САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ПЕТРА ВЕЛИКОГО

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

Институт компьютерных наук и технологий

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

Отчет

по заданию lab5

Дисциплина

«Языки описания аппаратных средств вычислительных систем»

выполнил:

Анциферов С.А.

группа:

3530901/90201

преподаватель:

Федотов А. А.

Санкт-Петербург

2021

Оглавление

[1 Задание lab5\_1 4](#_Toc85579027)

[1.1 Задание 4](#_Toc85579028)

[1.2 Описание на языке Verilog 4](#_Toc85579031)

[1.3 Результат синтеза (RTL) 6](#_Toc85579032)

[1.4 Моделирование 7](#_Toc85579033)

[1.5 Тестирование на плате miniDiLaB- CIV 7](#_Toc85579034)

[1.6 Выводы 7](#_Toc85579035)

[2 Задание lab5\_2 8](#_Toc85579036)

[2.1 Задание 8](#_Toc85579037)

[2.2 Описание на языке Verilog 8](#_Toc85579038)

[2.3 Результат синтеза (RTL) 10](#_Toc85579039)

[2.4 Моделирование 10](#_Toc85579041)

[2.5 Тестирование на плате miniDiLaB-CIV 11](#_Toc85579042)

[2.6 Выводы 11](#_Toc85579043)

[3 Задание lab5\_3 12](#_Toc85579044)

[3.1 Задание 12](#_Toc85579045)

[3.2 Описание на языке Verilog 12](#_Toc85579046)

[3.3 Результат синтеза (RTL) 14](#_Toc85579047)

[3.4 Моделирование 14](#_Toc85579049)

[3.5 Тестирование на плате miniDiLaB-CIV 15](#_Toc85579051)

[3.6 Выводы 15](#_Toc85579052)

[4 Выводы 16](#_Toc85579053)

Список иллюстраций

[Рис. 1‑1 Описание на языке Verilog 4](#_Toc83927050)

[Рис. 1‑2 Синтезированная схема 4](#_Toc83927051)

[Рис. 1‑3 Результат моделирования средствами QII 6](#_Toc83927052)

[Рис. 1‑4 Обзор выводов в приложении Pin Planner 6](#_Toc83927053)

[Рис. 2‑1 Описание на языке Verilog 8](#_Toc83927054)

[Рис. 2‑2 Синтезированная схема 9](#_Toc83927055)

[Рис. 2‑3 Результат моделирования средствами QII 10](#_Toc83927056)

[Рис. 2‑4 Обзор выводов в приложении Pin Planner 11](#_Toc83927057)

[Рис. 3‑1 Описание на языке Verilog 12](#_Toc83927058)

[Рис. 3‑2 Синтезированная схема 12](#_Toc83927059)

[Рис. 3‑3 Результат моделирования средствами QII 14](#_Toc83927060)

[Рис.3‑4 Обзор выводов в приложении Pin Planner 15](#_Toc83927061)

# Задание lab5\_1

## Задание

На языке Verilog разработать

параметризированный модуль, который

содержит:

# 2 входных регистра; Умножитель;

# Выходной регистр

* ena – (key[0]) – разрешение работы (при 1)
* arst - (key[1]) – асинхронный сброс (при 0)
* clk - тактовый сигнал
* da[3..0] – sw[7..4],
* db[3..0] – sw[3..0]
* result [7..0] – led [7..0]

Необходимо создать:

* Параметризированный регистр
* Параметризированный умножитель
* Параметризированное описание верхнего уровня

Где:

Входы данных:

Clk – тактовая частота 25МГц

переключатели sw[7:0]

кнопка PB[0]

Выходы:

светодиоды led[3:0]

