計算機アーキテクチャ 119−43
（1996. 8. 29）

RWC-1の要素プロセッサ
- 細粒度並列処理機能の強化 -

松岡浩司† 岡本一晃† 廣野英雄†
横田隆史† 坂井修一††

並列実行モデルとプロセッサ間通信の観点から超並列計算機RWC-1の要素プロセッサの概要について述べる．RWC-1の要素プロセッサは通信と処理を融合し，かつ単純化したアーキテクチャであるRICA（Reduced Inter-processor Communication Architecture）を採用し，細粒度の並列処理を効率良く実行することができる．本稿では，スーパスカラプロセッサのデータパスを利用することによって現実的なコストでRICAを実装する方式を提案する．

RWC-1 Processor Elements
- an extension for fine grain parallel executions -

HIROSHI MATSUOKA†, KAZUAKI OKAMOTO†,
HIDEO HIRONO†, TAKASHI YOKOTA† and SHUICHI SAKAI††

From the view point of parallel execution models and inter-processor communication architecture, an overview of processor elements of massive parallel computer RWC-1 is discussed. RWC-1 processor elements adopt the RICA(Reduced Inter-processor Communication Architecture) in which communications and processing are fused and simplified to increase fine grain parallel execution efficiency. A cost effective implementation method for RICA which uses the data paths of conventional super-scalar processors are proposed.

1. はじめに

新情報処理開発機構（RWC）では，中長期的な展望に立った汎用超並列計算機RWC-1の研究開発を進めている．RWC-1のアーキテクチャについては既に述べたが，その中でも中心になるのは，通信と処理を融合し，かつ単純化したアーキテクチャRICA(Reduced Inter-processor Communication Architecture)[1]である．我々は，このRICAを採用したプロセッサの開発を進めてきたが[2)~4)]，新規にプロセッサを開発する期間を短縮するために，開発を検証と実証の2つのフェーズに分割して行なっている．検証フェーズでは，機能を限定した先行試作チップを作成し，プロセッサを稼働させるために必要とする支援システムなどの開発を先行して行なった．実証フェーズでは，検証フェーズにおける先行試作チップの開発で得られた知見を元に，量産版チップの開発を進めている．

† （技組）新情報処理開発機構 つくば研究センタ
Tsukuba Research Center, Real World Computing Partnership
†† 電子技術総合研究所
Electrotechnical Laboratory

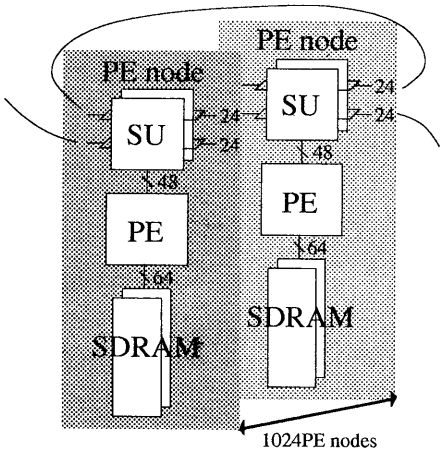
本稿では，RWC-1の要素プロセッサの概要を並列実行モデルとプロセッサ間通信の観点から述べるとともに，先行試作チップにおける性能的な課題と，その課題を解決するために開発した量産版チップの概要について述べる．

2. RWC-1の要素プロセッサ

2.1 RWC-1の概要

図1に示すように，RWC-1は1024のPE（Processing Element）ノードから構成される並列計算機システムである．各PEノードは命令を実行するプロセッサ（PE）と通信を管制する2つのスイッチングユニット（SU）[5)]から構成される．各PEノードは32MBのメモリ有し，このメモリには高いデータ供給能力を持つSynchronous DRAMを採用している．RWC-1では，このようなPEノードがMDCE(Multi-Dimensional Directed Cycles Ensemble)[6)]と呼ばれる直接網によって接続されている．

