

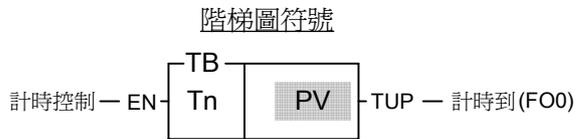
第 6 章：基本應用指令

T	6-2
C	6-5
SET	6-8
RST	6-10
0 : MC	6-12
1 : MCE	6-14
2 : SKP	6-15
3 : SKPE	6-17
4 : DIFU	6-18
5 : DIFD	6-19
6 : BSHF	6-20
7 : UDCTR	6-21
8 : MOV	6-23
9 : MOV /	6-24
10 : TOGG	6-25
11 : (+)	6-26
12 : (-)	6-27
13 : (*)	6-28
14 : (/)	6-30
15 : (+ 1)	6-32
16 : (- 1)	6-33
17 : CMP	6-34
18 : AND	6-35
19 : OR	6-36
20 : →BCD	6-37
21 : →BIN	6-38

基本應用指令

T	一般計時器 (TIMER)	T
----------	------------------	----------

指令說明



運算元

Tn：計時器號碼，為儲存累計之計時時間（即現在值 **CV**）。

PV：(Preset Value) 為計時器之設定值。

TB：時基 (Time Base)，有 0.01S，0.1S，1S 三種。

運算元	範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K
		WX0	WY0	WM0	WS0	T0	C0	R0	R3840	R3904	R3968	R5000	D0	0
		WX240	WY240	WM1896	WS984	T255	C255	R3839	R3903	R3967	R4167	R8071	D4095	32767
	Tn					○								
	PV	○	○	○	○	○	○	○	○	○	○	○	○	○

- 計時器總數為 256 個 (T0~T255)，共有 0.01 秒、0.1 秒、1 秒三種時基 (Time Base)。PLC 在出廠時已將這三種時基之計時器之個數、編號分配如下：【若這不符您的需要可利用“架構配置”(CONFIGURATION)功能自行調整】。
 - T0~T49：0.01 秒計時器 (設定值 0.00~327.67 秒)。
 - T50~T199：0.1 秒計時器 (設定值 0.0~3276.7 秒)。
 - T200~T255：1 秒計時器 (設定值 0~32767 秒)。
- **FP-08** 在您鍵入計時器號碼後，能依“架構配置”之分配自動檢知該計時器之時基，並將之顯示於 LCD 畫面之右上方。而計時器之計時時間=時基×設定值。如下圖例 1，T0 為 0.01 秒時基，而 PV 值為 1000，故 T0 之計時時間=0.01 秒×1000=10.00 秒。
- **PV** 若為暫存器，則計時時間=時基×暫存器內容值，只要變化暫存器內容值即可改變該計時器之計時時間，請參考例 2。
- ※ 計時器之最大誤差為一個時基加一個掃描時間，為了減少應用上之計時誤差，請儘量使用時基愈小之計時器。

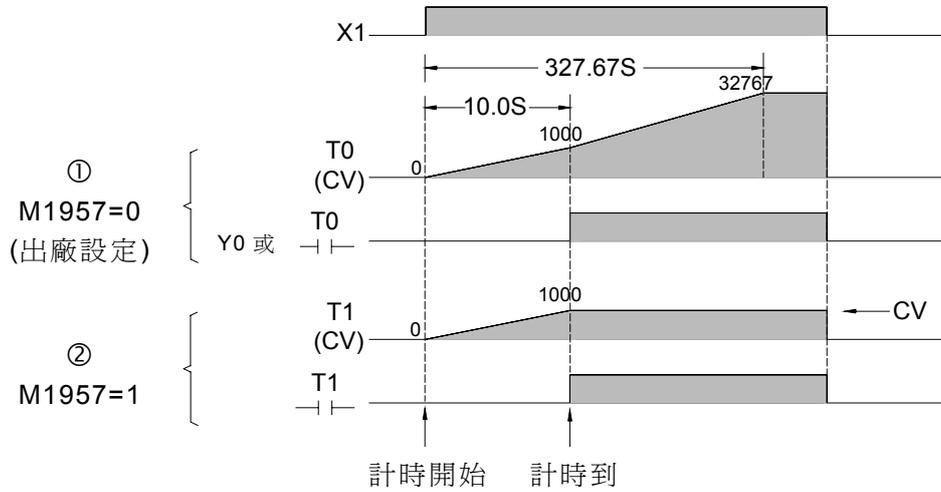
功能敘述

- 當計時控制“EN”為 1 時，計時器開始計時 (現在值由 0 開始累加) 直至“計時到”(Time Up，即現在值≥設定值) 後，該計時器之 Tn 接點和計時到旗號 TUP (FO0) 均會變為 1，表示計時到。只要計時控制“EN”輸入一直保持 1，即使計時器 Tn 之現在值 CV 已到達或超過設定值，計時器之現在值 CV 仍會持續累加計時 (M1957=0 時)，一直累積到最上限 (32767) 為止，而 Tn 接點狀態和旗號則只要 CV≥PV 就會為 1，除非 EN 輸入為 0。當計時控制“EN”為 0 時，Tn 現在值 CV 會立刻被清為 0，同時 Tn 接點和計時到旗號 TUP 均將變回 0 (請參閱下圖①)。
- **FBs** 主機可控制 M1957 為 1，使計時器“計時到”時，現在值 CV 不再累加而保持在設定值。M1957 之出廠設定為 0，您可在程式中任一計時器指令執行之前，設入 M1957 之狀態，而能個別設定該計時器在“計時到”後 CV 為繼續計時累加，或保持在設定值 (請參閱下圖②)。

T	一般計時器 (TIMER)	T
---	------------------	---

程式範例 1	固定時間計時器
--------	---------

階梯圖	按鍵操作	簡碼指令
<div style="border: 1px solid black; border-radius: 50%; width: fit-content; margin: 10px auto; padding: 5px; text-align: center;"> “計時到”信號直接由 FO0 取出之範例。 </div>		<pre> ORG X 1 T0 PV: 1000 FO 0 OUT Y 0 ORG SHORT SET M 1957 ORG X 1 T1 PV: 1000 </pre>

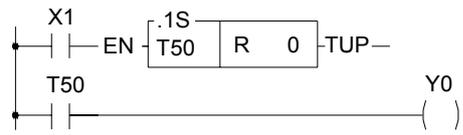
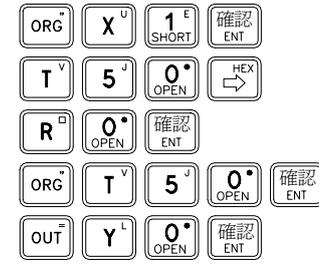


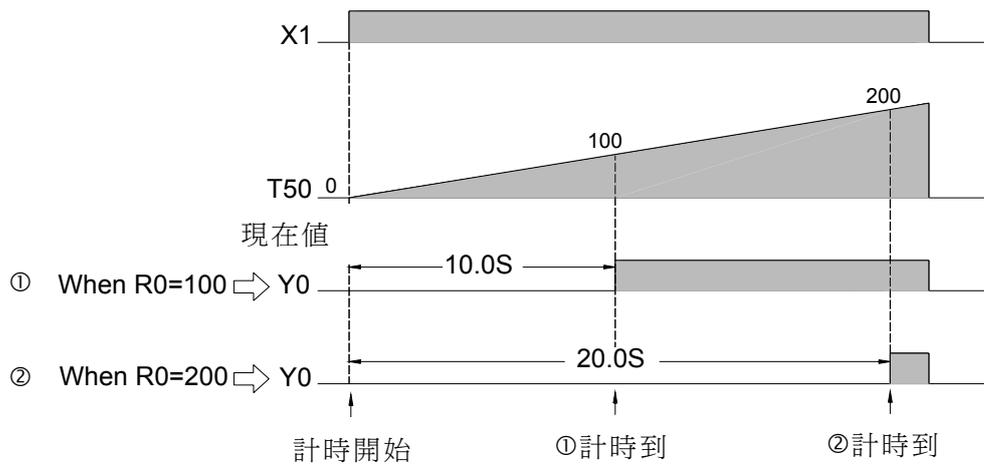
程式範例 2	可變時間計時器
--------	---------

上例之計時器設定值 PV=1000 為常數，故一旦程式輸入完成後即固定無法改變。但在許多應用場合，計時器之計時時間需要隨時地動態改變。欲達成此要求，可將 PV 改成暫存器 (R 或 D …… 等)，再改變暫存器之內容值，則可動態地改變計時時間。如本例 R0 值若設為 100，則 T 為 10 秒計時器，若將 R0 值變成 200，則 T 變為 20 秒之計時器。如此您可很容易地在 PLC 運轉 (RUN) 中動態地改變計時時間。

基本應用指令

T	一般計時器 (TIMER)	T
---	------------------	---

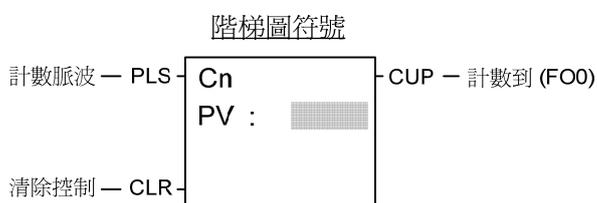
階梯圖	按鍵操作	簡碼指令
 <p style="text-align: center; border: 1px solid black; border-radius: 50%; padding: 10px; margin: 10px auto; width: 80%;">“計時到”信號間 接由 T50 接點取出 之範例。</p>		<pre> ORG X 1 T 50 PV: R 0 ORG T 50 OUT Y 0 </pre>



註：計時器之設定值 PV 若為 0，則此計時器在 PLC 一開始 RUN 時計時到輸出接點立即為 1（但 EN 輸入必須為 1），且一直為 1，而不管 CV 值如何變化，直到 EN 輸入變為 0 止。

C	一般計數器 (COUNTER) (16 位元：C0~C199，32 位元：C200~C255)	C
----------	--	----------

指令說明



運算元

Cn：計數器號碼，為儲存累計之計數值（即現在值 CV）。
 PV：(Preset Value) 為計數器之設定值。

運算元	範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K
		WX0	WY0	WM0	WS0	T0	C0	R0	R3840	R3904	R3968	R5000	D0	0
		WX240	WY240	WM1896	WS984	T255	C255	R3839	R3903	R3967	R4167	R8071	D4095	2147483647
	Cn						○							
	PV	○	○	○	○	○	○	○	○	○	○	○	○	○

- C0~C199 為 16 位元計數器（共 200 個），其設定值可為 0~32767 次。其中 C0~C139 為保持型（斷電後再開或 PLC STOP 後再 RUN 計數值仍保留），C140~C199 為非保持型（斷電再開或 PLC STOP 後再 RUN 計數值變為 0）。
- C200~C255 為 32 位元計數器（共 56 個），其設定值可為 0~2147483647 次。其中 C200~C239 為保持型，C240~C255 為非保持型。
- 上列 16 位元及 32 位元計數器之保持／非保持之個數分配為出廠之原始設定，若這不符合您的需求，可利用“架構配置”功能自行調整。
- 為確保 C0~C255 能夠正確計數，計數脈波之狀態為 1 時或為 0 時，皆必須大於一個掃描時間，否則會造成計數不正確。
- 本指令所能計數之最高頻率大概祇能 20Hz 以下；高速計數必須使用軟體或硬體之高速計數器。

