

**Организация  
ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ**

**Иван Андреевич Сукин**

**Тема 1:  
ОСНОВЫ**

# Тема 1 :: Разделы

- **Искусство управления сложностью**
- **Системы счисления**
  - Двоичные числа
  - Шестнадцатеричные числа
  - Байты, биты, nibbles
  - Сложение
  - Представление чисел со знаком
  - Расширение чисел
- **Логические вентили**
- **Логические уровни**
- **КМОП транзисторы**
- **Низкоуровневое представление вентилей**
- **Энергопотребление**

Тема 1: Основы

**Искусство  
управления  
сложностью**

# Управление сложностью

- Как проектировать объекты, слишком сложные для того, чтобы один человек мог осмыслить их целиком?
- Абстракция
- Дисциплина
- Три «-ности»:
  - Иерархич**ность**
  - Модуль**ность**
  - Регуляр**ность**

# Абстракция

Соккрытие ненужных  
деталей

# Дисциплина

- Сознательное ограничение множества вариантов дизайна
- Пример: цифровая дисциплина
  - Дискретные уровни напряжения вместо непрерывных
  - Дизайн цифровых систем проще чем у аналоговых – можно разрабатывать более сложные системы
  - Цифровые системы замещают аналоговых предшественников

# Три «-НОСТИ»

- **Иерархичность**

- 

- **Модульность**

- 

- **Регулярность**

-

# Цифровая абстракция

- Большинство физических величин **непрерывны**
  - Напряжение на участке цепи
  - Частота колебаний
  - Координаты тела
- Цифровая абстракция имеет дело с **дискретным подмножеством** этих значений

# Цифровая дисциплина: биты

- **Два дискретных значения:**
  - 1 и 0
  - 1, ИСТИНА, HIGH
  - 0, ЛОЖЬ, LOW
- **1 и 0:** уровни напряжения или жидкости, положения шестеренок и т.д.
- Цифровые схемы используют уровни напряжения:
  - 0: низкое напряжение (земля, GND)
  - 1: высокое напряжение ( $V_{DD}$ )
- **Bit:** двоичная цифра (*Binary digit*)

Тема 1: Основы

**Системы  
счисления:  
двоичные числа**

# Системы счисления

- Десятичная

- Двоичная

# ДВОИЧНЫЙ СЧЕТ

**Двоичная**

**Десятичная**

0

0

1

1

10

2

11

3

100

4

101

5

...

# Степени двойки

- $2^0 =$

- $2^1 =$

- $2^2 =$

- $2^3 =$

- $2^4 =$

- $2^5 =$

- $2^6 =$

- $2^7 =$

- $2^8 =$

- $2^9 =$

- $2^{10} =$

- $2^{11} =$

- $2^{12} =$

- $2^{13} =$

- $2^{14} =$

- $2^{15} =$

**Полезно  
запомнит**

# Перевод чисел

- Из двоичной в десятичную:
  - Перевести  $10011_2$  в десятичную систему
  -
  
- Из десятичной в двоичную:
  - Перевести  $47_{10}$  в двоичную систему
  -

# Из десятичной в двоичную

- Два способа:
  - **Способ 1:** Найти подходящую старшую степень двойки, вычесть её и повторить процедуру
  - **Способ 2:** Последовательно делить на 2, записывая остатки справа налево

# Из десятичной в двоичную

$53_{10}$

**Способ 1:** Вычитать подходящую старшую степень двойки

**Способ 2:** последовательно делить на 2

$53 \rightarrow 26 \rightarrow 13 \rightarrow 6 \rightarrow 3 \rightarrow 1$

# Двоичные числа

- **Десятичное число из  $N$  цифр**

- Сколько их?
- Диапазон значений?
- Пример: десятичное число из 3 цифр:

- **Двоичное число из  $N$  цифр**

- Сколько их?
- Диапазон значений
- Пример: двоичное число из 3 цифр:

•  $2^3 = 8$  возможных значений

Тема 1: Основы

**Системы  
счисления:  
шестнадцатеричны  
е числа**

# Шестнадцатеричные числа

Hex	Десятичное	Двоичное
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

# Шестнадцатеричные числа

- Основание системы счисления – 16
- Удобны для сокращенной записи двоичных чисел

# Переводы

- Из шестнадцатеричной в двоичную:
  - Перевести  $4AF_{16}$  (также  $0x4AF$ ) в двоичную
  -
- Из шестнадцатеричной в десятичную:
  - Перевести  $4AF_{16}$  в десятичную
  -

# Префиксы

- Нижние индексы неудобны
- В некоторых ЯП используют префиксы:
  - Шестнадцатеричный: 0x
    - $0x23AB = 23AB_{16}$
  - Двоичный: 0b
    - $0b1101 = 1101_2$

# Тема 1: Основы

**Системы  
счисления: байты,  
нибблы и все такое**

# Биты, байты, нибблы

- Байт: 8 бит
  - Всего \_\_\_\_\_ значений
  - [\_\_, \_\_]
- Ниббл: 4 бита
  - Всего \_\_\_\_\_ значений
  - [\_\_, \_\_]

Одна двоичная цифра – это \_\_ бит

Одна hex цифра – это \_\_\_\_ бит или \_\_\_\_ нибблов

Две hex цифры – это \_\_\_\_ байт

Старший – слева

Младший – справа

# Еще степени двойки

- $2^{10} = 1$  кило  $\approx 10^3$  (1024)
- $2^{20} = 1$  мега  $\approx 10^6$  (1,048,576)
- $2^{30} = 1$  гига  $\approx 10^9$  (1,073,741,824)
- $2^{40} = 1$  тера  $\approx 10^{12}$
- $2^{50} = 1$  пета  $\approx 10^{15}$
- $2^{60} = 1$  экса  $\approx 10^{18}$

# Оценки степеней двойки

- Чему равно  $2^{24}$ ?

В

- Насколько большое число можно представить 32 битами?

