計算機アーキテクチャ 150−11
（２００２．１１．２８）

DIMMnet-1 における Martini オンチッププロセッサによる通信の性能評価

田邊 昇[†1] 濱田芳博[†2] 三橋彰浩[†2]
山本淳二[†3] 今城英樹[†4] 中條拓伯[†2]
工藤知宏[†5,☆] 天野英晴[†6]

我々は DIMM スロット搭載型ネットワークインタフェース DIMMnet-1 を開発した．そのコントローラである Martini は，リモートメモリアクセスをハードだけで実現できるが，メッセージ交換のハードサポートは今回は実装できていない．しかし，Martini は TLB 等のメインテナンスを意図したオンチッププロセッサを有するため，この上のファームウェアによりメモリーベーストプロセッサ的な動作をさせることも可能である．本報告ではリモート間接書込みを用いたメッセージ交換について言及し，DIMMnet-1 プロトタイプの実機上で Martini のオンチッププロセッサを用いたメッセージ交換やバリア同期の性能評価結果を示す．その上で，受信側の強化に関する今後の DIMMnet の改良方針について考察する．

Performance Evaluation of Communication with Martini's On-Chip-Processor on DIMMnet-1

Noboru Tanabe,[†1] Yoshihiro Hamada,[†2] Akihiro Mitsuhashi,[†2]
Junji Yamamoto,[†3] Hideki Imashiro,[†4] Hironori Nakajo,[†2]
Tomohiro Kudoh[†5,☆] and Hideharu Amano [†6]

DIMMnet-1 prototype network interface plugged into a DIMM slot is developed. The controller LSI named Martini can realize remote write operation and remote read operation only by hardware. However it has no hardware support for message passing in this implementation. Message passing can be emulated like memory based processor by the firmware of on-chip-processor for TLB maintenance. In this report, message passing with remote indirect write operation is presented. Performance of message passing with remote indirect write operation and barrier synchronization emulated by on-chip-processor is evaluated. We considered how to enhance the receiver side of DIMMnet based on the evaluation.

1. はじめに

近年，高性能 PC を多数用いて並列処理を行なういわゆる PC クラスタが注目されている．高性能な PC クラスタ用に Myrinet[☆☆1)]，PCI-SCI[2)]，MEMORY CHANNEL2[3)],QsNET[4)],InfiniBand[5)] 等の高速ネットワークインタフェース (NIC) が各種開発されており，これらはいずれも PCI バスに接続される．

†1 (株) 東芝, 研究開発センター
Corporate Research and Developpment Center, Toshiba
†2 東京農工大学
Tokyo University of Agriculture and Technology
†3 (株) 日立製作所
Hitachi Ltd.
†4 (株) 日立インフォメーションテクノロジー
Hitachi Information Technology
†5 新情報処理開発機構
Real World Computing Partnership
☆ 現在，産業技術総合研究所
Presently with National Institute of Advanced Industrial Science and Technology
†6 慶應義塾大学
Keio University
☆☆ Myrinet は Myricom Inc. の商標です．本書に記載の商品の名称は，それぞれ各社が商標および登録商標として使用している場合があります．

全てをコモディティ部品で構築するシステムよりも十分優れた性能を，Moore の法則に追随した形で継続的に提供しつつ，価格性能比を最大にする PC クラスタを構築するために，我々は，従来のように PCI バス等の入出力バスではなく，メモリスロット (DIMM スロット) に搭載されるタイプの NIC である DIMMnet-1[10) 11)] を開発した．

この DIMMnet-1 や，同一の Martini LSI[12)] を用いた PCI 版 NIC である RHiNET2/NI[12)] には，AOTF[10)] および BOTF[11)] というプロテクションを確保しつつ低遅延な通信を実現する通信機構が搭載されている．これらは，1990 年頃に東芝で開発された高並列計算機 Prodigy[13)] の S-BUS 版ホストインタフェースに適用されている 2 ポートメモリへの書き込みをベースにした低遅延高バンド幅通信技術 [14)] や，RWCP 超並列東芝研究室で設計された超並列計算機 TS/1 の分散共有メモリアクセス機構である CTLB という通信制御情報の再利用機構 [16)] を，PC クラスタ用 NIC 向けに改良を施したものである．

低遅延通信を実現する他のアプローチとしては，1992 年頃から超並列計算機 JUMP-1 の通信機構として提唱された MBP[21)] は，多機能なメモリーベースト通信を実現することが特徴とされている．この「CPU の MMU を介したメモリアクセスにより通信を起動することで低遅延通信とプロテクション維持を両立する方式」は，Prodigy の S-BUS 版ホストインタフェースにおいて JUMP-1 に先立って実現され，その

流れを汲む DIMMnet-1 の AOTF や BOTF にも，そのメモリーベースト通信の特徴は受け継がれた．実際に試作された JUMP-1[22] の通信機構自体は，枯れた世代の ASIC による MBP-light という，近年の PC 上の CPU と比較すると桁違いに低速な周波数で動作するプロセッサによるソフト処理を経由するために，十分な高速化は得られていない．

