

高速パケット伝送路用前置処理装置の一構成法†

寺田 松 昭** 関 高 明**
 樫 尾 次 郎** 堀 雄 太 郎***

計算機制御システムにおいては、データウェイに代表される高速パケット伝送路が、複数の計算機を相互に接続するために広く用いられている。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。本論文では、前置処理装置 (FEP) によって、データ転送性能を向上する方式を採り上げ、FEP の方式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 (IOP) と 1 台のマルチプレクサとで構成し、計算機と伝送路との間に挿入している。(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/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、IOP 2 台から成る試作装置によって実験的に評価し、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 40%~80% になることを明らかにしている。

1. はじめに

計算機制御システムにおいては、データウェイに代表される高速パケット伝送路が、複数の計算機を相互に接続するために広く用いられている¹⁾。近年、計算機制御システムは大規模化が進み、計算機間のデータ転送スループットの向上、応答時間の短縮、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²⁾⁻⁵⁾。

これらの要求を満たすためには、伝送路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が必要である。伝送速度は、光ファイバ通信技術の適用により、1~2 Mbps から 10~32 Mbps へと約 1 桁高速化された⁶⁾⁻⁹⁾。一方、データ転送処理は、あいかわらず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ファームウェアで行う方法と専用の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は高い。

パケット単位の処理をハードウェアで実現する方法には、専用の論理回路を組む方式³⁾と前置処理装置を計算機と伝送路との間に設ける方式¹⁰⁾⁻¹³⁾とがある。と

ころが、高速パケット伝送路を対象とした前置処理装置の報告は少なく、その効果の評価は十分でない。

本論文では、前置処理装置を設ける方式を採り上げ、前置処理装置の一構成法の提案と性能向上度の評価を行う。提案する前置処理装置は、複数個の高速マイクロコンピュータと 1 台のマルチプレクサとで構成している。計算機内のメッセージは、マイクロコンピュータでパケット化され、マルチプレクサ経由で伝送路に送出される。

各マイクロコンピュータには、メモリを少なくするため、1 パケット分のバッファを備え、メッセージのセグメンティングとアセンブリは、マイクロコンピュータが、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り行う。計算機は、メッセージ単位に、前置処理装置の起動と終了の処理を行えば済むので、負荷の削減が期待される。提案方式の効果は、マイクロコンピュータ 2 台から成る試作装置を、10 Mbps の光データウェイに適用して、実験的に評価する。

2. データ転送処理の高速化

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。

2.1 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 1 に示すような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

† A Front End Processor for High Speed Packet Multiplexed Line by MATSUAKI TERADA, TAKA AKI SEKI, JIRO KASHIO (Systems Development Laboratory, Hitachi, Ltd.) and YUTARO HORI (Omika Works, Hitachi Ltd.).

** (株)日立製作所システム開発研究所

*** (株)日立製作所大みか工場

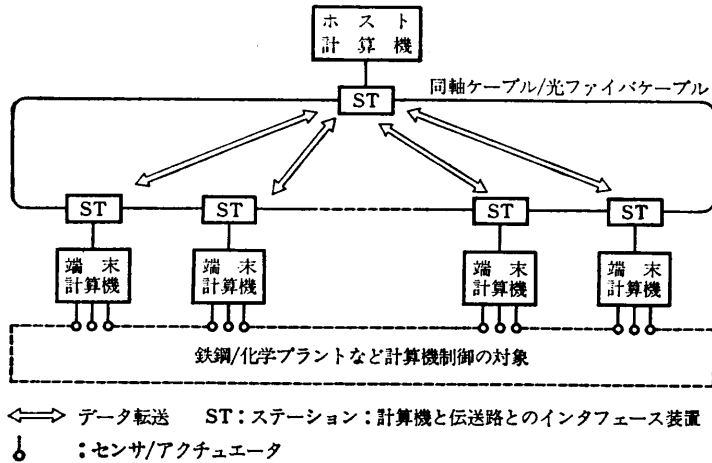


図1 高速パケット伝送路を用いた計算機制御システム
 Fig. 1 Computer control system with high speed packet multiplexed line.

