

Исследование методов создания эмуляторов периферийных устройств для удаленной отладки проектов в FPGA-лаборатории

В.С. Ленч (КСМ-15 КИ), Ю.Е. Зинченко к.т.н. доц., А.В. Коваль асс., Зинченко Т.А. асс.

Донецкий национальный технический университет
lenchvov@rambler.ru

Ленч В.С., Зинченко Ю.Е., Коваль А.В., Зинченко Т.А. Исследование методов создания эмуляторов периферийных устройств для удаленной отладки проектов в FPGA-лаборатории. В данной статье описываются методы организации удаленного управления периферийными устройствами, подключенных к отладочному FPGA-стенду, методы организации передачи, ввода и вывода информации. Также рассматривается схема взаимодействия пользователя с FPGA-лабораторией и проблемы многопользовательского управления комплексом в целом.

Ключевые слова: периферийные устройства, FPGA, удаленный доступ, websockets, виртуальная FPGA-лаборатория.

Введение

FPGA (Field Programmable Gate Arrays) широко применяются для моделирования микроэлектронных устройств и их детального исследования и отладки [1-4], поэтому так важно студентам специальностей, связанных с компьютерной инженерией, иметь навыки работы с FPGA разработкой. Но для детального освоения навыков работы с FPGA и получения практического опыта студентам необходима инфраструктура. Высокая стоимость не позволяет студентам иметь собственные отладочные стенды, а ВУЗы не имеют возможности приобретать их достаточное количество, чтобы обеспечить беспрепятственную работу студентам во время учебных занятий. Поэтому одним из выходов из данной ситуации является разработка удаленного доступа к FPGA-лаборатории посредством технологии Интернет, что позволило бы студентам отлаживать свои проекты в любое удобное для них время, и более детально углубляться в разработку и исследование электронных устройств.

Одной из проблем удаленной отладки на FPGA-стендах является отсутствие доступа к подключенным периферийным устройствам ввода/вывода. Именно этой проблеме и посвящена данная статья.

Постановка задачи

Вопрос решения проблемы удаленного доступа к FPGA комплексам уже поднимался в других работах [5-7], где был продемонстрирован определенный успех, но в этих трудах решалась задача организации управления периферийными устройствами, такими как мышь, клавиатура и т.п. Чаще всего авторы ограничивались светодиодами и переключателями. Попытка решения такой задачи предпринималась студентами ДонНТУ [8, 9, 12], но в целом решение достигнуто не было. Также ранее был проведен обзор текущей проблемы и было выбрано направление ее решения [10, 11]. Данная статья описывает конкретно решение проблемы эмуляции периферийных устройств, для обеспечения прозрачного управления платой и тестирования своих проектов, использующие подобные устройства.

С помощью среды разработки Xilinx ISE Design Studio студенты имеют возможность собирать свою систему на кристалле (СнК), добавляя в нее необходимые периферийные компоненты, которые потом можно программировать. Далее собранное устройство и программа загружаются в FPGA плату, и она тестируется при помощи подключенных физических устройств. В ядре СнК в FPGA-стендах компании Xilinx располагаются спроектированные микропроцессор Microblaze или микроконтроллер Picoblaze. Цель работы заключается в том, чтобы пользователи могли создавать свои программы для СнК и могли выполнять их как в удаленной Web-среде, так и при непосредственном доступе к FPGA-плате, без каких-либо изменений в программном коде. Также пользователь, при использовании Web-среды, не должен ощущать трудностей при тестировании подключенных периферийных устройств.

Организация удаленного доступа

Управление платой структурно организовано в виде трех основных частей: клиентская, серверная и FPGA-стенд. Структурная схема представлена на рисунке 1.

