

高エネルギー効率ルータバイパス法 (EERB) による 省電力NoC

成子 貴洋^{1,a)} 平木 敬^{1,b)}

受付日 2014年7月23日, 採録日 2014年12月22日

概要: チップあたりのコア数が増加するのにもない, コア間の通信基盤としてチップ上のインターコネクションネットワークである Network-on-Chip (NoC) が用いられている. NoC はバスと比べ高いスループットを提供できるものの, コア数が大きくなるにつれネットワークのサイズが大きくなるため, チップ全体の性能や消費電力に及ぼす影響が大きくなるものとなっている. ルータバイパス法は, パケットの中間ルータの非同期的な通過を可能にする手法である. この手法を用いることで, ネットワークのレイテンシの削減が可能である. 一方で, ルータバイパス法の先行研究では消費電力の削減は得られない. 本論文では, 高エネルギー効率ルータバイパス法 (EERB) を提案する. EERB はルータバイパス法を活用し, 性能改善に加え, 消費電力の削減を実現する. シミュレーションにより, EERB を用いることで, 低レイテンシと低消費電力を同時に実現できることが示された.

キーワード: network-on-chip, 消費電力

Reducing Energy Consumption of NoC by Energy Efficient Router Bypassing (EERB)

TAKAHIRO NARUKO^{1,a)} KEI HIRAKI^{1,b)}

Received: July 23, 2014, Accepted: December 22, 2014

Abstract: Since the number of cores on a chip has been increasing, processors have on-chip interconnection networks to realize efficient intra-chip communication. These networks are called Network-on-Chip, or NoC. Although NoCs provide abundant throughput, they affect performance and power consumption of a chip because they become bigger in proportion to the core-count. Router bypassing technique is a technique which reduces network latency by enabling flits to skip intermediate routers asynchronously. This technique, however, cannot reduce power consumption. In this work, we propose Energy Efficient Router Bypassing (EERB) that employ router bypassing to reduce power consumption in addition to network latency. Simulation results show that EERB reduces both latency and power consumption of an NoC.

Keywords: network-on-chip, energy consumption

1. はじめに

ムーアの法則に従ってチップあたりに使用できるトランジスタ数は増加の一途をたどっているが, 命令レベル並列性によるマイクロアーキテクチャの性能改善は限界を迎え

つつある. そのため, 現在では1つのチップ上に複数のコアを配置するのが一般的である. しかしこれにより, 複数のコアやメモリコントローラの間での通信をいかに実現するかという問題が新たに生じている.

チップ内の通信基盤としてバスを候補にあげることができる. バスは, スヌーピングによるキャッシュコヒーレンシの管理が可能であるという利点を有するが, コア数が多い場合に十分なスループットを提供できないという問題がある. そのため, 現在ではチップ上のインターコ

¹ 東京大学大学院情報理工学系研究科
Graduate School of Information Science and Technology,
The University of Tokyo, Bunkyo, Tokyo 113-8656, Japan
a) cinccinaru@is.s.u-tokyo.ac.jp
b) hiraki@is.s.u-tokyo.ac.jp

ネットワークである Network-on-Chip (NoC) が一般に用いられている。NoC を用いるシステムにおいては、コア間の通信はパケットを用いて行われる。コアから送出されたパケットは、フリット (flow-control-digit, flit) と呼ばれる単位に分割されたうえで、ルータを経由しながら目的のコアまで転送される。ネットワークのトポロジとしては、配線の都合からリングやメッシュが好まれる。リングトポロジを採用した NoC としては Intel@ XeonPhi™ [6] のものがあげられるが、リングはノード数 N に対して $O(1)$ の二分スループットしか提供できない。一方で、メッシュトポロジは $O(\sqrt{N})$ の二分スループットを提供できる。Tilera@TILE64™ [2] などの製品は、NoC のトポロジとしてメッシュを採用している。メッシュトポロジの NoC は、バスに比べて大きなスループットを提供できる一方、パケットが経由するルータの数の期待値は $O(\sqrt{N})$ で増加するため、レイテンシが $O(\sqrt{N})$ で増加してしまうという課題をかかえる。ネットワークのレイテンシの増加はチップ全体の性能の低下を招く。そのため、メッシュなどの NoC に適したトポロジを保持したまま、レイテンシを削減する手法が必要である。たとえば、Express Virtual Channel [8], Prediction Router [9] などが提案された。SMART [7] はそのような手法の 1 つである。SMART は中間ルータにおいてパケットを非同期的に通過させることで、レイテンシの削減を図る。SMART を用いたネットワークにおいては、パケットは中間ルータのバッファを通過し、1 サイクルで複数ホップ進むことが可能である。この手法により、NoC のレイテンシが大幅に削減される。本論文では、このような中間ルータの通過を可能とする手法をルータバイパス法と呼ぶ。

レイテンシの削減に並び、消費電力の削減は NoC がかかえる課題である。たとえば、16 コアの MIT Raw プロセッサでは、NoC の消費電力がチップ全体の消費電力の 36% を占めると報告されている [11]。消費電力がシステム設計における主要な制約となっている現在においては、NoC における消費電力を削減することは重要な課題である。しかし、SMART はレイテンシの削減に効果的であるものの、消費電力削減の効果はほぼ得られない [7]。

本論文では、高エネルギー効率ルータバイパス法 (Energy Efficient Router Bypassing, EERB) を提案する。EERB は SMART のルータバイパス法を、レイテンシの削減だけでなく消費電力の削減にも活用する手法である。EERB を導入することで、バイパスの円滑化によるバッファ消費電力の削減、ならびにバイパス経路の見直しによるクロスバースイッチの消費電力の削減を試みる。

本論文の構成は以下のとおりである。2 章で高エネルギー効率ルータバイパス法のハードウェア構成と動作について説明する。3 章でルータのバイパスを円滑にするための最適化を導入する。4 章でシミュレーションによる提案

手法の性能評価を行う。5 章で EERB 導入にともなうオーバヘッドについて考察する。6 章で関連研究について議論する。最後に 7 章で本論文のまとめを行う。

2. 高エネルギー効率ルータバイパス法 (EERB)

