並列推論エンジン PIE64 の推論ユニットのアーキテクチャ

The architecture of the inference unit of Parallel Inference Engine PIE64

日高 康雄, 小池 汎平, 田中 英彦
Yasuo HIDAKA, Hanpei KOIKE, Hidehiko TANAKA
東京大学 工学部
University of Tokyo, Faculty of Engineering

概要

高並列推論エンジン PIE64 は, 64 台の推論ユニットと 2 系統の相互結合網から構成されている. 1 台の推論ユニットは, UNIRED, NIP, SPARC の 3 種類のプロセッサと, それらの間で共有されるローカルメモリを持つ. 推論ユニット内部の処理は, これらのプロセッサの分散協調処理によって進められる. このような協調処理においては, 共有メモリのアクセスとプロセッサ間のコマンド発行がボトルネックとなりやすい. メモリアクセスのボトルネックは, メモリの構成を 3 ウェイ 4 バンクとし, アービトレーションをパイプライン化した同期バスプロトコルを用いることで避けている. また, プロセッサ間を接続するコマンドバスには, アービトレーションオーバヘッドのない同期バスプロトコルを用いて, コマンド発行のボトルネックを避けている. 本稿では, 推論ユニット内部の分散協調処理と, それを効率良く実行するローカルメモリの構成, メモリバス, コマンドバスのプロトコルについて述べる.

1 はじめに

我々は, 大規模知識処理のための高並列推論エンジン - PIE64 - の研究, 開発を進めている. PIE64 は, 64 台の推論ユニット (IU － Inference Unit) と呼ばれる要素プロセッサを, 2 系統の相互結合網で接続した構成をしている.

一般に, プロセッサ数の多い高並列の MIMD 型並列計算機の要素プロセッサには, 次のような事柄が要求される. すなわち,

1. 強力な同期機構を持つこと.
   高並列計算機では, 並列度を高く保つために処理の粒度を小さくしなければならない. 粒度を小さくすると, 並行する処理の間の同期を頻繁にとらなければならないため, オーバヘッドの少ない強力な同期機構をハードウェアとして持つことが必要である.
2. レイテンシに強いこと.
   規模の大きい相互結合網を介する要素プロセッサ間の通信が, ある程度のレイテンシを持つことは避けられない. このため, 要素プロセッサは, レイテンシに対して強くなければならない.
3. 単体性能が高いこと.
   逐次性が高く, 並列度の低いプログラムに対しても, 高い性能を出すためには, 要素プロセッサの単体性能も高くなければならない.

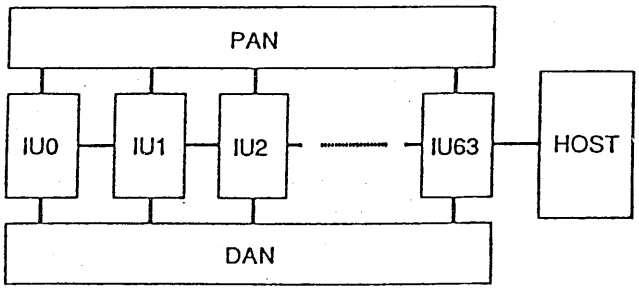
PIE64 の 1 台の推論ユニットは 3 種類 6 個のプロセッサを持つ. 3 種類のプロセッサが, コマンドとリプライのやり取りによって協調処理を行なうことで, 上記の要件が満たされている. 一般に複数のプロセッサで協調処理を行なう場合, 共有メモリアクセス, プロセッサ間のコマンド発行がボトルネックとなりがちであるが, PIE64 の推論ユニットでは, 高性能のメモリバス, コマンドバスを持つことでこのボトルネックを避けている.

まず, 2 節では, PIE64 と推論ユニットについて概説する. 3 節では, 3 種類のプロセッサの分散協調処理モデルについて述べる. 4 節では, 推論ユニットのハードウェア上の特徴であるメモリバス, コマンドバスについて述べる. そして, 5 節で予測性能を述べ, 6 節で本稿をまとめる.

2 並列推論エンジン PIE64

並列推論エンジン PIE64[1] は, 64 台の推論ユニットと 2 系統の自動負荷分散機能を持つ相互結合網とから構成される MIMD 型の並列計算機で, Fleng のプログラムを高速実行する専用マシンである. PIE64 の全体構成を図 1 に示す.

