A-10 高速通信インタフェースDIMMnet-1の通信バンド幅評価
Paformance evaluation of bandwidth for the DIMMnet-1

濱田芳博† 三橋彰浩† 中條拓伯†
田邊昇†† 工藤知宏†††,☆

YOSHIHIRO HAMADA ,† AKIHIRO MITSUHASHI ,†
HIRONORI NAKAJO ,† NOBORU TANABE †† and TOMOHIRO KUDOH†††,☆

1. はじめに

DIMMnet-1[1] は PC メモリスロットに搭載される NIC であり，従来の PCI バスに搭載される NIC ではなしえない，ホストプロセッサからの直接アクセス (PIO) による低遅延な通信が可能となった．このような通信機構は On-The-Fly(OTF) 通信機構と呼ばれ，1〜8bytes の転送を行う Atomic-On-The-Fly(AOTF) と連続ワード転送を行う Block-On-The-Fly(BOTF) がある．またこれ以外にリモート DMA(RDMA) による通信機構を備えている．

BOTF は CPU キャッシュサイズ程度のデータ量を即時に送受信する場合に適している．並列アプリケーションの最適化においてはキャッシュ上のデータを有効利用し，メモリアクセスを軽減する方法で行われる場合が多い．この時，キャッシュ上の計算結果を送信するために BOTF を用いれば，並列アプリケーションの最適化の可能性が増えると考える．また分散共有メモリのように，データ送受信においてリアルタイム性を要求される場合に効果的であると考える．

RDMA は，並列アプリケーションに適用可能である．特に転送データ量が大きく，CPU 上での他の処理と通信を並行して行う場合に有効であると考える．

本稿においては DIMMnet-1 における連続ワードの通信機構である BOTF と RDMA におけるリモートライトの通信バンド幅について評価を行い，両者の比較を行う．また BOTF の送信にかかるコストについて考察を行う．

2. DIMMnet-1 の概要

2.1 Martini

DIMMnet-1 のコントローラチップとしては，Martini[2] が用いられている．このチップは現在もマイナーチェンジが行われており，本稿に用いたシステムは 2 度目の試作品により構成された NIC を用いている．

2.2 ノード間通信用メモリ領域

ノード間の通信に利用するメモリは DIMMnet-1 上に搭載されており，本稿で使用したシステムでは，128(MBytes) の DRAM(SO-DIMM) と 16(KBytes) の SRAM（低遅延共有メモリ:LLCM）が利用できる．これらのメモリ領域はオペレーティングシステムの管理から外され，DIMMnet-1 を利用するユーザプロセスに割り当てられる．

2.3 通信機構

Martini は通信機構として RDMA を備えている．この通信機構を利用するためのプリミティブとして PUSH，PULL が組み込まれている．PUSH はローカルノードのメモリブロックをリモートノードのメモリブロックへ転送し（リモートライト），PULL はリモートノードのメモリブロックをローカルノードのメモリブロックへ転送する（リモートリード）．

† 東京農工大学
Tokyo University of Agliculture and Technology
†† （株）東芝 研究開発センター
TOSHIBA Corporate Research & Development Center
††† 新情報処理開発機構
Real World Computing Partnership
☆ 現在，独立行政法人 産業技術研究所 グリッド研究センター
Presently with National Institute of Advance Industrial Science and Technology

また主に DIMMnet-1 向けの通信機構として，1〜8(bytes) までのデータをホストプロセッサが PIO により書き込むことでリモートリード／ライトが行える AOTF や，474(bytes) 以下まで書き込むことでリモートリード／ライトが行える BOTF がある．AOTF においてはパケットヘッダを事前に Martini へ登録しておき，送信時に Martini 内部で送信データに付加してパケット送出を行う．これに対し BOTF においてはユーザプロセスがパケット全体を NIC へ書き込む必要がある．

3. 通信バンド幅評価

RDMA と BOTF によるリモートライトついて，通信バンド幅の評価を行う．使用する DIMMnet-1(2nd) においては，Martini の動作周波数が内部で 2 分周されており，システム中においてこの部分がボトルネックとなる．本稿で用いた DIMMnet-1 はメモリクロック 100MHz 上で動作するため，Martini 内部は 50MHz で動作する．これより通信バンド幅は 400(MB/s) で制限される．