功能敘述

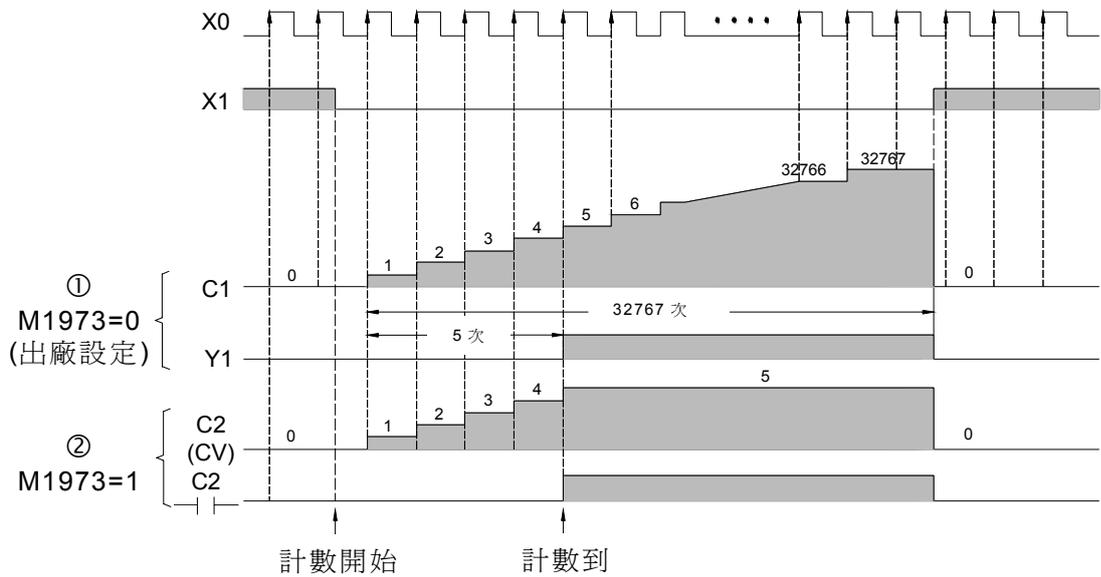
- 當清除控制“CLR”為 1 時，Cn 計數器之現在值 CV 和 Cn 接點以及計數到旗號 CUP(FO0) 均將被清為 0，計數器無法計數。
- 當“CLR”=0 時，計數器允許計數，因為計數器指令本質上為“P 指令”，因此只有在計數脈波“PLS”由 0→1 時，計數器 Cn 之現在值 CV 才會加 1，直至“計數到”（Count up，即 CV 值 ≥ 設定值）後，該計數器之計數到接點 Cn 及計數到旗號 CUP(FO0) 均會變成 1。若此時仍有計數脈波輸入，Cn 之現在值 CV 會超越設定值繼續累加（M1973=0 時），一直到最上限（32767 或 2147483647）為止，而其 Cn 接點和計數到旗號 CUP 則只要 CV ≥ PV，就會一直為 1，除非清除控制 CLR 輸入變為 1（請參考下圖①）。
- FBs 主機可控制 M1973 為 1，使計數器“計數到”時，現在值 CV 不再累加、而保持在設定值。M1973 之出廠設定為 0，您可在程式中任一計數器指令執行前，設入 M1973 之狀態，而能個別設定該計數器在“計數到”後其 CV 為繼續計數累加，或保持在設定值（請參閱下圖②）。

基本應用指令

C	一般計數器 (COUNTER) (16 位元 : C0~C199, 32 位元 : C200~C255)	C
---	---	---

程式範例 1	16 位元固定計數器
--------	------------

階梯圖	按鍵操作	簡碼指令
		ORG SHORT RST M 1973 ORG X 0 LD X 1 C 1 PV: 5 FO 0 OUT Y 1 ORG SHORT SET M 1973 ORG X 0 LD X 1 C 2 PV: 5
“計數到”信號直接由 FO0 取出之範例。		

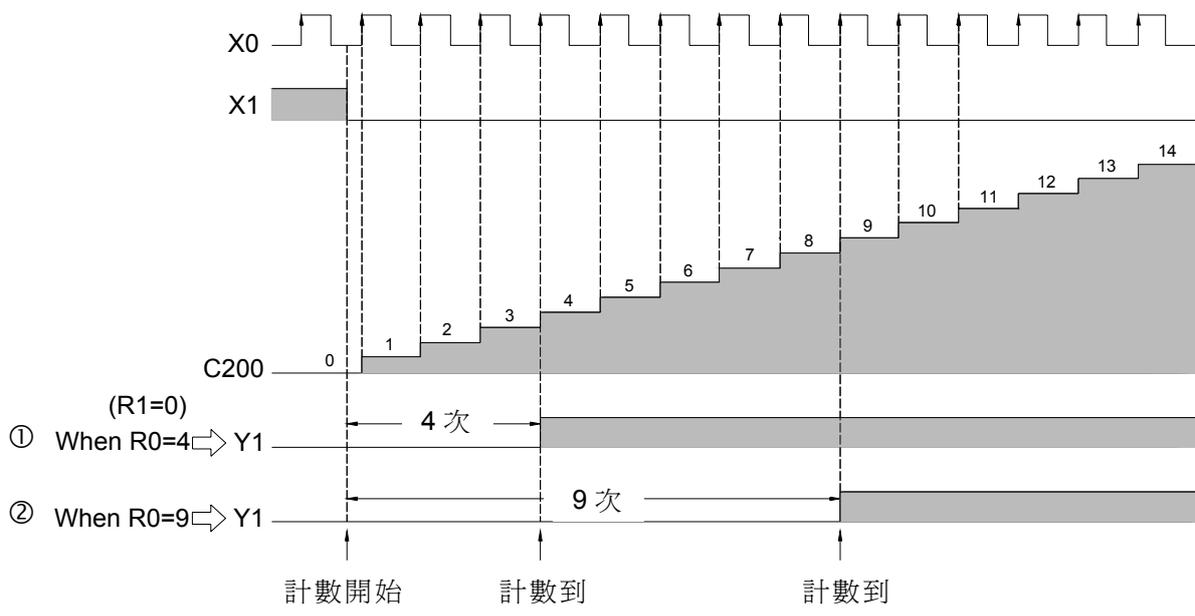


程式範例 2	32 位元可變計數器
--------	------------

如同計時器一樣，計數器之設定值 PV 若改成暫存器 (R、D.....等)，則計數器會以暫存器之內容為計數設定值。因此只要改變暫存器之內容值，即可動態地 (在 PLC RUN 中) 改變計數器之計數設定值。以下為 32 位元計數器以資料暫存器 R0 (實際上為 R1 與 R0 組成 32 位元設定值) 為設定值之範例。

C	一般計數器 (COUNTER) (16 位元 : C0~C199, 32 位元 : C200~C255)	C
---	---	---

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 LD X 1 C200 [PV:] R 0 ORG C 200 OUT Y 1 </pre>



註：計數器之設定值 PV 若為 0，則 PLC 一開始 RUN 其計數到輸出接點立刻為 1 (但必須 CLR 輸入為 0)，且一直為 1 而不管 CV 值如何變化，直到 CLR 輸入為 1 才會清除。

基本應用指令

SET D P	設定 (SET) (將線圈或暫存器之所有位元設為 1)	SET D P
-----------------------	--------------------------------	-----------------------

指令說明

運算元

階梯圖符號

設定控制 — EN

DP	SET	D
----	-----	---

D：設定之對象（線圈或暫存器號碼）。

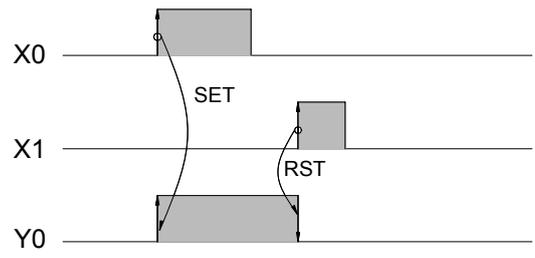
運算元 範圍	Y	M	SM	S	WY	WM	WS	TMR	CTR	HR	OR	SR	ROR	DR
	Y0 Y255	M0 M1911	M1912 M2001	S0 S999	WY0 WY240	WM0 WM1896	WS0 WS984	T0 T255	C0 C255	R0 R3839	R3904 R3967	R3968 R4167	R5000 R8071	D0 D4095
D	○	○	○*	○	○	○	○	○	○	○	○	○*	○*	○

功能敘述

- 當設定控制 “EN” =1 或由 0→1 (**P** 指令) 時，執行設定動作（將線圈或暫存器之所有位元設為 1）。

程式範例 1 單一線圈設定

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 SET P Y 0 ORG X 1 RST P Y 0 </pre>



基本應用指令

RST D P	清除 (RESET) (將線圈或暫存器之所有位元清為 0)	RST D P
----------------	----------------------------------	----------------

