

ВЫСОКОЭФФЕКТИВНАЯ СИСТЕМА ПЕРЕДАЧИ ДАННЫХ НА ПЛИС

Предложен эффективный способ построения высокоскоростного канала передачи данных на основе применения ПЛИС. Канал основан на использовании модулей 10 Gigabit Ethernet. Увеличение скорости передачи данных достигается распределением данных по нескольким физическим линиям с последующим их объединением в системе записи. Предложенное решение реализовано на базе ПЛИС virtex5-240 (XC5VSX240T) фирмы Xilinx с целью создания канала связи с пропускной способностью более 25 Гбит/с.

Введение

В связи с развитием инструментов анализа и обработки данных постоянно повышаются требования к каналам передачи информации. Одно из основных требований связано с увеличением скорости передачи больших объемов данных, генерируемых в реальном времени. В настоящее время это требование является актуальным, например, при создании аппаратных решений для передачи оцифрованных данных от АЦП на системы записи, объединении сегментов высокопроизводительных кластеров компьютеров. Для решения этих задач обычно используются различные стандарты FiberChanel, InfiniBand, Ethernet 100BASE, 1000BASE, 10GBASE [1], которые обеспечивают среднюю скорость передачи 1 – 1000 Мбит/с на канал. Вместе с тем, существует ряд задач, где требуется реализовать гораздо более высокие скорости при относительно невысоких затратах. Особенно остро эти задачи стоят при разработке систем обработки и записи данных широкополосных приемников.

Существуют исследования способов передачи цифровых потоков данных по волоконно-оптическим линиям связи с использованием солитонных импульсов [2]. Такой подход позволяет достигнуть скорости передачи данных до 160 Гбит/с по одной линии, но имеет целый ряд недостатков; кроме того, способы реализации таких методов пока не предложены.

Таким образом, на текущий момент построение каналов передачи данных со скоростями свыше 10 Гбит/с в реализации сложны и дороги.

Цель данной работы – создание системы передачи и записи данных для широкополосного анализатора сигналов. Частью этой системы является модуль для ПЛИС, который обеспечивает одновременную передачу по нескольким 10GBASE-CX4 Ethernet каналам суммарного потока данных свыше 3.5 Гбайт/с. В данном проекте было реализовано аппаратное распределение исходного потока данных на несколько независимых каналов, а на приемной стороне обеспечена их программная сборка в общий поток и запись. В качестве физической среды передачи использовались стандартные медные кабели 10GBASE-CX4, что позволяло передавать данные на расстояния до 15 м. В то же время при необходимости возможно использование активных оптических кабелей стандарта 10GBASE-CX4, что позволяет увеличить расстояние до 150 – 300 м и более. С целью уменьшения стоимости система записи строилась с использованием стандартных компьютерных компонентов компании SuperMicro.

Блок передачи данных был выполнен как неотъемлемая часть вычислителя, реализованного на 2х ПЛИС Xilinx Virtex5 240 t. Такой подход позволил упростить передачу данных от источника, представленного системой из нескольких АЦП, и уменьшить число необходимых аппаратных модулей в системе, что обеспечивает также снижение ее цены, уменьшение времени тестирования и повышение надежности.

В следующих разделах подробно описываются соответствующие решения, алгоритмы, реализация и результаты тестирования предложенного блока передачи данных.

Модуль контроллер Ethernet пакетов

Данный модуль был реализован на ПЛИС для обеспечения приема, обработки и сортировки Ethernet пакетов. Сортировка необходима для разделения управляющих пакетов и пакетов данных ввиду того, что данные и управление реализованы независимо.

Структура приемника Ethernet пакетов приведена на рис. 1. Приемник включает в себя три основных блока: приема и предварительной обработки, передачи пакетов в канал DMA и управления. Блок приема и предварительной обработки данных реализован на основе трех типов модулей, указанных на рис. 1: parser, CrcSum и PktFIFO.

