

3N-9

分散メモリを持つ並列計算機シミュレータ

波多野祥二 塚田雅博
(株)沖テクノシステムズラボラトリ

1. はじめに

最近、並列計算機の研究開発⁽¹⁾が活発に行なわれており、並列計算機の性能を十分に引き出すことができるソフトウェアを開発することは重要な課題である。特に完全に分散されたメモリシステムを持つ並列計算機においては、各プロセッサ上の処理とプロセッサ間通信の結合が性能に決定的な影響を与えるため、ソフトウェア開発時にこれらについて評価を行なっておく必要がある。そのため、並列計算機のシミュレータは応用プログラム(A P)開発において有効なツールとなる。

本論文では、A Pの開発支援を目的とした並列計算機シミュレータの設計について報告する。本シミュレータは、A Pに依存しないこと、計算機の内部状態が容易に監視できること、A Pのデバッグを支援する機能を持つことなどの特長がある。

2. 対象とするハードウェアの構成

本シミュレータが対象とする並列計算機はM I M D型疎結合マルチプロセッサシステムである。各プロセッサは図1の構造を持ちプロセッサ間通信用の一つのF I F O型入力バッファ(I B)と一つのF I F O型出力バッファ(O B)を持つ。各プロセッサ上には以下の機能を持つ同一のシングルプロセスのモニタを持っている。

(a) メッセージ通信機能

パケット通信方式を使用して、プロセッサ間のメッセージの送信受信を行なう。A Pからメッセージの読み込み要求があるまで、受信したメッセージをメッセージバッファと呼ばれる領域(M B)に保管する。

(b) 資源管理

タイマー、記憶領域等の管理をする。

3. 設計方針

シミュレータ上でA Pの効率的な開発が行なえるようにするために、本シミュレータを以下の設計方針のもとに設計した。

Simulator of a loosely coupled parallel computer
Shoji HATANO, Masahiro TSUKADA
Oki Techno Systems Laboratory, Inc.

