



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«МИРЭА - Российский технологический университет»

РТУ МИРЭА

Институт Информационных Технологий
Кафедра Вычислительной Техники (ВТ)

ОТЧЁТ ПО ЛАБОРАТОРНЫМ РАБОТАМ

по дисциплине

«Архитектура вычислительных машин и систем»

Выполнил студент группы
ИКБО-30-22

Сенькевич Гаврил Дмитриевич

Принял преподаватель кафедры ВТ

Воронов Дмитрий Юрьевич

Москва 2023 г.

Содержание

Лабораторная работа №1. Часть 1	3
Лабораторная работа №1. Часть 2.....	7
Лабораторная работа №2. Часть 1	10
Лабораторная работа №2. Часть 2.....	12
Лабораторная работа №3	14

Лабораторная работа №1. Часть 1

Введение

1. Цель работы

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

2. Основные теоретические сведения

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».

Основными логическими функциями являются:

- логическое отрицание (инверсия);
- логическое сложение (дизъюнкция);
- логическое умножение (конъюнкция).

К более сложным функциям алгебры логики относятся:

- функция равнозначности (эквивалентности);
- функция неравнозначности (сложение по модулю два);
- функция Пирса (логическое сложение с отрицанием);
- функция Шеффера (логическое умножение с отрицанием).

Для булевой алгебры справедливы следующие законы и правила:

- распределительный закон;
- правило повторения;
- правило отрицания;
- теорема де Моргана.

Схемы, реализующие логические функции, называются логическими элементами. Основные логические элементы имеют, как правило, один выход и несколько входов, число которых равно числу аргументов.

Ход работы

В ходе выполнения лабораторной работы была реализована логическая функция, определённая персональным вариантом №20:

$$Y = ABC + \overline{BD} + \overline{AD}.$$

Таблица истинности для этой функции представлена в таблице 1.

Таблица 1 – Таблица истинности функции Y.

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

Логическая схема функции Y представлена на рисунке 1.

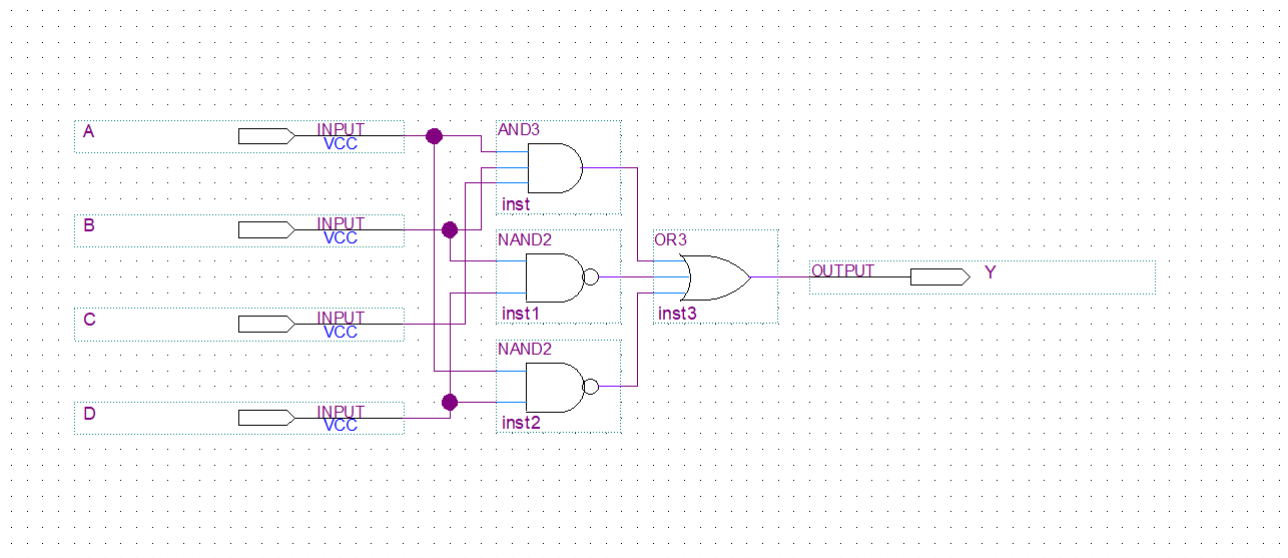


Рисунок 1 – Логическая схема функции Y.

Построение симуляции работы схемы представлено на рисунке 2.

