マイクロコンピュータ 43－1
（１９８７・２・６）

マルチマイクロプロセッサＤｉａｌｏｇの通信機構

濱崎　陽一　　岡田　義邦　　田島　裕昭　　鈴木　基史
電子技術総合研究所

本報告では光を用いた特殊なバスを持つマルチマイクロプロセッサDialogについて、通信機構の観点から述べる。　自由空間を光の伝達媒体とする２種類の光バス、円筒鏡による光バスとホログラムによる光バス、について原理を説明し、その特徴からマルチプロセッサシステムでの利用法を考察する。　Dialogは共有メモリと各プロセッサユニットがそのキャッシュメモリを持つシステムであって、円筒鏡による光バスを用いた共有メモリの管理について述べる。　また複数のホログラムによる光バスを用いて、プロセッサ間通信に使用されるネットワークを構成する方法についても述べる。　最後に現在試作中のDialog.Hプロトタイプシステムについて述べる。

The Communication in the Multi Microprocessor System Dialog

Youichi HAMAZAKI　　Yoshikuni OKADA　　Hiroaki TAJIMA　　Motohiro SUZUKI
Electrotechnical Laboratory
1-1-4, Umezono, Sakura-mura, Niihari-gun, Ibaraki-ken 305, JAPAN

The multi microprocessor system Dialog which hires the optical buses is described from the view point of the communication.　The principles of two optical buses,　the optical bus with cylindrical mirror and the optical bus with horograms,　are explained and applications of optical buses in a multiprocessor system are discussed.　Dialog system has common memory and cache memory in each processor unit.　The optical bus with cylindrical mirror is used for the management of common memory and cache memory.　The topology of communication network which is formed with optical buses with horograms is described.　The prototype system of Dialog.H is also described.

1. はじめに

マルチプロセッサシステムを構築しようとするとき、プロセッサ間の通信をどの様に行うかがシステム全体の構成を決定する大きな要因である。共有バスを持つシステムはその通信容量や接続可能ポート数に制限があって大規模なシステムには適さないという事情から、大規模なマルチプロセッサシステムでは各種のネットワークにより接続されたアーキテクチャを取るものが多い[1,2]。しかしその様なシステムでは規模が大きくなると共にプロセッサ間の接続のための配線が多くなり保守及びコストの点等で問題がある。筆者らはこの様な考察から光を使った新しい発想に基づく通信方式を開発し、それらを生かしたマルチプロセッサシステムDialogを構築しつつある。Dialogは数百台程度の汎用マイクロプロセッサから成るシステムを目指している。ここでは通信機構を主眼においてDialogシステムについて述べる。また現在試作を進めている数台規模のプロトタイプマシンについても述べる。

2. 光バス

光による通信は、高速大容量で電磁気雑音の問題が生じないなど利点が多い。しかし一般に使用されている光による通信は光ファイバーを通信路に使用した通信であり、光ファイバーを使用した通信は基本的に一対一通信であることから、大規模なマルチプロセッサシステムの通信路として使用するのには配線コストなどの点で問題が多い。

そこでマルチプロセッサシステムの通信路として使用する目的で、自由空間を通信路とする光による通信方式を開発した。光の経路を変更、規定するものとして、円筒鏡を使用したものと、ホログラムを使用したものの2種類があり、どちらも放送型の特性を持つことから、前者を円筒鏡による光バス、後者をホログラムによる光バスと呼ぶ。

2. 1 円筒鏡による光バス

図1に円筒鏡による光バスの原理図を示す。複数台の光送受信機は、円筒鏡の周囲に配置され、円筒鏡に向かって光信号の送受信を行う。ある光送受信機から送信された扇状の光は、円筒鏡により反射され、全ての光送受信機に到達する。これにより放送型の通信路（バス）を得ることが出来る。光受信機は全ての送信機からの信号を受信し、その信号の有無（光が入っているか否か）で1/0を判断するので、この光バスは論理和(logical OR)特性を持つ。簡単な計算により全ての光送受信機において送信する扇状の光の開き角を等しくするような配置を求めることができ、その様にして求められた曲線上の任意の位置にある方向を向けて光

