SR2001における高速プロセッサ間通信機能

2H－2

薗田浩二*、西門隆*、岩嵜正明*、宇都宮直樹*、森山健三**

*（株）日立製作所　システム開発研究所　**（株）日立製作所　ソフトウェア開発本部

1. はじめに

超並列計算機において高速な演算性能、プロセッサ数に比例した性能スケーラビリティを達成するためには、高速なプロセッサ間通信機能が不可欠である。超並列計算機SR2001[1][2][3]では、高速なプロセッサ間通信機能としてメモリコピーを排除した直接メモリ転送機能を提供する。本稿では、直接メモリ転送機能の概要とアプリケーションインタフェース(API)について述べる。

2. 汎用的なプロセッサ間通信機能の欠点

超並列計算機のネットワークハードウェアの性能を極限まで引き出すという観点から検討すると、汎用的なプロセッサ間通信機能は以下の様な問題点をもつ。

(1) プロトコル処理によるレイテンシの増加

プロトコル処理やプロトコルレイヤを渡るオーバヘッドによって、送信時のネットワーク起動及び受信プロセスの再起動までの遅れが生じ、通信レイテンシが増加する。

(2) メモリコピーによる通信スループットの低下

送受信プロセス間での非同期な通信を可能とするために、バッファリングが必要となる。このため、送信側・受信側のそれぞれで必ずメモリコピーが発生する。さらにこれらのメモリコピーと通信は逐次的に実行されるため、通信スループットはメモリコピーの性能と回数で抑さえられる。

3. 直接メモリ転送機能

3.1 直接メモリ転送機能の概要

２章での問題点を解決するプロセッサ間通信機能としてSR2001では直接メモリ転送機能を提供する。直接メモリ転送機能は、図1の様に送受信側とも連続物理メモリ領域をユーザプロセスの仮想アドレス空間に固定的に割り付けておき、送信側が受信側のデータ格納領域を指定して直接データを書き込む、送信者主導のプロセッサ間メモリコピーである。

送信側プロセスの仮想空間上のデータを、直接受信側プロセスの仮想空間に書き込むことにより、送受信側のそれぞれのプロセッサ上ではメモリコピーが発生しない。また、ハードウェアによるプロセッサ間メモリコピーであるため、プロトコル処理が不必要であり低レイテンシ・高スループットのプロセッサ間通信機能を提供できる。

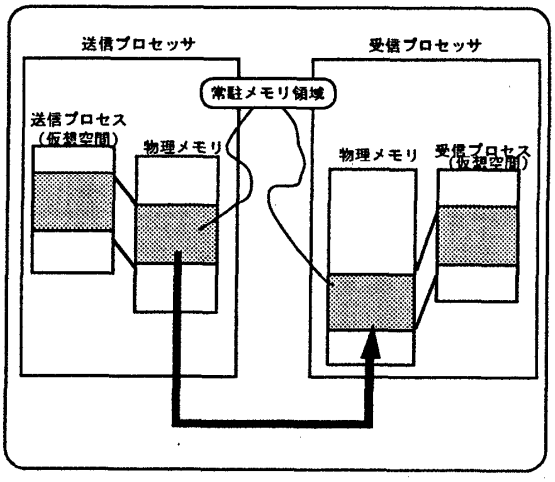

図1. 直接メモリ転送機能

3.2 直接メモリ転送機能のAPI

3.2.1 通信領域の確保

直接メモリ転送機能を行うプロセスは、通信に先だって連続物理メモリ領域を確保する。この領域にはプロセッサ内で一意な通信領域IDを定義し、次節で述べるコネクション設定時の通信先の指定に使用する。この領域はプロセッサ内の任意のプロセス間で共有可能であり、必要なプロセスが自プロセス空間にマッピングし、通信領域として使用する。

また図２に示すように通信領域を更に細分化し、複数のデータ受信領域を定義することができる（通信フィールド）。この通信フィールドには、同じ通信領域IDを持つ通信領域内で一意な、通信フィールド番号をユーザが割り当て、通信領域ID同様、コネクション設定時の通信先領域指定に使用する。

High performance inter processor communication facility on SR2001
Kouji Sonoda, Takashi Nishikado, Masaaki Iwasaki, Naoki Utsunomiya, Kenzou Moriyama
Systems Development Laboratory and Software Development Center, Hitachi, Ltd.

