

ОСОБЕННОСТИ РАБОТЫ С ПЛАТОЙ COSMOS+ OPENSSED ПРИ ПЕРЕНОСЕ НА СОВРЕМЕННЫЕ АППАРАТНЫЕ И ПРОГРАММНЫЕ ПЛАТФОРМЫ

Зайцев И.Д.

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Заливако С.С. – канд. техн. наук, доцент

В работе были рассмотрены проблемы, возникающие при переносе проекта Cosmos+ OpenSSD на актуальную в настоящее время версию инструментов фирмы Xilinx (Vivado 2020.1, Vitis 2020.1). В Xilinx Vivado была рассмотрена проблема несоответствия частот IP-ядер после их обновления. В Xilinx Vitis была рассмотрена проблема добавления новой аппаратной платформы и выбора корректных исходных файлов FTL (Flash Translation Layer). Отдельно была произведена сборка хоста и настройка его операционной системы (ОС).

В настоящее время твердотельные накопители информации (NAND Flash SSD – Solid State Drive) широко применяются во многих компьютерных системах наряду с традиционными накопителями на жестких магнитных дисках (HDD – Hard Disk Drive). Твердотельные накопители в сравнении с традиционными жесткими дисками имеют ряд преимуществ: более быстрое время доступа, меньшее энергопотребление и меньшую массу. Также стоит отметить, что из-за отсутствия движущихся элементов твердотельные накопители имеют лучшую устойчивость к вибрациям. К недостаткам можно отнести: более высокую стоимость на единицу хранения информации, уменьшенную пропускную способность во время выполнения процедур обслуживания диска (например сборки мусора, выравнивания износа и т. п.), а также большее ограничение на число циклов стирания для каждого блока устройства [1].

SSD устройства поддерживают три основные операции: чтение, запись и стирание. Операции чтения и записи работают в пределах страницы, в то время как операция стирания работает в пределах блока (блок является совокупностью страниц) [2]. Каждая страница в SSD имеет одно из трех доступных состояний: действительная (valid), недействительная (invalid) и свободная/стертая (free/erased). Операция записи может быть применена только к странице, находящейся в стертом состоянии.

Современное встроенное программное обеспечение должно включать в себя таблицу соответствия адресов (L2P – Logical to Physical), сборщик мусора (GC – Garbage Collector), выравниватель износа (WL – Wear-leveler) и т. д. Таким образом, особенности устройства NAND Flash памяти требуют создания дополнительных модулей, что приводит к усложнению разработки микропрограммного обеспечения современных SSD устройств. В связи с этим при разработке используют готовые платформы, с помощью которых осуществляется прототипирование аппаратных решений и программных алгоритмов.

Примером такой платформы служит проект Cosmos+ OpenSSD [3] – проект с открытым исходным кодом, который позволяет проводить исследования в области SSD технологий. Проект является модульным, т. е. программные и аппаратные компоненты являются взаимозаменяемыми. В качестве отладочной платы используется Cosmos/Cosmos+ FPGA Platform (FPGA – Field-Programmable Gate Array), на которой происходит проектирование аппаратуры микроконтроллера

SSD. Работа с проектом осуществляется при помощи программного обеспечения компании Xilinx. Данный проект изначально был разработан для версии Xilinx Vivado 2014.4. В связи с этим для переноса проекта на современную версию Xilinx Vivado 2020.1 требуются модификации аппаратного и программного обеспечения.

Ядром отладочной платы является система на кристалле Zynq-7000, содержащая программируемую логическую интегральную схему XC7Z045-FFG900-3. Вычислительные единицы представлены двумя ядрами ARM Cortex – A9. Оперативная память типа DDR3 объемом 1 Гб. Модуль памяти представляет собой печатную плату форма-фактора SO-DIMM, состоящую из 8 микросхем NAND Flash памяти. Каждая микросхема объемом 128 Гб. Размер страницы составляет 18048 байт. Размер запасной области (spare area) – 1664 байта. Проект Cosmos+ OpenSSD поддерживает 2 модуля памяти.

Целью данной работы является анализ и обобщение изменений для переноса проекта на актуальную версию средств разработки Xilinx. Опишем особенности трех этапов работы над проектом: сборки аппаратной платформы, создания проекта и загрузки промежуточного микропрограммного обеспечения, а также сборки хоста и настройки его ОС. Для работы над проектом была выбрана среда автоматизированного проектирования Xilinx Vivado 2020.1 и среда разработки промежуточного микропрограммного обеспечения Xilinx Vitis 2020.1. Необходимо было также удостовериться, что материнская плата хоста поддерживает NVMe протокол [4]. Для Windows 7 необходимо скачать OpenFabrics Alliance (OFA) NVMe драйвер. В свою очередь, Windows 8/8.1 и Windows 10 поддерживают NVMe протокол по умолчанию.

На первом этапе производится сборка проекта аппаратного обеспечения контроллера твердотельного накопителя OpenSSD. В связи с тем, что IP-ядра проекта нуждаются в обновлении, тактовую частоту необходимо настроить вручную. Для этого необходимо выбрать блок ZYNQ7 Processing System (5.5) и выставить частоты, как показано на рисунке 1. Перед началом моделирования необходимо провести валидацию дизайна (Design Validation). После стадии синтеза (Synthesis) и размещения ресурсов FPGA в логические ячейки и трассировки (Implementation) конфигурационный файл FPGA сгенерируется корректно. В папке проекта Xilinx Vivado появится файл с расширением *.xsa. На этом этапе также возможно изменить существующую или создать свою аппаратную платформу контроллера твердотельного накопителя.

