

Кочетков Д.В.

*Научный руководитель старший преподаватель Смирнов М.С.
Муромский институт (филиал) федерального государственного образовательного
учреждения высшего образования «Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23
E-mail: Koschetkov1995@gmail.com*

Методическая разработка описания процесса создания цифровых устройств на ПЛИС.

Возрастание сложности СБИС на базе развития интегральной технологии позволяет иметь в аппаратуре все большее число компонентов, схемотехнически реализовывать все более многообразные и сложные функции. Всё это приводит к кризису разработки. Разработка аппаратного устройства и графической нотации схем современных СБИС занимает все большее количество времени и ресурсов. Для борьбы с кризисом разработки существует два альтернативных пути:

- переход к высокоуровневым средствам и методам проектирования;
- совершенствование элементной базы – создание аппаратно реализованных на кристаллах, функционально законченных и параметризованных модулей (умножители, модули памяти, интерфейсные модули и т.д.) [1].

Альтернативой «рисования» детализированных схем из низкоуровневых элементов являются языки описания аппаратуры высокого уровня. Собирают языки этого класса называют языками HDL (Hardware Description Language). Они не только обеспечивают компактную запись для проектируемой схемы, дают значительное сокращение трудоемкости и сроков разработки больших схем, но и упрощают миграцию, перенос проекта на разные варианты интегральных технологий, реализацию их в СБИС с учетом специфики технологий различных производителей. Разработчик получает возможность оценивать варианты реализации проектируемого устройства в СБИС при различных вариантах проектных ограничений, на различных технологиях, у различных производителей.

Языки HDL предназначены для решения комплекса задач в ходе проектирования и применения цифровых систем, их аппаратных средств, в том числе:

1. Описания структуры системы, декомпозиции системы на подсистемы, спецификации связей и взаимодействия подсистем.
2. Спецификации функционирования системы, узлов, блоков, реализуемых функций. Спецификация дается в алгоритмической форме, с использованием привычных современному специалисту программных конструкций алгоритмического языка, включающих в себя спецификацию временного поведения сигналов и блоков.
3. Моделирования системы и ее работы на основе четкой спецификации структуры системы, а также функционирования ее компонентов.
4. Синтеза схемотехнической реализации системы, автоматической генерации детальной структуры на основе строгой спецификации системы на языке HDL – спецификации на более абстрактном уровне.

На рис. 1 изображены диаграммы процессов синтеза и моделирования [2]. Для процесса моделирования используется синтезированный HDL проект и специальные библиотеки для создания симуляционной модели. Затем выполняется моделирование на основе этой модели. Это ускоряет процесс моделирования, по сравнению с повторной трансляцией кода симулятором. В процесс моделирования используется набор тестовых массивов и генерирует осциллограммы выходных сигналов или текстовый файл данных для последующего анализа результатов. В процессе моделирования, не конкретизируется микросхема. Это потому, что симулятор выполняет код, «как есть». Симулятор действительно не волнуется, где реализуется проект, его интересует только интерпретация описанных в коде функций.

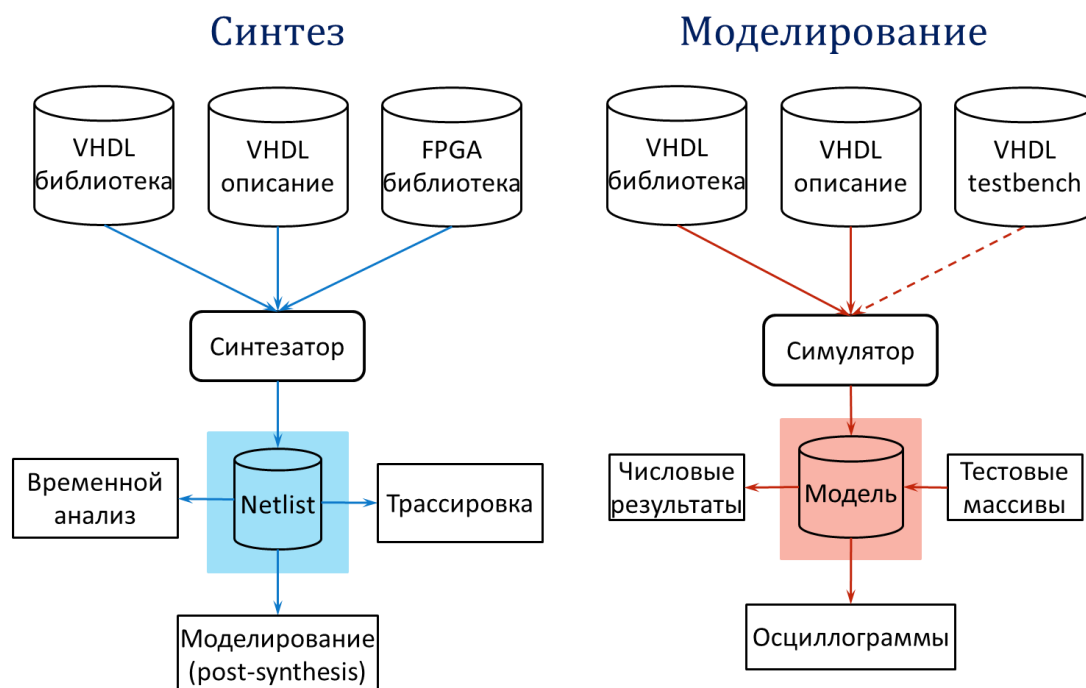


Рис. 1 Процессы синтеза и моделирования VHDL проекта

Изображенный слева синтезатор использует то же HDL описание и библиотеки для создания нетлиста. Нетлист – это вентиляльное описание схемы, созданной на основе HDL кода. В процессе синтеза используется технологическая библиотека. Каждое семейство ПЛИС имеет собственную технологическую библиотеку, которая используется синтезатором для реализации кода в данном конкретном устройстве. С помощью созданного синтезированного нетлиста, проект может быть размещен внутри конкретной ПЛИС. Если реализуется пост-синтезированное моделирование, то синтезатор генерирует HDL файлы с результатами синтеза. Данное моделирование дает уверенность, что на этапе синтеза не появились дополнительные ошибки.

Литература

1. Pedroni V., Circuit Design and Simulation with VHDL, 2nd ed., MIT Press, 2010.
2. Ciletti M., Advanced Digital Design with the Verilog HDL, 2nd ed., Prentice Hall, 2010.