

# Свободная от дедлоков маршрутизация в сетях на кристалле с циркулянтными топологиями

Н.М. Мячин<sup>1</sup>, А.Ю. Романов<sup>1</sup>, Э.А. Монахова<sup>2</sup>

<sup>1</sup>Национальный исследовательский университет «Высшая школа экономики», г. Москва

wertual08@gmail.com, a.romanov@hse.ru

<sup>2</sup>Институт вычислительной математики и математической геофизики, СО РАН, г. Новосибирск

emilia@rav.sccc.ru

**Аннотация** – В данной статье рассматривается решение проблемы предотвращения тупиковых ситуаций при применении различных алгоритмов маршрутизации в сетях на кристалле (СтНК) с циркулянтными топологиями. Описываются два возможных способа решения данной проблемы. Предложено эффективное решение, основанное на особенностях циркулянтной топологии, которое позволило добиться значительного улучшения пропускной способности сети по сравнению с получившей широкое распространение в СтНК топологией mesh. Сравнение пиковой пропускной способности сети при высокоуровневом моделировании показало прирост более чем на 60 % для циркулянтных топологий. Также рассмотрен универсальный метод борьбы с дедлоками, применимый для любых топологий и основанный на обходе заблокированных участков сети по ациклической подсети. Выполнена оценка разработанных методов маршрутизации при помощи универсальной высокоуровневой модели СтНК, созданной на основе модифицированного симулятора Noxim.

**Ключевые слова** – сеть на кристалле, СтНК, алгоритм маршрутизации, дедлок, циркулянтная топология, моделирование СтНК.

## I. ВВЕДЕНИЕ

Для проектирования СтНК необходимо определить ее тип (статический [1] или динамический [2]), топологию сети, метод контроля передачи данных [3–5], а также алгоритм маршрутизации. Циркулянтные топологии [6], в силу своих свойств имеющие лучшие характеристики по сравнению с топологиями mesh и torus [7], являются перспективной альтернативой для применения при проектировании сетей на кристалле (СтНК). Использование новых топологий в СтНК требует разработки алгоритмов маршрутизации, которые бы позволяли не только строить маршруты перемещения пакетов, но и гарантировали бы при этом отсутствие тупиковых ситуаций, приводящих к полной остановке передачи данных через сеть.

Существует множество алгоритмов маршрутизации в циркулянтных топологиях, имеющих разную эффективность и сложность реализации. Например, алгоритм парных обменов из [8], позволяющий

рассчитать маршрут пакета в виде количества необходимых переходов пакета по образующим циркулянта при его отправке. Алгоритмы маршрутизации для оптимальных циркулянтов [9–10] проще в реализации, но не подходят для циркулянтов кольцевого типа. Еще более простая реализация у алгоритма почасового обхода из [8], разработанного для применения в циркулянтах кольцевого типа, не гарантирует получения оптимальных маршрутов. Существует также алгоритм маршрутизации в мультипликативных циркулянтах [11], имеющий среднюю сложность реализации, но ограниченный тем, что может применяться только в мультипликативных циркулянтах.

Таким образом, существующие алгоритмы для маршрутизации пакетов в циркулянтных топологиях обладают своими преимуществами и недостатками и могут быть использованы в зависимости от поставленной задачи. Но у них есть общий критичный недостаток, заключающийся в том, что все эти алгоритмы не предоставляют гарантии отсутствия тупиковых ситуаций при передаче данных. Проблема предотвращения тупиковых ситуаций в СтНК хорошо изучена применительно к таким топологиям как mesh и torus, что привело к появлению целого класса алгоритмов, призванных бороться с дедлоками [12]. Поскольку циркулянты являются регулярными топологиями с симметричной структурой, возникновение дедлоков в них происходит из-за наложения путей, предоставляемых существующими алгоритмами маршрутизации.

В этой статье изложен новый алгоритм, названный Ring-Split (RS), решающий проблему возникновения дедлоков в СтНК с циркулянтными топологиями и позволяющий использовать уже существующие алгоритмы маршрутизации.

Дедлок (store-and-forward deadlock) – циклическая зависимость нескольких пакетов, которая приводит к взаимной блокировке их дальнейшего продвижения по сети. Достаточным условием для возникновения дедлоков является существование «узловых циклов» [13], образованных наложением возможных маршрутов. Такие алгоритмы как XY, Odd-Even [14],

SOE [15] и др. для топологии mesh путем использования простых правил гарантируют наличие множества возможных путей, не имеющих наложений, что обеспечивает отсутствие тупиковых ситуаций и значительно улучшает характеристики сети. Циркулянтные топологии содержат большое количество циклов и имеют сложную структуру, что усложняет разработку алгоритмов, обеспечивающих множество путей, удовлетворяющих ограничению на наличие «узловых циклов». Простейший пример «узлового цикла» представлен на рис. 1, где каждый из маршрутов имеет общий переход с двумя другими маршрутами, что приводит к возможности возникновения дедлоков в такой ситуации.

