

軽量なプロセスの集合を用いたマルチプロセッサ 用オペレーティング・システムの設計

田胡 和哉、中山 泰一(東京大学 工学部 計数工学科)

1. まえがき

マイクロプロセッサを用いたマルチプロセッサ・システムの開発が盛んに行なわれている。複数のプロセッサを利用する一つの方法として、オペレーティング・システム内部で並列処理を行なうことにより、システム性能の改善を図ることがあげられる。通信で結合された軽量なプロセスの集合であるプロセス・ネットワーク[1]を用いてシステムを実現することにより、システム機能を並列に処理することを試みる。既存のマルチマイクロプロセッサ・システム上で、UNIXシステムと同一の外部仕様を持つシステムの開発を行なっているので、その設計について述べる。

2. 並列処理によるシステム性能の改善

マルチプロセッサ・システムでは、オペレーティング・システム内部で並列処理を行なうことにより、システム・コールの処理に要する時間を短縮できる。また、システム機能の実行の負荷を複数のプロセッサに分散することにより、利用者プログラムに割当てられるCPU時間が減少することを防ぐことができる。これにより、単一プロセッサ用に開発された利用者プログラムについても、その処理時間の短縮が期待できる。さらに、並列処理を行なう利用者プログラムでは、並列処理単位ごとにシステム・コールを発行するために、システム・コールの発生頻度が増加することが予想される。システムがボトルネックとならないために、システム内部で並列処理を行なう必要性が高い。

3. プロセス・ネットワーク

(1) 実現方式

並列処理によってプログラムの処理性能の改善を図るために、プログラムができるだけ多数の並列実行単位に分割すること、および、並列処理の実現コストを軽減することが必要である。プロセス・ネットワークは、この目的に適合する性質を持つ。

プロセス・ネットワーク方式では、相互排除アクセスされる資源の各々にプロセスを配置し、それらを通信を用いて結合することによりシステムを実現する。プロセス・ネットワークを構成するプロセスをシステム・プロセスとよぶ。同一のプロセッサ上のシステム・プロセスは、単一の論理アドレス空間内で動作し、その実現コストはユーザ・プロセスのそれに比べて小さい。

システム・プロセス間の通信方式としてランデブ方式を用いる。通信は、OS核が実現する。マルチプロセッサ・システムにおけるプロセッサ間の同期は、OS核が実現する。マルチプロセッサ用のOS核を開発し、单一プロセッサ用のオペレーティング・システムのプロセス・ネットワークをそのまま利用してシステムを実現する。

(2) プロセス・ネットワーク方式によるオペレーティング・システム

これまでに、プロセス・ネットワーク方式を用いて、通信回線で結合された2つのプロセッサからなるシステム用の分散型オペレーティング・システムの開発を行なった[2]。システム評価の結果によれば、並列処理による性能の改善が可能であることが判明した。たとえば、ファイルのクローズ、および、削除を行なうシステム・コールでは、利用者プログラムは処理の終了を待合わせる必要はない。ファイルを管理するプロセスは、利用者プロセスと並列に実行することができる。さらに、ファイルの入出力を実行することができる。さらに、ファイルの入出力を行なうシステム・コールの処理では、全体の処理の約3分の1は、利用者プログラムへの返答を行なったのちに実行されている。この処理も、利用者プログラムと並列に実行することができる。また、並列処理向きにプログラムを改良できる余地が大きい。特に、ファイルの先読み、後書き処理の有効性が高いことが予想される。

4. 設計

4. 1 対象システム

設計したシステムは、沖電気製のNTC(Network Communication processor)システムを制御することを目的としている。NTCシステムは、68020プロセッサ、局所バス、および、局所メモリを持つプロセッサ・ユニットを、共有バスを用いて結合することにより構成されている。各プロセッサは、局所メモリ、および、共有バス上の共有メモリにアクセスすることができる。共有バスへのアクセスの調停は、VMEバスと同等の方法で実現されている。周辺機器は、局所バスに付加される。

4. 2 OS核

プロセスは、call、acc、および、endrプリミティブを発行することによって通信の実行をOS核に依頼する。callプリミティブを、通信相手プロセスの識別子、エントリの識別子、および、通信引数列を引数として呼出すことにより、エントリ呼出しが実現される。通信引数は、その実体のアドレス、サイズ、および、引数の転送方向によって識別される。accプリミティブをエントリの識別子を引数として呼出すことにより、受け付けが実現される。endrプリミティブによりランデブが終了する。

図1に、システム構成の概要を示す。システム・プロセスは、特定のプロセッサ上でのみ動作し、移動しない。異なるプロセッサ間の通信では、通信引数の実体を共有メモリ上のバッファに保持する。通信ごとに割当てられるこのバッファを、ICB(Inter-processor Communication Buffer)とよぶ。

図2に、ICBの構造を示す。ICBのはじめの3つのフィールドは、リンク構造を実現するために用いられる。この部分の構造は、システム・プロセスを制御するために用いられるPCB(Process Control Block)の対応する部分の構造と同一であり、両者を同一のリンクにつなぐことができる。第4のフィールドは、エントリ呼出しを行な

