

May 28, 2020

共有資源の競合を考慮したチップマルチプロセッサ向け低消費電力化手法

沼田 賢一

富山県立大学 情報基盤工学講座

1. はじめに
2. 提案手法
3. おわりに

May 28, 2020

本研究の背景

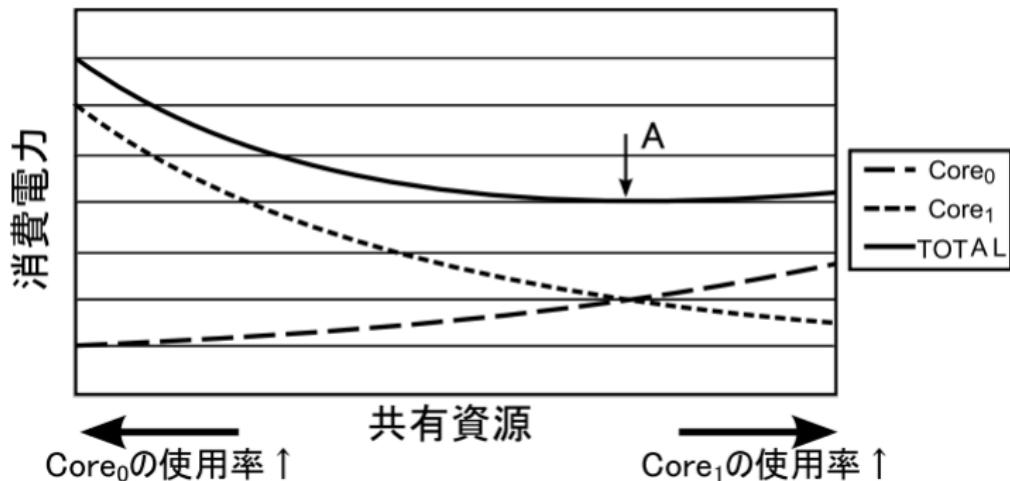
近年, VLSI (集積回路) は高機能化・高性能化に伴い消費電力の増加が引き起こされている. その解決策として動的電源電圧制御 (DVFS) というものがある.

ただし, ユニプロセッサからマルチプロセッサが搭載されるようになると複数コアがキャッシュやメモリを共有するため (共有資源), 競合を引き起こすことがある. そうなると実行中プログラムの性能予測が困難となるが, その中で消費電力を抑える必要がある.

目的

共有資源の競合状況を調整したうえで DVFS を行うことで消費電力の削減をする手法を提案する.

提案手法



共有資源の単位時間あたりの使用率と消費電力の関係
アクセス制御により共有資源の使用率を適切に調整することで消費電力を最小化することができる。

提案手法

4/16

メモリとキャッシュそれぞれのアクセス制御による消費電力の関係をモデル化する。

このモデルから最小化されたアクセス制御をマルチプロセッサ上で達成するような協調制御手法を構築し有効性を確認する。

- ① 全タスクの命令レベル並列度は等しい.
- ② ライトバックによるメモリバスの占有は無視できる.
- ③ 主記憶 DRAM バンクは競合しない.
- ④ 主記憶アクセスのレイテンシは一定であり, 主記憶アクセスの順番には依存しない.
- ⑤ 優先度を変えることでバスにおける待ち時間を各コアに任意に割り振れる.
- ⑥ Taski 内では L2 キャッシュミス率は一定である.
- ⑦ f_i , V_i は任意の連続値をとれる.

アクセス制御時の消費電力のモデリング

6/16

はじめに
提案手法
実験
おわりに

変数	説明
n	コア数
f_i	$Core_i$ のクロック周波数
V_i	$Core_i$ の電源電圧
P_i	$Core_i$ の消費電力
P_{total}	CMP 全体の消費電力
e_i	$Core_i$ の 1 命令あたりの消費エネルギー
T_i	$Task_i$ の実行時間
L_i	$Task_i$ の実時間制約
l_B	1 回のメモリアクセスがメモリバスを占有する時間
l_M	メモリアクセスレイテンシ
I_i	$Task_i$ における実行命令数
m_i	$Task_i$ における L2 キャッシュミス回数
s_i	L2 キャッシュミスによる生じる $Core_i$ のストール時間
t_i	$Core_i$ の実効稼働時間
a, b, k	CMP に依存する定数

