

デジタル電子交換機のマルチプロセッサ制御方式

堀 好徳、水沢 純一、小川 聡、金 克能
(日本電信電話公社 武蔵野電気通信研究所)

1. まえがき わが国の電子交換機分野においては、単一プロセッサによる集中制御方式が主流を占めているが、通話路のデジタル化を契機として、制御系の分散化をはかったマルチプロセッサ制御方式の電子交換機の研究を進めている。ここでは、その試作をベースに、交換制御という特殊目的のシステムにおける機能分担、負荷分担、及びプロセッサ間通信について述べる。

マルチプロセッサ制御方式の利点としては、次の項目があげられる。

- (1) 交換局の処理規模に応じた制御系のビルディングブロック増設による経済性、及び同一アーキテクチャによる適用領域の広域化
- (2) 複数プロセッサで負荷分担を行うことによる信頼性の向上
- (3) 機能分担のため1プロセッサの機能が単純化することによるプログラムの簡明化
- (4) 部品技術の進歩に追階できる分離性のよいシステム構成の追求(部分的な更改)

一方、機能分散においては、プロセッサ間通信のオーバーヘッドにより、構成要素プロセッサの処理能力を単純に足した値の能力にはならない。従って、オーバーヘッド削減のため、プロセッサ間の通信回数の削減や、効率的な通信処理が必要である。

2. システム構成

ここでは、マルチプロセッサ制御電子交換機として、次のような構成を前提として話を進める。(図1参照)(文献1)

- (1) 制御系の構成は、通話路系を制御するネットワークプロセッサ(NP)と呼接続を制御する呼制御プロセッサ(CP)の2階層構成とする。NPとCPは、機能を分担する。
- (2) NPは、実時間性が厳しいので負荷分配のオーバーヘッドやNP間での情報交換を避けるため、回線群対応に負荷を分担させる。
- (3) CPは、負荷が均等に分配されることを狙い、回線との対応をとらないで、各NPからどのCPへも負荷が分散できる。
- (4) プロセッサ間通信制御を専用に行う装置(IPC)を設ける。IPCは、バスの競合整理、通信のやりとり、及び自律的にプロセッサのメモリへの読出し、書込みを行う機能をもつ。

3. 機能分担

呼の接続制御(呼処理)に対するNPとCPの機能分担について、考察する。呼処理のNPへの機能分担の程度を、図2の4種類に整理し、通信回数を比較する。案Aは、短周期、単機能の回線の變化識別及び選択数字の送受信機能のみをNPに分担したもの。案Bは、案Aに対し、NPの自律処理をさらに向上させたもの。案Cは、案Bに対し、対向局への受信機能確認と数字送出手の自律的実行を追加したもの。案Dは、リソース(バス、バンク)管理、呼の接続制御までNPで

分担し、NP処理を高度化したものである。図3に、各案に対応したNPとCPでの量的な分担率を示す。プログラムのステップ数と、発呼から切断までの1コールのダイナミックステップ数に示す。この4案について、1コールの通信回数を同じく図3に示す。案Aは、NPの自律処理が少なく、アモセッサ間の通信量は多くなる。案Bは、案AでのNPに対し、全数字を蓄積し、1回でCPに送る機能や、ON、OFF信号を、応答、切断等と判断する分析機能を追加することにより通信回数が減少する。案Cは、数字送出の前の対向交換局の受信機能確認を数字送出前に自律的に実行すること等により、さらに通信回数が減少する。案Dの分担になると、リソース管理をNPで行うので、NP間にまたがる接続処理では、通信回数が増加する。従って、案Cの分担が最も初率的である。

4. 負荷分散法
ここでは、NPからCPへの負荷の分散を行なう場合に、負荷の均等な分配を実現する手段について考察する。(文献2.3)
負荷の均等分配の容易性、メモリ量の観点から、CPで受け付ける負荷の単位は、図4に示す発呼、応答