В

Тема 1: Основы

**Системы  
счисления:  
сложение**

# Сложение

- Десятичное



- Двоичное



# Двоичное сложение

- Сложите два четырехбитовых числа

$$\begin{array}{r} 1001 \\ + 0101 \\ \hline \end{array}$$

- Сложите два четырехбитовых числа

$$\begin{array}{r} 1011 \\ + 0110 \\ \hline \end{array}$$

# Переполнение

- Цифровые системы оперируют с **заданным количеством бит**
- Переполнение: когда результат слишком содержит больше бит, чем возможно сохранить
- Второй пример на предыдущем слайде

Тема 1: Основы

**Системы  
счисления: числа  
со знаком**

# Двоичные числа со знаком

- Прямой код
- Дополнительный код

# Прямой код

- 1 знаковый бит,  $N-1$  бит самого числа
- Знаковый бит – старший

- Положительное: знак = 0
- Отрицательное: знак = 1

$$A: \{a_{N-1}, a_{N-2}, \dots, a_2, a_1, a_0\}$$

$$A = (-1)^{a_{N-1}} \sum_{i=0}^{N-2} a_i 2^i$$

- Пример, 4-битовое представление  $\pm 6$ :

$$+6 =$$

$$-6 = 1100$$

- Диапазон  $N$ -битового числа в прямом коде:

$$[-(2^{N-1}), \dots, 2^{N-1}]$$

# Прямой код

## Проблемы:

- Не работает сложение, например  $-6 + 6$ :

$$\begin{array}{r} 1110 \\ + 0110 \\ \hline 10100 \text{ (неправильно!)} \end{array}$$

- Два представления нуля  $0 (\pm 0)$ :

$$\begin{array}{r} 1000 \\ 0000 \end{array}$$

# Дополнительный код

- Проблемы прямого кода отсутствуют:
  - Сложение работает
  - Ноль имеет единственное представление

# Дополнительный код

- Старший бит имеет вес  $-2^{N-1}$

$$A = a_{N-1}(-2^{N-1}) + \sum_{i=0}^{N-2} a_i 2^i$$

- Самое большое число из 4 б
- Самое маленькое число из 4 б
- Старший бит по-прежнему определяет знак (1 = отрицательный, 0 = положительный)
- Диапазон чисел в доп. коде:

$$[-(2^{N-1}), 2^{N-1}-1]$$

# Смена знака

- Как поменять знак числа в дополнительном коде?
- **Способ:**
  1. Инвертировать все биты
  2. Добавить 1
- **Пример:** изменить знак числа  $3_{10} =$   
0011
  1. 
  2.  $1101 = -3_{10}$

# Примеры доп. кода

- Изменить знак числа  $6_{10} = 0110_2$ 
  - 1.
  - 2.
  
- Перевести  $1001_2$  из дополнительного кода в десятичную систему
  - 1.
  - 2.

# Сложение в доп. коде

- Сложим  $6 + (-6)$  в доп. коде

$$\begin{array}{r} 0110 \\ + 1010 \\ \hline \end{array}$$

- Сложим  $-2 + 3$  в доп. коде

$$\begin{array}{r} 1110 \\ + 0011 \\ \hline \end{array}$$

# Вычитание

- Вычитание сводится к замене знака и сложению
- Пример:  $3 - 5 = 3 + (-5)$

$$\begin{array}{r} 0011 \quad 3 \\ + 1011 \quad -5 \\ \hline 1110 \quad -2 \end{array}$$

# Сравнение числовых систем

Система	Диапазон
Беззнаковая	$[0, 2^N-1]$
Прямой код	$[-(2^{N-1}-1), 2^{N-1}-1]$
Доп. код	$[-2^{N-1}, 2^{N-1}-1]$

Например, для четырех битов:

Тема 1: Основы

**Системы  
счисления:  
расширение**

# Расширение чисел

**Расширение представления числа из  $N$  в  $M$  бит ( $M > N$ ) :**

- **Знаковое расширение** для чисел в дополнительном коде
- **Расширение нулями** для беззнаковых чисел

# Знаковое расширение

- Бит знака копируется во все старшие разряды
- Величина числа остается прежней
- **Пример 1:**
  - 4-битовое представление 3: **0011**
  - 8-битовое представление: **00000011**
- **Пример 2:**
  - 4-битовое представление -5: **1011**
  - 8-битовое представление: **11111011**

# Расширение нулями

- Во все старшие разряды копируется ноль
- Значение отрицательных чисел меняется
- **Пример 1:**
  - 4-битовое представление:  $0011 = 3_{10}$
  - 8-битовое представление: **0000**0011 =  $3_{10}$
- **Пример 2:**
  - 4-битовое представление:  $1011 = -5_{10}$
  - 8-битовое представление: **0000**1011 =  $11_{10}$

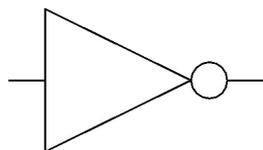
Тема 1: Основы

# **Логические вентили**

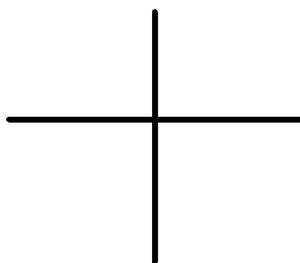
# Логические вентили

- **Выполняют логические операции:**
  - НЕ, И, ИЛИ, НЕ-И, НЕ-ИЛИ, ...
- **С одним входом:**
  - Инвертор (НЕ), буфер
- **С двумя входами:**
  - И, ИЛИ, НЕ-И, НЕ-ИЛИ, исключающее ИЛИ (XOR)
- **С несколькими входами**

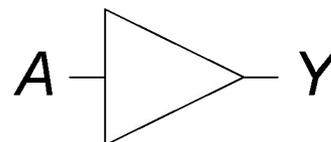
# Вентили с одним входом



—



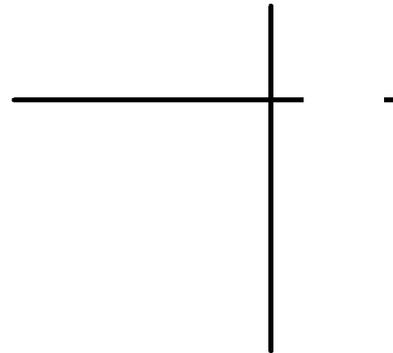
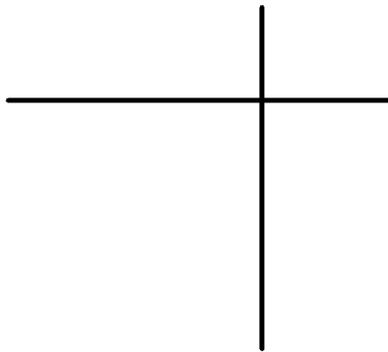
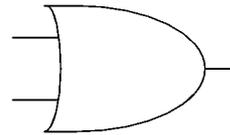
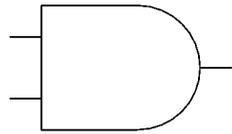
**BUF**



