マトリクス・スイッチ結合式マルチプロセッサシステムのバッファ合せの一方式†

山本 登††

密結合マルチプロセッサシステムでは，共有メモリを書き変えるとき，その写しをもつ他の処理装置のキャッシュの内容との不一致を解消する機構が不可欠である．本論文では，2，30 台の処理装置群と主記憶装置群とを，キャッシュ内蔵のマトリクス・スイッチで結合したシステムに適した，実用的なバッファ合せ機構を実現する方法を述べている．

1. まえがき

2，30 台の処理装置群と記憶装置群とを，マトリクス状のスイッチで結合した，マルチプロセッサシステムは，メモリ参照要求に対する応答特性を，処理装置が記憶装置を占有した場合の値に近づけることにより，その並列処理性を生かし，1台の超大型機以上に，広範囲の応用に適用される可能性をもっている．しかし，このシステムの実用化には，以下に述べる問題を解決する，実用的な方式を確立する必要がある．

（1） この形式のシステムの実現を困難にしている最大の要因は，スイッチ部の実現に，LSI 化に適さない膨大な量の回路が必要なことである[1]．

（2） 共有メモリへの書込み動作（ストア）に伴い，その写しをもつ他の処理装置のキャッシュと，共有メモリの内容とが一致しなくなる現象を，解決しなければならない[2),3)]（以後バッファ合せと呼ぶ）．

（1）についてはスイッチ装置にキャッシュを内蔵し，スイッチに要する回路の量を削減する方法を考え，その実現の見通しを得た[4]．本論文ではこの対策が施されたシステムで，（2）の問題，すなわち各処理装置のキャッシュと共有メモリとの内容の不一致を解消する方式を具体化することを試みた．

2. バッファ合せ問題へのこれまでの対応

2.1 バッファ合せの必要性

図1に示すように，共有メモリの領域 A_i の写しが処理装置 P_x のキャッシュに存在するとき，他の処理装置 P_y が同一領域 A_i を書き変えると，領域 A_i の内容と処理装置 P_x のキャッシュの内容とは一致しなくなる．この問題を解消する有効な機構をもたない限り，実用的な密結合マルチプロセッサシステムを構成することはできない．

2.2 従来技術

従来のバッファ合せの方式は，図2に示すように，ある処理装置がストア動作をするとき，他の処理装置群にストア対象領域のアドレスを送り，受け取った処理装置は，その都度そのアドレス領域の内容のキャッシュへの取込みの有無を調べ，もしあれば，そのブロックを無効とする方式であった[3]．ストア動作の割合はメモリ参照総数の 10〜30% ともいわれており[3]，処理装置が多くなると，この検査だけにキャッシュの動作時間の多くが費やされ，本来のメモリ参照動作に使えなくなるため，多数の処理装置を接続するシステムにこの方式を適用することは不可能である．

Tang は主記憶の各ブロックが，各キャッシュ間で共用されているか，独占的に使用されているかを，各処理装置のキャッシュのデレクトリに記憶するとともに，ストアコントローラと呼ぶ装置に各処理装置のデレクトリと同一内容の情報をもたせ（これを中央デレクトリと呼ぶ），両者の連係により矛盾を解決する原理的な方式を示した[2]．この方式では共用属性のブロックは複数のキャッシュに存在できるが，占有属性のブロックは占有権をもつ1台の処理装置にのみ存在する．たとえば処理装置Aは自分が占有権をもつブロックには自由に読書きできるが，他の処理装置Bがそのブロックの参照をストアコントローラに要求すると，占有権は消滅し，そのブロックの主記憶への書出しを命じられるとともに，該当ブロックは処理装置Bの要求がストアなら処理装置Bの占有となり，読出しなら共用属性に変わる．

一方 Censier らは，記憶装置の1ブロックごとに処

† A Solution to the Cache Conflict Problem of a Tightly Coupled Multi-processor System by NOBORU YAMAMOTO (Department of Electrical Engineering, Faculty of Engineering, Nihon University).

†† 日本大学工学部電気工学科

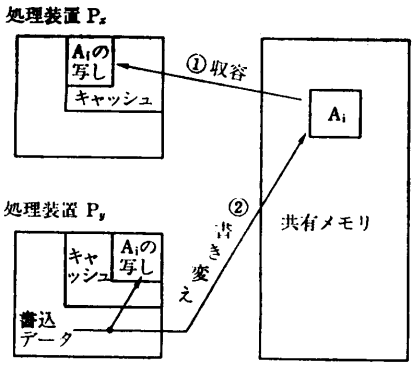

図 1 バッファ合せの必要性
Fig. 1 The occurrence of the buffer conflict.

