

# Анализ эффективности применения буферизации для маршрутизации в сети-на-кристалле

И.А. Медведев

ОАО Научно-производственный центр “Электронные вычислительно-информационные системы”, [imedvedev@elvees.com](mailto:imedvedev@elvees.com)

**Аннотация** — Развитие технологических возможностей микроэлектроники вкупе с требованиями постоянного повышения производительности вычислительных устройств ведет к изменению приоритетов при разработке механизмов межсоединения. Все больше внимания уделяется масштабируемым сетям-на-кристалле. В статье отображен опыт разработки параметризованной по количеству узлов и типу буферизации сети с топологией типа двумерная решетка. Получены качественная и количественная оценки эффективности применения того или иного типа буферизации, характеризующегося определенным месторасположением в маршрутизаторе и глубиной используемых FIFO очередей, для сетей с различным количеством узлов.

**Ключевые слова** — многоядерные процессоры; масштабируемость межсоединения; сети-на-кристалле; виртуальные каналы; буферизация каналов.

## I. ВВЕДЕНИЕ

На сегодняшний день процессорная индустрия твердо стоит на пути повышения производительности за счет увеличения количества ядер на кристалле [1]. Вопрос реализации высокопроизводительного межъядерного взаимодействия, отвечающего новым требованиям по масштабируемости, представляется все более актуальным. Исследования последних лет показывают, что использование сетей-на-кристалле позволяет эффективно решать задачи обмена информацией между ядрами, никак не ограничиваясь при этом в вопросе масштабируемости [2], [3]. Многие мировые производители современных процессоров уже имеют готовые кристаллы, реализующие сеть-на-кристалле в качестве основного элемента для межъядерного взаимодействия [4], [5]. Многоплановость исследований в данной области и в то же время универсальность основных принципов построения сетей (многоуровневый подход к проектированию, повторное использование аппаратных ресурсов) позволяют создавать эффективные механизмы межсоединения, сохраняющие прозрачность реализации и гибкость программирования, что существенно облегчает исполнение параллельных вычислений. Ключевым функциональным блоком сети является маршрутизатор, отвечающий за передачу данных в каждом узле. От эффективности его работы зависит

качество функционирования всего механизма межсоединения.

## II. МАРШРУТИЗАТОР В СЕТИ-НА-КРИСТАЛЛЕ

Структурно маршрутизатор можно разбить на три блока: входной, внутренний и выходной (рис. 1). Каждый из них содержит определенное количество каналов передачи, соответствующее его функциональному назначению. Способ организации каналов передачи сообщений (flow control) в маршрутизаторе влияет на его быстродействие и пропускную способность. Для того, чтобы исключить вероятность потери сообщений и повысить эффективность их передачи, в каждом из блоков можно реализовать либо буферизацию линий передачи, либо поддержку виртуальных каналов на ее основе. Под буферизацией в данной работе понимается формирование на одном канале передачи двух виртуальных путей, один из которых проходит через очередь типа FIFO. Это позволяет как пропускать сообщения напрямую через канал, так и выполнять сохранение сообщений в очереди при возникновении конфликтов в случае одновременного обращения к выходному порту со стороны нескольких входных запросов или отсутствия возможности у выходного порта на текущем такте принять запрос.

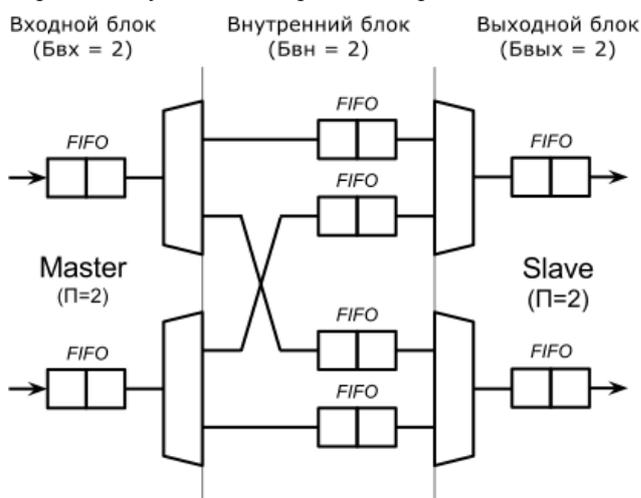


Рис. 1. Пример буферизации маршрутизатора с двумя ведущими (master) и двумя ведомыми (slave) портами; обозначение типа буферизации: 2-2-2

