

本書遵照教育部頒佈最新專科學校課程標準編輯之。

輯 邏 向 序

編 著 者

麥 松 梅

興業圖書股份有限公司

本書遵照教育部頒佈最新專科學校課程標準編輯之。

輯 邏 向 序

編 著 者

麥 松 梅

興業圖書股份有限公司

有著作權 不准翻印

中華民國 六十八年七月一日新二版

序向邏輯

定價：新台幣 150 元

編著者：麥 松 梅

發行人：王 志 康

出版登記證局版台業字第〇四一〇號

出版者：興業圖書股份有限公司

打字者：義 德 春 打 字 行

地 址：台南市正興街二十四號

發行所：興業圖書股份有限公司

地 址：台南市勝利路一一八號

郵政劃撥南字 31573號

團體訂購另有優待 (062) 373253

著 者 的 話

- 1 本書是根據教育部65年最新專科學校課程標準編著而成。
- 2 本書是數位電路四部份中之第二部份，數位電路分爲：
 - (1) 組合邏輯
 - (2) 序向邏輯
 - (3) 系統設計
 - (4) 週邊機械
- 3 本書之內容，力求簡單易於瞭解，決不涉及任何高深之理論，具有國中程度之學生，自可一目了然。
- 4 本書可供高級職校學生及一般技術員，作為參考用。
- 5 如授課時間不夠，第五章可提要講授。

著者：麥松梅

序向邏輯 目 錄

第一章 序向系統介紹

1-1	緒論	1
1-2	序向系統與組合系統的異同點	1
1-3	序向系統的描述	6
1-4	序向系統的運用	12

第二章 基本電路：正反器之應用與計數器之設計

2-1	緒論	24
2-2	二進位數儲存器—正反器的種類和特性	25
2-3	實用上之正反器	28
2-4	計數器之設計	32
2-5	實用上計數器之設計	71
2-6	同步電路之最大計時和最大計時容許誤差	86
2-7	正反器的其他運用	93

第三章 一般化同步序向邏輯之設計

3-1	緒論	106
3-2	由系統規定到狀態圖及狀態表	106
3-3	無開始狀態之序向系統	111
3-4	等效狀態與完全規定狀態表之簡化	112
3-5	不完全規定狀態表之簡化	116
3-6	狀態之指定	134
3-7	區分法	143
3-8	一般化同步電路之設計實例	152

第四章 一般化異步電路—基本模式系統的設計

4-1	緒論	160
-----	----	-----

4-2	異步序向系統研討	161
4-3	原始流程表之求得	163
4-4	原始流程表之合併	167
4-5	異步序向系統遷移圖與穩定性的分析	170
4-6	組合例題	174
4-7	等效狀態之簡化	187
4-8	輸出距陣	200
4-9	序向系統的雜波	205
4-10	設計實例	205

第五章 二次指定的方法

5-1	緒論	218
5-2	最少列之距陣	218
5-3	增大距陣之二次指定方法	228
5-4	四列流程表之一般化指定	232
5-5	循環二次動作	238
5-6	大流程表的列組合指定	241
5-7	迴授和延遲	242

第六章 一般化異步電路一脈衝式系統的設計

6-1	緒論	253
6-2	脈衝模式序向系統的研討—二次記憶裝置	254
6-3	脈衝模式序向系統的統程圖與流程表	255
6-4	激發距陣與輸出距陣之求得	258
6-5	設計實例	262
附錄一		270
附錄二		271
附錄三		278
附錄四		282

第一章 序向系統介紹

1-1 緒論:

在組合邏輯裡，我們曾經介紹一些數位系統的觀念和某些簡單的設計程序，由其動作的型態，我們稱那些電路為組合邏輯系統的電路，在本書將介紹另一類系統的電路即序向邏輯系統電路。組合和序向系統在很多方面不同，但在某些方面且是一樣的，在本章首先介紹一些有關序向與組合系統的某些定義與異同點。

