

Министерство науки и высшего образования Российской Федерации
Южно-Уральский государственный университет
Филиал в г. Миассе
Кафедра автоматики

004.3
Н842

М.В. Носиков

ЦИФРОВАЯ СХЕМОТЕХНИКА

Учебное пособие для выполнения
лабораторных работ

Часть 1

Челябинск
Издательский центр ЮУрГУ
2023

УДК 004.312(075.8) + 621.382.82(075.8)
Н842

*Одобрено
учебно-методической комиссией
филиала ЮУрГУ в г. Миассе*

*Рецензенты:
А.Г. Малышкин, А.В. Киселев*

Носиков, М.В.
Н842 Цифровая схемотехника: учебное пособие для выполнения лабораторных работ / М.В. Носиков. – Челябинск: Издательский центр ЮУрГУ, 2023. – Ч. 1. – 107 с.

ISBN 978-5-696-05367-7

Учебное пособие является руководством для выполнения лабораторных работ по цифровой схемотехнике и основам построения цифровых и микропроцессорных устройств, способствует развитию у студентов профессиональных компетенций, обеспечивающих владение приемами проведения экспериментов, анализа полученных результатов.

Пособие предназначено для студентов, обучающихся по направлениям подготовки 27.03.04 «Управление в технических системах», 12.03.01 «Приборостроение», а также рекомендовано студентам других специальностей и направлений обучения, интересующимся вопросами моделирования цифровых устройств.

УДК 004.312(075.8) + 621.382.82(075.8)

ISBN 978-5-696-05367-7

© Издательский центр ЮУрГУ, 2023

ВВЕДЕНИЕ

Настоящее учебное пособие является руководством для выполнения лабораторных работ по цифровой схемотехнике и основам построения цифровых и микропроцессорных устройств, способствует развитию у студентов профессиональных компетенций, обеспечивающих владение приемами проведения экспериментов и анализа полученных результатов.

Современное состояние микросхемотехники характеризуется практически полным замещением устройств, выполненных на отдельных микросхемах (объединенных между собой внешними электрическими соединениями с помощью печатного или объемного монтажа) на устройства, в состав которых входят большие и сверхбольшие интегральные схемы программируемой логики (программируемые логические интегральные схемы, ПЛИС) и так называемые системы на кристалле (SystemOnChip, SoC), объединяющие в одном корпусе микропроцессорное вычислительное ядро и перепрограммируемое логическое ядро, позволяющее гибко реализовать необходимый состав периферийных устройств, контроллеров, интерфейсов к внешним устройствам оперативной и энергонезависимой памяти и т.д. Данная концепция проектирования позволяет оперативно вносить изменения как в программное обеспечение микропроцессора, так и изменяя функциональные возможности логического устройства, распараллелить процесс обработки входной информации между микропроцессором и «жесткой» логикой и т.д.

С учетом достаточно высокой трудоемкости проектирования сложных вычислительных устройств в современных системах автоматизированного проектирования (САПР), автором ставилась задача освоения студентами методологии применения ПЛИС на основе выполнения лабораторных работ с использованием базовых блоков цифровой схемотехники, с постепенным усложнением логической структуры и переходу к проектированию логических устройств с использованием языков описания аппаратуры (Hardware Description Languages, HDL).

В части 1 содержатся лабораторные работы, посвященные проектированию комбинационных логических устройств.

Пособие предназначено для студентов, обучающихся по направлениям подготовки бакалавриата 27.03.04 «Управление в технических системах», 12.03.01 «Приборостроение», а также рекомендовано студентам других специальностей и направлений обучения, интересующимся вопросами моделирования цифровых устройств.

ЛАБОРАТОРНАЯ РАБОТА № 1

ИССЛЕДОВАНИЕ ПОВЕДЕНИЯ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ И ЛОГИЧЕСКИХ СХЕМ НА ИХ ОСНОВЕ

Цель работы: исследование логического функционирования и отработка практических навыков использования базовых логических элементов и синтеза логических схем на их основе. Изучение этапов создания проекта в среде проектирования ALTERAQUARTUS и практическое закрепление данных навыков. Изучение порядка работы (включения, отключения, безопасной работы) с цифровыми вычислительными устройствами. Изучение принципов анализа логических схем на аппаратных инструментальных средствах, построенных на базе программируемых логических интегральных микросхем (ПЛИС).

Оборудование: лабораторный комплекс «ALT-DIG-002».

Теоретический материал

Математическим аппаратом, используемым при анализе и синтезе цифровых устройств является алгебра логики (булева алгебра), оперирующая сигналами (переменными), имеющими лишь 2 устойчивых состояния («да»/«нет», «истина»/«ложь», «включено»/«отключено», «лог.1»/«лог. 0»). Основные постулаты алгебры логики следующие:

1. Переменные принимают только два значения: «0» и «1» (лог. «0» и лог. «1»). Такие переменные называют булевыми переменными.

2. В алгебре логики определено отношение эквивалентности (=).

Отношение эквивалентности удовлетворяет следующим свойствам:

а) $x = x$ – рефлексивность;

б) если $x = y$, то $y = x$ – симметричность;

в) если $x = y$ и $y = z$, то $x = z$ – транзитивность.

3. Из отношения эквивалентности следует принцип подстановки: если $x = y$, то в любой формуле, содержащей x , вместо x можно подставить y и будет получена эквивалентная формула.

4. В алгебре логики определены 3 базовые операции:

а) дизъюнкция (операция «логического ИЛИ», «логического сложения»), обозначаемая символами: $+$ или $\vee$;

б) конъюнкция (операция «логического И», «логического умножения»), обозначаемая символами: $\cdot$ или $\wedge$;

в) отрицание (операция «НЕ», «инверсия»), обозначаемая символами: $\neg$ либо чертой над переменной.

Используются следующие аксиомы алгебры логики:

$$\begin{cases} a = 0, \text{ если } a \neq 1 \\ a = 1, \text{ если } a \neq 0 \end{cases} \quad (1.1)$$

$$\begin{cases} \bar{0} = 1 \\ \bar{1} = 0 \end{cases} \quad (1.2)$$

$$\begin{cases} 0 + 0 = 0 \\ 0 + 1 = 1 \\ 1 + 1 = 1 \end{cases} \quad (1.3)$$

$$\begin{cases} 0 \cdot 0 = 0 \\ 0 \cdot 1 = 0 \\ 1 \cdot 1 = 1 \end{cases} \quad (1.4)$$

Также применимы следующие законы и тождества алгебры логики:

1. Закон нулевого множества:

$$\begin{aligned} 0 \vee a &= a \\ 0 \cdot a &= 0 \\ 0 \cdot a \cdot b \cdot \dots \cdot w &= 0 \end{aligned} \quad (1.5)$$

2. Закон универсального множества:

$$\begin{aligned} 1 \cdot a &= a \\ 1 \vee a &= 1 \\ 1 \vee a \vee b \vee \dots \vee w &= 1 \end{aligned} \quad (1.6)$$

3. Закон повторения (тавтология):

$$\begin{aligned} a \cdot a \cdot \dots \cdot a &= a \\ a + a + \dots + a &= a \end{aligned} \quad (1.7)$$

4. Закон двойной инверсии:

$$\bar{\bar{a}} = a \quad (1.8)$$

5. Закон дополнения:

$$\begin{aligned} a \cdot \bar{a} &= 0 \\ a + \bar{a} &= 1 \end{aligned} \quad (1.9)$$

6. Коммутативный (переместительный) закон:

$$\begin{aligned} a \cdot b &= b \cdot a \\ a + b &= b + a \end{aligned} \quad (1.10)$$

7. Ассоциативный (сочетательный) закон:

$$\begin{aligned}a \cdot (b \cdot c) &= (a \cdot b) \cdot c = a \cdot b \cdot c \\a + (b + c) &= (a + b) + c = a + b + c\end{aligned}\tag{1.11}$$

8. Дистрибутивный (распределительный) закон:

$$\begin{aligned}a \cdot (b + c) &= a \cdot b + a \cdot c \\a + (b \cdot c) &= (a + b) \cdot (a + c)\end{aligned}\tag{1.12}$$

9. Закон поглощения:

$$\begin{aligned}a \cdot (a + b) &= a \\a + a \cdot b &= a \\a + a \cdot b + a \cdot c + a \cdot c + \dots + a \cdot w &= a\end{aligned}\tag{1.13}$$

10. Закон инверсии (закон де Моргана, Augustus de Morgan)

$$\begin{aligned}\overline{a + b} &= \bar{a} \cdot \bar{b} \\ \overline{a \cdot b} &= \bar{a} + \bar{b}\end{aligned}\tag{1.14}$$

11. Закон склеивания

$$\begin{aligned}a \cdot b + a \cdot \bar{b} &= a \\(a + b) \cdot (a + \bar{b}) &= a\end{aligned}\tag{1.15}$$

Базовыми логическими операциями (логическими элементами, логическими функциями) являются:

1. Операция логического «ИЛИ»

Альтернативные наименования: — логическое сложение;
— дизъюнкция;
— «ИЛИ»;
— «OR».

Количество аргументов: 2 и более.

Формы записи:

$$\begin{aligned}Y &= A + B & y &= x_0 + x_1 \\Y &= A \vee B & y &= x_0 \vee x_1 \\Y &= A \vee B \vee C & y &= x_0 \vee x_1 \vee x_k\end{aligned}$$

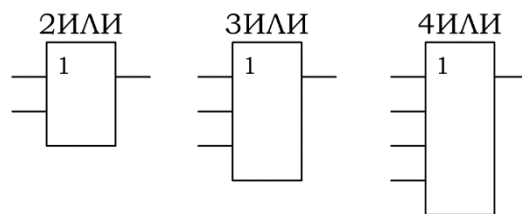
Таблица истинности для двухвходового элемента:

Входы		Выход
X_1	X_0	Y
0	0	0
0	1	1
1	0	1
1	1	1

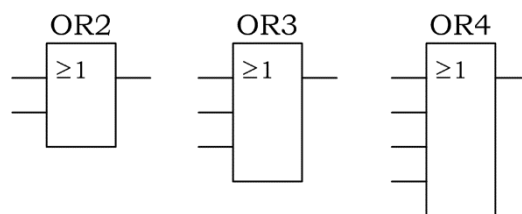
Таблица истинности для трехвходового элемента:

Входы			Выход
X_2	X_1	X_0	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

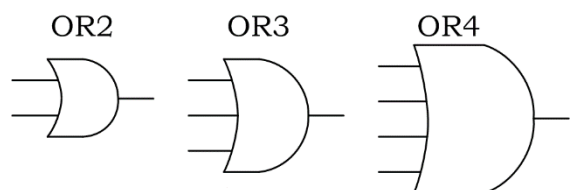
Схемотехническое обозначение по ГОСТ Р:



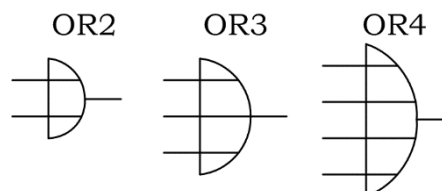
Схемотехническое обозначение по IEC:



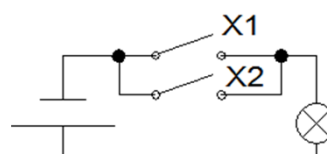
Схемотехническое обозначение по IEEE/ANSI:



Схемотехническое обозначение по DIN:



Реализация на ключевых элементах (тумблерах, кнопках, реле и т.д.):



2. Операция логического «И»

Альтернативные наименования:

- логическое умножение;
- конъюнкция;
- «И»;
- «AND».

Количество аргументов: 2 и более.

Формы записи:

$$Y = A \cdot B$$

$$Y = A \wedge B$$

$$Y = A \wedge B \wedge \dots \wedge W \quad y = x_0 \cdot x_1$$

$$y = x_0 \wedge z_1 \wedge k$$

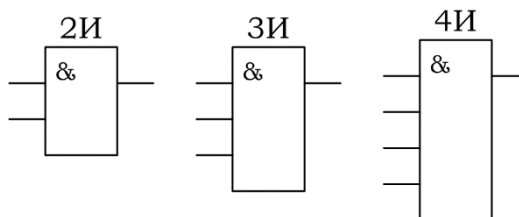
Таблица истинности для двухвходового элемента:

Входы		Выход
X_1	X_0	Y
0	0	0
0	1	0
1	0	0
1	1	1

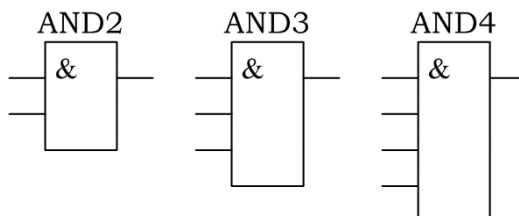
Таблица истинности для трехвходового элемента:

Входы			Выход
X_2	X_1	X_0	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

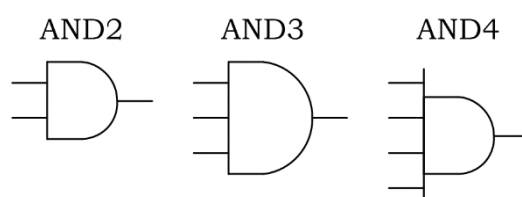
Схемотехническое обозначение по ГОСТ Р:



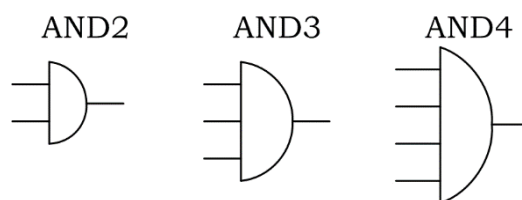
Схемотехническое обозначение по IEC:



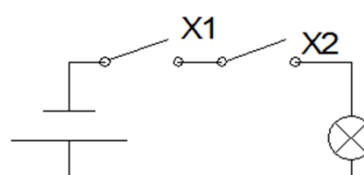
Схемотехническое обозначение по IEEE/ANSI:



Схемотехническое обозначение по DIN:



Реализация на ключевых элементах (тумблерах, кнопках, реле и т.д.):



3. Операция отрицания

Альтернативные наименования:

- логическое «НЕ»;
- инверсия;
- «НЕ»;
- «NOT».

Количество аргументов:

1.

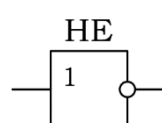
Формы записи:

$$Y = \bar{A} \quad Y = \neg A \quad y = \bar{x}_0$$

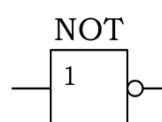
Таблица истинности:

Вход	Выход
X	Y
0	1
1	0

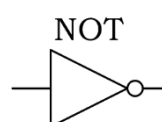
Схемотехническое обозначение по ГОСТ Р:



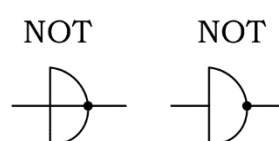
Схемотехническое обозначение по IEC:



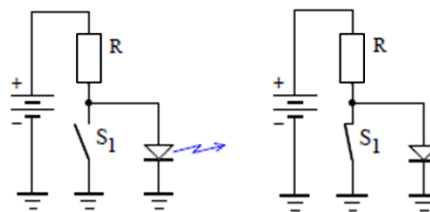
Схемотехническое обозначение по IEEE/ANSI:



Схемотехническое обозначение по DIN:



Реализация на ключевых элементах (тумблерах, кнопках, реле, транзисторных ключах и т.д.):



4. Операция «ИЛИ-НЕ»

Альтернативные наименования:

- стрелка Пирса;
- «NOR».

Количество аргументов:

2 и более.

Формы записи:

$$Y = \overline{A + B}$$

$$Y = \overline{A + B + C}$$

$$Y = A \downarrow B$$

$$y = \overline{x_0 + x_1}$$

$$y = x_0 \downarrow z_1$$

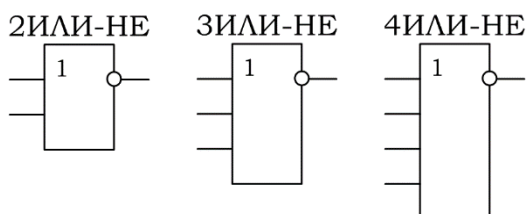
Таблица истинности для двухвходового элемента:

Входы		Выход
X ₁	X ₀	Y
0	0	1
0	1	0
1	0	0
1	1	0

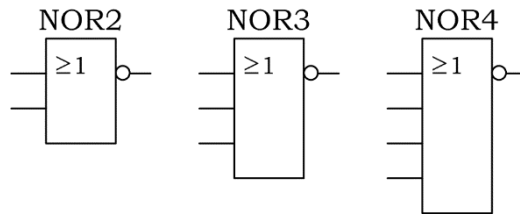
Таблица истинности для трехвходового элемента:

Входы			Выход
X ₂	X ₁	X ₀	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

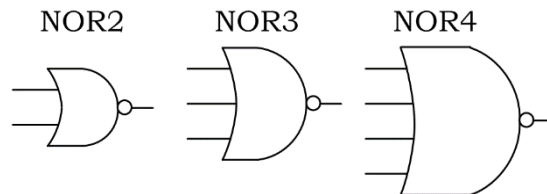
Схемотехническое обозначение по ГОСТ Р:



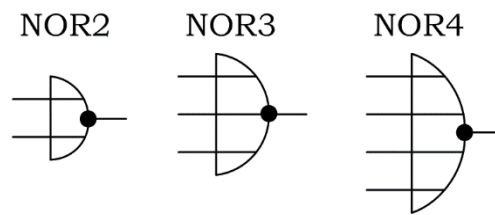
Схемотехническое обозначение по IEC:



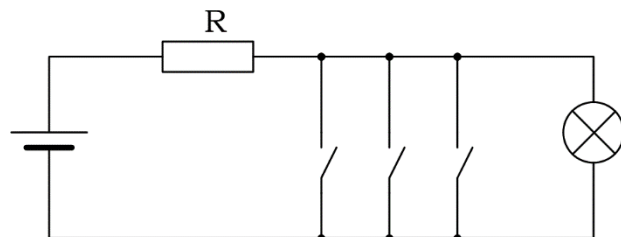
Схемотехническое обозначение по IEEE/ANSI:



Схемотехническое обозначение по DIN:



Реализация на ключевых элементах (тумблерах, кнопках, транзисторных ключах, реле и т.д.):



5. Операция «И-НЕ»

Альтернативные наименования: – штрих Шеффера;
– «NAND».

Количество аргументов: 2 и более.

Формы записи:

$$Y = \overline{A \cdot B}$$

$$Y = \overline{A \cdot B \cdot C}$$

$$Y = A \uparrow B$$

$$y = \overline{x_0 \cdot x_1}$$

$$y = x_0 \uparrow z_1$$

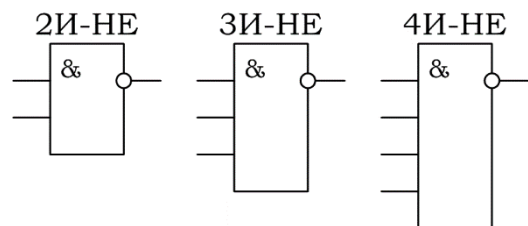
Таблица истинности для двухвходового элемента:

Входы		Выход
X_1	X_0	Y
0	0	1
0	1	1
1	0	1
1	1	0

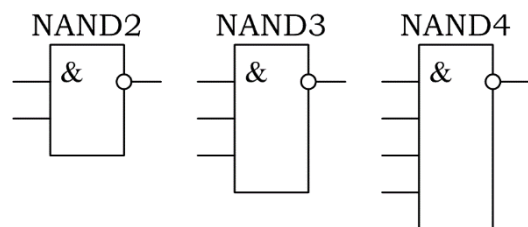
Таблица истинности для
трехвходового элемента:

Входы			Выход
X_2	X_1	X_0	Y
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

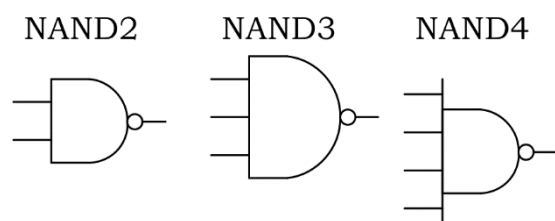
Схемотехническое обозначение
по ГОСТ Р:



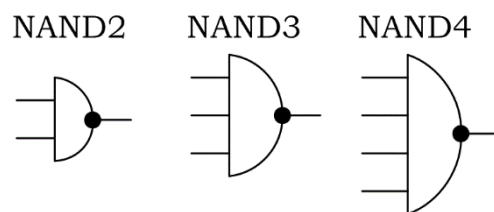
Схемотехническое обозначение
по IEC:



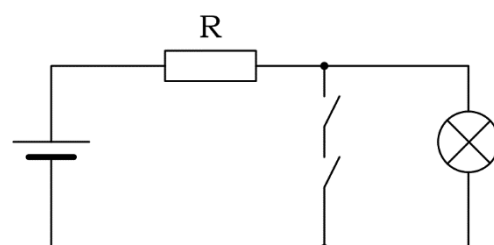
Схемотехническое обозначение
по IEEE/ANSI:



Схемотехническое обозначение
по DIN:



Реализация на ключевых эле-
ментах (тумблерах, кнопках,
транзисторных ключах, реле и
т.д.):



6. Операция «Исключающее ИЛИ»

Альтернативные наименования: «XOR»

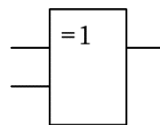
Количество аргументов: 2

Формы записи: $Y = A \oplus B$ $y = x_0 \oplus x_1$

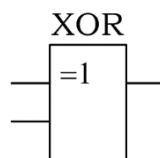
Таблица истинности:

Входы		Выход
X_1	X_0	Y
0	0	0
0	1	1
1	0	1
1	1	0

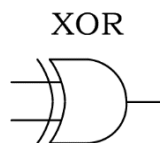
Схемотехническое обозначение по ГОСТ Р:



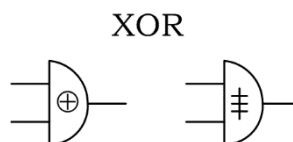
Схемотехническое обозначение по IEC:



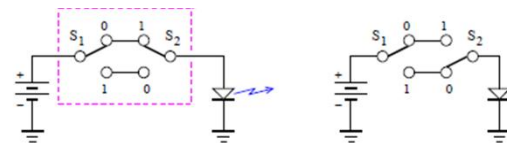
Схемотехническое обозначение по IEEE/ANSI:



Схемотехническое обозначение по DIN:



Реализация на ключевых элементах (тумблерах, кнопках, транзисторных ключах, реле и т.д.):



7. Операция «Исключающее ИЛИ + НЕ»

Альтернативные наименования: «XNOR»

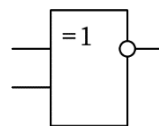
Количество аргументов: 2

Формы записи: $Y = A \odot B$ $y = x_0 \odot x_1$

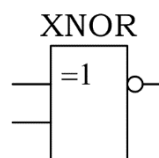
Таблица истинности:

Входы		Выход
X_1	X_0	Y
0	0	1
0	1	0
1	0	0
1	1	1

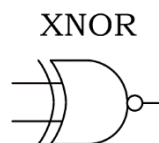
Схемотехническое обозначение по ГОСТ Р:



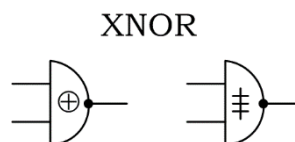
Схемотехническое обозначение по IEC:



Схемотехническое обозначение по IEEE/ANSI:



Схемотехническое обозначение по DIN:



Задание на лабораторную работу и порядок ее выполнения

Общие условия и требования

1. Перед началом выполнения данной и последующих работ ознакомиться с порядком работы с лабораторным комплектом «ALT-DIG-002» и порядком создания схмотехнического проекта в среде проектирования ALTERAQUARTUS (см. приложение).

2. Целью лабораторной работы является:

а) исследование функционирования логических элементов, реализующих базовые логические функции, заполнение таблиц истинности в соответствии с полученными экспериментальными результатами;

б) исследование функционирования логической схемы с несколькими входами и несколькими выходами, формирование таблицы истинности данной логической схемы.

Задание 1 (порядок выполнения). Исследование функционирования логических элементов.

1. В среде проектирования ALTERAQUARTUS подготовить проект с именем lab1, содержащий заготовку схемы с 3 логическими входами и 3 логическими выходами. Задать следующие имена входов-выходов (рис. 1.1):

для входов: x0, x1, x2;

для выходов: x0_led, x1_led, x2_led, y.

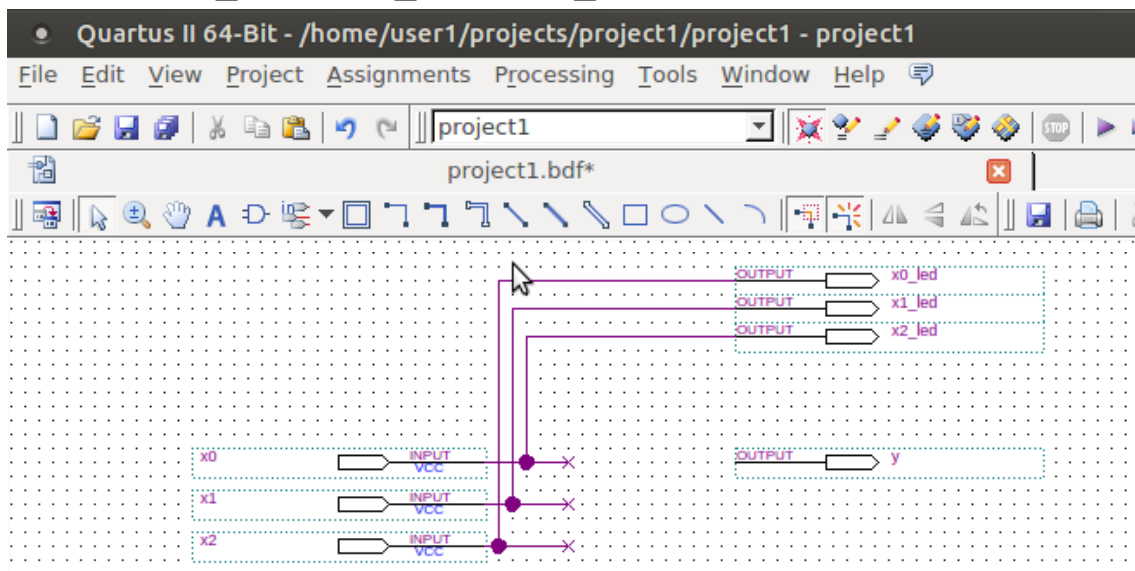


Рис. 1.1. Шаблон схемы для исследования функционирования логических элементов, подготовленный в схмотехническом редакторе среды проектирования ALTERAQUARTUS

Входные цепи «x0», «x1», «x2» будут использованы для подачи логических сигналов на трехвходовый (либо двухвходовый) исследуемый логический элемент. Выходные цепи «x0_led», «x1_led», «x2_led» будут использоваться для визуального контроля состояния входных сигналов. Выходная цепь «y» будет использоваться для визуального контроля состояния выхода исследуемого логического элемента.

2. Добавить исследуемый логический элемент на схему, подключить к нему входные сигналы и выходной сигнал. Пример итоговой схемы с установленным логическим элементом приведен на рис. 1.2.

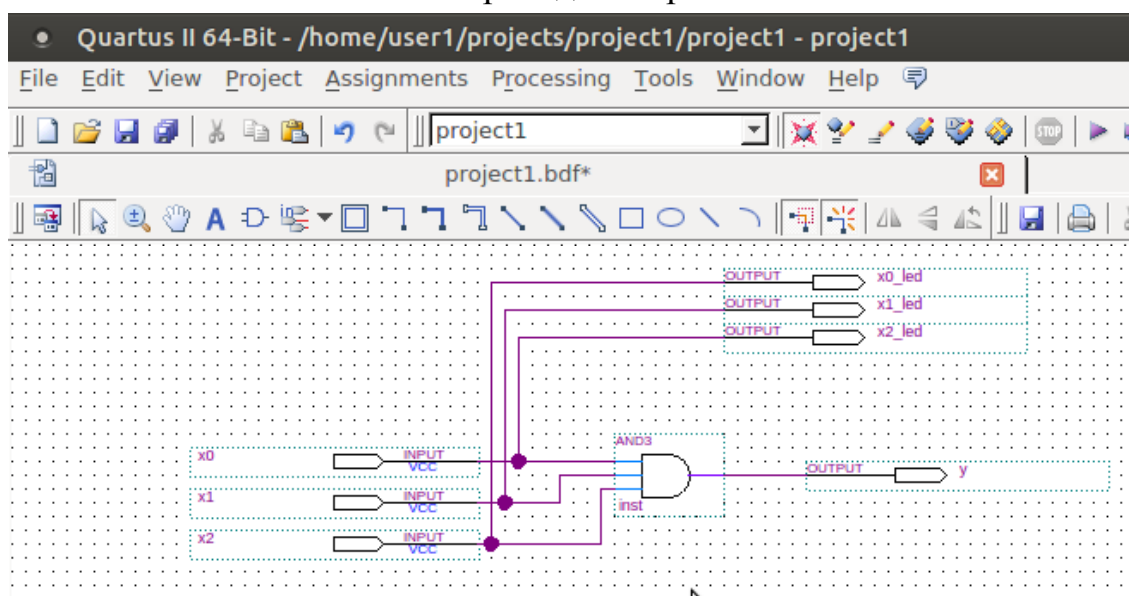


Рис. 1.2. Схема с исследуемым логическим элементом «3И»

3. Выполнить этап назначения логических входов-выходов схемы к физическим выводам микросхемы программируемой логики (ПЛИС) в соответствии с разделом 3 приложения и таблицами соответствия выводов ПЛИС и устройств ввода и индикации (таблицы на рис.1.4). Рекомендуется следующее соответствие логических входов-выходов (рис. 1.5): для входных логических сигналов «x0», «x1», «x2» – двухпозиционные тумблеры «SW1»–«SW8», для выходных логических сигналов «x0_led», «x1_led», «x2_led», «y» – единичные светодиоды «VD2»–«VD10».

4. Подготовить необходимое количество таблиц истинности для описания функционирования двухвходового и трехвходового логического элемента в соответствии с шаблонами (табл. 1.1, 1.2).

5. Выполнить компиляцию проекта, выполнить «программирование» микросхемы ПЛИС (раздел 3 приложения).

6. Задавая необходимые логические сигналы путем переключения тумблеров «SW1»–«SW3» (или иных, на которые назначены логические входы «x0», «x1», «x2») визуально (по назначенным светодиодам) контро-

лизовать состояние входных логических сигналов и выходного сигнала логического элемента. С учетом положения тумблера и формируемого им входного сигнала («лог. 0»/«лог. 1») заполнить таблицу истинности исследуемого элемента.

7. Заменить тип элемента на очередной исследуемый, путем удаления предыдущего из схемы и выбора нового (рис. 1.3). Выбор определенного типа осуществляется либо вводом наименования логического элемента в поле «Name», либо выбором из списка «Libraries».

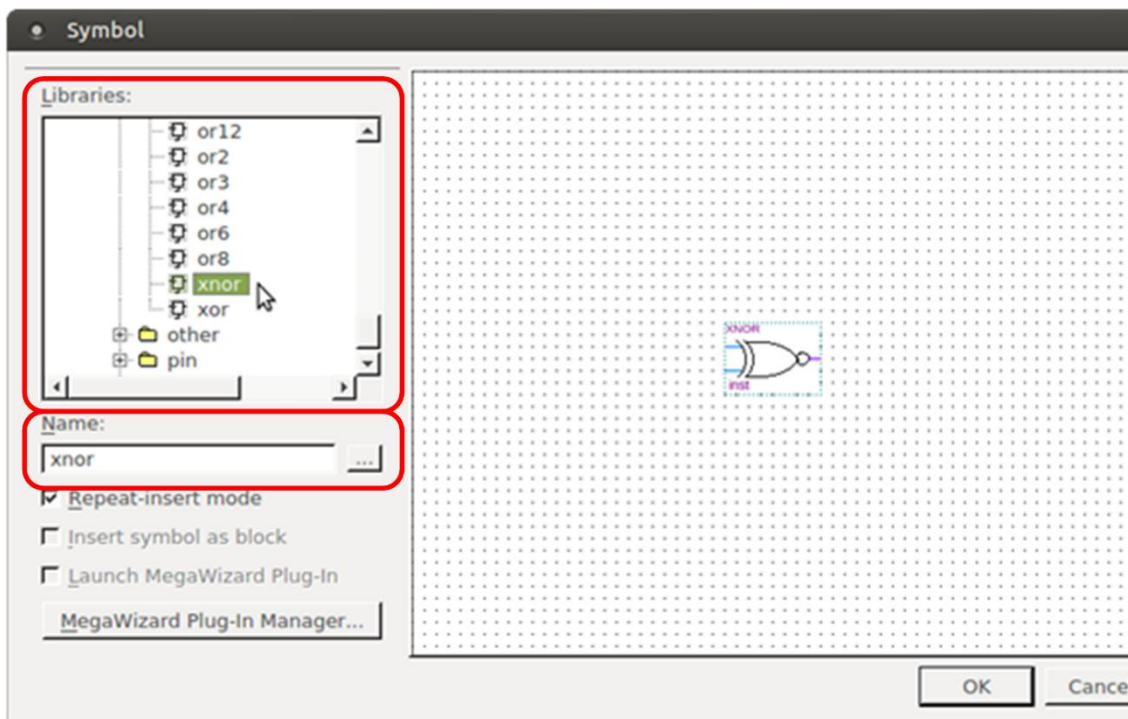


Рис. 1.3. Окно выбора логического элемента

8. Провести исследование поведения следующих логических элементов:

- «AND2» («2И», двухвходовый элемент «И»);
- «OR2» («2ИЛИ», двухвходовый элемент «ИЛИ»);
- «AND3» («3И», трехвходовый элемент «И»);
- «OR3» («3ИЛИ», трехвходовый элемент «ИЛИ»);
- «NAND2» («2И-НЕ», двухвходовый элемент «2И-НЕ»);
- «NAND3» («3И-НЕ», трехвходовый элемент «3И-НЕ»);
- «NOR2» («2ИЛИ-НЕ», двухвходовый элемент «2ИЛИ-НЕ»);
- «NOR3» («3ИЛИ-НЕ», трехвходовый элемент «3ИЛИ-НЕ»);
- «XOR» (элемент «ИСКЛЮЧАЮЩЕ ИЛИ»);
- «XNOR» (элемент «ИСКЛЮЧАЮЩЕ ИЛИ-НЕ»).

Таблица 1.1

Таблица истинности для исследования
двухвходового логического элемента

Логический элемент: «AND2» («2И»)		
Входные сигналы		Выходной сигнал
«x1»	«x0»	«y»
0	0	
0	1	
1	0	
1	1	

Таблица 1.2

Таблица истинности для исследования
трехвходового логического элемента

Логический элемент: «OR3» («3ИЛИ»)			
Входные сигналы			Выходной сигнал
«x2»	«x1»	«x0»	«y»
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

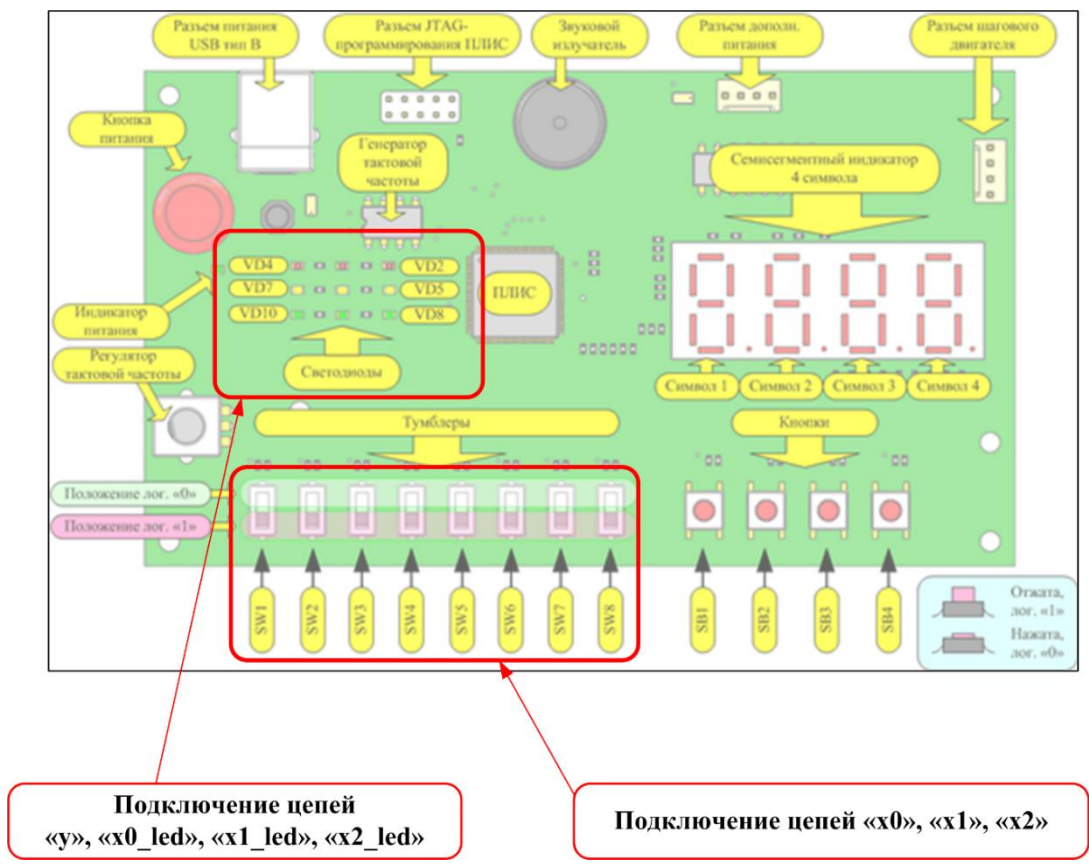


Рис. 1.4. Рекомендуемые группы для подключения логических входов

Таблицы соответствия органов управления и индикации и выводов ПЛИС ALTERA EPM7064

Тип	Напр.	Обознач.	Вывод ПЛИС
Тумблер	вход	SW1	19
		SW2	20
		SW3	21
		SW4	23
		SW5	25
		SW6	29
		SW7	30
		SW8	31
Кнопка	вход	SB1	32
		SB2	33
		SB3	35
		SB4	36

Тип	Напр.	Обознач.	Вывод ПЛИС
Единичный светодиод	выход	VD2	6
		VD3	8
		VD4	9
		VD5	10
		VD6	12
		VD7	13
		VD8	14
		VD9	16
		VD10	17
Звукоизлуч.		B1	100

Рис. 1.5. Соответствие органов управления и индикации модуля «ALT-DIG-002» и физических выводов микросхемы ПЛИС

Задание 2 (порядок выполнения). Исследование функционирования логической схемы.