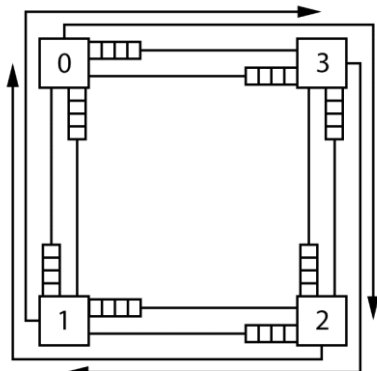


Рис. 1. Узловой цикл

Для оценки эффективности предложенных решений потребовалось провести высокоуровневое моделирование СтнК. Существует большое количество программных симуляторов (например, рассмотренных в [16]), разработанных для решения этой задачи для различных топологий. Но идея применения циркулянтных топологий при проектировании СтнК достаточно нова, и ни один из существующих симуляторов не предоставляет необходимых инструментов для данного исследования. По этой причине была разработана высокоуровневая модель [17], в основу которой лег симулятор Noxim [18]. Результаты моделирования приведены в разделе 6.

Таким образом, цель данной работы - разработка и тестирование алгоритмов предотвращения тупиковых ситуаций в сетях на кристалле с циркулянтными топологиями.

## II. ОПИСАНИЕ РАБОТЫ АЛГОРИТМА RING-SPLIT

Поскольку дедлоки возникают вследствие наложения возможных маршрутов пакетов в «узловых циклах» [13], необходимо отбросить часть возможных маршрутов для того, чтобы эти циклы разорвать. При этом разрыв циклов в графе приводит и к разрыву всех лежащих на нем «узловых циклов».

Пусть имеется неориентированный циркулянт  $C(N; s_1, s_2, \dots, s_k)$ , где  $N$  – количество узлов,  $k$  – размерность графа,  $s_i, 1 \leq i \leq k$ , – образующие графа [19]. Такой граф можно разделить на  $k$  подграфов,

построенных из ребер, соответствующих образующим, и инцидентных им вершин. Каждый подграф состоит из  $m_j \geq 1$  несвязных графов-циклов (циклических графов), где  $m_j = \text{НОД}(N, s_j)$ .

Назовем каждый из подграфов уровнем  $L_j$ . Все ребра, принадлежащие соответствующему уровню, назовем  $l_{ji} \in L_j$ . Графы-циклы, соответствующие каждому уровню, назовем кольцами  $R(j, r)$  где  $j$  – номер уровня, которому принадлежит кольцо,  $r$  – номер соответствующего графа-цикла. Таким образом, получим множество всех возможных циклов для каждого уровня, состоящее из всех принадлежащих ему колец.

Разделим все возможные циклы циркулянта на две группы. Циклы, которые лежат непосредственно на уровнях  $L_j$ , и все остальные циклы циркулянта, образованные изначальной связностью уровней. Введя ограничение на переход между ребрами, принадлежащим разным уровням, можно разорвать все циклы второй группы. Запишем условие, при котором разрешается переход из ребра  $l_{si} \in L_s$  в ребро  $l_{di} \in L_d$ . Переход между уровнями возможен исключительно вниз по их иерархии, то есть при выполнении условия  $s \geq d$ . Таким образом, получена система правил, при которой возникновение «узловых циклов» возможно только в рамках имеющихся уровней  $L_i$ . То есть задача сводится к организации свободной от тупиковых ситуаций маршрутизации на отдельных уровнях, каждый из которых состоит из  $m_j$  несвязных графов – циклов.

## III. ПРИМЕНЕНИЕ МЕТОДА RING-SPLIT К АЛГОРИТМУ МАРШРУТИЗАЦИИ

Каждый маршрут из  $A$  в  $B$  циркулянта  $C(N; s_1, s_2, \dots, s_k)$  может быть представлен как  $k$  – мерный вектор  $\{x_1, x_2, \dots, x_k\}$ , где  $x_i \in \mathbb{Z}$  – количество перемещений по  $i$  – ой образующей по часовой стрелке ( $x_i > 0$ ) или против часовой стрелки ( $x_i < 0$ ), необходимое для достижения узла  $B$ , начиная от узла  $A$ . Таким образом, необходимое для реализации алгоритма RS условие заключается в том, что перемещение по  $x_i, i = \overline{2, k}$ , должно осуществляться раньше, чем по  $x_{i-1}$ . Как следствие, пакеты будут проходить все образующие маршрута в строго определенном порядке.

