Linux/Alpha 活用講座

清水尚彦　nshimizu@keyaki.cc.u-tokai.ac.jp

メモリシステムと数値計算

この原稿を書いている今は、とても師走とは思えない気候なのですが(当り前か)、早いもので、もう12月号ですね。気候も良く、このところバイクで山道を走ることが多くなりましたが、私が乗っている大昔のバイクは、登りはともかく下りのエンジンブレーキの効きが悪く、走りに気を使わなくてはいけないのです。ブレーキが優秀ならそれでもいいのですが、このバイクは前輪ですら貧弱なドラムブレーキがついているだけです。そこで、快適に山道を走れるようにキャブレターのセッティングを変更しました。そんなに大げさなものではなく、ただニードルを一段下げてエアーとアイドリングの調整をやり直しただけなのですけどね。簡単な調整でエンジンブレーキは十分効くようになって山道の下りは見違えるように走りやすくなりました(私はローリング族ではありません、念のため)。

こういったアナログ的な機械いじりは、なかなか面白いものです。4輪の車は、このごろコンピュータ制御で燃料噴射しますから、設定を変えるためにはROMの交換をすることになったりして、素人が適当に遊ぶことはできないので、ただの道具になってしまいますね。

その点、Linuxなどのソースコードがあるフリーソフトを使っていると、「ここをこうしたい」ということが原理的に可能になるから便利です……と言いたいところですが、本当にカーネルを調整しようと思うと、それなりのスキルが必要で、あまり簡単ではないのが残念ですね。普通の人は最低限の設定ファイルで定義されていること以上のチューニングは必要ないのでしょうけれど。車もLinuxも、必要なところに積木細工のようにパーツをつけたり調整するだけで、直観に一致した特性が得られるようになってくれると、人が使う道具として一人前になったと思えると思っているのは私だけでしょうか？

さて、このところAlpha関連ニュースについてのフォローをしていなかったので、今回はまず、ニュースからお伝えします。といっても、この号が発刊されるころには、「News」でも何でもなくなっているのが、印刷情報の悲しさですが。

Alpha／Linux関連ニュース

●2次キャッシュ統合版EV68

0.18μmのプロセスを使うEV68に、2次キャッシュを統合したバージョンがあるという話です。これは、10月のMicroprocessor Forumで、Samsungから発表されることなっていますから、この雑誌がみなさんのお手元に届くころには、詳細がはっきりしていることでしょう。「21264E」となっていますが、1.75Mbytesの7ウェイのキャッシュを統合しているそうです。キャッシュはフルスピードではないようですが、7ウェイもあれば、バンクがぶつからない限りフルスピードで転送できるように設計するのが普通でしょうから、性能的にも期待できます。これは、「EV7」と言われているコアに良く似た構成なのですが、バスは、EV6の高速版(533MHzと言われています)を利用して、クロックは1.2GHzから1.4GHzとなるらしいです。

●待望の1GHz版Alphaは……

Alphaの1GHz版は、IBMの銅配線プロセスのチップをCompaqで評価中ということです。製品化は2001年とされていますが、早いこと出てこないと、実性能はともかく、Compaqの商売としては辛いものがあるでしょうね。

●Compaqが30TFLOPSシステムを落札

Compaqが、米国エネルギー省のシステムを落札したとの報道がありました。30TFLOPSのシステムですが、1250MHzのAlphaを1万2000個以上使用する模様です。実稼働は2002年とのことです。これだとピーク性能ですから、プロセッサ数はこの何割か上のような気がしますね。それとも、入札はピーク性能をベースにしているのでしょうかね。

A ベンチマーク再考

多くのプロセッサアーキテクチャ設計者は、HennessyとPattersonの有名な著書「Computer Architecture: A Quantative Approach」(記事末RESOURCE[1]を参照)の影響からか、ベンチマークによる定量的な性能評価に偏ってシステムパラメータを設定しているような気がします。ベンチマークを指標として性能を評価することの危険性は、この本にも書かれているにも関わらず、恐らく商業的な理由と手軽さから、SPECやLINPACKが未だにアプリケーションの代表とでもいう顔をして、アーキテクチャの設計時に参照されているように思います。

