Санкт-Петербургский Политехнический Университет Петра Великого

Институт Компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

Лабораторная работа 10

Предмет: Проектирование реконфигурируемых гибридных вычислительных систем

Тема: Упаковка данных

Задание 1

Студент: Медведев М.А.

Онищук М.П.

Гр. № 3540901/81501,

3540901/81502

Преподаватель: Антонов А.П.

Санкт-Петербург

2019

Оглавление

[1. Задание 4](#_Toc28690654)

[2. Моделирование 7](#_Toc28690655)

[3. Первое решение 7](#_Toc28690656)

[3.1. Синтез 7](#_Toc28690657)

[3.2. C/RTL моделирование 10](#_Toc28690658)

[4. Второе решение 11](#_Toc28690659)

[4.1. Синтез 11](#_Toc28690660)

[4.2. C\RTL моделирование 14](#_Toc28690661)

[5. Третье решение 15](#_Toc28690662)

[5.1. Синтез 16](#_Toc28690663)

[5.2. C\RTL моделирование 19](#_Toc28690664)

[6. Четвёртое решение 20](#_Toc28690665)

[6.1. Синтез 21](#_Toc28690666)

[6.2. C\RTL моделирование 24](#_Toc28690667)

[7. Выводы 25](#_Toc28690668)

1. Задание

* Создать проект lab10\_z1
* Микросхема: xa7a12tcsg325-1q
* Познакомиться с исходным кодом struct\_port.c
* Познакомиться с исходным кодом struct\_port\_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 ПО УМОЛЧАНИЮ
  + 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
  + Выполнить cosimulation и привести временную диаграмму (интерес представляет количество и тип портов)
* Solution\_2а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию DATA\_PACK
  + 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
  + Выполнить cosimulation и привести временную диаграмму (интерес представляет количество и тип портов)
* Сравнить два решения (solution\_1a и solution\_2a) и сделать выводы: зависимость от DATA\_PACK; объяснить (посчитать) число циклов Latency, II…
* Solution\_3а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию DATA\_PACK with struct\_level
  + 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
  + Выполнить cosimulation и привести временную диаграмму (интерес представляет количество и тип портов)
* Сравнить два решения (solution\_2a и solution\_3a) и сделать выводы: зависимость от типа интерфейса; объяснить (посчитать) число циклов Latency, II…
* Solution\_4а
  + задать: clock period 10; clock\_uncertainty 0.1
  + установить реализацию DATA\_PACK with field\_level
  + 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
  + Выполнить cosimulation и привести временную диаграмму (интерес представляет количество и тип портов)
* Сравнить два решения (solution\_3a и solution\_4a) и сделать выводы: зависимость от типа интерфейса; объяснить (посчитать) число циклов Latency, II…

Исходный текст подготовленной для синтеза функции и теста приведён ниже:

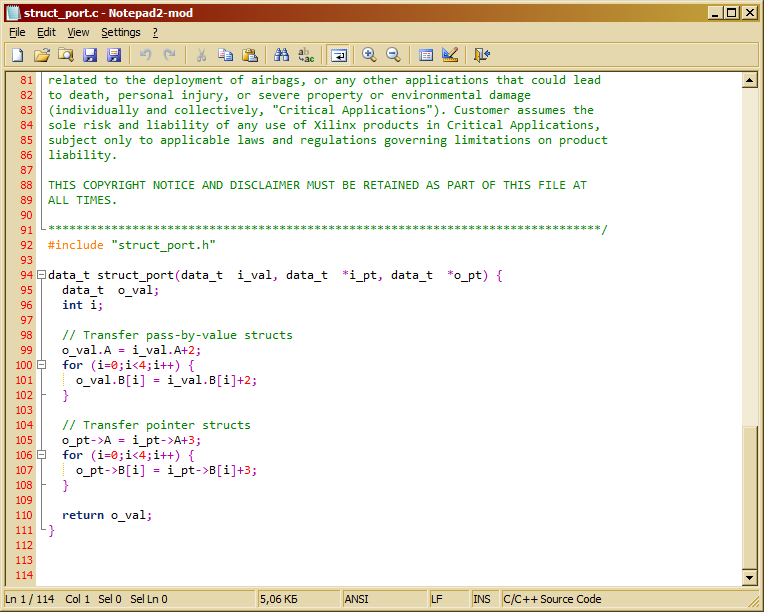


Рис. 1.1. Исходный код синтезируемой функции

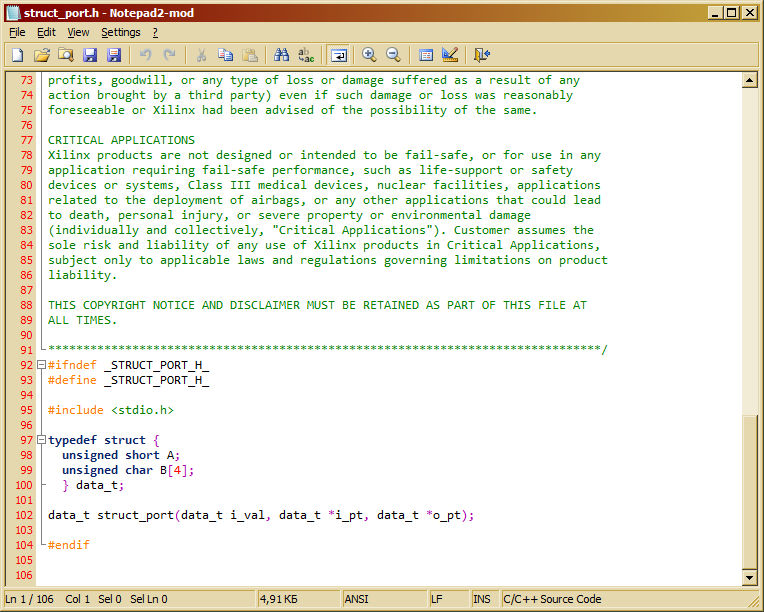


Рис. 1.2. Заголовочный файл

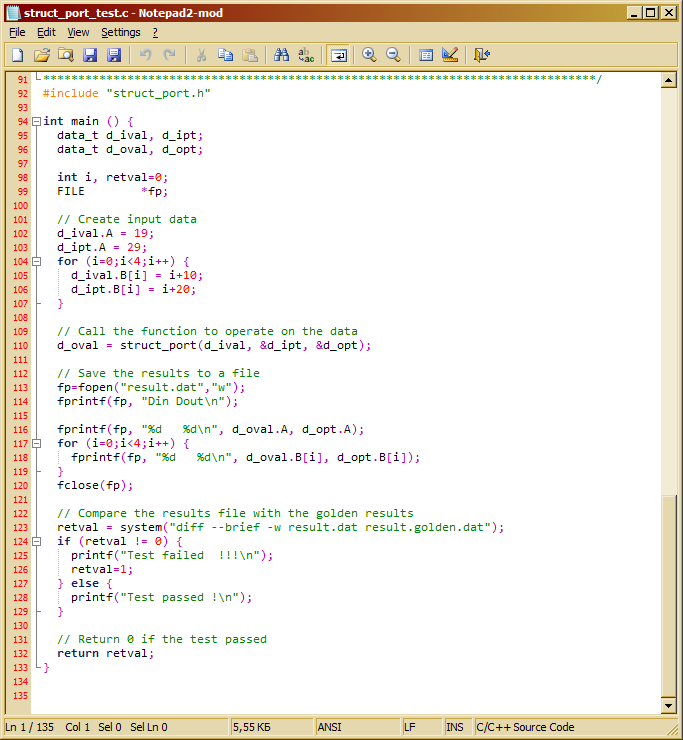


Рис. 1.3. Исходный код теста

1. Моделирование

При запуске моделирования можно увидеть, что тест успешно пройден:

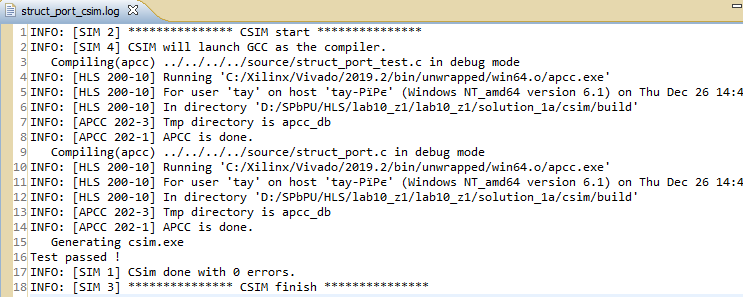


Рис. 2.1. Результаты моделирования

1. Первое решение
   1. Синтез

Приведем в отчете требуемые данные о проекте:

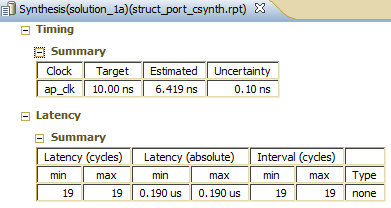


Рис. 3.1. Производительность

Здесь можно увидеть, что достигнутая задержка равна 6.419 + 0.1, что укладывается в заданные нами требования к тактовой частоте.

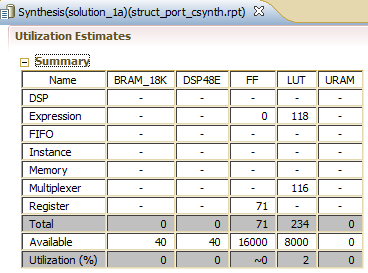


Рис. 3.2. Занимаемые ресурсы

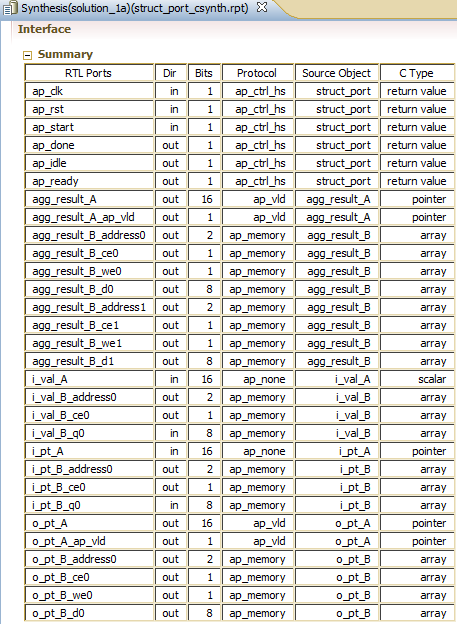


Рис. 3.3. Применяемые интерфейсы

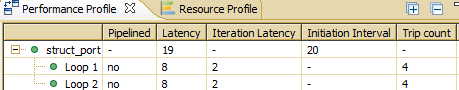


Рис. 3.4. Профиль производительности

На этом рисунке видно, что задержка получения первого выходного значения составляет 8 тактов с момента старта (всех данных – 19), а задержка после старта до готовности приема новых данных – 20:

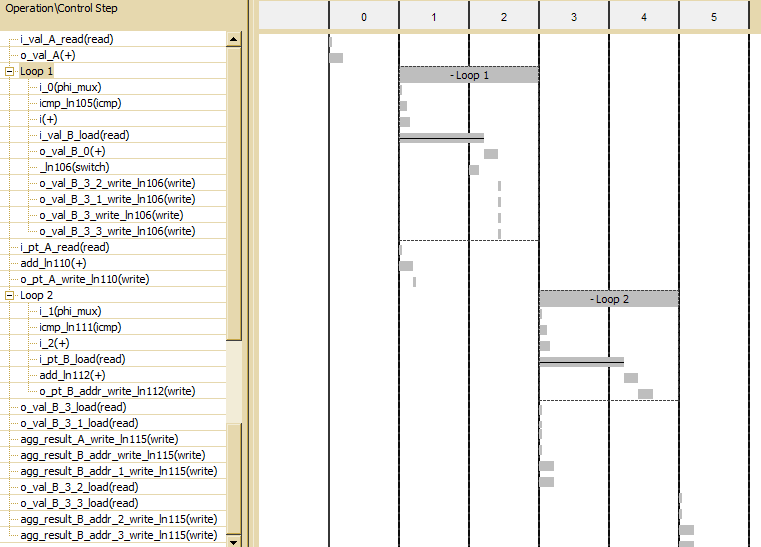


Рис. 3.5. Временная диаграмма

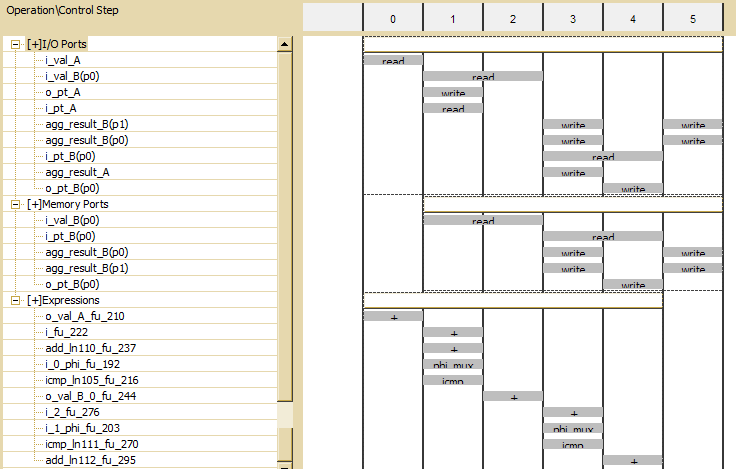


Рис. 3.6. Диаграмма использования ресурсов

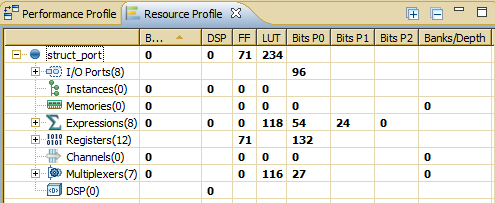


Рис. 3.7. Профиль ресурсов

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

* 1. C/RTL моделирование

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

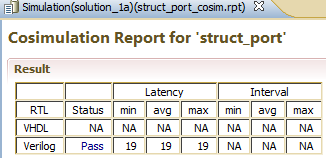


Рис.3.8. Результаты C\RTL моделирования

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

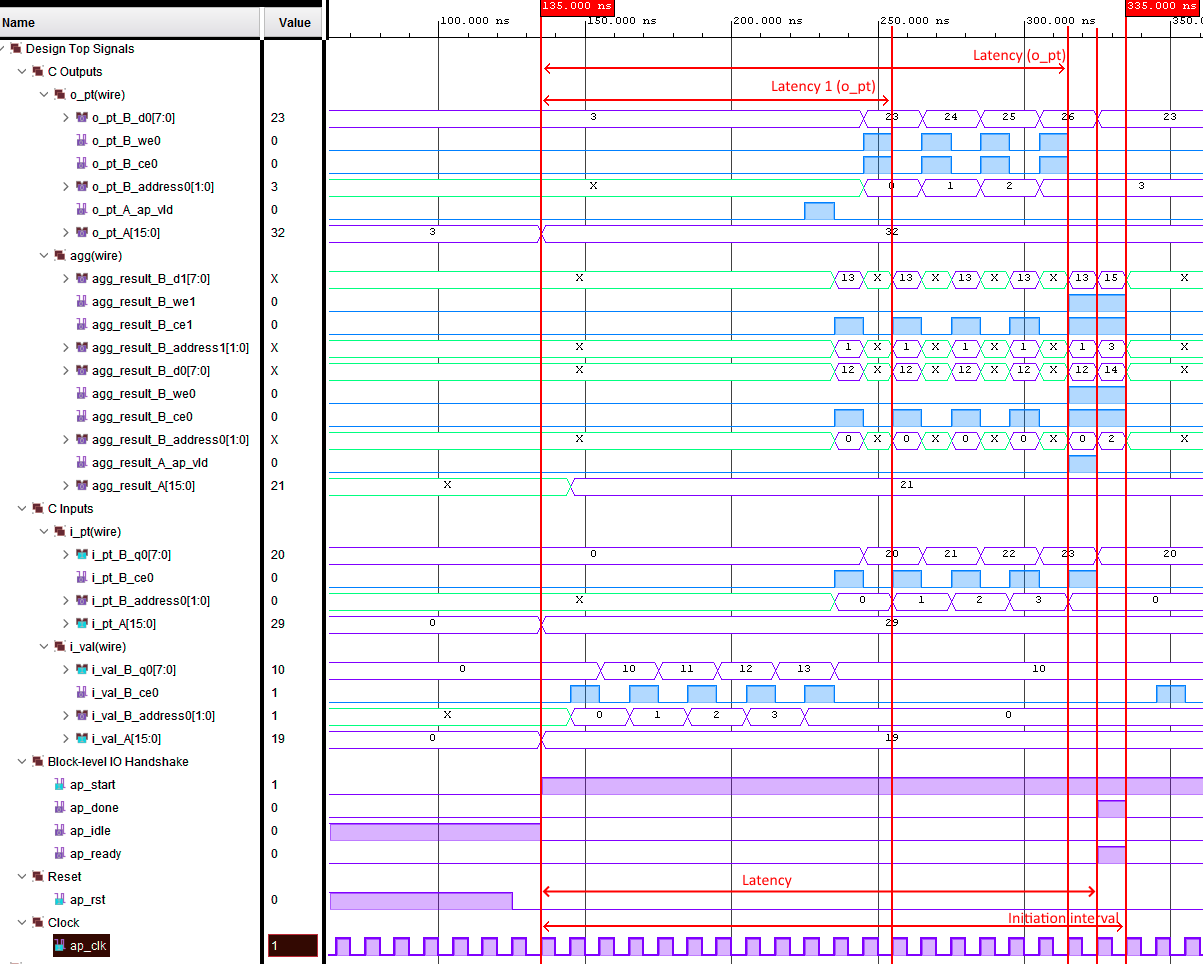


Рис. 3.9. Временная диаграмма совместного моделирования

1. Второе решение

Добавим директиву, которая изменяет способ передачи данных.

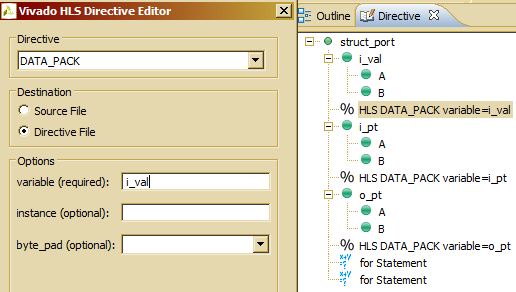


