

マルチプロセッサ形パケット交換方式の プロセッサ間通信プロトコル

濱田 晃 金重 州起 西脇 峰雄
(日本電信電話公社 武蔵野電気通信研究所)

1. まえがき

DDXパケット交換サービスの需要の増加に対処するために、現在、経済性及び処理能力の向上を目的としたマルチプロセッサ構成の大容量パケット交換機の実用化を進めている。(1) 一般にマルチプロセッサシステムにおけるシステムの性能低下を招く大きな要因として、プロセッサ間通信オーバーヘッドがあり、このオーバーヘッドを低減し、できる限り効率良くプロセッサ間通信の処理を行うことが、システムの性能向上のために必要である。(2)

本稿では、マルチプロセッサ形パケット交換機におけるプロセッサ間通信プロトコルについて、通信オーバーヘッドの低減及び通信のための処理の高速化の点から検討を行う。また、一般にパケット交換ネットワークでは、各交換機でOSI (Open Systems Interconnection) 参照モデルにおけるlayer 3 (Network layer) までをサポートすることを前提としているが、プロセッサ間通信プロトコルのOSI参照モデルに対する位置づけについても検討を加える。

まず、2章で前提となるマルチプロセッサ形パケット交換機のシステム構成について述べ、3章で本システムにおけるプロセッサ間通信の特徴についてまとめる。4章では、この特徴を基にプロセッサ間通信プロトコルについて、通信オーバーヘッドの低減・通信のための処理の高速化の観点から検討を加え、その階層構成とOSI参照モデルとの対応及び各層の機能概要について述べている。更に通信形態についても分類し、その用途を示す。

2. システム構成

同一の交換機で、小規模から大規模まで広範囲の適用領域を持つパケット交換機を実現するためには、単一プロセッサ方式ではハードウェア上の制約が多く、マルチプロセッサ構成を採る必要がある。本稿で述べるマルチプロセッサ形パケット交換機は、プロセッサの結合方式として、交換リ

ンクと呼ばれる高速の共通バスによる結合方式を採用しており、交換リンクの能力次第で性能を向上することができる。

また、実時間性を特に要求されない保守運用機能を専用のプロセッサ (AMU: 保守運用プロセッサ) に機能分散させることにより経済化を図っている。

回線対応部にはX.25制御LSIを採用することにより、X.25レベル2の機能をLSIで実行し、パケット処理プロセッサ (PPU) の負担を軽減している。

マルチプロセッサ形パケット交換機のハードウェア構成を図1に示す。最大62台のパケット処理プロセッサ (PPU)、及び2台の保守運用プロセッサ (AMU) を交換リンクによって結合しており、信頼度設計からPPUはN+1冗長構成 (N=3) を、AMUは完全二重化構成を採用している。

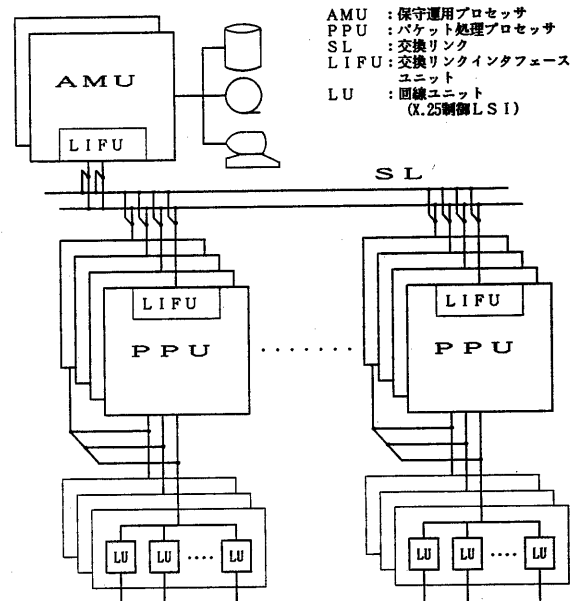


図1. システム構成

3. プロセッサ間通信の特徴

前章で述べたようなシステム構成のバケット交換機において、プロセッサ間通信は、以下の特徴を持つ。

- (1). 各PPUは、共通メモリを持たず、独立して動作することが可能であり、従来のシングルプロセッサ構成の小交換局に近似することができる。その意味で、プロセッサ間通信におけるデータの送受信は、交換局間のバケットの送受信と基本的に類似しているとみなすことができる。
- (2). 交換リンク上の転送データの大部分は、PPU-PPU間での加入者バケット(X.25バケット)と考えられる。従って、この転送を効率良く行うことが、システム全体のスループット向上に大きく寄与する。
- (3). 各プロセッサは、疎結合であるため、プロセッサ間通信は非同期通信が主体となるが、AMUとPPUに機能分散したプログラムの中には、互いに同期を取って通信をすることが必要なものもある。

