



Разработка доверенных средств проектирования ИС в базе гетерогенных ПЛИС

С.В. Гаврилов, ORCID: 0000-0003-0566-4482 <s00v@yandex.ru>

Д.А. Железников, ORCID: 0000-0003-2477-1793 <zheleznikov_d@ippm.ru>

М.А. Заплетина, ORCID: 0000-0001-9845-7823 <zapletina_m@ippm.ru>

И.В. Тиунов, ORCID: 0009-0008-6241-5451 <tiunov_i@ippm.ru>

В.М. Хватов, ORCID: 0000-0003-2285-4341 <khvatov_v@ippm.ru>

Р.Ж. Чочаев, ORCID: 0009-0007-0304-0496 <chochaev_r@ippm.ru>

Д.Б. Шокарев, ORCID: 0009-0009-2806-1570 <shokarev_d@ippm.ru>

*Институт проблем проектирования в микроэлектронике РАН,
124365, Россия, г. Москва, г. Зеленоград, ул. Советская, д. 3.*

Аннотация. Данная статья посвящена разработке доверенных средств проектирования цифровых схем в базе гетерогенных программируемых логических интегральных схем (ПЛИС). Проектирование гетерогенных ПЛИС представляет собой одно из наиболее активно развивающихся направлений в российской микроэлектронике в настоящее время. В статье рассматриваются основные проблемы и вызовы, связанные с разработкой целевых доверенных средств проектирования. Авторы предлагают актуальный подход к разработке системы автоматизированного проектирования, основанный на использовании программных средств с открытым исходным кодом совместно с собственными наработками для её критически важных компонентов. Такой подход позволяет повысить эффективность и надёжность процесса проектирования в базе гетерогенных ПЛИС. В статье рассмотрены такие этапы маршрута проектирования цифровых схем в базе гетерогенных ПЛИС, как логический синтез и технологическое отображение, этапы топологического синтеза и статического временного анализа. Работа представляет интерес для специалистов в области микроэлектроники, а также для исследователей, занимающихся разработкой средств и систем проектирования ИС. Результаты исследования способствуют улучшению существующих методов и инструментов проектирования ИС, а также развитию и расширению отечественной электронной компонентной базы.

Ключевые слова: СБИС; ПЛИС; физическое проектирование; топологический синтез; логический синтез; размещение; трассировка; доверенное программное обеспечение.

Для цитирования: Гаврилов С.В., Железников Д.А., Заплетина М.А., Тиунов И.В., Хватов В.М., Чочаев Р.Ж., Шокарев Д.Б. Разработка доверенных средств проектирования ИС в базе гетерогенных ПЛИС. Труды ИСП РАН, том 35, вып. 5, 2023 г., стр. 107–126. DOI: 10.15514/ISPRAS–2023–35(5)–8.

Development of the Trusted Tools for IC Design on Heterogeneous FPGAs

S.V. Gavrilov, ORCID: 0000-0003-0566-4482 <s00v@yandex.ru>
D.A. Zheleznikov, ORCID: 0000-0003-2477-1793 <zheleznikov_d@ippm.ru>
M.A. Zapletina, ORCID: 0000-0001-9845-7823 <zapletina_m@ippm.ru>
I.V. Tiunov, ORCID: 0009-0008-6241-5451 <tiunov_i@ippm.ru>
V.M. Khvatov, ORCID: 0000-0003-2285-4341 <khvatov_v@ippm.ru>
R.Z. Chochoev, ORCID: 0009-0007-0304-0496 <chochoev_r@ippm.ru>
D.B. Shokarev, ORCID: 0009-0009-2806-1570 <shokarev_r@ippm.ru>

*Institute for Design Problems in Microelectronics of RAS,
3, Sovetskaya st., Zelenograd, Moscow, 124365, Russia.*

Abstract. This paper focuses on the development of trusted tools for designing digital circuits in the basis of heterogeneous field programmable gate arrays (FPGAs). Designing heterogeneous FPGAs is one of the most actively growing areas in Russian microelectronics at present. The paper discusses the main problems and challenges associated with the development of trusted computer-aided design tools. The authors propose a relevant approach to the development of a computer-aided design system based on the use of open-source software tools together with proprietary developments for its critical components. This approach allows to increase the efficiency and reliability of the design process in the basis of heterogeneous FPGAs. The paper considers such stages of the design flow in the basis of heterogeneous FPGAs as logic synthesis and technology mapping, different stages of layout synthesis and static timing analysis. The work is of interest to specialists in the field of microelectronics, as well as to researchers involved in the development of IC design tools and systems. The research results contribute to the improvement of existing IC design methods and tools, as well as to the development and expansion of the Russian electronic component base.

Keywords: VLSI; FPGA; layout synthesis; logic synthesis; placement; routing; trusted software.

For citation: Gavrilov S.V., Zheleznikov D.A., Zapletina M.A., Tiunov I.V., Khvatov V.M., Chochoev R.Z., Shokarev D.B. Development of the Trusted Tools for IC Design on Heterogeneous FPGAs. *Trudy ISP RAN/Proc. ISP RAS*, vol. 35, issue 5, 2023. pp. 107-126 (in Russian). DOI: 10.15514/ISPRAS-2023-35(5)-8.

1. Введение

Одним из наиболее активно развивающихся направлений российской микроэлектроники в настоящее время является разработка программируемых логических интегральных схем (ПЛИС). Отечественные ПЛИС постепенно выходят на российский рынок и завоёвывают свою долю популярности на фоне сложившихся сложностей с получением их зарубежных аналогов. Модельный ряд программируемых микросхем расширяется и переходит от классических гомогенных решений на основе регулярных структур из программируемых логических блоков (ПЛБ) к более сложным гетерогенным вариантам, включающим специализированные программируемые сложно-функциональные блоки (СФ-блоки).

Разработка интегральных схем с применением ПЛИС неотрывно связана с применением специализированных систем автоматизированного проектирования (САПР). Чтобы следовать в ногу за постоянно расширяющейся номенклатурой отечественных ПЛИС, ИППМ РАН занимается разработкой доверенных средств автоматизированного проектирования в базе новейших программируемых микросхем, комплексно учитывающих особенности и специфику применения последних. Выбранная стратегия решения задачи создания специализированных программных средств заключается в сочетанном применении открытых программ проектирования, анализа и оптимизации схем в базе ПЛИС и собственных разработок коллектива ИППМ РАН в области логического и топологического синтеза интегральных схем.

Разрабатываемая доверенная САПР X-CAD построена по модульному принципу и включает следующие составные части: графическую оболочку пользователя; набор подключаемых

внешних программ для технологически-независимого логического синтеза, программу X-Mar для технологического отображения и логического ресинтеза, программный модуль X-Place для работы с размещением логических вентилях, СФ-блоков и контактов проектируемой схемы; внешние программные модули для кластеризации, быстрого анализа временных характеристик и функционального моделирования, а также ядро системы для выполнения топологического синтеза, обработки прошивки и загрузки её в конфигурационную память программируемого кристалла ПЛИС.

В данной работе рассматриваются ключевые архитектурные особенности некоторых целевых отечественных ПЛИС, включая гетерогенные, а также программные средства, применяемые в маршруте проектирования ИС в их базисе. Дальнейшее содержание работы организовано следующим образом. В разделе 2 рассматриваются используемые в X-CAD инструменты логического синтеза и технологического отображения. В разделе 3 приведен обзор применяемых методов и алгоритмов топологического проектирования. Раздел 4 посвящён инструментам статического временного анализа.

2. Логический синтез и технологическое отображение

В области проектирования интегральных схем под логическим синтезом понимается процесс трансляции RTL-описания проектируемого устройства с языка описания аппаратуры в так называемый список соединений логических вентилях. На данном этапе решаются задачи преобразования исходного представления схемы в многоуровневую или одноуровневую логическую сеть (или Булеву сеть), логической оптимизации полученной сети (например, по площади или быстродействию), и отображения этой сети в список соединений логических вентилях. Целевой логический базис, в который происходит отображение, определяется, как правило, заданной библиотекой элементов, передаваемой на вход средствам логического синтеза вместе с RTL-описанием схемы. Элементы такой библиотеки всегда оптимизированы под целевую технологию. Такой подход к отображению называется технологически ориентированным.

В маршруте проектирования схем в базисе ПЛИС и других программируемых устройств, основанных на таблицах соответствия (ТС, от англ. «LUT» – Lookup Table), помимо отображения в предопределённый библиотечный базис, также существует и технологически независимый подход. В этом случае исходная схема преобразуется в набор промежуточных LUT-элементов от k -входов, или k -LUT. Здесь k – количество входов комбинационной части ПЛБ ПЛИС, которая в подавляющем большинстве архитектур представлена таблицей соответствия. Таким образом, при таком подходе, синтез не ограничен библиотекой элементов и логический базис формируется из полного набора возможных функций от k входов. Полученный список соединений, как правило, не может быть транслирован в конкретную архитектуру без дополнительных преобразований и требует внесения корректировок, учитывающих особенности целевого кристалла ПЛИС.

