

OS とコンパイラとの連携による PG 制御方式の研究

小林 弘明^{†1} 茂木 勇^{†1} 木村 一樹^{†1} 薦田 登志矢^{†2}

佐藤未来子^{†1} 近藤 正章^{†3} 中村 宏^{†2} 並木 美太郎^{†1}

^{†1}東京農工大学 ^{†2}東京大学 ^{†3}電気通信大学

1. 緒言

近年、微細化技術の向上により、システム LSI の回路全体の消費電力に占めるリーク電力の割合が増大している。現在、筆者らは、回路への電源電圧の供給を遮断することでリーク電力を低減するパワーゲーティング(Power Gating, 以下 PG)技術を搭載した CPU コア Geysers の研究を進めている。

Geysers は、回路への電源供給の持続/遮断を制御する PG 制御命令を備えており、コンパイラの静的解析の観点から、これを利用して効果的に PG 制御を行う研究[1]を共同研究チームで進めている。本技術は、特定のコア温度において効果的に PG 制御を行う命令列を生成するもので、これまでの研究でその有効性を確認している。しかし、実行時のコア温度の変動によりその効果も大きく変動してしまう。

本研究では、この技術に OS の支援を加えることで、実行時のコア温度変化の問題に対処する新たな PG 制御方式を提案する。

2. Geysers アーキテクチャ

Geysers は、MIPS R3000 アーキテクチャをベースとした CPU コアである。独自仕様として、MIPS の R 形式命令を拡張した PG 制御命令(図1)を備えており、上位 6bit の PG 制御フィールドの値に応じて対象演算ユニットへの電源供給の持続/遮断を制御する。

3. 研究目標

コンパイラの静的解析技術では、プログラムをコンパイル時に解析し、各演算ユニットのアイドルサイク

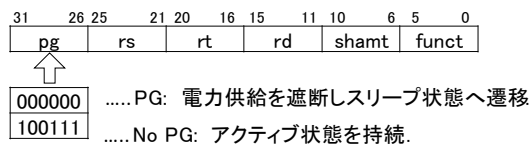


図 1 PG 制御命令

A Study on Power Gating Control with the Cooperation of Operating System and Compiler

Hiroaki KOBAYASHI^{†1}, Isamu MOGI^{†1}, Kazuki KIMURA^{†1}, Toshiya KOMODA^{†2}, Mikiko SATO^{†1}, Masaaki KONDO^{†3}, Hiroshi NAKAMURA^{†2}, Mitaro NAMIKI^{†1}

^{†1} Tokyo University of Agriculture and Technology

^{†2} The University of Electro-Communications

^{†3} The University of Tokyo

ルを予測する。この情報をもとに、各命令実行後の対象演算器への電源供給の持続/遮断の制御情報を PG 制御フィールドに記述し、特定のコア温度向けに最適化した命令列を生成する。[1]

本研究では、特定のコア温度用に最適化した命令列を各温度用に複数用意し、これを OS が動的に管理し、温度変化に応じて適切な命令列を選択、実行する新たな PG 制御方式を提案する。提案方式により、PG 制御のネックとなるコア温度変化に対応し、消費電力低減効果を向上させることを目標とする。

4. 設計

4.1. 温度対応オブジェクトファイル

主要なフォーマットの 1 つである ELF をベースに、提案方式で利用するオブジェクトファイルを図2のように 2 段階の処理により生成する。1 段階目では、前述のコンパイラを用いて特定のコア温度に最適なオブジェクトコードを生成する。2 段階目では、1 段階目で生成した複数のオブジェクトコードを、重複の無いように 1 つにまとめる。この際、温度ごとに text セクションを分け、各 text セクションに個別の text セグメントを対応付ける。また、ヘッダに記述する各 text セクション/セグメントの仮想アドレスをすべて等しくすることで、実行時のテキストセグメント切り替えによるプログラムへの影響を排除し、一貫性を保証する。