1-2 序向系統與組合系統的異同點

1-2-1 組合系統的定義

組合系統可用圖 1-1 之一般化組合系統圖形來說明，所謂組合系統，必須滿足下列的限制：

- (1)輸出信號的真偽值，只由輸入信號當時之真偽值而定。
- (2)不可能有回授 (Feedback)，即輸出信號不可以作為原來電路之輸入。

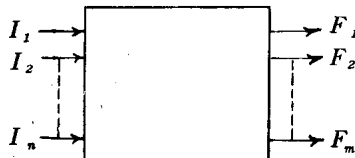


圖 1-1 一般化組合系統， I 表輸入， F 表輸出

當組合系統之輸入狀況改變後，緊接着的，其輸出亦假定變為新值，這些新值出現的時間，即落後輸入的延遲時間，與用來產生輸出之電路有關。理想化地，此等電路可不顧及延遲時間，即輸出信號隨着輸入信號而變。但實際上，在輸入與輸出間每一階層都有其本身的時間延遲，此全部之延遲時間，一般稱為組合系統的安頓時間 (Settling time)。

2 序向邏輯

敘述非理想電路性質的一般方法是將所有階層的延遲都想像中集中在一塊，且將延遲時間和電路的邏輯動作分開。此產生一如圖 1-2 所示的系統模型，於此一無延遲的組合系統跟隨着延遲元件。由理想系統來的輸出 $F_1, F_2, \dots, F_m$ 隨著輸入狀態而變且無延遲，而真實的組合系統輸出 $f_1, f_2, \dots, f_m$ 因電路延遲而落後於輸入信號的改變之後。故在某一時間，輸出 f 將不同於 F 信號，但此二組信號將永遠會相同，只要輸入信號在足夠久的期間內保持定值。

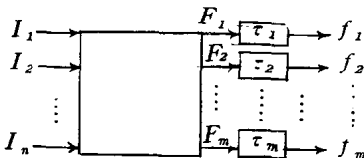


圖 1-2 一包括延遲的組合系統

沒有回授使輸入信號的每一改變將產生單一的輸出信號，而無不穩定或不確定的可能。此狀況使得組合系統的性質變的很簡單。可列出 2^n (n 表輸入個數) 個輸入組合，可決定每一 m 個輸出函數，並可列為每種可能的輸入組合的表。實在地，輸出不能馬上跟隨一個輸入改變，以符合輸入狀況，但此種情狀的存在時間，不會大於最壞情況的延遲時間。

1-2-2 序向系統之定義

在前節提及，組合系統之最大特性是其輸出只受當時輸入之狀況而定。而序向系統電路且不一樣，所謂序向系統，是其輸出非但與當時輸入情況有關，並且與電路本身在輸入前之狀態有關。此為序向系統基本上不同於組合系統，雖然它其中部份還是由組合系統所構成。

序向系統電路通常有下面兩個特徵：

- (1) 至少有一個回授通路，從系統的輸出接到系統的輸入。
- (2) 系統有保有過去狀態的記憶裝置或延遲裝置，使得以前之輸入及輸出值可與現在的輸入輸出值組合，以決定現在（或以後）之輸出信號。

一般序向電路基本構造圖如圖 1-3 所示，由其動作的特性可分為：

- (1) 同步 (Synchronous) 序向電路。
- (2) 異步 (Asynchronous) 序向電路。

1-2-2-1 同步序向電路

圖 1-4 顯示一般化之同步序向電路，可知序向電路是由組合電路與記憶器 (Memory) 所構成，輸入為 $x_1, x_2, \dots, x_n$ 及 y_1, y_2

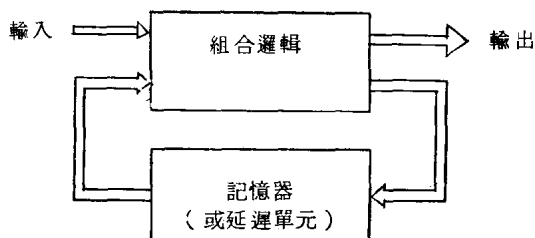


圖 1-3 一般化之序向邏輯電路

…… y_k 。其中 $x_1, \dots, x_n$ 是序向電路之實際輸入，而 $y_1, y_2, \dots, y_k$ 表可由記憶器獲得之回輸。輸出為 $Z_1, Z_2, \dots, Z_m$ 及 $Y_1, Y_2, \dots, Y_j$ ，其中 $Z_1, Z_2, \dots, Z_m$ 表序向電路之實際輸出，而 $Y_1, Y_2, \dots, Y_j$ 表儲存於記憶中資料。

