TOP－1 オペレーティング・システム
(2) プロセス・スケジューリング

3P-5

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

1．はじめに

TOP－1は主記憶共有型の小規模マルチプロセッサ・ワークステーションである[1]。IBMのオペレーティングシステム「AIX PS/2™」をベースにしてTOP－1のOSを開発した。このシステムは並列処理の研究のテストベッドとして使用することを目的にしており、OSも性能よりは開発期間の短さを重点にして設計された。本稿ではこのOS（TOP-1 OSと呼ぶ）のマルチプロセッサ対応化について概説したあと、そのプロセス・スケジューリングについて説明する。（なお、TOP-1 OSではプロセスの他にも、スレッド[2]と呼ばれる並列プログラミング要素がスケジューリングの単位になりうるが、以下では簡単のためこれらをまとめてプロセスと呼んでいる。）

2．カーネルのマルチプロセッサ化

AIX PS/2 はもともとユニ・プロセッサ用のOSであり、当然そのままではマルチプロセッサ上では動かない。最も大きな問題は、OSのカーネルがリエントラントな構造になっていないため、一時には1個のプロセスしかカーネルのコードを実行できないことである。もし同時に複数のプロセスがカーネルを実行するとカーネル内のデータ構造が壊れてしまう。マルチプロセッサ上でカーネルを正しく動かすには、これらのデータに対して排他制御を行い、その内容を保護することが必要である。

そのための1つの方法は、カーネル内の様々なデータ構造に対してそれぞれ排他制御を行うようカーネルのコードを書きかえてしまうことである。カーネル自体がリエントラントになるので、複数のプロセッサ（の上で走っている複数のプロセス）が並行してカーネルを実行できる。そのため、ユニ・プロセッサの場合とくらべてもカーネルの処理速度がさほど落ちないことが期待できる。しかしカーネルのコードは巨大なものであるから、必要な書きかえの量も大きいと予測される。我々はなるべく早い時期に実動するOSを作りたかったため、この方法はとらなかった。

もう1つの方法では、カーネル自体は非リエントラントのまま、カーネル全体を1つのブロックとして排他制御を行う。複数のプロセスが同時にカーネルを実行しようとしたときはその中の1個だけが実行を許可され、それ以外はカーネルの入口で待たされる。この方法の長所、短所は1番目の方法のちょうど逆である。カーネルの実行量が多いプロセスがシステム内にいくつも走れば、カーネルの処理速度の低下は避けられない。しかし、短期間でカーネルをマルチプロセッサ化できるメリットがあるためTOP－1OSではこの方法をとることにした。

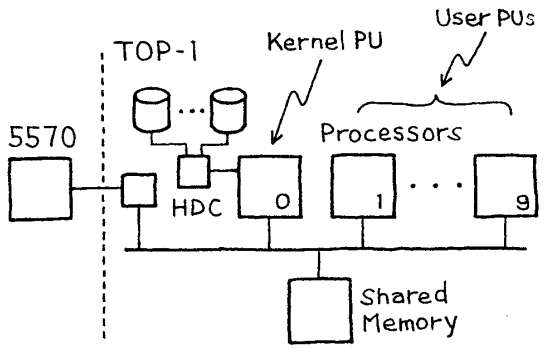

図1　TOP－1の構成

カーネルコードを実行するプロセッサの区分によりさらに2通りの実装が考えられる。すべてのプロセッサがカーネルを実行する権利を持つ方法と、特定の1個のプロセッサのみがカーネルを実行する方法である。我々は後者の方法をとっている。これはTOP－1のプロセッサすべてが均質ではないためであるが、詳しくは次節で述べる。

3．TOP-1 OSの仕組み

3．1　プロセッサの構成

図1にTOP－1のハードウェア構成を示す。TOP－1は本体とIBM PS/55™モデル5570 一台から成る。5570は外部との入出力専用のプロセッサとして用いられる。本体のプロセッサカード10枚のうちの1枚にハードディスク制御カード（HDCカード）が接続され、HDCカードを介してハードディスクの制御、入出力を行う。我々はこのプロセッサにカーネル実行の役目を割り当て、カーネルPU(Kernel Processing Unit)と呼んでいる。入出力操作はカーネルの仕事であり、またカーネルPUのみがハードディスクに直接アクセスできる構成のため、この設定はTOP－1のハードウェアに最も適していると考える。他の9台のプロセッサはプロセスのユーザープログラム部分を実行し、ユーザーPU(User Processing Unit)と呼ばれている。9台のユーザーPUに区別はなく、ユー

TOP-1 Operating System (2) Process Scheduling
Hisa Yamasaki, Takao Moriyama, Kiyokuni Kawachiya, Toshiyuki Shiratori, and Motokazu Hozumi
IBM Research, Tokyo Research Laboratory, IBM Japan, Ltd.

