解　説　高速プロセッシングデータバス技術

6. テクニカルサーバExemplarにおける高速データプロセッシング技術

A High Speed Data Processing Technology on Technical Server Exemplar by Ichizo KITAGAWA (Advanced Technology Center, Product Marketing, Technical Computing Marketing Center).

北　川　一　三[1]

1 日本ヒューレット・パッカード(株)

1. はじめに

はじめに今回題材にしているExemplar S/Xの簡単な紹介をします.

Exemplar S/Xクラスは，旧CONVEX社のSPPシリーズの第二世代としてHewlett-PackardとCONVEXの合併後に初めてリリースされた計算機です．このS/Xクラスは，ピーク性能720Mflopsという性能を引き出す64bitプロセッサPA8000(180MHz版)，高速15.3GBの8×8ノン・ブロッキング・クロスバー・スイッチなどの高速演算を能力を支える技術を採用したSクラス，さらにSクラスのマルチノード仕様としてXクラスに分けられています．それぞれの最大構成として，Sクラスでは PA8000を16プロセッサ，メモリを16GBまでの構成ができます．Xクラスでは64プロセッサ，64GBとなります．この余裕の計算リソースからS/Xクラスは Exemplarファミリーにおいて，高速演算性能，スループット性能や大規模な計算リソースを必要とするユーザ向けのハイエンド・マシンとして位置づけられています.

このS/Xクラスの高速演算性能を支える技術としてS/Xクラス共通のPA8000からメモリまでのアーキテクチャ，およびXクラスのマルチ・ノード間におけるデータバス技術の概要を述べます.

2. PA8000のプロセッサ・アーキテクチャ

PA8000プロセッサは，すべてのレジスタを64ビット幅とした完全64ビット・アドレッシングのRISCプロセッサです．プロセッサ内部の並列化を大幅に進め，10個の演算ユニット中，クロックごとに最大4命令が並列して稼動する4-Wayスーパースカラ技術を採用し，きわめて高い性能(720 Mflops)を発揮します.

PA8000の特長としては,

- 1MB命令用キャッシュと1MBデータ用キャッシュをサポート．(アーキテクチャ的には，さらに大容量のキャッシュにも対応できるようデザインされています.)
- 浮動小数点の積算／加算，除算／平方根計算，整数演算，シフト／マージ演算用に，それぞれ2つの演算器が用意されています.
- 2つのロード／ストア・ユニット(PA8000独自)
- 56個のエントリをもつ命令リオーダ・バッファ
- 12個のメモリ・リクエストの同時処理機能
- スペキュレーティブ実行
- プリフェッチ機能
- スタティック，およびダイナミックな分岐予測
- キャッシュ・アクセスのレイテンシは最大2クロック(連続したアドレスへのアクセスは1クロック)
- デュアル・ポート・キャッシュ
- 64bitアドレッシング

などがあげられます.

PA8000は，PA-RISC 2.0命令セットアーキテクチャを実装した最初のチップです．従来のPA-RISCとも完全バイナリ互換を保っています．2.0アーキテクチャの目的は，64bitの整数と64bitのアドレッシングをサポートすることです．性能の向上とともに，新しい命令セット・アーキテクチャに重要で不可欠な新しい機能も追加されました.

新しい特徴として，可変サイズのページ，新しい浮動小数点オペレーション，置換と予測の両方

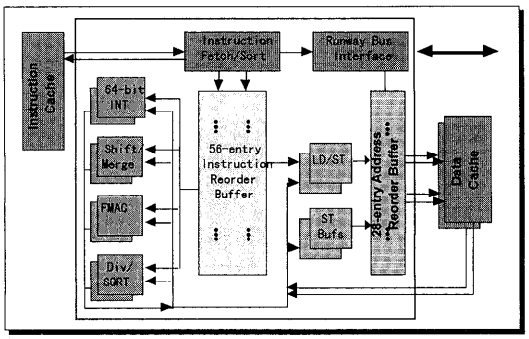


図-1 PA8000の概観

の手法を使った効果ある分岐処理などがあげられます．PA8000は，64bit整数がそのまま使える完全な64bitチップで，また，フラットな64bit仮想アドレス空間をサポートしていますが，チップ自体が出力するのは40bitの物理アドレスです．これは，直接1TBのメモリにアクセスすることが可能であることを意味します．PA-RISCの製品ラインの既存チップとの互換性を保つために，PA8000は，narrowとwideという2つのモードから1つを選択するコントロールbitをサポートしています．narrowアドレス・モードでは，32bitの物理アドレスだけがサポートされます．

