

Интернет-журнал «Наукovedение» ISSN 2223-5167 <http://naukovedenie.ru/>

Том 7, №5 (2015) <http://naukovedenie.ru/index.php?p=vol7-5>

URL статьи: <http://naukovedenie.ru/PDF/84TVN515.pdf>

DOI: 10.15862/84TVN515 (<http://dx.doi.org/10.15862/84TVN515>)

УДК 621.3

Зенин Евгений Юрьевич

ОАО «Байкал Электроникс»

Россия, Московская область, Красногорский район¹

Инженер

Соискатель

E-mail: zeninhouse@gmail.com

Терентьев Юрий Игоревич

ОАО «Байкал Электроникс»

Россия, Московская область, Красногорский район

Ведущий инженер

E-mail: zeninhouse@gmail.com

Трихин Пётр Олегович

ЗАО «МЦСТ»

Россия, Москва

Инженер

E-mail: zeninhouse@gmail.com

Проектирование верхнего уровня в иерархическом маршруте

¹ 115191, Москва, а/я 67, Шахову Денису Андреевичу

Аннотация. В данной статье рассмотрены проблемы современного проектирования СБИС и предложено одно из направлений - иерархическое проектирование (иерархический маршрут), включающее в себя ряд методов и способов решения задач физического проектирования. Приведены достоинства и недостатки такого подхода, представлены результаты измерения времени прохождения маршрута на примере тестовой сверх большой интегральной схемы (далее СБИС).

При проектировании СБИС, включающей от нескольких десятков до сотен миллионов транзисторов, весьма остро встает вопрос оптимального планирования топологии. Требуется точно задать размеры чипа, задать месторасположение макроблоков и блоков ввода-вывода, организовать систему питания. Очень важно также выяснить, из каких логических частей (блоков) состоит СБИС и выделить на плане области для размещения этих устройств. Эти задачи крайне важно решать, учитывая потребности дальнейших этапов физического проектирования и верификации всей схемы. Не оптимальное планирование областей для размещения блоков приводит к тому, что на более поздних этапах проектирования могут возникнуть проблемы с трассировкой, нарушения по таймингу и другие нарушения, которые потребуют выполнить планирование заново.

Ключевые слова: СБИС (сверх большая интегральная схема); иерархический маршрут физического проектирования; САПР (система автоматизированного проектирования); RTL (register transfer level).

Ссылка для цитирования этой статьи:

Зенин Е.Ю., Терентьев Ю.И., Трихин П.О. Проектирование верхнего уровня в иерархическом маршруте // Интернет-журнал «НАУКОВЕДЕНИЕ» Том 7, №5 (2015) <http://naukovedenie.ru/PDF/84TVN515.pdf> (доступ свободный). Загл. с экрана. Яз. рус., англ. DOI: 10.15862/84TVN515

Современные средства САПР позволяют автоматически разместить стандартные ячейки (логику) блоков на всей доступной для этого площади сверхбольшой интегральной схемы (СБИС). Пользователь САПР может задать области, куда будут размещены только ячейки определённых блоков, и таким образом эффективно манипулировать структурой чипа в крупном масштабе, отдавая автомату право решения о конкретном месте размещения каждой конкретной ячейки. Однако есть фундаментальное свойство САПР, которое заключается в переборе вариантов при поиске решения. В силу очень большого числа степеней свободы, все варианты перебрать невозможно и поэтому качество (быстродействие, площадь, мощность) решения всегда будет хуже теоретически достижимого. Число степеней свободы растёт экспоненциально с ростом числа элементов, поэтому большой чип будет иметь более низкое качество, в процентах от своего теоретически достижимого, чем маленький чип — в процентах от своего. Чтобы преодолеть эту тенденцию, необходимо упростить задачу автомата, то есть уменьшить число элементов, которые он *одновременно* пытается разместить. Это можно сделать, применяя метод иерархического проектирования, поддерживаемый современными САПР. Части чипа синтезируются и размещаются отдельным запуском автомата. На уровне всего чипа также запускается автоматический синтез и размещение, однако, уже готовые блоки представлены в нём не в виде множества логических элементов, а в виде одного элемента, хотя и более сложного чем библиотечная ячейка. В качестве блока может выступать любое объединение ячеек, но для начала удобно воспользоваться имеющимся разбиением логического описания чипа на модули, и объединять в блок один или несколько модулей целиком.

Помимо повышения качества чипа в терминах быстродействия, площади и мощности, метод иерархического проектирования позволяет сильно сократить время внесения изменений в дизайн, возникающих вследствие изменений в логической модели, либо по другим причинам. Если изменение попадает в блок, синтезированный отдельно, то нет необходимости проводить пересинтез уровня всего чипа.

