

ВИРТУАЛЬНОЕ МАКЕТИРОВАНИЕ ПЛИС

Общие сведения о виртуальном макетировании

Перед тем как перейти к рассмотрению принципов виртуального макетирования ПЛИС, имеет смысл вспомнить, как развивалась концепция *виртуальных макетов (прототипов) кристалла (SVP — silicon virtual prototype)* в среде заказных микросхем. Подобный экскурс позволит проследить миграцию некоторых проявлений виртуального макетирования из мира заказных микросхем в мир ПЛИС, а также выявить проблемы, связанные с этими проявлениями.

Среди заказных микросхем (ASIC) существуют высокотехнологичные устройства, содержащие десятки миллионов вентиляей. Размеры и сложность систем на подобных мегаустройствах фактически привели к тому, что средства проектирования в определённом смысле перестали соответствовать требуемому уровню.

Проблема заключалась в том, что при традиционных методах проектирования многие ошибки невозможно было выявить до проведения временного анализа, основанного на реальных физических значениях ёмкостей, сопротивлений и иногда индуктивностей, которые были получены на этапе размещения и разводки элементов. Для проведения подобного анализа разработчикам приходилось преодолевать весь путь проектирования от начала до конца, включая синтез, размещение и разводку элементов, и лишь только потом появлялась возможность выявить ошибки. Очевидно, что процедуру выявления и локализации ошибок желательно производить на ранних этапах проектирования.

Описанная ситуация зачастую приводила к многочисленным итерациям процесса проектирования и существенно задерживала выход систем на базе заказных микросхем на рынок. А как показала практика, как правило, на рынке место есть только победителю, второе место отсутствует!

Одно из решений описанной проблемы заключается в разработке виртуальных прототипов, которые являются макетами устройств, могут относительно быстро генерироваться, и, будем надеяться, содержат достаточно информации, чтобы позволить разработчикам идентифицировать и решать большую часть потенциальных проблем, не прибегая к трудоёмким этапам проектирования. Теоретически, чтобы повторить устройство обычным способом, нужно несколько дней или недель, а если использовать виртуальный прототип — несколько часов.

Применение виртуальных прототипов при проектировании заказных микросхем

В предыдущей главе мы рассмотрели ряд положений, согласно которым роль логического синтеза заключается в обработке описания устройства на уровне регистровых передач RTL вместе с набором вре-

менных ограничений. Средства логического синтеза автоматически конвертируют RTL-описание устройства в совокупность регистров и булевых (логических) выражений, одновременно выполняя различные процедуры минимизации и оптимизации, включая оптимизацию по площади, занимаемой на кристалле, и по быстродействию. После этого они генерируют таблицу соединений вентилях, которая, будем надеяться, удовлетворяет заданным временным ограничениям.

Большинство обычных приложений логического синтеза оперируют размерами вентиля, а не величиной задержки на нем. Другими словами, эти приложения постоянно ищут компромисс между размерами вентиля и связанной с ними задержкой. В соответствии со своим принципом действия эти приложения производят огромное количество трудоемких и ресурсоемких вычислений. Кроме того, некоторые оптимизационные решения, сформированные средствами логического синтеза, приводят к бессмыслице, когда дело доходит до физической реализации устройства, т. е. до размещения и разводки элементов.

Виртуальные прототипы на уровне вентилях. Быстрый и грубый синтез

Ключевая особенность *виртуальных прототипов* заключается в том, что они могут создаваться достаточно быстро и просто. Большинство современных виртуальных прототипов заказных микросхем основаны на использовании таблицы соединений вентилях, которая впоследствии разводится с помощью временного алгоритма. К сожалению, применяемые здесь средства традиционного синтеза потребляют слишком большое количество времени и вычислительных ресурсов в связи с необходимостью соответствия заданным временным ограничениям. Поэтому для создания некоторых виртуальных прототипов заказных микросхем используют так называемый метод *быстрого и грубого синтеза* (Рис. 10.1).

1887 г. Англия. Дж. Томсон открыл электрон.

