

実用処理における汎用プロセッサの拡張命令の性能評価

Practical Performance Evaluation and Analysis on Conventional Processor

三瀬 邦義 乃一 泰久 小糸 啓介† 酒居 敬一†
 Kuniyoshi MISE Yasuhisa NOICHI Keisuke KOITO Keiichi SAKAI

1 はじめに

近年、コンピュータで取り扱う情報のマルチメディア化が進み、音声や動画像などを処理する機会が増えている。このようなマルチメディアデータは情報量が非常に多いため、これらに対するフィルタリングなどの処理には莫大な計算量を要し、処理性能の向上した現在のプロセッサにおいても処理には時間がかかる。マルチメディア処理は膨大な計算量を必要とするが、その処理のほとんどが近傍箇所との演算や繰り返し処理であるため、マルチメディア処理は演算時にキャッシュやデータ並列処理などの工夫によって処理時間の短縮を図ることができる。そのため、プロセッサのこのような機能を拡張命令というマルチメディア処理に特化した命令として実装することで、その性能の向上を図ることが一般的になっている。このような拡張命令はマルチメディア処理に対して確かに有効であるが、時に拡張命令を使用してもさほど処理速度が向上したように感じられない。したがって、本稿では実験は拡張命令の性能を客観的に評価し、また処理能力の低下を引き起こす要因としてレジスタ間のデータ移動に関する命令が挙げられることを検証する。

2 拡張命令の性能評価

マルチメディア処理には一般的でかつ実用的な処理であるMP3(MPEG 1 Audio Layer-3)のエンコードを採用する。また、評価する汎用プロセッサとしてはIntel®社とAMD™社のIA-32アーキテクチャを用いる。これらは一般的で最も広く普及しており、より多くのサンプルデータを収集することが可能であることから採用した。プロセッサの性能評価は実際にマルチメディア処理をさせ、その処理のプロセスが開始し終了するまでの速度を測定することにより行う。

2.1 評価手段

性能を評価するためのサンプルデータは「午後べんち」を用いて収集した。これはMP3のエンコードに

†高知工科大学 情報システム工学科

よりベンチマークテストを行い、その結果をサーバに転送および蓄積することができるベンチマークシステムである。このシステムにより集まったおよそ12000のサンプルデータをプロセッサの開発コードごとに分類し、それぞれの拡張命令を使用した場合と使用しない場合の性能を集計し統計的に拡張命令の性能を比較することで評価した。

2.2 結果

ベンチマーク試験によるプロセッサの能力の結果の代表例としてIntel®社のPentium® 4の結果を図1に示す。

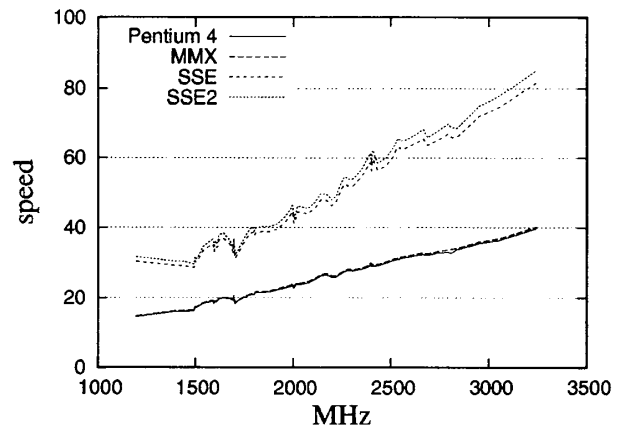


図1: Pentium® 4のベンチマーク試験結果

図1は横軸のプロセッサのクロック周期において縦軸は速度をPCMデータ再生時の何倍であるかを示している。今回実験を行った全プロセッサの結果は表1に示す。

この結果を見ると拡張命令を使わない場合に比べてSSEやSSE2を使用した場合を見ると処理能力が2倍に上昇しており、拡張命令は確かにマルチメディア処理に効果があることがわかる。しかし、SSEは128bitのXMMレジスタで単精度浮動小数点データ4つをSIMD演算することによる4並列処理が可能である[2]ことを考えると、4倍に近い処理能力が期待されるがそれほどの性能向上は得られておらず何らかの原因によって拡張命令の処理が阻害されていると考えられ

表 1: 拡張命令セットの処理速度向上倍率

プロセッサ	SSE	SSE2	3DNow!	E3DN
Katmai	1.67	-	-	-
Tualatin	2.17	-	-	-
Coppermine	1.92	-	-	-
Pentium 4	2.99	3.21	-	-
K6-2	-	-	1.74	-
K6-3	-	-	2.07	-
SlotA	-	-	1.53	1.55
ThunderBird	-	-	1.58	1.61

