

## Элементы комбинационных логических схем.

Работа любых электронных схем характеризуется изменением напряжений или токов в электронных компонентах. С этими напряжениями или токами можно связать сигналы, передающие полезную информацию. Сигналы в электронных схемах могут быть аналоговыми или цифровыми. Работы по созданию вычислительных устройств, точность результатов вычисления в которых не зависит от свойств электронных компонентов, привели к широкому использованию логических (цифровых) сигналов и обрабатывающих их логических электронных компонентов.

### 1.1. Сигналы и их модели

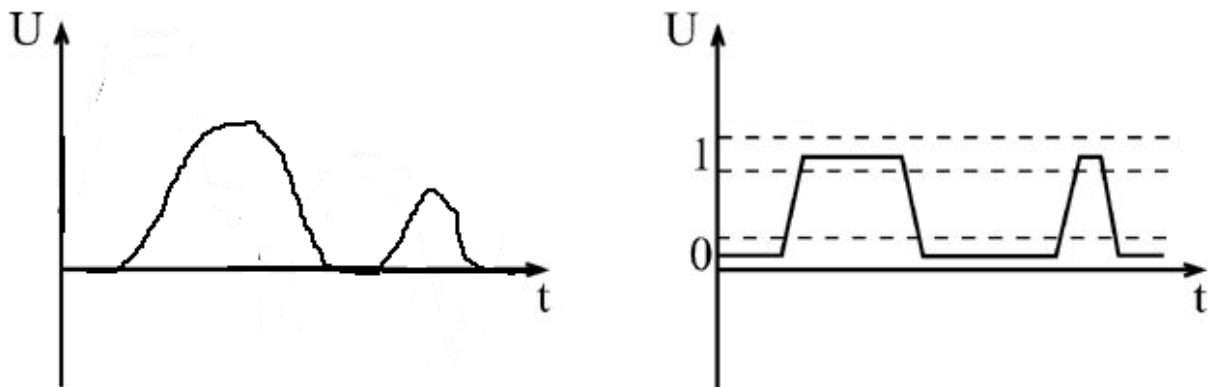
**Сигнал** - это любая физическая величина (например, напряжение, температура, давление воздуха, интенсивность света, сила тока и т.д.), изменяющаяся во времени и рассматриваемая как носитель информации.

**Электрический сигнал** - это электрическая величина (например, напряжение, ток, мощность). Современная электроника в основном работает с электрическими сигналами, хотя сейчас все больше используются оптические сигналы, которые характеризуются изменяющейся во времени интенсивностью света. К основным параметрам сигналов относятся амплитудные, частотные (временные) и мощностные характеристики.

#### **Аналоговые и дискретные (цифровые) сигналы.**

**Аналоговый сигнал (analog signal)** — это сигнал, который может принимать в определенных пределах любые различимые значения. Например, напряжение может плавно изменяться в пределах от нуля до десяти вольт. Устройства, работающие только с аналоговыми сигналами, называются **аналоговыми устройствами**. Название "аналоговый" подразумевает, что сигнал изменяется аналогично некоторой физической величине, то есть непрерывно.

