**МЕТОДИЧЕСКИЕ УКАЗАНИЯ**

**К лабораторным работам по дисциплине**

**«СХЕМОТЕХНИКА ЭВМ» с использованием САПР Altera Quartus II (Часть I)**

(для студентов специальности 220100)

Москва – 13.05.2013

**ЛАБОРАТОРНАЯ РАБОТА №1.**

**СИНТЕЗ И ИССЛЕДОВАНИЕ КОМБИНАЦИОННЫХ СХЕМ**

**НА ЛОГИЧЕСКИХ ЭЛЕМЕНТАХ**

**Цель лабораторной работы**

Изучение методов синтеза комбинационных схем на логических элементах заданной серии микросхем и получение навыков в моделировании, макетировании, наладке и экспериментальном исследовании синтезируемых схем.

**Подготовка к выполнению работы**

1. Произвести инсталляцию САПР Altera Quartus II, в соответствии с Приложением 1.

2. Изучить приведенные ниже примеры синтеза комбинационных схем.

3. Синтезировать и нарисовать одновыходную комбинационную схему по заданной функции *f*1 (табл. 1) на инверторах и элементах И-НЕ микросхем серии К155, приведенных в приложении 4. При этом полученная схема должна содержать минимальное количество корпусов микросхем, а при одинаковом количестве корпусов должна иметь максимальное количество выводов у свободных элементов.

4. Синтезировать и нарисовать многовыходную комбинационную схему, по заданным функциям *f*2,  *f*3, (табл. 1) на инверторах и элементах И-НЕ микросхем серии К155, приведенных в приложении 4. При этом полученная схема должна содержать минимальное количество корпусов микросхем, а при одинаковом количестве корпусов должна иметь максимальное количество выводов у свободных элементов.

5. Построить модели синтезированных схем (для каждой схемы должен быть создан индивидуальный проект) при помощиСАПР Altera Quartus II (см. Приложение 2).

6. Получить временные диаграммы работы моделей синтезированных схем путем моделирования их работы в режимах «functional» и «timing». Порядок изменения входных сигналов схемы на диаграммах должен соответствовать таблице 3, а минимальная длительность входного сигнала должна быть подобрана таким образом, чтобы задержки схемы, наблюдаемые в режиме «timing» составляли не более 30% длительности входного сигнала.

7. Сопоставить результаты моделирования с заданием на лабораторную работу и в случае несовпадения внести необходимые изменения в схему и её модель и вернуться к выполнению п. 6.

8. Ответить на контрольные вопросы.

9. Составить отчет о подготовке к лабораторной работе (в формате WORD) и выслать в электронном виде преподавателю для проверки (вместе с проектами).

10. Распечатать проверенный и одобренный преподавателем отчет.

Примечание:

а) при изображении схем использовать стандартные условные обозначения элементов с обязательным указанием их порядковых номеров в микросхеме и типа микросхемы.

б) студенты, не выполнившие подготовку в полном объеме, к лабораторной работе не допускаются.

**Таблица 1.**

Варианты заданий на лабораторную работу

|  |  |  |  |
| --- | --- | --- | --- |
| Вар. | *f*1 | *f*2 | *f*3 |
| 1 | 0,1,2,3,4,7,8,9,11,12,14,15 | 1,2,4,5,9,10,12,13,14,15 | 0,1,2,5,6,7,9,10,12,13,14,15 |
| 2 | 0,1,2,4,5,6,7,8,9,13 | 3,4,5,6,7,10,11,12,15 | 0,2,3,4,5,6,7,10,11,15 |
| 3 | 0,1,2,3,5,6,9,10,11,12,13,14 | 1,2,3,5,6,8,10,12,13,14,15 | 1,2,5,7,8,12,13,14,15 |
| 4 | 0,1,2,4,5,6,7,8,9,13 | 0,2,3,4,5,6,7,10,11,15 | 0,3,4,6,7,9,10,11 |
| 5 | 0,1,2,3,4,7,9,10,11,13,14 | 1,2,4,5,9,10,12,13,14,15 | 0,1,2,5,6,7,9,10,12,13,14,15 |
| 6 | 0,2,3,4,5,7,12,13,14,15 | 0,3,6,7,8,11,12,13,14,15 | 0,2,3,4,5,7,8,11,12,13,14,15 |
| 7 | 0,3,4,5,6,7,8,10,11,12,13,15 | 0,3,5,7,10,12,13,14,15 | 0,1,3,4,7,8,10,12,13,14,15 |
| 8 | 0,2,4,5,6,7,8,9,11,12,15 | 0,2,3,4,7,8,11,12,13,14 | 0,1,2,3,4,7,8,9,11,12,14,15 |
| 9 | 1,2,4,6,7,8,9,10,11,13,14 | 1,4,5,6,7,8,9,13,14 | 0,1,2,4,5,6,7,8,9,13 |
| 10 | 0,1,5,8,9,10,12,13,14,15 | 0,1,2,5,6,9,10,12,14,15 | 0,1,2,3,5,6,9,10,11,12,13,14 |
| 11 | 0,1,2,3,5,6,8,9,11,12,15 | 0,1,2,4,5,6,7,8,9,13 | 1,2,4,5,6,8,9,11 |
| 12 | 0,1,5,8,9,10,12,13,14,15 | 0,1,2,3,4,7,9,10,11,13,14 | 0,1,4,6,9,11,13,14 |
| 13 | 0,3,4,5,6,7,8,11,14,15 | 0,3,6,7,8,11,12,13,14,15 | 0,2,3,4,5,7,12,13,14,15 |
| 14 | 0,2,4,5,6,7,9,10,11,13,14 | 0,3,4,5,6,7,8,11,14,15 | 0,3,4,5,6,7,8,10,11,12,13,15 |
| 15 | 2,3,4,7,11,12,13,14,15 | 2,4,5,6,7,8,11,13,15 | 0,2,4,5,6,7,8,9,11,12,15 |
| 16 | 0,1,3,4,7,8,9,10,11,13,14 | 1,2,4,6,7,8,9,10,11,13,14 | 1,2,3,5,6,8,9,10,11,13,14 |
| 17 | 0,1,2,5,6,7,9,10,12,13,14 | 0,1,5,6,9,12,13,14,15 | 0,1,5,8,9,10,12,13,14,15 |
| 18 | 3,4,5,6,7,10,11,12,15 | 2,3,4,6,9,11,12,15 | 0,1,2,3,5,6,8,9,11,12,15 |
| 19 | 1,2,3,5,6,8,10,12,13,14,15 | 0,1,5,8,9,10,12,13,14,15 | 0,1,3,9,10,12,13,14 |
| 20 | 0,2,3,4,5,6,7,10,11,15 | 0,3,4,5,6,7,8,11,14,15 | 0,3,4,5,6,7,8,10,11,12,13,15 |
| 21 | 0,1,2,5,6,7,9,10,12,13,14,15 | 0,2,4,5,6,7,9,10,11,13,14 | 0,4,5,6,7,9,10,13,15 |
| 22 | 0,2,3,4,5,7,8,11,12,13,14,15 | 2,3,4,7,11,12,13,14,15 | 2,3,7,8,10,11,12,13,14,15 |
| 23 | 0,1,2,3,4,7,8,9,11,12,14,15 | 0,1,3,4,7,8,9,10,11,13,14 | 1,3,4,7,10,11,12,14 |
| 24 | 0,1,2,4,5,6,7,8,9,13 | 0,3,4,5,6,8,9,10,11,12,15 | 0,1,3,4,6,7,8,9,10,11,12,15 |
| 25 | 0,1,2,5,6,9,10,12,14,15 | 2,3,7,8,10,11,12,13,14,15 | 1,2,3,8,11,12,14,15 |
| 26 | 0,1,2,4,5,6,7,8,9,13 | 1,2,4,5,6,7,9,10,12,13 | 1,2,4,5,6,7,8,9,10,13,14,15 |
| 27 | 0,1,2,3,4,7,9,10,11,13,14 | 1,3,5,6,8,9,12,14 | 1,2,3,5,6,8,9,10,11,12,15 |
| 28 | 0,2,3,4,5,7,12,13,14,15 | 1,2,4,5,6,7,9,10,12,13 | 1,2,4,5,6,7,8,9,10,13,14,15 |
| 29 | 0,3,4,5,6,7,8,10,11,12,13,15 | 2,10,11,13 | 0,1,4,8,9,11,12,13 |
| 30 | 0,2,4,5,6,7,8,9,11,12,15 | 0,1,3,4,5,8,9,12 | 2,3,5,10 |
| 31 | 0,1,3,9,11,13,15 | 4,5,6,7,9,12,13,15 | 0,1,9,15 |
| 32 | 3,4,5,7,8,9,11,12,14,15 | 2,3,4,6,7,8,10,11,13,14 | 1,2,4,7,8,10,11,12,14 |
| 33 | 2,5,6,7,9,10,!2,13,14 | 3,11,12,14 | 1,2,5,9,10,11,12 |
| 34 | 1,2,4,7,8,10,11,12,14 | 0,5, 6,7,8,9,10,11,15 | 5, 6,7,8,9,10,12, 14,15 |
| 35 | 1,2,3,5,6,7,10,12,13,15 | 0,1,5,7,9,10,12,13,14,15 | 0,1,3,6,8,10,12,13,14 |
| 36 | 0,3,4,5,6,7,8,10,11,12,13,15 | 0,1,5,6,8,9,10,13,15 | 1,2,3,4,6,8,9,13,14 |

**Примеры синтеза комбинационных схем**

Пусть заданы функции четырех переменных *f*1(*Х*1,*Х*2,*Х*3,*Х*4), *f*2(*Х*1,*Х*2,*Х*3,*Х*4), *f*3(*Х*1,*Х*2,*Х*3,*Х*4), в форме перечня десятичных номеров конституент единицы (табл. 2), что соответствует 31 варианту задания на лабораторную работу.

**Таблица 2.**

|  |  |  |  |
| --- | --- | --- | --- |
| Функция | *f*1 | *f*2 | *f*3 |
| Перечень десятичных номеров конституент | 0,1,3,9,11,13,15 | 4,5,6,7,9,12,13,15 | 0,1,9,15 |

1. Составим таблицу истинности для заданных функций (табл. 3).

**Таблица 3.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Десятичный номер конституенты | Десятичный код номера конституенты | | | | Значения функций | | |
|  | *Х*4 | *Х*3 | *Х*2 | *Х*1 | *f*1 | *f*2 | *f*3 |
| 0  1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | 0  0  0  0  0  0  0  0  1  1  1  1  1  1  1  1 | 0  0  0  0  1  1  1  1  0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1  0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1  0  1  0  1  0  1  0  1 | 1  1  0  1  0  0  0  0  0  1  0  0  0  1  0  1 | 0  0  0  0  1  1  1  1  0  1  0  0  1  1  0  1 | 1  1  0  0  0  0  0  0  0  1  0  1  0  0  0  1 |

**2.** Запишем заданные функции в совершенной дизъюнктивной нормальной форме:







**3.** Синтезируем одновыходную комбинационную схему, реализующую функцию *f*1(*Х*1,*Х*2,*Х*3,*Х*4) на элементах И-НЕ. Для этого:

а) произведем минимизацию функции с помощью карты Карно и получим

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 0 | 0 | 0 | 00 | *Х*1,*Х*2 |
| 0 | 0 | 0 | 0 | 01 |
| 1 | 1 | 1 | 0 | 11 |
| 1 | 1 | 1 | 0 | 10 |
| 00 | 01 | 11 | 10 |  |
| *Х*3,*Х*4 | | | | | *f*1 |



б) преобразуем полученное выражение к виду, удобному для реализации на элементах И-НЕ, используя следующие эквивалентности:





в) построим комбинационную схему, реализующую функцию *f*1(*Х*1, *Х*2, *Х*3, *Х*4) на элементах И-НЕ и инверторах (рис. 1), и определим, на каких логических элементах серии К155 эта схема будет сделана.

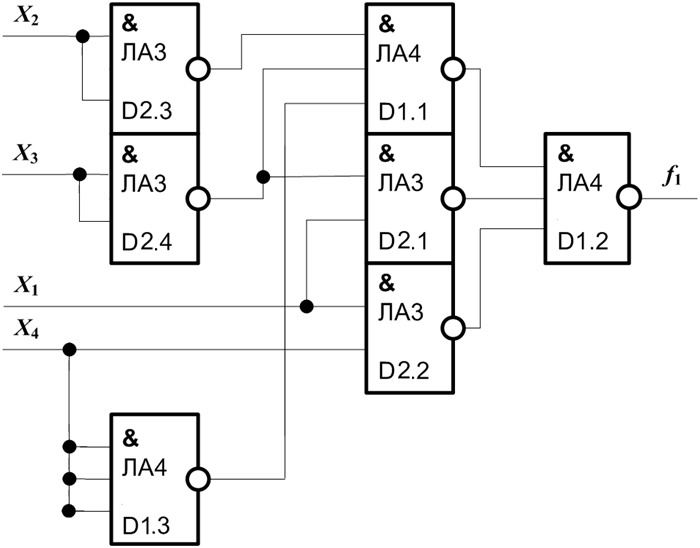


Рис. 1

4. Синтезируем многовыходную комбинационную схему, реализующую функции *f*2, *f*3 на элементах И-НЕ. Для этого:

а) занесем заданные функции на карты Карно и произведем их совместную минимизацию. При этом часто бывает целесообразным выделять общие для карт склейки (отмечены пунктиром);

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 0 | 0 | 0 | 00 | *Х*1,*Х*2 |
| 0 | 0 | 0 | 0 | 01 |
| 0 | 0 | 1 | 0 | 11 |
| 1 | 1 | 0 | 0 | 10 |
| 00 | 01 | 11 | 10 |  |
| *Х*3,*Х*4 | | | | | *f*3 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 0 | 1 | 1 | 00 | *Х*1,*Х*2 |
| 0 | 0 | 0 | 1 | 01 |
| 0 | 0 | 1 | 1 | 11 |
| 0 | 1 | 1 | 1 | 10 |
| 00 | 01 | 11 | 10 |  |
| *Х*3,*Х*4 | | | | | *f*2 |

б) выпишем минимальные выражения для заданных функций:





в) произведя преобразования аналогичные приведенным в пункте 3, получим для заданных функций выражения в виде, удобном для реализации на элементах И-НЕ и инверторах:





г) построим комбинационную схему, реализующую функции *f*2(*Х*1, *Х*2, *Х*3, *Х*4) и *f*3(*Х*1, *Х*2, *Х*3, *Х*4) на инверторах и элементах И-НЕ, и определим, на каких логических элементах серии К155 эта схема будет собрана (рис. 2).

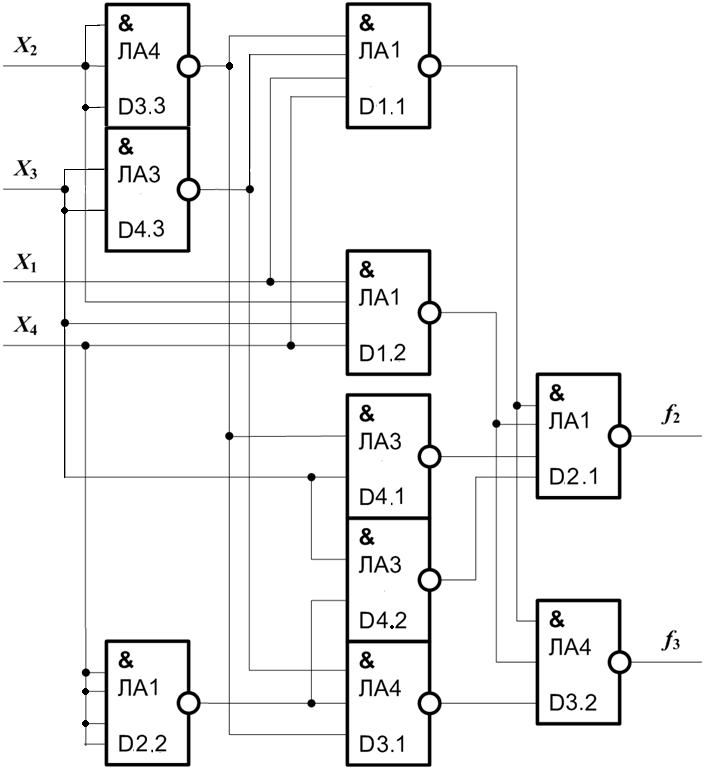


Рис. 2

**Порядок выполнения работы**

1. Представить преподавателю письменный отчет о подготовке к лабораторной работе, ответить на предложенные им контрольные вопросы и получить допуск к выполнению лабораторной работы.

2. Перенести на предоставленный компьютер разработанные проекты моделей комбинационных схем и убедиться в их работоспособности путем контрольного моделирования работы этих схем в режиме «timing».

3. Произвести (см. Приложение 3) закрепление входных и выходных сигналов одновыходной комбинационной схемы за элементами стенда (входные сигналы «Х1», «Х2», «Х3» и «Х4» должны поступать с тумблеров стенда, а выход схемы должен поступать на светодиодный индикатор стенда).

4. Осуществить прошивку проекта одновыходной комбинационной схемы на плату предоставленного стенда (см. Приложение 3).

5. Произвести проверку макета одновыходной комбинационной схемы в статическом режиме. Для этого:

а) убедиться, что при всех значениях входных сигналов «Х1», «Х2», «Х3» и «Х4» полученное значение функции совпадает с заданным;

б) в случае неправильной работы макета одновыходной комбинационной схемы ввести необходимые изменения в модель одновыходной комбинационной схемы и повторить выполнение п.п. «а» - «б».

в) продемонстрировать работу одновыходной комбинационной схемы преподавателю.

6. Произвести закрепление входных и выходных сигналов многовыходной комбинационной схемы за элементами стенда (входные сигналы «Х1», «Х2», «Х3» и «Х4» должны поступать с тумблеров стенда, а выходы схемы должны поступать на светодиодные индикаторы стенда).

7. Осуществить прошивку проекта многовыходной комбинационной схемы на плату предоставленного стенда.

8. Произвести проверку макета многовыходной комбинационной схемы в статическом режиме. Для этого:

а) убедиться, что при всех значениях входных сигналов «Х1», «Х2», «Х3» и «Х4» полученные значения функций совпадают с заданным;

б) в случае неправильной работы макета многовыходной комбинационной схемы ввести необходимые изменения в модель многовыходной комбинационной схемы и повторить выполнение п.п. «а» - «б».

в) продемонстрировать работу многовыходной комбинационной схемы преподавателю.

9. Получив разрешение преподавателя, осуществить выключение стенда.

**Содержание отчета**

1. Задание на лабораторную работу.

2. Материалы по синтезу комбинационных схем: таблица истинности для заданных функций, карты Карно, аналитические выражения и их преобразования, схемы аналогичные представленным на рис. 1 и рис. 2.

3. Модели разработанных схем.

4. Временные диаграммы работы синтезированных комбинационных схем, полученные путем моделирования их работы в режимах «functional» и «timing».

**Контрольные вопросы**

1. Что такое комбинационная схема?

2. В чем сущность синтеза комбинационных схем?

3. Всегда ли перед минимизацией переключательных функций их необходимо приводить к совершенным формам?

4. В чем состоит специфика синтеза многовыходных комбинационных схем?

5. В чем состоит преимущество комбинационных схем в базисе И-НЕ?

5. Как решить задачу построения комбинационной схемы на элементах И-НЕ, если число входов элемента меньше числа переменных в конъюнктивных термах минимизированной функции?

7. Какое количество микросхем необходимо для реализации многовыходной комбинационной схемы на элементах И-НЕ, исследованной при выполнении работы и можно ли его уменьшить путем минимизации функции инверсной относительно заданной?

**ЛАБОРАТОРНАЯ РАБОТА № 2**

**СИНТЕЗ И ИССЛЕДОВАНИЕ КОМБИНАЦИОННЫХ СХЕМ**

**НА БАЗЕ МУЛЬТИПЛЕКСОРОВ**

**Цель лабораторной работы**

Изучение методов синтеза комбинационных схем на базе мультиплексоров заданной серии микросхем и получение навыков в моделировании, макетировании, наладке и экспериментальном исследовании синтезируемых схем.

**Методика синтеза комбинационных схем на базе мультиплексоров**

При традиционном использовании мультиплексор является коммутатором цифровых сигналов, который в зависимости от кода, поданного на его адресные (управляющие) входы, передает на выход логический сигнал с одного из информационных входов. По этой причине основным назначением мультиплексора является организация управляемых информационных шин.

Однако мультиплексоры также имеют широкое применение для синтеза комбинационных схем общего вида.

Рассмотрим в качестве примера мультиплексор с двумя адресными входами. Обозначение и функциональная схема такого мультиплексора представлены на рис. 3.

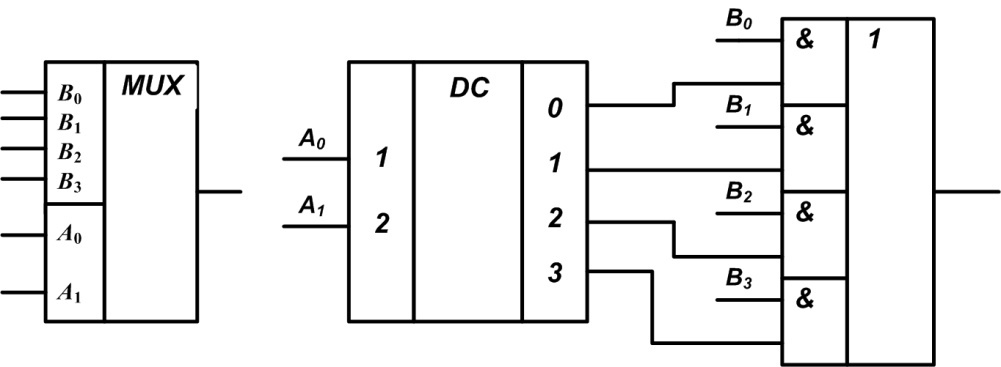


Рис. 3

Как видно из рисунка, мультиплексор состоит из дешифратора адреса информационного входа (DC) и логического элемента "И-ИЛИ". Входы дешифратора адреса называются адресными входами мультиплексора. Мультиплексор, имеющий n адресных входов, должен иметь 2n информационных входов (по одному входу на каждый выход дешифратора).

Микросхемы, реализующие мультиплексоры, входят в состав большинства современных серий микросхем. Эти мультиплексоры отличаются различным количеством адресных входов. Имеются также две модификации базовой схемы мультиплексора, придающие мультиплексорам дополнительные возможности.

Первая модификация мультиплексора состоит в наличии парафазного выхода. Функциональная схема мультиплексора с двумя адресными входами, соответствующая этой модификации, и ее обозначение представлены на рис. 4.

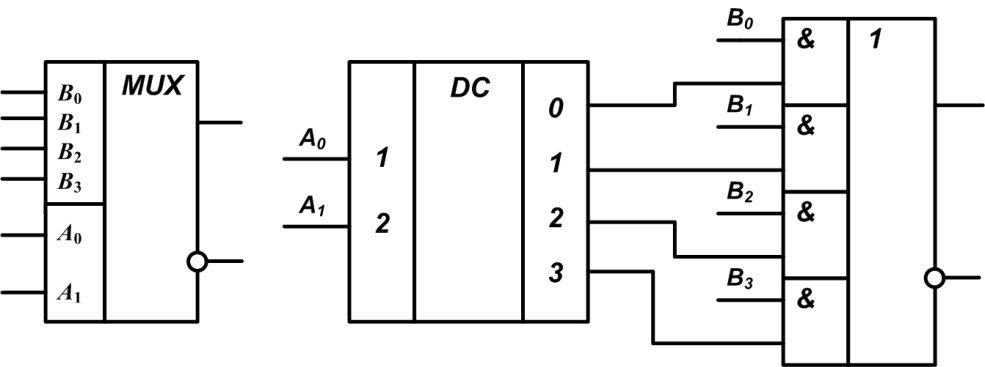


Рис. 4

Вторая модификация отличается от базовой схемы наличием разрешающего входа (Е). Функциональная схема мультиплексора с двумя адресными входами, соответствующая этой модификации, и ее обозначение представлены на рис. 5. Такие мультиплексоры называют селекторами-мультиплексорами.

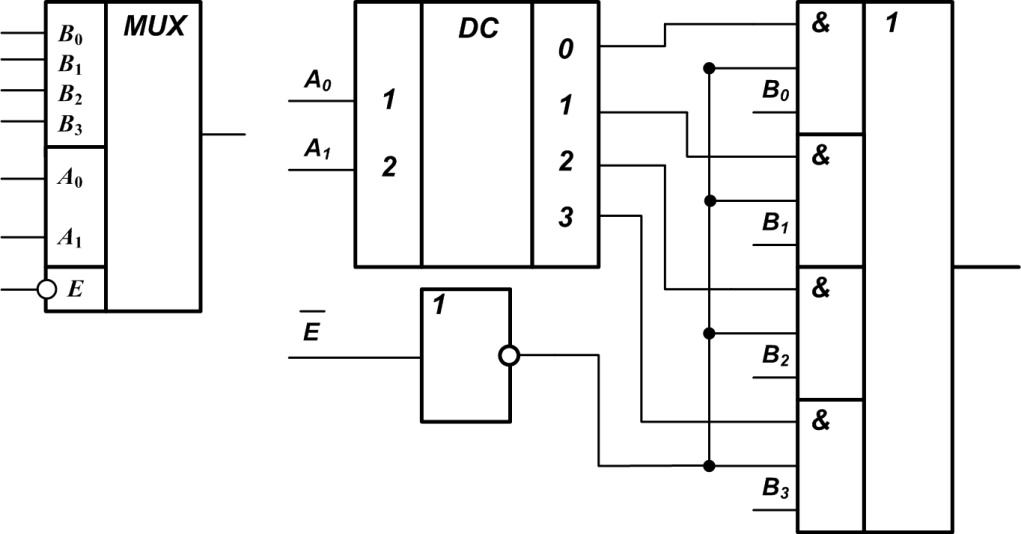


Рис. 5

В качестве примера рассмотренных модификаций схем мультиплексора на рис. 6 приведены микросхемы серии К155, реализующие функции мультиплексора.

