

## プログラム制御キャッシュレベルメモリの性能 - ハードウェア -

中瀨 光昭, 前田 敦司, 岡本 秀輔, 曾和 将容

電気通信大学 情報システム学研究所

e-mail: {nakasumi,maeda,okam,sowa}@sowa.is.uec.ac.jp

4 N-7

### 1 はじめに

プログラム制御キャッシュレベルメモリ [1] では、データ転送ユニット (DU) が、データ転送命令の実行と、キャッシュレベルメモリ (CLM) と分散共有メモリ (DUDM) 間のデータの移動を行なうため、データ転送ユニット (DU) のスループットがこのシステムの性能を決定する。そのため、これら2つの処理を可能な限り並行に処理できる設計が必要となる。本文では、プログラム制御キャッシュレベルメモリでの通信オーバーヘッドを隠蔽するためのアドレス演算と通信のオーバーラップを可能にする設計の概要を示す。

### 2 設計の特徴

通信では、メッセージのフォーマッティング、ネットワークへのメッセージ送信、メッセージバッファアクセス及びメッセージの CLM へのコピー、メッセージ受信待ちの処理が必要である。これらは、メッセージフォーマッティングと通信の初期設定など常に一定時間を要するものとメッセージ長に比例する時間を要するものに分けられる。前者は、メッセージフォーマッティングや初期設定をハードウェア化することで削減できる。後者については、本質的に必要な時間であるので、その削減はネットワーク性能の向上などに依らなければならないが、それには限界がある。本方式では、データ転送命令とアドレス計算命令を順次に並べてデータ転送プログラムを構成するが、アドレス計算等の命令実行と通信とをオーバーラップさせることにより性能が向上すると考えられる。メッセージをネットワークに送出する際、送信完了までの間に、オーバーラップする処理が送信に関係する資源にアクセスしなければ、これは可能である。同様に、メッセージを受信する際、メッセージ受信領域がすぐアクセスされない限り、メッセージヘッダの到着直後に DU は後続する命令を実行することができる。この時、メッセージが

DUDM へ転送されている間は、演算と受信がオーバーラップすることになる。今回の設計では、メッセージバッファ管理、メッセージコピーを許すことで、DU がアドレス計算等の命令実行と通信をオーバーラップすることが期待できる。又、リングバッファによりメッセージの常時受信を可能にしている。すなわち送信側の DU をブロックすることがないため、送信側の再送を少なくすることが期待できる。

### 3 通信とアドレス計算のオーバーラップ

- 送信とアドレス計算のオーバーラップ: 送信とアドレス計算のオーバーラップでは、DU は、メッセージ送信命令を発行した直後、バッファにメッセージを書き込むことができれば、以降の命令実行を開始することができる。DU がデータ転送命令を実行するとオペランドのアドレスがローカルメモリを指すかリモートメモリを指すかがチェックされ、リモートメモリへのアクセスであれば、DU は CCTLR に通信コマンド (CMD) とデータを送り、後続のアドレス計算を実行する。同時に CCTLR はネットワークにメッセージを転送する。バッファはネットワークのバンド幅に合わせて区切られており、メッセージは受信側の CCTLR に完全に受信されるまで保持される。
- 受信とアドレス計算のオーバーラップ: メッセージの受信要求が、メッセージの到着よりも遅い時、一旦メッセージをメッセージバッファに転送し、その後、メッセージの受信要求が発行された時、メッセージバッファからユーザー領域へのコピーと DU の命令実行とをオーバーラップさせることができる。より積極的な方法として、DU はメッセージ受信要求を発行した後、メッセージの到着を待たないで、命令実行を開始する方法がある。この方法を採用すると同期処理が複雑になるため、今回は前者のみを適用することを考える。受信とアドレス計算のオーバーラップでは、通信のオーバーヘッドを完全に隠蔽する可能性がある。CCTLR

The Performance Impact of the design of Program Controlled Cache Level Memory on Multiprocessor  
Mitsuaki Nakasumi, Atsusi Maeda, Shusuke Okamoto and Masahiro Sowa  
Graduate School of Information Systems, The University of Electro-Communications

はメッセージのヘッダが到着するまで待つが、メッセージ全体を待つ必要がない。メッセージの受信要求がメッセージの到着よりも早い場合、メッセージは指定された CLM のメモリセルに直接転送される。もしメッセージが到着既に到着している場合、バッファから指定された CLM のメモリセルにコピーされる。DU がデータ未着の領域にアクセスした時は、到着するまでアクセスを繰り返す。