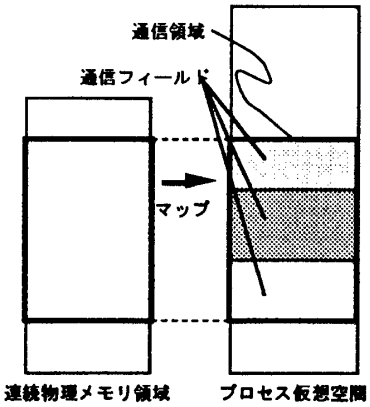

図2. 通信フィールド

3.2.2 コネクションの設定

直接メモリ転送機能では、データ通信時の処理を軽くするため、通信に先だって送信側から受信側へコネクションを設定するインタフェースとする。

このコネクションの設定は、受信側が定義した通信領域IDと通信フィールド番号を用いて、送信側が（プロセッサ番号、通信領域ID、通信フィールド番号）により送信先を指定することで行う。

3.2.3 送信インタフェース

通信のレイテンシを小さくするためには、ユーザアプリケーションが送信関数を呼び出してから、実際にネットワークが起動されるまでのソフトウェアのオーバヘッドを小さくする必要がある。

送信インタフェースは、次の２種類を提供する。

(1) 動的コマンド作成インタフェース
送信関数が呼ばれる度にネットワークハードウェアへのコマンドを作成するインタフェース。

(2) コマンド再利用インタフェース
一度作成したネットワークハードウェアへのコマンドを繰り返し再利用するインタフェース。

コマンド再利用インタフェースを使用すると、あらかじめ作成しておいたコマンドをネットワークハードウェアに与えるだけで送信処理が可能となり、高速なネットワーク起動が実現できる。

3.2.4 受信確認インタフェース

直接メモリ転送機能では、受信データは直接受信プロセスの通信フィールド上に書き込まれる。このため、ユーザプロセスはデータの受信処理を行う必要はなく、データ受信が完了したことを確認するだけでよい。受信確認のインタフェースとして、ブロッキングインタフェースとスピンチェックインタフェースの２種類を提供する。

(1) ブロッキングインタフェース
ネットワークハードウェアからのデータ受信完了割り込みをOSが受け取り、ユーザプロセスからの受信確認要求時に通知するインタフェース。

(2) スピンチェックインタフェース
データ受信完了時に、ネットワークハードウェアがあらかじめ指定された通信領域上のアドレスにフラグ（受信完了フラグ）をたてる。これをユーザプロセスがスピンチェックして受信確認を行うインタフェース。

スピンチェックインタフェースは受信確認時にOSの介入が無いため高速な受信確認が実現できる。

3.2.5 キャッシュ制御インタフェース

ネットワークハードウェアは、送信側プロセッサの主記憶上のデータを受信側プロセッサの主記憶上へと転送する。一方プロセッサ側からは、キャッシュメモリを介してしか主記憶にアクセスできず、キャッシュ上のデータは明示的にキャッシュフラッシュ命令を発行するか、ハードウェアのキャッシュ置換アルゴリズムによって書き戻されないかぎり主記憶には反映されない。このため、データ送受信時に主記憶とキャッシュ間での一貫性制御を行う必要があるが、通信性能高速化のためこの一貫性制御をOSが行うかユーザプロセスで行うかをユーザプロセスが指定できるインタフェースとする。これにより、主記憶とキャッシュの一貫性をユーザプロセスで保証できる場合には、OSによる一貫性制御を削除できるため通信処理が高速化できる。

4. おわりに

SR2001の直接メモリ転送機能のAPIについて述べた。直接メモリ転送機能は送受信側のユーザプロセスに固定的に通信領域を割り当て、ユーザプロセス空間間で直接データを転送する。このため、ネットワークハードウェアの性能を最大限に引き出す高速なプロセッサ間通信を提供できる。

参考文献

[1] 西門、他: SR2001 OSの開発コンセプト 情報処理学会第50回全国大会(1995)

[2] 山本、他: SR2001 におけるカーネルデバッガ 情報処理学会第50回全国大会(1995)

[3] 吉松、他: SR2001 における並列トレーサ／モニタ機能 情報処理学会第50回全国大会(1995)