Санкт-Петербургский политехнический университет Петра Великого

Кафедра компьютерных систем и программных технологий

**Отчёт по лабораторной работе №5**

**Дисциплина**: Языки описания аппаратных средств вычислительных систем

Выполнил студент гр. 350901/90201 Фам Ба Нам

(подпись)

Преподаватель А.А.Федотов

(подпись)

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021г.

г. Санкт-Петербург

2021

1. **Ход работы :**
   1. **Упражнение 1:**

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

- параметризованное описание верхнего уровня

Создать проект:

- Имя проекта – lab5\_1, имя модуля верхнего уровня иерархии lab5\_1

Создать параметризируемый модуль регистра (базовое значение параметра – 8),

посмотреть синтезированную пакетом схему, осуществить его моделирование

Создать параметризированный модуль умножителя (базовое значение параметра – 8), посмотреть синтезированную пакетом схему, осуществить его моделирование

Создать структурное описание верхнего уровня (оно должно быть

параметризированным, базовое значение параметра - 4);,

Посмотреть синтезированную пакетом схему (RTL Viewer)

Осуществить моделирование

Назначить выводы СБИС с помощью атрибутов,

Осуществить компиляцию и проверить работу на плате.

**Тестовое описание схемы:**

- Регистр:

module rgstr (arst, ena, clk, d, q);

parameter width\_rg = 8;

input clk, arst, ena;

input [width\_rg-1:0] d;

output reg [width\_rg-1:0] q;

always @ (posedge clk, negedge arst)

if (arst == 1'b0) q<= {width\_rg{1'b0}};

else if (ena) q<= d;

endmodule

- Умножитель :

module mult (a,b, result);

parameter width\_mult = 8;

input [width\_mult-1:0] a, b;

output [2\*width\_mult-1:0] result;

assign result = a\*b;

endmodule

- Файл лаб5\_1

module lab5\_1(arst, ena, clk, da, db, result);

parameter width = 4;

(\*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 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 = "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 input\_rg\_1 (arst, ena, clk , da, da\_int);

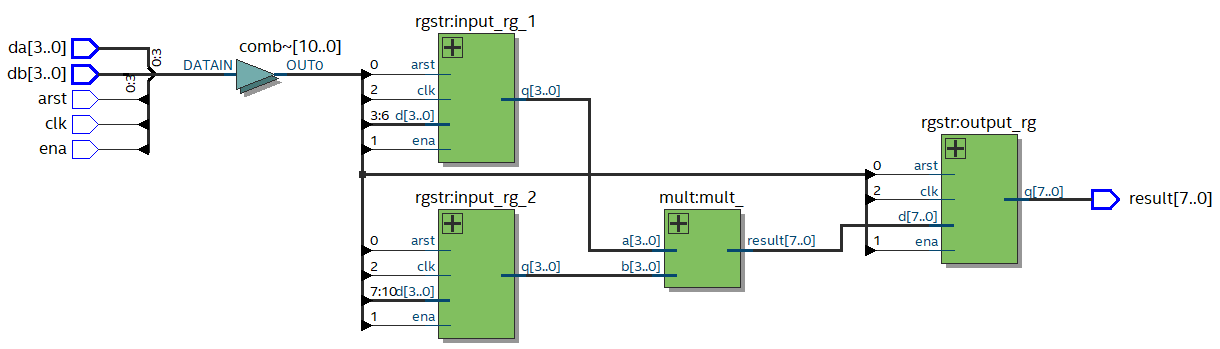
rgstr #width input\_rg\_2 (.arst(arst), .ena(ena), .clk(clk), .d(db), .q(db\_int));

mult #width mult\_ (da\_int, db\_int, mult\_res);

