

Процесс разработки проекта для ПЛИС в пакете Actel Libero IDE

Сергей КАРПОВ
karpov@actel.ru

С появлением новых семейств многократно программируемых ПЛИС Actel ProASIC3/E/L, IGLOO/E/Plus и Fusion, изготовленных по технологии FLASH, значительно возрос интерес к продукции компании и, соответственно, к средствам разработки — пакету Actel Libero IDE. Данная статья открывает цикл публикаций, в которых будут рассмотрены основные принципы и этапы процесса разработки и верификации проекта в интегрированной среде Libero. А также мы более подробно остановимся на тех нововведениях, которые появились в последней версии Actel Libero IDE v8.4.

Введение

Для начала остановимся немного на теории проектирования — проектирования системы в целом. У кого-то возникнет вопрос: а при чем здесь система в целом, если речь идет о проекте для ПЛИС? И это главная ошибка при разработке! Применение программируемой логики наиболее эффективно при системном подходе и позволяет значительно сократить время разработки, габариты и стоимость системы в целом, а в случае ошибки перепрограммировать ее значительно проще и быстрее, чем переделать печатную плату. ПЛИС — это не микросхема в привычном понимании. Это, по сути, часть печатной платы, часть схемы, которую раньше реализовывали на отдельных контроллерах, процессорах и логике. А теперь все это можно разместить в одной микросхеме. Поэтому разработку проекта ПЛИС нужно начинать не после проработки структуры системы, а одновременно с ней.

В качестве средства разработки для своих микросхем Actel предлагает Actel Libero IDE — полный интегрированный пакет, который включает в себя все необходимые программы и библиотеки для разработки, отладки и верификации проектов для микросхем программируемой логики и их программирования. Большая часть из них входит в инсталляционный пакет. Остальные, как, например, внутрисхемный отладчик прошивки ПЛИС Identify, можно скачать с англоязычного сайта Actel. Кроме того, пакет включает в себя большое число IP-модулей, многие из которых предоставляются бесплатно в виде списка соединений.

Уже достаточно много написано книг и статей по работе с программируемой логикой, поэтому здесь мы не будем рассматривать основы программирования на языках Verilog или

VHDL или работы в симуляторе ModelSim. Основная цель данной статьи — помочь разработчикам быстро освоить весь процесс разработки проекта, то есть разработку поведенческой модели (RTL-модели), синтез, трассировку и генерацию файла «прошивки», симуляцию и аппаратную верификацию проекта на разных этапах, а также будут рассмотрены вопросы оптимизации и стратегии трассировки. Более подробно мы остановимся на особенностях работы в Libero IDE версии 8.4 и разработки проектов применительно к ПЛИС Actel.

Процесс проектирования

Одной из основных проблем при разработке проекта ПЛИС является правильная структура. От этого зависит, насколько быстро и легко можно будет отладить и верифицировать проект, а при необходимости — внести изменения в структуру проекта и дизайн отдельных модулей, а также будет ли возможность использования готовых модулей в последующих проектах. Поэтому в проекте должно быть, как минимум, два модуля разных уровней: модуль поведенческой модели (RTL), которая не привязана к какому-либо конкретному семейству, и верхний уровень, включающий модуль RTL-модели и модули окружения для конкретного семейства и микросхемы, такие как блоки интегрированного ОЗУ, умножители частоты, порты ввода/вывода и т. д. Начинать лучше с проработки структуры RTL-модели с последующим ее заполнением. Для этого в Actel Libero IDE можно создавать иерархические проекты с предварительной прорисовкой структуры проекта без заполнения: создавать и вставлять в проект пустые блоки, в которых определены только основные порты ввода/вывода с последующей проработкой и корректировкой.

Actel Libero IDE — первое знакомство

После того как проработана детальная структура системы, определены структура, которая будет располагаться в ПЛИС, и элементная база, можно приступать непосредственно к разработке дизайна программируемой логики. Разработка проекта для ПЛИС Actel осуществляется в пакете Actel Libero IDE (рис. 1). Это интегрированная среда, которая поддерживает разработку проектов под все микросхемы всех семейств ПЛИС Actel и включает в себя весь набор инструментов для разработки, отладки и верификации, а также средства для прошивки ПЛИС. На данный момент последняя версия — Libero IDE 8.4. Пакет предлагается разработчикам в трех вариантах: полный платный пакет, бесплатный полный пакет на 45 дней и бесплатная версия пакета с рядом ограничений. Бесплатная версия поддерживает разработку проектов под микросхемы объемом до 1 500 000 системных вентилей включительно и имеет ряд ограничений по анализу результатов синтеза и симуляции.