このネットワークを構成するSUは複数の入力を持つが，それぞれの入力に到達したデータの位相を内部のクロックに同期化する機構を有している．このような機構



performance@1024nodes:
100-132GIPS(peak)
50-66GFLOPS(peak)
memory 32-64GB(total)

processing element:
RICA: Reduced Inter-Processor Communication Architecture
internal clock:50-66MHz
super scalar execution:100-132MIPS(peak)
internal FPU:50-66Mflops(peak)

inter-connection network:
MDCE: Multi-dimensional Directed Cycles Ensemble
100-132MTrans/sec(300-396MB/sec)/port

図1 RWC-1の構成
Fig. 1 Configuration of RWC-1

を有するため，PEノード間の接続を任意の長さのケーブルによって行なうことができる．超並列計算機では，数多くのプロセッサを接続する必要があるので必然的に物理的な形状が大きくなってしまうが，RWC-1では，このようにPEノード間の接続を任意の長さのケーブルによって行なうことができるため，システムの設置に大きな自由度がある．もちろん，PEノード間の距離を大きくとると，信号の伝搬遅延が大きくなるので，アプリケーションによってはシステムの実効性能が低下する場合がある．

2.2 並列実行モデル

複数のプロセッサ間でデータを共有する問題の並列処理では，他のプロセッサが有するデータをリモートアクセスすると待ちが生じ，プログラムの実行に要する時間が長くなってまう．このような性能の低下を回避するためには，以下2点について考慮し，対策を講じなければならない．

- アクセス時間の短縮
  データをコピーすることによって，共有データへのアクセスに要する時間そのものを短縮する．平均的なアクセス時間を短縮できる反面，共有データへの書き込みが行なわれた場合，すべてのコピーを更新する必要がある．
- アクセス遅延の隠蔽
  待ちが生じた場合，他のスレッドの実行に移行する．プロセッサの稼働率を高く維持することができ，結果的にプログラムの実行に要する時間を短縮できる．反面，コンテクストの退避/復帰などのオーバヘッドが生じる．

さまざまな問題を扱う汎用の超並列計算機では，問題の性質に応じてこれらの方式を選択できなければならないが，RWC-1では，主として，後者のスレッドの隠蔽を支援するハードウェアの拡張を行なっている．

RWC-1では，リモートアクセスを発行し待ちが生じると，並列処理の基本単位である“スレッド”は明示的にその実行を終了する（つまり，待ちが生じるまで実行を継続する）．キュー上に実行可能のスレッドを指定するパケットが格納されていて，スレッドの実行が終了すると，このキューから最も実行優先順位の高いパケットを1つ取り出し，そのパケットによって指定されるスレッドを起動する．リモートアクセスを行なう場合には，実行を再開すべき命令のアドレスを**continuation**として，データのアドレスとともにデータを所有しているプロセッサに送る．リモートアクセスを受理したプロセッサは，データをアクセスし，読み出したデータをペイロードとして，**continuation**で指定される元のプロセッサ上のスレッドを起動する．リモートアクセスを行ない待ちが生じた時点で起動されたスレッドの実行が終了した，つまり，リモートアクセスの遅延を隠蔽するために実行していたスレッドの実行が終了した時点で，**continuation**で指定されるスレッドが起動され，リモートアクセスで中断された処理が再開される．

RWC-1では，メモリ保護を実現するために，リモートアクセスをスレッドとして処理している．つまり，リモートアクセスは，データを所有するプロセッサにおいて，（アクセス保護例外を発生する）通常のメモリアクセスとして実行される．なお，データが得られるのを待っているスレッドが存在するわけであるから，リモートアクセスを処理するスレッドの実行には通常のスレッドよりも高い優先順位を与えている．

2.3 細粒度並列処理

処理の粒度が細かい，つまり，スレッドとして実行される命令の数が少ない場合には，スレッドの実行時間に対するスレッドの切替え時間が相対的に大きくなり，並列処理の効率が著しく低下してしまう．このような状況下では，スレッドの切替をいかに小さなコストで実現するかが最も重要な課題となる．スレッドの切替のコストを削減するためには，以下の方法が考えられる．

