

TR-81-010

派翠網理論在通信協定上之應用

研 究 報 告

計劃名稱：「散佈資料基系統的設計與建立」

計劃編號：NSC70-0404-E001-01

執行期限：69年12月1日至70年11月30日

主持人：柯志昇

研究生助理：鄭聖慶

劉龍駿

焦大偉

參 考 書
不 外 備

日期：中華民國 七十 年 十二 月

中研院資訊所圖書室



3 0330 03 00014 0

0014

目 錄

引言	1
一、通信協定描述和驗證技巧的回顧	2
二、派翠網引導模式系統	4
三、派翠網引導模式 (Petri-Net Derived Model)	6
四、通信協定描述語言 (Protocol Specification Language)	10
五、互換位元通信協定 (Alternating Bit Protocol)	11
六、結論	17
七、參考資料	18
附表一	
附表二	

派翠網理論在通信協定上之應用

引言：本篇報告主要的內容，是敘述一套發展中的派翠網引導模式系統 (Petri-Net Derived Model System, 簡稱 PNDM)。PNDM 可用來描述通信協定 (Protocol Specification) 和驗證 (Verification) 通信協定的正確性。本系統以派翠網引導模式 (Petri-Net Derived Model) 為描述通信協定的工具，並以模擬 (Simulation) 為驗證的工具，這份報告的內容以前者為主。

這份報告，第一部份，先將過去在這方面的研究成果做個回顧；第二部份，簡述這套發展中的系統；第三部份，介紹描述用的引導模式；第四部份，說明模式的描述語言；第五部份，以互換位元通信協定 (Alternating Bit Protocol) 為例，說明四、五部份實際使用的方式；最後，再探討未來發展的方向。

一、通信協定描述和驗證技巧的回顧

近年來，在通信領域上，由於電子技術的進步和電腦的使用，大眾資料通信網路(Public Data Communication Network) 日顯重要和發達。同時期內，在計算機領域上，由於計算機使用數量的增多，爲了有效利用機器資源，計算機通信網路(Computer Communication Network) 應運而生。大眾資料通信網路和計算機通信網路，由於電子技術和計算機的發展迅速，網路日趨複雜，使得它們被正確無誤地製作(Perfect implementation) 變得很不容易。爲了減少製作上的困難，各方面提出通信規則統一化的要求，經由一些國際性機構的努力，各種標準化的通信協定(Protocol) 相繼被提出，如 X.21, X.25, 等等。近五、六年來，學者專家們，融合軟品工程(Software Engineering) 的觀念，以解決網路日愈複雜所產生的問題，而相繼提出各種通信協定描述與驗證的技巧，使通信協定製作的方法化(Methodology) 日漸成熟。

首先，我們必須了解通信協定(Protocol) 的函意。簡單的說，通信協定是，通信單位(communication entities) 之間，相互傳播訊息的規則。

通信協定描述(Protocol Specification) 即是指這些規則的表明。一般而言，在通信系統中，通信協定是分層次(layers) 的，在通信協定描述上，對每一層都有三種描述：第 N 層的服務描述(N-Service Specification)，第 N 層的介面描述(N-Interface Specification)，第 N 層的協定描述(N-Protocol Specification)。

通信協定的驗證(Protocol Verification)，簡單的說，就是指證明一通信協定描述的正確性。驗證的對象，是以設計爲主。也就是在製作(implementation) 之前，先驗證設計(Design) 是否正確，以排除一些可避免的錯誤。一些較普通的特性，包括：避免死鎖的發生(Freedom from deadlock)，完全性(Completeness)，穩定性(Stability)，前進性(Progress)，結束性(Termination)。這些特性已被大多數專家共識。通信協定在應用之前，必須先被證明這些特性的正確無誤。

爲了使驗證的方法不致太困難，在描述上，我們就需要挑選適用於驗證的工具。相對的，描述的工具，也必須符合通信協定描述上的

需求。專家學者們，在上述二種原則下，提出各種不同方向的研究。在通信協定描述工具上，總結可分為兩大類。第一類為轉移模式(Transition Model)，主要包括有限狀態自動機 (Finite State Automata)，文法 (Grammars)，派翠網 (Petri Net)，UCLA圖 (UCLA Graph)。第二類為程式語言模式 (Programming Languages Model)。另外有混合第一類、第二類的特性所組成的模式。它們的驗證方式，第一類採取到達性 (Reachability)，第二類採取程式證明法 (Program Proof)。驗證的主要對象是前述的普通特性，其他的特性因方法不同，驗證的對象不同。但有一點，值得注意的，到目前為止，各方所提出的方法都祇能驗證一通信協定的部份正確 (Partially correct)，仍無法驗證一通信協定的完全正確 (Totally correct)，這是有待努力的。

二、派翠網引導模式系統

過去五、六年內，在通信協定描述和驗證方面有不少的研究成果，它們都是以轉移模式為主。其中較系統化和具體化的成果也都偏重於FSA和UCLA圖的模式。如[IMITS80]，以FSA模式為基礎，發展為語言，以語言輸入，自動的轉換成FORTRAN和PASCAL高階語言。在這篇論文中，它提出自動化製作(Automatic Implementation)的觀念。在[POST80]中，提出UCLA圖模式，它以數學方式驗證一些通信協定的特性。在[RAZO80]中，利用軟品工具SARA和UCLA圖模式，驗證CCITT X.21的問題，且更正了X.21的錯誤。這二篇論文很具體的引導我們研究的方向。在[SUNS81]中，利用軟品工具AFFIRM，配合FSA模式，提出服務描述(Service Specification)和協定描述(Protocol Specification)之間對應(correspondence)驗證的觀念。