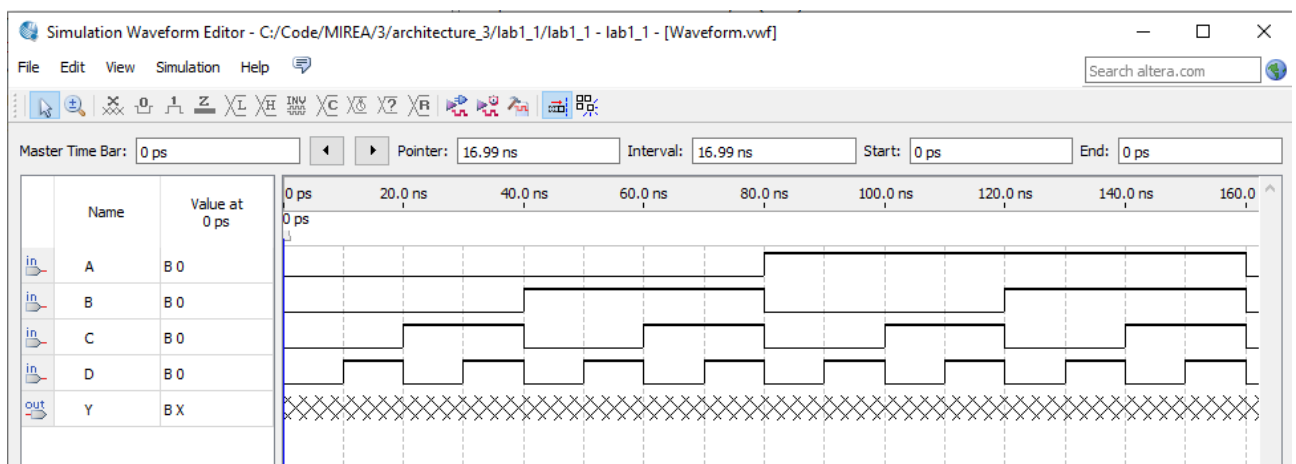


Рисунок 2 – Построение симуляции работы схемы.

Результаты запуска симуляции работы схемы представлены на рисунке 3.

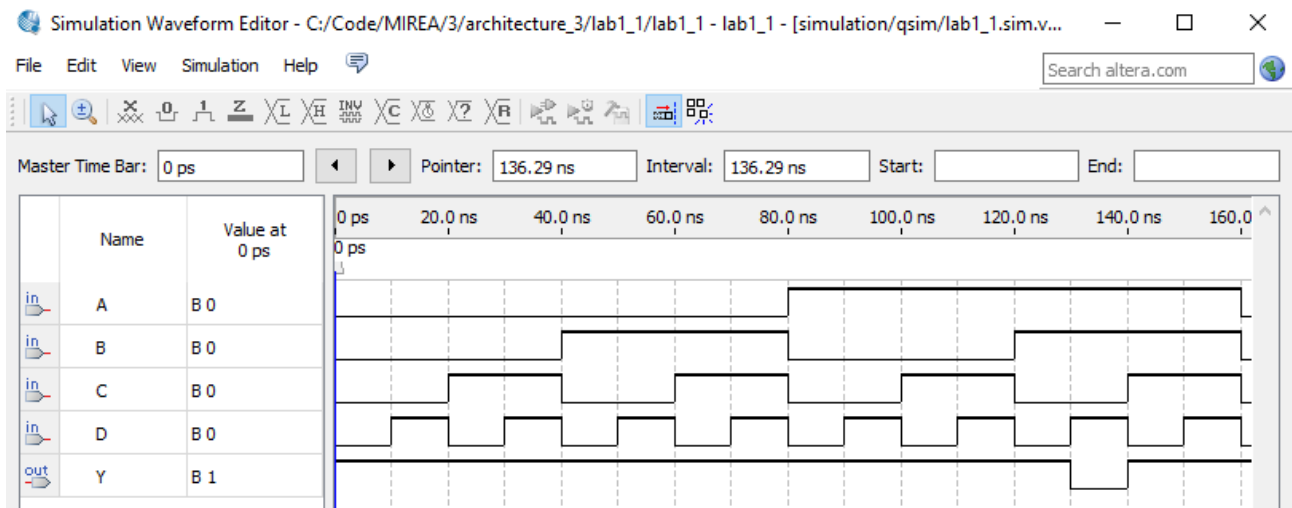


Рисунок 3 – Запуск симуляции работы схемы.

Как видно из результатов запуска симуляции, схема действительно реализует функцию Y (результаты совпадают с данными, полученными при построении таблицы истинности).

Заключение

В ходе выполнения работы была проанализирована и реализована в виде логической схемы логическая функция. Результаты тестирования показали корректность работы логической схемы.

Лабораторная работа №1. Часть 2

Введение

1. Цель работы

Смоделировать логическую схему при помощи текстового редактора САПР QUARTUS II.

2. Основные теоретические сведения

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 (Textdesignfile). 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.