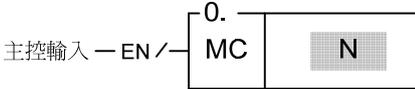
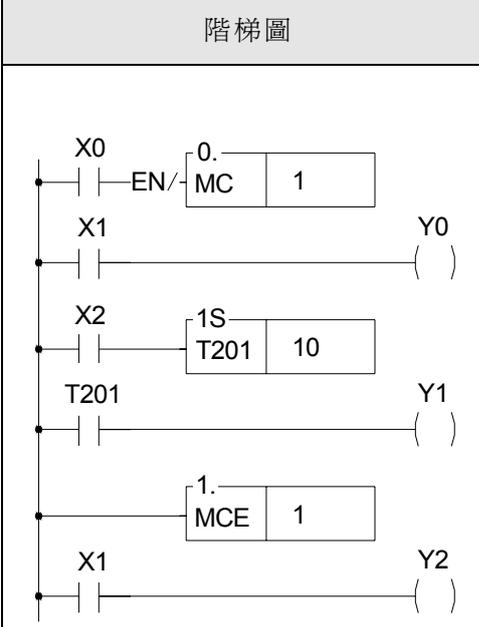
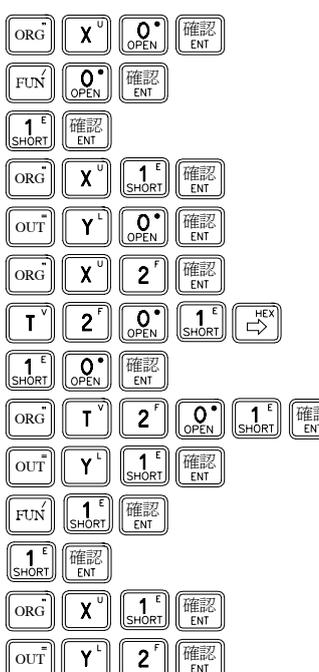
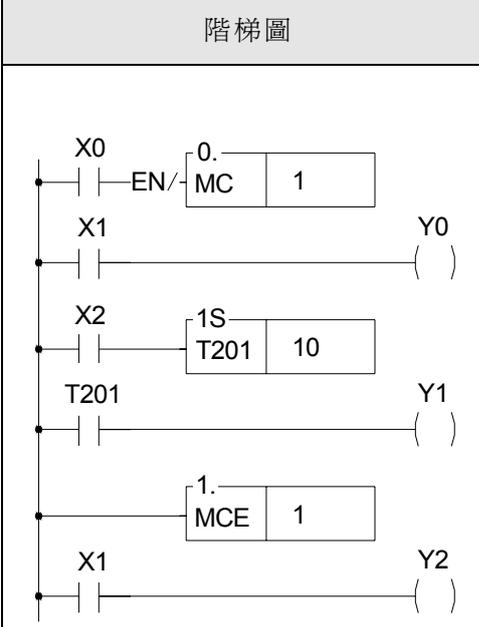
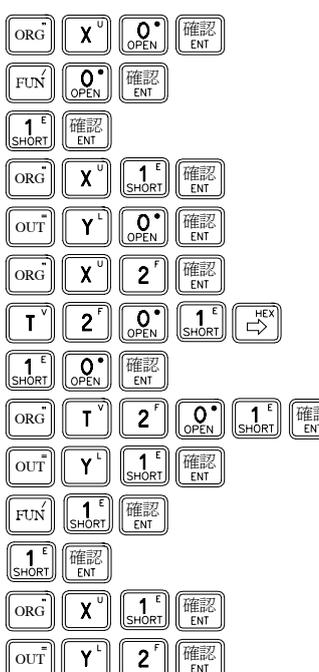
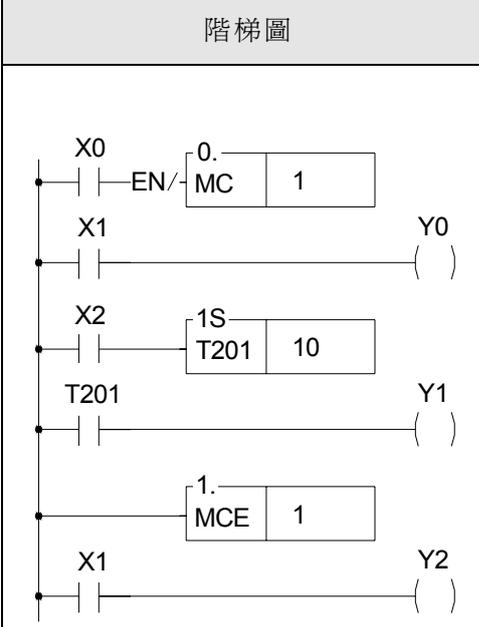
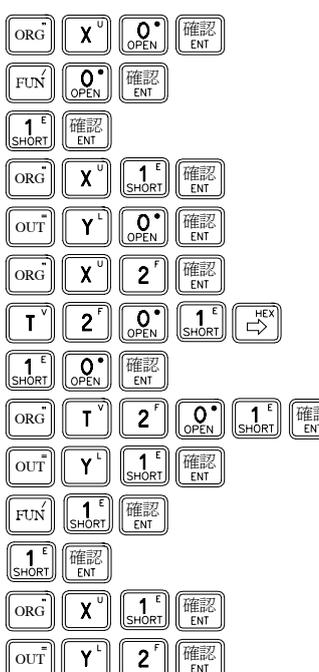
指令說明	<p style="text-align: right;">運算元</p> <p style="text-align: center;">階梯圖符號</p> <div style="display: flex; align-items: center; justify-content: center;"> <div style="margin-right: 10px;">清除控制 — EN</div> <div style="border: 1px solid black; padding: 5px; display: flex; align-items: center;"> <div style="border-right: 1px solid black; padding: 2px 5px;">RST</div> <div style="padding: 2px 5px; background-color: #cccccc; text-align: center;">D</div> </div> </div> <p style="margin-left: 200px;">D：清除之對象（線圈或暫存器號碼）</p>																																																													
	<table border="1" style="border-collapse: collapse; margin: 10px auto;"> <thead> <tr> <th style="text-align: center;">範圍</th> <th>Y</th> <th>M</th> <th>SM</th> <th>S</th> <th>WY</th> <th>WM</th> <th>WS</th> <th>TMR</th> <th>CTR</th> <th>HR</th> <th>OR</th> <th>SR</th> <th>ROR</th> <th>DR</th> </tr> </thead> <tbody> <tr> <td></td> <td>Y0</td> <td>M0</td> <td>M1912</td> <td>S0</td> <td>WY0</td> <td>WM0</td> <td>WS0</td> <td>T0</td> <td>C0</td> <td>R0</td> <td>R3904</td> <td>R3968</td> <td>R5000</td> <td>D0</td> </tr> <tr> <td></td> <td>Y255</td> <td>M1911</td> <td>M2001</td> <td>S999</td> <td>WY240</td> <td>WM1896</td> <td>WS984</td> <td>T255</td> <td>C255</td> <td>R3839</td> <td>R3967</td> <td>R4167</td> <td>R8071</td> <td>D4095</td> </tr> <tr> <td style="text-align: center;">D</td> <td style="text-align: center;">○</td> <td style="text-align: center;">○</td> <td style="text-align: center;">○*</td> <td style="text-align: center;">○</td> <td style="text-align: center;">○*</td> <td style="text-align: center;">○*</td> <td style="text-align: center;">○</td> </tr> </tbody> </table>		範圍	Y	M	SM	S	WY	WM	WS	TMR	CTR	HR	OR	SR	ROR	DR		Y0	M0	M1912	S0	WY0	WM0	WS0	T0	C0	R0	R3904	R3968	R5000	D0		Y255	M1911	M2001	S999	WY240	WM1896	WS984	T255	C255	R3839	R3967	R4167	R8071	D4095	D	○	○	○*	○	○	○	○	○	○	○	○	○*	○*	○
範圍	Y	M	SM	S	WY	WM	WS	TMR	CTR	HR	OR	SR	ROR	DR																																																
	Y0	M0	M1912	S0	WY0	WM0	WS0	T0	C0	R0	R3904	R3968	R5000	D0																																																
	Y255	M1911	M2001	S999	WY240	WM1896	WS984	T255	C255	R3839	R3967	R4167	R8071	D4095																																																
D	○	○	○*	○	○	○	○	○	○	○	○	○*	○*	○																																																

功能敘述	<p>● 當清除控制 “EN” =1 或由 0→1 (P 指令) 時，將 D (線圈或暫存器) 之所有位元清除為 0。</p>
------	---

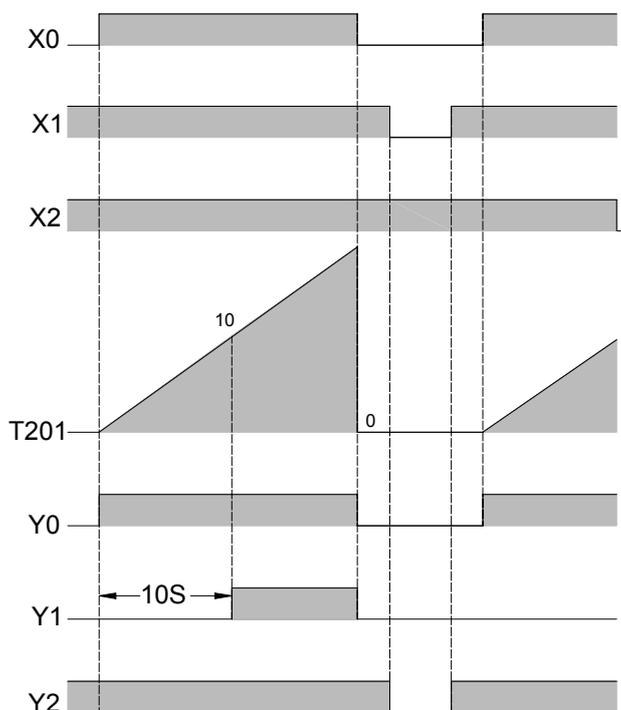
程式範例 1	<p>單一線圈清除例</p> <p>請參考第 7-8 頁 SET 指令程式範例 1 之說明。</p>
--------	--

程式範例 2	<p>16 位元暫存器清除例</p>					
<table border="1" style="width:100%; border-collapse: collapse;"> <thead> <tr> <th style="width:33%;">階梯圖</th> <th style="width:33%;">按鍵操作</th> <th style="width:33%;">簡碼指令</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;"> </td> <td style="text-align: center;"> </td> <td style="text-align: center;"> <pre>ORG X 0 RST P R 0</pre> </td> </tr> </tbody> </table>	階梯圖	按鍵操作	簡碼指令			<pre>ORG X 0 RST P R 0</pre>
階梯圖	按鍵操作	簡碼指令				
		<pre>ORG X 0 RST P R 0</pre>				

基本應用指令

FUN 0 MC	主控 (MASTER CONTROL) 回路開始指令	FUN 0 MC						
指令說明	<div style="display: flex; justify-content: space-between;"> <div style="text-align: center;"> <p>階梯圖符號</p>  </div> <div style="text-align: center;"> <p>運算元</p> <p>N: 主控回路之號碼 (N=0~127), 號碼不得重複。</p> </div> </div>							
功能敘述	<ul style="list-style-type: none"> ● MC 回路總共可有 128 個 (N=0~127)。每個主控回路 MC N 指令均要有一個相同號碼之主控回路終止指令 MCE N 與之對應 (但需確保 MCE N 指令要在 MC N 指令之後)。 ● 當主控輸入 "EN/" 為 1 則此指令不執行 (等效 MC N 指令不存在)。 ● 當主控輸入 "EN/" 為 0 則主控回路動作, 由 MC N 指令開始, 一直到相同號碼之 MCE N 指令間 (稱之為主控回路動作區) 之程式, 若為 OUT 線圈或一般計時器將其狀態均清為 0, 其他指令則不執行。 							
程式範例	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 33%;">階梯圖</th> <th style="width: 33%;">按鍵操作</th> <th style="width: 33%;">簡碼指令</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">  </td> <td style="text-align: center;">  </td> <td style="vertical-align: top;"> <pre> ORG X 0 FUN 0 N: 1 ORG X 1 OUT Y 0 ORG X 2 T201 PV: 10 ORG T 20 OUT Y 1 FUN 1 1 N: 1 ORG X OUT Y 1 </pre> </td> </tr> </tbody> </table>		階梯圖	按鍵操作	簡碼指令			<pre> ORG X 0 FUN 0 N: 1 ORG X 1 OUT Y 0 ORG X 2 T201 PV: 10 ORG T 20 OUT Y 1 FUN 1 1 N: 1 ORG X OUT Y 1 </pre>
階梯圖	按鍵操作	簡碼指令						
		<pre> ORG X 0 FUN 0 N: 1 ORG X 1 OUT Y 0 ORG X 2 T201 PV: 10 ORG T 20 OUT Y 1 FUN 1 1 N: 1 ORG X OUT Y 1 </pre>						

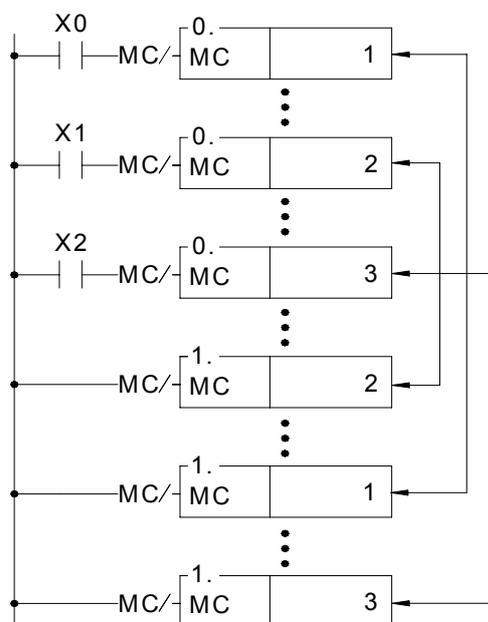
FUN 0 MC	主控 (MASTER CONTROL) 回路開始指令	FUN 0 MC
-------------	----------------------------	-------------