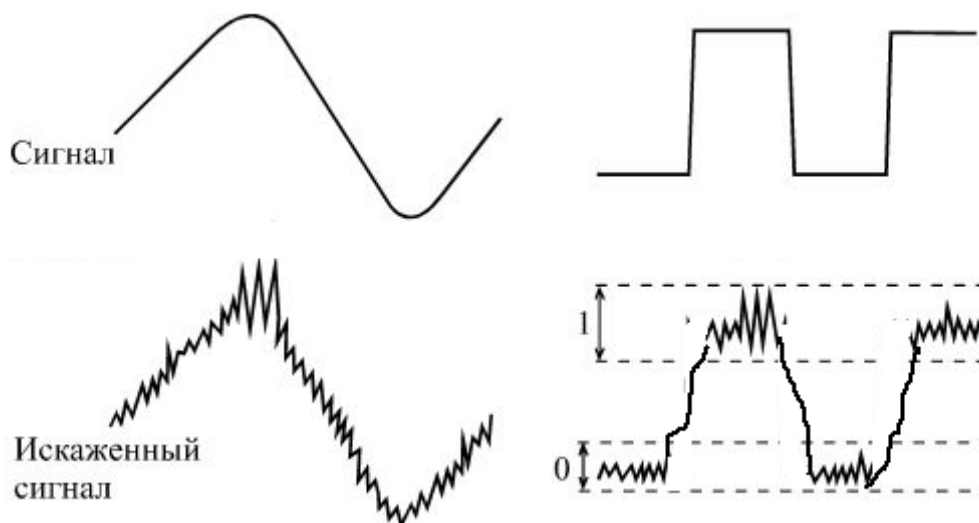
**Дискретный (цифровой) сигнал (digital signal, discrete signal)** - это сигнал, в котором различают лишь несколько значений. Для систем обработки информации характерно использование двоичных сигналов, т.е. сигналов, в которых различимы только два из возможных значений (рис. 1.1). Например, напряжение может принимать значения: от 0 до 5 В, но различимыми будут считаться лишь два диапазона значений. Значение сигнала в диапазоне 0- 0,5 В принимают за уровень нуля, а от 2,5 до 5 В за уровень единицы. **Системы, работающие исключительно с цифровыми сигналами, называются цифровыми.**



**Рис.1.1. Электрический сигнал (слева) и его цифровое представление (справа)**

Сигналы подвергаются действию шумов, наводок, помех.

**Наводки и помехи (pickup)** – это внешние или возникающие внутри системы возмущения, искажающие полезный сигнал. Например, источником внешних помех может стать проникающее в систему электромагнитное излучение, пролет космических частиц и т.п., источником внутренних помех могут быть электромагнитные наводки в проводниках от переключений сигналов в соседних проводниках, нестабильность источников питания и т.п.). В случае аналоговых систем все это существенно ухудшает полезный сигнал, так как все его значения разрешены (рис. 1.2).



**Рис. 1.2. Искажение помехами идеального аналогового (слева) и цифрового (справа) сигналов**

В отличие от аналоговых, цифровые сигналы, имеющие обычно всего два различимых значения, лучше защищены от действия помех. Небольшие

отклонения от разрешенных значений никак не искажают цифровой сигнал, так как существуют зоны допустимых отклонений (рис. 1.2). Все эти преимущества в настоящее время обеспечивают опережающее развитие цифровых систем обработки информации.

## 1.2. Логические функции и логические элементы.

Известны различные математические модели сигналов. Наиболее простой является двоичная или Булева модель.

*В этой модели принято, что сигналы имеют лишь два уровня:*

- **1(истинно-True)**
- **0(ложно-False)**

*Все логические операции по преобразованию сигналов в этом двоичном алфавите описываются Булевой алгеброй.*

Ниже в **табл.1.1** представлены Булевские таблицы истинности логических операций (элементарных логических функций):

- **НЕ (NOT-инверсия)**
- **И (AND- конъюнкция)**
- **И-НЕ (NAND- конъюнкция с инверсией)**
- **ИЛИ (OR- дизъюнкция)**
- **ИЛИ-НЕ (NOR- дизъюнкция с отрицанием)**
- **ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR-сложение по модулю)**
- **ИСКЛЮЧАЮЩЕЕ ИЛИ-НЕ (XNOR- равнозначность).**

В **табл.1.1** столбцы А и В содержат значения аргументов Булевых функций, под названиями функций – столбцы с результатами соответствующих операций.

**Таблица 1.1. Булевские таблицы истинности ряда логических функций.**

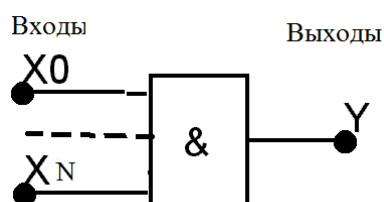
Вход	Выход
A	NOT
0	1
1	0

Входы		Выходы Булевых функций					
A	B	AND	NAND	OR	NOR	XOR	NXOR
0	0	0	1	0	1	0	1

0	1	0	1	1	0	1	0
1	0	0	1	1	0	1	0
1	1	1	0	1	0	0	1

Логические операции над цифровыми сигналами выполняют **логические элементы**.