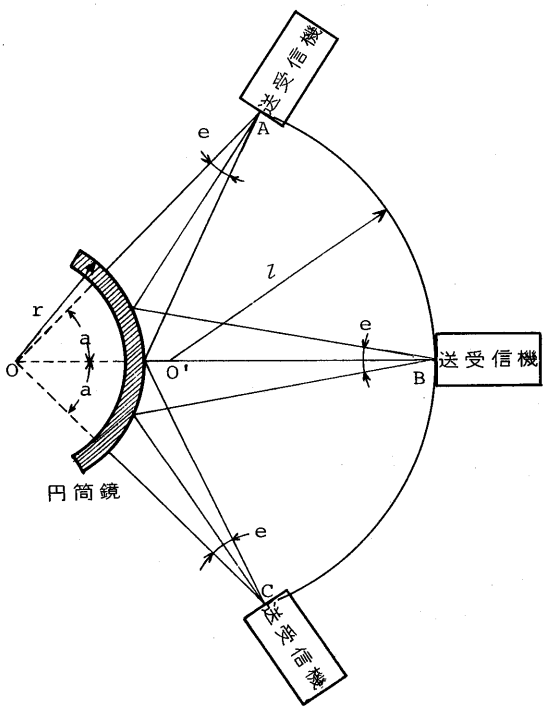

図1 円筒鏡による光バスの原理

送受信機をおけばよい。図から明らかなように光バスの原理は二次元的であり、垂直方向（図では紙面に垂直な方向）に積み上げることにより複数チャネルのバスを得ることが出来る。

円筒鏡による光バスの利点は、光を用いたことによる高速大容量性や耐雑音性の他に次のようなものがある。

① バスに接続し得るポート（送受信機）数は、送信光出力と光受信機の感度により決まり、1000以上のポートを1つのバスに接続することも可能である。

② ポート間の光路長がほぼ等しいために、スキューの問題が起きにくい。

③ 全ての光送受信機において光放射特性を同じに出来るので、光学系の製作が容易

④ バスの速度が、そのバスに接続されるポート数に影響されない。

⑤ 配線が不用で、光送受信機の追加、削除が容易。

他のバスと同様に、同時に光バスに送信できるポートは高々一つであるから、同時に複数のポートからバスの要求がある場合にはバスの使用権を決定するアービトレーション（調停）が必要となる。円筒鏡による光バスは接続ポート数が大きいため集中型アービトレーションではアービタがボトルネックになりやすく、またポートの追加削除に柔軟な特性を生かすためにも集中型アービタは不都合である。そのため光バスに適した分散型アービトレーション法、一進多段比較法を開発した[3]。一進多段比較法は、光バスの持つ論理和特性と、一進数どうしの桁毎の論理和を取った結果が元の一進数のうちの最大のものと等しくなる性質を利用したものである。

アービトレーションに必要な時間は一段当り光を送信してから受信するまでの時間に判断のための時間を加えたものとなる。前者は光バスの光路長、即ち物理的な大きさにより制限されるために、アービトレーションに必要な時間には下限がある。それに対して光によるデータ通信の技術は既にGHz以上の転送速度を実現しており、今後更に高速化されることが期待される。例えば現在試作中である光バスのパイロットモデル[4]ではポートを配置する曲線の長さが200cmで光路長は約300cmあり、光の伝播に必要な時間は約10nsである。データおよびプライオリティのためのチャネルの数が8でポート数が数百（$\leqq 729=(8+1)^3$）とすると3段のアービトレーションが必要となり、それに要する時間は数十nsとなる。それに対してデータの転送に必要な時間は各チャネルの転送速度を1GHzとするとバイト当り1nsである。このことから円筒鏡による光バスに適した通信形態は、アービトレーションの占める割合が比較的小さな大きなパケットによるパケット通信である。また多数のポートを接続可能であることから、マルチプロセッサシステムの共有バスに使用するのに適している。

2．2 ホログラムによる光バス

円筒鏡による光バスでは円筒鏡が光信号を配布する働きをしたが、ホログラムを使用すると選択的な光の配分が可能である[5]。図2にホログラムによる光バスの原理図を示す。各光送受信機に対応したホログラムにはその光送受信機から出されたコヒーレント光が残りの光送受信機に達するようなパターンが記録されている。そうすると任意の光送受信機から送信された信号は他の全ての光送受信機で受信できるから、放送型の通信路を構成することが出来る。この様なホログラムは各受信位置を多重露光により一枚の乾板に記録することにより得られる。

