

【指令篇】

第 1 章：PLC 階梯圖程式基本原理及簡碼指令之轉譯法則

本章將介紹 PLC 階梯圖程式之基本原理，以及將階梯圖程式轉換成簡碼指令（Mnemonic）之轉譯法則。

1.1 階梯圖工作原理

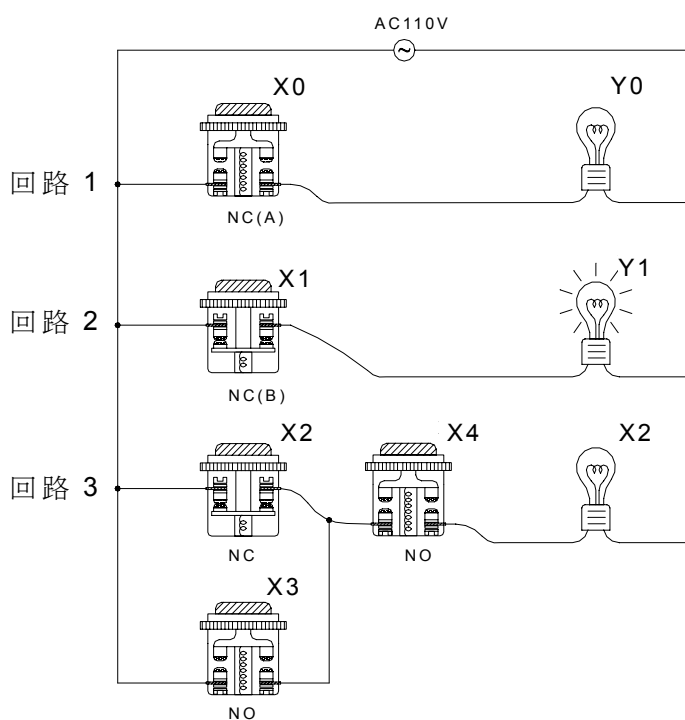
階梯圖為二次世界大戰期間所發展出來之自動控制圖形語言，是歷史最久、使用最廣之自動控制語言，最初只有 A（常開）接點、B（常閉）接點、輸出線圈、計時器、計數器等基本機構元件（今日仍在使用之配電盤即是），直到微電腦 PLC 出現後，階梯圖之元件（語言）除上述元件外尚增加了諸如微分接點、保持線圈等元件（請參閱 1-6 頁之元件類別）以及傳統配電盤無法達成之應用指令。

無論傳統階梯圖或 PLC 階梯圖其工作原理均相同，只是在符號表示上傳統階梯圖以較接近實體之符號表示，而 PLC 則採用較簡明且易於電腦或報表上表示之符號表示。在階梯圖邏輯方面可分為組合邏輯和順序邏輯兩種，茲分述如下：

1.1.1 組合邏輯

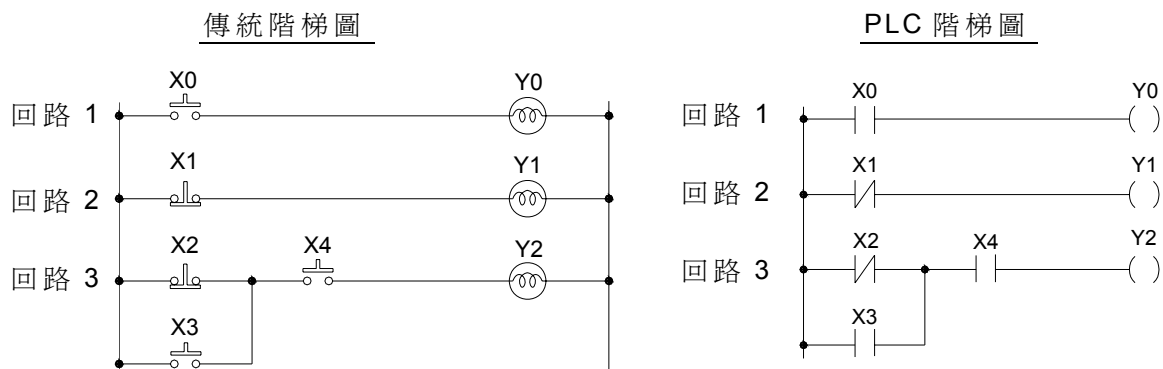
組合邏輯之階梯圖係單純地將單一或一個以上之輸入元件組合（串、並聯等）後再將結果送到輸出元件（線圈、計時／計數器或應用指令等）之回路結構。

