Санкт-Петербургский Политехнический Университет Петра Великого

Институт Компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

Лабораторная работа 9 Задание 2

Предмет: «Проектирование реконфигурируемых гибридных вычислительных систем»

**Тема: «Оптимизация работы с массивами»**

Студент: Ерниязов Т.Е.

Гр. № 3540901/81502

Преподаватель: Антонов А.П.

Санкт-Петербург

2019

Оглавление

[1. Задание 7](#_Toc27862082)

[2. Часть 1. 16](#_Toc27862083)

[2.1. Исходный код 16](#_Toc27862084)

[2.2. Моделирование 17](#_Toc27862085)

[2.3. Решение 1а 17](#_Toc27862086)

[2.3.1. Директивы 17](#_Toc27862087)

[2.3.2. Синтез 17](#_Toc27862088)

[2.3.3. Использование ресурсов 18](#_Toc27862089)

[2.3.4. Анализ решения 20](#_Toc27862090)

[2.4. Решение а2 20](#_Toc27862091)

[2.4.1. Директивы 20](#_Toc27862092)

[2.4.2. Синтез 20](#_Toc27862093)

[2.4.3. Использование ресурсов 21](#_Toc27862094)

[2.4.4. Анализ решения 22](#_Toc27862095)

[2.5. Решение а3 22](#_Toc27862096)

[2.5.1. Директивы 22](#_Toc27862097)

[2.5.2. Синтез 23](#_Toc27862098)

[2.5.3. Использование ресурсов 23](#_Toc27862099)

[2.5.4. Анализ решения 25](#_Toc27862100)

[2.6. Решение а4 25](#_Toc27862101)

[2.6.1. Директивы 25](#_Toc27862102)

[2.6.2. Синтез 25](#_Toc27862103)

[2.6.3. Использование ресурсов 26](#_Toc27862104)

[2.6.4. Анализ решения 27](#_Toc27862105)

[2.7. Решение а5 27](#_Toc27862106)

[2.7.1. Директивы 27](#_Toc27862107)

[2.7.2. Синтез 28](#_Toc27862108)

[2.7.3. Использование ресурсов 28](#_Toc27862109)

[2.7.4. Анализ решения 30](#_Toc27862110)

[2.8. Решение а6 30](#_Toc27862111)

[2.8.1. Директивы 30](#_Toc27862112)

[2.8.2. Синтез 30](#_Toc27862113)

[2.8.3. Использование ресурсов 31](#_Toc27862114)

[2.8.4. Анализ решения 33](#_Toc27862115)

[2.9. Решение а7 33](#_Toc27862116)

[2.9.1. Директивы 33](#_Toc27862117)

[2.9.2. Синтез 33](#_Toc27862118)

[2.9.3. Использование ресурсов 34](#_Toc27862119)

[2.9.4. Анализ решения 35](#_Toc27862120)

[2.10. Выводы 36](#_Toc27862121)

[3. Часть 2 37](#_Toc27862122)

[3.1. Исходный код 37](#_Toc27862123)

[3.2. Моделирование 38](#_Toc27862124)

[3.3. Решение 1b 38](#_Toc27862125)

[3.3.1. Директивы 38](#_Toc27862126)

[3.3.2. Синтез 38](#_Toc27862127)

[3.3.3. Использование ресурсов 39](#_Toc27862128)

[3.3.4. Анализ решения 40](#_Toc27862129)

[3.4. Решение 2b 41](#_Toc27862130)

[3.4.1. Директивы 41](#_Toc27862131)

[3.4.2. Синтез 41](#_Toc27862132)

[3.4.3. Использование ресурсов 41](#_Toc27862133)

[3.4.4. Анализ решения 43](#_Toc27862134)

[3.5. Решение 3b 44](#_Toc27862135)

[3.5.1. Директивы 44](#_Toc27862136)

[3.5.2. Синтез 44](#_Toc27862137)

[3.5.3. Использование ресурсов 44](#_Toc27862138)

[3.5.4. Анализ решения 46](#_Toc27862139)

[3.6. Решение 4b 47](#_Toc27862140)

[3.6.1. Директивы 47](#_Toc27862141)

[3.6.2. Синтез 47](#_Toc27862142)

[3.6.3. Использование ресурсов 47](#_Toc27862143)

[3.6.4. Анализ решения 50](#_Toc27862144)

[3.7. Решение 5b 50](#_Toc27862145)

[3.7.1. Директивы 50](#_Toc27862146)

[3.7.2. Синтез 50](#_Toc27862147)

[3.7.3. Использование ресурсов 50](#_Toc27862148)

[3.7.4. Анализ решения 53](#_Toc27862149)

[3.8. Решение 6b 53](#_Toc27862150)

[3.8.1. Директивы 53](#_Toc27862151)

[3.8.2. Синтез 53](#_Toc27862152)

[3.8.3. Использование ресурсов 54](#_Toc27862153)

[3.8.4. Анализ решения 56](#_Toc27862154)

[3.9. Решение 7b 56](#_Toc27862155)

[3.9.1. Директивы 56](#_Toc27862156)

[3.9.2. Синтез 56](#_Toc27862157)

[3.9.3. Использование ресурсов 57](#_Toc27862158)

[3.9.4. Анализ решения 59](#_Toc27862159)

[3.10. Решение 8b 60](#_Toc27862160)

[3.10.1. Директивы 60](#_Toc27862161)

[3.10.2. Синтез 60](#_Toc27862162)

[3.10.3. Использование ресурсов 60](#_Toc27862163)

[3.10.4. Анализ решения 62](#_Toc27862164)

[3.11. Выводы 63](#_Toc27862165)

1. Задание

* Создать проект lab9\_2
* Микросхема: xa7a12tcsg325-1q

ЧАСТЬ 1

* Создать функцию

*foo\_a: входной массив short d\_in[N]; выходной массив short d\_out [N/4].*

*for (short i=0; i<N/4; i++){*

*d\_out[i] = d\_in[i]\*d\_in[i+8] + d\_in[i+4]\*d\_in[i+12];*

*}*

*N=16*

* Создать тест lab9\_2\_test.c для проверки функции. Осуществить моделирование (с выводом результатов в консоль)
* Исследование:
* Solution\_1а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию bram; RAM\_1P\_BRAM для входного (и выходного) массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Solution\_2а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного (и выходного) массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_1a и solution\_2a) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_3а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_2P для входного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_2a и solution\_3a) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_4а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного массива
  + установить array\_partition; block; factor =2 для входного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_3a и solution\_4a) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_5а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_4a и solution\_5a) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_6а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_2P для входного и выходного массивов
  + установить array\_partition; block; factor =2 для входного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_5a и solution\_6a) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_7а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_2P для входного и выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_6a и solution\_7a) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…

Сделать сводную таблицу (*S\_x/Latency/II – номер решения/Latency/II)*

|  |  |  |
| --- | --- | --- |
|  | RAM\_1P | RAM\_2P |
| Без block | *S\_x/Latency/II* |  |
| block; factor =2 |  |  |
| block; factor =4 |  |  |

ЧАСТЬ 2

* Создать функцию

*foo\_b: входной массив short d\_in[N]; выходной массив short d\_out [N].*

*for (short i=0; i<N/4; i++){*

*d\_out[i] = d\_in[i]\*d\_in[i+4];*

*d\_out[i+1]= d\_in[i+8]\*d\_in[i+12];*

*d\_out[i+2]= d\_in[i]\*d\_in[i+12];*

*d\_out[i+3]= d\_in[i+4]\*d\_in[i+8];*

*}*

*N=16*

* Solution\_1b
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и выходного массивов
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Solution\_2b
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_1b и solution\_2b) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_3b
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + установить array\_partition; cyclic; factor =2 для выходного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_2b и solution\_3b) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_4b
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + установить array\_partition; cyclic; factor =4 для выходного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_3b и solution\_4b) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_5b
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и RAM\_2P для выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + установить array\_partition; cyclic; factor =1 для выходного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_3b и solution\_5b) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_6b
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и RAM\_2P для выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + установить array\_partition; cyclic; factor =2 для выходного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_5b и solution\_6b) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…
* Solution\_7b
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и RAM\_2P для выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + установить array\_partition; cyclic; factor =4 для выходного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_6b и solution\_7b) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…

Сделать сводную таблицу (*S\_x/Latency/II – номер решения/Latency/II)*

|  |  |  |
| --- | --- | --- |
|  | RAM\_1P | RAM\_2P |
| Без cyclic | *S\_x/Latency/II* |  |
| cyclic; factor =2 |  |  |
| cyclic; factor =4 |  |  |

* Solution\_8b
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию ap\_memory; RAM\_1P для входного и выходного массивов
  + установить array\_partition; block; factor =4 для входного массива
  + установить array\_partition; complete для выходного массива
  + осуществить синтез для:
    - привести в отчете:
      * performance estimates=>summary (timing, latency)
      * utilization estimates=>summary
      * performance Profile
      * Resource 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
* Сравнить два решения (solution\_2b и solution\_8b) и сделать выводы: зависимость от типа интерфейса; объяснить количество использованных умножителей; объяснить (посчитать) число циклов Latency, II…

1. Часть 1.
   1. Исходный код

Исходный код функции:

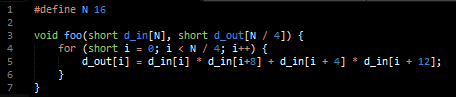


Рис. 2.1.1. source code

Исходный код теста:

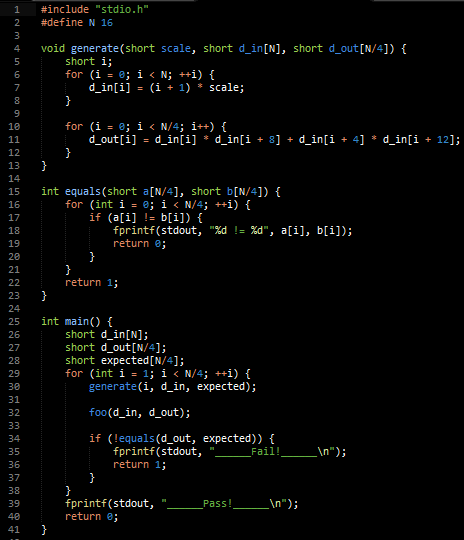


Рис. 2.1.2 test code

* 1. Моделирование

По результатам моделирование, приведённым ниже, видно, что устройство проходит тесты.

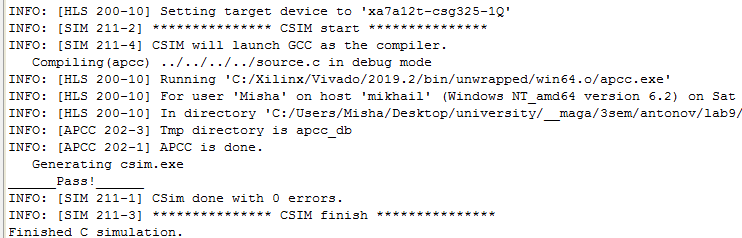


Рис. 2.2. Modeling results

* 1. Решение 1а
     1. Директивы

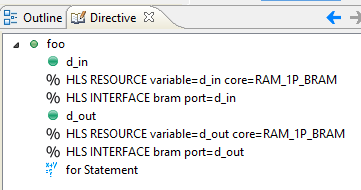


Рис. 2.3.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка **НЕ укладывается** в заданное значение.

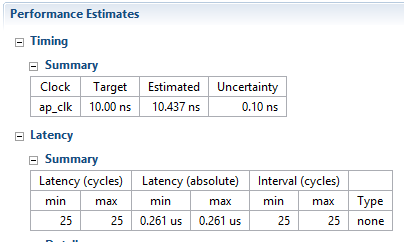


Рис. 2.3.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 2 блока DSP48E, 111 LUT и 67 триггеров.