#### 4 ユニット構成と動作

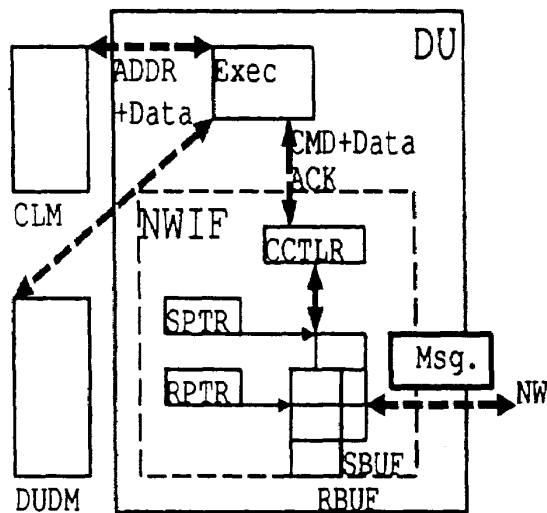


図 1: ユニット構成

まず、データ転送ユニット DU に含まれるネットワークインターフェース (NWIF) の構成について述べる。NWIF はメモリコントローラと通信コントローラを持つ。図 1 に NWIF の構成を示す。図中点線はデータ線を表す。CLM はキャッシュレベルメモリ、DUDM は分散共有メモリの一部である。実行ユニット (Exec) はデータ転送に関する命令を実行するためのユニットである。SBUF は、メッセージの送信、RBUF は、メッセージの受信が完了するまでメッセージを保管するリングバッファである。NWIF は、リモートデータ転送を行なうユニットであり、送信バッファ SBUF、受信バッファ RBUF と通信コントローラ (CCTLR) で構成される。CCTLR は Exec からコマンドとメッセージを送受して Exec と非同期に通信処理を進める。送信ポイント (SPTR)、受信ポイント (RPTR) はそれぞれリングバッファのポイントである。SPTR は送信したデータが書き込まれる位置、RPTR は受信したデータが書き込まれる位置を保持している。DU 起動時に SPTR は、

SBUF の先頭アドレスを、RPTR は RBUF の先頭アドレスをを指す様に初期化される。

次に NWIF の動作を示す。

1) 受信時: ネットワーク (NW) からメッセージ (Msg.) が到着すると、受信側の CCTLR は RBUF がオーバーランしないことを確かめる。もし SPTR と RPTR が同じ値である時、Exec を停止する。RBUF が書き込み可能であれば、RPTR を更新し RBUF にデータを取り込む。取り込みが完了すると、送信側の CCTLR に受信完了を Msg. として返送する。受信側の CCTLR は、既にこの Msg. に対する要求があるかをチェックする。あればデータ到着を Exec に通知するとともにデータを Exec に送信する。Exec は CLM へデータを書き込む。Exec から書き込み完了のコマンドを受け取ると SPTR を更新し RBUF の使用済領域を解放する。

2) 送信時: Exec は CLM から予めデータを取得し、送信コマンド (CMD) とこのデータを CCTLR に送る。CCTLR は、SPTR を更新し、SBUF にこのコマンドとデータを書き込む。ネットワークインターフェースは、ネットワークが使用可能であれば、コマンドとデータを Msg. として送出する。受信側の CCTLR から受信完了の Msg. を受け取ると SPTR を更新し SBUF の使用済領域を解放する。

#### 5 まとめ

本文では、プログラム制御キャッシュレベルメモリでの通信オーバーヘッドを隠蔽するため、アドレス演算等の命令実行と通信のオーバーラップを許す設計の概要を示した。しかしこの設計では、バッファへのコピーのオーバーヘッドがあること、各メモリの同期機構を単純化するため、Exec のみがメモリにアクセスすることによるコストが高いと思われる。これに関しては、今後シミュレーションを通じて、これらがオーバーヘッドにどの程度影響するかを評価した後、ハードウェア実現性も踏まえて、適用の可否を決定したいと考えている。

#### 参考文献

- [1] Nakasumi, et al.: "Program Controlled Cache Memory on Parallel Computer", PDPTA'97 Int'l Conf., Vol.III, pp.1423-1428 (1997).