- スレッドを切替えるための処理そのものに要する時間を短縮する．
- スレッドの実行に独立性を保証した上で，スレッドを切替えるための処理とスレッドの実行を重畳化する．

スレッドを切替えるためには，例えば，現在のスレッ

ドのコンテクストを退避し，新しいスレッドのコンテクストを復帰する必要がある．RWC-1では，コンテクストの退避領域のポインタを，スレッドを指定する命令アドレスとの組であるcontinuationとして持ち回るようにしており，主としてレジスタ上のコンテクストが退避/復帰の対象となる．なお，RWC-1では，パケットのペイロードとして最大8つのデータを扱うことができるので，ネットワークの負荷が高くなければ，最大8つまでのレジスタ上のデータをcurrentのコンテクストとして持ち回ることもできる．

2.4 細粒度並列処理機能

“細粒度”の並列処理を効率良く実行するために，RWC-1では以下のようなハードウェアが拡張されている．

- パケットキュー
  ネットワークから到達したパケットは，一旦，このパケットキュー上に格納される．パケットキューは実行優先順位に応じて3つ用意されている．
- スレッド起動機構
  現在実行中のスレッドの実行が終了した時点で，パケットキュー上に格納されたパケットから，最も実行優先順位が高いものを1つ取り出し，パケット上の命令アドレスで指定されるスレッドを起動する．
  現在実行中のスレッドよりも実行優先順位が高いパケットが到着した場合には，現在実行中のスレッドの実行を中断し，新しいパケットが指定するスレッドを起動する．このようなプリエンプションが生じた場合，パケットはあたかも実行優先順位の変更をともなう分岐命令のように振舞う(分岐先のアドレスはパケットの上の命令アドレスによって指定される)．
- パケット注入機構
  命令アドレスに後続するペイロードデータをパケットの属性によって指令される領域のレジスタに格納する．このデータの書き込みは，命令の実行とは独立して行なわれる．
- パケット生成/送出パイプライン
  命令によってパケットの生成/送出が指示されると，パケット生成/送出パイプラインは，ヘッダ情報を生成しネットワーク上に送り出すともに，ペイロードとして運ぶデータをレジスタから読み出しネットワーク上に送り出す．このデータの読み出しは命令の実行とは独立して行なわれる．なお，パケット生成/送出パイプラインは1つの命令によって起動されので，パケットの生成/送出は途中で中断することはない．つまり，パケットの生成/送出は不可分である．

パケットキューとスレッド起動機構により，ハードウェアがスレッドの実行スケジューリングを行い，直接的に命令フローを制御する．このため，商用のプロセッサなどで一般的に行なわれている割り込みを用いたメッセージハンドリングなどのオーバヘッドは一切なく，スレッドを起動するまでに要する時間は考えられるものの中で最も短い．この点がRWC-1で採用したRICAの最も大きな特徴の1つとなっている．なお，パケットキュー上のパケットは実行可能状態のスレッドを指定するので，パケットキュー上に1つ以上のパケットがあり現在実行中のスレッドが明示的に実行を終了した場合，スレッドの起動とスレッドの切替えは等価である．つまり，スレッドの起動に要する時間が短いということは，スレッドの切替えそのものに要する時間も短いことを意味している．

次にコンテクストの退避/復帰とスレッドの実行との重畳化を考える．スレッドを切替えるために退避/復帰すべきコンテクストは，(1)パケットのペイロードとして持ち回るものと，(2)メモリ上に退避/復帰するもの，に大別することができる．これらの中で，パケットのペイロードとして持ち回るコンテクストの退避は，命令によって起動されたパケット生成/送出パイプラインによって，命令の実行とは独立して行われる．また，パケットのペイロードとして持ち回るコンテクストの復帰（新規のロード）は，パケット注入機構によって，スレッドが起動された直後から，命令の実行とは独立して行なわれる．レジスタのリード/ライトパスにおいて競合が生じなければ，このような機構によって，コンテクストの退避/復帰とスレッドの実行を重畳化でき，スレッド切替の見かけ上のコストを小さくできる．