1. В среде проектирования ALTERAQUARTUSподготовить проект с именем lab1a, содержащий схемы с 3 логическими входами и 6 логическими выходами, логическими элементами и связями между ними в соответствии с рис. 1.6. Задать следующие имена входов-выходов:

для входов: x0, x1, x2;

для выходов: x0_led, x1_led, x2_led, y, y1, y2.

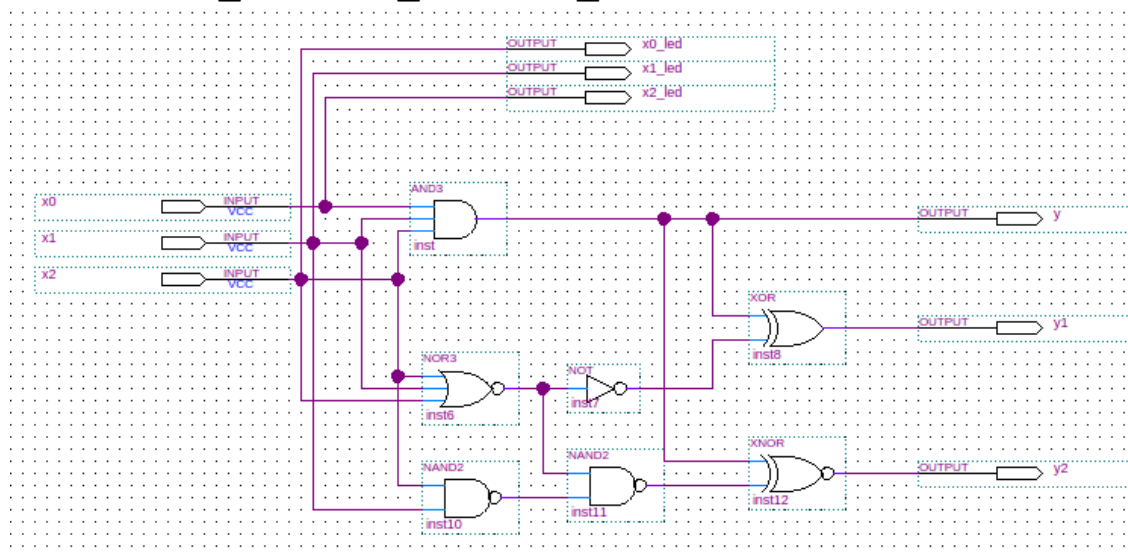


Рис. 1.6. Исследуемая схемы, подготовленная в схемотехническом редакторе среды проектирования ALTERAQUARTUS

Входные цепи «x0», «x1», «x2»будут использованы для подачи логических сигналов. Выходные цепи «x0_led», «x1_led», «x2_led» будут использоваться для визуального контроля состояния входных сигналов. Выходные цепи «y»,«y1»,«y2»будут использоваться для визуального контроля состояния выхода логической схемы.

2. Выполнить этап назначения логических входов-выходов схемы к физическим выводам микросхемы программируемой логики (ПЛИС) в соответствии с разделом 3 приложения и таблицами соответствия выводов ПЛИС и устройств ввода и индикации (табл. на рис.1.4). Рекомендуется следующее соответствие логических входов-выходов (рис. 1.5): для входных логических сигналов «x0», «x1», «x2» – двухпозиционные тумблеры «SW1»–«SW8», для выходных логических сигналов «x0_led», «x1_led», «x2_led», «y», «y1»,«y2» – единичные светодиоды «VD2»–«VD10».

3. Подготовить совмещенную таблицу истинности для описания функционирования логической схемы в соответствии с шаблоном (табл. 1.3).

4. Выполнить компиляцию проекта, выполнить «программирование» микросхемы ПЛИС (раздел 3 приложения).

5. Задавая необходимые логические сигналы путем переключения тумблеров «SW1»–«SW3» (или иных, на которые назначены логические входы «x0», «x1», «x2») визуально (по назначенным светодиодам) контролировать состояние входных логических сигналов и выходного сигнала логического элемента. С учетом положения тумблера и формируемого им входного сигнала («лог. 0»/«лог. 1») заполнить таблицу истинности логической схемы.

Таблица 1.3

Таблица истинности для исследования логической схемы

Входные сигналы			Выходные сигналы		
«x2»	«x1»	«x0»	«y»	«y1»	«y2»
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Содержание отчета

1. Копии экрана среды разработки ALTERAQUARTUSc подготовленными схемами (для каждого логического элемента и логической схемы).
2. Фото состояния органов управления и индикации для каждого состояния устройства.
3. Заполненные таблицы истинности (для каждого логического элемента и логической схемы).
4. Выводы.

Контрольные вопросы

1. Перечислите способы задания логических функций. Каковы достоинства и недостатки каждого из них?
2. Возможна ли реализация логического устройства с одним входным сигналом и одним выходным сигналом? В случае положительного ответа, какие логические функции оно может реализовывать?
3. Каково количество строк таблицы истинности, если число входных сигналов (булевых переменных) равно 6?

4. Запишите функцию и приведите таблицу истинности для следующих логических элементов: а) ИЛИ; б) И; в) НЕ; г) ИЛИ-НЕ; д) И-НЕ; е) исключающее ИЛИ.

5. Приведите простейшую электрическую схему логического элемента: а) «ИЛИ»; б) «И»; в) «НЕ». Поясните принцип работы.

6. Как поступают на практике с неиспользуемыми входами микросхем ТТЛ / КМОП?

7. Приведите УГО логического элемента с открытым коллектором. Укажите области применения таких элементов.

8. Как составить таблицу истинности для трех аргументов x_1, x_2, x_3 , если функция принимает значение 1, тогда когда x_2 принимает значение 1. Напишите булево выражение для функции, принимающей значение 1.

ЛАБОРАТОРНАЯ РАБОТА № 2

СИНТЕЗ КОМБИНАЦИОННЫХ СХЕМ НА БАЗОВЫХ ЛОГИЧЕСКИХ ЭЛЕМЕНТАХ. ОПТИМИЗАЦИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ

Цель работы: построение логической схемы по заданному табличному, словесному, либо алгебраическому описанию логической функции (ЛФ). Синтез алгебраической записи ЛФ по заданной таблице истинности в совершенной дизъюнктивной и совершенной конъюнктивной формах. Выполнение оптимизации ЛФ с помощью тождеств алгебры логики и диаграмм Карно. Синтез логической схемы по оптимизированному алгебраическому выражению на базовых логических элементах. Выполнение исследования корректности требуемого функционирования логической схемы.

Оборудование: лабораторный комплекс «ALT-DIG-002».

Теоретический материал

Логическое (цифровое) устройство оперирует булевыми входными сигналами (переменными), преобразуя их в выходные булевы сигналы (переменные) с помощью базовых логических функций логического умножения «И», логического сложения «ИЛИ», отрицания (инверсии) «НЕ». Характерной особенностью комбинационных логических устройств является отсутствие в них «памяти», т.е. хранения и использования предыстории изменения входных сигналов (входных переменных). Выходные сигналы комбинационного логического устройства зависят только от состояния

входных сигналов в данный конкретный момент времени. Общий вид логического устройства данного типа представлен на рис. 2.1.

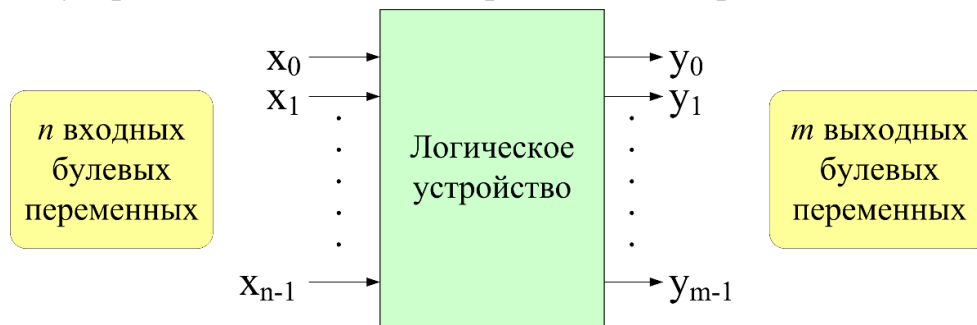


Рис. 2.1. Общий вид комбинационного логического устройства, имеющего пвходных сигналов (переменных) и мвыходных сигналов (переменных)

Общий вид функциональных зависимостей между входными сигналами и выходными выглядит следующим образом:

$$\begin{aligned} y_0 &= f_0(x_0, x_1, \dots, x_{n-1}); \\ y_1 &= f_1(x_0, x_1, \dots, x_{n-1}); \\ &\vdots \\ y_{m-1} &= f_{m-1}(x_0, x_1, \dots, x_{n-1}); \end{aligned} \quad (2.1)$$

Вариантами состава входов-выходов логического устройства (ЛУ) являются:

а) ЛУ с пвходными сигналами и единственным выходным (рис. 2.2), зависимость выходного сигнала от входных описывается выражением:

$$y = f(x_0, x_1, \dots, x_{n-1}) \quad (2.2)$$

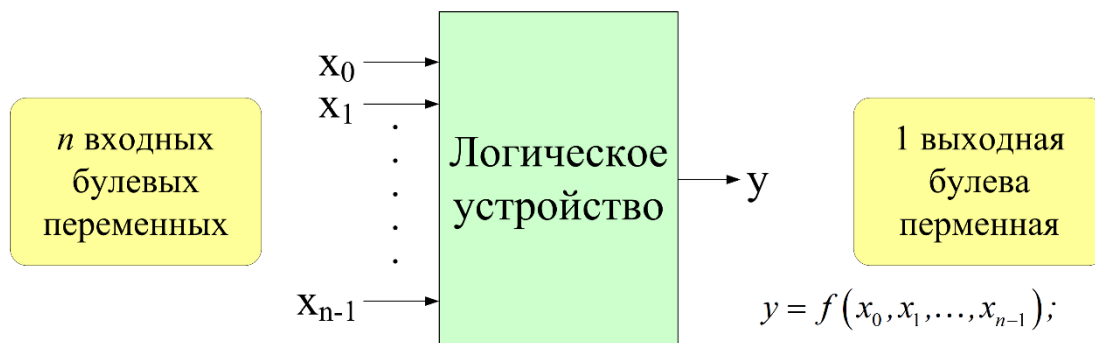


Рис. 2.2. Общий вид комбинационного логического устройства, имеющего пвходных сигналов (переменных) и единственнымвыходным сигналом

б) ЛУ с единственным входным логическим сигналом и m выходными сигналами (рис. 2.3), зависимость выходных сигналов от входного описывается выражениями:

$$y_0 = f_0(x); \quad y_1 = f_1(x); \quad \dots; \quad y_{m-1} = f_{m-1}(x); \quad (2.3)$$

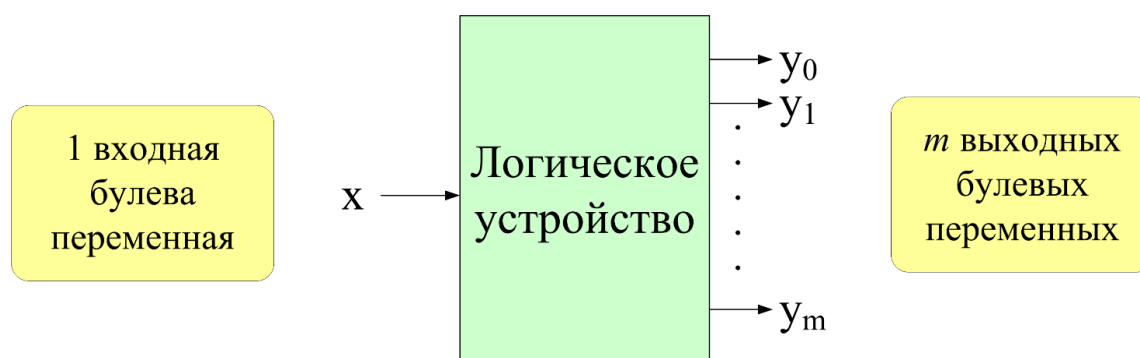


Рис. 2.3. Общий вид комбинационного логического устройства, имеющего единственный входной сигналов (переменную) и m выходных сигналов

В выражениях (1)–(3) символами $x_0, \dots, x_{n-1}$ обозначены входные сигналы (переменные), $y_0, \dots, y_{m-1}$ – выходные сигналы, $f, f_0, \dots, f_{m-1}$ – функциональные зависимости, построенные с помощью операций алгебры логики.

Задачей синтеза логического устройства является формирование (построение) логических функций $f, f_0, \dots, f_{m-1}$ таким образом, чтобы логическое устройство выполняло заданные режимы работы, функции. При этом также стоит задача оптимизации логических выражений и/или схемотехнической части логического устройства.

Рассмотрим вариант синтеза ЛУ на примере управления индикацией семисегментного индикатора в части отображения десятичных цифр «0» и «1» в зависимости от состояния единственного дискретного входа. Семисегментный индикатор представляет собой 7 единичных светодиодов (в ряде случаев – 8, если используется десятичная точка), конструктивно организованных таким образом, чтобы при определенных сочетаниях засвеченных и погашенных сегментов формировалось однозначно идентифицируемое человеком изображение символа либо цифры. Соответствие расположения светодиода (сегмента) в конструкции индикатора стандартно (рис. 2.4а), при этом возможны следующие внутренние электрические схемы соединения: независимые светодиоды (б), схема соединения с общим катодом (ОК) (рис. 2.4в), схема соединения с общим анодом (ОА) (рис. 2.4г). Наиболее популярными в современных индикаторах являются схемы включения с ОА и ОК.

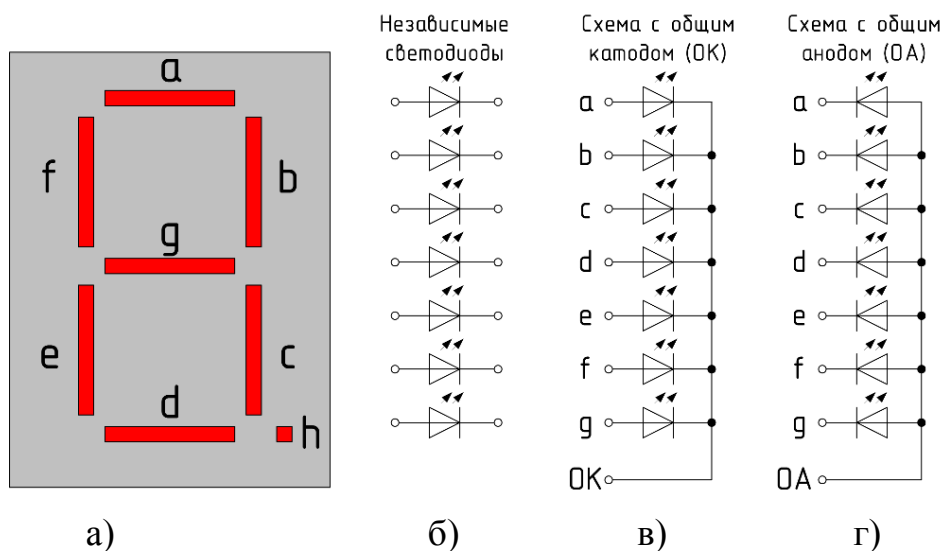


Рис. 2.4. Схема расположения сегментов семисегментного индикатора (а) и варианты организации соединения его единичных светодиодов (б, в, г)

Для организации свечения светодиода применяется его прямое включение в схеме постоянного тока (рис. 2.5). Образ символа формируется путем индивидуальным управлением свечением отдельных светодиодов (на рис. 2.6 приведено отображение символа «С» путем включения сегментов «а», «d», «е», «f» при включении индикатора по схеме с общим катодом).

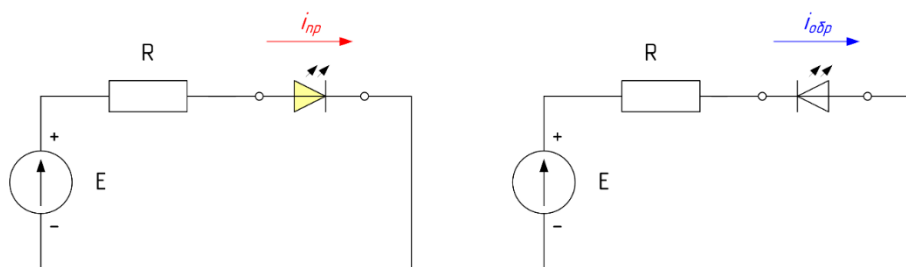


Рис. 2.5. Прямое и обратное включение светодиода

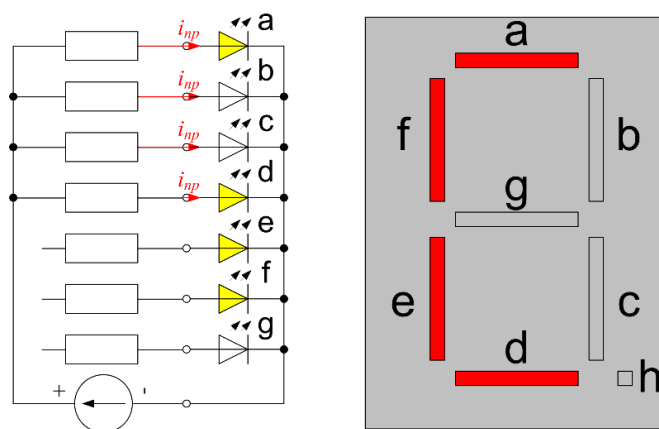


Рис. 2.6. Формирование образа символа путем включения отдельных сегментов (на примере схемы включения с общим катодом)

С учетом того, что выходными сигналами логических элементов являются уровни «лог. 1» и «лог. 0», соответствующими коммутации выходным каскадом логического элемента цепей питания ЛЭ на его выход (рис. 2.7), то задачей синтеза является корректное формирование состояния выходных ключей (или, что тоже самое состояния выхода: лог. «0» либо лог. «1») таким образом, чтобы, с учетом схемы включения индикатора, отображался необходимый символ.

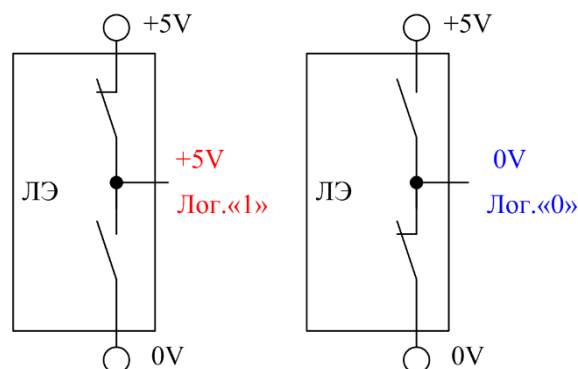


Рис. 2.7. Эквивалентная схема формирования уровней напряжения на выходе логического элемента (коммутация цепей питания)

На рис. 2.8 приведено соответствие входного булевого сигнала X_i внешний вид отображаемых цифр на семисегментном индикаторе.

Как известно из теоретической части курса «Цифровая схемотехника», логическая функция может быть задана одним из трех способов: словесным описанием, таблицей истинности, алгебраическим выражением. Каждый из них можно преобразовать в любой другой выполнением анализа входных данных и синтеза словесной или алгебраической структуры, заполнением таблицы.

В данном случае словесным описанием может быть следующий текст:

«Логическое устройство управляет состоянием семи светодиодов (сегментов) индикатора, имеет 7 булевых выходных сигналов y_a – y_g , один входной булев сигнал X . При логическом нуле на входе X все сегменты, кроме сегмента g должны быть засвечены. При логической единице на входе X должны быть засвечены только сегменты b и c . Схема включения семисегментного индикатора – общий катод».

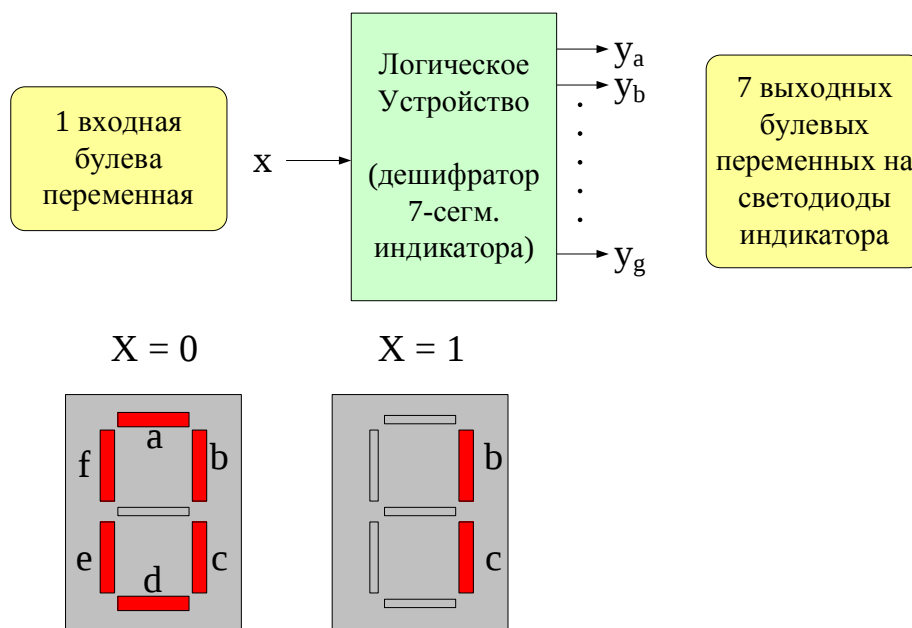


Рис. 2.8. Соответствие значения входного сигнала и состояния сегментов семисегментного индикатора

Учитывая наглядность работы логического устройства (рис. 2.8), наиболее простым вариантом описания логических функций является, тем не менее, табличный (рис. 2.9).

Вход	Выходы						
x	y_a	y_b	y_c	y_d	y_e	y_f	y_g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0

7 выходных логических функций

Рис. 2.9. Табличное описание (совмещенная для нескольких функций) требуемых в соответствии с заданием логических функций

Анализ таблицы показывает, что имеется всего два возможных состояния входной логической переменной (соответственно, 2 строки в таблице истинности, без учета заголовка), при этом значения логических функций « y_b », « y_c », « y_g » являются константами, не зависят от состояния входного сигнала. Это следует лишь из образа формируемых цифр «0» и «1» на индикаторе (рис. 2.10), т.к. в каждом из двух вариантов отображения, сегменты «b» и «c» подсвечены, сегмент «g» не должен светиться.

Вход	Выходы						
x	y_a	y_b	y_c	y_d	y_e	y_f	y_g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0

Рис. 2.10. Константные значения логических функций « y_b », « y_c », « y_g »

Анализируя таблицу в части остальных функций можно отметить, что для сегментов «а», «d», «е», «f» значение соответствующего выходного сигнала ЛУ должно быть противоположным по отношению к входному сигналу, т.е. применяется логическая функция инверсии (функция, логический элемент «НЕ»). Таким образом, алгебраические выражения для искомых функций будут представлять вид:

$$\begin{aligned}
 y_b &= 1; \\
 y_c &= 1; \\
 y_g &= 0; \\
 y_a &= y_d = y_e = y_f = \bar{x};
 \end{aligned}
 \tag{2.4}$$

Учитывая простейшую форму алгебраических выражений искомых функций оптимизацию выражений выполнять не имеет смысла.

По полученным функциям синтезируется логическая схема. С учетом возможности «распараллеливания» выхода логического элемента на несколько выходных каскадов, нагрузок (в данном случае – на несколько светодиодов), логическая схема устройства может выглядеть следующим образом (рис. 2.11).

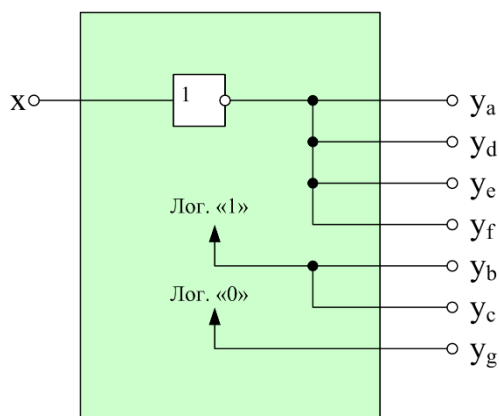


Рис. 2.11. Логическая схема устройства управления индикацией

Следует отметить, что при фактическом подключении исполнительных устройств, устройств индикации и т.д. необходимо выполнить проверку фактических токов, требуемых для корректной работы нагрузки с длительным допустимым выходным током логического элемента.

Синтезированная в среде проектирования ALTEAQUARTUS схема приведена на рис. 2.12. Следует отметить, что в данной среде проектирования источниками логической единицы (лог. «1») и логического нуля (лог. «0») при схемотехническом вводе являются элементы с именами «vcc» (лог. «1», рис. 2.13а) и «gnd» (лог. «0», рис. 2.13б). Функционирование устройства приведено на рис. 2.14.

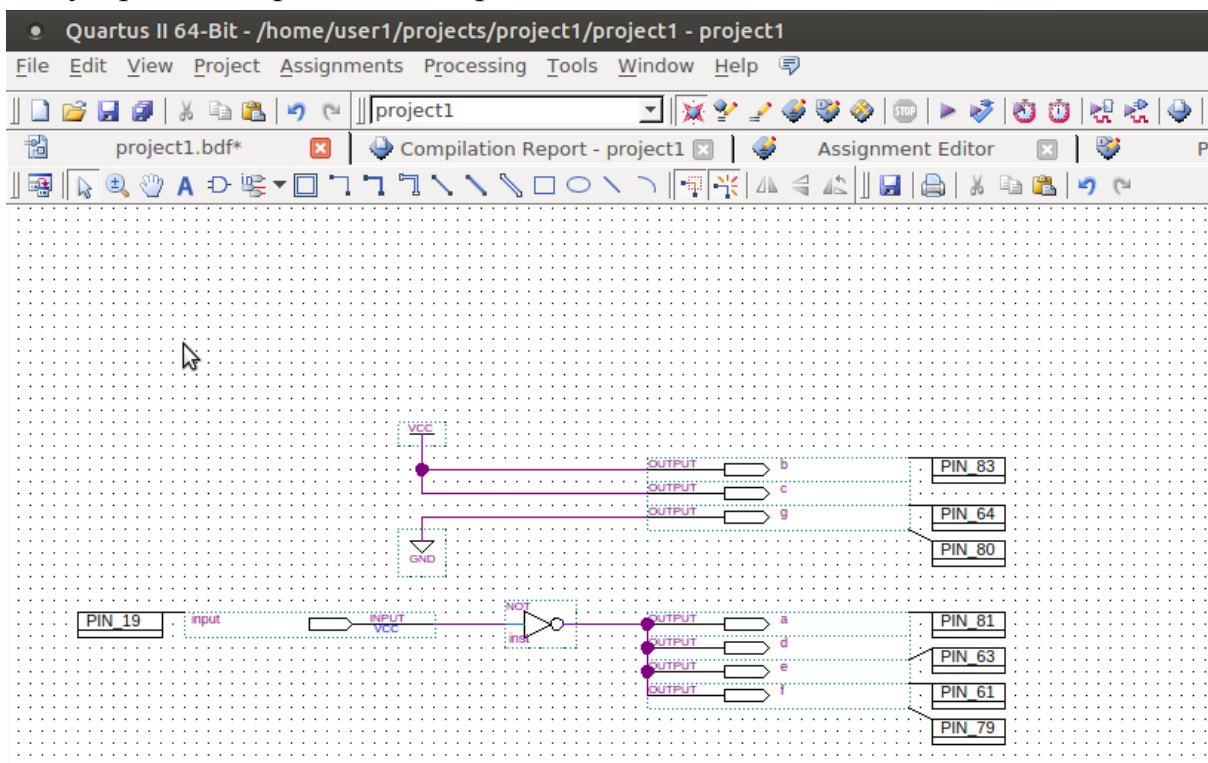


Рис. 2.12. Логическая схема устройства управления индикацией в среде проектирования ALTEAQUARTUS



Рис. 2.13. Элементы-источники сигналов «лог. 1» и «лог. 0» в среде проектирования ALTEAQUARTUS

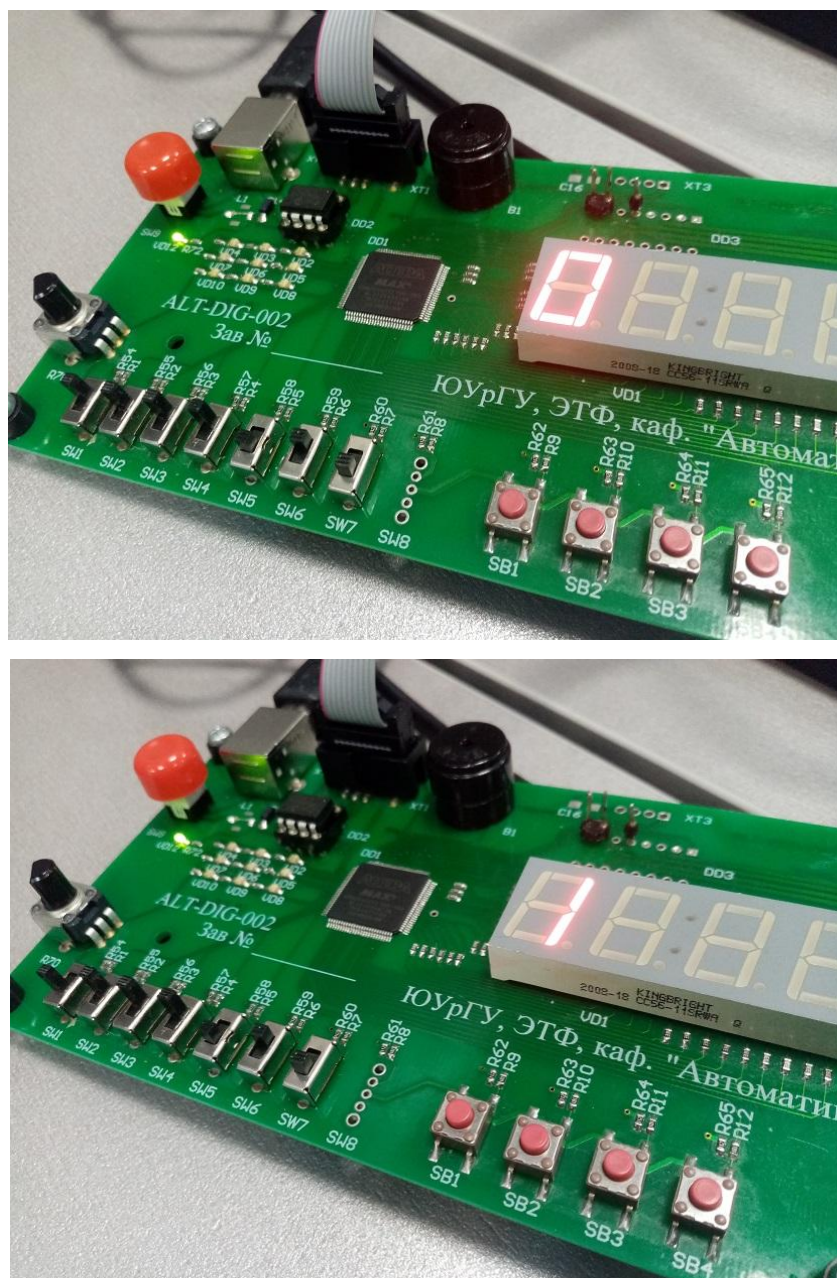


Рис. 2.14. Пример функционирования устройства

Рассмотрим вариант синтеза ЛУ на примере управления индикацией семисегментного индикатора в части отображения десятичных цифр от «0» до «7» в зависимости от подаваемого на вход двоичного кода, соответствующего десятичному числу.

С учетом того, что задана необходимость индикации 8 вариантов десятичных цифр, соответствующий эквивалентный десятичному числу двоичный код будет представлять собой трехразрядное двоичное число.

$$n = \lceil \log_2 N \rceil, \quad (2.5)$$

где N – количество заданных состояний (двоичных комбинаций и т.д.), p – требуемое для формирования данного количества число двоичных разрядов (рис. 2.15).

Десятичное число	Двоичное число			
0	000	X_2	X_1	X_0
1	001	0	0	0
2	010	0	0	1
3	011	0	1	0
4	100	0	1	1
5	101	1	0	0
6	110	1	0	1
7	111	1	1	0
		1	1	1

Старший разряд
 Младший разряд

Рис. 2.15. Десятичное число, двоичный эквивалент и соответствие номеров разрядов и имен булевых входных переменных

Таким образом, логическое устройство будет иметь 3 входных булевых сигнала и 7 выходных булевых сигналов, подключаемых к соответствующим сегментам индикатора.

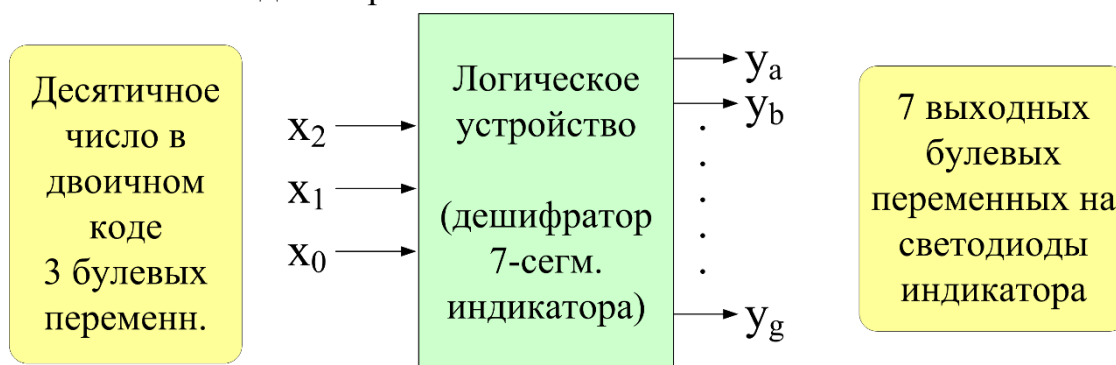


Рис. 2.16. Входные и выходные сигналы (переменные) логического устройства управления индикацией

Образы десятичных цифр «0»–«7» можно представить следующим образом (рис. 2.17):

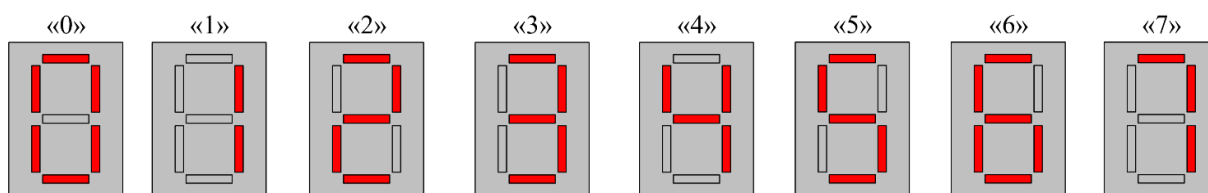


Рис. 2.17. Образцы десятичных цифр для индикации

Аналогично предыдущему примеру логические функции можно представить в табличном виде, получить алгебраические выражения и затем выполнить оптимизацию. Заполнение совмещенной таблицы истинности выполним для схемы индикатора «с общим катодом» с учетом рис. 2.17. Заполненная таблица выглядит следующим образом (рис. 2.18).

Дес. число	Входы			Выходы						
	x_2	x_1	x_0	y_a	y_b	y_c	y_d	y_e	y_f	y_g
0	0	0	0	1	1	1	1	1	1	0
1	0	0	1	0	1	1	0	0	0	0
2	0	1	0	1	1	0	1	1	0	1
3	0	1	1	1	1	1	1	0	0	1
4	1	0	0	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1	0	1	1
6	1	1	0	1	0	1	1	1	1	1
7	1	1	1	1	1	1	0	0	0	0

7 выходных логических функций

Рис. 2.18. Заполненная таблица истинности для логического устройства управления индикацией (отображение цифр «0»–«7»)

Совмещенные таблицы истинности можно разбить на индивидуальные для каждой выходной функции и рассматривать независимо от остальных логических функций (что и так должно выполняться с учетом, что для комбинационных устройств выходные сигналы зависят только от состояния входов в данный момент времени). Индивидуальные таблицы истинности приведены на рис. 2.19.

Сегмент "а"				Сегмент "b"				Сегмент "с"				Сегмент "d"				Сегмент "е"			
Входы			Вых	Входы			Вых	Входы			Вых	Входы			Вых	Входы			Вых
x_2	x_1	x_0	y_a	x_2	x_1	x_0	y_b	x_2	x_1	x_0	y_c	x_2	x_1	x_0	y_d	x_2	x_1	x_0	y_e
0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	0
0	1	0	1	0	1	0	1	0	1	0	0	0	1	0	1	0	1	0	1
0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	0
1	0	0	0	1	0	0	1	1	0	0	1	1	0	0	0	1	0	0	0
1	0	1	1	1	0	1	0	1	0	1	1	1	0	1	1	1	0	1	0
1	1	0	1	1	1	0	0	1	1	0	1	1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0

Сегмент "Г"				Сегмент "g"			
Входы			Вых	Входы			Вых
x_2	x_1	x_0	y_f	x_2	x_1	x_0	y'_g
0	0	0	1	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	1
0	1	1	0	0	1	1	1
1	0	0	1	1	0	0	1
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	0	1	1	1	0

Рис. 2.19. Индивидуальные таблицы истинности для логических функций устройства индикации (отображение цифр «0»–«7»)

Для дальнейшего построения схемы из таблицы истинности необходимо сформировать алгебраические выражения и, на последующем этапе, их упростить. Данный этап выполняется одним из двух способов:

Вариант А. Построение алгебраического выражения в дизъюнктивной нормальной форме (ДНФ).