$$Y = A$$

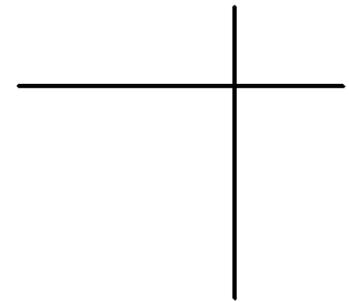
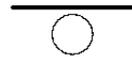
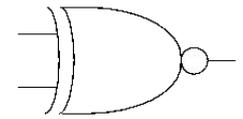
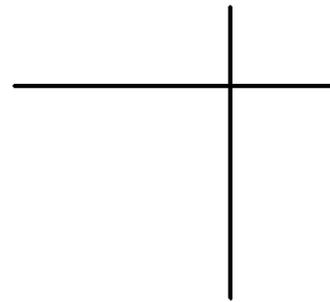
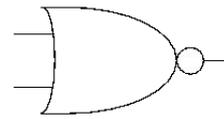
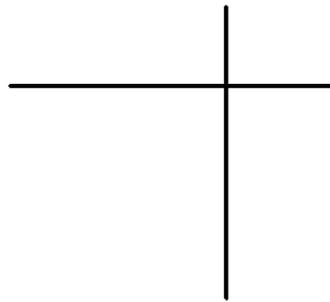
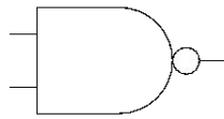
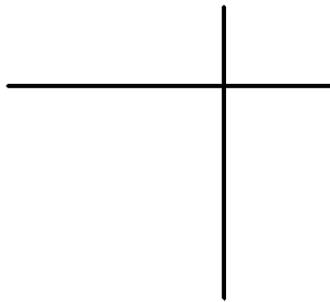
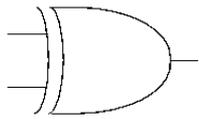
A	Y
0	
1	

# Вентили с двумя входами

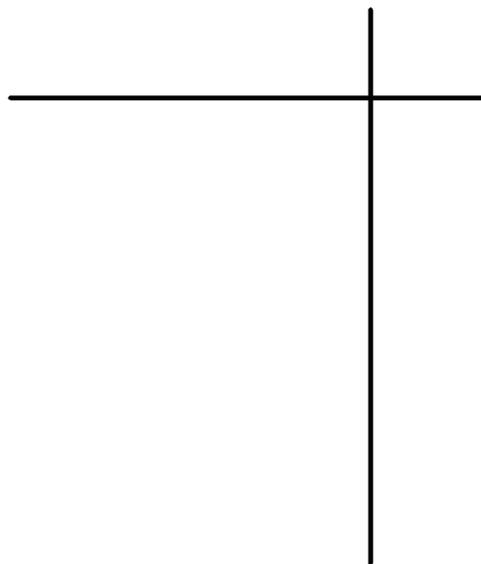
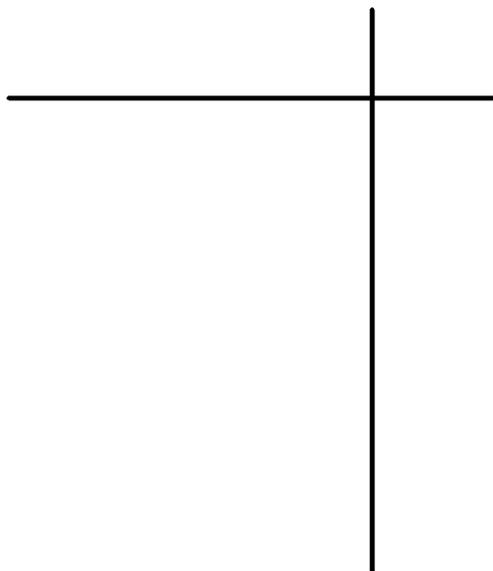
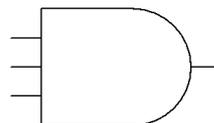
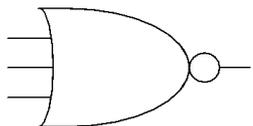


# Вентили с двумя входами

---



# Вентили с многими входами



- XOR с многими входами: проверка

# Описание на SystemVerilog

```
module gates(input logic a, b, c,  
             output logic y1, y2, y3, y4, y5);  
  
    not g1(y1, a);  
    and g2(y2, a, b);  
    or g3(y3, a, b, c);  
    nand g4(y4, b, c);  
    xor g5(y5, a, c);  
endmodule
```

Тема 1: Основы

# Логические уровни

# Логические уровни

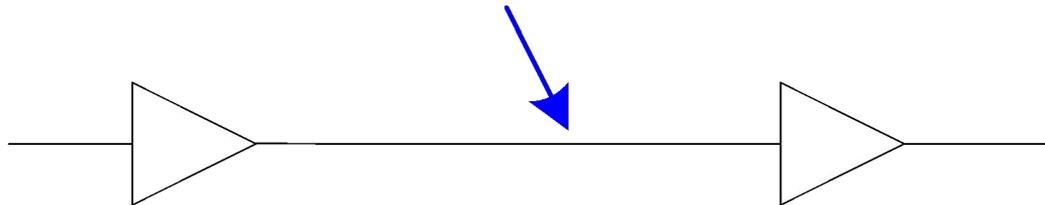
- Дискретные напряжения определяют 1 и 0
- Например:
  - 0 = земля (GND) или 0 вольт
  - 1 =  $V_{DD}$  или 5 вольт
- Как насчет 4.99 вольт? Это 0 или 1?
- А как насчет 3.2 вольт?

