

Сложно-функциональный блок коммуникационной среды для систем на кристалле

И.И. Насонов, А.П. Панфилов, А.Л. Эйсымонт, Д.И. Дрягалкин

ЗАО НТЦ «Модуль», {nasonov | panfilov | eisymont | didryaga} @module.ru

Аннотация — Технологическое развитие микроэлектроники позволяет интегрировать большое количество компонентов в СБИС. Одна из важнейших задач при этом – разработка эффективного механизма коммуникации между блоками микросхемы. В статье представлен результат разработки модульного параметризуемого IP-блока коммуникационной среды с конфигурируемой структурой. С помощью данного блока можно реализовывать коммуникационные среды различной сложности: от простых коммутаторов до сложной системы с несколькими локальными узлами. Исследованы архитектурные методы повышения быстродействия коммуникационной среды.

Ключевые слова — система на кристалле, коммуникационная среда, AMBA AXI.

I. ВВЕДЕНИЕ

Взаимодействия между IP-блоками на кристалле СБИС осуществляются с помощью сигнальных интерфейсов, проблема стандартизации которых, в целом, решена – имеется ряд протоколов: AMBA, OCP, PLB, VCI, Wishbone.

Помимо выбора IP-блоков одной из важных задач для разработчика системы на кристалле является построение коммуникационной среды в виде простого или сложного коммутатора. Архитектура коммуникационной среды зависит от количества IP-блоков, их топологического размещения на кристалле и требований по временным характеристикам.

Усложнение архитектур систем на кристалле сопровождается следующими тенденциями [1]:

- возрастанием доли схем передачи данных в общем разделении ресурсов кристалла (площади и мощности);
- существенным влиянием систем коммуникации на общую производительность систем;
- ростом количества и сложности IP-блоков, а также большим разнообразием их функций;
- унификацией элементов систем коммуникации;
- усложнением анализа систем коммуникации.

II. ТИПЫ АРХИТЕКТУР КОММУНИКАЦИОННЫХ СРЕД

Простой вариант коммутатора используется для построения системы с небольшим набором устройств,

которые расположены на кристалле относительно близко друг от друга.

При большом количестве IP-блоков коммутатор разбивается на локальные сегменты. Сегментированная архитектура состоит из нескольких простых коммутаторов, запросы проходят несколько стадий арбитража. Разбиение коммутатора обусловлено тем, что относительно более простые схемы сегментов могут работать на более высокой частоте, обеспечивая общую производительность системы.

В архитектурах, где число IP-блоков достигает десяти и больше, коммуникационные среды принимают вид сети на кристалле. Основное отличие сетевой архитектуры состоит в том, что она не содержит в себе существенных ограничений по количеству узлов [1].

Наиболее распространены коммуникационные среды на основе протокола шины AMBA AXI [2]. Существуют различные решения по реализации таких коммутаторов от фирмы ARM (NIC-301 [3]) или от фирмы Synopsys (Designware AMBA Interconnect [4]) и других фирм. Однако не всегда такие решения позволяют получить приемлемые реализации коммутаторов для систем с интенсивным произвольным доступом в память, которые наиболее чувствительны к пропускной способности коммутаторов.

Далее описывается разработанный подход к проектированию коммутаторов с повышенной производительностью.

III. ПРЕДСТАВЛЕНИЕ АРХИТЕКТУРЫ КОММУТАТОРА

Разработка архитектуры коммуникационной среды базируется на использовании универсального конфигурируемого IP-блока коммутатора. Такой IP-блок состоит из узлов и межсоединений узлов. Применение IP-блока включает в себя задание параметров всех составных частей и последующий синтез требуемой реализации.

Разработанный IP-блок имеет в составе следующие компоненты: библиотеку базовых модулей, шаблон узла, шаблон межсоединения узлов и интерфейсов коммутатора, шаблон верхнего уровня среды, а также инструменты синтеза, позволяющие получать

доступный для САПР исходный код коммутатора и тестового окружения.

А. Библиотека базовых модулей

Библиотека базовых модулей коммуникационных схем состоит из параметризуемых элементов – цифровых блоков, на базе которых строится система передачи данных между функциональными блоками системы на кристалле. Основа библиотеки – простейшие блоки – мультиплексоры и демultipлексоры сигналов квитирования, кодеры и декодеры, арбитры, буферы FIFO. Имеются также элементы с более сложной функциональностью – схемы предотвращения взаимных блокировок. Элементы библиотеки параметризуемы – можно задать такие характеристики как количество портов, ширину шин, размер буфера.

