

可編程式控制器

SYSMAC CP系列

CP1E CPU組件

CP1E-E D -

CP1E-N D -

CP1E-NA D -

指令參考手冊

禁止事項

- (1) 本使用手冊禁止部分或全部內容之轉載、複寫及複製。
- (2) 本使用手冊中所記載之相關內容可能未事先告知而逕行更改，敬請見諒。
- (3) 本使用手冊的內容力求盡善盡美，倘若您發現仍有未盡之處或錯誤時，請您通知封底處所記載的OMRON營業處。同時一併將封底處所記載的手冊編號告知我們。

前言

非常感謝您購買 SYSMAC CP 系列可編程式控制器 CP1E，在此謹致上十二萬分的謝意。
本手冊中記載有使用 CP1E 時之必要資訊。
請仔細閱讀本手冊並確實瞭解內容後再行使用。

閱讀對象

本手冊以下列人員為閱讀對象。
具備電器知識 (電氣工程師或相同等級的知識) 的人員

- 負責導入 FA 裝置的人員
- 設計 FA 系統的人員
- 管理 FA 現場的人員

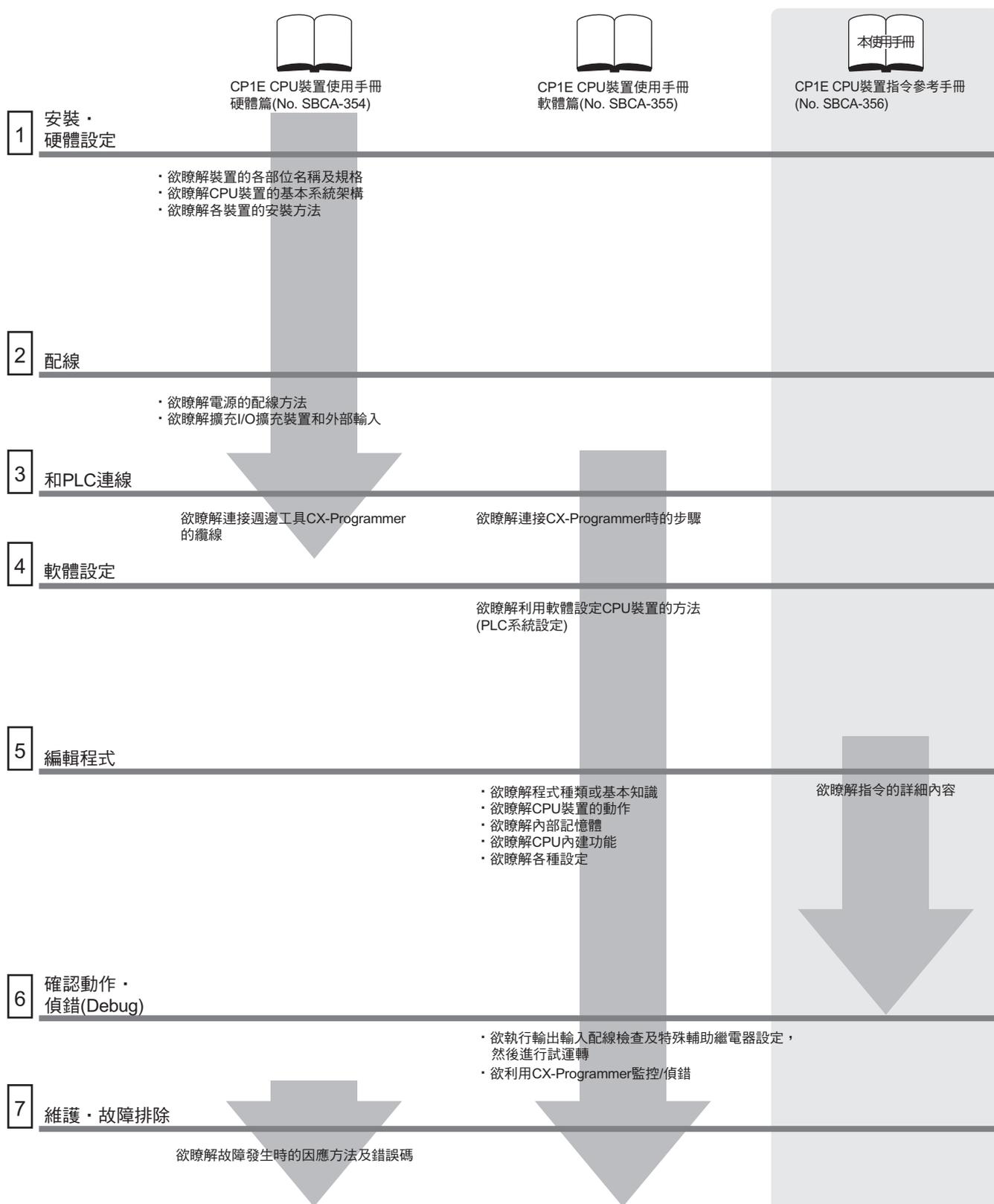
適用的產品

- **CP 系列 CP1E CPU 裝置**
 - 基本型 CP1E-E □□ D □ - □
基本 / 傳送 / 四則運轉 / 比較指令等基本控制專用之基本型產品。
 - 應用型 CP1E-N/NA □□ D □ - □
可支援顯示器、變頻器、伺服驅動器之應用型產品。

