

# СИСТЕМА ЛОГИЧЕСКОГО МОДЕЛИРОВАНИЯ RIVIERA

## ПРОИЗВОДИТЕЛЬНОСТЬ ИМЕЕТ ЗНАЧЕНИЕ

Объемы логического моделирования, необходимые для верификации современных сложных цифровых микросхем, чрезвычайно велики, поэтому при выборе системы логического моделирования решающую роль играет фактор производительности. Система Riviera компании Aldec создавалась как инструмент, ориентированный на достижение максимальной производительности при работе с большими объемами тестовых воздействий, обеспечивающий возможность использования всех вычислительных ресурсов (на базе различных программно-аппаратных платформ), доступных разработчикам.

Любопытный факт: недавно компания Aldec анонсировала новые пакетные лицензии, предусматривающие установку тысячи и десятки тысяч копий системы логического моделирования Riviera. И некоторые компании приобрели такие пакеты. Действительно, для верификации современного микропроцессора требуется затратить тысячи часов на проведение логического моделирования. Поскольку время вывода продукта на рынок сегодня один из наиболее критичных факторов коммерческого успеха, компании стараются сократить сроки и обеспечить безошибочность проектирования, в том числе и за счет использования при логической верификации максимально доступного числа компьютеров. Система Riviera предусматривает возможность организации такой работы благодаря поддержке всех типов многомашиных серверных систем. К тому же производительность моделирующего ядра системы на сегодняшний день одна из лучших в своем классе. Поддержка методологии групповой разработки, высокая производительность моделирующего ядра и возможность подключения аппаратных ускорителей позволяют эффективно организовать отладку проектов на уровне регистровых передач, длительное регрессивное тестирование и временное моделирование на уровне списков цепей.

### ПОДДЕРЖКА СМЕШАННОГО МОДЕЛИРОВАНИЯ

Система Riviera позволяет интегрировать в рамках единого проекта модули, описанные на языках VHDL, Verilog, C/C++, SystemC, SystemVerilog, EDIF. Блоки, представленные на уровне списков цепей, могут быть дополнены данными о реальных задержках в различных стандартных форматах (например, SDF – Standard Delay Format). Все модули, независимо от формата их исходной спецификации, в процессе моделирования обрабатываются единообразно. Отлаженные блоки можно занести в библиотеку системы и использовать в последующих проектах без предварительной компиляции. При каждом старте моделирования иерархия проекта строится заново и производится инициализация блоков. Все библиотечные блоки, изначально



А.Бухтеев

описанные на различных языках, сшиваются единообразно и обрабатываются внутри одного моделирующего ядра.

Для разработки блоков на языках C/C++/SystemC система предусматривает интеграцию с компилятором C/C++ , всеми необходимыми файлами заголовков и библиотеками для поддержки различных типов C-приложений (SystemC, SCV, PLI, VHPI). Вопросы организации компиляции C-блоков решаются внутренними средствами, а поддержка SystemC встроена непосредственно в ядро системы моделирования, поэтому нет необходимости использовать промежуточные интерфейсы. Блоки на SystemC компилируются аналогично HDL-блокам, иерархия проекта может включать любую комбинацию SystemC-, HDL- и EDIF-блоков. Для SystemC-модулей доступна вся та же отладочная информация, что и для HDL-модулей. Кроме того, может быть подключен внешний отладчик C-кода (например, Microsoft Visual Studio). Поддерживаются стандарты разработки и использования приложений для популярной библиотеки SCV (SystemC Verification Library).

Сопряжение с модулями, созданными разработчиками на C/C++ , осуществляется на основе стандартных интерфейсов PLI и VHPI, разработанных комитетами IEEE соответственно для языков Verilog и VHDL. Эти интерфейсы позволяют проводить анализ иерархии проекта, контролировать прохождение сигналов в блоках и формировать события для моделирующего ядра. Доступно также большое число готовых C-моделей памяти, процессоров и других стандартных устройств.

В системе Riviera предусмотрены различные режимы моделирования, выбор которых позволяет организовать процесс таким образом,

