

疎結合計算機システムにおける 高速計算機間通信方式

HIGH-SPEED INTER-PROCESSOR COMMUNICATION

IN LOOSELY COUPLED MULTIPROCESSOR SYSTEMS

中野 良平

RYOHEI NAKANO

森 道直

MICHINAO MORI

日本電信電話公社

横須賀電気通信研究所

YOKOSUKA ELECTRICAL COMMUNICATION LABORATORY, N.T.T.

1. まえがき

疎結合多重計算機(LCMP)システムは、密結合多重計算機(TCMP)システムと比べて以下の利点があり、注目されて来ている。

- (1) 大きなシステム構成が可能である。密結合では通常2系〜4系である。
- (2) システム拡張が容易である。中断なしにシステム拡張ができる。
- (3) 異なるタイプの計算機が結合できる。特徴のある専用計算機の結合、異機種汎用計算機間の結合など。
- (4) システム信頼性を向上できる。

LCMPシステムは負荷分散型と機能分散型に大別できる。前者は計算機間でDASDなどを共用してシステムの負荷を分散し、同時にノーゲウンシステムの構築を図るもので、バンキングなどに適用され始めている。^[1] また、後者は計算機間で機能を分散して処理する形態であり、データベース処理を機能分散するBEPアプローチはこの例に該当する。^{[2][3]} 更に、システムが大形化すると両型の複合も考えられる。^[4]

ほとんどのLCMPシステムにおいて、計算機間通信が行われる。その頻度が多くなると、通信の全体の処理に占める割合が大きくなり、システムとして許容できないものとなる。これに対処するためには、通信回数を減らす工夫も考えられるが、計算機間通信処理そのものの高速化(高性能化)を達成することがより一般的の解決策となる。

また、これまではLCMPシステムの計算機

間通信性能はEXCレベルの低水準のものが見られる。計算機間通信の性能が低水準であると、その使用者側の負担が大きくなる。通信性能を n 種の用途に使うとすれば、最悪の場合には n 個の通信性能を作成することになる。従って、高水準の通信性能を開放することは重要な意味がある。

以上より、LCMPシステムが今後多種のシステムに適用されて行くに当って、高性能でかつ高水準の計算機間通信性能は欠かせないものとなると思われる。

本論文は、LCMPシステムの計算機間通信用に開発した高性能かつ高水準の通信法であるPCAM(Processor-processor Communication Access Method)について、システム条件、性能、高性能化の機構、性能評価などを述べたものである。

2. システム条件

PCAMが走行するシステム構成例を図1に示す。計算機間は各々のチャネル間でチャネル間結合アダプタ(CCA)にて結合する。PCAMは従来のIBM370流のCCA^[5]に新機能^[6]を追加した改良形CCA(改CCAと略称する)を使用している。

改CCAの新機能の概略を述べる。図1において、計算機Aから計算機Cにデータ転送を行う場合を考える。従来のCCAでは、計算機Aからデータを転送する時には、計算機Cに割込み(アテンション割込み)を起こす。これに対

して、改CC Aでは計算機Cに割込みず、転送要求を改CC A上で保留するコマンド（CNWT：Connect Wait）状態を有しており、計算機Cから対応するコマンドが発行されたときに始めてコマンドマッチングが行われ、データ転送を開始する。転送終了後、更にCNWTコマンドは改CC Aでコマンドデインングされており、かつ計算機Cから改CC Aに対して対応するコマンドが未発行のときは、再び改CC A上で保留され計算機Cには割込まない。また、コマンドデインングがない場合には、そのチャネルプログラムは終了し、計算機Aに終了の割込みを起す。このように従来のCCAを改良することにより、転送時の計算機への割込み回数を削減することができる。

この改良はチャネルの使用効率の改善にも寄与している。計算機Aから計算機Cにデータ転送を行う場合、従来のCCAでは、計算機A側のチャネルでチャネルプログラムが起動されると、計算機Cに割込みを起した後、計算機C側から対応するコマンドが発行され転送が終了する迄、チャネルはビジー状態にある。これに対して、改CC Aでは、転送要求が改CC A上で保留されると、チャネルに対してチャネル終了の通知を行うので、チャネルは直ちに別のチャネルプログラムの実行に移ることが可能となる。このように、CCAの改良により、チャネルのビジー時間を短縮することができる。

改CC Aは多数の仮想的なデバイスを持つ。チャネルプログラムはその仮想デバイスを指定して起動される。同一の仮想デバイスに対して、双方の計算機から共に起動が行われ、コマンドマッチングしたときに転送が可能となる。仮想デバイスを以下サブチャネルと呼ぶ。

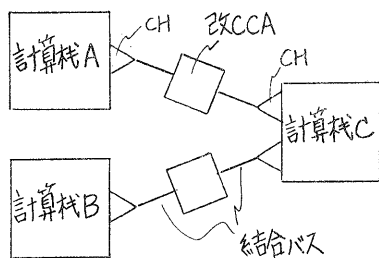


図1. PCAMが走行するシステム構成例

3. PCAM 機能

PCAMは利用者に本章に示す高水準の機能を開放する。

3.1 通信基本要素

PCAM通信は以下に述べる通信基本要素を用いて行われる（図2）。

- (1) 通信プロセス：計算機間通信を行う主体を通信プロセスと呼ぶ。
- (2) 論理バスとポート：
 - (i) PCAM通信は論理バスを指定して行われる。従って、通信する通信プロセス対は通信に先立って論理バスを開設する必要がある。
 - (ii) 各計算機内に論理バスを開設するための窓口としてポートが定義される。論理バスは各計算機内に定義されたポート間改CCAを指定して張られる。
- (3) イベント：
 - (i) PCAMから通信プロセスへの非同期の通知はイベントにて行う。
 - (ii) 通信プロセスは自らが確保したイベントを付して通信要求を行い、通信の終了を該イベントにてウェイトする。
 - (iii) PCAMから通信プロセスへの全く非同期な通知（障害通知など）はPCAMが確保するイベントにて行う。