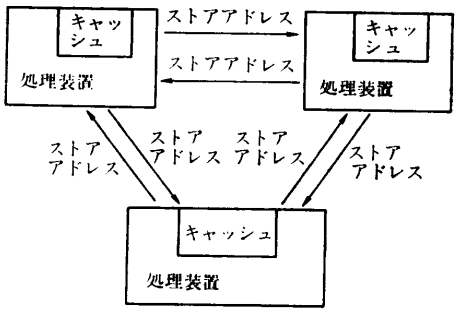

図 2 バッファ合せ方式（従来方式の例）
Fig. 2 A solution to the cache conflict problem (an old example).

理装置単位に1ビットの存在表示子（presence flag）を設け，各処理装置のキャッシュへの複写の有無を表示させるとともに，ブロックに1ビットの書変え表示子（modified flag）を設けることで，Tang と同一の機能を実現可能なことを示した[3]．

しかし，これらの方式では各ブロックの占有／共有情報をシステムに一つしかない装置で管理するため，そこへの参照の競合をさけるため，ストアイン方式のキャッシュをもつシステムにしか適用できない．このため占有権が他の処理装置に移るたびに，占有権のあったキャッシュから主記憶へ，主記憶から新たに占有権を得たキャッシュへと，ブロックの情報の転送が必要なほか，上述の占有／固有情報管理部と各処理装置間でこれらに伴う信号の授受や，制御の同期化が必要となる．このため制御が複雑となるほか，占有権の移動に時間がかかるという欠点がある．そのうえ，占有／固有情報が主記憶を対象に設けられるため，主記憶容量の増大とともにこれらの管理情報を記憶するメモリの容量も増大するという欠点がある．

そこで以下では，キュッシュ内蔵のスイッチ装置を用いた密結合マルチプロセッサシステムにふさわしい，バッファ合せ方式を検討する．

3. 対象システムの構成

本論文で検討の対象とするシステムの構成と特徴を以下に示す（図6参照）．

（1） p 台の処理装置は，それぞれ処理装置インタフェース（P接続路）により，スイッチ装置に接続される．スイッチ装置の内部とは処理装置接続部（PIFC）を介して接続される．PIFC 部は参照アドレス情報の特定の領域を解読し，後述の m 組のメモリ参照制御部の一つに，参照要求を転送する．

（2） それぞれ独立に動作する m 群の主記憶装置は，記憶装置インタフェース（M接続路）によりスイッチ装置に接続される．スイッチ装置の内部とは記憶接続部（MIFC）を介して接続される．

（3） スイッチ装置には $p \times m$ 組の交点をもつマトリクス・スイッチ部があり，p 台の処理装置と m 組の記憶装置群とを接続する．メモリ参照に付随する情報はこのスイッチ路により伝送される．

（4） 各処理装置にはキャッシュ（固有キャッシュ）が内蔵されるが，スイッチ装置にもキャッシュ（共有キャッシュ）が内蔵される．

（a） 両キャッシュのブロックは同じ大きさで，システムのデータバス幅（b バイト）と同じ容量をもつ n 個のサブブロックからなる（n は2の巾乗の値）．

（b） キャッシュの参照アドレスを，論理アドレスと実アドレスのいずれにするかは，非常に重要な検討項目である．しかし本論文の主要な論点とは直接関係はないので，その詳細は別途報告するとして，ここでは固有キャッシュ，共有キャッシュとも論理アドレスで参照されるものとする．この論理アドレスからキャッシュのアドレスへの写像は，セットアソシアテブ方式を用いて行う．

（c） 固有キャッシュはストアスルー方式を用いるが，固有キャッシュに書き終えると共有キャッシュへのストア動作の終了を待たずに，次の処理を開始する．一方，共有キャッシュはストアイン方式を用いるため，ストアは共有キャッシュのみに行われ，主記憶から転送後内容が書き変えられたブロックが置換される場合のみ，主記憶に書き戻される．

（5） 1回のストア動作で書かれる情報は，システムのデータバス幅の範囲内である．したがって1バイトの場合もあれば b バイトの場合もある．そこでどの

バイト位置が書き変わるかを表示するため，bビットの書込みフラグを設け，参照アドレスや，書込みデータとともに処理装置からスイッチ装置へ送出する．

（6） スイッチ内部でのメモリ参照の主要なシーケンス制御は，メモリ参照制御部（MAC）が行う．たとえば各 PIFC からのメモリ参照要求のなかから，あらかじめ定められた選択アルゴリズムにより一つを選び，選択した PIFC からスイッチを経由してメモリ参照に必要な情報を受け，まずキャッシュを起動する．ヒットすればキャッシュからの読出し情報をスイッチを経由して PIFC へ送るが，キャッシュミスの場合，キャッシュのデレクトリ検査と並行して行われるアドレス変換動作の結果得られた実アドレスにより，主記憶から読み出すよう MIFC に指示し，読み出した情報をスイッチを経由して PIFC に送るとともに，キャッシュにも書き込むよう各部を制御する．なおアドレス変換は，まず高速アドレス変換バッファ(TLB)による変換が試みられ，これに失敗すると共通アドレス変換部によるセグメント表・ページ表を用いたアドレス変換が行われる．