3. 細粒度並列処理機能の強化

3.1 スレッドの先行起動

スレッドを切替えるためには，概して，以下のような処理を必要とする．まず，スレッドのコンテクストをレジスタ上から退避させ，次いで，スレッドの実行を終了させる．さらに，起動するスレッドを選択し，そのスレッドのコンテクストをレジスタ上に復帰させる．最後に，命令フローを更新し新しいスレッドの実行を開始する．

RWC-1の仕様では，パケットのペイロードを汎用レジスタ上に格納することになっている．このようにすると，データの依存関係を管理することが可能となり，コンテクストの復帰とスレッドの起動の順序を逆転させることができる．命令フェッチには時間を要するため，このような制御は，命令実行パイプラインのvoidを最小限に抑えることに大きく寄与している．このような制御を採用した場合の動作を具体的に示すと以下のようになる．(1)実行中のスレッドの実行が終了した時点で，直ちに新しいスレッドが起動される．(2)命令実行パイプラインは，新しいスレッドの命令がまだデータの格納されていないレジスタを参照しようとした時点で，始めて，ストールする．

依存関係を管理するために必要となるregister score

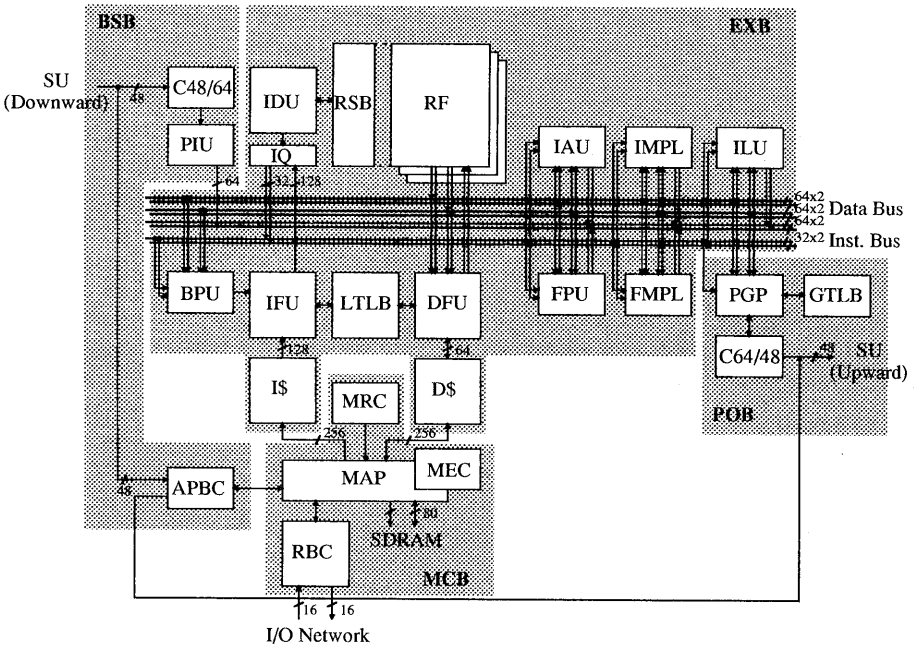


図 2　プロセッサの内部構成
Fig. 2　Processor configuration

board などには命令実行のために用意されたものをそのまま流用するため，上記のような制御を採用してもハードウェア的なコストはほとんど生じていない．なお，説明の都合で前後したが，このようなスレッドの先行起動によって，始めて，パケット注入機構によるコンテクストの復帰とスレッドの実行を重畳化することができる．

3.2　ldm/stm 命令の拡張

RWC-1 では，レジスタ上のコンテクストをメモリ上に退避 / 復帰させるために，最大 8 つまでのデータを連続してメモリ上にセーブ / ロードできる ldm/stm(load/store multipe) 命令を用意している．この命令によって連続したレジスタ / メモリ間のデータ転送が起動され，データの転送は後続する命令の実行とは独立して行なわれる．

