

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»
(БГТУ им. В.Г.Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №3

Дисциплина: Теория цифровых автоматов

по теме Синтез и анализ комбинационных схем с одним
выходом с учетом неопределенностей

Выполнил: ст. группы ВТ-32
Воскобойников

Проверил: Рязанов Ю. Д.

Цель работы: научиться строить эффективные по быстродействию и затратам оборудования комбинационные схемы с учетом неопределенностей.

Задание

1. Составить таблицу истинности заданной частично определенной булевой функции (см. варианты заданий в таблице 2). Булева функция здесь задана двумя условиями (условие 1 и условие 2), зависящими от значений аргументов. Если на наборе аргументов условие 2 истинно, то значение функции на этом наборе не определено. Если же на наборе аргументов условие 2 ложно, то значение функции на этом наборе равно значению условия 1 на этом наборе аргументов. В условии значение аргумента отождествляется с двоичной цифрой, а последовательность аргументов — с двоичным числом. Для составления таблицы истинности рекомендуется написать программу
2. Решить задачу минимизации частично определенной булевой функции в классе дизъюнктивных нормальных форм.
3. Написать программу, строящую таблицу истинности булевой функции, полученной при выполнении п. 2
- 2 Сравнить полученную таблицу с таблицей истинности исходной частично определенной булевой
4. Применить факторизационный метод синтеза многоярусной комбинационной схемы в базисе И-ИЛИ-НЕ с двухвходовыми элементами И и ИЛИ по полученной при выполнении п. 2 минимальной дизъюнктивной нормальной форме булевой функции.
5. Решить задачу минимизации частично определенной булевой функции в классе конъюнктивных нормальных форм.
6. Написать программу, строящую таблицу истинности булевой функции, полученной при выполнении п. 5
- 5 Сравнить полученную таблицу с таблицей истинности исходной частично определенной булевой
7. Применить факторизационный метод синтеза многоярусной комбинационной схемы в базисе И-ИЛИ-НЕ с двухвходовыми элементами И и ИЛИ по полученной при выполнении п. 5 минимальной конъюнктивной нормальной форме булевой функции.
8. Написать программы, моделирующие работу схем, полученных в пунктах 4 и 7, на всех входных наборах и строящие таблицу истинности каждой схемы. Сравнить полученные таблицы истинности с таблицей истинности исходной частично определенной функции.
9. Сравнить схемы, построенные в лабораторных работах №1 и №3 по Квайну и по быстродействию.

1. Составить таблицу истинности заданной частично определенной булевой функции (см. варианты заданий в таблице 2). Булева функция здесь задана двумя условиями (условие 1 и условие 2), зависящими от значений аргументов. Если на наборе аргументов условие 2 истинно, то значение функции на этом наборе не определено. Если же на наборе аргументов условие 2 ложно, то значение функции на этом наборе равно значению условия 1 на этом наборе аргументов. В условии значение аргумента отождествляется с двоичной цифрой, а последовательность аргументов — с двоичным числом. Для составления таблицы истинности рекомендуется написать программу

