

N対Nのループシステムにおけるプロセッサ間通信効率化に関する検討

星子 隆幸 魚住 栄市 小柳津 育郎

(日本電信電話公社 横須賀電気通信研究所)

1. はじめに

ループ伝送路を介したN対Nのプロセッサ間通信におけるデータ転送方式として一般に採用されているメッセージバッファ間通信は、高トラヒック、長データ転送の環境下では送受信バッファ管理、割込み処理、データ移送等に要する通信処理ダイナミクスステップが、ループシステムの性能上のネック要因として無視し得なくなることが予想される。また、伝送路断等のN対N通信全体に影響を及ぼす障害発生時、通信への擾乱を極力回避するため早急な障害回復が望まれる。この処理をソフトウェアを介して実行すると、障害箇所切分け、ループ運用構成の選択、ループ再構成指示等の複雑な手順を経ることが必要となり、ソフト実現上の負荷が大ききばかりでなく、一連の手順に要する時間を上記擾乱が無視できる程に短縮することが極めて困難になる。

本稿は、ループ伝送路を介したN対Nのプロセッサ間通信に関し、通信処理に要するプログラムのダイナミクスステップ数低減並びに伝送路断等のN対N通信全体に影響を及ぼす障害に対する高速の障害回復を汎用性を有し経済的に実現するため、本通信制御用に設置するプロセッサ間結合装置（以下PCUと称す）の機能条件を明らかにする。

2. システム構成

HOST、FEP及びBEP等から構成されるセンタシステムにおいて、従来のチャンネル間アダプタ（CTCA等）を介して実現していたプロセッサ間通信を、多数台プロセッサ間結合時の経済化並びにN対Nの完全分散システム構築のため、ループを介して実現する。各プロセッサとループとの接続は、PCUを通して行う。システム構成概要を図1に示す。

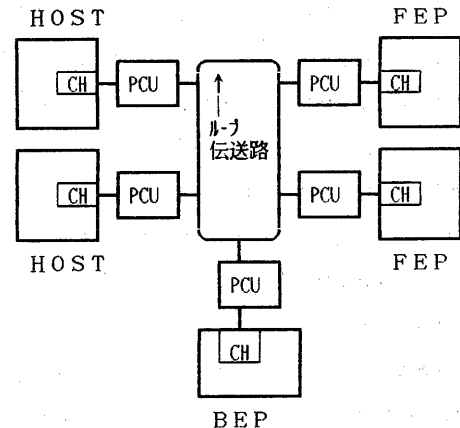


図1 システム構成概要

3. プロセッサ間通信方式

3.1 通信方式選択の指針

従来のメッセージバッファ間通信は、1対1チャンネル結合のプロセッサ間通信で採用されている複数サブチャンネルを用いた多重通信の概念は無く、通信制御プログラムが汎用的な通信機能提供のために用意している送受信バッファを介して実施される。個々の通信プロセスは通信制御プログラムが管理する送信バッファに一旦転送データを移送し、送信要求を行う。受信時は通信制御プログラムが管理する受信バッファに転送し格納されたデータを、受信バッファのアドレスを得て自らの通信領域に移送するか又は自ら指定した通信領域に通信制御プログラムから移送してもらう。この為、1章で指摘したように高トラヒック、長データ転送の環境下では送受信バッファ管理、割込み処理、データ移送等に要する通信処理ステップが、性能上のネック要因として無視し得なくなることが考えられる。

一方、1対1チャンネル結合のプロセッサ間通信で採用されている複数サブチャンネルを用いた多重通信方式では

個々の通信プロセス相互間で通信条件が規定出来るため、送受信バッファを個々の通信プロセス側で用意し、該バッファ間でデータの直接転送が可能である。また、従来通信制御プログラムで行っていた送受信バッファのフィックス/フロート制御も、同一送受信バッファを連続して使用することが可能なためフィックス化、フロート化を最適な契機で実施することが可能となる。この為、従来のメッセージバッファ間通信方式と比べて、データ移送時間が短い数百バイト程度のデータの場合でも通信処理に要するダイナミックステップ数を約1/2に低減可能である（詳細は参考文献(3) 参照）。

