

# ИНСТРУМЕНТАРИЙ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ЦИФРОВЫХ УСТРОЙСТВ С ПРОГРАММИРУЕМОЙ ЛОГИКОЙ

Ладыженский Ю.В., Голубенко М.А.

Кафедра ПМИИ, ДонГТУ

ladyzhen@r5.dgtu.donetsk.ua

michael@r5.dgtu.donetsk.ua

## Abstract

*Ladyzhensky Y.V., Golubenko M.A. Tools for automation of design of digital devices with programmable logic. The classification is carried out, regularities and perspective directions of development of automated designing tools for programmable devices are revealed. The opportunities of usage of parallel calculations and computer networks for the decision of an acceleration problem of logic simulation are marked.*

## Введение

При разработке современной электронной аппаратуры широко применяются программируемые пользователем большие интегральные схемы (ПЛИС). Это повышает надежность цифровых устройств, снижает их габариты, потребляемую мощность, улучшает другие технико-экономические характеристики. ПЛИС дают возможность разработчику самостоятельно создавать специализированные устройства для различных применений.

Актуальной задачей является создание и совершенствование систем автоматизированного проектирования (САПР) цифровых устройств на ПЛИС с использованием инструментальных персональных ЭВМ (ПЭВМ). Ресурсы ПЭВМ позволяют в приемлемое время реализовать основные этапы проектирования устройств на ПЛИС. Используя САПР на ПЭВМ, можно выполнить автоматизированный синтез устройства по описанию на некотором входном языке, проверить корректность синтеза моделированием на логическом уровне и сформировать файлы программирования ПЛИС.

Данная работа является продолжением исследований, проведенных в [1], и посвящена инструментам электронного проектирования и моделирования под Windows.

## 1. Инструменты проектирования на основе языков описания аппаратуры (HDL)

В настоящее время язык Verilog стал стандартом de facto для описания аппаратуры. Его интерфейс позволяет встроенным инструментам легко получить доступ к результатам и другим внутренним функциям моделирования [2].

Язык VHDL, с другой стороны, создавался многими конкурирующими разработчиками, и стандарт VHDL не определял программный интерфейс языка или общие форматы файлов для различных инструментов. Поэтому возникала необходимость в спецификациях для каждой системы моделирования.

Ряд современных инструментов имеет возможность работы с двумя языками, но не все инструменты имеют возможности объединения средств для моделирования и синтеза, облегчающего отладку и верификацию [2].

**Категории инструментов.** Среди широкого спектра инструментов и языков, некоторые инструменты доступны только для одного HDL, а не для всех. Некоторые инструменты обладают "размытой" функциональностью. Такие инструменты не являются частью моделирования или синтеза, поэтому они могут непосредственно не отображаться в какой-либо HDL. В целом инструменты перекрывают диапазон функций от ввода проекта через верификацию до физической разводки проводников, или даже изготовления [2].

На рисунке 1 приведена классификация инструментов.

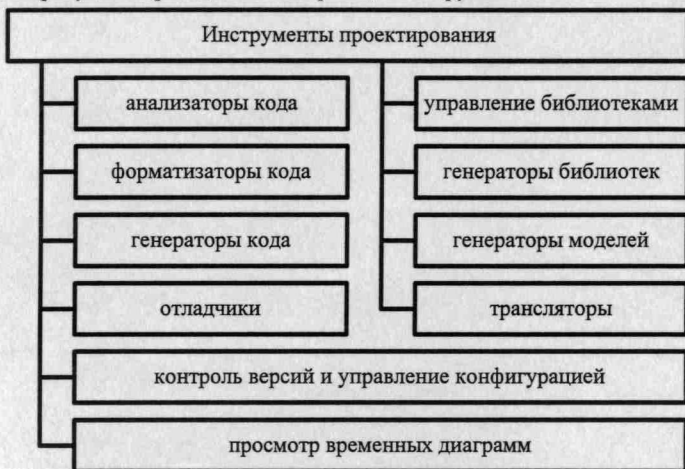


Рисунок 1 – Классификация инструментов проектирования

**Анализаторы кода.** Инструменты анализа кода исторически происходят от инструментов разработки программного обеспечения. Новизной отличаются инструменты определения зон действия кода. Результаты работы таких инструментов показывают, насколько проект соответствует требованиям некоторых видов функциональности, таких как тестируемость, зона действия неисправностей или зона действия кода проекта.

**Форматизаторы кода.** Эти инструменты преобразуют входные файлы, чтобы обеспечить инструментам моделирования и синтеза работу с правильными данными. Они могут быстро просматривать все описание проекта для поиска неправильных конструкций, неопределенных точек входа или выхода, кодов, не поддающихся синтезу. Форматизаторы структурируют листинг обработки для повышения читабельности.

