

通信制御におけるプロトコルの記述とプログラム化に関する検討

入江稔純 野口正一 高橋薫 菊老原義彦 川台英俊
(東北大学) (筑波大学) (電総研)

1. はじめに

従来、計算機間通信制御プロトコルの記述法について、種々の立場から、各種方式が提案されてきており、大別して、下記の3つの立場からアプローチされている。

- (1)、プロトコル設計における、制御の正当性を検証する。
- (2)、標準的記法による統一的なドキュメント化のため。
- (3)、システムの作成者へ詳しい制御機能を提供する。

その1のものは、Petri-Net⁽¹⁾ などにより代表されるグラフ系言語系⁽²⁾⁽³⁾⁽⁴⁾ によるものが多く、その2は、ALGOL、PL/I などの高級言語系⁽⁵⁾ によるもの、その3は、各種図、表を含めた自然語によるものが多い。⁽⁶⁾⁽⁷⁾

ここで、計算機通信網などのように、システム作成者のシステム環境がそれぞれ異なるような場合、システム作成者として、重要な事は、プログラムの作成、デバッグなどを行う点から、制御の流れが把握しやすく、かつ、正確な制御機能が記述されている事である。

この点から上記立場における記述法は、それぞれ次の問題を各々下あろう。

その1は、設計を目的としているため、正確、かつ厳密な記法ではあるが、実際のシステムに全て適用できず、複雑で膨大なものとなり、システム作成者として、理解が又変わつてしまつてしまう。

複雑化する記述をモジュールにより単純化する考之⁽²⁾を提案されているが、これにより必ずしも正当性の検証を目的としている。

その2は、標準的記法として成功しているが、制御の流れの把握、つまり通信相手との動作の動きが、把握しにくく、又、機能の定義が、詳しくあざまつ、狭くなる場合がある。

その3は、種々存在するが、一般に、厳密に記述されているものの、全体的な制御の流れの把握がむづかしく、全体的な流れの把握が容易なのは、厳密さに欠ける場合がある。又一般に標準性に欠ける場合がある。

本研究は、異なるシステム環境におけるシステム作成者へ提供可能な記述法について、状態遷移図を用いた記述法の性能を検討したので、述べるものであり。

なお、本方式はT-NET⁽⁴⁾の応用新サイドで、システム作成に実際に適用しており、本論文では、IMP、NCPを例として述べる。

2. 通信制御システムの階層性

一般に通信制御システムを、すべて1層に1つの方法で細部まで、おしめて記述するより、システムの性能に依りて分割を行うシステム全体の、各分割部の集合として示す(マクロ記述)、各分割部で細部機能を示す(ミクロ記述)とする合理的と云えるであろう。

ここで、通信制御システム(計算機間における)の性能(特に階層性について)検討を進めよう。

2-1. 計算機間におけるプロトコル階層

計算機網アーキテクチャ I におよぶ 70 以上の階層は、Fig. 1 に示すように、ULP (User Level Process), NCP (Network Control Process), IMP (Interface Message Process) に階層化される。これは、それぞれ次の位置づけで、機能をもち、

- IMP; データ通信回線を効率的に使用し、かつ計算機網におよぶ仮想計算機システムの単位をもち HOS T 計算機に対し、信頼性が高く透過的なデータ通信網を提供する。
- NCP; HOS T 計算機間通信の論理的通信路 (Link) の設定、切断および、この Link におよぶフロー制御を行う。計算機網におよぶ通信は、手続的により、ULP のインターフェイス間通信にあり、Link のインターフェイス間通信路である。
- ULP; 計算機網において、1 HOS T 計算機に 1 つの OS が存在し、HOS T におよぶ計算機網に対するサービス処理の単位が、ULP であり、これはあべて OS の管理下にある。通信を行う ULP は、ある HOS T の S-ULP (あるいは U-ULP) と別な HOS T の U-ULP (あるいは S-ULP) との対である。

