

時間保護機能を持つ組み込みシステム向け RTOS の開発 —リアルタイムアプリの統合を実現する OS—

1. 背景

近年、組み込みシステムの高性能化・高機能化により複数の機能を統合した複合システムの開発が増加している。個々の動作が保証されたアプリケーションを混在させる場合、あるアプリケーションの実行時間の遅れがシステム全体へ波及すること防ぐ保護機能、すなわち時間保護機能が必要となる。時間保護機能の目的とは、単一プロセッサ上に混在するアプリケーションの実行時間を個別に保証することである。例えば、図1は高優先度アプリケーションAの実行時間の遅れが、アプリケーションBの実行に影響を及ぼし、アプリケーションBのデッドラインミスの原因となってしまう例である。アプリケーションの見積もり実行時間は、十分にその実行を完了できるように設計されている。そこで、複合システムでは個々のアプリケーションに対して静的に決定された時間的设计を厳守することが最も重要となる。この例では、アプリケーションAが見積もり実行時間を越えた時点で、その実行をアプリケーションBに切り替える必要がある。時間保護機能は複合システムの開発段階においても有効な機能である。例えば、図2に示すようにシステム・インテグレータは、複数のベンダが低速プロセッサ上で開発・検証したアプリケーションを高速プロセッサで動作する時間保護機能付RTOS上で統合することで、容易に複合システムを構築することが可能となる。

現状では、ハードリアルタイム・アプリケーションの実行時間を保証することを目的とした時間保護機能の研究・開発はあまり行われていない。時間保護の理論的なアルゴリズムを提案しているいくつかの研究は、実装や実用を考慮していないため、提案されているアルゴリズムをそのまま実装することは困難である。

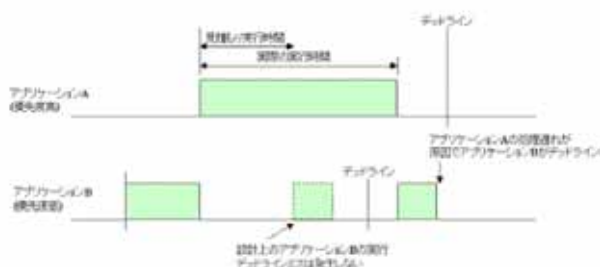


図 1: アプリケーションのデッドラインミスの例

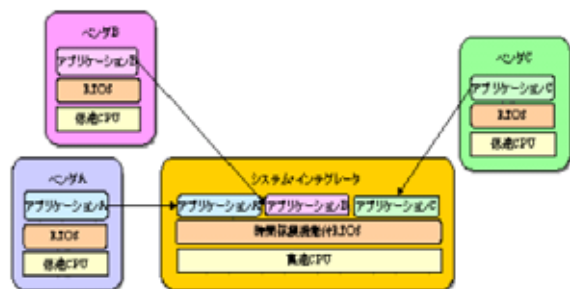


図 2: 分散開発されたアプリケーションの統合

2. 目的

本プロジェクトでは、時間保護機能をOSレベルでサポートするRTOSの開発を行う。具体的にはまず、現在提案されているアルゴリズムを、実装を念頭においた時間保護機構へ拡張する。さらに、この時間保護機構をμITRON仕様準拠のTOPPERS/JSPカーネルと自動車制御システムで普及しているOSEK/VDX仕様準拠のTOPPERS/OSEKカーネルに統合することで、時間保護機能をもつRTOSを開発する。本プロジェクトでは、TOPPERS/JSPカーネルの開発で蓄積された実装技術及び、自動車業界からの要求事項を積極的に取り入れ、実用を念頭に置いた基盤的RTOSの開発を目指す。

3. 開発の内容

本プロジェクトでは、ハードリアルタイム性が要求される組込みシステムのアプリケーションを対象として、複数のアプリケーションを単一のCPUで動作させるための時間保護機能を考案する。さらに、それを既存のRTOSをベースに実装することを目標とする。具体的な開発内容は以下のとおりである。

時間保護機能の仕様検討

時間保護機能を実現するためにスケジューリングアルゴリズムが満たすべき要求事項を3つにまとめた。さらに、スケジューリングの対象タスクとして、従来の階層型スケジューラがタスクのリリース時刻と実行時間を要求していたのに対して、タスクのリリース時刻とデッドライン時刻のみで時間保護機能を実現する階層型スケジューラを考案した。そして、既存のRTOSへの実装を念頭においた時間保護仕様を検討した。

時間保護機能をもつμITRON4.0仕様準拠のリアルタイムOSの開発

TOPPERSプロジェクトから公開されているμITRON4.0仕様準拠のリアルタイムOSであるTOPPERS/JSP Release 1.4に、考案した時間保護機能を実装する。

開発の結果、TOPPERS/JSPをベースとした時間保護機能をもつRTOSを開発することができた。

時間保護機能をもつOSEK/VDX仕様準拠のリアルタイムOSの開発

TOPPERSプロジェクトから公開されているOSEK/VDX仕様準拠のリアルタイムOSであるTOPPERS/OSEK Release 1.0に、時間保護機能を実装する。

開発の結果、TOPPERS/OSEKをベースとした時間保護機能をもつRTOSを開発することができた。

機能検証及び評価

開発したTOPPERS/JSPベースのRTOSとTOPPERS/OSEKベースのRTOSの2つの

ソフトウェアについて、動作機能の検証と性能評価を行う。評価では、プログラムのソースコードの行数やオブジェクトファイルのサイズ、さらに、動作のオーバヘッドなどを測定する。

