

TOP-1 オペレーティングシステム

3P-4

(1) 基本方針

穂積元一, 白鳥敏幸, 吉永秀志, 大澤暁, 山崎秘砂, 河内谷清久仁, 森山孝男
日本アイ・ビー・エム株式会社 東京基礎研究所

1. はじめに

我々は高速並列処理ワークステーション(TOP-1)の上にAIX[®]をベースとした並列処理オペレーティングシステムを設計作成中である。第1版は既に稼働中である。

TOP-1プロジェクトの目的とは、密結合共有メモリ型の並列処理ワークステーションを実現することで、汎用性の高いコストパフォーマンスの実現と、それに伴う各種ソフトの研究にある。

今回は特にオペレーティングシステムに焦点を絞っているが、言語処理系及びアプリケーションも重要な研究課題として捉えている。本論に入る前にアーキテクチャに若干触れるが、詳しくは参考文献[1]を参照されたい。

2. TOP-1システムの構成

図1に示されるとおり、TOP-1のシステム構成は、10台のインテル80386プロセッサを使用した、共有メモリ型マルチプロセッサシステムである。このシステムはローカルメモリをほとんど使用しておらず、すべてのCPUから完全に均質に共有された共有メモリを中心としている。従来はローカルメモリを使うことで、各ノードに固有なデータを割りつけ、これによって共有メモリとバスの競合を軽減するのが普通であった。しかし本システムはそのシステムソフトと相まって汎用性を大きな特徴としているため、複雑なメモリ管理を要求するローカルメモリを使用せず、大部分を共有メモリとしている。ただし共有メモリ型アーキテクチャの欠点であるバス競合を大幅に低下させるべき工夫としてスヌープキャッシュと高速バスを採用して、この問題に対処している。

浮動小数点プロセッサとしては、80387より高速のWeitek 1167を採用している。又、I/Oに関しては、図1にも示されるとおりハードディスクとしてSCSIの300MBのものを4台実装しており、その他のI/Oについては弊社の製品IBM PS/55[®] Model 5570をI/Oカードを通じて接続することで行っている。

3. TOP-1オペレーティングシステム

3.1. TOP-1オペレーティングシステムのアプローチ

TOP-1オペレーティングシステムはシングルプロセッサ用のOSであるAIXをベースとしてMP-OSを作成しようとしている。そこで次のようなアプローチが考えられる。

- a. OSのカーネルをできるだけエントラント化する。
- b. OSのカーネルはどれか一つの特定のCPUでのみ実行させ、ユーザプロセスを他のCPUで並列動作させる。
- c. OSのカーネルはシリアリリユーザブルのまま、任意のCPUで動作させるようにする。

bは概念が明確である分作成は容易であるが、より良いパフォーマンスを求めると、どうしてもaを実現しなければならない。ただしaはbに比べて作成は困難であり、又幾通りかの実現方法があり、将来的な並列処理に耐えられるカーネルの構造をも実現しようとする、どのようなアプローチでも良いというわけではない。更に作成上のことを考慮に入れると、並列処理OSのデバッグはかなり困難なものであり、初期の設計が余程完全であるか、ステップ・バイ・ステップに作成するかのいずれかと思われる。我々は既存のOSの並列化を行っているので、一からの完全な設計は望めない。そこでb→c→aのアプローチで作成することとし、最終段階のaを実現する段で十分な将来的な機能分割を達成する事を目指している。

3.2. TOP-1オペレーティングシステムの研究課題

このような並列処理オペレーティングシステムを作成する上で、どのような研究課題を考えて行っているかを述べる。

1. OSのカーネルの機能分割
2. ユーザに対する並列処理のより良いモデルの提供
3. 並列処理特有の問題の解決

1については、今後の並列オペレーティングシステムを考える上で本質的な問題であり、現時点では多くを論文に

TOP-1 Operating System (1) Basic Direction

Motokazu Hozumi, Toshiuki Shiratori, Hideshi Yoshinaga, Gyo Ohsawa, Hisa Yamasaki,

Kiyokuni Kawachiya, and Takao Moriyama

IBM Research, Tokyo Research Laboratory, IBM Japan Ltd.