アクセス制御時の消費電力のモデリング

7/16

これらの変数を用いると、クロック周波数と電源電圧、マルチプロセッサ全体の消費電力、メモリアクセスのストール時間 w が計算できます

$$f_i = c \frac{I_i}{t_i} \quad (1)$$

$$V_i = af_i + b \quad (2)$$

$$\begin{aligned} P_{total} &= \sum_i P_i \\ &= \sum_i \frac{I_i}{T_i} e_i = \sum_i \frac{k I_i V_i^2}{T_i} \end{aligned} \quad (3)$$

$$w_i = m_i l_M \quad (4)$$

7/16

式 (1)(2)(3)(4) を用いるとアクセス制御の適用時のクロック周波数と消費電力を求めることができる

$$\begin{aligned}
 f_i &= c \frac{I_i}{L_i - (w_i + r_i l_{total} L_i)} \\
 &= c \frac{I_i}{L_i - (m_i l_M + r_i l_{total} L_i)} \quad (5)
 \end{aligned}$$

$$P_{total} = \sum_i \frac{I_i}{L_i} (C_2 f_i^2 + C_1 f_i + C_0) \quad (6)$$

よって、 P_{total} を最小化することで、電力最適点が求まる。

モデルから最適な電力を計算し制御できるように構成したものを使い実験を行う。

用いるハードウェアは、主記憶装置である DRAM バンクを共有資源としないモデル (IDEAL) と共有するモデル (REAL) を用いる。

	<i>Env: IDEAL</i>	<i>Env: REAL</i>
Cores		2
Clock frequency	800–2,400 MHz (200 MHz step)	
Supply voltage	0.988–1.441 V	
L1-Inst (private)	32 KB, 2-way, 32 B line, 1-cycle	
L1-Data (private)	32 KB, 2-way, 32 B line, 2-cycle	
L2 (unified)	1 MB, 16-way, 64 B line, 10-cycle	
Memory bank	4 bank 10-bus cycle	4 bank Row hit: 4-bus cycle Row closed: 8-bus cycle Row conflict: 16-bus cycle
Memory bus	8 B, 400 MHz	
DVFS penalty		10 us

ハードウェアの構成

また、提案手法の有効性を評価するために以下の4つの手法を用いて2コア及び4コアで実験を行う。

- ① DVFS only : 既存手法. DVFS 制御のみ適用.
- ② priority : DVFS 制御と優先度制御を適用.
- ③ DVFS 制御とキャッシュパーティショニング制御を適用.
- ④ priority+partition : 提案手法. DVFS 制御, 優先度制御, およびキャッシュパーティショニング制御を適用.

実験結果

11/16

2コア時の各アプリケーションごとの IDEAL の消費電力の結果

はじめに
提案手法
実験
おわりに

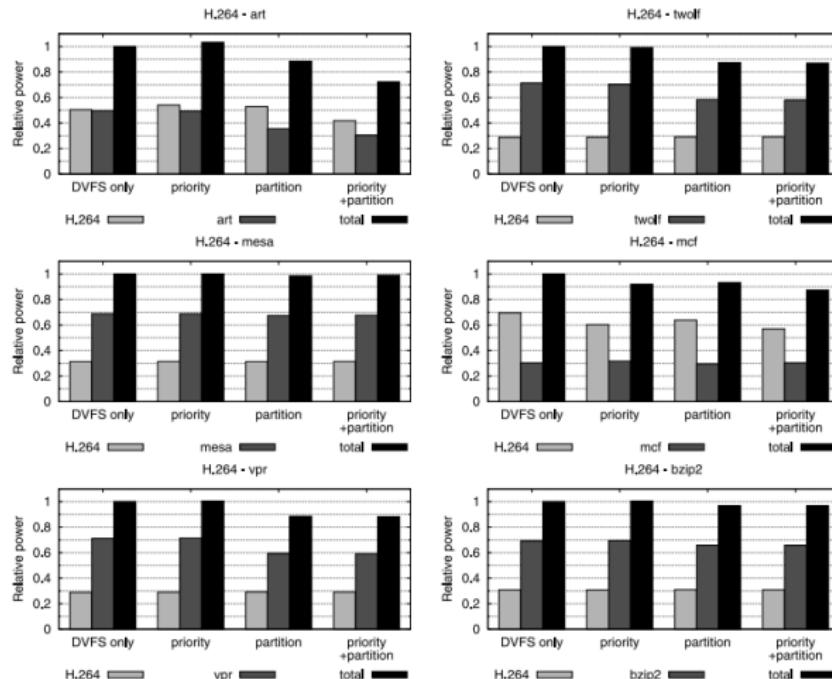


図 9 消費電力 (IDEAL, 2コア)

