5F－4

連続メディア処理向きマイクロカーネルの開発（１）
－システムの概要と基本設計－

岩嵜正明，中原雅彦，竹内　理，中野隆裕，芹沢　一
(株)日立製作所　システム開発研究所

１．はじめに

MPEG、ATMネットワーク等を始めとするマルチメディア技術の進歩により、コンピュータ内部にデジタル・ビデオ等の連続メディアを取り込み、さらに、これをネットワーク経由で転送できる様になってきた。この背景には、プロセッサやネットワークの高速化、半導体メモリやディスク装置の大容量化等のハードウェア技術の進歩がある。

その一方で、これらのハードウェア技術の進歩によって、システム構成要素間の性能バランスが大きく変化し、OSの基本設計にも変更が必要となって来ている。例えば、近年、プロセッサの性能は飛躍的に向上してきたが、これに比べると半導体メモリのアクセス速度の向上巾は小さく、プロセッサとメモリの性能差が拡大している。このため、大量データ入出力が本質的に不可欠な連続メディア応用では、バッファ間のメモリコピー処理、あるいは、キャッシュとメモリ間の一貫性保証オーバヘッドが無視できない問題となっている。また、QoS（Quality Of Service）保証の困難さ、割り込み処理やコンテキスト・スイッチのオーバヘッド等も解決を要する課題である［1～3］。

現在、我々は、上述の課題解決に向けて連続メディア処理向きマイクロカーネルHiTactixの研究開発を進めている。以下では、HiTactixの設計方針と機能概要について述べる。

２．基本設計方針

HiTactixの設計目標は、QoS保証機能を有した連続メディア・リアルタイム処理に必要なカーネル機能を提供することである。この目標に向け、具体的には下記３項目の実現を設計の方針としている。

１）低ジッタの周期的なスレッド駆動機能の実現
２）優先順位逆転（Priority Inversion）問題の解消
３）入出力処理オーバヘッドの低減

尚、上記１）については、
・周期５ミリ秒～数百ミリ秒での周期駆動、
・駆動時刻のゆらぎ１ミリ秒未満、
・周期の異なる複数のスレッドを同時に駆動可能、
の実現を設計目標としている。上記３）については、ソフトウェア・オーバヘッドが、入出力スループットの制約要因とならないことを設計目標としている。

また、これら以外にも、アドレス空間構造やスレッド構造の最適化により、
・コンテキスト・スイッチや割り込み処理の発生回数、
・キャッシュミスやTLBミスの発生率、
・メモリとキャッシュ間の一貫性保証オーバヘッド、
等の低減をはかっている。

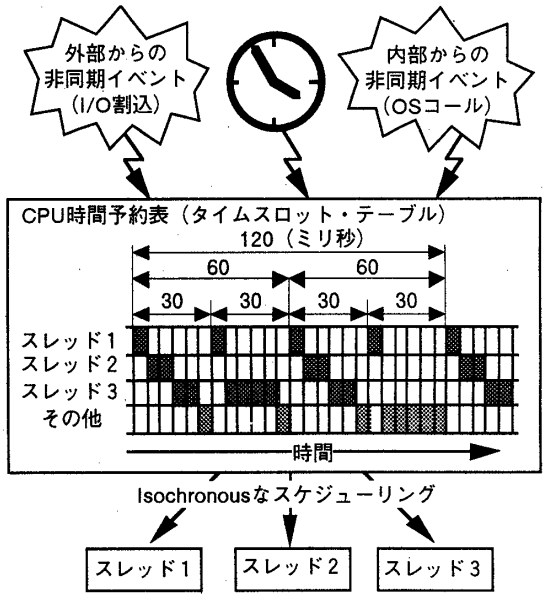
さらに、カーネル内で使用するキュー操作関数を統一する等の工夫により、高信頼化に関しても設計の初期段階から配慮している。

３．機能概要

本節では、HiTactixの特徴的な機能について概説する。

3.1. サイクリック・スケジューリング

サイクリック・スケジューリング方式は、連続メディア処理において、各スレッドの動作に周期性（即ち予測可能性）がある点に着目し、各スレッドの実行開始前に静的にCPU時間を予約する。各スレッド毎に周期や周期内の必要CPU時間は予測可能であるから、これらの値をアプリケーションに指定させることにより、無駄のないCPU時間の予約スケジュールを組むことが可能である（図１）。



上図は、
・スレッド１（周期 30ミリ秒、CPU時間 5ミリ秒）
・スレッド２（周期 60ミリ秒、CPU時間10ミリ秒）
・スレッド３（周期120ミリ秒、CPU時間40ミリ秒）
の３スレッドを予約スケジュールした状態を示す。イベント駆動のスレッドは、予約の空き時間を使ってスケジュールする。予約CPU時間未満で周期内の処理を完了した場合、余剰CPU時間はシステムに返却できる。

図１．サイクリック・スケジューリング方式

A Micro-kernel for Countimuous Media Processing －Design Principles －
Masaaki IWASAKI, Masahiko NAKAHARA, Tadashi TAKEUCHI, Takahiro NAKANO, Kazuyoshi SERIZAWA
Systems Development Laboratory, Hitachi Ltd.