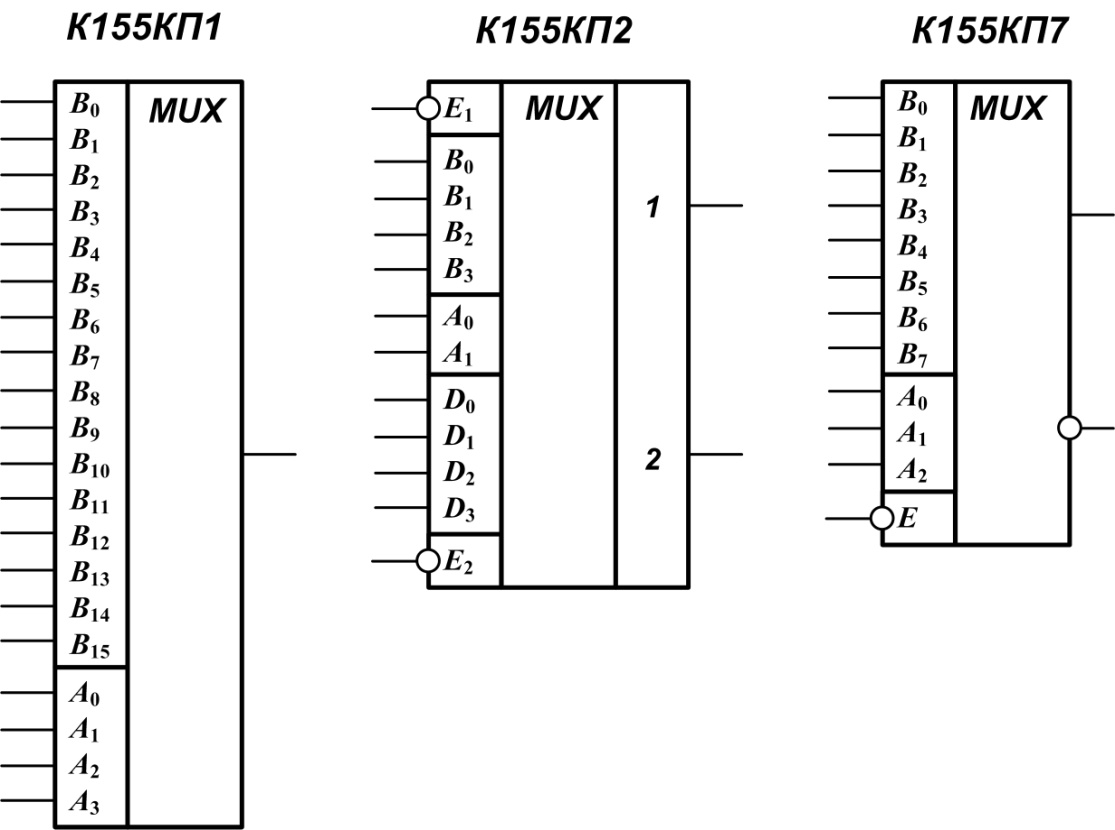


Рис. 6

Микросхема К155КП1 представляет собой мультиплексор, имеющий четыре адресных входа. Микросхема К155КП7 реализует мультиплексор с тремя адресными входами, разрешающим входом и парафазным выходом. А микросхема К155КП2 представляет собой два мультиплексора с двумя адресными входами. Адресные входы у этих мультиплексоров (АO и АI) общие, а разрешающие входы (Е1 и Е2) у каждого свой.

Вернемся к базовой схеме мультиплексора с двумя адресными входами (рис. 3). Функция, описывающая этот мультиплексор, имеет следующий вид:



Этой функции соответствует таблица истинности для выходного сигнала (*F*), представленная в табл. 4.

**Таблица 4.**

|  |  |  |
| --- | --- | --- |
| *А*1 | *А*0 | *F* |
| 0  0  1  1 | 0  1  0  1 | *B*0  *B*1  *B*2  *B*3 |

Очевидно, что, задавая значения Bi, т.е. подключая информационные входы мультиплексора к логическим константам "0" или "1", можно при помощи мультиплексора получить любую из 16 возможных функций двух переменных, поданных на его адресные входы.

Пусть требуется реализовать на мультиплексоре с двумя адресными входами, функцию "ИЛИ":

Закрепим переменные за адресными входами мультиплексора. Пусть, например, переменная *X*1 подана на вход *А*0 мультиплексора, а *Х*2 - на вход *А*1.

Составим с учетом обозначений входов мультиплексора, закрепленных за переменными, таблицу истинности для реализуемой функции (табл. 5).

**Таблица 5.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адресные входы и поданные на них переменные | | Значение функции | Информационные входы мультиплексора | |
| *А*1(*Х*2) | *А*0(*Х*1) | Обозначение | Сигнал |
| 0  0  1  1 | 0  1  0  1 | 0  1  1  1 | *B*0  *B*1  *B*2  *B*3 | 0  1  1  1 |

Схема, соответствующая полученной таблице, приведена на рис. 7.

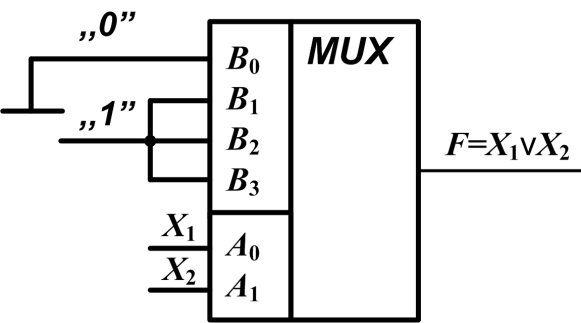


Рис. 7

Описанный выше способ позволяет реализовать на мультиплексоре с n адресными входами любую функцию от n переменных. Однако на практике больший интерес представляет реализация на мультиплексоре с тем же количеством адресных входов функций от большего, чем n числа переменных.

Пусть задана функция трех переменных *F* (*Х*1, *Х*2, *Х*3) которую требуется реализовать на мультиплексоре с двумя адресными входами. Для решения поставленной задачи производится функциональная декомпозиция исходной функции следующего вида:



При этом функции *F*1 *– F*4 могут принимать значения из следующего перечня: . Поэтому, если на адресные входы мультиплексора с двумя адресными входами подать переменные *Х*1 и *X*2 , а на информационные входа завести сигналы из приведенного перечня в соответствии с таблицей истинности функций, то на мультиплексоре с двумя адресными входами можно реализовать любую функцию трех переменных.

В качестве примера рассмотрим реализацию на мультиплексоре с двумя адресными входами следующей логической функции трех переменных:



Произведем закрепление переменных за адресными входами мультиплексора. Пусть, например, переменная *Х*1 подана на вход *А*1 мультиплексора, а переменная *Х*2 - на вход *А*0.

Составим с учетом обозначений входов мультиплексора, закрепленных за переменными, таблицу истинности для реализуемой функции (табл. 6).

**Таблица 6.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Адресные входы и поданные на них переменные | | Переменные, не поданные на адресные входы | Значение функции  *F* | Информационные входы мультиплексора | |
| *А*1(*Х*1) | *А*0(*Х*2) | *Х*3 | Обозначение | Сигнал |
| 0  0 | 0  0 | 0  1 | 0  1 | *B*0 | *Х*3 |
| 0  0 | 1  1 | 0  1 | 0  0 | *B*1 | 0 |
| 1  1 | 0  0 | 0  1 | 1  0 | *B*2 |  |
| 1  1 | 1  1 | 0  1 | 1  1 | *B*3 | 1 |

Строки полученной таблицы истинности разбивают на зоны постоянства значения сигналов на адресных входах мультиплексора. Каждой зоне постоянства соответствует информационный вход мультиплексора. Значение сигналов на информационных входах мультиплексора должно быть выражено через переменную, которая не подана на адресные входы мультиплексора, путем сопоставления значения этой переменной со значением функции в пределах зоны постоянства.

После определения значения сигналов на информационных входах мультиплексора может быть получена схема, реализующая заданную функцию (рис. 8).

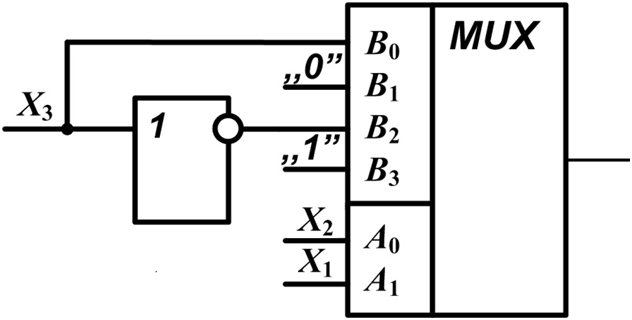


Рис. 8

Таким же образом, на базе мультиплексора с n адресными входами можно реализовать любую функцию от (n+1) переменной.

Аналогичный способ может быть применен также для реали­зации булевых функций в случае, если количество переменных в функции превышает количество адресных входов мультиплексо­ра более чем на единицу. Рассмотрим этот случай на примере реализации функции четырех переменных на мультиплексоре с двумя адресными входами: 

Выберем произвольно две переменные, которые подадим на адресные входы мультиплексора. Пусть переменная  будет подана на вход , а  - на вход . Так как функция со­держит 4 переменные, то количество переменных, не поданных на адресные входа мультиплексора (К), будет равно двум. Следо­вательно, зоны постоянства значения сигналов на адресных вхо­дах мультиплексора в соответствующей таблице истинности (табл. 7) должны содержать по 2к строк (в нашем случае 4 строки).

**Таблица 7.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Адресные входы и поданные на переменные  сигналы | | Переменные не поданные на адресные входы | | Значение функции | Информационные входы мультиплексора | | |
|  | |  |
|  |  |  |  | F | Обозначения | | Сигнал |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | |
| 0  0  0  0 | 0  0  0  0 | 0  0  I  I | 0  I  0  I | 0  0  I  I |  |  | |
| 0  0  0  0 | I  I  I  I | 0  0  I  I | 0  I  0  I | I  0  I  I |  |  | |
| I  I  I  I | 0  0  0  0 | 0  0  I  I | 0  I  0  I | 0  I  0  I |  |  | |
| I  I  I  I | I  I  I  I | 0  0  I  I | 0  I  0  I | I  I  0  I |  |  | |

Значение сигнала на i - ом информационном входе мультиплексора можно получить либо путем сопоставления значения функции со значениями переменных, которые не поданы на адрес­ное входы мультиплексора (как в предыдущем примере), либо путем подстановки в исходное выражение функции значений пере­менных на адресных входах мультиплексора, которые соответствуют i - му информационному входу.

Например, на вход  мультиплексора мы должныподатьследующеезначение сигнала:

**

Для входа  мультиплексора выражение, определяющее значение сигнала будет иметь следующий вид:

*.*

Таким образом, для реализации на мультиплексоре с n*.* адресными входами функции, содержащей более чем (n+1) переменную, могут потребоваться дополнительные логические элементы.

Количество этих элементов может быть уменьшено или даже сведено к нулю путем проработки других вариантов схемы, связанных с подачей на адресные входы мультиплексора другого сочетания переменных.

Например, если на адресный вход  подать переменную , а на вход -переменную  то таблица истинности для ранее рассмотренной функции примет вид, приведенной в табл. 8.

**Таблица 8.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Адресные входы мультиплексора и поданные на переменные | | Переменные, не поданные на адресные входы | | Значения функции  F | Информационные входы мультиплексоры | |
|  |  |  |  | Обозначение | сигнал |
| 0  0  0  0 | 0  0  0  0 | 0  0  I  I | 0  I  0  I | 0  0  I  I |  |  |
| 0  0  0  0 | I  I  I  I | 0  0  I  I | 0  I  0  I | I  0  I  0 |  |  |
| I  I  I  I | 0  0  0  0 | 0  0  I  I | 0  I  0  I | 0  I  0  I |  |  |
| I  I  I  I | I  I  I  I | 0  0  I  I | 0  I  0  I | I  I  I  I |  | 1 |

Из полученной таблицы видно, что заданная функция может быть реализована на мультиплексоре с двумя адресными входами без применения дополнительных логических элементов.

Если перебор вариантов реализации функции на выбран­ном мультиплексоре покажет, что без дополнительных логических элементов заданную функцию реализовать невозможно, то в слу­чае наличия у мультиплексора разрешающего входа следует по­пытаться преобразовать заданную функцию, вынеся за скобки одну из ее переменных. В этом случае вынесенная за скобки переменная подается на разрешающий вход мультиплексора, а вы­ражение в скобках реализуется на мультиплексоре как самосто­ятельная функция от меньшего числа переменных.

Рассмотрим этот случай на примере. Пусть требуется реа­лизовать на мультиплексоре с двумя адресными входами и раз­решающим входом следующую функцию от пяти переменных:



Данную функцию можно преобразовать, вынеся за скобки переменную . При этом выражение примет следующий вид: 

Реализация функции четырех переменных, которая приведена в скобках, на мультиплексоре с двумя адресными входами была рассмотрена ранее (ей соответствует табл. 8). Исхода из этого, комбинационная схема, реализующая функцию F , должна иметь вид, представленный на рис. 9.

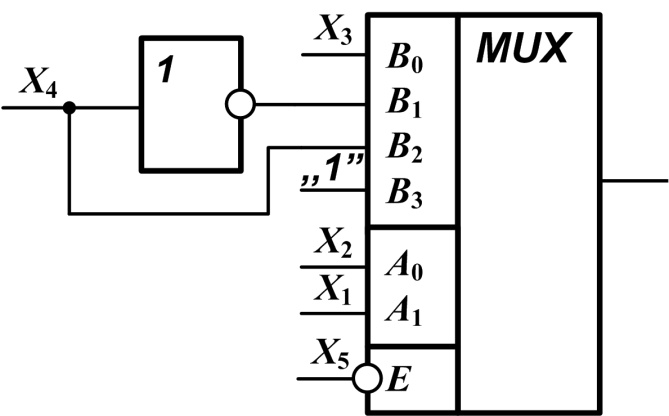


Рис. 9

В случае, если невозможно преобразовать заданную функцию рассмотренным выше образом, то для реализации функции следует выбрать мультиплексор с большим количеством адресных входов и повторить на нем процедуру синтеза комбинационной схемы. Если же в составе используемых при проектировании микросхем отсутствует мультиплексор с необходимым количеством адресных входов, то для решения задачи может быть предложено два различных метода.

Первый метод состоит в том, что из рассмотренных ранее вариантов реализации комбинационной схемы выбирается тот вариант, который обеспечивал бы использование минимального количества дополнительных логических элементов.

Второй метод решения задачи связан в построении мультиплексора с необходимым для реализации функции количеством адресных входов на основе микросхем, реализующих мультиплек­соры с количеством адресных входов меньшим, чем необходимо. Рассмотрим, как строятся на основе мультиплексо­ров, имеющих n адресных входов, мультиплексоры с (n+1)адресным входом. Для решения данной задачи используются два способа.

Первый способ связан с применением разрешающего входа у мультиплексора. В качестве примера, иллюстрирующего данный способ, может быть рассмотрена схема, представленная на рис. 10, на которой показана реализация мультиплексора с тремя адресными входами на базе двух мультиплексоров с двумя адресными входами и разрешающим входом.

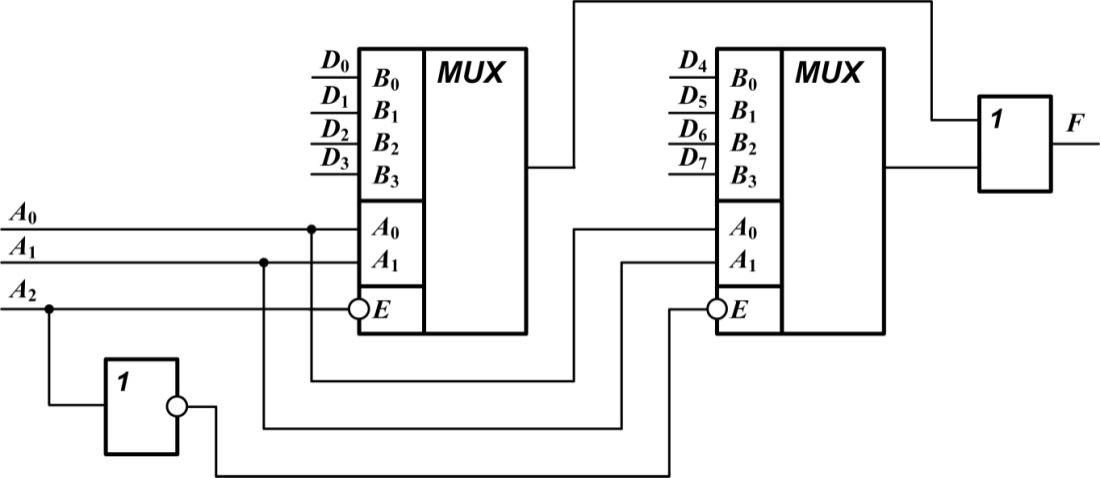


Рис. 10

При использовании данного способа одноименные адресные входы используемых мультиплексоров объединяются, а на их разрешающе входы подаются взаимоинверсные сигналы.

Второй способ позволяет использовать мультиплексоры без разрешающего входа. В качестве примера для иллюстрации этого способа может быть рассмотренасхема, представленная на рис. 11, на которой показана реализация мультиплексора с тремя адресными входами на базе двух мультиплексоров с двумя адресными входами.



Рис. 11

При использовании данного способа, как и в первом способе, одноименные адресные входы мультиплексоров, на базе которых строится схема, объединяются.

При необходимости построения на базе мультиплексоров, имеющих n адресных входов мультиплексора, количество адрес­ных входов которого превышает (n+1), применяются многокаскадные схемы. При этом, в зависимости от наличия у базового мультиплексора разрешающего входа, также может быть применено два способа построения схемы.

Первый способ связан с применением разрешающего входа мультиплексора и может быть проиллюстрирован схемой, пред­ставленной на рис. 12.

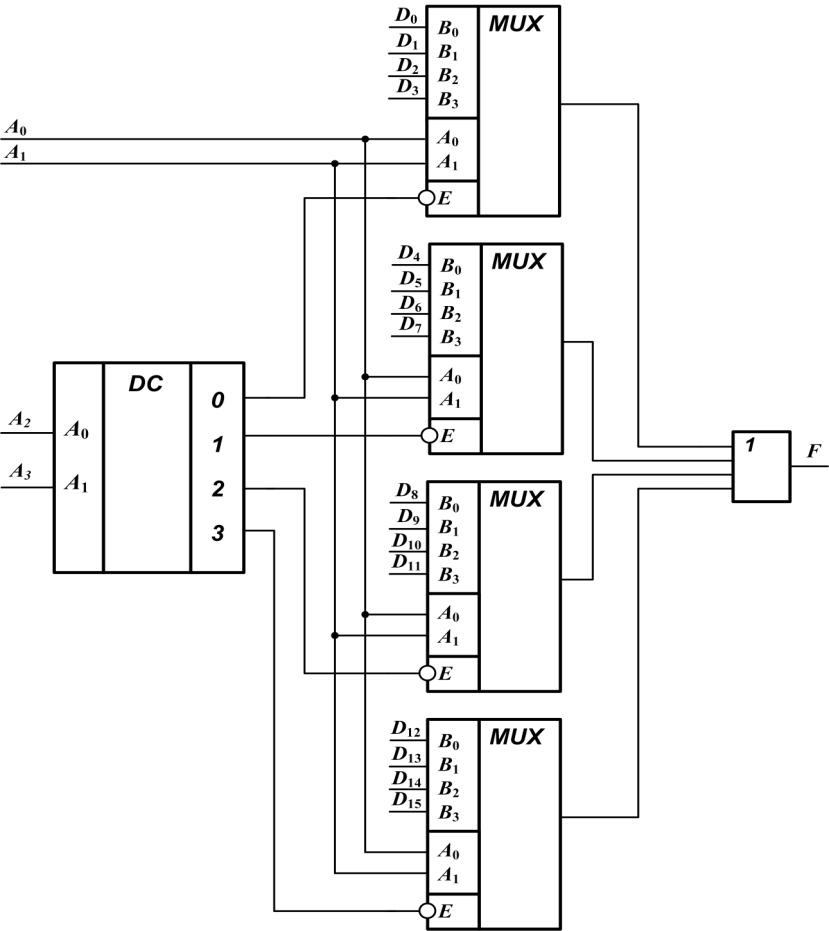


Рис. 12

На данном рисунке показана схема мультиплексора на четыре адресных входа, построенная на базе дешифратора и четырех мультиплексоров с двумя адресными входами и разре­шающим входом. Аналогичным способом, путем использования мультиплексоров и дешифраторов с различным количеством адрес­ных входов, могут быть построены схемы мультиплексоров с любым количеством адресных входов.

Второй способ построения многоярусных схем мультиплексо­ра обеспечивает применение для этой цели мультиплексора без разрешающего входа. Данный способ может быть проиллюстрирован схемой, представленной на рис. 13.

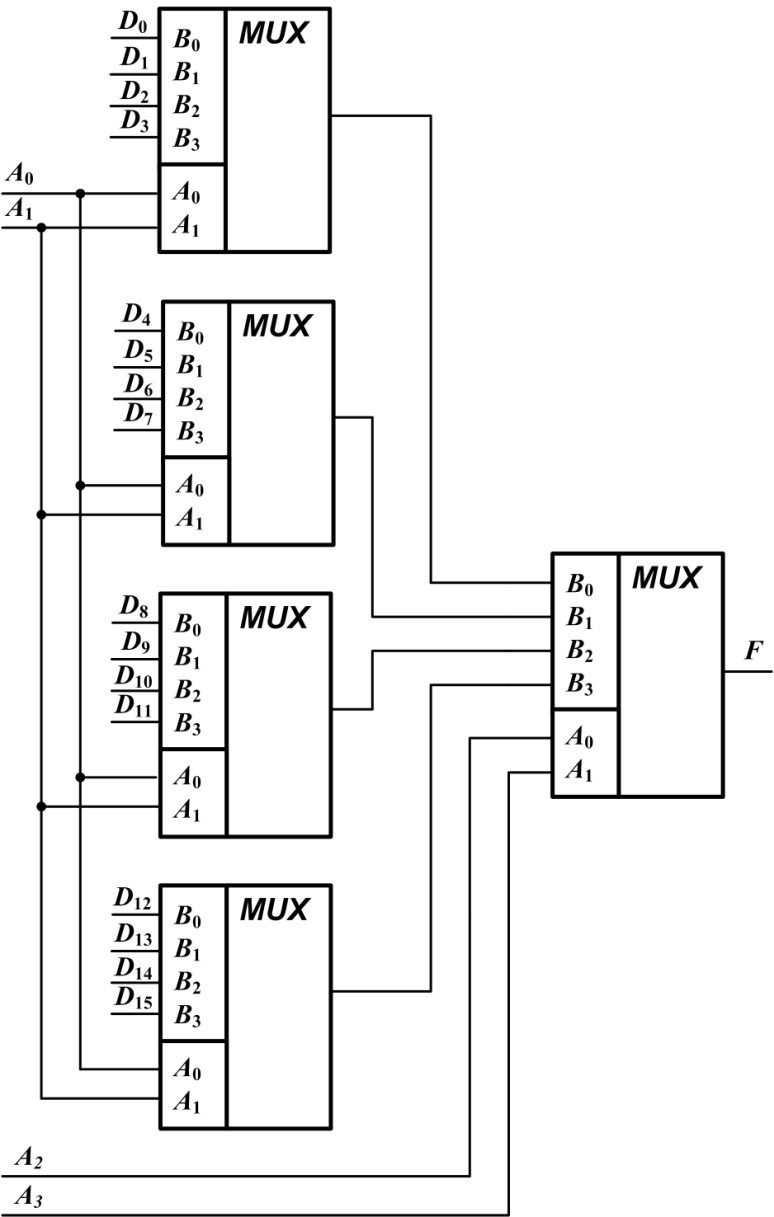


Рис. 13

На данном рисунке показана схема мультиплексора на четы­ре адресных входа, построенная на базе мультиплексоров с двумя адресными входами. Аналогичным способом путем использо­вания мультиплексоров с различным количеством адресных входов могут быть построены схемы мультиплексоров с любым количест­вом адресных входов.

Выбор способа построения мультиплексора с заданным коли­чеством адресных входов на базе мультиплексоров с меньшим количеством адресных входов зависит от состава используемой для построения схемы серии микросхем и от требований, предъ­являемых к схеме.

Окончательный выбор метода построения комбинационной схемы производится после сравнения оптимальной комбинационной схемы с использованием мультиплексора и дополнительных логи­ческих элементов с комбинационной схемой, полученной на синтезированной схеме мультиплексора с повышенным количеством адресных входов.

**Подготовка к выполнению работы**

1. Изучить методику синтеза комбинационных схем на базе мультиплексора.

2. Изучить приведенный ниже примерсинтеза комбинацион­ныхсхем.

3. Используя мультиплексор с двумя адресными входами, синтезировать на базе микросхем серии К155, приведенных в Приложении 4, комбинационную схему, реализующую соответствующие заданному номеру варианта задания функции  и , представленные в табл. 1.

4. Реализовать те же функции при помощи мультиплексоров с тремя адресными входами.

5. Построить модели синтезированных схем (для каждой схемы должен быть создан индивидуальный проект) при помощиСАПР Altera Quartus II (см. Приложение 2).

6. Получить временные диаграммы работы моделей синтезированных схем путем моделирования их работы в режимах «functional» и «timing». Порядок изменения входных сигналов схемы должен соответствовать таблице 3, а минимальная длительность входного сигнала должна быть подобрана таким образом, чтобы задержки схемы, наблюдаемые в режиме «timing» составляли не более 30% длительности входного сигнала.

7. Сопоставить результаты моделирования с заданием на лабораторную работу.

8. Составить отчет о подготовке к лабораторной работе (в формате WORD) и выслать преподавателю для проверки (вместе с проектами).

9. Распечатать проверенный и одобренный преподавателем отчет.

10. Ответить на контрольные вопросы.

Примечание:

а) при синтезесхемы должен быть осуществлен выбор такого варианта схемы, который обеспечивал бы минимальные аппаратные затраты;

б) при синтезе схемы допустимо использование, кроме ука­занных выше мультиплексоров, также логических элементов, которые входят в состав серии К155 и приведены в приложении 4;

в) при изображении схем необходимо использовать стандартные условные обозначения элементов с обязательным указанием их порядковых номеров в микросхеме и типа микросхемы.

г) студенты, не выполнившие подготовку в полном объеме, к лабораторной работе не допускаются.

**Пример синтеза комбинационных схем на базе мультиплексоров**

Пусть заданы функции четырех переменных  и  в форме перечня десятичных номеров конституент единицы (табл. 9), что соответствует варианту 31 зада­ния на лабораторную работу.

**Таблица 9.**

|  |  |  |
| --- | --- | --- |
| Функция |  |  |
| Перечень десятичных номеров конституент единицы | 4,5,6,7,9,12,13,15 | 0,1,9,15 |

Произвольно выберем две переменные, которые будем пода­вать на адресные входы мультиплексора с двумя адресными входами, и составим таблицу истинности, соответствующую сде­ланному выбору (табл. 10).

Определив и проанализировав значения сигналов, которые необходимо подать на информационные входы мультиплексора, можно сделать вывод, что при выбранном сочетании сигналов на адресных входах мультиплексора для реализации комбинационной схемы дополнительно потребуется три логических элемента и три инвертора. Это приведет к тому, что общие аппаратные затраты, необходимые для реализации схемы, составят три корпуса микросхем.