一方，DIMMnet-1 ではメモリーベースト通信という MBP と共通のアプローチを取りつつも，高周波動作するホスト CPU からオフロードする機能を十分に絞り，送信側 CPU から受信側 CPU に至る経路全体に渡り，肝心な部分のみのハード化の徹底を進めた．具体的には，送信側としては AOTF や BOTF，受信側については AOTF で生成した極めて細粒度 (データサイズ 1 から 8 バイト) のパケットを受信するための Mini-OTF 受信部をハードで実装した．受信側についてはリモート間接書込みを用いたメッセージ交換やメッセージのパッキング・アンパッキングをサポートする TS/1 と同様なハードウェアの実装が計画 [6] されていたものの，RHiNET と共通の LSI として実現されるという設計上のリソースの制約から，今回の Martini LSI には実装はされず，受信側については基本的には従来の RDMA の枠組みを逸脱しない RHiNET 向けに設計された保守的な受信回路を流用する実装が行われた．

DIMMnet-1 においてはこれまでの性能評価から，AOTF によるリモート書込みによる低遅延通信は比較的優れた性能を実現できている．しかし，BOTF によるパケットは全て RHiNET 向けに設計された受信回路に回されるために，低遅延な細粒度通信と高バンド幅の両立という目標は，プロトタイプの実機上では十分には達成できているとはいえない．とりわけ，BOTF を用いたメッセージ交換に関しては何らハード的なケアがなされていないので，課題が残されている．Martini は TLB 等のメインテナンスを意図したオンチッププロセッサを有するため，これを用いることでメモリーベーストプロセッサ的な動作をさせ，メッセージ交換の機能をエミュレーションすることが可能である．これでメッセージ交換を実装しようとすると，リモート書込みよりもさらに受信側オーバーヘッドの問題点が顕在化すると思われる．

リモートライトやリモートリードといったハードで提供された one sided 通信の高い性能から，MPI-2 やコンパイラの助けによる並列アプリケーションの高速化は，現状の DIMMnet-1 でも効率的に実装できる可能性はあると思われる．しかし，one sided 通信の API が存在しない MPI-1 ベースのプログラムが大半であるという現状を鑑みるに，DIMMnet-1 の後継ではメッセージ交換を効率的に実装できることが望ましい．

メッセージ交換の効率的実装を目指す研究としては，並列計算機 AP1000+の PUT・GET 機能を用いた MPI の実装である MPIAP[18]，並列計算機 EM-X 上の MPI 実装である MPI-EMX[19]，汎用 EWS 上にソフト的に実装されたリモート書込みをベースにした MPI 実装である MPI/MBCF[20] などがあるが，リモートアドレスは直接指定のみとなっている．リモート間接書込みを用いたメッセージ交換は並列計算機 TS/1 で実現されることが予定されていたが，LSI 開発の途中でプロジェクトが中断になったため，現時点では実際に実装した例は無い．その後，Cenju-4[17] にはそれと似たリモートのレジスタを指定して間接アドレスに書き込む機構が実装された．DIMMnet-1 には当初 TS/1 流のリモート間接書込みを用いたメッセージ交換の実装が計画されていたが，RHiNET と共通の LSI に実装されるという制約から実装が見送られた．

今後の DIMMnet の受信側の改良として，オンチッププロセッサによるエミュレーション性能の向上という，Intel の IXP2800[24] に代表される近年のいくつかのフルカスタムなネットワークプロセッサでとられているアプローチ [25] で行なうべきか，何らかの機能を追加ハードで実現すべきかについては，当研究チーム内でも現時点では明確な統一見解に至っておらず，どうしていくべきかを決定していかねばならない．そのためには，現状の DIMMnet-1 のオンチッププロセッサによるエミュレーションを通して，いくつかのソフトウェア的実装方法で実際にどの程度の性能が得られるのか，どこに問題が残っており，いかなる機能をどのようにハードやオンチッププロセッサやホストで分担していくことが望ましいのか，を把握していくことが必要となる．

本報告では，第 2 章，第 3 章で DIMMnet-1 のコントローラである Martini チップのオンチッププロセッサと，リモート間接書込み方式について紹介し，リモート間接書込み機能の一部を DIMMnet-1 のコントローラである Martini チップのオンチッププロセッサによってエミュレーションする方式と，それを応用したメッセージ交換の実現法を示す．第 4 章では DIMMnet-1 プロトタイプの実機上で測定されたエミュレーションによるメッセージ交換やバリア同期の性能を評価し，今後の DIMMnet の受信側の改良方針について考察する．第 5 章で関連研究について述べ，第 6 章でまとめる．

2. Martini チップのオンチッププロセッサ

DIMMnet-1 と RHiNET2/NI のコントローラである Martini チップには R3000 互換のオンチッププロセッサが内蔵されている．Martini のハードコア部と同じ周波数で動作するので，現状の DIMMnet-1 プロトタイプでは 50〜66MHz で動作するものである．