もちろんホログラムに記録するパターンによっていろいろな通信路を構成することが出来るが、Dialogシステムで使用したものは放送型の通信路（バス）である。ホログラムによる光バスには円筒鏡による光バスの利点②④以外に次のような特徴がある。

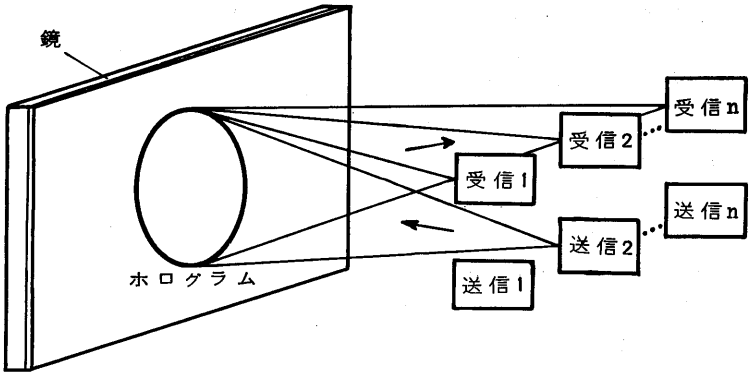

図２　ホログラムによる光バスの原理

❶　バスに接続し得るポート数はホログラムに多重露光可能な数により制限され、現時点で10以下である。

❷　配線が不用で、ホログラムの交換によりネットワークのトポロジーの変更が可能である。

❸　ホログラムは写真技術により複写できるので、チャネル数を増やすことは比較的容易。

ホログラムによる光バスも論理和特性を持つから、一進多段比較法によるアービトレーションが可能である。ホログラムによる光バスは上の❶であげたように、接続し得るポート数が小さいことから、複数の光バスを用いてネットワークを構成し、マルチプロセッサシステムにおけるプロセッサ間の通信に用いるのに適している。

3.　Dialogシステムの構成

Dialogシステムの構成図を図３に示す。Dialogは共有メモリをもつマルチマイクロプロセッサシステムで、各プロセッサユニットは共有メモリのキャッシュメモリ（仮想共有メモリ）を持っている[6)]。Dialogのプロセッサユニット及びメモリユニット間には２つの通信路、共有バスとローカル通信ネットワーク、があり分散型オペレーティングシステムDialog.Mの元で動作する。

3．１　共有バスと仮想共有メモリ

共有バスは円筒鏡による光バスである。円筒鏡による光バスは比較的大きなパケットの通信に適することから、メモリユニット（共有メモリ）とプロセッサユニットの持つキャッシュメモリとの間の通信に用いる。この場合パケットの大きさはキャッシュメモリのページの大きさとな

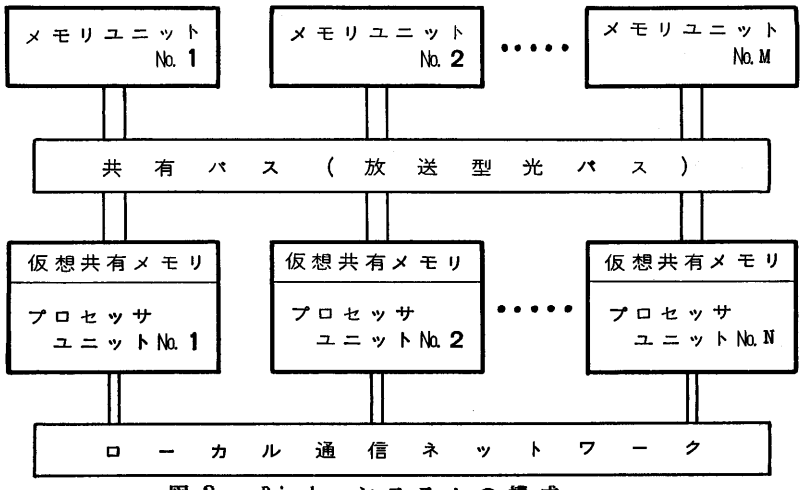

図３　Dialogシステムの構成

る。各プロセッサユニットから見た場合には一般の仮想メモリのように動作し、キャッシュメモリのページの大きさが比較的大きいことからこの方式を仮想共有メモリ方式と呼んでいる。

