

Программные средства разработки систем ЦОС на основе ПЛИС Xilinx

Павел Львов, Александр Тумачек (Москва)

В статье приведён обзор программных средств проектирования систем цифровой обработки сигналов на основе ПЛИС.

ВВЕДЕНИЕ

Цифровая обработка сигналов (ЦОС) применяется в любой аппаратуре, где информация поступает в цифровом виде или управляется цифровым устройством. Проектирование систем ЦОС на основе ПЛИС – сложный процесс, требующий от разработчика определённых знаний и навыков, поэтому развитие методов автоматизированного проектирования данного класса устройств является актуальной задачей.

В данной статье описывается, как объединить набор программных средств проектирования Active-HDL и Xilinx ISE компаний Aldec и Xilinx соответственно, с целью оптимизации процесса разработки устройства на основе ПЛИС. Такое решение доказало свою эффективность при проектировании модулей демультимплексирования цифрового потока и цифровой фильтрации сигналов.

ПРОГРАММНЫЕ СРЕДСТВА, ПРИМЕНЯЕМЫЕ ДЛЯ ПРОЕКТИРОВАНИЯ ПЛИС

В данном разделе приводится обзор программных средств, необходимых для проектирования цифрового устройства на основе ПЛИС. Набор не является стандартным для ПЛИС фирмы Xilinx, поскольку в него входят программные продукты сторонних производителей. Каждый программный продукт отвечает за отдельный этап проектирования:

- 1) создание проекта разрабатываемого устройства и его моделирование в визуальной среде;
- 2) выбор и настройка блоков (оптимизированных IP-ядер), предоставленных фирмой-разработчиком;
- 3) создание описания устройства на языке проектирования VHDL, объединение модулей, тестирование методом временных диаграмм;

- 4) синтез проекта;
- 5) размещение и трассировка узлов проекта;
- 6) конфигурация ПЛИС, отладка методом сквозного проектирования;
- 7) создание виртуального прибора для управления созданным устройством.

Ниже приведено краткое описание программных средств, используемых на каждом этапе.

Продукт фирмы Aldec

Пакет Active-HDL фирмы Aldec представляет собой полностью интегрированную среду разработки цифровых устройств на основе текстовых HDL-описаний. Среда проектирования Active-HDL (см. рис. 1) даёт разработчикам независимость в использовании других средств проектирования из единого, полностью совмещённого окружения.

Именно свойство объединения разрозненных средств проектирования в оболочку Active-HDL является привлекательным для разработчика. Возможно подключение всех необходимых продуктов для проведения всех этапов разработки конфигурационной последовательности. В текстовом редакторе среды Active-HDL разработчик создаёт описание устройства на языке VHDL и выполняет проверку методом временных диаграмм (этап № 3). Этот метод обеспечивает визуализацию всех сигналов в тестируемом модуле.

Для детального тестирования проекта методом временных диаграмм, на языке VHDL вручную создаётся временный модуль верхнего уровня. Он управляет тестовыми сигналами, направленными во входные порты исследуемого модуля, в то время как среда отображает отклики модуля на временной диаграмме в виде цифровых сигналов (стробов), обеспечивая возможность оценки работы реализованных в модуле функций. Для проведения остальных этапов разработки подключаются специализированные среды. Таким образом, среда Active-HDL играет в проекте разработки не только связующую, но и основообразующую роль.

