

# 布林代數演算法在條件序控的應用以線控機器人操作為例

周立強<sup>1</sup> 程安邦<sup>2</sup>

- 1、 國立宜蘭技術學院生物機電工程系講師
- 2、 國立宜蘭技術學院生物機電工程系副教授

## 摘要

本文目的旨在以布林代數演算法中 Quine 與 McCluskey 氏所提出的列表法去推演條件序控中複雜的接點電路邏輯問題，並藉助 Matlab 程式加以求出並簡化輸入與輸出變數間的布林函數式，再以 EXCEL 電子試算表檢驗其結果的正確性。在應用上可藉由布林函數式轉換成接點邏輯電路，以配合可程式控制器轉換成其階梯電路圖。本文中係以參加第四屆全國技專院校創思設計與製作競賽之線控機器人的操作桿控制器為例，具體化此一構想。經實際測試該控制器均能正確無誤地達到所要求的動作功能，顯示此一方式的運用相當可行，可作為機電整合領域中條件順序控制教學上的一個範例與應用。

關鍵字：布林代數、接點邏輯電路、可程式控制器。

# **Application of Boolean Algebra Algorithm on Condition-Sequence Control —Case study of A Manual-driven Robot**

Li-John Jou<sup>1</sup> An-Pan Cherng<sup>2</sup>

1. Lecturer, Department of Biomechatronic Engineering,  
National Ilan Institute of Technology

2. Associate Professor, Department of Biomechatronic  
Engineering, National Ilan Institute of Technology

## **Abstract**

The objective of this paper was to demonstrate how Boolean algebra algorithms (Quine-McCluskey procedure) can be applied on deducing the complex logical problems of Condition-Sequence control switching circuits. A Matlab program was developed to resolve and simplify the Boolean relations between input and output variables. The results were then verified by an Excel program. In practical applications, the Boolean relations can be easily transformed to switching circuits and used in Programmable Logical Controller (PLC). A manual-driven robot, hand-made for the “4<sup>th</sup> Annual Contest of Creative Design among Technological Colleges” and controlled by a PLC, was finally presented to show how the idea works. The example can also serve as a case study for Condition-Sequence control course in mechatronics.

Keywords : Boolean algebra , switching circuit, programmable logical controller(PLC).

## 一、前言

布林代數是離散數學的一支，尤其在計算機科學與數位邏輯電路設計上扮演相當重要的角色[1]。在其他的應用方面，在1990年黃與蘇氏曾將布林代數用於電腦輔助製程設計的最佳化[2]。他們依一定的邏輯演算出每一加工物件的製造程序，並以布林代數結構來表示所有加工的順序與狀態，最後配合動態規劃以求出最佳的製造程序[2]。1993及1995年蘇氏等人也連續以布林代數邏輯來推導在列車運行時各種號誌顯示、速度與轉轍器反位關係，並研究出三種以布林代數式表示的模式，發展了鐵路自動控制數學模式[3,4]。1996年游與黃氏曾提出將布林代數融入至類神經網路與模糊理論的系統設計理念中[5]。爾後1995年Tseng與Huang提出以網路分割特性與交換邏輯理論，將前饋式類神經網路缺點改善，設計一邏輯函數類神經網路，使學習成功率提高及速度變快[6]。在2001年周與游氏曾將布林代數演算法則應用在避障車的可程式控制器階梯電路[7]。因此布林代數演算對於決定邏輯演譯路徑而言，是一個應用極廣的運算法則。在上述這些各領域應用當中，機械領域對於複雜的且條件式的致動器動作順序控制，存在著很多「on-off」機械式的接點動作。在過去大多憑經驗法則或套用組合式電路如自保、互鎖、「on-delay」或「off-delay」等加以設計。對於稍微複雜的邏輯順序動作，有時容易錯誤，往往耗時修改並一再測試。同樣地在氣壓迴路設計時的順序控制，直覺法也會碰到類似的問題，在1993年王氏曾根據邏輯設計中的卡諾圖法找出邏輯方程式作為各氣壓缸動作順序的基本判別原則[8]，同時該文亦提到邏輯設計的過程相當繁瑣，並以手算化簡且變數不能太多。本文的提出是以便於在機械領域中對於機電整合部份，有關於液氣壓控制，或者條件式的順序控制，在其接點邏輯的致動器控制動作法則中，找到一個可行方式。

至於其邏輯演算的簡化方面，李與周氏分別在1992與1994年就布林函數之最簡化及多輸出邏輯網路提出化簡方法[9,10]，經比較後發現較不適合發展成計算機程式，尤其是電子試算表EXCEL。事實上布林代數的化簡演算方法若以計算機完成則以Quine與McCluskey氏(以下簡稱Q&M)所提出的列表法較合適，因為此方法可解決卡諾圖(Karnaugh map)的限制[11,12](即輸入的變數不能超過6個)。列表法是以0、1數字碼做演算變數，經一連串重複性演算步驟最後再將0、1數字碼轉成符號變數，極適合計算機程式完成[11]。

因此本文即藉由布林代數的演算法(即Q&M的列表法)去推演條件式的順序控制中存在的複雜邏輯問題，並配合Matlab高階程式語言及Excel電子試算表的普及性，建立一套可程式化的演算過程，其中輔以的EXCEL電子試算表及Matlab程式是可正確求出並簡化輸入與輸出變數間的邏輯方程式。藉由此邏輯方程式可轉換成接點邏輯電路，以配合可程式控制器再轉換成其階梯電路圖，最後應用到控制致動器動作順序上。本文中係以線控機器人的操作桿控制器為例，加以具體化此一構想，經實地操作已驗證此構想之可行。