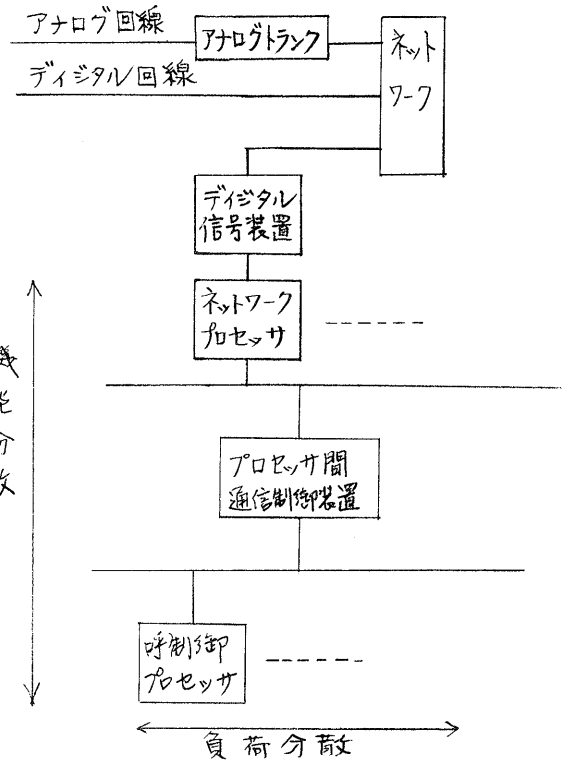


図1. システム構成

| | 案 A | 案 B | 案 C | 案 D |
|-------------------|--------------|-------|-----|-----|
| NP の 機 能 | 回線状態の変化検出 | | | |
| | 1数字の受信 | 全数字受信 | | |
| | 全数字送出 | | | |
| | 監視信号の送出 | | | |
| | オーダ順序制御 | | | |
| | 状態監視の制御 | | | |
| | パス制御 | | | |
| | 局間信号のシーケンス制御 | | | |
| | リソースの管理 | | | |
| | 自NW内の接続制御 | | | |

図2. 機能分担の分類

などの信号単位とし、ひとつの呼に対する処理を、ひとつのCPにくり返さず、信号単位に各CPに分配する。したがってひとつの呼の発呼から終話までの処理に複数CPが関与する。負荷分散の実現法としては、NPが負荷を分散する、CPが負荷を持つNPを深することも考えられるが、IPCは、負荷の少ないCPを知り、NPからの負荷を分配する方式とする。上記の発呼、応答などの信号単位対応の内部処理量は均等ではないので、IPCがイベント数によって分配しても、各CPの負荷は、均等とはならない。従って、IPCは、各プロセッサの負荷状態を知る必要がある。このため、各CPは、信号単位対応の処理が終了し、無負荷状態になったときに、負荷要求をIPCに対して出す。この負荷要求により、IPCはプロセッサの無負荷状態を知り、転送先CPを決定する。