この ldm/stm 命令は，パケット生成 / 送出パイプラインとパケット注入機構によるコンテクストの退避 / 復帰を補完するものである．以下の例に示すように，ldm/stm 命令を用いることによって，スレッド切替の見かけ上のコストをさらに削減することが可能である．

ldm/stm を用いた場合のコンテクストの退避 / 復帰は以下のように行なわれる．まず，スレッドは，パケット生成 / 送出パイプラインを起動し，かつ，stm 命令によってレジスタ上のコンテクストのメモリ上への退避を指示したあと，実行を終了する．その後，直ちに，新しいスレッドが起動され，それに後続してパケット注入機構によりパケットのペイロードがレジスタ上に格納される．一方，スレッドの先頭で ldm 命令が発行されメモリ上に退避されたコンテクストがメモリ上に格納される．一般に書き込みパスの制約の方が大きいため，RWC-1 においても，この例では，レジスタの書き込みパスで競合が発生し，ldm 命令に後続する命令実行が数クロックだけストールする．

3.3　データパスの拡張

レジスタ間転送アーキテクチャを採用した RWC-1 では，命令の実行とコンテクストの退避 / 復帰を重畳化するために，レジスタファイルのリード / ライトパスの拡張が必要となる．一般に，このリード / ライトパスの拡張はハードウェアコストが高くプロセッサの設計上大きな負担になる可能性がある．例えば，レジスタファイルのリード / ライトパスを拡張しようとすると，チップの占有面積が大きくなってしまうだけではなく，アクセス時間が遅くなるといった問題がある．そこで我々は，もっとも効率的なインプリメントとして，multi-instruction issue のために拡張されたアクセスパスを流用する方式を提案している．プロセッサの内部構成を図 2 に示す．この方式の場合，コンテクストの退避 / 復帰を行なわない場合，そのリード / ライトパスを命令実行に使用することができ，性能の低下をもたらすことはない．

multi-instruction issue の観点から言うと，コンテク

ストの退避 / 復帰を行なう場合，1(最大で2) だけ少ない命令しか実行できなくなるので性能が低下する可能性がある．ところが，パケットのデータの注入を考えると明らかなように，パケットのデータがレジスタ上に格納されなければ，プロセッサがストールするため，コンテクストの退避 / 復帰によるパス使用は実際上ほとんど問題とならない．また，新しいスレッドを起動した場合，命令フェッチにおいてキャッシュミスが生じる可能性が高いが，この場合には，外部メモリから命令コードを読み込む遅延が大きく，その間にペイロードの書き込みが完了するため，命令実行には全く影響を与えない．

3.4 専用命令の拡張

スレッドを切替えるためには，コンテクストを退避 / 復帰させるだけではなく，並列処理のための準備が必要である．例えば，RWC-1 では再実行開始点を指定する continuation などを作成する必要などがあるが，このような準備作業を単純化するために，コンテクストが格納されるデータ領域のアドレスとスレッドの命令アドレスを 64bit に組み込む専用の命令などを用意している．

4. PE ノード間通信

PE ノード間でいかに高い通信性能を実現するかが，並列計算機における最も重要な課題の1つである．通信性能を評価する性能指標としては，基本的に，遅延とスループットが考えられる．これらは PE ノードを相互に接続するネットワークの構成などによって大きく左右されるが，一方では，PE ノードにもネットワークの性能を十分引き出すための用意が必要である．ネットワークの高い通信性能を十分引き出すことができる PE ノードを設計する上で最も大きな障害となるのは，使用する部品の物理的な形状に起因する実装上の制約である．最も大きな制約は，LSI のピンボトルネックとボード間接続用コネクタのピンボトルネックである．例えば，PE ノードとネットワーク間のデータ転送のスループットをいかに高くするかという問題は，まず，PE ノードを構成する LSI から何本の信号線を引っ張り出せるかという問題に帰着される．この問題に対する回答として，RWC-1 では，図に示すように PE ノードを命令を実行するプロセッサ (PE) と通信を管制する2つのスイッチングチップ (SU) に分割している．PE ノードのデータパス構造を図3に示す．

