



УДК 004.315

О. В. Непомнящий

*канд. техн. наук, доц.
Сибирский федеральный университет, г. Красноярск,
Красноярский край, Россия*

А. В. Хныкин

*канд. техн. наук
Железногорский филиал СФУ, г. Железногорск,
Красноярский край, Россия*

АНАЛИЗ ПРОЕКТИРОВАНИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ НА КРИСТАЛЛЕ

Рассмотрены современные проблемы функциональной и системной верификации проектирования сверхбольших интегральных схем в целом и систем на кристалле в частности. Приведен обзор методик верификации и предложены пути решения означенных проблем.

Ключевые слова: функциональная верификация, системная верификация, автоматизированное проектирование, СБИС, система на кристалле.

O. V. Nepomnyashy

Siberian federal university, Krasnoyarsk, Russia

A. V. Khnykin

The Branch of Siberian Federal University in Zheleznogorsk, Russia

ANALYSIS OF COMPUTER SYSTEM DESIGN ON THE CHIP

The article deals with modern problems of functional and system verification of very large scale integrated (VLSI) circuits and systems on a chip in particular. There is a review of existing methods of verification and propose ways to solve the aforesaid problems.

Key words: functional verification, system verification, CAD, VLSI, system on chip.

Степень интеграции современных сверхбольших интегральных схем (СБИС) растет экспоненциально [1]. Сегодня переход к субмикронным технологическим нормам позволяет размещать на кристалле практически любой по сложности проект. Тем не менее имеется сдерживающий фактор – проблема функциональной верификации СБИС. При разработке сложнофункциональных устройств, т.е. систем, включающих не только цифровые, но и аналоговые, смешанные, а также процессорные ядра со встроенным про-

граммным обеспечением, означенная проблема стоит наиболее остро.

По последним данным [2], примерно половина всего инженерного состава, работающего над крупными проектами, занята функциональной верификацией, а временные затраты на нее в общем цикле проектирования впечатляют еще сильнее – более 60 %.

Анализ проблемы показал, что технологии верификации проекта на нынешний день заметно отстают от технологий и вычислительных возможностей систем проектирования. Еще более серьезным видится отставание возможностей верификации от техноло-

гических возможностей производства СБИС (рис. 1).

В то же время затраты на создание комплекта фотошаблонов для субмикронных СБИС настолько высоки, что их повторное изготовление из-за обнаруженных ошибок зачастую недопустимо как с точки зрения задержки выхода изделия на рынок, так и в связи с ростом его конечной стоимости [3].

Поэтому необходим поиск принципиально новых решений, разработка передовых технологий верификации и подход к новым методам анализа проекта.

Основным недостатком традиционного подхода к функциональной верификации яв-

При таком принципе нисходящего проектирования каждый блок проектируется и верифицируется отдельно, и только после объединения в более крупные блоки происходит совместная верификация. Таким образом, более мелкие блоки приходится верифицировать дважды. Сначала функционирование блока тестируется на соответствие его спецификации, а затем на соответствие общим требованиям в составе интегрированной системы. При таком подходе имеется высокая вероятность обнаружения ошибки на одном из последних этапов интеграции, в особенности – на уровне системной верификации, и как следствие – возврат к началу блочного проектирования. Все