x_1	x_2	x_3	x_4	x_5	$(x_4x_5 + x_1x_2x_3)=0,5,8,10$	$(x_1x_2x_4)=1$	f
0	0	0	0	0	$(00+000)=0,5,8,10$	$(000)=1$	1
0	0	0	0	1	$(01+000)=0,5,8,10$	$(000)=1$	0
0	0	0	1	0	$(10+000)=0,5,8,10$	$(001)=1$	-
0	0	0	1	1	$(11+000)=0,5,8,10$	$(001)=1$	-
0	0	1	0	0	$(00+001)=0,5,8,10$	$(000)=1$	0
0	0	1	0	1	$(01+001)=0,5,8,10$	$(000)=1$	0
0	0	1	1	0	$(10+001)=0,5,8,10$	$(001)=1$	-
0	0	1	1	1	$(11+001)=0,5,8,10$	$(001)=1$	-
0	1	0	0	0	$(00+010)=0,5,8,10$	$(010)=1$	0
0	1	0	0	1	$(01+010)=0,5,8,10$	$(010)=1$	0
0	1	0	1	0	$(10+010)=0,5,8,10$	$(011)=1$	0
0	1	0	1	1	$(11+010)=0,5,8,10$	$(011)=1$	1
0	1	1	0	0	$(00+011)=0,5,8,10$	$(010)=1$	0
0	1	1	0	1	$(01+011)=0,5,8,10$	$(010)=1$	0
0	1	1	1	0	$(10+011)=0,5,8,10$	$(011)=1$	1
0	1	1	1	1	$(11+011)=0,5,8,10$	$(011)=1$	0
1	0	0	0	0	$(00+100)=0,5,8,10$	$(100)=1$	0
1	0	0	0	1	$(01+100)=0,5,8,10$	$(100)=1$	1
1	0	0	1	0	$(10+100)=0,5,8,10$	$(101)=1$	0
1	0	0	1	1	$(11+100)=0,5,8,10$	$(101)=1$	0
1	0	1	0	0	$(00+101)=0,5,8,10$	$(100)=1$	1
1	0	1	0	1	$(01+101)=0,5,8,10$	$(100)=1$	0
1	0	1	1	0	$(10+101)=0,5,8,10$	$(101)=1$	0
1	0	1	1	1	$(11+101)=0,5,8,10$	$(101)=1$	1
1	1	0	0	0	$(00+110)=0,5,8,10$	$(110)=1$	0
1	1	0	0	1	$(01+110)=0,5,8,10$	$(110)=1$	0
1	1	0	1	0	$(10+110)=0,5,8,10$	$(111)=1$	1
1	1	0	1	1	$(11+110)=0,5,8,10$	$(111)=1$	0
1	1	1	0	0	$(00+111)=0,5,8,10$	$(110)=1$	0
1	1	1	0	1	$(01+111)=0,5,8,10$	$(110)=1$	1
1	1	1	1	0	$(10+111)=0,5,8,10$	$(111)=1$	0
1	1	1	1	1	$(11+111)=0,5,8,10$	$(111)=1$	1


```

        printf(" %2d|", X[i]);
        printf(" %2d  ", TableValue6(X));

        printf("\n");
    }

    return 0;
}

```

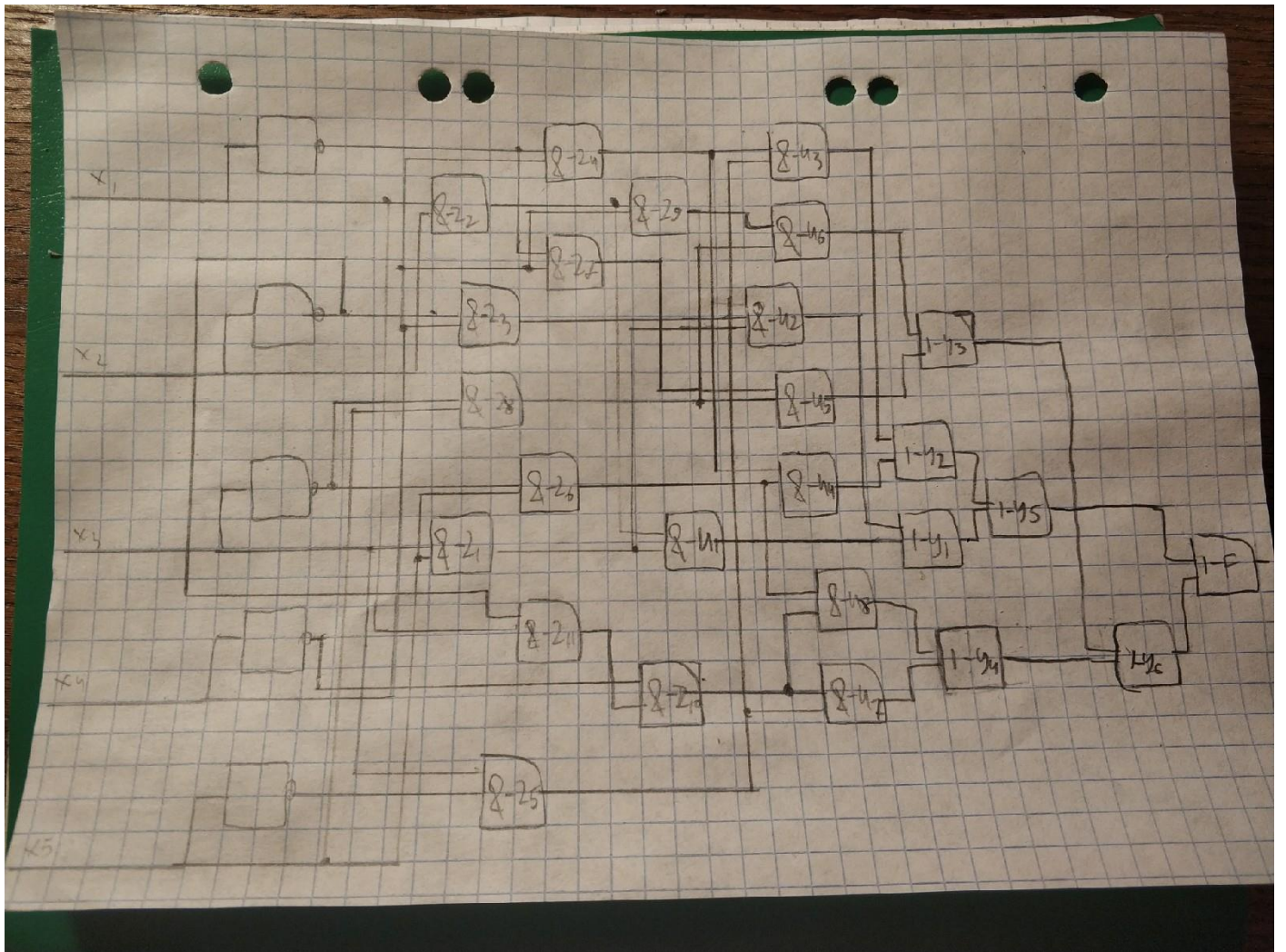
x1	x2	x3	x4	x5	F	f
0	0	0	0	0	1	1
0	0	0	0	1	0	0
0	0	0	1	0	1	-
0	0	0	1	1	1	-
0	0	1	0	0	0	0
0	0	1	0	1	0	0
0	0	1	1	0	1	-
0	0	1	1	1	1	-
0	1	0	0	0	0	0
0	1	0	0	1	0	0
0	1	0	1	0	0	0
0	1	0	1	1	1	1
0	1	1	0	0	0	0
0	1	1	0	1	0	0
0	1	1	1	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	0	0
1	0	0	0	1	1	1
1	0	0	1	0	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	1
1	0	1	0	1	0	0
1	0	1	1	0	0	0
1	0	1	1	1	1	1
1	1	0	0	0	0	0
1	1	0	0	1	0	0
1	1	0	1	0	1	1
1	1	0	1	1	0	0
1	1	1	0	0	0	0
1	1	1	0	1	1	1
1	1	1	1	0	0	0
1	1	1	1	1	1	1
1	1	1	1	1	0	0
1	1	1	1	1	1	1



