

МЕТОД МОДЕЛИРОВАНИЯ РЕКОНФИГУРИРУЕМЫХ СЕТЕЙ НА КРИСТАЛЛЕ

Проведено огляд існуючих підходів до моделювання мереж на кристалі. Запропоновано метод агентного моделювання реконфігуруємих мереж на кристалі при будь-якому рівні абстракції.

A review of existing approaches to modeling networks on chip is presented. The method for agent based modeling of reconfigurable networks on chip at any level of abstraction is proposed.

Введение

В связи с интенсивным развитием технологий производства интегральных микросхем (ИМС) остро встает вопрос объединения компонентов в больших системах на кристалле. Наиболее распространенный подход, использующий принцип общей шины, показывает отсутствие масштабируемости и уменьшение пропускной способности с увеличением числа соединяемых элементов. Одним из методов устранения подобных недостатков может стать использование сетевых технологий для обмена данными между подсистемами ИМС. Впервые концепция сети на кристалле (СтнК) сформулирована в [1]. Такой подход к организации связи в ИМС обладает преимуществами масштабируемости (с увеличением размера сети растет ее пропускная способность) и параллелизма (данные в разных сегментах сети передаются одновременно). Основные тенденции развития СтнК отображены в [2,3]. Подходы к реализации сетевого взаимодействия внутри микросхем с реконфигурированной архитектурой (FPGA) описаны в [4,5].

Использование универсальной статической архитектуры подсистемы связи для решения всего спектра возможных задач приводит к избыточности и, как следствие, к увеличению энергопотребления и необходимых для реализации аппаратных ресурсов. Для решения проблемы избыточности в [6] предложена сеть с реконфигурируемой архитектурой. Такой подход позволяет в реальном времени адаптировать структуру сети к требованиям пропускной способности и потребляемой мощности.

Важную роль в процессе проектирования и верификации СтнК играет моделирование. Основные результаты исследований, принятых в этом направлении, отражены в [7-13]. В [8] утверждается, что на момент пуб-

ликации работы не существовало средств, позволяющих моделировать реконфигурируемые СтнК при любом уровне абстракции. Анализ более поздних источников [10] подтвердил эту информацию.

Выше изложенное предопределяет актуальность данной работы и диктует необходимость разработки метода моделирования реконфигурируемых СтнК на разных уровнях абстракции, что и является целью предлагаемой статьи.

Общие сведения о СтнК

Обобщенная структура СтнК показана на рис.1. Для обозначения подсистем ИМС использован термин ГВМ, что расшифровывается как Готовый Вычислительный Модуль. Непосредственно СтнК состоит из маршрутизаторов М, соединенных согласно определенной топологии. Блоки М управляют потоком пакетов данных, проходящим через порты ввода/вывода М, осуществляя маршрутизацию на основании адреса назначения указанного в заголовке пакета. Сетевой интерфейс СИ преобразует данные из формата передачи характерного для ГВМ в поток дискретных пакетов и наоборот.

Использованием описанной структуры подсистемы связи достигаются следующие преимущества:

- масштабируемость – для увеличения пропускной способности сети достаточно добавить новые маршрутизаторы и подключить к ним необходимые ГВМ;
- параллелизм – данные в разных сегментах сети передаются одновременно, что приводит к увеличению пропускной способности подсистемы связи.

