

リアルタイムオンチップネットワーク向け 先読みアービトレーション機構付ルータの設計と実装

笹川 雄二郎^{†1} 向後 卓磨^{†2} 山崎 信行^{†2}

Chip Multiprocessor (CMP) におけるコア数の増加に伴い、コア間の相互接続方式として Network-on-Chip (NoC) が注目を集めてきており、リアルタイムシステムにおいてもその有用性が期待される。NoC はバスと比較してバンド幅が大きい一方で、ルータ遅延を主要因とする大きな転送遅延を持つ。転送遅延を抑えるために、様々な低遅延ルータが提案されているが、現在提案されているルータの低遅延化手法はリアルタイム処理を考慮しておらず、リアルタイムシステム向けルータに適用した場合、優先度逆転問題が生じてしまう。本研究では、優先度逆転問題を生じさせずに、低遅延化を行うリアルタイムシステム向け先読みアービトレーション機構付ルータの設計および実装を行った。先読みアービトレーション機構付ルータと通常のルータの最高優先度パケットの転送遅延を比較した結果、7.6% の信号線の増加と 2.5% のルータ面積の増加で、最高優先度パケットの転送遅延を平均 8.1% 削減することができた。

Design and Implementation of a On-Chip Router with Pre-Arbitration Mechanism for Real-Time Systems

YUJIRO SASAGAWA,^{†1} TAKUMA KOGO^{†2}
and NOBUYUKI YAMASAKI^{†2}

Since the number of cores on a multi-processor is recently increasing, Network-on-Chips (NoC) is anticipated for method of interconnections between cores on a multi-processor and it is expected to come to support real-time systems. Although NoCs have higher bandwidth than buses, they have high transfer latency mainly depending on router latency. Although a number of low latency router is proposed to reduce transfer latency, they can cause priority inversion. In this paper, we design and implement a pre-arbitration router for real-time systems. The results of comparison between basic router and pre-arbitration router shows that the wire and area increases by 7.6% and 2.5% respectively, however the transfer latency of the highest priority packet is reduced by 8.1%.

1. はじめに

近年の半導体技術の飛躍的な進歩により、半導体の集積度は大きく向上し、単一チップ上に多数のコアを載せることが可能となっている。コア数が増加するに従って、バスではバンド幅が不足して来ており、メニーコアアーキテクチャでは、バスに代わる通信方式として Network-on-Chip (NoC) ¹⁾²⁾ が用いられている。

NoC ではルータを通じたパケット転送により通信を行う。ルータはパケット転送の際に、ヘッダ情報を基にルーティング計算やアービトレーションなどの処理を行うため、パケットがルータを経由する度に大きな転送遅延が生じる。チップ上のリンク遅延は非常に小さいため、ルータ遅延が占めることになる。そのため、NoC の転送遅延を削減するため、現在までに様々なルータの低遅延化手法が提案されている³⁾⁴⁾。NoC はリアルタイムシステムにおいても有効であるが、現在提案されている低遅延化手法ではリアルタイム性が考慮されていない。従って、現在提案されている低遅延化手法をリアルタイムシステム向けルータに適用した場合、優先度逆転問題が生じるため、リアルタイムシステムには不向きである。

そこで、本研究では、優先度逆転問題が生じない低遅延化手法を提案する。そして、その手法を用いたリアルタイムシステム向けルータの設計および実装を行い、その評価を行う。

本論文の構成は以下の通りである。2. 章では本研究でベースとするリアルタイムシステム向けルータを説明し、3. 章では現在提案されている低遅延ルータについて述べる。4. 章では本研究で提案する先読みアービトレーション機構付ルータについて述べる。5. 章で先読みアービトレーション機構付ルータを評価し、6. 章で本論文をまとめる。

2. リアルタイムシステム向けルータ

本章では、本研究でベースとするリアルタイムシステム向けルータについて述べる。以降ではこのルータをベーシックルータと呼ぶ。ネットワークトポロジは一般的な 2 次元メッシュを想定している。

^{†1} 慶應義塾大学理工学部情報工学科

Department of Information and Computer Science, Faculty of Science and Technology, Keio University

^{†2} 慶應義塾大学大学院理工学研究科開放環境科学専攻

〒 223-8522 横浜市港北区日吉 3-14-1

Department of Computer Science, Graduate School of Science and Technology, Keio University

