

УДК 004.94

А.Ю. Романов, Д.А. Феськов

Разработка программного симулятора сетей на кристалле

В статье рассмотрена концепция сетей на кристалле (СтнК) как перспективная альтернатива подсистеме связи многопроцессорных систем с шинной архитектурой. В качестве необходимого программного средства оценки параметров производительности СтнК разработан симулятор сетей, рассмотрены его возможности и представлены результаты апробации.

The article presents the concept of networks-on-chip (NoC) as a promising alternative to communication subsystem for multiprocessor systems with bus architecture. The networks simulator developed as a necessary software tool to evaluate NoC performance parameters. Its possibilities are considered and the results of its approbation are given.

Ключевые слова: система на кристалле (СнК), сеть на кристалле (СтнК), маршрутизатор, симулятор.

Введение

Как известно, подсистема соединений между множеством различных по функциональным возможностям интеллектуальных IP-блоков большинства современных систем на кристалле (СнК) представляет собой общую шину. Хотя подход к организации подсистем коммуникаций с такой архитектурой в настоящее время занимает доминирующее положение, а современный уровень развития полупроводниковой технологии способствует увеличению функциональной сложности СнК и, в частности, развитию направления гетерогенных мультипроцессорных СнК, подобные сети имеют ряд недос-

татков, основным из которых является уменьшение их пропускной способности с увеличением количества подключаемых IP-блоков. Таким образом, архитектурные решения с общей шиной в ближайшем будущем становятся мало перспективными [1-3].

В качестве альтернативы в последнее время используется концепция СтнК, которые позволяют создавать гибкие, программируемые и реконфигурируемые сети. Основные тенденции развития и результаты исследований СтнК представлены в работах [2-5].

В общем случае СтнК представляет собой совокупность сетевых адаптеров (Network Interfaces, NI), маршрутизаторов (Routers) и соединений (Links) между ними (рис. 1). При помощи сетевых адаптеров реализуется интерфейс между вычислительными IP-блоками и СтнК. Маршрутизаторы выполняют функцию управления трафиком сети в соответствии с целевым протоколом обмена. Способ соединения маршрутизаторов определяет топологию сети.

К основным преимуществам СтнК можно отнести их масштабируемость, снижение временных затрат на разработку, простоту маршрутизации, а также гибкое управление энергопотреблением системы. Основным недостатком СтнК являются повышенные аппаратные затраты на их реализацию [2, 3]. Поэтому важную роль при проектировании и верификации таких систем играет их моделирование, которое позволяет исследовать их существующие и потенциальные возможности на раннем этапе разработки, что и предопределяет актуальность данной работы.

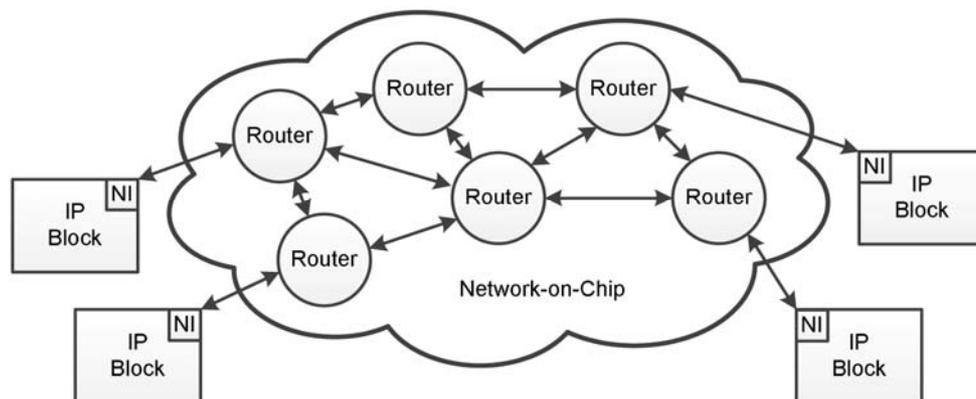


Рис. 1. Обобщенная структура сети на кристалле

Существует два принципиально различных подхода к моделированию СтнК. Первый состоит в эмуляции работы сети и используется для оценки эффективности частных решений при построении СтнК. При этом сеть или ее компоненты реализуются с помощью языков описания аппаратуры (например, Verilog или VHDL), и их функционирование анализируется с помощью специализированных программных средств моделирования аппаратуры (например, пакет Modelsim). Развитием данного подхода является реализация сети в FPGA, где анализ выполняется в реальном масштабе времени на уровне аппаратуры [6]. Данный подход требует больших временных затрат и применим на поздних этапах проектирования, когда разработчик уже определился с будущей архитектурой СтнК, моделью арбитража и передачи трафика.