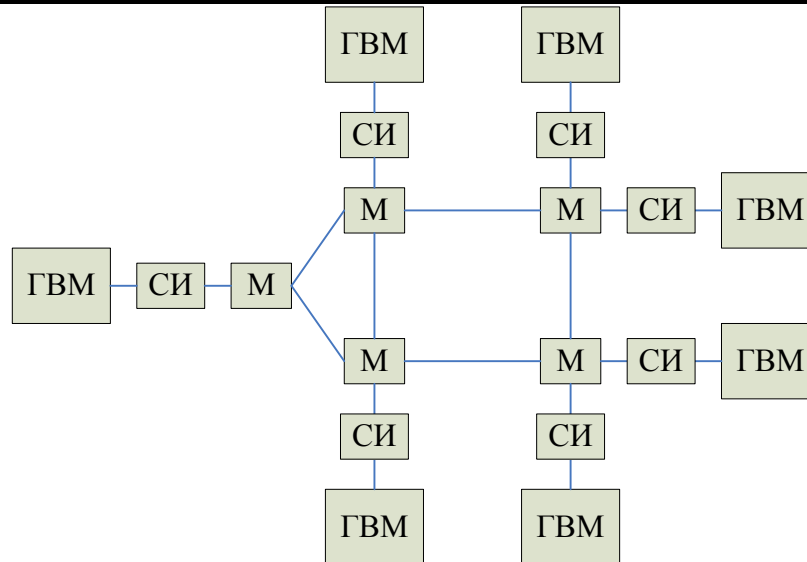


Рис. 1. Структура сети на кристалле

Структура СтнК с реконфигурируемой архитектурой

Кроме достоинств, описанных в предыдущем разделе, СтнК присущи следующие недостатки [1,2]:

- сложность разработки;
- большое количество аппаратных ресурсов необходимых для реализации;
- большая потребляемая мощность.

Первый из недостатков является общим свойством систем, предназначенных для эффективного решения сложных задач. Путем решения проблемы является создание специализированных САПР, облегчающих работу инженера.

Второй и третий недостатки проистекают из статичности архитектуры СтнК. В [6] показано, что использование универсальной статической архитектуры подсистемы связи для решения всего спектра возможных задач приводит к избыточности и, как следствие, к увеличению энергопотребления и необходи-

мых для реализации аппаратных ресурсов. Для решения проблемы в [6] предлагается использовать динамически реконфигурируемую архитектуру СтнК, которая позволяет в реальном времени адаптировать ее структуру к требованиям пропускной способности и потребляемой мощности (рис. 2).

Как видно из рис.2, реконфигурируемая архитектура СтнК состоит из самой сети, подсистемы мониторинга контролируемых параметров и блока модификации структуры. На основании ошибки управления e , характеризующей отклонение требуемых параметров сети от текущих, блок модификации осуществляет минимизацию e путем изменения структуры СтнК. В качестве контролируемых параметров могут выступать пропускная способность, необходимые для реализации ресурсы ИМС и потребляемая мощность.

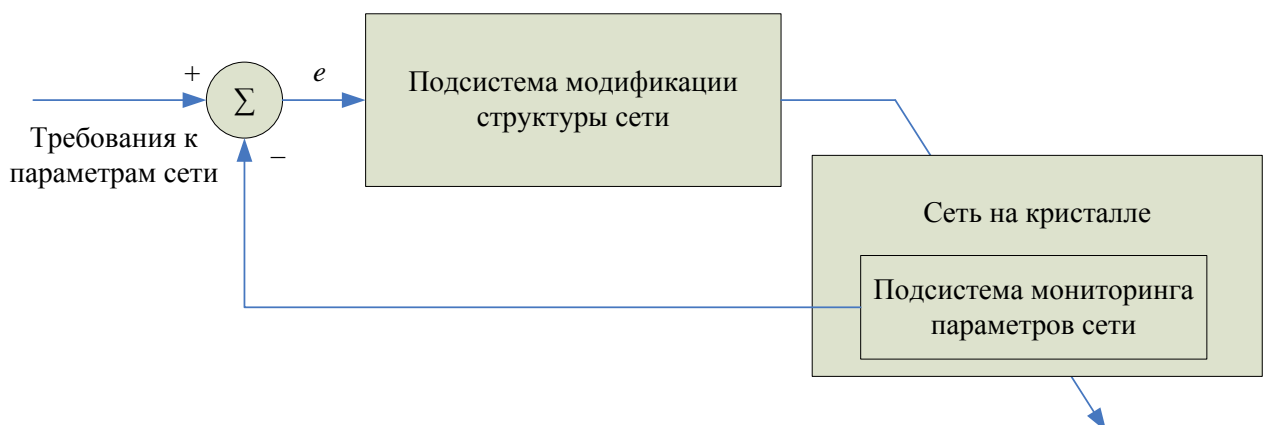


Рис. 2. Обобщенная структура СтнК с реконфигурируемой архитектурой

Действия блока модификации зависят от выбора параметров, подлежащих оптимизации. Например, для управления энергопотреблением и необходимыми для реализации ресурсами можно динамически изменять размер очередей портов ввода-вывода маршрутизаторов [14]. Управление пропускной способностью осуществляется путем модификации топологии сети и таблиц маршрутизации блоков М [6].