частично определенная функция

4. Применить факторизационный метод синтеза многоярусной комбинационной схемы в базисе И-ИЛИ-НЕ с двухвходовыми элементами И и ИЛИ по полученной при выполнении п. 2 минимальной дизъюнктивной нормальной форме булевой функции.

[illegible][illegible]



5. Решить задачу минимизации частично определенной булевой функции в классе конъюнктивных нормальных форм.

СКНФ:

$$\begin{aligned}
 &(\overline{x_1} \vee \overline{x_2} \vee \overline{x_3} \vee \overline{x_4} \vee x_5) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3} \vee x_4 \vee \overline{x_5}) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3} \vee x_4 \\
 &\vee x_5) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3 \vee \overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3 \vee \overline{x_4} \vee x_5) \wedge (\overline{x_1} \vee \overline{x_2} \\
 &\vee x_3 \vee x_4 \vee \overline{x_5}) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3 \vee x_4 \vee x_5) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3} \vee \overline{x_4} \\
 &\vee \overline{x_5}) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3} \vee \overline{x_4} \vee x_5) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3} \vee x_4 \vee \overline{x_5}) \wedge (\overline{x_1} \vee x_2 \\
 &\vee x_3 \vee \overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_1} \vee x_2 \vee x_3 \vee \overline{x_4} \vee x_5) \wedge (\overline{x_1} \vee x_2 \vee x_3 \vee x_4 \\
 &\vee x_5) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3} \vee \overline{x_4} \vee \overline{x_5}) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3} \vee x_4 \vee \overline{x_5}) \wedge (x_1 \vee \overline{x_2} \\
 &\vee \overline{x_3} \vee x_4 \vee x_5) \wedge (x_1 \vee \overline{x_2} \vee x_3 \vee \overline{x_4} \vee x_5) \wedge (x_1 \vee \overline{x_2} \vee x_3 \vee x_4 \\
 &\vee \overline{x_5}) \wedge (x_1 \vee x_2 \vee \overline{x_3} \vee \overline{x_4} \vee \overline{x_5}) \wedge (x_1 \vee x_2 \vee \overline{x_3} \vee \overline{x_4} \vee x_5) \wedge (x_1 \vee x_2 \\
 &\vee \overline{x_3} \vee x_4 \vee x_5) \wedge (x_1 \vee x_2 \vee x_3 \vee \overline{x_4} \vee \overline{x_5}) \wedge (x_1 \vee x_2 \vee x_3 \vee x_4 \vee \overline{x_5})
 \end{aligned}$$

