九州大学学術情報リポジトリ
Kyushu University Institutional Repository

メモリ・ウォール問題への挑戦

三輪, 英樹
九州大学大学院システム情報科学府

https://hdl.handle.net/2324/9136

出版情報：SLRC プレゼンテーション, 2006-07-19. 九州大学システムLSI研究センター
バージョン：
権利関係：

KYUSHU UNIVERSITY

NGArch Forum 2006

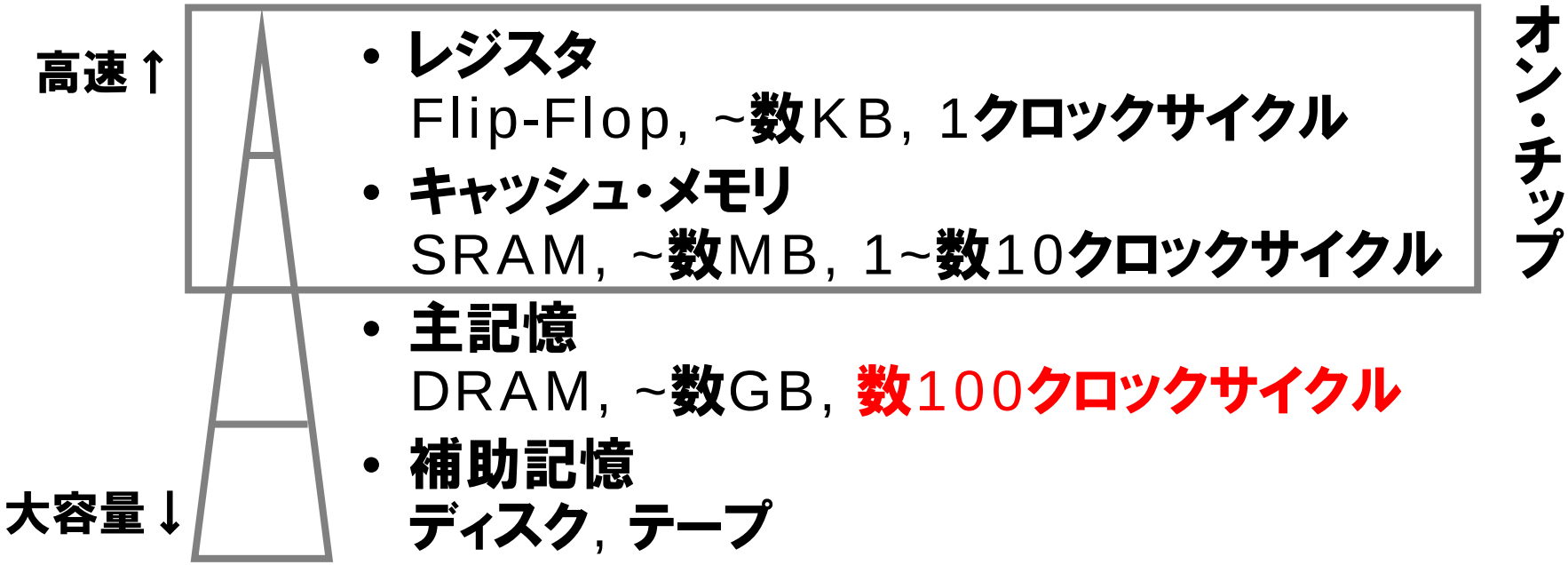
メモリ・ウォール問題への挑戦

三輪英樹
九州大学 大学院システム情報科学府
miwa@c.csce.kyushu-u.ac.jp

メモリ・ウォール問題とは?