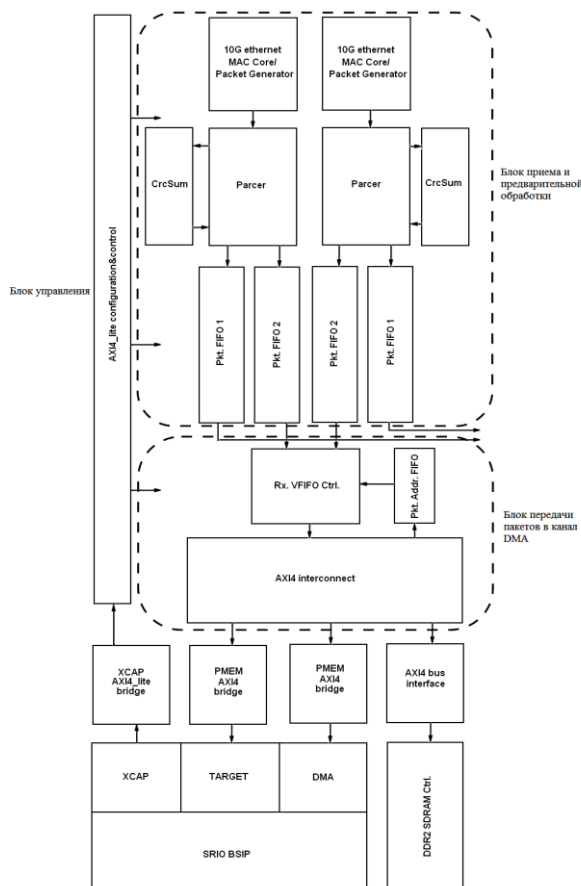


Рис. 1. Укрупненная блочная структура приемника Ethernet пакетов

Данные принятых сетевых пакетов поступают в модуль parser. Этот модуль был разработан для классификации данных, содержащихся в пакете, и для направления пакета в соответствующий канал данных или его сброса. Классификация пакетов производится на основании анализа следующих параметров:

- MAC адрес,
- IP адрес,
- Маркер ширококестельного пакета,
- Маркеры типа пакета,
- MAC адрес источника,
- IP адрес источника,
- Исходящий порт,
- Входящий порт
- Номер входного слова пакета,
- Длина IP заголовка,
- Маркеры сегментации IP пакета.

Модуль CrcSum используется для расчета контрольной суммы пакета [3]. Расчет происходит в конвейере с сохранением промежуточного результата. При несовпадении контрольных сумм осуществляется сброс пакета.

Блок передачи пакетов в канал DMA состоит из модулей Rx.VFIFOctrl, Pkt.Addr.Buffer, AXI4interconnect (см. рис. 1). Если после классификации принято решение, что пакет предназначен для обработки процессором, то после его буферизации он поступает в модуль Rx.VFIFOctrl. Этот модуль предназначен для формирования цепочек (chain) DMA структур (команд) и передачи этих цепочек и соответствующих данных во внешнюю память процессора. Структура DMA команды приведена на рис. 2 и включает в себя следующие поля:

- Remote (CPU) address – поле адреса, куда должны быть сохранены данные для обработки процессором,
- Next major pointer – поле адреса следующей цепочки команд,
- Type data/lenth of transfer – поле, содержащее тип и длину данных команды,
- Local (FPGA) address – адрес, откуда должны быть взяты данные для передачи,
- Control/next chain –команда и указатель на следующую команду в цепочке,
- Mailbox address – адрес обработчика команд, на который отправляется команда,
- Type (Msg.)/lenth of transfer – тип передаваемых данных и их длина,
- 64 bit message – сопутствующее описание, которое может, например, служить идентификатором,
- Stride – команда прямого перехода.

Поля, помеченные “not use“, не используются и заложены для дальнейшего расширения функционала без нарушения структуры команд.