## 二、應用例背景說明

本節將依序地敘述順序控制中資訊與訊號的轉換關係、布林代數演算法的原理及線控機器人的操作

控制器，其應用例背景說明如下。

(一)、順序控制的資訊與訊號轉換關係

我們常用 0 與 1 的訊號來傳送一些物理現象，例如 0 是關 1 是開、0 是低電位 1 是高電位、0 是消磁 1 是激磁 ... 等，這些用來區別物理狀態的訊號叫做雙值訊號。順序控制所能處理的訊號為雙值訊號，但是訊號本身並不一定是雙值。例如電動機的運轉或停止可以用雙值訊號表示，但是如果顯示正轉跟反轉跟停止，則必須使用三值，稱為三值訊號。三值以上的訊號稱為多值訊號。在技術上為了處理多值訊號，是以雙值訊號的組合表示多值訊號與傳遞多值訊號。訊號的組合又可分類串聯訊號與並聯訊號，其間是可以互換的。一個訊號所佔的傳輸電路稱為波道。加入依時間順序來組合雙值信號的處理方式稱為串聯訊號。傳輸串聯訊號的大多是一個波道，倘若以此訊號形式傳輸資訊，在原理上會花費較多時間，有時為明確表示訊號開始或結束時的傳輸，亦可再加一波道。在本文中的例子是以數個雙值訊號的組合方式來表示多值資訊，並分別以不同的回路(在應用上即為可程式控制器之階梯電路圖)同時輸送各雙值訊號稱為並聯訊號。由於並聯訊號可分成擇一訊號與組合訊號，而組合訊號中的訊號值有各種組合方法，故以組合訊號傳輸資訊時可以少於資訊值之數的波道數來傳輸，不會像串聯訊號那樣耗時[13]。在順序控制中加入多個感測器輸入往往即形成有條件式的順序控制(以下簡稱條件序控)，並非固定式的動作順序，其間亦存在一些複雜的條件邏輯問題。並聯訊號間之變換迴路稱為組合迴路，要如何快速有效而正確獲得此一組合迴路，這就是我們所要解決的問題。組合迴路中可分成組合訊號與擇一訊號之轉換，在硬體上是可以單純地區隔開，但在軟體上由於人機介面操作上的人性化想法，的確交錯地混合在一起，本文中線控機器人的操作桿控制器就是此例。以下本文分別針對組合訊號與擇一訊號之轉換作一說明。

組合訊號之轉換，如圖 1 所示。例如有 5 個訊號輸入(A~E)，經過邏輯判斷輸出為 X。在判斷邏輯的方塊圖中，可進行某一邏輯判斷的電路往往不僅只有一種形式，有時還可以設計成另外不同型式的電路。在這種情形下，一方面保持邏輯判斷的內容，另一面可轉換成其他電路的操作稱之為邏輯電路的等效轉換，一般常有補數迴路、選擇迴路、二極體矩陣及迂迴電路。在本文例中由於採可程式控制器，因此邏輯電路的等效轉換僅應用到補數與選擇迴路二種。補數迴路實際上即是布林代數運算法則中狄摩根定理(De Morgan's Theorems)之應用，而選擇迴路則是將組合訊號(例如 A、B、C)轉換成擇一訊號(例如 X、Y、Z、U 及 V)後，再將輸出訊號作波道選擇輸出，如圖 2 所示。

至於擇一訊號之轉換是將擇一訊號變換為組合信號，是選擇迴路的反向轉換，例如圖 3 所示[13]。



圖 1 組合訊號之轉換



圖 2 組合訊號轉換成擇一訊號



圖 3 擇一訊號變換為組合信號

圖 4 4 個輸入與 4 個輸出訊號邏輯系統

(二)、布林代數演算法原理

以布林代數為基礎的演算法，在本文中即依據圖 4 所示的 4 個輸入訊號與 4 個輸出訊號的邏輯系統。在這樣類似的邏輯系統中，其輸入與輸出之間確實存在一轉換關係，此一轉換關係是可以表示成符號數學的運算式(即布林代數式)，加以求得。在建立符號數學運算式之前，必須先就輸入與輸出之對應關係以邏輯真值表建立，輸入訊號為 1 表示有輸入，0 表示沒有輸入，同理輸出訊號為 1、0 即分別表示有與沒有輸出。而根據輸入與輸出所對應的關係邏輯真值表，其基本演算式一般可表示成「極小項之和(sum of minterm)」或「積項之和(sum of product)」與「極大項之積(product of maxterm)」或「和項之積(product of sum)[14,15]。前者是把輸出訊號變數  $Y_i$  出現邏輯值為 1 的所有組輸入訊號變數的邏輯「積」(AND) 運算相加，輸入訊號變數的邏輯值為 0 視為反相變數，1 則為非反相變數。後者是把輸出訊號變數  $Y_i$  出現邏輯值為 0 的所有組輸入變數的邏輯「或」(OR) 運算相乘，輸入訊號變數的邏輯值為 0 視為非反相變數，1 則為反相變數。在所有邏輯真值表中，輸出訊號變數的邏輯值為 0 的項數多於邏輯值為 1 的項數時，採用分隔正規形式(disjunctive normal form,dnf)即「積項之和」之演算方式較為方便，反之輸出訊號變數的邏輯值為 0 的項數少於邏輯值為 1 的項數時，採用連接正規形式(conjunctive normal form,cnf) 即「和項之積」之演算方式[14,15]。

「積項之和」可寫成如下型式

$$Y_i = f_{dnf} \left( \prod_{i=1}^b \tilde{x}_i \right) = \sum_{j=0}^{2^b-1} \left( \prod_{i=1}^b \tilde{x}_i \right) = \sum_{j=0}^{2^b-1} m_j \dots \dots \dots (1)$$

