



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА - Российский технологический университет»

РТУ МИРЭА

Институт Информационных Технологий
Кафедра Вычислительной Техники (ВТ)

ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 1

«Графический ввод схемы и симуляция в САПР QUARTUS II. Описание логических схем
при помощи языка AHDL»

по дисциплине

«Архитектура вычислительных машин и систем»

Выполнил студент группы
ИКБО-13-22

Тринеев Павел Сергеевич

Принял преподаватель кафедры ВТ

Рыжова Анастасия Андреевна

Практическая работа выполнена

«__»_____2023 г.

«Зачтено»

«__»_____2023 г.

Москва 2023 г.

СОДЕРЖАНИЕ

<u>1</u>	<u>ВВЕДЕНИЕ</u>	3
<u>1.1</u>	<u>Графический ввод схемы и симуляция в САПР QUARTUS II (Q11)</u>	3
<u>1.1.1</u>	<u>Постановка задачи (Q11)</u>	3
<u>1.1.2</u>	<u>Теоретическое введение (Q11)</u>	3
<u>1.2</u>	<u>Описание логических схем при помощи языка AHDL (Q12)</u>	5
<u>1.2.1</u>	<u>Постановка задачи (Q12)</u>	5
<u>1.2.2</u>	<u>Теоретическое введение (Q12)</u>	5
<u>2</u>	<u>ХОД РАБОТЫ (Вариант 9)</u>	8
<u>2.1</u>	<u>Задания 1</u>	8
<u>2.1.1</u>	<u>Цифровая схема</u>	10
<u>2.1.2</u>	<u>Диаграмма схемы</u>	10
<u>2.1.3</u>	<u>Таблица истинности</u>	11
<u>2.2</u>	<u>Задания 2</u>	12
<u>2.2.1</u>	<u>Описание цифровой схемы на языке AHDL</u>	12
<u>2.2.2</u>	<u>Диаграмма описания</u>	13
<u>2.2.3</u>	<u>Таблица истинности</u>	14
<u>2.3</u>	<u>Сравнение результатов</u>	15
<u>3</u>	<u>ВЫВОД</u>	16

1 ВВЕДЕНИЕ

1.1 Графический ввод схемы и симуляция в САПР QUARTUS II (Q11)

1.1.1 Постановка задачи (Q11)

Спроектировать логическую схему при помощи графического редактора САПР QUARTUS II. Исследовать работу схемы с использованием сигнального редактора САПР QUARTUS II.

1.1.2 Теоретическое введение (Q11)

Алгебра логики и основные логические элементы

Математической основой цифровой электроники и вычислительной техники является алгебра логики или булева алгебра (по имени английского математика Джона Буля).

В булевой алгебре независимые переменные или аргументы (X) принимают только два значения: «0» или «1». Зависимые переменные или функции (Y) также могут принимать только два значения: «0» или «1». Функция алгебры логики (ФАЛ) представляется в виде:

$$Y = F(X_1; X_2; X_3 \dots X_n)$$

Данная форма задания ФАЛ называется алгебраической.

Основными логическими функциями являются:

- логическое отрицание (инверсия):

$$Y = \bar{X}$$

- логическое сложение (дизъюнкция):

$$Y = X_1 + X_2 \text{ или } Y = X_1 \vee X_2$$

- логическое умножение (конъюнкция):

$$Y = X_1 \cdot X_2 \text{ или } Y = X_1 \wedge X_2$$

К более сложным функциям алгебры логики относятся:

- функция равнозначности (эквивалентности):

$$Y = X_1 \cdot X_2 + \overline{X_1} \cdot \overline{X_2} \text{ или } Y = X_1 \sim X_2$$

- функция неравнозначности (сложение по модулю два):

$$Y = X_1 \cdot \overline{X_2} + \overline{X_1} \cdot X_2 \text{ или } Y = X_1 \oplus X_2$$

- функция Пирса (логическое сложение с отрицанием):

$$Y = \overline{X_1 + X_2}$$

- функция Шеффера (логическое умножение с отрицанием):

$$Y = \overline{X_1 \cdot X_2}$$

Для булевой алгебры справедливы следующие законы и правила:

- распределительный закон:

$$X_1(X_2 + X_3) = X_1 \cdot X_2 + X_1 \cdot X_3$$

$$X_1 + (X_2 \cdot X_3) = (X_1 + X_2) \cdot (X_1 + X_3)$$

- правило повторения:

$$X \cdot X = X, X + X = X$$

- правило отрицания:

$$X \cdot \overline{X} = 0, X + \overline{X} = 1$$

- теорема де Моргана:

$$\overline{X_1 + X_2} = \overline{X_1} \cdot \overline{X_2}, \quad \overline{X_1 \cdot X_2} = \overline{X_1} + \overline{X_2}$$

- тождественности:

$$X \cdot 1 = X, X + 0 = X, \quad X \cdot 0 = 0, X + 1 = 1$$

Схемы, реализующие логические функции, называются логическими элементами. Основные логические элементы имеют, как правило, один выход (Y) и несколько входов, число которых равно числу аргументов ($X_1; X_2; X_3; \dots X_n$).

На электрических схемах логические элементы рисуют в виде прямоугольников с выводами для входных (слева) и выходных (справа)

переменных. В середине прямоугольника изображается символ, обозначающий функциональное назначение элемента.

Логические элементы используются для построения интегральных микросхем, которые выполняют разнообразные логические и арифметические операции.

ФАЛ любой сложности можно реализовать при помощи обозначенных логических элементов. В качестве примера рассмотрим ФАЛ, заданную в алгебраической форме, в виде:

$$Y = X_1 + \overline{X_2} \cdot X_3$$

Для реализации заданной функции на элементах «И-НЕ» используют двойную инверсию функции (теорему де Моргана):

$$Y = \overline{\overline{Y}} = \overline{\overline{X_1 + \overline{X_2} \cdot X_3}} = \overline{\overline{X_1} \cdot (\overline{\overline{X_2} \cdot X_3})}$$

1.2 Описание логических схем при помощи языка AHDL (Q12)

1.2.1 Постановка задачи (Q12)

Приобретение основных навыков описания цифровых схем с помощью языка описания аппаратуры AHDL. Смоделировать логическую схему при помощи текстового редактора САПР QUARTUS II.

1.2.2 Теоретическое введение (Q12)

Язык описания аппаратуры AHDL разработан фирмой Altera и предназначен для описания комбинационных и последовательностных логических устройств, групповых операций, цифровых автоматов (state machine) и таблиц истинности с учетом архитектурных особенностей ПЛИС фирмы Altera. Он полностью интегрируется с системой автоматизированного проектирования ПЛИС QUARTUS II. Файлы описания аппаратуры, написанные на языке AHDL, имеют расширение *.TDF (Text design file). Для создания TDF-файла можно использовать как текстовый редактор системы QUARTUS II, так и любой другой. Проект, выполненный в виде TDF-файла,

компилируется, отлаживается и используется для формирования файла программирования или загрузки ПЛИС фирмы Altera.

Операторы и элементы языка AHDL являются достаточно мощным и универсальным средством описания алгоритмов функционирования цифровых устройств, удобным в использовании. Язык описания аппаратуры AHDL дает возможность создавать иерархические проекты в рамках одного этого языка или же в иерархическом проекте использовать как TDF-файлы, разработанные на языке AHDL, так и другие типы файлов.

При распределении ресурсов устройств разработчик может пользоваться командами текстового редактора или операторами языка AHDL для того, чтобы сделать назначения ресурсов и устройств. Кроме того, разработчик может только проверить синтаксис или выполнить полную компиляцию для отладки и запуска проекта. Любые ошибки автоматически обнаруживаются обработчиком сообщений и высвечиваются в окне текстового редактора.

Элементы языка AHDL. Зарезервированные ключевые слова.

Зарезервированные ключевые слова используются для следующих целей:

- для обозначения начала, конца и переходов в объявлениях языка AHDL;
- для обозначения предопределенных констант, т.е. GND и VCC.

Ключевые слова можно использовать, как символические имена, только если они заключены в символы одинарных кавычек ('). Их можно также использовать в комментариях.

Имена в кавычках и без кавычек

В языке AHDL есть три типа имен:

- Символические имена – это определяемые пользователем идентификаторы. Они используются для обозначения следующих частей TDF:
 - внутренних и внешних узлов (вершин);
 - констант;

- переменных цифрового автомата, битов состояний, имен состояний;
 - примеров (Instance).
- Имена подпроекта(модуля) — это определяемые пользователем имена для файлов проекта более низкого уровня. Имя подпроекта должно быть таким же, как имя файла TDF.
- Имена портов — это символические имена, идентифицирующие вход или выход примитива или макрофункции.