これらの処理は、IPCとCP内のプロセッサ間通信制御プログラムによって、図5のように実現する。

- (1) 信号単位対応の処理を終了したCPは、IPCに対し、負荷要求を出す。

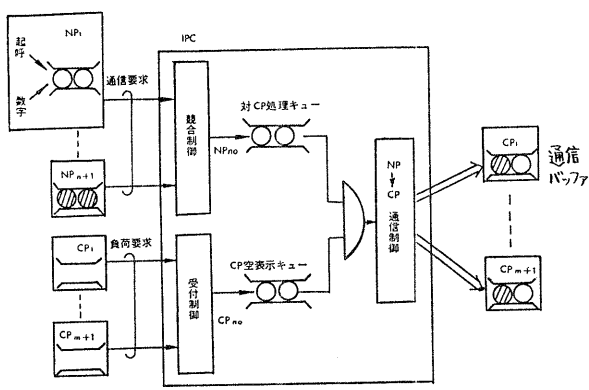


図5. 負荷分散方式

(注) S:スタティックステップ数分担率
D:ダイナミックステップ数分担率

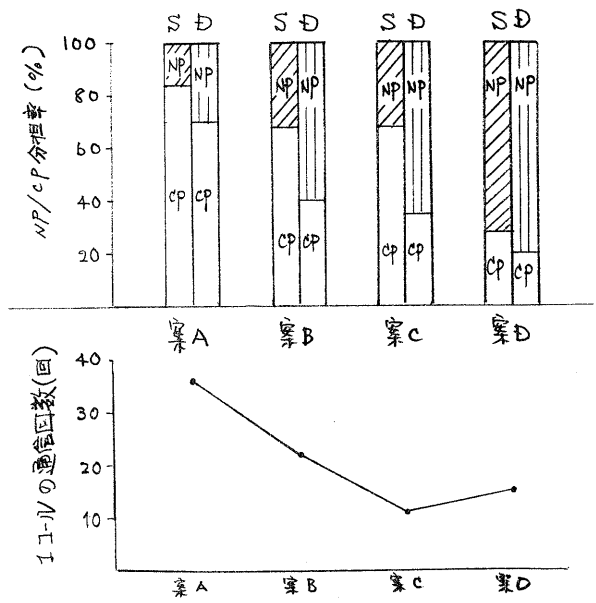


図3. 機能分担率と通信回数

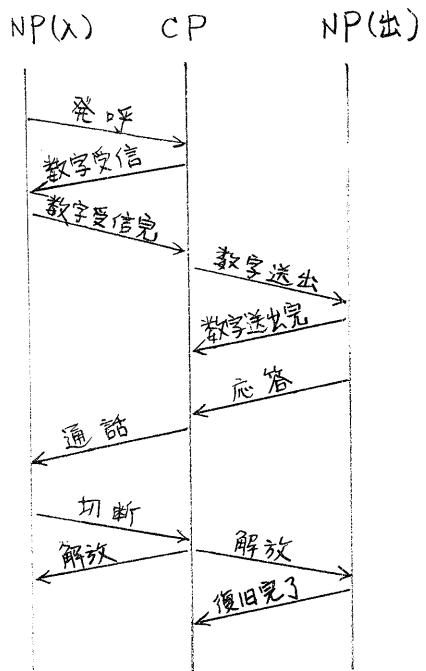


図4. 呼接続シーケンス

IPCでは、これを空表示キューにつなぐ。

- (2) NPから通信要求が起ると、IPCは、これを対CP処理キューにつなぐ。
- (3) 上記両キューがともに空でないとき、両キューから1個ずつ取り出し、CP空表示キューにつなぐ。つなぐたCPに、NPからの通信情報を転送する。

5. 通信処理

5.1 通信規約

プロセッサ内の処理は、大きくみて、呼処理(CP)、運用処理(AP)、障害処理及び実行管理処理(OS)に分割できるので、図6に示す階層構成をとり、通信処理の単純化、ユーザ間インタフェースの論理化を行う。

プロセッサ間通信制御プログラム(PC)は、IPCを介して、プロセッサ間通信情報の送受を行なう。

各信号分析プログラムは、CP、AP、OS対応に設置し、通信機能と内部プログラムの構造上の分離を明確にしている。

これらの処理の階層化に伴い、通信規約を設ける。プロセッサ間通信制御レベルでは、CID(通信識別子)において、各プロセッサの信号分析プログラムを識別する。表1にその内容例を示す。信号分析プログラムレベルでは、SID(信号識別子)において、目的となる内部処理プログラムを決定する。表2にその一例を示す。

5.2 通信処理

IPCは、プロセッサが動作中に自律的にメモリへの書き込み、読み出しを行う機能を有するため、PCは、信号を受信バッファから、処理待ちキューに転

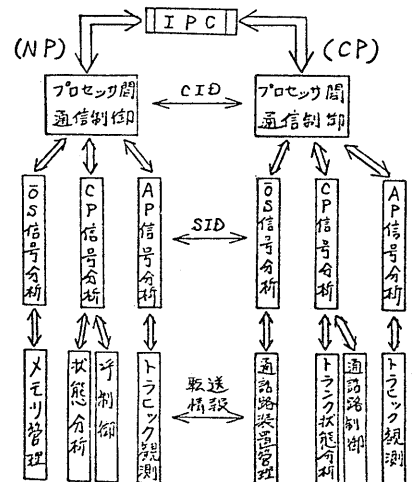


図6. 階層構成

表1. CID種別例

| CID | 信号種別 |
|-----|-------------------|
| NCP | NP側 呼処理用信号 (CP) |
| NAP | NP側 運用管理用信号 (AP) |
| NOS | NP側 交換DC用信号 (OS) |
| CCP | CP側 呼処理用信号 (CP) |
| CAP | CP側 運用管理用信号 (AP) |
| COS | CP側 交換DC用信号 (OS) |
| MAP | MCP側 運用管理用信号 (AP) |
| MOS | MCP側 交換DC用信号 (OS) |

表2. SIDの一例

| 通信経路 | CID | SID | |
|-----------------|-----|-----|--------|
| | | 記号 | 意味 |
| NP → PC → CP | CCP | ST | 起動 |
| | | ID | 解放 |
| | | RIG | 呼出し |
| | | AD | アドレス送信 |
| | | AC | 応答 |

