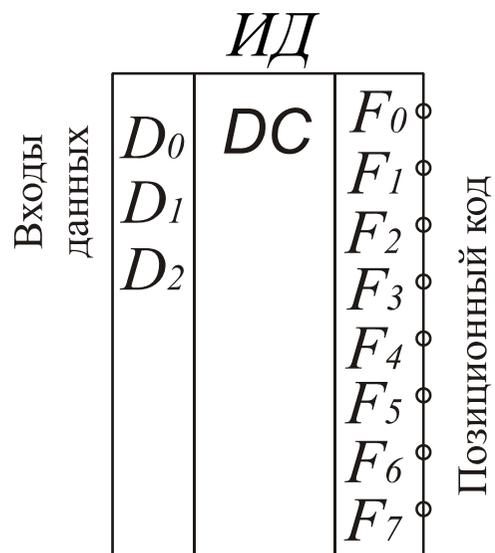
$$F_7 = x_2 x_1 x_0$$

Входной код			Состояния на выходах							
$x_2$	$x_1$	$x_0$	$F_0$	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$	$F_7$
0	0	0	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

В любой момент времени дешифратор активизирует только *один* из  $n$  выходов. Выбор осуществляется при помощи управляющих сигналов, поступающих на входы  $D_0$ - $D_i$ .



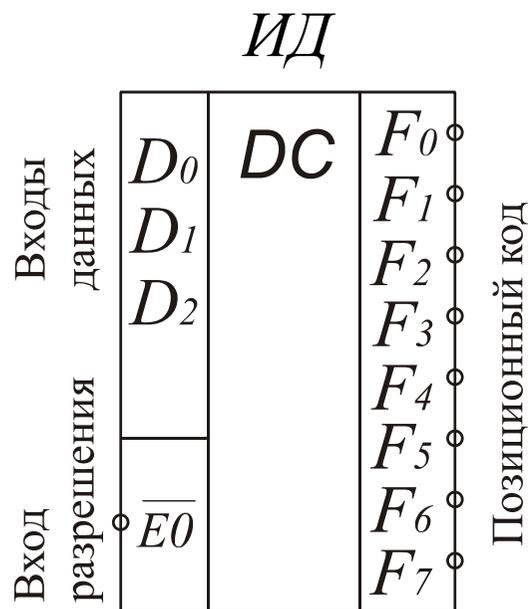
## Дешифратор с активным низким уровнем напряжения



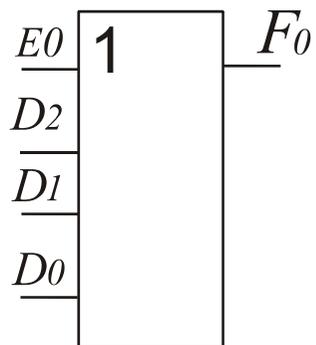
Входной код			Состояния на выходах							
$x_2$	$x_1$	$x_0$	$F_0$	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$	$F_7$
0	0	0	0	1	1	1	1	1	1	1
0	0	1	1	0	1	1	1	1	1	1
0	1	0	1	1	0	1	1	1	1	1
0	1	1	1	1	1	0	1	1	1	1
1	0	0	1	1	1	1	0	1	1	1
1	0	1	1	1	1	1	1	0	1	1
1	1	0	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	0



## Дешифратор с входом разрешения



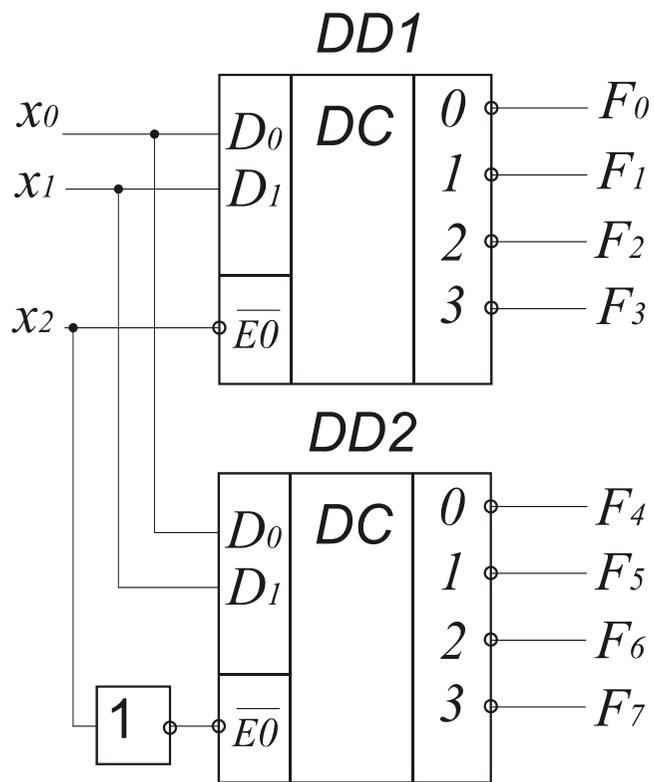
Входной код				Состояния на выходах							
$E_0$	$x_2$	$x_1$	$x_0$	$F_0$	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$	$F_7$
1	x	x	x	1	1	1	1	1	1	1	1
0	0	0	0	0	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1
0	0	1	1	1	1	1	0	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1	1
0	1	0	1	1	1	1	1	1	0	1	1
0	1	1	0	1	1	1	1	1	1	0	1
0	1	1	1	1	1	1	1	1	1	1	0



$$F_0 = D_2 + D_1 + D_0 + E_0$$



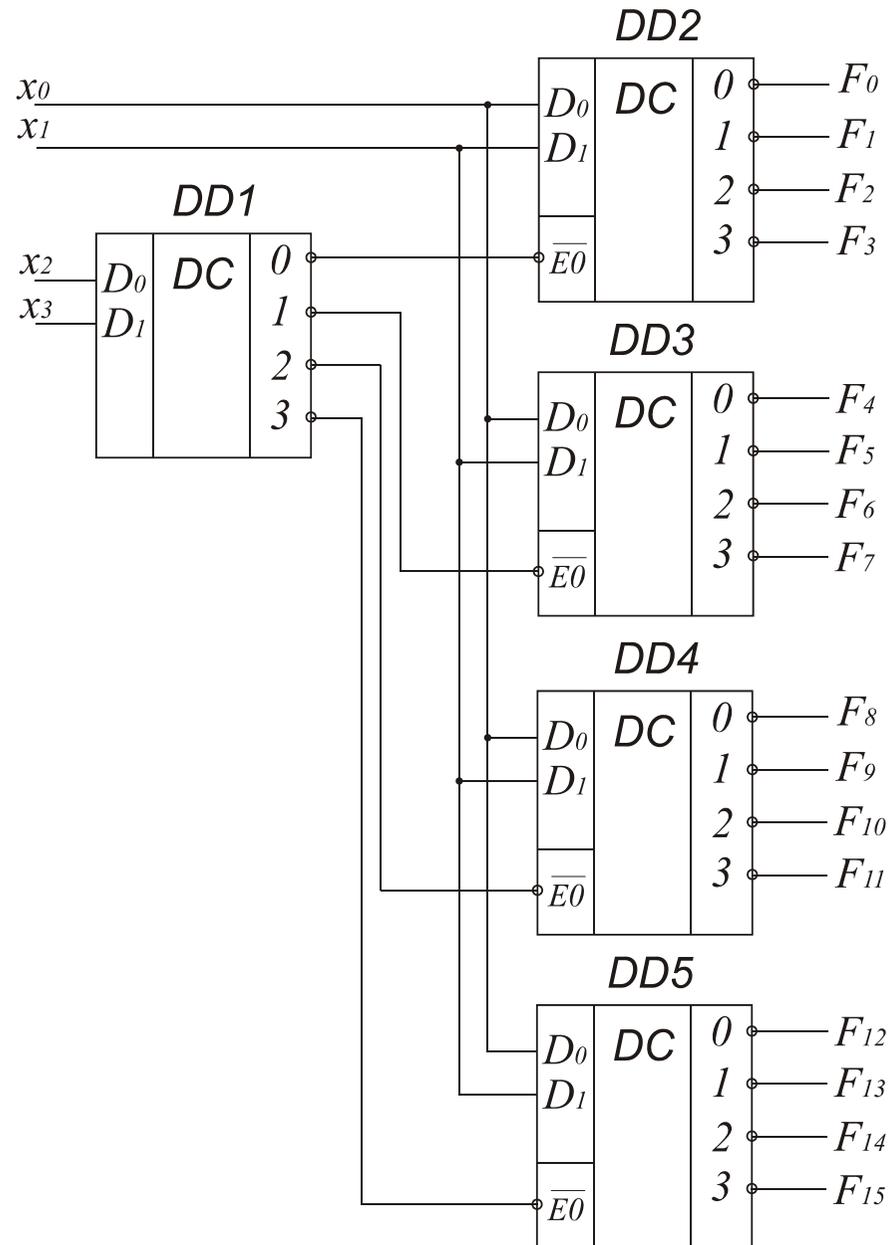
## Увеличение разрядности дешифратора



		Входной код			Состояния на выходах							
		$x_2$	$x_1$	$x_0$	$F_0$	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$	$F_7$
DD1	0	0	0	0	0	1	1	1	1	1	1	1
	0	0	1	1	1	0	1	1	1	1	1	1
	0	1	0	0	1	1	0	1	1	1	1	1
	0	1	1	1	1	1	1	0	1	1	1	1
DD2	1	0	0	0	1	1	1	1	0	1	1	1
	1	0	1	1	1	1	1	1	1	0	1	1
	1	1	0	0	1	1	1	1	1	1	0	1
	1	1	1	1	1	1	1	1	1	1	1	0
					DD1				DD2			



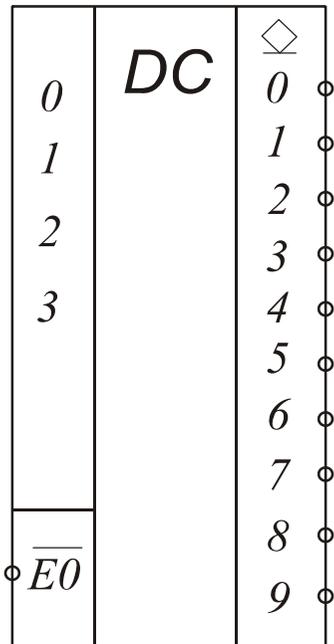
## Увеличение разрядности дешифратора





## Неполные дешифраторы

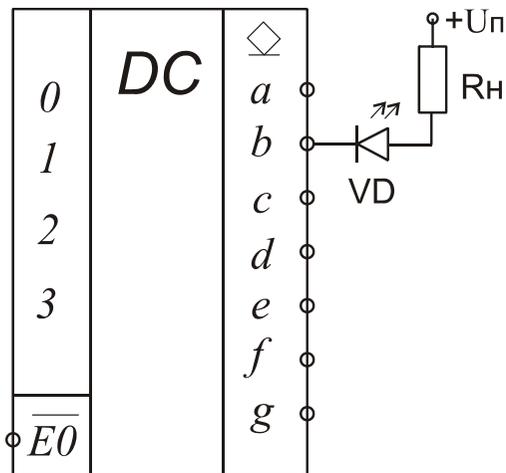
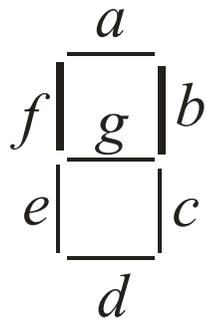
**Дешифратор 1:10** - преобразует двоично-десятичный код в десятичное число. Часто с открытым коллектором, и используется для управления газоразрядными индикаторами.