3-14-1 Hiyoshi, Kouhoku-ku, Yokohama, Kanagawa 223-8522 Japan

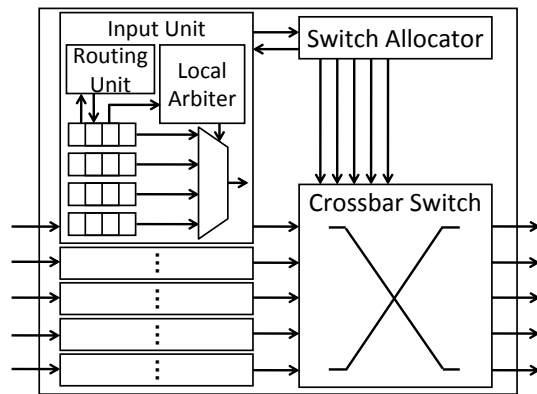


図 1 リアルタイムシステム向けルータのアーキテクチャ
Fig. 1 Basic router architecture for real-time systems

2.1 フロー制御方式

NoC のフロー制御方式として、パケットサイズよりも小さなバッファを持つ Wormhole 方式と、パケットサイズ分のバッファを持つ Virtual Cut-Through 方式がある。Wormhole 方式ではフリット単位でフロー制御を行い、Virtual Cut-Through 方式ではパケット単位でフロー制御を行う。

一般的な NoC ではバッファ数を抑えるために Wormhole 方式が採用されるが、Wormhole 方式によるフリット単位の優先度制御を行うとスタベーションが頻発してしまう。そのため、リアルタイムシステム向けルータでは、パケット単位で優先度制御を行う Virtual Cut-Through 方式を採用する。

2.2 ルーティングアルゴリズム

NoC のルーティングアルゴリズムとして、1つのデスティネーションに対して経路が一意に決定される固定型ルーティングと、ネットワークの状況に応じて動的に経路を決定する適応型ルーティングの2種類に分類することができる⁹⁾。適応型ルーティングはパケットの到着順序がばらつくため、受信ノード側で大きなバッファが必要となる。そのため、本研究では最も一般的な固定型ルーティングである次元順ルーティングを想定する。

2.3 ルータアーキテクチャ

図 1 にリアルタイムシステム向けルータのアーキテクチャを示す。ルータには 5 つの入

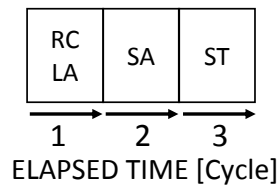


図 2 リアルタイムシステム向けルータのパイプライン
Fig. 2 Basic router pipeline for real-time systems

力ポートと出力ポートがあり、その内、4つが隣接ルータ、1つがコアへと接続されている。Input Unit では受信したパケットをバッファに格納し、Routing Unit による出力ポートの計算と Local Arbiter による入力仮想チャネルのアービトレーションを行う。Switch Allocator では Crossbar Switch のアービトレーションを行い、Crossbar Switch では Switch Allocator の結果に従って、パケットを各出力ポートへ通過させる。

パケットの転送処理は以下のステップに分けられる。

- **Routing Computation (RC) / Local Arbitration (LA)**
 - **RC:** ヘッダ情報に含まれるデスティネーションアドレスから出力ポートを計算する。
 - **LA:** 優先度に基づいて、Switch Allocator にリクエストする入力仮想チャネルをアービトレーションする。
- **Switch Allocator (SA):** 優先度に基づいて、パケットに Crossbar Switch を通過させる Input Unit をアービトレーションする。
- **Switch Traversal (ST):** Switch Allocator が選択した Input Unit からのパケットを出力ポートへ転送する。

従って、リアルタイムシステム向けルータのパイプライン構成は図 2 のようになる。

3. 関連研究

3.1 Prediction Router

Prediction Router⁶⁾ は、次に到着するパケットの経路を予測し、パケットが到着する前に SA ステージを完了させることで、ルータ遅延を削減する。予測が成功した場合、ルータ遅延は ST ステージのみとなる。予測が外れた場合、通常のパイプラインステージを通り、通常のルータ遅延となる。

予測時はパケットの到着が確定していないため、SA ステージでのアービトレーションは、

実際に到着しているパケットが優先される。そのため、Prediction Router の機構をリアルタイムシステム向けルータに適用した場合、予測時におけるアービトレーションでは優先度制御を行うことができず、優先度逆転問題が生じる。