В разрабатываемой доверенной САПР X-CAD за этап логического синтеза отвечает инструмент с открытым исходным кодом – Yosys [1]. На текущий момент Yosys считается эталонным программным средством логического синтеза с открытым исходным кодом и используется повсеместно. За счет открытости исходного программного кода он обеспечивает довольно высокую степень доверенности, однако детальный анализ используемых алгоритмов и структур данных не проводился. К сожалению, на данный момент, Yosys является единственным возможным выбором и наиболее узким местом в процессе построения полностью доверенного маршрута проектирования, в следствии полного отсутствия отечественного аналога инструмента логического синтеза.

Программное обеспечение Yosys, а вместе с ним и САПР X-CAD, поддерживают оба упомянутых подхода к логическому синтезу: как технологически ориентированный – отображение в заданную библиотеку элементов, так и не зависящий от технологии – отображение в k -LUT элементы. С помощью Yosys в X-CAD, кроме отображения схем в

набор вентиляей, также осуществляется экстракция и отображение сложно-функциональных блоков и частичная генерация блоков ввода/вывода.

Несмотря на то, что Yosys хорошо зарекомендовал себя в качестве средства логического синтеза схем для ПЛИС, на данный момент он поддерживает лишь некоторые из зарубежных архитектур [2] и при этом – ни одной отечественной. В связи с этим, в X-CAD используется дополнительное средство для трансляции результатов синтеза Yosys в специальный внутренний формат САПР на основе языка Tcl – инструмент X-Map. Помимо преобразования Verilog-описания, создаваемого Yosys, во внутреннее Tcl-описание, он также используется в маршруте для проведения технологически ориентированного ресинтеза, а также генерации необходимых периферийных и управляющих блоков ячеек ввода/вывода. Результат работы X-Map формируется в виде внутреннего формата на основе языка Tcl - библиотеки элементов и списка соединений. Также, X-Map, при необходимости, может дополнительно сформировать результат в виде набора файлов на языке Verilog – библиотек элементов и списка соединений. Это может потребоваться, если необходимо передать результаты обработки в другие инструменты маршрута, не поддерживающих Tcl-описание. Кроме того, результат может быть дополнительно сформирован в DOT формате для дальнейшей визуализации схемы в виде графа (например, с помощью программы Graphviz).

2.1 Отображение логических элементов и логический ресинтез

Каждая из архитектур ПЛИС (как зарубежных, так и отечественных) имеет уникальный набор особенностей, учёт которых влияет не только на конечные характеристики проектируемой схемы, но и на саму возможность реализации её в целевой ПЛИС. В САПР X-CAD за учёт этих особенностей отвечает инструмент X-Map, поддерживающий маршрут синтеза схемы как в рамках заданной библиотеки, так и на основе k-LUT элементов. Результат логического синтеза в виде Verilog-описания, вместе с библиотеками элементов во внутреннем формате и командным сценарием, передаётся в инструмент X-Map, который проводит дополнительную обработку схемы, позволяя подготовить её к имплементации на ПЛИС, выдавая в качестве выходных данных список соединений уже не просто логических вентиляей, а экземпляров ПЛБ, запрограммированных на выполнение необходимых функций. Эти выходные данные и составляют внутренний формат САПР X-CAD на основе языка Tcl: список соединений элементов и библиотеки ячеек, содержащих информацию о программировании элементов: ПЛБ, ЯВВ и СФ-блоков.

Возможности X-Map в контексте задач ресинтеза и упаковки покрывают полный набор особенностей архитектур поддерживаемых отечественных ПЛИС. Кроме того, с помощью набора команд управляющего файла сценария, X-Map может быть настроен на новую архитектуру с указанием какие из возможностей по ресинтезу и упаковке поддерживаются. При этом, на данный момент, алгоритмические возможности инструмента не могут быть расширены пользователем извне, однако работа по разработке инструментов гибкой настройки процесса работы X-Map ведется.

2.1.1 Технологически ориентированный ресинтез

В случае, когда программа Yosys выполняет отображение в заданный набор библиотечных элементов, их дополнительное преобразование требуется только в том случае, когда для реализации схемы в целевой ПЛИС необходимо внести изменения в структуру самой схемы. Под «ресинтезом» здесь следует понимать процесс преобразования схемы с учетом особенностей целевой архитектуры: преобразование логических элементов (изменение подключений и логических функций), генерация источников земли и питания, вставка буферов и инверторов и др.

Например, в архитектуре ПЛИС 5400ТС015 разработки АО «Дизайн центр «Союз» [3] сигнал с тактового дерева ПЛИС может приходиться только на определённый комбинационный вход

ПЛБ. Для того чтобы учесть эту особенность архитектуры, вентиль, на вход которого приходит глобальный синхросигнал, необходимо преобразовать таким образом, чтобы этот сигнал попадал на подходящий вход. Это означает, что необходимо изменить не только подключение, но и логическую функцию, выполняемую этим вентиляем. Программа X-Map осуществляет необходимые преобразования, после чего элемент с получившейся логической функцией необходимо отобразить в заданную библиотеку. Если подходящий по функции элемент в библиотеке не найден, программа возвращает прежний вариант вентиля и внедряет в схему буфер, проходя через который, тактовый сигнал попадает на нужный вход этого вентиля.

Кроме описанного выше, X-Map может, при необходимости, генерировать дополнительные источники питания (VDD) и земли (GND) в проектируемой схеме, учитывая при этом особенность архитектуры некоторых отечественных ПЛИС – наличие на отдельных входах ПЛБ собственных источников VDD- или GND-сигналов.

2.1.2 Упаковка элементов

Подготовка схемы для имплементации в ПЛИС – не единственная задача X-Map. Основная его задача – упаковка элементов в ПЛБ и формирование набора сигналов для их программирования. В большинстве архитектур ПЛИС в ПЛБ присутствует как программируемая комбинационная часть, реализованная в виде k-LUT элемента, так и последовательная часть, представленная триггером (как правило, D-триггером). Программный модуль X-Map, используя внутренние межсоединения, может объединять в одном ПЛБ комбинационный и последовательный вентили. При этом инструмент может учитывать особенности и ограничения внутренней структуры программируемого блока. Например, в ПЛИС 5510XC3AT разработки АО «НИИМЭ» [4], в ПЛБ, основанной на 4-LUT, встроенный D-триггер может получать данные либо с выхода LUT, либо с одного из четырех информационных входов ПЛБ (рис. 1).

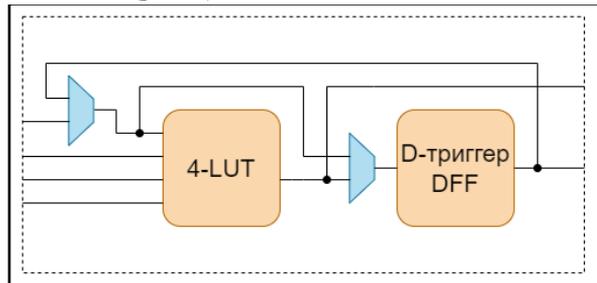


Рис. 1. Обобщенная структура ПЛБ ПЛИС 5510XC3AT.
Fig. 1. Generalized structure of 5510XC3AT FPGA's PLB.

Такая особенность ПЛБ позволяет объединить последовательно соединенные 4-LUT и триггер, или 3-LUT с триггером, независимые друг от друга (рис. 2).

В архитектуре ПЛИС 5400TC015, в отличие от 5510XC3AT, комбинационная часть представлена элементом 3-LUT, а для триггера имеется выделенный вход данных, что позволяет объединить его с независимым LUT-элементом с функцией от всех трех информационных входов (рис. 3).

В каждой из упомянутых выше архитектур также присутствует и другая особенность – наличие обратной связи с выхода триггера на вход LUT, что позволяет объединить элементы, как показано на рис. 4.

В процессе упаковки также существуют два подхода к объединению в случае множественной нагрузки: обычное объединение с активацией дополнительного входа и объединение с копированием [5]. В первом случае объединение с триггером происходит стандартным образом, описанным выше, а второй триггер подключается через прямой выход с LUT, как

показано на рис. 5. Такой вариант объединения через прямой выход LUT – очевидное решение в такой ситуации.

Второй вариант объединения – объединение с копированием – продемонстрирован на рис. 6. Данный режим используется в X-Мар по умолчанию. Его особенность заключается в том, что для комбинационного элемента создается копия, и затем каждый из двух элементов (оригинал и копия) объединяется со своим триггером. Это позволяет синхронизировать приход сигналов на каждый из триггеров, но при этом уменьшает общее количество доступных комбинационных блоков ПЛИС, что может повлиять, например, на возможность независимого объединения с другими элементами, и на реализуемость межсоединений схемы на этапе трассировки.