3.1 評価環境

評価環境を表 1 に示す

表 1 測定環境

CPU/FSB	PentiumIII 850MHz/100MHz
マザーボード	D6VAA
OS	Linux(Kernel 2.4.2)
NIC	DIMMnet-1(Martini2nd)
スイッチ	RHiNET2/SW2[3]

3.2 RDMA バンド幅

3.2.1 測定方法

RDMA 発行用の送信 Window はキャッシュ属性を Uncached にし，データ送受信領域には SO-DIMM を用いた．送信側において，送信完了の ACK 受信先には LLCM を用い，RDMA 発行後 ACK を受信するまでの時間を測定することでバンド幅を求めた．

3.2.2 測定結果

RDMA によるバンド幅の測定結果を図 1(1) に示す．図 1 中 (1)(A) はスイッチを介した 2 台での測定であり，(B) はスイッチを介した 1 台での測定である．(A) より RDMA による継続バンド幅において 330(MB/s) が得られていることが判った．しかし (B) のように一台の NIC 内部で送信と受信が並行して動作した場合，SO-DIMM が送信／受信 DMA の競合資源となり，ピーク時の 40%の継続バンド幅低下が発生することが判った．

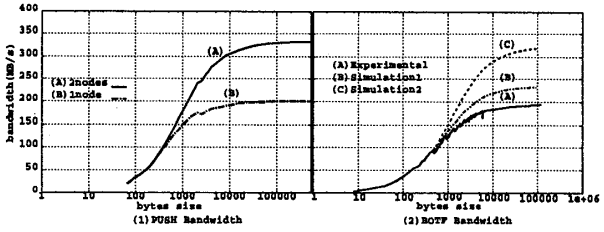

図 1 DIMMnet-1 通信バンド幅

3.3 BOTF バンド幅

3.3.1 測定方法

BOTF によるデータ送信は，送信データ量を 464(bytes) 毎に分割し 2 枚の送信 window により交互にパケットを発生させることにより行う．ここで，送信 Window のキャッシュ属性は WriteCombining にし，データ受信領域には LLCM を用いた．また送信時のデータは CPU キャッシュ上に乗っているものとしている．送受信にかかる時間はデータを送信し始めてから受信領域に全てのデータが受信されるまでとし，この時間を測定することでバンド幅を算出した．

3.3.2 測定結果

スイッチを介した 2 台による BOTF バンド幅の測定結果を図 1(2)(A) へ示す．1 台の測定でも RDMA のように継続バンド幅が低下することはなかった．しかし，BOTF の継続バンド幅は 190(MB/s) となり，RDMA に対し 40%低下していた．この原因については次章で述べる．

4. BOTF における継続バンド幅低下の原因

4.1 BOTF 処理時間

928(bytes) のデータを 464(bytes) の 2 つのパケットに分割して送信する場合の BOTF 送受信処理の経過時間を図 **2** へ示し，ここで使用している記号について**表 2** へ示す．図 2 よりホストからの書きこみ (HostWrite) と，送信側 NIC 内部における wconti は最初の一回分以降は並行して行われており，受信側 NIC 内部における receive と reccont 及び DMA はパケット受信毎に順序だてて実行されていることが判る．つまり BOTF によるデータ送受信に要する時間 (BOTFtime) は式 1 で表されることになる．ここで式 1 より算出したバンド幅を図 1(2)(B) へ示す．

式 1 より算出したバンド幅に対し，図 1(2)(A) の実測値はバンド幅が継続する部分で 50(MB/s) 程度低下している．これは継続分の低下であるから式 1 中の Period 項へ式 2 のように不明分 (Unknown) を加えることで補正でき，その値は 20clocks となる．Unknown はハードウェアの論理設計では現れない値であり，ネットワークと NIC の間に存在する周波数吸収用バッファの応答時間が不定期に変化するために発生していると考える．

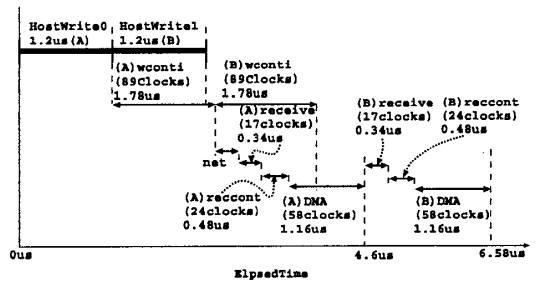

図 2 BOTF 送信受信処理

$$BOTF_{time} = Overhead + Period$$
$$Overhead = HostWrite + wconti + net$$
$$Period = (receive + reccont + DMA) \times packets \quad (1)$$
$$Period = (receive + reccont + DMA + Unknown) \times packets \quad (2)$$

