ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»**

Институт компьютерных наук и технологий

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

**ЛАБОРАТОРНАЯ РАБОТА №2**

**Знакомство с пакетом Vivado**

по дисциплине «Технологии проектирования аппаратных средств компьютерных систем»

Выполнил:

студент группы 3540901/02001

Бараев Д. Р.

Проверил:

А. П. Антонов

Санкт-Петербург 2020

**Оглавление**

[Список иллюстраций: 3](#_Toc55355203)

[Введение: 5](#_Toc55355204)

[Цели лабораторной работы: 5](#_Toc55355205)

[Описание устройства: 5](#_Toc55355206)

[Этапы выполнения лабораторной работы: 6](#_Toc55355207)

[Этап 1. Создание проекта в пакете Vivado 6](#_Toc55355208)

[Этап 2. Анализ файлов с описанием проекта 6](#_Toc55355209)

[Этап 3. Анализ структуры RTL описания проекта 7](#_Toc55355210)

[Этап 4. Назначение выводов микросхемы FPGA 11](#_Toc55355211)

[Этап 5. Синтез проекта и анализ результатов 13](#_Toc55355212)

[Этап 6. Анализ временных параметров синтезированного проекта 16](#_Toc55355213)

[Этап 7. Изменение временных параметров, синтез и анализ 20](#_Toc55355214)

[Этап 8. Анализ аппаратных затрат и потребляемой мощности 26](#_Toc55355215)

[Этап 9. Создание контрольной точки 27](#_Toc55355216)

[Этап 10. Изменение настроек синтеза и анализ результатов 27](#_Toc55355217)

[Этап 11. Загрузка контрольной точки 29](#_Toc55355218)

[Вывод: 31](#_Toc55355219)

Список иллюстраций:

[Рисунок 1 - Настройки проекта 6](#_Toc55355152)

[Рисунок 2 – Иерархия модулей с описанием проекта 7](#_Toc55355153)

[Рисунок 3 – Содержимое файла uart\_led 7](#_Toc55355154)

[Рисунок 4 - Структурная схема RTL описания верхнего уровня 8](#_Toc55355155)

[Рисунок 5 – Объявление meta\_harden\_btn\_i0 8](#_Toc55355156)

[Рисунок 6 - Структурная схема проекта 8](#_Toc55355157)

[Рисунок 7 – Окно DRC 9](#_Toc55355158)

[Рисунок 8 – Изменения в назначения проекта для исправления важного предупреждения 10](#_Toc55355159)

[Рисунок 9 – Два сообщения о критических ошибках 10](#_Toc55355160)

[Рисунок 10 - Сохранения назначений в файл uart\_led\_conf.xdc 11](#_Toc55355161)

[Рисунок 11 - Содержимое файла uart\_led\_conf.xdc 11](#_Toc55355162)

[Рисунок 12 - Режим I/O Planning 12](#_Toc55355163)

[Рисунок 13 - Назначение выводов для платы 12](#_Toc55355164)

[Рисунок 14 - Текстовый редактор с файлом uart\_led\_conf.xdc. 13](#_Toc55355165)

[Рисунок 15 - Сообщение об отсутствие нарушений 13](#_Toc55355166)

[Рисунок 16 - Таблица использованных/доступных ресурсов и процент их использования 14](#_Toc55355167)

[Рисунок 17 - Структурную схему синтезированного проекта 15](#_Toc55355168)

[Рисунок 18 - Текстовый редактор с описанием элемента 16](#_Toc55355169)

[Рисунок 19 - Содержимое файла uart\_led\_timing.xdc 16](#_Toc55355170)

[Рисунок 20 – Окно Timing 17](#_Toc55355171)

[Рисунок 21 – Список из 8 цепей, для которых временные параметры не были достигнуты 17](#_Toc55355172)

[Рисунок 22 – Детальный отчет о полученных и заданных параметрах 18](#_Toc55355173)

[Рисунок 23 – Структурная схема элементов, образующих Path21 19](#_Toc55355174)

[Рисунок 24 - Пути тактового сигнала 20](#_Toc55355175)

[Рисунок 25 - Детальная информация о предупреждении 20](#_Toc55355176)

[Рисунок 26 - Запуск Timing Constraints Wizard 21](#_Toc55355177)

[Рисунок 27 - Изменение периода тактового сигнала virtual\_clock 21](#_Toc55355178)

[Рисунок 28 – Set Input Delay 22](#_Toc55355179)

[Рисунок 29 - Результаты синтеза 23](#_Toc55355180)

[Рисунок 30 - Результаты после завершения временного анализа 23](#_Toc55355181)

[Рисунок 31 - Список из цепей, для которых параметры были достигнуты, но они имеют минимальный зазор 24](#_Toc55355182)

[Рисунок 32 - Детальный отчет о параметрах 25](#_Toc55355183)

[Рисунок 33 - Изменения в файл uart\_led\_timing.xdc 25](#_Toc55355184)

[Рисунок 34 - Временные параметры проекта 26](#_Toc55355185)

[Рисунок 35 - Power 26](#_Toc55355186)

[Рисунок 36 - Write Checkpoint 27](#_Toc55355187)

[Рисунок 37 - Результаты синтеза 28](#_Toc55355188)

[Рисунок 38 - Создание контрольной точки 29](#_Toc55355189)

[Рисунок 39 - Загрузка контрольной точки 29](#_Toc55355190)

[Рисунок 40 - Окно Vivado с загруженной контрольной точкой 30](#_Toc55355191)

[Рисунок 41 - Схемное представление результатов синтеза для загруженной точки 30](#_Toc55355192)

[Рисунок 42 - Результаты временного анализа результатов 31](#_Toc55355193)

Введение:

Лабораторная работа демонстрирует процедуру синтеза в рамках пакета Vivado и эффекты от изменения некоторых настроек процедуры синтеза. Создаваемый и синтезируемый в рамках лабораторной работы проект ориентирован на микросхемы семейства Artix7, использованные на платах Basys3 или Nexys4 DDR.

В описании лабораторной работы:

* ***{sources}***означает: C:\xilinx\_trn\FPGA\_Vivado\_Int\labs\_src
* ***{labs}*** означает: C:\xilinx\_trn\FPGA\_Vivado\_Int\labs
* ***{boards}*** означает: C:\xilinx\_trn\FPGA\_Vivado\_Int\boards

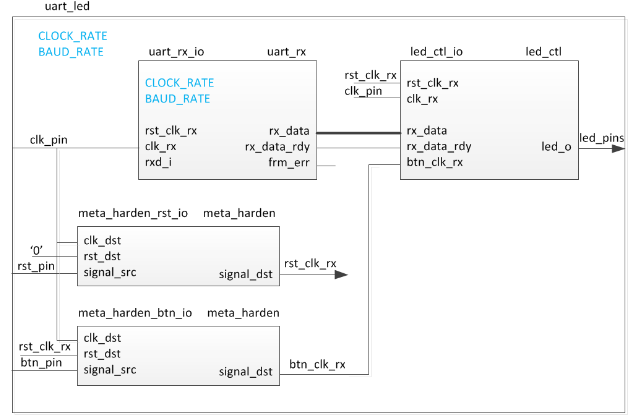
Цели лабораторной работы:

После выполнения лабораторной работы вы узнаете, как:

* назначить выводы микросхемы;
* задать параметры проекта;
* синтезировать проект;
* анализировать результаты синтеза;
* проводить временной анализ;
* анализировать и изменять задаваемые в XDC файле временные параметры;
* создавать и загружать контрольные точки;
* проводить анализ аппаратных затрат и потребляемой мощности после процедуры синтеза.

Описание устройства:

Структурная схема проекта приведена на рисунке ниже.



Этапы выполнения лабораторной работы:

Этап 1. Создание проекта в пакете Vivado

1. Запустите пакет **Vivado 2020.1** (или Vivado WebPack 2020.1)
2. Создайте новый проект:

* Папка для проекта {labs}.
* Имя проекта lab2.
* Для описания проекта и теста используется язык Verilog
* Перейдите к папке *{sources}*\lab2, в которой выберите файлы: "led\_ctl.v", "meta\_harden.v", "uart\_baud\_gen.v", "uart\_led.v", "uart\_rx.v", "uart\_rx\_ctl.v"(это описания на языке Verilog алгоритма работы создаваемого проекта).

1. Следует выбрать конкретную микросхему FPGA для реализации проекта.

* Тип микросхемы: XC7A100TCSG324-1 (если используется плата Nexys4 DDR)*.*

1. После создания проекта проверьте, что все настройки заданы правильно – соответствуют приведенному ниже рисунку.

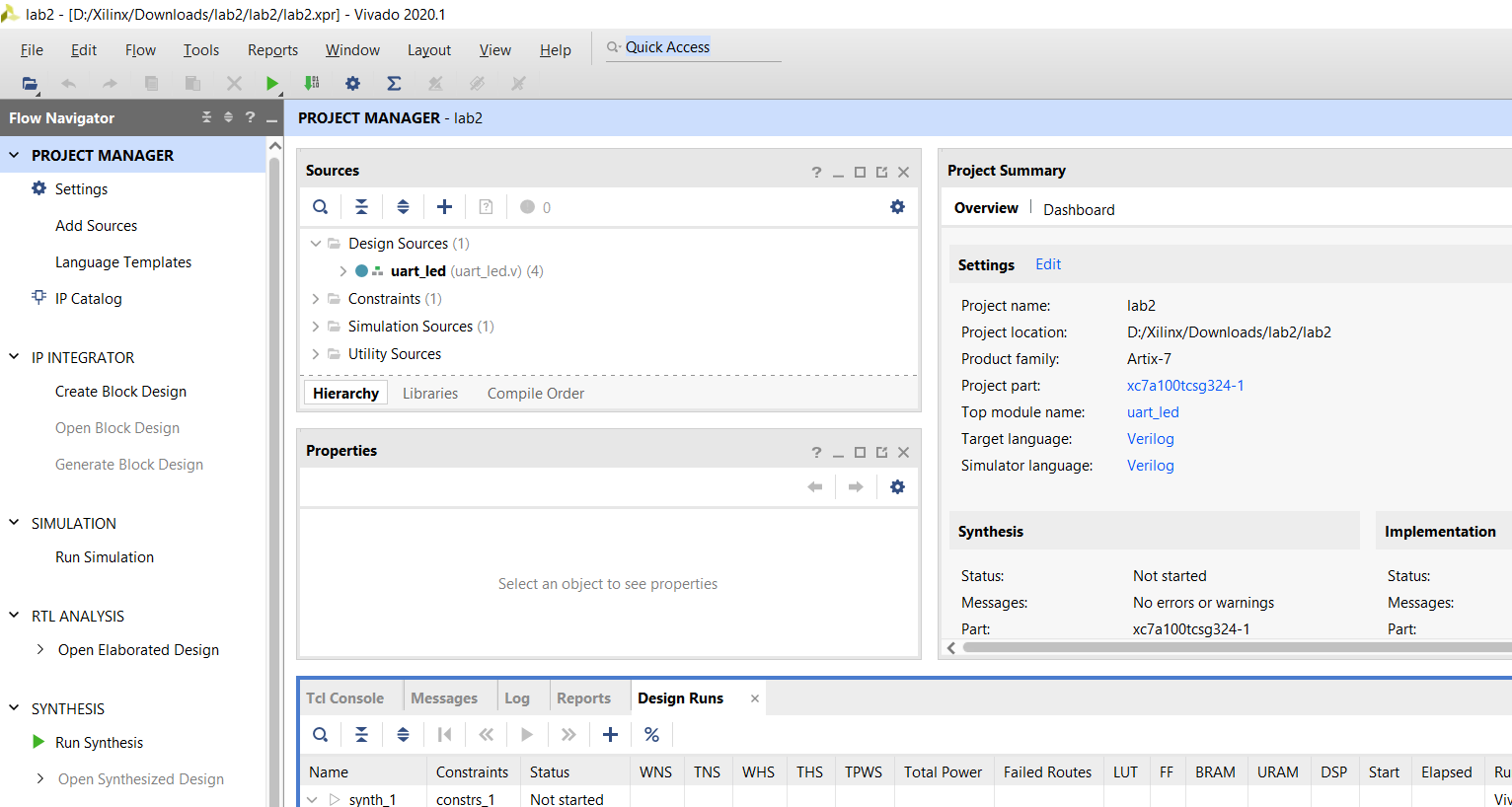


Рисунок 1 - Настройки проекта

Этап 2. Анализ файлов с описанием проекта

1. В разделе **Sources**, в папке **Design Sources**, щелчком по стрелке около модуля uart\_led (а затем около модуля uart\_rx\_i0) - откройте иерархию модулей (файлов) с описанием проекта.

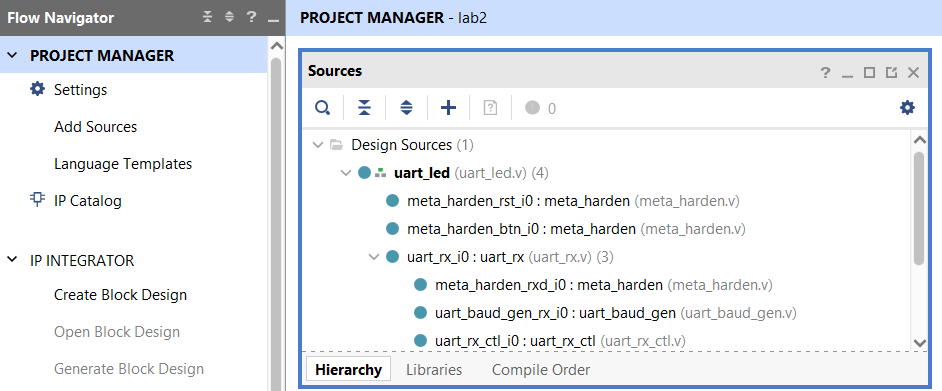


Рисунок 2 – Иерархия модулей с описанием проекта

1. В разделе **Sources**, в папке **Design Sources**, двойным щелчком по модулю (файлу) **uart\_led** **(uart\_led.v)** запустите текстовый редактор пакета Vivado, в котором будет отображено содержимое файла.