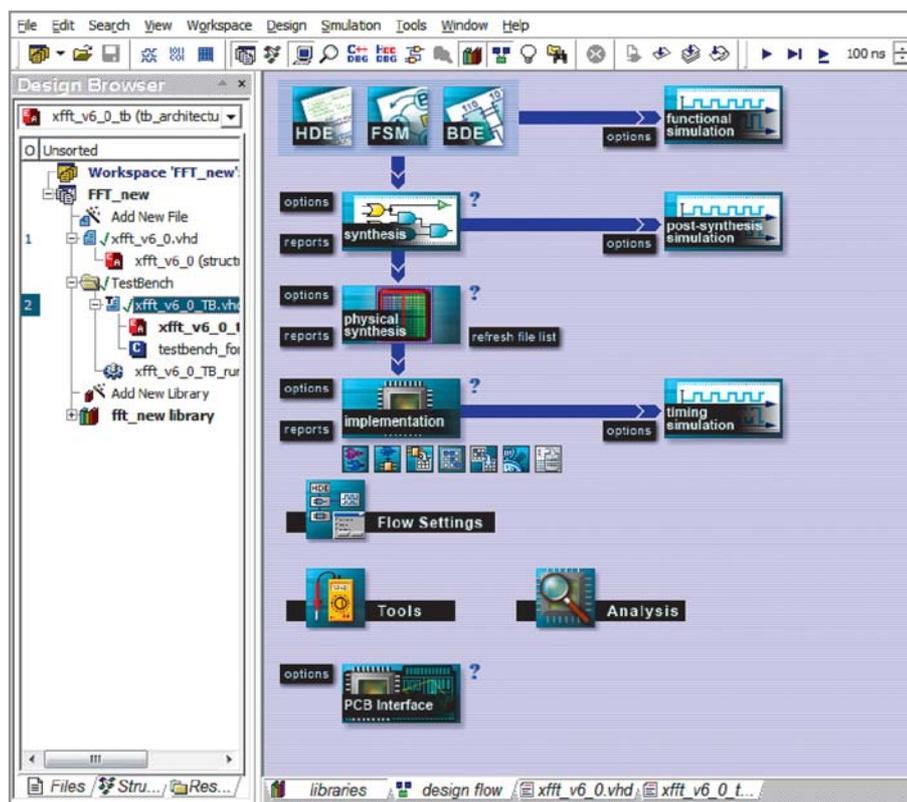


Рис. 1. Рабочий стол Active-HDL

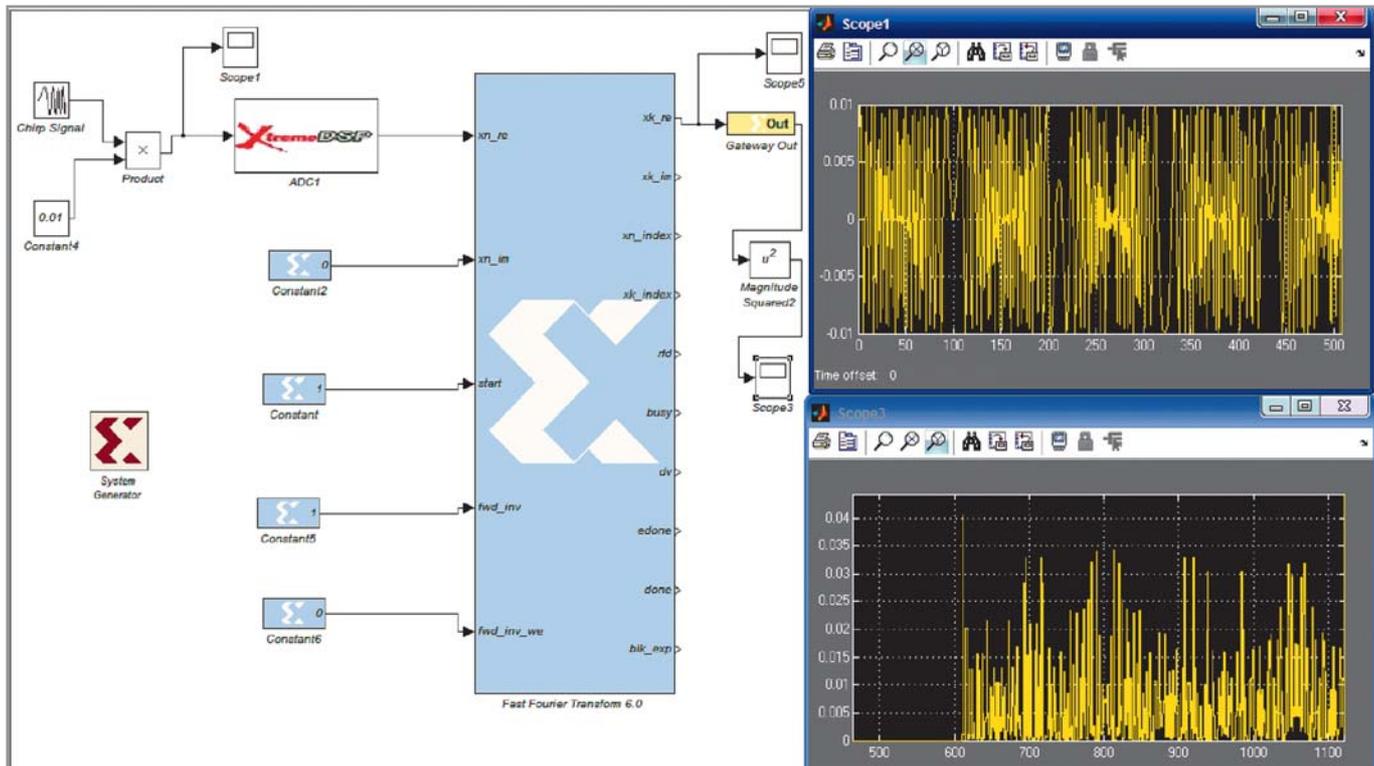


Рис. 2. Настройка блока БПФ в среде System Generator

Продукты фирмы Xilinx (Xilinx ISE)

Компания-разработчик ПЛИС Xilinx предлагает среду автоматизированного проектирования Xilinx ISE, в которую включены разнообразные средства выполнения различных задач. Некоторые из них использованы при подготовке данной работы.

System Generator (эман № 1)

Для решения задач ЦОС при разработке сложных устройств фирма Xilinx разработала среду визуального проектирования System Generator. Эта среда интегрирует функции MATLAB, дополняя их специфическими библиотеками, которые совместимы с ПЛИС. Разработчик представляет своё устройство в графическом редакторе в виде функциональных блоков, не отвлекаясь на редактирование мелких узлов, что позволяет сфокусироваться на решении основных задач. Поэтому использование данного продукта существенно экономит время, затрачиваемое на моделирование.

Фирмой Xilinx также предусмотрена функция генерации битового потока моделируемого устройства для конфигурирования ПЛИС непосредственно в среде System Generator. Однако для качественной разработки необходимо выполнение всех этапов, и такой вариант не может заменить их последо-

вательность, указанную выше для проектирования эффективного и сложного устройства. Также в System Generator не реализованы некоторые необходимые концепции. Например, в MATLAB не существует понятия «порт ввода-вывода», что является существенным недостатком. Поэтому использование продукта System Generator целесообразно только на этапе визуального моделирования.

