



Title	機能階層形多重通信プロセッサによる分散ネットワークの高性能化に関する研究
Author(s)	寺田, 松昭
Citation	大阪大学, 1992, 博士論文
Version Type	VoR
URL	https://doi.org/10.11501/3060227
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

機能階層形多重通信プロセッサによる
分散ネットワークの高性能化に関する研究

平成4年1月

寺 田 松 昭

機能階層形多重通信プロセッサによる 分散ネットワークの高性能化に関する研究

平成 4 年 1 月

寺田 松昭

内 容 梗 概

オフィスや工場において、L A N (Local Area Network) などの高速パケット伝送路で複数の計算機を相互に接続した分散ネットワークが広く用いられている。近年、分散ネットワークは大規模化が進み、計算機間でやりとりされるデータ量が増大して、計算機での通信制御負荷とデータの転送遅延時間が共に増大してきた。このため、計算機とネットワークとのインタフェース部分におけるデータ転送スループットの向上、データ転送遅延時間の短縮、計算機の通信制御負荷低減が強く求められている。

これらの要求を満たすためには、伝送路である L A N (産業用ではデータウェイと呼ぶこともある) の伝送速度を向上することと、計算機におけるデータ転送処理を高速化することとが必要である。L A N の伝送速度は、光ファイバ通信技術の適用などにより、1 ~ 2 M b p s から 1 0 ~ 1 0 0 M b p s へ 1 桁 ~ 2 桁高速化されてきている。一方、データ転送処理は、従来どおり計算機のソフトウェアによってパケット単位に行われており、高速化のあい路となっている。計算機とネットワークとの間のデータ転送処理を高速化するには、パケット単位の処理を計算機本体のソフトウェアで実行するのではなく、計算機と伝送路との間に高性能の F E P (Front End Processor) を置いて高速に実行する方式が考えられる。ところが、1 0 M b p s を超える高速パケット伝送路を対象とした F E P の報告は少なく、その効果の評価は十分でない。

本論文では、ホスト計算機と伝送路との間に、データ転送のための通信プロトコル処理を専用に行う機能階層形多重通信プロセッサを置くことにより、従来は、ホスト計算機で実行していた通信プロトコル処理の大部分を F E P で高速に実行させるような機能階層形多重通信プロセッサ方式を提案し、その性能評価を行っている。この方式により、通信プロトコル処理の高速化とホスト計算機の負荷削減を図ることができる事を示す。

具体的には、提案方式では、階層化された通信プロトコルのうち、第 1 層から第 4 層までの処理を、2 階層に分け、機能階層毎に高速ビットスライスマイクロプロッサ (語長 3 2 ビットのマイクロ命令を 2 0 0 n s で実行する) でパイプライン処理する専用の多重通信プロセッサを開発し、その効果を明らかにしている。トランスポートプロセ

ッサの構成方法としては、マルチプロセッサ・シングルコマンド処理方式とシングルプロセッサ・マルチコマンド処理方式の2つの方式を提案し、比較検討している。提案方式の評価を行うために、伝送路であるデータウェイの伝送速度が10Mbpsで、トランスポートプロセッサ2台、データリンクプロセッサ1台からなる機能階層形多重通信プロセッサを試作した。その結果、データ転送スループットが約3倍に、応答時間が約1/2に、計算機負荷がデータ長8バイトから10kバイトの範囲で、40%~80%削減でき、提案している機能階層形多重通信プロセッサ方式が通信プロトコル処理高速化の要請に十分応えうることを明らかにしている。

今後の高速伝送路時代に対応するためには、プロトコル処理の高速化が益々重要になる。そこで、本論文では、プロトコル処理をさらに高速化するため、正常処理をハードウェアで実行するようにした高性能プロトコルプロセッサ方式を提案している。提案方式は、今後のマルチベンダによるネットワーク環境を考慮し、標準プロトコルであるOSI (Open Systems Interconnection) に基づくプロトコル方式を実行できるようにしている。提案方式の効果は、実験的に評価し、従来方式にくらべ7~12倍高速化できることを明らかにしている。

本論文の構成は、次のとおりである。まず、第2章において、第1層と第2層のプロトコルを実行するデータリンクプロセッサと第3層と第4層のプロトコルを実行するトランスポートプロセッサとから構成される機能階層形多重通信プロセッサ方式によるFEPを提案する。また試作装置による性能向上効果について評価する。試作装置は、機能分散と負荷分散とにより性能を向上するために、トランスポートプロセッサを複数台のマイクロプロセッサで、データリンクプロセッサを1台のマイクロプロセッサで構成した点に特徴がある。第3章では、第2章で提案した方式を実用化した機能階層形多重通信プロセッサ方式について述べる。実用版では、ハード量を削減するために、トランスポートプロセッサを一台にすると共に、プロセッサの利用効率を向上するためにソフトウェアでコマンドを多重処理する方式にしている。第4章では、さらにプロトコルの高速処理を行うために、正常処理をハードウェアで実行する方式を提案し、その評価を行う。

関 連 発 表 論 文

- [1] M.Terada, J.Kashio, K.Yokota, Y.Hori, H.Fushimi: " A Network Operating System for High Speed Optical Fiber Loop Transmission System ", PP.641-646, Proc. of 5th International Conf. on Computer Communication (Atlanta), 1980
- [2] 寺田 松昭、関 高明、樫尾 次郎、堀 雄太郎、「高速パケット伝送路用前置処理装置の一構成法」、情報処理学会論文誌、V o l . 2 3、N o . 6、P P . 7 0 7 - 7 1 5、昭和 5 7 年 1 1 月
- [3] 寺田 松昭、高木 悟、安元 精一、中西 宏明、「機能階層形通信プロセッサによる計算機間メッセージ転送スループットの向上方式について」、情報処理学会「ローカルエリアネットワーク」シンポジウム、P P . 1 6 3 - 1 7 0、昭和 5 8 年 9 月
- [4] 福澤 淳二、寺田 松昭、高木 悟、溝河 貞生、末木 雅夫、「マルチ仮想回線動的チャネル割当方式による高性能 L A N アダプタの一方式」、情報処理学会「L A N / マルチメディアの応用と分散処理」シンポジウム、P P . 6 9 - 7 6、昭和 5 9 年 1 0 月
- [5] 寺田 松昭、高木 悟、樫尾 次郎、安元 精一、伏見 仁志、中西 宏明、「高速パケット伝送路用コマンド多重処理形前置処理装置」、情報処理学会論文誌、V o l . 2 6、N o . 2、P P . 2 1 9 - 2 2 7、昭和 6 0 年 3 月
- [6] 福澤 淳二、寺田 松昭、溝河 貞生、「高性能 M A C ブリッジ構成方式の提案と評価」、情報処理学会論文誌、V o l . 3 1、N o . 8、P P . 1 2 6 0 - 1 2 6 8、平成 2 年 8 月
- [7] O.Takada, M.Tsukakoshi, M.Terada, M.Yamaga, " An FDDI Bridge for the Super Backbone LAN ", Proc. of The 15th Conf. on Local Computer Networks (Minneapolis), PP.399-404, 1990
- [8] T.Hirata, S.Matsui, T.Yokoyama, M.Mizutani, M.Terada, " A High Speed Protocol Processor to Boost Gateway Performance " , Proc. of IEEE Global Telecommunications Conference (San Diego), PP.1426-1430, 1990

- [9] 松井 進、平田 哲彦、横山 達也、水谷 美加、
寺田 松昭、「専用ハードウェア化による通信プロトコル処
理高速化の一方式」、情報処理学会論文誌、V o l . 3 . 2 、
N o . 2 、P P . 2 7 2 - 2 7 9 、平成3年2月
- [10] 平田 哲彦、横山 達也、水谷 美加、寺田 松昭、
「高速LAN用通信制御装置の一構成法」、電子情報通信学会
情報ネットワーク研究会、I N 9 0 - 9 8 、P P . 1 3 -
1 8 、平成3年3月
- [11] M.Terada, T.Yokoyama, T.Hirata, S.Matsui," A High Speed
Protocol Processor to Execute OSI " , Proc. of IEEE
INFOCOM'91 (San Francisco), PP.944-949, 1991
- [12] 平田 哲彦、横山 達也、水谷 美加、寺田 松昭、
三巻 達夫、「プロトコル高速処理装置により高性能化をはか
ったLAN用通信制御装置の一構成法」、情報処理学会論文誌、
V o l . 3 3 、N o . 2 、平成4年2月（掲載決定）
- [13] M.Tsukakoshi, O. Takada, T. Murakami, M.Terada, M.
Yamaga, " Large-scale and High-speed Interconnection of
Multiple FDDIs using ATM-based Backbone LAN" , Proc. of
IEEE INFOCOM'92 (to appear)

機能階層形多重通信プロセッサによる分散ネットワークの高性能化に関する研究

目 次

第1章 序 論	1
1. 1 本研究の目的	1
1. 1. 1 分散ネットワークと課題	1
1. 1. 2 従来の研究	2
1. 1. 3 本研究の目的とアプローチ	3
1. 2 本論文の概要	4
第2章 機能階層形多重通信プロセッサ方式による高性能化 ..	6
2. 1 緒言	6
2. 2 データ転送処理高速化の課題	7
2. 2. 1 計算機制御システムの特徴	7
2. 2. 2 従来方式の問題点	9
2. 2. 3 高速化の方策	9
2. 3 機能階層形多重通信プロセッサ方式の提案	14
2. 3. 1 要求条件と解決方式	14
2. 3. 2 F E P - ステーション間インタフェース	17
2. 3. 3 F E P - 計算機間インタフェース	18
2. 3. 4 F E P の構成例	19
2. 4 実験的評価	23
2. 4. 1 評価の対象と目的	23
2. 4. 2 評価方法	23
2. 4. 3 結果	28
2. 5 結言	35
第3章 機能階層形多重通信プロセッサにおける コマンド多重処理方式	36
3. 1 緒言	36
3. 2 機能階層形多重通信プロセッサ方式実用化の課題 ..	36
3. 2. 1 F E P の位置付け	36
3. 2. 2 従来方式の問題点	37
3. 2. 3 改良形 F E P (F E P II)	37
3. 3 コマンド多重処理方式の開発	40
3. 3. 1 要求条件と F E P の方式提案	40
3. 3. 2 F E P II のソフトウェア処理方式	44

3. 3. 3	F E P II の性能評価	4 5
3. 3. 4	F E P II の方式評価	4 7
3. 4	適用評価	5 2
3. 4. 1	評価の対象と目的	5 2
3. 4. 2	評価方法	5 2
3. 4. 3	結果	5 8
3. 5	結言	5 8
第4章	プロトコル処理ハード化による高速化の提案	6 2
4. 1	緒言	6 2
4. 2	プロトコル処理高速化の課題	6 2
4. 2. 1	O S I プロトコル	6 2
4. 2. 2	プロトコル処理高速化の課題	6 2
4. 2. 3	高速化の方策	6 4
4. 3	プロトコル処理ハード化方式の提案	6 5
4. 3. 1	要求条件	6 5
4. 3. 2	構成方式の検討	6 6
4. 3. 3	プロトタイプの開発	6 8
4. 3. 4	L A N アダプタへの適用	6 8
4. 4	実験的評価	7 4
4. 4. 1	評価の目的	7 4
4. 4. 2	評価の方法	7 4
4. 4. 3	結果	7 4
4. 4. 4	L A N アダプタへの適用効果	7 4
4. 5	結言	7 7
第5章	結論	7 8

謝 辞

参 考 文 献

第 1 章 序 論

1. 1 本 研 究 の 目 的

1. 1. 1 分 散 ネットワークと課題

計算機により、鉄鋼プラントや化学プラントを制御するシステムは、計算機制御システムと呼ばれる。近年、計算機制御システムは、大規模化、トータルシステム化、分散化が進んでおり、ホスト計算機と多数の端末計算機を高速データウェイ[1]などの高速パケット通信ネットワークによって相互に接続した分散ネットワーク構成が広く採用されている[2][3]。計算機制御システムの大規模化に伴い、分散ネットワークには、①計算機間のデータ転送スループットの向上、②応答時間の短縮、③計算機負荷の削減など、性能の向上が強く求められている[4]。

また1970年代に開発が始まったLAN(Local Area Network)[5]は、その後急速に進展し、近年、LANを高速パケット通信ネットワークとして、サーバとなる計算機とクライアントとなる計算機/ワークステーション/パソコンを相互に接続し、クライアントからサーバの資源の共有を図るサーバ/クライアントシステムとよばれる新しい分散ネットワーク形態が急速に普及してきた。このような分散ネットワークにおいても、規模の増大や扱うデータ量の増大に伴い、①ファイル転送スループットの向上や②応答時間の短縮が強く求められている。

分散ネットワークの性能を向上するためには、①計算機間を接続するための伝送路であるLANやデータウェイの伝送速度を向上することと、②計算機におけるデータ転送処理を高速化することとが必要である。

データウェイの伝送速度は、光ファイバ通信技術の適用により、1～2Mbpsから10～32Mbpsへと約1桁高速化された[4][6]。一方、データ転送処理は、相変わらず計算機のソフトウェアによって、データウェイの転送単位であるパケット毎に行われている。このため、①計算機でのデータ転送処理時間が、データウェイのパケット転送時間に比べて大きく、伝送速度の向上に見合う計算機間での実効転送ス

ループットが得られない、②計算機のデータ転送処理に係る負荷が大きい、という問題があった。

1. 1. 2 従来の研究

上記課題に対して、従来下記の研究が行われてきた。

計算機におけるパケット単位の処理を高速化する方法として、外付けのハードウェアで実現する方法がある。ハードウェアで行う方式には、専用論理回路を組む方式[3]と専用通信プロセッサを計算機と伝送路との間に設ける方式[10][11][12][13]とがある。通信プロセッサの研究開発は、1970年代半ばころから活発となり、主に多種の通信手順を持つ多数の回線を制御する目的で研究開発が行われた。これらは、大型計算機のフロントエンドプロセッサとして位置付けられるものであり、回線速度も比較的低速（一般には、48kbp/s以下）で、サポートプロトコル範囲も2層以下（物理層とデータリンク層）に限られていた。

1980年代になると、LANが出現し、パソコンやワークステーション向けに、10Mbpsクラスの伝送速度を対象としたLANアダプタが開発されるようになった[7][8]。これは、一本の高速回線を対象としている点で従来の通信プロセッサとは異なるものであった。しかし、その機能範囲は2層以下（物理層とデータリンク層）であり、16ビットの汎用マイクロプロセッサでデータ転送処理を行うので、性能的にも制限されていた。研究開発の中心は、データリンク層の下位半分位置付けられ伝送路の共用制御を司るLANの媒体アクセス方式にあった。

制御用ネットワークのように、ミリ秒オーダーの応答時間が要求される分散システムにおいて、データウェイのような10～32Mbpsの高速パケット通信ネットワークを対象とし、1層から4層までのプロトコルを高速に実行する通信プロセッサの報告は少なく、その効果の評価は十分行われていない。

近年、FDDI（Fiber Distributed Data Interface）の標準化が進み計算機に直結できるLANの速度が100Mbpsに向上した[9]。しかし、LANの速度向上にもかかわらず、プロトコル処理オーバーヘッドなどによって、エンドツーエンドでの実効スループットが向上しないということが大きな問題になってきた。この原因は、プロトコ

ルの処理を計算機のプログラムやLANアダプタの中のマイクロプロセッサでパケット毎に処理していることによる。このため、エンドツーエンドの実効スループットを向上するための研究が活発になってきた。LAN環境でのトランスポートサービスのスループットが詳細に測定され報告されている[36][45]。高速ネットワークのための新しい軽量プロトコルが提案されている[37]。そのような軽量プロトコルアーキテクチャに基づく第3層と第4層のプロトコルを実行するVLSIの方式が提案されている[38]。このように、新しいプロトコルを導入して、実効スループットを向上するアプローチがある。しかしながら、ISOで標準化が進んでいるプロトコル体系であるOSI (Open Systems Interconnection: 開放型システム間相互接続) [40] は、マルチベンダによるネットワーク環境の増加に伴い、着実に普及することが予想でき、OSIのプロトコル処理を高速化することも益々重要になると考えられる。このアプローチの例としては、パケット交換網のための標準プロトコルであるX.25の第2層と第3層を実行するVLSIの試作をした例が報告されている[39]。またOSIのような多層構造プロトコルの性能評価をした例が報告されている[46][47][48]。

1. 1. 3 本研究の目的とアプローチ

本研究の目的は、計算機間での実効データ転送スループットを向上するための分散ネットワークの高性能化方式について論ずることにある。

計算機におけるデータ転送処理を高速化する方法としては、パケット単位の処理を計算機のファームウェアで行う方法と専用の外付けハードウェアで行う方法とが考えられる。両法を計算機負荷の観点から比較すると、後者のほうが性能の改善度合いは高い。そこで、本研究では、ホスト計算機と伝送路との間に、データ転送処理を専門に実行する機能階層形多重通信プロセッサを置くことにより、従来は、ホスト計算機で実行していた通信プロトコル処理の大部分を通信プロセッサで高速に実行させるようにした。この方式により、通信プロトコル処理の高速化とホスト計算機の負荷削減を図ることができる事を示す。

次に本研究では、LANアダプタに適用することを意図して、第4層以下のプロトコルの高速処理方式を検討する。高速ネットワーク対応の新プロトコルを考案して高速化を図る方法もあるが、本研究ではマルチベンダ環境でのネットワーク化を重視し、プロトコルとしてO

S I を対象にして高速処理方式を検討する。O S I の高速処理には、プロトコル処理アルゴリズムに工夫をするアプローチもあるが、それでは効果が少ないと考え、本研究ではプロトコル処理のハード化を検討する。

1. 2 本論文の概要

本論文は、機能階層形多重通信プロセッサによる分散ネットワークの高性能化について述べている。本論文の構成は、次のとおりである。

第2章では、通信プロセッサを第1層と第2層のプロトコルを実行するデータリンクプロセッサと第3層と第4層のプロトコルを実行するトランスポートプロセッサとから構成する機能階層形多重通信プロセッサの提案と性能向上度の評価を行う[16][20][21][22]。提案した方式は、ホスト計算機と伝送路との間に、データ転送のための通信プロトコルを専用に処理する機能階層形多重通信プロセッサを置くことにより、従来は、ホスト計算機で実行していた通信プロトコル処理の大部分を通信プロセッサで高速に実行させるようにした点に特徴がある。この方式により、通信プロトコル処理の高速化とホスト計算機の負荷削減を図ることができる事を示す。

具体的には、階層化された通信プロトコルのうち、第1層から第4層までの処理を、2階層に分け、機能階層毎に高速ビットスライスマイクロプロセッサ（語長32ビットのマイクロ命令を200nsで実行する）でパイプライン処理する専用の多重通信プロセッサを開発してその効果を明らかにしている。トランスポートプロセッサの構成方法としては、マルチプロセッサ・シングルコマンド処理方式（前置処理装置：F E P - I）を提案し検討する。効果の評価は、パケット通信ネットワークであるデータウェイの伝送速度が10Mbpsで、トランスポートプロセッサを2台のマイクロプロセッサで、データリンクプロセッサを1台のマイクロプロセッサで構成した機能階層形多重通信プロセッサを試作して実験的に行い、トランスポートプロセッサを用いない場合に比べてデータ転送スループットが3倍に、応答時間が1/2に、計算機負荷削減がデータ長8バイトから8kバイトの範囲で、40%～80%になることを明らかにする。この結果より、機能階層形多重通信プロセッサ方式が通信プロトコル処理高速化の要請に応えうることを結論づける。

第3章では、第2章で提案した方式（F E P - I）のハード量を削減するために、トランスポートプロセッサを一台にして、ソフトウェアでコマンドを多重処理する方式について述べる[23][24][25][26]。

具体的には、コマンド多重処理形トランスポートプロセッサ（前置処理装置：F E P - II）を提案する。提案するF E P - IIは、1台の高速マイクロコンピュータで多数のコマンドを多重に処理するので、少ないハードウェアで、高い性能を得ることが期待できる。提案方式の効果は、32ビットスーパーミニコンと高速データウェイとの接続に適用し、シミュレーションにより評価する。評価の結果、提案したF E P構成方式は、以下の効果があることを明らかにする。(1) ハード量は、基板2枚（第2章で提案したマルチプロセッサ方式の2/3）で済む、(2) 性能は、データウェイの伝送速度との比に換算して約0.14（第2章で提案したマルチプロセッサ方式のプロセッサ2台のときの性能に相当）である、(3) F E Pを適用した32ビットスーパーミニコンは、従来の16ビット機種に比べ、平均2000バイトのメッセージ転送性能において、スループットで約3倍、応答時間で約1/2の性能向上効果がある

第4章では、プロトコル処理ハード化による高速化方式の提案について述べる[27][28][29][30][31][32]。データウェイの伝送速度の向上は、めざましくやがて100Mbpsの時代も到来すると考えられる。本論文の主眼は、高性能通信プロセッサによる高性能化であるが、100Mbps時代には、単にプロセッサの高性能化だけでなく、プロトコル処理のハード化を考える必要がある。第4章では、プロトコル処理ハード化の課題を整理し、OSIのレイヤ2、レイヤ3、レイヤ4のプロトコルを対象に、プロトコル処理のハード化による高速化方式を提案する。

第5章では、本論文で得られた結果のまとめと今後の課題について述べる。

第2章 機能階層形多重通信プロセッサ方式による高性能化

2.1 緒言

近年、計算機制御システムの大規模化が進み、データウェイで接続された計算機間のデータ転送性能の向上が強く求められている[4]。性能を向上するためには、計算機におけるデータ転送処理を高速化することが必要である。従来、データ転送処理は計算機のソフトウェアによって、データウェイの転送単位であるパケット毎に行われている。このため、①計算機でのデータ転送処理時間が、データウェイのパケット転送時間に比べて大きく、伝送速度の向上に見合う計算機間での実効転送スループットが得られない、②計算機のデータ転送処理に係る負荷が大きい、という問題があった。

計算機におけるパケット単位の処理負荷をオフロードする方式として通信プロセッサの研究開発が、1970年代半ばころから、主に多種の通信手順を持つ多数の回線を制御する目的で行われた[10][11]。これらは、大型計算機のフロントエンドプロセッサとして位置付けられるものであり、回線速度も比較的低速（一般には、48kbps以下）で、サポートプロトコル範囲も2層以下に限られていた。1980年代になると、LAN[5]が出現し、パソコンやワークステーション向けに、10Mbpsクラスの伝送速度を対象としたLANアダプタが開発されるようになった[7][8]。LANアダプタの機能範囲は2層以下であり、16ビットの汎用マイクロプロセッサでデータ転送処理を行うので、性能に限界があった。

計算機制御システムにおける分散ネットワーク[14][15]のように、ミリ秒オーダーの応答時間が要求される環境において、10～32Mbpsの高速パケット通信ネットワークを対象とし、1層から4層までのプロトコルを高速に実行する通信プロセッサの報告は少なく、その効果の評価は十分行われていない。

本章では、計算機と伝送路との間に、トランスポートプロセッサ（以下前置処理装置：FEPと呼ぶ）とデータリンクプロセッサ（データウェイの場合は、ステーションに相当）とから構成する機能階層形多重通信プロセッサを置くことによってデータ転送性能を向上する方式の提案と性能向上度の実験的評価を行っている。

- (1) 高いデータ転送スループットが、ミリ秒オーダーの厳しい応答時間の制約下で得られるように、FEPを、複数の高速マイクロコンピュータ（I

OP)と1台のマルチプレクサとで構成している。

- (2) IOPのメモリを少なくするために、IOPには、1パケット分のバッファだけをもたせ、計算機とIOPとの間の転送単位は、パケットにしている。
- (3) 計算機の負荷を低減するため、メッセージのセグメンティング／リアセンブリは、IOPが計算機のメモリに置かれたメッセージを直接アクセスすることによって行う。
- (4) 上記提案方式の性能向上度を、IOP2台から成る試作装置によって実験的に評価し、FEPを挿入することによって、ステーションのみの場合に比し、データ転送スループットが3倍に、応答時間が1/2に、計算機負荷削減がデータ長8バイトから10kバイトの範囲で40%～80%になることを明らかにしている。

2. 2 データ転送処理高速化の課題

計算機制御システムにおける計算機間データ転送の性能を定義するとともに高速化に当たっての問題点と解決法を論じる。

2. 2. 1 計算機制御システムの特徴

計算機制御システムの中で、図2.1に示すようなデータウェイに代表される高速パケット通信ネットワークによって、複数の計算機が接続されている例[17][18][19]をとりあげ、以下の議論を進める。データウェイは、光ファイバ／同軸ケーブルとステーションとで構成されている[4]。ステーション相互間では、パケットと呼ばれる最大500バイト程度の可変長データブロックが、10～32Mbpsでビットシリアルに伝送される。計算機間のデータ転送は、主にホスト計算機と10～20台の端末計算機との間で行われる。近年、計算機制御システムは、大規模化が進み、下記性能の改善が強く求められている。

- (1) データ転送スループット
- (2) 応答時間
- (3) 計算機負荷

これらの述語は、次のように定義する。

- (1) 単位時間に転送できるデータ量
- (2) メッセージの送信要求がユーザ・プログラムから発行されてから、送信を完了し、ユーザ・プログラムに制御が戻るまでの時間
- (3) データ転送処理プログラムが動作している時間と計算機の動作時間との比