従来のメッセージバッファ間通信の方式イメージ及び個々の通信プロセス間直接転送の方式イメージをそれぞれ図2、図3に示す。また、後者の通信処理高速化の過程を図4に示す。

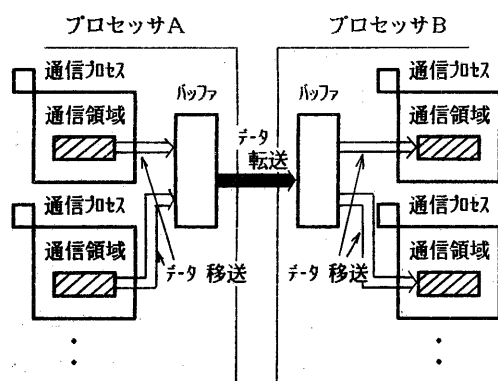


図2 メッセージバッファ間通信方式

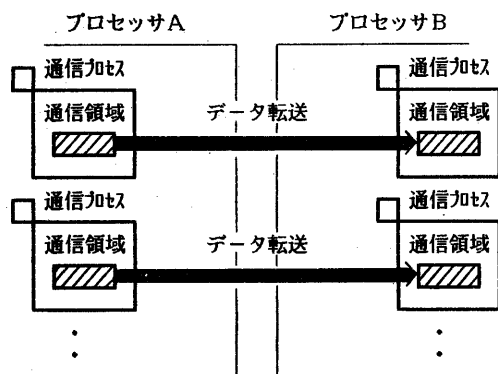


図3 通信プロセス間直接転送方式

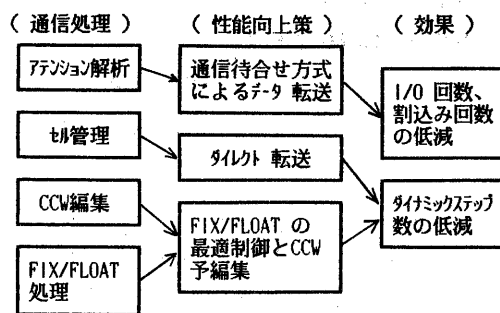


図4 通信処理高速化の過程

3.2 N対N通信高速化のポイント

1対1のチャネル結合プロセス間通信における通信高速化手法を基に、N対N通信高速化のためにPCUに具備すべき機能検討のポイントとして、以下のものが考えられる。

①PCUサブチャネルを用いて個々の通信プロセスが設定する通信バスに変動があれば、通信バスの再割付け、送受信バッファのフィックス化、フロート化等の処理が必要となり、通信処理ステップ増を伴う。従って、個々の通信プロセスが利用出来るPCUサブチャネル数は多い程良い。

②チャネル配下のPCU接続を前提に考えた場合、チャネルからPCUサブチャネルがI/Oデバイスに見え、通信相手のPCU並びにPCUサブチャネルをチャネル仕様上ダイレクトに指定する手段がない。従って、通信処理への影響の少ない通信先アドレス指定法の設定が望まれる。

③複数サブチャネルを用いた多重通信実現のためには通信起動時チャネルを早期解放することが不可欠である。また、入出力回数、割込み処理削減のためには、アテンション割込みを伴わず相手からの通信起動を待合せる通信待合せ機能サポートが不可欠である。