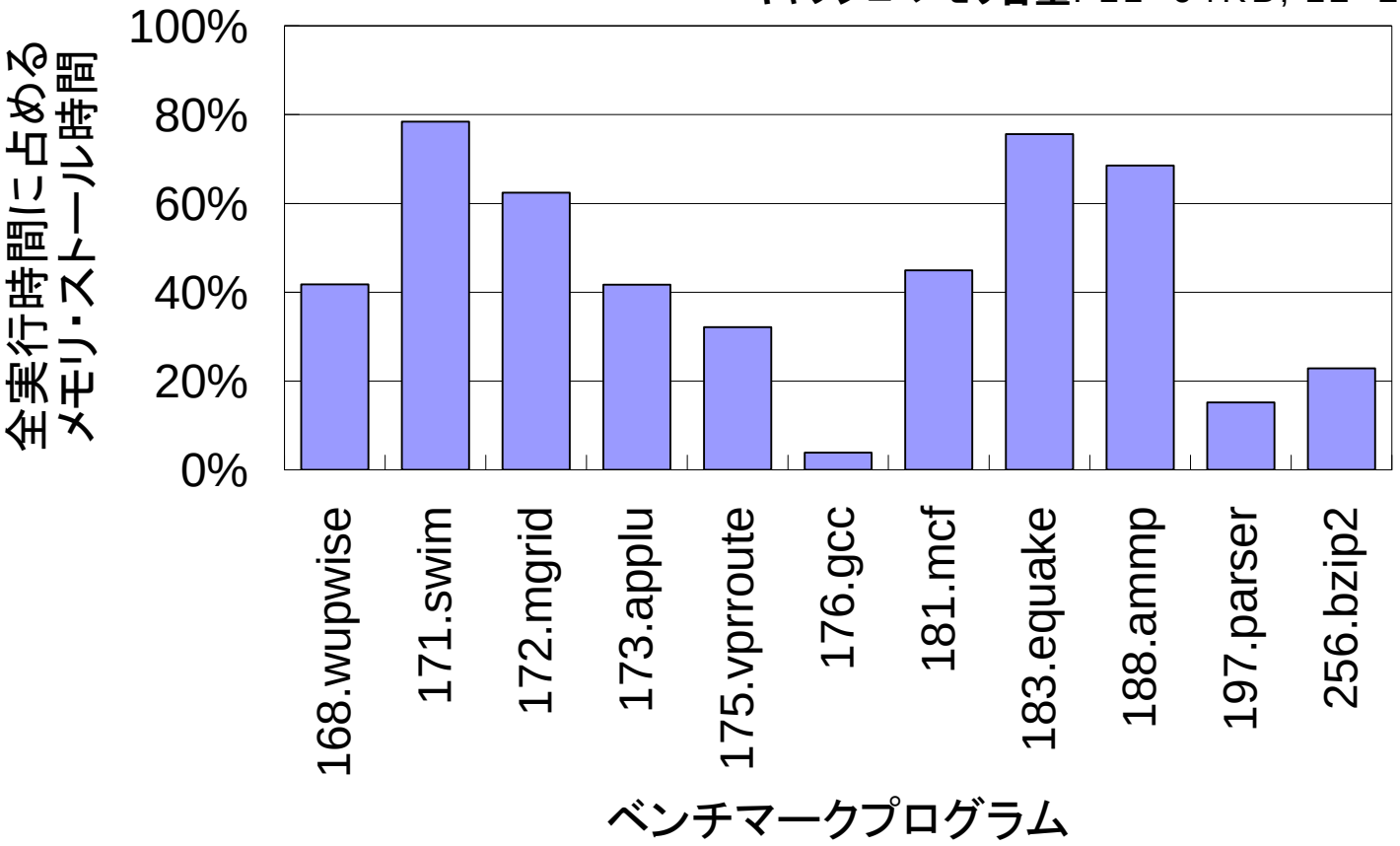
- 計算機の性能ボトルネック = メモリ (主記憶)

階層メモリ構造:「高速かつ大容量の記憶の実現」



メモリ・ストール時間の割合

出典: 筆者らによるシミュレーション実験
キャッシュ・メモリ容量: L1=64KB, L2=2MB



キャッシュ・ミスによる性能低下を抑えれば大幅に性能向上

キャッシュ・ミスはどのように発生しているのか?