По умолчанию в основном окне пакета присутствуют окна «Менеджер проекта» (Design Explorer), «Процесс разработки» (Project Flow), библиотеки с макрофункциями и IP-модулями и ядрами (Catalog), окна лога работы и свойств объектов проекта.

По сравнению с предыдущими версиями пакета сразу видно одно отличие — отсутствует CoreConsole среди инструментов разработки проекта в окне процесса разработки (предполагается, что он был установлен и интегрирован в Libero IDE). Это связано с тем, что в версии 8.4 все функции CoreConsole теперь доступны в SmartDesign, а все IP-модули, включая процессорные ядра, присутству-

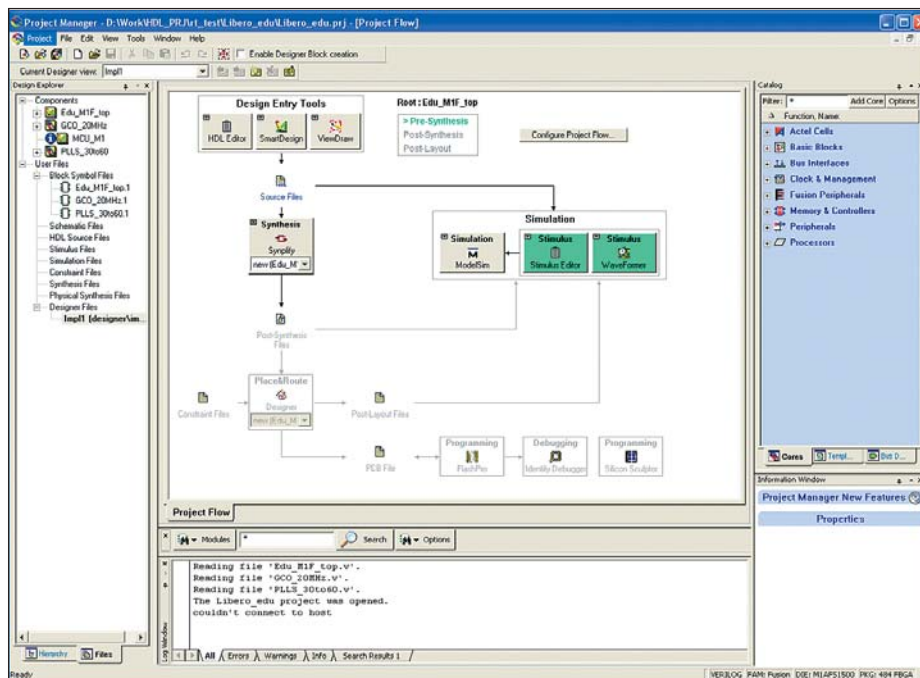


Рис. 1. Actel Libero IDE 8.4

ют в библиотеке (окно “Catalog”). Запустить CoreConsole из основного окна можно через меню «Инструменты» (“Tools”).

Создание проекта

При создании нового проекта запускается «Мастер». Необходимо задать название проекта, язык описания (Verilog или VHDL) и выбрать семейство, под которое будет создаваться проект. Можно также выбрать микросхему и корпус и добавить, если имеются, готовые файлы в проект или сделать это позже в процессе работы. Бесплатная версия позволяет использовать модули только на одном языке, выбранном при создании проекта, а платная поддерживает смешанный способ разработки.

Пакет Libero IDE позволяет создавать иерархические проекты. В окне менеджера проекта (“Design Explorer”) на закладке «Иерархии» (“Hierarchy”) показаны только модули и их зависимость друг от друга. На закладке «Файлы» (“Files”) представлены все файлы проекта, включая модули тестбенча, результаты синтеза и трассировки, файлы «прошивки» и т. д.

