

Автоматизация низкоуровневого моделирования сетей на кристалле

Е.В. Лежнев

Национальный исследовательский университет «Высшая школа экономики», г. Москва,

elezhnev@hse.ru

Аннотация — Процесс проектирования подсистемы связи сетей на кристалле является сложным и затратным по времени процессом, задачей которого является подбор оптимальных характеристик в заданном диапазоне значений. Низкоуровневое моделирование хоть и является долгим по сравнению с высокоуровневым, но только оно позволяет получить точные характеристики сети. Комплексное исследование, которое применяется в большинстве моделей сетей на кристалле зачастую является избыточным, т.к. за одну итерацию моделирования сети проверяется только один ее параметр. Предлагается низкоуровневая модель, которая позволяет проводить автоматизированное моделирование подсистемы связи, а также алгоритма маршрутизации для сетей на кристалле. Проведенные эксперименты на примере исследования циркулянтных топологий показали корректность и полезность такой модели для различных применений.

Ключевые слова – сеть на кристалле, автоматизация моделирования, низкоуровневое моделирование.

I. ВВЕДЕНИЕ

В настоящее время одним из важнейших направлений исследований в области информатики и вычислительных систем является построение многоядерных процессоров. Переход к многоядерным процессорам позволяет преодолеть снижение производительности при проектировании все более сложных одноядерных систем [1]. В условиях растущего интереса к технологиям построения систем на кристалле и мультипроцессорных систем на кристалле приобретают широкое распространение сети на кристалле. В многоядерном процессоре с небольшим количеством ядер (2–8 ядер) коммуникация между IP-ядрами и другими компонентами происходит с помощью общей шины, которая не способна обеспечить коммуникацию между большим количеством ядер, поскольку снижается ее рабочая скорость, в результате чего шина уже не отвечает требованиям пропускной способности мультипроцессорных систем на кристалле [1].

Процесс проектирования сети на кристалле в общем виде состоит в определении множества различных характеристик сети: выбор основных компонентов сети на кристалле, топологии соединений маршрутизаторов, алгоритма маршрутизации, структуры и особенностей работы маршрутизаторов,

метода управления и арбитража потоками данных в сети [2]. Для анализа влияния на производительность проектируемых сетей на кристалле, принятия тех или иных решений требуется проводить моделирование.

В зависимости от того какими средствами и на каком этапе проектирования СтнК выполняется моделирование можно выделить несколько уровней абстракции моделирования.

1) высокоуровневое моделирование. Обычно осуществляется на ранних этапах проектирования и осуществляется с использованием языков высокого уровня [3, 4, 5];

2) поведенческое моделирование. Используется для оценки сетевого трафика и анализа пропускной способности и задержек передачи пакетов, исследования эффективности алгоритмов маршрутизации в СтнК [6, 7];

3) низкоуровневое моделирование. Применяется для исследования потребления ресурсов чипа и энергопотребления проектируемой системы на уровне ее прототипа, описанного на HDL [8, 9].

Высокоуровневые и поведенческие модели обычно разрабатываются на языках высокого уровня. Все высокоуровневые модели объединяет то, что они не синтезируются в реальную СтнК. Их недостаточно чтобы оценить преимущества того или иного варианта конфигурации СтнК. Они выдвигают лишь гипотезу о том, какие параметры сети лучше.

HDL модели являются более точными, поскольку фактически представляют собой прототип СтнК. При этом моделирование происходит гораздо дольше, так как анализируются события в каждом элементе цепи в каждый момент времени, и для этого используются средства событийного моделирования (например, ModelSim [10]). Это позволяет получить точную модель СтнК, примером которой является Netmaker [11]. Основной недостаток данного подхода заключается в слишком долгом процессе моделирования.

