

# Языки описания схем

(mk.cs.msu.ru → Лекционные курсы → Языки описания схем)

## Блок 3

### Комбинационные схемы

лектор:

Подымов Владислав Васильевич

e-mail:

**valdus@yandex.ru**

Осень 2018

# Вступление: 0 и 1

“Схема — это граф соединения логических и других элементов”

Реальная схема управляет сигналами на своих внешних контактах в реальном времени: использует и изменяет уровни напряжения на этих контактах в реальном времени

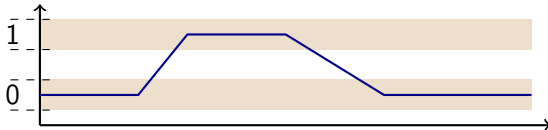
Цифровая схема работает с двумя фиксированными уровнями напряжения: 0 (низкий, GND) и 1 (высокий, Vcc)



# Вступление: погрешности напряжения

В реальном мире невозможно получить “в точности заданное” значение напряжения

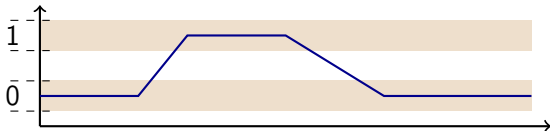
Следует иметь в виду, что на самом деле логические значения 0 и 1 — это два диапазона напряжений:



Эти диапазоны выбираются так, чтобы с помощью любых напряжений этих диапазонов можно было корректно управлять напряжениями в других точках схемы

Например, подача любого напряжения диапазона 0 должна гарантированно закрывать канал n-МОП-транзистора, а диапазона 1 — открывать этот канал в технологии КМОП

# Вступление: погрешности напряжения



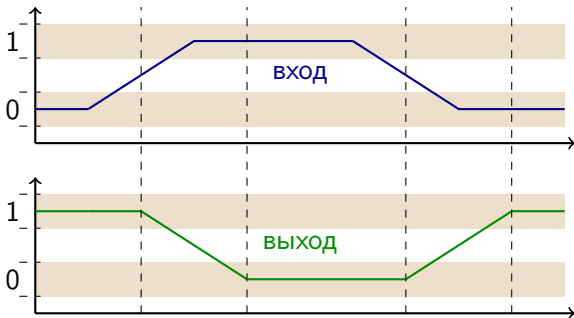
Между 0 и 1 располагается диапазон уровней напряжения, расцениваемых двояко:

- ▶ это напряжения, не отвечающие никаким логическим значениям
- ▶ в реальной схеме подача напряжения из этого диапазона приводит к корректному, но неоднозначному результату:
  - ▶ уровень напряжения на выходе зависит от многих непредсказуемых факторов, хотя и может попадать в диапазоны 0, 1
  - ▶ схема продолжит исправно функционировать, когда напряжение понизится до 0 или повысится до 1

## Вступление: реальное время

**Распространение сигнала** элементом схемы — это в общем случае **немгновенное** получение *требуемого* **стабильного** уровня напряжения на выходных контактах при подаче **стабильного** уровня напряжения на входные контакты

Например, в схеме, реализующей логическое отрицание по технологии КМОП, сигнал *скорее всего* будет распространяться *примерно* так:



# Логические вентили

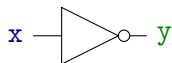
**Логические вентили** — это “чуть более реальные” аналоги (абстрактных математических) функциональных элементов, управляющие распространением сигналов в реальном времени в цифровой схеме согласно булевой логике