4. プロセッサ間通信プロトコル

4.1 プロセッサ間通信のオーバーヘッド低減および高速化対策

マルチプロセッサシステムにおいて、プロセッサ間通信オーバーヘッドは、システムの性能低下を招く大きな要因となるため、これを極力低減し高速化を図ることが必要である。このオーバーヘッド低減対策として、本システムでは以下のような方策を採用した。

(1). 応答確認手順の簡略化

プロセッサ間通信における伝送制御手順を、ハードウェアの信頼性との兼ね合いで、どこまでサポートするか、またハードウェア/ソフトウェアの機能分担により、表1に示す4案を考える。

案1及び案2は、Data link layerの代表的プロトコルであるHDLC手順をサポートするものであり、データ伝送の信頼性は他の2案に比べ高く、またデータの順序逆転も防止可能である。案1では、HDLC手順をソフトウェアでサポートするため処理ステップ数も一番多く、交換リンクの高速化に比例してソフトウェアの処理量が増え、プロセッサ間通信のオーバーヘッドが非常に大きくなる。また、案2はソフトウェアのオーバーヘッドをなくすため、HDLC手順をX.25制御LSI等の利用によりハードウェアでサポートするものである。案2は案1に比べるとソフトウェアのオーバーヘッドはなく処理の高速化が図れるが、一方、ハードウェアの動作速度により交換リンクの転送速度の上限が決まってしまうため、プロセッサ間通信の高速化は困難である。

プロセッサ間通信は、システム内でのデータ転送であり、交換リンクの信頼性は通常の伝送路に比べて高いことを考えると、HDLC手順では複雑過ぎるため、もっと単純な伝送制御手順の採用による転送能力の高速化が望ましい。

案3、案4は共にACK/NAKによる簡単な伝送制御手順をサポートするものであるが、これをソフトウェアのみで行うのが案3であり、ハードウェア(ファームウェア)/ソフトウェアで機能分担するのが案4である。いずれもデータ転送の信頼性の点では案1、2に若干劣ると思われる

表1. プロセッサ間通信における伝送制御手順

比較項目	案 1	案 2	案 3	案 4
伝送制御手順とハードウェア/ソフトウェアの機能分担	・HDLC手順 ・ソフトウェアで全て行う	・HDLC手順 ・ハードウェアで行う (X.25制御LSIの利用)	・ACK/NAKによる応答確認 ・ソフトウェアで全て行う	・ACK/NAKによる応答確認 ・ハードウェア: 応答確認 ・ソフトウェア: 異常時の再送処理
処理ステップ数比	1	0.4 (ハードウェア動作速度により交換リンク転送速度が制限)	0.6~0.8 (案1/4の中間で手順の複雑さによる)	0.4
信頼性	案3/4に比べ信頼性が高い	同左 (但しハードウェアの信頼性による)	案1/2に比べるとやや低い	同左 (但しハードウェアの信頼性による)
データの順序逆転	順序制御を行うため逆転は起こらない	同左	逆転は起こるが上位で防止可能	同左

が、ハードウェアの信頼性が高く伝送誤りの起きる確率が小さいこと、またデータの順序逆転については、上位レベルで防止可能であることから、特に問題は無い。そこで、ソフトウェアでの処理量が少なくプロセッサ間通信のオーバーヘッドの少ない案4を採用する。

具体的には、ハードウェア（LIFU：交換リンクインタフェースユニット）によりデータ転送に使用したタイムスロットと同一のスロット内において、ACK/NAKによる送達確認を行い、また、ソフトウェアではNAK時の再送のみを行うことにし、通信に必要な処理の高速化を図る。