В файле .fit проекта могут появиться генерируемые компилятором имена выводов, с символом “тильда” (~). Этот символ зарезервирован для имен, генерируемых компилятором, пользователю запрещается его использовать для обозначения имен выводов, узлов (вершин), групп (шин). Существуют две формы записи для всех трех типов имен (символических, подпроекта и портов): в кавычках (') и без кавычек.

Если разработчик создает символ по умолчанию для файла TDF, который включает в себя имена портов в кавычках, собственно кавычки не входят в имена выводов.

2 ХОД РАБОТЫ (Вариант 9)

2.1 Задания 1

- 1) Изучить правила построения, принцип работы логических схем.
- 2) Синтезировать электрическую принципиальную схему логического устройства, описанного заданным преподавателем уравнением в алгебраической форме.
- 3) Нарисовать синтезированную схему в графическом редакторе САПР QUARTUS II.
- 4) Произвести симуляцию работы схемы. Зарисовать диаграммы работы и по ее результатам заполнить таблицу истинности смоделированной схемы.
- 5) Ответить на контрольные вопросы, оформить отчет о выполненной работе.

Вариант 9: $Y = CD(A + B) + AB(C + D)$

Таблица 1. Таблица истинности для функции варианта

A	B	C	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

2.1.1 Цифровая схема

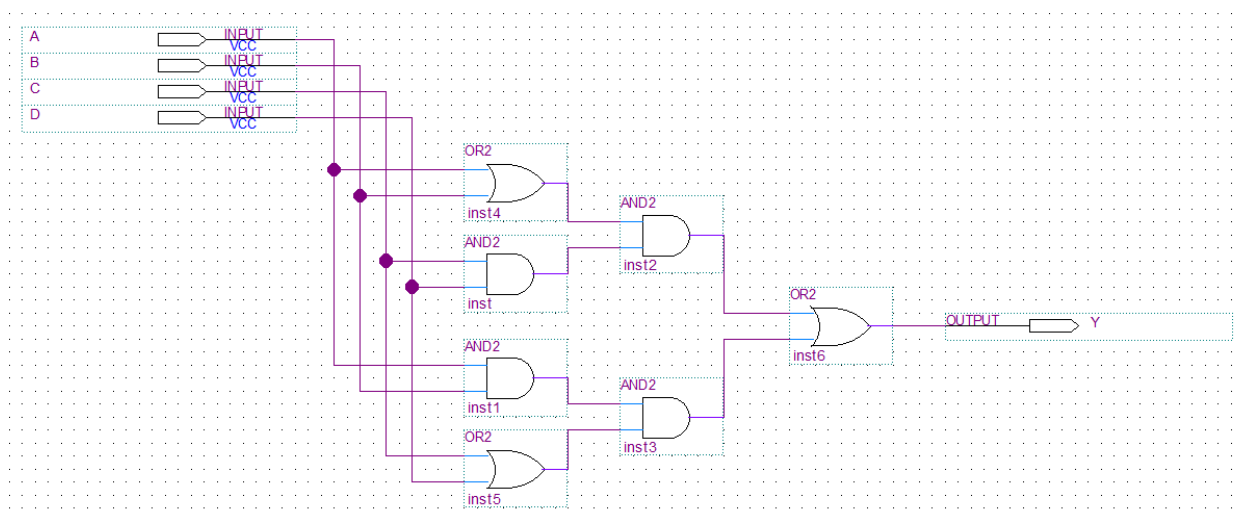


Рисунок 1. Реализация цифровой схемы логического выражения

2.1.2 Диаграмма схемы

