



Министерство науки и высшего образования Российской Федерации
Калужский филиал
федерального государственного бюджетного
образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и управление»


КАФЕДРА ИУК2 «Информационные системы и сети»

ЛАБОРАТОРНАЯ РАБОТА №7

«Логические основы функционирования ЭВМ»

ДИСЦИПЛИНА: «Теоретическая информатика»

Выполнил: студент гр. ИУК4-11Б


(подпись)

(Суриков Н.С)
(Ф.И.О.)

Проверил:

(Гладских А.П)
(подпись) (Ф.И.О.)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Цель: сформировать практические навыки анализа функциональных узлов компьютерных систем, навыки выбора архитектуры вычислительных систем сосредоточенной обработки информации

ВАРИАНТ 21

Задача 1:

Найти значение выражения:

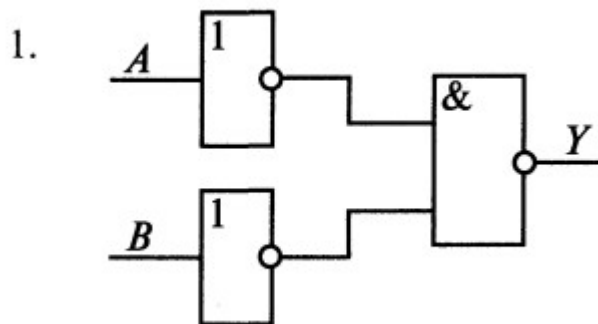
21) $(x \geq y) \text{ OR } (z > -4)$ при а) $x = 5, y = 7, z = 0$; б) $x = 5, y = -7, z = 10$;

а) $5 \geq 7 \text{ OR } 0 > -4 = \text{False OR False} = \text{False}$

б) $5 \geq -7 \text{ OR } 10 > -4 = \text{True OR True} = \text{True}$

Задача 2:

По заданной логической схеме составить логическое выражение и заполнить для него таблицу истинности:



Логическое выражение: **NOT(NOT(A) AND NOT(B))**

Если подумать(применить Закон Де Моргана), то выражение можно упростить до: **A OR B**

Тогда таблица истинности будет выглядеть следующим образом:

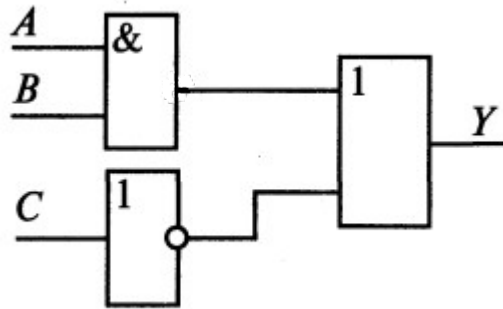
A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

Задача 3:

По заданному логическому выражению составить логическую схему и построить таблицу истинности:

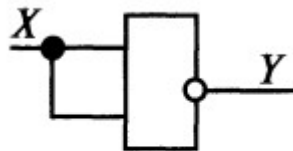
1.A AND B OR NOT C

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



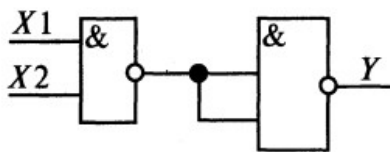
Задача 4.1:

Построить таблицу истинности для логических элементов реализованных через базовый И—НЕ:



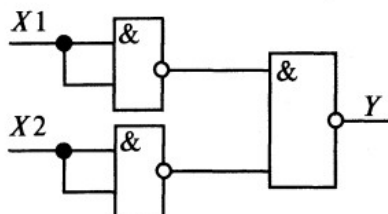
НЕ

X	F
0	1
1	0



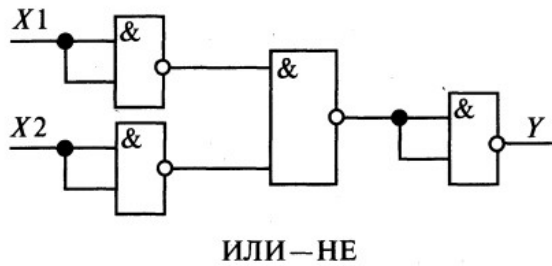
И

X1	X2	F
0	0	0
0	1	0
1	0	0
1	1	1



ИЛИ

X1	X2	F
0	0	0
0	1	1
1	0	1
1	1	1



X1	X2		F
0	0		1
0	1		0
1	0		0
1	1		0

Листинг программы для выполнения задания 4.1:

```

1  from itertools import product
2
3  print('X | F')
4  for X, in product((0, 1), repeat=1):
5      print(X, '|', int(not (X and X)))
6
7  def F1(x1, x2):
8      return int(not ((not (x1 and x2)) and (not (x1 and x2))))
9
10 print('X1 X2 | F')
11 for X1, X2 in product((0, 1), repeat=2):
12     print(X1, X2, '|', F1(X1, X2))
13
14 def F2(x1, x2):
15     return int(not ((not (x1 and x1)) and (not (x2 and x2))))
16
17 print('X1 X2 | F')
18 for X1, X2 in product((0, 1), repeat=2):
19     print(f'{X1} {X2} | {F2(X1, X2)}')
20
21 def F3(x1, x2):
22     return int(not ((F2(x1, x2) and F2(x1, x2))))
23
24 print('X1 X2 | F')
25 for X1, X2 in product((0, 1), repeat=2):
26     print(f'{X1} {X2} | {F3(X1, X2)}')
27