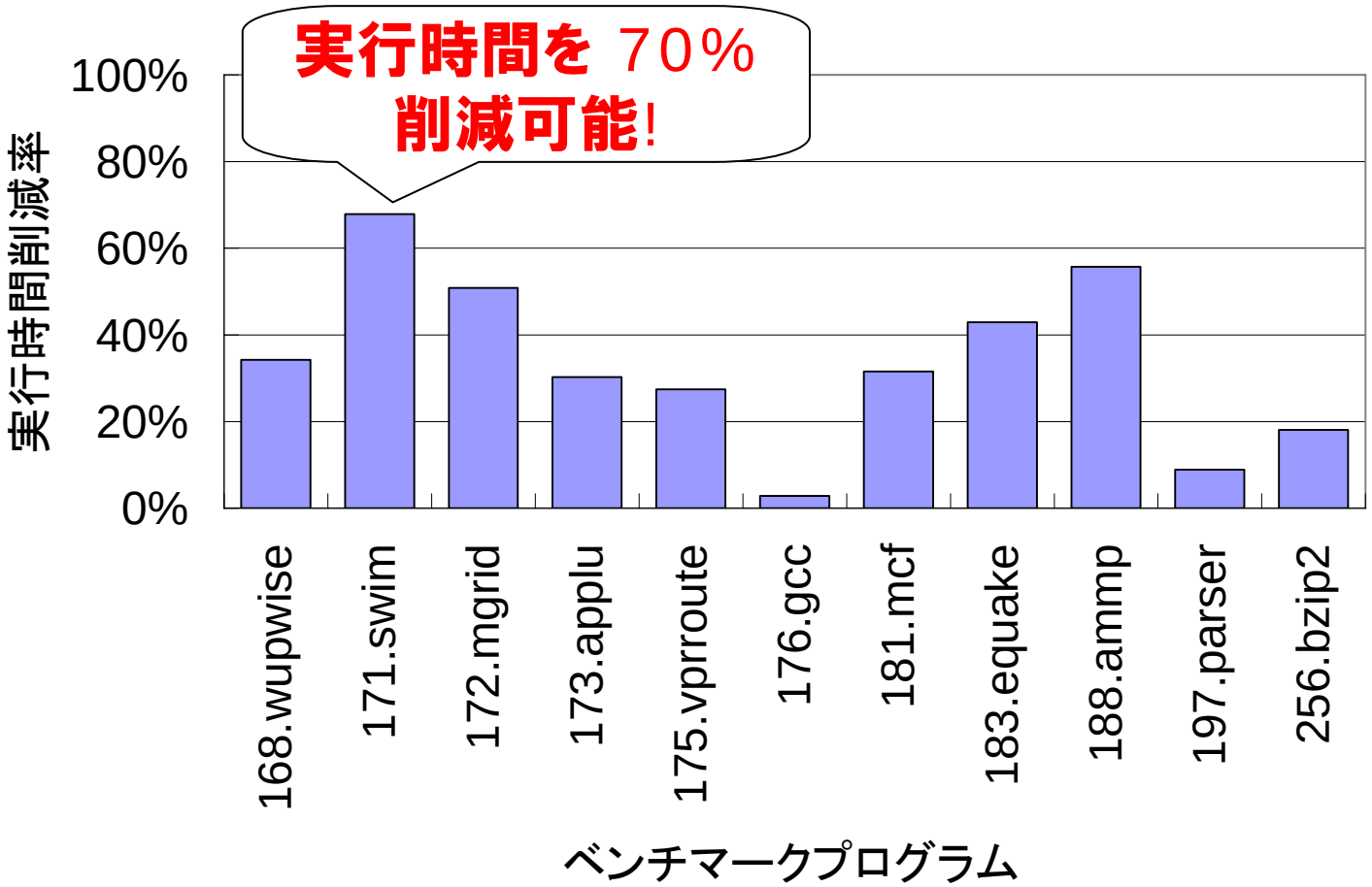
- キャッシュ・ミスの 80% ~ 90% は特定の **数10個のロード命令** により発生(少人数の犯人グループが繰り返し犯行)

↓

キャッシュ・ミスを頻発させるロード命令

(Delinquent Load 命令; DL命令)

DL命令を全て退治できたら?



プログラム実行速度を最大3倍に高速化可能!

