

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 № 6**

«Делитель частоты»

по дисциплине

«Архитектура вычислительных машин и систем»

Выполнил студент группы Зырянов М.А.

ИВБО-01-22

Принял ассистент кафедры ВТ Дуксина И.И.

Практическая работа выполнена «\_\_»\_\_\_\_\_\_\_2023 г.

«Зачтено» «\_\_»\_\_\_\_\_\_\_2023 г.

Москва 2023

**АННОТАЦИЯ**

Данная работа включает в себя 3 рисунков и 3 листинга.

Количество страниц в работе — 11.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_Toc151314894)

[1 ПРОЕКТИРОВАНИЕ МОДУЛЕЙ 5](#_Toc151314895)

[1.1 Создание модуля Счётчика 5](#_Toc151314896)

[1.2 Создание модуля Делителя частоты 6](#_Toc151314897)

[1.3 Создание тестового модуля 6](#_Toc151314898)

[2 ВРЕМЕННЫЕ ДИАГРАММЫ 8](#_Toc151314899)

[ЗАКЛЮЧЕНИЕ 9](#_Toc151314900)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 10](#_Toc151314901)

# ВВЕДЕНИЕ

В данной практической работе требуется создать проект САПР Vivado, создать модуль, описывающий параметрический универсальный реверсивный счётчик, экземпляр модуля счётчика, реализующий делитель частоты по выданному варианту и тестовый на языке Verilog HDL. Протестировать работу модулей и убедиться в их правильности. Затем подготовить отчёт о проделанной работе и защитить её.

Персональный вариант: Делитель частоты на 10.

# 1 ПРОЕКТИРОВАНИЕ МОДУЛЕЙ

## 1.1 Создание модуля Счётчика

В данной практической работе нужно реализовать параметрический универсальный реверсивный счётчик. У модуля счётчика будет один вход clk для тактового генератора и регистр cnt на выходе, передающий 2n битов. В качестве параметров модуль принимает значение step — шаг счётчика, и modul — модуль счётчика.

В самом начале своей работы модуль инициализирует регистр cnt значением 0. После каждый раз, когда приходит восходящий фронт clk проверяется условие, что если параметр step отрицателен, а значение cnt равно 0, то регистр принимает максимально возможное значение счётчика. Иначе в регистр запишется значения остатка от деления на modul суммы текущего состояния счётчика и значения параметра step (Листинг 1.1).

*Листинг 1.1 — Модуль параметрический универсальный реверсивный счётчик на языке Verilog HDL*

|  |
| --- |
| `timescale 1ns / 1ps  module counter #(step = 1, modul = 2)(  input clk,  output reg [$clog2(modul) - 1:0] cnt  );  initial  cnt = 0;  always@(posedge clk)  if(step < 0 && cnt == 0)  cnt = modul - 1;  else  cnt <= (cnt + step) % modul;  endmodule |

## 1.2 Создание модуля Делителя частоты

По выданному варианту нужно реализовать делитель частоты на 10, но модуль будет создан параметрический. У модуля будет один вход clk для тактового генератора и 2 выхода: clk\_d — тактовый генератор делителя частоты, cnt — регистр со значениями внутреннего счётчика. В качестве параметров подаётся md — модуль, на который частота должна делиться.

В самом начале работы модуль инициализирует регистр clk\_d значением 0. После этого подключается модуль счётчика, куда в качестве параметров передаётся шаг 1 и модуль md. На вход модуля счётчика подаётся clk, а на выход cnt. После этого каждый раз, как изменяется значение регистра cnt модуль проверяет, равно ли значение регистра cnt единице, и если утверждение истинно, то регистр clk\_d переходит в противоположное положение (Листинг 1.2).