Благодаря использованию подобных особенностей архитектуры может быть достигнуто значительное улучшение по площади, выражаемое в количестве использованных ячеек ПЛИС и количестве межсоединений, участвующих в трассировке, относительно исходного синтезированного описания [6].

Помимо упомянутых возможностей, инструмент X-Мар также учитывает и ограничение на количество подключений ПЛБ к тактовому дереву, если такие имеются.

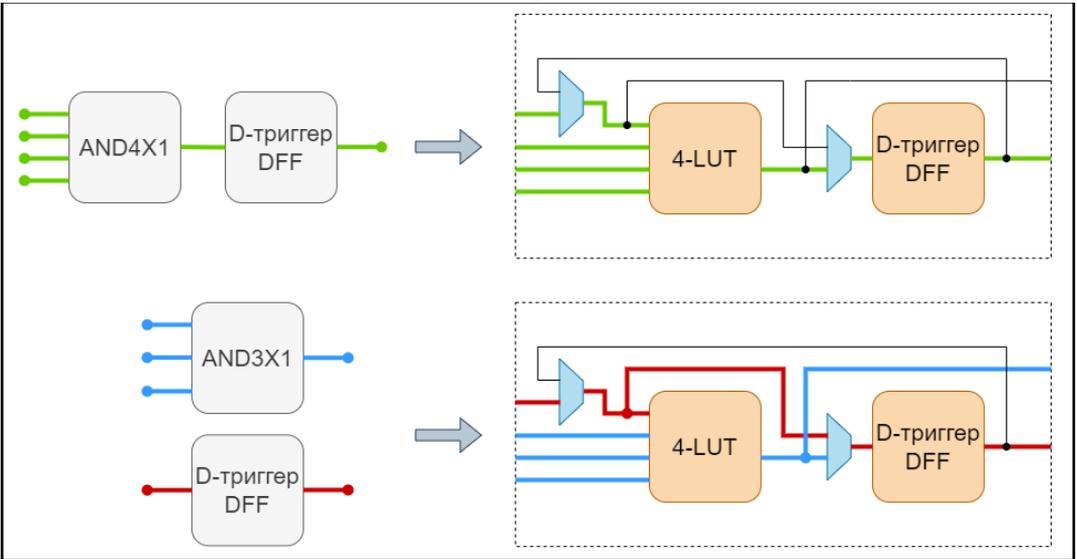


Рис. 2. Пример использования архитектурных особенностей ПЛИС 5510XC3AT в процессе упаковки элементов.

Fig. 2. An example of using the architectural features of FPGA 5510XC3AT during the packing stage.

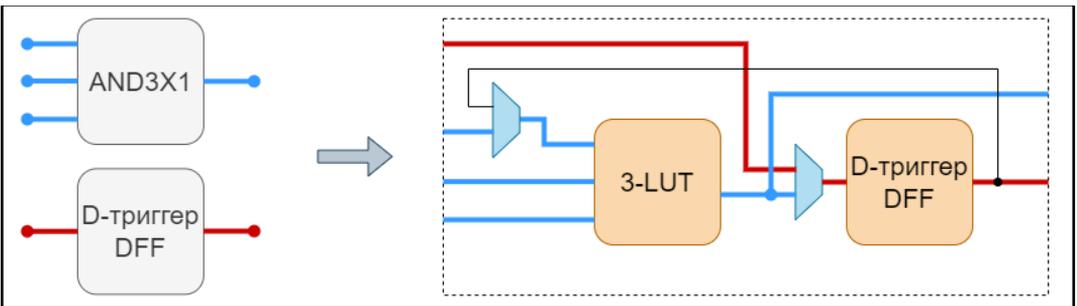


Рис. 3. Пример использования архитектурных особенностей ПЛИС 5400TC015 в процессе упаковки элементов.

Fig. 3. An example of using the architectural features of FPGA 5400TC015 during the packing stage.

2.2 Отображение контактов ввода/вывода

Отображение входных, выходных и двунаправленных контактов из функционального описания разрабатываемой схемы в библиотечные ячейки ввода/вывода (ЯВВ) в разрабатываемой САПР X-CAD выполняется совместно с помощью программы Yosys и инструмента X-Мар. Yosys генерирует в процессе логического синтеза экземпляры библиотечных ЯВВ для двунаправленных контактов, а после окончания его работы на этапе ресинтеза полученного списка соединений для генерации экземпляров ЯВВ для однонаправленных входных и выходных контактов используется X-Мар.

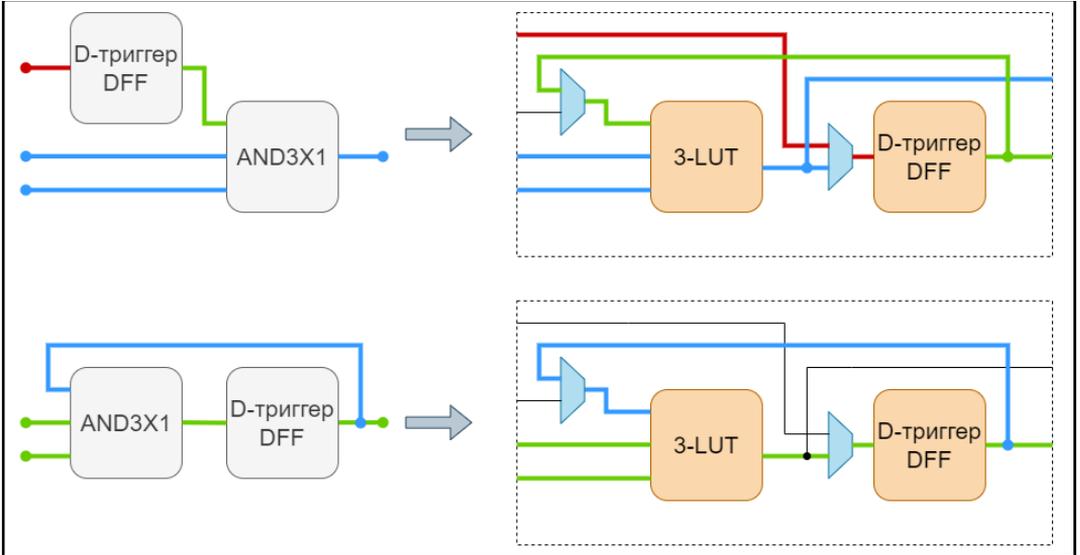


Рис. 4. Пример использования обратной связи в ПЛБ ПЛИС 5400TC015.
Fig. 4. An example of using the feedback in 5400TC015 FPGA's PLB.

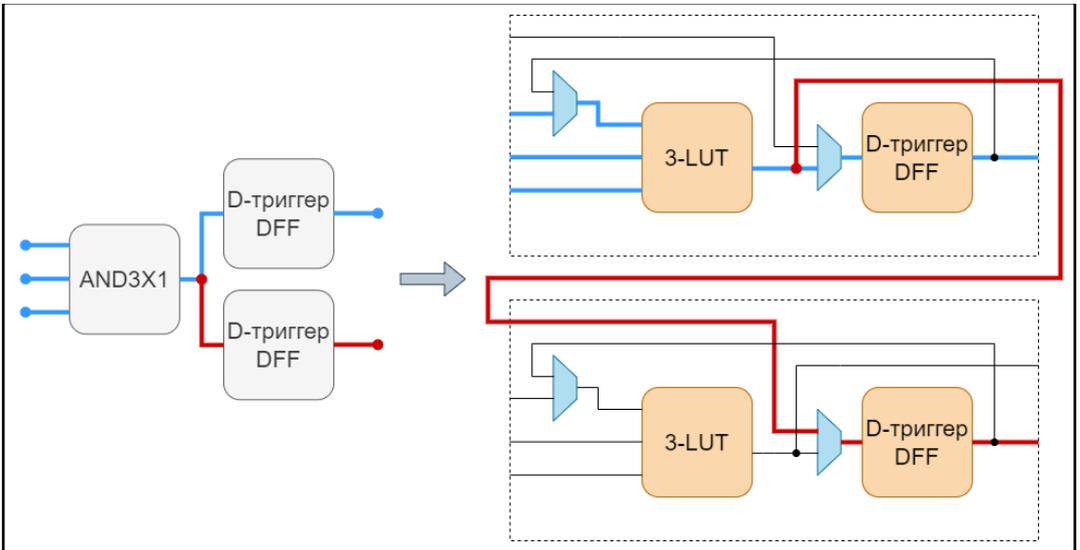


Рис. 5. Пример упаковки элементов для ПЛИС 5400TC015 в случае нескольких триггеров, соединенных с выходом комбинационного элемента.
Fig. 5. Example of packing elements for FPGA 5400TC015 in the case of several triggers connected to the output of a combinational element.