Входной код				Состояния на выходах									
$x_3$	$x_2$	$x_1$	$x_0$	$F_0$	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$	$F_6$	$F_7$	$F_8$	$F_9$
0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1	1	1
0	0	1	1	1	1	1	0	1	1	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1	1	1	1
0	1	0	1	1	1	1	1	1	0	1	1	1	1
0	1	1	0	1	1	1	1	1	1	0	1	1	1
0	1	1	1	1	1	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0

# Неполные дешифраторы

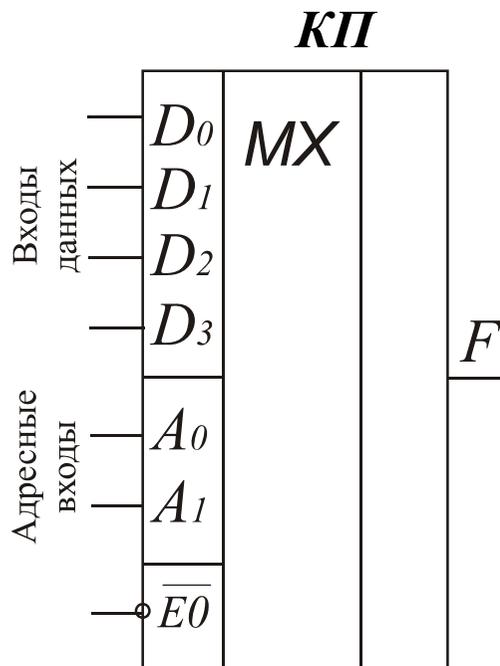
**Семисегментные дешифраторы** используются для преобразования двоичного кода в семисегментный код. Семисегментный код необходим для отображения на цифровых семисегментных индикаторах. Выходы дешифратора, как правило, с открытым коллектором.



Входной код					Состояния на выходах						
<i>dec</i>	$x_3$	$x_2$	$x_1$	$x_0$	$F_a$	$F_b$	$F_c$	$F_d$	$F_e$	$F_f$	$F_g$
0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	1	1	0	0	1	1	1	1
2	0	0	1	0	0	0	1	0	0	1	0
3	0	0	1	1	0	0	0	0	1	1	0
4	0	1	0	0	1	0	0	1	1	0	0
5	0	1	0	1	0	1	0	0	1	0	0
6	0	1	1	0	0	1	0	0	0	0	0
7	0	1	1	1	0	0	0	1	1	1	1
8	1	0	0	0	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0	1	0	0



### 3. Мультиплексор – схема выборки с электронным управлением.



**Мультиплексор** подключает *один* из  $n$  входных сигналов к *единственной* выходной линии. Выбор подключаемого входа осуществляется при помощи адреса, передаваемого по специальным линиям. Мультиплексоры иногда называют *селекторами данных*.

Обозначение на схеме: ***MX, MS, MUX***

$A_1$  и  $A_0$  – входные линии адреса,  
 $D_3$ ,  $D_2$ ,  $D_1$  и  $D_0$  – входные  
информационные линии.  
 $E_0$  – сигнал разрешения

$E_0$	$A_1$	$A_0$	$F$
1	x	x	0
0	0	0	$D_0$
0	0	1	$D_1$
0	1	0	$D_2$
0	1	1	$D_3$

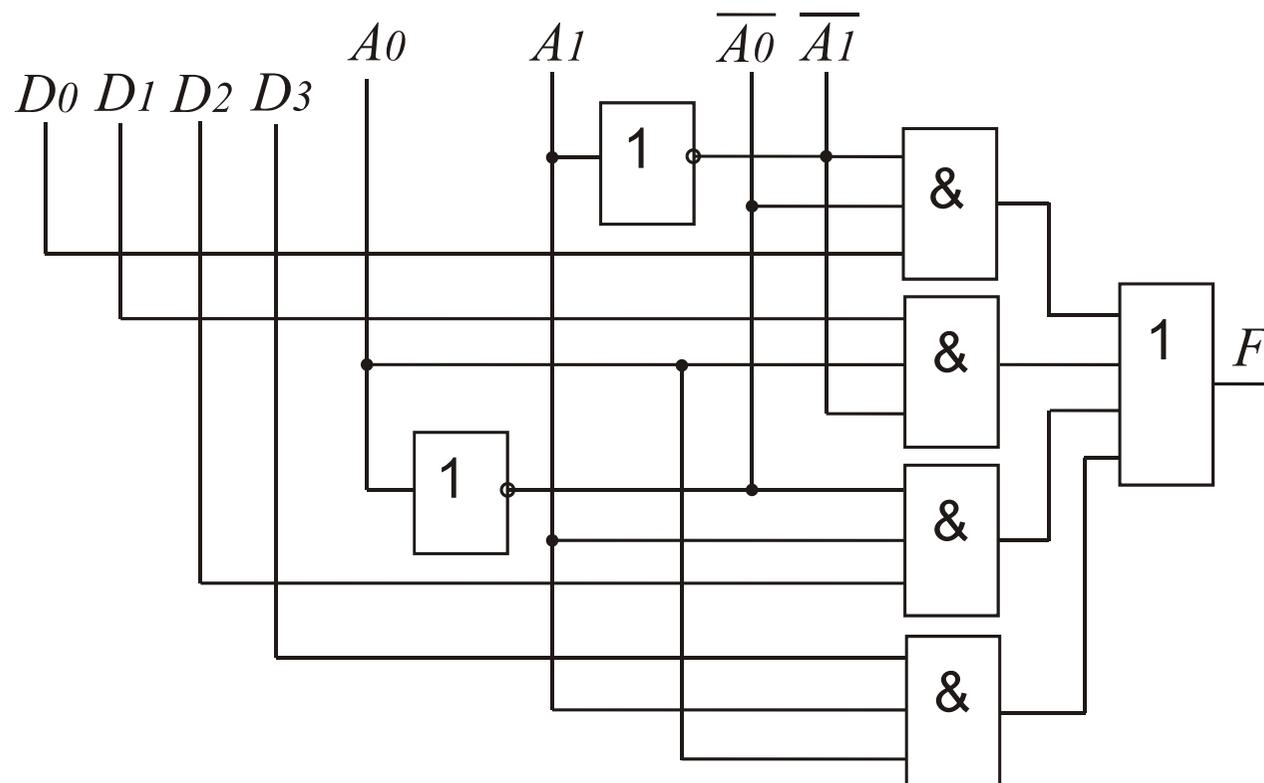
При наличии активного разрешающего сигнала  $E$  на адресные линии подается двоичный код адреса. При этом на выход  $F$  будет копироваться информация с выбранного в соответствии с этим адресом информационного входа  $D_i$ .

$$F = D_0 \overline{A_1} \overline{A_0} + D_1 \overline{A_1} A_0 + D_2 A_1 \overline{A_0} + D_3 A_1 A_0$$



## Схема мультиплексора

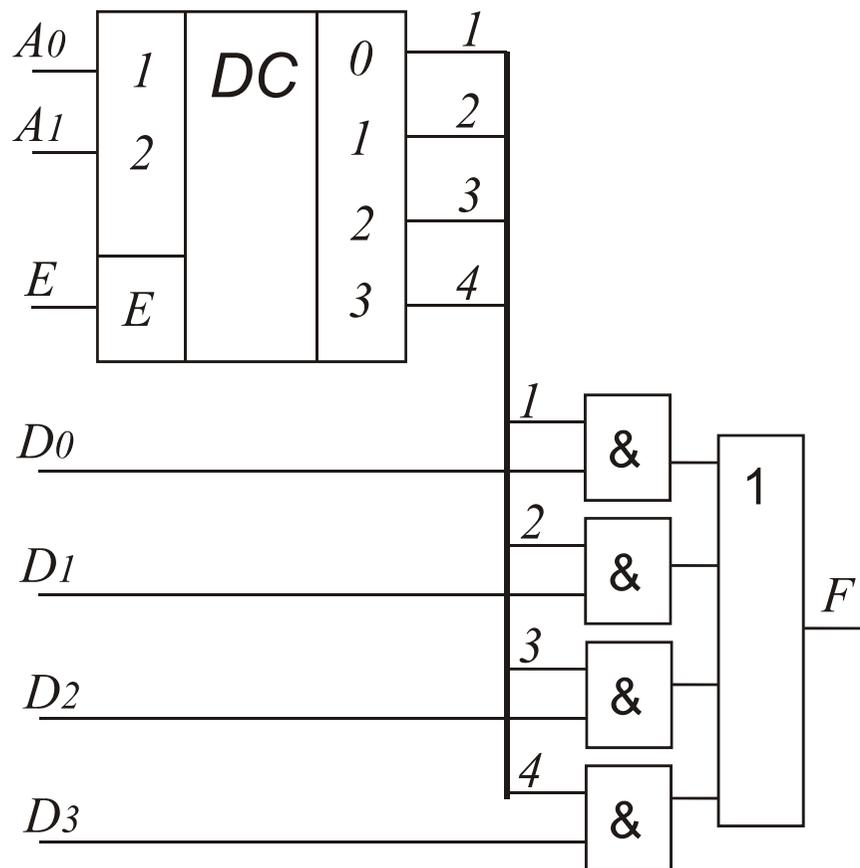
$$F = D_0 \overline{A_1} \overline{A_0} + D_1 \overline{A_1} A_0 + D_2 A_1 \overline{A_0} + D_3 A_1 A_0$$





## Схема мультиплексора

$$F = D_0 \overline{A_1} \overline{A_0} + D_1 \overline{A_1} A_0 + D_2 A_1 \overline{A_0} + D_3 A_1 A_0$$



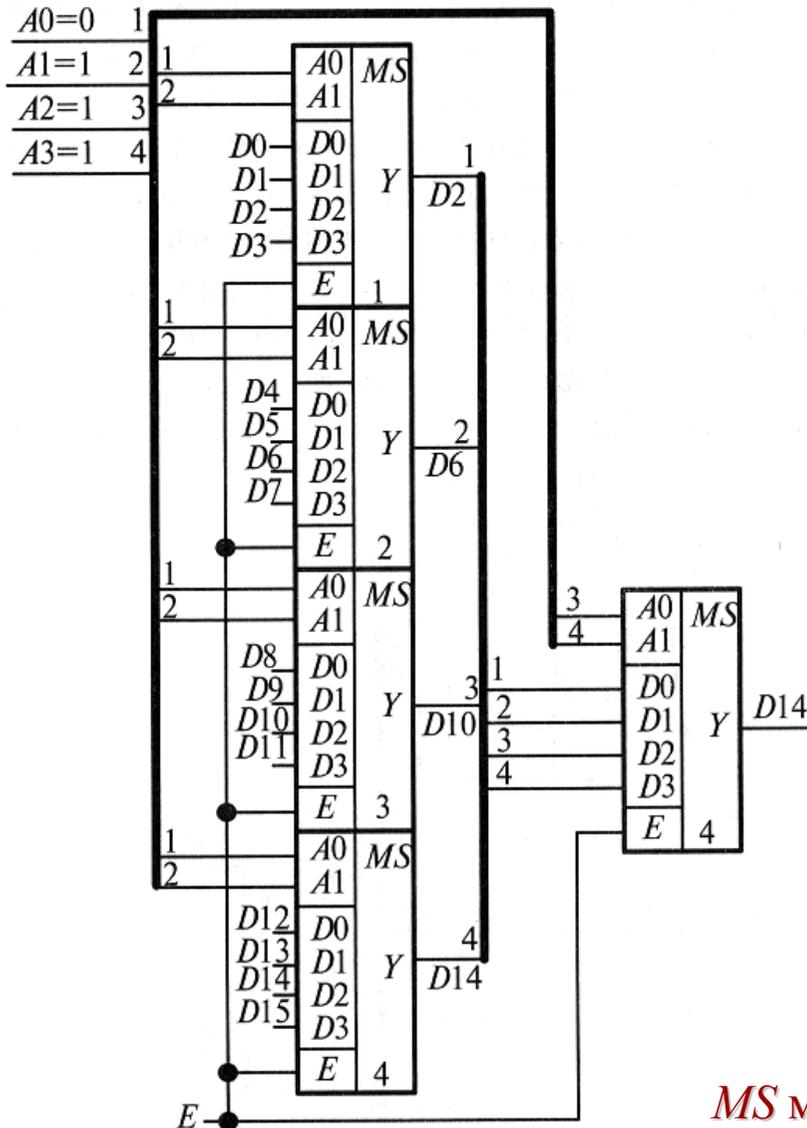
Дешифратор  $DC$  осуществляет выбор информационного входа  $D_1...D_0$  в соответствии с адресом  $A_1A_0$ . Он обеспечивает логическую единицу на входе логического элемента  $И$ , соответствующего адресу выбранного информационного входа. При наличии разрешающего сигнала  $E=1$  состояние этого логического элемента  $И$  определяется информацией на выбранном информационном входе. Так, при  $A_1A_0 = 11_2 = 3_{10}$  на выход  $F$  будет передаваться информация с входа  $D_3$ .