(2). プロセッサ間通信機能の単純化

プロセッサ間通信機能の豊富さにより上位レベルでの処理が軽減されるのは望ましいことであるが、一方プロセッサ間通信の処理を複雑にし、全体的なオーバーヘッドを増大させる要因となる可能性がある。そこで、プロセッサ間通信機能はできる限り単純化し、処理のオーバーヘッドを抑えることが望ましい。また、プロセッサ間通信は、PPU-PPU間のX.25パケットの転送が主体であり、この通信を効率良く行うことが重要である。

この観点から、プロセッサ間通信における通信相手プロセッサの選択機能を、プロセッサ間通信機能の一つとするか、それとも上位アプリケーション（パケット交換処理等）の機能にするかによってそれぞれ表2に示す案1、2が考えられる。

各プロセッサは疎結合であり、また通信回線が各プロセッサにくくり付けであるため、それぞれが小交換局に近似できる。従って、従来のシングルプロセッサのパケット交換局間で行うルーティング機能が、通信相手プロセッサの選択論理を伴うことになり、案2のように通信相手プロセッサの選択機能を、パケット交換処理プログラムの持つルーティング機能の中に含ませることが可能で

ある。これにより、通信のための処理をできるだけ単純化して、プロセッサ間通信のオーバーヘッドを抑えることができる。

これに対し、案1のように通信相手プロセッサの選択をプロセッサ間通信機能の一つとした場合、上位パケット交換処理のパケットの送出方路及び回線の選択機能から無理に分離することになり、冗長な処理となる。また、関連の深いルーティング機能の多くがプロセッサ間通信機能の中にも含まれることになり、全体としての処理オーバーヘッドが増大する。

(3). 同報通信機能の活用

複数のプロセッサに対して同一の情報を転送する同報通信機能の実現方法としては、図2に示すようにソフトウェアでサポートする案(a)と、ハードウェアによりサポートする案(b)が考えられるが、交換リンクが共通のバスであること、ソフトウェアでn回同一処理を行うのはオーバーヘッドが大きいことより案(b)を採る。1回のタイムスロットで複数のプロセッサにデータ送信が可能な、同報通信機能（グローバルアドレス転送機能）をハードウェアによりサポートし、これをうまく利用することによりプロセッサ間通信の高速化を図る。

具体的には、各プロセッサのファイルが同一である場合のシステムプログラムのIPL（初期プログラムローディング）や、幅狭規制指令の一斉通知等の用途に利用することができる。(3)

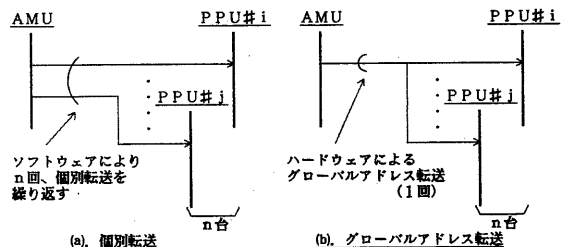


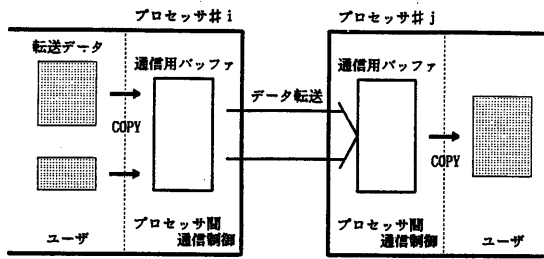
図2. 同報通信機能

表2. プロセッサ間通信機能の単純化

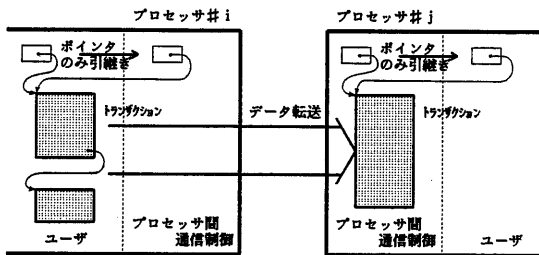
比較項目	案 1	案 2
機能概要	通信相手プロセッサの選択をプロセッサ間通信制御で行う。関連の深いルーティング機能の多くも含むことになる。	通信相手プロセッサの選択を上位パケット処理の持つルーティング機能に含ませる。
処理オーバーヘッド	一ヶ所で行える処理を無理に上位/下位レベルの二ヶ所に分けて行うこととなり、冗長な処理となる。	案1に比べ処理を一ヶ所でまとめて行うため、無駄な処理がなくなり、全体としてのオーバーヘッドは小さくなる。