出現在本手冊中的「CP 系列」代表 CP1H CPU 裝置、CP1L CPU 裝置及 CP1E CP 裝置的總稱。基本上來說，CP 系列和 CS/CJ 系列採取相同的設計架構。
新增 I/O 時，可使用 CP 系列擴充 I/O 裝置。又，本產品和 CPM1A/CPM2A 系列相同，可利用輸出和輸入，將 I/OCH 配置至固定區。

分冊架構

本產品的使用手冊係由下列部分所組成。
請依使用使用詳細閱讀，以期達到充分運用產品之目的。



章節架構

CP1E CPU 裝置的手冊係由下列章節所組成。請依使用場合讀取所需要的章節。

指令參考手冊 (使用手冊 No. SBCA-356) (本使用手冊)

章	內容
第 1 章 指令一覽表	說明 CP1E CPU 裝置適用的指令一覽表。
第 2 章 各命令明	說明 CP1E CPU 裝置所適用的指令功能、運算元種類的内容及樣本例等。
第 3 章 指令處理時間 / STEP 數	說明 CP1E CPU 裝置各種指令的處理時間。
第 4 章 監控及計算週期時間的方法	說明程式所提供的監控及計算 CP1E CPU 裝置週期時間的方法。
附錄	依指令記號的順序列出 CP1E CPU 裝置的指令一覽表，並說明 ASCII 碼一覽表。

使用手冊 軟體篇 (手冊編號：SBCA-355)

章	內容
第 1 章 概要	說明 CP1E PLC 的概要及使用步驟
第 2 章 CPU 裝置的內部記憶體	說明 CP1E CPU 裝置的內部記憶體種類及可儲存的資料等。
第 3 章 CPU 裝置的動作	說明 CP1E CPU 裝置的動作原理
第 4 章 認識程式	說明 CP1E CPU 裝置的階梯程式在設計方面的基礎知識。
第 5 章 I/O 記憶體	說明 CP1E CPU 裝置的 I/O 記憶體區種類及詳細內容
第 6 章 I/O 配置	說明 CP1E CPU 裝置和裝置之間用來交換資料的 I/O 配置功能。
第 7 章 PLC 系統設定	說明如何利用 PLC 系統設定，以便為 CP1E CPU 裝置進行基本設定。
第 8 章 內建功能概要及配置	說明內建功能一覽表、使用流程及功能配置等。
第 9 章 脈衝擷取功能	說明如何利用脈衝擷取輸入來擷取小於週期時間的訊號。
第 10 章 中斷功能	說明中斷功能概要、輸入中斷、定時中斷等。
第 11 章 高速計數功能	說明高速計數輸入、高速計數中斷及頻率測量功能。
第 12 章 脈衝輸出功能	說明梯形控制、JOG 運轉、原點搜尋等之定位功能。
第 13 章 PWM 輸出功能	說明 PWM (脈衝寬度調變) 輸出功能。
第 14 章 序列通訊功能	說明和可程式化終端器 (PT) 之間的無程式 (Programless) 通訊、和共用模組之間的無程序通訊、Modbus-RTU 簡易遮罩、序列 PLC 連接、和上位 PC 連接等。
第 15 章 類比輸出輸入功能	說明 CP1E NA 型 CPU 裝置所內建的類比輸出輸入功能。
第 16 章 各種功能	說明 PID 溫度控制、時鐘功能、DM 備份功能、安全功能。
第 17 章 使用週邊工具時的操作方法	說明如何建立階梯程式，以便使用 CX-Programmer 啟動 CP1E CPU 裝置，以及傳送資料至 CP1E CPU 裝置、偵錯方法及 CX-Programmer 的基本功能。
附錄	說明指令一覽表、特殊輔助繼電器一覽表、週期時間的回應功能、電源中斷時的 PLC 動作等。