*Листинг 1.2 — Модуль параметризованного делителя частоты на языке Verilog HDL*

|  |
| --- |
| `timescale 1ns / 1ps  module clk\_div #(md = 1)(  input clk,  output reg clk\_d, wire [$clog2(md) - 1:0] cnt  );    initial clk\_d = 0;  counter #(.step(1), .modul(md)) c(  .clk(clk),  .cnt(cnt)  );  always@(cnt)  if(cnt == 1)  clk\_d = ~clk\_d;  endmodule |

## 1.3 Создание тестового модуля

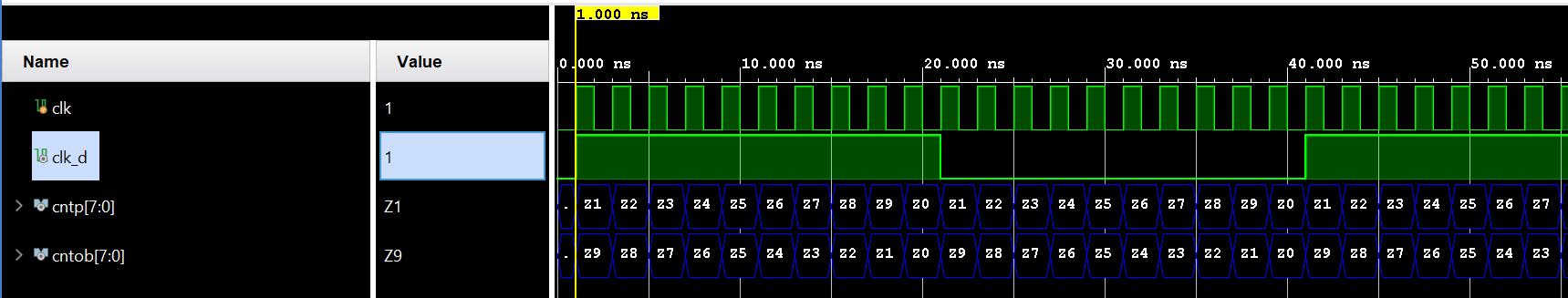
Для запуска и проверки работоспособности модулей нам нужен симулирующий тестовый модуль. В него будет подключены модуль делитель частоты с параметром md = 10, чтобы показать выполнение выданного варианта, а также модуль счётчика с параметрами step = -1 и modul = 10, чтобы показать способность модуля счётчика работать в обоих направлениях. На временной диаграмме будут показаны изменения регистра clk — изначальный тактовый генератор, изменяющийся каждую единицу времени, цепь clk\_d — такты тактового генератора через делитель частоты на 10, цепь cntp — значения, которые принимает счётчик с шагом 1 и цепь cntob — значения, которые принимает счётчик с шагом -1 (Листинг 1.3).

*Листинг 1.3 — Тестовый модуль testbench.v*

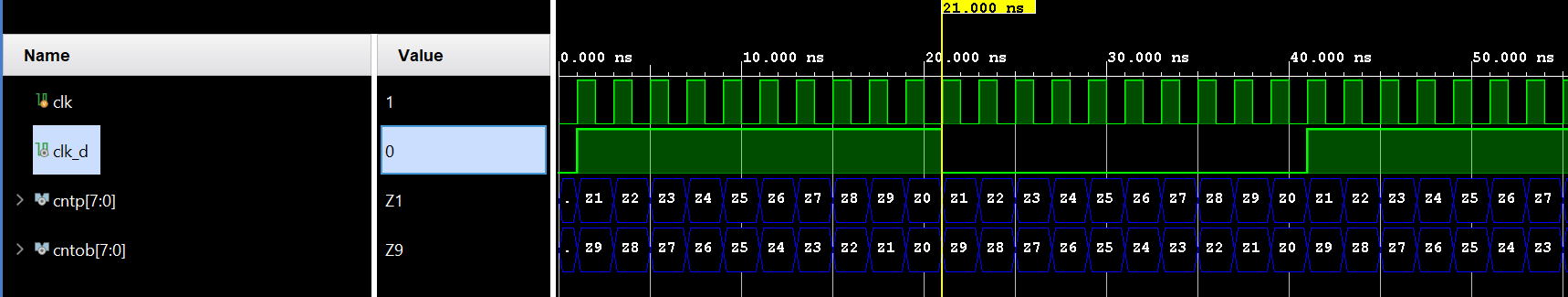
|  |
| --- |
| `timescale 1ns / 1ps  module testbench;  reg clk = 0;  wire clk\_d;  wire [7:0] cntp;  wire [7:0] cntob;  always #1 clk = ~clk;  clk\_div #(.md(10)) div(  .clk(clk),  .clk\_d(clk\_d),  .cnt(cntp)  );  counter #(.step(-1), .modul(10)) d(  .clk(clk),  .cnt(cntob)  );  endmodule |