$Y_i$ ：為一 n 個輸出訊號的變數。

$f_{dnf}$ ：為分隔正規形式函數

：係指輸入訊號變數的個數。

$\tilde{x}_i$ ：是指輸入訊號變數為非反相變數  $X_i$  或反相變數  $\overline{X_i}$ 。

$m_j$ ：係指使輸出訊號變數邏輯值為 1 的每組輸入變數「積」(AND)的邏輯運算式。

「和項之積」可寫成如下型式

$$Y_i = f_{cnf} \left( \sum_{i=1}^b \tilde{x}_i \right) = \prod_{j=0}^{2^b-1} \left( \sum_{i=1}^b \tilde{x}_i \right) = \prod_{j=0}^{2^b-1} m_j \dots \dots \dots (2)$$

$Y_i$ ：為一 n 個輸出訊號的變數。

$f_{cnf}$ ：為連接正規形式函數

：係指輸入訊號變數的個數。

$\tilde{x}_i$ ：是指輸入訊號變數為非反相變數  $X_i$  或反相變數  $\overline{X_i}$ 。

$m_j$ ：係指輸出訊號變數邏輯值為 0 所對應的輸入變數「或」(OR)的邏輯運算式。

早在 1938 年 Shannon 即以符號數學與邏輯(symbolic mathematics and logic)來演算化簡接點電路(relay and switching circuits)[16]，在後來的應用於 1956 年 McCluskey 發表「布林函數之最小化」(Minimization of Boolean Function) [17]即以列表的方法來化簡邏輯系統中輸入與輸出間布林函數式，可解決接點邏輯電路簡化問題。

### (三)、線控機器人的操作控制器

本文中線控機器人包含母車與子車兩部分。當母車到達適當地點時可以釋放子車前進至得分區。母車在前進的過程中必須完成諸如障礙排除與重置、取旗插旗等動作，這些動作均以控制直流馬達正反轉傳動收拉繩索以帶動連桿機構方式完成。至於行走部分，母車與子車均採前輪驅動，左右兩輪分別各以直流馬達傳動並控制其正、反轉及停止來達到直走、轉彎及後退等動作。母體後輪輔以兩個惰輪行走，子車則是以一個惰輪行走並著地，以減輕母體重量，故母體底盤設計很低（僅離地 2 公分），以方便子車釋放時的脫離。在其操作控制器是採用電動玩具中操縱搖桿加以改裝。由於操縱桿四個位置的極限開關（分別為前、後、左、右）可相當於四個輸入狀態的感測器，為有限的且離散化的狀態，可分別或同時操縱機器人行進方向。為避免操作時誤觸造成致動器馬達迴路短路，因此彼此間互鎖相當重要。我們採離散數學中布林代數演算法則推演出交換函數(switching function)[13]，經 Quine 與 McCluskey 氏所提出的列表法[11, 15, 17, 18]可化簡其布林代數式再轉換成邏輯電路(logic network)[14, 15]，最後再轉換成程式控制器階梯電路圖加以應用成線控機器人的操作控制器，解決傳統線路邏輯開關之間互動與互鎖電路難懂的部分，簡化配線問題。該系統為 4 個輸入訊號 A、B、C、D，分別代表操縱桿四個方位的前後左右。4 個輸出訊號  $Y_1$ 、 $Y_2$  與  $Y_3$ 、 $Y_4$  分別是控制行走部左與右馬達的輸出，以示意圖 4 表示。圖 5 所示為線控機器人之全貌。



圖 5 線控機器人行走部之實際運動操作測試(母車排除障礙)

## 三、材料與方法

本節中即根據 Quine & McCluskey 氏所提出的列表法要義寫成 Matlab 程式，其演算程序說明如下。以配合下節的實例說明。

符號說明(Notation)：

大寫黑體字母為矩陣，小寫黑體字母為向量，同一個字母的小寫為大寫的列向量，其餘字母為純量，希臘字母為單一系列向量(不是某矩陣的部分)或是純量。分號表示行向量的疊加，下標(subscript)表示元素的排列代號，[]為空矩陣或空向量。

積項之和化簡演算法

Input : 含有 N 個十進位元素的行向量  $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]$  以及指定的位元數 M, 其中  $\max_{i=1}^N \{x_i\} < 2^M$ 。

Output : 含 M 行的矩陣 W, 其中每一列向量為經化簡之後的[積項之和]二進位代碼, 各列之間為互質, 無法再合併化簡。

$$\text{Step 1 : Set } x_i = \sum_{j=1}^M q_{ij} 2^{j-1}, \text{ 建立一矩陣 } \mathbf{Q}_{N \times M} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \dots \\ \mathbf{q}_N \end{bmatrix} = \begin{bmatrix} q_{11} & q_{12} & \dots & q_{1M} \\ q_{21} & q_{22} & \dots & q_{2M} \\ \vdots & \vdots & \vdots & \vdots \\ q_{N1} & q_{N2} & \dots & q_{NM} \end{bmatrix}, q_{ij} \in \{0,1\}$$

(將十進位轉為二進位並以 [0, 1] 方式儲存於矩陣 Q)

Step 2 : Set  $m=N, U=[]$  (設定初始值)

Step 3 : for  $k=1:m$  do steps 4-12 (分離孤立列與合併列, 最多共需 m 次)

Step 4 : Set  $m=\text{the number of rows of } Q$  (Q 的總列數)

Set  $\mathbf{a}_i = \|\mathbf{q}_i\|_1 = \sum_{j=1}^M q_{ij}$  (即  $L_1$  模數 ( $L_1$  norm), 同列元素之和即為含 1 元素之和)

Set  $\mathbf{b}=0$  for  $i=1:m$  ( $\mathbf{b}$  為判別是否為孤立列或配對列的指標, 0 為孤立列)

Step 5 : for  $i=1:m-1$  do steps 6-10 (逐列比較)

Step 6 : for  $j=i+1:m$  do steps 7-10

Step 7 : if  $|\mathbf{a}_i - \mathbf{a}_j| = 1$  (恰巧含 1 的元素數目相差一個) then (do steps 8-10)

step 8 :  $\lambda_r = \text{NAND}(q_{ir}, q_{jr})$  for  $r=1:M$  (bit-by-bit 逐元素比較是否相異)

step 9 : if  $\sum_{r=1}^M \lambda_r = 1$  where  $\lambda_l = \delta_{rl}$  then do step 10

