

Санкт-Петербургский политехнический университет Петра Великого  
Институт компьютерных наук и технологий  
Кафедра компьютерных систем и программных технологий

**Отчет по лабораторной работе №1\_2**  
**Курс: «Проектирование реконфигурируемых гибридных**  
**вычислительных систем»**  
**Тема: «Введение в Vivado HLS»**

Выполнил студент гр. 3540901/81501

Селиверстов С.А.

(подпись)

Руководитель

Антонов А.П.

(подпись)

“ \_\_\_\_ ” \_\_\_\_\_ 2019 г.

Санкт – Петербург  
2019

## Оглавление

Задание.....	3
Решение №1 .....	5
Моделирование .....	5
Синтез .....	5
C/RTL моделирование.....	8
Решение №2 .....	9
Моделирование .....	9
Синтез .....	10
C/RTL моделирование.....	12
Выводы .....	13

## Задание

Создать проект, подключив готовые файлы исходного кода устройства и теста к нему.

Исходный код функции:

```
void lab1_2 (int in[3], char a, char b, char c, int out[3]) {
    int x,y;
    for(int i = 0; i < 3; i++) {
        x = in[i];
        y = a*x + b + c;
        out[i] = y;
    }
}
```

Исходный код теста:

```
#include <stdio.h>
int main()
{
    int In[3] = {1,3,9};
    int inA, inB, inC;
    int res[3];
    // For adders
    int refOut[9] = {30, 40, 70, 90, 140, 290, 150, 240, 510};
    int pass;
    int i;
    inA = 5;
    inB = 10;
    inC = 15;
    for (i=0; i<3; i++)
    {
        lab1_2(In, inA, inB, inC, res);
        for (int j=0; j<3; j++)
        {
            fprintf(stdout, " %d*d+%d+%d=%d \n", inA, In[j], inB, inC, res[j]);

            // Test the output against expected results
            if (res[j] == refOut[i*3+j])
                pass = 1;
            else
                pass = 0;
        }
        inA=inA+20;
        inB=inB+20;
        inC=inC+20;
    }

    if (pass)
    {
        fprintf(stdout, "-----Pass!-----\n");
        return 0;
    }
    else
    {
        fprintf(stderr, "-----Fail!-----\n");
        return 1;
    }
}
```

Создать 2 решения для синтеза устройства на основе *xa7a12tcsg325-1q*: задать clock period 6 и clock period 10, а также clock uncertain 0.1 в обоих случаях.

Для обоих решений осуществить моделирование и синтез, привести в данном отчете:

- performance estimates=>summary
- utilization estimates=>summary
- Performance Profile
- scheduler viewer (выполнить Zoom to Fit)
  - На скриншоте показать Latency
  - На скриншоте показать Initiation Interval
- resource viewer (выполнить Zoom to Fit)
  - На скриншоте показать Latency
  - На скриншоте показать Initiation Interval

Также для обоих решений осуществить C|RTL моделирование:

- Отобразить два цикла обработки на одном экране
  - На скриншоте показать Latency
  - На скриншоте показать Initiation Interval

Написать выводы, где среди прочего объяснить отличие двух решений.

## Решение №1

При создании решения зададим настройки: clock period 6, clock uncertain 0.1, устройство *xa7a12tcs325-1q*.