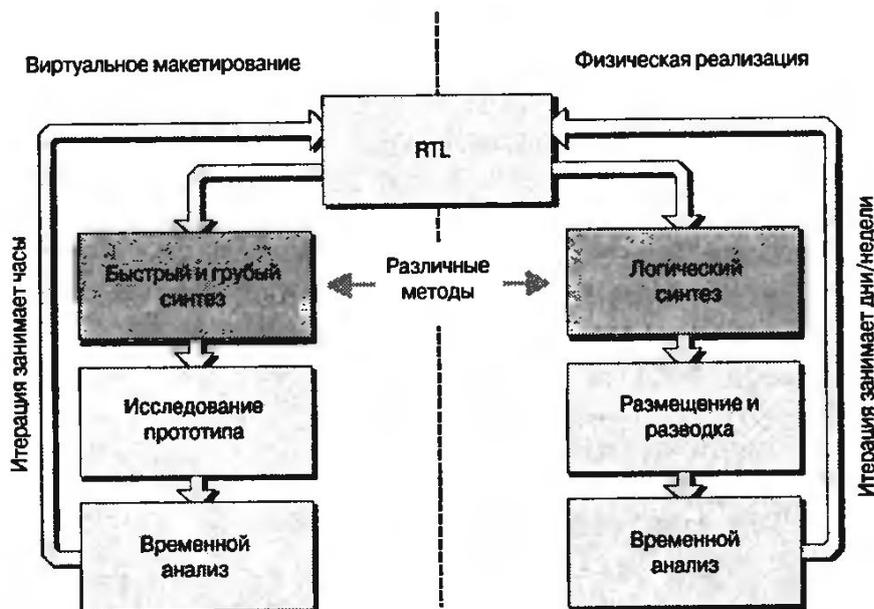


Рис. 10.1. Виртуальное макетирование, на основе алгоритма быстрого и грубого синтеза

Метод быстрого и грубого синтеза обычно основывается на алгоритмах, совершенно отличных от тех, которые применяются в процессе создания реального экземпляра устройства. Это может быть, например, алгоритм прямого отображения RTL-кода. Следовательно, гене-

1887 г. Англия. Вильям Крукс (William Crookes) продемонстрировал прямолинейное распространение катодных лучей.

1887 г. Германия. Генрих Герц (Heinrich Hertz) продемонстрировал эффекты передачи, приёма и отражения радиоволн.

рируемая таблица соединений вентилях, используемая для формирования описания уровня регистровых передач, может отличаться от точного образа окончательного варианта устройства.

Это значит, что, используя виртуальный прототип для выверки RTL-описания и временного анализа, разработчики по-прежнему применяют совершенно другие методы для создания реальной таблицы соединений, которая затем поступает на этап физической реализации устройства, т. е. передается средствам размещения и разводки.

Другими словами, проблема, и довольно сложная, реализации рассматриваемого подхода применения виртуального прототипа состоит в том что средства макетирования и их методики никак не связаны и существенно отличаются от средств практической, реальной реализации. Это приводит к несоответствию прототипа и реального устройства, то есть к плохой корреляции между ними, что может стать причиной длительных по времени итераций, тем самым, сводя на нет все преимущества использования виртуальных прототипов.

Виртуальные прототипы на уровне вентилях. Синтез, оптимизированный по быстродействию

В отличие от обычного логического синтеза, который основывается на поиске компромисса между размерами и задержкой вентилях, в основе так называемого *синтеза, оптимизированного по быстродействию*¹⁾ (gain-based synthesis), лежит концепция *логической работы*²⁾.

В этом алгоритме концепция *логической работы* используется для установки фиксированных временных параметров, которые средства физической реализации устройства будут использовать для размещения и разводки элементов. Это значит, что по окончании этапа синтеза будет завершена временная оптимизация, а также определены и зафиксированы все задержки на участках схемы. На этапе размещения элементов используется алгоритм изменений размера, который, подстраиваясь под жестко определённые временные параметры, производит изменение размера логической ячейки. Затем очередь за алгоритмом разводки, который для поддержки начальных временных ограничений и сохранения целостности сигнала производит подстройку ширины проводников и расстояний между ними.

