

Конвертация проектов FPGA семейства Spartan 3 в заказные СБИС

Пётр Бибило, Данила Городецкий, Дмитрий Новиков,
Артём Соловьев, Дмитрий Черемисинов (Минск, Белоруссия)

В статье описаны программные средства конвертации проектов цифровых систем, реализованных на микросхемах FPGA семейства Spartan 3, в VHDL-описания, пригодные для синтеза в библиотеках проектирования заказных СБИС.

ВВЕДЕНИЕ

Программируемые логические интегральные схемы (ПЛИС) типа FPGA фирмы Xilinx широко применяются в цифровой аппаратуре и позволяют быстро оценить сложность проекта, его быстродействие и получить экспериментальный образец действующей аппаратуры. Однако для специальных применений требуется использование только отечественной элементной базы. Перевод проекта, реализованного на FPGA, в описание, пригодное для реализации на отечественной элементной базе, включая разработку автоматизированной программы конвертации, представляет собой трудоёмкую и наукоёмкую задачу.

В предлагаемой статье рассмотрен процесс получения синтезируемого VHDL-описания проекта, реализованного на FPGA фирмы Xilinx семейства Spartan 3 [1] в системе проектирования WebPack ISE 13.1 [2]. В качестве базовой микросхемы для экспериментов была выбрана ИС FPGA xc3s1000-4ft256. Реализованный на FPGA проект может иметь смешанные исходные спецификации: некоторые блоки могут быть представлены на высокоуровневых языках VHDL и Verilog, другие – логическими схемами в графическом редакторе системы ISE либо графическими диаграммами в подсистеме StateCAD [3]. Кроме того, возможны «ручные» изменения проекта в топологическом редакторе (FPGA Editor) системы ISE.

Наличие всего проекта в виде VHDL-описания, пригодного для автоматического синтеза в целевых библиотеках отечественных заказных СБИС, позволяет провести реализацию проекта на заказной СБИС и упрощает перепроектирование за счёт «однородности» описания данных. Следует отметить, что FPGA семейства Spartan 3

имеют в своём составе макроэлементы ОЗУ и умножителей, однако процессы их конвертации (и перепроектирования) для реализации в составе заказных СБИС заслуживают отдельного рассмотрения. При конвертации блоки ОЗУ, используемые как ПЗУ, извлекаются вместе с записанными в них данными, однако далее такие ОЗУ (либо ПЗУ) выделяются из части проекта, реализованной на конфигурируемых логических блоках. С умножителями можно поступать аналогичным образом либо переходить к их синтезируемому алгоритмическому VHDL-описанию.

Основное внимание при конвертации проектов FPGA уделяется получению синтезируемого VHDL-описания той части проекта, которая является совокупностью взаимосвязанных конфигурируемых логических блоков.

АРХИТЕКТУРА FPGA СЕМЕЙСТВА SPARTAN 3

Основными блоками рассматриваемой микросхемы FPGA семейства Spartan 3 являются блоки ввода-вывода (Input-Output Block, IOB), конфигурируемые логические блоки (configurable logical block, CLB), блочная память (Block RAM, BRAM) и 18-разрядные умножители (Multiplier) [1].

Основной структурной единицей FPGA является блок CLB. Он состоит из двух секций «левой руки» (SliceM, где M – от слова Memory) и двух секций «правой руки» (SliceL, где L – от слова Logic), входы и выходы которых связаны с матрицей коммутации соединений (Switch Matrix), реализующей глобальные связи. На рисунке 1 изображены структурные схемы секций SliceL и SliceM. Для получения схемы секции SliceM штриховые линии следует заменить непрерывными. Если же на рисунке 1 удалить те элементы и соедине-

ния, которые изображены штриховыми линиями, получится структурная схема секции SliceL.

Секция SliceL является более простой [4] и состоит из двух LUT (Look-Up Table), двух программируемых элементов памяти (FFY, FFX), нескольких логических элементов И (AND), ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR) и мультиплексоров с одним управляющим входом (MUX2). Связи структурных элементов являются фиксированными, однако их можно задать с помощью т.н. программируемых соединений, что позволяет реализовать в одной секции SliceL разнообразные логические схемы, как комбинационные, так и схемы с памятью. Точки коммутации соединений (точки программирования) показаны на входах элементов треугольниками.

Программируемый элемент LUT является универсальным логическим элементом и может реализовать любую булеву функцию не более чем от четырёх переменных. Элементы памяти могут функционировать как FF (Flip-Flop – синхронизируемый фронт-триггер) либо как Latch (триггер с потенциальным управлением – защёлка). Таким образом, в секции SliceL можно реализовать схему с памятью – конечный автомат. В статье [5] рассматриваются функциональные модели триггеров, входящих в состав CLB семейства Spartan 3.