Для создания модулей проекта имеется текстовый и два графических редактора. Текстовый HDL-редактор позволяет создавать модули проекта — компоненты или тестбенч — на языках Verilog или VHDL. В бесплатной версии можно создавать и использовать модули, написанные на языке, который выбран для проекта при его создании. Платная версия поддерживает смешанный режим проектирования. HDL-редактор достаточно простой, и мы на нем не будем останавливаться. Можно лишь заметить, что если создается

модуль более высокого уровня иерархии в виде текста, то для облегчения добавления модулей в окне менеджера проекта на закладке с файлами (“Files”) для графического модуля можно открыть сгенерированный HDL-код (рис. 2) и скопировать в новый файл необходимые текстовые блоки (определения модулей и портов ввода/вывода).

SmartDesign — это графический редактор схем. Первоначально он предназначался для

создания макросов и простых блоков, но начиная с версии 8.4 он стал полноценным схемным редактором. Хотя и несколько необычным, что вызывает поначалу некоторые трудности у специалистов, ранее работавших с пакетами для ПЛИС Altera или Xilinx. Поэтому уделим ему особое внимание.

Создадим новый модуль SmartDesign — процессорный модуль с ядром Cortex-M1 и несколькими контроллерами. После завершения работы «Мастера» создания нового проекта модуля откроется основное окно для разработки — “Canvas” (рис. 3). На поле можно добавлять примитивы, макросы и IP-модули из библиотеки (окно “Catalog”), а также компоненты, созданные в HDL-редакторе, SmartDesign или ViewDraw. Для добавления нового библиотечного элемента в окне “Catalog” выберите требуемый компонент и кликните по нему дважды левой клавишей мыши или кликните один раз правой и в выпадающем меню выберите “Instantiate in <имя_SmartDesign_модуля>”. Если выбранный компонент конфигурируемый, то откроется соответствующее диалоговое окно конфигурации. Необходимо отметить, что набор макросов и IP-модулей изменяется в зависимости от выбранного семейства. Для добавления ранее созданных HDL или графического модулей в окне «Менеджер проекта» нужно кликнуть один раз правой клавишей мыши на соответствующем компоненте и в выпадающем меню выберите “Instantiate in <имя_SmartDesign_модуля>”.

Добавим все необходимые компоненты на поле. Теперь их надо соединить между собой и определить интерфейс верхнего уровня. Основная часть соединений задается с использованием таблиц (рис. 4). Для этого кликните один раз правой клавишей мыши на компоненте и в выпадающем меню выберите “Edit Connection” или нажмите соответствующую кнопку на панели инструментов. Некоторые связи, например, связи верхнего уровня модуля или связи подключения к шине AMBA (AHB-APB), можно определять непосредственно в основном окне. Для этого кликните один раз правой кнопкой мыши по выводу компонента на схеме и в выпадающем меню выберите “Promote to Top Level” или “Fine Compatible Bus Interface” для шины AMBA. Слева и справа располагаются поля внешнего интерфейса (верхнего уровня), закрашенные серым цветом. Если соединение с верхним уровнем подключено, то на поле появляется порт и связь между этим портом и выбранным портом компонента. Если это схема верхнего уровня, то порты являются выводами ПЛИС. Тип порта ввода/вывода (CMOS, TTL, PCI и т. д., 1,2; 1,5; 2,5 и 3,3 В) можно задать непосредственно в схеме, добавив соответствующий примитив или блок из библиотеки, или сделать это позже в синтезаторе Synplify либо в пакете трассировщика Designer через файлы констрейнов.