上記3ポイントのうち、③は1対1通信と全く同様にイミディエイト動作を行い、通信待合せを行うことの出来るコマンドサポートにより容易に実現出来る。また、

①②は通信先アドレス指定法に関係し、共通の問題として対処可能である。このため、次章でN対N通信高速化に適した通信先アドレス指定法を明確化する。

4. 通信先アドレス指定法

4.1 通信の基本概念

PCUを介したプロセッサ間通信は、図5に示す基本概念の下で実施されるものとする。

(1) 論理バス：個々の通信プロセスは、論理バスを指定して通信を行う。この為、通信に先立って論理バス開設が必要である。

(2) ポート：各プロセッサ内に論理バスを開設するための窓口としてポートを設定する。論理バスはポート名の方で意識される。

ポート名：PCUアドレス+ポート番号

(3) 実通信バス：PCU-PCU間のデータ転送は、実通信バスを指定して行われる。

実通信バス：送受信双方のPCUアドレス

+PCUサブチャネルアドレス

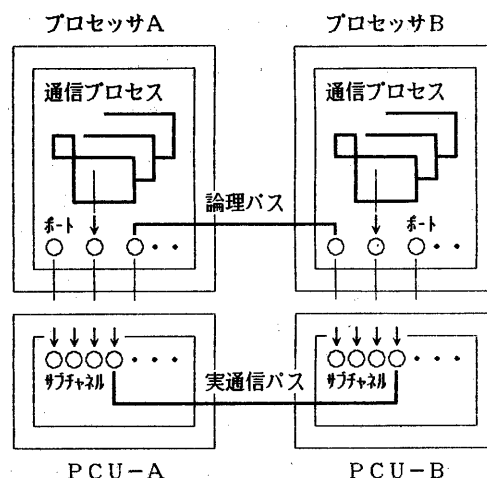


図5 通信の基本概念

4.2 論理バスから実通信バスへの変換

PCUを介したプロセッサ間通信実現のためには、論理バスの実通信バスへの変換が必要となる。PCUでのサポート機能の違いに着目して選択した3通りの変換手法を表1に示す。

本表に示す3案は、要求条件、適用領域等の違いにより優劣が異なるが、既存のセンタ構成（2章参照）への

表1 論理バスから実通信バスへの変換手法

項番	項目	変換手法			記事
		案A	案B	案C	
1	概要	通信制御プログラムで論理バスを実通信バスに変換	①実通信バスを通信制御プログラムに意識させない。同プログラムは通信相手のポート名のみを指定して入出力命令発行 ②論理バスの実通信バスへの変換はPCU-F内のバス変換バッファ（PTB）にて実施 ③PTBは変換前後の対応を記憶し、記憶容量を越える変換要求があった場合は一定のアルゴリズムにて対応のタイミク切替え実施	①②：案Bの①②と同じ ③：PTBは変換前後の対応を記憶するが、左記タイミク切替えは実施しない。通信制御プログラム側で論理バス指定個数を制限するか又はPTBの書き換え実施	※1：論理バスによる新たなPTB書き込みを可能とするため
2	PCUサポート機能	特に無し ○	①PTB設置 ②対応のタイミク切替え機能 ×	①PTB設置 ②PTBのREAD/WRITE用コマンド設置 △	
3	所要サブチャネル数	多 △	少 ○	多（案Aと同じ） △	
4	ソフト対処の容易性	論理バスの実通信バスへの変換実施 △	・論理バス対応の管理とPTB変更時の対応書き換え処理実施 ○	①論理バス指定個数の制限又は論理バスによるPTB記憶内容の消去※1 ②論理バス対応の管理 ○	

(注) ○：適用性あり △：適用性にやや問題あり ×：適用性に問題あり

適用を考慮すると以下の理由により案Cが最適である。

(1) N対N通信システムの中で他の全てのPCUアドレス、PCUサブチャンネルアドレスを意識した実通信バスの設定を全てソフトで実施するのはソフト実現上の負担が大きい。特に障害時のバス張りかえは高速性を要求されるので一層の工夫が必要となる。

(2) PCUを介した通信はPCUサブチャンネルがソフト窓口となって送受信動作が実行される。従って、論理バスとPCUサブチャンネルとの対応関係をソフトで意識するのは不可欠である。現行チャンネル仕様で入出力割込み時、割込み情報から論理バスを識別するのが困難であれば割込み処理の中でコマンドにより論理バスをPCUから読取る必要がある。この読取り動作を入出力割込み毎に実施するのはソフト通信オーバーヘッドが大きいので、論理バスと実通信バスとの対応関係に変更があった時にソフト通知があるのが最適である。

案Bは、PTBのダイナミック書き換えを伴うので上記ソフト通知が必要となり、ソフトは該通知時上記対応関係の変更処理を行わなければならない、ソフト通信オーバーヘッドの観点からの優位性は低い。従って、ハードウェア規模の大きい案B採用は困難である。

(3) HOST, FEP, BEP間論理バス数はHOST
-FEP間が特に多く、10⁴オーダ必要である。これだけのバスをPCU内で管理・記憶するのはハードウェア量的に現実的でない。従って、ソフトでPCU起動時のバス数を絞り込むのは不可欠である。

