САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ПЕТРА ВЕЛИКОГО

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

Институт компьютерных наук и технологий

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

Отчет

Пакеты Quartus и ModelSim. Лабораторная работа 4.

Дисциплина

«Технологии проектирования аппаратных средств компьютерных систем»

Выполнил:

Бараев Д.Р.

Группа: 3540901/02001

Преподаватель: А. П. Антонов

Санкт-Петербург

2021

Оглавление

[1 Lab 4\_1 5](#_Toc73958658)

[1.1 Цели 5](#_Toc73958659)

[1.2 Обзор проекта 5](#_Toc73958660)

[1.3 Создание модуля делителя. 6](#_Toc73958661)

[1.4 Моделирование. 7](#_Toc73958662)

[1.4.1 Тест класса 1 7](#_Toc73958663)

[1.4.2 Тест класса 2 8](#_Toc73958664)

[1.5 Создание IP-компонентов 11](#_Toc73958665)

[1.6 Отладка проекта 13](#_Toc73958666)

[1.7 Выводы 15](#_Toc73958667)

[2 Lab 4\_2 15](#_Toc73958668)

[2.1 Цель работы 15](#_Toc73958669)

[2.2 Структура проекта 15](#_Toc73958670)

[2.3 Создать описание модулей сортировки на языке Verilog 16](#_Toc73958671)

[2.4 Моделирование в пакете ModelSim 18](#_Toc73958672)

[2.4.1 Тест класса 1 18](#_Toc73958673)

[2.4.2 Тест класса 2 20](#_Toc73958674)

[2.5 Выводы 20](#_Toc73958675)

1. **Lab4\_1**

[Рисунок 1. 1 - Модуль делителя Lab4\_1.v 5](#_Toc73958692)

[Рисунок 1. 2 - Структура модуля делителя Lab4\_1.v в RTL viewer 5](#_Toc73958693)

[Рисунок 1. 3 - Модуль tb1\_Lab4\_1.v 6](#_Toc73958694)

[Рисунок 1. 4 - Временная диаграмма 7](#_Toc73958695)

[Рисунок 1. 5 - Содержимое файла tb1\_log.txt 7](#_Toc73958696)

[Рисунок 1. 6 - Файл tb2\_Lab4\_1.v 8](#_Toc73958697)

[Рисунок 1. 7 - Содержимое файла rem.dat 9](#_Toc73958698)

[Рисунок 1. 8 - Содержимое файла res.dat 9](#_Toc73958699)

[Рисунок 1. 9 - Содержимое файла D.dat 10](#_Toc73958700)

[Рисунок 1. 10 - Содержимое файла DD.dat 10](#_Toc73958701)

[Рисунок 1. 11 - Модуль PLL\_Unit 11](file:///D:\Политех\Бараев\2%20семестр\Антонов\Lab4%20Q_M_Debug%20Baraev.docx#_Toc73958702)

[Рисунок 1. 12 - Настройка SP\_unit для отладки 11](#_Toc73958703)

[Рисунок 1. 13 - Модуль db\_Lab4\_1.v 12](#_Toc73958704)

[Рисунок 1. 14 - Схема проекта в RTL Viewer 12](#_Toc73958705)

[Рисунок 1. 15 - Отладка (1) 13](#_Toc73958706)

[Рисунок 1. 16 - Отладка (2) 13](#_Toc73958707)

[Рисунок 1. 17 - Отладка (3) 13](#_Toc73958708)

[Рисунок 1. 18 - Отладка (4) 14](#_Toc73958709)

[Рисунок 1. 19 - Отладка (5) 14](#_Toc73958710)

# Lab 4\_1

## Цели

В ходе данной лабораторной работы на языке Verilog опишите беззнаковый делитель с повышенной точностью (4 знака после запятой). Разрядность чисел – параметр N.

## Обзор проекта

**Выводы модуля:**

* Входы
  + CLK – тактовый вход
  + ENA – вход разрешения работы
  + RST – вход асинхронного сброса всех регистров проекта.
  + [N-1:0] D - Делимое
  + [N-1:0] DD - Делитель
* Выходы
  + [N-1:0] RES
  + [3:0] REM

**Требования к модулю.**

На входах и выходах данных должны быть использованы регистры с входами сброса и разрешения работы.

Входы ENA и RST должны быть синхронизированы (использовать два последовательно включенных триггера).