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（計算機とのインタフェース装置）とで構成されている。ステーション相互間では、パケットとよばれる最大500バイト程度の可変長データブロックが、10~32 Mbps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、おもに、ホスト計算機と10~20台の端末計算機の間で行われる。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

これらの術語は、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

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転送する。

従来は、計算機とステーションとが、図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。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/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（論理チャンネル）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。

2.3 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、

- (1) ファームウェア化
- (2) ハードウェア化

が考えられる。(1)は、計算機のデータ転送処理すべて(2.2節のA~D)にわたって高速化を行うことは

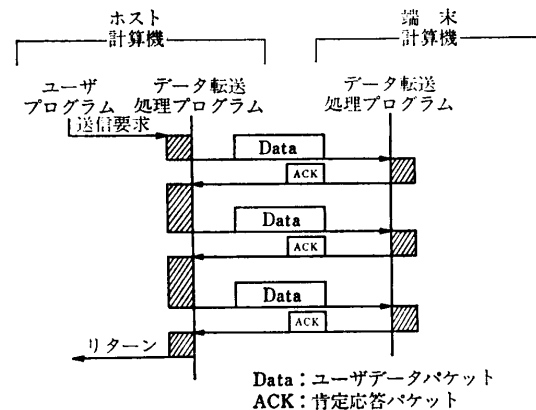
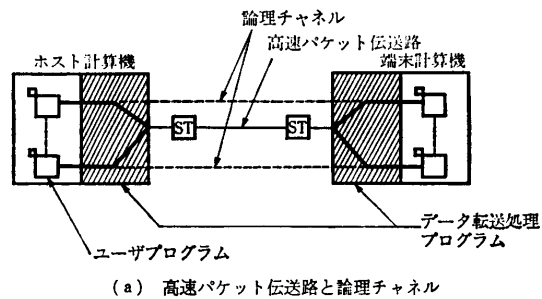


図2 従来システムのデータ転送モデル
 Fig. 2 Data transfer model of conventional system.

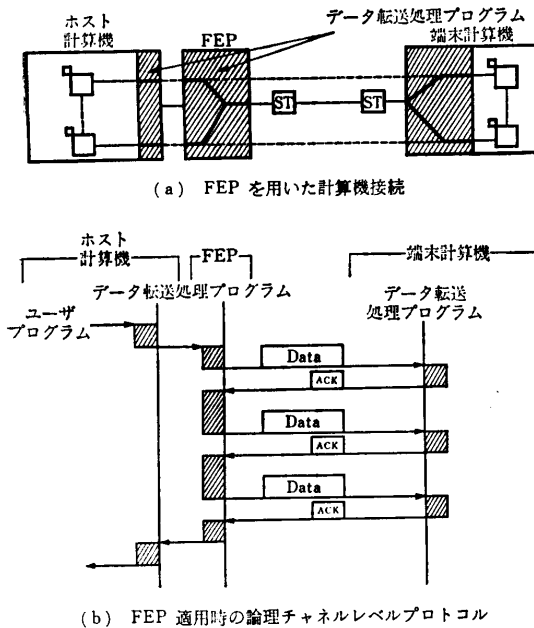


図 3 FEP 適用時のデータ転送モデル。
FEP: 前置処理装置

Fig. 3 Data transfer model of a system with front end processor.

できるが、計算機負荷の低減効果が少ない。(2)は、パケット単位の処理(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、論理チャン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方式を探りあげる。FEPの導入によって、計算機は、図3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、FEPを高性能なプロセッサで構成すれば、FEPで行うパケットごとの処理を、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。

3. 前置処理装置(FEP)の構成法

高速パケット伝送路に適した前置処理装置(FEP)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。

3.1 要求条件

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FEPには、

- (1) 10~32 Mbps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) 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
- (4) メッセージのセグメンティング/アセンブリおよび伝送路のパケットを論理チャンネルに分ける処理をFEPが行えるようにする計算機インタフェース

が必要である。

3.2 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、FEPの命令実行能力で定まる。したがって、10~32 Mbpsの高速パケット伝送を可能にするには、FEPに十分な処理能力をもたせる必要がある。そこで、提案方式では、FEPを複数のプロセッサ(以下IOP: Input/Output Processorと称す)で構成することにした。