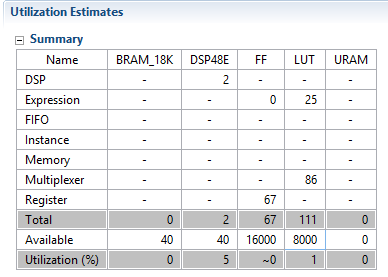


Рис. 2.3.3.1. Utilization estimates

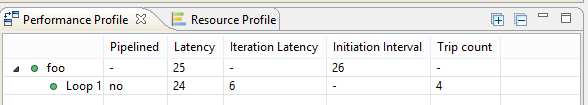


Рис. 2.3.3.2. Performance profile

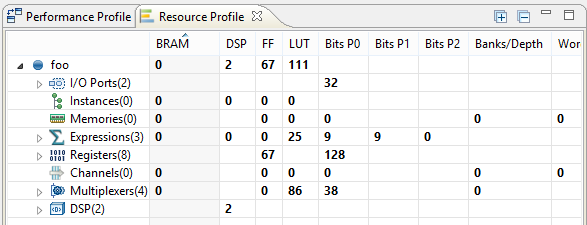


Рис. 2.3.3.3. Resource profile

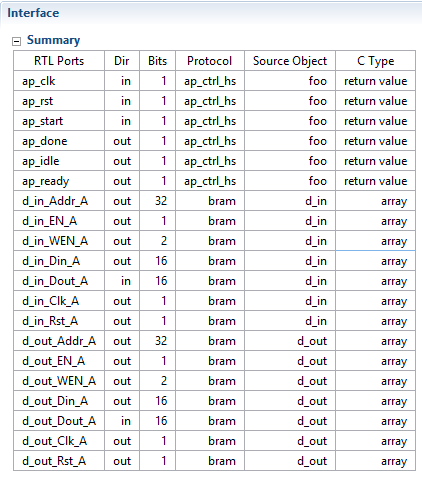


Рис. 2.3.3.4. Interfaces

Ниже, на рисунке видны блоки, которые не укладываются во временной интервал.

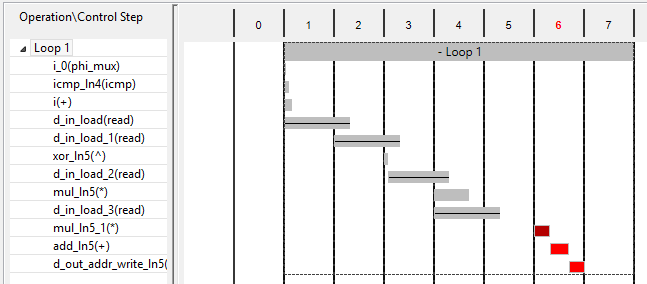


Рис. 2.3.3.5. Schedule viewer

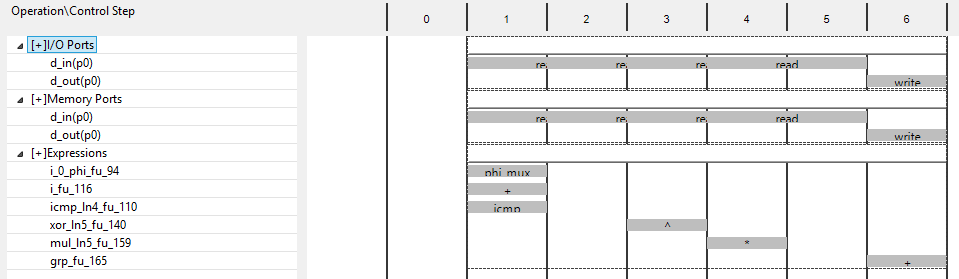


Рис. 2.3.3.5. Resource viewer

* + 1. Анализ решения

На каждой итерации для двух операций умножения в устройстве используется 2 умножителя. 25 тактов задержки – это 1 начальный такт инициализации и 4 итерации цикла, каждый из которых занимает 6 тактов. II совпадает с задержкой, так как устройство не конвейеризировано и работает последовательно. Устройство не уложилось в заданные 10 нс периода тактовой частоты.

* 1. Решение а2
     1. Директивы

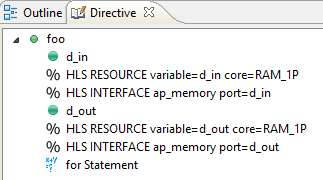


Рис.2.4.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

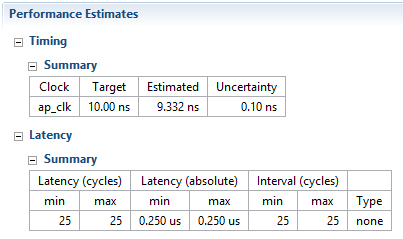


Рис. 2.4.2. Performance estimates

* + 1. Использование ресурсов

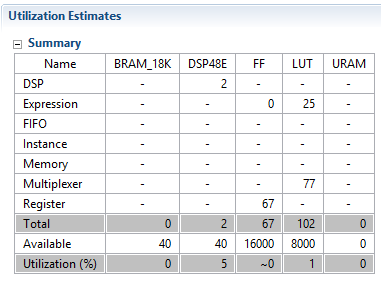


Рис. 2.4.3.1. Utilization estimates

Оценка использования ресурсов показывает, что будут использованы 2 блока DSP48E, 102 LUT и 67 триггеров.

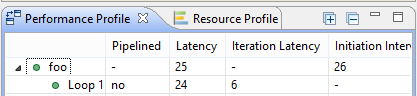


Рис. 2.4.3.2. Performance profile

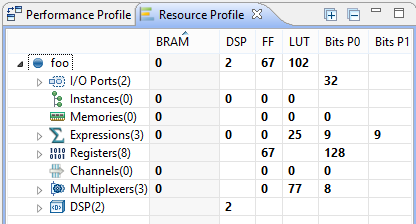


Рис. 2.4.3.3. Resource profile

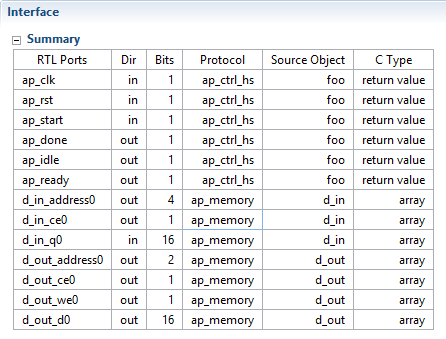


Рис. 2.4.3.4. Interfaces

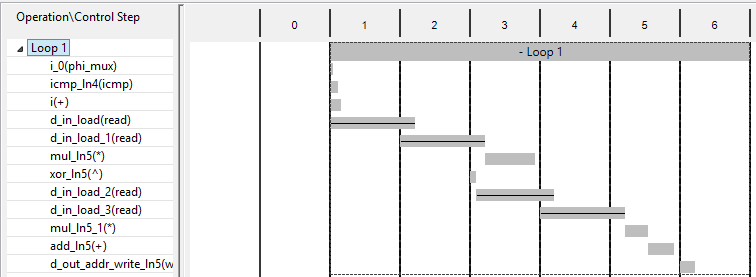


Рис. 2.4.3.5. Schedule viewer

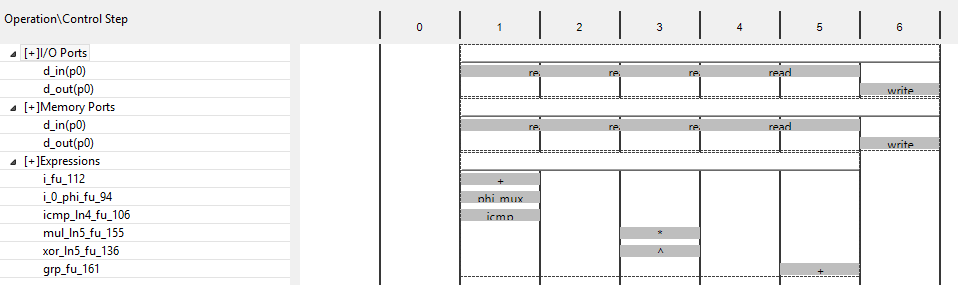


Рис. 2.4.3.6. Resource viewer

* + 1. Анализ решения

Данное решение укладывается во установленный временной интервал. Это связано с другими используемыми элементами памяти.

* 1. Решение а3
     1. Директивы

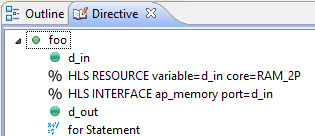


Рис. 2.5.1. Directives

* + 1. Синтез

Ниже приведены оценки производительности. По ним видно, что устройство соответствует заданным критериям.

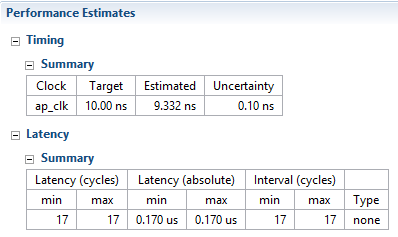


Рис. 2.5.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 2 блока DSP48E, 97 LUT и 46 триггеров.

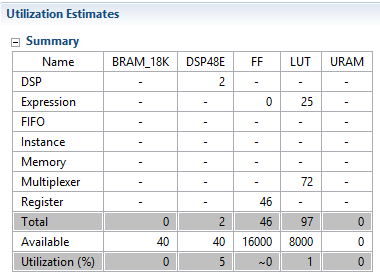


Рис. 2.5.3.1. Utilization estimates

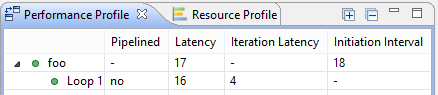


Рис. 2.5.3.2. Performance profile

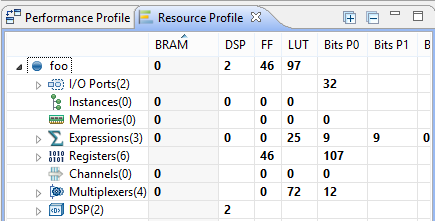


Рис. 2.5.3.3. Resource profile

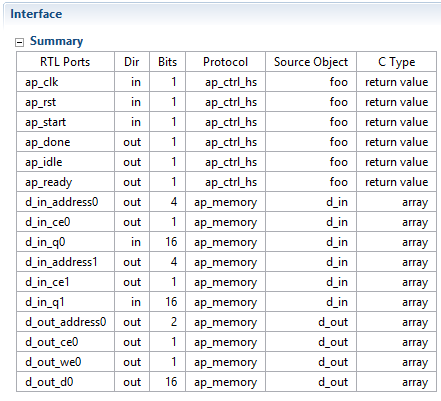


Рис. 2.5.3.4. Interfaces

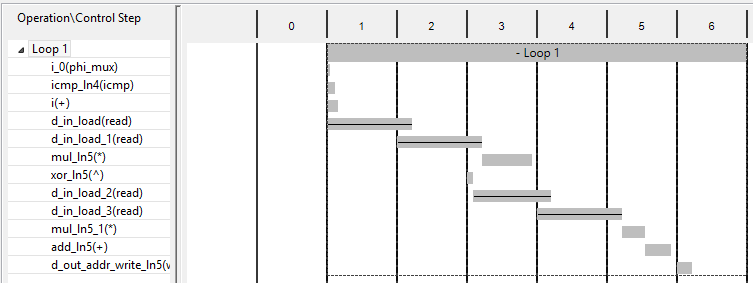


Рис. 2.5.3.5. Schedule viewer

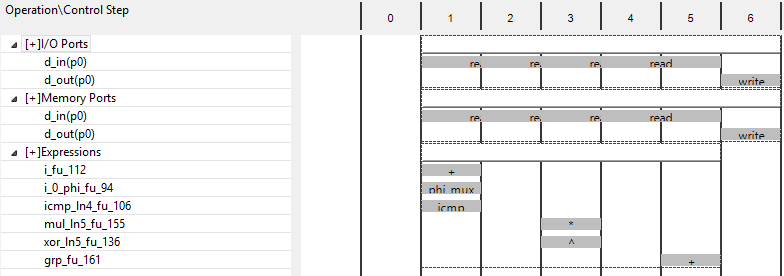


Рис. 2.5.3.6. Resource viewer

* + 1. Анализ решения

Данное решение быстрее и экономичнее относительно предыдущего. В нём также используется 2 умножителя, но благодаря двухпортовой памяти можно производить по 2 чтения из входного массива, что сокращает выполнение 1 итерации с 6 до 4 тактов. Следовательно, latency и II меньше на 8 тактов ( -2 такта на 4х итерациях).