實際配線圖



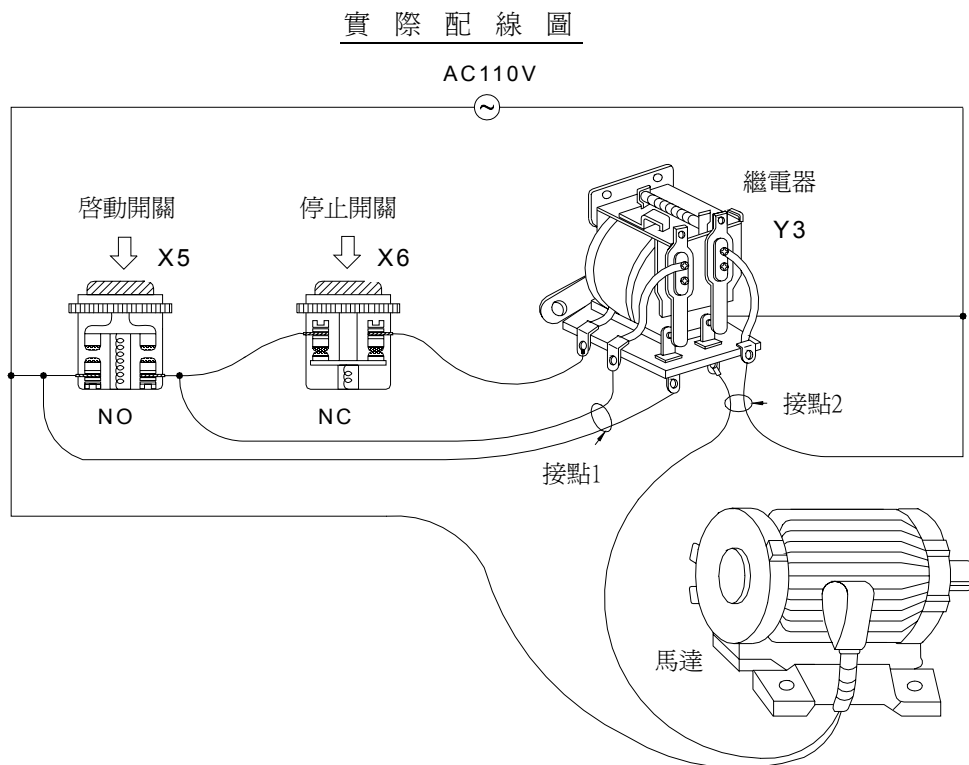
本例為組合邏輯分別以實際配線、傳統階梯圖及 PLC 階梯圖表示之範例，其中回路 1 使用一常開開關（NO：Normally Open）亦即一般所謂之“A”開關或接點。其特性是在平常（未壓下）時其接點為開路（OFF）狀態，故燈泡不亮，而在開關動作（壓下按鈕）時其接點變為導通（ON），故燈泡點亮。相對地，回路 2 使用一常閉開關（NC：Normally Close）亦即一般所稱之“B”開關或接點，其特性是在平常時其接點為導通，故燈泡點亮，而在開關動作時其接點反而變成開路，故燈泡熄滅。

回路 3 為一個以上輸入元件之組合邏輯輸出範例，其輸出 Y2 燈泡只有在 X2 不動作或 X3 動作且 X4 為動作時才會點亮。

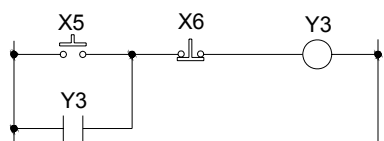


1.1.2 順序邏輯

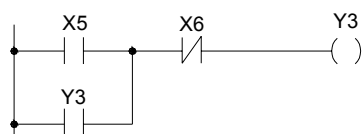
順序邏輯為具有回授結構之回路，亦即將回路輸出結果拉回當輸入條件，如此在相同輸入條件下，會因前次狀態或動作順序之不同，而得到不同之輸出結果，茲以下圖具自保功能之馬達啟動/停止回路作說明。



傳統階梯圖



PLC 階梯圖



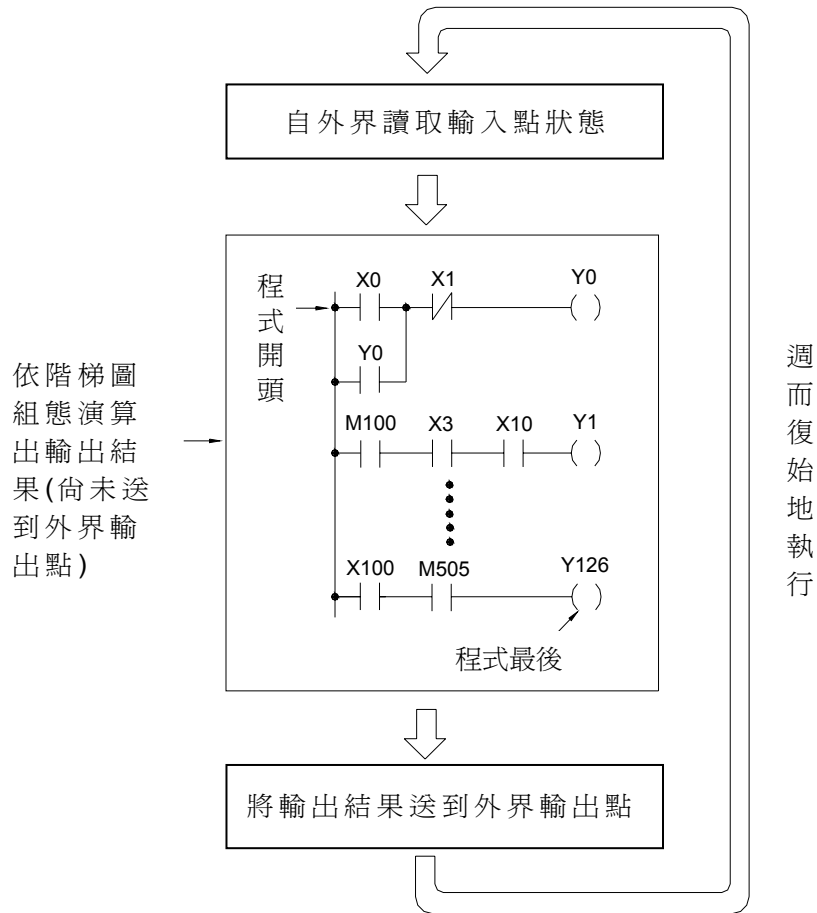
在此回路剛接上電源時，雖 X6 開關為 ON，但 X5 開關為 OFF，故繼電器不動作，而繼電器之輸出接點 1 和接點 2 均為 A 接點（繼電器動作時才 ON），故接點 1 和接點 2 均不導通，馬達在停止狀態。在啓動開關 X5 按下後，繼電器動作，接點 1 及接點 2 同時 ON，馬達開始運轉，一旦繼電器動作後，即使放開啓動開關（X5 變成 OFF）繼電器電源因為自身之接點 1 回授而仍可繼續保持動作（此即為自我保持回路），其動作可以下表表示：

	X5 開關 (NO)	X6 開關 (NC)	馬達（繼電器）狀態
①	放開	放開	停止
↓			
②	壓下	放開	動作
↓			
③	放開	放開	動作
↓			
④	放開	壓下	停止
↓			
⑤	放開	放開	停止