できないが、概念のみ説明する。

2については、我々は一般にスレッドと呼ばれているモデルを導入しているが、これについては幾通りかの考え方がある。本大会でも発表する。

3については、TLBフラッシュの問題を含めた並列処理特有の資源管理の問題がある。メモリ管理については本大会にも発表する。

3. 2. 1. カーネルの機能分割

カーネルのできるだけ多くの部分をリエントラントにして、並列度を上げ性能向上を計ることは、並列処理オペレーティングシステムの作成上の一つの眼目である。もちろん必要なクリティカルセクションをきちんと認識し、ロックを取る事によってこれらは達成される。しかしかかしてデッドロックを回避するか、更には将来的な展望を考えた場合望まれるカーネルの構造との調和が問題として残る。

そこで我々はマルチサーバモデルというものを採用しており各サーバは一つのみまとめた機能単位として、サーバ単位の並列性の実現を考えている。これによってサーバの入れ替えや追加、更には機能分割型のアーキテクチャにも柔軟に対処できることを目指している。

3. 2. 2. スレッドモデル

一般にスレッドと呼ばれているモデルのインプリメントには幾通りか存在し、もちろんそれぞれに意味が異なっている。例えばSUNのlight-weight-processはある意味のコーチンとして実現されている(参考文献[2])。又CMUで開発されたオペレーティングシステムMACHにもスレッドモデルは採用されている(参考文献[3, 4, 5])。我々はスレッドを共有メモリを持つ並列計算のモデルとして実現している。この共有メモリのもたらす実行環境をプロセスといい、その中の実行文脈(コンテキスト)をスレッドと言っている。

4. おわりに

現在TOP-1オペレーティングシステムは稼働中である。我々は新しいバージョンのMP-O/Sを作成中であり、同時にCommon Lispの並列化も行った(参考文献[6])。更には並列Fortranの研究にも着手し、いくつかの並列アプリケーションを走行させる予定である。もちろん我々のOSはAIXの完全上方互換であるので、AIXのアプリケーションはそのまま並列に走行させる事ができる。今後これら並列アプリケーションから有用なデータの収集が期待される。

5. 参考文献

- [1] 鈴木他;「高速並列処理ワークステーション(TOP-1)」、情報処理学会第37回(昭和63年後期)全国大会論文集(1988)
- [2] "System Service Overview," Sun Microsystems Inc.
- [3] Robert V. Baron, David Black, et al, "MACH Kernel Interface Manual,"Carnegie-Mellon Univ. (1987)
- [4] Avadis Teranian Jr., Richard F. Rashid, "MACH: A Basis for Future UNIX Development,"Carnegie-Mellon Univ. (1987)
- [5] Avadis Teranian Jr., Richard F. Rashid, et al, "MACH Threads and UNIX Kernel," Carnegie-Mellon Univ. (1987)
- [6] Hozumi, Kurokawa, Suzuki, Tanaka, and Uzuhara, "Multiprocessor Common Lisp on TOP-1," In Preprints of US/Japan Workshop on Parallel Lisp. (1989)

* AIX, PS/55はIBM Corp. の商標です。

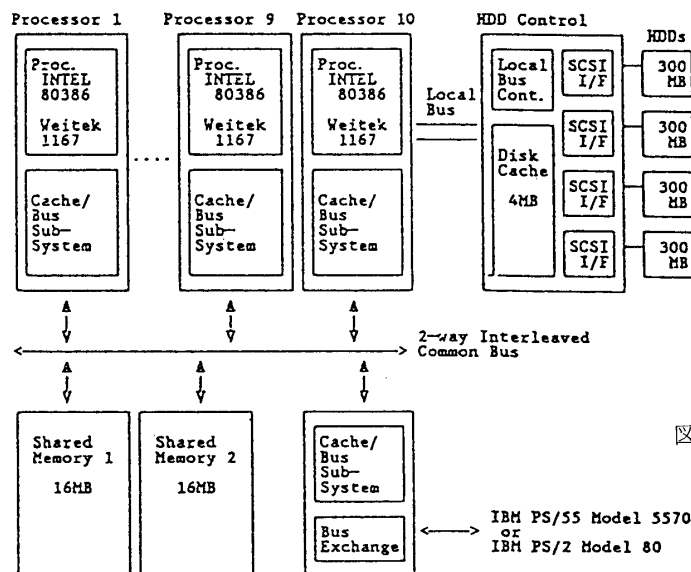


図1 TOP-1のハード構成