* 1. Решение а4
     1. Директивы

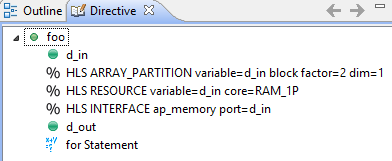


Рис. 2.6.1. Directives

* + 1. Синтез

Ниже приведены оценки производительности. По ним видно, что устройство соответствует заданным критериям.

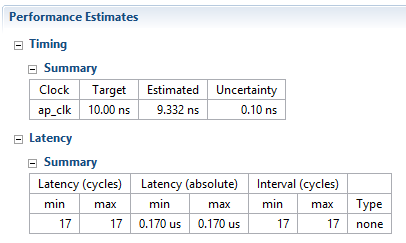


Рис. 2.6.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 2 блока DSP48E, 97 LUT и 62 триггера.

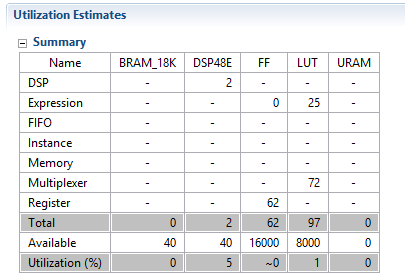


Рис. 2.6.3.1. Utilization estimates

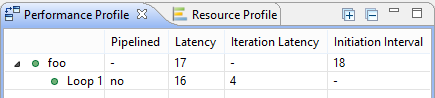


Рис. 2.6.3.2. Performance profile

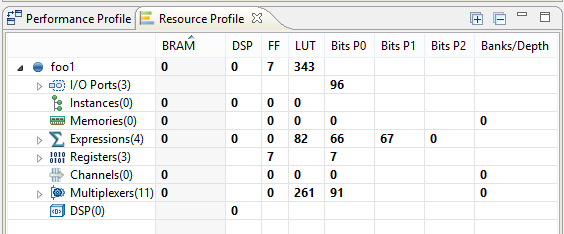


Рис. 2.6.3.3. Resource profile

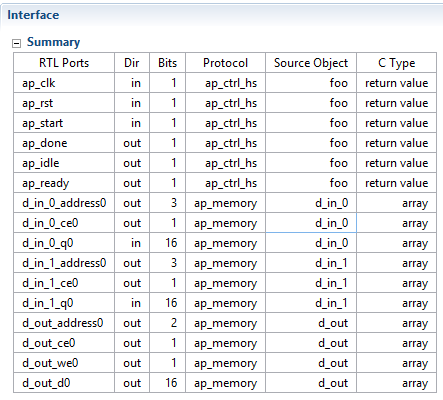


Рис. 2.6.3.4. Interfaces



Рис. 2.6.3.5. Schedule viewer

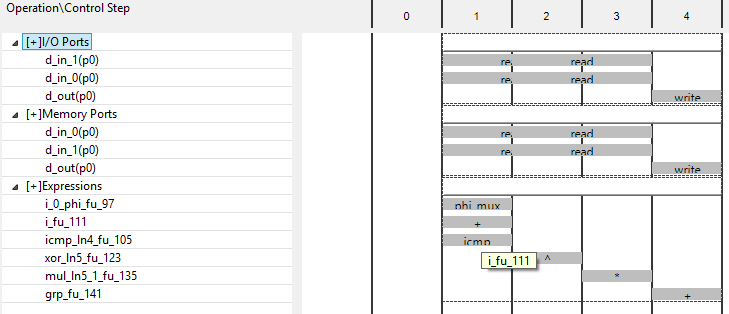


Рис. 2.6.3.6. Schedule viewer

* + 1. Анализ решения

В данном решение отсутствует двухпортовая память и используется 2 экземпляра однопортовой памяти. На результирующие характеристики устройства это не повлияло.

* 1. Решение а5
     1. Директивы

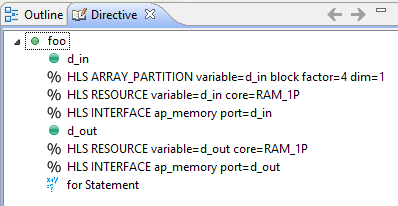


Рис. 2.7.1. Directives

* + 1. Синтез

Ниже приведены оценки производительности. По ним видно, что устройство соответствует заданным критериям.

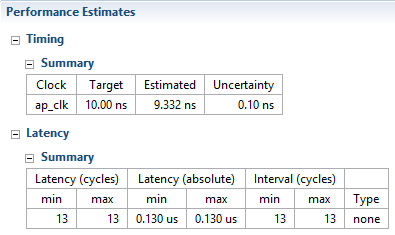


Рис. 2.7.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 2 блока DSP48E, 57 LUT и 61 триггер.

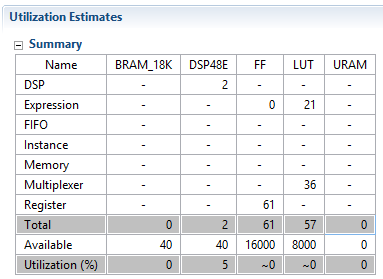


Рис. 2.7.3.1. Utilization estimates

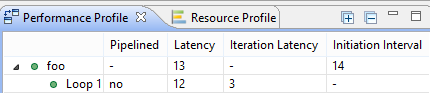


Рис. 2.7.3.2. Performance profile

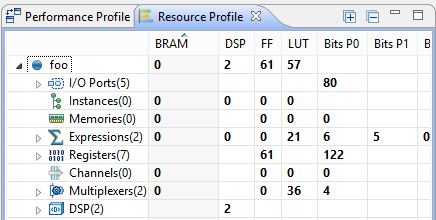


Рис. 2.7.3.3. Resource profile

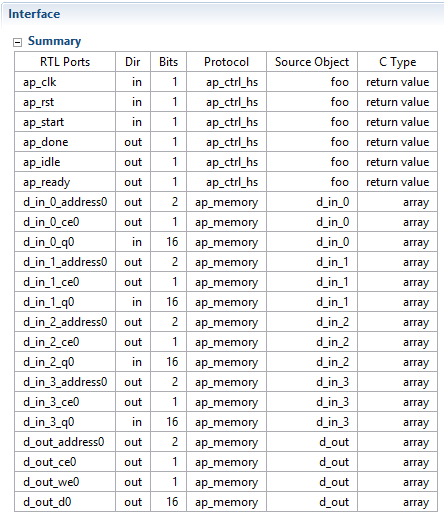


Рис. 2.7.3.4. Interfaces

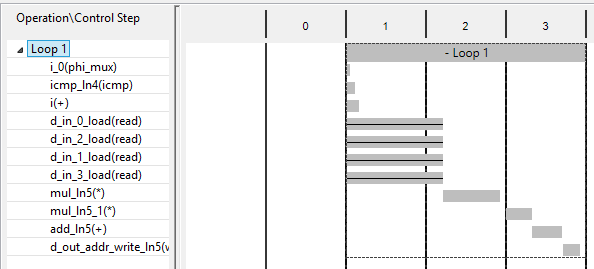


Рис. 2.7.3.5. Schedule viewer

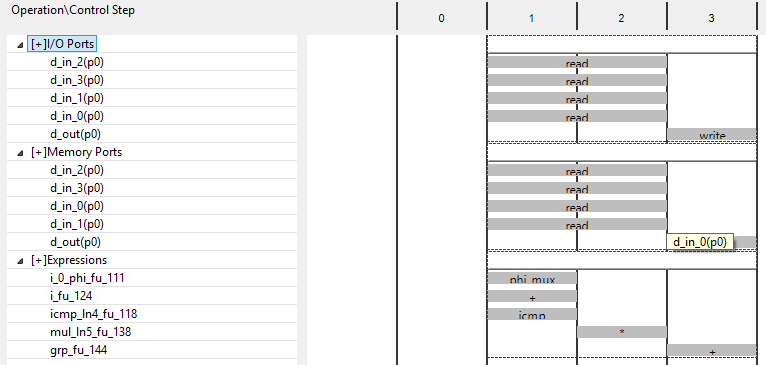


Рис. 2.7.3.6. Schedule viewer

* + 1. Анализ решения

В данном решение latency и II сократились на 4 такта. Это можно объяснить, что используется 4 экземпляра однопортовой памяти и все 4 операции чтения проходят одновременно. Это сокращает выполнение одной итерации на 1, что и создает улучшение в задержке на 4 такта.

* 1. Решение а6
     1. Директивы

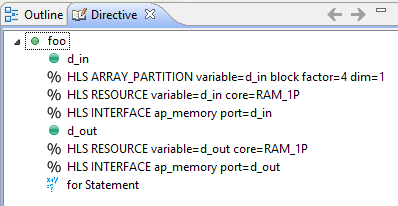


Рис. 2.8.1. Directives

* + 1. Синтез

Ниже приведены оценки производительности. По ним видно, что устройство соответствует заданным критериям.

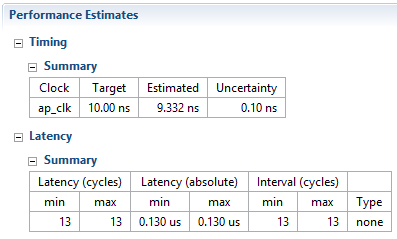


Рис. 2.8.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 2 блока DSP48E, 61 LUT и 61 триггер.