(4). トランザクションを用いたダイレクト転送

プロセッサ間のデータ転送において、図3に示すように、転送するデータのあるユーザエリアとは別に専用のバッファを設けて、その間でデータを転送する間接転送方式と、転送データをトランザクション（ユーザエリア・通信用バッファ兼用）に入れたまま、そのポインタのみを引き継いで、データ転送をトランザクション間で直接行う直接転送方式の2つが考えられる。



(a). 間接転送方式



(b). 直接転送方式

図3. トランザクションを用いたダイレクト転送

間接転送方式は、ユーザエリアと通信用バッファが区別されているため、メモリエリアの管理が簡単で済む反面、ユーザエリアと通信用バッファとの間でデータのコピー処理が必要となり、これによるオーバーヘッドが大きい。これに比べて直接転送方式は、ポインタのみを持ち回るようにすれば良いため、トランザクションの管理は困難であるが、転送処理のダイナミックステップ数をかなり削減できる。

4.2 プロトコル構成

パケット交換機は、加入者通信パケットの交換処理が基本的な機能であり、この意味で図4に示すようにOSI参照モデルにおける階層構成のうちNetwork layer (layer 3) までを実現するものであるが、その他に交換機間での網制御パケットの通信、あるいは加入者と交換機との間のオンラインファシリティ登録関連パケットの通信等により、表3に示すような通信形態が存在する。また、AMUとPPUのプロセッサ間の機能分担をOSI参照モデルとの対応で現すと、図5のようになる。

ここでマルチプロセッサ形パケット交換機におけるプロセッサ間通信に着目して、プロセッサ間通信プロトコルの各階層と、OSI参照モデルと

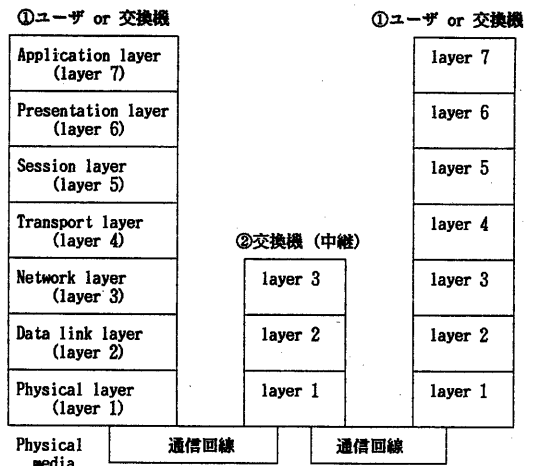


図4. OSI参照モデルとの対応

表3. パケット通信形態

通信形態	転送パケット種別	交換機の形態
ユーザ↔ユーザ	加入者パケット (X.25パケット)	②: 中継
交換機↔交換機	網制御パケット (PVC登録, 輻輳制御, 再開通知等)	①: エンドシステム
ユーザ↔交換機	オンラインファシリティ登録, 交換機情報の遠隔アクセス, 試験等のパケット	①: エンドシステム

OSI参照モデル

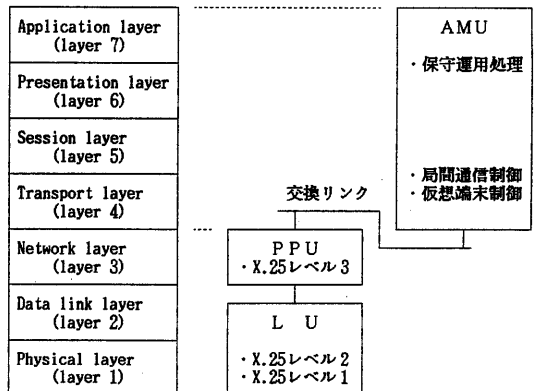


図5. プロセッサ間の機能分担とOSI参照モデルとの対応

の対応を考えると、図6~8に示す3つのケースがある。

図6において、ユーザ↔(PPU↔PPU)↔ユーザ間での通信の場合について考えてみると、上位より、Network layer に対してはパケット処理 (X.25 レベル 3) が対応し、Data link layer