0	1	2	3	4	5
	00001+ 00010+ 00100+ 01000+ 10000+	00011+ 00101+ 00110+ 01001+ 01010+ 01100+ 10010+ 11000+	00111+ 01101+ 10011+ 10101+ 10110+ 11001+ 11100+	01111+ 11011+ 11110+	
	000-1+ 0001-+ 00-01+ 0-001+ 00-10+ 0-010 -0010+ 001-0+ 0-100+ 0100-+ 010-0 01-00+ -1000+ 100-0 1-000	00-11+ -0011+ 001-1+ 0-101+ -0101 0011-+ -0110+ 01-01+ -1001+ 1010- 10-10+ 0110-+ -1100+ 1001-+ 1100-+ 11-00	0-111+ 1-011 1-110 110-1 111-0		
	00--1 00-1- -001- 0--01 -0-10 001-- 0-10- -100- 01-0- -1-00	0-1-1			

	00001	00100	01000	10000	00101	01001	01010	01100	10010	11000	01101	10011	10101	10110	11001	11100	01111	11011	11110
0-010							+												
010-0			+				+												
100-0				+					+										
1-000				+						+									
-0101													+						
1010-													+						
11-00										+						+			+
1-011												+						+	
1-110														+					+
110-1															+				
111-0																+			+
00-1	+				+			+			+								
00-1-																			
-001-									+			+							
0-01	+				+	+					+								
-0-10					+				+										
001--		+			+	+													
0-10-		+						+											
-100-			+																
01-0-		+	+					+											
-1-00			+					+		+									
0-1-1					+						+						+		

Минимальная КНФ:

$$(\overline{x_1} \vee \overline{x_3} \vee x_4 \vee \overline{x_5}) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3} \vee \overline{x_5}) \wedge (\overline{x_2} \vee x_3 \vee \overline{x_4} \vee x_5) \wedge (x_1 \vee \overline{x_3} \vee x_4 \vee x_5) \wedge (x_1 \vee x_3 \vee x_4 \vee \overline{x_5}) \wedge (x_1 \vee x_2 \vee \overline{x_3} \vee x_5) \wedge (x_1 \vee x_2 \vee x_3 \vee \overline{x_5}) \wedge (\overline{x_1} \vee \overline{x_4} \vee x_5) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3) \wedge (x_2 \vee \overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_1} \vee x_3 \vee x_5)$$

6. Написать программу, строящую таблицу истинности булевой функции, полученной при выполнении п. 5 Сравнить полученную таблицу с таблицей истинности исходной частично определенной булевой

```
#include <iostream>
```

```
const int NotUsed = system("color F0");
```

```
int TableValue6(int *X)
```

```
{
    bool Value =
        (!X[1]||!X[3]||X[4]||!X[5])&&(X[1]||!X[2]||!X[3]||!X[5])&&(!X[2]||X[3]||!X[4]||X[5])&&
        X[1]||!X[3]||X[4]||X[5])&&(X[1]||X[3]||X[4]||
        !X[5])&&(X[1]||X[2]||!X[3]||X[5])&&(X[1]||X[2]||X[3]||!X[5])&&(!X[1]||!X[4]||X[5])&&(!X
        [1]||!X[2]||X[3])&&
        (X[2]||!X[4]||!X[5])&&(!X[1]||X[3]||X[5]);
    return Value;
}
```

```
int main(int argc, const char * argv[]) {
    int X[6];
```

```

for (int i = 1; i < 6; i++)
    printf(" x%d|", i);
printf(" F \n");

for (X[1] = 0; X[1] < 2; X[1]++)
    for (X[2] = 0; X[2] < 2; X[2]++)
        for (X[3] = 0; X[3] < 2; X[3]++)
            for (X[4] = 0; X[4] < 2; X[4]++)
                for (X[5] = 0; X[5] < 2; X[5]++)
                {
                    for (int i = 1; i < 6; i++)
                        printf(" %2d|", X[i]);
                    printf(" %2d ", TableValue6(X));

                    printf("\n");
                }

return 0;
}