(4) 送信機能：

- (i) PCAMは送信機能として、SEND、RECEIVE、SAR（Send And Receive）を提供する。SARは送信、受信の一括要求機能である。

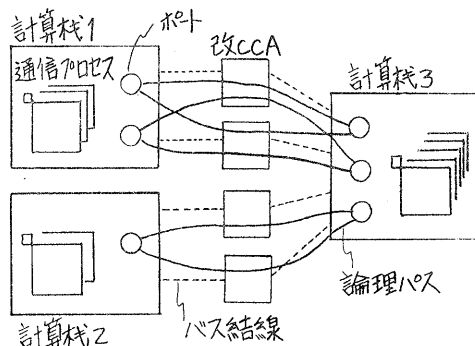


図2. 論理バスとポート

- (ii) 交信要求の撤回にページ機能を使用できる。
 (iii) 障害が原因で論理パスが異常となった場合、障害回復を待合させることもできる。

(5) 通信バッファ

- (i) PCAM通信では、通信バッファは送信用、受信用共に通信プロセスが確保する。
 (ii) PCAMは送信用バッファから受信用バッファにデータをダイレクトに転送する。
 (iii) 通信バッファ域は転送時にはフロッグ化されている必要があるが、性能向上を配慮して、そのフロッグス、フロート制御は通信プロセスに任せる。

3.2 マクロ機能

PCAMが利用者に開放する主要なマクロとその入出力情報および機能内容を表1に示す。

3.3 PCAM機能の特徴

PCAM機能の主な特徴について述べる。

(1) ダイレクトモード転送

一方の通信プロセスが確保した送信用バッファ内のデータを他方の通信プロセスが確保した受信用バッファに中間バッファを経由せず直接転送する機能をダイレクトモード転送と呼ぶ。

一般には、通信データの大きさが事前に分らない、また通信プロセス側の事情とは全く独立に利用は通信機能を構築するといった理由から、通信バッファは通信制御プログラムが管理している。通信プロセスは通信制御プログラムが管理する通信バッファを確保し、そこに転送したいデータをムーブし、送信要求を行う。また、受信時は、通信制御プログラムが受信用通信

表1 主要なPCAMマクロ一覧

項番	マクロ名	入力 情報	出力情報	機能 内容
1	OPEN	改CCAid, 自ポート名, 相手ポート名, オープン種別, リソースジョインバッド	論理パス名	自ポートと相手ポートの間に論理パスを開設する。
2	CLOSE	論理パス名	なし	指定した論理パスを閉塞する。
3	LISTEN	自ポート名, イベントアドレス	OPEN情報	ポート名を指定し相手通信プロセスからの論理パス開設要求を待合させる。
4	GET EVENT	論理パス名, 送信バッファ情報, 受信バッファ情報	交信イベントアドレス	交信要求の終了同期用イベント(交信イベント)を確保する。
5	FREE EVENT	交信イベントアドレス	なし	交信イベントを解放する。
6	SEND	交信イベントアドレス	なし	交信イベント情報をもとにデータを送信する。
7	RECT	交信イベントアドレス, タイマオプション	なし	交信イベント情報をもとにデータを受信する。
8	SAR	交信イベントアドレス, タイマオプション	なし	交信イベント情報をもとにデータを送受信する。
9	PURGE	論理パス名, パラメータ	なし	発行済みの交信要求をページする。
10	RECOVER	論理パス名, イベントアドレス	なし	障害となった論理パスの復旧を待合する。
11	FIX	交信イベントアドレス	なし	交信イベント情報で指定した通信バッファ域をフロッグ属性とする。
12	FLOAT	交信イベントアドレス	なし	交信イベント情報で指定した通信バッファ域をフロート属性とする。

バッファを確保し、そこにデータを受信する。受信する通信プロセスは通信バッファのアドレスを得る(ロケートモード)が、自ら指定した領域にムーブしてもらう(ムーブモード)。このように、通信制御プログラムがバッファ管理を行う方式は汎用であるが、バッファ管理処理(通信バッファの確保/解放)、および通信バッファからの、またそこへのムーブ処理が必要となる。

通信プロセス相互間で、通信条件が規定できる場合には、通信バッファは通信プロセス側で用意する方が効率的である。通信バッファを通信プロセスの業務(アプリケーション)で使用する領域と兼用できれば、上記のバッファ管理処理、ムーブ処理は削除できる。また、兼用できない場合でも、通信プロセス側で自らの通信条件に合った通信バッファを用意すれば、汎用のバッファ管理の処理が削減できる。以上の考えに基づきPCAMはダイレクトモード転送を提供している(図3)。

また、従来は通信バッファのフィックス/フロート制御は通信制御プログラムの中で行っていた。通信バッファがフロート属性の場合、通信時にフィックス化し、通信終了後にフロート化している。しかし、同一通信バッファを用いて何回も通信を行う場合には、フィックス化、フロート化を通信プロセスが最適な契機で行うことにより、従来方式に比べて、処理が削減できる。従って、PCAMでは、通信バッファのフィックス/フロート制御を通信プロセスに任けている。

(2) 半二重通信

PCAM論理バス上は一時には一方向の通信のみが許されている。即ち、半二重通信を提供する。LCMPシステムでは、処理依頼の転送、処理結果の返送の処理が大きな割合を占めると考え、半二重通信の高速化を重視した。なお、全二重通信は半二重論理バスを複数用いることにより構築することが可能である。