キャッシュメモリを持つ共有メモリシステムでは、システム内に散在する共有メモリのコピーの管理が大きな問題となる。すなわちあるプロセッサがキャッシュメモリに書き込みを行った場合に他のプロセッサユニット及びメモリユニットの内容の更新をどう行うかの問題である。書き込みを行うページのコピーをどのプロセッサユニットが持っているかを知ることは非常に困難であり、たとえそれが知れたとしてもその全てに個別に情報を送ることは現実的ではない。Dialogでは放送型の共有バスを持っているので、キャッシュメモリに書き込みを行ったプロセッサユニットはそのページを共有メモリに書き戻す際に同時に全てのプロセッサユニットにもそのページの新しい内容を伝えることが出来る。共有メモリへの書き戻しの情報を受け取ったプロセッサユニットはそれに対応するページがキャッシュメモリ内に存在するかどうかを調べ、もし存在すればキャッシュメモリの内容の更新を行う。もし該当するページが存在しなければ情報を無視するだけでよい。この方法はページのコピーの存在の有無及びその存在場所を気にする事なく確実に全てのコピーの内容を同時に更新でき、更にその手続きが各プロセッサユニットに分散されているため大規模なマルチプロセッサシステムにおいてもボトルネックになりにくい方法である。

但し同じページに同時に複数のプロセッサユニットが書き込みを行った場合には内容の矛盾を避けられないので、ある共有メモリのページに書き込みが許されるプロセッサユニットは一時に高々１つに制限しなくてはならない。

３．２　ローカル通信ネットワーク

Dialogのプロセッサユニット間の通信路はホログラムによる光バスにより構成される。ホログラムによる光バスは接続できるポート数が比較的小さく全てのプロセッサユニットを１つのバスに接続することは不可能であるから、複数の光バスを使って対数階層的なトポロジーを持つネットワークを構成する。

ユニット（ＰＵまたはＭＵ）の総数がk^nであり、一本のバスに接続されるユニットの数をｋ（ｋは２の累乗）とすると、ユニットは一辺の長さがｋ-1のｎ次元立方体の格子点に配置され、各ユニットにはｋ進数ｎ桁の番地が与えられる。バスの接続は次のように行う。

① 全てのユニットを第一次元の方向に接続する。このバスを低次バスと呼ぶことにする。つまり番地$A_{n-1}A_{n-2}\ldots A_1X_0$($0 \leqq X_0 \leqq k-1$)を持つｋ個のユニットが一つのバスにより接続される。

② ユニットのアドレスａとして、$(a \bmod 2^i) = 2^{i-1}-1$となる最小のｉを求め、ｉ＜ｎならば第ｉ+1次元の方向に接続する。このバスを高次バスと呼ぶことにする。つまり番地$A_{n-1}A_{n-2}..X_i..A_1A_0$($0 \leqq X_i \leqq k-1$)を持つｋ個のユニットが一つのバスにより接続される。このｉはアドレスを二進数表現したときに最も右に現れる０の位置である。

③ ②で求めたｉがｎ以上ならば、第ｎ次元の方向に接続する。このバスも高次バスと呼ぶことにする。

上記の接続により全てのユニットは低次バスと高次バスの各々１本に接続される。同じバスに接続されていないユニット間の通信には、メッセージの中継が必要で

ある。メッセージの中継を行う一般的なアルゴリズム[7]（アルゴリズムA）は次のようになる。

①一時的な行き先番地Tを最終的な行き先番地Fとする。

②もし現在メッセージが在る番地Cが最終的な行き先番地Fに等しければ到達。

③k進数で表した番地CとTで、値の異なる一番左の桁を見つける。i桁目が異なるとすれば、第i次元の方向のバスが必要となる。

④もしユニットCが第i次元の方向のバスに接続されていれば、そのバスを使って、Cのi桁目をTのi桁目で置き換えた番地を持つユニットにメッセージを送り、①から繰り返す。

