

Е.К.Досумбеков

ПЛИС XILINX как аппаратная база встраиваемых систем

(Евразийский национальный университет им. Л.Н. Гумилева, г. Астана)

В статье рассматриваются особенности и границы применимости ПЛИС при проектировании встраиваемых систем, показаны преимущества и недостатки использования этой аппаратной базы, а также описаны факторы, делающие ПЛИС привлекательными для разработчиков.

Основной материал: С момента своего появления программируемые логические интегральные схемы (ПЛИС) прошли достаточно долгий путь от экзотических и дорогих цифровых устройств, предназначенных, прежде всего для макетирования, до отдельного направления в современной микроэлектронике, делающего основной упор на гибкости аппаратной базы и малом времени разработки цифровой части устройства. Существующие тенденции в мировой электронике демонстрируют, с одной стороны, снижение стоимости микросхем для потребителя, однако с другой — постоянное повышение стоимости разработки самих микросхем. Это обуславливает необходимость высоких тиражей для разработчиков собственно микросхем, и освобождающаяся ниша изделий со средними тиражами успешно заполняется программируемой логикой. По оценкам производителей ПЛИС, граница, когда разработка уникальной микросхемы будет выгоднее, чем конфигурирование ПЛИС, поднялась за последние несколько лет с примерно 50 тыс. изделий до 200—300 тыс. В освободившуюся нишу попали, в частности, контроллеры ЖК- и плазменных панелей, медиацентры, автомобильная электроника, ряд систем промышленной автоматизации, системы проводной и беспроводной связи. Требования к упомянутым системам изменяются достаточно динамично, что повышает риски для разработчиков специализированных микросхем. В итоге, несмотря на то, что аналогичная по функциям ПЛИС оказывается несколько медленнее и дороже, специализированные микросхемы для решения каких-то задач могут просто не появиться из-за риска получить морально устаревшее изделие [1, 344].

Немаловажным фактором, делающим ПЛИС привлекательными для разработчиков, является также наличие в ряде случаев специальных ограничений, накладываемых заказчиком. Например, задачи интеграции периферийного оборудования для системы промышленной автоматизации могут вывести на первый план не стоимость комплектующих (система может быть заказана и в единичном экземпляре), а возможность гибкой реакции на изменяющиеся требования заказчика и выявляемые в процессе разработки особенности объектов, подлежащих анализу и управлению. В данном случае разработчик, вероятнее всего, предпочел бы аппаратную платформу с возможностью реконфигурирования и некоторым запасом возможностей, а не минимальную стоимость электронных компонентов.

В настоящее время фирмой Xilinx, которая занимает лидирующие позиции по объему продаж ПЛИС, выпускаются несколько семейств таких

устройств. К устройствам начального уровня с архитектурой CPLD относится серия CoolRunner-II, отличающаяся пониженным (для ПЛИС) энергопотреблением. К «магистральному» направлению, с архитектурой FPGA, относятся серия Virtex, представленная наиболее актуальными семействами Virtex-4 и Virtex-5, и серия недорогих микросхем Spartan-3 с несколькими разновидностями. Назначением серии Spartan было кардинальное снижение цены ПЛИС с сохранением основных функциональных возможностей, и на протяжении эволюции этой серии цена действительно постоянно падала вплоть до величины в 2 доллара за 100 тыс. логических вентилях (в крупных партиях) [2, 200].

Разработка систем на кристалле на базе ПЛИС с помощью САПР Embedded Development Kit. С момента снижения цен на ПЛИС до приемлемого уровня естественным образом появилась своеобразная конкуренция технических решений. Фактически, будучи высокоинтегрированными (хотя бы за счет возможности размещения на кристалле различных модулей) цифровыми устройствами, ПЛИС с архитектурой FPGA с технической точки зрения стали альтернативой микроконтроллерам во встраиваемых системах. Естественным, полного вытеснения микроконтроллерных решений не произошло по совершенно объективным причинам, и соотношение объемов продаж тех и других продуктов наглядно показывает, в каком проценте случаев ПЛИС действительно оказываются привлекательнее. Но даже имеющееся на сегодняшний день проникновение ПЛИС на рынок встроенных систем вряд ли произошло бы без появления САПР, позволяющих превратить ПЛИС по сути дела в микроконтроллерную систему с очень гибко программируемым набором периферийных устройств. Действительно, одним из важнейших препятствий при рассмотрении применимости ПЛИС становится необходимость проведения разработки на одном из языков описания аппаратуры (VHDL/Verilog). Сложность такой работы существенно выше, чем сложность работы в интегрированной среде для микроконтроллера, с использованием языков высокого уровня и готовым набором системных библиотек. Более того, модификацию и сопровождение проекта на ПЛИС тоже приходилось бы выполнять на VHDL/Verilog, в том числе и реализацию небольших, «косметических» изменений алгоритмов, протоколов обмена и интерфейсных функций.