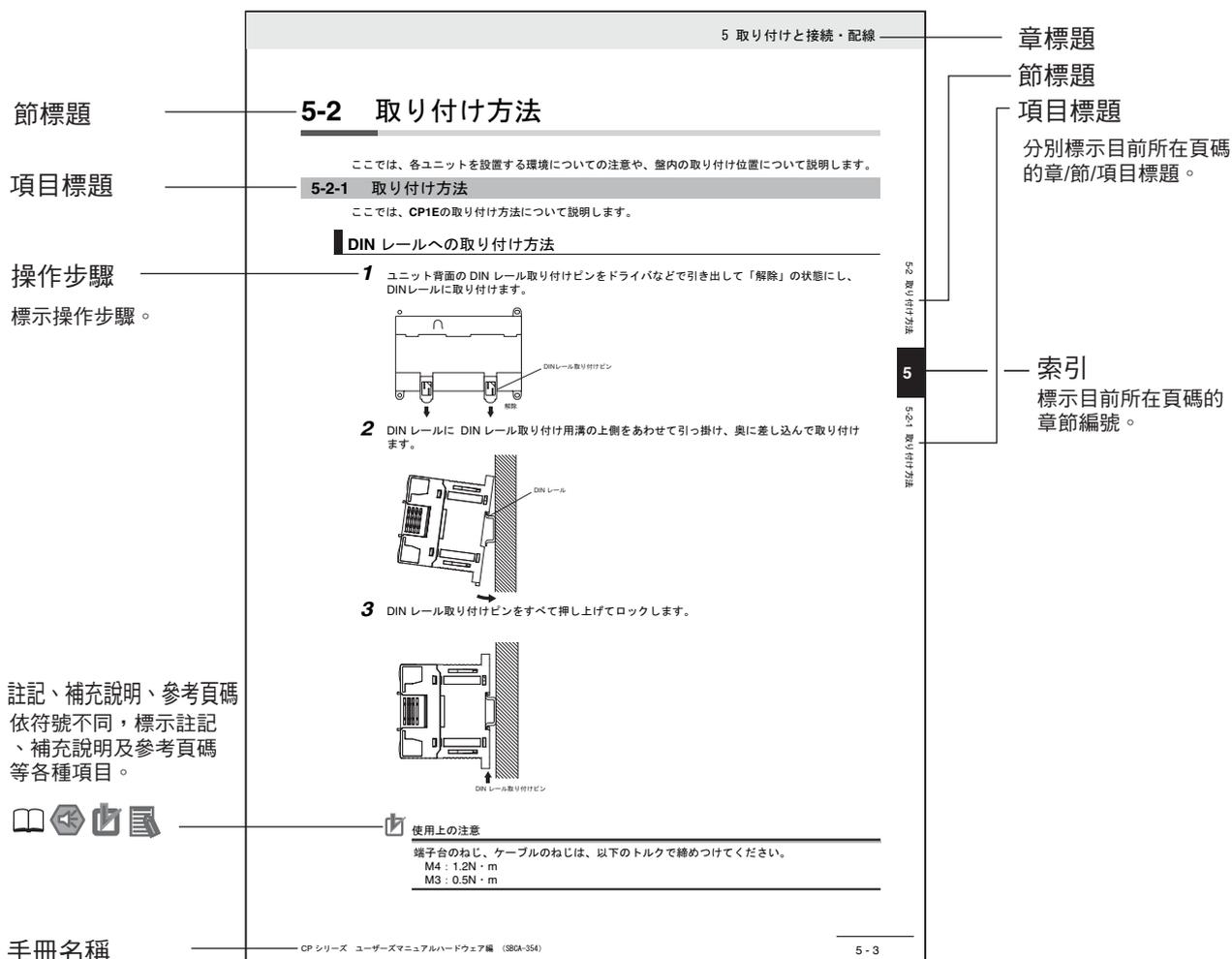
使用手冊 硬體篇 (手冊編號：SBCA-354)

章	內容
第 1 章 概要及規格	說明 CP1E PLC 的概要、特長及規格等。
第 2 章 基本系統及組成裝置	說明 CP1E PLC 的系統架構及裝置一覽表。
第 3 章 各部位名稱及功能	說明 CP1E PLC 的 CPU 裝置、擴充 I/O 裝置的名稱及功能。
第 4 章 週邊工具	說明 PLC 編寫程式、偵錯所使用的週邊工具 CX-Programmer 概要及利用 USB 連接 PLC 和週邊工具的方法。
第 5 章 安裝、連接及配線	說明 CP1E 各裝置的安裝、連接及配線方法。
第 6 章 故障排除	說明故障發生時的確認方法及因應方法，以及各裝置的錯誤顯示等。
第 7 章 維護及檢查	說明定期檢查的內容以及電池使用壽命及更換方法。
第 8 章 擴充裝置的使用方法	說明擴充裝置的使用方法。

如何閱讀本手冊

每頁架構及符號

本手冊每頁的架構及符號如下。



本頁僅為說明範例，與手冊中的實際內容並不相同。

標記

本手冊所使用的標記各具有下列意義。

- 安全上的重點**
記載欲安全使用本產品時，所需進行或避免的事項。
- 使用上的注意事項**
記載為了預防產品發生無法動作、錯誤動作或是在性能與功能上造成不良影響，所應進行或避免的事項。
- 參考**
希望使用者能依實際需要閱讀的項目。
說明必須瞭解的方便資訊、使用時可提供參考的內容。
- 標記**
標記記載詳細資訊或相關資訊的位置。

用語及標記

用語	內容
E 類型	基本 / 傳送 / 四則運轉 / 比較指令等基本控制專用之基本型產品。 本手冊簡稱此裝置為「E 型」。
N 類型	可支援顯示器、變頻器、伺服驅動器之應用型產品。 本手冊簡稱此裝置為「N 型」。
NA 類型	除了 N 型的功能外，還內建類比輸出輸入功能的應用型裝置。 本手冊簡稱此裝置為「NA 型」。
CX-Programmer	可執行 PLC 程式編輯及偵錯的週邊工具。 CX-Programmer 又分為 Micro PLC 限定版 CX-Programmer (CX-One Lite)、CX-Programmer (CX-One) 及 CX-Programmer for CP1E 等不同種類。 本手冊將針對 Micro PLC 限定版 CX-Programmer Ver. 9.03 以上版本 / CX-Programmer for CP1E 本身的使用方法及功能加以說明。所謂「CX-Programmer」指的就是「Micro PLC 限定版 CX-Programmer Ver. 9.03 以上版本 / CX-Programmer for CP1E」。 (註) CP1E E20/30/40 點、N20/30/40 點 CPU 裝置支援 Ver. 8.2 以上的版本。 E10/14 點、N10/14 點、NA20 點 CPU 裝支援 CX-Programmer Ver. 9.03 以上版本。