案CのPTBによる実通信バス設定の実現例を図6に示す。

4.3 アドレス指定の契機

アドレス指定契機によりプロセッサ間通信を分類すると以下の2方式となる。

(1) データグラム (DG) 方式

通信バスの設定処理をデータ転送処理と分離せず、データ転送毎に通信バスを決定する方式

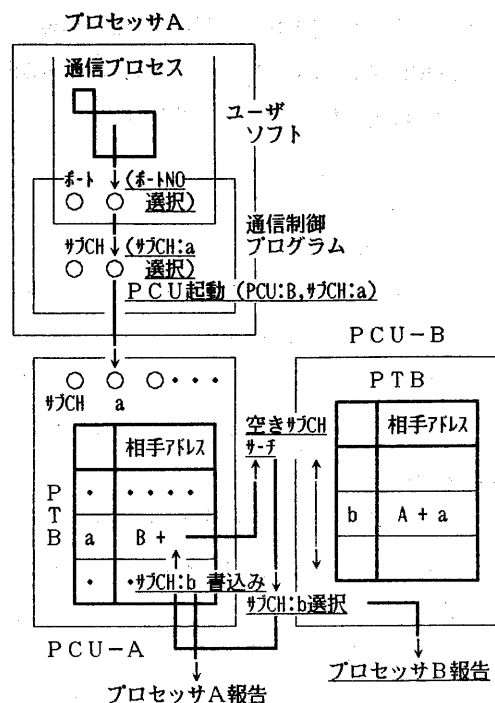


図6 案Cによる実通信バス設定動作例

(2) バーチャルサーキット (VC) 方式

通信バス設定処理をデータ転送処理と分離し独立に行う方式

上記DG方式とVC方式との比較を表2に示す。

表2 DG方式とVC方式との比較

項目	DG方式	VC方式
通信効率	①受信側PCUサブチャネルBUSYのケースがあるため、BUSY時の対処要 ②メッセージ受信時、メッセージ内アドレス情報から初めて送信元が識別できる。 ③送信側通信起動時、相手アドレス情報の指定要	①受信側PCUサブチャネルBUSYは存在しない。 ②受信側PCUサブチャネルアドレスから一意に送信元識別可能 ③送信側通信起動時、相手アドレス情報の指定不要
サブチャネル利用率	サブチャネルの長時間リザーブを必要としない(所要サブチャネル数少)	通信解放まで長時間のサブチャネルリザーブ要(所要サブチャネル数多)
PCUサポート機能	・サブチャネルBUSY解除報告または送信側でのBUSY RETRY機能	①通信バスの設定・記憶・変更・管理機能

ハードウェア量的にはD G方式の方が優れているが、複数のP C Uサブチャネルを利用した高速のプロセッサ間多重通信実現のためには、顕著な通信効率の違いを無視できない。従って、P C Uを介したプロセッサ間通信はV C方式を基本とし、P C Uサブチャネル数不足による通信バス変更が多くその為のソフト通信オーバーヘッドが問題となるシステムに関しては、アクセス頻度の低い論理バスでの通信はD G方式で実施する等の対処を行うのが最適と考えられる。

5. 同報通信との競合対策

5. 1 前提条件

- (1) 個別通信、同報通信共にV C方式で実行され受信側で個別通信と同報通信とがサブチャネルビジーにより競合することはない。

- (2) 個別通信によるデータ転送は受信バッファ空きを確認後実行される。従って、同データ転送が他の個別通信によるデータ転送と受信側で競合することはない。
- (3) 同報通信によるデータ転送は、全P C Uの受信バッファ空を確認するのはハードウェア量的に困難なので送信側から一方的に実行される。
- (4) 送信権は、1フレーム送出時点で即放棄する。
- (5) 長データは、P C Uハードが一定長のフレームに分割して転送する。
- (6) チャネルP C U間のデータ転送はバーストモードで実行され、転送終了までチャネルビジー。