Ход работы

В ходе выполнения лабораторной работы была реализована логическая функция, определённая персональным вариантом №20:

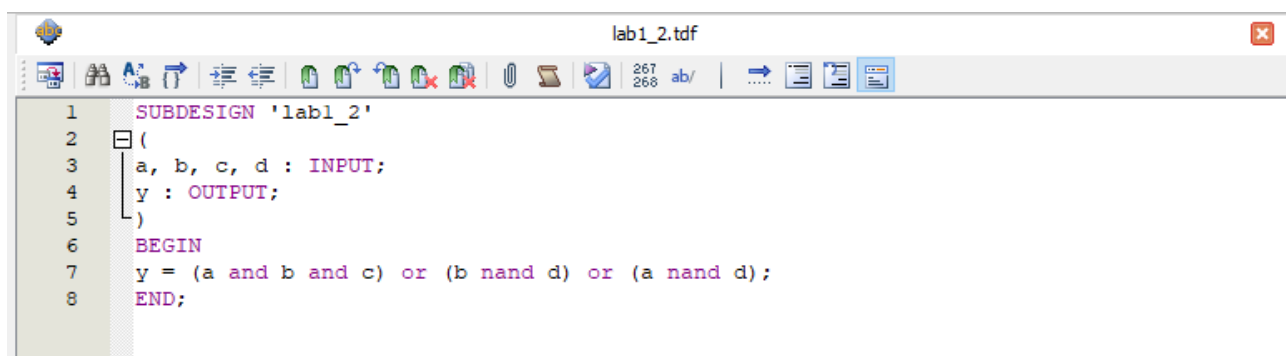
$$Y = ABC + \overline{BD} + \overline{AD}.$$

Таблица истинности для этой функции представлена в таблице 1.

Таблица 1 – Таблица истинности функции Y.

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

Код на языке AHDL, реализующий функцию Y, представлен на рисунке 4.



```

1  SUBDESIGN 'lab1_2'
2  (
3    a, b, c, d : INPUT;
4    y : OUTPUT;
5  )
6  BEGIN
7    y = (a and b and c) or (b nand d) or (a nand d);
8  END;
  
```

Рисунок 4 – Код на языке AHDL, реализующий функцию Y.

Построение симуляции работы схемы представлено на рисунке 5.

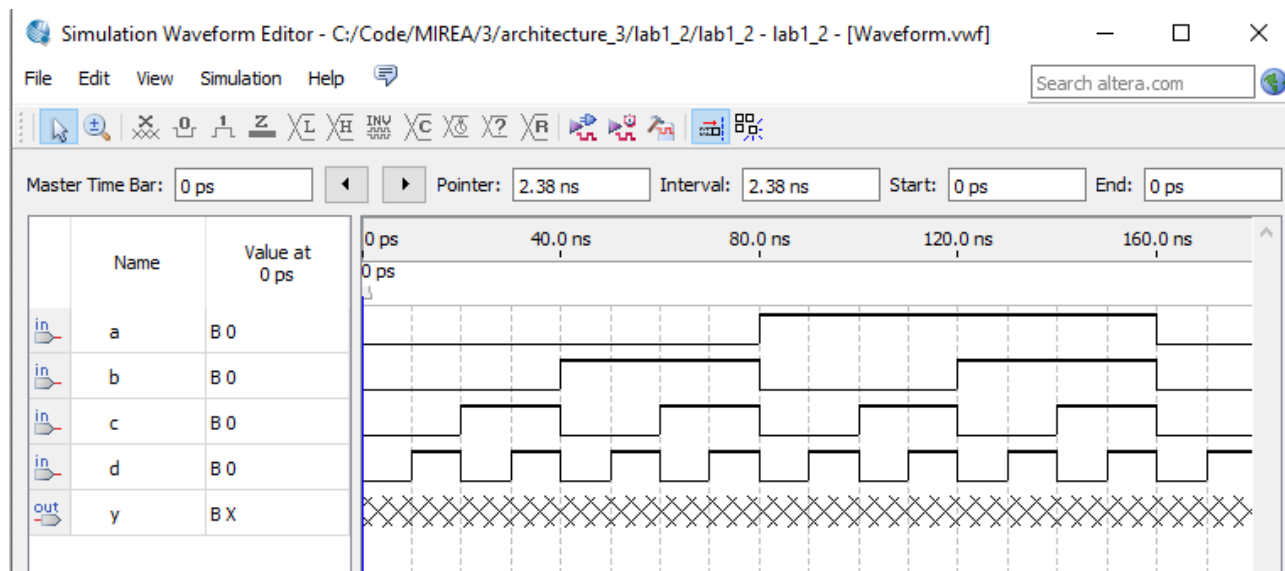


Рисунок 5 – Построение симуляции работы схемы.

Результаты запуска симуляции работы схемы представлены на рисунке 6.

