# Введение в поток инструментов Vivado HLS

2020.1

## Аннотация

В этой лаборатории рассказывается о том, как выполнять основные действия с помощью потока проектирования Vivado® High-Level Synthesis (HLS).

Эта лаборатория должна занять примерно 45 минут.

## Цели

После завершения этой лаборатории вы сможете:

* Создайте новый проект в графическом интерфейсе инструмента Vivado HLS
* Имитируйте конструкцию C с помощью испытательного стенда с самоконтролем bench
* Синтезируйте дизайн
* Выполните анализ проектирования с использованием перспективного представления анализа
* Выполните совместное моделирование сгенерированного проекта RTL с помощью предоставленного испытательного стенда C
* Реализуйте дизайн

## Введение

В этой лаборатории вы познакомитесь с основными функциями графического интерфейса инструмента Vivado® High-Level Synthesis (HLS). Вы будете использовать инструмент Vivado HLS в режиме графического интерфейса для создания проекта. Вы также будете моделировать, синтезировать и реализовывать предоставленный дизайн.

В этой лаборатории вы будете использовать конструкцию C для реализации дискретного косинусного преобразования (DCT). Функция реализует алгоритм 2D DCT, сначала обрабатывая каждую строку входного массива с помощью 1D DCT, затем обрабатывая столбцы результирующего массива с помощью того же 1D DCT. Он вызывает функции read\_data, dct\_2dи write\_data.

The Функция function read\_data определена в строке 54 и состоит из двух циклов: RD\_Loop\_Row и RD\_Loop\_Col. Функция write\_dataопределена в строке 66 и состоит из двух циклов для записи результата. The Функция dct\_2d, определенная в строке 23, вызывает dct\_1dфункцию dct\_1d и выполняет транспонирование.

Наконец, dct\_1dфункция dct\_1d, определенная в строке 4, использует dct\_coeff\_table и выполняет требуемую функцию, реализуя базовую итеративную форму алгоритма DCT 1D типа II.

На следующем рисунке показана иерархия функций слева, циклы в порядке их выполненияи поток данных справа.

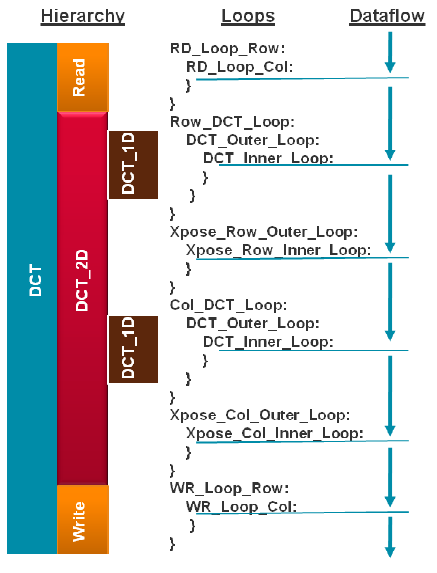


Рисунок 1‑1: Иерархия проектирования и поток данных

## Общий поток

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Шаг 1:  Создание проекта инструмента Vivado HLS |  | Шаг 2:  Запуск  C  моделирования C |  | Шаг 3:  Синтезирование  проекта |  | Шаг 4:  Использование  перспективы анализа |  | Шаг 5:  Выполнение совместного моделирования |

|  |
| --- |
| Шаг 6.  Экспорт  RTL  в качестве IP |

Создание проекта инструмента Vivado HLS Шаг СЛЕДУЮЩИЙ Шаг 1

На этом шаге вы запустите графический интерфейс инструмента Vivado HLS и создадите новый проект для предоставленного дизайна дискретного косинусного преобразования на основе C (DCT).

Существует несколько способов запуска инструмента Vivado HLS. Здесь показаны два наиболее популярных механизмаare shown.

СЛИЯНИЯ 1-1. Запустите инструмент Vivado HLS.

СЛИЯНИЯ 1-СЛИЯНИЯ 1-1. Для Windows 7: Выберите Пуск > >Все программы > >Инструменты проектирования Xilinx > >Vivado 2020.1 > >Vivado HLS 2020.1.

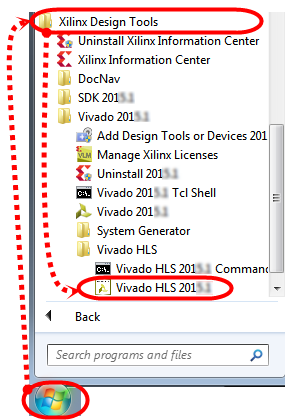


Рисунок 1‑2: Запуск инструмента Vivado HLS

Для Windows 10: Выберите Пуск > >Инструменты проектирования Xilinx > >Vivado HLS 2020.1.

-- ИЛИ --

Дважды щелкните значок быстрого доступа Vivado HLS () на рабочем столе.

Инструмент Vivado HLS откроется в окне приветствия. В окне приветствия windowвы можете создать новый проект, открыть примеры и получить доступ к документации и примерам.

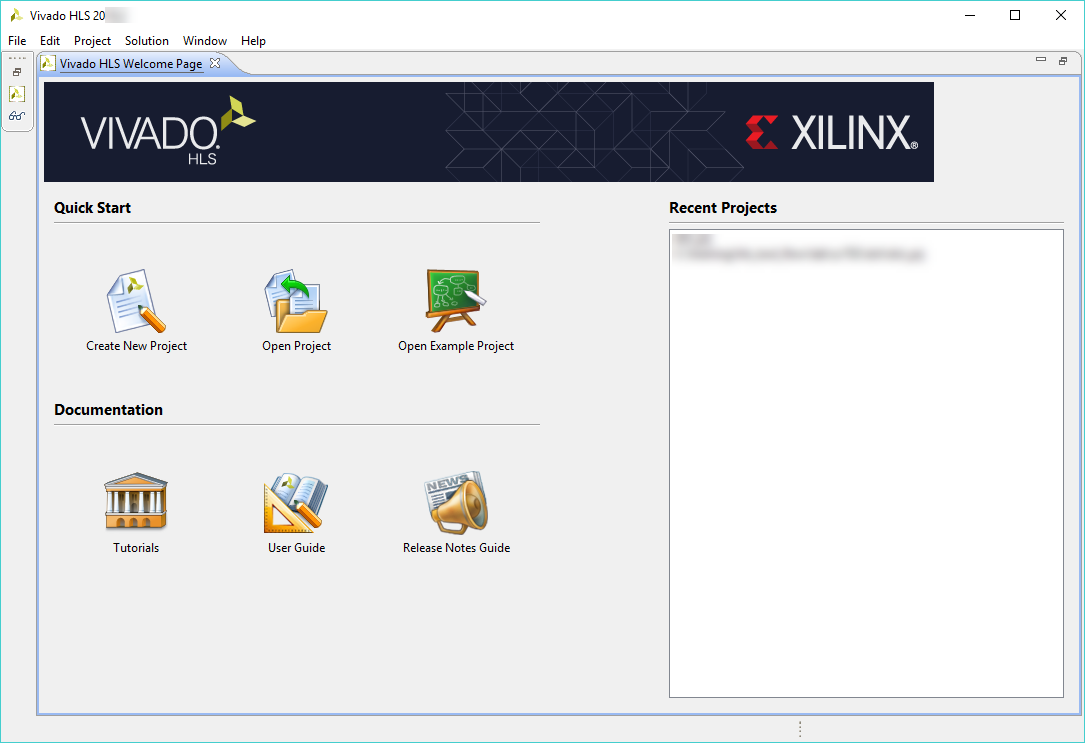


Рисунок 1‑3: Страница приветствия Vivado HLS

Здесь вы научитесь создавать новый проект Vivado HLS с нуля.