На рис. 2 показано разделение на уровни и возможные перемещения пакетов для циркулянта  $C(10; 1, 4)$ . Пакет, следующий из узла  $A$  в узел  $B$  попадает в одно из колец  $R(2, 1)$ ,  $R(2, 2)$  и совершает, согласно приоритету, все необходимые  $x_2$  перемещений, после чего переходит в кольцо  $R(1, 1)$ , где совершает оставшиеся  $x_1$  перемещений и достигает узла  $B$ . При такой схеме перемещение пакета возможно только из уровня  $L_2$  в уровень  $L_2$  или  $L_1$ , что удовлетворяет условию  $l_{si} \in L_s, l_{di} \in L_d, s \geq d$ .

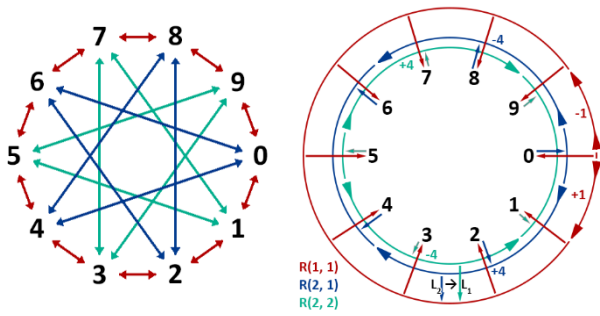


Рис. 2. Пример структуры уровней циркулянта  $C(10; 1, 4)$

#### IV. ОРГАНИЗАЦИЯ МАРШРУТИЗАЦИИ В КОЛЬЦАХ

Предлагаемые ограничения на перемещения пакетов являются универсальными и не зависят от выбранного типа маршрутизации. При этом тип маршрутизации сильно влияет на маршрутизацию в рамках каждого отдельного кольца  $R(j, r)$ . Рассмотрим возможные типы маршрутизации, которые можно применять в комплексе с представленной системой ограничений для устранения дедлоков.

##### A. Store-and-Forward маршрутизация

Store-and-forward [20] – наиболее простой для реализации тип маршрутизации. Для того чтобы гарантировать отсутствие тупиковых ситуаций в рамках каждого кольца необходимо в первую очередь производить передачу тех пакетов, которые лежат в буферах, принадлежащих соответствующему кольцу. Таким образом, для уровня  $L_1$  будет гарантировано движение всех пакетов в обоих направлениях, а для уровня  $L_2$  будет гарантировано движение всех пакетов, кроме тех, которые стоят в очереди на переход на уровень  $L_1$ , или ими заблокированных (рис. 3). Это можно реализовать путем фиксированного порядка арбитража пакетов в буферах. Порядок принятия решений для пакетов соответствует порядку уровней, к которым принадлежат буферы соответствующих пакетов. Порядок в рамках одного уровня может быть произвольным. Решение о передаче пакетов, принадлежащих локальным буферам, принимается в последнюю очередь.

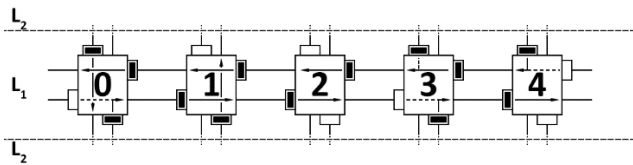


Рис. 3. Порядок маршрутизации пакетов

##### B. Wormhole маршрутизация

Реализация маршрутизации типа wormhole [21] имеет значительные ограничения в силу того, что хвосты одних пакетов могут препятствовать непрерывному движению других. Для решения этой проблемы предлагается разделить каждое кольцо  $R(j, r)$  циркулянта на два виртуальных канала [22, 23]. Поскольку в процессе перемещения по кольцу пакет делает не более  $\lfloor N/2 \rfloor$  переходов, разделив пакеты,

приходящие в разные половины кольца по разным виртуальным каналам, можно гарантировать отсутствие блокировок и равномерную нагрузку на каждый из каналов. Таким образом, в соответствии с рис. 4, пакеты, попадающие в кольцо из первой половины узлов цикла, направляются в канал VC1, в то время как пакеты второй половины будут перемещаться по каналу VC2. При этом пакет в любой момент может выйти из кольца (соблюдая ограничение на перемещение между уровнями), но не может сменить виртуальный канал в рамках одного кольца. Если формулировать в терминологии «узловых циклов», то циклы будут разорваны в нулевом узле для пакетов, движущихся по часовой стрелке, и узле ему противоположном для пакетов, движущихся в другую сторону. Гарантированный разрыв «узлового цикла» объясняется тем, что ни один из пакетов не пересечет границу, являющуюся следствием ограничения на смену виртуального канала и максимального количества перемещений по кольцу, равного  $\lfloor N/2 \rfloor$ . Например, исходя из рис. 4, пакеты, попавшие в узлы 0-4 и движущиеся по часовой стрелке, никогда не будут переходить из 9-го узла в 0-ой, так как двигаясь по часовой стрелке, в 9-й узел можно попасть только из 4-го, не превысив при этом ограничения на количество перемещений.

