



SYNTEC
TECHNOLOGY CO.,LTD.

PLC發展工具操作手冊

匯出日期：2026-07-20

修改日期：2026-07-20

1 PLC編輯器操作介紹

1.1 軟體下載

1. 進入新代網站<http://www.syntecclub.com.tw>，完成線上註冊手續並確定權限開通。
2. 至『下載中心』 => 『04.應用工具』 => 『Ladder Editor』 下載PLC編輯器。

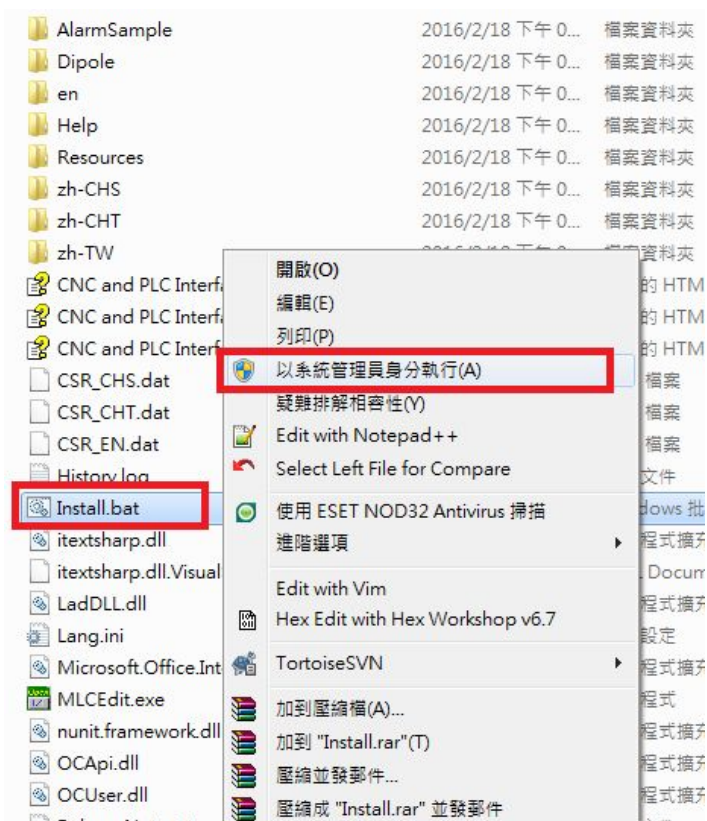
The screenshot shows the SYNTEC website's 'Customer Support' (客戶支援) section. Under the 'Download Center' (下載中心) tab, the breadcrumb path is 'Home > Customer Support > Download Center'. The filters are set to '04. Application Tools' (04.應用工具), 'Ladder Editor', and '1. Official Version' (1.正式版). The 'Show latest five versions' (顯示最新五個版本) dropdown is also visible. The version number '2.33.0' is displayed. Below this, a table lists the available download files.

檔案名稱	描述
LadEditor_2.33.0.zip	LadEdit(Zip) CHT

1.2 軟體安裝

- 2.26.3 (含) 以前為安裝版
 - 執行LadEdit_vx.xx.x.msi並依畫面指示，即可完成安裝程式。
 - 安裝完成後，依序點選『開始』 => 『程式集』 => 『OpenCNC』 => 『MLCEdit.exe』，即可開啟PLC編輯器。
- 2.29.0 (含) 以後為免安裝版
 - a. 第一次執行時：
 - i. 下載完LadEditor_x.xx.x.zip，解壓縮後右鍵點擊『Install.bat』，選擇『以系統管理員身分執行』，將.lad檔的預設開啟程式設定為PLC編輯器

ii. 執行『MLCEdit.exe』 開啟PLC編輯器



b. 第二次之後執行：執行『MLCEdit.exe』，或直接雙擊.lad檔，皆可開啟PLC編輯器

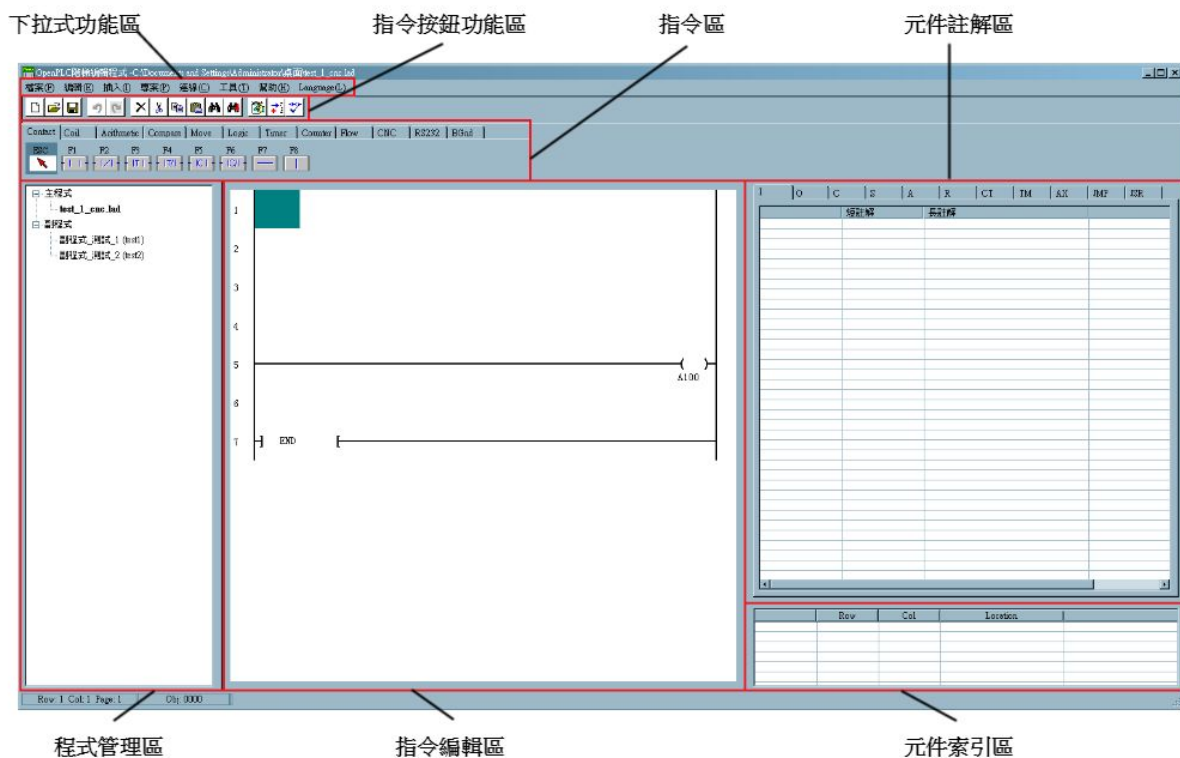
c. 注意事項：

- i. 搬移免安裝包位置時，請再次執行免安裝包內的『Install.bat』，否則會無法正常開啟.lad檔。
- ii. 下載新版本PLC編輯器時，請執行新版本免安裝包內的『Install.bat』，否則仍會以舊版PLC編輯器開檔。

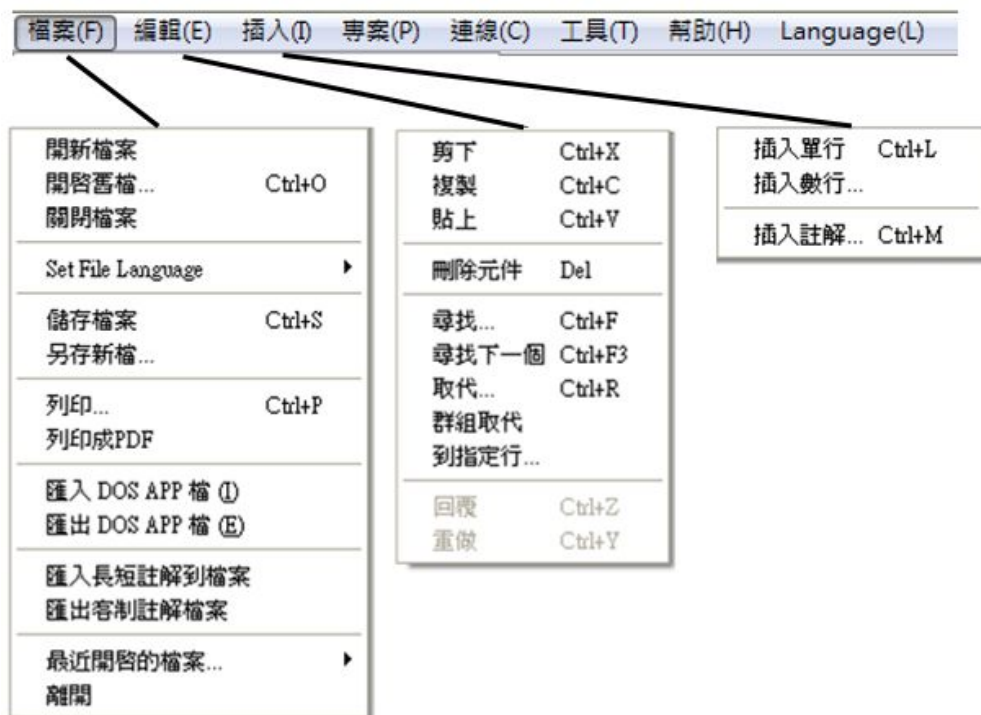
1.3 介面操作

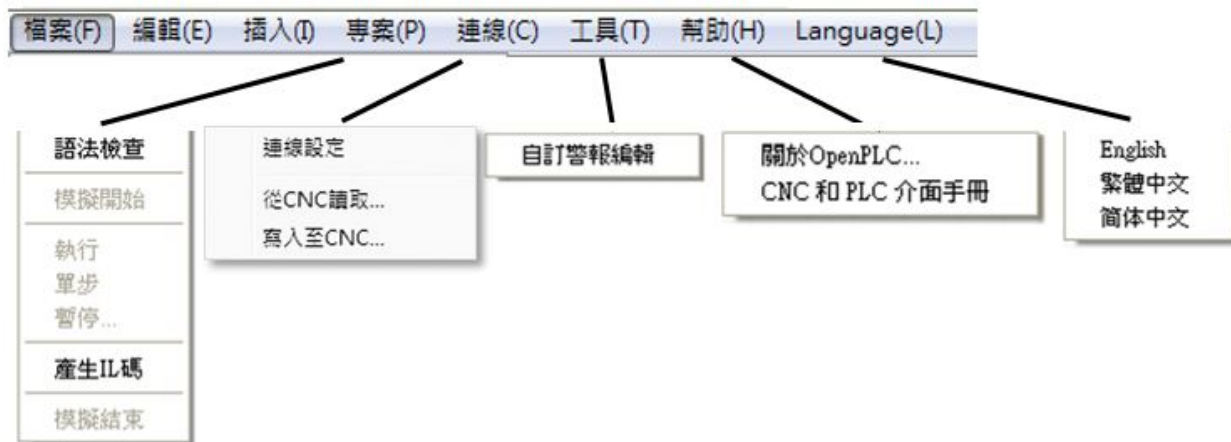
PLC編輯器介面共有七個區域，如下圖所示：

- 下拉式功能區
- 指令按鈕功能區
- 指令區
- 程式管理區
- 指令編輯區
- 元件注解區
- 元件索引區

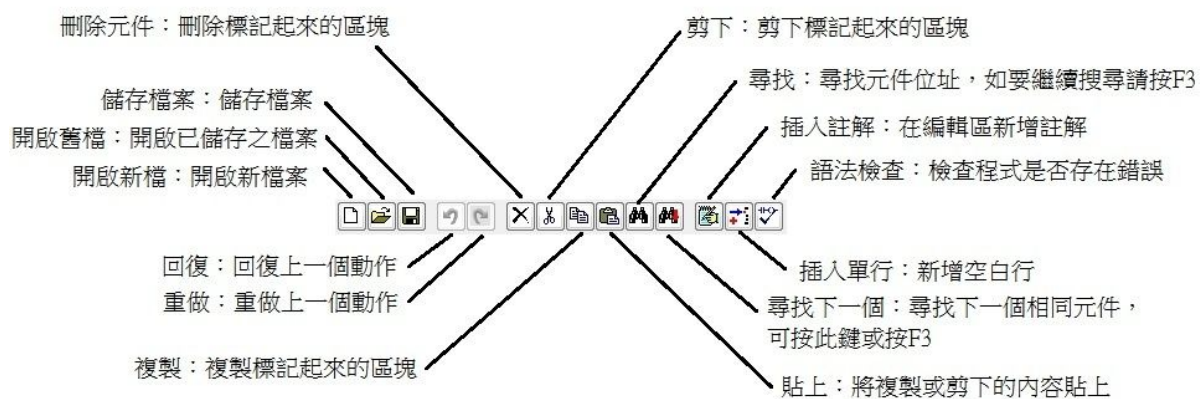


下拉式功能區



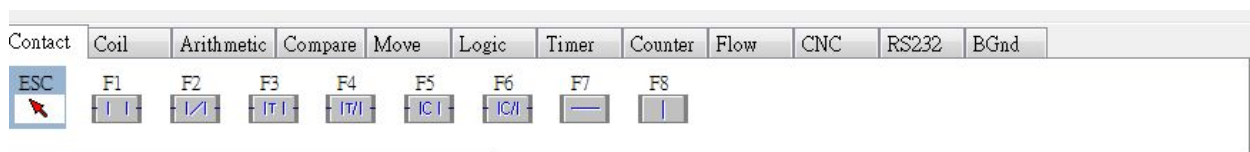


指令按鈕功能區



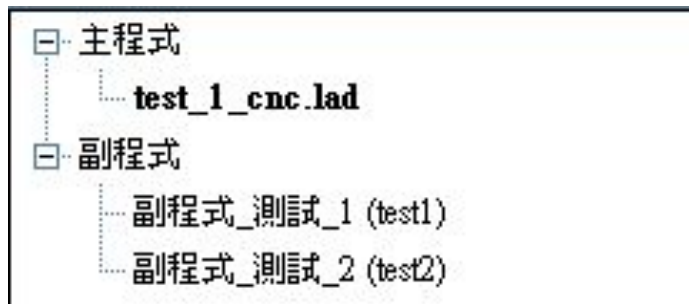
指令區

指令區提供各種元件讓使用者選取、應用，詳細說明請參閱後續章節。



程式管理區

顯示目前檔案中的主程式及副程式



指令編輯區

編輯階梯圖內容之區塊

參數輸入欄

選擇指令元件後，在指令編輯區之適當位置按滑鼠左鍵，會出現如下圖所示之元件參數設定視窗，在輸入適當數值後，按下確認按鈕，指令元件就會被寫入到指定位置。

元件參數設定	
來源暫存器	0000
目標暫存器	0000
確認 取消	

編輯注解

選擇指令元件後，在指令編輯區之元件上方按滑鼠右鍵會出現選單，此時選擇『編輯 XXXX 注解』（XXXX為元件名稱），會出現注解編輯視窗，此視窗提供注解編輯功能。

短注解至多8個字元，可顯示於指令編輯區的元件上方，長注解字數不受限，但無法出現於指令編輯區中。

SYNTEC

鍵盤操作說明

按鍵	功能說明
Page Up	編輯視窗時移動至游標位置前一頁
Page Down	編輯視窗時移動至游標位置後一頁
Home	將游標移至最左端
End	將游標移至最右端
Delete	刪除游標所在的元件
↑ ↓	游標在編輯區時，可移動游標上下
← →	游標在編輯區時，可移動游標左右

元件注解區

此區塊又稱元件列表區，為元件使用總表，使用者可於此確認元件使用情形，共有以下分類。

I、O、C、S、A、R、CT、TM、AX、JMP、JSR

SYNIEC

於元件注解區的元件按下滑鼠右鍵，支援下述功能：

- | TM | AX | JMP | JSR | | | |
|-----|---|-----|-------------------------------|---|---|----|
| I | O | C | S | A | R | CT |
| C0 | 短註解
Cycle Start | | 長註解
auto mode 下,此 bit O... | | | |
| C1 | <div> <div>編輯C1註解</div> <div>套用預設註解至選取項目</div> <div>全選</div> </div> | | | | | |
| C38 | | | | | | |
| | | | | | | |

使用步驟：

1. 點選指令編輯區的任一元件，此區塊會列出所選元件在階梯圖中有被寫入的位址，像是座標的概念，包含在主程式與副程式內的所有元件
2. 以滑鼠左鍵在特定位址點擊兩下，指令編輯區的游標會快速跳躍到該位址，若該位址在副程式內，指令編輯區也會切換到副程式

	Row	Col	Location
A100	5	10	test_1_cnc.lad
A100	11	1	副程式_測試_1(test1)
A100	12	1	副程式_測試_2(test2)

1.4 功能說明

檔案操作



Set File Language

如果階梯圖是用舊版PLC編輯器產生的，當使用新版PLC編輯器開啟時，介面語系有可能判斷錯誤（繁中、簡中），此時可以透過此功能設定開檔的語系。

2.26.3（含）之後的PLC編輯器產生的階梯圖不會出現此問題，不需特別去修改。

匯入/匯出DOS APP檔

此功能可匯入/匯出DOS系統之長短注解檔，簡體中文與繁體中文件名共用APPCHI.STR，英文件名為APPENG.STR。

匯入長短注解

此功能用來匯入新版注解檔（與DOS不同），執行此動作前需注意，若當下開啟之階梯圖已存在自行編輯之注解，需在匯入外部注解檔前，先行匯出注解檔。

若未先匯出，則已存在之注解內容將在匯入動作完成後被覆蓋。

匯入注解來源有兩種：

- 匯入標準注解

系統會先跳出確認視窗，請使用者確認是否要匯入標準注解，按下"是"即匯入標準注解。

匯入對象為與標準核心功能有關的C、S、R元件。

另外，在匯入標準注解後，每當新增與標準核心功能有關的C、S、R元件，會自動填入注解。



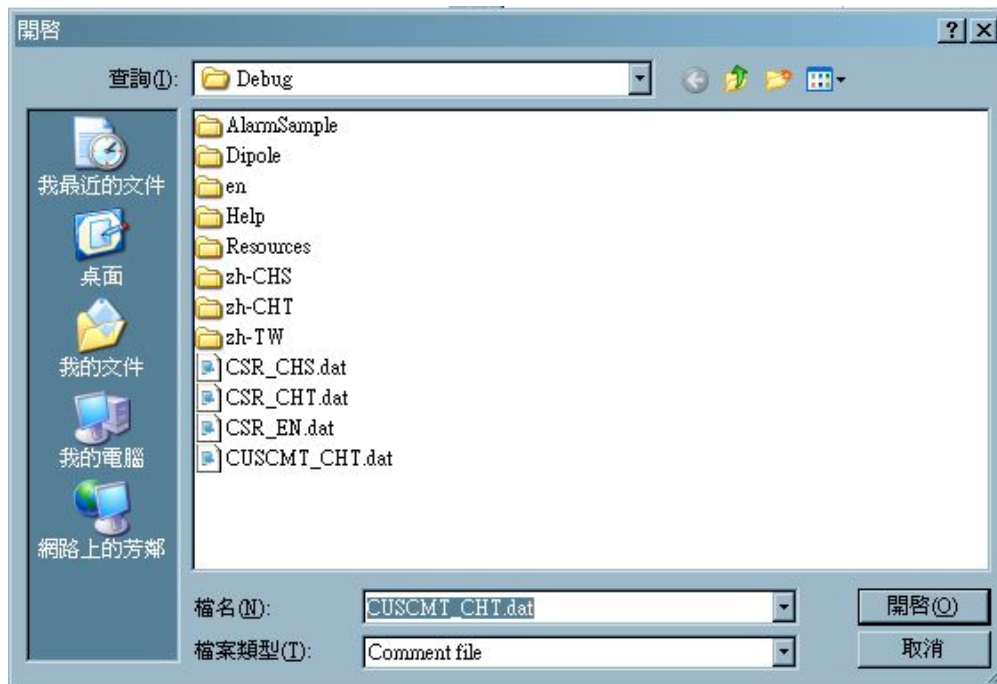
- 匯入客製注解

系統會先跳出確認視窗，請使用者確認是否要匯入客製注解。



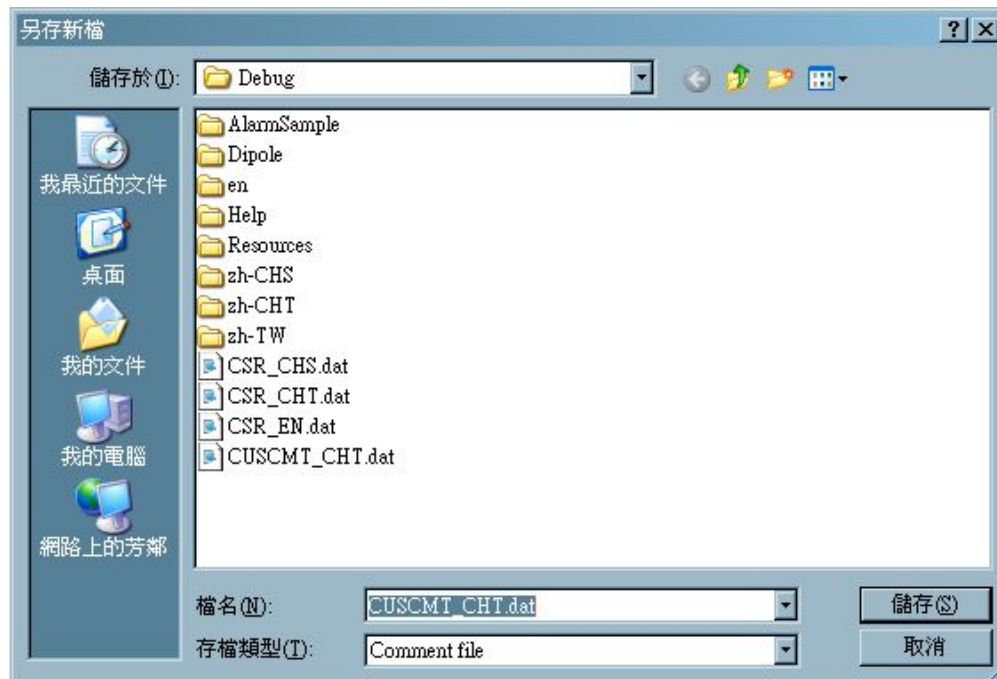
按下"是"會跳出系統對話視窗，可自行選擇要匯入之客製注解檔案。

SYNTEC



匯出長短注解

此功能用來匯出注解內容，以便同樣的注解內容可以轉移到其他階梯圖使用，內容包含長/短注解。執行時會跳出系統對話視窗，可自行設定匯出的位置及檔名。



若指定了相同的檔名，系統會跳出提示，提醒使用者。



匯出注解檔案時，會顯示檔案預計儲存之路徑，讓使用者確認。



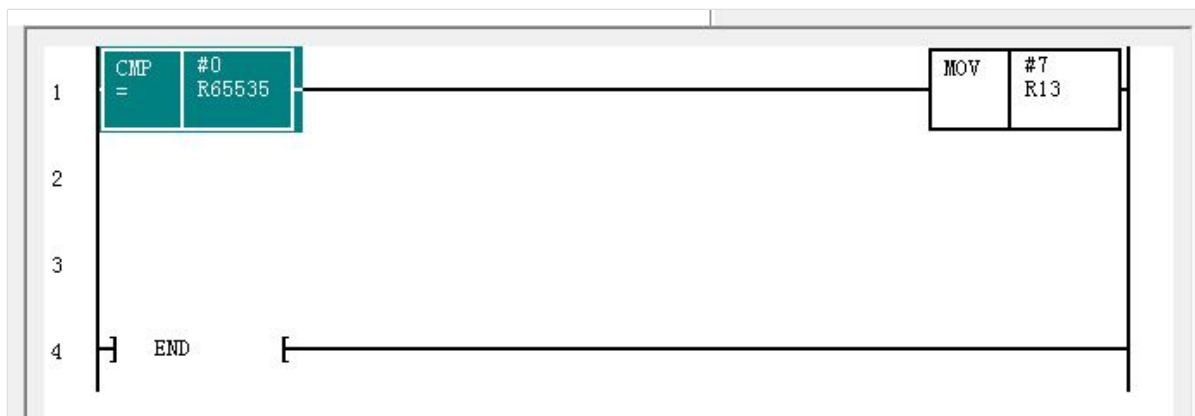
匯出注解檔之預設路徑為PLC編輯器執行之位置，也就是LadEditor資料夾。
預設之客製注解檔名為CUSCMT_*.dat，L為各語系之名稱（目前僅支援：繁中CHT、簡中CHS、英文EN）。
在繁中介面下匯出之注解檔案為CUSCMT_CHT.dat，在簡中介面下匯出之注解檔案為CUSCMT_CHS.dat。

編輯操作

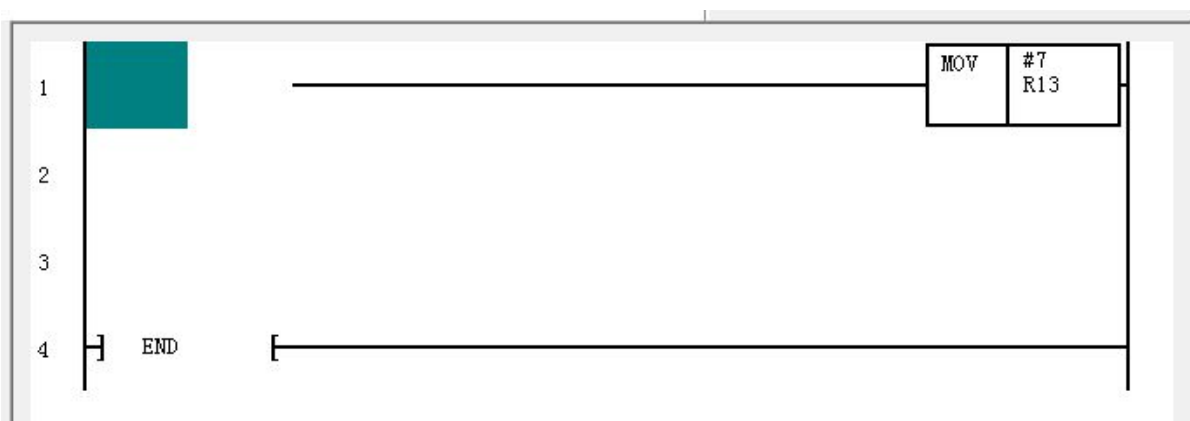


剪下元件

- 將游標移至元件上

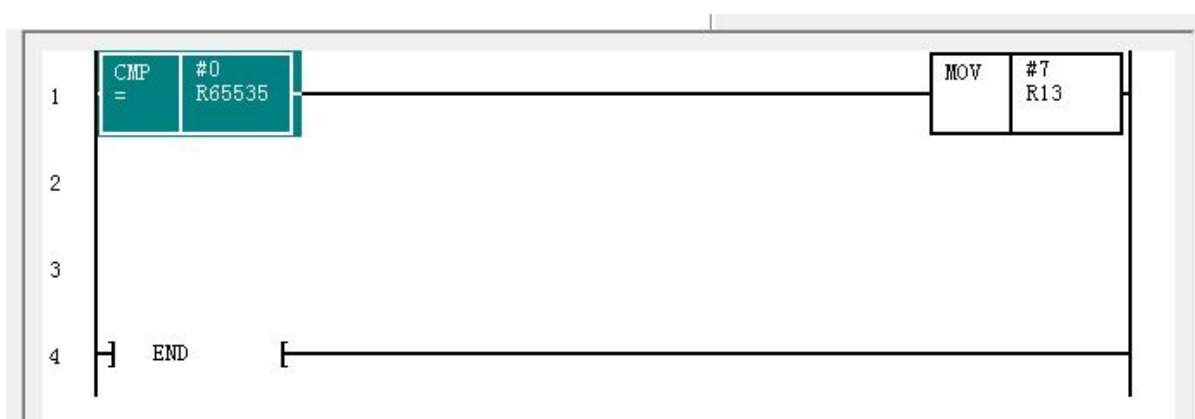


- 以滑鼠點選上方「編輯」里的「剪下」，或用鍵盤按下Ctrl+X，可將元件剪下至剪貼簿



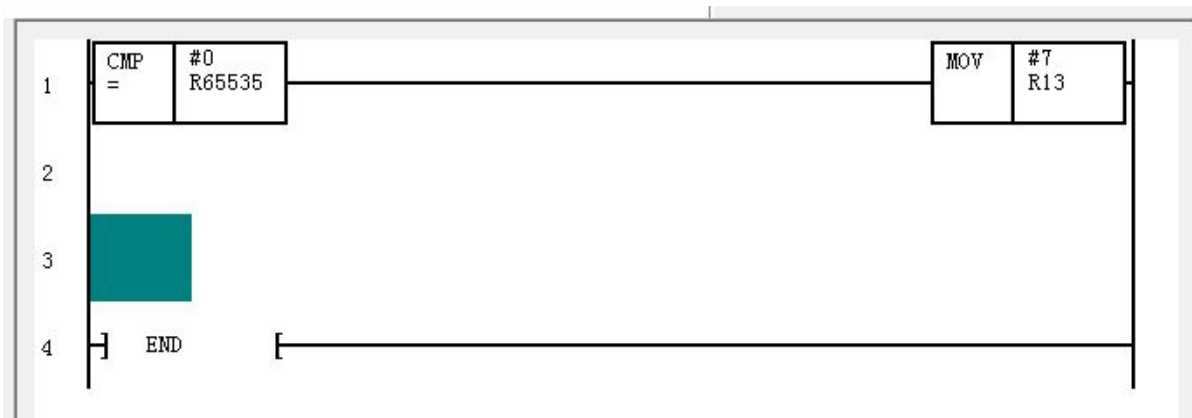
複製元件

- 將游標移至元件上，以滑鼠點選上方「編輯」里的「複製」，或用鍵盤按下Ctrl+C，可將元件複製到剪貼簿

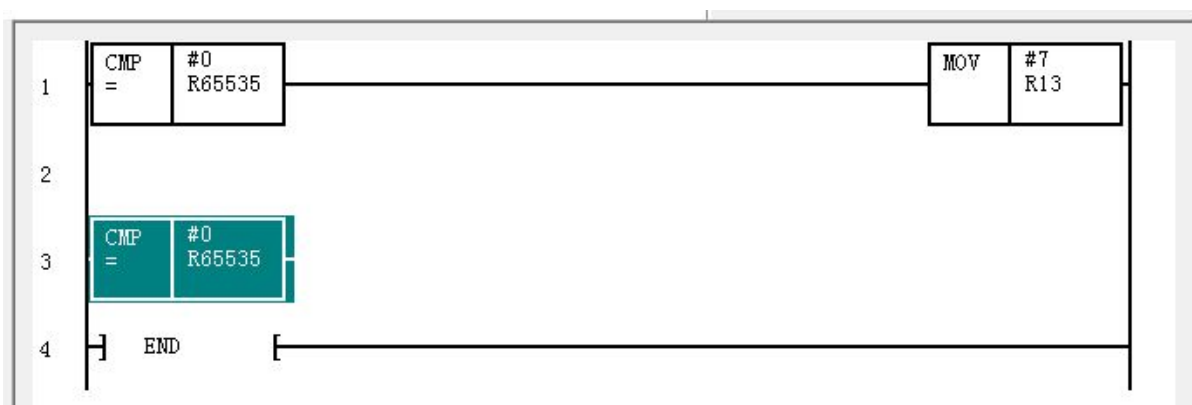


貼上元件

- 將游標移至某一位置

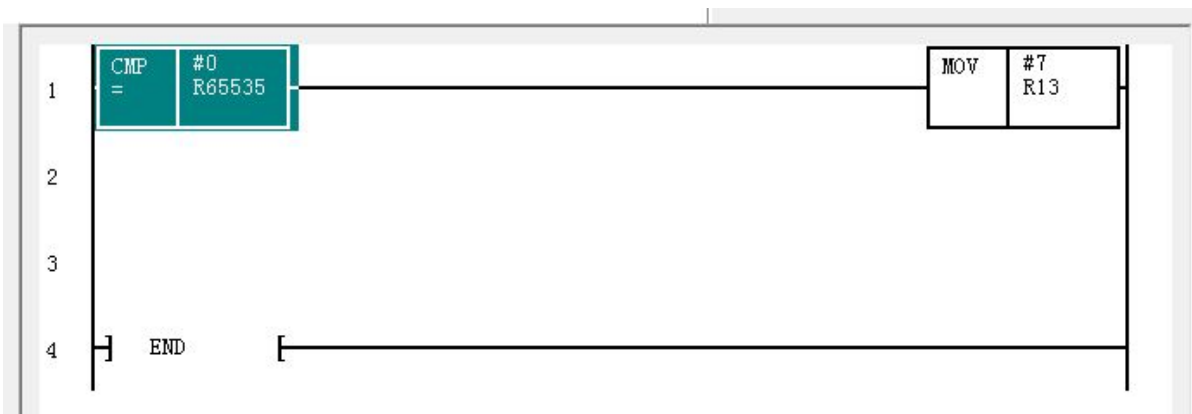


- 以滑鼠點選上方「編輯」里的「貼上」，或用鍵盤按下Ctrl+V，可將剪貼簿的元件貼至該位置

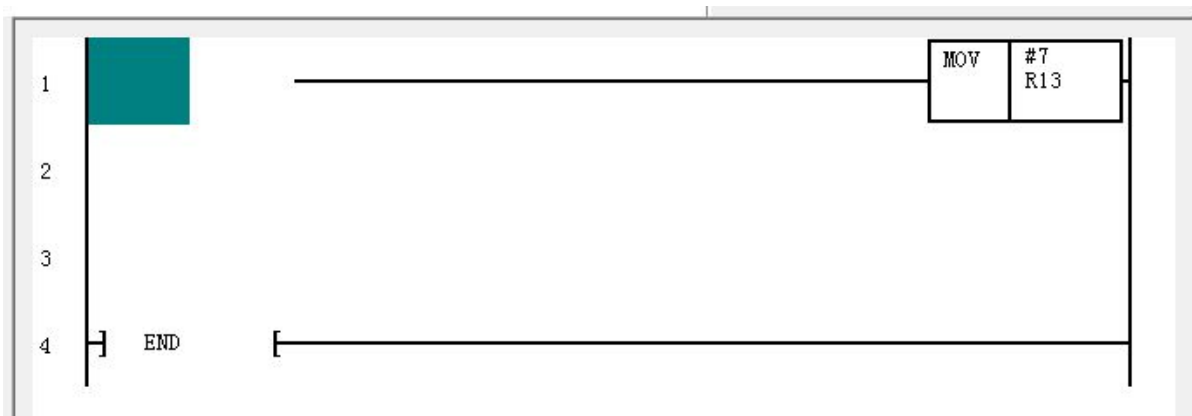


刪除元件

- 將游標移至元件上



- 以滑鼠點選上方「編輯」里的「刪除元件」，或用鍵盤按下Delete，可刪除元件



尋找 / 取代 元件

尋找元件

- 搜尋
 - 以滑鼠點選上方「編輯」里的「尋找」，或用鍵盤按下 Ctrl+F 可開啟搜尋視窗

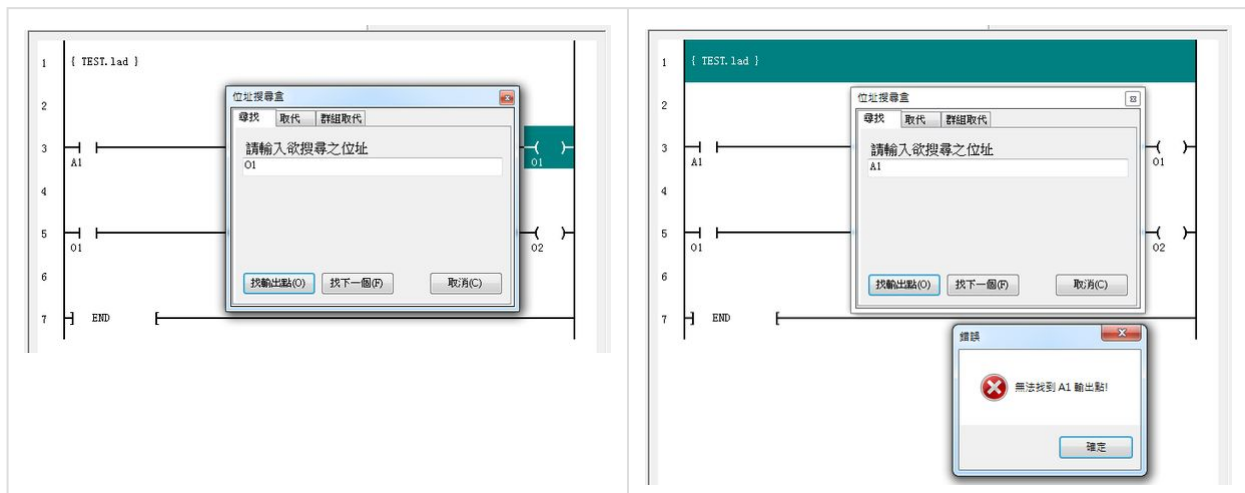


- 輸入欲搜尋元件之名稱
 - 若是當時游標上有元件存在，開啟尋找功能時，會自動載入該元件名稱，當做預設搜尋之名稱。
 - 若是當時游標上有元件存在，執行「尋找下一個」時（或以鍵盤按Ctrl + F3），會以該元件為目標執行「尋找下一個」動作。



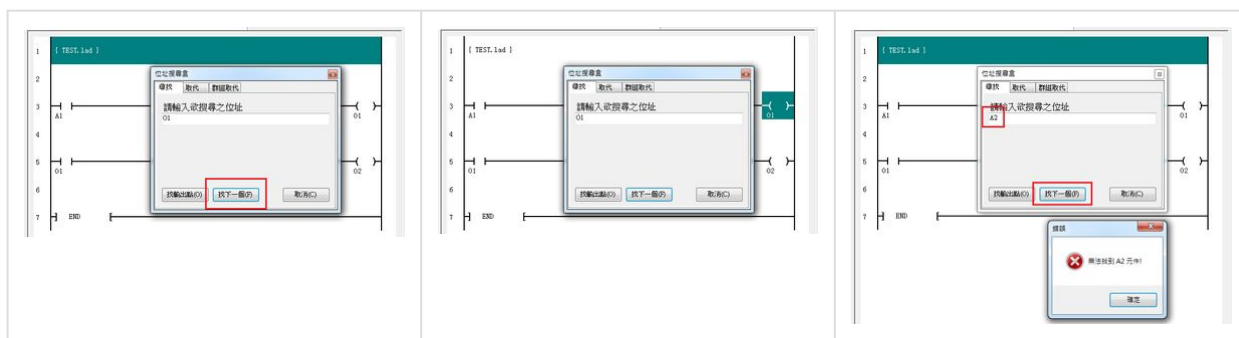
• 找輸出點

- 輸入欲搜尋之元件名稱，按下"找輸出點"，會搜尋符合該名稱，且為輸出點（Coil）的元件，並跳至該位置。
- 若沒有符合條件之元件，會跳出提示訊息，"無法找到 xxx 輸出點"，且游標不移動。
 - 例：A1 並非輸出點（Coil）



• 找下一個

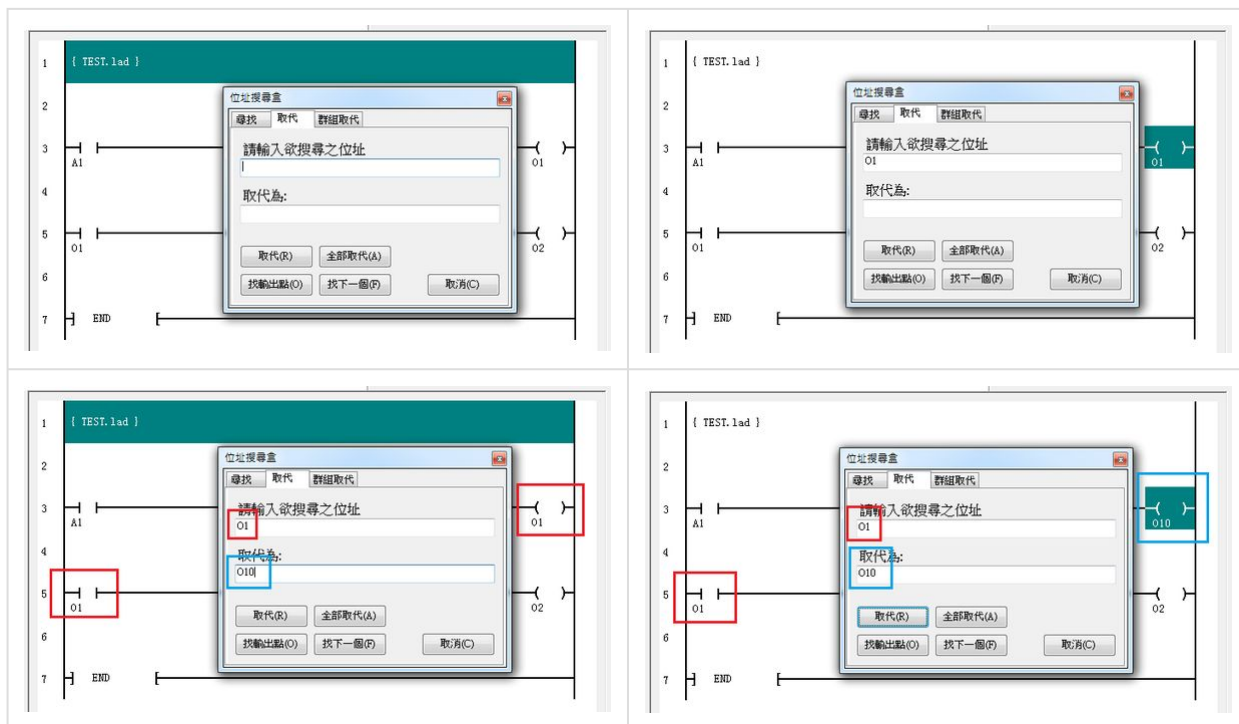
- 輸入欲搜尋之元件名稱，按下"找下一個"，會搜尋符合名稱，且位於當下游標位置之後的下一個元件，並跳至該位置。
- 若沒有符合名稱之元件，會跳出提示訊息，"無法找到 xxx 元件"，且游標不移動。
 - 例：A2 不存在