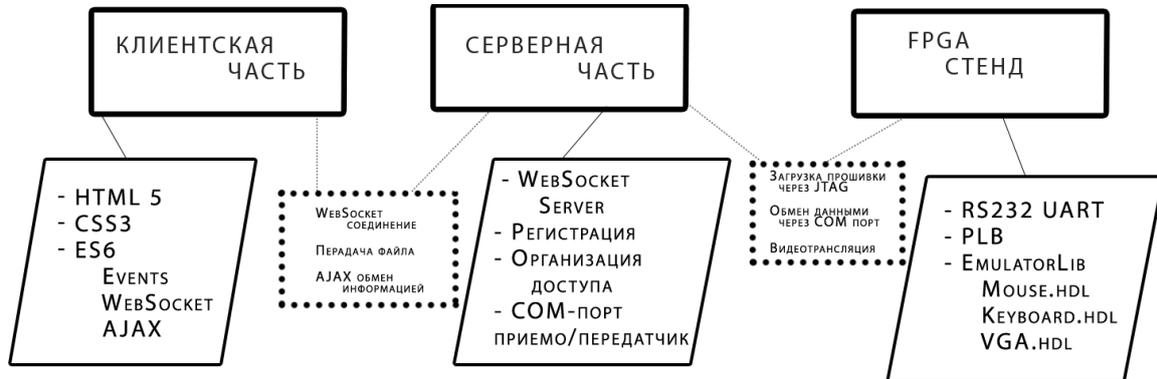


Рисунок 1 – Структура организации удаленного доступа

Управление платой реализуется при помощи Web-интерфейса через интернет соединение, поэтому клиентская (визуальная) часть создается при помощи технологий HTML, CSS и JavaScript. Клиентская часть планируется состоять из главной страницы, которая должна презентовать виртуальную FPGA-лабораторию, обратную связь, регистрацию и сам кабинет виртуальной лаборатории. Доступ в виртуальный кабинет должны иметь только пользователи с особыми разрешениями, которыми может наделить только администратор сайта. Структура интерфейса планируется быть максимально простой и интуитивно понятной. Все запросы к серверной части в виртуальном кабинете должны производиться без перезагрузки страницы, это возможно при помощи технологии AJAX (Asynchronous Javascript and XML). Также за платой должно быть наблюдение в реальном времени при помощи видеотрансляции. В текущей реализации видеотрансляция на клиентской части организована при помощи HTML5 тега video. На стороне сервера видеотрансляцию производит программа VLC media player. Имеется также возможность вместо видео запрашивать снимок с камеры в реальном времени.

Личный кабинет виртуальной лаборатории оснащен упрощенным файловым менеджером, который позволяет управлять выделенным пространством на сервере для загрузки файлов прошивок. В файловом менеджере есть возможность переименовывать файлы и папки, создавать папки, загружать файлы прошивок, инициировать запись прошивки в плату. Для того, чтобы во время работы с платой другие пользователи не могли загружать в нее свои проекты, необходимо обеспечивать контроль над сессией. Если плата занята, не позволять загружать новые файлы прошивок, но если пользователь, работающий с платой, не проявляет активности некоторое время, то необходимо освобождать плату автоматически.

Структура виртуального личного кабинета приведена на рисунке 2.



Рисунок 2 – Структура виртуально кабинета лаборатории

Основная часть, которой следует уделить особое внимание, это организация прозрачного управления платой. Обмен данными с клиентом по протоколу HTTP реализован таким образом, чтобы клиент запрашивал данные, а сервер их отдавал. Для эмуляции периферийных устройств такая схема

работы не подходит, потому что необходимо в минимальные промежутки времени передавать и принимать данные без перезагрузки страницы. Для решения этой проблемы можно применять два подхода. Первый подход заключается в использовании технологии Long Polling. В основе этой технологии лежит использование фоновых AJAX запросов в короткие промежутки времени, то есть клиент запрашивает постоянно данные у сервера, и если данные есть, то получает соответствующий ответ. Передача данных в таком случае происходит по запросу пользователя. Минусом такой технологии является большая нагрузка на сервер за счет множества запросов и синхронность обмена, что сказывается на производительности. Плюсом является высокая поддержка современными браузерами.

Второй подход заключается в использовании протокола Websocket. Этот протокол позволяет производить обмен с сервером в реальном времени. На сегодняшний день эту технологию поддерживают все современные браузеры, поэтому на стороне клиента технология обладает всеми преимуществами перед Long Polling. Трудности возникают на стороне сервера. Websocket требует отдельной программы сервера, который бы обрабатывал запросы, а также Websocket-сервер не может работать на том же порте, что и веб сервер. Websocket-сервер в текущем проекте реализован при помощи PHP cli скрипта, который постоянно слушает соединение с сокетом и COM-портом.

Плата подключена к серверу через COM-порт, и именно через этот интерфейс производится обмен данными. Так как стандартная библиотека PHP не поддерживает работу с COM-портом, необходимо использовать внешнюю программу, которая бы «слушала» изменения на этом интерфейсе и передавала данные на Websocket-сервер, а сервер уже передавал данные клиенту. Именно по такой схеме и реализован обмен в текущей реализации проекта. Связующим звеном между Websocket-сервером и «слушателем» COM-порта является стандартный буфер ввода/вывода ОС (stdout/stdin).