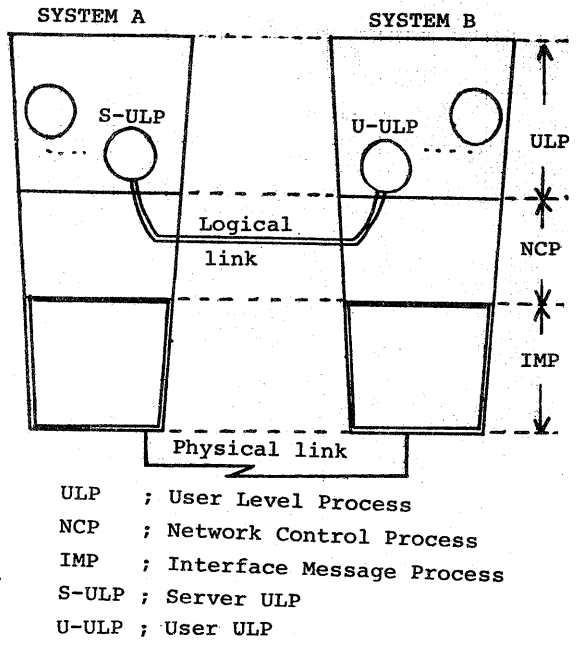


Fig.1 Network Control Hierarchy

上記3階層は、それぞれ、計算機内で1つのプロセスを形成していき、IMP, NCPは、HOS T計算機の網に対する特別な制御プロセスであり、特にOSからみれば、システムレベルのプロセスと言えよう。又、上記は機能的な面からみれば階層下であるが、扱われるべき、伝送情報単位からみても同様に階層化できる (Fig. 1)。

2-2. 副階層 (IMPを中心として)

前記で、計算機網におよぶ70以上の階層を、3つに分類したが、それぞれの機能は大きく、かつ、広範であり、適当にブロッック化し、ブロッック間の関係で全体の制御の流れを示し、細部をブロッック毎に記述する事は合理的と言えよう。

ここでは、特に複合的な機能をもちIMPを中心として、さらに階層化の考えを進めよう。一般に、分散形態網におよぶIMPは下記a. - g. の機能を有する。

- a. データ通信回線の制御
- b. 伝送制御手順の制御
- c. エラー制御
- d. 蓄積交換制御
- e. 経路選択制御

- f. 網の監視
- g. HDS T計算機とのインターフェイス

2-1. 示したIMPの階層の位置付けから、さらにデータ通信回線が、下にあると考えると、IMPを通信制御システムの1ポートとすると、IMPはFig. 3に示すような4つの通信入/出力対を有すると考えられる。

つまり、
 () の縦は上、下階層のNCP、データ通信回線部
 () の横は、同階層のIMPと、ローカルシステムのI/O (あるいはモニタ)
 ここでIMPポートの場合、

実際に入/出力事象を交換するのは縦の関係 (NCP、データ通信回線部) とローカルシステムのI/Oとの間であり、同層間では、直接的な入/出力事象の交換を行わず、プロトコルに従って処理される内容を示す。

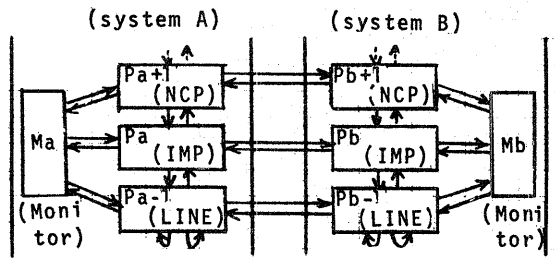


Fig.2 Model of Communication Control Process

IMPにおける副階層化を考えると、上記の縦の関係から、同層間の処理内容について、考え

ると、下記の4つのカテゴリーによる副階層化が考えられる。

- (1) レベル1; データ通信回線の制御機能レベル
- (2) レベル2; 隣接IMPとの1対1通信制御機能レベル
- (3) レベル3; 通信の網化制御機能レベル
- (4) レベル4; NCPとのインターフェイス機能レベル