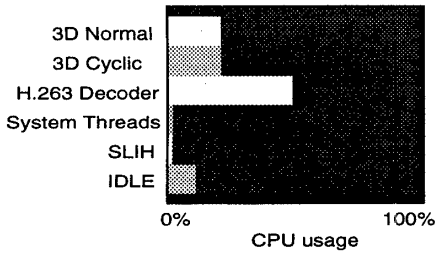


図2．CPUパワーに余裕がある場合

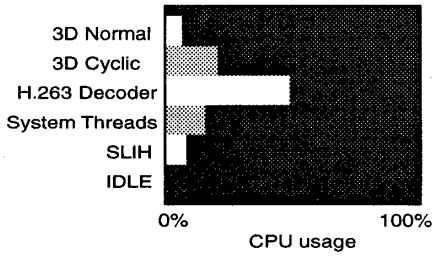


図3．CPUが過負荷状態の場合

サイクリック・スケジューラは、タイマ割り込みを契機に予約されたCPU時間の経過をチェックし、予約スケジュールに従って各スレッドにCPU時間を割り当てる。これによって、HiTactixは複数のスレッドを高精度に周期駆動し、複数の連続メディアストリームを円滑に処理することを可能にしている。

さらに、HiTactixのスケジューラは、非定常的に発生する外部割り込み等によるイベント駆動のスケジューリングと周期的な予約スケジューリングを共存させる機能を有する。図2と図3は、縦軸に実行中のスレッドを、横軸にCPU利用率を示す。IDLEはCPUのアイドル時間を示す。SLIHとSystem Threadsは割り込みによって起動されるスレッドである。H.263 Decoderと3D Cyclicは周期駆動スレッドを、3D Normalは通常のスレッドを示す。図2は、外部割り込みが少なく、CPU能力に余裕がある場合を示す。図3は、ネットワークからの負荷を増大させ、CPUが過負荷になった状態を示す。これらの図から、過負荷状態でも周期駆動スレッドのCPU割り当て時間は、ほとんど減少していないことが確認できる（これらのデータは90MHzのPentiumを搭載した日立FLORA3100SPを用いて計測した）。

3.2. 細粒度プリエンプト制御

HiTactixは、Priority Inversion 問題を、細粒度プリエンプト制御によって解決している。HiTactixは、他のリアルタイムOSと同様にカーネル内部をマルチスレッド化し、プリエンプト可能とすることで、リアルタイム応答性を向上させている。同時に、カーネル内部では、共有資源の排他アクセス制御にロック機構を一切使用していない。共有資源へのアクセス時には、スレッドは、必ずプリエンプト禁止状態に遷移し、共有資源へのアクセス権を保持している期間は、CPUの使用権を剥奪されないことを保証している。さらに、1回のプリエンプト禁止時間が一定値（100μ秒）を超えない様にカーネル内部の設計を最適化している。

カーネル内部の排他制御にロック機構を使用しているOSと異なり、HiTactixでは、カーネル内資源の競合による高優先度スレッドの実行開始遅延は100μ秒以内に保証される。

3.3. ダイレクト・バッファ・マッピング

連続メディアを扱うシステムでは、デジタイズしたビデオデータをフレームバッファに転送し、アプリケーションがフレームバッファに直接アクセスするといった機能が必要となる。 read, write等の既存入出力インタフェースでは、バッファ間のメモリコピー回避が困難であり、数十MB/secを超える高速入出力処理では、これが性能上のボトルネックとなっていた。

ダイレクト・バッファ・マッピングは、ユーザ仮想アドレス空間の一部を、入出力用のDMA転送領域として使用可能にする機能である。この機能によって、入出力に伴うメモリコピー処理やリマップ処理が不要となり、入出力処理にOS内部で費やされるCPU時間を大幅に低減できる。

3.4. その他の機能

上記の他にHiTactixは、

1）仮想アドレス空間上でOS領域等を共有することで、ページテーブル等に必要な物理メモリ量を削減するとともに、キャッシュやTLBのヒット率を向上させる仮想領域共有機能
2）高速デバイスに必要な大容量DMAバッファの割り当てを可能にする連続物理ページ割り当て機能
3）入出力割り込みやこれに伴うコンテキスト・スイッチ回数を低減する一括入出力機能

等のインタフェースを提供する。

4．おわりに

以上、連続メディア処理に適した機能と性能を有するマイクロカーネルHiTactixの概要を述べた。現在、PC-AT互換機上への実装を完了し、VODサーバへの適用実験等を通し、サイクリック・スケジューリングやダイレクト・バッファ・マッピングの有効性を定量的に評価している。

参考文献

[1] J.Pasquale, E.Anderson, P.K.Muller, "Container Shipping: Operating System Support for I/O-Intensive Application", IEEE COMPUTER, March.1994

[2] P.Druschel, M.B.Abbott, M.A.Pagels, L.L.Peterson, "Network Subsystem Design", IEEE Network, July.1993

[3] Clifford W. Mercer, Stefan Savage, and Hideyuki Tokuda, "Processor Capacity Reserves: Operating System Support for Multimedia Applications", Proc. Intl. Conf. on Multimedia Computing and Systems, 1994

[4] 竹内、中原、中野、中村他、連続メディア処理向きマイクロカーネルの開発（2～4）、情報処理学会第53回全国大会予稿集、1996