

Д.В. Снигирев, В.В. Азаров, И.А. Есаков, Ю.Л. Погребной, С.А. Соин,
к.т.н. Г.В. Кристовский

Dmitry Sniguirev, Vitaly Azarov, Igor Esakov, Yuri Pogrebnoy, Sergey Soin,
Guntis Kristovskiy

ОПТИМИЗАЦИЯ МАРШРУТА ФИЗИЧЕСКОГО ПРОЕКТИРОВАНИЯ НА ПРИМЕРЕ РАЗРАБОТКИ L2-КЭША

PHYSICAL DESIGN FLOW OPTIMIZATION APPLIED TO L2 CACHE DESIGN

На примере L2-кэша рассмотрены возможности оптимизации стандартного маршрута физического проектирования крупных устройств микропроцессора. Основное внимание уделено анализу критических путей, использованию элементов иерархического проектирования, методике включения в проект полузаказных блоков.

Keywords: L2\$ physical implementation, design methodology, critical path grouping, semi-custom sub-designs.

Введение

В ЗАО «МЦСТ» физическое проектирование ведется на базе ASIC-методики с использованием САПР фирмы Synopsys. Исходным является описание проекта на уровне регистровых передач (RTL) на языке описания аппаратуры Верилог. Однако, рабочая частота, которая может быть получена с использованием этой методики, существенно меньше частот, достигаемых при заказном (custom) проектировании. Причины такого расхождения анализируются в [1]. Основные недостатки следующие:

1) Результаты проектирования практически непредсказуемы: реальные временные характеристики и рассеиваемая мощность могут быть получены только после завершения проекта.

2) При одноуровневом (flat) проектировании результат зависит, в основном, от возможностей САПР. При этом с увеличением сложности (количества вентилях) блока качество выполнения каждого этапа ухудшается, а время, необходимое для его выполнения, резко возрастает.

3) При проектировании не учитывается внутренняя структура блока.

4) Низкая воспроизводимость. Небольшие изменения исходного Верилог-описания могут потребовать полного повторения физического проектирования при отсутствии гарантий получения требуемого результата.

5) Необходимость достижения высокой рабочей частоты приводит к тому, что разработчик вынужден доводить проект вручную. Это сводит на нет основное преимущество ASIC-маршрута – скорость и еще более затрудняет воспроизводимость при повторном перепроектировании.

В статье исследуются пути усовершенствования ASIC-маршрута применительно к L2-кэшу с целью повышения частоты, уменьшения площади и получения более предсказуемых результатов. Основное внимание уделено следующим аспектам:

- учету внутренней структуры кэша при планировании площади (floorplaning); выбору конфигурации блоков памяти и оптимизации их размещения;
- разработке маршрута иерархического проектирования, включающего изменение иерархии RTL описания; настройке САПР под этот маршрут;
- созданию блоков нижнего уровня методами полузаказного проектирования; включению вновь созданных блоков в маршрут проектирования; оценке выигрыша по площади и времени.

1. Планирование площади L2-кэша

Началом этого этапа стало изучение и анализ RTL. Логическая структура L2-кэша содержит две части: арбитр и информационный блок – массив сохраняемых данных и

служебной информации. Для того чтобы одновременно обслуживать четыре запроса, информационный блок разделен на четыре подблока – l2_bank, имеющих идентичную схемотехнику. Арбитр принимает запросы от устройств процессора и через схему приоритета коммутирует их на подблоки. При невозможности немедленного обслуживания запросы буферизуются, и формируются очереди запросов к подблокам. Каждый из них содержит память данных, организованную в виде четырех столбцов, память тегов, схемы управления, а также буфер хранения модифицированных данных (STMB).

Помимо арбитра и информационного блока, в состав кэша входят устройство очереди и устройство обслуживания запросов к внешней памяти на чтение/запись – модуль l2_queues, а также устройство выравнивания выдаваемых из кэша данных согласно внешнему запросу – модуль l2_mx_algn. Все эти модули вместе с арбитром организуют работу по обслуживанию запросов к информационному блоку кэша в направлении процессора или внешней памяти. Такое взаимодействие модулей предопределяет их размещения на топологическом плане: l2_bank-модули располагаются по краям, а арбитр и остальные устройства – посередине.

Таким образом, для планирования площади имеем четыре крупных блока: три блока в середине и один l2_bank. Первоначальное расположение блоков на топологическом плане, использованное для оценок временных характеристик в проекте, представлено на рис. 1.