由上表可知在不同順序下，雖輸入狀態完全一致，其輸出結果亦可能不一樣，如表中之狀態①和③其 X5 和 X6 開關均為放開，在①狀態下馬達為停止，但狀態③時馬達卻為運轉，此種繼電器輸出狀態拉回當輸入（即所謂之回授）而使回路具有順序控制效果是階梯圖回路之主要特性，因之有人稱階梯圖為“順序控制回路”，而將 PLC 稱為順序控制器（Sequencer）。在本節範例中僅列舉 A、B 接點和輸出線圈作說明，其他元件之用法和此相同，請參考第 5 章“順序指令說明”。

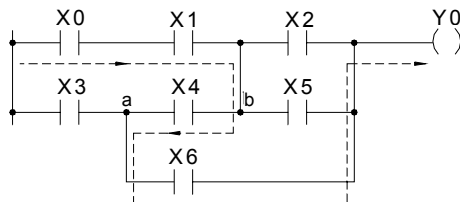
1.2 傳統階梯圖和 PLC 階梯圖之差異

雖然傳統階梯圖和 PLC 階梯圖之工作原理是完全一致的，但實際上 PLC 僅是利用微電腦（CPU）來模擬傳統階梯圖之動作，亦即利用掃描的方式逐一地查看所有輸入元件及輸出線圈之狀態，再將此等狀態依階梯圖之組態邏輯來演算出和傳統階梯圖一樣之輸出結果，但因 CPU 只有一個，只能逐一地查看階梯圖程式，並依該程式及輸入/出狀態演算輸出結果，再將結果送到輸出界面，然後又重新讀取輸入狀態、演算、輸出，如此週而復始地循環執行上述動作，此一完整之循環動作所費之時間稱之為掃描時間，其時間會隨著程式之增大而加長，此掃描時間將造成 PLC 從輸入檢知到輸出反應之延遲，延遲時間愈長對控制所造成之誤差愈大，甚至造成無法勝任控制要求之情況，此時就必須選用掃描速度更快之 PLC，因此 PLC 之掃描速度是 PLC 之重要規格，惟拜微電腦及 ASIC（特定用途 IC）技術精進之賜，現今之 PLC 在掃描速度上均有極大之改善，以 FBs-PLC 為例 1K step 接點之掃描時間只需 0.33ms，下圖為 PLC 之階梯圖程式掃描之示意圖。



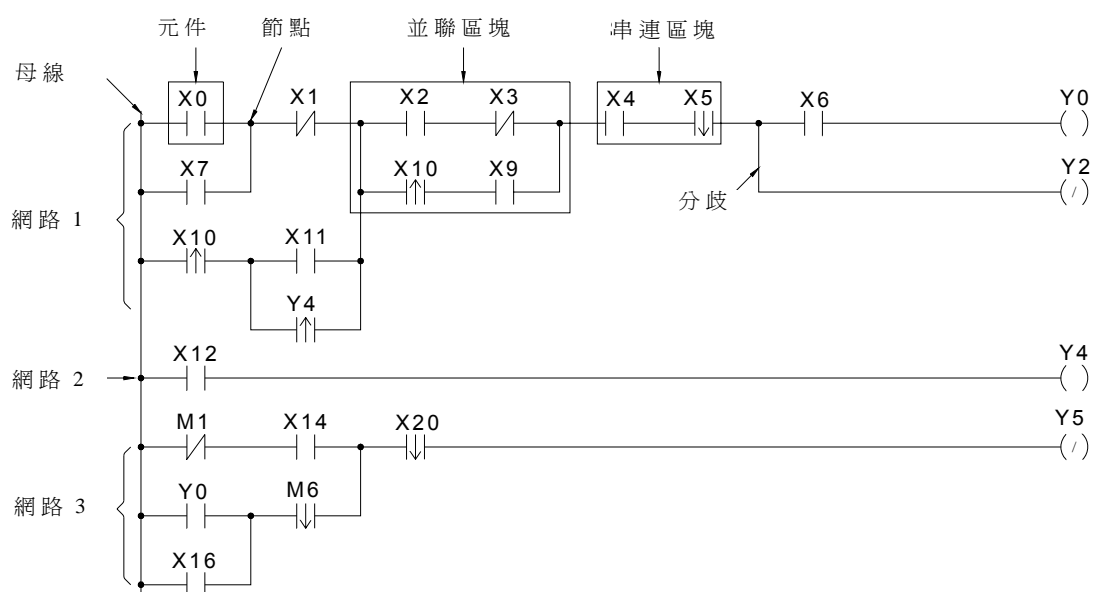
除上述掃描時間差異外，PLC 階梯圖和傳統階梯圖尚有如下之“逆向回流”之差異，如下圖所示圖中若 X0，X1，X4，X6 為導通，其他為不導通，在傳統之階梯圖回路上輸出 Y0 會如虛線所示形成迴路而為 ON，但在 PLC 階梯圖因 PLC 之 CPU 在演算階梯圖程式之結果時，係由左而右，由上而下地掃描。在同樣輸入條件下，本圖例中之 a 點狀態因 X3 接點 OFF 故 CPU 認定為 OFF，雖然 a 點經由 X4 接至 b 點均為 ON，但因 PLC 階梯圖只由左至右掃描，CPU 無法查覺，故 Y0 輸出為 OFF。

傳統階梯圖之逆向回流



1.3 階梯圖組成及其術語定義

圖一：階梯圖程式範例



(註：FBs 系列 PLC 之網路最大為 22 行 × 16 列)