（7） 実記憶から削除されたページに属するブロックをキャッシュ中に残存させると，後でそれらのブロックが新しいブロックと置換されるとき，主記憶へ書き戻せなくなる．したがって実記憶に存在しないページに属する情報は，演算数としてキャッシュに存在することは許されない．一方，書変えのない命令コードは，主記憶に書き戻す必要はないので，所属するページが実記憶から削除されてもキャッシュに残存できる．しかし任意のブロックの中に演算数として使われる情報があるか否かの判断はできないので，セットされていれば演算数としての参照を禁止する機能をもつ，演算数無効表示子をブロックごとに設け，実記憶から削除されるとき，キャッシュ中のそのページに属するブロックの演算数無効表示子を1にセットする．

（8） 仮想空間が消滅しても，その空間に含まれていたブロックをそのままキャッシュに残留させておくと，後に別のプログラムに同一空間を割り当てたとき，キャッシュに残存している以前のプログラムが誤って使われる可能性がある．このため仮想空間の消滅時，その空間に属するブロックはキャッシュから除かれねばならない．

4. バッファ合せ方式の設計

4.1 バッファ合せの原理

本論文で用いるバッファ合せの原理を以下に述べる．

（1） 処理装置のキャッシュと直接情報を授受するメモリのアドレス空間を，キャッシュのブロックの大きさずつに分割し，その分割単位ごとにその写しが各固有キャッシュに存在するか否かを示す表示子を設ける(複写表示子と呼ぶが Censier の存在表示子と同じ)．この表示子は，システムに接続可能な処理装置の数だけのビット数で一語が構成され（複写表示語），ブロックの数だけの語数をもつメモリ（複写表示メモリ）として実現される．

（2） 複写表示子は，1が固有キャッシュに写しがあることを，0がないことを示し，固有キャッシュにブロックが転送されるとき，そのブロックの複写表示子が1にセットされ，固有キャッシュから削除されるとき，0にリセットされる．

（3） ストア動作のとき，ストア対象のブロックの複写表示語を読み出し，ストア動作中の処理装置を除く他の処理装置で，固有キャッシュにそのブロックの写しをもつものの有無を調べる．もしあれば，それらの処理装置へバッファ合せ情報を送り，バッファ合せを行わせる．

4.2 矛盾の解決法の設計

矛盾の解決方法としては，書き変えられたサブブロック，あるいはブロックの内容を無効にする方法(サブブロック無効化方式，ブロック無効化方式)とサブブロック内の書き変えられた情報のバイト位置のみを書き直す方法（再書込み方式）とが考えられる．これらのどの方式を採用するかを決めるにあたっては，以下に示す評価項目の充足度を検討する必要がある．

（1） キャッシュ内の情報の保存性の確保と，そのため必要となる回路の量やバッファ合せ所要時間の程度．

（2） バッファ合せの発生頻度を減少させるため，必要となる回路量の増大の程度．

（3） バッファ合せの所要時間．

表1は前記の三つの方式について，情報の保存性，バッファ合せの発生頻度，バッファ合せの所要時間，およびバッファ合せに必要な情報という見地から分析したものである．これによれば，ブロック無効化方式と再書込み方式とは，バッファ合せの発生頻度，所要

表 1 矛盾の解決方法の評価
Fig. 1 Evaluation of the solution for the cache conflict.

評価項目＼処理方式		サブブロック無効化方式	ブロック無効化方式	再書込み方式
機能		変更されたサブブロックのみ無効とする.	変更されたブロック全体を無効とする.	変更されたサブブロックのみ書き直す.
情報の保存性		無し	無し	有
バッファ合せの発生頻度		サブブロック内のどこかが書き変えられるときバッファ合せが行われる.	ブロック内のどこが書き変えられてもバッファ合せが行われる.	ブロック内のどこが書き変えられてもバッファ合せが行われる.
バッファ合せの所要時間		固有キャッシュのデレクトリの更新に要する時間と同じ.	同左	同左
必要情報	アドレス	要	要	要
	再書込みデータ	不要	不要	要
	書込みフラグ	不要	不要	要
	複写表示子	サブブロックごとに必要	ブロックごとに必要	ブロックごとに必要

時間，複写表示子の必要量とも同じで，書き変えられた部分を書き直すか，ブロック全体を無効とするかの点が異なるのみである．書き直すには，書き変える情報とともに書き変えるバイト位置を示す情報があればよい．これには3章で述べた書込み表示子を用いればよい．これだけの情報を追加するのみでキャッシュの内容を保存することができるのだから，性能を重視する立場からは再書込み方式を採用したい．