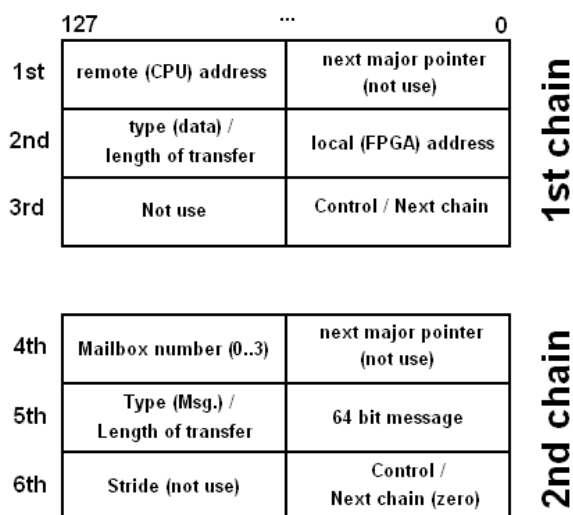


Рис. 2. Структура DMA-команд (chain)

Цепочки формируются при наличии валидных данных на выходе модуля Pkt.FIFO, дескриптора свободных блоков внутренней памяти процессора в модуле Pkt.Addr.Buffer и при наличии свободной внешней памяти SDRAM. Сформированная цепочка передается через внутреннюю AXI4 шину и сохраняется во внешней SDRAM памяти. Модуль Rx.VFIFOctrl обеспечивает прием данных одновременно от двух модулей Pkt.FIFO.

Формирование и передача цепочек DMA структур происходит «на лету». Это означает, что сформированная цепочка не запоминается в какой-либо памяти и каждое слово передается сразу после окончания его формирования. После передачи первых шести полей (один chain) начинается передача пакета во внешнюю память.

По окончании передачи пакета в DMA передается адрес начала первой структуры во внешней памяти. Начиная с этого адреса DMA разделяет цепочку на подзадачи и сначала

передает данные в память процессора, а потом передает сообщение в процессор об окончании передачи. Затем из модуля DMA в модуль Rx.VFIFOctrl передается адрес начала загруженной цепочки. На основании этого адреса модуль Rx.VFIFOctrl рассчитывает место, занимаемое пакетом, и перемещает указатель чтения на адрес, следующий за пакетом. Этим обеспечивается освобождение занимаемой пакетом памяти. Модуль Pkt.Addr.Buffer предназначен для хранения и выдачи указателей на выделенные блоки внутренней памяти процессора и представляет собой стандартное FIFO, которое было адаптировано для последовательной выдачи свободных адресов.

Для арбитража доступа к памяти был использован AXI4 interconnect модуль. Блок управления состоит из модулей AXI_liteCTRL и AXI4_littleBridge. Модуль AXI_liteCTRL управляет записью значений в управляющие регистры контролируемых модулей. Запись этих регистров происходит по *xf_car* шине. Данная шина представляет собой замкнутую, двунаправленную цепочку из одного ведущего модуля и нескольких ведомых модулей. Каждый модуль независимо от его назначения имеет одинаковый интерфейс, состоящий из двух пар шин *cmd* и *rsp* – входной и выходной. Такая структура обеспечивает легкое добавление новых ведомых модулей в цепочку. В данном проекте использована схема с одним *xf_car* модулем в качестве ведущего модуля AXI4_lite шины.

Передатчик

Передача данных осуществляется в виде UDP пакетов длиной до 8 Кбайт. Для формирования пакетов был разработан специальный модуль генерации пакетов. Выбор этого типа пакетов был обусловлен простотой их обработки и передачи, а также возможностью их передачи без ожидания подтверждения получения [4, 5]. Целостность потока данных обеспечивается настройкой каналов передачи данных в *lossless* (без потерь) режиме, а также контролем последовательности пакетов программным обеспечением верхнего уровня. Упакованные данные поступают на вход модуля 10 Gigabit Ethernet MAC core. Ввиду того, что передаваемый поток данных значительно превышает возможности передачи одного 10 Gigabit Ethernet канала, передаваемые данные перед упаковкой в пакеты разделялись на несколько каналов передачи как показано на рис. 3.

