

2. Соболев И.М., Выбор оптимальных параметров в задачах со многими критериями. - М.: Наука, 1981. - 101с.
УДК 681.3

ВСТРОЕННЫЕ СИСТЕМЫ VHDL/FPGA

Матрунчик Ю.Н., Гутич И.И., Захаревич А. А.

Белорусский национальный технический университет
Минск, Республика Беларусь

Целью проекта является создание образовательного демонстратора, который подчеркивает различные варианты платы Basys3 и для решения проблем с внутренним выделенным оборудованием FPGA Xilinx Artix 7 с использованием VHDL и IP-генератором программного обеспечения Vivado от Xilinx. Демонстратор использует плату Basys3, VGA-выход платы и USB-клавиатуру с USB разъемом платы. Цель состоит в том, чтобы создать фон с VGA и иметь две отдельные фигуры, так называемые спрайты, на экране которых можно перемещать отдельно с помощью клавиш клавиатуры USB. В проекте используется интерфейс PS2 между ПЛИС и микроконтроллером PIC, который преобразует вывод USB-клавиатуры в выход PS2. Он использует Multimedia Clock Manager (MMCM), чтобы построить устойчивую частоту 25 МГц. Для синхронизации VGA он использует Block RAM для хранения двух спрайтов и использует VGA для адресации VGA-экрана.

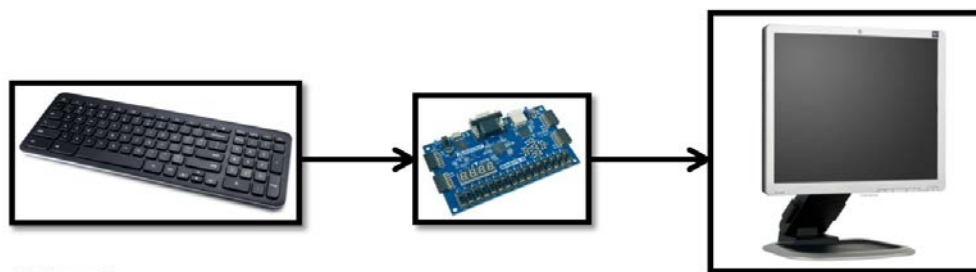


Рис. 1 - Внешняя блок-схема: VGA-экран, стандартная клавиатура USB и плата Basys3

VHDL расшифровывается, как Very high speed Integrated Circuit Hardware Description Language. HDL – это язык описания аппаратуры. Т.е. основная задача таких языков – описывать электрические цифровые цепи подобно графическим схемам. Традиционные языки программирования не пригодны для описания аппаратуры, т.к. они создавались для описания алгоритмов, а это совсем не то же самое, что цифровая цепь.

Языки программирования характеризуются синтаксисом и семантикой. Синтаксис определяет грамматические правила написания программ и использования конструкций языка, а семантика – значение этих конструкций. Когда создается новый язык, он создается для описания конкретных процессов, со своим синтаксисом и семантикой. Программы,

написанные на традиционных языках программирования, моделируют последовательный процесс. Так как результат выполнения отдельных операторов часто влияет на последующие операторы, то нельзя менять их местами. У такого подхода есть два плюса:

1. Так проще создавать алгоритм, потому что человек мыслит последовательно.

2. Последовательный алгоритм легче переложить на машинные инструкции, потому что команды выполняются последовательно на процессоре.

Такие языки не могут описывать цифровые схемы, потому что последние состоят из большого количества мелких компонентов, сигналы через которые распространяются параллельно. При этом на каждом компоненте сигналы просто испытывают некоторую задержку. Для описания таких систем требуется язык с другой семантикой. Операторы такого языка должны выполняться одновременно (или параллельно). Т.к. операторы выполняются параллельно, то результат их выполнения не зависит от их места в коде относительно друг друга. Программа на языке VHDL не может быть «выполнена» на компьютере, подобно программам на языке C. В отличие от них VHDL программа решает другие задачи:

1. Формально описать цифровую схему, т.е. описать ее интерфейс и функциональное назначение в форме понятной человеку (подобно рисунку)

2. На основе VHDL программы можно провести моделирование работы цифровой схемы на компьютере. Т.е. подать на входы схемы воздействия и последить за выходами.

3. На основе VHDL программы создается реальное цифровое устройство. Подбираются реальные компоненты и создаются связи между ними.

Фундаментальные свойства цифровой системы описываются понятиями: модульность, связность, одновременность и продолжительность. Модульность означает, что любая схема может быть разбита на некоторые модули со входами и выходами. При этом сама эта схема может быть представлена как один большой модуль. Связность означает, что эти модули связаны друг с другом посредством сигналов. Сигналы распространяются одновременно. При этом каждый элемент цифровой схема (или модуль) характеризуется временем распространения сигнала через этот элемент. Т.е. каждый элемент схемы вносит некоторую задержку в распространение сигнала. Семантика языка VHDL включает в себя эти понятия.

1. Digital Design Using Digilent FPGA Boards *Richard E. Haskell / Darrin M. Hanna* LBE Books – Third Edition, 2014
2. <http://www.allaboutcircuits.com/textbook/digital/chpt-4/contact-bounce/>

3. https://eewiki.net/pages/viewpage.action?pageId=28278929&preview=28278929/28508225/ps2_keyboard.vhd#PS/2KeyboardInterface%28VHDL%29-CodeDownloads

УДК 004.04

АВТОМАТИЗАЦИЯ ПРОЦЕССА ТЕСТИРОВАНИЯ ОБУЧАЮЩЕЙ СИСТЕМЫ CATS

Попова Ю.Б., Серeda П.В.

Белорусский национальный технический университет
Минск, Республика Беларусь

На кафедре программного обеспечения вычислительной техники и автоматизированных систем БНТУ разработана и используется обучающая система CATS (англ., Care About The Students), расположенная в локальной сети университета по адресу [http://172.16.11.72:2020]. Поскольку система находится в постоянном расширении функциональных возможностей при одновременной работе нескольких участников проекта, то возникает проблема постоянной интеграции новых частей, а также их тестирование. Следует добавить, что при появлении новых функциональностей необходимо проводить регрессионное тестирование с целью проверки отсутствия повреждений старых частей при добавлении новых.

Для решения данной проблемы авторы применили практику непрерывной интеграции посредством инструмента Jenkins. Схема последовательности проведения автоматизации тестирования LMS (англ., Learning Management System) CATS приведена на рисунке 1. Здесь на этапе 1 Пользователь обращается к Jenkins с целью собрать LMS. На этапе 2 происходит скачивание исходного кода LMS из удалённого репозитория по сети Интернет.