又、このカテゴリーを伝送情報単位を基準とした場合 (T-NETを参考とした場合) 各レベルで処理される伝送情報単位と、階層、副階層間の通信単位との関係をFig. 3に示し、IMP機能a. ~ g. を(1) ~ (4)のカテゴリーに分類したものをTable. 1に、又T-NETのIMPプロトコルを例とした副階層化をFig. 4に示す。

Level	Functions
Level 1	(a)
Level 2	(a), (b), (c)
Level 3	(d), (e), (f)
Level 4	(g)

Table.1 Functions at each level

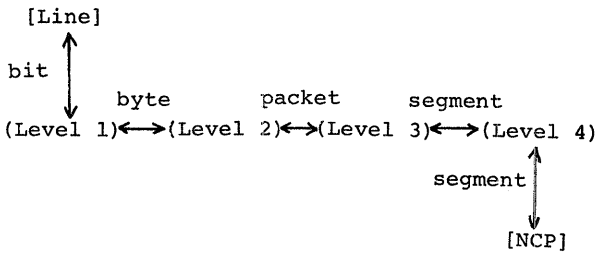


Fig.3 Information unit inter Level

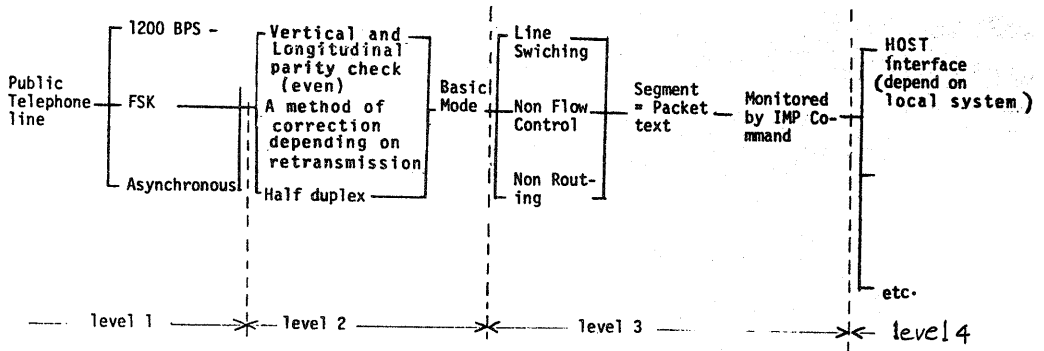


Fig.4 Functional Structure of IMP at T-NET

この考え方をNCPに適用した場合、IMPインターフェイス、論理通信路の制御（フロー制御も含め）、ULPインターフェイスの3レベルと考えられる。

3. マクロ記述とミクロ記述

本章において、通信制御システム（計算機網における）の階層性から、ポートの分割を試みる。

ここでは、この分割をもとにし、ポートの記述を全体的な制御の流れの把握を目的としたマクロ記述と、細部表現を行うためのミクロ記述について、IMPを中心に述べる。

3-1. マクロ記述

マクロ記述の目的は、通信制御システムの機能的な全体の流れを把握し、かつ送、受信時の相手との機能的な動作の位置付けを明示する事である。

この手法として、制御プロセスを機能ブロックの集合としてとらえるため、前述の副階層分割をもとにして機能をブロック化する。

ブロック化のため、まず、各副階層で制御すべき対象をまとめる。

例として、Fig. 3, Fig. 4 をもとにありて下記のように示せる。

- レベル 1 ; 回線部ポートレベルはNTT規格に従う。
 - レベル 2 ;
 - NCU (AA型) 制御 (destination への回線接続, 被接続)
 - モデム制御 (半二重通信)
 - 伝送制御手順制御 (パケットを単位とする)
 - レベル 3 ;
 - destination の選択, 検出.
 - パケットとセグメントの変換.
 - destination IMP とのステータス情報の交換 (IMP コマンドの交換)
 - レベル 4 ; NCP との通信 (HOST I/O Queue を介する)
- 機能ブロックを上記対象をもとに以下に示すようにまとめる。
- NCU CNT (NCU 制御)
 - MDM CNT (モデム制御)
 - PKT CNT (伝送制御手順)
 - PKT (セグメント → パケット)
 - DPKT (パケット → セグメント)
 - CQM CNT (ステータス情報の交換)
 - DILMAN (destination 情報の管理)

