

- Создать проект lab6_z1
- Микросхема: xa7a12tcsg325-1q
- Для всех решений задать: clock period 10 ; clock_uncertainty =1

- Создать на языке C++ функцию (N=32, din_type – short, dout_type - short),

```

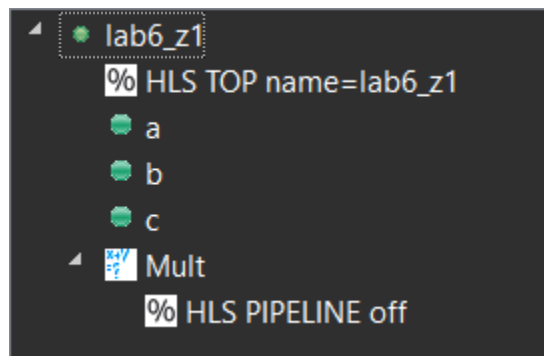
1  #include "lab6_z1.h"
2
3  void lab6_z1(dout_type a[N], din_type b[N], din_type c[N])
4  {
5      Mult:for (int i = 0; i < N; i++){
6          a[i] = b[i] * c[i];
7      }
8  }

```

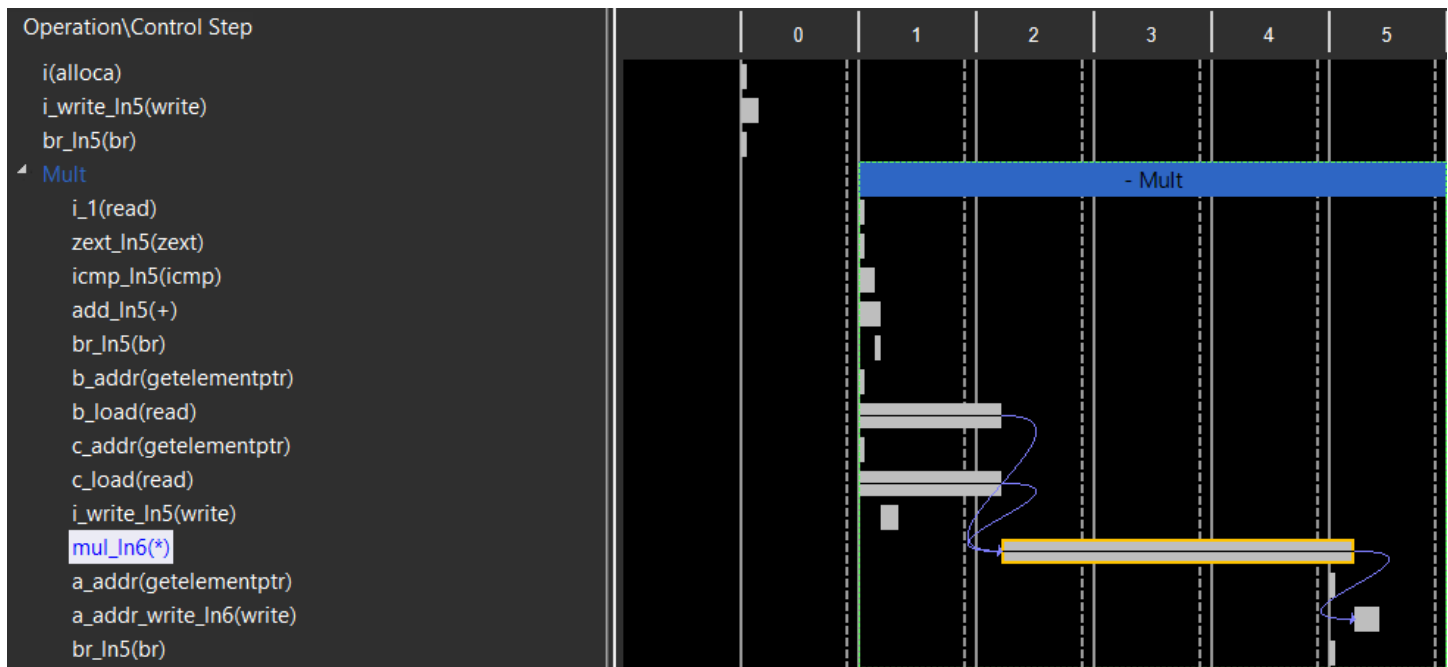
- Создать тест lab6_z1_test.cpp для проверки функции (не менее трех запусков функции) . Осуществить моделирование (с выводом результатов в консоль)

Исследование:

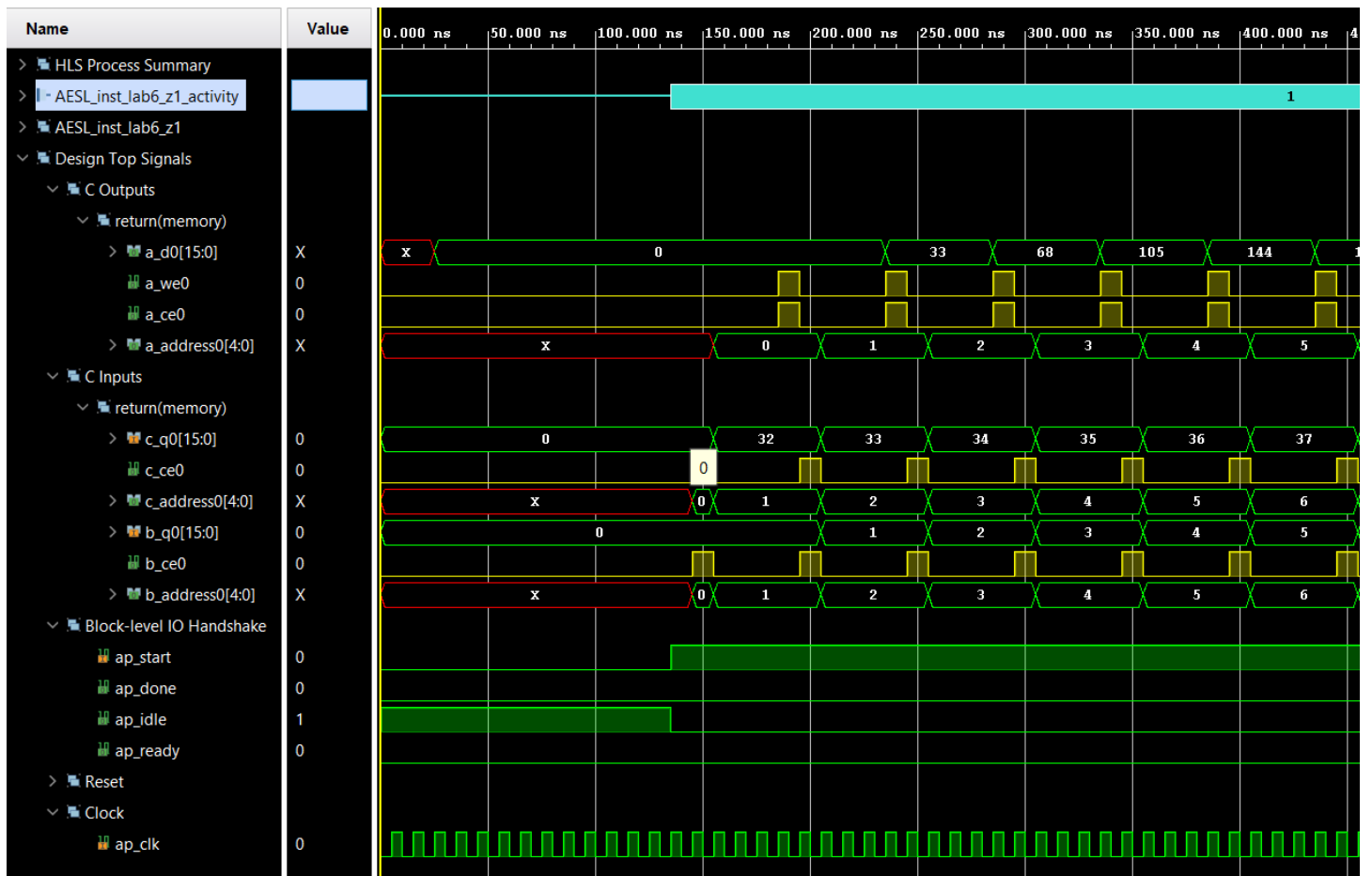
- Solution1
 - Выключите конвейеризацию для цикла



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

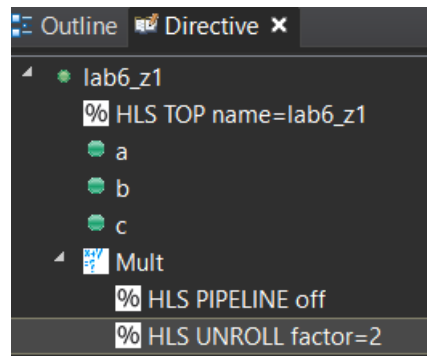


- Осуществите cosim. Приведите временную диаграмму (диаграмма должна быть похожа на приведенную ниже). Приведите ее анализ (что бы быть готовым ее пояснить).