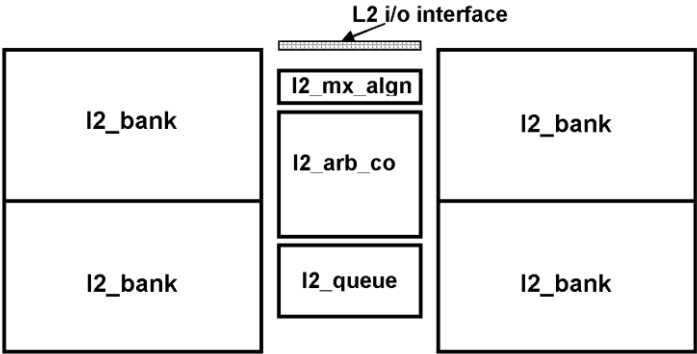


Рис.1

Предварительный топологический план L2-кэша

Было решено реализовать устройство l2_bank как законченный топологический модуль и использовать четыре его подстановки на верхнем уровне L2-кэша, а остальное оборудование разместить в средней части топологического плана с применением механизмов размещения в группах и блокирования областей размещения.

Площади банок памяти были получены с помощью промышленных компиляторов. Площадь логических блоков была оценена на основе синтеза RTL-кода L2-кэша. Сравнительный анализ площадей на основе синтеза и окончательных результатов проектирования представлен в табл. 1.

Таблица 1

Источник информации	Имя модуля	Количество элементов			Общая площадь, занимаемая элементами (мкм ²)			Общая площадь, занимаемая модулем (мкм ²)	Отношение к площади топологии (%)
		памяти	комб	триг	памяти	комб	триг		
Синтезированный нетлист	l2 - верхний уровень	1	2037	590	33820	21080	19896	74787	
	l2_arb_com	2	79834	19241	38120	855736	701896	1595753	
	l2_queues	0	21269	5096	0	258240	164597	422837	
	l2_mx_algn	0	8250	1033	0	123822	36852	160674	
	l2_bank	38	43990	9343	4207127	512342	277358	4996828	
	l2-сумм	155	287350	63332	16900450	3308250	2032676	22241376	-9
Измерения в топологии								24382628	

1.1. Выявление и анализ критических путей

Оценка критических путей является важнейшим инструментом выявления проблем, связанных с достижением требуемого быстродействия. Она проводится на всех этапах проектирования и, соответственно, подразделяется на следующие этапы:

- определение первоначальных критических путей на основе RTL описания;
- учет влияния топологического плана проекта;

- учет влияния реального размещения и трассировки.

На начальном этапе проводится создание структурной схемы потока данных, на которые приходится не менее 70% всего оборудования. По ней формируются группы критических путей, которые в дальнейшем используются для контроля и оптимизации временных характеристик. Логический синтез позволяет провести временной анализ по выделенным путям.

После создания топологического плана проводится дальнейшее уточнение существующих критических путей. Как правило, для длинных связей (более 1 мм) влияние топологии значительно, а проводная модель, используемая на этапе синтеза, недостаточно реалистична. Поэтому топологические поправки (Twd) для длинных связей рассчитывались по формуле:

$$Twd = (Lw / Ls) (Tbd + Trcd),$$

где: Lw – длина всей связи;

Ls – длина оптимального сегмента;

Tbd – задержка на оптимальном буфере;

$Trcd$ – задержка на оптимальном сегменте.

Для выявления оптимальных буфера и длины сегмента был проведен многовариантный анализ в HSpice. В результате, для используемой библиотеки оптимальным оказался буфер X20 с длиной сегмента 1300 мкм и суммарной задержкой 192 пс при длительности фронта 170 пс.

Таким образом, было определено 256 групп критических путей. В табл. 2 приведен фрагмент этого списка после синтеза и предварительного планирования топологии.

Наиболее чувствительными к топологии оказались критические цепи, проходящие через блоки памяти (топологическая поправка до 30% времени цикла). Они ограничивали рабочую частоту до 454 МГц, поэтому для достижения заданной рабочей частоты были приняты меры по оптимизации топологического плана.

Уточнение групп критических путей было продолжено на этапах размещения и трассировки. Созданные группы использовались для контроля временных характеристик и управления процессом оптимизации путем изменения приоритетов (весовых коэффициентов) отдельных групп.