複数IOPによるFEPの構成方式としては、機能分散方式と負荷分散方式とが考えられる。両方式を図4に示す。

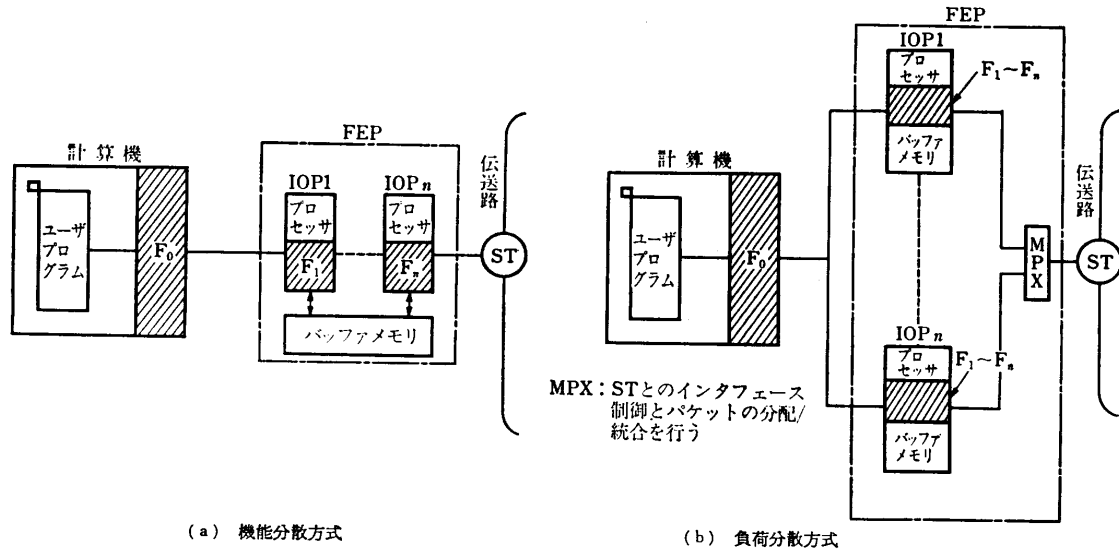
(a) 機能分散方式

各IOPは、個有の機能を分担する。計算機からのデータは、まずIOP1に渡され、以降IOP2, IOP3, ...と渡され、最後にステーションに渡される。

(b) 負荷分散方式

各IOPは、同一の機能を備えている。計算機からのデータは、各IOPに分配され、各IOPでは、これらのデータを一時、IOP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IOPを対等にするため、マルチプレクサ: MPXとよ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待行列モデルを仮定する限り、負荷分散方式がすぐ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、IOPのメモリ容量を削減する工夫を行うこととした。



(a) 機能分散方式
ST: ステーション, FEP: 前置処理装置, IOP: IOP を構成するマイクロコンピュータ, ㊦: データ転送処理プログラム

図 4 FEP の構成方式代替案

Fig. 4 Two types of system configuration of front end processor: function distribution and load distribution.

3.3 応答時間の短縮

応答時間は、FEP の処理時間とデータを IOP 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、並列には行えないので、FEP の処理時間を短縮するには、IOP 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と、バッファメモリの読み出し/書き込みを高速にする（すなわち、高速メモリを使う）ことが必要になる。

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して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。

- (1) FEP 内でのデータ蓄積は、IOP のみにとどめ、MPX では行わない。
- (2) IOP のバッファメモリは、1パケット分のみを準備する。
- (3) 計算機のメモリを各 IOP の共有メモリとして用いる。

3.4 FEP—伝送路インタフェース

高速パケット伝送路の特徴は、同一伝送路上を、異なる論理チャンネルの 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 IOP 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、FEP と伝送路/計算機とのインタフェースの実現方法が重要な課題である。

FEP と伝送路とは、マルチプレクサ (MPX) によって接続され、受信 packets は MPX によって IOP に分配される。MPX には、IOP n 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 packets 分配方式とが求められる。

MPX は、上記目的を満たすため、packet ヘッダの参照、論理チャンネルへの振り分けをすることなく、packet を IOP に分配する。すなわち、MPX は、各 IOP のバッファメモリに packets 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 packets 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 packets 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。

3.5 FEP—計算機インタフェース

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 FEP 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/アセンブリとパケットを論理チャン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 とのインタフェースの動作は、大きく、送信と受信に分けられ、おのおのは、

- (1) 計算機から IOP への起動
- (2) IOP から計算機への終了報告

となる。

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出して、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、セグメントごとに受信側から返されてくる応答を受信した IOP が、計算機メモリのメッセージを読み出して送る。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(該当論理チャンネル制御テーブル)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ン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。パケットを受けとった IOP は、パケット・ヘッダに定義されている論理チャンネル番号に対応する論理チャンネル制御テーブル(計算機のメモリにある)より、受信パケットのストア・アドレスを取り出し、そこへパケットをストアする(パケットのアセンブリ)。最終セグメントのパケットをストアした IOP が計算機に対して受信完了割込みを入れる(終了報告)。