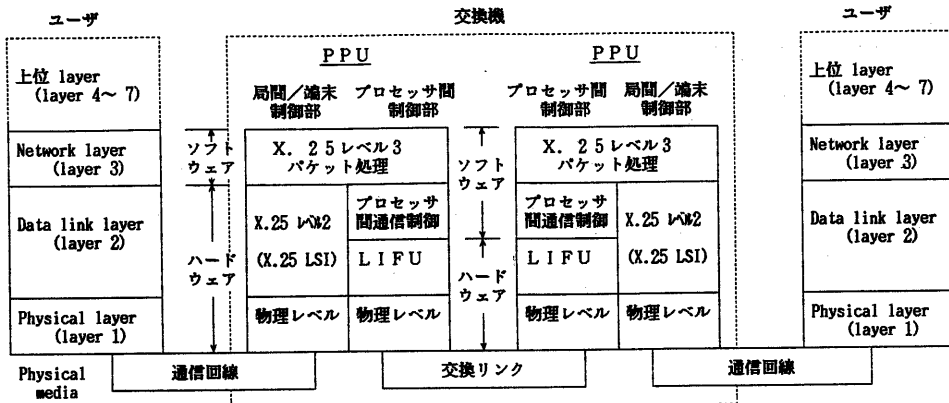


図6. プロセッサ間通信プロトコルの階層構成 (PPU-PPU間通信: 中継 or end-to-end)

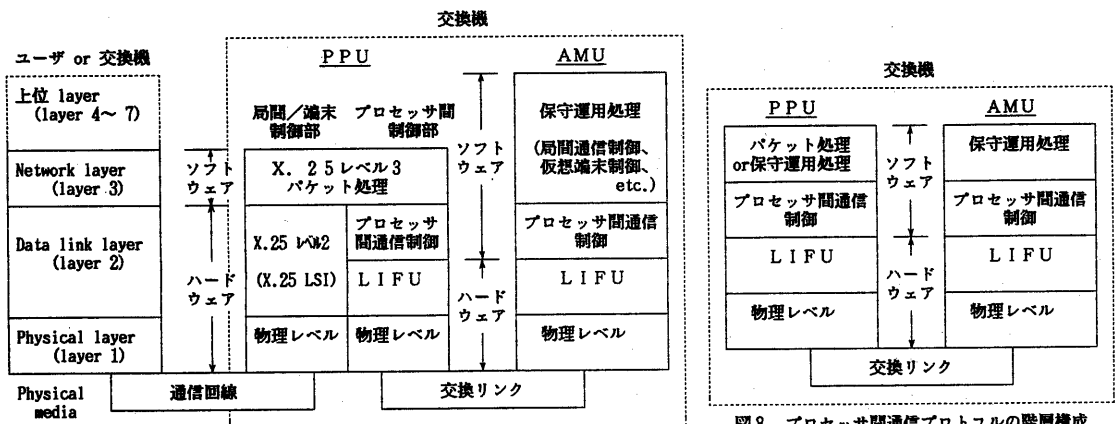


図7. プロセッサ間通信プロトコルの階層構成 (PPU-AMU間通信: 交換機間)

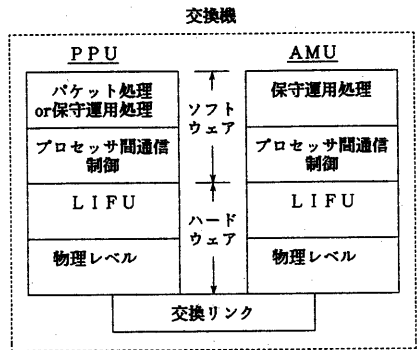


図8. プロセッサ間通信プロトコルの階層構成 (PPU-AMU間通信: 交換機内)

に対しては、回線側でHDL C制御 (X.25レベル2) が、交換リンク側でプロセッサ間通信制御及びLIFUが対応し、また Physical layer に対しては、回線側で回線ユニット (LU) が、交換リンク側でLIFUが対応する。さらに Physical media に対しては、回線側で通信回線が、交換リンク側で交換リンクが対応する。

前にも述べたように、回線側でのHDL C制御については、LU内部のX.25制御LSIによって実行されるため、ソフトウェアではLU (X.25制御LSI) とパケット処理とのインタフェースを取るだけで良い。従って、OSI参照モデルとの対応を考えると図6に示すように Data link layer の大部分はX.25制御LSIでサポートすることになり、Network layer をソフトウェアでサポートする構成となる。これに対し交換リンク側では、Data link layer の下位をLIFU