Этот метод требует существенно меньшего количества памяти и вычислительных ресурсов, чем обычный синтез. Другими словами, этот метод синтеза, оптимизированного по быстродействию (gain-based synthesis), по своим возможностям превосходит обычные алгоритмы.

Интересно, что методы синтеза, оптимизированного по быстродействию (gain-based synthesis), используют все имеющиеся резервы по задержке сигнала. Это значит, что каждый вентиль будет иметь минимально допустимый размер, удовлетворяющий временным ограничениям. Вследствие этого итоговая реализация устройства будет зани-

¹⁾ Во время написания этой книги главным сторонником синтеза, оптимизированного по расширению, выступала компания Magma Design Automation (www.magma-da.com).

²⁾ Концепция *логической работы* была разработана в 1999 году и подробно рассмотрена в книге «Логическая работа: разработка быстрых КМОП-схем» («Logical Effort: Designing Fast CMOS Circuits», авторами которой были Иван Сюзерланд (Ivan Sutherland), Боб Спрул (Bob Sproull) и Дэвид Харрис (David Harris).

Иван Сюзерланд приобрёл мировую известность благодаря исследованиям в области логического проектирования.

мать наименьшую площадь на поверхности кристалла, что позволит существенно снизить перегрузки устройства, мощность потребления и уровень шумов.

«Но позвольте, — скажет читатель, — причем здесь виртуальные прототипы?» Да притом, что скорость и производительность, присущие такому синтезу, позволяют одно и то же программное ядро использовать как для макетирования, так и для физической реализации (Рис. 10.2).

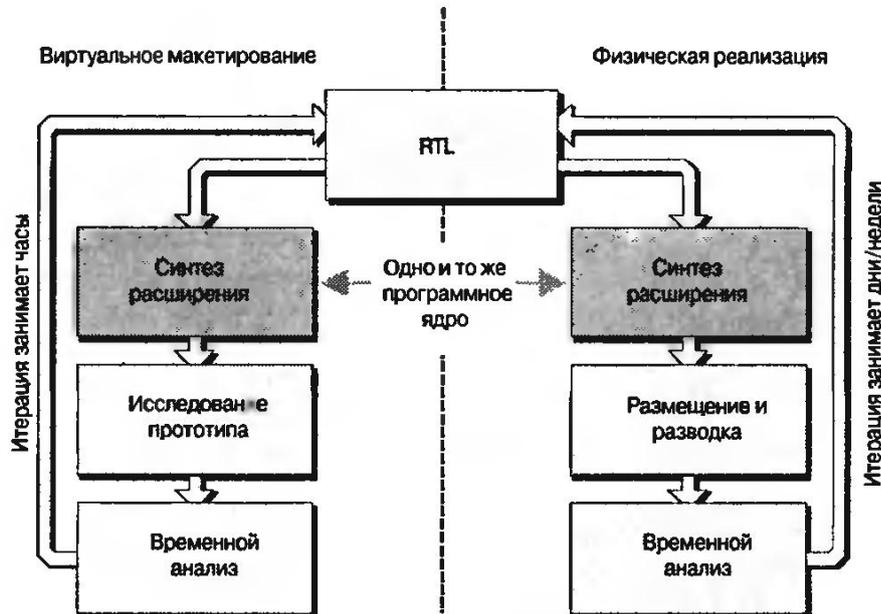


Рис. 10.2. Виртуальное макетирование, на основе синтеза оптимизированного по быстродействию

Использование одинаковых алгоритмов, средств и технологий, как при макетировании, так и при физической реализации, позволяет достичь высокой корреляции между прототипом и реальным устройством, а также значительно уменьшить время на повторные итерации в процессе разработки.