Рис. 4.1. Добавление директивы

* 1. Синтез

Приведем в отчете требуемые данные о проекте:

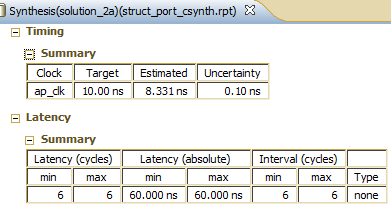


Рис. 4.2. Производительность

Здесь можно увидеть, что достигнутая задержка равна 8.331 + 0.1, что укладывается в заданные нами требования к тактовой частоте.

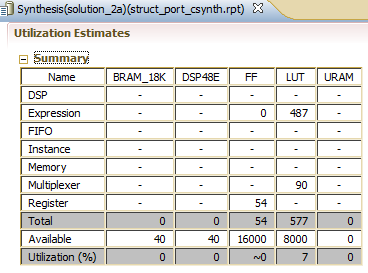


Рис. 4.3. Затрачиваемые ресурсы

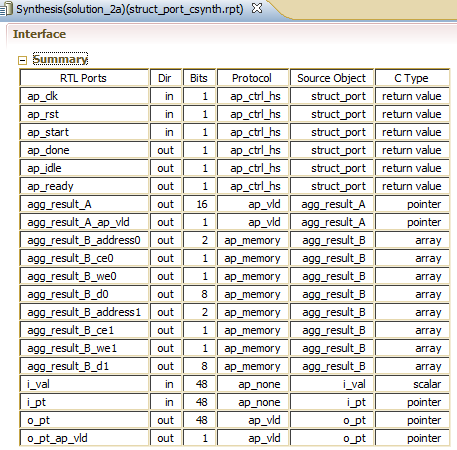


Рис. 4.4. Применяемые интерфейсы

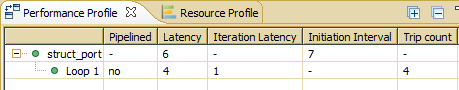


Рис. 4.5. Профиль производительности

На этом рисунке видно, что задержка получения первого выходного значения составляет 4 такта с момента старта (6 для всех), а задержка после старта до готовности приема новых данных – 7:

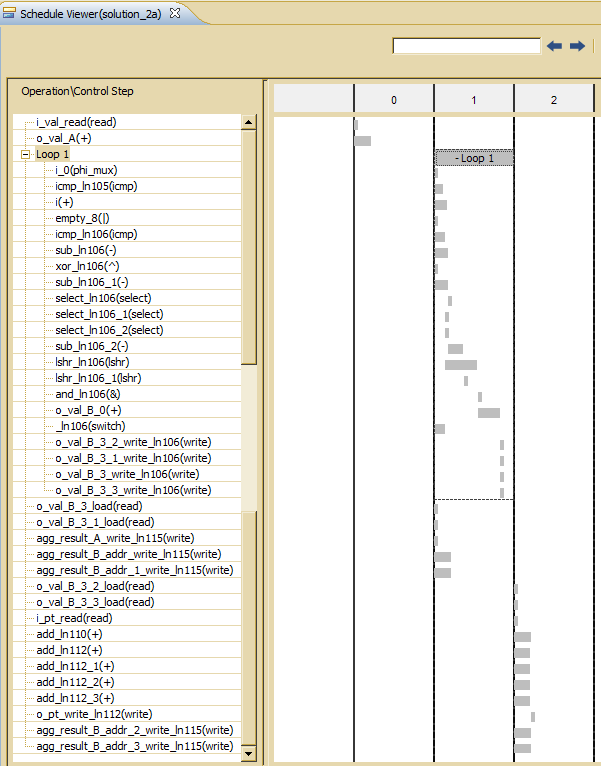


Рис. 4.6. Временная диаграмма

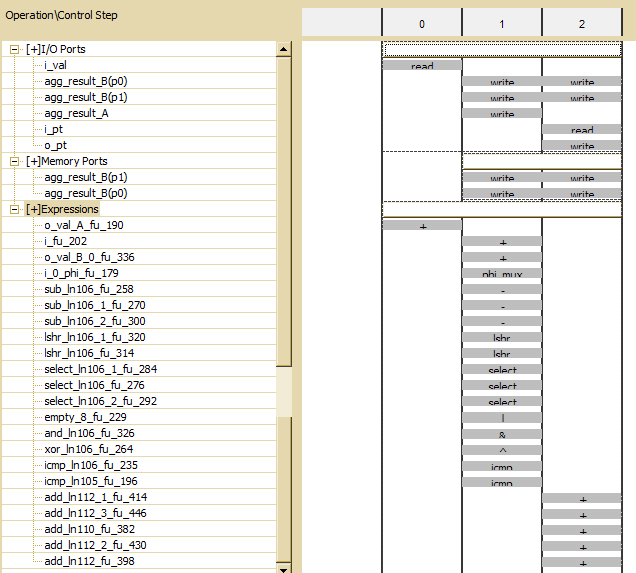


Рис. 4.7. Диаграмма использования ресурсов

Наконец покажем профиль ресурсов:

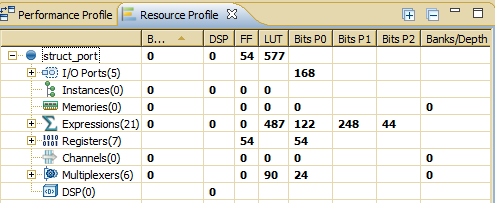


Рис. 4.8. Профиль ресурсов

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

* 1. C\RTL моделирование

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

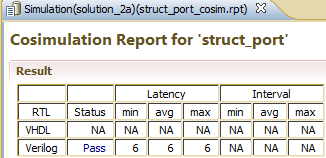


Рис. 4.9. C\RTL моделирование

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

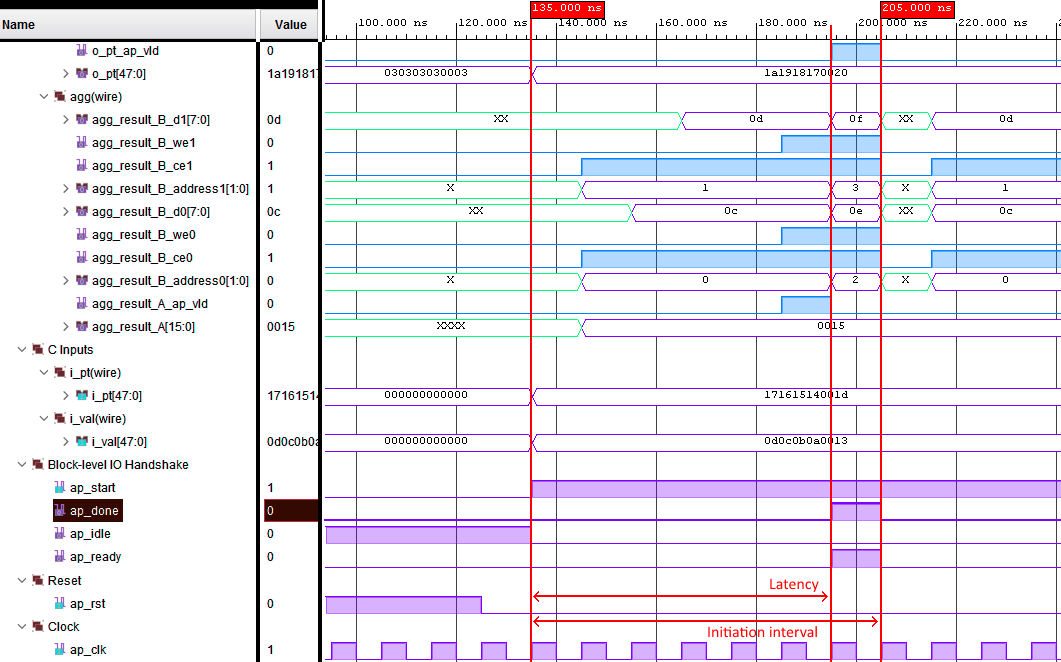


Рис. 4.10. Временная диаграмма совместного моделирования

Здесь также видны отличия во времени выполнения итераций и протоколе работы. Видно, директива была успешно применена к аргументам функции сделав из них один порт шириной 48 бит, это позволило получить одновременный доступ ко всем элементам структуры. Однако, структуру o\_val развернуть не удалось т.к. не удаётся применить директиву к «return», вследствие чего не удалось выполнить распараллеливание первый цикл.

1. Третье решение

Добавим директиву, которая изменяет способ передачи данных.

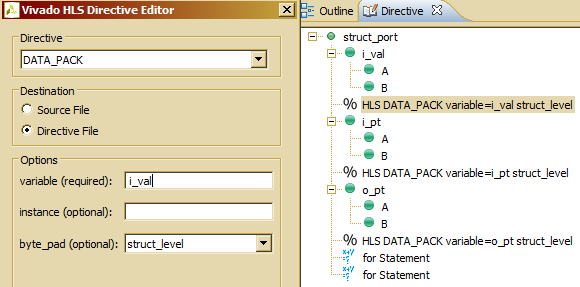


Рис. 5.1. Добавление директивы

* 1. Синтез

Приведем в отчете требуемые данные о проекте:

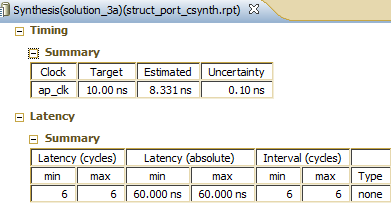


Рис. 5.2. Производительность

Здесь можно увидеть, что достигнутая задержка равна 8.331 + 0.1, что укладывается в заданные нами требования к тактовой частоте.

