

- Создать проект lab7\_z1
- Микросхема: **xc7a100tcsg324-2**
- Создать на языке C++ функцию (N=128, din\_type – int, dout\_type - int),

```
1  #include "lab7_z1.h"
2
3  void lab7_z1(dout_type a[N], din_type b[N], din_type c[N])
4  {
5      dout_type temp_mult;
6      Mult:for (int i = 0; i < N; i++)
7      {
8          temp_mult = b[i] * c[i] ;
9          a[i] = temp_mult;
10     }
11 }
```

- Создать тест lab7\_z1\_test.cpp для проверки функции (не менее трех запусков функции) . Осуществить моделирование (с выводом результатов в консоль).

## Исследование:

- Solution1
  - clock period 10 ; clock\_uncertainty =1
  - Выключите конвейеризацию для цикла Mult
  - осуществите синтез.
  - Посмотрите на отчет – должен быть похож на приведенный ниже

Target	Estimated	Uncertainty
10.00 ns	8.470 ns	1.00 ns

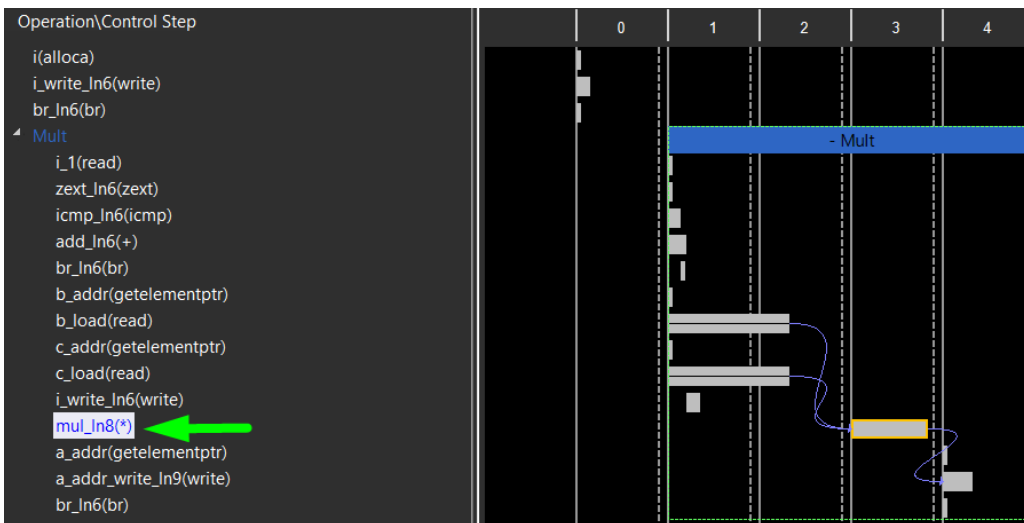
  

Modules & Loops	Issue Type	Violation Type	Distance	Slack	Latency(cycles)	Latency(ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	DSP	FF	LUT	URAM
lab7_z1				-	513	5,130E3	-	514	-	no	0	3	117	81	0
Mult				-	512	5,120E3	4	-	128	no	-	-	-	-	-

Сколько модулей умножения (DSP) требуется для реализации?

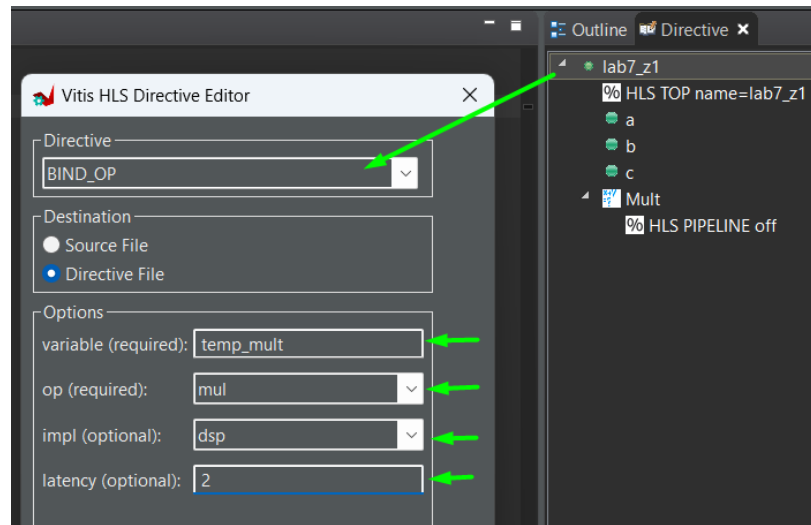
Какой период тактового сигнала (оцениваемый)?