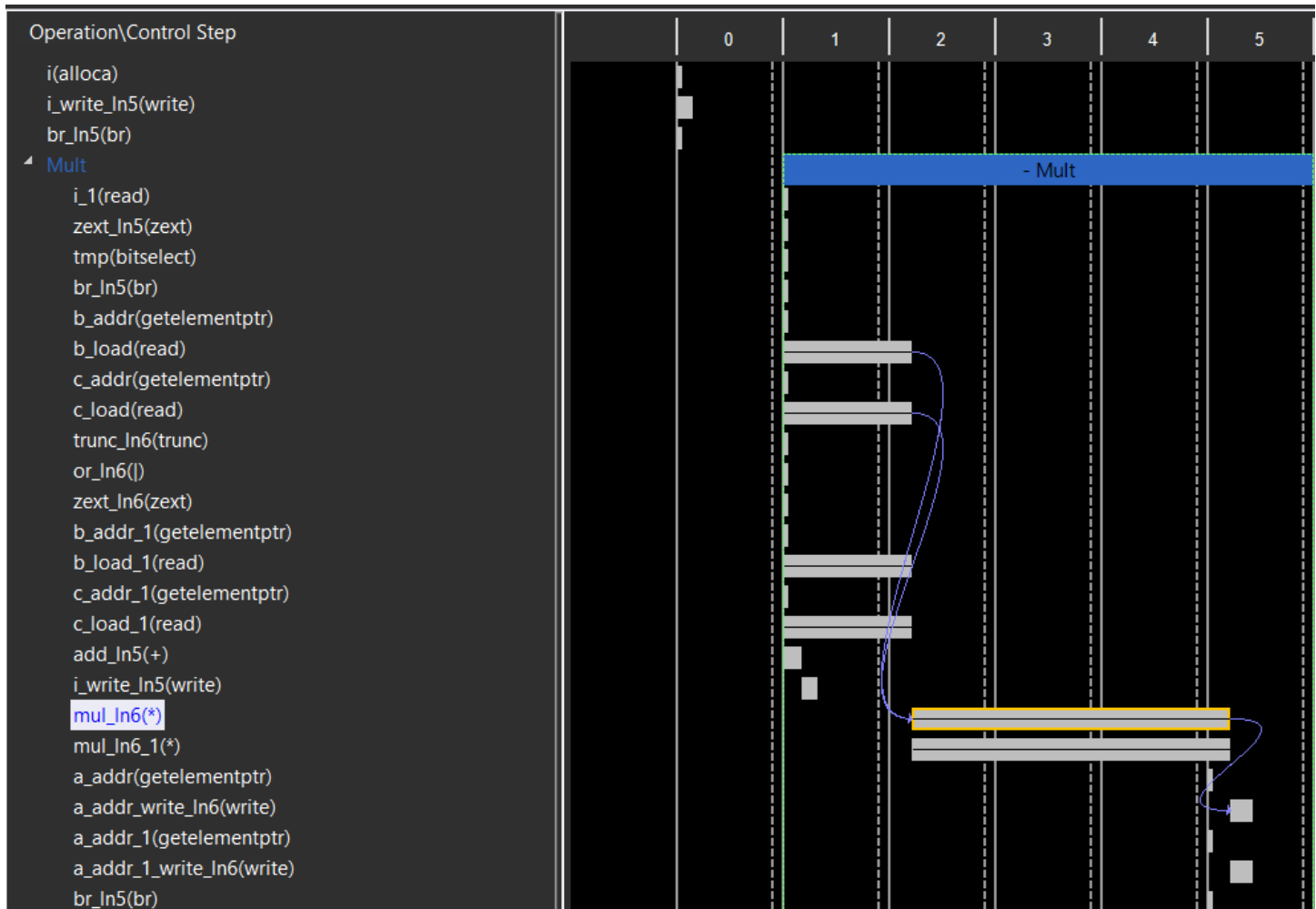


- Solution2

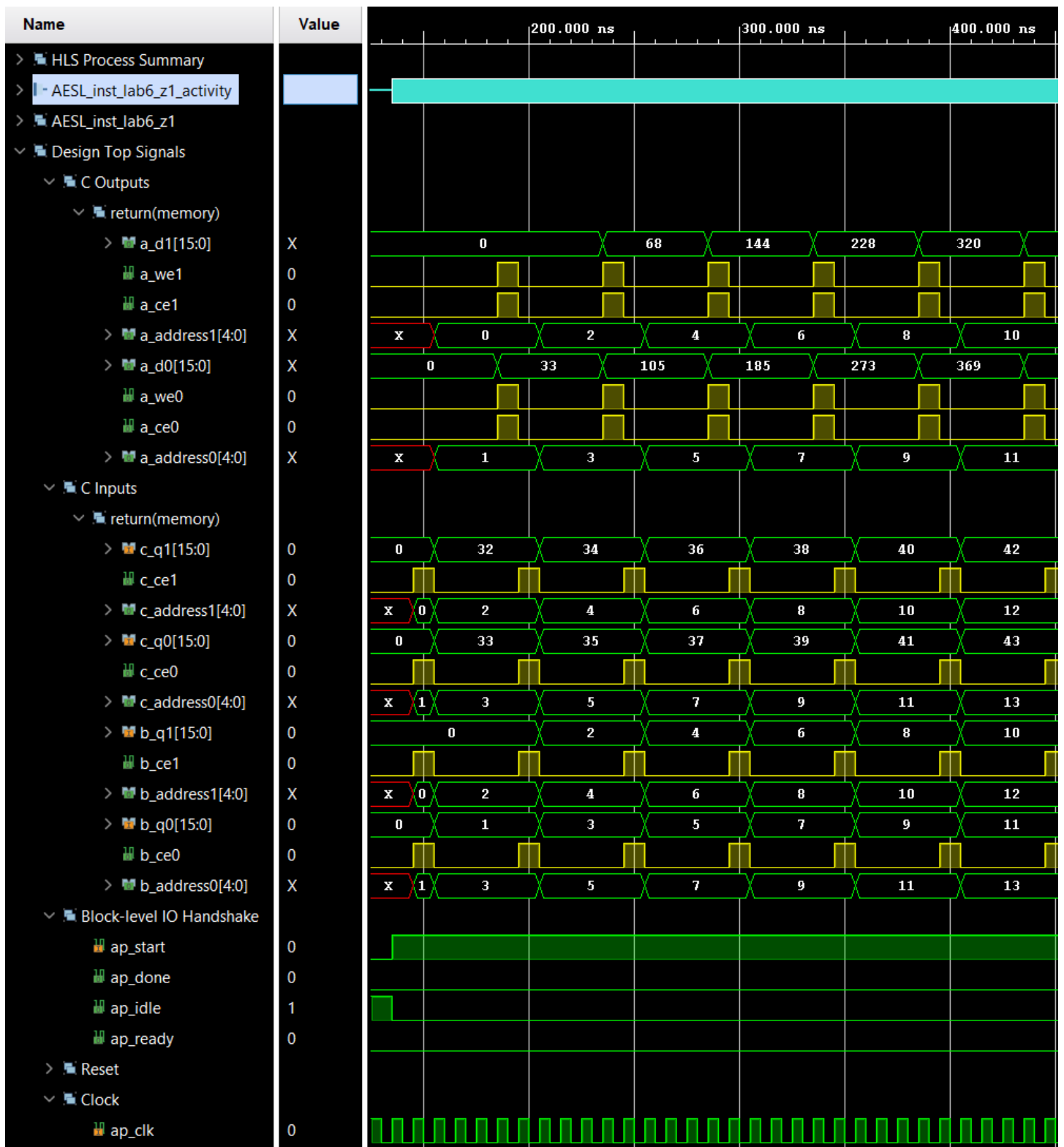
- установите: UNROLL FACTOR =2
- Выключите конвейеризацию для цикла