2.1 EERB の基本動作

EERB を用いることで図 1 に描かれているように、フリットの間中ルータにおける非同期的な通過が可能になる。つまり、フリットは 1 サイクル内に複数ホップ進むことが可能となる。このようなルータのバイパスは、ネットワークのレイテンシを減少させるだけでなく、消費電力の削減にも効果的である。なぜならば、通過したルータにおいてはバッファやクロスバースイッチを使用しないため、これらの電力消費が発生しないからである。

図 2 は EERB のルータ構成を示している。簡単のため、2 つの入力ポートと 1 つの出力ポートのみ図示している。古典的な 5 サイクル仮想チャネル [5] ルータとの違いは、次のような構成要素の存在にある。

- バッファのバイパス経路
- Switch Allocation Global ユニット
- クロスバースイッチ直後のマルチプレクサ
- BR (Bypass Request) リンク
- クレジットリンクに代わる free_vc リンク
- 出力ポートごとの Asynchronous Repeater

この構成は SMART のルータとはほぼ同様であるが、マルチプレクサの位置が SMART ではクロスバースイッチの前方であったのに対し、EERB では後方である点が異なっ

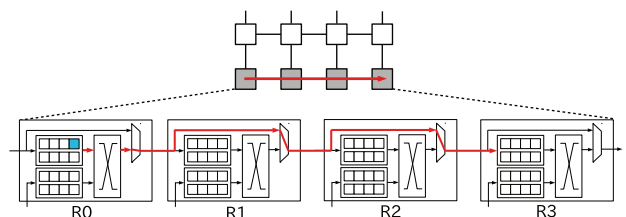


図 1 EERB によりフリットがルータをバイパスする例
Fig. 1 Example where a flit bypasses routers by EERB.

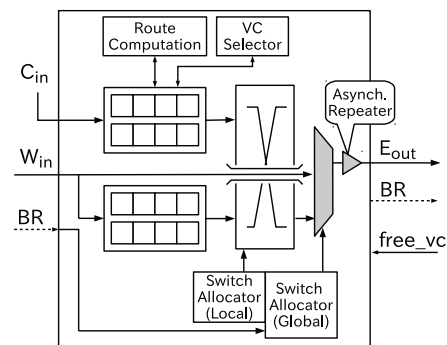


図 2 EERB のルータ構成
Fig. 2 Router architecture of EERB.

ている。この変更により、バイパス経路を通るフリットはクロスバースイッチを経由する必要がなくなるため、クロスバースイッチにおける競合を防ぐことができる。さらに、バイパスするルータにおけるクロスバースイッチの消費電力を削減できるという利点が得られる。一方で、この変更の欠点はバイパス路形成の自由度が下がる点である。SMARTではバイパス経路がクロスバースイッチを通るようにとられるため、任意の出力ポートへのバイパス経路を形成することが可能である。EERBではクロスバースイッチを迂回するようにバイパス経路が引かれるため、各入力ポートに対してバイパス先の出力ポートを1つに決め打ちする必要がある。より具体的には、EERBでは2次元メッシュの x 軸もしくは y 軸に沿った直線的なバイパス路のみを提供する。これらの得失に関する議論は6章で行う。

図1はフリットがルータ R0 からルータ R3 まで1サイクルで到達する例を示している。ルータ R0 は転送の始点(このようなルータを開始ルータと名付ける)であり、マルチプレクサはバッファ側が選択される。ルータ R1, R2 はフリットの中継を行うルータ(このようなルータを通過ルータと名付ける)であり、マルチプレクサはバイパス路側が選択される。転送の終点であるルータ R3 (このようなルータを終止ルータと名付ける)において、フリットはバッファに納められる。これらの各ルータの役割は後述のバイパス要求に基づき、サイクルごとに動的に変化する。ルータの視点に立つと、扱うフリットを2種類に分類することができる。1つは自ルータ内にバッファされていたフリット、もう1つは自ルータを通り過ぎるだけのフリットである。ここでは、前者を近郊フリット、後者を通過フリットと名付ける。1サイクルで移動可能なホップ数の最大値は配線遅延などに基づいてチップのデザイン時に決定される。以下ではこの値を HPC_{max} と表記する。

図3はフリットの転送が行われるまでの流れを示している。第1ステージは、VS, BW, RC, SA-L という4つの処理から成り立っている。これらの処理は開始ルータのみで行われる。それぞれ以下の処理を行う。

VS 空の仮想チャネル(バッファ)の選択。

BW VS で選んだ仮想チャネルへのフリットの書き込み。
RC 次の終止ルータ、そこまでのホップ数、そこへ到達するための出力ポートの計算。

SA-L クロスバースイッチ使用要求の調停。

第1ステージでクロスバースイッチの使用権が取得できると、第2ステージに進む。このステージはBRとSA-Gの2つの処理からなる。

BR RC で求めた経路上のルータへのバイパス要求信号の送出。(@開始ルータ)

SA-G バイパス要求の受諾・拒否の決定。(@通過ルータ)
 最後の第3ステージでは、SA-L や SA-G の結果に基づきマルチプレクサやクロスバースイッチのスイッチングが行われたうえで、経路上のクロスバースイッチやリンクを経由(ST & LT)したフリットの転送が行われる。

上述のとおり、フリットの転送に先行してバイパス要求の送出が行われる。バイパス要求の送出は専用の信号線であるBRリンクを介して行われる。信号線の幅は $\log_2(HPC_{max}+1)$ ビット、長さは HPC_{max} ホップである。この信号線は1つのルータによって駆動され、そのルータから特定の方向に HPC_{max} ホップ以内にある各ルータによって読まれる。たとえば、 $HPC_{max} = 4$ のときのBRリンクの配線は図4のようになる。バイパス要求は次のサイクルに開始ルータを発つフリットが希望する移動ホップ数を表している。つまり、ルータ A にバッファされているフリットが d ホップ先のルータ B までの移動を希望している場合、ルータ A はルータ B を通る経路のBRリンクに値 d を流す。ルータ A と B の中間にあるルータ C_1, \dots, C_{d-1} は値 d を受け取ることになるが、これらのルータはルータ A からの距離が d 未満であるため、値 d をバイパス要求と判断する。ルータ B の先にあるルータ C_{d+1} は、ルータ A からの距離が d より大きいため、受け取った値 d をルータ A からのバイパス要求とは判断しない。