СЛИЯНИЯ 1-2. Создайте проект Vivado HLS с именем dct\_prj.

СЛИЯНИЯ 1-СЛИЯНИЯ 2-1. Нажмите Создать новый проект на странице приветствия.

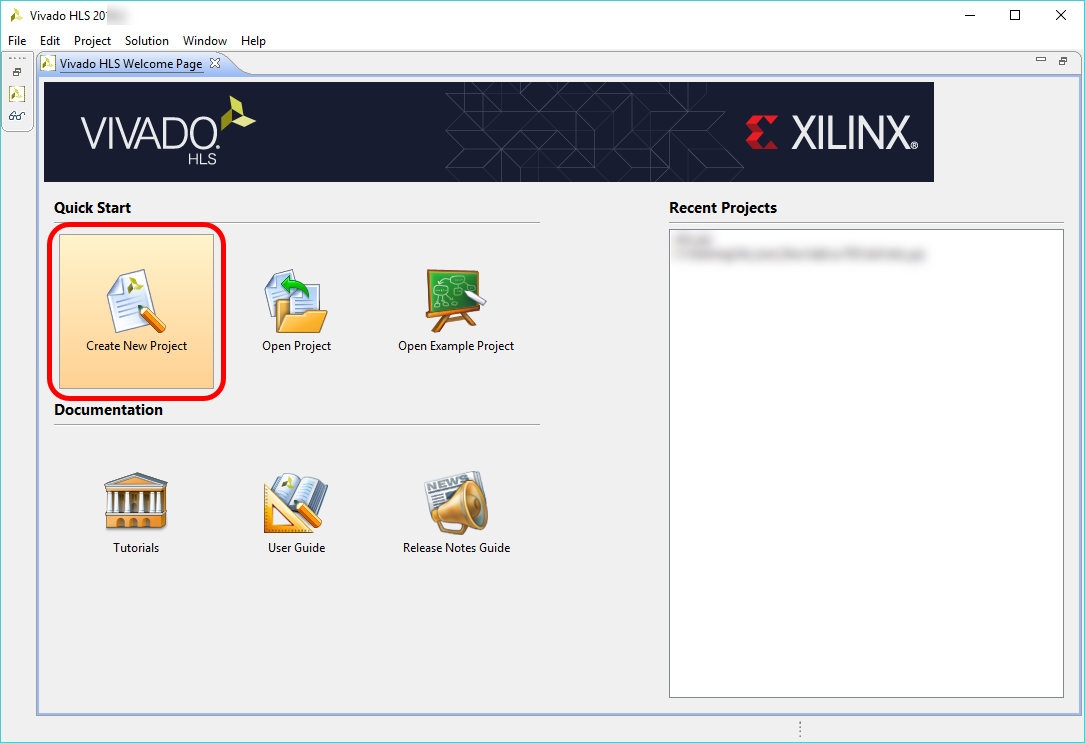


Рисунок 1‑4: Создание нового проекта инструмента Vivado HLS

СЛИЯНИЯ 1-3. В диалоговом окне конфигурации проекта запрашиваются имя и местоположение проекта.

СЛИЯНИЯ 1-СЛИЯНИЯ 3-1. Введите dct\_prj в поле Название проекта (1).

СЛИЯНИЯ 1-СЛИЯНИЯ 3-2. Введите C:\xilinx\_trn\HLS\lab1\_hls\_tool\_flow в поле Местоположение (2).

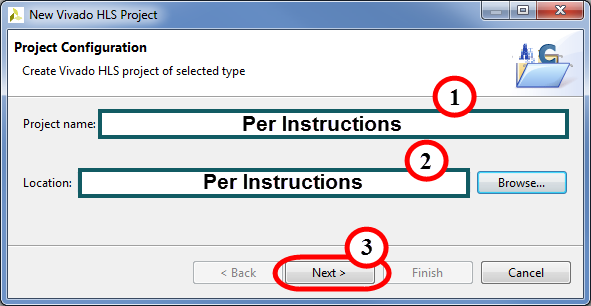


Рисунок 1‑5: Настройка нового проекта HLS

СЛИЯНИЯ 1-3-3. Click Next).

СЛИЯНИЯ 1-4. Откроется диалоговое окно "Добавление/удаление файлов". Здесь вам будет предложено добавить существующие файлы или создать новые источники.

СЛИЯНИЯ 1-4-1. Нажмите Добавить файлы.

Откроется диалоговое окно Открыть.

Примечание: Если в данный момент у вас нет существующих файлов и вы хотите создать новые, нажмите кнопку Создать файл.

1-4-2. Перейдите к C:\xilinx\_trn\HLS\lab1\_hls\_tool\_flow/source.

1-4-3. Выберите dct.c.

Инструмент Vivado HLS автоматически добавляет рабочий каталог (каталог проекта) и любой каталог, содержащий файлы C, добавленные в проект, в путь поиска. Следовательно, файлы заголовков, находящиеся в этих каталогах, автоматически включаются в проект (нет необходимости явно указывать их). Вы должны указать путь ко всем другим файлам заголовков (если таковые имеются), нажав кнопку Изменить CFLAGS.

СЛИЯНИЯ 1-СЛИЯНИЯ 4-4. Нажмите "Открыть", чтобы добавить эти файлы.

Обратите внимание, что на этом этапе вы можете добавить директивы компилятора, специфичные для каждой записи.

СЛИЯНИЯ 1-СЛИЯНИЯ 4-5. Нажмите "Обзор" рядом с верхним функциональным полем.

Откроется диалоговое окно "Выбрать верхнюю функцию", в котором перечислены все функции, доступные из указанных исходных файлов.

1-4-6. Выберите dct (dct.c) из списка и нажмите кнопку ОК.

Примечание: Вы также можете вручную ввести имя верхней функции в поле Верхней функции.

В любой программе на языке Си функция верхнего уровня называется main(). В потоке проектирования Vivado HLS вы можете указать любую подфункцию ниже main() в качестве функции верхнего уровня для синтеза. Вы не можете синтезировать функцию main () верхнего уровня.

Ниже приведены дополнительные правила:

* Only one function is allowedВ качестве функции верхнего уровня для синтеза допускается только одна функция.
* Любые подфункции в иерархии под функцией верхнего уровня для синтеза также синтезируются.
* Если вы хотите синтезировать функции, которые не находятся в иерархии под функцией верхнего уровня для синтеза, необходимо объединить функции в одну функцию верхнего уровня для синтеза.

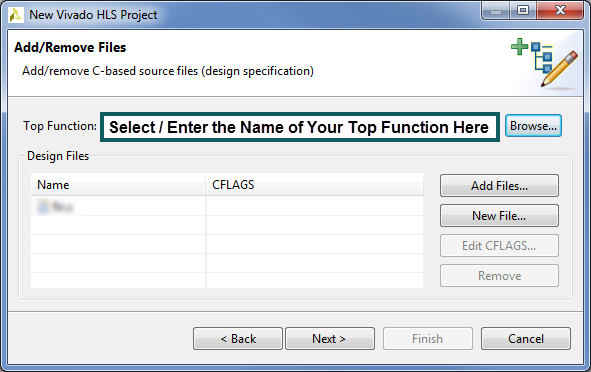


Рисунок 1‑6: Добавление файлов в новый проект Vivado HLS

СЛИЯНИЯ 1-4-7.Нажмите Далее. Click Next.

СЛИЯНИЯ 1-5. Добавьте все существующие файлы тестового стенда.