目錄架構

1	指令一覽表	1
2	各項指令說明	2
3	指令處理時間/STEP數	3
4	監控及計算週期時間的方法	4
附	附錄	附

目錄

前言	1
閱讀對象	1
適用的產品	1
分冊架構	2
章節架構	3
如何閱讀本手冊	5
每頁架構及符號	5
標記	5
用語及標記	6
關於「選購時的注意事項」	12
安全上的注意事項	14
安全使用上的標示與意義	14
圖示記號的說明	14
安全上的重點	17
法規及規則	19
使用於其它國家時	19
關於商標	19
相關使用手冊	20
1 指令一覽表	1-1
1-1 指令一覽表	1-2
2 各項指令說明	2-1
本章的閱讀方法	2-2
順序控制輸入指令	2-5
LD/LD NOT	2-7
AND/AND NOT	2-9
OR/OR NOT	2-11
AND LD/OR LD	2-13
NOT	2-16
UP/DOWN	2-17
順序輸出指令	2-18
OUT/OUT NOT	2-18
TR	2-20
KEEP	2-21
DIFU	2-25
DIFD	2-27
SET/RSET	2-29
SETA/RSTA	2-31
SETB/RSTB	2-33
序列控制指令	2-35
END	2-37
NOP	2-38
IL/ILC	2-39

MILH/MILR/MILC	2-42
JMP/CJP/JME	2-51
FOR/NEXT	2-54
BREAK	2-57
計時器 / 計數器指令	2-58
TIM/TIMX	2-64
TIMH/TIMHX	2-67
TMHH/TMHHX	2-70
TTIM/TTIMX	2-72
TIML/TIMLX	2-75
CNT/CNTX	2-78
CNTR/CNTRX	2-81
CNR/CNRX	2-84
資料比較指令	2-86
=, <>, <, <=, >, >=	2-86
=DT, <> DT, < DT, <= DT, > DT, >= DT	2-89
CMP/CMPL	2-93
CPS/CPSL	2-95
TCMP	2-98
BCMP	2-100
ZCP/ZCPL	2-102
資料傳送指令	2-105
MOV/MOVL/MVN	2-105
MOVB	2-107
MOVD	2-109
XFRB	2-111
XFER	2-113
BSET	2-115
XCHG	2-117
DIST	2-119
COLL	2-121
資料移位指令	2-123
SFT	2-123
SFTR	2-125
WSFT	2-127
ASL	2-129
ASR	2-130
ROL	2-131
ROR	2-133
SLD/SRD	2-135
NASL/NSLL	2-137
NASR/NSRL	2-140
加 / 減指令	2-143
++/++L	2-143
/ L	2-145
++ B/ ++ BL	2-147
-- B/ -- BL	2-149
四則運算指令	2-151
+ / + L	2-151
+ C/ + CL	2-153
+ B/ + BL	2-155
+ BC/ + BCL	2-157
- / - L	2-159
- C/ - CL	2-162
- B/ - BL	2-164
- BC/ - BCL	2-167
* / * L	2-169
* B/ * BL	2-171
/, / L	2-173
/ B, / BL	2-175
資料轉換指令	2-177

BIN/BINL	2-177
BCD/BCDL	2-179
NEG	2-181
MLPX	2-183
DMPX	2-188
ASC	2-193
HEX	2-197
邏輯演算指令	2-201
ANDW/ANDL	2-201
ORW/ORWL	2-203
XORW/XORL	2-205
COM/COML	2-207
特殊演算指令	2-209
APR	2-209
BCNT	2-218
浮動小數點轉換・演算指令	2-220
FIX/FIXL	2-223
FLT/FTL	2-225
+ F, - F, * F, / F	2-227
=F, <>F, <F, <=F, >F, >=F	2-230
FSTR	2-233
FVAL	2-237
資料表執行資料處理時的指令	2-241
SWAP	2-241
FCS	2-243
資料控制指令	2-245
PIDAT	2-245
TPO	2-255
SCL	2-261
SCL2	2-265
SCL3	2-268
AVG	2-271
副程式的指令	2-274
SBS	2-274
SBN/RET	2-278
中斷控制指令	2-281
MSKS	2-282
CLI	2-284
DI	2-287
EI	2-288
高速計數器 / 脈衝輸出指令	2-289
INI	2-289
PRV	2-292
CTBL	2-297
SPED	2-301
PULS	2-304
PLS2	2-306
ACC	2-311
ORG	2-315
PWM	2-318
製程步進控制指令	2-320
SNXT/STEP	2-321
I/O 模組專用指令	2-331
IORF	2-331
SDEC	2-333
DSW	2-335
MTR	2-338
7SEG	2-341
序列通訊指令	2-345