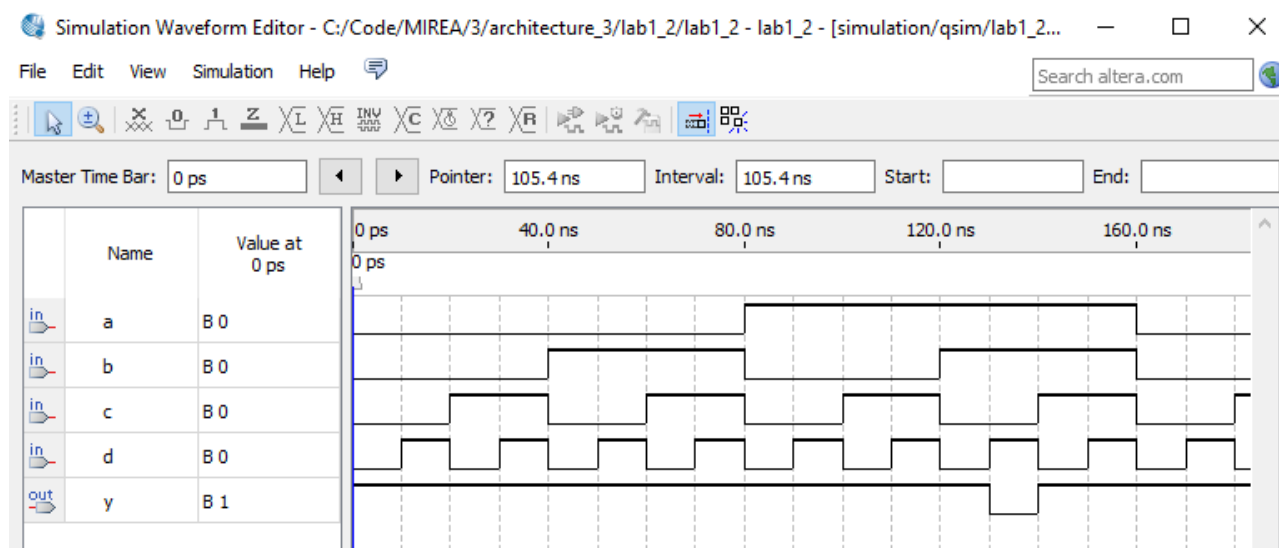


Рисунок 6 – Запуск симуляции работы схемы.

Как видно из результатов запуска симуляции, схема действительно реализует функцию Y (результаты совпадают с данными, полученными при построении таблицы истинности).

Заключение

В ходе выполнения работы была проанализирована и реализована в виде программы на языке AHDL логическая функция. Результаты тестирования показали корректность работы логической схемы, результаты совпали с результатами, полученными в ходе выполнения работы №1.1.

Лабораторная работа №2. Часть 1

Введение

1. Цель работы

Составить более сложную цифровую схему из элементарных логических элементов.

Ход работы

В ходе выполнения лабораторной работы была реализована логическая схема, определённая персональным вариантом №5: демультиплексор 1 в 4.

Логическая схема представлена на рисунке 7.

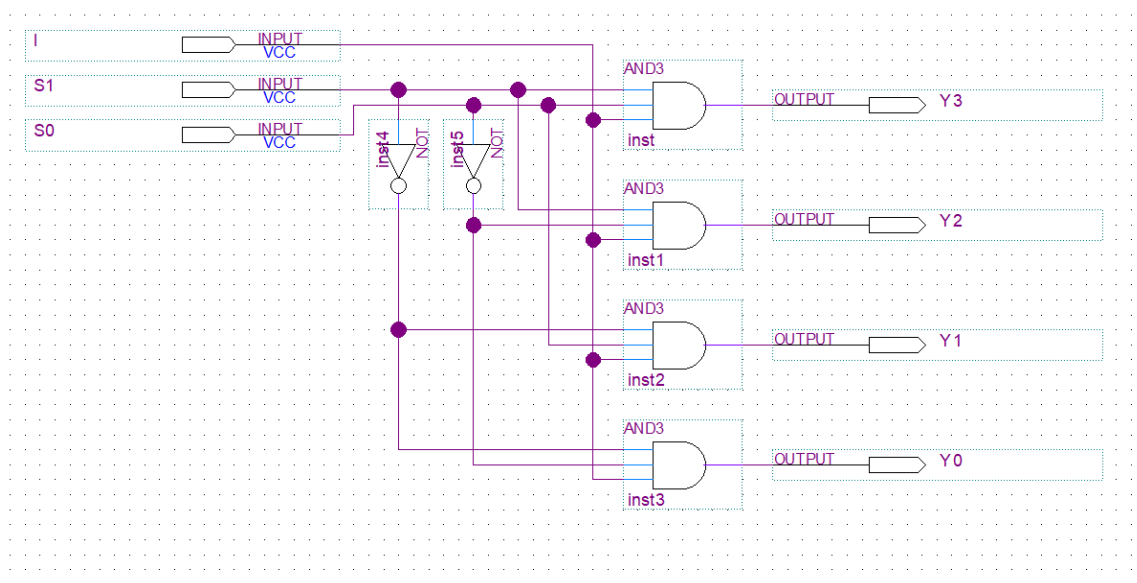


Рисунок 7 – Логическая схема.