EERBでは、自ルータ内の近郊フリットだけでなく、通過フリットを含めて出力ポートの調停を行う必要がある。SA-Gはこの調停を行うステージであり、以下のような方針で調停が行われる。

- あるルータにおいて、通過フリットと近郊フリットとの間で衝突が発生している場合、近郊フリットがつねに資源を獲得する。
- それ以外の場合、最も近傍のルータからの要求が受諾

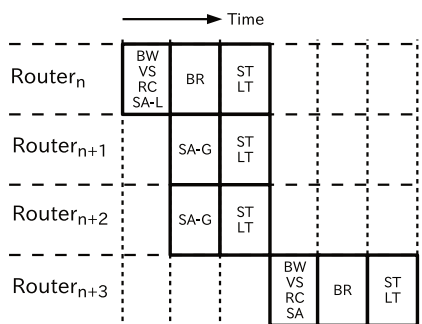


図3 EERBのパイプライン構成
 Fig. 3 Pipeline of EERB.

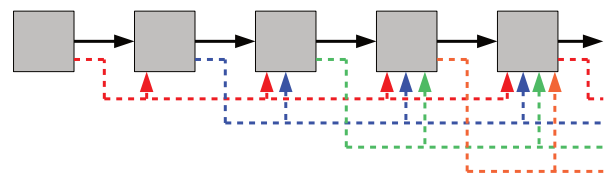


図4 $HPC_{max} = 4$ のときのBRリンクの配線例
 Fig. 4 Example of BR links with $HPC_{max} = 4$.

される。

つまり、そのバッファ内のフリットの転送が行われているような入力ポートが経路上に存在すると、通過フリットの転送はそのポートにおいて中断される。中断の有無を事前にフリットに通知することはせず、ST< ステージにおいて強制的に中断が行われる。

このような理由から、フリットはどのルータに停止するのかを事前に知る事ができないため、仮想チャネルの割当てを前もって行うことができない。そのため、到達したルータにおいてバッファの割当てが行われる (VS ステージ)。下流のルータの空きバッファの有無は、free_vc リンクを通して上流ルータに送られる。下流のルータに空きバッファが存在しない場合、上流のルータにバッファされているフリットは下流のバッファに空きが出るまで待機する必要がある。また、そのような下流のルータへの通過を試みるフリットは、上流のルータにおいて転送を中断する必要がある。

物理ネットワークが複数の論理ネットワークで共有されており、各論理ネットワークごとにバッファがあらかじめ割り当てられているような場合、free_vc リンクは論理ネットワークの数だけ必要になる。また、どの free_vc リンクを参照すればよいかを明確にするために、バイパス要求に論理ネットワーク番号を含める必要がある。

2.2 転送中断条件

出力ポートにおいて競合が発生した場合や下流のルータの空きバッファがない場合に、転送の中断が発生することは上で述べたとおりである。ここで、転送の中断が必要となる状況を以下にまとめる。

順序逆転 通過フリットが近郊フリットを追い越すことによるフリット間順序の逆転が発生する場合がある。point-to-point ordering を保つ必要がある場合や、1 パケットが複数フリットで構成されている場合には、保つべき順序を崩すような追い越しが発生する手前で転送を中断する必要がある。

出力ポート競合 マルチプレクサにおいて出力ポートに対する競合が発生した場合、通過フリットの転送を中断する必要がある。

空きバッファなし 下流のルータに空きバッファが確保できない場合、転送をその上流ルータにおいて中断する必要がある。

当然ネットワークの負荷が大きくなるにつれて、これらの転送中断条件の発生頻度は高くなる。頻繁な転送の中断はネットワークのレイテンシを増やすばかりでなく、NoC の消費電力の増加につながる。なぜならば、転送の中断によるバッファやアービタの使用が電力消費を招くからである。

3. 最適化

EERB を用いることで、通過ルータにおけるバッファやクロスバースイッチの消費電力を削減することが可能である。バッファとクロスバースイッチは NoC の消費電力の約 60% を占める構成要素である [7] ため、EERB は NoC 全体の消費電力削減に効果的であると期待できる。しかし、2.2 節で述べたような要因で転送の中断が発生すると、通過できるルータの数が減少するため、電力削減効果はその分だけ薄くなってしまふ。中断の発生頻度を減らし、電力削減効果を保つため、2 つの最適化 (区間番号、通過待ち) を導入する。

3.1 区間番号

フリット間の順序を保つための最も単純な方法は、通過フリットによる近郊フリットの追い越しを無条件に禁止することである。しかし、Point-to-Point Ordering では、送信元ノードか宛先ノードのいずれかが異なるフリット間の順序は保つ必要がないため、この方針では制限が強すぎるといえる。

過剰な追い越し禁止は、バイパス要求にフリットの送信元ノード番号と宛先ノード番号を含めることで、完全に排除することができる。つまり、バイパス要求を受け取ったルータは、次サイクルにやってくる通過フリットの転送を中断しなければならないか否かを、以下の基準に基づいて判断できる。

- バイパス要求に含まれるものと同一の送信元・宛先ノードのペアを持つフリットがバッファ内に存在する場合
有害な追い越しであるため、転送を中断。
- それ以外の場合
追い越しを許可。

この方法は不要な追い越し禁止をすべて除去できるものの、多くのハードウェアコストを要する。たとえば、コア数が 64 の場合、送信元・宛先ノードの組合せは $64 \times 63 = 4032$ 通りにのぼり、新たに $\lceil \log_2 4032 \rceil = 12$ 本のワイヤが各 BR リンクに必要なことになる。