Второй подход относится к симуляции и состоит в тестировании модели распространения данных в NoC. Так в работе [7] в виде выражения представлена зависимость скорости параллельной обработки данных от параметров СтнК и анализируется влияние на нее задержек при передаче данных с увеличением размерности сети. В работе [8] используется моделирование на сетях Petri в среде симулятора Visual Object Net для анализа конкуренции, взаимодействия и конфликтов данных в коммуникационной среде СтнК. Интерес представляет также работа [9], где передача данных в СтнК представляется в виде набора параллельно выполняемых задач, описанных на языке Си. Недостатком указанных работ является использование ими упрощенной сетевой модели без учета всех параметров сети.

Вышеуказанное обуславливает необходимость разработки симулятора СтнК, позволяю-

щего проводить оценку различных параметров сетей различных видов и топологий, что и является целью данной статьи.

Разработка симулятора OnChipNetwork Simulator

В архитектурных решениях СтнК используются концептуальные принципы компьютерных сетей. Предложено организацию внутренней многоуровневой подсистемы связи между IP-блоками СтнК выполнить в соответствии с базовой сетевой моделью OSI, и положить в основу разработанного симулятора. На рис. 2 представлен поток данных через сеть при обмене сообщениями между двумя IP-блоками, а также соответствие уровней модели OSI уровням представления СтнК [10].

Хотя современные топологии СтнК являются производными от архитектурных решений высокопроизводительной модели параллельных вычислительных систем, только некоторые из них, как Mesh, Torus, Octal и Butterfly-Fat-tree применимы для СтнК [2-4]. Поэтому поддержка этих топологий реализована в OnChipNetworkSimulator.

Функция IP-блоков заключается в генерации сетевого трафика, управление которым выполняется при помощи маршрутизаторов. Кроме того, IP-блоки и маршрутизаторы выполняют сбор статистической информации, которая впоследствии используется для определения параметров производительности СтнК. Поскольку на начальном этапе моделирования получаемая статистика не отображает реальной картины происходящего, в симуляторе предусмотрена поддержка wait-up циклов, в течение которых статистические данные не собираются.

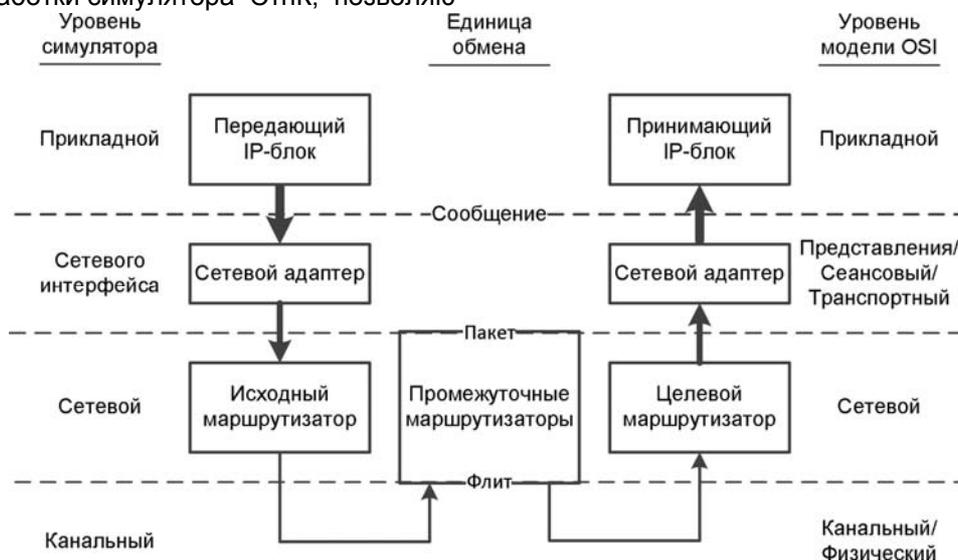


Рис. 2. Сопоставление структуры симулятора и сетевой модели OSI

OnChipNetworkSimulator позволяет моделировать СтнК, структура маршрутизаторов которых аналогична предлагаемой в работе [11], а управление потоком передачи данных выполняется при помощи WFC (Wormhole-Flow-Control) технологии с поддержкой виртуальных каналов. Моделируются Mesh-подобные сети, маршрутизация которых выполняется по XY-алгоритму. Реализована также возможность моделирования GALS (Globally Asynchronous Locally Synchronous) сетей, для которых характерно наличие нескольких временных доменов [2-5]. При моделировании предполагается, что все маршрутизаторы сети тактируются от источников синхронизации с одинаковой частотой, но со случайным сдвигом по фазе.