取代元件

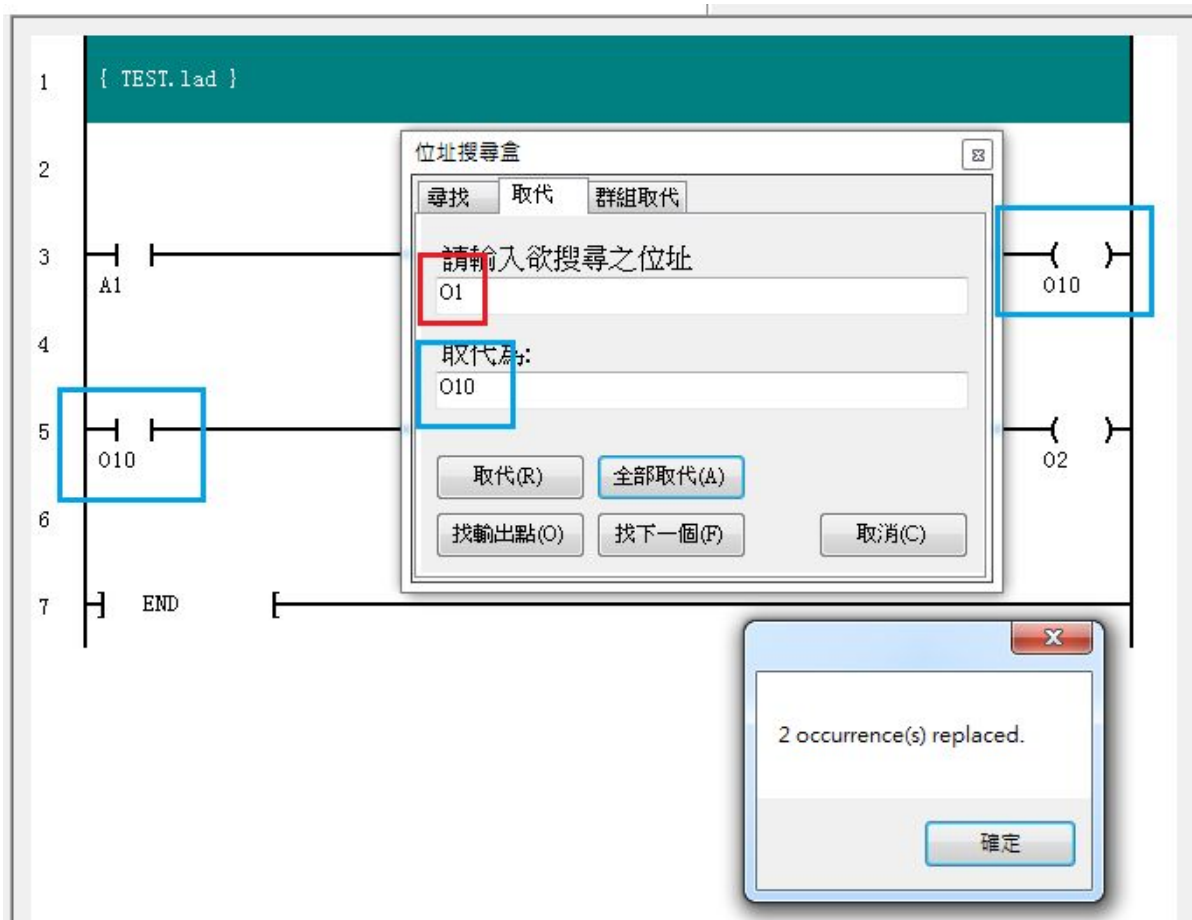
• 取代

- 以滑鼠點選上方「編輯」里的「取代」，或用鍵盤按下 Ctrl+R 可開啟取代視窗。
- 若是當時游標上有元件存在，執行取代時，會自動載入該元件名稱，當做預設搜尋之名稱。
- 輸入欲取代之元件名稱，及取代後的新元件名稱，按下取代，會搜尋符合該名稱的下一個元件，替換成新的名稱，取代完畢後，游標移動到該位置。



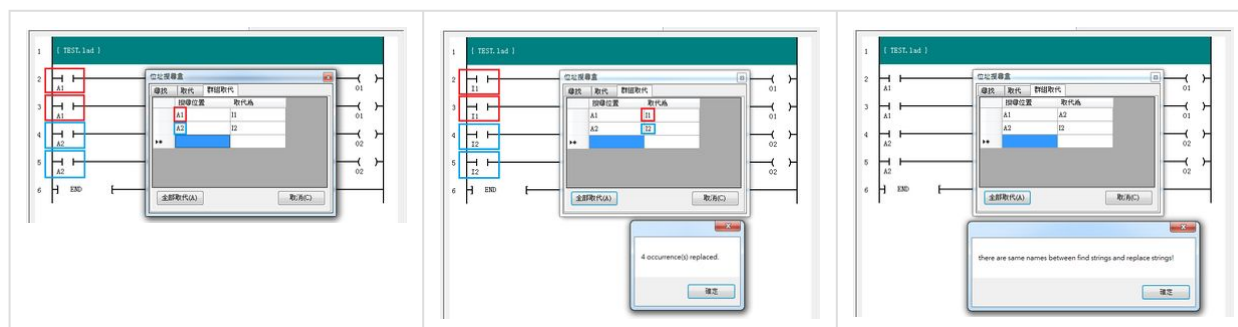
• 全部取代

- 輸入欲取代之元件名稱，及取代後的新元件名稱，按下全部取代，會搜尋所有符合該名稱的元件，替換成新的名稱，取代完畢後，游標不動。



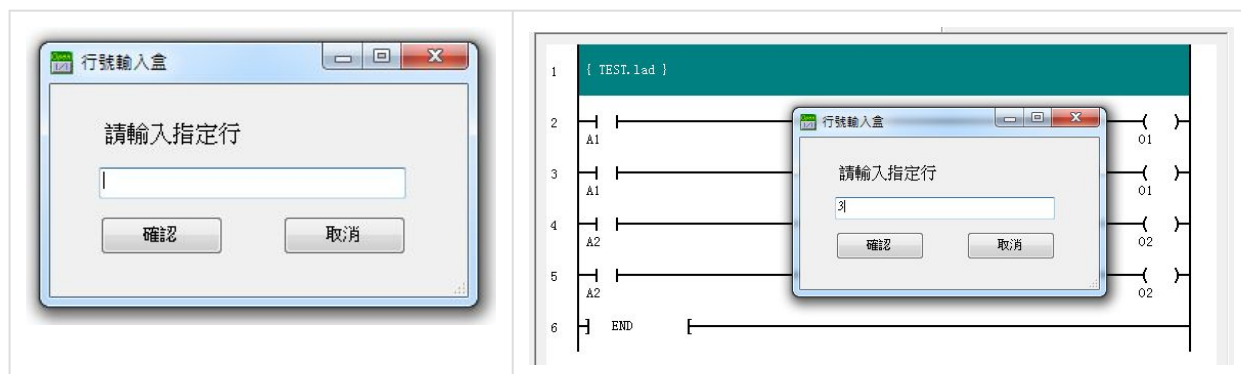
群組取代

- 以滑鼠點選上方「編輯」里的「群組取代」，可開啟群組取代視窗，可輸入多組的搜尋字串與取代字串，執行全部取代。
- 搜尋的字串與取代的字串中不可有相同的內容。



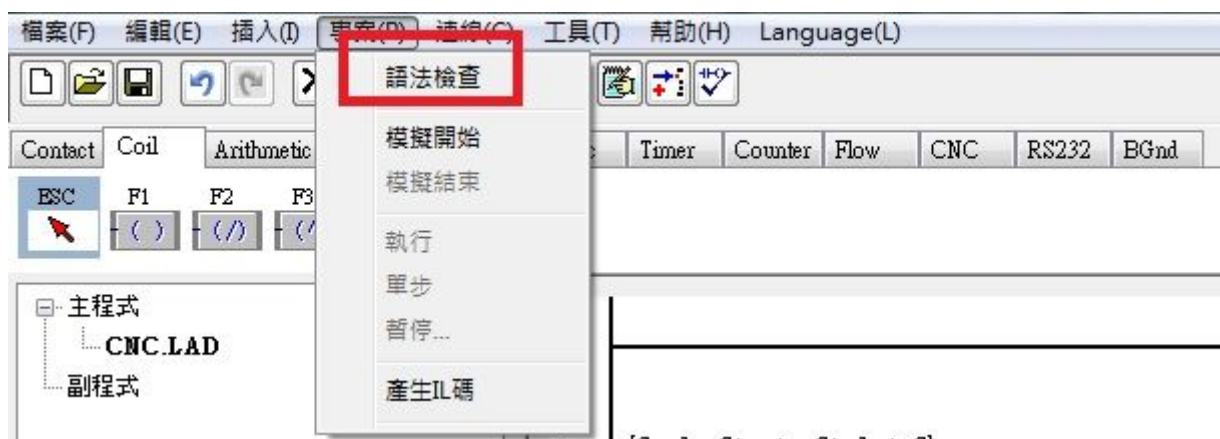
跳行功能

- 以滑鼠點選上方「編輯」里的「到指定行」，或用鍵盤按下 Ctrl+G 可開啟跳行視窗
- 輸入指定行號，按下確認，即可跳至該行

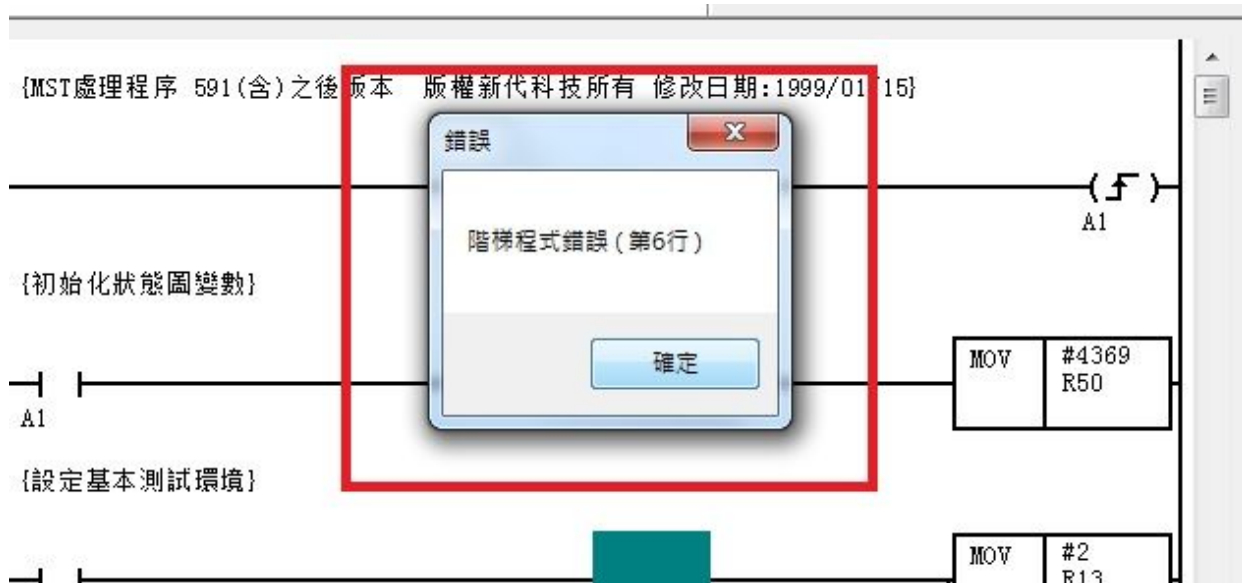


語法檢查

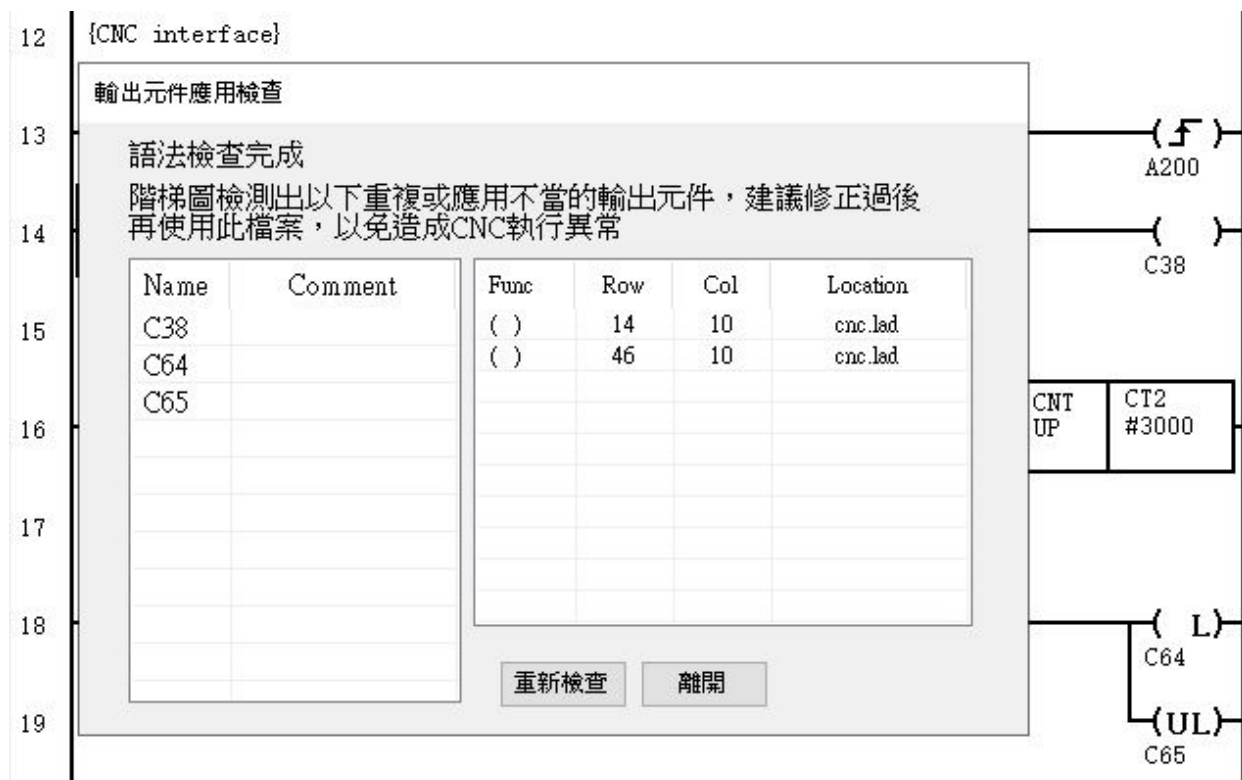
- 此功能可針對指令編輯區內的階梯圖內容做語法的檢查：
 - 若沒有檢查到錯誤，游標會留在原處。
 - 若有檢查到錯誤，會提示使用者發生錯誤的行號，並將游標移至該行。



SYNTEC



- 如果檢查到重複輸出或應用不當的狀況，會出現元件列表的視窗，列出所有重複輸出及應用不當的元件名稱、元件類型、位址與所屬之主副程式名稱。
 - 應用不當的定義 (支援版本: 2.35.2)
 - 保持線圈 (Latch Coil) 及復歸線圈 (Unlatch Coil) 未成對出現



- 如果看到以下訊息，表示階梯圖編譯的結果超出PLC編輯器之限制，請嘗試將JSR元件往下方移動。



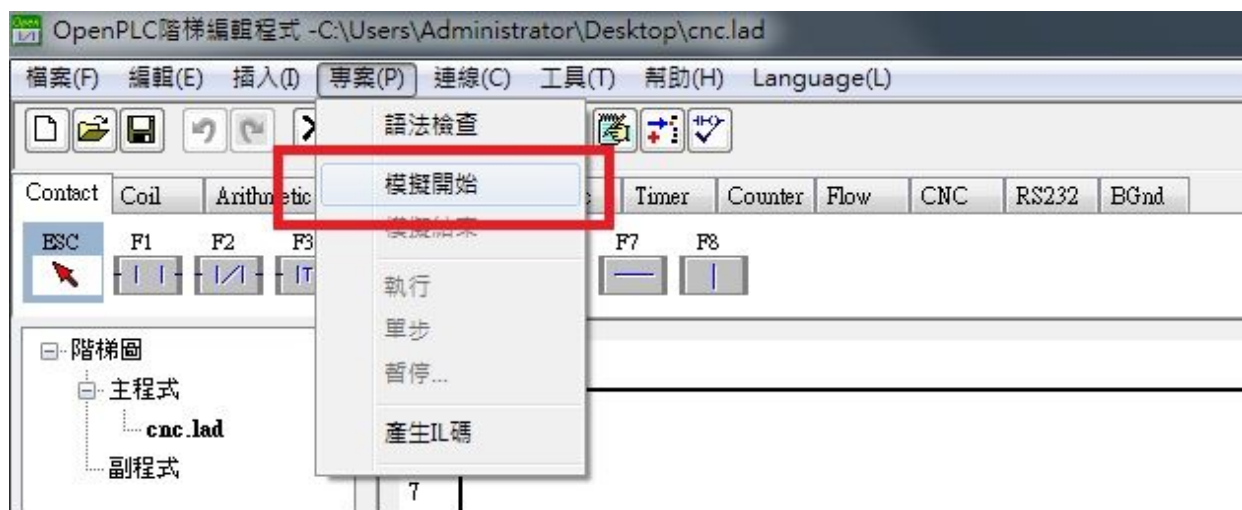
模擬功能

此功能可在PLC編輯器內模擬當下階梯圖內容執行的結果，指令編輯區及元件列表區會同步顯示目前的接點狀態。

在模擬期間，支援以下動作：

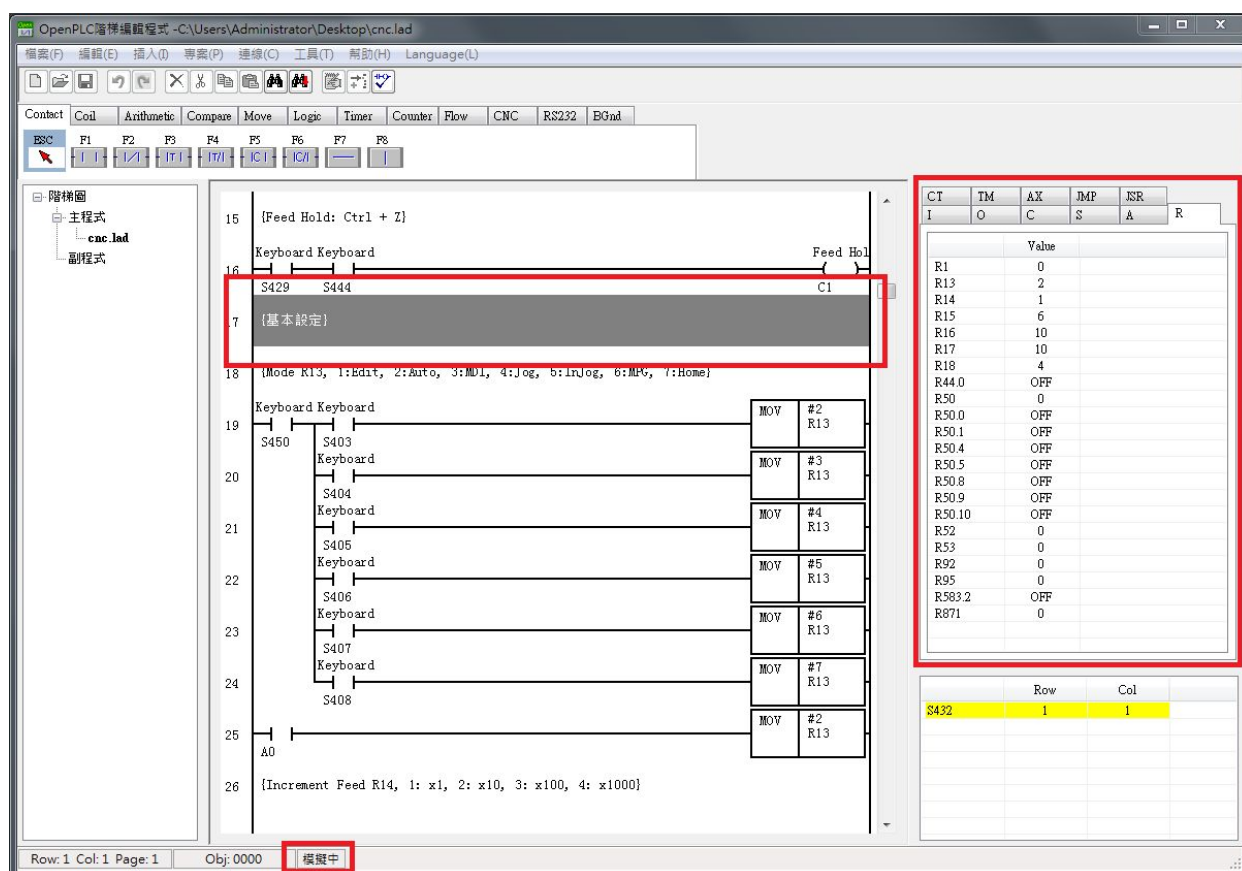
- 即時觸發I、A、R元件
- 即時更新PC鍵盤相關的S bit狀態
- 修改R bit數值

SYNTEC

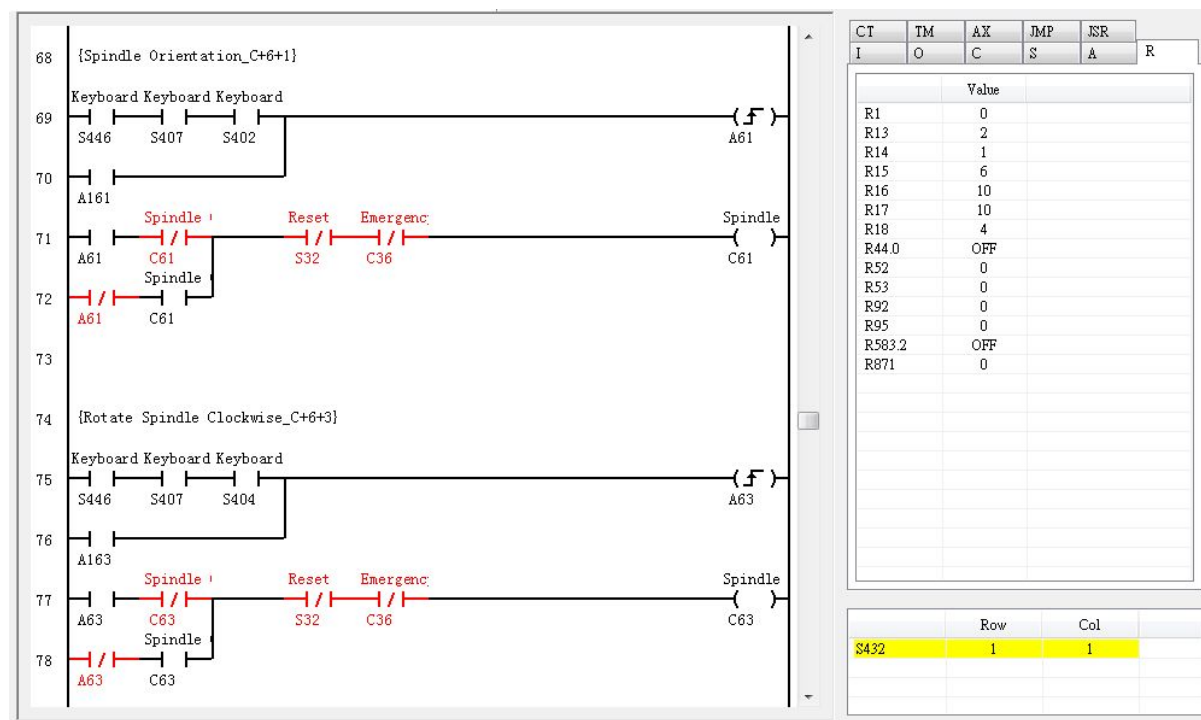


模擬介面

模擬開始後，下方狀態列會顯示"模擬中"，指令編輯區的游標顏色會由綠色變為灰色。同時元件列表區會取消注解編輯功能，改成顯示各類接點目前的狀態，包含On/Off及數值。



當元件導通時，指令編輯區的元件會變紅色



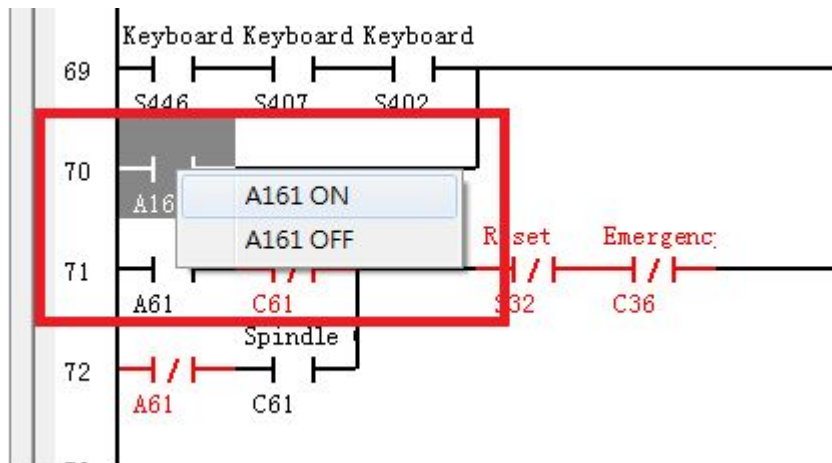
Timer 與 Counter 元件提供即時數值更新的功能

Timer元件	模擬中動態更新數值	元件導通狀態

觸發I-bit、A-bit、R-bit

模擬中，可在欲觸發的I-bit、A-bit、R-bit 元件上按滑鼠右鍵，會彈出接點開關選單。

SYNTEC

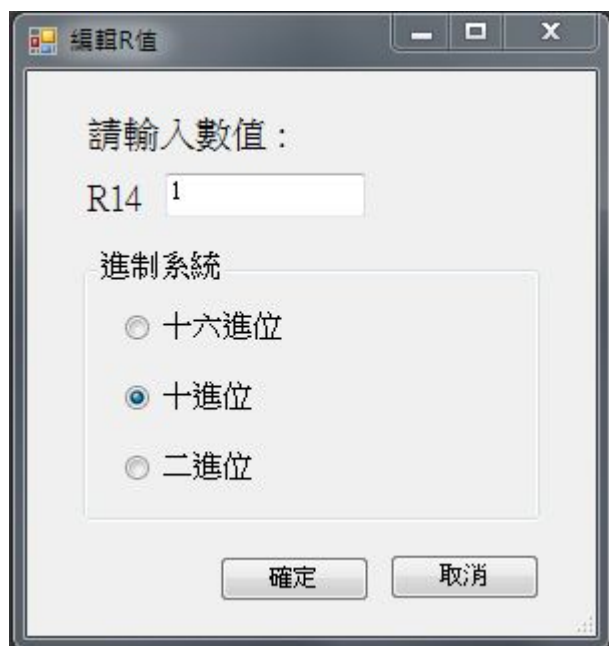


修改R值

模擬中亦可在元件列表區上，以滑鼠右鍵點選欲修改數值的R bit。

CT	TM	AX	JMP	JSR	
I	O	C	S	A	R
		Value			
R1		0			
R13		2			
R14					
R15					
R16		10			
R17		10			
R18		4			
R44.0		OFF			
R52		0			
R53		0			
R92		0			
R95		0			
R583.2		OFF			
R871		0			

R bit的數值修改有16進位、10進位與2進位共三種輸入方式。



元件監看列表

模擬開始之後，會同時跳出元件監看列表，可在輸入框內設定想觀察的接點，下方的列表在模擬期間會即時更新。

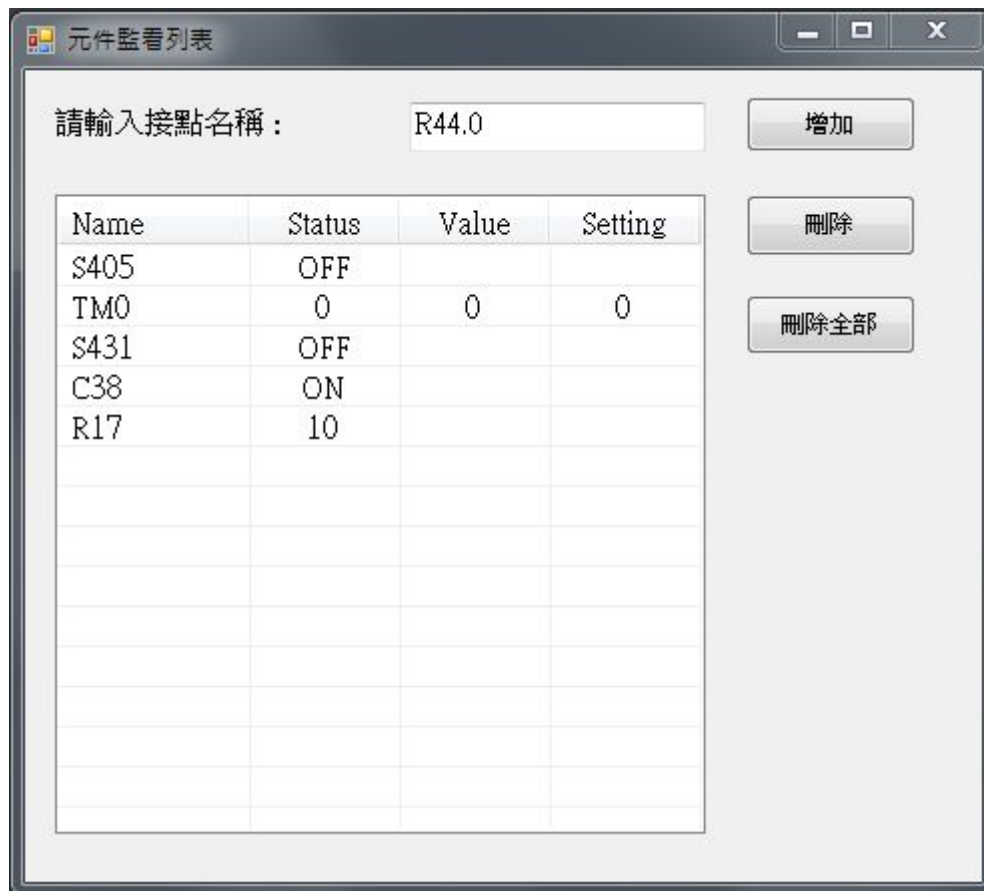
增加元件方式：在輸入欄內輸入接點名稱，點選增加，或是按下Enter鍵。

刪除元件方式：滑鼠選擇元件，點選刪除。也可一次刪除全部監看清單。

提供監看的元件類型：

元件類型	I-Bit	O-Bit	C-Bit	S-Bit	A-Bit	Register	Timer	Counter
------	-------	-------	-------	-------	-------	----------	-------	---------

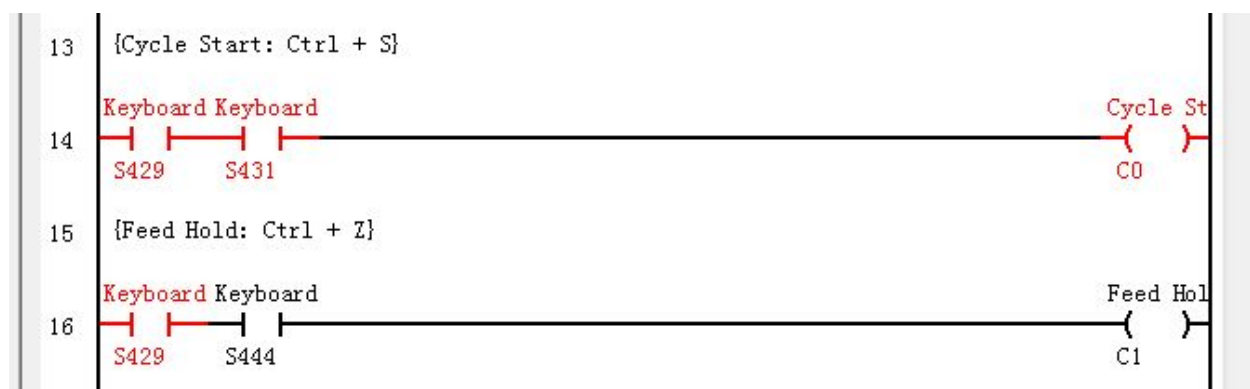
SYNTEC



模擬鍵盤觸發S bit

模擬期間可透過PC鍵盤觸發對應S bit。(注2)

下圖範例為按下Ctrl 與 S鍵。



注意事項

- 模擬前會先做語法檢查，如果有語法錯誤，則無法進行模擬。
- 模擬前會判斷內容是否修改過，並詢問是否存檔。
- 模擬過程中，不開放編輯元件與任何檔案操作。
- 監看列表中的Timer 與 Counter 元件，其秒數或次數必須在元件被觸發之後才會同步更新數值。

注1: 支援模擬功能的元件

元件類型	項目			
接點Contact	- -	- T -	- C -	
	- / -	- T/ -	- C/ -	
線圈Coil	-()-	-(/)-		
計時器Timer	tim1ms TIM, #	tim10ms TIM, #	tim100ms TIM, #	tim1s TIM, #
	tim1ms TIM, R	tim10ms TIM, R	tim100ms TIM, R	tim1s TIM, R
計數器Counter	cntUp CT, #	cntDn CT, #	cntRingUp #	cntRingDn #
	cntUp R	cntDn R	cntRingUp R	cntRingDn R
比較元件	CMP > R, R	CMP < R, R	CMP = R, R	
	CMP > #, R	CMP < #, R	CMP = #, R	

注2: PC鍵盤對應S bit



SYNTEC

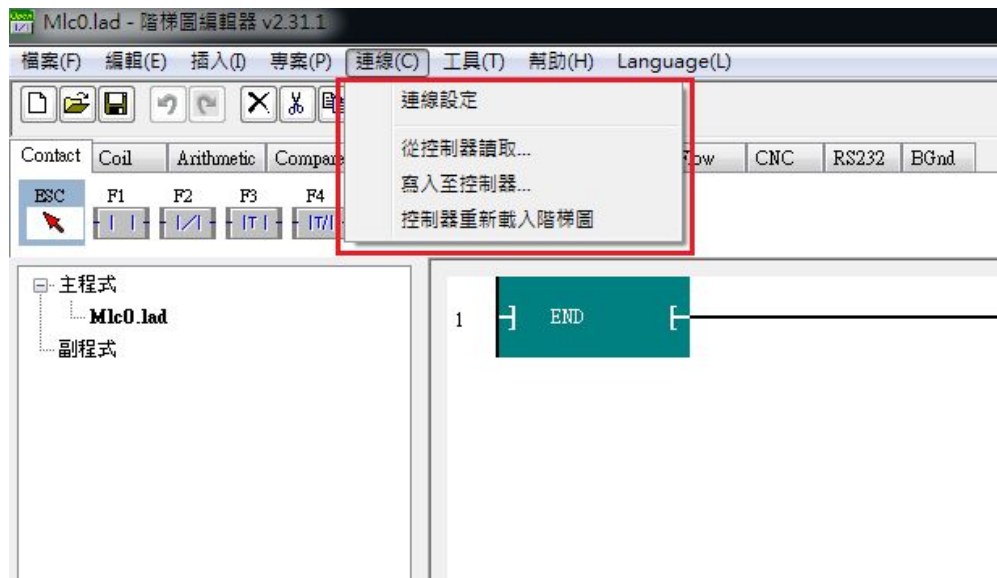
基本按鍵區

Esc 401	F1 458	F2 460	F3 461	F4 462	F5 463	F6 464	F7 465	F8 466	F9 467	F10 468	F11 469	F12 470		
` 441	1 402	2 403	3 404	4 405	5 406	6 407	7 408	8 409	9 410	0 411	- 412	= 413	\ 443	Back 414
Tab 415	Q 416	W 417	E 418	R 419	T 420	Y 421	U 422	I 423	O 424	P 425	[426	] 427	Enter 428	
Caps 458	A 430	S 431	D 432	F 433	G 434	H 435	J 436	K 437	L 438	; 439	' 440			
Shift 444	Z 444	X 445	C 446	V 447	B 448	N 449	M 450	, 451	. 452	/ 453	Shift 454			
Ctrl 429		Alt 456		Space 457								Alt 456		Ctrl 429

擴充按鍵區

Print 470	Scroll 470	Pause		Num 469	/ 453	* 455	- 474
Insert 482	Home 471	Up 473		7 Home 471	8 ↑ 472	9 Up 473	+ 478
Delete 483	End 479	Down 481		4 ← 475	5 476	6 → 477	
	↑ 472			1 End 479	2 ↓ 480	3 Down 481	Enter 428
← 475	↓ 480	→ 477		0 482		. 483	

連線操作



1. PLC編輯器與後台控制器軟體版本配對表：

PLC編輯器版本	控制器匹配軟體版本
2.29.4以後	10.116.36B、10.116.37和10.117.37以後
2.29.3以前	10.116.36A、10.116.36和10.117.36以前

2. 使用連線功能前，須先確認控制器與PC已連線成功，可參考網路連線設定說明文件設定。
3. 完成上述連線設定後，請至控制器端開啟核心伺服器。
4. 在後台伺服器的設定上，建議逾時時間設成0，可以避免部分網路速度不穩定的連線問題。

SYNTEC

G54	TEST N0 L1	網路設定	2015/7/31	14:50:54	管理者
-----	------------	------	-----------	----------	-----

核心伺服器設定參數

開機是否啟動伺服器 開

逾時時間(毫秒) 0

0=Disable Timeout
● 就緒
自動執行
警報

啟動伺服器
確認
取消
Dipole Log

PC連線設定

與後台（控制器）連線的PC端相關設定

連線設定

設定

遠端核心IP :

錯誤重試次數 :

錯誤重試等待時間 :

逾時時間 :