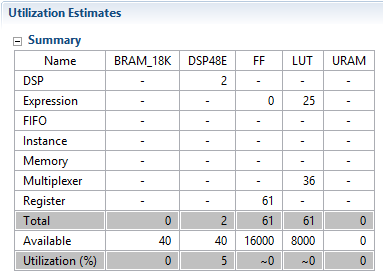


Рис. 2.8.3.1. Utilization estimates

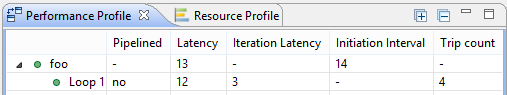


Рис. 2.8.3.2. Performance profile

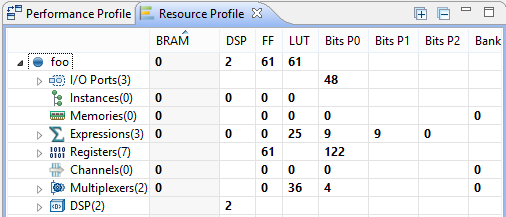


Рис. 2.8.3.3. Resource profile

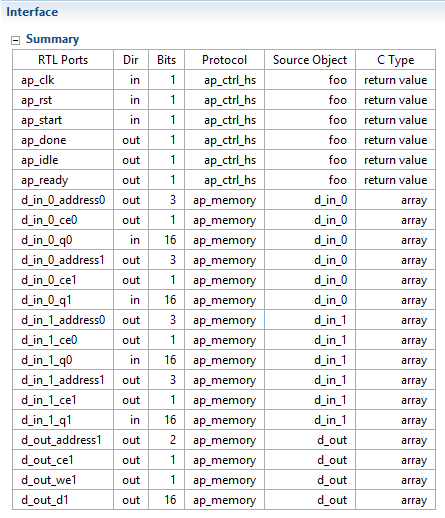


Рис. 2.8.3.4. Interfaces

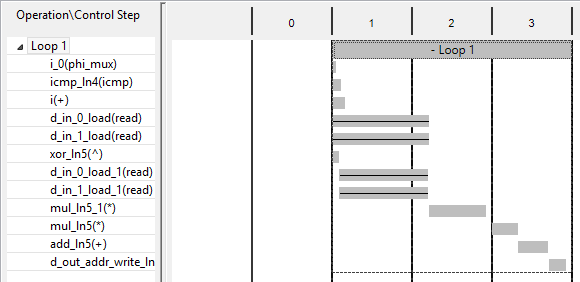


Рис. 2.8.3.5. Schedule viewer

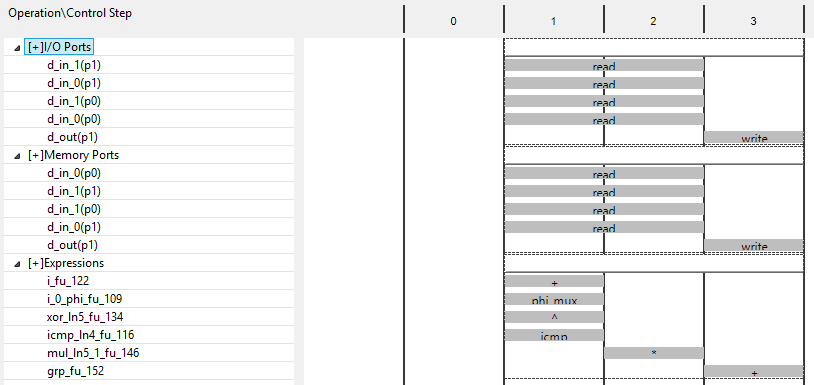


Рис. 2.8.3.6. Schedule viewer

* + 1. Анализ решения

В данном решение latency и II оказались одинаковыми. Использовать 4 однопортовых памяти или 2 двухпортовых оказалось практически одинаковыми. Разницу можно заметить только в том, что в решении 6а есть ещё одна небольшая операция получения новых индексов для чтения из памяти. Однако, это операция очень быстрая, и все 4 чтения происходят практически одновременно.

* 1. Решение а7
     1. Директивы

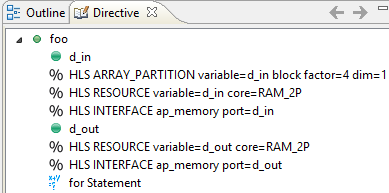


Рис. 2.9.1. Directives

* + 1. Синтез

Ниже приведены оценки производительности. По ним видно, что устройство соответствует заданным критериям.

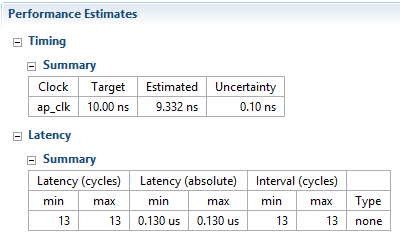


Рис. 2.9.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 2 блока DSP48E, 57 LUT и 61 триггер.

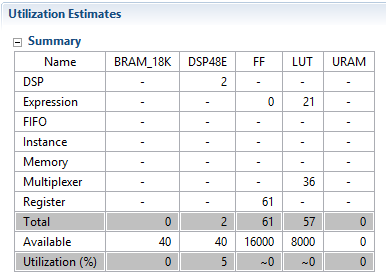


Рис. 2.9.3.1. Utilization estimates

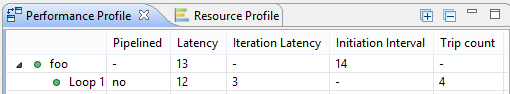


Рис. 2.9.3.2. Performance profile

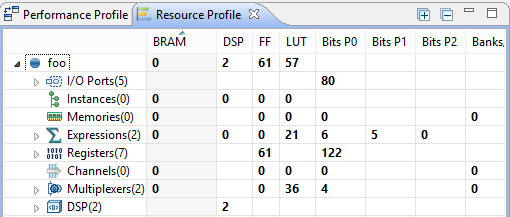


Рис. 2.9.3.3. Resource profile

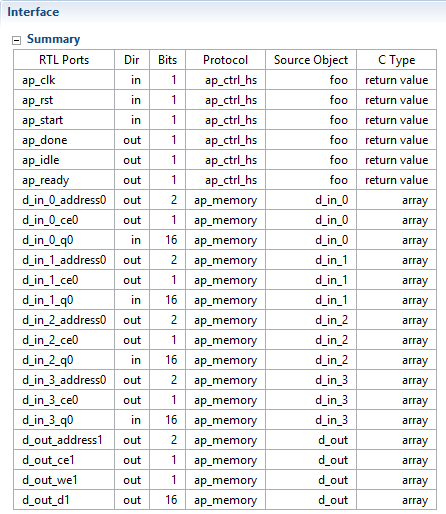


Рис. 2.9.3.4. Interfaces

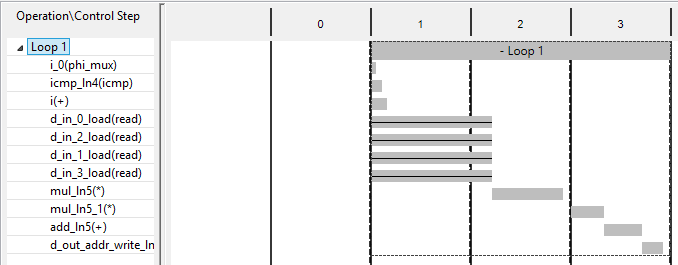


Рис. 2.9.3.5. Schedule viewer

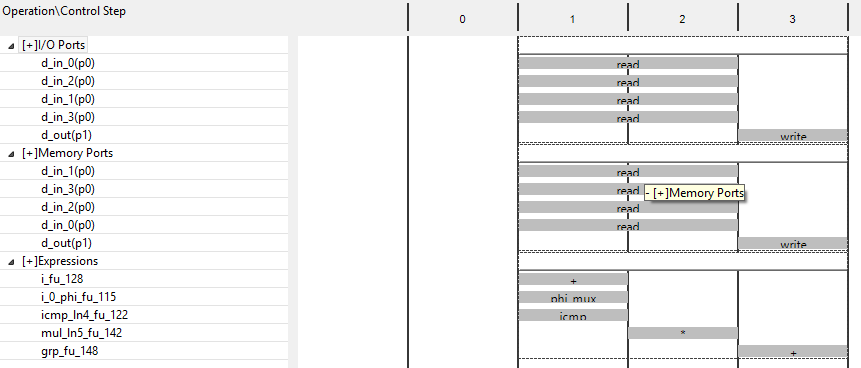


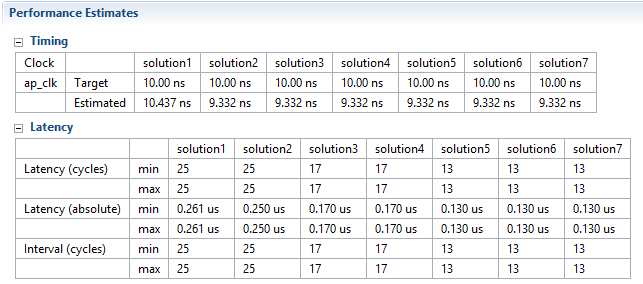
Рис. 2.9.3.6. Schedule viewer

* + 1. Анализ решения

Видно, что задержки остались прежними. Данное решение аналогично решению 5а по всем параметрам. Это означает, что для данного устройства нет необходимости иметь 4 двухпортовых памяти, и программа автоматически создала 4 однопортовых, как более экономичное решение.

* 1. Выводы

Ниже представлена таблица сравнения всех решений первой части.



Наименьшие задержки получили те решения, где есть возможность осуществлять более 4 чтений массива памяти. Решение, которые имеет больше 4 чтений (7а), реализуется, как решение (5а), так как оно избыточно, ведь в устройстве на каждой итерации всего 4 операции чтения. Такая избыточность может понадобиться только, если использовать конвейеризацию, а пока 4 итерации проходят последовательно, этого не требуется.

1. Часть 2
   1. Исходный код

Исходный код функции:

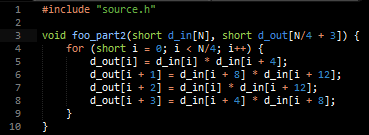


Рис. 3.1.1. source code

Исходный код теста:

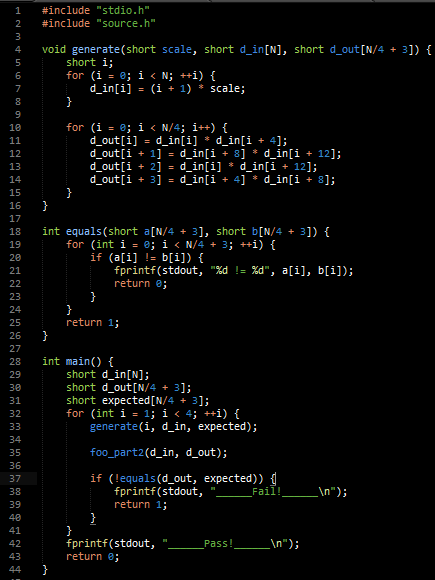


Рис. 3.1.2 test code

* 1. Моделирование

По результатам моделирование, приведённым ниже, видно, что устройство проходит тесты.

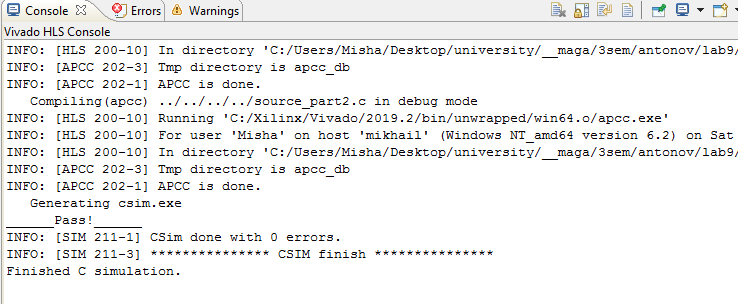


Рис. 3.2. Modeling results

* 1. Решение 1b
     1. Директивы

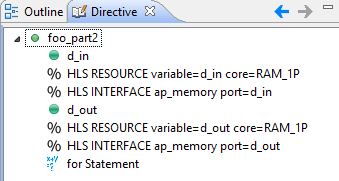


Рис. 3.3.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

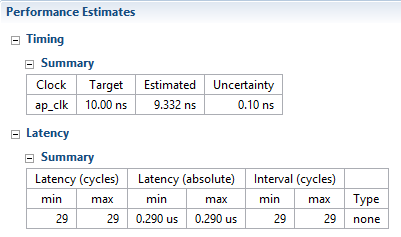


Рис. 3.3.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 4 блока DSP48E, 183 LUT и 116 триггеров.