PA8000は，分離アーキテクチャを採用しています．つまり，命令デコード・ロジックは，演算・ユニットのパイプライン・ロジックとは異なります．これにより，プロセッサは，演算・ユニットで命令が実際に実行されるよりも早く，命令をデコードすることができます．デコードされた命令は，チップ内のキューに入れられます．

PA8000は，実行中に最大56命令をもつことが可能で，クロック・サイクルごとに4命令を発行することができます．スーパースカラ性能をできるだけ高いレベルで保つために，PA8000は，独立した浮動小数点ファンクショナル・ユニット，独立した除算／平方根演算・ユニット，独立した64bit整数ALU，シフト／マージ・ユニット，そして独立したロード／ストア・ユニットをそれぞれ2個ずつ，計10個の演算ユニットをもっています．なお，クロックごとに発行できる4命令のうち，浮動小数点ファンクショナル・ユニット，除算／平方根ファンクショナル・ユニット，64bit整数ALU，シフト／マージ・ユニット用の命令は2個です．ここで重要なポイントは，2個の浮動小数点演算・ユニット用に，2個のFloatingPointMultiplyAndAccumulate(FMAC演算)命令をクロックごとに発行できるというです．つまり，ピークの浮動小数点演算性能は，4×動作周波数180MHzで計算できます．FMAC命令は，複合命令です．1つの命令で2つの浮動小数点演算が実行されます．また，FMAC命令のレイテンシは3サイクルですが，浮動小数点演算・ユニットは，クロックごとに命令を受け付けることができますので，演算結果もクロックごとに出力されます．

PA8000のキャッシュ・アーキテクチャは，PA-RISC製品ラインの既存製品と同じです．命令データとも1次キャッシュの1階層です．容量としては1MBから4MBのキャッシュ・サイズが構成できます．Sクラス，Xクラスでは，命令，データ各1MBのサイズで，ライト・バック処理のダイレクト・マップ方式を採用しています．キャッシュは，オフチップ上にSSRAM(Synchronous SRAM)で構成します．データ・キャッシュは，2クロックでデータを転送しますが，パイプライン処理およびout-of-order実行により，通常，レイテンシがないのと同じになります．命令キャッシュは，プロセッサにクロックごとに4命令を供給します．

豊富な演算・ユニットを十分に使うために，プロセッサは，56エントリのIRB(Instruction Reorder Buffer)，デュアル・ポート・データ・キャッシュ，クロックごとに4命令が命令キャッシュからフェッチできる能力を備えています．プロセッサは，IRBに最大56個の命令を保持することができ，必要なデータとファンクショナル・ユニットの準備ができた時に命令を発行します．IRBの中で，命令のデータ依存性はチェックされ，データと必要なファンクショナル・ユニットが利用可能になった時，その命令が演算・ユニットに送られます．分離アーキテクチャを採用した結果，PA8000は，out-of-order実行が可能になりました．IRBの中に命令があれば，レジスタ，データ，演算・ユニットが利用可能になった時に発行されます．プログラムに書かれた順序どおりに命令を実行する必要は必ずしもありません．しかし，命令は，必ずプログラムに書かれた順序で完了しなければなりません．このように，次々と