4.1 PE と SU の分割

PE にはメモリをアクセスするポートが必要で，メモリアクセスのバンド幅を確保するために，これとネットワークを構成するポートの共存は難しい．PE と SU を分割すれば，PE はネットワーク用のポートを1つだけ持てば良く，ピンボトルネックの回避が容易である．また，SU からもネットワークを構成する複数のポートの信号を取り出すことが可能となる．なお，SU は PE ノードを相互に接続するケーブルをドライブする必要があり大きな発熱が想定されるため，熱的設計上からも分割が好ましい (RWC-1 では，SU には集積度は低いけれどタフな BiCMOS LSI を，PE には高集積の最先端テクノロジによる CMOS LSI を採用している)．

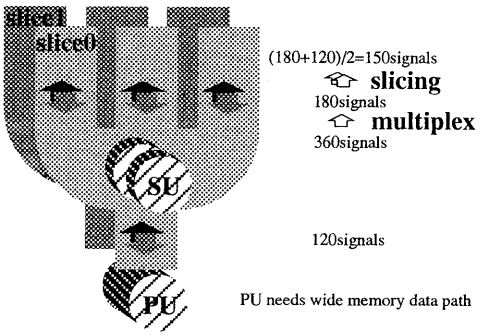


図3 PE ノードのデータパス構成
Fig. 3 Data path configuration of PE nodes

4.2 SU のスライス

SU の機能はデータ交換である．データ交換のためには，すべてのポートが1つの LSI に集約されているほうが好ましい．ところが，ピンボトルネックのためその実現は難しい．RWC-1 では，SU をデータ幅方向に2分割することによって，データ交換のためにアドレス情報をどのように分配するかといった技術的な課題は残るものの，この課題を実現している．

4.3 SU の倍速動作

PE ノード間におけるデータ転送のスループットを高めるには，データ転送の周期を短くするか，転送するデータの幅を大きくすれば良い．RWC-1 では，データ転送の周期を 1/2 とすると同時にデータ信号線数を 1/2 とすることにより (マルチプレクス / デマルチプレクス)，同一のスループットを維持しながらポート当たりの信号線数を 5/8 程度に削減している．転送周期をより短くすれば，ポート当たりの信号線数を更に削減できるが，分割した部分に対し一定数の制御線が必要になるためデータ線の幅が少なくなった場合の削減効果が小さいことと，後述するように，ケーブルの周波数特性上の問題から，RWC-1 では，m を2に留めている (ここで，m はマルチプレクスの多重度)．

4.4 ボード間接続

RWC-1 の実装設計は，メインテナンス容易性を重視した設計となっている．例えば，一般的なエッジボードコネクタを採用すると，通常のバックプレーン配線の他に，前面からもケーブルを導出することが可能である．しかし，このような実装を採用すると，PE ノードで故障が発生した場合，このケーブルを脱着するためにシステムの運用を停止させる必要が生じる．超並列計算機では，数多くのプロセッサを稼働させるわけであるから，システムの availability を向上させるために，RWC-1

では，実装上のメインテナンス容易性を重視して，バックプレーン側からの配線のみでPEノード間の接続を行なっている．

このようなエッジボードコネクタを用いた実装でポイントとなるのはコネクタの物理的な形状である．基板の物理的なサイズには限界があり，基板のエッジの長さをコネクタのピン間幅で割ったn倍が基板から導出できる信号線数である（ここで，nは接点のスタック数）．つまり，接続する信号線数を多くするためには，ピン間幅を狭くれば良い．コネクタは機械的な接点を有するため，ピン間幅を狭くすることには限界があるものの，2mm～1.27mm間隔のコネクタが実用されている．RWC-1では，2mm間隔を用いて，1200信号程のボード間接続を実現している．

4.5 データ転送周波数

このような限られた信号線数で，できるだけ高いデータ転送スループットを実現するためには，データ転送周期をできるだけ短くしなければならない．ところが，物理的に大きな形状を持つ超並列計算機であるRWC-1では，PEノード間を接続するケーブルの線長最大で10mを越える可能性がある．このケーブル線長がRWC-1の実装では大きな障害となっている．