ハードウェアコストを抑えるため、送信元・宛先ノード番号の代わりに、区間番号という値を導入し、バイパス要求に含める。区間番号は送信元ノード番号 s と宛先ノード番号 d に適当な圧縮関数 $f_{code} : \mathbb{N}^2 \rightarrow \mathbb{N}$ を適用して算出した値 $f_{code}(s, d)$ である。バイパス要求内の区間番号とバッファ内のフリットの区間番号を比較し、一致した場合には追い越しを禁止する。区間番号を用いる方法では、誤って問題のない追い越しを禁止してしまう場合が存在するが、適切に圧縮関数を定義することで誤りの確率を抑えることが可能である。たとえば、 8×8 のメッシュトポロジを仮定する。送信元ノードの x 座標を s_x とし、圧縮関数を

以下のように定義する.

$$f_{code}(s, d) = s_x \pmod 8$$

この関数は 4032 通りの端点の組合せを 8 通りの区間番号に圧縮する. これにより, BR リンクへのワイヤの追加本数を 12 本から 3 本に削減することができる. 関数の定義より, x 軸方向の移動中には誤った追い越しの禁止は発生しない. トラフィックが空間的に一様分布であると仮定すると, y 軸方向の移動中における誤りの確率は $7 \div 64 \approx 11\%$ である. 以上より, ネットワーク全体での誤った追い越し禁止の発生確率は 5.5% に抑えられると推定できる.

圧縮関数は要求するハードウェア資源量と得られる性能向上率との間にトレードオフの関係があるため, 想定されるネットワーク負荷や利用できるワイヤ資源の量などに応じて決定される必要がある.

3.2 通過待ち

マルチプレクサにおいて, 近郊フリットと通過フリットの間で出力ポートに対する競合が発生する可能性がある. この競合に対して, 近郊フリットを優先する方針と, 通過フリットを優先する方針の 2 通りの解法が存在する. 通過フリットを優先する方針をとると, 近郊フリットの出発が 1 サイクル遅れるものの, 通過フリットの転送中断とそれにとまなう 3 サイクルのルータパイプライン遅延を防ぐことができる. 直感的にはこのような方針のほうが効率が良いように思われるが, 低負荷時におけるレイテンシに両方針で差がほぼ見られないうえ, 通過フリット優先の方針は飽和スループットに関して近郊フリット優先の方針に大きく劣ることが分かっている [7]. なぜならば, 複数のバイパス要求信号 BR_1, BR_2, \dots, BR_n が重なりあうように出された場合, BR_i の通過フリットを優先するために BR_{i+1} の近郊フリットの送が見送られるという譲りの連鎖が発生し, 結果として 1 番目のフリットの送しが行われなためである. この結果に基づき, EERB は近郊フリットを優先する方針をとっている.

つねに近郊フリットを優先する方針は, 譲りの連鎖の防止には最適であるものの, 通過フリットの優先が効果的である場合が存在することは事実である. そのような場合に, 通過フリットの転送を継続しやすくするために, 通過待ちを導入する. 通過待ちは, バイパス要求の受け取りが予測される場合に, バッファされているフリットの送を遅らせることで, より円滑なバイパスを促す最適化である. バイパス要求の予測はアルゴリズム 1 によって行われる. このアルゴリズムは, 「2 サイクル前に 2 つ以上のフリットからバイパス要求を受け取っていた場合, いずれかのフリットは途中で転送の中断を被ったはずであり, 次サイクルでバイパス要求の再試行を行うであろう」という考察に基づいている. ただし, バイパス要求の再試行を行うと予測さ

Algorithm 1 Passage Wait

```

if there is a timeouted flit destined to  $P$  then
    return No passage wait
end if
 $S \leftarrow$  Set of sender routers of bypass requests in cycle  $t - 2$ 
 $R_1 \leftarrow$  The nearest router among  $S$ 
 $R_2 \leftarrow$  The second nearest router among  $S$ 
 $d_1 \leftarrow$  Hop-count between  $R_1$  and this router
 $d_2 \leftarrow$  Hop-count between  $R_2$  and this router
 $r_2 \leftarrow$  Hop-count requested by  $R_2$  in cycle  $t - 2$ 
 $vnet_2 \leftarrow$  Vnet number included in the bypass request from  $R_2$  in cycle  $t - 2$ 
if  $vnet_2$  is a Pt-to-Pt ordered vnet or a multi-flit packets vnet then
    return No passage wait
end if
if  $r_2 - (d_2 - d_1) \geq 2$  then
    return Do passage wait
else
    return No passage wait
end if
    
```

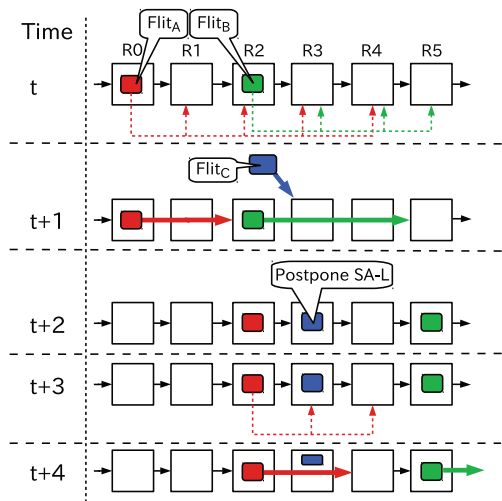


図 5 通過待ちの例

Fig. 5 Example of passage wait.

れるフリットが, point-to-point ordering の保持を約束している仮想ネットワークか, 複数フリットからなるパケットが通る仮想ネットワークに属している場合は, 通過待ちを行わない. なぜならば, 追い越し禁止により通過待ちが無駄になってしまう可能性があるからである. また, 連続した通過待ちの発生によるデッドロックを防ぐため, 各仮想チャンネルにフリットの待ち時間を管理するタイマを用意し, それがタイムアウトした仮想チャンネルが存在する場合には, 通過待ちを行わないこととする.