# Методы увеличения разрядности мультиплексора

## 1. Каскадное соединение мультиплексоров

(последовательное соединение).



Для выбора одного информационного входа из 16 необходимы 4 входные линии адреса:  $A_3$ ,  $A_2$ ,  $A_1$  и  $A_0$ . Четыре мультиплексора 1-го каскада обеспечивают выбор в зависимости от кода, поданного на  $A_1$  и  $A_0$ , соответственно: первый - одного из сигналов  $D_0$ ,  $D_1$ ,  $D_2$  или  $D_3$ , второй - из сигналов  $D_4$ ,  $D_5$ ,  $D_6$  или  $D_7$ , третий - из сигналов  $D_8$ ,  $D_9$ ,  $D_{10}$  или  $D_{11}$ , четвертый - из сигналов  $D_{12}$ ,  $D_{13}$ ,  $D_{14}$  или  $D_{15}$ . Пятый  $MS$  (2-й каскад) обеспечивает выбор одного из ранее выбранных сигналов в зависимости от кода, подаваемого на  $A_3$  и  $A_2$ .

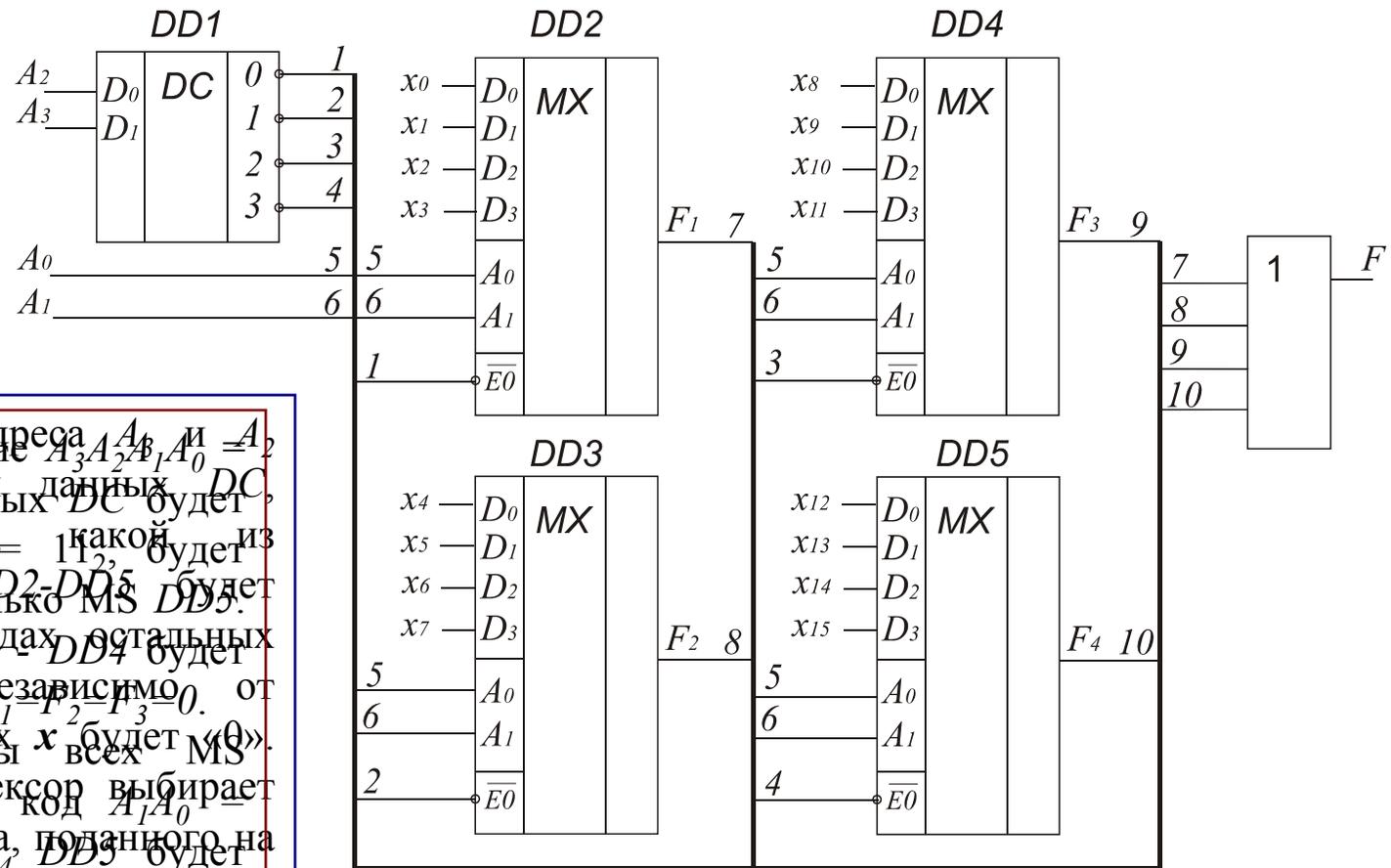
**Например**, при  $A_3A_2A_1A_0 = 1110_2$  на выходы  $MS$  первого каскада будут поступать  $D_2$ ,  $D_6$ ,  $D_{10}$  и  $D_{14}$ . На выход всей схемы  $D_{14}$ .

*MS* могут не иметь входа разрешения *E*.



# Методы увеличения разрядности мультиплексора

## 1. Параллельное соединение мультиплексоров (реализуется с помощью DC)



Старшие разряды адреса  $A_3, A_2, A_1, A_0 = A_2, A_1, A_0$  подаются на входы данных DC, который выбирает, какой из мультиплексоров DD2-DD5 будет разрешена работа только MS DD3. На выходах MS DD2-DD4 будет мультиплексоров независимо от логический «0», т.е.  $F_1 = F_2 = F_3 = 0$ . На адресные входы MS. Активный мультиплексор выбирает поступает двоичный код  $A_1 A_0$  в зависимости от кода, поданного на  $A_2, A_1$  на выход  $F_4$  DD5 будет передаваться информация с входа один из сигналов  $x_0, x_1$  второй — из  $D_3$ , т.е. значение переменной  $x_{14}$ . После суммирования на выходе всей схемы будет  $F = F_1 + F_2 + F_3 + F_4 = x_0 + x_1 + 0 + x_{14} = x_{14}$ . Сигналы с выходов MS суммируются.

**MS должны иметь вход разрешения**

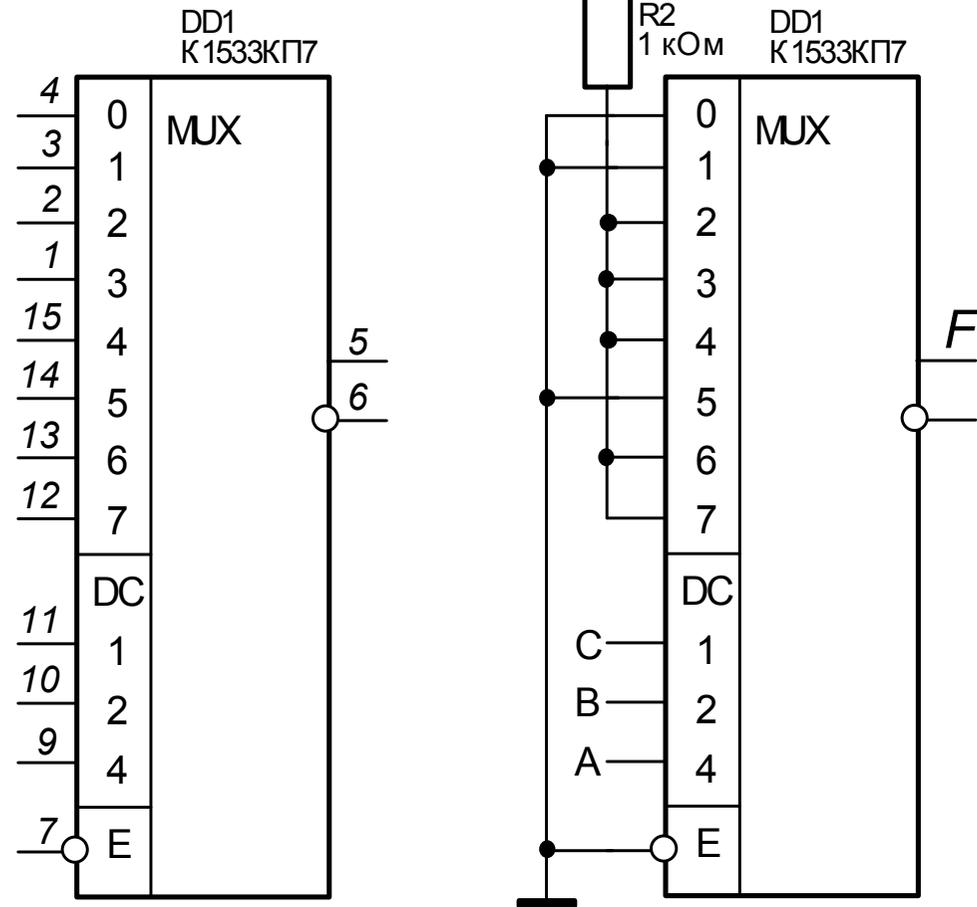


## Реализация логических функций с помощью мультиплексора

### 1. Функция 3-х переменных

<i>A</i>	<i>B</i>	<i>C</i>	<i>F</i>
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