提示：

連線測試
儲存

設定選項	建議值
遠端核心IP	控制器端IP位址
錯誤重試次數	5
錯誤重試等待時間	3000ms
逾時時間	5000ms

設定步驟：

1. 設定上述的四項連線資訊
2. 按下連線測試
3. 確定連線成功（可參考左下方的提示字串）
4. 按下儲存



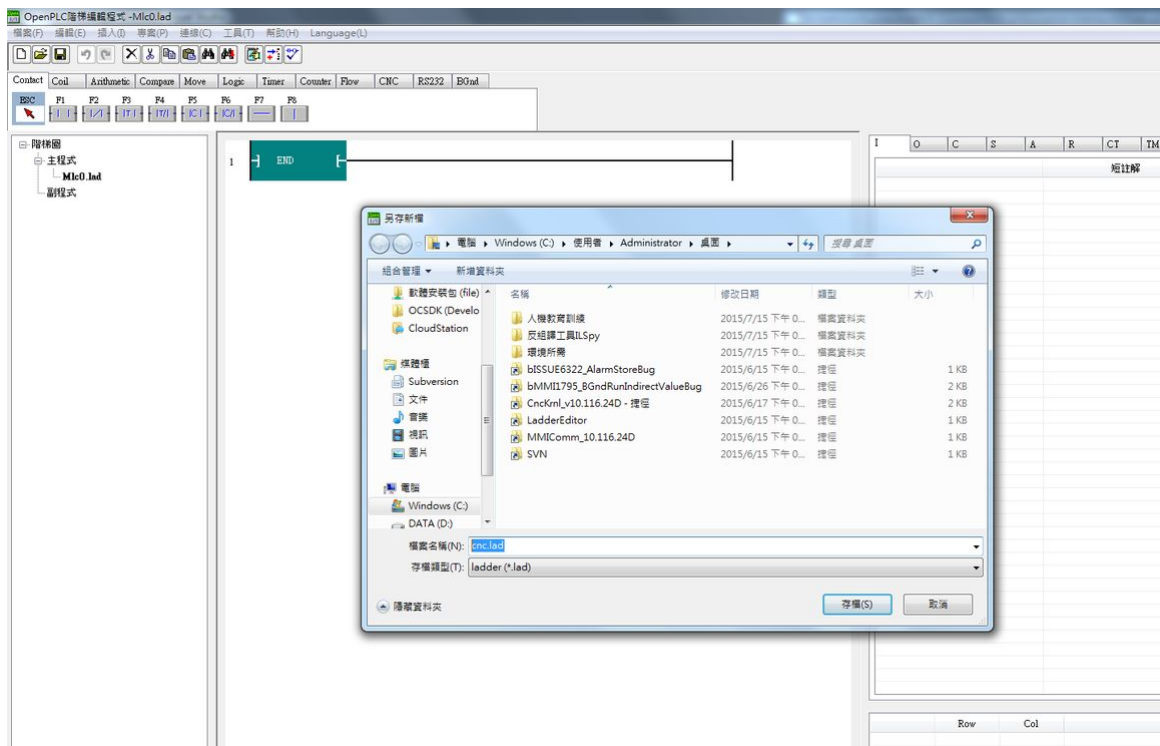
從控制器讀取

從PC端下載控制器上的階梯圖

操作步驟：

1. 選擇階梯圖存放的位置與檔名

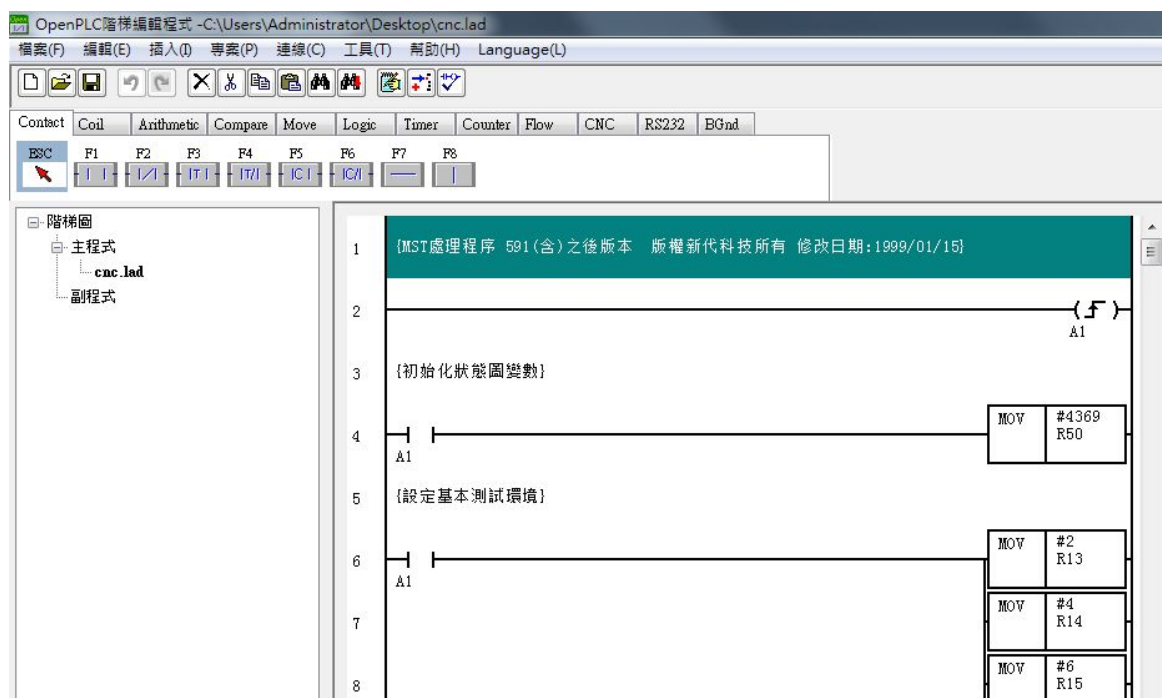
SYNTEC



2. 確認下載結果，下載成功後會有提示。



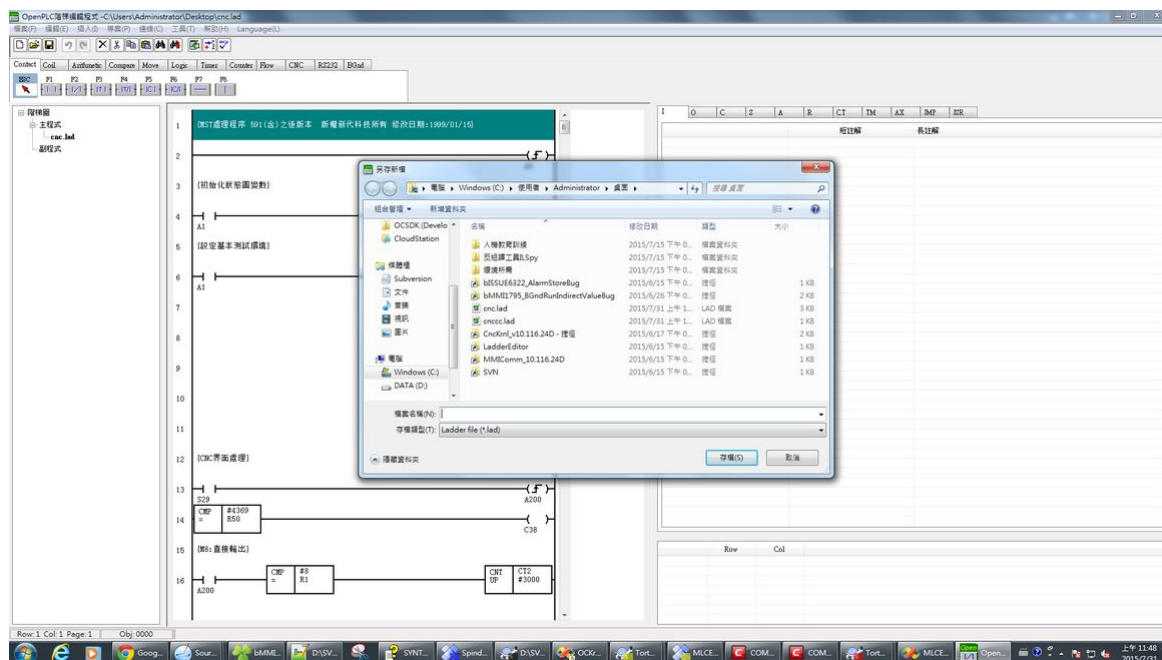
3. 下載結束之後，PLC編輯器會自動載入該階梯圖。



寫入至控制器

把目前開放的階梯圖上傳到控制器

1. 寫入前會先儲存檔案，如果是新開的檔案，會請使用者選擇存檔路徑。
2. 上傳前建議先做語法檢查，避免寫入至控制器後出現錯誤，嚴重可能導致無法開機。



3. 確認上傳結果，上傳成功後會有提示。



4. 寫入成功後，控制器端需斷電重開，才會讀取新的階梯圖。

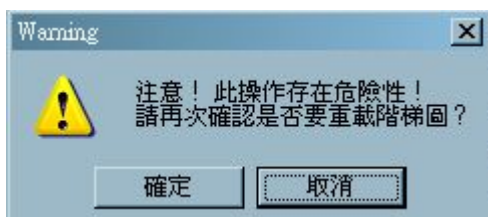
控制器重新載入階梯圖

PLC編輯器支援版本：2.31.0

控制器支援版本：10.116.45（含）以後

此功能可讓機台在不斷電的狀況下，直接重新載入階梯圖，但此操作必須在機台處於未就緒狀態下才能進行

1. 跳出彈窗詢問是否要重新讀取

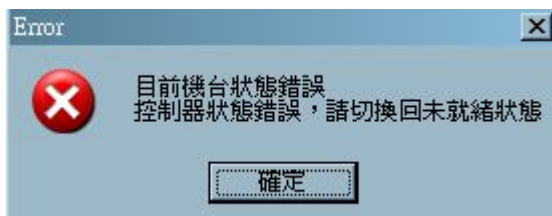


2. 回報成功與否

成功：



失敗：





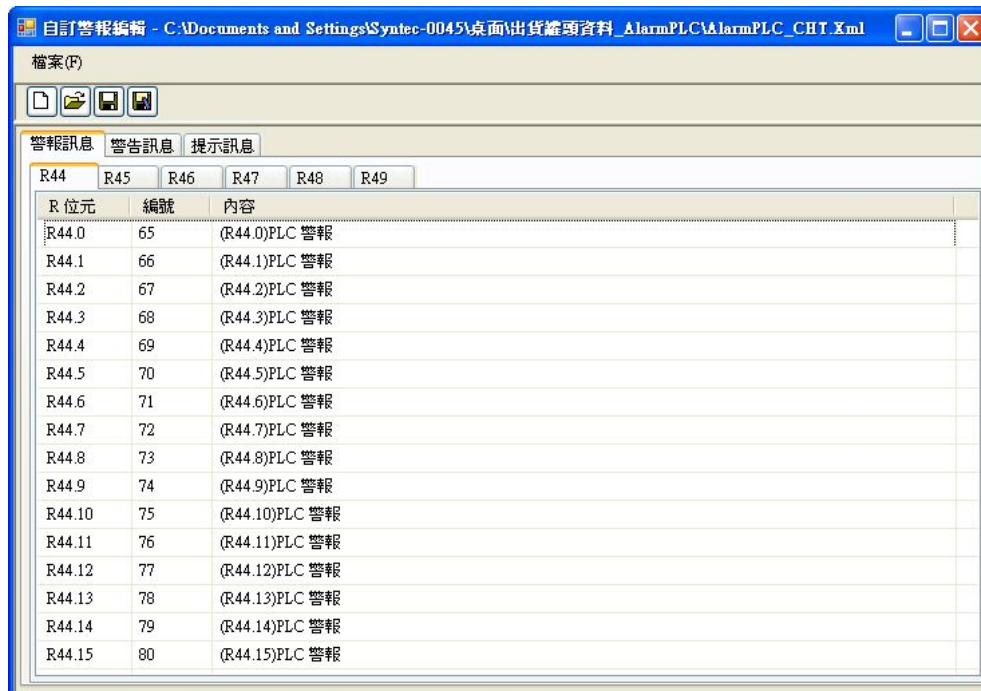
自訂警報編輯

PLC編輯器支援版本：2.20.0

開啟PLC編輯器後，點選下拉式功能區："工具"→"自訂警報編輯"，開啟自訂警報編輯介面，如下圖。

目前可編輯之警報區間為R44~R49，各警報的對應字串編號可從編輯介面上查詢。

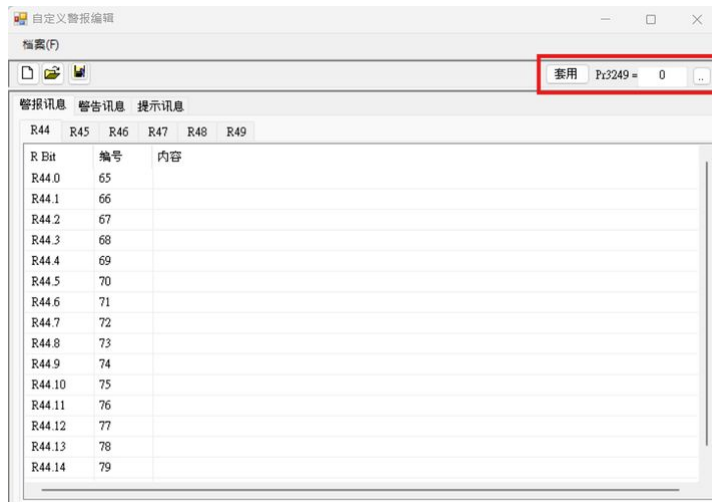
亦可更改參數Pr3249擴充使用者自行定義的PLC提示訊息區R590~R599為警報。若是由提示轉換為警報，則提示訊息編號1~319分別對應到警報編號501~819。



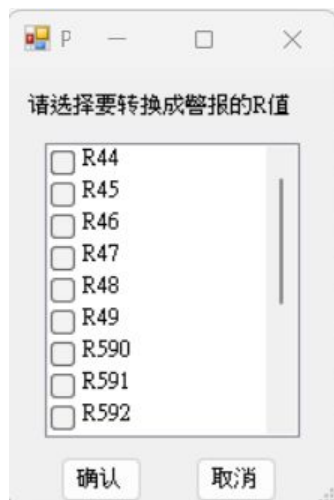
設定介面支援輸入Pr3249參數：

支援版本：from 2.35.4~

使用介面說明：



- 使用者可以手動輸入目標控制器的Pr3249參數，按下套用後可以將UI根據設定值移至正確的頁籤
 - 輸入值範圍：0~65535
 - 輸入"0"即為忽略Pr3249設定
- 情境舉例：Pr3249輸入"65535"並套用 (所有PLC提示、警告均轉換為警報)，此時UI介面將轉變成以下模樣
 - 警報訊息頁籤含有R44~R49 的0~31bits、R590~R599，而警告跟提示頁籤均為空
 - 使用者也可以輸入框右側的"..."按鈕來輸入Pr3249的數值，按下後會出現以下彈窗



- 勾選欲轉換成警報的R值後將把設定轉換為Pr3249數值回填至輸入框，需手動按下套用才會生效
- :error: 其他注意事項
 - 使用者"主動觸發"讀取檔案時(ex: 開啟舊檔)，若當前Pr3249的數值跟檔案內容沒有匹配，將"自動修正"Pr3249的數值，讓文件內容能完整呈現
 - 當讀取的檔案內容有衝突時(ex: 同個R值隨機存取記憶體在MLC跟MLCHint)，此時將有彈窗提示衝突，並在介面上以紅字提示，使用者須自行開啟原檔排查
 - 發生該衝突情境時UI將以顯示MLCHint為優先
 - 錯誤情境下若存檔可能造成檔案內容遺失，請小心謹慎

自訂警報編輯在存檔時必須先選擇語系，而後PLC編輯器會自動生成對應的檔名。

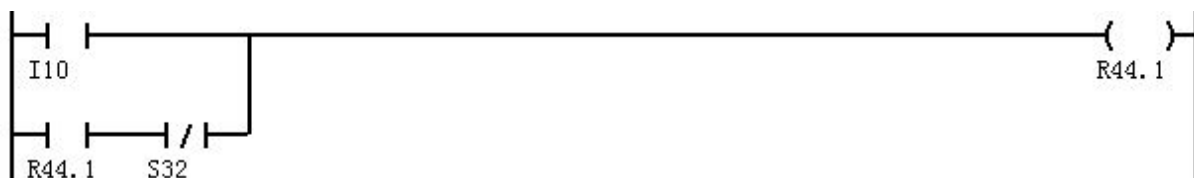


補充事項：

1. 開啟DOS的str檔，格式可以存成xml跟str，開啟CE的xml檔，格式只能存成xml。
2. 開啟CE舊檔時，檔名只能為下列其一：AlarmPLC_CHT.xml / AlarmPLC_CHS.xml / AlarmPLC_Com.xml。

自訂警報觸發方式

如下圖所示，外部訊號I-10觸發R44.1，而R44.1對應PLC_Alarm編號66。



語系切換 Language

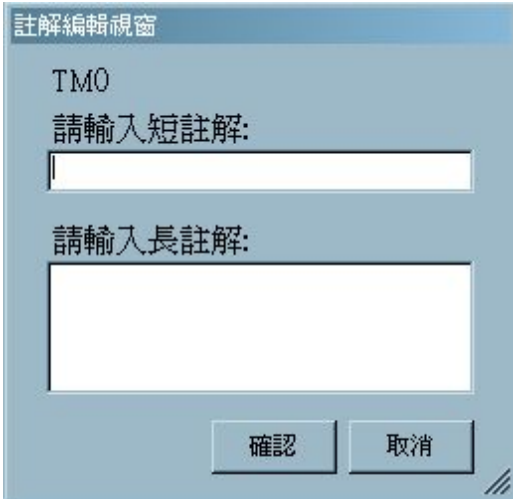
此功能可變更PLC編輯器操作介面的顯示語系，點選下拉式功能表之Language，選擇語系後，重新開啟PLC編輯器即切換至另一語系。

目前提供：繁體中文、簡體中文、英文。

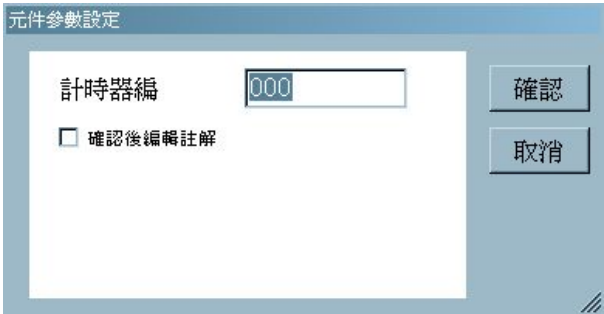
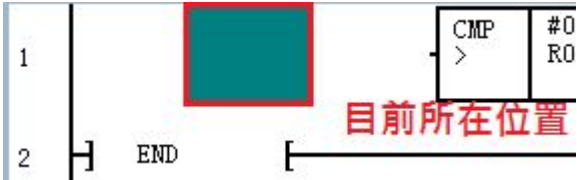
SYNTEC

快捷鍵功能說明

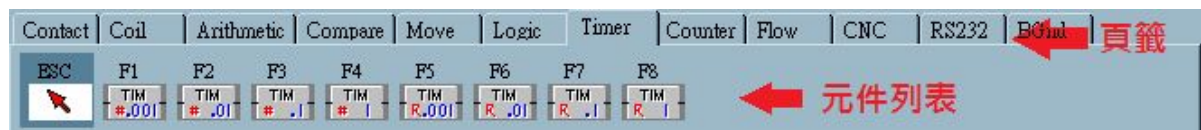
介面操作快捷鍵

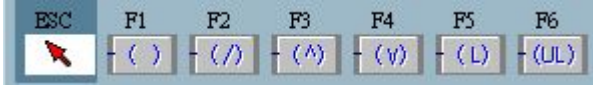
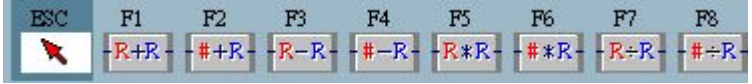
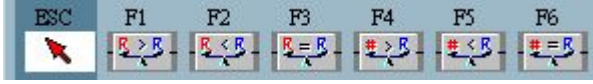
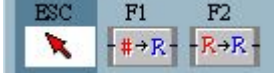
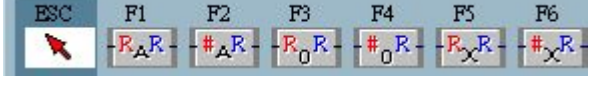
功能說明	快捷鍵
插入注解 注解行會加在游標所在位置的上方  The diagram shows a ladder logic step. Line 1 contains a comment box with the text 'This is comment'. Line 2 contains a normally open contact followed by the instruction 'END'. A cursor is positioned at the end of line 2, indicating where a new comment would be inserted above it.	Ctrl + M
注解編輯視窗 游標需在欲編輯之元件上  The screenshot shows a dialog box titled '註解編輯視窗' (Comment Edit Window). It contains the text 'TMO' and two input fields: '請輸入短註解:' (Please enter short comment:) and '請輸入長註解:' (Please enter long comment:). At the bottom, there are two buttons: '確認' (Confirm) and '取消' (Cancel).	Ctrl + Enter

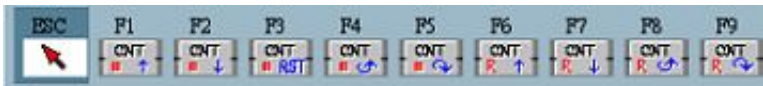
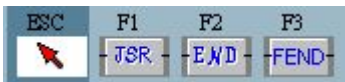
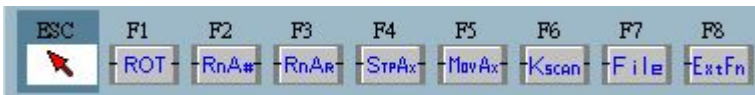
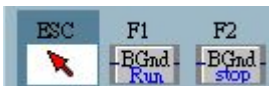
SYNTEC

功能說明	快捷鍵
元件參數設定 游標需在欲編輯之元件上 	Enter
移動目前位置 游標需在編輯區中 	方向鍵
新增行 新增的一行會加在游標所在位置的上方 	Ctrl + L
跳到指定行號 	Ctrl + G

元件切換快捷鍵



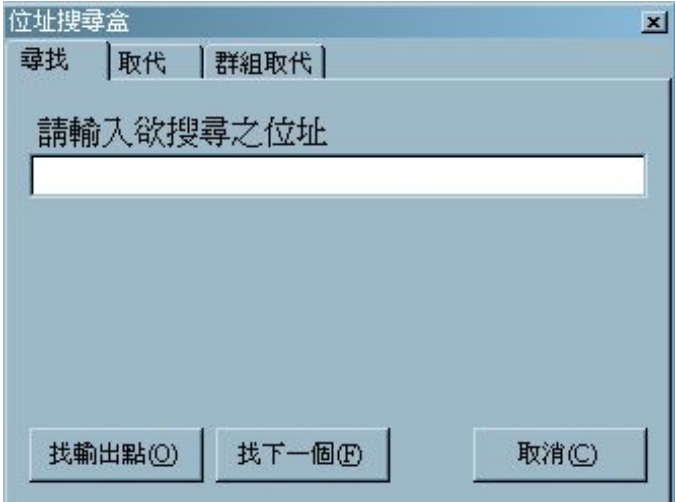
功能	快捷鍵
切換頁籤	Ctrl + 左/右
到指定頁籤	Ctrl + 數字鍵
切換該頁籤下的元件	Esc, F1~F10
切換到Contact頁籤	Ctrl + 1
	
切換到Coil頁籤	Ctrl + 2
	
切換到Arithmetic頁籤	Ctrl + 3
	
切換到Compare頁籤	Ctrl + 4
	
切換到Move頁籤	Ctrl + 5
	
切換到Logic頁籤	Ctrl + 6
	

功能	快捷鍵
切換到Timer頁籤 	Ctrl + 7
切換到Counter頁籤 	Ctrl + 8
切換到Flow頁籤 	Ctrl + 9
CNC頁籤 	目前只能使用方向鍵做切換
RS232頁籤 	目前只能使用方向鍵做切換
BGnd頁籤 	目前只能使用方向鍵做切換

檔案相關快捷鍵

功能	快捷鍵
開新檔案	Ctrl + N
開啟舊檔	Ctrl + O
儲存檔案	Ctrl + S

檢視相關快捷鍵

功能	快捷鍵
位址搜尋 	Ctrl + F
取代 	Ctrl + R

功能	快捷鍵
移到下一個相同的物件 若是游標在元件上，會跳到下一個相同物件 使用範例：游標在元件上  按下Ctrl + F3 後會變成 	Ctrl + F3

程式編輯快捷鍵

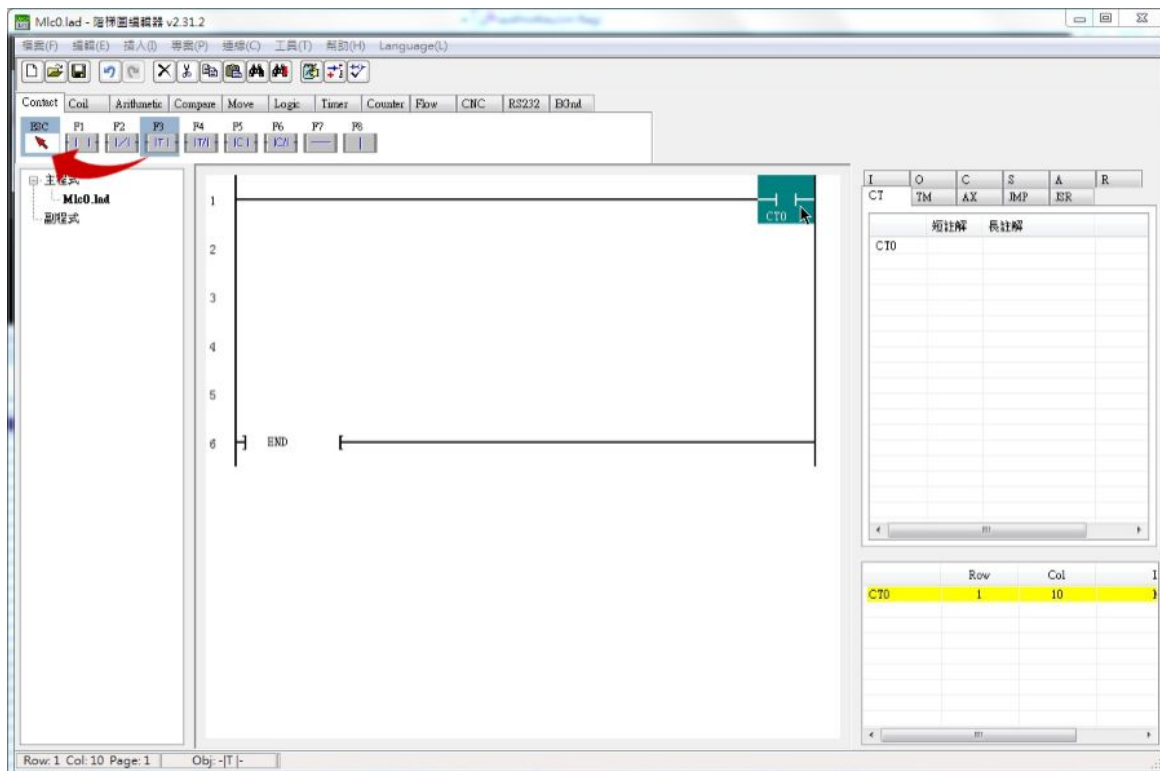
功能	快捷鍵
剪下	Ctrl + X
複製	Ctrl + C
貼上	Ctrl + V
刪除文件	Delete
復原 (undo)	Ctrl + Z
取消復原 (redo)	Ctrl + Y

操作功能小技巧

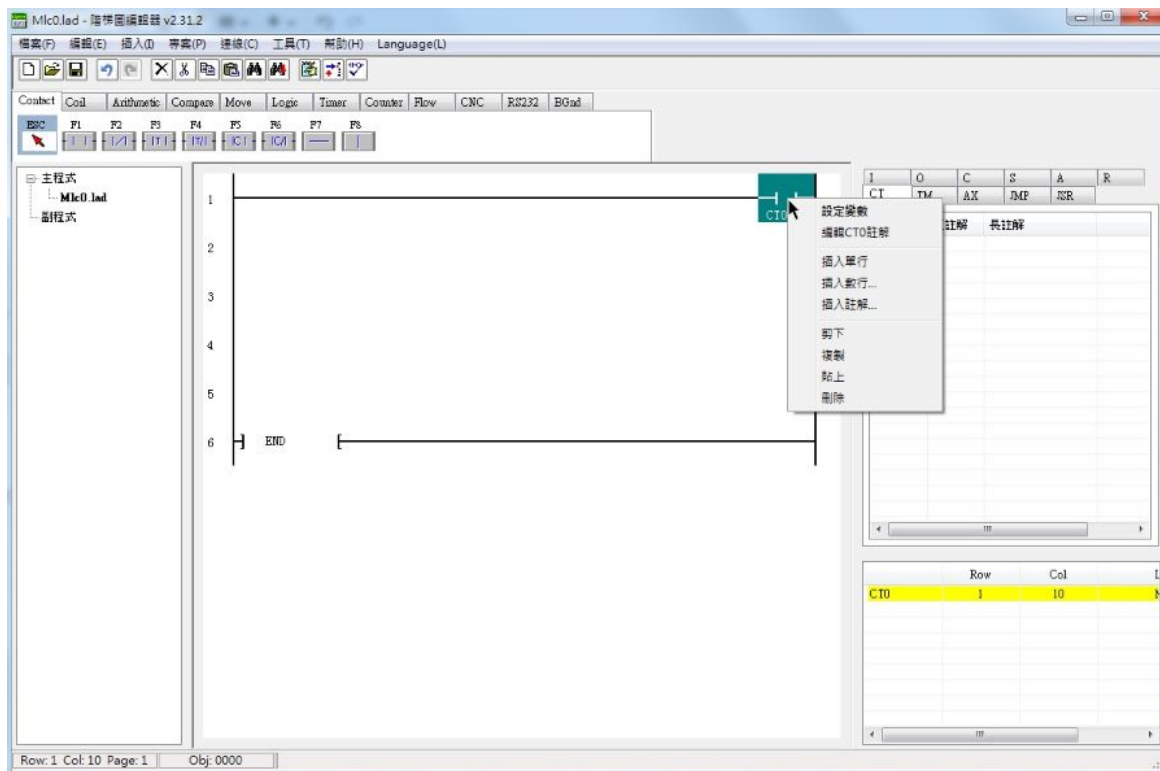
指令編輯區

在指令編輯區按下滑鼠右鍵時，有以下兩種狀況：

- 若此時上方指令區為"非 ESC 狀態"時，按下滑鼠右鍵會將狀態復原為"ESC 狀態"，意即沒有選到任一指令元件



- 若此時上方指令區本來就是"ESC 狀態", 按下滑鼠右鍵會跳出右鍵功能選單



元件索引區

使用者透過滑鼠選擇"指令編輯區"或"元件列表區"中的元件時，右下角"元件索引區"的黃色游標會標示出被選中的元件。

- 雙擊元件索引區中的元件，指令編輯區中的游標會自動移動到被選中的元件，但不會移動元件索引區的黃色光標（支援版本：2.26.0以前，2.31.4以後）
- 雙擊元件索引區中的元件，指令編輯區中的游標會自動移動到被選中的元件，同時移動元件索引區的黃色光標（支援版本：2.26.1 ~ 2.31.3）



SYNTEC

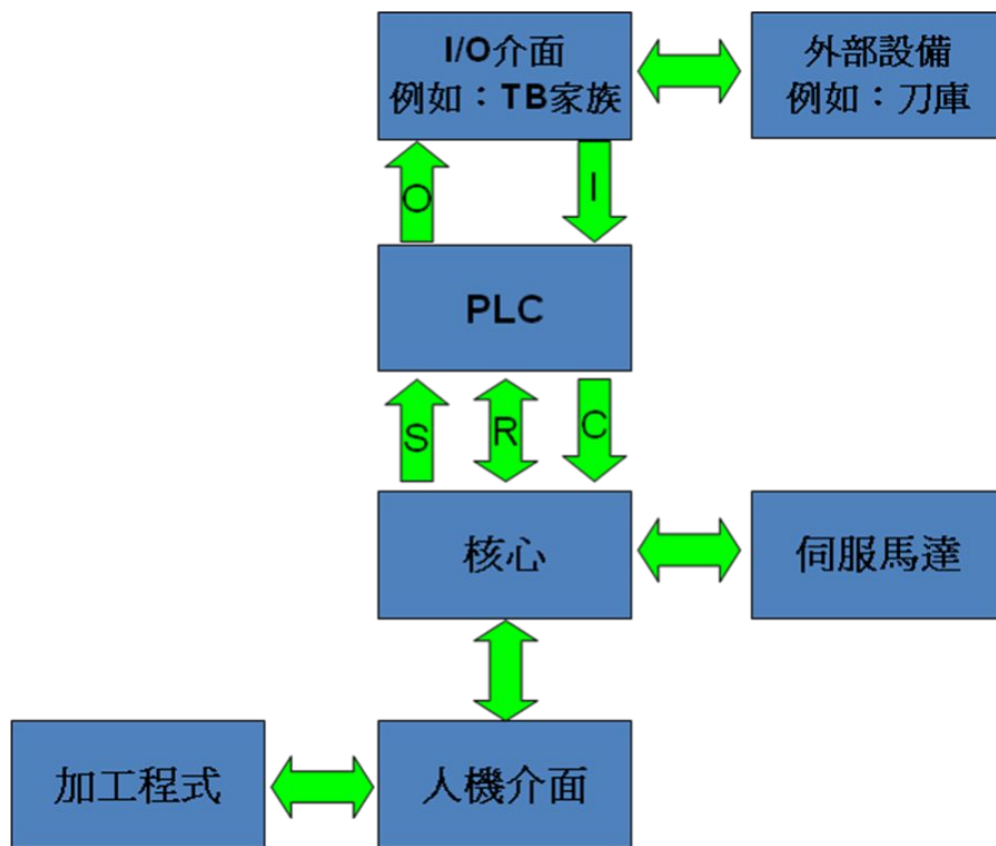
2 PLC元件介紹

2.1 階梯圖元件簡介

新代階梯圖主要提供I、O、A、C、S、R、CT、TM等8類元件，供程式編輯人員使用，詳細說明如下表所示：

名稱	定義	規格	數量
I	實體輸入接點 (Input)	規格與數量依所選用之I/O卡而定，只有On/Off兩種狀態	
O	實體輸出接點 (Output)		
A	虛擬輔助接點 (Auxiliary)	只有On/Off兩種狀態	A0~A511共512個
C	由階梯圖發送指令給核心	只有On/Off兩種狀態	C0~C511共512個
S	由核心傳送系統狀態給階梯圖	只有On/Off兩種狀態	S0~S511共512個
R	階梯圖與核心雙向溝通	32位元暫存器，可存取數值或拆開成位元使用	R0~R65535共65536個 *DOS僅支援R0~R7999
CT	虛擬輔助計數器 (Counter)	可選擇上數、下數或循環計數	CT0~CT255共256個
TM	虛擬輔助計時器 (Timer)	可選擇4種單位：1秒、0.1秒、0.01秒、0.001秒	TM0~TM255共256個





2.2 編輯器元件說明

2.2.1 接點元件（Contact）

常開接點（Normal Open Contact）

1. 圖示：

一般接點	計時器接點	計數器接點

2. 說明：俗稱A接點。常開接點平時斷開，有訊號時才導通。

3. 數量：I/O/C/S/A各有512個，計時器/計數器各有256個，R bit暫存器只有R0~R639可於階梯圖中進行32位元之存取，例如：R50.0 ~ R50.31，這區間的R bit才能使用接點元件。

常閉接點（Normal Close Contact）

1. 圖示：

		
一般接點	計時器接點	計數器接點

2. 說明：俗稱B接點。常閉接點平時導通，有訊號時斷開。

3. 數量：I/O/C/S/A各有512個，計時器/計數器各有256個，R bit暫存器只有R0~R639可於階梯圖中進行32位元之存取，例如：R50.0 ~ R50.31，這區間的R bit才能使用接點元件。

連接線 (Connector)

1. 圖示：

	
水平連接線	垂直連接線

2. 說明：用於連通其它元件或填補階梯圖中空白的地方。

2.2.2 線圈元件 (Coil)

常開線圈 (Normal Open Coil)

1. 圖示


常開線圈

2. 說明：當輸入訊號為ON，輸出為ON；當輸入訊號為OFF，輸出為OFF。

3. 數量：O/C/A各有512個，R bit暫存器只有R0~R639可於階梯圖中進行32位元之存取，例如：R50.0 ~ R50.31，這區間的R bit才能使用線圈元件。

常閉線圈 (Normal Close Coil)

1. 圖示


常閉線圈

2. 說明：當輸入訊號為ON，輸出為OFF；當輸入訊號為OFF，輸出為ON。

3. 數量：O/C/A各有512個，R bit暫存器只有R0~R639可於Ladder中進行32位元之存取，例如：R50.0 ~ R50.31，這區間的R bit才能使用線圈元件。

正緣觸發脈沖線圈 (Positive Edge Triggered Coil)

1. 圖示



正緣觸發脈沖線圈

2. 說明：
 - a. 當輸入訊號由OFF變成ON時，輸出一個脈沖信號。
 - b. 在快速掃描區的脈沖信號之持續時間為1個插補時間(Pr3203)。
 - c. 在一般掃描區的脈沖信號之持續時間為1個階梯圖掃描時間(Pr3204)。
 - d. 正緣觸發脈沖線圈放在快速掃描區時，無法觸發放在一般掃描區的元件。
3. 數量：O/C/A各有512個，R bit暫存器只有R0~R639可於階梯圖中進行32位元之存取，例如：R50.0 ~ R50.31，這區間的R bit才能使用線圈元件。

負緣觸發脈沖線圈（Negative Edge Triggered Coil）

1. 圖示



負緣觸發脈沖線圈

2. 說明：
 - a. 當輸入訊號由ON變成OFF時，輸出一個脈沖信號。
 - b. 在快速掃描區的脈沖信號之持續時間為1個插補時間(Pr3203)。
 - c. 在一般掃描區的脈沖信號之持續時間為1個階梯圖掃描時間(Pr3204)。
 - d. 負緣觸發脈沖線圈放在快速掃描區時，無法觸發放在一般掃描區的元件。
3. 數量：O/C/A各有512個，R bit暫存器只有R0~R639可於階梯圖中進行32位元之存取，例如：R50.0 ~ R50.31，這區間的R bit才能使用線圈元件。

保持線圈（Latch Coil）

1. 圖示



保持線圈

2. 說明：當輸入訊號ON時，輸出為ON且一直保持ON。
3. 數量：O/C/A各有512個，R bit暫存器只有R0~R639可於階梯圖中進行32位元之存取，例如：R50.0 ~ R50.31，這區間的R bit才能使用線圈元件。

復歸線圈（Unlatch Coil）

1. 圖示



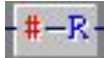
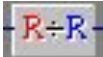
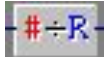
復歸線圈

2. 說明：當輸入訊號ON時，輸出為OFF且一直保持OFF。

3. 數量：O/C/A各有512個，R bit暫存器只有R0~R639可於階梯圖中進行32位元之存取，例如：R50.0 ~ R50.31，這區間的R bit才能使用線圈元件。

2.2.3 算數運算元件 (Arithmetic)

1. 圖示

	
$R2 = R2 + R1$	$R2 = R2 + \text{常數}$
	
$R2 = R2 - R1$	$R2 = R2 - \text{常數}$
	
$R2 = R2 \times R1$	$R2 = R2 \times \text{常數}$
	
$R2 = R2 \div R1$	$R2 = R2 \div \text{常數}$

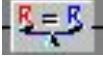
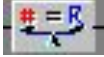
PS: 符號左邊為R1；右邊為R2。

2. 說明:

- 當輸入訊號為ON時，執行算數運算。
- #表示常數，其輸入範圍為-32768到32767。
- 本元件可支援R bit暫存器間接定值功能

2.2.4 比較元件 (Compare)

1. 圖示

	
是否R1大於R2	是否常數大於R2
	
是否R1小於R2	是否常數小於R2
	

是否R1等於R2	是否常數等於R2
----------	----------

PS: 符號左邊為R1; 右邊為R2。

2. 說明:

- 當比較結果成立 (True) 時, 輸出訊號為ON; 當比較結果不成立 (False) 時, 輸出訊號為OFF。
- #表示常數, 其輸入範圍為-32768到32767。
- 本元件可支援R bit暫存器間接定值功能

2.2.5 搬移元件 (Move)

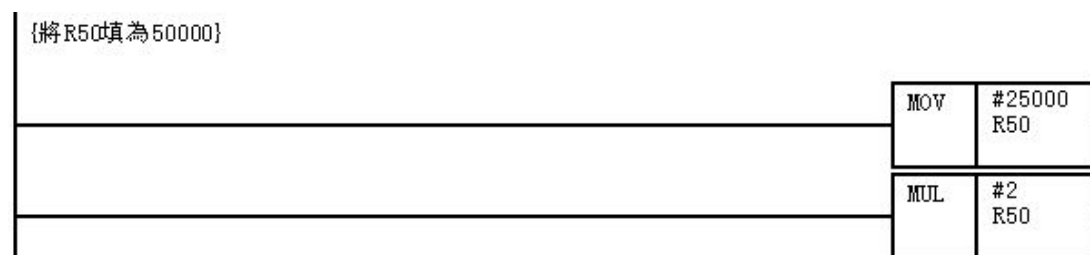
1. 圖示

	
將常數搬移到R2	將R1搬移到R2

PS: 符號左邊為R1; 右邊為R2。

2. 說明:

- 當輸入訊號ON時, 執行搬移動作。
- #表示常數, 其輸入範圍為-32768到32767。
- 本元件可支援R bit暫存器間接定值功能
- 若欲填入之數值超過範圍, 請參考下圖方式處理。



2.2.6 邏輯運算元件 (Logic)

1. 圖示

	
R2 = R2 AND R1	R2 = R2 AND 常數
	
R2 = R2 OR R1	R2 = R2 OR 常數

	
R2 = R2 XOR R1	R2 = R2 XOR 常數

PS: 符號左邊為R1; 右邊為R2。

2. 說明: 當輸入訊號ON時, 執行邏輯運算。每一個R bit暫存器數值均為32位元形式, 因此每次運算, 均針對32位元單獨執行邏輯運算。

AND真值表			OR真值表			XOR真值表		
R1	R2	結果	R1	R2	結果	R1	R2	結果
Off	Off	Off	Off	Off	Off	Off	Off	Off
Off	On	Off	Off	On	On	Off	On	On
On	Off	Off	On	Off	On	On	Off	On
On	On	On	On	On	On	On	On	Off

2.2.7 計時器元件 (Timer)

1. 圖示

由常數設定的計時器	由暫存器設定的計時器
	
單位: 0.001秒	單位: 0.001秒
	
單位: 0.01秒	單位: 0.01秒
	
單位: 0.1秒	單位: 0.1秒
	
單位: 1秒	單位: 1秒

2. 說明:

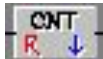
- a. 當輸入訊號由OFF變ON, 該計時器歸零。當輸入訊號一直保持ON時, 該計時器開始計時。

- b. 當計時器到達設定時間，輸出訊號為ON，其對應的計時器接點亦為ON。當輸入訊號為OFF時，計時器停止計時，輸出訊號為OFF，其對應的計時器接點亦為OFF。
- c. 使用暫存器設定的計時器時，若暫存器數值改變，僅當輸入訊號由OFF變ON時，計時器元件的設定時間才會更新為新的數值，同時該計時器歸零並重新計時。
3. 數量：共可使用256個（TM0~TM255）

2.2.8 計數器元件（Counter）

單程計數器（One Way Counter）

1. 圖示

	
由常數設定的向上計數器	由暫存器設定的向上計數器
	
由常數設定的向下計數器	由暫存器設定的向下計數器

2. 說明：

- a. 當輸入訊號由OFF變ON時，計數器向上增加（或向下減少）一次，直到次數到達目標值才停止計數。
- b. 當計數器到達目標值時，輸出訊號為ON，其對應的計數器接點亦為ON。
- c. 向上計數器的初始值為0，目標值為暫存器或常數之數值；向下計數器的初始值為暫存器或常數之數值，目標值為0。
- d. 使用暫存器設定的計數器時，若暫存器數值改變，僅當輸入訊號由OFF變ON時，計數器元件的目標值才會更新為新的數值並繼續計數，目標值改變不會造成計數器從初始值計數。
- e. 若在修改暫存器數值後，出現目標值小於目前值的狀況，在輸入訊號由OFF變ON時，計數器更新後會判斷到達目標值，輸出訊號為ON，其對應的計數器接點亦為ON。
- f. 單程計數器意指到達目標值後不會自動回復初始值，需使用計數器復歸元件。
3. 數量：共可使用256個（CT0~CT255）

循環計數器（Ring Counter）

1. 圖示

	
由常數設定的向上循環計數器	由暫存器設定的向上循環計數器
	
由常數設定的向下循環計數器	由暫存器設定的向下循環計數器

2. 說明：

- a. 當輸入訊號由OFF變ON時，計數器向上增加（或向下減少）一次，直到計數器數值到達目標值才停止計數。
 - b. 當計數器到達目標值時，輸出訊號為ON，其對應的計數器接點亦為ON。
 - c. 循環計數器數值到達目標值時，再一次觸發會回到初始值加一（或減一）之數值，一直持續循環計數。
 - d. 向上計數器的初始值為0，目標值為暫存器或常數的值；向下計數器的初始值為暫存器或常數的值，目標值為0。
 - e. 使用暫存器設定的計數器時，若暫存器數值改變，僅當輸入訊號由OFF變ON時，計數器元件的目標值才會更新為新的數值並繼續計數，目標值改變不會造成計數器從初始值計數。
 - f. 循環計數器到達目標值會自動從初始值計數，因此若計數到7，暫存器數值修改變更為5時，在輸入訊號由OFF變ON時，因計數器更新後同時「數值到達目標值」及「再一次觸發」，因此計數器會直接回到初始值加一（或減一）之數值，輸出訊號不會ON。
3. 數量：共可使用256個（CT0~CT255）。

計數器復歸（Counter Reset）

1. 圖示



2. 說明：當輸入訊號由OFF變ON時，該計數器數值變為初始值。符號#所輸入的數值，對實際執行並無任何影響。

2.2.9 流程相關元件（Flow）

副程式呼叫（Jump Subroutine）

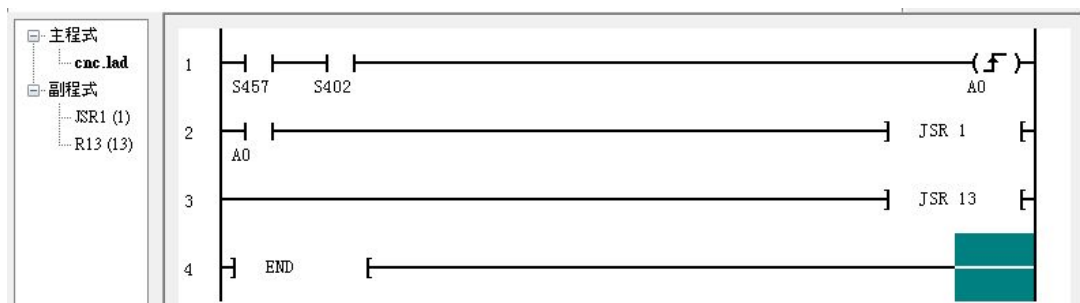
1. 圖示



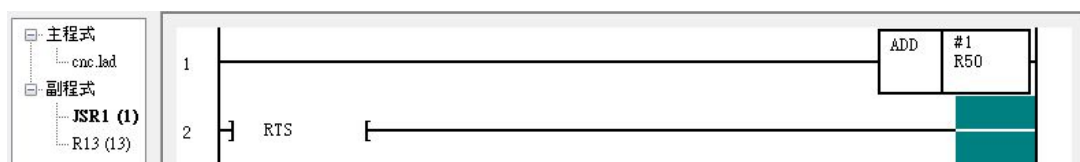
2. 說明：當輸入訊號為ON時，執行副程式呼叫指令，並跳到對應標籤代號之副程式內開始執行；副程式結束，則返回主程式執行下一個元件指令。
3. 注意事項：
- a. 使用副程式前，請妥善進行內容規劃，控制副程式數量。例如使用間接定址的方式減少內容相同的副程式。
 - b. 當主程式內容與副程式內容的行數加總超過4000行(含注解)時，副程式執行完畢時可能會無法正確跳躍回主程式，造成階梯圖執行形為異常。
 - c. 副程式數量過多可能造成階梯圖異常，導致JSR元件找不到對應的副程式，此為目前架構之限制。請減少副程式數量，以避開程式跳躍超過系統限制的語法檢查。
 - d. JSR元件被觸發的次數等於副程式被掃描的次數，換言之，若主程式內並未觸發副程式，則副程式的內容也不會被即時更新，只有在下一次觸發時才會被更新。
4. 範例：
- a. 主程式中有2個副程式，副程式JSR1被執行的次數記錄於R50中，副程式R13被執行的次數等同於主程式的掃描次數。

b. 當在副程式R13內進行模式切換時，R13暫存器的數值會在下一個掃描周期被更新。

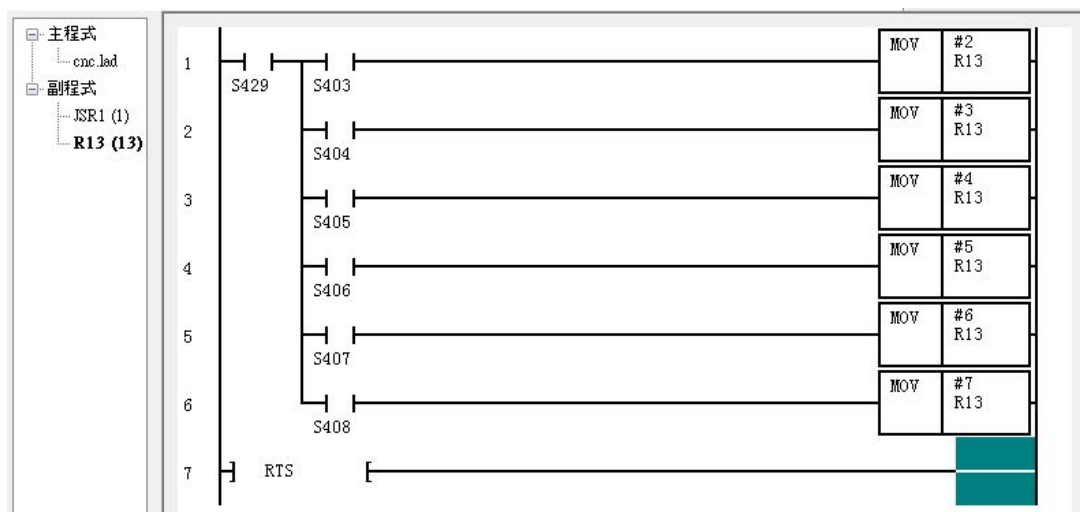
c. 主程式



副程式JSR1，標籤代號 "1"