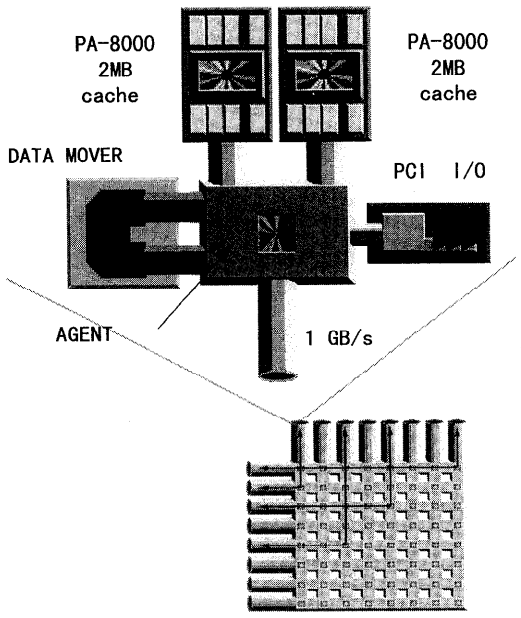


図-2 プロセッシング・サブシステムとクロスバー

ロード・リクエストを同時処理できるという機能に加えて，プロセッサは命令レベルでのスケジューリングを行うことができるということは非常に重要なポイントです．これは，X クラス・システムのような cc-NUMA(cache coherent Non-Uniform Memory Access)アーキテクチャを補完する機能です．つまり，PA8000 は，ローカル・キャッシュやメモリ内に配置されたデータへのリクエストと，プロセッサ・インタコネクトにわたったリクエストを同時にサポートします．その結果，非常に多くのレイテンシを隠してしまう，つまり，実際には，レイテンシは存在しますが，プロセッサはその影響を受けずに，実行を続けることができます．PA8000 のもう 1 つの優れた機能は，スペキュレーティブ(speculative)実行を行うことです．スペキュレーティブ実行とは，実行のパスをプロセッサに予測させ，予測したパス上の命令を実行させることです．予測が正しくなければ，その予測で実行された命令は排除されます．スペキュレーティブ実行が行われるケースとして，まず分岐のケースがあげられます．これは最も重要なケースです．PA8000 は，分岐によってどちらのパスがとられるかということを予測するための優れたメカニズムをもっています．分岐予測では実行されるべき命令の流れ，すなわち，分岐で選択されると予測されるパスを示します．そして，このパス上にある命令がスペキュレーティブ実行されます．もし分岐が誤って予測された場合実行された命令は単に排除されます．

3. 高速メモリ・データバス技術

Exemplar のアーキテクチャは大きくわけてクロスバー・スイッチ，プロセッシング・サブシステム，メモリ・サブシステム，I/O サブシステムから成り立っています．S クラス 1 台には，最大で 16 個までのプロセッサ，8 枚のメモリ・ボード，8 本の I/O チャネル(この時，24 個の PCI I/O コントローラが接続可能)を搭載することができます．(この S クラス 1 台分に相当するノードを Exemplar ではハイパノードと呼んでいます．)

3.1 クロスバー・スイッチ

S/X クラスのメモリ・サブシステムは，クロスバー・テクノロジをベースとして，15.3GB／秒という高いデータ転送性能を実現しています．

クロスバーのポート数は，8 ポート×8 ポートで，プロセッサ側の各ポートはエージェントに接続され，各エージェントには，2 個の PA8000 プロセッサ，240MB/s の I/O チャネル，および，データムーバが接続されています．一方メモリ側の各ポートには，4-way インタリーブのメモリ・ボードが接続されています．クロスバーは，120MHz(8.33ns)で動作し，エージェントとメモリ・コントローラ・チップ間のデータ幅は 64bit です．クロスバーのバンド幅は，ポートあたり単方向で 960MB／秒ですが，入力と出力の 2 つのクロスバーが組み込まれ，8 個のエージェントと 8 個のメモリ・コントローラは，入力用と出力用クロスバーの 2 つの接続路をもっています．したがってクロスバーの総バンド幅は，15.36GB/s になります．さらに，クロスバーはノンブロッキング・アクセスが可能なため，すべてのポートが最大限のバンド幅でデータを転送することができます．したがってプロセッサや I/O チャネルからメモリ・サブシステムへのアクセスを，クロスバーがブロックすることがないため，データ転送にバスを使用する場合に避けられない性能の低下を防ぐことができます．