Виртуальные прототипы на уровне кластеров

Как обсуждалось ранее, большинство современных виртуальных прототипов основаны на таблицах соединений вентилях. Даже если эти таблицы генерируются путем быстрого-и-грубого синтеза, они по-прежнему могут содержать миллионы вентилях. В свою очередь, огромное количество вентилях существенно замедляет процесс размещения и разводки виртуального прототипа.

Одно из решений этой проблемы заключается в использовании принципа *кластеризации* в качестве основы для размещения в виртуальном прототипе и оценке задержки сигналов. В этом случае ячейки, т. е. вентилях и регистры, сгенерированные путем быстрого-и-грубого синтеза или синтеза, оптимизированного по быстродействию, автоматически собираются в группы, называемые *кластерами*. Каждый кластер может состоять из нескольких десятков или нескольких сотен ячеек, т. е. имеет достаточно малые размеры, чтобы сохранить качество размещения компонентов на поверхности кристалла, но в то же время количество кластеров существенно меньше количества ячеек, что весьма существенно сказывается на времени выполнения алгоритма размещения.

В действительности, от кластера к кластеру число ячеек может меняться, при этом площадь, занимаемая кластером, поддерживается, по

1888 г. Америка.
Начал работу первый общественный таксофон.

1889 г. Америка.
Алмон Строуджер
(Almon Brown
Strowger) изобрёл
первую шаговую
АТС.

возможности, на одном уровне. Чтобы сделать вычисления не такими сложными, а требования по производительности не столь жесткими, все процедуры оптимизации и анализа выполняются применительно к кластерам, а не к вентилям. Кроме того, если два кластера связаны между собой большим количеством проводников, как и бывает в большинстве случаев, для предварительной оценки разводки, влияющей на итоговое расположение кластеров, эта совокупность проводников рассматривается как один «утяжелённый» проводник.

Виртуальные прототипы на основе RTL

В инженерной практике существует эмпирическое правило, которое гласит: обнаружение и исправление проблем на любой из стадий проектирования, реализации и внедрения обходится в 10 раз дороже, чем обнаружение и устранение тех же проблем на предыдущем этапе. В области проектирования устройств на цифровых микросхемах существует три главные контрольные точки, в которых можно провести анализ занимаемой площади на кристалле, временных параметров и так далее (Рис. 10.3).

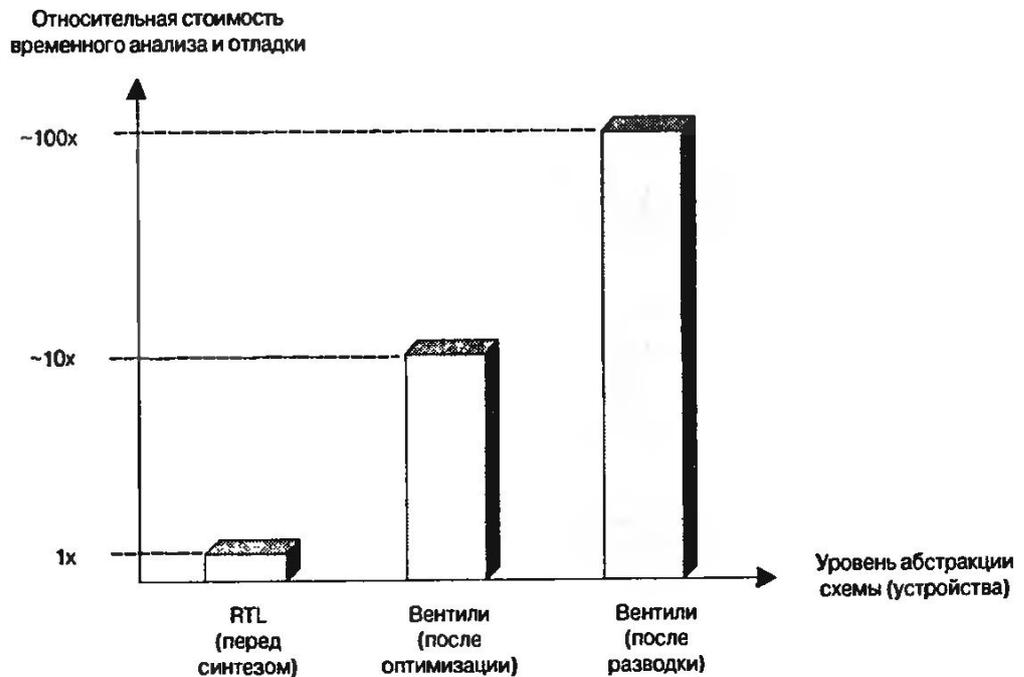


Рис. 10.3. Контрольные точки для проведения анализа занимаемой площади, временных и других параметров

