計算機アーキテクチャ　121－3
設　計　自　動　化　82－3
（1996．12．12）

VHDLによるハイパクロスバ網用ルータ・チップの設計

村上 祥基†　朴 泰祐†　中村 宏‡　中澤 喜三郎§

†筑波大学 電子・情報工学系

‡東京大学 先端科学技術研究センター

§電気通信大学 情報工学科

あらまし

ハイパクロスバ・ネットワークは，超並列計算機用相互結合ネットワークとして，高い性能と柔軟性を持つ．同ネットワークにおける適応ルーティングに関するこれまでの研究では，主として計算機シミュレーションによる評価のみが行なわれ，その際，固定ルーティングの場合と同一クロックレートでシステムが動作すると仮定されてきた．

本研究では，VHDLを用いて両ネットワーク要素回路の設計を行ない，ハードウェア・コスト及び論理遅延時間を求め，ハードウェアによる実現可能クロックレートを推定することにより，両者の性能比較をより現実的な条件の下で行なう．その結果と計算機シミュレーションにより求めた転送性能に基づく性能評価では，適応ルーティングの方が総合的に高い性能を持ち，特に通信に偏りのある場合に有利であるという結果を得た．

Design of Router Chip for Hyper-Crossbar Network Using VHDL

Yoshiki MURAKAMI†　Taisuke BOKU†

Hiroshi NAKAMURA‡　Kisaburo NAKAZAWA§

†Institute of Information Sciences and Electronics, University of Tsukuba

‡Research Center for Advanced Science and Technology, University of Tokyo

§Department of Computer Science, University of Electro-Communications

Abstract

Hyper-Crossbar Network provides very high performance and large flexibility as an interconnection network for massively parallel processors. Up to the present, performance with two routing algorithms for this network; fixed and adaptive routings, have been studied mainly by the computer simulation. In these performance evaluations, network clock rate in both routings were assumed to be the same. For more exact evaluation, however, we should design the actual circuits for them to consider differences in the clock rates.

In this paper, we design router chips for both routing algorithms on Hyper-Crossbar Network, with VHDL description. After the logic synthesis, we can evaluate the number of gates and maximum frequency of system clock, which make the evaluation much more exact.

As a result of comparison, it was shown that the adaptive routing algorithm achieves totally higher performance than the fixed one. Especially, the adaptive routing algorithm shows very high performance in the case of irregular data communication pattern.

1 はじめに

数千台以上の高性能RISCプロセッサをPU (Processing Unit) として，それらをネットワークによって相互結合し，非常に高い性能を持つ超並列計算機を実現しようとする試みが現在盛んに行なわれている．このようなアーキテクチャにおいて，ネットワーク・トポロジとルーティング・アルゴリズムの選択は最も重要な問題の一つである．そのため現在までに様々なネットワーク・トポロジとルーティング・アルゴリズムが提案され，主として計算機シミュレーションによってその有効性が確かめられてきた．しかし，計算機シミュレーションは仮想モデルを対象としてシミュレーションを行うために，実際に回路を設計・評価した際に期待した性能が得られない，またはハードウェア・コストを考慮すると有効とはいい難いといった場合が生じる可能性がある．そのため，真に提案したものの有効性を示すのであれば，ゲート・レベルで実際に設計を行なってから再度評価を行う必要がある．

超並列計算機に採用されているネットワーク・トポロジの1つに，ハイパクロスバ・ネットワーク (以下HXB と略) がある．HXB は，他の典型的な超並列計算機向けネットワークに比べてPU 間距離が非常に小さく，そして通信チャネル数も多いため，大規模なシステムにおける複雑な転送パターンにおいても比較的高い交信性能を保つことができる．特に，転送パターンがランダムな場合，他の典型的な超並列計算機向けネットワークに比べて高いスループットを実現することが可能である [1].

これまでHXB 上でのルーティング・アルゴリズムとして，デッドロックを回避するため wormhole 方式の固定ルーティングが採用されてきた．wormhole 方式は store&forward 方式よりも転送性能に優れ，メッセージの制御が単純である．また，HXB に単純に適応ルーティングを導入すると，デッドロックを生じる．これを防ぐためには複雑な制御が必要となる．本学において稼働中の超並列計算機CP-PACS[2] でも wormhole 方式の固定ルーティングが採用され，現在 HXB 上で wormhole 方式の適応ルーティングを実現した実機は存在しない．しかしながら，HXB 上で wormhole 方式の適応ルーティングを行なった場合，固定ルーティングに比べ転送性能が大きく向上することが文献 [3] 等で研究されている．