Design of a Multiprocessor Operating System by
the Set of Lightweight Processes

Kazuya TAGO, Yasuichi NAKAYAMA
University of Tokyo

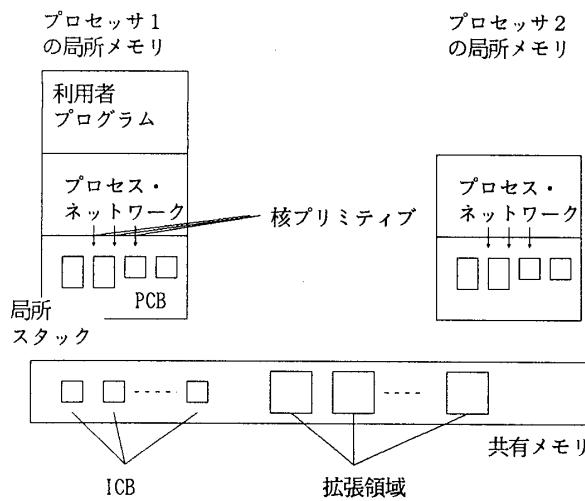


図1 システムの構造

ったプロセスの識別子を保持する。残りのフィールドは、プロセスの局所スタックと同一の構造を持ち、通信引数の識別子、および、引数の実体を保持している。accは、異なるプロセッサからのエントリ呼出しを受け付けると、返り値としてICBの第6フィールドへのポインタを返す。このポインタを用いて、通信引数の結合を実現する。ICBは、64バイト固定長である。これに通信引数の実体が保持できない場合は、別個に割当てられる拡張領域を用いる。

異なるプロセッサへのエントリ呼出しの実現法の概要を図3に示す。エントリ呼出しを行なったプロセスが存在するプロセッサのOS核は、共有メモリ上の空きICB

ンデブの終了を知る。OS核は、通信実体のうち必要なものを送り手のプロセスの局所スタックにもどし、ICBを解放する。

共有メモリは、プロセッサ間の通信において、通信引数を保持するためのみ用いられる。そこで、共有バスの負荷は、共有メモリのみを用いる方式に較べて、はるかに小さくすることができる。

4.3 オペレーティング・システム

単一の68010システム上で、プロセス・ネットワーク方式によって実現された、UNIXシステムと同一機能を持つオペレーティング・システムが動作している。このプロセス・ネットワークを、複数のプロセッサ上に分散して配置することにより、システムを実現する。さらに、複数のプロセッサ上で利用者プログラムが実行できるように、拡張を行なう。これは、利用者プロセスの起動を行なうforkシステム・コールの処理において、プロセス・イメージの複写を直接実行せずに通信を経由するよう変更すること、および、システム・コールの受け付けを行なうシステム・プロセスをすべてのプロセッサに配置することにより実現できる。

5. むすび

システム内部で並列処理を行なうことによりシステム性能の改善を図ることを目的とした、マルチプロセッサ・システム向きオペレーティング・システムの実現方式について述べた。実現を完了し、評価を行なうことが今後の課題である。これをもとに、通信で結合されたプロセスの集合体の性能に関するより一般的な議論を行いたい。

プロセス・ネットワークを用いることにより、共有メモリを持つマルチプロセッサに限らず、多様な構成の並列処理システムを結合して統合的なシステムを実現するために利用することができる。提案方式を、このようなシステムの実現に適用することができる。今一つの課題である。

謝辞

NTCシステムについて御配慮いただいた、沖電気株式会社システム開発センターの関係各位に感謝いたします。

参考文献

[1]田胡、益田：オペレーティング・システムの構造記述に関する一試み、情報処理学会論文誌、Vol. 25, No. 4, pp. 524-534 (1984).

[2]高野、田胡、益田：プロセス・ネットワークによる分散型オペレーティング・システムの設計、情報処理学会論文誌、Vol. 29, No. 4, pp. 359-367 (1988).

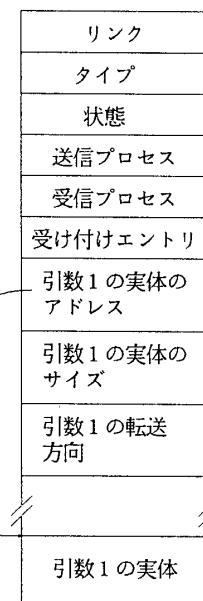


図2 ICBの構造

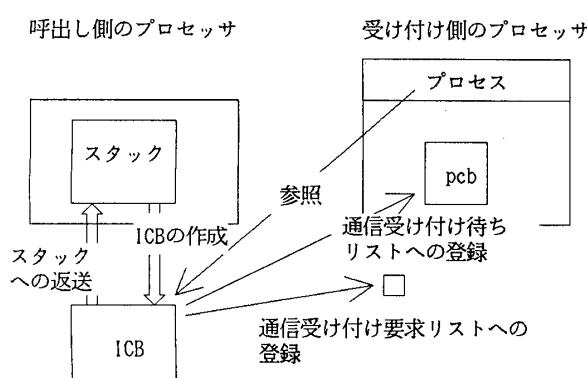


図3 異なるプロセッサ間の通信の実現

リストからICBの割当てを行なう。プロセスの局所スタック上に積まれた引数を走査し、ICBに引数の実体を転送する。作成したICBを、プロセッサごとに割当てられる通信受け付け要求リストにつなぐ。リストが空である場合には、相手プロセスに割込みを発生させる。

受け付けを行なうプロセッサのOS核は、通信受け付け要求リストからICBを取り出し、エントリごとに設けられる通信受け付け待ちリストにつなぐ。accプリミティブにより、リストからはずされてプロセスに渡される。endrプリミティブによってランデブが終了すると、OS核は、ICBを、プロセッサごとに割当てられる返答リストにつなぎかえる。これにより送り手のOS核はラ