

Аппаратное ускорение цифрового моделирования

В.С. Зайцев

Белорусский государственный университет, zaicevvs@tut.by

Аннотация — Описывается один из возможных вариантов ускорения процесса моделирования цифровой синхронной интегральной схемы за счет использования FPGA прототипа вместо ее HDL модели. Проводится обзор существующих решений такого рода, анализ производительности портов компьютера на предмет их использования при реализации бюджетного решения системы ускорения.

Ключевые слова — *FPGA прототип, SystemC, моделирование, COM-порт, LPT-порт, USB-порт.*

I. ВВЕДЕНИЕ

Верификация проекта цифровой интегральной схемы (ИС) - один из самых затратных по времени этапов разработки. Для достижения достаточной полноты проверки необходимо провести моделирование на большом наборе тестов и, если выявлены ошибки, после корректировки необходимо весь процесс моделирования повторить. Если схема содержит большое число элементов, то после определенного этапа верификации переходят от моделирования к работе с прототипом. Прототип функционирует на частоте реальной схемы и позволяет максимально быстро проводить тесты функционального контроля. Работа с прототипом реального устройства требует специальных программных средств верификации и их коммуникации. Кроме этого необходима разработка отдельного тестового окружения или адаптация существующего для подачи входных воздействий и анализа реакции прототипа на них.

Это приводит к дополнительным затратам средств и времени на этапе верификации. Оптимальным, с точки зрения временных затрат, было бы решение использовать в одной системе среду верификации для генерации воздействий и анализа ответов, разработанную для моделирования HDL-описания, и прототип.

II. СУЩЕСТВУЮЩИЕ РЕШЕНИЯ

Время, затрачиваемое на проведение одного теста в программной среде моделирования на HDL модели разрабатываемой ИС, объективно всегда значительно больше времени обработки того же теста в ИС. Этим объясняются многочисленные попытки использования для выполнения моделирования различных систем на базе аппаратного прототипа ИС непосредственно в стандартной среде верификации.

Такие решения реализованы фирмами Aldec, Synopsys, Mentor Graphics. Решение, предлагаемое фирмой Aldec [1], включает программное обеспечение (ПО) для моделирования HDL-моделей (Riviera-PRO), ПО для прошивки прототипа (Active-HDL) и специальную плату (например, ALDEC HES-5, 6, 7) для прототипирования с каналом подключения через PCIe порт к компьютеру. В состав платы в зависимости от ее серии входит несколько микросхем FPGA Virtex-7, ARM® Cortex™-A9 и целый спектр портов для подключения периферийных устройств. Число вентилях в этом решении, доступных для размещения прототипа, от 4 до 96 миллионов. Цена системы без учета ПО от 20000\$.

Аналогичный набор предлагает и фирма Synopsys: ПО для прошивки FPGA и моделирования Synopsys Virtualizer, система для прототипирования Synopsys HAPS [2]. Система HAPS-6 в зависимости от серии содержит до 18 микросхем FPGA фирмы Xilinx Virtex-6 LX760, что позволяет получить решение, содержащее до 81 миллионов ASIC вентилях, что эквивалентно 13,5 миллионов конфигурируемых ячеек в FPGA. Новая система 2013 года HAPS-7 содержит 12-144 миллионов ASIC вентилях, а стоимость комплекта без учета стоимости лицензии на включение в ПО опции работы с прототипом в диапазоне от 20 000\$ до 800 000\$ для самых дорогих решений.

Решение фирмы Mentor Graphics – система Veloce и Veloce2. Решение включает в себя ПО и стационарную систему, содержащую от 16 до 256 плат, предоставляющих разработчику возможность конфигурировать цифровую схему с числом вентилях от 256 млн. до 2 млрд.

Особенности алгоритма работы данных систем не документированы и являются коммерческой тайной производителей, но анализ спецификаций и конструкции данных решений позволяет получить общую схему работы таких систем (рис. 1).