このプロセッサは内部バスを介して TLB を含むオンチップメモリのみならず，一部のステートマシンの状態レジスタの書き換えも可能な構造を有する．TLB ミスの他，ハード的に未実装な type を有するパケットを受信することで起動される割り込みハンドラによって，未実装な新機能やバグを有する回路ブロックの代替エミュレーションが可能である．

しかし，このプロセッサの能力的には，ホストに用いられる MPU とは周波数で 40〜50 倍程度遅いことに加え，内部並列度や分岐予測器等のアーキテクチャ的要素がもたらす性能面でも数倍劣る．さらに Martini チップ内は 64bit 幅の資源がほとんどであるにも関わらず，32bit のプロセッサであるために内部資源へのアクセスも余計にサイクル数を消費する．つまりこのプロセッサは性能的には低い位置に限界点があり，使用に際しては TLB ミスの回復等の例外的な処理向け，あるいはハードの致命的バグを機能面で回避したり，ハードで実現できなかった新機能の動作検証用という位置づけのものであることを認識する必要がある．

このことは，ハードで処理される部分や高速な MPU をベースに処理されるソフト部分との住み分け上のバランスを欠いた ASIC 内のソフトマクロ型 CPU によるファーム処理を多用したシステムにおける性能上の限界性の一例を実機上で再現することが可能である，ということを意味する．

3. リモート間接書込み

3.1 リモート間接書込みの概要

リモート間接書込みの基本コンセプト [15][16][6] は筆者らによって 1993 年から提唱されているが，本論文では「送信側がポインタと書込みデータを含むパケットを送信することで，そのポインタで指し示される受信側にあるアドレス情報に基づき遠隔書込みが行われる通信方式」と定義する．**図 1** にリモート間接書込みのコンセプトを示す．

また通常，リモート書込みと呼ばれている「送信側がポインタと書込みデータを含むパケットを送信することで，そのポインタで指し示される受信領域に遠隔書込みが行われる通信方式」を本論文では「リモート直接書込み」と定義する．

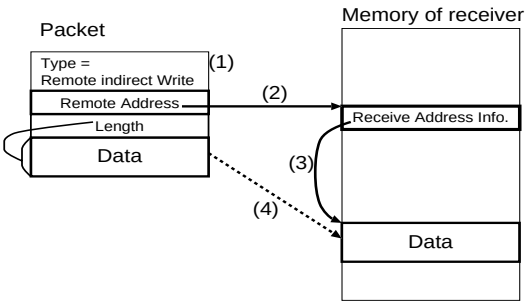

図 1 リモート間接書込み
Fig. 1 Remote Indirect Write

リモート間接書込み方式の実装は，RWC 超並列東芝研究室が設計していた超並列計算機 TS/1 におけるハードウェアのみによる実装 (実装途上でプロジェクトが中断になったため LSI は未完成に終わった) と，DIMMnet-1 上に今回作成したオンチッププロセッサ上でのソフトウェアを併用した実装の二つがある．この他に類似したものとして，Cenju-4[17] における受信側のレジスタを指定する通信方式がある．

3.2 TS/1 における実装

TS/1 におけるリモート間接書込み機構は，送信側から受信コマンドチェインの先頭アドレスと書込みデータのペアを送り，受信側では受信側のメモリ上に配置されたチェイニングされた複数の受信コマンドに従って受信した書込みデータを書き込む機構であった．ここで，受信コマンドとはリードライト種別と先頭アドレスとデータ長と next ポインタの組である．

この機構では，送信したデータが受信側で実際には書き込まれる場所を送信側が知らなくても，受信側が指定した適切な場所に書き込むことができる．受信コマンドがチェイニング可能なので, かなりの複雑なアンパックをソフトウェア処理なしで全てをハードウェアで行えるように設計されていた．

3.3 Cenju-4 における実装

Cenju-4[17] にも Remote DMA with address register プロトコルと呼ばれるリモート間接書込み機構に類似したハードウェアが実装されている．ただしこちらはリモートアドレスを指定するのではなく，リモートにおける 3 本のレジスタ番号 (0,1,2) のいずれかを指定するものである．また message プロトコルという機構も実装されていて，上記と同様に 2 つのポインタ (0,1) のいずれかを選択する．

3.4 DIMMnet-1 における実装

3.4.1 リモート間接書込み

DIMMnet-1 ではそのコントローラ LSI である Martini 上に，RHiNET のために用いられる機能も実装しなければならないという設計リソース上の厳しい制約から，TS/1 と同様なリモート間接書込みのための専用ハードウェアを実装することができなかった．

DIMMnet-1 上に現時点までに実装されたリモート間接書込みでは，上記の TS/1 におけるリモート間接書込みとは若干異なり，複雑なアンパックを行う機能はないものの，低速なオンチッププロセッサ上で動作するソフトウェアを併用した類似した手法で，TS/1 におけるリモート間接書込みが実現する一部の機能を実現している．なお，性能を別とすれば，原理的にはオンチッププロセッサによるエミュレーションにより TS/1 と同様な機能を実現することは可能と考えられる．