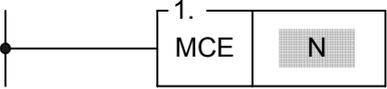
註 1：MC/MCE 指令可作多層巢狀或交錯使用。如右例：

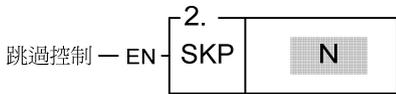
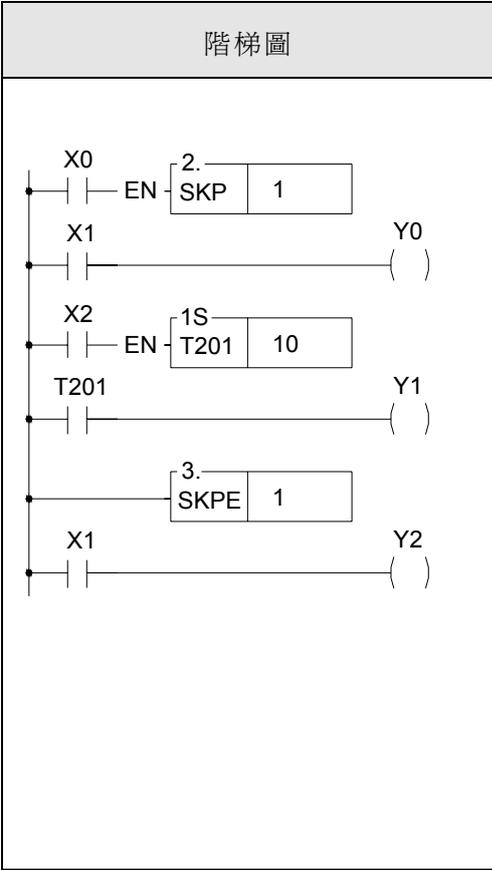
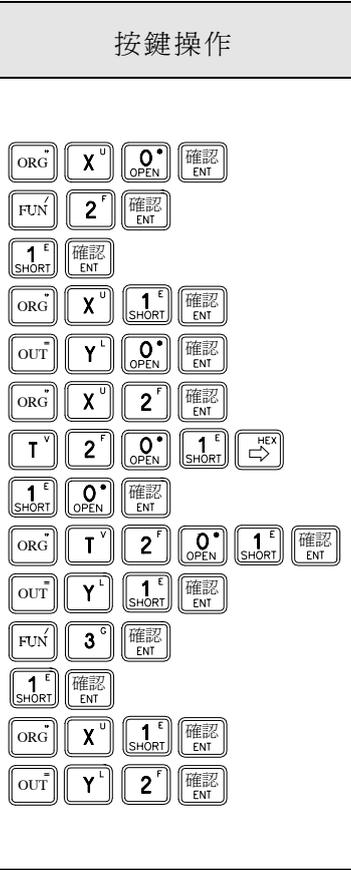
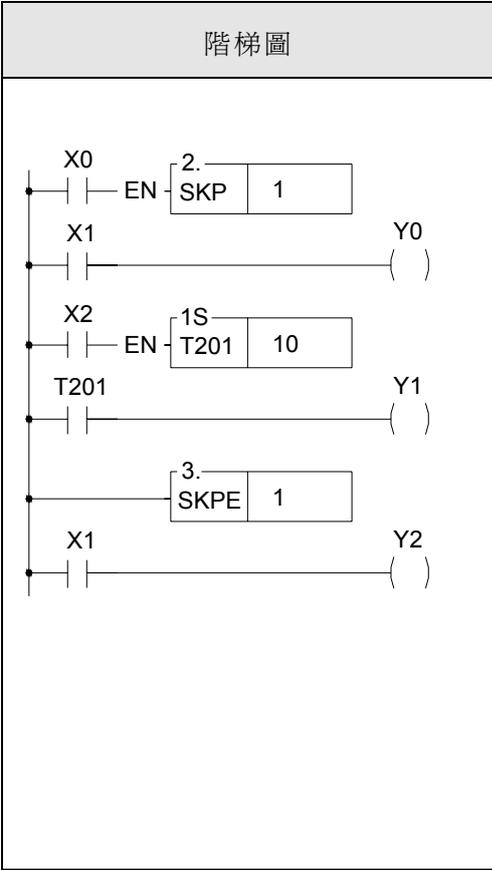
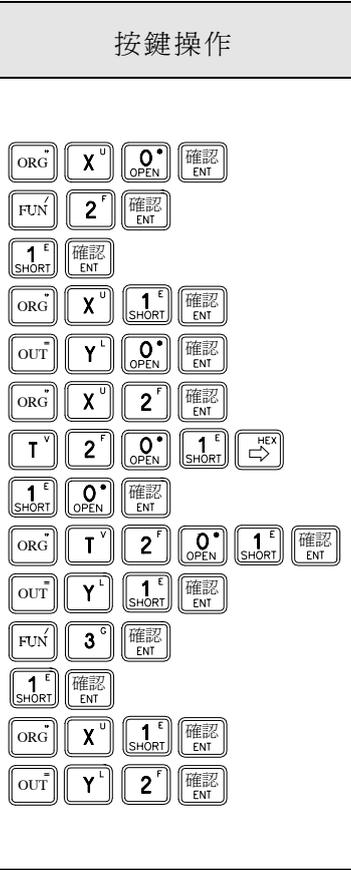
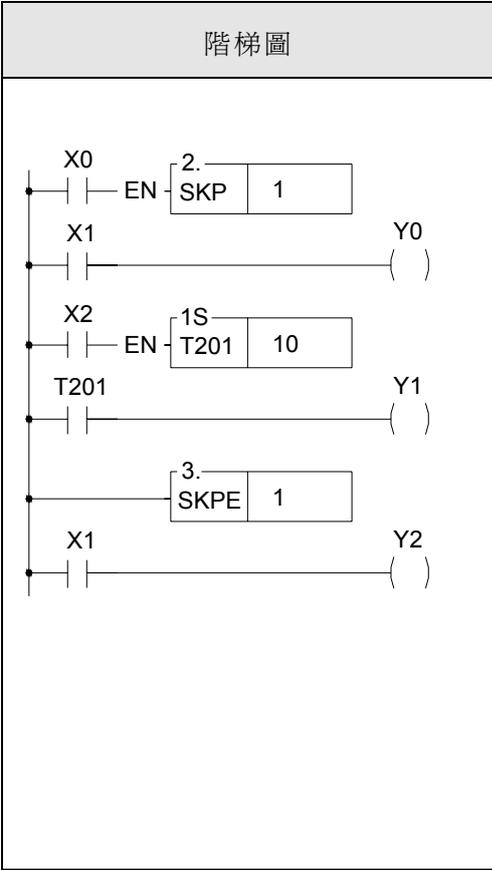
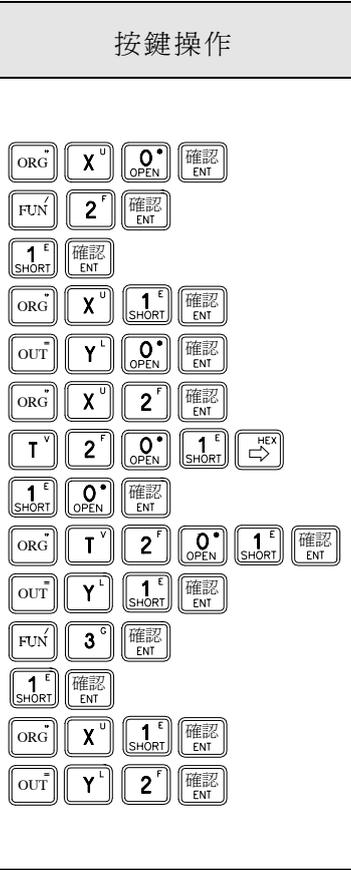
註 2：

- 當 M1918=0，主控輸入有 0→1 變化時，如主控回路內有 Pulse 型功能指令，則這些 Pulse 型功能指令僅會在主控輸入之第一次 0→1 變化時有機會被執行一次；其後不管主控輸入 0→1 變化多少次，在主控回路內之 Pulse 型功能指令皆不會再執行。
- 當 M1918=1，主控輸入有 0→1 變化時，如主控回路內有 Pulse 型功能指令，則每次主控輸入有 0→1 變化，在主控回路內的這些 Pulse 型功能指令只要動作條件滿足皆會被執行。
- 當主控回路內有計數指令時，控制 M1918=0，可避免錯誤的計數。
- 當主控回路內之 Pulse 型功能指令必須與主控輸入之 0→1 變化同動時，則控制 M1918=1 可達成。

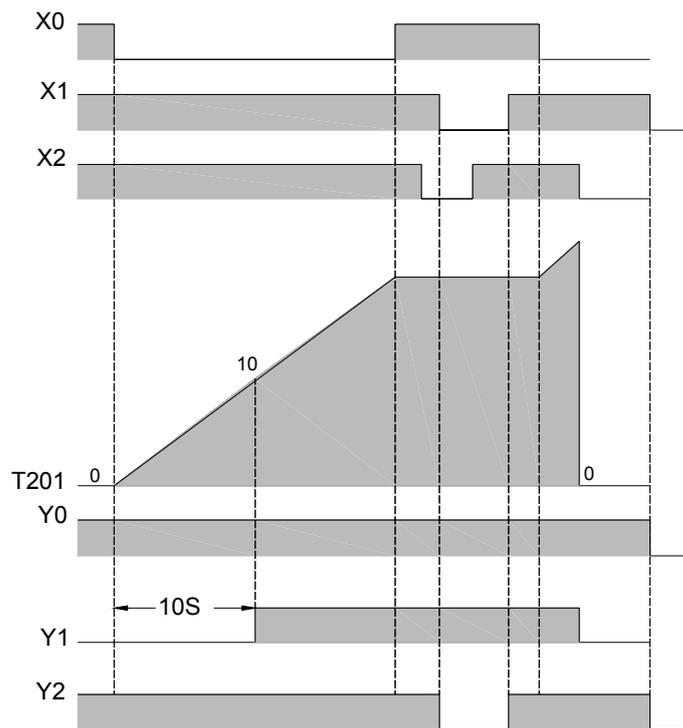


基本應用指令

FUN 1 MCE	主控終止 (MASTER CONTROL END) 指令	FUN 1 MCE
指令說明	<div style="display: flex; justify-content: space-between;"> <div style="text-align: center;"> <p>階梯圖符號</p>  </div> <div style="text-align: right;"> <p>運算元</p> <p>N：終止之主控回路號碼 (N=0~127)， 號碼不得重複。</p> </div> </div>	
功能敘述	<ul style="list-style-type: none"> ● MCE N 是搭配 MC N 使用，單獨存在並無意義。在 MC N 指令動作後，其後之程式若為 OUT 線圈或一般計時器狀態均被清為 0，其他指令則不執行，直到遇到相同號碼 (N) 之 MCE 指令，才會解除主控動作，恢復正常之程式執行動作。 ● MCE 指令無需輸入控制，其本身自成一個網路，不能串接其他指令，在程式執行中只要遇到 MCE N 指令，若已發生 MC N 主控動作則此指令會將主控動作解除，若未發生則此指令無效 (無任何影響)。 	
程式範例	<ul style="list-style-type: none"> ● 請參閱 MC 指令之範例說明。 	