(3) 送受信一括要求

PCAMでは、論理バスの半二重特性を生かして送受信一括要求性能SARを提供している。既に述べたように、LCMPシステムでは、処理依頼の転送、処理結果の返送のパターンが大

きな割合を占めると考えられるので、この性能は重要で有効である。SARはSENDとRECVを各々独立に発行する場合に比べて、計算機とチャネルのインタラクション回数を半減させることができ、計算機側の大幅な性能向上の要因となる。

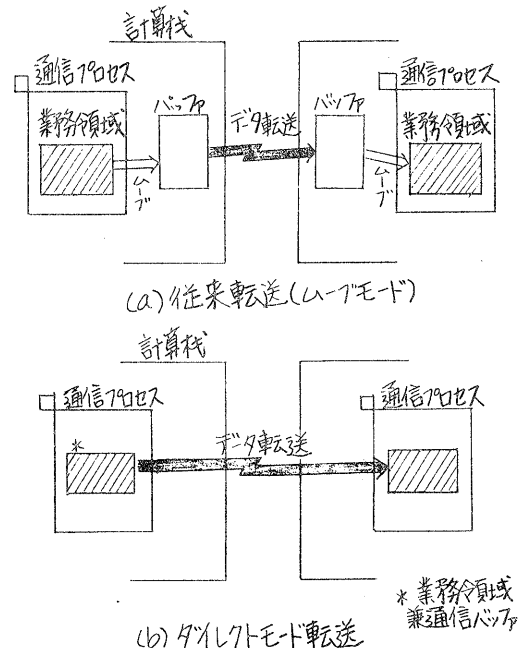
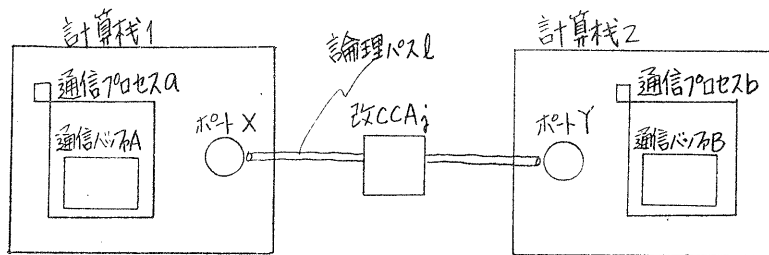


図3. ダイレクトモード転送

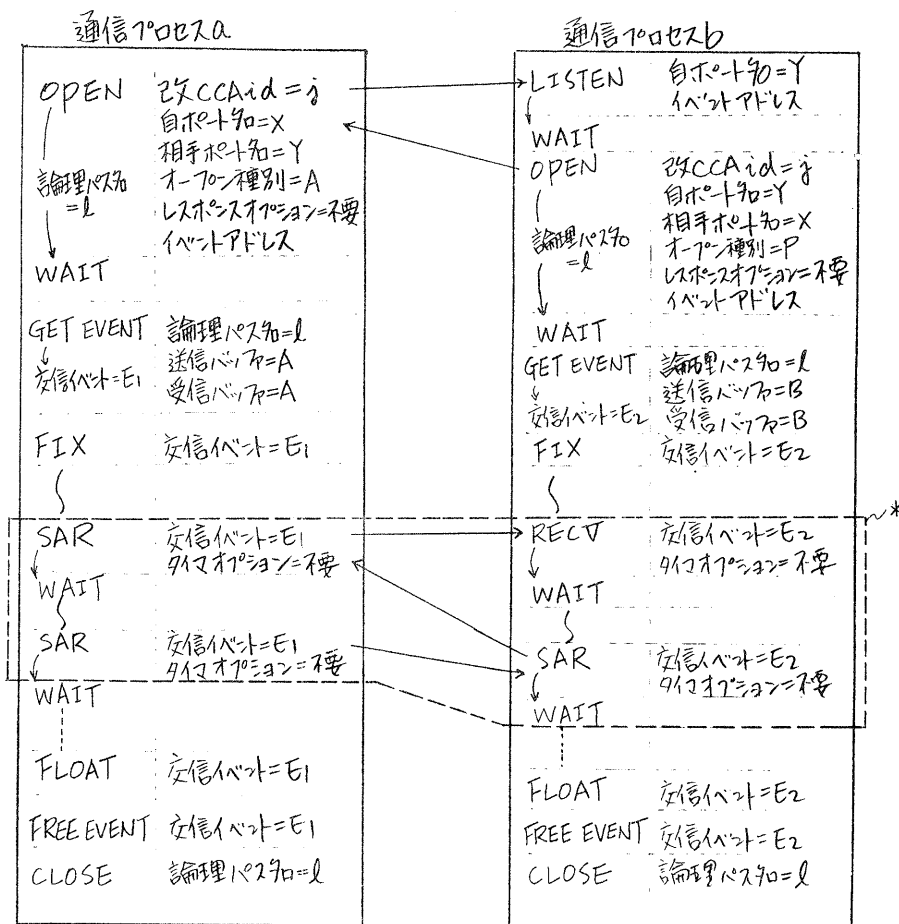
3.4 使用例

使用例として、PCAM性能を最も性能良く用いる例を述べる。システム環境を図4(a)に示す。

通信プロセスa, bは各々の業務領域A, Bを通信バッファとして用いる。通信プロセスa, bの記述例を図4(b)に示す。論理バスの開設に当たっては、通信プロセスbがLISTENで受身となり、LISTEN終了後、オープン種別PにてOPENする。通信バッファA, Bいずれも送信、受信兼用である。いずれのプロセスも交信に先立って通信バッファをフィックス化し、一連の交信の終了後にフロート化している。交信は双方SARを用いているが、最初だけは通信プロセスbはRECVを発行する。なお、交信部分は4章で詳細な機構を説明する。