図 5 は通過待ちの例を示している. 時刻 $t + 2$ において, $Flit_C$ はアルゴリズム 1 に基づき, クロスバースイッチの要求を見送っている. R3 におけるバイパス要求の履歴を参照すると, 2 サイクル前の時刻 t において $Flit_A$ と $Flit_B$ の間で競合が発生していたことが分かるため, 次の時刻 $t + 3$ において再び $Flit_A$ からのバイパス要求を受け

取ると予測するのである。時刻 $t+3$ において、予測どおり $Flit_A$ がバイパス要求を送出する。 $Flit_C$ はクロスバースイッチの要求を見送っていたため、このバイパス要求は受諾される。時刻 $t+4$ において、 $Flit_C$ の追い越しをとまなう、 $Flit_A$ の R2 から R4 への転送が行われる。もしも $Flit_C$ が通過待ちを行っていないならば、 $Flit_A$ の転送は R3 において中断させられていたはずである。この例では、 $Flit_C$ の 1 サイクルの通過待ちにより、 $Flit_A$ のレイテンシが 3 サイクル削減できた。さらに、R3 におけるバッファやクロスバースイッチの消費電力が削減されている。このように通過待ちにより、レイテンシと消費電力を同時に削減することが可能である。

4. 性能評価

4.1 評価手法

EERB が性能や消費エネルギーに与える効果を調べるため、PARSEC ベンチマーク [3] による評価を行った。評価対象は、ルータのバイパスを行わない従来の NoC (Base), SMART [7], EERB の 3 種類である。シミュレータには gem5 [4] を用いた。gem5 はインターコネクションネットワークのシミュレーションに GARNET [1] を採用しており、本研究ではこの部分の実装を EERB の動作を模倣するよう適宜修正した。gem5 のパラメータは表 1 のとおりに設定した。パケットのルーティングには、次元順ルーティングを用いた。ルータ内のクロスバースイッチは 6 つのポート (East, West, North, South, CPU, Directory) を有する。消費電力の評価には Orion [10] を用いた。Orion に与えたパラメータは表 2 のとおりである。これらのパラメー

表 1 評価するシステムの構成

Table 1 Target system.

OS カーネル	Linux 2.6.27
ISA	ALPHA
CPU モデル	Timing Simple
L1 キャッシュ (Instruction)	Private 32 KB
L1 キャッシュ (Data)	Private 32 KB
L2 キャッシュ	Private 1 MB
コヒーレンス・プロトコル	MOESI hammer
ネットワークポロジ	4x8 Mesh
# of VCs per Vnet	4
フリットサイズ	128 ビット

表 2 消費電力算出に用いた仮定

Table 2 Parameters for energy evaluation.

プロセスルール	32 nm
電圧	1.0 V
動作周波数	1 GHz
コアサイズ	1 mm * 1 mm
クロスバースイッチ実装	マルチプレクサ・ツリー
入力バッファ実装	レジスタ

タに基づき、各機能部の 1 フリットあたりの消費エネルギーを、Orion [10] が自動で算出する。各機能部の最終的な消費エネルギーは、Orion が算出した 1 フリットあたりの消費エネルギーに各機能部の使用回数を掛けたものとして計算する。EERB の評価では、通過ルータにおけるバッファとクロスバースイッチの使用回数をインクリメントしないことで、バイパスの効果を評価する。EERB 固有のパラメータは表 3 のとおりに与えた。EERB, SMART における HPC_{max} は 7 に設定した。この値は SMART で達成可能とされる $HPC_{max} = 11$ [7] よりも小さな値である。EERB ではクロスバースイッチを経由しない分、バイパス経路の遅延が SMART よりも短縮されるため、 $HPC_{max} = 7$ という仮定は妥当である。7 よりも大きな値としないのは、ネットワークポロジが 4×8 Mesh であるため、7 よりも大きな HPC_{max} は無意味だからである。EERB や SMART のバイパス経路は East↔West, North↔South の直線方向 4 経路のみを設けた。Mesh エッジ部分のルータでは West→North のようなバイパスが可能であるが、本評価はこのようなバイパス経路を含めずに行う。north 出力ポートに { バッファ側, South からのバイパス経路, West からのバイパス経路 } の 3 経路を切り替えるマルチプレクサが必要となり、実装コストや消費電力の増加を招くからである。エッジ部分のバイパス経路を除いた評価は、EERB の効果の下限を調べられるという観点からも妥当である。

各ベンチマークアプリケーションは 32 本のスレッドを生成し並列処理を行う。スレッドのコアへの割当てが動的に変化することによる結果の望ましくない変動を防ぐため、32 本のスレッドと 32 個のコアの対応を静的に固定した。シミュレーション結果は、アプリケーションの並列実行部分におけるもののみを用い、初期化や終了処理における結果は含まれない。

4.2 結果

図 6 は各アプリケーションにおけるネットワークレイテンシの結果を示している。EERB を用いることで、従来の NoC と比べ、ネットワークレイテンシが平均 31%削減されている。SMART と比べて平均 6%の削減が得られている。EERB が SMART 以上の低レイテンシを実現できたのは以下の 2 つの理由による。1 つ目は、マルチプレクサの位置をクロスバースイッチの後方へ変更したためである。EERB では図 9 のような場合に通過フリットの転送

表 3 EERB 固有のパラメータ

Table 3 Parameters of EERB.

HPC_{max}	7
区間番号生成関数	$f_{code}(s, d) = s \bmod 8$
通過待ちタイムアウト時間	6 サイクル

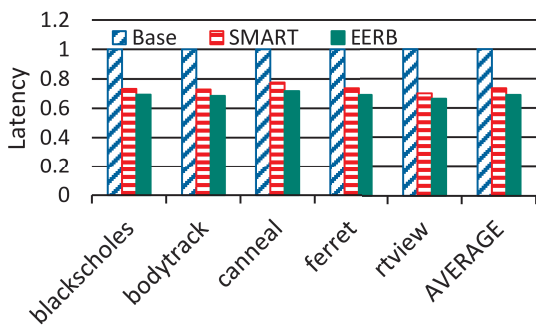


図 6 各アプリケーション実行時のネットワークレイテンシ
Fig. 6 Results of the network latency.