CoreGenerator (эман № 2)

В помощь разработчику фирма Xilinx выпустила пакет программ Xilinx ISE – функционально ёмкий инструмент, позволяющий оптимизировать время разработки. Мощные средства создания сложных элементов принципиальной схемы ПЛИС встроены в систему автоматизации проектных работ (САПР) Xilinx ISE. Например, генератор системных IP-ядер является необходимым и очень удобным инструментом для разработчика. Он содержит специальные, генерируемые библиотеки символов, оптимизированные под конкретное семейство ПЛИС. Важно, что использование оптимизированных IP-ядер позволяет избежать ошибок, связанных с реальными задержками распространения сигнала внутри ПЛИС при работе на высокой частоте.

Генерируемое ядро представляется в виде пакета файлов настройки про-

екта, который содержит и файл на языке проектирования VHDL, включаемый в текст разрабатываемой программы. Используя данный инструмент, инженер может сосредоточиться на специфических аспектах проектирования.

Применительно к задачам ЦОС, продукт Core Generator обладает большими возможностями, предоставляя оптимизированные модули фильтров, блоков вычисления преобразования Фурье, линейных сдвиговых регистров, блоков с алгоритмами решения тригонометрических, гиперболических и уравнений с квадратным корнем. Все эти модули разработчик может настроить по своему усмотрению.

Выбранные ядра могут использоваться в качестве библиотек для блоков визуального проектирования в среде System Generator (см. рис. 2), ускоряя настройку функциональных модулей.