3.2 メモリ・サブシステム

S クラスは，256MB から 16GB の SDRAM を

サポートしています．実際には，メモリは，2枚から8枚のメモリ・ボードに分散して配置されますが，SDRAMを使用することにより，メモリ・サブシステムを普通のDRAMより高いクロック周波数で動作させ，より高いバンド幅を得ることができます．メモリは，DIMM(Double In-line Memory Module)の形で，システムに実装されます．DIMMを使うことにより，メモリを偶数枚ずつシステムに増やすことによって，メモリ・インタ・リーブ性能の向上が可能になります．実際には，DIMMの数ではなくて，メモリ・コントローラの数によってメモリ・インタ・リーブは決まります．それぞれのメモリ・コントローラは，4-Wayのインタ・リーブをサポートしていますので，8個のメモリ・コントローラをすべて実装した場合は，搭載されているメモリの量に関係なく，32-Wayのインタリーブがサポートされていることになります．

3.3 プロセッシング・サブシステム

2個のPA-8000プロセッサ，I/Oサブシステム，DATA MOVERから構成されています．

• DataMover

I/OとDMAの性能をさらに高めるために，Sクラス，Xクラスは，ハイパノード内，およびハイパノード間の高速なデータ転送を行うことを目的としたハードウェアが実装されています．

このハードウェアは，データムーバと呼ばれ，プロセッシング・エージェントの一部分として，それぞれのプロセッシング・サブシステム内に含まれます．データムーバは，シングル・スレッドによるデータ転送を，450MB／秒の実効転送速度で行います．データムーバは，メッセージ・パッシング・ライブラリを用いるプログラミング環境での性能を向上するという重要な役割も担っています．

• I/Oサブシステム

Exemplar Sクラス，Xクラスは，ハイパノードあたり24PCIコントローラを接続可能な，非常にI/Oの接続性が高いシステムです．さらに，XクラスのI/Oサブシステムは，高いスケーラビリティを発揮し，システム内のハイパノードに分散して配置することができます．ハイパノード上の1組のプロセッサごとに，32bit PCIサブシステムへのI/Oチャネル(バンド幅：240MB／秒)があります．実装には32bit PCI規格を使用し，120MB／秒のバンド幅を実現しています．しかし実際には，Sクラス，XクラスのI/Oチャネルは，将来の64bit PCI規格も取り扱える設計になっているため，そのI/Oスループットは，将来的に現在の倍の240Mbyetes／秒になります．

PCIコントローラはI/Oチャンネルごとに3枚あり，ハイパノードあたり最大24PCIコントローラが接続可能です．インテリジェント化されたI/Oポートは，システム内のハイパノードに分散して配置された各メモリと直接DMA(Direct Memory Access)転送を行うことが可能です．これにより，CPUはデータ転送にほとんど介在することなく，本来の演算処理などユーザのための作業を行うことができます．また，大きなブロック・サイズを使うディスクのアクセスや高速なネットワークでの効率的なデータ転送を可能にします．これらのI/Oオペレーションは，データムーバにより大幅に性能が向上しています．Xクラスの I/Oサブシステムは，物理的にはシステム内にあるハイパノードのすべてに，あるいはいずれかに分散して配置することができます．しかも，システム内のすべての周辺装置，ネットワーク・インタフェースは，ノード内の接続を司るクロスバー・スイッチ，およびノード間を接続するインタコネクトを介して，システム内のどのメモリにもアクセスすることができます．同様に，すべてのプロセッサは，システム内のハイパノードに接続されているいずれのデバイスやディスクにもアクセスすることができます．たとえ，プロセッサとディスクが異なるハイパノードに接続されていても問題は起きません．

4. ノード間データバス技術

ここではXクラスのもつのマルチ・ノード・システムを実現する技術を紹介します．

Xクラスのノード自体は，Sクラスのハイパノードと同じですが，複数のハイパノードに分散配置されたメモリをどのプロセッサからも共有できようにし，さらにシステム全体にわたってキャッシュ・コヒーレンシをとるためのインターコネクト・ハードウェアが，ハイパノードを接続しています．特徴として(1)階層化されたccNUMAアーキテクチャ，(2)ノード間を結ぶCTI，(3)ハ