このような IOP の連係動作を可能にするためには、論理チャンネル制御テーブルを計算機のメモリ

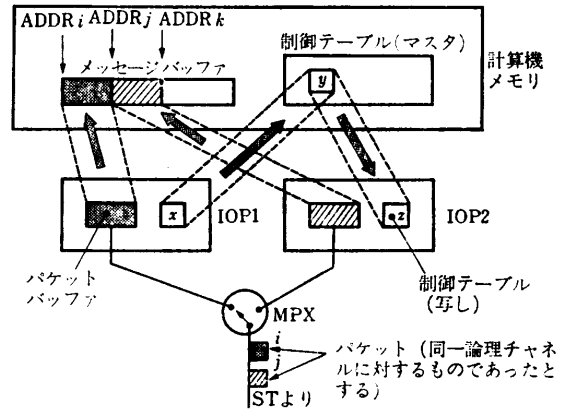


図 5 制御テーブル・リレー方式(受信時のパケット・アセンブリ)

Fig. 5 Control table relaying mechanism for packet assembly.

を介して、ある IOP から別の IOP へリレーできるようにする必要がある。すなわちマスタを計算機のメモリに置き、セグメンティング/アセンブリを行う IOP がそれを自分のバッファメモリに一時的に COPY 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よぶ(図 5)。

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、IOP には、パ

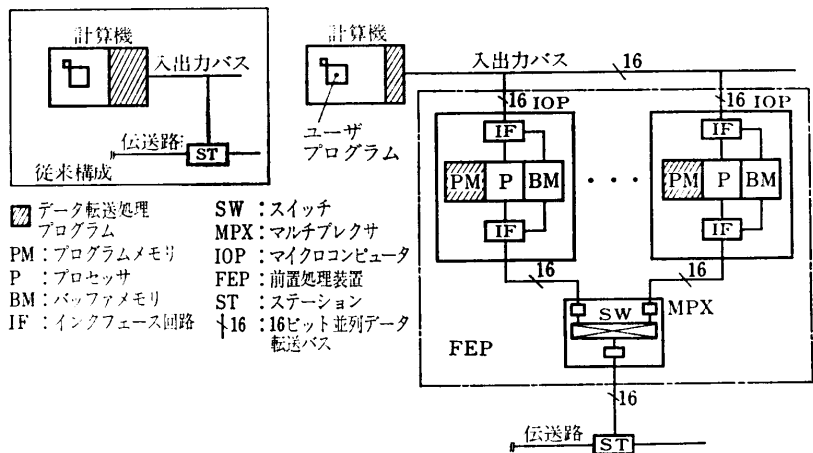


図 6 提案前置処理装置のシステム構成

Fig. 6 System configuration of proposed front end processor.

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/アセンブリを、IOP が行える。

3.6 FEP の構成例

提案した FEP のシステム構成をミニコンへの適用を例にして図6に示す。FEP を、複数のマイクロコンピュータ (IOP) と1台のマルチプレクサ (MPX) とから構成する。各 IOP 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 IOP を集約して、ステーションに接続するため MPX を、IOP とステーションとの間に入れる。MPX には、パケット蓄積バッファを設けない。

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スループットが期待できる。さらに、FEP でメッセージのセグメンティング/アセンブリを行うので、計算機は、メッセージ単位に FEP 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低減が期待できる。

4. 実験的評価

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、FEP 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。

4.2 評価方法

3章で提案した方式に従って FEP を製作し、FEP

付システムと従来システムとで性能を実測し、比較する。

FEP は、2台の IOP と MPX とから構成し、計算機の入出力バスとステーション (ST) との間に挿入した。MPX は、すべてハードロジックで構成した。IOP 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 200 ns で実行できるようにした。IOP の仕様を表1に示す。

IOP 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

表 1 IOP の仕様

Table 1 Specification of the microcomputers of which our front end processor consists.

項番	項目	仕様	備考	
1	プログラム・メモリ	語長	32 ビット	
		容量	4,000 語	
		サイクルタイム	40 nsec	
2	データ・メモリ	語長	16 ビット	・制御テーブル ・パケットバッファ として 600 語を使用している。
		容量	1,000 語	
		サイクルタイム	55 nsec	
3	演算部	命令数	70	AMD 社 2900 シリーズ使用 4 ビット×4
		レジスタ	16 個	
		速度	200 nsec/命令	
		語長	16 ビット	
4	マイクロプログラム	3,600 ステップ		