Если у вас есть (или вам нужны) какие -либо файлы тестового стенда, их можно ввести здесь. Иногда испытательный стенд встроен в синтезируемый файл.

СЛИЯНИЯ 1-5-1. Нажмите Добавить файлы.

СЛИЯНИЯ 1-5-2. Перейдите к C:\xilinx\_trn\HLS\lab1\_hls\_tool\_flow/source.

1-5-3. Выберите dc\_test.c, in.dat, out.golden.dat.

1-5-4. Нажмите кнопку Открыть, чтобы добавить эти файлы.

1-5-5. Нажмите кнопку Далее.

СЛИЯНИЯ 1-6. Наконец, пришло время указать некоторые физические параметры конструкции.

По умолчанию решение 1 заполняется в поле Имя решения. Никаких изменений не требуется.

1-6-1. Установите тактовый период в 10.

Вы можете оставить поле Неопределенности пустым.

форм 1-форм 6-2. Нажмите кнопку Обзор, чтобы выбрать деталь или плату.

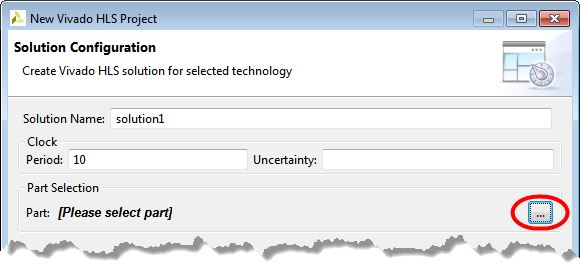


Рисунок 1‑7: Поиск кнопки просмотра доски

СЛИЯНИЯ 1-6-3. Щелкните Детали , как показано ниже.

1-6-4. Enter xa7a12tcsg325-1q (МОЖНО ВЫБРАТЬ ЛЮБУЮ МИКРОСХЕМУ) in the Search field.

СЛИЯНИЯ 1-6-7. Нажмите Готово.

Вы увидите созданный проект на вкладке Проводник.

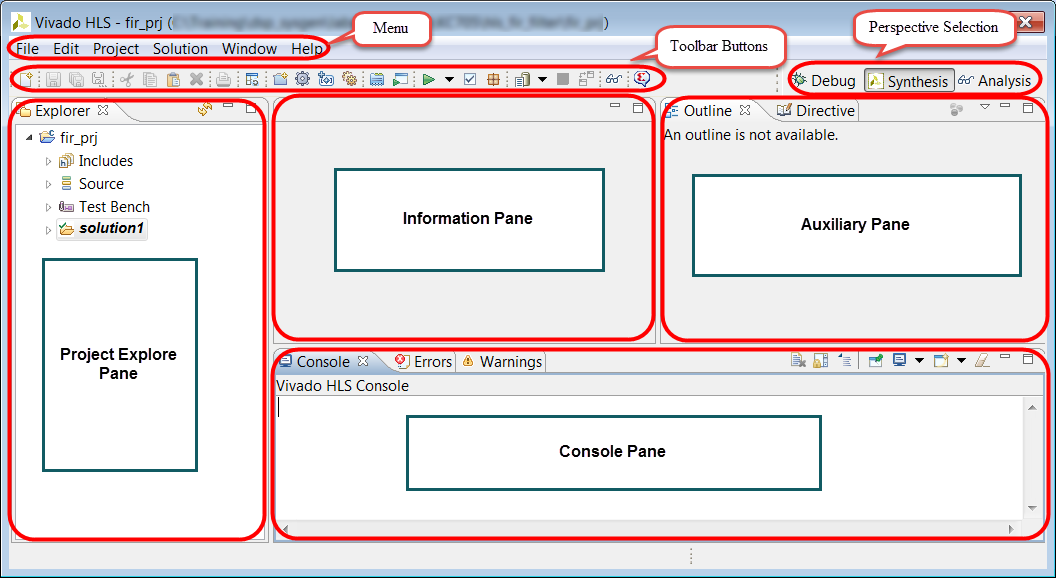


Рисунок 1‑9: Vivado HLS с недавно созданным проектом

Графический интерфейс инструмента Vivado HLS состоит из различных панелей для продолжения работы над разработкой. Вы увидите созданный проект в представлении Проводника. Разверните различные подпапки, чтобы просмотреть записи в каждой подпапке. Исходная папка состоит из исходных файлов, связанных с проектом, а папка тестового стенда состоит из файлов тестового стенда, связанных с проектом.

Запуск шага моделирования C Step 2

После создания проекта следующим шагом является проверка правильности функции C, прежде чем приступить к синтезу проекта. На этом шаге вы проверите конструкцию с помощью предоставленного испытательного стенда с самоконтролем.

2-1. Ознакомьтесь с предоставленными исходными файлами и файлами тестового стенда.

2-1-1. Разверните исходную папку на панели Обозреватель проектов.

2-1-2. Дважды щелкните dct.c, чтобы открыть файл.

Это откроет исходный файл на панели информации.

2-1-3. Просмотрите код и структуры данных.

2-1-4. Разверните Включает > >C:\xilinx\_trn\HLS\lab1\_hls\_tool\_flow\source\source папка на панели обозревателя проектов.

2-1-5. Дважды щелкните dct.h, чтобы открыть файл заголовка.

2-1-6. Просмотрите содержимое файла заголовка.

2-1-7. Разверните папку Тестовый стенд на панели Обозреватель проектов.

2-1-8. Дважды щелкните dc\_test. c, чтобы открыть его в области информации.

Этот тестовый стенд является стендом для самоконтроля ; т. е. Вычисленный вывод сравнивается с эталонным золотым выводом и возвращает либо пройденный, либо неудачный.

2-2. Имитируйте дизайн инструмента Vivado HLS.

2-2-1. Выберите Проект > >Запустить моделирование C или щелкните значок Запустить моделирование C ().

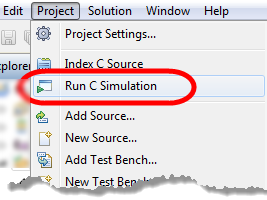


Рисунок 1‑10: Запуск моделирования C

Откроется диалоговое окно Запустить моделирование на языке C.

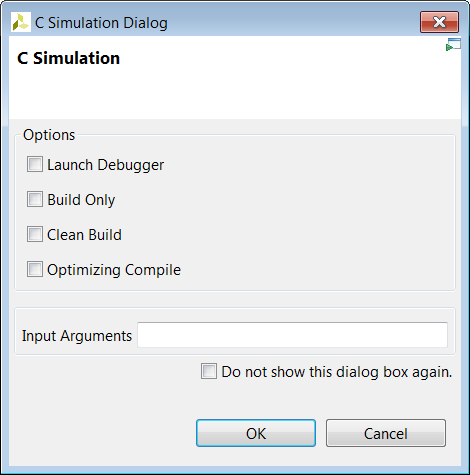


Рисунок 1‑11: Диалоговое окно C Моделирование

Каждый из параметров управляет тем, как выполняется моделирование:

* Запустите отладчик: После компиляции автоматически откроется перспектива отладки, в которой вы сможете выполнить пошаговое выполнение кода.
* Только сборка: Компилирует код C, но моделирование не выполняется.
* Чистая сборка: Удаляет все существующие исполняемые и объектные файлы перед компиляцией кода.
* Оптимизация компиляции: Используется опция gcc/g++ -O (нет информации об отладке, и это взаимоисключает параметры отладки; это может выполняться быстрее, но разница не существенна).

2-2-2. Выберите параметры по умолчанию (т. Е. Ничего не выберите).

