4C-10

マルチプロセッサ間での メッセージ通信オーバヘッドの短縮について

永松　礼夫　　森下　巖　(東京大学工学部)

はじめに

本稿では、さきに提案した共有バス方式[1)]を採用して試作したマルチマイクロプロセッサ・システムでの経験をもとに、マルチプロセスを扱うＯＳの同期通信プリミティブのオーバヘッドを低下させる改善方式について論じる。

試作システムの概要

システムは図１のような構成のマルチプロセッサであり、試作したＯＳはひとつのプロセッサ上に複数のプロセスを配置し実行する方式である。各プロセス間では、ＯＳの提供する同期通信プリミティブによってのみ制御と情報の伝達がなされる。

メッセージ通信のために専用の高速バス(Mバス)を設け、Mバス・インタフェースをハードウェア化することで低いバス使用率のままシステムを運用することができる。

ユーザに公開されるＯＳプリミティブはblocking send方式をとっている。また、プロセス識別子からプロセスの置かれたプロセッサが判定可能なので、Mバスの上のメッセージはヘッダ情報を判定することにより各プロセッサの持つインタフェース回路に自動的にとりこまれる。

なお、ユーザ側からはプロセッサを意識せずにプログラムできるので、送り手と受け手のプロセスが同じプロセッサ上にあっても、違うプロセッサ上にあっても通信が似た処理量で行えることが望ましい。

メッセージ通信の動作

メッセージ通信は図２のような順序で行なわれる。図中で ── はプロセスがready状態に、▬ はプロセスがwait状態にあることを示す。また、≡ はプロセスの状態のかわるスイッチングである。

送り側で一つのblocking sendが完了する間(図のＡからＢまで)には、３つのスイッチングと２つのシステム・コールがある。

メッセージの転送

ユーザ・プロセスによって作られたメッセージは、図３のようにいくつかの段階をへて他のプロセスへと転送される。まず、ユーザ空間上のメモリ・イメージ(A)はＯＳの通信プリミティブによってMバス・バッファ(B)へと移さ

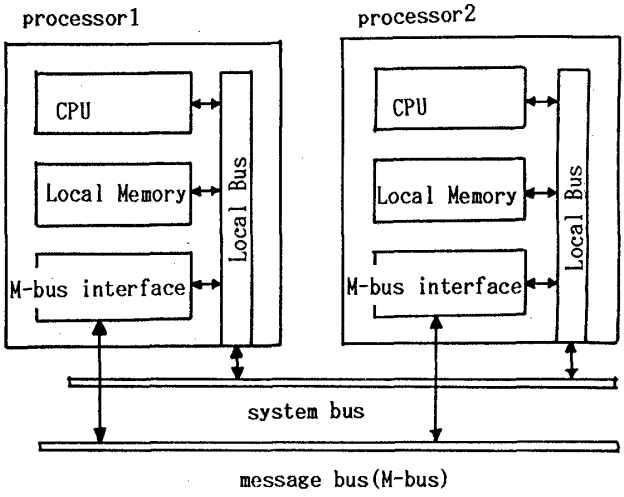


図１　システムの構成

れる。つぎに、Mバス・インタフェース回路によって相手のバッファ(C)へおくられる。さらに、相手側のプロセッサのランタイム・ルーチンがシステム空間上のキュー(D)に置く。そして、受け手のプロセスがreceiveプリミティブを行ったとき、キューから受け手プロセスの空間(E)にデータが移る。図中の短絡路のS1は同一プロセッサ内でのメッセージのときに、S2は受け手のプロセスが既にreceiveを行っていてかつキューが空のとき使われる。

試作システムでは、Mバス・バッファ相互の転送はハードウェアにより0.125μｓサイクルで、その他の転送は主記

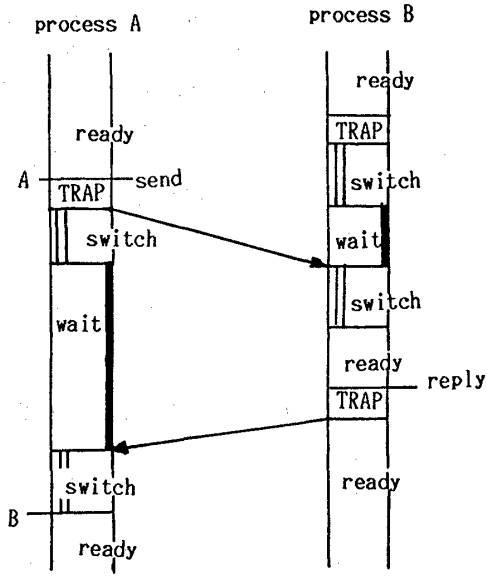


図２　Blocking Send方式

Overhead Reduction Methods
for Interprocessor Message Transfer
L.Nagamatsu I.Morishita
University of Tokyo