Секция SliceM (см. рис. 1) сложнее в части реализации LUT: программируемые элементы, кроме реализации логических функций, могут выполнять функции блока RAM либо функции сдвигового регистра. Один элемент LUT может выполнять функции 16-разрядного сдвигового регистра, два элемента могут быть соединены в один 32-разрядный сдвиговый регистр, три – в 48-разрядный, и т.д. С этой целью вводится вспомогательный элемент WSGEN, осуществляющий синхронизацию сдвигов, и другие вспомогательные элементы и соединения.

ПРОГРАММА КОНВЕРТАЦИИ

Исходными данными для программы конвертации являются представле-

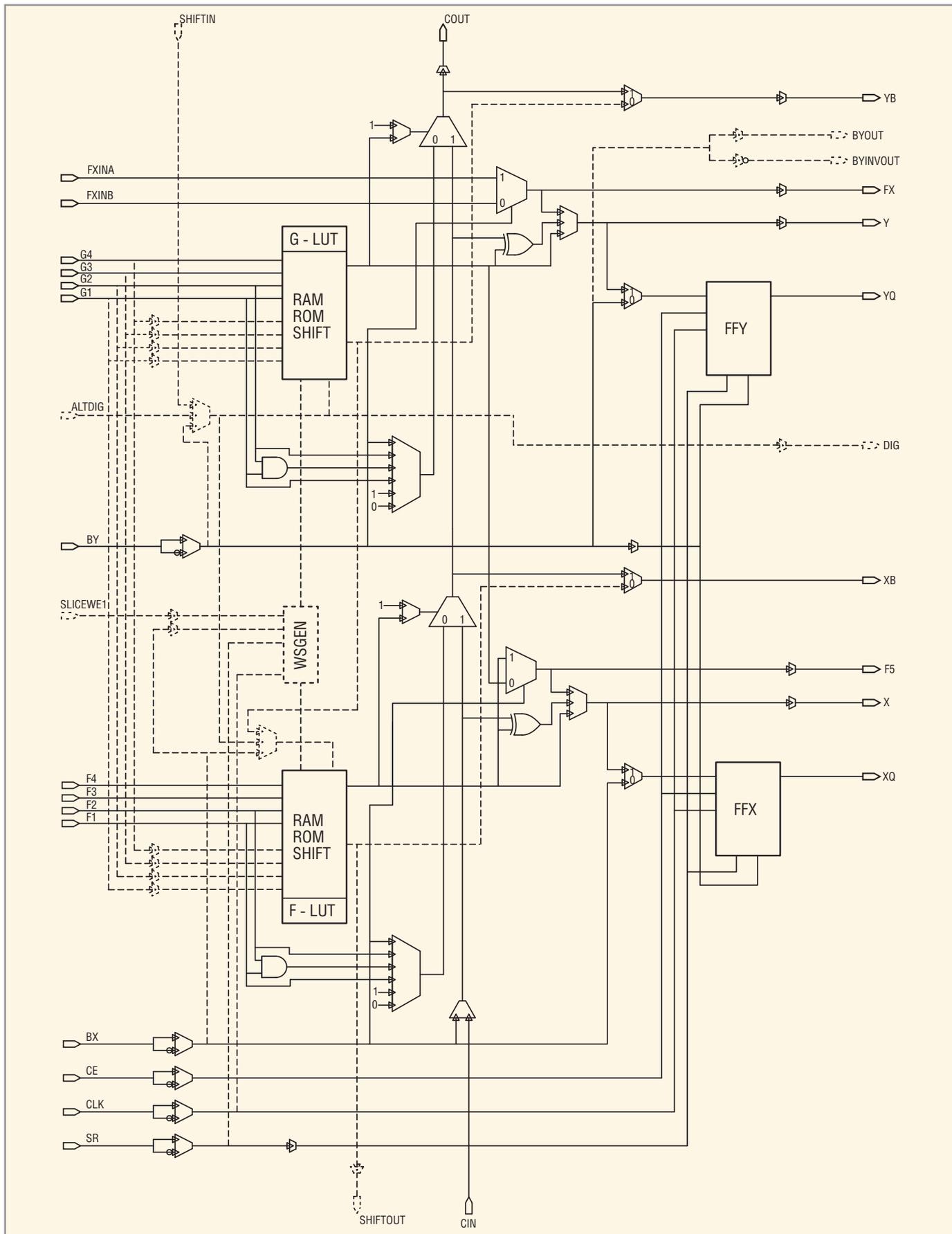


Рис. 1. Структурные схемы секций SliceM и SliceL

ния FPGA-проектов в формате .ncd (Native Description Language – формат представления проекта после размещения элементов FPGA и разводки со-