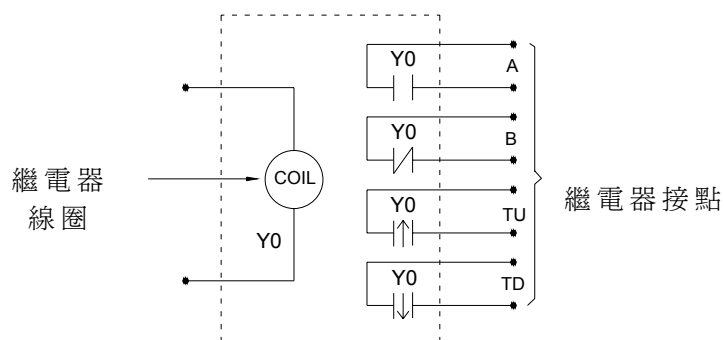
如上階梯圖程式可分為一個個小方塊（本圖例為 8 列 × 11 行 = 88 個小方塊），每個小方塊均可放置一個元件，將所有元件依控制需求作成各種不同之連結即構成所謂之階梯圖程式，茲就階梯圖程式相關之術語及其意義，分述如下：

① 接點 (Contact)

接點為表示導通 (ON) 與不導通 (OFF) 狀態之元件，共有兩類。一為“輸入接點”（編號以 X 開頭者），其狀態是來自外界（端子台上之輸入點）。另一為“繼電器附屬之接點”（請參考②項說明），其狀態是反應（來自）繼電器線圈之狀態。FBs 系列 PLC 所提供之接點有 A 接點、B 接點、上/下微分接點、開/短路接點 6 種，請參閱④元件之說明。

② 繼電器 (Relay)

正如同傳統繼電器，它包含線圈 (Coil) 和接點 (Contact)，如下圖例所示。

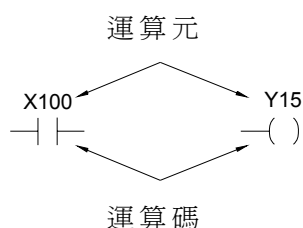


如圖示繼電器必有線圈，欲使繼電器動作，需驅動其線圈（用 OUT 指令驅動），在線圈被驅動後，其接點狀態會受到影響。如上圖例若將 Y0 以 1 驅動（使之為 ON），則繼電器之 A 接點為 1，B 接點為 0，TU 接點只 ON 一個掃描時間，TD 接點為 0。當 Y0 變成 OFF 時，A 接點為 0，B 接點為 1，TU 接點為 0，而 TD 接點只 ON 一個掃描時間（A、B、TU、TD 接點之動作請參閱第 4 章“順序指令說明”）。

FBs-PLC 之繼電器有四種，分別為 Y△△△（輸出繼電器），M△△△△（內部輔助繼電器），S△△△（步進繼電器）和 TR△△（暫存繼電器），其中輸出繼電器 Y△△△之狀態會被送到外界（端子台上之輸出點）去。

③母線（Origin）：階梯圖最左側之起始線。

④元件（Element）：元件（即線圈或接點）為組成階梯圖程式之最基本單位。元件之表示分為兩部分，一為元件之符號，稱之為運算碼（OP Code），另一為數字部分，稱之為運算元（Operand），如下圖所示。



FBs 系列 PLC 之元件有下列 9 種：

元 件 類 別	符 號	簡 碼 指 令 表 示 方 式	備 註
A 接點 (常開接點)	□△△△△ — —	(ORG、LD、AND、OR) □△△△△	□可為 X、Y、M、S、T、C (請參閱 2.2 節說明)
B 接點 (常閉接點)	□△△△△ — /—	(ORG、LD、AND、OR)NOT □△△△△	
上微分接點	□△△△△ — ↑—	(ORG、LD、AND、OR)TU □△△△△	
下微分接點	□△△△△ — ↓—	(ORG、LD、AND、OR)TD □△△△△	
開路接點	—○—	(ORG、LD、AND、OR)OPEN	
短路接點	—●—	(ORG、LD、AND、OR)SHORT	
輸出線圈	□△△△△ —()	OUT □△△△△	□可為 Y、M、S
倒相輸出線圈	□△△△△ —(/)	OUT NOT □△△△△	
保持型外部輸出線圈	Y△△△ —(L)	OUT L Y△△△	

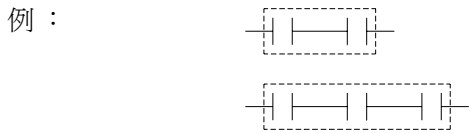
註：X、Y、M、S、T、C 等接點或線圈範圍請參閱 2.2 節、其元件特性請參閱 4.2 節。

另外尚有三個特殊順序指令（OUT TRn、LD TRn 及 FOn）亦屬元件之一種，但卻不顯示在階梯圖上，請參考第 1.6 節“暫存繼電器（TR）之使用”及第 5.1.4 節“功能輸出 FO”之說明。

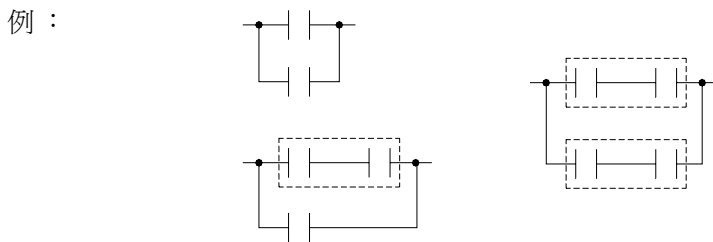
⑤節點 (Node)：任兩個或兩個以上元件相連接之點 (FBs-PLC 可對節點狀態作運作，請參考第 4.3 節“節點運作指令”之說明)。