FUN 2 SKP	跳過 (SKIP) 回路之開始指令	FUN 2 SKP						
指令說明	<div style="display: flex; justify-content: space-between; align-items: center;"> <div style="text-align: center;"> <p>階梯圖符號</p>  </div> <div style="text-align: center;"> <p>運算元</p> <p>N：跳過回路之號碼 (N=0~127)，號碼不得重複。</p> </div> </div>							
功能敘述	<ul style="list-style-type: none"> ● SKP 回路共有 128 個 (N=0~127)。每個跳過回路開始指令 SKP N 至少要有一個相同號碼之跳過回路終止指令 SKPE N 與之對應(但需確保 SKPE N 指令要在 SKP N 指令之後)。 ● 當跳過控制“EN”為 0，此指令不執行 (等效 SKP N 指令不存在)。 ● 當跳過控制“EN”為 1 則跳過回路動作，在 SKP N 以後一直到遇到相同號碼之 SKPE N 指令間 (稱之為跳過回路動作區) 之程式均不執行 (跳過)。故此區域內所有之單點或暫存器狀態均保持不變。 							
程式範例	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 33%;">階梯圖</th> <th style="width: 33%;">按鍵操作</th> <th style="width: 33%;">簡碼指令</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">  </td> <td style="text-align: center;">  </td> <td style="vertical-align: top;"> <pre> ORG X 0 FUN 2 N: 1 ORG X 1 OUT Y 0 ORG X 2 T201 PV: 10 ORG T 201 OUT Y 1 FUN 3 N: 1 ORG X 1 OUT Y 2 </pre> </td> </tr> </tbody> </table>		階梯圖	按鍵操作	簡碼指令			<pre> ORG X 0 FUN 2 N: 1 ORG X 1 OUT Y 0 ORG X 2 T201 PV: 10 ORG T 201 OUT Y 1 FUN 3 N: 1 ORG X 1 OUT Y 2 </pre>
階梯圖	按鍵操作	簡碼指令						
		<pre> ORG X 0 FUN 2 N: 1 ORG X 1 OUT Y 0 ORG X 2 T201 PV: 10 ORG T 201 OUT Y 1 FUN 3 N: 1 ORG X 1 OUT Y 2 </pre>						

<p>FUN 2 SKP</p>	<p>跳過 (SKIP) 回路之開始指令</p>	<p>FUN 2 SKP</p>
----------------------	--------------------------	----------------------



FUN 3 SKPE	跳過回路終止 (SKIP END) 指令	FUN 3 SKPE
指令說明	<p style="text-align: right;"><u>運算元</u></p> <p style="text-align: center;">階梯圖符號</p>  <p style="text-align: right;">N：終止之跳過回路號碼 (N=0~127)， 號碼不得重複。</p>	
功能敘述	<ul style="list-style-type: none"> ● SKPE N 指令係搭配 SKP N 指令使用，單獨存在並無意義。在 SKP N 跳過指令動作後，其後之程式即跳過不執行，所有狀態亦不變，一直要到遇到相同號碼 (N) 之 SKPE 指令，才會解除跳過動作，恢復正常之程式執行動作。 ● SKPE 指令無需輸入控制，其本身即是一個網路，不能串接其他指令，在程式執行中只要遇到 SKPE N 指令，若已發生 SKP N 之跳過動作，則此指令會立即將跳過動作解除。若未發生，則此指令無效 (無任何影響)。 	
程式範例	<ul style="list-style-type: none"> ● 請參閱 SKP 指令之範例說明。 <p>註：SKP/SKPE 指令可作多層巢式或交錯使用，其用法規則和 MC/MCE 指令之巢式或交錯用法完全相同，請參閱 MC/MCE 指令。</p>	

基本應用指令

FUN 4 P DIFU	上微分 (DIFFERENTIAL UP) 指令	FUN 4 P DIFU
-----------------	--------------------------	-----------------

指令說明



運算元

D : 存放上微分結果之繼電器線圈號碼。

運算元	範圍	Y	M	SM	S
		Y0 Y255	M0 M1911	M1912 M2001	S0 S999
	D	○	○	○*	○

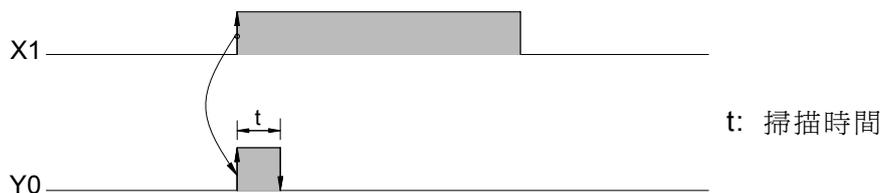
功能敘述

- DIFU 指令係將狀態輸入 "TGU" 之狀態取上微分 (在 TGU 升緣變換時產生一波寬為掃描時間 T 之單擊脈波) 後，將此單擊脈波信號存入 D 所指定之線圈。
- 本指令可以用順序指令之 TU 接點直接取代，其較本指令方便好用。

程式範例

以下兩例結果完全相同

階梯圖	按鍵操作	簡碼指令
<p>例 1</p>		<pre>ORG X 1 FUN 4 [D]: Y 0</pre>
<p>例 2</p>		<pre>ORG TU X 1 OUT Y 0</pre>



FUN 5 P DIFD	下微分 (DIFFERENTIAL DOWN) 指令	FUN 5 P DIFD
-----------------	----------------------------	-----------------

指令說明



運算元

N：存放下微分結果之繼電器線圈號碼。

運算元 \ 範圍	Y	M	SM	S
	Y0 Y255	M0 M1911	M1912 M2001	S0 S999
D	○	○	○*	○

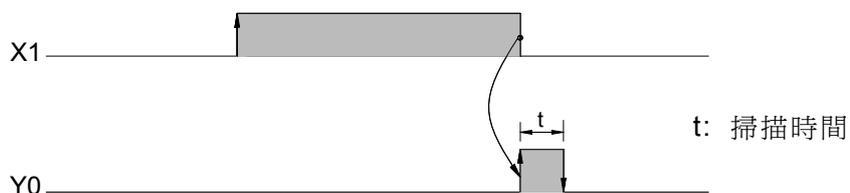
功能敘述

- DIFD 指令係將狀態輸入 "TGD" 之狀態取下微分 (在 TGD 降緣變換時產生一波寬為掃描時間 T 之單擊脈波) 後，將此單擊脈波信號存入 D 所指定之線圈內。
- 本指令可以用順序指令之 TD 接點直接取代，其較本指令方便好用。

程式範例

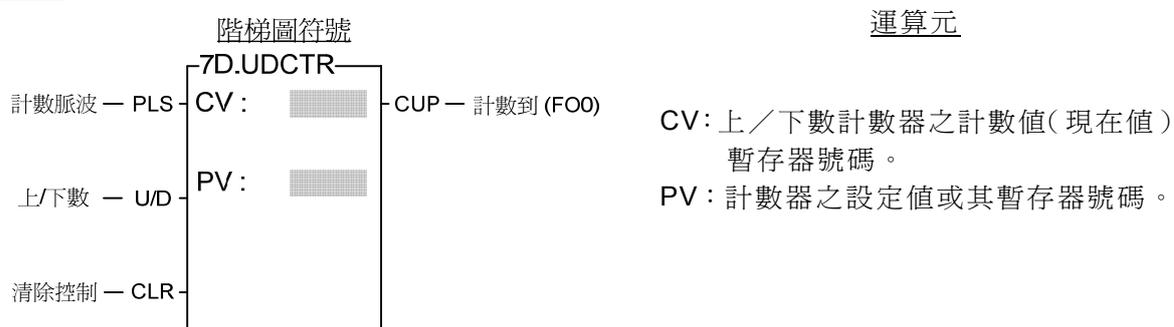
以下兩例結果完全相同

階段圖	按鍵操作	簡碼指令
<p>例 1</p>		<pre>ORG X 1 FUN 5 [D:] Y 0</pre>
<p>例 2</p>		<pre>ORG TD X 1 OUT Y 0</pre>



FUN 7 D P UDCTR	上／下數計數器 (UP/DOWN COUNTER) (16 位元或 32 位元可上數、下數之雙相計數器)	FUN 7 D P UDCTR
---------------------------	---	---------------------------

指令說明



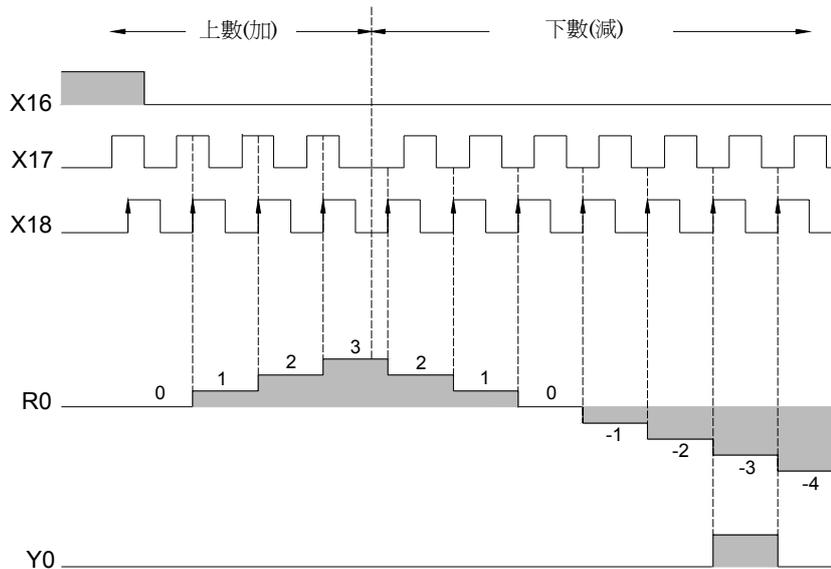
運算元	範圍													
	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	
	WX0	WY0	WM0	WS0	T0	C0	R0	R3840	R3904	R3968	R5000	D0	16 或 32 位元正、負數	
CV		○	○	○	○	○	○		○	○*	○*	○		
PV	○	○	○	○	○	○	○	○	○	○	○	○	○	

功能敘述

- 當清除控制“CLR”為 1 時，則計數器現在值 CV 清為 0，計數器無法計數。
- 當清除控制“CLR”為 0，則允許計數，本指令本質上為 P 指令，當計數脈波“PLS”由 0→1（升緣）時，計數值 CV 才會加 1（當 U/D=1 時）或減 1（當 U/D=0 時）。
- 當現在值=設定值時，FOO 計數到（Count-up）會變為 1，若再有計數脈波輸入計數器將繼續計數，使現在值≠設定值，此時 FOO 會立刻變回 0，亦即計數到信號只有在現在值=設定值時為 1，否則便為 0，（此點和一般計數器之計數到信號不同請特別注意）。
- 上數計數值之上限為 32767（16 位元）或 2147483647（32 位元），到達上限後，若再來個上數計數脈波，計數值將會變成-32768 或-2147483648（下數之最下限）。
- 下數之最下限為-32768 或-2147483648，達此值後，若再來一個下數計數脈波，則計數值會跳到 32767 或 2147483647（上數之最上限）。
- 若將 U/D 固定為 1，則本指令將變成單相上數計數器，反之若固定為 0 則變成單相下數計數器。

FUN 7 D P UDCTR	上／下數計數器 (UP/DOWN COUNTER) (16 位元或 32 位元可上數、下數之雙相計數器)	FUN 7 D P UDCTR
---	---	---

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 18 LD X 17 LD X 16 FUN 7 CV: R 0 PV: - 3 FO 0 OUT Y 0 </pre>

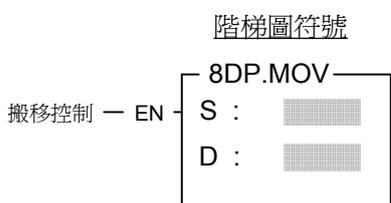


註 1：因 UDCTR 係以軟體掃描方式計數，因此若計數脈波速度高於掃描速度，就會造成漏掉之情形（一般情況計數脈波不要超過 20Hz，視程式大小而有變異）。此時請用 PLC 內部之軟體或硬體高速計數器，請參考進階功能篇手冊之高速計數器使用方法。

註 2：為確保本指令能夠正確計數，計數脈波之波寬無論為 1 或 0，皆必須大於一個掃描時間。

FUN 8 D P MOV	搬移 (MOVE) (將 S 之資料搬移到 D 去)	FUN 8 D P MOV
-------------------------	--------------------------------------	-------------------------

指令說明



運算元

S：來源 (Source) 資料或其暫存器號碼。
 D：搬移之目的 (Destination) 暫存器號碼。
 S,D 可結合 V、Z、P0~P9 作間接定址應用。