Блок-схема алгоритма работы разработанного симулятора представлена на рис. 3.

Среди основных входных параметров симулятора можно выделить следующие: топология СтнК, количество IP-блоков, средняя длина сообщения, размер флота, количество физических и виртуальных каналов, количество и размер буферов маршрутизаторов и IP-блоков, количество циклов симуляции, количество warm-up циклов.

В результате выполнения моделирования симулятор предоставляет следующие параметры производительности СтнК: количество отправленных и принятых пакетов, среднее время доставки пакета, среднее количество промежуточных сегментов проходимых пакетом (хопов), пропускная способность маршрутизатора, пропускная способность сети, средняя загрузка физического канала связи, загрузка входных и выходных буферов маршрутизатора и IP-блока.

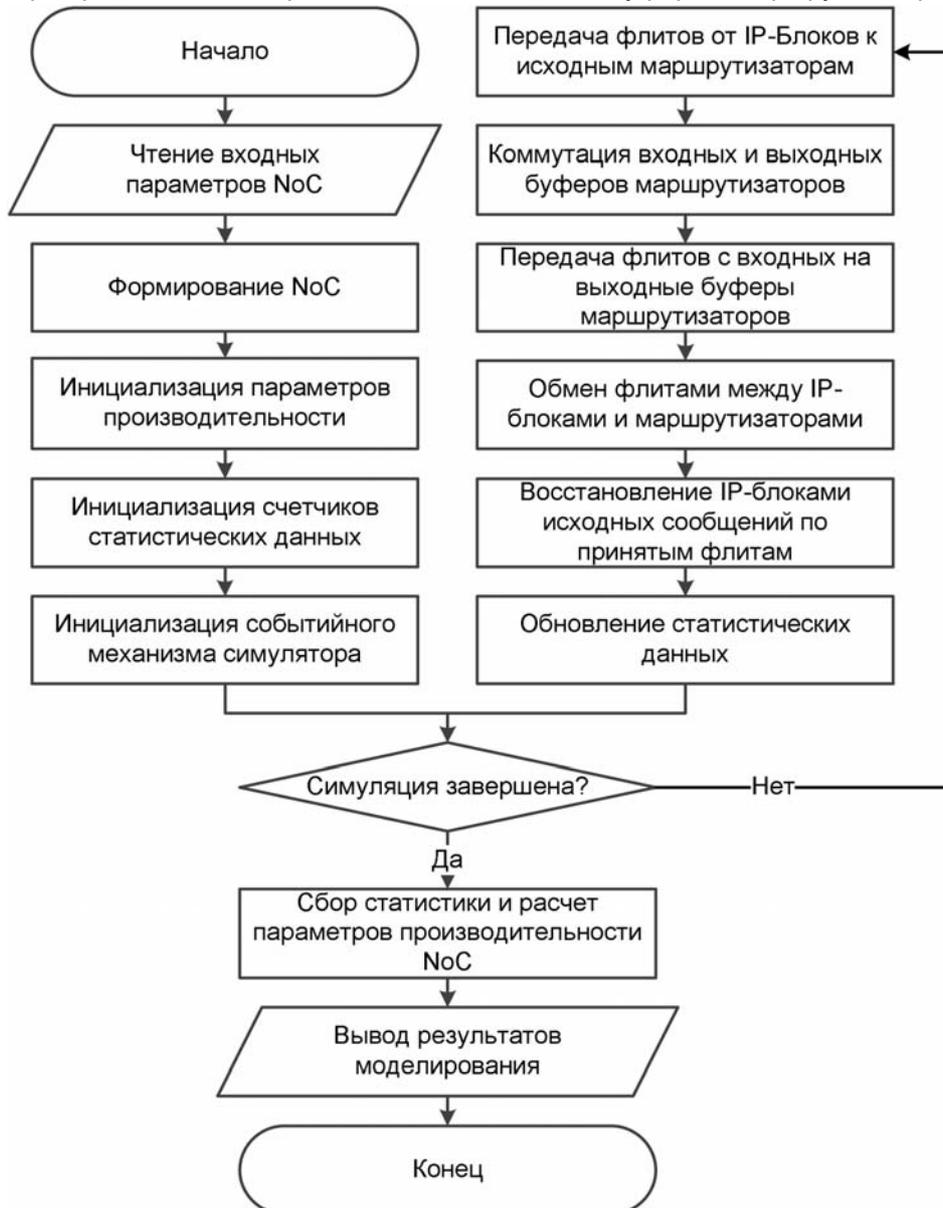


Рис. 3. Блок-схема OnChipNetworkSimulator

Симулятор реализован на языке программирования Java с использованием фреймворка Qt Jambi [12], который предоставляет все преимущества объектно-ориентированного программирования, кроссплатформенности программных решений и скорости их разработки. Полная независимость уровней представления OnChipNetworkSimulator (рис. 2) позволяет выполнять разработку, модификацию и апробацию различных моделей указанных выше типов СтНК и их компонентов с минимальными временными затратами.

Апробация OnChipNetworkSimulator