3.2 Express Virtual Channel

Express Virtual Channel (EVC) ⁷⁾ は直線方向に 2 ホップ以上離れたルータとの間のルータに直進専用の仮想バイパスチャネルを設け、仮想的にバイパス経路を構成する。仮想バイパスチャネルを通過するパケットは ST ステージのみのルータ遅延で転送されるが、仮想バイパスチャネルを通過しないパケットは通常のパイプラインステージを通り、通常のルータ遅延となる。

EVC では、仮想バイパスチャネルを通過するパケットは確実に転送される必要があるため、優先して転送されなければならない。そのため、EVC の機構をリアルタイムシステム向けルータに適用した場合、仮想バイパスチャネルを通過するパケットがあるときのアービトレーションでは優先度制御を行うことができず、優先度逆転問題が生じる。

4. 先読みアービトレーション機構付ルータアーキテクチャ

4.1 設計方針

先読みアービトレーション機構付ルータでは、パケットの転送よりも 1 サイクル先行して優先度情報を転送することで、優先度逆転問題を生じさせずにルータ遅延を削減する。

リアルタイムシステム向けルータの Local Arbitration に必要な情報は優先度情報である。先読みアービトレーションを実現するには、ヘッダフリットの転送よりも 1 サイクル先行して、優先度情報を次ルータに転送する必要があり、そのためには、ヘッダフリットが ST ステージを通過しているときに、優先度情報とルーティング情報を次ルータに転送する必要がある。そこで、優先度情報を転送する方向を 1 つに限定することで、Crossbar Switch を通過せずに優先度情報を転送する。これにより、優先度情報は ST ステージを飛ばして転送され、1 サイクル先行することが可能となる。

次元順ルーティングではパケットは直進する確率が高いため、限定する方向は直進方向とする。

4.2 信号線オーバーヘッド

優先度情報を先行して転送するためには追加の信号線が必要となる。優先度が 256 レベルあれば、理論的に RateMonotonic スケジューリングが可能⁸⁾ であるため、優先度が 8 ビットであることを想定すると、先読みアービトレーションに必要なビット数は、優先度に 8 ビッ

ト、先行して転送される優先度情報が有効であることを示すために 1 ビット、合計 9 ビットとなる。

リアルタイムシステム向けルータに必要な信号線は、フリット幅を 64 ビットとすると、フリットの転送に 64 ビット、フリットが有効であることを示すために 1 ビット、クレジットコントロールに 1 ビット、合計 66 ビットとなる。従って、先読みのために必要な信号線のオーバーヘッドは 13.6% と大きな値となる。

そこで、優先度情報を RC/LA ステージと SA ステージの 2 ステージに分けて転送する。これにより、優先度情報に必要な信号線は半分の 4 ビットにまで削減され、先行して転送する優先度情報が有効であることを示す 1 ビットと合わせて、必要な信号線は 5 ビットとなる。各ステージで転送する優先度情報の詳細は以下の通りである。

- **RC/LA ステージ:** 優先度上位 4 ビット + invalid 1 ビット

- **SA ステージ:** 優先度下位 4 ビット + valid 1 ビット

valid ビットを受信すると、1 サイクル前に受信した優先度情報 4 ビットと現在受信している優先度情報 4 ビットから優先度を算出し、先読みアービトレーションを行う。

この方式により、先読みアービトレーションに必要な信号線は 5 ビットとなり、信号線のオーバーヘッドは 7.6% にまで削減される。

4.3 ネットワーク上における動作

図 3 に、先読みアービトレーション機構付ルータのネットワーク上における動作例を示す。次元順ルーティングであるので、経路は図に示されている経路で固定である。

パケットが直進方向に進むルータでは優先度情報が先行して転送されている。優先度情報が先行して転送されたルータでは先読みアービトレーションが行われるため、ルータ遅延が削減される。図の例では、通過する 8 つのルータの内、4 つのルータでルータ遅延を削減されており、パケットの経路の半分が直進方向に進んでいる。

4.4 パイプライン構成

先読みアービトレーション機構付ルータのパイプライン構成を図 4 に示す。先読みアービトレーション機構付ルータでは、優先度情報を出力ポートに転送する Priority Transmit (PT) が追加されている。PT1 で優先度上位 4 ビットを、PT2 で優先度下位 4 ビットを転送する。