これらの情報の流し、生起する事象の関係を図示したものが、マクロ記述である。記述の定数以下記の通りである。

- 内の機能ブロック
- 矢印のゲートの流し
- 細矢印の事象の方向

○ / の左側は機能ブロックに生起した入力事象であり右側は処理結果の出力事象を示す。

Fig. 5に Fig. 3, 4 のマクロ記述を示す。

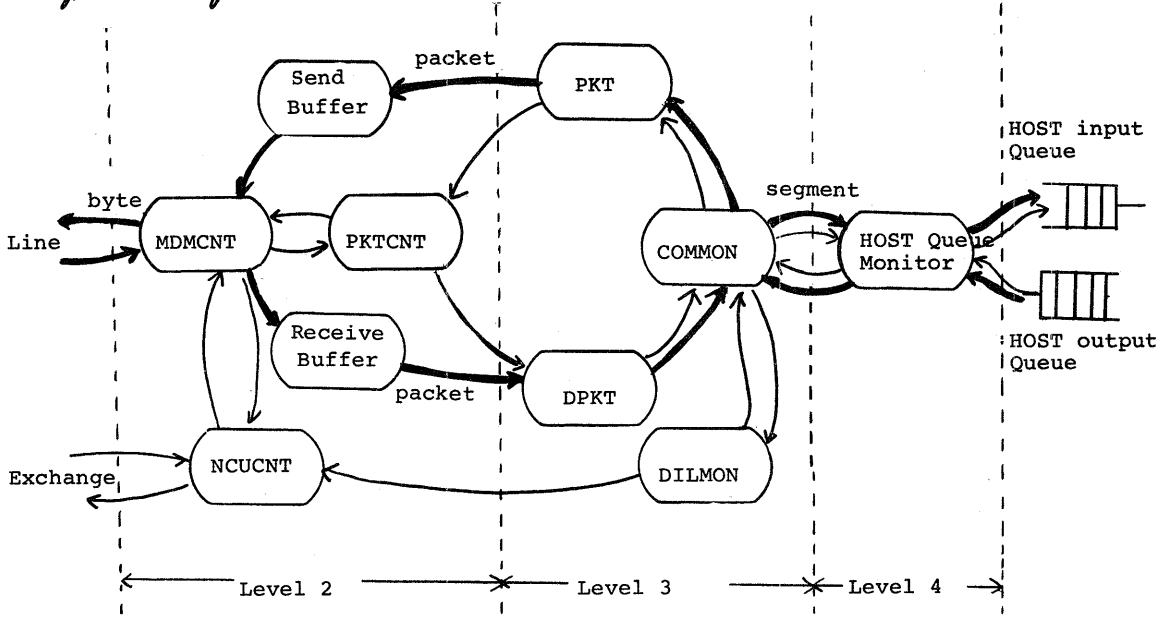


Fig.5 Macro Description of T-NET IMP

3-2. ミクロ記述

3-2.1. 各機能ブロックについて、ポートの細部機能を示すのがミクロ記述である。

ここでは、制御機能を正確にかつ例外規定をば、適格に記述する必要があり。各機能ブロックをその状態遷移機構と考之、理解を容易にするため、図により記述(状態遷移図)あり。

この記述の定数以下記の通り。

- 節点(Node)内の状態を示し、何等計算処理の意味はない。
- / の左側はその状態遷移機構に対する入力事象であり右側は入力に対する処理結果としての出力事象を意味する。
- 記号Nは入力事象の場合、無条件入力(CPUの実行権を与之する)を示し、出力の場合、no-operation。