運算元 範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR
		WX0 WX240	WY0 WY240	WM0 WM1896	WS0 WS984	T0 T255	C0 C255	R0 R3839	R3840 R3903	R3904 R3967	R3968 R4167	R5000 R8071	D0 D4095	16 或 32 位元 正、負數
S	○	○	○	○	○	○	○	○	○	○	○	○	○	○
D		○	○	○	○	○	○		○	○*	○*	○		○

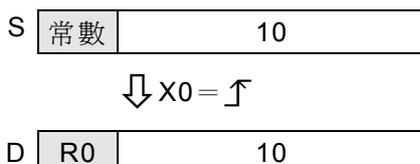
功能敘述

- 當搬移控制 "EN" =1 或由 0→1 (P 指令) 時，將 S 之資料搬移 (寫入) 到 D 去。

程式範例

16 位元暫存器搬移例

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 FUN 8 P S: 10 D: R 0 </pre>



FUN 10 TOGG	交替開關 (TOGGLE SWITCH) (輸入每觸動一次，輸出 D 轉態一次)	FUN 10 TOGG
----------------	---	----------------

指令說明

運算元

階梯圖符號

觸發輸入—TGU

10.	TOGG	D
-----	------	---

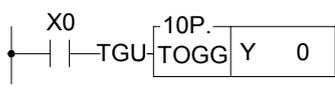
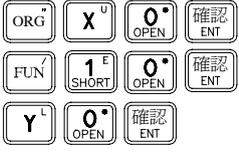
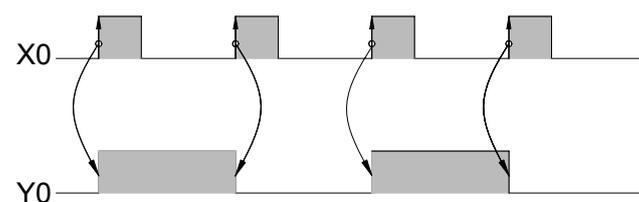
D：交替開關線圈號碼

運算元	範圍	Y	M	SM	S
		Y0 Y255	M0 M1911	M1912 M2001	S0 S999
D		○	○	○*	○

功能敘述

- 當觸發輸入“TGU”每由 0→1 一次，D 線圈之狀態就交替轉態一次（0 變 1，1 則變 0）。

程式範例

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 FUN 10 [D:] Y 0 </pre>
		

基本應用指令

FUN11 D P (+)	加法運算 (ADDITION) (將 Sa 加 Sb 之和存入 D)	FUN11 D P (+)
--------------------------------	---	--------------------------------



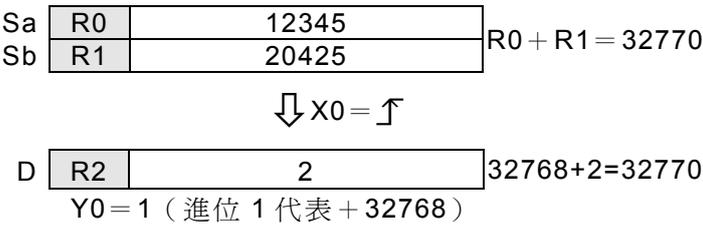
運算元	範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR
		WX0	WY0	WM0	WS0	T0	C0	R0	R3840	R3904	R3968	R5000	D0	16 或 32 位元正、負數	V、Z
Sa		○	○	○	○	○	○	○	○	○	○	○	○	○	○
Sb		○	○	○	○	○	○	○	○	○	○	○	○	○	○
D			○	○	○	○	○			○*	○*	○			○

功能敘述

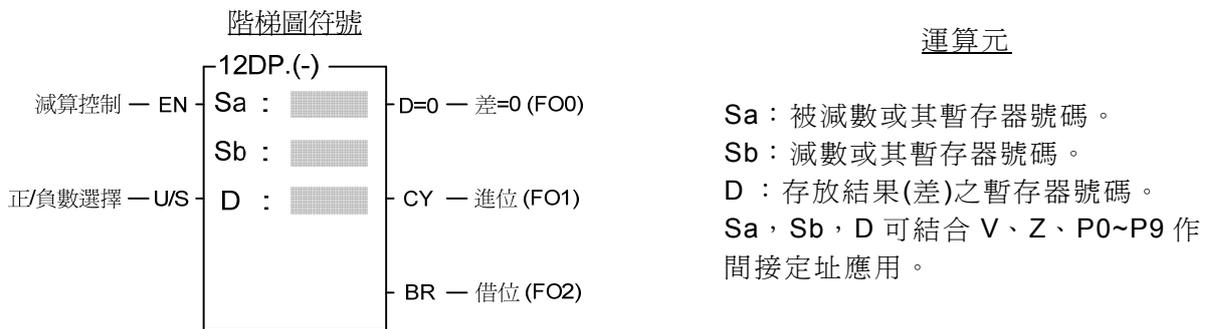
- 當加算控制 "EN" =1 或由 0→1 (P 指令) 而 "U/S" =0 時，將 Sa 與 Sb 以正負數 (Sign) 運算法則作加法運算並將結果寫入 D 去。同時若和為 0，則 FO0 (D=0) 設為 1，若發生進位 (和超過 32767 或 2147483647)，則將 FO1 (CY) 設為 1，若發生借位 (負數加負數，使和小於 -32768 或 -2147483648)，則 FO2 (BR) 設為 1。所有 FO 之狀態均維持到本指令下次執行時才被新執行結果所取代。
- 當加算控制 "EN" =1 或由 0→1 (P 指令) 而 "U/S" =1 時，將 Sa 與 Sb 以正整數 (Unsign) 運算法則作加法運算並將結果寫入 D 去。同時若和為 0，則 FO0 (D=0) 設為 1，若發生進位 (和超過 65535 或 4294967295)，則進位 FO1 (CY) 設為 1。

程式範例

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 FUN 11P Sa: R 0 Sb: R 1 D: R 2 FO 1 OUT Y 0 </pre>



FUN12 D P (-)	減法運算 (SUBTRACTION) (將 Sa 減 Sb 之和存入 D)	FUN12 D P (-)
---------------------------------------	--	---------------------------------------



運算元 \ 範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR
	WX0	WY0	WM0	WS0	T0	C0	R0	R3840	R3904	R3968	R5000	D0	16 或 32 位元正、負數	V、Z
Sa	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Sb	○	○	○	○	○	○	○	○	○	○	○	○	○	○
D		○	○	○	○	○	○		○	○*	○*	○		○

功能敘述

- 當減算控制“EN”=1 或由 0→1 (P 指令) 而“U/S”=0 時，將 Sa 與 Sb 以正負數 (Sign) 運算法則作減法運算並將結果寫入 D 去。同時若差為 0，則 FO0 (D=0) 設為 1，若發生進位 (正數減負數，使差超過+32767 或+2147483647) 則 FO1 (CY) 設為 1，若借位發生 (負數減正數，使差小於-32768 或-2147483648) 則 FO2 (BR) 設為 1。所有 FO 之狀態均維持到本指令下一次執行時才被新執行結果所取代。
- 當減算控制“EN”=1 或由 0→1 (P 指令) 而“U/S”=1 時，將 Sa 與 Sb 以正整數 (Unsign) 運算法則作減法運算並將結果寫入 D 去。同時若差為 0，則 FO0 (D=0) 設為 1，若借位發生 (Sa - Sb < 0) 則 FO2 (BR) 設為 1。

程式範例

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 FUN 12 [Sa]: R 0 [Sb]: R 1 [D]: R 2 FO 2 OUT Y 2 </pre>

Sa

R0	-5
----	----

 Sb

R1	32767
----	-------

 R0 - R1 = -32772
 ↓ X0 = 1
 D

R2	-4
----	----

 -32768 - 4 = -32772
 Y2 = 1 (借位 1 代表 -32768) 請參閱 5.5 節之說明

FUN13 D P (*)	乘法運算 (MULTIPLICATION) (將 Sa 乘 Sb 之積存入 D)	FUN13 D P (*)
--------------------------------	---	--------------------------------

階梯圖符號

運算元

Sa : 被乘數或其暫存器號碼。
 Sb : 乘數或其暫存器號碼。
 D : 存放結果(積)之暫存器號碼。
 Sa, Sb, D 可結合 V、Z、P0~P9 作間接定址應用。

運算元 \ 範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR
	WX240	WY240	WM1896	WS984	T255	C255	R3839	R3903	R3967	R4167	R8071	D4095	16 或 32 位元正、負數	V、Z P0-P9
Sa	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Sb	○	○	○	○	○	○	○	○	○	○	○	○	○	○
D		○	○	○	○	○	○		○	○*	○*	○		○

功能敘述

- 當乘算控制“EN”=1 或由 0→1(P 指令)而“U/S”=0 時，將 Sa 與 Sb 以正負數 (Sign) 運算法則作乘法運算並將結果寫入 D 去。同時若積為 0，則 FO0 (D=0) 設為 1，若積為負數則將 FO1 (D<0) 設為 1。
- 當乘算控制“EN”=1 或由 0→1(P 指令)而“U/S”=1 時，將 Sa 與 Sb 以正整數 (Unsign) 運算法則作乘法運算並將結果寫入 D 去。同時若積為 0，則 FO0 (D=0) 設為 1。

程式範例 1 16 位元乘法例

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 FUN 13 P Sa : R 0 Sb : R 1 D : R 2 </pre>

被乘數	R0						
Sa	12345						
x							
乘數 Sb	R1						
	4567						
<table style="margin-left: auto; margin-right: auto;"> <tr> <td style="padding-right: 10px;">積 D</td> <td style="border: 1px solid black; padding: 2px;">R3</td> <td style="border: 1px solid black; padding: 2px;">R2</td> </tr> <tr> <td></td> <td colspan="2" style="border: 1px solid black; padding: 2px;">56379615</td> </tr> </table>		積 D	R3	R2		56379615	
積 D	R3	R2					
	56379615						

FUN13 D P (*)	乘法運算 (MULTIPLICATION) (將 Sa 乘 Sb 之積存入 D)	FUN13 D P (*)
---	---	---

程式範例 2 32 位元乘法例

階梯圖	按鍵操作	簡碼指令
		ORG X 0 FUN 13D [Sa:] R 0 [Sb:] R 2 [D:] R 4

被乘數 Sa	<table border="1" style="display: inline-table;"> <tr><td style="width: 50px;">R1</td><td style="width: 50px;">R0</td></tr> <tr><td colspan="2" style="text-align: center;">12345678</td></tr> </table>	R1	R0	12345678							
R1	R0										
12345678											
×	乘數 Sb										
	<table border="1" style="display: inline-table;"> <tr><td style="width: 50px;">R3</td><td style="width: 50px;">R2</td></tr> <tr><td colspan="2" style="text-align: center;">456</td></tr> </table>	R3	R2	456							
R3	R2										
456											
<table style="margin-left: auto; margin-right: auto;"> <tr> <td style="padding-right: 20px;">積 D</td> <td style="border: 1px solid black; padding: 5px;"> <table border="1" style="display: inline-table;"> <tr><td style="width: 50px;">R7</td><td style="width: 50px;">R6</td><td style="width: 50px;">R5</td><td style="width: 50px;">R4</td></tr> <tr><td colspan="4" style="text-align: center;">5629629168</td></tr> </table> </td> </tr> </table>		積 D	<table border="1" style="display: inline-table;"> <tr><td style="width: 50px;">R7</td><td style="width: 50px;">R6</td><td style="width: 50px;">R5</td><td style="width: 50px;">R4</td></tr> <tr><td colspan="4" style="text-align: center;">5629629168</td></tr> </table>	R7	R6	R5	R4	5629629168			
積 D	<table border="1" style="display: inline-table;"> <tr><td style="width: 50px;">R7</td><td style="width: 50px;">R6</td><td style="width: 50px;">R5</td><td style="width: 50px;">R4</td></tr> <tr><td colspan="4" style="text-align: center;">5629629168</td></tr> </table>	R7	R6	R5	R4	5629629168					
R7	R6	R5	R4								
5629629168											