⑥區塊 (Block)：兩個或兩個以上之元件組成之回路稱之。
基本之區塊有兩種：

- 串聯區塊：兩個或兩個以上元件串接而成之單列回路。

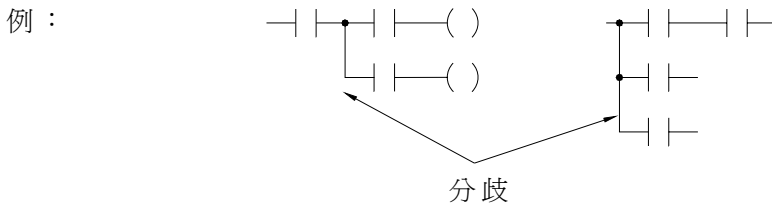


- 並聯區塊：由元件或串聯區塊並聯組成之平行 (矩形) 封閉回路稱之。

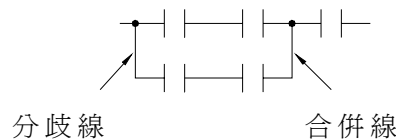


註：由元件、串聯區塊及並聯區塊等三種基本單元可以組成許多更複雜之串並聯區塊回路。在階梯圖程式輸入時，若以簡碼指令輸入，必須先將所有網路拆成上述之元件、串聯區塊、並聯區塊等基本單元後才能輸入，請參閱 1.5 節“階梯圖網路之拆解”說明。

⑦分歧 (Branch)：任一網路中之垂直線右方有兩列或兩列以上之回路連接，此即為分歧，而此垂直線即稱分歧線或稱為支線。

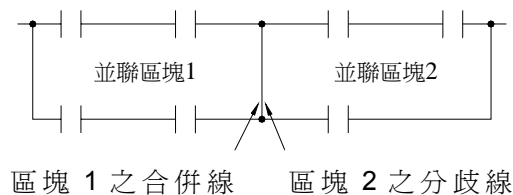


分歧線之右邊若有另一垂直線將分歧之兩列回路予以合併 (此垂直線稱之為合併線)，則此回路即形成一封閉之回路 (形成並聯區塊)，此回路即非分歧回路。



若垂直線左、右邊均有兩列以上之回路連接，則此垂直線既是合併線，又是分歧線。

如下例：

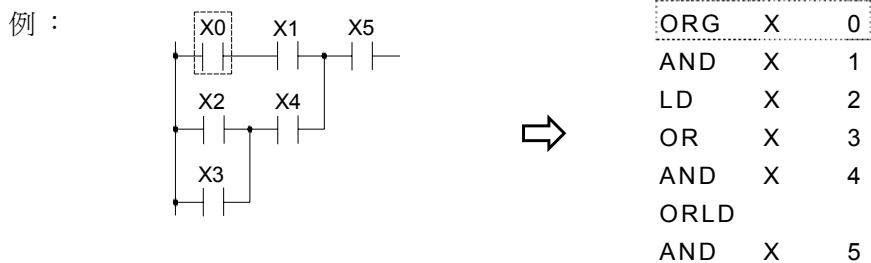


③網路 (Network)：由元件、分歧、區塊組成一能執行特定功能之回路，即稱為網路。網路是階梯圖程式中能執行完整功能之基本單位，而階梯圖程式就是由一連串網路所組成。網路之起始必須由母線開始，任一無垂直線連接之兩列回路即屬不同之兩個網路（有垂直線相連者則屬於同一網路）。依此法則，如圖一可區分成網路 1~3 三個網路。

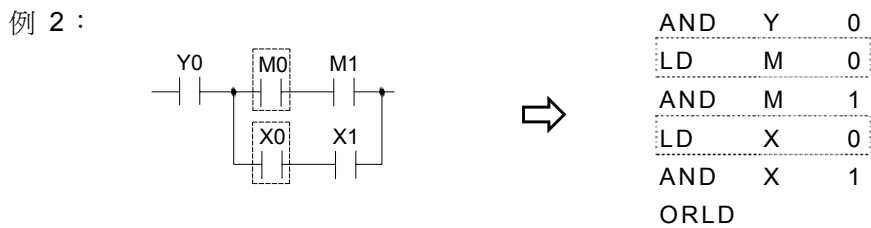
1.4 階梯圖程式轉成簡碼指令之轉譯法則 (WinProladder 使用者請略過)

FBs-PLC 若以 WinProladder 套裝軟體當規劃工作，則可由螢幕直接以階梯圖輸入，使用簡易、方便。但若您用 FP-08 當輸入工具，則因 FP-08 沒有電腦螢幕以供繪圖輸入，使用者必須依本節至 1.6 節所述之法則以人工方式先將階梯圖轉譯成等效之簡碼指令 (Mnemonic) 後才能輸入。以下為其轉譯法則：

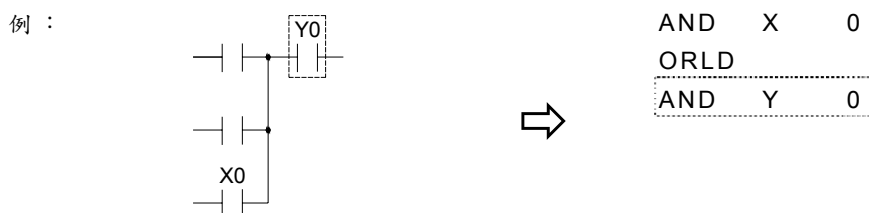
- 程式編輯係由左而右、由上而下，故網路之開頭一定在回路之最左上角，網路開頭指令必須用 ORG 指令，且一個網路只能有一個 ORG 指令（無輸入控制之應用指令除外，請參閱第 5.1.1 節之說明）。



- 接於垂直線（母線或支線）之指令用 LD 指令（網路之開頭除外）。

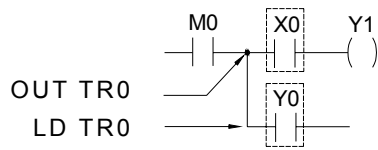


註 1：若支線上僅串接一系列元件則直接用 AND 指令。



註 2：若支線上已使用 OUT TR 指令將節點狀態暫存起來（分歧回路用），則亦用 AND 指令。

例：

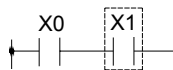


```

AND    M    0
OUT TR 0
AND    X    0
OUT    Y    1
LD     TR 0
AND    Y    0
    
```