一方サブブロック無効化方式は，他の方式ではブロック内のどの情報が書き変わっても，バッファ合せ要求が発生するのに対し，その処理装置で参照しない情報を含むサブブロックは，バッファ合せの対象から除外していき，バッファ合せの頻度を減少させようとするものである．しかし，複写表示子はサブブロック単位に必要となるため，複写表示メモリの容量は増大する．また一つのブロック中にその処理装置が参照する情報が連続して分布している場合，この方式はその効果を発揮することができないこともあり，必ずしも性能価格比のよい方式とはいえない．

以上の検討から，多少はハードウェアの量が増大しても，性能の向上を優先するシステムでは，再書込み方式を採用するのが妥当といえる．

4.3 複写表示語の設置対象

複写表示語の設置対象として，論理空間，実記憶空間，および共有キャッシュの空間の3案が候補として考えられるが，次の評価尺度を考慮して設置対象を決める必要がある．

複写表示メモリの必要量

バッファ合せ検査の所要時間

複写表示語の滅却制御の容易性

(1) 複写表示メモリの容量

表示対象が大きいと複写表示メモリの必要量も大きくなる．近年汎用電子計算機では 2^{31} 程度の論理アドレス空間をもつようになった．このため，論理アドレス空間を対象に複写表示語を設ける方法は，複写表示語の設置を，各ジョブに実際に存在するプログラム空間の分だけに限るとしても，各処理装置が別々のジョブを実行する場合，その必要量は膨大となるため，採用することは不可能である．

また，実記憶を対象とした場合も，現在の大型システムでは大容量の実記憶をもっているため，複写表示メモリの容量が大きくなる．たとえば32メガバイトの実記憶，1ブロックの容量が64バイト，接続処理装置数16のシステムの場合，複写表示メモリの必要量は1メガバイトとなる．しかも，このメモリは次の(2)項で述べる速度特性をもつ必要があり，管理用メモリにこれだけの負担をするのは，現在の技術水準では許容されない．

一方，共有キャッシュを対象に複写表示語を設ける場合，先のシステムの例で仮に1メガバイトという大容量の共有キャッシュを設けたとしても，複写表示メモリの容量は先の例の 1/32，すなわち32キロバイトしか必要としないため，実現は容易である．

(2) バッファ合せ時間

ストアは共有キャッシュになされるので，ヒット時は共有キャッシュのストア動作が終了するまでに，バッファ合せ検査は終わっていなければならない．もしそれまでにバッファ合せ検査が終了しないと，次の

メモリ参照動作が開始できないため，システムの性能が低下する．したがって複写表示メモリは，共有キャッシュのデレクトリと同速度で読書きできなければならない．共有キャッシュを対象に複写表示メモリを設ける場合，この条件を満足するよう設計するのは容易だが，他の場合は複写表示メモリの量が多いので困難である．

（3） 複写表示子の消滅の制御

（a） 仮想記憶を対象にした場合，仮想アドレス空間の消滅時に，複写表示子を消去すればよい．

（b） 実記憶を対象にした場合，実記憶に割り付けられていた論理ページがその割り付けを解除されるとき，該当する実ページの複写表示子を消去する．なお，仮想アドレス空間の消滅時も実ページへの割付けが解除されるため，この操作が行われる．

（c） 共有キャッシュを対象に複写表示語を設ける場合，原理上は収容されている任意のブロックが消滅するとき，そのブロックの複写表示語は消去されるべきだが，キャッシュのデレクトリと対になっているため，該当ブロックが無効になりさえすれば，複写表示語の抹消の必要はない．その代り，任意の論理ブロックが主記憶から共有キャッシュの任意のブロックに複写されるとき，複写表示語は初期化されねばならない．

なお，3章(7)，(8)で述べたように，仮想アドレス空間の消滅時は，該当空間に属するブロックの無効化操作が，また，実記憶のページ置換時は，演算数無効表示子を1にセットする操作が必要となる．

以上の検討の結果，複写表示メモリの容量が大きくなるため，仮想アドレス空間や実記憶を対象にする方法は実用的でない．一方，共有キャッシュを対象に複写表示子をもつ方式は，その必要容量が少ないことが，共有キャッシュの参照時間内にバッファ合せ検査を終了するための設計をも容易にする．一見するとキャッシュの無効化や，演算数無効表示子の操作の機会が多いようだが，実際には仮想空間の消滅や実ページの置換の機会は，システム内部の動作速度に比べ多くはなく，致命的な性能低下要因とはならない．

4.4 複写表示メモリの構成

複写表示メモリは，共有キャッシュの列数に等しい語数をもつメモリが，それぞれ共有キャッシュの行ごとに独立に動作可能な形式で，共有キャッシュの行単位に設けられたもので，記憶内容は別として，共有キャッシュのデレクトリと同一の動作特性をもつ．このため，共有キャッシュのデレクトリ中に含まれても，論理的に矛盾は生じない．**図3**は共有キャッシュの情報メモリ部，そのデレクトリ部，および複写表示メモリ部との対応を示す一つの実現例である．

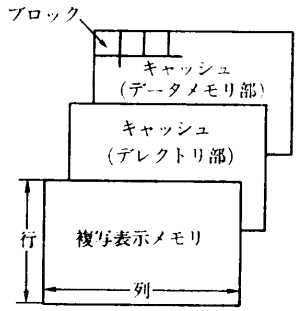

図3 複写表示メモリと共有キャッシュの関係
Fig. 3 The relation between the copy indicator memory and common cache.