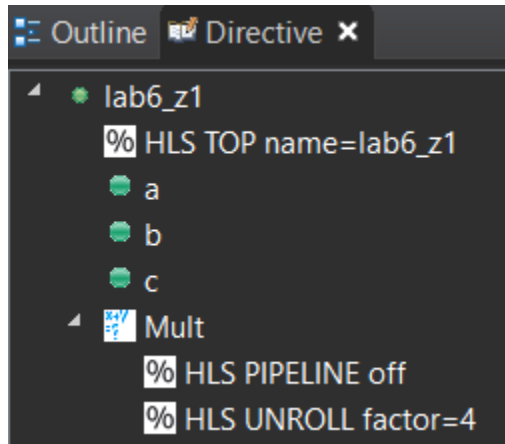
- осуществите синтез.
- Сравните результаты (schedule viewer, II, аппаратные затраты, реализацию памяти) с Solution1 и поясните отличия.



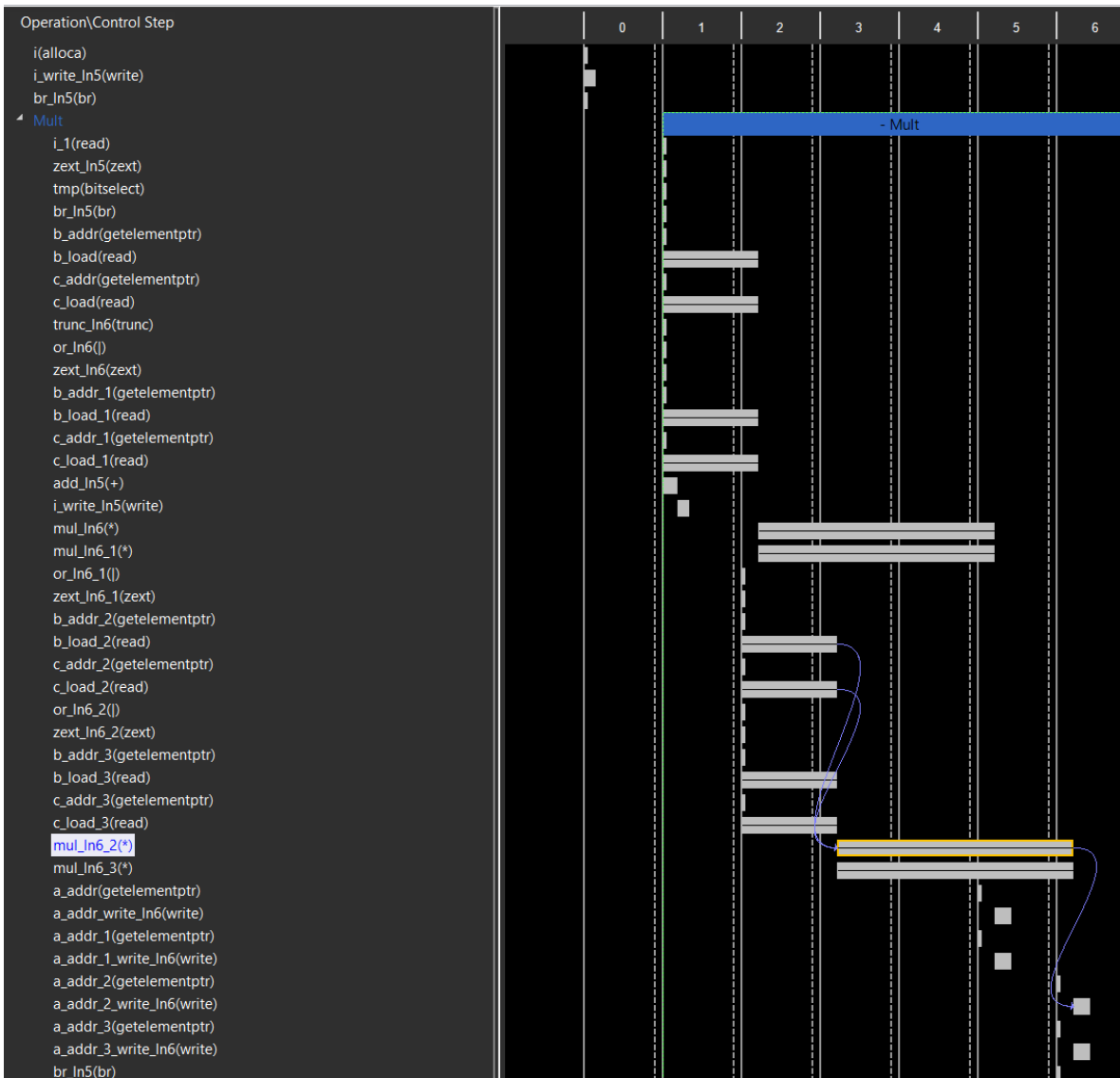
- Осуществите cosim. Приведите временную диаграмму (диаграмма должна быть похожа на приведенную ниже). Приведите ее анализ (что бы быть готовым ее пояснить).



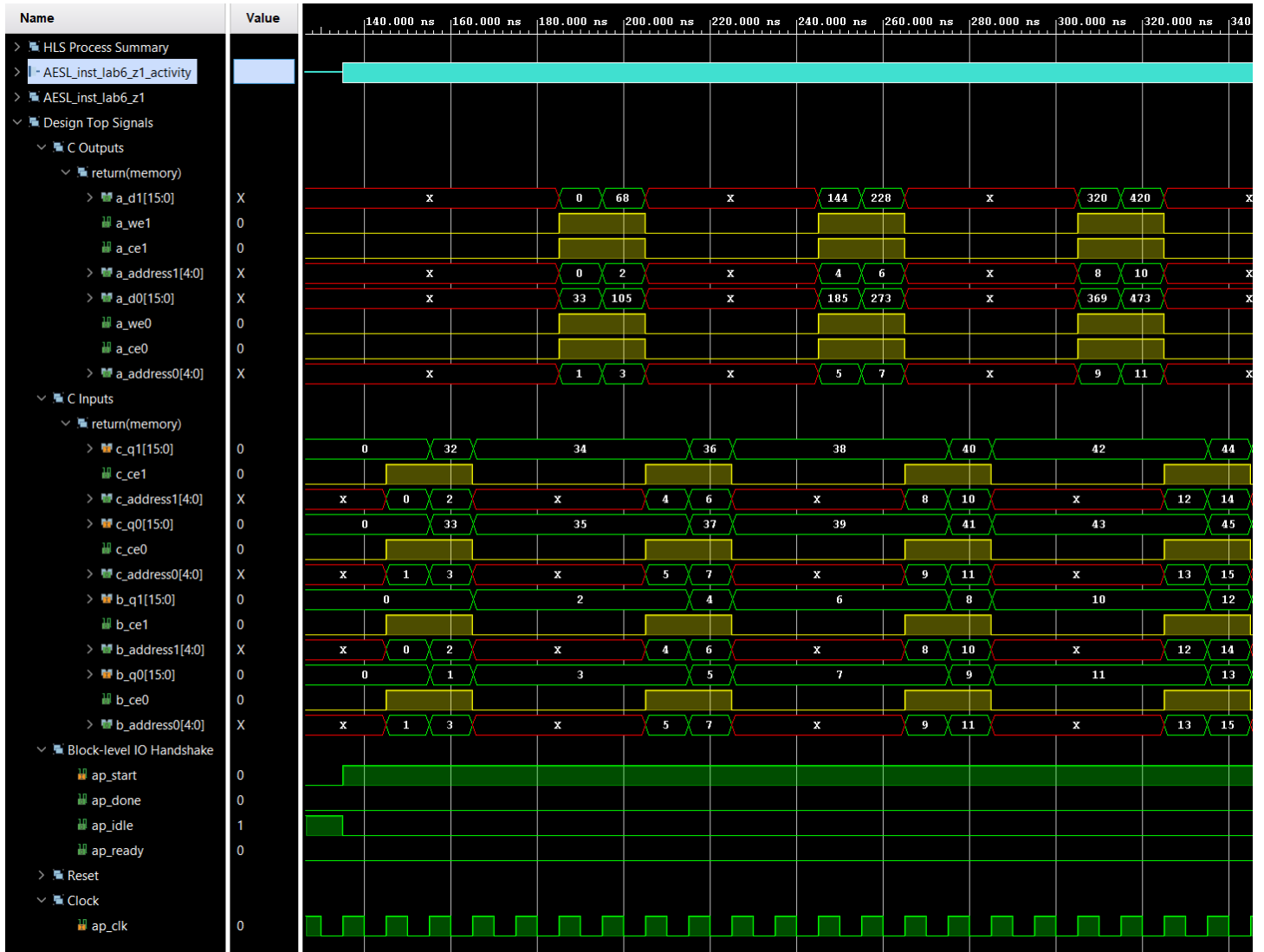
- Solution_3
 - установите: UNROLL FACTOR =4
 - Выключите конвейеризацию для цикла