Буферизация является основой для построения виртуальных каналов в сетях-на-кристалле. Виртуальные каналы реализуются посредством формирования на одном канале передачи нескольких виртуальных путей, проходящих через FIFO очереди. В случае буферизации доступна только одна очередь для сохранения данных, и она используется автоматически при возникновении блокировки. В случае виртуальных каналов доступны сразу несколько очередей, каждая из которых соответствует одному виртуальному пути. При этом ее выбор осуществляется либо программистом, либо аппаратным указанием номера пути для конкретного набора данных. Эффективность использования виртуальных каналов в сетях-на-кристалле уже не раз подтверждена в реальных проектах [6]. Применение виртуальных каналов является далеко не единственной техникой улучшения работы маршрутизатора, но зачастую становится эффективным способом повышения производительности. Учитывая функциональную и структурную наследственность реализаций виртуальных каналов и буферизации, важно провести оценку эффективности последней.

Возвращаясь к структурному разбиению маршрутизатора, встает вопрос определения блоков, буферизация которых приводит к наибольшему повышению производительности. Необходимо понимать, что существуют варианты буферизации одновременно нескольких блоков. Каждая реализация требует аккуратной оценки аппаратных параметров (максимальной рабочей частоты, занимаемой площади и потребляемой мощности). При этом отдельного внимания требует тот факт, что приложения, реализующие отличный друг от друга трафик, по-разному используют ресурсы буферизации.

### III. РАЗРАБОТКА МАСШТАБИРУЕМОГО МЕЖСОЕДИНЕНИЯ

Вопрос масштабируемости механизма межсоединения поднимался и в многоядерных процессорах серии “МУЛЬТИКОР” фирмы ОАО НПЦ “ЭЛВИС” [7]. В силу ряда архитектурных особенностей наследование существующего способа межсоединения посредством коммутатора уже для четырех ядер вызывает сложности. Учитывая, что в перспективе количество ядер будет только увеличиваться, было принято решение исследовать вопрос перехода к сетям-на-кристалле с топологией типа двумерная решетка (рис. 2). В первую очередь была реализована аппаратная поддержка синхронизации потоков с распределенным, масштабируемым способом взаимодействия. Ряд тестов показал, что переход от централизованного межсоединения (например, шина или коммутатор) к распределенному, обеспечивает требования по масштабированию и ведет к небольшому падению производительности (в среднем на несколько процентов для большинства приложений) и увеличению занимаемой площади. Более подробные результаты этого эксперимента представлены в статье [2]. Учитывая, что при увеличении числа ядер, централизованное межсоединение ведет к

стремительному усложнению логики коммутации и недопустимому повышению потребляемой мощности [8], результаты были признаны приемлемыми для дальнейшего исследования вопроса построения сети-на-кристалле.

Следующий этап работы, результаты которого приведены в данной статье, заключался в создании платформы, позволяющей по заданным параметрам конфигурации (количеству узлов в сети, топологии сети, глубины FIFO очередей буферизации и др.) получать пригодное к синтезу RTL-описание соответствующей сети-на-кристалле. По сути, платформа, содержащая определенный набор программных средств и базовое представление основных компонентов межсоединения является гибким, конфигурируемым IP-блоком. Дополнительно такая платформа позволяет вычленять RTL-описание маршрутизатора, используемого в сети, для дальнейшей независимой интеграции в другие проекты.

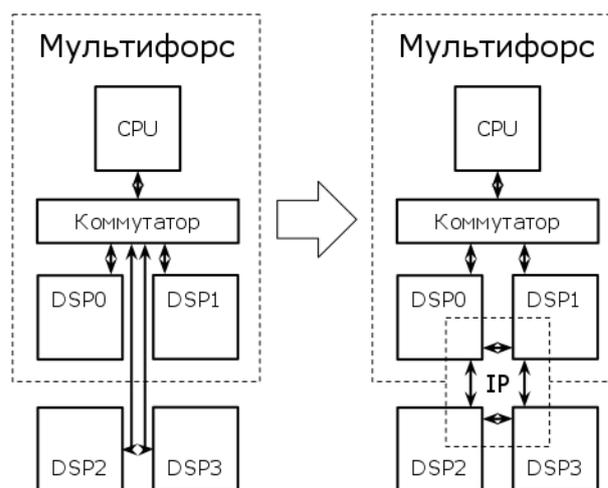


Рис. 2. Реализация масштабируемого межсоединения в процессорах серии “МУЛЬТИКОР”