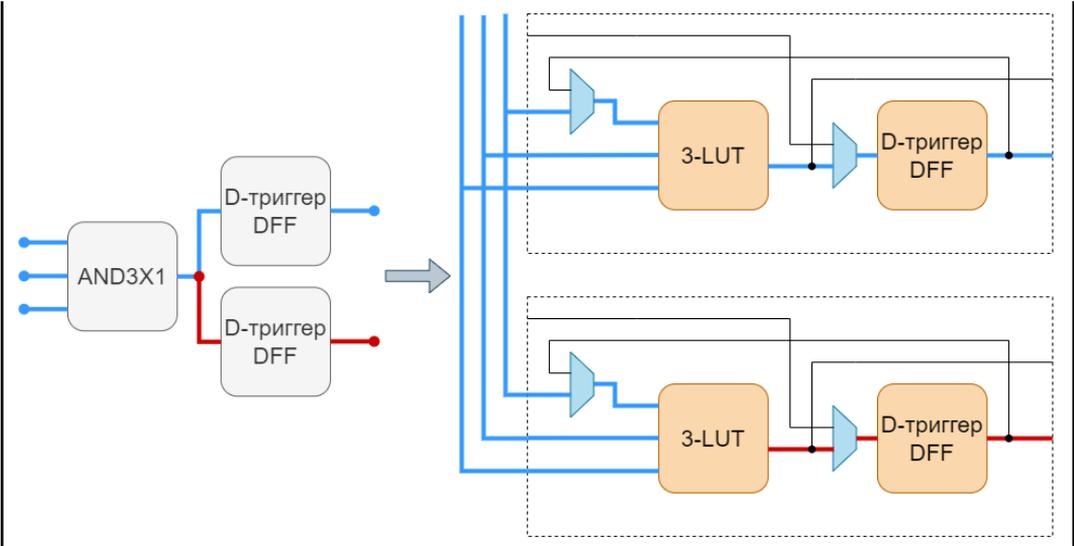


Рис. 6. Пример упаковки элементов для ПЛИС 5400TC015 в случае нескольких подсоединенных к элементу триггеров с копированием источника сигнала.

Fig.6. Example of packing elements for FPGA 5400TC015 in the case of several triggers connected to the output of a combinational element with copying.

Для генерации двунаправленных ЯВВ ПО Yosys имеет специальную команду «ioradmap», которая преобразует внешние контакты проектируемой схемы в элементы базиса ПЛИС. Данной команде требуется указать имя библиотечного элемента требуемой ячейки, его входные и выходные контакты для передачи данных, а также контакт для управления третьим состоянием выхода, соединенного с периферией ПЛИС.

X-Мар позволяет гибко настроить процесс генерации ячеек для связи с периферией под различные архитектуры ПЛИС и сформировать ЯВВ с учётом их особенностей, получая данные о доступных ЯВВ из библиотеки, разработанной во внутреннем формате САПР с помощью лингвистических средств языка Tcl, а также из управляющего сценария, содержащего дополнительные команды и данные, необходимые для работы программы. Одной из особенностей архитектуры каждой ПЛИС является количество глобальных тактовых сигналов. В случае, когда количество тактовых сигналов схемы превышает количество возможных ЯВВ ПЛИС для тактового сигнала, программа генерирует для недостающих тактовых сигналов обычные информационные ЯВВ. Очередность генерации ячеек для тактовых сигналов, при этом, определяется количеством подключений к различным блокам ПЛИС. Также X-Мар определяет ситуацию, когда контакт схемы указан как двунаправленный, но используется только в одном из направлений - вход или выход. В этом случае программа генерирует для контакта не двунаправленную ЯВВ, а ячейку нужного направления.

Задать требуемый библиотечный элемент ЯВВ и назначить его на определенный контакт ПЛИС в программе X-CAD возможно с помощью графического интерфейса X-Place. ЯВВ, назначенные таким образом, имеют приоритет над указанными в управляющем сценарии X-Мар для автоматической генерации. Генерация ЯВВ в X-Мар, по умолчанию выполняется для контактов разрабатываемой схемы, не имеющих соединения с экземпляром ЯВВ, и при необходимости может быть отключена.

Кроме ячеек ввода/вывода в библиотечный базис ПЛИС, интегрированных в X-CAD, добавлены блоки управления ячейками ЯВВ. Блоки управления присутствуют только в ряде архитектур ПЛИС, например, в архитектуре 5510XC3AT, и необходимы для

конфигурирования режима работы ячейки ввода/вывода. Они позволяют включить передачу данных от периферии к ПЛИС, от ПЛИС к периферии или в оба направления одновременно. С помощью представленных блоков можно активировать дополнительную задержку сигнала на ячейке ЯВВ, регистры, синхронизированные с ПЛИС, или режим удвоенной передачи данных (англ. Double Data Rate, DDR).

По умолчанию, для каждого из контактов функционального описания разрабатываемой схемы наряду с ЯВВ генерируется блок управления в стандартном режиме прямой передачи данных. В случае необходимости изменения режима работы ЯВВ, в схеме может быть вызвана требуемая конфигурация блока управления, доступная в библиотеке ПЛИС. X-Map позволяет настроить генерацию этих блоков, учитывая особенность их подключения в ПЛИС. Кроме того, X-Map не только учитывает уже имеющиеся во входном описании ЯВВ и управляющие блоки и генерирует недостающие, но и проверяет все имеющиеся (в том числе и заданные пользователем) ЯВВ и управляющие блоки на соответствие правлению порта (вход, выход или двунаправленный вывод) и его назначению, т.е. какого типа сигнал: информационный, синхросигнал, сигнал сброса, сигнал установки, а также подключен ли он к источнику земли или питания.

2.3 Отображение СФ-блоков

Кроме программируемых логических блоков и ячеек ввода/вывода гетерогенные ПЛИС могут содержать программируемые сложно-функциональные блоки (СФ-блоки) двух типов: жёсткие и гибкие [7]. Жёсткие СФ-блоки представляют собой полузаказные схемы, которые интегрируются в архитектуру ПЛИС, располагаясь в строках или столбцах регулярной матрицы ПЛБ. Они имеют общие коммутационные ресурсы с регулярной частью ПЛИС и, как правило, программируются с помощью конфигурационной памяти, также осуществляющей управление ПЛБ, ЯВВ и коммутационными элементами. В отличие от жёстких, гибкие СФ-блоки разрабатываются на основе логических элементов ПЛИС. Они нацелены на использование особенностей структуры групп ПЛБ и специализированных коммутационных ресурсов, не доступных при логическом синтезе на основе стандартных логических элементов. К данным ресурсам могут относиться цепи ускоренного переноса, связывающие соседние ПЛБ, цепи для прямой связи выходов со входами данных LUT соседнего ПЛБ, а также цепи для прямой связи выхода триггера со входом данных или входом тактового сигнала соседнего триггера.

Оба типа СФ-блоков разрабатываются под конкретные вычислительные задачи, что позволяет оптимизировать их структуру и схемотехнику таким образом, чтобы характеристики схем, спроектированных на их основе, были выше, чем у схем, спроектированных на основе стандартных ПЛБ ПЛИС. Использование СФ-блоков приводит к уменьшению количества используемых ПЛБ и коммутационных ресурсов, а также к повышению быстродействия разрабатываемых схем [8].

В САПР X-CAD автоматический синтез функционального описания схемы с использованием СФ-блоков и их технологическое отображение в библиотечный базис ПЛИС реализуется с помощью ПО Yosys [9]. На первом этапе логического синтеза Yosys производит экстракцию СФ-блоков из функционального описания схемы и на основе полученных результатов формирует элемент из встроенного базиса параметризованных блоков [10]. Набор блоков, которые могут быть идентифицированы таким образом, ограничен. Он включает в себя полные сумматоры, полусумматоры, вычитатели, умножители, счётчики, сдвиговые регистры, блоки цифрового процессора обработки сигналов и блоки памяти. На втором этапе Yosys выполняет отображение полученного элемента в технологически-зависимый СФ-блок из переданной ему библиотеки, разработанной под конкретную архитектуру ПЛИС. Данная библиотека разрабатывается на языке Verilog с добавлением специализированных конструкций ПО Yosys. Каждый её элемент представляет собой СФ-блок ПЛИС, сконфигурированный под определённый режим работы. Как элементы внутреннего базиса,

так и элементы разработанной библиотеки для технологического отображения, не классифицируются в Yosys по типам и являются для него «черными ящиками» с определённым набором параметров.

Связь СФ-блоков, полученных в процессе логического синтеза, с СФ-блоками из схемотехнического описания ПЛИС, т. е. с подсхемами ПЛИС, реализуется в X-CAD при помощи библиотеки для топологического синтеза, основанной на разработанном интерфейсе и лингвистических средствах языка Tcl. С помощью данной библиотеки выполняется интеграция СФ-блоков в САПР для топологического синтеза и их программирование. В состав библиотеки входят как элементы из библиотеки для технологического отображения, так и СФ-блоки, для которых невозможна автоматическая экстракция из функционального описания, в связи с их отсутствием во внутреннем базисе Yosys.