# 2 ВРЕМЕННЫЕ ДИАГРАММЫ

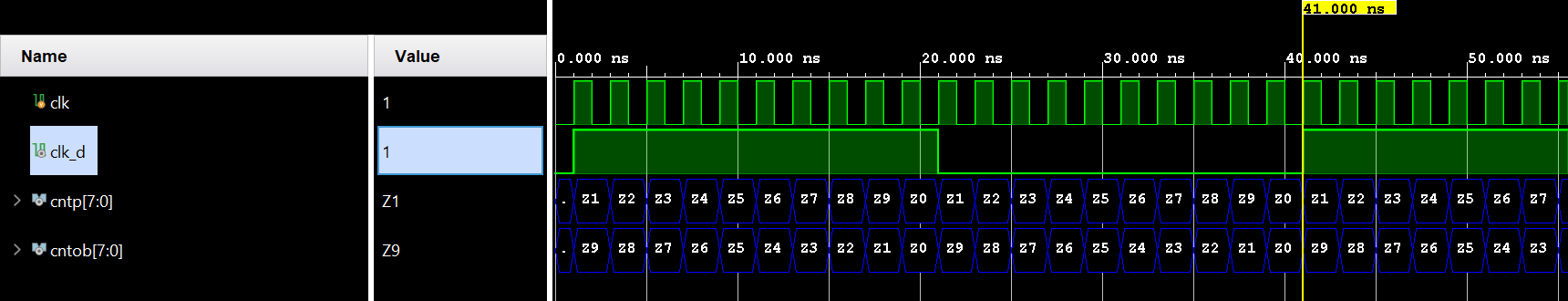
На временной диаграмме будут показаны изменения clk — тактовый генератор, clk\_d — такты тактового генератора через делитель частоты на 10, cntp — значения, которые принимает счётчик с шагом 1 и cntob — значения, которые принимает счётчик с шагом -1.



**Рисунок 2.1 – Временная диаграмма — восходящий фронт первого такта делителя частоты (часть 1)**



**Рисунок 2.2 – Временная диаграмма — убывающий фронт первого такта делителя частоты (часть 2)**



**Рисунок 2.3 – Временная диаграмма — восходящий фронт второго такта делителя частоты (часть 3)**

На временной диаграмме мы можем увидеть, первый так делителя частоты открывается в момент 1ns, а закрывается в момент 21ns. Из выражения можно понять, что такт делителя частоты открыт 10 тактов обычного тактового генератора, что и нужно было сделать. Для того, чтобы окончательно убедиться проверим сколько тактов тактового генератора такт делителя частоты имеет значение 0. Из выражения можно понять, что также 10 тактов.

# ЗАКЛЮЧЕНИЕ

Для заданной варианта в САПР Vivado были созданы модули параметрического универсального реверсивного счётчика, параметрического делителя частоты и тестовый testbench. Осуществлена симуляция работы данных модулей, а также проведена успешная проверка работы модулей.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Методические указания по ПР № 1 — URL: https://onlineedu.mirea.ru/mod/resource/view.php?id=405132 (Дата обращения: 13.09.2023).

2. Методические указания по ПР № 2 — URL: https://onlineedu.mirea.ru/mod/resource/view.php?id=409130 (Дата обращения: 13.09.2023).

3. Смирнов С.С. Информатика [Электронный ресурс]: Методические указания по выполнению практических и лабораторных работ / С.С. Смирнов — М., МИРЭА — Российский технологический университет, 2018. — 1 электрон. опт. диск (CD-ROM).

4. Тарасов И.Е. ПЛИС Xilinx. Языки описания аппаратуры VHDL и Verilog, САПР, приемы проектирования. — М.: Горячая линия — Телеком, 2021. — 538 с.: ил.

5. Антик М.И. Дискретная математика [Электронный ресурс]: Учебное пособие / Антик М.И., Казанцева Л.В. — М.: МИРЭА — Российский технологический университет, 2018 — 1 электрон. опт. диск (CD-ROM).