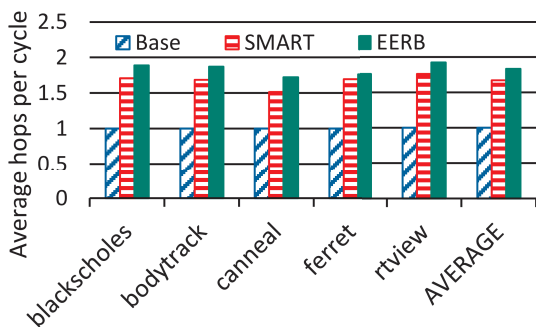


図 7 各アプリケーションにおいて達成されたバイパスの平均ホップ数
Fig. 7 Results of average hops per cycle.

の中断が回避され、通過フリットと近郊フリットの転送が同時に行われる。一方 SMART では、通過フリットはクロスバースイッチを経由する必要があるため、クロスバースイッチの入力ポートに対する競合が発生し、通過フリットの転送を中断する必要がある。よって、図 9 のような場合には EERB の方が SMART よりもレイテンシの面で有利である。2 つ目は、最適化の導入により EERB におけるバイパスの中断頻度が減少したためである。図 7 は各アプリケーションにおいて達成されたバイパスの平均ホップ数を表している。バイパスを行わない従来の NoC ではこの値はつねに 1 となる。EERB における平均バイパス・ホップ数は、SMART における値よりも平均 10% 大きい。この結果は、EERB は SMART と比べ、バイパスの中断頻度が少ないことを表している。

図 8 は各アプリケーションの、バッファとクロスバースイッチにおける動的消費エネルギーの結果を示している。上述のとおり、バッファとクロスバースイッチは NoC 全体の消費電力の約 60% を占める [7] ため、これらの消費電力を削減することは重要である。EERB を用いることで、バッファとクロスバースイッチにおける消費エネルギーが、従来の NoC と比べて平均 37%、SMART と比べて平均 36%、それぞれ削減されることがグラフから確認できる。EERB が消費エネルギーの面で SMART に大きく優っているのは、EERB ではマルチプレクサがクロスバースイッチの後方に配置されているためである。マルチプレクサが

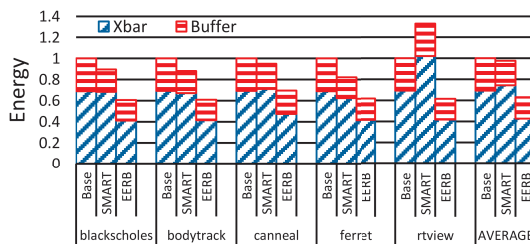


図 8 各アプリケーション実行時のバッファとクロスバースイッチの動的消費エネルギー
Fig. 8 Results of dynamic energy consumption of buffers and crossbar switches.

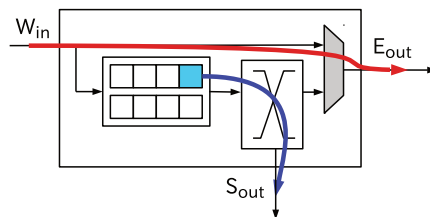


図 9 マルチプレクサの位置変更により通過フリットの転送の中断が回避される例
Fig. 9 Example where a bypass flit keeps its traversal by placing the multiplexer behind the crossbar switch.

クロスバースイッチの後方に配置されていることにより、通過フリットは中間ルータにおいてクロスバースイッチを経由する必要がなくなり、クロスバースイッチの消費エネルギーの大幅な削減が実現されている。

従来の NoC を比較基準とした EERB の効果は、ネットワークレイテンシの削減に関して 31% であるのに対し、バッファとクロスバースイッチにおける動的消費エネルギーの削減に関しては 37% であり、後者により大きく現れている。この結果をもたらした 1 つ目の原因は、ネットワークレイテンシにはネットワークインタフェースにおけるレイテンシが含まれる点にある。あるパケットが n 個のルータを経由しネットワーク上を移動する場合を考える。EERB によりうち m 個のルータをバイパスできたとする。1 パケットが消費するルータあたりのバッファとクロスバースイッチのエネルギーを e とする。コンテンションが発生しないと仮定すると、従来の NoC におけるレイテンシは $3(n+1)$ 、バッファとクロスバースイッチでの消費エネルギーは en である。レイテンシの計算において n に 1 が足されているのは、ネットワークインタフェースにおける遅延を加えるためである。EERB を用いた場合のレイテンシは $3(n-m+1)$ 、バッファとクロスバースイッチでの消費エネルギーは $e(n-m)$ と書ける。よって、EERB によるレイテンシの削減率は $m/(n+1)$ 、バッファとクロスバースイッチにおける動的消費エネルギーの削減率は m/n となる。任意の自然数 n, m について $m/(n+1) < m/n$ であるから、動的消費エネルギーの効果はレイテンシの効果よりも大きく現れる。2 つ目の原因は、通過待ちを行う側

のレイテンシ増加にある。通過待ちによるバイパス・ホップ数の増加は、消費エネルギー e の削減をもたらすが、レイテンシの削減は最大でも 2 サイクルにとどまる。バイパス中断を免れたフリットのレイテンシは 3 サイクルだけ減少するものの、待たされたフリットのレイテンシは 1 サイクル増加するためである。以上 2 つの原因により、EERB の効果は消費エネルギーにより大きく現れる。

5. EERB のオーバーヘッド

4 章では、EERB を用いることで従来の NoC と比較し、バッファとクロスバースイッチの消費エネルギーを平均 37%削減できることを示した。本章では、EERB で新たに必要になる以下の追加要素 (Switch Allocation Global ユニット, BR リンク, free_vc リンク, Asynchronous Repeater, バイパス・バッファ選択用マルチプレクサ, 圧縮関数演算器, 待ち時間タイマ) が消費する電力を考慮しても、ルータの消費電力が削減できることを示す。