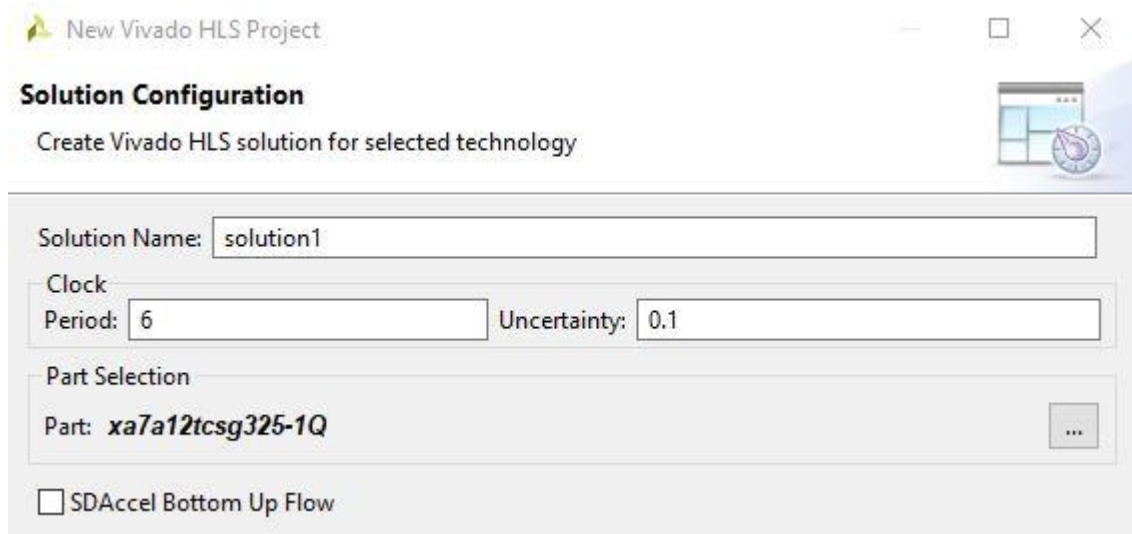


Рис. 1. Конфигурация проекта

## Моделирование

При запуске моделирования можно увидеть, что тест успешно пройден:

```
INFO: [HLS 200-10] Setting target device to 'xa7a12t-csg325-1Q'
INFO: [SIM 211-2] ***** CSIM start *****
INFO: [SIM 211-4] CSIM will launch GCC as the compiler.
Compiling(apcc) ../../../../../../Desktop/SPBGPU_Mag/Antonov/lab_1/lab1_2.c in debug mode
INFO: [HLS 200-10] Running 'C:/Xilinx/Vivado/2019.1/bin/unwrapped/win64.o/apcc.exe'
INFO: [HLS 200-10] For user 'Yaroslav' on host 'svytoslavpc' (Windows NT amd64 version 6.1) on Tue Oct 08 14:15:24 +0300 2019
INFO: [HLS 200-10] In directory 'C:/Users/Yaroslav/AppData/Roaming/Xilinx/Vivado/lab1_2/solution1/csim/build'
INFO: [APCC 202-3] Tmp directory is apcc_db
INFO: [APCC 202-1] APCC is done.
Generating csim.exe
5*1+10+15=30
5*3+10+15=40
5*9+10+15=70
25*1+30+35=90
25*3+30+35=140
25*9+30+35=290
45*1+50+55=150
45*3+50+55=240
45*9+50+55=510
-----Pass!-----
INFO: [SIM 211-1] CSim done with 0 errors.
INFO: [SIM 211-3] ***** CSIM finish *****
```

## Синтез

Приведем в отчете требуемые данные о проекте:

## Synthesis Report for 'lab1\_2'

### General Information

Date: Tue Oct 8 14:16:20 2019  
Version: 2019.1 (Build 2552052 on Fri May 24 15:28:33 MDT 2019)  
Project: lab1\_2  
Solution: solution1  
Product family: artix7  
Target device: xa7a12t-csg325-1Q

### Performance Estimates

#### Timing (ns)

##### Summary

Clock	Target	Estimated	Uncertainty
ap_clk	6.00	5.690	0.10

Рис. 2. Информация о проекте

Здесь можно увидеть, что достигнутая задержка равна  $5.690 + 0.1$ , что укладывается в заданные нами требования к тактовой частоте.

### Utilization Estimates

#### Summary

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	-	0	73	-
FIFO	-	-	-	-	-
Instance	-	2	166	49	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	50	-
Register	-	-	141	-	-
Total	0	2	307	172	0
Available	40	40	16000	8000	0
Utilization (%)	0	5	1	2	0

Рис. 3. Занимаемые ресурсы

Данный проект займет на микросхеме 2 DSP блока (в которых будут использованы и сумматоры и умножитель), 307 триггеров для хранения чисел, и 172 LUT.

	Pipelined	Latency	Iteration Latency	Initiation Interval	Trip count
lab1_2	-	19	-	20	-
Loop 1	no	18	6	-	3

Рис. 4. Профиль производительности

На этом рисунке видно, что задержка получения выходного значения составляет 19 тактов с момента старта, а задержка после старта до готовности приема новых данных – 20. Покажем эти интервалы на временной диаграмме:

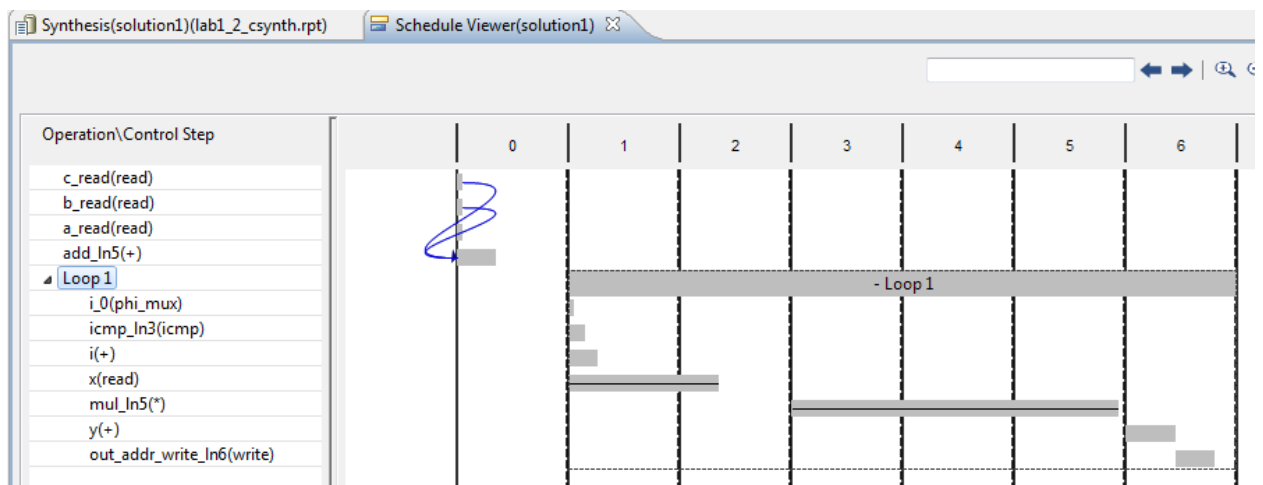


Рис. 5. Временная диаграмма

Здесь мы видим весь процесс получения результата. На первом такте происходит считывание операторов А, В и С, а также начинается их сложение. На втором такте начинает цикл длиной в 6 тактов, который повторяется 3 раза (всего 18 тактов). Таким образом суммарная задержка  $latency = 19$ , а со следующего 20-го такта можно подавать следующие данные ( $\Pi = 20$ ).

Приведем профиль ресурсов:

	BRAM	DSP	FF	LUT	Bits P0	Bits P1	Bits P2	Banks/Depth	Words	W*Bits*Banks
lab1_2	0	2	307	172						
I/O Ports(5)					88					
Instances(1)	0	2	166	49						
Memories(0)	0		0	0	0		0		0	0
Expressions(4)	0	0	0	73	45	44	0			
Registers(8)			141		203					
Channels(0)	0		0	0	0		0		0	0
Multiplexers(2)	0		0	50	3		0			
DSP(1)		0								

Рис. 6. Профиль ресурсов

Здесь можно увидеть те же числа, что и в отчете синтезатора.

### C/RTL моделирование

При совместном моделировании, программа отобразила те же самые, ожидаемые нами значения Latency и II:

#### Cosimulation Report for 'lab1\_2'

Result							
RTL	Status	Latency			Interval		
		min	avg	max	min	avg	max
VHDL	NA	NA	NA	NA	NA	NA	NA
Verilog	Pass	19	19	19	20	20	20

Export the report(.html) using the [Export Wizard](#)

Рис. 7. Результаты C/RTL моделирования

Покажем временную диаграмму совместного моделирования с отмеченными на ней Latency и II:

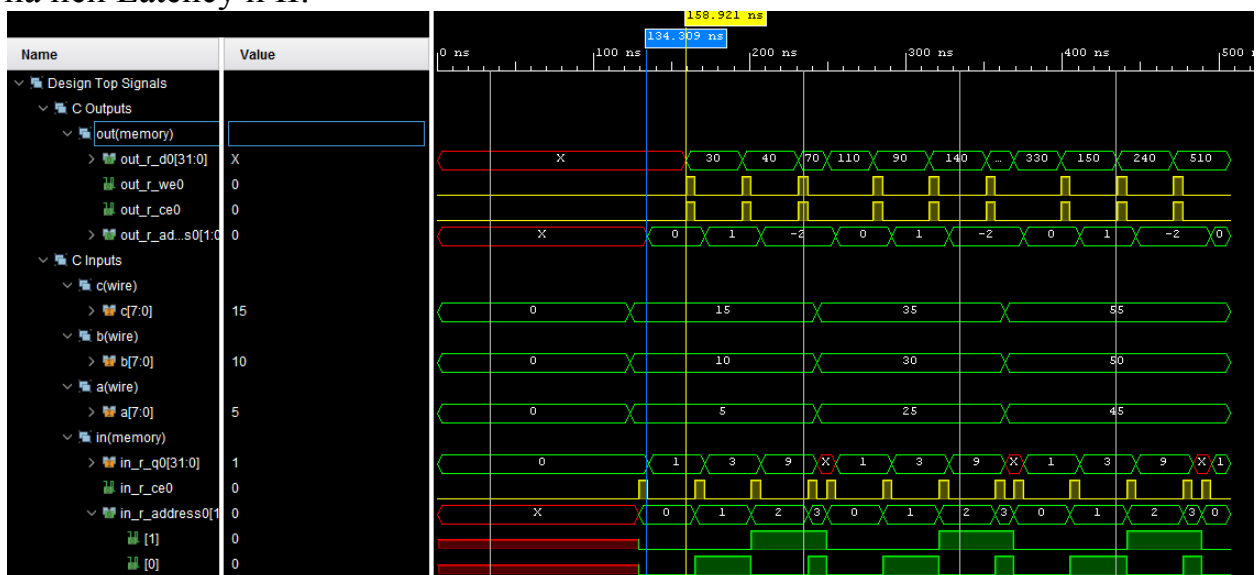


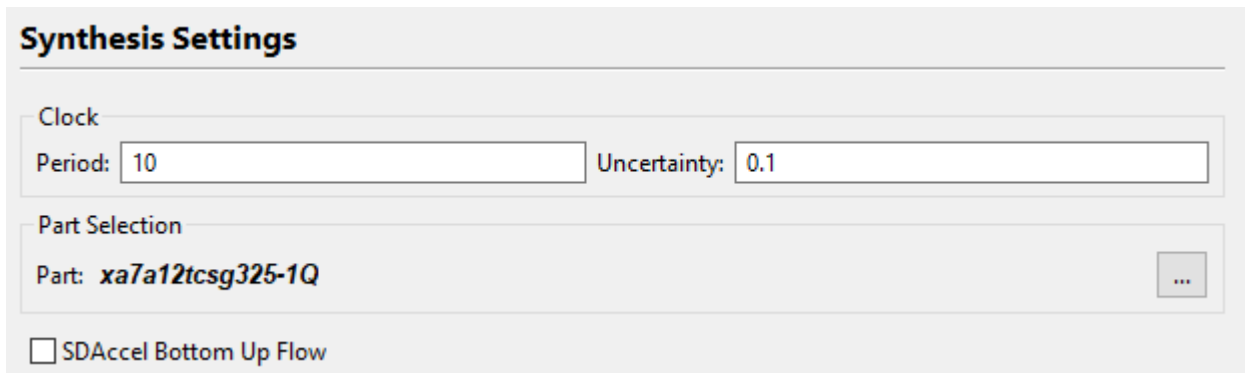
Рис. 8. Временная диаграмма совместного моделирования



## Решение №2

### Моделирование

Создадим второе решение для данного проекта. Его настройки:



**Synthesis Settings**

**Clock**

Period:  Uncertainty:

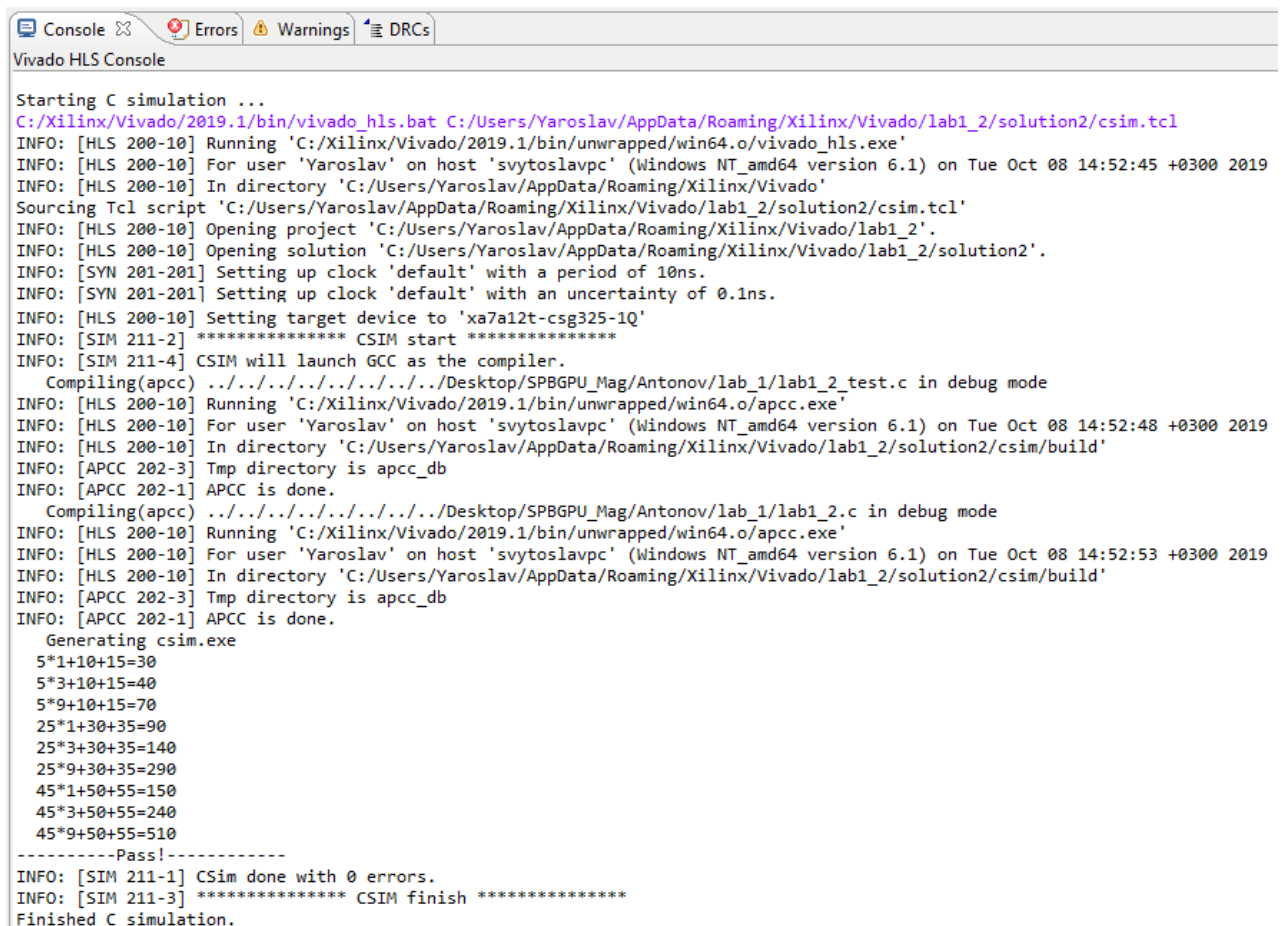
**Part Selection**

Part: **xa7a12tcs325-1Q**

☐ SDAccel Bottom Up Flow

Рис. 10. Конфигурация проекта

Настройки этого решения изменены согласно заданию.



```
Starting C simulation ...
C:/Xilinx/Vivado/2019.1/bin/vivado_hls.bat C:/Users/Yaroslav/AppData/Roaming/Xilinx/Vivado/lab1_2/solution2/csim.tcl
INFO: [HLS 200-10] Running 'C:/Xilinx/Vivado/2019.1/bin/unwrapped/win64.o/vivado_hls.exe'
INFO: [HLS 200-10] For user 'Yaroslav' on host 'svytoslavpc' (Windows NT_amd64 version 6.1) on Tue Oct 08 14:52:45 +0300 2019
INFO: [HLS 200-10] In directory 'C:/Users/Yaroslav/AppData/Roaming/Xilinx/Vivado'
Sourcing Tcl script 'C:/Users/Yaroslav/AppData/Roaming/Xilinx/Vivado/lab1_2/solution2/csim.tcl'
INFO: [HLS 200-10] Opening project 'C:/Users/Yaroslav/AppData/Roaming/Xilinx/Vivado/lab1_2'
INFO: [HLS 200-10] Opening solution 'C:/Users/Yaroslav/AppData/Roaming/Xilinx/Vivado/lab1_2/solution2'
INFO: [SYN 201-201] Setting up clock 'default' with a period of 10ns.
INFO: [SYN 201-201] Setting up clock 'default' with an uncertainty of 0.1ns.
INFO: [HLS 200-10] Setting target device to 'xa7a12t-csg325-1Q'
INFO: [SIM 211-2] ***** CSIM start *****
INFO: [SIM 211-4] CSIM will launch GCC as the compiler.
Compiling(apcc) ../../../../../../Desktop/SPBGPU_Mag/Antonov/lab_1/lab1_2_test.c in debug mode
INFO: [HLS 200-10] Running 'C:/Xilinx/Vivado/2019.1/bin/unwrapped/win64.o/apcc.exe'