Построение симуляции работы схемы представлено на рисунке 8.

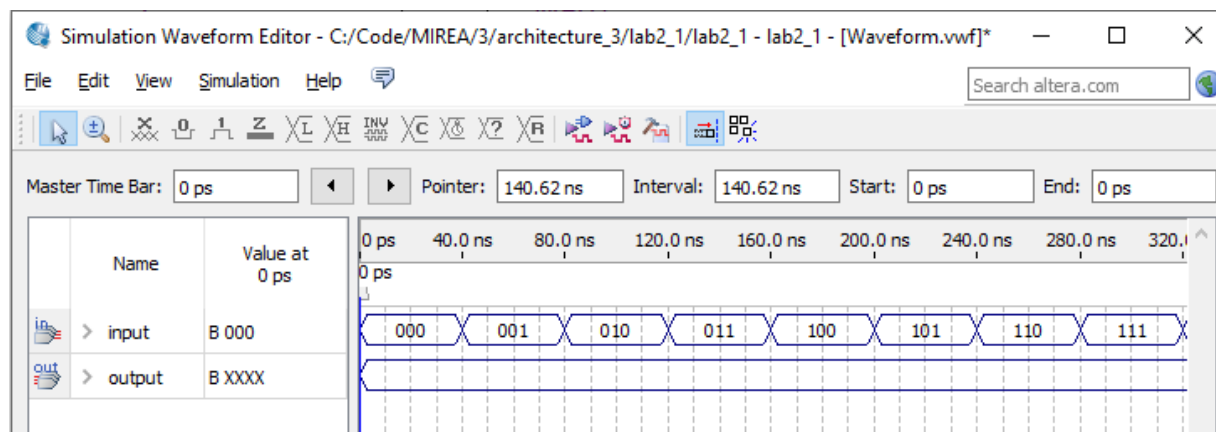


Рисунок 8 – Построение симуляции работы схемы.

Результаты запуска симуляции работы схемы представлены на рисунке 9.

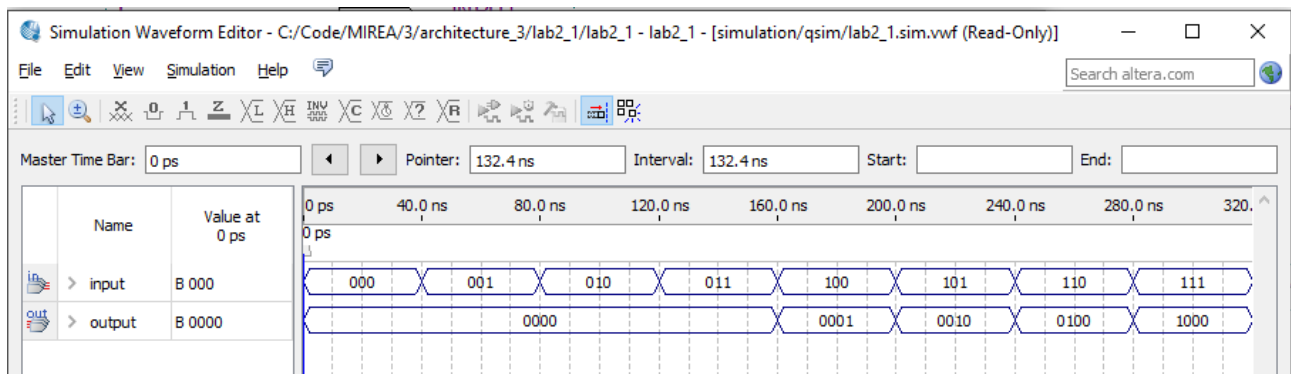


Рисунок 9 – Запуск симуляции работы схемы.

Как видно из результатов запуска симуляции, схема действительно реализует демультиплексор 1 в 4.

Заключение

В ходе выполнения работы было проанализировано устройство демультиплексора 1 в 4, после чего он был реализован в виде логической схемы. Результаты тестирования показали корректность работы логической схемы.

Лабораторная работа №2. Часть 2

Введение

1. Цель работы

Приобретение основных навыков описания цифровых схем с помощью языка описания аппаратуры AHDL.

Ход работы

В ходе выполнения лабораторной работы была реализована логическая схема, определённая персональным вариантом №5: демультиплексор 1 в 4.

Код на языке AHDL, реализующий схему, представлен на рисунке 10.