そこで本研究では，まだ実際に作られたことのない，HXB 上で wormhole 方式の適応ルーティングを行うネットワーク要素回路を設計し，併せて固定ルーティングを行うネットワーク要素回路も設計して，両者のハードウェア・コストの比較を行う．ルーティング手法としては，文献 [3] で提案されている fix-static と adp-static を対象とする．また，設計したネットワーク要素回路のモデルに基づいた計算機シミュレーションを行ない，同一クロックレートという条件下での転送性能を比較をする．そして設計した回路のハードウェア・コストの評価と計算機シミュレーションの評価を元に，今回設計した固定ルーティングを行うネットワーク要素回路と適応ルーティングを行うネットワーク要素回路の総合的な比較及び評価を行う．

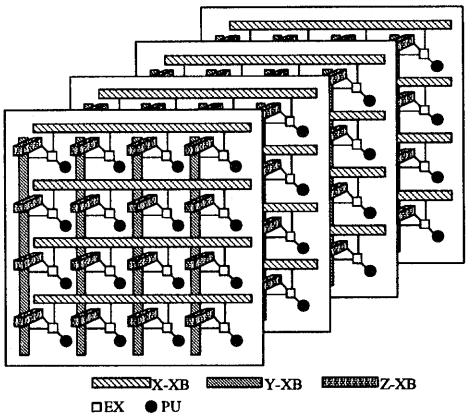

図 1: 3 次元 HXB（4×4×4）

2 ハイパクロスバ・ネットワーク

図 1に 4 × 4 × 4 構成の 3 次元 HXB を示す．図中，PU とクロスバ・スイッチを結合している四角は，wormhole ルーティングを行うためのルータで，エクスチェンジャ (以下 EX と略) と呼ばれる．これは小規模なクロスバ・スイッチで，X，Y，Z 方向のクロスバ・スイッチ (以下 XB と略) と PU の接続，あるいは各 XB 間の接続を行う．n 次元の HXB は，PU を n 次元空間に正方格子上に配置し，1 次元方向に並んだ各 PU を XB により完全結合する．これにより，送受信 PU のアドレスが 1 次元だけ異なる場合は，XB を 1 回通過するだけで相手 PU にメッセージを送信することができる．XB 間のメッセージ転送を EX において自動的に行うことにより，中継 PU を介さない，wormhole ルーティングが実現される．

3 ルーティング手法

本研究で対象とした HXB 上での固定型及び適応型の両ルーティング手法と，本研究における各ルーティング手法の導入について述べる．

固定ルーティング (fix-static) [3]　経路決定は固定型で，virtual channel の使用方法が静的であるルーティング手法である．本研究では，次元オーダーによる固定ルーティングを行う．HXB では，そのトポロジにより次元オーダーによる固定ルーティングを行うことでデッドロック・フリーが保証される [1]. よって，virtual channel の本数は 1 本とする．

適応ルーティング (adp-static) [3]　経路決定は適応型で，virtual channel の使用法が静的であるルーティング手法である．このルーティング手法では，ネットワーク中で選択可能な複数の経路におい

て，virtual channel の使用を制限することで，チャネルの接続にサイクルを生じないようにする．

このルーティング手法の HXB への適応は [4] で提案されている．その概要を以下に示す．

n 次元の HXB において，各 EX 及び XB の入力部に n 組のバッファを用意し，1 本の物理チャネルを n 本の virtual channel として利用する．このとき，ネットワーク中の各メッセージが使用する virtual channel をそのメッセージの転送ステップ数により決定する．n 次元 HXB では最大 n ステップ (n 回の XB の通過) でメッセージの転送が終了するため，n 本の virtual channel を各ステップ数毎で使い分ければ，経路上にサイクルが生じないので，デッドロック・フリーな適応ルーティングが可能となる．

本研究では 3 次元 HXB 用のネットワーク要素回路 (EX, XB) を設計対象とする．従って各物理チャネルで virtual channel は 3 本必要となるので，EX 及び XB 入力部に 3 組のバッファをそれぞれ用意する．また経路決定は EX でのみ行なわれるものとし，迂回は考えない．この場合，出力先を決定するとその先の XB での出力先も一意に決定される．そのため，EX において経路を決定する際，接続している XB の入力バッファの状態だけでなく，その先の EX の入力バッファの状態も考慮して決定しなければならない．この処理のことを「先読み」という．また，先読みの処理を行うだけでは不十分で，「予約」という処理もそれと同時に行なわなければならない．これは，先読みにより経路を決定して XB に到達しても，XB での物理線獲得要求の処理によっては，先読みした EX のバッファを他のメッセージが占有してしまい，結局 XB においてストールしてしまう可能性があるからである．これを防ぐために，先読みすると同時に，先読みした EX のバッファを予約するという処理が必要となる．先読み&予約の処理は，適応ルーティングの特性を活かし転送性能を向上させるのに必要な処理である．以下に本研究での先読み&予約を実現する方法について説明する．