WebPack (эмань № 4 и № 6)

Важнейшим этапом разработки является физический синтез кода VHDL в описание устройства более низкого уровня абстракции. В процессе физического синтеза производится начальное размещение логических вентилей и функций (таблиц соответствия). Основываясь на результатах размещения, средства временного

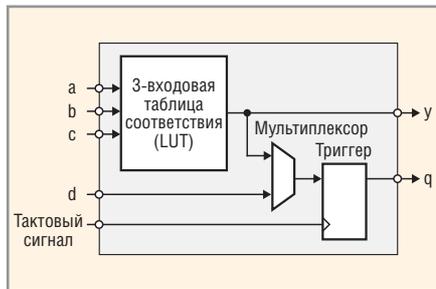


Рис. 3. Простой конфигурируемый логический блок

анализа, входящие в состав синтезатора, позволяют получить точные временные оценки, которые необходимы для компенсации негативных эффектов (задержек на проводниках и вентилях). Таким уровнем абстракции является представление в виде сопоставленной, скомпонованной и размещённой таблицы соединений конфигурируемых логических блоков (см. рис. 3) (КЛБ) и таблиц соответствия (netlist) [2].

На последующих этапах (физической реализации) эта информация используется в качестве отправной точки при выполнении процедуры локальной (точной) оптимизации размещения, после чего производится детальная разводка элементов устройства.

Синтез производится в автоматическом режиме, необходимо только указать название проектируемой ПЛИС

фирмы Xilinx и файл верхнего уровня синтезируемого проекта. На этапе конфигурации этот продукт, используя таблицу соединений КЛБ и таблиц соответствия (netlist), создаёт конфигурационный файл – битовый поток, который загружается в кристалл через интерфейс JTAG.

PlanAhead (эман № 5)

Этот продукт представляет собой редактор размещения отдельных цифровых узлов на кристалле ПЛИС. С помощью PlanAhead можно поделить проект на иерархические блоки и выполнять их размещение на кристалле на любом уровне иерархии, как вручную в редакторе (см. рис. 4), так и в автоматическом режиме. В среднем этот пакет увеличивает производительность проекта на 30%.

В предложенном способе проектирования ПЛИС используется лишь часть функций PlanAhead, а именно, оптимизация таблицы соединений КЛБ и таблиц соответствия (netlist), созданных синтезатором, с помощью методов трассировки и размещения логических блоков применительно к данному кристаллу. Для корректной работы трассировщика необходимо подготовить файл UCF, который содержит необходимую информацию в виде описания соответствий символьных идентификаторов портов ПЛИС

физическим контактам кристалла (pins):

```
#####
# System Clocks and Control Signals
#####

#System Clocks
#NET "clka" LOC = AF12;
#NET "clkb" LOC = A16;
#NET "clkc" LOC = AF11;

#Control and setup
#NET "fpga_done" LOC = Y1;
#NET "config_done" LOC = AC1;
#NET "reset1" LOC = H3;
#NET "slot_id0" LOC = H5;
#NET "slot_id1" LOC = H4;

#User Status LEDs
#NET "led1_green" LOC = D3;
#NET "led1_red" LOC = F3;
#NET "led0_green" LOC = E26;
#NET "led0_red" LOC = D26;

#Temp. Sensor device
#NET "alert1" LOC = H6;
```

Фирма – разработчик кристаллов Xilinx предлагает универсальный файл UCF для каждой ПЛИС, где описаны все порты. От разработчика устройства требуется грамотная оптимизация этого файла применительно к проекту. Все идентификаторы могут быть изменены произвольно, с необходимым учётом на этапе разработки № 3 (создание описания устройства на VHDL).

Размещение логических блоков производится в условиях экономного использования узлов кристалла. На выходе получается всё та же таблица (netlist), но уже полностью оптимизированная под выбранную ПЛИС.

ChipScope (эман № 6)

Конфигурация кристалла реализуется с помощью инструмента ChipScope. Продукт, оперируя битовым потоком, созданным WebPack, загружает его в ПЛИС с помощью программирующего кабеля.

Средство ChipScope также отвечает за внутрисхемную отладку сконфигурированного кристалла. Такой метод отладки называется сквозным проектированием и осуществляется с помощью дополнительных программных модулей, которые добавляются в код проекта на этапе проектирования № 3. Метод основан на управлении до-