II. НИЗКОУРОВНЕВЫЕ МОДЕЛИ СЕТЕЙ НА КРИСТАЛЛЕ

Низкоуровневые модели сетей на кристалле – это сложные комплексные системы, которые включают в свой состав вычислительные узлы, структуру соединения этих узлов (топологию связи),

маршрутизаторы, которые управляют передачей данных между вычислительными узлами. Процесс разработки сети на кристалле заключается в подборе таких параметров сети, при которых будут достигаться необходимые характеристики. Примером такой модели является модель сети на кристалле Netmaker [11]. Модель разработана на языке SystemVerilog и включает в себя описание маршрутизатора с виртуальными каналами, структуру топологии сети, алгоритмы маршрутизации, системы арбитража доступа к виртуальным каналам, а также другие компоненты.

Такая модель позволяет проводить комплексное исследование сети на кристалле. При этом в большинстве случаев, проводить моделирование всей сети в целом нет необходимости, особенно, когда происходит подбор параметров только какой-то одной ее части [12]. Раздельное моделирование компонентов системы позволит ускорить процесс моделирования сети как одного компонента, так и сети в целом за счет сокращения времени, необходимого для проверки параметров сети.

III. HDL МОДЕЛЬ ПОДСИСТЕМЫ СВЯЗИ СТНК

Была разработана собственная HDL модель, которая позволяет проводить моделирование отдельно подсистемы связи сети на кристалле. Ее структура приведена на рисунке 1.

Модель состоит из 4 компонентов:

- ядро модели – сгенерированные файлы на языке Verilog, которые реализуют подсистему связи исследуемой СТнК;
- модуль конфигурации – производит параметрическую генерацию подсистемы связи СТнК;
- модуль обработки данных – необходим для получения результатов работы модели и

генерации тестовых данных для подачи в модель в реальном времени;

- инфраструктура тестирования – содержит файлы для проведения автоматического тестирования работы модели на заранее подготовленных данных вместо проведения прототипирования.

Ядро модели [13, 14] состоит из двух основных файлов – Router.v, Topology.v и вспомогательного файла Signal_generator.v. Файл Router.v реализует исследуемый алгоритм маршрутизации и фактически представляет собой конечный автомат, на основе которого работает маршрутизатор в сети. С помощью хранимой в маршрутизаторе информации о топологии подсистемы связи, а также сервисной информации, хранящейся в пересылаемых пакетах данных, производится выбор порта, в который нужно отправить пакет, чтобы он достиг узла назначения. Для работы подавляющего большинства алгоритмов маршрутизации в маршрутизаторе необходимо хранить информацию о количестве маршрутизаторов в сети, о количестве портов, в которые возможна пересылка пакета, а также информация о типе соединения ближайших маршрутизаторов. Файл Topology.v реализует исследуемую топологию подсистемы связи. В нем производится соединение всех маршрутизаторов между собой, реализуется передача данных между ними и вывод информации в вычислительный узел. Файл Signal_generator.v является вспомогательным и фактически не является структурным элементом подсистемы связи; он нужен для эмуляции генерации пакетов данных при тестировании работы сети и является заменой вычислительных узлов, которые генерируют и получают данные в сети. Все порты маршрутизаторов для связи с вычислительными узлами соединены с модулем Signal_generator.v.



Рис. 1. Структура HDL модели подсистемы связи

Модуль производит генерацию тестовых пакетов данных, которые подаются в сеть, а также принимает пакеты из сети и производит анализ на возникновение ошибок при прохождении пакета по сети. Так как основная задача состоит в том, чтобы протестировать саму работу топологии сети и алгоритма маршрутизации, то в качестве полезных данных пакета данных выступает 1 бит и определенное количество бит сервисной информации, которая в общем случае содержит номера узлов источника и приемника пакета данных.

Так как `Signal_generator.v` является вспомогательным файлом, то моделирование подсистемы связи является избыточным и вносит искажения в занимаемое количество ресурсов кристалла при прототипировании. При этом можно вычислить сколько этот модуль занимает места на чипе и учесть это при анализе занимаемого места подсистемой связи.