**Таблица 10**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Адресные входы мультиплексора и поданные на них переменные | | Переменные, не поданные на адресные входы мультиплек-сора | | Значение функции | Информационные входы  I- го мультиплексора | | Значение функции | Информационные входы  2- го мультиплексора | |
|  |  |  |  | Обозна-чения | Сигнал | Обозна-чения | Сигнал |
| 0  0  0  0 | 0  0  0  0 | 0  0  I  I | 0  I  0  I | 0  0  0  0 |  | 0 | I  I  0  0 |  |  |
| 0  0  0  0 | I  I  I  I | 0  0  I  I | 0  I  0  I | I  I  I  I |  | I | 0  0  0  0 |  | 0 |
| I  I  I  I | 0  0  0  0 | 0  0  I  I | 0  I  0  I | 0  I  0  0 |  |  | 0  I  0  0 |  |  |
| I  I  I  I | I  I  I  I | 0  0  I  I | 0  I  0  I | I  I  0  I |  |  | 0  0  0  I |  |  |

Анализ функций показывает, что они не могут быть пред­ставлены в форме, позволяющей использовать разрешающие входы мультиплексоров для подачи на них какой-либо из переменных. По этой причине необходимо рассмотреть остальные варианты реа­лизации комбинационной схемы, которые связаны с подачей на адресные входы мультиплексоров других сочетаний переменных. Всего таких вариантов пять: [[1]](#footnote-1)\* Анализ указанных вариантов позволяет сделать вывод, что ни один из них не обеспечивает реализацию комбинационной схемы без дополнительных логических элементов, а минимальные аппаратные затраты достигаются при подаче на адресные входы мультиплексора переменных и . Таблица истинности, соответствующая этому варианту, приведена в табл. 11.

**Таблица 11.**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Адресные входы мультиплексора и поданные на них переменные | | Переменные, не поданные на адресные входы мультиплексора | | Значение функции | Информационные входы  I- го мультиплексора | | Значение функции | Информацион-ные входы  2- го мультиплексора | |
|  |  |  |  | Обозна-чения | Сигнал | Обозна-чения | Сигнал |
| 0  0  0  0 | 0  0  0  0 | 0  0  I  I | 0  I  0  I | 0  0  0  1 |  |  | I  I  0  I |  |  |
| 0  0  0  0 | I  I  I  I | 0  0  I  I | 0  I  0  I | I  I  I  I |  | I | 0  0  0  0 |  | 0 |
| I  I  I  I | 0  0  0  0 | 0  0  I  I | 0  I  0  I | 0  0  0  0 |  | 0 | 0  0  0  0 |  | 0 |
| I  I  I  I | I  I  I  I | 0  0  I  I | 0  I  0  I | I  I  0  I |  |  | 0  0  0  I |  |  |

Проанализировав значения сигналов, которые необходимо подать на информационные входы мультиплексора, можно сделать вывод, что при выбранном сочетании сигналов на адресных вхо­дах мультиплексоров для реализации комбинационной схемы дополнительно потребуется два логических элемента и два ин­вертора. По этой причине общие аппаратные затраты, необходи­мые для реализации схемы, составят два корпуса микросхем. Полученные результаты позволяют сделать вывод, что при­менение метода, который связан с синтезом мультиплексора с тремя адресными входами на базе заданного мультиплексора с двумя адресными входами, в данном случае не рационально, так как это требует аппаратных затрат в объеме четырех корпусов микросхем.

Таким образом, оптимальный вариант комбинационной схемы, реализующей заданные функции на базе мультиплексора с двумя адресными входами, соответствует данным из табл. 11. Схема соответствующая этому варианту реализации, приведена на рис. 14 .

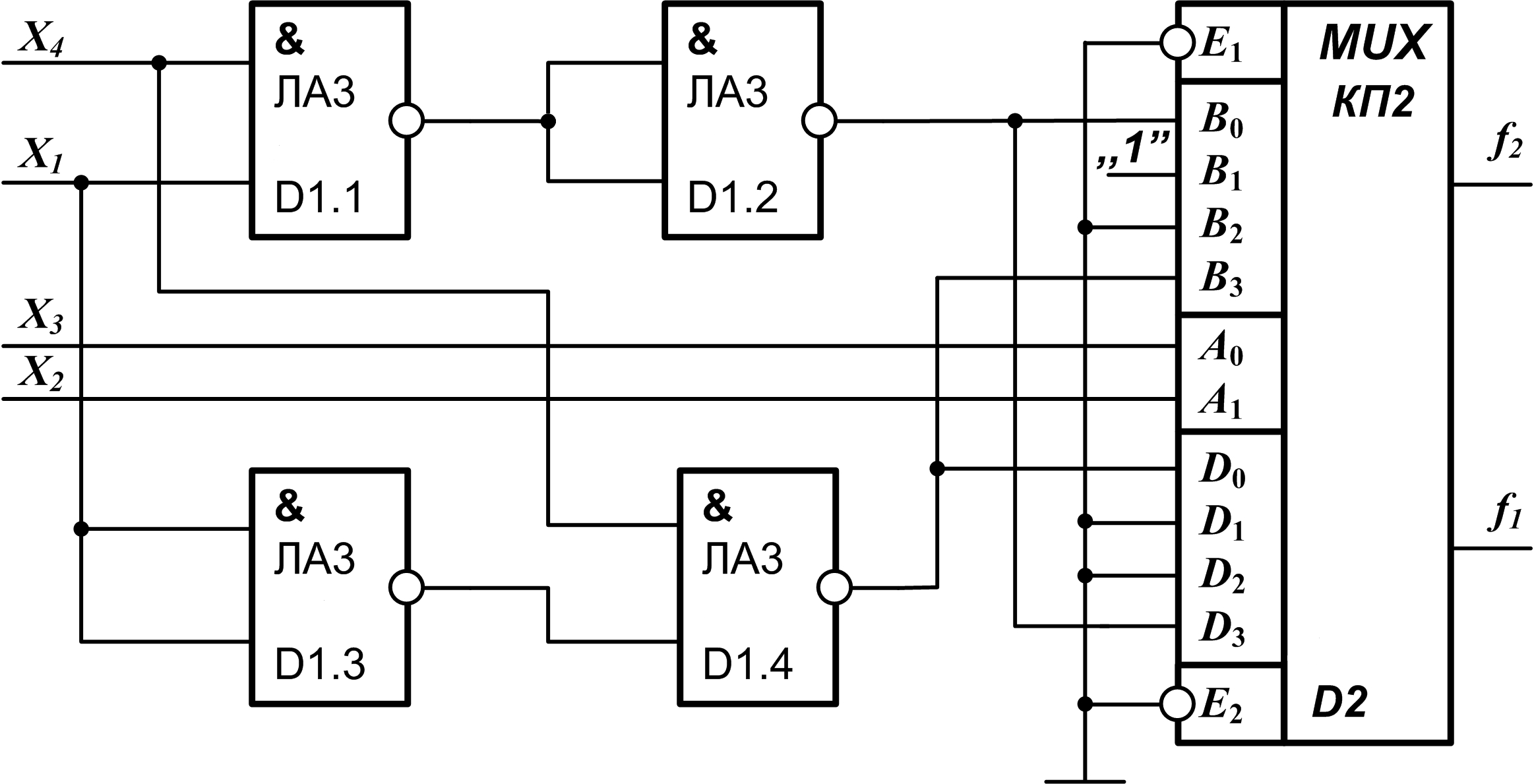
****

Рис. 14

Теперь произведем синтез комбинационной схемы, реализую­щей заданные функции на базе мультиплексора с тремя адресными входами.

Произвольно выберем три переменные, которые будем пода­вать на адресные входы мультиплексоров, и составим таблицы истинности для функций  (табл. 12) и  (табл. 13), кото­рые соответствуют сделанному выбору. Определив и проанализировав значения сигналов, которые необходимо подать на информационные входы мультиплексоров, можно сделать вывод, что при выбранном сочетании сигналов на адресных входах мультиплексоров для реализации комбинацион­ной схемы не потребуются дополнительные инверторы, что снима­ет необходимость рассмотрения других вариантов схемы и позволяет считать схему, полученную на основе данных, приведенных в табл. 12 и табл. 13 (рис.15), оптимальной.

**Таблица 12**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Адресные входы мультиплексора и поданные на них переменные | | | Переменная, не поданная на адресные входы | Значение  Функции | Информационные входы | |
| Обозначение | Сигнал |
|  |  |  |  |
| 0  0 | 0  0 | 0  0 | 0  I | 0  0 |  | 0 |
| 0  0 | 0  0 | I  I | 0  I | 0  0 |  | 0 |
| 0  0 | I  I | 0  0 | 0  I | I  I |  | I |
| 0  0 | I  I | I  I | 0  I | I  I |  | I |
| I  I | 0  0 | 0  0 | 0  I | 0  I |  |  |
| I  I | 0  0 | I  I | 0  I | 0  0 |  | 0 |
| I  I | I  I | 0  0 | 0  I | I  I |  | I |
| I  I | I  I | I  I | 0  I | 0  I |  |  |

**Таблица 13**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Адресные входы мультиплексора и поданные на них переменные | | | Переменная, не поданная на адресные входы | Значение  Функции | Информационные входы | |
| Обозначение | Сигнал |
|  |  |  |  |
| 0  0 | 0  0 | 0  0 | 0  I | I  I |  | I |
| 0  0 | 0  0 | I  I | 0  I | 0  0 |  | 0 |
| 0  0 | I  I | 0  0 | 0  I | 0  0 |  | 0 |
| 0  0 | I  I | I  I | 0  I | 0  0 |  | 0 |
| I  I | 0  0 | 0  0 | 0  I | 0  I |  |  |
| I  I | 0  0 | I  I | 0  I | 0  0 |  | 0 |
| I  I | I  I | 0  0 | 0  I | 0  0 |  | 0 |
| I  I | I  I | I  I | 0  I | 0  I |  |  |

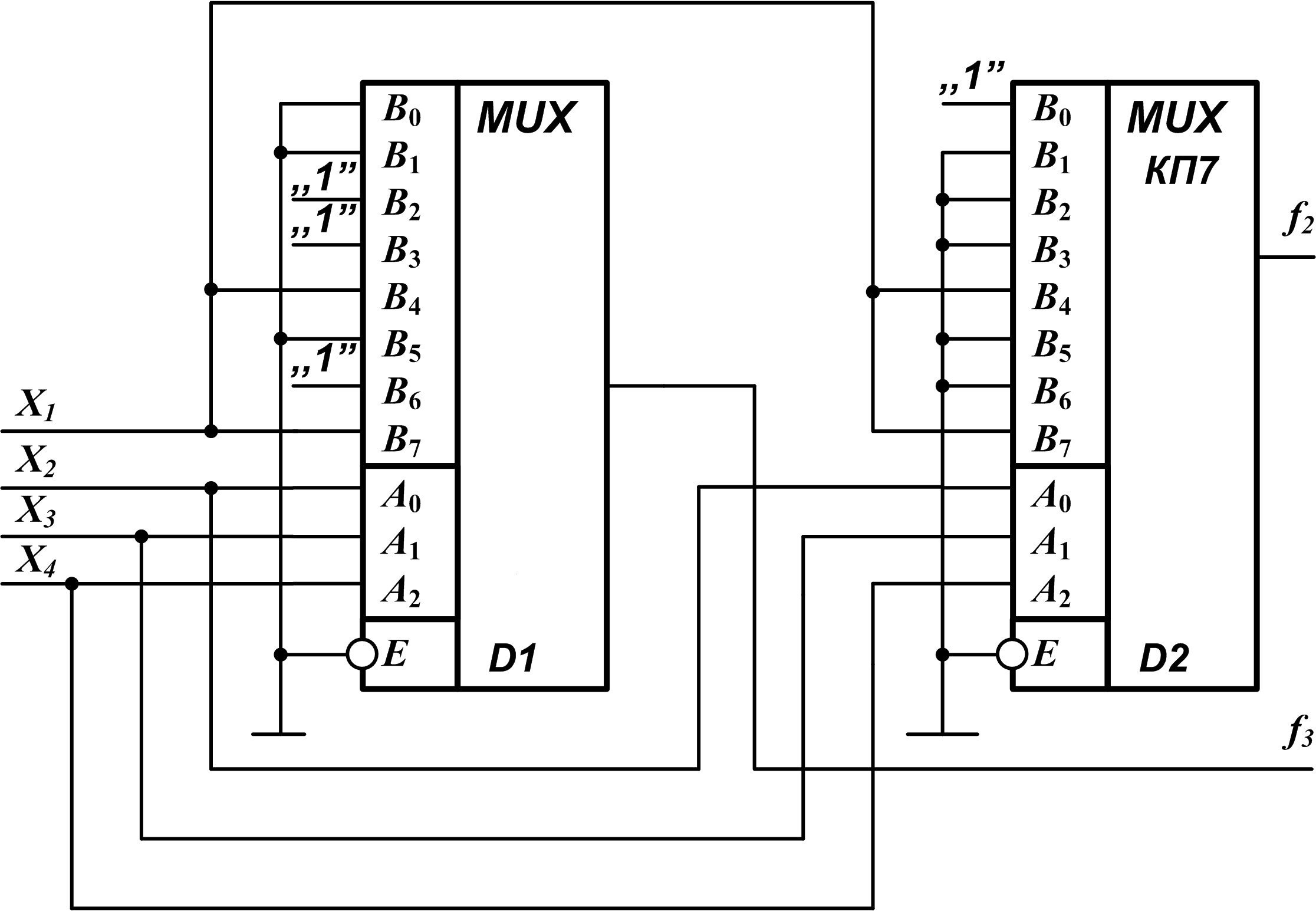
**

Рис. 15

Аппаратные затраты, необходимые для реализации этой схемы составляют два корпусамикросхем**.**

Таким образом, оба варианта реализации комбинационной схемы, выполненные на мультиплексорах с различным числом адресных входов, предполагают одинаковые аппаратные затраты. Однако они отличаются по другим параметрам, таким, как быстро­действие, надежность, потребляемая мощность и др. Это позволя­ет выбрать из них тот вариант, который наиболее полно отвеча­ет поставленным при проектировании требованиям.

**Порядок выполнения работы**

1. Представить преподавателю письменный отчет о подготовке к лабораторной работе, ответить на предложенные им контрольные вопросы и получить допуск к выполнению лабораторной работы.

2. Перенести на предоставленный компьютер разработанные проекты моделей комбинационных схем и убедиться в их работоспособности путем контрольного моделирования работы этих схем в режиме «timing».

3. Произвести (см. Приложение 3) закрепление входных и выходных сигналов комбинационной схемы, построенной на мультиплексорах с 2-мя адресными входами за элементами предоставленного стенда (входные сигналы «Х1», «Х2», «Х3» и «Х4» должны поступать с тумблеров стенда, а выходы схемы должны поступать на светодиодные индикаторы стенда).

4. Осуществить прошивку проекта комбинационной схемы, построенной на мультиплексорах с 2-мя адресными входами на плату стенда.

5. Произвести проверку макета комбинационной схемы, построенной на мультиплексорах с 2-мя адресными входами в статическом режиме. Для этого:

а) убедиться, что при всех значениях входных сигналов «Х1», «Х2», «Х3» и «Х4» полученные значения функций совпадает с заданным;

б) в случае неправильной работы макета комбинационной схемы, построенной на мультиплексорах с 2-мя адресными входами ввести необходимые изменения в модель комбинационной схемы, построенной на мультиплексорах с 2-мя адресными входами и повторить выполнение п.п. «а» - «б».

в) продемонстрировать работу комбинационной схемы, построенной на мультиплексорах с 2-мя адресными входами преподавателю.

6. Произвести (см. Приложение 2) закрепление входных и выходных сигналов комбинационной схемы, построенной на мультиплексорах с 3-мя адресными входами за элементами стенда (входные сигналы «Х1», «Х2», «Х3» и «Х4» должны поступать с тумблеров стенда, а выходы схемы должны поступать на светодиодные индикаторы стенда).

7. Осуществить прошивку проекта комбинационной схемы, построенной на мультиплексорах с 3-мя адресными входами на плату стенда.

8. Произвести проверку макета комбинационной схемы, построенной на мультиплексорах с 3-мя адресными входами в статическом режиме. Для этого:

а) убедиться, что при всех значениях входных сигналов «Х1», «Х2», «Х3» и «Х4» полученные значения функций совпадают с заданным;

б) в случае неправильной работы макета комбинационной схемы, построенной на мультиплексорах с 3-мя адресными входами ввести необходимые изменения в модель многовыходной комбинационной схемы и повторить выполнение п.п. «а» - «б».

в) продемонстрировать работу комбинационной схемы, построенной на мультиплексорах с 3-мя адресными входами преподавателю.

9. Получив разрешение преподавателя, осуществить выключение стенда.

**Содержание отчета**

I. Материалы по синтезу комбинационной схемы на базе мультиплексора с двумя адресными входами, содержащие:

а) все возможные варианты реализации комбинационной схемы;

б) сравнительный анализ по аппаратным затратам приведенных вариантов реализации и обоснование выбора оптимального варианта (при одинаковом количестве корпусов микросхем выбирают тот вариант, в котором в этих корпусах имеется максимальное количество выводов у неиспользованных логических элементов);

в) схему, соответствующую выбранному варианту.

2. Материалы по синтезу комбинационной схемы на базе мультиплексора с тремя адресными входами, содержащие:

а) все возможные варианты реализации комбинационной схемы**;**

б) сравнительный анализ приведенных вариантов реализации и обоснование выбора оптимального варианта;

в) схему, соответствующую выбранному варианту.

3. Сравнительный анализ комбинационныхсхем**,** полученных на базе мультиплексоров с двумя и тремя адресными входами, и выбор наилучшей из них аппаратным затратам (при одинаковом количестве корпусов микросхем выбирают тот вариант, в котором в этих корпусах имеется максимальное количество выводов у неиспользованных логических элементов).

4. Сравнительный анализ комбинационнойсхемы**,** разрабо­танной на базе мультиплексоров, и комбинационной схемы, разра­ботанной на базе логических элементов при выполнении первой работы по аппаратным затратам (при одинаковом количестве корпусов микросхем выбирают тот вариант, в котором в этих корпусах имеется максимальное количество выводов у неиспользованных логических элементов).

5. Модели разработанных схем.

6. Временные диаграммы работы синтезированных комбинационных схем, полученные путем моделирования их работы в режимах «functional» и «timing».

**Контрольные вопросы**

I. Какое количество адресных входов должен иметь мульти­плексор, чтобы на нем можно было реализовать любую функцию от n переменных без дополнительных логических элементов?

2. Какие имеются модификации схемы мультиплексора?

З. Как выглядит базовая функциональная схема мультиплексора?

4. Каково основное назначение мультиплексоров?

5. Как можно использовать разрешающий вход мультиплексо­ра при синтезе комбинационных схем?

6. Какие имеются способы построения мультиплексоров с n адресными входами на базе мультиплексоров с (n-1) адресным входом?

7. Какие имеются способы построения мультиплексоров с n адресными входами на базе мультиплексоров, имеющих менее чем (n-1) адресный вход?

**ЛАБОРАТОРНАЯ РАБОТА N 3**

**СИНТЕЗ И ИССЛЕДОВАНИЕ СИНХРОННЫХ ДВУХСТУПЕНЧАТЫХ ТРИГГЕРОВ**

**Цель лабораторной работы**

Изучение различных схем триггеров и овладение методами логического проектирования структуры синхронного двухступенчатого триггера; получение навыков в моделировании, макетировании, наладке и экспериментальном исследовании триггеров.

**Общие сведения о двухступенчатых триггерах**

Триггером называется простейший автомат Мура с двумя устойчивыми состояниями, служащий для приема и хранения одного бита информации.

В общем случае триггер должен содержать элемент памяти и схему управления им (рис. 16), на которую поступают вход­ные сигналы и сигналы обратной связи с элемента памяти.



Рис. 16

Состояние триггера (Q) определяется состоянием его эле­мента памяти, в качестве которого используется асинхронный RS -триггер.

Одной из наиболее распространенных разновидностей триггеров являются двухступенчатые синхронные триггеры. Особен­ностью этих триггеров является то, что их схема управления имеет сложную структуру и содержит в своем составе запоминающий элемент.

Общий вид структурной схемы двухступенчатого триггера представлен на рис. 17. Управляющая схема триггера выделена на рисунке пунктирной линией.

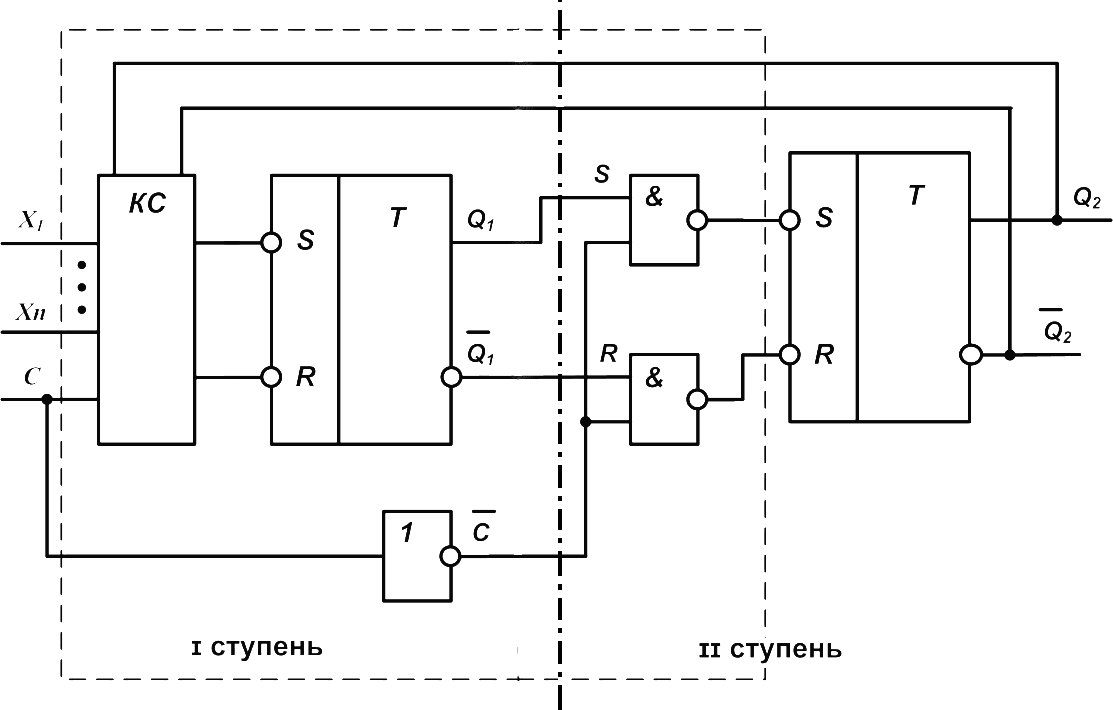


Рис. 17

Любой триггер должен выполнять две основные функции: прием информации и хранение информации. В двухступенчатом триггере эти функции разделены между первой и второй ступенями, функцию хранения информации выполняет вторая (вспомогательная) ступень (справа от жирной штрих-пунктирной разделительной линии на рис.17), которая представляет собой синхронный одноступенча­тый RS -триггер. Функциональная схема второй ступени, раскры­вающая структуру асинхронного RS -триггера, и ее функциональ­ное обозначение приведены на рис.18.

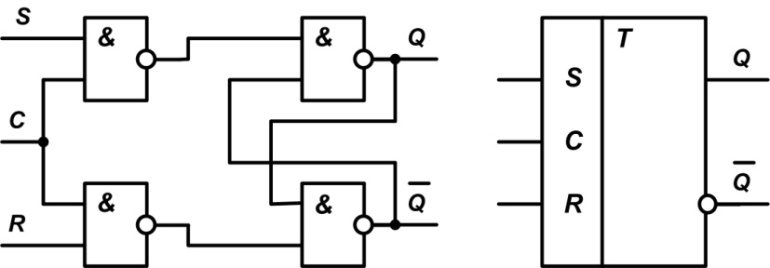


Рис. 18

Используя приведенное выше функциональное обозначение синхронного одноступенчатого триггера, можно представить обобщенную структурную схему двухступенчатого триггера в более наглядном виде (рис. 19).

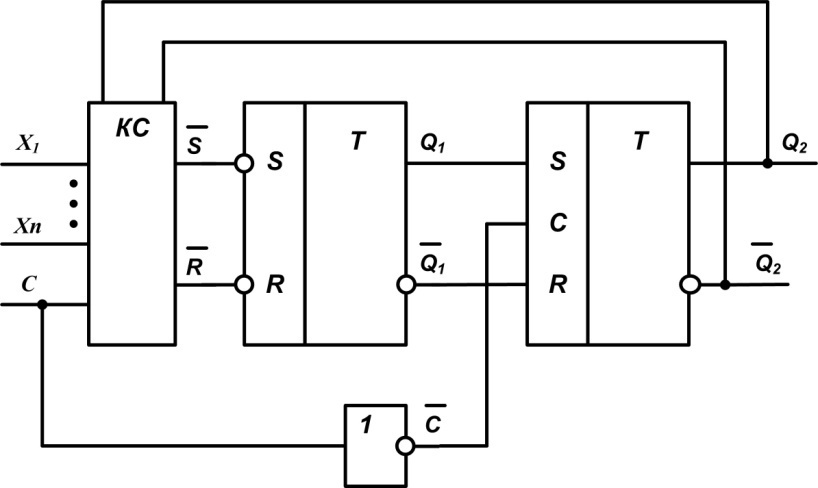


Рис. 19

Первая (главная) ступень двухступенчатого триггера обес­печивает прием информации в триггер. Прием информации происхо­дит при единичном значении внешнего синхронизирующего сигнала "С". При этом, благодаря имеющемуся в схеме инвертору, на синхровход триггера второй ступени подается нулевое значение сигнала. Таким образом, во время приема информации вторая ступень сохраняет старое состояние двухступенчатого триггера.

По истечении времени, достаточного для срабатывания пер­вой ступени, ее триггер устанавливается в новое состояние и запоминает его. Если теперь сбросить в ноль входной синхро­сигнал, то на синхровходе триггера второй ступени появится единичный сигнал, по которому состояние триггера первой сту­пени (Q1) перепишется во вторую ступень. Таким образом, новое состояние на выводе двухступенчатого триггера формируется по заднему фронту синхросигнала "С".

Двухступенчатые триггеры часто называют MS -триггерами (от англ. слов master и slave - хозяин и раб). Это название отражает наличие у триггера главной и вспомогательной ступени.

В виде двухступенчатого триггера могут применяться триг­геры любого функционального типа, но наибольшее распростране­ние получил двухступенчатый JK -триггер, на примере которого ниже рассмотрена методика синтеза двухступенчатых триггеров.

Эта методика относительно проста, так как из структуры двухступенчатого триггера вытекает, что синтез двухступенча­того триггера любого функционального типа сводится к синтезу комбинационной схемы его первой ступени.

**Подготовка к выполнению работы**

1. Изучить классификацию триггеров и законы функциониро­вания триггеров различных функциональных типов.

2. Изучить структуру и принцип работы двухступенчатого триггера.

3. Рассмотреть приведенный ниже пример синтеза двухсту­пенчатого триггера.

