

## Лабораторная 4-5

### Логические функции и устройства

**Цель занятия:** закрепление понятий, связанных с логической базой конструирования микросхем и представления целых чисел в дополнительном коде.

#### Пример.

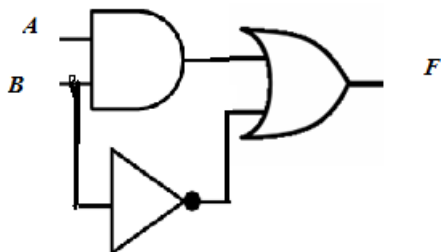
*Таблица истинности:*

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

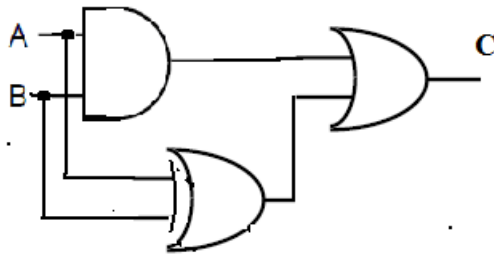
*Представление таблицы истинности логической функцией:  $F = \bar{A}\bar{B} + A\bar{B} + AB$ .*

*Упрощение логического выражения:  $F = \bar{A}\bar{B} + A\bar{B} + AB = \bar{B}(A + \bar{A}) + AB = \bar{B} + AB$ .*

*Схема соответствующего абстрактного устройства:*



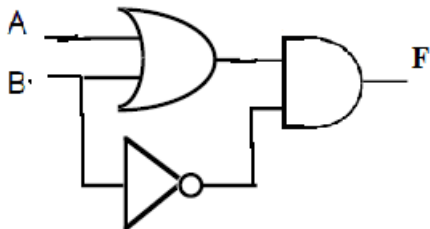
1. Записать таблицы истинности для логических функций, заданных следующими выражениями:
  - a.  $F = \text{not } (a \text{ and } b)$ ,
  - b.  $F = (\text{not } a \text{ or not } b) \text{ and } a$ ,
  - c.  $F = \text{not } (a \text{ or } b) \text{ xor } (\text{not } a \text{ and not } b)$ .
2. Выразить операцию **xor** через базовые логические операции.
3. Записать таблицу истинности для схемы абстрактных устройств:



4. Изобразить схему абстрактных устройств, реализующих логическую функцию:

$$F = (\text{not } a \text{ and } b) \text{ or } b$$

5. Записать логическую функцию, соответствующую схеме абстрактных устройств:



## *Представление целых чисел в дополнительном коде*

1. Процессоры семейства 80x86 трактуют отрицательные числа, как двоичные дополнения (которые содержат единичный бит в старшем разряде). Чтобы получить отрицательное число надо инвертировать все биты и добавить единицу.
  - a. определите какие из нижеперечисленных чисел являются отрицательными:
    - i. 0x8000
    - ii. 0x100
    - iii. 0x7fff
    - iv. 0x0ffff
    - v. 0x0fff
2. Из представления отрицательных чисел в виде двоичного дополнения следует, что если мы копируем отрицательное число в слово (переходим от 8-битового представления к 16-битовому), то старший байт надо заполнить единицами (если число положительное, то, очевидно, нулями).
  - a. перейдите от 8- к 16-битовому представлению
    - i. 0x80
    - ii. 0x28
    - iii. 0x9a
    - iv. 0x7f
  - b. перейдите от 16- к 8-битовому представлению (это не всегда возможно!)
    - i. 160
    - ii. 416
    - iii. -448
3. Выполните следующие операции не переходя к десятичному представлению
  - a.  $0x1234 + 0x9876$
  - b.  $0xffff - 0f34$
  - c.  $0x100 - 0x1$
  - d.  $0x0ffe - 1$
4. Перейдите к двоичному представлению
  - a. 0xDEAD
  - b. 0xADD
  - c. 0x1024
  - d. 0xFFFF