```

Задача 4.2:

Разработать схемы реализации элементов НЕ, И, ИЛИ, И—НЕ через базовый логический элемент ИЛИ—НЕ:

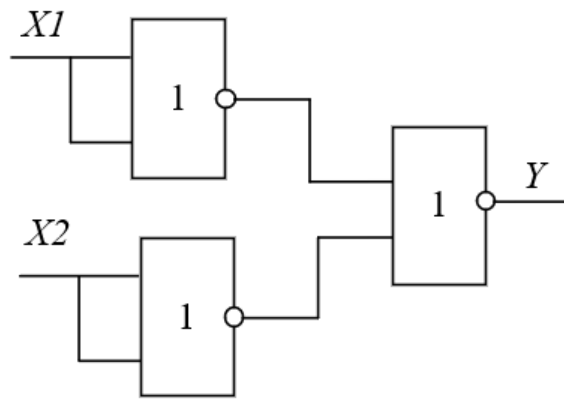


Схема «И»

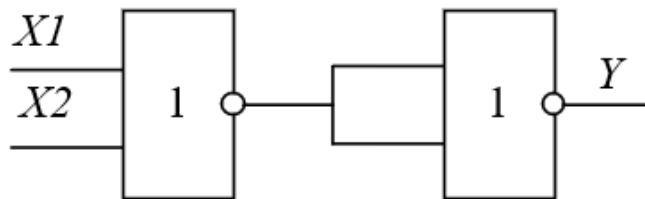


Схема «ИЛИ»

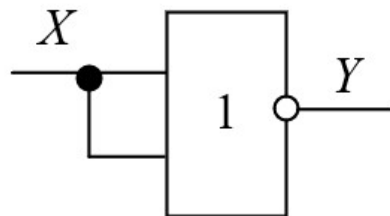


Схема «НЕ»

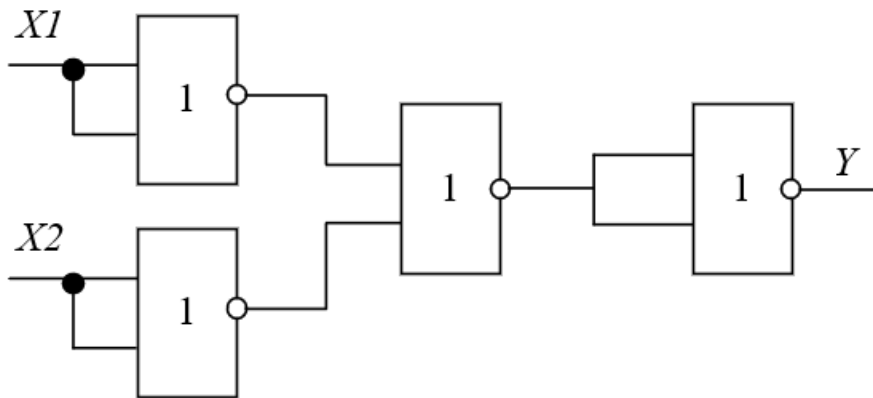
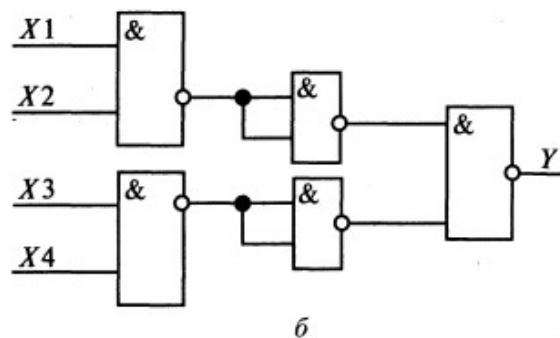
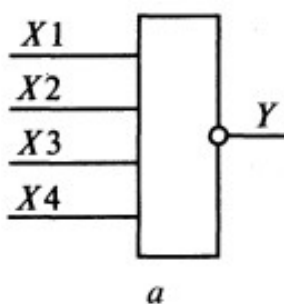


Схема «И-НЕ»

Задача 5:

Проверить, что четырехходовый элемент, изображенный на рис. А, эквивалентен комбинации двухходововых элементов, изображенной на рис. б.