Дизъюнктивная нормальная форма – представление алгебраического выражения в виде логической суммы элементарных логических произведений, в каждое из которых входная переменная (сигнал), или ее инверсия входит один раз.

Получить ДНФ можно из таблицы истинности с использованием следующего алгоритма:

- для каждого набора входных переменных, на которых функция равна «лог.1», записываются элементарные логические произведения входных переменных, причём переменные, равные лог.0 записываются с инверсией;
- логически суммируются все произведения.

Таким образом, функцию, заданную таблицей следует записать в следующем виде:

$$y = \overline{x_2} \cdot \overline{x_1} \cdot x_0 + \overline{x_2} \cdot x_1 \cdot \overline{x_0} + x_2 \cdot \overline{x_1} \cdot \overline{x_0} + x_2 \cdot x_1 \cdot x_0 \quad (2.6)$$

Такая форма называется совершенной дизъюнктивной нормальной формой (СДНФ). Она представляется логической суммой простых конъюнкций, каждая из которых содержит все переменные в прямом или инверсном виде. Входящие в СДНФ конъюнкции называются *минтермами* или *конституентами единиц*.

На рис. 2.20 приведено построение алгебраического выражения для логической функции в форме СДНФ, реализующей функционирование сегмента «е».

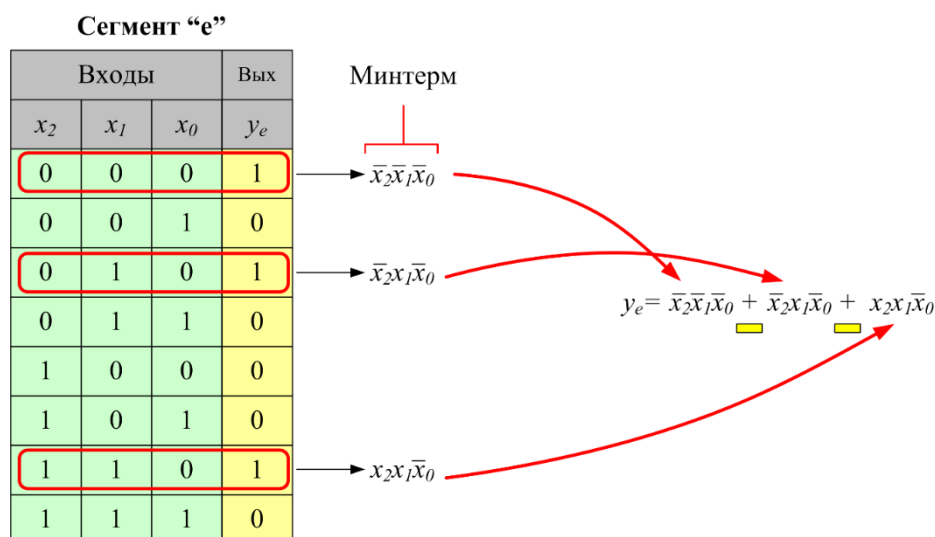


Рис. 2.20. Построение алгебраического выражения в форме СДНФ для логической функции сегмента «е»

В ряде случаев форма СДНФ является слишком избыточной (громоздкое алгебраическое выражение, что, без проведения этапа оптимизации, приводит к логической схеме с большим количеством ЛЭ). Например, логическая функция в форме СДНФ для сегмента «с» будет иметь вид, представленный на рис. 2.21. Оно содержит сумму 7 минтермов.

В связи с этим перед началом синтеза алгебраического выражения проводят оценку доли логических нулей и логических единиц в столбце значений логической функции, и в случае, если количество логических нулей меньше количества логических единиц, применяют второй вариант построения алгебраического выражения.

Вариант Б. Построение алгебраического выражения в конъюнктивной нормальной форме (КНФ).

Конъюнктивная нормальная форма – представление алгебраического выражения в виде логического произведения элементарных логических сумм, в каждое из которых входная переменная (сигнал), или ее инверсия входит один раз.

Получить КНФ можно из таблицы истинности с использованием следующего алгоритма:

- для каждого набора переменных, на которых функция равна «лог.0», записываются элементарные логические суммы входных переменных, причём переменные, равные «лог.1» записываются с инверсией;
- логически перемножают все логические суммы.

Таким образом, функцию, заданную таблицей следует записать в следующем виде:

$$y = (\bar{x}_2 + x_1 + x_0) \cdot (x_2 + \bar{x}_1 + x_0) \cdot (x_2 + x_1 + \bar{x}_0) \cdot (x_2 + x_1 + x_0) \quad (2.7)$$

Такая форма называется совершенной конъюнктивной нормальной формой (СКНФ). Она представляется логическим произведением простых дизъюнкций, каждая из которых содержит *все* переменные в прямом или инверсном виде (рис. 2.22). Входящие в СКНФ дизъюнкции называются *макстермами* или *конституентами нулей*. Используя основные законы алгебры логики одну форму всегда можно преобразовать в другую.

Сегмент «с»

Входы			Вых	
x_2	x_1	x_0	y_c	
0	0	0	1	→ $\bar{x}_2\bar{x}_1\bar{x}_0$
0	0	1	1	→ $\bar{x}_2\bar{x}_1x_0$
0	1	0	0	
0	1	1	1	→ $\bar{x}_2x_1x_0$
1	0	0	1	→ $x_2\bar{x}_1\bar{x}_0$
1	0	1	1	→ $x_2\bar{x}_1x_0$
1	1	0	1	→ $x_2x_1\bar{x}_0$
1	1	1	1	→ $x_2x_1x_0$

$$y_c = \bar{x}_2\bar{x}_1\bar{x}_0 + \bar{x}_2\bar{x}_1x_0 + \bar{x}_2x_1x_0 + x_2\bar{x}_1\bar{x}_0 + x_2\bar{x}_1x_0 + x_2x_1\bar{x}_0 + x_2x_1x_0$$

Рис. 2.21. Построение алгебраического выражения в форме СДНФ для логической функции сегмента «с»

Сегмент «с»

Входы			Вых
x_2	x_1	x_0	y_c
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Макстерм

$x_2 + \bar{x}_1 + x_0$

$y_c = x_2 + \bar{x}_1 + x_0$

Рис. 2.22. Построение алгебраического выражения в форме СКНФ для логической функции сегмента «с»

В ряде случаев применяют компактную форму записи алгебраических функции, используя номера минтермов и макстермов. Данным номера являются десятичными эквивалентами двоичного представления минтерма или макстерма.

Например, записи

$$y_e(x_2, x_1, x_0) = \sum m(0, 2, 6)$$

соответствует полная запись логической функции в следующем виде:

$$y_e(x_2, x_1, x_0) = \underbrace{\bar{x}_2 \bar{x}_1 \bar{x}_0}_0 + \underbrace{\bar{x}_2 x_1 \bar{x}_0}_2 + \underbrace{x_2 x_1 \bar{x}_0}_6.$$

Записи

$$y_c(x_2, x_1, x_0) = \prod M(4, 5, 7)$$

соответствует полная запись логической функции в следующем виде:

$$y_c(x_2, x_1, x_0) = \underbrace{(x_2 + \bar{x}_1 + \bar{x}_0)}_4 \underbrace{(x_2 + \bar{x}_1 + x_0)}_5 \underbrace{(x_2 + x_1 + x_0)}_7$$

За этапом синтеза алгебраического выражения может напрямую следовать этап проектирования схемы, ее подготовки в среде разработки. В ча-

стности, на рис. 2.23 приведена схмотехническая реализация логической функции для сегмента «е».

$$y_e = \bar{x}_2 \bar{x}_1 \bar{x}_0 + \bar{x}_2 x_1 \bar{x}_0 + x_2 x_1 \bar{x}_0$$

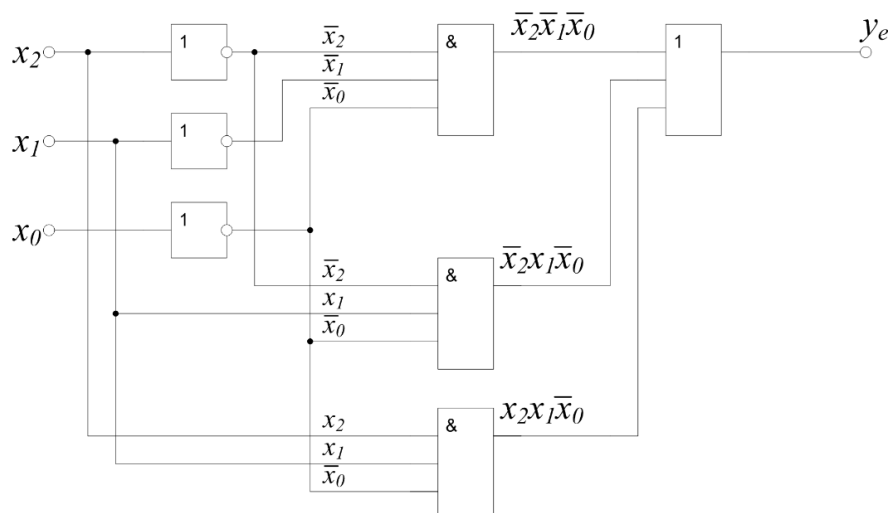


Рис. 2.23. Схмотехническая реализация логической функции сегмента «е» (с указанием промежуточных значений)

В большинстве случаев полученное алгебраическое выражение, особенно в случае сложных логических устройств, может быть оптимизировано с использованием тождеств алгебры логики, либо с использованием специализированного программного обеспечения. Для простых логических функций до 4–5 входных переменных возможно использование графического способа оптимизации – с использованием диаграмм (карт) Вейча/Карно (способ предложен в 1952–1953 годах).

Фактически диаграмма Карно представляет собой альтернативный вариант построения таблицы истинности и представляет собой прямоугольную двумерную таблицу (матрицу), содержащую столько же ячеек, сколько строк содержится в исходной таблице истинности. Столбцы и строки именуются именами переменных или их сочетаниями, в ячейки диаграммы заносятся данные из таблицы истинности. В частности, на рис. 2.24 приведен пример простейшей диаграммы Карно для функции двух переменных. На рис. 2.24а) столбцы именованы значениями переменной x_1 , строки – значениями переменной x_0 . Ячейки заполнены тремя единицами значений функции из таблицы истинности. Стрелками обозначены соответствия значений входных переменных, строк таблицы истинности и ячеек диаграммы Карно. На рис. 2.24б) приведен иной вид: столбцы теперь соответствуют значениям переменной x_0 , строки – переменной x_1 . Несмотря на иной вид итоговой заполненной диаграммы, результат оптимизации исходной функции будет идентичным. На рис. 2.24в) вместо значений пере-

менных применяется их прямое и инверсное обозначения, при этом адресация следующая: если в исходной таблице истинности значение входной переменной равно «лог. 0», то в диаграмме Карно она используется с символом инверсии и наоборот (если «лог. 1» – то без символа инверсии).

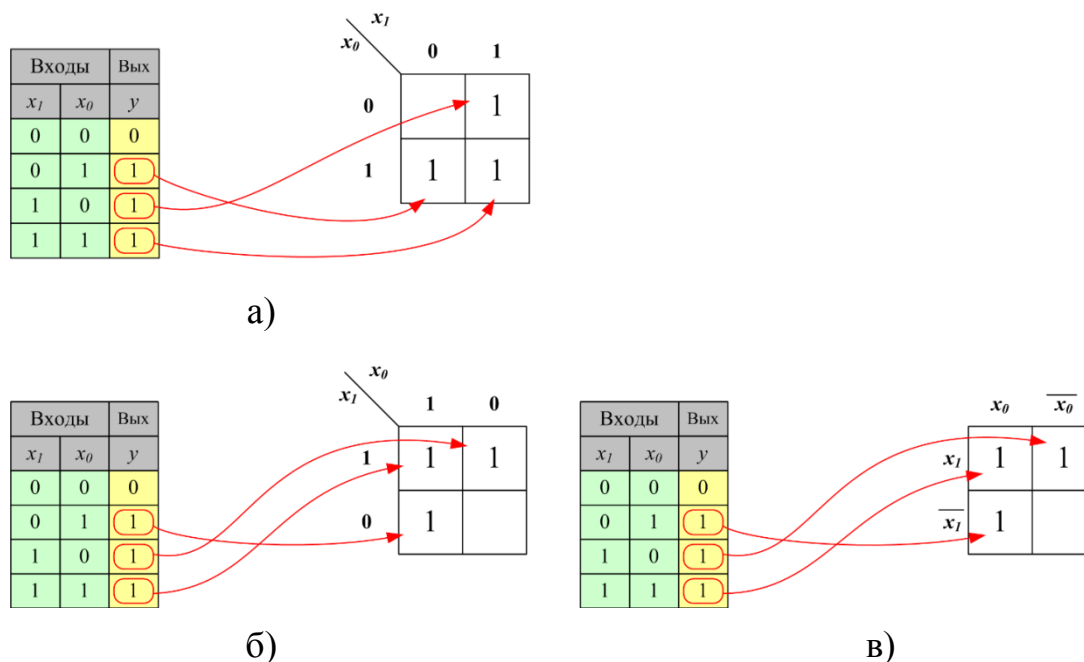


Рис. 2.24. Пример диаграммы Вейча/Карно для функции двух переменных (на примере функции «ИЛИ»)

На примере данной простейшей логической функции (по сути представляющей собой функцию «логическое ИЛИ») демонстрируется вариант оптимизации. При следовании записи алгебраического значения функции согласно правилам построения СДНФ получаем выражение:

$$y = \bar{x}_1 x_0 + x_1 \bar{x}_0 + x_1 x_0$$

Далее выполняем оптимизацию следующим образом:

1. Заполненные единицами ячейки, стоящие рядом по горизонтали или вертикали могут быть объединены в пары. На рис. 2.25 объединения отображены синим и красным цветом. Одна и та же ячейка может находиться в нескольких объединениях (например ячейка с адресом $x_1 x_0$ на рис. 2.25). Ячейки «по диагонали» не могут быть объединены.

2. Если рядом с ячейкой нет заполненных по горизонтали или вертикали ячеек, то она представляет собой самостоятельное изолированное объединение с одним элементом.

3. Все заполненные единицами ячейки должны входить в объединения.

4. Количество слагаемых в выходном оптимизированном алгебраическом выражении соответствует количеству объединений.

5. В результате объединения переменная, входящая в объединение в прямом и инверсном видах исключается из алгебраического выражения. Переменная, присутствующая в объединении только в прямом или только

инверсном виде, остается в выходном оптимизированном выражении (рис. 2.25).

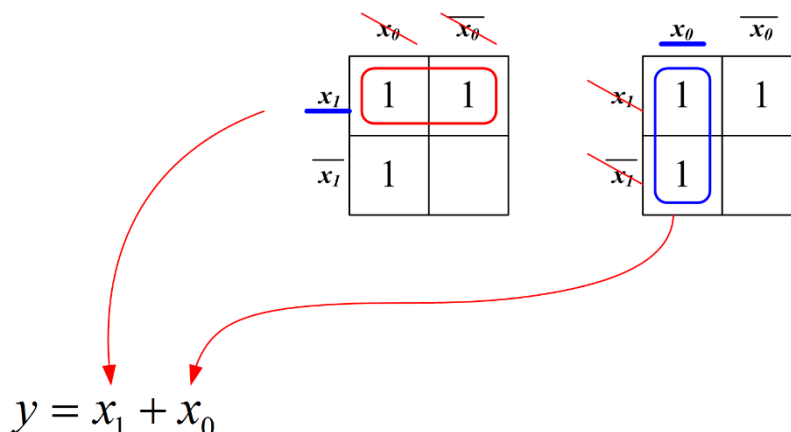


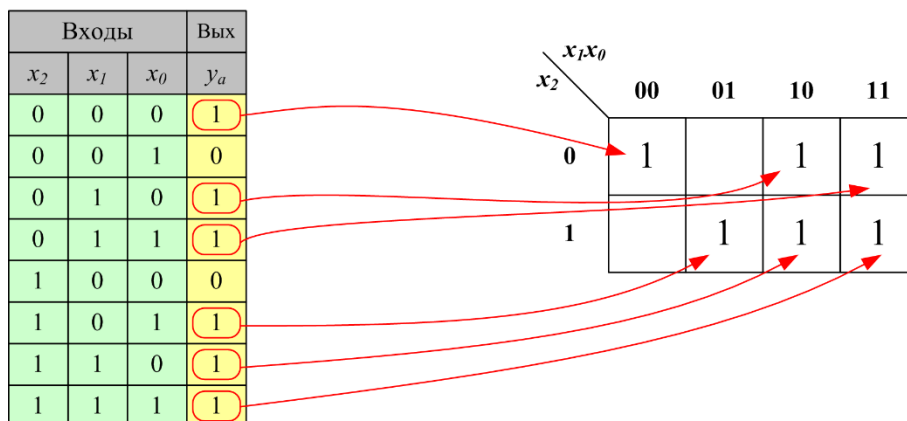
Рис. 2.25. Допустимые вертикальное и горизонтальное объединения, результат оптимизации

На практике способ оптимизации с помощью диаграмм Карно применяется для случая трех, четырех, пяти входных переменных.

Построение диаграммы Карно для трех входных переменных выполняется следующим образом: строится матрица с 4 строками и 2 столбцами. Обозначения строк и столбцов вводятся следующим образом: для 4 столбцов (или строк) **произвольно** выбирают пару входных переменных. Оставшуюся третью переменную указывают вдоль строк. Далее вводят обозначения следующим образом: для строк указывают значения переменной «0» и «1», для столбцов – пары «00», «01», «11», «10». Порядок может быть **произвольный**, важно лишь, чтобы при переходе от столбца к столбцу изменение значения (с «0» на «1» или с «1» на «0») происходило **только для одной переменной**. Аналогичным образом поступают в случае буквенного именования строк и столбцов (с прямым и инверсным обозначением переменных). Здесь также важно, чтобы при переходе от столбца к столбцу изменение знака инверсии происходило только у одной переменной. Примеры построения диаграммы Карно на 8 элементов для логической функции сегмента «а» из примеров выше, приведен на рис. 2.26.

Принцип последующей оптимизации аналогичен описанному выше, однако следуют учесть, что каждой единице в ячейке диаграммы Карно теперь соответствует 3 входные переменные. В результате объединения **парами (двойками)** из трех входных переменных исключается одна, в выходное выражение попадает произведение двух переменных (рис. 2.27а, б). Общие принципы остаются теми же, описанными выше (п. 1–5). Однако для 8-элементной диаграммы возможны и более оптимальные ситуации: объединения «четверками» вида 2x2 и 4x1 (рис. 2.27 в, г).

Сегмент "а"



Сегмент "а"

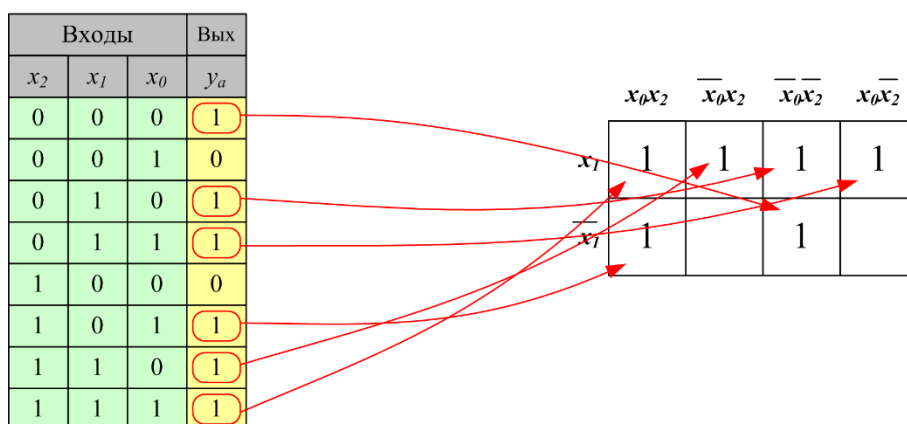


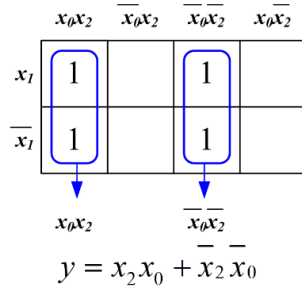
Рис. 2.26. Варианты построения 8-элементной диаграммы Карно для логической функции, заданной таблицей истинности

Стоит отметить, что в диаграмме Карно объединения возможны только парами, объединениями по 4, объединениями по 8, объединениями по 16 рядом стоящих и образующих прямоугольник заполненных ячеек. На рис. 2.28 приведены некорректные и корректные варианты объединений. В силу того, что изменение значения переменной (или знака инверсии при символьном обозначении) от столбца к столбцу происходит только у одной переменной, можно считать самый левый и самый правый столбец диаграммы расположенными «рядом» (рис. 2.29а). В случае полностью заполненной диаграммы (и таблицы истинности, соответственно) выходная функция равна «лог. 1» (рис. 2.29б). В случае полностью заполненной строки (рис. 2.29г) из трех переменных в оптимизированном выражении остается одна.

Для случая четырех входных переменных диаграмма Карно представляет собой 16-элементную матрицу размерности 4x4. Общие принципы оптимизации остаются аналогичными 8-элементной диаграмме (рис. 2.30, 2.31). Особенности являются: возможность объединения «прямоугольником» 4x2 либо 2x4, в том числе разнесенных по сторонам столбцов и

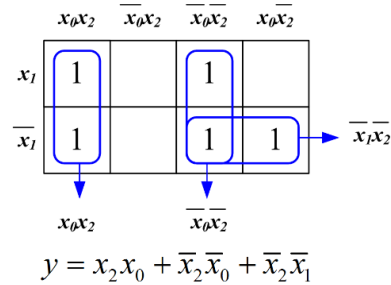
строк (2.31б) и объединение угловых ячеек (2.31г, в силу того, что и у столбцов и у строк изменение значения происходит лишь у одной переменной).

$$y = x_2 x_1 x_0 + x_2 \bar{x}_1 x_0 + \bar{x}_2 x_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 \bar{x}_0$$

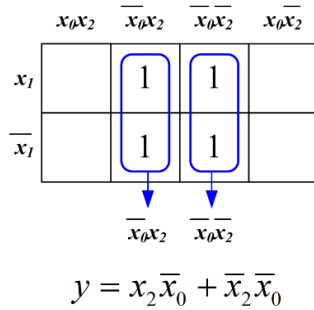


а)

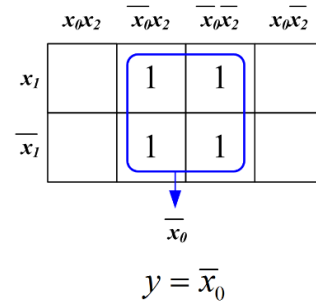
$$y = x_2 x_1 x_0 + x_2 \bar{x}_1 x_0 + \bar{x}_2 x_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 \bar{x}_0 + \bar{x}_2 \bar{x}_1 x_0$$



б)



в)



г)

Рис. 2.27. Варианты объединения элементов 8-элементной диаграммы Карно для логической функции, заданной таблицей истинности

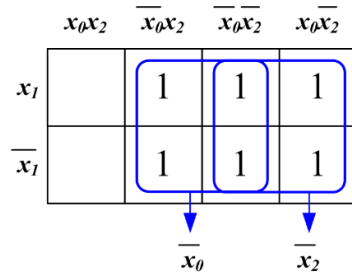
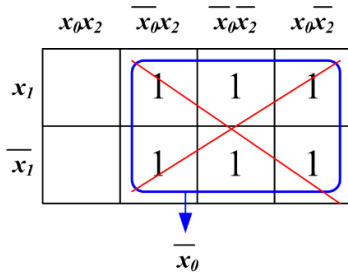
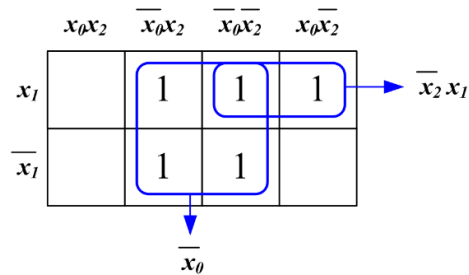
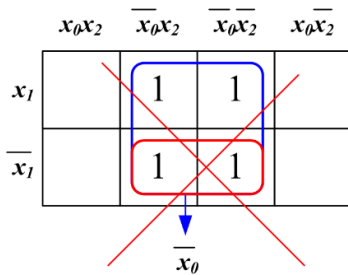


Рис. 2.28. Корректные и некорректные варианты объединений элементов 8-элементной диаграммы Карно

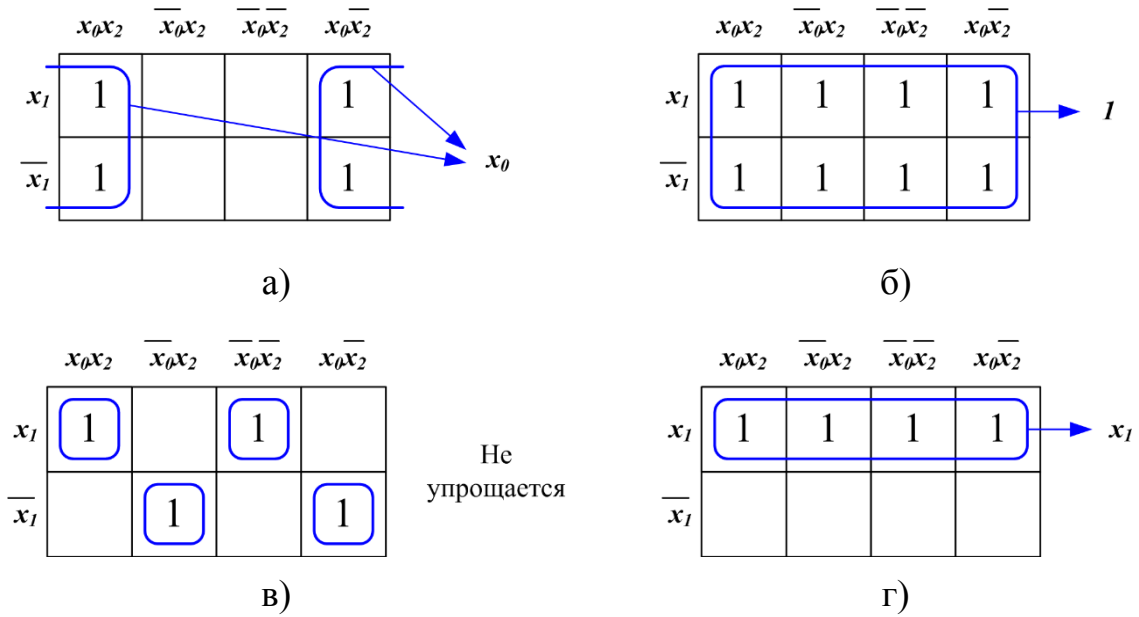


Рис. 2.29. Варианты объединений элементов 8-элементной диаграммы Карно

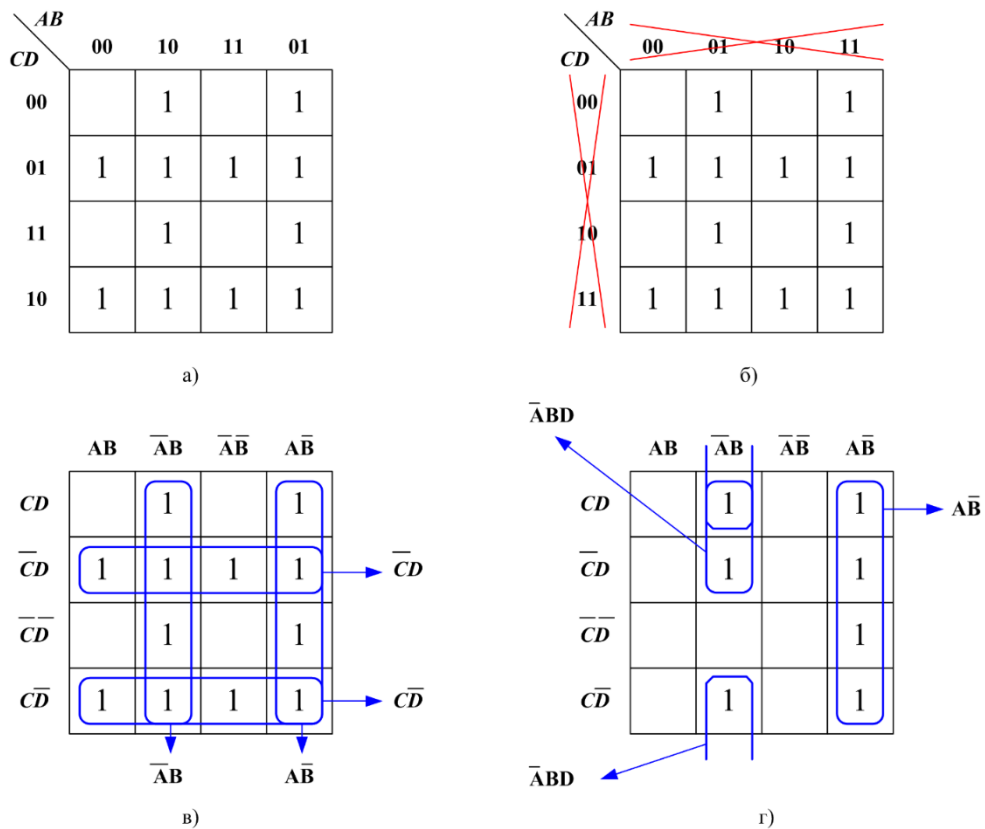


Рис. 2.30. Варианты объединений элементов 16-элементной диаграммы Карно

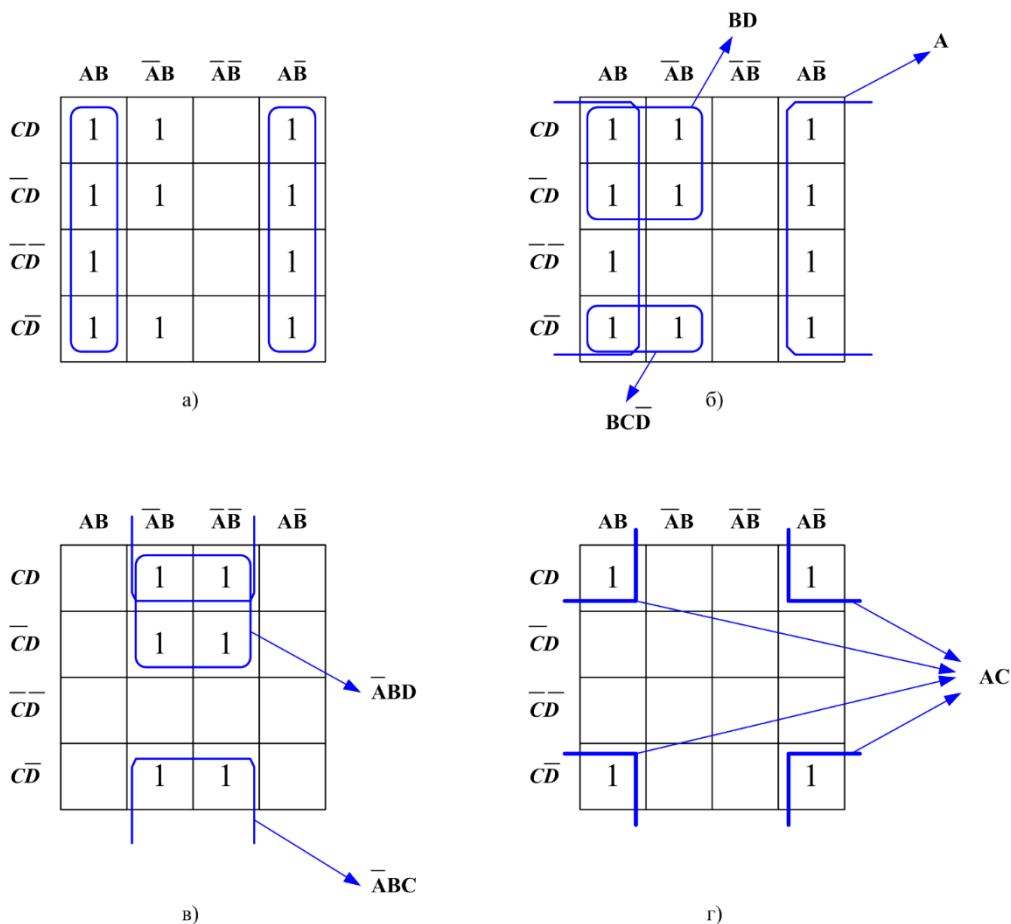


Рис. 2.31. Варианты объединений элементов 16-элементной диаграммы Карно

Задание на лабораторную работу и порядок ее выполнения

Задание 1. Синтез логического устройства управления индикацией десятичных цифр от «0» до «7».

1. Синтезировать логические функции в формах СДНФ и СКНФ для сегментов «а» – «г» семисегментного индикатора вариантов его включения «с общим анодом (ОА)» и «с общим катодом (ОК)» для реализации индикации десятичных цифр от «0» до «7». Результат оформить в виде совмещенной таблицы истинности.

2. Выполнить оптимизацию полученных логических функций используя диаграммы Карно в форме 4x2 (2x4).

3. Записать оптимизированные алгебраические функции. При наличии возможности выполнить дополнительную оптимизацию с помощью тождеств алгебры логики (см. лабораторную работу № 1).

4. Реализовать схему в виде базовых логических элементов («И», «ИЛИ», «ИЛИ-НЕ», «НЕ», «И-ИЛИ» и т.д.) с добавлением элементов типа «логический выход» и «логический вход» для последующего назначения соответствия между логическими входами-выходами и реальными выводами ПЛИС, подключенными к выбранным устройствам ввода (двухпозиционные тумблеры SW1-SW3 или иные) и индикации (4-символьный семисегментный индикатор) модуля «ALT-DIG-002».

5. Выполнить этап «компиляции» проекта в среде проектирования ALTERAQUARTUS. Выполнить программирование устройства «ALT-DIG-002».

6. Выполнить проверку функционирования устройства путем воздействия на заданные органы управления, наблюдая состояние органов индикации.

Содержание отчета

1. Совмещенная таблица истинности (подготовлена для данного задания, см. рис. 2.18).

2. Заполненные карты Карно в формате 4x2 (2x4) с указанием выполненных объединений и промежуточными логическими выражениями.

3. Оптимизированные алгебраические выражения для выходных логических функций каждого сегмента.

4. Копия экрана среды разработки ALTERAQUARTUS с подготовленной логической схемой.

5. Фото состояния органов управления и индикации для каждого состояния устройства.

6. Выводы.

Задание 2. Синтез логического устройства, обеспечивающего индикацию шестнадцатеричных цифр на семисегментном индикаторе.

1. Синтезировать внешний вид шестнадцатеричных цифр «0», «1», «2», «3», «4», «5», «6», «7», «8», «9», «A», «B», «C», «D», «E», «F» при их индикации на семисегментном индикаторе. Определить количество входных дискретных сигналов (переменных) для выбора номера отображаемой шестнадцатеричной цифры.

2. Синтезировать логические функции в формах СДНФ и СКНФ для сегментов «a» – «g» семисегментного индикатора вариантов его включения «с общим анодом (ОА)» и «с общим катодом (ОК)» для реализации индикации шестнадцатеричных цифр.

3. Выполнить оптимизацию полученных логических функций используя диаграммы Карно в форме 4x4.

4. Записать оптимизированные алгебраические функции. При наличии возможности выполнить дополнительную оптимизацию с помощью тождеств алгебры логики (см. лабораторную работу № 1).

5. Реализовать схему в виде базовых логических элементов («2И», «2ИЛИ», «2ИЛИ-НЕ», «НЕ», «3И» и т.д.) с добавлением элементов типа «логический выход» и «логический вход» для последующего назначения соответствия между логическими входами-выходами и реальными выводами ПЛИС, подключенными к выбранным устройствам ввода (двухпозиционные тумблеры SW1-SW8 или иные) и индикации (4-символьный семи-сегментный индикатор) модуля «ALT-DIG-002».

6. Выполнить этап «компиляции» проекта в среде проектирования ALTERAQUARTUS. Выполнить программирование устройства «ALT-DIG-002».

7. Выполнить проверку функционирования устройства путем воздействия на заданные органы управления, наблюдая состояние органов индикации.

Содержание отчета

1. Проект внешнего вида каждой из шестнадцатеричных цифр при их индикации на семисегментном индикаторе.

2. Заполненная совмещенная таблица истинности.

7. Заполненные карты Карно в формате 4x4 с указанием выполненных объединений и промежуточными логическими выражениями.

3. Оптимизированные алгебраические выражения для выходных логических функций каждого сегмента.

4. Копия экрана среды разработки ALTERAQUARTUS с подготовленной логической схемой.

5. Фото состояния органов управления и индикации для каждого состояния устройства.

6. Выводы.

Задание 3. Синтез логического устройства, обеспечивающего максимально полный набор (подмножество) заглавных символов латинского алфавита, индикация которых возможна на семисегментном индикаторе.

1. Сформировать подмножество заглавных символов латинского алфавита, индикация которых возможна на семисегментном индикаторе. Синтезировать внешний вид данных символов при их индикации на семисегментном индикаторе. Определить количество входных дискретных сигналов (переменных) для выбора номера отображаемого символа.

2. Синтезировать логические функции в формах СДНФ и СКНФ для сегментов «а» – «g» семисегментного индикатора вариантов его включе-

ния «с общим анодом (ОА)» и «с общим катодом (ОК)» для реализации индикации символов.

3. Выполнить оптимизацию полученных логических функций используя диаграммы Карно в форме 4х4 и алгебраические тождества.