TS/1 では受信側でメモリ上に受信コマンド (先頭アドレスとデータ長と間隔と next ポインタ) を設定し, メモリ上の先頭コマンドのアドレスをパケットが運び, 全てが専用ハードウェアで高速に処理される．

これに対し，DIMMnet-1 ではメモリ上には受信コマンドではなく受信アドレスしか設定せず，パケットはメモリ上のコマンドのアドレスは運ばず，type から一意に決定される命令列 (低速なオンチッププロセッサ上で実行されるソフトウェア) を用いて簡素な機能だけ実現したという点が異なる．

DIMMnet-1 上に今回実装されたリモート間接書込みは，以下のような手順で処理される．

(1)DIMMnet-1 ではハードウェアで実装されていない type フィールドを持つパケットを受信した際に，オンチッププロセッサに割り込みがかかるので，リモート間接書込みには一つの未使用 type を割り当て，この type 値を持った ACK 付きリモート書込み (PUSH) パケットを AOTF または BOTF または RDMA を用いて送信する．

(2) 上記パケットを受信すると受信側で未使用 type 割り込みがかかり，割り込みハンドラの中でまず type をチェックして，リモート間接書込み type に対応した命令列にジャンプする．

(3) その命令列の中で, パケットのリモートアドレスフィールドで示されるアドレス (ポインタへのポインタ) で指定される領域 (受信領域へのポインタ) を読む．

(4) 受信領域へのポインタ値でパケットヘッダのリモートアドレスフィールドを書き換え，中断していた以降のハードウェアによる受信処理を継続させ，ハンドラを抜ける．

3.4.2 メッセージ交換

DIMMnet-1 上に今回実装されたリモート間接書込みを用いたメッセージ交換は，以下のような手順で処理される．

(1)DIMMnet-1 ではハードウェアで実装されていない type フィールドを持つパケットを受信した際に，オンチッププロセッサに割り込みがかかるので，リモート間接書込みを用いたメッセージ受信には一つの未使用 type を割り当て，この type 値を持った ACK 付きリモート書込み (PUSH) パケットを AOTF または BOTF または RDMA を用いて送信する．

(2) 上記パケットを受信すると受信側で未使用 type 割り込みがかかり，割り込みハンドラの中でまず type をチェックして，リモート間接書込みによるメッセージ受信 type に対応した命令列にジャンプする．

(3) その命令列の中で, パケットのリモートアドレスフィールドで示されるアドレス (ポインタへのポインタ) で指定される領域 (受信領域へのポインタ) を読む．

(4) 受信領域へのポインタがシステムバッファの溢れを示す値だった場合は，パケットの送信元に NACK パケットを送信し，受信パケットを廃棄して，ハンドラを抜ける．送信側は NACK パケットを受けると再送を行う．

(5) 受信領域へのポインタがシステムバッファに対応しないアドレスだった場合は，次の受信に備えて受信領域へのポインタが書いてあった場所を対応するシステムバッファの空き領域の先頭アドレスで書き換える．

(6) 受信領域へのポインタでパケットのリモートアドレスフィールドを書き換え，中断していた以降のハードウェアによる受信処理を継続させ，ハンドラを抜ける．

(7) ハードウェアによる受信領域への書込みが完了すると，パケットの送信元にハードウェアによる ACK が返送される．送信側は ACK パケットを受け取ることで送信の成功を知り，再送用データを破棄する．

なお，上記のメッセージ交換の基本部の上に，MPI を実装するためには，送信プロセス (RANK) 毎にメモリ上に設けられた複数のシステムバッファに対応する部分や，TAG の異なるメッセージの退避に対応する部分を追加実装する必要がある．しかし，それらは MPI/MBCF でも既に行われていることである．

3.4.3 細粒度通信とエミュレーションの不整合

上記のメッセージ交換をオンチッププロセッサによるエミュレーションで実現するという方式には，細粒度通信における

限界が予想されている．DIMMnet-1 には BOTF という細粒度通信と高バンド幅を両立させる送信機構が実装されているが，この方式ではパケットサイズが Window メモリというオンチップメモリのサイズによって限定されるため，あまり長いデータを運ぶパケットを生成できない．このため，パケットを受信するたびに割り込み処理のソフトウェアで処理をする上記のエミュレーション法では BOTF により高いバンド幅を実現することが困難になると考えられる．

4. 性能評価

本章では，Martini のオンチッププロセッサを用いた通信の例として，メッセージ交換とバリア同期を取り上げる．

4.1 測定環境

以下の実験において用いた測定環境を**表 1** に示す．

表 1 測定環境
Table 1 Experimental environment

基板種別	光版
Martini バージョン	第 3 版
ハードコア部周波数	50MHz
Link モード	RHiNET2
Link 周波数	250MHz
Link バンド幅	500MB/s
スイッチ	光版 RHiNET2/SW[26]
ホスト CPU	Pentium3
CPU コア周波数	850MHz
FSB 周波数	100MHz
DIMM 周波数	100MHz
主記憶	256MB
チップセット	VIA Apollo Pro133A
OS	Linux kernel 2.4.2
Compiler	egcs-2.91.66