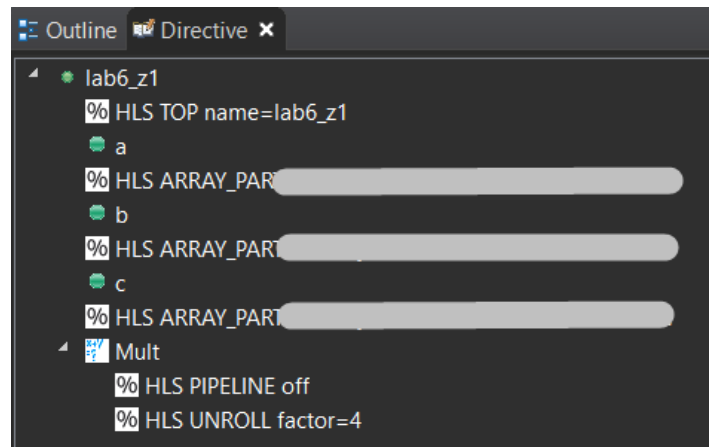
- осуществите синтез.
- Сравните результаты (schedule viewer, II, аппаратные затраты, реализацию памяти) с Solution1, 2 и поясните отличия.



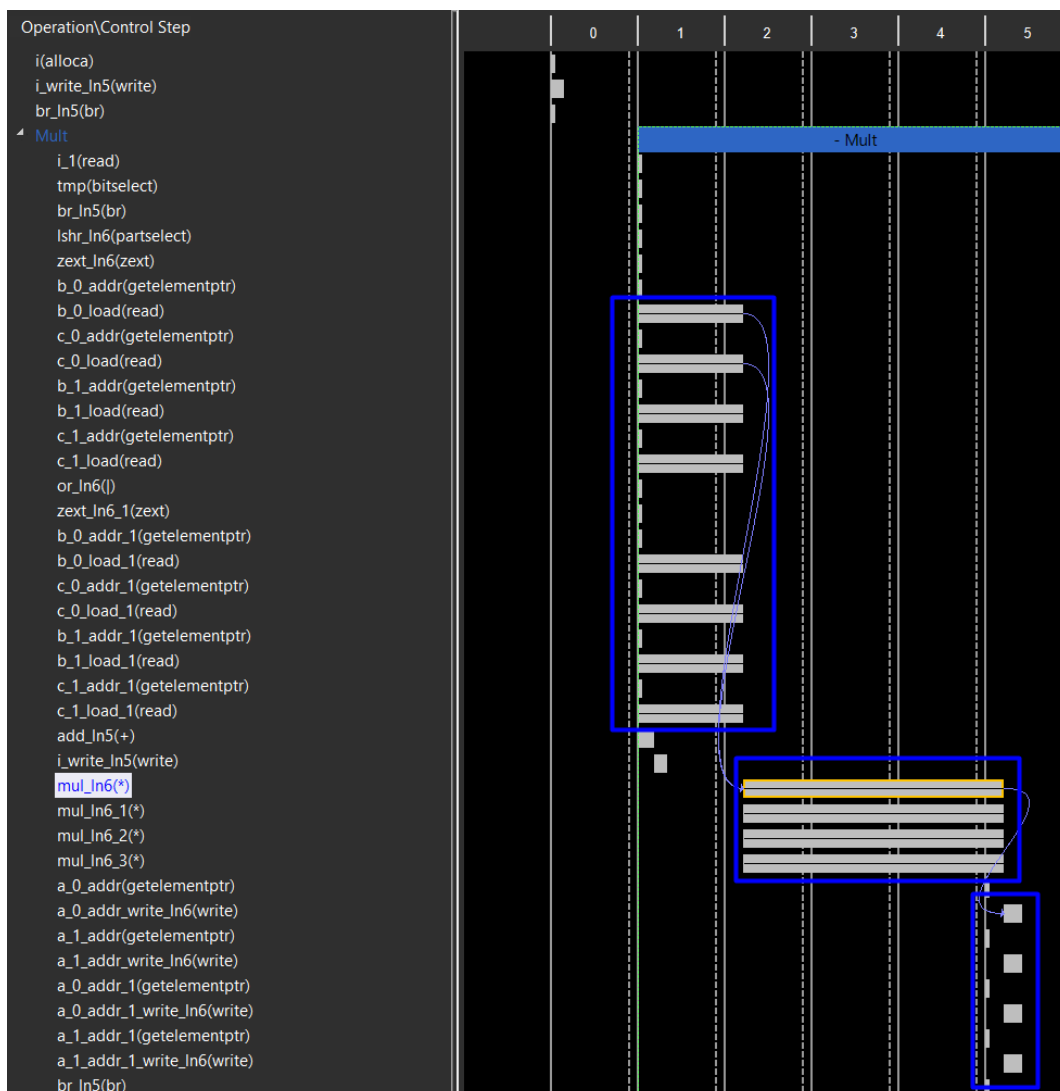
- Осуществите cosim. Приведите временную диаграмму (диаграмма должна быть похожа на приведенную ниже). Приведите ее анализ (что бы быть готовым ее пояснить).



- Solution_3_1
 - Создайте его на основе решения **solution_3**
 - К установкам решения solution_3 добавьте установки, обеспечивающие Schedule Viewer и временную диаграмму, показанные ниже.
 - подсказка

