САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ПЕТРА ВЕЛИКОГО

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Институт компьютерных наук и технологий

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

Отчет

по заданию lab7

Дисциплина

«Языки описания аппаратных средств вычислительных систем»

выполнил:

Фам Ба Нам

группа:

3530901/90201

преподаватель:

Федотов А. А.

Санкт-Петербург

2021

Оглавление

[1 Задание lab7\_1 4](#_Toc87451920)

[1.1 Задание 4](#_Toc87451921)

[1.2 Описание на языке Verilog 4](#_Toc87451922)

[1.3 Результат синтеза (RTL) 4](#_Toc87451923)

[1.4 Моделирование 5](#_Toc87451924)

[1.5 Назначение выводов СБИС 5](#_Toc87451925)

[1.6 Тестирование на плате miniDiLaB-CIV 6](#_Toc87451926)

[1.7 Выводы 6](#_Toc87451927)

[2 Задание lab7\_2 6](#_Toc87451928)

[2.1 Задание 6](#_Toc87451929)

[2.2 Описание на языке Verilog 7](#_Toc87451930)

[2.3 Результат синтеза (RTL) 8](#_Toc87451931)

[2.4 Моделирование 8](#_Toc87451932)

[2.5 Назначение выводов СБИС 9](#_Toc87451933)

[2.6 Тестирование на плате miniDiLaB-CIV 9](#_Toc87451934)

[2.7 Выводы 9](#_Toc87451935)

[3 Задание lab7\_3 10](#_Toc87451936)

[3.1 Задание 10](#_Toc87451937)

[3.2 Описание на языке Verilog 10](#_Toc87451938)

[3.3 Результаты синтеза (RTL) 11](#_Toc87451939)

[3.4 Моделирование 12](#_Toc87451940)

[3.5 Назначение выводов СБИС 13](#_Toc87451941)

[3.6 Тестирование на плате miniDiLaB-CIV 13](#_Toc87451942)

[3.7 Выводы 13](#_Toc87451943)

Список иллюстраций

[Рис. 1‑1. Описание на языке Verilog 4](#_Toc87451944)

[Рис. 1‑2. Синтезированная схема 5](#_Toc87451945)

[Рис. 1‑3. Результат моделирования средствами QII 5](#_Toc87451946)

[Рис. 1‑4. Назначение выводов в приложении Pin Planner 6](#_Toc87451947)

[Рис. 2‑1. Описание на языке Verilog 7](#_Toc87451948)

[Рис. 2‑2. Синтезированная схема 8](#_Toc87451949)

[Рис. 2‑3. Результат моделирования средствами QII 9](#_Toc87451950)

[Рис. 2‑4. Назначение выводов в приложении Pin Planner 9](#_Toc87451951)

[Рис. 3‑1. Описание на языке Verilog 11](#_Toc87451952)

[Рис. 3‑2. Синтезированная схема 12](#_Toc87451953)

[Рис. 3‑3. Результат моделирования средствами QII 12](#_Toc87451954)

[Рис. 3‑4. Назначение выводов в приложении Pin Planner 13](#_Toc87451955)

# Задание lab7\_1

## Задание

На языке Verilog создайте описание параметризированного без знакового умножителя двух чисел разрядностью N (параметр).

## Описание на языке Verilog

Описание разрабатываемого устройства на языке Verilog приведено ниже на Рис. 1‑1.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 1‑1. Описание на языке Verilog

## Результат синтеза (RTL)

Результат синтеза описания на языке Verilog в пакете Quartus приведен ниже, на Рис. 1‑2. Изображение схемы получено с помощью приложения RTL Viewer.