ザープロセスはこれらのＰＵで並列に実行される。

３．２　ユーザープロセスの動作

一般に、ユーザープロセスはユーザープログラムを実行している状態（ユーザーモードと呼ぶ）とカーネルコードを実行している状態（カーネルモードと呼ぶ）を交互に繰り返している。ユーザーモードのプロセスは、システム・コールをしたりページ・フォールトを起こしたりの理由でいずれはカーネルモードに移行する。そしてカーネルモードでの処理が終わるとプロセスは再びユーザーモードに戻り、もとのユーザープログラムを続行する。この繰り返しでプロセスは進行していくわけである。

ＴＯＰ-１ ＯＳではプロセスのユーザーモードとカーネルモードはそれぞれ別のプロセッサ（ユーザーＰＵとカーネルＰＵ）で実行されるため、プロセスはモードを移行するときにプロセッサを乗り換える（図２）。ユーザーモードからカーネルモードへの移行のときにはユーザーＰＵからカーネルＰＵへ (a)、カーネルモードからユーザーモードに戻るときはその逆 (b)である。

通常、システム内には多数のプロセスが走っているため、カーネルＰＵやユーザーＰＵに常に空きがあるわけではない。当然、乗り換えようとしたプロセスが待たされることもある。次節ではプロセスの各ＰＵへのスケジューリングについて説明する。

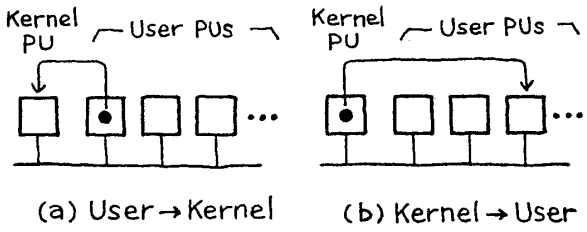

図２　プロセッサの乗り換え

４．プロセス・スケジューリング

４．１　プロセス・キュー

ＴＯＰ-１ ＯＳでは２個のプロセス・キューを使ってプロセスのスケジューリングを行っている（図３）。カーネル・キュー(Kernel run-queue)にはカーネルＰＵでの実行待ちのプロセスが入る。例えば、ユーザーＰＵからカーネルＰＵに乗り換えようとしているプロセスや、カーネル内で一時的に休止状態（ディスクの入出力待ちなどのため）になったあと再び実行可能状態になり、カーネルモードを続行しようとするプロセスである。もう一方のユーザー・キュー(User run-queue)にはユーザーＰＵでの実行待ちのプロセスが入る。これにはカーネルＰＵからユーザーＰＵへ乗り換えようとするプロセスや、ユーザーＰＵで実行中に、タイム・スライシングのためプロセッサを取り上げられた(preempted)プロセスが含まれる。

キューの順序付けには優先度を用いる。プロセスはそれぞれプロセッサ割り当ての優先度を表す数値を持っていて、各ＰＵでプロセス・スイッチングが起こると、最も高い優先度を持つプロセスがキューから取り出されてＰＵに割り付けられる。これはもとのＡＩＸ ＰＳ／２ と同等なやり方である。

４．２　ＴＯＰ－１メッセージ機構の使用

これら２個のキューはカーネルによって（つまりカーネルＰＵによって）管理されている。ユーザー・キューからプロセスを選び出すのはカーネルＰＵの仕事である。ユーザーＰＵでプロセス・スイッチングが必要なときは、ユーザーＰＵがカーネルＰＵに依頼して次に走らせるべきプロセスを選択させる。この依頼にはＴＯＰ－１のハードウェア・メッセージ機構が使われる。ハードウェア・メッセージは、あるプロセッサが他のプロセッサに短い（４バイト）メッセージとともに割り込みをかけられる機能である。プロセスがユーザーモードからカーネルモードに移るとき、そのプロセス（が動いているユーザーＰＵ）からカーネルＰＵに向かってプロセス選択を依頼するメッセージが送られる。反対に、タイム・スライシングなどの理由でカーネルがユーザーＰＵ上のプロセスを入れ替えたい場合には、カーネルＰＵからユーザーＰＵに向かってプロセス入れ換えのメッセージが送られる。

５．おわりに

ＴＯＰ－１オペレーティングシステムのマルチプロセッサ化と、そのプロセス・スケジューリングの方法を説明した。性能の面から見れば最良の方法ではないが、使用経験からすればテストベッドとして十分な性能と言えよう。

参考文献

[1] 鈴木 ："高速並列処理ワークステーション(ＴＯＰ－１）－開発方針－"，第３７回情報処理学会全国大会論文集（1988).
[2] 吉永,山内,穂積 ："ＴＯＰ－１ オペレーティング・システム （４)スレッド機構"，第３９回情報処理学会全国大会論文集（1989).

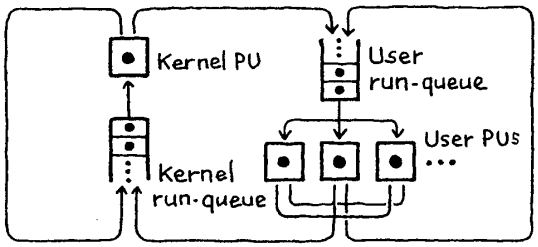

図３　プロセス・キュー

ＡＩＸ ＰＳ／２，ＰＳ／５５はＩＢＭ社の商標です。