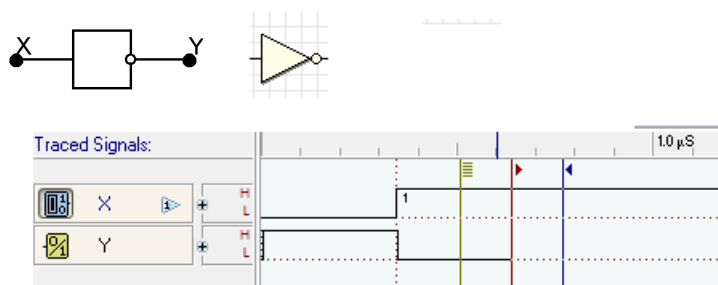
На **рис.1.3** представлено Условное Графическое Отображение (обозначение) -УГО логического элемента с N входами и одним выходом.



**Рис. 1.3.** Условное графическое отображение (УГО) N-входового логического элемента

### 1.2.1.Элемент НЕ (NOT-инвертор).

Функция логического отрицания **НЕ** (NOT) инвертирует значение аргумента. Этой функции соответствует логический элемент **НЕ** (инвертор) с одним входом и одним выходом. На **рис.1.4** представлены два варианта его условных графических изображений (УГО): принятое в нашей стране-прямоугольник и принятое на Западе-треугольник, а также временная диаграмма его работы.



**Рис. 1.4.** Условное графическое обозначение элемента НЕ (NOT) и временная диаграмма его работы.

Логический элемент **НЕ** инвертирует поступающий на вход сигнал X, другими словами, на своем выходе заменяет значение входного сигнала X на противоположное. Операция инверсии обозначается в Булевой алгебре надчеркиванием  $Y = \overline{X}$ , а в языке VHDL словом NOT. Не вдаваясь в детали языка описания аппаратуры VHDL ((Very high speed integrated circuits)

Hardware Description Language), приведем описание на нем Инвертора. Вначале описывается интерфейс(entity) элемента NOT\_1 (латинские прописные и заглавные буквы не различаются).

```
entity NOT_1 is  
  port ( X : in bit; Y : out bit );  
end ;
```

Элемент (entity) в примере с именем NOT\_1, имеет вход(сокращение от in- input), обозначенный как X и выход(out-output) обозначенный как Y двоичного типа (сигналы на X и Y принимают значения '0' или '1').

Функция элемента – инвертора NOT\_1 описана в разделе VHDL-описания, обозначенным словом архитектура с именем BEH.

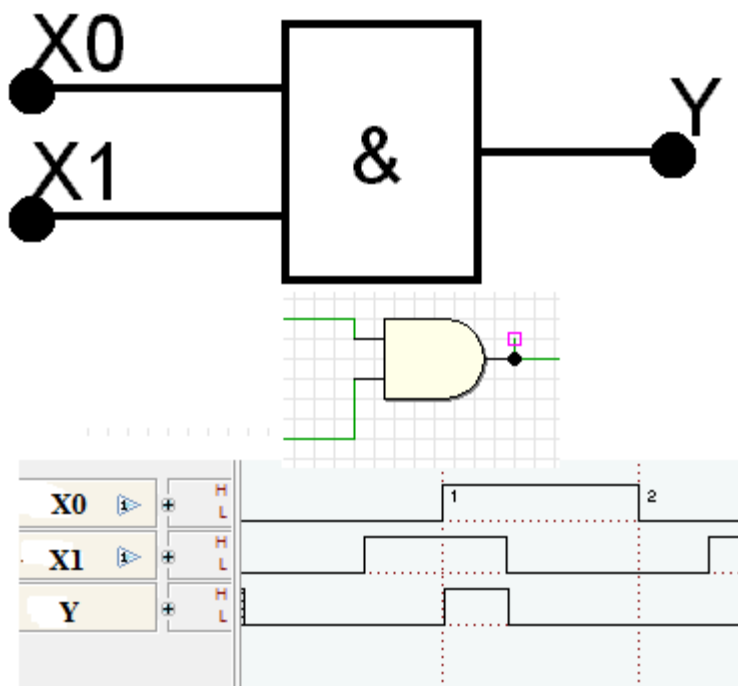
```
architecture BEH of NOT_1 is  
begin  
  Y <= NOT X;  
end;
```