Оценка влияния вносимой избыточности модулем `Signal_generator.v` в прототипирование подсистемы связи приведена на рисунках 2 и 3.

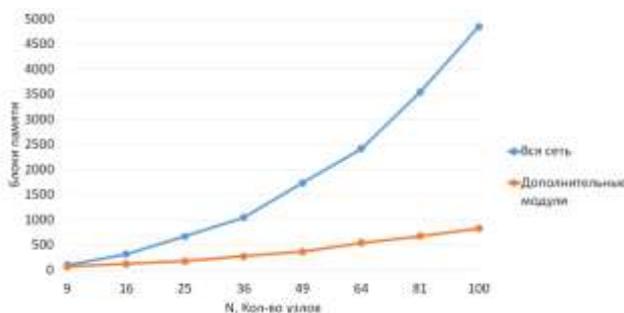


Рис. 2. Зависимость использования регистров ПЛИС для всей сети и вспомогательных модулей от количества узлов в сети.

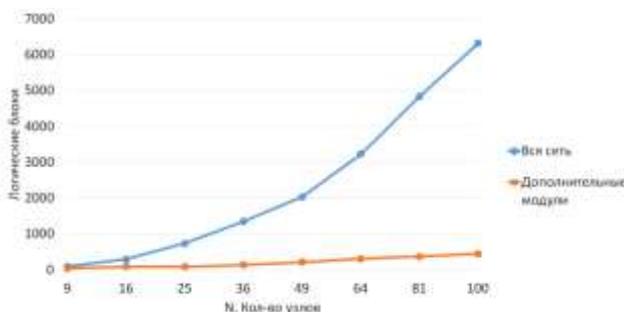


Рис. 3. Зависимость использования логических ресурсов ПЛИС для всей сети и вспомогательных модулей от количества узлов в сети.

Из рисунка следует, что модуль занимает много меньше ресурсов, чем подсистема связи. Численное вычисление ресурсов, занимаемых модулем `Signal_generator.v` можно представить в виде формул:

$$U_{alm} = 0,0034 * x^2 + 4,2420 * x + 2,7879 \quad (1)$$

$$U_{reg} = 0,0076 * x^2 + 7,7416 * x + 9,5715 \quad (2)$$

где x – количество узлов в сети.

Если для сетей с количеством узлов менее 10, вспомогательные модули занимают значительно место от общего количества занимаемых сетью ресурсов – более 60 % регистров и более 40 % логических блоков, то при увеличении количества узлов до 100, вспомогательные модули уже занимают 17 % регистров и 7 % логических блоков.

Для проведения моделирования подсистемы связи разработаны специализированные тестбенчи. Они необходимы для проведения предварительного тестирования работы модели. Дополнительно реализована возможность вызывать файлы для проведения тестирования, разработанные на высокоуровневом языке с помощью интерфейса DPI.

IV. АВТОМАТИЗАЦИЯ ПРОЦЕССА МОДЕЛИРОВАНИЯ ПОДСИСТЕМЫ СВЯЗИ

Для каждой анализируемой конфигурации подсистемы связи необходимо создавать свои файлы ядра модели, что является рутинной и ресурсоемкой по времени задачей. Для автоматизации этого процесса был разработан модуль `Configurator`. Данный модуль разработан на языке C#. Задача данного модуля – на основе информации о топологии и алгоритме маршрутизации произвести генерацию файлов ядра модели. На данном этапе разработчику необходимо произвести разовую разработку кодов ядра модели и тестбенчей. При последующих итерациях моделирования модулю конфигурации указывается какие параметры в файлах необходимо изменить, чтобы привести их в соответствие исследуемой конфигурации. Также возможно задать список конфигураций для моделирования. В дальнейшем модуль конфигурации может в автоматическом режиме проводить генерацию всех необходимых Verilog файлов и производить запуск моделирования и прототипирования подсистемы связи с помощью TCL команд. Дополнительно в модуле конфигурации реализована возможность высокоуровневой проверки корректности структуры подсистемы связи и алгоритма маршрутизации путем генерации путей маршрутизации из нулевого маршрутизатора во все остальные.