億のサイクルの0.5μｓでなされる。従って、平均的な10ワードの長さのメッセージでは転送のみで片道32.5μｓを要する。

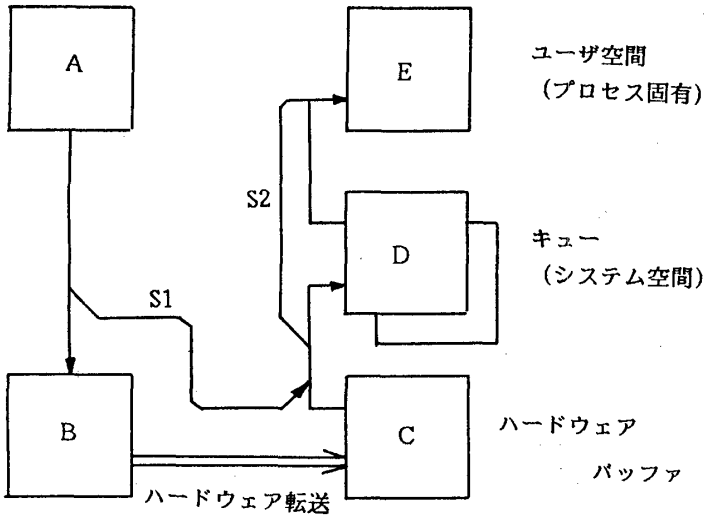


図３　メッセージ情報の移動

プロセス切り換えの評価

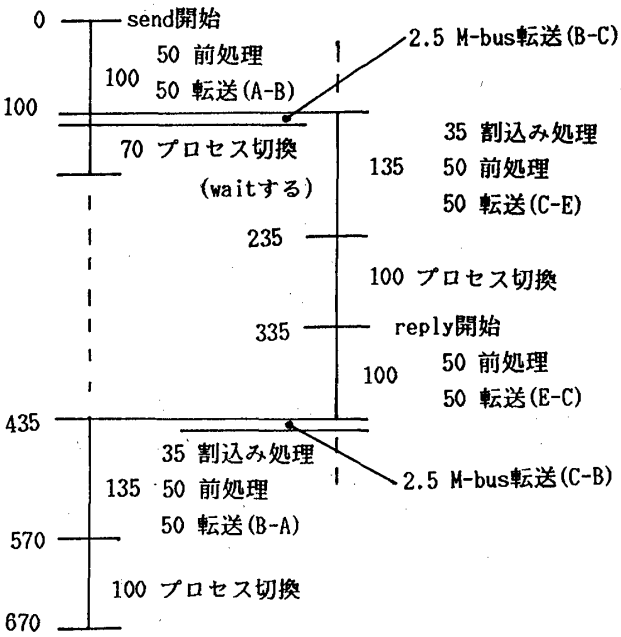
図４はプロセッサ間メッセージとプロセッサ内メッセージでの２プロセス間の往復メッセージ通信の過程を時間をおって示したものである。時間はモデル化したプロセッサで計算したもので、0.5命令／μｓとしてある、これは８ＭＨｚの６８０００にほぼ相当する。プロセッサ間メッセージの場合コンテクスト・スイッチの回数が一つ多いが相手プロセッサのreceiveと併行してなされるのでオーバヘッド増加には寄与してない。プロセッサ間での処理が多くなるのは主に送り手でも受け手でも可変長メッセージのチェックと転送をしていることによる。

メッセージ領域の割当て

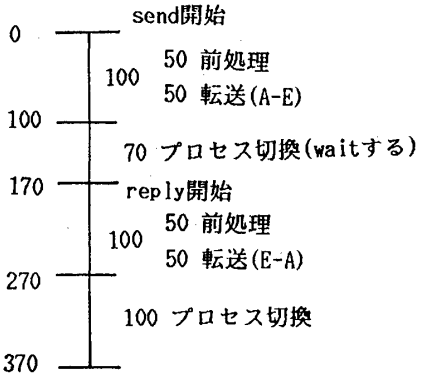
メッセージ通信をサポートする高級言語の導入によってメッセージを組立てるメモリを、ユーザ空間ではなくハードウェアバッファの上に取ることが出来る。現在の方式ではひとつのプロセッサにメッセージが届いた場合にどのプロセスへ宛てられたものかをソフトウェアが判定しているので、プロセッサ間メッセージの処理時間の増大の原因になっている。blocking send方式を用いると、同一のプロセスからのメッセージが複数個キューに入ることはないのでキューの容量の管理が容易になり実現性がたかい。

複数レジスタ対

すばやいプロセススイッチを実現するため多重レジスタ対をプロセッサ内に置く。プロセスの数だけレジスタ対を持たなくても、次のレジスタ退避までの時間が長ければ、あいているメモリサイクルを使ってセーブできるので２対で充分である。図４ではスイッチングの間隔は200μｓでレジスタ退避に必要な時間の10倍ちかくある。プロセス切り換えの最初の時間の節約になる。



（ａ）プロセッサ間メッセージ



（ｂ）プロセッサ内メッセージ

図４　メッセージ通信の過程

まとめ

いくつかのハードウェアの増強をすることでプロセッサ内メッセージとプロセッサ間メッセージとの処理時間の比を２以下にできることが判明した。実際のシステムとしてインプリメントすることが今後の課題である。

参考文献

［１］ワタリ、吉田、永松、森下：＂メッセージ／データ通信用　２重共有バス型マルチプロセッサシステム＂、第３２回情報処理学会全国大会論文集、5Q-6(1986)