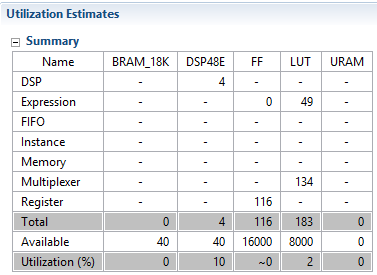


Рис. 3.3.3.1. Utilization estimates

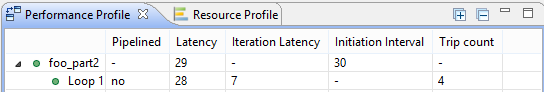


Рис. 3.3.3.2. Performance profile

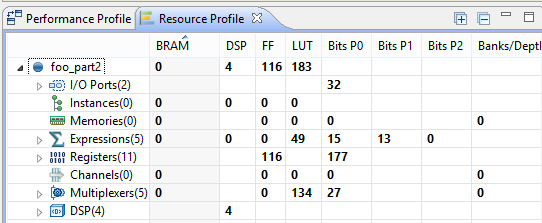


Рис. 3.3.3.3. Resource profile

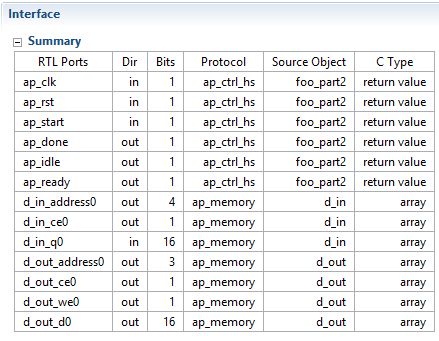


Рис. 3.3.3.4. Interfaces

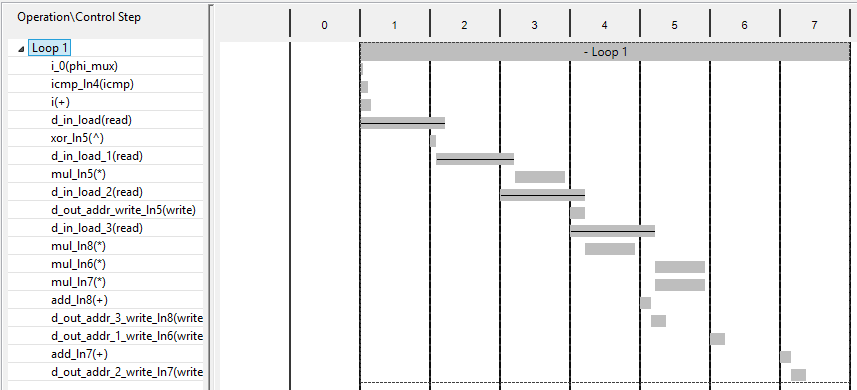


Рис. 3.3.3.5. Schedule viewer

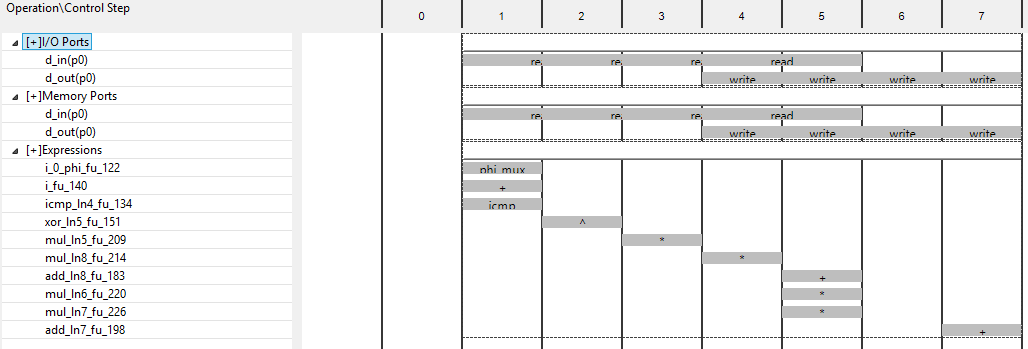


Рис. 3.3.3.5. Resource viewer

* + 1. Анализ решения

Устройство использует 4 умножителя и имеет большие задержки, потому что 4 чтения выполняются последовательно.

* 1. Решение 2b
     1. Директивы

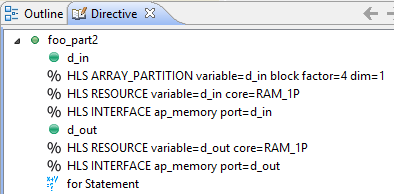


Рис. 3.4.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

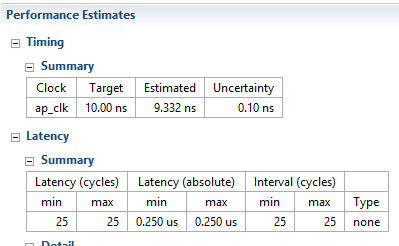


Рис. 3.4.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 4 блока DSP48E, 149 LUT и 112 триггеров.

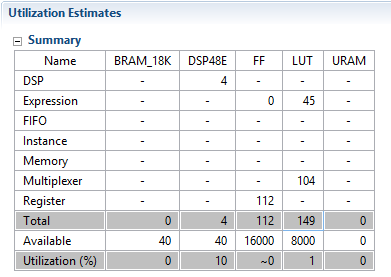


Рис. 3.4.3.1. Utilization estimates

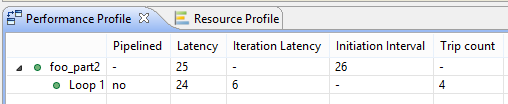


Рис. 3.4.3.2. Performance profile

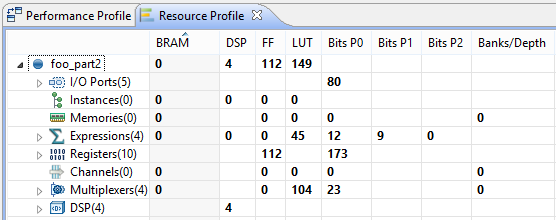


Рис. 3.4.3.3. Resource profile

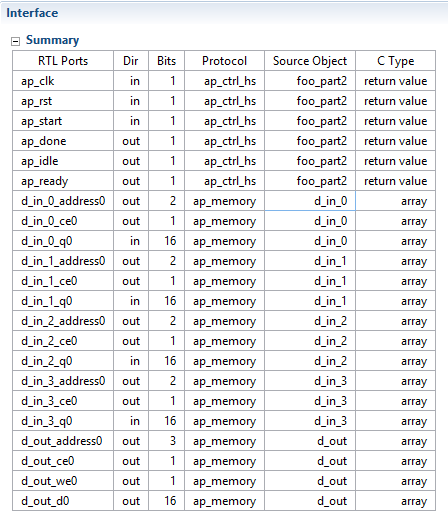


Рис. 3.4.3.4. Interfaces

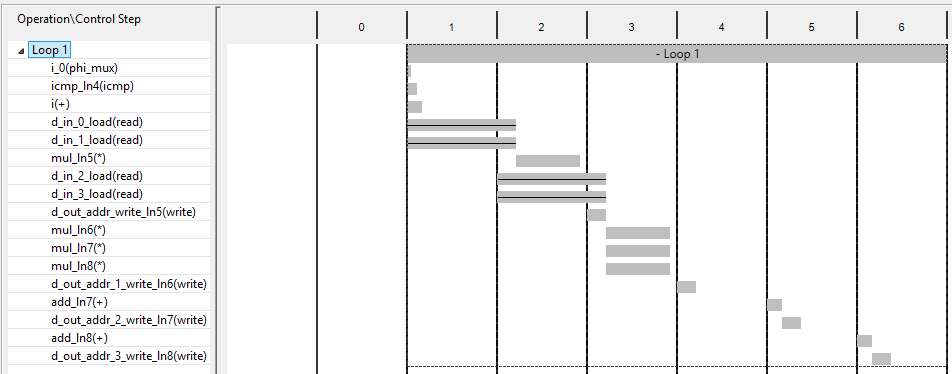


Рис. 3.4.3.5. Schedule viewer

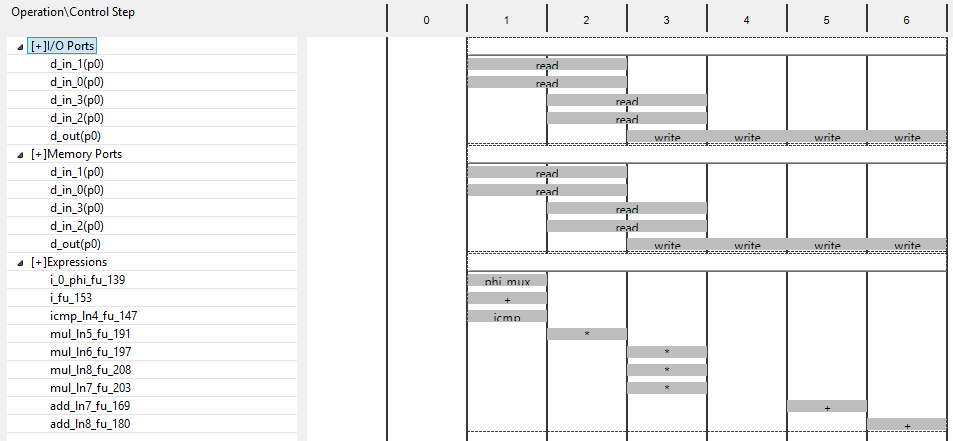


Рис. 3.4.3.5. Resource viewer

* + 1. Анализ решения

В данном решении используется такое же количество умножителей, как и в предыдущем, однако за счёт возможности двух параллельных чтений сократились задержки (цикл за 6 тактов, а на за 7).

* 1. Решение 3b
     1. Директивы

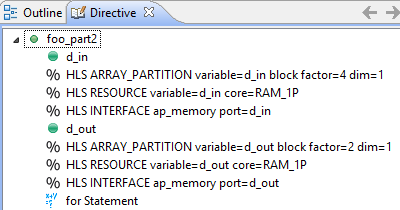


Рис. 3.5.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

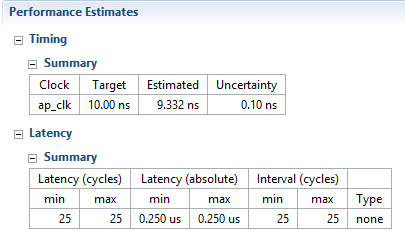


Рис. 3.5.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 4 блока DSP48E, 214 LUT и 159 триггеров.