Таблица 2

Группы критических путей	Кол-во тактов	Частота по синтезу (МГц)	Длина связи (мкм)	Временная задержка связи (пс)	Частота с учетом связей	Влияние связи на частоту (%)
all_ArbWco_StmbVmemRreadyReg	1,0	707	4090	576	502	-29
all_BnkcReg_ArbBuffer	1,0	531	4090	576	407	-23
all_BnkcReg_ArbDstMx	0,5	599	4090	576	445	-26
dw_RdMux_RF	0,5	723	2080	192	566	-22
dw_RdMux_TLU	0,5	972	2080	192	708	-27
dw_RdMux_internal	0,5	872			872	0
dw_dt_SRAM_REG0_p6	1,0	739	2970	384	576	-22

1.2. Оптимизация топологического плана проекта

Проведенный анализ списка критических путей позволил определить дальнейшие направления оптимизации топологии:

- выбор оптимальных размеров банок памяти для хранения массивов информации в блоках l2_bank (данных, тэгов адресов и служебной информации);
- оптимизация структуры данных при записи в блок l2_bank.

Выбор банок памяти и их размещение в блоке l2_bank

Массив банок памяти определяет размеры кэша и оказывает существенное влияние на временные характеристики. Анализ критических путей показал, что для реализации блока l2_bank подходят четыре варианта банок. Вариант V1 – базовый, заложенный при разработке RTL. В варианте V2 использована память с минимальным временем доступа. Вариант V3 использует компромиссную конфигурацию банки, с точки зрения быстродействия, площади, мощности. Вариант V4 выбран как наиболее экономичный по

потребляемой мощности (при чтении из кэша в варианте V4 активируется только половина банок памяти всего массива).

В результате топологического планирования блока, проведенного для всех четырех вариантов l2_bank, был определен оптимальный порядок размещения банок, выбрана их ориентация, а также выделено место для управляющих и буферных схем между рядами банок. Численная оценка критических путей потребовала предварительного физического проектирования, включающего этап оптимизации размещения. Результаты отражены в табл. 3.

Таблица 3

Вариант	Характеристики банки памяти			L2\$(1MB)			WNS для 560 МГц Тур (нс)
	Конфигурация	Число битовых колонок на один выход	Время чтения (нс)	Площадь	Мощность		
					Статическая	Динамическая	
V1	2k×19	8	1,4606	0,89	1,07	0,74	360
V2	2k×19	16	1,2504	1	1	1	286
V3	2k×38	8	1,4781	0,82	0,89	0,8	198
V4	1k×38	4	1,4518	0,87	1,14	0,58	323

Для наглядности площадь и мощность приведены в относительных единицах, где за «1» приняты соответствующие характеристики варианта V2. Несмотря на то, что банка варианта V2 обладает наибольшим быстродействием, в проекте применена банка варианта V3 (с наименьшей площадью), т.к. задержка всего критического пути при этом оказалась минимальной за счет оптимального топологического плана.

Оптимизация структуры данных при записи

Данные, хранимые в кэше, имеют следующую структуру:

- длина строки (cache-line) – 512 бит;
- за рабочий такт возможно обращение и обработка половины строки – 256 бит;
- половина строки состоит из четырех двойных машинных слов по 64 разряда

каждое;

- совместно с каждым словом в массиве данных хранится тэг данных и дополнительные биты ЕСС-кодировки, всего 76 бит.

Кроме того, в кэше применена схема резервирования, позволяющая заменить любую вышедшую из строя банку резервной.

Электрическая схема обработки данных в такте чтения приведена на рис 2.