(a) システム環境



(b) 通信プロセスの記述例

図4. PCAM 使用例

4. 高性能化の枠組

本章では3.4節で示した使用例を例にとり、その通信部分の実現枠組を詳細に述べる。説明の都合上、大きく4個のフェーズに分けて考える。また、通信プロセス内に論理パスは既に開設されており、通信バッファもフロックス化されているとする。説明対象となるシステム環境を図5に示す。

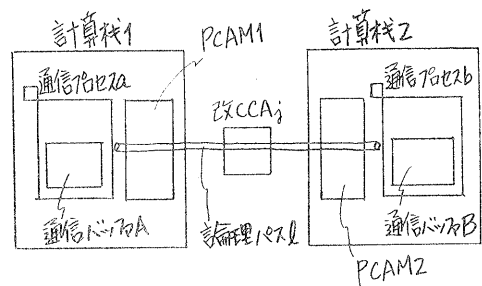


図5 PCAM枠組説明用システム環境

【フェーズ1】(図6)

双方からSARを用いて高速度な通信を行う場合、最初は論理パスとサブチャネルのマッピングが行われる。フェーズ1では、そのマッピングの処理が行われる。

(P1-1ステップ) PCAM2はチャネルコマンド語CCW₂のリードコマンドのデータアドレスに通信情報域C_I₂のアドレスを設定して、計算機1からの受信用サブチャネルに入出力命令を発行しておく。サブチャネルの最初のデータ転送の方向はPCAM1とPCAM2の間で了解がとれている。

(P1-2ステップ) 通信プロセスaは通信バッファAの中に処理依頼内容を格納し、論理パス名₁を指定してPCAM1にSARを要求する。PCAM1は通信情報域C_I₁に通信プロセスaの論理パス名₁、通信種別「SAR」を設定し、チャネルコマンド語CCW₁のライトコマンド1およびリードコマンド1のデータアドレスにC_I₁のアドレスを設定し、ライトコマンド2およびリードコマンド2のデータアドレスに通信バッファAのアドレスを設定して、計算機2への送信用サブチャネルの中から適当なサブチャネルを選び、CCW₁を用いて入出力命令を発行する。

(P1-3ステップ) 改CCA_iはCCW₁のライトコマンド1とCCW₂のリードコマンドのマッピングを確認し、C_I₁からC_I₂へデータを転送する。

(P1-4ステップ) 転送が終了すると、計算機2に入出力割込みが発生する。PCAM2はC_I₂の中の通信種別が「SAR」であるので、サブチャネル情報域S_I₂にサブチャネル番号_i、論理パス名₁、通信種別「SAR」、処理段階「未RECV」を設定する。即ち、ここで論理パスとサブチャネルが動的にマッピングされたことになる。

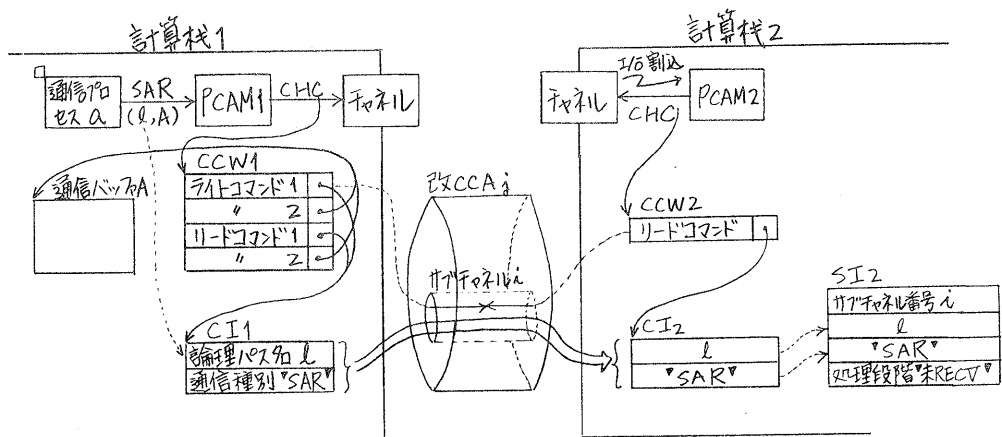


図6 フェーズ1の処理

【フェーズ2】(図7)

通信プロセスは最初はRECVを発行する。フェーズ2はRECV完了までの処理である。
 (P2-1ステップ) 通信プロセスは通信バッファB, 論理バス名 l を指定してPCAM2にRECVを要求する。PCAM2は論理バス名 l のサブチャネル情報域SI2を捜す。SI2の処理段階が「未RECV」なので、PCAM2はCCW2のリードコマンドのデータアドレスに通信バッファBのアドレスを設定し、サブチャネル i に入出力命令を発行する。
 (P2-2ステップ) 改CCA i はCCW2のリードコマンドとCCW1のライトコマンド2のマッチングを確認し、通信バッファAから通信バッファBへデータを転送する。
 (P2-3ステップ) 転送が終了すると、計算機2に入出力割込みが発生する。PCAM2はサブチャネル番号 i のSI2を捜し、通信種別が「SAR」を示しているのを、処理段階に「RECV完了」を設定して、通信プロセスにRECVの終了を通知する。