# Логические уровни

- *Диапазон* напряжений для 1 и 0
- Разные диапазоны для входных и выходных уровней для учета влияния *шума*

# Что такое шум?

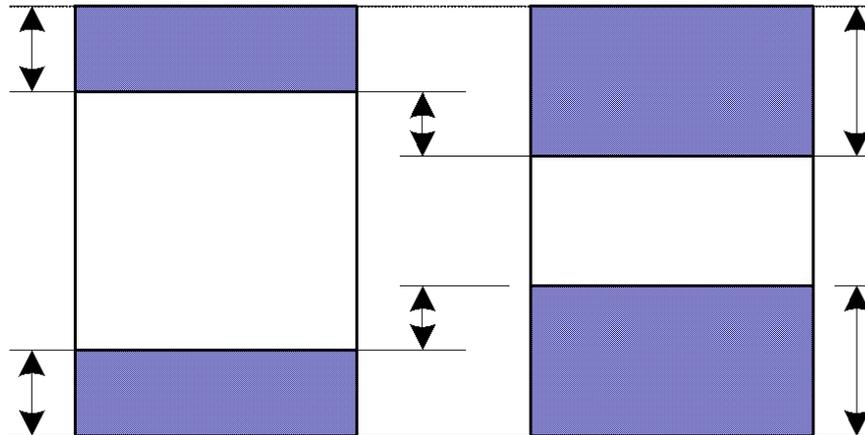
- **То, что приводит к ухудшению сигнала**
  - Сопротивления, наводки, помехи...
- **Пример:** вентиль выдает 5 В, но сопротивление снижает это напряжение до 4.5 В



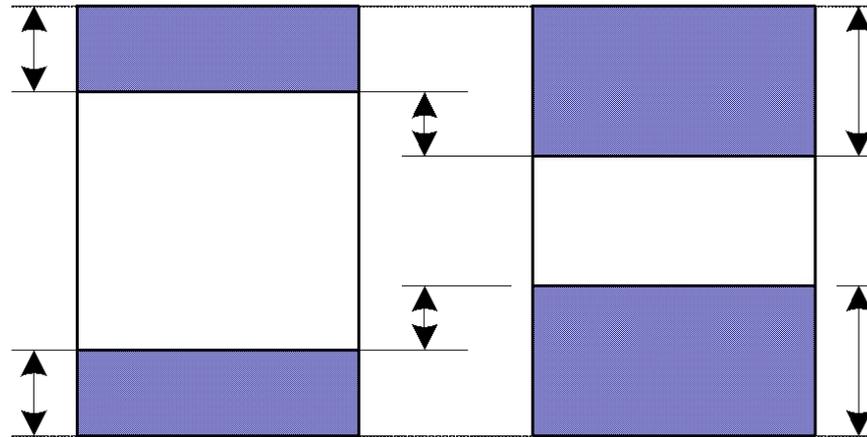
# Статическая дисциплина

- Для любых допустимых значений входов любой элемент цепи должен гарантировать правильное значение на выходе
- Нужно использовать ограниченные диапазоны напряжений для представления логических уровней

# Запасы помехоустойчивости



# Запасы помехоустойчивости

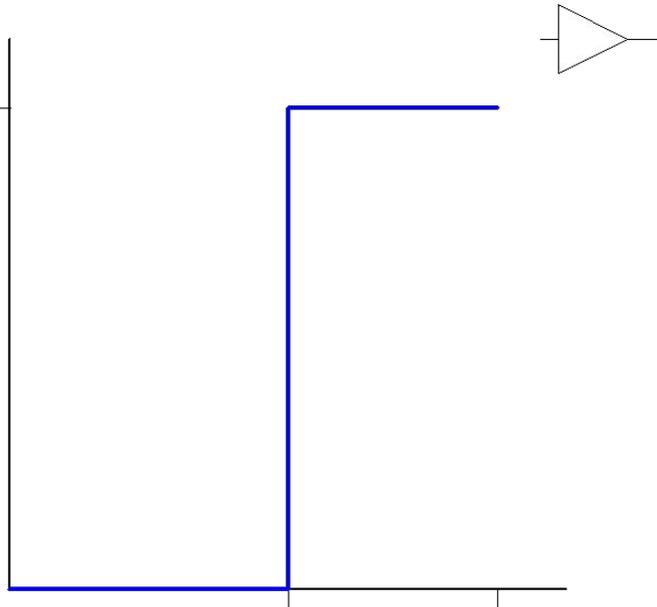


Верхний запас:  $NM_H =$

Нижний запас:  $NM_L =$

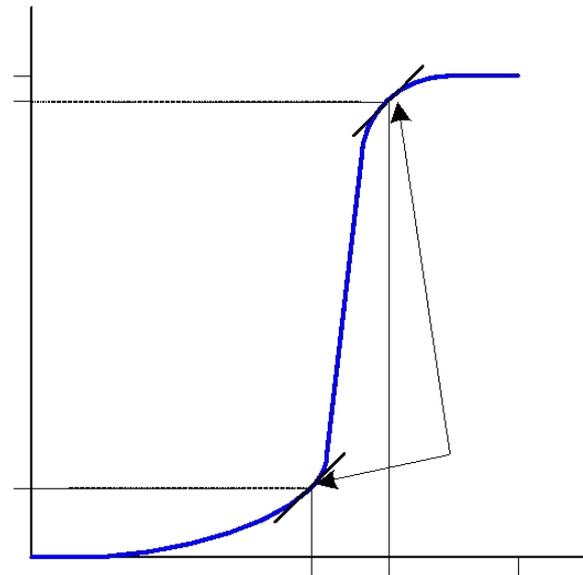
# Передаточные

Идеальный буфер:



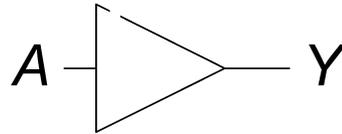
$$NM_H = NM_L = V_{DD}/2$$

Реальный буфер:



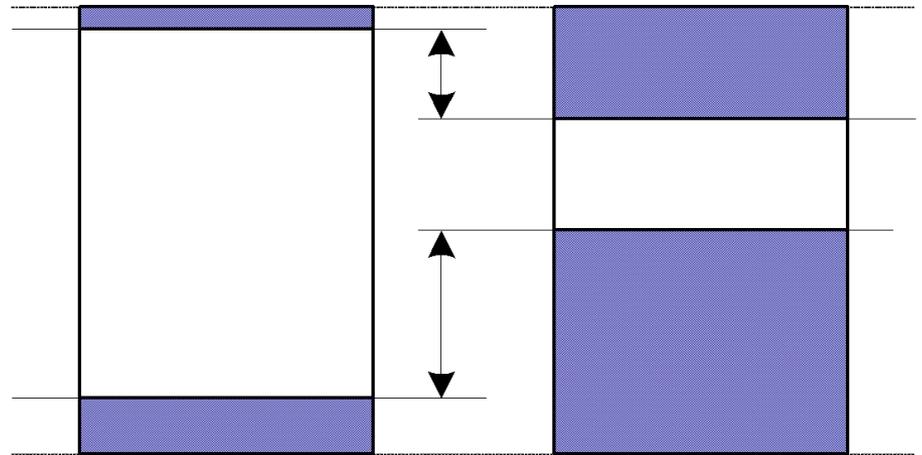
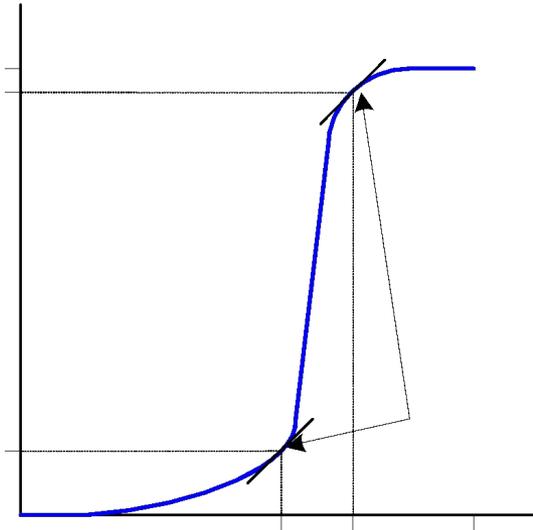
$$NM_H, NM_L < V_{DD}/2$$

# Передаточные



...

.....



# Уровень $V_{DD}$

- В 1970-х и 1980-х,  $V_{DD} = 5 \text{ В}$
- Уровень  $V_{DD}$  снизился
  - Транзисторы могут сгореть
  - Нужно низкое энергопотребление
- 3.3 В, 2.5 В, 1.8 В, 1.5 В, 1.2 В, 1.0 В, ...
  - Нужно быть аккуратным, соединяя схемы с разным уровнем напряжения питания

# Логические семейства

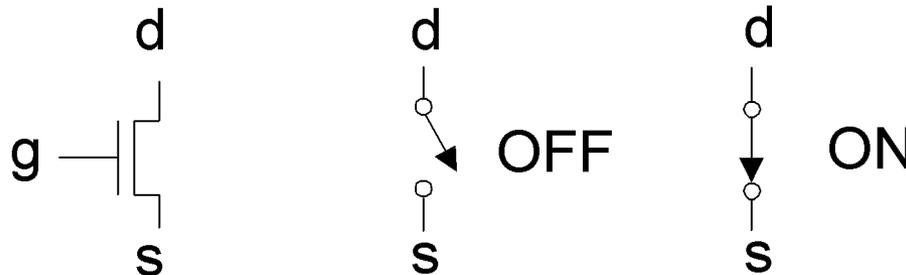
Семейство	$V_{DD}$	$V_{IL}$	$V_{IH}$	$V_{OL}$	$V_{OH}$
<b>ТТЛ</b>	5 (4.75 - 5.25)	0.8	2.0	0.4	2.4
<b>КМОП</b>	5 (4.5 - 6)	1.35	3.15	0.33	3.84
<b>НВТТЛ</b>	3.3 (3 - 3.6)	0.8	2.0	0.4	2.4
<b>НВКМОП</b>	3.3 (3 - 3.6)	0.9	1.8	0.36	2.7

Тема 1: Основы

**КМОП транзисторы**

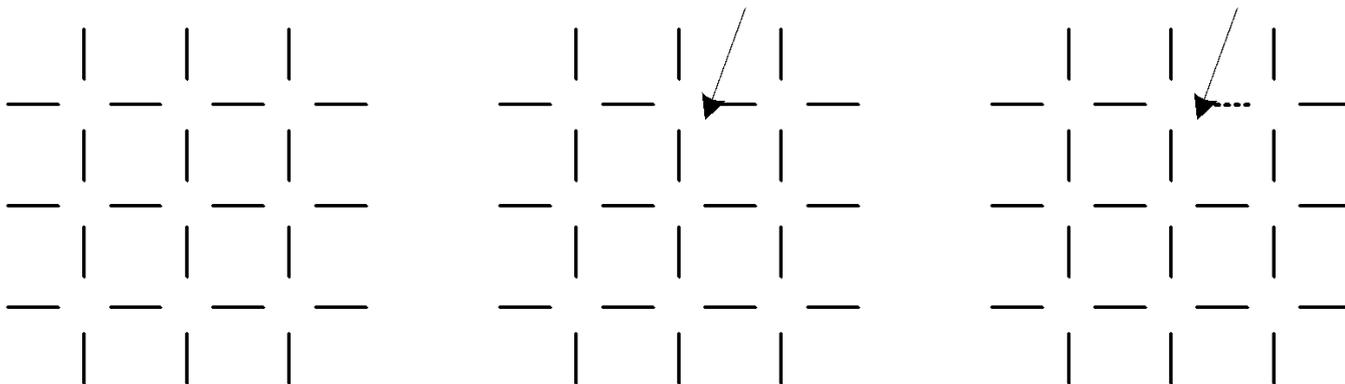
# Транзисторы

- Вентили построены на транзисторах
- Транзистор – управляемый ключ
  - 2 соединены в зависимости от напряжения на третьем
  - d и s соединены (ON) если  $g = 1$