Во время работы система выполняет определенную последовательность действий. Вначале запускается выполнение тестов на ПК в среде моделирования, во время запуска указывается часть схемы, которая должна быть имплементирована в FPGA. Эта часть схемы синтезируется и прошивается в одной или в целом массиве FPGA микросхем. Далее формируется список наблюдаемых сигналов и список сигналов, через которые будут поступать входные воздействия в прототип. Согласно этому списку формируется модуль, написан-

ный на языке SystemC/C++ [3], с аналогичным интерфейсом, выполняющим функции отправки, приема данных из прототипа и подключения к существующему тестовому окружению. В среде верификации синтезируемая модель заменяется на сгенерированный модуль и запускается моделирование.

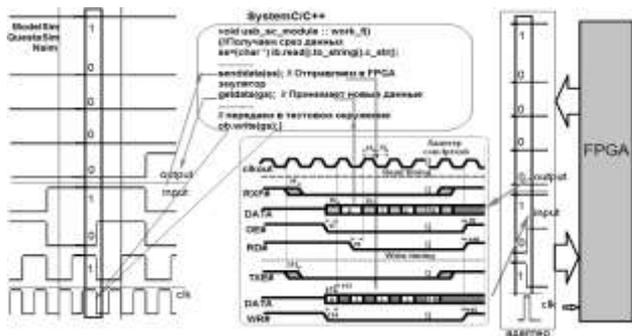


Рис. 1. Алгоритм работы системы ускорения моделирования

В процессе моделирования SystemC модуль отслеживает события на сигналах, определенных как входные, и формирует срез данных по каждому изменению сигнала из этого списка, если схема асинхронная, или по фронту тактового сигнала, если синхронная. Срез данных отправляется из ПК к прототипу, где устанавливаются соответствующие значения на входах. Прототип обрабатывает входные данные и формирует ответ, который отправляется в компьютер. SystemC модуль принимает данные и устанавливает их в среде моделирования как выходные. Выходные данные анализируются в тестовом окружении и формируются новые входные данные. Далее цикл повторяется до конца моделирования.

Эти системы обладают некоторыми недостатками – большая цена и привязка к среде моделирования. Большая цена таких решений ограничивает возможность их использования в проектах с небольшим бюджетом, а привязка к программному обеспечению не позволяет применять аппаратную часть на фирмах, уже имеющих лицензии на софт от другого производителя или использующих бесплатное программное обеспечение.

III. КОНЦЕПТ РЕШЕНИЯ СИСТЕМЫ УСКОРЕНИЯ МОДЕЛИРОВАНИЯ

Оптимальный вариант такой системы выглядит следующим образом:

- 1) аппаратная составляющая такой системы должна позволять разместить в ней схему и при этом находиться в рамках бюджета проекта;
- 2) программное обеспечение для моделирования – от любого производителя с поддержкой цифрового смешанного моделирования, где одним из доступных языков является SystemC;
- 3) блок подключения прототипа к компьютеру – адаптер, имеет пропускную способность и скорость

работы позволяющую увеличить скорость моделирования, иначе нет смысла его использовать.

Также эта система, кроме ускорения обработки, должна обеспечивать интерактивный режим работы с пошаговым отображением результатов на временных диаграммах, как будто обработка выполняется без использования аппаратного прототипа.

Известно, что в среде программного моделирования имитируется параллельный характер обработки информации, присущий аппаратуре. В соответствии с этим, при каждом переключении тактового сигнала компьютер последовательно обрабатывает все активные в данный момент времени операторы HDL-описания. При этом обработка на данном такте считается завершённой лишь после обработки всех таких операторов. Из этого следует, что при таком подходе модельное время одного такта обработки всегда больше его реального времени в аппаратуре и зависит от общего числа элементов в ИС, структуры их соединения, стиля и языка описания модели, производительности компьютера. При реализации в такой системе интерактивного режима обработки узким местом является производительность компьютера.