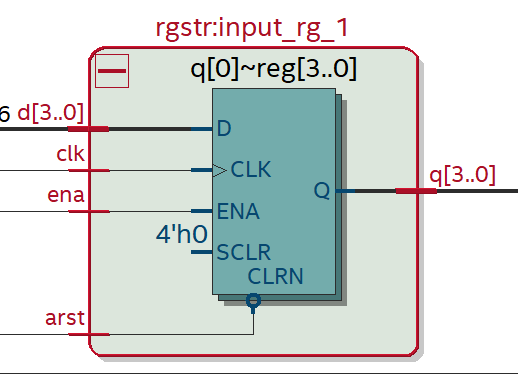
rgstr #(2\*width) output\_rg (arst, ena, clk, mult\_res, result);

endmodule

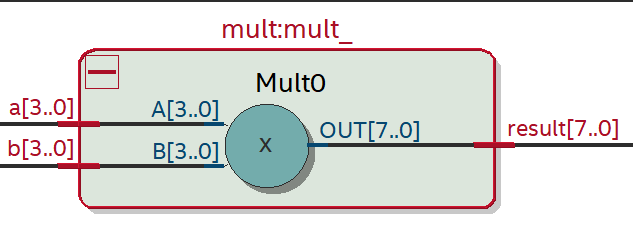
**Синтезированная схема :**



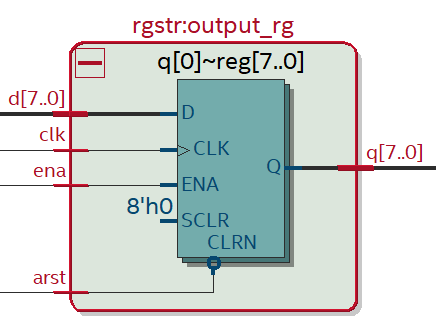
- Модуль регистра :



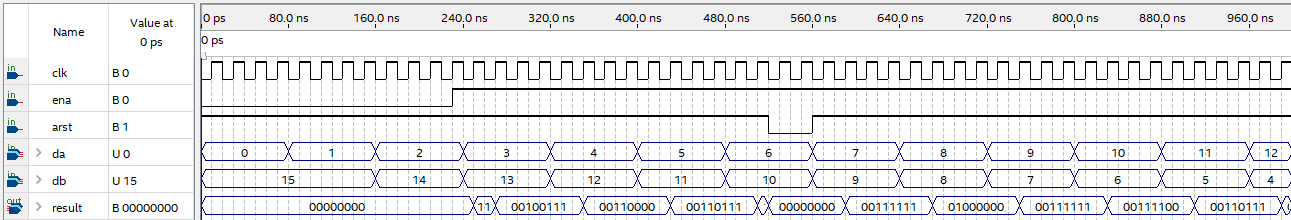
- Модуль умножителя :



-Модуль регистра (output)