Fleng[6] は, コミティッドチョイス型言語, 並列論理型言語と呼ばれる並列記号処理向けのプログラミング言語の一つであり, Flat GHC からガードゴールを省いたも



IU - Inference Unit
PAN - Process Allocation Network
DAN - Data Allocation Network

図 1: PIE64 の全体構成

のとほぼ同じである.

推論ユニットは, 図 2に示すように, 推論処理プロセッサ (UNIRED), ネットワークインタフェースプロセッサ (NIP), 管理プロセッサ (SPARC), ローカルメモリ (LMEM), SPARC 用メモリなどから構成されている.

UNIRED[5] は, Fleng のプログラムを実行するための専用プロセッサであり, ユニフィケーション, リダクションを効率良く実行する命令セットを持っている. 命令は全て 32 ビットの固定長であり, RISC 型の先行制御パイプラインによって, 各命令をパイプラインピッチで高速実行する. ほとんどのメモリ参照命令は, ローカルメモリの参照とリモートメモリの参照を自動的に判別し, リモートであれば NIP にリモートメモリをアクセスするコマンドを発行する.

UNIRED の持つ大きな特徴に多重コンテクスト処理がある. これは, 最大 4 つのコンテクストを並行実行するもので, NIP からのリプライを待つ間も他のコンテクストでパイプラインを充足し, 実行効率を維持する. UNIRED と NIP 間の機能分担と UNIRED の多重コンテクスト処理によって, 推論ユニットはリモートメモリ参照のレイテンシに強くなっている.

NIP[3] は, 相互結合網とのインタフェースを持ち, 推論ユニット間の通信処理と Fleng プログラムのプロセス間同期処理を行なうプロセッサである. 相互結合網の経路制御側を Master NIP, 被制御側を Slave NIP がそれぞれ担当する. 相互結合網が 2 系統あるため, 1 台の推論ユニットは 4 個の NIP を持つ.

UNIRED や SPARC が Master NIP に対してコマンドを発行すると, Master NIP は, 相互結合網を介して接続した先の Slave NIP と共に通信処理を行なう. Slave NIP は, 通信処理の他に, Master NIP からの指示に従って, Fleng プログラムの同期処理を行ない, 推論ユニットに強力な同期機構を提供する.

SPARC は, ゴールキューの管理, メモリ管理, 浮動少数点演算, 入出力処理などの様々な処理を行なう. これらの処理を行なうための汎用のマイクロプロセッサを持つことによって, UNIRED を Fleng に特化したアーキテクチャとすることができ, さらに, ゲートアレイ 1 チップでの実現やパイプラインピッチでの命令実行が可能となっている. SPARC と UNIRED の協調処理によって, 推論ユニットは高い単体性能を発揮する.

これら 3 種類 6 個のプロセッサは, コマンドバスによって結ばれ, 3 ウェイ 4 バンク構成のローカルメモリを共有する. また, SPARC は専用の SPARC バスと SPARC 用メモリを持ち, 他のプロセッサの 2 倍のクロックで動作する. そして, SPARC は, IO バスインタフェース (IOIF), IO バスを介して, ディスクなどの周辺機器をアクセスすることができる. フロントエンドとなるホスト計算機からは, ホストバス, ホストバスインタフェース (HOSTIF) を介して, SPARC バスをアクセスすることができ, SPARC のアクセス可能な資源を全てホスト計算機からアクセスすることができる. コマンドバスとローカルメモリについては, 4 節で詳しく述べる.

3 推論ユニット内の分散協調処理モデル

Fleng の実行処理は, 推論処理, 通信・同期処理, 実行管理に機能分割され, 推論ユニット内の UNIRED, NIP, SPARC が, コマンドとリプライのやり取りをしながら協調処理する. この分散協調処理の概略を図 3に示す. 以下に, 主要なコマンドを 5 種類に分類して説明する.