SPECも、真面目にベンチマークのアップデートを続けているので、そんなに悪いものではないのですが、やはり、ここぞというときには物足りません。私は定量評価自体を批判するつもりは全くないのですが、ベンチマークが本来そのプロセッサが狙っている市場と合致しているかを、もっと考えたほうが良いと思っています。

Alphaを使う人たちの中には、私のように用もないのに物好きだけで使っている人は少なく、皆さん何かしら必要に迫られた事情があることでしょう。そうすると、Alphaのプロセッサ設計者は、それらの事情に合わせたベンチマークを基準にマイクロアーキテクチャの色々なトレードオフを決定するべきだと思うのです。それをSPECでやってしまうと、本来の市場と離れたところでの競争を強いられるために、本来の市場では妥当なトレードオフが、別の基準によって歪められることになるのではないかと恐れています。

例えば、SPECfp95というベンチマークは、数値計算ベンチマークとして良く利用されているもので、Alphaもこの性能が高いことを「ウリ」にしてきたわけですが、このベンチマークは、8Mbytes程度のキャッシュを積めば、ほとんどキャッシュヒットする程度のものでしかないのです。ベンチマーク制定時には8Mbytesキャッシュなんて現実的ではなかったのですが、今となっては、別に特別なことをしなくても、簡単にこの程度のキャッシュは持てます。そうなると、SPECfp95の表す数値計算性能はキャッシュヒット時の性能ということになってしまい、数値計算のために使いたい人たちの基準とは大きく異なることになるのではないでしょうか？

また、小容量のフルアソシアティブなTLBの採用も、多くはSPECやLINPACKといったベンチマークによって勝手に正当化されていますが、「本当にそれでいいの？」と思うわけです。

A 数値計算とメモリシステムの話

数値計算に強いと評判が先行したため、Alphaを使う多くの人は数値計算をターゲットにしているのではないでしょうか？

そこでは、性能としてはおそらくスーパーコンピュータとの比較が重要になると思います。筆者の独断で言ってしまうとスーパーコンピュータとマイクロプロセッサの大きな違いは、実は、メモリシステムにあります。もちろん、演算器の数とかベクトルレジスタなんていうものもあるのですが、これらは実は、数値計算の本質からみると些細な違いです。メモリスループットを大きくしてメモリレイテンシを隠蔽できるだけのローカルな記憶領域を用意することさえできれば、数値計算用の高性能計算機の出来上がりです。演算器なんてスループットに見合う数を用意すれば一般の用途には十分で、何処かのゲーム機のように、積和演算器を10個も搭載しても、メモリシステムが貧弱では、データ供給が追いつかない宝の持ち腐れになります。もちろん、同じデータに何度も演算を行うような画像処理専用の計算機ならば、データ供給はあまり問題になりませんから、ゲーム機はゲーム機なりのターゲットがあって当然で、その設計を云々するつもりはありません。

さて、実はマイクロプロセッサのメモリスループットは、ここ数年、劇的に上がってきつつあります。これはバス周波数の向上と立上り立ち下がり両エッジでデータを転送する方式の普及(これがプロセッサバスに使われる日も近いでしょう)、さらには、プロセッサの実装がフェースダウンのBGAタイプになってきたことによって、入出力ピンの数が多く取れるようになりつつある、などの理由によります。さらにこの実装形態でベアチップをセラミック基板上に直接搭載することで熱抵抗を劇的に下げることに成功したことが、最近のクロック周波数の向上を支えています。

少し前のSRAMを主記憶に使ったスーパーコンピュータでは、例えば、25nsのサイクルでアクセスできるバンクを256ウェイで構成していたりします。1バンクは8bytesですから、約80Gbytes/秒程度の理論スループットになります。これは今

でも十分大きな値ですが、全然届かないほどのものでもありません。ニュースの項に書いた高速バスを使えば、プロセッサに4Gbytes/秒以上のデータ供給ができるようになりますが、これを20ウェイ以上のマルチプロセッサで構成すれば、すぐにこの程度になります(まだ製品は出ていませんが、EV7ではRambusを複数チャネル使って10Gbytes/秒以上の転送性能を実現するとしています)。

