

7L-5

ハードウェア記述言語による相互結合網のモデル化と性能評価

船越 誠司† 久我 守弘‡ 末吉 敏則††

(†九州工業大学 情報工学部)

(‡九州工業大学 マイクロ化総合技術センター)

1 はじめに

マルチプロセッサにおいて、プロセッサ要素間を結合する相互結合網を設計する際には、トポロジー、ルーティング方式、フロー制御方式を決定する必要があり、これらが適切に選ばれたとき、高い並列処理効率が期待できる[1]。

本稿では、上記の3つの要素のうちフロー制御方式に着目し、ハードウェア記述言語を用いて相互結合網のモデル化を行い、store and forward, wormhole 及び virtual cut-through の3種のフロー制御方式の違いが相互結合網の性能にどのような影響を及ぼすかを報告する。

2 相互結合網のモデル化

相互結合網には大きく分けて静的網と動的網があり、モデル化及び相互結合網の性能評価はこの両方について行うことが望ましい。しかしながら、メッシュやトーラスなどの静的網の場合、フロー制御方式の他にデータを転送する際の経路を決定するルーティング方式も考慮しなければならない。つまり、この場合には相互結合網の性能はフロー制御方式とルーティング方式の両者から影響を受けることが予想できる。一方、動的網のうち代表的な多段結合網であるオメガ網は、プロセッサ要素間に複数段の交換スイッチ群を配置し、その間をシャッフル置換で結合したもので、送信側と受信側のプロセッサ要素を決定すると、その間の経路が一意に決まる。したがって、この場合にはフロー制御方式の違いによる性能への影響のみを見ることが出来る。そこで、今回は動的網のうち代表的な多段結合網であるオメガ網についてモデル化を行った。

今回オメガ網をモデル化するにあたり、プロセッサ要素間の結合方式としては密結合型、通信方式としてはパケット交換方式を想定した。モデル化に用いたハードウェア記述言語は、Verilog-HDLと呼ばれるもので、相互結合網の動作を機能レベルで記述できる。さらに、回路の動作タイミングや内部遅延を適度に設定することにより、実際の相互結合網で実行する場合に近いシミュレーション結果を得ることができ、最終的にハードウェア化も行なうことができるという特徴を有する[2]。

オメガ網のモデル化は、交換スイッチ及び入出力インタフェース等の構成要素をVerilog-HDLで記述し、これらを図1のように接続して行った。モデル化を行った交換スイッチは、動作クロックを10MHz、フリットサ

イズ(データ線幅)を16bitとした場合、交換スイッチ当たり20MByte/sec程度のデータ転送能力を持つ。この動作クロックやパケットのフリットサイズ等の値は容易に変更することが可能である。

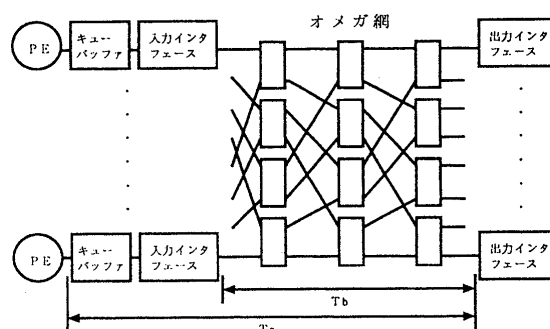


図1 オメガ網のモデル化

3 シミュレーションの概要

ここでは、プロセッサ要素数が8のオメガ網についてシミュレーションを行う。このオメガ網は、フリットサイズについては16bitとし、転送するパケットのサイズについては交換スイッチ内の制御の容易さから128bitの固定長とする。これは、ブロックサイズを64bitとし、それにヘッダや制御情報などを加えたものである。交換スイッチ内のバッファ数については、virtual cut-through と store and forward は1パケット(8フリット)分、wormhole は1フリット分を持つ。

シミュレーションでは、モデル化を行ったオメガ網に対して様々なパターンのパケットを転送し、wormhole, virtual cut-through, store and forward の3つのフロー制御方式について、それぞれ転送遅延の測定結果をもとに性能評価を行う。パケットの宛先については、(a)送信したプロセッサ要素を宛先とするもの(contention free)、(b)ランダムなもの(random)及び(c)全てのパケットが同じ宛先であるもの(hot spot)の3種類を用意した。また、転送するパケット数はプロセッサ要素当たり100パケットとする。このパケットはある一定の間隔でプロセッサ要素において生成され、図1に示す入力インタフェース側のキューバッファにキューイングされる。性能評価は転送遅延の平均値をもとに行う。

測定する転送遅延としては、プロセッサ要素がパケット生成要求を行ってからそのパケットが宛先に届くまでの時間(図1 Ta)と、オメガ網に対して入力インタフェースがパケットの送信開始を行ってからそのパケットが宛先に届くまでの時間(図1 Tb)の2種類を調べた。