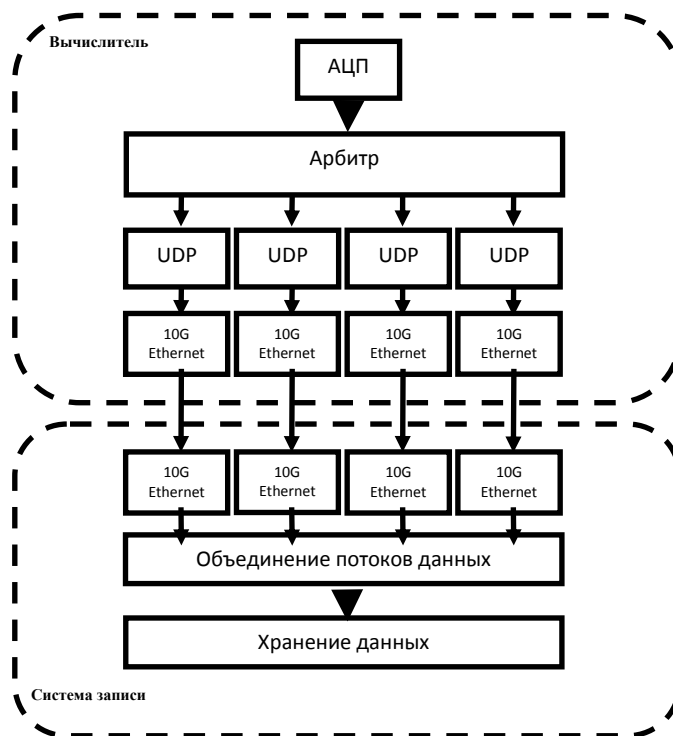


Рис. 3. Структура передачи данных

Число каналов ограничено только количеством физических линий. Для распределения пакетов по каналам был разработан соответствующий модуль, который также обеспечивает синхронизации данных с целью последующей программной сборки данных в один поток на приемной стороне.

Для разделения данных по потокам реализован арбитр. Арбитр состоит из нескольких буферов FIFO и автомата, контролирующего заполнение буферов. При поступлении данных на отправку данные направляются в доступный буфер. Из FIFO данные вычитываются в постоянном порядке. Автомат отслеживает распределение данных по буферам и маркирует их.

Тестирование канала связи

Модули системы и система в комплексе были функционально промоделированы с использованием тестовых модулей. В частности, был разработан модуль генерации и отображения ethernet пакетов. Моделирование происходило в среде ModelSim. Тестовые данные, как и реальные данные, характеризуются параметрами, которые приведены в таблице:

Параметр	Принимаемые значения
Тип пакета	UDP, TCP, IP, ARP, broadcast
Размер пакета	64 – 65535 байт
Содержимое пакета	Случайное, заданное
Флаги сегментации и сдвига	Есть, нет
Адрес отправителя и получателя пакета	IP-адрес и порт
Контрольная сумма	Есть, нет

Также для моделирования ситуации с помехами на линии передачи данных реализован метод внесения ошибки.

Результаты моделирования полностью подтвердили правильность принятых решений, а также достижение требуемых характеристик.

На рис. 4 изображен пример передачи пакета размером 8 Кбайт, смоделированный в этой среде.