- Посмотрите на Schedule Viewer – должен быть похож на приведенный ниже



Сколько тактов занимает выполнение операции умножения?

- Solution1\_1
  - clock period 10 ; clock\_uncertainty =1
  - Выключите конвейеризацию для цикла Mult
  - Для переменной temp\_mult задайте директиву BIND\_OP



- осуществите синтез.
- Посмотрите на отчет – должен быть похож на приведенный ниже

**Timing Estimate**

Target	Estimated	Uncertainty
10.00 ns	7.205 ns	1.00 ns

**Performance & Resource Estimates**

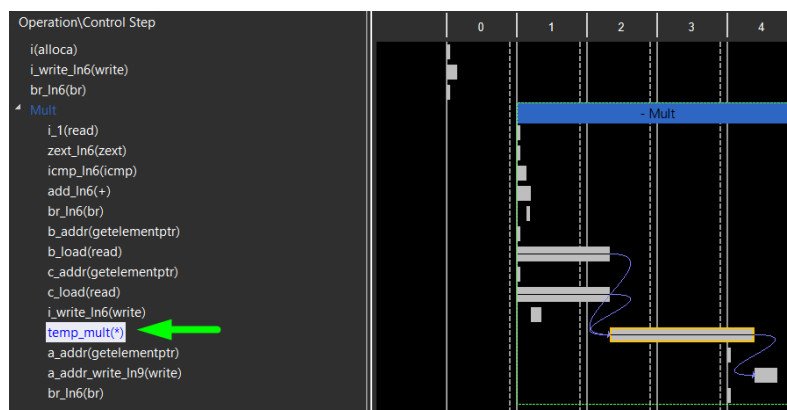
Modules & Loops	Issue Type	Violation Type	Distance	Slack	Latency(cycles)	Latency(ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	DSP	FF	LUT	URAM
lab7_z1				-	513	5,130E3		514	-	no	0	3	21	107	0
Mult				-	512	5,120E3	4	-	128	no	-	-	-	-	-

Сколько модулей умножения (DSP) требуется для реализации?

Сколько всего умножителей в данной микросхеме? Какой максимальный Unroll Factor цикла MULT (так чтобы использовались только DSP) может быть задан для данной микросхемы?

Какой период тактового сигнала (оцениваемый)?

- Посмотрите на Schedule Viewer – должен быть похож на приведенный ниже



Сколько тактов занимает выполнение операции умножения?

- Solution1\_2 (на основе решений solution 1\_1)
  - clock period **15** ; clock\_uncertainty =1
  - Выключите конвейеризацию для цикла Mult
  - Для переменной temp\_mult задайте директиву BIND\_OP: fabric

- осуществите синтез.
- Посмотрите на отчет – должен быть похож на приведенный ниже

**Timing Estimate**

Target	Estimated	Uncertainty
15.00 ns	12.379 ns	1.00 ns

**Performance & Resource Estimates**

Modules & Loops	Issue Type	Violation Type	Distance	Slack	Latency(cycles)	Latency(ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	DSP	FF	LUT	URAM
lab7_z1				-	769	1,154E4	-	770	-	no	0	0	119	1110	0
Mult				-	768	1,152E4	6	-	128	no	-	-	-	-	-

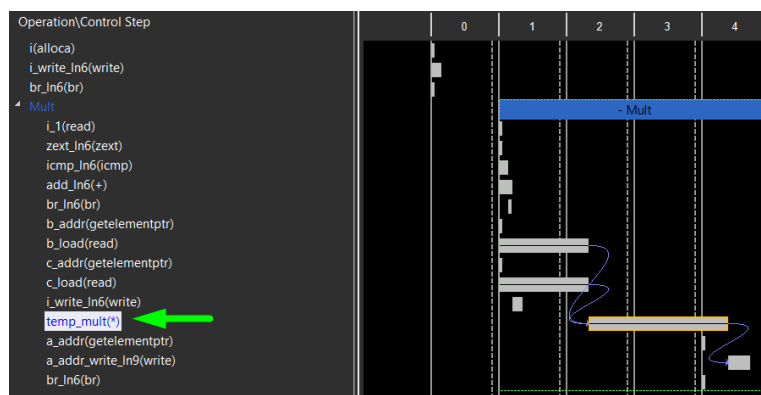
Сколько модулей умножения (DSP) требуется для реализации?

Сколько логических элементов (LUT) требуется для реализации?

Сколько всего LUT в данной микросхеме? Какой максимальный Unroll Factor цикла MULT (так чтобы использовались только LUT) может быть задан для данной микросхемы?