При использовании же аппаратного прототипа обработка всех активных на текущем такте сигналов выполняется параллельно. В этом случае реализация отображения результатов моделирования в пошаговом режиме требует передачи входных, считывания результирующих данных, их обработки и отображения в течение времени выполнения такта в прототипе. Именно в таком случае достигается максимальный выигрыш от использования в системе моделирования прототипа вместо его HDL-модели. Узким местом в таких системах ускорения [4] моделирования является шина обмена данными между персональным компьютером и FPGA прототипом, т. е. если шина обмена позволяет провести итерацию отправки и приема быстрее, чем компьютер произведет расчет нового состояния, то система ускоряет процесс моделирования.

IV. ПОРТЫ ПОДКЛЮЧЕНИЯ ПРОТОТИПА К КОМПЬЮТЕРУ

Порты подключения при реализации описанной выше системы должны удовлетворять следующим требованиям:

- 1) направления передачи данных: как прямое, так и обратное;
- 2) минимальная задержка между посылкой и приемом данных;
- 3) скорость передачи данных позволяет увеличить скорость моделирования.

Доступными для подключения периферии портами в ПК являются COM, LPT, USB. Был проведен анализ этих портов компьютера на предмет использования в системе аппаратного ускорения моделирования (рис. 2).

При расчете производительности ускорителя принято предположение - описание достаточно сложное и процесс проведения теста на компьютере занимает больше времени, чем в реальной схеме. Данное предположение правомерно, так как, например, моделирование в QuestaSim на частоте 1 МГц в течение 1 сек модельного времени двух 32-разрядных сумматоров, каждый разряд которого суммируется по модулю два, занимает 5.4 с, на компьютере AMD64 2.7 ГГц 6 ГБ DDR2. Таким образом, скорость моделирования схемы не 1 МГц, а 185 кГц - в 5.4 раза меньше частоты работы реальной схемы.

А. Последовательный СОМ-порт

СОМ-порт обладает следующими преимуществами – порт двунаправленный, порт имеет минимальную задержку чтения/записи, так как задержка на фоне длительности передачи одного байта данных незначительна. К недостаткам можно отнести невысокую скорость передачи: порт производит обмен данными на максимальной частоте 128 кГц, передача осуществляется последовательно, минимальная длина посылки 10 бит (стартовый бит, 8 бит данных, стоп бит), а максимальная длина может достигать 12 бит, если представлены 2 стоп бита и бит четности. В итоге максимальная частота, на которой выполняется функция посылки данных в порт плюс функция чтения данных из порта, равна $128 \text{ кГц} / 20 = 6.4 \text{ кГц}$. Стоит отметить, что в системах такого рода частота выполнения посылки и приёма данных обратно пропорциональна количеству наблюдаемых сигналов.

Обозначим частоту синхросигнала моделируемой ИС через X , число ее входов через N , а число выходов через M . Частота обмена запрос/ответ по шине подключения должна быть не меньше $F = X \times (N + M)$. В этом случае моделирование будет проходить в режиме реального времени с небольшой задержкой на выполнение прорисовки сигналов и обработку внутри блока тестового окружения. В итоге, получаем зависимость максимальной частоты работы системы, подключенной через СОМ-порт, от числа наблюдаемых входных и выходных сигналов (табл. 1).

В. Параллельный LPT-порт

Стандарт позволяет использовать LPT-порт в нескольких режимах:

- 1) SPP (Standard Parallel Port) – однонаправленный порт, полностью совместимый с интерфейсом Centronics.
- 2) Nibble Mode – позволяет организовать двунаправленный обмен данными в режиме SPP путём использования управляющих линий (4 бит) для передачи данных от периферийного устройства к контроллеру. Исторически это был единственный способ использовать Centronics для двустороннего обмена данными.
- 3) Byte Mode – редко используемый режим двустороннего обмена данными. Использовался в некоторых старых контроллерах до принятия стандарта IEEE 1284.

4) EPP (Enhanced Parallel Port) – разработан компаниями Intel, Xircom и Zenith Data Systems – двунаправленный порт, со скоростью передачи данных до 2 Мбайт/с.