4 性能評価

シミュレーションの結果を図2に示す。グラフは、紙面の都合により図1のTbについての測定結果を示している。

図2(a)はcontention freeの場合の結果である。どのフロー制御方式の場合も、パケットの生成間隔が大きい間は転送遅延はある一定値をとるが、短くなってくると、ある点で急に遅延が大きくなる。これは、この上昇点で経路上の交換スイッチ内のバッファが一杯になり、入力でパケットが待たされる状況が起きているためである。したがって、バッファが完全に一杯になった後は、転送遅延は再び一定値をとる。このとき、上昇の前後でwormhole以外の転送遅延が大きく変化するが、これは、経路上のバッファが一杯になったときに、フリットが入力から出力まで通過するのに要するステップ数がwormholeでは3であるのに対し、他の2つでは24となるためであると考えられる。

図2(b)はrandomの場合のグラフを示す。図2(a)と同様にあるパケット生成間隔で上昇するが、その後は一定値をとらない。これは宛先がランダムであるので、交換スイッチ内のトラフィックが一定にならないためである。また、上昇の前についても交換スイッチ内で他のパケットとの競合が起きているので、一定値をとっていない。

図2(c)はhot spotの場合のグラフを示す。このグラフでも他の場合と同じく、あるパケット生成間隔で転送時間が上昇する。しかしながら、他の2つと異なり、store and forwardとvirtual cut-throughがほぼ同じところで上昇する。これは、hot spotの場合にこの2つのフロー制御方式がほぼ同じ性能になることを示している。ここで、上昇した後virtual cut-throughの転送遅延がstore and forwardよりも大きくなっているのは、交換スイッチの実装の差に因るものである。また、上昇する前の転送遅延が図2(a)の場合よりも大きいのは、宛先がすべて同じであるので交換スイッチ内での他のパケットとの競合が早々に起きているためである。

このシミュレーション結果より、wormholeが他の2つに比べて、交換スイッチ内のバッファが一杯になる前後で転送遅延が安定していることがわかる。しかしながら、転送遅延をTaとした場合には、wormholeよりもvirtual cut-throughの方が転送遅延が小さい。これらについては、さらに様々な状況のもとでの比較が必要であると考えられる。

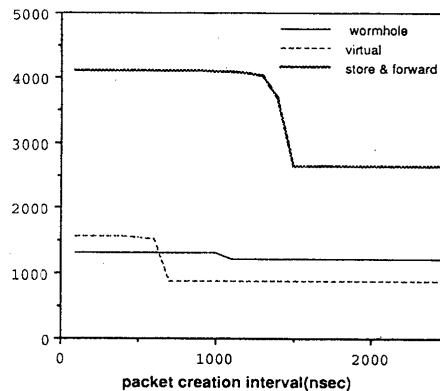
5 おわりに

本稿では、ハードウェア記述言語を用いてオメガ網のモデル化を行い、フロー制御方式の違いによる性能への影響を評価した。今回の評価では、オメガ網の性能評価を得たことに加え、ハードウェア記述言語によるシミュレーションが評価環境としても十分に利用できることが確認できた。今後はより大規模な相互結合網のモデル化や、様々な結合形態のモデル化を行っていく予定である。

参考文献

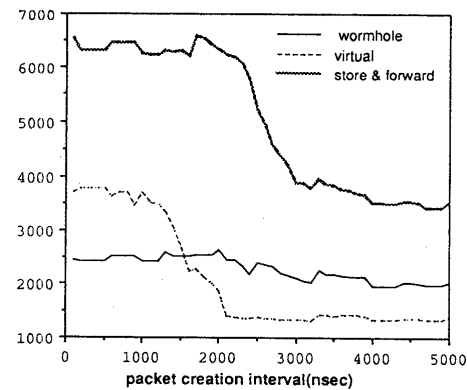
- [1] Bill Dally : *Network and Processor Architectures for Message Driven Computers*, VLSI and PARALLEL COMPUTATION, Edited by Robert Suaya and Graham Birtwistle, pp.140-222(Chapter 3) (1990)
 [2] Cadence Design Systems, Inc. : *Verilog-XL Reference Manual* (1991)

network latency(nsec/packet)



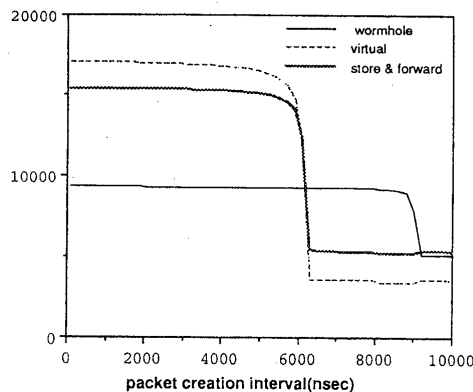
(a) contention free

network latency(nsec/packet)



(b) random

network latency(nsec/packet)



(c) hot spot

図2 シミュレーション結果