4. Записать оптимизированные алгебраические функции.

5. Реализовать схему в виде базовых логических элементов («2И», «2ИЛИ», «2ИЛИ-НЕ», «НЕ», «3И» и т.д.) с добавлением элементов типа «логический выход» и «логический вход» для последующего назначения соответствия между логическими входами-выходами и реальными выводами ПЛИС, подключенными к выбранным устройствам ввода (двухпозиционные тумблеры SW1-SW8или иные) и индикации (4-символьный семисегментный индикатор) модуля «ALT-DIG-002».

6. Выполнить этап «компиляции» проекта в среде проектирования ALTERAQUARTUS. Выполнить программирование устройства «ALT-DIG-002».

7. Выполнить проверку функционирования устройства путем воздействия на заданные органы управления, наблюдая состояние органов индикации.

Содержание отчета

1. Перечень (подмножество) заглавных символов латинского алфавита, индикация которых возможна на семисегментном индикаторе

2. Проект внешнего вида каждого из символов при их индикации на семисегментном индикаторе.

3. Заполненная совмещенная таблица истинности.

4. Заполненные карты Карно в формате 4х4 и примененные алгебраические тождества с указанием промежуточных логических выражений.

5. Оптимизированные алгебраические выражения для выходных логических функций каждого сегмента.

6. Копия экрана среды разработки ALTERAQUARTUSс подготовленной логической схемой.

7. Фото состояния органов управления и индикации для каждого состояния устройства.

8. Выводы.

Задание 4. Синтез логического устройства, обеспечивающего максимально полный набор (подмножество) заглавных и строчных символов кириллического алфавита, индикация которых возможна на семисегментном индикаторе.

1. Сформировать подмножество заглавных и строчных символов кириллического алфавита, индикация которых возможна на семисегментном индикаторе. Синтезировать внешний вид данных символов при их индика-

ции на семисегментном индикаторе. Вычислить требуемое количество входных дискретных сигналов (переменных) для выбора номера отображаемого символа.

2. Синтезировать логические функции в формах СДНФ и СКНФ для сегментов «а» – «g» семисегментного индикатора вариантов его включения «с общим анодом (ОА)» и «с общим катодом (ОК)» для реализации индикации символов.

3. Выполнить оптимизацию полученных логических функций используя диаграммы Карно в форме 4х4 и алгебраические тождества.

4. Записать оптимизированные алгебраические функции.

5. Реализовать схему в виде базовых логических элементов («2И», «2ИЛИ», «2ИЛИ-НЕ», «НЕ», «3И» и т.д.) с добавлением элементов типа «логический выход» и «логический вход» для последующего назначения соответствия между логическими входами-выходами и реальными выводами ПЛИС, подключенными к выбранным устройствам ввода (двухпозиционные тумблеры SW1-SW8 или иные) и индикации (4-символьный семисегментный индикатор) модуля «ALT-DIG-002».

6. Выполнить этап «компиляции» проекта в среде проектирования ALTERAQUARTUS. Выполнить программирование устройства «ALT-DIG-002».

7. Выполнить проверку функционирования устройства путем воздействия на заданные органы управления, наблюдая состояние органов индикации.

Содержание отчета

1. Перечень (подмножество) заглавных и строчных символов кириллического алфавита, индикация которых возможна на семисегментном индикаторе.

2. Проект внешнего вида каждого из символов при их индикации на семисегментном индикаторе.

3. Заполненная совмещенная таблица истинности.

4. Заполненные карты Карно в формате 4х4 и примененные алгебраические тождества с указанием выполненных промежуточных логических выражений.

5. Оптимизированные алгебраические выражения для выходных логических функций каждого сегмента.

6. Копия экрана среды разработки ALTERAQUARTUS с подготовленной логической схемой.

7. Фото состояния органов управления и индикации для каждого состояния устройства.

8. Выводы.

Контрольные вопросы

1. Перечислите способы задания логических функций. Каковы достоинства и недостатки каждого из них?
2. Возможна ли реализация логического устройства с одним входным сигналом и одним выходным сигналом? В случае положительного ответа, какие логические функции оно может реализовывать?
3. Каково количество строк таблицы истинности, если число входных сигналов (булевых переменных) равно 6?
4. Как получить запись алгебраической функции в формах СДНФ и СКНФ?
5. Что такое минтерм и макстерм? Как их суммарное количество зависит от количества входных переменных?
6. На каком тождестве алгебры логики основан принцип упрощения логических выражений с помощью диаграмм Карно?
7. Насколько возможна оптимизация логических функций с 6 входными переменными с помощью диаграмм Карно?

ЛАБОРАТОРНАЯ РАБОТА № 3 КОМБИНАЦИОННЫЕ СХЕМЫ СРЕДНЕЙ СТЕПЕНИ ИНТЕГРАЦИИ. ШИФРАТОРЫ, ПРЕОБРАЗОВАТЕЛИ КОДОВ

Цель работы: практическое изучение логических комбинационных схем средней степени интеграции: преобразователей логических кодов, шифраторов. Построение и исследование поведения заданных комбинационных схем, формирование таблицы истинности по анализу их функционирования.

Оборудование: лабораторный комплекс «ALT-DIG-002».

Теоретический материал

К комбинационным логическим устройствам относят логические устройства, состояние выходных логических сигналов зависит только от состояния входных логических сигналов в каждый конкретный момент времени и не зависит от предыстории данных сигналов или их комбинаций, т.е. комбинационные логические устройства не имеют свойства памяти. Общий вид комбинационного логического устройства показан на рис. 2.1 (лабораторная работа № 2), а функциональные зависимости выходных сигналов от входных – формулами 2.1.

Для понимания принципов работы шифраторов (и, в следующих работах – дешифраторов) необходимо описать один из применяемых способов

кодирования чисел (использованием двоичных разрядов) – унитарный прямой и унитарный обратный коды.

Дес. число	HEX число	Двоичный код				Прямой унитарный код																Обратный унитарный код															
		2 ³	2 ²	2 ¹	2 ⁰	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0		
1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1	0	1			
2	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	1	1	1	1	1	1	0	1	1			
3	3	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1	1	1	1	0	1	1	1	1			
4	4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1	1	1	1	1	1	0	1	1	1	1			
5	5	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1	1	1	1	1			
6	6	0	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	1	1	1			
7	7	0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1		
8	8	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1		
9	9	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1		
10	A	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1		
11	B	1	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1		
12	C	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1		
13	D	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1		
14	E	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1		
15	F	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1		

Рис. 3.1. Прямой и обратный позиционный унитарный код

Число в унитарном коде представляет собой комбинацию двоичных разрядов (многоразрядный код), в которой лишь в единственном из разрядов присутствует активный уровень. Для прямого унитарного кода активный уровень соответствует лог. «1», для обратного унитарного кода – уровню лог. «0». Остальные разряды комбинации имеют неактивный уровень (для прямого – лог. «0», для обратного – лог. «1»). При этом номер активного разряда совпадает с соответствующим десятичным числом (точнее, его двоичным представлением).

Шифратор – комбинационное логическое устройство, осуществляющее преобразование входных сигналов в позиционном коде в выходные сигналы – числа в двоичном коде. Шифратор имеет m входов, пронумерованных десятичными числами (0, 1, 2... $m - 1$) и n выходов, причем $2^n \geq m$. Подача активного сигнала на один из входов приводит к появлению на выходе шифратора n -разрядного двоичного числа, соответствующего номеру активного входа. Схемотехническое обозначение одного из вариантов шифратора приведено на рис. 3.2. Поскольку шифратор является комбинационным устройством, то состояние каждого из выходных сигналов может быть описано функциональной зависимостью от входных сигналов, в том числе в виде таблицы истинности (рис. 3.3).

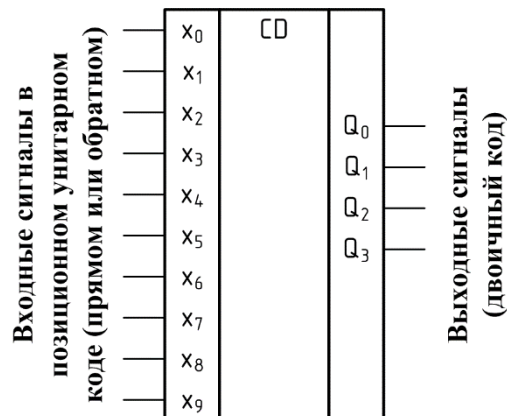


Рис. 3.2. Схематическое обозначение шифратора

N	Входные сигналы										Выходной код			
	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	Q_3	Q_2	Q_1	Q_0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0	0	1
2	0	0	1	0	0	0	0	0	0	0	0	0	1	0
3	0	0	0	1	0	0	0	0	0	0	0	0	1	1
4	0	0	0	0	1	0	0	0	0	0	0	1	0	0
5	0	0	0	0	0	1	0	0	0	0	0	1	0	1
6	0	0	0	0	0	0	1	0	0	0	0	1	1	0
7	0	0	0	0	0	0	0	1	0	0	0	1	1	1
8	0	0	0	0	0	0	0	0	1	0	1	0	0	0
9	0	0	0	0	0	0	0	0	0	1	1	0	0	1

Рис. 3.3. Совмещенная таблица истинности шифратора, представленного на рис. 3.2

Используя принципы анализа и синтеза логических функций (лабораторная работа №2, способы описания СДНФ и СКНФ) можно сформировать алгебраические выражения для выходных сигналов и синтезировать его внутреннюю схему (рис. 3.4).

N	Входные сигналы										Выходной код			
	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	Q_3	Q_2	Q_1	Q_0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0	0	0	0	0	0	1
2	0	0	1	0	0	0	0	0	0	0	0	0	1	0
3	0	0	0	1	0	0	0	0	0	0	0	0	1	1
4	0	0	0	0	1	0	0	0	0	0	0	1	0	0
5	0	0	0	0	0	1	0	0	0	0	0	1	0	1
6	0	0	0	0	0	0	1	0	0	0	0	1	1	0
7	0	0	0	0	0	0	0	1	0	0	0	1	1	1
8	0	0	0	0	0	0	0	0	1	0	1	0	0	0
9	0	0	0	0	0	0	0	0	0	1	1	0	0	1

Рис. 3.4. Анализ значений логических функций Q_3 – Q_0 шифратора

В соответствии с таблицей истинности (рис. 3.4) получены следующие алгебраические выражения (3.1) и синтезирована схема (рис. 3.5).

$$\begin{aligned}
 Q_3 &= x_8 + x_9 \\
 Q_2 &= x_7 + x_6 + x_5 + x_4 \\
 Q_1 &= x_7 + x_6 + x_3 + x_2 \\
 Q_0 &= x_9 + x_7 + x_5 + x_3 + x_1
 \end{aligned}
 \tag{3.1}$$

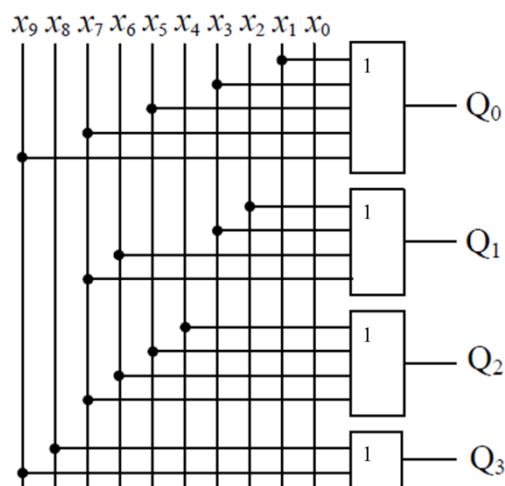


Рис. 3.5. Анализ значений логических функций Q_3 – Q_0 шифратора

Приведенная структура шифратора предназначена для обработки **зависимых** однопозиционных сигналов, когда из всех входных сигналов должен быть активным лишь один. При нескольких одновременно действующих входных сигналах на выходе шифратора получается код, соответствующий операции «поразрядное ИЛИ» над кодами отдельных сигналов.

Выходной двоичный код не соответствует ни одному активному сигналу, т.е. является искаженным. Такая ситуация в большинстве случаев неприемлема.

Одновременно действующие однопозиционные сигналы называются независимыми. Например, таковыми могут являться сигналы запросов на прерывание от периферийных устройств, взаимодействующих с микропроцессором. Для обработки независимых однопозиционных сигналов применяют приоритетные шифраторы, в которых различным одновременно действующим сигналам присваивается приоритет. Часто полагают, что входные сигналы с большим индексом обладают большим приоритетом.

Приоритетные шифраторы имеют более сложную логическую схему и могут быть выполнены по двум вариантам:

- схема выделения старшего разряда входного кода и шифратор + классический шифратор (рис. 3.6);
- приоритетный шифратор как единое устройство.

Схема выделения старшего сигнала имеет K входов и столько же выходов и работает по следующему алгоритму: из нескольких активных входных сигналов на выход этой схемы передается только один активный сигнал, имеющий наивысший приоритет, а все остальные активные сигналы с меньшим приоритетом передаются на выход как неактивные (таблица истинности, рис. 3.7). Выделенный один активный сигнал далее легко может быть преобразован в двоичный код обычным шифратором.

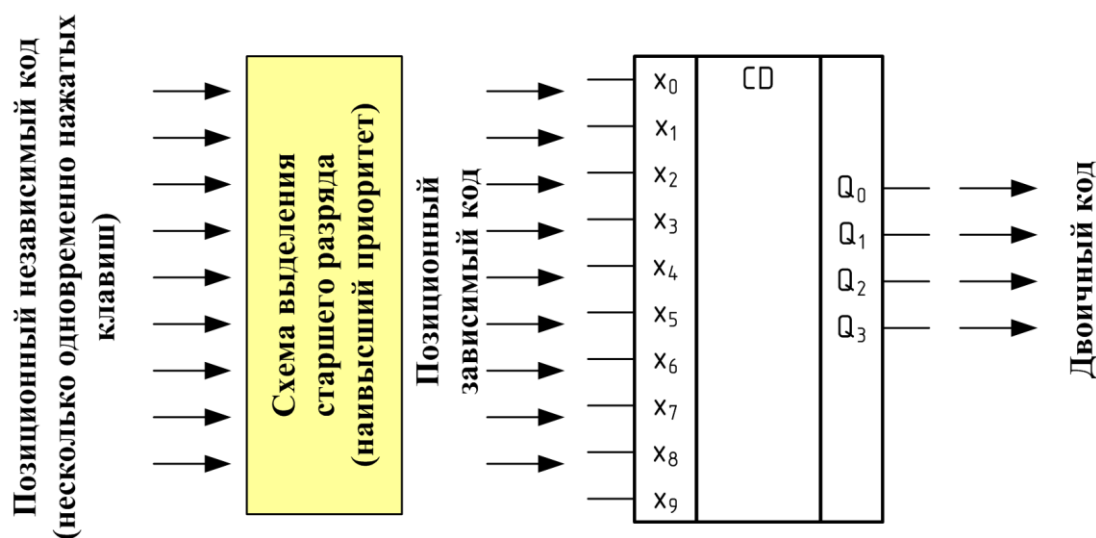


Рис. 3.6. Вариант построение приоритетного шифратора со схемой выделения старшего разряда

N	Входной код				Выходной код			
	x_3	x_2	x_1	x_0	f_3	f_2	f_1	f_0
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	0
3	0	0	1	1	0	0	1	0
4	0	1	0	0	0	1	0	0
5	0	1	0	1	0	1	0	0
6	0	1	1	0	0	1	0	0
7	0	1	1	1	0	1	0	0
8	1	0	0	0	1	0	0	0
9	1	0	0	1	1	0	0	0

Рис. 3.7. Таблица истинности четырехразрядной схемы выделения старшего разряда

Практически применяемые шифраторы помимо основных входов унитарного кода могут иметь дополнительные входные сигналы, помимо основных выходов двоичного кода – дополнительные выходные сигналы о состоянии шифратора. Рассмотрим один из вариантов приоритетного шифратора – микросхему K555IB1 (74HC148): микросхема принимает на вход двоичные сигналы в обратном унитарном коде, формирует на выходе поразрядно-инверсный двоичный код, имеет вход разрешения работы, дополнительные выходы состояния GS, EO. В схемотехническом обозначении данного шифратора в среде ALTEQAQUARTUS активный уровень «лог.0» на входе и выходе обозначен окружностью и дополнительным символом «N», заканчивающим наименование вывода (рис. 3.8). Младшему разряду трехразрядного выходного двоичного кода соответствует имя «A0N», старшему – имя «A2N».

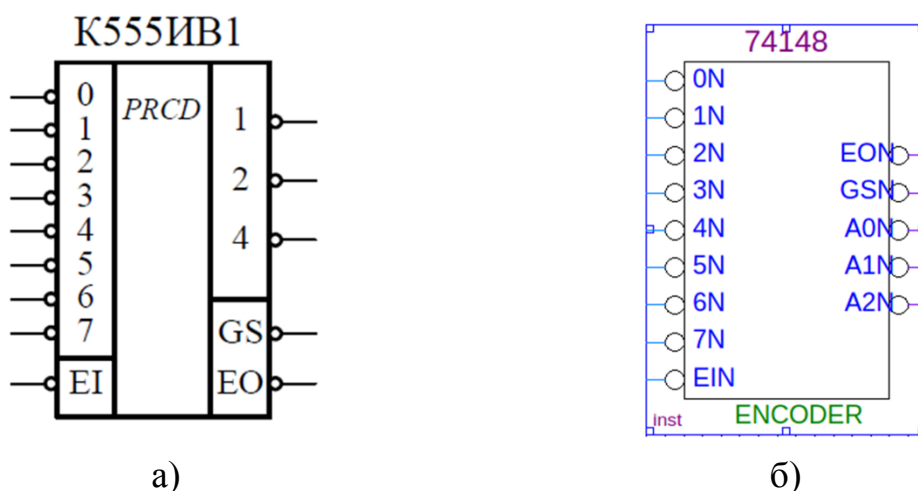


Рис. 3.8. Приоритетный шифратор K555IB1 (74HC148) и его схемный эквивалент в среде проектирования ALTEQAQUARTUS

Таблица истинности данного шифратора приведена на рис. 3.9.

Входы									Выходы				
EI	0	1	2	3	4	5	6	7	A2	A1	A0	GS	EO
1	X	X	X	X	X	X	X	X	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	X	X	X	X	X	X	X	0	0	0	0	0	1
0	X	X	X	X	X	X	0	1	0	0	1	0	1
0	X	X	X	X	X	0	1	1	0	1	0	0	1
0	X	X	X	X	0	1	1	1	0	1	1	0	1
0	X	X	X	0	1	1	1	1	1	0	0	0	1
0	X	X	0	1	1	1	1	1	1	0	1	0	1
0	X	0	1	1	1	1	1	1	1	1	0	0	1
0	0	1	1	1	1	1	1	1	1	1	1	0	1

Шифратор не активен
Ни один вход не активен

Есть активный вход, на выходе двоичный код

Рис. 3.9. Таблица истинности приоритетного шифратора K555ИБ1 (74НС148)

Преобразователь кодов—комбинационное логическое устройство, в общем случае осуществляющее преобразование входных сигналов, представленных в одном из вариантов двоичного кодирования в выходные сигналы, представленные в другом варианте двоичного кодирования. Примерами преобразователей кодов могут быть: преобразователь из двоичного кода в код Грэя, преобразователь из прямого двоичного кода в дополнительный код, преобразователь двоичного кода в двоично-десятичный код и т.д.

В качестве примера рассмотрим преобразователи кодов из двоично-десятичного кода в код семисегментного индикатора серии 74xx47. В среде проектирования ALERAQUARTUS схемотехническое изображение данного преобразователя кода выглядит следующим образом (рис. 3.10).

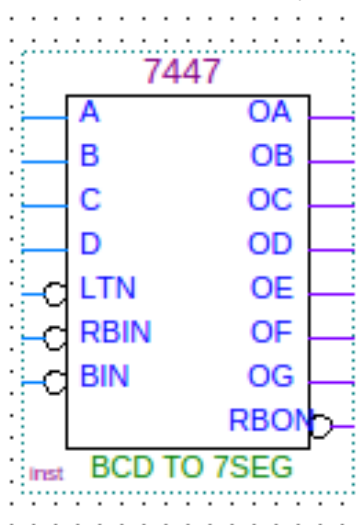


Рис. 3.10. Схемотехническое изображение преобразователя кодов 74xx47 в среде проектирования ALERAQUARTUS

Преобразователь кодов имеет 4 входных сигнала «А»–«D» (двоично-десятичный код, разряд «А» – младший, разряд «D» –старший), служебные входные сигналы «LTN», «BIN», «RBIN», 7 выходных сигналов «ОА»–«ОG» на каждый из сегментов «а»–«g» семисегментного индикатора и служебный выход «RBON».

Выходы на семисегментный индикатор предназначены для управления индикатором, имеющим схему включения «общий анод», поэтому для управления свечением сегментов индикатора, имеющего схему включения «общий катод», данные выходные сигналы необходимо инвертировать дополнительно установленными инверторами.

Сигнал «LTN» (активный уровень – лог. «0») переводит индикатор в режим тестирования свечения всех сегментов индикатора. Сигналы «BIN», «RBIN» («BlinkingInput», активный уровень – лог. «0») гасят семисегментный индикатор. Таблица истинности (англоязычный вариант) работы данного преобразователя кодов приведена на рис. 3.11. В таблице символом «L» обозначен низкий уровень входного/выходного сигнала (лог. «0»), символом «H» – высокий уровень (лог. «1»).

Вход. код или ф-я	Входы						BI/RBO	Выходы						
	LT	RBI	D	C	B	A		a	b	c	d	e	f	g
0	H	H	L	L	L	L	H	L	L	L	L	L	L	H
1	H	X	L	L	L	H	H	H	L	L	H	H	H	H
2	H	X	L	L	H	L	H	L	L	H	L	L	H	L
3	H	X	L	L	H	H	H	L	L	L	L	H	H	L
4	H	X	L	H	L	L	H	H	L	L	H	H	L	L
5	H	X	L	H	L	H	H	L	H	L	L	H	L	L
6	H	X	L	H	H	L	H	H	H	L	L	L	L	L
7	H	X	L	H	H	H	H	L	L	L	H	H	H	H
8	H	X	H	L	L	L	H	L	L	L	L	L	L	L
9	H	X	H	L	L	H	H	L	L	L	H	H	L	L
10	H	X	H	L	H	L	H	H	H	H	L	L	H	L
11	H	X	H	L	H	H	H	H	H	L	L	H	H	L
12	H	X	H	H	L	L	H	H	L	H	H	H	L	L
13	H	X	H	H	L	H	H	L	H	H	L	H	L	L
14	H	X	H	H	H	L	H	H	H	H	L	L	L	L
15	H	X	H	H	H	H	H	H	H	H	H	H	H	H
BI	X	X	X	X	X	X	L	H	H	H	H	H	H	H
RBI	H	L	L	L	L	L	L	H	H	H	H	H	H	H
LT	L	X	X	X	X	X	H	L	L	L	L	L	L	L

Рис. 3.11. Таблица истинности преобразователя кодов 74xx47

Задание на лабораторную работу и порядок ее выполнения

Общие условия и требования

1. Перед началом выполнения данной и последующих работ ознакомиться с порядком работы с лабораторным комплектом «ALT–DIG–002» и порядком создания схмотехнического проекта в среде проектирования ALTERAQUARTUS (см. приложение).

2. Целью лабораторной работы является:

- 1) синтез схемы и исследование функционирования шифратора;
- 2) синтез схемы и исследование функционирования приоритетного шифратора;
- 3) синтез схемы выделения младшего разряда и анализ ее функционирования в составе схемы шифратора;
- 4) синтез схемы для исследования функционирования преобразователя «двоично-десятичный код» – «код семисегментного индикатора»;
- 5) синтез комбинированной схемы «приоритетный шифратор» – «преобразователь кодов».

Задание 1 (порядок выполнения). Синтез схемы и исследование функционирования шифратора.

1. В среде проектирования ALTERAQUARTUS подготовить проект с именем lab3a, содержащий заготовку схемы с 10 логическими входами и 4 логическими выходами. Задать следующие имена входов-выходов:

для входов: $x_0, x_1, \dots, x_9$.

для выходов: Q_0, Q_1, Q_2, Q_3 .

2. Выполнить этап назначения логических входов-выходов схемы к физическим выводам микросхемы программируемой логики (ПЛИС) в соответствии с разделом 3 приложения и таблицами соответствия выводов ПЛИС и устройств ввода и индикации (рис. 1.4). Рекомендуется следующее соответствие логических входов-выходов (рис. 1.5): для входных логических сигналов « x_0 », « x_1 », ..., « x_9 » – двухпозиционные тумблеры «SW1»–«SW8» и кнопки «SB1»–«SB4», для выходных логических сигналов « Q_0 », « Q_1 », « Q_2 », « Q_3 ». Важно однозначно сопоставить соответствие номера входа, выходного разряда и расположение органа управления и индикации на модуле.

3. Выполнить сборку схемы шифратора (рис. 3.5), используя элементы «ИЛИ».

4. Подготовить таблицу истинности для описания функционирования шифратора в соответствии с шаблоном (рис. 3.3).

5. Выполнить компиляцию проекта, выполнить «программирование» микросхемы ПЛИС (раздел 3 приложения).

6. Задавая необходимые логические сигналы путем переключения тумблеров «SW1»–«SW8» и нажатия кнопок «SB1»–«SB4» (или иных, на которые назначены логические входы «x0», «x1»,...,«x9») визуально (по назначенным светодиодам) контролировать выходной двоичный код на выходах (светодиодах) «Q0», «Q1», «Q2», «Q3». Заполнить таблицу истинности шифратора. Выполнить проверку и запись в таблицу истинности состояний выходов шифратора при нескольких одновременно действующих активных уровнях на входах.

Содержание отчета

1. Копии экрана среды разработки ALTEAQUARTUSc подготовленной схемой.

2. Фото состояния органов управления и индикации для нескольких входных сигналов.

3. Заполненная таблица истинности шифратора.

4. Выводы.

Задание 2 (порядок выполнения). Синтез схемы и исследование функционирования приоритетного шифратора.

1. В среде проектирования ALTEAQUARTUSподготовить проект с именем lab3b, содержащий заготовку схемы с 9 логическими входами и 5 логическими выходами. Задать следующие имена входов-выходов:

для входов: x0, x1, ... , x7, ei.

для выходов: a0, a1, a2, eo, gs.

2. Выполнить этап назначения логических входов-выходов схемы к физическим выводам микросхемы программируемой логики (ПЛИС) в соответствии с разделом 3 приложения и таблицами соответствия выводов ПЛИС и устройств ввода и индикации (рис. 1.4). Рекомендуется следующее соответствие логических входов-выходов (рис. 1.5): для входных логических сигналов «x0», «x1»,...,«x7», «ei» – двухпозиционные тумблеры «SW1»–«SW8»и кнопки «SB1»–«SB4», для выходных логических сигналов «a0», «a1», «a2», «eo», «gs».Важно однозначно сопоставить соответствие номера входа, выходного разряда и расположение органа управления и индикации на модуле.

3. Установить на схему приоритетный шифратор 74148 (74xx148), подключить его входы и выходы к соответствующим цепям, подготовленным в пп. 1, 2 задания. Учитывая, что двоичный код на выходе шифратора по-разрядно инвертирован, добавить необходимые логические элементы инверсии.

4. Подготовить таблицу истинности для описания функционирования шифратора в соответствии с шаблоном (рис. 3.9).

5. Выполнить компиляцию проекта, выполнить «программирование» микросхемы ПЛИС (раздел 3 приложения).

6. Задавая необходимые логические сигналы путем переключения тумблеров «SW1»–«SW8» и нажатия кнопок «SB1»–«SB4» (или иных, на которые назначены логические входы «x0», «x1»,...,«x7», «ei») визуально (по назначенным светодиодам) контролировать выходной двоичный код на выходах (светодиодах) «a0», «a1», «a2», «eo», «gs». Заполнить таблицу истинности приоритетного шифратора. Выполнить проверку и запись в таблицу истинности состояний выходов шифратора при нескольких одновременно действующих активных уровнях на входах.

Содержание отчета

1. Копии экрана среды разработки ALTERAQUARTUS с подготовленной схемой.

2. Фото состояния органов управления и индикации для ряда входных сигналов.

3. Заполненная таблица истинности приоритетного шифратора.

4. Выводы.

Задание 3 (порядок выполнения). Синтез схемы выделения младшего разряда и анализ ее функционирования в составе схемы шифратора.

1. На базе проекта, подготовленного в задании 1 подготовить проект с именем lab3с с тем же количеством входных и выходных логических цепей.

2. Базируясь на описании схемы выделения старшего разряда и таблице истинности, приведенной на рис. 3.7, подготовить таблицу истинности и реализовать в виде логической схемы блок выделения младшего разряда при нескольких одновременно действующих активных уровнях входных сигналов. Включить разработанную логическую схему между входными цепями и входами дешифратора, подготовленного в задании 1 данной лабораторной работы.

3. Подготовить таблицу истинности для описания функционирования шифратора в соответствии с шаблоном (рис. 3.9).

4. Выполнить компиляцию проекта, выполнить «программирование» микросхемы ПЛИС (раздел 3 приложения).

5. Задавая необходимые логические сигналы путем переключения тумблеров «SW1»–«SW8» и нажатия кнопок «SB1»–«SB4» (или иных, на которые назначены логические входы «x0», «x1»,...,«x9») визуально (по назначенным светодиодам) контролировать выходной двоичный код на

выходах (светодиодах) «Q0», «Q1», «Q2», «Q3». Заполнить таблицу истинности шифратора. Выполнить проверку и запись в таблицу истинности состояний выходов шифратора при нескольких одновременно действующих активных уровнях на входах. Сопоставить поведение шифратора с поведением шифратора, подготовленного в задании 1 данной лабораторной работы.

Содержание отчета

1. Копии экрана среды разработки ALTERAQUARTUSc подготовленной схемой.
2. Фото состояния органов управления и индикации для нескольких входных сигналов.
3. Заполненная таблица истинности шифратора со схемой выделения младшего разряда.
4. Выводы.

Задание 4 (порядок выполнения). Синтез схемы для исследования функционирования преобразователя «двоично-десятичный код» – «код семи-сегментного индикатора».

1. В среде проектирования ALTERAQUARTUSподготовить проект с именем lab3d, содержащий заготовку схемы с 6 логическими входами и 7 логическими выходами. Задать следующие имена входов-выходов:

для входов: x0, x1, x2, x3, lt, bin.

для выходов: ya, yb, yc, yd, ye, yf, yg.

2. Выполнить этап назначения логических входов-выходов схемы к физическим выводам микросхемы программируемой логики (ПЛИС) в соответствии с разделом 3 приложения и таблицами соответствия выводов ПЛИС и устройств ввода и индикации (рис. 1.4). Рекомендуются следующее соответствие логических входов-выходов (рис. 1.5): для входных логических сигналов «x0», «x1», ..., «x3», «lt», «bin» – двухпозиционные тумблеры «SW1»–«SW8», для выходных логических сигналов «ya», «yb», «yc», «yd», «ye», «yf», «yg» – соответствующие сегменты любого из символов четырехразрядного семисегментного индикатора (приложение, рис. П.50).

3. Установить на схему преобразователь кода 7447 (7447), подключить его входы и выходы к соответствующим цепям, подготовленным в пп. 1, 2 задания. Учитывая, что код семисегментного индикатора на выходе преобразователя инвертирован, добавить необходимые логические элементы инверсии.

4. Выполнить компиляцию проекта, выполнить «программирование» микросхемы ПЛИС (раздел 3 приложения).

5. На базе таблицы истинности, приведенной на рис. 3.11 подготовить таблицу работы индикатора.

6. Задавая необходимые логические сигналы путем переключения тумблеров «SW1»–«SW8» визуально (по семисегментному индикатору) убедиться в корректности отображаемой информации (десятичных цифр). Дополнительно исследовать режимы работы индикатора при различных уровнях лог. «0» и лог. «1» (режим проверки, режим гашения). Провести исследование работы преобразователя при подаче на вход двоичного кода со значением, большим 9_{10} .

Содержание отчета

1. Копии экрана среды разработки ALTERAQUARTUSc подготовленной схемой.
2. Фото состояния органов управления и индикации для всех сочетаний входных сигналов.
3. Заполненная таблица истинности преобразователя кодов.
4. Выводы.

Задание 5 (порядок выполнения). Синтез комбинированной схемы «приоритетный шифратор» – «преобразователь кодов».

1. На основе схем, подготовленных в заданиях 1, 2, 3 и 4 данной лабораторной работы реализовать индикацию выходного кода шифратора(-ов) на семисегментном индикаторе.

2. Расширить возможности индикации состояния шифратора, обеспечив отключение индикатора при отсутствии активных сигналов на входах шифратора(-ов). Для этого выполнить анализ логики формирования выходных сигналов «EON», «GSN» (для шифратора 74148) и входных сигналов «LT», «BIN», «RBIN».

3. Задавая необходимые логические сигналы путем переключения тумблеров «SW1»–«SW8» визуально (по семисегментному индикатору) убедиться в корректности отображаемой информации (десятичных цифр).

Содержание отчета

1. Копии экрана среды разработки ALTERAQUARTUSc подготовленной схемой.
2. Фото состояния органов управления и индикации для нескольких характерных сочетаний входных сигналов.
3. Выводы.

Контрольные вопросы

1. Что такое унитарный код? Каким образом его можно получить из двоичного кода?
2. Объясните назначение и структурную схему шифратора?
3. Какой шифратор называется полным?
4. Каким должно быть соотношение между количеством входов и выходов в схеме полного шифратора?
5. Возможно ли построение шифратора на логических элементах «ИЛИ»?
6. Почему шифраторы называют преобразователями десятичного кода в двоичный код?

ЛАБОРАТОРНАЯ РАБОТА № 4 КОМБИНАЦИОННЫЕ СХЕМЫ СРЕДНЕЙ СТЕПЕНИ ИНТЕГРАЦИИ. МУЛЬТИПЛЕКСОРЫ

Цель работы: практическое изучение логических комбинационных схем средней степени интеграции – мультиплексоров. Построение и исследование поведения заданных комбинационных схем, формирование таблицы истинности по анализу их функционирования.

Оборудование: лабораторный комплекс «ALT-DIG-002».

Теоретический материал

Назначение мультиплексоров (*Multiplexer*– MUX) – коммутация одного из нескольких входных информационных сигналов на выход мультиплексора. Номер коммутируемого информационного канала определяется адресными входами. Мультиплексор имеет несколько информационных входов $D_0, D_1, \dots, D_{N-1}$, адресные входы $A_0, A_1, \dots, A_{N-1}$, вход для подачи сигнала разрешения работы (С либо Е) и один выход DO (рис. 4.1). Количество **информационных входов** N и число **адресных входов** n связаны соотношением $N = 2^n$. Коммутация выполняется на логическом уровне с помощью базовых логических элементов (рис. 4.2).

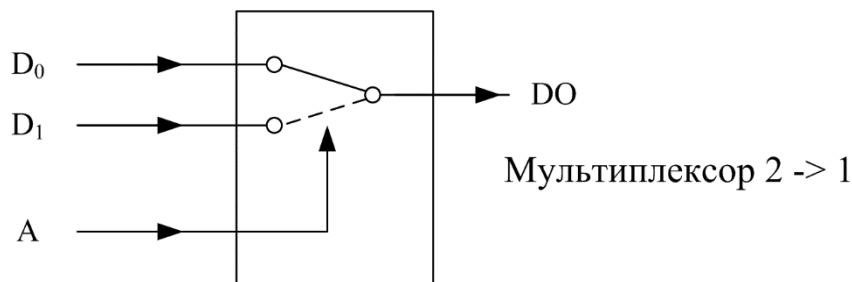


Рис. 4.1. Принцип функционирования мультиплексора 2->1

$$DO = D_0 \bar{A} + D_1 A$$

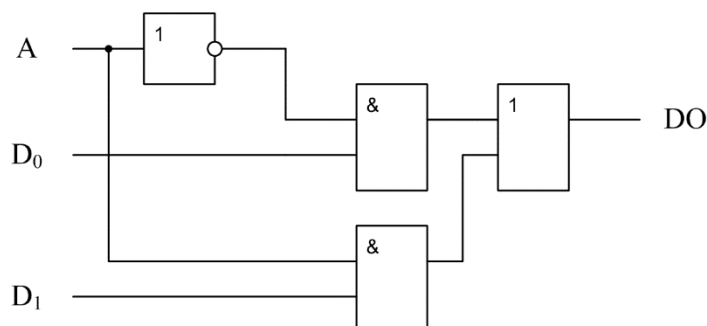


Рис. 4.2. Алгебраическое выражение выходной функции DO и схема ее реализации (для мультиплексора 2 -> 1)

В случае наличия входа разрешения работы (С либо Е) и наличия на нем неактивного уровня выходной информационный сигнал DO мультиплексора как правило равен лог. «0». При наличии входного управляющего сигнала перевода в третье состояние (в Z-состояние) выход мультиплексора может быть переведен в высокоомное состояние. Схемотехническое обозначение мультиплексоров в стандартах ГОСТ Р выглядит следующим образом (рис. 4.3, рис. 4.4).