**Пример.** Реализовать функцию, используя мультиплексор КП7.

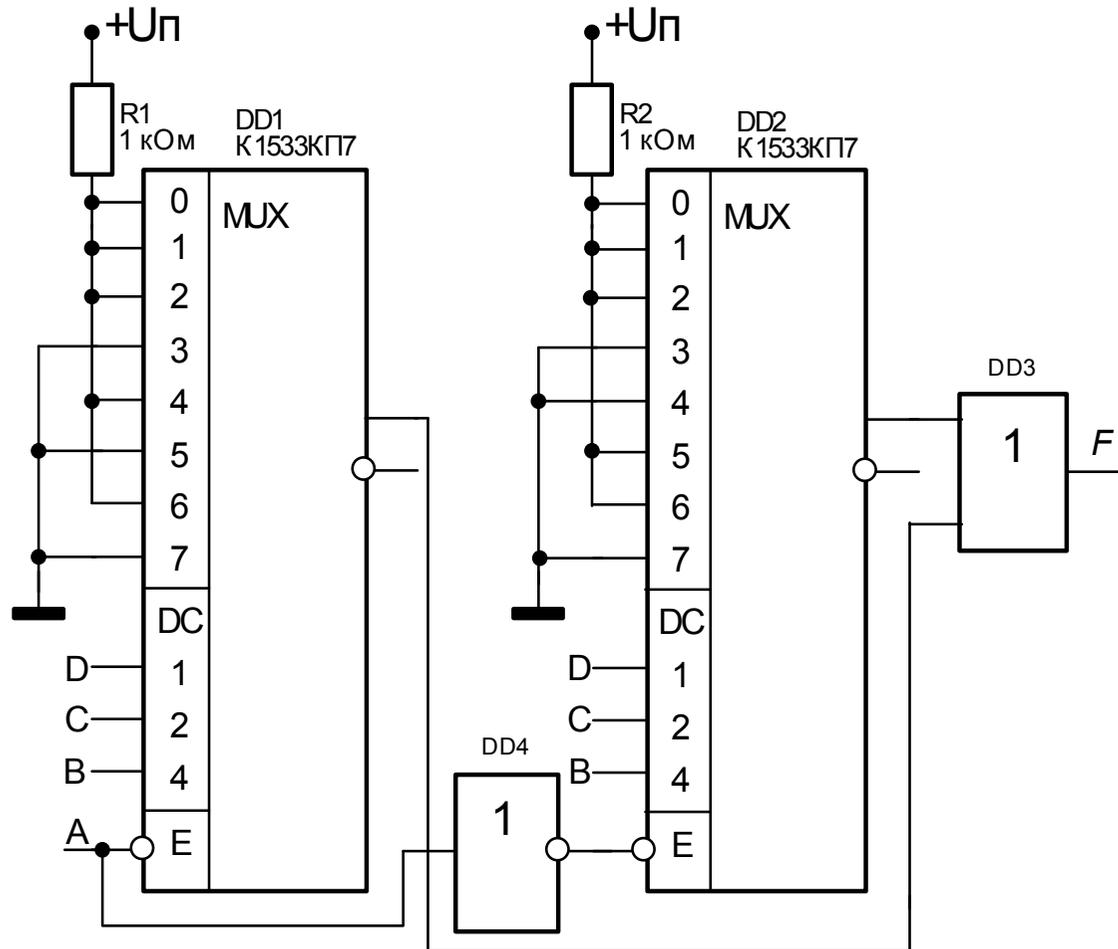




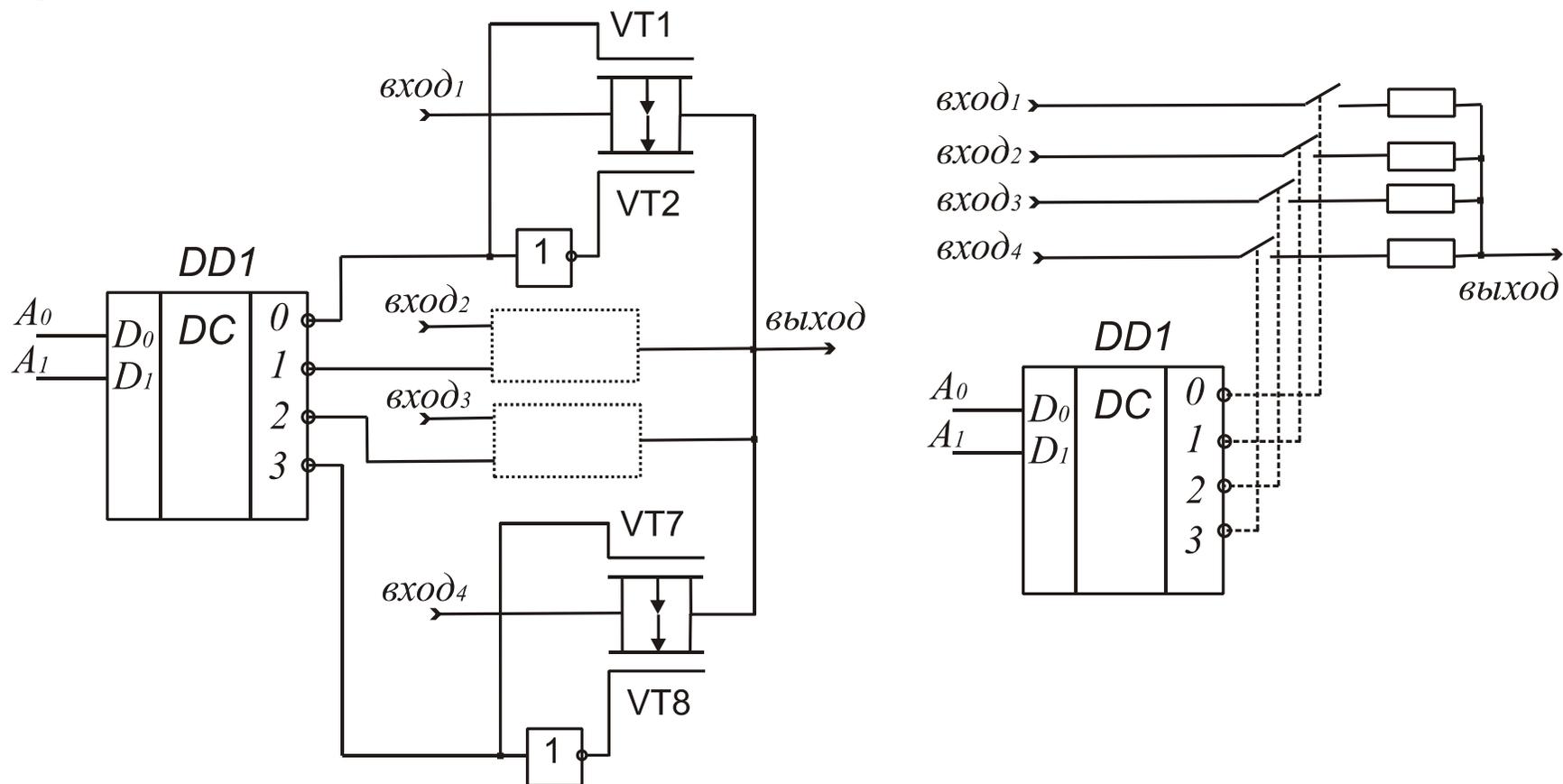
# Реализация логических функций с помощью мультиплексора

## 2. Функция 4-х переменных

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>f</i>
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0



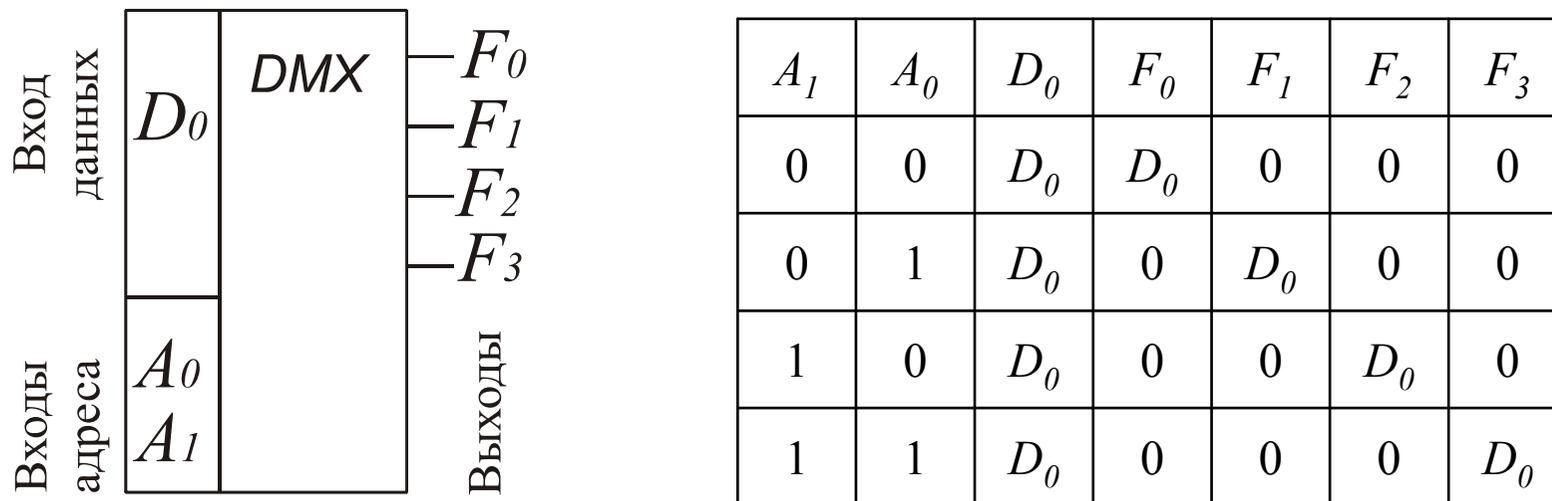
## Схема мультиплексора КМОП



Схемотехника мультиплексоров КМОП отличается тем, что они могут коммутировать аналоговые сигналы с размахом амплитуды до  $U_p$ . Пара транзисторов VT1-VT2 и др. обеспечивает одинаковое сопротивление в обоих направлениях, т.е. сигнал может распространяться, как от входа к выходу, так и от выхода к входу. В этом случае мультиплексор может использоваться в качестве демультимплексора.



**4. Демультимплексор** – устройство, функционально противоположное мультиплексору. Он подключает *единственный* входной сигнал к одному из  $n$  выходов, определяемому адресом.