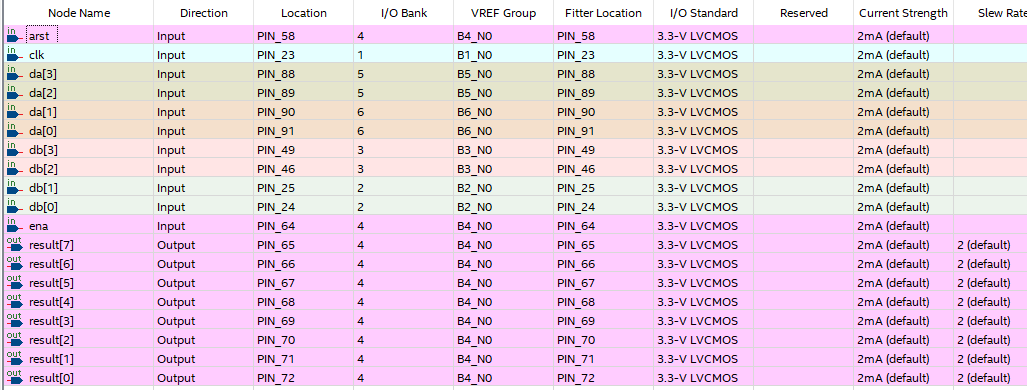


**Результат моделирования :**



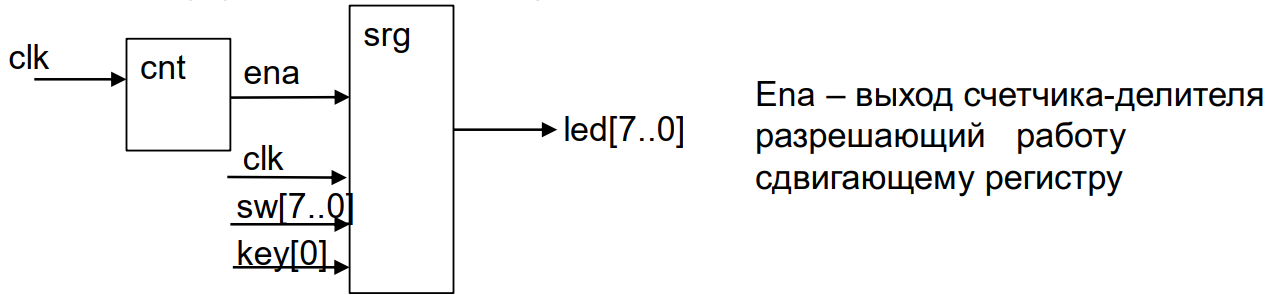
Результаты моделирования польностью совпадают с ожидаемыми

**Были заданы выводы СБИС :**



* 1. **Упражение 2 :**

На языке Verilog создать иерархическое описание представленной ниже схемы :



- cnt – парамеиризованный счеичик-делитель: коэффицент деления задается параметром; выход – сигнал перенос

- srg – параметризованный кольцевой сдвигающий регистр с параллельной загрузкой, описание которого выполнено с помощью цмкла for loop

- выход – led[7..0] – содеожимое сдвигающего регистра

Создать проект:

- Имя проекта – lab5\_2, имя модуля верхнего уровня иерархии lab5\_2

Создать параметризируемый модуль счетчика делителя (базовое значение параметра – 3), посмотреть синтезированную пакетом схему, осуществить его моделирование

Создать параметризированный модуль кольцевого сдвигающего регистра с параллельной загрузкой (базовое значение параметра – 8), посмотреть синтезированную пакетом схему, осуществить его моделирование

Создать структурное описание верхнего уровня (оно должно быть параметризированным: задаются параметры:

- деления частоты для счетчика делителя (базовое значение параметра - 4),

- деления частоты для счетчика делителя (базовое значение параметра - 4),

Посмотреть синтезированную пакетом схему (RTL Viewer)

Осуществить моделирование

Назначить параметр счетчик делителя для реализации на плпте

Назначить выводы СБИС с помощью атрибутов

Осуществить компиляцию и проверить работу на плате

**Тестовое описание схемы :**

- счетчик-делитель:

module cnt(clk,ena);

input clk;

output reg ena;

parameter width\_cnt = 3;

parameter width\_max =3;

reg [width\_cnt-1:0] tmp;

always @ (posedge clk)begin

tmp <= tmp +4'b1;

if (tmp == width\_max-1) begin

tmp <= 4'b0;

ena <= 1'b1;end

else ena <= 1'b0;

end

endmodule

- сдвигающий регистр:

module srg(clk, ena, sw70, key0,led70);

parameter width\_srg = 8;

input clk,ena,key0;

input [width\_srg-1:0] sw70;

output reg [width\_srg-1:0] led70;

integer i;

always @(posedge clk) begin

if(ena)

if(!key0) led70 <= sw70 ;

else begin

led70[0]<=led70[width\_srg-1];

for(i=1;i<=width\_srg-1;i=i+1)

led70[i]<=led70[i-1];

end

end

endmodule

- файл лаб5\_2:

module lab5\_2(clk, sw70, key0, led70);

parameter width\_cnt = 25;

parameter width\_max = 25000000;

parameter width\_srg = 8;

(\*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 = "64"\*)

input key0;

(\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"", chip\_pin = "88, 89, 90, 91, 49, 46, 25, 24"\*)

input [width\_srg-1:0]sw70;