В этом VHDL-описании архитектуры использована логическая операция отрицания- NOT(НЕ) и оператор назначения сигнала <= . Смысл этого оператора в том, что при изменении значения аргументов в его правой части, в сигнал, присутствующий в его левой части, устанавливается новое значение, в данном примере- инверсное.

Видно, что VHDL описание более громоздко, чем на языке Булевой алгебры ( $Y = \bar{X}$ ), но зато оно непосредственно может быть введено и обработано на ЭВМ.

### 1.2.2. Элемент И (AND).

Логический элемент **И** (AND) работает с двумя и более входными сигналами. Условное графическое обозначение логического элемента **И** с двумя входами (**2И**) и временная диаграмма его работы приведены на **рис.1.5**.



**Рис. 1.5.** Условное графическое обозначение логического элемента И (AND) и временная диаграмма его работы.

На вход логического элемента И подаются два сигнала X0 и X1, на выходе появляется сигнал Y. Операция **И** в булевой алгебре обычно обозначается символом \* или  $\wedge$  и его логическую функцию в булевой алгебре можно описать как  $Y = X0 \wedge X1$ , а в языке VHDL операция **И** обозначается как AND. В VHDL -примере этот элемент имеет имя **AND\_2**, где суффикс 2 в его имени поясняет количество входов.

```
entity AND_2 is port( X0 ,X1 : in bit; Y : out bit ); end ;
architecture BEH of AND_2 is
begin
    Y <= X0 AND X1; --
end ;
```

### 1.2.3. Элемент **ИЛИ** (OR).

Условное графическое обозначение логического элемента **ИЛИ**(OR) и временная диаграмма его работы приведены на **рис.1.6**. Операция **ИЛИ** булевой алгебре обозначается символом  $\vee$  или +, а в языке VHDL обозначается как OR.

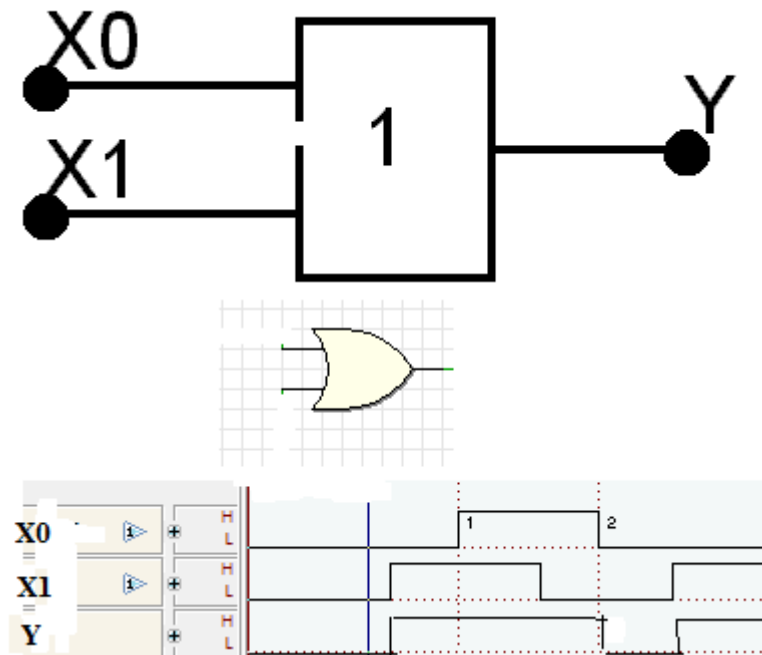


Рис. 1.6. Условное графическое обозначение логического элемента ИЛИ и его временная диаграмма.