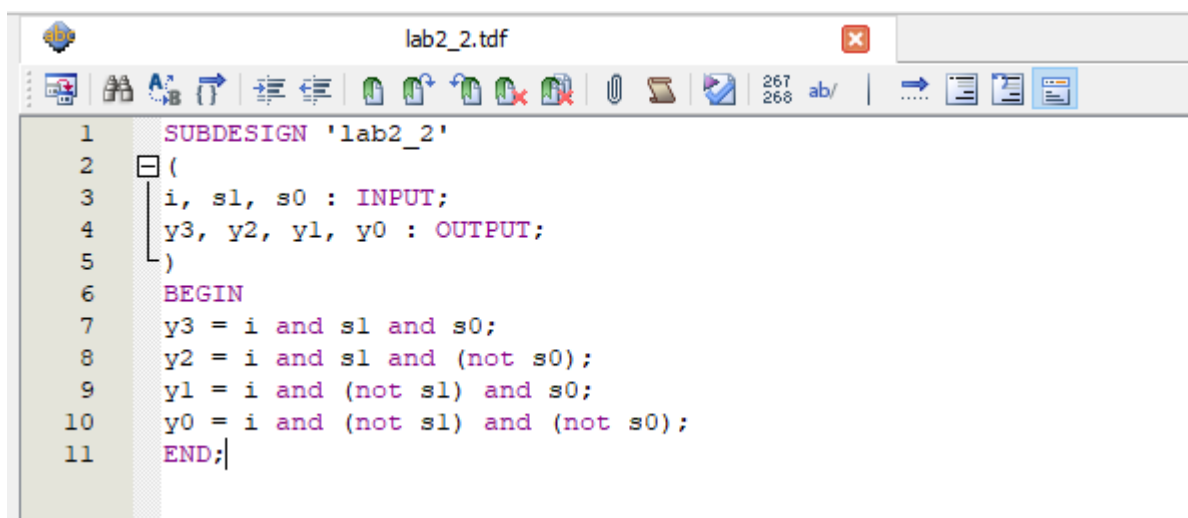


Рисунок 10 – Код на языке AHDL, реализующий схему.

Построение симуляции работы схемы представлено на рисунке 11.

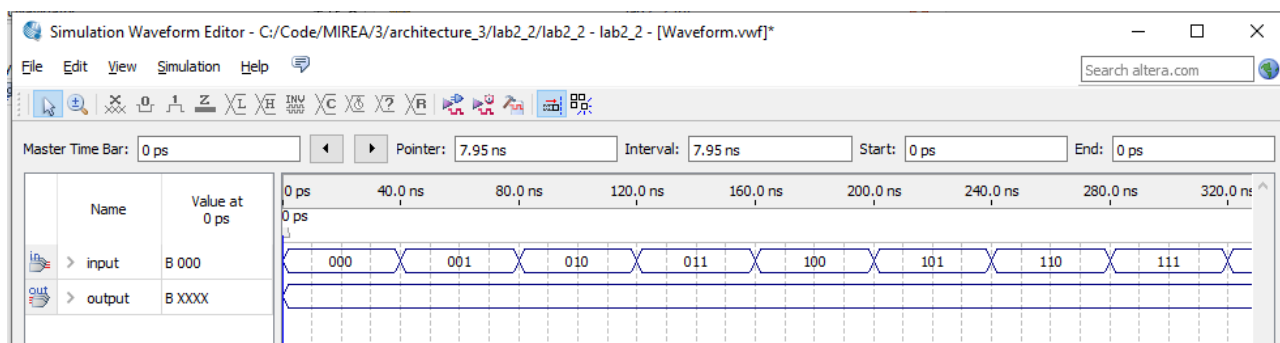


Рисунок 11 – Построение симуляции работы схемы.

Результаты запуска симуляции работы схемы представлены на рисунке 12.

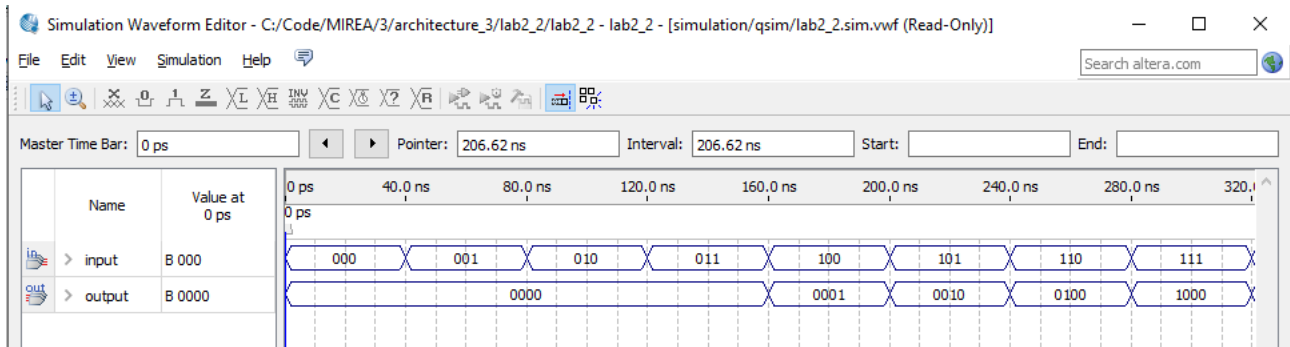


Рисунок 12 – Запуск симуляции работы схемы.