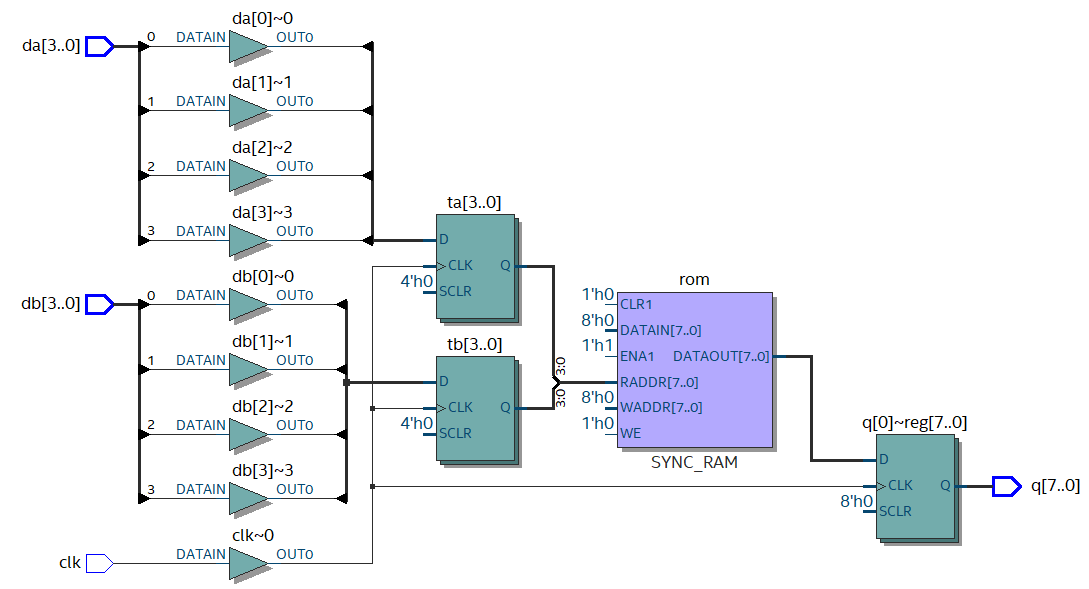


Рис. 1‑2. Синтезированная схема

## Моделирование

Для проверки правильности работы созданного Verilog описания использовались следующие тесты:

1. перемножение входных чисел:
   1. da = 15
   2. db = 15
   3. на следующем такте на выходе q получаем 225

Результаты моделирования приведены на Рис. 1‑3.

Graphical user interface

Description automatically generated with medium confidence

Рис. 1‑3. Результат моделирования средствами QII

## Назначение выводов СБИС

Назначение выводов СБИС и стандартов ввода вывода, выполненное в приложении Pin Planner пакета Quartus, приведено на Рис. 1‑4.

Изображение выглядит как стол

Автоматически созданное описание

Рис. 1‑4. Назначение выводов в приложении Pin Planner

## Тестирование на плате miniDiLaB-CIV

Для тестирования проекта на плате использовались тесты, описанные в разделе 1.4.

Проведенное на плате miniDiLaB-CIV тестирование разработанного устройства показало: результаты совпадают с ожидаемыми, устройство работает в соответствии с заданием.

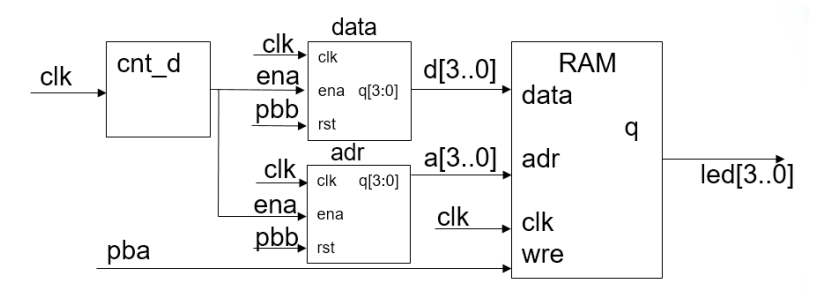
## Выводы

В ходе выполнения данного задания нами впервые была реализована ROM. ROM не использует резервы триггеров платы, а вместо этого работает на битах памяти (которых на плате во много раз больше, чем триггеров).