実験結果

12/16

2コア時の各アプリケーションごとのREALの消費電力の結果

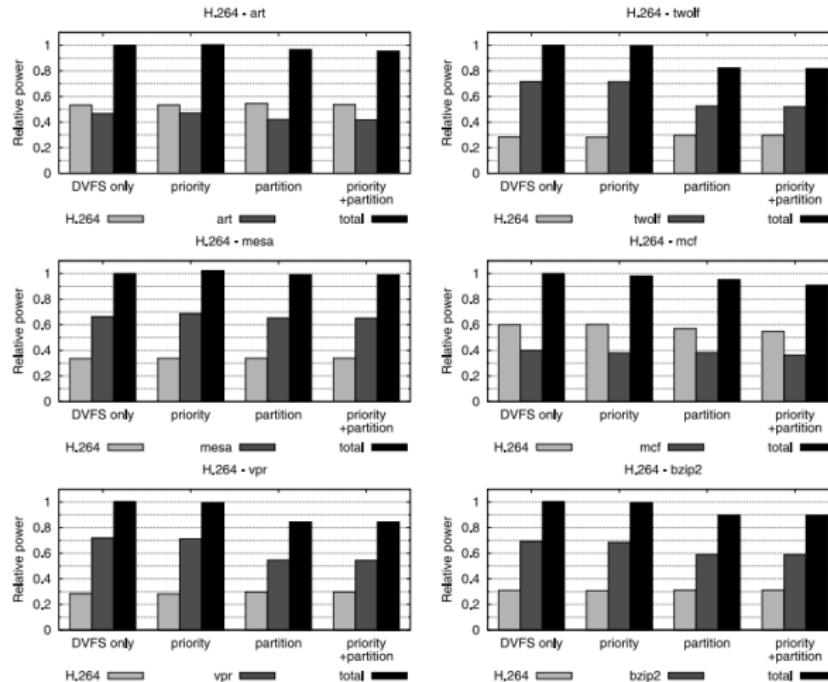


図 10 消費電力 (REAL, 2コア)

実験結果

4コア時の IDEAL と REAL の消費電力の結果

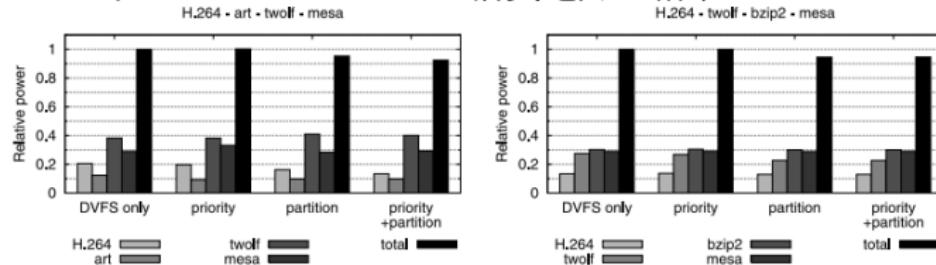


図 11 消費電力 (IDEAL, 4コア)

Fig. 11 Relative power consumption (IDEAL, quad core).

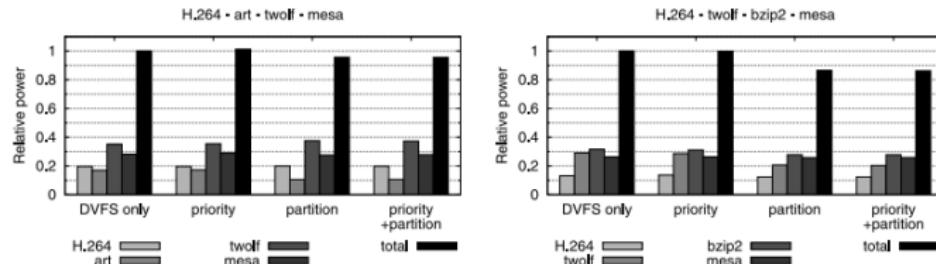


図 12 消費電力 (REAL, 4コア)

実験結果

14/16

はじめに
提案手法
実験
おわりに

2コア時 IDEAL では、全アプリケーションで既存手法よりもアクセス制御を用いた方が消費電力を削減することができている。(最大 28 %, 平均 12 % の削減) そして、提案手法が最も消費電力を削減できた。

2コア時の REAL でも同様に全アプリケーションにおいて消費電力の削減に成功した。(最大 19 %, 平均 10 % の削減)

4コアでも、IDEAL と REAL において両方とも提案手法が最も消費電力の削減に成功した。(平均 8 % と平均 8.5 %)

14/16

考察

15/16

理想的な環境（IDEAL）と現実的な環境（REAL）のどちらにおいても提案手法が有効であることが示すことができた。

ただし、現実的な環境では主記憶 DRAM の競合とアクセスレイテンシがあるため、消費電力の削減率が理想的な環境よりも差がでてしまうこともわかった。

はじめに
提案手法
実験
おわりに

15/16

まとめ

- ① アクセス制御をすることでマルチプロセッサの消費電力の削減する手法を提案し有効であることが確かめることができた。

今後の課題

- ① 主記憶 DRAM バンクの競合も考慮したモデルへの拡張が必要。