1　はじめに

JUMP-1[1]は、7大学共同で開発されたCC-NUMA型の超並列計算機であり、1000プロセッサを超える規模の超並列計算機における、効率的な分散共有メモリの実現に必要な各種の手法の開発を目的としている。

従来はHIPPI[2]に代表されるような高速I/Oを、特定のノード(クラスタ)に接続するのが一般的な方法であったが、これは入出力ノードの周辺にボトルネックを生じ、入出力がボトルネックとなって性能の低下を招くことがある。したがって、超並列計算機における入出力アーキテクチャは慎重に設計されなければならない。

JUMP-1の入出力機構の設計にあたっては、入出力機構のスケーラビリティ、柔軟性、アクセス容易性に配慮した。即ち、クラスタ数の増加にあわせて、計算能力と同じように入出力バンド幅も拡大されること、多様な入出力機器を接続できること、入出力インターフェイスへのアクセスがハードウェア的にもソフトウェア的にも簡単であること、の3点である。

JUMP-1のハードウェアは既に完成しており、システムソフトウェアの開発が進められている。今回、入出力関係のソフトウェア実装の第一段階を行い、そのソフトウェアを利用して、入出力性能の評価を行った。

2　JUMP-1の概略

JUMP-1は、1クラスタあたり4プロセッサ、最大256クラスタ1024プロセッサのシステムである(図1)。クラスタ間結合網にはRDT[3][4]を採用しており、効率の良い分散共有メモリの実現を目指している。

クラスタボード(図2)には要素プロセッサであるSuperSPARC+が、専用のL2キャッシュとともに4つずつ搭載されている。各クラスタはバス結合マルチプロセッサであり、クラスタバスにはSuperSPARC+のL2キャッシュと専用のメモリ管理プロセッサであるMBP-light[5]が接続されており、クラスタに搭載された16MBのSDRAMへのアクセスはMBP-lightを通じて行われる。

MBP-lightはSDRAM(クラスタメモリ)の管理の他に、RDTや本研究の対象である入出力機構STAFF-Linkの管理も行う。メモリ、I/OへのアクセスをMBP-lightが一括して行うことにより、分散共有メモリを実現したり、またメモリ空間上にI/Oをmapすることが可能となる。MBP-lightは、16bit RISCコアを持つとともに、分散共有メモリの管理を高速に処理するための専用ハードウェアを持っている。

3　JUMP-1の入出力サブシステム

3.1　設計方針

JUMP-1の入出力サブシステムの設計にあたっては、以下の点に特に注意が払われた。

拡張性(スケーラビリティ)　JUMP-1の分散共有メモリは、1000プロセッサを超える規模でのスケーラビリティを考慮して設計されている。したがって、計算機システム全体の処理能力の鍵のひとつである入出力性能も同じようにスケーラビリティを持つものでなければならない。このため、JUMP-1では、

- クラスタ本体と入出力サブシステムを分離
- 全てのクラスタに入出力のためのインターフェイスを装備
- 入出力サブシステムは複数の入出力ユニットで構成し、並列に動作させることによって物理的アクセスを分散
- 入出力ユニット間はEthernetやSTAFF-Link(後述)などで接続