で、上位をプロセッサ間通信制御ソフトウェアでサポートする構成になる。

また図7、8では、Network layer より上位のレイヤとして、AMUにおける保守運用処理が存在する。

4.3 各層の機能概要

前節で述べたプロセッサ間通信プロトコルの階層構成において、各層が具体的にどのような機能を持つかについて以下に述べる。

(1). 物理レベル (Physical layer)

交換リンクを介したプロセッサ間でのデータ転送における物理/電氣的インタフェースを規定する layerであり、物理コネクションの活性化維持・非活性化、及びビット伝送のための機械的電氣的制御を行う。

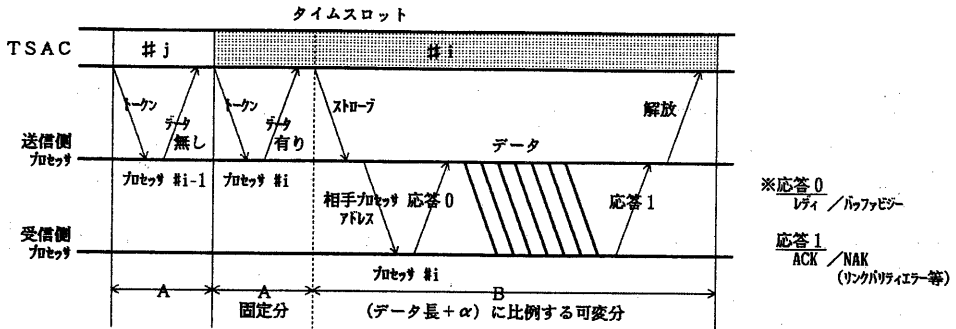


図9. 交換リンクの動作

(2). 交換リンク制御レベル

(Data link layer 下位)

交換リンク及びLIFUによる、プロセッサ間でのデータ転送のためのタイムスロット割り当て及びACK/NAK応答確認による伝送制御を行う。

交換リンクによるプロセッサの結合方式はトークン・バス（ポーリング）方式であり、交換リンクと呼ばれる共通バスがタイムスロット割り当て制御（TSAC）により、送出権（トークン）を各送出側プロセッサに対し順次割り当てていく。トークンを受け取ったプロセッサは送出するデータがあれば、割り当てられたタイムスロットを使用して、交換リンクに相手プロセッサアドレスとデータを送出し、受信側プロセッサではそのデータが自分宛のものならば受け入れる。トークンを受け取ったPPUで送出データが無い場合には、そのタイムスロットの使用を見送る（送出権の放棄）。この方式は、ハードウェア量が少なく、増設性に優れるという特徴を持っている。

交換リンクにおけるデータ転送シーケンスを図9に示す。以上の制御は交換リンク制御回路及びLIFUにより、ハードウェアで行われるため、ソフトウェアでは意識する必要はなく、プロセッサでの処理とは非同期に通信を実行することが可能である。これにより、プロセッサ間通信の処理に伴うオーバーヘッドを削減でき、プロセッサの負担を軽くすることができる。

また、交換リンクの信頼性を上げるために、ACK/NAKによる応答確認をハードウェアにより行う。

(3). プロセッサ間通信制御レベル

(Data link layer 上位)

同期通信に対する同期制御、同報通信処理、共

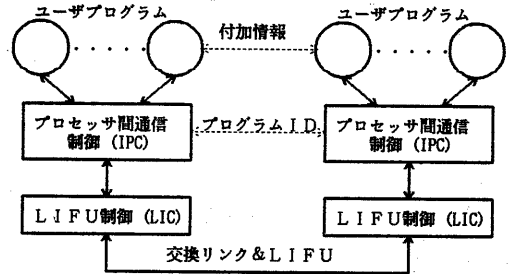


図10. 通信インタフェース構成

通的なタイミング処理、物理的なキュー管理等を行うレイヤであり、LIFUの送受信制御及び装置の状態管理等を行うハードウェア依存部であるLIFU制御（LIC）部と、その上位でプロセッサ間通信制御を行い、プロセッサ間通信の論理的インタフェースを提供するプロセッサ間通信制御（IPC）部に階層化している。