СФ-блоки для отображения в подсхемы ПЛИС, в отличие от СФ-блоков из библиотеки технологического отображения, разделены на жёсткие и гибкие. Логический базис каждого семейства ПЛИС имеет в своем составе различный набор СФ-блоков. В базис семейства ПЛИС 5510ХС входят только гибкие СФ-блоки – сумматоры с ускоренным переносом, многозарядные асинхронные счетчики и сдвиговые регистры, в состав 5510ТС, в зависимости от конкретной ПЛИС, входят как гибкие СФ-блоки, аналогичные тем, что представлены в семействе 5510ХС, так и жесткие – умножители, блоки памяти и цифровые процессоры обработки сигналов. Для добавления жёстких блоков в библиотеку используется имя подсхемы СФ-блока в ПЛИС, её информационные контакты, доступные при вызове экземпляра библиотечного элемента в проекте пользователя, а также конфигурационные контакты, которым, в соответствии с требуемым режимом работы, заданы постоянные значения логического 0 или 1 [11]. При добавлении гибких блоков используются экземпляры библиотечных ПЛБ, имена подсхем ПЛИС, их информационные и конфигурационные контакты. В библиотечном гибком СФ-блоке доступно определение ограничений топологического синтеза, к которым относится размещение ПЛБ относительно друг друга, соединение их между собой и назначение межсоединениям ПЛИС конкретных цепей из библиотечного СФ-блока. Корректное конфигурирование и назначение ограничений позволяет учесть особенности архитектуры жёстких блоков, а также особенности архитектуры ПЛИС, преимущества структуры группы ЛБ и специализированные коммутационные ресурсы при проектировании гибких СФ-блоков.

3. Топологический синтез

На этапе топологического синтеза в маршруте проектирования в базисе ПЛИС выполняется размещение элементов схемы и трассировка существующих соединений [12]. Традиционно процесс размещения включает в себя этапы глобального и детального размещения [13]. На глобальном размещении выполняется определение предварительного расположения элементов на кристалле, после чего на этапе детального размещения выполняется легализация, необходимая для назначения элементов на корректные посадочные площадки, и оптимизация полученного размещения по таким критериям, как длина соединений, задержки, потребляемая мощность и др. Программные модули размещения и трассировки являются собственной разработкой ИППМ РАН.

3.1 Кластеризация по группам логических блоков

В САПР X-CAD на этапе глобального размещения выполняется размещение групп программируемых логических блоков с учётом особенностей архитектуры целевой ПЛИС. Список соединений, полученный после этапа логического синтеза, представляется в виде гиперграфа $G=(V, E)$, где V – множество вершин (элементов схемы), E – множество гиперребер (соединений), после чего для формирования групп логических блоков используются специальные алгоритмы кластеризации гиперграфов.

Для кластеризации существуют различные программы и методы [14-16], среди которых наибольшей популярностью в академической среде обладает одна из самых старейших – hMETIS [17]. К сожалению, исходный код данной программы закрыт и последнее обновление было выпущено более двадцати лет назад. Среди программ с открытым исходным кодом [18-21] одним из наиболее эффективных и быстро развивающихся программ является программа KaHyPar [21]. В отличие от программ, использующих классические алгоритмы, в данной программе для получения оптимального разбиения используется комбинация алгоритмов. В программе KaHyPar пользователю доступны различные конфигурации запуска, которые позволяют выбрать требуемый алгоритм разбиения гиперграфа и целевую функцию, подходящую под конкретную задачу. Открытый исходный код программы, а также тот факт что на этапе кластеризации невозможна модификация списка соединений в следствии работы программы с абстрактным графом, данная программа была выбрана для интеграции в САПР X-CAD.

На вход KaHyPar подается гиперграф (список соединений), количество требуемых кластеров (групп логических блоков) и конфигурационный файл с настройками запуска алгоритмов. Выходной файл программы KaHyPar с кластеризованным гиперграфом конвертируется в специальный файл на языке Tcl, который загружается в САПР X-CAD. Так как гиперграф содержит вершины, представляющие как логические элементы, так и СФ-блоки из списка соединений, то при конвертации в Tcl-файл СФ-блоки удаляются из кластеров для получения только групп логических элементов.

Экспериментальные результаты, полученные в предыдущих работах [22], показали эффективность использования программы KaHyPar в маршруте проектирования в базисе ПЛИС. Однако, одним из недостатков KaHyPar является отсутствие учёта важных особенностей архитектуры групп программируемых логических блоков, например, количества доступных глобальных тактовых сигналов в пределах одной группы. Поэтому при применении программы KaHyPar в маршруте проектирования в базисе ПЛИС требуется дополнительная легализация полученного разбиения с учётом особенностей целевых архитектур ПЛИС.

3.2 Размещение логических элементов

Большинство современных ПЛИС имеет иерархическую архитектуру [12], поэтому для получения оптимального размещения обычно используются методы и алгоритмы, выполняющие размещение для каждого уровня иерархии по отдельности. В маршруте топологического синтеза в базисе ПЛИС в программе X-CAD на этапе размещения также применяется двухуровневый подход, описанный в работе [13]. Суть данного подхода заключается в разбиении размещения на глобальный и детальный этапы.

На этапе глобального размещения в X-CAD выполняется размещение групп программируемых логических блоков, полученных после кластеризации программой KaHyPar. Глобальный этап включает в себя два последовательных подэтапа: начальное размещение и оптимизация полученного размещения эвристическими методами. Начальный этап необходим для быстрой генерации предварительного размещения ПЛБ для его последующей оптимизации [23]. Существует множество различных методов и алгоритмов генерации начального размещения [24], среди которых популярны случайный и силовые методы [25]. В методе случайного размещения для каждого элемента из списка соединений последовательно генерируются случайные координаты посадочных мест. Данный подход малоэффективен, и не гарантирует детерминированных результатов. Поэтому в САПР X-CAD используется силовой алгоритм размещения [24]. В силовом алгоритме связи между размещаемыми элементами представляются в виде сил взаимного притяжения, которые определяют конечные координаты. Чем больше соединений между логическими элементами, тем выше сила притяжения и тем ближе друг к другу будут расположены элементы на кристалле. Благодаря тому, что сильно связанные элементы размещены ближе друг к другу,

достигается сокращение суммарной длины цепей и повышается скорость сходимости к оптимальному решению последующего алгоритма оптимизации размещения.

Для оптимизации начального размещения в X-CAD на глобальном этапе используются модифицированные алгоритмы размещения на основе метода имитации отжига [13]. В среде открытых и коммерческих САПР они завоевали наибольшую популярность. Преимуществами данного метода являются легкость программной реализации, простота учёта разных критериев оптимизации, возможность оптимизации нелинейных целевых функций. Основным отличием модифицированных алгоритмов является учёт архитектурных особенностей целевых ПЛИС, без которого невозможно получение легального размещения.

Этап детального размещения в X-CAD включает в себя те же шаги, что и этап глобального размещения. На первом шаге выполняется генерация начального размещения с помощью силового алгоритма для логических элементов в каждой группе с учётом результатов глобального размещения. На втором шаге выполняется оптимизация полученного размещения модифицированными алгоритмами на основе метода имитации отжига [26].

Помимо двухуровневого размещения в программе X-CAD поддерживается и плоское размещение логических элементов без предварительного этапа кластеризации [26]. В плоском варианте выполняется размещение логических элементов без учёта их расположения в группах на кристалле. Однако, чтобы не допустить слишком плотного размещения, которое может привести к нетрассируемому решению, вводится дополнительное ограничение на максимальное количество элементов в пределах групп. Экспериментальные результаты, полученные в предыдущих работах [22], показали, что в трассировке после плоского размещения используется в среднем меньше коммутационных элементов, чем после двухуровневого с алгоритмом KaNuPar, однако она работает заметно медленнее.

3.3 Трассировка межсоединений

Решение задачи трассировки в маршруте топологического синтеза в базисе ПЛИС представляет собой поиск детального отображения проектных межсоединений на имеющиеся на кристалле ПЛИС коммутационные ресурсы. Последние представлены совокупностью коммутационных магистралей и отдельных элементов, программируемых сигналами из общей конфигурационной памяти. Если на предыдущих этапах проектирования формируется часть прошивки (матрицы значений ячеек конфигурационной памяти), отвечающая за программирование логической части ПЛИС (ЯВВ, СФ-блоков и ПЛБ), то результат этапа трассировки является основой для формирования оставшейся части прошивки, определяющей программирование всей коммутационной сети ПЛИС. Обязательным требованием для успешного решения задачи трассировки является формирование целостных и непересекающихся путей для всех проектных межсоединений, исключающих короткие замыкания между ними, т.н. требование бесконфликтной разводки.

Работа модуля трассировки в составе разрабатываемой САПР X-CAD основывается на собственной реализации известного алгоритма трассировки Pathfinder [27], дополненного рядом модификаций. Алгоритм Pathfinder имеет доказанную высокую эффективность для решения задачи трассировки на ПЛИС. Он активно исследуется, развивается [28, 29] и в настоящее время применяется в составе крупнейших открытых САПР: OpenLane/OpenROAD [30], VPR [31], F4PGA [32].