**Генераторы кода.** Генераторы кода получают входные описания на других языках и конвертируют их в формат HDL для моделирования. Некоторые инструменты генерируют информацию для управления временем, преобразуют данные в ограничения для синтеза или в промежуточный формат файла описания задержек SDF.

**Отладчики.** Инструменты отладки помогают управлять исходными файлами при моделировании и обнаруживать проблемы в секциях кода.

Некоторые функции могут быть присоединены к другим инструментам, таким как инструменты просмотра временных диаграмм и конверторы из HDL в схемное представление, чтобы обеспечить перекрестные связи просматриваемых форматов и потока сигналов с HDL.

Управление библиотеками. Этот класс инструментов отслеживает версии инструментов и файлов и обеспечивает совместимость отдельных объектов. Эти функции особенно важны, когда различные версии программного обеспечения могут использовать различные количества или типы лицензий.

Генераторы библиотек. Эти инструменты генерируют библиотеки элементов проекта и макроячеек для использования инструментами моделирования и синтеза. Модели функций-примитивов должны отображаться в соответствующие специальные интегральные технологии и процессы для генерации наиболее точных моделей.

Генераторы моделей. Эти инструменты используются для разработки, как функциональных описаний типа черного ящика, так и полных точных по времени моделей библиотечных элементов. Инструменты обеспечивают описание характеристик всех уровней моделей для применения других инструментов проектирования.

Трансляторы. Инструменты трансляции преобразуют типы файлов. Среди типичных инструментов трансляции имеется три основных подкатегории. Первая – конверторы из языка в язык, изменяющие файлы проекта из формата одного HDL в другой. Эта функция полезна, когда соединяются ручные или ранее сделанные проекты с новыми проектами. Второй класс инструментов проектирования преобразует файлы стандартного формата в какой-либо другой формат или другие стандартные представления, такие как EDIF или SDF. Наконец, некоторые инструменты изменяют представления основного проекта. Так можно преобразовать описание из HDL в схему или другой графический формат, чтобы обеспечить лучшее документирование и понимание проекта пользователем.

Контроль версий и управление конфигурацией. Эти инструменты помогают поддерживать непрерывность и связность проекта на основе базы данных из файлов проекта, отмечая файлы - предшественники и приемники, и другие присоединенные файлы. Они отслеживают хронологию и порядок версий модулей проекта, помогают отыскивать аномалии в версиях на некотором отрезке времени путем проверки более ранней версии и возвратом к ней через проверку промежуточных версий.

Просмотр временных диаграмм. Инструменты просмотра диаграмм преобразуют последовательности единичных и нулевых состояний моделей в трассы, пригодные для логического анализа и показывающие направленность и временные свойства логических функций. Временные диаграммы более просты и наглядны, чем последовательности нулей и единиц или листинги HDL, при локализации нарушений установки и сброса триггеров, выявлении состязаний, неправильных логических последовательностей, ошибок синхронизации и сбоев.

Новой тенденцией в инструментах электронного проектирования и дополняющих инструментах является добавление языков сценариев, которые имеют доступ к внутренним данным и автоматизируют управление инструментами, и в некоторых случаях разрешают пользователю добавлять новую функциональность в инструменты.

Свойства основных применяемых на платформах Windows95/NT инструментальных систем приведены в табл.1-5 [3,4]

Таблица 1 Инструменты синтеза

Компания	Продукт	Описание	Цена, тыс.\$
Exemplar Logic	Galileo Logic Explorer	Синтез на VHDL и Verilog, архитектурная оптимизация FPGA и ASIC, специальная реализация заданных функций.	от 9
Synplicity	Synplify. Synplify-Lite	Синтез на Verilog и VHDL для FPGA/ПЛИС, отладчик проекта.	8-4
VeriBest	VeriBest Synthesis	Синтез и оптимизация на ASIC/ FPGA, технология AT&T/Bell Labs. Библиотека Synopsys. Форматы Verilog, VHDL и LPM.	30
Viewlogic Systems	View Synthesis for Windows	Синтез и оптимизация VHDL для FPGA и ASIC, ввод в формате VHDL, кремниевое проектирование на уровне вентилях.	