2-2-3. Нажмите кнопку ОК.

Журнал моделирования будет отображаться на панели редактора.

2-3. Просмотрите отчет моделирования.

Информацию, полученную с помощью инструмента Vivado HLS, можно найти в двух местах, оба из которых описаны здесь.

Первое-это окно консоли, в котором сообщается не только о результатах, полученных с помощью моделируемого кода, но и обо всех сообщениях механизма моделирования. Журнал моделирования содержит только несколько сообщений механизма моделирования и выходные данные имитируемого кода.

СЛИЯНИЯ 2-СЛИЯНИЯ 3-1. Выберите вкладку Консоль в нижней части графического интерфейса инструмента.

Возможно, вам потребуется прокрутить, чтобы просмотреть все выходные данные, полученные в результате моделирования.

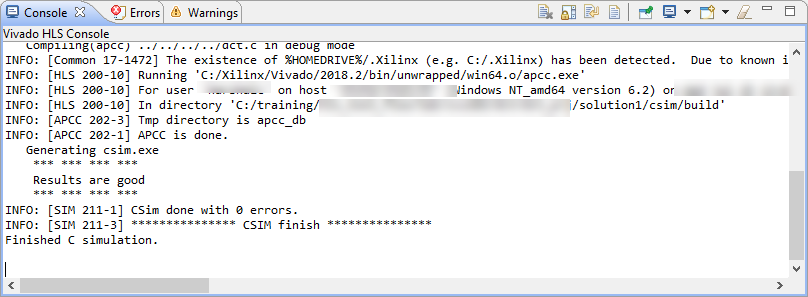


Рисунок 1‑12: Пример Вывода После Моделирования

Другое расположение, описанное ниже, предоставляет только несколько сообщений модуля моделирования и вывод имитированного кода. Обычно это открывается после завершения моделирования; однако, если вам нужно получить к нему доступ после закрытия области журнала, вот как получить доступ к отчету моделирования.

2-3-2. Разверните dct\_prj > >решение 1 > >csim > >отчет на панели проводника.

2-3-3. Дважды щелкните имя файла журнала, чтобы открыть его на панели редактора.

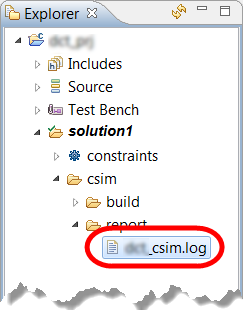


Рисунок 1‑13: Поиск файла журнала моделирования

Вы должны увидеть сообщение "Результаты хорошие" в файле журнала моделирования и в области консоли. Если вы не видите этого сообщения, обратитесь за помощью к своему инструктору.

Синтезирование шага проектирования Step 3

На этом шаге вы синтезируете дизайн с помощью настроек по умолчанию инструмента Vivado HLS и проанализируете, сколько ресурсов используется для реализации дизайна языка C.

3-1. Синтезируйте дизайн.

3-1-1. Выберите Решение > >Запустить синтез C > >Активное решение или щелкните значок Запустить синтез в строке меню.

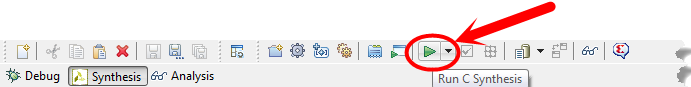


Рисунок 1‑14: Запуск синтеза

Этот параметр синтезирует текущее выбранное решение.

Все решения (или выбранные решения) можно синтезировать с помощью раскрывающегося меню рядом со значком синтез. Вы можете синтезировать все решения или синтезировать выбранные решения в дополнение к решениям по умолчанию.



Рисунок 1‑15: Параметры для синтеза

Когда синтез завершится, отчет о синтезе будет отображаться на панели информации.

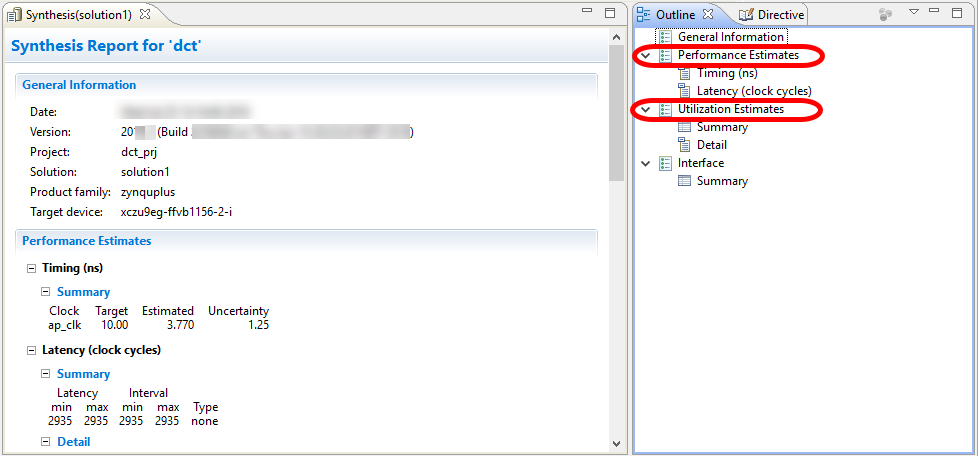


Рисунок 1‑16: Обобщающий отчет [Пример]

В сводном отчете показаны оценки производительности и площади, а также предполагаемая задержка при проектировании.

СЛИЯНИЯ 3-СЛИЯНИЯ 1-2. Щелкните Оценки производительности и оценки использования на панели "Схема", чтобы ответить на следующий вопрос.

Вопрос 1

Запишите следующие сведения из сводного отчета:

* Расчетная тактовая частота:
* Задержка в худшем случае:
* Номер BRAM\_18K:
* Количество используемых DSP48E:
* Количество используемых FFS:
* Количество используемых LUT:

СЛИЯНИЯ 3-СЛИЯНИЯ 1-3. Выберите "Интерфейс" > > "Сводка " На панели "Схема".

В отчете также показаны сигналы интерфейса верхнего уровня, генерируемые инструментами.

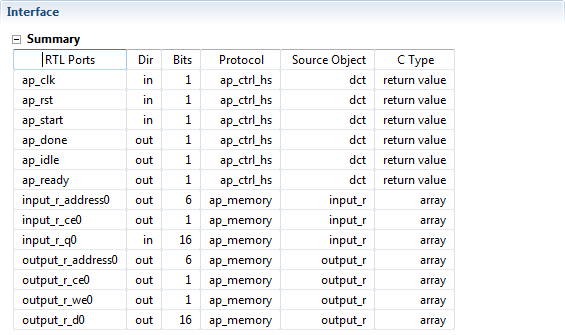


Рисунок 1‑17: Генерируемые интерфейсные сигналы

Вы можете видеть, что ap\_clk и ap\_rst добавляются автоматически. ap\_start, ap\_doneи ap\_idle-это сигналы верхнего уровня, используемые в качестве сигналов подтверждения, чтобы указать, когда проект может принять следующую команду вычисления (ap\_idle), когда начинается следующее вычисление (ap\_start) и когда вычисление завершено (ap\_done). Другие сигналы генерируются на основе самого дизайна.

СЛИЯНИЯ 3-1-4. Выберите вкладку Консоль.

Журнал синтеза доступен в консоли Vivado HLS.

Обратите внимание, что при расширении папки Решения1 > >Syn в представлении Проводника будут показаны reportподпапки report, systemc, verilogи vhdl, в которых доступны файлы отчета и созданные исходные файлы (VHDL, Verilog, заголовок и cpp). Дважды щелкнув любую из этих записей, вы откроете соответствующий файл на панели информации.