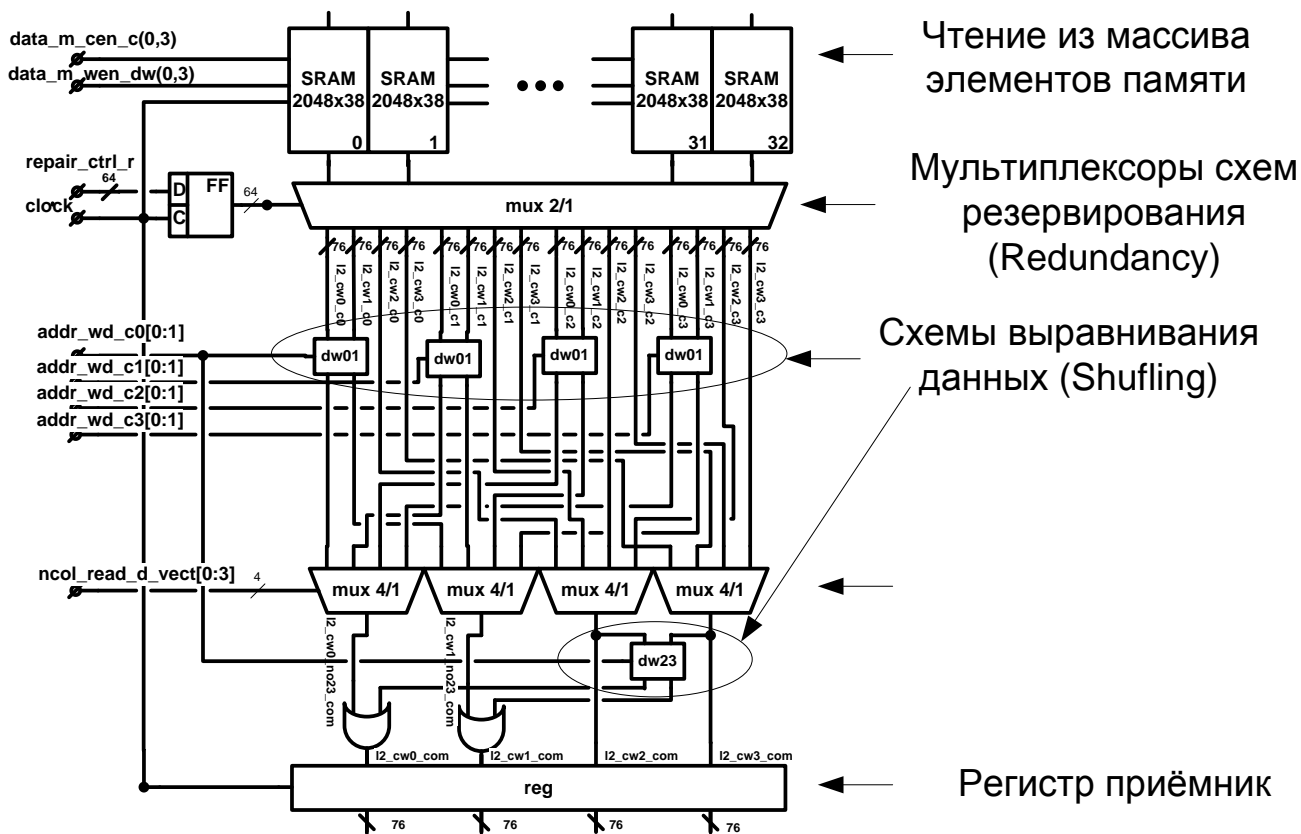


Рис. 2

Электрическая схема обработки данных в такте чтения из массива банок памяти

Поскольку L2 имеет четырехстолбцовую структуру, одновременно считываются четыре половины строки, всего 1216 бит информации.

В этом же такте чтения необходимо выполнить следующую обработку информации:

- мультиплексирование данных в схеме подключения резервной банки;
- выравнивание считанных данных (Shuffling);

- выбор столбца по результатам сравнения тэгов адресов.

Для минимизации критических связей были приняты меры по:

- обеспечению компактного размещения в банках памяти соответствующих разрядов всех четырех слов по всем четырем столбцам;
- реализации близкого размещения взаимозаменяемых банок.

Планирование размещения стандартных ячеек

Для улучшения воспроизводимости результатов при внесении изменений в проект и повторном проектировании были использованы следующие методы:

- ручное создание групп и областей размещения, рекомендуемых программе САПР;
- изменение иерархии RTL-описания для облегчения процесса разделения оборудования по группам размещения;
- применение блокировок размещения для повышения его эффективности (позволяет избавить программу размещения от необходимости равномерно заполнять области, удаленные от интерфейсов крупных модулей, таких как память).

Создание групп оборудования и размещение этих групп в заданных областях является первым шагом к управлению размещением, которое делает последующую оптимизацию более эффективной. Для того чтобы начальное размещение было удачным, необходимо понимание структуры потока данных проекта и знание его критических путей. Наилучшего результата можно добиться, осуществив стопроцентное предварительное размещение регистров потока данных.

В исходном RTL-модуле `l2_bnk_dmem` блока `l2_bank` описание схемотехники, ответственной за обработку данных, было представлено в виде иерархических модулей, обрабатывающих 76 бит информации. Это удобное, с функциональной точки зрения, описание, но не оптимальное для топологического планирования. Поэтому в RTL-коде модули, ответственные за обработку 76-разрядных данных, были заменены модулями

разрядностью 19 бит. Это дало возможность формировать группы размещения, используя новую систему иерархических имен и размещать комбинационные части схемы, отвечающие за обработку определенных разрядов данных, в областях, ближайших к банкам, в которых они записаны.

В результате размещения и оптимизации блока l2_bank путь от банок памяти данных до регистра приемника удалось сократить на 300–500 пс. Это позволило повысить рабочую частоту до 527 МГц, что на 17% лучше по сравнению с предыдущим результатом.