副程式R13，標籤代號 "13"

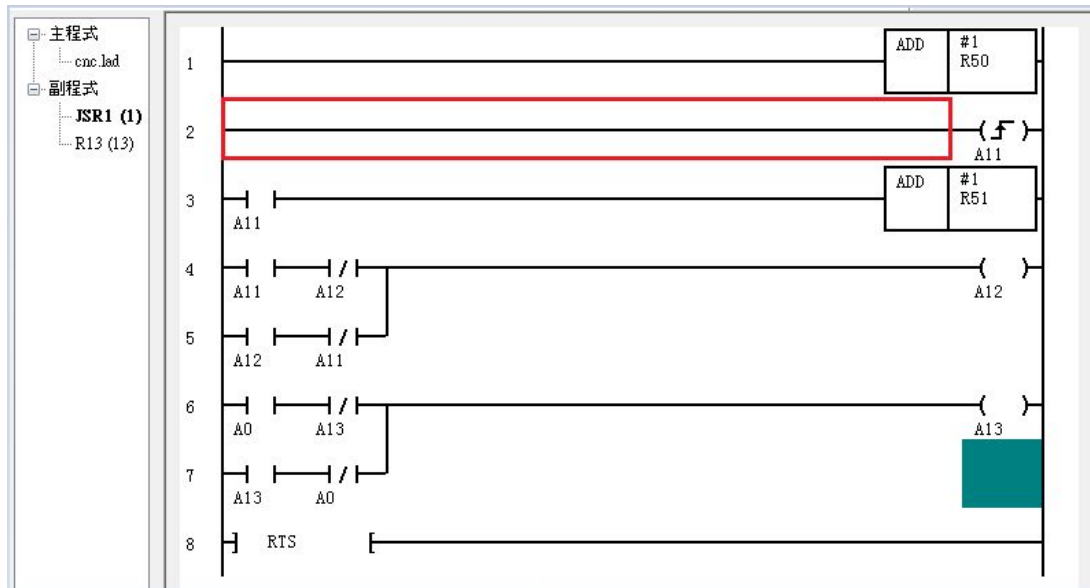


5. 特殊規格範例

延續上一範例，修改副程式JSR1內容如下圖：

- A11只在副程式JSR1第一次被執行時有效，故R50會隨著JSR1的執行次數遞增，R51在第一次執行過後恒為1。
- A12在第一次執行過後恒為ON，因為A11已經不會再有動作。

c. 隨著每一次副程式JSR1被執行，A13可進行一鍵開關的動作。



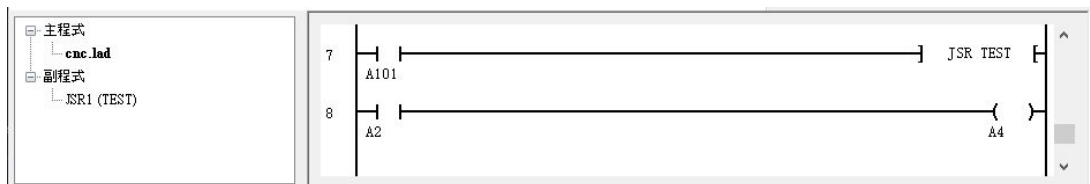
6. 特殊規格範例-2

此範例針對[3.注意事項]的c點進行演示

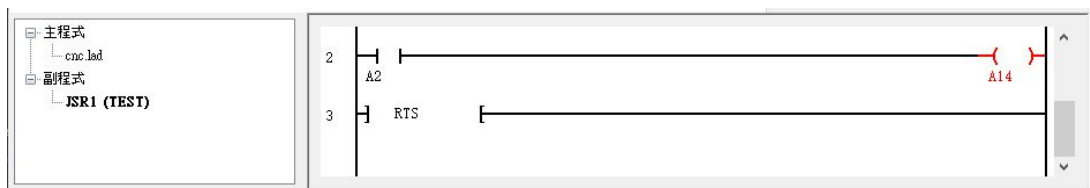
a. 副程式的內容僅在觸發(範例為A101 ON)時進行更新，當副程式不被執行時(A101 OFF)會出現A4與A14不同步(兩者皆由A2導通)的狀況

b. 演示內容

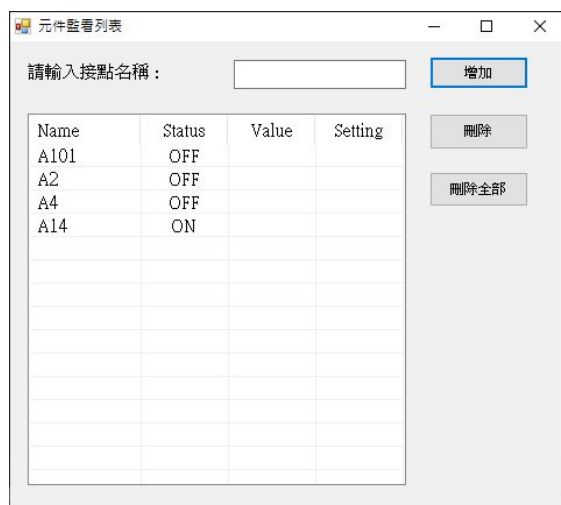
主程式內



副程式TEST，標籤代號 "TEST"內



由於A101為OFF，副程式內容不被更新，故會出現A4與A14狀態不同的情況



程式結束 (End)

1. 圖示



2. 說明：此為階梯圖程式結束指令，為階梯圖內最後一個判斷式，意即無法在此元件下方編輯元件。

快速PLC掃描 (FastPLC)

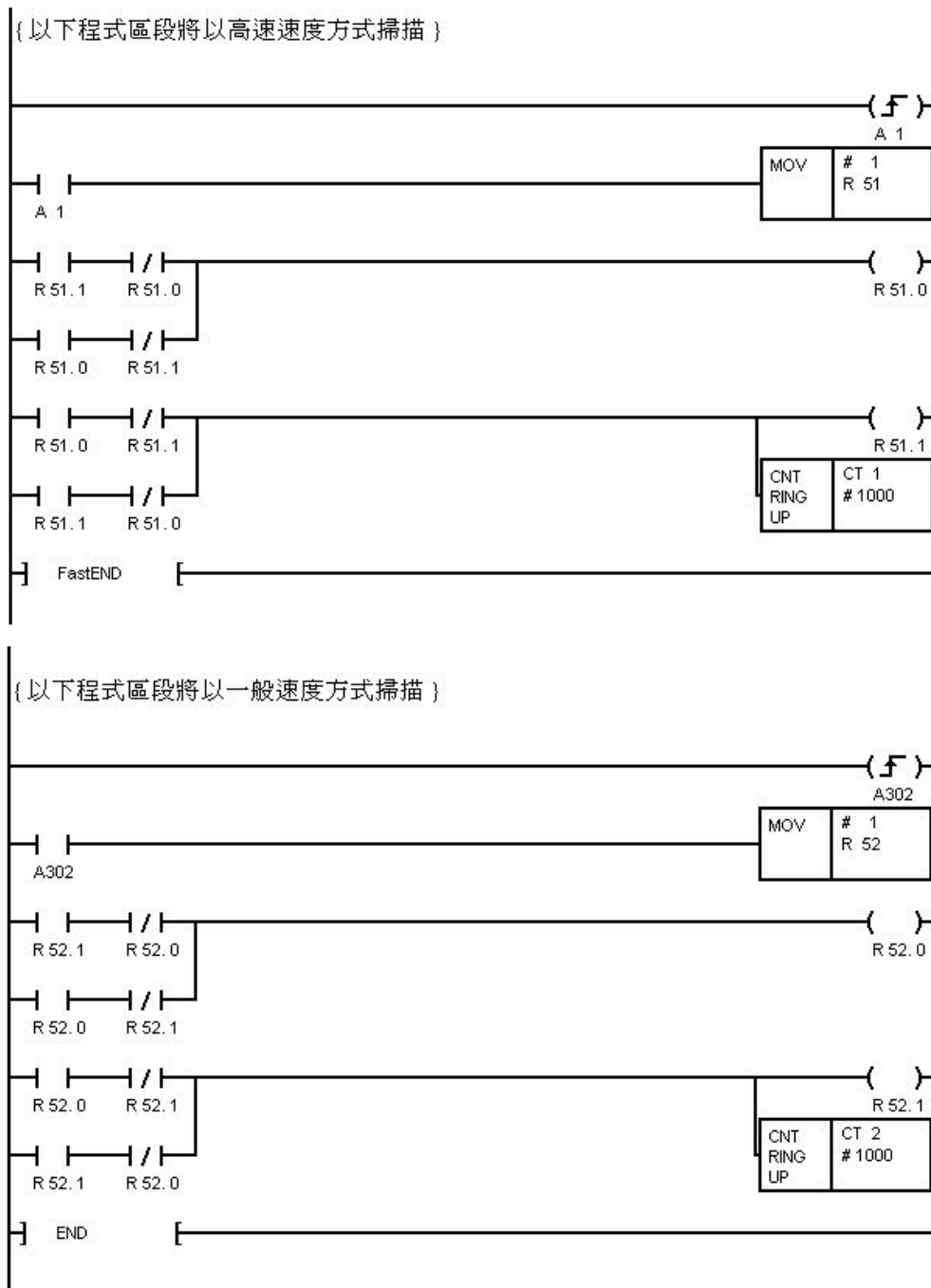
1. 圖示



2. 說明

- 快速PLC掃描的內容請寫在階梯圖的最前面，並以-]FEND[-元件結束。
- 在執行階梯圖內容時，-]FEND[-之前的程式內容會被載入到快速掃描區。
-]FEND[-之後到-]END[-之間的程式內容會被載入到一般掃描區。
- 快速掃描區的掃描時間與Pr3203（運動插補時間）相同。
- 一般掃描區的掃描時間與Pr3204（PLC掃描時間）相同。
- 注意：快速掃描區內，不可放入"副程式呼叫"元件

3. 範例



2.2.10 CNC相關元件 (CNC)

刀庫旋轉 (Rotate)

1. 圖示



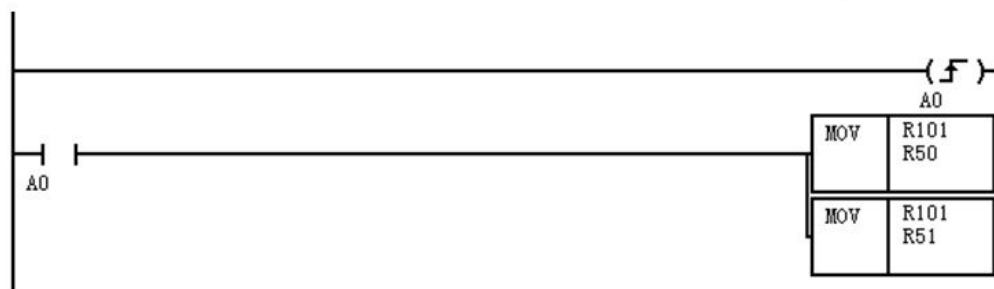
刀庫旋轉

2. 說明

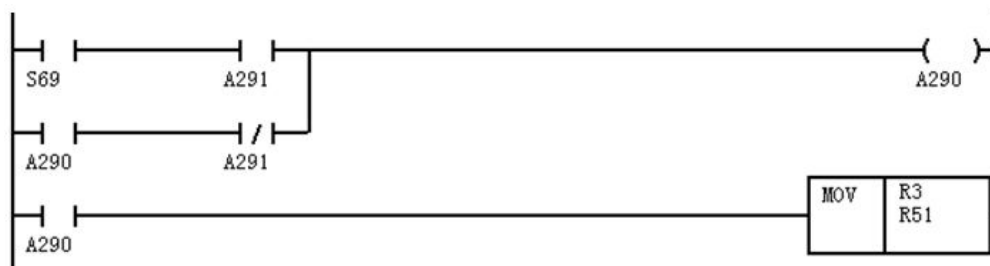
- a. 當輸入訊號為ON時，執行刀庫旋轉計算，並依據計算結果輸出1正轉（CW）與0反轉（CCW）兩種訊號。所用暫存器編號說明如下：
 - i. Ra為目前刀號1~n
 - ii. R[a+1]為目標刀號1~n
 - iii. Rb為總刀數n
 - b. 當此指令被執行時，目前刀號如超出總刀數範圍1~n時，會自動被修改換算為對等範圍內的刀號。
 - c. 此元件以最短距離方式，計算正反轉訊號，例如總刀數為8，目前刀號為4，當目標刀號為6時，將輸出0反轉訊號；反之，當目標刀號為2時，將輸出1正轉訊號。
3. 範例：假設將R50設定為目前刀號，R89設定為總刀數。系統將自行設定R51為目標刀號（R[50+1]）。以下是使用T碼換刀的簡易換刀PLC說明。

接點	說明	接點	說明
A0	上緣觸發	A293	換刀方向訊號
A16	刀庫正轉狀態	I21	數刀sensor
A17	刀庫反轉狀態	R101	主軸刀號
A290	換刀狀態	R102	刀庫刀號
A291	等待換刀狀態		

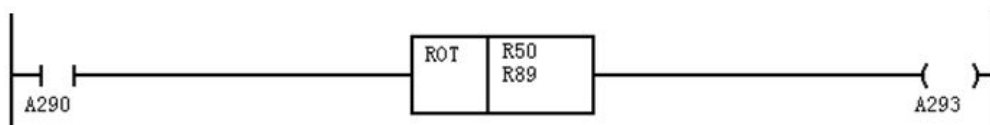
開機上緣將主軸刀號（R101）寫入目前刀號（R50）與目標刀號（R51），作一個初始化動作。



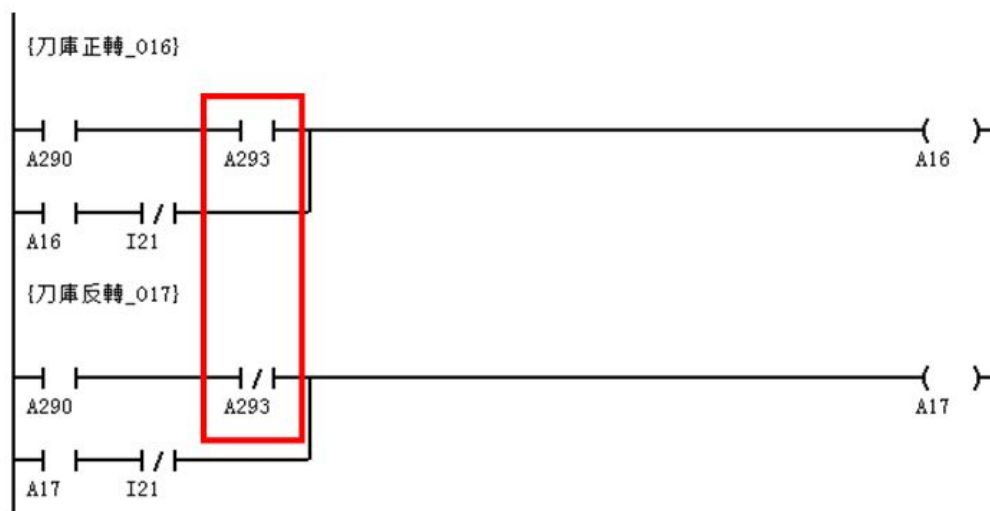
當使用者使用T碼換刀時，進入換刀狀態（A290），並將T碼的刀號（R3）寫入目標刀號（R51）



此時利用刀庫旋轉元件（ROT）決定正/反轉，正轉輸出"1"，A293為ON；反轉輸出"0"，A293為OFF。

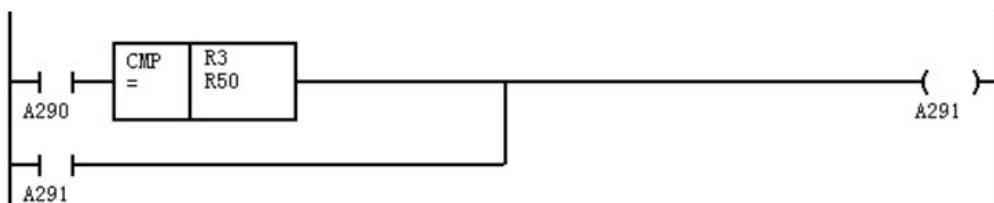
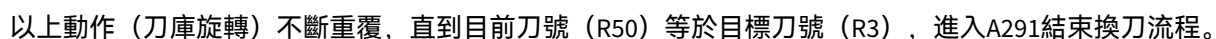


A293決定刀庫正轉（A16）或反轉（A17）



當收到數刀sensor（I21）的訊號後，表示刀庫已進行旋轉，將目前刀號加一（或減一）。同時更新主軸刀號（R101）與刀庫刀號（R102）。

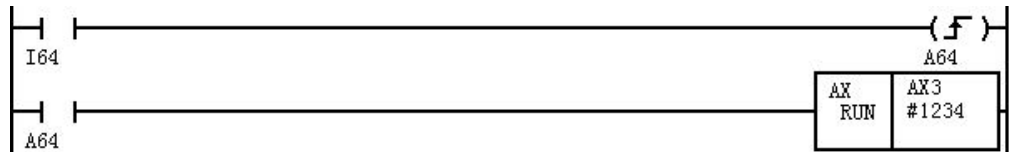
SYNTEC



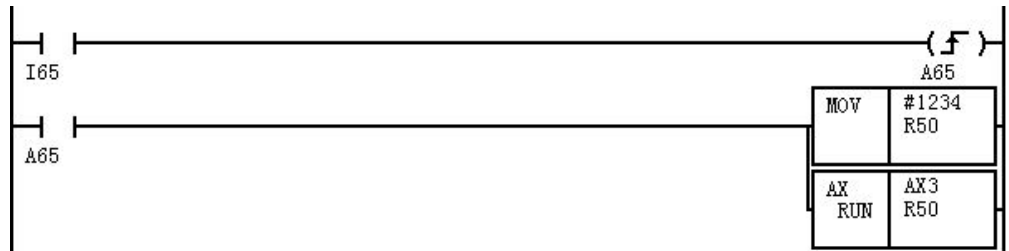
1. 圖示

2. 説明

- 加工程式檔名由數值指定，下圖範例為指定第三軸為PLC軸並執行檔名O1234的程式內容。



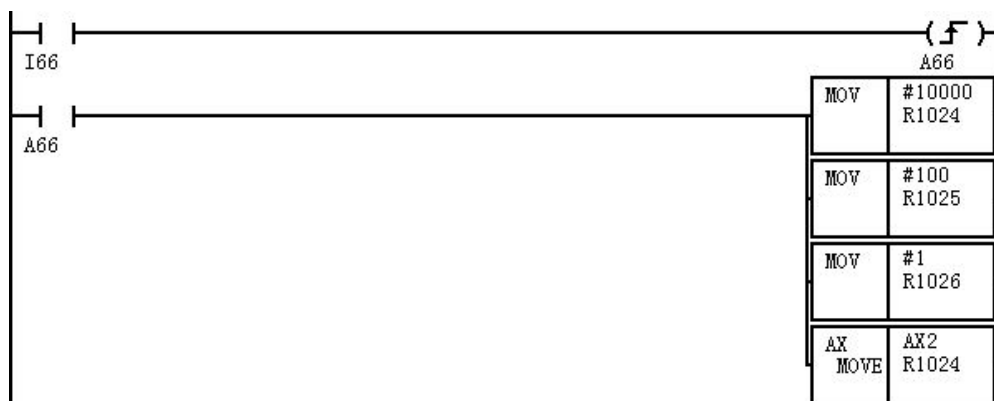
- 加工程式檔名由R bit暫存器指定，下圖範例為指定第三軸為PLC軸並執行檔名O1234的程式內容。



- PLC軸副程式說明 (Onnnn)
 - 程式名稱必須以"O"作檔名開頭，但是階梯圖內不需寫"O"，程式名稱"nnnn"一定是四碼自然數
 - DOS系統放於此路徑C:\CNC\MACRO
 - CE系統放於此路徑DiskC\OpenCnc\MACRO
 - 此程式需為MACRO格式，所以檔案內容起頭必須是%@MACRO
 - 只能執行單軸移動，且程式內容軸向名稱一律是X
 - 與軸向有關的系統變數只有X軸有效。例如#1301表示PLC軸單節終點座標，而#1302在PLC軸無定義，請不要使用
 - 所有單軸G碼都可使用 (G00、G01、G04、G90、G91、G94、G95...)
 - PLC軸副程式中的G00/G01，其速率段數 (Override) 由R661~R676決定，而非R18/R16
 - 所有移動G碼的速度/加速度/加加速度，均參考控制器參數設定，例：G00速度參考Pr461~，G00加速度參考Pr441~；G01速度參考Pr621~，G01加速度參考Pr541~。
 - 程式結尾不可下M02、M30或M99
 - 程式中若要執行M碼時，M碼讀取/ M碼完成/ M碼內容請改用R629/ R630/ R681，相當於S29/ C38/ R1。
 - 程式中不支援G10/L10/L11/L12/L13/L14等刀具相關設定功能

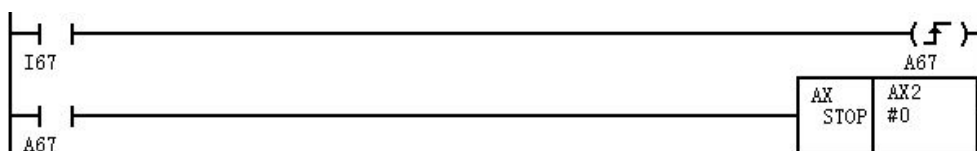
b. MoveAx元件，依指定的速度、位移量與G90/G91模式移動 (只支援公制單位的應用)

- MoveAx元件需要指定4個連續暫存器來設定動作內容，其定義如下：
 - R (基址+0)：位移量，單位：BLU。
 - R (基址+1)：移動速度，單位：mm/min。
 - R (基址+2)：絕對/增量模式選擇，0：增量模式，1：絕對模式。
 - R (基址+3)：不用寫，目前無功能，保留將來擴充用。
- MoveAx元件必須放在設定內容之後才會正常動作
- 範例：第二軸在絕對模式下，以100mm/min速度移動10mm，假設基址為1024。



- MoveAx的加速度設定參考Pr441~, 其速率段數 (Override) 由R18決定

3. Stp元件：中斷PLC軸正在執行的動作，例如：中斷正在移動中的第二軸。其中符號#所輸入的數值，對實際執行並無任何影響。



- 執行中的PLC軸動作也可以利用重置鍵 (Reset) 來中斷，但也會影響到其他動作，例如執行加工。
- 當指定軸向正在執行加工指令時，該軸的PLC軸元件**只能被觸發一次**，並可配合一次StpAx來停止該PLC軸元件的操作。後續則應等該軸完成加工指令，才能再次觸發該軸的PLC軸功能。
- 承 5，須注意，當軸向在加工且仍在移動中，若透過PLC軸元件對該軸下命令，將造成軸向指令疊加，可能產生錯誤的運動路徑。
- 執行PLC軸控制時，請以上緣觸發的方式，以免相同動作被多次觸發執行。
- PLC軸支援手輪模擬功能，細節請參閱**R601**說明。
- PLC軸控制元件不應在 JOG, INCJOG, MPG, HOME 模式下使用，可能會因為指令疊加而產生預期外的運動路徑，如需使用請做好相關的保護措施。
- 如果 PLC Rn副程式元件未成功借用漫游軸，就對該軸下命令，會發出警報 COR-365 對未借用的漫游軸，發出移動指令。
- 如果 PLC MoveAx元件觸發時，該軸向正被其他軸群或 PLC Rn副程式元件借用中，會發出警報 MOT-085 對借用中的漫游軸，觸發PLC MoveAx元件。

鍵盤掃描

1. 圖示



2. 說明

- 此元件被觸發時，啟動PLC鍵盤掃描，輸入資料由控制器上之PIO5 (I/O板) 的DB-25PIN讀入；輸入資料放置順序由低位元 (Low Bit) 往高位元 (High Bit) 依序放置。

b. 從PIO5上的DB-25PIN接腳定義可得知一個O點可掃入8個I點。如O點數目定義為5，則最多可輸入40個I點，輸入資料排列順序如下：

- i. 輸入資料暫存器Rin（基址+0）的位元0~31對應I點0~31
- ii. 輸入資料暫存器Rin（基址+1）的位元0~7對應I點32~39

c. PLC鍵盤掃描指令最少要使用兩個連續暫存器，R（基址+0）內容規劃如下，R（基址+1）的用法則依定址方式有所不同。

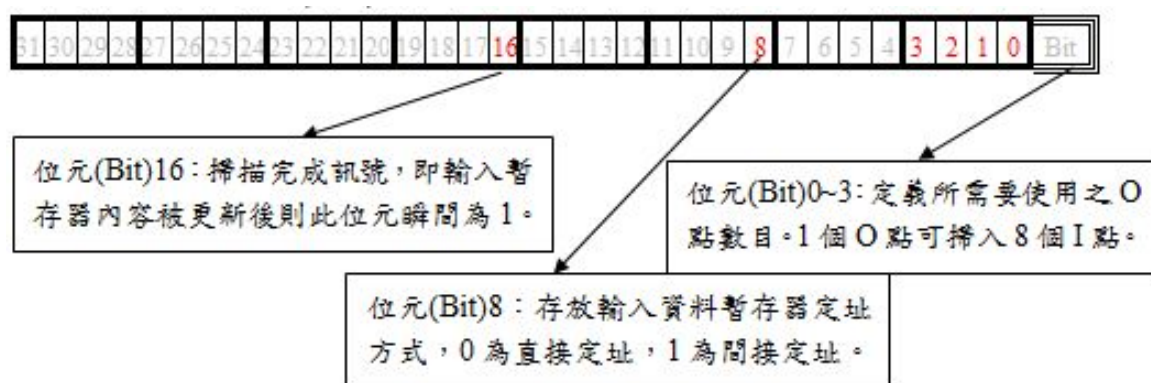
R（基址+0）：

位元0~3：定義所需要使用之O點數目，其值需要大於"所需I點數/8"。

位元8：存放輸入資料暫存器之定址方式，0為直接定址，1為間接定址。

位元16：掃描完成旗標，即輸入暫存器內容被更新後，此位元瞬間為1。

R（基址+0）的功能定義：



Rin為輸入資料暫存器，也就是最後被指定要存放資料的暫存器，其定義說明如下：

直接定址時，掃描內容將被放在R（基址+1）~ R（基址+4）；間接定址時，Rin將由R（基址+1）指定。

下表為Rin存放I點輸入的示意圖：

15	14	13	12	1	1	1	8	7	6	5	4	3	2	1	0	Bit
15	14	13	12	On	On	On	8	7	6	5	4	3	2	1	0	I點
31	30	29	28	27	26	25	24	23	1	1	1	19	18	17	16	Bit
31	30	29	28	27	26	25	24	23	On	On	On	19	18	17	16	I點

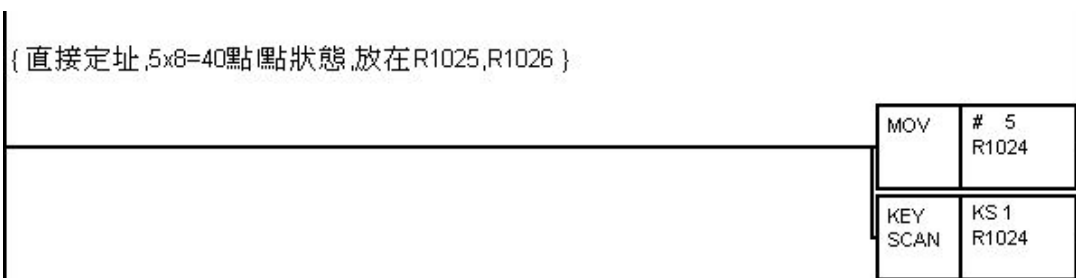
4. KS元件僅支援1組，且只有在使用PIO5硬體時有效。

5. 範例

a. 直接定址：使用5個O點當控制訊號，可以掃入40點I點，它的內容將被放在R1025與R1026。

R（基址+0）=R1024，Rin=R（基址+1）=R1025是第一個資料輸入暫存器，R（基址+2）

=R1026是第二個資料輸入暫存器



- b. 間接定址：使用5個O點當控制訊號，可以掃入40點I點，它的內容將被放在R2000與R2001。
 R（基址+0）= R1028，R1028=261表示bit 8=1為間接定址，bit 2=bit 0=1表示要使用5個O點
 R（基址+1）=R1029=2000，表示Rin=R2000是第一個資料輸入暫存器，R2001是第二個



PLC座標教導系統

1. 圖示



2. 說明

- 當輸入訊號由Off變On時，啟動PLC座標教導系統，PLC座標教導系統一經啟動後便一直有效，直到關機。
 - 使用者可以透過階梯圖來建立以暫存器為基礎的檔案系統，PLC座標教導系統元件可以依使用者的動作來做快速定位、直線切削等運動指令教導，並將教導結果存入檔案，讓使用者可以透過階梯圖做檔案管理。
 - PLC座標教導系統元件使用8個連續暫存器內容，其功能定義如下：
 - R（基址+0）：[In]，資料顯示區起始的暫存器位址
 - R（基址+1）：[In]，資料顯示區每一行占去幾個字元
 - R（基址+2）：[In]，資料顯示區一次顯示幾行
 - R（基址+3）：[In]，控制指令暫存器位址（教導、刪除、上、下等）
- 位元0：快速定位教導，由Off變On時，以此時機械座標做為終點。
 位元1：直線切削教導，由Off變On時，以此時機械座標做為終點。
 位元2：圓弧切削教導，目前為保留功能，無作用。
 位元3~5：保留區

位元6: 刪除行, 由Off變On時, 將游標所在的那一行刪除。

位元7: 儲存檔案, 由Off變On時, 將現在的編輯檔寫回磁碟裝置。

位元8: 由Off變On時, 將游標上移一行。

位元9: 由Off變On時, 將游標下移一行。

位元10: 由Off變On時, 將畫面切換到上一頁。

位元11: 由Off變On時, 將畫面切換到下一頁。

位元12~31: 保留區

R (基址+4) : [In], 指定教導檔 (加工檔) 檔名的暫存器, 檔名開頭為"O", 範圍為1 ~ 8999。

R (基址+5) : [Out], 座標教導系統目前的教導檔 (加工檔) 檔名。

R (基址+6) : 保留

R (基址+7) : 保留

[In]表示由使用者填入, [Out]表示由階梯圖回應。

3. 目前僅支援使用1個PLC座標教導系統

4. 附注: 資料顯示區單行格式

在使用者設定的單行長度中, 第一個暫存器 (占去4個字元) 用來存放該行的資料, 第二個暫存器開始才是存放字串資料 (一個暫存器可放4個字元), 第一個暫存器 (Line Header) 格式說明如下:

位元0~15: 存放該行的行號, 亦即該文件的第幾行 (由0行開始算起)

位元16: On表示該行是游標行, Off表示該行不是游標行

位元17~31: 保留區

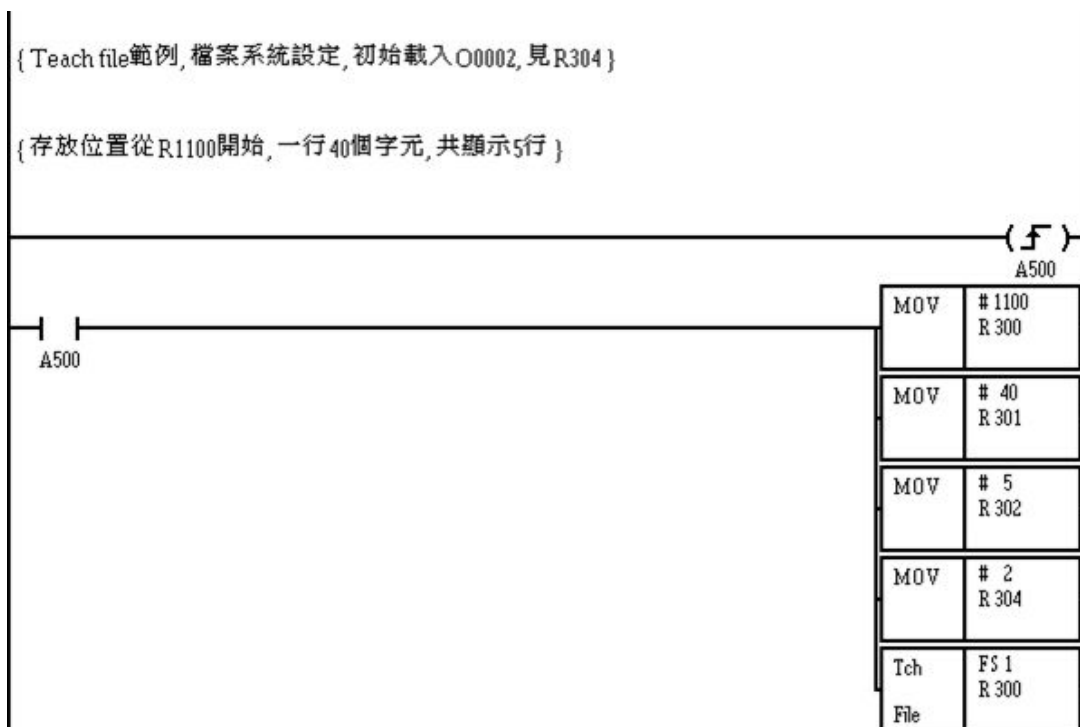


5. 範例

a. 設定PLC座標教導元件

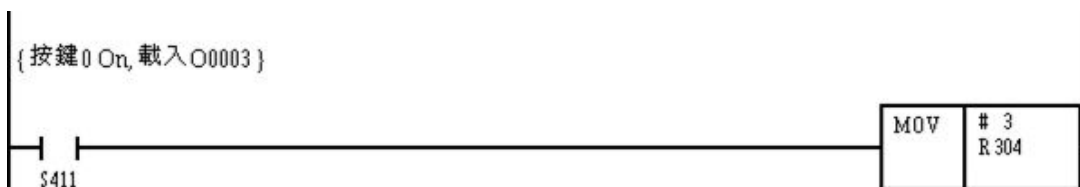
指定資料顯示區從R1100開始 (基址=300), 每行有40個字元, 總共有5行, 並設定初始化時載入的加工檔為O0002, 設定步驟執行一次即可

SYNTEC



b. 選擇教導檔（加工檔）

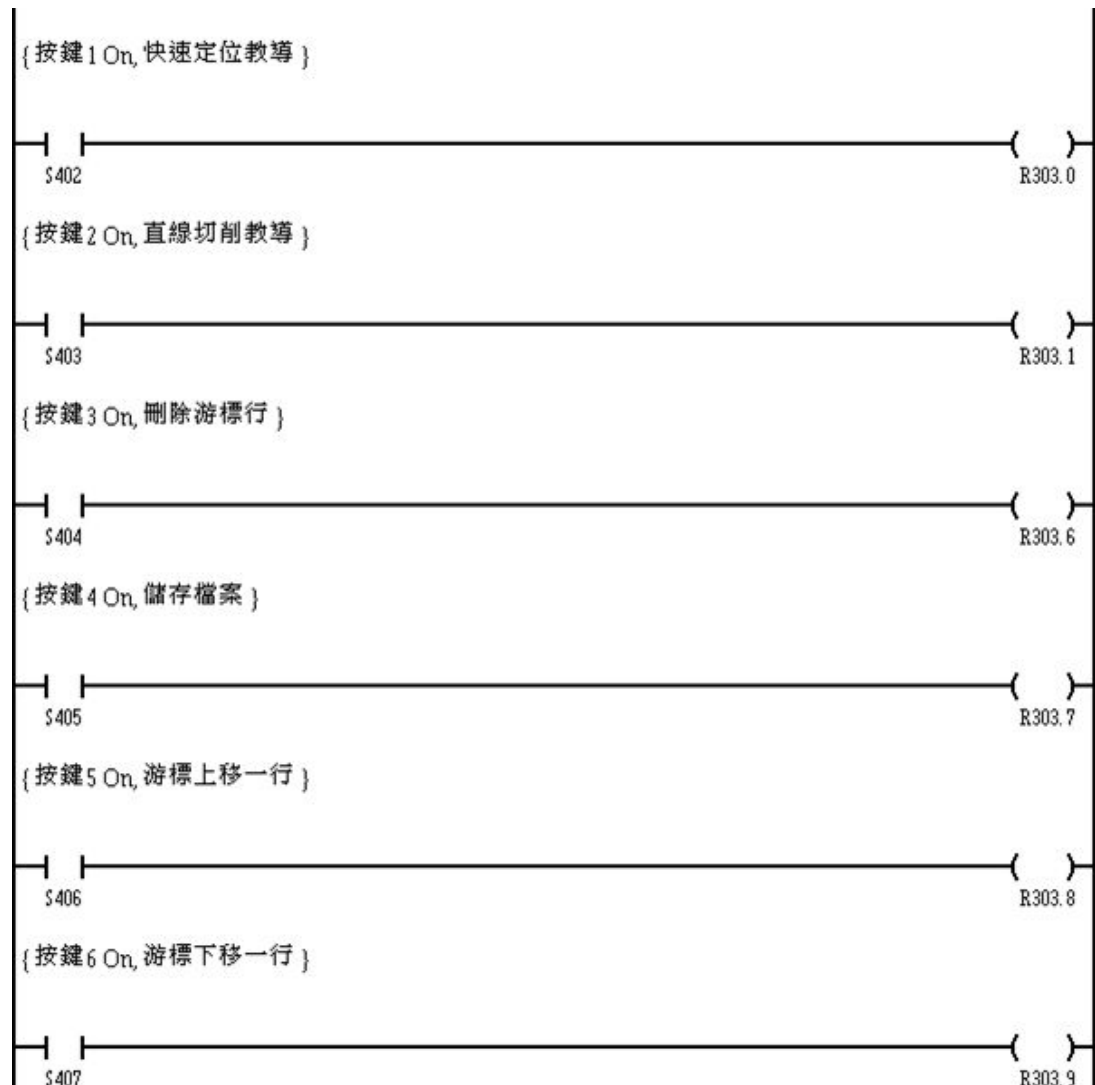
將欲教導加工的檔名放入R（基址+4）所指定的暫存器，在此例為R304，如下所示，表示要載入的加工檔為O0003



c. 教導動作控制

依照需要的動作去設定R（基址+3），讓暫存器對應的位元為On，在此例為R303，如下所示，依序可做快速定位教導、直線切削教導、刪除行、儲存檔案、游標上下移動等動作。

SYNTEC



2.2.11 RS232相關元件（RS232）

RS232傳輸指令

1. 圖示

		
	資料傳出與接收	

2. 說明

- a. 在輸入訊號由Off變為On時，啟動RS232傳送指令，從指定的傳送起始暫存器開始，將指定傳送個數的暫存器的數值，透過RS232傳送出去。
如有回傳值，RS232元件會將收到的資料從指定的接收暫存器開始依序填入，並將收到的位元組數記錄下來，然後回到等待狀態。
- b. 此元件需要8個連續暫存器來設定動作內容，其定義如下：
R（基址+0）：[傳出/接收]，RS232傳送元件控制暫存器。
位元0：傳送等待控制位元，此位元On，表示資料送出後元件需等待回傳值，才可繼續下次的資料傳送，此位元Off，表示無需等待即可進行下一次的資料傳送。
位元1~7：保留區
位元8：資料接收位元，正常情況為On，在元件輸入訊號由Off變為On時，會先將此位元變為Off，然後將資料傳送出去，當收到回傳值或是不需等待回傳值，才又變回On。
位元9：此位元On表示RS232傳送時發生錯誤，此位元Off表示傳送時無錯誤發生。
位元10：此位元On表示RS232接收時發生逾時錯誤（Time Out），此位元Off表示接收時無錯誤發生。
位元11：此位元On表示RS232傳輸元件的暫存器數值錯誤，此位元Off表示正確。
位元12：此位元On表示RS232傳輸接收時，有輸出入錯誤（I/O error）。
位元13~31：保留區
R（基址+1）：[使用者填]，傳送起始暫存器位址。
R（基址+2）：[使用者填]，傳送位元組個數（以R為單位）。
R（基址+3）：[使用者填]，回傳值存放起始暫存器位址。
R（基址+4）：[傳送/接收填]，指定接收的最大位元組個數，RS232會填回實際接收到的位元組個數。
R（基址+5）：[使用者填]，接收結束位元（Terminator）。
R（基址+6）：保留區
R（基址+7）：保留區
- c. 硬體Com Port與參數對應如下表所示

	SUPER/10/20/200	eHMC/3/6/11/21/22
RS232 (Pr3921~Pr3930)	COM1	-
RS422 (Pr3941~Pr3950)	COM2	-
RS485 (Pr3961~Pr3970)	COM3	COM2

3. 注意事項

- a. RS232元件只支援單一Com Port，例如：200系列控制器，若已使用RS232元件在COM1傳輸資料，就無法再使用RS232元件於其他Com Port進行傳輸。
- b. 應用上請勿使用上緣觸發RS232元件，因為上緣觸發無法一次跑完元件內部狀態流程(訊號必須On一段時間才能夠跑完)，導致每次觸發都不一定能送出封包的情況。

4. 範例

- a. 以4個狀態A500~A503來控制RS232的傳出與接收，下列分別說明四個狀態的動作，按鍵1（S402）可使流程進入A500。
A500：
RS232傳送資料準備，會觸發一個正緣觸發訊號A510以做傳送資料的準備，並立刻進入A501。
A501：
為RS232的要求（Request）狀態，會將RS232元件控制暫存器R（基址+0）的資料接收位元8設定為Off，以進入傳送接收資料狀態A502。
A502：
為RS232的要求（Request）狀態，若控制暫存器R（基址+0）的位元0為On的話，在此狀態會透過RS232傳出資料並等待回傳值，若位元0為Off，則不等待。

A503:

RS232回傳值處理，會觸發一個正緣觸發訊號A509，以做回傳值的處理，在收到按鍵1 (S402) 訊號後才會再進入A500

b. 傳出資料的準備

將預計由RS232傳出的資料從R1210開始，依序放入84、65、49、13，此4個數字為"TA1\n"的ASCII值

將傳送起始暫存器位址設為R1210，所以要將1210填入R301

傳送的長度設為4填入R302

回傳值存放起始暫存器位址設為R1410，所以要將1410填入R303

接收最大長度設為30填入R304，並指定13為接收結束符號

c. 回傳值的處理

在此預設回傳值為"EV 301 0 1 23.827 159.447\n"，從R1410開始填入，所以將資料開始處理的位址R1410填入R1400

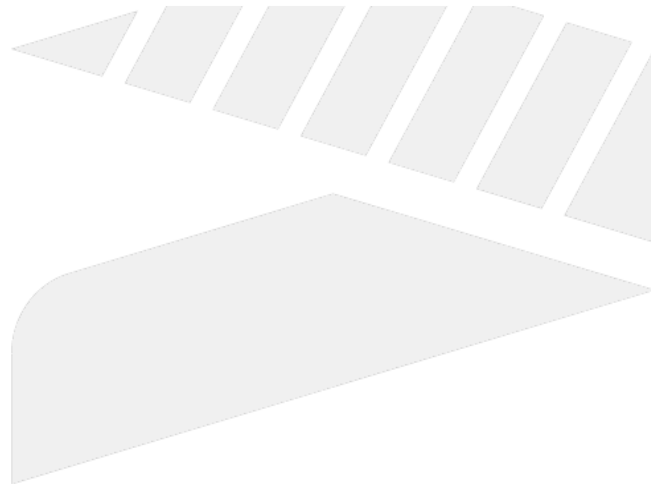
指定小數點位數為0（表示整數）填入R1401

指定數值的分隔符號為空白，其ASCII值為32，填入R1402

執行第一次的字元轉換成數字元件，可將"EV"轉成數字填入R1380，因為是英文字所以轉出來為0

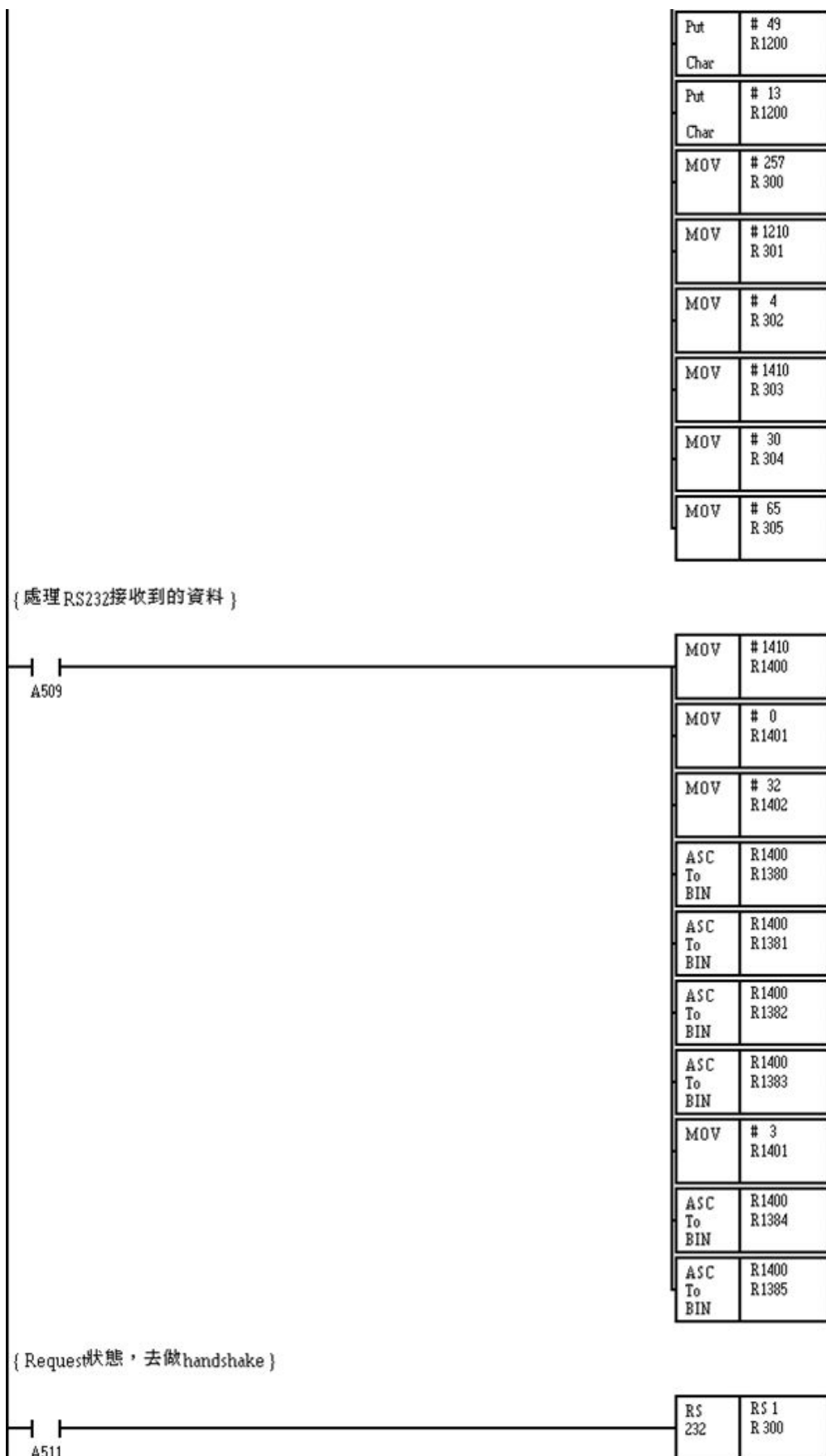
第二次執行可將"301"轉成301填入R1381，並依序將"0" 轉成0填入R1382，將"1" 轉成1填入R1383

接著指定小數點位數為3，然後將"23.827" 轉成23827填入R1384，"159.447" 轉成159447填入R1385。



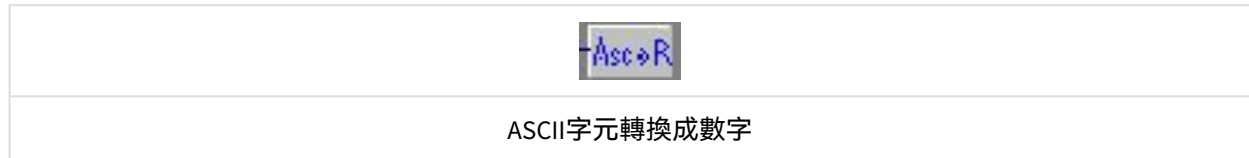
SYNTEC





ASCII字元轉換成數字

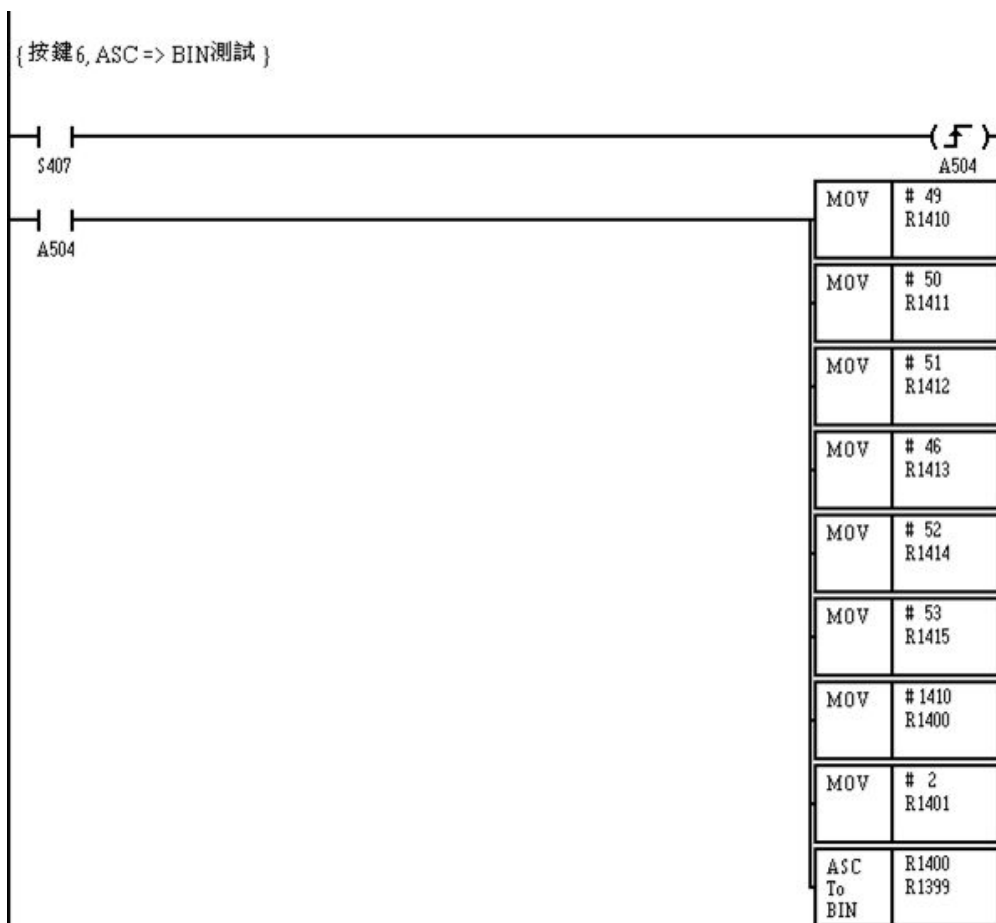
1. 圖示



2. 說明

- a. 在輸入訊號為On時，將指定的一連續區塊的暫存器內容值，以符合ASCII編碼的數值轉換成數字，並依照指定的小數點位數將數值偏移後填入目標暫存器，例如轉出來的數值為12.345，小數點為三位（放大1000倍），則目標暫存器會被填入12345。
- b. 此元件需要設定的連續暫存器定義如下：
 - 來源暫存器R1（基址+0）：存放ASCII要轉換成數值的暫存器起始位址。
 - 來源暫存器R1（基址+1）：設定小數點位數，會依小數點位數偏移（放大倍數）。
 - 來源暫存器R1（基址+2）：連續的暫存器中數值與數值分隔的符號（Splitter）。
 - 來源暫存器R1（基址+3）：保留區
 - 目標暫存器R2（基址+0）：轉換成數字後，要填入的暫存器起始位址。
- c. 範例
R1410開始存放的值為49（為字元'1'的ASCII值）、50（字元'2'的ASCII值）、51（字元'3'的ASCII值）、46（字元'.'的ASCII值）、52（字元'4'的ASCII值）、53（字元'5'的ASCII值）、0（字元'\0'的ASCII值），來源暫存器R1（基址為1400）設定存放的暫存器位址為1410，R1401存放要轉換的小數點位數2；元件執行後轉換成數值123.45，放大100倍後為12345存入目標暫存器R1399，並將來源暫存器R1設定的暫存器位址從1410改為1417，為轉換完成後的下一個位址。

SYNTEC



數字轉換成ASCII字元

1. 圖示


數字轉換成ASCII字元

2. 說明

- 在輸入訊號為On時，將來源暫存器內所存放的數值讀出，並依照指定的小數點位數偏移後所產生的數值，一個一個位元轉成ASCII值，依序從目標暫存器起始位址開始填入
例如來源暫存器讀出的值為12345，小數點為三位，則轉換成12.345，從指定目標暫存器的起始位址開始，會被填入ASCII '1', ASCII '2', ASCII '.', ASCII '3', ASCII '4', ASCII '5'
需注意此元件非正緣觸發元件
- 此元件需要設定的暫存器定義如下：
來源暫存器R1：存放要轉換的數值的暫存器位址。
目標暫存器R2：轉換成ASCII值後，要填入的暫存器起始位址。

目標暫存器R2 (基址+1)：設定小數點位數，會依小數點位數偏移 (縮小倍數)

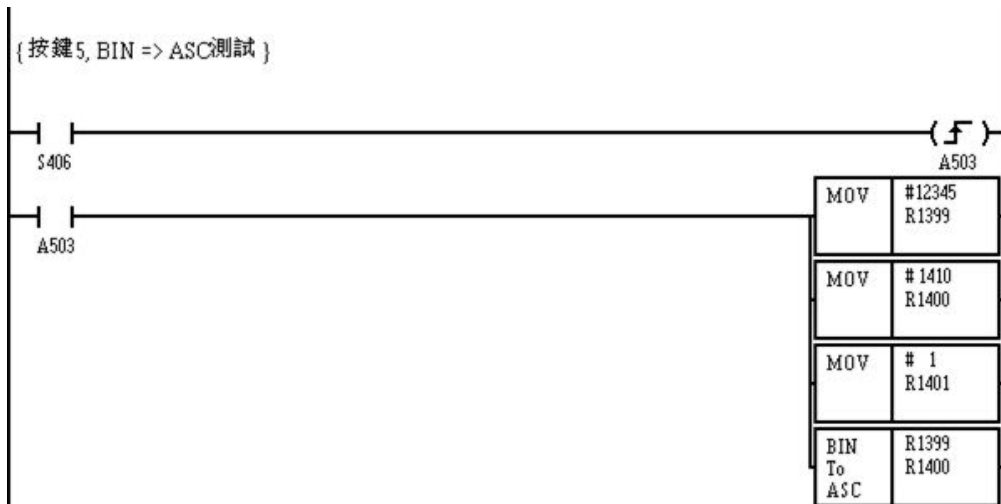
目標暫存器R2 (基址+2)：保留區

目標暫存器R2 (基址+3)：保留區

c. 範例

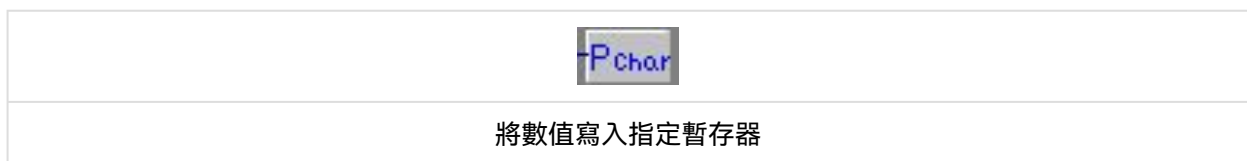
來源暫存器 (R1399) 設定為12345，目標暫存器 (R1400) 設定要存入ASCII值的暫存器起始位址R1410，R1401設定小數點位數1

元件執行後將12345縮小10倍為1234.5，並從R1410開始依序填入49 (字元'1'的ASCII值)、50 (字元'2'的ASCII值)、51 (字元'3'的ASCII值)、52 (字元'4'的ASCII值)、46 (字元'.'的ASCII值)、53 (字元'5'的ASCII值)、0 (字元'\0'的ASCII值)，並將R1400設定的暫存器起始位址從1410改為R1417，為轉換完成後的下一個位址。



數值寫入暫存器

1. 圖示



2. 說明

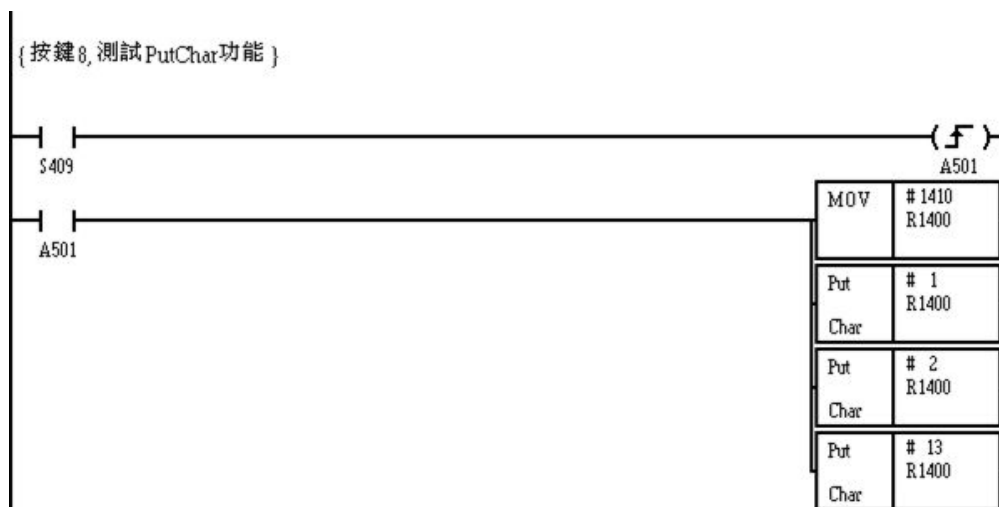
- 在輸入訊號On時，將輸入的數值寫到目標暫存器，寫入數值後，將目標暫存器所指向的目標暫存器位址往下移一個。
- 此元件非正緣觸發元件
- 此元件需要設定的暫存器定義如下：
 - 來源暫存器R1：存放要寫入的數值的暫存器位址
 - 目標暫存器R2：存放被寫入數值的暫存器位址
- 注意: 此元件目的是搬移字串，不支援數值為0，當填入數值為0會被忽略

3. 範例

來源暫存器為R1400，目標暫存器為R1410

按下按鍵8之後，先將被寫入數值的暫存器位址1410寫入R1400，然後依序將1、2、13填入R1400，這三個數字會被依序填入R1410、R1411、R1412

寫完之後R1400的值會指到R1413，即下一個可寫入的暫存器位址



從暫存器讀出數值

1. 圖示



2. 說明

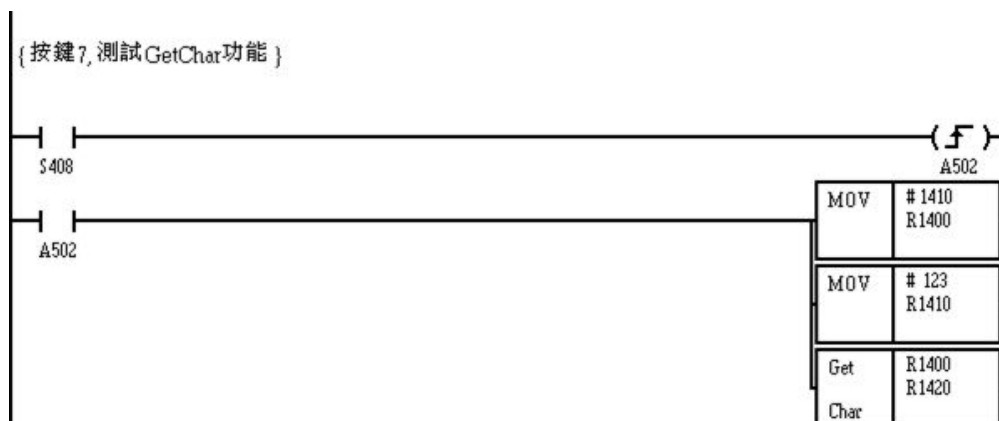
- 在輸入訊號On時，將來源暫存器中存放的暫存器位址內所存放的數值，讀出到目標暫存器，並將來源暫存器所指向的暫存器位址往下移一個。
- 此元件非正緣觸發元件
- 此元件需要設定的暫存器定義如下：
 - 來源暫存器R1：存放要讀取數值的暫存器位址
 - 目標暫存器R2：存放讀取出來的數值
- 注意：此元件目的是搬移字串，不支援數值為0，當填入數值為0會被忽略

3. 範例

將R1400所指向的暫存器位址內所存的數值讀出，並寫入R1420

按下按鍵7之後，先將要讀取的暫存器位址1410寫入R1400，然後將數值123填到R1410，此時123被讀出後會被填入R1420

讀完之後R1400的值會指到R1411，即下一個可讀取的暫存器位址。



Modbus讀取指令

1. 圖示



2. 說明

- 指定裝置可設定站號：0~255。
- MODR為MODBUS RTU16讀取元件，可讀取連續1~15個資料。
- MODBUS RTU是使用8位元格式，對應COM Port的位元數設定（Pr3942或Pr3962），需設定為8 bit。
- 系統會以R608.0/.10/.31來顯示Modbus的傳輸狀態（參考附錄4-a），傳送時若有錯誤產生，會將錯誤內容顯示於R5039（參考附錄4-c）。
- Modbus通訊埠，用來設定實體接線的通訊埠接口。若硬體為SUPER/10/20/200，COM Port為3（R5040=3）；若為eHMC/3/6/11/21，COM Port為2（R5040=2）。
- 設定為9為Modbus TCP，可透過網路（LAN）來傳輸資料，使用規格與Modbus RS485相同，差異之處如下：
 - 控制器Client（Master）與連接裝置Server（Slave）必須在同一個網域。
 - 連接裝置Server（Slave）的站號設定需與IP第4碼相同。例如IP是192.168.1.5，則必須把該裝置的站號設為5。
 - 最多可連接2個Server（Slave）裝置
 - 通訊逾時異常時間可於Pr3971設定
- 若有即時變動通訊協定的需求，需使用R5030（參考附錄4-b）。如果R5030未指定或R5030.0未啟用，需依照COM Port號碼設定標準通訊協定參數。
例如使用COM2，需設定Pr3941~Pr3950；COM3需設定Pr3961~Pr3970。
- 元件需要設定的內容如下：

讀取起始位址：設定外部裝置（Slave）要被讀取的資料位址
例如：讀取起始位址設定為2020，表示外部裝置要被讀取的資料位址要填入R2020，假設資料位址為0801₍₁₆₎，就要讓R2020=2049（通常外部裝置的位址會以16進位表示，請自行轉換成10進位，0801₍₁₆₎=2049₍₁₀₎）。

回傳起始位址：從外部裝置讀回控制器（Master）的資料儲存位址

例如：回傳起始位址設定為2021，而外部裝置讀回的值要填入到R60，就要讓R2021=60。

裝置位址：外部裝置（Slave）的站號

例如：裝置位址設定為1，表示外部裝置的站號為1（外部裝置有參數可設定其站號）。

讀取資料長度：從外部裝置（Slave）讀回的資料筆數

例如：讀取資料長度設定為2，表示傳回的資料筆數為2，此時會傳回讀取起始位址指定的位址資料（0801）+下一個位址的資料（0802），其它以此類推。

i. 在階梯圖中，所有位址皆以十進位填寫。

j. 其他細部說明請參考附錄

3. 範例：

a. 需求規格

使用eHMC控制器（使用COM2），與外部裝置的0801位址進行Modbus資料讀取通訊（站號=001、通訊位置=0801H），MODR元件的讀取起始位址為R2020（R2020=2049），回傳起始位址為R2021=60，表示資料儲存起始位址為R60。當IO導通時，讀取外部裝置0801位址的數值，如果正常收到訊號，就可以在指定的R值位址收到正確的數值；過程中如果發生傳輸錯誤，即立刻停止傳輸通訊，並發出警報，按下重置鍵後，可重新恢復通訊就緒狀態。

b. 狀態規劃

i. A400通訊就緒狀態：

- 開機上緣、Reset重置，或上一筆資料確認傳輸完成時，進入A400通訊就緒狀態。
- A400需保持觸發R608.0
- 資料傳輸完成的確認方法是透過R608得知，當R608.0 off與R608.10 on時，表示上一筆資料成功傳完（參考附錄4-a）。

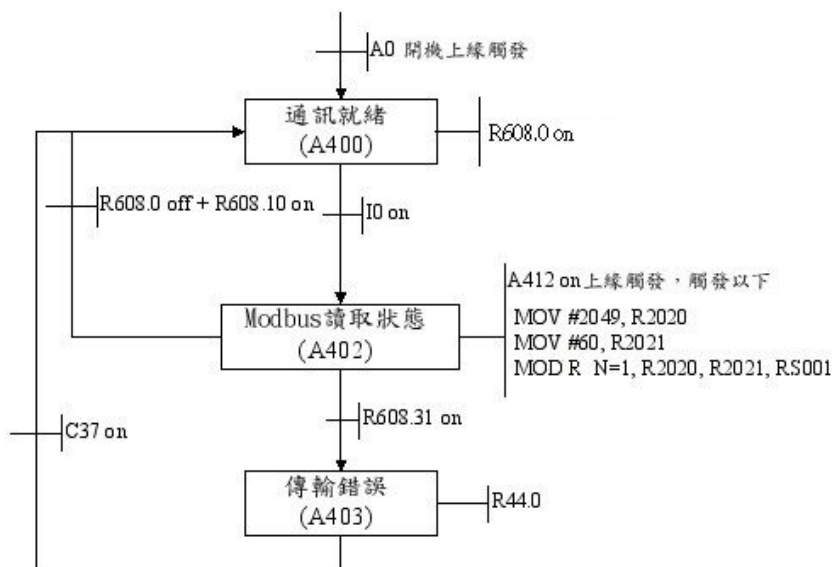
ii. A402 Modbus讀取狀態：

- 於A400通訊就緒狀態，當IO導通時，會進入A402讀取狀態。
- A402會以上緣觸發A412，單次啟動MODR元件，進行Modbus讀取動作。

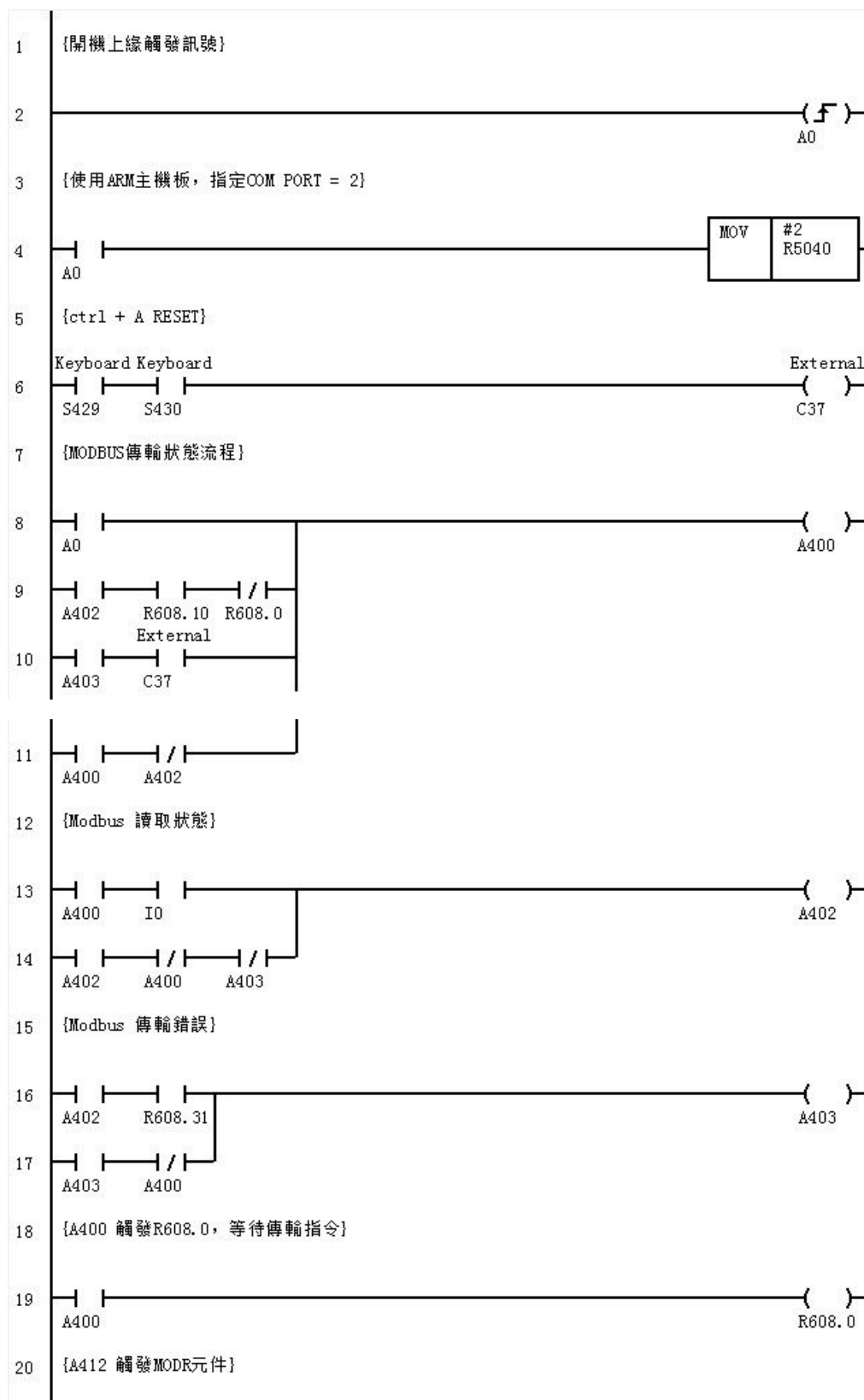
iii. A403傳輸錯誤狀態：

- 於A402 Modbus讀取狀態，若無法成功完成Modbus動作，控制器會自動觸發R608.31，表示傳輸交握錯誤，此時會進入A403傳輸錯誤狀態，並啟動警報R44.0，通知傳輸錯誤。（錯誤碼會顯示在R5039，參考附錄4-c）
- 於A403狀態，如果觸發Reset，可回到A400狀態。

c. 狀態流程圖



d. 階梯圖範例





e. IOCSAR規劃

編號	描述	備註
I0	觸發讀取動作的I點	
C37	重置鍵	核心定義
S429	鍵盤按鍵Ctrl	
S430	鍵盤按鍵A	
A0	開機上緣觸發	
A400	通訊 就緒狀態	
A402	Modbus讀取狀態	
A403	傳輸錯誤狀態	
A412	讀取動作觸發旗標	
R44.0	Modbus傳輸錯誤自訂警報	
R608.0	開始傳輸資料	核心定義
R608.10	傳輸完畢，且資料正確	核心定義
R608.31	傳輸錯誤	核心定義
R2020	外部裝置要被讀取的資料位址	

編號	描述	備註
R2021	從外部裝置讀回控制器的資料儲存位址	
R5040	Modbus通訊 的指定埠	

附錄一：Modbus通訊 相關R值說明

旗標	功能	讀寫規則
R608.0	待機狀態，表示目前可發送資料。 當R608.0 On時，R608.10、R608.31和R5039將自動清除。	W
R608.1	傳送中	R
R608.2	接收等待	R
R608.3	接收中	R
R608.10	接收完畢	R
R608.31	交握錯誤	R
R5030	通訊 協定設定	W
R5039	錯誤碼	R
R5040	設定Modbus通訊 用COM Port號碼	W

附錄二：R5030通訊協定設定說明

1. 當R5030的bit 0設定為0不啟用時，Modbus元件在進行通訊時，將參考Pr39xx之設定。
 - a. 例如當Modbus用COM2進行通訊時，會使用Pr3941~Pr3950，COM3則為Pr3961~Pr3970。

位元編號	功能說明	0 (Off)	1 (On)
b ₀	啟用通訊協定設定	不啟用	啟用
b ₁	資料長度	7 bit	8 ibt
b ₂ b ₃	同位元	無 (None) : 00 奇同位 (Odd) : 01 偶同位 (Even) : 11	
b ₄	停止位元	1 bit	2 bit

位元編號	功能說明	0 (Off)	1 (On)
b ₈ b ₇ b ₆ b ₅	Baud Rate/波特率	2400 bps: 0000 4800 bps: 0001 9600 bps: 0010 19200 bps: 0011 38400 bps: 0100 57600 bps: 0101 115200 bps: 0110 230400 bps: 0111 460800 bps: 1000 912600 bps: 1001	
b ₉ ~b ₃₁		系統保留區	

2. R5030設定範例:

Baud Rate"9600"/ 停止位元"2 bit"/ 同位元"None"/ 資料長度"8"/ 啟用"1"

b₈b₇b₆b₅ = 0010 / b₄ = 1 / b₃b₂ = 00 / b₁ = 1 / b₀ = 1

R5030=0010 1 00 1 1₍₂₎

二進位換算成十進位後，R5030=83₍₁₀₎

階梯圖範例:



附錄三：R5039錯誤碼說明

ID	R5039	說明
0x00	0	通訊 成功
0x01	1	通訊 指令不合法 (Function Code error)
0x02	2	通訊 地址不合法 (Address error)
0x03	3	資料數值被從站判定不合法 (Data Value error)
0x04	4	從站裝置錯誤 (Slave Device error)
0x05	5	確認應答 (ACK)，當從站判定需長時間處理，將回傳ACK避免逾時 (TimeOut) 產生
0x06	6	從站裝置忙碌中 (Slave Device Busy)
0x08	8	同位元確認錯誤 (Memory Parity error)
0x0A	10	閘道路徑無法使用 (Gateway Path Unavailable)
0x0B	11	目標閘道設備回應錯誤 (Gateway target Device failed to Respond)

0xF5	245	不支援的功能代碼
0xF6	246	Modbus封包格式錯誤
0xF7	247	R5040所指定的通訊 COM Port不正確
0xF8	248	裝置忙錄中，暫時停止處理通訊 命令
0xF9	249	資料確認錯誤 (CRC error)
0xFA	250	通訊 超過指定時間 (TimeOut error)
0xFB	251	讀取資料錯誤 (Read COM Port error)
0xFC	252	寫入資料錯誤 (Write COM Port error)
0xFD	253	開啟通訊 埠錯誤 (Open COM Port error)
0xFE	254	未知的錯誤 (Unkown error)

Modbus寫入指令

1. 圖示


寫入單筆資料至指定裝置的特定位址

2. 說明

- a. 指定裝置可設定站號：0~255。
- b. MODW為MODBUS RTU16寫入元件，僅可寫入1個資料。
- c. MODBUS RTU使用8位元格式，對應COM Port號的位元數設定（Pr3942或Pr3962），需設定為8 bit。
- d. 系統會以R608.0/.10/.31來顯示Modbus的傳輸狀態（參考Modbus讀取指令的附錄4-a），傳送時若有錯誤產生，會將錯誤內容顯示於R5039（參考Modbus讀取指令的附錄4-c）。
- e. Modbus通訊 埠，用來設定實體接線的通訊 埠接口。若硬體為SUPER/10/20/200，COM Port為3（R5040=3）；若為eHMC/3/6/11/21，COM Port為2（R5040=2）。
- f. 設定為9為Modbus TCP，可透過網路（LAN）來傳輸資料，使用規格與Modbus RS485相同，差異之處如下：
 - i. 控制器Client（Master）與連接裝置Server（Slave）必須在同一個網域。
 - ii. 連接裝置Server（Slave）的站號設定需與IP第4碼相同。例如IP是192.168.1.5，則必須把該裝置的站號設為5。
 - iii. 最多可連接2個Server（Slave）裝置
 - iv. 通訊 逾時異常時間可於Pr3971設定

g. 若有即時變動通訊協定的需求，需使用R5030（參考附錄4-b）。如果R5030未指定或R5030.0未啟用，需依照COM Port號碼設定標準通訊協定參數。

例如使用COM2需設定Pr3941~Pr3950；COM3需設定Pr3961~Pr3970。

h. 元件需要設定的內容如下：

寫入資料位址：設定外部裝置（Slave）要被寫入的資料位址

例如：寫入資料位址設定為2010，表示外部裝置要被寫入的位址要填入R2010，假設要被寫入的位址為0801₍₁₆₎，就要讓R2010=2049（通常外部裝置說明會以16進位表示，請自行轉換成10進位，0801₍₁₆₎=2049₍₁₀₎）。

寫入資料內容：寫入至外部裝置（Slave）的資料內容

例如：寫入資料內容設定為2011，表示寫入至外部裝置的資料內容要填入R2011，如果資料內容為70，就要讓R2011=70

裝置位址：外部裝置（Slave）的站號

例如：裝置位址設定為1，表示外部裝置的站號為1（外部裝置有參數可設定其站號）。

i. 在階梯圖中，所有位址皆以十進位填寫。

j. 其他細部說明請參考Modbus讀取指令的附錄

3. 範例

a. 需求規格

使用eHMC控制器（使用COM2），與外部裝置的0801位址進行Modbus資料寫入通訊（站號=001、通訊位置=0801H），MODW元件的寫入資料位址為R2010（R2010=2049），寫入資料內容為R2011（R2011=70）。當I0導通時，將預先存好的數值寫入外部裝置的0801位址，如果正常寫入，就可以在外部裝置上看到正確的數值；過程中如果發生傳輸錯誤，即立刻停止傳輸通訊，並發出警報，按下重置鍵後，可重新恢復通訊就緒狀態。

b. 狀態流程圖規劃

i. A400通訊就緒狀態：

- 開機上緣、Reset重置，或上一筆資料確認傳輸完成時，進入A400通訊就緒狀態。
- A400需保持觸發R608.0
- 資料傳輸完成的確認方法是透過R608得知，當R608.0 off與R608.10 on時，表示上一筆資料成功傳完。（參考Modbus讀取指令的附錄4-a）

ii. A401 Modbus寫入狀態：

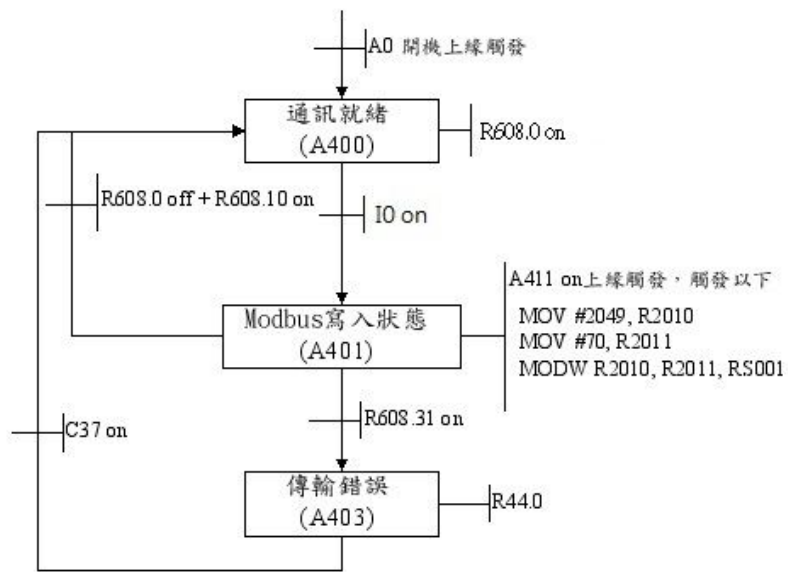
- 於A400通訊就緒狀態，當I0導通時，會進入A401寫入狀態。
- A401會以上緣觸發A411，單次啟動MODW元件進行Modbus寫入動作。

iii. A403傳輸錯誤狀態：

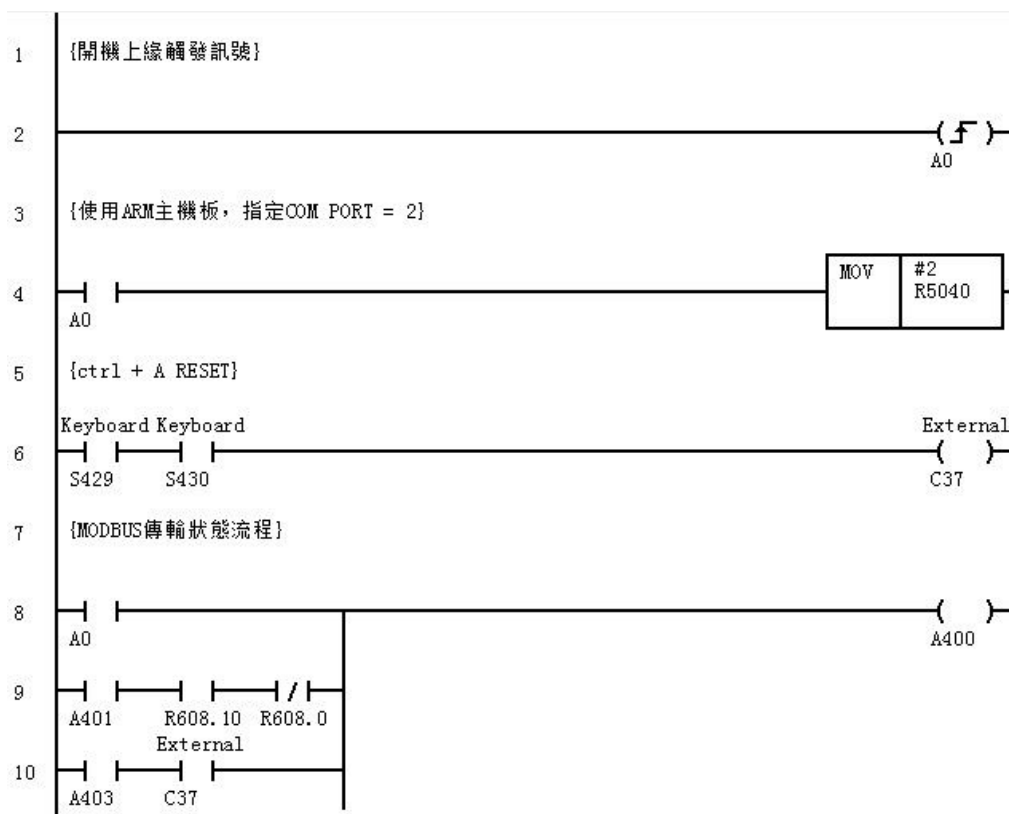
- 於A401 Modbus寫入狀態，若無法成功完成Modbus動作，控制器會自動觸發R608.31，表示傳輸交握錯誤，此時會進入到A403傳輸錯誤狀態，並且啟動警報R44.0，通知傳輸錯誤。（錯誤碼會顯示在R5039，參考Modbus讀取指令的附錄4-c）
- 於A403狀態，如果觸發Reset，可回到A400狀態。

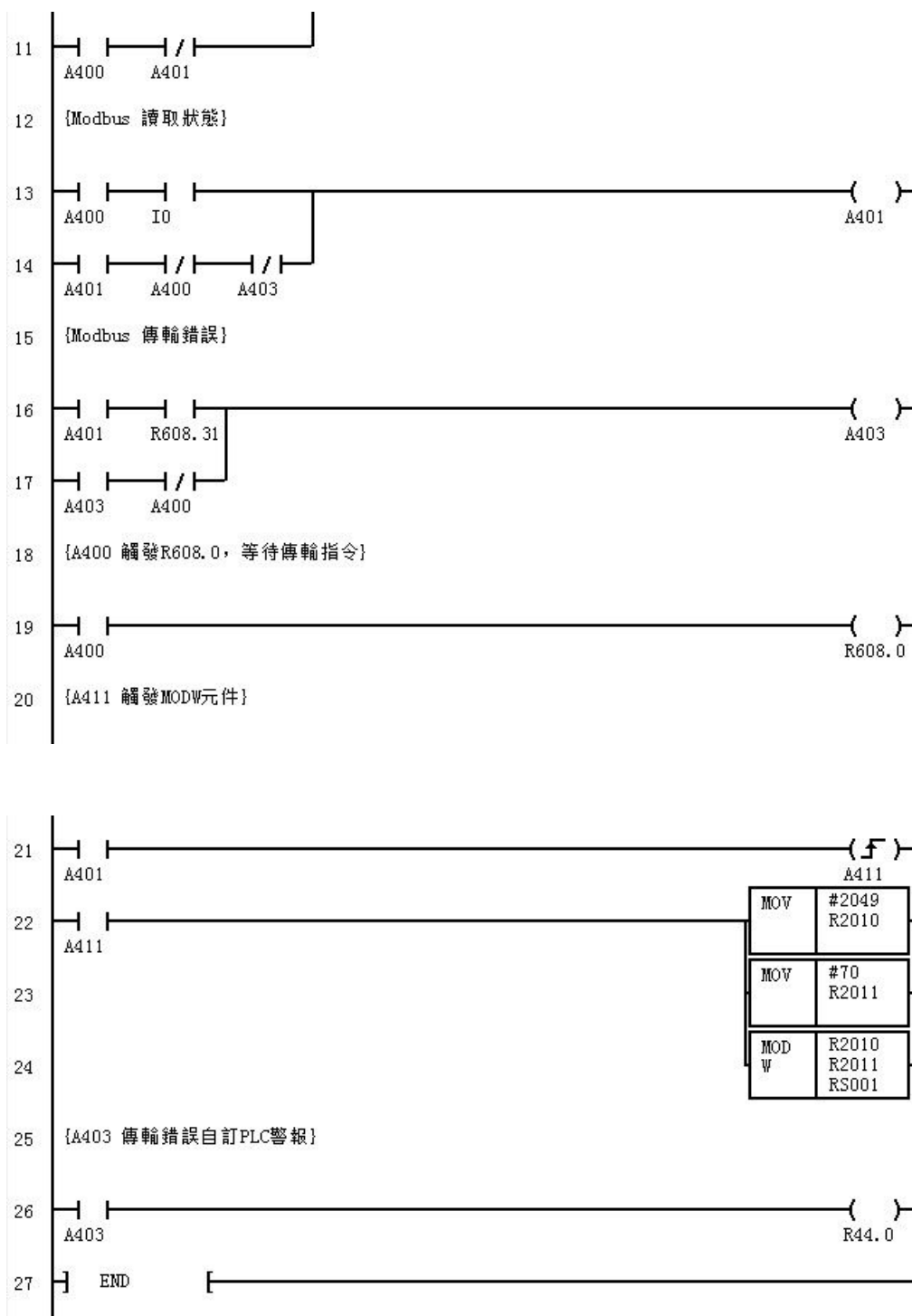
c. 狀態流程圖

SYNTEC



d. 階梯圖範例





e. IOCSAR規劃

編號	簡短描述	備註
10	觸發寫入動作的I點	

編號	簡短描述	備註
C37	重置鍵	核心定義
S429	鍵盤按鍵Ctrl	
S430	鍵盤按鍵A	
A0	開機上緣觸發	
A400	通訊 就緒狀態	
A401	Modbus寫入狀態	
A403	傳輸錯誤狀態	
A411	寫入動作觸發旗標	
R44.0	Modbus傳輸錯誤自訂警報	
R608.0	開始傳輸資料	核心定義
R608.10	傳輸完畢，且資料正確	核心定義
R608.31	傳輸錯誤	核心定義
R2010	外部裝置要被寫入的資料位址	
R2011	寫入至外部裝置的資料內容	
R5040	Modbus通訊 的指定埠	