Файл `Signal_generator.v` позволяет подавать сгенерированные пакеты для проверки работы подсистемы связи в автоматическом режиме. Для получения результатов работы сети в виде файла был разработан модуль обработки данных `Data receiver`. Данный модуль также разработан на высокоуровневом языке C#. Модуль обработки данных позволяет подключиться к модулю `Signal_generator.v`, получать результаты моделирования и сохранять их в файл для дальнейшего анализа. Также реализована возможность подавать данные в ручном режиме с помощью UART интерфейса, реализованного в `Signal_generator.v`.

V. АПРОБАЦИЯ РАЗРАБОТАННОЙ МОДЕЛИ СТНК

Для тестирования предложенной модели было проведено моделирование сетей с топологией циркулянт с количеством образующих 2 и количеством узлом от 9 до 100 [15]. Была получена оценки занимаемых ресурсов подсистемой связи на ПЛИС в зависимости от выбранного алгоритма маршрутизации. Результаты апробации приведены на рисунках 5 и 6.

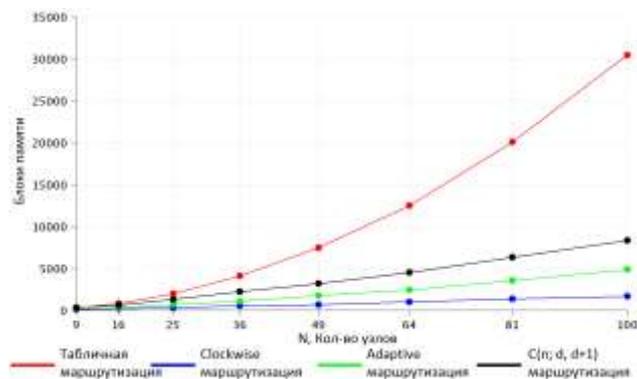


Рис. 5. Сравнение использования регистров алгоритмами для топологий с параметрическими описаниями $C(n; 1, s_1)$ и $C(n; d, d + 1)$.

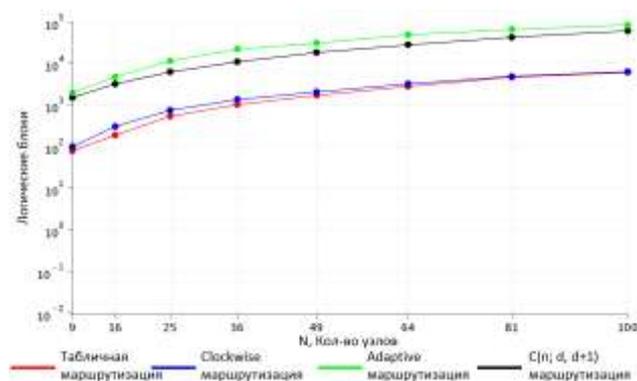


Рис. 6. Сравнение использования ALM блоков алгоритмами для топологий с параметрическими описаниями $C(n; 1, s_1)$ и $C(n; d, d + 1)$.

Полученные результаты моделирования соответствуют ожиданиям, что подтверждает корректность работы разработанной модели.

VI. ЗАКЛЮЧЕНИЕ

Разработанная низкоуровневая модель подсистемы связи СтнК позволяет автоматизировать процесс разработки однотипного кода моделей подсистем связи системы на кристалле с изменением определенных параметров, а также процесс моделирования системы, получения и первичной обработки результатов моделирования. Следствием автоматизации разработки и проведения моделирования является увеличение скорости моделирования за счет уменьшения времени, необходимого для подготовки модели. Проведенная апробация модели на примере исследования количества занимаемых ресурсов подсистемой связи в зависимости от выбранного алгоритма маршрутизации

показала работоспособность модели и корректность создаваемых ею HDL файлов.