(\*altera\_attribute = "-name IO\_STANDARD \"3.3-V LVCMOS\"", chip\_pin = "65, 66, 67, 68, 69, 70, 71, 72"\*)

output [width\_srg-1:0]led70;

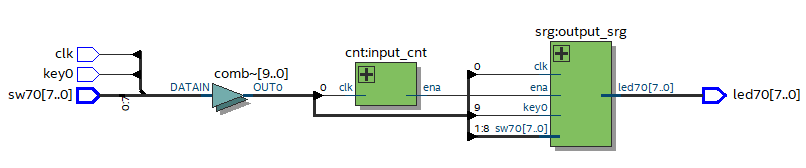
wire ena;

cnt #(.width\_cnt(width\_cnt),.width\_max(width\_max)) input\_cnt (clk, ena);

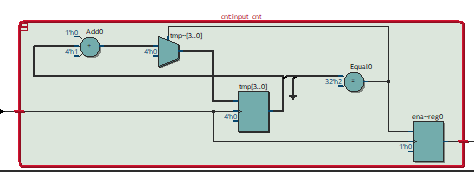
srg #(width\_srg) output\_srg (.clk(clk), .ena(ena), .sw70(sw70), .key0(key0), .led70(led70));

endmodule

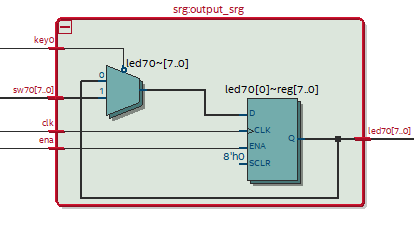
**синтезированная схема :**



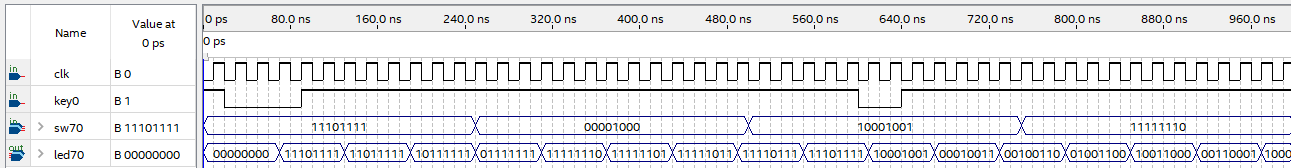
- модуль счетчик :



- модуль сдвигающего регистра :

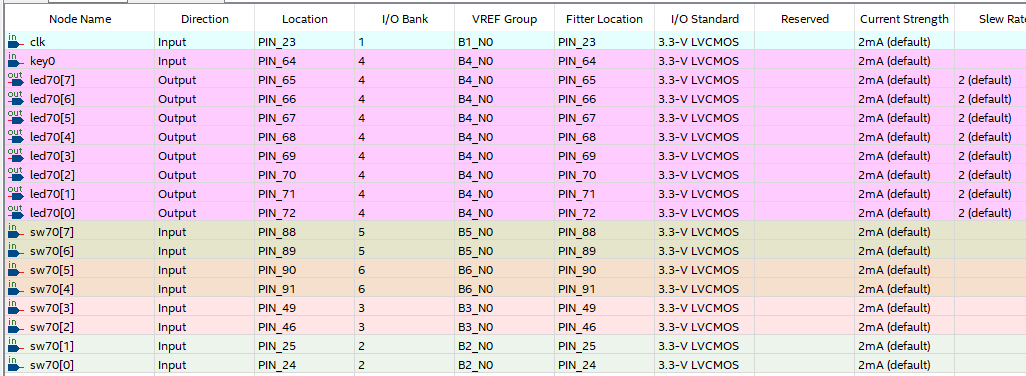


**Результаты моделирования** :



Результаты моделирования полбностью совпадают с ожидаемыми

**Были заданы выводы СБИС :**



1. **Вывод :**

В ходе работы были получены навыки работы с модулями. Данные навыки полезны с практической точки зрени, 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.