自行設定Modbus指令

1. 圖示


在Modbus上送出指定的命令

2. 說明

- 自定義通訊 封包，使用上只需要撰寫Modbus的通訊 內容，CRC由此元件自行處理。
- 通訊 格式為RTU格式，使用8位元，對應COM Port的位元數設定（Pr3942或Pr3962），需設定為8 bit。
- 在程式中可無限次數使用ModRS指令，但不可同一時間執行2個以上的ModRS指令。
- ModRS指令無法於執行當中變更傳送資料的內容

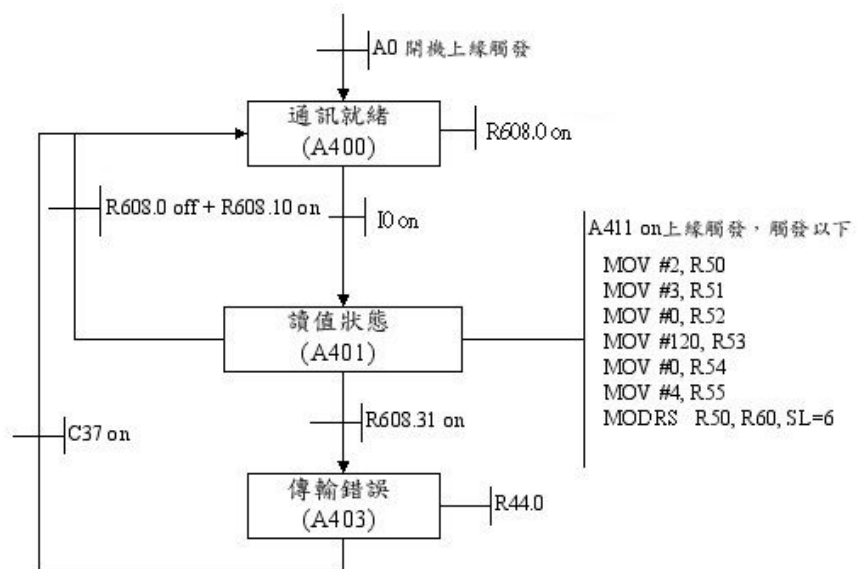
- e. 系統會以R608.0/10/31來顯示Modbus的傳輸狀態（參考Modbus讀取指令的附錄4-a），傳送時若有錯誤產生，會將錯誤內容顯示於R5039（參考Modbus讀取指令的附錄4-c）。
- f. Modbus通訊埠，用來設定實體接線的通訊埠接口。若硬體為SUPER/10/20/200，COM Port為3（R5040=3）；若為eHMC/3/6/11/21，COM Port為2（R5040=2）。
- g. 設定為9為Modbus TCP，可透過網路（LAN）來傳輸資料，使用規格與Modbus RS485相同，差異之處如下：
 - i. 控制器Client（Master）與連接裝置Server（Slave）必須在同一個網域。
 - ii. 連接裝置Server（Slave）的站號設定需與IP第4碼相同。例如IP是192.168.1.5，則必須把該裝置的站號設為5。
 - iii. 最多可連接2個Server（Slave）裝置
 - iv. 通訊逾時異常時間可於Pr3971設定
- h. 若有即時變動通訊協定的需求，需使用R5030（參考附錄4-b）。如果R5030未指定或R5030.0未啟用，需依照COM Port號碼設定標準通訊協定參數。
- i. 元件需要設定的內容如下：
 - 來源起始地址：**要送出資料的起始暫存器位址
例如：來源起始地址設定為50，表示要送出的資料從R50開始。
 - 回傳起始位址：**回傳值要存入的起始暫存器位址
例如：回傳起始位址設定為60，表示回傳值會從R60開始儲存。
 - 來源資料長度：**要發送的資料長度
例如：來源資料長度設定為6，表示要送出的資料總共6個，每一個資料都會以10進位的方式填入來源起始地址。若是來源起始地址設定為50，表示要送出的資料需依序填在R50、R51、R52、R53、R54、R55。
- j. 在階梯圖中，所有內容皆以十進位填寫。
- k. 其他細部說明請參考Modbus讀取指令的附錄

3. 範例1

- a. 需求規格
 - 使用eHMC控制器（使用COM2/作為Master），與另一台eHMC控制器（使用COM2/作為Slave）的R60與R61進行Modbus資料讀取通訊（站號=002、通訊起始位置=120）。
 - 當I0導通時，會將Master端預先存好的命令傳出，如果Slave端正常讀到訊號，會正確的將指定的暫存器數值傳回給Master，並依序存至Master端的R60開始的數個暫存器內（Master的MODRS元件設定：來源起始地址=50，表示要發送的資料從R50開始；回傳起始位址=60，表示收到的值會從R60開始儲存；來源資料長度=6，表示發送資料總共6個）。
 - 過程中如果發生傳輸錯誤，即立刻停止傳輸通訊，並發出警報，按下重置鍵後，可重新恢復通訊就緒狀態。
- b. Slave端控制器設定
 - i. 參數設定
 - Pr3234（啟用Modbus Slave COM Port）=2
 - Pr3235（Slave站號）=2
 - 對應COM Port號的位元數設定（Pr3942或Pr3962），需設定為8 bit
 - ii. 重開機後，確認Modbus Slave有正確開啟，如有正常運作，R5029會等於0。（Modbus Slave錯誤代碼R5029，參考附錄5-a說明）
- c. Master端來源資料內容規劃
 - 依照需求規格，參照附錄5-b. 多個R值讀取功能，指定R50=2、R51=3、R52=0、R53=120、R54=0、R55=4
- d. 狀態流程圖規劃
 - i. A400通訊就緒狀態：
 - 開機上緣、Reset重置，或上一筆資料確認傳輸完成時，進入A400通訊就緒狀態。
 - A400需保持觸發R608.0
 - 資料傳輸完成的確認方法是透過R608得知，當R608.0 off與R608.10 on時，表示上一筆資料成功傳完（參考Modbus讀取指令的附錄4-a）。
 - ii. A401讀值狀態：
 - 於A400通訊就緒狀態，當I0導通時，會進入A401讀值狀態。

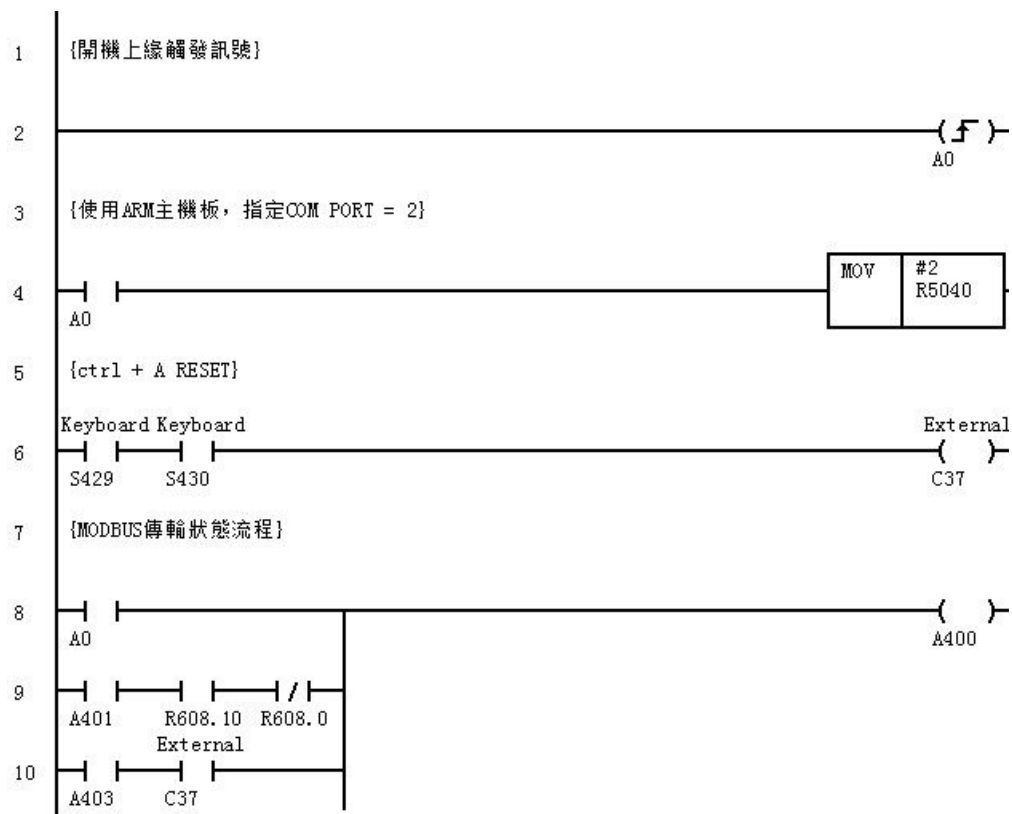
- A401會以上緣觸發A411，單次啟動MODRS元件進行Modbus讀值動作。
- iii. A403傳輸錯誤狀態：
- 於A401讀值狀態，若無法成功完成Modbus動作，控制器會自動觸發R608.31，表示傳輸交握錯誤，此時就會進入到A403傳輸錯誤狀態，並且啟動警報R44.0，通知傳輸錯誤。（錯誤碼會顯示在R5039，參考Modbus讀取指令的附錄4-c）
 - 於A403狀態，如果觸發Reset，可回到A400狀態。

e. 狀態流程圖

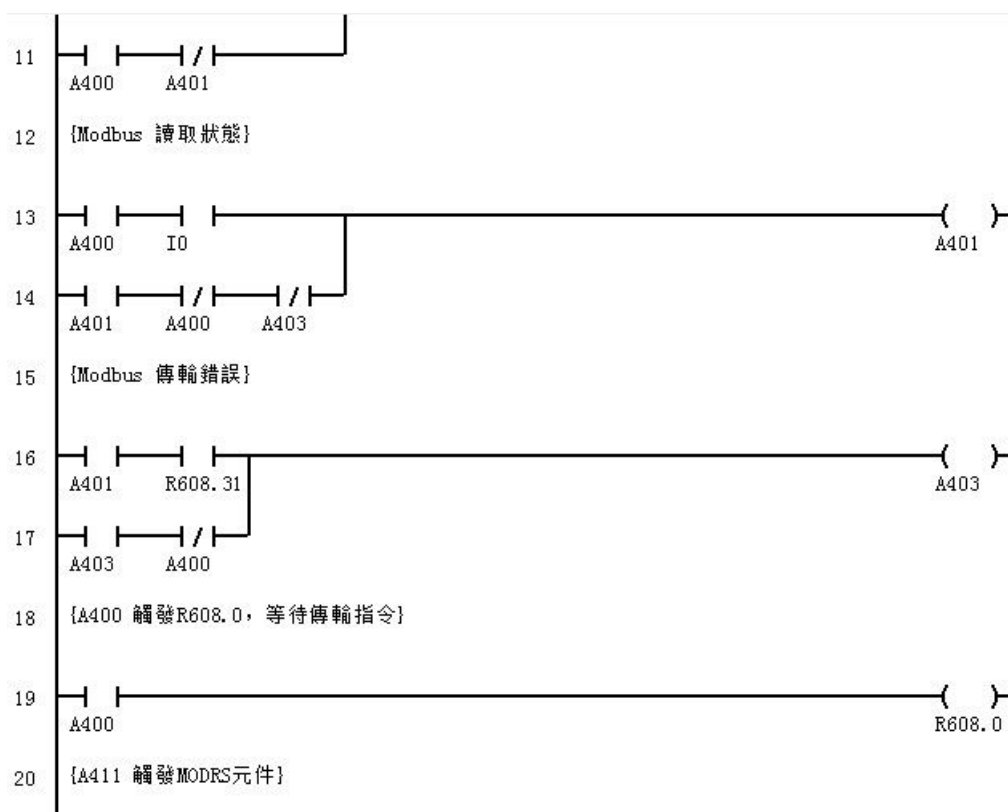


f. 階梯圖範例

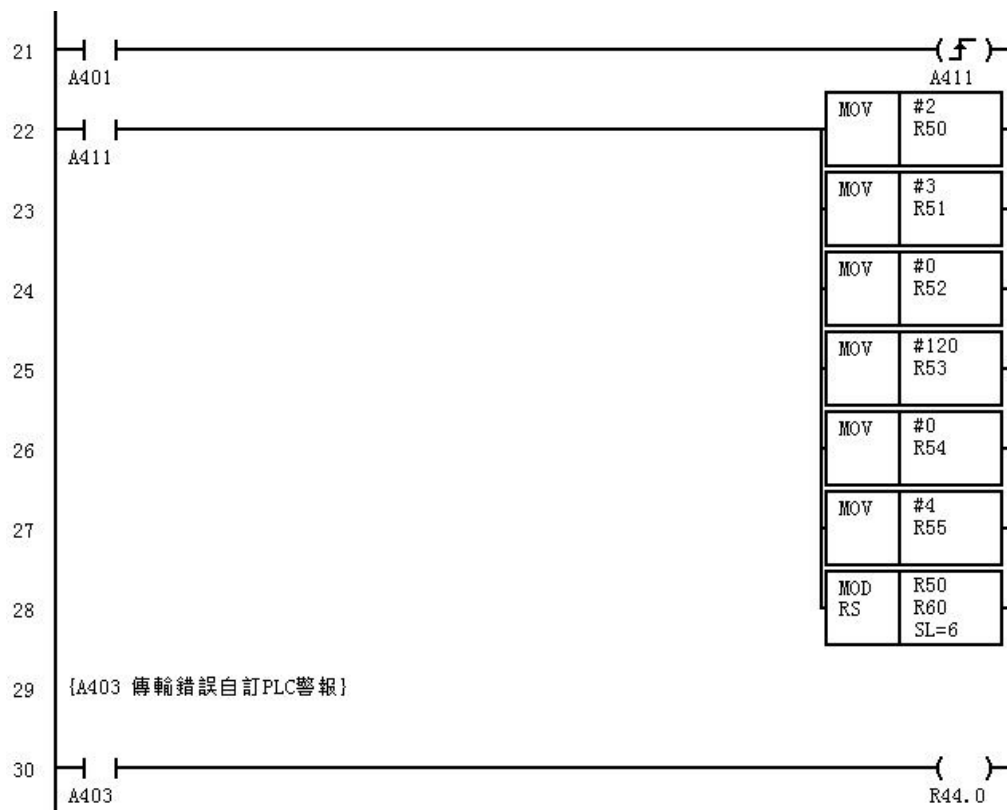
SYNTEC



SYNTEC



SYNTEC



g. IOCSAR規劃

編號	簡短描述	備註
I0	觸發Modbus動作的I點	
C37	重置鍵	核心定義
S429	鍵盤按鍵Ctrl	
S430	鍵盤按鍵A	
A0	開機上緣觸發	
A400	通訊 就緒狀態	
A401	讀值狀態	
A403	傳輸錯誤狀態	
A411	讀值動作觸發旗標	
R44.0	Modbus傳輸錯誤自訂警報	
R608.0	開始傳輸資料	核心定義

編號	簡短描述	備註
R608.10	傳輸完畢，且資料正確	核心定義
R608.31	傳輸錯誤	核心定義
R50	要發送資料的暫存器起始位址	
R60	回傳值要存入的暫存器起始位址	
R5040	Modbus通訊 的指定埠	

4. 範例2

a. 需求規格

使用eHMC控制器（使用COM2/作為Master），與另一台eHMC控制器（使用COM2/作為Slave）的R60與R61進行Modbus資料寫入通訊（站號=002、通訊 起始位置=120）。
當I0導通時，會將Master端預先存好的命令傳出，如果Slave端正常收到訊號，其R60與R61就會被寫入數值，並回傳正確的回傳值，依序儲存至Master端的R70開始的數個暫存器內。
（Master的MODRS元件設定：來源起始地址=50，表示要發送的資料從R50開始；回傳起始位址=70，表示收到的值會從R70開始儲存；來源資料長度=15，表示發送資料總共15個）
過程中如果發生傳輸錯誤，即立刻停止傳輸通訊，並發出警報，按下重置鍵後，可重新恢復通訊 就緒狀態。

b. Slave端控制器設定

i. 參數設定

Pr3234（啟用Modbus Slave COM Port）=2

Pr3235（Slave站號）=2

對應COM Port號的位元數設定（Pr3942或Pr3962），需設定為8 bit

ii. 重開機後，確認Modbus Slave有正確開啟，如有正常運作，R5029會等於0。（Modbus Slave錯誤代碼R5029，參考附錄5-a說明）

c. Master端來源資料內容規劃

依照需求規格，參照附錄5-c.多個R值寫入功能，指定R50=2、R51=16、R52=0、R53=120、R54=0、R55=4、R56=8、R57=0、R58=1、R59=0、R60=1、R61=7、R62=91、R63=205、R64=21

d. 狀態流程圖規劃

i. A400通訊 就緒狀態：

- 開機上緣、Reset重置，或上一筆資料確認傳輸完成時，進入A400通訊 就緒狀態。
- A400需保持觸發R608.0
- 資料傳輸完成的確認方法是透過R608得知，當R608.0 off與R608.10 on時，表示上一筆資料成功傳完（參考Modbus讀取指令的附錄4-a）。

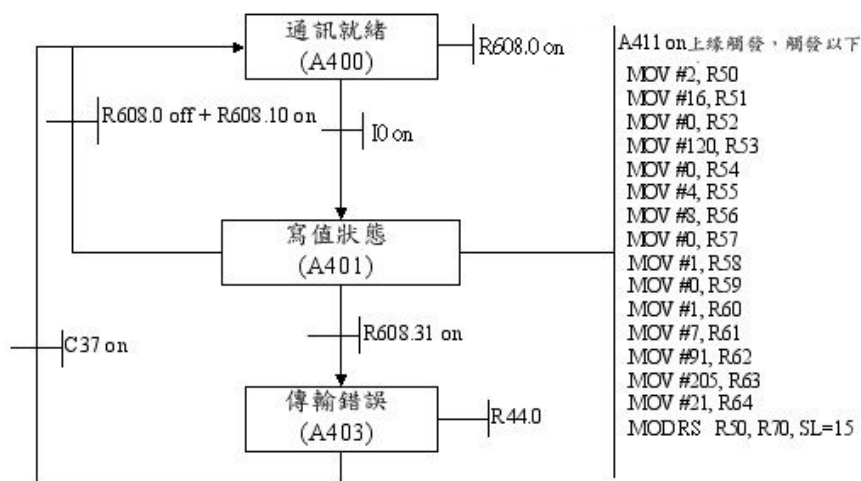
ii. A401寫值狀態：

- 於A400通訊 就緒狀態，當I0導通時，會進入A401寫值狀態。
- A401會以上緣觸發A411，單次啟動MODRS元件進行Modbus寫值動作。

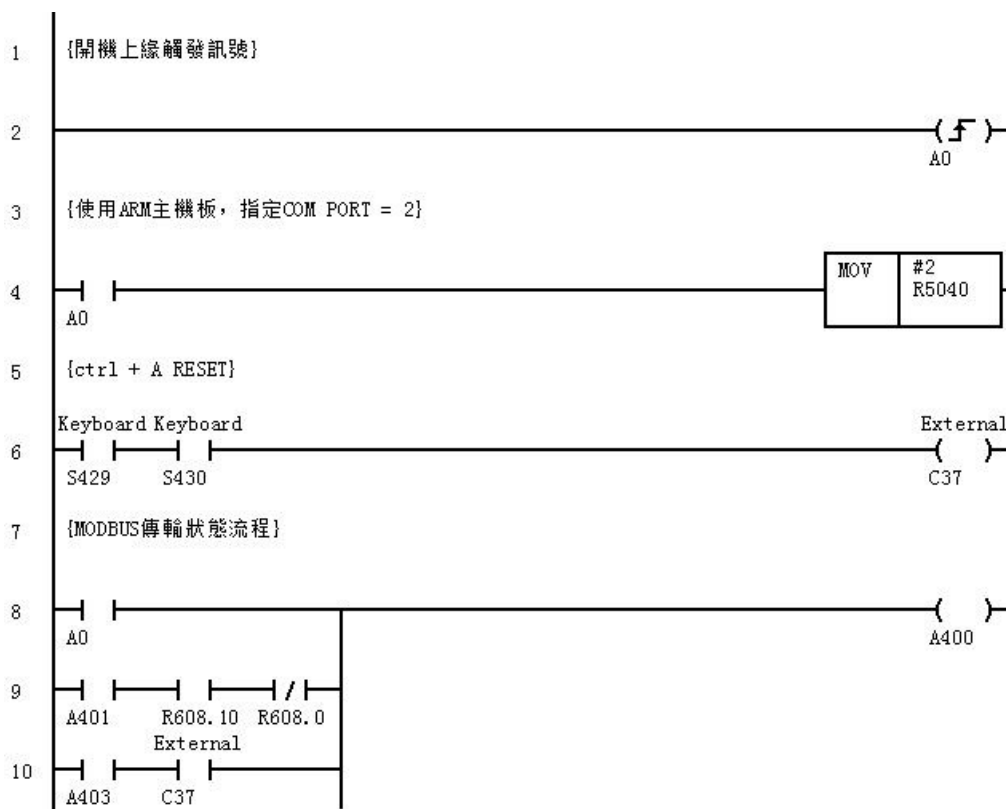
iii. A403傳輸錯誤狀態：

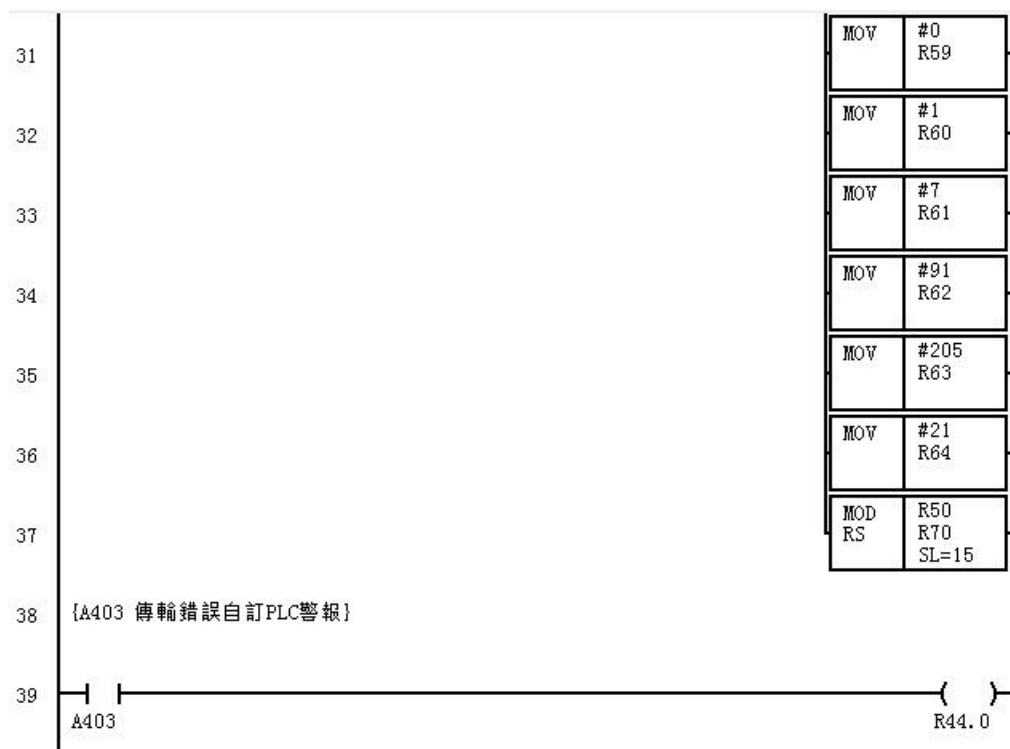
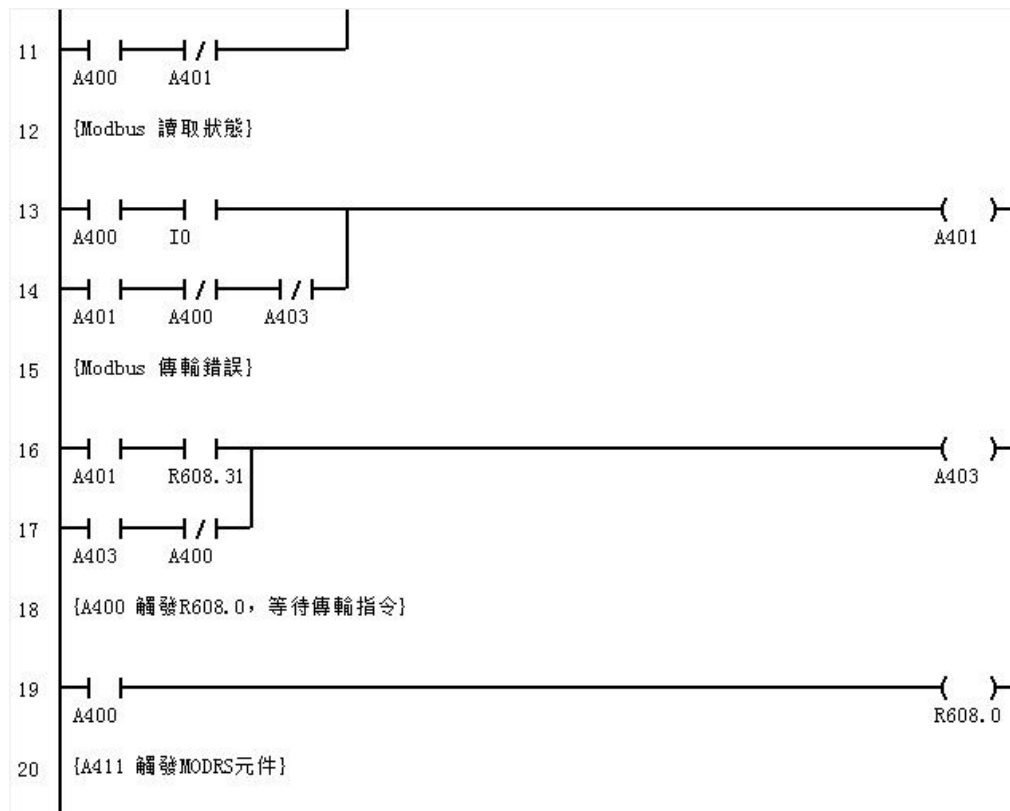
- 於A401讀值狀態，若無法成功完成Modbus動作，控制器會自動觸發R608.31 on，表示傳輸交握錯誤，此時就會進入到A403傳輸錯誤狀態，並且啟動警報R44.0，通知傳輸錯誤。（錯誤碼會顯示在R5039，參考Modbus讀取指令的附錄4-c）
- 於A403狀態，如果觸發Reset，可回到A400狀態。

e. 狀態流程圖



f. 階梯圖範例





g. IOCSAR規劃

編號	簡短描述	備註
I0	觸發Modbus動作的I點	
C37	重置鍵	核心定義
S429	鍵盤按鍵Ctrl	
S430	鍵盤按鍵A	
A0	開機上緣觸發	
A400	通訊 就緒狀態	
A401	寫值狀態	
A403	傳輸錯誤狀態	
A411	寫值動作觸發旗標	
R44.0	Modbus傳輸錯誤自訂警報	
R608.0	開始傳輸資料	核心定義
R608.10	傳輸完畢，且資料正確	核心定義
R608.31	傳輸錯誤	核心定義
R50	要發送資料的暫存器起始位址	
R70	回傳值要存入的暫存器起始位址	
R5040	Modbus通訊 的指定埠	

5. 附錄：Modbus Slave通訊，相關R值說明（R5029）

錯誤代碼	R5029	說明
0x00	0	啟動Slave成功
0x01	1	開啟COM Port 失敗
0x02	2	不合法的COM Port 號碼（1~3）
0x03	3	不合法的站號（0~255）
0x04	4	封包長度不正確
0x05	5	封包內容不正確

錯誤代碼	R5029	說明
0x06	6	Master傳送的封包，其CRC錯誤
0x07	7	未收到加工檔上傳中止命令錯誤
0x08	8	讀取NC檔案錯誤
0x09	9	回傳失敗

6. 多個R值讀取功能

6.a 傳出：

- 傳出資料格式與範例：站號 (1) + 功能代碼 (1) + 讀取起始位址 (2) + 讀取資料數量 (2)
 - P.S. () 內的數字表示資料筆數或資料占用的Byte數
- 範例：Slave的站號2，若想讀取Slave的兩個暫存器，分別為R60與R61，Master端的來源起始地址從R50開始，R50~R55，共6筆

站號 (1)	功能代碼 (1)	讀取起始位址 (2)	讀取資料數量 (2)
2	3	0、120	0、4
R50=2	R51=3	R52=0 R53=120	R54=0 R55=4

- 資料說明

資料說明	資料筆數 (占用的Byte)	指令值/範圍	說明
站號	1 Byte	0~255	
功能代碼	1 Byte	3	此功能為MODBUS標準功能代碼0x03
讀取起始位址*	2 Bytes	暫存器：R0~R30000 對應讀取起始位址：0~60000 (十進位)	讀取起始位址= 暫存器編號*2 例如： 要讀取的起始位址為R60，對應到讀取起始位址=60*2=120 依照格式填入0、120
讀取資料數量(N)	2 Bytes	1~125	一個控制器暫存器需使用2個讀取資料量 例如： 要讀取R60、R61就需要使用4個，依照格式填入0、4。

資料說明	資料筆數 (占用的Byte)	指令值/範圍	說明
讀取起始位址計算方法* (1) 先計算讀取起始位址 (2) 判斷是否大於255，如果小於或等於255的話，則在第一個Byte直接設定0，起始位址填入第二個Byte。 (3) 如果大於255，需先換算成16進位，再分拆成前兩位與後兩位的數值。 (4) 將前兩位與後兩位數值（16進位），換算成10進位，計算後再填入使用。 範例1: R60 - 讀取起始位址=60 x 2=120₍₁₀₎ 120小於255，所以第一及第二個Byte分別為0、120 範例2: R6000 - 讀取起始位址=6000 x 2=12000₍₁₀₎ 12000大於255，需換算 12000₍₁₀₎=2EE0₍₁₆₎，分拆成2E₍₁₆₎、E0₍₁₆₎ - 將前兩位與後兩位換算成10進位，所以第一及第二個Byte分別為46、224			

6.b 回傳:

- 回傳資料格式與範例: 站號 (1) +功能代碼 (1) +位元數 (1) +資料值 (2 x N)
 - N=讀取資料數量
 - P.S. () 內的數字表示資料筆數或資料占用的Byte數
- 範例: Slave的站號2，讀取兩個Slave的暫存器，分別為R60=65537、R61=123456789。回傳資料放置到Master的回傳起始位址從R60開始，R60~R70，共11筆

站號 (1)	功能代碼 (1)	位元數 (1)	資料值 (2xN)
2	3	8	0、1、0、1、7、91、205、21
R60=2	R61=3	R62=8	R63=0 R64=1 R65=0 R66=1 R67=7 R68=91 R69=205 R70=21

- 資料說明

資料說明	資料筆數 (占用的Byte)	指令值/範圍	說明
站號	1 Byte	0~255	
功能代碼	1 Byte	3	
位元數	1 Byte	2N	N=要讀取資料數量 例如: 傳出時，設定要讀取資料數量為4，位元數就會是4 x 2=8

資料說明	資料筆數 (占用的Byte)	指令值/範圍	說明
資料值*	2N Bytes	0~255	根據位元數判斷會有幾筆資料值，每一筆資料在控制器讀取會自動換成十進位
資料值的計算方法* (1) 控制器的一個暫存器會拆成四筆資料值傳出，先得到的數值為高位，後得到的為低位 (2) 例如得到的數值為a、b、c、d，需由左至右排列運算，$a*16^6+b*16^4+c*16^2+d*16^0$ 範例1: R值=65537，得到的數值為0、1、0、1 1. 回傳的值依序為0、1、0、1 2. 套用公式計算$0*16^6+1*16^4+0*16^2+1*16^0=65537$ 範例2: R值=123456789，得到的數值為7、91、205、21 1. 回傳的值依序為7、91、205、21 2. 套用公式計算$7*16^6+91*16^4+205*16^2+21*16^0=123456789$			

7. 多個R值寫入功能

7.a 傳出:

- 傳出資料格式與範例：站號 (1) +功能代碼 (1) +寫入起始位址 (2) +寫入資料數量 (2) +寫入資料位元數 (1) +寫入資料值 (2N)
 - N=寫入資料數量
 - P.S. () 內的數字表示資料筆數或資料占用的Byte數
- 範例：Slave的站號2，需寫入Slave的兩個暫存器，寫入值分別為R60=65537與R61=123456789，Master端的來源起始位址從R50開始，R50~R64，共15筆

站號 (1)	功能代碼 (1)	寫入起始位址 (2)	寫入資料數量 (2)	寫入資料位元數 (1)	寫入資料值 (2N)
2	16	0、120	0、4	8	0、1、0、1、7、91、205、21
R50=2	R51=16	R52=0 R53=120	R54=0 R55=4	R56=8	R57=0、R58=1 R59=0、R60=1 R61=7、R62=91 R63=205、R64=21

- 資料說明

資料說明	資料筆數 (占用的Byte)	指令值/範圍	說明
站號	1 Byte	0~255	
功能代碼	1 Byte	16	
寫入起始位址*	2 Bytes	暫存器: R0~R30000 對應寫入起始位址: 0~60000 (10進位)	寫入起始位址= 暫存器編號*2 例如: 要寫入的控制器起始位址為R60，對應到寫入起始位址=60*2=120 依照格式填入值為0、120

資料說明	資料筆數 (占用的Byte)	指令值/範圍	說明
寫入資料數量	2 Bytes	2~122	要寫入控制器的一個暫存器需使用2個讀取資料量。 例如： 要寫值到R60、R61，需要使用4個資料數量，依照格式填入值為0、4。
寫入資料位元數	1 Byte	2N	N=要寫入資料數量 例如： 要寫入的資料數量如果設定4，位元數得到的值=4 x 2=8
寫入資料值**	2N Bytes	0~255	一個控制器暫存器需使用4個資料值。 例如： 要寫入R60、R61需要使用8個資料值。



SYNTEC

資料說明	資料筆數 (占用的Byte)	指令值/範圍	說明
寫入起始位址計算方法* (1) 先計算寫入起始位址 (2) 判斷是否大於255，如果小於或等於255的話，則在第一個Byte直接設定0，起始位址填入第二個Byte。 (3) 如果大於255，需先換算成16進位值，再分拆成前兩位與後兩位的數值。 (4) 將前兩位與後兩位數值（16進位），換算成10進位，計算後再填入使用。 範例1: R60 1. 寫入起始位址=60 x 2=120₍₁₀₎ 2. 120小於255，所以第一及第二個Byte分別為0、120 範例2: R6000 1. 寫入起始位址=6000 x 2=12000₍₁₀₎ 2. 12000大於255，需換算 3. 12000₍₁₀₎=2EE0₍₁₆₎，分拆成2E₍₁₆₎、E0₍₁₆₎ 4. 將前兩位與後兩位換算成10進位，所以第一及第二個Byte分別為46、224 寫入資料值計算方法** (1) 控制器的一個暫存器要拆成四個資料值寫入，先寫入的數值為高位，後寫入的為低位 (2) 例如欲寫入的數值為x，需拆成4個資料值a、b、c、d， $x/16^6 = a \dots a1$ $a1/16^4 = b \dots b1$ $b1/16^2 = c \dots c1$ $c1 = d$ a為最高位，d為最低位，依序排列a、b、c、d，作為寫入值 範例1: 要寫入R值=65537，得到的數值為0、1、0、1 1. 套用公式計算 $65537/16^6 = 0 \dots 65537 \quad (a=0)$ $65537/16^4 = 1 \dots 1 \quad (b=1)$ $1/16^2 = 0 \dots 1 \quad (c=0)$ $1 = d$ a=0、b=1、c=0、d=1 範例2: 要寫入R值=123456789，得到的數值為7、91、205、21 1. 套用公式計算 $123456789/16^6 = 7 \dots 6016277 \quad (a=7)$ $10668473/16^4 = 91 \dots 52501 \quad (b=91)$ $52501/16^2 = 205 \dots 21 \quad (c=205)$ $21 = d$ a=7、b=91、c=205、d=21			

7.b 回傳:

- 回傳資料格式與範例: 站號 (1) +功能代碼 (1) +寫入起始位址 (2) +寫入資料數量 (2)
 - P.S. () 內的數字表示資料筆數或資料占用的Byte數
- 範例: Slave的站號2，需寫入Slave的兩個暫存器，寫入值分別為R60=65537與R61=123456789，回傳資料放置到Master的回傳起始位址從R70開始，R70~R75，共6筆

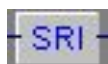
站號 (1)	功能代碼 (1)	寫入起始位址 (2)	寫入資料數量 (2)
2	16	0、120	0、4
R70=2	R71=16	R72=0 R73=120	R74=0 R75=4

- 資料說明

資料說明	資料筆數 (占用的Byte)	指令值/範圍	說明
站號	1 Byte	0~255	回傳值與傳出值相同，提供確認用
功能代碼	1 Byte	16	
寫入起始位址	2 Bytes	暫存器：R0~R30000 對應寫入起始位址：0~60000 (10進位)	
寫入資料數量	2 Bytes	2~122	

SRI自訂Modbus指令

1. 圖示



2. 說明

- 透過SRI 接口，發出自定義Modbus指令
- SRI 的參數須設定為**軟體模式**，此功能才可正確執行。例如

SYNTEC



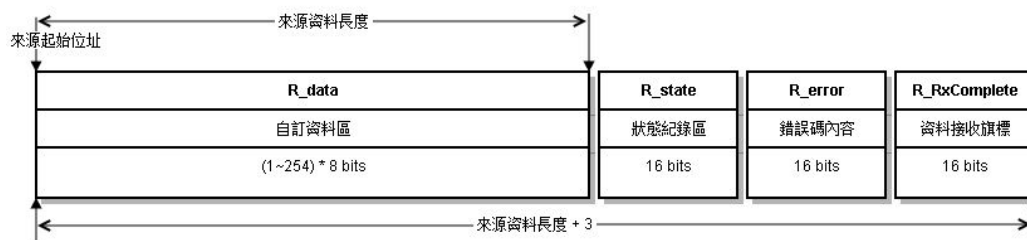
A configuration dialog box titled "COM3". It contains several input fields and dropdown menus. The first field is a text box containing "0". Below it is a dropdown menu with "啟用" (Enable) selected and highlighted in yellow. The next dropdown menu shows "RS485". Below that is a text box containing "115200". The next dropdown menu shows "軟體" (Software). Below that are two text boxes containing "10" and "5" respectively. At the bottom is a button labeled "設定" (Set).

- c. 控制器的SRI 接口只支援控制器作為Master，若需要作為Slave，請參考Modbus系列元件的說明。
- d. 自定義通訊 封包，使用上只需要撰寫Modbus的通訊 內容，CRC由元件自行處理。
- e. 通訊 格式為RTU格式，使用8位元格式。



A dialog box titled "元件參數設定" (Element Parameter Setting). It contains four rows of labels and text boxes: "來源起始地址" (Source Start Address) with "0000", "回傳起始位址" (Return Start Address) with "0000", "來源資料長度" (Source Data Length) with "0", and "裝置位址" (Device Address) with "0". To the right of these fields are two buttons: "確認" (Confirm) and "取消" (Cancel).