Switch Allocation Global ユニット, BR リンクが消費する動的消費電力は、NoC 全体の動的消費電力の 3%に満たない [7]。従来の NoC では各仮想チャネルごとに 1 本のクレジットリンクが必要であるが、EERB では各仮想ネットワークごとに 1 本の free_vc リンクを用意すればよい。クレジットリンクを free_vc リンクで置き換えることで配線リソースの使用量を削減することが可能であり、この置き換えにともない消費電力が増加することはない。Asynchronous Repeater の消費エネルギーは、従来の Clocked Driver の消費エネルギーよりも 14.3%小さく [7]、Asynchronous Repeater を用いることによる消費電力面でのオーバーヘッドは発生しない。

マルチプレクサにおける動的消費電力は、最大で従来の NoC のクロスバースイッチにおける動的消費電力の 13%と見積もることができる。クロスバースイッチのポート数は 6、スイッチを構成するのに必要なポートあたりのマルチプレクサ数は 5 であり、クロスバースイッチに含まれるマルチプレクサ数は 30 である。よって、マルチプレクサ 1 つあたりの消費電力はクロスバースイッチ全体の消費電力 E_x の 30 分の 1 と見積もることができる。バイパス側とバッファ側を選択するマルチプレクサは各ルータにたかだか 4 個であり、これらのマルチプレクサの追加による動的消費電力のオーバーヘッドは最大で $\frac{1}{30}E_x \times 4 \approx 0.13E_x$ と書ける。この値は EERB によるクロスバースイッチの動的消費電力の削減幅 37%よりも十分小さい値である。

区間番号の計算に用いる圧縮関数にはソースノードの x 座標を用いている。ネットワークのトポロジは 4rows \times 8columns Mesh であるから、区間番号はソースノード番号の下位 3 ビットを取り出すことで得られる。よって、区間番号の計算においてオーバーヘッドは発生しない。

通過待ちは 6 サイクルの待ち時間でタイムアウトする設

計となっており、待ち時間はフリットあたり 6 ビットのシフトレジスタにより管理できる。これらのシフトレジスタの消費電力はバッファの消費電力の 4.7%以下である。バッファは 128 bit 幅 5 エントリの計 640 ビットのフリップフロップにより構成されている。よって、タイマに用いるシフトレジスタの動的消費電力は最大で、従来の NoC におけるバッファの動的消費電力の $(6 \times 5) / (128 \times 5) \times 100 \approx 4.7\%$ 程度と見積もることができる。このオーバーヘッドは EERB によるバッファの動的消費電力の削減幅 37%よりも十分小さい値である。

以上の議論から、EERB の導入に必要なオーバーヘッドを考慮しても、ルータ全体で消費電力の削減が得られる。

6. 関連研究

6.1 SMART

ルータバイパス法を採用する手法として SMART [7] がある。EERB は SMART から多くのアイデアを継承している。バイパス先が入力ポートの反対側に固定される EERB とは異なり、SMART では X 方向から Y 方向へのターンをとともなうバイパスを形成できる。バイパス形成の自由度が高いのは、マルチプレクサがクロスバースイッチの前方に配置されており、通過フリットがクロスバースイッチを経由して任意の出力ポートを利用可能だからである。一方で、ターンをとともなうバイパスの実現には以下に述べる 4 つの欠点をとともなう。まず、EERB と比べ、必要な BR リンクの本数が増加するという点である。カーブをとともなうバイパスを形成するには、その骨組みとなる BR リンクを 2 次元状に配置する必要がある。たとえば $HPC_{max} = 7$ の場合、出力ポートあたりの BR リンクの本数は、EERB では 1 本なのに対し、SMART では 13 本と大きな差がある。各 BR リンクは 3 ビット幅 7 ホップ長であるから、フリットのデータサイズが 128 ビットの場合、EERB における BR リンクのオーバーヘッドは、

$$(3 \times 7) \div 128 \times 100 \approx 16\%$$

ターンをとともなうバイパスを許す SMART における BR リンクのオーバーヘッドは、

$$(3 \times 7 \times 13) \div 128 \times 100 \approx 213\%$$

よって、前者と比べると、後者のオーバーヘッドは非常に大きいといえる。2 つ目は、Switch Allocation Global ユニットの複雑化と消費電力の増加 [7] を招く点である。ターンをとともなうバイパスの存在により SA-G ステージにおける競合の組合せが増えるからである。3 つ目は、バイパス中断の確率が大きくなるという点である。EERB ではクロスバースイッチの出力ポートにおける競合がなければバイパスを継続することができるが、SMART では入力・出力ポートの両方が空いている必要がある。最後に、クロ

スバースイッチの消費電力の削減が得られない点である。EERB のバイパス経路はクロスバースイッチを経由しないため、通過ルータにおけるクロスバースイッチの消費電力を削減することができるが、SMART では通過ルータにおいてもクロスバースイッチを経由する必要があるため、クロスバースイッチの消費電力の削減は得られない。

ターンをとまなうバイパスを許しても、アプリケーションの実行時間は 4%程度しか削減されない [7]。以上の理由から、EERB ではバイパスの自由度を落とし、クロスバースイッチの位置変更による恩恵を享受する選択をとった。

6.2 Express Virtual Channel

経路上のルータにおけるレイテンシや消費電力の削減を目的とした先行研究として Express Virtual Channel (EVC) [8] があげられる。EVC は中間ルータにおける仮想チャネルやクロスバースイッチの割当てを約束する仮想チャネルを設けることで、 k ホップ先のルータまで k サイクルで到達可能にする手法である。EERB では途中でバイパスが中断される可能性があるため、各仮想チャネルに 1 パケット分のバッファを用意しておく必要があり、バッファが効率良く利用されているとはいえない。EVC では、 k ホップ先まで競合がいつさい発生しないことが保証されているため、wormhole routing が利用でき、バッファの効率的な活用が実現されている。また、文献 [8] はバッファの利用効率をさらに改善するため、プール内のバッファを仮想チャネルに動的に割り当てる手法を提案している。バッファの利用効率の悪さは EERB や SMART の大きな欠点の 1 つであり、動的なバッファの割当てはその欠点の克服に利用できる可能性がある。