という方針をとっている。

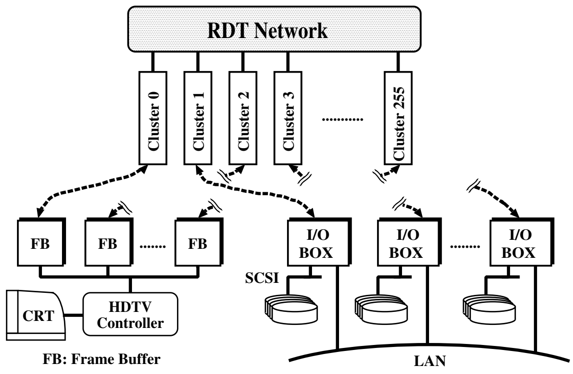

図 1: JUMP-1の構成

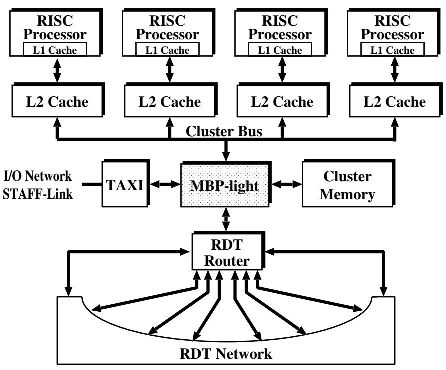

図 2: JUMP-1クラスタの構成

図 3: JUMP-1クラスタと入出力ユニット

柔軟性 計算機を単にベンチマークのために使うのでなく、様々なアプリケーションを実現するために用いるには、様々な入出力機器を接続できること、また計算機本体が大きくなるため、入出力機器の設置場所が自由に選べることが求められる。したがって、

- 入出力ユニットには汎用のワークステーション(SunのSPARCstation5 (SS5))を使用
- クラスタ本体と入出力ユニットの間は伝送距離を長く取ることのできる高速シリアルリンクで接続

アクセス容易性 入出力のための手続きが複雑でなく、簡単にソフトウェアやハードウェアの開発ができること。

3.2 STAFF-Link

JUMP-1向けに設計された入出力用のPoint-to-PointリンクがSTAFF-Link(Serial Transparent Asynchronous First-in First-out)[6]である。パラレルリンクでなくシリアルリンクを採用するのは、以下のような理由による。

- 多数のクラスタと接続するため、ケーブルやコネクタが細く、小さいことが望ましい。シリアルリンクは線も細く、コネクタのピン数も少なくて済む
- クラスタシステムが大きくなるため、入出力ユニットの設置場所が自由に選べる小と、即ちリンク長の制約が緩いことが望ましい。ノイズ対策等の面でパラレルリンクは不利である
- 高速なシリアル通信を行うための専用LSIの出現

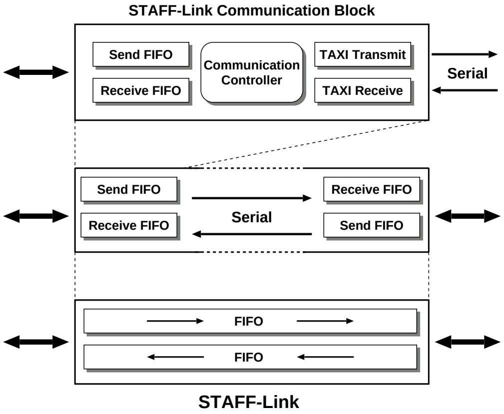

図 4: STAFF-Linkの構成

また、STAFF-Linkは、図4に示すように両端のインターフェイスにFIFOを持っており、FIFOにシリアル—パラレル変換を行うTAXIチップなどが接続されている。STAFF-LinkへのアクセスはFIFOへのアクセスとして行うことができる。これは簡単なハードウェアで、簡単な入出力ソフトウェアの実現を可能にしている。

STAFF-Linkは、以下のようなコンポーネントにより構成される。

- STAFF-Link ドータカード
  FIFO, TAXI, およびそれらのコントローラを搭載したカードで、ドータカード間をカテゴリ5のツイストペアケーブル(STP)で接続する
- JUMP-1 STAFF-Linkインターフェイス
  JUMP-1のメモリ管理プロセッサであるMBP-lightのローカルバス上に接続された、STAFF-Linkドータカードを制御するためのコントローラ。小規模なPLD 2チップで構成される単純な構成である
- STAFF-Linkマザーボード/SBusカード
  マザーボードはSTAFF-Linkドータカードを4枚搭載することができ、SBusカードとフラットケーブルで接続する。SBusカードを、入出力ユニットであるSS5のSBusスロットに挿すことにより、SS5からSTAFF-Linkへのアクセスが可能となる

3.3 入出力ネットワーク

入出力ネットワーク(図5)は、入出力ユニット間の負荷の分散やデータの共有のために必要である。当初STAFF-Linkでの構成を前提としていたが、現在の評価システムではEthernetで構成している。