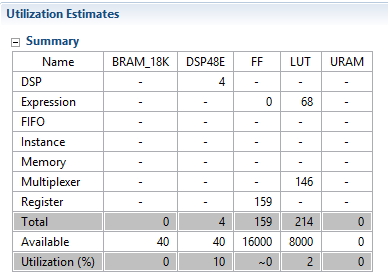


Рис. 3.5.3.1. Utilization estimates

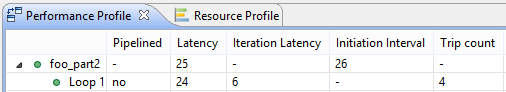


Рис. 3.5.3.2. Performance profile

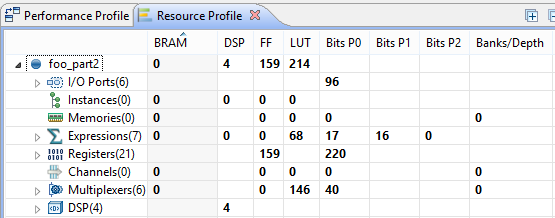


Рис. 3.5.3.3. Resource profile

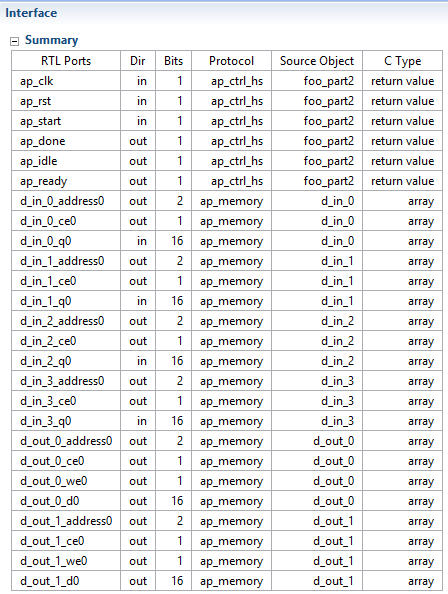


Рис. 3.5.3.4. Interfaces

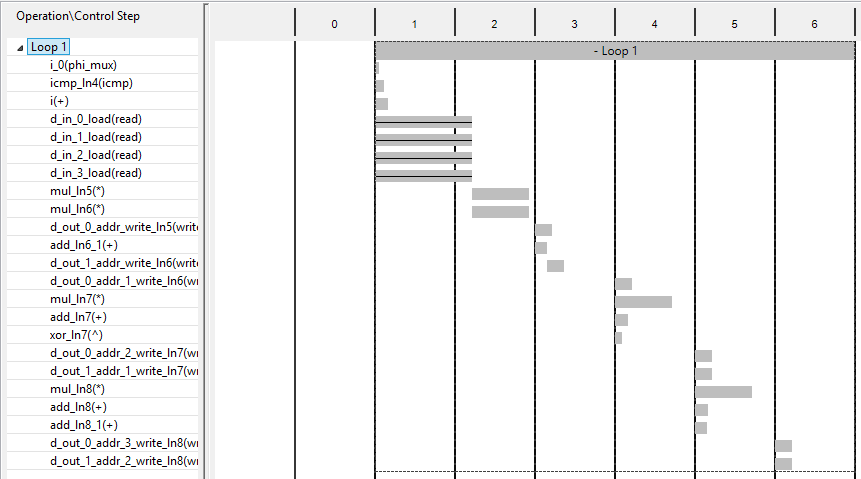


Рис. 3.5.3.5. Schedule viewer

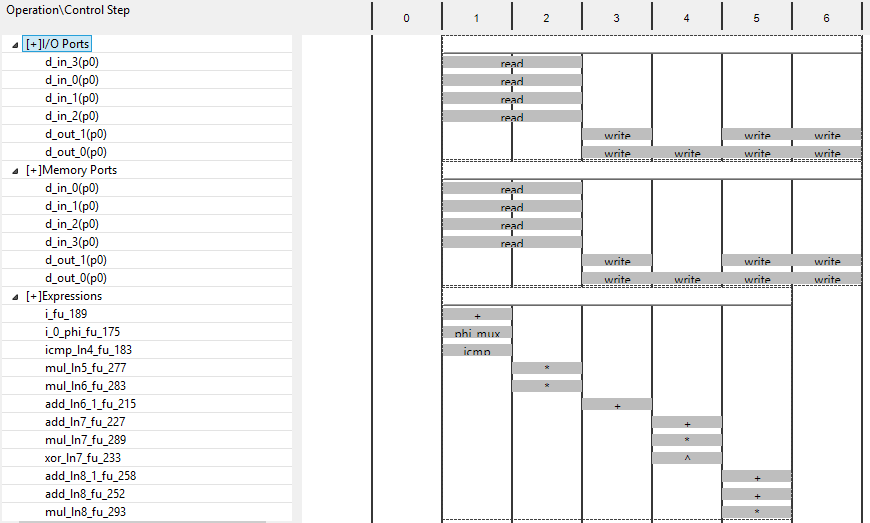


Рис. 3.5.3.5. Resource viewer

* + 1. Анализ решения

В данном решении 4 чтения выполняются одновременно, однако итерация всё равно выполняется 6 тактов. Это связано с тем, что устройство не может одновременно писать 4 значения в выходной массив.

* 1. Решение 4b
     1. Директивы

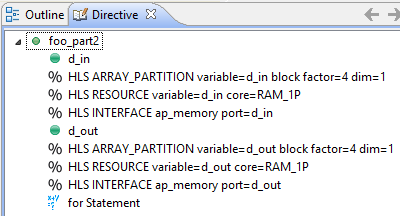


Рис. 3.6.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

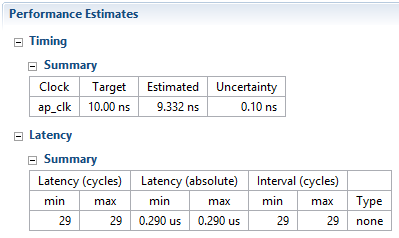


Рис. 3.6.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 4 блока DSP48E, 225 LUT и 161 триггер.

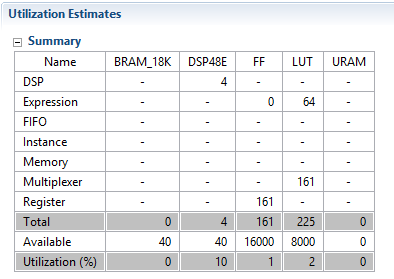


Рис. 3.6.3.1. Utilization estimates

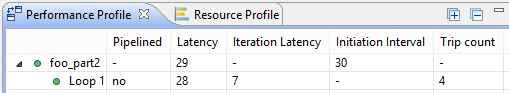


Рис. 3.6.3.2. Performance profile

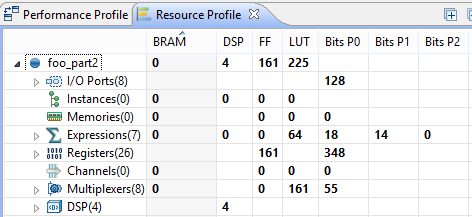


Рис. 3.6.3.3. Resource profile

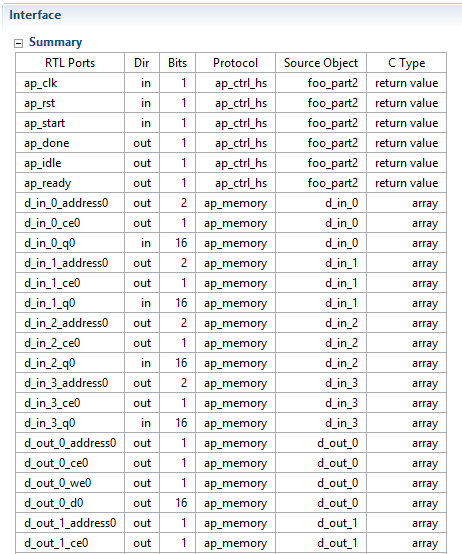


Рис. 3.6.3.4. Interfaces

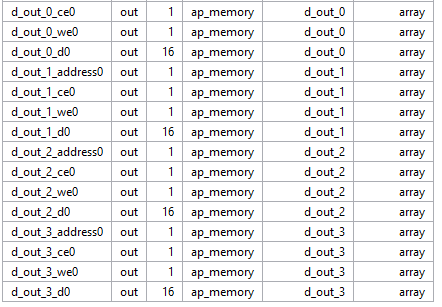


Рис. 3.6.3.5. Interfaces continue

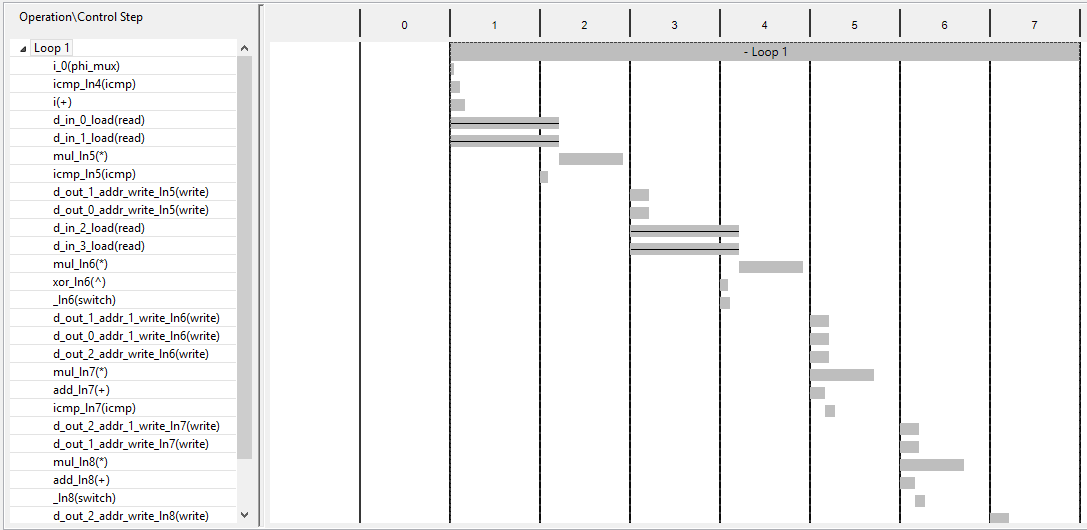


Рис. 3.6.3.6. Schedule viewer

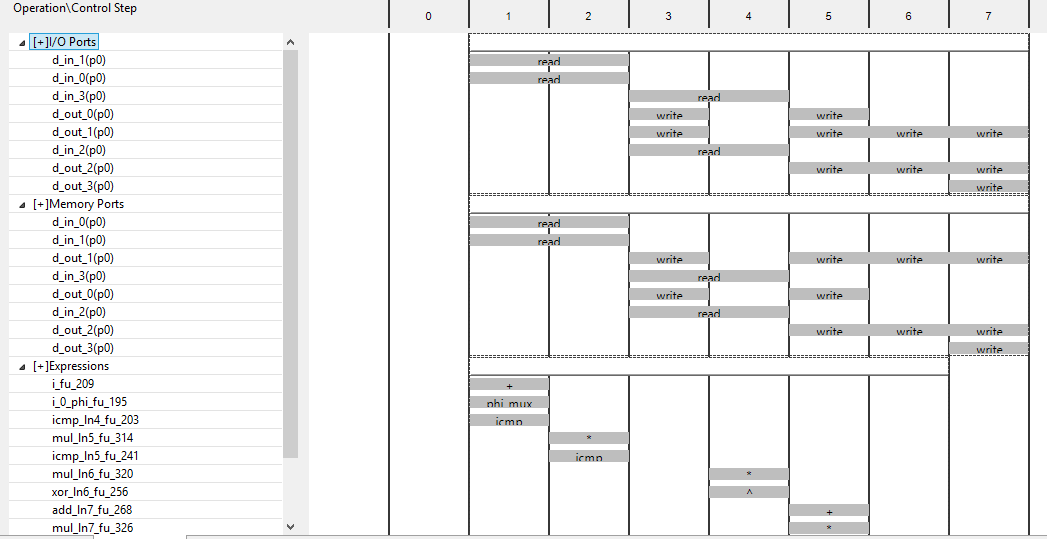


Рис. 3.6.3.7. Resource viewer

* + 1. Анализ решения

В данном решении всё стало хуже, при возможности на запись 4 памяти. В устройстве используется очень много перезаписей в регистры, а не сразу в выходные линии, что не дает ни делать 4 чтения, ни записывать 4 значения на выходы.

* 1. Решение 5b
     1. Директивы

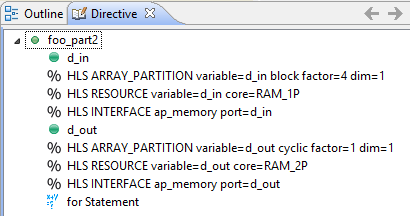


Рис. 3.7.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

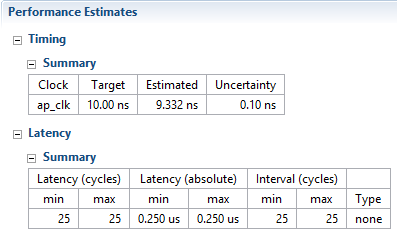


Рис. 3.7.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 4 блока DSP48E, 149 LUT и 112 триггер.