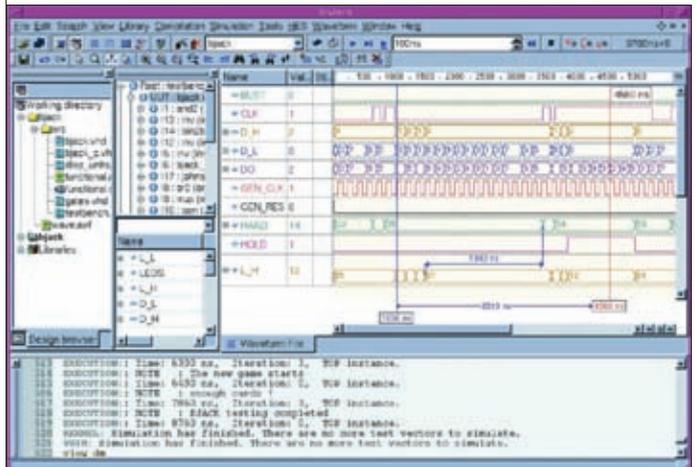
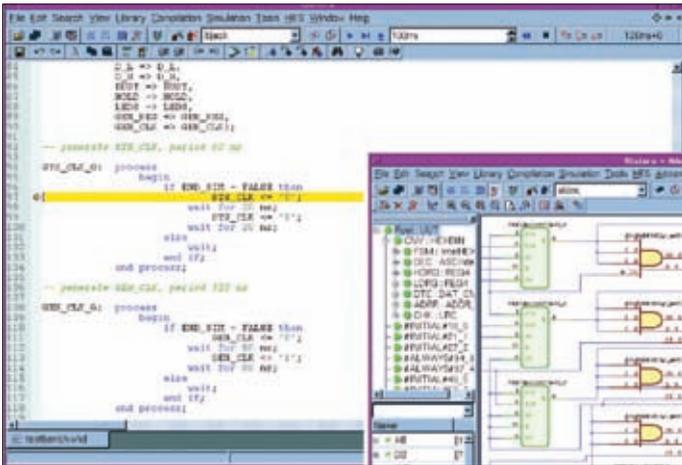


Рис. 1. Редактор временных диаграмм системы Riviera



**Рис.2. Поддержка взаимосвязи между различными представлениями проекта**

чтобы обеспечить достижение оптимальных параметров производительности. Задание режимов очень простое, переход от одного из них к другому не требует перекомпиляции кода. Управлять производительностью можно с помощью ограничения объема отладочной информации в неконтролируемых областях проекта, а также целого набора программных средств ускорения. Система располагает встроенными средствами акселерации моделирования VHDL-кода с библиотекой VITAL, средствами высокоуровневого моделирования, а также специальными методами оптимизации процесса моделирования на уровне списка цепей (в том числе с учетом файла задержек). Новая технология моделирования SLP (System-Level Platform) предусматривает использование дополнительного моделирующего ядра, ориентированного на обработку Verilog-блоков на уровне списков цепей. Система автоматически распознает типы модулей и распределяет задачи между стандартным ядром моделирования и дополнительным SLP-ядром.

Моделирование можно проводить в пакетном режиме (при отключении графического режима производительность возрастает) с сохранением истории переключений всех сигналов для их последующего анализа. База данных, в которой сохраняются результаты, построена таким образом, чтобы минимизировать влияние записи промежуточных результатов на время моделирования.

### СРЕДСТВА ОТЛАДКИ

Анализ временных диаграмм остается основным способом отладки описания проекта, поэтому наличие высокопроизводительного редактора диаграмм жизненно важно для пользователей системы логического моделирования. В системе Riviera все результаты записываются в файлы, структура которых, с одной стороны, позволяет минимизировать влияние записи результатов на процесс моделирования, а с другой — обеспечивает быструю загрузку в редактор временных диаграмм (1 Гбайт данных загружается в течение нескольких секунд). Прокрутка диаграмм на экране осуществляется без осязательной для пользователя задержки. Можно создать несколько конфигураций для просмотра и сохранять их в отдельных файлах, ассоциированных с одним и тем же файлом результатов. Предусмотрены различные механизмы просмотра диаграмм, такие как временные закладки, переход курсора к заданному временному интервалу, к следующему событию выбранного сигнала и другие (рис. 1).

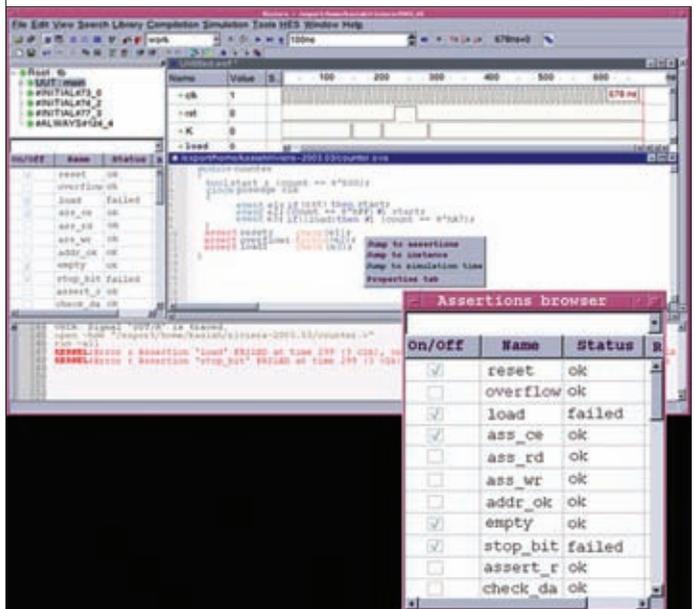
Редактор временных диаграмм интегрирован с моделирующим ядром и редактором HDL-описания. При интерактивной отладке результаты отображаются в режиме реального времени, можно управлять процессом моделирования, изменяя текущие значения сигналов