В зависимости от контекста рассуждений о вентилях, можно полагать, что

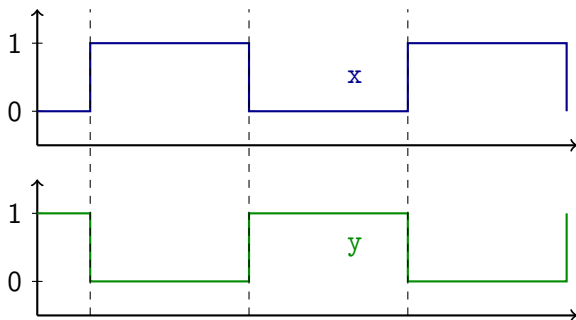
- ▶ сигнал распространяется вентилем
  - ▶ мгновенно (*СФЭ/СФЭЗ*)
  - ▶ за фиксированное время  
(*простые взвешенные модели СФЭ/СФЭЗ*)
  - ▶ за время, варьирующееся в заданных пределах  
(*продвинутые взвешенные модели СФЭ/СФЭЗ*)
- ▶ на выходе вентиля
  - ▶ всегда имеется значение либо 0, либо 1  
(*одна трактовка диапазонов напряжений*)
  - ▶ могут возникать неопределённые значения  
(*другая трактовка диапазонов напряжений*)

# Логические вентили

## Пример

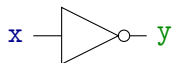


Логический вентиль, аналогичный функциональному элементу отрицания, может работать так:

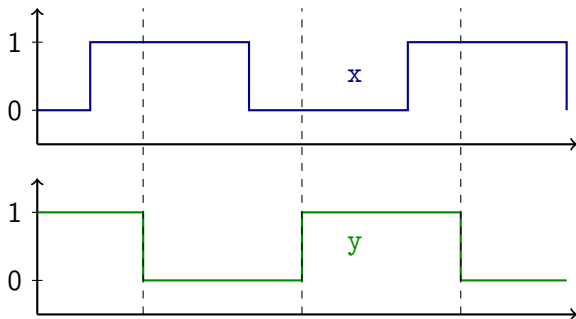


# Логические вентили

## Пример



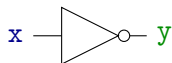
Логический вентиль, аналогичный функциональному элементу отрицания, может работать так:



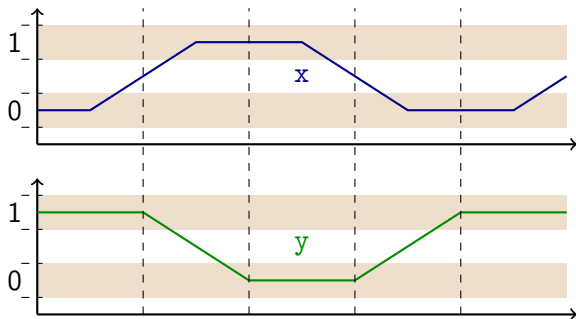


# Логические вентили

## Пример

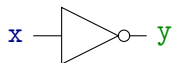


Логический вентиль, аналогичный функциональному элементу отрицания, может работать так:

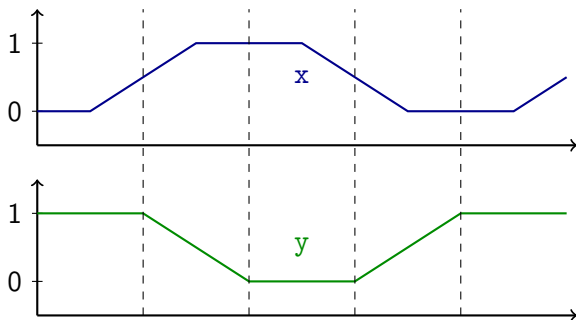


# Логические вентили

## Пример

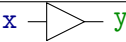
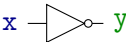
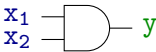




Логический вентиль, аналогичный функциональному элементу отрицания, может работать так:



# Логические вентили

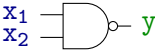


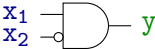

Наиболее популярные логические вентили имеют свои названия и изображения:

Название	Изображение	Функция
буфер		$y = x$
НЕ		$y = \bar{x}$
И		$y = x_1 \& x_2$
ИЛИ		$y = x_1 \vee x_2$
Исключающее ИЛИ		$y = x_1 \oplus x_2$

Кружок, дорисованный на входе или выходе вентиля, обозначает отрицание значения на соответствующем входе или выходе

# Логические вентили

Кружок на выходе добавляет в название вентиля суффикс “-НЕ”

Название	Изображение	Функция
И-НЕ		$y = \overline{x_1 \& x_2} = x_1   x_2$
ИЛИ-НЕ		$y = \overline{x_1 \vee x_2} = x_1 \downarrow x_2$
Исключающее ИЛИ-НЕ		$y = \overline{x_1 \oplus x_2} = x_1 \sim x_2$
—		$y = x_1 \& \overline{x_2}$
—		$y = \overline{\overline{x_1} \vee \overline{x_2}}$
	...	