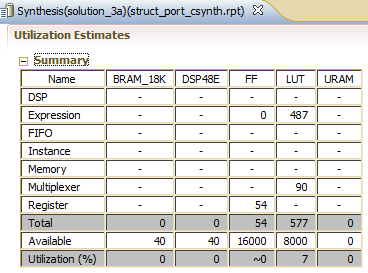


Рис. 5.3. Затрачиваемые ресурсы

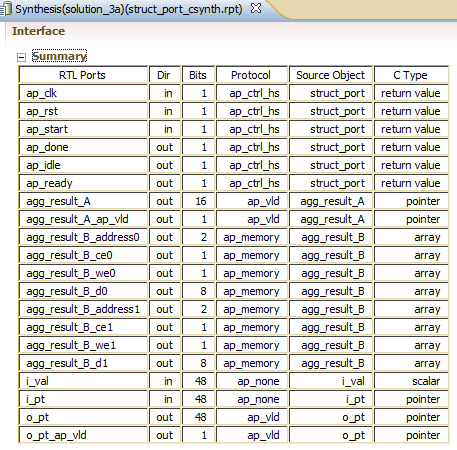


Рис. 5.4. Применяемые интерфейсы

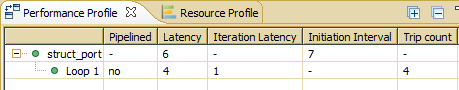


Рис. 5.5. Профиль производительности

На этом рисунке видно, что задержка получения первого выходного значения составляет 4 такта с момента старта (6 для всех), а задержка после старта до готовности приема новых данных – 7:

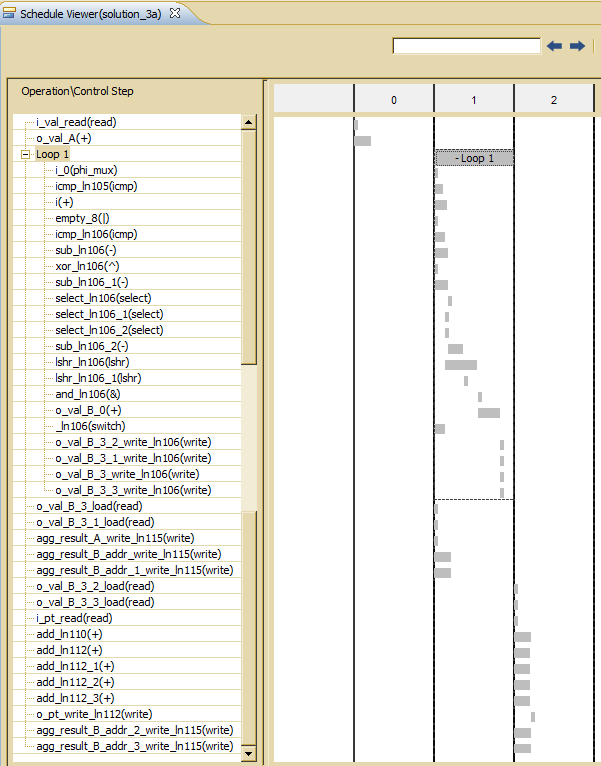


Рис. 5.6. Временная диаграмма

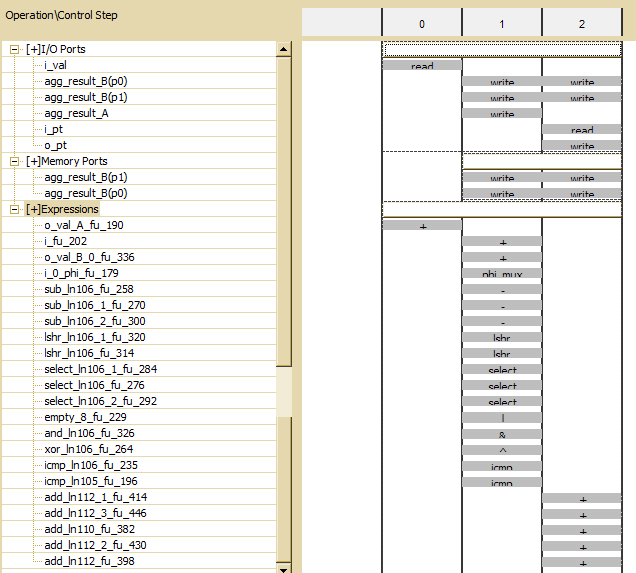


Рис. 5.7. Диаграмма использования ресурсов

Наконец покажем профиль ресурсов:

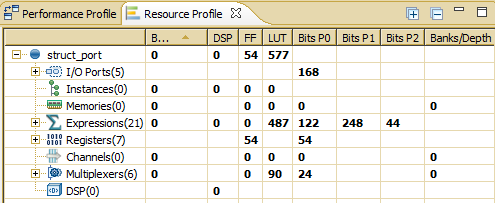


Рис. 5.8. Профиль ресурсов

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

* 1. C\RTL моделирование

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

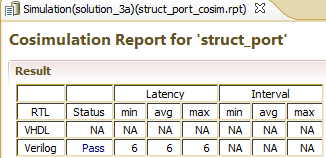


Рис. 5.9. C\RTL моделирование

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

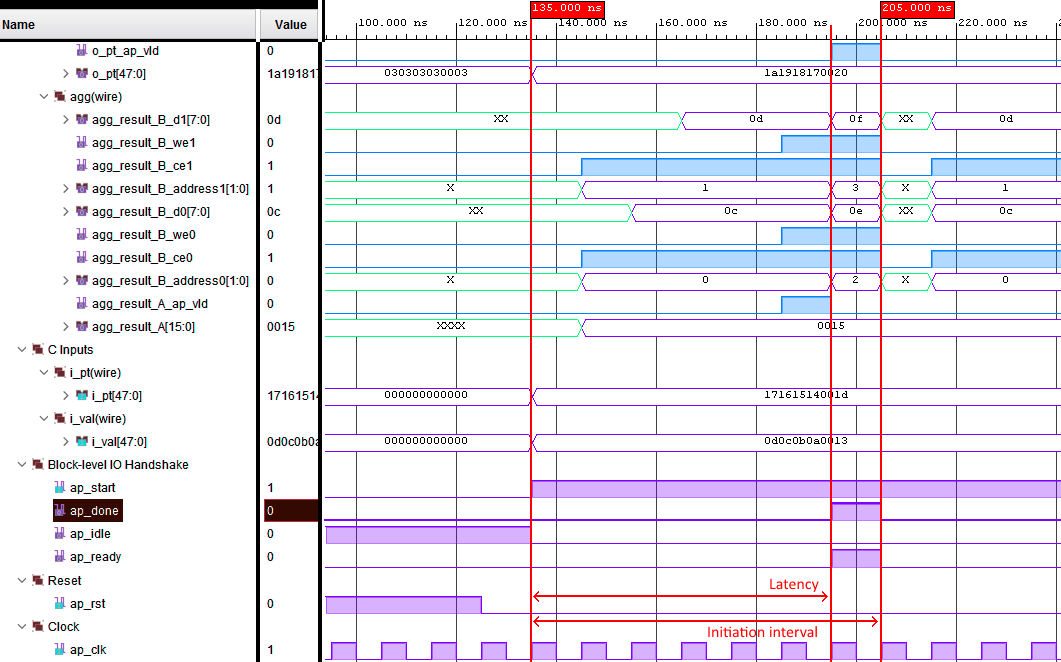


Рис. 5.10. Временная диаграмма совместного моделирования

Результаты полученного решения совпадает с предыдущим.

1. Четвёртое решение

Добавим директиву, которая изменяет способ передачи данных.

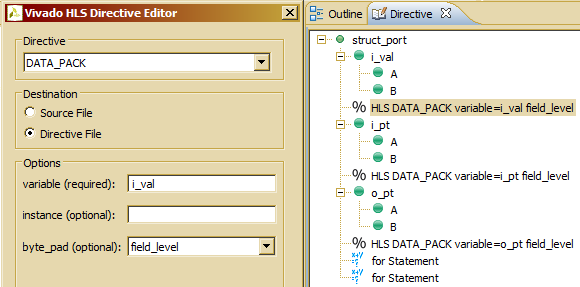


Рис. 6.1. Добавление директивы

* 1. Синтез

Приведем в отчете требуемые данные о проекте:

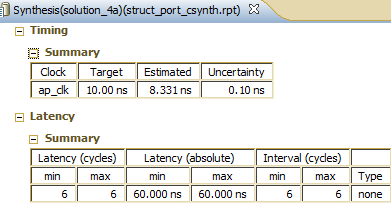


Рис. 6.2. Производительность

Здесь можно увидеть, что достигнутая задержка равна 8.331 + 0.1, что укладывается в заданные нами требования к тактовой частоте.

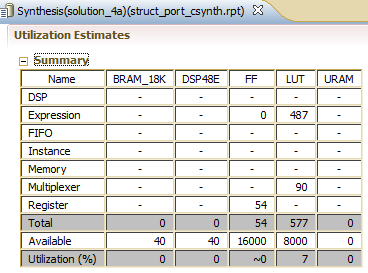


Рис. 6.3. Затрачиваемые ресурсы

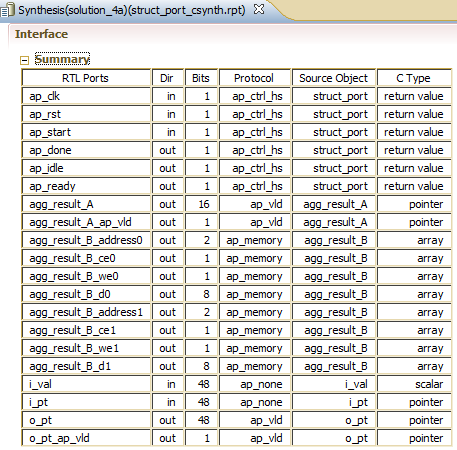


Рис. 6.4. Применяемые интерфейсы

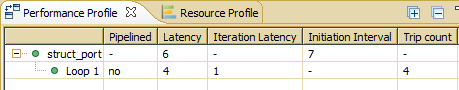


Рис. 6.5. Профиль производительности

Заметно сходство с результатами предыдущего решения.

На этом рисунке видно, что задержка получения первого выходного значения составляет 4 такта с момента старта (6 для всех), а задержка после старта до готовности приема новых данных – 7:

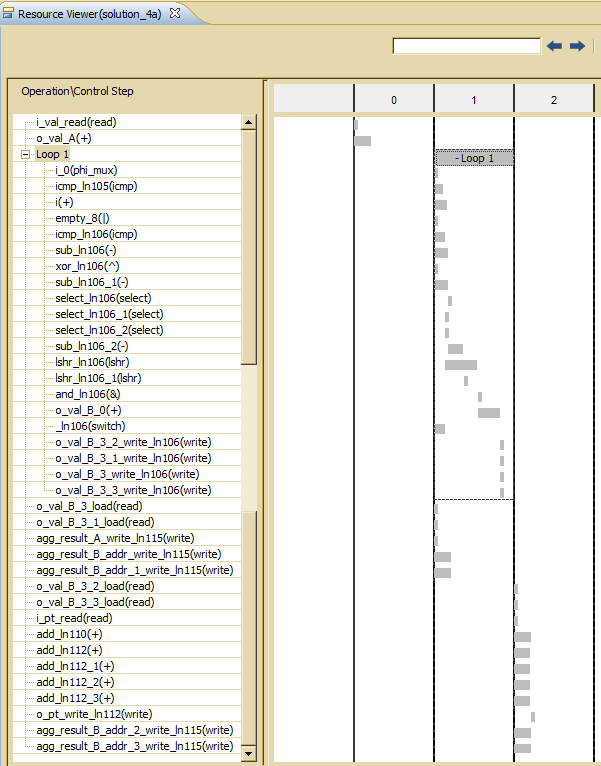


Рис. 6.6. Временная диаграмма

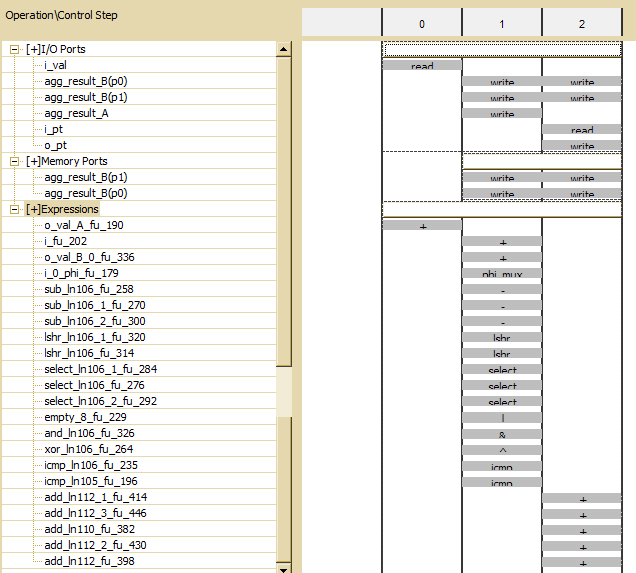


Рис. 6.7. Диаграмма использования ресурсов

Наконец покажем профиль ресурсов:

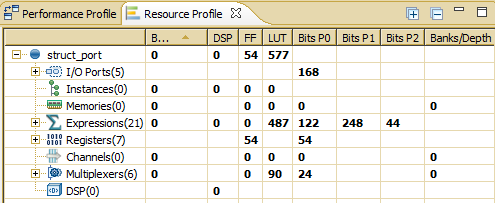


Рис. 6.8. Профиль ресурсов

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

* 1. C\RTL моделирование

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

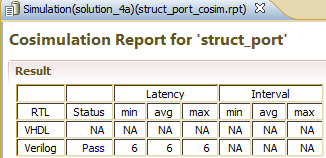


Рис. 6.9. C\RTL моделирование

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

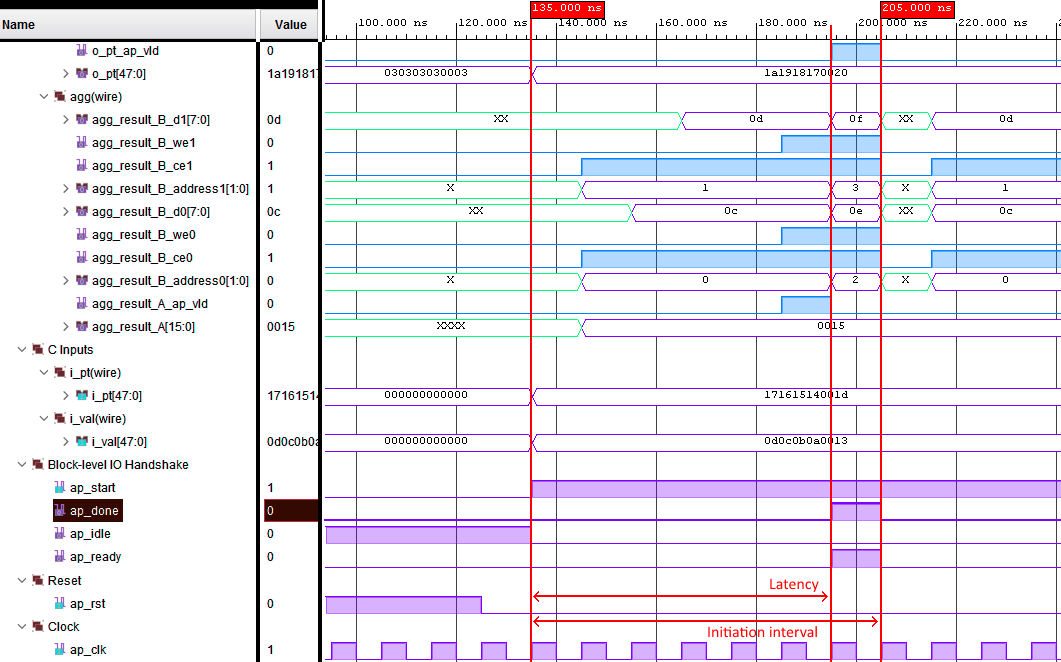


Рис. 6.10. Временная диаграмма совместного моделирования

Результаты полученного решения совпадает с предыдущим.

1. Выводы

В данной работе было проведено исследование влияния директивы DATA\_PACK на функции, оперирующие со структурами данных. Данная директива позволяет развернуть структуру в один порт соответствующей ширины, однако это требует большего количество ресурсов. Изменения параметра byte\_pad в данной работе никак не сказалось на результате, вследствие, вероятно, простоты синтезируемой функции.