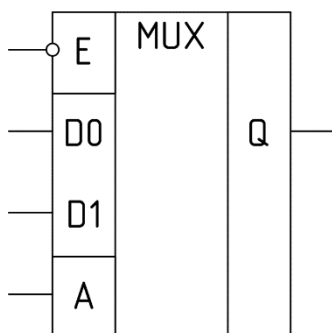


Рис. 4.3. Схемотехническое обозначение мультиплексора «2→1» имеющего вход разрешения работы

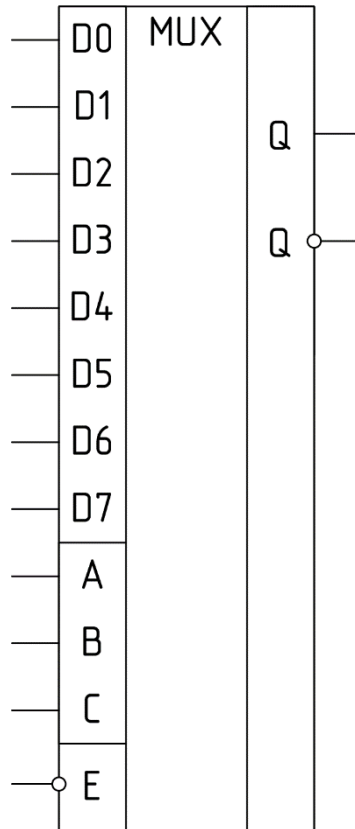


Рис. 4.4. Схемотехническое обозначение мультиплексора «8→1»
имеющего вход разрешения работы (К1533КП15)

Алгебраическое выражение для выходной функции мультиплексора, приведенного на рис. 4.3 выглядит следующим образом:

$$Q = D_0 \bar{A} \bar{C} + D_1 A \bar{C} \quad (4.1)$$

Алгебраическое выражение для выходной функции мультиплексора, приведенного на рис. 4.4 выглядит следующим образом:

$$Q = D_0 \cdot \bar{E} \cdot (\bar{C} \bar{B} \bar{A}) + D_1 \cdot \bar{E} \cdot (\bar{C} \bar{B} A) + D_2 \cdot \bar{E} \cdot (\bar{C} B \bar{A}) + \dots + D_7 \cdot \bar{E} \cdot (C B A) \quad (4.2)$$

Таблица 4.1

Таблица истинности мультиплексора «8→1»

E	C	B	A	Q	$\overline{Q}$
1	X	X	X	0	1
0	0	0	0	D0	$\overline{D0}$
0	0	0	1	D1	$\overline{D1}$
0	0	1	0	D2	$\overline{D2}$
0	0	1	1	D3	$\overline{D3}$
0	1	0	0	D4	$\overline{D4}$
0	1	0	1	D5	$\overline{D5}$
0	1	1	0	D6	$\overline{D6}$
0	1	1	1	D7	$\overline{D7}$

В табл. 4.2 приведен перечень и функциональное назначение наиболее популярные микросхем мультиплексоров отечественных и зарубежных серий.

Таблица 4.2

Перечень отечественных и зарубежных микросхем мультиплексоров

№	Обозначение отечественной м/сх	Обозначение зарубежной м/сх	Тип мультиплексора
1	K1533КП2	74xx153	Сдвоенный мультиплексор 4→1
2	K1533КП7	74xx151	Мультиплексор 8→1 с входом разрешения
3	K1533КП16	74xx157	Счетверенный мультиплексор 2→1 с общими адресными линиями
4	K155КП1	—	Мультиплексор 16→1 с входом разрешения

На рис. 4.5 в нотации ANSI-стандарта обозначений логических элементов приведена внутренняя логическая схема мультиплексора 74xx157 (K1533КП16). Аналогичным образом строятся мультиплексоры иной структуры.

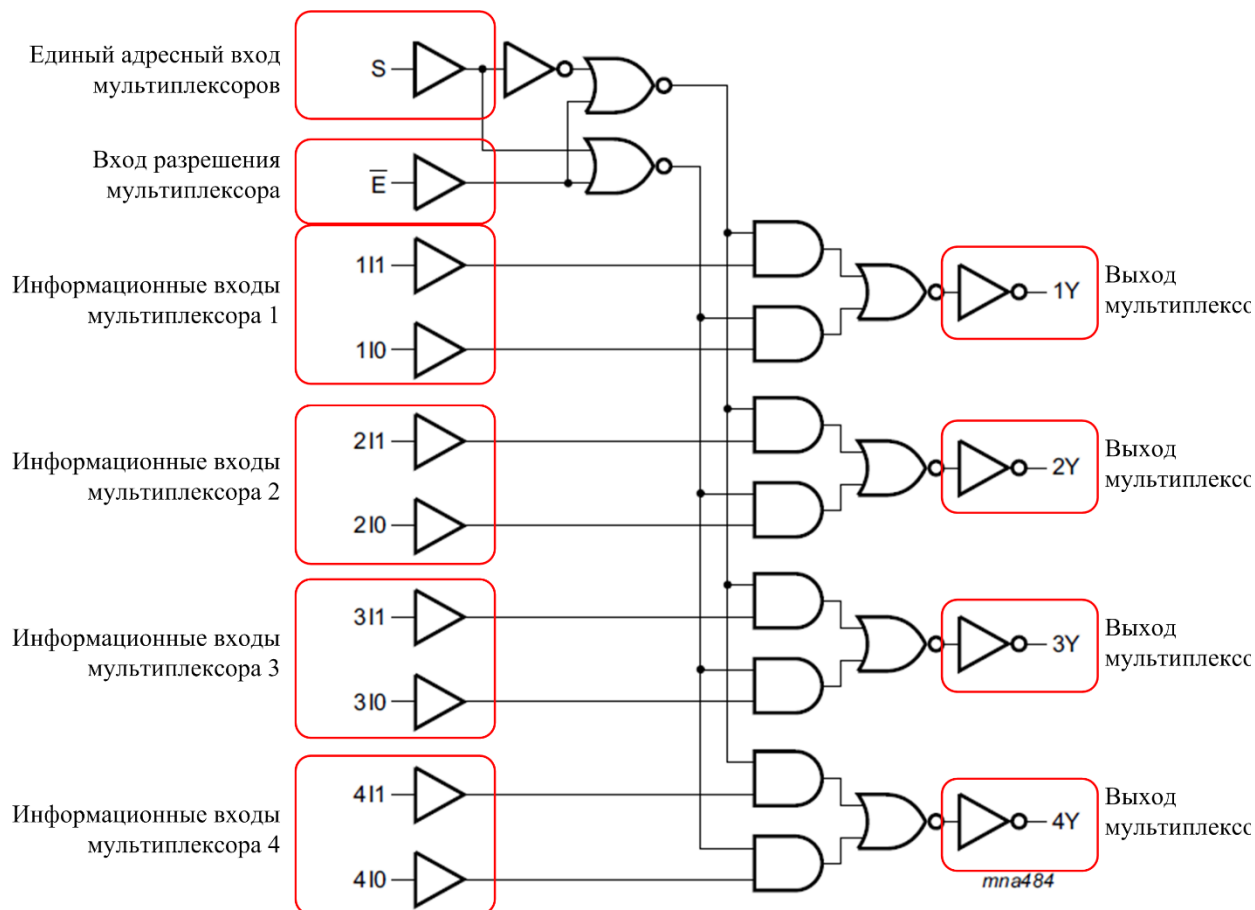


Рис. 4.5. Внутренняя логическая схема мультиплексора

При необходимости коммутации большого числа информационных каналов и отсутствия мультиплексора с данным количеством входов применяют схему каскадного соединения мультиплексоров. Мультиплексоры первой ступени (рис. 4.6, левая часть рисунка) выполняют предварительное мультиплексирование нескольких информационных каналов, мультиплексор второй ступени (рис. 4.6, правая часть рисунка) выполняет мультиплексирование конкретного информационного канала. При этом на его адресные входы поступают старшие разряды адресной шины.

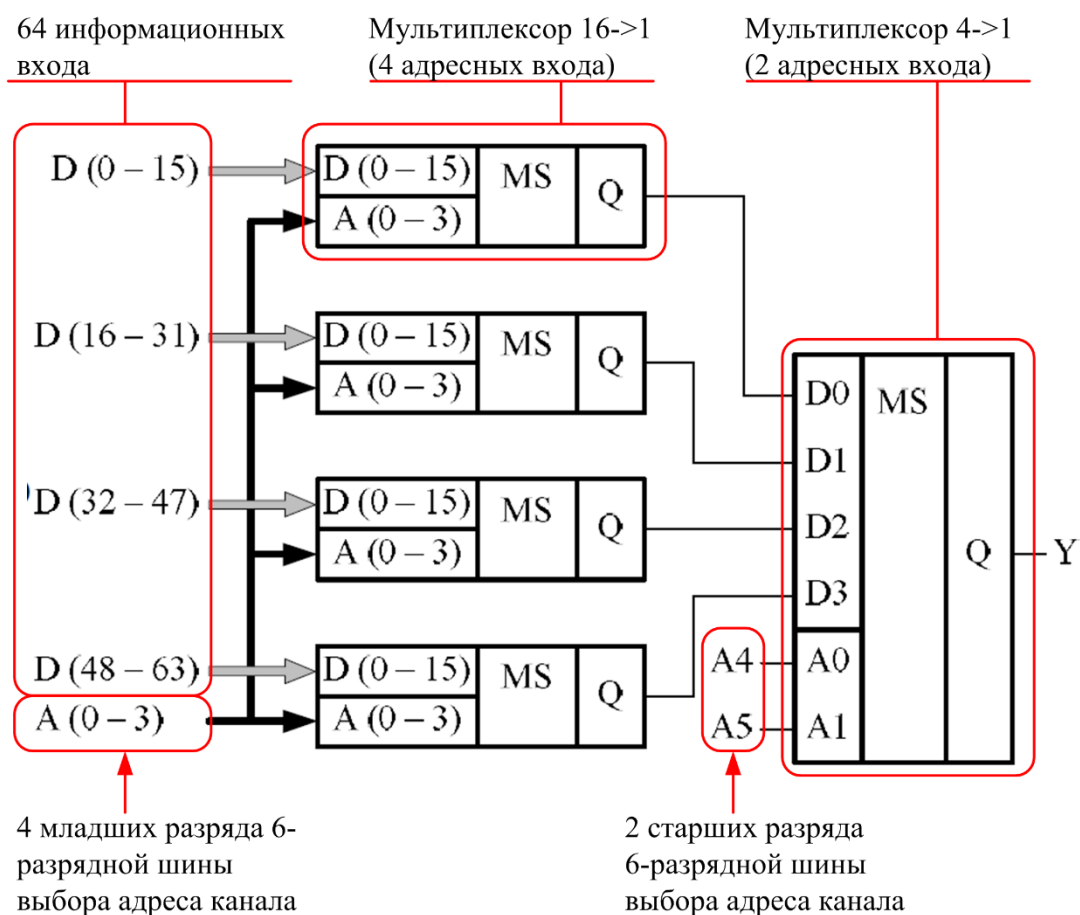


Рис. 4.6. Схема каскадного соединения мультиплексоров для увеличения количества мультиплексируемых информационных каналов

Одним из вариантов применения мультиплексоров является их использование для реализации логических функций. При этом на адресные входы мультиплексора поступают входные переменные, на информационные – логические константы (лог. «0», лог. «1») в соответствии с заданной таблицей истинности. Данный подход поясняет рис. 4.7.

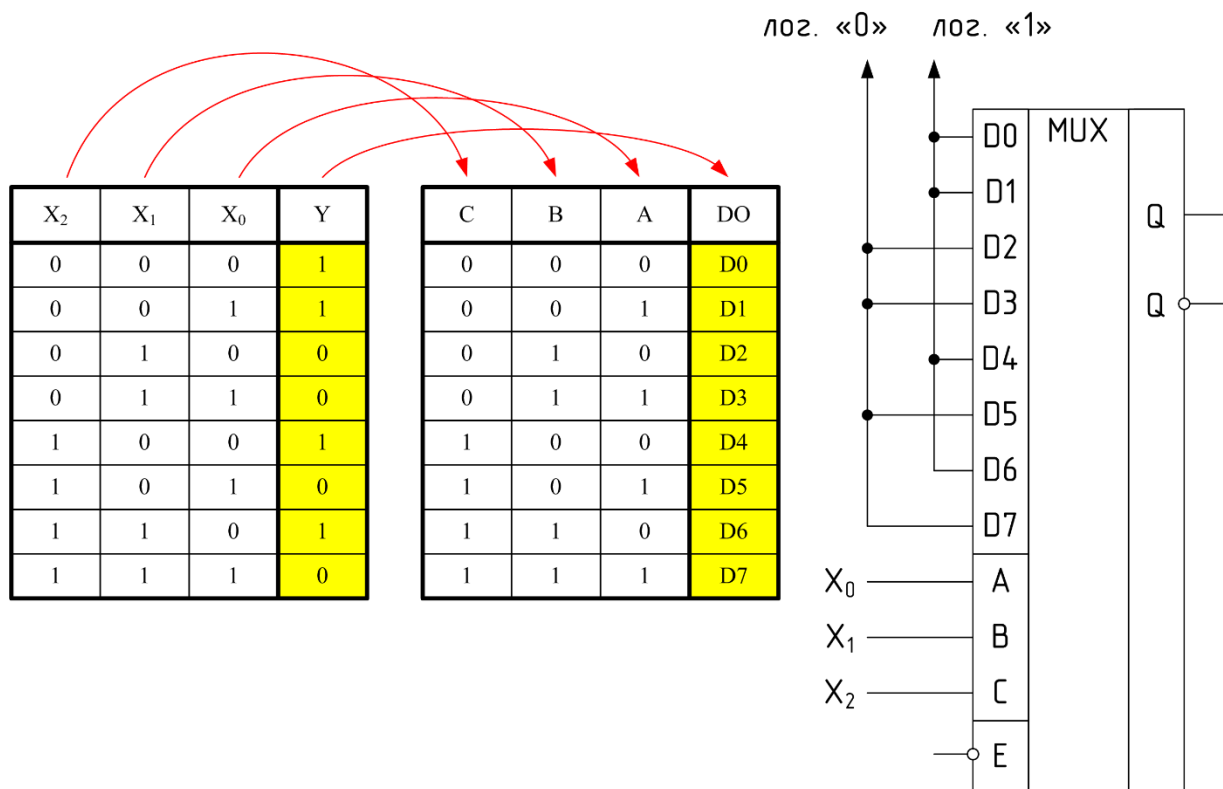


Рис. 4.7. Схема реализации логической функции трех входных переменных на мультиплексоре «8→1»

Задание на лабораторную работу и порядок ее выполнения

Общие условия и требования

1. Перед началом выполнения данной и последующих работ ознакомиться с порядком работы с лабораторным комплектом «ALT-DIG-002» и порядком создания схмотехнического проекта в среде проектирования ALTERAQUARTUS (см. приложение).

2. Ознакомиться с технической документацией на мультиплексоры 74xx151, 74xx153, 74xx157 (где xx – символы, относящиеся к технологии изготовления микросхем. Могут быть следующие сочетания: LS, HC, HCT, ALS).

3. Целью лабораторной работы является:

1) синтез схемы и исследование функционирования мультиплексора 8→1;

2) синтез схемы и исследование функционирования каскадного соединения мультиплексоров;

3) синтез схемы для реализации логических функций для управления свечением семисегментного индикатора с помощью мультиплексоров.

Задание 1 (порядок выполнения). Синтез схемы и исследование функционирования мультиплексора 8-→1.

1. В среде проектирования ALTERA QUARTUS подготовить проект с именем lab4a, содержащий заготовку схемы с 12 логическими входами и 2 логическими выходами. Задать следующие имена входов-выходов:

для входов: D0, D1, ..., D7, A0, A1, A2, EI.

для выходов: Y, W.

2. Выполнить этап назначения логических входов-выходов схемы к физическим выводам микросхемы программируемой логики (ПЛИС) в соответствии с разделом 3 приложения и таблицами соответствия выводов ПЛИС и устройств ввода и индикации (таблицы на рис.1.4). Рекомендуется следующее соответствие логических входов-выходов (рис. 1.5): для входных логических сигналов «A0», «A1», «A2» – двухпозиционные тумблеры «SW6»–«SW8», для входных логических сигналов «D0», «D0»,...,«D7» – двухпозиционные тумблеры «SW1»–«SW4» и кнопки «SB1»–«SB4», для входного логического сигнала «EI» – тумблер «SW5», для выходных логических сигналов «Y», «W» – светодиоды «VD2», «VD3».

3. Установить на схему мультиплексор 74151 (рис. 4.8), выполнить следующие подключения: логические сигналы «A0», «A1», «A2» – на входы «A», «B», «C» мультиплексора, логические сигналы «D0»,...,«D7», «EI» – на соответствующие входы мультиплексора, логические сигналы «Y», «W» – на соответствующие выходы мультиплексора.

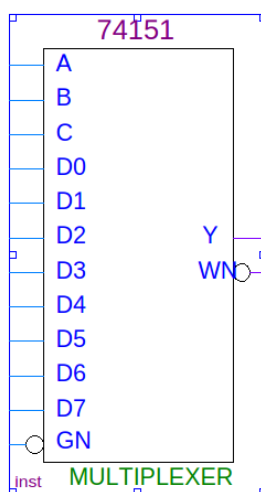


Рис. 4.8. Схематическое обозначение мультиплексора 74xx151 в среде проектирования «ALTERAQUARTUS»

4. Подготовить таблицу истинности для описания функционирования мультиплексора в соответствии с шаблоном (таблица 4.1).

5. Выполнить компиляцию проекта, выполнить «программирование» микросхемы ПЛИС (раздел 3 приложения).

6. Задавая необходимые логические сигналы путем переключения тумблеров «SW1»–«SW8» и нажатия кнопок «SB1»–«SB4» визуально (по назначенным светодиодам) контролировать выходной двоичный код на выходах (светодиодах) «Y», «W». Заполнить таблицу истинности мультиплексора для каждого из двух состояний информационного входа. Подтвердить факт, что на выход поступают данные только от выбранного информационного входа (изменение состояния других информационных входов не влияет на выходной сигнал мультиплексора).

Содержание отчета

1. Копии экрана среды разработки ALTERAQUARTUSc подготовленной схемой.
2. Фото состояния органов управления и индикации для нескольких входных сигналов.
3. Заполненная таблица истинности мультиплексора.
4. Выводы.

Задание 2 (порядок выполнения). Синтез схемы и исследование функционирования каскадного соединения мультиплексоров.

1. Используя результаты выполнения задания 1 данной лабораторной работы разработать схему мультиплексирования 32 логических сигналов используя однотипные мультиплексор 74xx151 («8→1»).
2. Учитывая состав и количество органов управления модуля «ALT-DIG-002» разработать схему подключения информационных и адресных входов для частичного проведения испытаний разработанной схемы.
3. Выполнить этапы формирования схемы в среде проектирования ALTERAQUARTUS и проверку ее функционирования.

Содержание отчета

1. Копии экрана среды разработки ALTERAQUARTUSc подготовленной схемой.
2. Фото состояния органов управления и индикации для нескольких входных сигналов.
3. Выводы.

Задание 3 (порядок выполнения). Синтез схемы для реализации логических функций для управления свечением семисегментного индикатора с помощью мультиплексоров.

1. Используя результаты, полученные в ходе выполнения лабораторной работы № 2 (таблицы истинности для сегментов семисегментного индикатора) и способ реализации логических функций на мультиплексорах

(рис. 4.7), разработать схему управления семисегментным индикатором (используя мультиплексоры 74xx151 («8→1»)).

2. Учитывая состав и количество органов управления модуля «ALT-DIG-002» разработать схему подключения информационных и адресных входов для испытаний разработанной схемы.

3. Выполнить этапы формирования схемы в среде проектирования ALTERAQUARTUS и проверку ее функционирования.

Содержание отчета

1. Копии экрана среды разработки ALTERAQUARTUS с подготовленной схемой.

2. Фото состояния органов управления и индикации для всех входных сигналов.

3. Выводы.

Контрольные вопросы

1. Каково назначение и структурная схема мультиплексора?

2. Каким должно быть соотношение между количеством информационных и адресных входов в интегральной схеме мультиплексора.

3. Может ли схема мультиплексора иметь несколько выходов?

4. В каких случаях объединяют схемы в мультиплексное дерево?

5. В чем заключаются различия схем мультиплексора и демультиплексора?

6. Синтезируйте схему воспроизведения функции $F = \bar{X}_1 \vee X_2 \vee \bar{X}_3 \vee X_4$ с помощью следующих микросхем:

а) мультиплексора «16→1»;

б) мультиплексора «8→1» и логических элементов «ИЛИ-НЕ»;

в) мультиплексора «4→1» и элементов «ИЛИ-НЕ»;

ПРИЛОЖЕНИЕ

ОПИСАНИЕ КОМПЛЕКТА ЛАБОРАТОРНОГО ОБОРУДОВАНИЯ «ALT-DIG-002»

1. Состав комплекта лабораторного оборудования

Комплект лабораторного оборудования ALT-DIG-002 «Основы цифровой схемотехники» предназначен для выполнения студентами комплекса практических и лабораторных работ по дисциплинам «Цифровая схемотехника», «Микропроцессоры и микропроцессорные системы» и смежных с ними, для приобретения устойчивых навыков работы с современными техническими средствами реализации цифрового оборудования. В состав рабочего места (комплекта) входят:

1.1. Отладочный модуль «ALT-DIG-002» с установленными на нем микросхемой программируемой логики и средствами пользовательского ввода-вывода (рис. П.1);

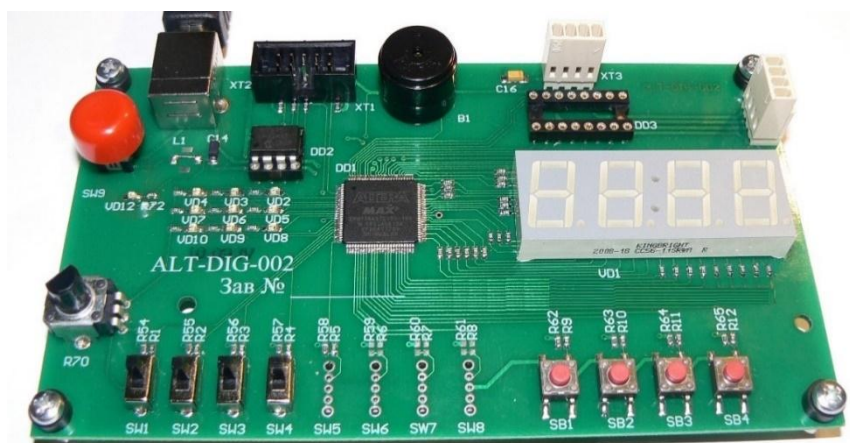


Рис. П.1. Отладочный модуль «ALT-DIG-002»

1.2. Программатор «LDM-PB» («ByteBlaster MV») для передачи модулю «ALT-DIG-002» исполнительной программы путем подключения к порту параллельного порта (LPT) персонального компьютера (рис. П.2);



Рис. П.2. Программатор «LDM-PB» («ByteBlaster MV»)

1.3. Программатор «USB–Blaster» для передачи модулю «ALT–DIG–002» исполнительной программы путем подключения к порту USB персонального компьютера (рис. П.3);



Рис. П.3. Программатор «USB–Blaster»

1.4. USB-кабель «Вилка А– вилка В» длиной 3 м для питания модуля «ALT–DIG–002» (рис. П.4);



Рис. П.4. USB-кабель «Вилка А– Вилка В»

1.5. Установленное на персональный компьютер программное обеспечение разработки и отладки цифровых устройства ALTERA QUARTUS 13.0 либо иной, поддерживающей микросхемы ПЛИС серии ALTERA 7000S;

1.6. Техническое описание и руководство по эксплуатации в электронном (печатном) виде.

Таблица П.1

Технические характеристики модуля «ALT–DIG–002»

Параметр	Описание (значение) параметра
Микросхема программируемой логики (ПЛИС)	ALTERA EPM7064STC-10 1000 логических вентилей, 64 макроячейки, 87 пользовательских выводов с логическими уровнями КМОП +5В/0В
Источники входных логических сигналов	8 переключателей; 4 тактовые кнопки

Параметр	Описание (значение) параметра
Приемники выходных логических сигналов	7-сегментный светодиодный индикатор на 4 символа; 9 единичных SMD-светодиодов типоразмера 0603 (3 красного цвета свечения, 3 желтого цвета свечения, 3 зеленого цвета свечения)
Микросхемы сопряжения с внешними устройствами	мостовой драйвер шагового двигателя L293D
Интерфейсы программирования	JTAG-интерфейс, используя поставляемый в комплекте программатор ByteBlaster
Напряжение электропитания	+5В от интерфейса USB либо от внешнего источника питания (ток потребления до 200 мА)

Расположение органов управления и индикации на модуле «ALT-DIG-002» (рис. П.5):

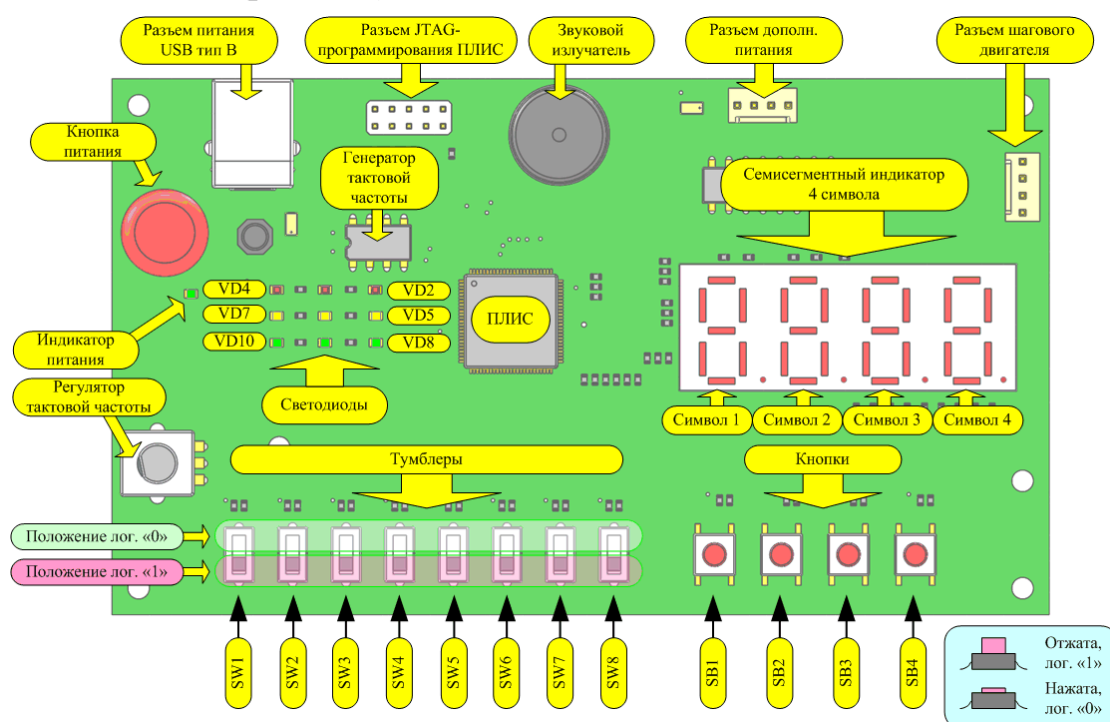


Рис. П.5. Расположение органов управления и индикации на модуле «ALT-DIG-002»

Программаторы «LDM-PB» («ByteBlasterMV») и «USB-Blaster» выполнены в виде автономных устройств и предназначены для передачи исполнительной программы от ПК к модулю «ALT-DIG-002». Для передачи

данных из среды разработки в микросхему программируемой логики используется один из программаторов, в зависимости от наличия интерфейса параллельного порта LPT или интерфейса USB на персональном компьютере. Выходной интерфейс программаторов – стандартный для ПЛИС интерфейс отладки и программирования JTAG (10 контактный разъем вилка/розетка IDC–10).

2. Порядок работы с комплектом (подготовка к работе, завершение работы)

2.1. Перед выполнением любых подключений к модулю «ALT–DIG–002» убедиться, что кнопка питания НЕ НАЖАТА (при этом питание на модуль при подключении кабеля питания подаваться не будет).

2.2. Выполнить подключение USB-кабеля, входящего в состав рабочего места, к персональному компьютеру.

2.3. Выполнить подключение ответной части USB-кабеля к модулю «ALT–DIG–002» (рис. П.6).

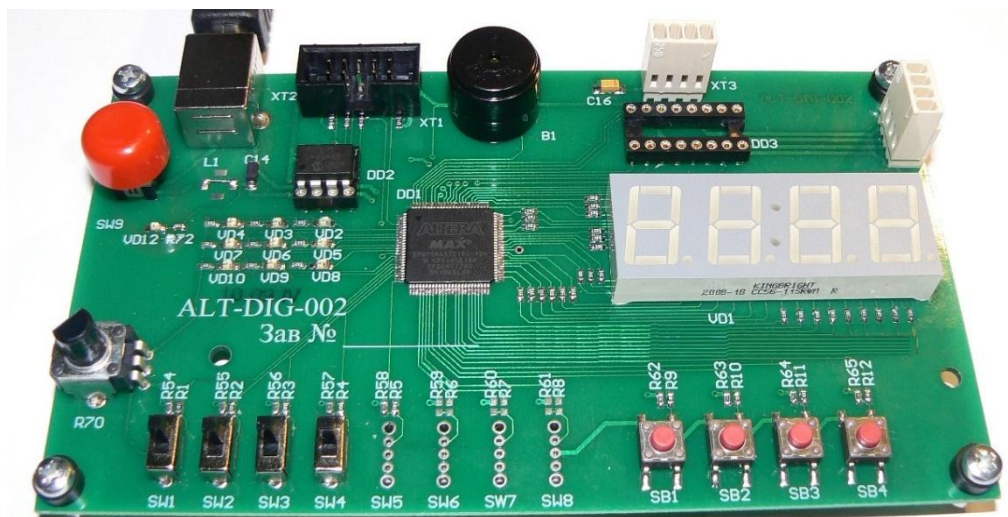


Рис. П.6. Подключение ответной части USB-кабеля питания к модулю «ALT–DIG–002»

2.4. Выполнить подключение программатора к USB-порту (либо LPT-порту) персонального компьютера.

2.5. Выполнить подключение JTAG-разъема (розетка IDC–10) к модулю «ALT–DIG–002» (рис. П.7).

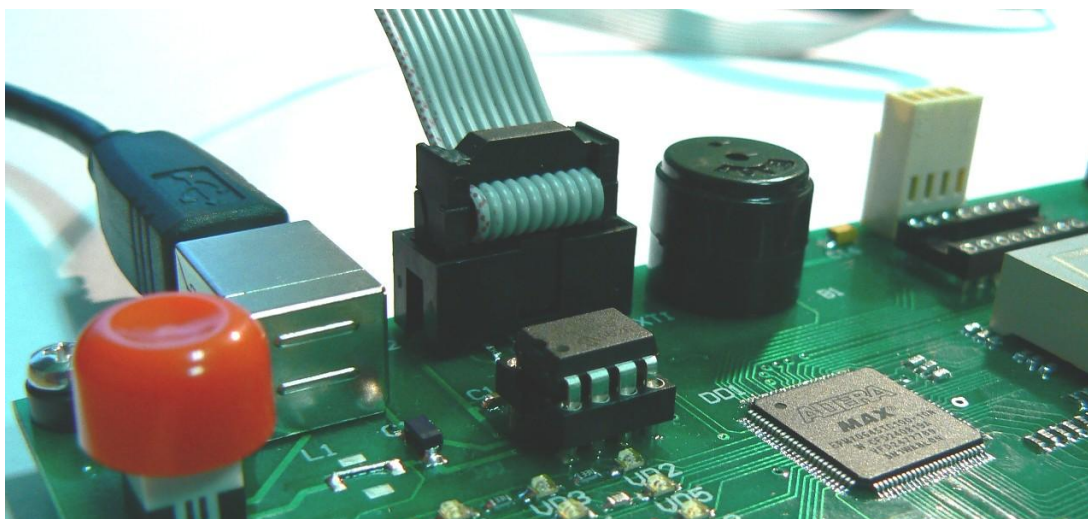


Рис. П.7. Подключение ответной части кабеля программатора (розетки IDC-10) к модулю «ALT-DIG-002»

2.6. Нажать кнопку включения питания, при этом должен загореться индикатор подачи питания и сконфигурированная ПЛИС начинает функционировать в соответствии со сформированной (ранее загруженной) пользователем логической схемой (логическими функциями) (рис.П.8).

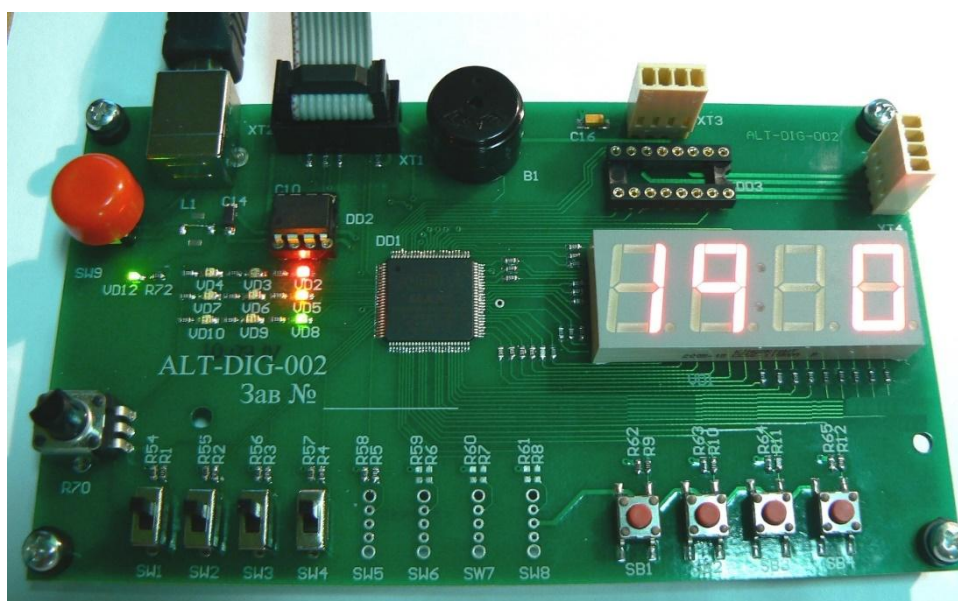


Рис. П.8. Пример работы модуля «ALT-DIG-002», функционирующего в соответствии с заданными логическими функциями

2.7. После включения модуля возможен запуск среды проектирования ALTERA QUARTUS, разработка, отладка и загрузка новой логической схемы в модуль (см. раздел 3 «Работа в среде проектирования ALTERA QUARTUS»).

2.8. После окончания работы с модулем «ALT-DIG-002» отключение выполняется в следующей последовательности (рис. П.9):

- а) отключаем кнопку питания, по индикатору убеждаемся что питания снято;
- б) отключаем 10-контактный разъем программирования;
- в) отключаем USB-кабель питания от модуля;
- г) отключаем программатор и USB-кабель от персонального компьютера;
- д) укладываем модуль, кабели в заводскую упаковку.

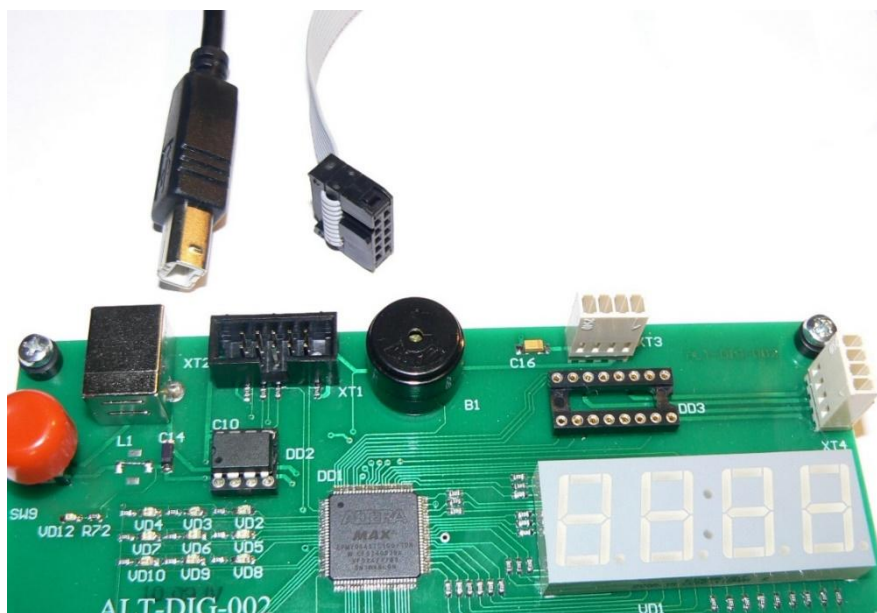


Рис. П.9. Модуль «ALT-DIG-002» после окончания работы (отключены кабели питания и программирования)

3. Работа в среде проектирования «ALTERAQUARTUS» (на примере работы в версии 13.0 для операционной системы Linux Ubuntu)

3.1. Запуск среды проектирования «ALTERA QUARTUS» из графической среды операционной системы Linux выполняется кликом ярлыка «QUARTUSII (64-bit) WebEdition», расположенного на «Рабочем столе» либо в основном меню списка приложений операционной системы (рис. П.10).

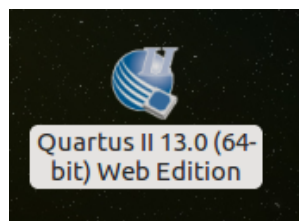


Рис. П.10. Ярлык запуска среды проектирования «ALTERAQUARTUS», расположенный на «рабочем столе» графической подсистемы операционной системы LINUX

3.2. После осуществления загрузки среды проектирования основное ее окно выглядит следующим образом (рис. П.11):



Рис. П.11. Основное окно среды проектирования «ALTERAQUARTUS», после запуска

3.3. Работа в среде проектирования начинается с создания проекта (иерархической структуры, включающей исходные файлы, файлы конфигурации устройства, файлы назначения логических входов-выходов к выводам ПЛИС и т.п.). Создание нового проекта выполняется путем нажатия кнопки «CreateNewProject» или вызовом пункта меню «File – NewProjectWizard...» (мастер создания нового проекта) (рис. П.12).



a)

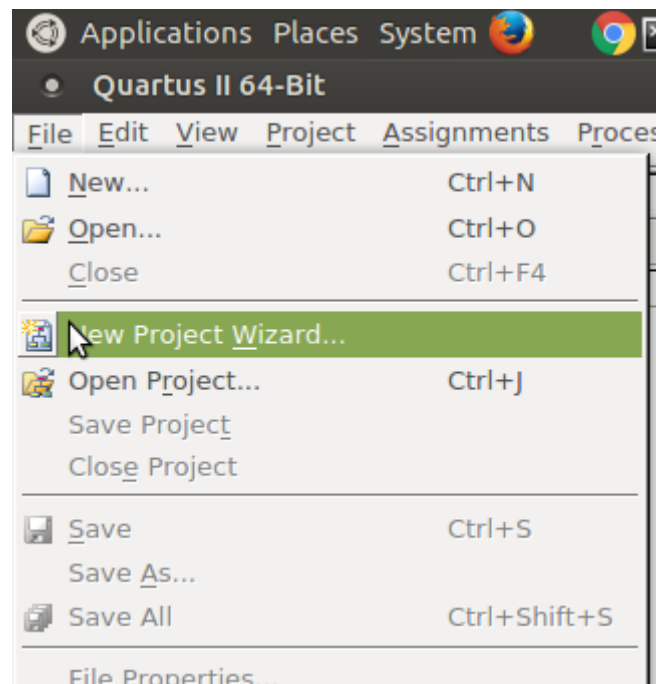
Start Designing

Designing with Quartus II software requires a project

**Create a New Project
(New Project Wizard)**

Open Existing Project

б)



в)

Рис. П.12. Варианты запуска мастера создания нового проекта

3.4. После вызова данного пункта меню формируется диалоговое окно пошагового создания проекта. На первом шаге система информирует о последовательности этапов создания проекта. Действие пользователя – нажатие кнопки диалогового окна «Next» («Далее») (рис. П.13).