```

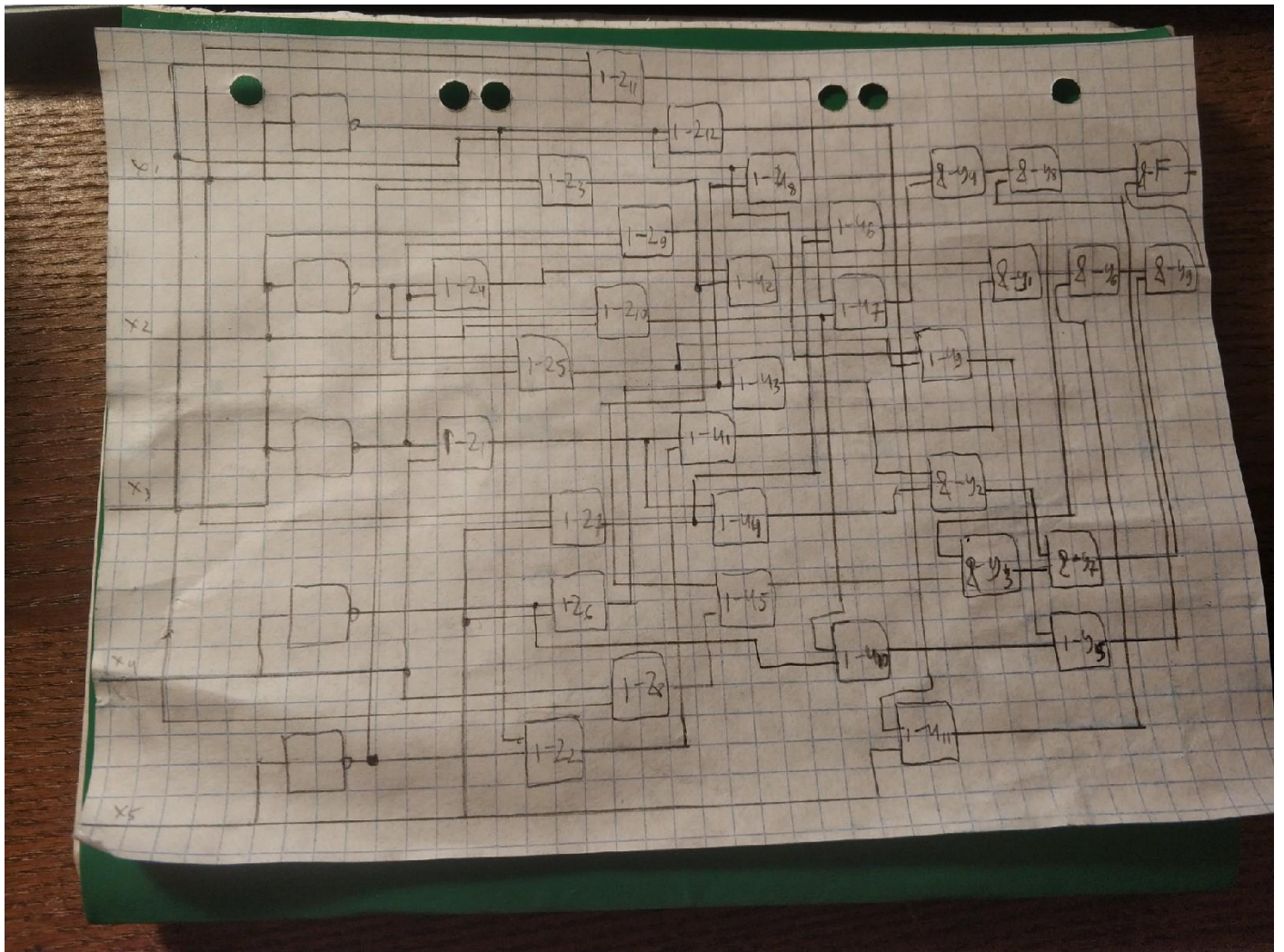
x1	x2	x3	x4	x5	F	f
0	0	0	0	0	1	1
0	0	0	0	1	0	0
0	0	0	1	0	0	-
0	0	0	1	1	0	-
0	0	1	0	0	0	0
0	0	1	0	1	0	0
0	0	1	1	0	0	-
0	0	1	1	1	0	-
0	1	0	0	0	0	0
0	1	0	0	1	0	0
0	1	0	1	0	0	0
0	1	0	1	1	1	1
0	1	1	0	0	0	0
0	1	1	0	1	0	0
0	1	1	1	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	0	0
1	0	0	0	1	1	1
1	0	0	1	0	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	1
1	0	1	0	1	0	0
1	0	1	1	0	0	0
1	0	1	1	1	1	1
1	1	0	0	0	0	0
1	1	0	0	1	0	0
1	1	0	1	0	1	1
1	1	0	1	1	0	0
1	1	1	0	0	0	0
1	1	1	0	1	1	1
1	1	1	1	0	0	0
1	1	1	1	1	1	1



частично определенная функция

7. Применить факторизационный метод синтеза многоярусной комбинационной схемы в базисе И-ИЛИ-НЕ с двухвходовыми элементами И и ИЛИ по полученной при выполнении п. 5 минимальной конъюнктивной нормальной форме булевой функции.