1. ゴール実行に関するコマンド
   ゴールのユニフィケーションおよびリダクションは, SPARC が UNIRED に reduce コマンドを出すことによって開始される. UNIRED は, ゴールがフォークした時, 実行を終了した時, ユニフィケーションに失敗した時にそれぞれ, newgoal, enddreduce, fail コマンドを SPARC に出す.
2. サスペンドに関するコマンド
   UNIRED は, ユニフィケーション中にサスペンドが起きると, suspend, suspendend コマンドによって, サスペンドを起こした変数へのポインタを SPARC に渡す. SPARC はサスペンションレコードを作成し, suspend コマンドを master NIP に出す. master NIP は, 相互結合網を介したその Slave NIP に依頼して, 変数のサスペンションレコードリストへそのレコードを追加する.
3. 変数のバインド及びアクティベイトに関するコマンド
   変数のバインドは, UNIRED や SPARC から NIP に bind コマンドを出して行なうのが基本である. リモート変数の場合は Master NIP に, ローカル変数の場合は Slave NIP にコマンドを出す. bind コマンドを受けた Master NIP は, 相互結合網

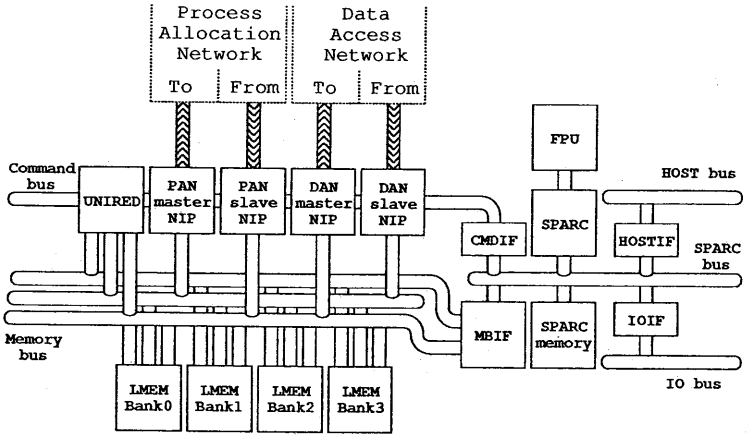


図 2: 推論ユニットの構成

介した先の Slave NIP にバインドを行なわせる. Slave NIP はバインドを行なった後, アクティベイト動作, つまり, 変数のサスペンションレコードリストを読み出して, SPARC または Master NIP に activate コマンドを出す. また, 変数同士のバインドにおいては, Master NIP に suspend コマンドを出して, サスペンションレコードリストのマージを行なう. 例外は, UNIRED がローカル変数のバインドをする時で, バインド自体は UNIRED 自身が行なった後, Slave NIP に activates コマンドを出して, アクティベイト動作を行なわせる.

4. リモートメモリ読みだしに関するコマンド
   UNIRED や SPARC は, リモートメモリの読み出しや, デレファレンスのために, read, deref コマンドを Master NIP に出す.
5. 負荷分散に関するコマンド
   SPARC は, 他の推論ユニットにゴールを転送する時に, また, UNIRED は, 他の推論ユニット上のヒープを割り当てる時に, Master NIP に writem, writel コマンドを出す. 特定の推論ユニットを指定する場合は writem コマンド, 相互結合網の自動負荷分散機能を用いる場合は writel コマンドを使う. 転送終了後, 転送先の Slave NIP は同じ推論ユニット上の SPARC 宛に, newmsg, newload コマンドを出す.

上記のコマンドの他に, デッドロック回避関係, ガーベジコレクション関係, メモリ管理関係, エラー処理関係のコマンドがある.

4 推論ユニットのハードウェア

前節で述べたような分散協調処理の実現においては, メモリアクセス, コマンド / リプライ発行の衝突とアービトレーションオーバヘッド, SPARC のソフトウェアによるコマンド発行 / 受信のリアルタイム性が問題となる. これらを解決するために, ローカルメモリを 3 ウェイ 4 バンク構成とし, メモリバス, コマンドバスを同期バスとし, バスプロトコルをアービトレーションオーバヘッドの少ないものにした. また, SPARC のコマンドバスインタフェースには FIFO を用いて, ソフトウェア制御による遅延を吸収している. 以下では, ローカルメモリとコマンドバスについて述べる.

