

GRAFCET 離散事件建模

陳慶瀚

MIAT 實驗室/中央大學資工系

2009 年 9 月 24 日

1. GRAFCET 基本原理

多數的嵌入式系統屬於離散事件系統(Discrete Event System, DES)，由於包含大量的輸入輸出變數和複雜的程序性和同步性事件，設計其控制器常展現高度複雜性。因此一個良好的離散事件控制器設計方法需要符合下列的需求：

- (1) 能描述出離散事件系統的大量且連續的狀態。
- (2) 必須考慮同時發生情況，且能以簡單容易瞭解的方式表達。
- (3) 通常一個系統的狀態只會受到幾個輸入的影響，且只有一些輸出被改變，因此只描述因輸入的改變而產生的行為。
- (4) 可以清楚的瞭解輸入與輸出的行為

為了因應上列的需求，我們採用 Grafcet 模型。Grafcet 是由 Petri nets 發展而來，用來描述連續行為與同時發生情況的一種圖形化模型。Grafcet 模型是由一個結合學界和工業界的法國團體所定義出來，1987 年成為國際標準的方法。

Grafcet 是一種具有兩種類型的圖示元件，分別是步驟 (Step)和步驟轉移 (Transition)。一個 Grafcet 至少包含一個步驟和一個步驟轉移，連結時由一個步驟連結到一個步驟轉移或是由一個步驟轉移連結到一個步驟。

A. 步驟與步驟轉移

一個步驟以一個方塊來表示，如圖 1 所示。步驟的狀態可以是動作(Active)如圖 1 步驟 1 所示在步驟中加入一個標記，或是不動作(Inactive) 如圖 1 步驟 2 所示。當一個系統開始時它的初始步驟定義為動作的，它的表示方式為一雙

方塊如圖 1 步驟 0 所示。

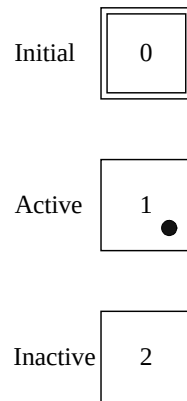


圖 1、步驟的表示方式

步驟轉移的表示方式如圖 2 所示。步驟轉移的符號是一橫槓，但在某些情形下會在其之前或之後加入一雙橫槓。如圖 2(b, d)所示有兩個或多個步驟一起進入步驟轉移，這時就必須在步驟轉移之前加入一雙橫槓。它意味著在步驟轉移產生之前必須等待多個步驟皆變成動作。當步驟轉移後有兩個或多個步驟一起離開，這時就必須在步驟轉移之後加入一雙橫槓如圖 2(c, d)。它意味著在步驟轉移後之後多個步驟將同時變成動作。每一個步驟轉移與通過 P(Pass)的真(True)、假(False)有關， P_i 的狀態則由 Grafcet 輸入變數或是內部狀態來決定。

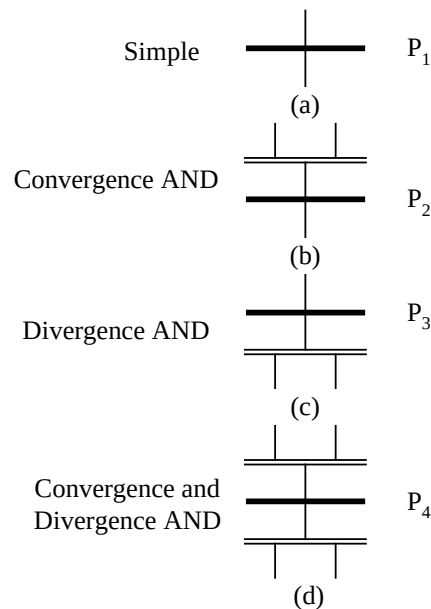


圖 2、步驟轉移的表示方式

如圖 3 所示為直接連結。一個直接連結永遠都是由一個步驟連結到一個步驟轉移或是由一個步驟轉移連結到一個步驟。當有兩個或多個連結到同一步驟時，它們將會聚集在一起，如圖 3(b)所示。當有兩個或多個離開同一步驟時，