由前述四篇論文的提示，我們概略的將PNDM系統繪出如圖 1：

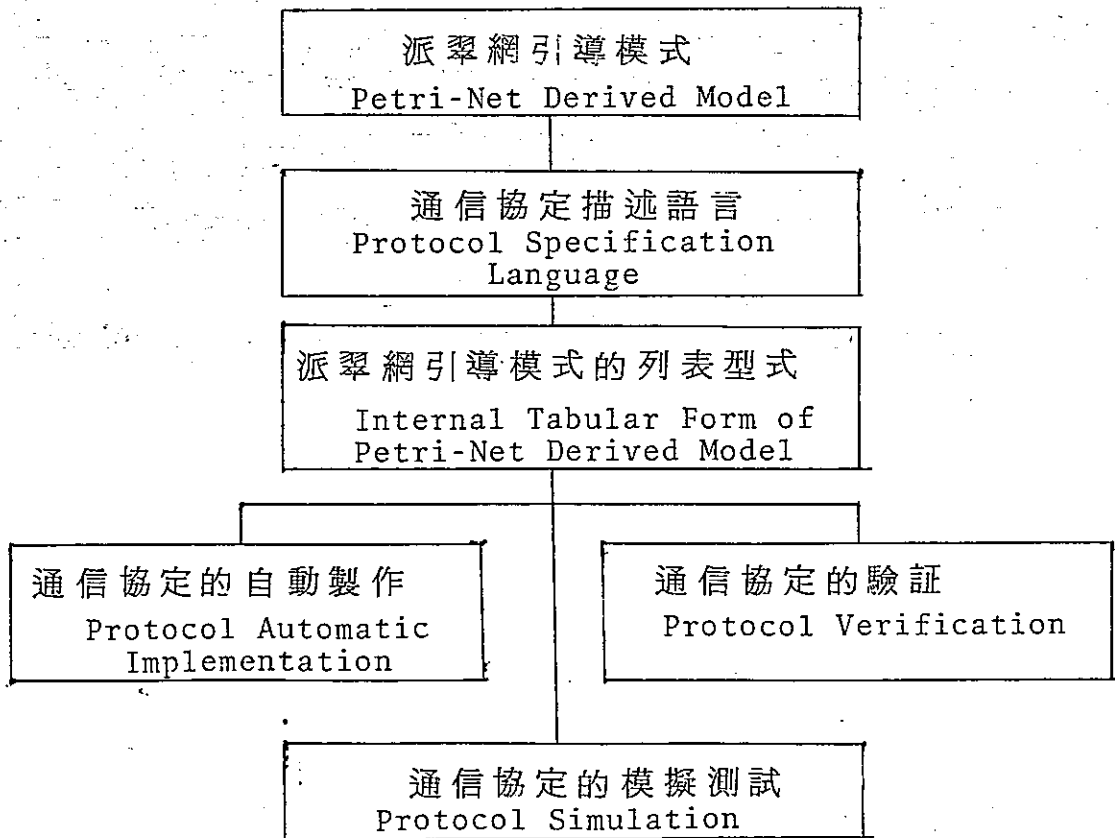


圖 1 PNDM 系統

對於一通信協定，首先將它以派翠網引導模式(PNDM)表示。爲了使計算機瞭解，我們必須將 PNDM 模式以通信協定描述語言寫成高階語言程式，再轉換爲利於處理的派翠網引導模式的列表型式，最後利用它來做模擬測試。

將來，本系統可延伸成具有通信協定自動製作的能力和採取數學模式的通信協定驗證方式。以下，介紹派翠網引導模式和通信協定描述語言，再以互換位元通信協定 (Alternating Bit Protocol) 爲例加以說明。

三、派翠網引導模式 (Petri-Net Derived Model)

派翠網是種雙方性複元圖 (bipartite directed multigraph) ，如圖 2 ，圖形上包含了二種節點 (node)：圓圈稱為地點 (places) ，橫槓稱為移轉點 (transition) ，這些地點及移轉點以方向性弧綫 (directed arc) 相連接。

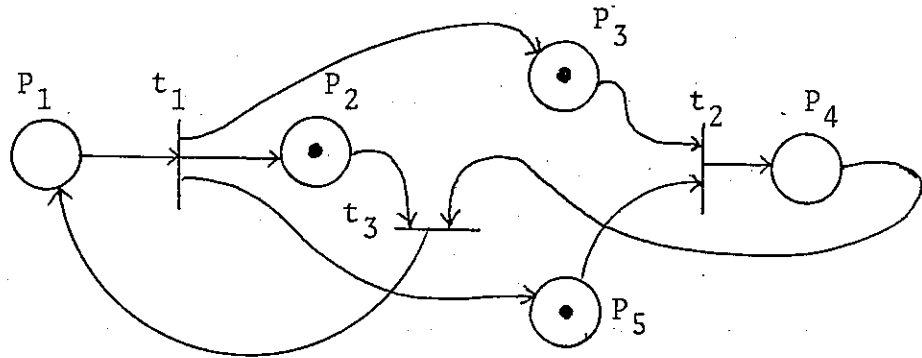


圖 2

地點和移轉點的連結，僅描述系統的靜態特性，若欲描述系統的動態特性，必須利用標幟 (tokens, 即圖 2 中， P_2, P_3, P_5 中的黑子) 的變化表示。而標幟的移動，決定於各移轉點和它相連地點的結構。在圖 2 中，移轉點 t_2 ，與它輸入弧綫相連的地點 P_3, P_5 都有一個標幟，則移轉點 t_2 可被發射 (fire) ，將地點 P_3, P_5 中的標幟吸收，而由輸出弧綫送出一個標幟到地點 P_4 ，如圖 3 所示。