При анализе схемы или архитектуры устройства, процесс обнаружения проблемных, с точки зрения временных параметров, участков цепи и внесения в них исправлений называется поиском *временного соответствия (timing closure)*. Независимо от этапа применения установление временного соответствия является итеративным процессом, т. е. для устранения ошибок в устройстве необходимо несколько раз провести операции в цикле анализ-обнаружение-коррекция.

Что касается уровней абстракции, показанных на Рис. 10.3, несомненно, источником самых точных данных будет временной анализ, выполняемый после разводки элементов. Однако результаты этого анализа будут чрезвычайно дорогими, и чтобы их получить, потребуются много времени. Процесс итерации на этом этапе, как правило, является тяжелой процедурой, поэтому разработчик всеми правдами и неправдами старается избежать внесения изменений в устройство на этом этапе разработки.

При обычном подходе к проектированию устройств первая контрольная точка, пригодная для проведения относительно точного временного анализа, располагается после синтеза вентилях и *локальной оптимизации*¹⁾.

Проблема заключается в том, что при типовой последовательности проектирования на этом этапе потребуется использование алгоритма физического синтеза для создания таблицы соединений вентилях. Поэтому этот подход является чрезвычайно сложным и длительным, и в случае больших блоков может потребоваться несколько дней на прохождение полного цикла синтеза и временного анализа. Такой подход не только затягивает процессы разработки и достижение состояния временного соответствия, но и требует использования дорогих средств разработки, которые должны использоваться в большей степени для реализации кристалла, чем для временного анализа.

Возможным решением является использование виртуального прототипа, но при этом следует иметь в виду, что данному решению присущи собственные недостатки, включая требование использовать некоторые виды сложного и достаточно продолжительного синтеза и таких же средств размещения элементов.

Еще один подход заключается в использовании виртуального прототипа на уровне регистровых передач²⁾. Это позволит инженерам быстро обнаруживать и исправлять участки схемы, которые могут оказаться потенциальными причинами проблем. Чтобы понять принцип этого подхода, сначала необходимо отметить, что существуют родственные приложения, которые из *файлов логических (LEF) и физических (DEF) определений*, связанных с библиотекой логических элементов ASIC, генерируют соответствующую базу данных наборов конструктивных компонентов, которая затем используется в виртуальном прототипе на уровне регистровых передач (RTL), см. Рис. 10.4.

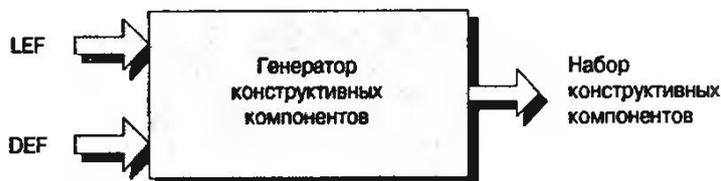


Рис. 10.4. Создание набора конструктивных компонентов

Важно, что такой набор конструктивных компонентов не является библиотекой типовых логических элементов, но служит набором типовых логических функций (таких как: счетчики, элементы ИСКЛЮЧАЮЩЕЕ ИЛИ и так далее). В ходе создания набора компонентов генератор описывает режим работы этих функций, в том числе временные параметры и параметры расположения на поверхности кристалла.