它們將會有一個共同的出發點，如圖 3(c)所示。如果是由下方回到上方時則由一個帶有方向性箭頭的連結線來表示如圖 3(d)。

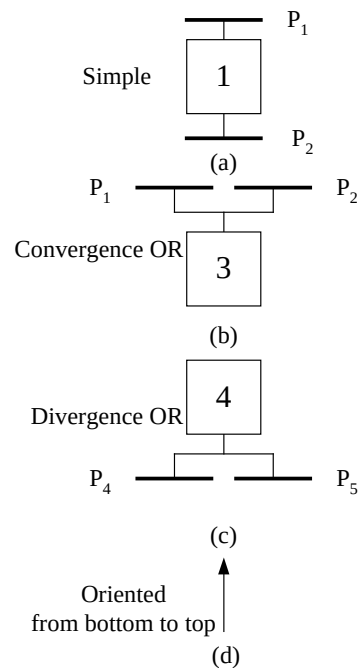


圖 3、直接連結的表示方式

B. 產生步驟轉移

一個步驟轉移的產生必須符合下列兩個條件：

- (1) 在步驟轉移之前的所有步驟狀態都必須是動作的。
- (2) 步驟轉移的 P_i 要為真。

如圖 4 所示為一個具有 4 個步驟與 3 個步驟轉移的 Grafcet 圖形，由(a)→(c)→(a)表達出 Grafcet 的步驟轉移情形。舉例來說當系統一開始時，系統為初始狀態步驟{0}，此時只有步驟轉移(0)被觸發，當 P_0 為真時則產生步驟轉移。產生步驟轉移時，步驟轉移之前的所有步驟狀態將轉變成不動作，而步驟轉移之後的所有步驟狀態將轉變成動作。當步驟轉移(0)觸發且 P_0 為真時狀態由步驟{0}轉變成步驟{1, 2}，當步驟轉移(1)觸發且 P_1 為真時狀態轉變成步驟{1, 3}，當步驟轉移(2)觸發且 P_2 為真時將會回到初始狀態。P 可以是個條件(Condition)如布林方程式(Boolean Function)、一個事件(Event)、或是包含了條件與事件。

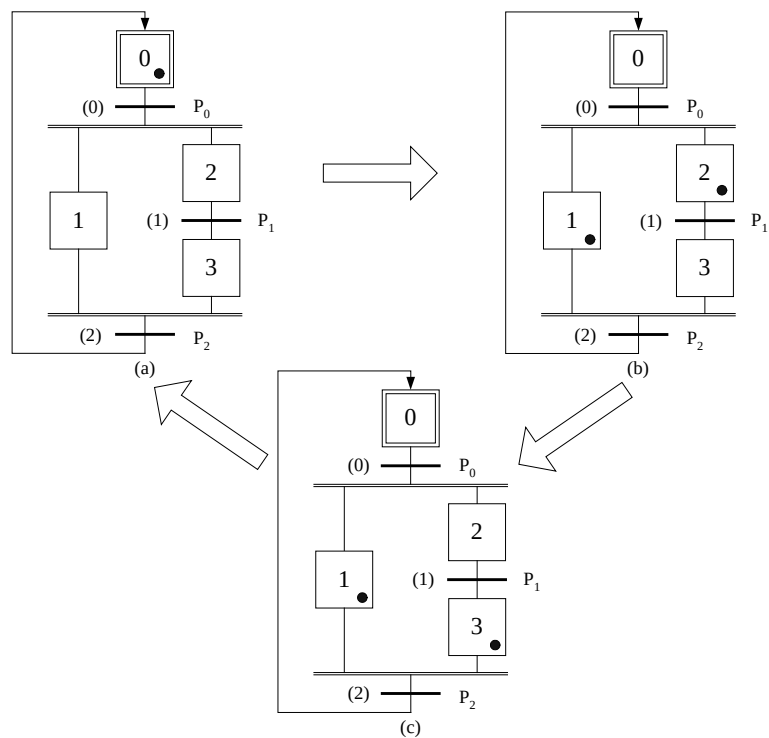


圖 4、Grafset 的步驟轉移情形

當 P 是一個條件時，如圖 5 所示，步驟轉移 $P_2=a$ ， a 為一個布林變數。當步驟{2}狀態為動作時步驟轉移(1)被觸發。如果步驟{2}狀態為動作且 $P_2=a=1$ 時步驟轉移(1)發生。如圖 5(a)所示，當 $a=0$ 時步驟{2}動作。當 $a=0$ 時步驟{2}會持續動作，直到 a 改變為 1 則步驟轉移發生。如圖 5(b)所示，當 $P_1=b=1$ 時步驟轉移(0)發生，步驟{2}動作。如果這時 $a=0$ 步驟{2}是穩定的。但如果 $a=1$ 步驟轉移(1)立即發生，步驟{3}動作，則步驟{2}處於不穩定狀態。