**Моделирование:**

* Значение параметра N = 8
* Разработать тест класса 1 (tb1\_Lab4\_1.v) и провести моделирование. значения входов, выходов и время записывать в файл используя любую команду, данные должны быть отформатированы для удобного чтения. Имя файла – tb1\_lab4\_1.dat
* Разработать тест класса 2 – входные и ожидаемые данные хранятся во внешних файлах (tb2\_Lab4\_1.v) и провести моделирование.

**Отладка**

Значение параметра N = 8

С помощью In System Source and Probe и Signal Tap II провести отладку на плате, зафиксировать результаты,

продемонстрировать преподавателю. Для синхронизации Signal Tap II использовать удвоенную частоту CLK.

**Реализация**

Не требуется.

## Создание модуля делителя.

1. Создание модуля делителя Lab4\_1.v в Quartus Prime.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. 1 - Модуль делителя Lab4\_1.v

Структура делителя в RTL viewer (N=8):

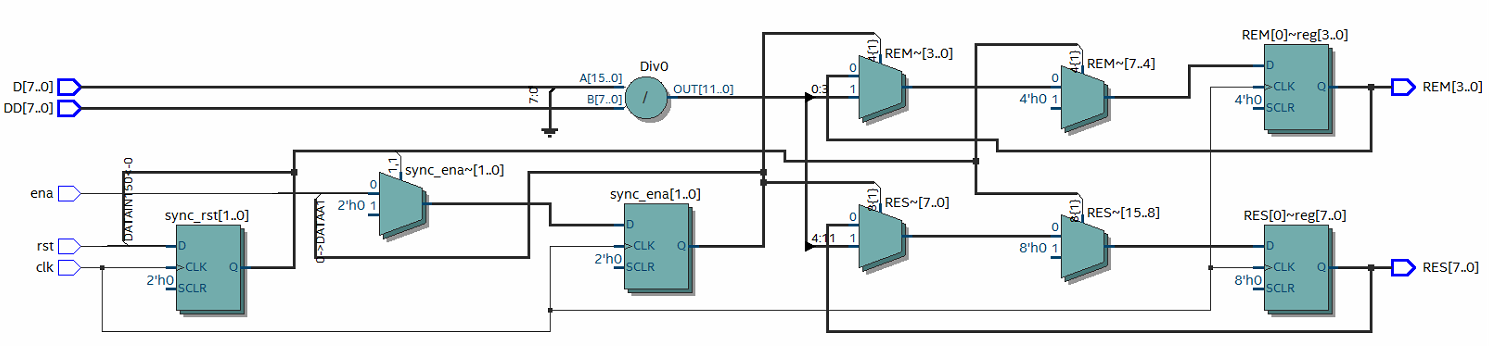


Рисунок 1. 2 - Структура модуля делителя Lab4\_1.v в RTL viewer

## Моделирование.

Задаем параметр N = 8.

### Тест класса 1

Тест класса 1 (модуль tb1\_Lab4\_1.v).

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. 3 - Модуль tb1\_Lab4\_1.v

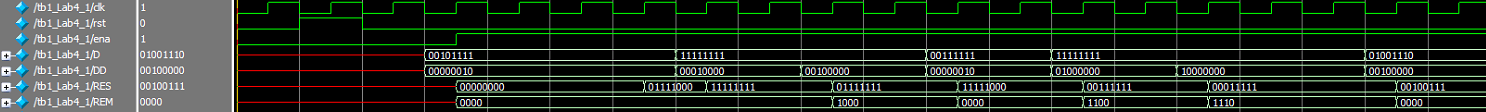


Рисунок 1. 4 - Временная диаграмма

Вывод симулятора (файл tb1\_log.txt):

Изображение выглядит как текст, снимок экрана, внутренний

Автоматически созданное описание

Рисунок 1. 5 - Содержимое файла tb1\_log.txt

### Тест класса 2

1. Проводим разработку теста класса 2.

Тест проверяет правильность данных автоматически, получая входные и итоговые данные из файлов. Результат записывается в двоичной форме в файлы res.dat и rem.dat.



Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. 6 - Файл tb2\_Lab4\_1.v

Результаты, записанные в DAT-файлы:

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 1. 7 - Содержимое файла rem.dat

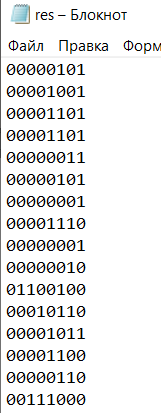


Рисунок 1. 8 - Содержимое файла res.dat

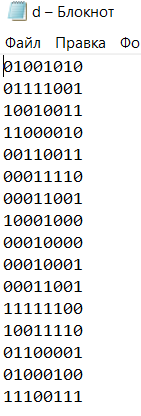


Рисунок 1. 9 - Содержимое файла D.dat

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. 10 - Содержимое файла DD.dat

В случае неправильного выполнения какого-то из 16 тестов – будет происходить и выводиться ошибка. Ход тестирования выводится в файл tb2\_Lab4\_1.dat.