(恰巧只有第 l 個元素相異)

step 10 : Set  $\mathbf{b}_i=1$  and  $\mathbf{b}_j=1$  (第 i、j 列都標註為配對列)

Set  $n=n+1$  (可以合併成新一列)

$$\text{Set } s_{nr} = \begin{cases} -1 & \text{if } r=l \\ q_{ir} & \text{otherwise} \end{cases} \text{ for } r=1:M$$

(產生新的合併列  $s_n$ , 元素為  $s_{nr}$ , 第 l 個元素改為 -1, 其餘不變)

step 11 :  $\bar{\mathbf{S}} \leftarrow \mathbf{S}$  wher  $\bar{s}_i \neq s_j$  if  $i \neq j$

(合併列可能有重複的, 故需要剔除多餘的組合)

step 12 : Set  $\mathbf{U} = [\mathbf{U}; \mathbf{q}_i]$  if  $\mathbf{b}_i = 0$  for  $i=1:m$  (儲存孤立列的矩陣)

Set  $\mathbf{Q} = \bar{\mathbf{S}}$  (合併列成為下一輪可以再化簡的新集合)

Step 13 : Set  $W = [\mathbf{U}; \mathbf{S}]$  (每一輪淘汰的孤立列加上最後一輪的合併列就是簡化的結果)

Step 14 : OUTPUT(W)

以 Matlab 程式應用可解決日趨複雜的邏輯變數冗長計算，除了快速而有效地求出布林函數式，並可以電子試算表檢驗其正確性，最後依此繪出接點邏輯電路再轉換成可程式控制器的階梯電路圖，達到實用化的目的。

## 四、實例說明

本節主要分成演算法推演過程、布林代數函數式轉換成接點邏輯電路、可程式控制器外部線路與階梯電路圖撰寫以及實測結果四個流程。以下將各流程進行方式內容分述如下：

### (一)、演算法推演過程

#### 1、輸入狀態定義描述

操縱桿前後左右四個位置的極限開關共 4 點，因此其狀態輸入點對可程式控制器而言共有 4 點輸入並形成  $2^4=16$  種狀態組合。當輸入接點閉合，定邏輯值為 1，開啟則為 0。而各輸入點所定義的邏輯變數及對照涵義 A、B、C、D 分別是操縱桿前、後方、右、左側位置的極限開關。此定義當操縱桿向前、向後、向右及向左碰觸對應之極限開關意味對車體分別作前進、後退、原地順時針轉及逆時針轉方向控制，如同表 1 欄位(1)序號的 1、2、4、8 是屬擇一訊號輸入。同理，操縱桿向左後、右後、左前及右前碰觸即表示控制車體作左後、右後、左前及右前的方向控制，如同欄位(1)序號的 5、6、9、10 是屬組合訊號輸入。這樣的定義方式是考慮到人機介面操作上的人性化想法並減少輸入訊號的波道數。因此根據上述定義可先建立輸入狀態的真值表與其所對照的方向控制定義，如表 1 中欄位(2)及(3)所示。表 1 中 4 個輸入狀態可視為二進位中的 4 位元，而欄位序號(1)則表示其十進位的轉換，共有 16 種狀態組合。然而有些狀態是不存在的，例如操縱桿不可能同時前後或左右，所以這種情形將被定義成無關狀況(即停止)，如表 1 中序號 3、7、11~15 即是。

#### 2、對應策略的形成與輸入出真值表建立

根據表 1 中欄位(2)及(3)所定義的輸入狀態的真值表與其所對照的方向控制，再轉換成底盤左右馬達正反轉方向，其中欄位(4)為三值之訊號輸出，1 代表正轉，-1 代表反轉，0 表示停止。將此三值訊號轉換成雙值訊號的組合即欄位(5)之  $Y_1$  及  $Y_2$ 、 $Y_3$  及  $Y_4$  分別是控制左右馬達正反轉的輸出變數，而真值表中排列順序 1 及 0、0 及 1、0 及 0 分別是代表馬達正轉、反轉、停止之意。依據表 1 欄位(2)及(5)欄位可演算出輸出變數與輸入變數的布林函數式。此布林函數式所表現出來就是並聯訊號的組合迴路，複合了補數、選擇及擇一迴路。

#### 3、演算布林代數表示式

依前述所反映出的邏輯真值表，本例中依前述運算原則[6, 10]，採積項之和運算方式，依(1)式及表 1 輸出變數真值表可分別寫出  $Y_1$ 、 $Y_2$ 、 $Y_3$ 、 $Y_4$  輸出變數基本演算式

$$Y_1 = m_2 + m_8 + m_{10} \dots\dots\dots (4-a)$$

$$Y_2 = m_1 + m_4 + m_6 \dots\dots\dots (4-b)$$

$$Y_3 = m_1 + m_8 + m_9 \dots\dots\dots (4-c)$$

$$Y_4 = m_2 + m_4 + m_5 \dots\dots\dots (4-d)$$

以(4-a)~(4-d)式為例， $m_1$ 、 $m_2$ 、 $m_4$ 、 $m_5$ 、 $m_6$ 、 $m_8$ 、 $m_9$ 、 $m_{10}$  所代表的每組輸入變數「積」(AND)的邏輯運算，因此表示如下：

$$m_1 = \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot D \dots\dots\dots (5)$$

$$m_2 = \overline{A} \cdot \overline{B} \cdot C \cdot \overline{D} \dots\dots\dots (6)$$

$$m_4 = \overline{A} \cdot B \cdot \overline{C} \cdot \overline{D} \dots\dots\dots (7)$$

$$m_5 = \overline{A} \cdot B \cdot C \cdot D \dots\dots\dots (8)$$

$$m_6 = \overline{A} \cdot B \cdot C \cdot \overline{D} \dots\dots\dots (9)$$

$$m_8 = A \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} \dots\dots\dots (10)$$

$$m_9 = A \cdot \overline{B} \cdot C \cdot D \dots\dots\dots (11)$$

$$m_{10} = A \cdot \overline{B} \cdot C \cdot \overline{D} \dots\dots\dots (12)$$