## Описание на языке Verilog

|  |
| --- |
| module lab5\_1  #(parameter WIDTH = 4)  (  (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "58"\*)  input arst,    (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "64"\*)  input ena,    (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "23"\*)  input clk,    (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "88, 89, 90, 91"\*)  input [WIDTH-1:0] da,    (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "49, 46, 25, 24"\*)  input [WIDTH-1:0] db,    (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "65, 66, 67, 68, 69, 70, 71, 72"\*)  output [2\*WIDTH-1:0] result  );    wire [WIDTH-1:0] da\_int, db\_int;  wire [2\*WIDTH-1:0] mult\_res;    rgstr #WIDTH reg1 (arst, ena, clk, da, da\_int);  rgstr #WIDTH reg2 (arst, ena, clk, db, db\_int);    mult #WIDTH mult\_a\_b (da\_int, db\_int, mult\_res);  rgstr #(2\*WIDTH) out\_reg (arst, ena, clk, mult\_res, result);  endmodule  module mult (  input [width\_mult-1:0] a, b,  output [2\*width\_mult-1:0] result  );  parameter width\_mult = 8;  assign result = a \* b;  endmodule  module rgstr(arst, ena, clk, d, q);  parameter width\_rg = 8;  input arst, ena, clk;  input [width\_rg - 1:0] d;  output reg [width\_rg - 1:0] q;  always @ (posedge clk, negedge arst)  begin  if ( !arst )  q <= {width\_rg{1'b0}};  else if (ena)  q <= d;  end  endmodule |