4.5 バッファ合せ検査機構

4.5.1 機能要件

バッファ合せ検査機構は共有キャッシュの行単位に設けられ，次の機能を具備する必要がある．

（1） ストア動作のとき，各行のバッファ合せ検査機構は参照アドレスから列アドレスを求め，列アドレスに対応するブロックの複写表示語を読み出し，ストア動作中の処理装置を除く他の処理装置の複写表示子で，1になっているものの有無を調べる．デレクトリ検索の結果，ヒットした行に対応するバッファ合せ検査機構は，バッファ合せの要否とバッファ合せの必要な処理装置の機番を出力する（検査機能）．

（2） 処理装置からの読出し要求の場合，各行のバッファ合せ検査機構は，参照列アドレスに対応するブロックの複写表示語を読み出しておく．デレクトリ検索の結果，ヒットした行のバッファ合せ検査機構は，読出し要求を発した処理装置の複写表示子を1にして複写表示メモリに書き戻す（更新機能）．

（3） キャッシュミスの場合，参照アドレス領域を含むブロックを主記憶から共有キャッシュに転送するが，この収容先となったブロックの複写表示語は，参照中の処理装置のビットのみ1とし，複写表示メモリに書かねばならない（初期設定機能）．

（4） 処理装置で固有キャッシュのブロックが置換されるとき，演算数無効表示子が0なら，複写表示子を0にリセットするようスイッチ装置に要求する．スイッチ装置ではこの要求を受けて，要求した処理装置の複写表示子を0にリセットする（更新機能）．

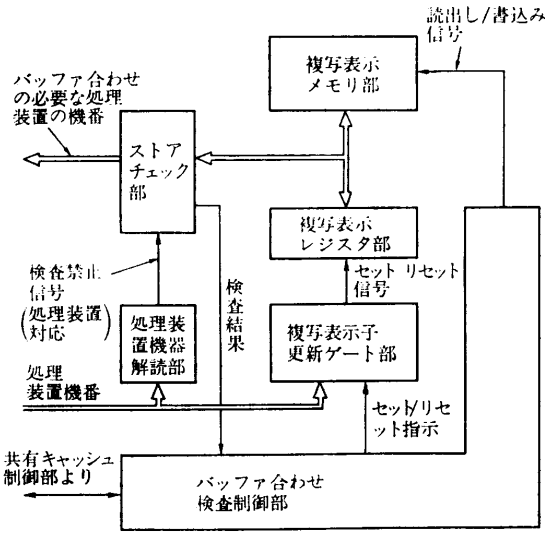

図 4 バッファ合せ検査機構の内部構成
Fig. 4 The internal structure of the buffer conflict detecting mechanism.

4.5.2 バッファ合せ検査機構

図4にバッファ合せ検査機構のブロック図を示す.

(1) バッファ合せ検査制御部は，共有キャッシュ制御部からバッファ合せ検査，複写表示語の更新や初期設定等の指示を受け，複写表示メモリの読書きの制御，複写表示レジスタの更新や，初期設定の制御を行う.

(2) ストアチェック部は，複写表示メモリの出力によりバッファ合せ検査を行う．なお，ストア動作中の処理装置はこの検査対象から除外するため，処理装置機番解読部の出力を用いている.

(3) 複写表示子更新ゲート部は，複写表示語の初期設定や更新のため，複写表示レジスタのセット信号やリセット信号を作る.

図4は共有キャッシュの1行分であり，共有キャッシュがn行からなる場合はn組必要となる．しかし，たとえば1行につき1024列のキャッシュで，接続処理装置数が16の場合でも，複写表示メモリは2キロバイトしか必要としない．また，バッファ合せ検査回路をはじめ，その他の回路も単純なため，LSI 化は容易であり，実現上大きな負担にはならない.

4.6 バッファ合せ情報伝送路の設計

4.6.1 伝送路の必要条件

(1) バッファ合せ要求の同時発生

スイッチ装置内のm組のMACは，それぞれ独立に動作している．したがって複数の MAC から同時にバッファ合せ要求が発生することがある．この場合，MAC にバッファ合せ情報の送達の責任をもたせると，バッファ合せ情報を送達し終わるまで，次のメモリ参照要求の処理を開始できず，システムの性能が低下する．これを避けるため，次の機構が必要となる.