INFO: [HLS 200-10] For user 'Yaroslav' on host 'svytoslavpc' (Windows NT_amd64 version 6.1) on Tue Oct 08 14:52:48 +0300 2019
INFO: [HLS 200-10] In directory 'C:/Users/Yaroslav/AppData/Roaming/Xilinx/Vivado/lab1_2/solution2/csim/build'
INFO: [APCC 202-3] Tmp directory is apcc_db
INFO: [APCC 202-1] APCC is done.
Compiling(apcc) ../../../../../../Desktop/SPBGPU_Mag/Antonov/lab_1/lab1_2.c in debug mode
INFO: [HLS 200-10] Running 'C:/Xilinx/Vivado/2019.1/bin/unwrapped/win64.o/apcc.exe'
INFO: [HLS 200-10] For user 'Yaroslav' on host 'svytoslavpc' (Windows NT_amd64 version 6.1) on Tue Oct 08 14:52:53 +0300 2019
INFO: [HLS 200-10] In directory 'C:/Users/Yaroslav/AppData/Roaming/Xilinx/Vivado/lab1_2/solution2/csim/build'
INFO: [APCC 202-3] Tmp directory is apcc_db
INFO: [APCC 202-1] APCC is done.
Generating csim.exe
5*1+10+15=30
5*3+10+15=40
5*9+10+15=70
25*1+30+35=90
25*3+30+35=140
25*9+30+35=290
45*1+50+55=150
45*3+50+55=240
45*9+50+55=510
-----Pass!-----
INFO: [SIM 211-1] CSim done with 0 errors.
INFO: [SIM 211-3] ***** CSIM finish *****
Finished C simulation.
```

Моделирование второго решения также прошло успешно.

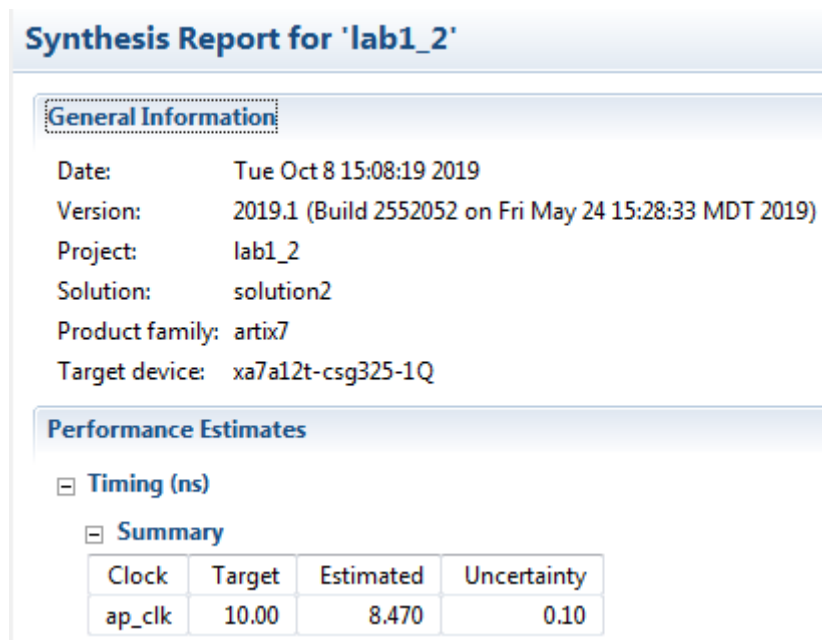


Рис. 11. Производительность

На этом рисунке видно, что данное решение также укладывается в установленный период тактовой частоты, однако задержки сильно отличаются, что означает, что изменения периода тактовой частоты повлекло за собой существенные изменения устройства.