Обоснование выбора метода моделирования реконфигурируемой СтнК

В контексте моделирования СтнК будем различать следующие уровни абстракции [7]:

- системный (высокий) – моделирование динамического изменения структуры сети – модификация количества маршрутизаторов и топологии;
- поведенческий (средний) – моделирование обмена данными на пакетном уровне – пропускной способности, задержек, алгоритмов маршрутизации, потерь пакетов;
- физический (низкий) – моделирование на регистровом/транзисторном уровнях – исследование потребляемой мощности и требуемых для реализации ресурсов ИМС.

В [7-9] предлагаются средства моделирования СтнК лишь на поведенческом (среднем) уровне абстракции. Используемый метод – дискретно-событийная имитация. Для исследования особенностей работы сети на физическом уровне в [10-13] разработаны точные RTL (Register Transfer Level) модели с использованием языков описания цифровых схем – Verilog и VHDL. Большим недостатком такого подхода является значительные временные затраты при моделировании на компьютере либо необходимость в наличии дорогостоящих устройств для аппаратурной имитации. Таким образом, ни один из предложенных методов не позволяет проводить моделирование СтнК при любом уровне абстракции, а преобразование модели от одного уровня к другому требует значительных временных затрат.

Проведенный обзор литературы показал, что на сегодняшний день отсутствуют средства для моделирования реконфигурируемых СтнК. Такая ситуация обусловлена тем, что существующие подходы к поведенческому

моделированию СтнК используют дискретно-событийную имитацию (средний уровень абстракции), в то время как реконфигурируемая архитектура требует принятия решений о модификации подсистем и связей между ними (системный уровень), что затруднительно при использовании дискретно-событийного подхода.

В [15] выполнен сравнительный анализ методов, применяемых в имитационном моделировании. Результаты приведены на рис.3.

Из рис.3 видно, что наиболее универсальным является агентный подход к имитационному моделированию. Суть его состоит в том, что предметная область представляется в виде множества агентов, взаимодействующих между собой. Разработчиком модели описываются правила создания, уничтожения и изменения агентов. При этом под термином “агент” понимается некоторый объект (в терминах ООП), обладающий памятью и способностью принимать решения, а значит и собственным поведением разного уровня сложности. Внутренняя структура агента может быть описана разными способами – от формальной логики до нейронных сетей. В момент запуска процесса моделирования, каждый агент начинает функционировать согласно индивидуальному алгоритму работы, и глобальное поведение системы возникает, как результат взаимодействия всего множества агентов. Поэтому агентное моделирование (АМ) называют еще моделированием снизу вверх. Очевидно, что АМ существенно децентрализовано. В отличие от системной динамики или дискретно-событийных моделей, здесь нет такого места, где централизованно определялось бы поведение (динамика) системы в целом. Из этого следуют два важных следствия: во-первых, отсутствие координационного центра создает предпосылки для распараллеливания (ускорения) процесса моделирования, во-вторых, появляется возможность постепенно вносить коррективы в алгоритм работы агента, тем самым, детализируя

модель. Таким образом, наличие нескольких сценариев функционирования агента разного уровня сложности дает возможность моделировать работу системы на различных уровнях абстракции.



Рис. 3. Подходы в имитационном моделировании на шкале уровня абстракции

К недостаткам вычислительных систем с агентами следует отнести значительно большую опасность взаимоблокировок, чем в других системах моделирования. Решением проблемы может стать использование готовых САПР АМ (например, AnyLogic) [15], в которых проблема взаимодействия агентов решена разработчиками.