必要条件1 複数の MACから同時に発生したバッファ合せ要求を待たせることなく MAC から受け取り，MAC に代わり必要な処理装置へ送達する機構.

(2) 特定の処理装置への集中

バッファ合せ要求が特定の処理装置に集中する場合が考えられる．この場合バッファ合せ要求を受けきれないと，待たされている MAC は後続のメモリ参照要求の処理を開始できず，システムの性能が低下する．このため次の機構が必要である.

必要条件2 特定の処理装置にバッファ合せ要求が集中しても，MAC のもつバッファ合せ情報を即座に受け取れる機構が必要である.

(3) 複数の処理装置へのバッファ合せの発生

任意の処理装置が書き変えた共有キャッシュの任意のブロックが，複数の処理装置の固有キャッシュに複写されている場合がある．この場合のため，次の機構が必要となる.

必要条件3 バッファ合せ情報を複数の処理装置へ同時に伝送し，最後の受信先が受信し終わるまで送信し続けるとともに，全部の送達先が受信し終わったことを認知する機構が必要となる.

4.6.2 バッファ合せ情報分配器

以下に述べる理由により，メモリ参照情報の伝送に使用するスイッチ部は，バッファ合せ情報の伝送に使用できないので，スイッチ装置内では別途専用の伝送路を設ける.

理由1：バッファ合せ要求が時間的に先に発生したため，先にバッファ合せ情報の伝送が行われ，メモリ参照動作の終了時に返送される応答情報（読出しデータなど）の返送がその間待たされ，システムの性能が低下するおそれがある.

理由2：複数の MAC から同一の PIFC へ同時にバッファ合せ情報を送ることがあるが，PIFC のスイッチ部とのインタフェースは一つなので，同時に複数の MAC からバッファ合せの情報を受けることはできない．このため複数の MAC 間で送出順序を調整する必要があり，スイッチの制御が複雑になる.

図5にバッファ合せ情報分配器のブロック図を示す.

（1） 必要条件2を満足させるため，PIFC部にn段のバッファレジスタを設け，n個のバッファ合せ情報を収容する（nは任意の整数）．

（2） 必要条件1を満足させるため，m組のMAC対応にバッファ合せ情報を記憶するレジスタを設ける．このレジスタが空いている限り，MAC部からのバッファ合せ情報を待時間なしで受け取れる．

（3） m組のレジスタから先入れ先出し方式でバッファ合せ情報を取り出し，送達すべき処理装置に関する情報とともに，後述のバッファ合せ情報配送部へ送達を依頼し，送達の終了を待つ制御機構を設ける（選択部）．

（4） 必要条件2を満足させるため，p組のPIFC部を単一のバスで接続する．

（5） 選択部からの要求を受け，バッファ合せ情報をバスに送出するとともに，送達すべきPIFC部に受け取るよう指示し，それらのすべてが受け取り終わるのを待ち，受信し終わるとその旨選択部へ連絡する機能をもつ制御部を設ける（バッファ合せ情報配送部）．

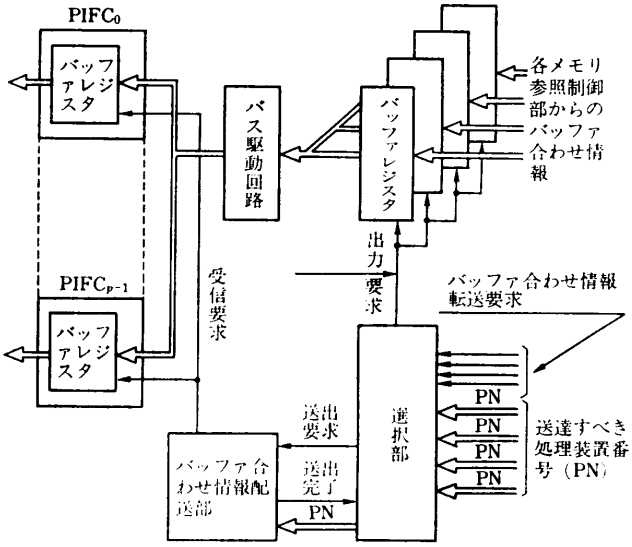

図5 スイッチ装置内のバッファ合せ情報伝送系路
Fig. 5 The transmission path of the information necessary for adjusting cache conflict in the switch.