## Создание IP-компонентов

1. Добавили IP-компоненты для отладки (SP\_unit, PLL\_unit).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. 11 - Модуль PLL\_Unit

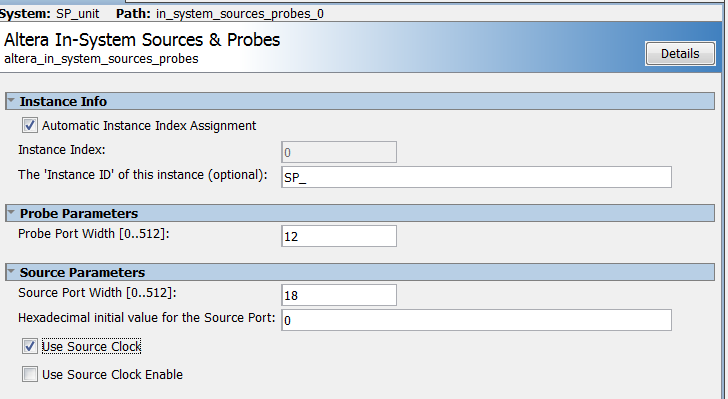


Рисунок 1. 12 - Настройка SP\_unit для отладки

1. Описываем новый файл верхнего уровня db\_Lab4\_1.v и запускаем полную компиляцию.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. 13 - Модуль db\_Lab4\_1.v

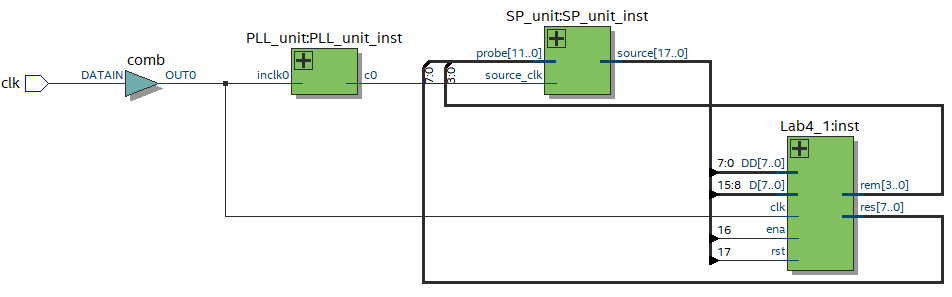


Рисунок 1. 14 - Схема проекта в RTL Viewer

## Отладка проекта

Заходим в окно Programmer, выбираем необходимый интерфейс.

Загружаем проект на плату.

Затем открываем окно In-System Source and Probes Editor и применяем необходимые настройки перед стартом.

1. Подключили плату к USB-порту ПК и включили её.

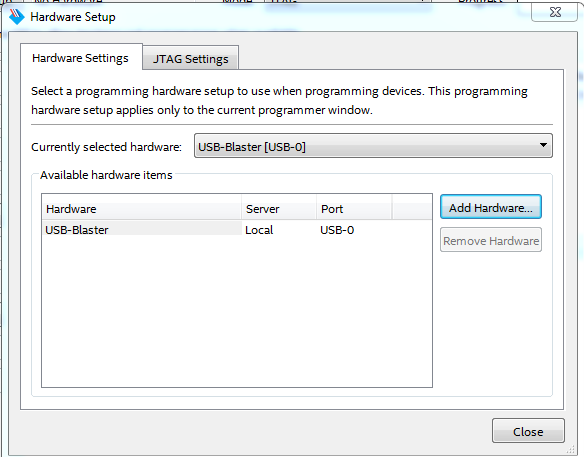


Рисунок 1.17 - Доступное оборудование - USB-Blaster

1. При помощи отладчика проверяем разные режимы работы устройства.