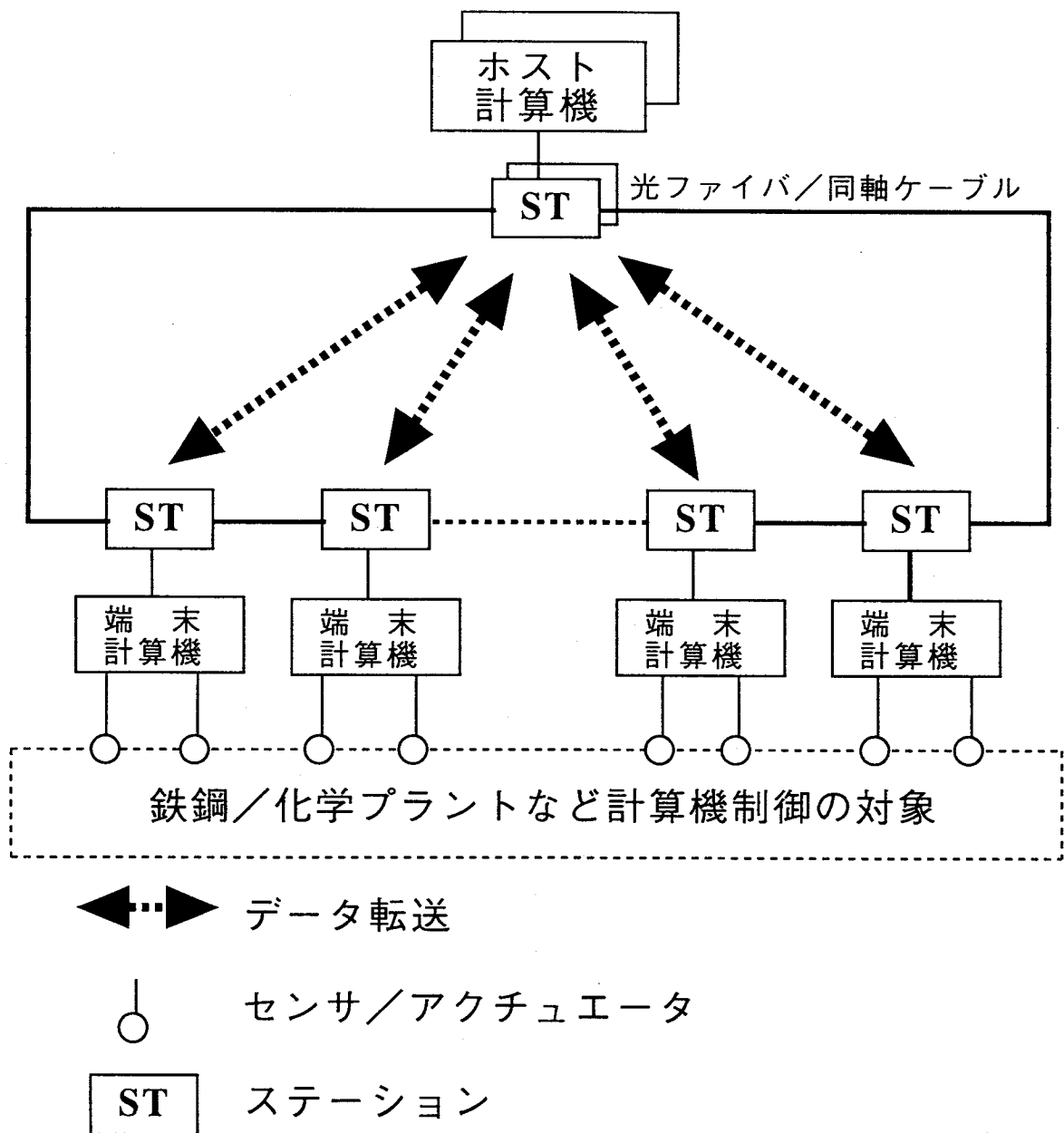


図 2.1 高速パケット通信ネットワークを用いた
計算機制御システム

2. 2. 2 従来方式の問題点

計算機間で転送される情報内容はメッセージといわれている。メッセージの長さは不定のため、計算機間のデータ転送では、メッセージをセグメントに分割し、各セグメントをパケットとして伝送路に送出する。

従来は、計算機とデータウェイのステーションとが、図 2. 2 のように直接接続され、データ転送処理は、計算機のソフトウェアによって、パケット単位に行われていた（図 2. 3）。すなわち、データ転送処理は、次の四つの制御により実施されていた。

- (A) ユーザプログラムとのインタフェース制御
- (B) メッセージのセグメンティング／リアセンブリ
- (C) 伝送路に時分割多重化されたパケットを複数の論理的な伝送路（論理チャネル）に分ける制御
- (D) ステーションとのインタフェース制御

このように、多くの処理をすべて計算機のソフトウェアで行っているため、計算機間のデータ転送性能は、伝送路の伝送速度が向上したにもかかわらずほとんど向上していない。

2. 2. 3 高速化の方策

データ転送処理を高速化する方法としては、下記の 2 つが考えられる。

- (1) ファームウェア化
- (2) ハードウェア化

ファームウェア化は、計算機のデータ転送処理すべて（2. 2. 2 節の A ～ D）にわたって高速化を行うことができるが、計算機負荷の低減効果が少ない。

ハードウェア化は、パケット単位の処理（2. 2. 2 節 B ～ D）を、外づけのハードウェアで行う方法で、専用の論理回路を組む方式と前置処理装置（FEP）を設ける方式とがある。専用回路は、高速性は、大であるが、論理チャネルの数が多くとれないなど融通性に問題がある。これらに比し、FEP を設ける方式は、計算機負荷低減効果が大きく、融通性に富む。

本論文では、FEP 方式を取り上げる。FEP の導入（図 2. 4）によって、計算機は、図 2. 5 に示すようにパケット単位の処理から解放され、メッセージ単位の処理のみ行えばよく、計算機負荷の大幅な低減が期待できる。さらに、

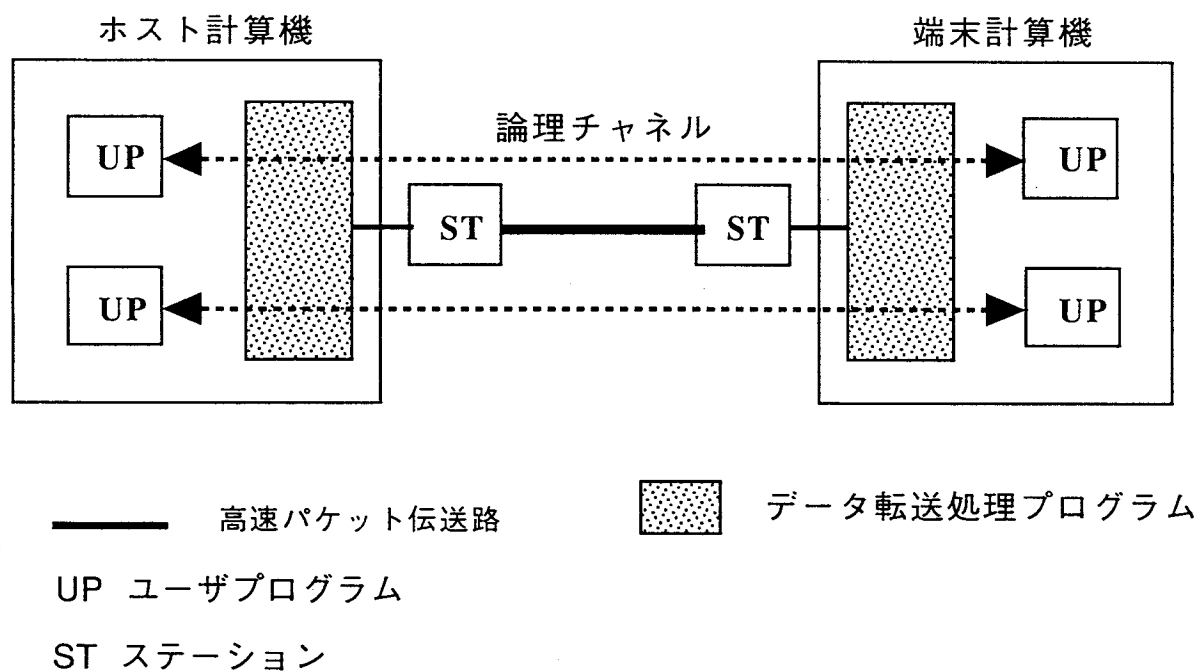
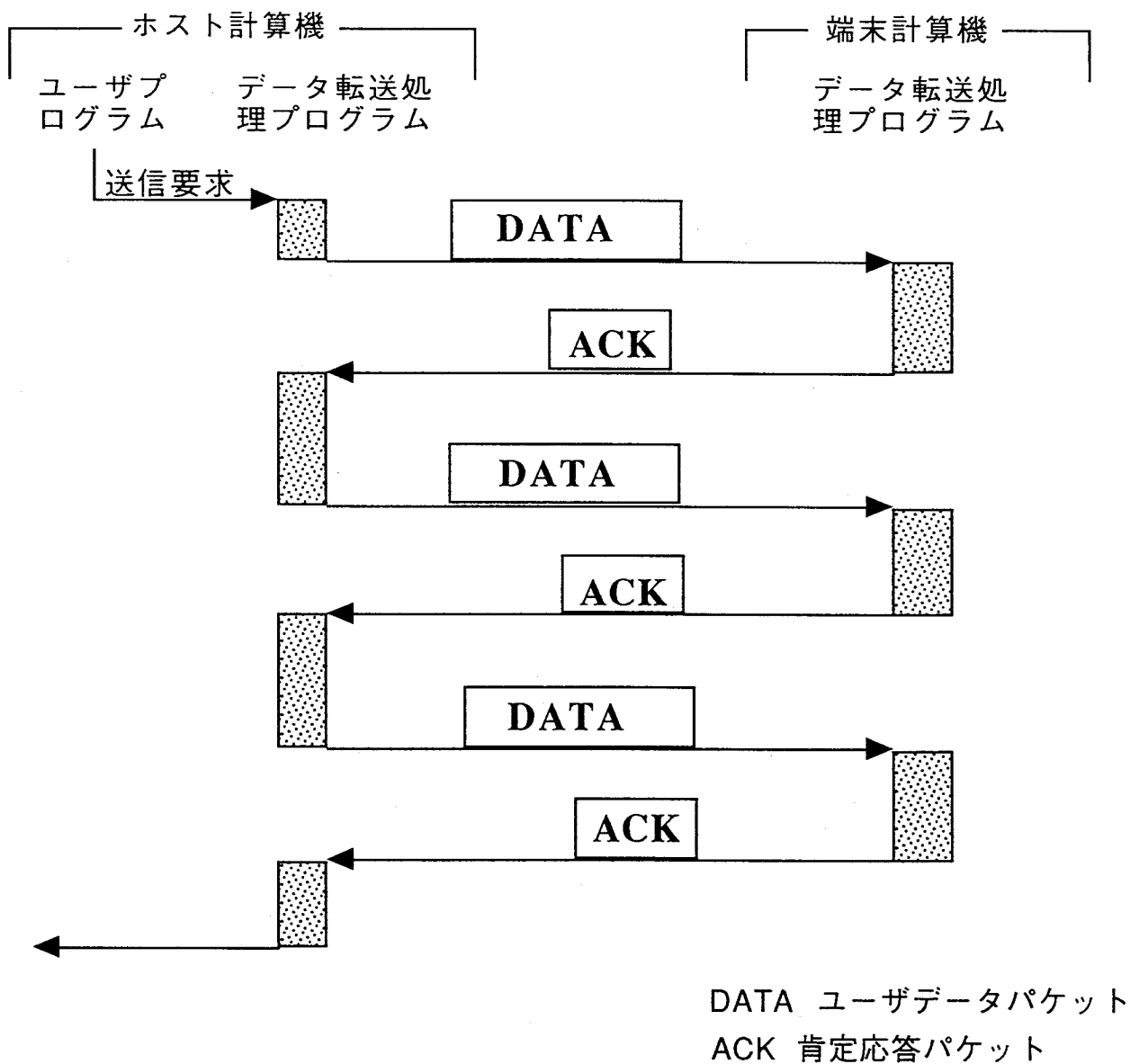
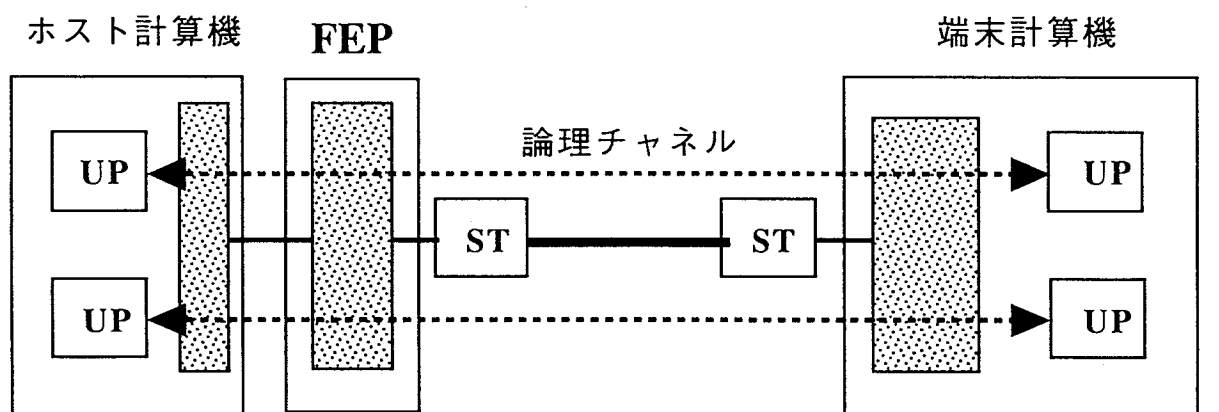


図 2.2 高速パケット伝送路と論理チャネル



論理チャネルレベルのデータ転送モデルを示している

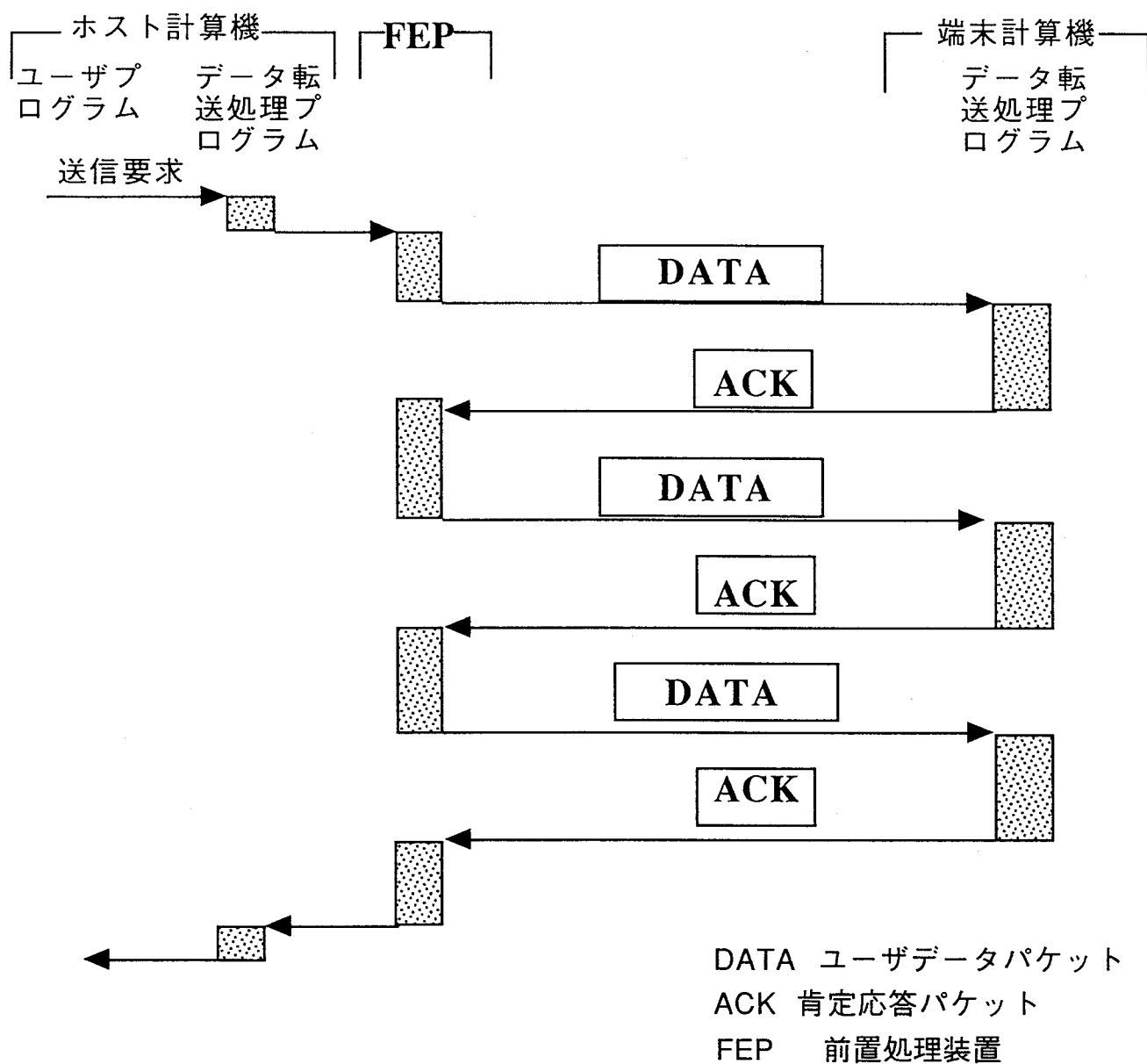
図 2.3 従来システムのデータ転送モデル



——— 高速パケット伝送路
 FEP 前置処理装置
 UP ユーザプログラム
 ST ステーション

 データ転送処理プログラム

図 2.4 前置処理装置 (FEP) を用いた計算機間接続



論理チャネルレベルのデータ転送モデルを示している

図 2.5 FEP適用時のデータ転送モデル

F E Pを高性能なプロセッサで構成すれば、F E Pで行うパケット毎の処理を、計算機で行っていたよりも高速に行えるので、データ転送スループット、応答時間の改善も期待できる。

2. 3 機能階層形多重通信プロセッサ方式の提案

高速パケット通信ネットワークに適した前置処理装置（F E P）が具備すべき条件を明らかにし、その解決方法について論ずる。

2. 3. 1 要求条件と解決方式

(1) 要求条件

高速パケット通信ネットワークを用いた計算機制御システムにおいて使用されるF E Pには、下記の条件が必要である。

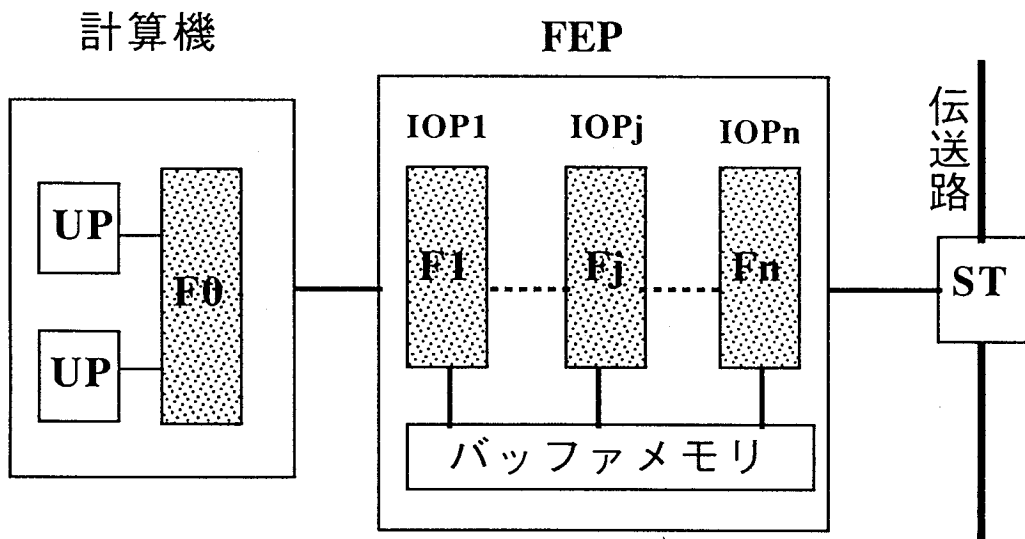
- (a) 10～32Mbpsの伝送速度に適応できる高いデータ転送スループット
- (b) 応答時間が数ミリ秒という高速応答性
- (c) パケットが多重化された高速伝送路とのインタフェース機能
- (d) メッセージのセグメンティング／リアセンブリおよび伝送路のパケットを論理チャネルに分ける処理をF E Pが行えるようにする計算機インタフェース

(2) データ転送スループットの向上

データ転送スループットは、平均パケット長と、パケット処理能力の積で定義され、パケット処理能力は、パケット当たりの処理量（プログラムのダイナミックステップ数）が一定ならば、F E Pの命令実行能力で定まる。したがって、10～32Mbpsの高速パケット伝送を可能にするには、F E Pに十分な処理能力を持たせる必要がある。そこで、提案方式では、F E Pを複数のプロセッサ（以下I O P:Input/Output Processorと称す）で構成することにした。複数I O PによるF E Pの構成方式としては、機能分散方式（図2. 6）と負荷分散方式（図2. 7）とが考えられる。

(a) 機能分散方式

各I O Pは、固有の機能を分担する。計算機からのデータは、まずI O P 1に渡され、以降I O P 2, I O P 3, …と渡され、最後にステーションに渡される。



 データ転送処理プログラム

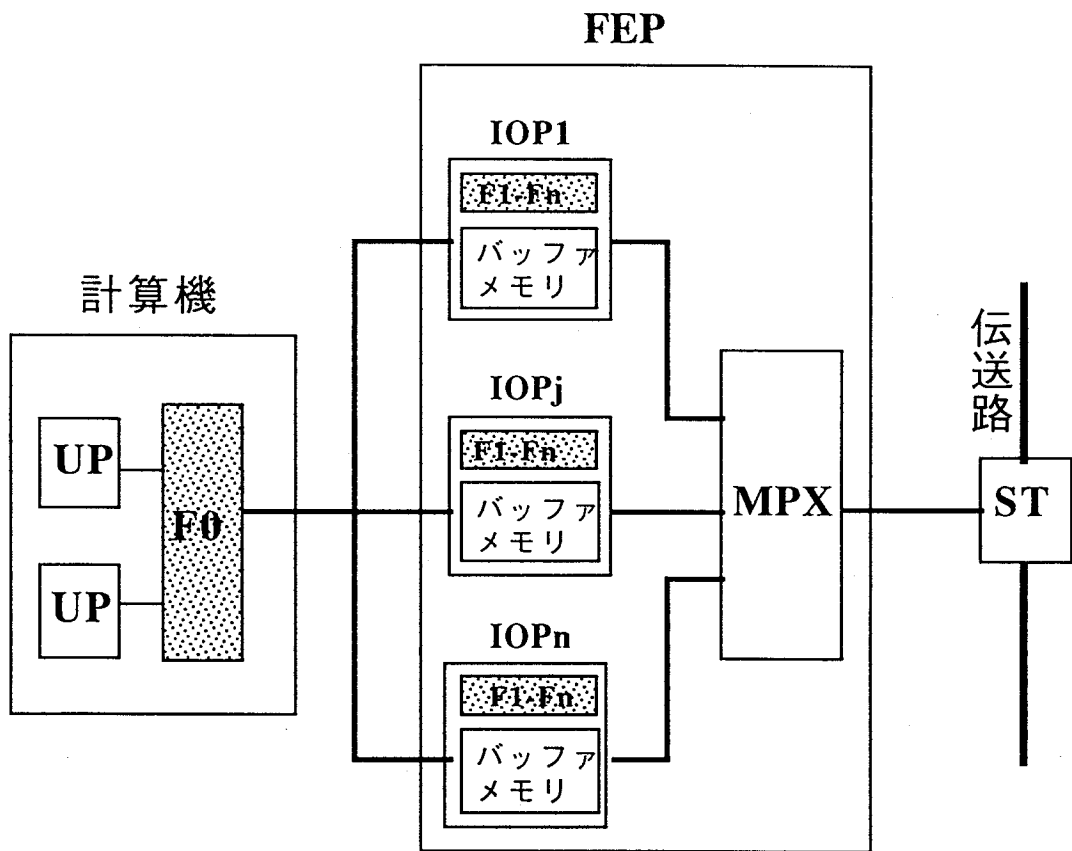
UP ユーザプログラム

ST ステーション

FEP 前置処理装置

IOP 入出力プロセッサ

図 2.6 機能分散方式によるFEPの構成例



UP ユーザプログラム

ST ステーション

FEP 前置処理装置

IOP 入出力プロセッサ

MPX STとのインタフェース制御
とパケットの分配／統合を実行

 データ転送処理プログラム

図 2.7 負荷分散方式によるFEPの構成例

(b) 負荷分散方式

各 I O P は、同一の機能を備えている。計算機からのデータは、各 I O P に分配され、各 I O P では、これらのデータを一時、I O P のバッファメモリに蓄えて、所定の処理を行った後、ステーションへ渡す。このとき、各 I O P を対等にするため、マルチプレクサ：M P X と呼ぶ装置を介してステーションに接続する。

両方式を比較すると、応答時間特性は、M / M / S 待ち行列モデルを仮定する限り、負荷分散方式が優れていることがよく知られている。一方、プログラムと制御テーブルを格納するためのメモリは、機能分散方式が少なく済む。計算機制御システムでは、応答時間に対する要求が厳しいので、負荷分散方式を採用し、I O P のメモリ容量を削減する工夫を行うことにした。

(3) 応答時間の短縮

応答時間は、F E P の処理時間とデータを I O P のバッファメモリに転送する時間とを短縮することによって改善される。一つの packets に関する処理は、並列に行えないので、F E P の処理時間を短縮するには、I O P 単体の命令実行速度を大きくする（すなわち、高速プロセッサを使う）必要がある。バッファメモリへの転送時間を少なくするには、転送回数を減らすこと、バッファメモリの読みだし／書き込みを高速にする（すなわち高速メモリを使う）が必要になる。

高速プロセッサと高速メモリは、実装密度が低く、高価なので、少なくともバッファメモリは、極力少なく済む方式が望ましい。応答時間を短縮し、メモリを減らすため、次の方針を立てた。

(a) F E P 内でのデータ蓄積は、I O P のみにとどめ、M P X では行わない。

(b) I O P のバッファメモリは、1 packets 分のみを準備する。

(c) 計算機のメモリを各 I O P の共有メモリとして用いる。

2. 3. 2 F E P -ステーション間インターフェース

高速 packets 通信ネットワークの特徴は、同一伝送路上を、異なる論理チャネルの packets が、時分割多重で高速に伝送されていることである。これら高速伝送されている packets を複数の I O P で高速に処理し、データ転送スループット、応答時間を改善するには、F E P とステーションおよび F E P と計算機のインターフェースの実現方法が重要な課題である。

F E P とステーションとは、マルチプレクサ（M P X）によって接続され、

受信パケットは、MPXによってIOPに分配される。MPXには、IOP複数台分以上の高いスループットとIOPの負荷に片寄りを生じさせないパケット分配方式とが求められる。

MPXは、上記目的を満たすため、パケットヘッダの参照、論理チャネルへの振り分けをすることなく、パケットをIOPに分配する。すなわち、MPXは、各IOPのバッファメモリにパケットが蓄積されている（使用中）か否かを保持しておき、バッファメモリが使用中でないIOPにパケットを渡す。バッファメモリの使用状況は、IOPからMPXに対して、専用の信号線によって知らされる。

この方式によれば、MPXでのパケットの蓄積は不要であるばかりでなく、分配アルゴリズムが単純なので、分配処理をハードウェア化しやすく、高速化できる。

2. 3. 3 FEP－計算機間インターフェース

データ転送性能を向上するためには、FEPと計算機とのインターフェースを単純にするとともに、計算機の処理をできるだけFFPに移すことが重要となる。とくに、高速パケット伝送路用FEPでは、メッセージのセグメンティング／リアセンブリとパケットを論理チャネルに振り分ける処理が、データ転送処理プログラムの大きな部分を占めるので、これらの処理をFEPでどのように行うかが課題となる。とりわけ、複数のマイクロコンピュータでの処理方法が中心課題となる。

計算機とFEPとのインターフェースの動作は、大きく送信と受信とに分けられ、各々は、計算機からIOPへの起動とIOPから計算機への終了報告とからなる。

送信時には、計算機のデータ転送処理プログラムが、使用中でないIOPを1台選択してメッセージの送信を要求し、このIOPを使用中の状態にする（起動）。起動されたIOPは、計算機メモリのメッセージを読みだして、最初のセグメントを送信し、送信が終わると、計算機にこのIOPが空状態になったことを知らせる。メッセージの残りの部分は、セグメント毎に受信側から返送されてくる応答を受信IOPが、計算機メモリのメッセージを読みだして送る。最後のセグメントを送信したIOPが、計算機に対して送信完了の割り込みを入れる（終了報告）。