○ 論理演算記号は下記の通り

A AND ; &

B OR ; or

○) - マルチポートを1つの図にするため、例外規定は行記す。

○ 入出力事象の意味は、Tableに記す。

記述例の4章で示す。

4. 状態遷移図記述(マイクロ記述)の例

本章では 3 章で述べた記述の基本的な考えをもとに実際の例を示す

(1) IMP K および子例

IMP K および子例として、モデムの本多重通信制御を行う MDMCNT K について Fig. 6 と Table. 2 を示す。

Event symbols	
DR on	; Line events
RS on	
RS off	
CS on	
CS off	
CD on	
CD off	
ER off	
Send Start ev.	; Packet transmission request to TRC
RQ.Send	; Sending request from TRC(1 Byte)
Write Byte	; Write 1 Byte to TRC
EOP	; End of Packet
RQ.Read	; Receiving request from TRC(1 Byte)
Read Byte	; Read 1 Byte from TRC
(term ev.)	; Line cutting off request from COMMON module via IMP monitor
LRC insert	; Insert LRC to end of packet
LRC check	; Check LRC
(closed)	; Line closed event to IMP monitor
(RD.Send)	; Ready report of sending Packet from PKTCNT module via IMP monitor

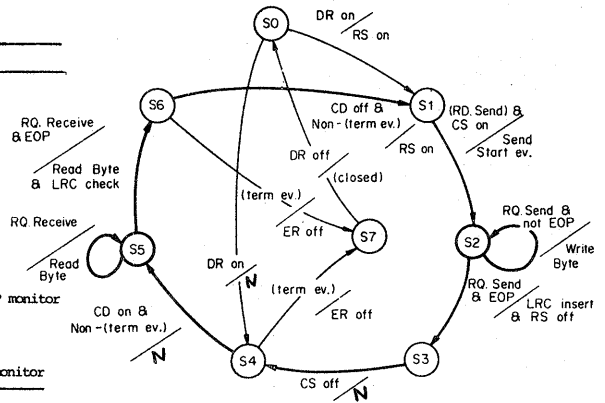
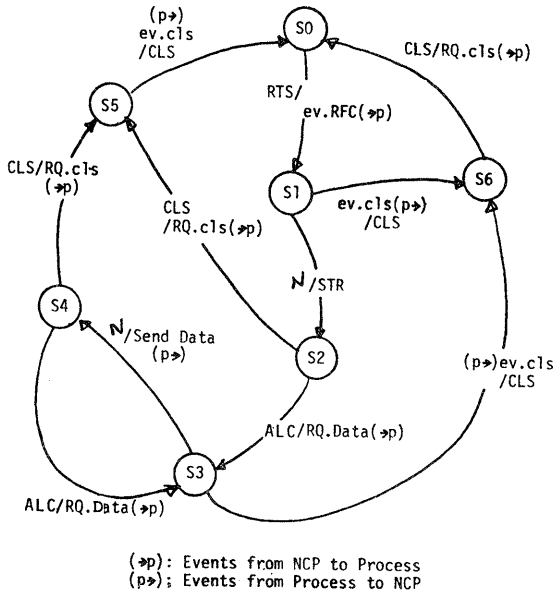