- f. 來源起始位址：設定傳輸封包的起始暫存器，每個暫存器對應8位元。注意最後須預留3個暫存器，分別為**Rstate**、**Error** 與**RrxComplete**



- g. 回傳起始位址：設定回傳封包的起始暫存器，每個暫存器對應8位元，注意須根據Modbus的封包回傳預留所需的暫存器。
- h. 來源資料長度：根據Modbus的封包格式，扣除CRC之後的長度，因此最大為254 Bytes。
- i. 裝置位址：對應SRI的COM Port。
- j. 在程式中可無限次數使用SRI指令，但不可同一時間執行2個以上的SRI指令。
- k. SRI指令無法於執行當中變更傳送資料的內容。
- l. 系統會於**R_{state}**顯示通訊的狀態，詳細說明如下：

數值	功能說明
0 (0x00)	發送待機，表示目前可發送資料
1 (0x01)	傳送中
2 (0x02)	接收等待
3 (0x03)	接收中
16 (0x10)	通訊失敗，詳細錯誤碼為 R_{error} 值

- 當通訊失敗時，指定資料來源位址倒數第二位暫存器「**R_{error}**」將記錄對應之錯誤碼。（可參考R5039的說明）
- 接收回傳值成功後，最末位暫存器「**R_{RxComplete}**」將為1，其餘時候為0。
- 發送指令成功，將會初始化**R_{error}**與**R_{RxComplete}**所記錄之狀態。

3. 範例

- a. 請參考ModRS的範例中Master的部分。
- b. 封包格式、設定方法都一樣，差異的部分列出如下（傳輸狀態、錯誤碼、完成旗標）：

		Mod系列元件	SRI元件
開始傳送資料		R608.0 on	SRI元件觸發且 R_{state} = 0即開始傳送
傳輸狀態	傳送待機		R_{state} = 0
	傳送中	R608.1 on	R_{state} = 1
	接收等待	R608.2 on	R_{state} = 2
	接收中	R608.3 on	R_{state} = 3
	傳輸錯誤	R608.31 on	R_{state} = 16

	Mod系列元件	SRI元件
完成旗標（傳輸完畢且資料正確）	R608.10 on	R_{RxComplete} =1
錯誤碼	R5039	R_{error}

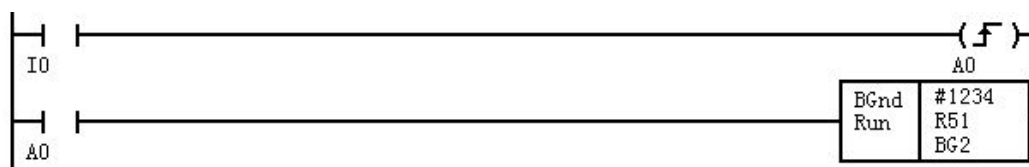
2.2.12 背景運算元件

1. 圖示



2. 說明

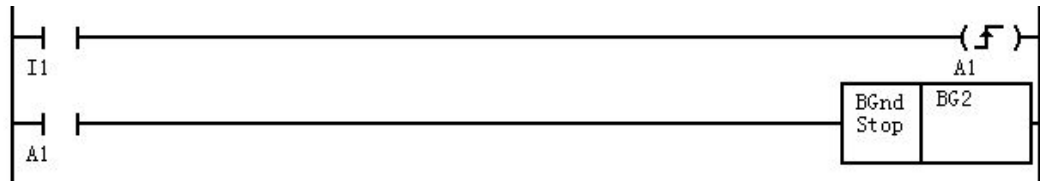
- a. BGnd Run是在階梯圖執行指定的巨集程式，當元件被觸發時，控制器會執行巨集內容。所用的引數說明如下：



元件參數設定

檔名編號	1234	確認 取消
行號暫存器	51	
元件編號	2	

- 檔名編號：輸入巨集之檔名編號（00001~09999）
 - 行號暫存器（Rn）：記錄目前巨集正在執行之行號，請輸入暫存器編號（R1~R65535）。Rn+1為警報提示。
 - 例如：行號暫存器設定51，當警報發生時，R52會從0變成1，直到使用BGnd Stop清除後，R52變回0。建議使用此旗標來確定巨集是否執行完成。
 - 元件編號：設定背景運算元件編號，目前支援BG00~BG19共20個。
- b. BGnd Stop是用來重置執行中的巨集或警報



i. 裝置編號：輸入想要重置的元件編號（BG00~BG19）。

3. 使用限制

- PLC編輯器有效版本始於2.25.0，且需配合控制器10.116.22以後版本。
- 提供**20**個背景運算裝置（BG00~BG19）。
- 欲執行的巨集程式名稱**一定以“O”作檔名開頭**，但是階梯圖內不需寫“O”；程式名稱（nnnn）一定是**四碼**的自然數。
- 巨集放置路徑：DiskC\OpenCnc\Ncfiles或DiskC\OpenCnc\MACRO
- 巨集內容限制：
 - G碼可支援**
 - G04等待指令
 - G10 L1000，暫存器填值指令
 - G10 L1801，清除所有 G10 L1800 指定的O-bit輸出之指令
 - G10L1810-設定訊號條件
 - G10L1820-等待訊號條件成立
 - 其餘G碼指令皆不支援
 - 函數表可支援**
 - 數學運算相關指令(例如ABS、SIN、COS、MAX 等)
 - SETDO指令
 - 其餘函數皆不支援
 - 不支援M、S、T、B碼指令**
 - 不支援增量模式**
 - 僅支援讀寫區域變數#1~#400，不接受其他區域變數。**
 - 可使用公用變數@值，規格同一般巨集。**
- 相關警報請參閱警報手冊BGND類別。
 - 警報發生時，會顯示引發警報之裝置編號。
 - 當使用過多BGND元件時，可能造成剩餘記憶體不足，將會發出警報。
- 此元件僅支援放置於一般掃描區，即使將元件移至快速掃描區，其掃描速度仍不會改變。
- 請使用**上緣觸發**執行本元件，以免相同動作被多次執行。
 - 當元件已被觸發並開始執行，在未完成前被二次觸發，第二次觸發指令將被忽略。
- 系統重置與急停無法停止BGND元件，僅可使用BGnd Stop使元件停止或重置。

k. 當 PLC 內有多個背景運算元件時，元件編號不可重複，否則會導致執行的動作異常。

2.2.13 暫存器間接定值功能

某些可以套用暫存器的特定元件，在輸入暫存器編號時若加入@符號，即可啟用間接定值功能。
如下圖範例所示，R61內容為50，當掃描到第4行時，會將R50內容指定給R62，故最終R62內容為1。



SYNTEC

3 階梯圖編寫開發

3.1 M/S/T碼

3.1.1 M/S/T碼功能目的

加工程式中一般會編入M碼以驅動周邊設備（油冷機、料機等）；編入S碼以設定主軸轉速；編入T碼以進行換刀作業。

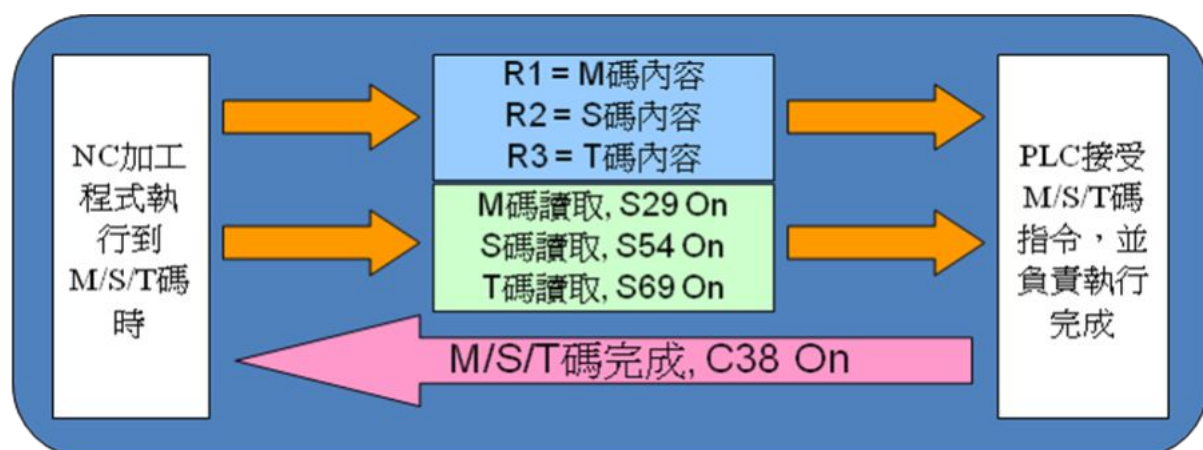
透過M/S/T碼的使用，在加工程式內也能透過階梯圖對外部設備做同步控制。

加工程式	動作說明
N1 G00 X50.0 Y100.0	N1 X軸/Y軸以G00移動到50./100.mm
N2 T1	N2 換1號刀
N3 S1000 M3	N3 主軸以1000 RPM正轉
N4 G01 Z-10.0 F3000	N4 Z軸以G01移動到-10.mm
N5 M8	N5 打開切削液
N6 G01 Z-15. F500	N6 Z軸以G01移動到-15.mm

上面範例中，程式第二行透過T1進行刀具更換；程式第三行透過S1000來設定主軸轉速為1000 RPM，並以M3啟動主軸馬達正轉；程式第五行透過M8來開啟切削液。

至於控制器是如何處理M/S/T碼並正確驅動周邊設備，將於接下來的章節說明其處理架構。

3.1.2 M/S/T碼動作流程



- 當控制器核心執行到加工程式內的M/S/T碼時，會將M/S/T碼內容放在R1/R2/R3暫存器內。
 - 例如當執行到M3時，核心會將3存入R1，也就是說R1=3；當執行到T5時，核心會將5存入R3，也就是說R3=5。
- 控制器核心執行到M/S/T碼時，會發出M/S/T碼讀取旗標給階梯圖使用，M/S/T碼分別對應S29/S54/S69。
 - 例如執行到M碼，S29會被觸發，此時就可以使用S29當作某些動作的啟動訊號。

- 階梯圖接收到M/S/T碼讀取旗標後，即開始執行規劃好的動作。
- 使用者如果將M/S/T碼寫在加工程式內的同一單節，在執行到該單節時，核心將同時發出S29/S54/S69通知階梯圖。
- 等到M/S/T碼各自規劃的動作執行完成後，需在階梯圖內觸發C38，此為M/S/T碼完成旗標。
- 控制器核心接收到C38被觸發的訊號後，才會繼續執行加工程式接下來的內容。
- 編輯階梯圖時，需特別注意C38開/關的時機，C38與S29/S54/S69之關係如下圖所示。
 - 紅色框框內的動作是編輯階梯圖時應遵守之規則
 - 綠色框框內的動作是控制器核心偵測到C38導通後的自然規格



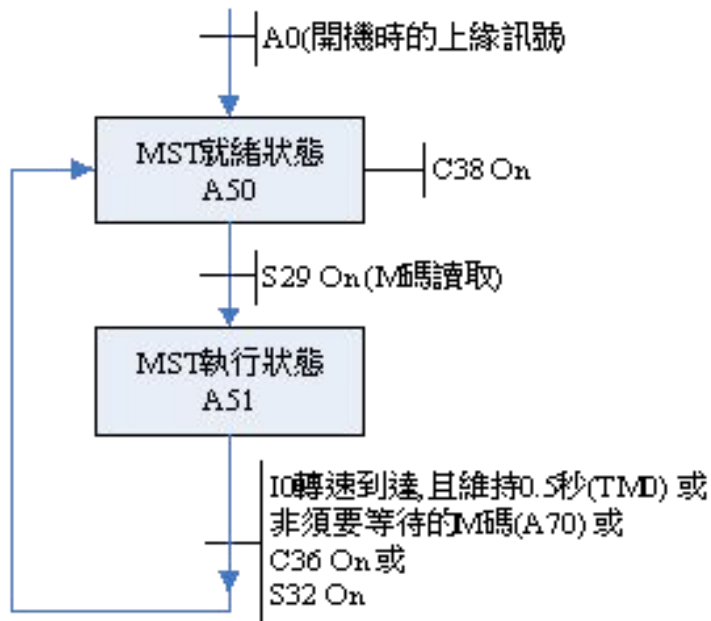
3.1.3 如何撰寫M/S/T碼

基本規則

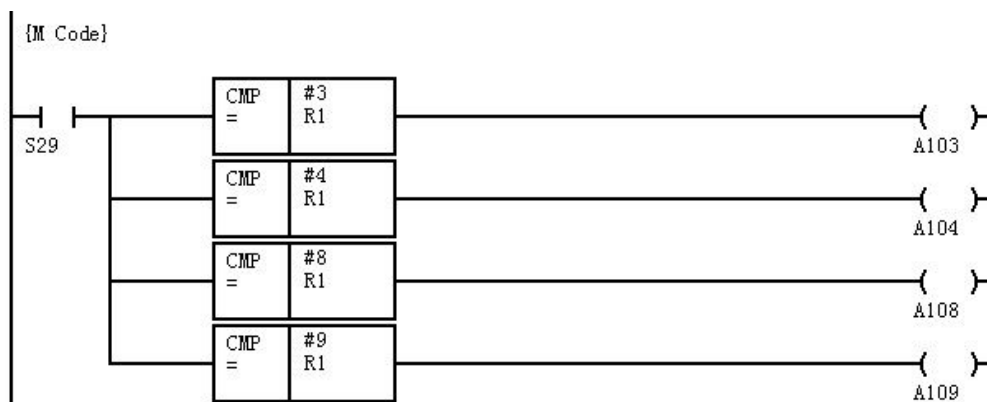
- 需要確認執行完成之M碼，請用『狀態流程圖（Machine State Diagram, MSD）』的觀念撰寫階梯圖。
- 在一般狀態或未執行M/S/T碼時，C38建議常態保持導通。
- 對於切削液、工作燈、排削機等不需確認執行完成之M碼，在M碼執行期間，不需關閉C38。
- 對於主軸正反轉、車床夾料、主軸定位等需確認執行完成之M碼，在M碼執行期間，需關閉C38，等到確認M碼執行完成後，再觸發C38。

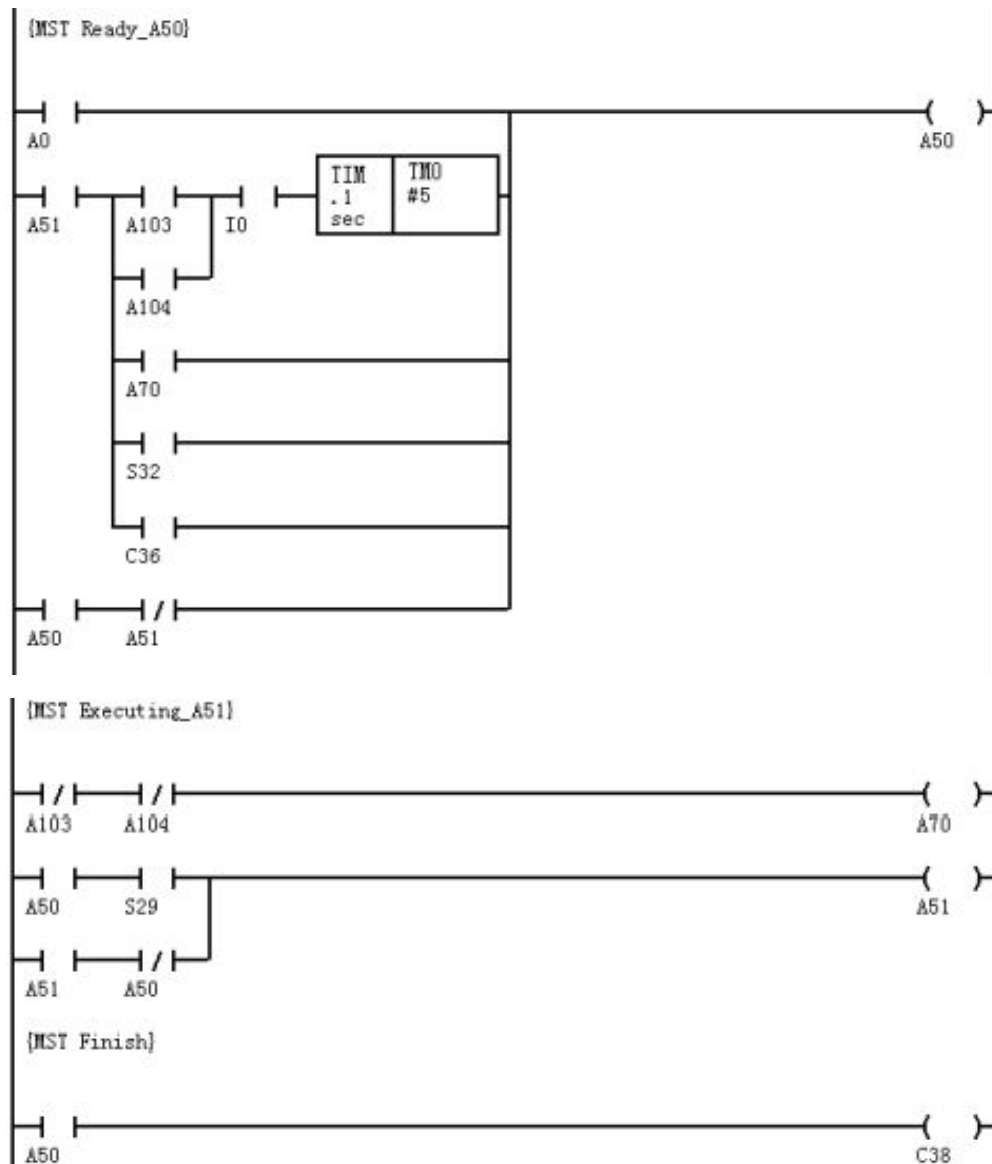
MSD範例

- 下圖採用兩個狀態來區隔M/S/T碼執行狀態
 - 當M/S/T碼處於就緒狀態（A50）時，C38為導通。
 - 當M/S/T碼處於執行狀態（A51）時，C38為關閉，以通知控制器進行等待。



- **不需確認完成之M碼：**例如M8/M9切削液開/關，規劃2個輔助接點A108/A109，直接觸發M碼完成條件A70，此時控制器不進行等待。
- **需確認完成之M碼：**例如M3/M4主軸正/反轉，規劃2個輔助接點A103/A104，由於主軸轉動必須考慮馬達之加速時間，因此必須等待主軸轉速到達後，才能繼續往下加工，故在M3/M4執行狀態下，C38需關閉，等到轉速到達訊號I0導通且維持0.5秒後，才能再次觸發C38。





元件列表:

I0	主軸轉速到達	A103	M3主軸正轉
A50	MST就緒	A104	M4主軸反轉
A51	MST執行中	A108	M8切削水開
A70	M碼不等待	A109	M9切削水關

3.1.4 M/T碼呼叫巨集功能

新代控制器提供10組M碼呼叫巨集功能，及1組T碼呼叫巨集功能，故M/T碼除了可在階梯圖內規劃動作，也能使用完整的巨集功能。

當控制器執行到欲呼叫巨集之M/T碼時，會先執行對應巨集之內容，而不會觸發S29/S69，也不會進行R1/R3填值之動作。

但如果該巨集中仍有M/T碼的話，則會觸發S29/S69並進行R1/R3填值的動作。



M碼呼叫巨集功能使用設定

- 在控制器參數Pr3601~3610中，登錄要呼叫巨集之M碼。
- 將M碼欲呼叫的巨集預先編輯好，並存入控制器中，路徑如下。
 - DOS系統：C:\CNC\MACRO
 - CE系統：DiskC/OpenCnc/MACRO
- 巨集之檔名需與M碼互相對應，例如控制器參數Pr3601=50，則巨集檔名需為M0050（無副檔名）。

T碼呼叫巨集功能使用設定

- 依照使用需求進行Pr3215之設定，可參考參數手冊：Pr3215 *選刀時呼叫模式
- 將T碼欲呼叫的巨集預先編輯好，並存入控制器中，路徑如下。
 - DOS系統：C:\CNC\MACRO
 - CE系統：DiskC/OpenCnc/MACRO
- T碼呼叫的巨集檔名固定為T0000（無副檔名）。

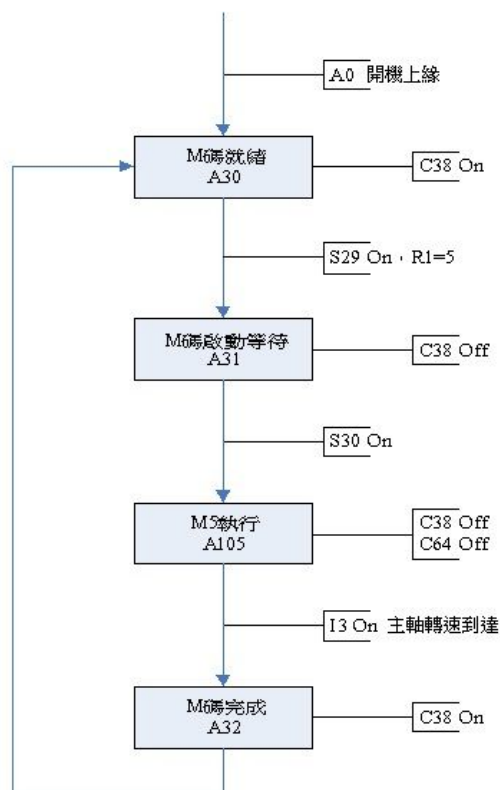
3.1.5 M/S/T碼補充說明

同一單節有G/M碼時

當G碼與M碼指令出現在同一單節時，控制器會同時發出G碼運動指令與M碼讀取旗標（S29），但由於各種M碼特性不同，有些可以跟G碼同時執行（例如：M8切削液打開），有些則必須等G碼完成後，才能執行（例如：M5主軸停止）。當後者的情況出現時，建議利用S30（同單節G碼執行結束訊號），於狀態流程圖中增加等待G碼完成狀態，以作為保護。

範例：M5主軸停止之狀態流程圖

當同一單節中有G01與M5時，需先完成G01後再執行M5。



多個M碼同時執行功能

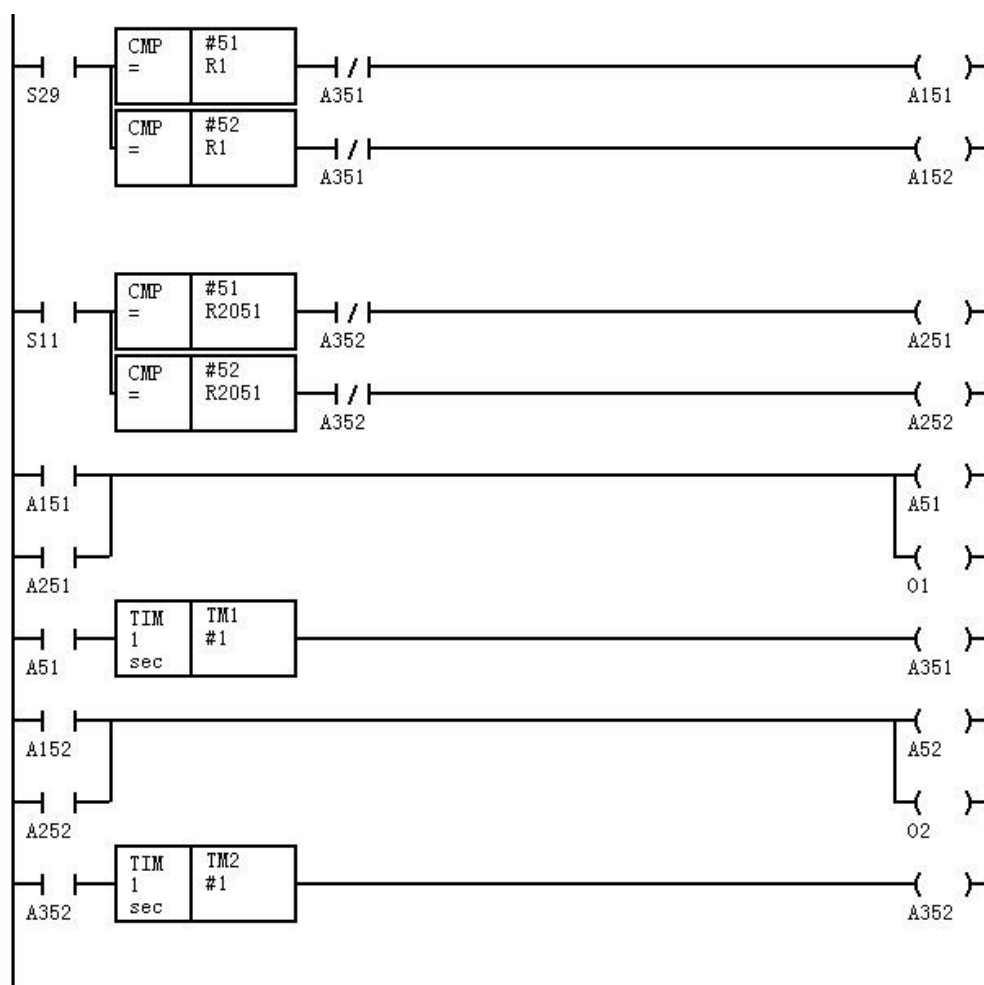
在某些應用場合，使用者需要同時執行多個M碼，而不是依序執行。此時建議開啟同一單節M碼同時執行功能，以滿足使用需求。

- 請參閱參數手冊：Pr3810 *啟動同一單節M code同時執行功能
- 下方為啟用Pr3810時，四個MST通道的相關元件。

	M碼讀取	M碼內容 (00-99)	M碼完成
第一組	• S29 • S11 • S12 • S13 • S14	• R2050 (R1) • R2051 • R2052 • R2053 • R2054	C38
第二組	• R615.24 (R615.2) • R615.25 • R615.26 • R615.27 • R615.28	• R2060 (R616) • R2061 • R2062 • R2063 • R2064	R615.0

	M碼讀取	M碼內容 (00-99)	M碼完成
第三組	• R619.24 (R619.2) • R619.25 • R619.26 • R619.27 • R619.28	• R2070 (R620) • R2071 • R2072 • R2073 • R2074	R619.0
第四組	• R623.24 (R623.2) • R623.25 • R623.26 • R623.27 • R623.28	• R2080 (R624) • R2081 • R2082 • R2083 • R2084	R623.0

- 範例：不管M碼順序為M51 M52或是M52 M51，以下方寫法皆可同時輸出O1與O2



3.2 如何運用階梯圖發出警報

階梯圖內發出之警報訊息，會自動記錄在控制器的歷來警報中，以提供使用者診斷分析。

當階梯圖內發出警報訊息時，控制器只會將警報字串顯示於控制器畫面上，並不會自動中止任何執行中的動作，如果需要中止加工或其他執行中的動作，需於階梯圖中另行處理。

3.2.1 系統內定警報

當階梯圖輸出R40～R43時，控制器畫面會依據下表內容，彈出警報視窗。

機械邏輯控制警報-MLC	
001	R40.0: X軸超過正向硬體行程極限。
002	R40.1: X軸超過負向硬體行程極限。
003	R40.2: X軸尚未原點復歸。
004	R40.3: X軸伺服驅動器警報。
009	R40.8: Y軸超過正向硬體行程極限。
010	R40.9: Y軸超過負向硬體行程極限。
011	R40.10: Y軸尚未原點復歸。
012	R40.11: Y軸伺服驅動器警報。
017	R41.0: Z軸超過正向硬體行程極限。
018	R41.1: Z軸超過負向硬體行程極限。
019	R41.2: Z軸尚未原點復歸。
020	R41.3: Z軸伺服驅動器警報。
025	R41.8: A軸超過正向硬體行程極限。
026	R41.9: A軸超過負向硬體行程極限。
027	R41.10: A軸尚未原點復歸。
028	R41.11: A軸伺服驅動器警報。
033	R42.0: B軸超過正向硬體行程極限。

機械邏輯控制警報-MLC	
034	R42.1: B軸超過負向硬體行程極限。
035	R42.2: B軸尚未原點復歸。
036	R42.3: B軸伺服驅動器警報。
041	R42.8: C軸超過正向硬體行程極限。
042	R42.9: C軸超過負向硬體行程極限。
043	R42.10: C軸尚未原點復歸。
044	R42.11: C軸伺服驅動器警報。
049	R43.0: 主軸伺服驅動器警報。
050	R43.1: 切削液馬達過載。
051	R43.2: 油壓系統警報。
052	R43.3: 風壓不足。
053	R43.4: 潤滑油不足。
054	R43.5: 切削液液位不足。
055	R43.6: 夾爪未關。
058	系統剩餘記憶體空間不足。
063	(R43.14) 釋放/夾緊抱閘失敗 (注: 此警報為機器人產品專用)
064	(R43.15) 未夾緊抱閘時無法就緒 (注: 此警報為機器人產品專用)
299	R43.16: 切削液馬達過載。
300	R43.17: 油壓系統警報。
301	R43.18: 風壓不足。
302	R43.19: 潤滑油不足。
303	R43.20: 切削液液位不足。
313	R43.30: 啟動保護, 請按正向試跑移動到目標點。(注: 此警報為機器人產品專用)
314	R43.31: 啟動保護完成, 按下啟動按鈕以啟動程式。(注: 此警報為機器人產品專用)

1. MLC-001~MLC-048等描述軸向的警報，以1軸8個警報為單位依序排列。
例如：
MLC-001~MLC-008為第1軸警報
MLC-009~MLC-016為第2軸警報
....
MLC-041~MLC-048為第6軸警報
2. 若對應軸向未開啟，則相關警報都無法被觸發
例如：
第5軸未開啟(Pr25設0)時，警報MLC-033~040無法被觸發
3. 若對應軸向與上表警報字串中的軸名稱不匹配時，使用者應自行定義警報字串。
例如: Pr321【第1軸名稱】設為200(Y軸)，Pr322【第2軸名稱】設為100(X軸)時，則須定義：
MLC-001~MLC-008: 須將警報字串內容調整為「Y軸」
MLC-009~MLC-016: 須將警報字串內容調整為「X軸」

警報觸發方式

如下圖所示，外部訊號I10觸發R40.0，而R40.0對應PLC Alarm編號01。



3.2.2 自定義警報

新代控制器提供96個PLC Alarm、96個PLC Warning及319個PLC Hint供使用者自行定義，其中Alarm/Warning/ Hint之編號與R Bit對應關係，請參閱下表。
PLC Alarm編號01至64為系統內定警報區，使用者請勿重複使用。

型態	R Bit編號	警報字串編號
PLC警報 (Alarm)	R44.0 ~ R44.15	65 ~ 80
	R45.0 ~ R45.15	81 ~ 96
	R46.0 ~ R46.15	97 ~ 112
	R47.0 ~ R47.15	113 ~ 128
	R48.0 ~ R48.15	129 ~ 144
	R49.0 ~ R49.15	145 ~ 160
PLC警告 (Warning)	R44.16 ~ R44.31	315 ~ 330
	R45.16 ~ R45.31	331 ~ 346
	R46.16 ~ R46.31	347 ~ 362
	R47.16 ~ R47.31	363 ~ 378

	R48.16 ~ R48.31	379 ~ 394
	R49.16 ~ R49.31	395 ~ 410
PLC提示 (Hint)	R590.1 ~ R590.31	1 ~ 31
	R591.0 ~ R591.31	32 ~ 63
	R592.0 ~ R592.31	64 ~ 95
	R593.0 ~ R593.31	96 ~ 127
	R594.0 ~ R594.31	128 ~ 159
	R595.0 ~ R595.31	160 ~ 191
	R596.0 ~ R596.31	192 ~ 223
	R597.0 ~ R597.31	224 ~ 255
	R598.0 ~ R598.31	256 ~ 287
	R599.0 ~ R599.31	288 ~ 319
備註	1. PLC警報 (Alarm) 會儲存於歷來警報；PLC警告 (Warning) 與PLC提示 (Hint) 不會儲存於歷來警報。 2. PLC提示 (Hint) 不需自保持，警報視窗即可持續顯示在畫面上。 3. PLC警報 (Alarm) 與PLC警告 (Warning) 必須在階梯圖內自保持，才能讓警報視窗持續顯示在畫面上。	

DOS版本警報字串檔

- 檔案路徑
繁/簡中文：C:\CNC\APP\APPCHI.STR
英文版：C:\CNC\APP\APPENG.STR
其他：C:\CNC\APP\CNCLOC.STR
- 檔案格式
20xxx="1;MSG=警報內容"，其中xxx對應PLC Alarm/Warning編號，請注意識別代號為20。
23xxx="1;MSG=警報內容"，其中xxx對應PLC Hint編號，請注意識別代號為23。
- 撰寫範例
APPCHI.STR範例：
20001="1;MSG=X軸超過正向硬體行程極限"
23010="1;MSG=第10號提示"
- APPENG.STR範例：
20001="1;MSG=X+ hardware stroked limit exceed"
23010="1;MSG=10th Hint"

CE版本警報字串檔

- 檔案路徑 (10.114.46以前)
中文版: DiskC/OCRes/CHT/String/AlarmMsg_CHT.xml
英文版: DiskC/OCRes/Common/String/AlarmMsg_common.xml
- 檔案路徑 (10.114.46以後)
繁體中文版: DiskC/OCRes/CHT/String/AlarmPLC_CHT.xml
簡體中文版: DiskC/OCRes/CHS/String/AlarmPLC_CHS.xml
英文版: DiskC/OCRes/Common/String/AlarmPLC_COM.xml
- 檔案格式
<Message ID="AlarmMsg::MLC::ID=xxx" Content="警報內容" />, 其中xxx對應PLC Alarm/Warning編號, 請注意識別字母為MLC。
<Message ID="AlarmMsg::MLCHint::ID=xxx" Content="警報內容" />, 其中xxx對應PLC Hint編號, 請注意識別字母為MLCHint。
警報內容的字串長度可容納48個英文字, 或是31個中文字, 多餘的字串會超出警報視窗而無法顯示。
- 撰寫範例
AlarmPLC_CHT.xml:
<Message ID="AlarmMsg::MLC::ID=1" Content="X軸超過正向硬體行程極限" />
<Message ID="AlarmMsg::MLCHint::ID=10" Content="第10號提示" />
AlarmPLC_COM.xml:
<Message ID="AlarmMsg::MLC::ID=1" Content="X+ hardware stroked limit exceed " />
<Message ID="AlarmMsg::MLCHint::ID=10" Content="10th Hint " />

3.3 操作面板應用範例

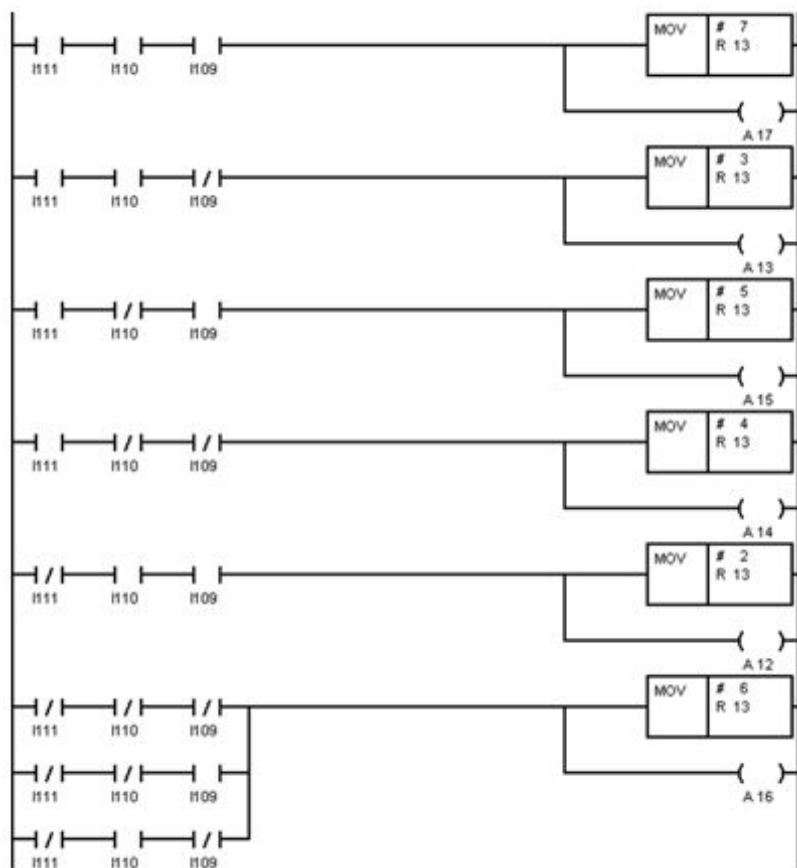
模式選擇



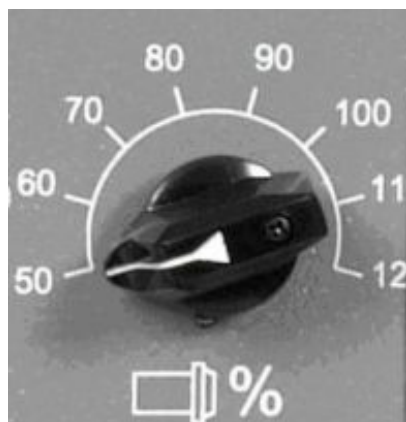
控制器模式 (R13) 選擇由I-109/I-110/I-111三個I bit以二進位碼組合而成。
其輸出與輸入之轉換關係如下表所示:

輸入			輸出
I-111	I-110	I-109	R13
1	1	1	7: HOME (尋原點)
1	1	0	3: MDI (手動輸入)
1	0	1	5: INJOG (增量寸動)
1	0	0	4: JOG (連續寸動)
0	1	1	2: Auto (自動執行)
0	1	0	6: MPG (手輪)
0	0	1	6: MPG (手輪)
0	0	0	6: MPG (手輪)

階梯圖範例:



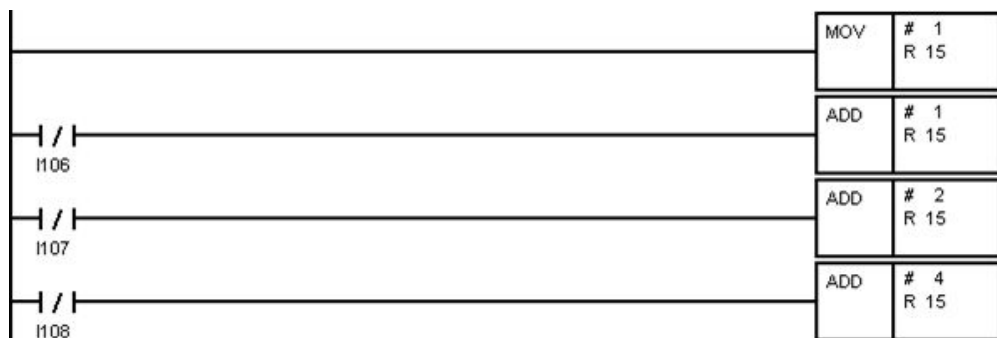
主軸倍率



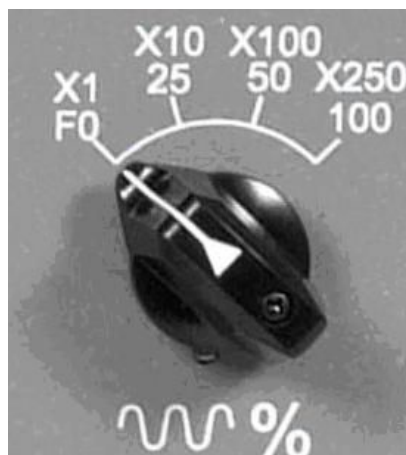
主軸倍率（R15）是由I-106/I-107/I-108三個I bit以二進位碼組合而成。
其輸出與輸入之轉換關係如下表所示：

輸入			輸出
I-108 (+4)	I-107 (+2)	I-106 (+1)	R15
1	1	1	1+0+0+0=1: 50%
1	1	0	1+0+0+1=2: 60%
1	0	1	1+0+2+0=3: 70%
1	0	0	1+0+2+1=4: 80%
0	1	1	1+4+0+0=5: 90%
0	1	0	1+4+0+1=6: 100%
0	0	1	1+4+2+0=7: 110%
0	0	0	1+4+2+1=8: 120%

備註：由Pr3207 *C_S界面版本編號來決定此R bit表示方式為百分比或內訂段數。
階梯圖範例：



MPG/ INJOG/ G00倍率



MPG/INJOG段數 (R14) 以及G00倍率 (R18) 是由I-104/ I-105二個I bit以二進位碼組合而成。
其輸出與輸入之轉換關係如下表所示：

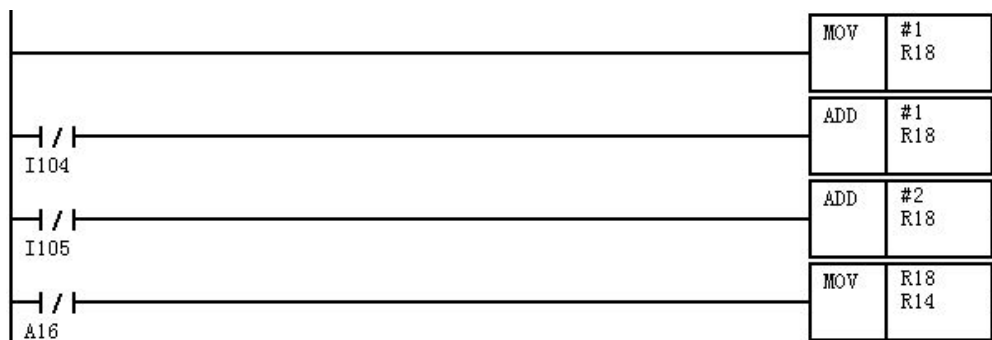
輸入		輸出	
I-105 (+2)	I-104 (+1)	R14	R18
1	1	1+0+0=1: *1	1+0+0=1: F0
1	0	1+0+1=2: *10	1+0+1=2: 25%
0	1	1+2+0=3: *100	1+2+0=3: 50%
0	0	1+2+1=4: *1000	1+2+1=4: 100%

備註：

1. R14設定可參考Pr2001 手輪第四段倍率設定
2. R14設定手輪段數，僅適用一號手輪。各號手輪段數對應Rbit列表如下所示：

手輪編號	R bit
一號手輪	R14
二號手輪	R22
三號手輪	R23
四號手輪	R934
五號手輪	R935
六號手輪	R936
七號手輪	R937

階梯圖範例：



輸 入				輸 出
I-103 (+8)	I-102 (+4)	I-101 (+2)	I-100 (+1)	R16 & R17
1	1	1	1	0+0+0+0+0=0: 0%
1	1	1	0	0+0+0+0+1=1: 10%
1	1	0	1	0+0+0+2+0=2: 20%
1	1	0	0	0+0+0+2+1=3: 30%
1	0	1	1	0+0+4+0+0=4: 40%
1	0	1	0	0+0+4+0+1=5: 50%
1	0	0	1	0+0+4+2+0=6: 60%
1	0	0	0	0+0+4+2+1=7: 70%

輸入				輸出
0	1	1	1	0+8+0+0=8: 80%
0	1	1	0	0+8+0+0+1=9: 90%
0	1	0	1	0+8+0+2+0=10: 100%
0	1	0	0	0+8+0+2+1=11: 110%
0	0	1	1	0+8+4+0+0=12: 120%
0	0	1	0	0+8+4+0+1=13: 130%
0	0	0	1	0+8+4+2+0=14: 140%
0	0	0	0	0+8+4+2+1=15: 150%

備註：由Pr3207 *C_S界面版本編號來決定此R bit表示方式為百分比或內訂段數。

階梯圖範例：



手輪模擬

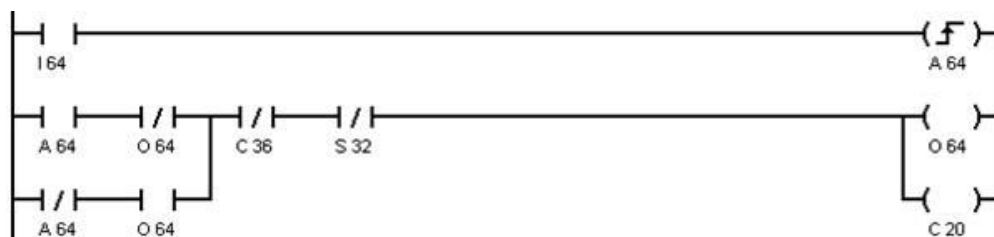


手輪模擬（MPG Simulation）需在階梯圖內編輯C20元件才可使用。

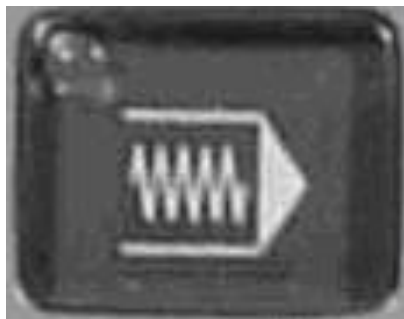
當切換至Auto（自動執行）或MDI（手動輸入）模式且C20導通的條件下，啟動加工後，G00/G01/G02/G03速度將由手輪轉動速度決定。