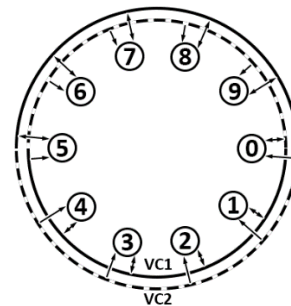
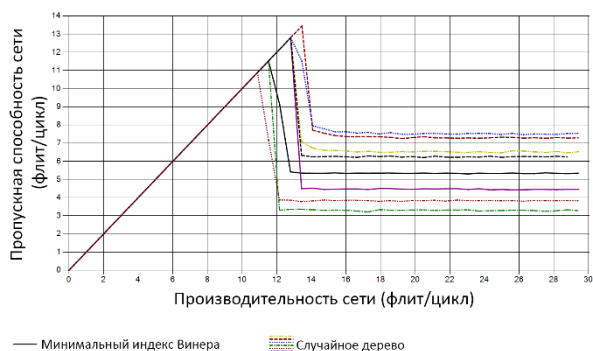


Рис. 4. Разделение кольца на виртуальные каналы

#### V. ПРИМЕНЕНИЕ АЦИКЛИЧНОЙ ПОДСЕТИ ДЛЯ РАЗРЕШЕНИЯ ВОЗНИКАЮЩИХ ТУПИКОВЫХ СИТУАЦИЙ

Более универсальное решение, позволяющее организовывать свободную от дедлоков store-and-forward маршрутизацию в любой топологии, заключается в организации ациклической подсети. Основные идеи данного подхода описаны в [24]. Суть подхода состоит в том, что любой пакет, попавший в тупиковую ситуацию, может быть направлен в подсеть для обхода заблокированного участка. В качестве ациклической подсети проще всего использовать покрывающее дерево с корнем в нулевой вершине. Существуют два варианта организации маршрутизации в подсети. Однократное перенаправление пакетов в подсеть гарантирует отсутствие ливлоков, но дает очень низкую пропускную способность. Перенаправление пакетов в подсеть с возможностью выхода из нее может приводить к циркуляции пакета по сети, однако данный метод снижает нагрузку на подсеть и дает хорошие показатели пропускной способности всей сети. Назовем этот метод Acyclic Subnetwork

Method (ASM). При этом в результате экспериментов было показано, что характеристики сети сильно зависят от того, какое покрывающее дерево выбрать. Так, например, на рис. 5 приведены результаты моделирования в СтнК пропускной способности сети с mesh топологией 8x8 и стандартным алгоритмом XY с использованием различных деревьев в качестве подсети.

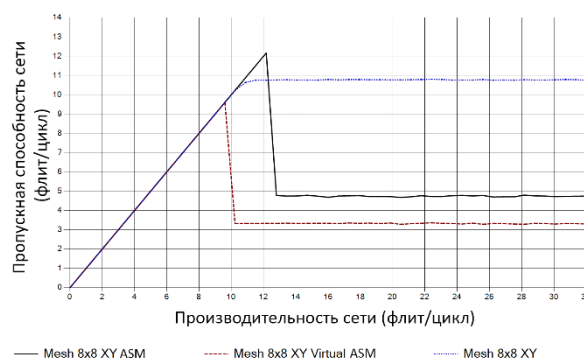


**Рис. 5. Сравнение различных деревьев в качестве подсети для обхода дедлоков**

Выбор в качестве подсети дерева с минимальной стоимостью маршрутизации (минимальным индексом Винера [25]) дает хорошие показатели пропускной способности, но, как показано на рис. 5, могут существовать варианты деревьев, которые дают показатели лучше. Таким образом, остается открытым вопрос поиска оптимального покрывающего дерева для организации подсети. Кроме того, основанный на ациклической подсети подход [24] не гарантирует оптимальности маршрутов пакетов, но при этом не приводит к ливлокам [26]. Необходимо также учитывать дополнительные затраты на реализацию нестандартных маршрутизаторов и связей покрывающего дерева.

Подсеть также может быть организована с использованием виртуальных каналов, что уменьшит аппаратные затраты. Поскольку основной вклад в пропускную способность вносит базовая топология, а подсеть лишь разрешает тупиковые ситуации, перенос подсети из физических каналов в виртуальные не сильно сказывается на общей пиковой пропускной способности.

Результаты моделирования предлагаемого универсального подхода с вариантами ациклической подсети, организованной с реальными и виртуальными каналами приведены на рис. 6. СтнК с виртуальной подсетью имеет меньшие показатели пропускной способности даже чем СтнК с топологией mesh и стандартным алгоритмом XY. Это говорит о том, что применение такого способа обхода блокировок пакетов имеет смысл в топологиях, для которых еще не было найдено решения, непосредственно учитывающего их особенности. При этом применение другой топологии подсети или выбор оптимального покрывающего дерева может значительно улучшить показатели пиковой пропускной способности сети.