先読み&予約の処理をするにあたって問題となるのは，この処理にかかる時間である．ここでは，先読み&予約する EX が XB を介した先の回路であることが問題となる．よって，単純にこの処理を実現しようとすると，3 回路を介した処理となり，処理に要する時間が許容できるものでなくなる．この問題を解決するため，本研究で設計する回路の各 XB は，常に出力ポートに接続されている EX の入力バッファの状態を監視するものとする．これにより，EX は XB の入力バッファの状態とその先 EX の入力バッファの状態を XB に問い合わせを行うだけで，先読み&予約の処理に要する時間を短縮することができる．

次に，本研究での先読み&予約のアルゴリズムと，文献 [3] で紹介されているものとの違いを説明する．文献 [3] の先読み&予約のアルゴリズムは，1 クロックで 3 次元方向に対して処理が行なわれるものとしている．しかし，現実的にこのような 3 方向同時チェックを行うには，3 方向に先読み&予約要求を出し，複数の予約ができてしまった場合には，1 つのみを残し，他はキャンセルをするという手間が生じる．これを 1 クロックで処理するという仮定はあまりに現実的ではない．また，文献 [3] では，1 次元方向のみを考えた場合でも，3 回路 (EX→XB→EX) を介して行なわれる処理の時間を考慮していない．そのため，1 クロック内で先読み&予約したいバッファの状態を見て，その後に先読み&予約の処理を行うことができるため，予約のキャンセルは必要ない．ところが実際には 2 回路を介した処理のため，1 クロック内で先読み&予約をしたいバッファを見ることはできず，このバッファの状態に関わらず先読み&予約の処理を XB に対して行う．このため予約はしたが XB への出力ポートが獲得できなかった場合が生じ得るため，この場合に予約のキャンセルが必要となる．以上のことより，設計する回路において 3 次元方向に対して先読み&予約の処理を行うと，1 つのメッセージが予約のキャンセルが行なわれるまで，3 次元方向全ての EX のバッファを予約し続けることになり，ネットワーク全体でのメッセージ転送の流れに支障をきたす．そこで，本研究で設計する方式では 3 方向に対する先読み&予約処理を 1 方向ずつ逐次的に行うことにする．

4 設計

ここでは，設計方針と，各ネットワーク要素回路の構成及びメッセージ転送アルゴリズムについて説明する．

メッセージの転送アルゴリズムの説明には，以下に定義する変数を用いている．

- count: バッファにメッセージ・ヘッダが到着した時，メッセージ長を格納する．その後，メッセージ・ボディの各 flit が到着する度にデクリメントする．各バッファに用意される．
- message-length: メッセージ・ヘッダに格納されているメッセージ長の情報．

4.1 設計方針

本研究ではルーティング手法の違いがどの程度ハードウェア・コストに影響を与えるかを見ることを目的としている．従って，ルーティング手法が異なることによる影響のみを見るために，ネットワークの仕様及びルーティング手法の違いに影響されない回路の仕様は共通のものとする．以下に共通する設計方針を示す．

- ネットワーク仕様
- メッセージ・フォーマット
- バッファ構成
- 優先度決定のアルゴリズム
- システム全体は同期

ネットワーク仕様　表1にネットワーク仕様を示す．

ネットワーク・トポロジは，8×8×8 のサイズの 3 次元 HXB とする．次にフロー制御方式は，wormhole 方式とする．最後にチャネル・サイズだが，出力信号ピン数及び高速なデータ転送を考慮し，16bit とする．

表 1: ネットワーク仕様

トポロジ	3 次元 HXB (8×8×8)
ルーティング手法	virtual channel を静的に使用する固定ルーティング及び適応ルーティング
フロー制御方式	wormhole 方式
プロセッサ数	512
チャネル・サイズ	16bit (= 1 flit)
バッファ・サイズ	2 flit

メッセージ・ヘッダのフォーマット　メッセージ・ヘッダにはデスティネーション・アドレス及びメッセージ長の情報が格納されているものとする．図 2 にその具体例を示す．

バッファ構成　バッファ制御を 2 フェーズで行うため，バッファ・サイズは 2 flit とする．

優先度決定のアルゴリズム　本研究で設計した回路では，複数のメッセージ間で，クロスバ・スイッチの出力先の物理チャネルの獲得時と，適応ルーティングにおける EX のバッファの予約時に，各々衝突が生じる．これらのメッセージを調停し，優先度を決定しなければならない．本研究では，送受信 PU アドレスに依存しない公平なメッセージ転送を行うために，原則としてランダムな優先順位決定を行う．また，優先度は入力ポートに対して設定するものとする．同じ入力ポートに属する virtual channel の優先度は virtual channel 番号によって決定され，番号の大きいものが最も優先度が高いとする．これは，デスティネーションに近いものを優先し，経路を早く空けるためである．