1 サイクル目の RC/LA ステージに PT1 を追加し、2 サイクル目の SA ステージに PT2 を追加する。PT2 による優先度情報の転送が完了した 3 サイクル目には、Router 2 では LA が実行可能となり、Router 1 の ST ステージと Router 2 の LA ステージがオーバーラップしている。4 サイクル目では、パケットが到着してすぐに SA ステージの実行が行われている。

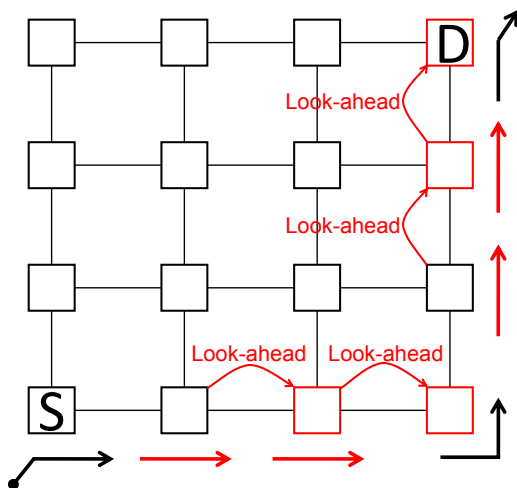


図3 ネットワーク上における先読みアービトレーション機構付ルータの動作例(4x4 2次元メッシュ)
Fig.3 Network behavior example of pre-arbitration router (4x4 2-D Mesh)

ヘッダ情報が届くのは Router 1 の ST ステージの後であるため、Router 2 では RC の実行を SA と同じステージで行う。

パケットが直進方向に進んでいない場合は、通常のルータと同じパイプライン構成となる。

4.5 実装

図3に、先読みアービトレーション機構付ルータのアーキテクチャを示す。先読みアービトレーション機構付ルータでは Priority Transmitter モジュールを追加し、入出力ポートには優先度情報を先行して転送するための信号線を追加する。

Priority Transmitter は、優先度情報の先行転送を行うモジュールであり、通常時は、各 Input Unit のパケットの進行方向にかかわらず、各 Input Unit のパケットの優先度上位 4 ビットと invalid シグナルを直進方向のルータへ送信する。パケットが実際に直進方向に転送される場合は、そのパケットの優先度下位 4 ビットと valid シグナルを送信する。

5. 評価

5.1 評価環境

2章で述べたベーシックルータと提案手法の先読みアービトレーション機構付ルータを評

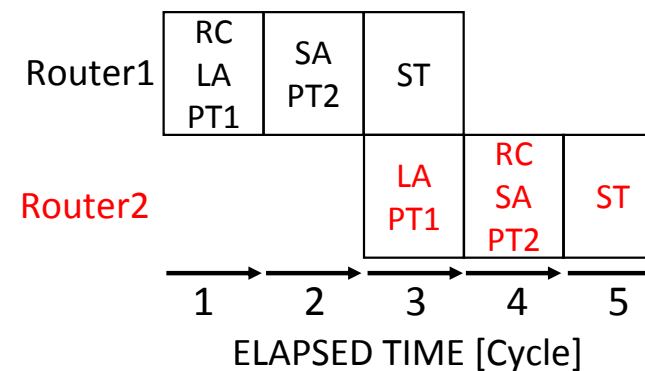


図4 先読みアービトレーション機構付ルータのパイプライン構成
Fig.4 Pre-arbitration router pipeline

表1 ネットワークパラメータ

Table 1 Network parameters

ネットワークポロジ	2次元メッシュ
コア数	8×8
ルーティング	次元順ルーティング
仮想チャンネル数/ポート	4
フリットサイズ	64ビット
パケット長	9フリット

価し、比較を行う。

ベーシックルータおよび先読みアービトレーション機構付ルータは Verilog HDL を用いて実装し、NC-Verilog を用いてシミュレーションによる評価を行い有効性について検討する。

評価に用いたパラメータを表1に、トラフィックパターン表2に示す。Bit complement は平均ホップ数は長いですが、次元順ルーティングにおいて衝突が生じにくく、Transpose は次元順ルーティングにおいて衝突の生じやすい。

シミュレーションでは、最初の 2000 サイクルでネットワークにパケットを注入し、続く 20000 サイクルの期間で評価を行った。優先度は各コアにランダム割り当て、最高優先度のコアは複数存在しないようにした。