⑤もしユニットCが第i次元の方向のバスに接続されていなければ、第i次元の方向のバスに接続されているユニットで一番近いものを求め、それを一時的な行き先番地Tとして②から繰り返す。第i次元の方向のバスに接続されている最寄りのユニットの番地は、二進数で表した番地Cの下i桁を011..11のパタンに置き換えることにより得られる。但しiがnと等しい場合は下i-1桁をすべて1に置き換えることにより得られる。

しかし、このアルゴリズムでは最短の経路とならない場合が多い。それは必ず次元の高い方から転送を始めるために余分な低次バスでの転送が生じるからである。メッセージが中継及び転送される全てのユニットについて一度の低次バスによる転送によって、必要な高次バスに接続されているユニットに転送可能であるならば、次に提案するアルゴリズム（アルゴリズムB）によって最短の経路でメッセージを送ることが出来る。

❶もし現在メッセージが在る番地Cが最終的な行き先番地Fに等しければ到達。

❷ユニットCが接続されている高次バスの次元がjで、k進数で表した番地CとFのj桁目の値が異なっていたら、Cのj桁目をFのj桁目で置き換えた番地を持つユニットに高次バスによりメッセージを送り、❶から繰り返す。

❸k進数で表した番地CとFで、値の異なる一番左の桁を見つける。i桁目が異なるとすれば、第i次元の方向のバスが必要となる。i=1ならば低次バスを使ってユニットFにメッセージを送り、❶から繰り返す。

❹第i次元の方向のバスに接続されているユニットで一番適当なものを求め、そのユニットにメッセージを送り、①から繰り返す。適当なユニットとは、番地の値の異なる桁がi桁目と1桁目のみで、かつ第i次元の方向のバスに接続

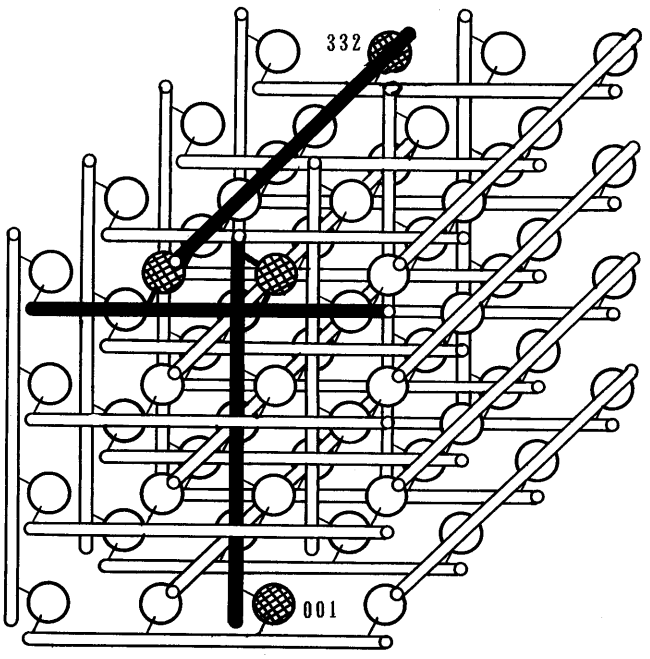

図4　ローカル通信ネットワークの例

されているユニットの番地の1桁目がFの1桁目と等しいようなものが存在する場合にはそのユニット、それ以外の場合には前のアルゴリズムと同様に求めた第i次元の方向のバスに接続されている最寄りのユニットである。

このアルゴリズムが適用できるのは、上記アルゴリズムの③ステップで求めたiが$(\log_2 k)+1$以下の場合である。ただし$n \leqq (\log_2 k)+2$の場合はつねに適用できる。図4にk=4、n=3の場合のネットワークを示す。太線で示したのはアルゴリズムBにより得られるユニット001_4とユニット332_4の通信経路である。

マルチプロセッサシステムの実行効率は、プロセッサユニット間の負荷分散の良否によって大きく左右される。負荷分散を集中制御すれば均等な負荷の分散を得ることが出来るが、大規模なマルチプロセッサシステムでは集中制御する部分がボトルネックになりやすいため分散制御による負荷分散を行った方が望ましい。負荷分散を分散制御で行うためには各プロセッサがその負荷を分散する先である隣接プロセッサの負荷状況を知る必要がある。Dialogのローカル通信ネットワークにおいて通信のたびに各プロセッサが負荷状況（ビジー度）を放送するようにすれば、通信量を余り増やすことなく隣接プロセッサの負荷状況のよい近似値を得ることが出来る。その様にして得られた負荷状況を元にしてシミュレーション実験を行った結果均等な負荷分散が得られた[8]。