- 單一元件串聯用 AND 指令。

例：

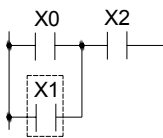


```

ORG    X    0
AND    X    1
    
```

- 單一元件並聯用 OR 指令。

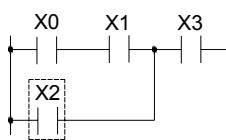
例 1：



```

ORG    X    0
OR     X    1
AND    X    2
    
```

例 2：

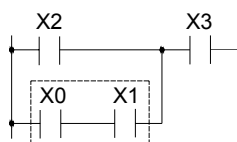


```

ORG    X    0
AND    X    1
OR     X    2
AND    X    3
    
```

- 並聯元件為串聯區塊時須用 ORLD 指令。

例：

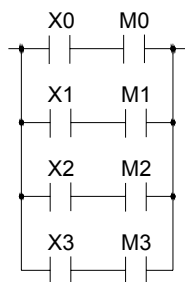


```

ORG    X    2
LD     X    0
AND    X    1
ORLD
AND    X    3
    
```

註：若並聯區塊不只兩列，則應由上而下，先並聯第 1、第 2 列後再和第 3 列並聯，餘此類推。

例：

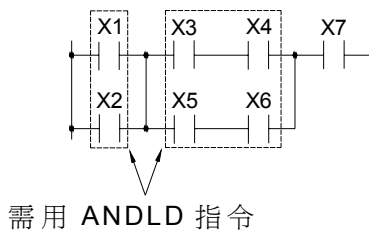


```

LD     X    0
AND    M    0
LD     X    1
AND    M    1
ORLD
LD     X    2
AND    M    2
ORLD
LD     X    3
AND    M    3
ORLD
    
```

- 並聯區塊和並聯區塊串聯需用 ANDLD 指令。

例：



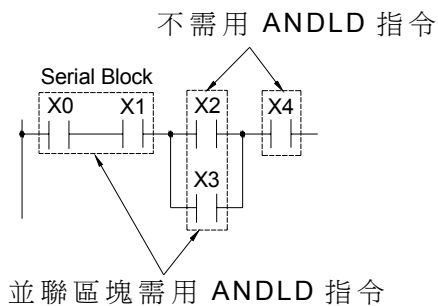
```

ORG X 1
OR X 2
LD X 3
AND X 4
LD X 5
AND X 6
ORLD
ANDLD
AND X 7

```

- 元件或串聯區塊和並聯區塊串聯時，若元件或串聯區塊在前，並聯區塊在後須用 ANDLD 指令。若並聯區塊在前，元件或串聯區塊在後則直接用 AND 指令將並聯區塊和元件或串聯區塊 AND 起來即可。

例：



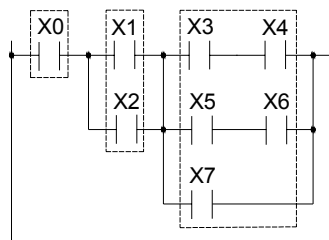
```

ORG X 0
AND X 1
LD X 2
OR X 3
ANDLD
AND X 4