DL命令の退治を狙った主な研究

- 投機的にロード対象データアドレスを計算し，プリフェッチ
  - Speculative Pre-computation [J.P.Shen et al. 2001]
  - Data-driven Multithreading [G.Sohi et al. 2001]
- キャッシュ・ミス時に後続命令の実行を可能にする機構を搭載
  - Run-ahead Execution [Y.Patt et al. 2003]
  - Kilo-instruction Processor [M.Valero et al. 2004]
- 頻繁に実行されるループ内にプリフェッチ命令を動的に追加
  - Self-Repairing Prefetcher [D. M. Tullsen et al. 2006]
- コンパイル時に DL 命令を特定
  - Static Identification [W.Wong et al. 2004]
- ロード対象データを再計算
  - Computing Centric Computation [K.J.M. et al. 2004]

DL命令をどのように“退治”するか?

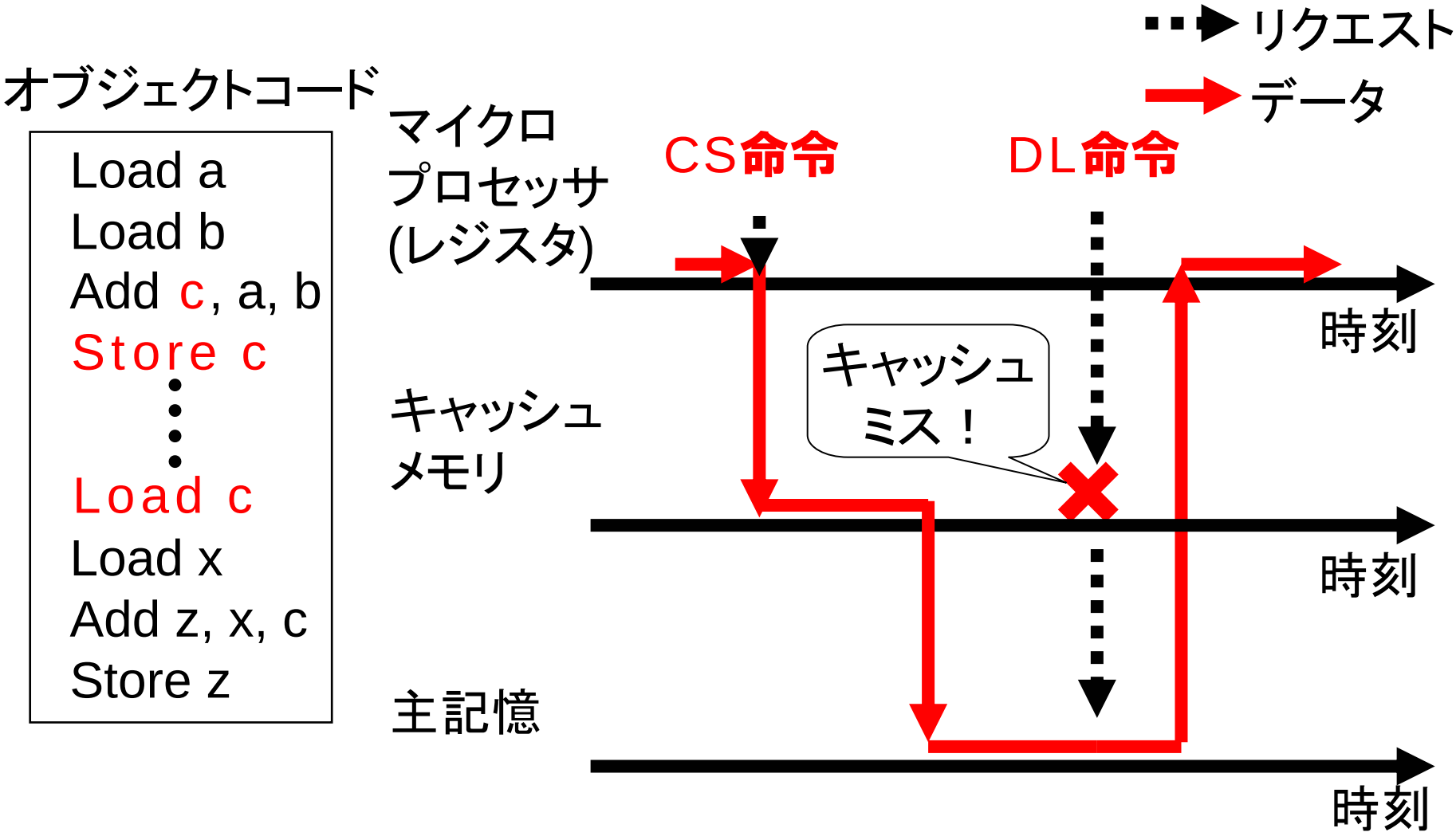
- ロード対象データをストアしている命令が存在
  (犯人グループと密接な関係にある黒幕)