TXD	2-345
RXD	2-350
時鐘功能專用指令	2-356
CADD/CSUB	2-356
DATE	2-360
故障診斷指令	2-362
FAL	2-362
FALS	2-367
特殊指令	2-372
STC/CLC	2-372
WDT	2-373

3 指令處理時間 /STEP 數 3-1

3-1 CP1E 系列指令處理時間 /STEP 數	3-2
---------------------------------	-----

4 監控及計算週期時間的方法 4-1

4-1 監控週期時間	4-2
4-1-1 監控週期時間方法	4-2
4-2 週期時間的計算方法	4-3
4-2-1 PLC 的動作流程	4-3
4-2-2 週期時間計算概要	4-4
4-2-3 各裝置的 I/O 更新時間	4-5
4-2-4 週期時間的計算範例	4-6
4-2-5 執行線上編輯時的週期延遲時間	4-6

附錄 -1

指令英文字母排序表	-2
-----------------	----

關於「選購時的注意事項」

- 保固內容

- (1) 保固期間

- 本公司的產品保固期間為購買產品後，抑或是將產品交貨至指定地點的 1 年以內。

- (2) 保固範圍

- 於上述的保固期間內，若產品因非人為因素而發生故障，本公司將於原購買地點提供免費的替代品更換與維修等服務。

- 但下列故障原因不在保固對象的範圍內：

- (a) 未依本目錄或操作說明書中所規定之條件、環境、操作及使用方法所造成的故障

- (b) 非產品本身原因所造成的故障

- (c) 非透過本公司所進行的改裝或維修所造成的故障

- (d) 未依照原本設計之使用方式所造成的故障

- (e) 出貨時的科技水準所無法預測之原因所造成的故障

- (f) 其它天然災害等不可抗力所造成的故障

- 此外，上述保固僅限於本公司產品本身，因產品故障所導致之相關損失並不包含在本保固範圍內。

- 責任限制

- (1) 關於因本公司產品所引發之一切特別損害、間接損害、消極損害 (喪失應得之利益)，本公司不負任何責任。

- (2) 針對本公司之可程式化產品，如該程式是透過非本公司之技術人員所執行，或因此而造成之所有結果，本公司不負任何責任。

● 符合用途之條件

- (1) 將本公司產品與其他的產品搭配使用時，請確認是否符合顧客所需之規格、法規或限制等。此外，請顧客自行確認目前所使用的系統、機械或是裝置是否適用於本公司商品。如未確認是否符合或適用時，本公司無須對產品的適用性負責。
- (2) 於以下用途使用時，請先洽詢本公司業務人員後根據規格書等進行確認，同時建議您最好做好安全措施，例如採取在額定、性能方面較有餘裕的使用方法；或是採用在發生故障時可將危險程度降至最小的安全回路等。
 - (a) 用於戶外、使用上會遭受潛在化學污染或電力會遭受阻礙、或是在本型錄及使用說明書未記載的條件或環境下使用。
 - (b) 核能控制設備、焚燒設備、鐵路、航空、車輛設備、醫療器材、娛樂用途機械設備、安全裝置以及遵照政府機構或個別業界規定的設備。
 - (c) 危及生命或財產的系統、機械、裝置。
 - (d) 瓦斯、供水 / 電系統，是 24 小時持續運轉的系統等需要高度信賴性的設備。
 - (e) 他、上記 (a) ~ (d) 準、高度安全性必要用途
- (3) 當顧客將本公司產品使用於可能嚴重危害生命、財產等用途時，請務必事先確認系統整體是否有危險告示、並採用備援設計等可確保安全性的設計，以及本公司產品針對整體設備的特定用途上的配電與設置是否適當。
- (4) 由於本型錄所刊載的應用程式範例純屬參考性質，如欲採用時請先確認機械、裝置的功能與安全性後再行使用。
- (5) 懇請顧客務必以正確的方法來使用本公司產品，並了解使用時的禁止事項與注意事項，以免不當的使用而造成他人意外的損失。

● 規格變更

本型錄及使用說明書所記載的產品規格以及附屬品，可能會在必要時進行改良，或因其他事由而有所變更。請洽詢本公司或特約店之業務人員，以確認本公司產品的實際規格。

● 服務對象範圍

本產品的價格不包括維修人員派遣等服務方面的費用。
如顧客有特別需求，請洽本公司的營業負責人。

● 適用範圍

以上的內容僅針對在日本國內的交易及使用行為而編寫。

安全上的注意事項

安全使用上的標示與意義

為了讓使用者能夠安全地使用 CP 系列可編程式控制器，本使用說明書使用下列標示及圖示符號來告知注意事項。

在此所記載的注意事項表示攸關安全性的重大內容，請務必確實遵守。標示與意義如下。

	警告	若未正確使用可能會引發危險，並導致輕傷、中等程度的傷害，甚至是重傷或死亡。另外，同時也可能會導致重大的財物損失。
---	-----------	--

	注意	如未正確操作本機器，有可能會因發生危險而造成人員的輕傷、中度傷害，或是導致重大的財物損失。
---	-----------	---

-  **安全上的重點**
記載欲安全使用本產品時，所需進行或避免的事項。
-  **使用上的注意事項**
記載為了預防產品發生無法動作、錯誤動作或是在性能與功能上造成不良影響，所應進行或避免的事項。