```

註：若區塊之串聯不只兩個，則應由左至右先將第 1、第 2 個串聯起來後，再和第 3 個區塊串聯，餘此類推。

例：



```

ORG X 0
LD X 1
OR X 2
ANDLD
LD X 3
AND X 4
LD X 5
AND X 6
ORLD
OR X 7
ANDLD

```

- 輸出線圈指令（OUT 指令）只能放於網路之最後（最右邊），即其後不能再接任何元件。輸出線圈不能直接接母線。若有此需求可用短路接點串接之。如下例：



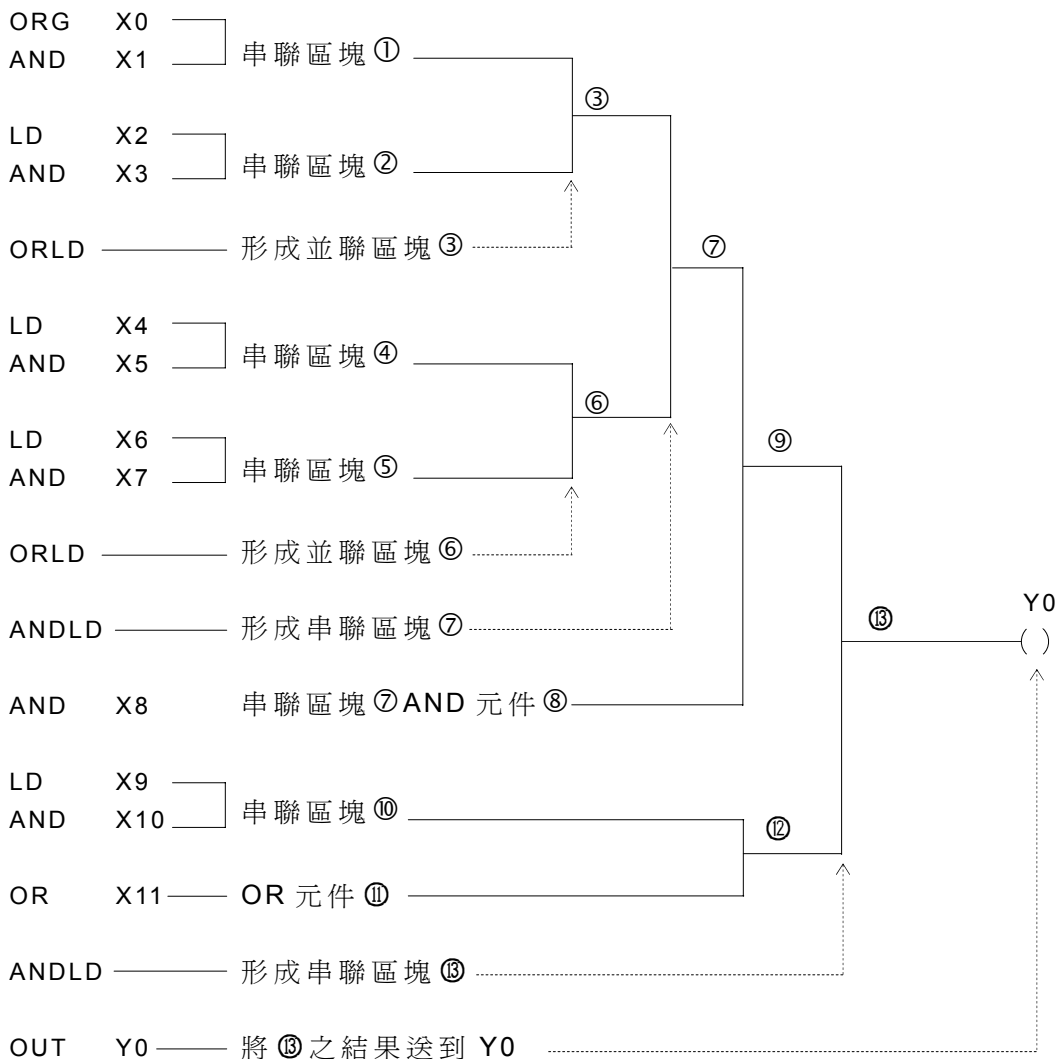
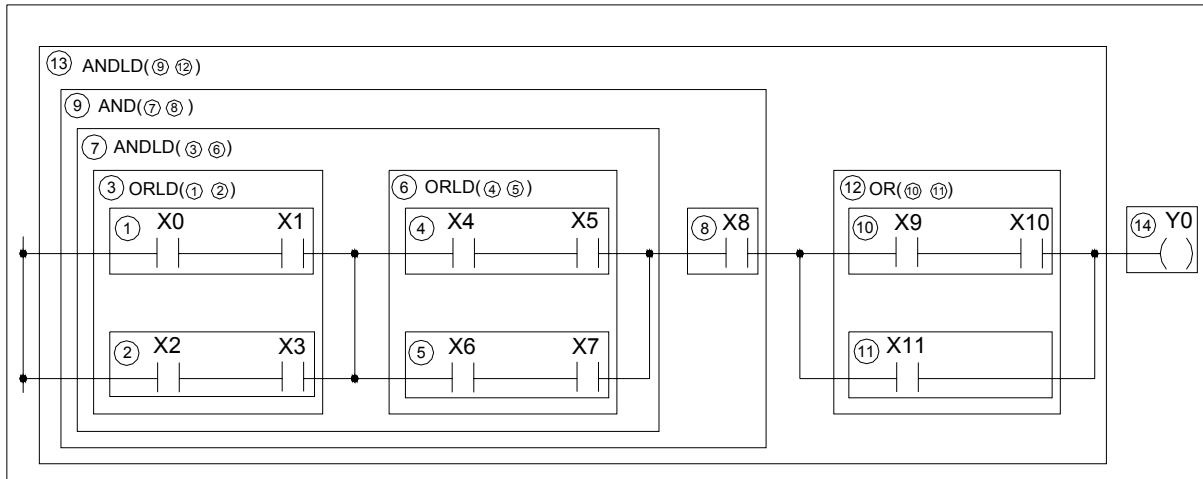
```

ORG SHORT
OUT Y 0

```

1.5 階梯圖網路之拆解 (WinProladder 使用者請略過)

網路拆解要領為將介於任兩垂直線之回路區分成獨立之元件或串聯區塊，再依上節所述之簡碼轉譯法則轉譯成簡碼指令，再由左而右、由上而下、由小而大將之連結成並聯區塊或串並聯區塊（用 ANDLD 或 ORLD 指令），直到整個網路均連結完成，如下圖範例：



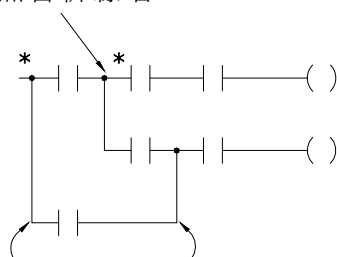
1.6 暫存繼電器(TR)之使用 (WinProladder 使用者請略過)

對分歧回路或分歧區塊而言，無法單純地利用 1.5 節所述之方法來拆解輸入，必須利用暫存接點先將分歧點之節點狀態存起來，再利用 1.5 節之方法進行輸入。因此回路設計應盡量避免形成分歧回路或區塊（請參閱下節“程式簡化技巧”所述）。茲就必須使用 TR 之兩種回路敘述如下：

- 分歧回路：分歧線之右邊無合併線者，或雖有合併線但和分歧線不同列者。

例： *表需設定 TR 點

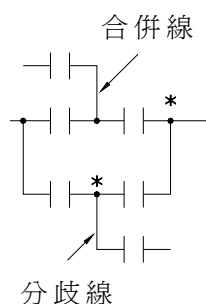
無合併線者



此分歧雖有合併線但不同列，亦屬分歧回路

- 分歧區塊：雖為平行（矩形）之並聯區塊，但區塊之任一系列有分歧者。

例：



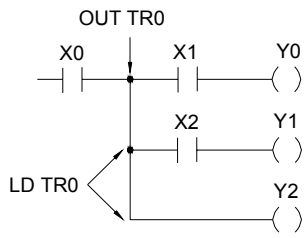
註 1：TR 點之設定必須在分歧回路或分歧區塊之分歧線之第一列（最頂端）處，而第二列以後之回路開始前必須先用 LD TRn 指令取回該分歧線之狀態後，才開始串接（AND）該列之第一個元件.....。（在 OUT TRn 或 LD TRn 指令後之第一個元件必須用 AND 指令，不能用 LD 指令）。