Также обратите внимание, что целевой дизайн имеет иерархические функции, и также создаются отчеты, соответствующие функциям более низкого уровня (в этом примере dct\_1d2\_csynth.rpt и dct\_2d\_csynth.rpt в дополнение к dct\_csynth.rpt). По умолчанию отчет для функции верхнего уровня отображается в области информации после завершения синтеза.

Анализ дизайна С использованием анализа перспективного шага 4

Перспектива анализа используется после завершения синтеза для детального анализа проекта. Эта перспектива содержит значительно больше деталей, чем Сводный доклад.

4-1. Переключитесь на перспективу анализа и поймите поведение проекта.

слияния 4-СЛИЯНИЯ 1-1. Выберите Решение > >Открыть перспективу анализа или щелкните значок AnalysisПерспективы анализа (), чтобы открыть средство просмотра анализа.

Перспектива анализа состоит из пяти панелей, как показано ниже. Обратите внимание, что иерархии модулей и циклов по умолчанию отображаются в развернутом виде.

Панель Иерархии модулей предоставляет обзор всего дизайна RTL:

* Это представление может перемещаться по всей иерархии проектирования.
* Панель Иерархия модулей показывает вклад ресурсов и задержек для каждого блока в иерархии RTL.

Панель Иерархии модулей отображает информацию о производительности и области для всего проекта и может использоваться для навигации по иерархии. Видна панель "Профиль производительности", на которой отображаются сведения о производительности для этого уровня иерархии. Информация в этих двух панелях аналогична информации, рассмотренной ранее в сводном отчете.

Представление производительности [Просмотр расписания(решение 1)] также отображается в правой боковой панели. Это представление показывает, как операции в этом конкретном блоке планируются в тактовые циклы.

* В левой колонке перечислены ресурсы.
* В верхней строке перечислены состояния управления (от C0 до C5) в дизайне. Состояния управления-это внутренние состояния, используемые инструментом Vivado HLS для планирования операций в тактовые циклы. Существует тесная корреляция между контрольными состояниями в RTL FSM, но нет однозначного сопоставления.

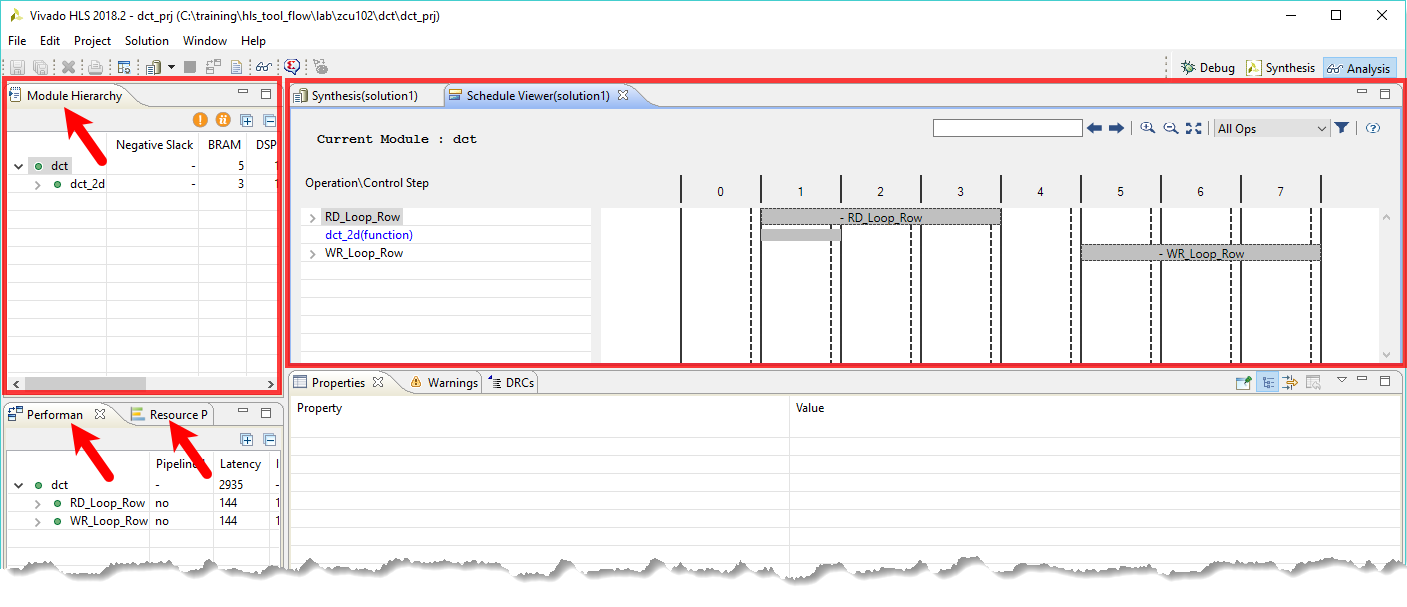


Рисунок 1‑18: Пример анализа перспективных панелей

СЛИЯНИЯ 4-2. Проанализируйте производительность модуля dct.

Модуль dct имеет три основных ресурса:

* Цикл называется RD\_Loop\_Row.
* Подблок под названием dct\_2d.
* Цикл под названием WR\_Loop\_Row.

4- РАЗДЕЛ Шаг X2-1. Щелкните цикл RD\_Loop\_Row и выберите подцикл RD\_Loop\_Col.

СЛИЯНИЯ 4-СЛИЯНИЯ 2-2. Выберите tmp\_22(+) (1).

СЛИЯНИЯ 4-СЛИЯНИЯ 2-3. Щелкните правой кнопкой мыши выделенное поле и выберите "Перейти к источнику" (2).

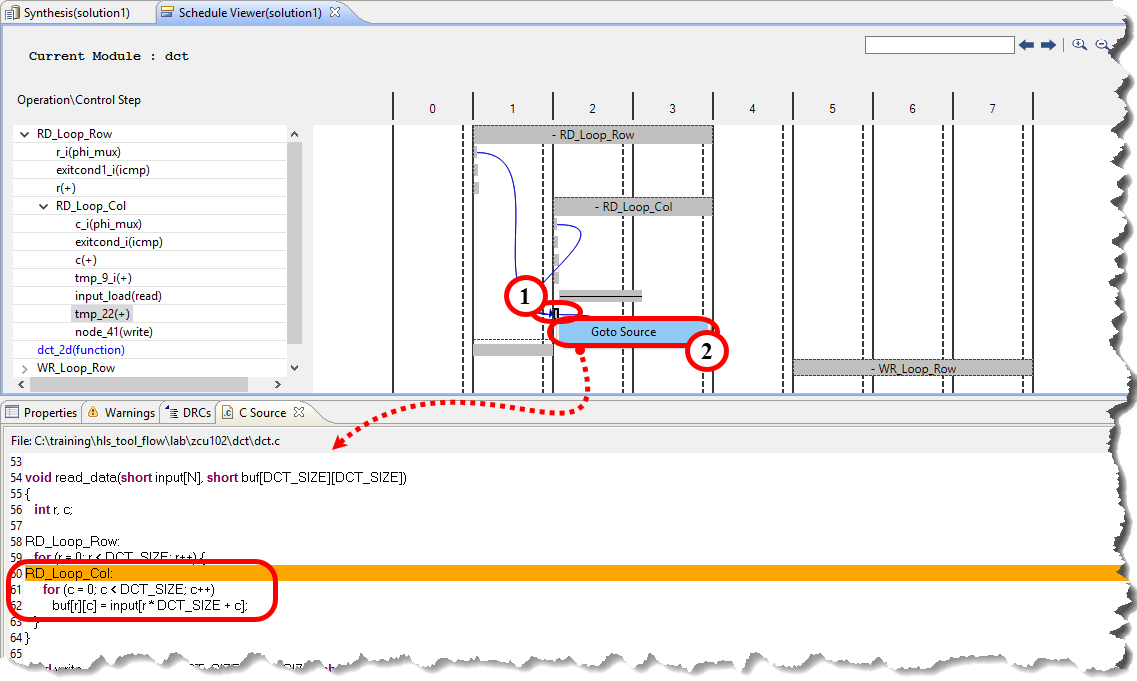


Рисунок 1‑19: Производительность работы цикла постоянного тока

Информация, представленная в представлении Расписания, объясняется ниже путем просмотра первого набора ресурсов, которые должны быть выполнены (RD\_Loop\_Rowцикл RD\_Loop\_Row):