Рис.1.2 Описание на языке SystemVerilog

## Результат синтеза (RTL)

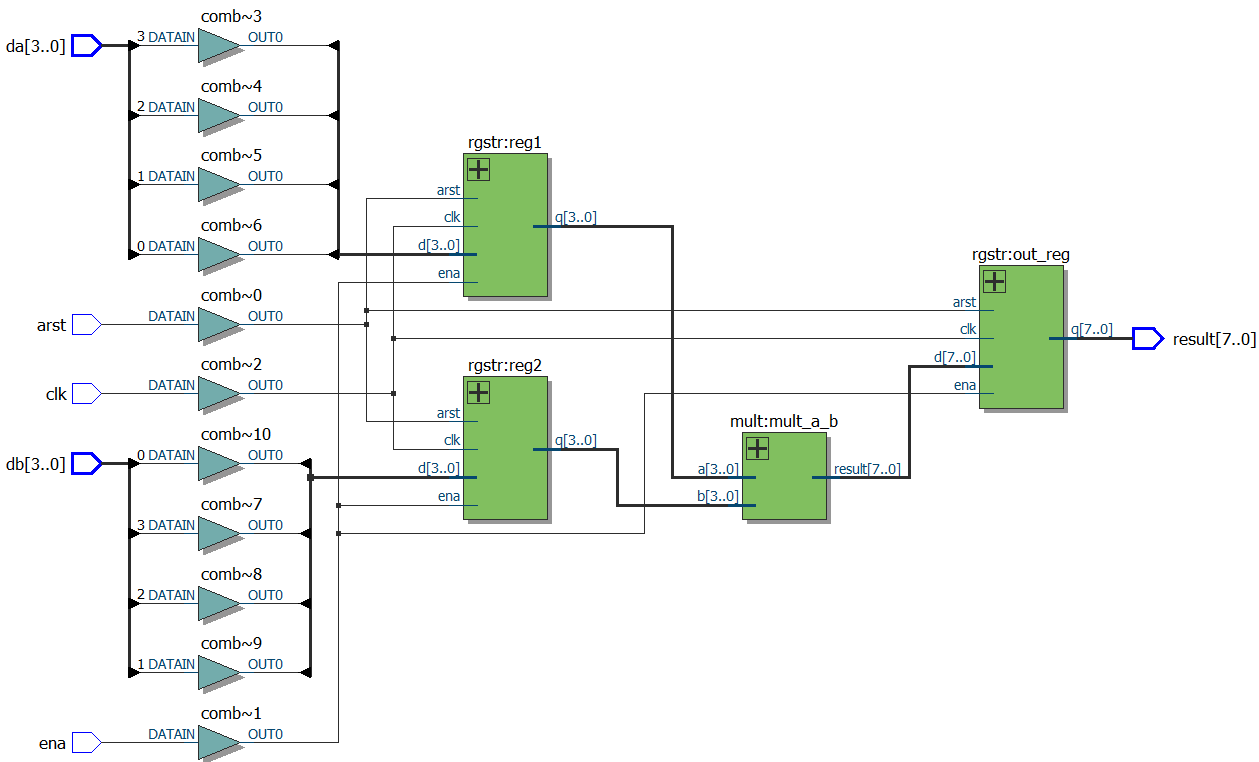


Рис.1.3 Синтезированная схема

## Моделирование

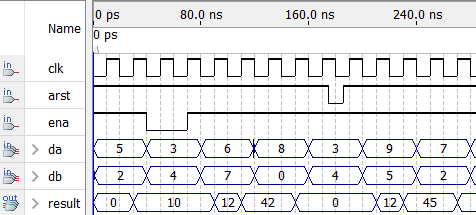


Рис.1.4 Результаты моделирования

## 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.

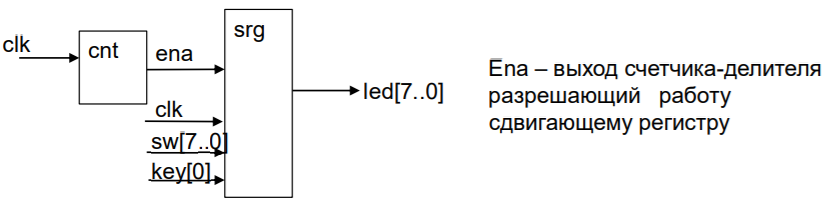
## Выводы

В ходе выполнения работы был спроектирован умножитель, работа которого синхронизирована тактовой частотой clk, ena разрешает работу, arst – асинхронный сброс. Выводы СБИС были назначены с использованием атрибутов.

# Задание lab5\_2

## Задание

На языке Verilog создать иерархическое (структурное) описание представленной ниже схемы:



* сnt – параметризированный счетчик-делитель: коэффициент деления задается параметром; выход – сигнал переноса
* srg – параметризированный кольцевой сдвигающий регистр с параллельной загрузкой, описание которого выполнено с помощью цикла for loop
* Входы:

– Clk – тактовый (25 МГц), sw[7..0] – данные загружаемые в сдвигающий регистр, key[0] – сигнал синхронной загрузки данных в сдвигающий регистр

* Выход – led[7..0] – содержимое сдвигающего регистра

## Описание на языке Verilog

|  |
| --- |
| module lab5\_2 (  (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "23"\*)  input clk,    (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "58"\*)  input load,      (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "88, 89, 90, 91, 49, 46, 25, 24"\*)  input [WIDTH\_REG-1:0] data,      (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "65, 66, 67, 68, 69, 70, 71, 72"\*)  output [WIDTH\_REG-1:0] led  );  parameter COUNT = 4;  parameter WIDTH = 24;  parameter WIDTH\_REG = 8;  wire ena;    counter #(WIDTH, COUNT) cnt (clk, ena);    shift\_rg #(WIDTH\_REG) srg (ena,clk, load, data, led);  endmodule  module shift\_rg(ena, clk, key, data, out);  parameter width = 8;  input ena, clk, key;  input [width-1:0] data;  output reg [width-1:0] out;  integer i;  always @ (posedge clk)  begin  if (ena)  begin  if(!key)  out <= data;  else  begin  out[0]<=out[width-1];  for(i=width-1; i>0; i=i-1)  out[i] <=out[i-1];  end  end  end  endmodule  module counter (  input clk,  output reg ena  );  parameter WIDTH = 3;  parameter COUNT = 4;  reg [WIDTH-1:0] tmp;  always @ (posedge clk)  begin  tmp <= tmp + 1'b1;    if (tmp == COUNT - 1)  begin  tmp <= 1'b0;  ena <= 1'b1;  end else  ena <= 1'b0;  end  endmodule |

Рис.2.2 Описание на языке SystemVerilog

## Результат синтеза (RTL)

## 

Рис. 2.3 Синтезированная схема

## Моделирование

Изображение выглядит как стол

Автоматически созданное описание

Рис.2.4 Результаты моделирования

## 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 работы был спроектирован счётчик-делитель c коэффициентом деления 4, кольцевой сдвигающий регистр с помощью цикла for loop, load – разрешение загрузки с входа data. Выводы СБИС были назначены с использованием атрибутов.