受信時には、受信要求を計算機のテーブル（該当論理チャネル制御テーブル）にセットしておく（IOPは起動しない）。やがて、伝送路からこの論理チャネルあてのパケットがMPXに届く、MPXは、1台のIOPにパケットを分

配する。パケットを受け取った I O P は、パケット・ヘッダに定義されている論理チャネル制御テーブル（計算機のメモリにある）より、受信パケットのストア・アドレスを取だし、そこへパケットをストアする（パケットのリアセンブリ）。最終セグメントのパケットをストアした I O P が計算機に対して受信完了割り込みを入れる（終了報告）。

このような I O P の連携動作を可能にするためには、論理チャネル制御テーブルを計算機のメモリを介して、ある I O P から別の I O P へリレーできるようにする必要がある。すなわち、マスタ計算機のメモリに置き、セグメンティング／リアセンブリを行う I O P がそれを自分のバッファメモリに一時的にコピーして使い、必要な部分を変更して再び計算機のメモリに返すのである。これを制御テーブルリレー方式と呼ぶ。以下では、この方式を図 2. 8 にもとづき、詳しく説明する。

- (a) I O P 1 は、パケット i をメッセージ・バッファにストアし終わると A D D R (i) を A D D R (j) にして、制御テーブル (x) を計算機の制御テーブル (y) に返却する。
- (b) 次に、パケット j を I O P 2 が受けると、テーブル (y) を I O P 2 の制御テーブル (z) にコピーして、パケット j のストア・アドレスである A D D R (j) を知る。
- (c) パケット j をメッセージ・バッファの A D D R (j) にストアすると A D D R (j) を A D D R (k) にしてテーブル (z) をテーブル (y) に返却する。

この方式によれば、メッセージおよび主要な制御テーブルを計算機のメモリに配置し、I O P には、パケットバッファとわずかな制御テーブルを置くだけで済む。しかも、セグメンティング／リアセンブリを I O P が行える。

2. 3. 4 F E P の構成例

提案した F E P のシステム構成をミニコンへの適用を例にし、従来の構成（図 2. 9）と対比させて、図 2. 10 に示す。F E P を、複数のマイクロコンピュータ（I O P）と 1 台のマルチプレクサ（M P X）とから構成する。各 I O P を、マイクロプロセッサ、プログラムメモリ、バッファメモリによって構成し、計算機の入出力バスに接続する。各 I O P を集約して、ステーションに接続するため M P X を、I O P とステーションとの間に入れる。M P X には、パケット蓄積バッファを設けない。

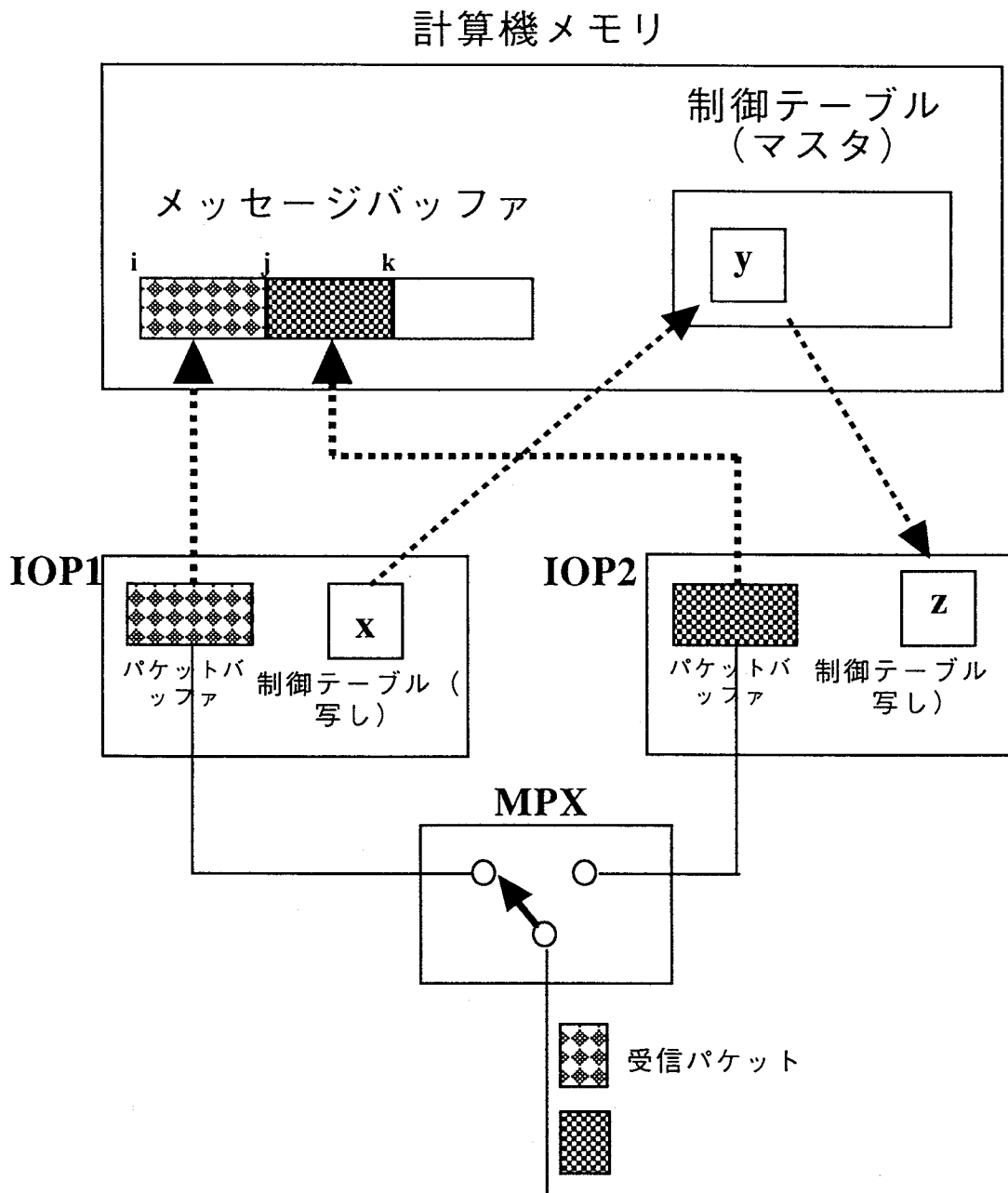


図 2.8 制御テーブルリレー方式

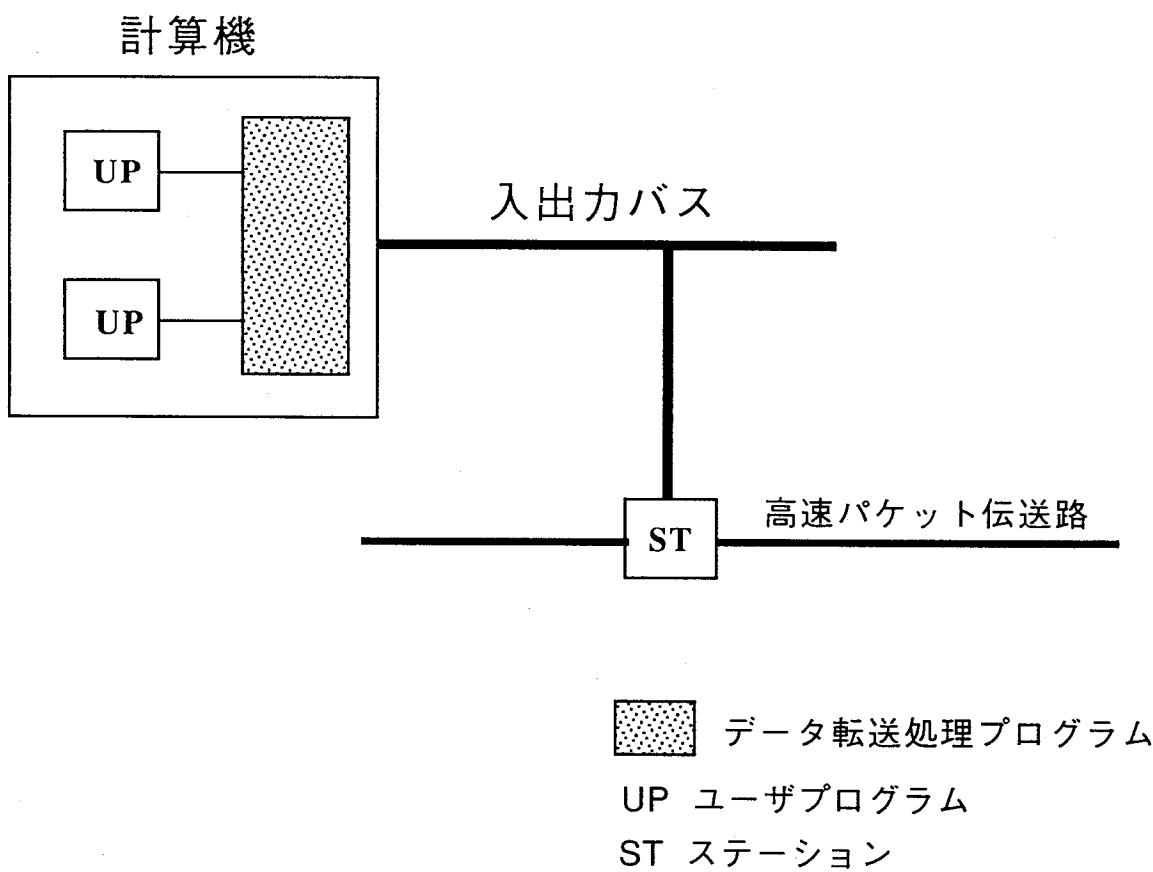


図 2.9 従来の計算機とステーションの接続形態

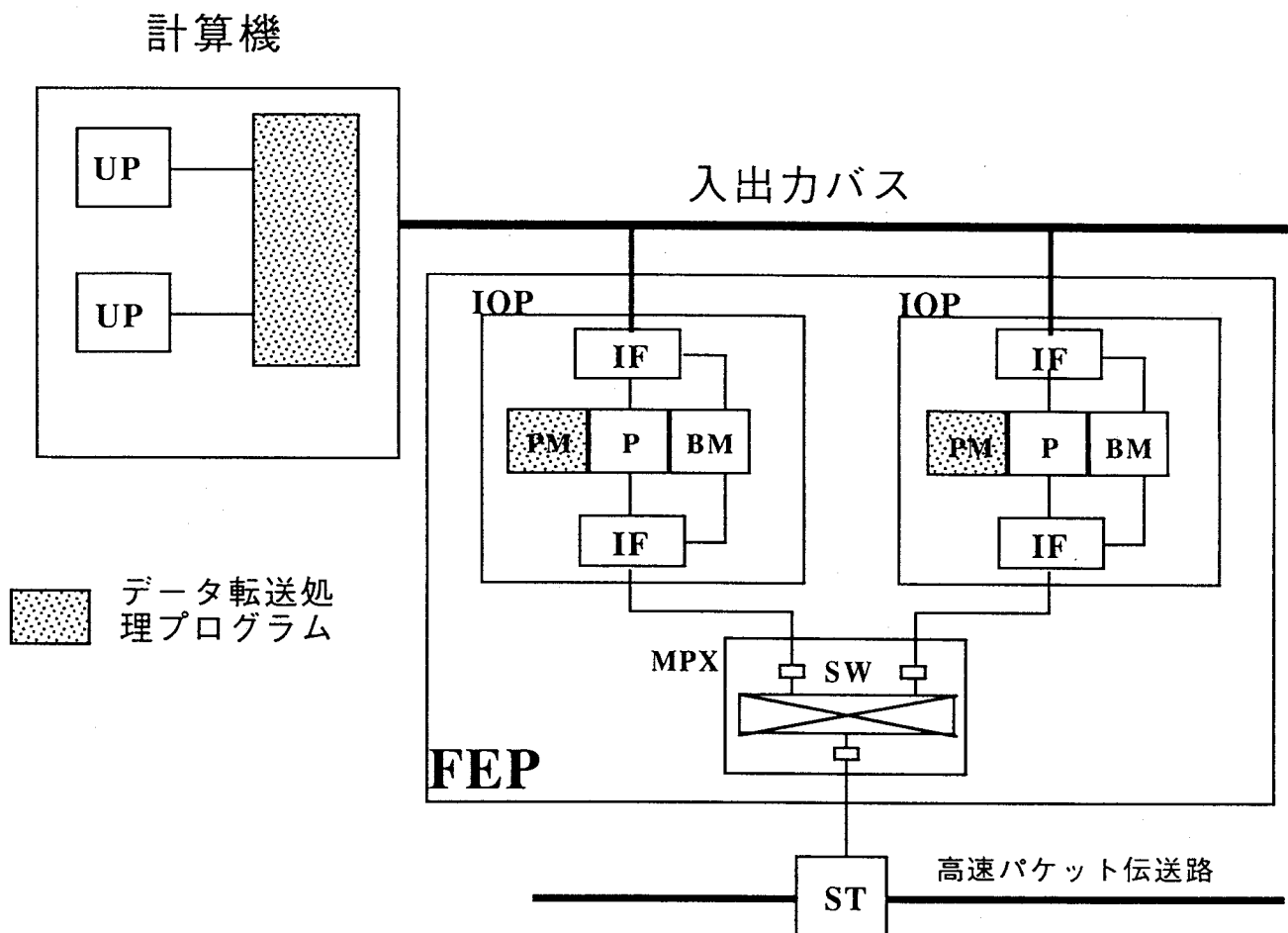


図 2.1 0 提案前置処理装置のシステム構成

本方式によれば、各マイクロコンピュータのメモリを少なくできるので、高速プロセッサ、高速バッファメモリを使うことができ、高速応答を得やすい。複数のマイクロコンピュータによる並列処理方式のため高いデータ転送性能が期待できる。さらに、F E Pでメッセージのセグメンティング／リアセンブリを行うので、計算機は、メッセージ単位にF E Pの起動と終了の処理を行うだけで済み、計算機負荷の大幅な削減が期待できる。

2. 4 実験的評価

2. 4. 1 評価の対象と目的

高速パケット伝送路を用いた計算機間のデータ転送をとりあげ、F E Pの導入効果を、スループット、応答時間、計算機負荷の三つの観点から評価する。

2. 4. 2 評価方法

3章で提案した方式に従ってF E Pを製作し、F E P付システムと従来システムとで性能を実測し、比較する。

F E Pは、2台のI O PとM P Xとから構成し、計算機の入出力バスとステーション（S T）との間に挿入した。M P Xは、すべてハードロジックで構成した。I O Pは、ビットスライス・マイクロプロセッサを用い、語長32ビットのマイクロ命令を200nsで実行できるようにした。I O Pの仕様を表2.1に示す。I O Pのマイクロプログラムは、約3600ステップで、従来、計算機のソフトウェアで行われていたデータ転送処理のうち、論理チャンネルへの振り分け、セグメンティング／アセンブリ、およびステーションの制御を行う。輻輳制御、エラーリカバリは、上位レベルで行うものとし、本評価の対象外とする。

実測には、計算機2台を光データウェイで接続したシステム（図2.11）を用いた。光データウェイは、伝送速度が10Mbpsで、パケットサイズが512バイトの高速パケット伝送路であり、ステーションを介して、F E P／計算機と伝送路とを接続する。データウェイの仕様を表2.2に示す。データリンクの制御は、このステーションによって行われる。ステーションの仕様を表2.3に示す。計算機は、語長16ビットのミニコンで、性能は、0.5MIPSである。

測定は、テストプログラムにより行った。テストプログラムは、計算機間デ

表 2. 1 I O P の仕様

項番	項 目		仕 様	備 考
1	プログラム メモリ	語長	3 2 ビット	
		容量	4 0 0 0 語	
		サイクル タイム	4 0 n s	
2	データ メモリ	語長	1 6 ビット	制御テーブル、バケットバッファ として600語を使用している
		容量	1 0 0 0 語	
		サイクル タイム	5 5 n s	
3	演算部	語長	1 6 ビット	AMD社の2900シリーズ を使用している 4ビット×4=16ビット
		レジスタ	1 6 個	
		速度	2 0 0 n s / 命令	
		命令数	7 0	
4	マイクロプログラム		3 6 0 0 ステップ	

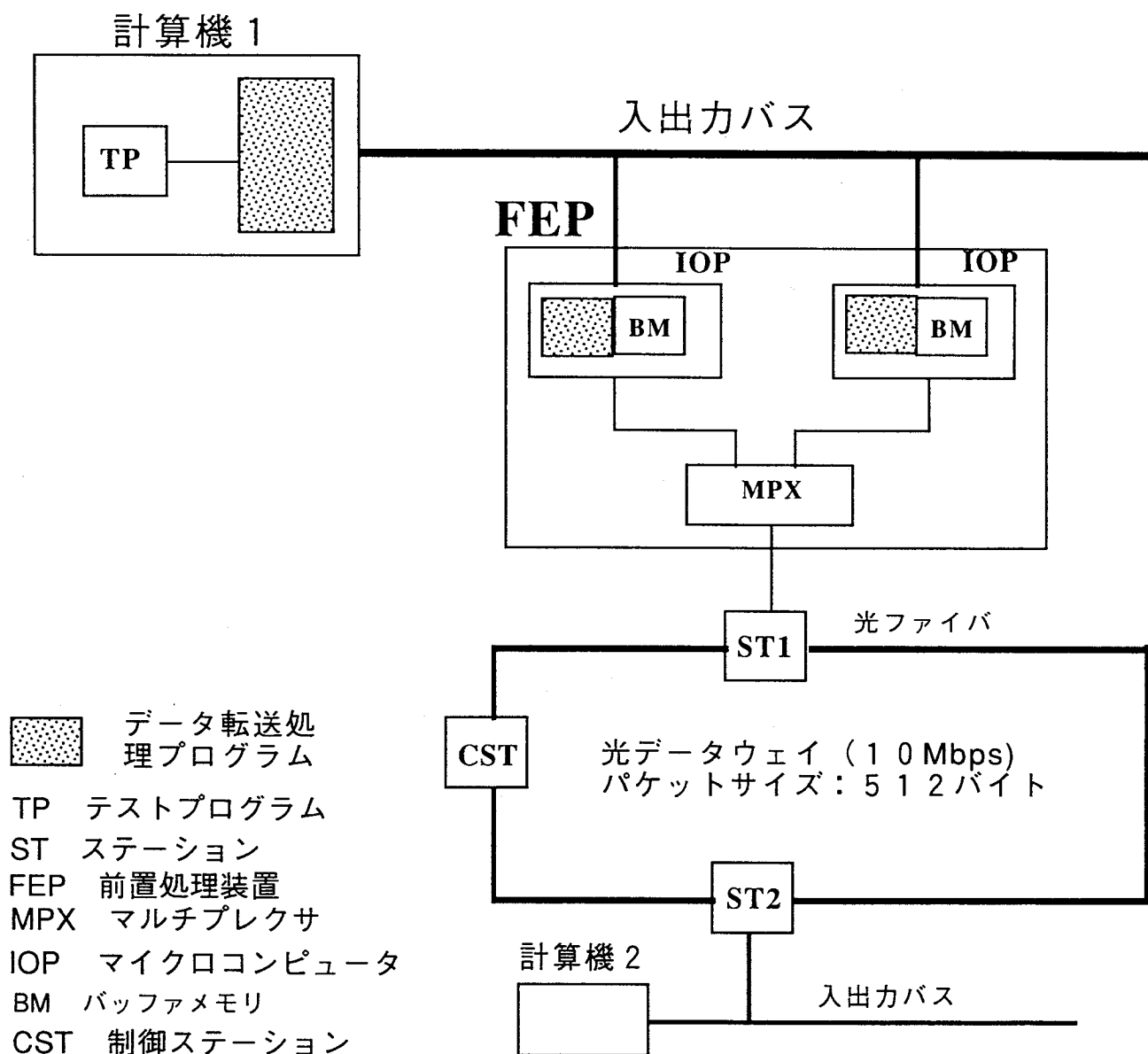


図 2.1 1 実測システムの構成

表 2. 2 データウェイの仕様

項 目	仕 様
伝送速度	1 0 M b p s
ネットワーク形態	ループ
ステーション数	最大 2 5 5 台／ループ
ステーション間距離	最大 2 k m
伝送媒体	光ファイバケーブル
パケット長	最大 5 1 2 バイト（可変長）
実効転送速度	最大 9 0 0 k バイト／秒 （3 2 ステーション、1 0 k m／ループ）
誤り制御	C R C チェック フレーム合理性チェック
障害対策	制御ステーションの二重化 ステーション障害時のバイパス ハードウェア自動診断 リモート診断 統計 R A S 情報トレース 伝送品質の監視 ループ使用率の監視

表 2. 3 ステーションの仕様

項番	項 目	仕 様	備 考
1	プログラム メモリ	語長	3 2 ビット
		容量	2 0 0 0 語
		サイクル タイム	4 0 n s
2	データ メモリ	語長	1 6 ビット
		容量	4 0 0 0 語
		サイクル タイム	5 5 n s
3	演算部	語長	1 6 ビット
		レジスタ	1 6 個
		速度	2 0 0 n s / 命令
		命令数	7 0
4	マイクロプログラム	2 0 0 0 ステップ	

ータ転送処理プログラムを用いて、計算機 1 のメモリ上のデータを計算機 2 のメモリに転送する動作を繰り返し行う。テストプログラムからの転送要求を受けて、データ転送処理プログラムは、計算機 1 から ST_2 に、データと指令（計算機 2 のメモリアドレス、データの長さなど）を送る。 ST_2 は、この指令に基づき、DMA 方式で計算機 2 のメモリに書き込む。転送要求のあったデータ長がパケットサイズより大きい場合は、何回かに分けて ST_2 に書き込み要求を行う（セグメンティング）。1 回の書き込み要求ごとに、計算機 2 のメモリへのデータ書き込みと応答パケットの返送を ST_2 が行う。従来システムと FEP 付システムのタイムチャートを図 2. 12、図 2. 13 に示す。各々の処理時間 (t_{ij}) を表 2. 4 に示す。測定条件は、下記とする。

- (a) 測定時は、テストプログラムだけが動いており、他の業務処理プログラムは全く動いていない。
- (b) スループットの測定は、二つのテストプログラムを同時に走らせて行い（多重度 2）計算機負荷と応答時間の測定は、多重度 1 で行う。
- (c) 性能は数分間同じ動作を繰り返し、その間の平均値で算出する。

2. 4. 3 結果

(1) データ転送スループット

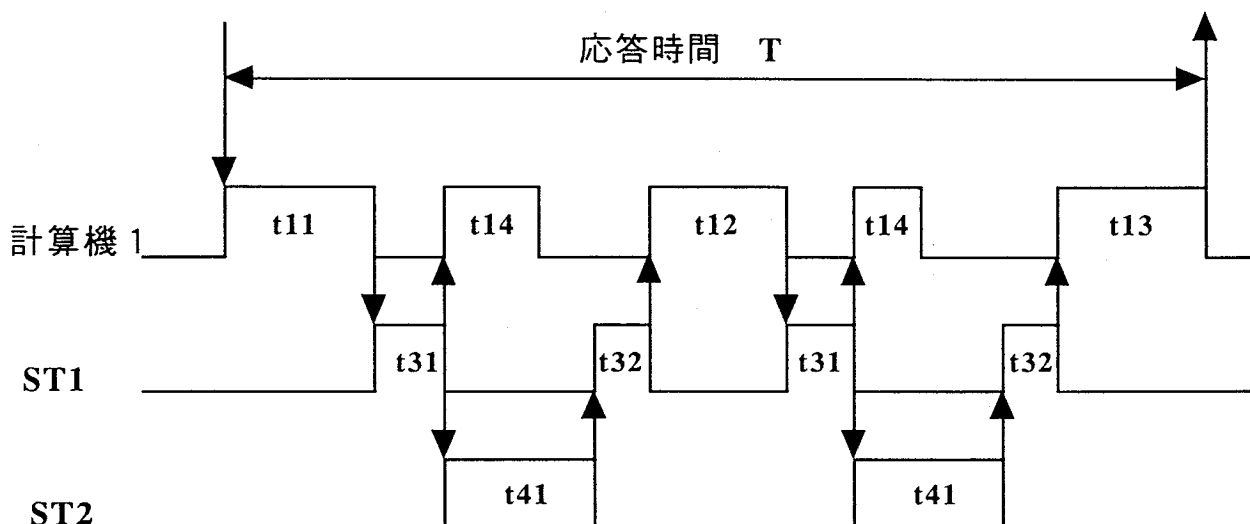
単位時間あたり、計算機 1 から計算機 2 に転送できたデータ量でスループットを定義する。この定義のもとに、種々のデータ長でスループットを測定し、比較したのが図 2. 14 である。これにより、データ長が 1 k バイトのとき 2.5 倍、8 k バイトのとき 3.5 倍スループットが向上することがわかった。

(2) 応答時間

応答時間として図 2. 12 の T および図 2. 13 の T' をとり、種々のデータ長で測定した結果を図 2. 15 に示す。サイクルタイム 55 ns の高速データバッファメモリ、5 MIPS の高速マイクロコンピュータを用いたこと、マルチプレクサでのパケットの蓄積を不要にしたことにより、応答時間は、データ長 60 バイトで 1/2 に改善できた（データ長が長くなると改善度は少なくなるが、通常応答時間が最も問題になるのはデータ長の短いところである）。

(3) 計算機負荷

データ長を変えて、テストプログラムを繰り返し走らせて計算機負荷を測定し、FEP 付システムと従来システムの差をプロットしたのが図 2. 16 である。これより、データ長が 8 バイトと短いところでも 40% 計算機負荷が減り、8 k バイトという長いデータの場合は、80% 負荷が減ることがわかった。