(値は拡張命令を使用しない速度を1とした場合の比率)

る。他のプロセッサに関してしてみると、そのほとんどが Pentium® 4 同様に拡張命令の処理は期待されるほど向上していない。その中でも 3DNow!™ に関しては 64bit レジスタで単精度浮動小数点データ 2 つを SIMD 演算する [3] ことから 2 倍の性能が期待できることを考えると、SSE に比べて性能向上の割合は高い。

3 拡張命令の性能低下要因の検証

拡張命令の効果が期待ほど得られないのは演算とは関係のないレジスタ間のデータ移動を目的とする MOV や SHIFT などの命令 (以下は単にデータ移動命令と呼ぶ) が多く含まれているためではないかと考えられる。そこで、プログラムをプロセッサの提供する命令レベルに分解して、プログラム中に含まれるデータ移動命令の割合調べることで、拡張命令の処理性能との相関関係があるという仮説を検証する。

3.1 検証手段

性能評価で利用した午後べんちの MP3 エンコーダのエンジン部分は拡張命令の性能を極限まで引き出すため、人の手によって高速処理のための可能な限りの最適化が図られている [1]。したがって、午後べんちは拡張命令の最大の処理性能に限りなく近い処理性能を引き出せるプログラムであると考えられる。そこで、これらのプログラムを拡張命令セットごとに分類し、それぞれの全アセンブリ命令とレジスタ間でのデータ移動に関する命令を数え上げ、全命令に対する割合と演算命令に対する割合を調べることで、データ移動命令が拡張命令の処理性の阻害要因となっていることを検証する。

3.2 結果

プログラムを解析した結果、データ移動に関する命令数と演算命令数およびその比率、総命令数は表 2 のようになった。FPU を使用しているプログラムではデータ転送命令は非常に少なく、3DNow!™ を使用し

ているプログラムでも比較的少ない。それに対して、SSE を使用しているプログラムではデータ転送命令を使用している割合が高くなっている。3DNow!™ が SSE に比べて処理性能の向上率が高いことを考えると、データ転送命令が拡張命令の処理を阻害している要因となっていることが考えられる。

表 2: 解析結果

拡張命令	ソース	N_m	N_p	N_t	N_m/N_p
3DNow!	gogo2.fht.nas	138	111	334	1.25
	mdct3dn.nas	43	69	142	0.62
	psy3dn.nas	387	733	1450	0.53
	sband3dn.nas	114	269	434	0.42
	quant3dn.nas	302	297	791	1.02
SSE	ftsse.nas	176	135	401	1.30
	gogo2.fht.nas	94	81	251	1.16
	mdctsse.nas	119	181	439	0.66
	psysse.nas	418	610	1353	0.69
	sbandsse.nas	174	272	538	0.64
	quantizea.nas	288	217	731	1.33
SSE2	quantizea.nas	47	57	130	0.82

N_m : データ転送命令数, N_p : 演算命令数, N_t : 総命令数

4 まとめ

今回の調査で拡張命令はマルチメディア処理に対して性能向上が見られるが期待されるほどの向上ではないこと、レジスタ間のデータ移動命令が拡張命令の処理性能の向上を妨げている傾向が見られることが分かった。演算命令とは関係のないレジスタ間のデータ移動命令が含まれてしまうのは、演算によって使用するレジスタが違うためその演算に合わせてデータを使用するレジスタに移動させなくてはならないためである。つまり、演算の組み合わせによってはレジスタに対するデータの再配置を行わなくてはならず、この目的とは直接関係の無い処理のため拡張命令が阻害されていると考えられる。したがって、さらにマルチメディア処理の高速化を考えるのであれば、処理すべきデータに合わせてレジスタ間で余計な移動をしなくてもよいように演算回路の構成を再配置することができるリコンフィギュラブルプロセッサや、マルチメディア処理に効果的なパイプライン処理と相性がよいデータ駆動型のプロセッサなど、プロセッサ自体のアーキテクチャも考え直す必要がある。

謝辞. ベンチマークデータを提供して下さった午後べんち開発者の方々には心よりお礼申し上げます。

参考文献

- [1] 酒居 敬一, 光成 滋生, 成田 剛, 石田 計, 藤井 寛, 庄司 信利: MP3 エンコーダの高速化実装, 情報処理学会論文誌 Vol.43 No.4 pp1028-1038 (2002).
- [2] Intel® "IA-32 Intel® Architecture Software Developer's Manual".
- [3] AMD™ "3DNow!™ Technology Manual".