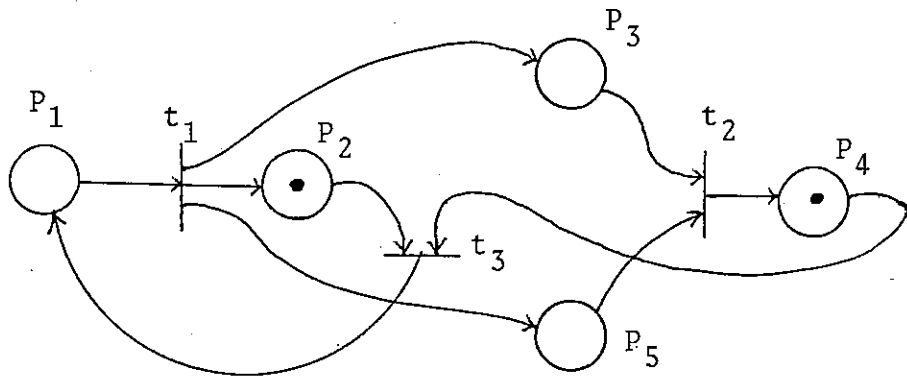


圖 3

派翠網的介紹詳見 [Pete 81]。限於篇幅我們不再加以詳述。

派翠網引導模式，是以派翠網為基礎，以各地點中標幟出現的情形，表示系統的狀態 (states)；相對的，轉移點的發射表示系統的動作。為使系統能力增強，各轉移點可以次轉移 (Sub-transition)，或動作指令集 (Action Statements) 表示動作。簡單說，派翠網引導模式，就是一般化派翠網 (Generalized Petri-Net) 加上轉移點具有層次性 (Transition Hierarchy)，和轉移點可以動作指令集表示動作。

由於，通信協定上的需要，系統允許特別的地點和轉移點。如圖 4 所示，以地點表示一計時器 (Timer)，在時效終止 (Time out) 時，地點會自動產生標幟。另外，以一地點表示資料傳送的要求，(Message Transmit Request)，在有新資料時，地點中會自動產生標幟如圖 4 所示。前者為固定 (constant) 的時限發生，後者則為任意 (Random)

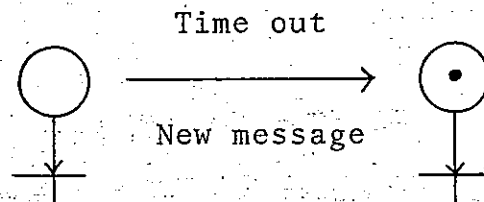


圖 4

時刻都可能發生。另外，有一種特別的轉移點，在描述資料傳送遺落 (message transmission loss) 用到它。如圖 5 所示，轉移點 M_L 沒有輸出的弧綫，當資料傳送遺落，標幟會被吸收 (absorb)。

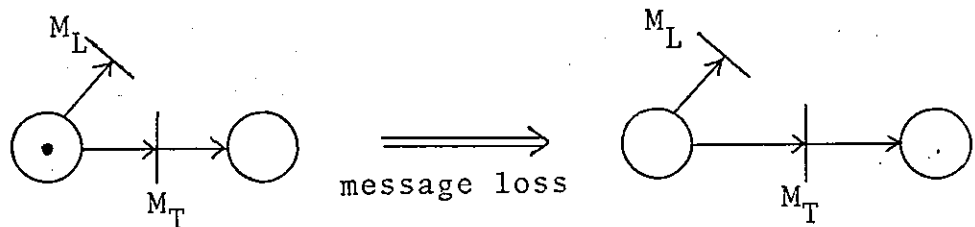


圖 5

其次，我們介紹動作指令集 (Set of transition action statements) 的意義。基本上，這些指令都可換為派翠網，在模擬時，做為計算時間單位的基礎。一般而言，指令可以分為兩種。第一種做為控制用 (Selection Control Statement)，是 E-net [NOE 73] 的延伸 (如圖 6、圖 7 所示)。它可以變數來決定標幟的移動。在圖 6 中，如變數 $K=1$

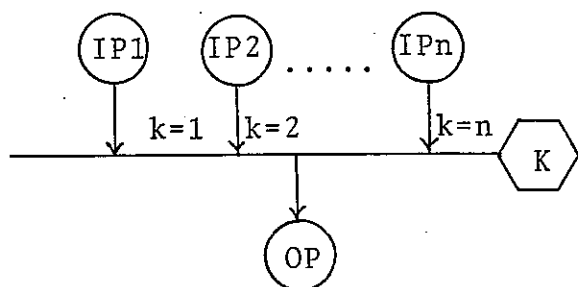


圖 6

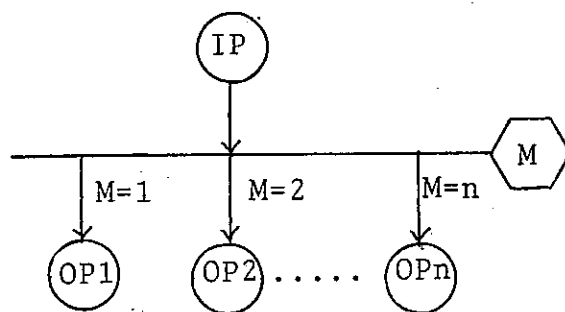


圖 7

時，地點OP祇由地點IP1吸收標幟。在圖7中，如變數M=2則標幟由地點IP移到地點OP2。控制用指令是以WHEN指令表示，如圖6，我們以

WHEN(IP1 · (K=1):OP | IP2 · (K=2):OP | ... | IPn · (K=n):OP)

圖7，則以

WHEN(IP · (M=1):OP1 | IP · (M=2):OP2 | ... | IP · (M=n):OPn)