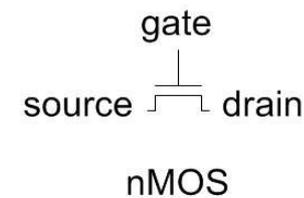
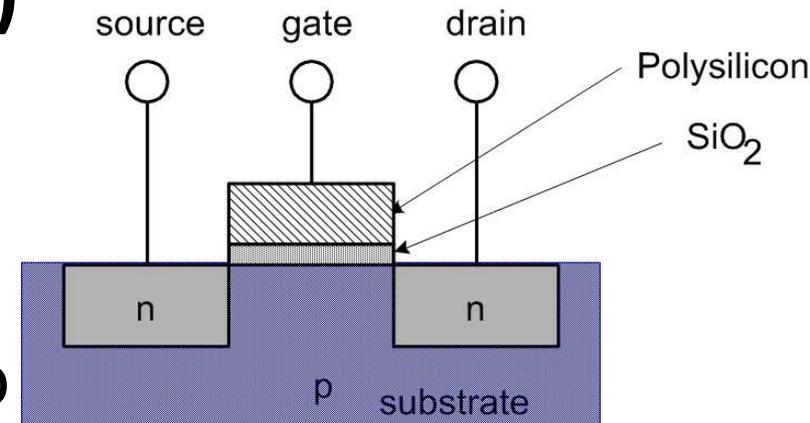
# Кремний

- Транзисторы сделаны из кремния
- Чистый кремний – плохой проводник
- Кремний с примесями – хороший проводник
  - n-тип (свободные электроны, *n*egative)
  - p-type (свободные дырки, *p*ositive)



# МОП транзисторы

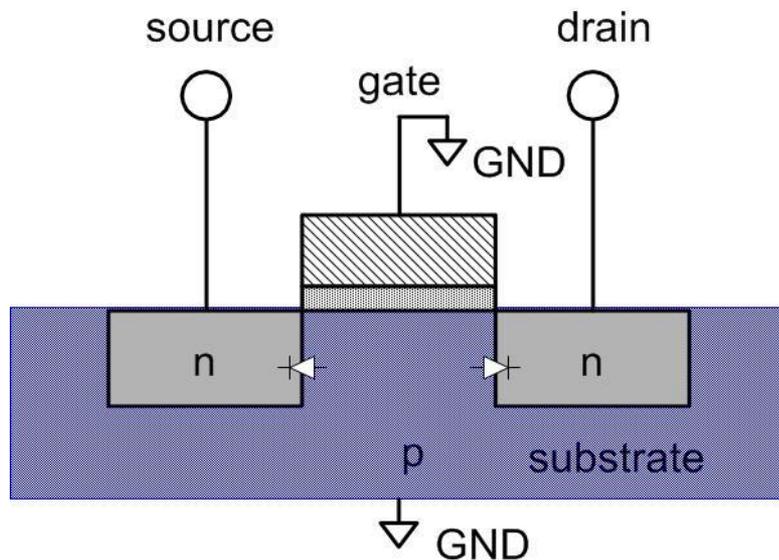
- **Металл-оксид-полупроводник (МОП) транзисторы:**
  - Поликремниевый (в прошлом **металлический**) затвор
  - **Оксидный** (диоксид Si) изолятор
  - Легированный **полупроводник**



# Транзисторы: нМОП

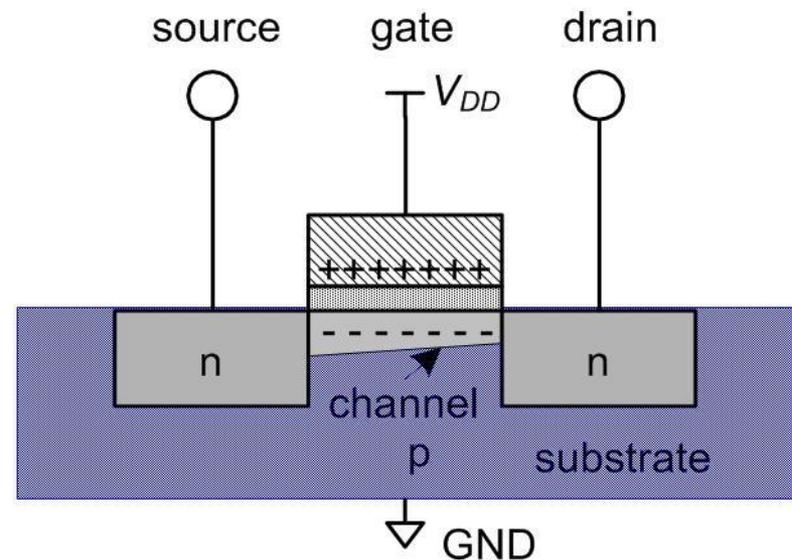
**Затвор = 0**

**OFF** (СТОК И ИСТОК НЕ СВЯЗАНЫ)



**Затвор = 1**

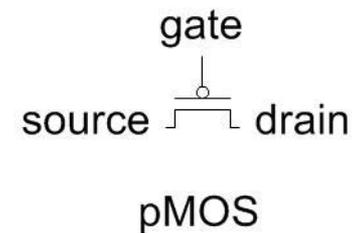
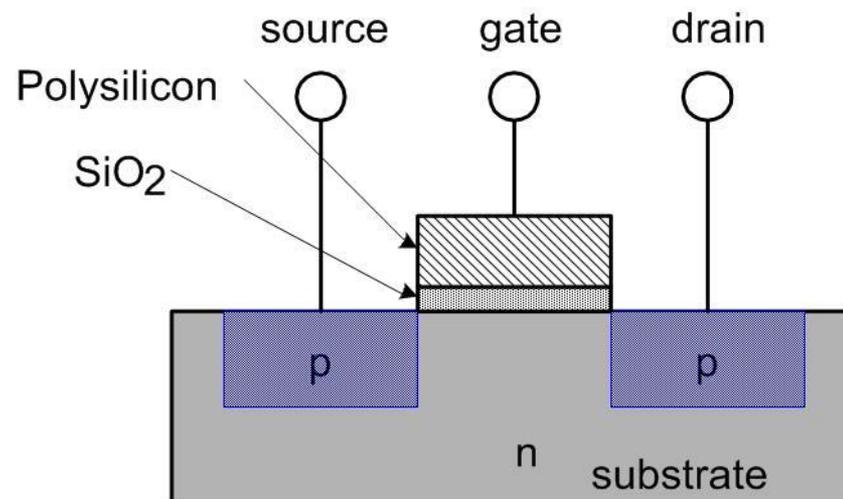
**ON** (СТОК И ИСТОК СВЯЗАНЫ)



# Транзисторы: пМОП

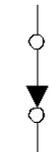
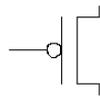
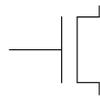
## пМОП транзисторы противоположны нМОП