Table.2 Event Symbols of MDMCNT

(2) NCP K および子例

送、受信論理通信路の制御部について、Fig. 7(a), (b) を示し、各事象の説明を Table. 3 を示す。



(p): Events from NCP to Process
(p): Events from Process to NCP

Fig.7(a) Send Link

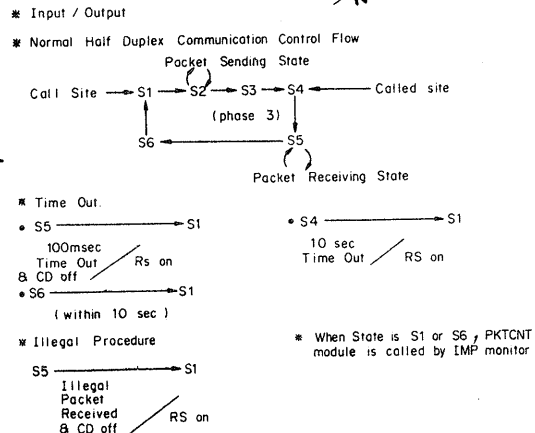


Fig.6 Micro Description of MDMCNT

Event symbols	
RQ.Connection	; Link setting request from process
RQ.cls	; Link close report to process
RQ.Data	; Send data request to process
RV.Data	; Data received report to process
ev.cls	; Link close request from process
ev.ALC	; Buffer allocation request from process
RTS	; } NCP Control Command
STR	
CLS	
ALC	

Table.3 Event symbols of NCP

5. 諸性値について

3. 4章で示した記述方式の諸性値について。以下考察を進めよう。

5-1. 表記法について

(1) 全体的表記と細部表記について

一般に、細部の表記法と、制御の全体的な流れの把握を与える表記法とは相反する関係がある。

ここで、特にプロトコルの業態と場合立場から、単一表記にとらわれず、多段階表記を採る。

つまり、マイクロ表記で機能のブロック単位を作成し、これに制御の流れと情報の流れ方向を明示し、全体的な関係を明示し、マイクロ表記で、機能毎の細部動作を記してやる。但し、ここで機能動作を中心としており、バケットフォーマット、制御情報の位置、意味などについては、付随説明が必要となる。

(2) 同期関係の表記法について

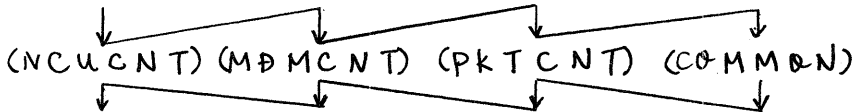
本方式では、1つのシステムを1つの単位にグループで表記するので、2つのシステムを同一表記で示す。このため Petri-Net などに比較し、厳密な同期関係を表記できないが、Fig. 7(a) 図に示したように送信、受信のシステムを別個に作成し、各状態毎に、検証を進めようとする。矛盾の検出が可能であり。(Fig. 6 の 2重通信制御であるため、送・受同一表記としてやる。)

5-2. 実装性について

(1) 機能ブロック間

通信制御システムの階層性から各機能ブロックの分割を行なう。

ここで、3章における機能ブロックは、制御上、下記に示す。キヌラインブ構造とする。



このこと、各状態遷移機械として独立して実装する場合、各機械は、実行の条件を、他の機械の状態値を参照する事により、1意的に到達しようとする。

又、デバック時において、各ブロックの状態値をトレースする事で、動的な制御の動きが把握できる。

(2) 操作性

本方式による各ブロックは、それ自身が状態遷移機械として考えられているため、プログラムの作成が容易である。

つまり、マイクロ表記において、各状態遷移機械間の状態値は、それを用いた実行条件を示し、入力事象は、状態における実行条件、出力事象は処理結果を示す。これをプログラム化した場合のフローチャートを Fig. 8 に示す。

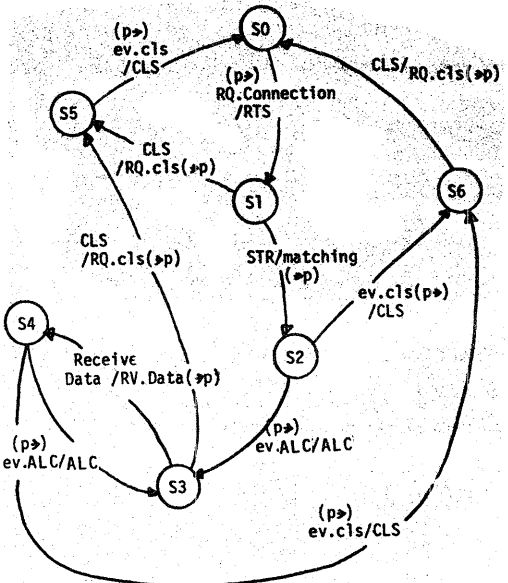


Fig.7(b) Receive Link

Fig. 8 からわかるように、各状態における処理構造は、すべて並列的同等であるため、仕様の変更、訂正などに對して、柔軟性が高い。
 又、多重レベルの多重制御を行う場合、状態遷移テーブルを拡張するだけで、容易にリエントラントな処理が可能であり、効果的である（NCPKにおいて特に示す）。