送り、受信バッファを空にして、IPCに対し、負荷要求(空表示オーダ)を出す。NPからの通信イベントを受信できる状態にして、すでに受信した通信イベント対応の内部処理を行なう。内部処理がすべて終了したときに、受信バッファを調べ、バッファに入っている通信イベントがあれば、上記処理をくり返し行なう。

5.3. 通信誤りに対する処理

ハードウェア障害を検出した時は、ソフトウェア処理とは、プロセス間通信制御プログラムでは、再送処理を行わず、呼処理プログラム側で一部再送処理を行なっている。その理由は、通信誤りの確率が低い、処理能力向上をはかりたい、また、合理性検査によって対策が立てられていることによる。

合理性検査とは、NPが最初に発呼信号をCPに送った後、CPから数字受信を指示する信号を受けとるが、例えば、回線番号情報が抜けていた場合、発呼してはいない空き回線に対して数字受信の指示が与えられるので、これを、「不合理」としてその信号を廃棄する。

もし、何らかの原因で、上記のことが起り、数字受信信号が廃棄された場合、もとの実際に発呼した回線は、保留したまま、数字受信信号を待ち続けることになる。これをさけるため、待ち時間がタイムアウトした場合、発呼信号をCPへ再送することとする。再送処理を行う信号は、発呼、切断、解放の3種あり、図8に示す。その他の場合は、最終的には、切断信号によって解放状態に遷移する。例として、解放信号の再送

処理を回9に示す。CPでは、トランク状態を復旧完了信号待ちとし、タイミング登録を行うと同時に、解放信号を送出する。復旧完了信号を受信せず、タイムアウトした時は、解放信号を再送出する。復旧完了信号を受信した時は、トランク

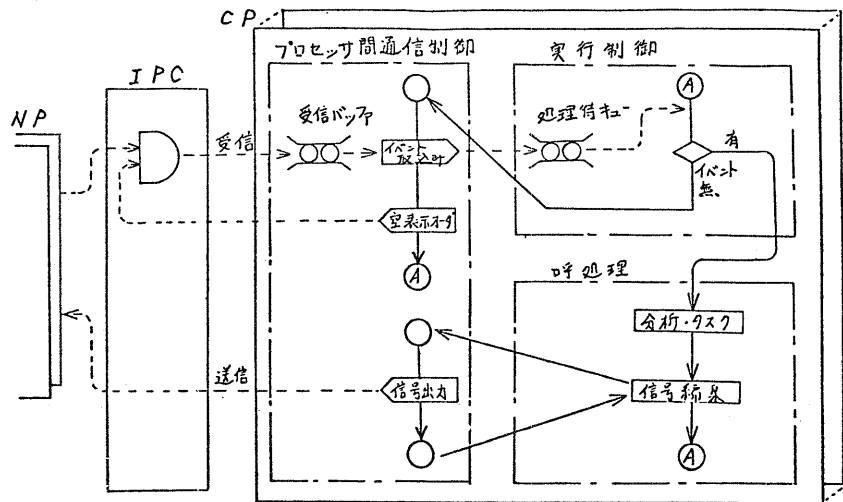


図7. 通信処理概要図

状態を空とし、タイマをリセットする。一方、NPでは、解放信号を受信後、トランクを空状態とするが、空きトランクに対し、解放信号を受けた場合は放置せず、復旧完了信号をCPに対して送出することとする。(文献4)

6. ソフトウェアの設計

プロセッサ間の機能の独立性に対する評価尺度について考察する。指標として、G.J. Myersのモジュール強度、モジュール結合度などの適用が考えられるが、ここでは著者が提案する結合密度を適用し、ソフトウェアの分割、及び分割したプログラム間の独立性の尺度とし、プログラム構造改良のひとつの目安としたい。

結合密度の概念は、次のようなものである。AとBの2つのグループにプログラムがそれぞれ、 m 、 n 個あるとする。Aのすべてのプログラムから、Bのすべてのプログラムへの起動があるとすれば、その本数は $m \cdot n$ 本である。ソフトウェアの設計を行なって、起動本数が l 本だったとすると、そのときの結合密度 D は、 $D = l / m \cdot n$ で得られる。

本プログラムシステムへの適用では、 l の値を次のようなものとする。1入口、1出口の procedure 間での caller と callee の関係とし、A、B各グループ間での procedure の組合せの数をインタフェース本数 l とする。従って、図10に示すように、procedure x から、procedure y を何回呼び出しているか、同一種の呼びと考えられるので、インタフェース本数は1本とする。