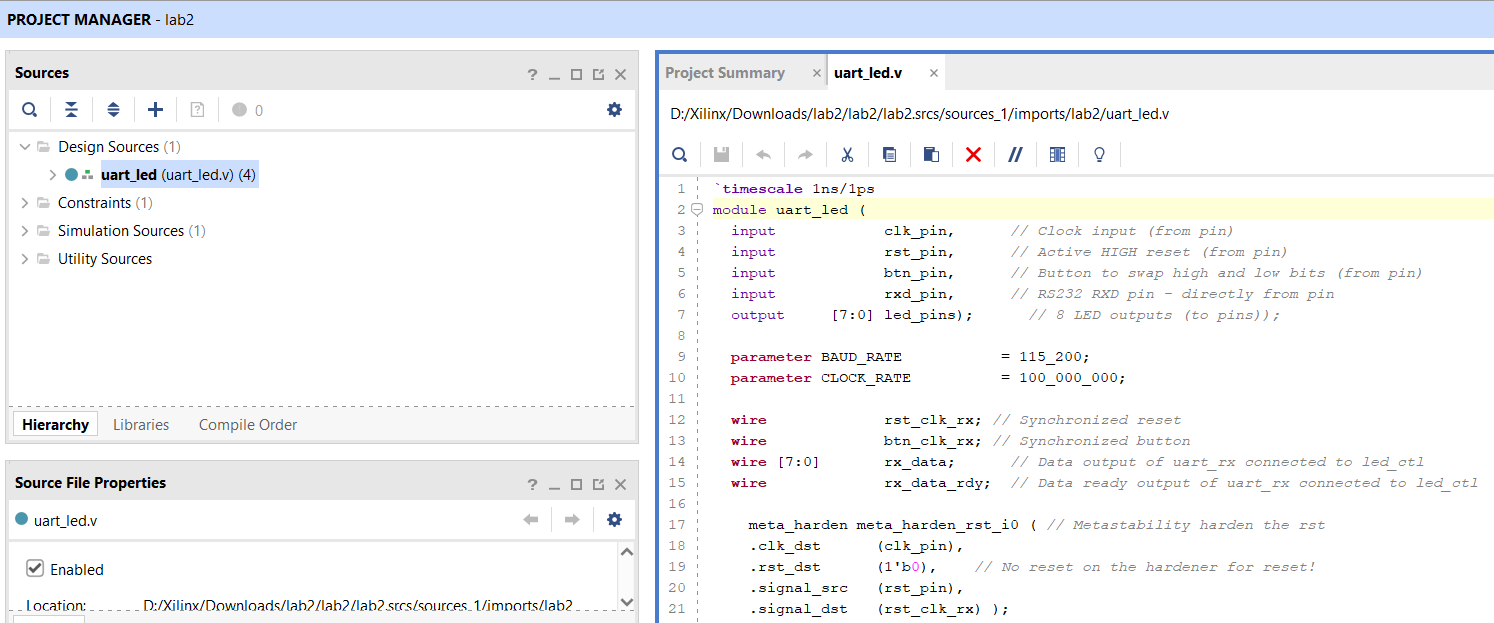


Рисунок 3 – Содержимое файла uart\_led

Этап 3. Анализ структуры RTL описания проекта

1. В разделе **Flow Navigator**, в папке **RTL Analysis** откройте папку **Open Elaborated Design** и щелчком запустите команду Schematic. В появившемся окне щелкните кнопку ОК.

*Пакет Vivado автоматически осуществит компиляцию (Elaboration) RTL описания проекта и отобразит структурную схему RTL описания верхнего уровня (так, как ее «понял» пакет).*

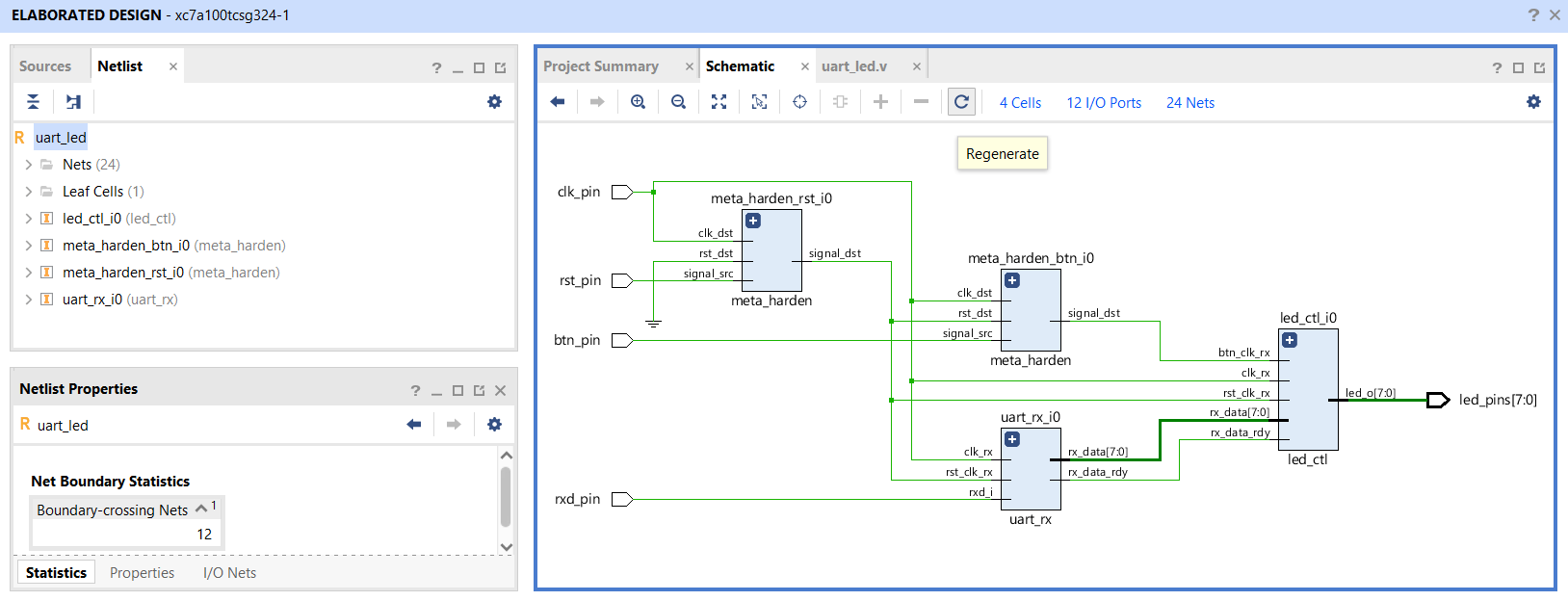


Рисунок 4 - Структурная схема RTL описания верхнего уровня

1. Посмотрите для экземпляра модуля, отображенного на структурной схеме, его объявление в исходном текстовом файле.

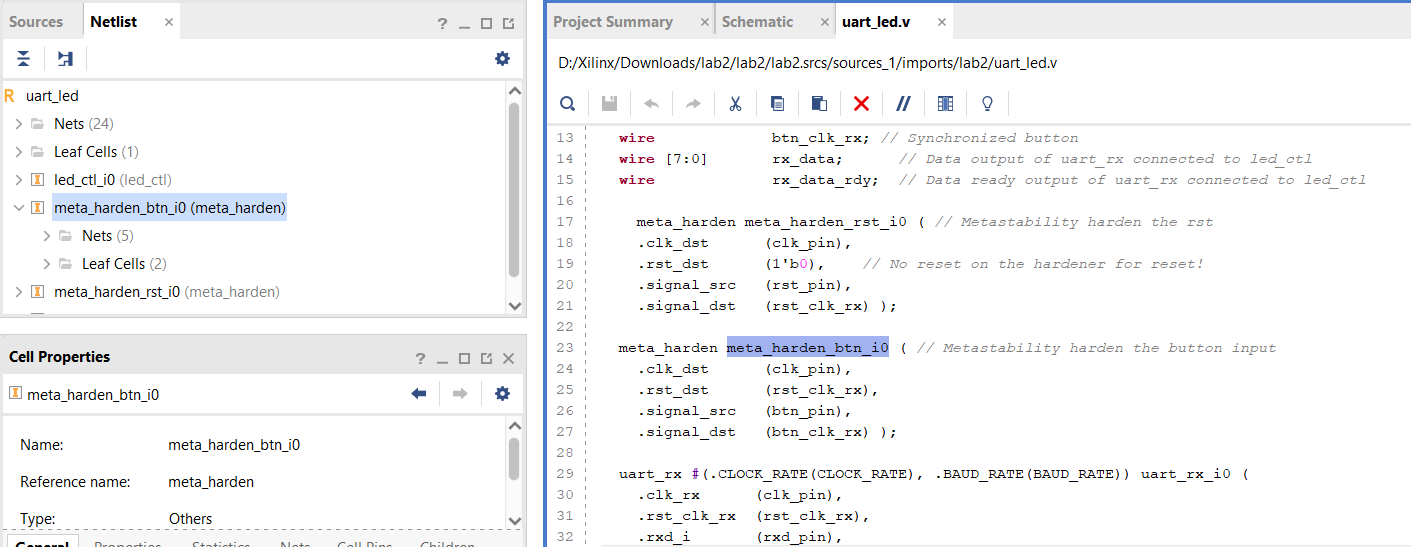


Рисунок 5 – Объявление meta\_harden\_btn\_i0

1. Отобразите структурное представление некоторых экземпляров модулей на общей структурной схеме проекта:

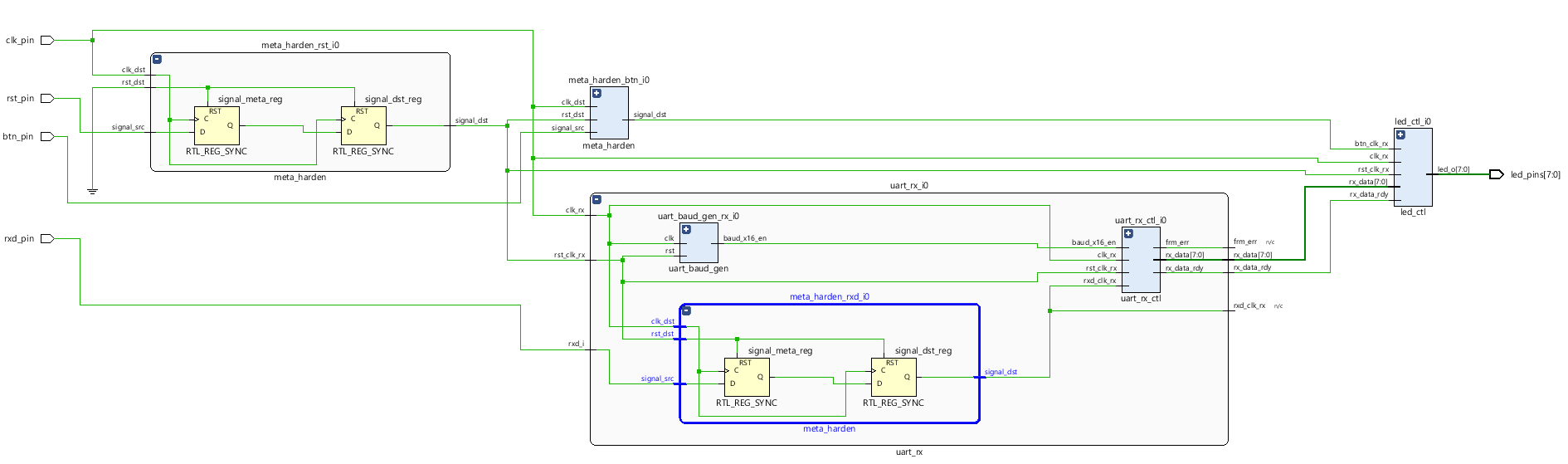


Рисунок 6 - Структурная схема проекта

1. Проверьте выполнение правил проектирования и внесите исправления в установки проекта:

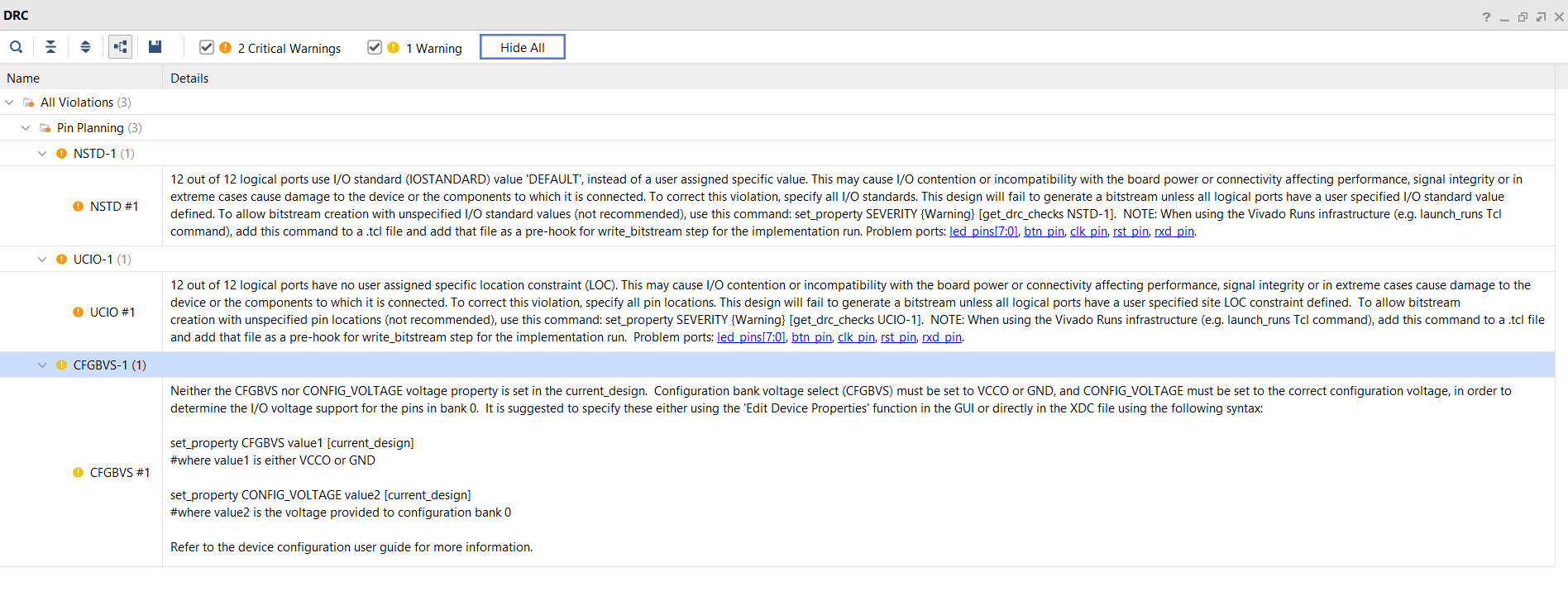


Рисунок 7 – Окно DRC

В разделе **Flow Navigator**, в папке **RTL Analysis** откройте папку **Open Elaborated Design** и щелчком запустите команду **Report DRC**. В появившемся окне щелкните кнопку **ОК**.