2. Маршрут иерархического проектирования

При проектировании современных микропроцессоров на технологии 90 нм и ниже основной вклад в задержки вносят линии связи [2], длина которых определяется размещением стандартных ячеек. Применяя одноуровневый (flat) подход, добиться оптимального размещения стандартных ячеек для устройств, содержащих сотни тысяч вентилях, практически невозможно; кроме того, резко возрастает время работы программ САПР – одна итерация может длиться несколько суток. Решением проблемы является иерархическое проектирование [3].

Часто физическая иерархия проекта, создание которой связано с планированием площади кристалла [4], отличается от логической. Ее создание оправдано, т.к. позволяет оптимизировать размещение, управляя длиной связей. В нашем маршруте проектирования для целей синтеза, размещения и трассировки использовались Design Compiler, Physical Compiler и Astro, соответственно (программы компании Synopsys).

На рис. 3 приведена схема основных стадий проектирования L2-кэша. Были введены следующие уровни иерархии: верхний – L2-кэш, включающий в себя блоки второго уровня l2_bank, блоки выравнивания данных, блоки очередей и арбитр, в котором были выделены блоки, созданные методами полузаказного проектирования.

Отдельные стадии проектирования проводились с 10% запасом по длительности периода синхронизации ($T_{сус}$) по отношению к последующим стадиям. Переход к ним осуществлялся лишь тогда, когда превышение задержек составляло не более 10% от длительности периода синхронизации, принятой на данной стадии ($Slack < 10\%$ от $T_{сус}$).