- ON когда Gate = 0
- OFF когда Gate = 1



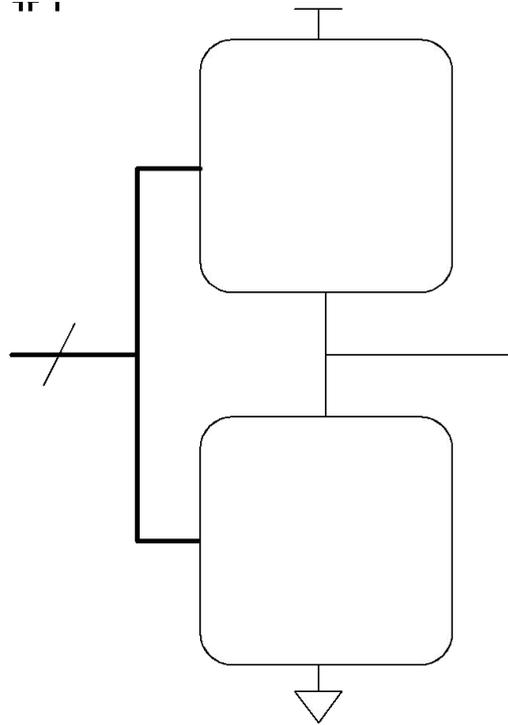
# Транзисторы

с -



# Транзисторы

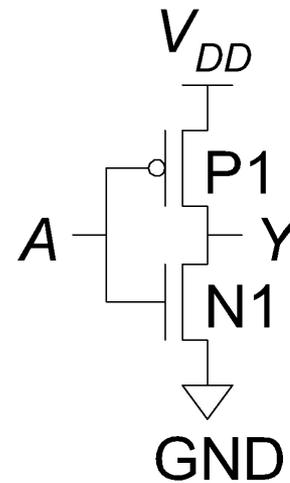
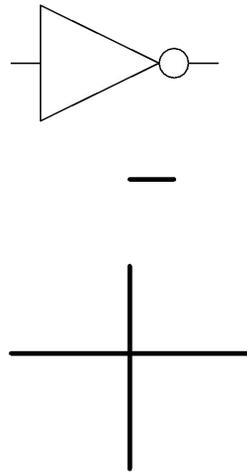
- **нМОП**: пропускают **0**, подключать к GND
- **пМОП**: пропускают **1**, подключать к  $V_{DD}$



Тема 1: Основы

# **Вентили и транзисторы**

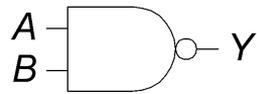
# КМОП вентили: НЕ



$A$	P1	N1	$Y$
0			
1			

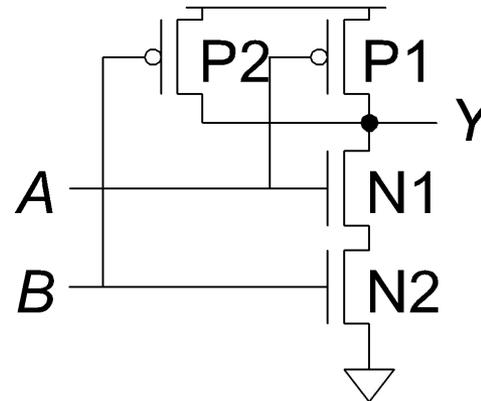
# КМОП вентили: НЕ-И

## NAND



$$Y = \overline{AB}$$

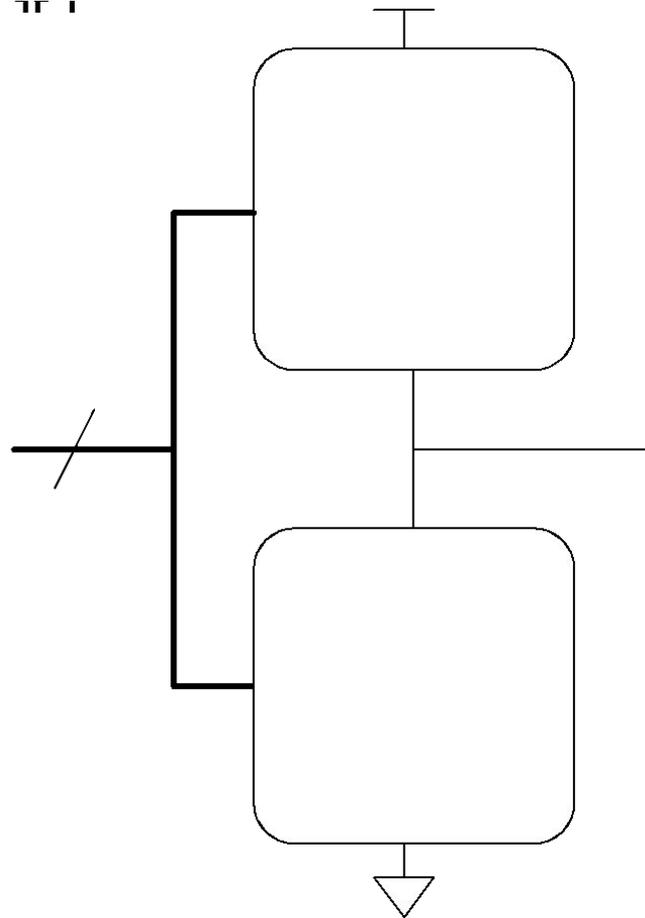
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0



A	B	P1	P2	N1	N2	Y
0	0					
0	1					
1	0					
1	1					

# Структура КМОП вентиля

рис. 1.1



# Вентиль HE-I-3

Как построить вентиль HE-I-3?

# Вентиль И

Как построить вентиль И?