### IV. СОЗДАНИЕ ПАРАМЕТРИЗОВАННОГО RTL-ОПИСАНИЯ

К сожалению, на сегодняшний день стандарты языка Verilog и SystemVerilog, а также некоторые программные средства для моделирования и синтеза (в особенности не самые новые) имеют достаточно слабые функциональные возможности по созданию параметризованных RTL-описаний (например, невозможна параметризация количества портов в модуле). Существует несколько подходов к реализации недостающих возможностей языков HDL. Этот вопрос представляет собой материал для отдельной статьи, поэтому здесь ограничимся лишь кратким описанием того подхода, который использован в данной работе. Для его реализации на языке программирования Ruby были созданы программные средства для парсинга базового RTL-описания и генерации выходного, соответствующего выбранным параметрам и удовлетворяющего синтаксическим требованиям синтезатора.

Базовым (входным) описанием, используемым для генерации, является представление двухпортового (master/slave AMBA AXI) маршрутизатора. После его чтения и разбиения на основные функциональные блоки производится установка параметров и генерация набора выходных модулей. Среди которых - определенные топологией и размерами сети маршрутизаторы с различным набором портов, модуль верхнего уровня с соответствующим межсоединением узлов-маршрутизаторов и вспомогательный модуль-обвязка для синтеза. Например, для сети с четырьмя узлами с топологией двумерной решетки размерностью  $2 \times 2$  формируется четыре маршрутизатора с портами *LSE*, *LWS*, *LNW*, *LWN*, где ориентация портов определена как *Local*, *North*, *East*, *South*, *West*. Такая гибкая платформа позволяет исследовать архитектурные особенности построения сети-на-кристалле с различной размерностью и топологией.

#### V. РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ ЭФФЕКТИВНОСТИ БУФЕРИЗАЦИИ В МАРШРУТИЗАТОРЕ

Рассмотрим типовую задачу для анализа производительности механизма межсоединения, который в общем случае можно представить в виде блока, имеющего определенное количество ведущих (master) и ведомых (slave) портов. Набор портов определяется исходя из числа ведущих и ведомых устройств, входящих в состав конкретной системы и использующих межсоединение для обмена данными. Для оценки его производительности необходимо понять, насколько быстро все данные, отправленные каждым ведущим устройством, достигают ведомых адресатов. Зачастую ответ на этот вопрос зависит от нескольких характеристик, так или иначе связанных между собой и влияющих друг на друга. В контексте данной работы можно выделить две из них: пропускную способность (throughput) и задержку (latency) при передаче данных. Пропускная способность показывает объем успешно переданных данных через механизм межсоединения за единицу времени и в общем случае измеряется в количестве битов за секунду (бит/с). Имея дело в нашем случае с процессорными архитектурами, где передача данных осуществляется через инструкции пересылки, можно говорить о количестве исполняемых инструкций пересылки за секунду (IPC). Аналогично и задержка при передаче данных может измеряться либо в секундах, либо как в нашем случае в тактах.

Теперь остановимся подробнее на тех вариантах сетей, которые использовались при проведении экспериментов. В рамках данной работы для каждой генерируемой реализации межсоединения была выбрана топология типа двумерная решетка. Количество узлов в сети и размерность решетки варьировались в диапазоне, представленном в табл. 1. Там же отражено и общее количество всех портов всех маршрутизаторов в сети. Для наглядности на рис. 3 показаны варианты межсоединений с размерностями  $3 \times 1$ ,  $3 \times 2$ ,  $3 \times 3$  и пример маршрутизатора с четырьмя ведущими и четырьмя ведомыми портами. По такому

принципу генерировались и остальные варианты реализации.