*Пакет Vivado осуществит проверку выполнения установленных по умолчанию правил проектирования и результат, если обнаружены ошибки и/или предупреждения, будет отображен в закладке DRC окна Elaborated Design.*

*В отчете о выполнении правил проектирования указаны: два критически важных предупреждения и одно обычное предупреждение.*

*Критически важные предупреждения связаны с отсутствием назначений, сделанных для выводов проекта (выводы проекта не связаны с выводами микросхемы FPGA).*

*Важное предупреждение связано с тем, что нет назначения для:*

* *вывода CFGBVS, определяющего допустимый диапазон напряжений конфигурационного банка;*
* *напряжения питания (Config\_Voltage) конфигурационного банка.*

1. Внесите изменения в назначения проекта для исправления важного предупреждения:

Создайте файл **uart\_led\_conf.xdc**

Щелчком запустите команду **Tools** (меню пакета Vivado) => **Edit Device Properties**. В окне **Edit Device Properties** щелчком выберите раздел **Configuration**. В разделе **Configuration**, в поле **Configuration Setup** задайте опции **Configuration** **Voltage** и **Configuration** **Bank** **Voltage** **Selection** так, как показано на рисунке ниже.

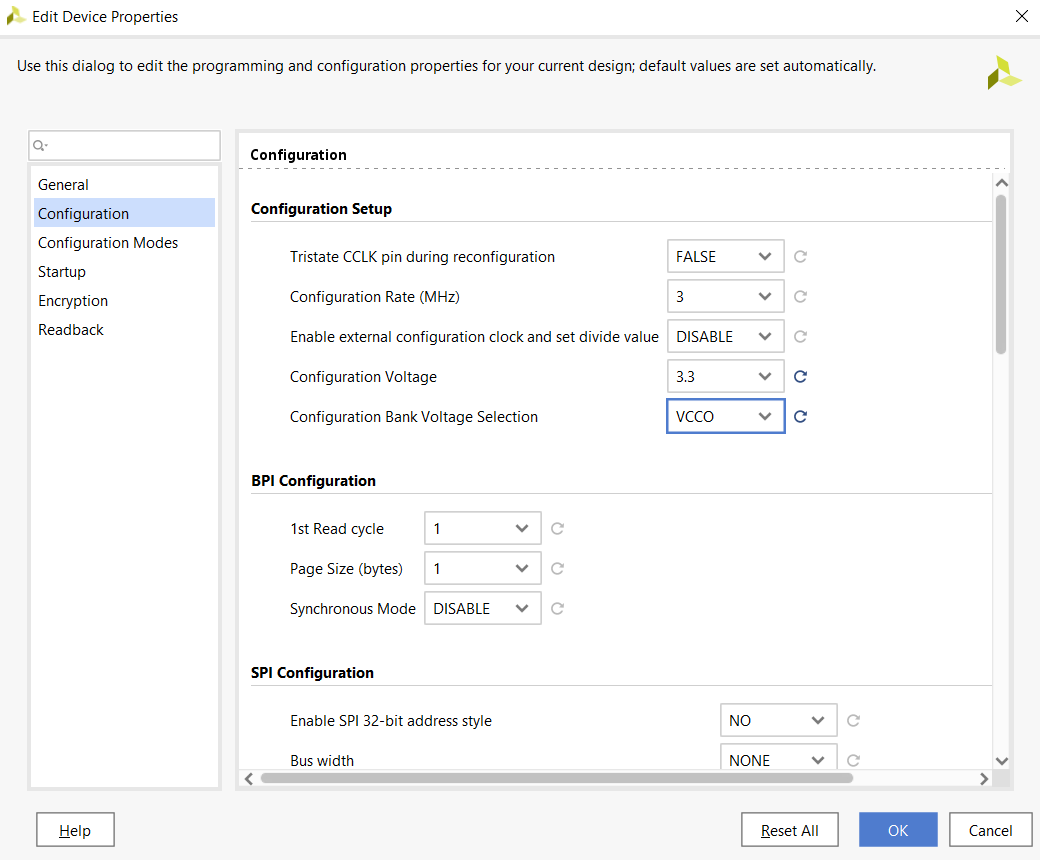


Рисунок 8 – Изменения в назначения проекта для исправления важного предупреждения

В разделе **Flow Navigator**, в папке **RTL** **Analysis** откройте папку **Open** **Elaborated Design** и щелчком запустите команду **Report** **DRC**. В появившемся окне щелкните кнопку ОК.

Убедитесь, что в отчете нет сообщения о важной ошибке, а остались только два сообщения о критических ошибках*.*

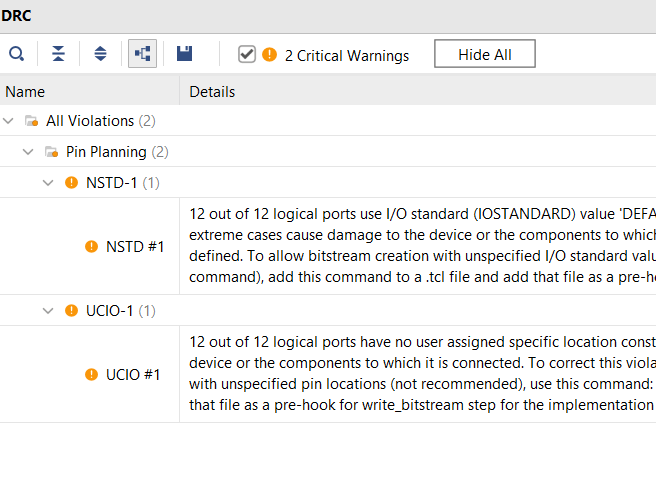


Рисунок 9 – Два сообщения о критических ошибках

Сохраните сделанные назначения в файл **uart\_led\_conf.xdc**.

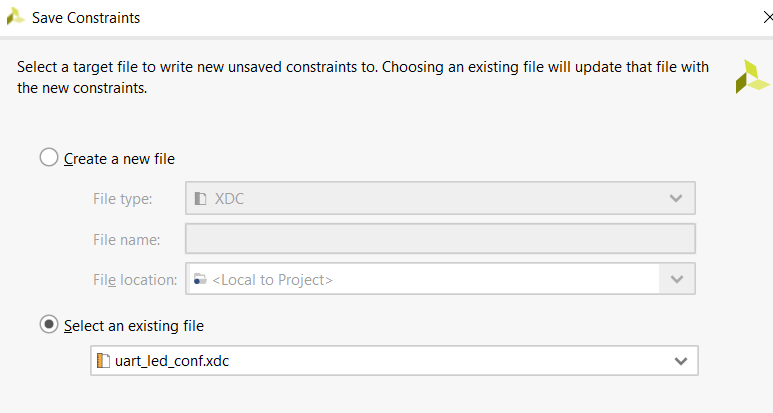


Рисунок 10 - Сохранения назначений в файл uart\_led\_conf.xdc

Проверьте содержимое файла **uart\_led\_conf.xdc**.

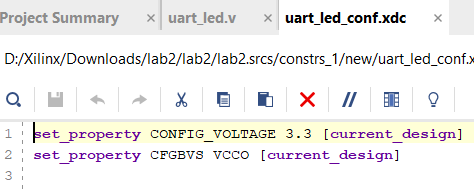


Рисунок 11 - Содержимое файла uart\_led\_conf.xdc

Этап 4. Назначение выводов микросхемы FPGA

1. Выполните команду *Layout* (меню пакета Vivado) => *I/O Planning* – раскладка главного окна пакета Vivado будет переключена в режим I/O Planning (будут добавлены несколько закладок, связанных с назначением выводов *FPGA*).
2. Обратите внимание на появившиеся закладки *Package* и *I/O Ports.*

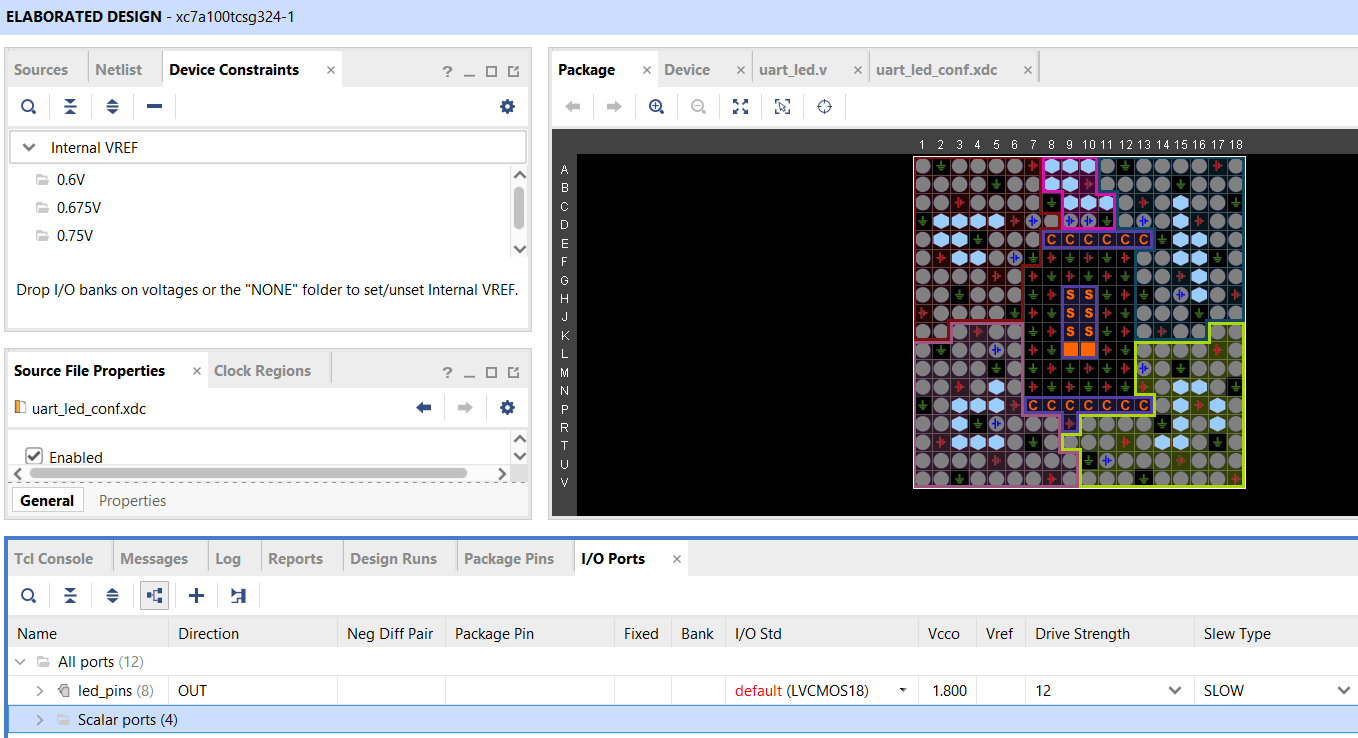


Рисунок 12 - Режим I/O Planning

1. Назначьте выводам микросхемы FPGA выводы проекта и задайте параметры выводов микросхемы FPGA.

Для каждого вывода проекта (каждого вывода в группе led\_pins (8) и в группе Scalarports (4)) в ячейке Package Pin введите номер вывода микросхемы FPGA.

Задайте стандарт для всех выводов группы led\_pins(8).

Задайте стандарт для всех выводов группы Scalarports(4).

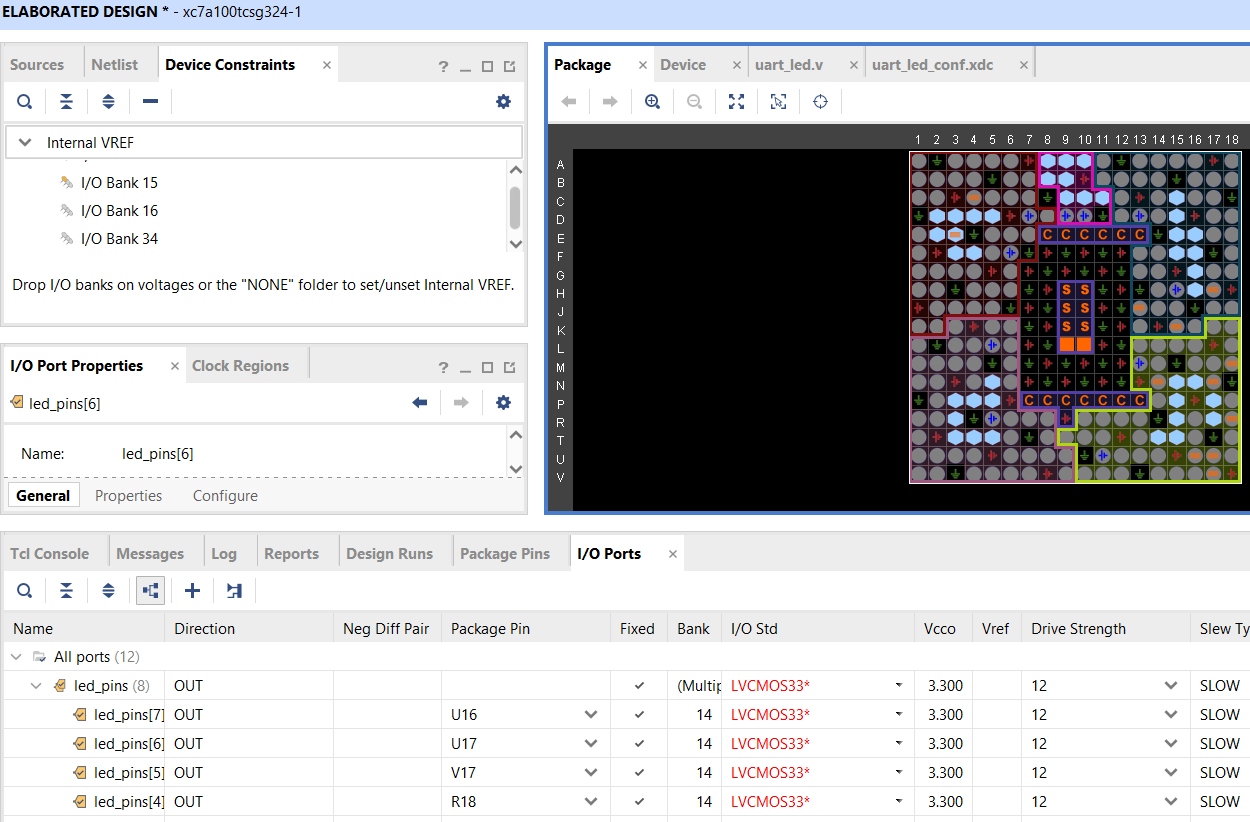


Рисунок 13 - Назначение выводов для платы

Убедитесь, что файл содержит назначения для выводов микросхемы: стандарт ввода вывода и имя вывода проекта