# Задание lab7\_2

## Задание

На языке Verilog создайте структурное описание устройства, приведенного на рисунке.



## Описание на языке Verilog

Описание разрабатываемого устройства на языке Verilog приведено ниже на Рис. 2‑1.

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Рис. 2‑1. Описание на языке Verilog

## Результат синтеза (RTL)

Результат синтеза описания на языке Verilog в пакете Quartus приведен ниже, на Рис. 2‑2. Изображение схемы получено с помощью приложения RTL Viewer.

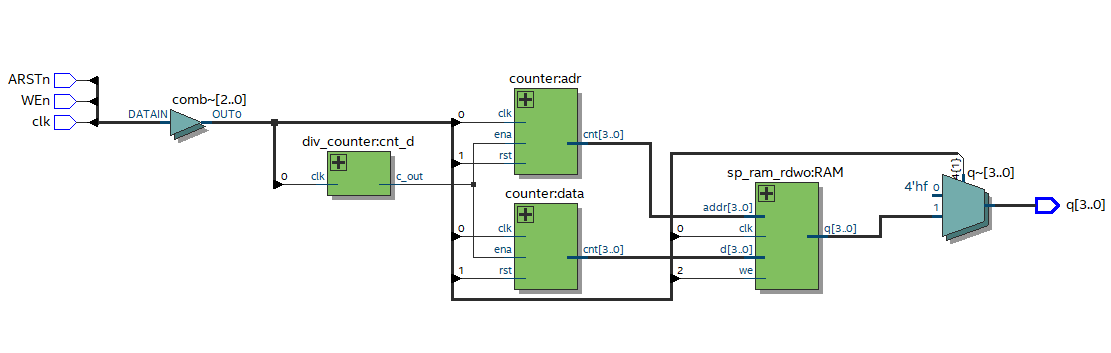


Рис. 2‑2. Синтезированная схема

## Моделирование

Для проверки правильности работы созданного Verilog описания использовались следующие тесты:

1. запись по счётчику:
   1. на вход WEn подаем 0
   2. на вход ARSTn подаем 1
   3. на выходе каждые 4 такта записывается новое число (оно же выводится с задержкой в 1 такт)
2. сброс счётчика:
   1. на вход WEn подаем 0
   2. на вход ARSTn подаем 0
   3. на выходе получаем значения 0-й ячейки памяти (0)
3. чтение по счётчику:
   1. на вход WEn подаем 1
   2. на вход ARSTn подаем 1
   3. каждые 4 такта меняется ячейка для чтения (значение в ячейке подается на вывод)

Результаты моделирования приведены на Рис. 2‑3.

A picture containing graphical user interface

Description automatically generated

Рис. 2‑3. Результат моделирования средствами QII

## Назначение выводов СБИС

Назначение выводов СБИС и стандартов ввода вывода, выполненное в приложении Pin Planner пакета Quartus, приведено на Рис. 2‑4.

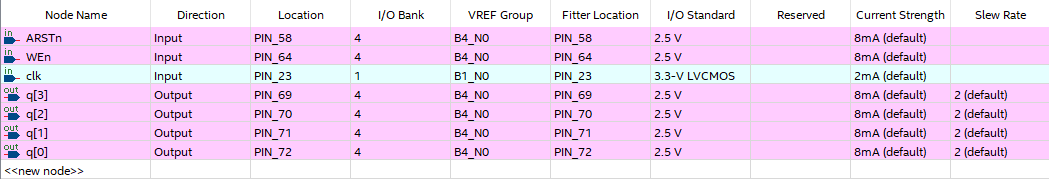


Рис. 2‑4. Назначение выводов в приложении Pin Planner

## Тестирование на плате miniDiLaB-CIV

Для тестирования проекта на плате использовались тесты, описанные в разделе 2.4

Проведенное на плате miniDiLaB-CIV тестирование разработанного устройства показало: результаты совпадают с ожидаемыми, устройство работает в соответствии с заданием.