Таблица 1

Описание генерируемых сетей, используемых для проведения экспериментов

Количество узлов в сети	Размерность 2D решетки ( $X$ на $Y$ )	Общее кол-во портов всех маршрутизаторов
2	2 на 1 ( $2 \times 1$ )	8
3	$3 \times 1$	14
4	$2 \times 2$	24
6	$3 \times 2$	40
8	$4 \times 2$	56
9	$3 \times 3$	66
12	$4 \times 3$	108
16	$4 \times 4$	128

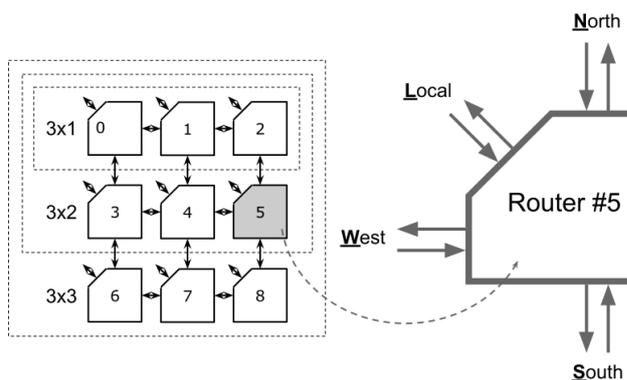


Рис. 3. Варианты сетей размерностями  $3 \times 1$ ,  $3 \times 2$ ,  $3 \times 3$  и пример маршрутизатора

Была внедрена поддержка буферизации маршрутизаторов сети в силу эффективности такого подхода в повышении производительности всего механизма межсоединения и формирования основы для построения виртуальных каналов. В каждом из трех блоков (входном, внутреннем, выходном) базового описания маршрутизатора была включена возможность буферизации посредством FIFO очереди различной глубины (2, 4, 8, 16, ...). При этом была сохранена опция отключения буферизации в каждом из блоков. Пример маршрутизатора показан на рис. 1, где глубины FIFO очередей определяют тип буферизации, обозначение которого в дальнейшем будет приводиться через  $X-Y-Z$ , где  $X$  – глубина очереди входного блока,  $Y$  – внутренней и  $Z$  – выходного. Например, обозначение  $2-4-0$  соответствует конфигурации, при которой при выключенной выходной буферизации глубина входной очереди равна 2, а внутренней – 4. Общий объем FIFO очередей (буферизации) в системе определяется из набора используемых маршрутизаторов, для каждого

из которых общая глубина буферизации ( $O$ ) вычисляется по формуле:

$$O = B_{вх} * П + B_{вн} * П^2 + B_{вых} * П, \quad (1)$$

где  $B_{вх}$ ,  $B_{вн}$  и  $B_{вых}$  – глубины буферизации входного блока, внутреннего и выходного соответственно, а  $П$  – количество портов одного типа (ведущих или ведомых).

В силу специфики реализации инструкций передачи данных в системе, заключающейся в необходимости обязательного сохранения запроса при возникновении конфликта одновременного доступа к одному ведомому порту маршрутизатора,

накладывается ограничение на использование некоторых типов буферизации. Так как конфликтная ситуация обрабатывается еще до выходного блока, то обязательна должна существовать возможность сохранения запроса во входном или внутреннем блоке. Поэтому типы, соответствующие такому условию исключены из анализа (например, комбинации 0-0-2, 0-0-4 и т.п.). Остальные типы, ограниченные глубиной FIFO очередей до 4, использованы в сетях с топологией двумерной решетки размерностями 2x1, 3x1, 2x2, 3x2, 3x3, 4x4. При этом тип буферизации устанавливается одинаковый для всех маршрутизаторов в выбранной сети.