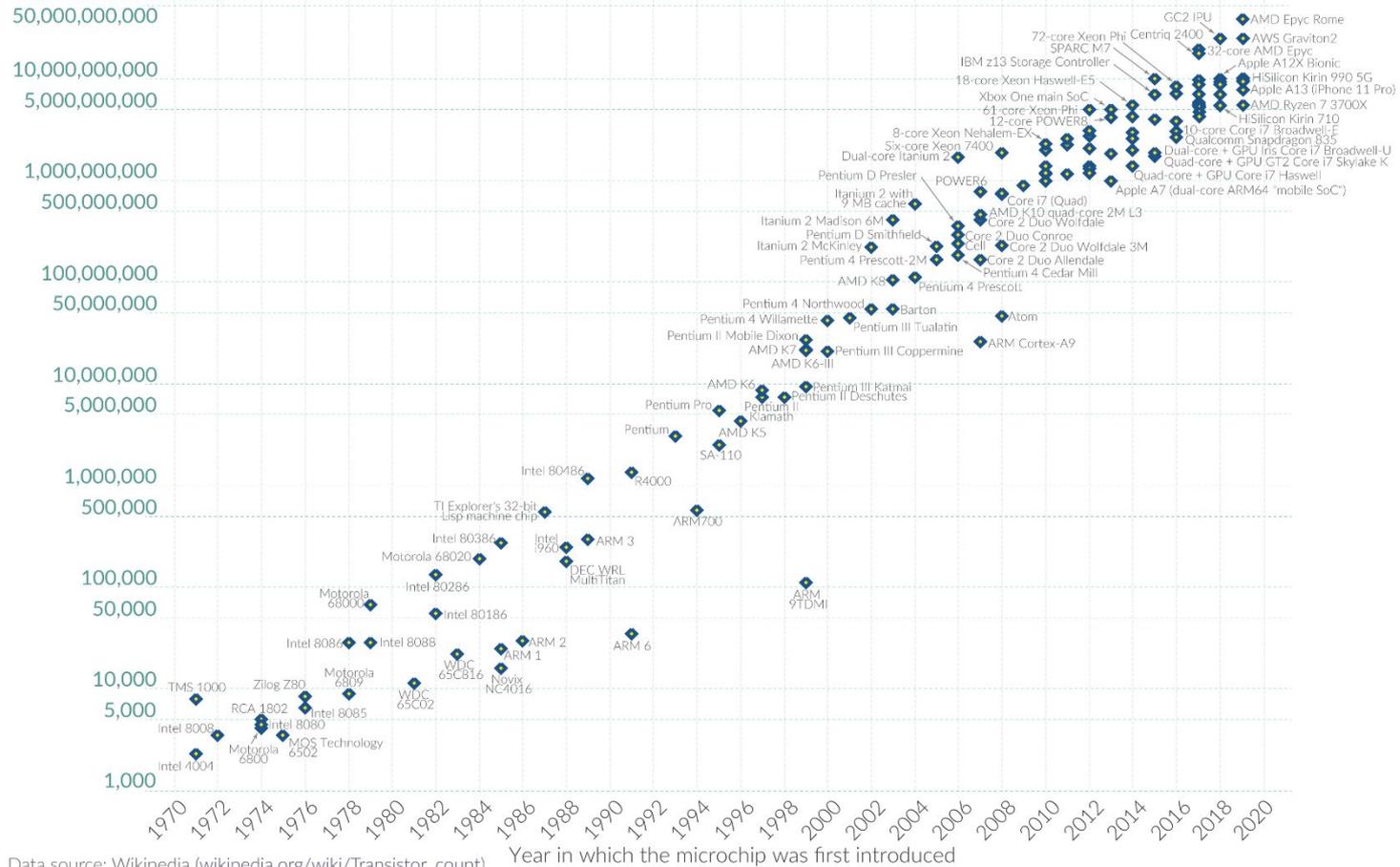
# Закон Мура

## Moore's Law: The number of transistors on microchips doubles every two years

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.

Our World  
in Data

### Transistor count



Тема 1: Основы

# Энергопотреблени е

# Энергопотребление

**Мощность = Энергия, потребляемая в единицу времени**

- **Динамическое энергопотребление**
- **Статическое энергопотребление**

# Динамическое потребление

- **Мощность для зарядки емкостей**
  - Энергия для зарядки емкости  $C$  до напряжения  $V_{DD}$  равна  $CV_{DD}^2$
  - Схема работает на частоте  $f$
  - Емкость заряжается  $\alpha$  раз за цикл (разрядка не тратит энергию)
- **Динамическое энергопотребление:**

$$P_{dynamic} = \alpha CV_{DD}^2 f$$

# Коэффициент активности $\alpha$

- **$\alpha$  это доля цикла, потраченная на зарядку**
  - Сигнал часов  $\alpha = 1$
  - Изменение раз в цикл  $\alpha = 0.5$
  - Случайный сигнал  $\alpha = 0.25$
  - Типичный сигнал  $\alpha = 0.1$

# Статическое потребление

- Энергия, потребляемая, когда значения на вентилях не меняются
- Вызвано *токами утечки*  $I_{DD}$
- Статическое энергопотребление:

$$P_{static} = I_{DD} V_{DD}$$

# Единицы измерения

- Схемы имеют маленькие емкости, низкие токи и высокие частоты.

Префикс	Обозначение	Величина
Тера	Т	$10^{12}$
Гига	Г	$10^9$
Мега	М	$10^6$
Кило	к	$10^3$
Милли	м	$10^{-3}$
Микро	мк	$10^{-6}$
Нано	н	$10^{-9}$
Пико	п	$10^{-12}$
Фемто	ф	$10^{-15}$

# Пример

- Оценить энергопотребление смартфона во время работы мобильной игры
  - $V_{DD} = 0.8 \text{ В}$
  - $C = 5 \text{ нФ}$
  - $f = 2 \text{ ГГц}$
  - $\alpha = 0.1$
  - $I_{DD} = 100 \text{ мА}$

$$\begin{aligned} P &= \alpha C V_{DD}^2 f + I_{DD} V_{DD} \\ &= (0.1)(5 \text{ нФ})(0.8 \text{ В})^2(2 \text{ ГГц}) + (100 \text{ мА})(0.8 \text{ В}) \\ &= (0.64 + 0.08) \text{ Вт} \approx 0.72 \text{ Вт} \end{aligned}$$

# Пример

- Если у батарея телефона имеет емкость 4000 мАч, сколько времени она проживет в режиме ожидания?

- $V_{DD} = 0.8 \text{ В}$

- $I_{DD} = 100 \text{ мА}$

$$P_{static} = I_{DD} V_{DD} = 0.08 \text{ Вт}$$

$$\begin{aligned} \text{Время работы} &= (\text{Емкость}/1000) * V_{DD} / \text{Потребление} \\ &= (4000 \text{ мАч} / 1000) * 0.8 \text{ В} / (0.08 \text{ Вт}) \\ &= 40 \text{ часов} \end{aligned}$$