Как видно из результатов запуска симуляции, схема действительно реализует демультиплексор 1 в 4.

Заключение

В ходе выполнения работы было проанализировано устройство демультиплексора 1 в 4, после чего он был реализован в виде программы на языке AHDL. Результаты тестирования показали корректность работы логической схемы.

Лабораторная работа №3

Введение

1. Цель работы

Приобретение навыков использования параметрических элементов (LPM function) в САПР QUARTUS II, экспериментальное исследование счетчиков и регистров, построенных на их основе.

2. Основные теоретические сведения

Регистры и счетчики относятся к разряду цифровых устройств и являются одним из наиболее распространенных элементов вычислительной техники. Они широко используются для построения устройств ввода, вывода и хранения информации, а также для выполнения некоторых арифметических и логических операций.

Для построения счетчиков и регистров используются синхронные триггеры, переключение которых происходит только при наличии синхронизирующего сигнала (синхроимпульса) на входе С. Наиболее часто для построения регистров и счетчиков используется D-триггер, имеющий специальный информационный вход D, и динамический вход С.

Устройство, называемое счетчиком, предназначено для подсчета числа поступающих на вход сигналов (импульсов) в произвольной системе счисления. Двоичные счетчики строятся на основе триггеров, работающих в счетном режиме (Т - триггер или счетный триггер). Счетный триггер может быть получен из универсального D - триггера путем соединения его инверсного выхода Q со входом D.

Ход работы

В ходе выполнения лабораторной работы была реализована логическая схема, определенная персональным вариантом №2: 4-разрядный сумматор.

Логическая схема представлена на рисунке 13.

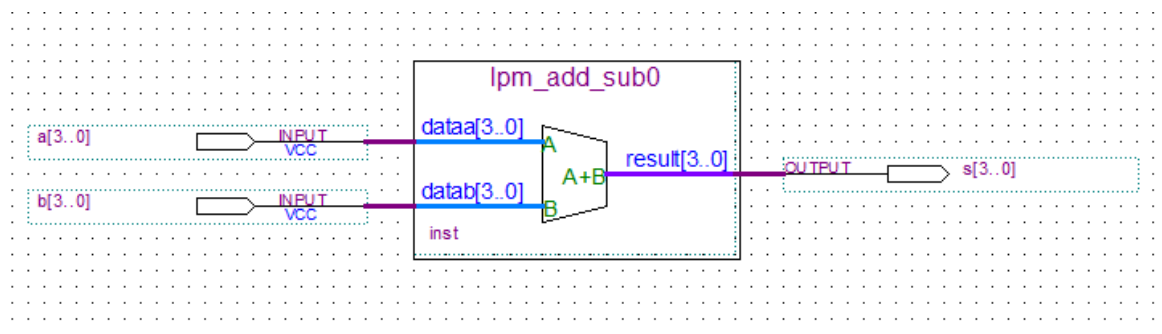


Рисунок 13 – Логическая схема.

Построение симуляции работы схемы представлено на рисунке 14.

