ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ ЯЗЫКОВ ОПИСАНИЯ АППАРАТУРЫ VHDL И VERILOG

Садковая М.В.
Научный руководитель – Филиппенко И.В.
Харьковский национальный университет радиоэлектроники
(61166, Харьков, пр. Науки, 14, каф. АПВТ, тел. (050)861-15-14)
e-mail: mariia.sadkova@nure.ua

VHDL and Verilog are hardware descriptive languages. These languages are designed for simulate electronic circuits at the level of valve, register transmissions, microcircuit cases. Therefore, they can be called languages of through fвnunctional and logical design. However, they have a number of differences, which we will consider in this article.

Языки VHDL и Verilog относятся к языкам описания аппаратуры. Эти языки предназначены для моделирования электронных схем на уровнях вентилей, регистровых передач, корпусов микросхем. Поэтому их можно назвать языками сквозного функционально-логического проектирования. Однако они имеют ряд отличий, которые необходимо учитывать при проектировании различных цифровых устройств.

VHDL (Very high speed integrated circuits Hardware Description Language) – данный язык предназначен для описания проектируемых систем на схемотехническом уровне проектирования и замены классического подхода к схемотехническому проектированию на уровне отдельных элементов. Язык позволяет описывать цифровые системы на алгоритмическом уровне. При помощи специального программного обеспечения описание на языке VHDL преобразовывается в схему на уровне простейших элементов цифровой электроники.

Verilog – это язык описания аппаратуры, используемый для разработки и моделирования электронных систем. Этот язык (также известный как Verilog HDL) позволяет осуществить проектирование, верификацию и реализацию (например, в виде СБИС) аналоговых, цифровых и смешанных электронных систем на различных уровнях абстракции.

По сравнению с Verilog, VHDL более богатый и строго типизованный и строго детерминистический язык, более детализированный. В результате проекты, написанные на VHDL, считаются самодокументированными. Синтаксис сильно отличается от стиля языка C, и инженеры, работающие в VHDL, постоянно сталкиваются с необходимостью явного преобразования из одного типа данных в другой. VHDL часто сразу показывает ошибки, которые пропускает Verilog, а так же имеет подчеркнуто однозначно недвусмысленную семантику, и поэтому легче переносится между разными системами разработки (в том смысле, что перенос точнее переносит все тонкости работы исходного проекта).

Verilog выглядит более гармонично и удобочитаемо. Что достигается отсутствием длинных названий типов данных и строк объявления сигналов и регистров. Так же проще и аккуратнее реализована запись векторов.

Алфавит моделирования в VHDL включает в себя 9 значений: {‘U’, ‘X’, ‘0’, ‘1’, ‘Z’, ‘W’, ‘L’, ‘H’, ‘-’}. Verilog же имеет всего 4 значения: {‘0’, ‘1’, ‘X’, ‘Z’}.

Verilog слабо проверяет типы, и более краток, с эффективной нотацией. Он также детерминистический. Все типы данных заранее определены в Verilog, и каждый из них имеет битовое представление, по сравнению с VHDL, в котором пользователь имеет возможность вводить свои типы данных. Синтаксис похож на C. Из-за своей структуры VDHL отлавливает больше ошибок уже на ранних стадиях процесса разработки. С другой стороны Verilog позволяет инженерам быстро описывать модели.

Так же в VHDL, для расширения языка, позволяется использовать внешние пакеты и библиотеки, в то время как, Verilog не имеет подобной возможности. Из-за своей структуры VDHL отлавливает больше ошибок уже на ранних стадиях процесса разработки. С другой стороны Verilog позволяет инженерам быстро описывать модели.

Таким образом, VHDL является более академичным, многословным и сложным языком. Требуется написание большего объема кода, но строгость означает, что он с большей вероятностью будет работать. Verilog проще для типичного цифрового дизайна, но, соответственно, упрощает создание сложных ошибок. Выбор одного или другого зависит от используемых инструментов. Например, некоторые из популярных инструментов FPGA лучше работают с VHDL, когда популярные инструменты ASIC улучшают работу с Verilog

Список источников:

1. microsin URL: http://microsin.net/programming/xilinx/difference-between-vhdl-verilog-systemverilog.html
2. VHDL с нуля. // easyelectronics URL: http://we.easyelectronics.ru/plis/vhdl-s-nulya.html
3. Исследование комбинационных устройств: // URL: http://we.easyelectronics.ru/plis/vhdl-s-nulya.html
4. Особенности языков описания архитектуры // parallel URL: https://parallel.ru/fpga/hdl.html
5. habr URL: https://habr.com/ru/post/191606/