Загрузка прошивки на плату организована с помощью batch-скрипта, который запускается по запросу пользователя. На FPGA-стенде все операции ввода/вывода должны перенаправляться на порт RS232. Для этого необходимо разработать эмуляторы периферийных устройств, которые должны принимать данные с порта RS232 и перенаправлять их на шину process local bus (PLB), а также ответ с шины передавать в порт RS232.

Эмуляция периферийного устройства

Из описанного выше вытекает то, что эмулятор периферийного устройства должен быть задействован на всех трех каскадах структуры виртуальной лаборатории (рисунок 3).

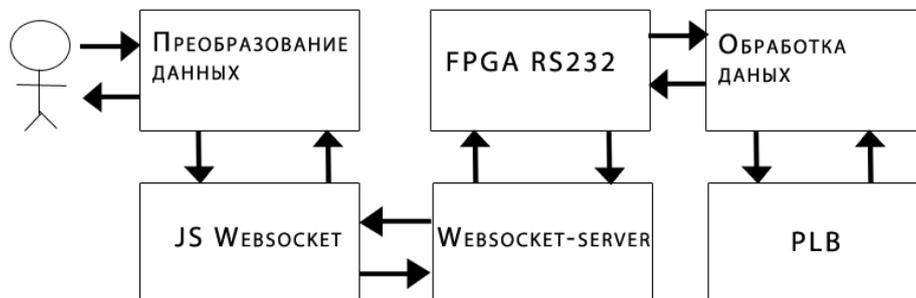


Рисунок 3 – Жизненный цикл одного действия эмулятора периферийного устройства

В случае эмуляции устройств ввода данные, которые вводит клиент в виртуальном кабинете (например, нажатие клавиш, или координаты движения мыши), необходимо преобразовать в пакеты, которые аналогичны пакетам данных устройства. Далее эти пакеты данных должны передаваться через Websocket-соединение на сервер. Сервер передает данные на COM-порт, после чего эти данные принимает FPGA-стенд, где основная программа-эмулятор обрабатывает данные в соответствии со стандартами периферийных устройств и передает дальше на шину PLB. Далее данные может получить программа, написанная для СнК и использовать их в собственных целях. После, с шины PLB, может прийти ответ, который передается обратно клиенту через сервер, где уже javascript-сценарий для конкретного периферийного устройства должен принять решения о последующих действиях.

В случае эмуляции устройств вывода, например, VGA монитора или LCD дисплея, на клиентской стороне можно использовать такую технологию HTML5 как canvas. В таком случае инициатором служит FPGA-плата, которая передает данные клиенту, а на стороне клиента должен работать сценарий, который обрабатывает принятые данные и выводит их в соответствующем виде. При таком подходе FPGA-эмуляторы должны передавать свой уникальный идентификатор, чтобы javascript-сценарий мог идентифицировать устройство, и обработал принятые данные соответствующим образом.

Проблема многопользовательского доступа и производительности

COM-порт как приемо-передатчик был выбран из-за простоты работы. Но использование такого метода передачи информации накладывает некоторые ограничения. Одним из таких ограничений является использование одного сервера на одну плату, это обусловлено аппаратным обеспечением персонального компьютера. Эту проблему можно было бы разрешить, используя в качестве приемо-передатчика порт RJ-45 и Ethernet-протокол. В таком случае, каждая плата служила бы отдельным сетевым устройством, которое могло бы подключаться через сетевой концентратор к серверу, и можно было бы клиенту выбирать свободное устройство и работать непосредственно с ним.

Также передача данных через COM-порт обладает низкой пропускной способностью, что может создать ограничения при обмене большими объемами информации. Эту проблему также решает Ethernet протокол. Цель текущей работы создать эмуляцию периферийных устройств, но созданная структура взаимодействия позволяет в перспективе переделать любой из способов передачи информации, так как структура пакетов обмена изменяться не будет.