**Рис. 6. Сравнение пропускной способности виртуальной и физической подсети при использовании ASM**

Важной особенностью рассматриваемого подхода является сильное падение пропускной способности после достижения некоторого пикового значения. Происходит это по причине того, что в момент после пика нагрузка на сеть, создаваемая возникающими дедлоками, превышает пропускную способность подсети. В результате общая пропускная способность снижается практически до уровня пропускной способности самой подсети. Таким образом, необходимо контролировать общую нагрузку на сеть для максимально эффективного ее использования. Также этот метод не допускает возникновения хвостов пакетов, блокирующих передачу данных, и, следовательно, применим только для store-and-forward маршрутизации.

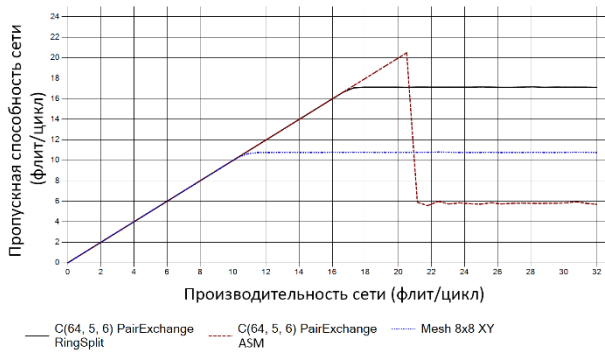
## VI. МОДЕЛИРОВАНИЕ АЛГОРИТМОВ МАРШРУТИЗАЦИИ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМА RING-SPLIT

Для анализа эффективности разработанного алгоритма RS проведено сравнение пропускной способности сети, которой можно добиться при его использовании, с той, которую можно получить с помощью алгоритма XY в сетях с mesh топологией.

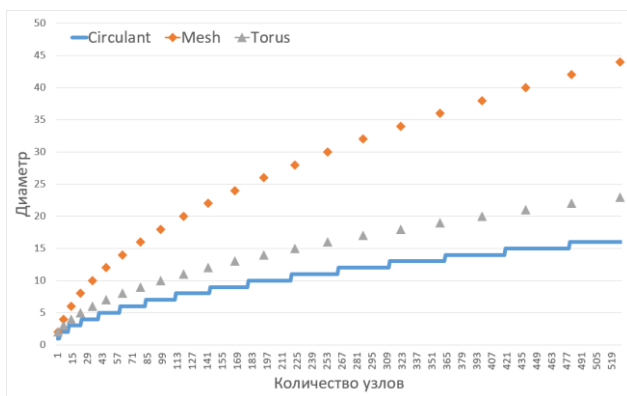
В качестве алгоритма маршрутизации в циркулянтах был выбран алгоритм парных обменов Pair Exchange [10].

При размере буферов в 2 пакета и store-and-forward маршрутизации, оптимальный циркулянт  $C(64; 5, 6)$  показал прирост пропускной способности более 60 % по сравнению с mesh топологией (Рис. 7). Пиковая пропускная способность при использовании физической реализации ациклической подсети оказалась практически вдвое больше.

Поскольку у циркулянтных топологий с увеличением количества узлов диаметр увеличивается медленнее, чем у mesh топологий, то для большего количества узлов прирост пропускной способности будет еще заметней. Так рис. 8 демонстрирует зависимость диаметра графов различных топологий от количества узлов.



**Рис. 7. График пропускной способности для различных реализаций сети из 64 узлов при использовании RS, ASM, XY**



**Рис. 8. Зависимость диаметра графов различных топологий от количества узлов**

В качестве циркулянтов использовались оптимальные графы семейства  $C(N; D, D + 1)$ , свойства которых рассмотрены в [10,19], так как для них применим рассматриваемый алгоритм парных обменов. С увеличением количества узлов все больше проявляется значительное преимущество циркулянтных топологий над mesh топологиями. Топология 2D-torus имеет близкие показатели диаметра к циркулянтам, поскольку является более связным графом, чем mesh. Как известно, некоторые варианты 2D-torus являются частным случаем неоптимальных циркулянтов и, следовательно, к ним тоже можно применять разработанный алгоритм RS.

## VII. ВЫВОДЫ

Предложенный алгоритм RS (Ring-Split) имеет немного меньшую пиковую пропускную способность по сравнению с методом, использующим физическую подсеть в виде покрывающего дерева с минимальным индексом Винера, но его реализация значительно проще и не требует нестандартных маршрутизаторов. К тому же, алгоритм Ring-Split позволяет при необходимости использовать маршрутизацию типа wormhole. Разработанный новый подход к предотвращению тупиковых ситуаций открывает возможность применения циркулянтов как топологической основы СтНК с использованием любого

из имеющихся алгоритмов маршрутизации. В результате высокоуровневого моделирования было показано, что достигается прирост до 60 % производительности сети с использованием нового подхода для маршрутизации в циркулянтах по сравнению с mesh топологией при равномерном распределении нагрузки на сеть.