В. Узлы

Узлы коммуникационной среды – коммутаторы, перенаправляющие сигналы шин (адреса, данных и подтверждений) из одного интерфейса в другой. Внутри узла осуществляются арбитраж и коммутация передач адреса и данных. Через узел сигналы шин проходят асинхронно, то есть внутри узлов нет последовательных схем для сигналов шин.

В интерфейсе AXI имеются 2 рода сигналов: сигналы квитирования (valid и ready) и информационные сигналы. Соответственно, узел коммутатора условно можно разделить на две части: управляемую и управляющую. Управляемая часть содержит мультиплексоры информационных сигналов шин, переключаемые селекторами из управляющей части.

Управляющая часть узла (рис. 1) состоит из набора блоков, называемых портами узла (Slave-порт и Master-порт). Порты разных типов соединены между собой перекрёстно (схема подключения портов) внутренними сигналами квитирования.

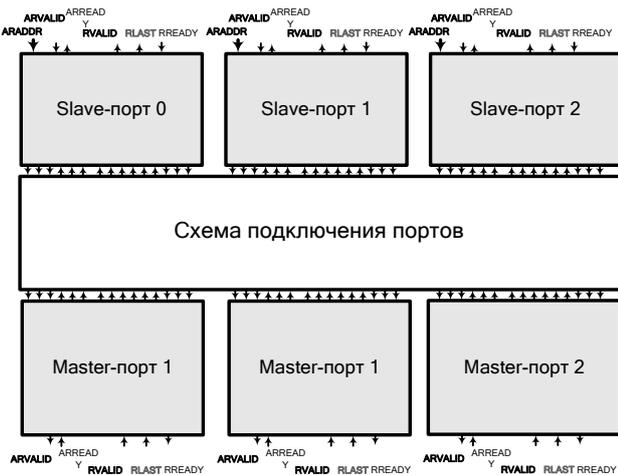


Рис. 1. Структурная схема управляющей части узла

Архитектура управляющей части определяет, по каким принципам производится передача запросов, данных и подтверждений. Разработаны две архитектуры управляющей части с разными свойствами: архитектура с коммутацией каналов и архитектура с коммутацией пакетов.

Архитектура с коммутацией каналов (или in-order) рассчитана на строгую последовательность ответов на запросы в каждом интерфейсе, обратный маршрут для возврата ответа хранится внутри узла в специальных очередях.

Архитектура с коммутацией пакетов (или out-of-order) использует возможности протокола AXI по внеочередному завершению транзакций, сигналы маршрутизации при этом передаются вместе с информационными сигналами каналов.

Преимущества архитектуры с коммутацией каналов: лучшие временные характеристики, а также возможность выдачи через один slave-интерфейс нескольких запросов в разные master-интерфейсы одновременно. Преимущество архитектуры с коммутацией пакетов: поддержка возможностей внеочередного выполнения транзакций AXI. В зависимости от характера коммуникаций в системе одна из двух архитектур может оказаться эффективнее с точки зрения производительности системы.

С. Межсоединения узлов

Межсоединение по принципу точка-точка подключается к двум интерфейсам разных узлов или к интерфейсу узла и непосредственно к интерфейсу коммуникационной среды. Межсоединение может быть конвейеризовано настраиваемым числом буферов FIFO (из библиотеки базовых модулей); в качестве параметра также можно указать размер каждого буфера.

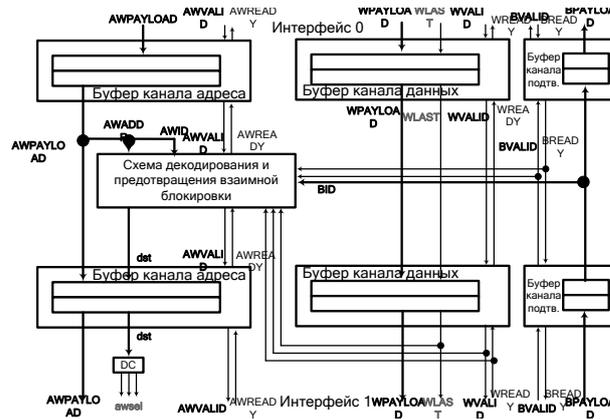


Рис. 2. Структурная схема межсоединений узлов

На рис. 2 показана структура межсоединения (только запись), где в каждом канале стоит цепочка из двух буферов FIFO. На данном примере также показано, как между конвейерными буферами встроена схема декодирования адреса и схема

предотвращения взаимной блокировки. Данная возможность позволяет улучшить временные характеристики узла за счёт переноса части критичных по задержке цепей декодирования адреса на другую ступень конвейера. Интерфейс 0, показанный сверху, имеет тип slave (то есть подключается к master-устройству шины AXI3), интерфейс 1, показанный снизу – имеет тип master (подключается к slave-устройству).