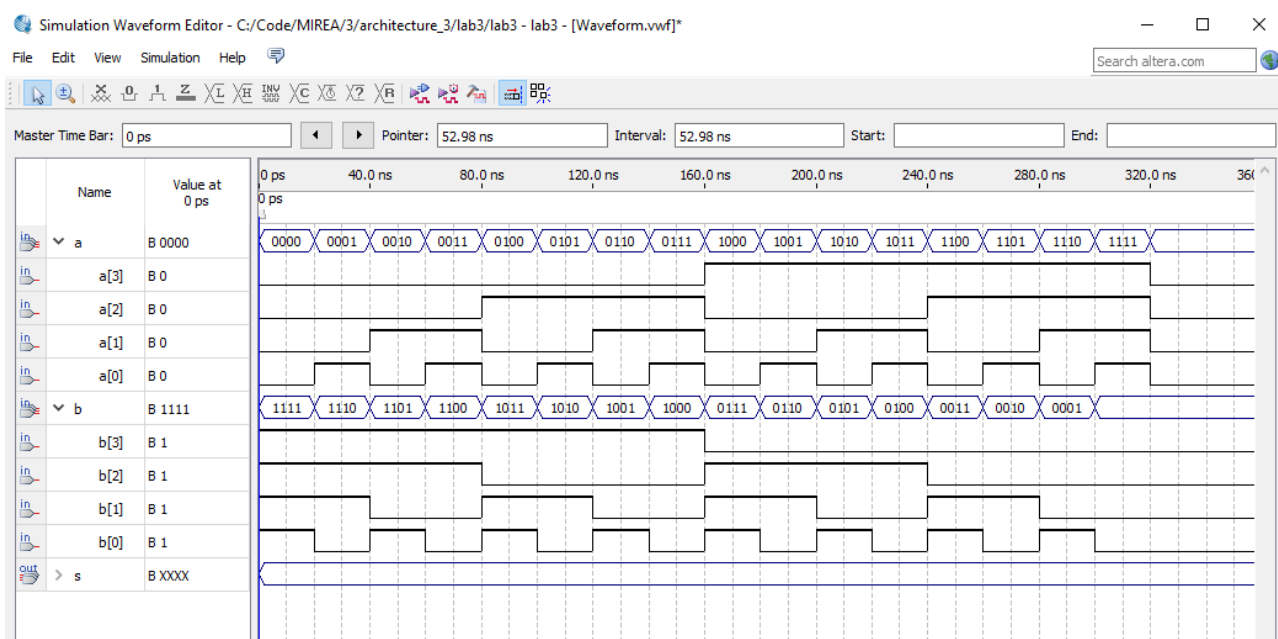


Рисунок 14 – Построение симуляции работы схемы.

Результаты запуска симуляции работы схемы представлены на рисунке 15.

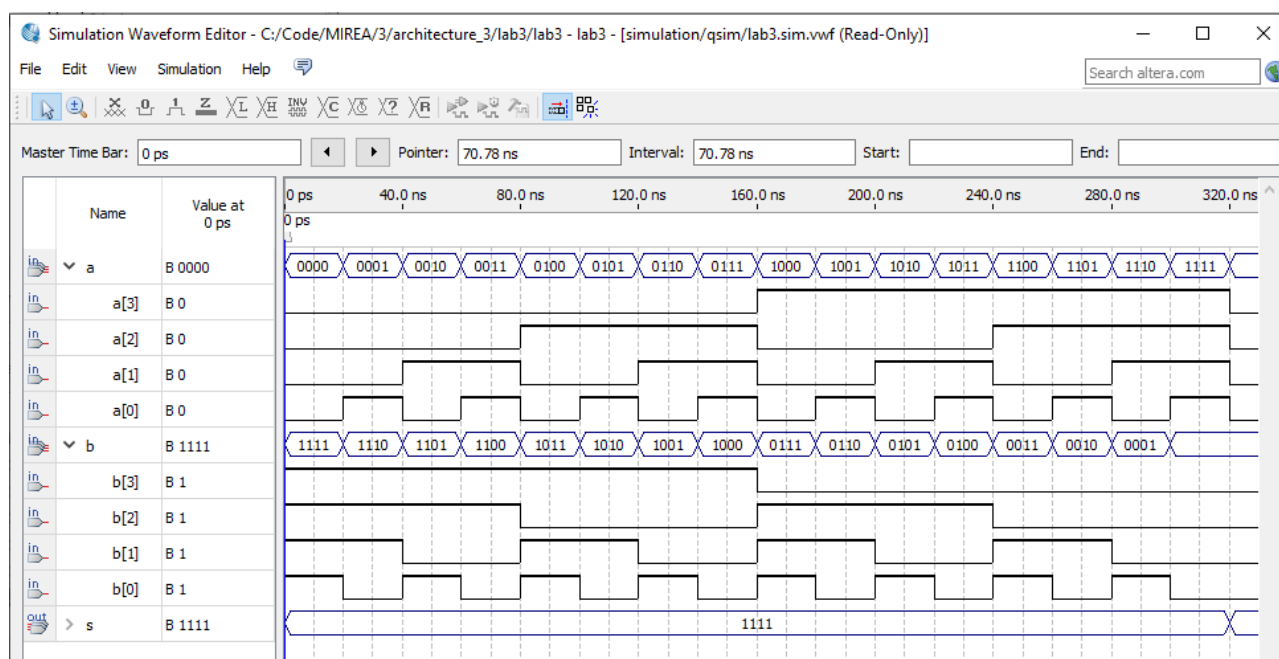


Рисунок 15 – Запуск симуляции работы схемы.

Как видно из результатов запуска симуляции, схема действительно реализует 4-разрядный сумматор.

Заключение

В ходе выполнения работы была реализована логическая схема, выполняющая функцию 4-разрядного сумматора. Тестирование показало корректность работы схемы.