バスのピークスループットが出るからといって、それを使い切れるかどうかは、転送方式にもよります。現行のEV6のバスでは、データが戻るまでに8個の転送を発行できるようになっています。1個の転送要求はキャッシュラインの64bytesとなるので、約500bytesのデータの転送を発行できることに相当します。メモリアクセスのトータルのレイテンシが100nsで、スループットが4Gbytes/秒だとすると、切れ目なくデータ転送を続けておくためには、400bytesのデータをバッファしなくてはいけないのですが、このバスではなんとかぎりぎり間に合いますね。そうは言っても、レイテンシが増えたらもうだめですね。

実は、たいていのメモリにとっては、100nsのレイテンシは、かなり厳しい条件になります。そこで、このレベルの転送をするためには、転送の発行数を増やす必要がありそうです。EV67までは、バスの速度も333MHzと低いので、こういった基本部分の変更はないのですが、クロックも1GHzを超え、バススピードも向上するといわれるEV68では、どのように変更されているか興味深いですね。

ちなみに心配症の人のために、現行の333MHzのEV6バスで、どの程度のレイテンシが許されるか概算してみましょう。

333MHzで8bytes幅のバスでは、プロセッサには2.6Gbytes/秒のスループットでデータが転送されます。そこで、このバスをフルスピードで動かすためには、メモリのレイテンシは197ns以内であればよいということになります。この程度なら、普通のDRAMを用いてもなんとか対応できる値です(といっても楽々実現できるものではなく、レイテンシに特に気をつけて設計したワークステーションがようやく達成できる程度のレベルだと思ってください。マルチプロセッサのサーバでプロセッサから見たメモリレイテンシが200nsを切ったマシンは、あまりないのではないかと思います)。

なんだ、それなら数値計算のプロセッサなんて簡単じゃないかと思った方は、もう少し考えてみてください。確かに、バスのスループットを使い切るためには、レイテンシが197nsでいいかもしれませんが、このレイテンシの間に命令はいくつ動作するのでしょう？

計算を簡単にするために、クロック500MHzのプロセッサを考えましょう。するとマシンサイクルは2nsとなります。1つのクロックで浮動小数点命令は2つ動作するので、197nsのレイテンシの間に、浮動小数点演算は197個実行可能となります。プロセッサのピーク性能を出そうと思うと、512bytesのデータに対して197個の演算を(しかも加減算と乗算を同数)行う計算をする必要があります。倍精度の数を前提とすると、1つの数値は8bytesで表現されるため、64個のデータに対して197個の演算という割合になります。つまり、1つのデータに3個の演算の割合が要求されて、これより演算が多いとバスが空いてしまいます。

これは演算器の不足を意味していますから、アーキテクチャの設計のバランスが悪いと言われかねません。反対に、演算が少ないとバスが飽和してしまいます。するとピークスループットより低いところでしか演算ができないわけですから、やっぱりバランスが悪いと言われてしまいます。どちらかと言えば、カタログからすぐにばれる演算ピーク性能が出ないよりは、バランスが悪くてもバスが空いて演算器はフルに動いているほうが印象は良いのは確かでしょう。

ところが、アプリケーションを見ると、なかなかそう、うまくはいかないことが分かります。大規模な数値計算では、積和演算の形をとることが非常に多いのですが、積和演算では、乗算と加算の割合はいいのですが、1つのデータに対して2つの演算しか必要とされないのです。ですから、3個の演算を要求されても、それだけの演算を用意できず、結果としてピーク性能より低いところで動作することになります。しかも演算した結果を格納するためにストア命令が発行されて、さらに余分なデータ転送が必要となります。

これを改善するには、データを使い回すことで転送回数を減らすしかありません。そのための代表的な手法がLINPACKのTPPなどで有名な「BLASレベル3演算におけるブロック化」があります。ここでは詳細は示しませんが、数値計算の教科書などを見ると出ていますので、興味のある方は参照してください。

さて、ここまで解析してみると、バスのピークスループットを実現するのは難しくても、同じ程度まで転送だけで近付くことはできそうな気がしてくると思います。

ところが、STREAMベンチマークの結果([3])を見ると、思ったより転送性能が出ていないと思われる方が多いのではないでしょうか？　EV6系列のマシンでは、大体1Gbytes/秒程度の転送性能になっています。これはピークスループットの2.6Gbytes/秒からみると、ずいぶん低い値ですね。さらに多くのマシンでは、COPYよりもADDのほうが良い値となっています。これは、実は、ADDはデータの読み出しが2回に書き込み