Коммутационные ресурсы значительной части отечественных ПЛИС (например, серий 5510XC, 5510TC, схемы 5400TC015, цифровой части 5400TC194 и др.) могут быть формально описаны смешанным графом [33] $G=(V,E)$, вершины $v_i, v_j \in V$ которого соответствуют неразрывным коммутационным магистралям и контактам ПЛБ, СФ-блоков и ЯВВ, а ребра и дуги $e_{ij}, e_{ij} \in E, e_{ij}=(v_i, v_j)$ – коммутационным элементам между ними. Важные схемотехнические особенности коммутационных ресурсов этих схем снижают

эффективность применения оригинального алгоритма Pathfinder, существующих открытых решений и классической модели ориентированного графа для представления коммутационной сети, что потребовало внесения модификаций в алгоритм трассировки и разработки модели смешанного графа. В частности, такими особенностями являются применение широкого набора трассировочных элементов, в том числе, специализированных, дублирование и редуцирование коммутационных возможностей для повышения надежности схем ПЛИС и экономии площади кристалла в рамках используемой кремниевой технологии. С точки зрения доверенности собственная реализация алгоритма также позволяет гарантировать отсутствие возможности внесения сторонних изменений на этапе трассировки межсоединений.

Общий принцип работы алгоритма Pathfinder состоит в итерационной последовательной трассировке списка соединений проектируемой схемы с учётом результатов предшествующих итераций. Суммарное их количество определяется сложностью задачи трассировки. Она, в свою очередь, зависит от степени перегруженности коммутационных ресурсов, необходимых для реализации всех соединений между отдельными логическими блоками, блоками ввода-вывода и сложно-функциональными блоками схемы, расстановка которых была получена на этапе размещения. Перегруженными считаются те коммутационные магистрали и отдельные элементы, которые задействованы в трассировке более чем одного проектного межсоединения. К завершению трассировки все перегруженные области должны быть устранены, поскольку в процессе функционирования схемы на ПЛИС они способны вызвать короткое замыкание между разнопотенциальными участками цепей.

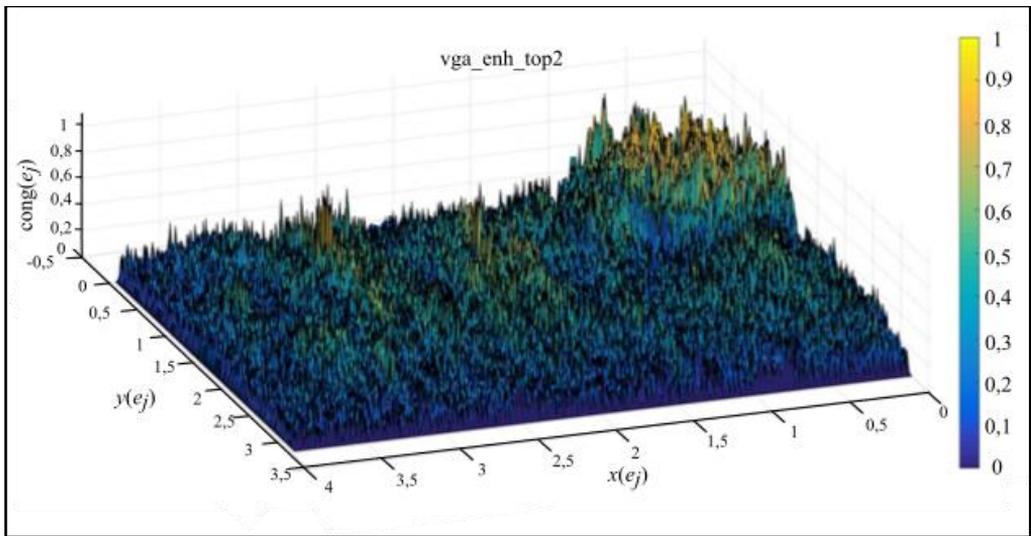


Рис. 7. Тепловая карта перегруженности коммутационных ресурсов на примере схемы vga_enh_top [31], $x(e_j)$, $y(e_j)$ — координаты трассировочного элемента e_j .

Fig. 7. Congestion heatmap for routing resources for example circuit vga_enh_top [31], $x(e_j)$, $y(e_j)$ are coordinates of routing element e_j .

Алгоритм Pathfinder и его модификации не требуют сведения трассировки межсоединений к задаче целочисленного линейного программирования с множеством ограничений для соблюдения требования бесконфликтной разводки. Необходимый эффект сходимости итерационного процесса к корректному трассировочному решению достигается учётом истории перегруженности трассировочных ресурсов. Графически перегруженность можно представить в виде тепловой диаграммы (рис. 7), где оси абсцисс и ординат образуют координатную сетку, в узлах которой расположены трассировочные элементы кристалла

ПЛИС, а по оси аппликата отмечается величина нормализованной перегруженности, которая рассчитывается как

$$\text{cong}(e_{ij}) = \frac{\text{nets}(e_{ij})}{\max(\text{nets}(e_{ij}))},$$

где $\text{nets}(e_{ij})$ – количество проектных межсоединений, в разводке которых участвует перегруженный ресурс.

Для наиболее эффективной работы модуля трассировки в составе разрабатываемой САПР X-CAD произведено внесение модификаций в классическую версию алгоритма Pathfinder. В первую очередь, они направлены на детальный учёт особенностей коммутационных ресурсов новейших отечественных ПЛИС и реконфигурируемых систем на кристалле для достижения наилучших выходных характеристик проектируемых схем [35]. Например, к таким особенностям нужно отнести неклассическую структуру программируемых связей, которая вызвана технологическими ограничениями проектирования. Схемотехнически она реализована посредством разреженности коммутационной сети и внедрения широкого набора коммутационных элементов с комплексным управлением проводимостью. Во вторую очередь, модификации классического Pathfinder позволяют значительно ускорить [36] прохождение этапа трассировки без значительного ухудшения итоговых результатов.

4. Статический временной анализ

Инструменты статического временного анализа играют ключевую роль в оценке результатов топологического синтеза, позволяя проверить соответствие временных и мощностных характеристик схемы заданным ограничениям. В САПР X-CAD для выполнения статического временного анализа было интегрировано открытое программное обеспечение OpenSTA [37], имеющее широкий спектр возможностей. В отличие от аналогичных программных продуктов, таких как iSTA [38] или OpenTimer [39], данная программа включает в себя многие возможности, позволяющие качественно и эффективно проводить статический временной анализ. Основные достоинства OpenSTA заключаются в способности работать с иерархическим списком соединений и поддержке учёта вариаций на кристалле (от англ. On-Chip Variation, OCV). Кроме того, программа OpenSTA проверяет, что тактовый сигнал распространяется через блоки стробирования синхросигнала (англ. clock gating), и выполняет анализ мощности, что даёт дополнительные возможности при работе с энергоэффективными схемами. Немаловажным является тот факт, что OpenSTA обладает широким набором команд управления, распознаваемых его собственным интерпретатором команд на языке Tcl. Благодаря этому, инструмент статического временного анализа может быть легко интегрирован в сторонние программные продукты, сохраняя свою функциональность.

Алгоритм работы с OpenSTA в составе САПР X-CAD состоит из нескольких этапов. Сначала задаются подключаемые Verilog файлы проекта и файлы библиотек в формате Liberty, после чего происходит сборка проекта. Далее пользователем задаются проектные ограничения и указывается требуемый формат результатов анализа. Так как работа с программой OpenSTA осуществляется посредством её собственных команд, то для взаимодействия с САПР X-CAD используется управляющий сценарий на языке Tcl, который на основе заданных временных ограничений создаёт командный файл, передаваемый программе OpenSTA. При этом ограничения можно задать как с помощью параметров консоли, так и с помощью сформированных файлов в формате Synopsys Design Constraints (SDC).

В маршруте проектирования ИС на основе ПЛИС, реализованном в САПР X-CAD, временной анализ проводится два раза: после логического и после топологического синтеза. Анализ, выполняющийся первым, называется предварительным. Анализ после размещения и трассировки называется итоговым. В ходе предварительного анализа анализируется список соединений после этапа технологического отображения. В процессе итогового анализа

список соединений дополняется коммутационными элементами и ёмкостями межсоединений, полученными в результате трассировки межсоединений.

Для формирования ёмкостей для каждого кристалла ПЛИС проводится паразитная экстракция и формируется специальный файл, в котором каждому межсоединению поставлена в соответствие паразитная ёмкость. При работе над проектом в САПР после трассировки генерируется файл, содержащий успешно трассированные цепи и связанные с ними ёмкости. Данный файл в ходе статического временного анализа передается программе OpenSTA. Процесс генерации и загрузки этого файла в программу полностью автоматизирован и не требует вмешательства пользователя.