в ходе моделирования. При выборе события на временной диаграмме система покажет его источник в HDL-коде. Предусмотрены функции автоматического сравнения результатов. Все различия отображаются на временных диаграммах удобными маркерами с возможностью автоматического перехода от одного маркера к другому. Одновременно с временными диаграммами можно просматривать результаты и в табличной форме, причем все варианты представления результатов будут синхронизированы между собой.

Текстовый редактор описания проектов интегрирован с компиляторами и ядром моделирования. Здесь есть возможность провести синтаксический и семантический анализ описания, указать точки останова при моделировании, организовать пошаговое выполнение проекта. При необходимости можно перейти к нужной временной диаграмме в редакторе, к развернутому (до нужного уровня иерархии) графическому представлению определенных цепей, а также контролировать сигналы и управлять ими в любом месте иерархии проекта (рис.2).

В системе Riviera реализована всесторонняя поддержка активно развивающейся в последнее время методологии ABV (Assertion Based Verification). Методология предусматривает поддержку встроенных в код функций Assert, в которых проверяется истинность утверждений, характерных для правильного функционирования устройства. Система поддерживает ABV-верификацию на базе языков Open Vera (OVA), PSL и SystemVerilog. Assert-функции органично встраиваются в иерархию проекта, отражаются и выделяются в текстовых и графических средствах, поддерживаются механизмом перекрестных ссылок, допускают установку точек останова (рис.3). Моделирующее ядро системы можно сконфигурировать с учетом Assert-функций, чтобы сделать более удобной отладку в интерактивном режиме.

Анализ тестового покрытия позволяет разработчику оценить эффективность используемой системы тестов. Обычно включение этой функциональности значительно замедляет процесс моделирования, однако в системе Riviera функции анализа тестового покрытия интегрированы в моделирующее ядро, что существенно сокращает накладные расходы, связанные с проведением такого анализа. Система располагает средствами анализа использования строк кода (с учетом модуля и экземпляра модуля) и ветвей условных операторов, а также анализа переключений сигналов. Поддерживается функция интегральной оценки тестового покрытия на основе результатов, получен-



**Рис.3. Использование Assert-функций при отладке в системе Riviera**

ных на множестве независимых прогонов моделирования. Предусмотрены средства графической визуализации тестового покрытия (рис.4) для всех видов анализа.

Функция профилирования проекта направлена на выявление блоков, моделирование которых требует большого времени. Оптимизация кода таких блоков может значительно сократить время моделирования. Графические средства визуализации результатов профилирования позволяют облегчить анализ временных затрат и выбрать правильный вариант оптимизации кода или принять решение об аппаратном варианте моделирования наиболее затратных модулей проекта.

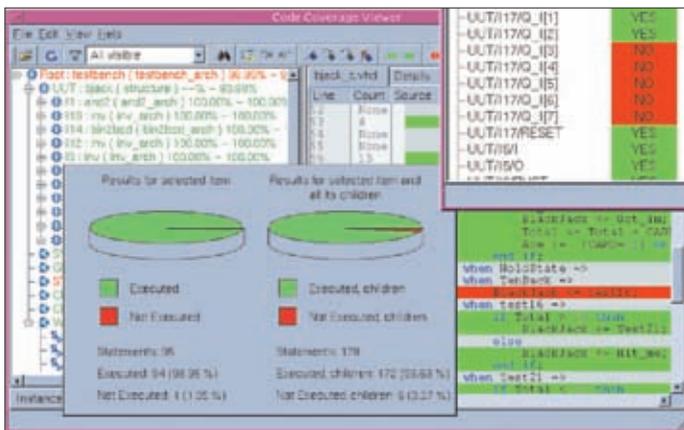


Рис.4. Анализ тестового покрытия

## АППАРАТНОЕ МОДЕЛИРОВАНИЕ

Поддержка аппаратного моделирования в системе реализована в рамках методологии Riviera-IPT (Incremental Prototyping Technology), которая предусматривает инкрементальный перенос отлаженных программными средствами блоков в аппаратную форму. Таким образом, в каждый момент времени программно моделируются только новые блоки, а все отлаженные обрабатываются аппаратно, что существенно (в 10–50 раз) сокращает общее время моделирования.