圖示記號的說明



△符號代表需注意 (包含警告) 之意。
具體的內容會記載於△及文章之中。
左圖表示「當心觸電」。



⊘符號代表禁止之意。
具體的內容會記載於⊘及文章之中。
左圖表示「禁止分解」。



●符號代表強制之意。
具體的內容會記載於●及文章之中。
左圖代表「一般強制事項」之意。



△符號代表需注意 (包含警告) 之意。
具體的內容會記載於△及文章之中。
左圖代表「一般注意事項」之意。



△符號代表需注意 (包含警告) 之意。
具體的內容會記載於△及文章之中。
左圖表示「注意高溫」。

警告

傳送程式或 I/O 記憶體、或是執行變更 I/O 記憶體的動作時，請確實確認變更裝置的安全性後再行變更。
無論 CPU 裝置的動作模式為何，皆有可能造成連接至輸出裝置的機械發生誤動作。



注意

使用 E 型或 N/NA 型 (未安裝電池)、或是導入電源時，有可能會造成資料記憶體 (D)(*)、維持繼電器 (H)、計數器現在值 /UP 旗標 (C)、時鐘功能相關的輔助繼電器 (A) 的值出現不穩定的情形。

* 不包含可利用 DM 備份功能，備份至 EEPROM 的區域。
因此，請務必使用下列任一方法來進行初始化。

1. 希望將所有區域歸零時

請利用 PLC 系統設定，進入 [電源 ON 時讀取資料] 並選擇 「維持記憶體 (HR/DM/CNT) 歸零」的核取方塊。

2. 若希望將特定區域歸零或是將某個數值設定為初始值時，請利用階梯程式來設定。
若未進行初始化的話，有可能會因為資料不穩定，導致裝置或機械發生意想不到的動作。



請在確認延長週期時間並不會造成影響後，再進行線上編輯。
否則可能會出現無法讀取輸入訊號的情形。



當 CP1E-N/NA □□ D □ - □ 安裝電池時，即可利用電池來維持資料記憶體 (D)、維持記憶體 (H) 及計數器旗標 / 現在值 (C) 等記憶資料。不過，一旦電池電壓過低時，將造成電池所維持的 I/O 記憶體區 (包含 DM/HR/CNT) 出現不穩定的動作，資料不穩定時，將導致裝置或機械發生意想不到的動作。

欲利用階梯程式來參照 DM 等 I/O 記憶體區的內容，並輸出至外部時，必須採取像是利用 「電池亦常旗標」來停止輸出的措施。



注意

利用階梯程式的區段視窗來監控導通 / 現在值，或是利用監控視窗來監控現在值時，請確實確認其安全性後，再行使用。
倘因快捷鍵操作錯誤，而執行強制設定 / 復歸或設定 / 復歸時，無論 CPU 裝置的動作模式為何，皆有可能導致連接至輸出裝置的機械發生錯誤動作。



以 CH 位址或變數等作為位址的偏移值時，請重新編寫程式，以避免該資料超過啟始位址所屬的記憶體種類。
例如，使用符號比較指令時，請設定為只有間接指定的位置在記憶體區範圍內時才執行處理。
當您所指定的資料超過啟始位址所屬的記憶體區的種類時，系統有可能會讀取其他運算元種類的資料，並發生無法預期的動作。



安全上的重點

使用 CP 系列可編程式控制器時，請注意下列事項。

● 操作時

- 請利用下列方法，將資料記憶體的初始值備份至備份記憶體 (內建的 EEPROM) 中。
 - 請利用 PLC 系統設定，進入「電源 ON 時讀取資料」並設定「需要備份的 DM 的 CH 數」，來設定以 D0 為啟始且需要備份的 DM 大小。
 - 將特殊輔助繼電器 A751.15 (開始儲存 DM 備份旗標) 由 OFF 上升為 ON 後，即可開始編寫階梯程式，以便將指定的資料記憶體備份至內建的 EEPROM 中。
- 請在充分確認所編輯的階梯程式的動作後，再開始進行本裝置的運轉。
- CP1E CPU 裝置會將階梯程式、參數區的資料備份至備份記憶體 (內建的 EEPROM)。備份時，CPU 裝置前方的「BKUP」LED 會亮燈，當 LED 亮燈時，請將 PLC 本體的電 OFF，當電源 OFF 後，系統將不會再備份資料，而且有可能會在下一次導入電源時導致記憶體異常。
- CP1E CPU 裝置可將資料記憶體備份至備份記憶體 (內建的 EEPROM) 中。備份時，CPU 裝置前方的「BKUP」LED 會亮燈，當 LED 亮燈時，請將 PLC 本體的電 OFF，當電源 OFF 後，系統將不會再備份資料，而且不會在下次電源 ON 時將資料傳送至資料記憶體。
- 記憶體的內容有可能因為某些因素而損壞，因此更換電池前必須通電 30 分鐘以上，並且在電源 OFF 後的 5 分鐘之內更換新電池。
- 當您所設定的參數不適當時，有可能會導致無法預期的動作發生。另外，即使已設定適當的參數，亦請先確認不會對設備造成影響後再進行傳送。
- 開始運轉前，請確認資料記憶體的設定是否正確。
- 請將重新運轉時所需要的資料記憶體或保持繼電器之內容、程式、參數及資料傳送至已更換的 CPU 裝置後，再重新執行運轉。
- 請勿拆解、維修或是改造本產品。
- 執行下列操作項目時，請在確認不會影響設備後再進行。
 - 切換 PLC 的動作模式 (包含電源 ON 時的動作模式)
 - 強制設定 / 重設接點
 - 變更現在值或設定值