D. Общая архитектура коммуникационной среды

На приведённом ниже примере (рис. 3) показано, каким образом составные элементы (узлы и межсоединения) образуют сегментированный коммутатор.

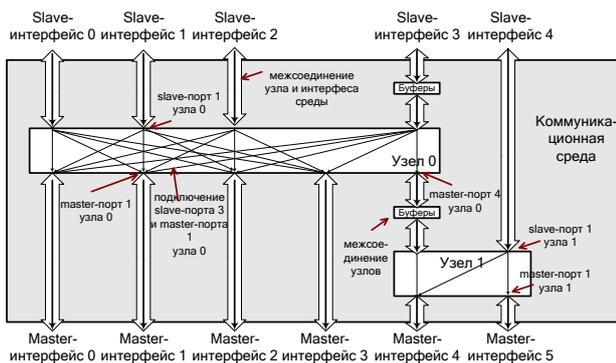


Рис. 3. Элементы архитектуры коммутатора

Следует упомянуть также формальный элемент архитектуры узла — подключение портов внутри узла. Наличие подключения определяет возможность доступа из slave-порта в master-порт узла.

E. Инструменты синтеза и анализа архитектур

Синтез реализации коммутатора заключается в получении его RTL-описания на основе вводимых пользователем параметров IP-блока. Для этого разработан специальный обработчик шаблонов коммуникационной среды (рис. 4).

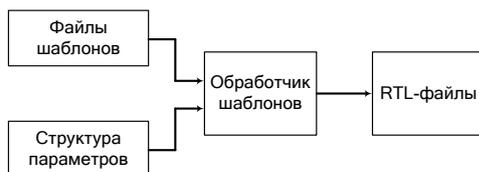


Рис. 4. Схема синтеза RTL-описания коммутатора

Анализ полученных описаний производился на следующих стадиях маршрута проектирования:

- верификация (моделирование) коммутатора в симуляторе языков Verilog и e (Specman),
- логический синтез принципиальной схемы коммутатора на элементах библиотеки КМОП с технологическими нормами 65 нм и анализ полученных характеристик.

Специально разработанные тесты позволяют исследовать системные характеристики коммуникационной среды. Сценарий теста подразумевает обмен транзакциями между функциональными моделями master- и slave-устройств; коммутатор играет роль тестируемого устройства. Каждая модель master-устройства инициирует транзакцию к slave-устройству, выбор которого производится случайно с равномерным распределением по всем доступным slave-устройствам. Для теста на максимальную производительность выбран одиночный тип транзакции (с одной передачей в пакете данных), так как транзакции такого типа максимально используют пропускную способность всех каналов интерфейса AXI. Модели slave-устройств в системе одинаковые и могут обрабатывать запросы каждый такт в конвейере с минимальной (но не ограниченной по максимальному значению) латентностью в 2 такта. Тесты могут быть запущены в одном из двух режимов:

- режим с виртуальными каналами, использующий разные идентификаторы AXI3,
- режим без виртуальных каналов, использующий постоянный идентификатор 0.

Разработанная система синтеза и анализа позволяет пользователю получать различные реализации коммутатора: можно разбивать архитектуру на сегменты, выбирать типы узлов и подбирать другие параметры.

IV. ПРИМЕР РЕАЛИЗАЦИИ КОММУТАТОРА AMBA AXI3

Рассмотрим типичный пример для анализа архитектуры со следующими характеристиками:

- полностью синхронная схема (единственный тактовый сигнал на весь коммутатор),
- 4 slave-интерфейса и 4 master-интерфейса,
- по 8 конвейеризуемых в узле транзакций каждого типа (запись и чтение),
- slave-интерфейсы 0-2 не используют идентификаторы AXI, интерфейс 3 использует переменные идентификаторы в транзакциях.