システム全体は同期　今回設計するネットワーク要素回路で構成するシステム全体は同期しているものとし，非同期に動作する回路はないとする．

4.2 ネットワーク要素回路の基本構成

ネットワークの要素回路の基本構成となる，固定ルーティングを行う EX を図 3に示す．本研究でのネットワーク要素回路の設計は，この回路を拡張して行うものとする．

設計の基本となる固定ルーティングを行う EX を構成する回路を，以下に説明する．

アービタ回路 (AR)　アービタ回路は各ネットワーク要素回路の出力ポートに用意されていて，バッファ制御回路から送られてくる物理線獲得要求を処理し，同じ出力ポートに接続されているスイッチ回路の制御を行う回路である．

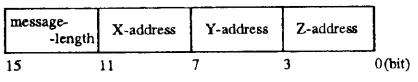

図 2: メッセージ・ヘッダのフォーマット

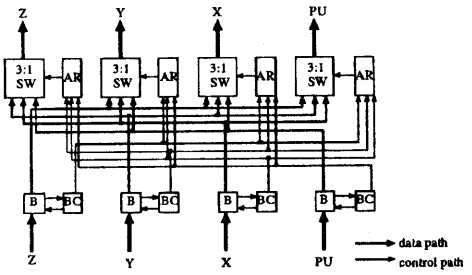

図 3: ネットワーク要素回路の基本構成

要求を処理するアルゴリズムについて説明する．アービタ回路は，その中に各物理線獲得要求の優先度を示す priority という値を常に保持している．アービタ回路はこの priority を元に各バッファ制御回路からの要求を処理し，出力ポートを獲得できたバッファ制御回路に対し ack を返す．その処理と同時に，出力ポートを獲得したバッファからのデータ線を選択するように制御信号をスイッチ回路に対して送る．priority の値は常に一定ではなく，出力ポートを獲得できたバッファからメッセージが全て出力された時に値をシフトし，今まで出力ポートを獲得していたバッファの優先度が最も下がるようにする．これによって各バッファからの要求を疑似ランダムに処理することができ，公平性を保つ．

バッファ制御回路 (BC)　各バッファ毎に配置されている，バッファに入力されたメッセージの経路決定を行う回路である．メッセージ・ヘッダがバッファに入力された時に起動され，メッセージ・ヘッダの情報を基に経路決定のための処理をする．

スイッチ回路 (SW)　n 入力 1 出力のスイッチである．AR からの制御信号を受けて，n 入力のうちの 1 本だけを出力につなぐ (ここでは n=3)．

バッファ回路 (B)　容量が 2 flit あるバッファである．

4.3 固定ルーティングを行う EX, XB 回路の構成とメッセージ転送アルゴリズム

固定ルーティングを行う EX 回路のサイズは 4×4 (4 方向の入出力は PU と X，Y，Z の各次元方向用) とし，XB 回路のサイズは 8×8 とする．

固定ルーティングを行う EX の構成は先に述べた．XB は，EX と比べると，AR，BC 等の各回路の数と SW のサイズが異るだけで新種の回路の追加は必要としない．また，メッセージ転送アルゴリズムもほとんど同じである．EX と XB で異るのは，バッファ制御回路において，EX では EX 固有のアドレスとデスティネーション・アドレスを比較しメッセージの転送先を決定していたのに対し，XB では

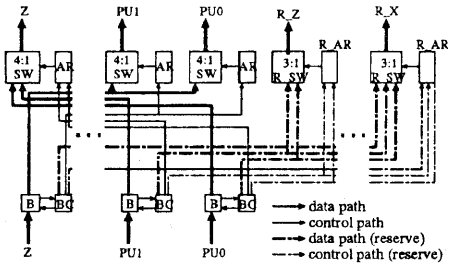

図 4: 適応ルーティングを行う EX の構成

デスティネーション・アドレスとその XB が X, Y, Z 次元のどの XB であるかで転送先を決定する点である．メッセージの転送アルゴリズムを説明する．バッファにメッセージが入力されていない時は BC は待機状態となり，count = 0 となっている．メッセージ・ヘッダがバッファに入力されると BC はメッセージ・ヘッダの情報をデコードし，デスティネーション・アドレスを見て目的の出力ポートに接続している AR に要求を出す．また, count = message-length とする．要求を受けとった AR は各要求に対して設定されている優先度に従って処理を行ない，出力ポートを獲得できたメッセージに対して ack を返す．次に ack を受けとった BC はメッセージの出力状態となり，メッセージのボディの flit が入力されるたびに count の値がデクリメントされる．count の値が正の間はこの状態が保たれ，それが 0 になると再び待機状態になる．