ヘッダフリットがネットワークに注入されてから、出てくるまでのサイクル数を転送遅延とする。

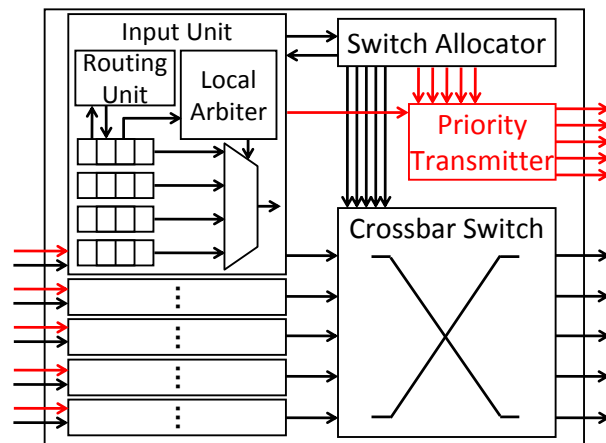


図 5 先読みアービトレーション機構付ルータのアーキテクチャ
Fig.5 Pre-arbitration router architecture

表 2 トラフィックパターン
Table 2 Traffic patterns

トラフィックパターン	デスティネーションアドレス
Uniform Random	ランダム
Bit complement	ソースアドレスをビット反転
Transpose	ソースアドレスの上位ビットと下位ビットを交換

5.2 スループット

図 7 に Uniform Random, 図 9 に Bit complement, 図 11 に Transpose のトラフィックに対するスループットを示す。

いずれのトラフィックパターンにおいても, 先読みアービトレーション機構付ルータとベーシックルータのスループットに差異がほとんど見られない。これは, 高優先度パケットのルータ遅延が削減されたことで, 低優先度パケットがブロックされやすくなったため, スタベーションが生じやすくなり, 転送遅延削減の効果が相殺されたことが原因である。

トラフィックを増加させていったときに, Uniform Random が最もネットワークの飽和が遅く, 次に Bit complement, そして Transpose が最も飽和が速い。これは, Uniform Random は最も負荷が分散するため, 飽和が遅く, Bit complement は次元順ルーティングにおいて衝

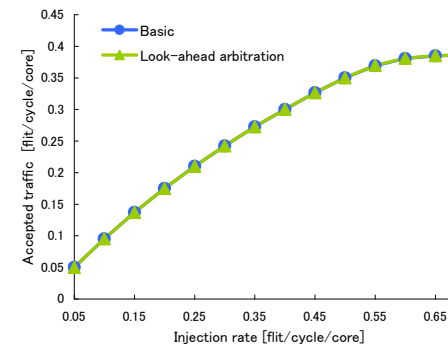


図 6 スループット (Uniform Random)

図 7 Throughput (Uniform Random)

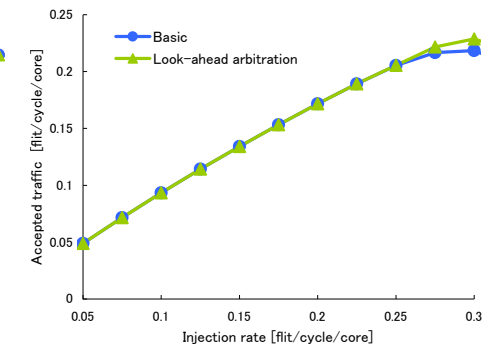


図 8 スループット (Bit complement)

図 9 Throughput (Bit complement)

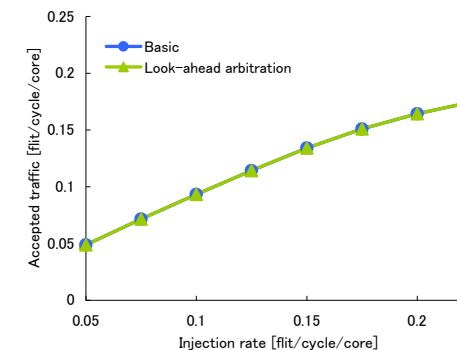


図 10 スループット (Transpose)

図 11 Throughput (Transpose)

突が生じづらいため, 次に飽和が遅い。Transpose は次元順ルーティングにおいて, 衝突が生じやすいトラフィックであるため, 飽和が速かった。

5.3 転送遅延

図 12 に Uniform Random, 図 13 に Bit complement, 図 14 に Transpose のトラフィックに対する転送遅延を示す。いずれのトラフィックパターンにおいても, 先読みアービレー