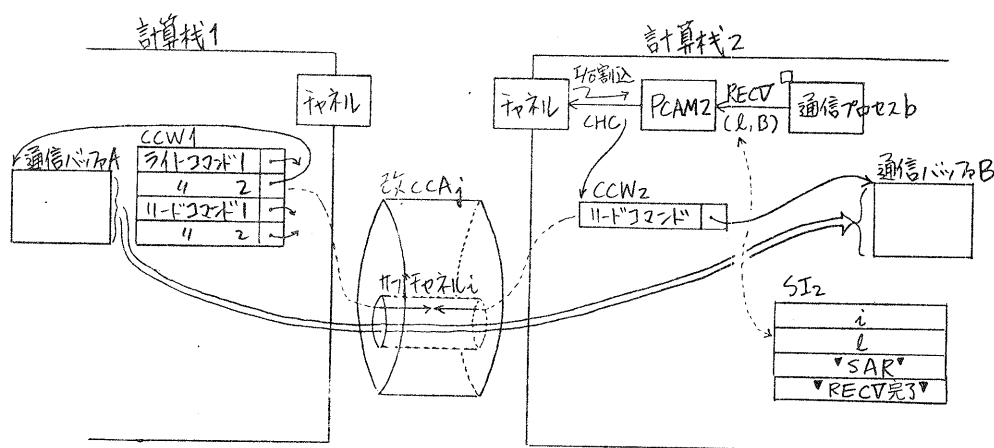


図7. フェーズ2の処理

【フェーズ3】(図8)

通信プロセスは依頼された処理を行った後、処理結果を通信プロセスAに返送する。フェーズ3は通信プロセスAが処理結果を受け取るまでの処理である。

(P3-1ステップ) 通信プロセスは依頼データを処理した後、処理結果を通信バッファBに格納し、論理バス名 l を指定してPCAM2にSARを要求する。PCAM2は論理バス名 l のSI2を捜し、処理段階が「RECV完了」であることを確認して、CCW3のライトコマンド1およびリードコマンド1のデータアドレスにCI2のアドレスを設定し、ライトコマンド2およびリードコマンド2のデータアドレスに通信バッファBのアドレスを設定して、サブチャネル i に入出力命令を発行する。

(P3-2ステップ) 改CCA i はCCW3のライトコマンド1, ライトコマンド2とCCW1のリードコマンド1, リードコマンド2のマッチングを各々確認し、CI2からCI1へ、通信バッファBから通信バッファAへデータを転送する。

(P3-3ステップ) 転送が終了すると、計算機1に入出力割込みが発生する。PCAM1はCI1の通信種別が「SAR」なので、サブチャネル情報域SI1にサブチャネル番号 i , 論理バス名 l , 通信種別「SAR', 処理段階「RECV完了」を設定する。こうして、論理バスとサブチャネルのマッピングは保持され続ける。PCAM1は通信プロセスAにSARの終了を通知する。

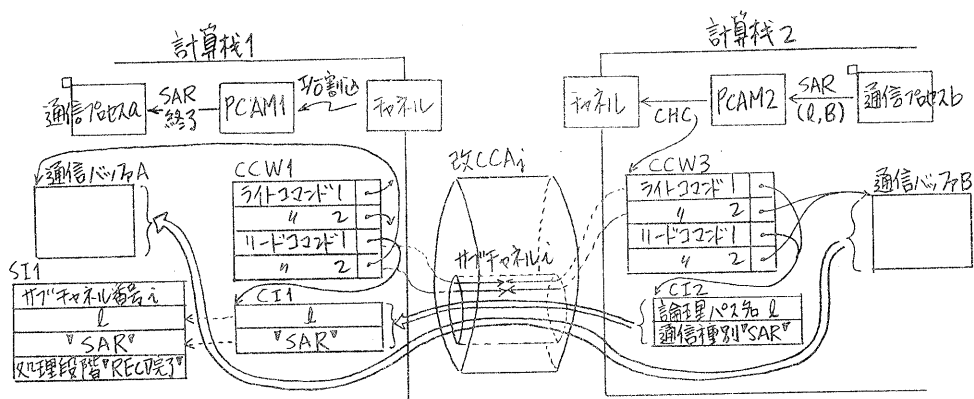


図8. フェーズ3の処理

[フェーズ4] (図9)

通信プロセスAは再度SARにて処理依頼の転送を要求する。これにより、通信プロセスBが発行したSARの受信処理が行われ、SARが終了する。フェーズ4は通信プロセスBのSARの終了までの処理である。その後、フェーズ3に戻ることで、通信プロセスA, Bはフェーズ3と4の処理をくり返す。

(P4-1ステップ) 通信プロセスAは2回目のSARをPCAM1に要求する。PCAM1は論理バス名がiのSIIを捜し、処理段階が「RECV完了」であることを確認して、CCW1の各コマンドのデータアドレスに照るべきアドレスを設定してサブチャネルiに入出力命令を発行する。

(P4-2ステップ) 改CCA_iはCCW1のライトコマンド1, ライトコマンド2とCCW3のリードコマンド1, リードコマンド2のマッチングを各々確認し、CI1からCI2へ、通信バッファAから通信バッファBへデータを転送する。

(P4-3ステップ) 転送が終了すると、計算機2に入出力割込みが発生する。PCAM2はCI2の通信種別が「SAR」を示しているのを、SII2にサブチャネル番号i, 論理バス名j, 通信種別「SAR」, 処理段階「RECV完了」を設定する。PCAM2は通信プロセスBにSARの終了を通知する。通信プロセスBは1回目の処理結果の転送が無事終了したことと2回目の処理依頼が到着したことを知る。