ЛИТЕРАТУРА

- [1] Nychis G., Fallin C., Moscibroda T., Mutlu O. Next Generation On-Chip Networks: What Kind of Congestion Control Do We Need? // Hotnets-IX: Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, ACM, 2010. P. 1–6.
- [2] Romanov O., Lysenko O. The Comparative Analysis of the Efficiency of Regular and Pseudo-optimal Topologies of Networks-on-Chip Based on Netmaker // Proceedings of Mediterranean Conference on Embedded Computing, 2012. P. 13–16.
- [3] Kourdy R., Yazdanpanah S., Rad M. Using the NS-2 network simulator for evaluating multi-Protocol label switching in network-on-Chip // Second International Conference on Computer Research and Development, 2010. P. 795–799.
- [4] Jiang N., Michelogiannakis G., Becker D., Towles B., Dally W.J. BookSim 2.0 User's Guide, 2013. 10 p.
- [5] Прилепко П.М., Романов А.Ю., Лежнев Е.В. Модификация высокоуровневой модели NoCModel 2.0 для моделирования сетей на кристалле с циркулянтными топологиями // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. Вып. 4. С. 23–30.
- [6] Genko N., Atienza D., Benini L. Feature – Noc Emulation: A Tool and Design Flow for MPSoC // IEEE Circuits and Systems Magazine, 2008. V. 7. № 4. P. 42–51.
- [7] Bertozzi D., Jalabert A., Benini L. NoC Synthesis Flow for Customized Domain Specific Multiprocessor Systems-on-Chip // IEEE Transactions on Parallel and Distributed Systems, 2005. V. 16. № 2. P. 113–129.
- [8] Romanov A., Ivannikov A. SystemC Language Usage as the Alternative to the HDL and High-level Modeling for NoC Simulation // International Journal of Embedded and Real-Time Communication Systems (IJERTCS), 2018. V. 9. № 2. P. 18–31.
- [9] Duan X., Li Y. A Multiphase Routing Scheme in Irregular Mesh-Based NoCs // Fourth International Symposium on Parallel Architectures, Algorithms and Programming, 2011. P. 277–280.
- [10] Oukaira A. et al. Simulation, Validation and FPGA Implementation of a Ring Oscillator Sensor for Thermal Management and Monitoring // Procedia Computer Science, 2019. V. 155. P. 83–88.
- [11] Mullins R., West A., Moore S. Low-latency virtual-channel routers for on-chip networks // Proceedings 31st Annual International Symposium on Computer Architecture, 2004. P. 188–197.
- [12] Venkataraman N., Kumar R. Design and analysis of application specific network on chip for reliable custom topology // Computer Networks, 2019. V. 158. P. 69–76.
- [13] Lezhnev E.V., Romanov A.Yu. HDLNoCGen. URL: <https://github.com/evgenii-lezhnev/HDLNoCGen> (дата обращения 29.05.2021).
- [14] Романов А.Ю., Лежнев Е.В. Генератор Verilog кода подсистемы связи сетей на кристалле Verilog Code Generator of Communication Subsystem for Networks-on-Chip (HDLNoCGen). Свидетельство о регистрации программы для ЭВМ RU 2021616623, 23.04.2021. Заявка № 2021615675 от 20.04.2021.
- [15] Romanov A. Development of routing algorithms in networks-on-chip based on ring circulant topologies // Heliyon, 2019. V. 5. № 4. P. e01516.