По мнению авторов работы, использование агентного подхода для имитационного моделирования СтНК позволит достичь следующих преимуществ:

- моделирование работы СтНК на любом уровне абстракции; согласно [15] ни системная динамика, ни дискретно-событийный подход не предоставляют такой возможности;
- моделирование реконфигурируемых СтНК. Реконфигурируемая архитектура предполагает мониторинг параметров СтНК и принятие решений о модификации ее подсистем в соответствии с требованиями, предъявляемыми к качеству обслуживания в сети; дискретно-событийный подход не дает возможности принятия

решений на системном уровне, а системная динамика не позволяет эффективно моделировать информационные системы с пакетной передачей данных [15]; таким образом, для моделирования адаптивной структуры реконфигурируемой СтНК единственным приемлемым решением является использование агентного подхода;

- одновременное функционирование всех агентов в мультиагентной системе приводит к уменьшению времени моделирования по сравнению с дискретно-событийным подходом, в котором алгоритм имитации реализуется, как правило, в виде последовательной программы, обрабатывающей заявки.

Метод моделирования реконфигурируемой СтНК при любом уровне абстракции

Для реализации представленного в предыдущем разделе метода моделирования реконфигурируемых СтНК авторами предложен подход, содержащий следующие положения:

- все ресурсы реконфигурируемой СтнК (ГВМ, сетевые интерфейсы, маршрутизаторы, подсистемы мониторинга и оптимизации) представляются в виде агентов, функционирующих одновременно;
- индивидуальный алгоритм работы каждого агента описывается при помощи конечного автомата;
- в качестве единого языка описания структуры и связей между агентами используются UML-диаграммы классов;

Рассмотрим предложенную концепцию более подробно.

В зависимости от уровня абстракции модели количество используемых типов агентов может меняться. Например, агентное представление пакетов является приемлемым при моделировании пропускной способности и задержек в СтнК, когда исследователя интересует поведение отдельно взятых пакетов и их групп. Такой подход позволяет придать пакетам индивидуальные свойства, например возможность случайного изменения содержимого, что весьма полезно при моделировании помех на линиях связи. С другой стороны, при низкоуровневой имитации необходимость в агентном представлении пакетов исчезает, поскольку при данной степени абстракции важны другие характеристики системы, такие как потребляемая мощность и взаимное расположение компонентов СтнК на кристалле.

Индивидуальный алгоритм работы каждого агента предлагается задавать в виде конечного автомата. Высокий уровень формализации теории конечных автоматов обуславливает ее применение для описания дискретных цифровых систем и облегчает верификацию созданной модели. С понижением уровня абстракции к алгоритму работы агента добавляются все новые детали, точность модели увеличивается, а сложность соответствующего конечного автомата возрастает.

Предлагаемый метод не ограничивает подходы к описанию функционирования агента одной лишь теорией конечных автоматов. Поскольку концепция агента предполагает инкапсуляцию его внутренней структуры, алгоритм работы отдельно взятого компонента СтнК может быть описан любым удобным для разработчика образом. При условии соблюдения неизменности интерфейса, такая замена внутреннего содержимого агента не

повлияет на корректность функционирования остальных подсистем СтнК.

Одним из качеств, обуславливающих эффективное моделирование сложных систем, является использование унифицированного языка представления модели. В предлагаемом методе в качестве такого языка используется UML 2.0. Сделанный выбор обусловлен тем, что UML является наиболее распространенной и удобной нотацией для описания статических и динамических объектно-ориентированных систем, к которым, безусловно, можно отнести и мульти-агентные модели [15]. Для представления структуры СтнК в предложенном методе используется UML-диаграмма классов. На рис. 4 показана СтнК на системном уровне абстракции в терминах UML.

Отношения между объектами задаются при помощи технологий агрегирования и осведомленности. Агрегирование подразумевает, что один объект (агент) владеет другим и несет за него ответственность (в терминах создания, удаления и изменения). Например, из рис. 4 видно, что объект СтнК содержит объекты маршрутизаторов, сетевых интерфейсов, ГВМ, подсистем мониторинга и модификации структуры сети. Таким же образом можно описывать буферы ввода-вывода как составную часть, например, маршрутизатора на более низком уровне абстракции.