4.4 適応ルーティングを行う EX 回路の構成とメッセージ転送アルゴリズム

この回路の構成を図 4に示す．この適応ルーティングを行う EX のサイズは，EX↔PU 間のデータ・パスのバンド幅を固定ルーティングのものの 2 倍と増強したため，5×5 (5 本の入出力は PU×2 と X, Y, Z の各次元方向用) となる．これは，適応ルーティングを行うことで PU から送出されたメッセージは X, Y, Z の各次元方向に移動できるので，PU は 2 つのメッセージを同時に送出でき，2 倍にしたバンド幅を有効に使用できるからである [4]．固定ルーティングではバンド幅を 2 倍にしてもほとんど効果はないため，バンド幅の増強は行なわない．

文献 [3] のモデルに近付けるため，メッセージを送るデータ・パスと先読み&予約のデータ・パスは別のものとする．そのため，固定ルーティング用 EX に対し以下の回路を追加している．また，BC に先読み&予約を行う機構を追加している．

先読み&予約用アービタ回路 (R_AR) 先読み&予約の要求を処理する回路で，要求処理のアルゴリズムは基本構成の AR と同じである．

先読み&予約用スイッチ回路 (R_SW) 上で述べた先読み&予約用アービタ回路と同様，扱うデータが先読み&予約用のデータとなったものである．

次に，メッセージの転送アルゴリズムについて固定ルーティング用 EX と異る点のみ説明する．メッセージ・ヘッダをデコードすると，最初に BC は先読み&予約の動作に入る．BC はデスティネーション・アドレスを見て転送候補となる出力ポートを知り，そのポートが使用可能かどうかをまず見る．使用可能な場合はそのデータ出力ポートに接続している AR に要求を出す．また，それと同時に，要求を出したデータ出力ポートと同じ次元方向の先読み&予約用の出力ポートに接続している，R_AR にも要求を出す．両アービタ回路で処理が行なわれた後，両方の ack を受けとることができた BC は XB からの先読み&予約 ack の待機状態になる．両方の ack を受けとれなかった場合は，次のクロックで再度先読み&予約の処理を行う．この時，R_AR からの ack が返ってきた場合は，先読み&予約のデータを送り出した XB に対して予約キャンセルの信号を送る．先読み&予約の待機状態になっているバッファは XB からの ack を 2 クロック待つ．これは先読み&予約の処理が 2 回路を介して行なわれるため，処理を行う時間が 2 クロックかかるからである．XB から先読み&予約の ack を受けとった BC は，メッセージの出力状態となる．XB から先読み&予約の ack を受けとれなかった場合は，再度先読み&予約の処理を行う．

4.5 適応ルーティングを行う XB 回路の構成及びメッセージ転送アルゴリズム

この回路のサイズは固定ルーティング用 XB と同様，8×8 である．固定ルーティング用 XB と異る点は，EX から送られてくる先読み&予約用のメッセージを処理するために次の回路が追加されていることである．また，AR にも先読み& 予約を処理する機構が追加されている．

先読み&予約用バッファ制御回路 EX から送られてくる先読み&予約用のメッセージを処理し，AR に予約の要求を出す．

この回路のメッセージ転送アルゴリズムは，固定ルーティングを行うネットワーク要素回路と AR の動作以外は同じなので，AR の動作のみ説明する．

AR は出力先の EX のバッファのうちのどれかが予約されるまで待機状態となる．先読み&予約用のバッファ制御回路から EX のバッファ予約の要求が来ると，各要求毎に処理し，予約できた EX の BC に対して ack を返す．出力先の EX のバッファが予約されると，XB の BC からの出力ポート獲得要求待ちとなる．XB の BC からの要求が来ると，その要求が予約を行なったメッセージからの要求かどうかを判定し，予約を行なったメッセージからの要求だったら BC に ack を返す．予約のキャンセルは ack を返したメッセージが全て出力された時か，EX の BC から予約キャンセルの信号が送られ来た時に行なわれる．予約がキャンセルされると AR は，先読み&予約の要求が来るまで再び待機状態となる．

表 2: 回路の合成結果

回路名	ゲート数	信号ピン数
EX (fix-static)	4022	154
XB (fix-static)	16452	293
EX (adp-static)	17778	232
XB (adp-static)	60508	421

5 評価

評価はハードウェア・コストと最大動作周波数，計算機シミュレーションによる転送性能の 3 つについて行ない，最後にそれらの評価をもとに総合的な評価を行う．

5.1 ハードウェア・コストの評価

ハードウェア・コストを評価するために，設計した回路をハードウェア記述言語を使用して記述し，論理合成ツールを用いて回路合成を各ネットワーク要素回路に対して行う．

ハードウェア記述言語には VHDL[5] を用いる．VHDL を選択した理由は，VHDL の階層構造の記述が，本研究で設計する回路に最も適しているからである．また，この言語が HDL として標準化されつつあることとも理由の 1 つである．