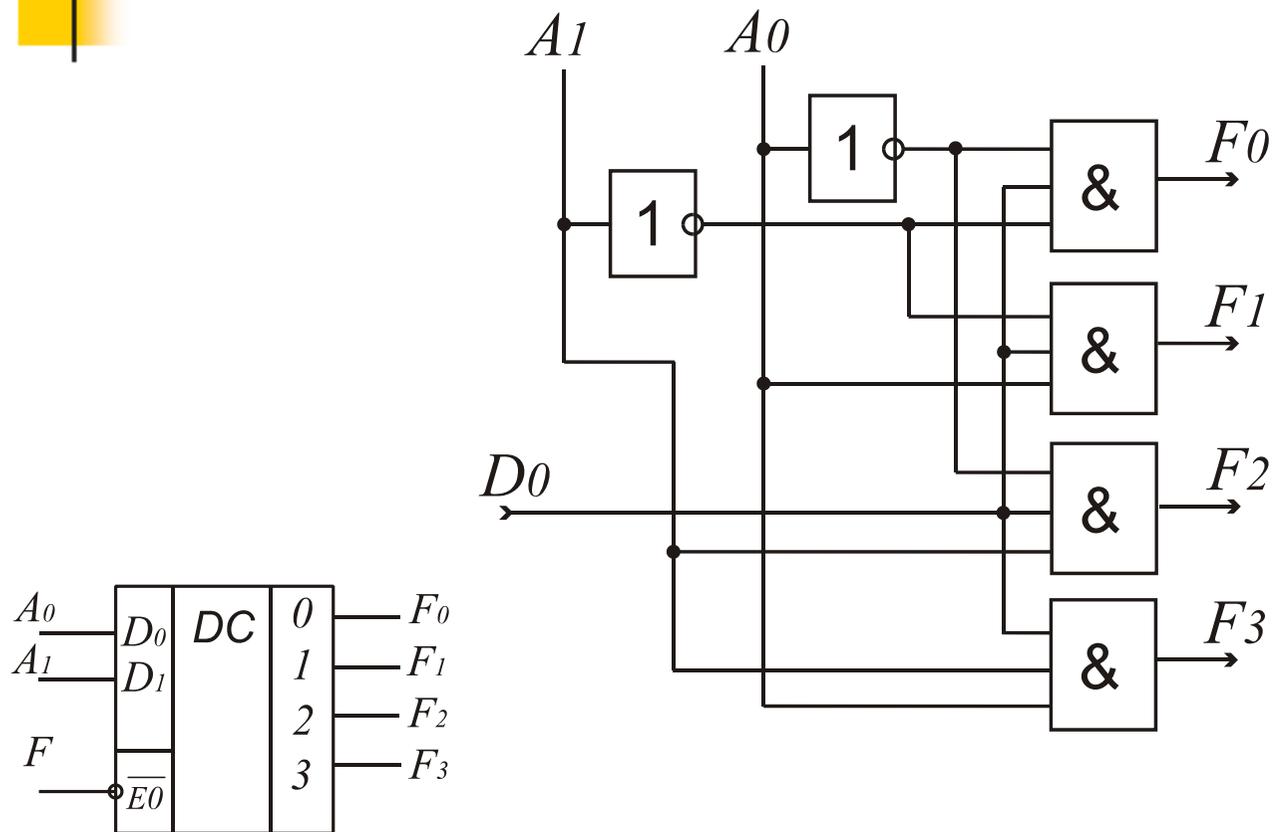


$$F_0 = D_0 \overline{A_1} \overline{A_0} \quad F_2 = D_0 A_1 \overline{A_0}$$

$$F_1 = D_0 \overline{A_1} A_0 \quad F_3 = D_0 A_1 A_0$$



## Схема демультиплексора



$$F_0 = D_0 \overline{A_1} \overline{A_0}$$

$$F_1 = D_0 \overline{A_1} A_0$$

$$F_2 = D_0 A_1 \overline{A_0}$$

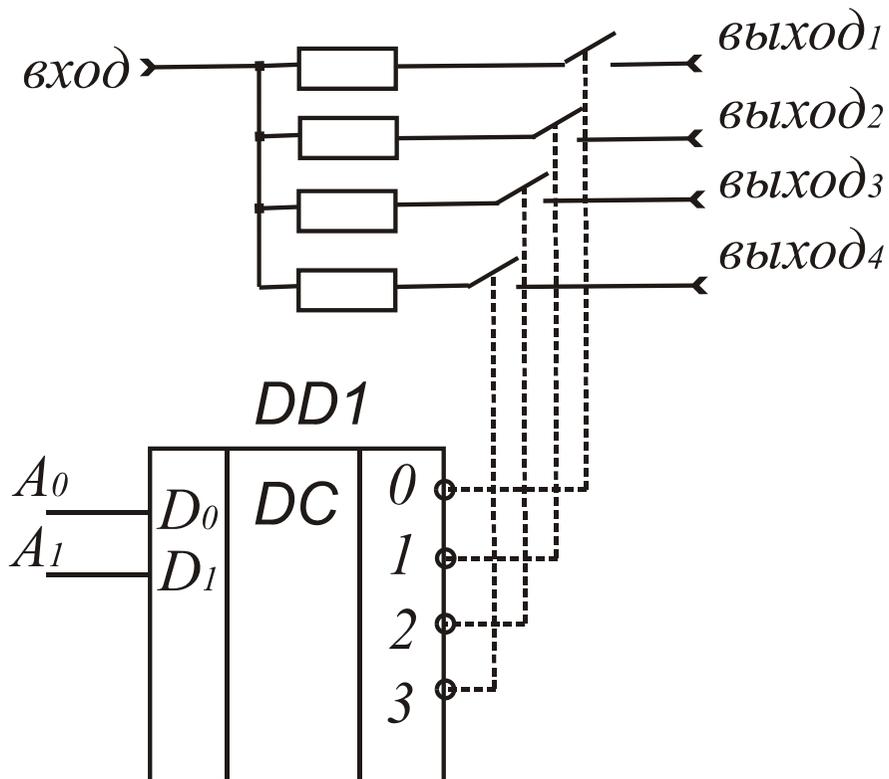
$$F_3 = D_0 A_1 A_0$$

$\overline{E0}$	$A1$	$A0$	$F0$	$F1$	$F2$	$F3$
F	0	0	$\overline{F}$	0	0	0
F	0	1	0	$\overline{F}$	0	0
F	1	0	0	0	$\overline{F}$	0
F	1	1	0	0	0	$\overline{F}$

**В сериях ТТЛ** в качестве демультиплексоров применяются дешифраторы с входом разрешения. В этом случае входные данные подаются на вход разрешения.



## Схема демультиплексора КМОП



**В сериях КМОП** в качестве демультиплексора можно использовать мультиплексор. Данные подаются на выход, а выходные функции снимаются с входов.



## ***5. Полусумматоры и сумматоры***

- предназначены для арифметического сложения двух переменных.

### ***Классификация сумматоров***

По числу выводов различают:

- ***полусумматоры***
- ***одноразрядные сумматоры***
- ***многоразрядные сумматоры***

***Полусумматором*** называется устройство, предназначенное для сложения двух одноразрядных переменных, имеющее два входа и два выхода и формирующее из сигналов входных слагаемых сигналы суммы и переноса в старший разряд.

***Одноразрядным сумматором*** называется устройство, предназначенное для сложения двух одноразрядных кодов (переменных), имеющее три входа и два выхода, и формирующее из сигналов входных слагаемых и сигнала переноса из младших разрядов сигналы суммы и переноса в старший разряд.

***Многоразрядным сумматором*** называется устройство, предназначенное для сложения двух многоразрядных кодов, формирующее на выходе код суммы и сигнал переноса в случае, если результат сложения не может быть представлен кодом, разрядность которого совпадает с разрядностью кодов слагаемых.



## Классификация сумматоров

В *последовательных* сумматорах операция сложения выполняется последовательно разряд за разрядом, начиная с младшего.

В *параллельных* сумматорах все разряды входных кодов суммируются одновременно.

*Комбинационные* сумматоры – устройства, не имеющие собственной памяти.

*Накапливающие* сумматоры, снабженные собственной внутренней памятью, в которой аккумулируются результаты выполненной операции. При этом каждое очередное слагаемое прибавляется к уже имевшемуся в устройстве значению.

В *синхронных* сумматорах время выполнения операции арифметического суммирования двух кодов не зависит от вида самих кодов и всегда остается постоянным. В *асинхронных* сумматорах время выполнения операции зависит от вида слагаемых. Поэтому по завершении выполнения суммирования необходимо вырабатывать специальный сигнал завершения операции.

В зависимости от используемой системы счисления различают *двоичные*, *двоично-десятичные* и другие типы сумматоров.

**5.1 Полу сумматор** применяется для сложения одnorазрядных двоичных чисел или младших разрядов многоразрядных слов.

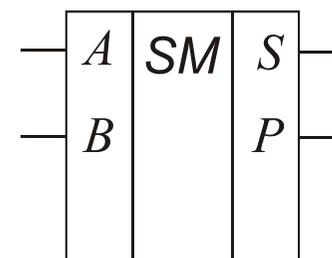
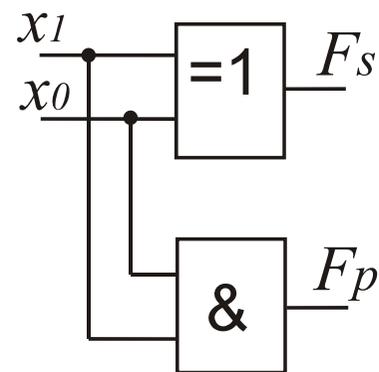
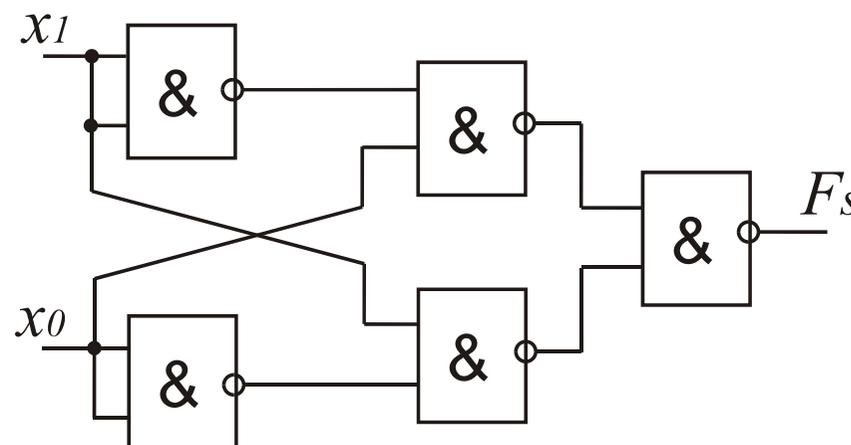
**Слово** – величина, измеряемая в битах и равная разрядности данных, обрабатываемых цифровым устройством.

$x_1$	$x_0$	$Fs$	$Fp$
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$Fs = \overline{x_1}x_0 + x_1\overline{x_0} = x_1 \oplus x_0$$

$$Fp = x_1x_0$$

$t_{\Sigma} = 3t_{з.п}$ , где  $t_{з.п}$  - задержка распространения элемента ИЛИ-НЕ

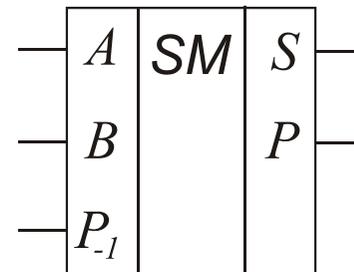




## 5.2 Полный сумматор

$x_1$	$x_0$	$p_{-1}$	$Fs$	$Fp$
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

Таблица сложения старших разрядов многоразрядных двоичных слов должна быть дополнена переменной возможного переноса из более младшего разряда ( $p-1$ ).

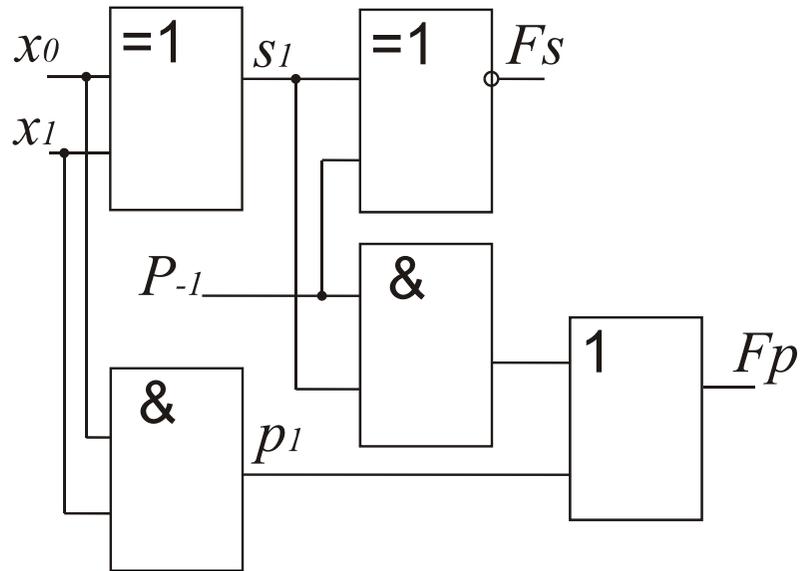


$$Fs = (\overline{x_1 x_0} + x_1 \overline{x_0}) \overline{p_{-1}} + (\overline{\overline{x_1 x_0} + x_1 \overline{x_0}}) p_{-1} =$$

$$= (x_1 \oplus x_0) \overline{p_{-1}} + (\overline{x_1 \oplus x_0}) p_{-1}$$

$$Fp = x_0 x_1 + (\overline{x_1 x_0} + x_1 \overline{x_0}) p_{-1} =$$

$$= x_0 x_1 + (x_1 \oplus x_0) p_{-1}$$



$$Fs = (\bar{x}_1 x_0 + x_1 \bar{x}_0) \bar{p}_{-1} + (\bar{x}_1 x_0 + x_1 \bar{x}_0) p_{-1} =$$

$$= (x_1 \oplus x_0) \bar{p}_{-1} + (x_1 \oplus x_0) p_{-1}$$

$$Fp = x_0 x_1 + (\bar{x}_1 x_0 + x_1 \bar{x}_0) p_{-1} =$$

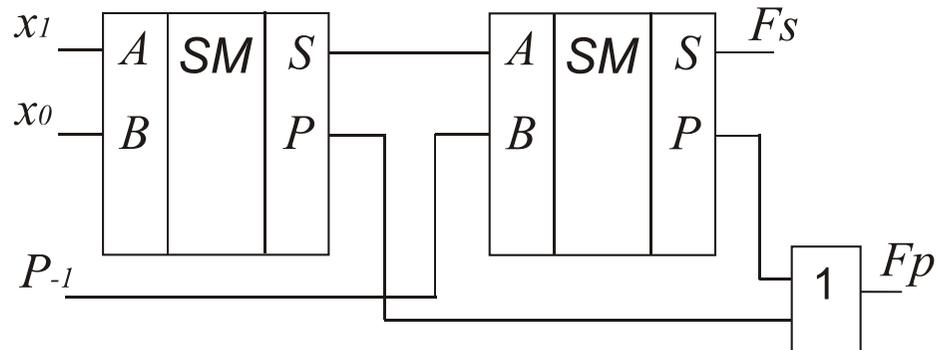
$$= x_0 x_1 + (x_1 \oplus x_0) p_{-1}$$

Время суммирования полного сумматора:

$$t_{0\Sigma} = 2 * t_{\Pi\Sigma} = 6t_{3.p}$$

Формирование сигнала переноса

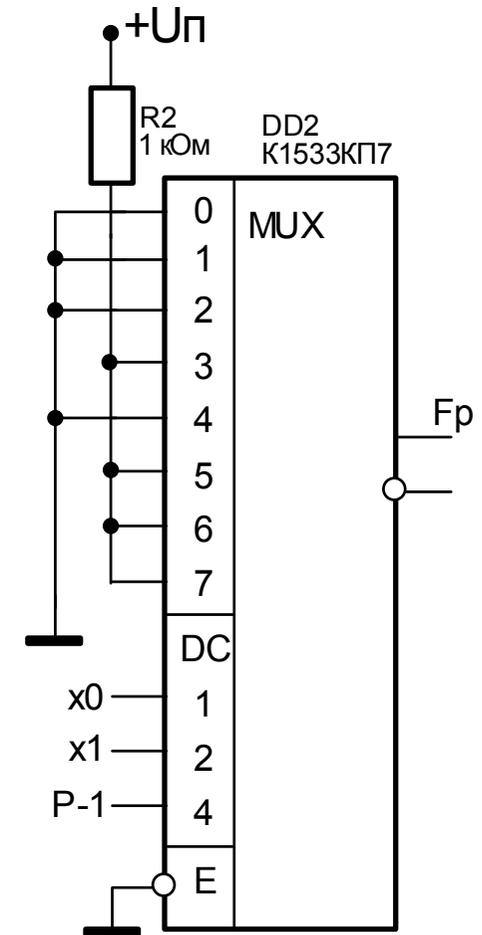
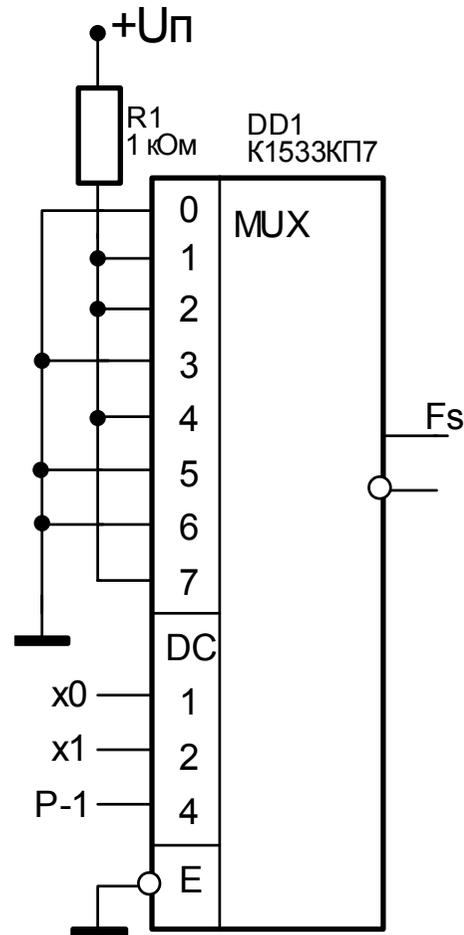
$$t_p = 5t_{3.p}$$



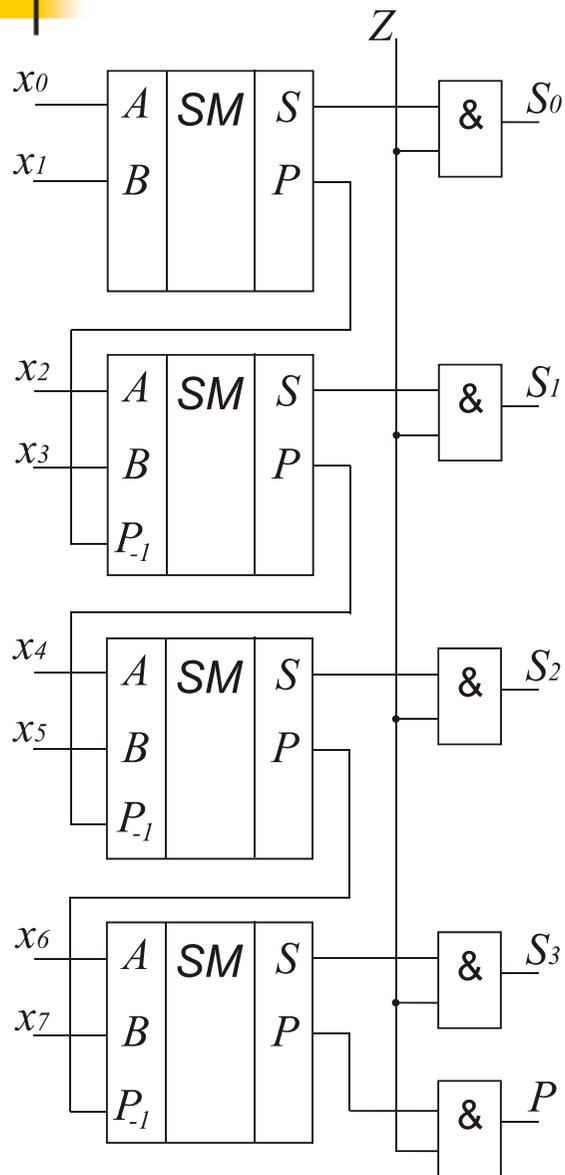


## Полный сумматор может быть реализован с использованием мультиплексора

$x_1$	$x_0$	$p_{-1}$	$Fs$	$Fp$
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1



## 5.3 Многоразрядные сумматоры



Разряды кодов слагаемых подаются на соответствующие входы сумматоров. Выход сигнала переноса сумматора нулевого разряда подается на вход переноса сумматора первого разряда и т. д. Для получения на выходе сигнала, равного реальной сумме входных кодов, необходимо, чтобы сигнал переноса последовательно сформировался на выходах сумматоров всех разрядов. Следовательно, реальное время выполнения операции в данной схеме определяется последовательным переносом сигнала  $P$  из разряда в разряд.

Для исключения получения ложного результата на выходе схемы установлены элементы И. Сигнал  $Z$  на входах этих элементов И должен появляться не ранее, чем после последовательной передачи сигнала переноса по всем разрядам сумматоров.

$$t_{0\Sigma} = t_{3.p.} [6 + 2(n-1)] = 12t_{3.p.}$$

$$t_p = 11t_{3.p.}$$



## 5.4 Полувычитатель

$$F = x_1 - x_0$$

$x_1$	$x_0$	$Fd$	$Fb$
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

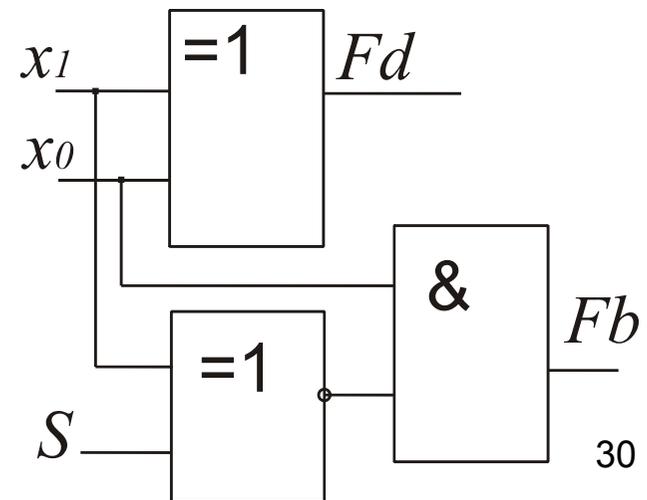
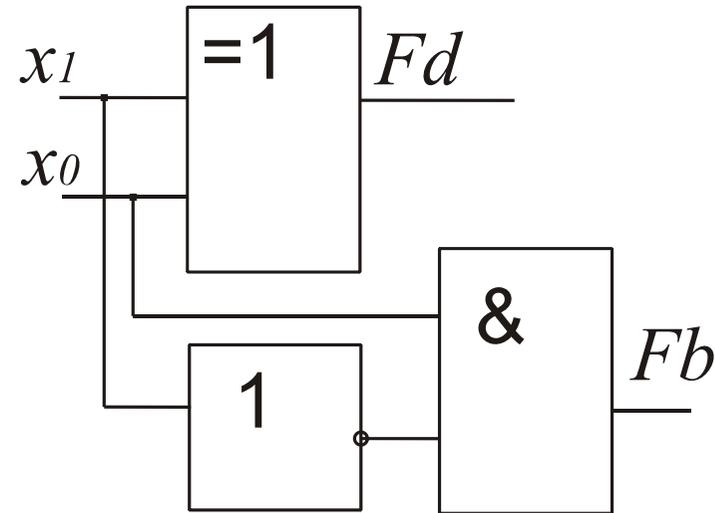
$$Fs = \overline{x_1}x_0 + x_1\overline{x_0} = x_1 \oplus x_0$$

$$Fb = \overline{x_1}x_0$$

$S$	$x_1$	$F$	
0	0	1	$\overline{x_1}$
0	1	0	
1	0	0	$x_1$
1	1	1	

$S = 0$  – вычитатель

$S = 1$  – сумматор





## 5.5 Полный вычитатель

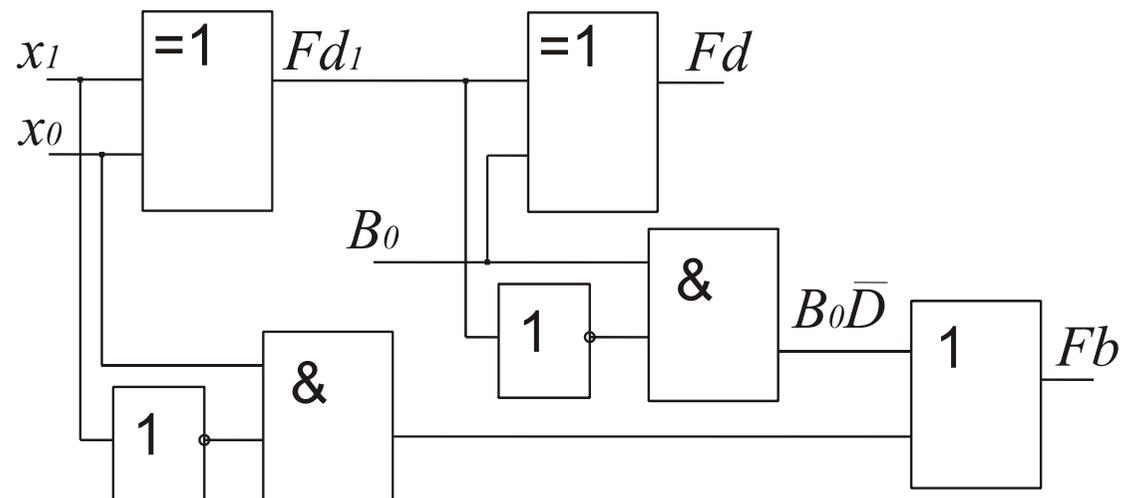
Выполняет арифметическое вычитание с учетом заема из предыдущего разряда

$$F = x1 - x0 - b0$$

$B_0$	$x_1$	$x_0$	$Fd$	$Fb$
0	0	0	0	0
0	0	1	1	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	0
1	1	1	1	1

$$Fd = \overline{x_1}x_0\overline{B_0} + x_1\overline{x_0}\overline{B_0} + \overline{x_1}\overline{x_0}B_0 + x_1x_0B_0$$