единений), результирующими данными – синтезируемые RTL-описания проектов на VHDL. Для получения VHDL-описания схемы в библиотеке

проектирования заказной СБИС используется синтезатор LeonardoSpectrum (фирма Mentor Graphics). Рисунок 2 иллюстрирует этап работы про-

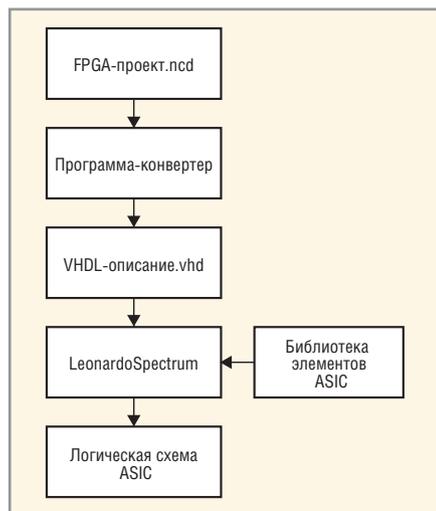


Рис. 2. Маршрут перепроектирования

граммы конвертации в процессе перепроектирования.

Представим общую характеристику проектов, которые могут быть конвертированы в синтезируемые VHDL-описания, и ограничения процесса синтеза проектов FPGA:

- головное описание FPGA-проекта должно быть написано на VHDL, входные и выходные порты должны быть заданы типами `std_logic` и `std_logic_vector`;
- при реализации на FPGA исходного проекта использование функциональных элементов в блоках ввода-вывода (*Xilinx Specific Options* → *Pack I/O Registers into IOBs*) должно быть запрещено (No);
- VHDL-модели макроэлементов DCM (Digital Clock Manager) и макроэлементов блочной памяти, входящих в проект FPGA и используемых в системных библиотеках ISE, остаются после конвертации без изменений, т.е. не преобразуются в VHDL-модели, пригодные для синтеза в целевой библиотеке пользователя;
- конвертированные VHDL-описания при моделировании имеют нулевые задержки, поскольку все функциональные модели структурных эле-

ментов CLB используют только нулевые задержки [4]. Поэтому функциональная эквивалентность реализованного на FPGA проекта и конвертированного VHDL-проекта обеспечивается по окончаниям временных интервалов соответствующих тактов моделирования.

Программа конвертации продолжает совершенствоваться с целью расширения её функциональных возможностей. Очередными задачами являются:

- обработка портов типа inout;
- обработка проектов FPGA, использующих ресурсы блоков ввода-вывода;
- расширение функций программы на другие семейства FPGA.

ИССЛЕДОВАНИЕ ПРОГРАММЫ КОНВЕРТАЦИИ

Разработанная программа-конвертер прошла экспериментальную проверку на тридцати проектах FPGA. Исходными данными для реализации на микросхеме xc3s1000-4ft256 FPGA семейства Spartan 3 были алгоритмические VHDL-описания, в том числе и доступные по сети Интернет, логические схемы в графическом редакторе системы ISE либо описания в State-CAD, а также генерируемые IP-блоки системы ISE.

Проверка правильности работы программы конвертации осуществлялась путём сравнения результатов поведения исходных проектов и конвертированных VHDL-описаний. Исходные проекты моделировались как в системе ISE, так и отдельно в системе ModelSim [3]. Моделирование конвертированных описаний осуществлялось только в системе ModelSim без использования системных библиотек ISE. В качестве синтезатора логических схем заказных СБИС (ASIC) использовалась программа LeonardoSpectrum [3], к которой можно подключать различные целевые библиотеки пользователя, например, [3, с. 342], которая содержит

35 логических элементов. Для оценки сложности схемы СБИС использовались два параметра – число элементов схемы и их суммарная площадь. Значения этих параметров выдаёт программа LeonardoSpectrum после выполнения этапа синтеза.

Результаты синтеза восьми проектов приведены в таблице 1. Первые четыре проекта – это IP-блоки, получаемые в системе ISE с помощью программы Core Generator. Описание пятого проекта [6] доступно в сети Интернет. Шестой и седьмой проект являются отладочными. Восьмой проект представляет собой реализацию на распределённой логике (CLB) постоянного запоминающего устройства (ПЗУ), длина адресного слова которого равна девяти битам, а хранимого слова – 128 битам.