Литература

1. Зинченко Ю., Калашников В., Хайдук С. и др. FPGA-технологии проектирования и диагностика компьютерных систем / Сборник научных трудов VI Междунар. научн.-практ. конф. «Современные информационные технологии и ИТ-образование». - Москва: МГУ, 2011. - Т. 1. 787 С. 422-429. <http://conf.it-edu.ru/conference/2011/works>
2. Зинченко Ю., Дяченко О. и др. Новые технологии проектирования в ДонГТУ/ Материалы междунар. научно-техн. конф. "Новые информационные технологии в САПР и АСУ". Киев.- 2001.- С 15-19.
3. Медведев К.В., Зинченко Ю.Е. Навигационная система на базе FPGA-технологий с возможностью самотестирования // Сборник трудов VII международной научно-технической конференции студентов, аспирантов и молодых ученых Информатика и компьютерные технологии – 2011. – Донецк: ДонНТУ – 2011.
4. Волошин Д.Н., Зинченко Ю.Е. Разработка системы реального времени на базе Xilinx XC3S500E Spartan-3E FPGA // Сборник трудов VII международной научно-технической конференции студентов, аспирантов и молодых ученых Информатика и компьютерные технологии, 22-23 ноября 2011 г., Донецк: ДонНТУ. – 2011. В 2-х томах, Т. 1. – с. 44-47.
5. Rusen J., Kolberg S. Online FPGA laboratory for interactive digital design //International Conference on Engineering Education 2004 (ICEE2004). – 2004. – С. 16-21. https://www.researchgate.net/publication/237336992_Online_FPGA_laboratory_for_interactive_digital_design
6. A Remote FPGA Laboratory for Digital Design Students. Авторы: Joao Soares, Jorge Lobo. Режим доступа: <http://ap.isr.uc.pt/archive/jlobo-REC2011.PDF>
7. Лаборатория Электронных Средств Обучения (ЛЭСО) СибГУТИ. Интернет ресурс. Режим доступа: <http://www.labfor.ru/online/fpga>
8. Персональный сайт на портале магистров ДонНТУ. Автор: Беседа Д. Г. 2014 г. Интернет ресурс. Режим доступа: <http://masters.donntu.org/2014/fknt/beseda/diss/index.htm>
9. Зинченко Ю.Е., Зинченко Т.А., Асланов И.Г. Виртуальная FPGA-лаборатория с дистанционным доступом // Современные компьютерные информационные технологии: Материалы III Всеукраинской школы-семинара молодых ученых и студентов АСИТ'2013. – Тернополь: ТНЕУ, 2013 <http://ea.donntu.org/handle/123456789/19624>
10. Персональный сайт на портале магистров ДонНТУ. Автор: Ленч В. С. 2016 г. Интернет ресурс. Режим доступа: <http://masters.donntu.org/2016/fknt/lench/diss/index.htm>
11. В.С. Ленч, В.А. Гень, Ю.Е. Зинченко, А.В. Коваль. Исследование организации удаленного доступа к fpga-лаборатории // Международная научно-техническая конференция студентов, аспирантов и молодых ученых «Компьютерная и программная инженерия – 2015», Донецк, 15-16 декабря 2015г., с. 43. <http://cs.conf.donntu.org/docs/sbornik2015.pdf> (Дата обращения 29 октября 2016г.)
12. Беседа Д.Г., Зинченко Ю.Е. Разработка метода эмуляции работы FPGA-платы с клавиатурой через порт PS/2 // Сборник трудов V Всеукраинская научно-техническая конференция студентов, аспирантов и молодых ученых, Информационные управляющие системы и компьютерный мониторинг (ИУС КМ-2014), 22-23 апреля 2014 г., г. Донецк : – Донецк: ДонНТУ, 2014 - т.1 - С. 157-162

Ленч В.С., Зинченко Ю.Е., Коваль А.В. Исследование методов создания эмуляторов периферийных устройств для удаленной отладки проектов в FPGA-лаборатории. Рассмотрен масштабируемый способ организации удаленного управления FPGA-стендом на основе трех каскадной модели взаимодействия пользователя с конечным устройством, которая позволяет студентам моделировать и отлаживать собственные системы на кристалле. Разработан способ отладки периферийных устройств на плате. Описаны возможные проблемы при организации удаленного доступа к FPGA-лаборатории.

Ключевые слова: *FPGA-лаборатория, эмулятор периферийных устройств, удаленное управление, websockets, PLB*

Lench V.S., Zinchenko Y.E., Koval O.V. Research of methods creation emulator peripheral devices for remote debug in FPGA-lab. *Considered a scalable method of organization remote manage FPGA based in three-cascade model of user interaction with end device, which allows for students simulate and debug own system-on-chip. Developed a method of debug peripheral devices on the board. Considered problems with the organization of remote access to the FPGA-lab.*

Keywords: *FPGA, FPGA KIT, Spartan-3E, SoC, Microblase, Picoblase, PLB, FPGA Remote Control, Peripheral emulator, websockets,*