5. 2 競合対策

同報通信によるデータ転送は、個別通信又は他の同報通信によるデータ転送との競合が考えられる。表3にこれらの競合対策と各対策間の利害得失を示す。

表3 同報通信との競合対策

項 目	競 合 対 策 案				
	A	B-1	B-2	C-1	C-2
概要	専用バッファ方式 ・同報通信専用の受信バッファを設置	強制終了方式 ①個別-同報間の競合は個別通信を強制終了 ②同報-同報間の競合は先行側を強制終了	同報無視方式 ①個別-同報間の競合は同報通信を無視 ②同報-同報間の競合は後発側を無視	送信開始抑止方式 (強制終了) ①同報受信時、新たなコマンド送出抑止。該状態は同報通信終了又は一定時間同報受信なしの場合自動解除 ②データ転送中を想定し、同報データは一定時間後に送出 ③個別-同報間で競合回避不成功時は個別通信を強制終了	送信開始抑止方式 (同報無視) ①②: 案C-1の①②と同じ ③: 個別-同報間で競合回避不成功時は同報通信無視
ソフトへの影響	同報の全データ長を専用バッファに退避する必要があるので、1転送のデータ長制限要	①強制終了時のソフト再試行要 ②同報受信確認はソフトで対処要	同報受信の確率がB-1より低く、ソフトでの受信確認不可欠	個別通信のデータ長に一定長以上のものがある場合、強制終了を見越したソフト再試行の対処要	個別通信のデータ長に一定長以上のものがある場合、同報受信確認はソフトで対処要
通信性能への影響	特に無し	強制終了時の再リトライによるソフト再試行のオーバーヘッド大	同報通信の頻度が高ければ、該通信未受信局への再試行オーバーヘッド大	送信開始抑止により伝送路が同報通信に占有されてしまう。	同左
付加ハードウェア/ファームウェア量	ソフトに開放したデータ長分の同報専用バッファ設置要	①個別通信強制終了制御 ②上記強制終了報告	特に無し (競合時の同報無受信制御)	①同報受信時、送信開始抑止/再開制御 ②個別通信強制終了制御	同報受信時、送信開始抑止/再開制御
	△	×	×	○	◎
	○	×	×	△	△
	×	△	◎	△	○

本表に示す5案のうち、B-1、B-2はソフト及び通信性能への影響が大きく適用困難である。案Aは同報通信の頻度、利用形態と関係するが、プロセス間通信では利用頻度が低く個別通信と同様のソフトインタフェース開放が望ましいので案A適用は困難である。ハードで100%保証出来ない以上、同報通信においてもソフトでの受信確認は必要との前提にたてばC-2のソフトへの影響は無視できる。従って、個別通信への擾乱がなく、ファームウェア制御的に複雑な強制終了が不要なC-2が最適である。

6. 障害自動回復方式

6. 1 前提条件

- (1) ループ伝送路は2重化され、一方は現用、他方は予備として使用される。
- (2) 障害波及範囲限定のため、予備バス切替え、ループバック及び各PCUでのバイパス機能サポートを行う
- (3) ループシステム高信頼化の経済的実現をねらって、各PCUは同一の障害処理機能を有し、特定の障害検出・回復専用の局は設置しない。
- (4) クロック同期は各PCUが独立に実施する。
- (5) クロック異常、送信権消滅等のループシステム全体の運用に致命的な支障を及ぼす障害はPCUハードで自動回復を行う。ループ接続状態の変更を伴う場合はデータリンクレベルでフレームを用いず該変更処理を実施する。
- (6) 送信権はトークンフレーム受信により獲得し、トークンフレーム送出により放棄する。

6. 2 基本方式

ループ伝送路断、リビータ障害等によるクロック異常並びに送信権消滅を対象に障害自動回復方式を明らかにする。

6. 2. 1 送信権消滅対策

送信権はトークンフレームによりループ上を巡回するためビット耐力が必ずしも十分でない。このため、最多

ケースとして間欠障害による送信権消滅が考えられる。

これ以外の送信権消滅ケースとしては、クロック異常、PCU論理障害等の固定障害に起因したものが考えられる。特にクロック異常に関しては、障害自動回復機能作動の契機として検出されるものと検出されないものとが存在し得る。従って、以下の2段階で送信権回復動作を実施するのが望ましい。

(第1段階) 送信権消滅検出PCUがトークンフレームを生成し、該フレームのループ一巡チェックを行う。トークンフレームが戻ってくれば送信権回復とみなして通常動作を継続する。戻ってこなければそのまま放置する。