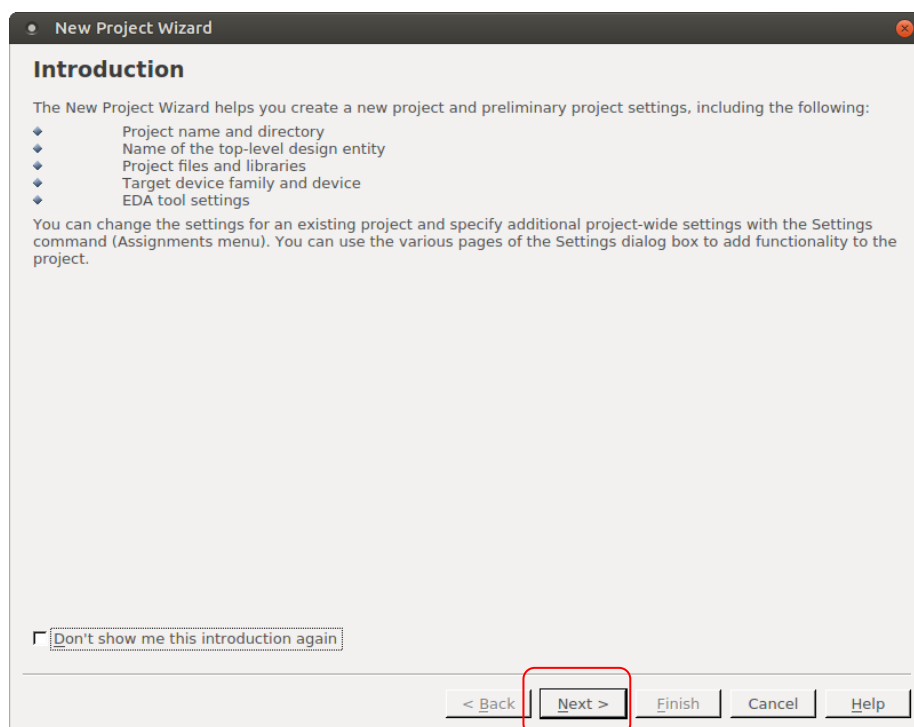


Рис. П.13. Стартовое окно мастера создания нового проекта

3.5. На следующем экране пользователь должен указать расположение папки с проектом в файловой системе персонального компьютера. В зависимости от операционной системы персонального компьютера возможен несколько различный внешний вид данного окна, а также последующих диалоговых окон выбора рабочей папки. Для операционной системы Ubuntu в рабочей папке пользователя (/home/<имя пользователя в системе>) рекомендуется создать отдельную папку «projects» и использовать ее как корневую при создании последующих проектов. Обязательны к исполнению следующие соглашения: а) использование в пути и имени проекта только латинских заглавных и строчных символов, цифр и символа подчеркивания; б) недопустимость использования пробела. Процесс указания пути к проекту и его имени продемонстрирован на рисунках П.14–П.22 для пользователя с именем «user1».



Рис. П.14. Окно указания пути к рабочей папке проекта и его имени

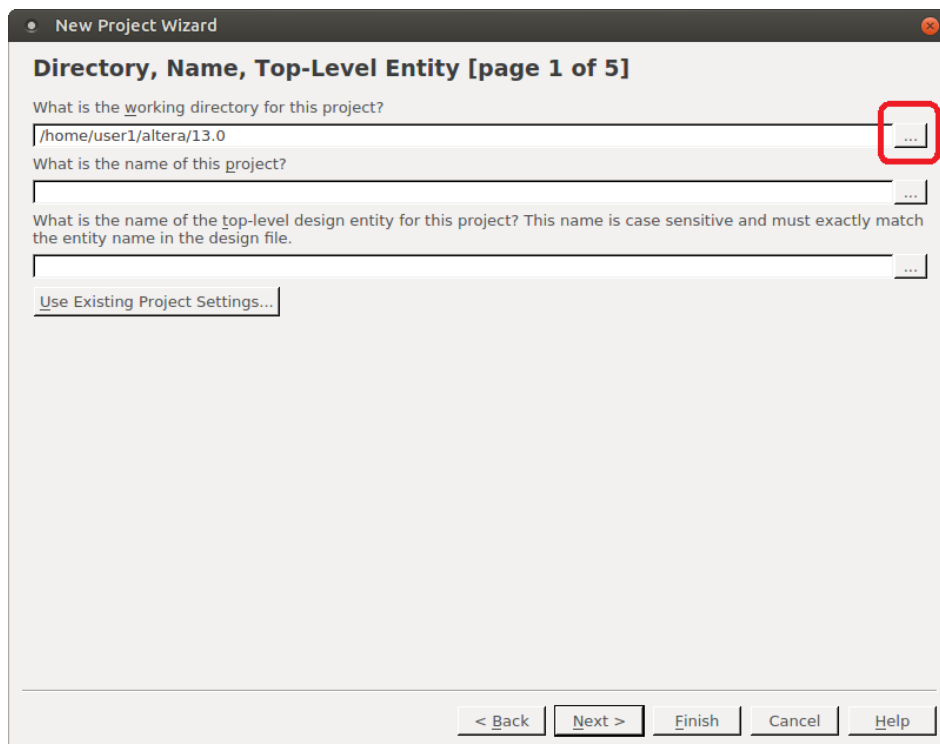


Рис. П.15. Запуск кнопки вызова диалогового окна
выбора/создания папки

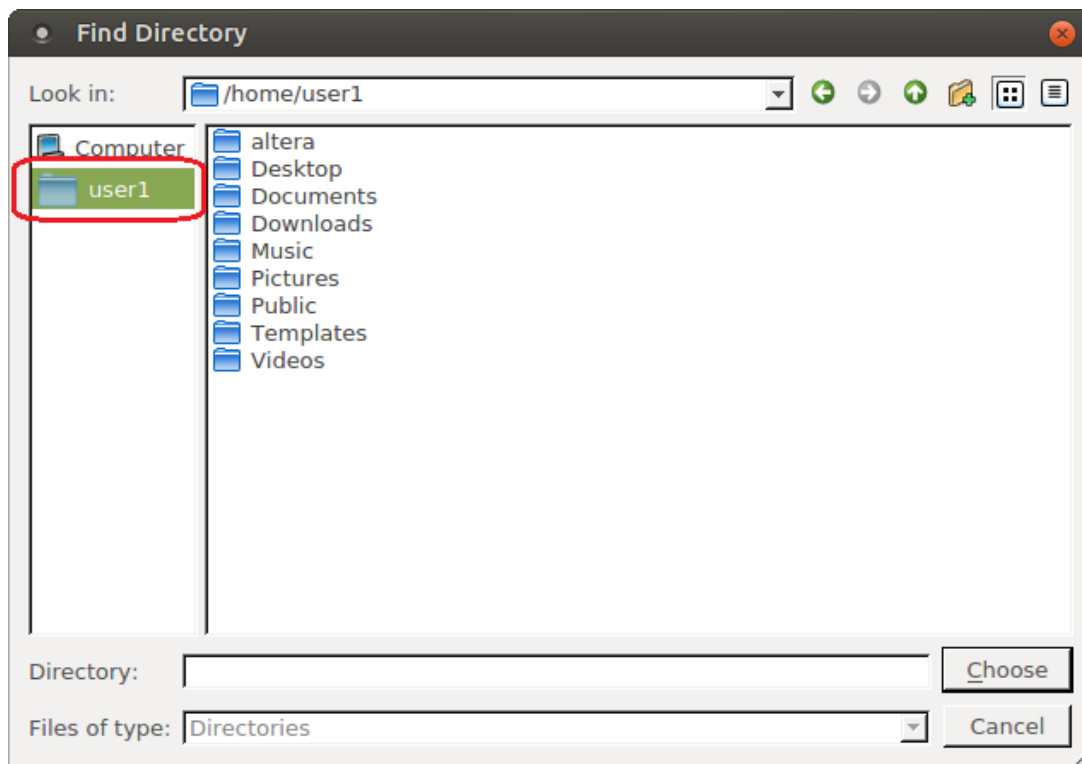


Рис. П.16. Выбор папки текущего пользователя

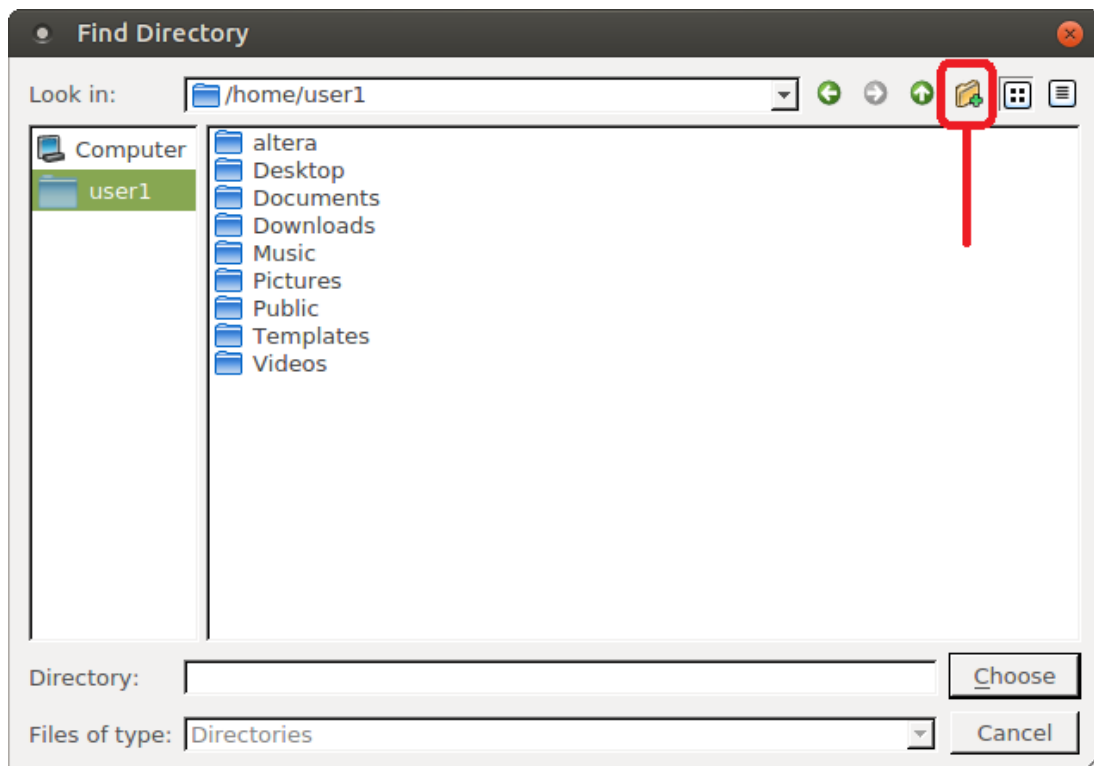


Рис. П.17. Кнопка создания новой папки

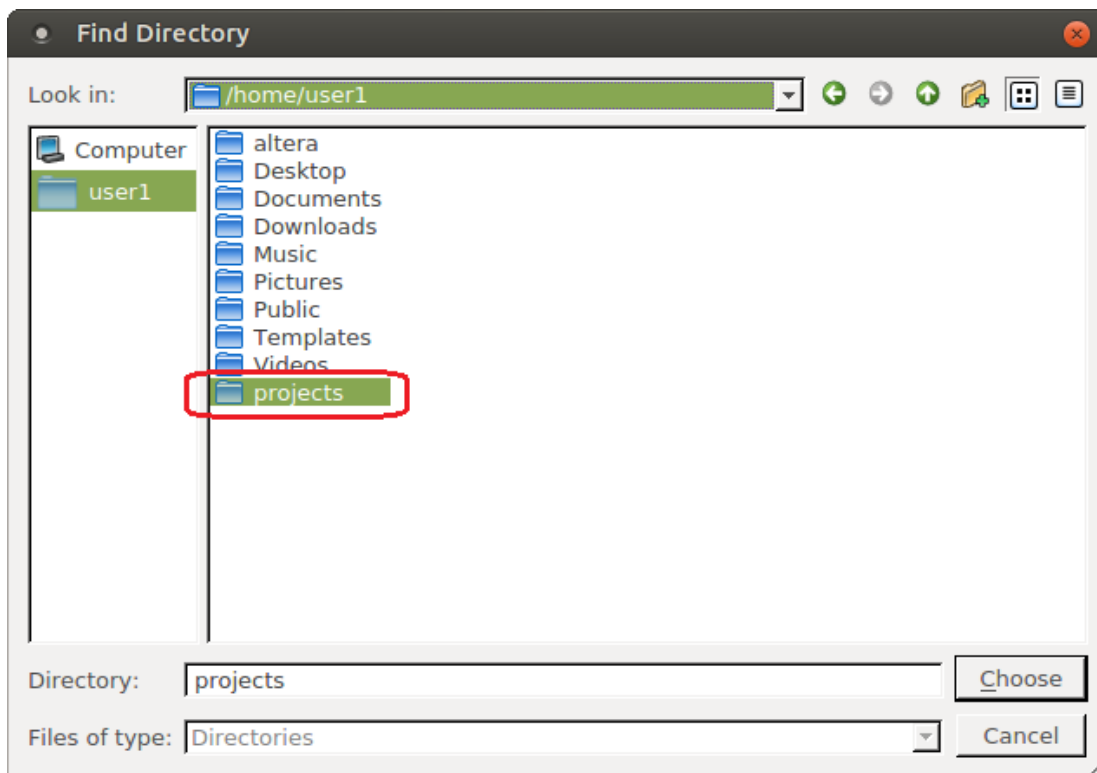


Рис. П.18. Создание папки «projects»

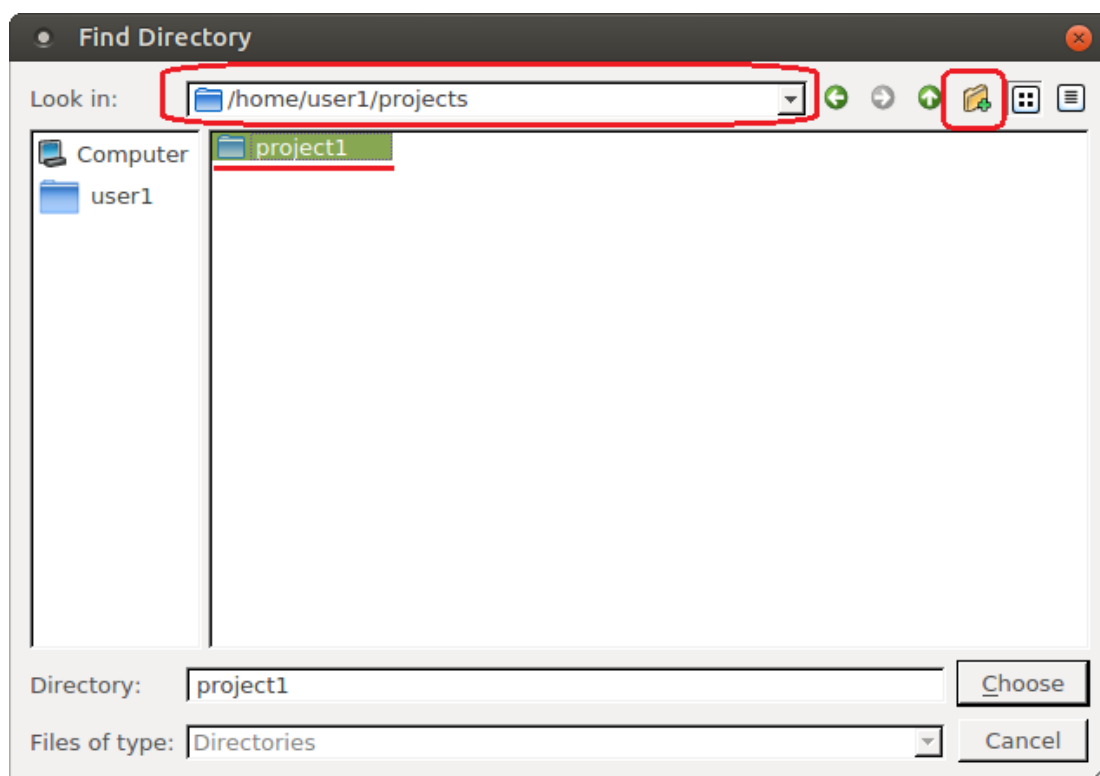


Рис. П.19. Вход в папку «projects» и создание подпапки «project1»

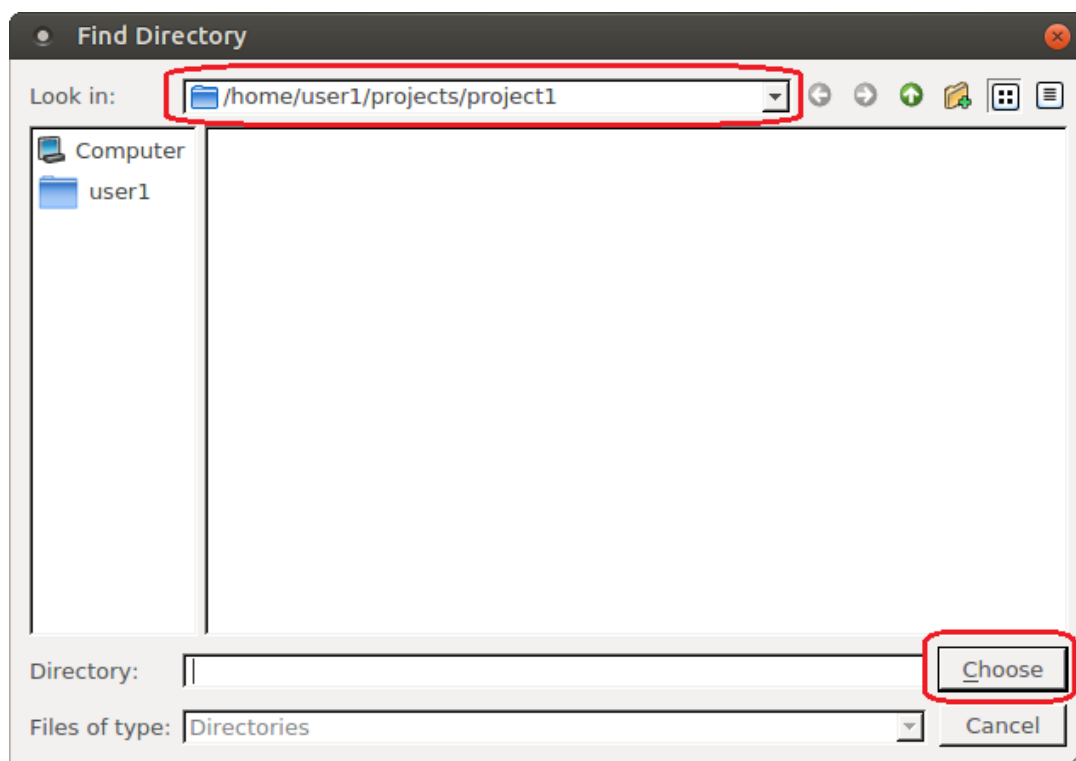


Рис. П.20. Вход в подпапку «project1» и ее выбор в качестве текущей рабочей папки проекта

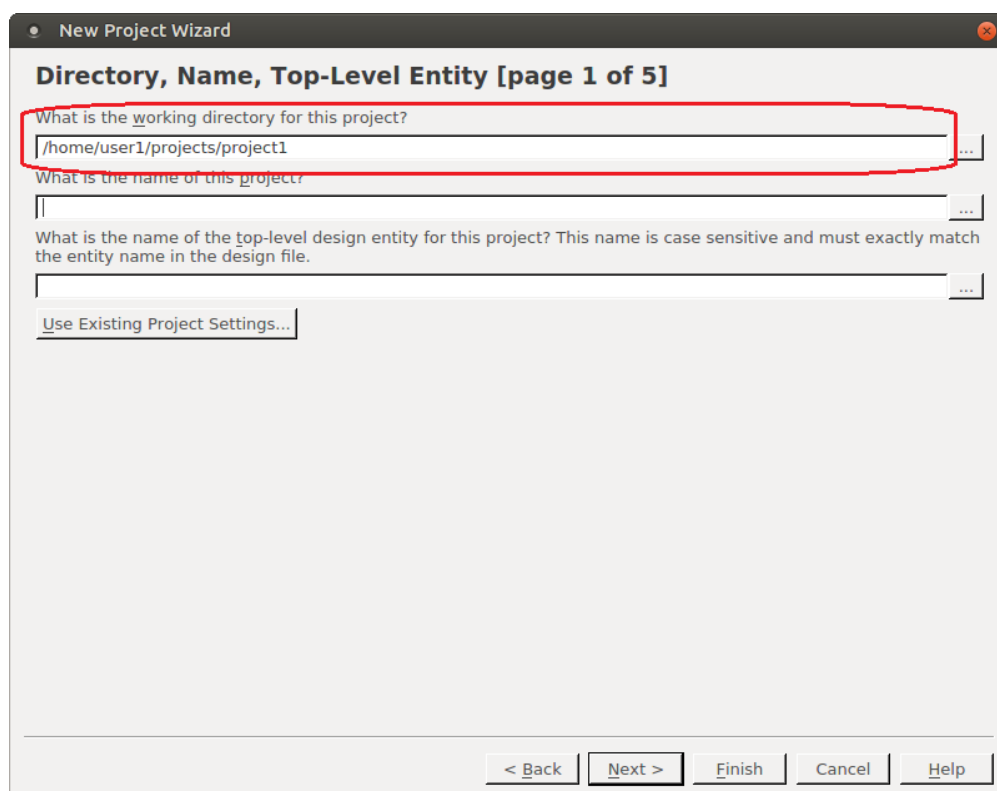


Рис. П.21. Выполнение проверки корректности пути к рабочей папке проекта

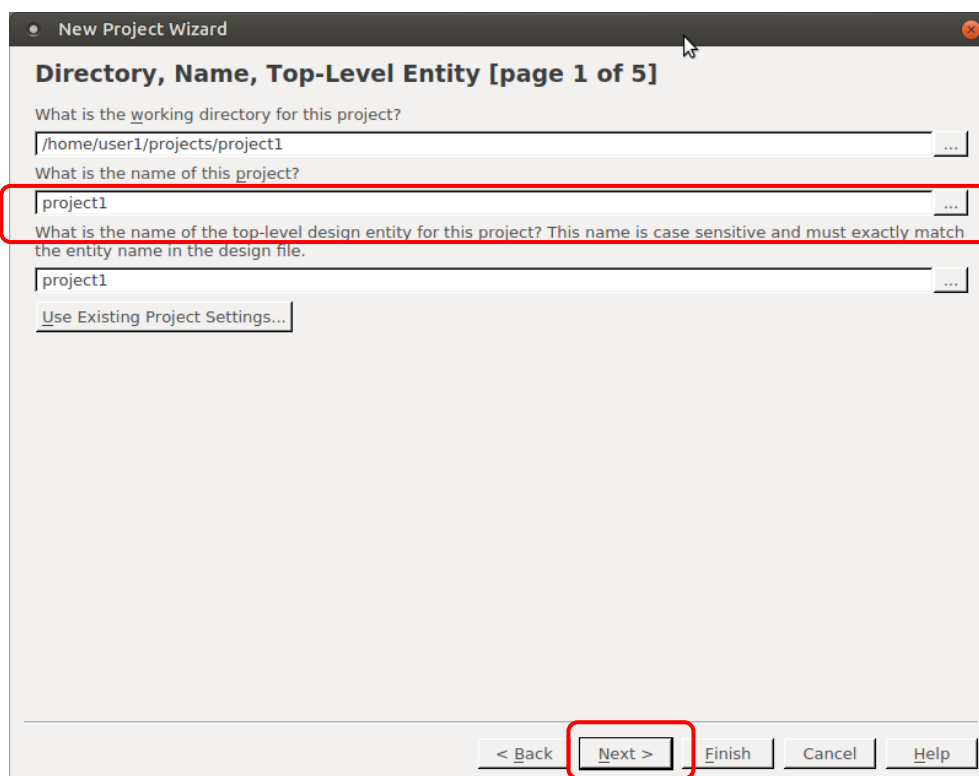


Рис. П.22. Ввод имени проекта (латиницей) и переход к следующему шагу мастера создания проекта

3.6. На следующем шаге предлагается подключить к проекту имеющиеся файлы исходных схем (или описание схемы на языках HDL). Поскольку на данном этапе файл (-ы) схемотехнического описания отсутствуют, данный шаг пропускается (нажатие кнопки «Next») (рис. П.23).

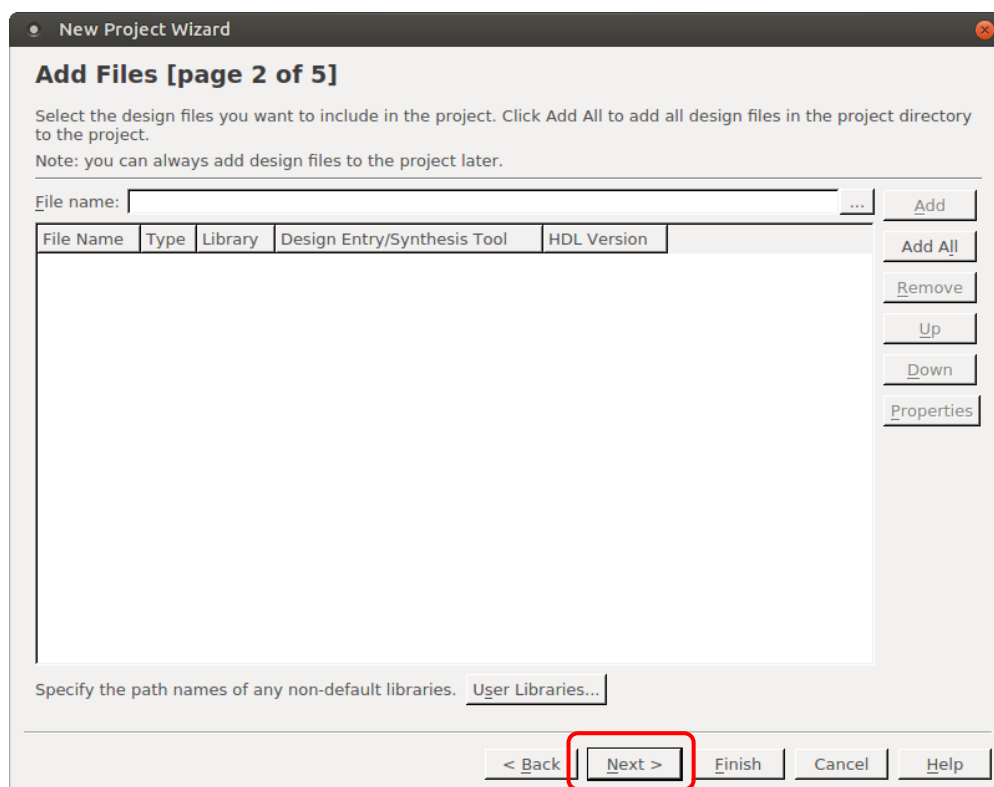


Рис. П.23. Этап подключения существующих
схемотехнических файлов к создаваемому проекту

3.7. На следующем шаге необходимо выбрать серию и тип микросхемы ПЛИС фирмы ALTERA, с которой предполагается работать. На модуле «ALT-DIG-002» установлена микросхема ПЛИС серии MAX7000S, тип микросхемы MAX7064STC100-10. Необходимо в выпадающем списке выбрать серию MAX7000S, в перечне микросхем (нижняя часть окна) – выбрать конкретный тип микросхемы. После выполнения выбора необходимо нажать клавишу «Next» (рис. П.24).

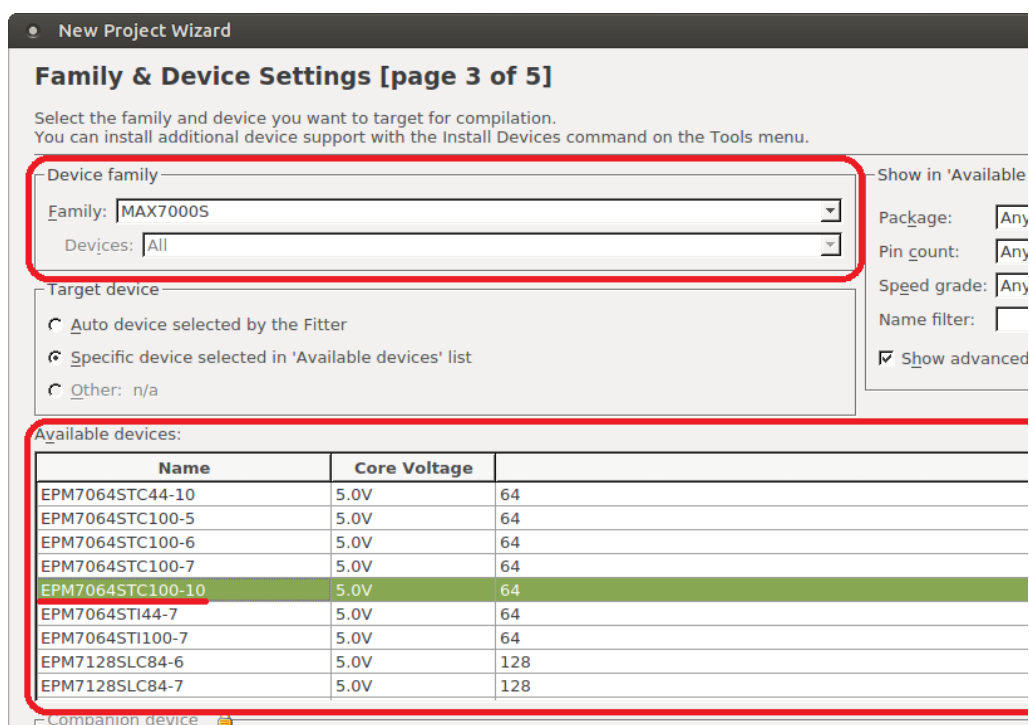


Рис. П.24. Выбор серии и типа микросхемы ПЛИС
(для модуля «ALT-DIG-002» микросхема серии MAX7000S,
модель EPM7064STC100-10)

3.8. Следующим шагом необходимо выбрать средства компиляции (синтеза) выходных файлов из входных исходных файлов, а также тип входной информации для компилятора (схемный ввод либо языка HDL). В данном окне (рис. П.25) в верхней строке в выпадающих списках необходимо выбрать пункты «DesignCompiler» и «EDIF» (рис. П.26).

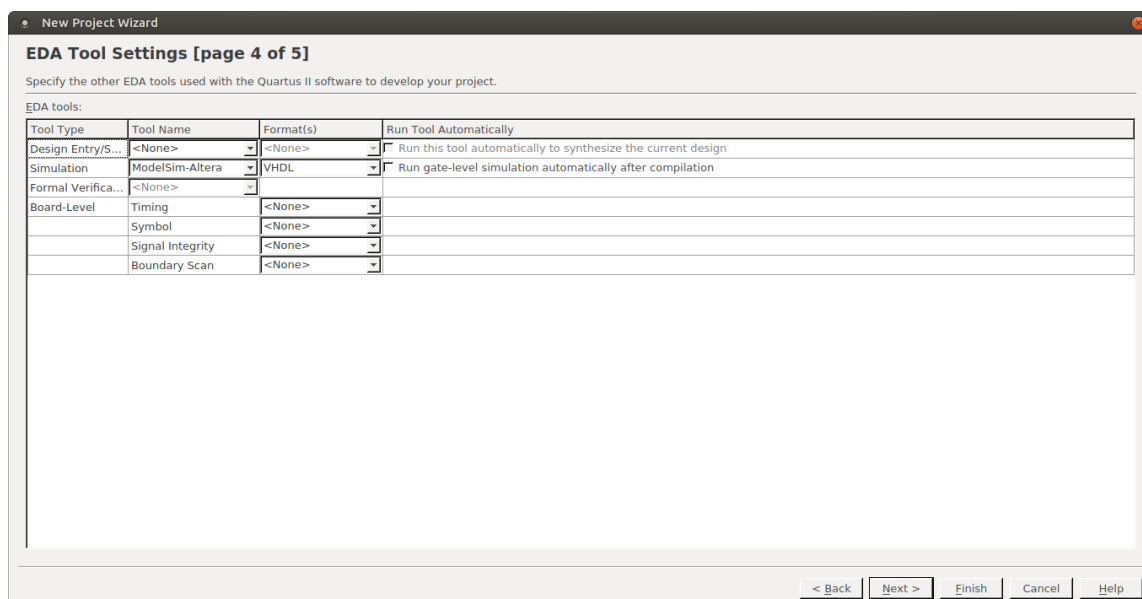


Рис. П.25. Окно выбора средств компиляции проекта

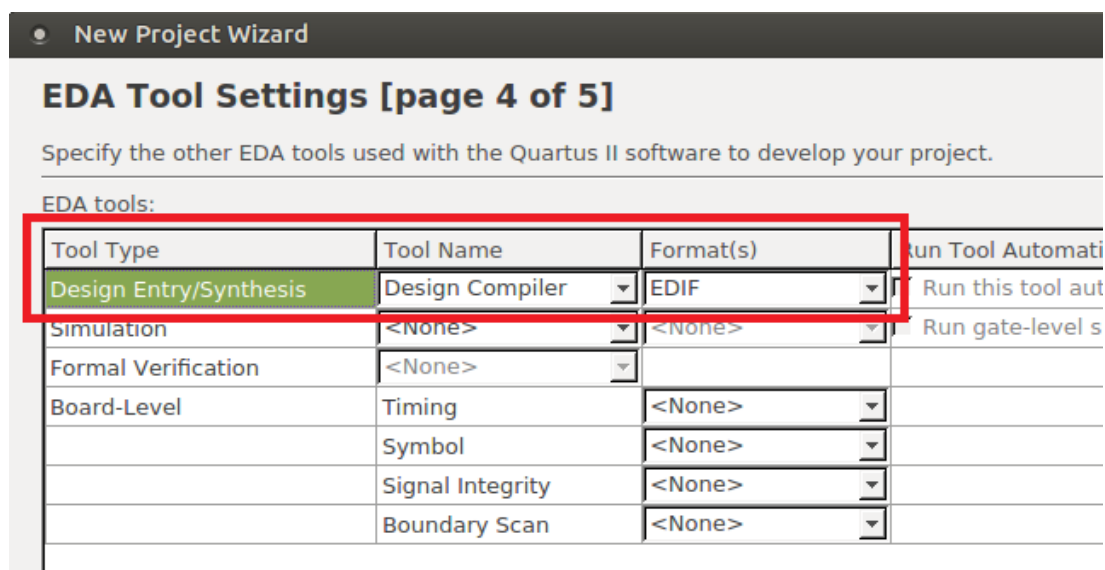


Рис. П.26. Окно выбора средств компиляции проекта

3.9. Последним шагом мастера создания проекта система выдает сводную информацию о созданном проекте. Завершаем создание проекта нажатием кнопки «Finish» (рис. П.27).

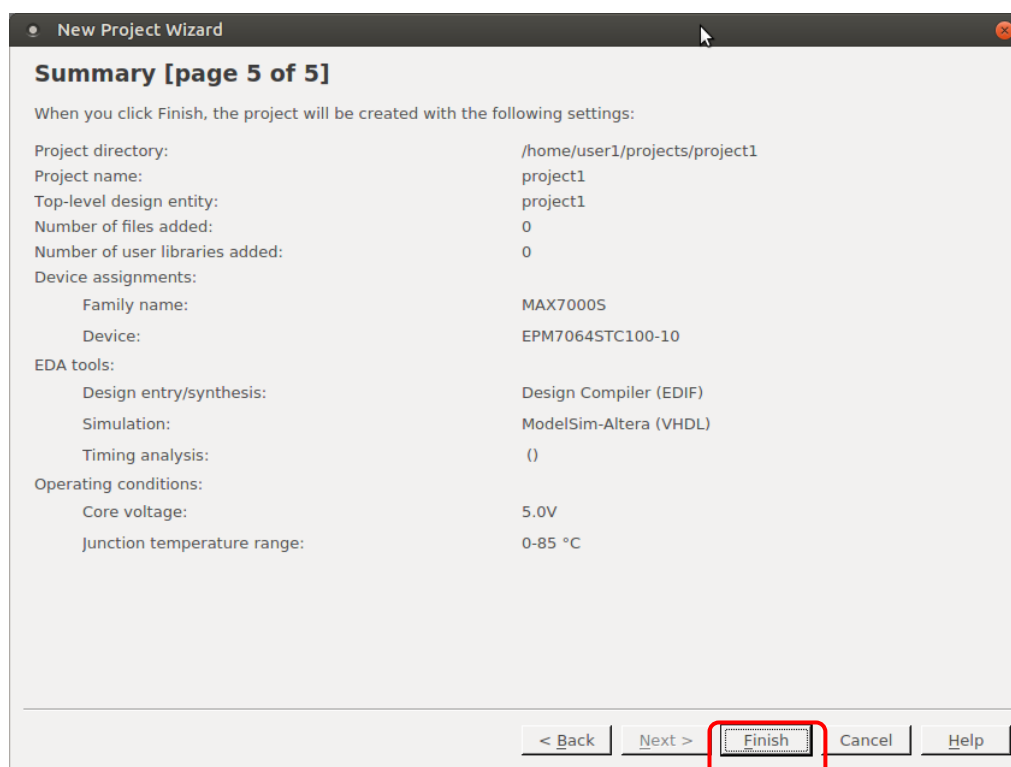


Рис. П.27. Окно выбора средств компиляции проекта

3.10. После создания проекта необходимо выполнить подключение к нему существующего файла исходной схемы, либо создать ее в процессе работы с проектом. Для этого требуется выполнить следующие действия: выбрать пункт меню «File»–«New...» (рис. П.28).