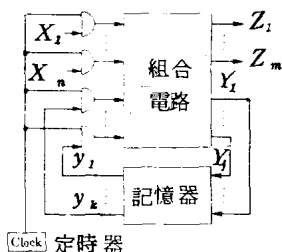


圖 1-4 同步序向電路之方塊圖



圖 1-5 定時器輸出之理想波形

計時器 (Clock) 實際上係一脈波產生器，圖 1-5 顯示其理想輸出。由於組合電路之輸入有 AND 閘，故序向電路之輸入必需與計時脈波同步。此即是“同步”序向電路名稱的由來。此計時脈波使輸入“同”時“步”入電路。

1-2-2-2 異步序向電路

圖 1-6 顯示一般化之異步序向電路，此電路與同步序向電路最大的不同是沒有同步之計時脈衝，故其輸入到達電路之時間不一致。

在大的序向電路系統，大都是屬於異步序向電路，而在此大的異步序向系統中，可能包含數部份之同步序向電路。一般來講，異步電路之速度較快，而同步電路之設計較簡單。

4 序向邏輯

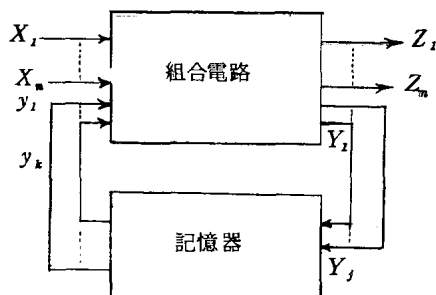


圖 1-6 異步序向電路之方塊圖

序向系統與組合系統比較，最明顯的好處是可節省系統電路的裝置。例如在圖 1-7 所示的二進位加法器，是用來將兩個四數元的二進位數相加，需要三個全加器和一個半加器，此二數作為組合系統全加器之輸入，經過時間延遲以後，可得四數元的總和及進位輸出（四數元的總和 S_1, S_2, S_4, S_8 及一數元進位 C_{16} ）。此種加法器是並聯加法器。因為所有之輸入資料在同一時間到達，且如不計延遲時間，所有之輸出在同時間獲得。

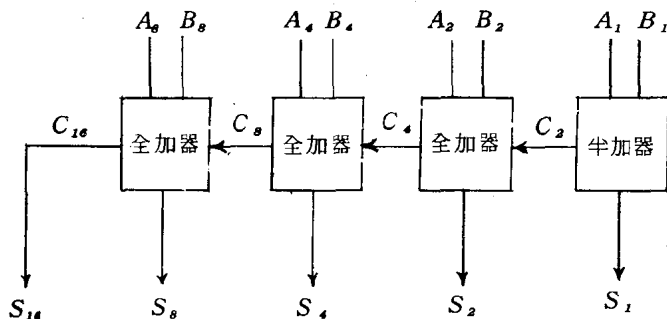


圖 1-7 並聯式四數元之二進位數加法器

並聯式的加法器可重新組合成序向的型式，如圖 1-8 所示，用一個全加器來完成每個數元相加的動作，因為相加的過程可依序重複動作。而每次相加之結果，可用儲存器來儲存做為下個數元進位的輸入。開始二進位數之最低數元碼 A_1 與 B_1 相加，其結果和及進位被儲存。儲存的進位加到下一數元碼 A_2 和 B_2 。重複此過程直到所有之數元皆相加完為止。