## Выводы

При описании заданной схемы мы впервые описали single-port memory. В тестах нами было показано, что по такту происходит чтение старого (записанного на предыдущем такте) значения ячейки памяти.

# Задание lab7\_3

## Задание

## Описание на языке Verilog

Описание разрабатываемого устройства на языке Verilog приведено ниже на Рис. 3‑1.

Graphical user interface, text, application

Description automatically generated

Рис. 3‑1. Описание на языке Verilog

## Результаты синтеза (RTL)

Результат синтеза описания на языке Verilog в пакете Quartus приведен ниже, на Рис. 3‑2. Изображение схемы получено с помощью приложения RTL Viewer.

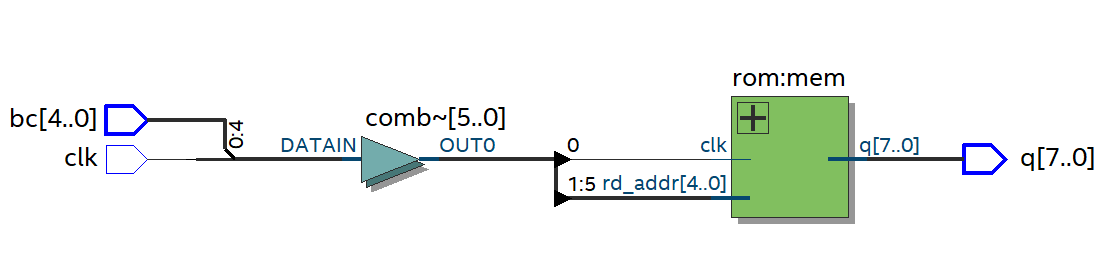


Рис. 3‑2. Синтезированная схема

## Моделирование

Для проверки правильности работы созданного Verilog описания использовались следующие тесты:

1. перевод в 2-10 код:
   1. на вход bc подаем 21
   2. на выходе получаем [0010\_0001]
2. перевод в код грея:
   1. на вход bc подаем 21
   2. на выходе получаем [00011111]
3. вывод константы (1):
   1. на вход bc подаем 1
   2. на выходе получаем [00000001]

Результаты моделирования приведены на Рис. 3‑3.

A picture containing diagram

Description automatically generated

Diagram

Description automatically generated

Calendar

Description automatically generated

Рис. 3‑3. Результат моделирования средствами QII

## Назначение выводов СБИС

Назначение выводов СБИС и стандартов ввода вывода, выполненное в приложении Pin Planner пакета Quartus, приведено на Рис. 3‑4.

Изображение выглядит как стол

Автоматически созданное описание

Рис. 3‑4. Назначение выводов в приложении Pin Planner

## Тестирование на плате miniDiLaB-CIV

Для тестирования проекта на плате использовались тесты, описанные в разделе 2.4

Проведенное на плате miniDiLaB-CIV тестирование разработанного устройства показало: результаты совпадают с ожидаемыми, устройство работает в соответствии с заданием.

## Выводы

В ходе выполнения данного задания мы с помощью ROM создали полноценный транслятор из 2-го кода в 2-10 код/код Грея. Кроме того для сокращения затраченных ресурсов мы использовали конструкцию generate. Она позволила нам изменять описание устройства с помощью параметра (не расходуя ресурсы на неиспользуемые данные).

В ходе данной лабораторной работы мы научились работать с блоками памяти и конструкцией generate языка Quartus. Мы научились как создавать память только для чтения, так и описывать блок памяти с режимами чтения/записи. Умение создавать такие блоки важно, т.к. они используют вместо триггеров биты памяти платы. С учетом того, что битов памяти на плате во много раз больше, чем триггеров, использование блоков памяти значительно оптимизирует использование ресурсов платы.