4.2 リモート間接書込みによるメッセージ交換

オンチッププロセッサを用いたリモート間接書込みによるメッセージ交換のバンド幅の測定結果を**図 2 図 3** に示す．比較のために BOTF，RDMA によるハード式のリモート直接書き込みのバンド幅も併記している．

バンド幅は送信開始から，受信側 Martini から送信側に Ack が返ってくるまでの時間を測定した．つまり測定したバンド幅はブロッキング型の Send の実行時間を元にしたバンド幅ということになる．

本実験では BOTF で送信しているが，その際，2 つの window メモリを交互に使用して, データ長 464bytes のパケットに分割して送信している．RDMA の場合はこれ以上の長さのデータでもパケットは分割されない．

オンチッププロセッサを用いた通信の測定では，受信側での割り込みハンドラ内で停止中のハードに対し，リモートアドレス上 (今回の実験では LLCM 上) のポインタ値によるアドレス書き換えを行う．図 2 中の MP by soft indirect という条件では，リモート間接書込みを用いたメッセージ交換の測定値であり，指示される領域への DMA の起動，フロー制御，パケットに分割された全転送サイズを受け取った際の受信側への ACK のための LLCM への書込み，送信側への ACK パケットの返信を全てソフトウェアで行っている．シーケンシャル番号をパケットに付けるか否かを制御する f_seq フラグが on の場合と off の場合の両方を測定した．残りの 3 つの測定はリモート間接書き込みのバンド幅であり，RVATLB miss-continue という条件では RVATLB のミスを，PATLB miss-continue という条件では PATLB のミスを割込みへのトリガとしており，指示される領域への DMA をハンドラ内で書き換えられた後のアドレスへのリモートライトの継続という形で実現した．PATLB miss-continue(non-intr) という条件では割込みではなく割込み条件成立をオンチッププロセッサによるポーリングで実装したものである．

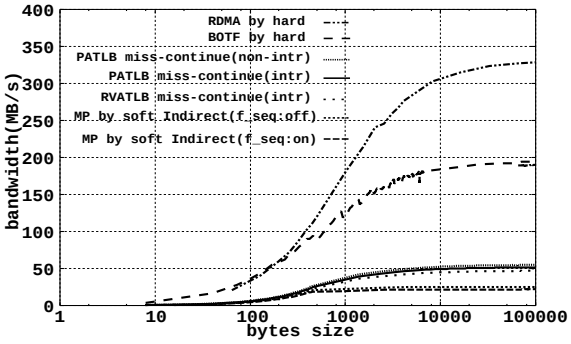

図 2 DIMMnet-1 におけるバンド幅
Fig. 2 Bandwidth on DIMMnet-1

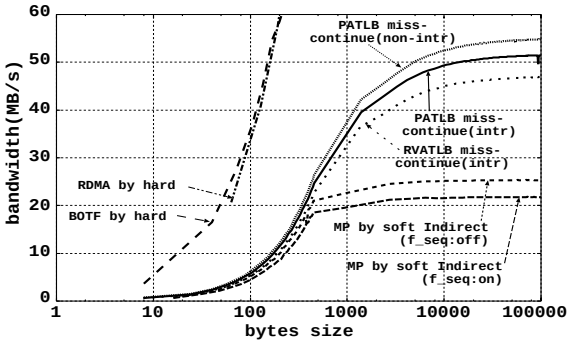

図 3 DIMMnet-1 におけるバンド幅 (拡大図)
Fig. 3 Zoomed figure of bandwidth on DIMMnet-1

4.3 オンチッププロセッサによるバリア同期

オンチッププロセッサを全く使わずに簡素な支援ハードウェア (Mini-OTFR,LLCM) とホスト上のソフトウェアを組み合わせて高速なバリア同期 [27] が既に実装されている．ここでは，Martini のオンチッププロセッサのハンドラを使って計数型のバリア同期を実装し，性能を比較する．計数型のバリア同期の処理を以下に示す．

(1) DIMMnet-1 ではハードウェアで実装されていない type フィールドを持つパケットを受信した際に，オンチッププロセッサに割り込みがかかるので，計数型のバリア同期に一つの未使用 type を割り当て，この type 値を持ったリモート書込み (PUSH) パケットをホストから AOTF を用いて送信する．

(2) 上記パケットを受信すると受信側で未使用 type 割り込みがかかり，割り込みハンドラの中でまず type をチェックして，計数型バリア同期 type に対応した命令列にジャンプする．