5) ECP (Extended Capabilities Port) – разработан компаниями Hewlett-Packard и Microsoft. Появились такие возможности, как наличие аппаратного сжатия данных, наличие буфера.

Таблица 1

Зависимость частоты эмулятора от числа наблюдаемых сигналов

Сумма отправленных и полученных сигналов	Частота, Гц		
	СОМ порт	LPT порт	USB порт
16	6400	185000.0	500
32	3200	92500.0	500
64	1600	46250.0	500
128	800	23125.0	500
256	400	11562.5	500
512	200	5781.2	500
1024	100	2890.6	500
2048	50	1445.3	500
4096	25	722.6	500
8192	12.5	361.3	500
16384	6.25	180.6	500
32768	3.125	90.3	250

Согласно требованиям нас интересует самый скоростной EPP режим. В этом режиме доступна как запись, так и чтение. Действие производится в одном цикле ввода/вывода процессора, это означает, что задержки между этими циклами отсутствуют. При этом скорость обмена данными равна 2 Мбайт/с. Так как управление в этом режиме реализовано аппаратно в контроллере и используется механизм «рукопожатия» при обмене данными, то предполагаем, что есть возможность моделировать схему в режиме реального времени на этой частоте, если адаптер максимально быстро производит запись/считывание и генерирует управляющие сигналы. Однако в спецификации есть уточнение, что максимальный битрейт достигается в режиме отправки пакета из 32 бит, а в режиме обмена по 8 бит максимальный битрейт равен 0.5 Мбайт/с [5].

Был реализован программный модуль чтения/записи в LPT-порт, аппаратный блок обработки данных и согласования уровней 3 и 5 В и проведена оценка производительности. В результате 1 Мбайт можно отправить и столько же принять в среднем за 5.4 с в побайтовом режиме, что соответствует 0.37 Мбайт/с или за одну секунду 185 000 операций чтения и записи одного байта. При использовании этого порта в системе получаем зависимость частоты эмулятора от числа наблюдаемых сигналов, приведенную в табл. 1.

C. USB-порт

Следующий шаг – использование быстродействующего USB-порта, скорость обмена данными по которому согласно спецификации 480 Мбит/с. В качестве адаптера для работы с USB-портом на данный момент существуют решения фирмы FTDI – серия микросхем FTX232X и FTX232H, представляющих собой аппаратный USB-мост [6], работающий в высокоскоростном режиме. Он способен работать в нескольких режимах: UART, асинхронное и синхронное FIFO, MPSSE. Самым производительным режимом этих микросхем является режим синхронного FIFO. Его назначение - передача данных между компьютером (любым другим USB-хостом) и оконечным устройством со скоростью обмена от 8 Мбайт/с до максимальной пропускной способности шины USB. Максимальная скорость обмена достигается в режиме Bulk (режим обмена большими пакетами данных).

Работа микросхемы в режиме синхронного FIFO осуществляется под управлением драйвера D2xx в установленном для канала А режиме FIFO, поставленного в открытом доступе на сайте фирмы FTDI [7]. Для реализации рассматриваемого варианта системы аппаратно-программного моделирования был выбран модуль USB to Serial + (UART/I2C/SPI/JTAG) Adapter на базе микросхемы FT232H. Согласно временным диаграммам работы с FT232H был разработан модуль-прошивка для FPGA, отвечающий за обмен данными в режиме FIFO (рис. 2), а также программный модуль отправки и приема данных.

При программной отсылке или приеме пакетов, согласно временным диаграммам, снятым осциллографом на стороне FIFO, между пакетами данных будет пауза 240 мкс. В пакетном режиме передача данных происходит побайтно на частоте 60 МГц. Пакет на стороне FIFO разделен на блоки по 512 байт, между которыми пауза 10 мкс, а максимальный размер пакета 64 кбайт. Т.е., если программно отправлять пакеты размером меньше чем 64 кбайт, то между ними будет добавляться пауза 240 мкс, что будет снижать производительность шины. Был разработан программный и аппаратный модуль отправки. В режиме только приёма данных или отправки была получена следующая скорость работы: 10000 циклов отправки по 64 кбайт выполняется за 31 с, что эквивалентно 20.64 Мбайт/с. Это реальная предельная скорость при использовании FT232H контроллера в режиме только отправки или приема.