Естественно, недостаток ПЛИС в качестве аппаратной базы для решения задач, в которых традиционно сильны микроконтроллеры, прекрасно осознавался производителями. Для фирмы Xilinx итогом стала разработка продукта Embedded Development Kit (EDK), который позволял выполнять визуальное конструирование системы на базе микропроцессора. В качестве такого микропроцессора выступило разработанное Xilinx ядро Microblaze, представляющее собой так называемый софт-процессор, т.е. процессор, выполненный в виде особым образом сконфигурированных ячеек ПЛИС. В итоге Microblaze стало возможным разместить в любой ПЛИС подходящего объема (а на практике — в любой современной микросхеме FPGA Xilinx), и

превратить ее с точки зрения разработчика системы в своеобразный процессор [3, 298].

Embedded Development Kit включает в себя два основных программных приложения — САПР Xilinx Platform Studio (XPS), предназначенную в основном для конфигурирования аппаратной части системы, в том числе подключения периферийных устройств, и Xilinx Platform Studio SDK, которая представляет собой широко распространенную среду разработки Eclipse, адаптированную для создания программ для Microblaze. Внешний вид интерфейса САПР XPS показан на рисунок 1.

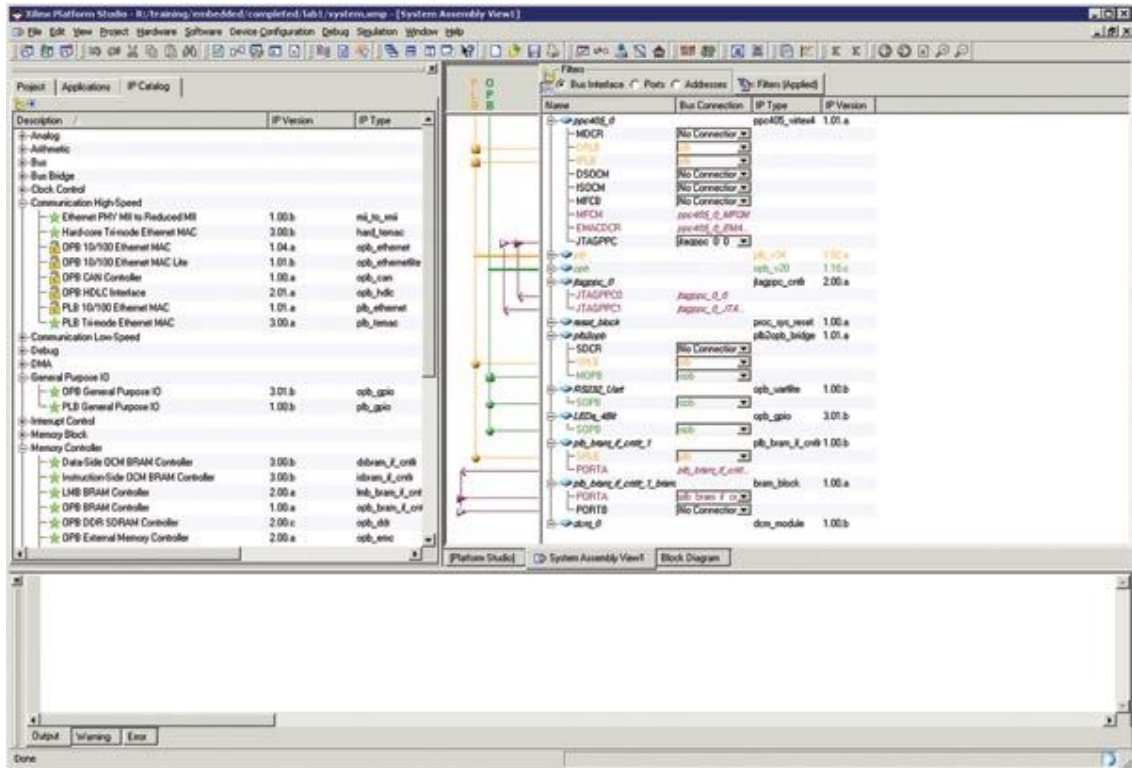


Рисунок 1 - Внешний вид интерфейса САПР Xilinx Platform Studio

Первоначальное конфигурирование системы удобно выполнять с помощью мастера, который позволяет буквально в течение 5 минут настроить желаемые характеристики и выбрать периферийное оборудование. В составе библиотеки оборудования находятся около 200 IP-ядер, реализующих контроллеры широко распространенных периферийных устройств (порты ввода-вывода, контроллеры памяти различных типов и др.). САПР также создает примеры программных приложений и драйверы для выбранного оборудования. Можно отметить, что Microblaze создавался как недорогая альтернатива аппаратному ядру PowerPC, которое присутствует в high-end FPGA Xilinx серий Virtex-II-Pro, Virtex-4FX и Virtex-5FXT. Соответственно, схож и состав программных средств (по умолчанию для компиляции используется gcc). Последняя версия ядра Microblaze допускает использование менеджера памяти (MMU — Memory Management Unit), что