4.7 同期化機構およびバッファ合せ機構

スイッチ装置内のm組のMACと処理装置群とは，互いに非同期で動作している．したがってバッファ合せ要求をスイッチ装置から処理装置へ伝送しようとする場合，受信する処理装置は次のいずれかの状態にある．

S1：メモリ参照動作は起動されていない．

S2：メモリ参照動作が起動され，処理装置の内部で動作が進行している．

S3：スイッチ装置にメモリ参照要求を発し，その終了を待っている．

S1の状態にあれば，バッファ合せ情報を受信し，バッファ合せ処理を開始できるが，S2，S3の状態ではメモリ参照動作の終了を待たねばならない．またS1の状態にあっても，いつ次のメモリ参照動作が開始されるかわからないので，次のメモリ参照動作の開始を抑止する必要がある．そこで処理装置内では次の同期化規則によりバッファ合せ制御を行う．

（1） 処理装置内ですでに開始されたメモリ参照動作は，最後まで実行させる．

（2） 後続のメモリ参照動作の開始を抑止するため，メモリ参照禁止信号を設ける．本信号はスイッチ装置からのバッファ合せ要求が存在する限り1となっている．処理装置はこの信号が1の間，および現にバッファ合せ処理が進行中の間は，メモリ参照動作は開始できない．

（3） スイッチ装置内のPIFC部は，バッファ合せ情報がある限り，バッファ合せ要求を送出し続けるが，最後のバッファ合せ要求を処理装置に送り終わったら，バッファ合せ要求信号を0としなければならない．

（4） 処理装置は一つのバッファ合せ処理を終了したとき，スイッチ装置からのバッファ合せ要求がまだ存在するかどうか検査し，あればバッファ合せ情報を受信してバッファ合せ処理を再開し，なければ待機させていたメモリ参照要求の処理を開始する．

4.8 バッファ合せを考慮したシステム構成

本節ではこれまでに検討してきたバッファ合せ方式を実現する各要素を，システム全体の視野から概観する．**図6**はバッファ合せに関連した要素を含めたシステム全体の構成を示したものである．複写表示メモリを含むバッファ合せ検査部は，共有キャッシュ部の一員として，共有キャッシュ制御部の直接の制御下で動作する．ここからのバッファ合せ情報は，MAC部を経由してバッファ合せ情報分配器に送られ，図5に示した接続路によりバッファ合せの必要な処理装置に接続されているPIFCのバッファに渡される．バッファ合せ情報がバッファ中にあれば，PIFCは処理装置にバッファ合せ要求があることを知らせる．処理装置は

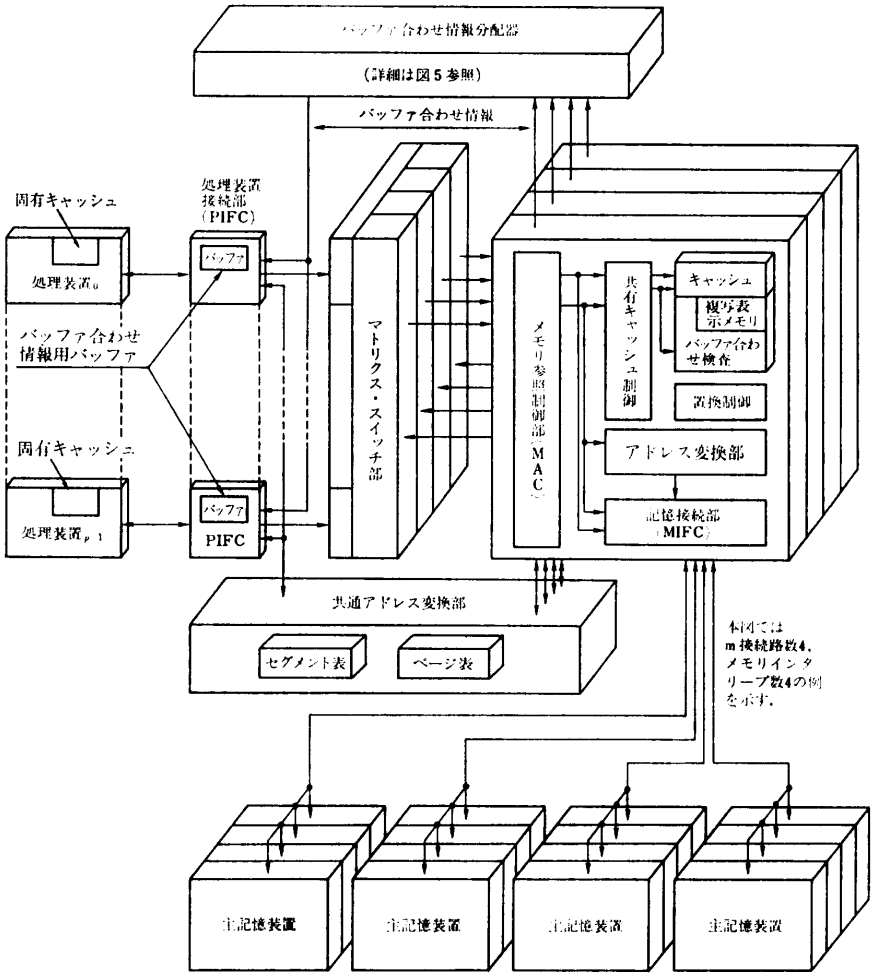

図 6 システム構成
Fig. 6 System configuration.