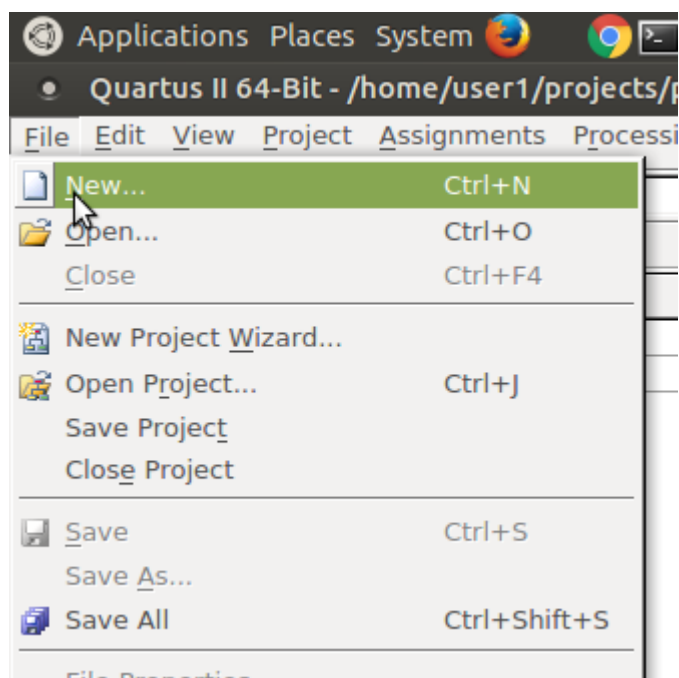


Рис. П.28. Выбор пункта меню создания нового файла схемы

3.11. В диалоговом окне необходимо выбрать пункт «BlockDiagram / SchematicFile» (рис. П.29).

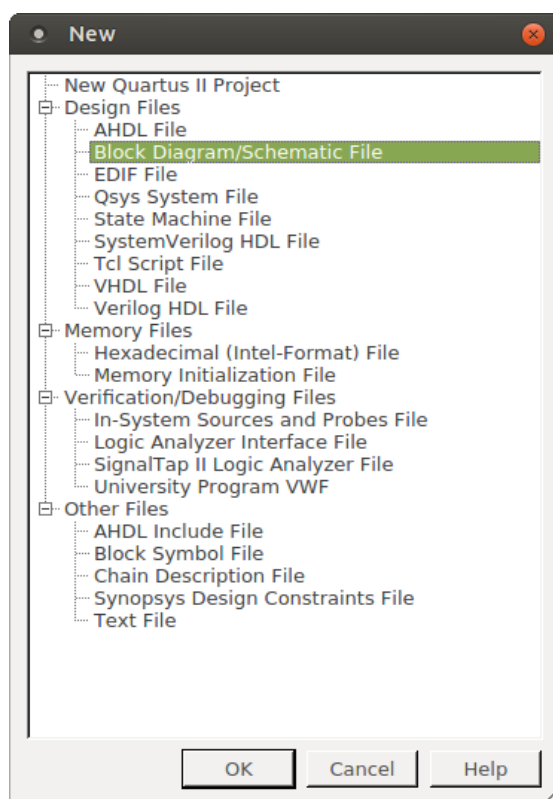


Рис. П.29. Окно выбора типа создаваемого файла

3.12. Основной вид окна среды разработки будет иметь следующий вид (рис. П.30).

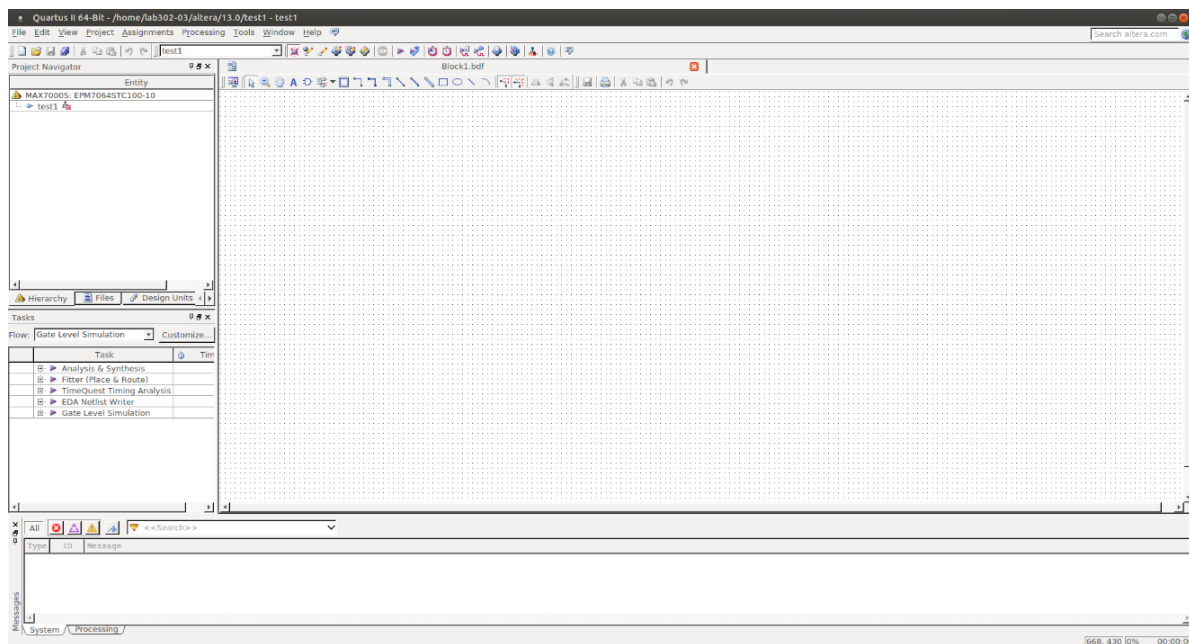


Рис. П.30. Основной вид окна среды разработки после создания нового схемотехнического файла

В верхней части окна расположена панель инструментов для выбора определенных элементов, позиционируемых на лист схемы (рис. П.31).

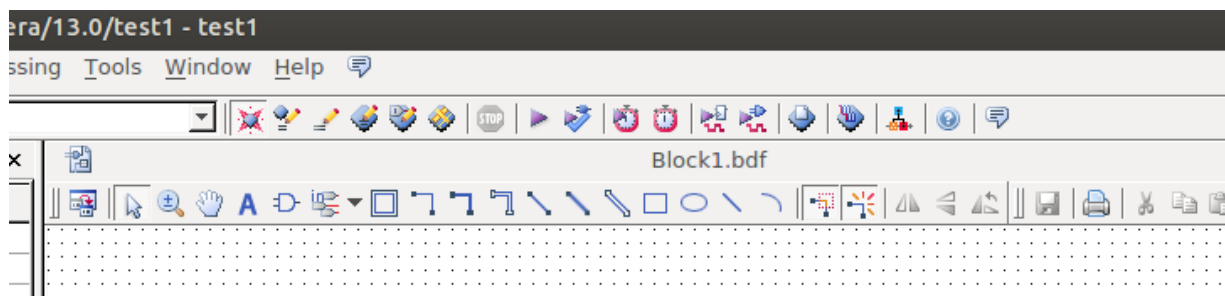


Рис. П.31. Панель инструментов основного окна среды разработки

3.13. Созданный файл будет подключен к проекту автоматически, если при его сохранении будет задано имя, идентичное имени проекта (рис. П.32). Файл необходимо сохранить в папку, указанную средой разработки автоматически (рис. П.33).

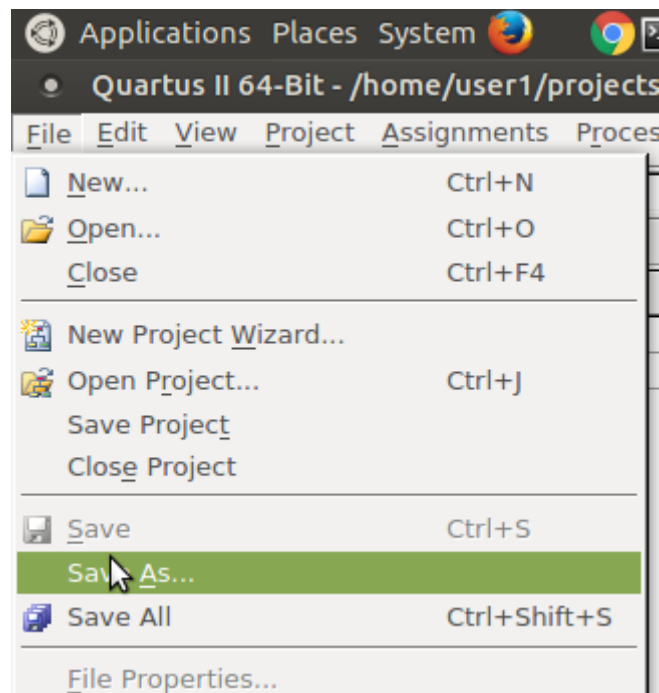


Рис. П.32. Вызов диалога сохранения схмотехнического файла

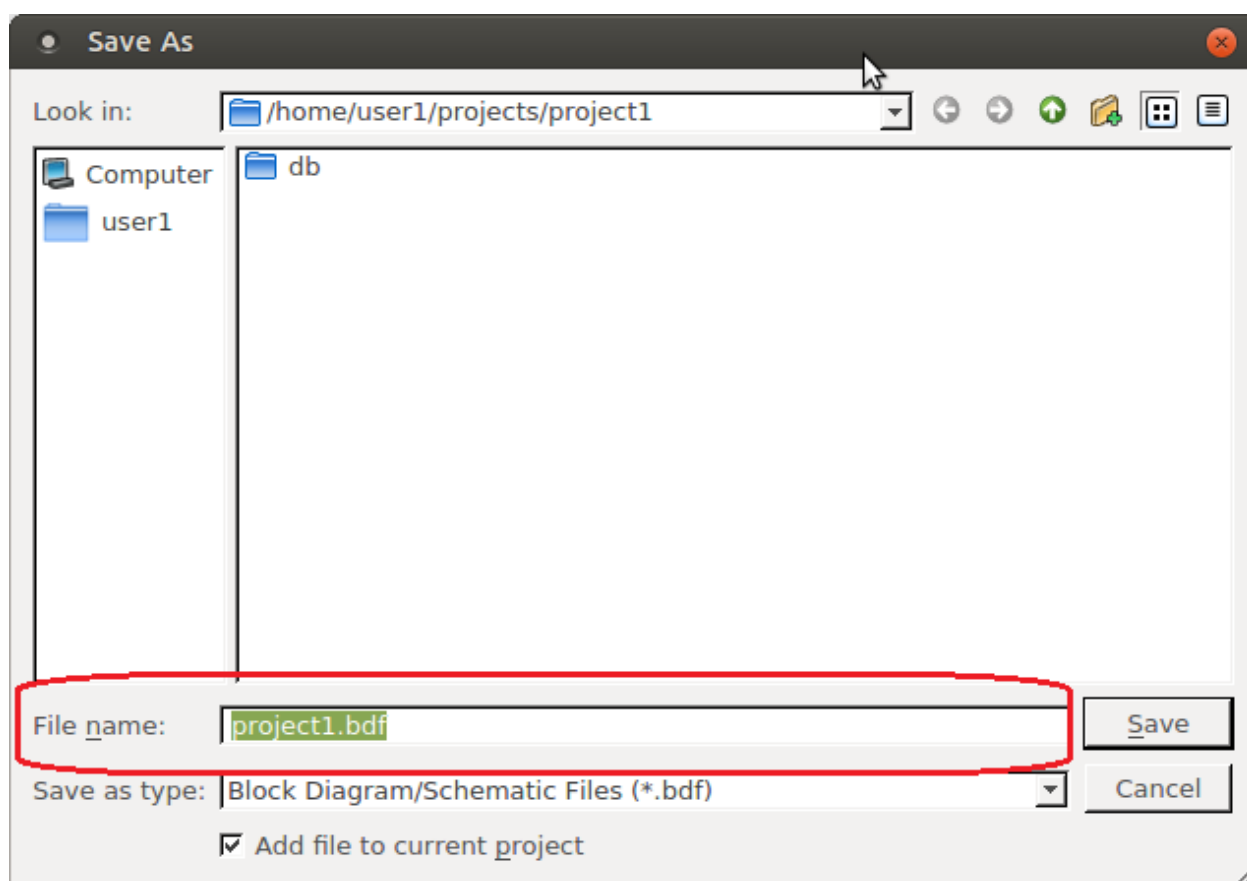


Рис. П.33. Диалоговое окно сохранения схмотехнического файла

3.14. Логическая схема устройства создается путем размещения на основном поле редактора логических элементов (базовых логических элементов/вентилей, логических блоков средней степени интеграции – шифраторов, преобразователей, счетчиков, регистров и т.д.), организации между ними соединений, добавления на схему логических входов и выходов.

3.15. Логические элементы размещаются в основном поле редактора с помощью инструмента «Symboltool» из панели инструментов (рис. П.34).

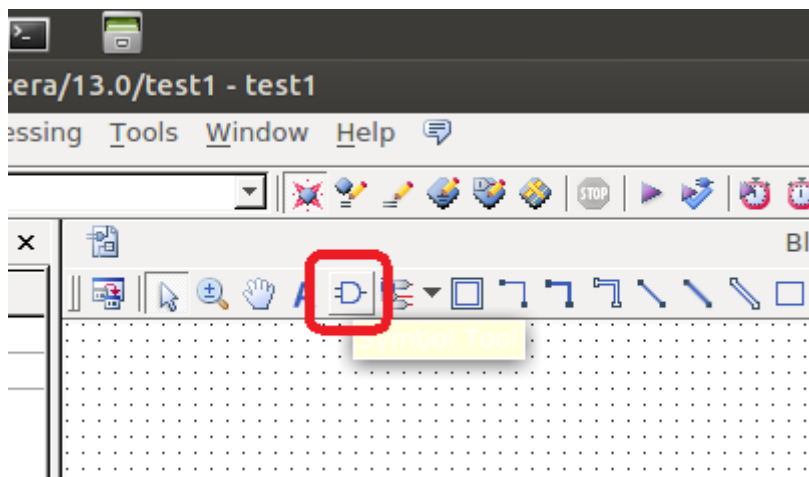


Рис. П.34. Вызов инструмента «Symboltool» из панели инструментов

3.16. После выбора инструмента «Symbol Tool» появляется диалоговое окно выбора логического блока (рис. П.35).

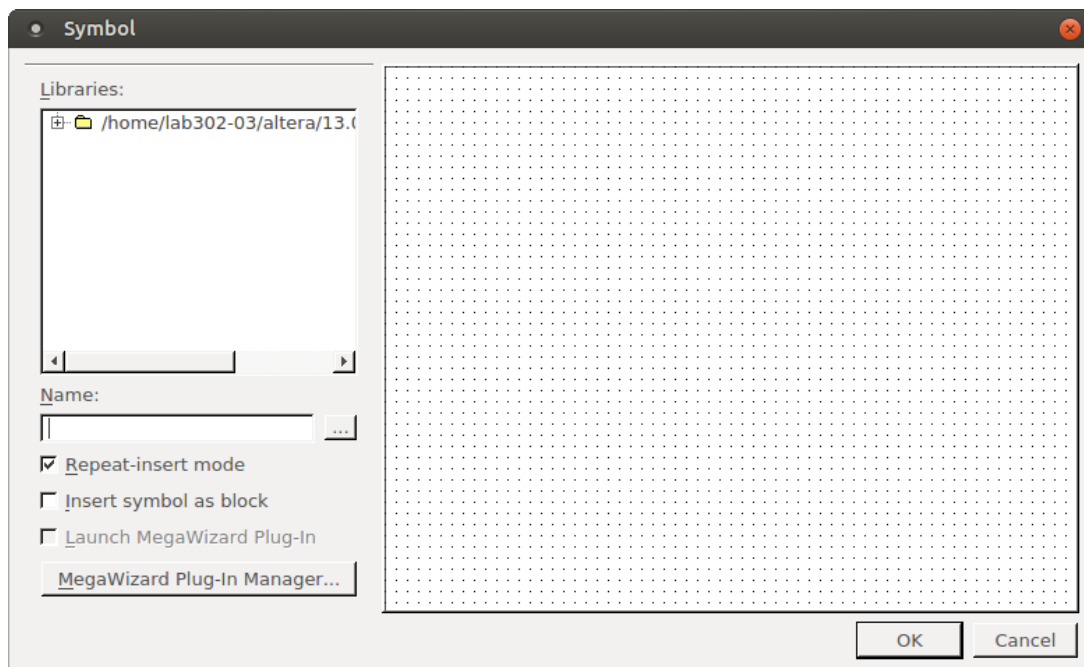


Рис. П.35. Диалоговое окно выбора логического блока

3.17. В левой части окна в раскрывающемся иерархическом списке выбираются элементы определенного типа. Дополнительно, в текстовом

поле ввода «Name» можно ввести имя или номер логического блока, система выполняет поиск и отображает его в правой части диалогового окна. После нажатия кнопки «ОК» элемент можно позиционировать в основном поле схемы (рис. П.36).

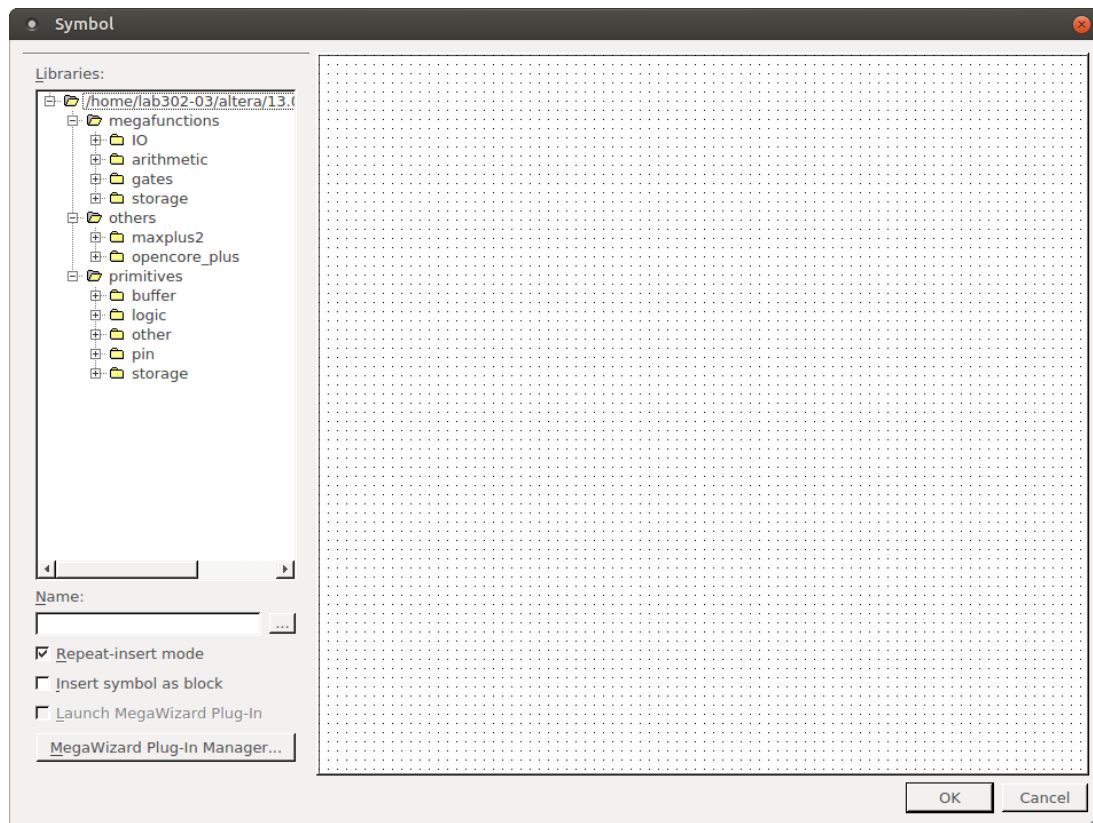


Рис. П.36. Древоподобный список групп логических элементов
(левая часть диалогового окна)

3.18. В следующих пунктах на примере повторителя логического сигнала (схемы, формирующей выходной логический сигнал равный входному) приведена последовательность создания схемы, этапы назначения соответствия логических и физических выводов, компиляции проекта, передачи файла конфигурации в ПЛИС и ее запуска.

3.19. Для организации входных и выходных сигналов используются элементы двух типов: «Вход» и «Выход». Для размещения на схеме элемента типа «Вход» («Input») в диалоговом окне выбора типа элемента последовательно раскрываем подтипы «primitives» и «pin». Далее выбираем элемент «input» («Вход») (рис. П.37).

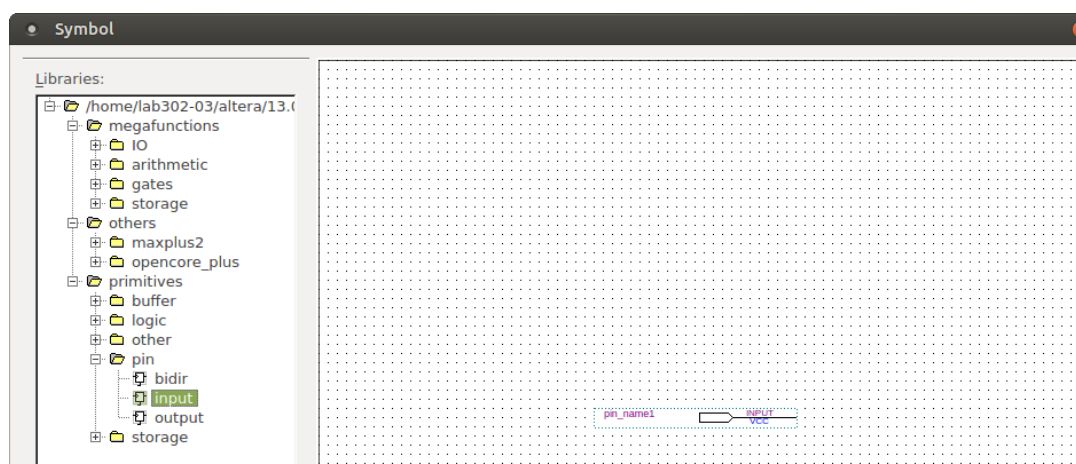


Рис. П.37. Выбор элемента типа «Input» («Вход»)

3.20. После выбора элемента в правой части диалогового окна появляется его условное графическое обозначение. Для размещения элемента на «листе» схемы необходимо нажать кнопку «ОК», мышью позиционируем изображение элемента в нужном месте схемы и нажимаем левую кнопку мыши (рис. П.38а). Дальнейшие манипуляции мышью приводят к размещению следующего элемента выбранного типа (таким образом можно разместить несколько однотипных элементов). Выход из режима размещения элементов выполняется двойным нажатием клавиши «Esc» либо выбором инструмента «Selection Tool» на панели инструментов (рис. П.38 б).

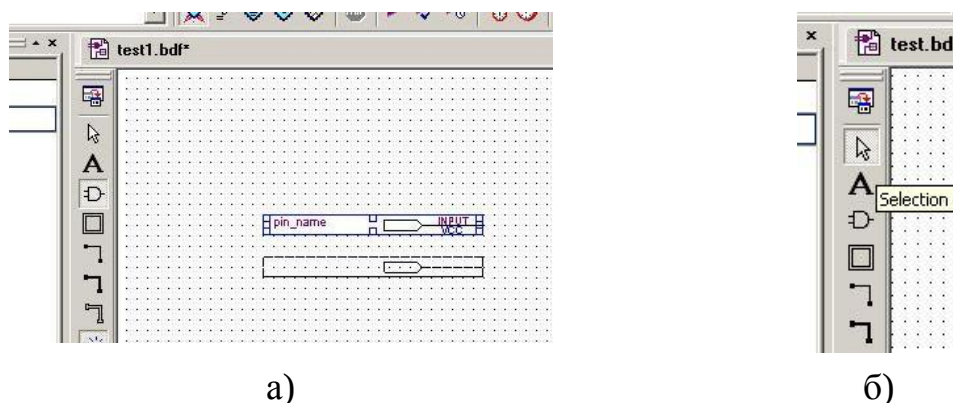


Рис. П.38. Последовательное размещение однотипных элементов (а), выбор инструмента «Selection Tool» на панели инструментов (б)

3.21. Аналогичным образом необходимо разместить в редакторе схемы элемент типа «output» («Выход»). Для этого в диалоговом окне выбора типа элемента необходимо выбрать данный элемент («primitives» – «pin» – «output»), нажать «ОК» и разместить элемент на схеме (рис. П.39).

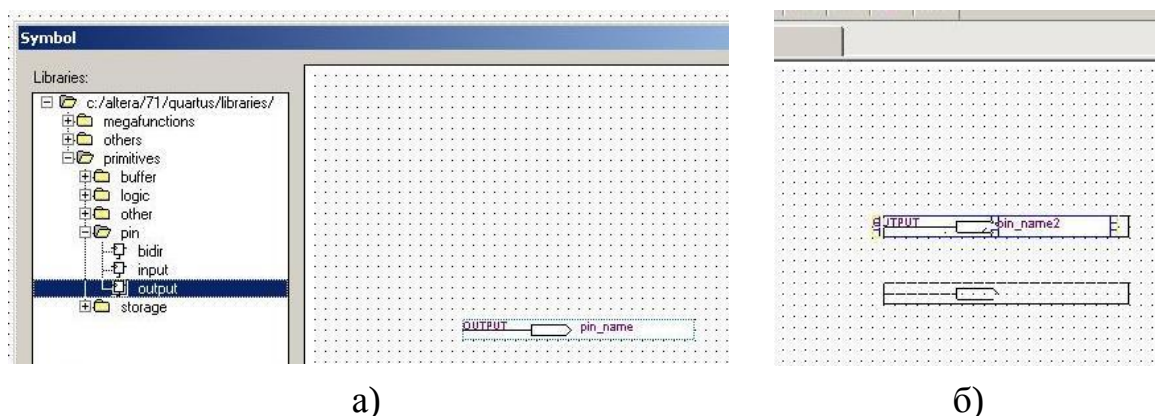


Рис. П.39. Выбор элемента типа «output» («Выход») (а) и его размещение на поле схемы (б)

3.22. Для изменения наименования входной или выходной цепи необходимо выполнить двойной клик мышкой на элементе типа «input», «output», при этом появляется диалоговое окно, представленное на рис. П.40. В верхнем поле ввода «Pinname(s):» необходимо ввести требуемое имя вывода (например, «KEY_1», «reset» и т.п.), соответствующее функциональному назначению данного вывода в схеме.

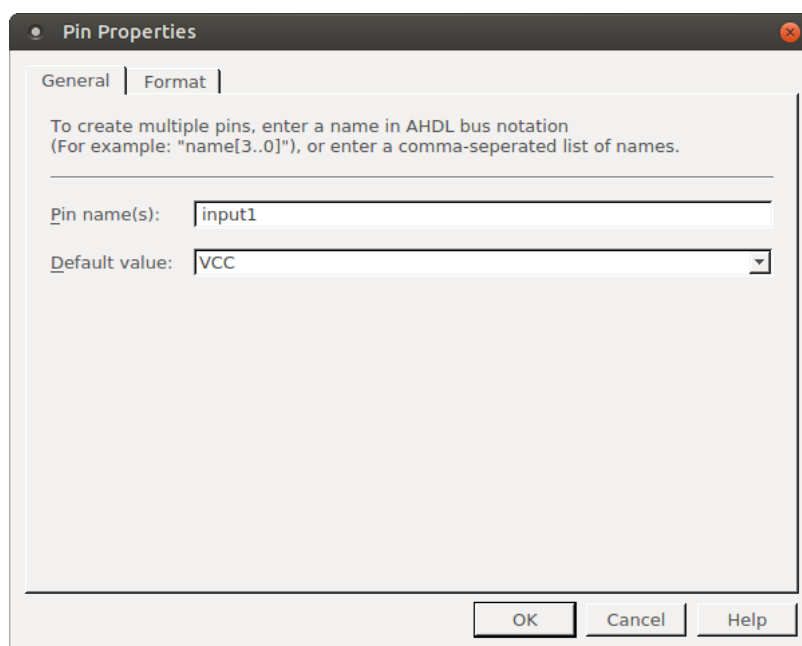


Рис. П.40. Диалоговое окно изменения свойств элемента «Input», «Output»

3.23. Для выполнения соединений между элементами используется инструмент панели инструментов «Orthogonal Node Tool» (рис. П.41). После выбора данного инструмента необходимо выполнить графические соединения между выводами установленных элементов аналогично отрисовке соединений в схемах электрических принципиальных. Для начала вы-

полнения соединения между входным и выходным элементами схемы подводят мышку к правому краю входного элемента (на рис. П.41 обозначено красным кругом), нажать левую кнопку мыши и, не отпуская кнопку мыши двигать мышь в сторону элемента «выход». При достижении левого края элемента «выход» курсор мыши приобретет форму перекрестия, если в этот момент отпустить левую кнопку мыши, то выполнение соединения завершится (рис. П.42).

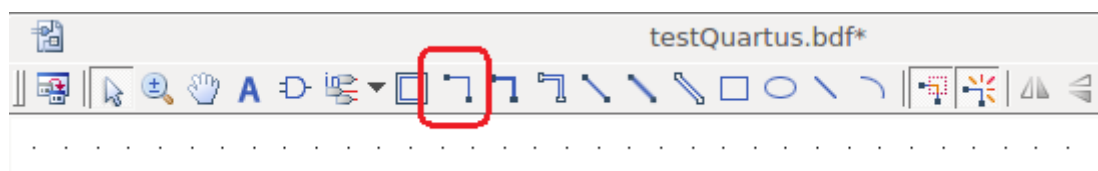


Рис. П.41. Выбор инструмента инструмент «OrthogonalNodeTool»

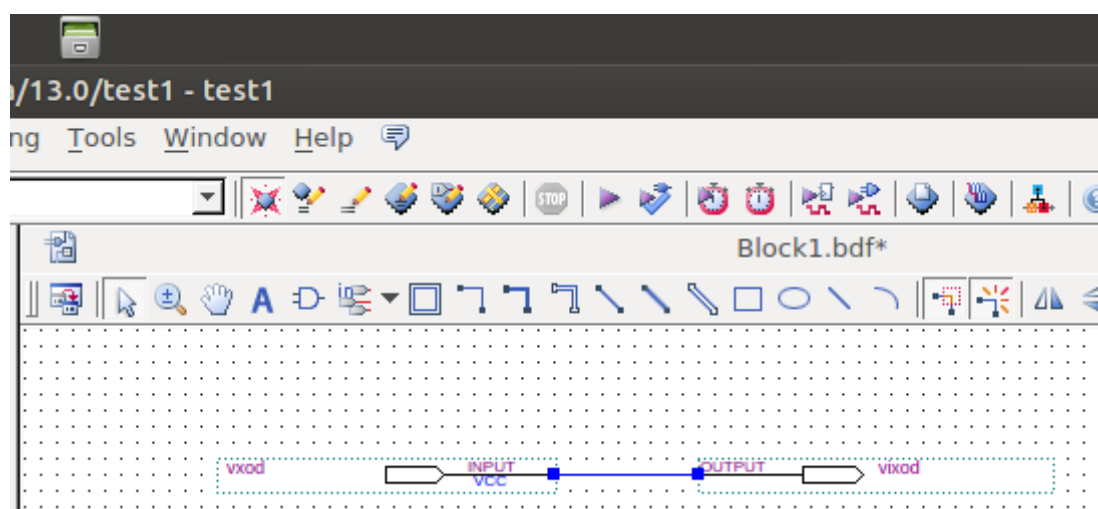


Рис. П.42. Выполнение межэлементного графического (логического) соединения

3.24. На данном этапе создание простой схемы завершено. Ее функционирование заключается в транслировании логического сигнала от входного вывода ПЛИС на ее выходной вывод. Следующим этапом требуется выполнить этап компиляции исходной схемы в выходной файл, загружаемый в ПЛИС, а также назначить соответствие между элементами типа «вход» и «выход» и физическими выводами микросхемы ПЛИС модуля «ALT-DIG-002». Для компиляции проекта в панели инструментов необходимо нажать кнопку «Start Compilation» (рис. П. 43). Окно среды разработки при этом примет вид, изображенный на рис. П.44. В левой части окна среды разработки отображается процент выполнения промежуточных этапов компиляции, в центральной части окна – суммарная информация о результатах компиляции.

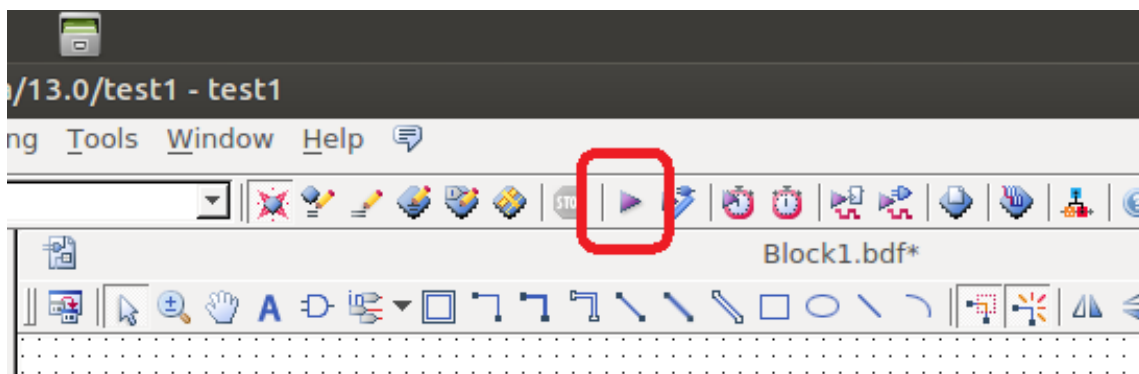


Рис. П.43. Расположение кнопки «Start Compilation»
на панели инструментов

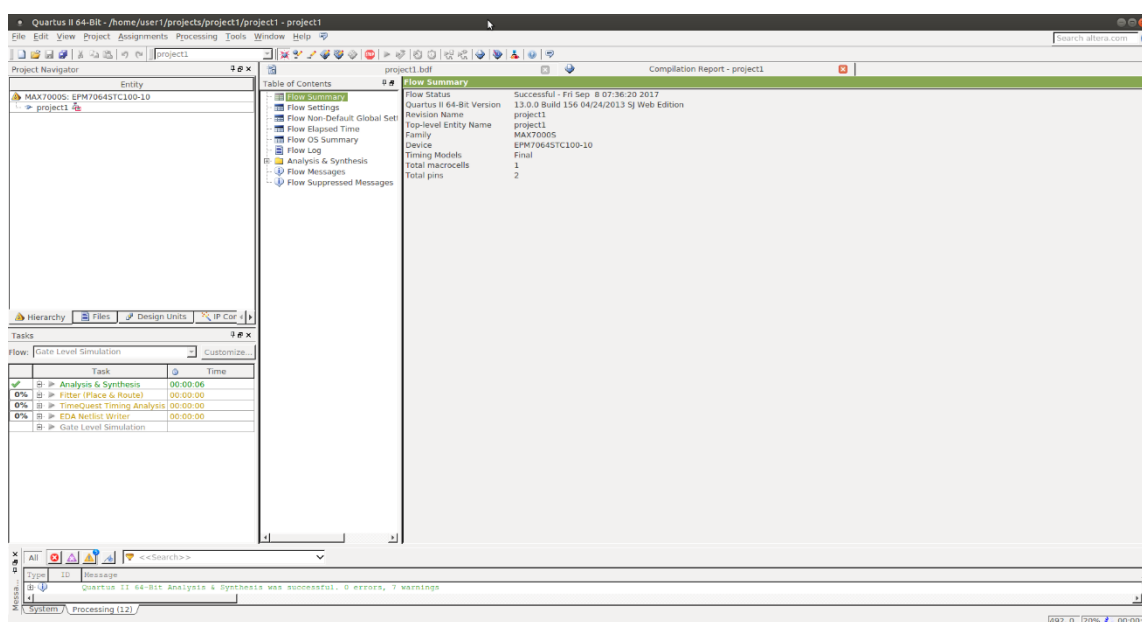


Рис. П.44. Внешний вид экрана среды разработки
во время компиляции проекта

3.25. В случае успешного завершения компиляции среда разработки выдает информационное сообщение (возможно с указанием количества предупреждений во время компиляции) (рис. П.45).