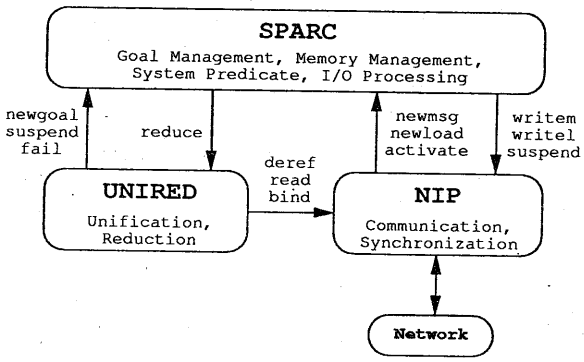


図 3: 推論ユニット内の分散協調処理

4.1 ローカルメモリ, メモリバス

ローカルメモリには, UNIRED の命令コード, Fleng のゴールフレーム, ヒープ上のデータ, ガーベジコレクション時の間接参照テーブルなどを格納する. ローカルメモリは 4M バイトの容量を持っており, 4 つのバンクに分けられて, 3 本のメモリバスを介してアクセスされる.

1 本のメモリバスは, 32 ビットのデータ, 20 ビットのアドレス, 数本の制御信号線とバスアービタからなっている. また, それぞれのバンクはバンクアービタを持って

表 1: プロセッサ間の主要なコマンド一覧

分類	From	To	コマンド	説明
ゴール実行	SPARC	UNIRED	reduce	ゴール実行の開始
	UNIRED	SPARC	newgoal	フォークゴールのキューへの登録
			endreduce	ゴール実行終了の通知
			fail	ゴール実行失敗の通知
サスペンド	UNIRED	SPARC	suspend suspendend	サスペンドの通知
	SPARC	Master NIP	suspend	リモート変数へのサスペンドの登録
		Slave NIP	suspend	ローカル変数へのサスペンドの登録
変数のバインド, アクティベイト	UNIRED	Master NIP	bind	リモート変数へのバインド
		Slave NIP	activates	一連の activate コマンドの発行
	SPARC	Master NIP	bind	リモート変数へのバインド
		Slave NIP	bind	ローカル変数へのバインド
	Slave NIP	SPARC	activate	ローカルゴールのアクティベイト
		Master NIP	activate	リモートゴールのアクティベイト
			suspend	サスペンションレコードのマージ
リモートリード	UNIRED, SPARC	Master NIP	read[12nx]	リモートメモリの読みだし
			deref	リモートポインタのデレファレンス
負荷分散	SPARC, UNIRED	Master NIP	writem[12nx], writel[12nx]	ゴール転送, リモートヒープ割り当て
	Slave NIP	SPARC	newmsg, newload	転送されたゴールの通知

いる. 各プロセッサは, まずバスアービタにリクエストを出し, バスの使用許可を受けてアドレスを出力する. 続いて, バンクアービタからバンクのアクセス許可を受け, 次のサイクルで実際のメモリアクセスを行なう.

ロックをかけているプロセッサは 3 本のバスのアドレスを監視し, 衝突時にはロック信号を出力する. ロック信号が出されたバスには, バンクのアクセス許可が返らない.

メモリのアクセスは, バンクのアクセス許可を受けるまでのアービトレーションサイクルと, 実際のアクセスを行なうアクセスサイクルとに分けて行なわれるため, 1 回のアクセスには 2 サイクル (1 サイクルは 100n sec) を要する. しかし, この二つのサイクルはパイプライン化されており, 毎サイクルのデータ転送が可能になっている.

三本のバスには, それぞれ以下のプロセッサ等を接続する.

1. UNIRED(命令フェッチ), MBIF(ダイレクトアクセス)
2. UNIRED(データリード), PAN master NIP, DAN slave NIP
3. UNIRED(データライト), PAN slave NIP, DAN master NIP, MBIF(ステージングアクセス)

これは, UNIRED の各ポートの性質と各 NIP のアクセス頻度に関する考察を元に決定した.[4] また, Fleng 処理系の実装においては, 2 系統の相互結合網の使い分け方, ローカルメモリの使用方法などに関して各種方式を検討する必要があるため, 様々な方式に適するように, 各アービタの優先順位交換の方式, バンク分けに用いるアドレス線等は, ソフトウェアによって数種類の中から選択できるようになっている.

4.2 メモリバスインタフェース (MBIF)

SPARC バスとメモリバスをつなぐメモリバスインタフェース (MBIF) は, ダイレクトアクセスとステージングアクセスを提供する. ダイレクトアクセスは, SPARC バスのアドレス空間上に, 4M バイトのローカルメモリを直接マッピングするものである. ステージングアクセスは, FIFO メモリとアドレスカウンタを使って, ローカルメモリ上の連続する領域を読み書きするものであり, ダイレクトアクセスよりも高い転送効率を提供する.

