Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и кибербезопасности

Высшая школа компьютерных технологий и информационных систем

**Отчёт по лабораторной работе № 2\_3**

Дисциплина: Автоматизация проектирования дискретных  
устройств (на английском языке).

Выполнил студент гр. 5130901/10101 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.Л. Симоновский (подпись)

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А. Федотов (подпись)

“07” февраля 2024 г.

Санкт-Петербург

2024

Оглавление

[1. Список иллюстраций: 2](#_Toc158125266)

[2. Задача: 3](#_Toc158125267)

[3. Решение: 3](#_Toc158125268)

[4. Вывод: 5](#_Toc158125269)

# Список иллюстраций:

[Рис. 2.1. Схема разрабатываемого устройства. 3](#_Toc158125271)

[Рис. 3.1. Код модуля верхнего уровня lab2\_3. 4](#_Toc158125272)

[Рис. 3.2. RTL Viewer модуля lab2\_3. 4](#_Toc158125273)

[Рис. 3.3. Код теста первого класса для модуля lab2\_3. 5](#_Toc158125274)

[Рис. 3.5. Результат моделирования. 5](#_Toc158125275)

# Задача:

На языке Verilog разработать устройство по следующей схеме:

Изображение выглядит как текст, диаграмма, План, Шрифт

Автоматически созданное описание

Рис. 2.1. Схема разрабатываемого устройства.

**Выводы:**

1. **Входы:**
   1. CLK – тактовый сигнал.
   2. aRSTin – вход асинхронного сброса (активный уровень для сброса – 1).
   3. [7:0] Din – вход данных для управления ШИМ.
2. **Выходы:**
   1. PWM – выход ШИМ.

**Модули:**

1. **CNT** – счетчик, создаваемый с помощью IP модуля LPM\_COUNTER:
   1. Разрядность: 8 бит.
   2. Двоичный счетчик на сложение.
   3. Выход переноса (carry\_out).
   4. Вход асинхронного сброса (clear).
2. **CMP** – модуль сравнения, создаваемый с помощью IP модуля LPM\_COMPARE:
   1. Два входа по 8 бит.
   2. a < b
   3. Без знаковый.
   4. Без конвейеризации.
3. **RG** – регистр, описываемый на Verilog в файле верхнего уровня:
   1. arst – вход асинхронного сброса (активный уровень – 1) устанавливает 8`d128.
4. **DFF** – триггеры, описываемые на Verilog в файле верхнего уровня:
   1. aset – вход асинхронно устанавливает триггер в 1.

# Решение:

Создадим модуль верхнего уровня на языке Verilog. Его код будет выглядеть следующим образом:

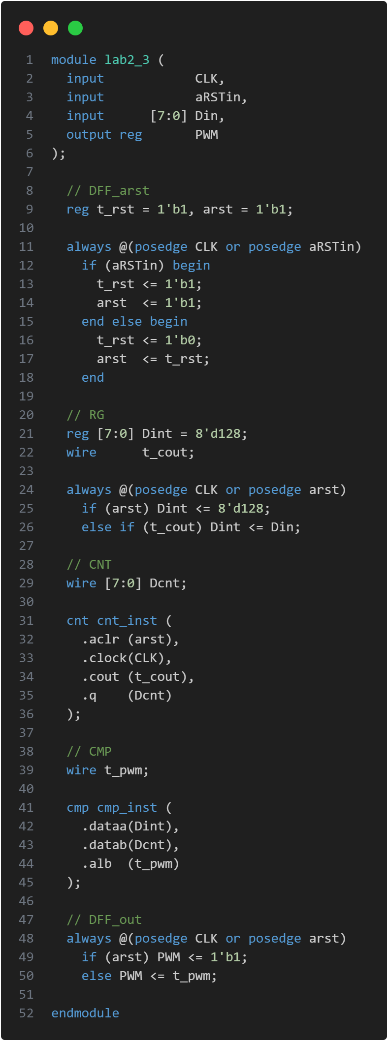


Рис. 3.1. Код модуля верхнего уровня lab2\_3.

Первый блок always описывает два триггера DFF для сигнала асинхронного сброса. Поскольку сброс асинхронный, always реагирует не только на фронт clk, но и на фронт сигнала сброса.