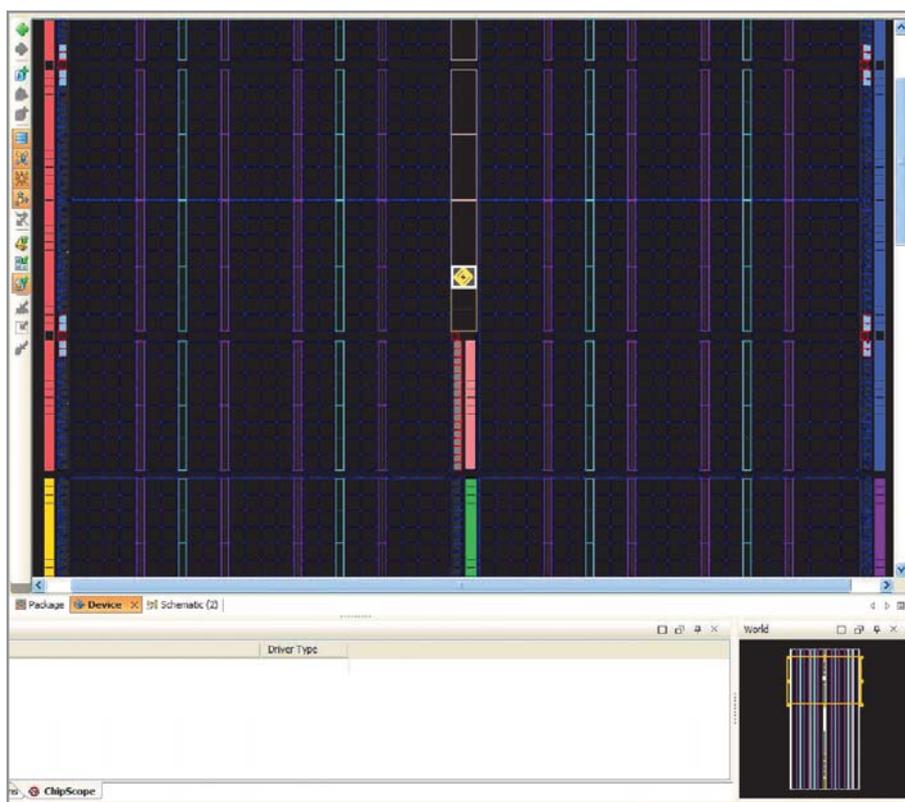


Рис. 4. Графическое изображение логических блоков ПЛИС в среде PlanAhead

полнительными модулями в среде ChipScore посредством загрузочного кабеля во время работы готового устройства. Чаще всего используются три модуля:

- Integrated Controller;
- Integrated Logic Analyzer;
- Virtual Input/Output.

Возможности каждого модуля уникальны и предоставляют разработчику удобные инструменты для отладки программной составляющей разрабатываемого устройства.

Модуль Integrated Controller обеспечивает соединение между двумя другими модулями и средой ChipScore и способен поддерживать до 15 каналов передачи одновременно.

Модуль Integrated Logic Analyzer используется для проверки любых внутренних сигналов проекта. Он создаёт триггеры на свободном от узлов проекта пространстве кристалла, которые контролируют определённые разработчиком сигналы как в единый момент времени, так и за определённый промежуток. Глубина триггеров также настраивается разработчиком и зависит только от количества свободных КЛБ кристалла.

Настраиваемый модуль Virtual Input/Output может управлять внутренними сигналами ПЛИС в реальном времени. Он создаёт виртуальные указатели статуса через асинхронные и синхронные входные порты. Таким образом, разработчик может в любой момент времени определить состояния внутренних сигналов, что позволяет надёжно фиксировать неполадки и совершенствовать проект.

LABVIEW – СРЕДА РАЗРАБОТКИ ВИРТУАЛЬНЫХ ПРИБОРОВ

Важная задача, которая не относится к проектированию ПЛИС, выполняется разработчиком на этапе № 7 и заключается в организации управления созданным устройством и обработке полученных результатов. На данном этапе удобно использовать виртуальные приборы (см. рис. 5), которые предлагает программный продукт LabVIEW [5].