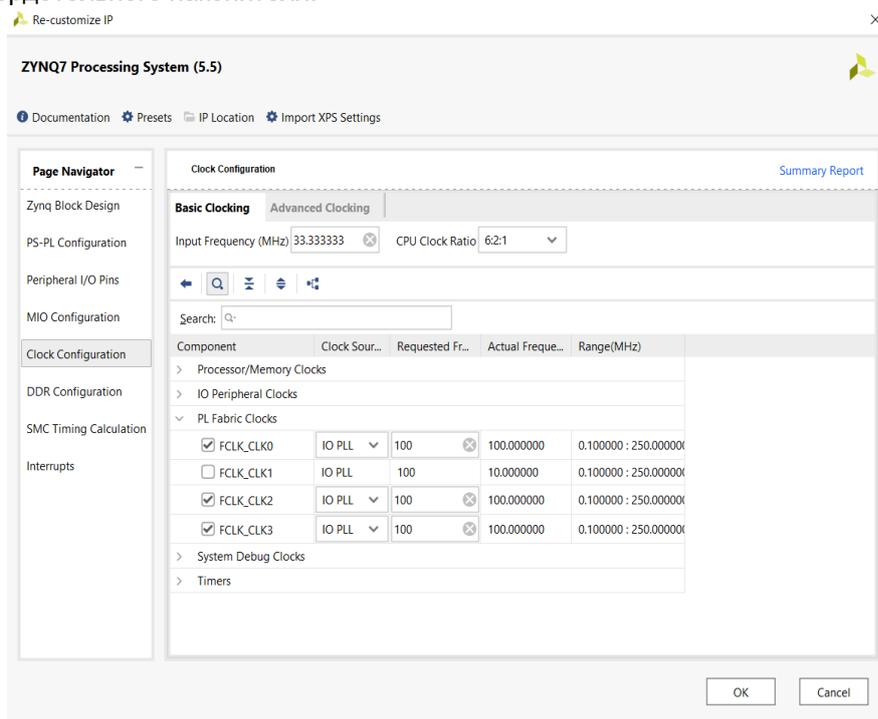


Рисунок 1 – Корректировка частот в IP-компоненте ZYNQ7 Processing System

На втором этапе происходит работа с микропрограммным обеспечением. В среде Xilinx Vitis необходимо создать новый проект (Create Application Project), добавить новую аппаратную платформу (во вкладке Create a new platform from hardware (XSA) нужно будет выбрать файл с расширением *.xsa в папке проекта Xilinx Vivado). После создания проекта необходимо добавить исходный код на языке C из проекта GreedyFTL-3.0.0. FTL представляет собой программный слой, который обслуживает запросы записи и чтения хоста, а также реализует дополнительные процедуры (GC, WL и т. д.). На этом этапе возможно редактировать или создавать свои модули для микропрограммного обеспечения. Это может быть необходимо для экспериментов с разработанными алгоритмами различных процедур работы диска. После второго этапа происходит загрузка конфигурационного файла в память FPGA и микропрограммного обеспечения во Flash память.

57-я Научная Конференция Аспирантов, Магистрантов и Студентов БГУИР, Минск, 2021

На третьем этапе, после подключения платы к компьютеру, в диспетчере устройств должен появиться накопитель с названием NVMe Cosmos+ OpenSSD. Далее необходимо открыть панель управления → администрирование → управление компьютером → управление дисками, выбрать накопитель и создать составной том. В результате появится диск, с которым можно работать. Дальше необходимо установить программу теста производительности (benchmark) и осуществлять оценку производительности полученного накопителя.

В ходе проделанной работы были рассмотрены проблемы переноса проекта Cosmos+ OpenSSD на новую версию инструментов фирмы Xilinx. Для запуска проекта в новых версиях необходимо было провести корректировку частот, провести валидацию дизайна и сгенерировать конфигурационный файл FPGA в Xilinx Vivado. В Xilinx Vitis была добавлена новая аппаратная платформа и исходные файлы FTL. В конце была произведена сборка и настройка хоста. В результате чего был настроен твердотельный накопитель, который корректно отображается в ОС и в дальнейшем может быть использован для оценки производительности аппаратных решений и программных алгоритмов.

Список использованных источников:

1. Tavakkol, A. MQSim: A Framework for Enabling Realistic Studies of Modern Multi-Queue SSD Devices / A. Tavakkol [et al.] // 16th USENIX Conference on File and Storage Technologies, Oakland, USA, Feb. 12–15, 2018 / USENIX Association. – 2018. – P. 49–65.
2. Van Houdt, B. Performance of garbage collection algorithms for flash-based solid state drives with hot/cold data / B. Van Houdt // Performance Evaluation. – 2013. – Vol. 70, iss. 10. – P. 692–703.
3. OpenSSD: open-source solid-state drive project for research and education [Электронный ресурс]. – Режим доступа : <http://www.openssd.io/>. – Дата доступа : 04.04.2021.
4. NVMe specification [Электронный ресурс]. – Режим доступа : <https://nvmexpress.org/>. – Дата доступа : 04.04.2021.