FUN14 D P (/)	除法運算 (DIVISION) (將 Sa 除以 Sb 所得之商和餘數存到 D 去)	FUN14 D P (/)
---	---	---

階梯圖符號

運算元

Sa : 被除數或其暫存器號碼。
 Sb : 除數或其暫存器號碼。
 D : 存放結果(商和餘數)之暫存器號碼。
 Sa, Sb, D 可結合 V、Z、P0~P9 作間接定址應用。

運算元 範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR
		WX0 WX240	WY0 WY240	WM0 WM1896	WS0 WS984	T0 T255	C0 C255	R0 R3839	R3840 R3903	R3904 R3967	R3968 R4167	R5000 R8071	D0 D4095	16 或 32 位 元正、負數
Sa	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Sb	○	○	○	○	○	○	○	○	○	○	○	○	○	○
D		○	○	○	○	○	○		○	○*	○*	○		○

功能敘述

- 當除算控制“EN”=1 或由 0→1 (**P** 指令) 而 “U/S” =0 時，將 Sa 與 Sb 以正負數 (Sign) 運算法則作除法運算並將結果寫入 D 去。同時若商為 0，則 FO0 (商=0) 設為 1，若除數 Sb=0 則錯誤旗號 FO1 (除數=0) 設為 1 且本指令不執行。
- 當除算控制“EN”=1 或由 0→1 (**P** 指令) 而 “U/S” =1 時，將 Sa 與 Sb 以正整數 (Unsign) 運算法則作除法運算並將結果寫入 D 去。同時若商為 0，則 FO0 (商=0) 設為 1，若除數 Sb=0 則錯誤旗號 FO1 (除數=0) 設為 1 且本指令不執行。

程式範例 1 16 位元除法例

階梯圖	按鍵操作	簡碼指令
		ORG X 0 FUN 14 Sa : R 0 Sb : R 1 D : R 2

被除數 Sa	R0
	256
÷	
除數 Sb	R1
	12
D	
	R3
	4
	R2
	21
餘數	商

FUN14 D P (/)	除法運算 (DIVISION) (將 Sa 除以 Sb 所得之商和餘數存到 D 去)	FUN14 D P (/)
---	---	---

程式範例 2 32 位元除法例

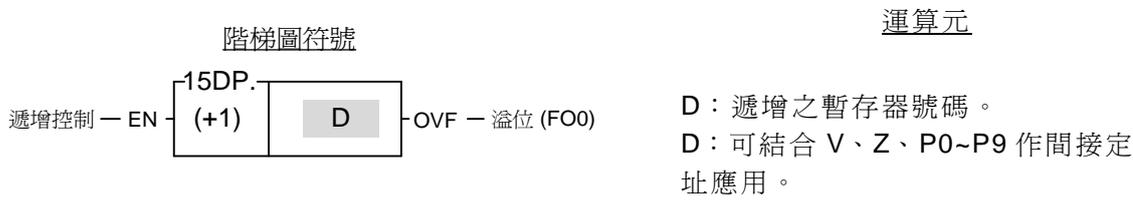
階梯圖	按鍵操作	簡碼指令																																																
	<table border="1" style="margin: auto;"> <tr> <td>ORG</td> <td>X^U</td> <td>0^O</td> <td>確認</td> </tr> <tr> <td></td> <td></td> <td>OPEN</td> <td>ENT</td> </tr> <tr> <td>FUN</td> <td>1^E</td> <td>4</td> <td>SHIFT</td> </tr> <tr> <td></td> <td>SHORT</td> <td></td> <td>S^D</td> </tr> <tr> <td></td> <td></td> <td></td> <td>確認</td> </tr> <tr> <td></td> <td></td> <td></td> <td>ENT</td> </tr> <tr> <td>R</td> <td>0^O</td> <td>確認</td> <td></td> </tr> <tr> <td></td> <td>OPEN</td> <td>ENT</td> <td></td> </tr> <tr> <td>R</td> <td>2</td> <td>確認</td> <td></td> </tr> <tr> <td></td> <td></td> <td>ENT</td> <td></td> </tr> <tr> <td>R</td> <td>4</td> <td>確認</td> <td></td> </tr> <tr> <td></td> <td></td> <td>ENT</td> <td></td> </tr> </table>	ORG	X ^U	0 ^O	確認			OPEN	ENT	FUN	1 ^E	4	SHIFT		SHORT		S ^D				確認				ENT	R	0 ^O	確認			OPEN	ENT		R	2	確認				ENT		R	4	確認				ENT		<pre> ORG X 0 FUN 14D Sa: R 0 Sb: R 2 D: R 4 </pre>
ORG	X ^U	0 ^O	確認																																															
		OPEN	ENT																																															
FUN	1 ^E	4	SHIFT																																															
	SHORT		S ^D																																															
			確認																																															
			ENT																																															
R	0 ^O	確認																																																
	OPEN	ENT																																																
R	2	確認																																																
		ENT																																																
R	4	確認																																																
		ENT																																																



基本應用指令

FUN15 D P (+1)	遞增 (加 1) (將 D 之內容值加 1)	FUN15 D P (+1)
---------------------------------	---------------------------	---------------------------------

指令說明



運算元	範圍	WY	WM	WS	TMR	CTR	HR	OR	SR	ROR	DR	XR
		WY0	WM0	WS0	T0	C0	R0	R3904	R3968	R5000	D0	V、Z
		WY240	WM1896	WS984	T255	C255	R3839	R3967	R4167	R8071	D4095	P0-P9
D		○	○	○	○	○	○	○	○*	○*	○	○

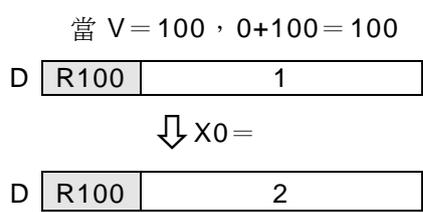
功能敘述

- 當遞增控制“EN”=1 或由 0→1 (P 指令) 時，將 D 之內容值加 1。若 D 之值已在正數之最上限 (32767 或 2147483647)，加 1 之結果會使 D 之值變成負數之最下限 (-32768 或 -2147483648)，同時 FO0 溢位旗號 (OVF) 設為 1。
- 溢位之詳細說明請參考 5.4 節之敘述。

程式範例

16 位元暫存器遞增例 (間接定址)

階段圖	按鍵操作	簡碼指令
		<pre>ORG TU X 0 FUN 15 D: R 0V</pre>



FUN16 D P (-1)	遞減 (減 1) (將 D 之內容值減 1)	FUN16 D P (-1)
--------------------------	---------------------------	--------------------------

指令說明



運算元

D：遞減之暫存器號碼。
 D：可結合 V、Z、P0~P9 作間接定址應用。

運算元 範圍	WY	WM	WS	TMR	CTR	HR	OR	SR	ROR	DR	XR
	WY0	WM0	WS0	T0	C0	R0	R3904	R3968	R5000	D0	V、Z
	WY240	WM1896	WS984	T255	C255	R3839	R3967	R4167	R8071	D4095	P0-P9
D	○	○	○	○	○	○	○	○*	○*	○	○

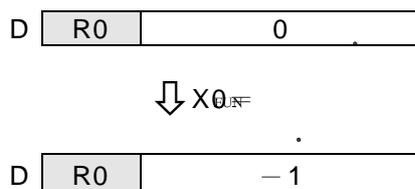
功能敘述

- 當遞減控制“EN”=1 或由 0→1 (P 指令) 時，將 D 之內容值減 1。若 D 之值已在負數之最下限 (-32768 或 -2147483648)，減 1 之結果會使 D 之值變成正數之最上限 (32767 或 2147483647)，同時欠位 (UDF) 旗號 FO0 變成 1。
- 欠位之詳細說明請參考 5.4 節之敘述。

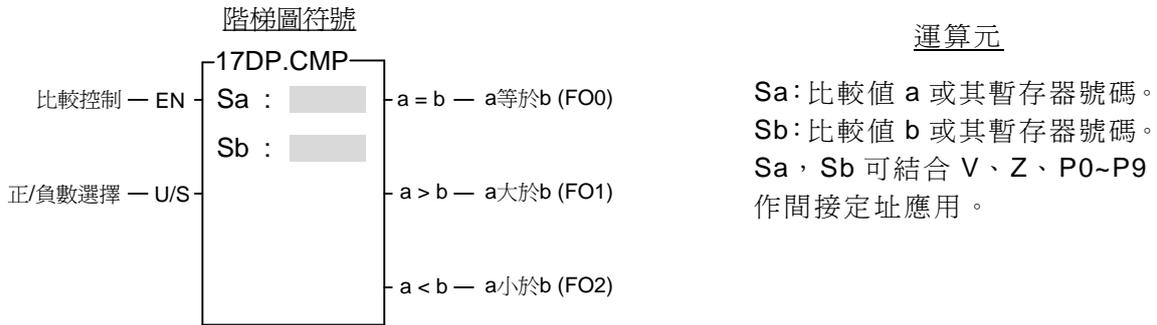
程式範例

16 位元暫存器遞減例

階梯圖	按鍵操作	簡碼指令
		ORG X 0 FUN 16 P [D:] R 0



FUN17 D P CMP	數值比較 (COMPARE) (比較 Sa 與 Sb 數值之大小並產出結果)	FUN17 D P CMP
--------------------------------	---	--------------------------------



運算元	範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR
		WX0	WY0	WM0	WS0	T0	C0	R0	R3840	R3904	R3968	R5000	D0	16 或 32 位元正、負數	V、Z
Sa		WX240	WY240	WM1896	WS984	T255	C255	R3839	R3903	R3967	R4167	R8071	D4095		P0~P9
Sb		○	○	○	○	○	○	○	○	○	○	○	○	○	○

功能敘述

- 當比較控制 “EN” =1 或由 0→1 (P 指令) 而 “U/S” =0 時，本指令以正負數 (Sign) 運算法則執行 Sa 和 Sb 之數值大小比較，若 Sa=Sb 則 FO0 設為 1，若 Sa>Sb 則 FO1 設為 1，若 Sa<Sb 則 FO2 設為 1。
- 當比較控制 “EN” =1 或由 0→1 (P 指令) 而 “U/S” =1 時，本指令以正整數 (Unsign) 運算法則執行 Sa 和 Sb 之數值大小比較，若 Sa=Sb 則 FO0 設為 1，若 Sa>Sb 則 FO1 設為 1，若 Sa<Sb 則 FO2 設為 1。

程式範例

16 位元暫存器數值比較例

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 FUN 17P [Sa]: R 0 [Sb]: R 1 FO 2 OUT Y 0 </pre>

- 上例假若 R0 之值為 1，R1 之值為 2，則當 X0=1 時，CMP 指令執行比較工作，並得出 a < b 之結果，故會將 FO0 及 FO1 設為 0，FO2 (a < b) 設為 1。
- 若您需要複合結果，如 ≥、≤、<> 等，請先將 =、>、< 等結果送到繼電器再由繼電器取出 OR 起來即可。
- 當 M1919=0，本指令不執行時，FO0、FO1、FO2 會保持在上次執行時之狀態。
- 當 M1919=1，本指令不執行時，FO0、FO1、FO2 皆被清除為 0。
- 適當控制 M1919 可得到功能指令輸出有無記憶保持功能。