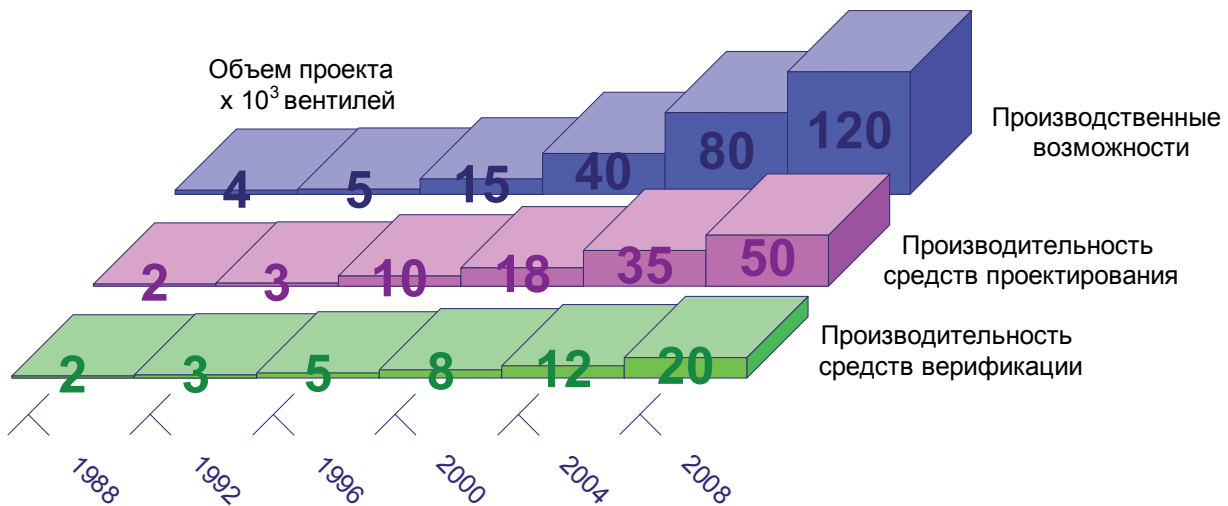


Рис. 1. Динамика роста производственных возможностей в отношении к производительности средств проектирования и верификации СБИС

ляется общепринятый алгоритм проектирования. Дело в том, что здесь система сначала разрабатывается и лишь затем – тестируется [4]. При 60–70 %-й загрузке по времени на верификацию такая последовательность действий недопустима. При увеличении числа транзисторов на кристалле на первое место выходит факт резкого увеличения необходимого числа и длины тестовых векторов и, как следствие, размера и сложности программно-аппаратных средств тестирования. Параллельно усложняется процесс поиска неисправности, обнаруженной тестом.

Традиционный подход к проектированию и верификации означает, что проектирование начинается с разработки спецификации на систему в целом с последующим разбиением системы сначала на крупные, а затем на меньшие блоки, реализуемые на уровне RTL-описания [4].

это приводит к недопустимым затратам времени и ресурсов на внесение изменений.

Итак, с увеличением степени интеграции СБИС процесс верификации усложняется, но следует учесть и функциональные возможности современных систем на кристалле, которые тоже растут с экспоненциальной скоростью. В состав современной системы на кристалле могут входить несколько высокопроизводительных процессоров, высокоскоростные интерфейсы ввода-вывода, встроенные модули памяти, аналоговые, смешанные и СВЧ-модули обработки данных. Управление такими системами может осуществляться при помощи программного обеспечения, базирующегося на встроенных операционных системах и т.д.

Таким образом, на первый план выходит системная верификация – т.е. верификация взаимодействия, а не верификация отдельных

блоков на блочно-модульном уровне. Здесь особую сложность представляет взаимодействие цифровых подсистем с аналоговыми и СВЧ-схемами.

На современном этапе развития применение готовых модулей «интеллектуальной собственности» – IP-блоков от сторонних фирм в составе проекта также создает целый ряд проблем в области верификации. Дело в том, что IP-модули поставляются, как правило, в виде «черных ящиков», не позволяющих анализировать их внутреннее поведение [5].

На рис. 2 изображена диаграмма зависимости требуемой скорости верификации от сложности объектов.

Проведенный анализ текущего состояния проектов СБИС, в том числе систем на кристалле, реализуемых ведущими компаниями, показал, что сегодня решить проблему функциональной верификации только путем повышения производительности традиционного метода моделирования на уровне RTL не представляется возможным.

процесса создания программно-аппаратных средств тестирования с разработкой методологии сквозной верификации проекта на всех этапах.

Современные средства проектирования должны обеспечивать сквозную верификацию на всех уровнях по преимущественным направлениям. Здесь необходимо использовать технологии имитационного моделирования, аппаратной эмуляции, интегрированной программно-аппаратной верификации и обязательно аналого-цифрового смешанного моделирования. Программные системы создания и верификации проекта должны поддерживать все стандартные языки проектирования, в том числе весь ряд HDL (VHDL, Verilog, VHDL_AMS, Verilog_A). Также должны поддерживаться системные языки (Spice, C, C++, SystemC, System Verilog, MATLAB, PSL assertions и др.)