(第2段階) 送信権消滅の発生状況を基に障害箇所の識別を行い、該箇所を予備バス切替え又はループバック切替えにより切離す。その後、第1段階の処理を行い送信権を回復する。

6. 2. 2 クロック異常対策

クロック異常は必ず送信権消滅を伴う。この為、経済性等を考慮して実現性の高い障害回復動作として大きく3つの案が考えられる。これら各案の概要並びに利害得失を表4に示す。

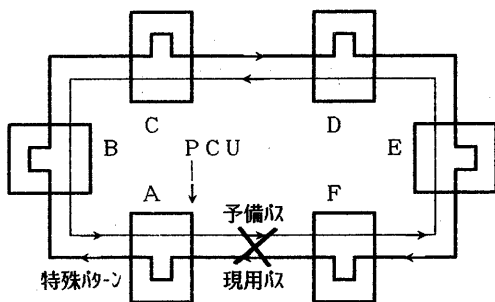
本表において、案Bは付加ハードウェアを最も必要とするためPCUに対するコスト要求との観点で適用の有無を判断する必要がある。案Aはクロック異常に伴う障害自動回復用ファームウェアを必要としない点で経済的に最も優れている。しかし、リビータ部でのクロック異常検出を前提に考えれば、案Cとの差はファームウェアの共用化をすすめることで小さくなる。従って、PCUに対するトータルの要求条件と具体的実現技術とを考慮して上記3案の採否を決定することが肝要である。

6. 3 実現例

ループ接続状態変更手順は、送信権消滅に伴うものとクロック異常に伴うものは極めて類似している。従って、送信権消滅に伴うループ接続状態変更手順を代表例として以下に示す。

表4 クロック断時の障害自動回復動作

項 目	障 害 自 動 回 復 案		
	A	B	C
概要	①クロック断検出に伴う障害自動回復動作は一切実施しない。 ②送信権消滅検出時、送信権自動回復機能(第2段階)を用いて予備冗切替え又はル-バック切替えにより障害箇所の切離しを行う。	①各PCU は2本の伝送路に同時にクロック送出とクロック抽出実施 ②クロック異常検出時、ビット耐力に優れた特殊冗-ンを検出した伝送路に対して送出 ③上記特殊冗-ン送出PCU が他方の伝送路で特殊冗-ンを検出すれば、両方の伝送路に支障のある障害とみなしてル-バック実施。他方の伝送路で特殊冗-ンの検出が無い時は、クロック異常検出側が現用冗の場合は別の特殊冗-ンで予備冗切替えを指示し、予備冗の場合はそのまま放置する。	①各PCU は現用の伝送路に対してのみクロック送出とクロック抽出実施 ②クロック異常検出時、ビット耐力に優れた特殊冗-ンを送出 ③上記特殊冗-ン送出と共に予備冗への切替えを行い、他のPCU は特殊冗-ン検出時予備冗への切替えを行う。 ④予備冗切替え後各PCU が正常クロックを受信すればそのまま放置する。一方、クロック異常検出PCU があれば、該PCU は上記特殊冗-ン送出後両方の伝送路に支障のある障害とみなしてル-バック実施。本ル-バック実施により、他方のル-バックPCU が④の論理によりル-バック条件を検出しル-バック実施
障害箇所切分け時間	全PCU の送信権監視タイマが0-ル-して初めて片側伝送路の障害箇所が識別できるので、即応性に欠ける。 △	両冗障害が瞬時且つほぼ同時に検出できる。 ◎	片冗障害は案B 同様瞬時に検出出来るが、両冗障害は予備冗切替え後AGC 等の安定化時間を加算し且つ2回の冗切替え後に初めて検出できる。 ○
ソフト/通信性能への影響	特に無し ○	特に無し ○	特に無し ○
付加ハードウェア/ファームウェア量	障害自動回復用のファームウェアを完全に一本化できる。 ◎	①両冗同時のクロック送出/抽出が必要となる。 ②障害自動回復用の論理は案C より簡単になる。 △	クロック異常に伴う伝送路切替えと送信権消滅に伴う伝送路切替え用のファームウェアが一部2本立てとなる。 ○



T2オ-バ-

図7 送信権自動回復方式(その1)