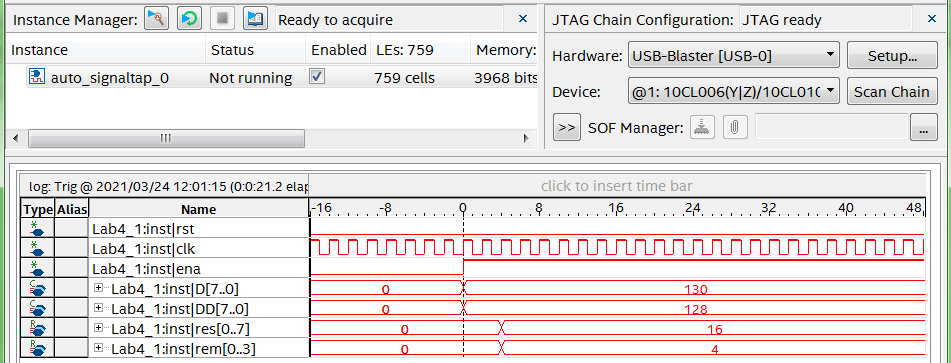


Рисунок 1. 15 - Отладка (1)

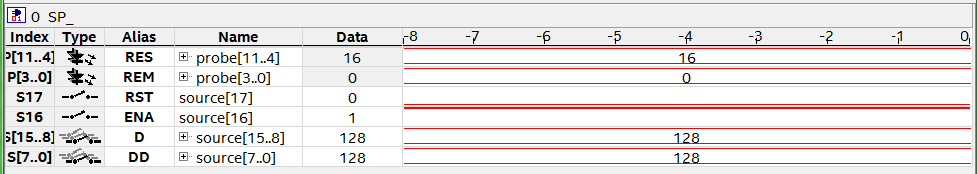


Рисунок 1. 16 - Отладка (2)

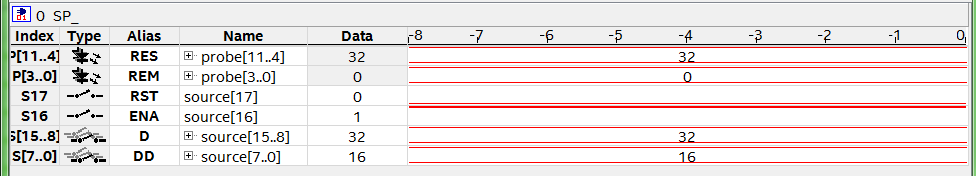


Рисунок 1. 17 - Отладка (3)

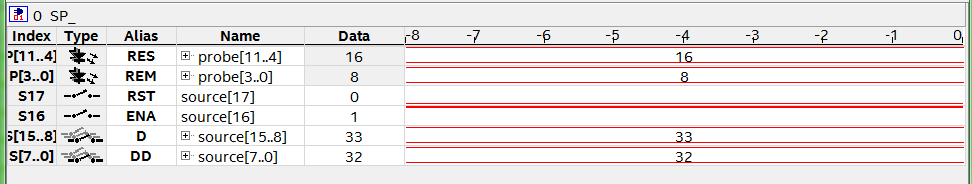


Рисунок 1. 18 - Отладка (4)

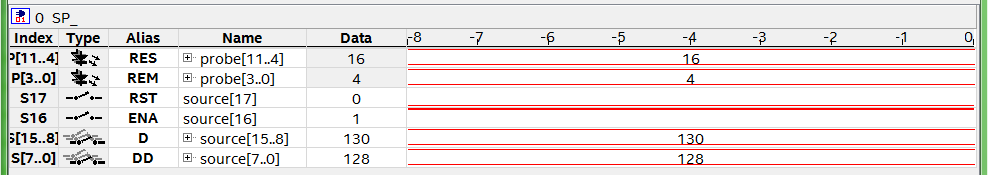


Рисунок 1. 19 - Отладка (5)

## Выводы

В ходе данной лабораторной работы научились:

1. Создавать делители с фиксированной точностью
2. Проводить тесты типа 1 и типа 2 для разработанного устройства
3. Отлаживать проект на плате с помощью ISSP и SignalTap II.

Устройство вместе с отладчиком SignalTap загружено на ПЛИС и там проверена его работоспособность.

# Lab 4\_2

## Цель работы

На языке Verilog опишите устройство сортировки 4-х чисел (числа N-разрядные, знаковые). Разрядность чисел N – параметр.

## Структура проекта

MiniDiLaB-CIV: Микросхема - EP4CE6E22C8, Вход тактового сигнала (25МГц) – 23.

**Выводы модуля:**

* Входы
* CLK – тактовый вход
* ENA – вход разрешения работы
* RST – вход асинхронного сброса регистров проекта.
* [N:0] D [3:0] - входные данные
* Выходы
* [N:0] RES [3:0] - входные данные (большее число в слове [N:0] RES [3])

**Требования к модулю.**

На входах и выходах данных должны быть использованы регистры с входами сброса и разрешения работы.

