

リアルタイム処理に適した密結合メモリの利用法

劉 遠哲 (北陸先端大)・請園 智玲 (北陸先端大)・田中 清史 (北陸先端大)

1 はじめに

一部の ARM プロセッサ [1] は“密結合メモリ”(TCM: Tightly-Coupled Memory [2]) を備える。TCM は命令実行パイプラインの近傍に配置される SRAM である。TCM 内に命令及びデータを配置することで、命令実行パイプラインへの命令/データの供給バンド幅を向上させ、高速なプログラム実行を可能とすることができる。このことから、TCM は一般的なキャッシュメモリと同様の目的を持った機能と言える。TCM とキャッシュメモリの大きな違いは格納するデータの管理ポリシーである。TCM はソフトウェアで格納するデータの選別・配置制御を行う機能と通常のキャッシュメモリと全く同様の管理ポリシーで動作するスマートキャッシュと呼ばれる機能の両方を有する。本稿では、組み込みシステムのリアルタイム性向上のために、ソフトウェアで格納データを管理する TCM の機能を利用する手法を検討する。

本研究の最終的な目標はこの機能をリアルタイム性の向上に利用することである。リアルタイムシステムにおいて、各タスクは終了しなければならない締め切り時刻(デッドライン)を持ち、例えば各タスクが完了したとしても、デッドラインが守られなければ、そのタスク実行の価値が下がると評価される。このため、リアルタイム OS では、汎用の OS とは異なり、リソース割り当てをタスク間で公平に行うことを良しとせず、できるだけデッドラインを守れるように行われる。この時に問題となるのが、タスクの実行時間予測である。通常のキャッシュの管理ポリシーでは、どのような順序で各タスクが呼び出されるかでキャッシュの効果が変動し、各タスクの実行時間の見積もりを難しくしている。一方、TCM はマッピングをソフトウェアでコントロール可能であるため実行時間の変動を抑制することができる。

2 TCM のデータ管理手法の概要

スマートキャッシュ制御を行わない TCM はメモリマップド I/O を用いて、主記憶を TCM 内の区画単位(バンク)に対応させる。対応した主記憶上のデータは DMA 転送機能を用いてプログラム実行のバックグラウンドでロードすることが可能である。本来、この機能は予めソフトウェアの設計段階で高速応答が要求されることが判っている割り込みハンドラ等をマップするために利用される。本研究では、この TCM マッピングをリアルタイムタスクのワーキングセットに適用し、タスクのリアルタイム性向上を狙う。

3 リアルタイムシステムへの適用

本研究の提案は 2 つある。1 つ目は TCM マッピングの細粒度化、2 つ目はリアルタイム OS による TCM 制御である。TCM マッピングの細粒度化に関する提案は単純である。ARM926EJ-S [3] の実装では 1 バンク 4KB で構成されておりマッピングは 4KB 以下を許すが、この 1 バンクを複数のマッピングで共有することはできない。本研究ではリアルタイム OS が動的に TCM リソースの割り当てを行う際に無駄なくタスクのワーキングセットをマップできるように、更に小さい粒度のバンクで実装される TCM を提案する。

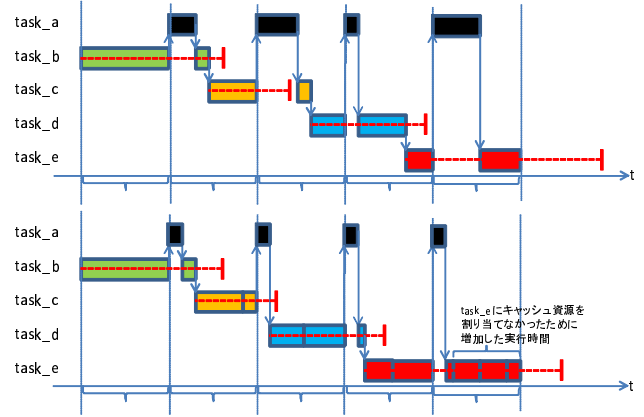


Figure 1: TCM によるリアルタイム性の向上の例

2 つ目のリアルタイム OS による TCM 制御の適用例を図 1 に示す。図には 2 つのタイムライン(横軸)があり、上が TCM 制御無し、下が TCM 制御ありの場合のタスクの実行経過が示される。縦軸の位置は task_a ~ task_e の 5 つのタスクが示され、高い位置ほど優先度が高い。また、task_a のみ周期タスクであり、同じ周期で繰り返し起動される。それぞれのタスクの実行を示す矩形を貫通する破線で T 字赤い線はデッドライン時刻を示している。本研究の目的は TCM によるタスクの実行時間変動抑制である。上図と下図の違いは、上図ではタスクはキャッシュで実行されており、下の図は task_a のワーキングセットに TCM バンクを動的に割り当て、その代償として task_e にキャッシュ資源を割り当てていない。上図では task_c がデッドラインミスしているのに対し、下図では全てのタスクでデッドラインミスが見当たらない。これは、リアルタイム OS がタスクスケジューリングに大きく影響のある task_a の実行時間変動を抑え、余裕時間の多い task_e の実行時間を故意に延ばす資源割り当てを行った事により発生している。本研究では TCM 制御によりこのような例のデッドラインミス回避が可能なリアルタイム OS を開発している。

4 終わりに

本稿では 2 つの提案を行った 1 つ目は ARM に実装される TCM マッピングの細粒度化、2 つ目はタスクのリアルタイム性向上のために TCM の資源管理をするリアルタイム OS の提案である。提案する OS は現在開発中であり、完成次第、1 つ目の提案を組み込んだ ARM プロセッサシミュレータ上で開発制作した OS のリアルタイム性能を計測する予定である。

References

- [1] ARM Architecture Reference Manual, ARM Ltd, 2005.
- [2] Jason R. Andrews, “Co-Verification of Hardware and Software for ARM SoC Design”, Newnes, 2005.
- [3] Technical Reference Manual, Revision r0p5, ARM Ltd, 2008.