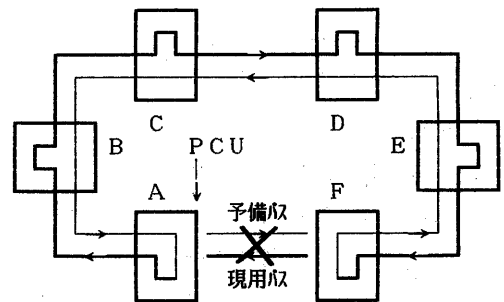
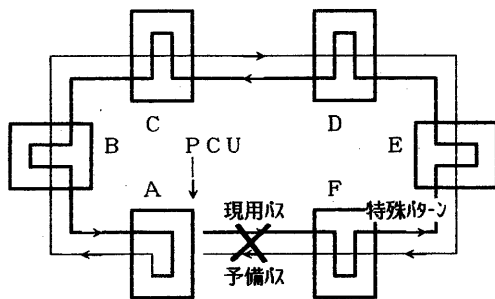


図9 送信権自動回復方式(その3)



T2オ-バ-

図8 送信権自動回復方式(その2)

＜障害発生状況＞

PCU-A～Fで構成されるル-ブシステムにおいてPCU-AとPCU-Fとの間の両伝送路異常。

＜PCU機能条件＞

①各PCUは送信権監視用のタイマ(T1&T2)を有し、以下の条件を満足するタイマ値が設定される。

$$T1 = \alpha + \beta \cdot n \quad (n: \text{PCUアドレス}, \beta > \text{フレームのル-ブ一周時間})$$

$T2 > \max. T1$

②各PCUはT1オーバフロー時、自PCU宛のトークンフレームを送出し、間欠障害による送信権回復を試みる。一方、トークンフレームを検出したPCUはT1、T2をリセット&スタートする。なお、上記T1オーバフローを検出したPCUでT2リセットは行わない。

③各PCUは現用バスに対してのみクロック送出／抽出を行う。

＜ループ接続状態変更手順＞

(1; 図7参照) A以外のPCUは、A～EでのT1オーバフローにより最低1回のトークンフレームを受信する。このため、T2オーバフローを検出するのはAだけである。T2オーバフロー時、Aは現用バスに対して特殊パターンを送出する。

(2; 図8参照) 特殊パターンを受信したB～Fは現用バスを予備系に切替えてT1、T2をリセット&スタートする。Aは特殊パターン送出後、同様にバス切替えを行ってT1、T2をリセットする。その後、自PCU宛

(3) Aは自PCU宛のトークンフレームを送出しT1、T2をスタートする。上記自PCU宛トークンフレームを受信しない場合は両バス障害とみなしてループバック状態に切替わる。

(4) (1)と同様の理由により、今度はFでT2オーバフローを検出する。このため、上記(1)～(3)の動作をFからの特殊パターンを契機に実行する。

(5; 図9参照) AとF両者がループバック状態に切替わると、その後最初にT1オーバフローを検出したPCUが発行する自PCU宛トークンフレームが送信元で受信されるので、送信権回復とみなして運用を再開する。

7. おわりに

ループ伝送路を介したN対Nのプロセッサ間通信において、通信処理及び障害処理に要するソフトウェアのダイナミックステップ数低減及びソフトウェア開発負担の軽減を可能とするプロセッサ間結合装置の機能条件を明らかにした。本機能条件適用により、1対1チャンネル

結合のプロセッサ間通信で採用されている複数サブチャネルを用いた多重通信方式と同等のソフトウェア通信効率を、N対N通信で容易に実現可能である。

謝辞

本検討にあたり有意義なご指導、ご助言を頂いた関係各位に深謝します。

参考文献

- (1) 中野、森：“疎結合計算機システムにおける高速計算機間通信方式”、情処第32回計算機アーキテクチャ研究会、1981。
- (2) 小柳津他：“高速の光ファイバループを用いたプロセッサ間結合方式に関する検討”、情処分散処理システム研究会23-5、1984。
- (3) 星子他：“共通バスを介したプロセッサ間通信効率化の検討”、情処第24回全国大会6H-6、1982
- (4) 木村他：“リング状ネットワークシステム高信頼化に関する一検討”、情処第26回全国大会3G-9、1983。
- (5) 魚住他：“トークンリング方式における同報通信の競合対策について”、情処第26回全国大会3G-10、1983。