Программы, созданные в среде LabVIEW, называются виртуальными приборами, потому что их внешний вид и выполняемые функции схожи с реальными физическими приборами, такими как осциллограф или мультиметр. Среда LabVIEW содержит обширный набор инструментальных средств для

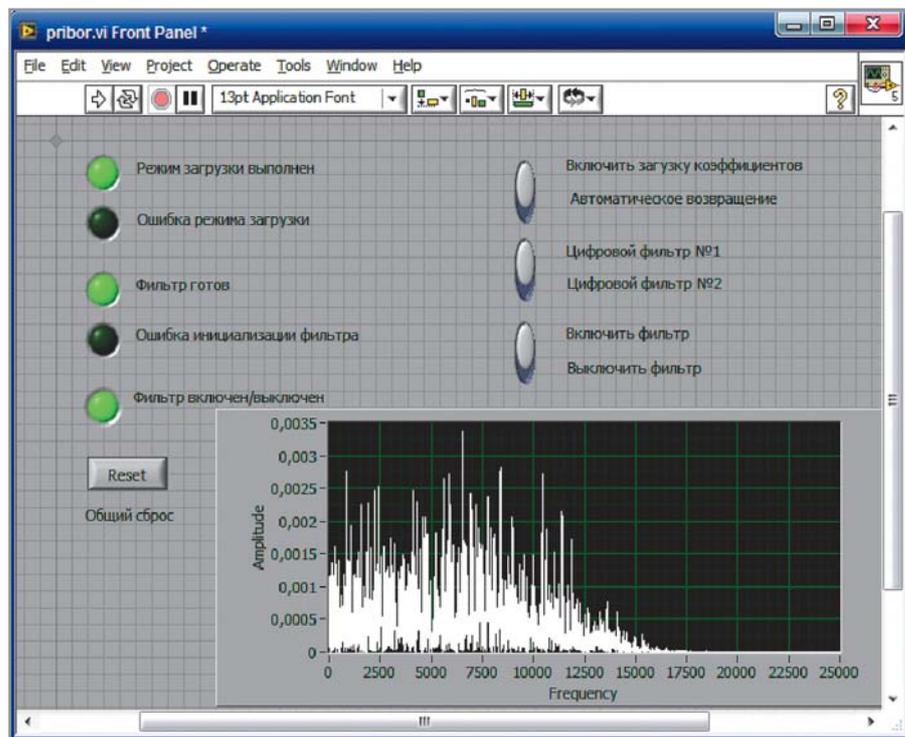


Рис. 5. Использование виртуального прибора для управления ПЛИС

сбора, анализа, представления и хранения данных, а также инструменты, помогающие отладить создаваемый код.

Написание программы в LabVIEW начинается с создания интерфейса пользователя, содержащего элементы управления и индикаторы. Примерами элементов управления являются ручки управления, кнопки, круговые шкалы и другие элементы ввода. Индикаторами являются графики, светодиодные индикаторы и другие элементы вывода. Затем разработчик создаёт на блок-диаграмме код, используя другие виртуальные приборы и структуры для управления объектами лицевой панели.

Программная среда LabVIEW может применяться для взаимодействия с аппаратными средствами с помощью интерфейсов GPIB, PXI, VXI, RS-232 и RS-485. Использовать виртуальные приборы очень удобно, поскольку они могут работать независимо от среды разработки. Графический язык программирования G позволяет ускорить процесс разработки, а база из более чем 600 драйверов даёт возможность организовать подключение внешних устройств.

ЗАКЛЮЧЕНИЕ

В статье проведён анализ программных средств проектирования и предложен алгоритм создания устройств ЦОС с использованием выбранного инструментария. Даны рекомендации

по проектированию устройства ЦОС на основе ПЛИС. Подобраны инструменты, соответствующие алгоритму проектирования.

В ходе работы отмечены следующие особенности:

- визуальное проектирование в среде System Generator позволяет сократить время настройки оптимизированных блоков по сравнению с настройкой методом временных диаграмм;
- использование программного продукта PlanAhead позволяет значительно увеличить производительность проекта, а также пространственно оптимизировать логические блоки на кристалле ПЛИС;
- сквозное проектирование ускоряет отладку проекта устройства.

ЛИТЕРАТУРА

1. Лобов В.И. Теоретические основы цифровой обработки сигналов. Москва, 2008.
2. Максфилд К. Проектирование на ПЛИС. Додэка-XXI, 2007.
3. Крушиницкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах с программируемой структурой. БХВ-Петербург, 2006.
4. Зотов В.Ю. Проектирование цифровых устройств на основе ПЛИС фирмы Xilinx в САПР WebPACK ISE. Горячая линия -Телеком, 2003.
5. Вводный курс LabVIEW 8.5. National Instruments, ftp://ftp.ni.com/pub/branches/russia/software/labview_8.5.pdf.