Сравниваемые архитектуры:

- NIC-301 – соответствующая конфигурация IP-блока фирмы ARM,
- in-order – реализация разработанного IP-блока с узлом, построенным по архитектуре с коммутацией каналов,
- in-order (с вынесенным декодером) – архитектура с коммутацией каналов, в которой вынесен блок декодирования адреса,

- out-of-order – реализация разработанного IP-блока с узлом, построенным по архитектуре с коммутацией пакетов,
- out-of-order (с вынесенным декодером) – архитектура с коммутацией пакетов, в которой вынесен блок декодирования адреса.

В архитектурах с вынесенным декодером на slave-интерфейсе устанавливаются конвейерные буферы FIFO глубиной в 2 ячейки.

Таблица 1

Результаты теста на максимальную производительность в режиме с использованием виртуальных каналов

Архитектура	Отн. производительность	Среднее время обраб. запросов
NIC 301	30%	4,6 такта
in-order	65%	2,5 такта
in-order (вын. декодер)	65%	5,6 такта
out-of-order	31%	3,4 такта
out-of-order (вын. декодер)	24%	6,1 такта

Таблица 2

Результаты теста на максимальную производительность в режиме без использования виртуальных каналов

Архитектура	Отн. производительность	Среднее время обраб. запросов
NIC 301	30%	4,6 такта
in-order	65%	2,5 такта
in-order (вын. декодер)	65%	5,6 такта
out-of-order	77%	2,3 такта
out-of-order (вын. декодер)	76%	4,7 такта

Таблица 3

Технологические характеристики коммутатора

Архитектура	Макс. частота, МГц	Площадь, тыс. мкм ²
NIC 301	601	69,0
in-order	770	28,6
in-order (вын. декодер)	833	49,2
out-of-order	569	35,2
out-of-order (вын. декодер)	692	57,6

Значение относительной производительности рассчитывалось для отдельно взятого slave-интерфейса как отношение полезного времени (такты передачи данных на slave-интерфейсе) к общему времени теста. Высокие значения этого параметра говорят о большей интенсивности обслуживания запросов. Показаны значения для slave-интерфейса 3 при максимальной интенсивности, когда тестовое окружение выдаёт запросы на всех интерфейсах каждый такт.

Время обработки запроса – это количество тактов между моментом, когда тестовое окружение выдаёт запрос на slave-интерфейс коммутатора и моментом, когда приходит ответ. В таблицах 1 и 2 показано среднее арифметическое для всех запросов теста через slave-интерфейс 3.

Технологические характеристики (табл. 3) получены по результатам логического синтеза RTL-моделей на библиотеке с технологическими нормами 65 нм.

Основные выводы:

- узел с коммутацией каналов может работать на большей частоте, чем узел с коммутацией пакетов,
- конвейеризация (установка буферов и вынос схем декодирования адреса) даёт выигрыш в максимальной частоте и проигрыш во времени доступа,
- в условиях описанного теста архитектура с коммутацией каналов показывает большую производительность за счёт особенностей схемы предотвращения взаимной блокировки,
- реализации IP-блока могут иметь в 1,5-3 раза большую производительность, чем NIC-301.

V. ЗАКЛЮЧЕНИЕ

В статье рассмотрен метод формального представления архитектуры коммуникационной среды и показано, как с помощью данного метода разработан IP-блок коммутатора AXI3 для систем на кристалле. Разработанное средство позволяет синтезировать RTL-описание коммутатора в соответствии с вводимыми параметрами.

При реализации пользователь может выбрать конфигурацию, соответствующую его требованиям по частоте тактового сигнала, производительности и времени обработки запросов.

Показана эффективность архитектурных решений, позволяющих повысить быстродействие схемы. К этим методам относятся конвейеризация каналов AXI, вынос декодера адреса и схем предотвращения блокировки за конвейерные регистры, использование узла с маршрутизацией каналов.

Разработанный IP-блок коммутатора превосходит аналог NIC-301 по максимальной частоте тактового сигнала и по производительности на описанных тестах.

ЛИТЕРАТУРА

- [1] Jantsch A., Tenhunen H. Networks On Chip // Kluwer Academic Publishers. 2003. P. 3–16.
- [2] AMBA AXI Protocol Specification, Version 2.0, ARM, 2010. URL: <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ih0022e/index.html> (дата обращения 31.01.2014).
- [3] CoreLink NIC-301 Network Interconnect Revision: r2p3 Technical Reference Manual URL: <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ih0022e/index.html> (дата обращения 31.01.2014).
- [4] DesignWare IP Solutions for AMBA Interconnect. URL: <http://www.synopsys.com/IP/SoCInfrastructureIP/DesignWare/AMBA/Pages/default.aspx> (дата обращения 31.01.2014).