Однако в режиме запрос/ответ скорость значительно снижается: 10000 циклов отправка/прием независимо от числа посылаемых данных в диапазоне от 1 до 16 кбайт занимает примерно 20 с, это значит, что длительность цикла запись/чтение равна 2 мс. Описанный эффект связан с особенностью работы USB-порта: все действия инициируются хостом. В нашем случае действие инициирует компьютер. И согласно спецификации, минимальное время между командами хоста на переключение с режима запись на чтение и

обратно равно 1 мс. В результате получаем зависимость частоты работы эмулятора от числа наблюдаемых сигналов, приведенную в табл. 1.

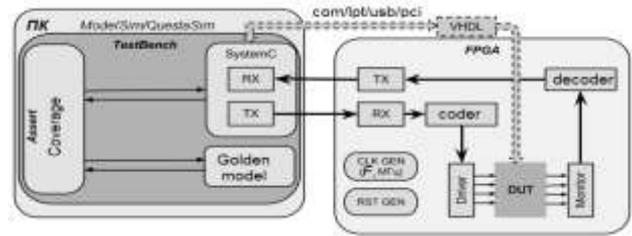


Рис. 2. Структура системы моделирования при использовании FPGA-прототипа

V. ЗАКЛЮЧЕНИЕ

Созданы VHDL модули для прошивки в FPGA, обеспечивающие прием данных и передачу ответов в (из) компьютера через COM/LPT/USB-порты. Разработаны программные модули на SystemC/C++ для отправки приема данных при моделировании. Для компиляции SystemC модуля использовался gcc-4.2.1. Разработанные функции модуля для COM/LPT портов работают в среде QuestaSim 10.1 в ОС Windows, для USB-порта в QuestaSim 10.1 в ОС Linux (Suse).

В ходе экспериментов были выявлены особенности работы портов в режимах приема и отправки данных. Произведена оценка производительности портов при использовании их в системе аппаратного ускорения моделирования. На основе оценок производительности портов и оценки скорости выполнения моделирования на компьютере, используя модули приема/передачи, можно реализовать бюджетный вариант системы ускорения цифрового моделирования синхронных схем.

ЛИТЕРАТУРА

- [1] Emulation Dynamic Debug with HVD Technology [Electronic resource]. – Mode of access: http://www.aldec.com/en/solutions/hardware_emulation_solutions/hvd_technology. – Date of access : 01.08.2012.
- [2] Synopsys [Electronic resource]. – Mode of access: <http://www.synopsys.com/SYSTEMS/FPGABASEDPROTOTYPING/Pages/HAPS.aspx> – Date of access : 20.09.2013.
- [3] Black D.C and Donovan J. Systemc: from the ground up. Kluwer Academic Publishers, Boston, 2004.
- [4] Зайцев В.С. Аппаратное ускорение цифрового моделирования / В. С. Зайцев, В. Я. Степанец // Материалы конференции ITS2012. Минск. 2012. С. 206-207.
- [5] IEEE 1284-1994 Standard [Electronic resource]. – Mode of access: <http://www.fapo.com/1284int.htm>– Date of access : 10.01.2014.
- [6] Работа аппаратного USB-моста FTDI FT2232H в режиме синхронного FIFO [Electronic resource]. – Mode of access: http://kit-e.ru/assets/files/pdf/2010_08_90.pdf. – Date of access : 10.09.2013.
- [7] Software Application Development D2XX Programmer's Guide. [Electronic resource]. – Mode of access: <http://www.ftdichip.com/FTDrivers.htm> – Date of access : 10.09.2013