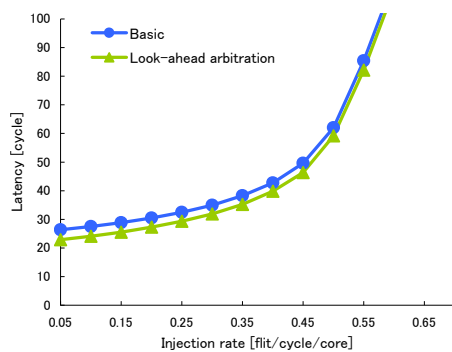


図 12 転送遅延 (Uniform Random)
Fig. 12 Transfer latency(Uniform Random)

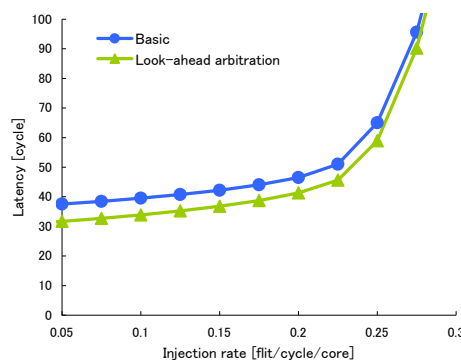


図 13 転送遅延 (Bit complement)
Fig. 13 Transfer latency(Bit complement)

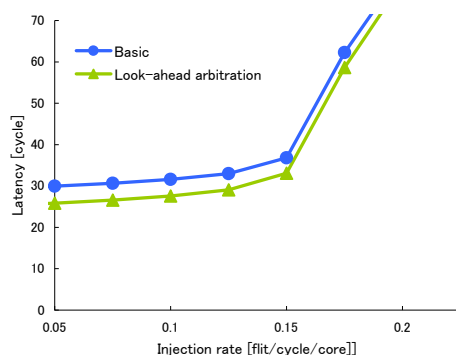


図 14 転送遅延 (Transpose)
Fig. 14 Transfer latency(Transpose)

シヨンルータの転送遅延はベーシックルータよりも低く抑えられている。

表 3 に各トラフィックパターンにおける平均転送遅延を示す。Bit complement はデスティネーションが遠くなりやすいため、次元順ルーティングにおいて、パケットの直進が生じやすい。そのため、他のトラフィックパターンよりも転送遅延の削減幅が大きくなっている。

表 3 平均転送遅延 (cycles)

Table 3 Average transfer latency(cycles)

	ベーシックルータ	先読みアービトレーションルータ
Uniform Random	61.2	58.2
Bit complement	57.0	51.5
Transpose	51.9	48.3

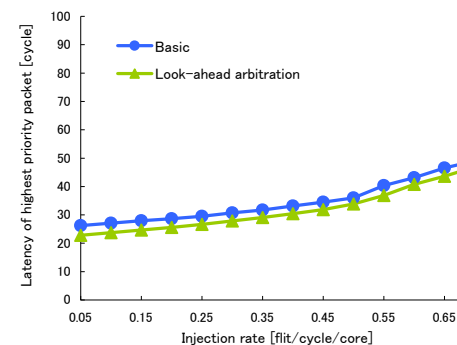


図 15 最高優先度パケットの転送遅延 (Uniform Random)
Fig. 15 Transfer latency of highest priority packets(Uniform Random)

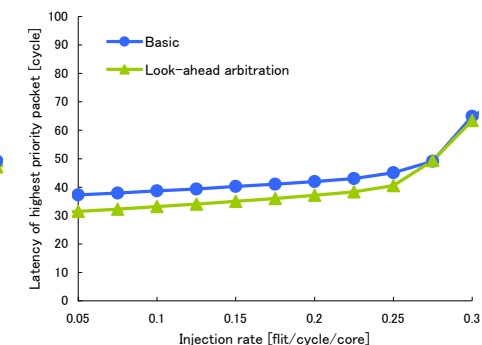


図 16 最高優先度パケットの転送遅延 (Bit complement)
Fig. 16 Transfer latency of highest priority packets(Bit complement)

5.4 最高優先度パケットの転送遅延

図 15 に Uniform Random, 図 16 に Bit complement, 図 17 に Transpose のトラフィックに対する最高優先度パケットの転送遅延を示す。いずれのトラフィックパターンにおいても、先読みアービトレーションルータの最高優先度パケットの転送遅延はベーシックルータよりも低く抑えられている。