Второй описанный метод маршрутизации с обходом дедлоков за счет организации ациклической подсети является универсальным, но, вследствие сложности реализации, хуже алгоритмов, учитывающих особенности топологии, и, следовательно, актуален только для топологий сетей, для которых не найдено формализованного способа обойти тупиковые ситуации.

## ПОДДЕРЖКА

Исследование Мячина Н.М. и Романова А.Ю. осуществлено в рамках Программы фундаментальных исследований НИУ ВШЭ. Исследование Монаховой Э.А. выполнено в рамках бюджетного проекта ИВМиМГ СО РАН № 0251-2021-0005.

## ЛИТЕРАТУРА

- [1] Moraes F. et al. A Low Area Overhead Packet-switched Network on Chip: Architecture and Prototyping // IFIP VLSI-SoC 2003. 2003. № II. P. 318–323.
- [2] Chen S.-J. et al. Reconfigurable Networks-on-Chip // Reconfigurable Networks-on-Chip. New York, NY: Springer New York, 2012. 203 p. doi: 10.1007/978-1-4419-9341-0.
- [3] Pullini A. et al. Fault Tolerance Overhead in Network-on-Chip Flow Control Schemes // 2005 18th Symposium on Integrated Circuits and Systems Design. IEEE, 2005. P. 224–229. doi: 10.1109/SBCCI.2005.4286861.
- [4] Al-Tawil K.M., Abd-El-Barr M., Ashraf F. A survey and comparison of wormhole routing techniques in a mesh networks // IEEE Netw. 1997. V. 11. № 2. P. 38–45. doi: 10.1109/65.580917.
- [5] Agarwal A., Iskander C., Shankar R. Survey of network on chip (NoC) architectures & contributions // J. Eng. Comput. Archit. 2009. V. 3. № 1. P. 4133–4144.
- [6] Beivide R. et al. Chordal Topologies for Interconnection Networks // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2003. V. 2858. P. 385–392. doi: 10.1007/978-3-540-39707-6\_33.
- [7] Mirza-Aghatabar M. et al. An Empirical Investigation of Mesh and Torus NoC Topologies Under Different Routing Algorithms and Traffic Models // 10th Euromicro Conference on Digital System Design Architectures, Methods and Tools (DSD 2007). IEEE, 2007. P. 19–26. doi: 10.1109/DSD.2007.4341445.
- [8] Romanov A.Yu. Development of routing algorithms in networks-on-chip based on ring circulant topologies // Heliyon. Elsevier, 2019. V. 5. № 4. P. e01516. doi: 10.1016/J.HELIYON.2019.E01516.
- [9] Romanov A.Yu. et al. Development of routing algorithms in networks-on-chip based on two-dimensional optimal circulant topologies // Heliyon. Elsevier, 2020. V. 6. № 1. P. e03183. doi: 10.1016/j.heliyon.2020.e03183.
- [10] Monakhova E.A., Romanov A.Yu., Lezhnev E.V. Shortest Path Search Algorithm in Optimal Two-Dimensional Circulant Networks: Implementation for Networks-on-Chip