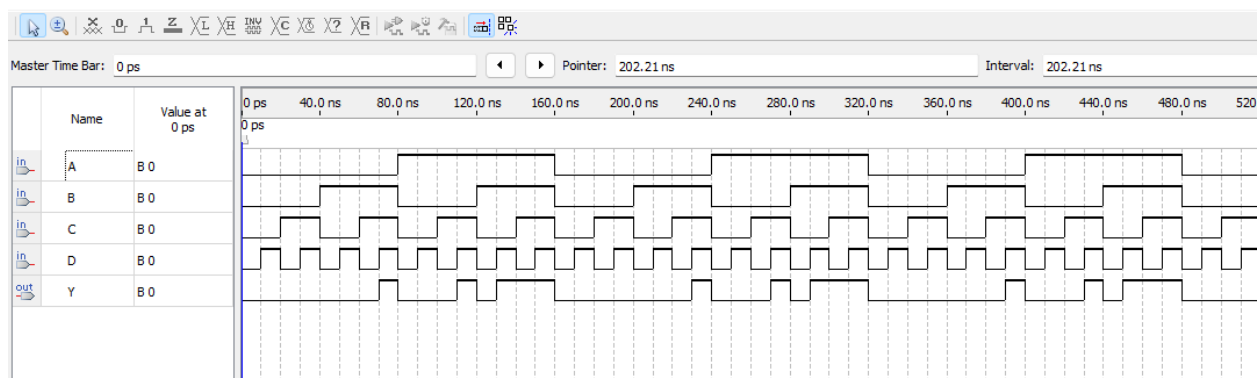


Рисунок 2. Диаграмма полученной цифровой схемы

2.1.3 Таблица истинности

Таблица 2. Таблица истинности по диаграмме цифровой схемы

A	B	C	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

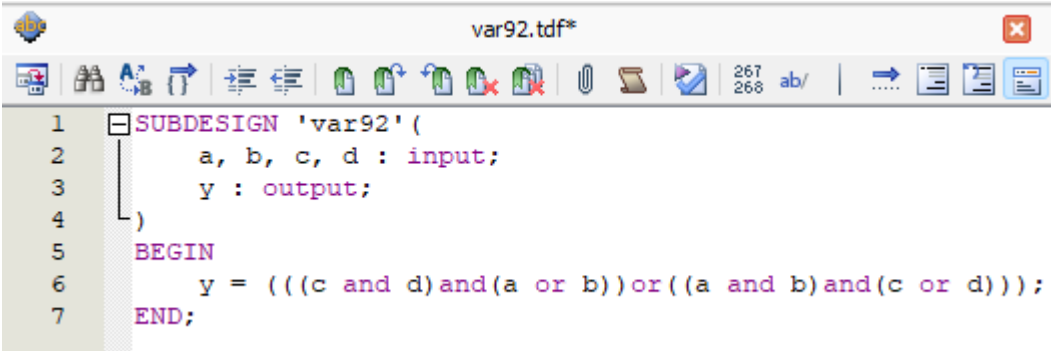
2.2 Задания 2

- 1) Изучить основные элементы языка AHDL и правила описания логических схем.
- 2) Сделать описание электрической схемы, заданной в предыдущей работе при помощи текстового редактора САПР QUARTUS II.
- 3) Произвести симуляцию работы схемы. Зарисовать диаграммы работы и по ее результатам заполнить таблицу истинности смоделированной схемы.
- 4) Сравнить результаты, полученные в ходе выполнения лабораторной работы с результатами, полученными в работе №1.
- 5) Ответить на контрольные вопросы, оформить отчет о выполненной работе.

2.2.1 Описание цифровой схемы на языке AHDL

Код программы на языке AHDL описывает логическую функцию, которая принимает четыре входных сигнала a, b, c и d, и выдаёт один выходной сигнал y.

Внутри блока BEGIN...END определена логическая функция, которая вычисляется на основе входных сигналов и применяет операции И (AND), ИЛИ (OR) и скобки для определения выходного сигнала y.



```
1 SUBDESIGN 'var92'(  
2     a, b, c, d : input;  
3     y : output;  
4 )  
5 BEGIN  
6     y = ((c and d) and (a or b)) or ((a and b) and (c or d));  
7 END;
```