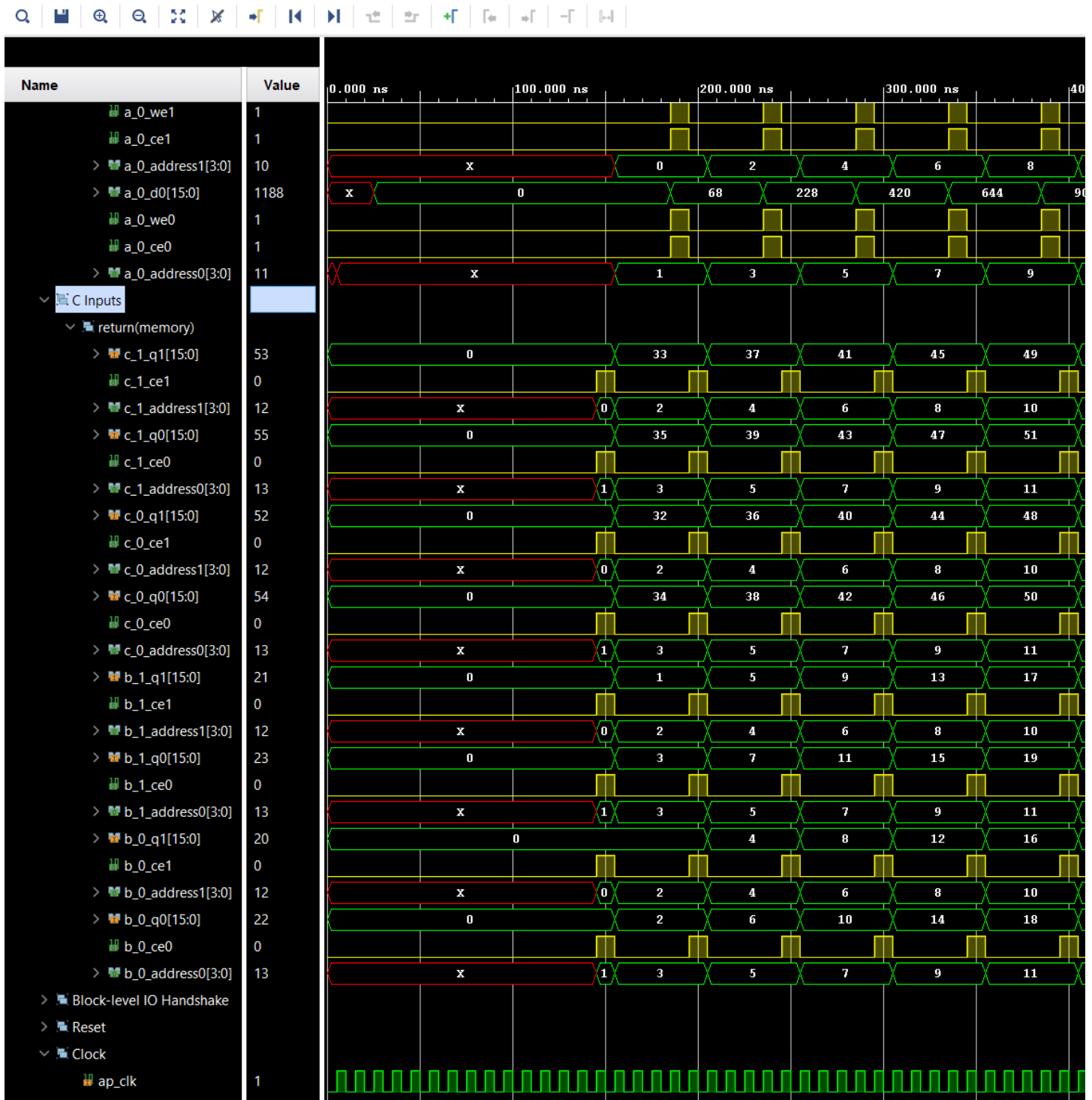


- осуществите синтез.
- Сравните результаты (schedule viewer, II, аппаратные затраты, реализацию памяти) с Solution1, 2, 3 и поясните отличия.

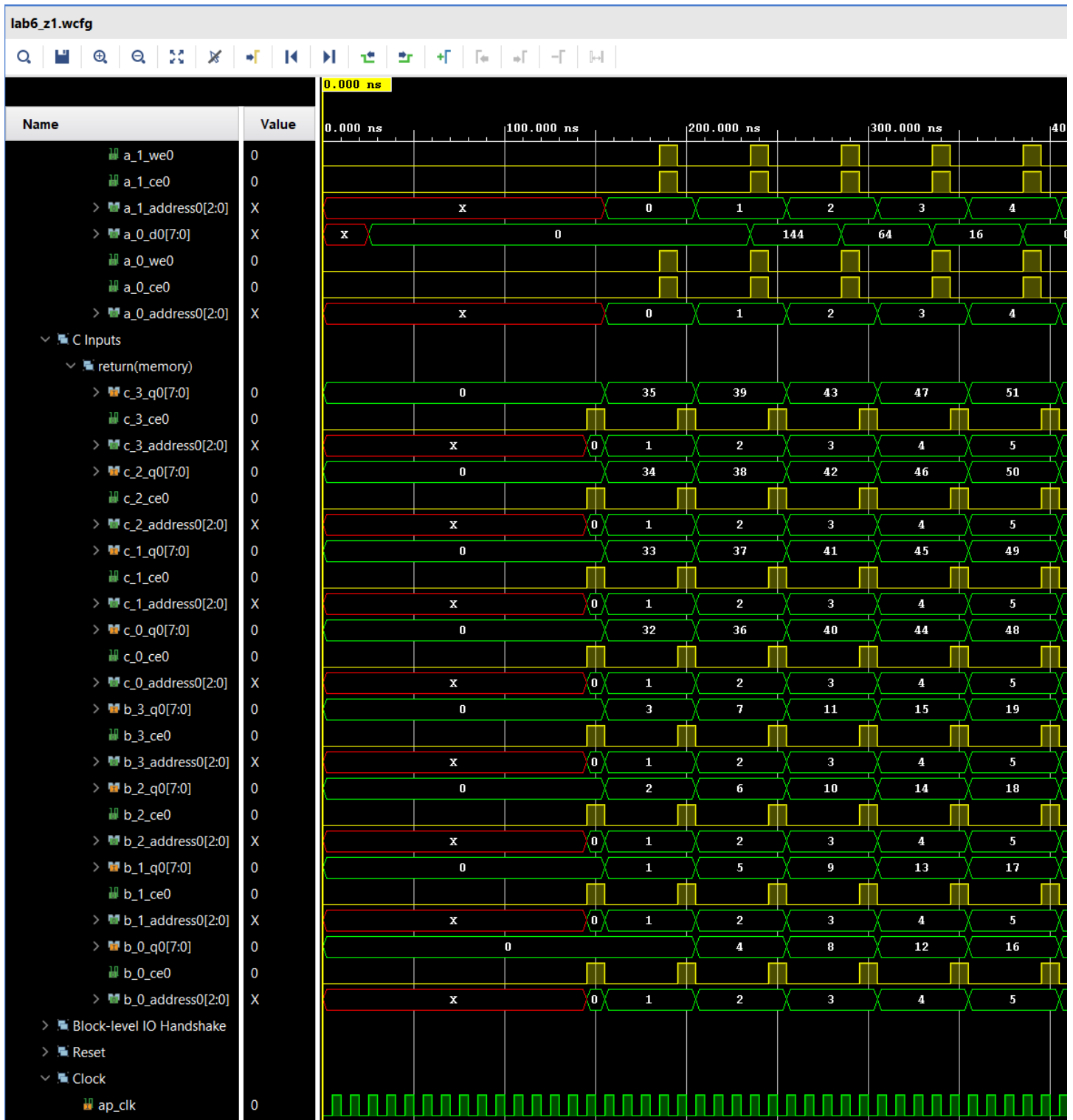


- Осуществите cosim. Приведите временную диаграмму (диаграмма должна быть похожа на одну, из приведенных ниже)
 - Вариант 1: (на рисунке показана только часть диаграммы – входные данные и часть формируемых выходных данных). Приведите ее анализ (что бы быть готовым ее пояснить) и сравнение с решением solution3.

lab6_z1.wcfg



- Вариант 2: (на рисунке показана только часть диаграммы – входные данные и часть формируемых выходных данных). Приведите ее анализ (что бы быть готовым ее пояснить) и сравнение с решением solution3.

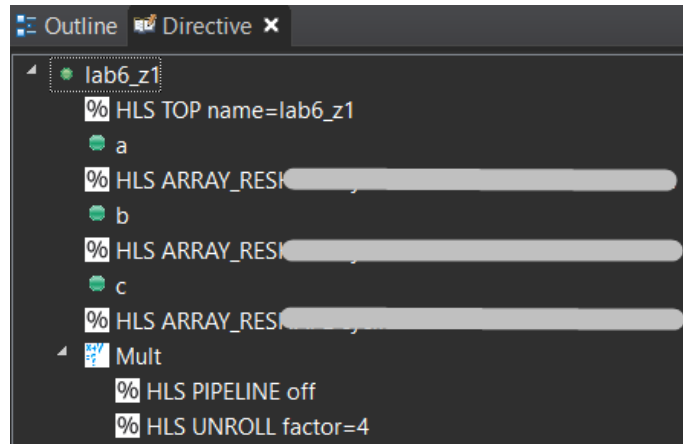


Вопросы:

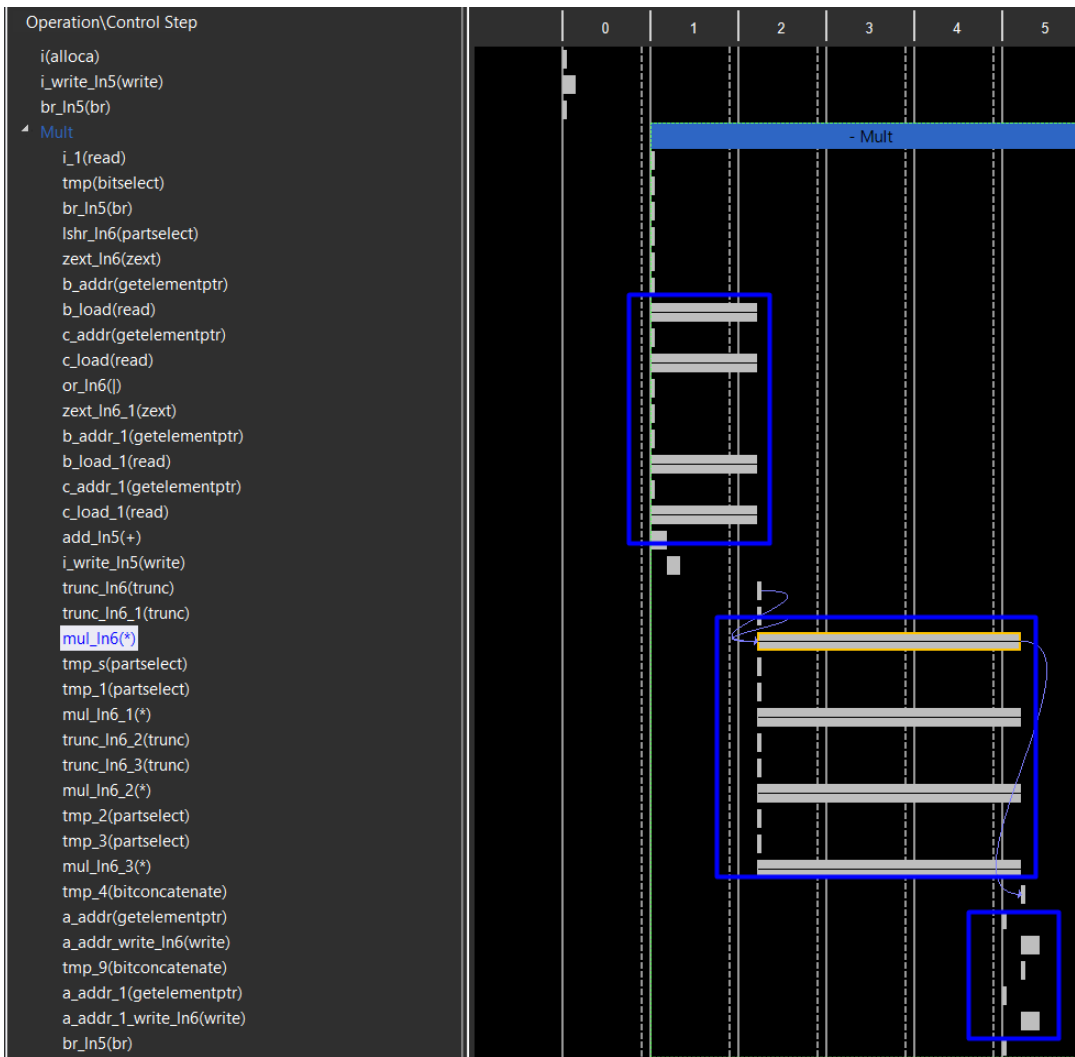
Почему при одинаковой Schedule Viewer возможны две разные временные диаграммы?

Чем отличаются диаграммы Варианта 1 и Варианта 2? Какие были установлены директивы для Варианта 1 и Варианта 2?

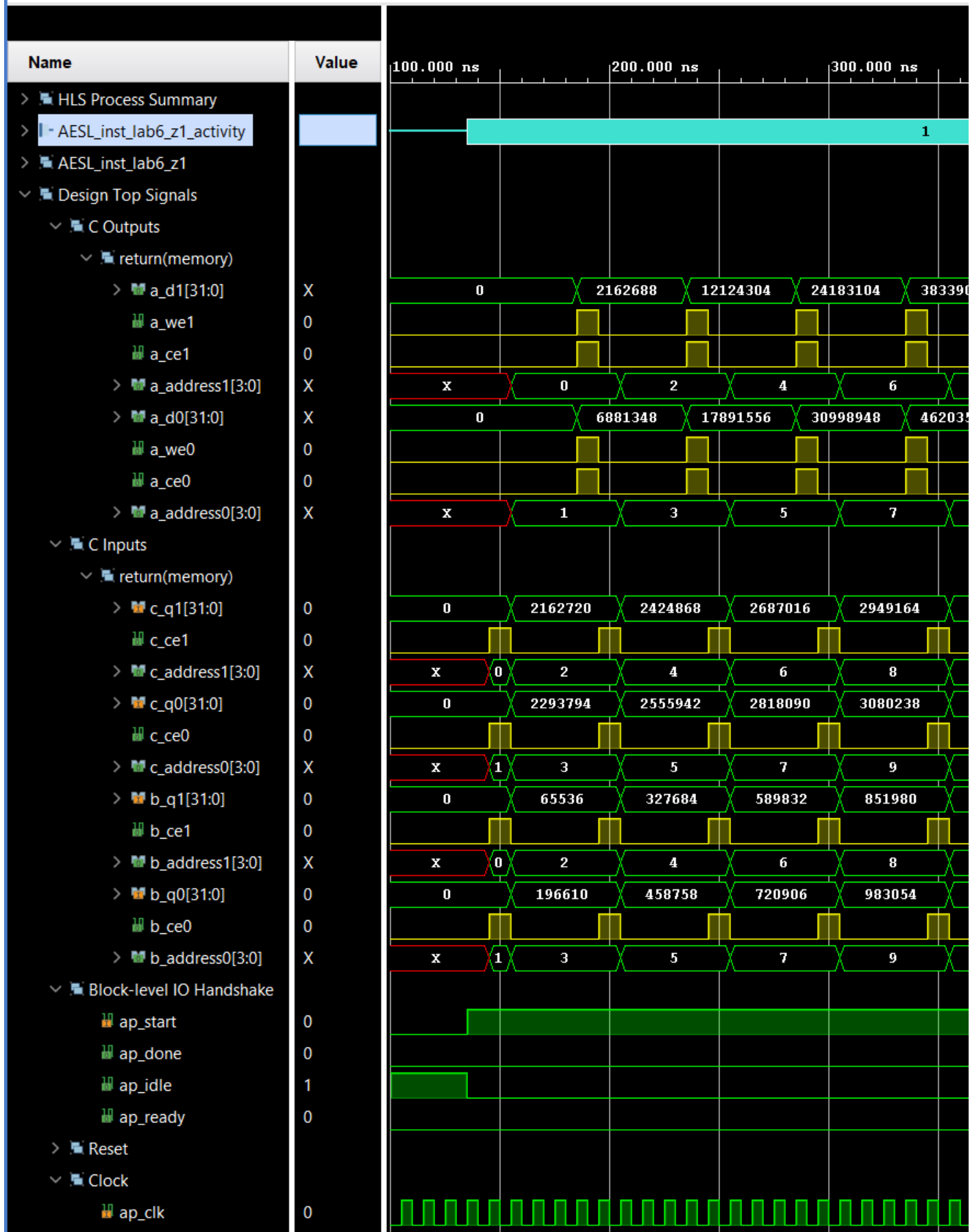
- Solution_3_2
 - Создайте его на основе решения **solution_3**
 - К установкам решения solution_3 добавьте установки, обеспечивающие Schedule Viewer и временную диаграмму, показанные ниже.
 - подсказка



- осуществите синтез.
- Сравните результаты (schedule viewer, II, аппаратные затраты, реализацию памяти) с Solution3_1 , поясните отличия.

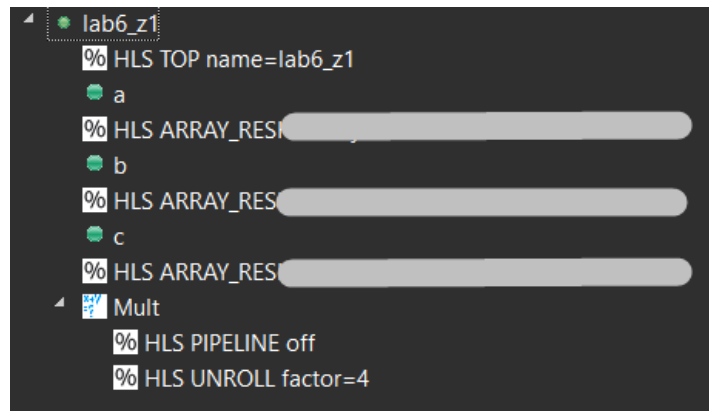


- Осуществите cosim. Приведите временную диаграмму (диаграмма должна быть похожа на приведенную ниже). Приведите ее анализ (что бы быть готовым ее пояснить) и сравнение с решением solution3_1.