':' 引號前，表示“條件”，其後表標幟的移動目標，如當地點IP1有標幟時，且K=1，標幟由地點IP1移到地點OP。

第二種為一般指令，包括：

- .基本指令 RECEIVE buffer name; SEND buffer name; 分別表示接收端接收資料和傳送資料的狀態，接收和傳送的資料在緩衝帶(buffer)中；一切正常，則跳過(Skip)一指令才執行，有問題時，則執行下一指令。
- .結束指令 EXIT; 表離開此指令集，不再處理其他指令。
STOP; 表此通信系統停止動作。
- .動作指令 它們的對象有二種，包括自然數和計數變數(counter)，其中自然數祇看成一種抽象的變數，它祇能用來做比較(Comparing)，不能變更其值。計數變數才可被變更，包括設定初值，如設定計數變數A為1，以“A+1”表示；另外有拆除計數變數的指令，以“←A”表示；和使計數變數加1，以“++A”表示。
- .IF 指令 它的格式為“IF ... THEN ... ELSE...”;例如“IF A=B THEN ++A ELSE ++B”，IF指令中的“條件”，可用自然數變數比較。

至於轉移點具有層次性，簡單的說，一轉移點，可以一派翠網引導模式替代，祇要將它的輸入，輸入弧綫加以描述。如圖 3 中的 t_2 ，可以圖 8 中黑線方格內的派翠網引導模式表示。在轉移點 t_2 發射時，

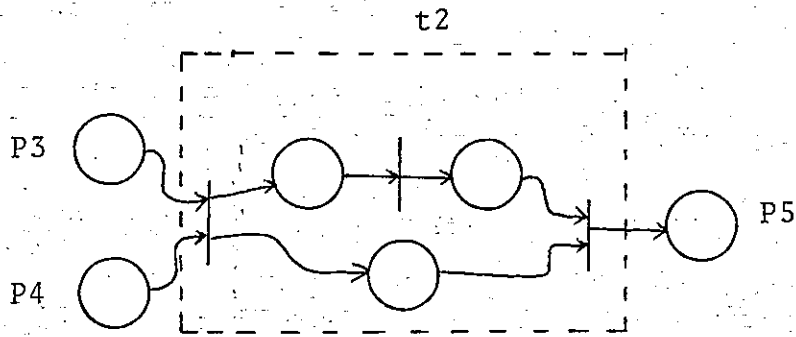


圖 8

標幟由地點 P_3, P_4 進入，再由地點 P_5 出去，它表示一些執行的動作。在分析上，首先肯定次轉移 t_2 的正確性，再以圖 3 的派翠網直接分析系統，這種分析方式比以整個派翠網分析簡易許多。

四、通信協定描述語言 (Protocol Specification Language)

通信協定描述語言，是讓通信協定設計者，將PNDM模式以高階語言的程式表示，本系統將它轉換成利於証明的形式，而後利用它模擬測試設計者輸入的通信協定的正確性。在附表1中，列出通信協定描述語言的內容。

它的內容依序是地點(place)、轉移點(transition)和變數(Variable)的宣告(declaration)，起始(Initialization)和終止(Termination)的設定，強制力(Forcing)的描述，和轉移點的描述(Transition Description)；在表中，大寫為關鍵語(Keyword)，小寫則為變值(Variable)；在地點，轉移點和變數的宣告部份，設計者必須將用到的地點、轉移點、和變數的名稱在此宣告。而變數的宣告，包括自然數(Non-negative integer)和計數(counter)變數兩種；其中計數變數以 modulo value-1 為其最大值，而計時計數變數(Timer Counter Variable)，在 modulo value-1 值時，再加1，會有時效終止(Timeout)的訊號產生。起始的設定，是設定系統在開始時，各地點中標幟的數目和變數的初值。終止的設定，則是設定系統停止動作的條件。強制力的描述，是指當系統中某些地點中標幟數與描述相同，或加上時效終止因素，同時符合時，系統強制到 FORCE 的狀態。轉移點的描述，包括接收端和輸出端弧線的描述，以及以轉移點的動作指令集或以次轉移來描述轉移點的動作。附表一和附表二詳述通信協定描述語言的架構和實例。

五、互換位元通信協定 (Alternating Bit Protocol)

互換位元通信協定的特性，是在不可靠的傳送介質上 (Unreliable transmission medium) 提供可靠的資料傳送服務 (Reliable message transfer Service)。它處理的方式是將傳送的資料依序加上序別 (Sequence number) 0 或 1 依 010101... 順序送出。接受資料的一端 (Receiver) 則依照收到資料的序別，送出相同序別 0 或 1 的回應 (Acknowledge)。它們的順序如圖 9 所示。但是因為在不可靠傳送介

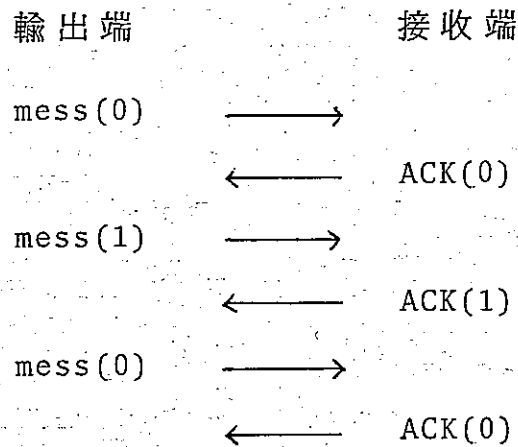


圖 9

質上，資料傳送可能發生錯誤或遺落，使輸出端無法在時效內收到正確的回應，因此必須使用時效終止 (Time out) 的功能使系統回復 (Recovery) 到正常情況運行，(如圖 10 所示)。