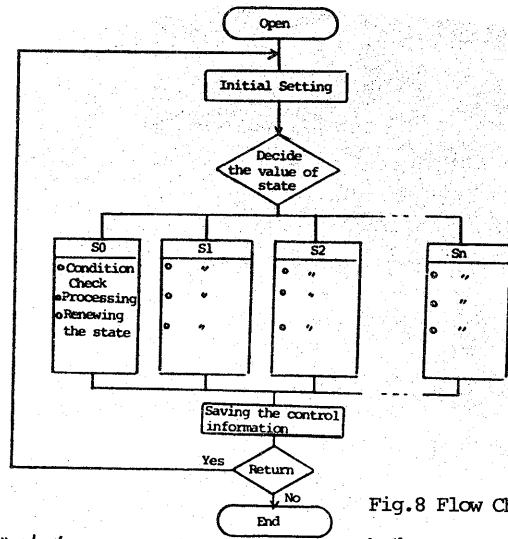


Fig.8 Flow Chart

6. まとめ

通信制御システムの主な階層性について、機能の分割を行う考えを示し、この分割がロックをもちきいて、特に通信制御システムの製作の立場から、製作者がプログラムを理解容易で、かつ突換へ早く等とすることを目的として記述方式について示した。
 この問題点として、通信の同期関係が1つの圏で、同時に把握しにくい点と、細かいビットの意味、フォーマットなどについて、付記が必要になってくる。又、状態数を多くすると、複雑になり、わかりにくくはなってしまう。
 最大12、3個が適当であろう。
 突製作面では、各状態の処理が独立であり構造のため、多少のヘッドが増える事である。
 しかし、プログラムの機能を状態遷移機械として関係する事は、製作者にとり合理的であろう。
 おわりに、本研究を御援助、御教示下さった北川・九州大名誉教授、大泉 千葉工大教授、城戸 東北大教授、中山 筑波大学教授、黒川、西野 電総研部長、又 熱心に意見を交換していただいたRCCの萩島、三工両氏に感謝いたします。

[参考文献]

- 1) James.L. Peterson : Petri Nets, Computing Surveys, vol 9, No. 3, Sept 19
- 2) Jonathan.B.P : A Graph Model Analysis Computer Communication Protocols, UCLA-ENG-7410 Jan, 1974
- 3) Rusbridge, R.E & Lagford.A : Formal representation of Protocols for Computer networks, AERE Hawel, Oxfordshire Dec, 1974
- 4) Slutz, D.R : The flow-Graph schemata Model of parallel Computation, Ph.D, MAC-TR-53 MIT Sept 1968
- 5) Danthine, A.S : An Axiomatic Description of The Transport Protocol of CYCLADES, Professional Conference on Computer
- 6) IBM : System Network Architecture Formal and Practical Reference Manual : Architecture Logic, IBM 1976
- 7) 日経電通 : N6240 Data Station COMI-212号, 1976, 5, 6.
- 8) Gauda.M.G & Manning E.G : Protocol Machines: A concise Formal Model and its automatic Implementation, Proc ICCO 1976
- 9) 友永充宏, 他 : 700121記述法の1次訂正情報誌, CN研究 12-2, 9, 1977
- 10) 高橋, 11号誌 : IMP 700121仕様, HUST 700121仕様, T-NET 4号, 内部報告書 1976.
- 11) 入彦理, 他 : 通信制御700121の記述方式理化学的1号誌, 情報学会, 全国大会 3A-7, 1978