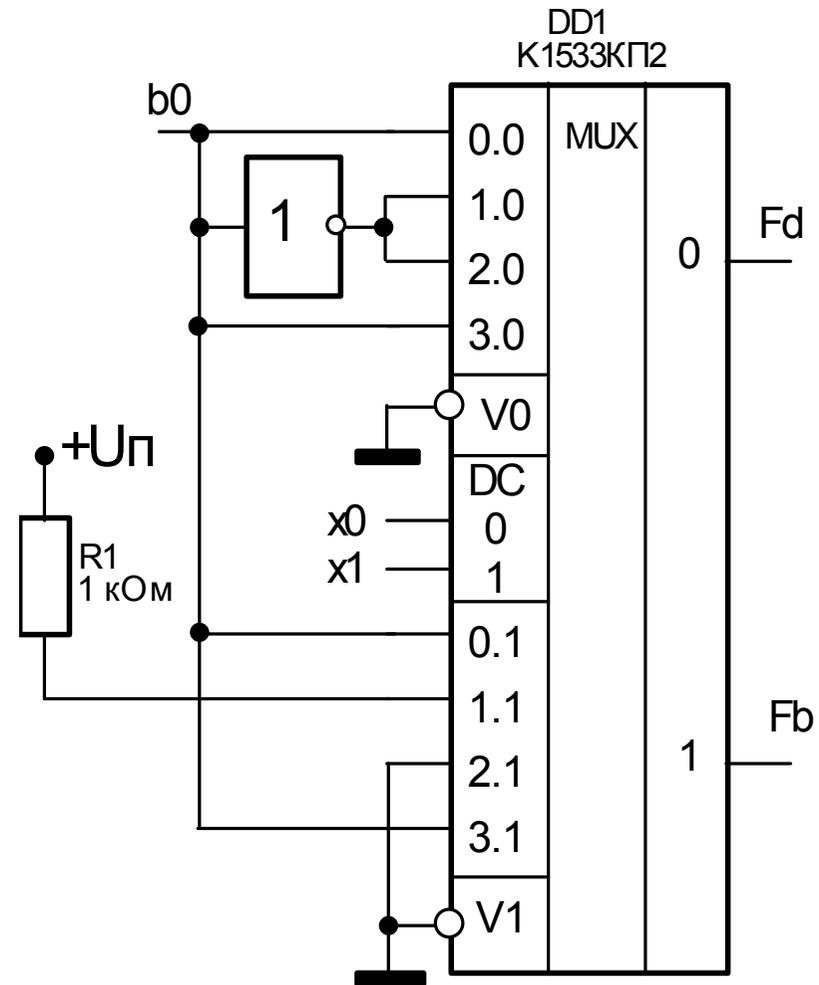
$$Fb = \overline{x_1}x_0\overline{B_0} + \overline{x_1}\overline{x_0}B_0 + \overline{x_1}x_0B_0 + x_1x_0B_0$$





## Полный вычитатель на мультиплексоре

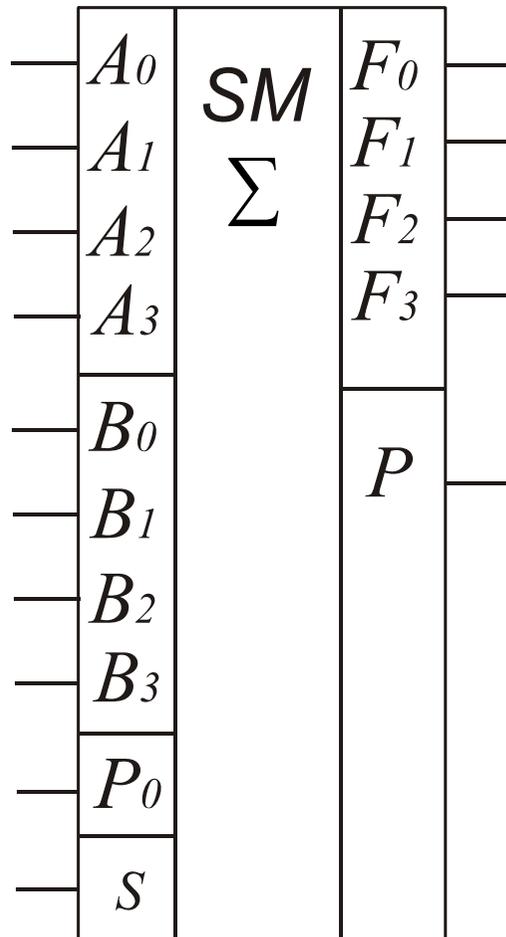
$x_1$	$x_0$	$B_0$	$Fd$	$Fb$
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1





## Сумматор - вычитатель

### *K555ИМ7*



$P_0$  – перенос в случае сумматора,  $B_0$

– заем в случае вычитателя.

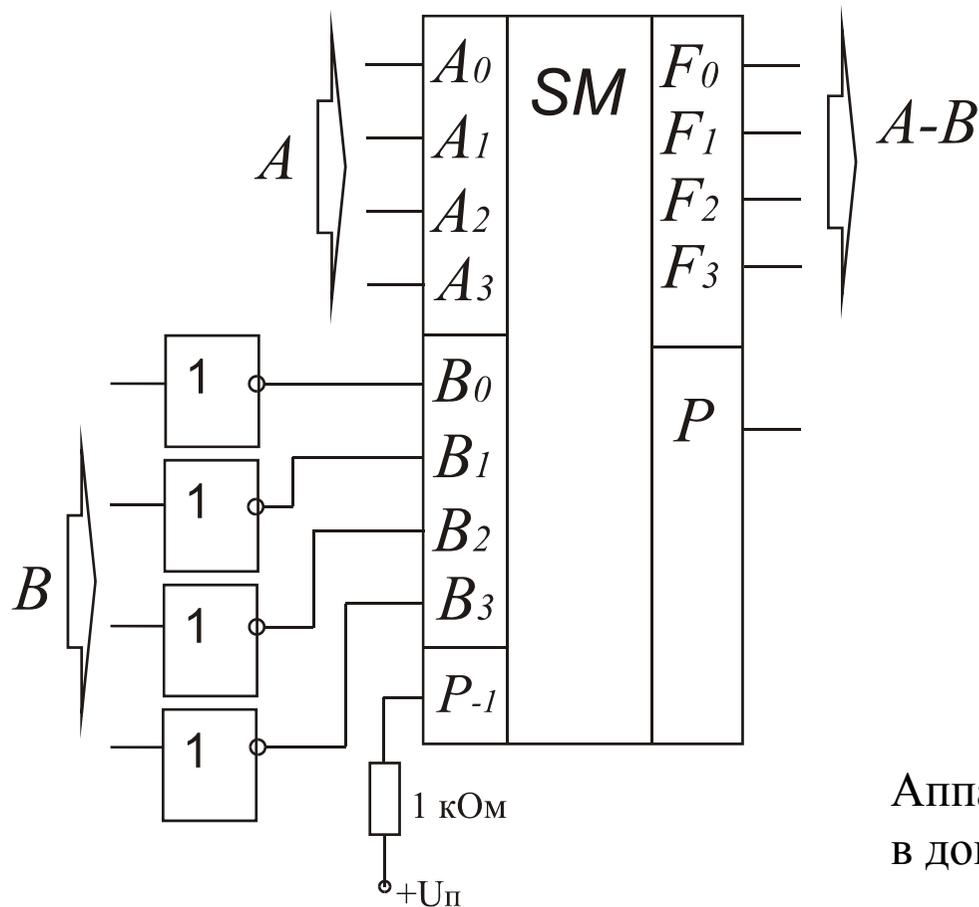
$S$  – выбор «сумматор - вычитатель».

Для K555ИМ7 при  $S=“0”$  работает как сумматор, при  $S=“1”$  как вычитатель.

## Вычитание с использованием дополнительного кода числа

### Алгоритм получения дополнительного кода двоичного числа:

1. Записывают обратный код исходного числа, для чего все его разряды инвертируют (заменяют дополнениями);
2. К полученному после инвертирования коду добавляют единицу.



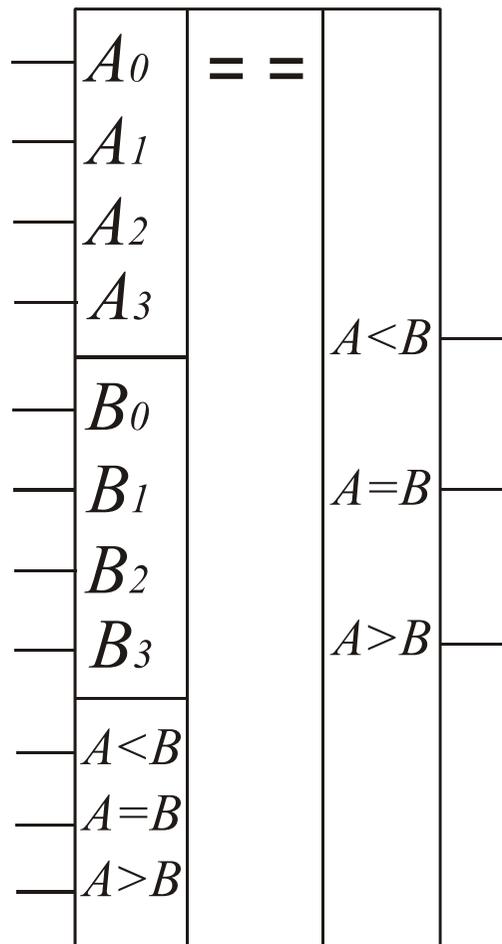
Аппаратная реализация вычитания  
в дополнительном коде



## 6. Схемы сравнения (цифровой компаратор) -

предназначены для сравнения двух двоичных чисел.

1533СП1 (561ИП2)

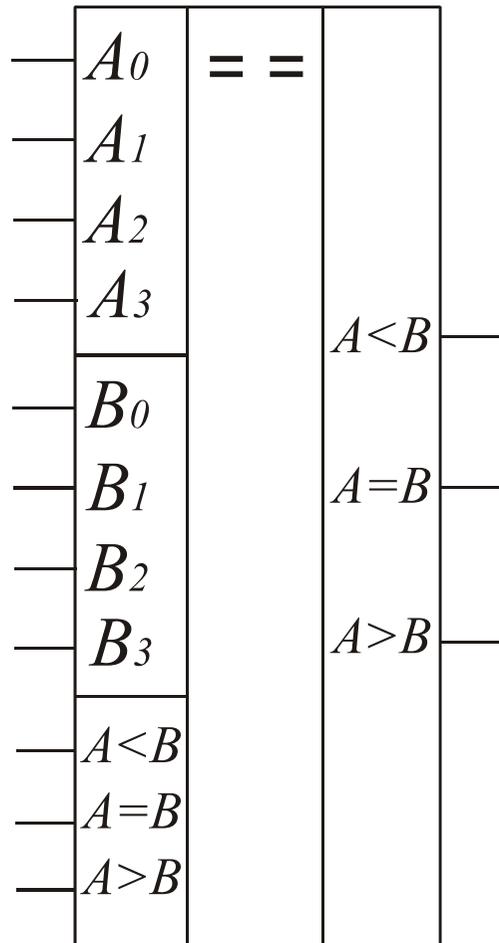


Компаратор имеет две группы входов. На одну из них поступают разряды первого числа (А), на другую группу - разряды второго числа (В). Три выхода компаратора фиксируют результат сравнения появлением логической “1”. На одном выходе она устанавливается при равенстве чисел ( $A=B$ ), на другом - при  $A < B$ , на третьем - при  $A > B$ .