4. Dialogのプロトタイプシステム －現況－

試作を進めているDialogのプロトタイプシステムは、目的とするシステムの約1/100の規模で数台のプロセッサユニットと1台のメモリユニットからなる[10]。円筒鏡による光バスの試験システムではデータ転送速度がチャンネルあたり100Mbpsという成果が得られているので[9]、プロトタイプシステムではバスに対する要求がプロセッサユニットの数に比例するという仮定から共有バスの速度を1MByte/sと決めた。ローカル通信ネットワークを構成するバスの速度は1Mbit/sとした。

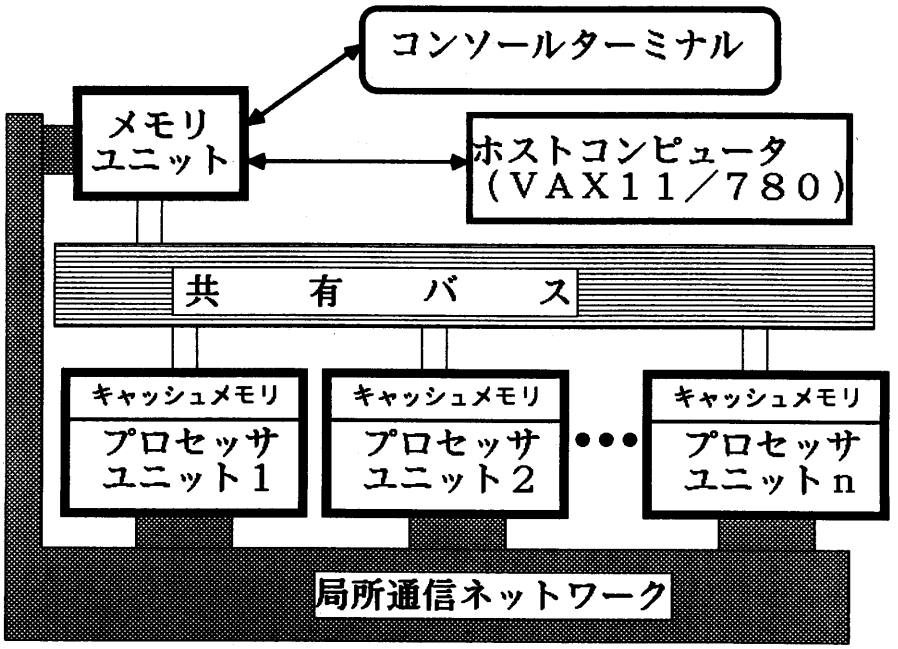

図5　プロトタイプシステムの構成

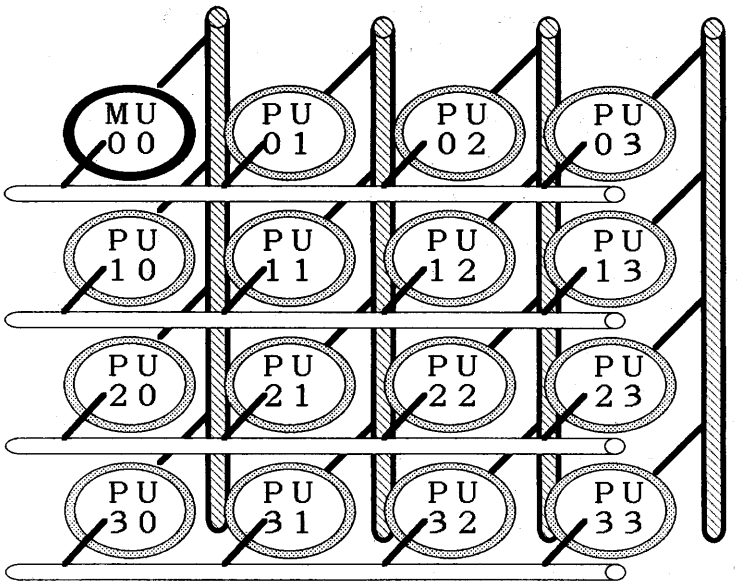

図６　プロトタイプシステムのローカル通信ネットワーク

ローカル通信ネットワークは２次元格子状のトポロジーとした。これはk=4, n=2のときのネットワークでもある。円筒鏡による光バス及びホログラムによる光バスを実際に使用するのがよいのであるが、まだ光バス自身が研究段階にありコストの問題もあるために、それぞれwired-OR接続した電線により実現されている。