Генератор виртуального прототипа уровня регистровых передач и подсистема анализа обрабатывают RTL-код устройства, заданные устройству временные ограничения (в промышленном стандарте SDF) и набор конструктивных компонентов, связанный с целевой библиоте-

Файл LEF (logical exchange format — формат логического обмена) детализирует логическую функциональность ячеек в библиотеке. DEF-файл (design exchange format — формат конструктивного обмена) детализирует физические аспекты ячеек, например значения их сопротивлений и ёмкостей, а также физические размеры.

¹⁾ *Локальная оптимизация* подразумевает, что после работы алгоритма размещения во время первого прохода можно произвести некоторые «настройки» (выполнить оптимизацию). Например, можно изменить размер ячеек, основываясь на уточнённых оценках длин проводников.

²⁾ Во время написания этой книги ведущим сторонником применения виртуальных прототипов на уровне регистровых передач (RTL) являлась компания InTime Software (www.intimesw.com).

1890 г. Америка.
В ходе переписи населения использовались 80-колоночные перфокарты Германа Холлерита (Herman Hollerith) и машины для автоматического составления таблиц.

кой. Генератор виртуального прототипа воспринимает данные в RTL-коде и преобразует его в таблицу соединений модулей, называемых *рабочими функциями*. Каждая рабочая функция представляет собой некую абстракцию, которая затем напрямую отображается на эквивалентную функцию в наборе конструктивных компонентов.

Преобразовав RTL-код в таблицу соединений рабочих функций, генератор виртуального прототипа формирует идентичные логические операции, которые обычно реализуются на уровне вентилях. В их число входят общие подвыражения исключений, константы распространения, развертывающиеся циклы, средства удаления излишних функциональных вычислений и так далее.

Генератор виртуального прототипа и подсистема анализа используют сформированную минимальную избыточную сеть рабочих функций для формирования их «виртуального размещения». Затем это размещение используется для создания оценок площади, занимаемой элементами на поверхности кристалла, которые, в свою очередь, будут использоваться для генерации точных временных параметров. Набор конструктивных компонентов, генератор виртуального прототипа и алгоритм анализа оценивают, каким образом различные методы синтеза будут определять значимость различных факторов, и затем производят модификацию стратегии их реализации. Эта процедура необходима для соответствия результата синтеза заданным временным ограничениям. Все эти факторы учитываются в процессе работы подсистемы анализа.

Сторонники виртуальных прототипов, основанных на уровне регистровых передач, говорят о 40-кратном скоростном преимуществе по сравнению с использованием таблицы соединений вентилях после этапа локальной оптимизации и перед этапом размещения и разводки. Например, в 2003 году в случае устройства, состоящего из 4.5 миллиона логических элементов, для создания и анализа виртуального прототипа на основе RTL потребовалось итерация длительностью 2.5 часа. В то же время для генерации и анализа таблицы соединения вентилях после этапа локальной оптимизации потребовалось 99 часов.

Конечно, остается еще получить ответ на главный вопрос: на сколько точно виртуальное макетирование на основе RTL? Сторонники этой формы виртуального макетирования утверждают, что результаты временного анализа обычно отличаются от данных, полученных из анализа таблицы соединений, не более чем на 20%, в худшем случае ошибка может составлять до 30%. Несмотря на то что это значение может показаться чрезмерно большим, замечу, что последние поколения средств синтеза дают погрешность временных параметров на уровне 20...30%. Но и такое значение не вызывает каких-либо проблем. Следовательно, виртуальное макетирование на основе RTL являются достаточно точным, чтобы позволить разработчикам создавать описание RTL, которое может быть использовано для реализации этапов низкочастотного синтеза и разводки элементов.

Знаю, знаю, мы снова отвлеклись. Но согласитесь, что этот материал весьма интересен. Ну а теперь давайте вернемся к размышлениям о ПЛИС.

Виртуальные прототипы ПЛИС

Не удивительно, что ПЛИС, содержащие миллионы вентилях, сталкиваются с теми же проблемами, что и заказные микросхемы, в том числе и с длительностью этапов размещения, разводки, временного анализа и другими.