(3) Martini のハードコア部の状態を IDLE 状態に書き換える．

(4) ヘッダー内のリモートアドレスに対応するオンチップ SRAM 上にあるカウンタをデクリメントし，カウンタがゼロでない場合，割り込み終了．

(5) 同期完了通知用パケットを生成，送信する．(送信先へのパケットは SRAM 上に前もって書き込んであり，この値を同期完了通知用パケットとしている．スイッチのマルチキャスト機能使用時は 1 回の同期完了通知パケットの生成，送信で済み，マルチキャスト未使用時はこれを rank 番号の小さい

方から同期完了通知パケットの生成，送信を rank 数分行っている）

(6) カウンタの初期化後，割り込み終了

なお，時間計測は番号が一番大きい rank がホストから AOTF 送信を行ってから LLCM をポーリングして同期完了通知が帰ってくるまでの時間を計測した．比較のために SCore 上の Myrinet による MPI のバリア同期関数での実測値，Mini-OTFR と LLCM とホスト上のソフトウェアを組み合わせた実装 (スイッチのマルチキャスト機能使用時および未使用時) での実測値 [27] を併せて**図 4** に示す．

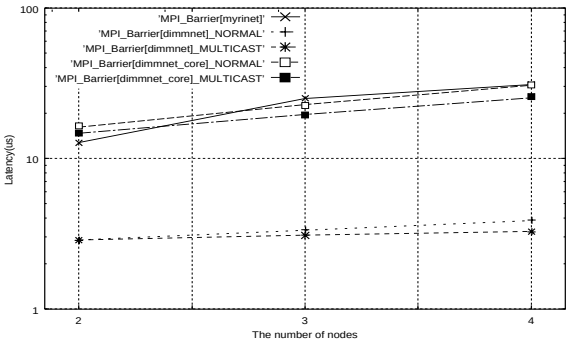

図 4 DIMMnet-1 におけるバリア同期遅延
Fig. 4 Barrier Synchronization Latency on DIMMnet-1

4.4 考　　察

BOTF によって送り出す場合は，パケットサイズが Window メモリという Martini 内部のメモリのサイズの限界からデータ長 464bytes のパケットに分割せざるを得ない．このため今回の実装のように，メッセージ交換時に受信側でパケット毎に割り込みがかかる方式では，割り込み処理がボトルネックとなり，メッセージサイズを大きくしても全てオンチッププロセッサのソフトで処理した場合で 22MB/s からバンド幅が上がらない．

割り込み処理の後半をできるだけハードウェアで実行できるように継続実行にさせた場合は，割り込み処理のオーバーヘッドが若干減ることによる実行バンド幅が 52MB/s と 2 倍以上に向上した．これは受信側のハード化が劇的にバンド幅に効くことを予感させる結果といえる．500B 程度のメッセージ長に対する Myrinet2000[1] が 133MHz のプロセッサを用いて 50MB/s 程度のバンド幅であるので，50MHz で動作している Martini のオンチッププロセッサの能力で律速されがちな使用状況の割には概ね良好といえる．しかし、いずれにしてもハード式のリモート直接書き込みのバンド幅のような，DIMMnet-1 が本来持っている高バンド幅を生かしきれていないと言わざるをえない．

一方，RDMA を送信側で用いた場合は，BOTF よりも送信側でのオーバーヘッドが大きい分メッセージ長が短い時はバンド幅が低いものの，メッセージ長が長い場合はパケットサイズを大きくすることができるので，BOTF の場合と異なりバンド幅が向上すると考えられる．ただし，長いメッセージにおけるバンド幅の向上よりも，短いメッセージ長におけるバンド幅の向上の方が技術的に困難であり，RDMA で処理することはその課題の本質的な解決にはなっていない．

以上のような観点から，ASIC で作成されるセミカスタムなオンチッププロセッサを用いている限り，受信側でのオンチッププロセッサによるエミュレーションでは細粒度通信時のメッセージ交換のバンド幅維持は難しく，受信側に PC クラスタ向けの NIC としての何らかのハード支援を導入していくことが必要かつ望ましいと思われる．

バリア同期においてはバンド幅ではなく遅延時間が重要になるが，この場合は後半の実験からも簡素なハード支援と高速なホスト CPU 上でのソフト処理の協調を行うことで一桁の性能向上が得られることが明らかになった．ここでも，ASIC 上での非力なプロセッサを用いるよりも，簡素なハード支援とホスト処理を組み合わせた方が高い性能を得られることが示されたといえる．

5. 関 連 研 究

リモート直接書込みを用いた MPI の実装としては MPIAP[18]，MPI-EMX[19]，MPI/MBCF[20] などがある．

MPIAP[18] は並列計算機 AP1000+のリモート直接書込み (put) およびリモート読出し (get) を用いて実装された MPI である．送信側がプロトコルメッセージを put で送り，受信側ではそれを受けて get を行う．パケットが 1.5 往復することになり遅延時間が大きくなる．