4. Согласно заданному номеру варианта (табл. 14) синте­зировать на основе логических элементов серии К155 схему синхронного двухступенчатого триггера со входами асинхронной установки в «0» и «1», обеспечив минимум аппаратных затрат.

5. Вычертить схему разработанного триггера с обязатель­нымуказанием для каждого элемента схемы номера соответствую­щегоему элемента и типа микросхемы серии К155 (аналогичную приведенной на рис 20).

6. Построить модель синтезированной схемы при помощиСАПР Altera Quartus II (см. Приложение 2).

7. Получить временные диаграммы работы модели синтезированной схемы путем моделирования её работы в режимах «functional» и «timing». Порядок изменения сигналов возбуждения триггера должен быть подобран таким образом, чтобы на временных диаграммах были видны все возможные переходы его состояния. Необходимо также учесть, что изменение сигналов возбуждения триггера во время единичного значения синхросигнала недопустимо. Минимальная длительность входного сигнала должна быть подобрана таким образом, чтобы задержки схемы, наблюдаемые в режиме «timing» составляли не более 30% длительности входного сигнала.

8. Сопоставить результаты моделирования с заданием на лабораторную работу.

9. Составить отчет о подготовке к лабораторной работе (в формате WORD) и выслать преподавателю для проверки (вместе с проектом).

10. Распечатать проверенный и одобренный преподавателем отчет.

11. Ответить на контрольные вопросы.

**Таблица 14**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № варианта | 0 | I | 0 | I |  |
| 0 | 0 | I | I |  |
| 1 | Q(t) | I | 0 | Q(t) | Q(t+1) |
| 2 | 0 | 0 | I | Q(t) | Q(t+1) |
| 3 | 0 | Q(t) | ‾Q(t) | I | Q(t+1) |
| 4 | Q(t) | 0 | ‾Q(t) | I | Q(t+1) |
| 5 | Q(t) | 0 | I | 0 | Q(t+1) |
| 6 | 0 | Q(t) | I | I | Q(t+1) |
| 7 | 0 | 0 | Q(t) | I | Q(t+1) |
| 8 | I | ‾Q(t) | Q(t) | 0 | Q(t+1) |
| 9 | Q(t) | Q(t) | I | 0 | Q(t+1) |
| 10 | I | Q(t) | Q(t) | 0 | Q(t+1) |
| 11 | Q(t) | 0 | I | I | Q(t+1) |
| 12 | I | Q(t) | I | 0 | Q(t+1) |
| 13 | I | 0 | Q(t) | 0 | Q(t+1) |
| 14 | I | I | 0 | Q(t) | Q(t+1) |
| 15 | 0 | I | Q(t) | Q(t) | Q(t+1) |
| 16 | I | Q(t) | 0 | Q(t) | Q(t+1) |
| 17 | I | Q(t) | ‾Q(t) | 0 | Q(t+1) |
| 18 | Q(t) | I | 0 | 0 | Q(t+1) |
| 19 | 0 | Q(t) | 0 | I | Q(t+1) |
| 20 | I | I | Q(t) | 0 | Q(t+1) |
| 21 | I | 0 | I | Q(t) | Q(t+1) |
| 22 | Q(t) | I | Q(t) | 0 | Q(t+1) |
| 23 | Q(t) | 0 | I | Q(t) | Q(t+1) |
| 24 | 0 | ‾Q(t) | Q(t) | I | Q(t+1) |
| 25 | 0 | Q(t) | I | Q(t) | Q(t+1) |
| 26 | Q(t) | I | 0 | I | Q(t+1) |
| 27 | I | Q(t) | 0 | 0 | Q(t+1) |
| 28 | 0 | I | Q(t) | I | Q(t+1) |
| 29 | 0 | I | 0 | Q(t) | Q(t+1) |
| 30 | I | 0 | Q(t) | Q(t) | Q(t+1) |
| 31 | Q(t) | I | 0 | ‾Q(t) | Q(t+1) |
| 32 | ‾Q(t) | 0 | 1 | Q(t) | Q(t+1) |
| 33 | ‾Q(t) | Q(t) | 0 | 1 | Q(t+1) |
| 34 | ‾Q(t) | 1 | Q(t) | 0 | Q(t+1) |
| 35 | 0 | ‾Q(t | ‾Q(t | Q(t) | Q(t+1) |
| 36 | ‾Q(t | 0 | 0 | ‾Q(t | Q(t+1) |

**Пример синтеза двухступенчатого триггера**

Пусть требуется синтезировать на элементах И-НЕ синхрон­ный двухступенчатый триггер, закон функционирования которого задан таблицей входов, представленной в табл. 15 (вариант 31 задания на лабораторную работу), где Q (t) - состояние тригге­ра в такте *t,* a Q (t+1) - состояние триггера в такте (t+1), т.е. после подачи синхроимпульса (при отсутствии синхроимпуль­са триггер своё состояние не меняет).

**Таблица 15**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| (t)  (t)  Q(t+1) | 0  0  Q(t) | 1  0  1 | 0  1  0 | 1  1 |

Составим с учетом значения синхроимпульса (С) и текущего состояния (Q (t)) триггера его полную таблицу переходов (табл. 16).Внесем в полную таблицу переходов значения функции возбуждения RS- триггера первой ступени (R и S), обеспечивающие переход синтезируемого триггера из старого в новое состояние.

**Таблица 16**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| 0  0  0  0  0  0  0  0  I  I  I  I  I  I  I  I | 0  0  0  0  I  I  I  I  0  0  0  0  I  I  I  I | 0  0  I  I  0  0  I  I  0  0  I  I  0  0  I  I | 0  I  0  I  0  I  0  I  0  I  0  I  0  I  0  I | 0  I  0  I  0  I  I  I  0  I  0  0  0  I  I  0 | -  0  -  0  -  0  0  0  -  0  -  I  -  0  0  I | 0  -  0  -  0  -  I  -  0  -  0  0  0  -  I  0 |

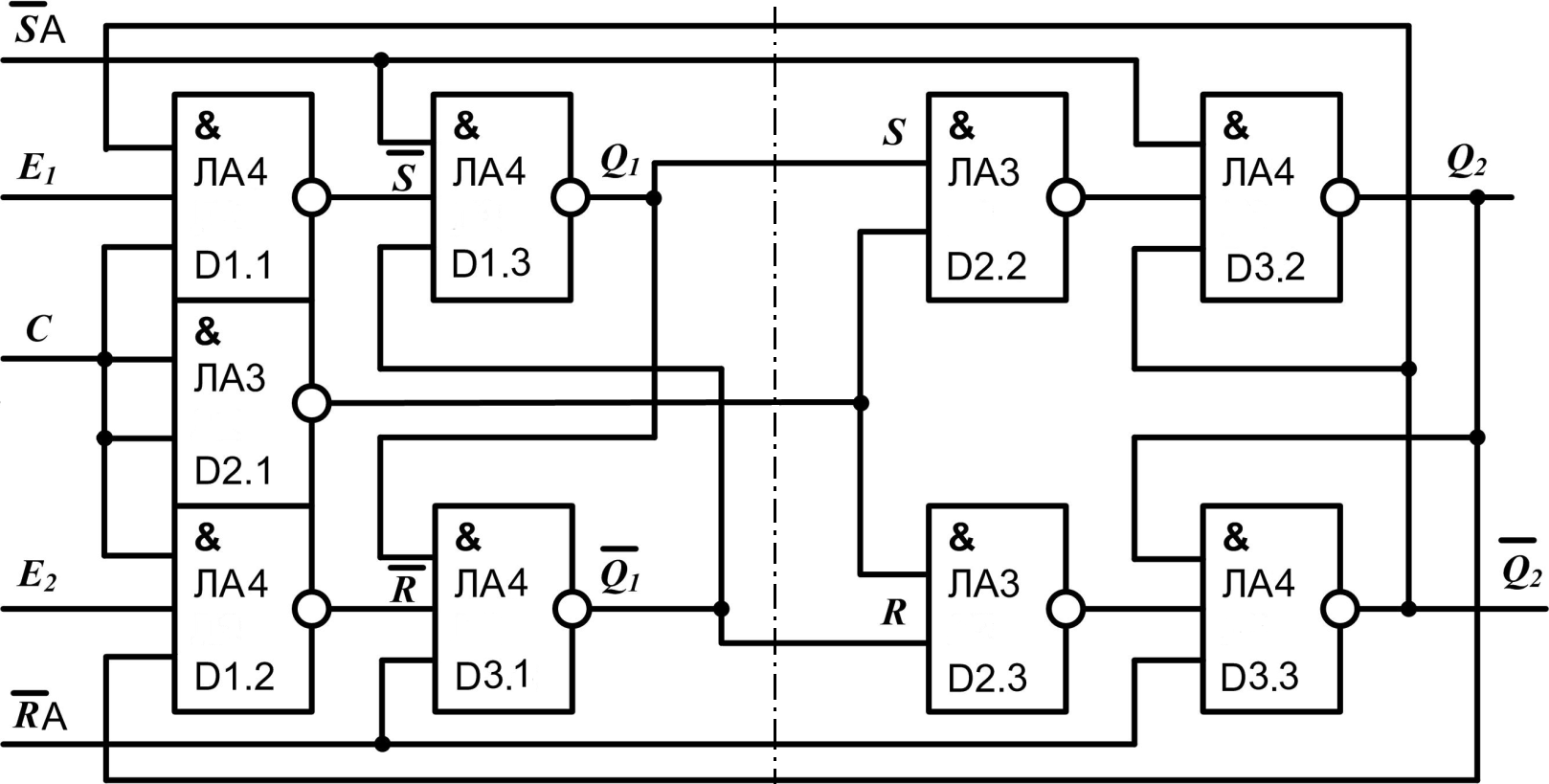
Произведем запись значений функций возбуждения R S - триггера в карты Карно

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CQ** |  |  |  | |  |  | **CQ** |  |  | |  |  |
| 00 | - | 0 | | 0 | - |  | 00 | 0 | - | - | | 0 | |
| 01 | - | 0 | | 0 | 0 |  | 01 | 0 | - | - | | 1 | |
| 11 | - | 0 | | 1 | 0 |  | 11 | 0 | - | 0 | | 1 | |
| 10 | - | 0 | | 1 | - |  | 10 | 0 | - | 0 | | 0 | |
| **R** | 00 | 01 | | 11 | 10 |  | **S** | 00 | 01 | 11 | | 10 | |

Произведя минимизацию функций возбуждения R и S , получим для них следующие минимальные выражения:  .

Поскольку в качестве первой ступени синтезируемого триг­гера используется асинхронный RS -триггер с инверсным управле­нием, то комбинационная схема должна формировать сигналы ‾S и ‾R, согласно выражениям: ; .

В соответствии с полученными выражениями для функций возбуждения, обобщенной схемой двухступенчатого синхронного триггера (рис. 17) и функциональной схемой синхронного одно­ступенчатого триггера (рис. 19), а также с учетом необходимости возможности асинхронной установки триггера в «0» и «1», построена схема триггера за­данного типа (рис. 20). Штрих-пунктирная линия разделяет ее на I ступень и II ступень



I СТУПЕНЬ II СТУПЕНЬ

Рис. 20

**Содержанке отчета**

1. Материалы по синтезу схемы синхронного двухступенчатого триггера (согласно примеру, приведенному в методических указаниях, включая схему аналогичную рис.20).

2. Модель разработанной схемы.

3. Временные диаграммы работы разработанной схемы, полученные путем моделирования её работы в режимах «functional» и «timing», обеспечивающие полноценный контроль работы триггера для всех строк табл. 16.

4. Функциональное обозначение полученной схемы триггера.

**Порядок выполнения работы**

1. Представить преподавателю письменный отчет о подготовке к лабораторной работе, ответить на предложенные им контрольные вопросы и получить допуск к выполнению лабораторной работы.

2. Перенести на компьютер предоставленного стенда разработанный проект модели двухступенчатого триггера и убедиться в его работоспособности путем контрольного моделирования работы триггера в режиме «timing».

3. Включить в состав модели триггера схему подавления дребезга контактов, установив ее на вход синхронизации работы двухступенчатого триггера («С»).

4. Произвести закрепление входных и выходных сигналов двухступенчатого триггера за элементами стенда (см. Приложение 3). При этом сигнал синхронизации счетчика «С» должен поступать с кнопки стенда на вход схемы подавления дребезга; сигналы возбуждения триггера «Е1» и «Е2», а также сигналы асинхронного сброса триггера «RА» и асинхронной установки триггера «SА» - с тумблеров стенда, а выходы триггера (Q1 и Q2) должны поступать на светодиодные индикаторы стенда. Рекомендуется также вывести на светодиодные индикаторы стенда сигналы «S» и «R» и сигнал с выхода схемы подавления дребезга.

5. Осуществить прошивку проекта на плату стенда.

6. Произвести проверку макета схемы триггера в статическом режиме. Для этого:

а) подав на вход «RA» триггера импульс с тумблера стенда, проконтролировать при помощи светодиодных индикаторов, что обе ступени триггера обнулились;

б) подав на вход «SA» триггера импульс с тумблера стенда, проконтролировать при помощи светодиодных индикаторов, что обе ступени триггера установились в «1»;

в) подавая на вход «С» триггера (через схему подавления дребезга) импульсы с кнопки стенда, убедиться при помощи светодиодных индикаторов, что триггер изменяет свое состояние в соответствии с заданным значением сигналов возбуждения триггера «Е1» и «Е2»;

7. Продемонстрировать работу макета схемы триггера преподавателю.

9. Получив разрешение преподавателя, осуществить выключение стенда.

**Контрольные вопросы**

1. Какие типы триггера имеются в составе серии микросхем К155?

2. В чем отличие асинхронных и синхронных триггеров?

3. Для чего используется двухступенчатая структура триггера?

4. В чем сущность синтеза двухступенчатых триггеров?

5. Какие функциональные типы триггероввам известны?

6. Какие триггера называют комбинированными?

7. Составьте таблицу входов для JK, D, DV,T, или RS триггера.

8. Выведите характеристическое уравнение для JK, D, DV,T, или RS триггера.

**ЛАБОРАТОРНАЯ РАБОТА № 4**

**СИНТЕЗ И ИССЛЕДОВАНИЕ СДВИГОВЫХ И МНОГОФУНКЦИОНАЛЬНЫХ РЕГИСТРОВ**

**Цель лабораторной работы**

Изучение одного из основных узлов ЭВМ - регистра; овладение методами синтеза сдвиговых и многофункциональных регистров, приобретение навыков в моделировании, макетировании, наладке и экспериментальном исследовании регистров.

**Общие сведения о регистрах**

Регистры являются самыми распространенными узлами цифровых устройств. Имеется большое разнообразие типов регистров, отличающихся друг от друга по назначению, организации приема и выдачи информации и выполняемым операциям.

Все операции, выполняемые регистрами, разделяется на четыре группы.

Первую группу составляют операции, связанные с параллельным приемом входного слова, установкой регистра в нулевое или единичное состояние, приемом входного слова в обратном или дополнительном коде и т.п.

Вторая группа операций связана с параллельной выдачей слов из регистра. Выдача слов может производиться в прямом, обратном иди дополнительном коде.

Третью группу составляют сдвиговые операции.

К четвертой же группе относятся поразрядные логические операции над многоразрядными словами. К таким операциям относятся поразрядное логическое сложение, поразрядное логическое умножение и т.п.

В зависимости от того, какие из перечисленных групп операций выполняются регистрами, они делятся по своему назначению на регистры хранения, сдвигающие регистры и многофункциональные регистры.

Регистры классифицируются также по способу приема и выдачи информация. По этому признаку все регистры делятся на параллельные регистры, последовательные регистры и параллельно-последовательные регистры.

В параллельных регистрах прием и выдача слов по всем разрядам производится одновременно. В последовательных же регистрах слова принимаются и выдаются последовательно разряд за разрядом за счет выполнения сдвиговых операций. Параллельно-последовательные регистры имеют одновременно входы для параллельного приема и выдачи слов и входы для последовательного приема и выдачи слов. Такие регистры могут выполнять взаимное преобразование последовательного кода в параллельный код и наоборот.

По количеству каналов передачи данных регистры классифицируются на парафазные и однофазные. В парафазных регистрах обрабатываемые слова выдаются в парафазном коде. При этом каждый разряд передаваемого слова представлен двумя сигналами, один из которых имеет значение передаваемого разряда слова, а второй противоположное значение.

Регистры хранения наиболее просты по своей организации. Они могут выполнять только те операции, которые относятся к первым двум группам.

Сдвиговые регистры обеспечивают выполнение операций, принадлежащих к первым трем группам. Многофункциональные же регистры обеспечивают выполнение операций, относящихся ко всем четырем группам операций.

Несмотря на отличие в перечне выполняемых операций, структура сдвиговых и многофункциональных регистров однотипна.

По этой причине методы синтеза и способы построения сдвиговых и многофункциональных регистров одинаковы.

Обобщенная схема логической структуры многофункциональных и сдвиговых регистров приведена на рис. 26.

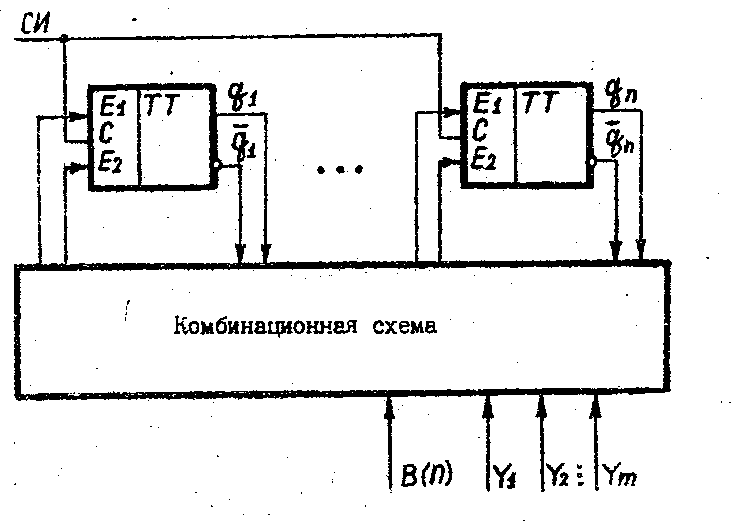


Рис. 26

Наиболее часто сдвиговыми и многофункциональными регистрами выполняются следующие операции:

а) сдвиг вправо на r разрядов;

б) сдвиг влево на L разрядов;

в) обнуление разрядов регистра;

г) инвертирование слова Q(n), хранящегося в регистре;

д) прием входного слова B(n);

е) поразрядное логическое сложение B(n)∨Q(n);

ж) поразрядное логическое умножение B(n)∧Q(n);

з) поразрядное логическое сложение по mod 2 B(n)⊕Q(n);

и) хранение записанной в регистр информации;

к) установка в единичное состояние разрядов регистра.

Поскольку структура регистра регулярна, комбинационную схему регистра можно разделить на n одинаковых разрядных комбинационных схем, каждая из которых принадлежит одному разряду регистра (рис. 27) и формирует сигналы возбуждения входов триггера этого разряда. Структура разрядной комбинационной схемы определяется заданным набором операций, типом используемого триггера и выбранной элементной базой.

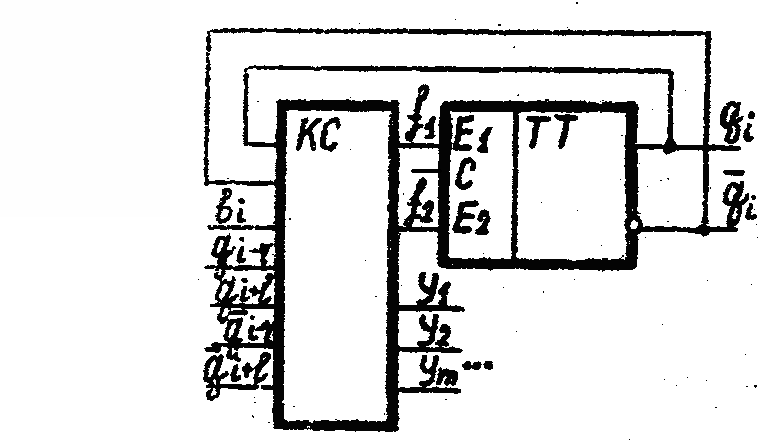


Рис. 27

**Подготовка к выполнению работы**

1. Изучить структуру регистров и их классификацию.

2. Изучить методы синтеза схем регистров.

3. Синтезировать 3-разрядный регистр на базе D –триггеров, выполняющий микрооперации, указанные в табл. 17, в соответствии с заданным номером варианта задания на микросхемах серии К155.

4. Вычертить схему синтезированного регистра с обязательным указанием для каждого элемента схемы порядкового номера в корпусе соответствующего ему элемента микросхемы.

**Таблица 17.**

Варианты заданий на лабораторную работу

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вар. | Типы выполняемых операций | | | |
| 1 | а(*r*=1) | д | е | з |
| 2 | б(*l*=1) | е | д | в |
| 3 | д | а(*r*=1) | е | г |
| 4 | д | е | б(*l*=1) | ж |
| 5 | е | д | а(*r*=1) | к |
| 6 | а(*r*=1) | д | з | в |
| 7 | б(*l*=1) | з | д | г |
| 8 | д | а(*r*=1) | з | ж |
| 9 | з | д | б(*l*=1) | к |
| 10 | а(*r*=1) | д | в | г |
| 11 | б(*l*=1) | в | д | ж |
| 12 | а(*r*=1) | д | г | ж |
| 13 | д | б(*l*=1) | к | г |
| 14 | а(*r*=1) | е | з | в |
| 15 | з | б(*l*=1) | ж | е |
| 16 | к | з | е | а(*r*=1) |
| 17 | е | в | г | б(*l*=1) |
| 18 | ж | е | а(*r*=1) | в |
| 19 | г | е | ж | б(*l*=1) |
| 20 | е | к | г | а(*r*=1) |
| 21 | ж | е | к | б(*l*=1) |
| 22 | з | а(*r*=1) | в | г |
| 23 | в | з | б(*l*=1) | ж |
| 24 | а(*r*=1) | к | в | з |
| 25 | г | б(*l*=1) | г | ж |
| 26 | з | к | з | а(*r*=1) |
| 27 | г | ж | к | б(*l*=1) |
| 28 | а(*r*=1) | в | г | ж |
| 29 | в | ж | б(*l*=1) | к |
| 30 | г | а(*r*=1) | ж | к |
| 31 | а(*r*=1) | г | в | е |
| 32  33  34  35  36 | а(*r*=1)  а(*r*=1)  а(*r*=1)  а(*r*=1)  а(*r*=1) | б(*l*=1)  б(*l*=1)  б(*l*=1)  б(*l*=1)  б(*l*=1) | в  и  в  в  к | г  к  к  и  г |

5. Построить модель синтезированной схемы при помощиСАПР Altera Quartus II (см. Приложение 2).

6. Получить временные диаграммы работы модели синтезированной схемы путем моделирования её работы в режимах «functional» и «timing». Порядок изменения входных сигналов регистра должен быть подобран таким образом, чтобы на временных диаграммах были видны все возможные переходы его состояния. Необходимо также учесть, что изменение входных сигналов регистра во время единичного значения синхросигнала недопустимо. Длительность синхросигнала должна быть подобрана таким образом, чтобы задержки схемы, наблюдаемые в режиме «timing» составляли не более 30% длительности синхросигнала.

7. Сопоставить результаты моделирования с заданием на лабораторную работу.

8. Составить отчет о подготовке к лабораторной работе (в формате WORD) и выслать преподавателю для проверки (вместе с проектом).

9. Распечатать проверенный и одобренный преподавателем отчет.

10. Ответить на контрольные вопросы.

**Пример синтеза регистра**

Пусть требуется синтезировать 3-разрядный регистр, выполнявший перечень микроопераций, соответствующий варианту 31 задания на лабораторную работу:

1) сдвиг вправо на 1 разряд;

2) инвертирование слова, хранящегося в регистре;

3) поразрядное сложение по mod 2 слова, хранящегося в регистре с входным словом;

4) поразрядное логическое сложение слова, хранящегося в регистре с входным словом.

Сначала определим количество управляющих сигналов, необходимых для кодирования заданного количества микроопераций:



Зададим коды микроопераций, выполняемых регистром (табл. 18).

**Таблица 18.**

|  |  |  |  |
| --- | --- | --- | --- |
| №  п/п | Микрооперация | Код микрооперации | |
| У2 | У1 |
| 1  2  3  4 | Сдвиг вправо на 1 разряд  Инвертирование  Сложение по mod 2  Логическое сложение | 0  0  1  1 | 0  1  0  1 |

Исходя из заданного перечня микроопераций, можно сделать вывод, что состояние каждого разряда регистра в следующем такте зависит от значения следующих независимых переменных:

У1 - управляющий сигнал;

У2 - управляющий сигнал;

 - состояние разряда, из которого принимается информация при выполнении микрооперации сдвига вправо;

 - текущее состояние *i*-го разряда;

*bi* - состояние *i*-го разряда входного слова.

Составим таблицу переходов *i*-го разряда синтезируемого регистра (табл. 19).

**Таблица 19.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Микрооперация | У2 | У1 | *bi* |  |  |  |
| Сдвиг вправо: | 0  0  0  0  0  0  0  0 | 0  0  0  0  0  0  0  0 | 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | 0  1  0  1  0  1  0  1 |
| Инвертирование: | 0  0  0  0  0  0  0  0 | 1  1  1  1  1  1  1  1 | 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | 1  1  0  0  1  1  0  0 |
| Сложение по mod 2: | 1  1  1  1  1  1  1  1 | 0  0  0  0  0  0  0  0 | 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | 0  0  1  1  1  1  0  0 |
| Логическое сложение: | 1  1  1  1  1  1  1  1 | 1  1  1  1  1  1  1  1 | 0  0  0  0  1  1  1  1 | 0  0  1  1  0  0  1  1 | 0  1  0  1  0  1  0  1 | 0  0  1  1  1  1  1  1 |

В качестве запоминающего элемента в разрабатываемом регистре должен использоваться D-триггер, характеристическое уравнение которого имеет вид . Поэтому значение функции возбуждения триггера *i*-го разряда *D*i(*t*) совпадает со значением *i*-го разряда регистра в следующем такте .

Анализ вариантов построения комбинационной схемы, реализующей функцию *D*i(*t*), показывает, что наиболее эффективной по затратам оборудования будет построение данной комбинационной схемы с использованием мультиплексоров с тремя адресными входами (К155КП7).

Используя методику, приведенную в лабораторной работе № 2, произведем синтез комбинационной схемы на базе выбранного типа мультиплексора.

Анализ возможных вариантов реализации комбинационной схемы доказывает, что в нашем случае минимальные аппаратные затраты при построении схемы достигаются при подаче на адресные входы мультиплексоров переменных У1, У2 и *bi*.

Составим таблицу истинности для реализуемой функции в форме, принятой при использовании мультиплексора для реализации комбинационной схемы (табл. 20).