# Задание lab5\_3

## Задание

На языке Verilog на структурном уровне создайте иерархический проект, содержащий:

* На верхнем уровне: мультиплексор 2(4-разрядных входа)=>1 (4-разрядный выход),
* Состоит из мультиплексоров mux2\_1 (мультиплексор 2(1-разрядных входа)=>1 (1-разрядный выход))
* Мультиплексор mux2\_1 опишите на структурном уровне с использованием Gate-Level примитивов

языка Verilog.

* Выводы устройства:
* Входы:

– Переключатели sw[3:0] – вход inа

– Переключатели sw[7:4] – вход inb

– Кнопка pba – вход управления мультиплексором (кнопка нажата – на выход передается значение с входа ina,

кнопка не нажата - inb).

* Выходы

– led[3:0] подключены к светодиодам.

## Описание на языке Verilog

|  |
| --- |
| module lab5\_4 (  (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "49, 46, 25, 24"\*)  input [3:0] ina,  (\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"",  chip\_pin = "88, 89, 90, 91"\*)  input [3:0] inb,  (\*altera\_attribute = "-name IO\_STANDARD \"2.5-V\"",  chip\_pin = "64"\*)  input pba,  (\*altera\_attribute = "-name IO\_STANDARD \"2.5-V\"",  chip\_pin = "69, 70, 71, 72"\*)  output [3:0] led  );  mux2\_1 mux0 (.ina(ina[0]), .inb(inb[0]), .pba(pba), .res(led[0]));  mux2\_1 mux1 (.ina(ina[1]), .inb(inb[1]), .pba(pba), .res(led[1]));  mux2\_1 mux2 (.ina(ina[2]), .inb(inb[2]), .pba(pba), .res(led[2]));  mux2\_1 mux3 (.ina(ina[3]), .inb(inb[3]), .pba(pba), .res(led[3]));  endmodule  module mux2\_1 (ina, inb, pba, res);  input ina, inb, pba;  output res;  wire ina\_m;  wire inb\_m;  wire not\_pba;  not not\_r (not\_pba, pba);  and r1 (ina\_m, ina, not\_pba);  and r2 (inb\_m, inb, pba);  or (res, ina\_m, inb\_m);  endmodule |

Рис.2.1 Описание на SystemVerilog

## Результат синтеза (RTL)

## 

Рис. 3.3 Синтезированная схема

## Моделирование

## 

Рис.3.4 Результаты моделирования

## Тестирование на плате miniDiLaB-CIV

Для тестирования проекта на плате использовались тесты, описанные в разделе 3.4

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

## Выводы

В ходе выполнения работы был спроектирован мультиплексор, у которого есть три входа – ina, inb- информационные, pba – кнопка, по состоянию которой определяется, какой из сигналов будет подан на выход. Кнопка нажата – на выход передается значение с входа ina, кнопка не нажата – inb. Ina, inb, led – четырехразрядные. Проект иерархический: на верхнем уровне: мультиплексор, состоит из мультиплексоров mux2\_1, мультиплексор mux2\_1 описан на структурном уровне с использованием Gate-Level примитивов языка Verilog.Выводы СБИС были назначены с использованием атрибутов.

# Выводы

В ходе работы были получены навыки работы с модулями. Данные навыки полезны с практической точки зрения, потому что разбиение на модули необходимо использовать в сложных схемах для удобного чтения и повторного использования модулей. Тем самым в верхнем модуле мы описываем структурное описание всей схемы, а в модулях более низкого уровня мы описываем поведенческое описание работы устройства.