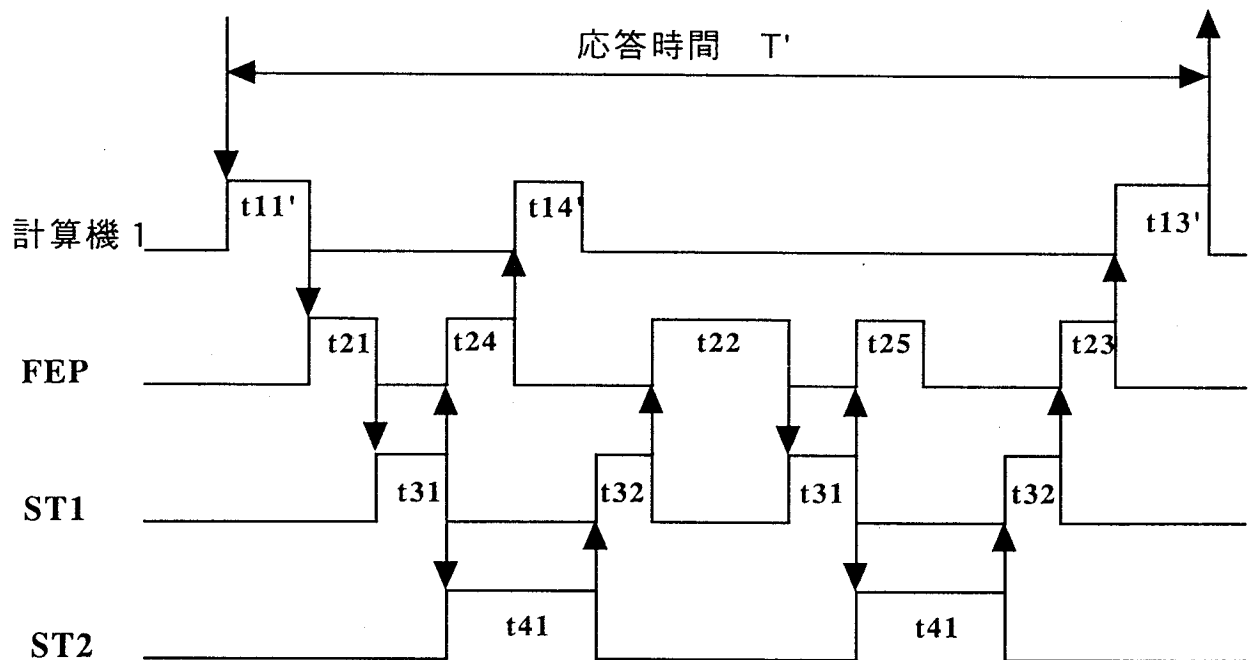


t_{ij} 処理時間（詳細は、本文参照）

FEP 前置処理装置

（2 セグメントの例、セグメント数はデータ長により変わる）

図 2.1 2 従来システム（FEPのない場合）の
データ転送タイムチャート



t_{ij} 処理時間（詳細は、本文参照）

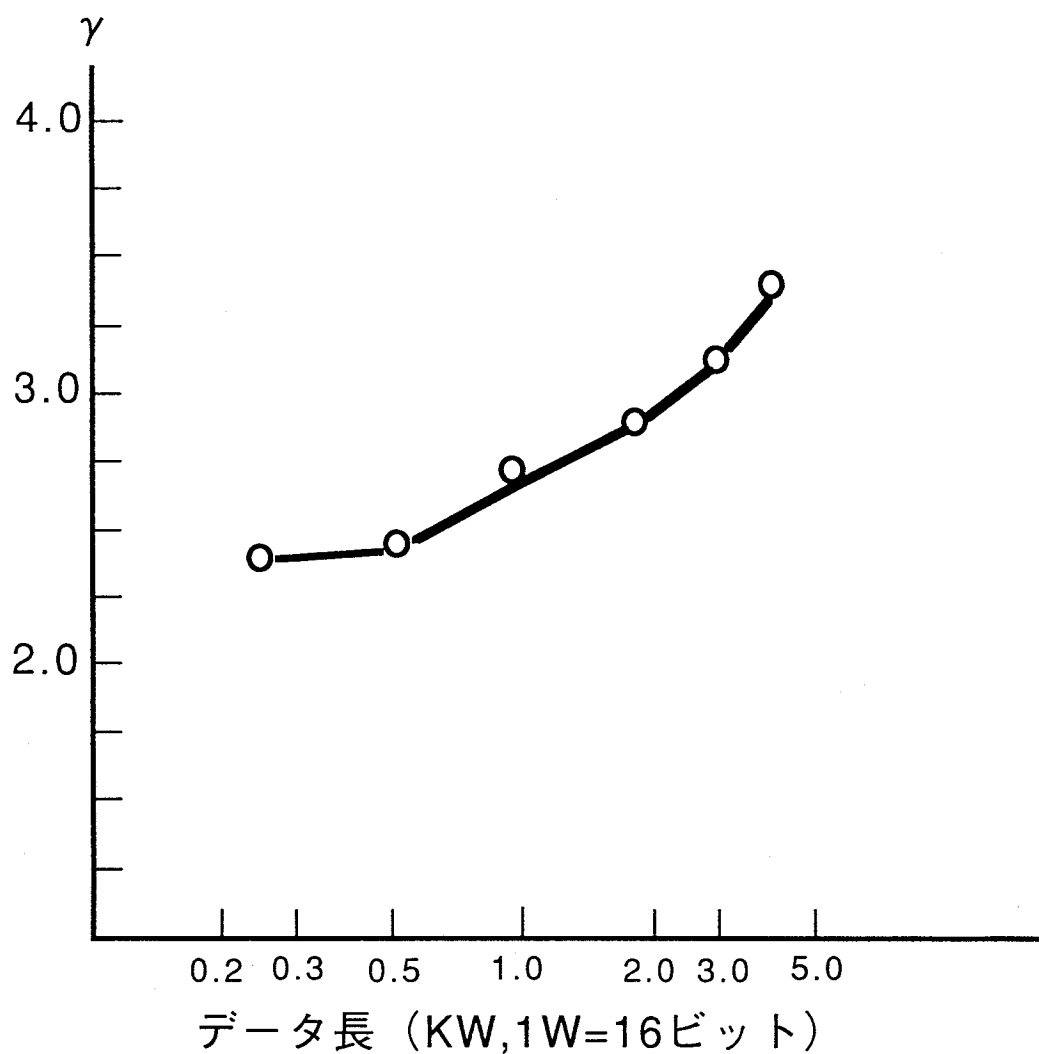
FEP 前置処理装置

（2 セグメントの例、セグメント数はデータ長により変わる）

図 2.1 3 FEP付きシステムのデータ
転送タイムチャート

表 2. 4 処理時間 t_{ij} の内容

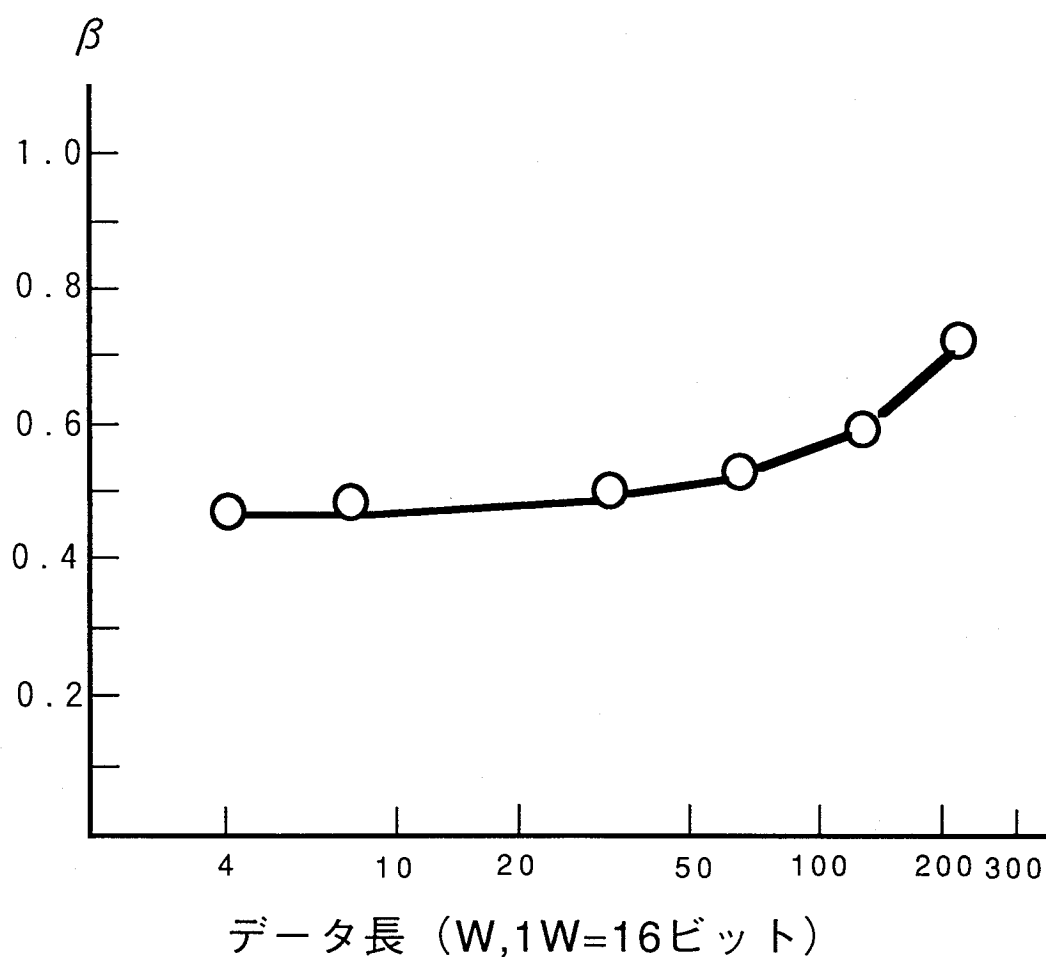
t_{ij}	処 理 内 容
t_{11}	テストプログラムから要求を受けて必要なヘッダを作成し、ステーションに送信要求を行う。
t_{12}	相手 ST からのレスポンスを受け、次のセグメントの送信要求を行う。
t_{13}	最後のセグメントに対するレスポンスを受けテストプログラムを再起動する。
t_{14}	パケット送信後処理。
t_{11}'	テストプログラムから要求を受けて FEP に送信要求を行う。
t_{13}'	FEP からの割り込みを受け、テストプログラムを再起動。
t_{14}'	IOP の BUSY を解除。
t_{21}	ヘッダを作成し、ステーションに送信要求を行う。
t_{22}	相手 ST からのレスポンスを受け、次のセグメントの送信要求を行う。
t_{23}	最後のセグメントに対するレスポンスを受け、計算機に割り込みを発生する。
t_{24}	IOP の BUSY を解除。
t_{25}	パケット送信後処理。
t_{31}	パケットの送信 (ST_1)
t_{32}	パケットの受信 (ST_1)
t_{41}	メモリへの書き込みを行い応答パケットを送信 (ST_2)



$\gamma = \gamma_1 / \gamma_0$
 γ_0 : FEPのないシステムのスループット
 γ_1 : FEP付きシステムのスループット

条件： 2つのテストプログラムを同時に走らせ単位時間に計算機1から計算機2に転送できた合計データ量で比較した。

図 2.1 4 スループット比



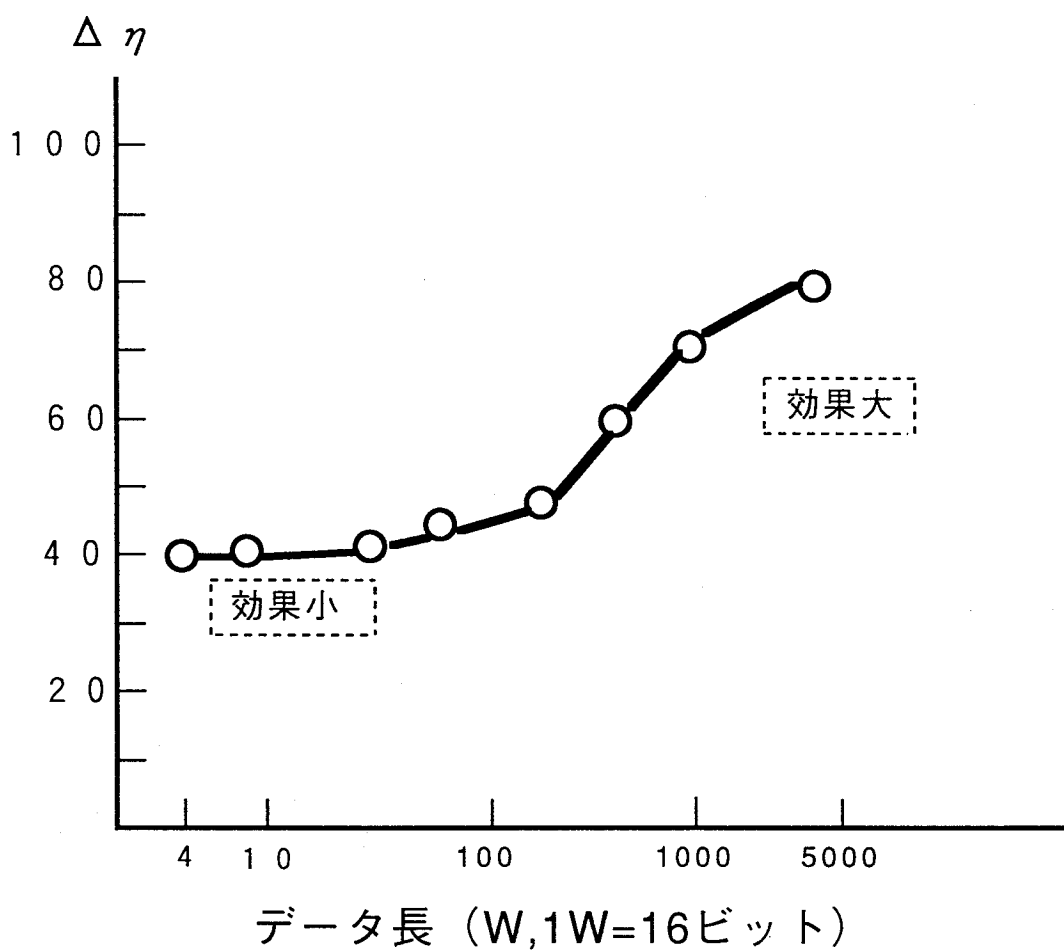
$$\beta = \beta 1 / \beta 0$$

$\beta 0$: FEPのないシステムの平均応答時間

$\beta 1$: FEP付きシステムの平均応答時間

条件： 1つのテストプログラム動かしたときの平均応答時間（図2.12、図2.13のTおよびT'）を測定する。

図2.15 応答時間比（FEPなしのときを1）



$$\Delta \eta = \eta_0 - \eta_1$$

η_0 : FEPのないシステムの計算機1の負荷

η_1 : FEP付きシステムの計算機1の負荷

条件：1つのテストプログラム動かしたときの計算機1の負荷を測定する。

図 2.1 6 計算機負荷低減効果

2. 5 結言

高速パケット通信ネットワークによって複数の計算機を接続した計算機制御システムを対象に、計算機間のデータ転送スループット、応答時間、計算機負荷などの性能を向上する方式を検討し、マルチマイクロコンピュータ構成のトランスポートプロセッサ（前置処理装置処理：F E P）とデータウェイステーションとから構成する機能階層形多重通信プロセッサ方式を提案した。提案方式による性能向上度を把握するため、マイコン2台から成るF E Pを試作し、伝送速度10Mbpsの高速パケット通信ネットワークに適用した。実験の結果、計算機間データ転送性能は、F E Pが無いときに比べて、次のように改善されることが明らかになった。

- (1) データ転送スループットが3倍になる（データ長が4kBのとき）。
- (2) 応答時間が1/2になる（データ長が60Bのとき）。
- (3) 計算機負荷が40%（データ長が8Bのとき）～80%（データ長が8kBのとき）削減できる。

提案方式は、計算機制御システムにおいて、データウェイを用いた場合について論じた。これ以外のシステムでも高速パケット通信ネットワークと計算機との接続の場合には、提案方式が性能向上に役立つものと考えられる。

第 3 章 機能階層形多重通信プロセッサにおけるコマンド多重処理方式

3. 1 緒言

近年計算機制御システムの大規模化が進み、計算機間のデータ転送性能の向上が強く求められている[4]。第2章では、トランスポートプロセッサ（前置処理装置：F E P）とデータリンクプロセッサ（データウェイのステーション）とから構成する機能階層形多重通信プロセッサによって、データ転送性能を向上する方式をとりあげ、F E Pの方式提案（F E P I）と性能向上度合の実験的評価とを行った。本章では、F E P Iで問題になったハード量を削減するため、新しいF E Pの方式提案（F E P II）と実測およびシミュレーションによる性能評価とを行っている。

- (1) ハード量削減のため、F E Pを1台の高速マイクロコンピュータで構成し、ソフトウェア方式の工夫によって多数のコマンドを同時に処理する。
- (2) 性能を向上するため、F E Pのソフトウェア構造をコンパクトなスケジューラのもとで動くマルチタスク構造とし、処理の並列化を行う。
- (3) この方式に基づき、ビットスライスマイクロプロセッサを用いて、F E Pを開発し、性能をシミュレーションで求め、マイコン1台でも十分な性能が得られることを確認している。

3. 2 機能階層形多重通信プロセッサ方式実用化の課題

3. 2. 1 F E Pの位置付け

計算機制御システムの中で、データウェイによって、複数の計算機が接続されている例[17][18][19]をとりあげ、以下の議論を進める。データウェイは、光ファイバ／同軸ケーブルとステーションとで構成されている。ステーション相互間では、パケットと呼ばれる500～1500バイト程度の可変長データブロックが10～32Mbpsでビットシリアルに伝送される。

計算機間のデータ転送は、主にホスト計算機と10～20台の端末計算機との間で行われる。すなわち、計算機のユーザプログラム相互間には、論理チャネルと呼ばれる仮想的な伝送路が定義され、論理チャネルを介して、ユーザプ

ログラム間でメッセージがやりとりされる。F E Pは、図 3. 1 に示すように、ホスト計算機とステーションとの間にあって、従来ホスト計算機で行っていたパケット単位 of データ転送処理をホスト計算機に代わって高速に実行する。

ホスト計算機とF E Pのインタフェースは、仮想的な伝送路である論理チャネルである。各論理チャネル上でのデータ転送手順は、図 3. 2 に示すように、パケット毎に応答をとる方式である。ホスト計算機からは、メッセージ単位にF E Pを起動し、パケットへの分割（セグメンティング）／編集（アセンブリ）は、F E Pで行う。F E Pは、プロトコルのレイアで表現すれば、トランスポートレベルの処理まで行う。

3. 2. 2 従来方式の問題点

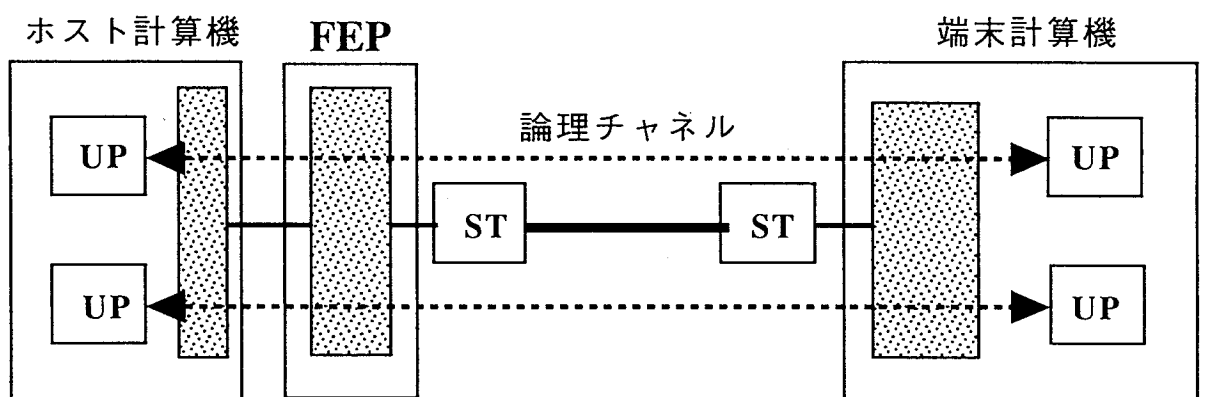
F E Pの処理は、コマンドが発行されている多数の論理チャネルに対して並列に行われる。この結果、F E Pの負荷は重く、メモリも多く必要になる。これを解決するため、第2章では、F E Pを複数のマイクロプロセッサで構成する方式を提案し、効果を評価した。この方式は、ステーションとF E Pのマイクロプロセッサとの間に、マルチプレクサ（M P X）と呼ぶ装置を挿入することによって、高速に受信パケットをF E Pのプロセッサに分配し、各プロセッサで並列に処理する方式である（以下、この方式のF E PをF E P Iと呼ぶ）。しかし、F E P Iには、実用化する上で次の問題があることがわかってきた。

- (1) 処理を高速化するため、マイクロプロセッサとしては、集積度の低いバイポーラ技術を用いたビット・スライス・マイクロプロセッサを使う必要がある。このため、マイクロプロセッサ一台が基板（30.5×42.5 cm）一枚のハード量となる。
- (2) マルチプレクサが、マイクロプロセッサ2～6台とインタフェースするために基板一枚のハード量となり、最小構成でもプロセッサと合わせて基板二枚になる。
- (3) 個々のマイクロプロセッサは、同時には、一つのコマンドのみ処理する方式にしているため、パケットの送信中、マイクロプロセッサがアイドル状態になっている。

このように、F E P Iでは、性能向上効果はあるが実用化するには、ハード量が大きくなりすぎるという問題があった。

3. 2. 3 改良形F E P（F E P II）

本章では、前記F E P Iの問題点を解決した新しい方式のF E Pを提案する



FEP 前置処理装置

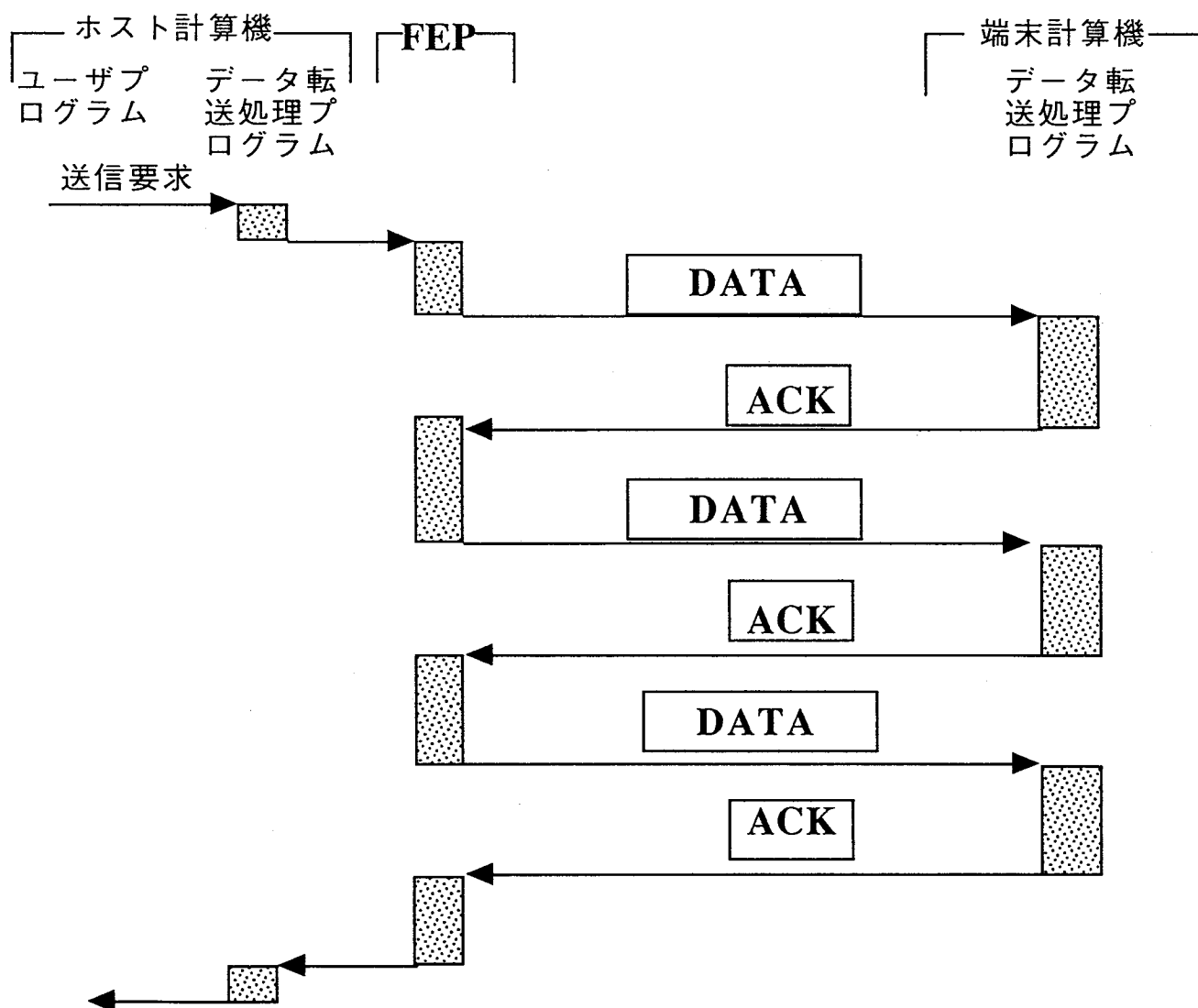
—— 高速パケット伝送路

UP ユーザプログラム

ST ステーション

 データ転送処理プログラム

図 3.1 前置処理装置 (FEP) の位置付け



DATA ユーザデータパケット
ACK 肯定応答パケット
FEP 前置処理装置

論理チャネルレベルのデータ転送モデルを示している

図 3.2 FEP適用時のデータ転送モデル

(これをF E P IIと呼ぶ)。F E P IIでは、下記の方針により、ハード量を減らす工夫をする。

- (1) マイクロプロセッサはステーション対応に一台として、マルチプレクサ(M P X)を不要にする。
- (2) マイクロプロセッサは、同時に複数のコマンドを処理できるようにして、アイドルタイムを極力減らし、一台である程度の性能を確保する。

上記改良により、マルチプレクサ(M P X)が不要になり、マイクロプロセッサが一台ですむので、ハード量の低減が期待できる。

3. 3 コマンド多重処理方式の開発

本節では、高速パケット伝送路に適したコストパフォーマンスのよい前置処理装置(F E P II)のコマンド多重方式について論じる。

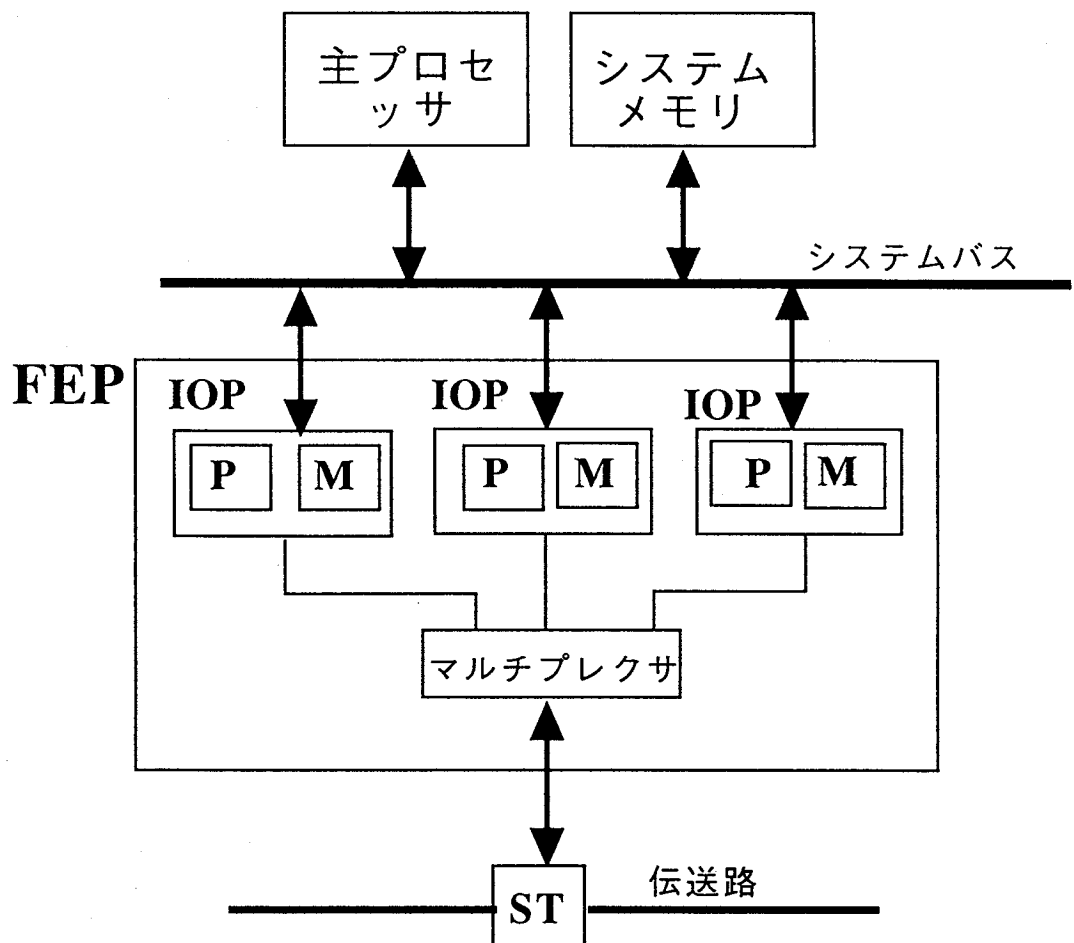
3. 3. 1 要求条件とF E Pの方式提案

高速パケット伝送路を用いた計算機制御システムにおいて使用されるF E Pには、下記の要求条件がある。

- (1) 10～32Mbpsの伝送速度に適應できる高いデータ転送スループット
- (2) 応答時間が数ミリ秒という高速応答性
- (3) 制御用計算機のハード規模に見合うコンパクトなハードウェア