FUN19 D P OR	邏輯或 (OR) 運算	FUN19 D P OR
-------------------------------	-------------	-------------------------------

階梯圖符號

運算元

Sa : OR 資料 a 或其暫存器號碼。
Sb : OR 資料 b 或其暫存器號碼。
D : 存放 OR 結果之暫存器號碼。
Sa, Sb, D 可結合 V、Z、P0~P9 作間接定址應用。

運算元 範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR
		WX0 WX240	WY0 WY240	WM0 WM1896	WS0 WS984	T0 T255	C0 C255	R0 R3839	R3840 R3903	R3904 R3967	R3968 R4167	R5000 R8071	D0 D4095	16 或 32 位元 正、負數
Sa	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Sb	○	○	○	○	○	○	○	○	○	○	○	○	○	○
D		○	○	○	○	○	○		○	○*	○*	○		○

功能敘述

● 當運算控制“EN”=1 或由 0→1 (P 指令) 時，將 Sa 和 Sb 之資料作邏輯 OR 運算，亦即將 Sa 和 Sb 之各同階位元 (B0~B15 或 B0~B31) 作比較，任兩同階位元中有任一位置元為 1，則 D 之該同階位元設為 1，若兩位元均為 0，才設為 0。

程式範例 16 位元邏輯OR運算例

階梯圖	按鍵操作	簡碼指令																								
		<pre>ORG X 0 FUN 19 [Sa] R 0 [Sb] R 1 [D] R 2</pre>																								
<table style="margin: auto;"> <tr> <td></td> <td style="text-align: center;">B15</td> <td></td> <td style="text-align: center;">B0</td> </tr> <tr> <td style="text-align: right;">Sa</td> <td style="text-align: center;">↓</td> <td style="border: 1px solid black; padding: 2px;">R0</td> <td style="border: 1px solid black; padding: 2px;">1 0 1 1 1 0 1 1 0 1 1 0 1 1 0 1</td> </tr> <tr> <td style="text-align: right;">Sb</td> <td></td> <td style="border: 1px solid black; padding: 2px;">R1</td> <td style="border: 1px solid black; padding: 2px;">1 1 1 0 1 1 1 0 1 0 1 0 0 1 1 0</td> </tr> <tr> <td colspan="4" style="text-align: center; padding: 10px;"> ↓ X0 = 1 FUN </td> </tr> <tr> <td style="text-align: right;">D</td> <td style="text-align: center;">B15</td> <td></td> <td style="text-align: center;">B0</td> </tr> <tr> <td></td> <td style="text-align: center;">↓</td> <td style="border: 1px solid black; padding: 2px;">R2</td> <td style="border: 1px solid black; padding: 2px;">1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1</td> </tr> </table>				B15		B0	Sa	↓	R0	1 0 1 1 1 0 1 1 0 1 1 0 1 1 0 1	Sb		R1	1 1 1 0 1 1 1 0 1 0 1 0 0 1 1 0	↓ X0 = 1 FUN				D	B15		B0		↓	R2	1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1
	B15		B0																							
Sa	↓	R0	1 0 1 1 1 0 1 1 0 1 1 0 1 1 0 1																							
Sb		R1	1 1 1 0 1 1 1 0 1 0 1 0 0 1 1 0																							
↓ X0 = 1 FUN																										
D	B15		B0																							
	↓	R2	1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1																							

FUN20 D P →BCD	BIN→BCD 變換 (將 S 之二進碼轉成 BCD 碼後存入 D)	FUN20 D P →BCD
--	--	--

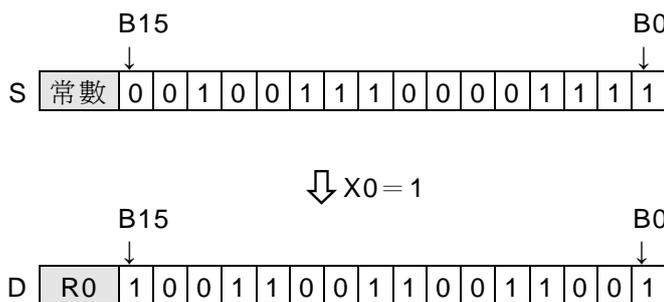
指令說明	運算元																																																																									
<p style="text-align: center;">階梯圖符號</p>	<p>S : 被轉換之資料或其暫存器號碼。 D : 存放轉換結果(BCD 碼)之暫存器號碼。 S, D 可結合 V、Z、P0~P9 作間接定址應用。</p>																																																																									
<table border="1" style="border-collapse: collapse; font-size: small;"> <tr> <td rowspan="3" style="writing-mode: vertical-rl; text-orientation: upright;">運算元</td> <td>WX</td><td>WY</td><td>WM</td><td>WS</td><td>TMR</td><td>CTR</td><td>HR</td><td>IR</td><td>OR</td><td>SR</td><td>ROR</td><td>DR</td><td>K</td><td>XR</td> </tr> <tr> <td>WX0</td><td>WY0</td><td>WM0</td><td>WS0</td><td>T0</td><td>C0</td><td>R0</td><td>R3840</td><td>R3940</td><td>R3968</td><td>R5000</td><td>D0</td><td>16 或 32 位元 正、負數</td><td>V、Z</td> </tr> <tr> <td>WX240</td><td>WY240</td><td>WM1896</td><td>WS984</td><td>T255</td><td>C255</td><td>R3839</td><td>R3903</td><td>R3967</td><td>R4167</td><td>R8071</td><td>D4095</td><td></td><td>P0-P9</td> </tr> <tr> <td>S</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td> </tr> <tr> <td>D</td><td></td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td>○</td><td></td><td>○</td><td>○*</td><td>○*</td><td>○</td><td></td><td>○</td> </tr> </table>		運算元	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR	WX0	WY0	WM0	WS0	T0	C0	R0	R3840	R3940	R3968	R5000	D0	16 或 32 位元 正、負數	V、Z	WX240	WY240	WM1896	WS984	T255	C255	R3839	R3903	R3967	R4167	R8071	D4095		P0-P9	S	○	○	○	○	○	○	○	○	○	○	○	○	○	○	D		○	○	○	○	○	○		○	○*	○*	○		○
運算元	WX		WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	K	XR																																																											
	WX0		WY0	WM0	WS0	T0	C0	R0	R3840	R3940	R3968	R5000	D0	16 或 32 位元 正、負數	V、Z																																																											
	WX240	WY240	WM1896	WS984	T255	C255	R3839	R3903	R3967	R4167	R8071	D4095		P0-P9																																																												
S	○	○	○	○	○	○	○	○	○	○	○	○	○	○																																																												
D		○	○	○	○	○	○		○	○*	○*	○		○																																																												

功能敘述

- 因 PLC 內部之數值均以二進碼 (Binary) 來儲存和演算，因此若欲將 PLC 內部資料送到外界之顯示器去 (如 7 段顯示器)，需利用此指令將之變換成合乎人們習慣之 BCD 碼後再送到顯示器去，以方便閱讀 (例如二進碼 "1100"，若將之轉換為 BCD 碼 "12" 就容易閱讀多了)。
- 當變換控制 "EN" =1 或由 0→1 (P 指令) 時，將 S 之二進碼轉成等值之 BCD 碼後寫入 D 去。若 S 之數值非 BCD 之 0~9999 或 99999999，則 FO0 錯誤旗號設為 1，D 維持舊值。

程式範例 16 位元 BIN→BCD 轉換例

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 FUN 20 [S]: 9999 [D]: R 0 </pre>



基本應用指令

FUN21 D P →BIN	BCD→BIN 變換 (將 S 之 BCD 碼轉成二進碼後存入 D)	FUN21 D P →BIN
--	--	--

指令說明	<div style="text-align: right; margin-bottom: 10px;"><u>運算元</u></div> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p style="text-align: center;">階梯圖符號</p> </div> <div style="width: 50%;"> <p>S : 被轉換之資料或其暫存器號碼。 D : 存放轉換結果(BIN 碼)之暫存器號碼。 S, D 可結合 V、Z、P0~P9 作間接定址應用。</p> </div> </div> <table border="1" style="width:100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 5%;">範圍</th> <th style="width: 5%;">WX</th> <th style="width: 5%;">WY</th> <th style="width: 5%;">WM</th> <th style="width: 5%;">WS</th> <th style="width: 5%;">TMR</th> <th style="width: 5%;">CTR</th> <th style="width: 5%;">HR</th> <th style="width: 5%;">IR</th> <th style="width: 5%;">OR</th> <th style="width: 5%;">SR</th> <th style="width: 5%;">ROR</th> <th style="width: 5%;">DR</th> <th style="width: 5%;">XR</th> </tr> </thead> <tbody> <tr> <td>運算元</td> <td>WX0 WX240</td> <td>WY0 WY240</td> <td>WM0 WM1896</td> <td>WS0 WS984</td> <td>T0 T255</td> <td>C0 C255</td> <td>R0 R3839</td> <td>R3840 R3903</td> <td>R3904 R3967</td> <td>R3968 R4167</td> <td>R5000 R8071</td> <td>D0 D4095</td> <td>V、Z P0-P9</td> </tr> <tr> <td>S</td> <td>○</td> </tr> <tr> <td>D</td> <td></td> <td>○</td> <td>○</td> <td>○</td> <td>○</td> <td>○</td> <td>○</td> <td></td> <td>○</td> <td>○*</td> <td>○*</td> <td>○</td> <td>○</td> </tr> </tbody> </table>	範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	XR	運算元	WX0 WX240	WY0 WY240	WM0 WM1896	WS0 WS984	T0 T255	C0 C255	R0 R3839	R3840 R3903	R3904 R3967	R3968 R4167	R5000 R8071	D0 D4095	V、Z P0-P9	S	○	○	○	○	○	○	○	○	○	○	○	○	○	D		○	○	○	○	○	○		○	○*	○*	○	○
範圍	WX	WY	WM	WS	TMR	CTR	HR	IR	OR	SR	ROR	DR	XR																																												
運算元	WX0 WX240	WY0 WY240	WM0 WM1896	WS0 WS984	T0 T255	C0 C255	R0 R3839	R3840 R3903	R3904 R3967	R3968 R4167	R5000 R8071	D0 D4095	V、Z P0-P9																																												
S	○	○	○	○	○	○	○	○	○	○	○	○	○																																												
D		○	○	○	○	○	○		○	○*	○*	○	○																																												

- 功能敘述**
- 欲由外界以人們習用之 10 進制 (BCD 碼) 輸入裝置 (如指撥開關) 來輸入數值到 PLC 內部, 需利用此指令將 BCD 數字轉成 BIN 碼, PLC 才能處理。
 - 當變換控制 "EN" =1 或由 0→1 (P 指令) 時, 將 S 之 BCD 碼轉換成等值之 BIN 碼後寫入 D 去。若 S 值非 BCD 碼, 則 FO0 錯誤旗號變為 1, D 則維持原值。
 - 常數數值在輸入時就已轉換成二進碼儲存起來, 故不能再以常數當作 S 轉成二進碼。

程式範例 16 位元 BCD→BIN 轉換例

階梯圖	按鍵操作	簡碼指令
		<pre> ORG X 0 FUN 21P S : WX 0 D : R 1 </pre>

X15	1	2	3	4	X0
↓	↓	↓	↓	↓	↓
S	WX0	0	0	0	1
	0	0	1	0	0
	0	0	1	1	0
	1	0	1	0	0
	0	1	0	0	0
	0	0	1	0	0
	0	0	0	1	0
	0	0	0	0	0

↓ X0 =

B15	B0
↓	↓
D	R1
0	0
0	0
0	0
0	0
0	1
0	0
0	0
1	0
0	1
0	0
1	0
0	0
0	1
0	0
0	0