Ethernetは広く使われているネットワークであり、入出力ユニットとして用いているSS5でもサポートされている。Ethernet (IP)では、OSの提供する分散ファイルシステムであるNFSが標準で使用できる。しかし、Ethernetはブロードキャスト型のネットワークであるため、帯域幅に限界があり、スイッチを用いればある程度解決できるものの、

多くはStored Forward方式であるため遅延が存在し、性能に悪影響を及ぼす。

STAFF-LinkはPoint-to-Pointのリンクであるため、Ethernetのようなバス型のトポロジをとることができない。したがって、複数の機器を接続するためにはスイッチを接続する必要がある。DSPを用いたSTAFF-Linkスイッチ機構が既に開発、評価されている。しかし、このスイッチを用いた場合には分散ファイルシステムを独自に実装する必要があるため、今回の実装と評価ではEthernetを用いた。

今回の評価で使用した、4クラスタ4入出力ユニットのシステムの写真を図3に示す。手前に見える大きな箱がクラスタであり、後ろに積み重ねてあるのが入出力ユニットの一部である。

4　I/Oソフトウェアの実装

JUMP-1プロジェクトは終盤に入っており、さまざまな評価ソフトウェアの実装が進められている。今回実装されたI/Oソフトウェアは、I/Oサブシステムの評価という目的もあるが、同時に他のプログラムの開発を支援するという意義も持っている。SS5には様々なI/Oデバイスを接続することができるが、評価をとることと、他のプログラムへの流用を考慮して、ファイルシステムの実装を行った。

4.1　I/O動作の流れ

JUMP-1の要素プロセッサであるSuperSPARC+が、SS5にアクセス要求を出すためにはいくつかの手順を経る必要がある。ここではそれについて、順を追って述べる。

1. コマンドおよび引数の設定(SuperSPARC+) :SuperSPARC+は、メモリ上に、入出力コマンドとそれに対する引数を、予め決められた順序にしたがって配置する。
2. MBP-lightの呼出し(SuperSPARC+) :SuperSPARC+は、L2キャッシュコントローラを通じてMBP-lightに要求を送信する。
3. メモリ読み出し(MBP-light) :MBP-lightは、SuperSPARC+がメモリに格納したコマンドと引数を取得必要がある。実際にはこれはL2キャッシュに置かれているため、そこから読み出すことになる。そのため、MBP-lightは、L2キャッシュコントローラに対して読み出し要求を送信する。
4. SS5へコマンド転送(MBP-light) :L2キャッシュコントローラが応答すると、MBP-lightに割り込みがかかる。MBP-lightは受信されたデータ(32バイト単位)をそのままSS5に転送する。転送後、結果が来るまで待つ。
5. コマンド実行(SS5) :SS5は、受信したコマンドと引数を解釈して実行し、結果を返送する。
6. 受信と終了通知(MBP-light) :MBP-lightは受け取った結果をクラスタメモリに書き込み、メモリ上のフラグを通じてSuperSPARC+に終了を通知する。

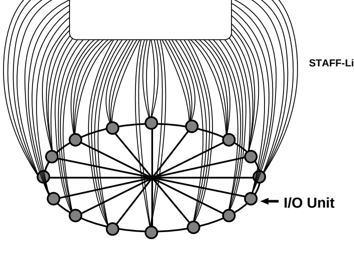

図5: JUMP-1本体と入出力サブシステム

4.2　実装

前節で述べたように、I/O処理はSuperSPARC+, MBP-light, SS5で順番に実行されるため、それぞれでのソフトウェアの実装が必要となる。ここで、それぞれの実装について述べる。

SuperSPARC+ MBP-lightに要求を送るための関数を実装し、それをベースに様々なコマンドを送り、結果を受け取る入出力関数を実装した。入出力関数はメモリ上にコマンドブロック(32バイト)とデータブロック(32バイトの整数倍)を展開し、MBP-lightにそのアドレスを渡すことによってSS5にデータを転送する。コマンドブロックには、コマンドの他に固定長の引数などを納めることもできる。これらの関数は、SuperSPARC+のセットアップに必要な初期化関数群とともに、JUMP-1用のCライブラリに納められている。