**Таблица 20.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Адресные входы и поданные на них переменные | | | Переменные, не поданные на адресные входы | | Значение функции *D*i(*t*) | Информационные входы мультиплексора | |
| *А*2(У2) | *А*1(У1) | *А*0(*b*i) | *q*i(*t*) | *q*i+1(*t*) | Обознач. | Сигнал |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 0  0  0  0 | 0  0  0  0 | 0  0  0  0 | 0  0  1  1 | 0  1  0  1 | 0  1  0  1 | B0 |  |
| 0  0  0  0 | 0  0  0  0 | 1  1  1  1 | 0  0  1  1 | 0  1  0  1 | 0  1  0  1 | B1 |  |
| 0  0  0  0 | 1  1  1  1 | 0  0  0  0 | 0  0  1  1 | 0  1  0  1 | 1  1  0  0 | B2 |  |
| 0  0  0  0 | 1  1  1  1 | 1  1  1  1 | 0  0  1  1 | 0  1  0  1 | 1  1  0  0 | B3 |  |
| 1  1  1  1 | 0  0  0  0 | 0  0  0  0 | 0  0  1  1 | 0  1  0  1 | 0  0  1  1 | B4 |  |
| 1  1  1  1 | 0  0  0  0 | 1  1  1  1 | 0  0  1  1 | 0  1  0  1 | 1  1  0  0 | B5 |  |
| 1  1  1  1 | 1  1  1  1 | 0  0  0  0 | 0  0  1  1 | 0  1  0  1 | 0  0  1  1 | B6 |  |
| 1  1  1  1 | 1  1  1  1 | 1  1  1  1 | 0  0  1  1 | 0  1  0  1 | 1  1  1  1 | B7 | 1 |

Используя данные, приведенные в табл. 20, построим схему 3-разрядного регистра, выполняющего заданные микрооперации (рис. 28).

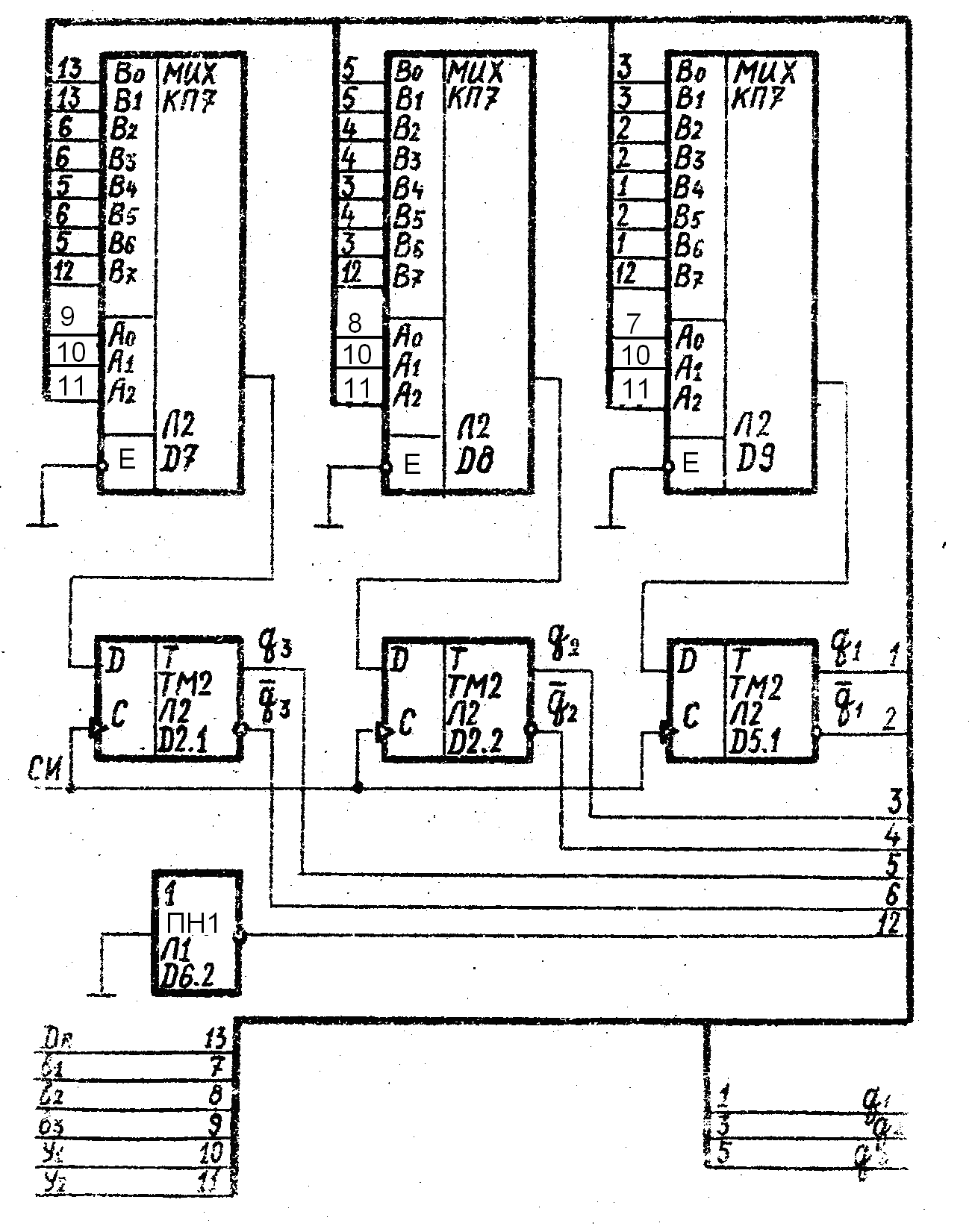


Рис. 28

**Порядок выполнения работы**

1. Представить преподавателю письменный отчет о подготовке к лабораторной работе, ответить на предложенные им контрольные вопросы и получить допуск к выполнению лабораторной работы.

2. Перенести на предоставленный компьютер разработанный проект модели многофункционального регистра и убедиться в его работоспособности путем контрольного моделирования работы триггера в режиме «timing».

3. Включить в состав модели многофункционального регистра схему подавления дребезга контактов, установив ее на вход синхронизации работы многофункционального регистра.

4. Произвести закрепление входных и выходных сигналов многофункционального регистра за элементами стенда (см. Приложение 3). При этом сигнал синхронизации многофункционального регистра должен поступать с кнопки стенда на вход схемы подавления дребезга; управляющие сигналы У1, У2, а также разряды входного слова «в1», «в2» и «в3» и входной сигнал «Dr» (или «Dl») - с тумблеров стенда, а выходы регистра (q1, q2 и q3) должны поступать на светодиодные индикаторы стенда. Рекомендуется также вывести на светодиодные индикаторы стенда сигналы с выходов мультиплексоров и сигнал с выхода схемы подавления дребезга.

5. Осуществить прошивку проекта на плату стенда.

6. Произвести проверку макета схемы многофункционального регистра в статическом режиме. Для этого устанавливая для каждого значения сигналов У1, У2 различные значения сигналов «в1», «в2», «в3» и «Dr» (или «Dl») и подавая на вход «C»регистра импульс с кнопки стенда убедиться, что многофункциональный регистр правильно выполняет все заданные операции при различных значениях операндов.

7. Продемонстрировать работу макета схемы многофункционального регистра преподавателю.

9. Получив разрешение преподавателя, осуществить выключение стенда.

**Содержание отчета**

1. Материалы по синтезу трехразрядного регистра (согласно примеру, приведенному в методических указаниях).

2. Модели разработанных схем.

3. Временные диаграммы работы синтезированных комбинационных схем, полученные путем моделирования их работы в режимах «functional» и «timing».

**Контрольные вопросы**

1. Какие операции можно выполнить при помощи многофункционального регистра?

2. В чем сущность синтеза многофункциональных и сдвиговых регистров?

3. Какую последовательность операций нужно выполнить на синтезированном вами регистре, чтобы записать в него двоичный код "101", если в исходном состоянии в нем записан код "001"?

4. Как регистры классифицируются по типу выполняемых операций?

5. Как регистры классифицируются по способу приема и выдачи информации?

6. Дайте полное классификационное определение регистра, синтезированного при выполнении лабораторной работы.

7. Сколько дополнительных микросхем потребуется для реализации схемы регистра, которая разработана в качестве примера, если на адресные входы мультиплексора вместо переменных У1 и У2 подать переменные  и ?

8. Чем будет отличаться синтез многофункционального регистра в случае применения в нем в качестве запоминающего элемента вместо D-триггера использовать Т-триггер?

9. Какие имеются регистры в составе серии микросхем К155 и какие операции они могут выполнять?

**Литература.**

1.Рахимов Т.М. Справочник по микросхемам серии к 155. – Новосибирск: Новосибирская картографическая фабрика, 1991.

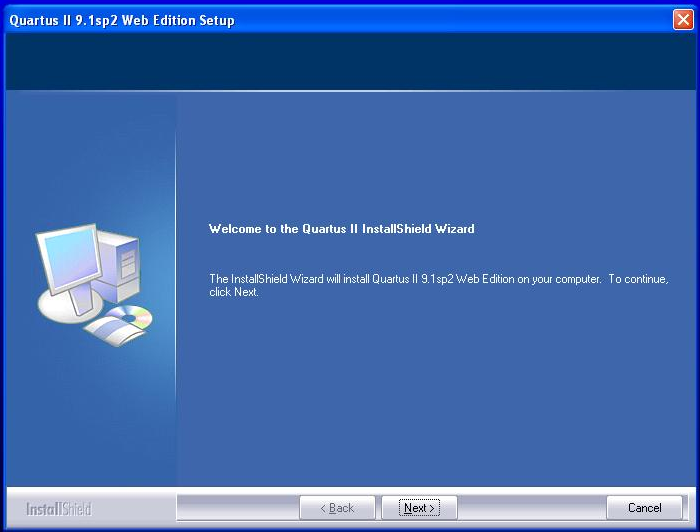
2.Справочник. Интегральные микросхемы /под ред. Тарабрина Б.В. –М.: «Радио и связь», 1983.

3.Угрюмов Е.П. Цифровая схемотехника. – СПб.: «БХВ-Перербург», 2007.

**Приложение 1.**

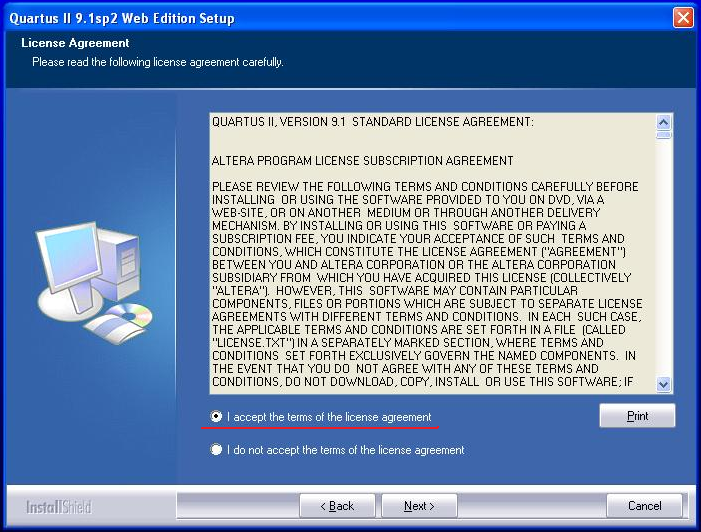
**Установка САПР Altera Quartus II 9.1sp2 Web Edition.**

Для начала установки необходимо запустить программу *91sp2\_quartus\_free.exe*, после чего появиться окно, представленное на рисунке ниже (Рис.1)



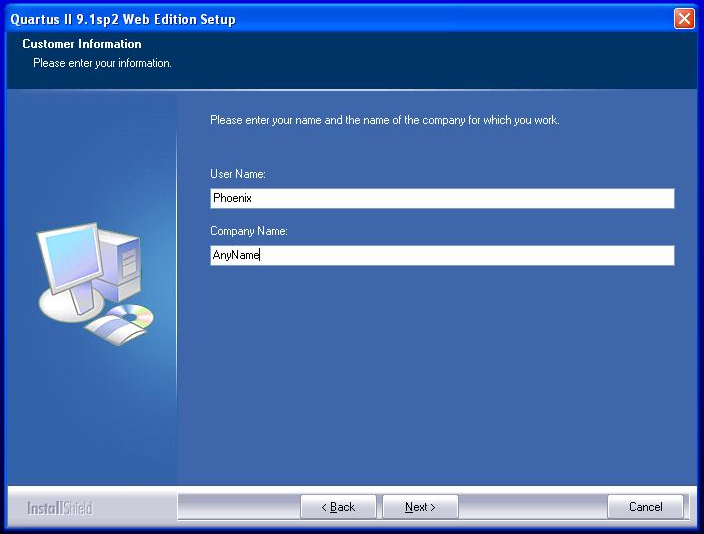
***Рис.1***

Нажать кнопку «*Next»*, после чего появляется окно с лицензионным соглашением (Рис.2), согласиться с условиями и нажать кнопку «*Next».*



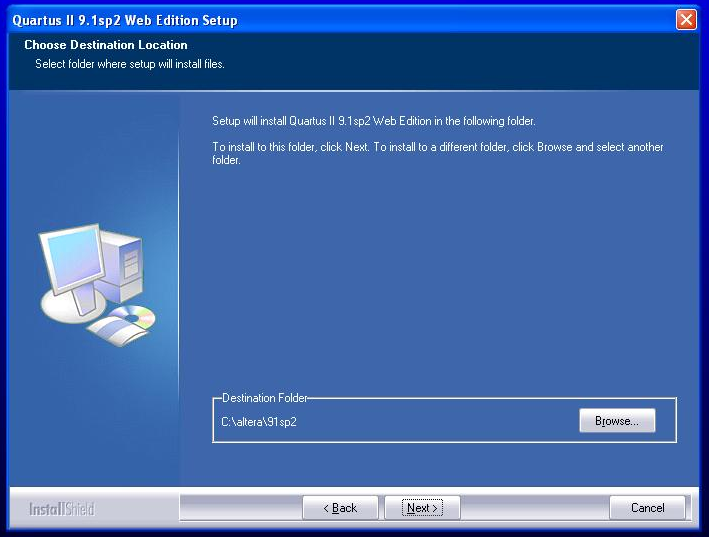
***Рис.2***

В следующем окне указать имя пользователя в графе «*User Name»* и имя компании в графе «*Company Name»* (Рис.3). Имена могут быть любыми. Данные поля обязательны.



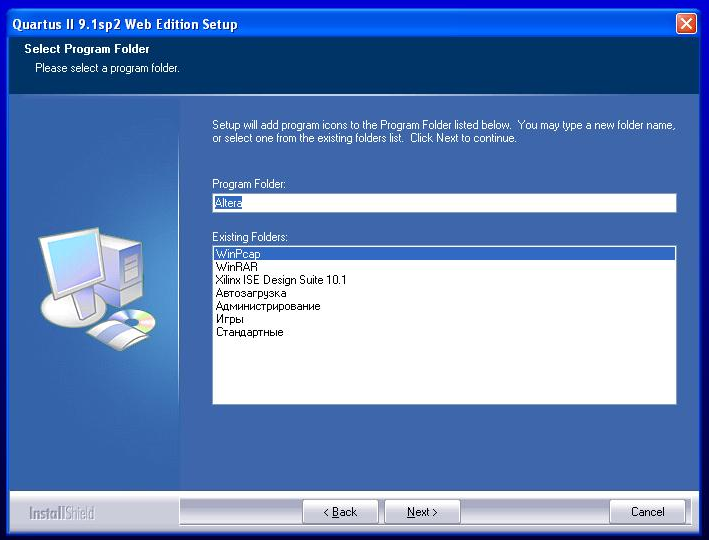
***Рис.3***

После заполнения полей нажать кнопку «*Next»*. В следующем появившемся окне нажать кнопку «*Next»* (Рис.4).



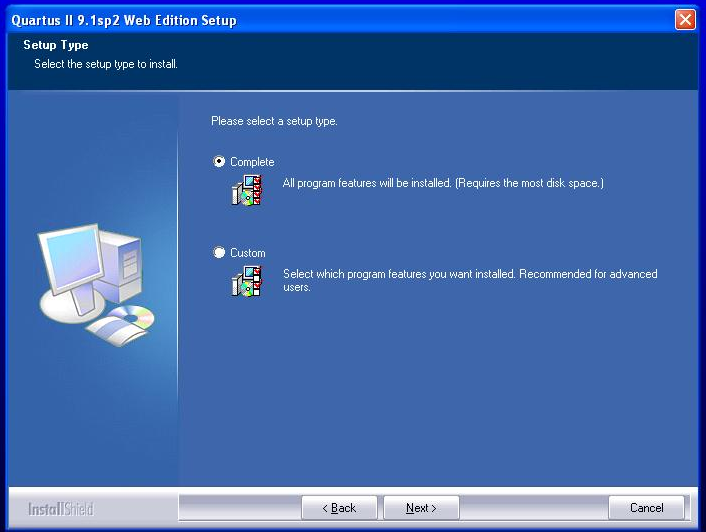
***Рис.4***

В появившемся окне нажать кнопку «*Next»* (Рис.5).



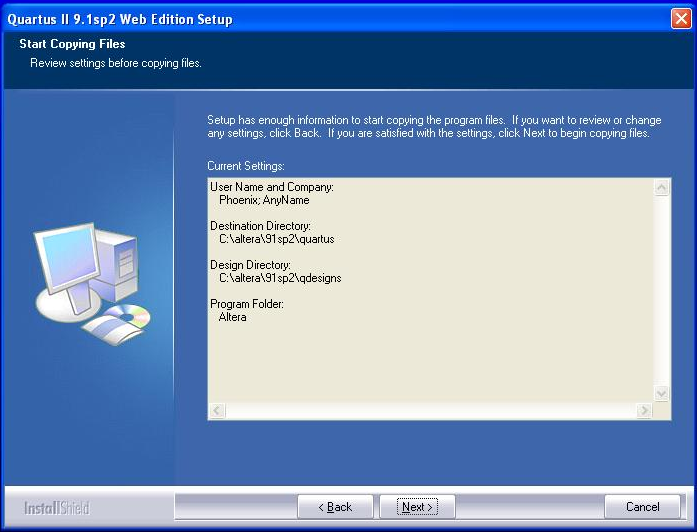
***Рис.5***

В следующем окне выбрать тип установки «*Complete»* и нажать кнопку «*Next»* (Рис.6).



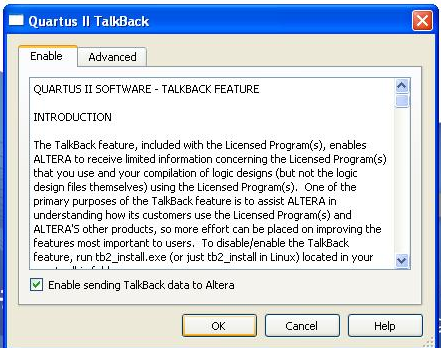
***Рис.6***

В следующем окне нажать кнопку «*Next»* (Рис.7). После чего начинается установка Altera Quartus II.



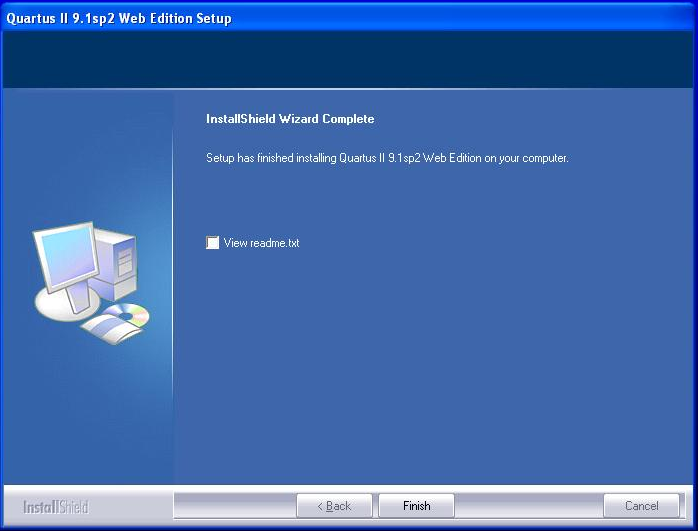
***Рис.7***

По окончании установки появиться окно, приведенное ниже (Рис.8)



***Рис.8***

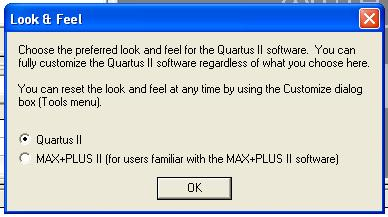
Нажать кнопку «*OK»*. При этом появится окно, приведенное ниже (Рис.9)



***Рис.9***

Нажать кнопку «*Finish»* (Рис.9). Установка завершена. После установки ярлык программы должен появиться на рабочем столе.

При первом запуске Quartus II появится окно (Рис.10)



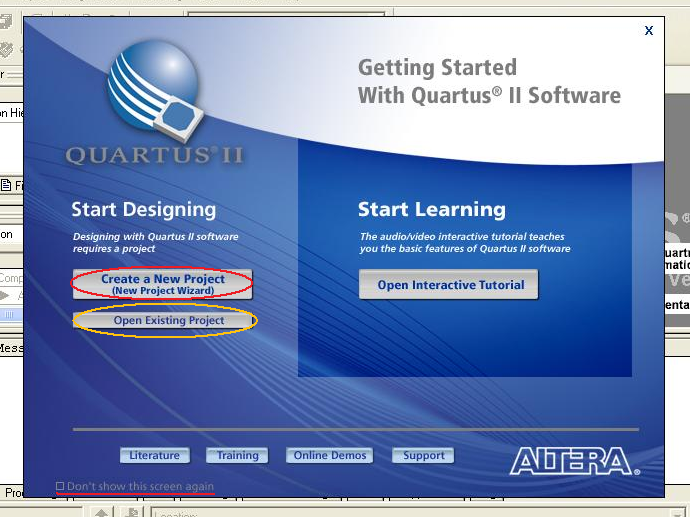
***Рис.10***

Выбрать «Quartus II» и нажать кнопку «OK». При этом появится следующее окно (Рис.11)



***Рис.11***

Выбрать «*Run the Quartus II software»* и нажать кнопку «*OK»*. После этого появится окно (Рис.12)



***Рис.12***

В этом окне нам предлагают создать новый проект (Create a New Project) или открыть существующий (Open Existing Project). Данное окно будет появляться при каждом запуске Quartus II. Чтобы оно не появлялось при каждом запуске – нужно поставить галочку в левом нижнем углу в поле «*Don't show this screen again».*

После установки необходимо добавить элементную базу микросхем 155 серии, добавление производится следующим образом:

1. Копировать папку из «комплекта студента» под названием **К155** в папку с установленной программой Quartus II (стандартно **C:/altera/**)
2. Создать новый (либо открыть любой старый проект.
3. На вкладке Assignments выбрать пункт Settings.
4. На вкладке «*Libraries»* в окошке «*Global libraries»* нажать кнопку с троеточием.
5. Выбрать скопированную папку **К155 ( C:/altera/K155)** и нажать «*открыть»* и затем «*add».*

По окончанию выполнения данных пунктов при создании проекта будет доступна элементная база микросхем серии К155.

**Приложение 2.**

**Создание и моделирование проекта на САПР Altera Quartus II.**

Рассмотрим работу Quartus II на примере логического элемента ”и” с двумя входами.

1. После запуска Quartus II создаем новый проект: меню *"File → New Project Wizard..."*(рис.1).

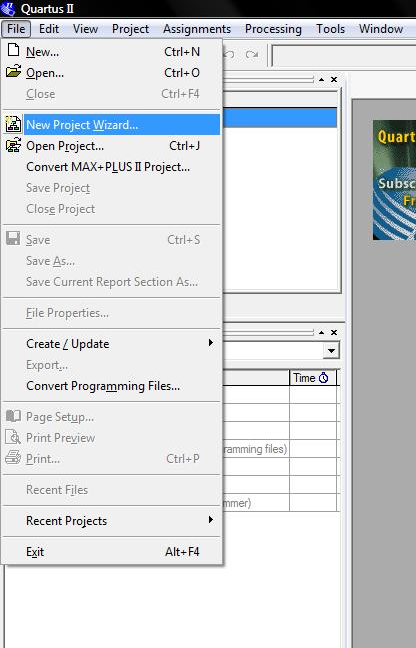


рис.1

1. В появившемся окне нажать на клавишу «*Next»* (рис.2). Так же можно отметить поле «Don’t show me this introduction again», чтобы в дальнейшем данное окно не появлялось.



рис.2

1. В следующем окне (рис.3), в первом поле, указать путь к рабочей директории (директория, где будут храниться наши файлы проекта). По умолчанию путь «*c:\altera\91sp2\quartus»*; мы же создадим папку «*c:\altera\91sp2\quartus \labs»*. Настоятельно рекомендуем для предотвращения потери файлов создавать отдельную папку для каждого проекта. Во втором поле указать имя проекта, в нашем случае «LabAND». Нажать кнопку «*Next»*.

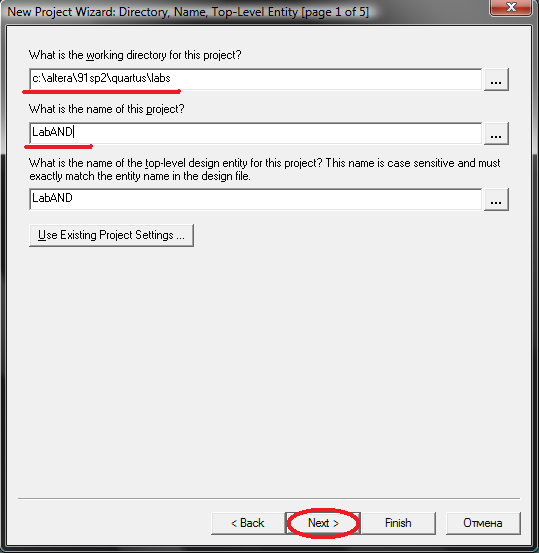


рис.3

1. В следующем окне нажать кнопку «*Next»* (рис.4). В данном окне нам предлагают подключить файлы к нашему проекту, т. к. мы создаем пустой проект, мы не указываем никаких файлов.

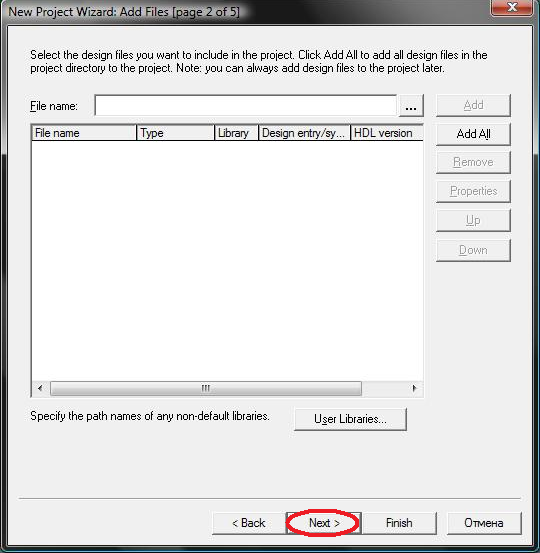


рис.4

1. В следующем окне (рис.5) указать тип устройства в поле «*Family»*, например, «*Cyclone II»* и выбрать имя устройства, например ***EP2C8F256C8***. Нажать кнопку «*Next»*.