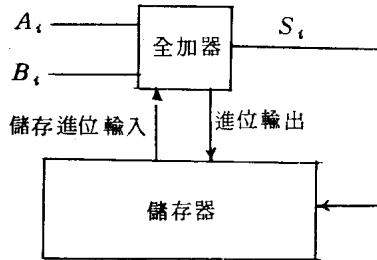


圖 1-8 序向的二進位數加法器

這種序向相加的例子，以圖 1-9 來說明之，如 $A = 1011$ 而 $B = 1010$ 為例，在(a)圖中，開始 1 與 0 相加，則總和 $S_1 = 1$ 而進位 $C_2 = 0$ ，接着(b)圖中 1 與 1 相加上原儲存器上個數元的進位 $C_2 = 0$ ，則總和 $S_2 = 0$ 而 $C_4 = 1$ 。接着(c)圖中 0 與 0 相加，再加上原儲存上之 $C_4 = 1$ 而得總和 $S_4 = 1$ ， $C_8 = 0$ 。最後(d)圖中 1 與 1 相加，再加上原來儲存器 $C_8 = 0$ 而得 $S_8 = 0$ ， $C_{16} = 1$ 。

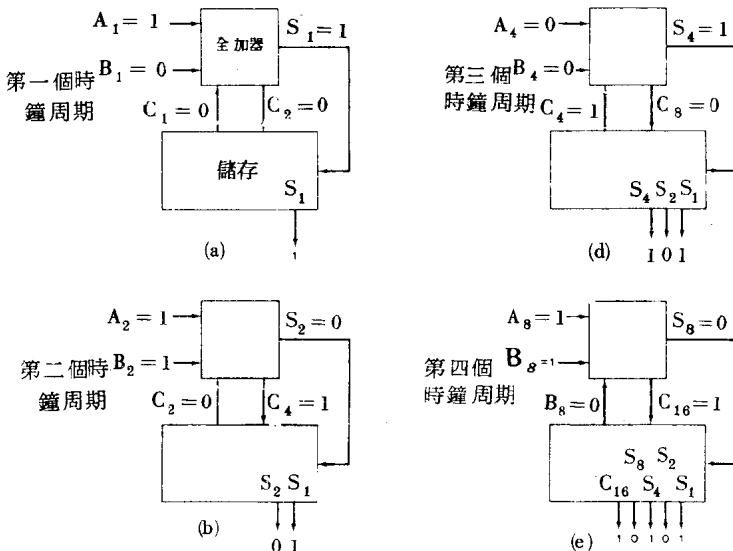


圖 1-9 兩個四數元之二進位數相加之過程

在上例中故能看出用序向電路來完成某些動作，較用組合電路來完成者用到較少的裝置。但同時亦可看出序向電路之動作步驟較複雜。即是說序向電路在設計上較困難，亦較容易產生錯誤的結果。

1-3 序向系統的描述

1-3-1 輸入，輸出和系統狀態

圖 1-10 是顯示一般化序向系統較詳細的邏輯方塊圖。此系統的組合邏輯部份分別接受兩組輸入信號：原先的輸入變數 (Primary input Variables) $X_1, X_2, \dots, X_n$ 和二次輸入變數 (Secondary Input Variable) $y_1, y_2, \dots, y_k$ 。二次輸入變數是從序向系統的記憶器部份而來。二次輸入變數的數值，用來定義系統的現在狀態 (Present state)。故二次輸入變數有時亦稱為現在狀態變數。這些二次輸入變數可有 2^k 個不同的組合，只允許序向系統有 2^k 個不同現在情況。因為此種原因，序向數位系統，有時亦稱為有限狀態 (Finite - state) 系統。

二次輸入變數是反映序向系統操作過去的結果。最初輸入變數定義現在輸入的情況 2^n 個不同可能組合之一。總合起來說，兩組輸入定義了系統全部的現在狀態 (Total present state)。全部共有 2^{n+k} 個不同的現在狀態。

序向系統的輸出部份亦分開成兩類：原先的輸出變數 (Primary Output Variable) 和二次輸出變數 (Secondary Output Variable)。原先的輸出變數 $Z_1, Z_2, \dots, Z_m$ 是由序向系統送出信號而接到其所要控制的另一系統。這些信號來用來推動其他系統：將指示燈打開或關上，或用此產生某種動作。