プロセッサ間通信における通信相手を識別するためには、通信相手プロセッサ番号（上位レイヤより指定）に加えて、相手プロセッサ内の通信先プログラムを識別するための情報が必要である。この情報として、プロセッサ間通信を行うプログラムと一対一対応に、プログラムID（識別子）を割り当て、これを送信側IPCでプロセッサ間通信ヘッダ情報として転送データに付加して送ることにより、受信側IPCで通信先プログラムを識別し、起動して通信データを渡す。（図10参照）

プロセッサ間通信の高速化・オーバーヘッド削減のため、通信相手プログラムの起動制御と、データの転送を同時に行う方法を採用した。

(4). アプリケーションレベル

(Network layer とその上位)

X.25パケットレベルプロトコルをサポートするレイヤで、パケット交換機の基本的機能である

パケット交換処理を行い、パケットのルーティング・フロー制御等の機能を実現する。

但し、プロセッサ間での機能分散により、AMUでは保守運用処理機能（系構成管理、保守コマンド処理、障害処理等）を実現するレイヤとなる。

4. 4 通信形態

マルチプロセッサ形パケット交換システムにおけるプロセッサ間通信の通信形態としては、大きく分けて、プロセッサ種別によりAMU-PPU間通信と、PPU-PPU間通信の2種類が考えられる。前者については、更に通信方向と個別/同報（グローバル）通信種別により3種類に分けることができる。また、これらの通信形態各々について、同期/非同期通信の両方が考えられる。

プロセッサ間通信における通信形態について、以上に述べた基準に従って分類したものを表4に示す。また、各々の通信形態に対する用途についても併せて示す。

転送されるデータの種別から大きく分類するとPPU⇒指定PPU非同期通信は加入者通信パケット（X.25パケット）の転送に用いられ、それ以外の通信形態はシステム制御データ、あるいは網制御用パケットの転送に用いられる。

5. あとがき

マルチプロセッサ形パケット交換機におけるプロセッサ間通信プロトコルについて、通信オーバーヘッドの低減・プロセッサ間通信の高速化及びOSI参照モデルとの対応、プロトコル構成の階層化の点から検討を行った。

通信オーバーヘッド低減・プロセッサ間通信の

高速化のために、以下の対策を採った。

- ・プロセッサ間通信における伝送制御手順としては、簡単なACK/NAKによる応答確認をハードウェアにより行うだけとする。
- ・通信相手プロセッサの選択機能は上位パケット処理の持つルーティング・フロー制御機能に含ませる。
- ・ハードウェアによる同報通信機能を設け、これを活用する。
- ・トランザクション間で直接データを転送する直接転送方式を採る。

これらの対策により、プロセッサ間通信プロトコルにおいて、Data link layer（OSI参照モデル layer2）に対応する部分が、ハードウェア/ソフトウェアの機能分担により二層（上位/下位）に分かれることになった。

また、プロセッサ間通信の通信形態としては、8種類の形態が存在し、そのうち6種類がシステム制御データの転送に用いられる。

以上、マルチプロセッサ形パケット交換機のプロセッサ間通信プロトコルについて、定性的な検討・評価を行ったが、今後は、プロセッサ間通信の性能、オーバーヘッド低減・高速化対策の効果及び信頼性等について、定量的に検討・評価していく予定である。

謝辞 本検討を進めるに当たり、御指導頂いた武蔵野電気通信研究所複合交換研究部パケット交換研究室吉田室長をはじめ、御討議頂いたパケット交換研究室及び複合交換装置研究室の関係各位に感謝します。

〔参考文献〕

- (1). 矢代他：“分散制御形パケット交換機の構成法”，信学会部全大，No.500，1983.
- (2). 濱田他：“マルチプロセッサ構成パケット交換機のプロセッサ間通信制御方式”，信学会部全大，No.501，1983.
- (3). 野口他：“マルチプロセッサ構成パケット交換機におけるプログラム転送方式の検討”，信学会部全大，No.503，1983.

表4. プロセッサ間通信における通信形態

通信種別	通信方向	同報	同期	用途
AMU- PPU間通信	AMU⇒PPU	個別	同期	保守コマンド実行要求
			非同期	PPU/PMX/回線制御
		同報	同期	加入者データ・局データの変更
			非同期	再開通知、輻輳通知
PPU⇒AMU	個別	同期	I/O機器入出力要求	
		非同期	障害通知、課金情報転送	
PPU- PPU間通信	PPU⇒PPU	個別	同期	—
			非同期	加入者パケットの転送