

論文の内容の要旨

論文題目 大規模データベース・プロセッサの研究

氏名 安島 雄一郎

1971年に最初のマイクロプロセッサ4004が誕生して以来、マイクロプロセッサはその製造技術、設計技術とも絶間なく改良されてきた。これによる性能向上は留まるところを知らず、未だに衰える傾向がない。マイクロプロセッサの急速な性能向上は計算機の高性能化、小型化に大きく貢献し、社会のデジタル化、IT化の原動力となっている。しかし、デジタル化によって様々なメディアがマイクロプロセッサで処理されるようになり、またネットワークの普及によってサーバーに膨大な処理が要求されるようになるなど、情報処理能力に対する要求の増大もまた留まるところを知らない。このため、情報処理の中核として、マイクロプロセッサには常により高い性能が求められている。

マイクロプロセッサの性能向上は、半導体デバイス技術とアーキテクチャ技術によって支えられてきた。デバイス技術の進歩によって高密度の集積が可能になり、利用できるトランジスタ数はより多く、トランジスタのスイッチング時間はより短くなってきた。スイッチング時間の短縮はプロセッサの動作速度の向上に貢献する一方、トランジスタ数の増加は新しいアーキテクチャ技術の導入を可能にする。この結果、パイプライン・アーキテクチャ、分岐予測による投機的実行、レジスタリ

ネーミング、アウトオブオーダー実行など、さまざまなアーキテクチャ技術が開発、実装された。

半導体デバイス技術は、半導体の集積度と性能は18ヶ月で2倍になるという Moore の法則を維持しており、米国半導体工業会によって今後も Moore の法則をに從った性能向上が続くと予想されている。一方、アーキテクチャ技術による性能向上は停滞している。現在主流のスーパースカラ・アーキテクチャでは、多数のトランジスタを使用して1サイクルに処理できる命令を増やそうとすると処理機構の複雑さが増大し、動作速度の低下を引き起こす。このため増加したトランジスタを有効に活用することができない。また、スーパースカラの構造の複雑さを改善する VLIW (Very Long Instruction Word) アーキテクチャもアーキテクチャ・コアの小型化には成功したものの、大規模化を進めると急速に柔軟性が失われるアーキテクチャであるため汎用プロセッサとしてスーパースカラ・アーキテクチャを超える性能を実現することはできなかった。スーパースカラ・アーキテクチャの限界は1命令ずつ処理する機構の複雑さに起因している。投機実行の範囲を広げることが困難な為、命令レベル並列性の解析範囲拡大や、近年提案されている分岐の複数パス実行、データ投機実行などの高度な投機技術による性能向上を得ることが難しい。

このような状況で、利用可能なトランジスタ数の増加を性能向上に結びつけるため、様々な新しいアーキテクチャ技術が研究されている。EPIC (Explicitly Parallel Instruction Computing) 技術とコードモーフィング技術は VLIW の柔軟性のなさを克服し、大規模実行を目指している。EPIC 技術は複数の VLIW 命令ワード間の並列性を明示的に指定しておくことにより、多数の VLIW 命令ワードを同時に実行することを可能にする。また、コードモーフィング技術は読みこんだコードを実行時に VLIW 命令に変換して実行する。これによりアーキテクチャ・コアにあわせた柔軟なコードを実行することが可能になり、実行規模の拡大も容易になる。しかしどちらも固有の複雑性があり、スーパースカラを大きく超える性能は実現できていない。CMP (Chip Multi-Processor) アーキテクチャは一つのチップにスーパースカラもしくは VLIW の要素プロセッサを複数搭載し、要素プロセッサ間でメモリを共有して動作する。これにより、マルチスレッドで書かれたプログラムを効率的に処理する。しかし、並列プログラムによる並列性の取り出しは中粒度～疎粒度の並列処理技術であり、要素プロセッサ数が増えた場合に並列性の取り出しが難しくなる問題を抱えている。また、CMP は要素プロセッサ自体を改良する技術ではないため、命令レベル並列性抽出の性能は改善されない。SMT (Simultaneous Multi-Threading) アーキテクチャもまたスレッドレベルの並列性により性能向上を目指している。SMT はスーパースカラをベースにしているが、解析範囲を広げて並列性

を取り出すのではなく、複数のスレッドから並列に実行できる命令を取り出す。SMTではスレッド毎に独立に並列性抽出を行えるため、機構の複雑さはスレッドの数に対して爆発しない。SMTはCMPに比べてハードウェア規模が小さいというメリットがあるが、やはり細粒度の並列性利用は向上しない。このように、VLIWの改良による命令レベル並列性の抽出ではスーパースカラ同様に処理の複雑さが問題になり、複数の処理機構によるスレッドレベル並列性の抽出では命令レベルの並列性利用が向上しないことが問題である。

本論文では、大規模な命令レベル並列性抽出処理における複雑性を解消するため、分散化した実行機構で単一スレッドを処理するアーキテクチャ技術を検討した。

まず、大規模投機的実行の実現を目的として、多ポート、高速アクセス可能な多量の物理レジスタを構成する分散レジスタ技術を提案した。また、実行ユニット間のデータ転送を削減するため、連続で同じ実行ユニットに命令を割り当て、分岐命令で割り当て実行ユニットを切り替える方式を提案した。さらにシミュレータによって、投機実行の規模と命令連続割り当て長の関係を評価した。これらより、投機実行の規模が大きくなるに従い、命令の実行ユニットへの連続割り当て長も伸ばす必要があることが分かる。最大8分岐命令投機の規模では2分岐命令、最大16、32分岐命令投機の規模では4分岐命令、最大48、64分岐命令投機の規模では8分岐命令連続で一つの実行ユニットに命令を割り当てるのが望ましいことが分かった。

次に、命令ブロック技術を提案した。命令ブロック技術では多数の命令に対して一括して物理レジスタを割り当て、レジスタ・リネームを行い、レジスタマップを更新する。1命令ブロックにブレークポイントが1つでは命令ブロック幅が分岐命令の頻度により制限されるため、命令ブロックが複数ブレークポイントを持てるよう命令ブロック構成とリネーム方式に拡張を行った。また、任意のRMSに切り替えながら処理を続けられるよう、リネーム機構に拡張を行った。さらにRMS選択に2サイクルかかる仮定でソフトウェア・シミュレーションを行い、高スループットの命令デコードが達成できていることを確認した。

最後に、分散レジスタ、命令ブロックを利用し、大規模投機的実行を実現するアーキテクチャとして大規模データパス (Very Large Data Path: VLDP) アーキテクチャを提案した。同時にVLDPプロセッサに必要なレジスタリクエスト生成機構や分散レジスタ管理機構を提案した。また、シミュレーションによりVLDPプロセッサの特性を調べ、高スループットの命令フェッチが実現できていることを確認した。命令解析範囲は広がったが高いIPCは得られておらず、原因はデータ転送の遅延であると考察した。最後にVLDPアーキテクチャの今後の課題について考察した。