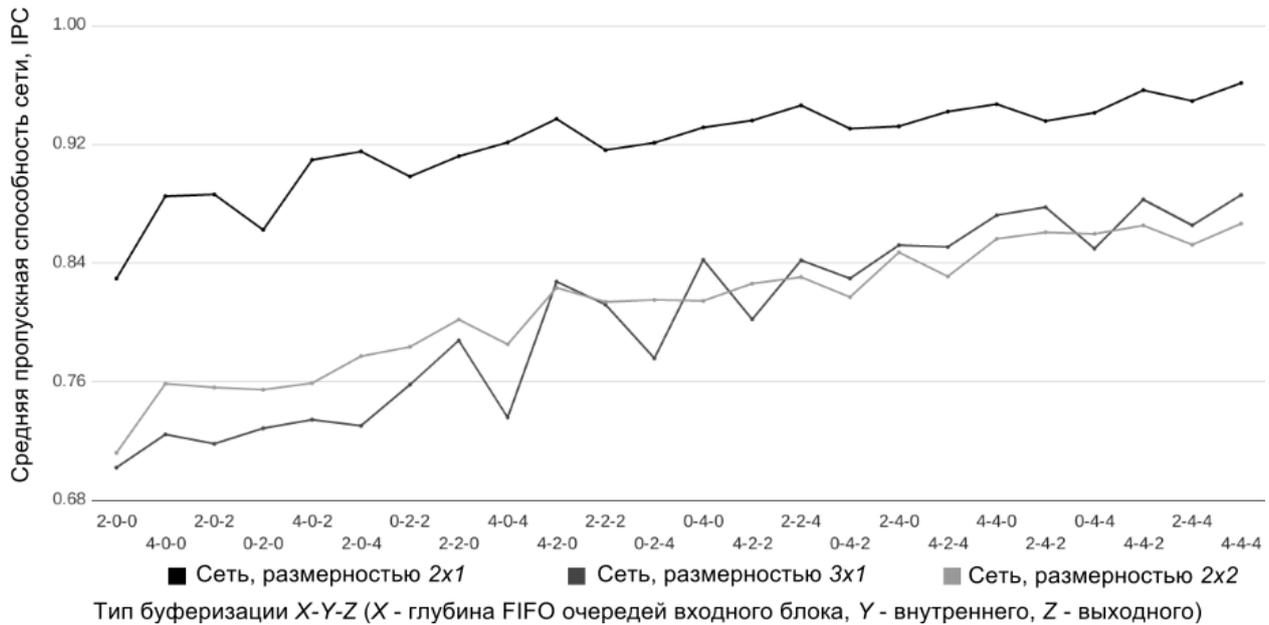


Рис. 4. Тест на производительность. Средняя пропускная способность для сетей 2x1, 3x1 и 2x2