* Проектирование начинается в 0состоянии 0.
* Затем он начинает выполнять логику в цикле RD\_Loop\_Row.
* Примечание: В первом состоянии цикла проверяется условие выхода и выполняется операция добавления.
* Цикл выполняется в трех состояниях: 1, 2 и 3.

СЛИЯНИЯ 4-2-4. Просмотрите информацию на панели "Профиль производительности".

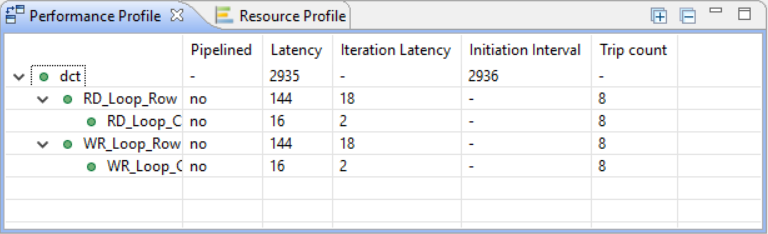


Рисунок 1‑20: Панель профиля производительности

Панель Профиля производительности показывает, что в этом цикле количество отключений равно восьми. Поэтому он повторяется вокруг этих трех состояний восемь раз.

Панель Профиля производительности показывает, что выполнение цикла RD\_Loop\_Row занимает 144 такта:

* Один цикл в начале цикла RD\_Loop\_Row.
* Для выполнения всех операций цикла RD\_Loop\_Col требуется 16 тактов.
* Плюс тактовый цикл для возврата к началу цикла RD\_Loop\_Row в общей сложности 18 циклов на итерацию цикла.
* Восемь итераций по 18 циклов - вот почему для завершения требуется 144 такта.

В цикле RD\_Loop\_Col вы можете видеть, что есть несколько сумматоров, двухцикловая операция чтения и операция записи.

СЛИЯНИЯ 4-СЛИЯНИЯ 2-5. Щелкните цикл WR\_Loop\_Row и щелкните подцикл WR\_Loop\_Col.

СЛИЯНИЯ 4-СЛИЯНИЯ 2-6. Выберите tmp\_23(+) (1).

СЛИЯНИЯ 4-СЛИЯНИЯ 2-7. Щелкните правой кнопкой мыши выделенное поле и выберите GotoИсточник перехода (2).

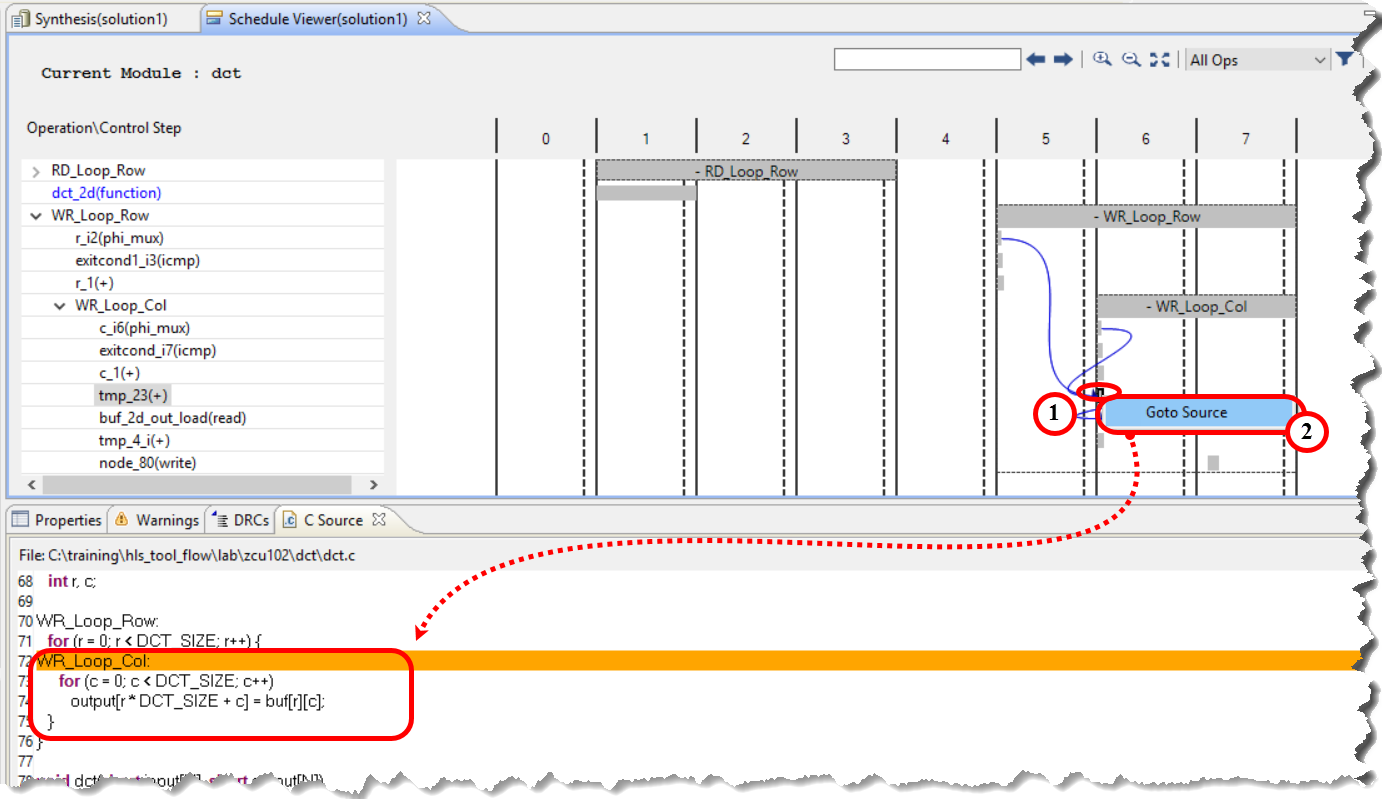


Рисунок 1‑21: Корреляция исходного кода C

Вы можете видеть, что операция записи реализует запись данных в массив buf из переменной входного массива.

СЛИЯНИЯ 4-3. Проанализируйте использование ресурсов для проектирования.

СЛИЯНИЯ 4-СЛИЯНИЯ 3-1. Выберите вкладки Иерархия модулей и Профиль ресурсов, как показано ниже.