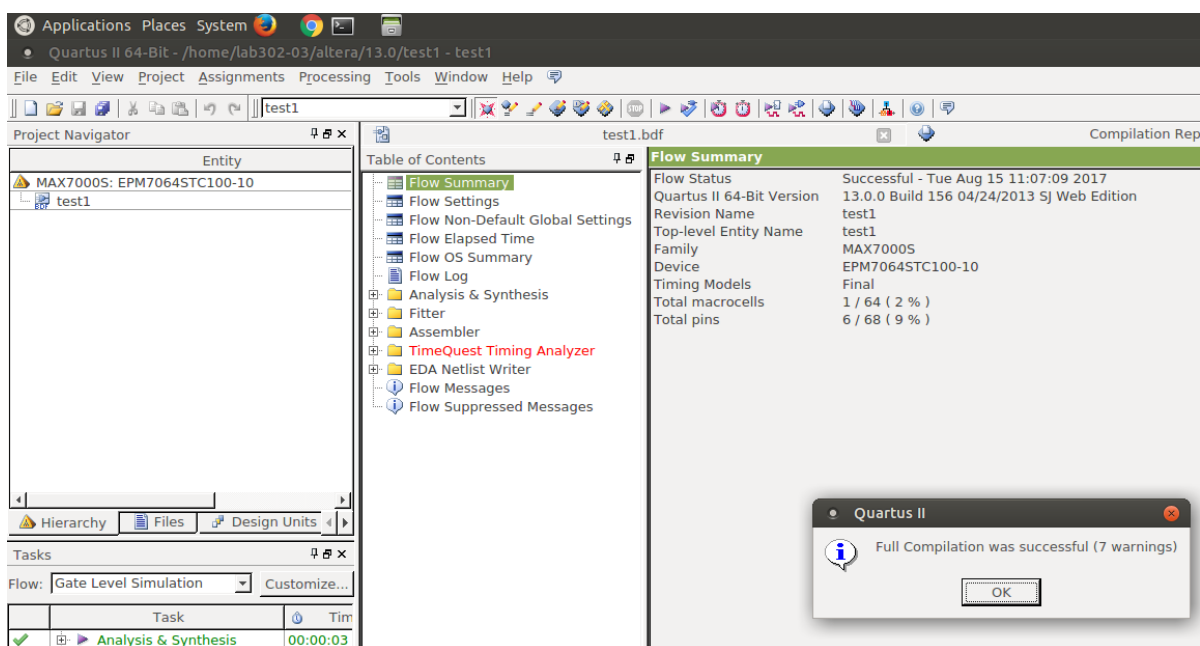


Рис. П.45. Диалоговое окно после завершения этапа компиляции

3.26. После этапа компиляции требуется выполнить назначение соответствия логических входов и выходов, используемых в схеме конкретным выводам микросхемы ПЛИС. Для этого используется средство «Pin Planner» («Планировщик выводов») (рис. П.46).

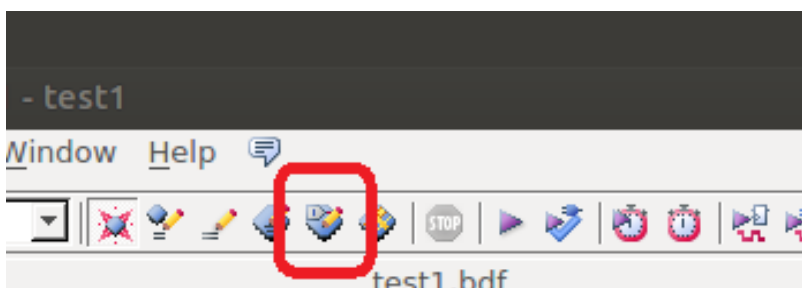


Рис. П.46. Расположение кнопки инструмента «Pin Planner» на панели инструментов

3.27. Экран в режиме назначения соответствия выводов имеет вид, приведенный на рис. П.47. В средней части окна схематично отображается корпус микросхемы (в случае модуля «ALT-DIG-002» – 100-выводный корпус TQFP-100), с условными обозначениями типа каждого вывода (вывод питания, вывод программирования, пользовательский вход-выход и т.д.). Ниже изображения ПЛИС представлена таблица входов-выходов, используемых в схеме (указаны назначенные имена входов-выходов, направление сигнала, но в столбце «Location» не указано физическое расположение входа либо выхода).

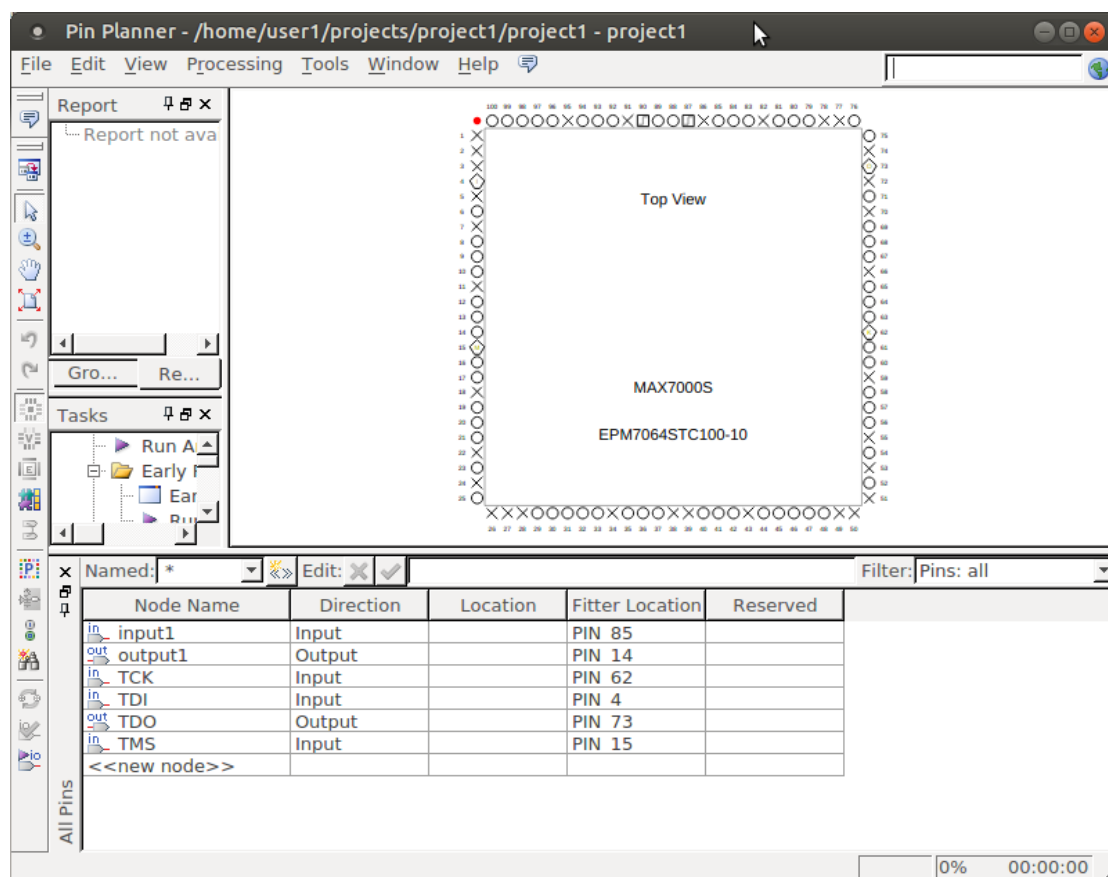


Рис. П.47. Экран среды разработки в режиме назначения соответствия логических входов/выходов схемы и физических выводов микросхемы ПЛИС

3.28. Для корректного назначения входа (выхода) на требуемый вывод ПЛИС необходимо знать электрическую принципиальную схему используемого устройства, либо таблицы соответствия внешних по отношению к ПЛИС устройств (устройств формирования входных сигналов, устройств индикации и т.д.) и выводов ПЛИС. Для модуля «ALT-DIG-002» внешний вид и таблицы соответствия устройств приведена на рис. П. 48–П.50. В случае использования иных цифровых устройств требуется аналогичные описания.

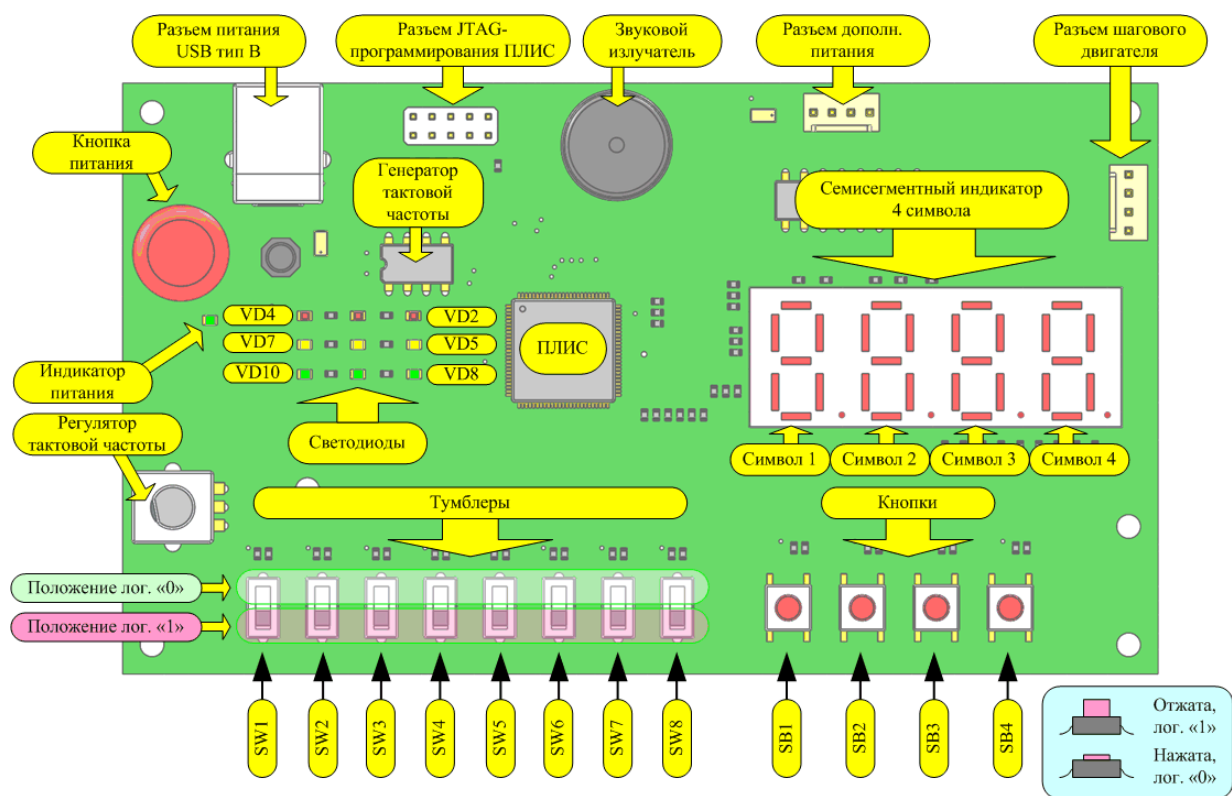


Рис. П.48. Эскиз внешнего вида модуля «ALT–DIG–002» с указанием групп его органов управления, индикации и т.д.

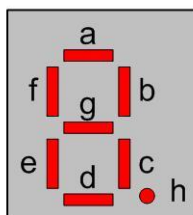
Таблицы соответствия органов управления и индикации и выводов ПЛИС ALTERA EPM7064

Тип	Напр.	Обознач.	Вывод ПЛИС
Тумблер	вход	SW1	19
		SW2	20
		SW3	21
		SW4	23
		SW5	25
		SW6	29
		SW7	30
		SW8	31
Кнопка	вход	SB1	32
		SB2	33
		SB3	35
		SB4	36

Тип	Напр.	Обознач.	Вывод ПЛИС
Единичный светодиод	выход	VD2	6
		VD3	8
		VD4	9
		VD5	10
		VD6	12
		VD7	13
		VD8	14
		VD9	16
		VD10	17
Звукоизлуч.		B1	100

Рис. П.49. Таблицы соответствия физических выводов микросхемы ПЛИС и подключенных к ним устройств

**Стандартизованные обозначения
сегментов 7-сегментного индикатора**



**Таблица соответствия 4-символьного 7-сегментного
индикатора и выводов ПЛИС ALTERA EPM7064**

Символ	Сегмент	Вывод ПЛИС	Символ	Сегмент	Вывод ПЛИС
Символ 1	a	81	Символ 3	a	71
	b	83		b	69
	c	64		c	46
	d	63		d	47
	e	61		e	48
	f	79		f	76
	g	80		g	75
	h	60		h	45
Символ 2	a	85	Символ 4	a	67
	b	92		b	65
	c	54		c	40
	d	57		d	42
	e	58		e	44
	f	84		f	68
	g	56		g	41
	h	52		h	37

Рис. П.50. Таблицы соответствия физических выводов микросхемы ПЛИС и подключенных к ним четырехсимвольного семисегментного индикатора

3.29. Для созданной в п.23 схемы в качестве источника входного сигнала возможно указать тумблер «SW1» (из таблицы рис. П. 49 следует, что он подключен к выводу 19 ПЛИС), выходной сигнал возможно подключить к единичному светодиоду «VD2» (из таблицы рис. П. 49 следует, что он подключен к выводу 6 ПЛИС). Для назначения соответствия логических и физических входов-выходов используется столбец «Location» диалогового окна, приведенного на рис. П.47. При двойном нажатии левой кнопки мыши в требуемой ячейке таблицы возникает выпадающий список с номерами неназначенных (свободных) выводов ПЛИС. Назначения соответствия выполняется выбором требуемого вывода из списка (рис. П.51).

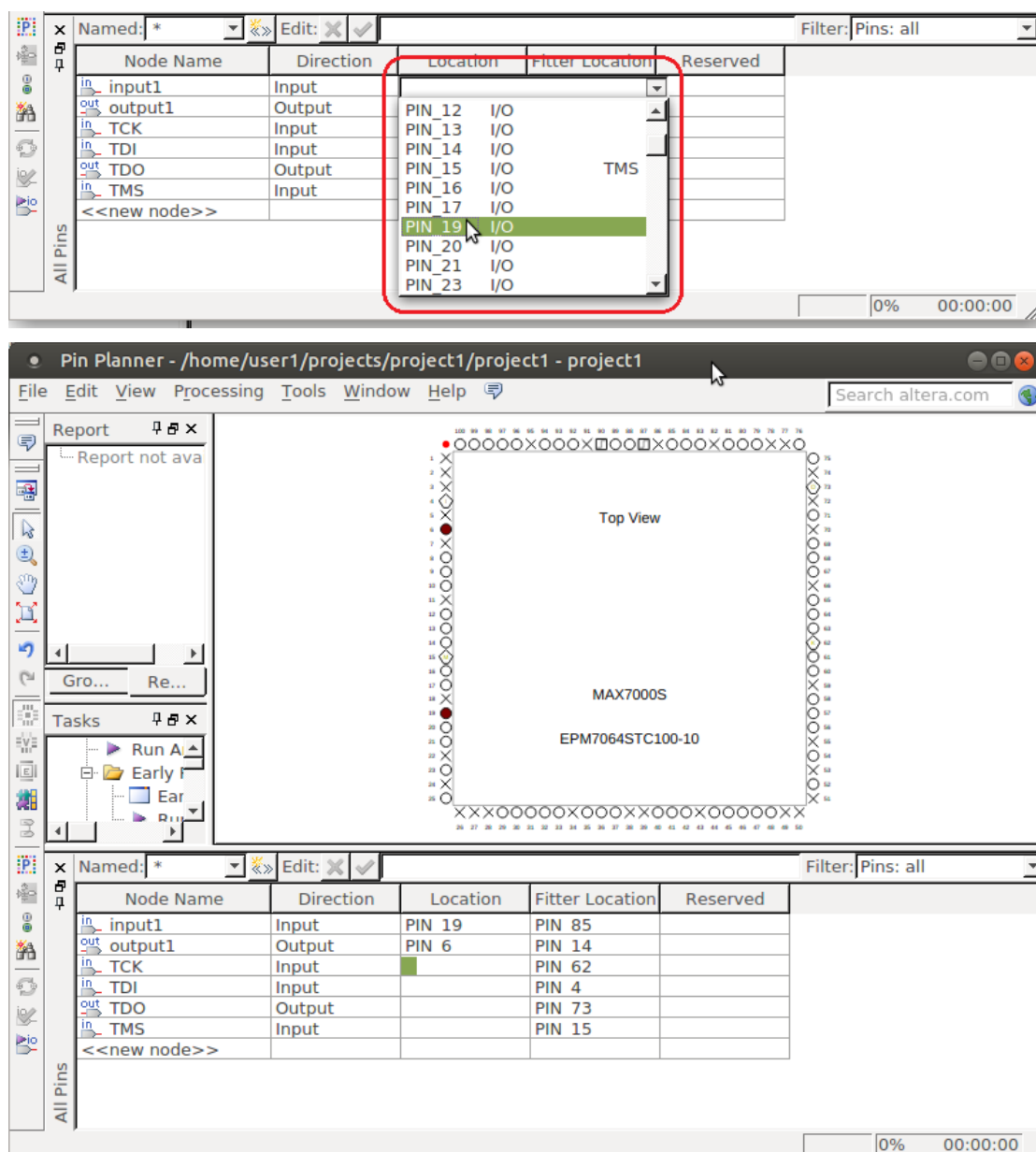


Рис. П.51. Формирование соответствия логического вывода схемы и физического вывода микросхемы ПЛИС

3.30. Следующим этапом необходимо выполнить повторную компиляцию (с учетом выполненных изменений в проекте по расположению выводов). Действие выполняется аналогично описанному в пункте 24.

3.31. После компиляции требуется выполнить загрузку скомпилированного выходного файла (конфигурации) в микросхему ПЛИС. Для этого используется инструмент «Programmer» («Программатор») (рис. П.52). Внешний вид экрана среды разработки в режиме конфигурирования приведен на рис. П.53.

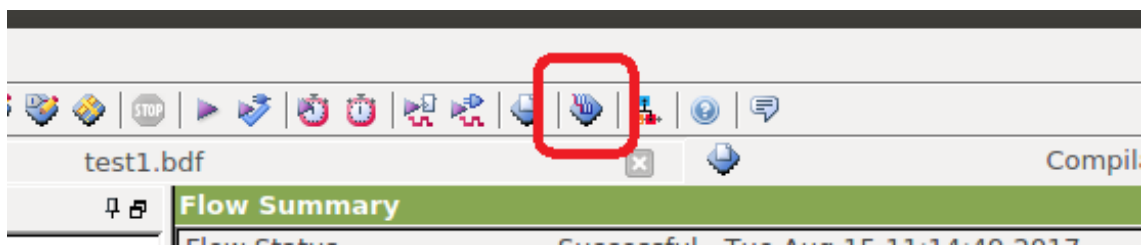


Рис. П.52. Расположение кнопки инструмента «Programmer» на панели инструментов

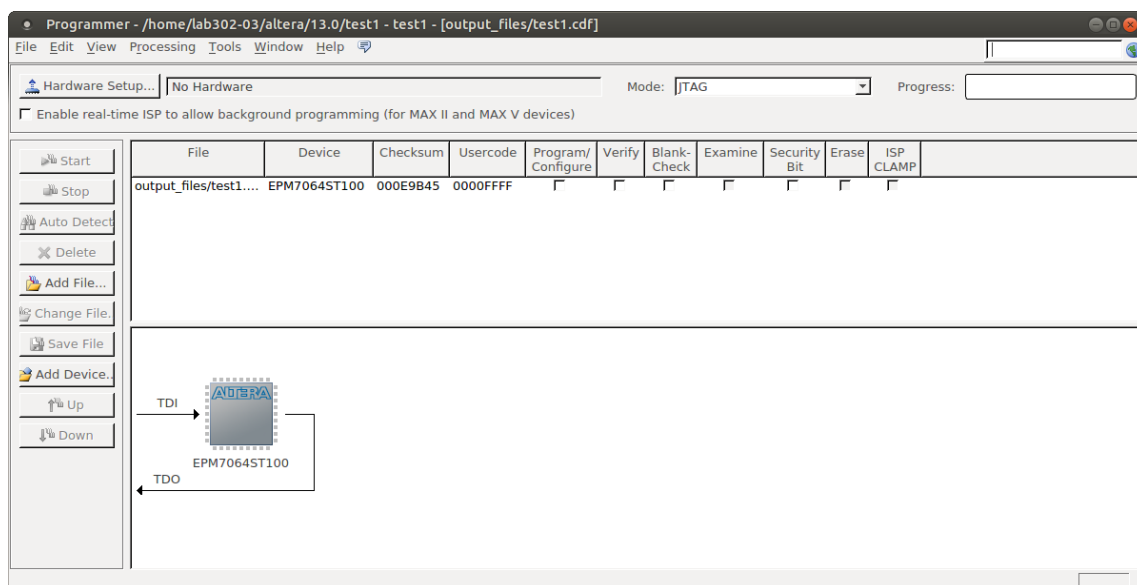


Рис. П.53. Внешний вид экрана среды разработки в режиме конфигурирования

3.32. При первом запуске операции конфигурирования ПЛИС требуется настроить оборудование, с помощью которого будет выполняться данная операция. Для этого необходимо нажать кнопку «Hardware Setup...» («Настройка оборудования...») (рис. П.54).

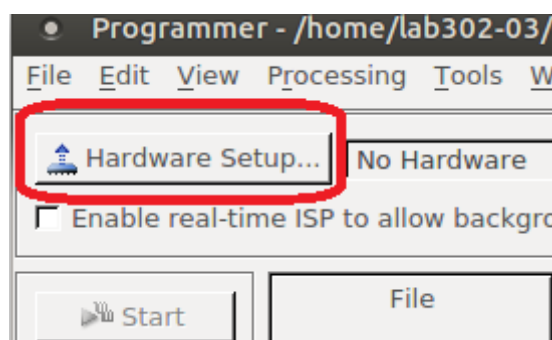


Рис. П.54. Кнопка перехода в режим настройки оборудования конфигурирования

3.33. После нажатия кнопки запускается следующее диалоговое окно настройки (рис. П.55).

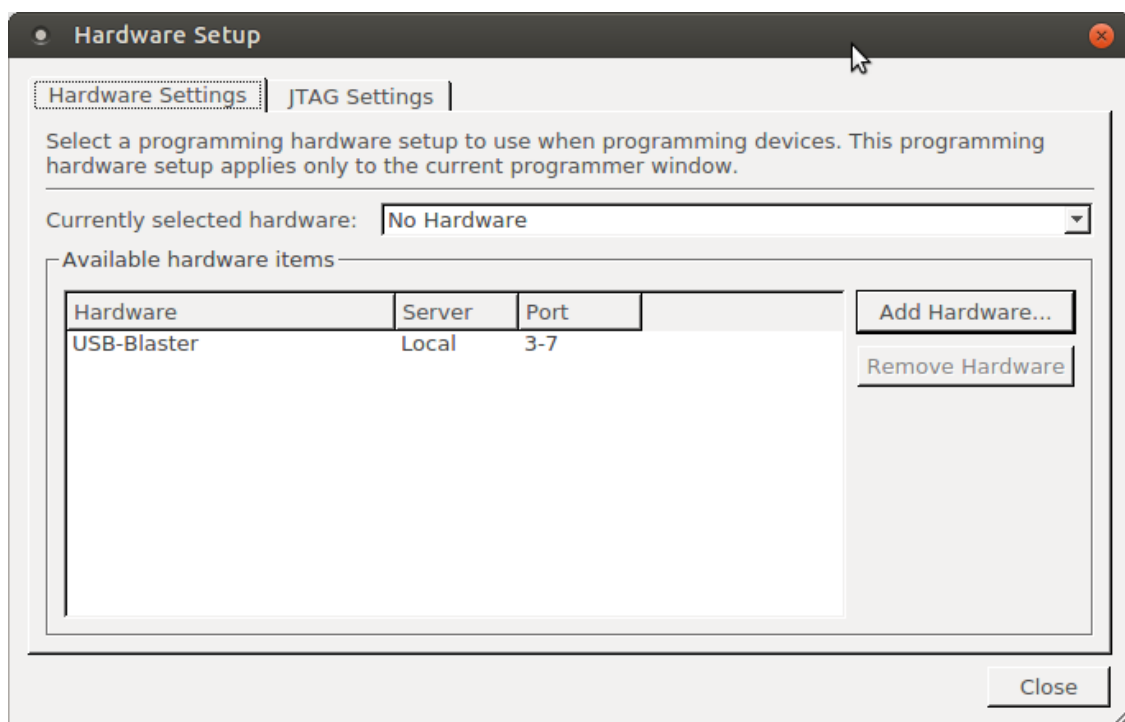


Рис. П.55. Диалоговое окно настройки оборудования конфигурирования

Перечень подключенного оборудования находится в средней части диалогового окна. Дополнительные настройки и добавление оборудования можно выполнить, нажав клавишу «Add Hardware...». Выбор типа программатора осуществляется выбором позиции в выпадающем списке в верхней части диалогового окна (рис. П.56).

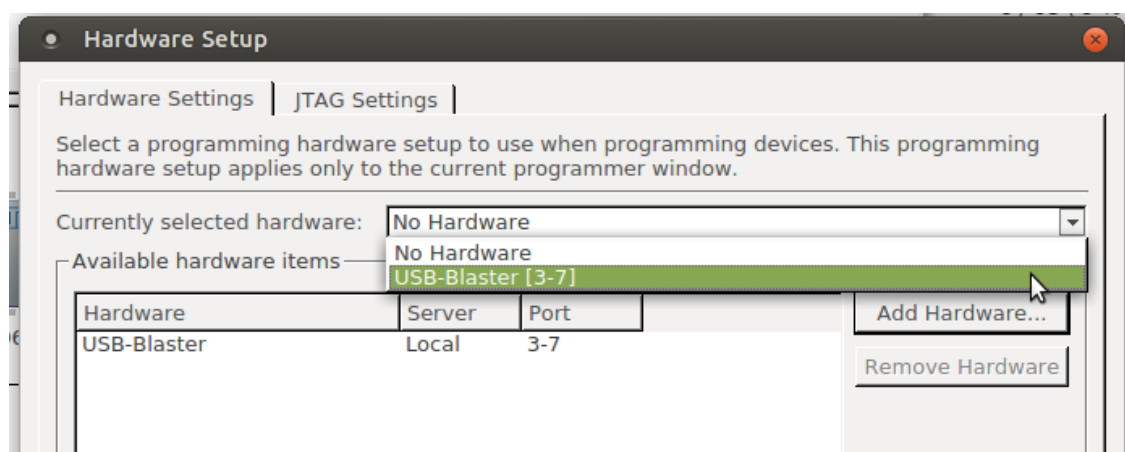


Рис. П.56. Выбор оборудования в выпадающем списке диалогового окна

3.34. После первоначальной (однократной, при первом запуске среды разработки) настройки оборудования возможно выполнять конфигурирование ПЛИС. В основном окне режима конфигурирования необходимо установить флажки режима конфигурирования (рис. П.57).

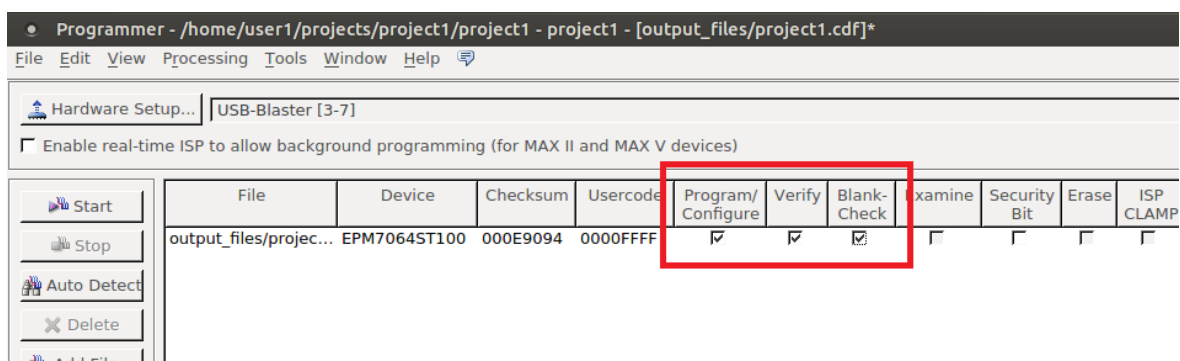


Рис. П.57. Установка режима программирования

Флажки означают:

«Blank-Check» – проверять микросхему на «чистоту», и в случае необходимости очищать ее (стирать предыдущую конфигурацию ПЛИС);

«Program/Configure» – программировать и выполнять конфигурацию;

«Verify» – проверять конфигурацию микросхемы после программирования на правильность;

3.35. Непосредственно конфигурирование выполняется нажатием клавиши «Start» (рис. П.58). Выполнение процесса конфигурирования индицируется на индикаторе «Progress» (рис. П.59).

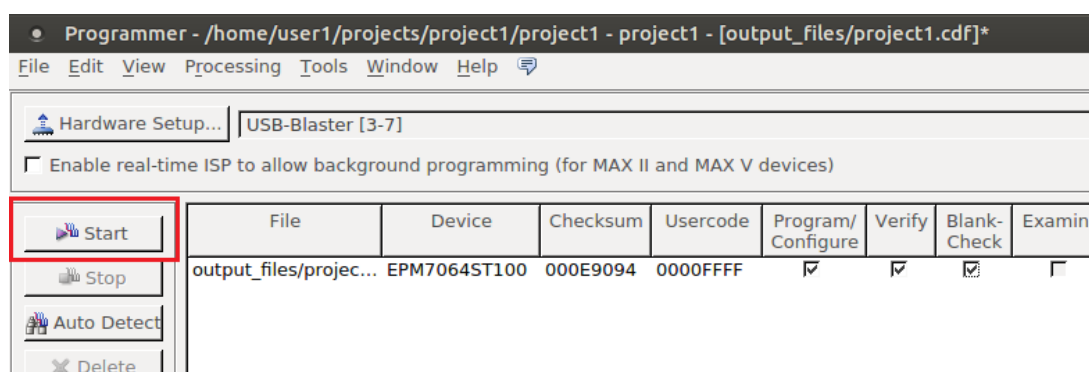


Рис. П.58. Запуск процесса конфигурирования

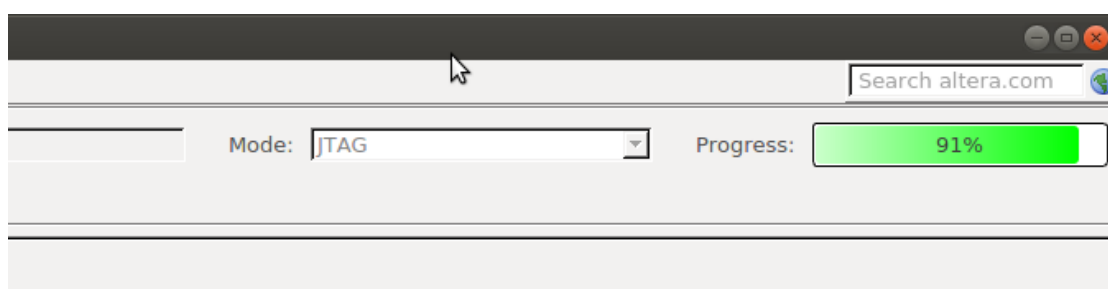


Рис. П.59. Индикация хода процесса конфигурирования

3.36. После окончания процесса конфигурирования микросхема ПЛИС (сконфигурированная логическая структура в ней) начинает принимать входные и формировать выходные сигналы согласно заданным логическим функциям.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Амосов, В.В. Схемотехника и средства проектирования цифровых устройств / В.В. Амосов. – СПб.: БХВ-Петербург, 2007. – 560 с.
2. Ангер, С. Асинхронные последовательностные схемы: пер. с англ. / С. Ангер, под ред. П.П. Пархоменко. – М.: Наука, 1977. – 400 с.
3. Ахметвалеева, Л.В. Основы цифровой электроники: учебно-методическое пособие/ Л.Г. Кулагина, Л.В. Ахметвалеева. – Казань: Казан. гос. энерг. ун-т, 2018. – 100 с.
4. Бабич, Н. П. Основы цифровой схемотехники: учебное пособие/ Н.П. Бабич, И.А. Жуков. – М.: ДМК Пресс, 2010. – 480 с. – ISBN 978-5-94120-115-0. – Текст: электронный// Лань: электронно-библиотечная система. – URL: <https://e.lanbook.com/book/60977> (дата обращения: 07.05.2023). – Режим доступа: для авториз. пользователей.
5. Бакшеева, Ю.В. Схемотехника цифровых устройств: учебное пособие/ Ю. В. Бакшеева. – СПб.: ГУАП, 2020. – 113 с. – ISBN 978-5-8088-1542-1. – Текст: электронный // Лань: электронно-библиотечная система. – URL: <https://e.lanbook.com/book/216512> (дата обращения: 07.05.2023). – Режим доступа: для авториз. пользователей.
6. Бойко, В.И. Схемотехника электронных систем. Цифровые устройства/ В.И. Бойко, А.Н. Гуржий и др. – СПб.: БХВ-Петербург, 2004. – 512 с.
7. Дэвид, М.Х. Цифровая схемотехника и архитектура компьютера/ М.Х. Дэвид, Л.Х. Сара. – М.: ДМК Пресс, 2017. – 792 с. – ISBN 978-5-97060-522-6. – Текст: электронный // Лань: электронно-библиотечная система. – URL: <https://e.lanbook.com/book/97336> (дата обращения: 07.05.2023). – Режим доступа: для авториз. пользователей.
8. Зубчук, В.И. Справочник по цифровой схемотехнике/ В.И. Зубчук, В.П. Сигорский, А.Н. Шкуро. – Киев: Техника, 1990. – 448 с.
9. Осадченко, В.Х. Функциональные узлы цифровой электроники: учебно-методическое пособие/ В.Х. Осадченко, Я.Ю. Волкова; Министерство науки и высшего образования Российской Федерации, Уральский федеральный университет им. первого Президента России Б.Н. Ельцина. – Екатеринбург: Изд-во Урал. ун-та, 2022. – 124 с. – ISBN 978-5-7996-3462-9.
10. Потемкин, И.С. Функциональные узлы цифровой автоматики/ И.С. Потемкин. – М.: Энергоатомиздат, 1988. – 320 с.
11. Пухальский, Г.И. Логическое проектирование цифровых устройств радиотехнических систем/ Г.И. Пухальский. – Л.: Изд-во Ленингр. ун-та, 1976. – 231 с.
12. Пухальский, Г.И. Цифровые устройства: учебное пособие для втузов/ Г.И. Пухальский, Т.Я. Новосельцева. – СПб.: Политехника, 1996. – 885 с.

13. Собакин, Е.Л. Цифровая схемотехника: учеб. пособие / Е.Л. Собакин. – Томск: Изд. ТПУ, 2002. – Ч. I. – 160 с.
14. Угрюмов, Е.П. Цифровая схемотехника / Е.П. Угрюмов. – СПб.: БХВ-Петербург, 2000. – 528 с.
15. Фридман, А. Теория и проектирование переключательных схем: Пер. с англ./ А. Фридман, П. Менон, под ред. В.А. Тафта. – М.: Мир, 1978. – 580 с.
16. Фролов, А.В. Схемотехника цифровых устройств: лабораторный практикум: учебное пособие/ А.В. Фролов. – Комсомольск-на-Амуре: КНАГУ, 2022. – 128 с. – ISBN 978-5-7765-1469-2. – Текст: электронный // Лань: электронно-библиотечная система. – URL: <https://e.lanbook.com/book/310778> (дата обращения: 07.05.2023). – Режим доступа: для авториз. пользователей.
17. Feher, J. Introduction to Digital Logic with Laboratory Exercises / James Feher. – Global Text Project, 2009. – 105 p.
18. Micheli, G. Synthesis and optimization of digital circuits / G. De Micheli. – McGraw Hill, 1994. – 597 p.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
ЛАБОРАТОРНАЯ РАБОТА № 1. ИССЛЕДОВАНИЕ ПОВЕДЕНИЯ ЛОГИЧЕСКИХ ЭЛЕМЕНТОВ И ЛОГИЧЕСКИХ СХЕМ НА ИХ ОСНОВЕ.....	4
Теоретический материал	4
Задание на лабораторную работу и порядок ее выполнения	15
Контрольные вопросы	21
ЛАБОРАТОРНАЯ РАБОТА № 2. СИНТЕЗ КОМБИНАЦИОННЫХ СХЕМ НА БАЗОВЫХ ЛОГИЧЕСКИХ ЭЛЕМЕНТАХ. ОПТИМИЗАЦИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ.....	22
Теоретический материал	22
Задание на лабораторную работу и порядок ее выполнения	43
Контрольные вопросы	48
ЛАБОРАТОРНАЯ РАБОТА № 3. КОМБИНАЦИОННЫЕ СХЕМЫ СРЕДНЕЙ СТЕПЕНИ ИНТЕГРАЦИИ. ШИФРАТОРЫ, ПРЕОБРАЗОВАТЕЛИ КОДОВ	48
Теоретический материал	48
Задание на лабораторную работу и порядок ее выполнения	56
Контрольные вопросы	61
ЛАБОРАТОРНАЯ РАБОТА № 4. КОМБИНАЦИОННЫЕ СХЕМЫ СРЕДНЕЙ СТЕПЕНИ ИНТЕГРАЦИИ. МУЛЬТИПЛЕКСОРЫ	61
Теоретический материал	61
Задание на лабораторную работу и порядок ее выполнения	67
Контрольные вопросы	70
ПРИЛОЖЕНИЕ ОПИСАНИЕ КОМПЛЕКТА ЛАБОРАТОРНОГО ОБОРУДОВАНИЯ «ALT–DIG–002»	
1. Состав комплекта лабораторного оборудования	71
2. Порядок работы с комплектом (подготовка к работе, завершение работы)	74
3. Работа в среде проектирования «ALTERAQUARTUS» (на примере работы в версии 13.0 для операционной системы Linux Ubuntu)	76
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	105

Учебное издание

Носиков Максим Владимирович

ЦИФРОВАЯ СХЕМОТЕХНИКА

Учебное пособие для выполнения
лабораторных работ

Часть 1

Техн. редактор *А.В. Миних*

Издательский центр Южно-Уральского государственного университета

Подписано в печать 14.04.2023. Формат 60×84 1/16. Печать цифровая.
Усл. печ. л. 6,28. Тираж 50 экз. Заказ 109/202.

Отпечатано в типографии Издательского центра ЮУрГУ.
454080, г. Челябинск, проспект Ленина, 76.