次に m 及び n について考察する。 m 及び n を、procedure の数とすると、A及びB内で procedure を細分化すると、インタフェース本数は、変わらず、procedure 数が増加するので、見かけ上結合密度が低くなり、正しい尺度になりえない。従

って、プログラムの任意の場所へのジャンプが可能であることから、プログラム個数をプログラム規模とする。プログラム規模は、概ね、機能の大きさを表わしているとする。結合密度の適用

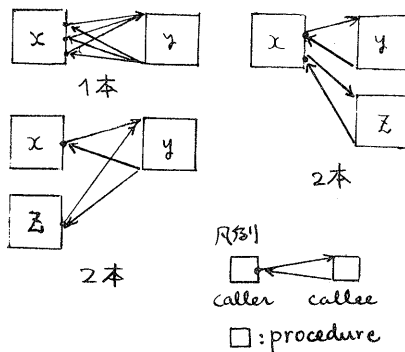


図10. インタフェース本数

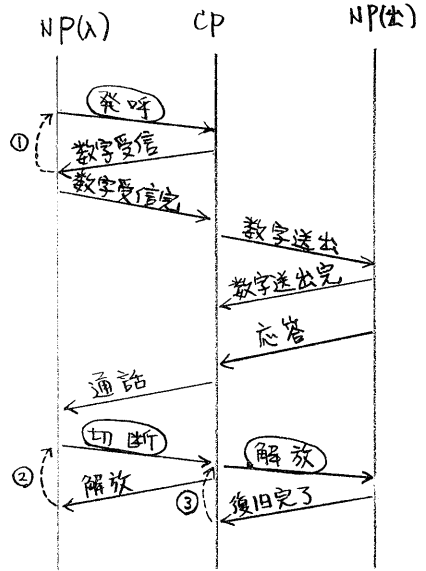


図8. 再送処理

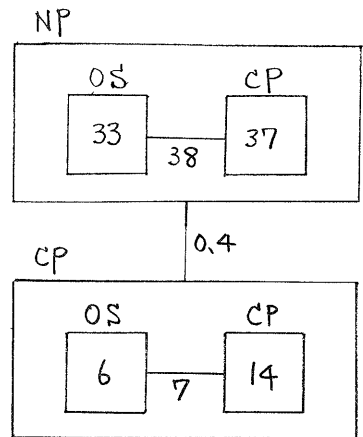


図11 結合密度(本数/ $kstep^2$)

例を図11に示す。ここでは、 $A \leftrightarrow B$ 双方向の結合密度を $A \rightarrow B$ 、 $B \rightarrow A$ の和としている。また、同一グループ内の結合密度についても、 $A \rightarrow A$ として計算したものである。NP側は、OS内、CP内、OSとCP間のいずれも結合密度が高い。一方、CP側は、NP側と比べてかなり結合密度が低い。

プロセッサ間のインタフェース本数は、本システムの場合、5節で述べたS

Iの数が相当すると思われる。その結果を、図11に示すが、低い値が得られ、NPとCPの機能分割の妥当性を示す。

最後にインタフェース本数の削減例を図12に示す。案1では、まず、トランク群が実装されており、未開塞であることを確認したのうち、使用トランク数を聞く方式なので、プログラムA、B間での本インタフェースがある。案2では、実装判定及び開塞判定を、プロ