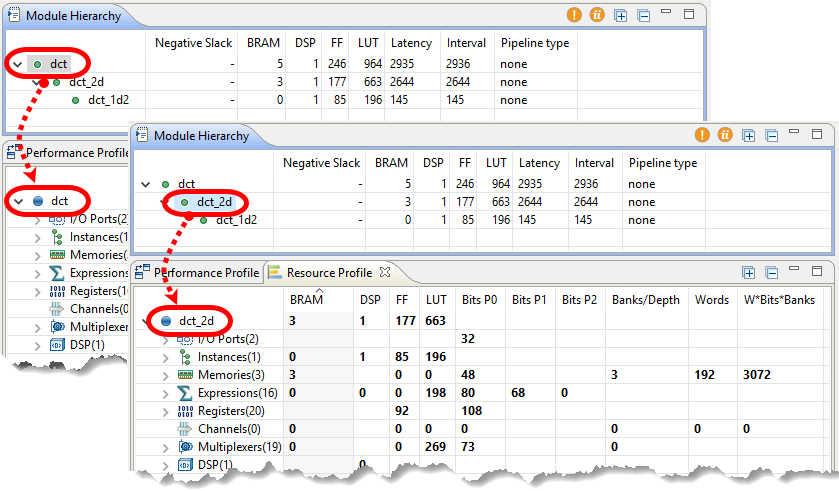


Figure 1‑22: Analysis Perspective – Resource Usage

На вкладке Профиль ресурсов отображаются ресурсы, используемые на этом уровне иерархии. В этом дизайне вы можете видеть, что большая часть ресурсов связана с экземплярами—блоками, которые создаются внутри этого блока. В соответствии с функцией, выбранной на вкладке Иерархия модулей, вы можете просматривать ресурсы, используемые функцией, на вкладке Профиль ресурсов.

СЛИЯНИЯ 4-СЛИЯНИЯ 3-2. Щелкните значок Synthesisперспективы синтеза (), чтобы вернуться к представлению синтеза.

Выполнение этапа совместного моделирования C/RTL , СЛЕДУЮЩИЙ шаг 5

Теперь, когда вы выполнили высокоуровневый синтез в проекте C, вы выполните совместное моделирование RTL на сгенерированном RTL с помощью испытательного стенда C.

Запустите совместное моделирование C/RTL, выбрав Verilog и пропустив VHDL. Убедитесь, что моделирование прошло успешно.

СЛИЯНИЯ 5-1. CosimulateИмитируйте дизайн инструмента Vivado HLS.

СЛИЯНИЯ 5-СЛИЯНИЯ 1-1. Выберите Решение > >Запустить совместное моделирование C/RTL или щелкните значок Запустить Cosimulationсовместное моделирование C/RTL ().

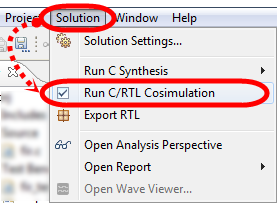


Рисунок 1‑23: Запуск из меню

Откроется диалоговое окно Запустить совместное моделирование C/RTL.

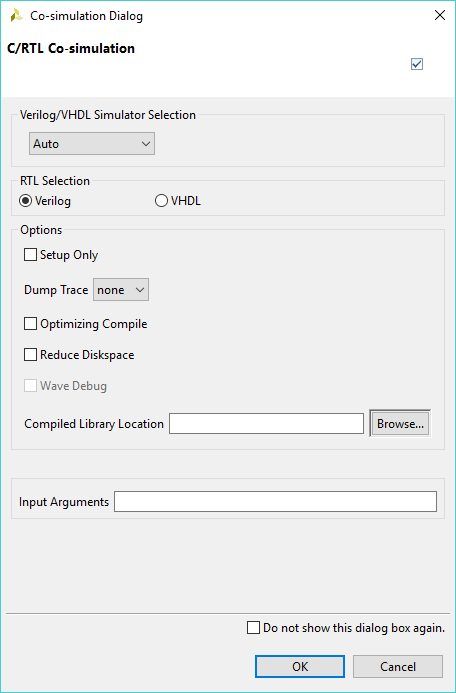


Рисунок 1‑24: Диалоговое окно совместного моделирования

Каждый из параметров определяет, как выполняется моделирование C/RTL co:

* Выбор RTL: Выберите смоделированный RTL (Verilog/VHDL).
* Только настройка: При этом создаются все файлы (оболочки, адаптеры и сценарии), необходимые для запуска моделирования, но симулятор не выполняется.
* Трассировка дампа: Во время проверки RTL файлы трассировки можно сохранить и просмотреть с помощью соответствующего средства просмотра. При выборе этого параметра файл трассировки будет сохранен в папке <решение><решение>/sim/<RTL><RTL>.
* Оптимизирующая компиляция: Это гарантирует, что is usedдля компиляции тестового стенда C используется высокий уровень оптимизации . Использование этой опции увеличивает время компиляции, но моделирование выполняется быстрее.
* Сокращение дисковогопространства: сохраняет результаты для всех транзакций перед выполнением моделирования RTL. В некоторых случаях это может привести к созданию больших файлов данных. Эту опцию можно использовать для выполнения одной транзакции за раз и уменьшения объема дискового пространства, необходимого для файла. Если функция выполняется N раз на стенде C, reduce\_diskspaceопция reduce\_diskspace гарантируетвыполнение s N отдельных RTL-симуляций. Это приводит к замедлению процесса моделирования.
* Расположение скомпилированной библиотеки: Указывает местоположение скомпилированной библиотеки для стороннего RTL-симулятора.
* Входные аргументы: Это позволяет указать любые аргументы, требуемые испытательным стендом.

СЛИЯНИЯ 5-СЛИЯНИЯ 1-2. Выберите **параметры**по oумолчанию (т. Е. Ничего не выбирайте).

СЛИЯНИЯ 5-1-3. Нажмите Кнопку ОК.

Журнал моделирования будет отображаться на панели редактора.

СЛИЯНИЯ 5-2. Просмотрите Cosimulationотчет о совместном моделировании.

Информацию, полученную с помощью инструмента Vivado HLS, можно найти в двух местах, оба из которых описаны здесь.

Первое-это окно консоли, в котором сообщается не только о результатах, полученных с помощью моделируемого кода, но и обо всех сообщениях механизма моделирования. Журнал моделирования содержит только несколько сообщений механизма моделирования и выходные данные имитируемого кода.

СЛИЯНИЯ 5-СЛИЯНИЯ 2-1. Выберите вкладку Консоль в нижней части графического интерфейса инструмента.

Возможно, вам потребуется прокрутить страницу, чтобы просмотреть все результаты, полученные врезультате совместного моделирования.

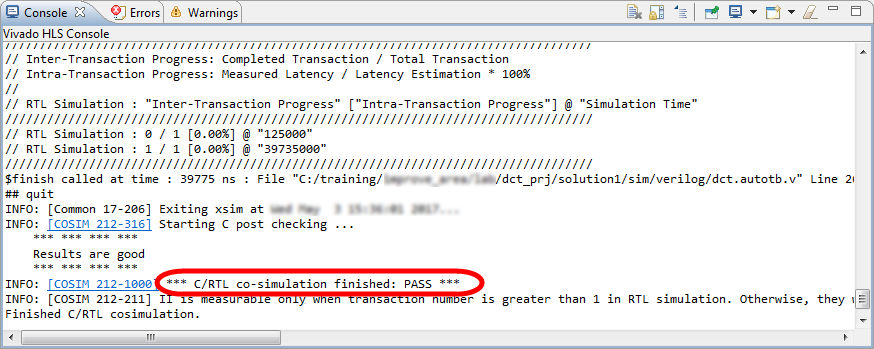


Рисунок 1‑25: Пример вывода После совместного моделирования C/RTL

Другое расположение, описанное ниже, содержит только несколько сообщений механизма моделирования и выходные данные имитируемого кода. Обычно он открывается после завершения моделирования; однако, если вам нужно получить доступ к нему после закрытия панели журнала, вот как получить доступ к отчету о моделировании.