Входы ENA и RST должны быть синхронизированы (использовать два последовательно включенных триггера).

**Моделирование:**

* Значение параметра N = 16
* Разработать тест класса 1 (tb1\_Lab4\_2.v) и провести моделирование. Значения входов, выходов и время записывать в файл используя любую команду, данные должны быть отформатированы для удобного чтения. имя файла – tb1\_lab4\_2.dat
* Разработать тест класса 2 – входные и ожидаемые данные хранятся во внешних файлах (tb2\_Lab4\_2.v) и провести моделирование.

**Отладка.**

Значение параметра N = 16

С помощью In System Source and Probe и Signal Tap II провести отладку на плате, зафиксировать результаты, продемонстрировать преподавателю. Для синхронизации Signal Tap II использовать удвоенную частоту CLK.

**Реализация.**

Не требуется.

## Создать описание модулей сортировки на языке Verilog

1. Создание модуля Lab4\_2.v.

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2. 1 - Программная реализация модуля Lab4\_2.v

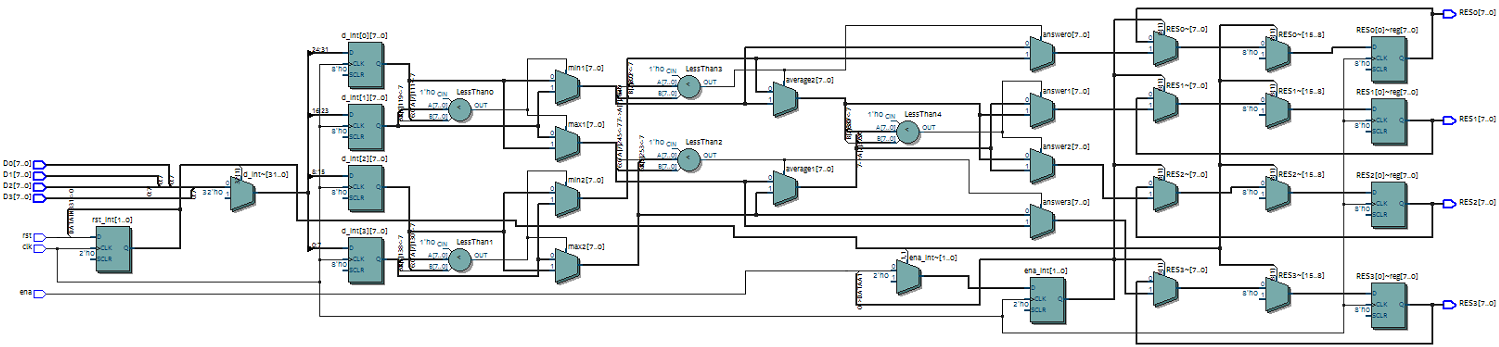


Рисунок 2. 2 - Структура модуля Lab4\_2.v в RTL Viewer

## Моделирование в пакете ModelSim

### Тест класса 1

1. Создать файл tb1\_Lab4\_2.v с тестом класса 1.

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2. 3 - Программная реализация тестового модуля tb1\_Lab4\_2.v

1. Для запуска теста создаём проект в ModelSim и скомпилируем в нём все исходные файлы.

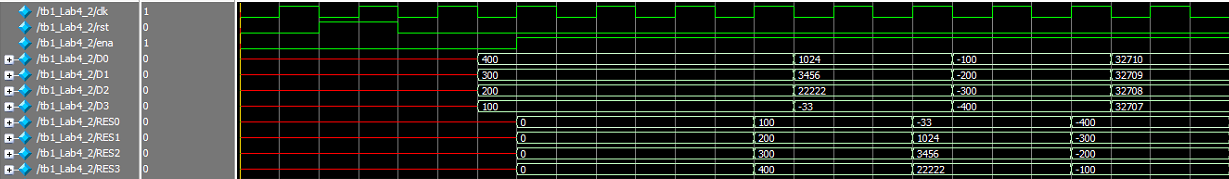


Рисунок 2. 4 - Временная диаграмма

Результаты заносятся в файл при тестировании tb1\_Lab4\_2.dat:

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2. 5 - Содержимое файла tb1\_Lab4\_2.dat

### Тест класса 2

## Выводы

В ходе данной самостоятельной лабораторной работы создали проект в пакете Quartus, успешно осуществили моделирование в пакете ModelSim, осуществили тестирование и отладку на плате и осуществили реализацию на плате.