Важным преимуществом OpenSTA является возможность выполнить статический временной анализ иерархического списка соединений. Данная функциональность необходима, поскольку в ходе технологического отображения проектируемой схемы, комбинационные и последовательностные элементы могут быть объединены в ПЛБ ПЛИС. Список соединений, содержащий подобные элементы, не может напрямую использоваться для статического временного анализа, так как они отсутствуют в библиотеках ПЛИС в явном виде. Однако, используя возможность OpenSTA работать с иерархическим представлением схемы и используя дополнительную библиотеку объединенных элементов становится возможным провести его временной анализ без генерации дополнительных списков соединений.

5. Заключение

Создание доверенных программных средств проектирования микроэлектроники позволяет избежать зависимости от зарубежных производителей коммерческих систем и способствует появлению современных решений в области формирования отечественной электронной компонентной базы. В данной работе рассмотрены основные особенности разработки отечественной доверенной САПР для проектирования в базисе гетерогенных ПЛИС. В частности, дан обзор основных используемых программных средств и алгоритмов, применяемых для построения доверенной САПР X-CAD. Рассмотрен полный маршрут проектирования ИС в базисе гетерогенных ПЛИС, включая логический синтез и технологическое отображение, этапы топологического синтеза и статического временного анализа. Показаны основные преимущества применяемых методов и подходов.

В процессе разработки доверенных программных средств и систем ИПМ РАН придерживается подхода, состоящего в сочетанном применении открытых программных средств проектирования и использовании собственных оригинальных разработок в области создания САПР ИС. Он не лишен недостатков, в частности, проявляющихся в сложности получения технической поддержки для открытых программных решений. Однако такая стратегия позволяет использовать наилучшие актуальные наработки научного сообщества в области САПР ИС, ускоряя процесс разработки и перенося наибольшую часть трудозатрат на разработку собственных решений для критически важных компонентов системы.

Список литературы / References

- [1]. Wolf C., Glaser J. Yosys - A Free Verilog Synthesis Suite. [Электронный ресурс] // Proceedings of the 21st Austrian Workshop on Microelectronics (Austrochip). 2013. URL: <https://yosyshq.net/yosys/files/yosys-austrochip2013.pdf> (дата обращения: 25.10.2023).
- [2]. Yosys Open SYnthesis Suite. Frequently Asked Questions [Электронный ресурс] // URL: <https://yosyshq.net/yosys/faq.html> (дата обращения: 26.10.2023).
- [3]. 5400TC015 Программируемая логическая интегральная схема (ПЛИС) [Электронный ресурс] // URL: <https://dcsouyz.ru/products/pais/art/1727> (дата обращения: 25.10.2023).
- [4]. ПЛИС емкостью 145 тыс. системных вентилях 5510XC3AT [Электронный ресурс] // URL: <https://mikron.ru/products/high-rel-ic/programmiruemyaya-logika-fpga/fpga/product/5510hs3at/> (дата обращения: 25.10.2023).

- [5]. Tiunov I.V., Lipatov I.A., Zheleznikov D.A. Digital Circuits Resynthesis Approach for FPGAs Based on Logic Cell with Built-In Flip-Flop, Problems of advanced micro- and nanoelectronic systems development, 2019, pp. 33-36. DOI 10.31114/2078-7707-2019-3-33-36.
- [6]. Тиунов И.В. Методы ресинтеза схем для ПЛИС на основе ячеек с разделенными выходами и обратной связью // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. №2. С. 50-56. DOI: 10.31114/2078-7707-2020-2-50-56. / Tiunov I.V. Resynthesis methods for FPGAs based on cells with separated outputs and built-in feedback // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2020. Issue 2. P. 50-56 (in Russian). DOI:10.31114/2078-7707-2020-2-50-56.
- [7]. Хватов В. М., Гаврилов С. В. Формирование библиотек СФ-блоков в маршруте проектирования пользовательских схем на ПЛИС и РСнК // Известия высших учебных заведений. Электроника. – 2021. – Т. 26, № 5. – С. 387-398. DOI: 10.24151/1561-5405-2021-26-5-387-398. / Khvatov V.M., Gavrilov S.V. Complex functional block libraries formation in the design flow of user circuits on FPGA and RSoC. Proc. Univ. Electronics, 2021, vol. 26, no. 5, pp. 387–398 (in Russian). DOI: 10.24151/1561-5405-2021-26-5-387-398.
- [8]. Khvatov V. M., Zheleznikov D. A., Gavrilov S. V. Analysis of the Programmable Soft IP-cores Implementation for FPGAs," 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), St. Petersburg, Moscow, Russia, 2021, pp. 2681-2685.
- [9]. Shah D., Hung E., Wolf C. et. al. Yosys+nextpnr: An Open Source Framework from Verilog to Bitstream for Commercial FPGAs. 2019. pp. 1-40.
- [10]. Yosys Manual, techmap - generic technology mapper. [Электронный ресурс] URL: <https://yosyshq.readthedocs.io/projects/yosys/en/latest/cmd/techmap.html> (дата обращения: 25.10.2023).
- [11]. Хватов В. М., Гарбулина Т. В., Лялинская О. В. Методы формирования и верификации библиотек стандартных элементов в составе маршрута проектирования ИС на базе ПЛИС отечественного производства // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). – 2018. – № 1. – С. 57-62. DOI:10.31114/2078-7707-2018-1-57-62. / Khvatov V.M., Garbulina T.V., Lyalinskaya O.V. Formation and Verification of Standard Element Libraries in the Design Flow for the Domestic FPGAs // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2018. Issue 1. P. 57-62 (in Russian). DOI:10.31114/2078-7707-2018-1-57-62.
- [12]. Hauck, S. Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation / S. Hauck, A Denon. –, 2008. – 944 p.
- [13]. Фролова П.И., Чочаев Р., Иванова Г.А., Гаврилов С.В. Алгоритм размещения с оптимизацией быстродействия на основе матриц задержек для реконфигурируемых систем на кристалле // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. Выпуск 1. С. 2-7. DOI:10.31114/2078-7707-2020-1-2-7. / Frolova P.I., Chochev R., Ivanova G.A., Gavrilov S.V. Timing-driven placement algorithm based on delay matrix model for reconfigurable system-on-chip // Problems of Perspective Micro- and Nanoelectronic Systems Development - 2020. Issue 1. P. 2-7 (in Russian). DOI:10.31114/2078-7707-2020-1-2-7.
- [14]. Miettinen P., Honkala M., Roos J. Using METIS and hMETIS Algorithms in Circuit Partitioning // Report ST49, Circuit Theory Laboratory, Helsinki University of Technology, 2006.
- [15]. Devine K. D., Boman E.G., Riesen L.A., Catalyurek U.V., Chevalier C. Getting started with zoltan: A short tutorial. // In Combinatorial Scientific Computing №09061 in Dagstuhl Seminar Proceedings, 2009, p. 10.
- [16]. Çatalyürek Ü., Aykanat C. PaToH (Partitioning Tool for Hypergraphs) // Encyclopedia of Parallel Computing, 2020. Springer US, Boston, MA, pp. 1479-1487. DOI: 10.1007/978-0-387-09766-4_93
- [17]. Karypis G., Kumarh V. hMETIS* A Hypergraph Partitioning Package Version 1.5.3 [Электронный ресурс] // Minnesota. 1998. URL: <https://course.ece.cmu.edu/~ee760/760docs/hMetisManual.pdf> (дата обращения: 13.10.2023).
- [18]. The first version of TritonPart: программа. / ABKGroup. URL: <https://github.com/ABKGroup/TritonPart> (дата обращения: 17.11.2023)
- [19]. FREIGHT: Fast stREamInG Hypergraph parTitioning: программа. / KaHIP. Лицензия: MIT. URL: <https://github.com/KaHIP/FREIGHT> (дата обращения: 17.11.2023)
- [20]. Mt-KaHyPar - Multi-Threaded Karlsruhe Graph and Hypergraph Partitioner: программа. / KaHyPar. Лицензия: MIT. URL: <https://github.com/kahypar/mt-kahypar> (дата обращения: 17.11.2023)