Элемент ИЛИ выполняет следующую логическую функцию

$$Y = X0 \vee X1$$

Его VHDL –описание представлено ниже с именем **OR\_2**

```
entity OR_2 is port( X0 ,X1 : in bit; Y : out bit ); end ; [1][2]
architecture BEH of OR_2 is
begin;
    Y <= X0 OR X1;
end ;
```

#### 1.2.4. Элементы И-НЕ (NAND) и ИЛИ-НЕ (NOR).

Обозначения логических элементов **И-НЕ** (NAND) и **ИЛИ-НЕ** (NOR) приведены на **рис.1.7**.



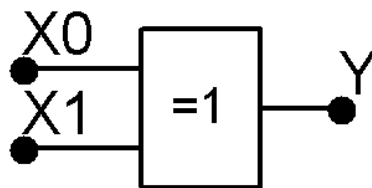
**Рис. 1.7. Условное графическое обозначение логических элементов И-НЕ и ИЛИ-НЕ.**

```
entity NAND_2 is port( X0 ,X1 : in bit; Y : out bit ); end ; [1][2]
architecture BEH of NAND_2 is
begin
    Y <= ( X0 NAND X1);
end ;
entity NOR_2 is port( X0 ,X1 : in bit; Y : out bit ); end ; [1][2]
architecture BEH of NOR_2 is
begin
    Y <= ( X0 NOR X1);
end ;
```

### 1.2.5. Элемент Исключающее ИЛИ (XOR)

Другое название логической функции **Исключающее ИЛИ(XOR)** -это "*сложение по модулю 2 или сложение по mod2*".

УГО элемента **Исключающее ИЛИ** (exclusive-OR или XOR) представлено на **рис.1.8**.



**Рис. 1.8. Условное графическое обозначение логического элемента Исключающее ИЛИ (XOR).**

В Булевой алгебре операция XOR обозначается символом  $\oplus$ . Для двухвходовых элементов XOR (входы X0 и X1) эта операция может быть представлена так же, например, с помощью операций НЕ, И, ИЛИ (входы X0 и X1, выход Y)

$$Y = X0 \oplus X1$$

$$Y = X0 \cdot \overline{X1} + \overline{X0} \cdot X1;$$

На языке VHDL описание этого элемента имеет вид:

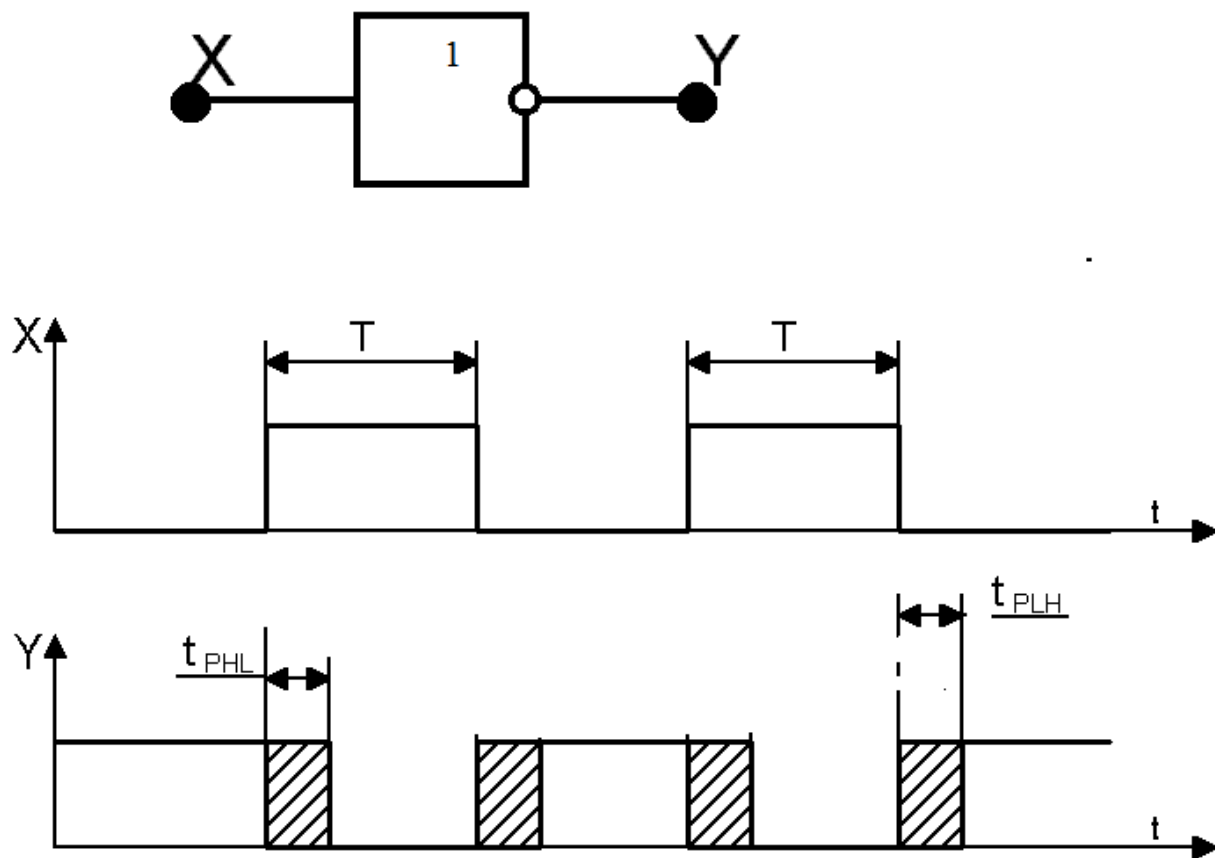
```
entity XOR_2 is port( X0 ,X1 : in bit; Y : out bit ); end ; [1][2]
architecture BEH of XOR_2 is
begin
    Y <= ( X0 XOR X1);
```



end ;

### 1.3. Задержки распространения сигнала и динамические характеристики элементов

Логические элементы обладают задержками. На **рис. 1.9.** показано условное обозначение, и временная диаграмма работы инвертора (элемент НЕ). На графике выходного сигнала инвертора  $Y$  показаны максимальные задержки его относительно входного сигнала  $X$  при изменении выходного сигнала из 1 в 0 и из 0 в 1- соответственно  $t_{PHL}$  и  $t_{PLH}$ .



**Рис. 1.9.** Условное обозначение и временная диаграмма работы инвертора (элемент НЕ).

В результате действия задержек длительность сигнала на выходе инвертора может уменьшиться или увеличиться. Это связано с тем, что времена переключения (**рис.1.9**) значения выходного сигнала из "1" в "0" ( $t_{PHL}$ -Time propagation from higher state to low)) и из "0" в "1" ( $t_{PLH}$ - Time propagation from low state to higher) разные. Области между минимальными (на **рис.1.9** приняты равным 0) и максимальными значениями задержек ( $t_{PHL}$ ,  $t_{PLH}$ ) отмечены на временной диаграмме заштрихованными областями.

Завод-изготовитель (в России) логических элементов обычно указывает только максимальные значения времен задержек при переключении из "1" в "0"-  $t_{PHL}$  и при переключении из "0" в "1" -  $t_{PLH}$ .

Для упрощения расчетов вводится **средняя величина времени задержки**, называемая также **задержкой распространения** (tpd-Time of Propagation delay) сигнала

$$t_{Pd} = (t_{PLH} + t_{PHL}) / 2;$$

Время задержки конкретного логического элемента определяется технологическим разбросом в процессе его изготовления. Понятно, что технологический разброс является случайной величиной.