↓

DL命令に対応するストア命令

(Corresponding Store命令; CS命令)

DL命令とCS命令との関係



実際の処理との関係

ベンチマークプログラム: 179.art (**画像認識処理**)

scanner.c

```
...

192: temp = bus[j][i];
193: bus[j][i] += ...
...
        (2重ループ中)
476: Y[tj].j += f1_layer[ti].P * bus[ti][tj];
...
```

DL**命令** (Top7, **ミス**:88343**回**, 1%)

CS**命令** (65% **の キャッシュミスに関与**)

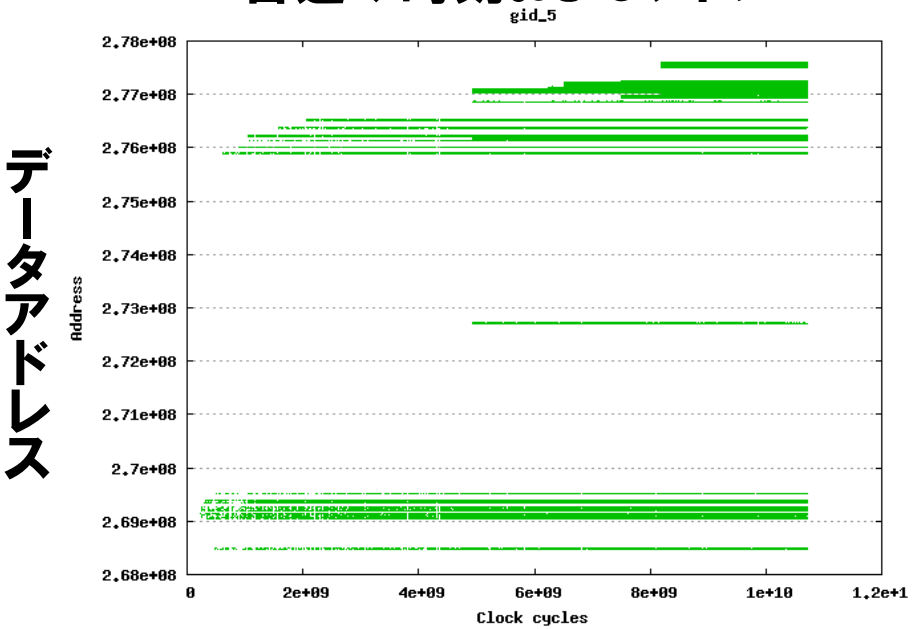
DL**命令** (Top1, **ミス**:5199928**回**, 64%)

(**全**CS**命令の発見回数**: 8051663)

アクセス時刻とアドレス

ベンチマークプログラム: 175.vpr_route (FPGAの配線処理)