- [21]. Schlag S., Heuer T., Gottesbüren L., et. al. High-Quality Hypergraph Partitioning // *ACM J. Exp. Algorithmics* Just Accepted, Association for Computing Machinery, New York, 2022. DOI: 10.1145/3529090.
- [22]. Фролова П.И., Хватов В.М., Чочаев Р. Сравнительный анализ методов кластеризации и размещения схем для реконфигурируемых систем на кристалле // *Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС)*. 2022. Выпуск 4. С. 63-70. doi:10.31114/2078-7707-2022-4-63-70. / Frolova P.I., Khvatov V.M., Chochaev R. Comparative Analysis of Clustering and Placement Methods for Reconfigurable System-on-Chips // *Problems of Perspective Micro- and Nanoelectronic Systems Development - 2022. Issue 4. P. 63-70. (in Russian)*. DOI:10.31114/2078-7707-2022-4-63-70.
- [23]. Lin Z., Xie Y., Qian G., et. al. Late Breaking Results: An Analytical Timing-Driven Placer for Heterogeneous FPGAs // *2020 57th ACM/IEEE Design Automation Conference (DAC)*, 2020, pp. 1-2, DOI: 10.1109/DAC18072.2020.9218699.
- [24]. Фролова П.И., Чочаев Р. Разработка и сравнительный анализ методов начального размещения на ПЛИС // *Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС)*. 2021. Выпуск 3. С. 57-64. doi:10.31114/2078-7707-2021-3-57-64. / Frolova P.I., Chochaev R. Development and Comparative Analysis of Initial Placement Methods for FPGA // *Problems of Perspective Micro- and Nanoelectronic Systems Development - 2021. Issue 3. P. 57-64 (in Russian)*. DOI:10.31114/2078-7707-2021-3-57-64.
- [25]. Eisenmann H., Johannes F. M. Generic global placement and floorplanning // *Proceedings 1998 Design and Automation Conference. 35th DAC. (Cat. No.98CH36175)*, 1998, pp. 269-274, DOI: 10.1145/277044.277119.
- [26]. Гаврилов С.В., Железников Д.А., Чочаев Р.Ж. Разработка и сравнительный анализ методов решения задачи размещения для реконфигурируемых систем на кристалле // *Изв. вузов. Электроника*. 2020. Т. 25. № 1. С. 48–57. DOI: 10.24151/1561-5405-2020-25-1-48-57. / Gavrilov S.V., Zheleznikov D.A., Chochaev R.Z. Development and comparative analysis of placement methods for reconfigurable systems-on-a-chip. *Proc. Univ. Electronics*, 2020, vol. 25, no. 1, pp. 48–57 (in Russian). DOI: 10.24151/1561-5405-2020-25-1-48-57.
- [27]. McMurchie L., Ebeling C. PathFinder: A negotiation-based performance-driven router for FPGAs. *Proceedings of the 1995 ACM third international symposium on Field-programmable gate arrays*, 1995, pp. 111-117.
- [28]. D. Wang, J. Feng, K. Liu, W. Zhou, X. Hao and X. Zhang, "A Fast FPGA Connection Router Using Prerouting-Based Parallel Local Routing Algorithm," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 42, no. 11, pp. 3868-3880, Nov. 2023, DOI: 10.1109/TCAD.2023.3274950.
- [29]. Mai, J., Wang, J., Di, Z., et.al. OpenPARF: An Open-Source Placement and Routing Framework for Large-Scale Heterogeneous FPGAs with Deep Learning Toolkit [Электронный ресурс] // *arXiv preprint arXiv:2306.16665*. 2023. URL: <https://arxiv.org/pdf/2306.16665.pdf> (Дата обращения: 20.10.2023).
- [30]. OpenROAD's unified application implementing an RTL-to-GDS Flow [Электронный ресурс] // URL: <https://github.com/The-OpenROAD-Project/OpenROAD> (дата обращения: 30.10.2023).
- [31]. Verilog to Routing – Open Source CAD Flow for FPGA Research: программа / VTR Development Team. Лицензия: MIT. URL: <https://github.com/verilog-to-routing/vtr-verilog-to-routing> (дата обращения: 17.11.2023)
- [32]. F4PGA: официальный сайт. URL: <https://f4pga.org/> (дата обращения: 17.11.2023)
- [33]. Заплетина М. А. Решение задачи трассировки на ПЛИС с применением модели расширенного смешанного графа коммутационных ресурсов // *Изв. вузов. Электроника*. 2022. Т. 27. № 6. С. 774–786. DOI: 10.24151/1561-5405-2022-27-6-774-78. / Zapletina M. A. Solving the FPGA routing problem using the model of an extended mixed routing graph. *Proc. Univ. Electronics*, 2022, vol. 27, no. 6, pp. 774–786 (in Russian). DOI:10.24151/1561-5405-2022-27-6-774-786.
- [34]. The OpenCores VGA/LCD Controller [Электронный ресурс] // URL: https://opencores.org/projects/vga_lcd (дата обращения: 30.10.2023).
- [35]. Zapletina M. A., Gavrilov S. V. Pathfinder Algorithm Modification for FPGA Routing Stage. *Russian Microelectronics*, 51(7), pp. 573-578. DOI: 10.1134/S1063739722070125.
- [36]. Zapletina M. A., Zheleznikov D. A., Gavrilov S. V. Improving Pathfinder Algorithm Performance for FPGA Routing // *2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus)*, St. Petersburg, Moscow, Russia, 2021, pp. 2054-2057, DOI: 10.1109/ElConRus51938.2021.9396608.

- [37]. Ajayi T., Blaauw D., Chan T.-B., et. al. OpenROAD: Toward a Self-Driving, Open-Source Digital Layout Implementation Tool Chain // Proc. Government Microcircuit Applications and Critical Technology Conference, 2019, pp. 1105-1110.
- [38]. Li X., Tao S., Huang Z., et al. iEDA: An Open-Source Intelligent Physical Implementation Toolkit and Library [Электронный ресурс] // arXiv preprint arXiv:2308.01857. 2023. URL: <https://arxiv.org/pdf/2308.01857.pdf> (дата обращения: 10.10.2023).
- [39]. Huang T. W., Wong M. D. F. OpenTimer: A high-performance timing analysis tool // 2015 IEEE/ACM International Conference on Computer-Aided Design (ICCAD). – IEEE, 2015. pp. 895-902. DOI: 10.1109/ICCAD.2015.7372666.

Информация об авторах / Information about authors

Сергей Витальевич ГАВРИЛОВ – доктор технических наук, профессор, директор Института проблем проектирования в микроэлектронике РАН. Область научных интересов: автоматизация проектирования микроэлектроники, САПР ИС.

Sergey Vitalievich GAVRILOV – Dr. Sci. (Tech.), Professor, CEO of Institute for Design Problems in Microelectronics of RAS. Research interests: electronic design automation, VLSI computer-aided design.

Даниил Александрович ЖЕЛЕЗНИКОВ – кандидат технических наук, старший научный сотрудник отдела САПР ИС Института проблем проектирования в микроэлектронике РАН. Область научных интересов: автоматизация проектирования микроэлектроники, САПР ИС.

Daniil Aleksandrovich ZHELEZNIKOV – Cand. Sci. (Tech.), Senior Research Scientist of Institute for Design Problems in Microelectronics of RAS. Research interests: electronic design automation, VLSI computer-aided design.

Мария Андреевна ЗАПЛЕТИНА – кандидат технических наук, научный сотрудник отдела САПР ИС Института проблем проектирования в микроэлектронике РАН. Область научных интересов: автоматизация проектирования микроэлектроники, САПР ИС, ПЛИС, РСнК.

Mariia Andreevna ZAPLETINA – Cand. Sci. (Tech.), Research Scientist of Institute for Design Problems in Microelectronics of RAS. Research interests: electronic design automation, VLSI computer-aided design, FPGA, SoC FPGA.

Иван Викторович ТИУНОВ – младший научный сотрудник отдела САПР ИС Института проблем проектирования в микроэлектронике РАН. Область научных интересов: автоматизация проектирования микроэлектроники, САПР ИС, ПЛИС, логический синтез, графические интерфейсы.

Ivan Victorovich TIUNOV – Junior Research Scientist of Institute for Design Problems in Microelectronics of RAS. Research interests: electronic design automation, VLSI computer-aided design, FPGA, logic synthesis, GUI.

Василий Михайлович ХВАТОВ – научный сотрудник отдела САПР ИС Института проблем проектирования в микроэлектронике РАН. Область научных интересов: автоматизация проектирования микроэлектроники, САПР ИС, ПЛИС, РСнК.

Vasili Mikhaïlovich KHVATOV – Research Scientist of Institute for Design Problems in Microelectronics of RAS. Research interests: electronic design automation, VLSI computer-aided design, FPGA, SoC FPGA.

Рустам Жамболатович ЧОЧАЕВ – инженер-исследователь отдела САПР ИС Института проблем проектирования в микроэлектронике РАН. Область научных интересов: автоматизация проектирования микроэлектроники, САПР ИС, ПЛИС, РСнК.

Rustam Zhambolatovich CHOCHAEV – Research Engineer of Institute for Design Problems in Microelectronics of RAS. Research interests: electronic design automation, VLSI computer-aided design, FPGA, SoC FPGA.

Дмитрий Борисович ШОКАРЕВ – инженер-исследователь отдела САПР ИС Института проблем проектирования в микроэлектронике РАН. Область научных интересов: автоматизация проектирования микроэлектроники, САПР ИС, ПЛИС, РСнК.

Dmitry Borisovich SHOKAREV – Research Engineer of Institute for Design Problems in Microelectronics of RAS. Research interests: electronic design automation, VLSI computer-aided design, FPGA, SoC FPGA.