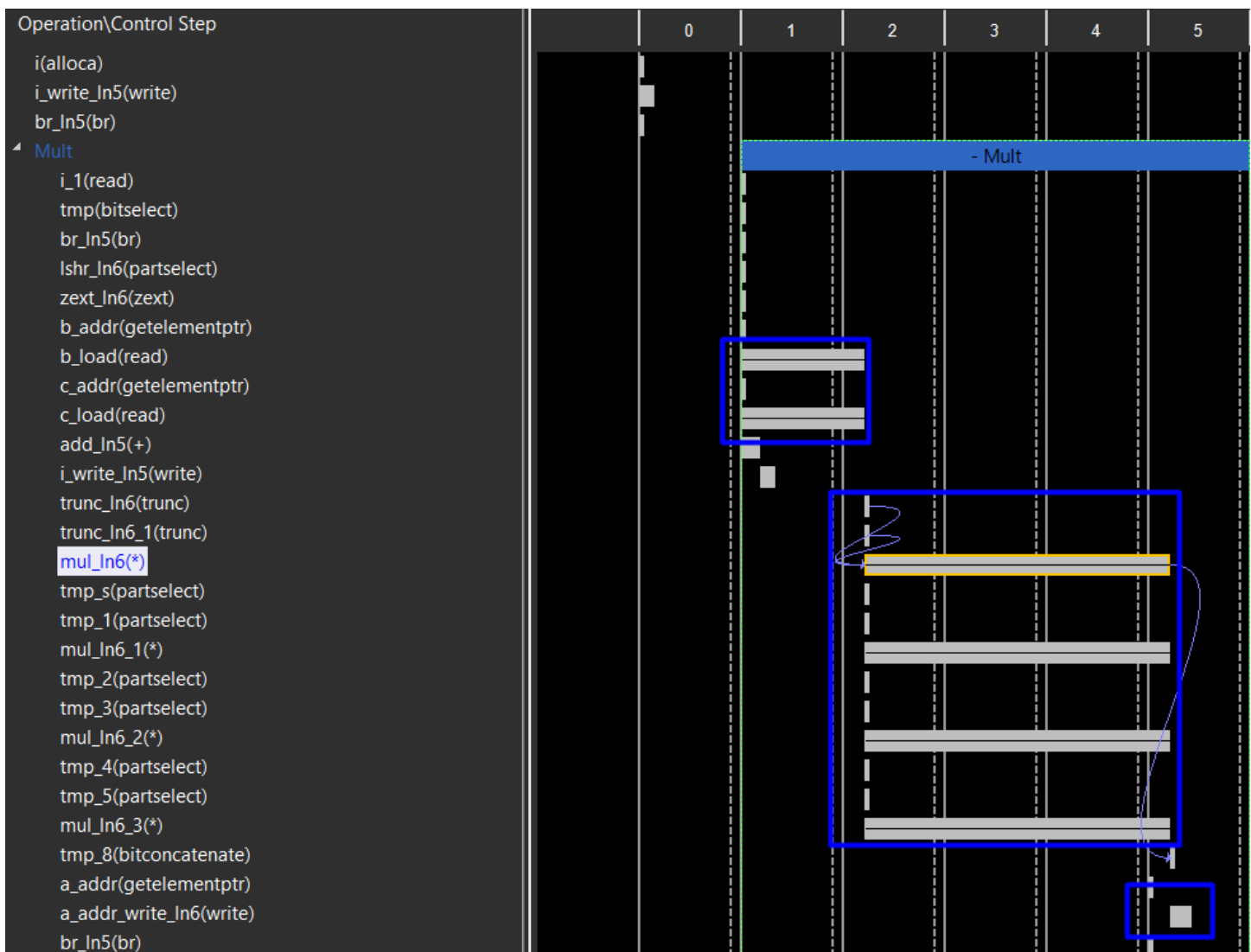


- Solution3_3

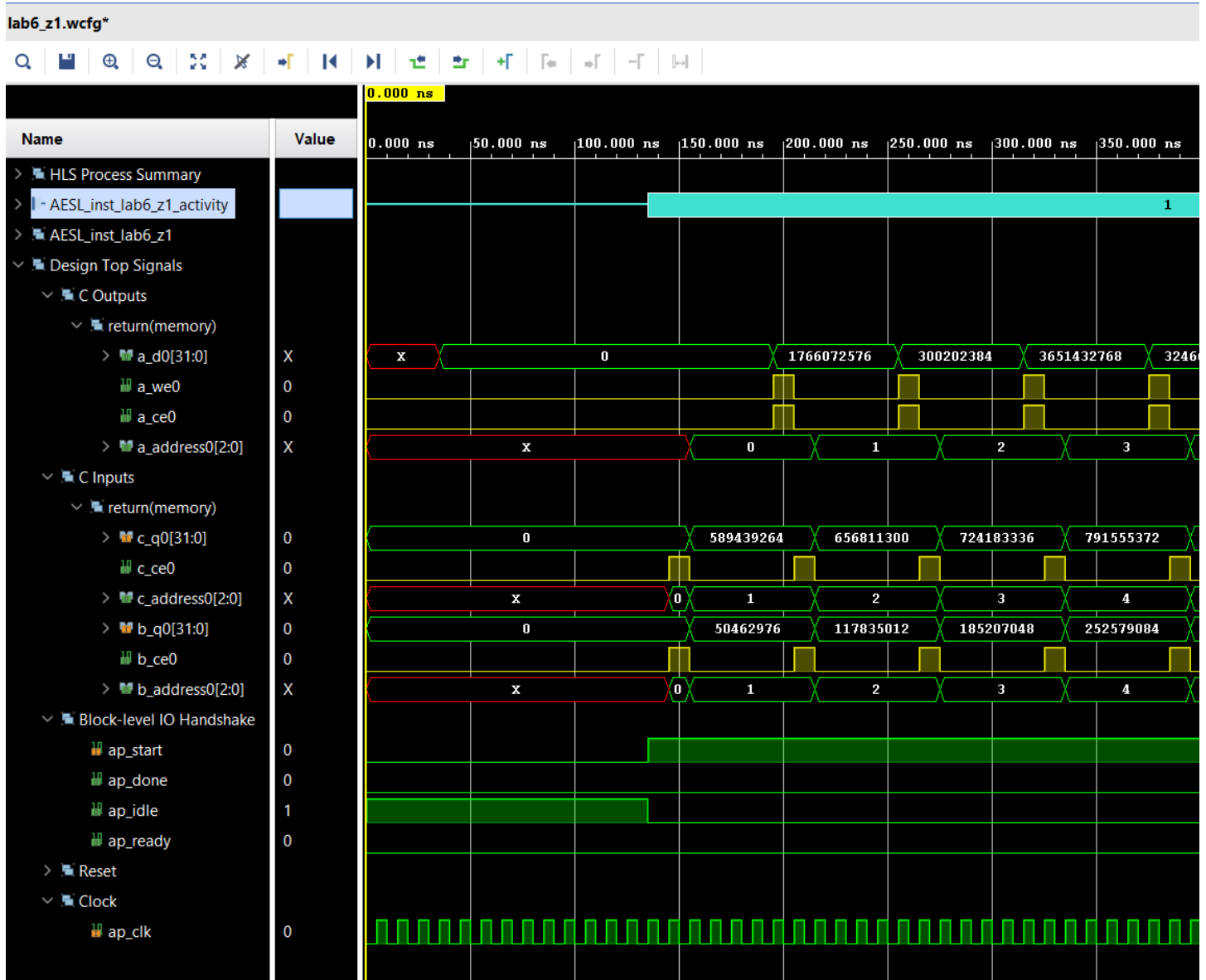
- Создайте его на основе решения **solution_3**
- К установкам решения solution_3 добавьте установки, обеспечивающие Schedule Viewer и временную диаграмму, показанные ниже.
 - подсказка



- осуществите синтез.
- Сравните результаты (schedule viewer, ll, аппаратные затраты, реализацию памяти) с Solution3_2.