機能検証の結果、開発したソフトウェアの正常な動作を確認した。さらに、性能評価を行い、コード量やオーバヘッドの点で、実用的な RTOS を開発できたことを確認した。

4. 従来の技術(または機能)との相違

リアルタイム・アプリケーションを対象とした組込みシステム向けの時間保護機能をもつ RTOS に関しては、実用段階のものは未だに存在しておらず、研究段階においても仕様が提案されている程度である。関連する従来技術として、以下の 2 つを挙げる。

デッドラインモニタリング

HIS(Herstellerinitiative Software:欧州の主要自動車メーカ・グループ)により公開されている、OSEK/VDX仕様を拡張する時間保護機能仕様では、実行状態にあるタスクがデッドラインを越えて実行されることを監視するモニタリング機能のサポートに留まり、十分な保護機能が実現されているとはいえない。すなわち、デッドラインモニタリング機能ではタスクのデッドラインを保護することは可能であるが、タスクの実質的なプロセッサ時間を保護することは困難である。その結果、あるタスクに過剰なプロセッサ時間を割り当ててしまい、無駄なプロセッサ時間が発生する可能性がある。本プロジェクトで開発したRTOSは、タスクが実質的に使用するプロセッサの時間を管理し、時間保護機能を実現する。

CPUリソース管理

PC向けOSの分野では、CPU実行時間をPCが持つリソースの一つとして考え、指定したCPU時間をタスクまたはプロセス毎に割り当てるリソース管理機能が提案されている。これらリソース管理機能の多くが対象とするのは、非リアルタイム・アプリケーションであるため、ハードリアルタイム・アプリケーションを扱う組込みシステムへ適用することは困難である。主な理由として、システムの応答性確保とハードリアルタイム性の保証が困難であることなどが挙げられる。本プロジェクトでは、 μ ITRONやOSEK/VDX準拠のRTOSをベースに開発するため、ハードリアルタイムシステムにも適用可能である。

5. 期待される効果

本プロジェクトでは、現在提案されているアルゴリズムを、実装を念頭において拡張し、 μ ITRON 仕様準拠の TOPPERS/JSP カーネルと自動車制御システムで普及している OSEK/VDX 仕様準拠の TOPPERS/OSEK カーネルに実装した。その際、TOPPERS/JSP カーネルの開発で蓄積された実装技術及び、自動車業界からの要求事項を積極的に取り入れながら開発を行った。今後、自動車業界を中心として、開発したソフトウェアを提案していくことにより、産業界に有用な選択肢を与えることが出来ると考えている。

自動車制御システムへの適用

自動車制御システムに時間保護を適用することにより、ECU と ECU 間通信のネットワークケーブル(ハーネス)の数を削減することが可能になる。例えば、エンジン制御用 ECU と AT 制御用 ECU などをもつ ECU に置き換えることが考えられる(図 3)。自動車制御の分野では、ECU 数の削減や制御アプリケーションの流動性向上を目的として、自動車メーカ、自動車部品メーカ、半導体メーカなどが中心となり、ソフトウェアプラットフォームの標準化を検討している。例えば、欧州では AUTOSAR、国内では JasPar などの団体がある。提案する時間保護は、アプリケーションの統合に必須の機能であることを述べた。そこで、時間保護機能を自動車制御システムに適用することが十分可能であると考えている。現段階では、自動車メーカとの共同研究により、本ソフトウェアに対する需要を確認している。今後はさらに、自動車制御システムへの具体的な適用を検討していきたいと考えている。

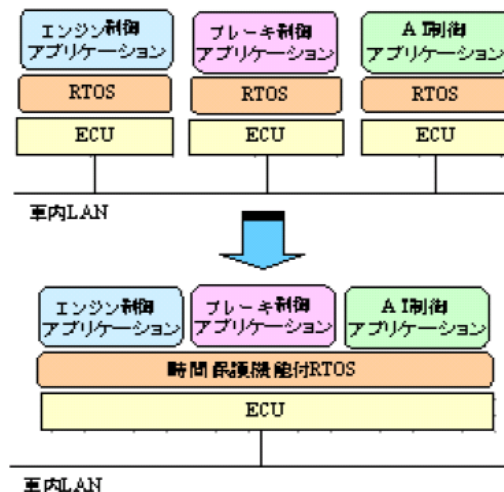


図 3: 自動車制御システムにおける ECU の統合

6. 普及(または活用)の見通し

現在、自動車メーカとの共同研究により、本ソフトウェアの有用性が確認されている。今後は、JasPar を通して他の自動車メーカや車載ソフトウェアメーカに対して本ソフトウェアを提案していく。また、実際に使用されているアプリケーションを用いた評価を検討していく。また、2006 年 3 月には TOPPERS プロジェクトの技術検討会議にて、本ソフトウェアの有用性を多方面の技術者と検討する。さらに、ソースコードレビューを行って完成度を向上させ、オープンソースソフトウェアとしての早期公開を目指す。

7. 開発者名(所属)

松原豊(名古屋大学大学院情報科学研究科)

(参考)開発者 URL: <http://www.ertl.jp/~yutaka/>