MBP-light MBP-lightは、SuperSPARC+用の実行バイナリをメモリに展開したり、またL2キャッシュコントローラから送られてくるメモリ読み出し要求などに応答する必要がある。現在の実装では、STAFF-Link及びMBIF(Maintenance Bus InterFace)からSuperSPARC+用の実行バイナリを読み込み、実行することができる。今回は、このプログラムを拡張して、SS5へのコマンド/データ転送を実現した。

SS5 SS5では、Solaris7が動作しており、この上で入出力管理プログラムを動作させる。このプログラムは常に

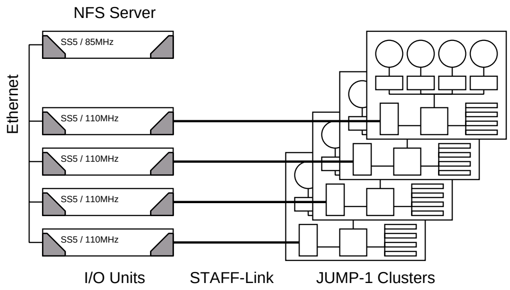

図 6: 評価システム

STAFF-Link の FIFO ステータスを確認しており、コマンドが到着すると必要な処理をする。

5 I/O 性能の評価

5.1 主眼

今回の評価における着目点はふたつある。

一点目は、複数の入出力ユニットを用いることの効果がどれだけあるのかという、スケーラビリティの観点である。これは、入出力ユニット間のネットワークの転送能力の制約を受ける。もう一点は、STAFF-Link の転送能力に対して様々な入出力処理のオーバーヘッドがどれだけあるのかという点である。これらは、今後のシステムの改良に大きく関わることになる。

5.2 システム

評価に用いた機材は以下に示す通りであり、図6のように接続した。

- JUMP-1 クラスタ
  4 クラスタを 16MHz で稼働させた。
- SS5
  5 台を用意した。内訳と仕様は以下の通り。
  - NFS サーバ
    MicroSPARC II 85MHz, 160MB RAM
  - 入出力ユニット
    MicroSPARC II 110MHz, 96MB RAM
- その他
  クラスタと入出力ユニットはカテゴリ 5 ツイストペアケーブル (STAFF-Link) で接続した。入出力ユニット、NFS サーバは 10Base-T Ethernet で相互接続した。

表 1: 実効転送速度の合計

ディスク	Cluster–SS5	`read`	`write`
Local	1 − 1	2.10	3.50
	2 − 1	1.64	0.64
NFS	1 − 1	2.10	3.51
	2 − 2	4.22	7.02
	3 − 3	6.41	10.93
	4 − 4	8.49	14.50

(Mbps)

表 2: STAFF-Link の転送速度

ディスク	Cluster–SS5	`read`	`write`
Local	1 − 1	3.86	12.29
	2 − 1	2.50	12.29
NFS	1 − 1	3.86	12.29
	2 − 2	3.86	12.29
	3 − 3	3.86	12.29
	4 − 4	3.86	12.29

(Mbps/cluster)

5.3 方法と結果

以下に示す条件で、ファイルに対するシーケンシャルリード / ライトを実行し、転送速度を測定した。Local disk を用いた場合は、各クラスタは同じ入出力ユニットのディスク上の別々のファイルに、NFS を用いた場合は、同じ NFS サーバ上の別々のファイルへのアクセスである。

- Local disk を使用
  - 1 クラスタ、1 入出力ユニット
  - 2 クラスタ、1 入出力ユニット
- NFS を使用
  - 1 クラスタ、1 入出力ユニット
  - 2 クラスタ、2 入出力ユニット
  - 3 クラスタ、3 入出力ユニット
  - 4 クラスタ、4 入出力ユニット

以上について、ファイル転送の実行転送速度 (表 1、図 7) のほかに、SS5 や JUMP-1 クラスタでの処理時間を含まない、純粋な転送速度 (表 2) を測定した。