К недостаткам иерархического метода можно отнести сложность согласования межблочных связей, необходимость организации в каждом устройстве своей системы питания и согласования ее с аналогичными системами в соседних блоках и на уровне чипа. Также при автоматизации проектирования каждого устройства необходимо создавать свои командные скрипты, файлы правил проектирования, выделять отдельного специалиста для физического проектирования.

Метод или маршрут иерархического проектирования делится на несколько этапов: планирование иерархии, проектирование блоков, проектирование верхнего уровня.

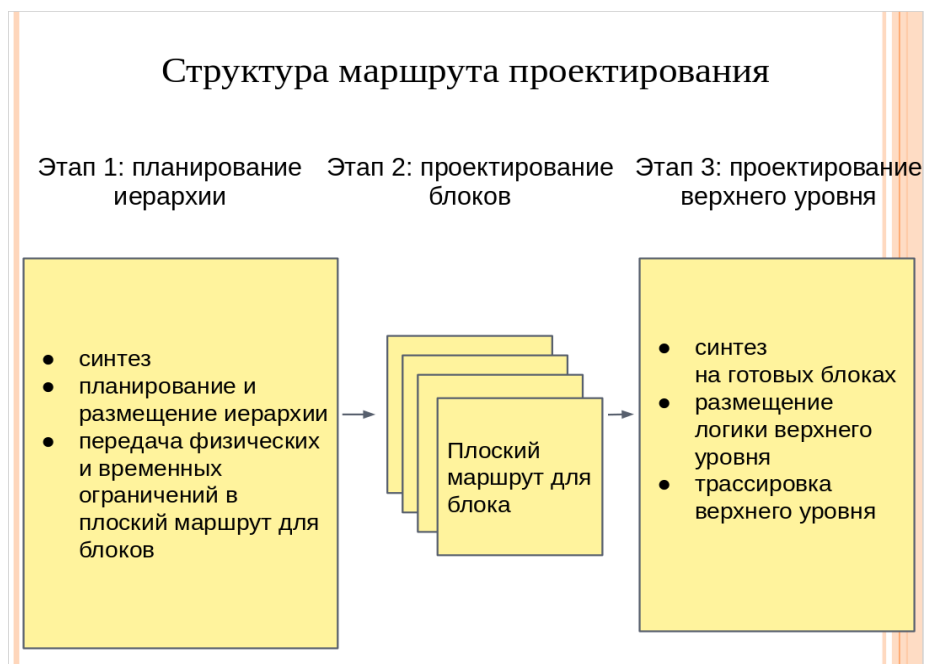


Рис. 1. Блок-схема иерархического маршрута

Этап планирования иерархии - один из самых важных в этом маршруте. Цель первого этапа - это обработка, анализ и изменение иерархии блоков верхнего уровня, разбиение верхнего уровня на блоки, размещение этих блоков друг относительно друга и бюджетирование блоков. Все это производится с целью улучшения временных характеристик, уменьшения площади или какой либо другой характеристики в дизайне. На этом этапе разработчик должен определить сколько и какие блоки будут в его дизайне. Забегая вперед, стоит сказать немного о разводке в данном маршруте, подразумевается, что после подстановки блоков на верхний уровень, разводка между ними будет проходить над блоками, а это означает что такие цепи нельзя будет буферизовать. Разумеется, это вызовет проблемы в слишком больших блоках, поэтому, основываясь на опыте полученном в создании этого маршрута, задается негласная рекомендация разработчикам планировать иерархию таким образом, чтобы размеры блока были меньше 200 мкм, что соответствует по порядку 10 000 ячеек в каждом блоке СБИС.

Результатом второго этапа маршрута является база данных для каждого из блоков на которые был разбит верхний уровень. Особенностью баз данных блоков является необходимость во временной/физической модели для успешной подстановки блока на верхний уровень. САПР позволяет использовать модель интерфейсной логики, то есть логика которая расположена между входным (или выходным) портом и первым регистром (или последним регистром в случае выходного порта). Так же к ней относится вся логика которая располагается между входным и выходным портом. Логика, расположенная между двумя регистрами, исключается из модели.