X1	X2	X3	X4		F1		F2
0	0	0	0		1		1
0	0	0	1		1		1
0	0	1	0		1		1
0	0	1	1		1		1
0	1	0	0		1		1
0	1	0	1		1		1
0	1	1	0		1		1
0	1	1	1		1		1
1	0	0	0		1		1
1	0	0	1		1		1
1	0	1	0		1		1
1	0	1	1		1		1
1	1	0	0		1		1
1	1	0	1		1		1
1	1	1	0		1		1
1	1	1	1		0		0

Листинг программы для выполнения задания 5:

```

1  from itertools import product
2
3  def F1(x1, x2, x3, x4):
4      return int(not (x1 and x2 and x3 and x4))
5
6  def F2(x1, x2, x3, x4):
7      return int(
8      not (
9      (not (((not (x1 and x2)) and (not (x1 and x2)))))
10     and (not (((not (x3 and x4)) and (not (x3 and x4)))))
11     )
12     )
13
14  print("X1 X2 X3 X4 | F1 | F2")
15  for x1, x2, x3, x4 in product((0, 1), repeat=4):
16      print(
17      f"{x1} {x2} {x3} {x4} \
18      | {F1(x1, x2, x3, x4)} \
19      | {F2(x1, x2, x3, x4)}"
20      )

```

Задача 6:

Схема реализует арифметическое действие $A + B = C_0S$, где A и B — одноразрядные двоичные числа, C_0 и S — соответственно старший и младший двоичные разряды суммы

Проверить, что имеют место логические формулы:

$$S = (\bar{A} \wedge B) \vee (A \wedge \bar{B}), \quad C_0 = A \wedge B$$

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Листинг программы для выполнения задания 6:

```
1 from itertools import product
2
3 def S(a, b):
4     return int(((not (a) and b) or (a and not (b))))
5
6 def C(a, b):
7     return int(a and b)
8
9 print("A B | C | S")
10 for A, B in product((0, 1), repeat=2):
11     print(f"{A} {B} | {C(A, B)} | {S(A, B)}")
12
```

Задача 7:

Для сложения двух двоичных разрядов A и B многоразрядного числа с учетом возможного добавления цифры C_i оставшейся от сложения предыдущих разрядов используется так называемый одноразрядный сумматор.

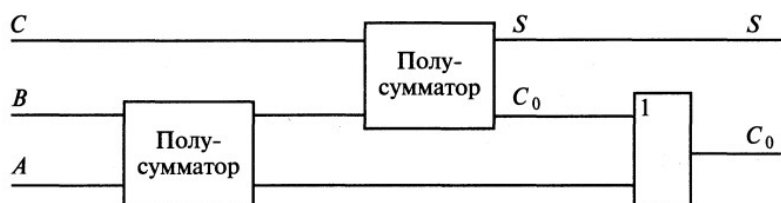


Рис. 4.8. Логическая схема одноразрядного сумматора

Проверить перебором всех возможных вариантов, что схема на рис. 4.8 действительно реализует указанное выше действие:

A	B	C		C0		S
0	0	0		0		0
0	0	1		0		1
0	1	0		0		1
0	1	1		1		0
1	0	0		0		1
1	0	1		1		0
1	1	0		1		0
1	1	1		1		1

Листинг программы для выполнения задания 7:

```
1  from itertools import product
2
3  def S(a, b):
4      return int(((not (a) and b) or (a and not (b))))
5
6  def C0(a, b):
7      return int(a and b)
8
9  def F(a, b, c):
10     s1 = S(a, b)
11     c0 = C0(a, b)
12     s2 = S(s1, c)
13     c1 = C0(s1, c)
14     return s2, (c0 or c1)
15
16 print("A B C | C0 | S")
17 for A, B, C in product((0, 1), repeat=3):
18     print(f"{A} {B} {C} | {F(A, B, C)[1]} | {F(A, B, C)[0]}")
19
```


Задача 8:

Разобрать на примерах работу трехразрядного сумматора:

A1	B1	A2	B2	A3	B3	S1	S2	S3	S4
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1	0
0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	1	0	0	0	1
0	0	0	1	0	0	0	1	0	0
0	0	0	1	0	1	0	1	1	0
0	0	0	1	1	0	0	1	1	0
0	0	0	1	1	1	0	1	0	1
0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	1	0	1	1	0
0	0	1	0	1	0	0	1	1	0
0	0	1	0	1	1	0	1	0	1
0	0	1	1	0	0	0	0	1	0
0	0	1	1	0	1	0	0	0	1
0	0	1	1	1	0	0	0	0	1
0	0	1	1	1	1	0	0	1	1
0	1	0	0	0	0	1	0	0	0
0	1	0	0	0	1	1	0	1	0
0	1	0	0	1	0	1	0	1	0
0	1	0	0	1	1	1	0	0	1
0	1	0	1	0	0	1	1	0	0
0	1	0	1	0	1	1	1	1	0
0	1	0	1	1	0	1	1	1	0
0	1	0	1	1	1	1	1	0	1
0	1	1	0	0	0	1	1	0	0
0	1	1	0	0	1	1	1	1	0
0	1	1	0	1	0	1	1	1	0
0	1	1	0	1	1	1	1	0	1
0	1	1	1	0	0	1	0	1	0
0	1	1	1	0	1	1	1	1	0
0	1	1	1	1	0	1	0	0	1
0	1	1	1	1	1	1	1	0	1
1	0	0	0	0	0	1	0	0	0
1	0	0	0	0	1	1	0	1	0
1	0	0	0	1	0	1	0	1	0
1	0	0	0	1	1	1	0	0	1
1	0	0	1	0	0	1	1	0	0
1	0	0	1	0	1	1	1	1	0
1	0	0	1	1	0	1	1	1	0
1	0	0	1	1	1	1	1	0	1
1	0	1	0	0	0	1	1	0	0
1	0	1	0	0	1	1	1	1	0
1	0	1	0	1	0	1	1	1	0
1	0	1	0	1	1	1	1	0	1
1	0	1	1	0	0	1	1	0	0
1	0	1	1	1	0	1	0	0	1
1	0	1	1	1	1	1	0	1	1
1	1	0	0	0	0	0	1	0	0
1	1	0	0	0	1	0	1	1	0
1	1	0	0	1	0	0	1	1	0
1	1	0	0	1	1	0	1	0	1
1	1	0	1	0	0	0	0	1	0
1	1	0	1	0	1	0	0	0	1
1	1	0	1	1	0	0	0	0	1
1	1	0	1	1	1	0	0	1	1
1	1	1	0	0	0	0	0	1	0
1	1	1	0	0	1	0	0	0	1
1	1	1	0	1	0	0	0	0	1
1	1	1	0	1	1	0	0	1	1
1	1	1	1	0	0	0	1	0	1
1	1	1	1	0	1	0	1	0	1
1	1	1	1	1	0	0	1	1	0
1	1	1	1	1	1	0	1	0	1
1	1	1	1	1	1	0	1	0	1

Листинг программы для выполнения задания 8:

```
1  from itertools import product
2
3  def S(a, b):
4  return int(((not (a) and b) or (a and not (b))))
5
6  def C0(a, b):
7  return int(a and b)
8
9  def oSum(a, b, c):
10 s1 = S(a, b)
11 c0 = C0(a, b)
12 s2 = S(s1, c)
13 c1 = C0(s1, c)
14 return s2, (c0 or c1)
15
16 def pSum(a, b):
17 return S(a, b), C0(a, b)
18
19 print("A1 B1 A2 B2 A3 B3 | S1 | S2 | S3 | S4")
20 for A1, B1, A2, B2, A3, B3 in product((0, 1), repeat=6):
21 s1, c0 = pSum(A1, B1)
22 s2, c1 = oSum(c0, A2, B2)
23 s3, s4 = oSum(c1, A3, B3)
24 print(
25 f"{A1} {B1} {A2} {B2} {A3} {B3} \
26 | {s1} | {s2} | {s3} | {s4}"
27 )
28
29
```

Вывод: благодаря проделанной работе были сформированы практические навыки анализа функциональных узлов компьютерных систем, навыки выбора архитектуры вычислительных систем сосредоточенной обработки информации.

Литература

1. Тюльпинова, Н. В. Алгоритмизация и программирование : учебное пособие / Н. В. Тюльпинова. — Саратов : Вузовское образование, 2019. — 200 с. — ISBN 978-5-4487-0470-3. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROОбразование : [сайт]. — URL: <https://profspo.ru/books/80539>
2. Соснин В.В. Облачные вычисления в образовании / Соснин В.В.. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. — 109 с. — ISBN 978-5-4486-0512-3. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/79705.html>
3. Шаманов А.П. Системы счисления и представление чисел в ЭВМ : учебное пособие / Шаманов А.П.. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2016. — 52 с. — ISBN 978-5-7996-1719-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/66204.html>
4. Минитаева А.М. Кодирование информации. Системы счисления. Основы логики : учебное пособие / Минитаева А.М.. — Москва : Московский государственный технический университет имени Н.Э. Баумана, 2019. — 108 с. — ISBN 978-5-7038-5244-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/110640.html>
5. Широков А.И. Информатика: разработка программ на языке программирования Питон: базовые языковые конструкции : учебник / Широков А.И., Пышняк М.О.. — Москва : Издательский Дом МИСиС, 2020. — 142 с. — ISBN 978-5-907226-76-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/106713.html>