На примере затрачиваемых на решение ресурсов это также хорошо видно:

**Utilization Estimates**

▢ Summary

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	2	0	94	-
FIFO	-	-	-	-	-
Instance	-	-	-	-	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	42	-
Register	-	-	139	-	-
<b>Total</b>	<b>0</b>	<b>2</b>	<b>139</b>	<b>136</b>	<b>0</b>
Available	40	40	16000	8000	0
Utilization (%)	0	5	~0	1	0

Рис. 12. Затрачиваемые ресурсы

В сравнении с предыдущим решением в схеме количество триггеров сократилось на 168 ( $\Delta FF = FF1 - FF2 = 307 - 139 = 168$ ), а LUT на 36 ( $\Delta LUT = LUT1 - LUT2 = 172 - 132 = 36$ ).

Performance Profile		Resource Profile			
	Pipelined	Latency	Iteration Latency	Initiation Interval	Trip count
lab1_2	-	13	-	14	-
Loop 1	no	12	4	-	3

Рис. 13. Профиль производительности

Сразу видно, что задержка цикла уменьшилась с 6 до 4 тактов, а новые данные можно подавать уже через 14 тактов. Покажем это на временной диаграмме:

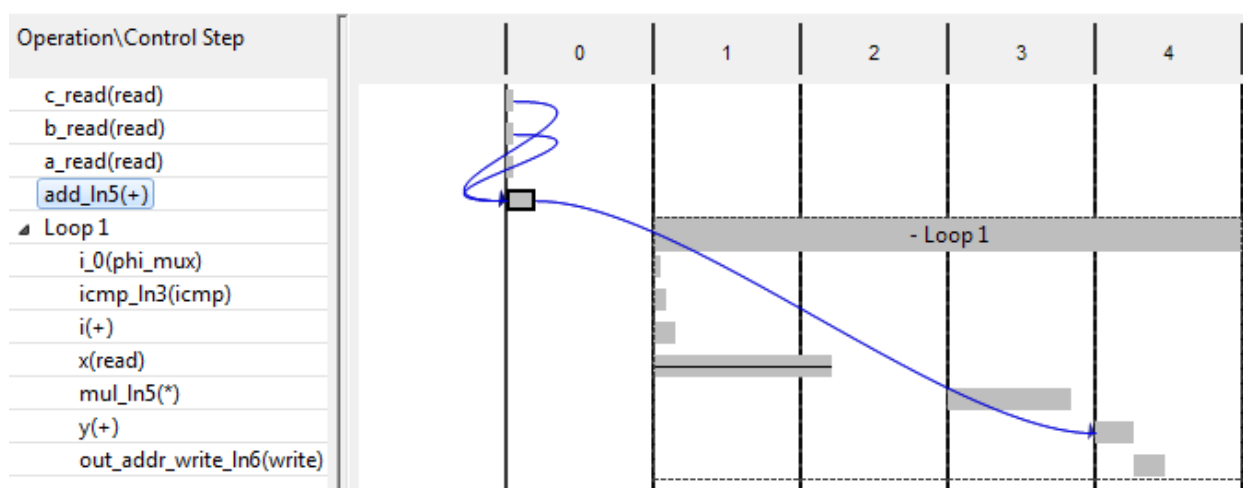


Рис. 14. Временная диаграмма

Если изучить эту диаграмму подробнее, то можно сказать, что умножение занимает значительно меньше тактов.

Performance Profile		Resource Profile								
	BRAM	DSP	FF	LUT	Bits P0	Bits P1	Bits P2	Banks/Depth	Words	W*Bits*Banks
lab1_2	0	2	139	136						
I/O Ports(5)					88					
Instances(0)	0	0	0	0						
Memories(0)	0		0	0	0		0		0	0
Expressions(5)	0	2	0	94	77	52	0			
Registers(8)			139		201					
Channels(0)	0		0	0	0		0		0	0
Multiplexers(2)	0		0	42	3		0			
DSP(0)		0								

Рис. 15. Профиль ресурсов

Здесь мы также видим отличия, согласно общему отчету о затраченных ресурсах.