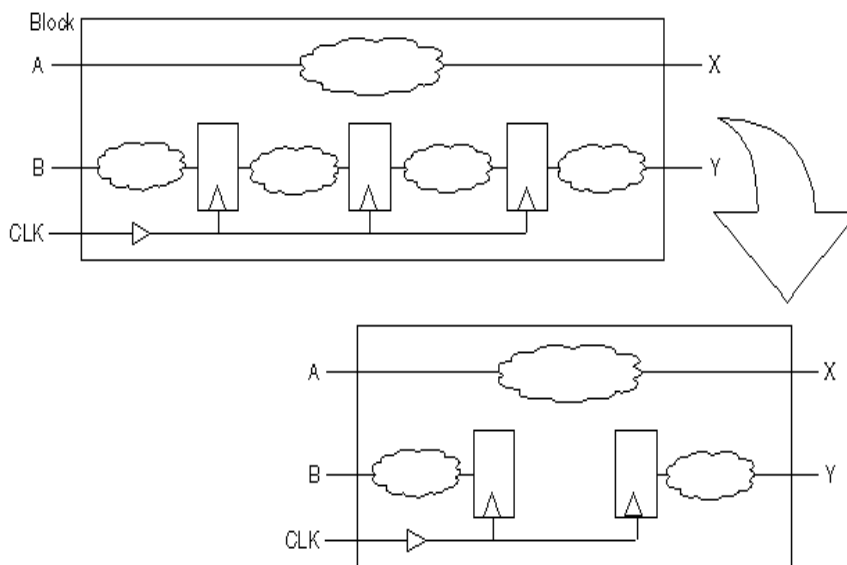


Рис. 2. Представление блока в виде интерфейсной модели

В некоторых моделях САПР при проведении оптимизаций верхнего уровня позволяет изменять интерфейсную логику внутри блоков, что приводит к улучшению временных характеристик и уменьшению критических путей.

На последнем этапе маршрута необходимо собрать блок, провести оптимизации верхнего уровня, межблочных связей, сделать разводку верхнего уровня. Третий этап делится на три основные части: синтез на готовых блоках, размещение логики верхнего уровня, трассировка верхнего уровня.

Синтез на готовых блоках (или иерархический синтез) - это процесс перевода логического RTL описания в эквивалентное описание на вентиляльном (транзисторном) уровне с использованием заданных библиотек, с учетом размещения крупных макроблоков, модулей памяти, элементов ввода/вывода и стандартных элементов с заданными ограничениями.

При решении САПР'ом задачи оптимального размещения используется *cost function* (или функция потерь). Минимизация данной функции позволяет уменьшить расхождение между результатом и требованием к тому или иному параметру задачи (временные характеристики, площадь, плотность разводки, время). Входными данными в задаче планирования являются координаты элементов размещения, вид элемента, его физические характеристики. Сложность и время решения задачи зависит от числа таких элементов.

В ходе иерархического синтеза число элементов верхнего уровня значительно меньше чем при плоском планировании, что позволяет САПР быстрее и оптимальнее разместить элементы на кристалле. Такой подход к синтезу позволяет разработчику тратить меньше времени и средств на работу со стандартными элементами и уделить больше внимания проектированию блоков.

На последних двух этапах маршрута решаются две проблемы: создание финального размещения блоков и логики на верхнем уровне и трассировка верхнего уровня. Отметим, что трассировка верхнего уровня является ключевой проблемой при проектировании, так как невозможность решения данной задачи может привести к принципиальному пересмотру всего проектирования в целом. На первых этапах маршрута (планирование иерархии) разработчик должен это учитывать и создавать размещение блоков таким образом, чтобы в дальнейшем задача трассировки верхнего уровня не стала краеугольным камнем.

При создании финального размещения необходимо учитывать количество слоев металла использованных в блоках. В некоторых случаях приходится уменьшать количество использованных слоев в блоках, тем самым, увеличивая плотность разводки на нижних металлах. Это делается из-за необходимости провести большое число связей над этим блоком и из-за большой плотности разводки в ограниченной области.