4.2. 実行時オブジェクトコード管理機構

前節で設計した温度対応オブジェクトファイルを管理し、実行時のコア温度変化に応じて text セグメントを切り替える管理機構を OS に加える。本管理機構

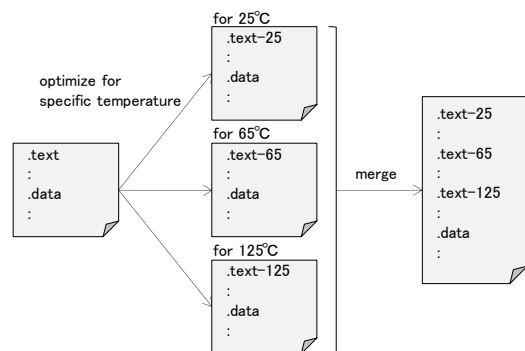


図 2 温度対応 ELF の生成

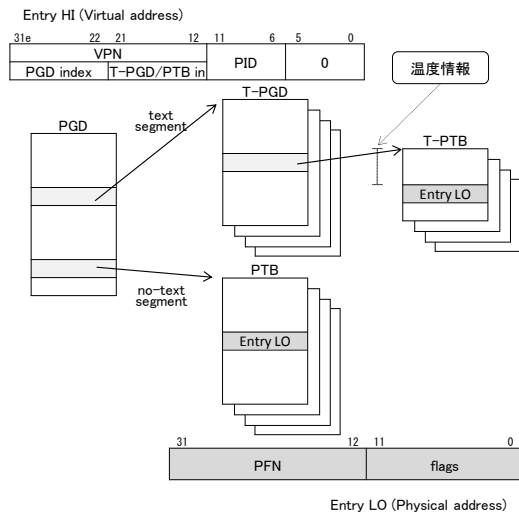


図3 ページテーブルの構成

では, text セグメントを多重化するために図3に示す特殊なページテーブルを利用する. 本ページテーブルは, text セグメントを3段, その他セグメントを2段で構成する. text セグメントのページテーブルの最下層は, 温度情報をインデックスとする配列である. OS は, これを用い, コア温度変化に応じて仮想-物理マッピングを変更し, 実行する命令列を切り替える.

5. 実装と評価

プロジェクト内で研究開発を進めている Geysler OS をベースに, 提案制御機構を実装する. 実行/評価環境としては, Geysler2-on-FPGA を利用する. 本環境は, PG パフォーマンスカウンタを備えており, 各演算ユニットのスリープ/アクティブサイクル情報を取得できる. ベンチマークは, matrix, qsort, dhrystone, 及び dijkstra を使用し, 電力評価には, 文献[2]で提案されている電力計算プログラムを利用する. 本環境では, コア温度を計測することができないため, 1~130 の範囲でランダムにコア温度情報を与える.

- i. ハードウェアの自律制御のみ
- ii. コンパイラの静的解析によるコードのみ
- iii. コンパイラの静的解析によるコード + OS 支援(提案方式)

パターン i ~ iii における alu, shift ユニットの平均

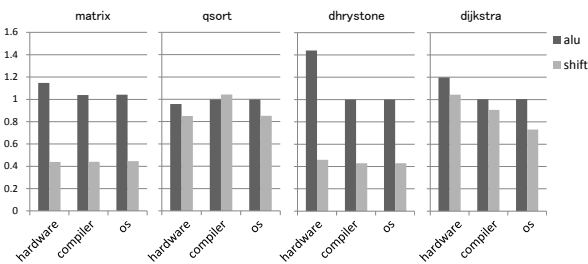


図4 ベンチマークごとの平均リーク電力

リーク電力を図4に示す. 縦軸は, PG なしの場合を1としたときの演算ユニットごとの相対的なリーク電力である. i ~ iii のすべてにおいて, グラフに示していない mult, div を含め回路全体で, リーク電力の低減効果を確認した.

i と ii とを比較すると, コンパイラの静的解析により平均 6%の低減効果がある. qsort において, リーク電力が増大しているが, これはコア温度変化により損益分岐点が変わったことに起因すると考えられる.

ii と iii とを比較すると, OS の支援により平均 3%の低減効果がある. qsort のようにコンパイラの静的解析が逆効果に働いた場合も, OS の支援によりその影響を最小限に抑えていることが確認できる. 本方式は, 効果にばらつきがあり, その有効性がプログラムの挙動特性に依存すると考えられるため, 挙動特性に応じて制御方式を変更する等の改良により, さらに低減効果を高めることができると考えられる.

6. 結言

本稿では, PG 技術を搭載した CPU コアを対象に, 消費電力低減効果を向上させることを目的とした新たな PG 制御方式を提案した. 提案方式は, 静的解析技術に OS の支援を加えることで, PG 制御のネックとなるコア温度変化に柔軟に対応する. Geysler OS をベースに構築した PG 制御機構は, ハードウェア単体と比較して平均 10%, コンパイラ単体と比較して平均 3%のリーク電力を削減できることをベンチマークテストにより確認した.

今後の課題としては, プログラムの挙動特性に応じて制御方式を切り替えるハイブリット方式化等が考えられる. また, プログラムの挙動特性と電力低減効果との関係を分析するために, より多くのベンチマークによる評価が必要である.

謝辞 本研究は, 科学技術振興機構「JST」の戦略的創造研究推進事業「CREST」における研究領域「革新的電源制御による次世代超低消費電力高性能システムLSIの研究」によるものである.

参考文献

[1] 薦田登志矢, 佐々木宏, 近藤正章, 中村宏, ”リーク電力削減のためのコンパイラによる細粒度スリープ制御”, SACSIS 2009, pp.11-18, May 2009.
 [2] 中田充貴, 白井利明, 香嶋俊裕, 武田清大, 宇佐美公良, 関直臣, 長谷川揚平, 天野英晴, ”ランタイムパワーゲーティングを適用した回路での検証環境と電力見積もり手法の構築”, 信学技報, vol.107, no.414, VLD2007-111, pp.37-42, Jan 2008.