- // IEEE Access. 2020. V. 8. P. 215010–215019. doi: 10.1109/ACCESS.2020.3040323.
- [11] Щеголева М.А., Романов А.Ю. Разработка алгоритма маршрутизации в сетях на кристалле с топологией мультипликативный циркулянт // Проблемы разработки перспективных микро- и наноэлектронных систем(МЭС).2018. Вып.3.С. 119–125. doi:10.31114/2078-7707-2018-3-119-125
- [12] Palesi M. et al. A methodology for design of application specific deadlock-free routing algorithms for NoC systems // Proceedings of the 4th international conference on Hardware/software codesign and system synthesis - CODES+ISSS '06. New York, New York, USA, New York, USA: ACM Press, 2006. P. 142–147. doi: 10.1145/1176254.1176289.
- [13] Jerger N.E., Peh L.S. On-Chip networks // Synth. Lect. Comput. Archit. 2009. doi: 10.2200/S00209ED1V01Y200907CAC008.
- [14] Ge-Ming Chiu. The odd-even turn model for adaptive routing // IEEE Trans. Parallel Distrib. Syst. 2000. V. 11. № 7. P. 729–738. doi: 10.1109/71.877831.
- [15] Su Hu, Xiaola Lin. A Symmetric Odd-Even Routing Model in Network-on-Chip // 2012 IEEE/ACIS 11th International Conference on Computer and Information Science. IEEE, 2012. P. 457–462. doi: 10.1109/ICIS.2012.20.
- [16] Benmessaoud Gabis A., Koudil M. NoC routing protocols – objective-based classification // J. Syst. Archit. Elsevier B.V., 2016. V. 66–67. P. 14–32. doi: 10.1016/j.sysarc.2016.04.011.
- [17] Myachin N.M., Romanov A.Yu. NewXim. URL: <https://github.com/Wertual08/newxim> (access date: 25.08.2021).
- [18] Catania V. et al. Noxim: An open, extensible and cycle-accurate network on chip simulator // 2015 IEEE 26th International Conference on Application-specific Systems, Architectures and Processors (ASAP). IEEE, 2015. V. 2015-Septe. P. 162–163. doi: 10.1109/ASAP.2015.7245728.
- [19] Monakhova E.A. A survey on undirected circulant graphs // Discret. Math. Algorithms Appl. 2012. V. 4. № 1. P. 17–47. doi: 10.1142/S1793830912500024.
- [20] Fratta L., Gerla M., Kleinrock L. The flow deviation method: An approach to store-and-forward communication network design // Networks. 1973. V. 3. № 2. P. 97–133. doi: 10.1002/net.3230030202.
- [21] Guz Z. et al. Network Delays and Link Capacities in Application-Specific Wormhole NoCs // VLSI Des. 2007. P. 1–15. doi: 10.1155/2007/90941.
- [22] Dally W.J., Seitz C.L. Deadlock-Free Message Routing in Multiprocessor Interconnection Networks // IEEE Trans. Comput. 1987. V. C–36. № 5. P. 547–553. doi: 10.1109/TC.1987.1676939.
- [23] Dally W.J., Aoki H. Deadlock-free adaptive routing in multicompiler networks using virtual channels // IEEE Trans. Parallel Distrib. Syst. 1993. V. 4. № 4. P. 466–475. doi: 10.1109/71.219761.
- [24] Монахова Э.А. Предотвращение тупиковых состояний при передаче информации в вычислительных системах с программируемой структурой // Электронное моделирование. 1986. Т. 8. № 5. С. 17–21.
- [25] Masone A. et al. The Minimum Routing Cost Tree Problem // Soft Comput. 2019. V. 23. № 9. P. 2947–2957. doi: 10.1007/s00500-018-3557-3.
- [26] Janfaza V., Baharlouei E. A new fault-tolerant deadlock-free fully adaptive routing in NOC // 2017 IEEE East-West Design & Test Symposium (EWDTS). IEEE, 2017. P. 1–6. doi: 10.1109/EWDTS.2017.8110139.

## Deadlock-Free Routing in Networks-on-Chip with Circulant Topologies

N.M. Myachin<sup>1</sup>, A.Yu. Romanov<sup>1</sup>, E.A. Monakhova<sup>2</sup>

<sup>1</sup>HSE University, Moscow

wertual08@gmail.com, a.romanov@hse.ru

<sup>2</sup>ICMMG SB RAS, Novosibirsk

emilia@rav.sccc.ru

**Abstract** – This paper is considering the solution of the deadlock problem when applying different routing algorithms to networks-on-chip (NoCs) with circulant topology. The paper describes two different solutions for this problem. There is an effective method based on special aspects of circulant topologies which provides significant increase of peak throughput compared with mesh topology widely applied in NoCs. The comparison of peak throughput based on high-level simulation results showed more than 60 % increase for circulant topologies. In addition, there is a review of universal deadlock avoidance method based on bypassing blocked network segments through acyclic sub-network. The presented methods were evaluated using a high-level universal simulator for NoCs developed on the base of Noxim simulator.

**Keywords** – network-on-chip, NoC, routing algorithm, deadlock, circulant topology, NoC modeling.

### REFERENCES

- [1] Moraes F. et al. A Low Area Overhead Packet-switched Network on Chip: Architecture and Prototyping // IFIP VLSI-SoC 2003. 2003. № II. P. 318–323.
- [2] Chen S.-J. et al. Reconfigurable Networks-on-Chip // Reconfigurable Networks-on-Chip. New York, NY: Springer New York, 2012. 203 p. doi: 10.1007/978-1-4419-9341-0.
- [3] Pullini A. et al. Fault Tolerance Overhead in Network-on-Chip Flow Control Schemes // 2005 18th Symposium on