Automation of Low-Level Modeling of Networks-on-Chip

E.V. Lezhnev

HSE University, Moscow,

elezhnev@hse.ru

Abstract — The process of designing a network-on-chip communication subsystem is a complex and time-consuming process, the task of which is to select the optimal characteristics in a given range of values. Although low-level modeling is time-consuming compared to high-level, but only this process allows to get the exact characteristics of the network. Comprehensive research, which is used in most models of networks-on-chip, is often redundant, because during one simulation iteration, only one of its parameters is checked. A low-level model is proposed that allows automated modeling of the communication subsystem, as well as the routing algorithm for networks-on-chip. The experiments carried out on the example of the study of circulant topologies, which showed the correctness and usefulness of such a model for various applications.

Keywords — network-on-chip, simulation automation, low-level simulation.

REFERENCES

- [1] Nychis G., Fallin C., Moscibroda T., Mutlu O. Next Generation On-Chip Networks: What Kind of Congestion Control Do We Need? // Hotnets-IX: Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, ACM. 2010. P. 1–6.
- [2] Romanov O., Lysenko O. The Comparative Analysis of the Efficiency of Regular and Pseudo-optimal Topologies of Networks-on-Chip Based on Netmaker // Proceedings of Mediterranean Conference on Embedded Computing. 2012. P. 13–16.
- [3] Kourdy R., Yazdanpanah S., Rad M. Using the NS-2 network simulator for evaluating multi-Protocol label switching in network-on-Chip // Second International Conference on Computer Research and Development. 2010. P. 795–799.
- [4] Jiang N., Michelogiannakis G., Becker D., Towles B., Dally W.J. BookSim 2.0 User's Guide. 2013. 10 p.
- [5] Prilepko P.M., Romanov A.Yu., Lezhnev E.V. Modification of a High-Level NoCModel 2.0 for Modeling Networks-on-Chip with Circulant Topologies // Problems of Perspective Micro- and Nanoelectronic Systems Development. 2020. Issue 4. P. 23–30.
- [6] Genko N., Atienza D., Benini L. Feature – Noc Emulation: A Tool and Design Flow for MPSoC // IEEE Circuits and Systems Magazine. 2008. V. 7. № 4. P. 42–51.
- [7] Bertozzi D., Jalabert A., Benini L. NoC Synthesis Flow for Customized Domain Specific Multiprocessor Systems-on-Chip // IEEE Transactions on Parallel and Distributed Systems. 2005. V. 16. № 2. P. 113–129.
- [8] Romanov A., Ivannikov A. SystemC Language Usage as the Alternative to the HDL and High-level Modeling for NoC Simulation // International Journal of Embedded and Real-Time Communication Systems (IJERTCS). 2018.V. 9. № 2. P. 18-31.
- [9] Duan X., Li Y. A Multiphase Routing Scheme in Irregular Mesh-Based NoCs // Fourth International Symposium on Parallel Architectures, Algorithms and Programming. 2011. P. 277–280.
- [10] Oukaira A. et al. Simulation, Validation and FPGA Implementation of a Ring Oscillator Sensor for Thermal Management and Monitoring // Procedia Computer Science. 2019. V. 155. P. 83–88.
- [11] Mullins R., West A., Moore S. Low-latency virtual-channel routers for on-chip networks // Proceedings 31st Annual International Symposium on Computer Architecture. 2004. P. 188–197.
- [12] Venkataraman N., Kumar R. Design and analysis of application specific network on chip for reliable custom topology // Computer Networks. 2019. V. 158. P. 69–76.
- [13] Lezhnev E.V., Romanov A.Yu. HDLNoCGen. URL: <https://github.com/evgenii-lezhnev/HDLNoCGen> (access date 29.05.2021).
- [14] Romanov A.Yu, Lezhnev E.V. Generator Verilog koda podsystemy svyazi setej na kristalle Verilog Code Generator of Communication Subsystem for Networks-on-Chip (HDLNoCGen). Certificate of registration of the computer program RU 2021616623, 23.04.2021. Request № 2021615675 from 20.04.2021.
- [15] Romanov A. Development of routing algorithms in networks-on-chip based on ring circulant topologies // Heliyon. 2019. V. 5. № 4. P. e01516.