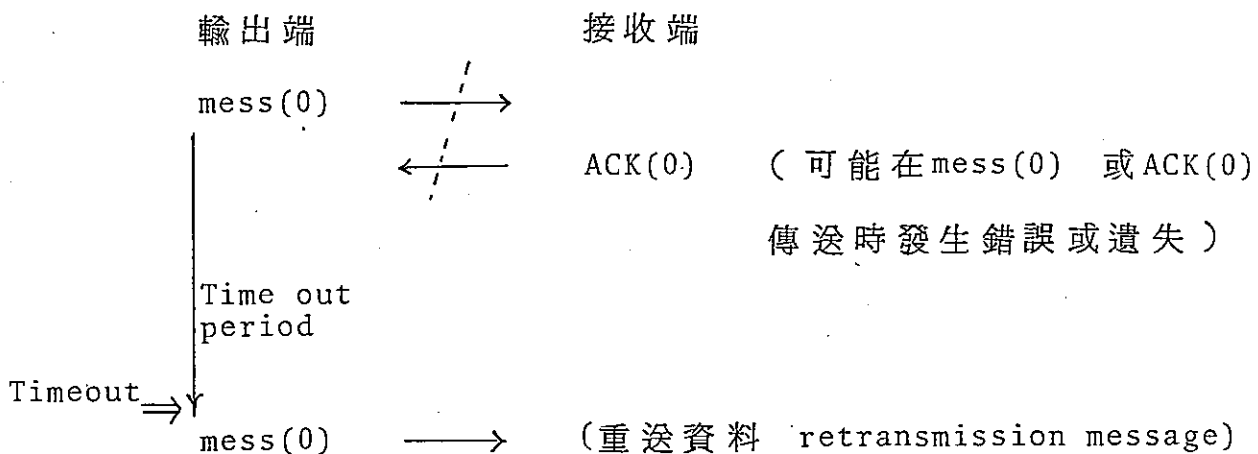


圖 10

接著，我們以服務的觀點看互換位元通信協定的用途。任一通信系統，以服務的觀點看，在資料輸送端可看成放置 (put) 資料於通信傳輸系統上，在接收端則可看成使用 (Use) 通信傳輸系統上的資料，如圖 11 所示

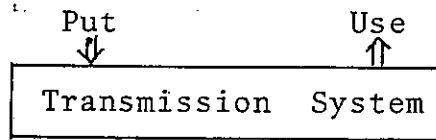


圖 11

接著，圖 12 以派翠網模式表示圖 11。

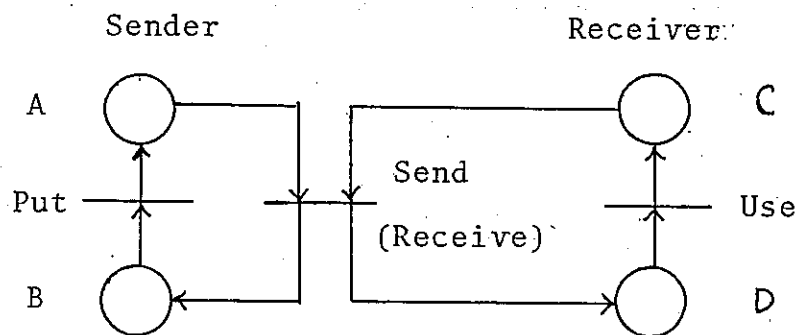


圖 12 服務描述 (Service Specification) 以 PN 表示

在圖 12 中，地點 A 表示資料可被送出 (Ready to Send)。地點 C 表示接收端可接收資料 (Ready to Receive)。地點 D 表示資料已收到 (Message Received)。地點 B 表示接收端送出的回應輸出端已收到。轉移點 Put, Use, Send (Receive) 各代表它們意義的動作。

以下，介紹一種分析派翠網的工具——標幟機器，簡稱 TM (Token Machine)，它是對應派翠網狀態的有限狀態自動機 (Finite State Automata)。如圖 13，將上述服務描述以它來表示。它是以派翠網中各地點的標幟數為其狀態 (States)。圖 13 中，狀態 "A, C" 表示派翠網中地點 A, C 各有一標幟，在地點 B, D 中則沒有標幟。經由轉移點

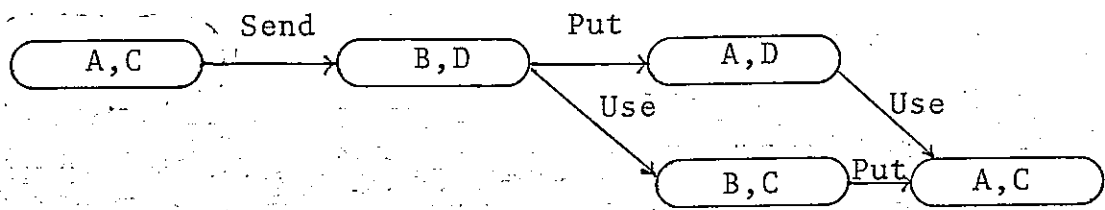


圖 13 : TM

Send 的動作，在圖 12 中，變為地點 B,D 各有一標幟，相對的，TM 由狀態 "A,C" 變為 "B,D"。接著，可能轉移點 Put 先做，或是轉移點 Use 先做，前者使 TM 由狀態 "B,D" 變為 "A,D"，後者則使 TM 由狀態 "B,D" 變為 "A,C"。由上，可看出 PNDM 模式中，允許轉移點 Put, Use 同時進行。但以標幟機器，則無法將此特性表示出來。換言之，TM 減低了 PNDM 用來描述並進動作 (Concurrent Action) 的能力。