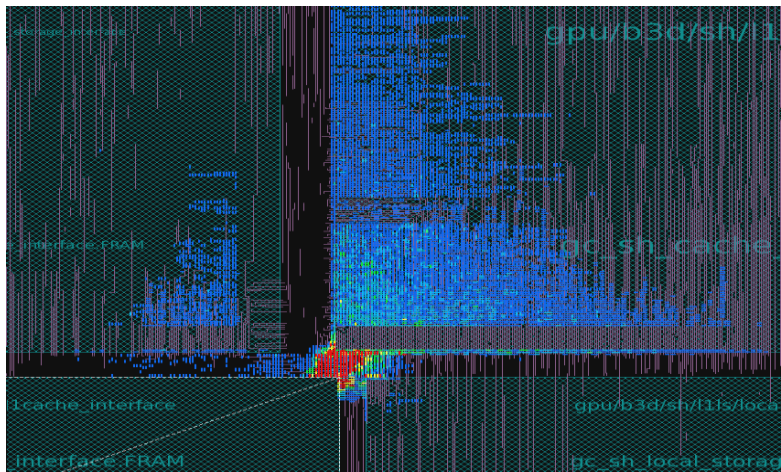


Рис. 3. Пример большой плотности разводки в ограниченной области

Заключительным этапом работы над кристаллом является этап верификации и проверки норм проектирования чипа. Использование иерархического подхода позволяет ускорить этот процесс, так как каждый блок в ходе маршрута проходит отдельную верификацию и проверку перед его использованием в маршруте. Это позволяет проводить верификацию и проверку верхнего уровня в ускоренном режиме, так как заметно снижается число нарушений.

Таким образом иерархический маршрут позволяет заметно уменьшить общее время проектирования. Ниже приведены графики времени прохождения иерархического и плоского маршрута на проекте с числом стандартных элементов порядка 4,6 млн. и площадью 17 кв. мм.

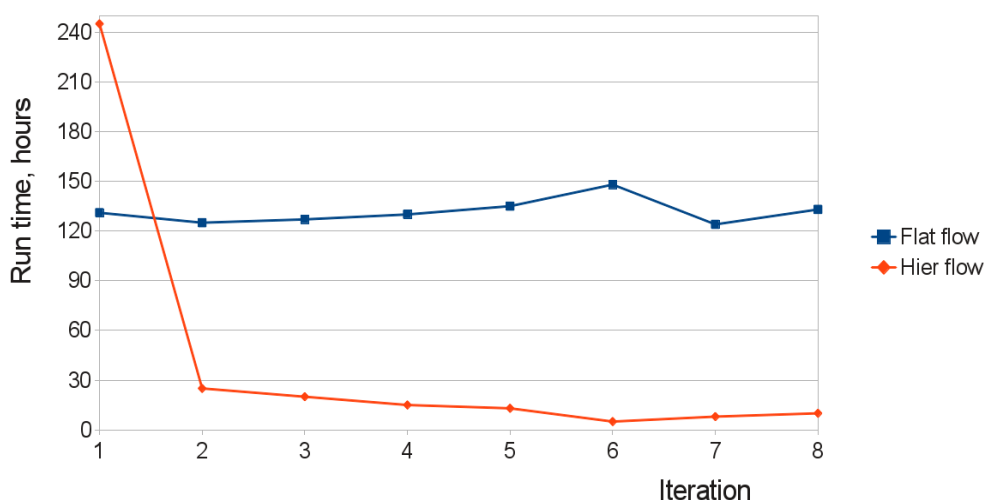


Рис. 4. График времени прохождения маршрута в зависимости от числа итераций (повторных запусков) для плоского и иерархического маршрута

На графике показана зависимость времени от начала проектирования кристалла до его верификации, с учетом внесения изменений на каждой итерации. Начало проектирования

занимает много времени при иерархическом подходе. Основная его часть тратится на планирование иерархии, так как от этого зависит дальнейшая сходимость проекта.

Из данного графика можно сделать вывод, что иерархический маршрут экономит время на больших проектах (~3 млн. стандартных элементов). Для небольших проектов такой подход является не оптимальным.

Правила проектирования микросхем в нанометровом диапазоне включают различные группы ограничений, для выполнения которых специалистами компании предпринимается ряд мер.

Самые жесткие ограничения определяются требованиями к форме и размерам элементов, их взаимному расположению и расстоянию между ними. Дополнительные ограничения на равномерность заполнения площади кристалла направлены на снижение влияния вариаций в параметрах элементов. При равномерном заполнении все физические структуры микросхемы находятся в примерно одинаковом окружении, и их взаимное влияние усредняется. Чтобы достичь этого, после трассировки все свободные места на кристалле были заполнены дополнительными полигонами металлов для достижения рекомендованных фабрикой значений плотности металлов по всему кристаллу.

Еще одна группа ограничений связана с так называемым «антенным эффектом». Во время технологических процессов плазменного травления и полировки на проводниках накапливается статический заряд, который может привести к пробое КМОП транзисторов или смещению порогов срабатывания транзисторов. Чтобы минимизировать «антенный эффект», были введены специальные ограничения на площадь соединений и внутренние слои металлизации.