Рисунок 3. Код программы

2.2.2 Диаграмма описания

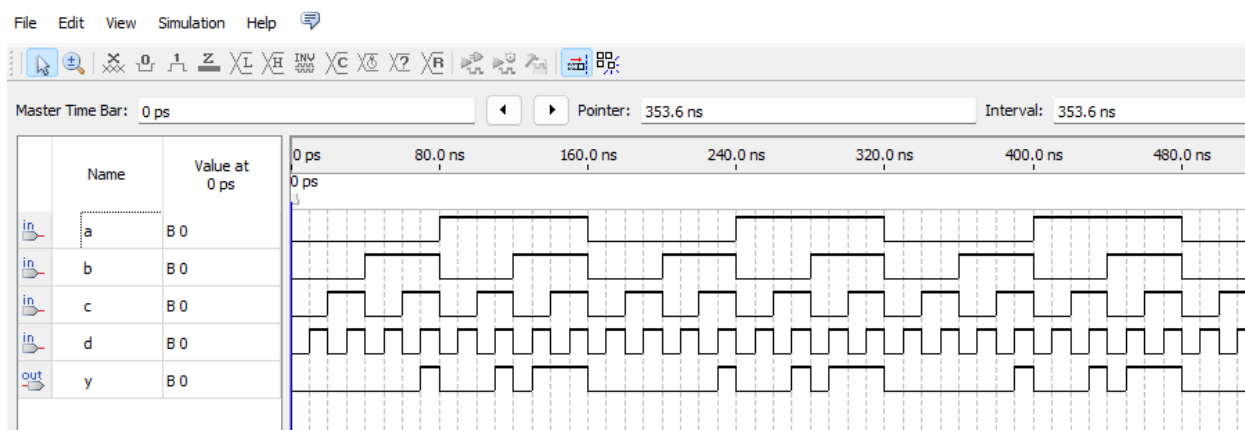


Рисунок 4. Диаграмма полученного описания

2.2.3 Таблица истинности

Таблица 3. Таблица истинности по диаграмме описания

A	B	C	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

2.3 Сравнение результатов

При выполнении работы были получены три таблицы истинности, составленные:

- по логическому выражению;
- по диаграмме цифровой схемы;
- по диаграмме описания.

Сравнивая все три таблицы, можно заключить, что цифровая схема и описание к ней на языке AHDL соответствуют первоначальной таблице логического выражения, что говорит о правильно выполненной работы.

3 ВЫВОД

В ходе выполнения задачи по проектированию логической схемы с применением графического редактора САПР QUARTUS II, а также исследованию работы данной схемы с использованием сигнального редактора САПР QUARTUS II, были достигнуты следующие результаты:

- Разработана логическая схема, отражающая заданные логические операции, с использованием инструментов и возможностей, предоставляемых САПР QUARTUS II.
- В ходе проектирования были использованы различные элементы, такие как логические вентили, мультиплексоры, и другие логические блоки, чтобы создать желаемую логическую функцию.
- После завершения проектирования была проведена проверка схемы на наличие ошибок и недоразумений, чтобы гарантировать правильность ее работы.
- Затем был осуществлен анализ схемы с использованием сигнального редактора САПР QUARTUS II. Этот этап включал в себя симуляцию работы схемы, а также анализ сигналов, прохождение сигнала через различные компоненты схемы, и выявление возможных проблем.
- Результаты анализа позволили убедиться в корректности работы схемы и выявить возможные улучшения или оптимизации.

Итак, проектирование и исследование логической схемы с использованием САПР QUARTUS II позволили успешно создать и проверить работоспособность логической схемы, что является важным шагом в разработке цифровых устройств и систем.

В результате приобретения основных навыков описания цифровых схем с использованием языка описания аппаратуры AHDL и последующей моделирования логической схемы при помощи текстового редактора САПР QUARTUS II, были получены следующие выводы:

- Овладение языком описания аппаратуры AHDL является важным шагом при проектировании и моделировании цифровых схем. Этот навык позволяет описывать логику и функциональность цифровых компонентов с высокой степенью контроля и гибкости.
- Моделирование логической схемы с использованием текстового редактора в САПР QUARTUS II предоставляет возможность создавать и анализировать цифровые схемы на более глубоком уровне. Это позволяет более точно контролировать функциональность и производительность цифровых устройств.
- В ходе работы была разработана логическая схема с использованием AHDL, которая включала в себя различные компоненты, описанные на языке AHDL.
- После создания схемы был проведен анализ ее работы с использованием инструментов САПР QUARTUS II. Это включало в себя симуляцию и проверку схемы на наличие ошибок или недоразумений.
- Полученные навыки описания цифровых схем и моделирования с их помощью в САПР QUARTUS II могут быть применены в процессе разработки цифровых устройств, что является важным компонентом в области цифровой электроники.

Итак, приобретение навыков описания цифровых схем с использованием AHDL и моделирование их в САПР QUARTUS II представляют собой важные этапы в процессе разработки и анализа цифровых систем, а также позволяют более эффективно и точно выполнять цифровые проекты.