論理合成にはシノプシス社の VHDL 論理合成ツールを用いる．また，セル・ライブラリには design compiler 付属のライブラリを用いる．

本研究における回路合成では，配線及び供給電源等は考慮せず，セルについてのみ考慮した合成を行なっている．これは本研究では特定のテクノロジを意識した設計を行なわず，ルーティング手法の違いによるハードウェア・コスト及び転送性能の比較を目的としているためである．従ってこの後の評価も各セルに設定されている値のみの評価である．

回路の合成結果を表 2に示す．ゲート数は，バッファの数に従って増加している．そのため，適応ルーティングを行う XB は同じ 8×8 の XB であっても，各ポートに用意されているバッファの数が 3 倍の数であるため，ゲート数は最大となっている．また，先読み&予約の処理を行うためアービタ回路が複雑になり，固定ルーティングを行なう XB のアービタ回路の約 4 倍のゲート数になっていることも影響している．適応ルーティングを行うネットワーク要素回路に先読み&予約の処理を行なう機構をつけることにより，EX のバッファ制御回路が固定ルーティング用のそれに比べ 2 倍となり，XB では先読み&予約用バッファ制御回路が 6144 ゲートになるなど，サイズの拡張性に不安が残る．

信号ピン数は，どの回路もチップに収めることのできる範囲に収めている．しかし電源ピンを考慮すると，適応ルーティングを行う XB 回路は危うい数値となっているが，この問題はデータ・パスと先読み&予約用のデータ・パスを同じパスにすることで解決できる．また，サイズの拡張性についても，バイト・スライスなどの手法を導入することで解決できる．

5.2 実行可能最大動作周波数の評価

回路の合成結果より求められるクリティカル・パスの値を元にして，各ネットワーク要素回路の最大動作周波数を求める．この結果を表 3に示す．

適応ルーティングを行う XB 回路の最大動作周波数の値が最も低くなっている．これは，先読み&予約の動作を行うアービタ回路が原因となっている．この回路は他のネットワーク要素回路のアービタ回路に比べて，予約の要求を処理する機構の追加と，処理する要求の数が最も多いことにより複雑となり，このような結果となった．適応ルーティングを行う EX が固定ルーティングを行なうものより遅くなっているのも，バッファ制御回路に先読み&予約の処理をする機構を追加したためである．

上の結果より，各ルーティング手法を用いたネットワーク要素回路でシステムを構築した場合の，各システムの実行可能最大動作周波数を求める．前提としてシステム全体は同期しているものとする．そのため，各ルーティング手法を実現する EX と XB の低い方の値がその値となる．よって，固定ルーティングを行なうシステムの実行可能最大動作周波数は 23.93 MHz で，適応ルーティングを実現するシステムは 23.45 MHz となる．

設計した回路でネットワークを構築した場合，固定ルーティングと適応ルーティングの最大動作周波数の差はあまりない．固定ルーティングは，その機構の単純さ故に機構が複雑な適応ルーティングよりも最大動作周波数が高くなることが予想されたが，それとは異る結果が得られた．この原因は，固定ルーティングを行う XB 回路での，出力ポートを獲得しているバッファに異るメッセージが連続して入力された際の優先度変更処理である．この処理は，ある入力ポートが独占して 1 つの出力ポートを使用することを防ぐために必要となる．固定ルーティングを行う XB 回路では，8 つのバッファに対してこの処理を行なわなければならないので，回路が複雑となる．固定ルーティングを行う EX 回路及び適応ルーティングを行う EX 回路にもこの処理はあるのだが，処理の対象となるバッファの数が 3 なため，回路はさほど複雑になっていない．ここで重要なことは，適応ルーティングを行う XB 回路ではこの処理が必要でないということである．これは，先読み&予約を行なったメッセージのみ XB 回路のバッファに入力されるからである．このため，XB 回路のバッファに異るメッセージが連続して入るという

表 3: 回路の実行可能最大動作周波数

回路名	最大動作周波数 (MHz)
EX (fix-static)	47.52
XB (fix-static)	23.93
EX (adp-static)	25.06
XB (adp-static)	23.45

ことはない．先読み&予約を行う機構は回路の複雑さを増すために，最大動作周波数を評価する上では不利となると予想されていた．しかしながら，このように複雑となる処理が必要となくなる場合もあり，一概には不利であるとはいえない．

5.3 計算機シミュレーションによる評価

シミュレータは，本研究室で開発された汎用ネットワーク・シミュレータ生成系 INSPIRE[6] を用いる．ネットワーク・シミュレータ INSPIRE (Interconnection Network Simulator with Programmable Interaction and Routing for performance Evaluation) は，ネットワークの構成を記述する NDF (Network Description File) と PU の動作を記述する PBF (PU Behavior File) の 2 つのファイルを入力として与えることにより，定義された動作を正確に実現するネットワーク・シミュレータを生成する．