PEノード間接続に使用するケーブルの直径は，コネクタのピン間幅によって決まるため，使用できるケーブルは外形が，通信分野で使用されるケーブルと比較すると，かなり細いものである．このようなケーブルでは時定数が大きく，高周波のデータ転送では，信号の立ち上がり／立ち下がりが劣化する．このような問題を回避するためには，例えば，ケーブルのドライブ能力を強化するか，あるいは，ケーブルの長さを短くすれば良い．しかし，後者は実装上の要求から不可能で，前者にはLSIの発熱上の制限がある．また，外部に放熱に関して考慮がなされたバッファを搭載する方式も考えられるが，このようなバッファを搭載すると要素プロセッサの実装面積が大きくなり，多くのプロセッサを実装する必要がある超並列計算機での採用は困難である．以上のような理由から，RWC-1では，PEノード間接続のネットワークのデータ転送周波数を100MHzとしている．なお，さらに高いPEノード間の通信性能を実現するために，我々は新情報処理開発機構の分散研究室と共同で光インタコネクションの採用に関する技術的検討を進めている[7]．

5. まとめ

超並列計算機RWC-1の要素プロセッサの概要について述べた．RWC-1の要素プロセッサは通信と処理を融合し単純化したアーキテクチャであるRICAを採用し，細粒度の並列処理を効率良く実行することができる．本稿では，スーパスカラプロセッサのデータパスを利用することによって現実的なコストでRICAを実装する方式を提案した．我々は，RWC-1の要素プロセッサの開発にあたり，プロセッサを稼働させるために必要とする支援システムを開発し，ソフトウェアの開発環境を構築してきた．今後は，提案した方式のプロセッサチップを量産し，支援システムの中に組み込むことによって，並列コンピュータとして稼働させ，システムとしての性能評価を進めていく予定である．

謝辞 本研究を遂行するにあたり，有益な御指導，御討論をいただきました島田つくば研究所長，超並列ソフトウェア研究室の諸氏，光日立研究室，光NEC研究室の担当諸氏に感謝いたします．

参考文献

1) 坂井修一, 岡本一晃, 松岡浩司, 廣野英雄, 児玉祐悦, 佐藤三久, 横田隆史: 超並列計算機RWC-1の基本構想, 並列処理シンポジウムJSPP'93, pp. 87–94 (1993).
2) 松岡浩司, 岡本一晃, 廣野英雄, 横田隆史, 坂井修一: 超並列計算機RWC-1用プロセッサチップの設計, 信学技報, CPSY95-18, pp. 55–62 (1995).
3) 坂井修一, 松岡浩司, 岡本一晃, 横田隆史, 廣野英雄, 児玉祐悦, 佐藤三久: RWC-1のシステム構成と基本動作, 情処研報, ARC-113-24, pp. 185–192 (1995).
4) 岡本一晃, 松岡浩司, 横田 隆史 廣野英雄, 坂井修一: RWC-1のマルチスレッド処理機構, 情処研報, ARC-113-26, pp. 201–208 (1995).
5) Yokota, T., Matsuoka, H., Okamoto, K., Hirono, H., Hori, A. and Sakai, S.: A Prototype Router for the Massively Parallel Computer RWC-1, *Proc. Int. Conf. on Computer Design (ICCD'95)*, Austin, Texas, pp. 279–284 (1995).
6) Yokota, T., Matsuoka, H., Okamoto, K., Hirono, H., Hori, A. and Sakai, S.: The Multidimensional Directed Cycles Ensemble Networks for a Multithreaded Architecture, *Proc. Int. Conf. on High Performance Computing (HiPC)*, New Delhi, India, pp. 355–360 (1995).
7) Nishimura, S., Inoue, H., Hanatani, S., Matsuoka, H. and Yokota, T.: Optical Subsystem Interconnections for the Massively Parallel Computer RWC-1, *Proc. of Optoelectronics and Communication Conference(OECC)*, Chiba, Japan, pp. 17–17 (1996).