表 2 記号説明

記号	説明	
HostWrite	ホストから NIC 送信領域への一つのパケットの書きこみ	
送信側 NIC 内部処理		
wconti	パケット解析，送信領域から SendFIFO へのパケットの転写 パケット送信開始処理	
net	ネットワークの通過遅延	
受信側 NIC 内部処理		
receive	パケット解析，仮想アドレスの取得，reccont の呼び出し	
reccont	rec1	物理アドレス取得，DMA 要求
	rec2	DMA 開始までの応答／待機
DMA	受信領域へのデータ転写	
Unknown	不明分	

4.2 BOTF 処理時間内訳

式 1，2 より算出した BOTF 送受信にかかる処理時間内訳を図 **3** に示す．図 3(A) において，継続バンド幅が期待できる 190544(bytes) を送受信する場合の処理時間では，30%程度が受信側における receive，reccont に費されている，また周期的に発生する Unknown の処理時間も 20%となり，DMA に費された時間は全体の 50%程度であることがわかる．これより，ボトルネックとなるバンド幅 400(MB/s) から 50%低下していることが理解できる．

4.3 継続バンド幅の改善

BOTF 送受信において，受信側 NIC における receive と reccont (rec1) までを先行するパケットの DMA と並行実行可能とし，Unknown の発生を抑制できた場合の BOTF 継続バンド幅について考察する．この時の BOTFtime は式 3 のようになる．これより算出したバンド幅を図 1(2)(C) へ示す．さらに前節同様 190544(bytes) 時の BOTF 処理時間内訳を図 3(B) に示す．これらより，継続バンド幅は 320(MB/s) に到達し，受信側 DMA の動作時間も処理時間全体の 80%となった．

$$BOTF_{time} = Overhead + Period$$
$$Overhead = HostWrite + wconti + net + receive + rec1$$
$$Period = (rec2 + DMA) \times packets \quad (3)$$

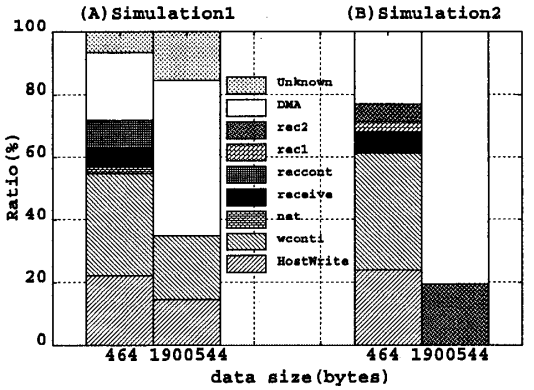

図 3 BOTF 処理時間内訳

4.4 継続バンド幅低下の原因

BOTF は送信データサイズを 464(bytes) 程度に分割して送信するため，RDMA に対しパケットサイズが小さくなる．このためパケット数が増加し，受信側における DMA 起動までの時間や Unknown 時間が累積して継続バンド幅を低下させている．

5. 結　論

BOTF では，一つの NIC で送信と受信が同時に発生するような場合，RDMA と異なりバンド幅低下を起さず安定した通信が可能である．しかし継続バンド幅において RDMA の 320(MB/s) やボトルネックとなるバンド幅 400(MB/s) に対し大きく低下し，190(MB/s) であった．これは BOTF のパケット長が小さいため，パケット受信毎の NIC 受信側における DMA 起動までの処理と Unknown が累積し，全送受信時間の 50%を占有するためである．

現在の Martini を改善して BOTF の継続バンド幅を向上させるためには，パケット受信から DMA 要求までの時間を先行するパケットの DMA に隠蔽し，Unknown として発生している不要な処理時間を抑制する必要がある．

参考文献

1) 田邊 他，“メモリスロットに搭載されるネットワークインタフェース MEMnet” 情報処理学会計算機アーキテクチャ研究会，Vol99，No67，pp.73-78(1999)
2) 山本 他 高速性と柔軟性を併せ持つネットワークインタフェース用チップ:Martini 情報処理学会研究報告 2000-ARC-140, pp,19-24(2000)
3) 西，他 LASN 用 8Gbps/port 8x8 One-chip スイッチ:RHiNET-2/SW，JSPP2000 pp173-180，(May 2000)