が1回であるのに対して、COPYはどちらも1回ずつとなっているからなのです。といってもピンと来ない方もいると思いますが、データの書き込みは、今の多くのマイクロプロセッサにとっては負荷が重い処理なのです。

というのは、ライトバックキャッシュを持つシステムでは、書き込みは、一度キャッシュにデータを取って来て、そのデータと書き込みデータをマージしてから書き戻すという手間がかかるからです。それでも1.5倍程度のオーバヘッドになるだけですが、データ線のピークスループットと転送プロトコルを含めた実スループットは必ずしも一致しないので、こういった結果になっているのでしょうね。実は、実転送スループットを前提にすれば、先ほど計算した演算数はもっと条件が悪くなって、本当に数値計算性能を出すのは大変だということになります。ちなみに、EV6で採用された書き込み専用のヒント情報をサポートしたコンパイラを使うと、キャッシュにデータを取ってくる必要がなくなるので、性能が向上するはずです(まだ確認していませんが、もしかすると、この命令を使った結果も公表されているかもしれません)。

ここでモデル化したような単純な話だけなら世の中簡単なのですが、アプリケーションにはさまざまなものがあって、特にビジネス上のうまみがあるアプリケーションの代表と言えば、データベースですね。

これはここ数年のOracleの株価を見れば分かるように、市場が急成長しています。で、データベースアプリケーションに対してOracleとCompaqが何をしてきたかというと、「Very Large Memory(VLM)」と呼ばれるような、メモリを主体とした性能向上への取り組みが有名です。

OpenVMSでのサポートになりますが、ページ粒度情報を使ってメモリ性能を向上させるようなこともやっています。といっても、私のやり方とは全く異なり、システム構成時に固定的に割り当てるメモリ領域にしか粒度情報は使えないので、柔軟性は全くありません。それでも、何でこんなことまでしなくてはいけないのかというと、連続メモリアクセスが多い数値計算よりも、データベースのようなデータ依存のメモリアクセスのほうがはるかにシステムに対する要求としては大きなものがあるからです。しかもデータ依存であるから予測が難しい。プリフェッチや分岐予測も外れまくるというたちの悪いアプリケーションです。こういったもので高い性能を得られるシステムこそ、良いシステムだろうと思うのですが、もちろん、データベースに興味のないユーザーにとっては関係のないお話しですね。

先月号まで3回にわたってAlphaのページ粒度ヒント情報をLinuxで利用する方法とカーネルの修正について書いてきました。

大規模なアプリケーションでは、メモリ性能が大きなネックになることが多いのですが、ベンチマーク結果などからも、この方式はある種のアプリケーションに対しては大変効果があることがお分かりいただけたことと思います。こういったカーネルへの修正をするときに、カーネルの動作すべてを理解しないとできないなんて大げさに考えずに、プロセスがどう動作するかを考えて、必要最小限の修正を加えて目的の機能を組み込むようにすれば、それほど難しくはありません。前述したように、「積木感覚」でプロセスの動きを追いかけて、必要なところに手を入れるというやり方が比較的容易だと思います。そうは言っても、今回のパッチにしても、いくらかの「考え落ち」は見つかっていて、修正版を出しているので、パッチを使おうとしている方は、筆者のホームページにある最新版をチェックしておいてください([4])。

おわりに

11月2日の「コンパックフォーラム2000」に、私もお呼びがかかって、のこのこ出かけていくことになりました。生意気な清水の顔を見てやりたいと思う方は、ぜひ会場でお会いしましょう。

RESOURCE

[1] 「Computer Architecture : A Quantitative Approach(2nd Edition)」(原書)
John Hennessy、John L. Hennessy、David Goldberg、David A. Patterson共著／ハードカバー760ページ／1996年刊行／Morgan Kaufmann Publishers／ISBN 1-558603298／価格：83.95ドル(2000年9月末におけるamazon.comでの価格)

[2] コンピュータ・アーキテクチャ――設計・実現・評価の定量的アプローチ」(第1版の邦訳)
富田眞治、村上和彰 訳／800ページ／1992年発行／日経BP社／ISBN 4-8222-7152-8／本体価格：1万1650円

[3] STREAMベンチマークの結果
http://www.cs.virginia.edu/stream

[4] 筆者のホームページ
http://shimizu-lab.et.u-tokai.ac.jp/~nshimizu/