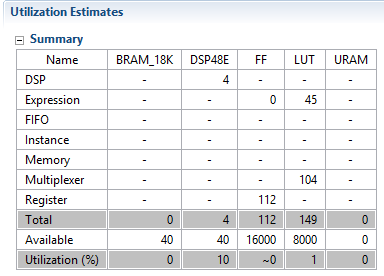


Рис. 3.7.3.1. Utilization estimates

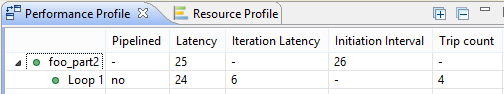


Рис. 3.7.3.2. Performance profile

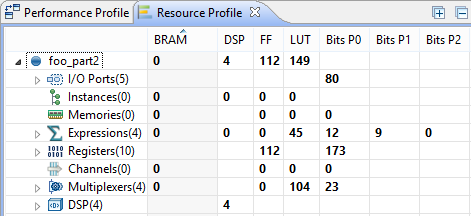


Рис. 3.7.3.3. Resource profile

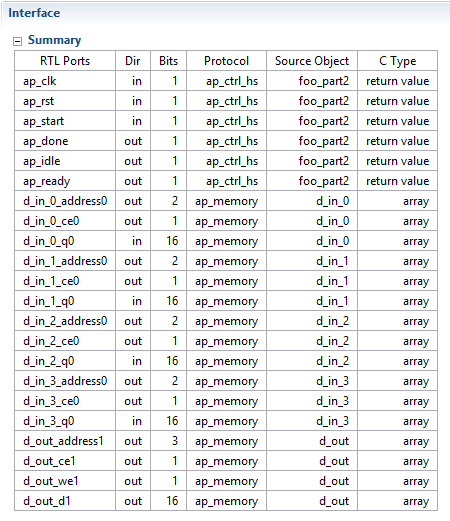


Рис. 3.7.3.4. Interfaces

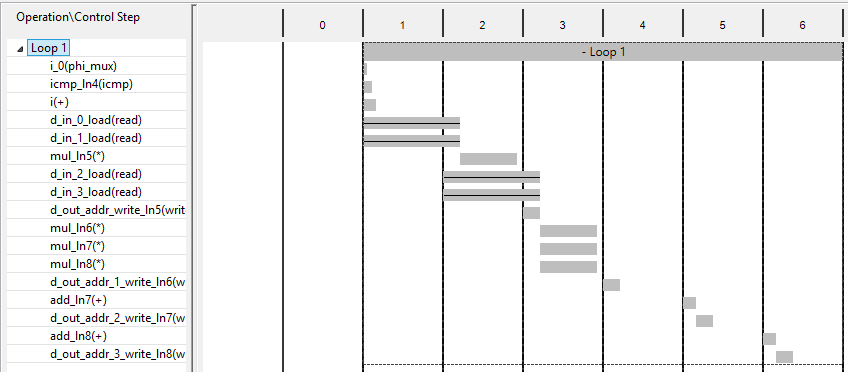


Рис. 3.7.3.5. Schedule viewer

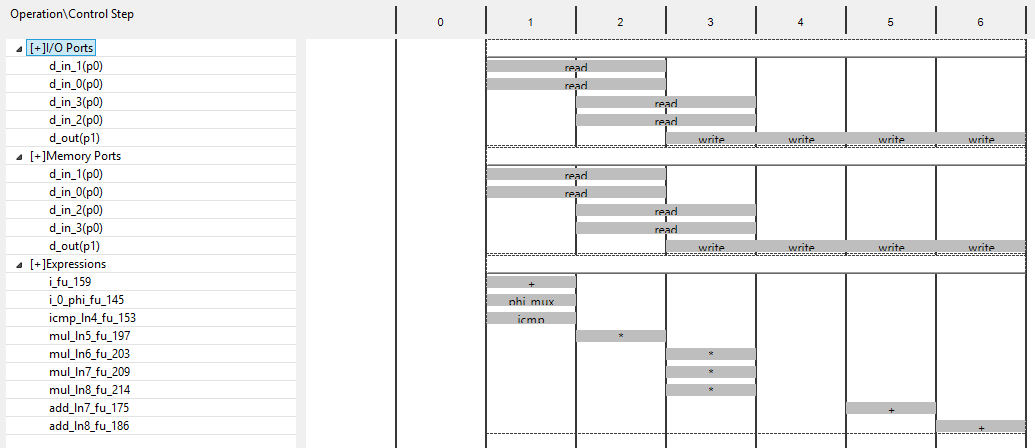


Рис. 3.7.3.6. Resource viewer

* + 1. Анализ решения

В данном решении была использована одна двух портовая память, и это вышло эффективнее, чем 4 одно-портовых. Но одновременно 2 записи всё также не производятся.

* 1. Решение 6b
     1. Директивы

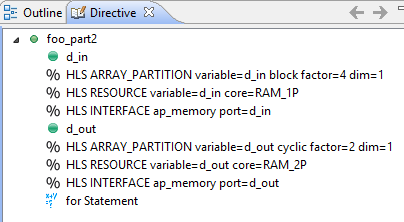


Рис. 3.8.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

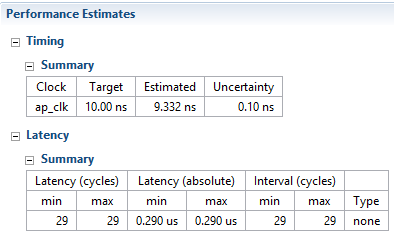


Рис. 3.8.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 4 блока DSP48E, 206 LUT и 164 триггер.

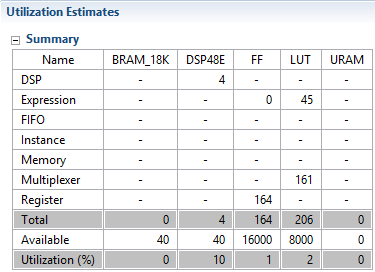


Рис. 3.8.3.1. Utilization estimates

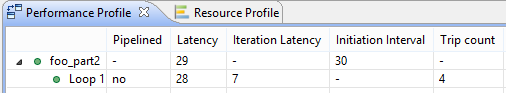


Рис. 3.8.3.2. Performance profile

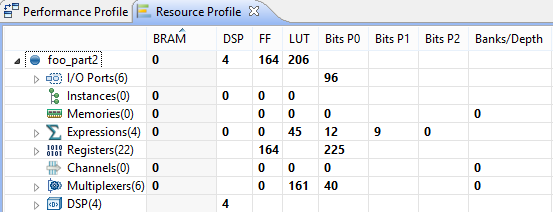


Рис. 3.8.3.3. Resource profile

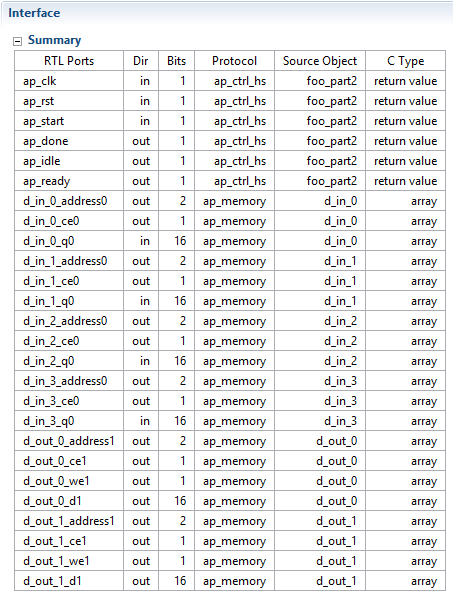


Рис. 3.8.3.4. Interfaces

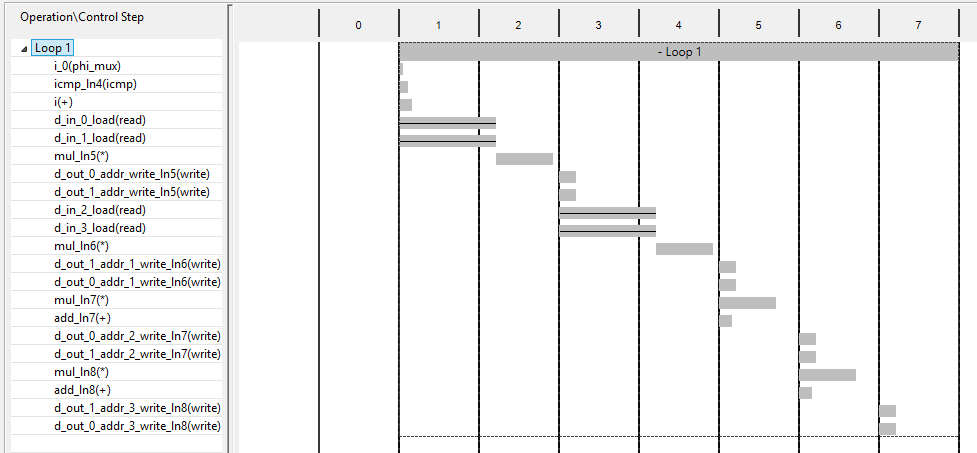


Рис. 3.8.3.5. Schedule viewer

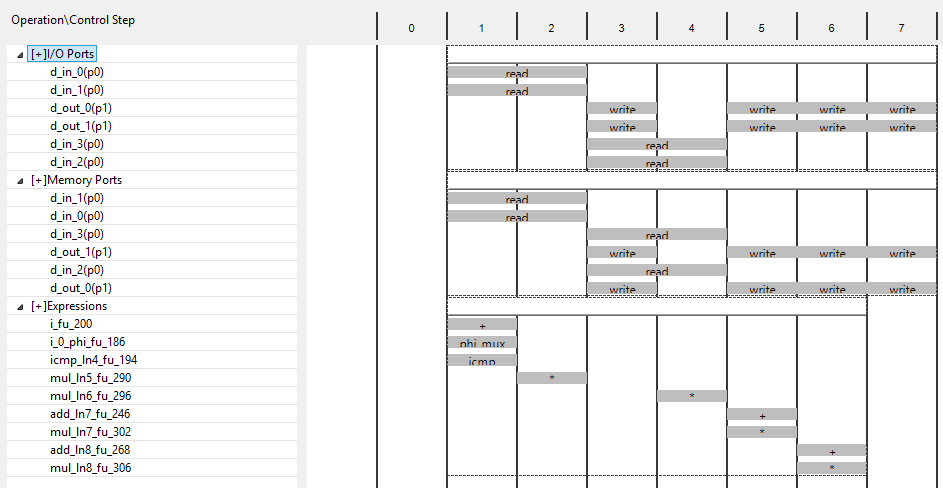


Рис. 3.8.3.6. Resource viewer

* + 1. Анализ решения

В данном решении 4 чтения выполняются одновременно, а также присутствует одновременная запись в выходной массив, но задержки остались такими же.

* 1. Решение 7b
     1. Директивы

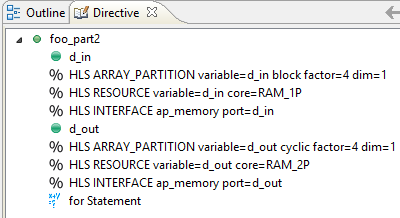


Рис. 3.9.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

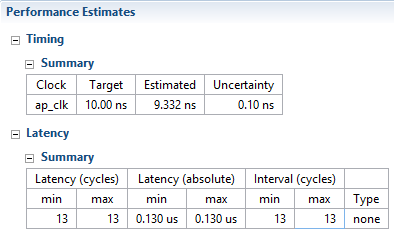


Рис. 3.9.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 4 блока DSP48E, 297 LUT и 89 триггер.