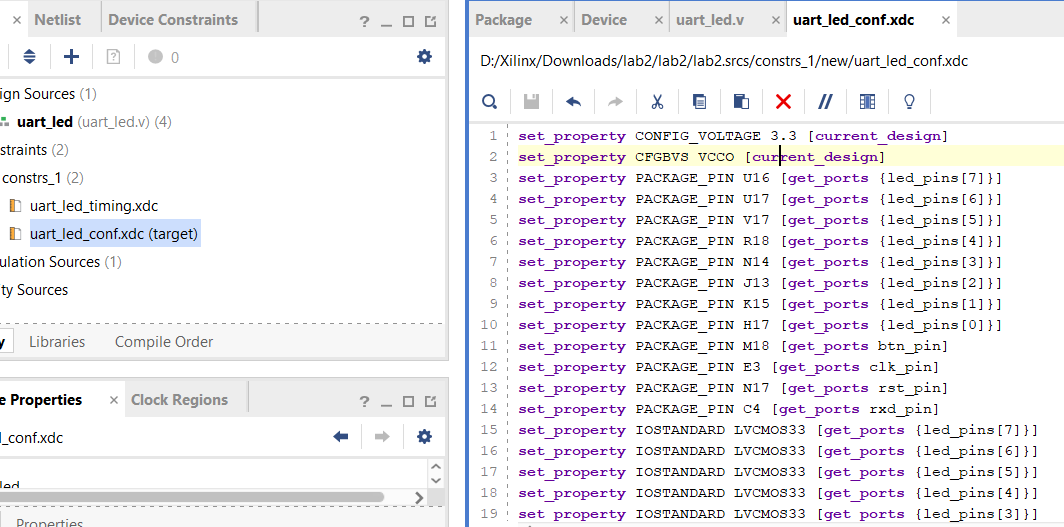


Рисунок 14 - Текстовый редактор с файлом uart\_led\_conf.xdc.

Выполните команду: Раздел Flow Navigator => RTL Analysis => Open Elaborated Design => Report DRC.

Убедитесь в том, что больше сообщений о нарушении правил проектирования нет.

После выполнения проверки появится окно Run DRC, в котором будет указано, что нарушений нет.

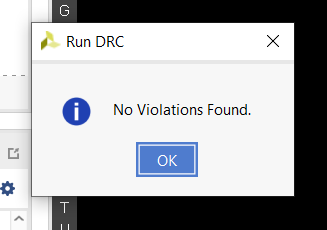


Рисунок 15 - Сообщение об отсутствие нарушений

Этап 5. Синтез проекта и анализ результатов

1. Запустите процедуру Синтеза:

* В разделе **Flow Navigator**, в папке **Synthesis** щелчком запустите команду **Run Synthesis**.
* В появившемся окне **Launch** **Runs** (если его появление не было запрещено установками пакета) оставьте все установки по умолчанию и щелкните кнопку **ОК**.

*Пакет Vivado выполнит процедуру синтеза описания проекта, приведенного в файле uart\_led.v и файлах нижних уровней иерархии описания проекта. Пакет автоматически находит все файлы в иерархии описания проекта при реализации процедуры синтеза.*

1. В разделе Synthesized Design выберите закладку Project Summary.

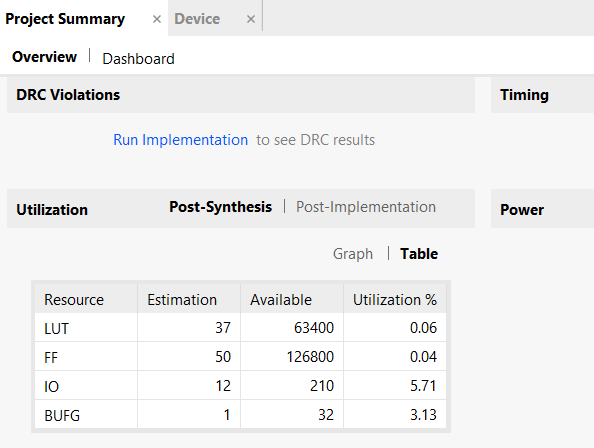


Рисунок 16 - Таблица использованных/доступных ресурсов и процент их использования

В приведенной таблице указано количество использованных/доступных ресурсов и процент их использования в выбранной для синтеза микросхеме FPGA:

* LUT (таблиц перекодировок)
* FF (триггеров)
* IO (выводов)
* BUFG (буферов тактовых сигналов)

1. Отобразите структурную схему синтезированного проекта и найдите соответствие элементов структурной схемы исходному текстовому описанию*:*

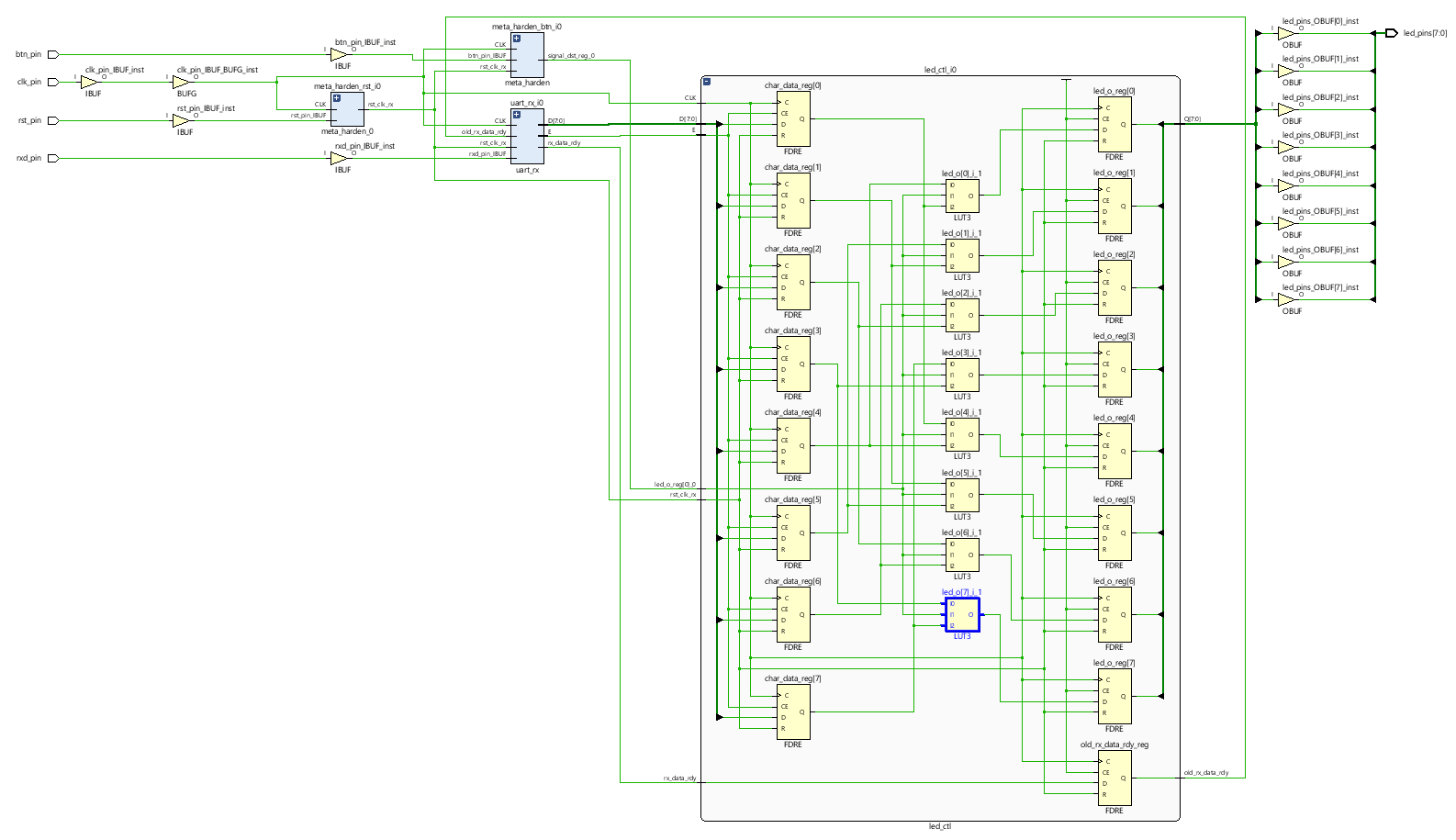


Рисунок 17 - Структурную схему синтезированного проекта

Обратите внимание на то, что структурная схема синтезированного проекта базируется на элементной базе, соответствующей выбранной микросхеме FPGA. В данном примере содержит:

* таблицы перекодировок (LUT),
* триггеры (FDRE),
* буферы элементов ввода/вывода (IBUF/OBUF),
* буфер тактового сигнала (BUFG).

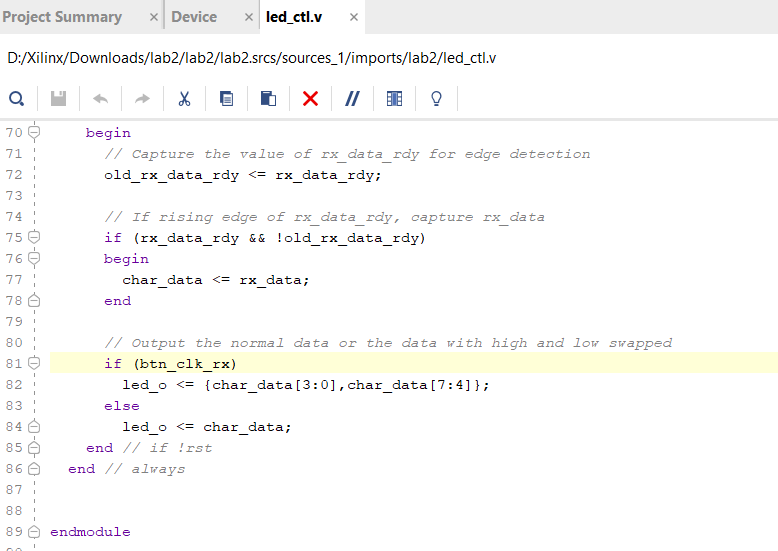


Рисунок 18 - Текстовый редактор с описанием элемента

Элемент LUT3 с именем led\_o[7]\_i\_1 реализует функцию мультиплексора (2=>1) для разряда [7]в соответствии со значением сигнала btn\_clk\_rx.

Этап 6. Анализ временных параметров синтезированного проекта

1. В разделе ***Sources***, в папке ***Constraints*** двойным щелчком по файлу ***uart\_led\_timing.xdc*** запустите текстовый редактор пакета Vivado, в котором будет отображено содержимое файла.

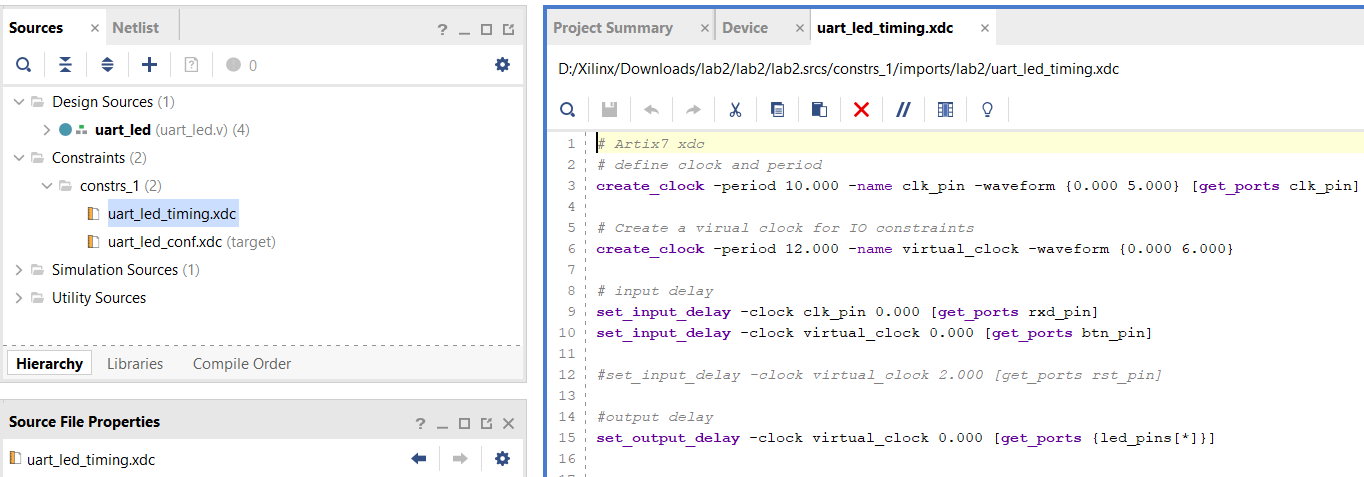


Рисунок 19 - Содержимое файла uart\_led\_timing.xdc

1. В разделе ***Flow Navigator***, в папке ***Synthesis*** щелчком запустите команду ***Report Timing Summary***.