дает возможность работать под управлением ОС Linux. Это очевидным образом позиционирует системы на базе FPGA с процессором Microblaze (возможна, разумеется, и работа без операционной системы) [4, 252].

Рассмотрим рисунок 2, на котором показан примерный состав системы на базе FPGA.

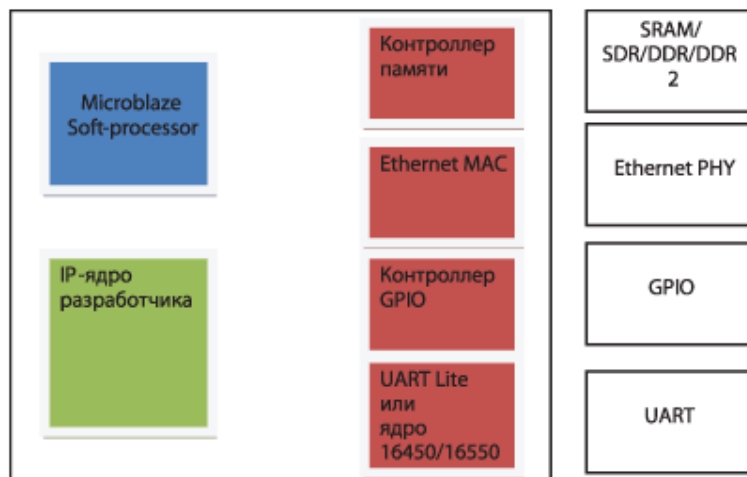


Рисунок 2 - Пример системы, разработанной с применением САПР EDK

В рамках такой системы можно реализовать в целом стандартный набор периферийных устройств, однако внимание можно обратить на блок, обозначенный как «IP-ядро разработчика». В EDK имеется штатная возможность подключать к процессору компоненты, описываемые разработчиком на VHDL или Verilog с автоматическим подключением их к процессорной шине (включая как реализацию программно-доступных регистров, так и прерываний, ПДП, возможности захвата системной шины). Таким образом, даже ранее написанные разработчиком модули могут быть сделаны программно-доступными со стороны Microblaze. Несмотря на возможно небольшой удельный вес такого блока в составе системы, именно он может стать тем ключевым элементом, который и обусловит выбор ПЛИС в качестве аппаратной базы. В данном случае речь идет не о каких-то специфических свойствах ПЛИС как готового устройства, а исключительно о возможности реализовать на ее базе know-how разработчика системы, который при позиционировании своего продукта на рынке делает основной упор на эксклюзивных возможностях, достигаемых благодаря разработанной им цифровой схеме. При этом разработчик, выбравший в качестве аппаратной базы стандартный микроконтроллер, не может полагаться на уникальность аппаратного состава своей системы (поскольку любой его конкурент легко может приобрести точно такой же микроконтроллер), и должен достигать более высоких потребительских свойств изделия другими способами — например, более эффективной программной обработкой, низкой ценой, высоким качеством, удачным дизайном и т.д. В данном случае ПЛИС предоставляет дополнительную степень свободы: возможность

реализовать уникальную аппаратную схему, которая и будет являться ключевым моментом для обеспечения конкурентоспособности продукта на рынке [5, 448].