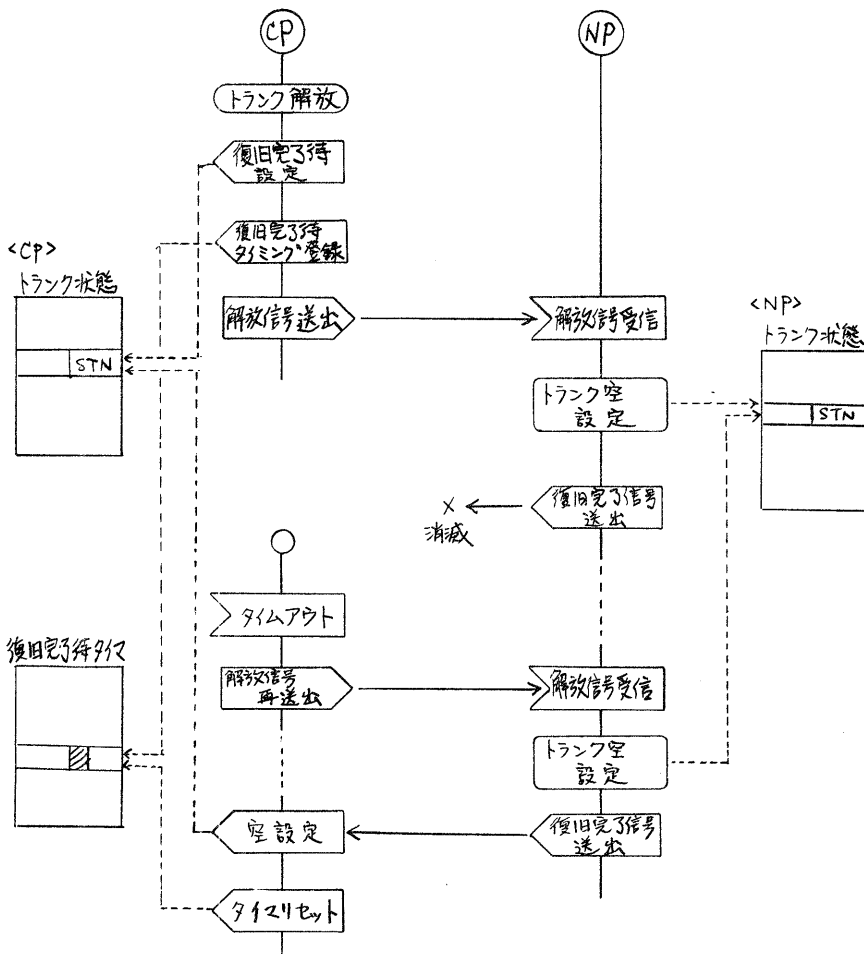


図9. 解放信号の再送処理

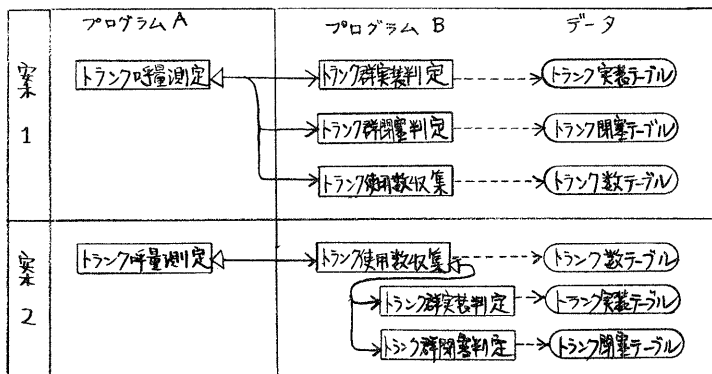


図12. インタフェース本数の削減

グラムB側のトランク使用数収集プログラムを行うため、インタフェースは1本となる。このことは、プログラムAとBの分離作成上、望ましいと考えられる。

7. あとがき

本システムでは、効率重視のため、通信処理が個別対応になっている面があるが、今後は、分散形マシンにおける通信インタフェース(または手順)の高度化(抽象化)の検討が必要となろう。また、ソフトウェアの設計法としては、各プロセッサへの機能配分に対する指標や、手順の確立が望まれる。

謝辞

本分散方式について、共同研究を行なった日本電気株式会社、株式会社日立製作所、沖電気工業株式会社、富士通株式会社の関係各位、IPCによる通信方式を御検討いただいた武蔵野研究所処理装置研究室の関係各位、及び同交換プログラム研究室の担当者各位に感謝します。

参考文献

- (1) 五嶋、池田、井上：「デジタル電話市外系システムDTS-1の方式構成」電気通信研究所研究実用化報告VOL.28, NO.7 (1979)
- (2) 堀、川崎、金：「複数プロセッサ制御のプログラム構成」同上
- (3) 甲斐、岡島他：「マルチプロセッサ電子交換機におけるプロセッサ間通信制御方式」電子通信学会交換研究会 SE79-93
- (4) 金、遠藤、高山：「マルチプロセッサ制御交換機におけるプロセッサ間信号転送の高信頼性の検討」電子通信学会56年度全国大会予定
- (5) C. J. Myers : 「Reliable Software Through Composite Design」New York Petrocelli/Charter (1975)
- (6) 小川、田中、水沢：「電子交換用マルチプロセッサ制御プログラムの作成技術と評価」電子通信学会交換研究会 SE80-16
- (7) 堀、小川、金：「電子交換機におけるマルチプロセッサ方式の評価」電子通信学会55年度通信部門全国大会