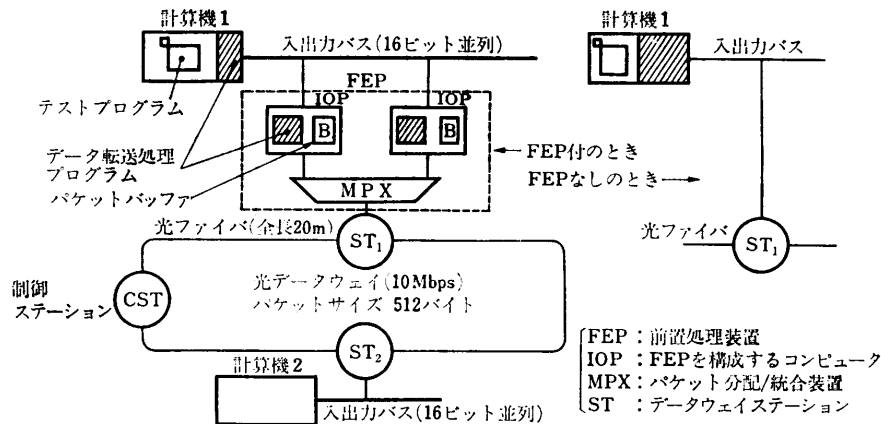


図 7 実測システムの構成

Fig. 7 System configuration for evaluating proposed front end processor.

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7)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、FEP/計算機と伝送路とを接続する。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。計算機は、語長16ビットのミニコンで、システムミックスは2μs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くり返し行う。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ST₂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。ST₂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ST₂に書き込み要求を行う(セグメントティング)。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ST₂が行う。

FEP付システムと従来システムとのタイムチャートを図8に示す。おのおのの処理時間(t_{ij})は、次の

とおりである。

- 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からの割込みを受け、テストプログラムを再起動
- t_{14}' : IOP BUSY 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込みを発生する
- t_{24} : IOP BUSY 解除要求
- t_{25} : パケット送信後処理
- t_{31} : パケットの送信(ST₁)
- t_{32} : パケットの受信(ST₁)
- t_{41} : メモリへの書き込みを行い応答パケットを送信(ST₂)

測定条件は、下記とする。

- (1) 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まったく動いていない。
- (2) 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。
- (3) 性能は数分間同じ動作をくり返し、その間の平均値で算出する。

4.3 結果

4.3.1 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9である。これより、データ長が500W(1W:16ビット)のとき2.5倍、4KWのとき3.5倍スループットが向上することがわかった。

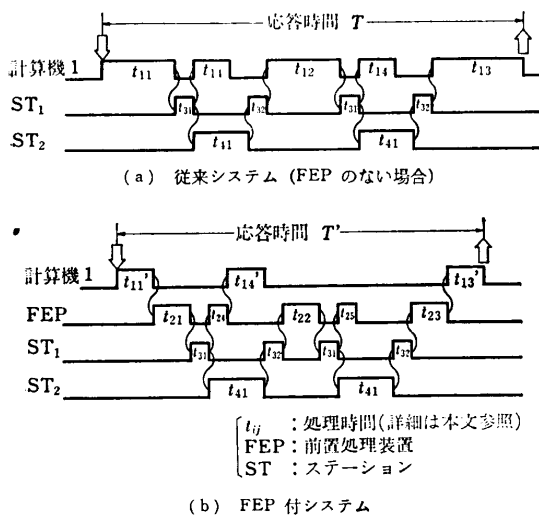


図8 データ転送タイム・チャート。2セグメントの例、セグメント数はデータ長により変わる

Fig. 8 Data transfer time chart. Case of two segments data.

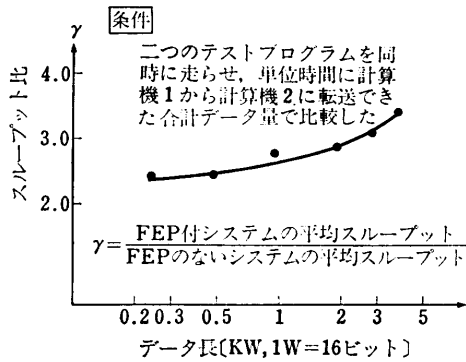


図 9 スループット比

Fig. 9 Throughput ratio compared to conventional system.