C/RTL моделирование

При осуществлении совместного моделирования программа показала ожидаемые результаты Latency, но П также стал нулевым:

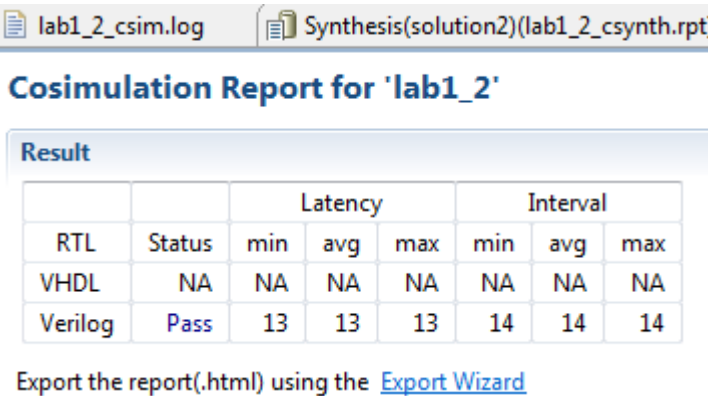


Рис. 16. C/RTL моделирование

Покажем временную диаграмму совместного моделирования:

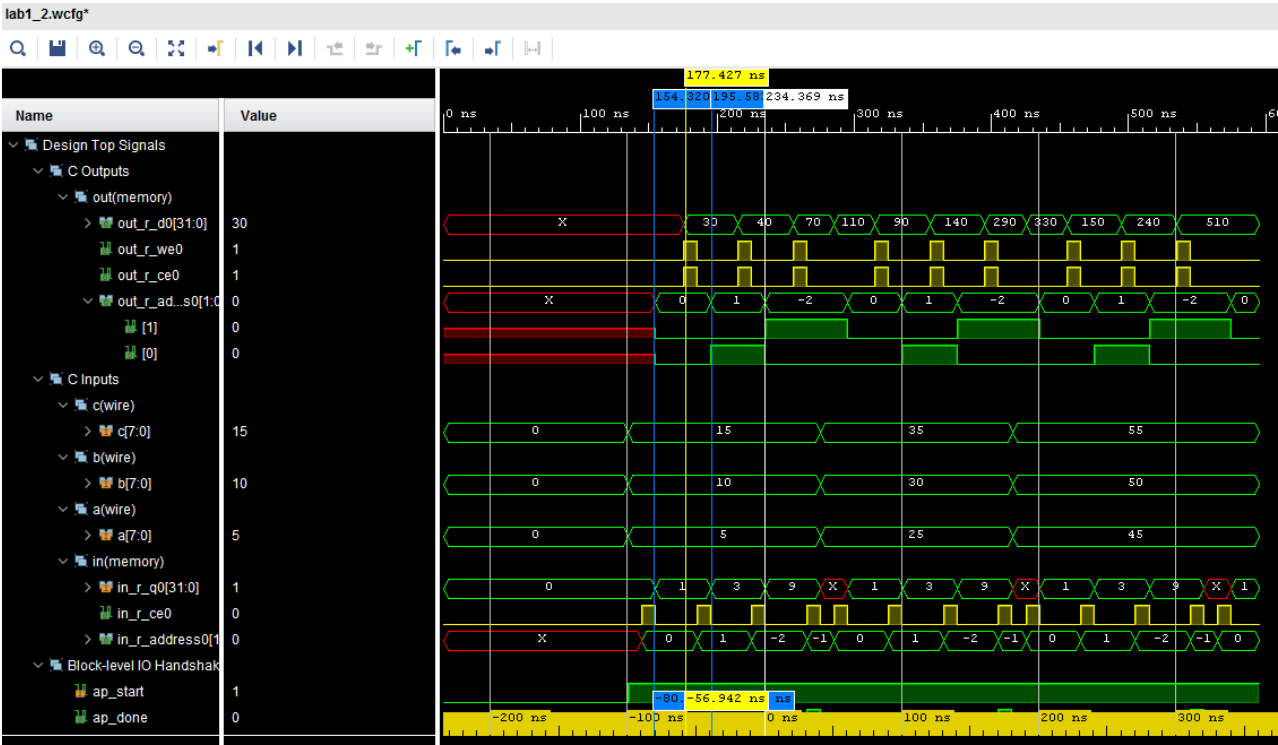


Рис. 17. Временная диаграмма совместного моделирования

Здесь видно, что задержка подачи новых данных сократилась до 14 циклов.

## Выводы

В данной работе главным образом были рассмотрены синтезируемые устройства при наличии в программе циклов. В данном примере видно, что при большем периоде тактов, программа сумела уместить расчет одного цикла в меньшее число тактов. Таким образом, получены 2 решения: первое – полный цикл выполнения 20 тактов, а максимальная задержка обработки сигнала на такте составляет 5.690 нс, и второе – 14 тактов, задержка в котором уже 8.470 нс, но все выполняется всего за меньшее число тактов.