表 1 線控操作與方向控制所對應的輸入出真值表

序號	輸入變數				應對策略	輸出變數					
	(2)		(3)			(4)		(5)			
(1)	前	後	右	左	方向控制	左馬達	右馬達	左馬達		右馬達	
m	A	B	C	D		轉向	轉向	Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	Y <sub>4</sub>
0	0	0	0	0	停止	0	0	0	0	0	0
1	0	0	0	1	原地左轉	-1	1	0	1	1	0
2	0	0	1	0	原地右轉	1	-1	1	0	0	1
3	0	0	1	1	停止	0	0	0	0	0	0
4	0	1	0	0	後退	-1	-1	0	1	0	1
5	0	1	0	1	左後退	0	-1	0	0	0	1
6	0	1	1	0	右後退	-1	0	0	1	0	0
7	0	1	1	1	停止	0	0	0	0	0	0
8	1	0	0	0	前進	1	1	1	0	1	0
9	1	0	0	1	左前進	0	1	0	0	1	0
10	1	0	1	0	右前進	1	0	1	0	0	0
11	1	0	1	1	停止	0	0	0	0	0	0
12	1	1	0	0	停止	0	0	0	0	0	0
13	1	1	0	1	停止	0	0	0	0	0	0
14	1	1	1	0	停止	0	0	0	0	0	0
15	1	1	1	1	停止	0	0	0	0	0	0

將(5)~(12)式分別代入(4-a)~(4-d)式，以布林代數演算法則中 Quine & McCluskey 氏所提出的列表法可化至最簡式，此處本文將列表法寫成 Matlab 程式[19]，以 0、1 數字碼做演算變數並加快演算速度，最後再將 0、1 數字碼轉成符號變數如(13)~(16)式所示。其部分程式畫面結果，以 Y<sub>1</sub> 為例，如圖 6 所示為 Matlab 程式電腦螢幕所運算畫面結果。

$$Y_1 = \overline{B} \cdot \overline{D} \cdot (C + A) \dots\dots\dots (13)$$

$$Y_2 = \overline{A} \cdot (\overline{B} \cdot \overline{C} \cdot D + B \cdot \overline{D}) \dots\dots\dots (14)$$

$$Y_3 = \overline{B} \cdot \overline{C} \cdot (D + A) \dots\dots\dots (15)$$

$$Y_4 = \overline{A} \cdot (\overline{B} \cdot C \cdot \overline{D} + B \cdot \overline{C}) \dots\dots\dots (16)$$