При повторной реализации восьмого конвертированного проекта на той же микросхеме FPGA сложность схемы возросла с 4275 до 6012 LUT (см. таблицу 1); это является свидетельством того, что ПЗУ лучше реализовать на блочной памяти, а не на распределённой логике. Для всех проектов было достигнуто одинаковое поведение (по окончаниям временных интервалов соответствующих тактов моделирования) исходного проекта и конвертированного. Кроме того, функциональная эквивалентность проектов 5 – 8 была подтверждена в системе FormalPro формальной верификации фирмы Mentor Graphics [7].

Для выбора оптимального маршрута перепроектирования был проведён эксперимент по реализации на заказной СБИС синтезируемых VHDL-описаний проектов 5 – 8 двумя способами:

- реализация исходных VHDL-описаний;
- реализация VHDL-описаний, полученных с помощью программы конвертации.

Результаты эксперимента представлены в таблице 2, где суммарная пло-

Таблица 1. Результаты синтеза проектов

№ проекта	Назначение проекта	Форма исходного задания	Число входо-выходов	Схема FPGA		Схема заказной СБИС	
				LUT	триггеров	число элементов	суммарная площадь элементов
1	Извлечение квадратного корня	IP-блок (ISE)	14	26	22	59	186
2	Умножитель	IP-блок (ISE)	41	114	174	661	2405
3	Вычисление синуса и косинуса (алгоритм CORDIC)	IP-блок (ISE)	49	1023	1018	3648	16266
4	Устройство деления	IP-блок (ISE)	66	97	220	291	872
5	VGA-контроллер	VHDL	67	192	76	551	2277
6	Устройство деления	VHDL	82	545	82	1440	7373
7	Два параллельных 12-разрядных умножителя и сумматора	VHDL	110	650	0	1507	7236
8	ПЗУ (9, 128)	VHDL	139	4275	128	12934	53790

Таблица 2. Результаты эксперимента

№ проекта	Назначение проекта	Форма исходного задания	Число входов-выходов	Схема заказной СБИС			
				синтез исходного проекта		синтез конвертированного проекта	
				число элементов	суммарная площадь элементов	число элементов	суммарная площадь элементов
5	VGA-контроллер	VHDL	67	523	2059	551	2277
6	Устройство деления	VHDL	82	1302	7245	1440	7373
7	Два 12-разрядных умножителя и сумматора	VHDL	110	1351	6746	1507	7236
8	ПЗУ (9, 128)	VHDL	139	13648	57974	12934	53790

щадь элементов схемы заказной СБИС измерена в условных единицах [3].

ЗАКЛЮЧЕНИЕ

Если спецификации исходного проекта заданы на VHDL и в составе проекта имеются IP-блоки, то целесообразно с помощью программы конвертации отдельно получить синтезируемые VHDL-описания этих IP-блоков и провести синтез схемы заказной СБИС от исходных описаний с учётом иерархии проекта.

Если же для реализации на заказной СБИС взять конвертированное описание всего проекта в целом, то результаты синтеза могут быть хуже, поскольку в конвертированном проекте все составляющие его подсхемы распределяются по CLB и описание становится

не иерархическим. Однако может быть и иначе, о чём свидетельствует реализация конвертированного описания восьмого проекта.

Элементы целевой библиотеки заказной СБИС имеют свои временные задержки, отличные от задержек элементов FPGA, поэтому длительности тактов сигналов синхронизации для FPGA и для схемы заказной СБИС не будут совпадать. В значительной мере это обусловлено различием технологической базы FPGA и СБИС. Конвертация может обеспечивать функциональную эквивалентность проектов по тактам, но чтобы добиться полного совпадения временных задержек микросхемы FPGA и заказной СБИС, потребуется переход на другую элементную базу или другие проектные нормы.

ЛИТЕРАТУРА

1. Кузелин О.М., Кнышев Д.А., Зотов Ю.В. Современные семейства ПЛИС фирмы Xilinx: Справочное пособие. Горячая линия – Телеком, 2004.
2. Зотов Ю.В. Проектирование цифровых устройств на основе ПЛИС фирмы Xilinx в САПР WebPack ISE. Горячая линия – Телеком, 2003.
3. Библио П.Н. Системы проектирования интегральных схем на основе языка VHDL: StateCAD, ModelSim, LeonardoSpectrum. СОЛОН-Пресс, 2005.
4. <http://xilinx.com>.
5. Библио П.Н., Соловьев А.Л. Функциональные модели триггеров и их реализация в FPGA. Современная электроника. 2012. № 1.
6. <http://opencores.org>.
7. Лохов А. Обзор средств функциональной верификации компании Mentor Graphics. Современная электроника. 2005. № 5. ©