[illegible][illegible]



8. Написать программы, моделирующие работу схем, полученных в пунктах 4 и 7, на всех входных наборах и строящие таблицу истинности каждой схемы. Сравнить полученные таблицы истинности с таблицей истинности исходной частично определенной функции.

Исходный код:

```
bool FuncDNF(bool *x)
{
    bool z1 = x[2] && x[4],
        z2 = x[0] && x[1],
        z3 = !x[1] && x[3],
        z4 = !x[0] && x[3],
        z5 = x[2] && !x[4],
        z6 = !x[2] && x[4],
        z7 = !x[0] && !x[1],
        z8 = !x[2] && !x[4],
        z9 = x[3] && z2,
        z11 = x[0] && !x[1],
        z10 = !x[3] && z11;

    bool u1 = z1 && z2,
        u2 = z1 && z3,
        u3 = z4 && z5,
        u4 = z4 && z6,
        u5 = z7 && z8,
        u6 = z8 && z9,
        u7 = z5 && z10,
        u8 = z6 && z10;
```

```

    bool y1 = u1 || u2,
           y2 = u3 || u4,
           y3 = u5 || u6,
           y4 = u7 || u8,
           y5 = y1 || y2,
           y6 = y3 || y4;

    return y5 || y6;
}

bool FuncKNF(bool *x)
{
    bool z1 = !x[2] || x[3],
           z2 = !x[0] || !x[4],
           z3 = x[0] || !x[4],
           z4 = !x[1] || !x[2],
           z5 = !x[1] || x[2],
           z6 = !x[3] || x[4],
           z7 = x[0] || x[4],
           z8 = x[2] || x[3],
           z9 = x[1] || !x[2],
           z10 = x[1] || !x[4],
           z11 = x[0] || x[2],
           z12 = !x[0] || x[2];

    bool u1 = z1 || z2,
           u2 = z3 || z4,
           u3 = z5 || z6,
           u4 = z1 || z7,
           u5 = z3 || z8,
           u6 = z7 || z9,
           u7 = z10 || z11,
           u8 = !x[0] || z6,
           u9 = !x[0] || z5,
           u10 = !x[3] || z10,
           u11 = x[4] || z12;

    bool y1 = u1 && u2,
           y2 = u3 && u4,
           y3 = u5 && u6,
           y4 = u7 && u8,
           y5 = u9 && u10,
           y6 = u11 && y1,
           y7 = y2 && y3,
           y8 = y4 && y5,
           y9 = y6 && y7;

    return y8 && y9;
}

```

x1	x2	x3	x4	x5	f(DNF)	f(KNF)
0	0	0	0	0	1	1
0	0	0	0	1	0	0
0	0	0	1	0	1	0
0	0	0	1	1	1	0
0	0	1	0	0	0	0
0	0	1	0	1	0	0
0	0	1	1	0	1	0
0	0	1	1	1	1	0
0	1	0	0	0	0	0
0	1	0	0	1	0	0
0	1	0	1	0	0	0
0	1	0	1	1	1	1
0	1	1	0	0	0	0
0	1	1	0	1	0	0
0	1	1	1	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	0	0
1	0	0	0	1	1	1
1	0	0	1	0	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	1
1	0	1	0	1	0	0
1	0	1	1	0	0	0
1	0	1	1	1	1	1
1	1	0	0	0	0	0
1	1	0	0	1	0	0
1	1	0	1	0	1	1
1	1	0	1	1	0	0
1	1	1	0	0	0	0
1	1	1	0	1	1	1
1	1	1	1	0	0	0
1	1	1	1	1	1	1

9. Сравнить схемы, построенные в лабораторных работах №1 и №3 по Квайну и по быстродействию.

	Лаб 1		Лаб 3	
	Сложность по Квайну	Сложность по быстродействию	Сложность по Квайну	Сложность по быстродействию
ДНФ:	63	7	57	6
КНФ:	129	8	71	6