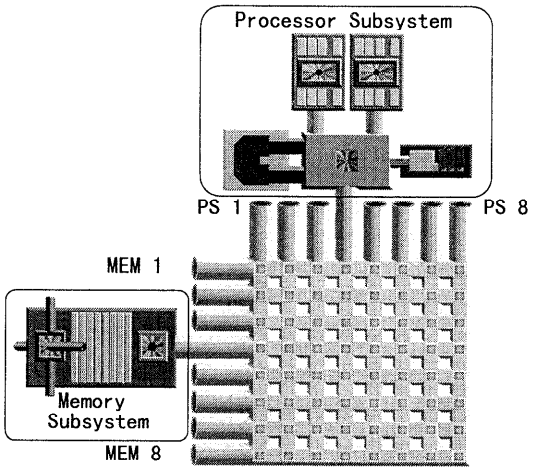


図-3

ードウェアでサポートさらたGSM(Global Shared Memory)を紹介します．(※ Exemplarではハイパノードと呼びますが，以下ノードと記述します．)

• 階層化メモリ・アーキテクチャ

ccNUMA アーキテクチャ採用により，メモリ・サブシステムのどの階層も，データ共有が最適に行えるようにデザインされています．この共有メモリ構造はは2つの階層にわけられています．

第1階層は基本的にSMPのメモリ・プロセッサ構造と同じです．この階層では16個までのプロセッサ，およびI/Oサブシステムと，物理メモリが，クロスバーによって接続され，複数のプロセッサ，I/Oサブシステムは，メモリへ同時にアクセスできます．

第2階層は複数のノードをCTI(Coherent Toroidal Interconnect)と呼ばれれるインタコネクトによって，それぞれの第1階層のメモリを結合することによって構築されています．CTIは2方向(X方向，Y方向)にそれぞれ複数のリングで構成されバンド幅を高めています．CTIによってノードが接続されることによってSMP型の複数ノードが，1つのシステムとして動作します．その結果，システム全体のメモリが1つの共有メモリとして使えるマルチ・プロセッサ・システムのように機能します．

• インタコネクト(CTI)

複数のノードを接続しているのは，IEEE1596-1992 SCI(Scalable Coherent Interface)に準拠した，低レイテンシのインタコネクトリングです．このインタコネクトは，共有メモリへコヒーレントなアクセスができるように，高バンド幅と低レイテンシという性能を提供します．CTIは，Point-To-Point接続の単方向リンクが集まったもので，X方向の1つのインタコネクションはリングとしてみなすことができます．このリングは，ノードに対し8個のインタフェース・コントローラをもち，インタフェース・コントローラからそれぞれ独立に出る8本のデータ・パスから構成されます．つまり，2つのノード間のリンクは，8本の完全に独立したパスであるインタコネクトをもっていることになります．8個のインタコネクトのデータは，リング上を単一方向かつすべて同じ方向に動くため，あるノードから，データの流れとは逆隣のノードにデータを送る場合，そのデータはリングを1周することになります．

CTIコントローラは，1つのリクエスト処理が完了する前に，次のリクエストの受付処理を開始することができます．多くのデータを高速に転送する上でこの機能は重要です．2つのノードが同じリング上にあるものと仮定した場合，ノード間のデータ転送は次のように行われます．(1)データを要求する側のノードが1キャッシュライン分のデータを応答側のノードに要求した場合，最初は，インタコネクトを通ってリクエスト・パケットが送られます．その時，リクエスト・パケットはそのパス上にあるターゲットでないノード上のCTIコントローラを通過します．(2)ターゲットとなる応答側のノードはそのリクエスト・パケットをみつけ，インタコネクトからそれを取り除きます．(3)そして，応答側のノード上のメモリ・サブシステムは，ローカルメモリから必要なキャッシュラインを引き出すための処理を開始します．(4)この時，応答側ノードは，ローカルのメモリ・サブシステムにアクセスすると同時に，要求側ノードに，ACK(acknowledge)パケットを送ります．(5)応答側ノードは，キャッシュラインを引き出すと，これを含めたレスポンス・パケットを作ります．(6)このレスポンス・パケットが，インタコネクトに置かれ，要求側ノードに送られます．(7)要求側ノードから応答側ノードにACKのパケットが送られデータ転送処理が完了します．