В качестве одного из примеров для апробации разработанного симулятора было выбрано моделирование зависимости количества хопов от размерности СтНК для топологий mesh и torus. Условия симуляции: варьировалось количество IP-ядер от 4 до 64, интервал между генерацией пакетов – 200 циклов, длина пакета – 160 бит при длине флита – 32 бита, число циклов симуляции – 10000, количество виртуальных каналов маршрутизатора – 2, размер буферов – 3 флита. Результаты моделирования приведены ниже на рис. 4.

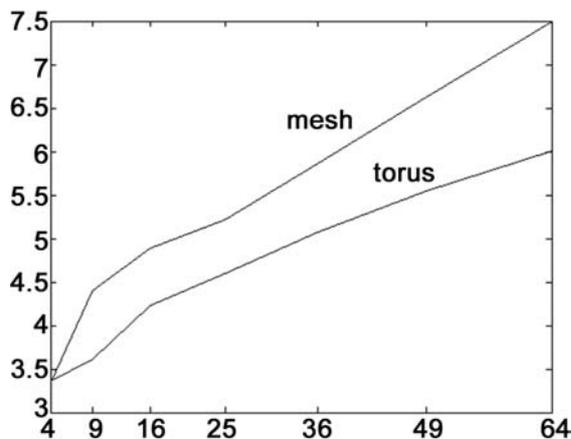


Рис. 4. Зависимость количества хопов от размерности СтНК для топологий mesh и torus

Из рис. 4 следует, что с увеличением размерности сети количество хопов, необходимое для прохождения пакетов, возрастает, при этом для топологии torus рост значительно меньше, что подтверждает выводы, полученные одним из авторов в работе [13].

В качестве второго примера проведена симуляция синхронных СтНК с различными топологиями соединений, маршрутизаторы которых имеют следующие характеристики: подключено 1 IP-ядро, 8 виртуальных каналов, размер буферов равен 1 флиту. Получена зависимость загруженности каналов связи от интервала генерации сообщений каждым IP-ядром, изменя-

ющимся от 25 до 200 циклов, то есть, чем меньше интервал, тем больше сообщений генерируется в сети за время симуляции (рис. 5).

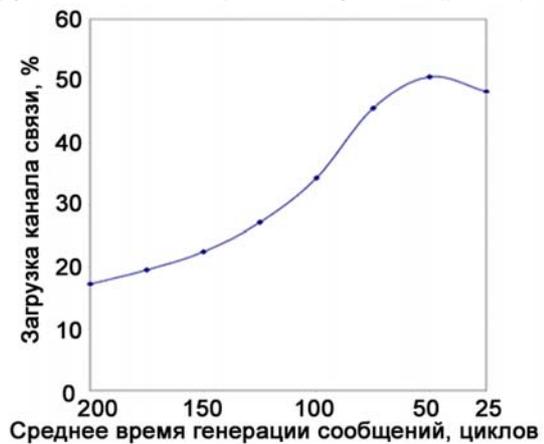


Рис. 5 Зависимость загрузки каналов связи от нагрузки на сеть с топологией mesh

Из приведенного выше графика видно, что загрузка каналов связи СтНК увеличивается с увеличением нагрузки на сеть до достижения четко выраженного экстремума. Его положение отличается для разных топологий СтНК, но не превышает показателя 60%. Характерным также является наличие спада при максимальной загрузке сети, что требует проведения дальнейших исследований.