Различие между агентом и объектом при таком подходе состоит в том, что агент имеет более сложное индивидуальное поведение и способен принимать решения.

Говоря же об осведомленности, имеется в виду, что объекту (агенту) известно о другом объекте (агенте). Осведомленные агенты могут запрашивать друг у друга операции (делегация), но они не несут никакой ответственности друг за друга (создание, удаление, модификация). Осведомленность – это более слабое отношение по сравнению с агрегированием, поскольку оно предполагает гораздо менее тесную связь между объектами (агентами). В предлагаемом методе осведомленность реализуется в виде ссылки на объект (агент) и используется для задания и динамической модификации топологии СтнК. Каждый маршрутизатор осведомлен о других маршрутизаторах и сетевых интерфейсах, подсоединенных к нему. Такой подход позволяет легко моделировать реконфигури-

руемую топологию сети, поскольку для ее изменения достаточно обновить соответствующие ссылки.

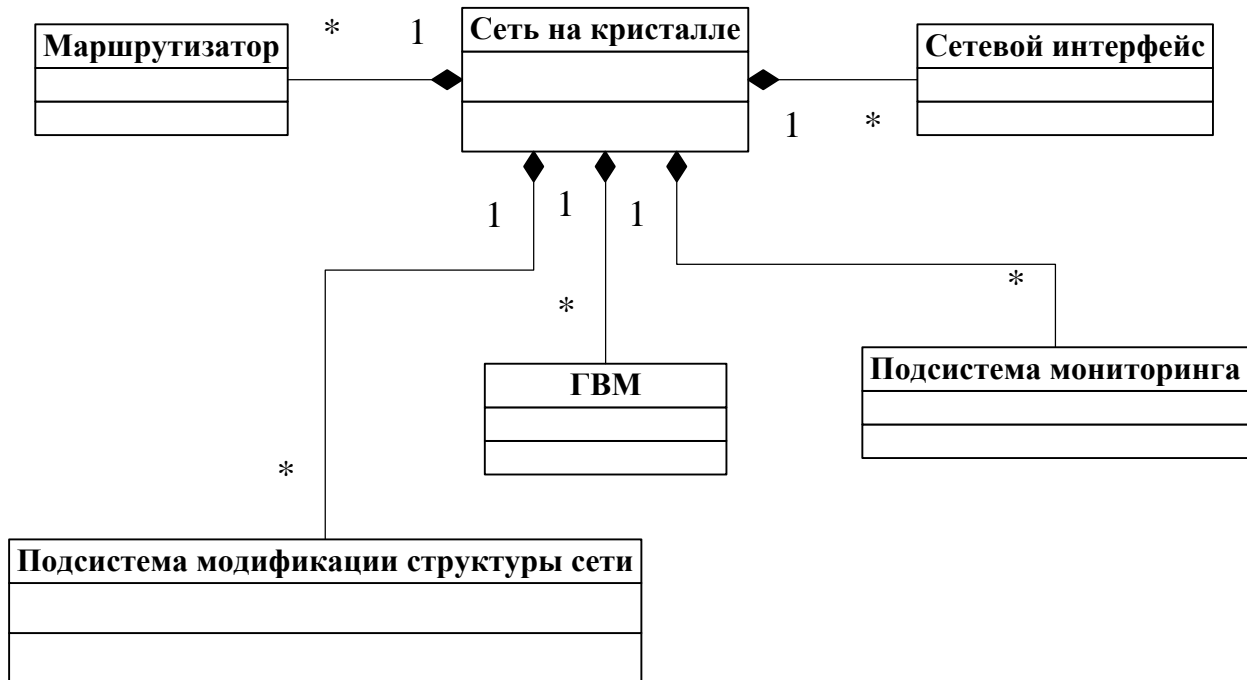


Рис. 4. Описание структуры СтнК при помощи UML-диаграммы классов

Выводы

По результатам проведенных исследований можно сформулировать следующие выводы:

- обосновано применение мульти-агентного подхода к моделированию СтнК. Такое решение позволяет имитировать работу СтнК с реконфигурируемой архитектурой за счет возможности внесения изменений на любом уровне абстракции от преобразования топологии сети на системном уровне до модификации таблиц маршрутизации (средний уровень) и размеров очередей портов ввода-вывода маршрутизаторов на низком уровне;
- предложен метод мульти-агентной имитации реконфигурируемых СтнК, что создает предпосылки для сокращения времени моделирования за счет одновременного функционирования всех агентов в системе;

- использование языка UML 2.0 для описания структуры модели и связей между агентами упрощает освоение разработчиками предложенного метода, поскольку UML является широко распространенным унифицированным языком описания статических и динамических объектно-ориентированных систем, к которым, безусловно, можно отнести и мульти-агентные модели;
- полученные результаты являются гипотезой, поэтому следующим шагом исследований в данном направлении будет создание рабочего макета системы и получение количественных сравнительных характеристик относительно других средств моделирования СтнК.

Список литературы

1. Dally W., Towles B. Route packets, not wires: on-chip interconnection networks // Proceedings of the 38th annual Design Automation Conference (June 2001). – Las Vegas, USA. – P.684-689.
2. Gebali F., Elmiligi H., Watheq El-Kharashi M. Networks-on-Chip: Theory and Practice.– Boca Raton (USA): CRC Press/Taylor and Francis Group LLC, 2009. – 307p.
3. Bjerregaard T., Mahadevan S. A survey of research and practices of Network-on-chip // ACM Computing Surveys. – 2006. –Vol.38, №1. – P.1-51.
4. Brebner G., Levi D. Networking on Chip with Platform FPGAs // Proceedings on Field-Programmable Technology (FPT) IEEE International Conference (December 2003). – P.13-20.

5. *Janarthanan A.* Networks-on-Chip based high performance communication architectures for FPGA's. – University of Cincinnati. Division of Research and Advanced Studies.– Cincinnati, USA, 2008. – 143p.
6. *Vincenzo R., Atienza D.* A Reconfigurable Network-on-Chip Architecture for Optimal Multi Processor SoC Communication // 16th IFIP/IEEE International Conference on Very Large Scale Integration (October 2008). – Rhodes, Greece. – P. 321-326.
7. *Rikard T.* A Network on Chip Simulator: Master of Science Thesis. – Royal Institute of Technology (KTH). Department of Microelectronics and Information Technology. – Kista, Sweden, 2002.– 55p.
8. *Ali M., Welzl M.* Using the NS-2 Network Simulator for Evaluating Network on Chips // Proceedings on IEEE International Conference of Emerging Technologies ICET '06 (November 2006). – P.506-512.
9. *Hang-Sheng W., Xinping Z.* Orion: a power-performance simulator for interconnection networks // Proceedings of the 35th annual ACM/IEEE international symposium on Microarchitecture (November 2002). – Istanbul, Turkey. – P. 294-305.
10. *Genko N., Atienza D., Benini L.* NoC Emulation: A Tool and Design Flow for MPSoC // IEEE Circuits and Systems Magazine. – 2007. – Vol.7, №4. – P. 42-51.
11. *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. – Vol.16, №2. – P.113-129.
12. *Goossens K., Dielissen J.* The Aethereal Network on Chip: Concepts, Architectures and Implementations // IEEE Design and Test of Computers. – 2005. – Vol.22, №5. – P.414-421.
13. *Siguenza-Tortosa D., Nurmi J.* VHDL-based simulation environment for Proteo NoC // Proceedings of the Seventh IEEE International High-Level Design Validation and Test Workshop (October 2002). – P.1-6.
14. *Matos D., Carro L.* The Need for Reconfigurable Routers in Networks-on-Chip // Proceedings of the 5th International Workshop on Reconfigurable Computing: Architectures, Tools and Applications (June 2009). – Karlsruhe, Germany. – P. 275-280.
15. *Борщев А.В.* Практическое агентное моделирование и его место в арсенале аналитика // Экспонента Про. Математика в приложениях. – 2004. – № 3-4. – С. 38-47.

Поступила в редакцию 17.12.2009