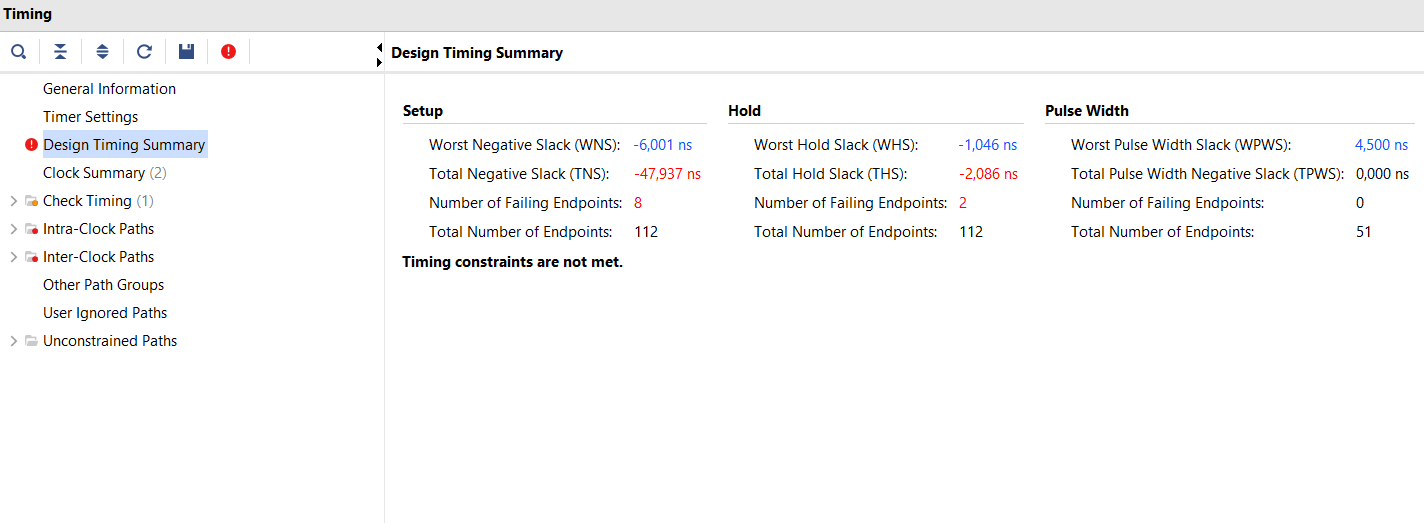


Рисунок 20 – Окно Timing

1. Щелчком по значению ***Worst Negative Slack (WNS),*** на рисунке выше значение выделено красным прямоугольником, откройте список из 8 цепей, для которых временные параметры не были достигнуты.

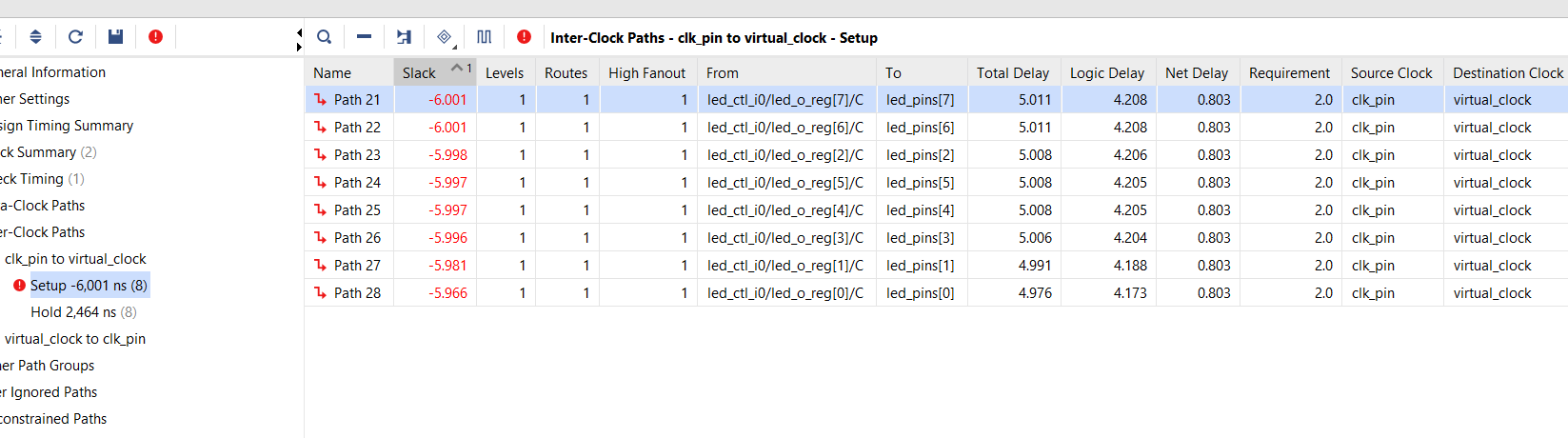


Рисунок 21 – Список из 8 цепей, для которых временные параметры не были достигнуты

Обратите внимание на то, что в списке указаны:

* *Name* – внутреннее имя цепи, данное пакетом
* *Slack* – отличие полученного временного параметра от заданного (отрицательное значение – получено хуже ожидаемого)
* *From* – источник цепи
* *To* – приемник цепи
* *Total* *delay* – общая задержка сигнала в цепи
* *Logic* *Delay* – задержка в логической части
* *Net delay* – задержка в цепях передачи данных (в цепях трассировки)
* *Requirement* – требование к задержке, вычисляемое из заданных в XDC файле параметров
* *Source* *Clock* – тактовый сигнал синхронизации источника
* *Destination* *Clock* – тактовый сигнал синхронизации приемника

*Из анализа этой таблицы следует, что в данной лабораторной работе возникли проблемы с достижением заданных в XDC файле временных параметров для выходных цепей проекта: при передаче данных с выхода микросхемы FPGA на вход внешнего устройства, синхронизируемого частотой virtual\_clock.*

1. Двойным щелчком по имени **Path21** откройте детальный отчет о полученных и заданных параметрах для этого пути.

*В появившемся окне обратите внимание на разделы:*

* *Summary – общая информация о выбранном критическом пути распространения сигнала*
* *Source Clock Path – информация о пути распространения тактового сигнала источника данных*
* *Data Path – информация о пути распространения данных*
* *Destination Clock Path – информация о пути распространения тактового сигнала приемника данных*

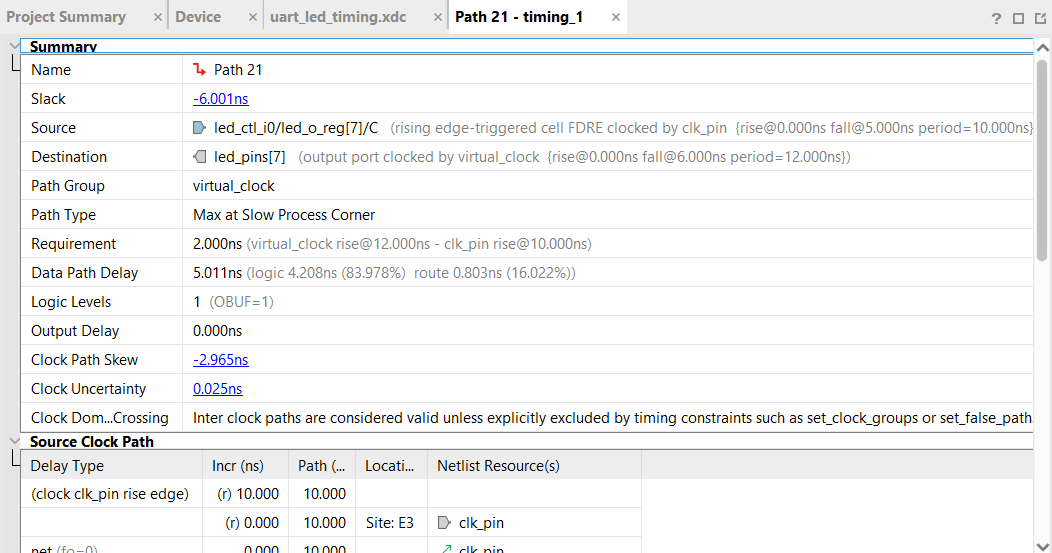


Рисунок 22 – Детальный отчет о полученных и заданных параметрах

Раздел, в частности, содержит информацию для цепи Path21:

* О требовании к временному параметру (*Requirement*) и о принципе его вычисления. *Requirement* — это тот промежуток времени, через который, с момента появления тактового сигнала на тактовом входе проектируемой микросхемы (строка *Source*), на указанном выводе проектируемой микросхемы (строка *Destination*) должны появиться данные.

*В данном случае т.к. в требованиях к временным параметрам выходная задержка (задержка на плате и входе внешней микросхемы) задана равной 0 (смотри строку Output Delay и строчку 15 в файле* ***uart\_led\_timing.xdc****), Requirements - это разница между фронтами тактовых сигналов:*

* *virtual\_clk – синхронизирующего внешнюю микросхему, приемник данных*
* *clk\_pin – синхронизирующего разрабатываемую микросхему.*
* О времени распространения данных от момента поступления тактового сигнала на тактовый вход триггера, синхронизирующего выходной сигнал проектируемой микросхемы, до появления данных на выходе микросхемы (строка *Data Path Delay*).
* О задержке распространения тактового сигнала от тактового входа проектируемой микросхемы до тактового входа триггера, синхронизирующего выходной сигнал проектируемой микросхемы (*Clock Path Skew*)
* О вычисленном значении неопределенности фронта тактового сигнала (строка *Clock* *Uncertainty*), связанного с дрожанием фронта тактового сигнала (*Jitter*)
* О значении разницы между требованием (*Requirement*)и вычисленной задержкой появления данных на выходе проектируемой микросхемы (*Destination*) от момента появления тактового сигнала на входе синхронизации проектируемой микросхемы (*Source*).

Вычисляется это значение, для данного случая, следующим образом: *Requirement* – (*Data* *Path* *Delay* – *Clock* *Path* *Skew* + *Clock* *Uncertainty*) = -6.001нс.

1. Щелчком выберите имя *Path21*, нажмите правую клавишу мыши и выполните команду *Schematic*.

Появится окно, в котором будет отображена структурная схема элементов, образующих *Path21*.

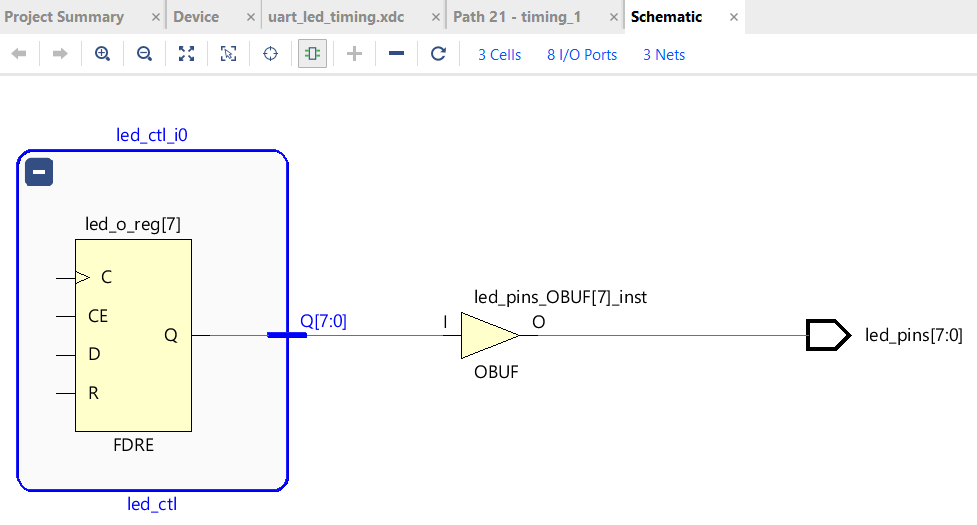


Рисунок 23 – Структурная схема элементов, образующих Path21

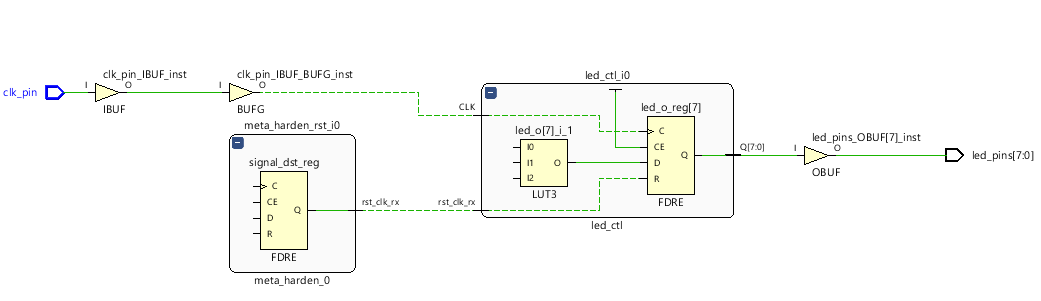


Рисунок 24 - Пути тактового сигнала

Пути распространения сигналов и компоненты в цепи, представленные на рисунке выше, соответствуют путям и компонентам, приведенным в детальном отчете для пути *Path21*, в разделах *Source Clock Path* и *Data Path.*

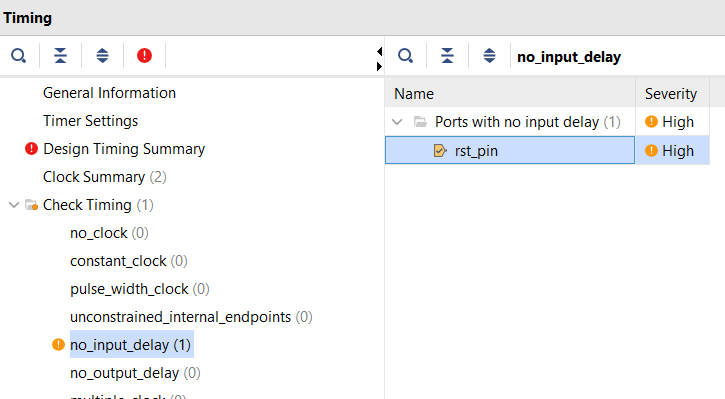


Рисунок 25 - Детальная информация о предупреждении

В правой части окна Timing появится детальная информация о предупреждении: в данном случае это отсутствие заданного параметра о входной задержке для входа rst.

Этап 7. Изменение временных параметров, синтез и анализ

1. В разделе *Flow Navigator*, в папке *Synthesis* щелчком запустите команду *Edit Timing Constraints* (*будет запущен Timing Constraints Wizard*).