● 外部回路

- 請架構外部回路，以便在 PLC 本體的電源 ON 後，也能夠將控制區的電源設定為 ON。如果在控制區的電源 ON 後，再將 PLC 本體的電源設定為 ON，一旦 PLC 本體的電源 ON 時，DC 輸出裝置等的輸出就會出現瞬間錯誤動作，並且導致控制區的輸出暫時無法正常動作。
- 當輸出端子的繼電器或電晶體等的內部回路故障時，有可能會造成輸出持續 ON 的狀態。因此，請在 PLC 的外部設置安全回路，以便輸出端子發生錯誤並持續 ON 的狀態時，也能讓系統在安全端執行動作。
- 當 I/O 記憶體維持旗標被設定為 1 (ON) 時，此時如果將系統由「運轉」或「監控」模式變更為「程式」模式，輸出繼電器將不會被清除 (OFF)，並且維持先前的狀態，這時候您必須注意外部負載的狀態。(又，利用停止運轉異常 (包含執行 FALS 指令) 指令來停止裝置運轉時，系統將會維持 CPU 裝置內部 I/O 記憶體的數值，而輸出接點將全部被 OFF。)

法規及規則

使用於其它國家時

若本產品符合外匯及國外貿易管理法所規定的外銷許可、核准產品 (或技術) 之對象，則在外銷產品 (或是供給非國內居住者) 時，必須取得該法則所規定之外銷許可、認證 (或交易許可)。

關於商標

SYSMAC 為 OMRON 製可編程式控制器的登錄商標。

CX-One 為 OMRON Corporation 製可編程式工具的登錄商標。

Windows 為微軟公司的註冊商標。

本手冊中所記載的其它系統名稱及產品名稱分別為各公司的商標或註冊商標。

相關使用手冊

CP1E 的 PLC 本體相關使用手冊是由下表所示的項目所組成的，請同時參考。

手冊名稱	Man. No.	型式	用途	內容
SYSMAC CP 系列 CP1E CPU 裝置 指令參考手冊 (本手冊)	SBCA-356	CP1E-E □□ D □ - □ CP1E-N □□ D □ - □ CP1E-NA □□ D □ - □	可瞭解指令記號的詳細說明	說明各指令記號的詳細內容。 編寫程式時，請同時參考使用手冊 軟體篇 (SBCA-355)。
SYSMAC CP 系列 CP1E CPU 裝置使用手冊 軟體篇	SBCA-355	CP1E-E □□ D □ - □ CP1E-N □□ D □ - □ CP1E-NA □□ D □ - □	可瞭解 CP1E 軟體方面的相關規格	<ul style="list-style-type: none"> • CPU 裝置的動作 • 內部記憶體 • 程式 • 各種設定 • CPU 內建功能 <ul style="list-style-type: none"> • 中斷功能 • 高速計數值輸入功能 • 脈衝輸出功能 • 序列通訊功能 • 類比輸出輸入功能 • 其他功能
			請同時使用 CP1E CPU 裝置使用手冊 硬體篇 (SBCA-354)、指令參考手冊 (SBCA-356)。	
SYSMAC CP 系列 CP1E CPU 裝置使用手冊 硬體篇	SBCA-354	CP1E-E □□ D □ - □ CP1E-N □□ D □ - □ CP1E-NA □□ D □ - □	可瞭解 CP1E 硬體方面的相關規格	說明 CP1E PLC 本體的相關內容如下。 <ul style="list-style-type: none"> • 概要 / 特長 • 基本系統架構 • 各部位名稱及功能 • 安裝及設定方法 • 故障時的因應方法
			請同時使用 CP1E CPU 裝置使用手冊 軟體篇 (SBCA-355)、指令參考手冊 (SBCA-356)。	
CS/CJ/CP/NSJ 系列通訊 指令參考手冊	SBCA-304	CS1G/H-CPU □□ H CS1G/H-CPU □□ -V1 CS1D-CPU □□ H CS1D-CPU □□ S CS1W-SCU □□ -V1 CS1W-SCB □□ -V1 CJ1G/H-CPU □□ H CJ1G-CPU □□ P CJ1M-CPU □□ CJ1G-CPU □□ CJ1W-SCU □□ -V1	可瞭解以 CS/CJ/CP 系列 CPU 裝置、NSJ 系列為對象的通訊指令的詳細內容	1) 說明 C 模式指令及 2) FINS 指令的詳細內容。欲瞭解以 CPU 裝置為對象的通訊指令 (C 模式指令或 FINS 指令) 的詳細內容時，請參閱本手冊。
			(註) 本手冊所刊載的通訊指令係以 CPU 裝置為對象之通訊指令，且不指定通訊路徑 (可經由 CPU 裝置的序列通訊埠、序列通訊埠 / 裝置的通訊埠、通訊裝置等)。	
SYSMAC CP1L/CP1E 導入篇	SBCA-344	CP1L-L10D □ - □ CP1L-L14D □ - □ CP1L-L20D □ - □ CP1L-M30D □ - □ CP1L-M40D □ - □ CP1L-M60D □ - □ CP1E-E □□ D □ - □ CP1E-N □□ D □ - □ CP1E-NA □□ D □ - □	可瞭解 CP1L/CP1E 的基本使用方法	說明 CP 系列 PLC 本體之相關內容如下： <ul style="list-style-type: none"> • 希望瞭解概要 / 特長 • 希望設計系統架構 • 希望進行安裝 / 配線 • 希望編寫程式 • 希望線上調整 / 偵錯