表 4 に最高優先度パケットの平均転送遅延を示す。Uniform Random では平均 8.1%, Bit complement では平均 10.3%, Transpose では平均 10.7%の転送遅延が削減された。

5.5 動作周波数とハードウェアコスト

提案手法の先読みアービトレーションルータの動作周波数とハードウェアコストについて評価を行う。まず、動作周波数とハードウェアコストの見積もりとして Synopsys 社の Design Compiler に TSMC90nm プロセスを用いて、論理合成を行った。

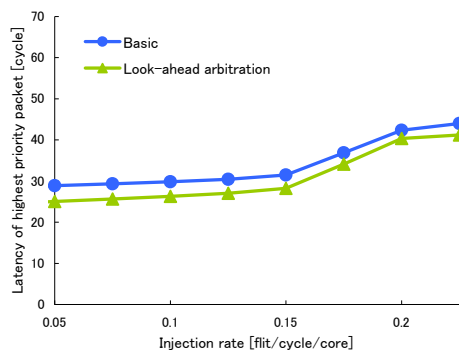


図 17 最高優先度パケットの転送遅延 (Transpose)
Fig. 17 Transfer latency of highest priority packets(Transpose)

表 4 最高優先度パケットの平均転送遅延 (cycles)
Table 4 Average transfer latency of highest priority packets(cycles)

	ベーシックルータ	先読みアービトレーションルータ
Uniform random	34.6	31.8
Bit complement	42.9	38.5
Transpose	37.3	33.3

表 5 論理合成結果
Table 5 Result of logic synthesis

	ベーシックルータ	先読みアービトレーションルータ
動作周波数 [GHz]	1.08	1.08
面積 [mm^2]	0.360778	0.369742

表 5 に 1GHz で論理合成を行った結果を示す。動作周波数の低下は 1%未満に収まり、面積の増加率は 2.5 %となった。

6. ま と め

本研究では、優先度逆転問題を生じさせずにルータ遅延を削減する先読みアービトレーション機構付ルータの設計および実装を行った。先読みアービトレーションに必要な優先度情報の先行転送を、転送方向を直進に限定することで解決し、優先度情報を 2 ステージに

分けて転送することで、追加の信号線のオーバヘッドを抑えた。先読みアービトレーション機構付ルータと通常のルータの最高優先度パケットの転送遅延を比較した結果、7.6%の信号線の増加と 2.5%のルータ面積の増加で最高優先度パケットの転送遅延を平均 8.1%削減することができた。また、転送遅延は削減されたが、スループットは向上しないことを確認した。

謝辞 本研究は科学技術振興機構 CREST の支援によるものであることを記し、謝意を表す。また、本研究の一部は文部科学省グローバル COE プログラム「環境共生・安全システムデザイン」の先導拠点に依るものであることを記し、謝意を表す。

参 考 文 献

- 1) W.J. Dally and B.Towles: "Route Packets, Not Wires: On-Chip Interconnection Networks", Proceedings of the Design Automation Conference (DAC'01), pp. 684-689 (2001).
- 2) D. Wentzlaff et al.: "On-chip interconnection architecture of the Tile processor", IEEE Micro (2007).
- 3) L.-S. Peh and W.J. Dally: "A Delay Model and Speculative Architecture for Pipelined Routers", In Proceedings of the 7th International Symposium on High-Performance Computer Architecture, pp. 255-266 (2001).
- 4) G. Michelogiannakis, D. Pnevmatikatos, and M. Katevenis: "Approaching Ideal NoC Latency with Pre-Configured Routes", In Proceedings of the ACM/IEEE International Symposium on Networks-on-Chip (2007).
- 5) Duato, J., Yalamanchili, S. and Ni, L. M.: "Interconnection Networks: An Engineering Approach", Morgan Kaufmann (2002).
- 6) Hiroki Matsutani, Michihiro Koibuchi, Hideharu Amano, and Tsutomu Yoshinaga: "Prediction Router: Yet Another Low Latency On-Chip Router Architecture", Proc. of the 15th IEEE International Symposium on High-Performance Computer Architecture (HPCA'09), pp. 367-378 (2009).
- 7) P.K. AmitKumar, Li-ShiuanPeh and N.K. Jha: "Express Virtual Channels: Towards the Ideal Interconnection Fabric", Proceedings of the International Symposium on Computer Architecture (ISCA'07), pp. 150-161 (2007).
- 8) Liu, J.W.: "REAL-TIME SYSTEMS", Prentice Hall (2000).