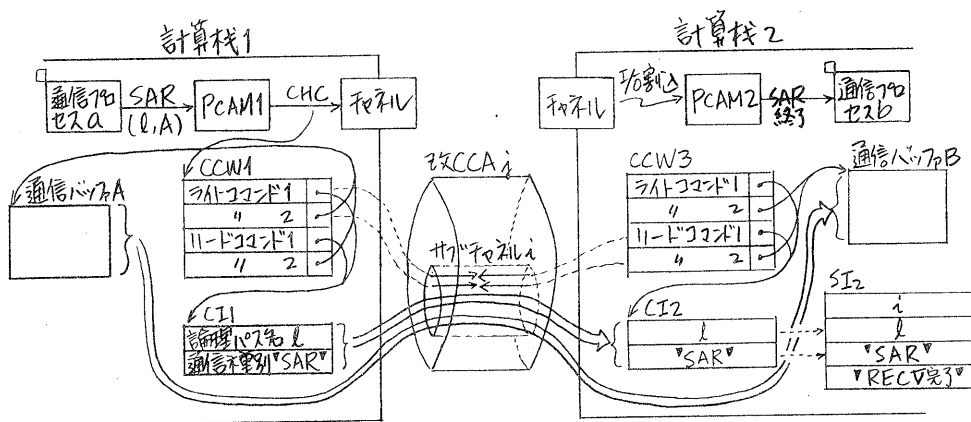


図9. フェーズ4の処理

5. 性能評価

5.1 比較方式の説明

PCAM方式の性能を評価するに当たって、改良前のCCAを用いた従来方式(ATT方式と仮称)を考える。ATT方式の特徴を以下に述べる。

(1)送信要求はアテンション割込みにて他方に通知される。従って、アテンション割込みの解析ルーチンが必要となる。

(2)送受信一括要求SARをPCAM方式のようにコマンドテイニングを用いて実現することはできない。従って、送信SEND、受信RECVのみのサポートとなる。

(3)送信、受信が独立した機能としてサポートされるので、欠送時の対応として送信、受信各

々にレスポンスが必要となる。

ATT方式の転送処理の流れを図10に示す。また、PCAM方式は双方SARを使用する形態を考える。PCAM方式の転送処理の流れを図11に示す。

5.2 高性能化の因果関係

PCAM方式がATT方式に比して著しい高性能化を達成するに至る因果関係を図12に示す。PCAM方式による効果は通信路の使用時間の短縮とダイナミックステップ数の削減として現われる。前者は通信路の転送力の向上に、後者はシステムスループットの向上に寄与する。通信路の転送力が向上すれば、通信路ボトルネックの範囲が小さくなる。

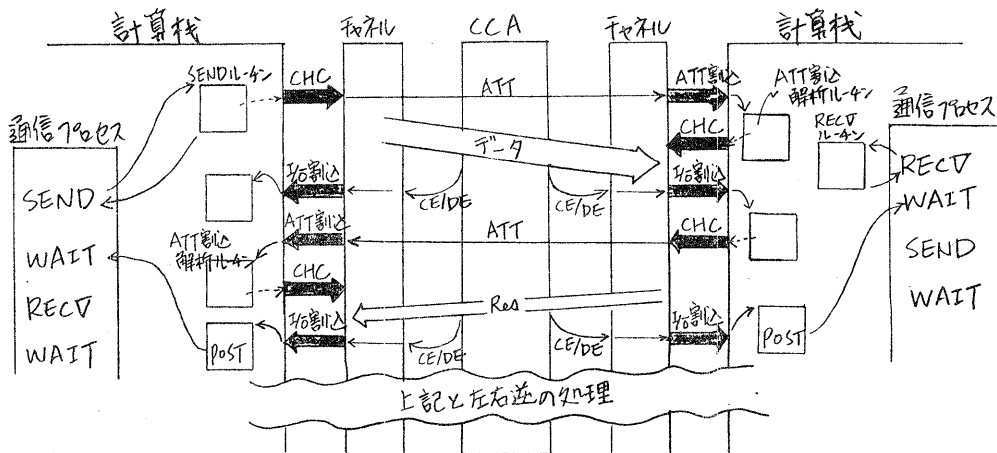


図10. ATT方式の処理の流れ

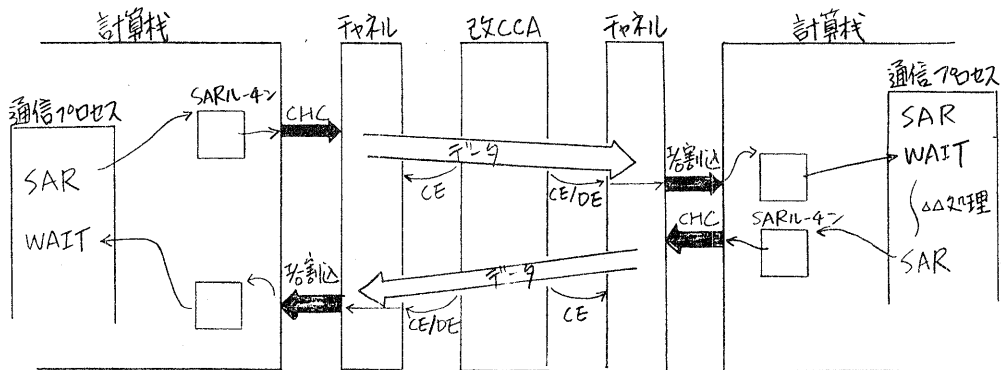


図11. PCAM方式の処理の流れ

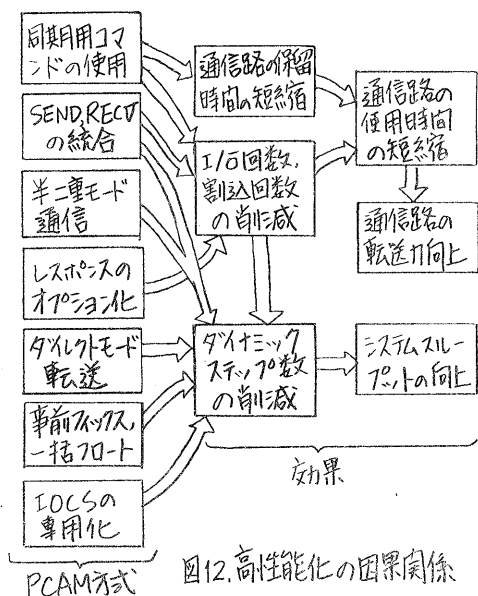


図12. 高性能化の因果関係

5.3 ダイナミックステップ数の内訳分析

PCAM方式とATT方式についてダイナミックステップ数を比較する。計測対象は送信と受信を対にした処理とする。PCAM方式では通信バッファはフィックスされているとし、またレスポンス不要のSARを用いるものとする。