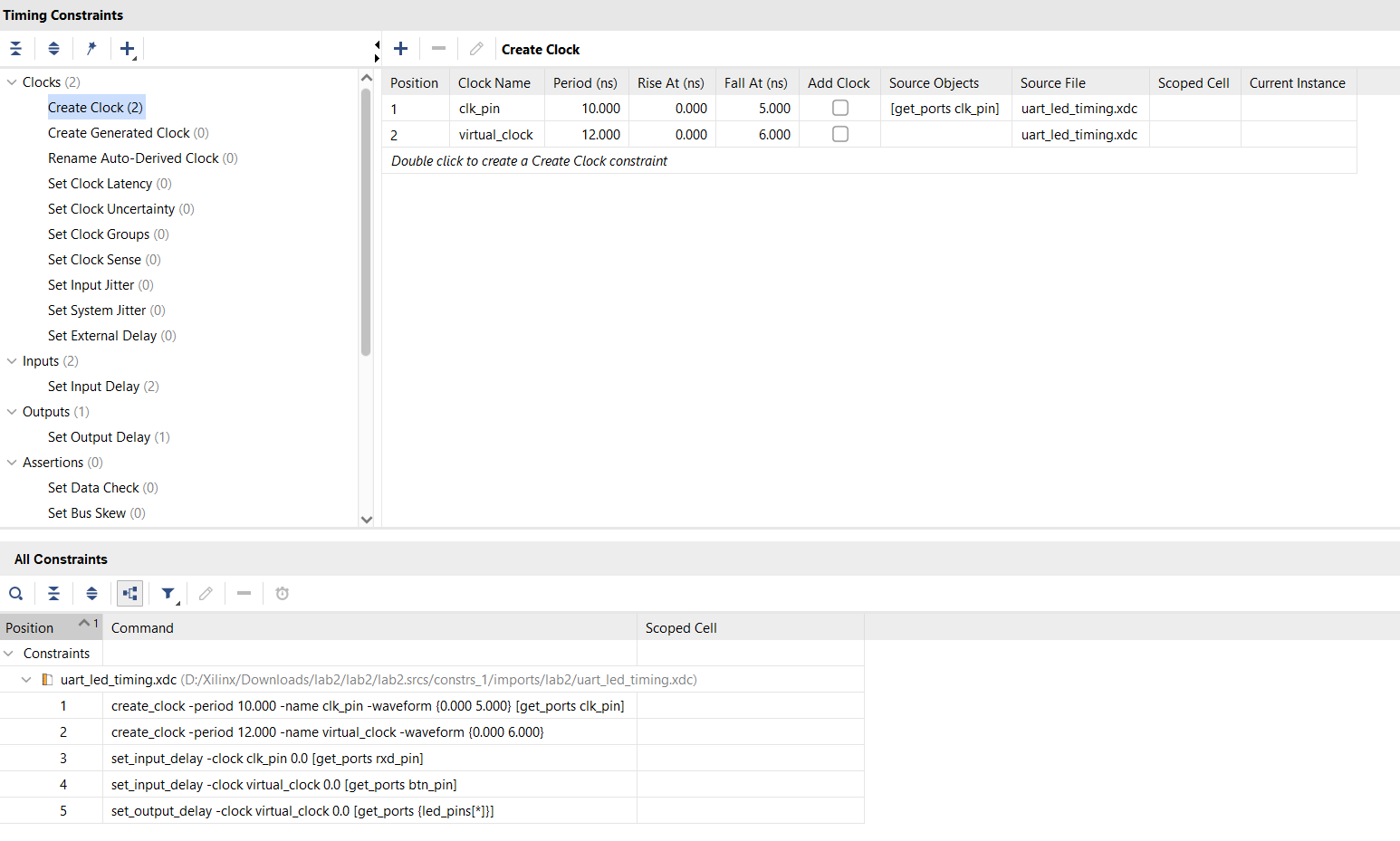


Рисунок 26 - Запуск Timing Constraints Wizard

1. Измените период тактового сигнала virtual\_clock с 12ns на 10ns*.*

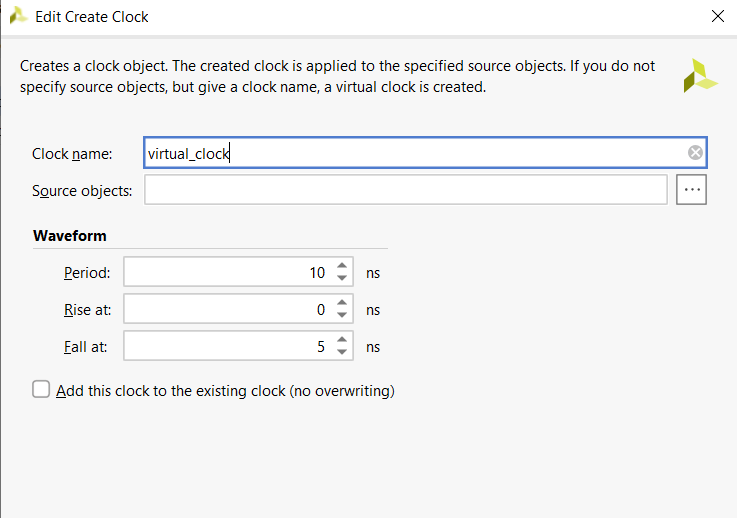


Рисунок 27 - Изменение периода тактового сигнала virtual\_clock

*Мы изменили тактовый сигнал для внешних микросхем, теперь они будут синхронизироваться тактовым сигналом virtual\_clock, синхронным с тактовым сигналом разрабатываемой микросхемы (clk\_pin)*

1. В разделе *Timing Constraints* разверните папку *Inputs* и щелчком выберите строку *Set Input Delay*.

* Одним щелчком выберите и измените значение *Delay Value* для входа *rxd\_pin* (синхронизируется тактовым сигналом clk\_pin) с 0 ns на 2ns.
* Одним щелчком выберите и измените значение *Delay Value* для входа *btn\_pin* (синхронизируется тактовым сигналом virtual\_clock) с 0ns на 2ns.

Обратите внимание на то, что на закладке *All Constraints* изменились строчки, описывающие параметры *set\_input\_delay*.

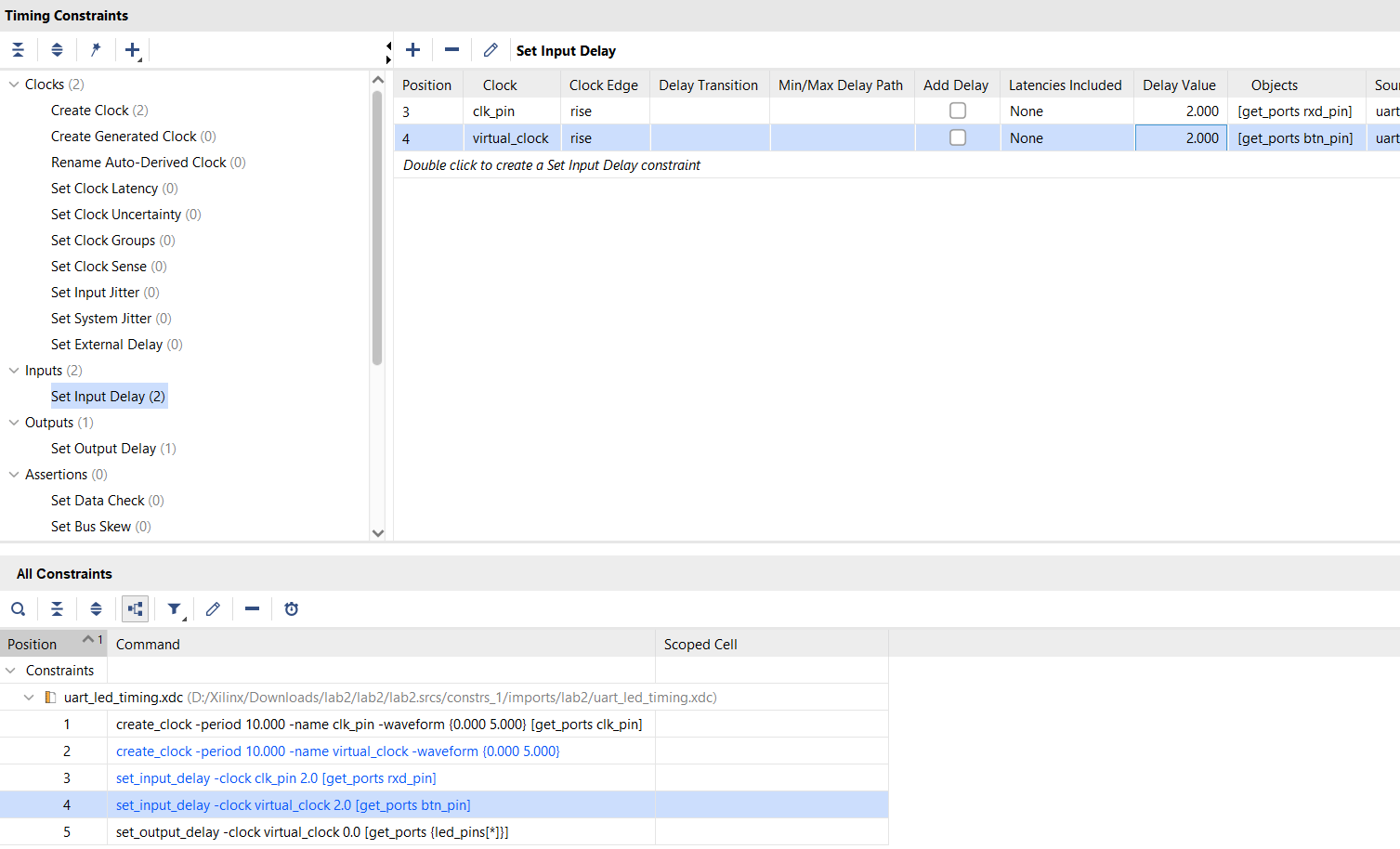


Рисунок 28 – Set Input Delay

1. На закладке *Design Runs*, окна *Synthesized Design* в ячейке *Status* щелкните правой клавишей мыши по надписи *Synthesis Out-of-Date* и в появившемся меню щелчком запустите команду *Force Up-to-Date*.

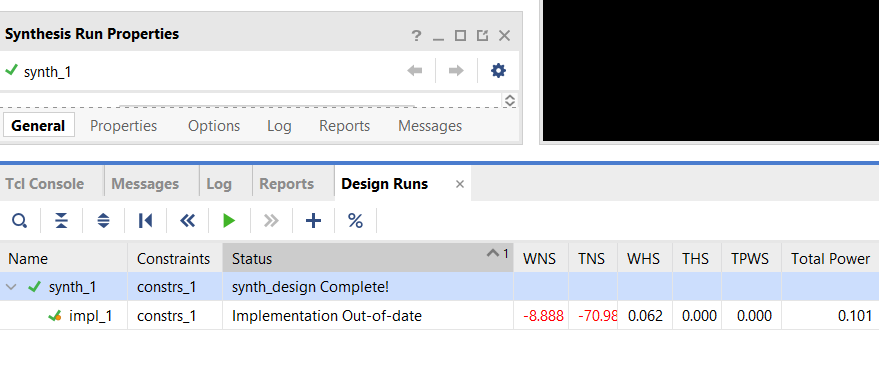


Рисунок 29 - Результаты синтеза

*Результаты синтеза будут обновлены без перезапуска всей процедуры синтеза с самого начала.*

Обратите внимание на то, что в результатах, полученных после завершения процедуры временного анализа, представленных на рисунке ниже, указано, что все требования к временным параметрам, заданным в XDC файле, выполнены.

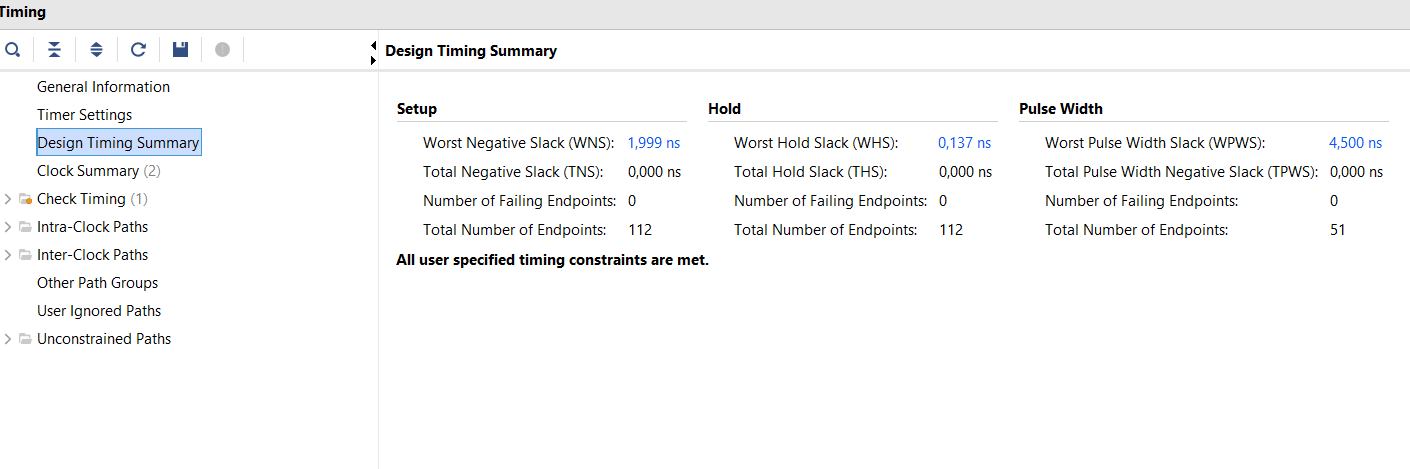


Рисунок 30 - Результаты после завершения временного анализа

Обратите внимание на параметры, приведенные в колонке Setup:

* *Worst Negative Slack (WNS)* – наибольшее отклонение среди всех, полученных в результате синтеза, временных параметров от заданных в XDC файле временных параметров.
* *Total Negative Slack (TNS*) – сумма всех Negative Slacks.
* *Number of Failing End points* – число приемников, для которых не достигнуты требуемые временные параметры.
* *Total Number of End points* – общее число приемников в проекте.

Все указанные в данной колонке отклонения в достижении заданных требований относятся к времени предустановки данных на входе синхронного элемента относительно фронта сигнала синхронизации на его входе синхронизации (т.е. относятся к так называемому Setup анализу).

1. Щелчком по значению *Worst Negative Slack (WNS*), на рисунке выше значение выделено зеленым прямоугольником, откройте список из 8 цепей, для которых временные параметры были достигнуты, но они имеют минимальный положительный зазор.