INSPIRE の仕様のため，設計した回路のモデルと計算機シミュレーションに用いた回路のモデルでは以下の 3 つの点が異なる．

1. 物理線獲得要求のアルゴリズム
2. バッファの予約キャンセルのアルゴリズム
3. メッセージの途切れのアルゴリズム

最初に，物理線獲得要求時のアルゴリズムの違いについて述べる．INSPIRE のシステムでは，1 クロックごとにネットワーク中のメッセージを FGFS (First Generate First Service) で処理し，各メッセージがネットワークに与える影響を評価している．そのため，INSPIRE においてメッセージの物理チャネル獲得要求の処理は FGFS で行なわれることになり，今回設計した回路のモデルが疑似ランダムで要求を処理することと異る．次に，バッファ予約キャンセルのアルゴリズムの違いについて述べる．INSPIRE は文献 [3] のモデルと同様，複数の回路を介して行なわれる処理の時間は考慮されていない．そのため，設計した回路のモデルにある予約キャンセルという処理はない．最後に，メッセージの途切れのアルゴリズムの違いについて述べる．設計したモデルでは回路間の信号の伝達に要する時間を考慮し，メッセージの途切れを許している．そのため，メッセージはネットワーク中でみかけの長さが本来のメッセージ長より長くなる可能性があり，1 つのメッセージが経路を獲得している時間が増加する可能性がある．これに対して計算機シミュレーションのモデルでは信号の伝達に要する時間は 0 なので，メッセージは常に連続して存在しこの様な事態は生じない．

本研究では上で述べたモデルの違いの影響は少ないものと仮定し，設計した回路でネットワークを構成したとして評価を行う．評価は一様ランダム転送とメッセージの転送先に偏りが存在する場合を対象として行なった．シミュレーション時間は 500,000 nsec とし，メッセージ長は 10 flit とする．システムの規模は設計したモデルと同じ 512PU とし，3 次元 HXB (8×8×8) を対象とする．また，各システムは実行可能最大動作周波数で求めた周波数で動作しているものとする．

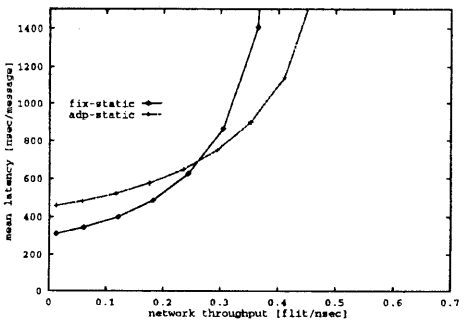

図 5: ランダム転送時の転送性能

まず，ランダム転送について評価する．モデルとして，各 PU の動作を以下のように仮定した．システム中の全 PU は完全に独立して動作し，シミュレータ時間内の各クロックにおいて，ある一定の確率によりメッセージの送信を行う．その際，各 PU はランダムに選んだ相手の PU に対して 10 flit のメッセージを転送する．以上の動作をシステム中の全 PU が繰り返す．

以上の仮定の下で，ネットワークの転送性能を評価する．評価としては，ネットワーク・スループットとメッセージの平均レイテンシを用いる．スループットは，システム中の各 PU が毎時間，1 flit のメッセージを受信できた場合を 1 として，実際に受信できた 1PU あたりの平均総メッセージ受信量をそれに対する比率で表す．また平均レイテンシは，メッセージ転送の際に送信 PU でメッセージが生成されてから受信 PU に到達するまでの時間である．評価結果を図 5に示す．グラフの横軸はネットワーク・スループットで，縦軸がその時のメッセージの平均レイテンシである．

先読み&予約には 3 次元を介してメッセージが転送された場合，4 クロック要する．スループットが 0.26 まではこの先読み&予約による遅延のため adp-static のほうが平均レイテンシが高くなっている．しかしスループットが 0.26 を越えるとこの遅延が隠蔽され，最終的には約 23%, adp-static が fix-static よりもスループットが高くなる．

次に，メッセージの転送先に偏りがある場合について評価する．システム中のメッセージの転送先に偏りが存在するパターンとして，以下に示すようなモデルを仮定する．システム中の全 PU は独立に動作し，メッセージを転送する際，ある一定の確率 (これをホットスポット率とする) でシステム中の特定の PU へメッセージを転送し，それ以外はランダムに選んだ他の PU にメッセージを転送する．評価はネットワーク・スループットを用いた．評価結果を図 6に示す．グラフの横軸はホットスポット率で，縦軸がその時のネットワーク・スループットである．