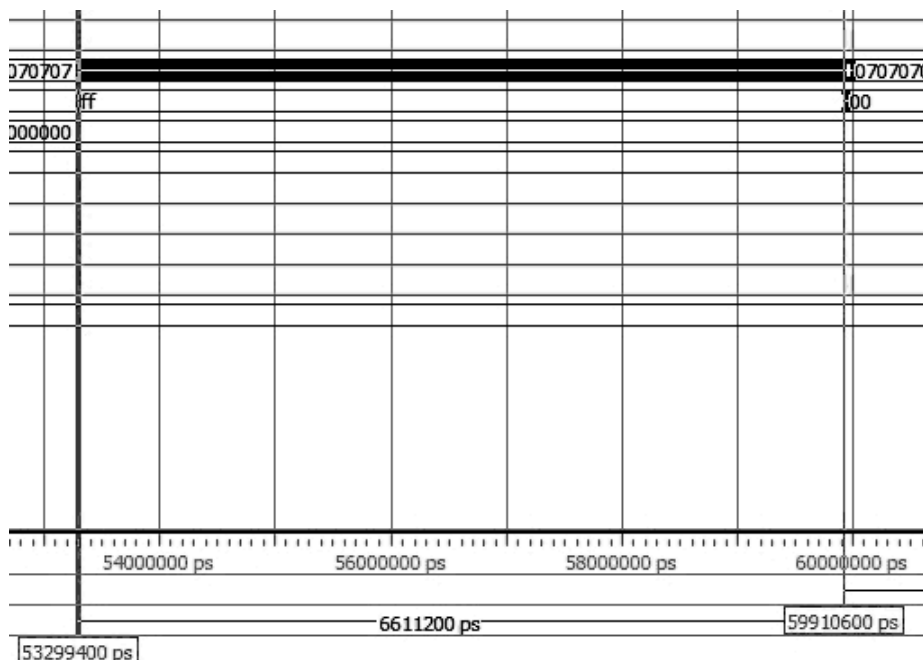


Рис. 4. Моделирование передачи пакета в среде ModelSim

Передача этого пакета занимает 6611,2 нс, что обеспечивает скорость передачи данных 1,154 Гбайт/с по одному каналу. Синтез проекта осуществлялся в среде Synplify. Это позволило оценить предельные частоты передачи сигналов между регистрами ПЛИС, которые, как

оказалось, превышают 300 МГц. Это также дало возможность оценить требуемые ресурсы ПЛИС для реализации проекта.

Проект был реализован на ПЛИС virtex5 и проведено физическое тестирование разработанного канала передачи данных. Установлено, что один канал обеспечивает пиковую скорость передачи 1,1 Гбайт/с, что близко к значению, предсказанному результатами моделирования. При тестировании одновременной передачи данных по трем каналам пиковая скорость составляет 3,3 Гбайт/с. Средняя скорость передачи данных, как показали результаты тестирования, составила 0,98 и 2,94 Гбайт/с при передаче данных соответственно по одному и трем каналам. Важно, что пиковые и средние скорости передачи данных отличаются незначительно.

Заключение

Таким образом, предложен и реализован эффективный способ построения высокоскоростного канала передачи данных. Он основан на стандартных модулях управления физической средой с использованием пакетов без подтверждения целостности и распределением данных по нескольким физическим линиям. Такое построение позволяет увеличивать объем передачи данных путем увеличения количества физических линий. Предложенные решения были реализованы при создании канала связи с пропускной способностью около 25 Гбит/с, при этом модули проекта имеют тактовую частоту 300 МГц.

Список литературы: 1. IEEE 802.3 ethernet working group. – Режим доступа: <http://www.ieee802.org/3/>
2. Цуканов, В. Н. Способ передачи цифровых потоков данных по волоконно-оптической линии связи / В.Н. Цуканов, М.Я. Яковлев // Поиск патентов и изобретений. – Режим доступа: <http://www.findpatent.ru/patent/245/2454805.html>
3. C. Borrelli IEEE 802.3 Cyclic Redundancy Check. – http://www.xilinx.com/support/documentation/application_notes/xapp209.pdf – 23.03.2001 г.
4. Axelson, J. Embedded Ethernet and Internet Complete. Designing and Programming Small Devices for Networking. – Madison : Lakeview Research LLC, 2003. – 482с.
5. Held, G. Ethernet Networks: Design, Implementation, Operation, Management. – John Wiley & Sons, 2003. – 583с.

*Радиоастрономический институт
НАН Украины*

Поступила в редколлегию 05.11.2013