MPI-EMX[19] は並列計算機 EM-X のリモート直接書込みを用いて実装された MPI であり，送信側と受信側の両方で Send と Receive の間の TAG 等のマッチングを行うことと，受信側からのリモート読み出しを用いることが特徴である．先行する Receive の実行により送信側に受信要求が届き，これとマッチングが取れる send を後から実行した場合はリモート直接書込みで処理される．ただし，必ず送信側でのマッチング処理が入り，遅延時間が大きくなる．Receive の実行が遅れた場合は MPIAP 同様に受信側からのリモート読出しとなるのでパケット 1.5 往復するために遅延時間が大きくなる．さらに，同時に Send と Receive が実行された場合は一貫性を取るための処理が必要になる．また，Receive で RANK に対してワイルドカードが指定された場合はリモート直接書込みによる高速化の恩恵が得られない．

MPI/MBCF[20] は汎用の NIC 上に MBCF というソフト的に構築されたリモート直接書込み用いて実装された MPI であり，MPI-EMX と同様に先行する Receive の実行により送信側に受信要求が届き，これとマッチングが取れる send を後から実行した場合はリモート直接書込みで処理される．ただし，必ず送信側でのマッチング処理が入り，遅延時間が大きくなる．送信側にも制御メッセージ受信用のメモリ上の FIFO を RANK 毎に設ける必要がある．Receive の実行が遅れた場合は RANK 毎に設けられたメモリ上の FIFO へのリモート直接書込みを行なう．さらに，同時に Send と Receive が実行された場合は一貫性を取るための処理が必要になる．

リモート間接書込みの実装としては DIMMnet-1 の前身である並列計算機 TS/1[15][16] があり，こちらは全てハードで実装される計画になっていたが，LSI 化までは至っていない．この他に Cenju-4 の通信機構も類似した方式を実現している．ただし並列計算機 Cenju-4 の通信機構はリモートアドレスではなくリモートにある少数のレジスタ番号を指定する点で異なる．レジスタ番号の指定では例えば RANK ごとに受信場所を分類しておくような効率的実装はできない．

6. ま と め

DIMMnet-1 のコントローラである Martini チップのオンチッププロセッサと，リモート間接書込み方式について紹介し，リモート間接書込み機能の一部を DIMMnet-1 のコントローラである Martini チップのオンチッププロセッサによってエミュレーションする方式と，それを応用したメッセージ交換の実現法を示した．DIMMnet-1 プロトタイプの実機上で測定されたエミュレーションによるメッセージ交換やバリア同期の性能を評価した．今後の DIMMnet の受信側の改良方針としては，受信側に PC クラスタ向けの NIC としての何ら

かのハード支援を導入していくことが必要かつ望ましいと思われる．

今後は，アプリケーションによる評価を中心に，第三版の Martini LSI を用いた DIMMnet-1 の実機上での評価と，ソフトウェア環境の整備を進める予定である．DIMMnet-1 における高速な細粒度通信性能を活かすと思われる shasta[28] のようにある程度通信の粒度を細かいところに最適化したコンパイラの開発が望まれており，開発の検討が進められている．さらに，今回の実験を通して得られた知見をはじめ，さらなるハードウェアの問題点や改良すべき点を洗い出し，ハードウェアに関する改良も加えられる予定である．

謝辞

本研究は新情報処理開発機構が推進してきた RWC (Real World Computing) プロジェクトの並列分散コンピューティング技術研究の一環として行われたものである．本研究のさらなる発展のための継続をご支援いただける決定をいただいた総務省戦略的情報通信研究開発制度の関係者の皆さまに感謝いたします．(株) 日立製作所の西氏，慶應義塾大学の渡辺氏，元・慶應義塾大学の土屋氏，元・東京農工大学の須田氏，(株) 日立 IT の上嶋氏，金野氏，寺川氏，慶光院氏，岩田氏，山本氏，柏原氏，大杉氏をはじめ Martini LSI および DIMMnet-1 の開発に携わった全ての方々に感謝いたします．