Можно также добавить, что применение ПЛИС отнюдь не исключает возможность использования в проекте и других решений, эффективных для построения встраиваемых систем. Скорее можно говорить о том, что ПЛИС может привнести в проекты новые свойства, и прежде всего, ликвидировать «бутылочные горлышки», образующиеся в ряде классов изделий.

Системы цифровой обработки сигналов. Важной особенностью современных FPGA является большой удельный вес ресурсов, пригодных для цифровой обработки сигналов. Операция умножения независимых операндов является настолько распространенной, что практически все производители FPGA располагают умножителями в виде аппаратных модулей на кристалле. Иными словами, умножитель в FPGA полностью аналогичен умножителю в сигнальном процессоре или микроконтроллере. В то же время, даже ПЛИС Spartan-3 имеют от 4 до 126 таких блоков, способных работать на тактовой частоте 250...300 МГц. Более дорогие микросхемы серии Virtex содержат умножители со встроенными аккумуляторами — т.н. блоки ExtremeDSP. Аппаратный 48-битный аккумулятор позволяет реализовать операцию «умножение с накоплением», являющуюся основой многих алгоритмов цифровой обработки сигналов, в готовом модуле, имеющем гарантированно высокую производительность, а следовательно, характеристики проекта в FPGA, ориентированного на цифровую обработку сигналов, можно предполагать достаточно высокими.

Часто приводятся оценки, показывающие, что FPGA обладают не только максимальными абсолютными показателями производительности в задачах DSP (оценим, к примеру, 1056 блоков «умножение с накоплением», работающих на частоте 550 МГц в ПЛИС Virtex5-SXT240), но и лучшим соотношением производительности и цены, чем современные сигнальные процессоры, выполненные по аналогичной технологии. Это утверждение становится тем более верным, чем лучше реализуемый алгоритм подлежит распараллеливанию. Действительно, при меньшей тактовой частоте FPGA способны параллельно выполнять десятки и даже сотни операций, что представляет проблему для большинства сигнальных процессоров. Попытка увеличить число каналов обработки данных неминуемо наталкивается на тот факт, что разработчику в действительности могут потребоваться немного отличающиеся операции для различных каналов, поэтому сигнальный процессор с большим числом модулей, работающих строго параллельно, имеет риск оказаться невостребованным. В то же время репрограммируемость FPGA не задает жестких рамок на использование вычислительных ресурсов — блоки DSP можно объединять, перегруппировывать между несколькими потоками данных, распределять между разными процессорами и модулями и т.д.

Можно обратить внимание, что в настоящее время существует большое количество проектов вида «сигнальный процессор + FPGA». В таких

устройствах FPGA обычно выполняет роль мощного параллельного вычислителя, выполняющего основной объем несложных операций, а сигнальный процессор служит аппаратной основой для реализации сложных алгоритмов верхнего уровня — постобработки, управления, мониторинга, интерфейсных функций и т.п. В роли сигнального процессора также легко представить любой микроконтроллер, удобный с точки зрения архитектуры, встроенной периферии и инструментов разработки, но не обладающий достаточной производительностью в «поточных» задачах.

Можно рассмотреть также и рекомендуемый маршрут проектирования для устройства класса «процессор + FPGA». В данной связке FPGA выступает как сопроцессор, ускоряющий работу основного процессора. Соответственно, основное внимание разработчика по доводке характеристик, алгоритмов и функциональности будет уделяться именно процессору и программам для него, а не FPGA. Поэтому было бы желательно как можно скорее создать требуемую конфигурацию сопроцессора и переключиться на основную задачу, уже имея в своем распоряжении запрограммированный ускоритель вычислений. Для решения типовых задач цифровой обработки сигналов имеется пакет Xilinx System Generator for DSP, интегрируемый с широко известным математическим пакетом Matlab. Внешний вид такой программы показан на рисунок 3.