4.3 コマンドバス

コマンドバスは, 6 つのプロセッサ間でやりとりされるコマンドとリプライを転送するための 32 ビット幅のバスである. コマンドは 1 〜 3 ワードであり, コマンド

の種類によってリプライがある場合とリプライがない場合がある．コマンドの第一ワードは，上位2ビット，下位3ビットの4ビットでコマンドのコードを表し，中位の23ビットはコマンドの引数に使われる．このため，推論ユニット番号とアドレスのみを引数とするコマンドは，1ワードで表される．また，構造データやゴールフレームなどの2ワード以上のデータは，実体をローカルメモリ上において，1ワードのポインタのみをコマンドの引数に含めるので，殆んどのコマンドは1～2ワードで構成されている．リプライは全て1ワードである．

コマンドバスは，32ビットのデータ，3ビットのアドレス，数本の制御信号線とバスアービタからなっている．コマンド，リプライを送出するプロセッサは，バスアービタにリクエストを出し，バスの使用許可を受けて，アドレス，データ，コマンド/リプライの種別などを出力する．コマンド送出の場合はアドレスに送出先のプロセッサ番号を指定し，リプライ送出の場合はアドレスに自分のプロセッサ番号を指定する．他のプロセッサは自分宛のコマンドかどうかを監視し，コマンドを受け付けられない場合はビジー信号を返す．また，リプライを待つプロセッサは，発行先のプロセッサがリプライを出しているかどうかを監視する．

各プロセッサのリクエストは，10MHzのクロックに同期して出力され，そのサイクル中にデータの先頭ワードの転送までを行ない，データの転送は最終ワードまで10MHzのクロックに同期して行なわれる．このため，数ワードのデータ転送毎のアービトレーションのオーバヘッドは存在しない．なお，NIP, UNIREDのレジスタの読み書きもコマンドバスを介して行なわれる．

4.4 コマンドバスインタフェース (CMDIF)

SPARCバスとコマンドバスの間のコマンドバスインタフェース (CMDIF) は，以下の特徴を持っている．

- SPARCから発行するコマンド，SPARCが受け取るコマンドは，それぞれ512ワードのFIFOメモリに格納され，SPARCのソフトウェアがコマンド転送上のネックとならないようにしている．また，FIFOとは別にリプライ受け取りレジスタを持ち，最後に発行したコマンドに対するリプライを受け取ることができる．
- FIFOの状態によって発生できるSPARCへの割り込みが2本あり，割り込みを起こす条件 (スレシホールドレベル，以上/以下など) をソフトウェアで柔軟に設定できる．これを用いて，FIFOのフロー制御を小さなオーバヘッドで行なうSPARCのプログラムを実現できる．
- 発行側のFIFOに書き込まれたコマンドをキャンセルして，コマンドバスへの送出を止めることができる．また，SPARCは，発行をキャンセルされたコマンドを読み出すことができる．これは，ガーベジコレクション開始時における通常処理の停止の時などに使われる．
- デバッグモードの設定により，SPARC以外のプロセッサに対するコマンドを受け取り側のFIFOに取り込むことができる．

5 予測性能

UNIREDとNIPのクロックは10MHzで，UNIREDの性能はappend/3で最大1.25MLIPSの予定，NIPの性能はリストセルの読み出しで約0.5MOPSの予定である．SPARCおよびFPU(SPARCのコプロセッサ) は20MHzのクロックで動作し，それぞれの性能は，カタログ値で12MIPS[7]と2.54MFLOPS[8]である．メモリバス，コマンドバスは，UNIRED, NIPと同じ10MHzのクロックに同期して動作し，コマンドバスの転送能力は，最大40M Bytes/sec，メモリバスの転送能力は，バス1本あたり最大40M Bytes/sec，バス3本で最大120M Bytes/secである．また，ローカルメモリは最小2サイクル (200n sec) でアクセスでき，コマンドおよびリプライは最小1サイクル (100n sec) で発行できる．推論ユニットのハードウェア上の諸元を表2にまとめる．

6 おわりに

本稿では，高並列MIMD計算機の要素プロセッサの要件として，強力な同期機構を持つこと，レイテンシに強いこと，単体性能が高いことの3つを上げ，それらを満たす推論ユニット内の分散協調処理モデルについて述べ，さらに，分散協調処理を実現する推論ユニットのハードウェアについて，ローカルメモリ，コマンドバスの構成を中心に述べた．