Таблица 2 Инструменты проектирования для вентилях матриц (FPGA)

Компания	Продукт	Описание	Цена, тыс.\$
Altera	MAX+PLUS II	Проектирование и верификация для сложных ПЛИС	0.5-8
Cypress Semiconductor	Warp 2/3	Схематический ввод или описание на VHDL, синтез по VHDL описанию, подбор устройств, размещение и связывание, временное моделирование и создание файлов программирования для ПЛИС и FPGA.	0.1 -5
QuickLogic Corp.	QuickWorks	Полная разработка FPGA: ввод схемы, Verilog, VHDL и смешанный способ, логический синтез, автоматическое размещение и связывание, функциональное и временное моделирование	3
VeriBest	VeriBest PLD	Синтез и оптимизация проектов ПЛИС/FPGA (Mips-технология), автоматическое разбиение и генерация модели для моделирования (Verilog/ VHDL).	от 1

Таблица 3 Инструменты анализа

Компания	Продукт	Описание	Цена, тыс.\$
1	2	3	4
Chronology	QuickBench	Генератор тестов для VHDL и Verilog	15

Продолжение таблицы 3

1	2	3	4
Design Acceleration	Signalscan for Windows	Вьювер временных диаграмм и анализатор результатов моделирования для Verilog и смешанного Verilog/ аналогового моделирования	3
Fintronic USA	FinVFI	Процедурный интерфейс доступа к встроенному формату Verilog, генератор и синтезатор кодов для совместного моделирования	7-8,4

Таблица 4 Инструменты схемного ввода

Компания	Продукт	Описание	Цена, тыс.\$
OrCAD	OrCAD Capture for Windows	Работа со схемными проектами на PCB и FPGA/ПЛИС, редактор схем и блоков, 30 форматов, VHDL и Verilog	1
VeriBest	VeriBest Design Capture	Схемное описание проектов, текстовые редакторы для Verilog/VHDL, графические редакторы состояний, диаграмм переходов и таблиц истинности	1

Таблица 5 Системы описания

Компания	Продукт	Описание	Цена, тыс.\$
CAD-Migos Software Tools	The Whole Enchilada for Windows	Система HDL, смешанного моделирования и верификации	2,8-4
R-Active Concepts	BetterState Pro	Проектирование автоматов по диаграммам состояний, генерация кода для VHDL, Verilog HDL, Visual Basic, C, C++, Delphi	0,2-1
Synario Design Automation	Synario Engineering Capture System	Описание структуры, библиотеки с броузерами; интерфейс языка моделирования VHDL, Verilog, TimeMill, SPICE	от 1

## 2. Основные тенденции развития систем моделирования

Системы моделирования являются ключевым инструментом, обеспечивающим защиту от возможного тиражирования ошибок проектирования и оценку выполнения технических требований по функциональности и быстродействию.

Свойства популярных систем моделирования цифровых проектов приведены в табл.6 [3,4].



Таблица 6 Инструменты моделирования

Компания	Продукт	Описание	Цена, тыс.\$
1	2	3	4
Antares Corp.	Model Technology's V-System	Моделирование с прямой компиляцией VHDL, моделирование проектов на VHDL и Verilog.	от 3.5
Exemplar Logic	Galileo Time Explorer	Работа с VHDL и Verilog, моделирование библиотек VITAL, анализ критических путей, статический временной анализ.	От 6
	Galileo V-System VHDL Simulator	Компиляция и моделирование VHDL, поддерживает стандарт IEEE.	от 4
Fintronic USA	FinSim Verilog Simulator	Язык Verilog, расширенные возможности Verilog-XL, поддержка библиотек ASIC/FPGA.	0.9 – 1.3
	FinSim Verilog Simulator/ Signalscan	Язык Verilog, связь с Design Acceleration's Signal-scan, FinSim средства отладки, просмотр временных диаграмм.	5.9-15.3
FrontLine Design Automation	PureSpeed OverDrive	Моделирование на Verilog, компилятор кода, интерпретатор, поддержка представления на уровне вентилях, возможности PLI, SDF и командного языка	9
Model Technology	V-System/ VHDL Windows	Прямой компилятор, интерактивное моделирование и отладка исходного описания. Полностью соответствует стандартам IEEE-1076'93 и VITAL'95.	3,5
OrCAD	OrCAD Simulate for Windows	Моделирование на логическом и временном уровнях FPGA/ ПЛИС, использует стандарты VHDL/ VITAL, EDIF	3,5
Simucad	SILOS III	Моделирование логики и неисправностей на VHDL, проектирование сверху вниз.	от 3
Viewlogic Systems	Speedwave for Windows	Моделирование на VHDL с поддержкой VITAL. Иерархический браузер проекта, отладчик исходного уровня, просмотр временных диаграмм. Моделирование на уровне вентилях и RTL.	
	VCS for Windows	Моделирование в Verilog, просмотр временных диаграмм и отладчик на исходном уровне.	

Продолжение таблицы 6

1	2	3	4
VeriBest	VeriBest Verilog Simulator	Логическое моделирование с идентификацией функциональных и временных результатов в Cadence Verilog. Библиотеки моделей на Verilog..	10

Анализ функций указанных систем [4,5] позволяет выделить тенденции в их развитии.