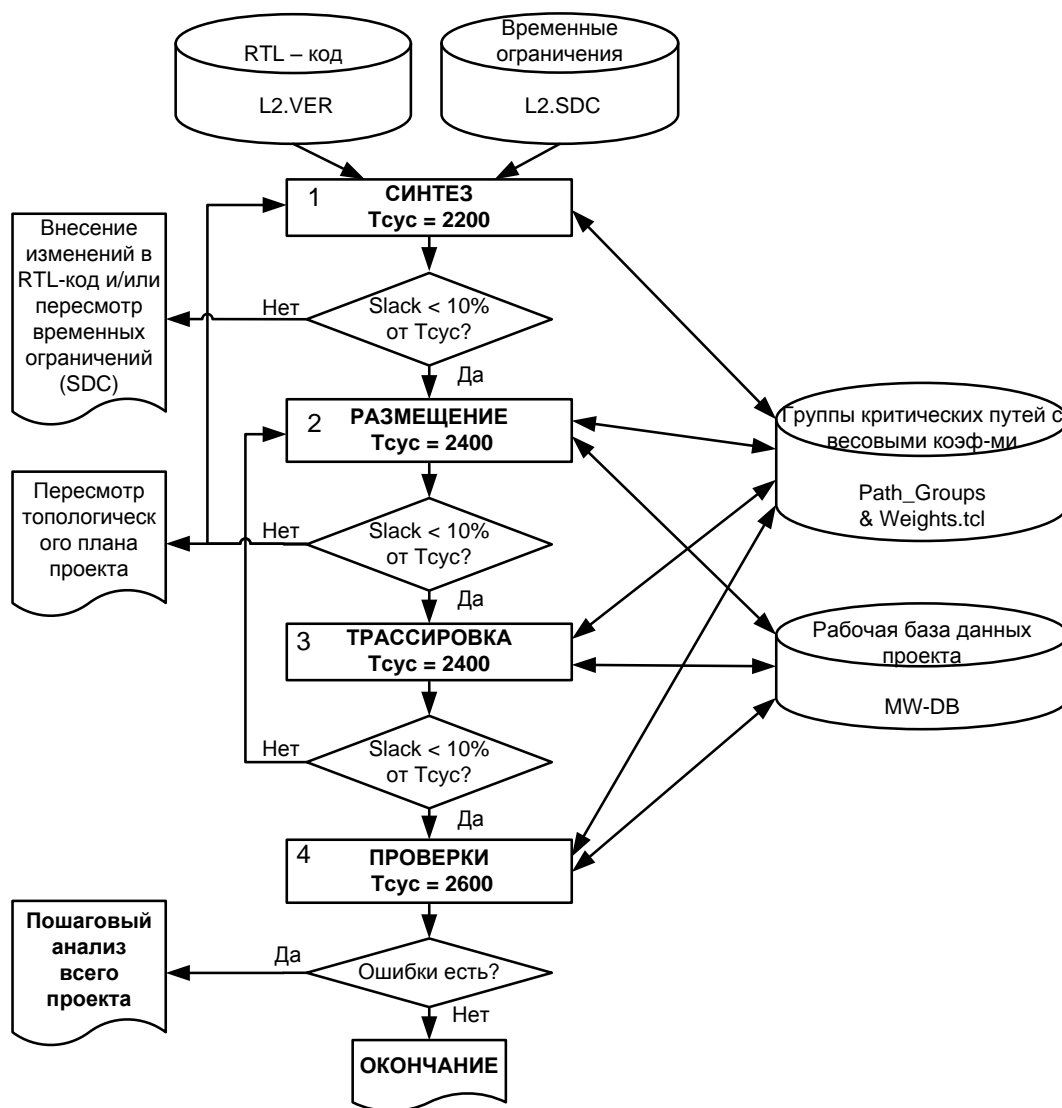


Рис. 3

Маршрут проектирования

Для постоянного контроля временных характеристик и управления процессом оптимизации при синтезе использовались группы критических путей, заданные в отдельном файле (Path_Groups&Weights.tcl). В последующих стадиях использовалась

единая база данных проекта, содержащая всю логическую, топологическую и служебную информацию (Design DataBase).

Маршрут включал следующие стадии:

- **Синтез** верхнего уровня L2-кэша и входящих в него модулей (этап 1). Он проводился при $T_{сус} = 2200$ пс по методике итерационного, иерархического синтеза. При недостижимости поставленной задачи ($Slack > 10\% T_{сус}$) вносились изменения в RTL или физический проект верхнего уровня (SDC файл). Опыт показал, что хороший результат достигается в результате трех итераций. Особенностью маршрута является использование механизма бюджетирования – процедуры генерации временных ограничений для синтеза всех входящих модулей (SDC).
- **Размещение**, оптимизация размещения и синтез дерева синхронизации (этап 2, $T_{сус} = 2400$ пс). При $Slack > 10\% T_{сус}$ возможно внесение в топологический план дополнительных уровней иерархии, а также возможен возврат к стадии синтеза.
- **Разводка** и оптимизация сигнальных проводников и дерева синхронизации (этап 3, $T_{сус} = 2400$ пс). При $Slack > 10\% T_{сус}$ возможен возврат к стадии размещения. В данном проекте при разводке используются все слои металлизации с 1-го по 7-й. При проектировании предпочтение отдается верхним слоям.
- **Проверки** (этап 4, $T_{сус} = 2600$ пс). Проверки топологии дизайна и анализ временных характеристик.

По результатам этапа 4 для устранения выявленных недочетов применяется ручное редактирование схемотехники и/или топологии.

3. Внедрение элементов полуказного проектирования в разработанный маршрут

Первые итерации проекта выявили целый ряд критических путей, временные характеристики которых существенно ухудшались на этапе размещения по сравнению с результатами, полученными на этапах синтеза и предварительной оценки критических

путей. Анализ устройств, через которые проходят указанные пути, показал, что требуемые параметры могут быть получены путем уменьшения длины внутренних связей и площади, а также сокращения расходов на буферизацию длинных связей.

Для этого было решено перейти к полузаказному проектированию таких блоков с учетом их внутренней структуры, т.е. к проектированию на базе стандартных ячеек с их ручным размещением. Наибольший выигрыш в таком случае достигается для устройств с регулярной структурой.

Были разработаны два маршрута: упрощенный, использующий механизм относительного размещения, поддерживаемый программой Physical Compiler фирмы Synopsys, и более полный, включающий в себя ручное размещение и трассировку устройства при помощи программы Custom Designer.

Начала обоих маршрутов совпадают (рис. 4). На этапе 1 анализируется структура устройства и оптимизируется его электрическая схема. При этом часто возникает необходимость изменения иерархии исходного RTL-описания (этап 2). Затем следует убедиться, что измененный RTL функционально эквивалентен исходному (этап 3). На этапе 4 проверяется соответствие модифицированной электрической схемы RTL-описанию (формальная верификация). Это важный этап, т.к. процесс оптимизации нацелен на упрощение электрической схемы.

Далее маршруты расходятся. В упрощенном варианте модифицированная электрическая схема служит основой для создания командного файла относительного размещения стандартных ячеек (этап 5). Полный маршрут включает в себя ручное размещение и трассировку (этап 6). Параллельно с разработкой топологии проверяются временные характеристики (этапы 7.1, 7.2). Результаты проектирования и временные модели устройства передаются в программу Astro для использования при проектировании блоков верхнего уровня.