- Integrated Circuits and Systems Design. IEEE, 2005. P. 224–229. doi: 10.1109/SBCCI.2005.4286861.
- [4] Al-Tawil K.M., Abd-El-Barr M., Ashraf F. A survey and comparison of wormhole routing techniques in a mesh networks // IEEE Netw. 1997. V. 11. № 2. P. 38–45. doi: 10.1109/65.580917.
- [5] Agarwal A., Iskander C., Shankar R. Survey of network on chip (NoC) architectures & contributions // J. Eng. Comput. Archit. 2009. V. 3. № 1. P. 4133–4144.
- [6] Beivide R. et al. Chordal Topologies for Interconnection Networks // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2003. V. 2858. P. 385–392. doi: 10.1007/978-3-540-39707-6\_33.
- [7] Mirza-Aghatabar M. et al. An Empirical Investigation of Mesh and Torus NoC Topologies Under Different Routing Algorithms and Traffic Models // 10th Euromicro Conference on Digital System Design Architectures, Methods and Tools (DSD 2007). IEEE, 2007. P. 19–26. doi: 10.1109/DSD.2007.4341445.
- [8] Romanov A.Yu. Development of routing algorithms in networks-on-chip based on ring circulant topologies // Heliyon. Elsevier, 2019. V. 5. № 4. P. e01516. doi: 10.1016/J.HELIYON.2019.E01516.
- [9] Romanov A.Yu. et al. Development of routing algorithms in networks-on-chip based on two-dimensional optimal circulant topologies // Heliyon. Elsevier, 2020. V. 6. № 1. P. e03183. doi: 10.1016/j.heliyon.2020.e03183.
- [10] Monakhova E.A., Romanov A.Yu., Lezhnev E.V. Shortest Path Search Algorithm in Optimal Two-Dimensional Circulant Networks: Implementation for Networks-on-Chip // IEEE Access. 2020. V. 8. P. 215010–215019. doi: 10.1109/ACCESS.2020.3040323.
- [11] Shchegoleva M.A., Romanov A.Yu. Development of Routing Algorithms in Networks on Chip with a Multiplicative Circulant Topology // Problems of Perspective Micro- and Nanoelectronic Systems Development. 2018. Issue 3. P. 119–125. doi: 10.31114/2078-7707-2018-3-119-125. (in Russian)
- [12] Palesi M. et al. A methodology for design of application specific deadlock-free routing algorithms for NoC systems // Proceedings of the 4th international conference on Hardware/software codesign and system synthesis - CODES+ISSS '06. New York, New York, USA, New York, USA: ACM Press, 2006. P. 142–147. doi: 10.1145/1176254.1176289.
- [13] Jerger N.E., Peh L.S. On-Chip networks // Synth. Lect. Comput. Archit. 2009. doi: 10.2200/S00209ED1V01Y200907CAC008.
- [14] Ge-Ming Chiu. The odd-even turn model for adaptive routing // IEEE Trans. Parallel Distrib. Syst. 2000. V. 11. № 7. P. 729–738. doi: 10.1109/71.877831.
- [15] Su Hu, Xiaola Lin. A Symmetric Odd-Even Routing Model in Network-on-Chip // 2012 IEEE/ACIS 11th International Conference on Computer and Information Science. IEEE, 2012. P. 457–462. doi: 10.1109/ICIS.2012.20.
- [16] Benmessaoud Gabis A., Koudil M. NoC routing protocols – objective-based classification // J. Syst. Archit. Elsevier B.V., 2016. V. 66–67. P. 14–32. doi: 10.1016/j.sysarc.2016.04.011.
- [17] Myachin N.M., Romanov A.Yu. NewXim. URL: <https://github.com/Wertual08/newxim> (access date: 25.08.2021).
- [18] Catania V. et al. Noxim: An open, extensible and cycle-accurate network on chip simulator // 2015 IEEE 26th International Conference on Application-specific Systems, Architectures and Processors (ASAP). IEEE, 2015. V. 2015-Sept. P. 162–163. doi: 10.1109/ASAP.2015.7245728.
- [19] Monakhova E.A. A survey on undirected circulant graphs // Discret. Math. Algorithms Appl. 2012. V. 4. № 1. P. 17–47. doi: 10.1142/S1793830912500024.
- [20] Fratta L., Gerla M., Kleinrock L. The flow deviation method: An approach to store-and-forward communication network design // Networks. 1973. V. 3. № 2. P. 97–133. doi: 10.1002/net.3230030202.
- [21] Guz Z. et al. Network Delays and Link Capacities in Application-Specific Wormhole NoCs // VLSI Des. 2007. P. 1–15. doi: 10.1155/2007/90941.
- [22] Dally W.J., Seitz C.L. Deadlock-Free Message Routing in Multiprocessor Interconnection Networks // IEEE Trans. Comput. 1987. V. C–36. № 5. P. 547–553. doi: 10.1109/TC.1987.1676939.
- [23] Dally W.J., Aoki H. Deadlock-free adaptive routing in multicomputer networks using virtual channels // IEEE Trans. Parallel Distrib. Syst. 1993. V. 4. № 4. P. 466–475. doi: 10.1109/71.219761.
- [24] Monakhova E.A. Deadlock avoidance during data transmission in computer systems with a programmable structure // Elektron. Model. 1986. V. 8. № 5. P. 17–21 (in Russian).
- [25] Masone A. et al. The Minimum Routing Cost Tree Problem // Soft Comput. 2019. V. 23. № 9. P. 2947–2957. doi: 10.1007/s00500-018-3557-3.
- [26] Janfaza V., Baharlouei E. A new fault-tolerant deadlock-free fully adaptive routing in NOC // 2017 IEEE East-West Design & Test Symposium (EWDTS). IEEE, 2017. P. 1–6. doi: 10.1109/EWDTS.2017.8110139.