在本模式下，協定描述 (Protocol Specification) 可由服務描述 (Service Specification)，經由轉移點層次性 (Transition Hierarchy) 引導出來，對於協定描述和服務描述間的對應 (correspondence) 關係是非常直接自然的。圖 14 [MERL761]，是引導出來的一通信協定模式。

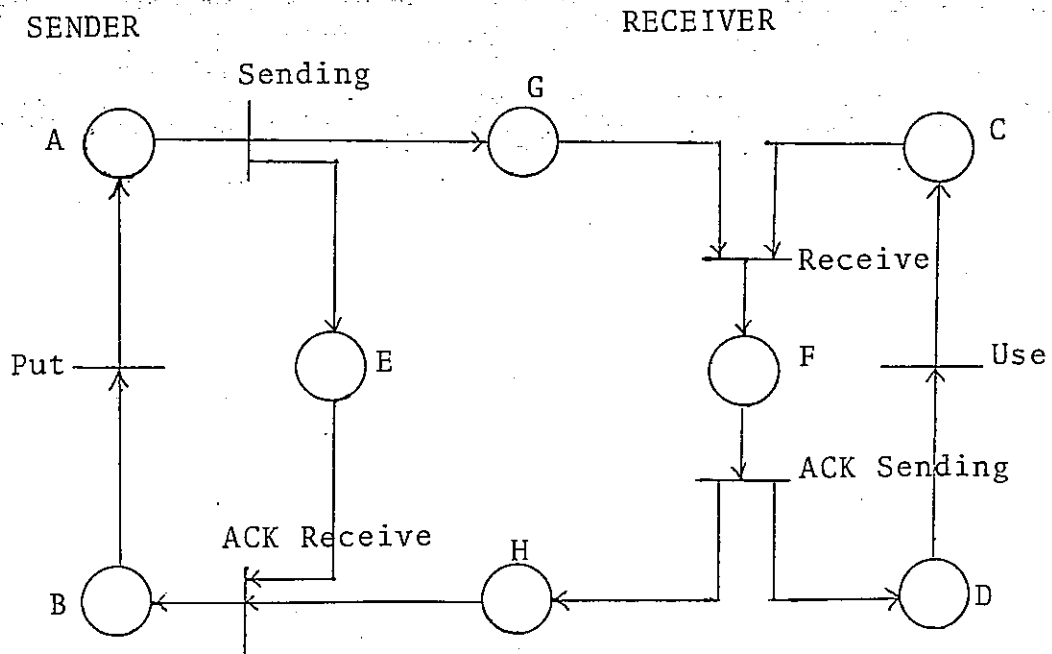


圖 14

地點 A,B,C,D 與服務描述圖 (圖 13) 中的 A,B,C,D 具有相同意義。地點 G 表示資料送出 (Message Sending)。地點 E 表示輸出端等待回應。地點 F 表示收到資料。地點 H 表示回應的送出 (Acknowledge Sending)。轉移點 Put,Use 與在圖 13 中意義相同,轉移點 Send (Receive)經由轉移點層次化擴展為圖 14。

由圖 14 可看出,它是未考慮錯誤發生的模式,由於資料可能有傳輸的錯誤或遺落 (transmission error or lose) 的情況,必須加以考慮。所以再將圖 14 擴展成為互換位元通信協定 (Alternating Bit Protocol),如圖 15 所示 [PETR79]。

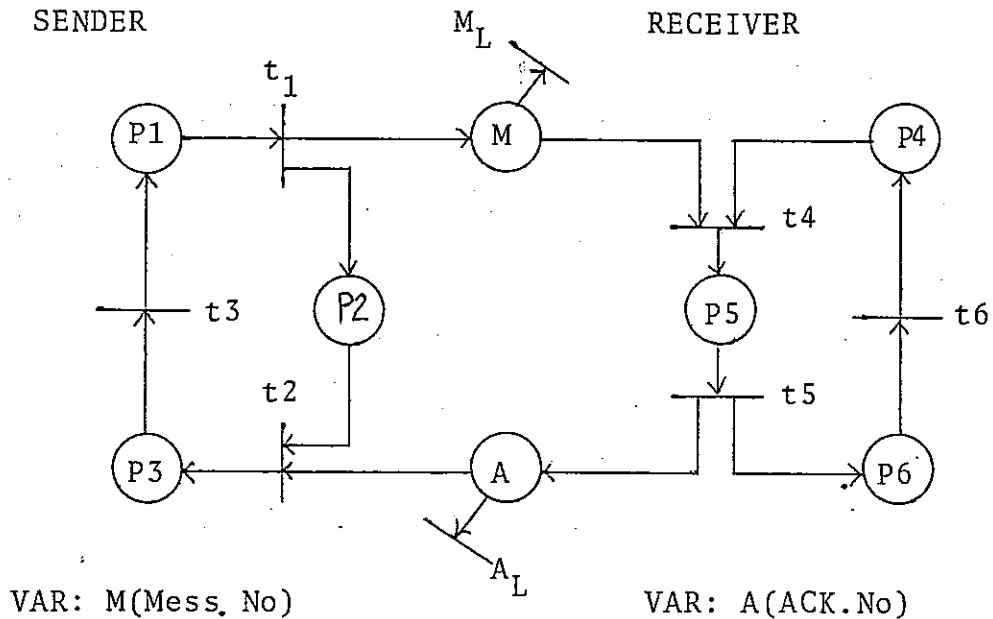


圖 15

圖 15 較 14 多出轉移點 M_L 和 A_L , 它們表示資料傳送的遺落; 但轉移點 t_4, t_5, t_1, t_2 具有更多的意義。轉移點 t_4, t_2 有轉移點動作子令集 (Transition Action Statement) 配合 t_5 和 t_1 則以轉移點層次化來擴充。簡述如下:

```

t4      Receive Message;
        EXIT;
        IF Message No.=1 THEN AX ← 1;
        IF MESSAGE No.=ϕ THEN AX ← ϕ;
        IF AX=A THEN ++A
    
```