註 2：一網路中最大可有 40 個 TR 點設定。TR 點之號碼可任意選用，只要不重複即可（為易讀起見最好由 0，1，2，.....順序排起）。同一分歧線其 TR 號碼必須一致（例如一分歧線用 OUT TR0，在該分歧線之第二列起必須用 LD TR0 來接續）。

註 3：分歧回路或分歧區塊之分歧線若為母線，則無需使用 TR 接點，直接用 ORG 或 LD 指令即可。

註 4：分歧回路若有任何一系列非直接接輸出線圖（中間有串接元件），且其下方（第二列以後）尚有回路，則該分歧點必須使用 TR 接點。

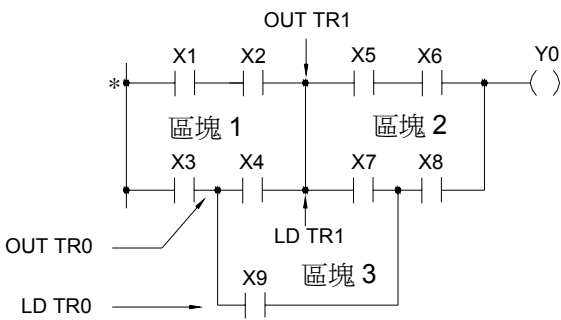
例 1：



```

AND      X   0
OUT TR 0
AND      X   1
OUT      Y   0
LD TR 0  ←第二列開始
AND      X   2
OUT      Y   1
LD TR 0  ←第三列開始
OUT      Y   2
    
```

例 2：



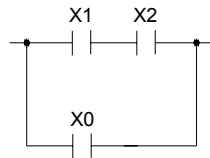
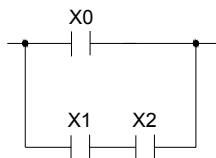
```

ORG      X   1
AND      X   2
LD       X   3
OUT TR 0
AND      X   4
ORLD
OUT TR 1  ←TR 指令後用 AND
AND      X   5  ←TR 指令後用 AND
AND      X   6  ←回 TR 點用 LD TR
LD TR 1
AND      X   7
LD TR 0
AND      X   9  ←TR 指令後用 AND
ORLD
AND      X   8
ORLD
OUT      Y   0
    
```

- 上圖例 2 之區塊 1、2 原本為典型之兩個並聯區塊串聯。但 X9 元件介入後不但形成區塊 3，尚使區塊 1、2 由原來單純之並聯區塊變成分歧區塊。
- (*) 處因為是母線，故不需用 TR 指令。
- 兩區塊串聯若已使用 TR 點作轉接，則無須使用 ANDLD 指令。

1.7 程式簡化技巧

- 單一元件和串聯區塊並聯，請將單一元件放於下方可省卻 ORLD 指令。



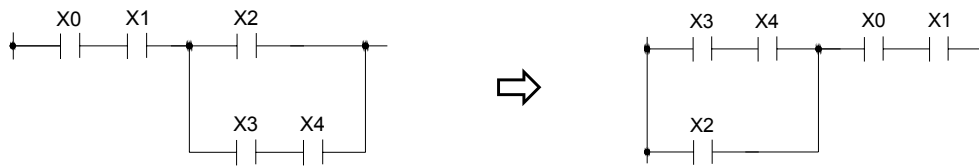
```

LD      X   0
LD      X   1
AND     X   2
ORLD
    
```

```

LD      X   1
AND     X   2
OR      X   0
    
```

- 單一元件或串聯區塊和並聯區塊並聯時，請將並聯區塊放於前方可省卻 ANDLD 指令。



```

ORG   X   0
AND   X   1
LD    X   2
LD    X   3
AND   X   4
ORLD
ANDLD

```

```

ORG   X   3
AND   X   4
OR    X   2
AND   X   0
AND   X   1

```

- 分歧回路之分歧點若直接接輸出線圈，應將該輸出線圈放於分歧線的最上面（第一列）。



```

OUT TR 0
AND   X   0
OUT   Y   0
LD TR 0
OUT   Y   1

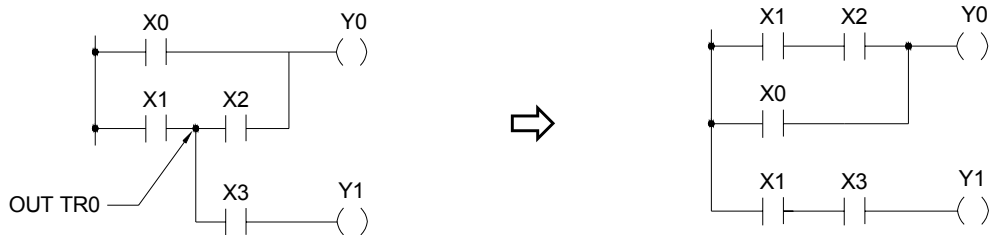
```

```

OUT   Y   1
AND   X   0
OUT   Y   0

```

- 下圖例可省卻 TR 接點及 ORLD 之使用。



```

ORG   X   0
LD    X   1
OUT TR 0
AND   X   2
ORLD
OUT   Y   0
LD TR 0
AND   X   3
OUT   Y   1

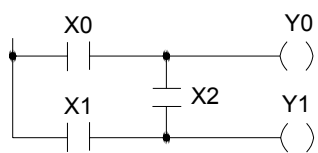
```

```

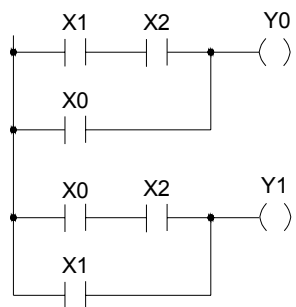
ORG   X   1
AND   X   2
OR    X   0
OUT   Y   0
ORG   X   1
AND   X   3
OUT   Y   1

```

- 橋式回路須作如下之轉換。



PLC 程式不容許
此網路結構



```

ORG  X  1
AND  X  2
OR   X  0
OUT  Y  0
ORG  X  0
AND  X  2
OR   X  1
OUT  Y  1

```