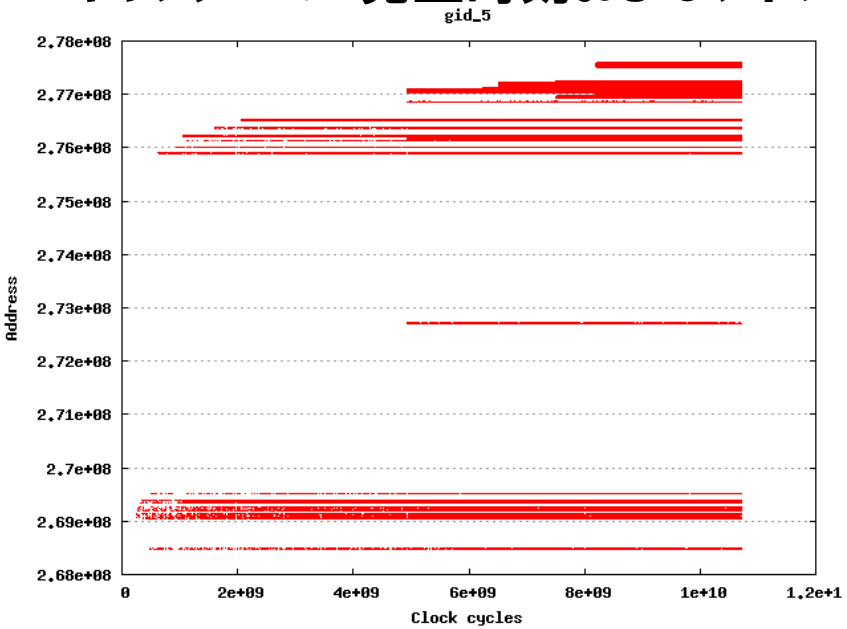
CS命令による
書込み時刻およびアドレス



データアドレス

データ書き込み時刻

DL命令による
キャッシュミス発生時刻およびアドレス

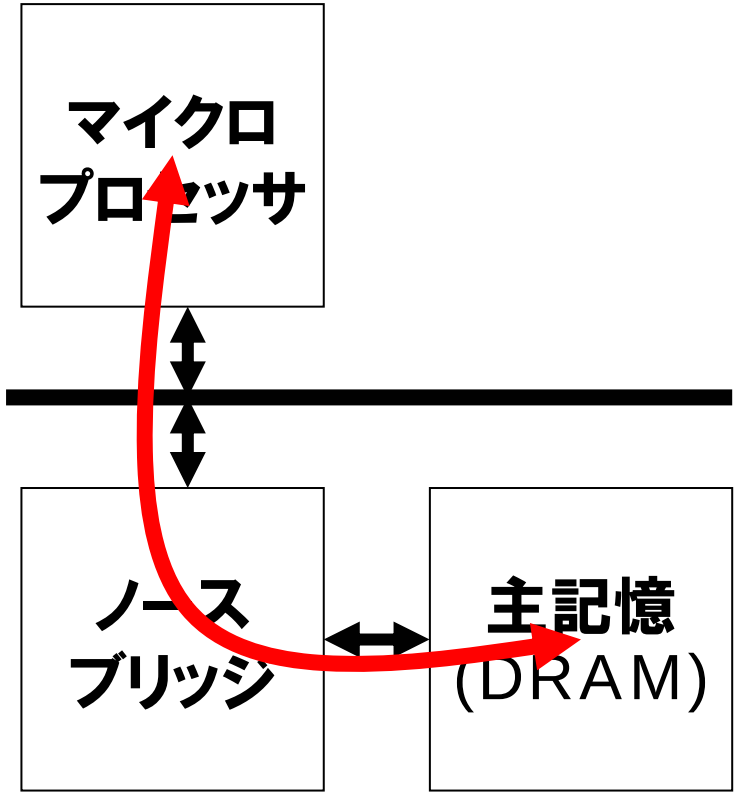


キャッシュミス発生時刻

書き込み~読み出し間隔 = 約4万クロックサイクル

検討中の対策手法

- メモリ・サイドプリフェッチの利用
  - CS命令実行時: 主記憶に設けた専用領域にデータを記憶.
  - DL命令実行時: 専用領域に記憶したデータをマイクロプロセッサに対して送付.



まとめ

- 計算機における性能ボトルネック ＝ メモリ
- メモリの性能を上がれば，計算機の性能は大幅に改善できる!

- 主な性能低下要因 (キャッシュ・ミスによる主記憶へのアクセス) の特徴を考慮し，メモリの利用効率を動的に向上させる技術を提案・評価