Помимо перечисленных был использован еще ряд приемов для увеличения процента выхода годных СБИС, в частности, дублирование межслойных переходов везде, где есть свободное место и нет нарушений по временным ограничениям (setup/hold); использование более строгих (оптимальных для производства или рекомендованных) правил проектирования топологии в самых критичных устройствах.

ЛИТЕРАТУРА

1. «Logic Synthesis and Verification Algorithms» G.D. Hathel, F. Somenzi. 2000 г.
2. Конференция «Перспективы развития высокопроизводительных архитектур. История, современность и будущее отечественного компьютеростроения», 26 июня 2008 года, Институт точной механики и вычислительной техники им. С.А. Лебедева.
3. Ким А.К. Развитие архитектуры вычислительных комплексов серии «Эльбрус».
4. Немудров В., Мартин Г. «Системы-на-кристалле. Проектирование и развитие», Техносфера, 2004 г.
5. «Hierarchical Flow Support in Synopsys® Design Compiler Topographical Mode», Application Note Version A-2007.12-SP2, March 2008, Synopsys Inc.
6. «Top-Level Flow Using Block Abstractions and Transparent Interface Optimization (TIO)», June 2012, Synopsys Inc.
7. «Faster Top-Level Closure with Transparent Interface Optimization (TIO)», March 2012, Synopsys Inc.
8. «Hierarchical Design Planning Workshop», Feodor Merkelov, Synopsys implementation AC.
9. «Minimizing the time needed to complete a hierarchical design», Steve Kister, February 2011, Synopsys Inc.
10. «Hierarchical physical design - issues and methodologies», Steve Lloyd, July 2013, Synopsys Inc.

Рецензент: Статья рецензирована членами редколлегии журнала.

Zenin Evgeniy Yur'evich

JSC «Baikal Electronics»
Moscow region, Krasnogorsk district
E-mail: zeninhouse@gmail.com

Terent'ev Yuriy Igorevich

JSC «Baikal Electronics»
Moscow region, Krasnogorsk district
E-mail: zeninhouse@gmail.com

Trikhin Petr Olegovich

ZAO «MCST»
Russia, Moscow
E-mail: zeninhouse@gmail.com

Designing the Top Level in a Hierarchical Route

Abstract. This article discusses the issues concerning the modern VLSI design and suggests one of its trends — the hierarchical design (hierarchical route), which includes a number of methods and ways of solving the problems of physical design. The article presents the advantages and disadvantages of this approach as well as the results of measurements of the time of route passage as in the case of a test VLSI. When designing a VLSI that includes from several tens to hundreds of millions of transistors, the issue of optimal topology planning is thrown into sharp relief. It is required to specify exactly the size of a chip, to set the location of macroblocks and input-output blocks, and to arrange a power system. It is also important to find out which logical units (blocks) a VLSI consists of and to distinguish on the layout the areas for placing these devices. Solving these tasks is of crucial importance, taking into account the needs of further stages of physical design and verification of the whole circuit. The non-optimal planning of areas for placing blocks results in that in the later stages of designing, the problems with tracing, timing disorders, and other irregularities may occur, which requires replanning.

Keywords: VLSI; hierarchical route of physical design; CAD; RTL.

REFERENCES

1. «Logic Synthesis and Verification Algorithms» G.D. Hathel, F. Somenzi. 2000 g.
2. Konferentsiya «Perspektivy razvitiya vysokoproizvoditel'nykh arkhitektur. Istoriya, sovremennost' i budushchee otechestvennogo komp'yuterostroeniya», 26 iyunya 2008 goda, Institut tochnoy mekhaniki i vychislitel'noy tekhniki im. S.A. Lebedeva.
3. Kim A.K. Razvitie arkhitektury vychislitel'nykh kompleksov serii «El'brus».
4. Nemudrov V., Martin G. «Sistemy-na-kristalle. Proektirovanie i razvitie», Tekhnosfera, 2004 g.
5. «Hierarchical Flow Support in Synopsys® Design Compiler Topographical Mode», Application Note Version A-2007.12-SP2, March 2008, Synopsys Inc.
6. «Top-Level Flow Using Block Abstractions and Transparent Interface Optimization (TIO)», June 2012, Synopsys Inc.
7. «Faster Top-Level Closure with Transparent Interface Optimization (TIO)», March 2012, Synopsys Inc.
8. «Hierarchical Design Planning Workshop», Feodor Merkelov, Synopsys implementation AC.
9. «Minimizing the time needed to complete a hierarchical design», Steve Kister, February 2011, Synopsys Inc.
10. «Hierarchical physical design - issues and methodologies», Steve Lloyd, July 2013, Synopsys Inc.