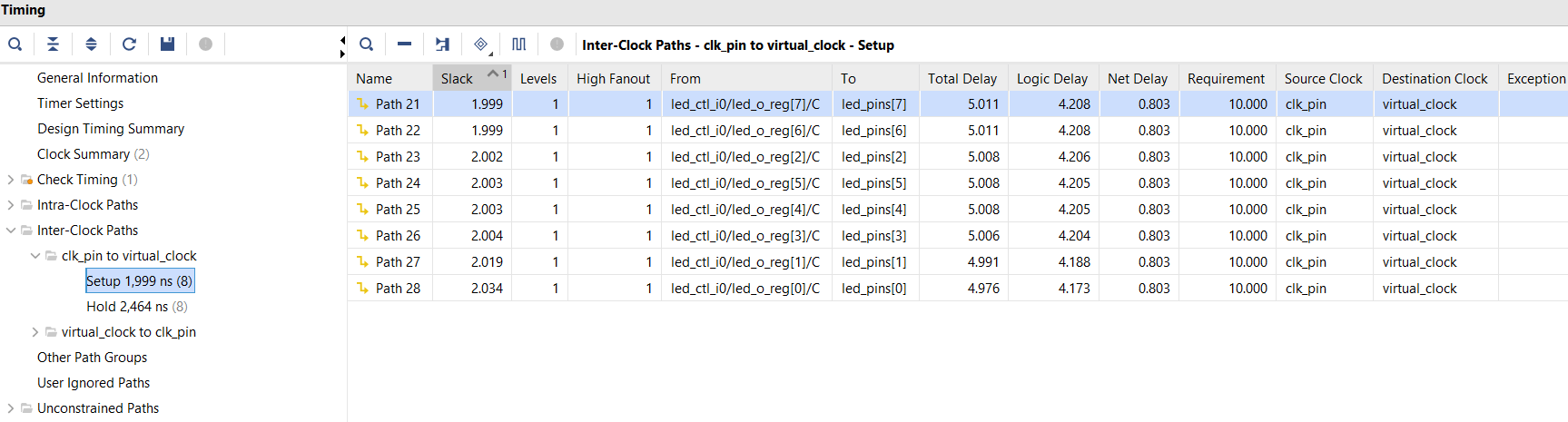


Рисунок 31 - Список из цепей, для которых параметры были достигнуты, но они имеют минимальный зазор

Обратите внимание на то, что в списке указаны:

* *Name* – внутреннее имя цепи, данное пакетом
* *Slack* – отличие полученного временного параметра от заданного (отрицательное значение – получено хуже ожидаемого)
* *From* – источник цепи
* *To* – приемник цепи
* *Total delay* – общая задержка сигнала в цепи
* *Logic Delay* – задержка в логической части
* *Net delay* – задержка в цепях передачи данных (в цепях трассировки)
* *Requirement* – требование к задержке, вычисляемое из заданных в XDC файле параметров
* *Source Clock* – тактовый сигнал синхронизации источника
* *Destination Clock* – тактовый сигнал синхронизации приемника.

Из анализа этой таблицы следует, что в данной лабораторной работе нет проблем с достижением заданных в XDC файле временных параметров для выходных цепей проекта: при передаче данных с выхода микросхемы FPGA на вход внешнего устройства, синхронизируемого частотой *virtual\_clock*.

1. Двойным щелчком по имени *Path21* откройте детальный отчет о полученных и заданных параметрах для этого пути.

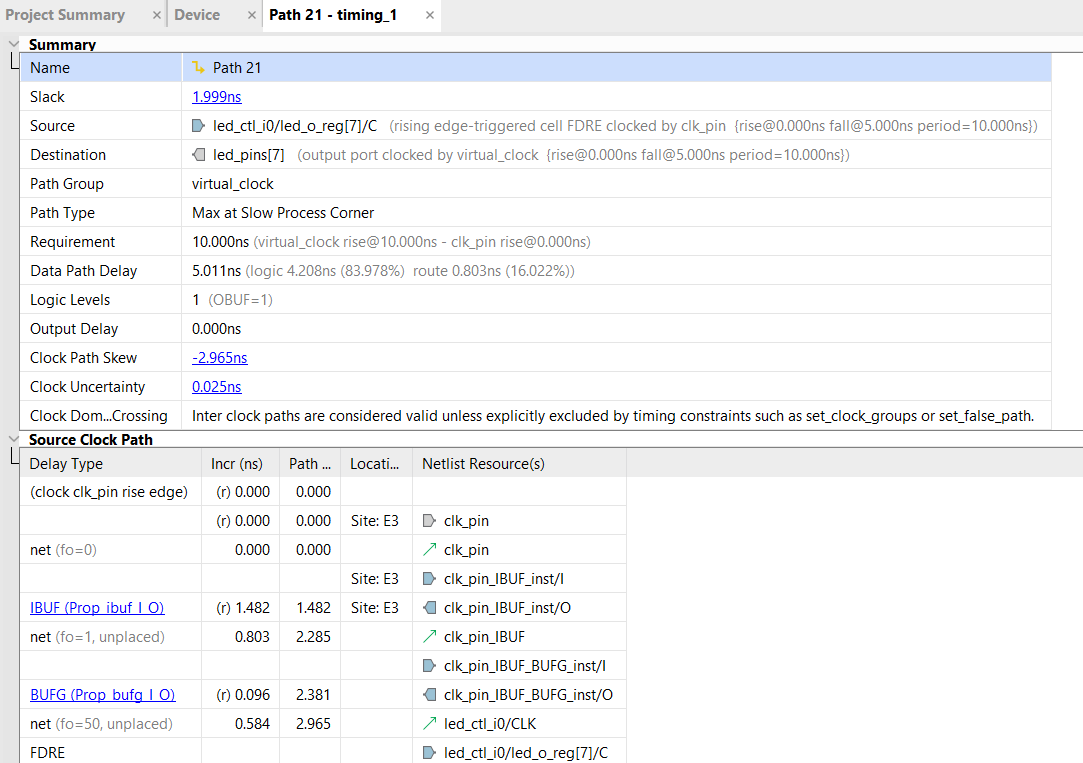


Рисунок 32 - Детальный отчет о параметрах

1. Внесите изменения в файл *uart\_led\_timing.xdc*.

* В окне *Synthesized Design* щелчком выберите закладку Sources.
* На закладке *Sources* откройте папку *Constraints* и в подпапке constrs\_1 двойным щелчком откройте файл *uart\_led\_timing.xdc*.
* В файле *uart\_led\_timing.xdc* найдите строчку 12 и удалите знак # (признак комментария).

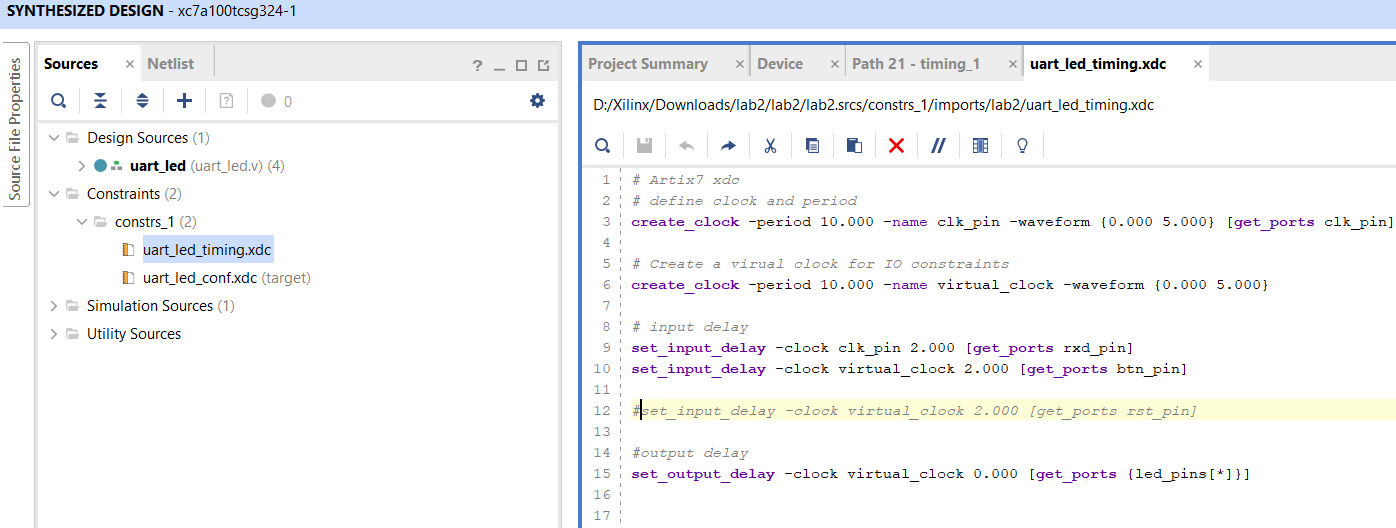


Рисунок 33 - Изменения в файл uart\_led\_timing.xdc

*Строчка 12 задает входную задержку для входа rst\_pin микросхемы FPGA относительно фронта тактового сигнала virtual\_clock.*

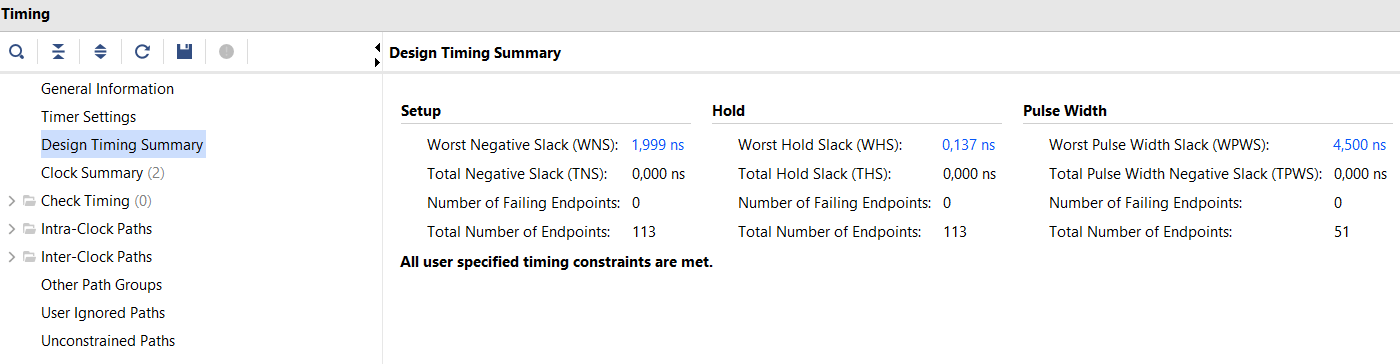


Рисунок 34 - Временные параметры проекта

Убедитесь в том, что полученные временные параметры проекта соответствуют сделанным назначениям и нет предупреждений о выводах без назначений временных параметров.

Этап 8. Анализ аппаратных затрат и потребляемой мощности

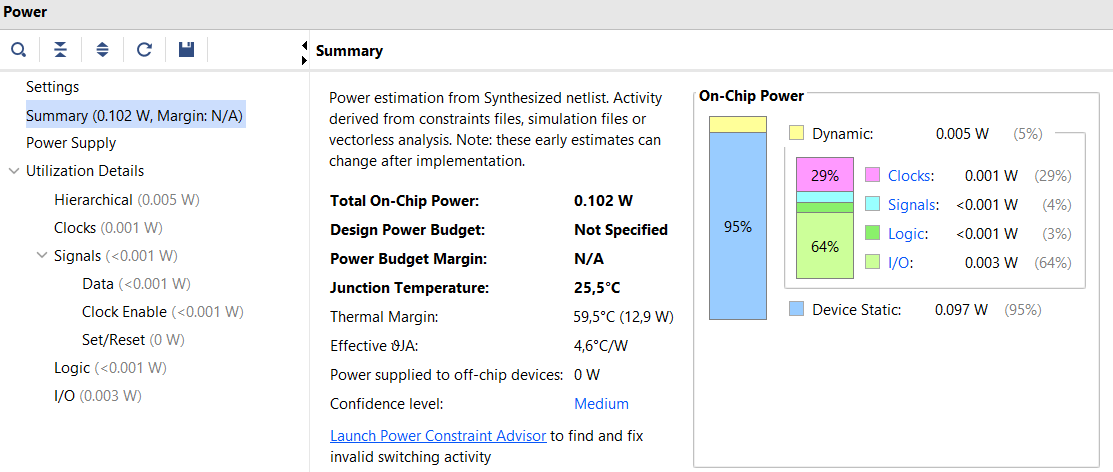


Рисунок 35 - Power

Обратите внимание на то, что на закладке отображаются:

* Оценки потребляемой мощности в статическом режиме (*Static*) и динамическом режимe (*Dynamic*);
* Оценки мощности, потребляемой логическими элементами (*Logic*); при передаче данных (Signal); тактовыми сигналами (*Clocks*); элементами ввода-вывода (*I/O*).
* Уровень точности (*Confidence level*) приведенных оценок.

Этап 9. Создание контрольной точки

*Контрольная точка позволяет сохранить результаты для использования их в дальнейшем как для анализа, так и в качестве исходных данных для выполнения следующих этапов проектирования.*

1. Выполните команду *File* (меню пакета Vivado) => *Checkpoint* => *Write*.