Уровни цифрового моделирования. Усложнение функциональности проектов и накопление библиотек типовых схемных решений ведет к переориентации симуляторов на поведенческое и RTL-моделирование вместо моделирования вентилей и логических примитивов. Повышаются требования к устойчивости и живучести поведенческих описаний проекта. В ряде случаев происходит объединение инструментов в единой оболочке для совместного моделирования проектов на VHDL и Verilog и смешанного моделирования. Проектирование сетевых контроллеров и сигнальных процессоров часто использует моделирование, основанное на управлении процессами (вместо событийного управления). Единые оболочки позволяют связать эти различные методы моделирования вместе.

Проблема ускорения моделирования. Эта проблема сохраняет актуальность и даже обостряется. Ускорение моделирования в ряде систем достигается за счет компиляции проектов в объектный код, оптимизированный для применяемого компьютера.

Новым является ускорение функциональной верификации проектов путем применения циклических алгоритмов. В отличие от стандартного событийного моделирования, циклические алгоритмы сжимают события, связанные с изменениями сигналов, на очередной задний фронт синхроимпульса, вместо того, чтобы моделировать каждое событие независимо. Это означает, что состояние каждой схемы должно перевычисляться всего один раз в одном цикле синхросигнала. Уменьшение разрешающей способности по времени ускоряет быстродействие в 10 – 100 раз. Это в сочетании с моделированием только в базисе 0,1,X и Z существенно экономит время вычислений и память. Конечно, циклические инструменты предполагают синхронный стиль проектирования и могут иметь другие ограничения..

Относительно новыми являются средства для ускорения моделирования на многопроцессорных рабочих станциях. Процесс моделирования разделяется на множество Unix-поток. Эти потоки разрешают всем доступным процессорам одновременно разделять моделируемую задачу с небольшими накладными расходами.

Расширение отладочных возможностей. Для улучшения диагностики ошибок применяются обратное аннотирование исходного кода проекта данными о временных и энергетических характеристиках, занимаемой площади сгенерированной схемы, указание соответствия между исходным кодом и сгенерированными частями схемы, выделение в исходном коде на HDL строк с ошибками.

Развитие графических интерфейсов проектировщика. Подобно визуальным системам программирования Delphi и Visual Basic создаются советчики, обеспечивающие создание текста HDL проекта на основе выбора из

типичных вариантов. Применяются графические программы просмотра и анализа больших проектов и унифицированные интерфейсы для различных операционных систем (Windows/Unix), облегчающие работу в смешанных средах проектирования

Интегрированные системы. Расширение функциональных возможностей систем моделирования достигается подключением через "С"-интерфейсы пользовательских моделей и выполняемого программного кода для совместного проектирования аппаратуры и программного обеспечения. Применяются также прикладные процедурные интерфейсы для связи с другими инструментами (подключение калькуляторов задержек, совместное моделирование проектов, описанных на различных языках).

Стандартизация. Внедряются новые IEEE-стандарты VITAL для описания переносимых библиотек высокого качества для программируемых СБИС. Этот стандарт определяет программируемые примитивы, временные ограничения, стиль моделирования, обеспечивающий для оптимизированных симуляторов ускорение в 10-20 раз. Расширяется применение стандартного формата SDF файла описания задержек для VHDL или Verilog симуляторов. Разработка стандартных API для вычисления задержек подчеркивает растущую важность временной точности при моделировании.

### **Заклучение**

Рост сложности перспективных проектов цифровых устройств опережает рост производительности средств автоматизации проектирования и логического моделирования. Поэтому расширяется применение стандартных языковых средств высокоуровневого описания для обеспечения преемственности, переносимости и совместимости проектов.

Расширяется круг автоматизируемых программными средствами функций проектирования, наблюдается тенденция увеличения спектра программ, обеспечивающих совместное использование различных инструментов.

Сохраняет свою актуальность проблема ускорения логического моделирования. Перспективным направлением ее решения является совершенствование алгоритмов управления модельным временем.

Мало исследованы и практически редко реализуются перспективные параллельные средства логического моделирования с применением многопроцессорных систем и компьютерных сетей.

### **Литература**

1. Ладьяженский Ю.В., Долгополов Д.Г. Высокопроизводительные цифровые устройства с программируемой логикой. Аналитический обзор / Информатика, кибернетика и вычислительная техника (ИКВТ-97). Сборник научных трудов Донецкого государственного технического университета. Выпуск 1. Донецк: ДонГТУ, 1997 – с. 177-122
2. R.T. Maniwa. Focus Report: HDL Add-In Tools, April 1997: <http://www.isdmag.com>
3. Tets Maniwa. Focus Report: Windows EDA Tools, April 1996: <http://www.isdmag.com>
4. Electronic design automation for Windows: a user's guide / David Pellerin., Prentice Hall PTR, Englewood Cliffs, New Jersey, 1995 – 500p.
5. Steven E. Schulz. Trends In Simulation and Synthesis. Focus Report: September 1995: <http://www.isdmag.com>