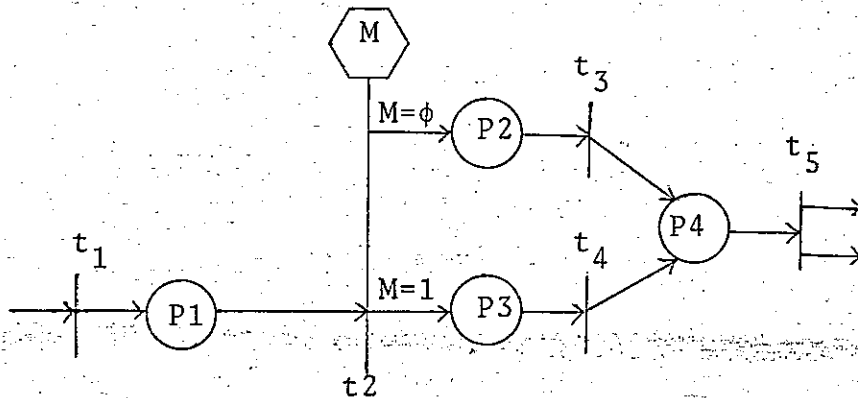


```

t2    Receive ACK;
      EXIT;
      IF ACK.NO= $\phi$  THEN MX  $\leftarrow$  1;
      IF ACK.NO=1 THEN MX  $\leftarrow$   $\phi$ ;
      IF MX=M THEN BEGIN ++M;
                           $\leftarrow$  R; (Retransmission number)
                          R  $\leftarrow$   $\phi$ ;
      END;

```

t1(T1.SUB)



t1.t3 Send Message ϕ

t1..t4 Send Message 1

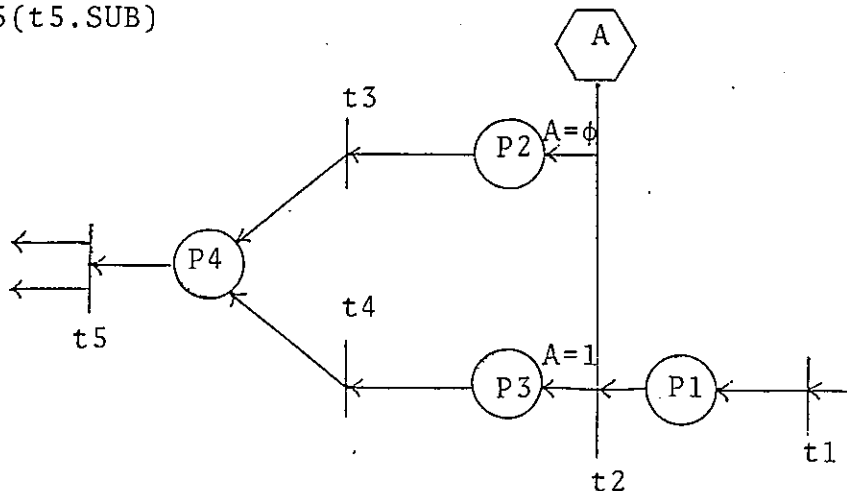
t1.t5 ++R;

IF R > allowance retransmission no THEN STOP;

\leftarrow Timer T01;

Timer.T01 \leftarrow ϕ ;

t5(t5.SUB)



t5.t3 Send ACK1;

t5.t4 Send ACK ϕ ;

以上，吾人以派翠網模式將互換位元通信協定描述。為利於計算機分析，尚須再將它以描述語言表達，詳見附表 2。

六、結論

軟晶工程日益進步，程式的證明 (proof) 和更正 (correctness) 愈來愈有希望，或許有朝一日本系統可以不用模擬來測試，直接以數學的方式驗證，但到目前為止，還沒有突破性的發展。在今日，我們祇能在部份更正 (Partially Correct) 上下手。也就是說，我們祇在一些通信協定特性上驗證它是正確的，而無法證明它是完全正確的。這是有待吾人努力的目標。另外，通信協定的自動製作也是可嘗試的方向，或許無法完全自動化，但半自動化製作是可能也是可行的。本系統尚需改進的地方還很多，這些都是未來研究的目標。

七、參考資料

- [DAVI73] David Misunas, "Petri Nets and Speed Independent Design," CACM, No.8, 1973.
- [DAY 79] J. Day & C. A. Sunshine, "A Bibliograph on the Formal Specification & Verification Computer Network Protocols," ACM Computer Communication Review, No.4, 1979.
- [MERL76] P. M. Merlin & David J. F., "Recoverability of Communication Protocols — Implications of a Theoretical Study," IEEE Tran. on Comm., No.9, 1976.
- [MITS80] "Automatic Implementation of Communication Protocol", Mitsubishi Electric Co. Computer Laboratory, Japan 1980.
- [NDE 73] J. D. Noe & G. V. Nutt, "Macro E-Nets for Representation of Parallel System," IEEE Tran. on Computer, No.3, 1973.
- [PETE81] J. L. Peterson, "Petri Net Theory and the Modeling of Systems," Prentice-Hall Inc., 1981.
- [PETR79] A. F. Petrenko, "On the Specification and Verification of Protocols Using Petri Nets," IIASA, Austria, 1979.
- [POST74] J. B. Postel, "A Graph Model Analysis of Computer Communication Protocol," UCLA, January 1974.
- [RAMA81] C. V. Ramamoorthy & Gary S. Ho, "Performance Evaluation of Asynchronous Concurrent System Using Petri Nets," IEEE Tran. on Soft. Eng., No.5, 1980.
- [RAZO79] R. R. Razouk & G. Estrin, "A Better Behavies X.21 Interface Derived From SARA Protocol Analysis," UCLA, Internal Memorandum #199, 1979.
- [THOM81] David H. Thompson & C. A. Sunshine & R. W. Erickson & S. L. Gerhant & Daniel Schwabe, "Specification and Verification of Communication Protocols in AFFIRM Using State Transition Models," University of Southern Californina, March 1981.