前記要求条件を満たすF E P構成方式としては、マルチプロセッサ・シングル・コマンド処理方式とシングルプロセッサ・マルチコマンド処理方式とが考えられる。前者は、高性能化には適すがハード量が増大するという欠点があり、後者は、ハード量を削減できるが、高性能化に弱点がある。本章では、後者のシングルプロセッサ・マルチコマンド処理方式を採用し、コマンド多重処理などのソフトウェア技術を駆使して高性能化を図ったF E P(F E P II)を提案する。

F E P IIの構成を、従来のマルチプロセッサ方式のF E P(F E P I)(図3. 3)と対比して、図3. 4に示す。F E P IIでは、マイクロプロセッサを一台にし、ステーションとマイクロプロセッサを1:1に対応させて、マルチプレクサ(M P X)を不要にしている。F E P IIのプロセッサ仕様を、F E P Iのプロセッサ(I O P)と対比させて、表3. 1に示す。F E P IIでは、基



FEP 前置処理装置
 ST ステーション
 IOP マイクロコンピュータ
 M メモリ
 P プロセッサ

図 3.3 マルチプロセッサ方式によるFEPの構成

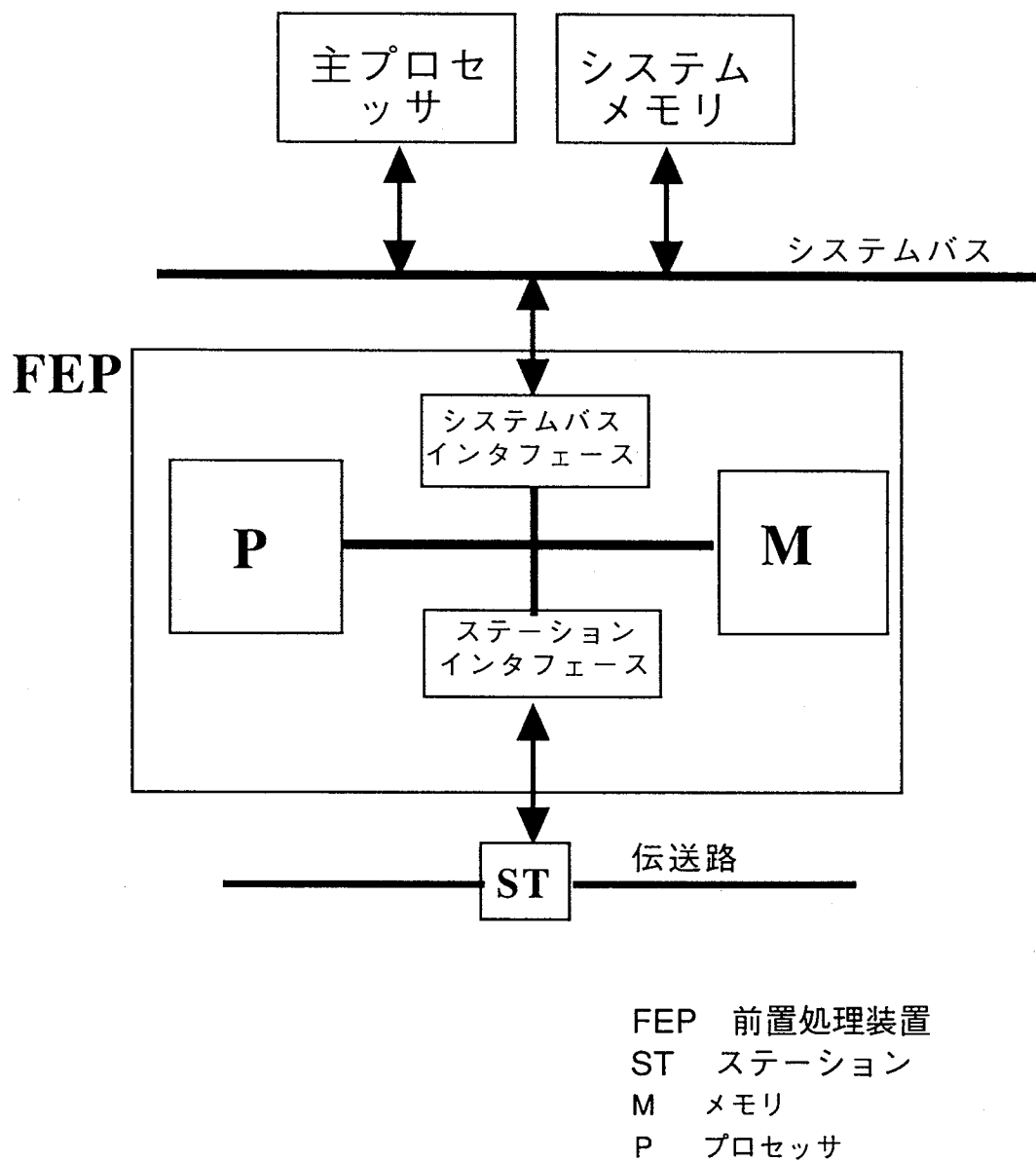


図 3.4 シングルプロセッサ方式によるFEPの構成

表 3. 1 F E P に使用したプロセッサボードの比較

項番	項 目		F E P - I の I O P	F E P - II
1	プログラム メモリ	語長	3 2 ビット	←
		容量	4 0 0 0 語	8 0 0 0 語
		サイクル タイム	4 0 n s	5 0 n s
2	データ メモリ	語長	1 6 ビット	←
		容量	1 0 0 0 語	1 6 0 0 0 語
		サイクル タイム	5 5 n s	←
3	演算部	語長	1 6 ビット	← (AMD社の2900シリーズ 4ビット×4=16ビット)
		レジスタ	1 6 個	
		速度	2 0 0 n s / 命令	
		命令数	7 0	
4	ハード量 (プラグイン枚数)		1	2

板 2 枚に、①一命令 200 ns の高速マイクロプロセッサ、②プログラムメモリ：8 kW（F E P I の 2 倍）、③データメモリ：16 kW（F E P I の 16 倍）④ステーションとのインタフェース回路、⑤システムバスとのインタフェース回路をすべて収容している。

上記方式により、ハード量は、基板 2 枚にできる見通しを得られたが、コマンド多重処理ソフトウェア方式をどうするか、そして、それによってどの程度の性能が得られるかが問題である。次に、これらの点について検討する。

3. 3. 2 F E P II のソフトウェア処理方式

F E P II では、ハード量を減らすため、マイクロプロセッサを 1 台にしたため、高度なソフトウェア処理によって、プロセッサの能力を十分に活用する必要がある。

(1) コマンドの多重受付

F E P がトランスポートレベルの処理まで行うためホストから F E P へは論理チャンネル対応に下記のコマンドを多重に発行できるようにする。

- ・ R E C V : 論理チャンネルを指定し、メッセージの到着を待つ。
- ・ S E N D : 論理チャンネルを指定し、メッセージを送信する。

ここで、メッセージの長さは、伝送路のパケット長を越えていてもよいものとする。コマンドは、論理チャンネル対応に多重に受け付けるが、同一論理チャンネルには、一つのコマンドのみ受け付ける。

(2) コマンドの並列処理

コマンド並列処理とは、ホスト計算機から受け付けたコマンドを F E P 内部で並列に処理することである。送信（S E N D）を例にとると、各コマンドはまずプロセッサのソフトによる起動処理を受け、送信データがステーションを経由して伝送路に送出され、相手計算機に送られる。やがて相手計算機から応答が返される。応答が、ステーションで受信されると、コマンドは、再びプロセッサによる継続処理（セグメンティングがある場合）または終了処理を受ける。

F E P II では、次のようにして、コマンドの並列処理を実行している。

- ・ プロセッサの処理を細分化し、各々の処理をタスクとして、コンパクトなスケジューラで制御するマルチタスク構造とする。
- ・ 各コマンドは、コマンド処理テーブルに格納された形で、順次タスクの処理を受けていけるように各タスクの前に待ち行列を作っておく。

- ・各タスク T_i は、起動されると自タスクの前の処理待ち行列 Q_i を調べ、処理待ちコマンドがあれば処理を行い、そのコマンドを次のタスク T_j の待ち行列 Q_j につけ、タスク T_j を起動する。
- ・スケジューラは起動されたタスクを次々とディスパッチしていき、プロセッサ資源を、処理を必要としているタスクにきめ細かく配分する。

(3) メモリの節約

コマンドの多重処理を行うには、多数のコマンド処理テーブルが必要になる。しかし、ハード量を減らすには、メモリを節約する必要がある。コマンド処理テーブルのメモリ容量を削減するため下記の方策をとった。

- ・コマンド処理テーブルの動的割当

コマンドがすべての論理回線に同時に発行される可能性は小さいことに着目し、コマンド処理テーブルを動的に論理チャンネルに割り当てる方式にした。

- ・パケットバッファリング

コマンドの処理には、ユーザデータのバッファリングが必要になる。とりわけ、FEPでセグメンティング／アセンブリを行う場合、メッセージバッファをFEPで持つと、FEPのメモリ量が増大する。そこで、FEPⅡでは、メッセージバッファはホストプロセッサの主メモリに置き、FEPではパケットバッファのみ持つことにした。セグメンティング／アセンブリは、FEPが主プロセッサのメモリを直接アクセスすることによって行う。

- ・パケットバッファとコマンド処理テーブルは分離しておき、送受信時のみ、パケットバッファを動的に確保する。

3. 3. 3 FEPⅡの性能評価

FEPⅡで採用したシングルプロセッサ・マルチコマンド処理方式では、コマンド多重処理による高性能化が重要な課題である。FEPⅡでは、前節で述べたソフトウェア処理方式によって、プロセッサの処理能力を最大限に引き出し、高性能化をはかっている。本節では、メッセージ転送スループットを例に、FEPⅠとFEPⅡの性能をシミュレーションにより求め、比較する。

(1) シミュレーションモデル

計算機間のメッセージ転送を想定する。メッセージは、計算機内のユーザプログラムから別の計算機のユーザプログラムに渡される。ここでは、図3.5に示すように、一台の計算機に付加されたFEPに着目して行う。

メッセージは、伝送路に送出されるとき、FEPによってパケットに分割さ

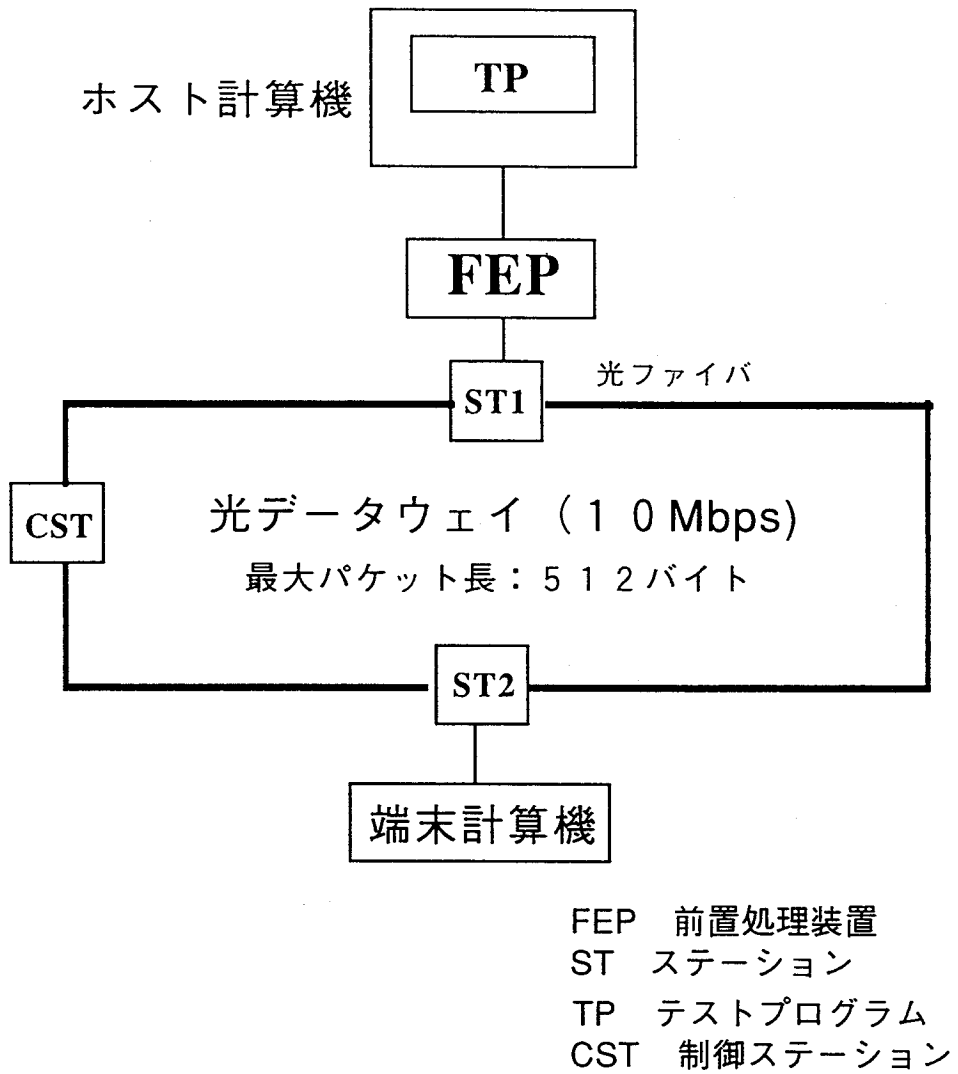


図 3.5 性能比較のためのモデルシステム構成

れ、パケット毎に相手計算機から応答（ACK）を受け取るものとする。この分割されたメッセージの断片をセグメントと呼ぶ。

メッセージ転送処理のタイムチャートを図3.6に、各処理時間を定義と共に表3.2に示す。処理時間は、1パケット（512バイト）の伝送時間を単位として示している（また処理時間は、FEP IとFEP IIの具体ハード、ソフトの処理時間を根拠としている）。なお、相手計算機でメッセージパケットを受信後、応答パケットを返送するまでの時間 t_{51} は、FEPの性能と直接関係しないので0としている。同様に、 t_{32} 、 t_{42} も0としている。

FEP Iでは、プロセッサがステーションに送信要求した後も、ステーションからの送信終了割り込みを待ってアイドル状態となり解放されないことに注意する必要がある（シングルコマンド処理方式）。これに対して、FEP IIでは、ステーションに送信要求を行った後、プロセッサは、次のコマンドを処理できる（コマンドの多重処理）。

(2) シミュレーションの条件

シミュレーションは、GPS Sを用いて下記の条件で行った。

- ・メッセージ送信要求は、ポアソン到着とする。
- ・メッセージの長さは、平均2000バイトの指数分布とする。
- ・パケットの最大長は、512バイトとする。
- ・伝送速度は、10Mbpsとする。

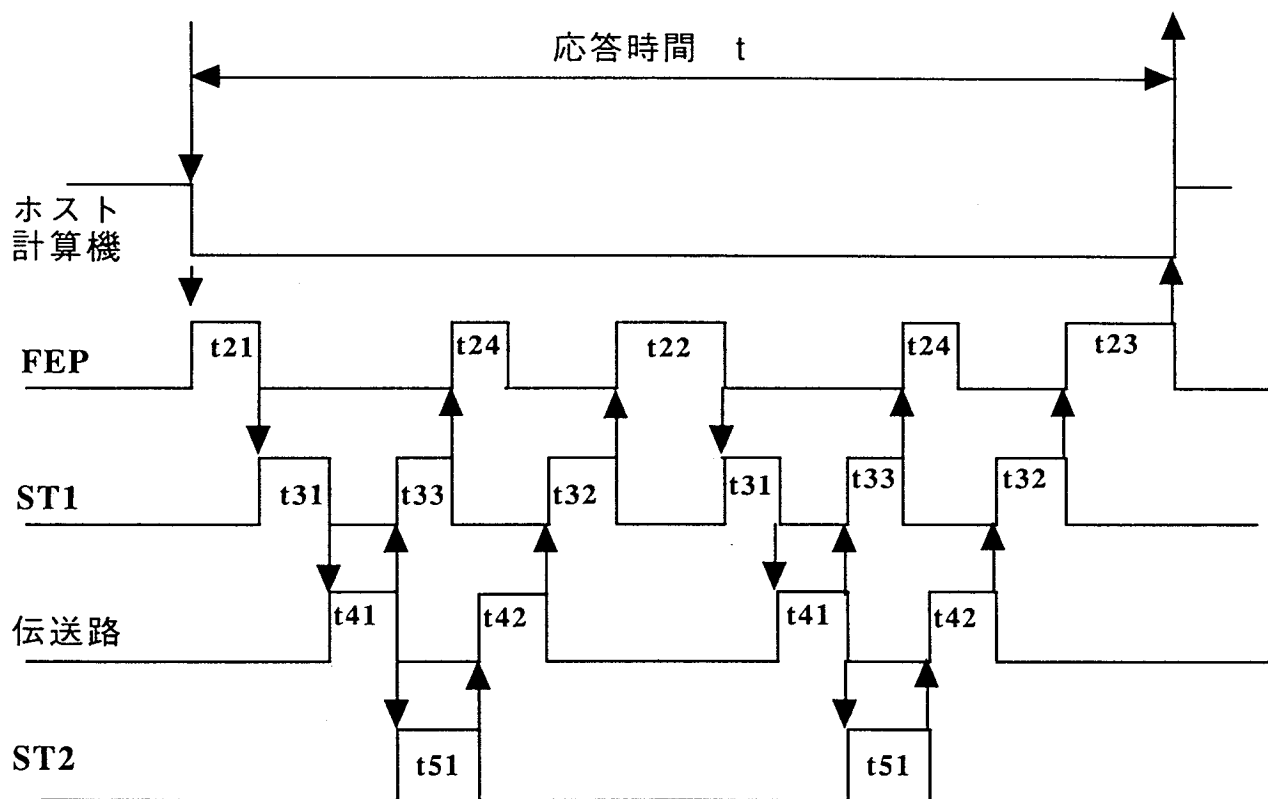
(3) シミュレーション結果

単位時間あたり、計算機から伝送路へ転送できたデータ量を、メッセージ転送スループット、計算機がFEPに送信要求してから、送信が終了して、計算機に終了を通知するまでの時間（図3.6の t ）を応答時間として、スループットと応答時間の関係を図3.7に示す。この結果、表3.2の処理時間のもとでは、FEP IIの性能は、FEP Iの2プロセッサとほぼ同等であることがわかった。

3.3.4 FEP IIの方式評価

シングルプロセッサ多重コマンド処理方式のFEP IIを提案し、FEP Iと対比しつつ、方式や性能を述べてきた。本節では、これまでの検討結果を整理し、FEP IIの方式をコストパフォーマンスを中心に評価する。

FEP IとFEP IIの二つの方式を表3.3に整理して示す。FEP Iのマルチプロセッサ方式では、プロセッサの数を、1、2、3と増やすことによって性能を向上することができ、3台では最大0.25程度のスループットが得ら



t_{ij} 処理時間（詳細は、本文参照）

FEP 前置処理装置

（2 セグメントの例、セグメント数はデータ長により変わる）

図 3.6 性能比較モデルタイムチャート

表 3. 2 性能比較モデルで用いた処理時間

t_{ij}	F E P I	F E P II	処 理 内 容
t_{21}	3. 1 *	4. 8 *	ホスト計算機から起動され、最初のセグメントをステーションに送信するまでの時間
t_{22}	4. 8 *	5. 9 *	端末計算機からの応答を受けて、次のセグメントをステーションに送信するまでの時間
t_{23}	2. 1	2. 2	端末計算機からの最終セグメントに対する応答を受けて、ホスト計算機に終了を通知するまでの時間
t_{24}	0. 6	0. 3	ステーションからの送信完了割り込みの処理時間
t_{31}	2. 2	2. 2	1 パケットの送信に要するステーションの処理時間
t_{33}			
t_{41}	1. 0 *	1. 0 *	1 メッセージパケット（最大 5 1 2 バイト）の伝送時間
t_{32}	0	0	ステーションにおける応答パケット受信処理時間
t_{42}	0	0	応答パケット（約 6 0 バイト）の伝送時間
t_{51}	0	0	受信ステーション側でメッセージパケットを受信して、応答を返すまでの時間

* データ長により異なる。この数値は、5 1 2 バイトのときのもの。

各処理時間 t_{ij} は、1 パケット（5 1 2 バイト）の伝送時間（ t_{41} ）を単位として表している。

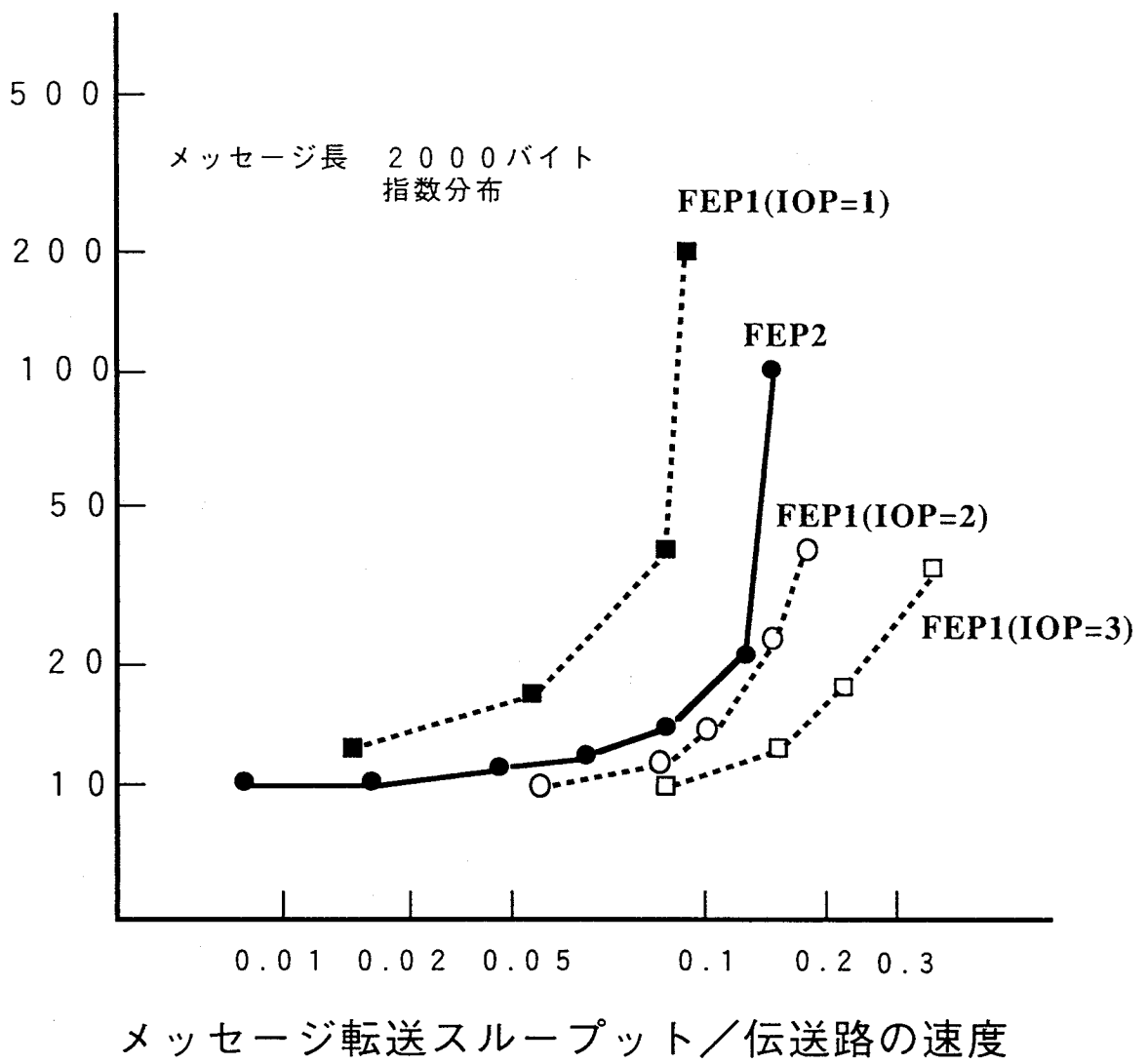


図 3.7 FEPの性能比較（シミュレーション）

表 3. 3 F E P の方式比較

項番	項 目	F E P - I の I O P		F E P - II
1	プロセッサ数	n		1
2	同時処理数 (1 プロセッサあたり)	1		m
3	マルチプレクサ	要		不要
4	制御の複雑さ	ハード複雑		ソフト複雑
5	ハード量 [基板枚数]	n + 1		2
6	プログラム規模 (ステップ)	3 6 0 0		8 0 0 0
7	性能 (伝送速度との比)	n = 1	0. 0 5	0. 1 4
		n = 2	0. 1 5	
		n = 3	0. 2 5	

れる。これに比してシングルプロセッサ方式である F E P II では、約 0.14 であり、これ以上の性能が必要な場合には、ステーション + F E P II をもう一式置く必要がある。ハード量と性能の関係を図 3.8 に示す。これより 0.05 ~ 0.14 においては、F E P II が F E P I よりコストパフォーマンスがよいことがわかる。