Оба маршрута имеют свои преимущества и недостатки. Первый маршрут позволяет

решать поставленные задачи при меньших усилиях на проектирование отдельных блоков. Однако само применение блоков не только не уменьшает нагрузки на основные средства проектирования, но и приводит к усложнению маршрута из-за комбинирования разных программных средств на этапе создания топологии верхнего уровня.

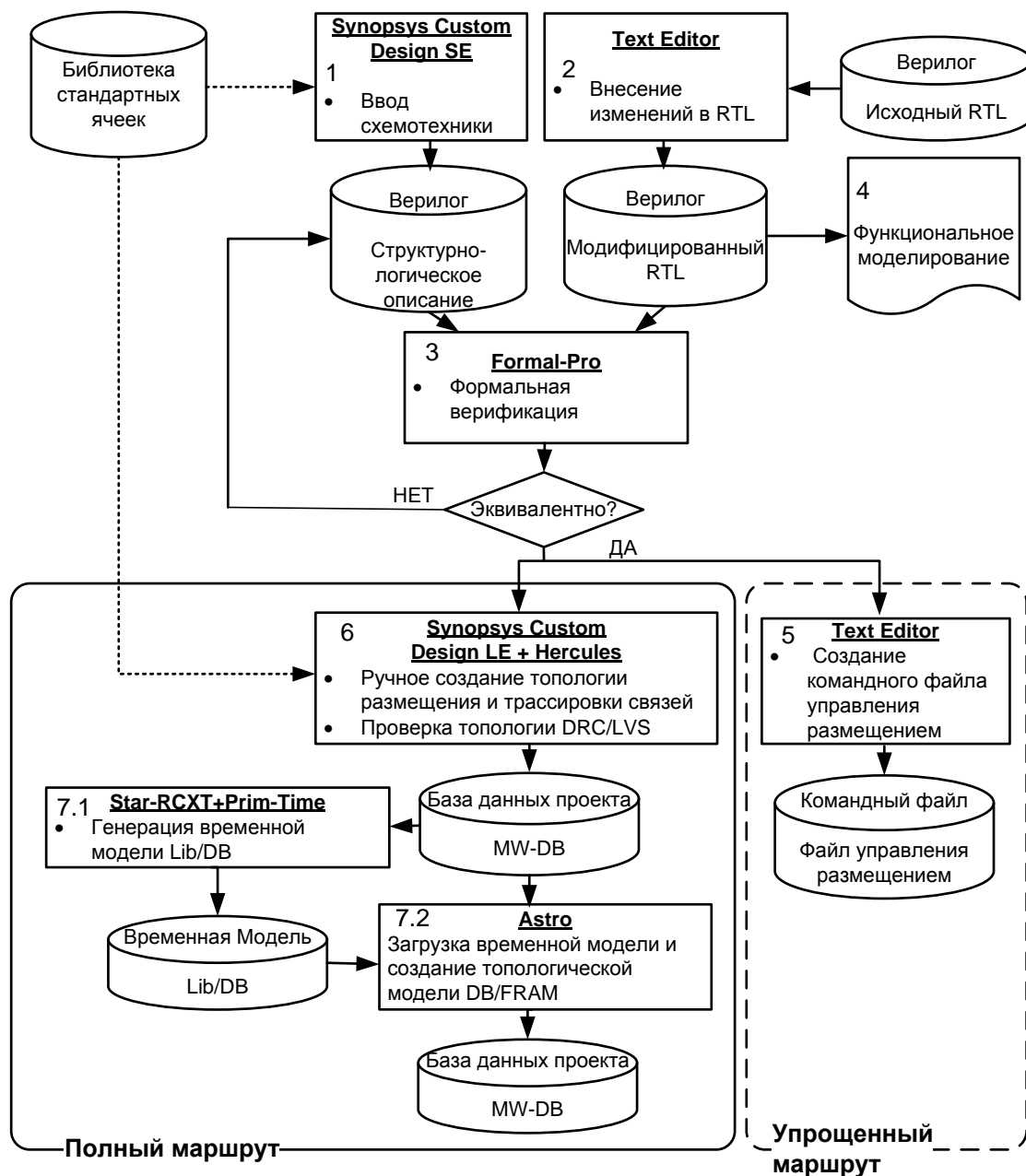
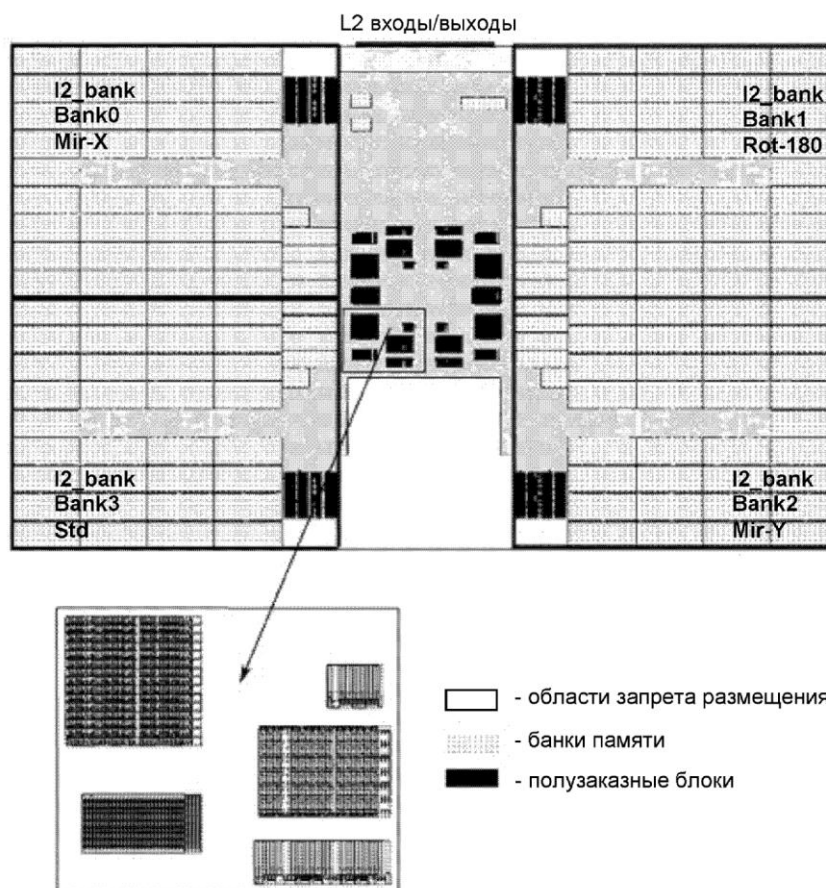