二次輸出變數 $Y_1, Y_2, \dots, Y_k$ ；此輸出回到序向系統的記憶器部份，這些信號敘述資料的新值，而在下一系統的動作發生時，將其存入記憶器。故二次輸出信號亦被稱為敘述序向系統的下一狀態 (Next state)。當 Y 的新值送到記憶器後，就取代 y 信號。

在每一個現在狀態，原先輸出與二次輸出是由原先輸入與二次輸入的組合動作而定。由現在的系統狀態 (由 $y_1, y_2, \dots, y_k$ 而定) 和原先輸入狀態 (共有 2^n 不同的組合) 經由序向系統的組合部份，而產生適宜的原先

輸出 $Z_1, Z_2, \dots, Z_m$ ，及二次輸出 $Y_1, Y_2, \dots, Y_j$ ，用以選擇此系統下個狀態的行進的方向。

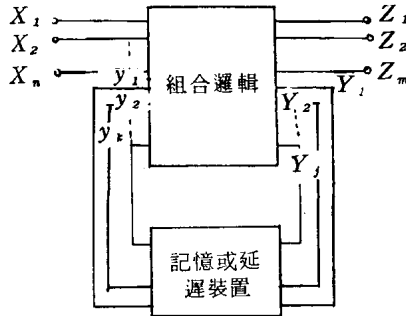


圖 1-10 一般序向系統之描述方式

1-3-2 狀態表和狀態圖

(State Table and State Diagram)

一序向電路之輸入與輸出，其現在狀態與下一狀態之變化情形，可以用幾種方法來說明。其中之一用的最多的方式是狀態表。在圖 1-11 中之簡單序向系統，有兩個原先的輸入變數 I_1 與 I_2 ，共有四個不同的組合輸入。一個原先的輸出變數 F ，和一個利用延遲作為記憶的狀態變數 (State Variables) 或二次變數 Y 。兩個現在狀態 $y = 0$ 和 $y = 1$ 。此兩個輸入與四個原先輸入，合併成八個不同的全部現在狀態。對應於每一現在情況，必需指定下一狀態變數值 Y ，和原先輸出變數 F 。圖 1-12 所示的排列是表示此種資料變換的通常方法，叫做狀態表 (state Table)

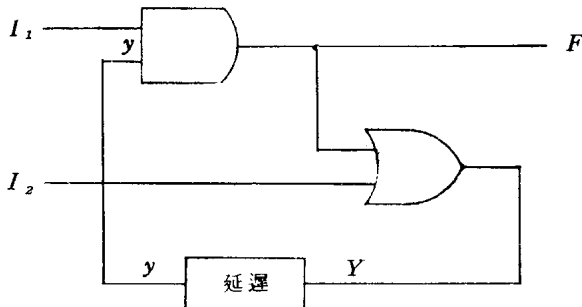


圖 1-11 一個簡單的序向系統

8 序向邏輯

在圖 1-12 中，現在狀態的數目決定列的數目。而下一狀態與輸出的行數，與輸入變數 2^n 個相當（ n 表輸入變數的數目）。對應於所有現在狀態的下一狀態與輸出值放在每一交叉點上。在圖 1-12 將下一狀態與輸出值分開。

現在狀態	下一狀態	輸出 F
y	$I_1 I_2 = 00 \ 01 \ 10 \ 11$	$I_1 I_2 = 00 \ 01 \ 10 \ 11$
0		
1		

圖 1-12 相當於圖 1-11 的狀態表

在圖 1-11 之組合電路顯示：

$$F = I_1 \cdot y \quad (1-1)$$

$$Y = I_1 \cdot y + I_2 \quad (1-2)$$

利用上兩式，可用來完成狀態表。

當 $y = 0$ 而 $I_1 I_2 = 00$ 時知 $Y = 0$ $F = 0$ ，如圖 1-13 所示。在此情況現在狀態 y 與下一狀態 Y 之值相等。故當延遲達到尾端時，將不會發生狀態的改變。在此種狀態，所有的下一狀態與現在狀態相同，即稱為穩定。穩定在狀態表上以在下一狀態的位置打圓圈表示之，如圖 1-13 (a) 所示。