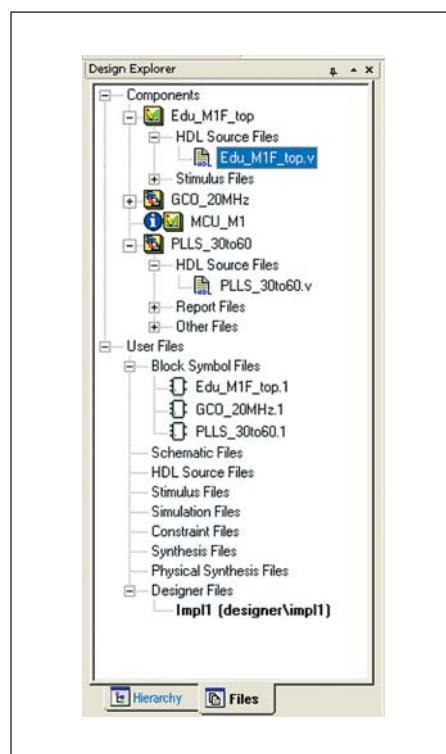


Рис. 2. Менеджер проекта

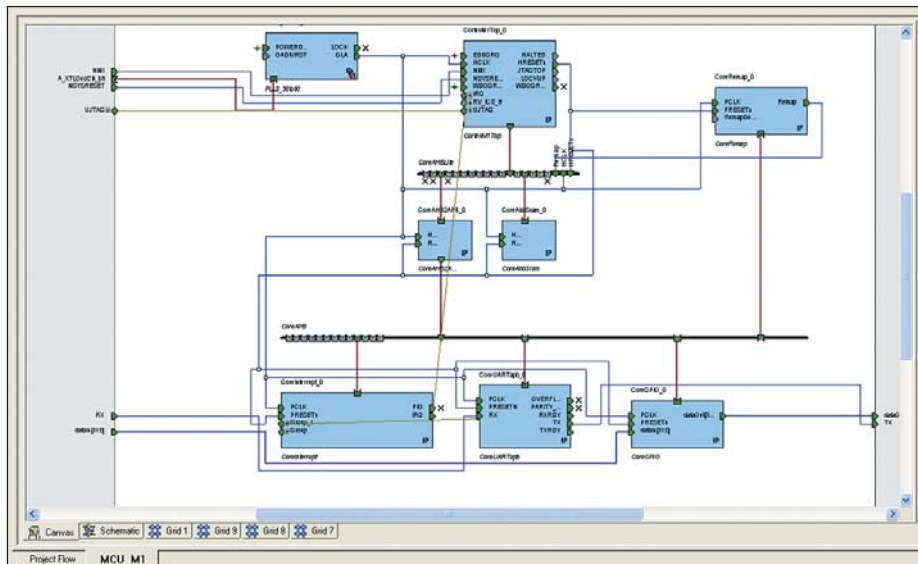


Рис. 3. Основное окно редактора SmartDesign

Instance-Instance View		Instances								
Instance	Port Name	Attribute	MCU_M1	CoreAHB2APB_0	CoreAHB1to_0	CoreAbbStam_0	CoreAPB_0	CoreGPIO_0	CoreInterrupt_0	CoreI
MCU_M1	A_XTL_OscClk_In0	[1:0]								
	dataIn	[15:0]								
	dataOut	[15:0]								
	IMM									
	NSYSRESET									
	RX									
	TX									
	UJTAG_TCK									
	UJTAG_TDI									
	UJTAG_TDO									
UJTAG_TMS										
UJTAG_TRSTB										
CoreInterrupt_0	APBtoI2C		0	2	2	1	2	8		
	PCLK									
	PRESETn									
	Remap									
CoreRemap_0	APBtoI2C									
	RemapDefault									

Рис. 4. Таблица соединений редактора SmartDesign

Для удобства разработки связи между элементами схемы можно сделать невидимыми.

При работе можно группировать порты и связи, добавлять и удалять порты интерфейса, переконфигурировать модули и многое другое. В частности, в SmartDesign можно создавать пустые компоненты, содержащие только порты ввода/вывода. Это позволяет сначала прорисовать иерархическую структуру будущего проекта, а затем ее заполнить. Это удобно и при работе над проектом в команде. Для создания и проверки работоспособности отдельных частей в составе проекта не требуется, чтобы все компоненты проекта были закончены. Пустые модули при компиляции игнорируются.

Для добавления порта в модуле (необязательно пустом) кликните один раз правой клавишей мыши на поле внешнего интерфейса — слева и справа на поле серого цвета — и в выпадающем меню выберите «Добавить порт» («Add port»). На открывшейся панели введите имя порта и тип — ввод, вывод или двунаправленный. Порт может быть как одиночным, так и шиной с разрядностью, которая задается непосредственно в имени.

В окне «Schematic» можно проконтролировать правильность созданной схемы.

ViewDraw — это схемотехнический редактор компании View Logic. К сожалению, он имеет большое количество ограничений, а также возникают определенные трудности в процессе работы над проектом, когда нужно внести коррективы. Это связано с тем, что для добавления во ViewDraw внешних модулей и наоборот требуется конвертация. Поэтому компания Actel не рекомендует использовать его при разработке новых проектов, а работать в SmartDesign.

Продолжение следует

Литература

1. www.actel.ru