# Комбинационные схемы

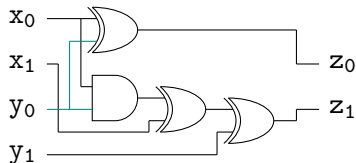
Комбинационная схема — это то же самое, что и СФЭ, с заменой ФЭ на логические вентили

Более точно, **комбинационная схема** — это ациклический ориентированный граф, имеющий следующий вид:

- ▶ истоки помечены попарно различными переменными
- ▶ стоки помечены попарно различными символами выходных функций
- ▶ остальные вершины помечены логическими вентилями
- ▶ полустепень захода каждого стока — 1
- ▶ полустепень захода вершины, помеченной вентилем, равна числу переменных булевой функции, реализуемой вентилем
- ▶ дуги, входящие в вершину, помеченную вентилем, пронумерованы числами начального отрезка натурального ряда

**Значение комбинационной схемы** — это способ преобразования входных логических значений в выходные в реальном времени  
(не очень математично, но что поделать)

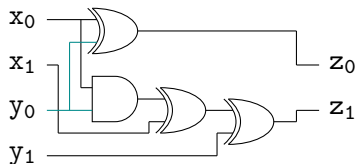
## Пример комбинационной схемы



Это сумматор с переполнением для двухбитовых чисел, реализованный в комбинационной логике

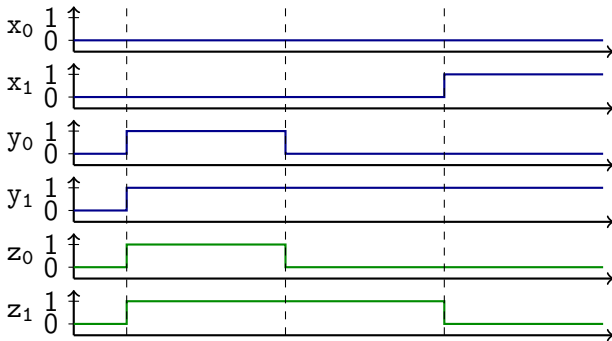
И как это работает?

## Пример комбинационной схемы

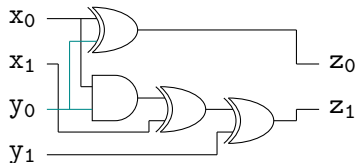


Это сумматор с переполнением для двухбитовых чисел, реализованный в комбинационной логике

И как это работает?

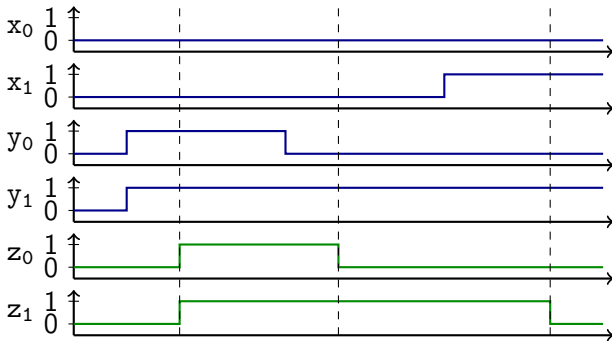


## Пример комбинационной схемы



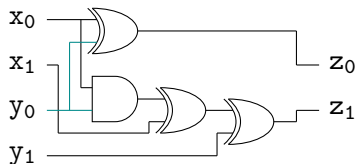
Это сумматор с переполнением для двухбитовых чисел, реализованный в комбинационной логике

И как это работает?





## Пример комбинационной схемы



Это сумматор с переполнением для двухбитовых чисел, реализованный в комбинационной логике

И как это работает?