ATT方式では図10に示す通り、アテンション割込解析処理が2回、入出力処理が4回走行する。アテンション割込解析処理は全体のダイナミックステップ数の約1/3を占め、残り2/3のうち、約1/8はSENDとRECVのマクロ処理であり、残りが4回の入出力処理となる。入出力処理は条件によりダイナミックステップ数は異なるが、無視できるものとする。

PCAM方式では図11に示す通り、入出力処理が1回走行するのみであり、アテンション割込解析処理は全く不要である。従って、他の条件がATT方式と同等とすると、PCAM方式のダイナミックステップ数はATT方式の11/48(22.9%)となる。しかし、インクリメントした実績値はATT方式の約1/20(5%)に達している。両者の比約1/5分のダイナミックステップ数の削減はダイレクトモード転送、バッファの事前FIX化、I/OCSの専用化、およびSENDとRECVの統合化によって実現されたものと考えることができる(図13)。

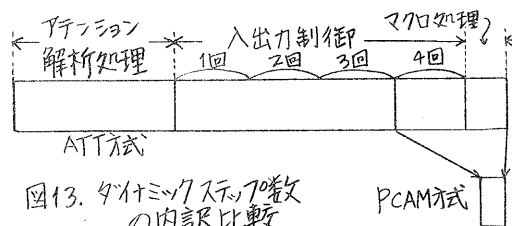


図13. ダイナミックステップ数の内訳比較

5.4 通信路の性能評価

通信路、即ちチャネル+結合バス+CCA(または改CCA)をモデル化し、その性能特性を明らかにする。

5.4.1 通信路モデル

PCAM方式、ATT方式における通信路のモデル化は既に提案している。^{[7][9]} しかし、ここではサグチャネルは転送終了後直ちに解放することを前提としており(即ち、サグチャネル数の制約が無いに等しい)、またレスポンスを考慮していない。通信路のデータ転送以外のコマンド処理の時間は無視してシングルサーバとしてモデル化した。ここではPCAM方式は双方SARを用いるため、サグチャネルは保持したままとなり、サグチャネル数の考慮が必要となる。また、レスポンスを考慮に入れ、通信路のデータ転送以外のコマンド処理の時間は同様に無視して、シングルサーバとしてモデル化する。なお、転送要求を呼と称す。

ATT方式の処理(図10)においては、呼毎に通信路に対しては2回の処理要求が発生する。また、サービス時間はアテンション割込み解析後、データ転送が完了するまで通信路が保留される事に着目する必要がある。1回目のサービス時間はデータ転送時間とアテンション割込み解析処理に要する時間の和となる。また、2回目のサービス時間はレスポンスの転送時間とアテンション割込み解析処理に要する時間の和である。

PCAM方式の処理(図11)においては、呼毎に通信路に対しては1回の処理要求となる。双方SARを用いるため、一連の処理のやりとり(例えば端末の電文単位)が終了するまで、サグチャネルを保持したままとなる。それ故、通信路の処理はサグチャネル数以上に多重に行われることはない。即ち、サグチャネル数に制限された系内呼数有限モデルとなる。通信路の

みを評価する場合には、シングルスサーバ故に、この制限は無いに等しいが、システム全体のモデル化を行う場合には、この制限を配慮する必要がある。また、サービス時間はデータ転送時間のみとなる。

両方式のモデルを図14に示す。また、両方式の入力率、サービス率は次式で与えられる。ただし、CPU性能は両計算機を同等とする。

λ_i : 入力率, μ_i : サービス率

[ATT方式]

$$\lambda_A = 2\lambda$$

$$1/\mu_A = (l + l_R)/2C + d/m$$

[PCAM方式]

$$\lambda_P = \lambda$$

$$1/\mu_P = l/C$$

ただし

λ : 呼の入力率

l : 転送データ長

d : 割込解析のDS

l_R : レスポンス長

m : CPU性能

C : 通信路の転送速度

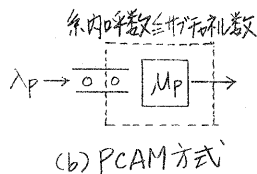
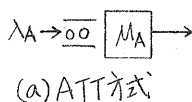


図14. 通信路モデル

5.4.2 解析結果

モデルを解析するに当って、呼のポアソン到着、指数サービス、 l_R/C は十分小さいという前提条件をおく。

評価項目として系内平均呼数 L 、系内の平均待ち時間 W および転送力比 γ を考える。 γ は

$$\gamma = \frac{\mu_P}{\mu_{A/2}} = \frac{2\lambda}{\mu_A / \lambda} = \rho_A / \rho_P$$