轉動越快，機台移動越快，手輪停止，機台就停止，適用於試加工階段。

階梯圖範例：



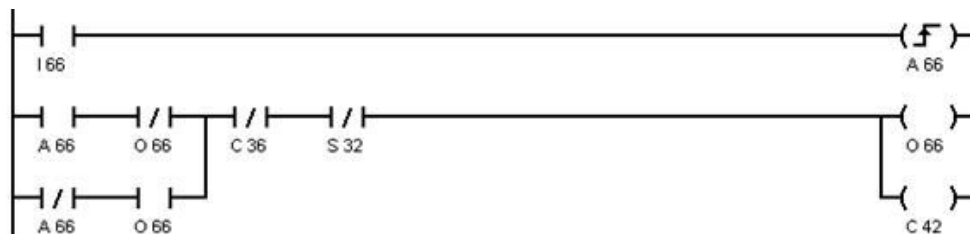
程式空跑



程式空跑（Dry Run）需在階梯圖內編輯C42元件才可使用。

C42導通，機台執行到移動G碼會以空跑速度移動，G01單節為程式速度的五倍，G00單節即為G00速度。

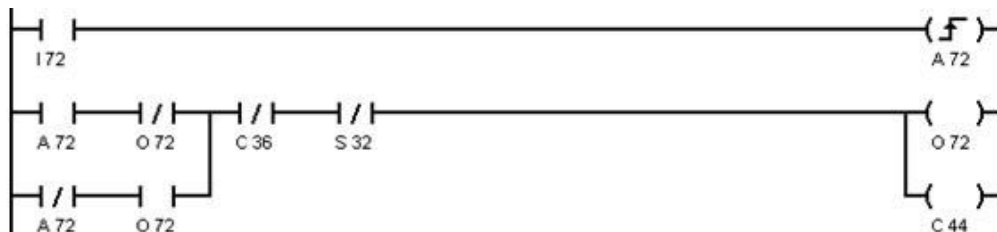
階梯圖範例：



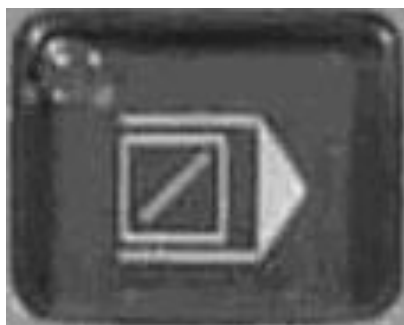
選擇性停止



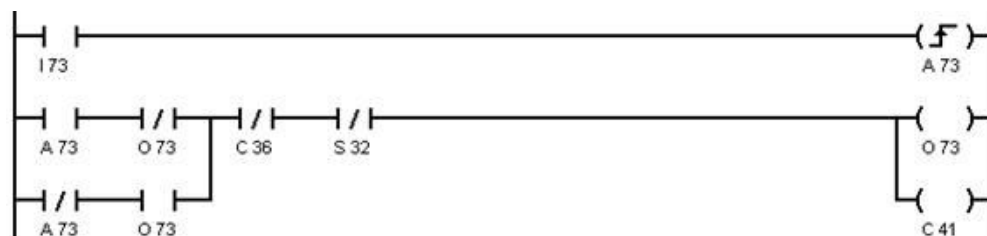
選擇性停止 (Optional Stop) 需在階梯圖內編輯C44元件才可使用。
C44導通，執行到M01時，控制器會進入單節停止狀態 (B-STOP)。C44斷開，控制器會忽略M01，繼續執行下個單節。
階梯圖範例：



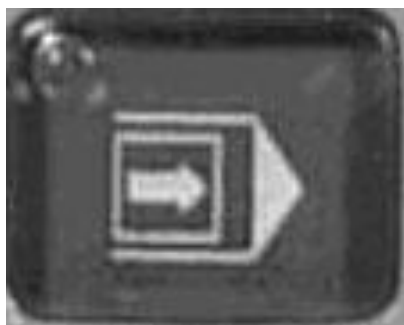
選擇性跳躍



選擇性跳躍 (Optional Skip) 需在階梯圖內編輯C41元件才可使用。
C41導通，當執行加工遇到跳躍符號"/"時，控制器會跳過該單節，直接執行下一單節。
階梯圖範例：

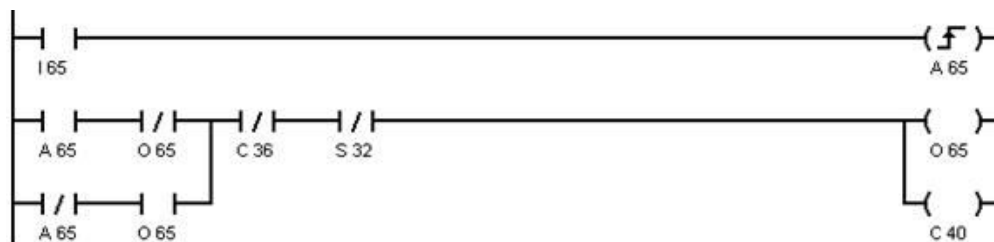


單節執行



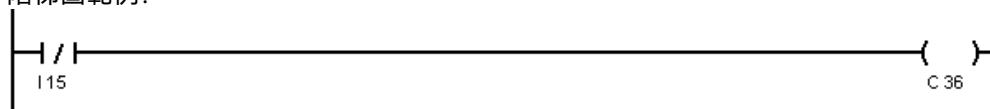
單節執行 (Single Block) 需在階梯圖內編輯C40元件才可使用。

C40導通，執行加工時走一個單節即暫停，需再按啟動（Cycle Start）才會繼續走下一個單節。
 階梯圖範例：



緊急停止

緊急停止（Emergency Stop）需在階梯圖內編輯C36元件才可使用。
 C36導通，控制器會暫停機台所有的動作，並進入未就緒狀態（Not Ready），直到C36斷開，系統恢復就緒（Ready）狀態。
 階梯圖範例：



面板JOG鍵



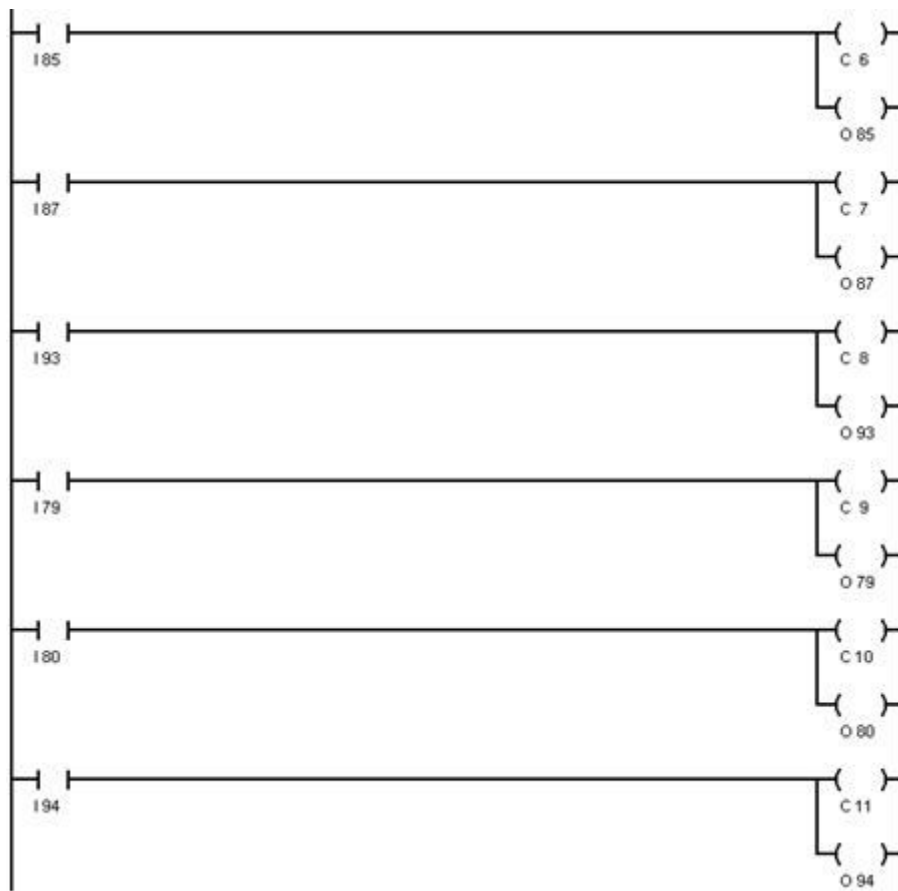
JOG模式下，按壓面板對應軸向的方向鍵，並在階梯圖內觸發下表對應的C Bit，控制器將產生對應的運動。

若再搭配快動鍵的觸發（C23），將使該軸向以G00速度移動。

PS：下表的I bit僅為某一款標準面板的對應編號，並非所有面板都相同。

軸向/方向	X+	X-	Y+	Y-	Z+	Z-
C Bit	C06	C07	C08	C09	C10	C11
車床	I-65	I-71	I-64	I-70	I-69	I-67
銑床	I-85	I-87	I-93	I-79	I-80	I-94

階梯圖範例：



程式啟動/ 暫停

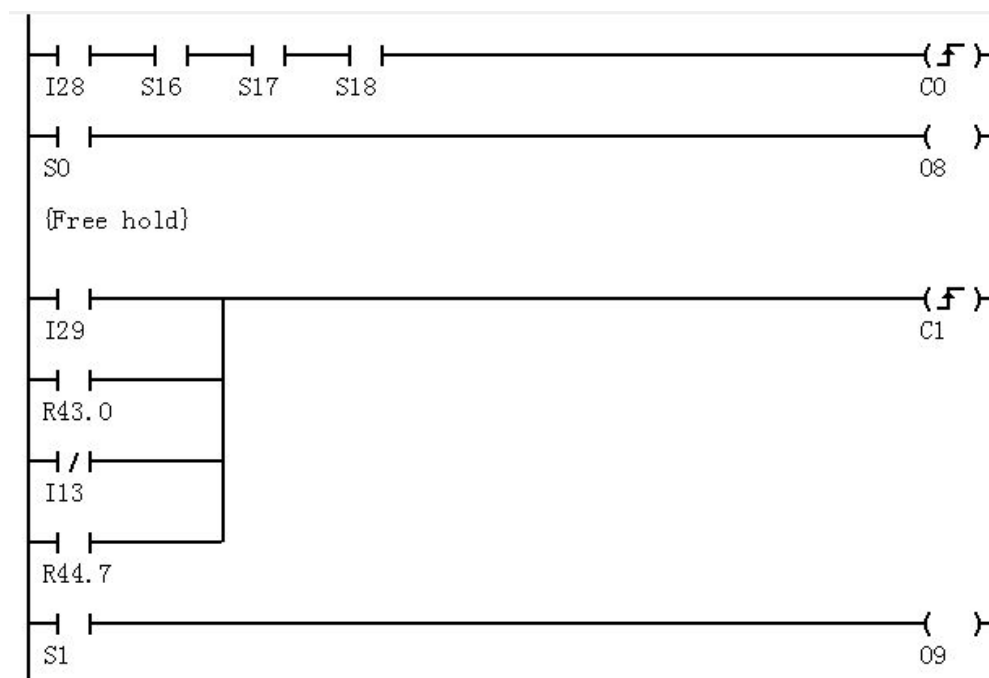
程式啟動（Cycle Start）需在階梯圖內編輯C0元件才可使用，程式暫停（Feed Hold）需在階梯圖內編輯C1元件才可使用。

在Auto或MDI模式下，C0僅需上緣觸發即可啟動加工，啟動加工後S0會導通。

C1同樣僅需上緣觸發即可暫停加工，暫停後S1會導通，S0被關閉。

階梯圖範例：

SYNTEC

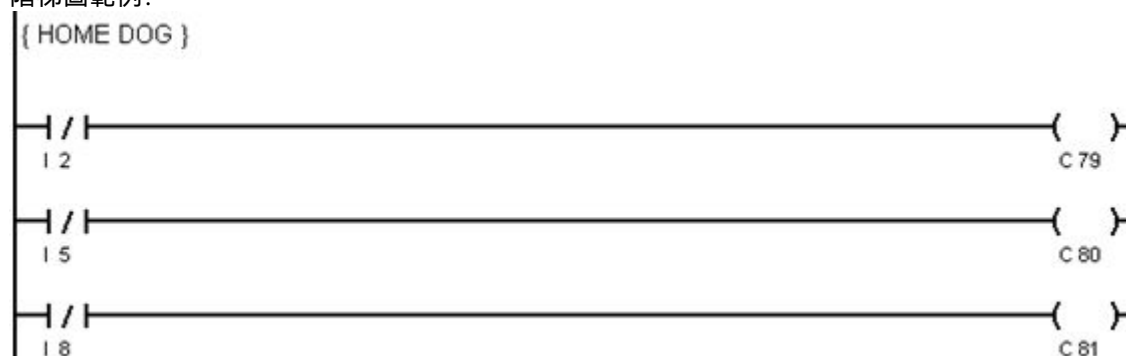


尋原點

HOME（尋原點）模式下，按壓各軸JOG鍵，控制器會根據參數設定往原點方向移動。當遇到原點開關（Home Dog）時，需觸發下表對應之C Bit；待尋原點完成後，控制器將觸發下表對應尋原點完成之S Bit。

	X軸	Y軸	Z軸
原點開關	C79	C80	C81
尋原點完成	S16	S17	S18

階梯圖範例：



硬體行程極限

下表為各軸硬體正負極限對應的C Bit，當硬體極限的C Bit被觸發時，控制器將自動發出警報並停止機台動作，此時控制器僅能接受反方向命令將機台帶出極限位置。

	X正極限	X負極限	Y正極限	Y負極限	Z正極限	Z負極限
C Bit	C50	C51	C52	C53	C54	C55

階梯圖範例：



PS: 有時候硬體極限開關故障，導致警報無法消除，此時就必須有手動解除的方法，例如上圖內的I-92。

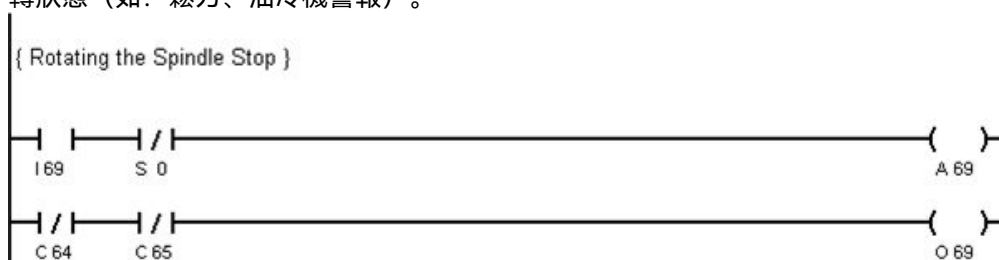
主軸正轉/ 反轉/ 停止

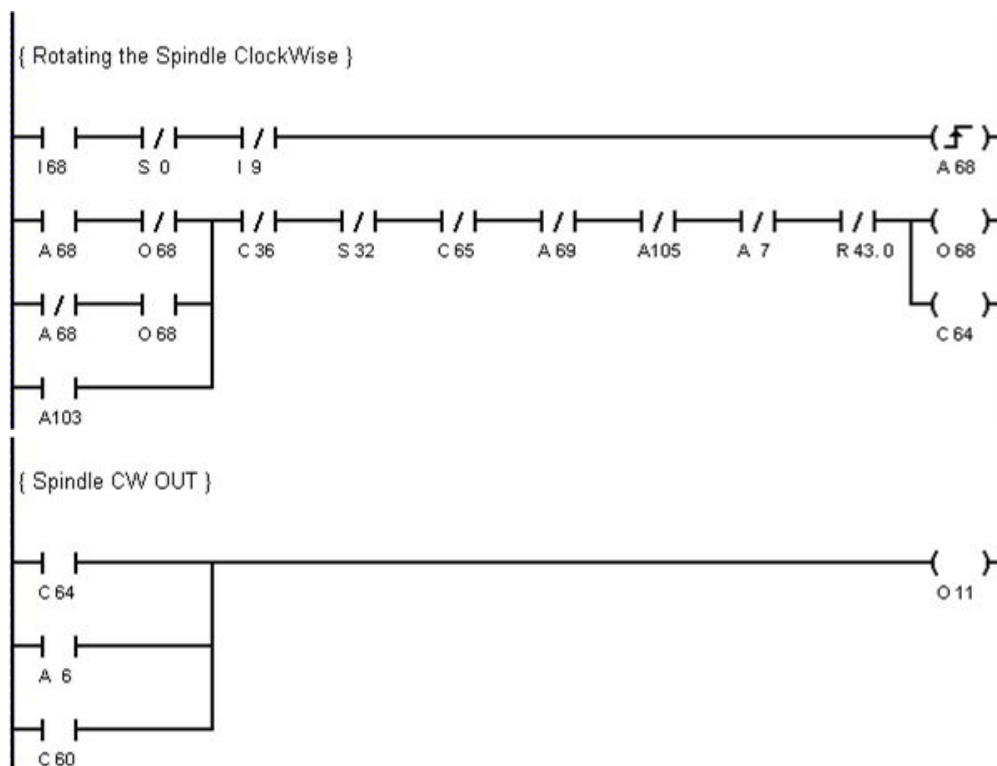
主軸正/反轉分別對應C64與C65，一般使用上有幾點需特別注意：

1. 按重置鍵（Reset）或程式執行到M05、M02與M30的時候，需切斷C64、C65。
2. 車床夾頭未閉合，及銑床在鬆刀狀態下，主軸不可轉動。
3. 加工中（S0 On）不可手動將主軸停止，但可以使用M05停止主軸。

階梯圖範例：

I-69為面板主軸停止按鍵；S32為重置訊號；A103為M03主軸正轉；A105為M05主軸停止；A7為主軸禁止運轉狀態（如：鬆刀、油冷機警報）。





3.4 銑床斗笠式刀庫應用範例

M碼說明

M碼定義總表

於標準銑床階梯圖中，提供下列M碼功能：

M碼	功能說明
M03	主軸正轉
M04	主軸反轉
M05	主軸停止
M07	加工吹氣電磁閥開啟
M08	切削液馬達開啟
M09	加工吹氣電磁閥關閉、切削液馬達關閉

M碼	功能說明
M19	主軸定位
M30	加工吹氣電磁閥關閉、切削液馬達關閉、主軸停止
M37	排屑機正轉
M38	排屑機反轉
M50	主軸鬆刀
M51	主軸夾刀
M52	刀庫推出
M53	刀庫退回

PS: M01選擇性停止、M02程式結束、M98呼叫副程式、M99程式返回等四個M碼，是由控制器核心提供，並非階梯圖處理，因此未列入上表中。

M碼操作規格說明

1. M03（主軸正轉）：此指令可使主軸作順時鐘方向旋轉，與S碼一起使用，讓主軸依設定轉速旋轉，動作規格如下：
 - a. 執行M03時，主軸正轉啟動直到主軸轉速（R36）到達目標轉速（R871~）的80%後，再繼續往下執行加工程式。
 - b. 主軸正轉訊號（C64 On）輸出期間，會一直檢查主軸轉速是否高於目標轉速的70%，如果轉速未達目標轉速的70%，會立即暫停加工，並發出警報訊息。
2. M04（主軸反轉）：此指令可使主軸作逆時鐘方向旋轉，與S碼一起使用，讓主軸依設定轉速旋轉，動作規格如下：
 - a. 執行M04時，主軸反轉啟動直到主軸轉速（R36）到達目標轉速（R871~）的80%後，再繼續往下執行加工程式。
 - b. 主軸反轉訊號（C65 On）輸出期間，會一直檢查主軸轉速是否高於目標轉速的70%，如果轉速未達目標轉速的70%，會立即暫停加工，並發出警報訊息。
3. M05（主軸停止）：此指令可使主軸停止轉動，同時也可以使用此指令取消主軸定位狀態，動作規格如下：
 - a. 執行M05時，切斷C64、C65，使主軸減速停止。
 - b. 當主軸轉速（R36）低於10 RPM，將關閉主軸控制。（泛用主軸適用）
4. M19（主軸定位）：此指令可使主軸進行定位動作，動作規格如下：
 - a. 執行M19時，輸出主軸定位訊號（C61）。
 - b. 當控制器收到主軸定位完成訊號（S61）後，再繼續往下執行加工程式。
5. M50（主軸鬆刀）：此指令可使主軸進行鬆刀動作，動作規格如下：
 - a. 執行M50時，輸出主軸鬆刀訊號。
 - b. 當控制器收到主軸鬆刀完成訊號後，再繼續往下執行加工程式。
6. M51（主軸夾刀）：此指令可使主軸進行夾刀動作，動作規格如下：
 - a. 執行M51時，切斷主軸鬆刀訊號。
 - b. 當控制器收到主軸夾刀完成訊號後，再繼續往下執行加工程式。
7. M52（刀庫推出）：此指令可使刀庫進行推出動作，動作規格如下：
 - a. 執行M52時，輸出刀庫推出訊號。
 - b. 當控制器收到刀庫推出完成訊號後，再繼續往下執行加工程式。
8. M53（刀庫退回）：此指令可使刀庫進行退回動作，動作規格如下：

- a. 執行M53時，切斷刀庫推出訊號。
- b. 當控制器收到刀庫退回完成訊號後，再繼續往下執行加工程式。

M碼技術規格說明

1. M03 (主軸正轉) :
 - a. 主軸夾刀狀態下，才可以執行M03。
 - b. 啟動主軸正轉10秒內，主軸轉速 (R36) 未達目標轉速 (R871~) 的80%，即發出『主軸正轉啟動異常』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
 - c. 在主軸正轉狀態下，持續檢查主軸轉速，如發現主軸轉速低於目標轉速的70%，即發出『主軸轉速異常偏低』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
2. M04 (主軸反轉) :
 - a. 主軸夾刀狀態下，才可以執行M04。
 - b. 啟動主軸反轉10秒內，主軸轉速 (R36) 未達目標轉速 (R871~) 的80%，即發出『主軸反轉啟動異常』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
 - c. 在主軸反轉狀態下，持續檢查主軸轉速，如發現主軸轉速低於目標轉速的70%，即發出『主軸轉速異常偏低』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
3. M05 (主軸停止) :
 - a. 執行M05時，不進行M碼等待，即C38保持導通。
 - b. 主軸轉速低於10 RPM才切掉主軸執行 (RUN) 的外部訊號。(泛用主軸適用)
4. M19 (主軸定位) :
 - a. 主軸夾刀狀態下，才可以執行M19。
 - b. 啟動主軸定位3秒內，未有主軸定位完成確認訊號傳回控制器，即發出『主軸定位失敗』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
 - c. 主軸定位中，變頻器傳回的定位完成訊號異常中斷，即發出『主軸定位異常中斷警報』，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
 - d. 主軸定位中，可接受M03、M04及M05解除，並直接啟動下一動作。
5. M50 (主軸鬆刀) :
 - a. 主軸正/反轉時，禁止執行主軸鬆刀。
 - b. 主軸轉速 (R36) 低於10 RPM時，才可以執行主軸鬆刀。
 - c. 啟動主軸鬆刀3秒內，未有主軸鬆刀確認訊號傳回控制器，即發出『主軸鬆刀失敗』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
 - d. 主軸鬆刀確認訊號持續0.1秒，主軸鬆刀才算完成。
6. M51 (主軸夾刀) :
 - a. 主軸正/反轉時，禁止執行主軸夾刀。
 - b. 主軸轉速 (R36) 低於10 RPM時，才可以執行主軸夾刀。
 - c. 啟動主軸夾刀3秒內，未有主軸夾刀確認訊號傳回控制器，即發出『主軸夾刀失敗』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
 - d. 主軸夾刀確認訊號持續0.5秒，主軸夾刀才算完成。
7. M52 (刀庫推出) :
 - a. 主軸正/反轉時，禁止執行刀庫推出。
 - b. 主軸轉速 (R36) 低於10 RPM時，才可以執行刀庫推出。
 - c. 啟動刀庫推出10秒內，未有刀庫推出確認訊號傳回控制器，即發出『刀庫推出失敗』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
 - d. 刀庫推出確認訊號持續0.1秒，刀庫推出才算完成。
8. M53 (刀庫退回) :
 - a. 主軸正/反轉時，禁止執行刀庫退回。
 - b. 主軸轉速 (R36) 低於10 RPM時，才可以執行刀庫退回。
 - c. 啟動刀庫退回10秒內，未有刀庫退回確認訊號傳回控制器，即發出『刀庫退回失敗』警報，同時暫停加工，此時需按壓重置 (Reset) 鍵解除。
 - d. 刀庫退回確認訊號持續0.1秒，刀庫退回才算完成。
9. 其他:
 - a. 加工中，於操作面板手動操控主軸無效。

- b. 當G碼指令與M05指令在同一單節時，須等待G碼動作執行完成後（S30導通），才可執行M05動作。
10. 注意事項：
 - a. 當發生換刀失敗警報且順利排除後，請務必主動確認控制器記錄之刀庫刀號、主軸刀號與實際刀號是否相符。如有異常，請修正控制器記錄的刀號資料表，然後關閉電源重新開機。

T碼說明

T碼操作規格說明

於加工程式中執行T碼，將進行主軸刀具變更，動作順序如下：

1. 主軸停止（M05）
2. 刀庫旋轉並數刀至目前主軸刀號
3. 主軸定位（M19）
4. Z軸快速移動至第二參考點（Pr2803）
5. 刀庫推出（M52）
6. 主軸鬆刀（M50）
7. Z軸快速移動至機械原點
8. 刀庫旋轉並數刀至目標刀號
9. Z軸快速移動至第二參考點（Pr2803）
10. 主軸夾刀（M51）
11. 刀庫退回（M53）
12. 主軸定位解除（M05）
13. 改寫主軸刀號
14. 繼續執行加工程式

T碼技術規格說明

1. 刀庫推出時，禁止主軸正/反轉及手動旋轉刀盤。
2. 啟動刀盤轉動3秒內，未有數刀訊號傳回控制器，即發出『數刀異常』警報，同時暫停加工。
3. T碼執行過程，若出現『主軸定位失敗』、『刀庫推出失敗』、『主軸鬆刀失敗』、『主軸夾刀失敗』、『刀庫退回失敗』或『數刀異常』等警報時，即發出『換刀失敗』警報，同時暫停加工，此時需待主軸夾刀與刀庫退回訊號出現後，按壓重置（Reset）鍵解除。

T碼呼叫巨集功能使用說明

斗笠式換刀是由一連串動作組成，所以使用T碼呼叫巨集方式來實現會比較恰當。

當使用者完成以下設定後，在執行T碼時，會優先進入T0000中，然後再根據T0000內容觸發相關PLC動作。

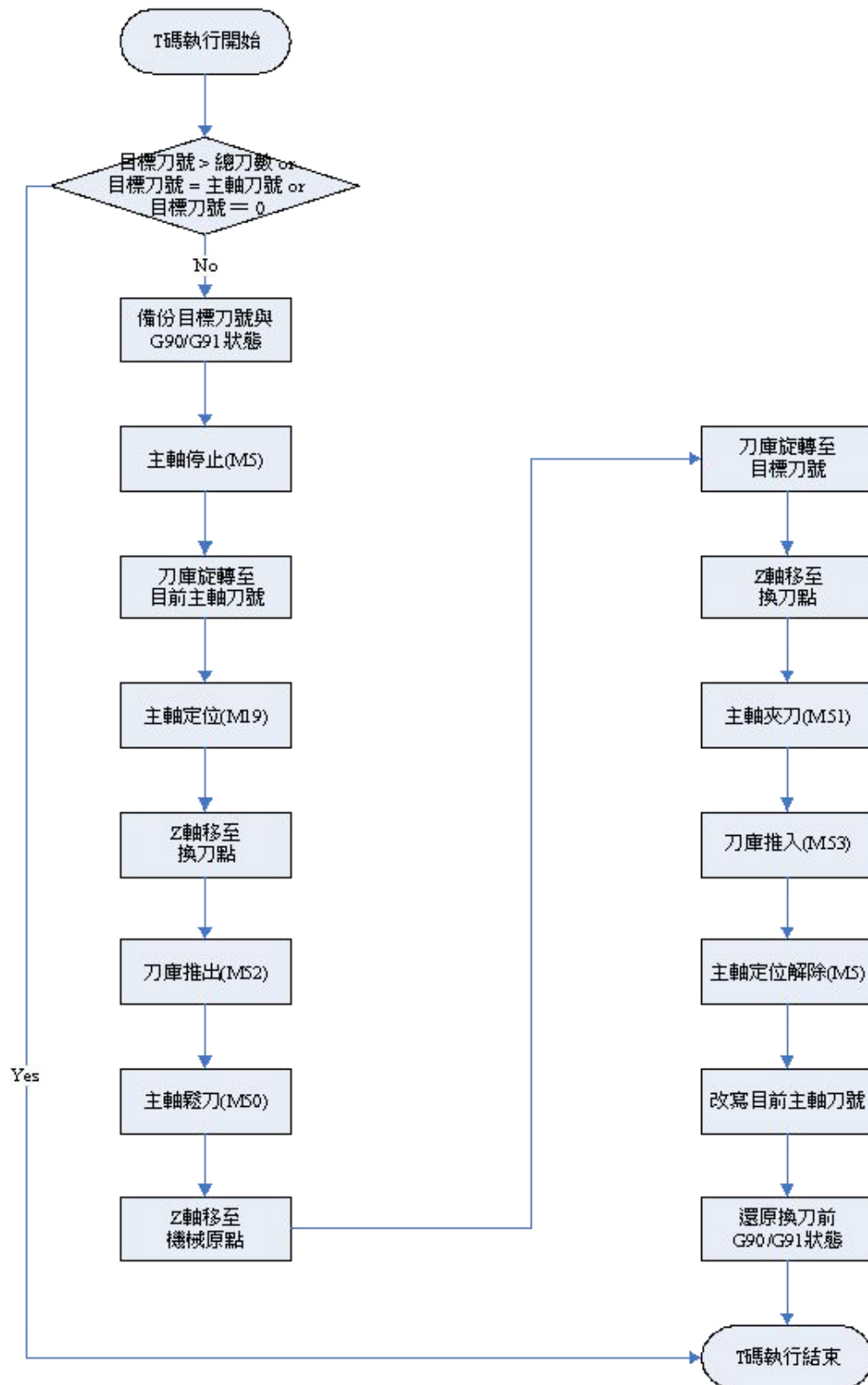
1. 將Pr3215設定為1。
2. 將T碼呼叫的巨集預先編輯好，並存入控制器。T碼呼叫的巨集檔名需為T0000（無副檔名）。

SYNTEC

T碼動作流程圖



SYNTEC



T0000巨集範例

```
%@MACRO
// @481=R81_總刀數

//@502=R102_目前主軸刀號

// #1004_G90/G91模式; #1036_目標刀號; #26023=Pr2803_Z軸換刀點

IF ( @502 = #1036 OR #1036 = 0 OR @481 < #1036 ) THEN
    M99;    //判斷若目標刀號=目前刀號 或 0號刀 或 大於總刀數, 就跳出T0000
END_IF;

@102:=#1036;    //儲存目標刀號
@103:=#1004;    //儲存進入T0000前的G90/G91狀態
M3S0;    //強迫轉速為0 RPM, 避免變頻器跳過電流警報
M05;
WAIT();
T@502;    //旋轉刀庫至目前刀號
M19;    //主軸定位
G90 G53 Z#26023;    //Z軸移動到換刀點 (Pr2803)
G4X0.5;
WAIT();
M52;    //刀庫推出
M50;    //主軸鬆刀
G90 G53 Z0;    //Z軸移動到機械原點
WAIT();
T@102;    //旋轉刀庫至目標刀號
G90 G53 Z#26023;    //Z軸移動到換刀點 (Pr2803)
G4X0.5;
WAIT();
M51;    //主軸夾刀
M53;    //刀庫推入
M05;    //主軸定位解除
WAIT();
@502:=@102;    //改寫目前刀號
G@103;    //回復進入T0000前的G90/G91狀態
M99;
```

S碼說明

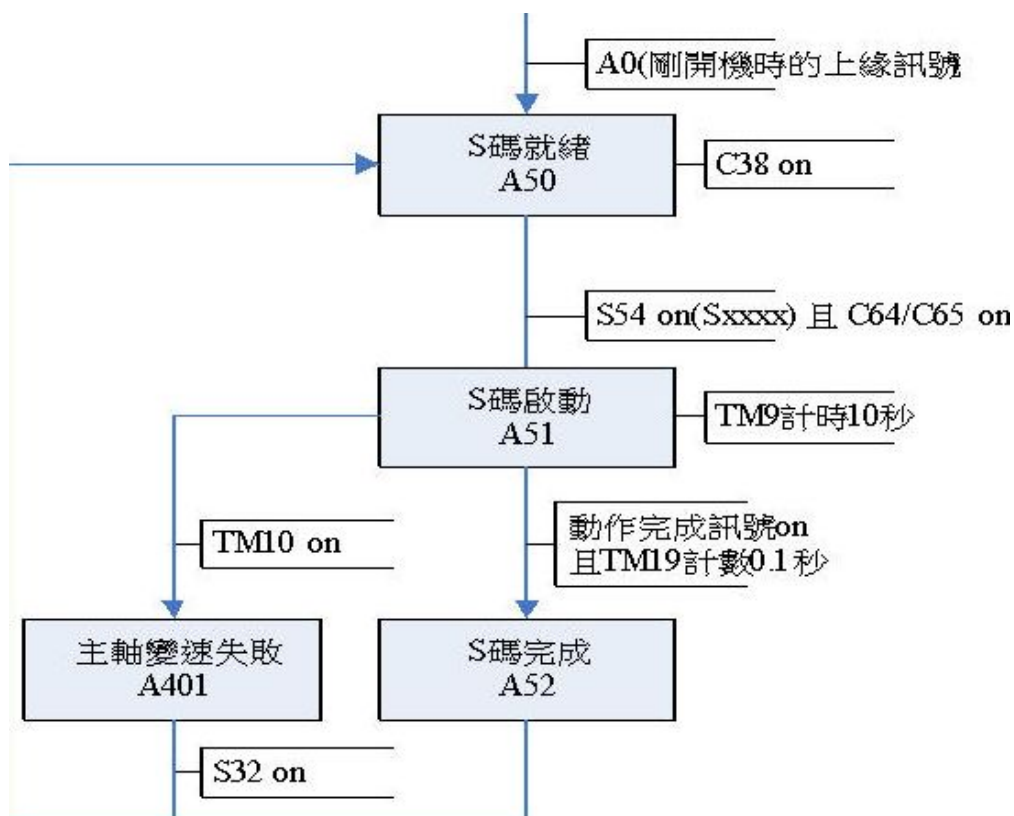
S碼操作規格說明

1. 在主軸轉動時執行S碼，將進行主軸轉速變換。
2. 當主軸轉速（R36）高於目標轉速（R871~）的80%，且低於目標轉速的110%，再繼續往下執行加工程式。

S碼技術規格說明

1. 啟動主軸轉速變換10秒內，主軸轉速（R36）無法高於目標轉速（R871~）的80%，且低於目標轉速的110%時，即發出『主軸轉速變換失敗』警報，同時暫停加工。
2. 須待主軸轉速（R36）高於目標轉速（R871~）的80%，且低於目標轉速的110%，狀態持續0.1秒，主軸轉速變換才算完成。

S碼機械狀態圖（MSD）



3.5 常用C/S/R整理

3.5.1 軸向相關

功能	方向	說明	1	2	3	4	5	6	7	8
			9	10	11	12	13	14	15	16

軸 向 旗 標 與 暫 存 器	C	軸向JOG (+)	C6	C8	C10	C170	C172	C12	C174	C176
			C178	C180	C182	C184	C186	C188	C190	C192
	C	軸向JOG (-)	C7	C9	C11	C171	C173	C13	C175	C177
			C179	C181	C183	C185	C187	C189	C191	C193
	C	手輪軸向選擇	C16	C17	C18	C215	C216	C19	C217	C218
			C219	C220	C221	C222	C223	C224	C225	C226
	C	設定軸向機械座標	C25	C26	C27	C230	C231	C28	C232	C233
			C234	C235	C236	C237	C238	C239	C240	C241
	C	手動軸向控制 (監看模式)	C31	C32	C33	C245	C246	C34	C247	C248
			C249	C250	C251	C252	C253	C254	C255	C256
	C	正向硬體行程極限	C50	C52	C54	C140	C142	C56	C144	C146
			C148	C150	C152	C154	C156	C158	C160	C162
	C	負向硬體行程極限	C51	C53	C55	C141	C143	C57	C145	C147
			C149	C151	C153	C155	C157	C159	C161	C163
	C	PLC軸控制旗標	C66	C67	C68	C260	C261	C69	C262	C263
			C264	C265	C266	C267	C268	C269	C270	C271
	C	原點開關訊號旗標	C79	C80	C81	C200	C201	C82	C202	C203
			C204	C205	C206	C207	C208	C209	C210	C211
	S	軸忙碌旗標	S6	S7	S8	S155	S156	S9	S157	S158
			S159	S160	S161	S162	S163	S164	S165	S166
功 能	方 向	說 明	1	2	3	4	5	6	7	8
			9	10	11	12	13	14	15	16

軸向旗標與暫存器	S	軸向尋原點完成旗標	S16	S17	S18	S140	S141	S19	S142	S143
			S144	S145	S146	S147	S148	S149	S150	S151
	S	伺服警報狀態旗標	S171	S172	S173	S174	S175	S176	S177	S178
			S179	S180	S181	S182	S183	S184	S185	S186
	R	各軸向程式座標	R721 (R26)	R722 (R27)	R723 (R28)	R724	R725	R726 (R29)	R727	R728
			R729	R730	R731	R732	R733	R734	R735	R736
	R	各軸向機械座標	R741 (R31)	R742 (R32)	R743 (R33)	R744	R745	R746 (R34)	R747	R748
			R749	R750	R751	R752	R753	R754	R755	R756
	C	單軸向機械鎖定	R600. 1	R600. 2	R600. 3	R600. 4	R600. 5	R600. 6	R600. 7	R600. 8
			R600. 9	R600. 10	R600. 11	R600. 12	R600. 13	R600. 14	R600. 15	R600. 16
	C	單軸手輪模擬 (PLC軸)	R601. 1	R601. 2	R601. 3	R601. 4	R601. 5	R601. 6	R601. 7	R601. 8
			R601. 9	R601. 10	R601. 11	R601. 12	R601. 13	R601. 14	R601. 15	R601. 16
	S	G31跳脫位置已記錄	R610. 1	R610. 2	R610. 3	R610. 4	R610. 5	R610. 6	R610. 7	R610. 8
			R610. 9	R610. 10	R610. 11	R610. 12	R610. 13	R610. 14	R610. 15	R610. 15
	S	軸向移動旗標	R611. 1	R611. 2	R611. 3	R611. 4	R611. 5	R611. 6	R611. 7	R611. 8
			R611. 9	R611. 10	R611. 11	R611. 12	R611. 13	R611. 14	R611. 15	R611. 16
	C	機械正方向鎖定旗標	R612. 1	R612. 2	R612. 3	R612. 4	R612. 5	R612. 6	R612. 7	R612. 8
			R612. 9	R612. 10	R612. 11	R612. 12	R612. 13	R612. 14	R612. 15	R612. 16

	C	機械負方向鎖定旗標	R613.1	R613.2	R613.3	R613.4	R613.5	R613.6	R613.7	R613.8
			R613.9	R613.10	R613.11	R613.12	R613.13	R613.14	R613.15	R613.16
	S	PLC軸M碼讀取旗標	R629.1	R629.2	R629.3	R629.4	X	X	X	X
			X	X	X	X	X	X	X	X
功能	方向	說 明	1	2	3	4	5	6	7	8
			9	10	11	12	13	14	15	16
軸向旗標與暫存器	C	PLC軸M碼完成旗標	R630.1	R630.2	R630.3	R630.4	X	X	X	X
			X	X	X	X	X	X	X	X
	R	PLC軸M碼內容	R681	R682	R683	R684	R685	R686	R687	R688
			R689	R690	R691	R692	R693	R694	R695	R696
	R	PLC軸G00/G01速率段數	R661	R662	R663	R664	R665	R666	R667	R668
			R669	R670	R671	R672	R673	R674	R675	R676
	R	各軸手輪手動比例常數	R641	R642	R643	R644	R645	R646	R647	R648
			R649	R650	R651	R652	R653	R654	R655	R656
	R	各軸單軸速度	R701	R702	R703	R704	R705	R706	R707	R708
			R709	R710	R711	R712	R713	R714	R715	R716
	R	斷點機械座標位置	R861	R862	R863	R864	R865	R866	R867	R868
			R869	R870	X	X	X	X	X	X
	R	各軸溫度補償量	R901	R902	R903	R904	R905	R906	R907	R908
			R909	R910	R911	R912	R913	R914	R915	R916
	R	原點柵格量	R961	R962	R963	R964	R965	R966	R967	R968
			R969	R970	R971	R972	R973	R974	R975	R976
	R	各軸絕對位置記錄器	R981	R982	R983	R984	R985	R986	R987	R988
			R989	R990	R991	R992	R993	R994	R995	R996

R	各軸追隨誤差量	R5001	R5002	R5003	R5004	R5005	R5006	R5007	R5008
		R5009	R5010	R5011	R5012	R5013	R5014	R5015	R5016

3.5.2 MST通道與多軸群

功能	方向	說明	通道1	通道2	通道3	通道4
MST 通道與多軸群	C	M/S/T碼完成訊號	C38	R615.0	R619.0	R623.0
	S	單節中G碼完成訊號	S30	R615.1	R619.1	R623.1
	S	M碼讀取 (同時多M碼的第1個M碼)	S29	R615.2 (R615.24)	R619.2 (R619.24)	R623.2 (R623.24)
	S	M碼讀取 (同時多M碼的第2個M碼)	S11	R615.25	R619.25	R623.25
	S	M碼讀取 (同時多M碼的第3個M碼)	S12	R615.26	R619.26	R623.26
	S	M碼讀取 (同時多M碼的第4個M碼)	S13	R615.27	R619.27	R623.27
	S	M碼讀取 (同時多M碼的第5個M碼)	S14	R615.28	R619.28	R623.28
	S	S碼讀取	S54	R615.3	R619.3	R623.3
	S	T碼讀取	S69	R615.4	R619.4	R623.4
	S	第1軸群發出MST旗標	R589.1	R615.8	R619.8	R623.8
	S	第2軸群發出MST旗標	R589.2	R615.9	R619.9	R623.9
	S	第3軸群發出MST旗標	R589.3	R615.10	R619.10	R623.10
	S	第4軸群發出MST旗標	R589.4	R615.11	R619.11	R623.11
	S	第1主軸對應軸群MST旗標	R589.11	R615.16	R619.16	R623.16
	S	第2主軸對應軸群MST旗標	R589.12	R615.17	R619.17	R623.17
	S	第3主軸對應軸群MST旗標	R589.13	R615.18	R619.18	R623.18

	S	第4主軸對應軸群MST旗標	R589.14	R615.19	R619.19	R623.19
	R	M碼內容 (同時多M碼的第1個M碼)	R1 (R2050)	R616 (R2060)	R620 (R2070)	R624 (R2080)
	R	M碼內容 (同時多M碼的第2個M碼)	R2051	R2061	R2071	R2081
	R	M碼內容 (同時多M碼的第3個M碼)	R2052	R2062	R2072	R2082
	R	M碼內容 (同時多M碼的第4個M碼)	R2053	R2063	R2073	R2083
	R	M碼內容 (同時多M碼的第5個M碼)	R2054	R2064	R2074	R2084
功 能	方 向	說 明	通道1	通道2	通道3	通道4
M S T 通 道 與 多 軸 群	R	S碼內容	R2	R617	R621	R625
	R	T碼內容	R3	R618	R622	R626
	R	多軸群主軸選擇	R791	R792	R793	R794
	R	軸群程式停止原因	R841	R842	R843	R844
	R	主軸最大切削速度	R5021	R5022	R5023	R5024
	R	多軸群CNC主系統執行模式	R21	同左	同左	同左
	R	B碼內容	R5	不支援	不支援	不支援
	S	B碼讀取	S53	不支援	不支援	不支援

SYNTEC