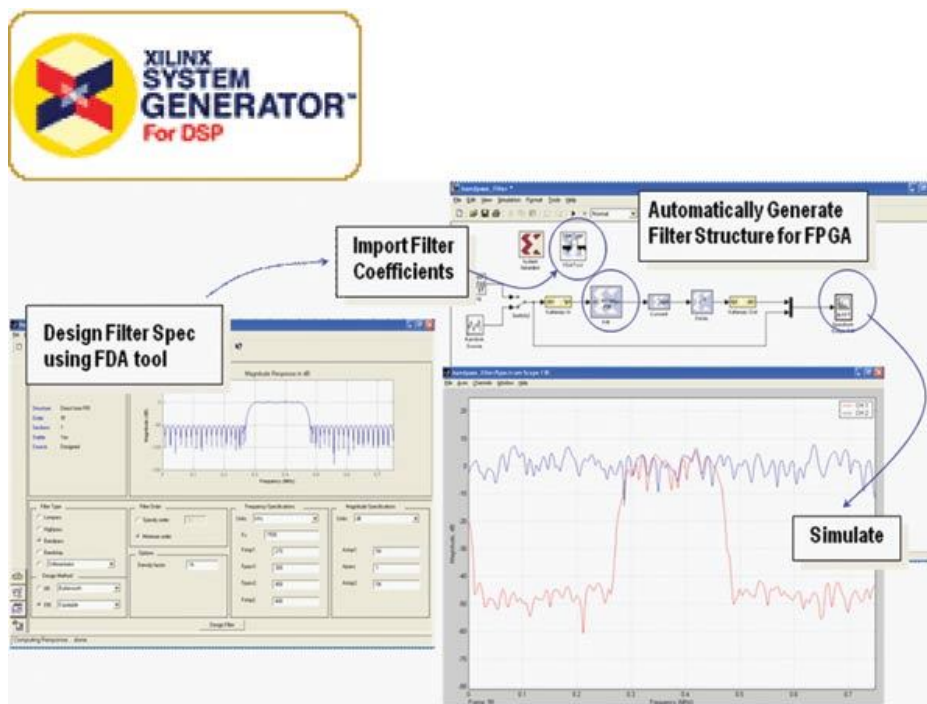


Рисунок 3 - Внешний вид интерфейса System Generator for DSP

Как можно видеть, разработка проекта на базе ПЛИС ведется в системе Simulink. На рабочем листе можно размещать источники сигналов, устанавливать точки наблюдения, а обработку выполняют библиотечные

элементы, имеющие однозначное соответствие с готовыми модулями, реализуемыми в FPGA. Актуальные задачи, такие как разработка цифровых фильтров, имеют готовое решение с удобным графическим интерфейсом (как и показано на рис. 3). Фильтр с заданной характеристикой можно «в один клик» перенести в FPGA, промоделировать поведение созданного устройства в Matlab, а затем проверить функционирование созданной системы в реальной микросхеме, наблюдая за поведением сигналов через штатный интерфейс JTAG. Таким образом, в ПЛИС можно несложным путем передать реализацию массовых параллельных вычислений, сосредоточив внимание на верхних уровнях решаемой задачи — алгоритмах постобработки, интерпретации данных, диагностике и пр.

В качестве примера можно рассмотреть систему на базе единственной FPGA, в которой софт-процессор Microblaze дополнен 8...16 параллельно работающими каналами «умножение с накоплением». Это решение относится к системам начального уровня с точки зрения требуемой логической емкости ПЛИС, но можно сразу отметить, что производительность составит 2...3 млрд. операций «умножение с накоплением» в секунду, что существенно больше, чем у большинства процессоров для встраиваемых систем. При этом проект не обязан сопровождаться на VHDL/Verilog, поскольку наиболее часто корректируемые алгоритмы выполняются софт-процессором Microblaze, который может программироваться на Си. Таким образом, появляется возможность создания устройств с достаточно высокой производительностью без необходимости привлечения высококвалифицированных проектировщиков цифровых устройств для всего цикла проектирования. С точки зрения коллектива разработчиков, речь идет о разработке обычной процессорной системы и ее программировании на Си, однако оказывается, что подключенные к этому процессору периферийные устройства представляют собой весьма мощные сопроцессоры.