Рис. 4

Маршрут полуавтоматического проектирования блоков

Второй маршрут более трудоемкий, но позволяет приблизительно на 10–15% улучшить временные характеристики блоков, а при выбранной конструкции сетки земли и питания – добиться 10% уменьшения их площади. В описываемом проекте первый маршрут был использован для разработки модуля STMB блока I2_bank, второй – проверен и отработан на устройствах очередей арбитра. Это позволило уменьшить площадь всего



L2-кэша и улучшить его временные характеристики. Всего было разработано 26 блоков, в которых разместилось 30% комбинационных и 50% триггерных ячеек L2-кэша. На рис. 5 приведен окончательный вариант топологического плана L2 кэша.

Рис. 5

Рабочий вариант топологического плана L2-кэша

Заключение

Описанные в работе методы позволили улучшить сходимость процесса

проектирования, достичь требуемых характеристик L2-кэша и уменьшить его площадь на 20% по сравнению с предварительными оценками.

Базируясь на опыте этого проекта, соответствующего новым требованиям отечественной практики, можно отметить:

1. Важнейшими шагами физического проектирования крупных устройств являются планирование площади и анализ критических путей.

2. Для высокочастотных систем на кристалле длина критических путей служит критерием качества выполнения всех этапов проектирования.

3. Наиболее сложной и трудоемкой задачей физического проектирования является создание оптимального размещения. Качественное размещение требует разбиения крупных блоков на более мелкие, т.е. приводит к иерархическому проектированию.

4. Переход к иерархическому проектированию требует более глубокой интеграции физического и логического проектирования.

Литература

1. David Chinnery, Kurt Keutzer «Closing the gap between ASIC and Custom», Cluver Academic Publishers, 2003.

2. Jason Cong «An interconnect centric design flow for nanometer technologies», Proceedings of IEEE, vol. 89, N 4, April 2001, pp. 505-528.

3. Vjay Gullapalli, Kaijani Shi «Hierarchical design techniques», Synopsys white paper, 2004.

4. S. N. Adya, S. Chaturvedi, J.A. Roy, I.L. Markov «Unification of partitioning, floorplaning, and placement», Proceedings of ICCAD, November 2004, pp. 550-557.

5. B.M. Beckmann and D.A. Wood «Managing wire delay in large chip multiprocessor caches», Proceedings of MICRO-37, 2004, pp. 316-330.