現在，推論ユニットは回路設計を1990年5月中旬に終了し，基板の設計，製造を行なっている．今後は，推論ユニットハードウェアの調整，デバッグを行ない，実測による推論ユニットの性能評価，SPARCの制御プログラムの開発，Flenng処理系の実装などを行なっていく．

謝辞

基板の設計，製造，実装を担当したヨシキ電子株式会社に深謝いたします．また，富士通株式会社には，メモリICを提供していただきました．なお，本研究は文部省特別推進研究No.62065002の一環として行なわれている．

参考文献

[1] 小池，田中：“並列推論エンジンPIE64”，並列コンピュータアーキテクチャ，bit臨時増刊，Vol.21, No.4,

表 2: 推論ユニットの諸元

項目		仕様
SPARC	IC	富士通製 S-20
	クロック, 性能	20MHz, 12MIPS
	SPARC用メモリ	SRAM 512K Byte, no wait
FPU	IC	WEITEK 製 Abacus 3170
	クロック, 性能	20MHz, 2.54MFLOPS
UNIRED	IC	富士通製 CMOS ゲートアレイ
	実ゲート数	約35000 ゲート
	パッケージ	256 ピン PGA
	クロック, 性能	10MHz, 最大1.25MLIPS (予定)
	コンテクスト数	4 コンテクスト
	汎用レジスタ数	32 本 (1 コンテクストあたり)
NIP	IC	富士通製 CMOS ゲートアレイ
	パッケージ	256 ピン PGA
	実ゲート数	約20000 ゲート
	クロック, 性能	10MHz, 約0.5MOPS (予定)
	個数	1IU 当たり 4 個
ローカルメモリ	容量	SRAM 4M Byte, 4 Bank
	メモリバス	3 本
	メモリバススループット	最大 40M Bytes/sec (バス 1 本)
		最大 120M Bytes/sec (バス 3 本)
	1 ワード転送	最小 2 サイクル (200n sec)
コマンドバス	スループット	最大 40M Bytes/sec
	1 ワード転送	最小 1 サイクル (100n sec)
使用 IC	40 ピン以上の LSI	12 個
	256Kbit 高速 SRAM	144 個
	PAL	82 個
	TTL	242 個
	合計	480 個

1989, pp. 488-497.

[2] 高橋, 小池, 田中: "並列推論マシン PIE64 の相互結合網の作成および評価", 並列処理シンポジウム '90 A1-1, 情報処理学会, May 1990.

[3] 清水, 小池, 田中: "並列推論マシン PIE64 の推論ユニット間通信", 計算機アーキテクチャ研究会 79-4, 情報処理学会, Nov. 1989.

[4] 清水, 小池, 田中: "PIE64 のネットワーク・インタフェース・プロセッサのシミュレーションによる性能評価", 第 40 回 情報処理学会 全国大会, 1L-8, Mar. 1990.

[5] 島田, 下山, 清水, 小池, 田中: "推論プロセッサ UNIREDII の命令セット", 計算機アーキテクチャ研究会 79-5, 情報処理学会, Nov. 1989.

[6] Nilsson, M.: *Parallel Logic Programming for SIMD Supercomputers and Massively Parallel Computers*, Doctorate Thesis, Info. Eng. course, Univ. of Tokyo, Mar. 1989.

[7] Fujitsu Microelectronics, Inc.: *SPARC MB86901 (S-25) High Performance 32-Bit RISC Processor – Product Description*, 50 Rio Robles, Bldg. 3, San Jose, CA 95134-1804, 1988.

[8] WEITEK Corp.: *Abacus 3170 Floating-Point Coprocessor for SPARC*, 1060 East Argues Avenue, Sunnyvale, CA 94086, May 1989.

[9] Jordan, H.F. : *Performance Measurement on HEP – A Pipelined MIMD Computer*, Proc. of the 10th Annual International Symposium On Computer Architecture, Stockholm, Sweden, June 1983, pp. 207-212.

[10] Arvind and Iannucci, R.A. : *Two Fundamental Issues in Multiprocessing*, Proc. of DFVLR – Conference 1987 on Parallel Processing in Science and Engineering, Bonn-Bad Godesberg, West Germany, June 1987.