プロセッサユニット及びメモリユニットのプロセッサはMC68010及びMC68000であり、CPU基板等は市販のものを、バスのインタフェースやキャッシュメモリはラッピングによる配線でユニバーサル基板上に作成した。図５にプロトタイプシステムの構成図を、図６にローカル通信ネットワークのトポロジーを示す。

現在ハードウェアの製作は一応終了し、ソフトウェアの開発の段階にある。オペレーティングシステムのＯＳカーネル部分（共有メモリ管理、通信管理、プロセス管理を行う）を作成し、その上で動作する論理型言語の並列インタプリタの設計を進めているところである。ＯＳカーネルにインプリメントされた通信のパケットには次のようなものがある[11]。

共有バスで使われるパケットは、下に示す６種である。右の括弧内にパケットの流れる方向が示されている。ＰＵｓは送信したプロセッサユニット（ＰＵ）以外の全てのプロセッサユニットの意味である。

スワップイン要求（ＲＳＩ）	［ＰＵ⇒ＭＵ］
スワップインページ（ＳＩＰ）	［ＭＵ⇒ＰＵ］
スワップアウトページ（ＳＯＰ）	［ＰＵ⇒ＭＵ，ＰＵｓ］
ワードライトスルー（ＷＷＴ）	［ＰＵ⇒ＭＵ，ＰＵｓ］
書き込みページ要求（ＲＷＰ）	［ＰＵ⇒ＭＵ］
書き込みページアドレス（ＷＰＡ）	［ＭＵ⇒ＰＵ］

キャッシュメモリへのアクセスでページフォールトを生じたプロセッサユニットは、メモリユニットに向けて必要な共有メモリのページをＲＳＩパケットにより要求する。ＲＳＩパケットを受け取ったメモリユニットはそのページをパケッ

トにしてＳＩＰパケットとして要求したプロセッサユニットに送る。新しいページを受け取るために書き込み、変更をしたページを割り出す必要があるときには、そのページをパケットにしてＳＯＰパケットとして送り出す。ＳＯＰパケットはメモリユニット及び送信したプロセッサユニット以外のプロセッサユニットによって受信され、各プロセッサユニットは受信したＳＯＰパケットと同じ番地のページを持っているかどうかを調べ、もしあればその内容を更新する。この機構により不特定多数のコピーの内容の更新が同時に出来る。ＷＷＴパケットはあるページの中の１語のみを放送するためのものである。またＲＷＰ、ＷＰＡパケットは書き込み可能ページの管理に用いるものである。パケットの種類及び行き先により受信するか否かはインタフェースをコントロールするシーケンサのプログラムにより判断される。