Неприятной особенностью этого процесса является то, что хотя изначально представление уровня регистровых передач, как правило, является иерархическим, средства размещения и разводки ПЛИС обычно преобразуют его в плоское, или одноуровневое, представление устройства. Это значит, что если произвести незначительные изменения в одном из блоков RTL-кода и пересинтезировать только этот блок, всё равно придётся вернуться и произвести повторный синтез всего устройства. Ещё это значит, что вы можете стать седым и старым прежде, чем достигнете достижения временного соответствия для данного устройства.

Чтобы решить эти проблемы, некоторые поставщики САПР электронных устройств стали создавать средства, поддерживающие концепцию виртуального макетирования ПЛИС и обеспечивающие планирование компоновки кристалла и предварительный, т. е. до размещения и разводки, временной анализ. Подобный подход позволяет выполнять размещение и разводку индивидуально для каждого блока устройства, что значительно повышает скорость процесса его реализации¹⁾.

Эта форма виртуального макетирования начинается с просмотра графического представления ПЛИС, на котором отображаются все внутренние логические ресурсы (таблицы соответствия, регистры, слои, логические блоки, встроенные ОЗУ, умножители и так далее).

Затем последует процедура логического синтеза, которая будет выполняться по выбранному алгоритму, и после этого генератор виртуального прототипа осуществит загрузку иерархической таблицы соединений, выполненной на уровне таблиц соответствия или логических блоков, а также временных и конструктивных ограничений. На основе полученных данных генератор формирует начальный компоновочный план в виде квадратов и прямоугольников, каждый из которых соответствует модулю из верхнего уровня иерархии устройства. Если какой-то из этих модулей включает в себя один или несколько субмодулей нижнего уровня, эти субмодули отображаются на плане в виде встроенных блоков. И так далее вниз по иерархии.

Генератор виртуального прототипа формирует собственное начальное размещение ресурсов, т. е. таблиц соответствия, регистров, блоков ОЗУ, умножителей и других ресурсов, используемых каждым блоком. Эти ресурсы также отображаются на детальном плане устройства вместе с графическим представлением ресурсов, требуемых для связи различных блоков между собой.

Интерактивная правка

Предварительное размещение элементов устройства в виртуальном прототипе позволяет получить точные временные оценки на уровне блоков до их размещения и разводки. При обнаружении какой-либо проблемы можно интерактивно изменить компоновочный план устройства и оперативно исправить сбойный участок схемы.

Простейший способ интерактивной обработки заключается в изменении границ прямоугольников на компоновочном плане устройства, чтобы сделать их выше, тоньше, ниже или толще. Вместе с тем, также можно создать более сложные фигуры, например «L», «U» или

1890-е гг. Джон Венн (John Venn) предложил представление логических величин (диаграммы Венна) в виде окружностей и эллипсов.

Под иерархическим мы подразумеваем, что верхний уровень устройства обычно формируется из ряда функциональных модулей, которые могут состоять из субмодулей и так далее.

¹⁾ Во время написания этой книги, ведущим сторонником описываемой здесь формы виртуального макетирования ПЛИС являлась компания Hier Design (www.hierdesign.com).

1892 г. Америка.
Начал работу первый автоматический телефонный коммутатор.

«Т»-образные; из квадратов и прямоугольников можно получить большое количество разнообразных форм.

При интерактивной правке можно передвигать блоки с одного места на другое. При захвате блока и перетаскивании его через изображение поверхности устройства система будет автоматически производить графическую индикацию, уточняя, имеется ли в данном положении необходимое количество ресурсов для реализации перетаскиваемого блока; при этом блок можно перетащить только в ту область, где имеется необходимое количество свободных ресурсов. По мере внесения изменений в компоновочный план устройства путём изменения формы или перетаскивания прямоугольников система будет динамически отображать расход ресурсов, т. е. таблиц соответствия, регистров, блоков ОЗУ, умножителей и так далее, выделенных на текущий блок, по отношению к общему количеству всех ресурсов устройства.