Помимо системы моделирования Riviera, Riviera-IPT включает менеджер проекта DVM (Design Verification Manager), платы для аппаратной реализации фрагментов проекта на базе ПЛИС, а также средства третьих фирм для синтеза, размещения и трассировки на ПЛИС (рис.5). Менеджер DVM обеспечивает поддержку разбиения проекта на части, моделируемые аппаратно и программно, реализацию аппаратных фрагментов с помощью средств третьих фирм на платах аппаратного ускорителя и организацию взаимодействия системы логиче-

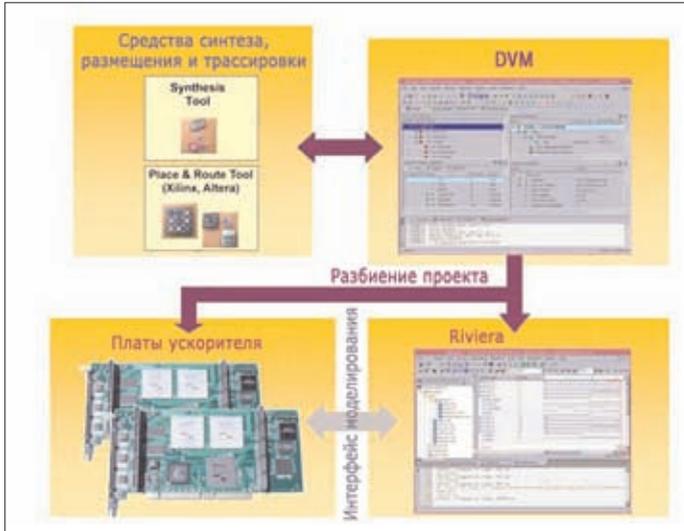


Рис. 5. Компоненты системы Riviera-IPT

ского моделирования и аппаратной части. Надо заметить, что при использовании средств аппаратного ускорения обеспечивается связь с редактором временных диаграмм и возможность визуализации внутренних сигналов аппаратной части. Поддерживаются функции Assert и верификация с использованием моделей на языках C/C++/SystemC.



Рис.6. Установка плат ускорителя в стек при внешнем подключении

Платы аппаратного ускорителя имеют стандартный PCI-интерфейс. Кроме установки в стандартные PCI-слоты системных блоков может использоваться внешнее PCI-подключение с установкой плат в стек (рис.6). Емкость каждой платы может варьироваться от одного до двенадцати миллионов вентилях ПЛИС. Помимо памяти (64 Мбайт–1 Гбайт), размещенной непосредственно на платах ускорителя, предусмотрено использование дополнительных плат (до 6 Гбайт) для аппаратной реализации различных моделей памяти. Существуют также специальные дополнительные платы, предназначенные для моделирования систем с процессорами ARM.

Система Riviera-IPT поддерживает несколько режимов верификации. В режиме аппаратного ускорения процесс верификации полностью контролируется программой моделирования. Производительность моделирования в этом режиме составляет от 10 до 200 кГц. При использовании готовых тестовых векторов, полученных при верификации в предыдущем режиме, производительность может достигать от 100 до 500 кГц (за счет того, что верификация идет без программного моделирования). Такая же производительность достигается в случае применения тестов на SystemC непосредственно для верификации реализованного на платах устройства. А максимальная производительность (от 100 кГц до 8 МГц) может быть достигнута при программно-аппаратной верификации систем на базе ARM-процессоров.

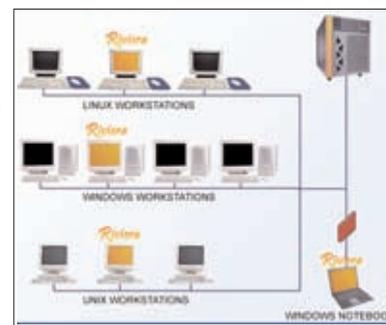


Рис.7. Использование системы Riviera в многомашиных комплексах

В заключение хотелось бы еще раз отметить, что как аппаратные средства ускорения, так и базовые средства моделирования включают расширенную поддержку групповой работы с многомашиными серверными системами в средах Unix, Linux или Windows на различных аппаратных платформах (рис.7). В целях повышения производительности для каждой аппаратной платформы применяются версии исполняемого кода, оптимизированные непосредственно для используемых типов процессоров. Разработчики могут распределить свои задачи между доступными вычислительными ресурсами и освободить свой персональный компьютер или рабочую станцию для решения текущих задач. При работе в группе предусмотрены средства компрессии и кодирования для безопасного и высокопроизводительного обмена проектными данными между разработчиками. Средства управления вычислительным процессом моделирования достаточно просты и эффективны, чтобы обеспечить поддержку работы с многомашиным комплексом, в котором установлены сотни и тысячи копий системы.