Значение на входах  $A=B$ ,  $A < B$ ,  $A > B$  влияют на результат сравнения, только если  $A_3A_2A_1A_0 = B_3B_2B_1B_0$ . При этом, если на вход  $A=B$  подана «1», то  $F_{A=B} = \langle 1 \rangle$ , значения на входах  $A < B$  и  $A > B$  могут быть произвольными. Если на вход  $A=B$  подан «0», то  $F_{A=B} = 0$ , а  $F_{A < B}$  и  $F_{A > B}$  будут определяться значением на входах  $A < B$  и  $A > B$ .



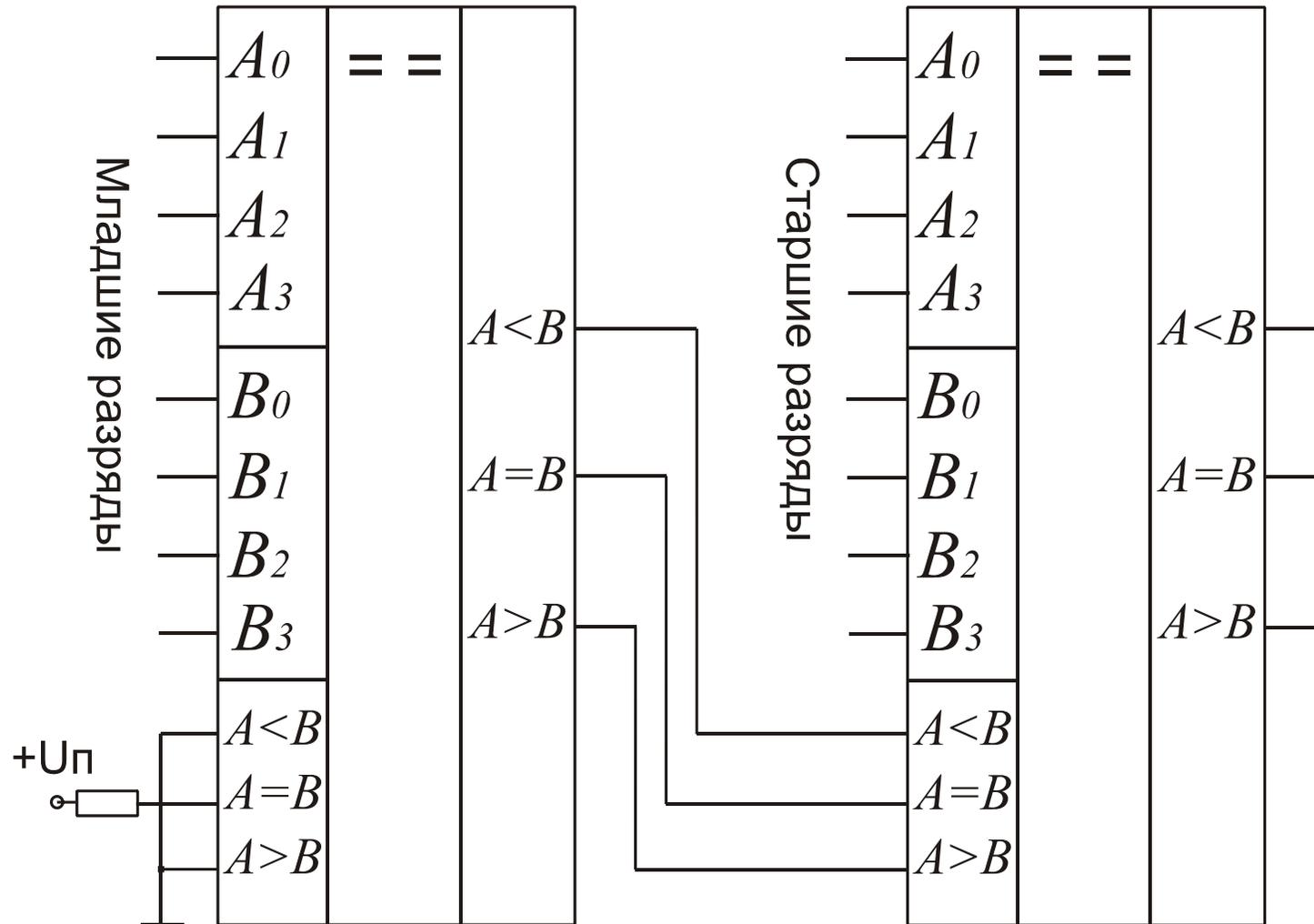
### 1533СП1 (561ИП2)



ВХОДЫ					ВЫХОДЫ		
$A_1A_0$	$B_1B_0$	$F^{-1}_{A=B}$	$F^{-1}_{A<B}$	$F^{-1}_{A>B}$	$F_{A=B}$	$F_{A<B}$	$F_{A>B}$
00	01	x	x	x	0	1	0
00	10	x	x	x	0	1	0
00	11	x	x	x	0	1	0
01	00	x	x	x	0	0	1
01	10	x	x	x	0	1	0
01	11	x	x	x	0	1	0
10	00	x	x	x	0	0	1
10	01	x	x	x	0	0	1
10	11	x	x	x	0	1	0
11	00	x	x	x	0	0	1
11	01	x	x	x	0	0	1
11	10	x	x	x	0	0	1
		1	x	x	1	0	0
00	00	0	1	0	0	1	0
01	01	0	0	1	0	0	1
10	10	0	1	1	0	0	0
11	11	0	0	0	0	1	1



## Наращивание разрядности компаратора



Каскадное соединение компараторов



## 7. Схема контроля четности

Проверка паритета (четности) двоичных чисел используется в системах передачи двоичной информации с целью повышения надежности передачи.

Суть заключается в суммировании по модулю 2 всех разрядов с целью выяснения четности числа. Допускается, что в каждый момент времени ошибка может возникнуть только в одном разряде, и проявляется она в лишней единице или в потере единицы. В обоих случаях число

если передаваемое слово содержит четное количество единиц, то в конце линии передачи это число окажется

На передающем конце схема сравнения (рис. 1), так называемый паритетный или контрольный бит, добавляется к выходной информации. Назначение паритета заключается в том, что на каждом передаваемом слове до четного разряда добавляется контрольный бит системы кодирования. На приемном конце системы кодирования поступивших сигналов. Если он правильно

Паритет может быть **четным** и **нечетным**. В случае нечетного паритета дополнительный бит формируется таким образом, чтобы сумма всех единиц в передаваемом слове, включая контрольный бит, была нечетной. Для четного наоборот.

Например, в числе  $0111_2$  число единиц нечетно. Поэтому для нечетного паритета дополнительный бит должен быть нулем, а для четного - единицей.

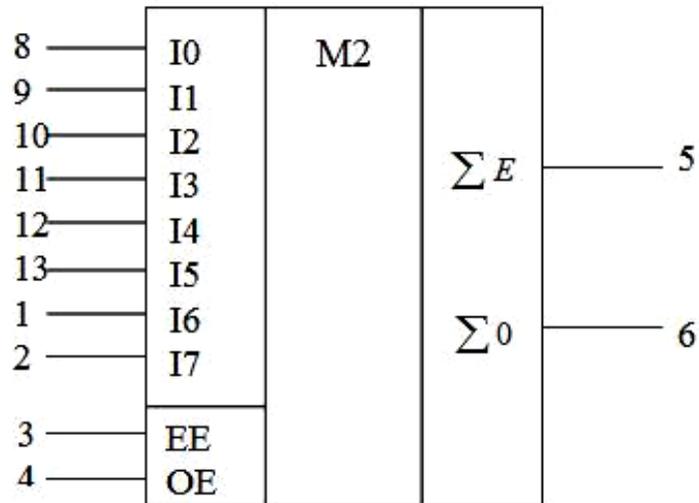
Передаваемое слово будет:

*00111 - нечетный паритет;*

*10111 - четный паритет.*



## 155ИП2



*I0-I7* – 8 информационных входов,  
два разрешающих входа для задания  
вида паритета:  
четный *EE* (even enable) и нечетный  
*OE* (odd enable),  
два выхода  $\Sigma E$  и  $\Sigma 0$ . Выходы  
взаимодополняющие.

Входы				Выход	
A	B	C	D	при E = 0 неч. пар.	при E = 1 чет. пар.
0	0	0	0	0	1
0	0	0	1	1	0
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	1	0
0	1	0	1	0	1
0	1	1	0	0	1
0	1	1	1	1	0
1	0	0	0	1	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	0	0	1
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	0	1

## 8. Арифметико-логическое устройство (АЛУ)

многофункциональное устройство, которое выполняет над входными числами различные арифметические и логические операции.



Тип выполняемой операции (логическая или арифметическая) определяется уровнем управляющего сигнала на входе  $M$  (mode control). Если  $M=1$ , то в схеме блокируются все внутренние переносы и она поразрядно выполняет логические операции. При  $M=0$  разблокируются внутренние переносы и ИС выполняет арифметические операции.

Конкретный вид выполняемой операции определяется кодом операции -  $S_3S_2S_1S_0$ .

Результаты выполненной операции снимаются с выводов  $F_3, F_2, F_1$  и  $F_0$ . Схема имеет вход  $P_0$  и выход  $P_n$  переноса.

Выход  $A=B$  является выходом встроенного компаратора, формирующего "1" при  $A=B$ .

Выходы  $G$  и  $H$  используются при построении многоразрядных АЛУ с параллельным переносом.



## Операции, выполняемые микросхемой 564ИПЗ

Код операции		Логические операции M=1	Арифметические операции M=0	
S <sub>16</sub>	S <sub>0</sub> S <sub>1</sub> S <sub>2</sub> S <sub>3</sub>		$\bar{P}_0 = 1$	$\bar{P}_0 = 0$
0	0 0 0 0	$\bar{A}$	A	A + 1
1	0 0 0 1	$\overline{A \vee B}$	$A \vee B$	$(A \vee B) + 1$
2	0 0 1 0	$\overline{A \wedge B}$	$A \vee \bar{B}$	$(A \vee \bar{B}) + 1$
3	0 0 1 1	0	-1	0
4	0 1 0 0	$\overline{A \wedge B}$	$A + (A \wedge \bar{B})$	$A + (A \wedge \bar{B}) + 1$
5	0 1 0 1	$\bar{B}$	$(A \vee B) + (A \wedge \bar{B})$	$(A \vee B) + (A \wedge \bar{B}) + 1$
6	0 1 1 0	$A \oplus B$	$A - B - 1$	$A - B$
7	0 1 1 1	$A \wedge \bar{B}$	$(A \wedge \bar{B}) - 1$	$A \wedge \bar{B}$
8	1 0 0 0	$\overline{A \vee B}$	$A + (A \wedge B)$	$A + (A \wedge B) + 1$
9	1 0 0 1	$\overline{A \oplus B}$	$A + B$	$A + B + 1$
A	1 0 1 0	B	$(A \vee \bar{B}) + (A \wedge B)$	$(A \vee \bar{B}) + (A \wedge B) + 1$
B	1 0 1 1	$A \wedge B$	$(A \wedge B) - 1$	$A \wedge B$
C	1 1 0 0	1	$A + A$	$A + A + 1$
D	1 1 0 1	$A \vee \bar{B}$	$(A \vee B) + A$	$(A \vee B) + A + 1$
E	1 1 1 0	$A \vee B$	$(A \vee \bar{B}) + A$	$(A \vee \bar{B}) + A + 1$
F	1 1 1 1	A	$A - 1$	A



## *Наращивание разрядности АЛУ*

*Наращивание разрядности АЛУ* осуществляется аналогично сумматорам, т. е. выход переноса АЛУ младших разрядов подключается к входу переноса АЛУ старших разрядов.

При большом числе разрядов время выполнения операций увеличивается из-за большого числа переносов из разряда в разряд. Для ускорения этого процесса используются специальные микросхемы ускоренного переноса (555ИП4, 564ИП4). Одна такая микросхема обслуживает до 4-х АЛУ. Получается 16-ти разрядное АЛУ.