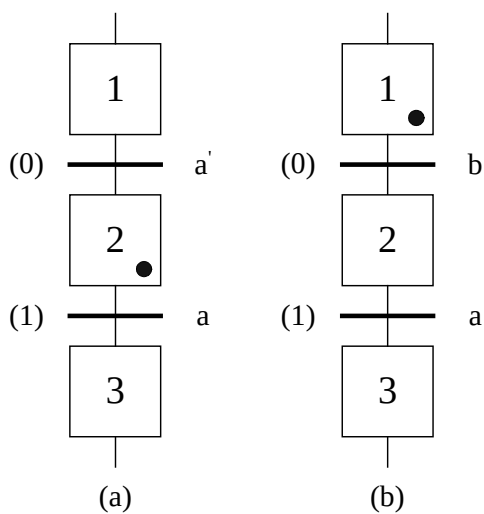


圖 5、 P 為條件 (a) $P_1=a'$ $P_2=a$ (b) $P_1=b$ $P_2=a$

當 P 是一個事件時，如圖 6 所示，步驟轉移 $P_2=\uparrow a$ 。當步驟{2}狀態為動作

時步驟轉移(1)被觸發。如果步驟{2}狀態為動作且事件 $\uparrow a$ 發生時步驟轉移(1)發生。如圖 6(a)所示，當 $a=0$ 時步驟{2}動作。步驟{2}會持續動作直到 $\uparrow a$ 發生，如 a 改變為 1。如圖 6(b)所示，當 $P_1=b=1$ 時步驟轉移(0)發生，步驟{2}動作。這時不論 $a=0$ 或 $a=1$ 都是穩定的。如果 $a=0$ 改變為 $a=1$ 步驟轉移(1)立即發生，步驟{3}動作。如果 $a=1$ 要產生 $\uparrow a$ 則必須先轉變 $a=0$ 後在轉變為 $a=1$ 。

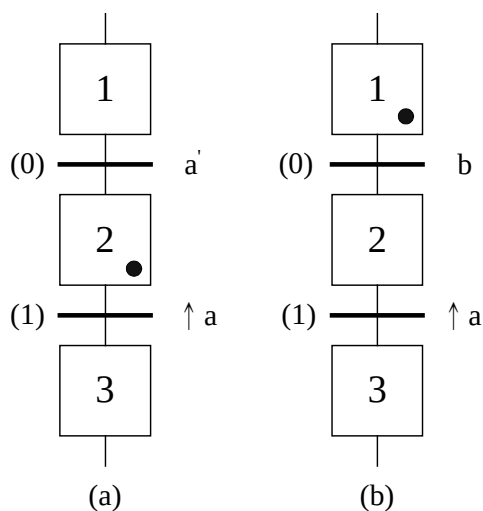


圖 6、P 為事件 (a) $P_1=a'$ $P_2=\uparrow a$ (b) $P_1=b$ $P_2=\uparrow a$

當 P 包含條件與事件時，如圖 7 所示，步驟轉移 $P_1=\uparrow a \cdot b$ 。當步驟{1}狀態為動作時步驟轉移(1)被觸發。如果步驟{1}狀態為動作，事件 $\uparrow a$ 發生且 $b=1$ 時步驟轉移(0)發生。步驟轉移(0)發生前一定是 $a=0$ 、 $b=1$ 。

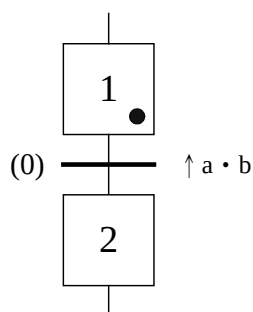


圖 7、P 包含條件與事件 $P_1=\uparrow a \cdot b$