5.4 結果に関する考察

5.4.1 スケーラビリティ

図7から、単一の入出力ユニットを用いるよりも、複数の入出力ユニットを並列動作させる場合のほうが明らかに性能面で有利であることがわかる。

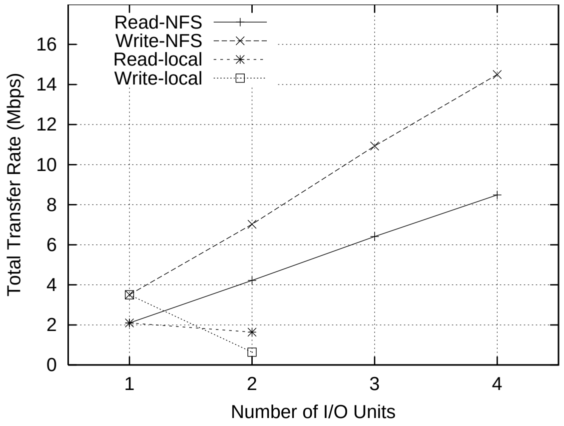

図 7: 実効転送速度の変化

また、入出力ユニットのOSのディスクキャッシュ機構が有効に動作しているため、今回は4クラスタ4入出力ユニットの14.50Mbpsまで直線的にバンド幅が拡大できたが、今後は同一のファイルに対する複数のクラスタからのアクセスや、大きな単位でのアクセスの集中などに対応するための並列分散ファイルシステムを検討していく必要があると考えられる。

5.4.2 オーバーヘッド

表2から、STAFF-Linkのリンクあたりの転送速度は、ひとつの入出力ユニットに複数のクラスタを接続した場合を除いて一定であることがわかる。この一定の値を、表1と比較すると、readで3割程度、writeで6割程度のオーバーヘッドの存在が明らかになる。

これは、JUMP-1のメモリアクセスの単位は32バイトであり、現在の実装では入出力処理全体が32バイトの転送をひと区切りとして行われている。これは余計なトランザクションを増やすことになり、性能低下につながる。SS5上でのファイルのread/writeも32バイト単位で行われており、このオーバーヘッドは少なくないものと考えられる。今後は、MBP-lightからSS5への転送ブロックサイズの拡大などを行う必要がある。

6 結論

JUMP-1上にファイル入出力のためのシステムソフトウェアを構築し、評価を行った。クラスタと同数の入出力ユニットを用いた場合、readでクラスタあたり2.1Mbps、writeでは4.5Mbpsの実効転送速度を得ることができ、4クラスタ4入出力ユニットまで直線的に性能が向上することを確認した。しかしその一方で、STAFF-Linkの転送能力はそれ以上であることが確認されており、より一層の性能向上が見込まれる。また、転送性能の向上に併せて、入出力ユニット側の実装の改善も必要であると考えられる。

参考文献

[1] Hidehiko Tanaka, et al., editors. *The Massively Parallel Processing System JUMP-1.* Ohmsha, 1996. ISBN4-274-90083-5.

[2] *ANSI document: High Performance Parallel Interface, document #X3T9.3*, 1987.

[3] Yule L. Yang, et al. Recursive Diagonal Torus: An interconnection network for massively parallel computers. In *Proceedings of the 5th IEEE Symposium on Parallel and Distributed P rocessing*, pp. 591–594, December 1993.

[4] Hiroaki Nishi, et al. The RDT Router Chip: A Versatile Router for Supporting a Distributed Shared Memory. *IEICE transaction on Information and Systems*, pp. 854–862, 197.

[5] Inoue Hiroaki, et al. MBP-light: A Processor for Management of Distributed Shared Memory. In *Proceedings of the 3rd International Conference on ASIC*, pp. 199–202, October 1998.

[6] Hironori Nakajo, et al. An I/O Network Architecture of the Distributed Shared-MemoryMassively Parallel Computer JUMP-1. In *Procedings of 11th International Conference On Supercomputing (ICS97)*, pp. 253–260, 1997.