附表一

MODULE Module name

PLACE DECLARATION

placename ; ;placename ;

END OF PLACE DECLARATION

TRANSITION DELCARATION

transition name;... ..;transition name;

END OF TRANSITION DECLARATION

VARIABLE DECLARATION

variable name : INTEGER ; ... ;

timer counter name : TIMERCOUNTER (modulo value) ; ... ;

counter variable name : COUNTER (modulo value) ; ... ;

END OF VARIABLE DELCARATION

MODULE INITIALIZATION

place name : token number ;

.
. .
. .

VAR variable name <-- assign value ; ... ;

END OF MODULE INITIALIZATION

MODULE TERMINATION

TERMINATION termination name

BEGIN place name : token number ;

. .
. .
. .

END OF TERMINATION termination name

.
. .
. .

END OF MODULE TERMINATION

MODULE FORCING

FORCING forcing name

BEGIN placename : tokens number ;

.
.

TIMEOUT timer counter name ; ... ;

FORCE placename : token number ;

.
.

END OF FORCING forcing name

.
.
.

END OF MODULE FORCING

TRANSITION DESCRIPTION

<TRANSITION DESCRIPTION BODY>

END OF TRANSITION DESCRIPTION

END OF MODULE module name

TRANSITION DESCRIPTION

TRANSITION transition name

RECEIEVE ARC BEGIN

placename (arc number) ;

.
.

END OF RECEIEVE ARC

TRANSMIT ARC BEGIN

placename (arc number) ;

.
.
.

END OF TRANSMIT ARC

TRANSITION ACTION

1.when statement

2.counter initi. inc. out.

3.counter esrs comparing

4.p.1. could be changed to petri net

END OF TRANSITION ACTION

[<MODULE>]

END OF TRANSITION transition name

END OF TRANSITION

附表二

MODULE Alternating Bit Protocol

PLACE DECLARATION

P1; P2: P3; M; A; P4; P5: P6;
END OF PLACE DECLARATION

TRANSITION DECLARATION

t1; t2; t3; ML; AL; t4; t5; t6;
END OF TRANSITION DECLARATION

VARIABLE DECLARATION

Message No, Ack.No : integer ;
A:counter (2); M: counter (2);
R: counter (10);
T01: Timer counter (8);
END OF VARIABLE DECLARATION

MODULE INITIALIZATION

P3:1 ; P4: 1;
VAR M ← ϕ ; A ← ϕ ; R ← ϕ ;
END OF MODULE INITIALIZATION

MODULE FORCING

FORCING FOC1
BEGIN Timeout T01;
FORCE P1:1 ; P2: ϕ ; P3: ϕ ;
END OF FORCING FOC1
END OF MODULE FORCING

TRANSITION DESCRIPTION

TRANSITION t6

RECEIVE ARC BEGIN P6

END OF RECEIVE ARC

TRANSMIT ARC BEGIN P4

END OF TRANSMIT ARC

Use data

END OF TRANSITION t6

TRANSITION t3

RECEIVE ARC BEGIN P3

END OF RECEIVE ARC

TRANSMIT ARC BEGIN P1

END OF TRANSMIT ARC

Get data

END OF TRANSITION t3

TRANSITION ML

RECEIVE ARC BEGIN M

END OF RECEIVE ARC

END OF TRANSITION ML

TRANSITION AL

RECEIVE ARC BEGIN A

END OF RECEIVE ARC

END OF TRANSITION AL

TRANSITION T4

RECEIVE ARC BEGIN P4; M;

END OF RECEIVE ARC

TRANSMIT ARC BEGIN P5;

END OF TRANSMIT ARC

TRANSITION ACTION

Receive message;

EXIT;

IF Message No.=1 THEN AX → 1;

IF Message No.=φ THEN AX → φ;

IF AX=A THEN ++A;

END OF TRANSITION ACTION

END OF TRANSITION t4

*TRANSITION t2 LIKE t4

TRANSITION t5

RECEIVE ARC BEGIN P5

END OF RECEIVE ARC

TRANSMIT ARC BEGIN A; P6;

END OF TRANSMIT ARC

MODULE t5.SUB

PLACE DECLARATION

t5.P1; t5.P2; t5.P3; t5.P4;

END OF PLACE DECLARATION

TRANSITION DECLARATION

t5.t1; t5.t2; t5.t3; t5.t4; t5.t5;

END OF TRANSITION DECLARATION

TRANSITION DESCRIPTION

TRANSITION t5.t1

RECEIVE ARC BEGIN P5

END OF RECEIVE ARC

END OF TRANSITION t5.t1

TRANSITION t5.t2

RECEIVE ARC BEGIN t5.P1

END OF RECEIVE ARC

TRANSMIT ARC BEGIN t5.P2; t5.P3;

END OF TRANSMIT ARC

TRANSITION ACTION

WHEN(t5.P1 (A=): t5.P2

t5.P1 (A=1): t5.P3)

END OF TRANSITION ACTION

END OF TRANSITION t5.t2

END OF TRANSITION DESCRIPTION

END OF MODULE t5.SUB

END OF TRANSITION t5

* TRANSITION t1 LIKE t5

END OF MODULE Alternating Bit Protocol