Блок RG реализует регистр на выходе сравнителя. Поскольку регистр должен иметь асинхронный сигнал сброса, поэтому always реагирует не только на фронт clk, но и на фронт сигнала сброса.

Блок CNT передает параметры IP-модулю счетчика, полученного из IP-модулей Quartus Prime.

Блок CMP, аналогично CNT, передает параметры IP-модулю сравнителя.

Блок DFF\_out – регистр на выходе сравнителя, реагирует как на clk, так и на сигнал асинхронного сброса, устанавливая значение в 1.

Проверим корректность разработанной схемы, используя RTL Viewer:

Изображение выглядит как диаграмма, снимок экрана, линия, План

Автоматически созданное описание

Рис. 3.2. RTL Viewer модуля lab2\_3.

Интересным тут является разделение блока регистров Dint на два. Один блок для регистров с 6 по 0, а второй отдельно для седьмого. Если посмотреть на то, что записывается в Dint, видно, что вход D инвертирован. Это свзяно с тем, что при сигнале сброса в Dint должно записываться значение 128, именно поэтому регистр хранит инвертированное значение, чтоб при сбросе автоматически устанавливалось значение 128. В остальном схема соответствует ожиданиям.

Далее разработаем тесты первого класса для разработанного модуля:

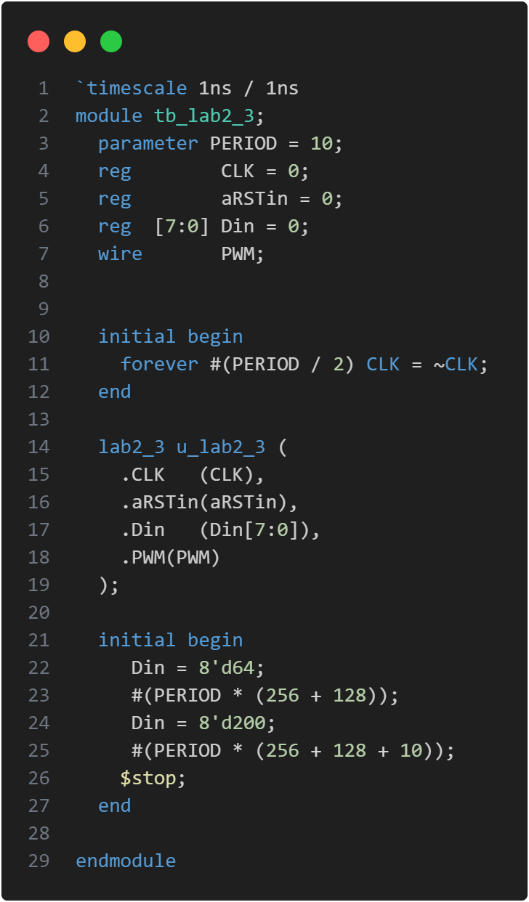


Рис. 3.3. Код теста первого класса для модуля lab2\_3.

В начале устанавливаем значение 64, однако в регистр модуля оно запишется только когда счетчик досчитает до максимума и даст сигнал carry\_out. Поэтому подождем 256 тактов. После этого ждем еще 128 тактов и меняем значение на 200, однако запись произойдет еще через 128 тактов. Смотрим на результат 256 + 10 тактов, чтоб увидеть, как выход сбросится опять до 0.

Запустим тестовый файл средствами Quartus Prime.

Результат запуска выглядит следующим образом:

Изображение выглядит как снимок экрана, Мультимедийное программное обеспечение, программное обеспечение, Графическое программное обеспечение

Автоматически созданное описание

Рис. 3.5. Результат моделирования.

Как видно, получившаяся waveform полностью соответствует ожиданиям и техническому заданию, что свидетельствует о корректно разработанном устройстве.

# Вывод:

В ходе лабораторной работы было разработано устройство на языке Verilog в соответствии с схемой. Получившееся устройство полностью соответствует техническому заданию. Разработанное устройство – обычный ШИМ (широтно-импульсный модулятор), что видно по выходу PWM. Оно принимает на вход Din величину ШИМ, которая записывается по сигналу t\_cout, чтоб не происходило каких-либо помех при смене этого значения в процессе вывода сигнала.

В процессе разработки использовались IP-модули из библиотеки Quartus Prime, стоит отметить, что они сильно ускорили процесс и помогли избавиться от написания стандартных модулей, дав возможность сосредоточиться на основном задании.