となり、両方式の利用率の比に当たる。また、

$$\gamma = 1 + 2(d/m)/(l/C)$$

系内平均呼数 L と系内平均待ち時間 W の代表的な解析結果を各々図15、図16に示す。また、 γ の変動の様相を図17に示す。

モデルによる解析の結果、次の事が明らかとなった。

(1) ATT方式の通信路の性能を決定する要因は l/C と d/m であり、PCAM方式では l/C である。

(2) l/C が相当大きく(10 msec以上)なれば、通信路の性能としては両方式に大差はなくなる。

(3) l/C が小さくなる程、また d/m が大きくなる程、PCAM方式はATT方式に比べ格段に通

信路の性能を向上させる。場合によっては、1桁以上の性能向上が実現できる。

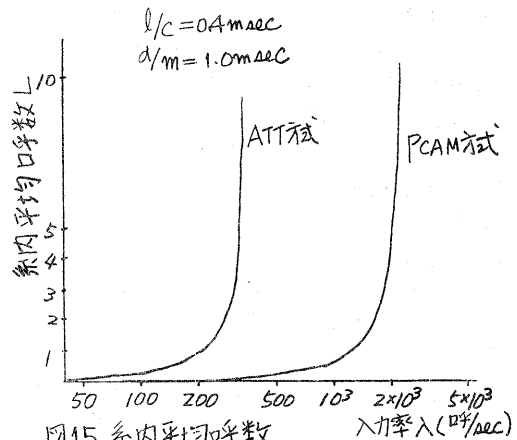


図15. 系内平均呼数

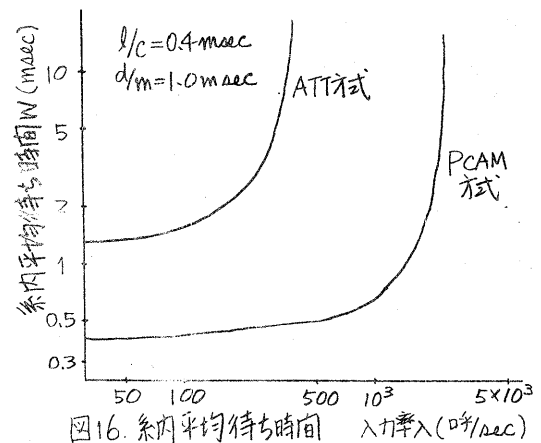


図16. 系内平均待ち時間

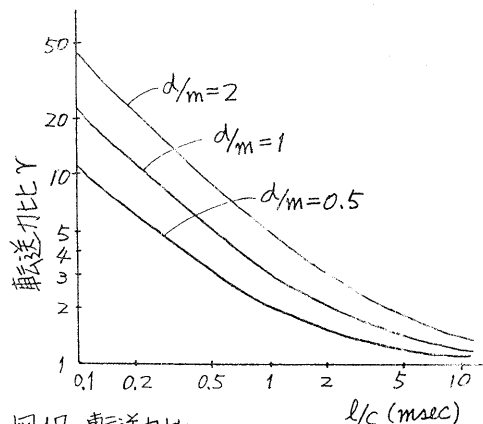


図17. 転送力比

6. あとがき

疎結合多重計算機システムは最近注目されて来ているが、そこでの計算機間通信用に高性能かつ高水準な通信法PCAMを開発した。

PCAM利用者は論理パスを始めとする通信基本要素を用いてファイル管理レベルの高水準な通信機能を利用することが出来る。PCAM機能の大きな特徴はダイレクトモード転送と送受信一括要求機能にある。ダイレクトモード転送により、送信側の通信プロセスが指定した通信バッファから受信側の通信プロセスが指定した通信バッファへ直接にデータ転送が行われる。

PCAMの高性能の機構は4章で詳細に説明した。PCAMはコマンドを保留する新機能を有する改良形CCAを使用しているが、改良前のCCAを用いた同種の通信法との比較の形で性能向上要因の因果関係と寄与度を分析した。PCAM方式のダイナミックステップ数は従来方式と比べて最高で約1/20に減少できた。また、通信路モデルを作成して、PCAM方式と従来方式における通信路性能を評価した。データ転送時間が小さい程、またアテンション割込解析時間が大きい程、PCAM方式は通信路性能を場合によっては1桁以上向上できることが明らかとなった。

今後は、PCAMのシステム全体の性能向上への貢献度の評価および改CCAの一層の改良に対する適用性の検討を行う予定である。

謝辞

日頃御指導戴くデータ応用研究室の高平敏室長ならびに開発に従事された馬場正和社員をはじめ同研究室、関連研究室の方々、また共同で開発に尽力された日本電気株式会社の水谷弘光氏、加藤善郎氏、大藤豊喜氏をはじめとする方々に厚く感謝します。

文献

- [1] Y. Abe, "A Japanese Online Banking System," *Dotamation*, Vol. 23, No. 9, pp. 89-97, 1977
- [2] M. Mori, T. Iida, "System Design for a Backend Processor," 8th Texas Conf. on Comp. System, 1979

[3] F. J. Maryanski, "Backend Database Systems," *Computing Surveys*, Vol. 12, No. 1, 1980

[4] 中野, 食田, "大規模バックエンドDBシステムの構築法," 昭和56年度信学会総全大会(投稿中)

[5] IBM System 370 Special Feature Description: Channel-to-Channel Adapter, GA22-6983-0, 1972

[6] 土反口 他, "チャネル経由の多重通信機能に関する一考察," 情処第19回全国大会, pp. 9-10, 1978

[7] 中野, 森, 水谷, "計算機間通信プロトコラムのプロセス分解法による設計について," 情処第21回全国大会, pp. 355-356, 1980

[8] 中野, "プロセス間通信における割込方式の評価," 昭和54年度信学会総全大会, No. 1409, 1979

[9] 中野, 森, 馬場, "バックエンドシステムにおける効率的通信法," 昭和55年度信学会総全大会, No. 1354, 1980