動作中のメモリ参照動作が終了すると，バッファ合せ情報を PIFC から受けとり，書き変えられた領域を含むサブブロックを書き直す．

5. バッファ合せ方式の評価

ここでは前章で設計したバッファ合せ方式の定性的な評価を行う．

5.1 システムの性能への影響

(1) バッファ合せ検査と複写表示子の更新は，共有キャッシュの動作時間内に終了するため，通常のメモリ参照時間を引き延ばすことはない．

(2) 書き変えられた領域の写しをもっていない処理装置にまで，バッファ合せ要求を発することはない．

(3) 書き変えられた内容で書き直すので，後で同一領域を参照する場合，共有キャッシュから改めて読み出す必要はない．

5.2 バッファ合せ情報伝送路

PIFC のバッファ容量を増加することにより，特定の処理装置にバッファ合せが集中しても，MAC は待時間なしに，バッファ合せ情報をバッファ合せ分配器へ渡すことができる．ただし，バッファ合せ分配器と PIFC 群間は1組の伝送バスで接続されているだけなので，複数の MAC で同時にバッファ合せ要求が発生すると待ちが生じる．これを避けるため，M接続路ごとの専用バスの設置も考えられるが，PIFC の回路規模が増大するため，標準的なシステムでは考えない．

5.3 回路規模

(1) 複写表示メモリは，共有キャッシュを対象に

設けられるため，きわめて少量でよい．

（2） バッファ合せ検査回路は，共有キャッシュの各行ごとに必要である．したがって，M 接続路の数が4でキャッシュが20行の場合，80組のバッファ合せ検査回路が必要になる．しかしバッファ合せ検査回路は単純な構成で必要な回路量も少ないので，1組のバッファ合せ検査回路は 1 個の LSI のなかに全部収容することも可能であることを考えると，実現上大きな負担とはならない．

（3） スイッチ装置内のバッファ合せ情報の伝送路は，アドレス，再書込み情報，および，b ビットの書込みフラグを伝送できる必要があるが，このため必要な回路の量は，実現不可能なほど大きな規模ではない．なお，スイッチ装置と処理装置間はメモリ参照動作に使用するバスを共用するので，バッファ合せ処理用に別途ケーブルを設置する必要はない．

6. むすび

本論文ではストアスルー方式のキャッシュをもつ処理装置群を，ストアイン方式のキャッシュを内蔵したスイッチ装置により，主記憶装置群と接続したシステムにおいて，共有キャッシュを書き変えたとき，その写しをもつ処理装置群のキャッシュとの，内容の不一致を解消する実用的な方法を具体的に示した．ここで示した方式は，以下の2点で特徴をもっている．

（1） Tang の中央デレクトリは，原則としてシステムに1組しか設けられないので，ストアスルー方式のシステムに適用すると，中央デレクトリへの参照の集中が，システムの隘路となる．これを避けるため中央デレクトリを複数設けると，回路やケーブルなどの大幅な増加となり実用的でなくなる．一方，本論文で提案した方式では，複写表示メモリはm組の主記憶インタフェースごとに分散管理され，参照要求は自動的に分散するほか，メモリ参照動作は，スイッチに内蔵された高速のキャッシュに対し行われるので，短時間に終了する．このため，処理装置に内蔵されるキャッシュがストアスルー方式の場合にも対応できる．

（2） Tang の共有/占有属性表示子や，Censier の存在表示子などは，実記憶を対象に設けられるので，実記憶の容量が大きくなるとともに，これら表示子に必要な管理用メモリの必要量も大きくなり，実用的でなくなる．一方，本論文の方式では，表示子はスイッチ装置に内蔵されるキャッシュを対象に設けられるため，その必要量はわずかである．

参考文献

1) Quatember, B.: Modular Crossbar Switch for Large Scale Multiprocessor Systems—Structure and Implementation, Proc. of the AFIPS, pp. 125-135 (1981).
2) Tang, C. K.: Cache System Design in the Tightly Coupled Multiprocessor System, Proc. of the AFIPS, Vol. 45, pp. 749-753 (1976).
3) Censier, L.M. and Feautrier, P.: A New Solution to Coherence Problems in Multicache Systems, *IEEE Trans. Comput.*, Vol. C-27, No. 12, pp. 1112-1118 (1978).
4) 山本：マルチマイクロプロセッサシステム・MμPS-1 のバッファメモリ方式とその効果，情学全大講演論文集，pp. 113-114 (1981).

(昭和59年 6 月 1 日受付)
(昭和59年10月18日採録)