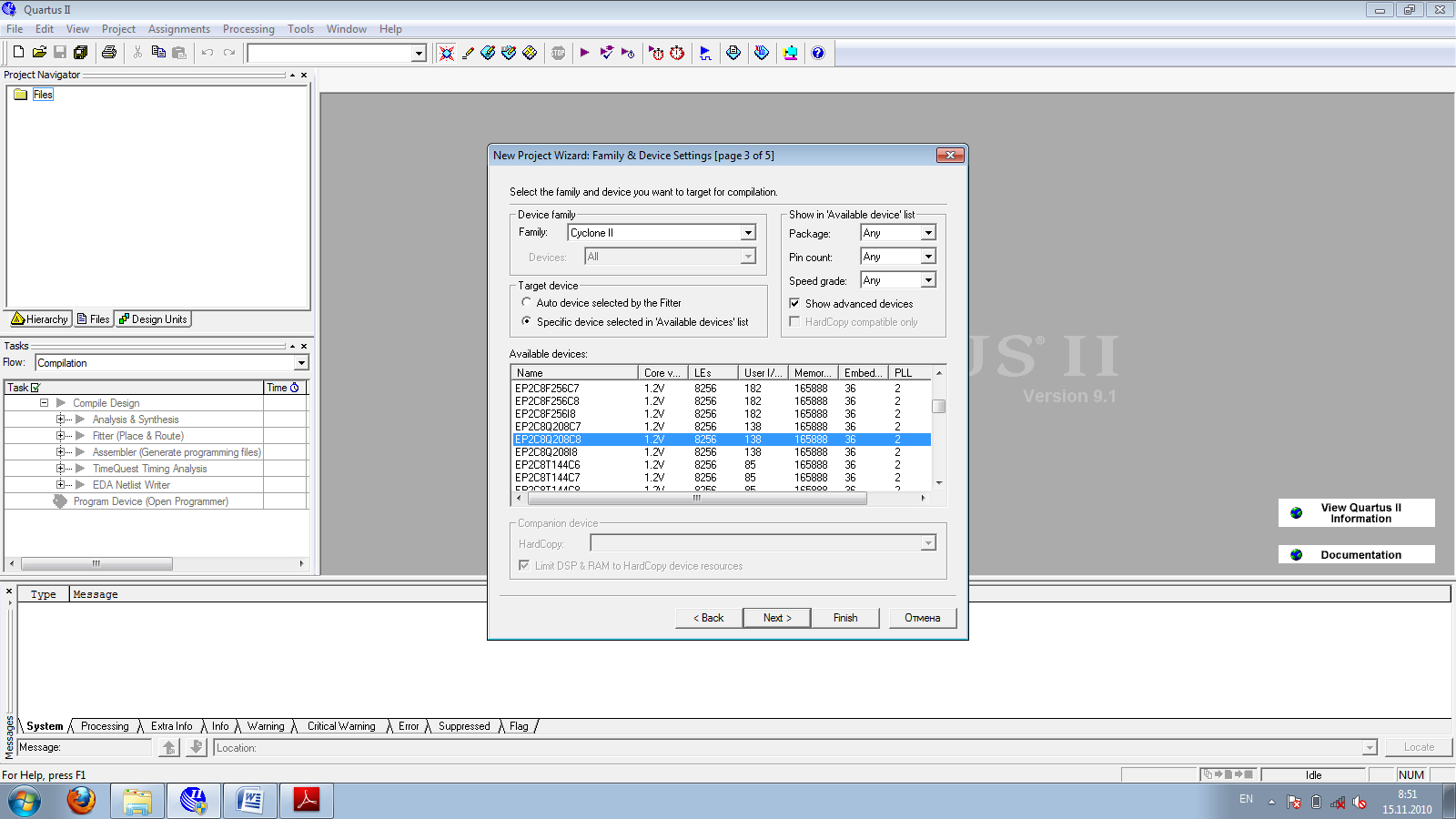


рис.5

1. В следующем окне (рис.6) ничего не надо выбирать, т.к. для создания проекта используется только Quartus II (другие инструменты не используются). Нажать кнопку «*Next»*.

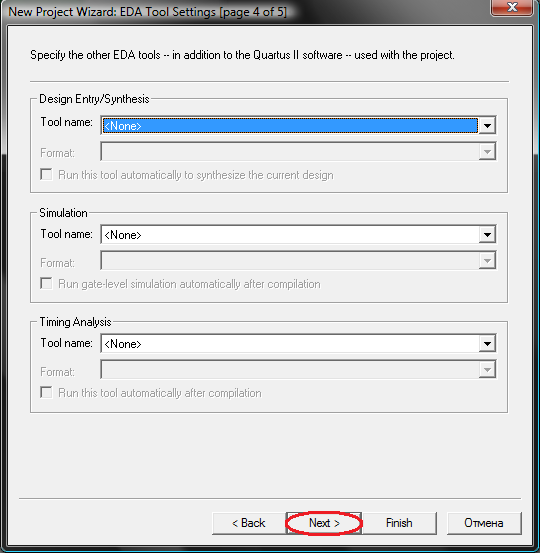


рис.6

1. В последней диалоговой панели мастера проекта можно проверить все параметры (рис.7). Нажать кнопку «*Finish»*.

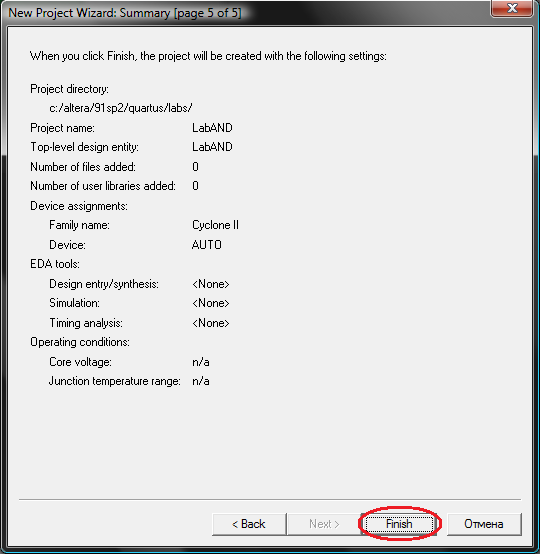


рис.7

Свойства проекта могут быть в любой момент времени просмотрены или изменены в диалоговой панели *"Settings"* – меню *"Assigments → Settings..."*

1. Для ввода схемы проектируемого устройства необходимо открыть редактор структурных схем (block diagram), создав соответствующий файл – меню *"File → New..."*, в открывшейся диалоговой панели *"New"* на закладке *"Design Files"* выбрать пункт *"Block Diagram/Schematic File"* (рис.8) и нажать кнопку «ОК».

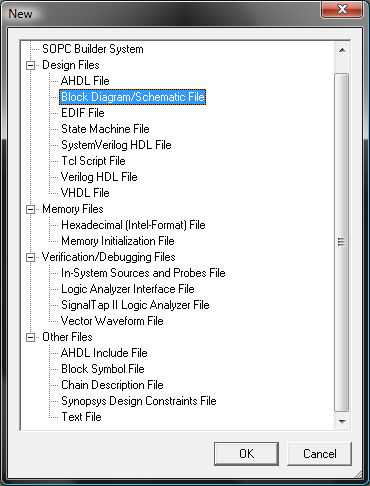


рис.8

Проект создан. Если всё сделано правильно, то должен появиться редактор схем, как на рис.9.

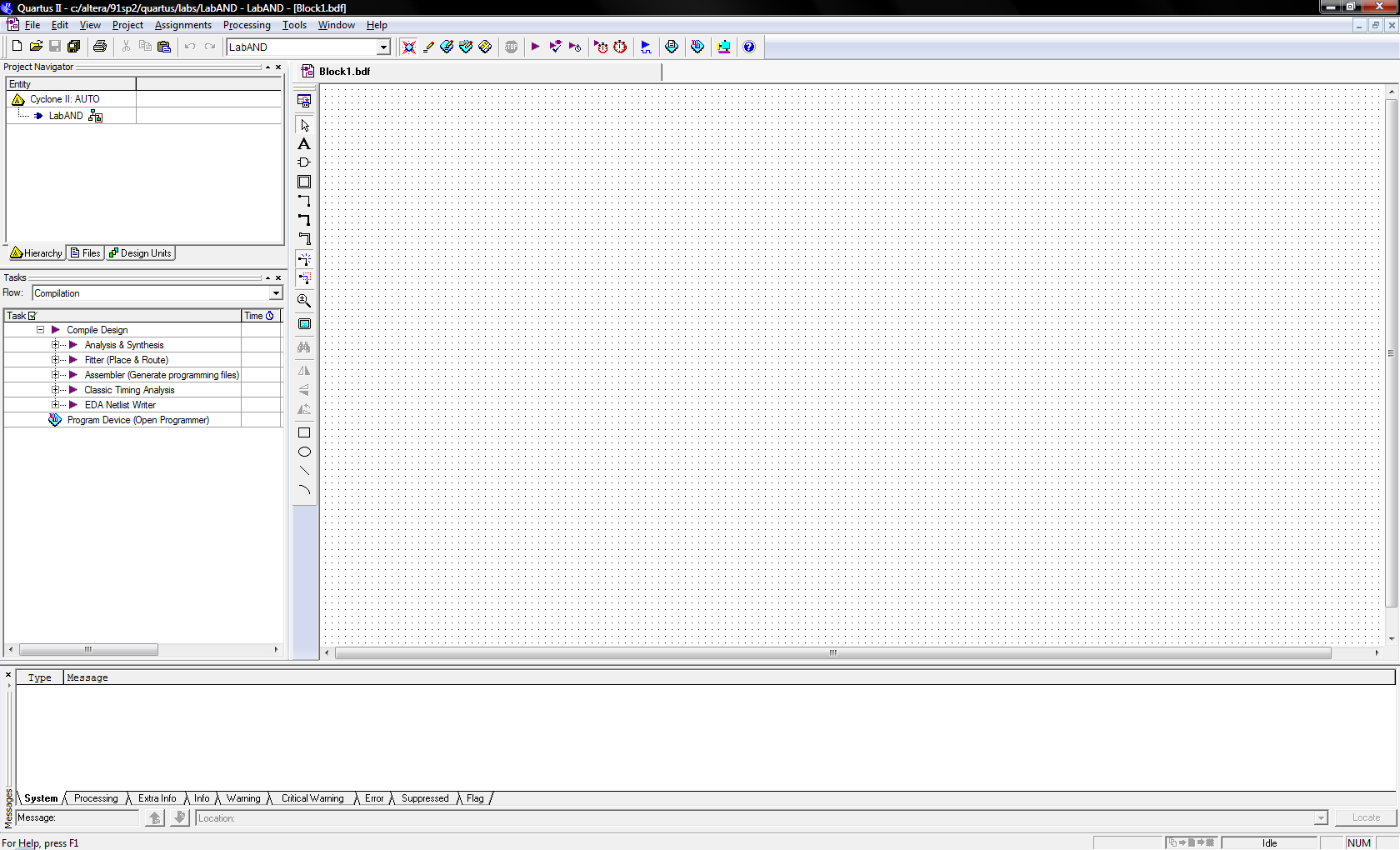


рис.9

Рассмотрим окна программы. В окне на рис.10 отображаются открытые проекты и файлы, подключенные к ним.

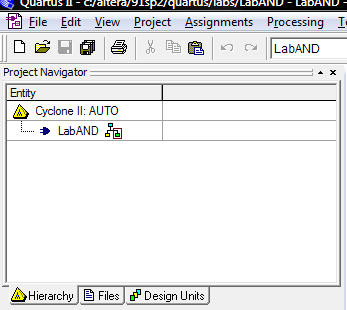


рис.10

В окне на рис.11 отображается процесс компиляции проекта.

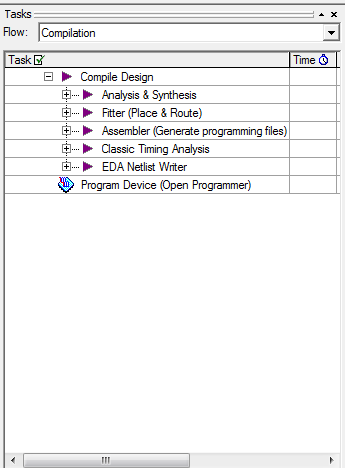


рис.11

В окне на рис.12 отображаются ошибки и предупреждения, возникающие при компиляции.

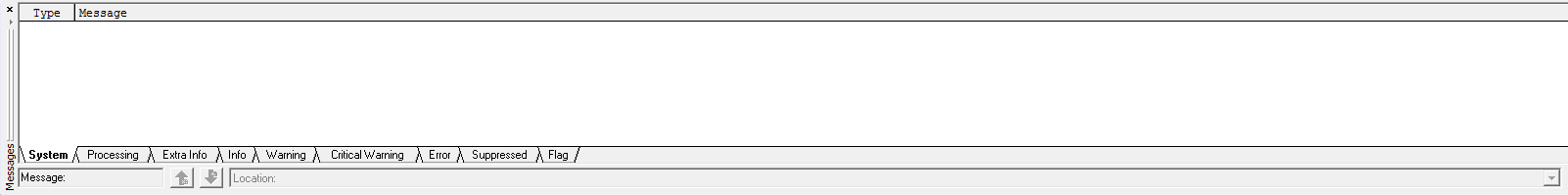


рис.12



**Рассмотрим панель инструментов редактора схем** (рис.13)**.**

Выделим наиболее важные инструменты:

 Text Tool – добавляет надпись.

 Symbol Tool – вызывает мастера добавления элементов. Так же можно вызвать двойным щелчком клавиши мыши в свободном месте редактора схем.

 Orthogonal Node Tool – “соединитель узлов”. Соединяет выходы/входы одних элементов со входами/выходами других элементов.

 Zoom Tool – увеличение/уменьшение масштаба. Левая клавиша мыши – увеличить, правая – уменьшить.

 Rotate Tool – поворот на 900 влево выбранного элемента.

рис.13

Рассмотрим работу с Symbol Tool (рис.14). Чтобы его вызвать – нажмем на соответствующую кнопку в панели инструментов. Появиться мастер добавления элементов.

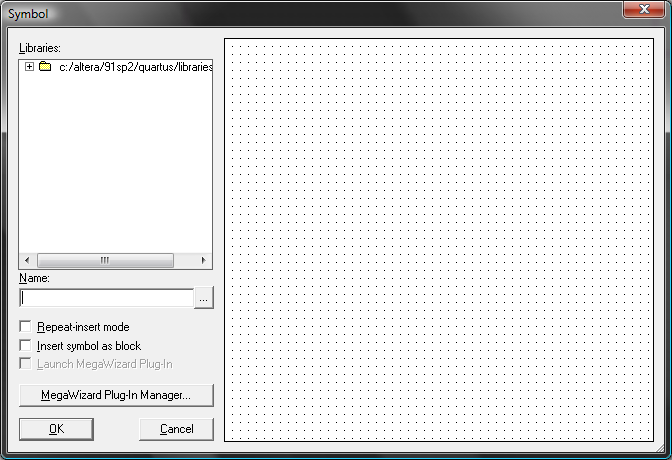


рис.14

Нам предлагается библиотека с логическими элементами. Раскроем её и выберем нужный элемент. Так же можно ввести имя элемента в поле *Name*, для его поиска в библиотеке (нужно точное название). Наглядное изображение элемента будет представлено в окне справа (рис.15).

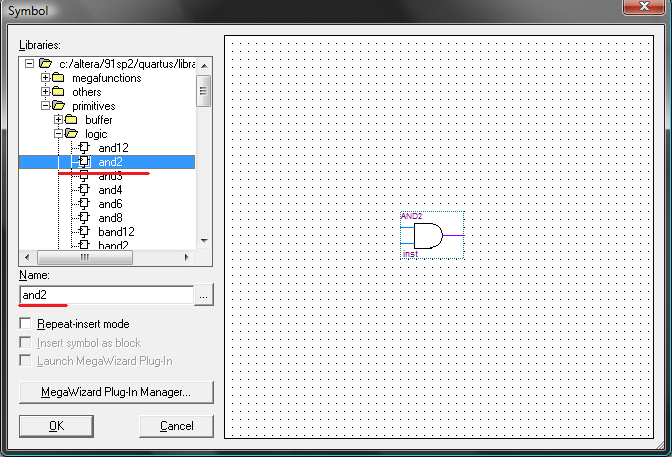


рис.15

После выбора нужного элемента из списка предлагаемых элементов, нажимаем кнопку *Ok*. После чего нам необходимо разместить элемент в любом свободном месте редактора схем.

Важнейшими элементами являются пины ввода/вывода, находящиеся в *primitives\pin* (рис.16). Добавим их в наш проект. Пин *input* подключается ко входам, *output* – к выходам.



Рис.16

После добавления пинов соединим их со входами логического элемента (рис.17), для этого выберем ”соединитель узлов” на панели инструментов. Инструмент можно и не выбирать, достаточно подвести курсор мыши к краю элементу и появиться соответствующий значок.

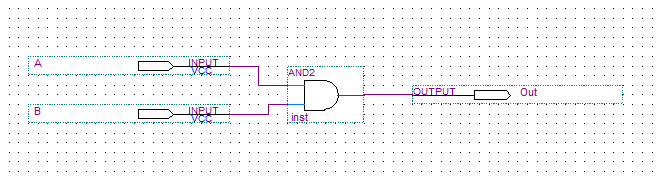


рис.17

После сборки схемы, сохраним её – меню *"File → Save"*.

При сохранении файл автоматически добавляется в текущий проект. В этом можно убедиться, просмотрев ветвь *Files* свойств проекта – меню *"Project → Add/Remove Files in Project..."* или закладку *Files* навигатора проекта – меню *"View → Utility Windows → Project Navigator"*.

Для отображения разработанной схемы на логику устройства, создания конфигурационных файлов для загрузки в ИС и анализа временных характеристик устройства необходимо выполнить компиляцию проекта – меню *"Processing → Start Compilation"*. При успешной компиляции выводится сообщение (возможно с предупреждениями) (рис.18)

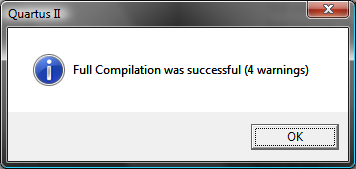


рис.18

Предупреждения, возникающие при компиляции:

***Warning: Feature LogicLock is only available with a valid subscription license. Please purchase a software subscription to gain full access to this feature***. - особенности LogicClock доступны только с полной лицензией.

***Warning: Found 1 output pins without output pin load capacitance assignment -*** не задана емкость нагрузки на выходном пине. По умолчанию емкость нагрузки принимается 0 pF.

***Warning: The Reserve All Unused Pins setting has not been specified, and will default to 'As output driving ground'. -*** пины не специфицированы и по умолчанию используются как заземленные.

***Critical Warning: No exact pin location assignment(s) for 3 pins of 3 total pins***

***Info: Pin not assigned to an exact location on the device -*** не задано точное местоположение пинов на плате.

**Данные предупреждения не влияют на проект.**

В противном случае причину неуспешной компиляции можно просмотреть в окне сообщений – меню *"View → Utility Windows → Messages"*. Нужно исправить в схеме ошибки и повторить компиляцию. Результаты компиляции можно просмотреть в окне отчета компилятора – меню *"Processing → Compilation Report"*.

Для исследования аппаратурной сложности устройства – количества занимаемых логических элементов, блоков памяти, внешних выводов – нужно просмотреть ветвь *"Fitter → Summary"*.

Для тестирования устройства можно построить временные диаграммы (waveform) его работы. Чтобы открыть редактор временных диаграмм, нужно в диалоговой панели *"New"* на закладке *"Verification/Debugging Files"* выбрать *"Vector Waveform Editor"* (рис.19).

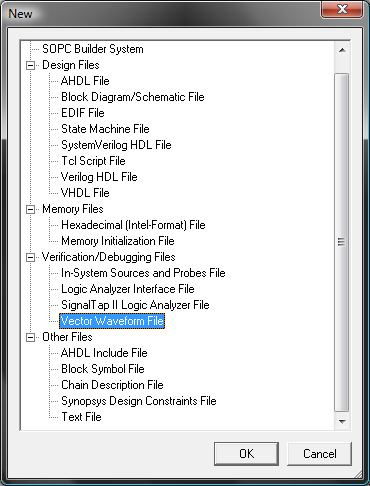


рис.19

Сохраним созданный пустой файл (меню "File → Save"), в результате чего он будет добавлен в проект. Если не отображаются входные/выходные пины, то добавить их, нажав правой клавишей мыши и выбрав *”Insert → Insert Node or Bus…”* (рис.20).

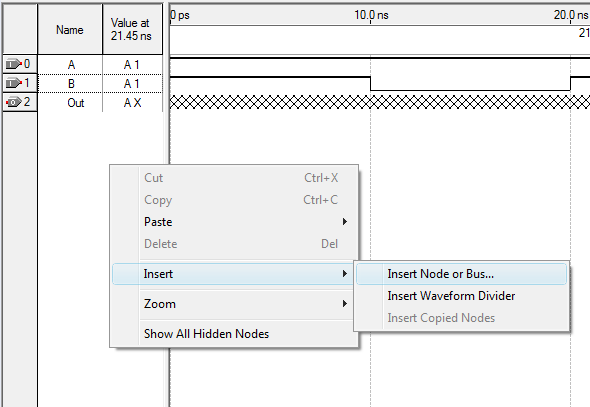


рис.20

В поле *Name* указать имя пина, в поле *Type* – тип (рис.21). После добавления сохраняем проект.

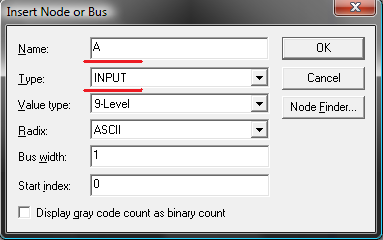


рис.21

При помощи редактора зададим значения на входах устройства. Чтобы задать постоянный уровень сигнала на протяжении всего времени, нужно нажать на имя входного сигнала, после чего выделиться вся строка, и выставить необходимое значение (рис.22).

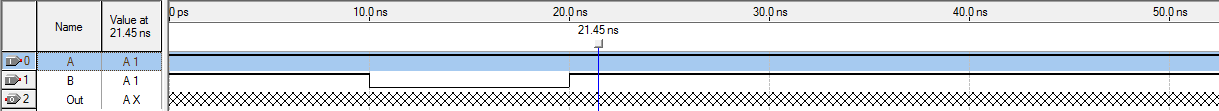


рис.22

Чтобы задать сигналы на определенных интервалах времени нужно снять выделение всей строки, щелкнув в свободном месте, если выделение есть, затем подвести курсор к левой границе такта, зажать левую кнопку мыши и провести курсор к правой границе. Границы обозначены вертикальными пунктирными линиями серого цвета (рис.23, 24).

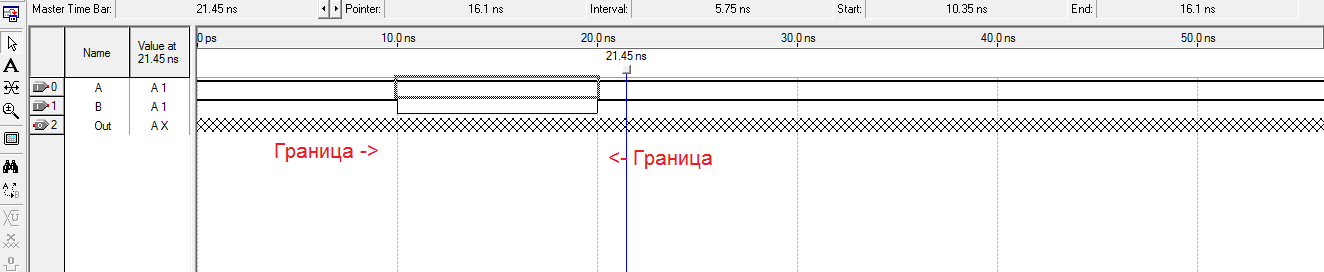


рис.23



рис.24

После выделения интервала установить нужное нам значение при помощи кнопок  и  на панели инструментов. Если нужно задать сигнал *clock*, то на нужном сигнале нажимаем правую кнопку мыши, выбираем "Value→Clock…" (рис.25).

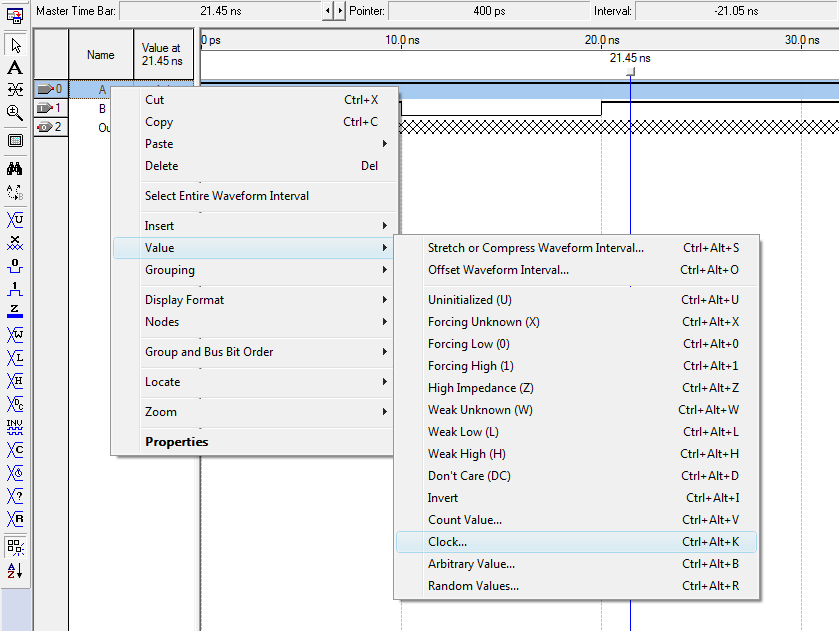


рис.25

Появится диалоговое окно (рис.26). По умолчанию время генерации сигнала стоит от 0 пс до максимального. Можно изменить эти параметры, изменив соответствующие поля Start time и End time. Также можно указать период длительности сигнала в поле Period, по умолчанию – 10 нс.

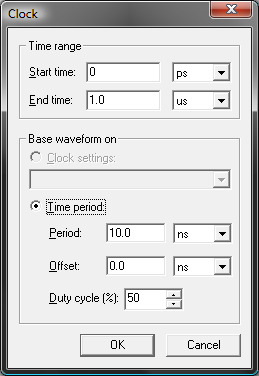


рис.26

После того как заданы все необходимые параметры – нажимаем *Ok*. Временная диаграмма будет выглядеть как на рис.27.