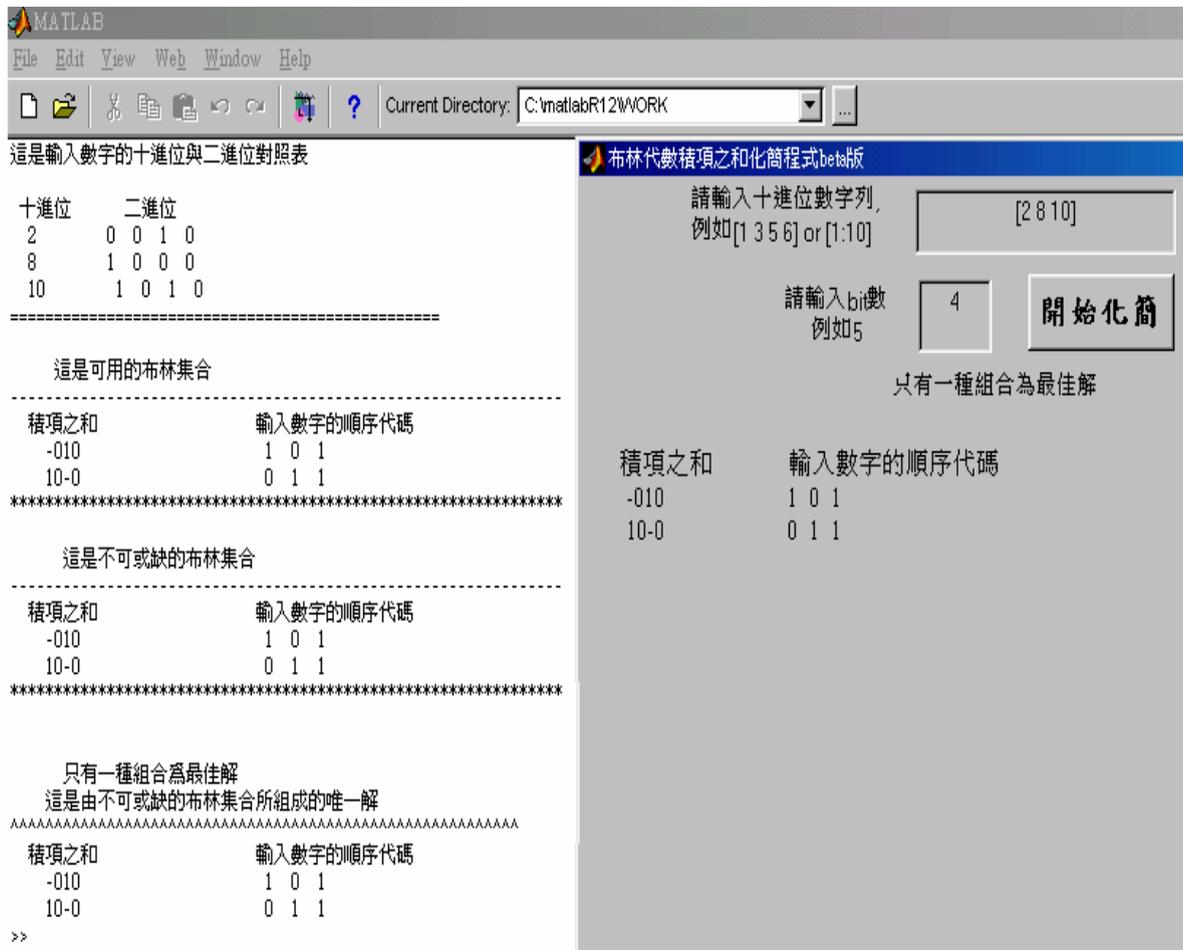


圖 6 為 Matlab 程式電腦螢幕所運算畫面結果。

#### 4、檢驗布林代數表示式

為避免演算錯誤，可將演算出的(13)~(16)式布林代數函數式輸入至 EXCEL 電子試算表，以其邏輯函數功能運算其布林代數式的邏輯值(如圖 7 中 EXCEL 電子試算表中欄位 K~N 所示)，以比對是否與先前所制定之真值(如圖 7 中 EXCEL 電子試算表中欄位 G~J 所示)相同，若完全相符則表示演算無誤(如圖 7 中 EXCEL 電子試算表中欄位 O~R 所示)。

#### (二)、布林代數函數式轉換成接點邏輯電路

完成布林代數式的檢驗後，才分別將(13)~(16)式轉換成接點的邏輯電路，如圖 8 所示。

#### (三)、可程式控制器外部線路與階梯電路圖撰寫

先規劃製作可程式控制器外部線路，如圖 9 所示，再根據布林代數函數式所轉換的接點邏輯電路再編輯成可程式控制器的階梯電路圖，如圖 10 所示。

$$N(\text{AND}(\text{NOT}(B3), \text{OR}(\text{AND}(\text{NOT}(C3), \text{NOT}(D3), E3), \text{AND}(C3, \text{NOT}(E3))))))$$

$$\Leftarrow Y_2 = \bar{A}(\bar{B}CD + B\bar{D})$$

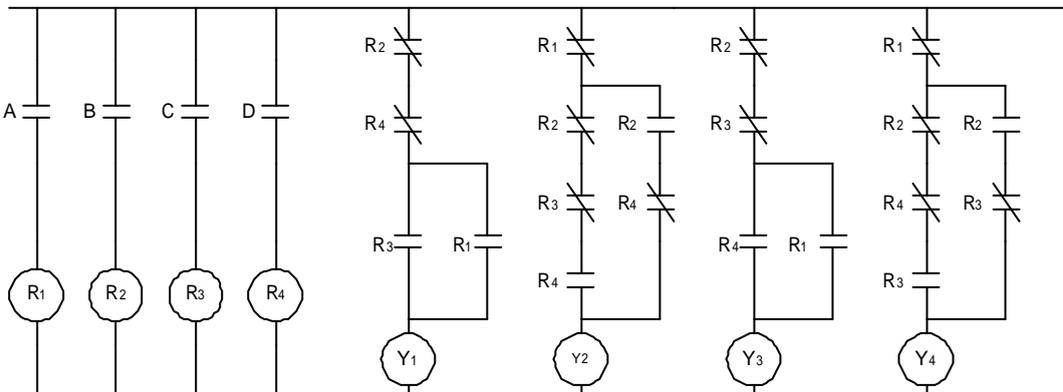
$$N(\text{AND}(\text{NOT}(C3), \text{NOT}(E3), \text{OR}(B3, D3)))$$

$$\Leftarrow Y_1 = \bar{B}\bar{D}(A + C)$$

EXCEL 試算表																		
=N(AND(NOT(C3),NOT(E3),OR(B3,D3)))																		
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	操縱桿					左馬達	右馬達	布林方程式輸出				檢驗布林方程式						
2	序號	A	B	C	D	方向控制	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4	Y1	Y2	Y3	Y4
3	0	0	0	0	0	停止	0	0	0	0	0	0	0	0	ok	ok	ok	ok
4	1	0	0	0	1	原地左轉	0	1	1	0	0	1	1	0	ok	ok	ok	ok
5	2	0	0	1	0	原地右轉	1	0	0	1	1	0	0	1	ok	ok	ok	ok
6	3	0	0	1	1	停止	0	0	0	0	0	0	0	0	ok	ok	ok	ok
7	4	0	1	0	0	後退	0	1	0	1	0	1	0	1	ok	ok	ok	ok
8	5	0	1	0	1	左後退	0	0	0	1	0	0	0	1	ok	ok	ok	ok
9	6	0	1	1	0	右後退	0	1	0	0	0	1	0	0	ok	ok	ok	ok
10	7	0	1	1	1	停止	0	0	0	0	0	0	0	0	ok	ok	ok	ok
11	8	1	0	0	0	前進	1	0	1	0	1	0	1	0	ok	ok	ok	ok
12	9	1	0	0	1	左前進	0	0	1	0	0	0	1	0	ok	ok	ok	ok
13	10	1	0	1	0	右前進	1	0	0	0	1	0	0	0	ok	ok	ok	ok
14	11	1	0	1	1	停止	0	0	0	0	0	0	0	0	ok	ok	ok	ok
15	12	1	1	0	0	停止	0	0	0	0	0	0	0	0	ok	ok	ok	ok
16	13	1	1	0	1	停止	0	0	0	0	0	0	0	0	ok	ok	ok	ok
17	14	1	1	1	0	停止	0	0	0	0	0	0	0	0	ok	ok	ok	ok
18	15	1	1	1	1	停止	0	0	0	0	0	0	0	0	ok	ok	ok	ok