参 考 文 献

1) Myricom Corp. http://www.myri.com/
2) Dolphin Corp. : PCI-SCI Adapter Card D320/D321 Functional Overview, Part no.:D1950-10299(1999.11)
3) Fillo and Gillett : Architecture and Implementation of MEMORY CHANNEL 2 , *Digital Technical Journal*, Vol.9(1) (1997)
4) F. Petrini, W. Feng, A. Hoisie, S. Coll, E. Frachtenberg “The Quadrics Network : High-Performance Clustering Technology”, *IEEE Micro*, pp.46-57 (2002)
5) InfiniBand Trade Association, http://www.infinibandta.org/
6) 田邊, 山本, 工藤 : 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， Vol. 99, No. 67, pp. 73-78, (1999.8)
7) Tanabe, Yamamoto, Nishi, Kudoh, Hamada, Nakajo, Amano : MEMOnet : Network interface plugged into a memory slot, *IEEE International Conference on Cluster Computing (CLUSTER2000)*, pp.17-26 (2000.11)
8) Tanabe, Yamamoto, Nishi, Kudoh, Hamada, Nakajo, Amano : On-the-fly Sending : A Low Latency High Bandwidth Message Transfer Mechanism, *5th International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN2000)*, pp.186-193 (2000.12)
9) Tanabe, Yamamoto, Nishi, Kudoh, Hamada, Nakajo, Amano : Low Latency High Bandwidth Message Transfer Mechanisms for a network interface plugged into a memory slot, *Cluster Computing Journal*, Vol.5, No.1, pp.7-17 (Jan. 2002)
10) 田邊, 濱田, 山本, 今城, 中條, 工藤, 天野 : DIMM スロット搭載型ネットワークインタフェース DIMMnet-1 とその低遅延通信機構 AOTF, 情報処理学会論文誌ハイパフォーマンスコンピューティングシステム, Vol.43, No.SIG(HPS7) (掲載予定 Dec. 2002)
11) 田邊, 山本, 濱田, 中條, 工藤, 天野 : DIMM スロット搭載型ネットワークインタフェース DIMMnet-1 とその高バンド幅通信機構 BOTF, 情報処理学会論文誌, Vol.43, No.4, pp.866-878 (Apr. 2002)
12) 山本, 渡邊, 土屋, 原田, 今城, 寺川, 西, 田邊, 上嶋, 工藤, 天野 ”高性能計算をサポートするネットワークインタフェース用コントローラチップ Martini”, 情報処理学会論文誌ハイパフォーマンスコンピューティングシステム, Vol.43, No.SIG6(HPS5), pp.122-133 (Sep. 2002)
13) 田邊, 中村, 鈴岡, 小柳 : 並列 AI マシン Prodigy の試作と通信性能評価”, 電子情報通信学会論文誌， Vol.J74-D-I,No.4,pp.264-272 (1991.4)
14) 田邊 : マルチプロセッサシステム，公開特許公報，特願平 2-157491(出願 1990.6)，特開平 4-48371 (公開 1992.2)
15) 田邊, 鈴木, 菅野 : 並列処理装置，公開特許公報，特願平 5-52718(出願 1993.3)，特開平 6-266678 (公開 1994.9)
16) 鈴木，田邊，菅野，小柳 : 超並列 Teraflops マシン TS1 〜分散共有メモリアーキテクチャ〜，情報処理学会第 48 回全国大会，4B-4 (1994)
17) 加納，中村, 広瀬, 細見, 中田：並列コンピュータ Cenju-4 のユーザレベルメッセージ通信機構, 並列処理シンポジウム’99(JSPP’99)， pp. 7-14 (1999.6)
18) D. Sitsky, K. Hayashi : Implementing MPI for Fujitsu AP1000/AP1000+ using Polling, interrupts and Remote Copying, 並列処理シンポジウム’96(JSPP’96), pp. 177-184 (1996)
19) 建部, 児玉, 関口, 山口 : メモリ書き込みを用いた MPI の効率的実装，情報処理学会論文誌，Vol.40, No.5，pp.2246-2255 (1999.5)
20) 森本, 松本, 平木 : メモリベース通信を用いた高速 MPI の実装と評価，情報処理学会論文誌，Vol.40, No.5，pp.2256-2268 (1999.5)
21) 松本，平木 : 超並列計算機上の共有メモリアーキテクチャ，電子情報通信学会コンピュータシステム研究会 CPSY92-26，pp.47-55 (1992)
22) 五島，斎藤，小西，秤谷，森，富田，並列計算機 JUMP-1 の分散共有メモリ・システム，情報処理学会論文誌，No.SIG8(HPS 2)，pp.15-27 (2000.11)
23) 天野，山本，渡邊，土屋，金子，工藤：クラスタコンピュータ用ネットワークインタフェースチップ Martini における代行処理機構，電子情報通信学会技術報告 CPSY2001-54, pp.23-30 (2001.9)
24) Intel Corp. : Intel IXP2800 Network Processor - For OC192/10 Gbps network edge and core applications, ftp://download.intel.co.jp/design/network/prodbrf /27905402.pdf (2002)
25) Intel Corp. : Next Generation Network Processor Technologies - Enabling Cost Effective Solutions for 2.5Gbps to 40Gbps Network Services, ftp://download.intel.co.jp/design/network/papers /27905001.pdf (2001.10)
26) 西, 多昌, 西村, 山本, 工藤, 天野：LASN 用 8Gbps/port 8x8 One-chip スイッチ: RHiNET-2/SW, 2000 年記念並列処理シンポジウム (JSPP2000)， pp. 173-180 (2000.5)
27) 田邊，濱田，三橋，山本，今城，中條，工藤，天野 : DIMMnet-1 プロトタイプによるバンド幅と大域演算性能の評価, 情報処理学会 ARC 研究会， Vol.2002, No.81, pp.97-102 (2002.8)
28) D. J. Scales, K. Gharachorloo, and C. A. Thekkath : Shasta: A Low Overhead, Software-Only Approach for Supporting Fine-Grain Shared Memory, *ASPLOS’96* (1996.10)