以上の検討から、シングルプロセッサ方式によってハード量を基板 2 枚に収め、しかも、2 台のプロセッサによるマルチプロセッサ方式に相当する高性能をソフトウェアのコマンド多重処理により得られることがわかった。

3.4 適用評価

3.4.1 評価の対象と目的

F E P II を制御用 32 ビットスーパーミニコンに適用し、高速パケット伝送路である 10 M b p s 光データウェイ用前置処理装置として機能させ、その効果を評価する。

3.4.2 評価方法

3.3 節で提案した方式に従って F E P を製作し、F E P の各部処理時間をマイクロプログラムのステップ数から算出する。こうして得られた処理時間をもとに、シミュレーションにより、応答時間、スループット、主プロセッサ利用率を求める。同じ事を、F E P を適用していない 16 ビット制御用ミニコンの場合に行い比較する。

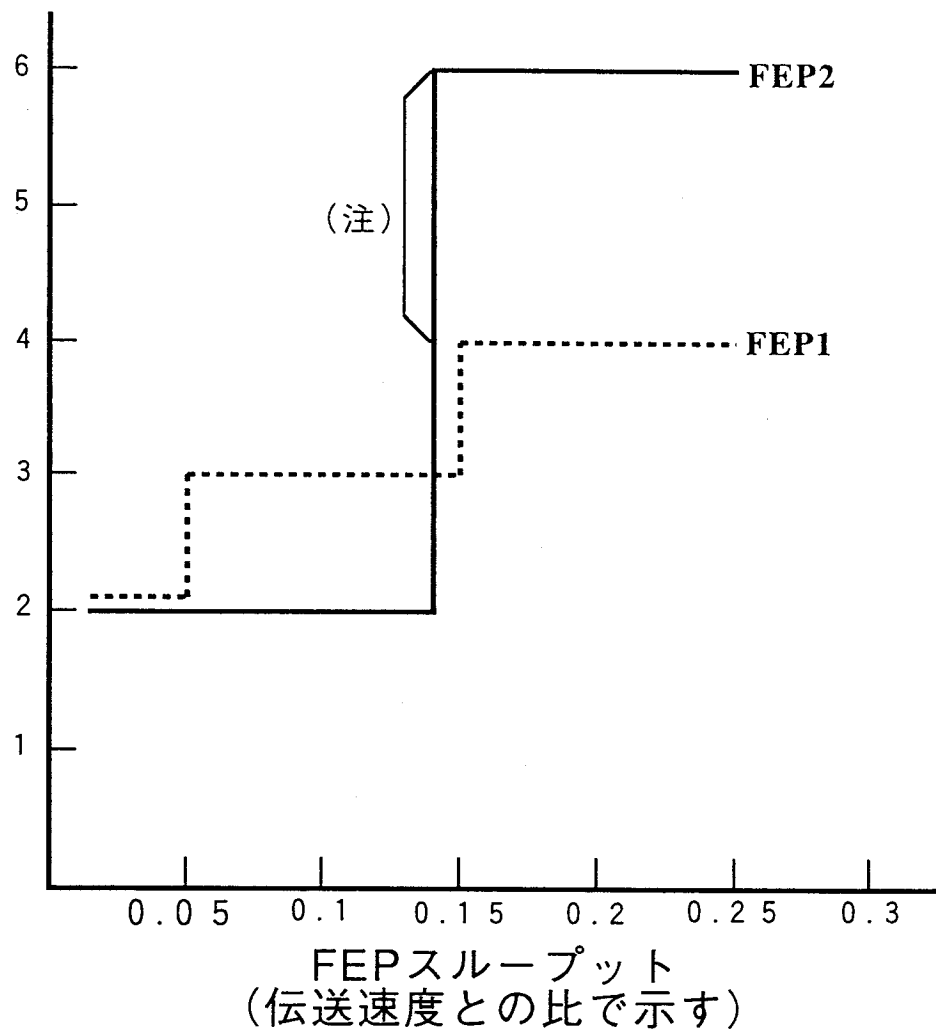
(1) 性能評価システムの構成

2 台の計算機が高速データウェイで接続された図 3.9 のようなシステムを考える。F E P II は、ホスト計算機に D W P として接続されている。データの転送は、ホスト計算機のユーザプログラムからの送信要求の発行によって行われる。パケット長を超えるメッセージの送信が指令された場合、F E P は、セグメンティングを行う。1 セグメントごとに、相手計算機からの応答を待つ。データ転送タイムチャートを従来システムの場合（図 3.10）と対比させて図 3.11 に示す。各々の処理時間（ t_{ij} ）は、表 3.4 のとおりである。

(2) 性能の定義

評価で用いる性能は、下記のごとく定義する。

応答時間：ユーザプログラムからの送信要求が発行された時点から、データ



(注) FEP2は、2 台目を設置するとき、ステーションがもう 1 台必要になる。
 図では、ステーションのハード量を基板 2 枚と仮定して示してある。

図 3.8 コストパフォーマンスの比較

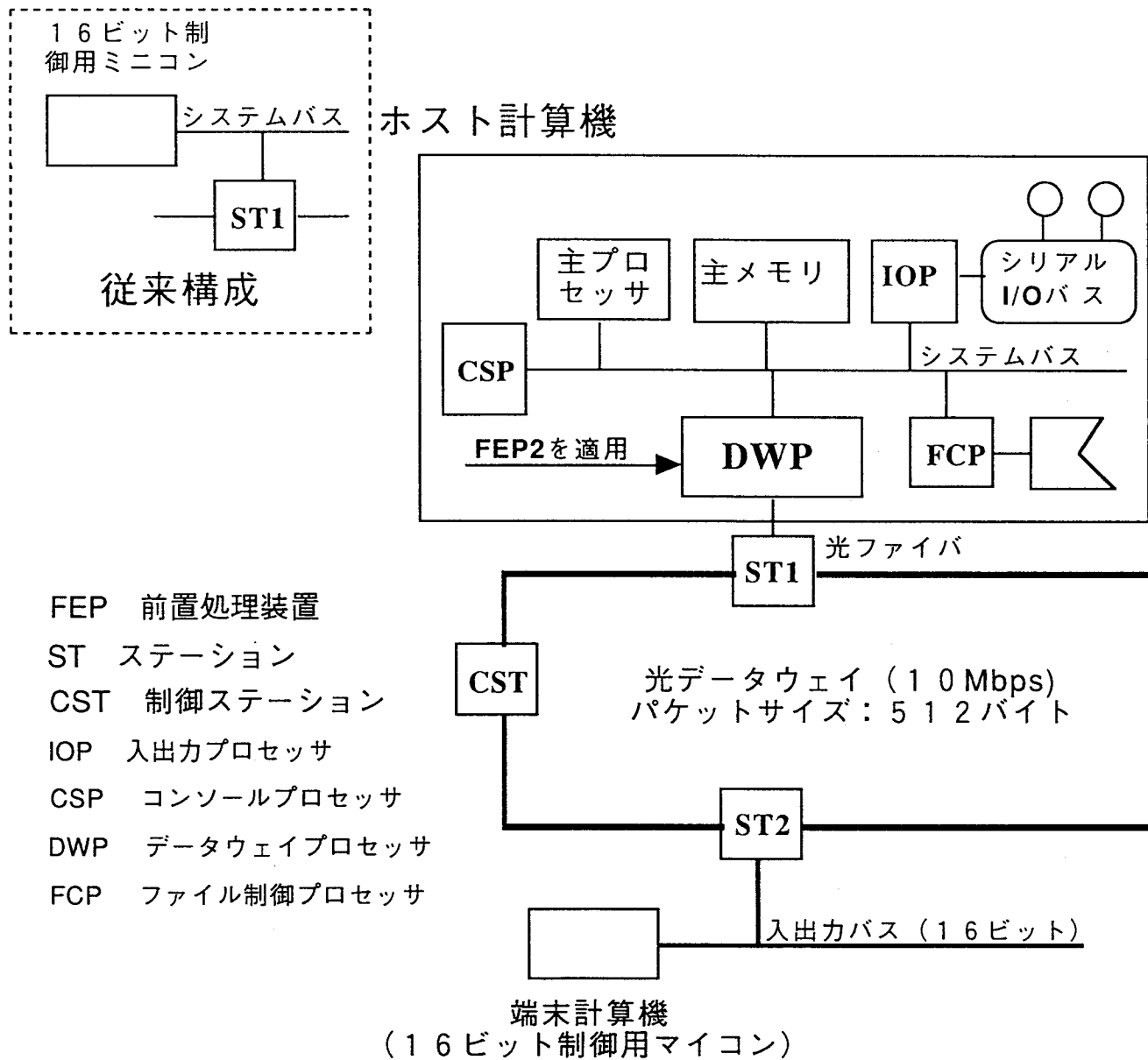
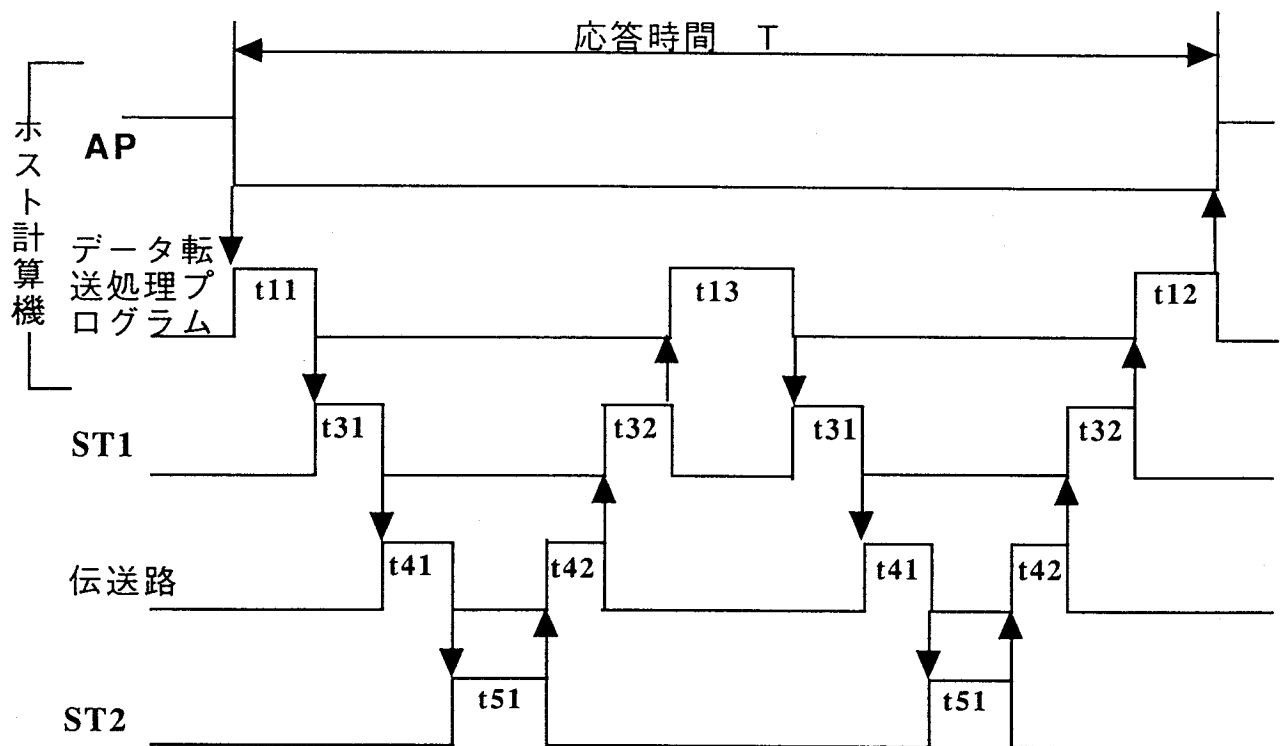


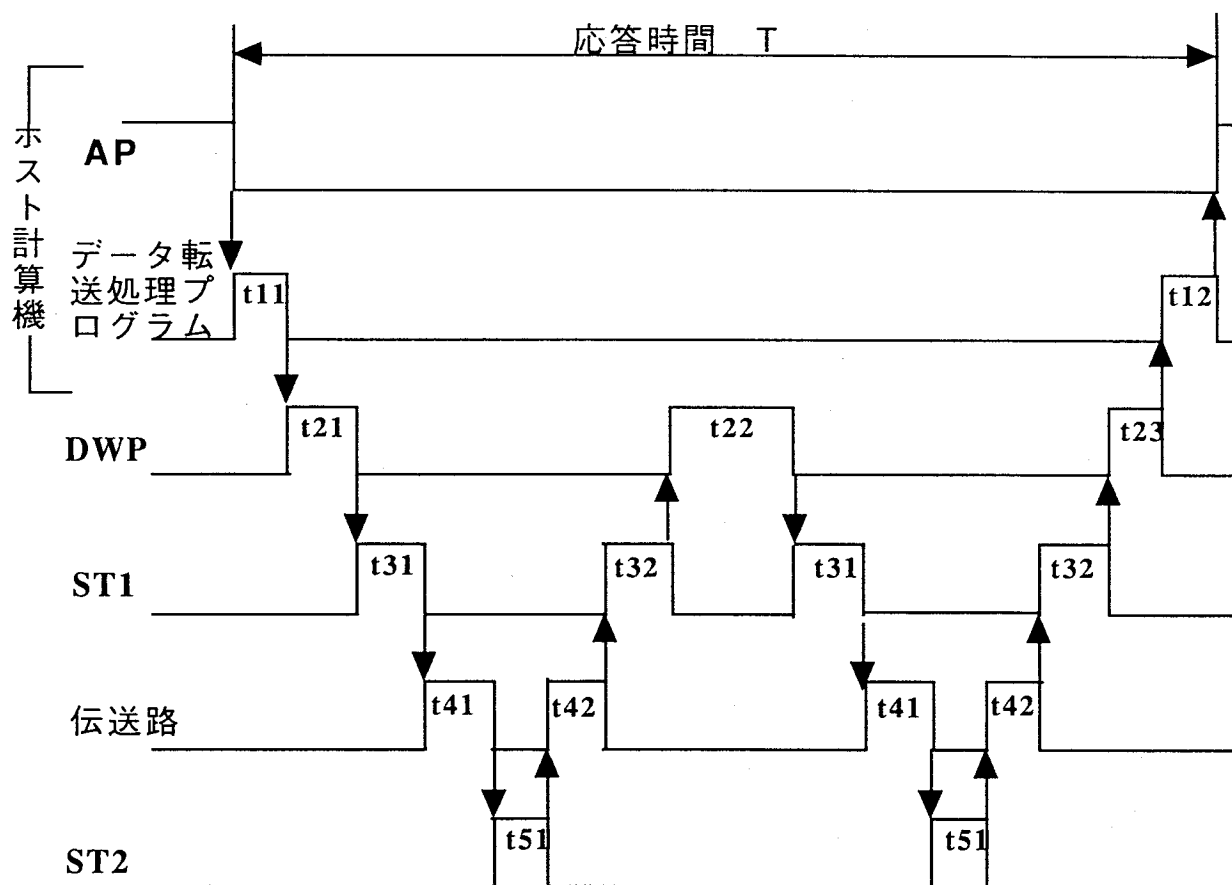
図 3.9 適用評価システムの構成



t_{ij} 処理時間（詳細は、本文参照）

（2 セグメントの例、セグメント数はデータ長により変わる）

図 3.1 0 適用評価モデルタイムチャート
（従来システム）



DWP データウェイプロセッサ

t_{ij} 処理時間（詳細は、本文参照）

（2 セグメントの例、セグメント数はデータ長により変わる）

図 3.1 1 適用評価モデルタイムチャート
(FEP2適用システム)

表 3. 4 適用評価モデルで用いた処理時間

t_{ij}	従 来 システム	F E P II 適用システム	処 理 内 容
t_{11}	1 0	1. 8	ユーザプログラムからの送信要求を受けステーション (従来システム) / DWP (FEP II適用システム) を起動するまでの時間
t_{12}	6. 9	1. 8	ステーション / DWP からの終了通知を受け、ユーザ プログラムを再起動するまでの時間
t_{13}	1 5	—	従来システムにおいて、応答パケットを受け、次の セグメントをステーションに送信するまでの時間
t_{21}	—	4. 8 *	ホスト計算機から起動され、最初のセグメントを ステーションに送信するまでの時間
t_{22}	—	5. 9 *	端末計算機からの応答を受けて、次のセグメントを ステーションに送信するまでの時間
t_{23}	—	2. 2	端末計算機からの最終セグメントに対する応答を受 けて、ホスト計算機に終了を通知するまでの時間
t_{31}	2. 2	2. 2	1 パケットの送信に要するステーションの処理時間
t_{32}	0	0	ステーションにおける応答パケット受信処理時間
t_{41}	1. 0 *	1. 0 *	1 メッセージパケット (最大 5 1 2 バイト) の 伝送時間
t_{42}	0	0	応答パケット (約 6 0 バイト) の伝送時間
t_{51}	0	0	受信ステーション側でメッセージパケットを受信して、 応答を返すまでの時間

* データ長により異なる。この数値は、5 1 2 バイトのときのもの。

各処理時間 t_{ij} は、1 パケット (5 1 2 バイト) の伝送時間 (t_{41}) を単位として表している。

転送をすべて終了し、再びユーザプログラムに制御が戻るまでの時間（図 3.9 の T）

- ・データ転送スループット：単位時間に、ユーザプログラムから DWP を経て、伝送路へ送出されるデータ量（送信要求は、複数発行されるものとする）
- ・プロセッサ利用率：主プロセッサの全稼働時間のうち、データ転送処理に費やされた時間の割合

（3）シミュレーションの条件

メッセージ送信要求は、ポアッソン到着とする。メッセージの長さは、平均 2000 バイトの指数分布とする。相手計算機から応答が返ってくるまでの時間（ t_{32} 、 t_{42} 、 t_{51} ）は、0 とする。

3. 4. 3 結果

表 3. 4 の処理時間 t_{ij} を用いて GPSS によるシミュレーションを行い、図 3. 12、図 3. 13 に示す結果を得た。スループットは伝送速度との比で表して、最大 0. 14 である（従来比約 3 倍）。応答時間は、メッセージ伝送時間（メッセージ長／伝送速度）との比で表して 10 である（従来比約 1／2）。主プロセッサの利用率は、15% 以下である。

3. 5 結言

高速パケットネットワークによって複数の計算機を接続した計算機制御システムを対象に、計算機間のデータ転送スループット、応答時間、計算機負荷などの性能を向上する方式を検討し、コマンド多重処理形トランスポートプロセッサ（前置処理装置：FEP）を提案した。提案した FEP は、1 台の高速マイクロコンピュータで多数のコマンドを多重に処理するので、少ないハードウェアで、高い性能を得ることが期待できる。

提案方式の効果は、32 ビットスーパーミニコンと高速データウェイとの接続に適用し、シミュレーションにより評価した。評価の結果、提案した FEP 構成方式は、下記の効果があることが明らかになった。

- （1）ハード量は、基板 2 枚（第 2 章で提案したマルチプロセッサ方式の 2／3）で済む
- （2）性能は、データウェイの伝送速度との比に換算して約 0. 14（第 2 章で提案したマルチプロセッサ方式のプロセッサ 2 台のときの性能に相当）である

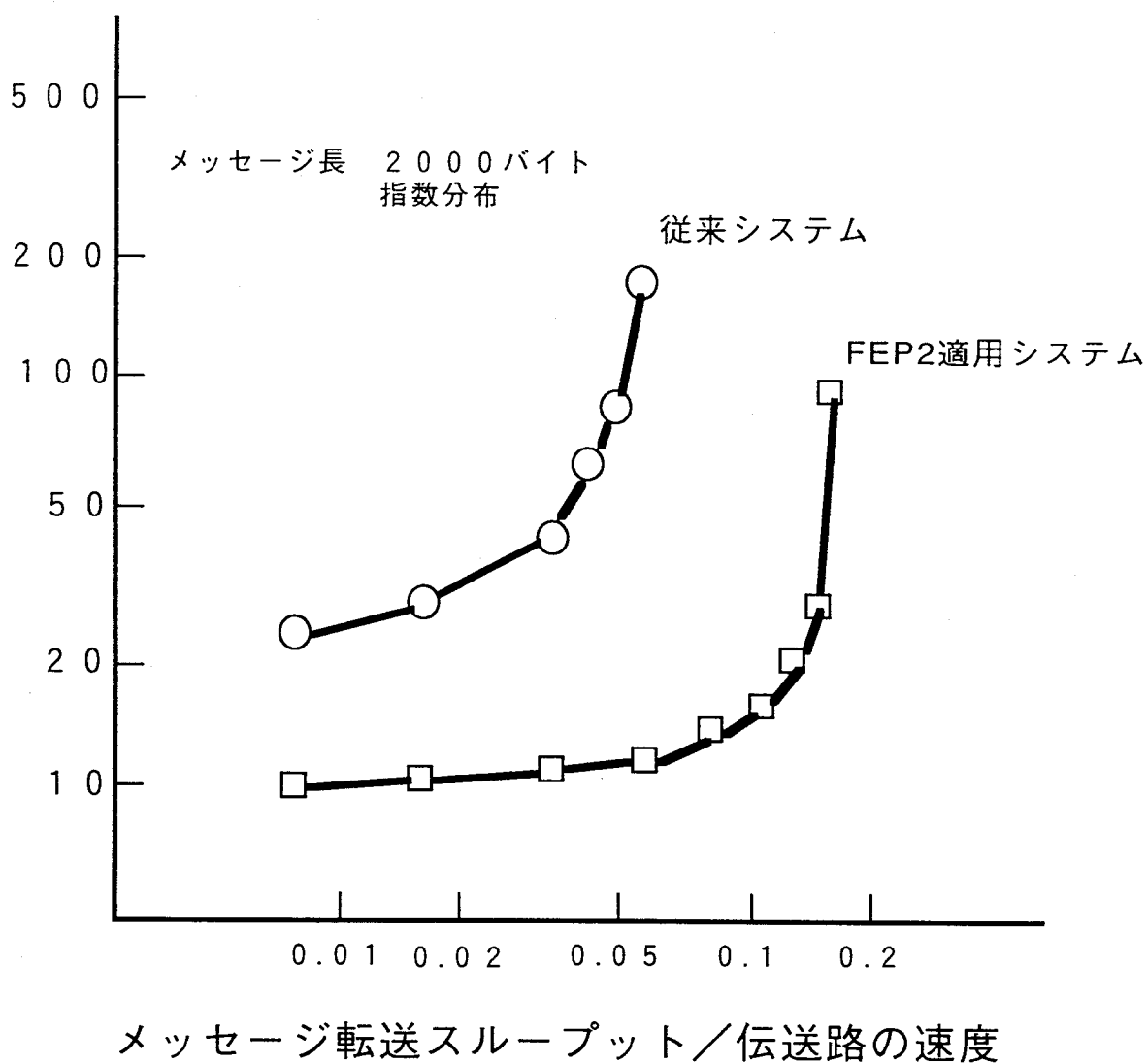


図 3.1 2 メッセージ転送スループット
(シミュレーション)

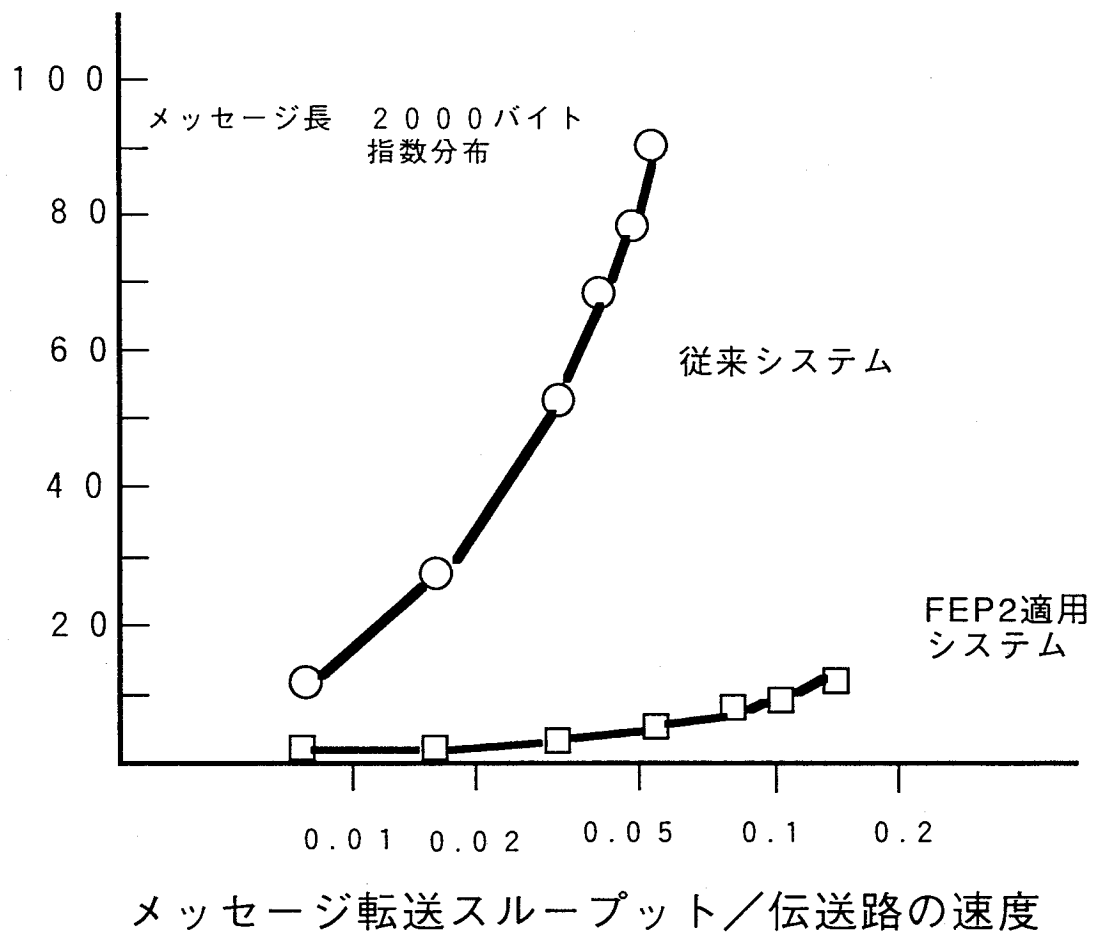


図 3.1 3 主プロセッサ利用率（シミュレーション）

- (3) F E Pを適用した32ビットスーパーミニコンは、従来の16ビット機種に比べ、平均2000バイトのメッセージ転送性能において、スループットで約3倍、応答時間で約1/2の性能向上効果がある

第4章 プロトコル処理ハード化による高速化の提案

4.1 緒言

計算機におけるパケット単位の処理を高速化する方式として、専用通信プロセッサを計算機と伝送路との間に設ける方式を提案し、高速化の効果があることを前章までに述べてきた。近年、LANの伝送速度は、光ファイバ通信技術の進展により、100Mbpsへと更に高速化される方向にある。また、プロトコルの標準化が進み、OSIが広く普及しようとしている。この結果、たとえばLANとして伝送速度100MbpsのFDDI[9]を使い、プロトコルとしてOSI[40]を用いるようなオープンなネットワーク環境では、OSIが従来の制御専用プロトコルに比べて複雑なため、データ転送処理を通信プロセッサにオフロードするだけでは十分な性能を確保できない恐れがある。このため、従来通信プロセッサのプログラムによって行われていたデータ転送処理のうち、プロトコルに係る処理をハードウェアで実行することを提案する。