$$\text{IF}(\text{delta}(J18, N18), \text{"ok"}, \text{"no"})$$

圖 7 布林代數函數式輸入至 EXCEL 電子試算表以比對驗算其輸出變數邏輯值之真值表



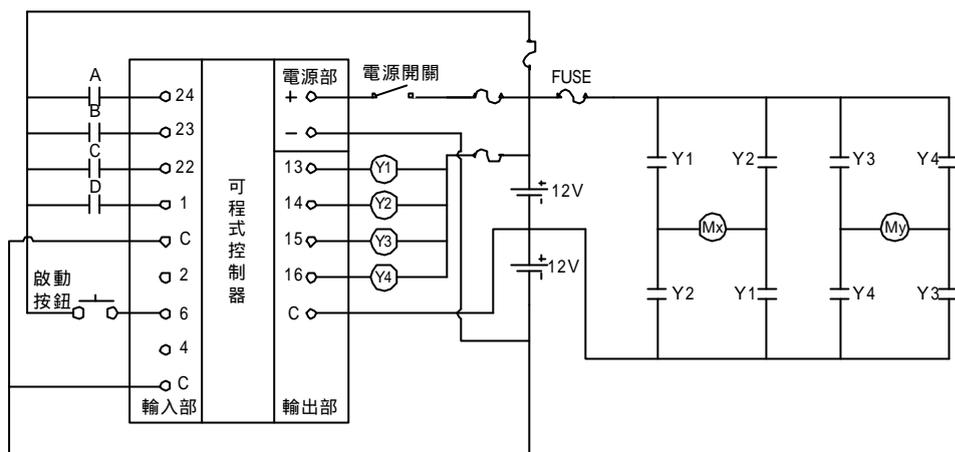
符號說明：

A~D：表示操縱桿四個方向前後右左控制的輸入接點，實際為極限開關。

R1~R4：為 A~D 四個輸入接點所對應的輔助繼電器規格是 MY-4N。

Y1~Y4：為分別控制左右馬達正反轉的輔助繼電器。

圖 8 將布林代數函數式轉換成接點的邏輯電路圖



符號說明：

A~D：表示操縱桿四個方向前後左右控制的輸入接點(實際為極限開關)，對應為 PLC 的輸入接點 Xin(X0024、X0023、X0022 及 X0001)。

Y1~Y4：為分別控制左右馬達正反轉的輔助繼電器，對應為 PLC 的輸出接點 Yout (Y0013、Y0014、Y0015 及 Y0016)。

Mx：行走部左馬達      My：行走部左馬達

圖 9 可程式控制器外部線路

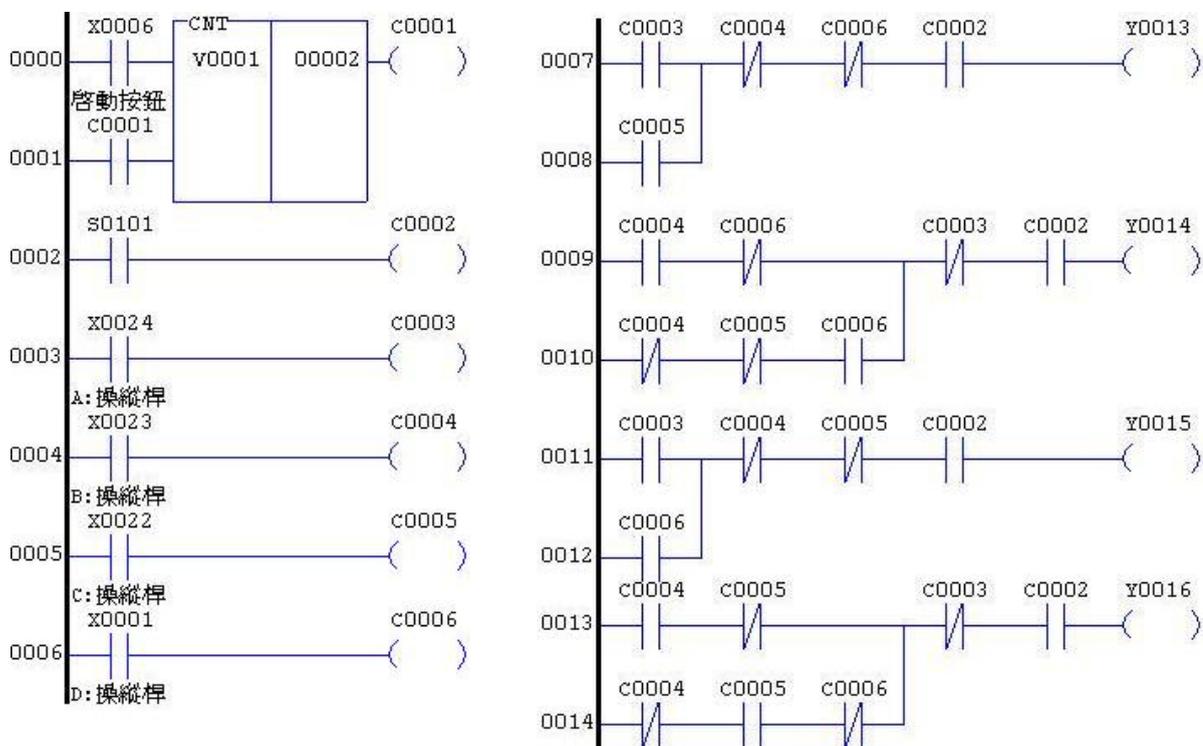


圖 10 接點邏輯電路轉換成可程式控制器的階梯電路圖

#### (四)、實測結果

本例中先以指撥開關作為可程式控制器之輸入並對照真值表模擬輸入，再比對輸出狀態均與真值表輸出完全符合。之後聯接線控機器人行走部負載，如圖 5 所示，經實際方向操作測試均無誤。線控機器人行走部有四個方位極限開關，共會有  $2^4=16$  種的組合狀態。而方向控制區分為 9 種動作，故輸入中有 7 種組合未有定義即視為沒有輸出，邏輯值視為 0。