5-2-2. Разверните dct\_prj > >решение 1 > >sim > >отчет на панели проводника.

5-2-3. Дважды щелкните имя файла журнала, чтобы открыть его на панели редактора.

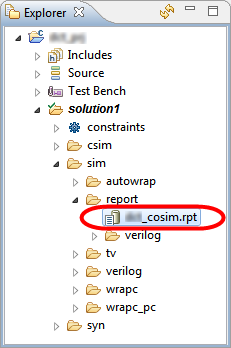


Рисунок 1‑26: Поиск файла журнала совместного моделирования

Отчет о копировании в формате HTML будет отображаться в главной области просмотра.

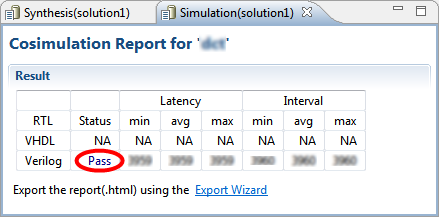


Рисунок 1‑27: CosimulationОтчет о совместном копировании – HTML

Вы можете быстро проверить cosimulationстатус сокрытия здесь.

Этот процесс займет несколько минут. После завершения совместного использования C/RTL has been completed, the Cosimulationотчет о совместном использовании будет доступен на панели информации, включая информацию о задержке.

Кроме того, на вкладке Консоль обратитеResults are Good "внимание, что отображается сообщение "Результаты хорошие".

Экспорт RTL в качестве основного IP -адреса Шаг 6

На этом шаге вы экспортируете RTL в качестве IP-ядра для использования в дизайне верхнего уровня.

СЛИЯНИЯ 6-1. Экспортируйте RTL, выбрав в качестве языка Verilog.

СЛИЯНИЯ 6-СЛИЯНИЯ 1-1. Выберите Решение >> Экспорт RTL.

Кроме того, вы можете нажать кнопку на панели инструментов.

Примечание.Если вы видите диалоговое окно Запроса обратной связи, нажмите кнопку Отмена.

СЛИЯНИЯ 6-СЛИЯНИЯ 1-2. Убедитесь, что каталог IP выбран из раскрывающегося списка "Выбор формата".

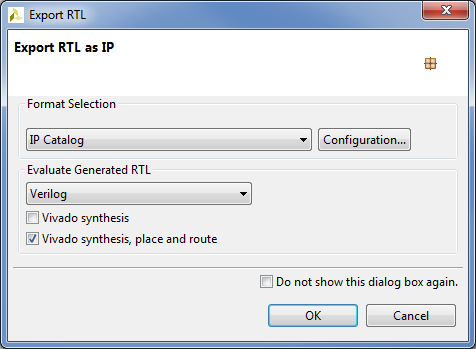


Рисунок 1‑28: Диалоговое окно экспорта RTL

СЛИЯНИЯ 6-СЛИЯНИЯ 1-3. Нажмите "Конфигурация" рядом с раскрывающимся списком выбора формата.

Обратите внимание, что вы можете предоставить информацию об IP-адресе, такую как поставщик, библиотека, версия и описание, в диалоговом окне идентификации IP-адреса.

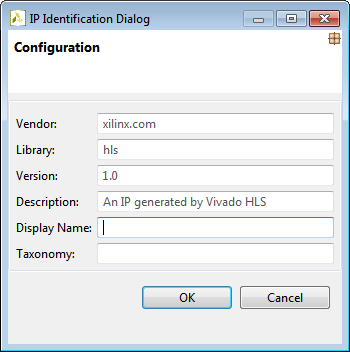


Рисунок 1‑29: Диалоговое окно идентификации IP-адреса на арабском языке.

СЛИЯНИЯ 6-СЛИЯНИЯ 1-4. Нажмите кнопку Отмена в диалоговом окне идентификации IP-адреса.

6-1-5. Убедитесь, что Verilog is selectedв качестве RTL для оценки выбран Verilog.

6-1-6. Выберите параметр Синтез, место и маршрут Vivado.

Это позволит выполнить синтез и реализацию языка RTL Vivado на сгенерированном IP. Внедрение выполняется для оценки и обеспечения уверенности в том, что RTL будет соответствовать своим предполагаемым срокам и областным целям, и что эти результаты не включены в экспортируемый пакет.

6-1-7. Нажмите кнопку ОК в диалоговом окне Экспорт RTL.

Вы можете наблюдать за ходом выполнения на вкладке консоли. По завершении выполнения Отчет о реализации отображается в области сведений.

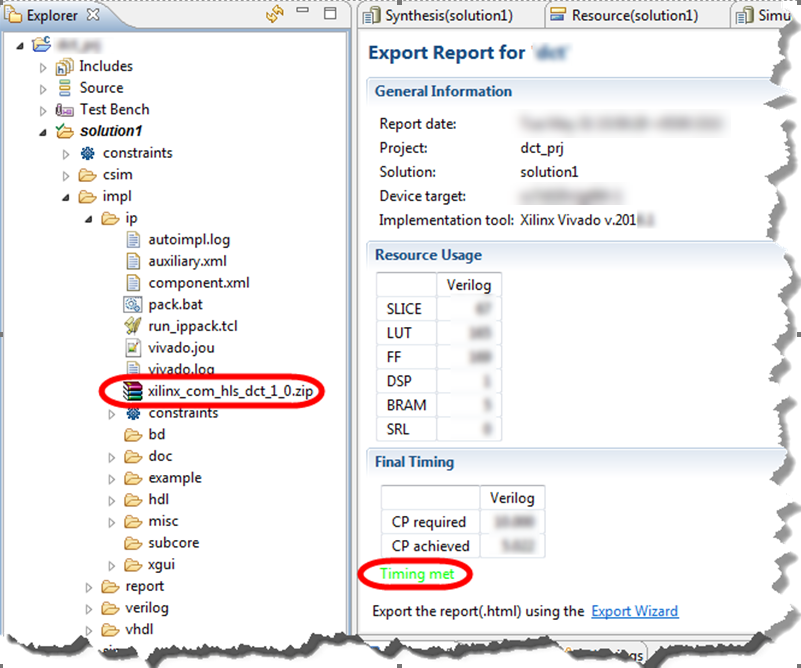


Рисунок 1‑30: Результаты реализации в средстве Vivado HLS

Как только реализация завершится, отчет о реализации откроется в области сведений. Достигнуты окончательные сроки реализации проектаhas been achieved.

Экспортированный IP-адрес доступен в папке <директориярешения>\>impl\ip.

СЛИЯНИЯ 6-СЛИЯНИЯ 1-8. Выберите Файл > >Выход, чтобы закрыть инструмент Vivado HLS.

## Резюме

В этой лаборатории вы узнали, как создать новый проект инструмента Vivado HLS в графическом интерфейсе и выполнить основные этапы (моделирование, синтез, совместное моделирование и экспорт) в потоке HLS Vivado Design Suite.

В следующих лабораториях вы изучите некоторые отчеты о программном обеспечении, определите, как был реализованпроект, и определите, были ли достигнуты цели проектирования по области и производительности.

## Ответы

1. Запишите следующие сведения из сводного отчета:

Расчетная тактовая частота: 3,77

Задержка в наихудшем случае: 2935

Количество BRAM\_18K: 5

Количество используемых DSP48E: 1

Количество используемых FFS: 246

Количество использованных ЛУТ: 964

# 