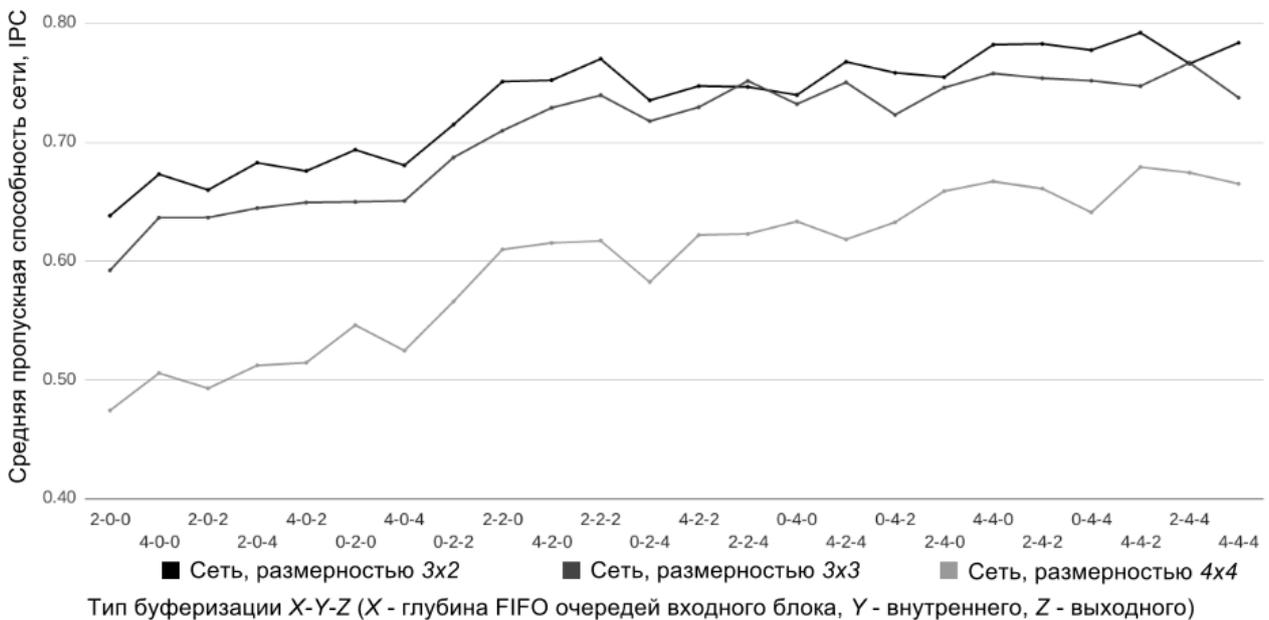
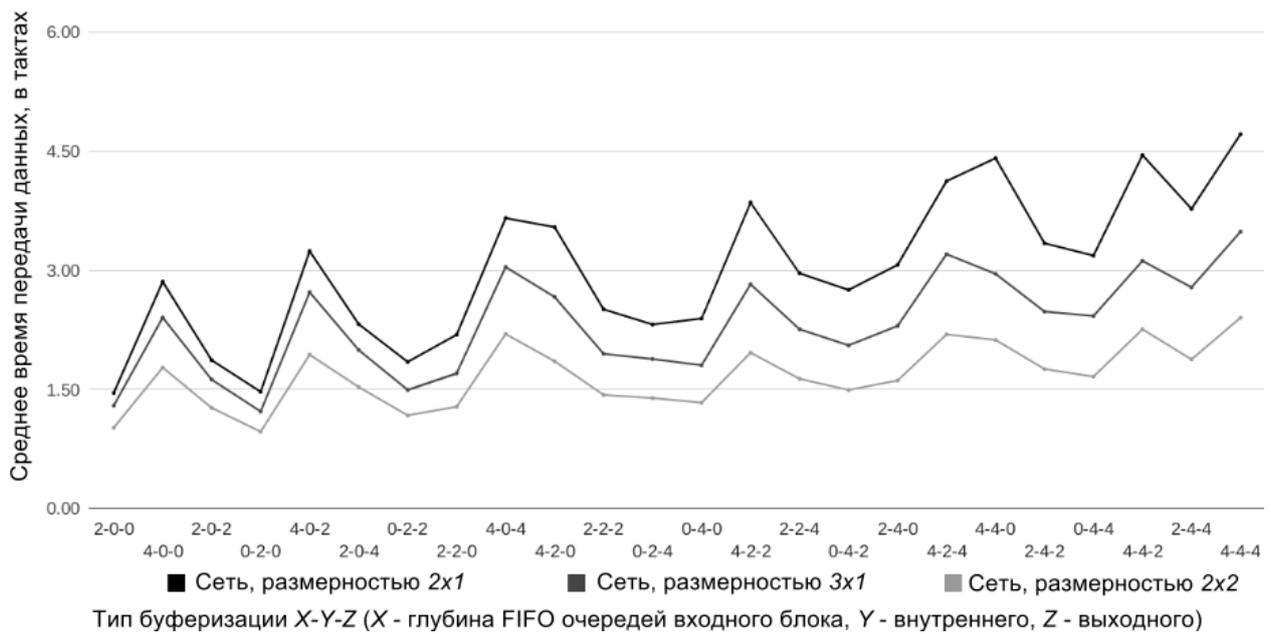
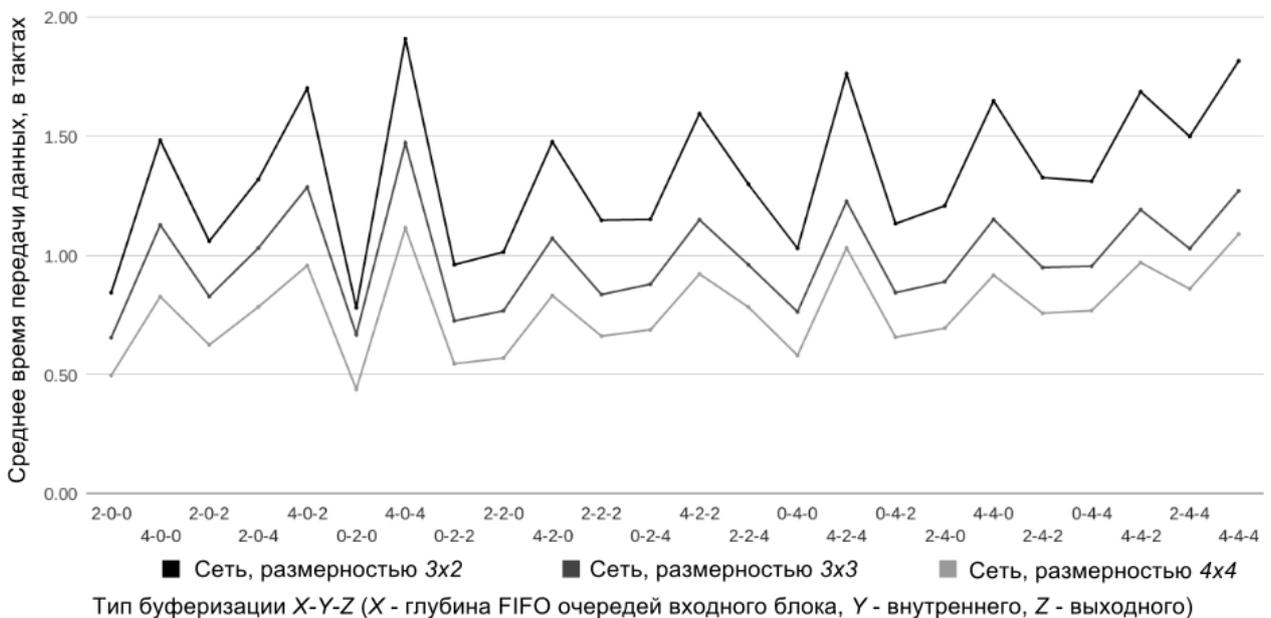