當 $y = 1$ ， $I_1 I_2 = 00$ 時，由 1-1 及 1-2 式知 $Y = 0$ 及 $F = 0$ 。因 $Y \neq y$ 經過延遲後 y 會隨 Y 而變，為不穩定情況，最後 y 變為 0，而變為穩定的下一狀態，即 $y I_1 I_2 = 000$ 。根據相同原理，最後得圖 1-13 (b) 之流程表。所有穩定的情況被圈起而不穩定則沒有。

注意在狀態表中當原先輸入改變時，則狀態表在水平移動。如果當輸入改變後，原先狀態變為不穩定，則將發生一個或多個的垂直改變，直到系統穩定為止。

描述序向系統性質的第二種方法是利用狀態圖 (State Diagram)。此種方法是利用圖示的方法表示序向裝置的現在狀態及下一狀態的變化順序。狀態的變化直接以箭頭註明之。同時將每一狀態變換時的原先輸入和輸出狀態置於箭頭兩端。

現在狀態	下一狀態 Y	輸出 F
y	$I_1 I_2 = 00 \ 01 \ 10 \ 11$	$I_1 I_2 = 00 \ 01 \ 10 \ 11$
0	①	0
1		

(a)

現在狀態	下一狀態 Y^*	輸出 F
y	$I_1 I_2 = 00 \ 01 \ 10 \ 11$	$I_1 I_2 = 00 \ 01 \ 10 \ 11$
0	① ① 1 1	0 0 0 0
1	0 ① ① ①	0 0 1 1

(b)

圖 1-13 圖系統的完整狀態表

圖 1-11 中之系統的狀態圖如圖 1-14 (a) 所示。每一可能的現在狀態以一圓圈代表之。並將現在狀態置於其中。狀態至狀態的移遷以箭頭指示之。箭頭的開始和尾端同在一狀態上表示穩定的情況。用斜線分開輸入與輸出的資料，斜線左邊表輸入，斜線右邊表輸出，而在輸入變數中，給予 I_1 , I_2 的次序。此種符號型式可擴成圖 1-14 (b) 的型式。圖 1-14 (b) 的序向系統狀態 X 移至狀態 Y 。當：

$$I_1 = 0, I_2 = 1, I_3 = 0 \quad (1-3)$$

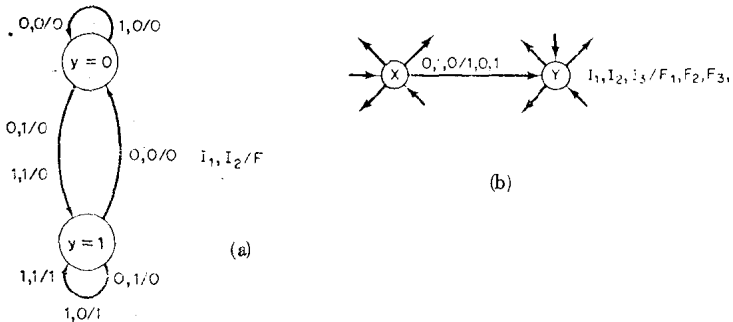


圖 1-14 (a)描述圖 1-13 系統的狀態圖。
(b)在較複雜的序向系統中之狀態轉移

10 序向邏輯

產生如下的輸出結果：

$$F_1 = 1, F_2 = 0, F_3 = 1 \quad (1-4)$$

例 1.1 如圖 E 1-1(a) 之序向系統，繪出其狀態表和狀態圖。

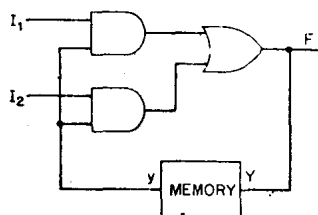


圖 E 1-1 (a)