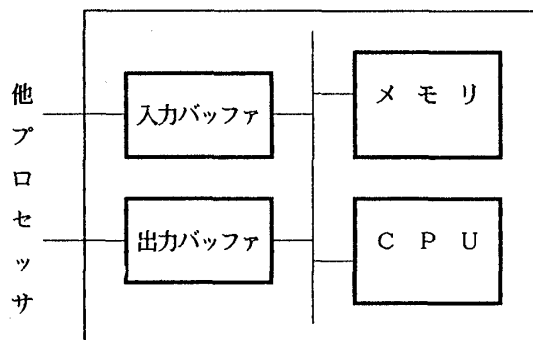


図1 プロセッサのハードウェア構成

(a) シミュレーションの対象となるA Pを本シミュレータとは独立に開発できるようにする。

(b) 操作方法を簡単にする。

(c) 並列計算機上でのデバッグに対応できるようにする。

(d) 仮想並列計算機の内部状態(プロセッサ、I B、O B、M B等)変化の表示をするモード(リアルモード)を持つようにする。

4. 機能

本シミュレータの機能を分類すると、シミュレーション機能とデバッグ機能の2種類の機能に分けられる。

シミュレーション機能として、各プロセッサ上のI B、O B、M Bの管理及びシングルタスクモニタの機能を擬似実行する。

デバッグ機能として、以下の機能がある。

(a) 各種バッファの内部状態の表示機能

コマンドにより指定されたプロセッサ上の各バッファに取込まれているメッセージ概要(メッセージのヘッダ部)を表示する。

(b) A Pの起動、一時中断、再開、強制終了機能

プロセッサ毎にA Pを割り当て起動させること、また、そのA Pの実行を一時中断、再開、強制終了させることがコマンドにより可能である。

(c) プロセッサ間通信の履歴記録機能

メッセージの送信受信毎にメッセージのヘッダ部、送信受信区別とその時刻をファイルに記録する。

(d) リアルモード機能

コマンドにより指定された時間間隔毎に、指定された回数だけ各プロセッサの状態を表示する。

(e) AP上の変数の値変更機能

AP上で指定された変数の値をコマンドにより変更できる。

5. プログラム構成

開発実行環境はUNIX⁴. 3BSDとした。本シミュレータは2種類のプロセスから構成される。一つはモニタのシステムコール、コマンド、バッファ管理、プロセッサ間通信などを行なうプロセス(シミュレータプロセス)である。もう一種類はプロセッサに割り当てられたAPの処理を行なうプロセス(APプロセス)群である。各APプロセス間の通信は全てシミュレータプロセスと通して行なう。また、シミュレータプロセスと各APプロセスとの通信にはUNIX⁴. 3BSDのソケットの機能を使用した。(図2参照)

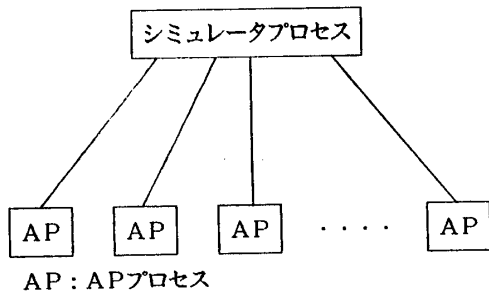


図2 プロセス構成

プロセッサ間でやり取りされるメッセージの流れを明確に表現するため、各APプロセスと端末からシミュレータプロセスへの要求(システムコールとコマンド)・各バッファ間のメッセージの転送を事象として、シミュレータプロセスを事象駆動型のプログラムとした。図3にシミュレータプロセスのプログラム構成を示す。以下、各処理の説明をする。

(a) 前処理

プロセス間通信の設定や変数の初期化等を行なう。

(b) 事象選択処理

システムコールやコマンド等の事象の中から次に実行すべき事象を選ぶ。

(c) 事象実行処理

事象に対する処理を行なう。アルゴリズムを簡潔に記述し、現実に近い並列処理を行なうために、各事象に対する処理をルーチンとして実現する。

(d) 事象実行後処理

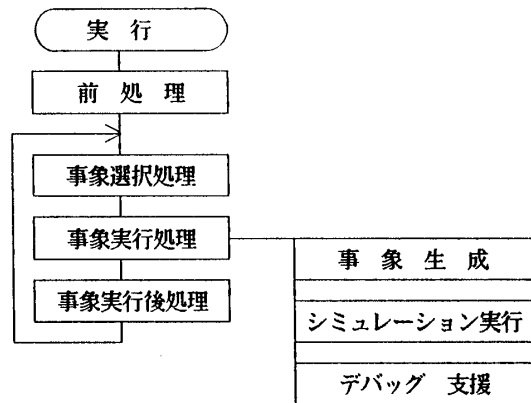


図3 シミュレータプロセスのプログラム構成

コマンド名	機能概要
runap	APの起動
stopap	APの中断
contap	APの再開
killap	APの強制終了
dlset	デバッグレベルの設定
cfexe	コマンドファイルの実行
end	本シミュレータの終了
help	コマンド一覧の表示
real	リアルモードへの移行
meshead	メッセージヘッダの表示
mesdump	メッセージ内容の表示
fifo	FIFOフルのワーニング回数

図4 コマンド一覧表

実行した事象とその事象を実行するために確保された領域を解放する。

事象のうち、コマンドを図4に示す。

APが本シミュレータを利用するために、本シミュレータはシステムコールをシミュレートするためのライブラリを提供する。APプロセスはこのライブラリを通してシミュレータプロセスとプロセス間通信を行なう。また、このライブラリにはAPプロセス側のシミュレーション実行の初期化を行なう関数も用意されている。

6. おわりに

以上でAP開発支援を目的とした並列計算機シミュレータについて述べた。現在本シミュレータの制作を行なっており、今後APの開発に使用する予定である。

[参考文献]

(1) 富田真治: 並列計算機構成論, 昭晃堂, 1986

* UNIXはAT&Tベル研究所が開発し、ライセンスしているOSです。