Рис. 5. Тест на производительность. Средняя пропускная способность для сетей 3x2, 3x3 и 4x4



**Рис. 6. Тест на производительность. Среднее время передачи данных для сетей 2x1, 3x1 и 2x2**



**Рис. 7. Тест на производительность. Среднее время передачи данных для сетей 3x2, 3x3 и 3x3**

Тест для анализа производительности представляет собой жесткий вариант использования ресурсов маршрутизации при передаче данных между ядрами. С начала моделирования в каждом узле одновременно на каждом такте формируется запрос на чтение или запись по AXI-протоколу, обращенный к выбираемому произвольно узлу сети, чем обеспечивается равномерная загрузка межсоединения. В силу произвольного выбора узлов, которым адресуется запрос, тест выполняется определенное количество раз для каждого варианта, после чего рассчитываются средние значения пропускной способности и задержек при передаче. Для получения нормированных результатов, выходные характеристики для каждой

сети поделены на количество содержащихся в ней узлов.

Для теста на производительность при отправке 1000 запросов каждым узлом зависимости пропускной способности сети от вариантов буферизации маршрутизаторов представлены на рис. 4 и рис. 5 для сетей с размерностями соответственно 2x1, 3x1, 2x2 и 3x2, 3x3, 4x4 соответственно. Аналогично времена передачи данных отображены на рис. 6 и рис. 7 (похожие результаты получены и для случаев формирования 10, 50 и 100 запросов). По оси абсцисс на графиках отложены типы буферизации, упорядоченные по возрастанию общей глубины используемых FIFO очередей во всей сети. Для

каждого типа это значение было рассчитано по формуле (1). При этом при увеличении количества узлов в сети, распределение общей глубины буферизации по типам начинает меняться. Для более аккуратного анализа варианты сетей (2x1, 3x1, 2x2 с одной стороны и 3x2, 3x3, 4x4 с другой) были разнесены на разные графики в соответствии с похожим характером изменения общей глубины FIFO очередей при различных типах буферизации. Вследствие чего порядок типов на оси абсцисс для разных графиков был изменен. Обобщенно можно говорить как о динамике роста производительности при увеличении общей глубины очередей для сохранения запросов, так и об изменениях результатов в группах с одинаковым или близким по значению количеством используемых буферов. Пример таких групп для простейшей сети с размерностью 2x1 отображен на рис. 8. Исходя из количества и типов маршрутизаторов, каждая сеть из группы 2-0-2, 4-0-0, 0-2-0 всего содержит 8 буферов, а каждая сеть из группы 2-0-4, 4-0-2, 2-2-0, 0-2-2 – 12. Здесь же приведены результаты синтеза в базе библиотеки 130нм для каждого варианта реализации.