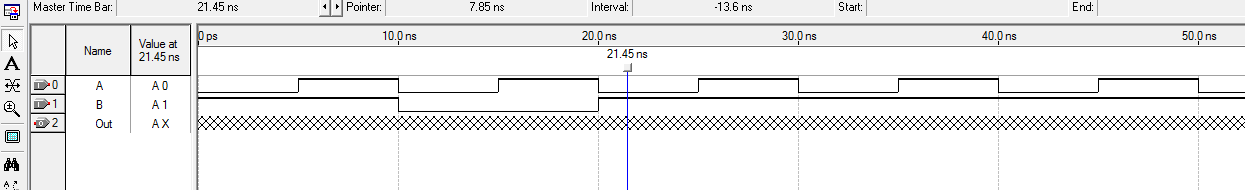


рис.27

Сохраняем проект.

Далее в меню *"Assigments → Settings→Simulator Settings"* выбираем функциональную симуляцию (симуляция происходит без учета задержки элементов, во временной – с учетом) (рис.28).



рис.28

После выбора функциональной симуляции, необходимо сгенерировать функциональный список – меню *"Processing→ Generate Functional Simulation Netlist"* .

Для запуска процесса симуляции выбрать пункт меню "Processing → Start Simulation". Если в процессе симуляции возникли ошибки, то их можно просмотреть в окне сообщений. Для вывода отчета выбрать меню "Processing → Simulation Report". Результирующие временные диаграммы изображаются в ветви "Simulator → Simulation Waveforms" (рис.29)

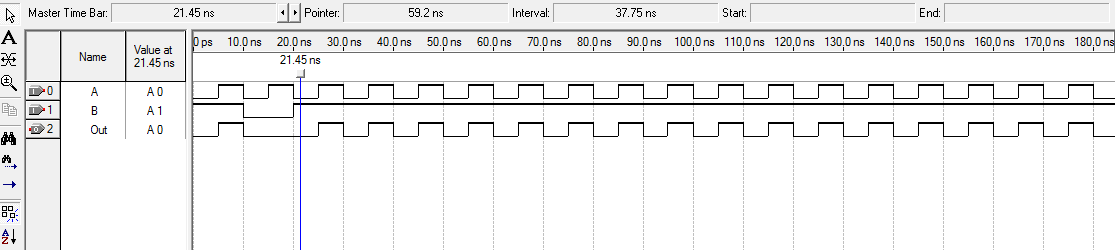


рис.29

Если в timing симуляции резултьтаты моделирования не учитывают задержку, то можно использовать такую настройку:



где прописать путь до папки с библеотекой и timing заработает нормально.

**Приложение 3**

**Подготовка и реализация проекта на стендах.**

В качестве стендов для проведения лабораторных работ могут быть использованы платы Altera DiLab, DE-0, DE2-70.

Рассмотрим подготовку к реализации проекта на стенде на примере 1– ой лабораторной работы.

1. Запускаем готовый проект: меню *" File →Open Project…" и* выбираем файл проекта первой лабораторной работы **Lab№1.qpf.**
2. Полностью компилируем проект: меню *"Processing → Start Compilation".*
3. Заходим в планировщик сигналов *"Assignments → Pin Planner"*

В появившемся окне – таблице снизу будут отображены входные и выходные пины нашего проекта (рис.30).

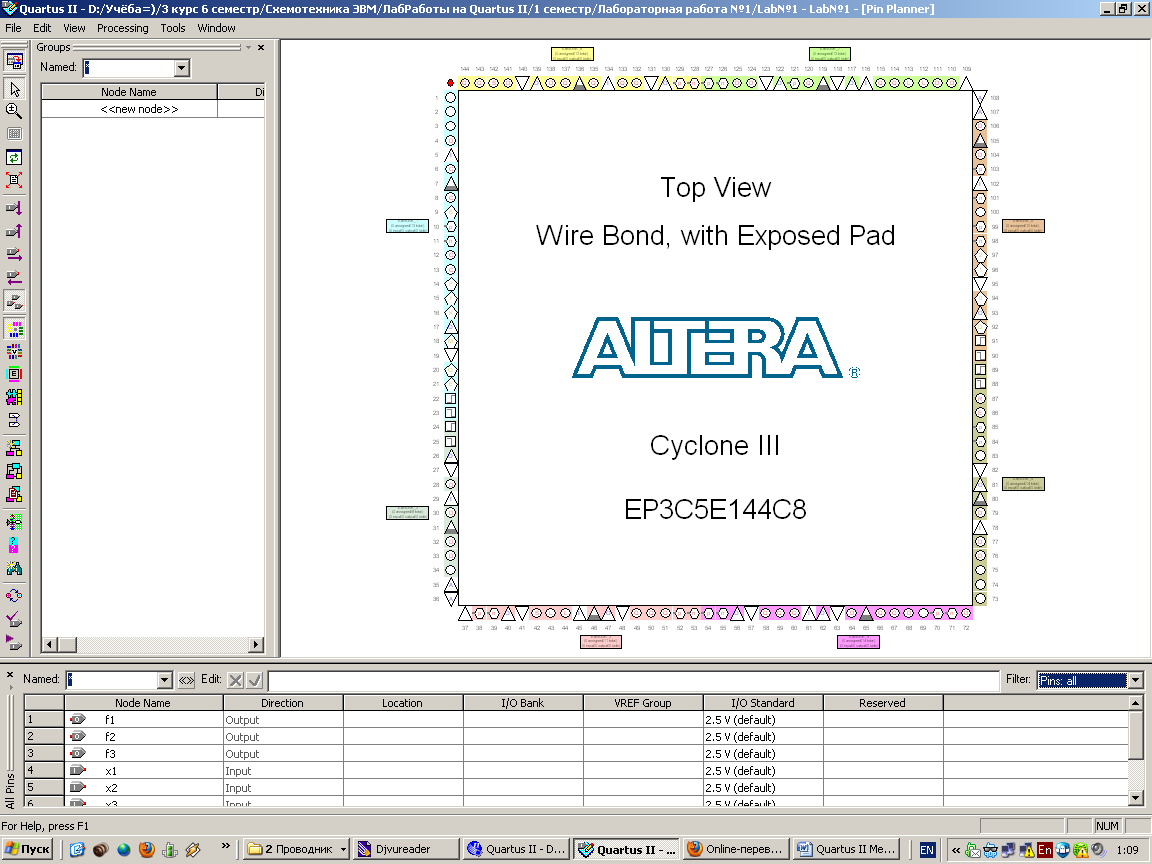


рис.30

1. В столбце *Location* для каждого пина выбираем, куда на плате будет выводиться (вводится) значение данного пина (кнопки, переключатели, светодиоды) и закрываем окно (рис.31).

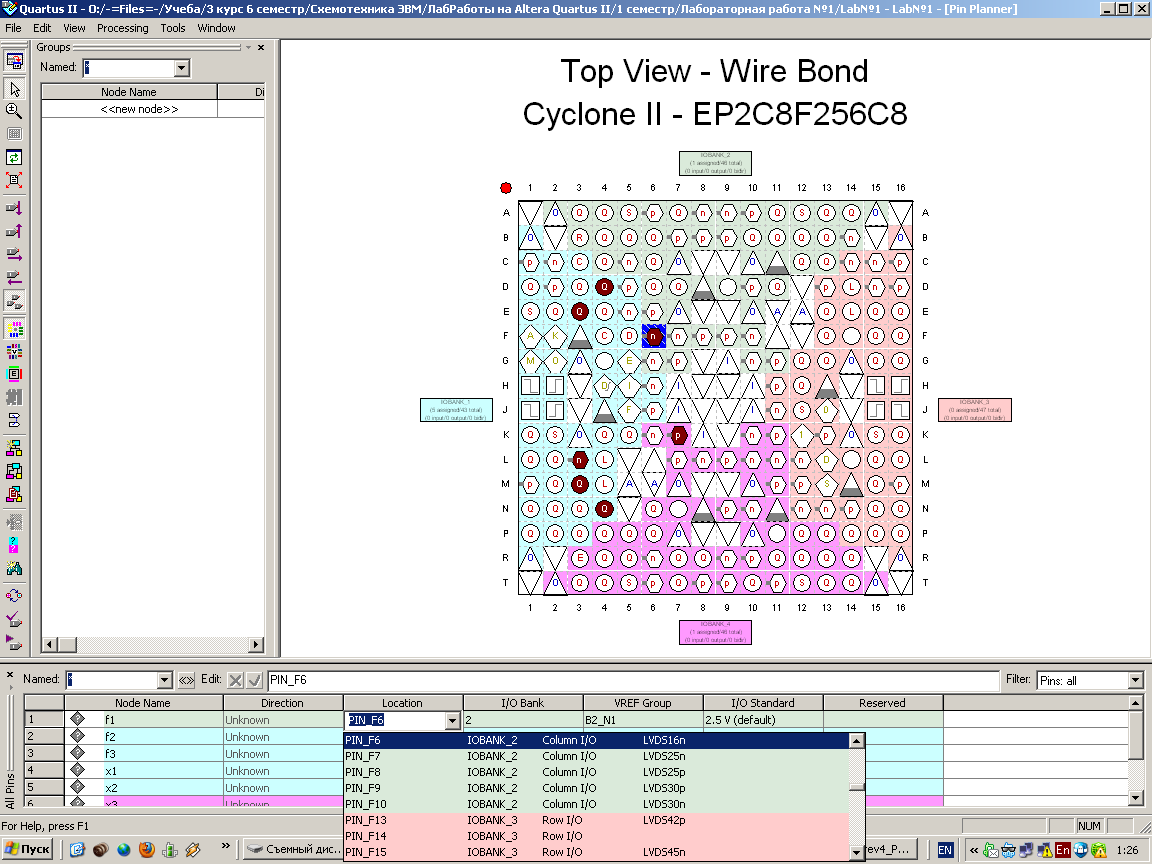


рис.31

Значения для этого столбца в зависимости от платы можно найти в методических пособиях

по платам (смотри приложение 1):

1. Если плата DE0 – методическое пособие DE0\_User\_manual.pdf
2. Если плата DE2-70 – методическое пособие DE2\_User\_manual.pdf
3. Если плата altera DiLab – методическое пособие DiLab\_PB-CII.pdf (рис.32,33,34,35)



рис.32



рис.33

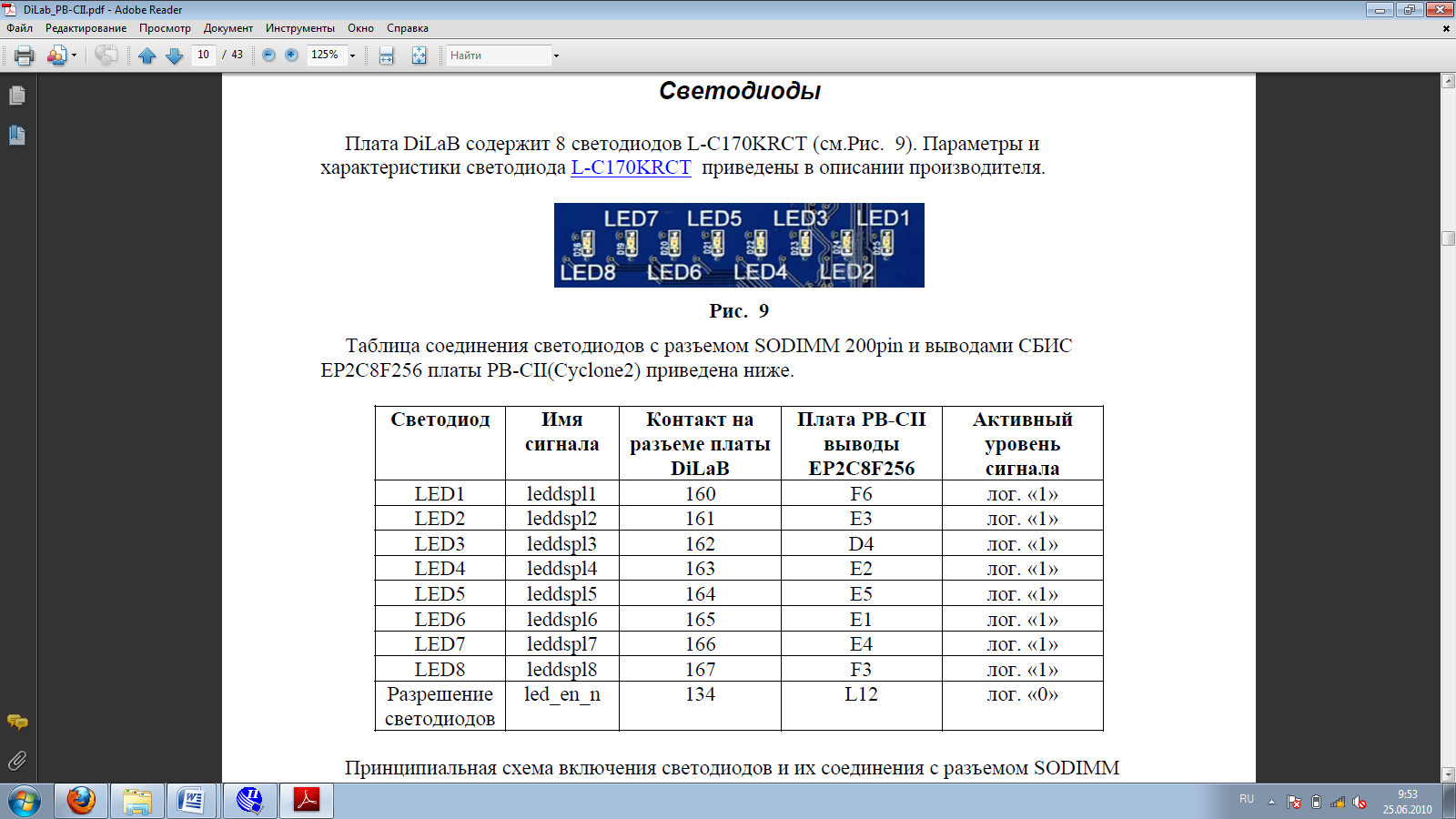


рис.34

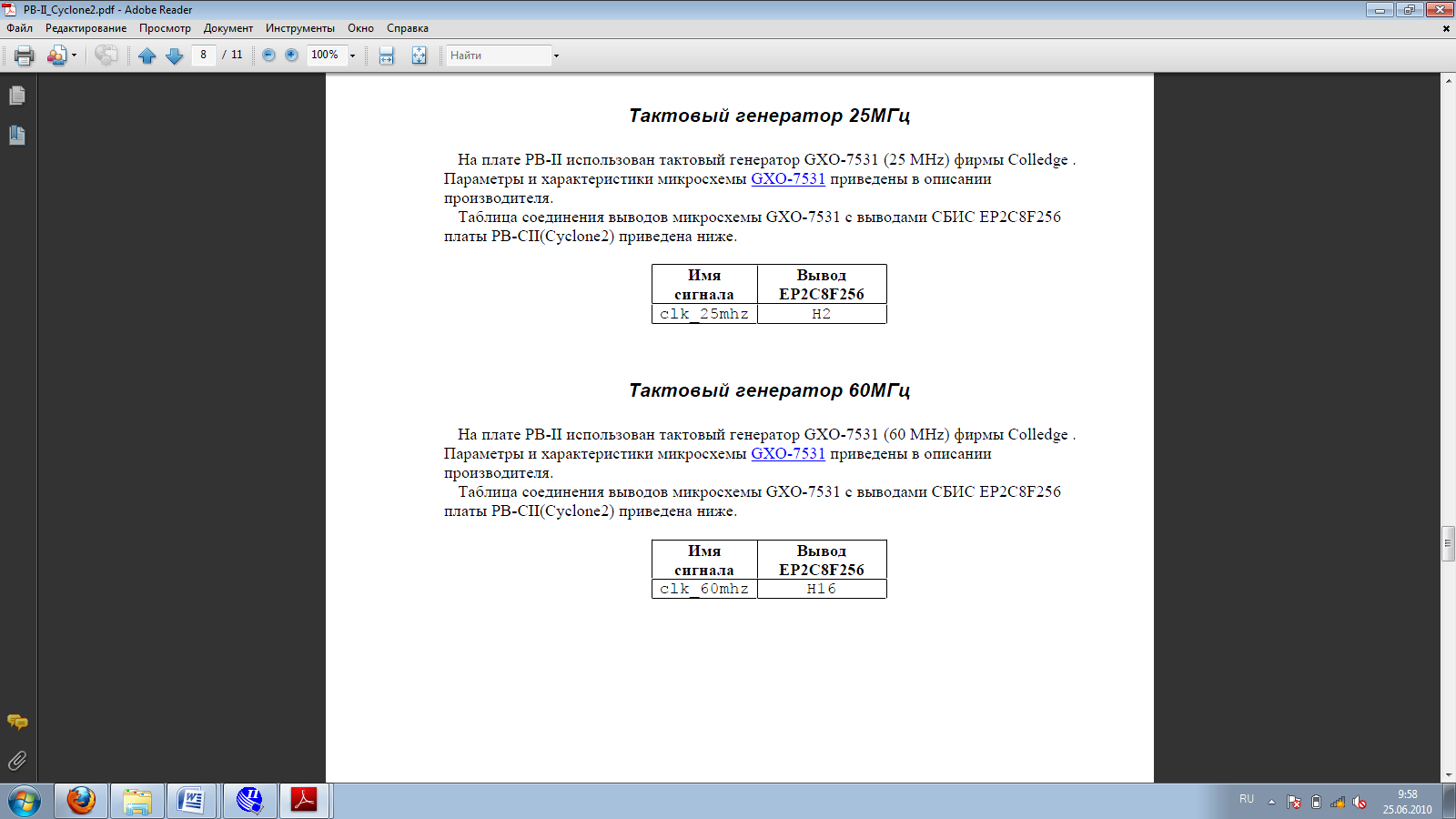


рис.35

1. Открываем меню: *"Tools → Programmer".*
2. Нажимаем на кнопку *Hardware Setup (рис. 36)*

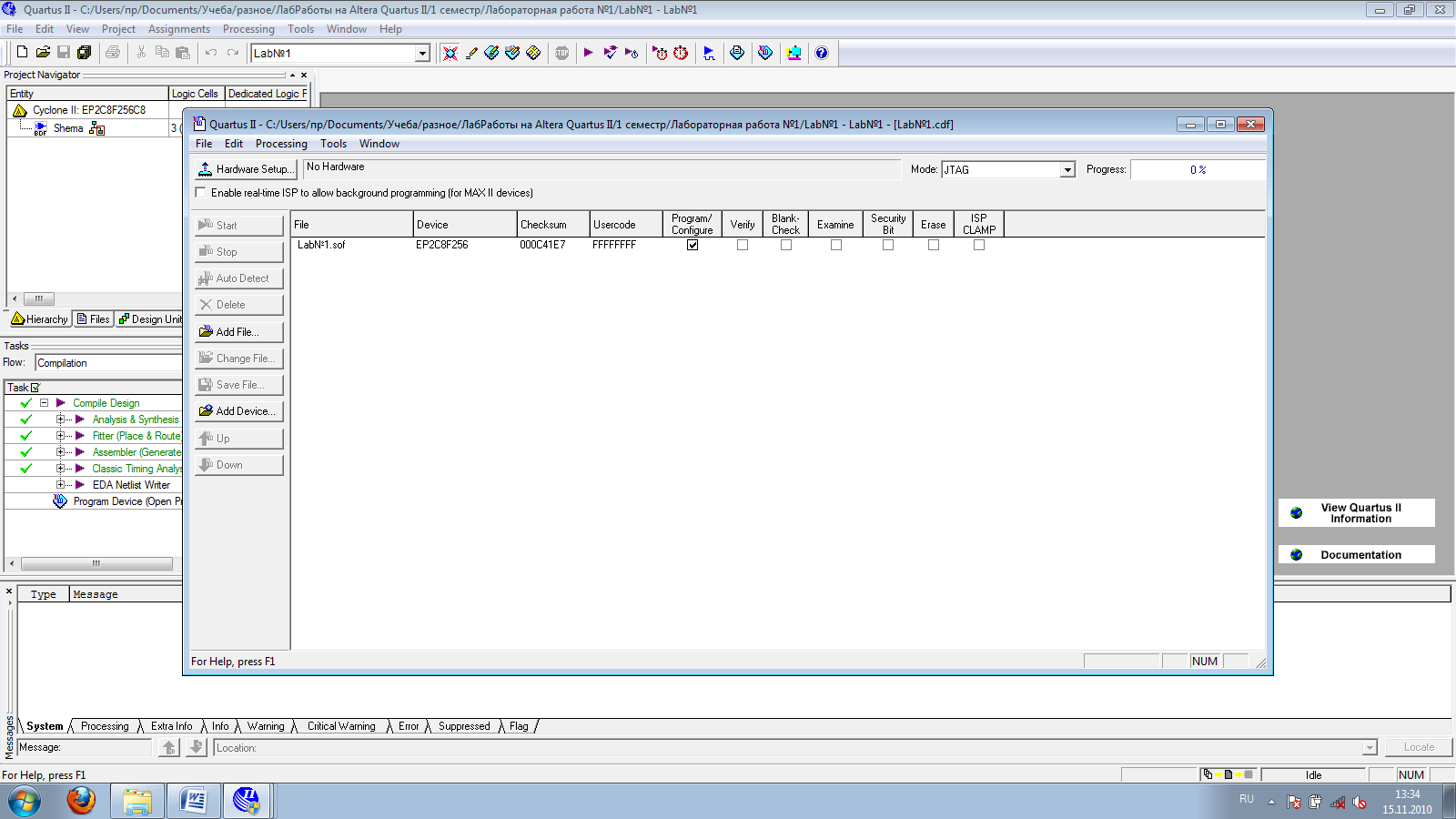


Рис. 36

1. В появившемся окне в выпадающем меню *Currently selected hardware* выбираем ***USB – Blaster*** и нажимаем *close*. (рис.37)

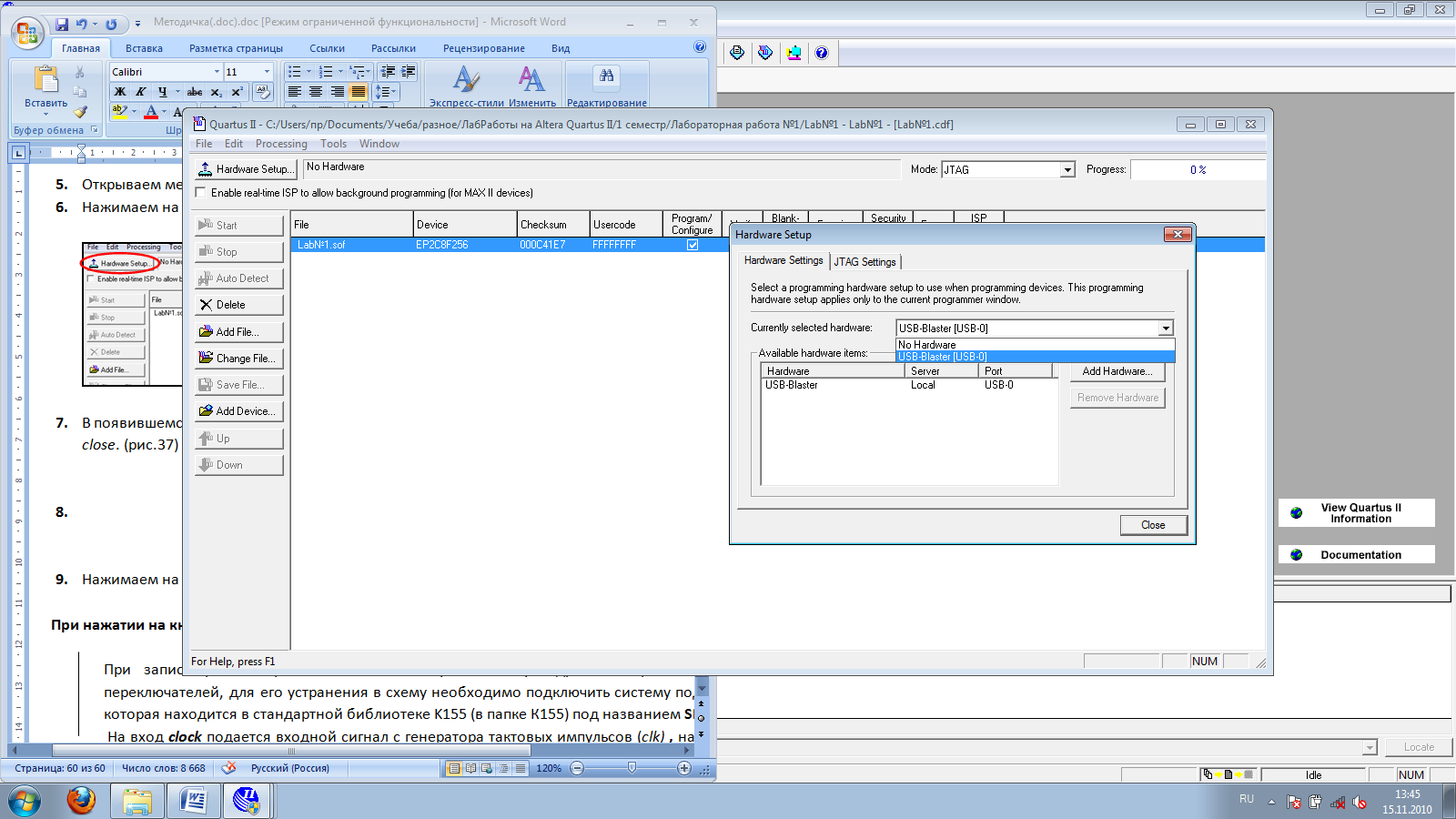


Рис.37

8. Нажимаем на кнопку *Start.* При нажатии на кнопку Start проект прошивается на плату и сразу начинает свою работу.