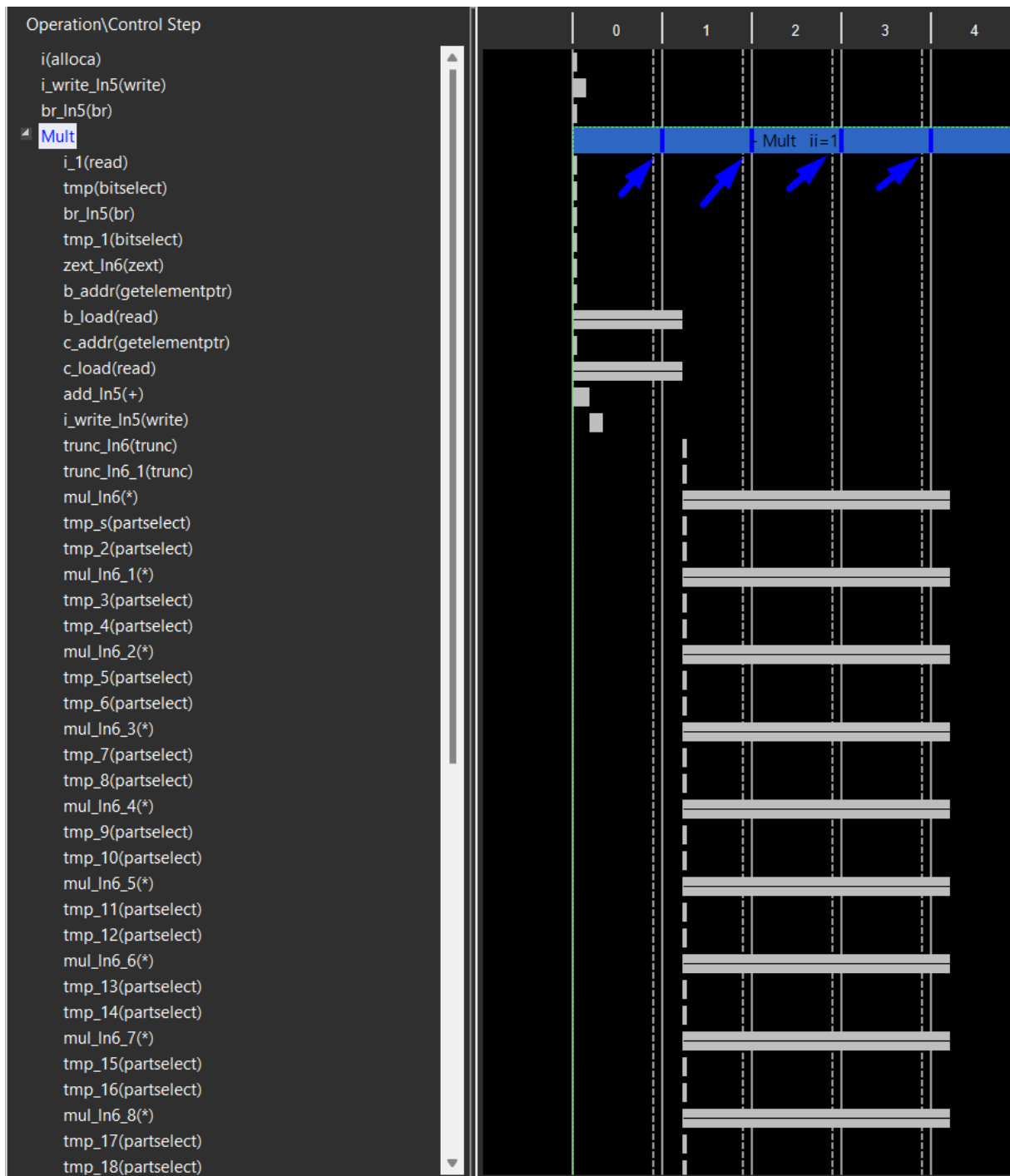


- Осуществите cosim. Приведите временную диаграмму (диаграмма должна быть похожа на приведенную ниже). Приведите ее анализ (что бы быть готовым ее пояснить) и сравнение с решением Solution3_2.

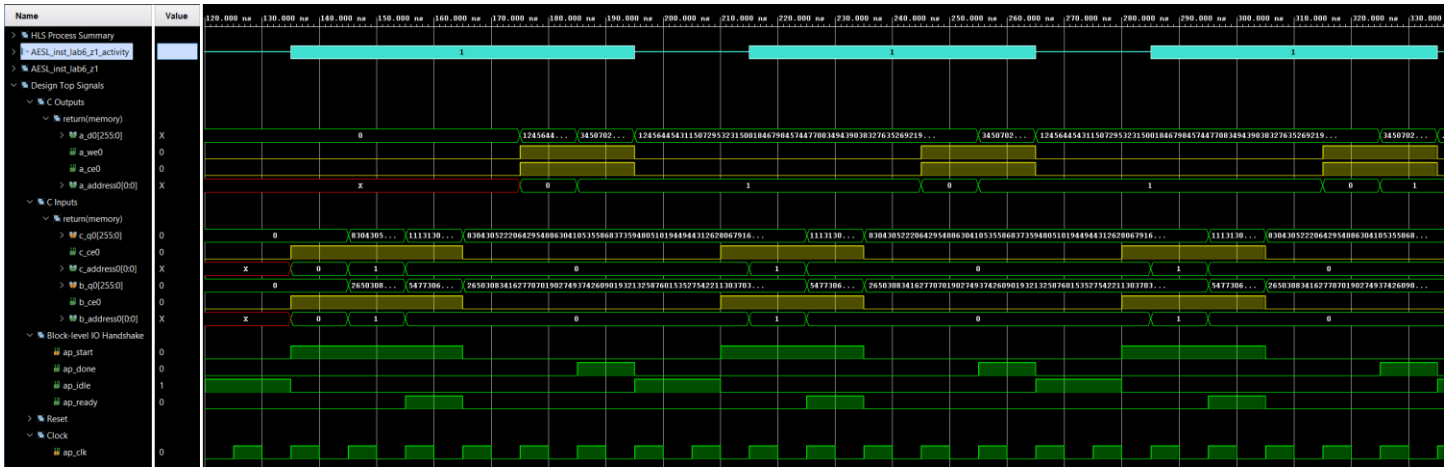


- Solution4

- Установите unroll, factor=16 для цикла.
- Сделайте установки такие, чтобы чтение всех необходимых данных осуществлялось за один такт, все 16 умножений выполнялись параллельно, запись данных выполнялась за один такт.
- **Включите конвейеризацию для цикла**
- осуществите синтез.
- Проверьте и зафиксируйте результаты (schedule viewer, II, аппаратные затраты, реализацию памяти).



- Осуществите cosim. Приведите временную диаграмму (диаграмма должна быть похожа на приведенную ниже). Приведите ее анализ (что бы быть готовым ее пояснить).



Измерение времени выполнения на ПК

- Используются исходные коды функции lab6_z1.cpp (**solution_4**)
- На базе теста **lab6_z1_test.cpp** **следует** создать отдельный, модернизированный, тест **lab6_z1_testSW.cpp** (сохранить в папке C:\Xilinx_trn\HLS2023\lab5_z1\source) для проверки времени выполнения функции lab5_z1 на ПК
- Следует осуществить компиляцию модернизированного теста и запускать его как отдельное приложение
- Следует провести измерение времени выполнения синтезируемой функции на Вашем ПК **для каждого** из случаев
 - N = 4096
 - N = 16384
 - N = 32768
- среди 32 запусков необходимо найти и зафиксировать максимальное, минимальное значения времени выполнения и медиану.

Измерение времени выполнения на аппаратной реализации

- Используются исходные коды функции lab6_z1.cpp (**solution_4**)
 - следует осуществить синтез для случаев
 - N = 4096
 - N = 16384
 - N = 32768
- 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 = 4096
 - N = 16384
 - N = 32768
 - по Y – время выполнения функции на ПК и аппаратной реализации

Исследовательское задание:

Увеличьте unroll factor так, чтобы максимально увеличить производительность учитывая возможности выбранной микросхемы (доступно максимум 40 умножителей), при этом поставьте все остальные директивы так, чтобы все использованные умножители работали параллельно. Проведите анализ быстродействия такого аппаратного решения и сравните с полученными ранее решениями.

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 и **скомпилированные приложения – папка ..\source**), отчет