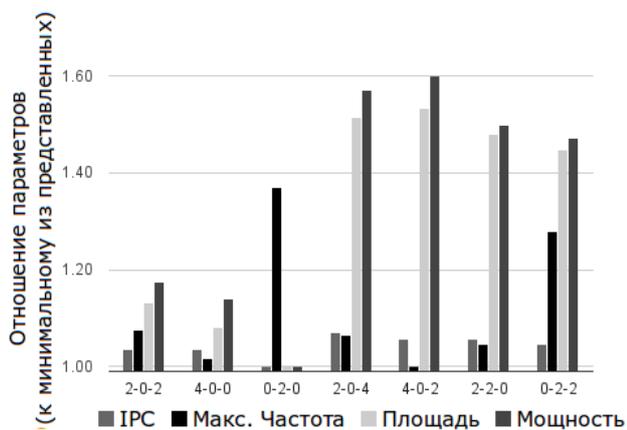


Рис. 8. Результаты синтеза сети размерности 2x1 с различными типами буферизации

По графикам можно выделить несколько закономерностей. В первую очередь наблюдается снижение производительности сетей без внутренней буферизации. Это хорошо видно на группе типов 0-2-0, 4-0-4, 0-2-2. Вариант 4-0-4 практически во всех случаях имеет худшие показатели (например, для сети 3x1 пропускная способность может быть ниже на 11%). При этом не самым лучшим вариантом является использование только внутренней буферизации. Например, из группы 2-2-4, 0-4-0, 4-2-4 производительность сети с типом 0-4-0 не всегда является самой высокой. В дополнение конфигурации, включающие входную и внутреннюю буферизацию (2-2-0, 4-4-0) имеют лучшие показатели, чем

соответствующие типы с внутренней и выходной (0-2-2, 0-4-4).

По результатам синтеза (рис. 8) можно отметить, что варианты сетей, использующие внутреннюю буферизацию, обладают наибольшей максимальной рабочей частотой (на 30-35% выше типов своей группы) и лучшими показателями по потребляемой мощности и занимаемой площади.

## VI. ЗАКЛЮЧЕНИЕ

Необходимо подчеркнуть, что все результаты в работе получены для сети-на-кристалле, а не для выделенного маршрутизатора, что позволяет оценивать эффект влияния различных типов буферизации на весь механизм межсоединения. По результатам исследования можно сделать вывод, что наиболее эффективными вариантами сетей для равномерно использующих ресурсы межсоединения приложений являются те, которые реализуют входную и внутреннюю буферизацию одновременно. Они обладают как более высокой пропускной способностью и более низкой задержкой при передаче данных, так и лучшими физическими характеристиками, полученными после синтеза. В общем, при выборе типа буферизации важно оценивать специфику исполняемых задач, топологию межсоединения и требования по основным физическим и функциональным параметрам.

## ЛИТЕРАТУРА

- [1] Путря Ф.М. Архитектурные особенности процессоров с большим числом вычислительных ядер // Информационные технологии. - 2009. - №4. - С. 2-7.
- [2] Путря Ф.М., Медведев И.А. Анализ механизмов синхронизации потоков для систем-на-кристалле с большим числом вычислительных ядер // Известия вузов. Электроника. - 2011. - №3. - С. 58-63.
- [3] Bjerregaard T., Mahadevan S. A Survey of Research and Practices of Network-on-Chip // ACM Computing Surveys (CSUR). - 2006. - V. 38. - Issue 1.
- [4] Held J., Bautista J., Koehl S. From a few cores to many: A tera-scale computing research overview / Intel White Paper. - 2006.
- [5] Bell S. (et al.) Tile64 Processor - A 64-Core SoC with Mesh Interconnect // Solid-State Circuits Conference. Digest of Technical Papers. - 2008. - P. 588-598.
- [6] Dobkin R. Credit-based Communication in NoCs / Introduction to Network on Chips: VLSI aspects - Technicon - 2007.
- [7] ГУП НПЦ "ЭЛВИС". URL: <http://www.multicore.ru> (дата обращения 10.01.12).
- [8] Lee H., Chang N., Orgas U., Marculescu R. On-chip communication architecture exploration: A quantitative evaluation of point-to-point, bus and network-on-chip approaches // ACM Transactions on Design Automation of Electronic Systems 2007. V. 12. - No. 3. - P. 1-20.