4.2 プロトコル処理高速化の課題

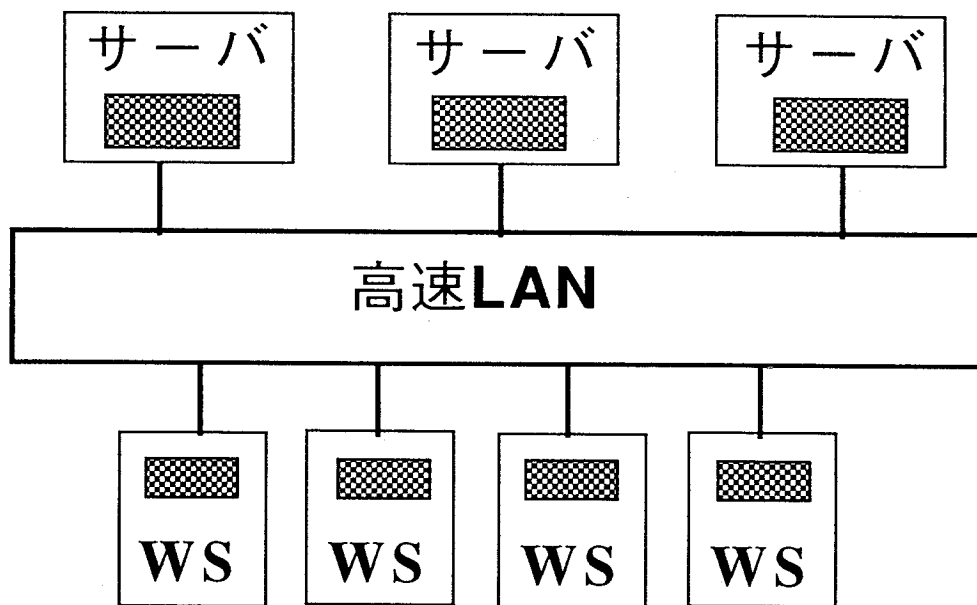
4.2.1 OSIプロトコル

OSIは、7層構造のアーキテクチャを持っている[40]。各層には、多くのプロトコル種別（コネクションオリエンテッド／コネクションレス）、プロトコルオプション／プロトコルクラスが定義されている。本章では、第2層から第4層のプロトコルを高速に実行することを考える。この理由は、図4.1に示すような分散ネットワークシステムにおいて、これらの層がLANアダプタで実行されるからである。プロトコルスタックもLAN環境を考慮し、第2層にLLCクラス1[41]、第3層にCLNP[42]、第4層にTP4[43][44]を選定している。

4.2.2 プロトコル処理高速化の課題

OSIの高速実行には、2つのアプローチがある。第1のアプローチは、RISC技術に基づく高速マイクロプロセッサを用いる方法であり、第2のアプローチは、VLSI技術に基づく専用のプロトコル処理ハードウェアを開発する方法である[38][39]。

前者は、多様なプロトコル／プロトコルオプションに対する柔軟性に利点がある。



 LANアダプタ

WS ワークステーション

図 4.1 高速**LAN**を用いた分散システムの構成例

あるが、ハードウェア規模がLANアダプタに実装するには大きくなりすぎるという欠点がある。後者は、高速化と低コスト化に有利である反面、柔軟性に欠ける。本論文では、後者のアプローチを採る。その理由は、LANアダプタに適用するには、ハードウェアサイズとコストが重要な要件であるからである。

OSIを実行する専用ハードウェアを開発するには、下記の問題を克服する必要がある。

(1) 各層のプロトコル処理が複雑

OSIのトランスポート層クラス4は、エンドツーエンドのトランスポートコネクションにおけるエラー回復責任を持っており、プロトコルが複雑になっているため、ハードウェアによる実行は困難である。多くのパラメータや付加機能について、コネクション設定時に通信する相手と取決めを行う必要がある。またデータ転送フェーズでは、ウィンドウフロー制御、シーケンス番号制御、セグメンティング/リアセンブリなどを実行する必要がある。OSIのプロトコルヘッダは、可変長でビット毎に意味付けされており、4バイトバウンダリに整頓されていないため、ハードウェアによる処理を困難にしている。

(2) 層間のインタフェースオーバーヘッド

OSIは7層構造のため、層間のインタフェース制御オーバーヘッド（たとえば、データの移動、パラメータの受渡し、プロトコルヘッダの受渡しなど）が発生する。したがって、多層構造のプロトコル処理に適した処理方式が必要である。

(3) タイマ制御が複雑

各トランスポートコネクションに3種類のタイマが使われる。したがって、同時処理可能なコネクション数を128～256に設定しようとする、384～768のタイマ制御機構が必要になる。

4. 2. 3 高速化の方策

OSIのハード化にまつわる前記課題を克服するため、以下の方針を設定した。

(1) 正常データ転送処理のみ専用ハードウェア化

OSIの全階層を実行する専用ハードウェアができれば、高速化には有効であるが、実際には柔軟性に乏しく、ハード量が非現実的なほど大きくなると予

想される。そこで、本論文では下記の点を考慮して選定された機能のみをハード化することにした。

- ・プロトコル処理においては、正常データ転送処理の高速化ニーズがコネクションの設定／開放処理より強い。
- ・プロトコルにおけるプロファイルの違いや各層のプロトコルの特徴は、コネクション設定時やデータ転送の異常処理に多く現れる。
- ・伝送路への光ファイバの適用、ディジタル化などにより、伝送路のビット誤り率は、低減化される方向にある。つまり、データ転送フェーズでの異常処理の頻度は少なくなる傾向にある。

そこで、プロトコル処理を、①正常データ転送、②異常処理、③コネクション設定／開放処理、の3つの部分に分け、正常データ転送処理のみを専用ハードウェアで高速実行し、残りの部分は汎用のマイクロプロセッサで処理する。

(2) 各層対応に専用ハードウェアを設置

O S I の各層におけるクラス／オプションを組み合わせて7層全体でセットにしたものは機能標準と呼ばれる。層毎にプロトコル処理を実行する専用ハードウェアを開発することによって、多様な機能標準にも専用ハードウェアの組合せで柔軟に対応できるようにした。

(3) 多層プロトコルの処理を複数の専用ハードウェアでパイプライン処理

層間のインタフェースオーバーヘッドを削減するため、F I F O によるハードウェアインタフェース機構を導入する。このようにして、F I F O で接続された各階層の専用ハードウェアがパイプラインで動作する。

これらの方策によって、下記の効果が期待できる。

- ・とりわけ高速化が要求される正常データ転送処理は、専用ハードウェアによって高速に処理できる。
- ・コネクション設定／開放制御や異常処理は、汎用マイクロプロセッサにより柔軟に処理できる。
- ・ハードウェア化を正常データ転送処理に限定することによって、専用ハードウェアの規模を削減でき、O S I の各層を1チップのV L S I にできる。

4. 3 プロトコル処理ハード化方式の提案

4. 3. 1 要求条件

(1) 高速

OSIのような階層型の通信プロトコルを前提とすると、100Mbpsから1Gbpsの伝送速度の向上に対応できる高速処理が求められる。

(2) コンパクトなハードウェア

LANアダプタに適用できるようなコンパクトなハードウェアが必要である。

(3) 融通性

OSIの4層以下に適用することを考えると、各層に定義されているプロトコルクラスや各種のオプションに柔軟に対応できる融通性が求められる。

4. 3. 2 構成方式の検討

高性能プロトコル処理プロセッサの構成方式を検討する。

(1) 各層に対応する専用ハードウェア

各層のプロトコル処理は、独立に並行して処理できるので、各層対応に専用ハードウェア(FEU: Front End Unit)を設けた。プロトコル処理をFEUにより、並列処理するかパイプライン処理するかは、大きな方式上の選択になる。

- ・並列処理方式：1フレームの処理を複数のFEUが同時に処理する方式である。下位の層でエラーが起きると、既に実行した他の層の処理をバックトラックする必要がある。
- ・パイプライン処理：1フレームの処理は、同時には1つのFEUのみが処理する。フレームは、次々とFEUをリレーされる方式である。下位層でエラーがあれば、そのフレームは次の層のFEUにはリレーされない。

2つの方式とも、スループットに大きな差は無いと考えられる。1つのフレームに着目したときの遅延時間は、並列処理方式がパイプライン処理方式に比べて、 $1/n$ (n はパイプラインの段数で、各段の処理時間が等しいとしたとき)にできる。本論文では、制御が簡単なパイプライン方式を採用した。

(2) バックエンドマイクロプロセッサ

専用ハードウェアはデータ転送の正常処理のみ実行するので、残りの処理(コククシヨンの設定/開放、データ転送の異常処理など)は、汎用のマイクロプロセッサ(BEU: Back End Unit)で処理する。高性能化する上で、送

受信フレームの分配方式が重要である。

- ・ B E Uで分配：一度フレームをB E Uで受けて分析し、正常データ転送処理でF E Uにより処理できると判定したもののみをF E Uに渡す方式。
- ・ F E Uで分配：まずF E Uにフレームを受け、自分で処理できないものをB E Uに渡す方式。

B E Uは汎用マイクロプロセッサであるので、判定に時間がかかる。一方、F E Uは、ハードウェアで構成するので判定処理が高速に行える。本論文では、F E Uにより判定する方式を採用した。

(3) ユーザデータ転送処理とプロトコルヘッダ処理の分離

フレームは、一般にプロトコルヘッダ部とユーザデータ部とから構成されている。フレームは、一つの層から隣接する次の層へ渡される。層間のインタフェースオーバーヘッドを削減することは、高速化にとって重要である。提案方式では、下記の方法を採った。

- ・ 各ユニットからアクセス可能な共通メモリを用意する。
- ・ 送受信フレームは、この共通メモリに一度だけバッファする。層間では、フレームを移動させない。
- ・ 各ユニットは、この共通メモリにアクセスすることによって、プロトコル処理を行う。
- ・ F E Uは、フレームのプロトコルヘッダ部のみを処理し、ユーザデータ部には関与しない。
- ・ 層間のインタフェースは、F E U間に設けたF I F O (First In First Out) 回路を経由して行う。
- ・ F E UとB E Uの間もF I F O回路で連絡する。

(4) バウンダリ制御回路

一般にプロトコルヘッダのフォーマットは、4 バイト境界を意識して設計されていないため、共通メモリに蓄えられたフレームのプロトコルヘッダは、4 バイト境界に整頓されていない。それゆえ、もし共通メモリにある情報に4 バイト境界を越えてアクセスすると、余分のメモリサイクルを要する。プロトコルヘッダにアクセスするときの余分のメモリサイクルを不要にするため、バウンダリ制御回路を設けた。この回路を用いることによって、1～4 バイトのデータであれば、4 バイト境界に整頓されていなくても、1メモリサイクルでアクセスできる。

(5) プロトコルヘッダ事前準備

プロトコルヘッダの情報には、送信相手の識別子、順序番号、長さ、コネクション識別子、フロー制御などがある。このうち、トランスポートのコネクションが設定された後は、いつも固定情報で済むものも多い。そこで、トランスポートコネクション対応に送信バッファを割り当て、コネクション設定後これらの固定情報を予めバッファに固定的にセットしておく方式とした。こうすることによって、データ転送フェーズでは順序番号や長さなどフレーム毎に変化する情報だけセットすれば済み、高性能化できる。

以上の検討結果を踏まえて提案した高性能プロトコルプロセッサのハードウェア構成を図4. 2に示す。

4. 3. 3 プロトタイプの開発

O S Iの2層から4層を実行するための高性能プロトコルプロセッサのプロトタイプを開発した。このプロトタイプの仕様を表4. 1に示す。このプロセッサの詳細なプロトコルサポート範囲を表4. 2 (C L N P)と表4. 3 (T P 4)に示す。

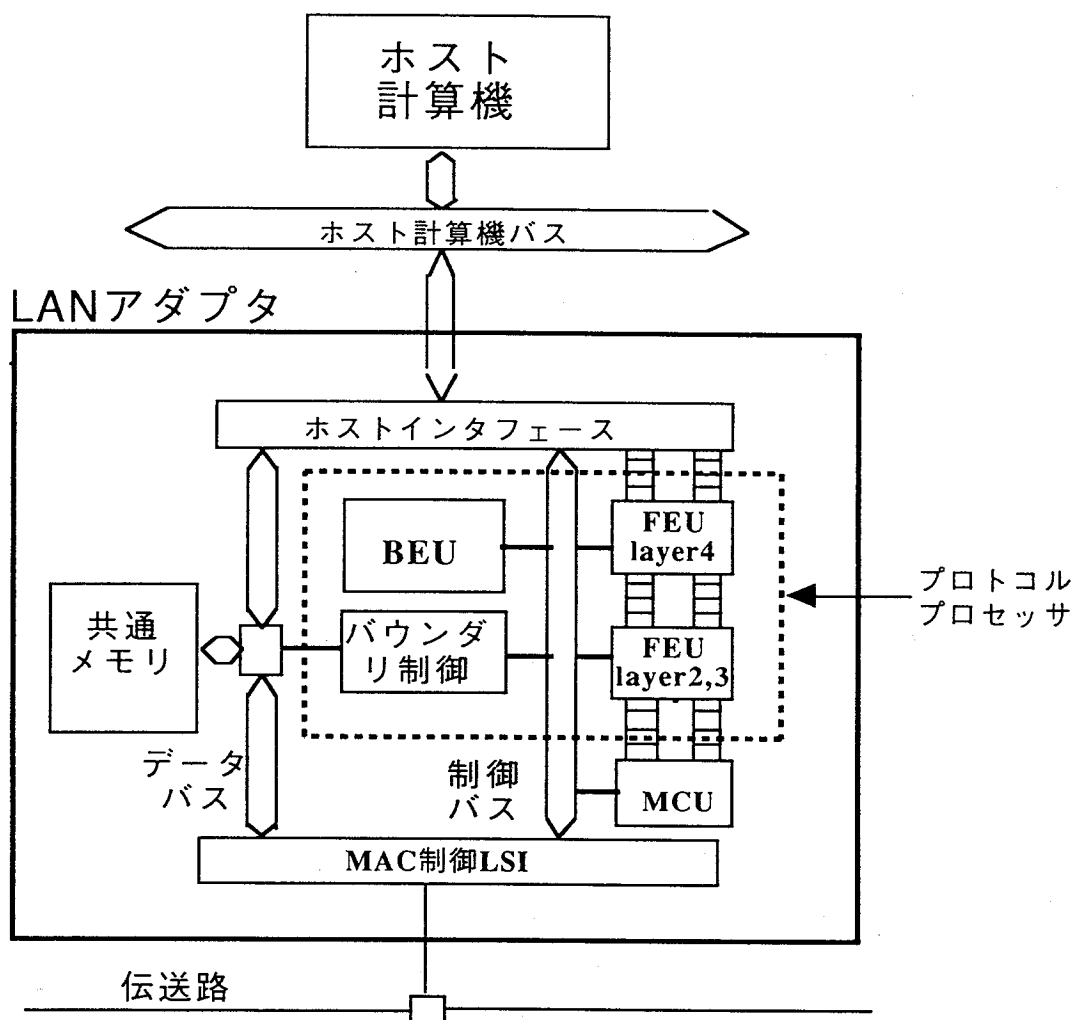
この実験モデルでは、M C 6 8 0 2 0 (2 5 M H z)をB E Uとして用いている。各F E Uの内部構造を図4. 3に示す。F E Uは、A L U、F I F O、D M A回路、ヘッダの符号化/複合化回路、コネクションの状態を保持するためのレジスタファイルおよび高速シーケンサから構成している。この汎用シーケンサは、マイクロプログラム機能を持ち、5 0 n s /命令の高速で動作する。このマイクロプログラムの変更により、多様なプロトコルに対応できる。

4. 3. 4 L A Nアダプタへの適用

図4. 2は、提案しているプロトコルプロセッサのL A Nアダプタへの適用方法も示している。L A Nアダプタは、プロトコルの1層から4層を実行する。アダプタは、ホストインタフェース、共通メモリ、M A C - L S I、M C U (M A C - L S I制御ユニット)、プロトコルプロセッサから構成している。

伝送路からM A C - L S Iを介して受信したフレームは、共通メモリにストアされ、プロトコルプロセッサが処理する。処理されたフレームは、ホストインタフェースを経由してホスト計算機の主メモリに転送される。送信の場合は、この逆に処理が行われる。

提案したプロトコルプロセッサは、M C Uを変更することによって、容易に各種M A C - L S Iに対応することができる。



FIFO

BEU:Back End Unit
FEU:Front End Unit

図4.2 高速プロトコルプロセッサの構成

表4. 1 プロトタイプ高速プロトコルプロセッサの仕様

項 目		仕 様	備 考
プロトコル	第4層	I S O 8 0 7 3	T P 4
	第3層	I S O 8 4 7 3	C L N P
	第2層	I S O 8 8 0 2 / 2	L L C クラス1
B E U (Back End Unit)		M C 6 8 0 2 0	2 5 M H z
共通メモリ		2 5 6 k B	バッファメモリ
内部バス		I E E E 1 0 1 4	3 2 ビット
バウンダリ制御回路		4バイトアクセス/1メモリサイクル	
F E U	制御方式	マイクロプログラム	5 1 2 ステップ
	シーケンサ	5 0 n s / 命令	
	演算器 (A L U)	1 6 ビット	
	層間インタフェース	F I F O	1 0 2 4 エントリ
	B E U インタフェース	F I F O	1 0 2 4 エントリ
	D M A 性能	4 M B / s	ヘッダの読み／書き
	ハードウェアサイズ	3 0 0 × 3 2 0 m m	シングルボード

表 4. 2 プロトタイププロトコルプロセッサの C L N P サポート範囲

項番	項 目	サ ポ ー ト
1	P D U の作成／分解	○
2	ヘッダフォーマットの分析	○
3	P D U の寿命制御機能	○
4	P D U のルート機能	×
5	P D U の分割／組立て	×
6	P D U の廃棄	○
7	エラー報告	○
8	P D U ヘッダエラーの検出	○
9	パディング	○
1 0	セキュリティ	×
1 1	ソースルーティング	×
1 2	ルートの記録	×
1 3	サービス品質の維持	×
1 4	優先データ	×

表4. 3 プロトタイププロトコルプロセッサのTP4サポート範囲

項番	プロトコル機構	サポート
1	ネットワークコネクションへの割当て	×
2	TPDUの転送	○
3	分割と組立て	×
4	連結と分離	×
5	コネクションの確立	○
6	コネクション確立の拒否	○
7	正常解放	○
8	TPDUのトランスポートコネクションへの関連付け	○
9	DT TPDUの番号付け	○
10	優先データ転送	×
11	TPDUの確認までの保持	○
12	多重化と逆多重化	×
13	明示的フロー制御	○
14	検査和（サムチェック）	×
15	凍結レファランス	○
16	タイムアウト時の再送	○
17	再順序付け	×
18	無活動監視制御	○
19	プロトコル誤りの扱い	○
20	分流と合流	×

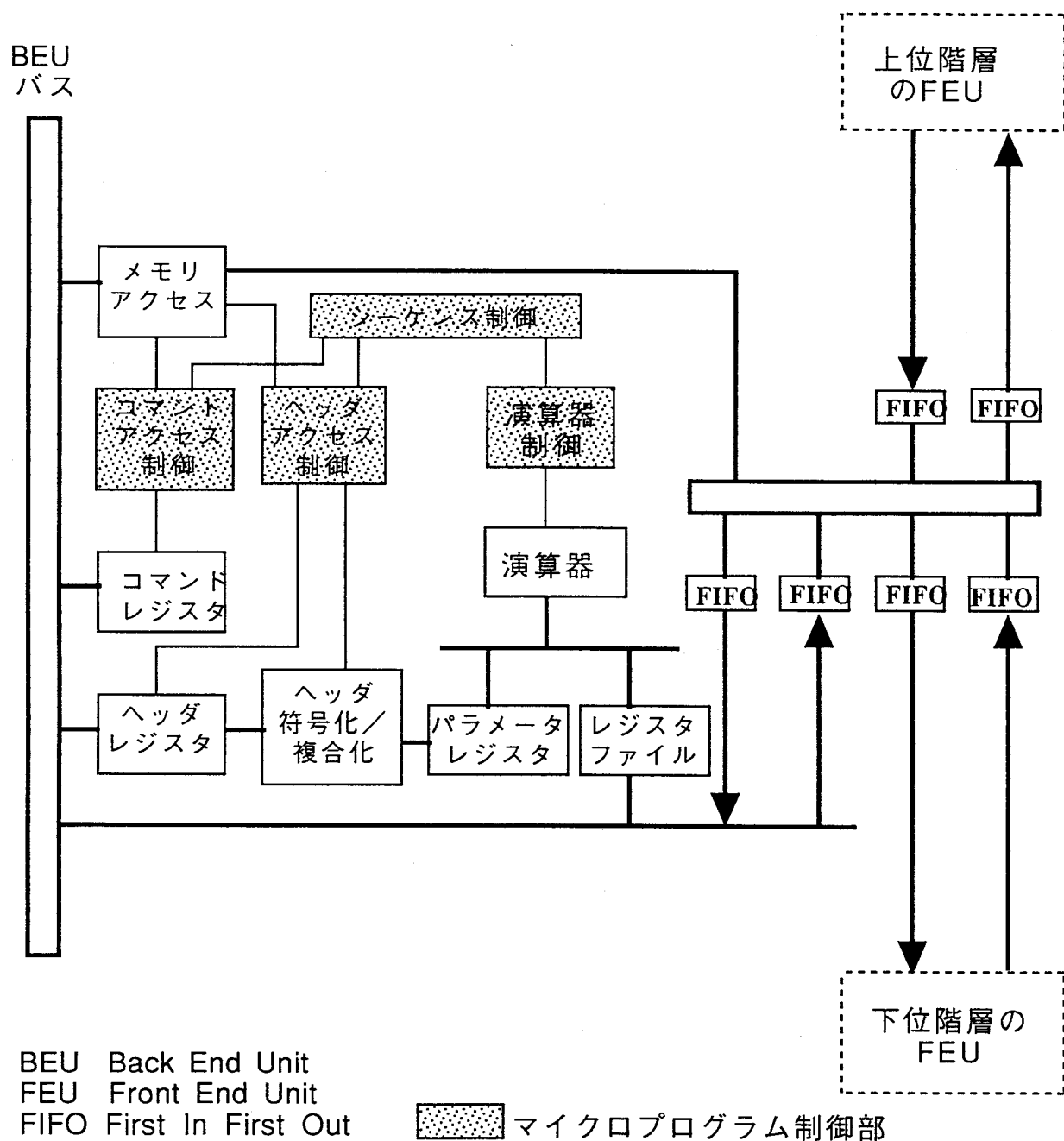


図4.3 FEUの内部構造

4. 4 実験的評価

4. 4. 1 評価の目的

このプロセッサで得られる処理速度の改善度合いを測定するため、第2層から第4層の処理時間を従来方式と実験的に比較した。さらに、この結果にもとづき、この高速プロトコルプロセッサをLANアダプタに適用した場合の処理時間を推定した。

4. 4. 2 評価の方法

実験的な高速プロセッサの性能は、次のように測定した。FEUに関する性能は実際に測定器により計測し、BEUのソフトウェアに関する性能はダイナミックステップを積算し、これに平均命令実行時間を乗ずることで求めた。MAC層を実装しないで擬似MACプログラムをBEU上に作成し、内部ループを形成して処理時間を測定した。

比較のために、正常プロトコル処理時間と擬似MACプログラム処理時間を、MC68020(20MHz)を用いたワークステーション上で測定した。比較に用いたデータ転送シーケンスを図4.4に示す。

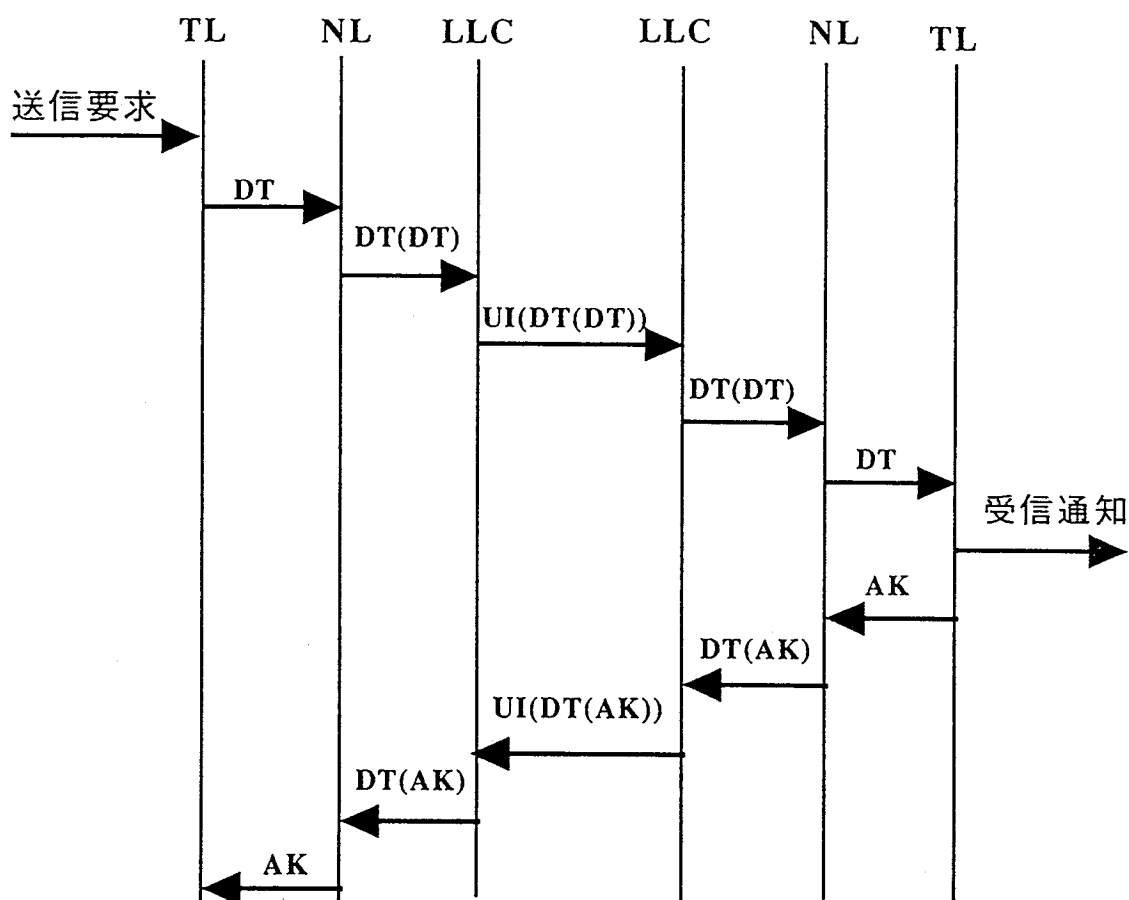
4. 4. 3 結果

比較結果を表4.4に示す。高速プロトコルプロセッサを適用した実験システムは従来方式による場合に比べ、正常データ転送において処理性能が7～12倍向上した。

4. 4. 4 LANアダプタへの適用効果

送信／受信に対するLANアダプタの総処理時間を各部の処理時間から計算した。得られた処理時間をFDDI(100Mbps)上のフレーム伝送時間で正規化し、DMAスループット4、8、16MB/sをパラメータとして表4.5(フレームサイズが4000バイトの場合)と表4.6(フレームサイズが300バイトの場合)に示した。