ローカル通信ネットワークで使われるパケットの構造を図７に示す。

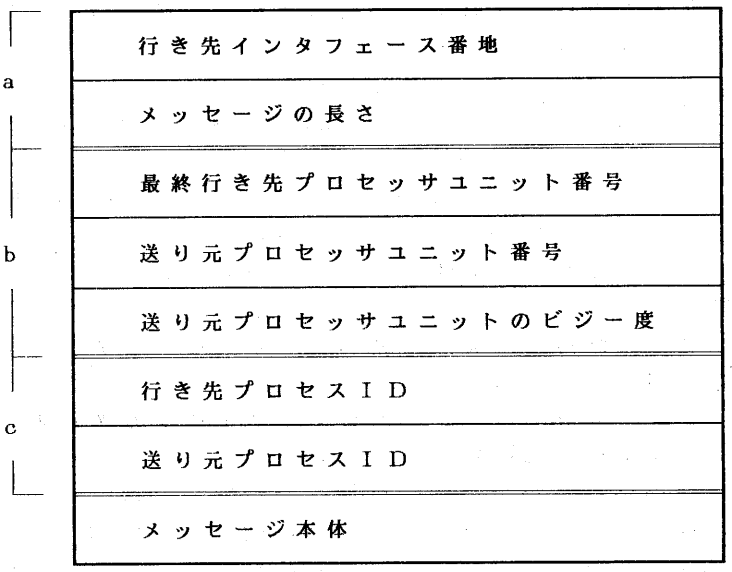

図７　ローカル通信パケットのフォーマット

ａの部分はインタフェースが解読する部分である。ｂの部分はＯＳカーネルの通信管理のために使う部分で、自分が最終行き先プロセッサユニットでない場合には中継の処理をする。送り元プロセッサユニットとはこのパケットを実際に送った（または中継をした）プロセッサユニットである。このビジー度の情報から隣接プロセッサユニットのビジー度を更新する。ｃの部分はＯＳカーネルのプロセス管理のために使われ、もし受信したプロセスがメッセージ待ちでサスペンドされていれば再開される。各プロセッサにはそのＯＳカーネルを表すプロセスＩＤがあって、プロセス間通信以外の通信に用いられる。例えば、子プロセスを生成する場合には、親プロセスから子プロセスが作られるプロセッサのＯＳカーネルに生成要求が送られ、プロセスが生成されるとそのＩＤがＯＳカーネルから親プロセスに知らされる。ローカル通信バスにパケットが送信されると、その行き先のプロセスはパケットの全てを受信して上に述べたような処理を行う。それ以

外のプロセッサユニットはパケットのうちａとｂの部分のみを受信し、隣接プロセッサのビジー度の情報を更新する。このビジー度の情報に基づき、新規生成プロセスはもっともビジー度の低いプロセッサに割り当てられる。

5. まとめ

光を使った新しいバスシステムを用いたマルチマイクロプロセッサシステムDialogについて通信の観点から述べた。光バスをマルチプロセッサシステムに使用することで共有バス型の大規模並列計算機が実現可能となる。円筒鏡による光バスには光学系の小型化などの問題があり、またホログラムによる光バスにはホログラムの効率の向上の問題や発光素子であるレーザダイオードの波長安定性の問題などまだ解決すべきものもある。しかしこれらの問題は近年急速に進んできた光素子技術およびOEICの開発などにより近い将来解決できるものと信じる。

謝辞

本研究の機会を与えて頂いた電総研白井制御部長ならびに熱心な議論をして頂いた論理システム研究室諸氏に感謝する。

参考文献

1) Gajski et al:"Ceder - A Large Scale Multiprocessor", Proc. 1983 Int. Conf. on Parallel Processing, 1983

2) Gottlieb et al: "The NYU Ultracomputer - Designing an MIMD shared Memory Parallel Computer", IEEE Trans. on Computers, vol.C-32, No.2, 1983.

3) 岡田、田島、田村、濱崎: "分散型アービタの一方式について"、情報処理学会第２３回全国大会、1981

4) 田島、鈴木、濱崎、岡田: "放送型光バスパイロットモデルの実験"、電子通信学会技術報告OQE85-176、1985

5) 鈴木、田島、濱崎、岡田、田村: "ホログラフィによる光バスとその基礎実験"、電子通信学会技術報告OQE85-175、1985

6) Okada, Tajima, Hamazaki and Tamura: "Dialog.H: A Highly Parallel Processor Based on Optical Common Bus", COMPCON 83 fall, 1983

7) Wittie: "Efficient message routing in Mega-Micro-Computer Network", Proc. of 3rd Ann. Symp. on Computer Architecture, 1976

8) 濱崎、栗田、岡田: "バス型ネットワークによる負荷分散の一方式"、昭和60年度電子通信学会情報・システム部門全国大会、1985

9) 濱崎、岡田、田島、田村: "５チャンネル、１００Mbit/s光バスの試作について"、情報処理学会第25回全国大会、1982

１０) 濱崎、岡田、田島、鈴木: "Dialog.Hのプロトタイプシステム"、情報処理学会「アーキテクチャワークショップインジャパン'84」シンポジウム、1984

１１) 濱崎、岡田: "Dialog.Hのカーネル"、情報処理学会第33回全国大会、1986

１２) Conery,: "The AND/OR Process Model for Parallel Interpretation of Logic Programs", Technical Report 204, Univ. of California Irvine, 1983