#### 五、結論與建議

本例中若使用直覺式經驗以組合順序控制中的各應用電路(如自保、互鎖、串聯優先等)或者以流程圖法方式，則會有許多盲點。同樣地在氣壓迴路設計時的順序控制，直覺法也會碰到類似的問題。根據邏輯設計中的卡諾圖法找出邏輯方程式作為各氣壓缸動作順序的基本判別原則，其過程相當繁瑣，並以手算化簡且變數不能太多[8]。所以本文就利用布林代數中 Q&M 列表法作為推演多輸入變數與多輸出變數間所存在的布林邏輯關係式，並將其演算過程予以程式化，以轉換成接點邏輯電路(switching network)，再將接點邏輯電路轉換成可程式控制器的階梯電路。其中為檢驗所推演的布林邏輯關係式之正確與否，本文利用 EXCEL 電子試算表中邏輯函數功能指令來檢驗演算結果的真值表與原始真值表所制定的結果是否相符，這樣可保證其正確性且電路可化簡。由於控制策略是透過真值表所定出，只要知道輸入狀況，再對應真值表即可預測動作結果，如此一來推演較複雜的階梯電路可節省大量時間。

本文目的是為了要證明運用布林代數的觀念作為接點邏輯電路的演算基礎是可行的，所以我們就僅採用 4 個方位的極限開關作為輸入基準，以驗證此方法的可行性。當然增加極限開關數目可使線控機器人動作操作更多元(例如對其他上半部機構的致動器控制)且對方向控制的解析度提高許多。但同時也會使得演算過程變得更為繁複，因此本文提供一 Matlab 語言之計算機程式來取代整個演算過程，可以更快速且正確地推演出多輸入與多輸出系統的複雜接點邏輯關係。

對於需要加上計時、計數及記憶等功能之序控電路，此方式所演算之接點邏輯電路可作為序控電路之主要核心，而以群組電路方式加入計時、計數及記憶功能電路，可解決更為複雜的條件序控接點邏輯電路。在線控電路方面的製作上，採布林代數演算法觀念去推演可程式控制器之階梯電路圖並將布林代數表示式運算化簡，再與傳統繼電器迴路的線路邏輯混合，可大量減少了繼電器數目及線路數[20]，換言之減少電路設計的基本元件[11]，並解決開關之間互動與互鎖電路難懂部分。此外該演算法可配合 Matlab 語言之計算機程式及 EXCEL 電子試算表中邏輯函數功能可作到列式、運算及檢驗。除本例的應用外，避障車、遙控車、線控機器人與自走子車製作[7,20]於本系學生專題製作均顯示出此一方式的應用，對於機電整合中條件式的順序控制課程極具教學意義。

#### 參考文獻

- 1、Buttner, W., and H. Simonis, (1987) "Embedding Boolean Expressions into Logic Programming", Journal of Symbolic Computation", Vol.4, pp.191-205.
- 2、黃智傑，蘇純繒，(1990)「應用語法外形辨識器之電腦輔助製程設計」，七十九年度中國工業工程學會年會論文集，台北，第 247-256 頁。

- 3、蘇昭旭，陳木松，(1993)「鐵路自動控制數學模式之研究與分析」，大葉學報第二卷，第一期，第 33-48 頁。
- 4、張有恆，蘇昭旭，(1995)「台鐵裝設列車自動控制系統以改善鐵路行車安全之研究」，中華民國第二屆運輸安全研討會論文集，新竹，第 172-181 頁。
- 5、游寶達，黃仁竝，(1996)「類神經網路與模糊理論的系統設計理念」，模糊系統學刊第二卷，第一期，第29-36頁。
- 6、Tseng, L.Y., and T.H. Huang,(1995) “The Logic Function Neural Network”, Proceedings of the National Science Council(Part A : Physical Science and Engineering), Vol.19(6),pp.604-613.
- 7、周立強，游哲文，(2001)「避障車方向控制器的製作」，宜蘭技術學報第七期，第 67-84 頁。
- 8、王丁，(1993)「以文字判別法為氣壓邏輯迴路直覺設計法之分析研究」，勤益學報第 11 期，第 1-14 頁。
- 9、李肇廉，(1992)「布林函數最簡化之研究」，勤益學報第九期，第 131-139 頁。
- 10、周貴生，(1994)「多輸出邏輯網路化簡方法研究」，明新學報第十二期，第 63-67 頁。
- 11、胡咸，(1998)離散數學，第 95-118 頁，國立編譯館，台北。
- 12、Ross,Kenneth A.,and Charles R. B. Wright，(1999) “Discrete Mathematics”，4<sup>th</sup> edition, pp.517-524, Prentice Hall Inc., New Jersey.
- 13、楊水枝，(1992)自動控制，第 31-62，73-79 頁，正文書局，台北。
- 14、張康，張茂盛，(1997)離散數學，第 271-311 頁，東華書局，台北。
- 15、Rosen, Kenneth H., (1999) “Discrete Mathematics and Its Applications”，4<sup>th</sup> edition, pp.593-617, WCB/McGraw-Hill Co., Singapore.
- 16、Shannon,C.E., (1938)”A Symbolic Analysis of Relay and Switching Circuits”,Trans.AIEE,Vol.57,pp.713-723.
- 17、McCluskey E.J.,(1956)” Minimization of Boolean Function”，Bell System Tech., Vol.35(6),pp.1417-1444.
- 18、Norris,(1994)Fletcher R., “Discrete Structures: an introduction to mathematics for computer science”，pp.109-133, Prentice Hall Inc., New Jersey.
- 19、Matlab R12 Users Guide(2000) ,The Mathworks Inc.,U.S.A.
- 20、周立強，高易宏，吳健昌，(2001)「線控機器人與自走子車製作在教學上之應用」，宜蘭技術學報第六期，第 85-101 頁。