グラフから分かるように，両ルーティング手法ともホットスポット率が大きくなるに従って，スループットが低下している．しかしながら，adp-static

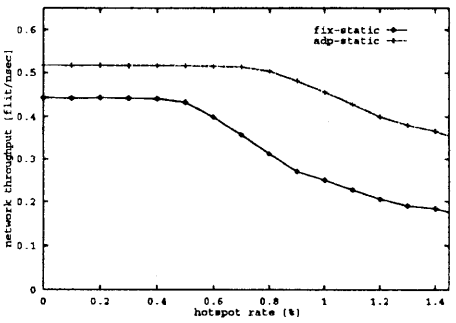

図 6: 転送に偏りがある時の転送性能

はfix-staticよりも低下が緩やかになり，ホットスポット率が0.9%を越えるあたりから約105%転送性能が良くなっている．これは，固定ルーティングではメッセージの転送経路が一意に決まってしまうため，メッセージの転送先に少しでも偏りが生じると転送性能が著しく低下してしまうからである．これに対して適応ルーティングでは空いてるチャネルがあればそちらにメッセージを送るので，メッセージの転送に偏りが生じてもそれほど転送性能が低下しない．

5.4 総合評価

以上3つの評価より設計した回路では，まず適応ルーティングを行うことによりランダム転送時に約23%，メッセージの転送先に偏りのある転送時に約105%転送性能が固定ルーティングよりも向上することがわかった．次に，ゲート数はEXでは約4.4倍，XBでは3.6倍増加することがわかった．しかし，ゲート数は，十分チップに収めることができる値となっている．また，信号ピン数も電源用のピン数を考慮しても，データをバイト・スライスして送出することと，メッセージ転送用のデータ・パスと先読み&予約用のデータ・パスを共通のものにすることによりチップ内に収めることのできる数となっている．以上のことより，設計したネットワーク要素回路においては，ハードウェア・コストを考慮してもHXB上で適応ルーティングを行うことは，特に通信に偏りのある場合に，有効であるといえる．

6 おわりに

本研究では，超並列計算機用ネットワークであるハイパクロスバ・ネットワークにおいて，固定ルーティングを行うネットワーク要素回路と適応ルーティングを行うネットワーク要素回路の設計を行ない，各ルーティング手法のハードウェア・コストの評価・比較を行なった．また，設計した各ルーティング手法のネットワーク要素回路の転送性能を計算機シミュレーションで評価し，ハードウェアコストの評価と合わせて総合的な性能評価を行なった．その結果，適応ルーティングを行うEXのゲート数は固定ルーティングを行うものの約4.4倍，XB回路は3.6倍となったがランダム転送時の転送性能は約23%向上し，メッセージの転送先に偏りのある転送時には約105%転送性能が向上した．そして，適応ルーティングを行う上で必要となるvirtual channel及び先読み&予約機構の追加によるゲート数及び信号ピン数の増加は，1チップに収まる範囲で済ませることができることを確認した．以上のことより，設計したネットワーク要素回路においてHXB上で適応ルーティングを行うことにより，特に通信に偏りのある場合に，高い転送性能を得ることができる．

今後の課題としては，先読み&予約への対処が考えられる．現在先読み&予約には，それ専用のデータ・パスが用意されている．この専用のデータ・パスをメッセージ転送用のデータ・パスと共用し，信号ピン数を減らす予定である．また，先読み&予約のアルゴリズムを変えることも検討している．例えば先読み&予約をせずにランダムに行き先を決める方法が考えられる．さらに今後の課題として，XBのサイズを増やすことが考えられる．適応ルーティングの転送性能を高めるのには，ネットワークのサイズを大規模にすることは有効なことである．しかし，サイズを増やすとデータ・パスのバンド幅が16 bitのままだと信号ピン数が現実に実装できる範囲を越えてしまう．よって，バイト・スライスなどピン数の増加を防ぐ手法の導入が必要となる．

謝辞

本研究に関し貴重な御意見を頂いた、筑波大学西川博昭助教授ならびにアーキテクチャ研究室グループ諸氏に深く感謝します。なお、本研究の一部は創成的基礎研究費 (08P0401) の補助によるものである。

参考文献

[1] 朴 泰祐ほか, “ハイパクロスバ・ネットワークの性能評価”, 信学技報 CPSY93-25, 1993

[2] 中澤喜三郎 他, “CP-PACS のアーキテクチャの概要”, 情報研報 ARC-108-9 , October 1994

[3] 曽根 猛ほか, “ハイパクロスバ・ネットワークにおける virtual channel の動的選択による適応ルーティング”, 情報処理学会論文誌, Vol.37, pp.1409-1418, 1996

[4] 朴 泰祐ほか, “ハイパクロスバ網における適応ルーチングの導入とその評価”, 電子情報通信学会論文誌 D-I, Vol.J78-D-I, No.2, pp.108-117, 1995

[5] R. Lipsett, C. Schaefer and C. Ussery, *VHDL: Hardware Description and Design*, Kluwer Academic Publishers, 1989

[6] 原田 智紀ほか, “並列処理ネットワークのための性能評価用シミュレータ生成系 INSPIRE”, 計算機アーキテクチャ, pp.65-72, 1995