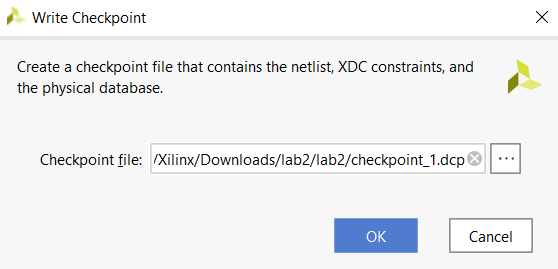


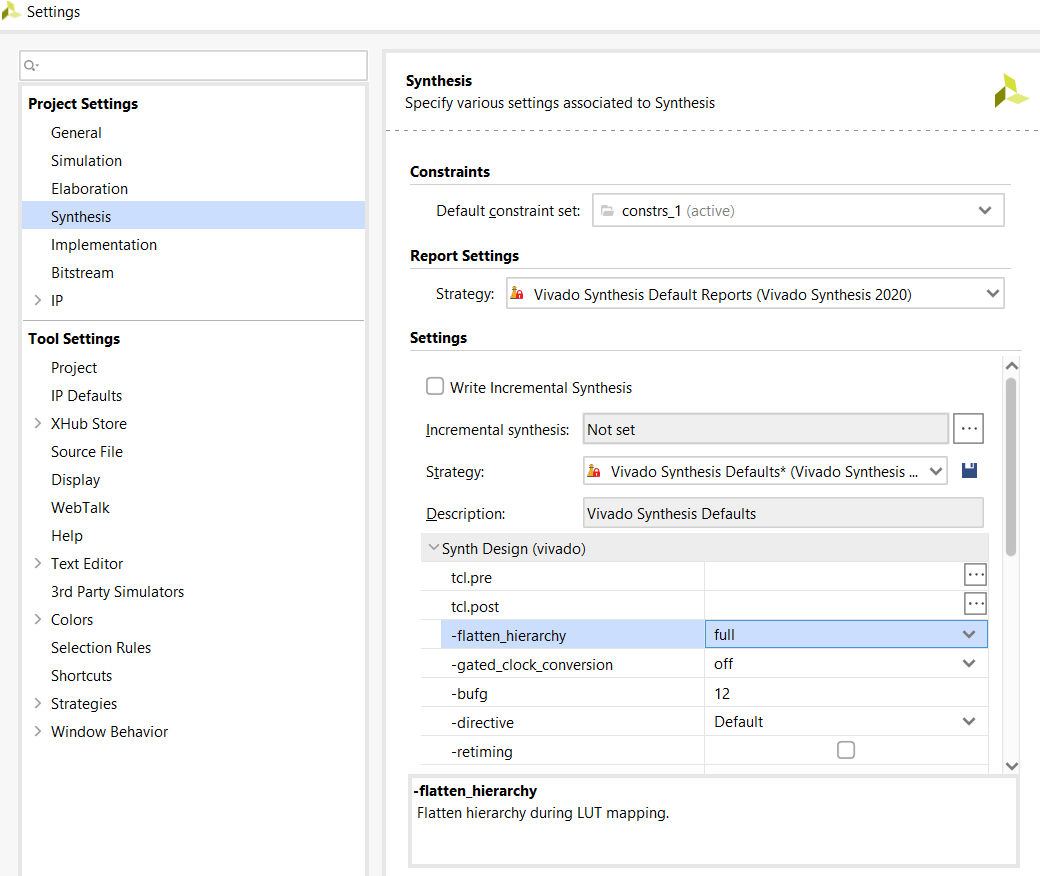
Рисунок 36 - Write Checkpoint

1. Появится окно *Write* *Checkpoint*, в котором будет указана папка сохранения контрольной точки и имя файла, созданное по умолчанию. Щелкните кнопку ОК – контрольная точка будет создана.

Этап 10. Изменение настроек синтеза и анализ результатов

1. В разделе *Flow Navigator* щелчком разверните папку *Synthesis*, нажмите правой клавишей мыши в строке *Run Synthesis* и выберите в появившемся окне команду *Synthesis Settings*.

* В появившемся окне *Settings* в разделе Project Settings щелчком выберите папку Synthesis.
* В окне настроек процедуры синтеза измените опцию – *flatten\_hierarchy* на *full*.



В появившемся окне (если его появление не было запрещено), в котором предлагается оставить неизменными результаты текущего синтеза (в выполняемой лабораторной работе это – *synth\_1*) и создать новые результаты синтеза, щелкните кнопку *ОК*.

Обратите внимание на закладку *Design Runs* – в окне закладки появилось имя *synth\_flatten*. И эта папка является активной (active).

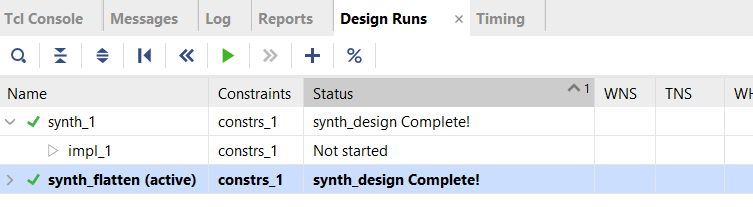


Рисунок 37 - Результаты синтеза

1. В окне раздела *Flow Navigator* выберите папку *Synthesis* и щелчком запустите команду *Schematic*.

* Появится окно *Schematic*, в котором будет отображена структурная схема проекта, полученная после синтеза с опцией *flatten*.

Обратите внимание на то, что структурная схема не содержит иерархических блоков, а реализована на элементной базе (простейших элементах), заданной выбором микросхемы FPGA для реализации проекта.

* Создайте для полученных результатов синтеза контрольную точку:
* Выполните команду *File* (меню пакета Vivado) => *Checkpoint* => *Write*.

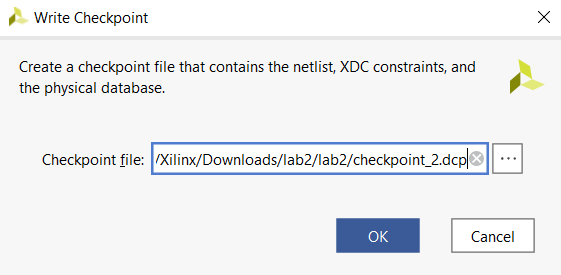


Рисунок 38 - Создание контрольной точки

В появившемся окне *Write Checkpoint*, в котором будет указана папка сохранения контрольной точки, задайте имя файла: *checkpoint\_flatten.dcp*.

Этап 11. Загрузка контрольной точки

*Контрольная точка позволяет сохранить результаты для использования их в дальнейшем как для анализа, так и в качестве исходных данных для выполнения следующих этапов проектирования.*

1. Выполните команду *File* (меню пакета Vivado) => *Checkpoint* => *Open*.

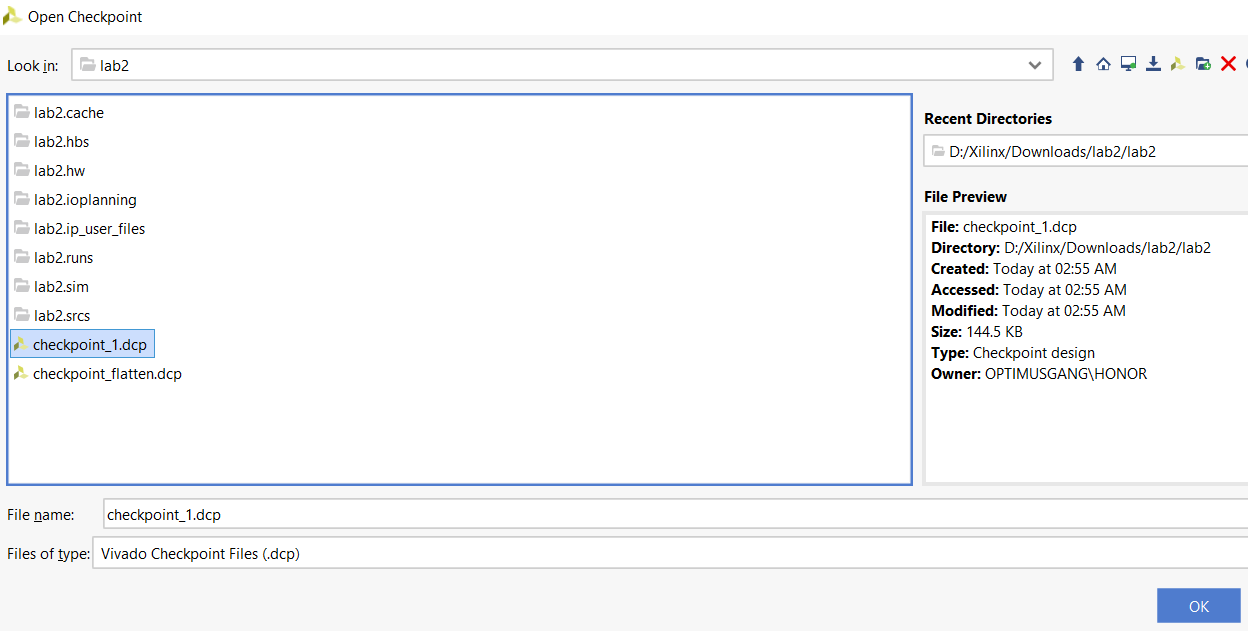


Рисунок 39 - Загрузка контрольной точки

1. После окончания процесса загрузки контрольной точки откроется окно пакета Vivado с загруженной контрольной точкой

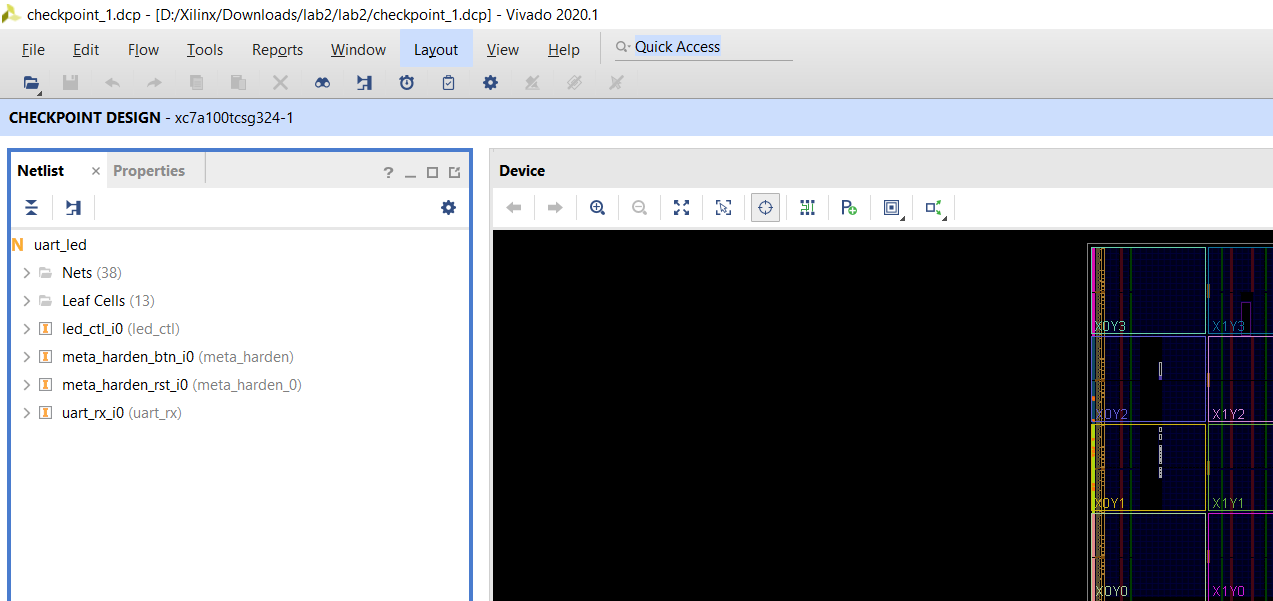


Рисунок 40 - Окно Vivado с загруженной контрольной точкой

1. Откройте схемное представление результатов синтеза для загруженной контрольной точки

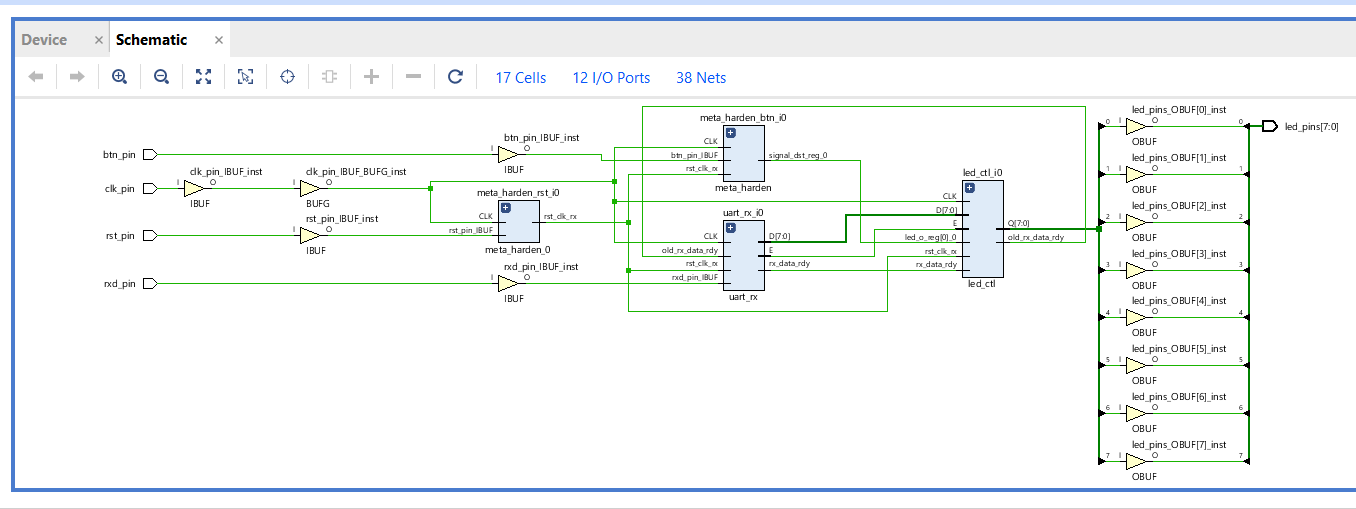


Рисунок 41 - Схемное представление результатов синтеза для загруженной точки

1. Посмотрите результаты временного анализа результатов загруженной контрольной точки

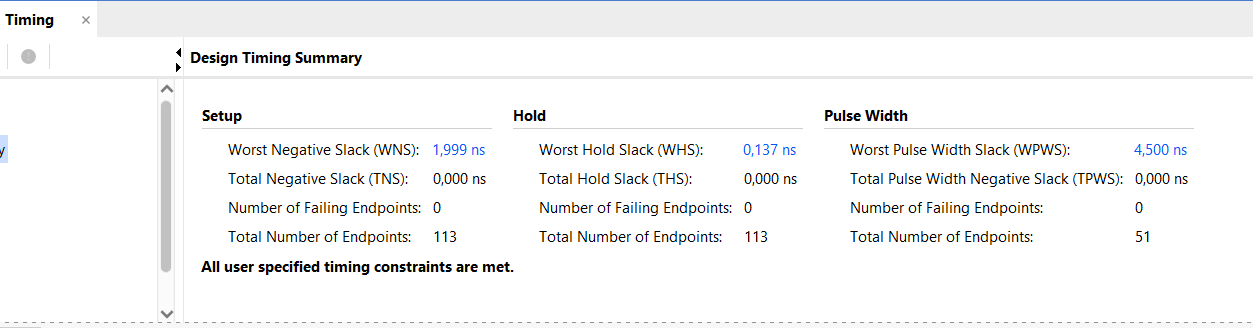


Рисунок 42 - Результаты временного анализа результатов

Вывод:

В ходе выполнения лабораторной работы мы узнали, как:

* назначить выводы микросхемы;
* задать параметры проекта;
* синтезировать проект;
* анализировать результаты синтеза;
* проводить временной анализ;
* анализировать и изменять задаваемые в XDC файле временные параметры;
* создавать и загружать контрольные точки;
* проводить анализ аппаратных затрат и потребляемой мощности после процедуры синтеза.