7. おわりに

本論文では、NoC の消費電力を削減するために高エネルギー効率ルータバイパス法 (EERB) を提案した。EERB のバイパス経路はバッファやクロスバースイッチを迂回するように引かれているため、これらの構成要素の消費電力の削減が期待できる。しかし、競合などの要因によりバイパスの中断が発生すると、EERB の効果が薄くなってしまふという問題がある。この問題を解決するため、2 つの最適化を導入した。1 つ目の最適化である区間番号は、フリット間の順序を保つ必要がない安全な追い越しをルータが検出できるようにすることで、不必要な中断を削減する手法である。2 つ目の最適化である通過待ちは、次サイクルにバイパス要求信号の受信が予測される場合に、バッファされているフリットの転送を遅らせることで、より円滑なバイパスを促す手法である。

EERB の効果を確認するためにシミュレーションによる評価を行った。評価結果から、EERB を用いることで、すべての中間ルータを 1 ホップずつ進む従来の NoC と比

べ、ネットワークレイテンシを平均 31%、バッファとクロスバースイッチにおける消費エネルギーを平均 37%、それぞれ削減できることが確かめられた。さらに、ルータバイパス法を採用する既存手法である SMART と比較して、EERB はネットワークレイテンシ、ならびに消費電力の両面において優位であることが示された。これらの結果から、EERB はルータバイパス法を、パフォーマンスの向上だけでなく、消費電力の削減に活用することのできる手法であるといえる。

謝辞 有益なご意見を数多くくださった匿名のレビュアーの方に感謝する。

参考文献

- [1] Agarwal, N., Krishna, T., Peh, L.-S. and Jha, N.: GARNET: A detailed on-chip network model inside a full-system simulator, *IEEE International Symposium on Performance Analysis of Systems and Software, 2009, ISPASS 2009*, pp.33–42 (online), DOI: 10.1109/ISPASS.2009.4919636 (2009).
- [2] Bell, S., Edwards, B., Amann, J., Conlin, R., Joyce, K., Leung, V., MacKay, J., Reif, M., Bao, L., Brown, J., Mattina, M., Miao, C.-C., Ramey, C., Wentzlaff, D., Anderson, W., Berger, E., Fairbanks, N., Khan, D., Montenegro, F., Stickney, J. and Zook, J.: TILE64 — Processor: A 64-Core SoC with Mesh Interconnect, *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, pp.88–598 (online), DOI: 10.1109/ISSCC.2008.4523070 (2008).
- [3] Bienia, C.: Benchmarking Modern Multiprocessors, Ph.D. Thesis, Princeton University (2011).
- [4] Binkert, N., Beckmann, B., Black, G., Reinhardt, S.K., Saidi, A., Basu, A., Hestness, J., Hower, D.R., Krishna, T., Sardashti, S., Sen, R., Sewell, K., Shoaib, M., Vaish, N., Hill, M.D. and Wood, D.A.: The Gem5 Simulator, *SIGARCH Comput. Archit. News*, Vol.39, No.2, pp.1–7 (online), DOI: 10.1145/2024716.2024718 (2011).
- [5] Dally, W.: Virtual-channel flow control, *IEEE Trans. Parallel and Distributed Systems*, Vol.3, No.2, pp.194–205 (online), DOI: 10.1109/71.127260 (1992).
- [6] Intel: Intel®Xeon Phi™Product Family, available from <http://www.intel.com/content/www/us/en/processors/xeon/xeon-phi-detail.html>.
- [7] Krishna, T., Chen, C.-H.O., Kwon, W.C. and Peh, L.-S.: Breaking the On-chip Latency Barrier Using SMART, *Proc. 2013 IEEE 19th International Symposium on High Performance Computer Architecture (HPCA), HPCA '13*, pp.378–389, IEEE Computer Society (online), DOI: 10.1109/HPCA.2013.6522334 (2013).
- [8] Kumar, A., Peh, L.-S., Kundu, P. and Jha, N.K.: Express Virtual Channels: Towards the Ideal Interconnection Fabric, *SIGARCH Comput. Archit. News*, Vol.35, No.2, pp.150–161 (online), DOI: 10.1145/1273440.1250681 (2007).
- [9] Matsutani, H., Koibuchi, M., Amano, H. and Yoshinaga, T.: Prediction router: Yet another low latency on-chip router architecture, *IEEE 15th International Symposium on High Performance Computer Architecture, 2009, HPCA 2009*, pp.367–378 (online), DOI: 10.1109/HPCA.2009.4798274 (2009).
- [10] Wang, H.-S., Zhu, X., Peh, L.-S. and Malik, S.: Orion:

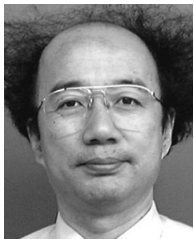
A power-performance simulator for interconnection networks, *Proc. 35th Annual IEEE/ACM International Symposium on Microarchitecture, 2002, (MICRO-35)*, pp.294-305 (online), DOI: 10.1109/MICRO.2002.1176258 (2002).

- [11] Wang, H., Peh, L.-S. and Malik, S.: Power-driven Design of Router Microarchitectures in On-chip Networks, *Proc. 36th Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 36*, IEEE Computer Society, pp.105-116 (2003).



成子 貴洋

1991年生。2014年東京大学理学部情報科学科卒業。現在、同大学大学院情報理工学系研究科コンピュータ科学専攻修士課程。計算機アーキテクチャの研究に従事。ACM会員。



平木 敬 (正会員)

東京大学大学院理学系研究科物理学専門課程修了，理学博士。電子技術総合研究所，IBM社T.J.Watson研究センター，東京大学理学部助教授を経て，現在，東京大学情報理工学系研究科教授。コンピュータ・アーキテクチャ，

並列分散計算，再構成可能デバイスを用いた計算，超高速ネットワーク，分散共有ファイルシステム研究に従事。