**Применение схемы системы подавления дребезга**

При нажатии кнопок или переключателей на плате стенда может возникнуть дребезг контактов. Для его устранения в состав схемы необходимо включить схему подавления дребезга, которая находится в стандартной библиотеке K155 (в папке К155) под названием **SPD.**

На вход ***clk*** **SPD** подается входной сигнал с генератора тактовых импульсов (*clk)* счастотой равно 25 MHz**,** на вход ***pd***  нужно подключить кнопку, у которой нужно подавить дребезг (рис.36). Выход ***out*** является отфильтрованным значением нажатой клавиши, подключенной к входу ***pd.***

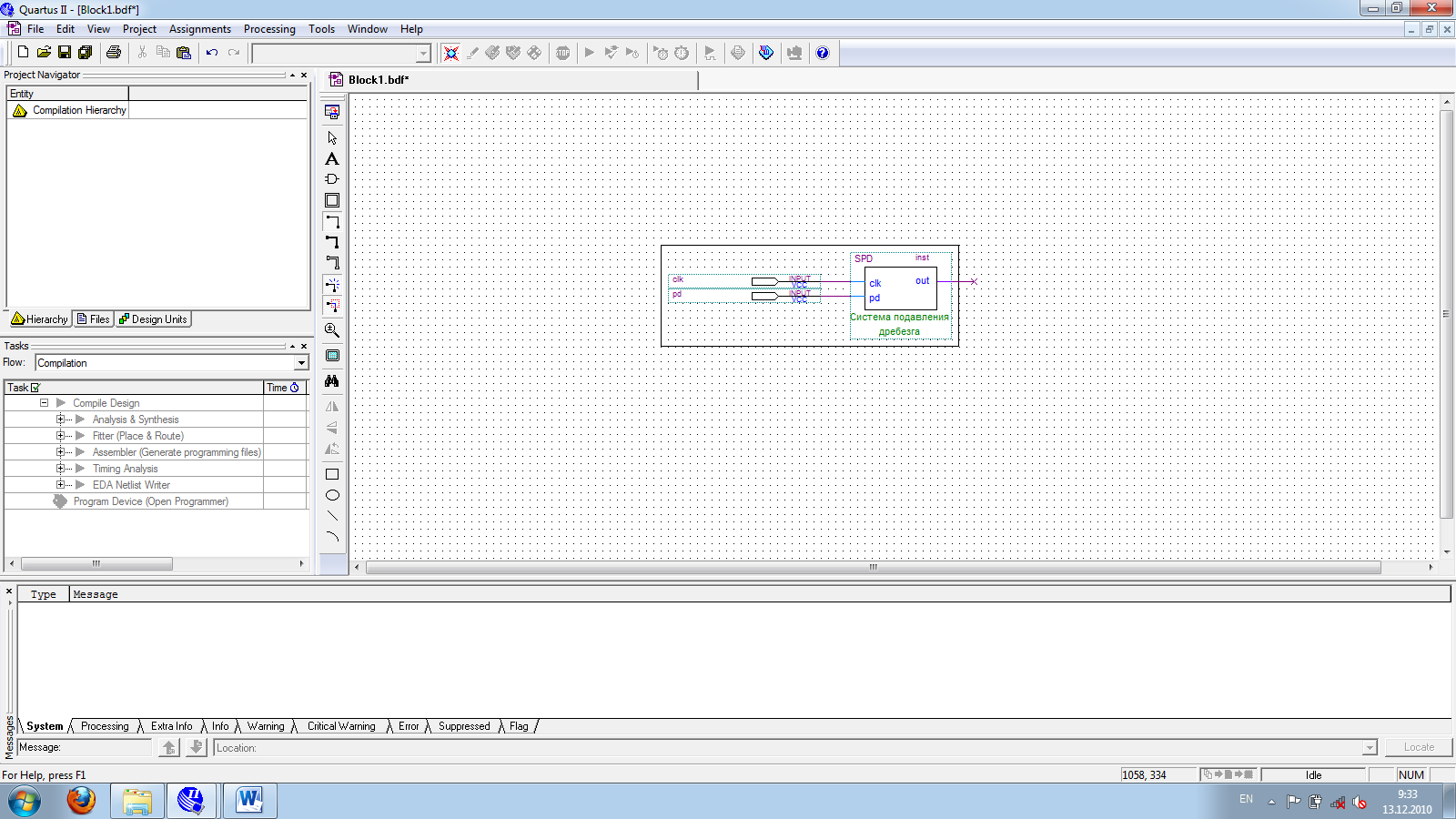


Рис.36

Схема подавления дребезга выглядит следующим образом (рис.37).

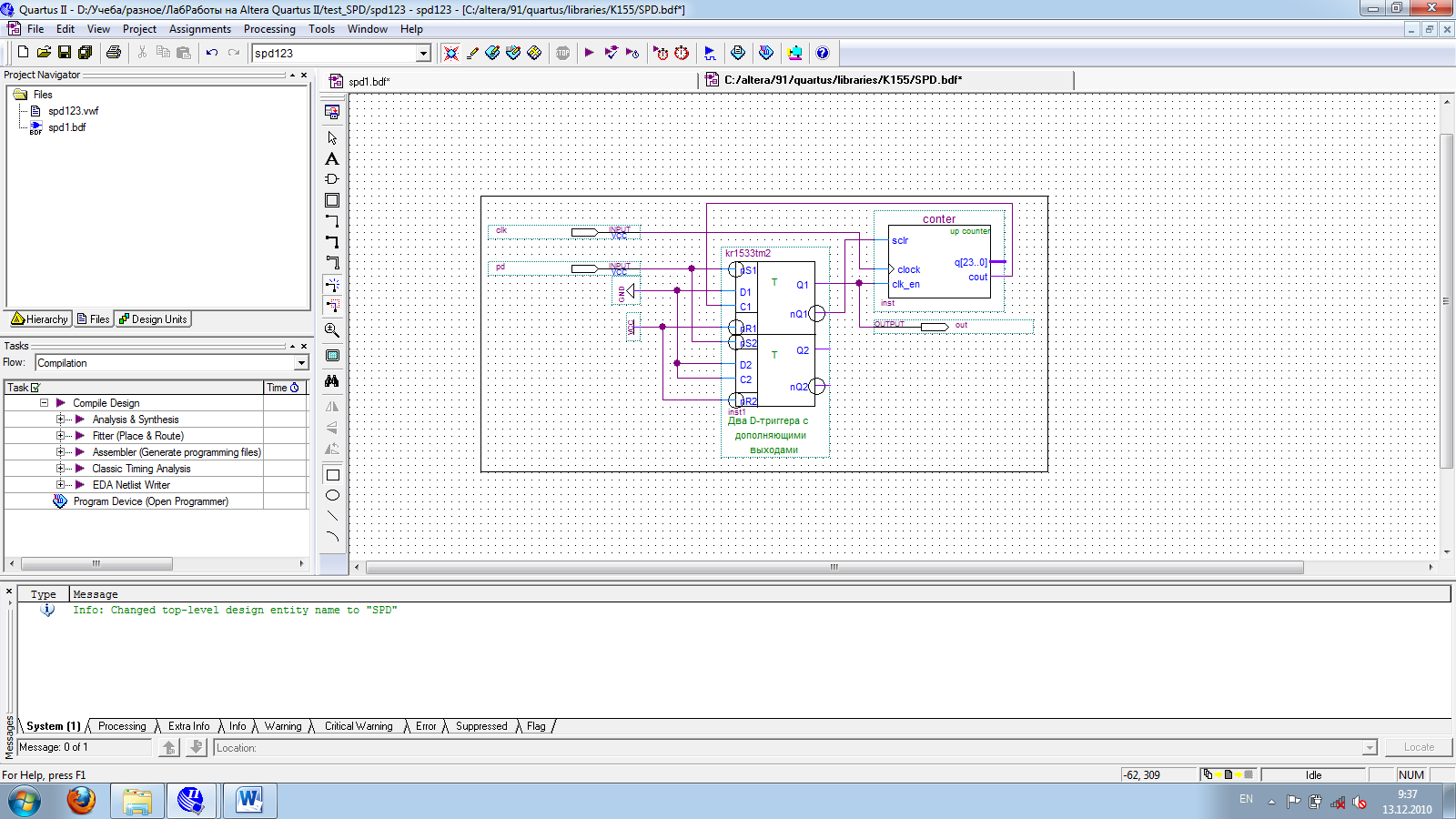


Рис. 37

Она состоит из одного элемента ТМ2 серии К155 (Два D-триггера с дополняющими выходами) и счётчика. Счётчик имеет 3 входа:

1. **Sclr –** вход синхронной очистки.
2. **Сlock –** вход для поступления тактовых импульсов.
3. **Clk-en –** вход установки тактовых импульсов.

И выход **count –** переполнение.

На вход ***pd*** поступает импульс с клавиши, который нужно отфильтровать, и попадает на вход **nS1** триггера, устанавливая его в единичное состояние. При этом на выходе триггера **Q1** установится единичное значение , на выходе **nQ1** установится нулевое значение. Сигнал с выхода **Q1** поступает на вход счётчика ***clk\_en*** и запускает счётчик. Время счета подобрано таким образом, чтобы в течениинего дребезг прекратился. При окончании счёта счетчика появляется единичный сигнал с выхода счётчика и подаётся на вход **C1** триггера, по которому триггер сбрасывается. Выход системы подавления дребезга берётся в выхода триггера **Q1.**

**Распиновка выходов для платы DiLab.**

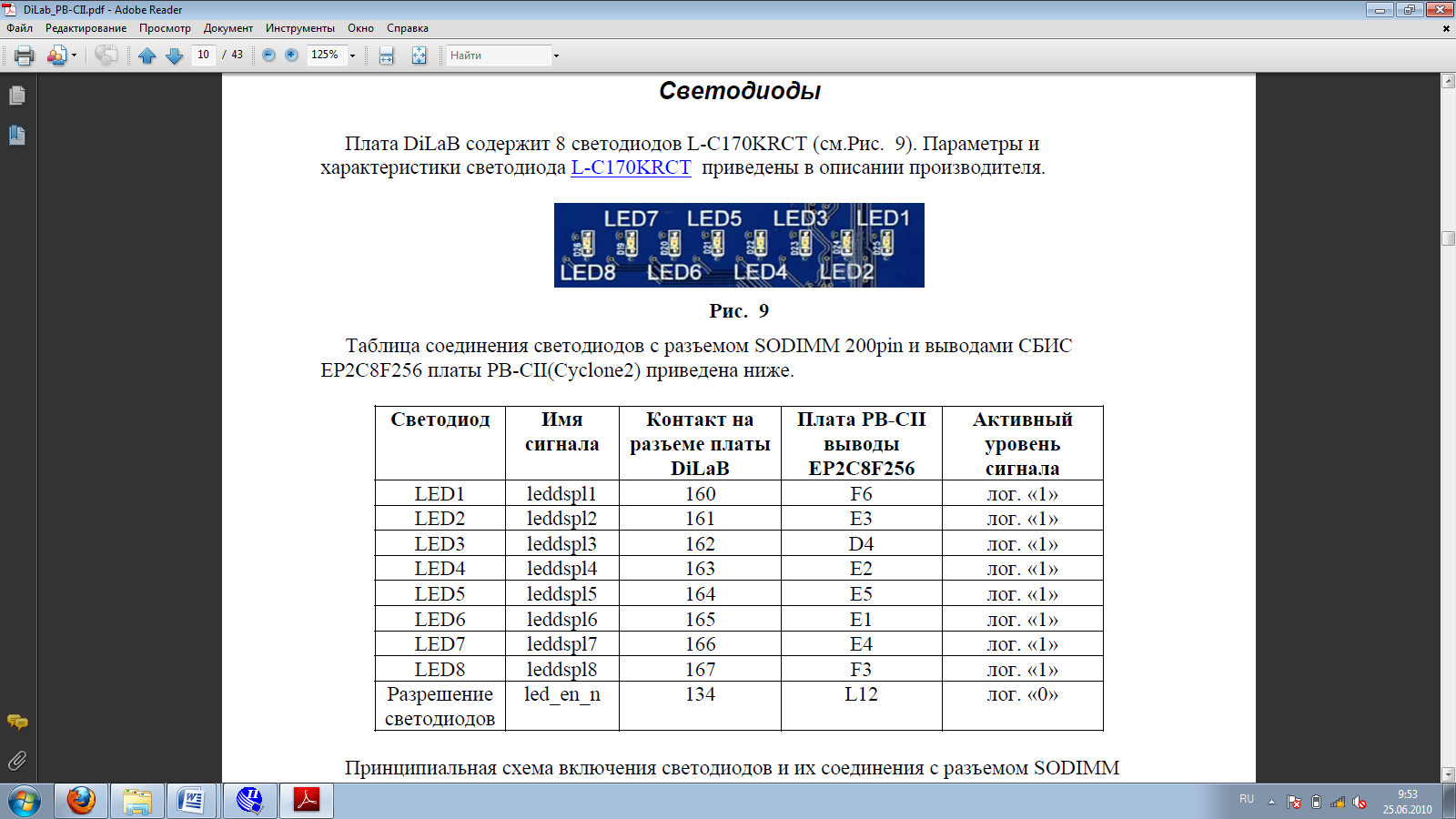
**Переключатели**



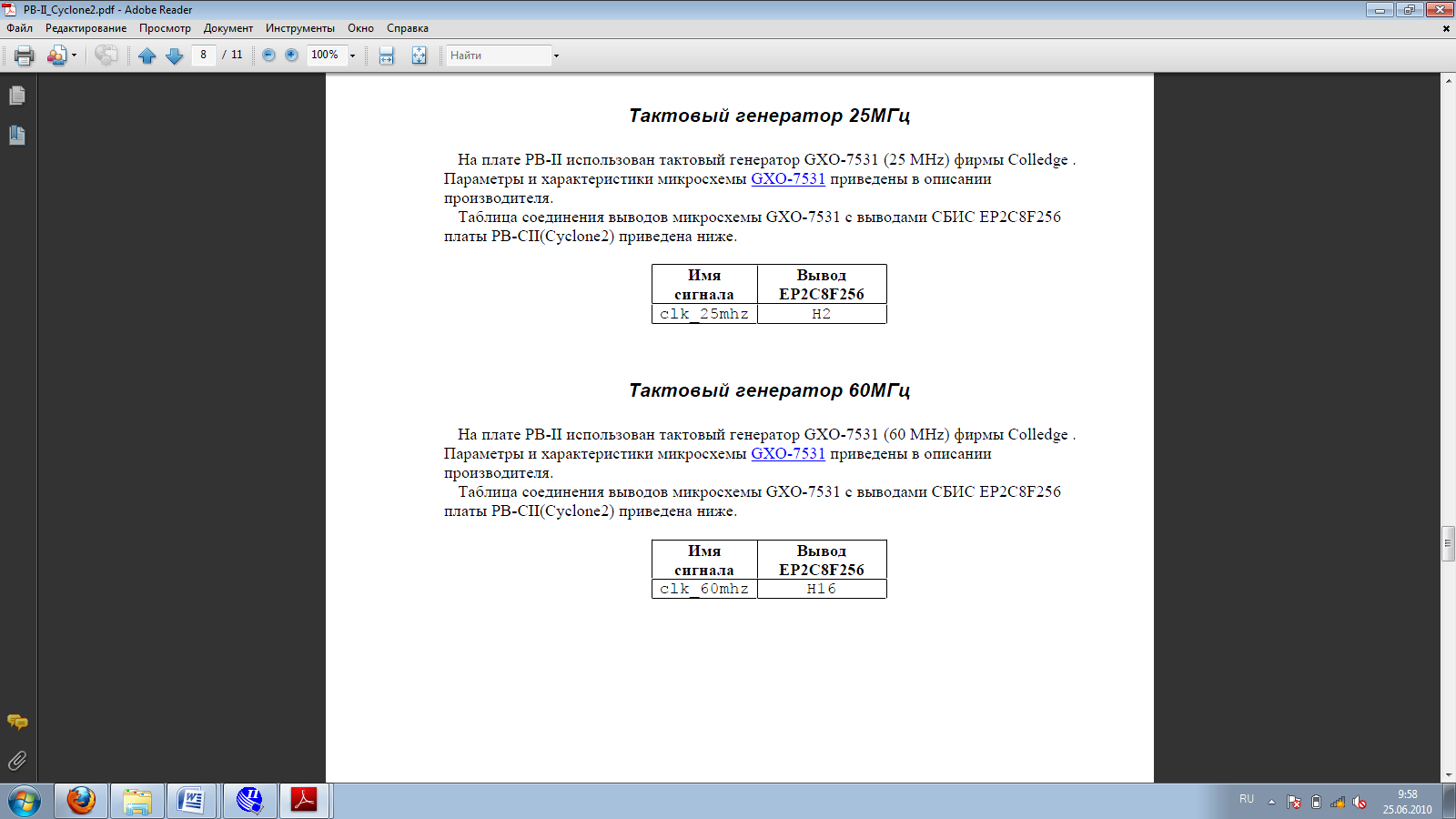
**Кнопки**



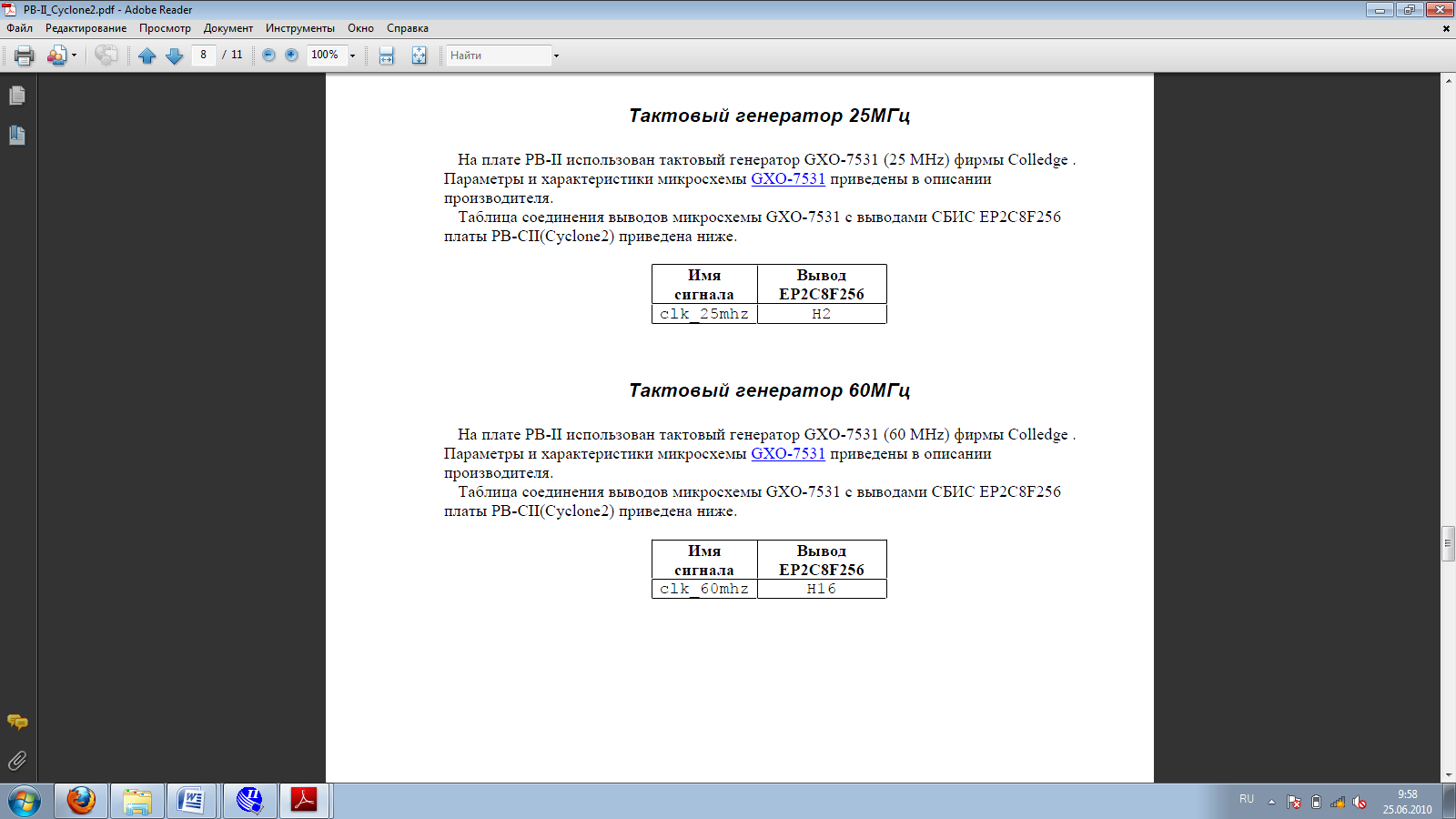
**Светодиоды**



**Тактовый генератор 25 МГц**

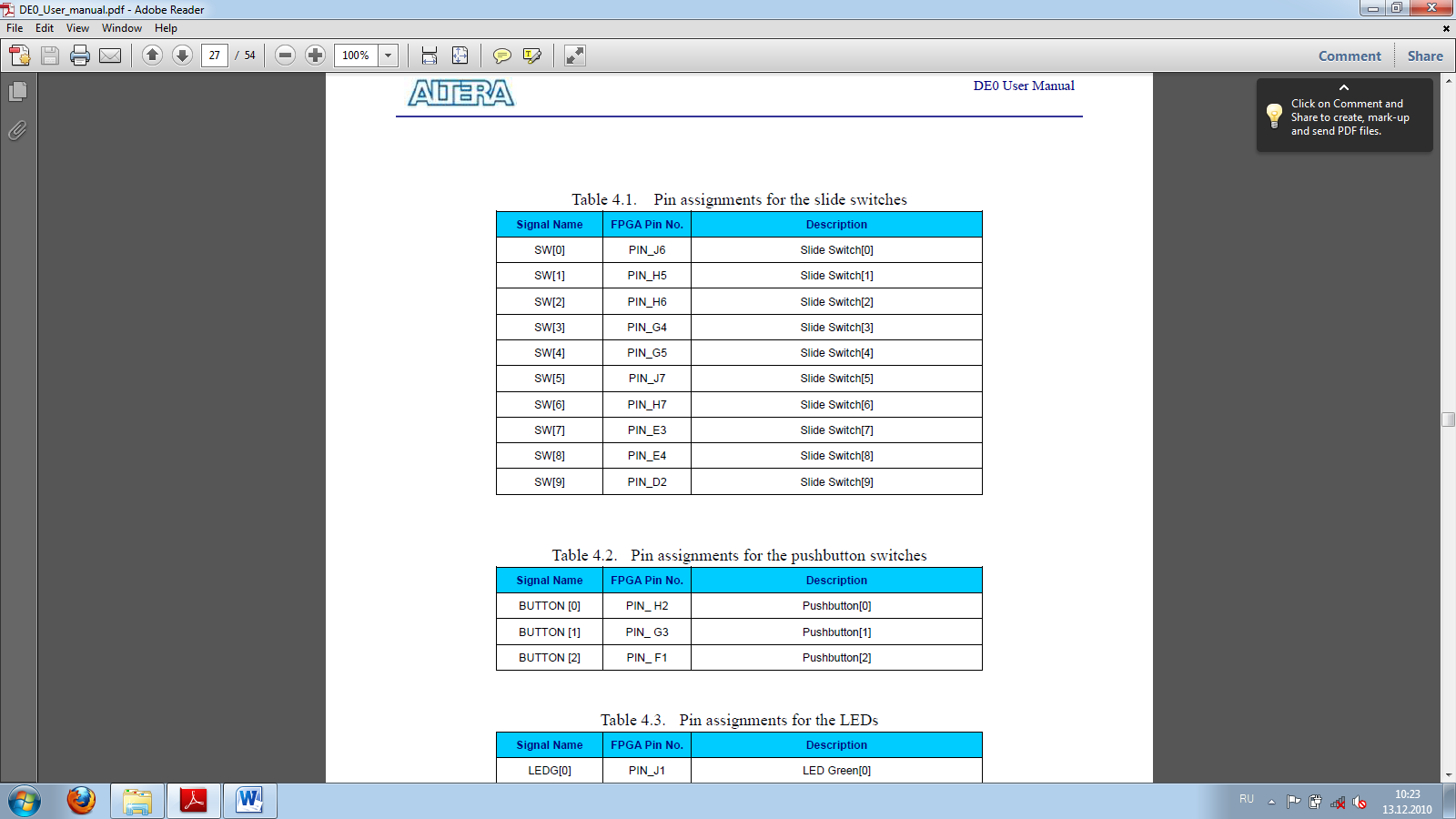


**Тактовый генератор 60 МГц**



**Распиновка выходов для платы DE0.**

**Переключатели**



**Приложение 4**

Состав библиотеки ИМС серии К155.

|  |  |
| --- | --- |
| K155LA1(K155ЛА1) | 2 элемента 4И-НЕ |
| K155LA2(K155ЛА2) | элемент 8И-НЕ |
| K155LA3(K155ЛА3) | 4 элемента 2И-НЕ |
| K155LA4(K155ЛА4) | 3 элемента 3И-НЕ |
| K155LЕ1(K155ЛЕ1) | 4 элемента 2ИЛИ-НЕ |
| K155LЕ4(K155ЛЕ4) | 3 элемента 3ИЛИ-НЕ |
| K155LI1(K155ЛИ1) | 4 элемента 2И |
| K155LI6(K155ЛИ6) | 2 элемента 4И |
| K155LL1(K155ЛЛ1) | 4 элемента 2ИЛИ |
| K155LN1(K155ЛН1) | 6 элементов НЕ |
| K155KP2(K155КП2) | Сдвоенный цифровой селектор-мультиплексор 4-1 |
| K155KP7(K155КП7) | Коммутатор 8 каналов на один со стробированием |
| K155KP16(K155КП16) | Четырёхразрядный селектор-мультиплексор 2-1 |
| K155KP18(K155КП18) | Четырёхразрядный селектор-мультиплексор 2-1 с инверсными выходами |
| K155KP19(K155КП19) | Сдвоенный селектор-мультиплексор 4х1 |
| K155TB6(K155ТВ6) | Сдвоенный JK триггер |
| K155TB9(K155ТВ9) | 2 JK-триггера со сбросом и установкой |
| K155TB10(K155ТВ10) | Сдвоенный JK триггер |
| K155TB15(K155ТВ15) | Два триггера D синхронных с дополняющими выходами |
| K155TM2(K155ТМ2) | 2 D-триггера с синхронными и дополняющими выходами |
| K155TM7(K155ТМ7) | 4 D-триггера с прямыми и инверсными выходами |
| K155TM8(K155ТМ8) | 4 D-триггера с прямыми и инверсными входами |
| K155TM9(K155ТМ9) | 6 синхронных D-триггеров |
| K155TR2(K155ТР2) | Четыре RS триггера |

**Содержание.**

1. **ЛАБОРАТОРНАЯ РАБОТА №1. …………………………………………………………... 3**
2. **ЛАБОРАТОРНАЯ РАБОТА №2. …………………………………………………………... 8**
3. **ЛАБОРАТОРНАЯ РАБОТА №3. ……………………………………………………….… 24**
4. **ЛАБОРАТОРНАЯ РАБОТА №4. …………………………………………………….…… 30**
5. **ПРИЛОЖЕНИЕ 1. Установка САПР Altera Quartus II 9.1sp2 Web Edition. …………. 38**
6. **ПРИЛОЖЕНИЕ 2. Создание и моделирование проекта на САПР Altera Quartus II. 44**
7. **ПРИЛОЖЕНИЕ 3. Подготовка и реализация проекта на стендах. …….………….… 57**
8. **ПРИЛОЖЕНИЕ 4. Состав библиотеки ИМС серии К155. …….…………………….… 63**
9. **Содержание. ……………………………………………………….……………………….… 63**

1. \*Количество вариантов реализации функции от n переменных на мультиплексоре от m адресными входами равно 

   [↑](#footnote-ref-1)