CTIはSCIと同じように分割処理プロトコルを

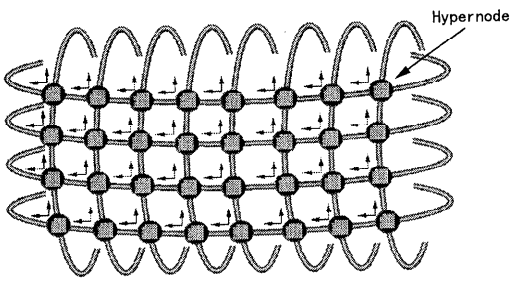


図-4 CTIリング

使っています．CTIの分割(split)とは，リクエスト・パケットとレスポンス・パケットとがまったく別のものであることを意味します．この分割処理プロトコルとACKパケットの同時処理という観点から，「一番近いノード」とか「ホップリクエスト」という考え方は存在しません．もし，要求側ノードと応答側ノードがリング上のデータの流れに沿った方向で物理的に近く配置されているならば，リクエスト・パケットは短いパス，そして，レスポンス・パケットは，長いパスを通ります．逆に2つのノードが離れている場合は，リクエスト・パケットは長いパス，そして，レスポンス・パケットは，短いパスを通ります．結局，リクエスト・パケットとレスポンス・パケットが通るパスの長さの合計は，リング1周分になります．

• CTIキャッシュ

CTIのレイテンシを最少限にするために，各ノードはCTIキャッシュと呼ばれる，ローカルなキャッシュを備えています．インタコネクトを介してほかのノードのメモリから参照されたデータが，一時的にここに格納されます．CTIを介してアクセスしたデータは，すべていったんCTIキャッシュに入れられ，そのデータがノード内のどこかのCPUキャッシュにあるかぎり，CTIキャッシュに存在します．CTIキャッシュのディレクトリ情報から，そのCTIキャッシュに入れられたグローバルデータが，現在どのCPUキャッシュに存在するかを知ることができます．CTIキャッシュは，グローバル物理アドレスをタグとしてもち，ソフトウェアのサポートを必要とせず，ハードウェアだけによるアドレッシングが行われます．Sクラス，Xクラスは，複数のハイパノード間でのキャッシュのコヒーレンシを保証しています．つまり，同じグローバル・アドレスを参照している複数のハイパノードは，常に同じ値をみることができます．これは，キャッシュラインを共有しているノードや，単独でキャッシュラインをもっているノードの情報をもっているリンクリストを使うことによって実現されています．また，どのプロセッサがCTIキャッシュ内のどのラインをもっているかについても記録しているため，ほかのノードから送られてきたコヒーレンシの要求は，適切なプロセッサへ転送されます．

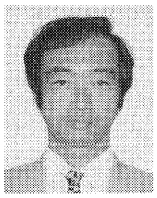
• ハードウェアGSM(Global Shared Memory)

Xクラス・システムは，ハードウェアでGSMをサポートしています．ノード間を跨いで共有メモリを実現する場合の重要な機能になっています．もしこの機能がなければソフトウェア・コントロールでノードからノードへページを動かして共有メモリをエミュレーションする必要があり，大きなオーバーヘッドが生じることになります．このハードウェア制御によりオーバーヘッドが削減され，効率よくパラレル・アプリケーションを実行することができ分散共有型計算機のメリットを受けることができます．

5. おわりに

本文は昨年HPで配布した冊子「Exemplar Technical Serverファミリ」の紹介記事および以下のサイト情報を参考にして本特集の趣旨に合うように加筆再構成をしたものです．この冊子にはソフトウェア関連も含めて総合的に紹介されていますのでExemplarにご興味をもたれましたらお問い合わせください．またExemplar S/Xクラスの最新情報はU.Sの公開サイトhttp://www.convex.comでも紹介されています．

(平成9年3月31日受付)



北川 一三

1962年生．1986年日本大学生産工学部建築工学科卒業．卒業後リアルタイム計測アプリケーションソフトウェアの開発に携わる．1991年コンカレント日本(株)に入社．引き続き建築，自動車関連を中心にリアルタイム実験計測アプリケーションを開発．1996年に日本コンベックスコンピュータ(株)に入社．同年合併により日本Hewlett Packaed(株)へ移籍し，現在TCBU Advanced Technology Center所属．