Немаловажной чертой является самостоятельность создаваемых в ПЛИС подсистем. Если для разработки системы реального времени на базе процессора с одним потоком команд большое внимание следует уделять алгоритмам разделения времени между задачами, то при наличии выделенного аппаратного ресурса проблема обеспечения своевременной реакции процессора может быть существенно сглажена. Например, однократно настроенные и запущенные цифровые фильтры на базе ПЛИС не требуют постоянного вмешательства центрального процессора в свою работу, что позволяет программисту использовать его для других целей. Критичные источники прерываний могут обрабатываться собственным ядром Microblaze, часть аварийных ситуаций может обрабатываться выделенными аппаратными модулями и т.д. В конечном итоге возможно реализовать комплекс схемотехнических и архитектурных мероприятий, которые помогут скомпенсировать несколько более высокую стоимость ПЛИС качественной стыковкой узкоспециализированных подсистем,

которые в итоге обеспечат высокие потребительские свойства разработанного продукта [6, 320].

Средства разработки. Для ПЛИС существуют достаточно удобные средства разработки, облегчающие быстрый старт для разработчиков, ранее не использовавших подобную технологию. Плата на базе ПЛИС Spartan-3E недорогая оценочная плата, предназначенная для быстрого освоения маршрута разработки цифровых систем на базе FPGA.

Приобретение такой платы (а точнее, комплекта разработчика) дает в распоряжение готовый комплекс программных и аппаратных средств, необходимых для проведения полного цикла разработки. Уже достаточно давно существует бесплатная САПР ISE WebPack, функциональные характеристики которой полностью аналогичны коммерческой версии (ограничения сохранены только в части максимального логического объема ПЛИС, планка которого установлена на довольно высоком уровне).

Выводы. Программирование платы осуществляется встроенным в плату программатором. Плата поддерживается также САПР EDK, которая предлагает реализовать интерфейсы для всего установленного на ней оборудования (включая DDR SDRAM, Ethernet, UART, АЦП и ЦАП). Наборы отладочных плат выпускаются фирмой Xilinx для всего спектра своей продукции, и практика показывает, что с учетом высочайшей степени реконфигурируемости сферы применения таких плат и решаемые с их помощью задачи оказываются весьма разнообразными.

ЛИТЕРАТУРЫ

1. Суворова Е.А., Шейнин Ю.Е. Проектирование цифровых систем на VHDL. – СПб.: БХВ-Петербург, 2003. – 344 с..
2. Бибило П.Н. Основы языка VHDL. – М.: «Солон-Р», 2000. –200 с.
3. Зотов В.Ю. Проектирование цифровых устройств на основе ПЛИС фирмы XILINX в САПР WebPACK ISE. – М.: Горячая линия-Телеком, 2003. – 298 с.
4. Тарасов И.Е. Разработка цифровых устройств на основе ПЛИС XILINX® с применением языка VHDL. М.: Горячая линия-Телеком, 2005. – 252 с.
5. Перельройзен Е.З. Проектируем на VHDL. - М.: Изд-во "СОЛОН-Пресс", 2004. - 448 с.
6. Поляков А. К. Языки VHDL и VERILOG в проектировании цифровой аппаратуры. - М.: Изд-во "СОЛОН-Пресс", 2003. - 320 с.

Досымбеков Ерлан Қадырұлы

Орналастырылған жүйелердің аппараттық негізі ретіндегі БЛИС XILINX

Анатпа. Мақалада орналастырылатын жүйелерді жобалау кезінде БЛИС –тің қолданудың жиектері мен ерекшеліктері қарастырылады, осы аппараттық базаны қолданудағы кемшіліктері мен артықшылықтары көрсетілген, сонымен қатар

құрастырушыларға БЛИС-ті қызықты болып әкелуіне байланысты факторлары сипатталған.

Dossumbekov Yerlan Kadyrovich

FPGA XILINX as, apparatus basis of built systems.

Annotation. The article describes the features and limits of using FPGA in designing built systems. There were shown advantages and disadvantages of using this equipment database, and also there were described the factors which make FPGA amazing for developers.