解：其狀態表及狀態圖如圖 E 1-1(b) 所示。

現在狀態	下一狀態 Y^*
y	$I_1 I_2 = 00 \ 01 \ 10 \ 11$
0	① ① ① ①
1	① ① ① 0

* 輸出， F ，等於 Y

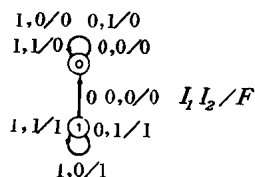


圖 E 1-1 (b)

狀態圖是表示序向系統全部性質的一種方便方法。其清楚地敘述狀態至狀態的順序而構成系統運用的特徵。狀態表和狀態圖是分析和設計序向數位系統的重要工具。此書的後面部份將用到此二種方法。

1-3-3 一般化的序向系統

(Generalized Sequential Systems) 之描述

並非所有的序向系統都如 1-3-2 節中例子那樣的被完全指定。在很多情況下，以不同名稱或符號來代表不同的現在狀態。以下面的例子說明之。

例 1.2 : 一序向系統動作成一簡單的組合鎖，允許那些知道正確組合的人進入一計算機控制室。系統接受四個輸入信號， R ， A ， B ，和 C 。輸入由按鈕而來，使得一時間內只能按下一鈕。系統

是以 R 信號清除 (重組) 。故打開鎖只依 B, C, A 順序。所有其它的順序將系統傳入錯誤的狀態。輸出信號是打開 (OPEN) 和錯誤 (ERROR)。求出敘述此系統的狀態表和狀態圖。

解：令系統的起先狀態稱作 a ，永遠以 R 信號迫使系統成狀態 a 。令錯誤狀態稱作 b 。到達狀態 b 之上，系統停留在狀態 b 直到接受一個 R 輸入時為止。如果接在 R 信號後的第一個輸入不是 B ，系統移至狀態 b 。否則其進入第三狀態，稱作 c ，來等待組合的其餘部份。當在狀態 c 時，除了輸入 c 外，其它的輸入皆會使系統移到狀態 b 。輸入 C 將使系統移到第四個狀態，稱作 d ，從此處一個 A 的輸入將使系統移到打開狀態上。稱作 e 。任何其它非 A 的輸入將使系統從狀態 d 移至狀態 b 。只有 B, C, A 順序會使系統到達狀態 e ，打開的狀態。錯誤和打開的輸出信號可從狀態 b 和 e 直接的得到。

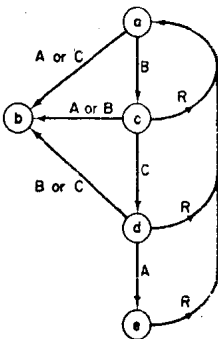


圖 E 1-2 (a)

現在狀態	下一狀態					輸出(打開,錯誤)				
	N*	R	A	B	C	N	R	A	B	C
a	Ⓐ	Ⓐ	b	c	b	0,0	0,0	0,0	0,0	0,0
b	Ⓐ	Ⓐ	Ⓐ	Ⓐ	Ⓐ	0,1	0,1	0,1	0,1	0,1
c	Ⓐ	b	b	b	d	0,0	0,0	0,0	0,0	0,0
d	Ⓐ	a	e	b	b	0,0	0,0	0,0	0,0	0,0
e	Ⓐ	a	Ⓐ	Ⓐ	Ⓐ	1,0	1,0	1,0	1,0	1,0

* N 指示無輸入狀態。所有現在狀態在輸入出現時為穩定

圖 E 1-2 (b)

此系統的狀態圖如 E1-2(a) 圖所示。同樣地，此系統的狀態表是圖 E1-2 (b) 在完成例 1-2 一般化的序向電路前，必需分配每一現在狀態的變數組合，至於變數的數目，與現在情況數目 N 有關，令 K 表變數的數目，則：

$$2^K \geq N$$

即

$$K \geq \log_2 (N)$$

(1-5)

K 必需為正整數