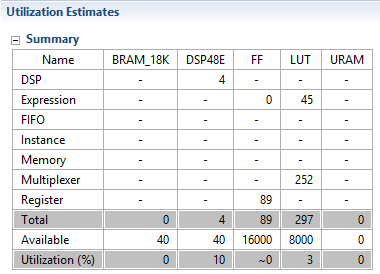


Рис. 3.9.3.1. Utilization estimates

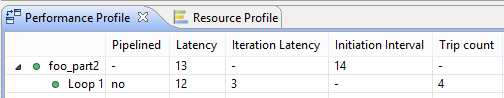


Рис. 3.9.3.2. Performance profile

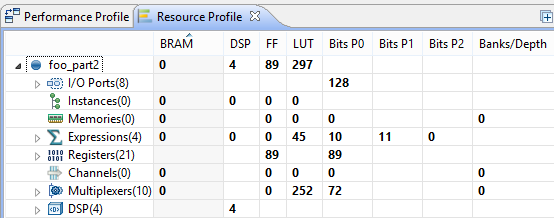


Рис. 3.9.3.3. Resource profile

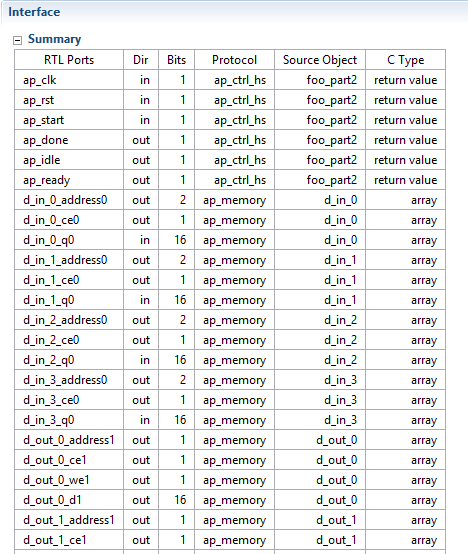


Рис. 3.9.3.4. Interfaces

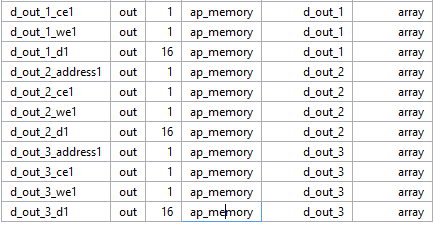


Рис. 3.9.3.5. Interfaces continue

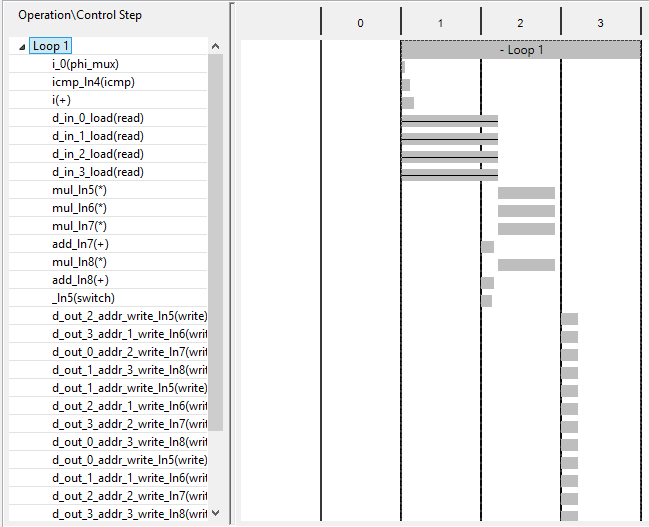


Рис. 3.9.3.6. Schedule viewer

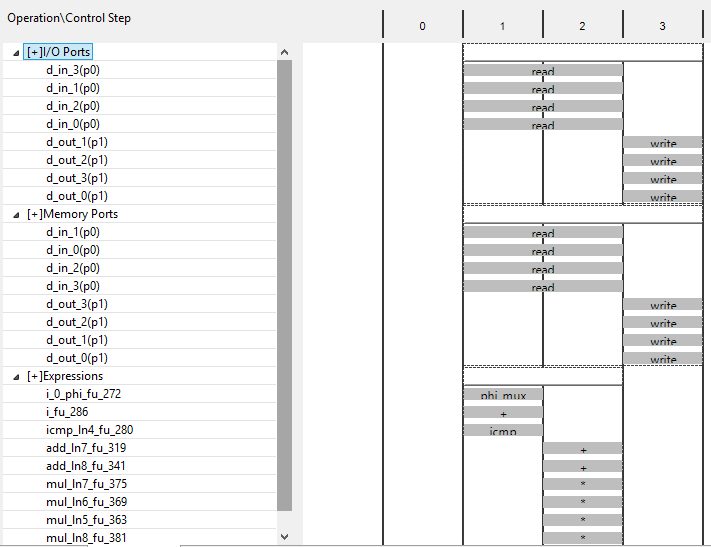


Рис. 3.9.3.7. Resource viewer

* + 1. Анализ решения

Данное решение аналогично с RAM\_1 хуже, чем предыдущее (с factor = 2).

* 1. Решение 8b
     1. Директивы

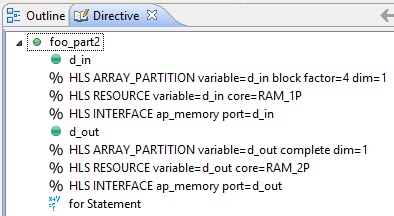


Рис. 3.10.1. Directives

* + 1. Синтез

На изображении видно, что полученная задержка укладывается в заданное значение.

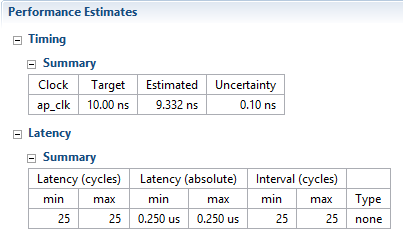


Рис. 3.10.2. Performance estimates

* + 1. Использование ресурсов

Оценка использования ресурсов показывает, что будут использованы 4 блока DSP48E, 149 LUT и 112 триггер.

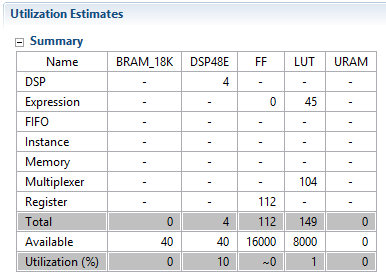


Рис. 3.10.3.1. Utilization estimates

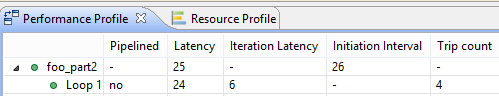


Рис. 3.10.3.2. Performance profile

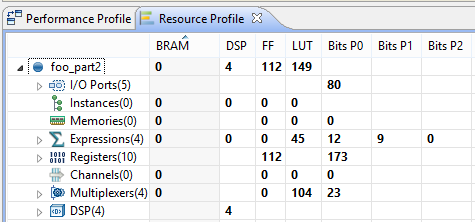


Рис. 3.10.3.3. Resource profile

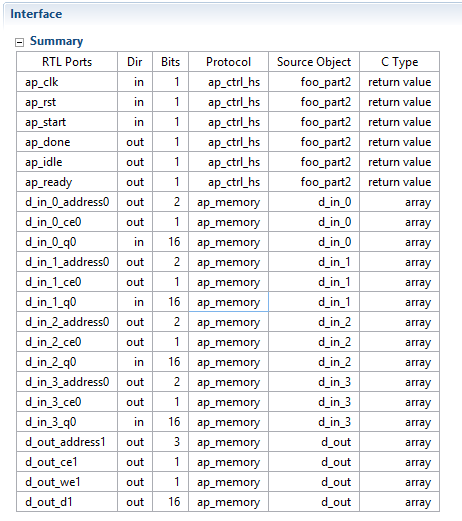


Рис. 3.10.3.4. Interfaces

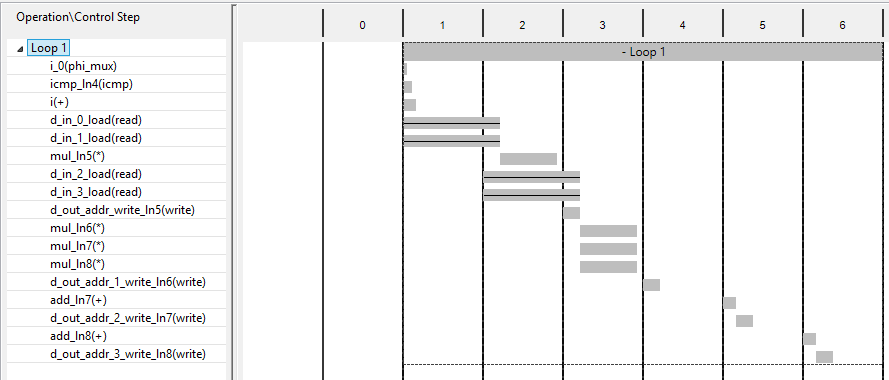


Рис. 3.10.3.5. Schedule viewer

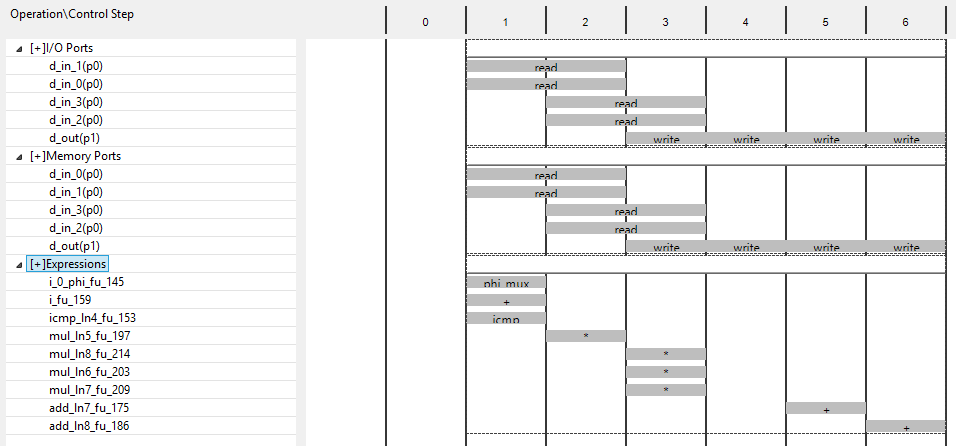


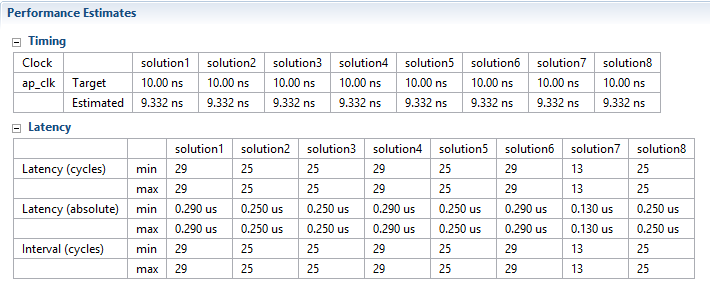
Рис. 3.10.3.6. Resource viewer

* + 1. Анализ решения

В данном решении было проигнорировано разбиение выходного массива. Возможно, это связано с особенностью устройства, в общем случае это должно дать наивысшую скорость записи выходных данных.

* 1. Выводы

Ниже представлена таблица сравнения всех решений второй части.



По разбиению выходной памяти не всегда получается ожидаемый прирост производительности, а часто даже ухудшение. Согласно временным диаграммам, запись в выходную память почти никогда не производится параллельно для нескольких ячеек. Вероятно, это связано с особенностью устройства, где на выход подаются различные комбинации одних и тех же входов.