これらの結果は、LANアダプタにおけるプロトコル処理に起因する性能上のボトルネックが提案した高速プロトコルプロセッサによって解消できることを示している。



LLC 論理リンク制御層
 NL ネットワーク層
 TL トランスポート層

図 4.4 データ転送シーケンス

表 4. 4 プロトコル処理時間の比較

項 番	処理内容	従来方式	プロトタイプとの比
1	D Tフレームの送信	1	1 / 1 4
2	A Kフレームの受信	1	1 / 1 1
3	D T送信 & A K受信	1	1 / 7
4	送信側 (項番 1 + 項番 2)	1	1 / 1 2
5	受信側 (項番 3)	1	1 / 7

表 4. 5 F D D I 用 L A N アダプタの推定性能

(フレームサイズ=4 0 0 0 バイト)

(処理時間は、フレーム伝送時間で正規化した値)

処 理 内 容	送 信 側			受 信 側		
DMAスループット (バイト/秒)	4 M	8 M	1 6 M	4 M	8 M	1 6 M
プロトコルプロセッサ	0. 6	0. 6	0. 6	0. 5	0. 5	0. 5
M A C 制御ユニット (M C U)	0. 7	0. 7	0. 7	0. 7	0. 7	0. 7
ホストインタフェース	3. 8	2. 2	1. 4	4. 1	2. 5	1. 8

送信側とは表 4. 4 の項番 4 を、受信側とは表 4. 4 の項番 5 の処理を意味する。

表 4. 6 F D D I 用 L A N アダプタの推定性能

(フレームサイズ=3 0 0 バイト)

(処理時間は、フレーム伝送時間で正規化した値)

処 理 内 容	送 信 側			受 信 側		
DMAスループット (バイト/秒)	4 M	8 M	1 6 M	4 M	8 M	1 6 M
プロトコルプロセッサ	8. 0	8. 0	8. 0	7. 1	7. 1	7. 1
M A C 制御ユニット (M C U)	9. 6	9. 6	9. 6	9. 6	9. 6	9. 6
ホストインタフェース	1 2	1 0	9. 5	1 6	1 5	1 4

送信側とは表 4. 4 の項番 4 を、受信側とは表 4. 4 の項番 5 の処理を意味する。

4. 5 結言

本論文では、LANアダプタに適した高速OSIプロセッサを提案した。この提案の効果を定量的に評価するために、OSIの第2層から第4層のプロトコル処理を実行できるプロトタイプを作成し、その性能を測定した。

実験システムにおける正常データ転送のプロトコル処理時間は、従来方式による場合に比べて、送信処理で1/12受信処理で1/7にできることを明らかにした。

これらの結果から、提案した高速プロトコルプロセッサを適用することによって、LANアダプタの性能を改善できることを結論づけた。

第 5 章 結 論

本論文は、機能階層形多重通信プロセッサによる分散ネットワークの高性能化について論じた。得られた結論を下記にまとめる。

高速パケット通信ネットワークによって複数の計算機を接続した分散ネットワークを対象に、計算機間のデータ転送スループット、応答時間、計算機負荷などの性能を向上する方式を検討し、通信プロセッサを第1層と第2層のプロトコルを実行するデータリンクプロセッサと第3層と第4層のプロトコルを実行するトランスポートプロセッサとから構成する機能階層形多重通信プロセッサ方式の提案と性能向上度の評価を行った。

提案した方式は、ホスト計算機と伝送路との間に、データ転送のための通信プロトコルを専用に処理する機能階層形多重通信プロセッサを置くことにより、従来は、ホスト計算機で実行していた通信プロトコル処理の大部分を通信プロセッサで高速に実行させるようにした点に特徴がある。この方式により、通信プロトコル処理の高速化とホスト計算機の負荷削減を図ることができる事を示した。

具体的には、階層化された通信プロトコルのうち、第1層から第4層までの処理を、2階層に分け、機能階層毎に高速ビットスライスマイクロプロセッサ（語長32ビットのマイクロ命令を200nsで実行する）でパイプライン処理する専用の多重通信プロセッサを開発してその効果を明らかにした。トランスポートプロセッサの構成方法としては、マルチプロセッサ・シングルコマンド処理方式とシングルプロセッサ・マルチコマンド処理方式の2つの方式を提案し検討した。

マルチプロセッサ・シングルコマンド処理方式については、マルチマイクロコンピュータ構成のトランスポートプロセッサ（前置処理装置：FEP）とデータリンクプロセッサとから構成する機能階層形多重通信プロセッサ方式を提案した。提案方式による性能向上度を把握するため、マイコン2台から成るFEPを試作し、伝送速度10Mbpsの高速パケット通信ネットワークに適用した。実験の結果、計算機間データ転送性能は、FEPが無いときに比べて、次のように改善されることを明らかにした。

- (1) データ転送スループットが3倍になる（データ長が4kBのとき）。
- (2) 応答時間が1/2になる（データ長が60Bのとき）。
- (3) 計算機負荷が40%（データ長が8Bのとき）～80%（データ長が8kBのとき）削減できる。

シングルプロセッサ・マルチコマンド処理方式については、F E P の構成方式としてコマンド多重処理形トランスポートプロセッサを提案した。提案した F E P は、1 台の高速マイクロコンピュータで多数のコマンドを多重に処理するので、少ないハードウェアで、高い性能を得ることが期待できる。提案方式の効果は、3 2 ビットスーパーミニコンと高速データウェイとの接続に適用し、シミュレーションにより評価した。提案した F E P 構成方式は、下記の効果があることを明らかにした。

- (1) ハード量を、3 0 . 5 c m × 4 2 . 5 c m の基板 2 枚に収めることができる。
- (2) 性能は、データウェイの伝送速度との比に換算して約 0 . 1 4 である。
- (3) F E P を適用した 3 2 ビットスーパーミニコンは、従来の 1 6 ビット機種に比べ、平均 2 0 0 0 バイトのメッセージ転送性能において、スループットで約 3 倍、応答時間で約 1 / 2 の性能向上効果がある。

これらの結果より、機能階層形多重通信プロセッサ方式が通信プロトコル処理高速化の要請に応えうることを結論づけた。

提案方式は、計算機制御システムにおいて、高速パケット通信ネットワークとしてデータウェイを用いた場合について論じた。これ以外のシステムでも、高速パケット通信ネットワークと計算機との接続の場合には、提案した機能階層形多重通信プロセッサ方式がデータ転送性能の向上に役立つものと考えられる。

データウェイなどの高速パケット通信ネットワークの伝送速度は、光通信技術の進歩に伴い今後も更に高速化されるもの予想される。この結果、本研究で取り上げた分散ネットワークにおけるデータ転送処理高速化の要求は、ますます増大するものと考えられる。1 0 0 M b p s から 1 G b p s の高速パケット通信ネットワークが実用化される時代にあつては、これらの伝送速度に見合うパケット単位のプロトコル処理時間は、マイクロ秒オーダーになる。従って、今後とも更に高速化の研究を続ける必要がある。

本研究では、独自のプロトコルでシステムを構築してきたが、その後プロトコルの標準化が進み、I S O で規定された O S I プロトコルが次第に普及してきている。また、本研究では、通信プロセッサに数枚のボードを要しているが、最近の半導体技術の進歩は、日進月歩であり、高度な機能を持った各種の V L S I が開発されるようになってきている。このような背景から、1 0 0 M b p s 以上の高速光 L A N を用いた分散ネットワークにおいて、O S I プロトコル

をハードウェアによって高速に処理するVLSIの開発が、今後の重要な課題になると思われる。このような背景から本論文では、単にプロセッサの高性能化だけでなく、プロトコル処理のハード化を考える必要があることを指摘すると共に、プロトコル処理ハード化の課題を整理し、OSIのレイヤ2、レイヤ3、レイヤ4のプロトコルを対象に、プロトコル処理のハード化による高性能化方式を提案し、今後の研究の道標とした。

謝 辞

本論文をまとめるにあたって本論文の細部にわたり検討を加え修正のご指導をいただいた大阪大学基礎工学部情報工学科宮原秀夫教授に心から感謝します。終始ご指導いただいた大阪大学基礎工学部情報工学科嵩忠雄教授、橋本昭洋教授、都倉信樹教授、岡山大学工学部情報工学科岡本卓爾教授に感謝の意を表します。

本研究を進めるに当たり、応用面からの貴重な助言をいただいた（株）日立製作所元大みか工場桑原洋部長（現在同社常務取締役）、平井浩二副部長（現在同社情報通信事業部副事業部長）、同社大みか工場井手寿之副工場長、伏見仁志部長、中西宏明部長、安元精一部長に深く感謝します。本研究の遂行を工場側から支援していただいた大みか工場の堀雄太郎主任技師、林慶治郎主任技師、岡田政和主任技師、溝河貞夫主任技師、元大みか工場の佐藤寛之氏（現在同社情報事業本部主任技師）に感謝します。

本研究の機会を与えていただいた（株）日立製作所システム開発研究所元所長三浦武雄博士（現在同社副社長）、前所長川崎淳博士（現在日立ソフトウェアエンジニアリング（株）専務）、堂免信義所長、大町一彦企画室長に感謝します。本研究遂行のきっかけを与えていただくと共に、その後の研究遂行、論文化、そして本論文の執筆にあたって終始親切なるご指導と督励をいただいた当研究所主管研究員三巻達夫博士に深く感謝します。また、研究途上において筆者の最初の学術論文を書く際に、懇切丁寧に御指導いただいた元当研究所主管研究員大成幹彦博士に感謝します。

本研究の遂行を直接ご指導いただいた当研究所主管研究員桎尾次郎博士、元当研究所平子叔男部長、元当研究所津田順二部長（現在同社システム開発本部副技師長）に心からお礼申し上げます。研究遂行を支援していただいた元当研究所の横田耕治氏、高木悟氏、関高明氏、黒木正彦氏、また当研究所第4部の福澤淳二研究員、松井進研究員、平田哲彦研究員、横山達也氏、水谷美加嬢、また学位論文申請にあたり事務処理面で支援していただいた木下由加里嬢に感謝の意を表します。

以上のほかにも、多くの方々の暖かい励ましと多大な御援助御協力をいただきました。これらの方々を含め、ここに深く感謝の意を表します。

参考文献

- [1] 上谷 晃弘、「データハイウェイの現状と将来」：情報処理、Vol.18、No. 1、pp.76-87、1977
- [2] Y. Matsumoto, O. Sasaki, T. Sumi: " A Distributed Processing System and Its Application to Industrial Control ", Proc. of National Computer Conference '78, pp.1273-1279 ,1978
- [3] Jensen E. D. : " The Honeywell Experimental Distributed Processor - An Overview ", IEEE COMPUTER, 11, 1, pp.28-39 ,1978
- [4] M. Takahashi, M. Yanaka, H. Fushimi, M. Maeda, J. Kashio, M. Mitsuoka: " Optical Fiber Data Freeway System - A Loop Network for Distributed Computer Control ", COMPCON'81, Spring, pp.458-463, 1981
- [5] David D. Clark, Kenneth T. Pogran, David P. Reed: " An Introduction to Local Area Networks ", Proceedings of The IEEE, 66, 11, pp.1497-1516, 1978
- [6] 石坂 充弘、中塚 茂雄、覚埜 高音、市橋 立機、松永 宏：「高速ループシステム（LOOP-3）」、情報処理学会第21回全国大会予講集、pp.662-656、1980
- [7] Robert M. Metcalfe, David R. Boggs: " Ethernet: Distributed Packet Switching for Local Computer Networks", CACM, Vol. 19, No. 7, pp. 395-404, 1976
- [8] M. Stark, A. Kornhauser, and D. Van-Mierop: " A High Functionality VLSI LAN Controller for CSMA/CD Network ", IEEE COMPCON'83 Spring, pp.510-517, 1983
- [9] Floyd E. Ross: " An Overview of FDDI: The Fiber Distributed Data Interface " , IEEE Journal on Selected Areas in Communications, Vol. 7, No. 7, pp. 1043-1051, Sept. 1989
- [10] 八木 駿：「通信制御の動向」、情報処理、Vol. 20、No. 1、pp.3-21、1979
- [11] 橋本 昭洋、山下 正秀：「通信制御プロセッサ」、電子通信学会誌、Vol. 62、No. 11、pp.1296-1303、1979
- [12] F. E. Heart, S. M. Ornstein, W. R. Crowther, W. B. Barker: " A New Minicomputer/Multiprocessor for the ARPA Network ", Proc. of National Computer Conference '73, pp.529-537, 1973
- [13] 福原 美三、小山 謙二：「マルチプロセッサ構成による機能階層形通信プロセッサの検討」、昭和55年度電子通信学会総合全国大会、論文

番号1414、1980

- [14] 森田 和夫、桑原 洋、井手 寿之：「制御用データ伝送システムと周辺装置」、電気学会雑誌、Vol. 96、No. 8、pp.699-704、1976
- [15] 桑原 洋、平子 叔男：「制御用コンピュータのネットワークシステム」電気学会雑誌、Vol. 98、No. 3、pp.199-203、1978
- [16] 桎尾 次郎、寺田 松昭、黒木 正彦、平井 浩二、伏見 仁志、今井 真澄：「H I D I C 80ネットワークシステム(D P C S)の開発」、情報処理学会第17回全国大会、論文番号260、1976
- [17] 桑原 洋、平井 浩二、平子 叔男、寺田 松昭：「分散形制御用計算機システム」、昭和53年電気四学会連合大会、論文番号212、pp.647-650、1978
- [18] 平子 叔男、平井 浩二、寺田 松昭：「制御用分散処理システム」、情報処理、Vol.20、No.4、pp.346-349、1979
- [19] 三巻 達夫、寺田 松昭：「インハウスネットワーク」、計測と制御、Vol.19、No.1、pp.103-109、1980
- [20] M.Terada, J.Kashio, K.Yokota, Y.Hori, H.Fushimi: "A Network Operating System for High Speed Optical Fiber Loop Transmission System", ICC'80, pp.641-646, 1980
- [21] 寺田 松昭、関 高明、桎尾 次郎、堀 雄太郎：「高速パケット伝送路を対象とした通信プロセッサの検討」、情報処理学会第23回(昭和56年後期)全国大会、論文番号5D-7、1981
- [22] 寺田 松昭、関 高明、桎尾 次郎、堀 雄太郎：「高速パケット伝送路用前置処理装置の一構成法」、情報処理学会論文誌、Vol. 23、No.6、pp.707-715、1982
- [23] 寺田 松昭、高木 悟、中西 宏明、伏見 仁志：「ローカルネットワークにおけるコンピュータインタフェース高速化の検討」、情報処理学会第26回(昭和58年前期)全国大会、論文番号5G-8、1983
- [24] 寺田 松昭、高木 悟、安元 精一、中西 宏明：「機能階層形通信プロセッサによる計算機間メッセージ転送スループットの向上方式について」、情報処理学会「ローカルエリアネットワーク」シンポジウム、pp.163-170、1983
- [25] 福澤 淳二、寺田 松昭、高木 悟、溝河 貞夫、末木 雅夫：「マルチ仮想回線動的チャネル割当方式による高性能LANアダプタの一方式」「LAN/マルチメディアの応用と分散処理」シンポジウム、pp.69-76、1984
- [26] 寺田 松昭、高木 悟、桎尾 次郎、安元 精一、伏見 志、中西 宏明：「高速パケット伝送路用コマンド多重処理形前置処理装置」、情報処理学会論文誌、Vol. 26、No. 2、pp.219-227、1985

- [27] 水谷 美加、松井 進、寺田 松昭：「プロトコル処理高速化方式の提案とOSIプロトコルへの適用」、情報処理学会第40回（平成2年前期）全国大会、論文番号5N-3、1990
- [28] 横山 達也、松井 進、平田 哲彦、水谷 美加、寺田 松昭：「通信プロトコル高速処理プロセッサの方式提案」、情報処理学会第41回（平成2年後期）全国大会、論文番号4Q-7、1990
- [29] T.Hirata, S.Matsui, T.Yokoyama, M.Mizutani, M.Terada: "A High Speed Protocol Processor to Boost Gateway Performance", Proc. of IEEE Globecom'90, PP.1426-1430, 1990
- [30] 松井 進、平田 哲彦、横山 達也、水谷 美加、寺田 松昭：「専用ハードウェア化による通信プロトコル処理高速化の一方式」、情報処理学会論文誌、Vol. 32, No.2, pp.272-279, 1991
- [31] 平田 哲彦、横山 達也、水谷 美加、寺田 松昭：「高速LAN用通信制御装置の一構成法」、電子情報通信学会 情報ネットワーク研究会、IN90-98, pp.13-18, 1991
- [32] M.Terada, T.Yokoyama, T.Hirata, S.Matsui: "A High Speed Protocol Processor to Execute OSI", Proc. of IEEE INFOCOM'91, pp.944-949, 1991
- [33] 横山 達也、平田 哲彦、水谷 美加、寺田 松昭：「プロトコル処理専用プロセッサのアーキテクチャに関する一考察」、情報処理学会第42回（平成3年前期）全国大会、論文番号5T-6、1991
- [34] 平田 哲彦、横山 達也、水谷 美加、寺田 松昭：「高速LAN通信制御装置の構成法に関する一考察」、情報処理学会第42回（平成3年前期）全国大会、論文番号5T-7、1991
- [35] 水谷 美加、平田 哲彦、横山 達也、寺田 松昭：「高速LAN通信制御装置の性能に関する一考察」、情報処理学会第42回（平成3年前期）全国大会、論文番号5T-8、1991
- [36] Liba Svobodova: "Measured Performance of Transport Service in LANs", Computer Networks and ISDN Systems, Vol. 18, pp.31-45, 1989/90
- [37] David R. Cheriton and Carey L. Williamson: "VMTP as the Transport Layer for High-Performance Distributed Systems", IEEE Communications Magazine, pp.37-44, June 1989
- [38] Greg Chesson: "Protocol Engine Design", USENIX '87, pp.313-319, 1987
- [39] H. Ichikawa, H. Yamada, T. Akaike, S. Kanno, and M. Aoki: "Protocol Control VLSI for Broadband Packet Communications", Globecom '88, pp.1494-1498, 1988

- [40] ISO8072, Information Processing Systems - Open Systems Inter-connection - Basic Reference Model, 1984
- [41] ISO8802-2, Information Processing Systems - Local Area Networks - Part 2: Logical Link Control, 1987
- [42] ISO8473, Information Processing Systems, Open Systems Inter-connection, Data Communications Protocol for Providing Connection less - Mode Network Service ,1986
- [43] ISO8072, Information Processing Systems - Open Systems Inter-connection -Connection Oriented Transport Service Definition, 1986
- [44] ISO8073, Information Processing Systems - Open Systems Inter-connection - Connection Oriented Transport Protocol Specification, 1986
- [45] Bernd W. Meister: "A Performance Study of ISO Transport Protocol", IEEE Trans. on Computers, Vol. 40, No. 3, pp.253-262, 1991
- [46] 斎藤 忠夫、白畑 厚志:「多層構造プロトコルをもつネットワークの伝送効率の評価」、信学論(D-I), Vol. J72-D-I, No. 7, pp.562-569, 1989
- [47] 稲井 寛、横平 徳美、村田 正幸、宮原 秀夫:「階層型通信プロトコルの性能評価」、電子情報通信学会論文誌 B-I, Vol. J73-B-I, No.5, pp.505-515, 1990
- [48] 斎藤 忠夫、福井 俊之:「多層構造プロトコルをもつネットワークの伝送効率の評価式」、電子情報通信学会、論文誌 D-I, Vol. J73-D-I, 4, pp.424-431, 1990
- [49] 平井 浩二、林 利弘、神内 俊郎、寺田 松昭:「HIDIC 80 マルチシステムをサポートするソフトウェア」、昭和51年電気学会全国大会、論文番号1148、1976
- [50] 横田 耕治、寺田 松昭、堀 雄太郎:「HIDIC 80 ネットワーク・システム(DPCS)におけるリソース排他制御システムについて」昭和54年電気学会全国大会、論文番号1184、1979
- [51] 寺田 松昭、堀 雄太郎、高木 悟:「異機種ネットワークに関する一考察—HIDIC 80 シリーズによるTSS端末エミュレーター—」、昭和54年度 情報処理学会第20回全国大会、論文番号3G-1、1979
- [52] 寺田 松昭、中西 宏明、樫尾 次郎、伏見 仁:「HIDIC シリーズ ネットワークアーキテクチャ」、昭和57年電気学会全国大会、論文番号1244、1982
- [53] 寺田 松昭、樫尾 次郎、安元 精一、中西 宏明:「低コストローカ

- ルエリアネットワーク (μ - Σ NET) の設計思想」、情報処理学会第 28 回 (昭和 59 年前期) 全国大会、論文番号 1D-5、1984
- [54] 寺田 松昭、福澤 淳二、高木 悟、溝河 貞生、大貫 健：「低コストローカルエリアネットワーク (μ - Σ NET) 用通信制御ボード (μ NCB) の開発」、情報処理学会第 28 回 (昭和 59 年前期) 全国大会、論文番号 1D-6、1984
- [55] 寺田 松昭：「ブロードバンド LAN における双方向 CATV」、情報処理学会 ニューメディア/マルチメディアと分散処理－現状と将来の課題－シンポジウム、pp. 13-19、1985
- [56] 松井 進、横山 達也、柳生 和男、寺田 松昭：「通信制御アダプタの性能に関する一考察」、情報処理学会 マルチメディア通信と分散処理研究会、27-5、pp. 1-8、1985
- [57] 高橋 康弘、高田 治、寺田 松昭、村橋 誠一、檜山 邦夫：
「LSI 化 Σ ネットワークの開発 (1) - システム構成 -」、情報処理学会第 31 回 (昭和 60 年後期) 全国大会、論文番号 1Q-7、1985
- [58] 高橋 康弘、高田 治、檜山 邦夫、寺田 松昭：多元情報光 LAN におけるハイブリッド交換アーキテクチャに関する一考察、電子通信学会情報ネットワーク研究会、IN85-115、1985
- [59] 寺田 松昭、伏見 仁、安元 精一、高橋 正弘、古寺 博：「超高速光 LAN を応用した分散処理システムの制御方式」、情報処理学会第 30 回 (昭和 60 年前期) 全国大会、論文番号 1P-6、1985
- [60] 寺田 松昭、柳生 和男、平田 哲彦、森藤 素良：「マルチメディアブロードバンド LAN の方式提案」、情報処理学会第 32 回 (昭和 61 年前期) 全国大会、論文番号 4D-3、1986
- [61] 柳生 和男、平田 哲彦、寺田 松昭、西沢 隆彦：「マルチメディアブロードバンド LAN のデータ通信方式」、情報処理学会第 32 回 (昭和 61 年前期) 全国大会、論文番号 7D-1、1986
- [62] 福澤 淳二、寺田 松昭、林 慶治郎：「LAN ベース分散処理用ネットワークソフトウェアの一考察」、情報処理学会第 33 回 (昭和 61 年後期) 全国大会、論文番号 3U-4、1986
- [63] 松井 進、柳生 和男、寺田 松昭：「LAN-WAN 相互接続網におけるネットワークレイヤプロトコルの提案」、情報処理学会第 33 回 (昭和 61 年後期) 全国大会、論文番号 4U-11、1986
- [64] 木名瀬 敏彰、寺田 松昭、大塩 剛至、岡田 重雄：「パソコン通信用ゲートウェイに関する一考察」、情報処理学会第 34 回 (昭和 62 年前期) 全国大会、論文番号 1Y-8、1987
- [65] 寺田 松昭：「音声・データ統合 LAN の応用サービス」、昭和 62 年電気・情報関連学会連合大会、論文番号 33-4、1987

- [66] 寺田 松昭、柳生 和男、森藤 素良：「マルチメディアブロードバンド LAN 技術のマルチメディアサービスシステムへの適用方式」、情報処理学会第 34 回（昭和 62 年前期）全国大会、論文番号 5Y-2、1987
- [67] 桎尾 次郎、寺田 松昭、星 徹：「地域、構内ネットワークの展望、電子情報通信学会誌」、Vol. 70, No. 11, pp.1103-1111, 1987
- [68] 寺田 松昭：「LAN、MAN への ATM の応用」、昭和 63 年電気・情報関連学会連合大会、論文番号 22-6、1988
- [69] 山内 雪路、滝安 美弘、田中 捷樹、寺田松昭：「ATM 技術を用いた高速基幹 LAN の検討」、電子情報通信学会 情報ネットワーク研究会、IN88-121、pp. 25-29、1989
- [70] 桎尾 次郎、寺田 松昭、天田 栄一：「広帯域 LAN の技術動向と国際標準化」、電子情報通信学会論文誌、Vol. J72-B-1, No. 11, PP.896-903、1989
- [71] 福澤 淳二、寺田 松昭、溝河 貞生：「高性能 MAC ブリッジ構成構成方式の提案と評価」、情報処理学会論文誌、Vol.31, No.8, pp.1260-1267, 1990
- [72] 塚越 雅人、高田 治、寺田 松昭：「LAN 間ブリッジ接続に関する一考察」、情報処理学会第 40 回（平成 2 年前期）全国大会、論文番号 3N-8、1990
- [73] O.Takada, M.Tsukakoshi, M.Terada, M.Yamaga: "An FDDI Bridge for the Super Backbone LAN", Proc. of Localnet'90, pp.399-404, 1990
- [74] 重差 秀彦、滝安 美弘、天田 栄一、寺田 松昭、畠山 靖彦、山鹿 光弘：「マルチメディア高速基幹 LAN $\Sigma-600$ の開発」、電子情報通信学会春季全国大会、論文番号 B-638、1991
- [75] 塚越 雅人、高田 治、寺田 松昭、山鹿 光弘：「マルチメディア高速基幹 LAN $\Sigma-600$ の FDDI ブリッジ方式」、電子情報通信学会春季全国大会、論文番号 B-640、1991
- [76] E. Amada, T. Morita, M. Terada, M. Yamaga, M. Miyamoto: "Architecture and technology for Next Generation Multimedia Private Networks ", Proc. of 6th Telecom'91 Forum, 1991
- [77] 平田 哲彦、横山 達也、水谷 美加、寺田 松昭、三巻 達夫：
「プロトコル高速処理装置により高性能化をはかった LAN 用通信制御装置の一構成法」、情報処理学会論文誌、Vol.33, No.2、平成 4 年 2 月（掲載決定）
- [78] M.Tsukakoshi, O. Takada, T. Murakami, M.Terada, M. Yamaga:" Large-scale and High-speed Interconnection of Multiple FDDIs using ATM-based Backbone LAN" , Proc. of IEEE INFOCOM'92 (to appear)