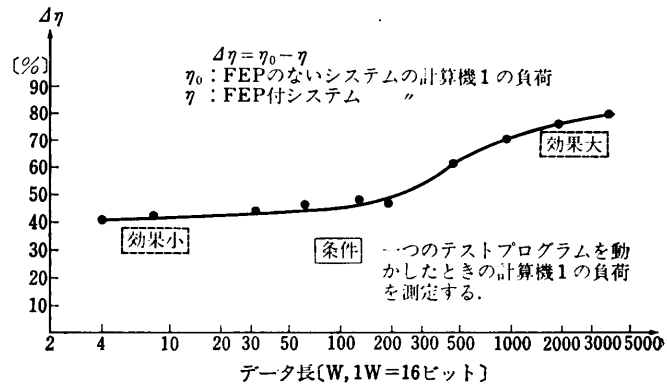


図 11 計算機負荷低減効果

Fig. 11 CPU load reduction by use of FEP.

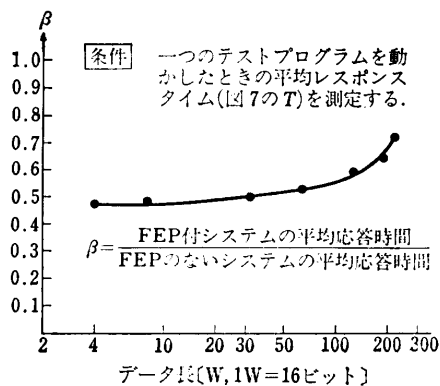


図 10 応答時間比 (FEP なしを1)。

Fig. 10 Response time ratio compared to conventional system.

4.3.2 応答時間

応答時間として図8の T および T' をとり、種々のデータ長で測定した結果を比較したのが図10である。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により、応答時間は、データ長 30 W 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)。

4.3.3 計算機負荷

データ長を変えて、テストプログラムをく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 11 である。これより、データ長が 4 W と短いところでも 40% 計算機負荷が減り、4 KW 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。

5. む す び

高速パケット伝送路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、マルチコンピュータ構成前置処理装置を提案した。

提案方式の前置処理装置による性能向上度を把握するため、マイコン 2 台から成る前置処理装置を試作し、伝送速度 10 Mbps の高速パケット伝送路に適用した。

実験の結果、計算機間データ転送性能は、前置処理装置がないときに比べて、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 2 KW)。
- (2) 応答時間が 1/2 になる (データ長 30 W)。
- (3) 計算機負荷が 40% (データ長 4 W) ~ 80% (データ長 4 KW) 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伝送路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。

謝辞 終りに、本研究を進めるに当たり、光データウェイの提供と、応用面からの貴重な助言とをいただいた(株)日立製作所・大みか工場 桑原 洋副工場長、平井浩二副部長、伏見仁志主任技師、本研究の機会を与えていただいた当社システム開発研究所所長 三浦武雄博士、本研究の遂行と本論文をまとめるに当たりご指導いただいた三巻達夫博士、大成幹彦博士に

深謝いたします。

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) Matsumoto, Y. et al.: A Distributed Processing System and Its Application to Industrial Control, *NCC '78*, pp. 1273-1279 (1978).
- 3) Jensen, E. D.: The Honeywell Experimental Distributed Processor—An Overview, *IE³ COMPUTER*, Vol. 11, No. 1, pp. 28-39 (1978).
- 4) 平子叔男，寺田松昭他：制御用分散処理システム，情報処理，Vol. 20, No. 4, pp. 346-349 (1980).
- 5) 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).
- 6) Takahashi, M. et al.: Optical Fiber Data Freeway System—A Loop Network for Distributed Computer Control, *COMPCON '81*, Spring pp. 458-463 (1981).
- 7) 石坂充弘他：高速ループシステム (LOOP-3)，情報処理学会第 21 回全国大会予稿集，pp. 663-656 (1980).
- 8) 匠 建太：光データリンクシステムとその分散処理への応用，情報処理学会分散処理システム研究会，9-4 (1981).
- 9) 藤井 哲，満岡弘雄：システム開発・保守効率向上のためのシステム構成面からのアプローチ，オートメーション，Vol. 26, No. 9, pp. 18-25 (1981).
- 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，Vol. 62, No. 11, pp. 1296-1303 (1979).
- 12) Heart, F. E. et al.: A New Minicomputer/Multiprocessor for the ARPA Network, *NCC '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) Terada, M. et al.: A Network Operating System for High Speed Optical Fiber Loop Transmission System, *Proc. the 5th ICCS*, pp. 641-646 (1980).
- 15) 寺田松昭他：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 回全国大会，論文番号 5D-7 (1981).
- 16) 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，電気学会誌，Vol. 98, No. 3, pp. 199-203 (1978).

(昭和 56 年 10 月 20 日受付)
(昭和 57 年 6 月 15 日採録)