Выводы

Предложено организацию внутренней многоуровневой подсистемы связи между IP-блоками СтНК выполнить в соответствии с базовой сетевой моделью OSI, и положить в основу разработанного симулятора. Созданы программные решения реализации симулятора на языке программирования Java с использованием фреймворка Qt Jambi, которые позволяют выполнять моделирование топологий Mesh, Torus, Octal и Butterfly-Fat-tree GALS WFC-сетей с поддержкой виртуальных каналов и XY-маршрутизацией. Представлены результаты апробации разработанного симулятора на примерах моделирование зависимостей количества хопов от размерности СтНК для топологий mesh и torus, а также загруженности каналов связи от частоты генерации сообщений IP-ядрами. Из первой зависимости следует, что с увеличением размерности сети количество хопов возрастает, причем для топологии mesh рост значительно больше. А вторая зависимость свидетельствует, что загрузка каналов связи СтНК увеличивается с увеличением нагрузки на сеть до достижения четко выраженного экстремума в районе 60%.

Перспективним направлением дальнейших исследований является разработка модели маршрутизатора, позволяющего выполнять коммутацию нескольких локальных по отношению к нему IP-блоков с возможностью прямого трафика между ними, доработка существующих и внедрение в симулятор новых топологий СтК и их компонентов, поддержка других алгоритмов маршрутизации, а также включение в набор параметров моделирования потребляемую мощность сети.

Литература

1. *Angiolini F., et al.* A layout-aware analysis of networks-on-chip and traditional interconnects for mpsoCs // *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*. – 2007. – Vol. 26, №3. – pp. 421-434.
2. *Axel J., Hannu T.* *Networks on Chip*. – Dordrecht: Kluwer Academic Publishers, 2003. – 303 p.
3. *Benini L., Bertozzi D.* Network-on-chip architectures and design methods // *Computers and Digital Techniques*. – *IEEE Proc.*, 2005. – Vol. 152. – No. 2. – pp. 261-272.
4. *Ankur A., et al.* Survey of Network on Chip (NoC) Architectures & Contributions // *Engineering, Computing & Architecture*. – 2009. – Vol. 3(1). – 15 p.
5. *Bjerregaard T., Mahadevan S.* A survey of research and practices of Network-on-chip // *ACM Computing Surveys*. – 2006. – Vol. 38(1). – 51 p.
6. *Genko N., et al.* NoC emulation: a tool and design flow for MPSoC // *IEEE Circuits and Systems Magazine*. – 2007. – Vol. 7. – No. 4. – pp. 42-51.
7. *Xiaowen Ch., et al.* Speedup Analysis of Data-parallel Applications on Multi-core NoCs. // 8th IEEE Conference on ASIC (ASICON). – Changsha, China, 2009. – pp. 105–108.
8. *Henrique C.F.* Evaluating On-Chip Interconnection Architectures for Parallel Processing. – IEEE International Symposium on Scientific and Engineering Computing. – Brazil, 2008. – pp. 188–193.
9. *Mingsong Lv., et al.* RTNoC: A Simulation Tool for Real-Time Communication Scheduling on Networks-on-Chips. – International Conference on Computer Science and Software Engineering. – 2008. – Vol. 4. – pp. 102-105.
10. *Concer N.* Design and Performance Evaluation of Network-on-Chip Communication Protocols and Architectures: Tesi di dottorato. – Università di Bologna, 2009. – 202 p.
11. *Pandle P., et al.* Design of a Switch for Network on Chip Applications. – IEEE International Symposium on Circuits and Systems. – Bangkok, Thailand, 2003. – Vol. 5. – pp. 217-220.
12. *Официальный веб-сайт “Qt Jambi”*. [Электронный ресурс]. – Режим доступа: <http://qt.nokia.com/>
13. *Романов О.Ю.* Аналіз топологій мереж на чипі // Сучасна інформаційна Україна: інформатика, економіка, філософія: матеріали доповідей конференції, 13-14 травня 2010 р. – Донецьк, 2010. – Т. 1. – С. 407-410.