Какой период тактового сигнала (оцениваемый)?

- Посмотрите на Schedule Viewer – должен быть похож на приведенный ниже



Сколько тактов занимает выполнение операции умножения?

- Solution 2 (на основе решения **solution 1**)
  - clock period 10 ; clock\_uncertainty =1
  - Задайте максимально возможный Unroll Factor цикла MULT (чтобы использовались только DSP)
  - Используйте Array Partition (или Array Reshape) нужного типа и фактора, обеспечивающего балансировку производительности умножителей и чтения/записи данных (использование одно портовой или двух портовой памяти – на ваш выбор).
  - **Включите** конвейеризацию для цикла Mult
  - Для переменной temp\_mult задайте директиву BIND\_OP

The screenshot shows the 'Directive' configuration window in the Xilinx IDE. The 'Directive' dropdown is set to 'BIND\_OP'. Under the 'Destination' section, 'Directive File' is selected with a radio button. The 'Options' section contains the following fields: 'variable (required):' with the value 'temp\_mult', 'op (required):' with the value 'mul', 'impl (optional):' with the value 'dsp', and 'latency (optional):' with the value '1'.

- осуществите синтез.
- Посмотрите на отчет

Сколько модулей умножения (DSP) требуется для реализации?

Какой период тактового сигнала (оцениваемый)?

Сколько тактов занимает выполнение операции умножения?

- Посмотрите на Schedule Viewer

Сколько операций считывания данных осуществляется параллельно?

Сколько операций умножения осуществляется параллельно?

Сколько операций записи данных осуществляется параллельно?

Осуществляется ли конвейеризация? Какой II?

- Запустите CoSimulation,
  - Посмотрите и зафиксируйте Wave Viewer – подготовьтесь дать пояснения.

### Измерение времени выполнения на ПК

- Используются исходные коды функции lab7\_z1.cpp (**solution\_2**)
- На базе теста **lab7\_z1\_test.cpp** **следует** создать отдельный, модернизированный, тест **lab7\_z1\_testSW.cpp** (сохранить в папке C:\Xilinx\_trn\HLS2023\lab7\_z1\source) для проверки времени выполнения функции lab7\_z1 на ПК. Исходные данные входных массивов должны быть псевдослучайными **из всего диапазона int**.
- Следует осуществить компиляцию модернизированного теста и запускать его как отдельное приложение
  - Следует сделать две реализации кода
    - Для одного ядра (потока) – базовая реализация
    - Для N ядер/потоков (где N число ядер/потоков в вашем ПК) – например так:  
<https://stackoverflow.com/questions/414714/compiling-with-g-using-multiple-cores>
- Следует провести измерение времени выполнения синтезируемой функции на Вашем ПК **для каждого** из случаев
  - Для одного ядра
    - N = 8192
    - N = 16384
    - N = 32768
    - N = 65 536
  - Для N ядер
    - N = 8192
    - N = 16384
    - N = 32768
    - N = 65 536
  -
- среди 32 запусков (каждого варианта) необходимо найти и зафиксировать медиану значения времени выполнения.

### Измерение времени выполнения на аппаратной реализации

- Используются исходные коды функции lab7\_z1.cpp (**solution\_2**)
- следует осуществить синтез для случаев
  - N = 8192
  - N = 16384
  - N = 32768
  - N = 65 536

и для каждого случая зафиксировать: II, Estimated period, время выполнения = II \* Estimated period

### Сравнительный анализ

- Составить xls таблицу и построить два графика (
  - по оси X – случаи
    - N = 8192
    - N = 16384
    - N = 32768
    - N = 65 536
  - по Y – время выполнения функции на ПК и аппаратной реализации

**Отчет, должен включать**

- Задание
- Раздел с описанием исходного кода функции
- Раздел с описанием теста
- Раздел с описанием созданного командного файла
- Раздел с описанием результатов сравнения решений (со снимками экрана)
- Раздел с анализом результатов
  - Анализ и выбор оптимального (критерий максимальная производительность) решения
- Раздел с описанием модернизированного теста
  - Следует указать компилятор, используемый для компиляции.
- Результаты измерения **времени выполнения на ПК**
  - Следует указать: тип процессора, базовую частоту работы, максимальную частоту работы, объем ОЗУ.
- Результаты измерения времени выполнения на аппаратной реализации
- Раздел с анализом результатов
- Выводы

Архив должен включать всю рабочую папку проекта (включая модернизированный тест, xls таблицу и **скомпилированные приложения – папка ..\source**), отчет