Имеющиеся блоки также можно разделить на два или более субблоков, с которыми впоследствии можно произвести независимую правку. Можно также соединить несколько блоков в один. Иногда возникает необходимость вытащить один или несколько субблоков за пределы родительских блоков и перевести их на высший уровень иерархии устройства, где можно их перетаскивать, менять форму, соединять вместе и так далее.

Многое из рассмотренных аспектов интерактивной правки отражает различие подходов к применению одноуровневых виртуальных прототипов, используемых среди заказных микросхем. Например, в случае заказной микросхемы при наличии двух блоков с большим количеством соединений между ними, естественно разместить их бок о бок. В случае же ПЛИС эти блоки целесообразно соединить вместе, тем самым позволяя средствам размещения и разводки произвести более эффективную работу по оптимизации распределения локальных и глобальных ресурсов, что обеспечит более эффективное решение.

Кроме того, правка блоков не ограничивается только в пределах первоначального описания иерархии уровня регистровых передач, и у вас есть возможность манипулировать индивидуальными ресурсами ПЛИС, такими как таблицы соответствия, регистры, секции, логические блоки и так далее. Этот процесс подразумевает изменение их местоположения в пределах текущего иерархического блока, перемещение их из одного блока в другой, создание новых блоков и перемещение групп таблиц соответствия из одного или нескольких блоков во вновь созданный и так далее.

Самое «забавное» начинается при внесении изменений в первоначальный RTL-код и повторном синтезе этих модулей. В этом случае при повторном импорте результирующей таблицы соединений КЛБ (и таблиц соответствия), генератор виртуального прототипа производит сортировку и загрузку необходимой логики в соответствующие скомпонованные блоки. Я не представляю, как это происходит!

Поэтапные размещение и разводка

Итак, если вы готовы станцевать рок-н-ролл, можете выбрать один или несколько скомпонованных блоков и выкинуть программы размещения и разводки, полученные от поставщика вашей ПЛИС. Каждый блок воспринимается как отдельная единица, и после его формирования он будет оставаться неизменным до тех пор, пока вы не решитесь его изменить.

Этот подход имеет ряд преимуществ. Во-первых, время работы процедуры размещения и разводки для отдельного блока чрезвычайно мало по сравнению с разводкой многомиллионного устройства.

Во-вторых, если сложить время работы средств размещения и разводки для всех отдельно взятых блоков, полученное общее время всё равно будет существенно меньше, чем при работе аналогичных средств для всей схемы целиком. Объяснить это можно тем, что с возрастанием сложности устройства по мере увеличения размеров блока время работы по размещению и разводке увеличивается нелинейно. Кроме того, при размещении и разводке всех блоков можно внести изменения в некоторые из них и повторить процедуру разводки только для изменённых блоков, не затрагивая остальные части кристалла.

Еще одно преимущество виртуальных прототипов состоит в том, что они могут применяться для создания и сохранения блоков интеллектуальной собственности. Другими словами, после размещения и разводки блока его можно закрыть и экспортировать как новую структурную таблицу соединений уровня КЛБ и таблиц соответствия вместе с соответствующими физическими и временными ограничениями. Впоследствии этот блок может быть использован в других устройствах. Как уже отмечалось, расположение этого блока является относительным, т. е. он может быть перемещён в другую часть кристалла.

Виртуальные прототипы ПЛИС уровня регистровых передач

Естественно, было бы очень удобно работать с виртуальными прототипами ПЛИС, функционирующими на уровне регистровых передач (RTL). Некоторые поставщики ПЛИС и САПР электронных устройств представляют на рынке такие RTL-средства компоновки различной степени сложности. Тем не менее, на момент написания этой книги в мире не существовало виртуальных прототипов ПЛИС, которые соответствовали бы современному уровню развития виртуальных прототипов заказных микросхем, работающих на уровне регистровых передач. Но, несомненно, они появятся в недалеком будущем.

1894 г. Германия.
Генрих Герц (Heinrich Hertz) открыл, что радиоволны распространяются со скоростью света и могут отражаться и поляризоваться.

1894 г. Италия. *Гульельмо Маркони (Guglielmo Marconi) изобрел беспроводной телеграф.*