В плане выработки перспективных методологий верификации следует отметить создание системных тестов, моделирование

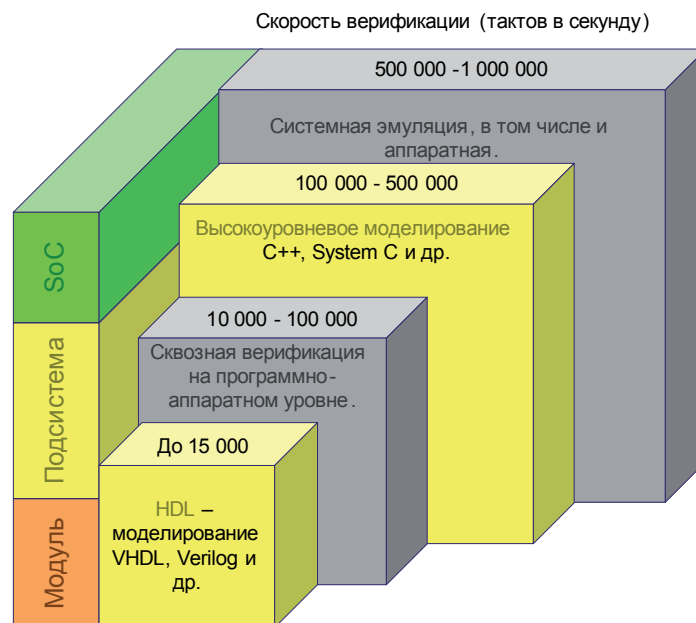


Рис. 2. Требуемая скорость верификации при переходе к более сложным объектам

В [6] рассматриваются два общих подхода к решению означенных проблем. Первый подход предполагает усовершенствование средств тестирования, а второй заключается в изменении самой методики верификации, а именно – в переносе соответствующих процедур верификации на более ранние этапы проектирования.

Тем не менее для решения проблем верификации следует осуществить интеграцию

на уровне транзакций, верификацию интерфейсов различных подсистем одновременно с их проектированием, т.е. возможность верификации системы, отдельные блоки которой представлены на разных уровнях абстракции.

При таком подходе возможно проектирование на системном уровне, т.е. создание моделей высокого уровня на первоначальных этапах проекта. Такие модели могут быть

созданы на основе C, C++, SystemC и System Verilog, при этом детали каждого отдельного блока не будут рассматриваются. Такой подход позволит верифицировать систему, не ожидая детальной проработки всех блоков и интерфейсов. Для реализации «узких» мест – интерфейсов между блоками следует использовать механизм транзакций.

На основании вышеизложенного рассматривается новая методология верификации, которую характеризует следующее:

- разработка архитектурного плана системы верификации на первоначальных этапах при определении спецификации системы;
- реализация принципа нисходящего проектирования с поитерационной детализацией блоков до уровня RTL. При этом реализуются возможности смешанного моделирования подсистем, представленных на различных уровнях;
- принцип нисходящей верификации проекта от системного до вентиляционного уровней как в плане формального описания, так и в динамическом и статическом режимах функционирования;
- верификация принятых архитектурных решений на системном уровне с помощью построения моделей подсистем верхних уровней на языках C++, SystemC, SystemVerilog и организация моделей подсистемных интерфейсов на уровне транзакций;
- автоматическая генерация тестбенчей – виртуальных объектов, содержащих тестируемый модуль и подсистему тестовых воздействий, причем как на системном, так и на RTL уровнях;
- возможность эмуляции отдельных модулей на аппаратном уровне, в том числе и при использовании внутрисхемных эмуляторов, при которой можно подключать модели разного уровня абстракции с управляющим виртуальным или эмулируемым процессором;
- возможности создания интегрированного виртуально-аппаратного прототипа посредством интегрированной системы программно-аппаратной верификации, реализующей совместное моделирование программной и аппаратной части проекта, включающей построение моделей на языках высокого уровня и интерфейсных моделей посредством транзакций;
- возможность применения встроенных функций проверки истинности утверждений, характерных для правильного функционирования устройства (assertion-based verification), и оптимизации функционального покрытия (coverage driven verification);
- стандартизация в области представления данных и языковых средств проектирования;
- поддержка проектирования с помощью блоков интеллектуальной собственности – IP для организации стандартных подсистем (I2C, USB, SCSI и т.д.).

Теперь можно сформулировать основные требования к перспективным системам верификации. Система должна обеспечивать:

- исключение верифицированных блоков и моделирование только новых добавляемых блоков;
- моделирование на разных языках, т.е. система должна иметь возможность выполнять сборку наиболее сложных проектов из проектных блоков на языках VHDL, Verilog, C/C++, SystemC и им подобных, а также списков цепей в стандартизированном формате, при этом следует обеспечить их полную интеграцию на основе общего модельного ядра.
- смешанное языковое моделирование, например, иметь возможность совместного моделирования тестов и проектных модулей C/C++ и SystemC вместе с модулями VHDL и Verilog. При этом компилятор C должен встраивать в систему моделирования, а результаты комбинированного моделирования могут просматриваться в редакторе временных диаграмм;
- оптимизацию процесса моделирования, в том числе осуществлять компиляцию и моделирование проектов на языках описания аппаратуры и смешанных языках совместно с тестами. Распределять системную память между модулями проекта в необходимом и достаточном объеме для верификации. Производить выгрузку не используемых в процессе верификации блоков из проекта на время моделирования;
- использование передовых технологий верификации, например верификации на основе утверждений с аппаратной поддержкой и поддержкой стандартов утверждения OpenVera (OVA), языка определенных свойств (PSL), библиотеки открытой верификации (OVL);

- поддержку интерфейсов верификации в стандартах IEEE, в том числе интерфейсов PLI, VPI и VHPI, для связи с другими средствами верификации на различных этапах проектирования. Кроме того, должна быть заложена возможность сопряжения со средствами проектирования сторонних фирм путем подключения нестандартизированных протоколов обмена и предоставления информации;
- поддержку параллельных вычислений и многомашинных серверных систем. Система должна быть совместима со всеми методологиями управления загрузкой многомашинных серверных систем и систем параллельных вычислений, конфигурируемых для различных операционных систем с целью разгрузки локального компьютера при моделировании;
- защиту информации при многопользовательском проектировании в виде упаковки и кодирования библиотек исходных файлов при обмене проектными модулями между членами рабочей группы локально или по сети;
- анализ тестового покрытия по строкам кода, по переключениям, по ветвлению, интегральное покрытие на множестве тестов (слияние) и графическую визуализацию тестового покрытия;
- выявление блоков повышенного потребления ресурсов при моделировании и выполнение их оптимизации с целью предотвращения деградации моделирования;
- наличие в составе среды высокопроизводительного редактора временных диаграмм, использующего различные методы сжатия для обработки больших объемов данных при моделировании и ускорения визуализации временных диаграмм. Необходимым требованием по реализуемым функциям редактора является наличие функции редактирования диаграмм с возможностью модификации цепей, поддержки форматов VCD, отображения источников событий, сравнения временных диаграмм и др.;
- поддержку встроенного многоуровневого текстового редактора языков описания аппаратуры, интегрированного с компилятором и ядром моделирования, обеспечивающего весь спектр функций отладки от генерации точек останова до локализации ошибок компиляции;

- поддержку аппаратных ускорителей с целью увеличения скорости и эффективности моделирования за счет соединения вместе многих различных элементов проектирования и верификации в единую ускорительную платформу системного уровня.

Современные технологии проектирования систем от компаний Mentor Graphics, Aldec и др. позволили снизить разрыв между производственными возможностями и производительностью средств верификации. Однако вопросы разработки технологий и методик верификации крупных проектов остаются открытыми. Здесь имеются существенные пробелы языкового плана, проблемы совместимости проектов и интерфейсов, проблемы защиты информации при сетевом и многопользовательском проектировании, проблемы международной стандартизации и др.

Рассмотренные проблемы в области верификации проектов и предложенные пути их решения позволят создавать системы верификации для более гибких интегрированных аппаратно-программных платформ класса «аппаратный ускоритель – язык моделирования», предназначенных для смешанного моделирования от системного уровня до уровня регистровых передач. С помощью таких платформ можно будет выполнять процедуры длительного регрессивного тестирования и моделирования во времени, а также применять методы и средства групповой разработки.

Библиографические ссылки

1. Rosenstiel W. Rapid Prototyping, Emulation and Hardware/Software Co-debugging. In System-Level Synthesis, edited by A.A. Jerraya and J. Mermel, NATO Science Series, Kluwer Academic Publisher, 1999.
2. Немудров В., Мартин Г., Системы-на-кристалле. Проектирование и развитие. М. : Техносфера, 2004. 216 с.
3. Стешенко В. Б., Руткевич А. В., Бумагин А. В. и др. Опыт разработки СБИС типа СнК на основе встроенных микропроцессорных ядер // Компоненты и технологии. 2008. № 9.
4. Рабаи Ж. М., Чандракасан А., Николитч Б. Цифровые интегральные схемы. Методология проектирования : пер. с англ. 2-е изд. М. : ООО «ИД Вильямс», 2007. 912 с.
5. Бухтеев А. В. Методы и средства проектирования систем на кристалле // Chip news. 2003. № 4.
6. Лохов А. Функциональная верификация СБИС в свете решений Mentor Graphics // Электроника: наука, технологии, бизнес. 2004. № 1.