指令一覽表

本章將針對 CP1E CPU 裝置所適用的指令一覽表。

1-1指令一覽表	1-2
----------------	-----

1-1 指令一覽表

以下將介紹 CP1E 所適用的指令記號一覽表。

CP1E 所使用的指令有 200 種。

以下將以一覽表的方式列出指令記號及其處理概要。關於指令記號的詳細內仍請參閱各指令記號的說明頁面。

指令種類	指令名稱	指令記號	FUN No.	功能說明	頁碼
順序控制指令	LOAD	LD	—	表示邏輯啟動。可用來讀取接點的 ON/OFF 內容。	2-7
		@LD	—		
		%LD	—		
		!LD	—		
		!@LD	—		
		!%LD	—		
	LOAD NOT	LD NOT	—	表示邏輯啟動。可將接點的 ON/OFF 內容反相後再讀取。	2-7
		@LD NOT	—		
		%LD NOT	—		
		!LD NOT	—		
		!@LD NOT	—		
		!%LD NOT	—		
	AND	AND	—	擷取接點的 ON/OFF 內容以及與前述輸入條件之間的邏輯積。	2-9
		@AND	—		
		%AND	—		
		!AND	—		
		!@AND	—		
		!%AND	—		
	AND/NOT	AND NOT	—	將接點的 ON/OFF 內容反相，並擷取與前述輸入條件之間的邏輯積。	2-9
		@AND NOT	—		
		%AND NOT	—		
		!AND NOT	—		
		!@AND NOT	—		
		!%AND NOT	—		
OR	OR	—	擷取接點的 ON/OFF 內容以及與前述輸入條件之間的邏輯和。	2-11	
	@OR	—			
	%OR	—			
	!OR	—			
	!@OR	—			
	!%OR	—			
OR/NOT	OR NOT	—	將接點的 ON/OFF 內容反相，並擷取與前述輸入條件之間的邏輯和。	2-11	
	@OR NOT	—			
	%OR NOT	—			
	!OR NOT	—			
	!@OR NOT	—			
	!%OR NOT	—			
AND/LOAD	AND LD	—	擷取回路區塊之間的邏輯積。	2-13	
OR/LOAD	OR LD	—	擷取回路區塊之間的邏輯和。	2-13	
NOT	NOT	520	將輸入條件反相。	2-16	
P.F. 上微分	UP	521	當輸入訊號啟動 (OFF → ON) 時，將執行 ON 動作 1 個週期，並且連接至下一段指令。	2-17	
P.F. 下微分	DOWN	522	當輸入訊號啟動 (ON → OFF) 時，將執行 ON 動作 1 個週期，並且連接至下一段指令。	2-17	
時序輸出指令	輸出	OUT	—	輸出邏輯演算處理的結果。	2-18
		!OUT	—		
	輸出否定	OUT NOT	—	將邏輯演算處理的結果反相後進行輸出。	2-18
		!OUT NOT	—		

指令種類	指令名稱	指令記號	FUN No.	功能說明	頁碼
時序輸出指令	暫時記憶繼電器	TR	—	TR (暫時記憶繼電器) 適用於指令欄程式需要暫時記憶回路執行時的 ON/OFF 狀態。	2-20
	KEEP	KEEP	011	可執行栓鎖繼電器 (自動維持) 的動作。	2-21
		!KEEP			
	上微分	DIFU	013	當輸入訊號上升 (OFF → ON) 時, 您所指定的接點將會持續 ON 動作 1 個週期。	2-25
		!DIFU			
	下微分	DIFD	014	當輸入訊號下降 (ON → OFF) 時, 您所指定的接點將會持續 ON 動作一個週期。	2-27
		!DIFD			
	設定	SET	—	當輸入條件 ON 時, 您所指定的接點就會被設定為 ON, 接下來, 無論輸入條件是否為 ON/OFF, 您所指定的接點將維持 ON 狀態。	2-29
		@SET	—		
		%SET	—		
		!SET	—		
		!@SET	—		
		!%SET	—		
	復歸	RSET	—	當輸入條件 ON 時, 您所指定的接點就會被復歸為 OFF。以降、入力件 ON/OFF、指定接 OFF。	2-29
		@RSET	—		
		%RSET	—		
!RSET		—			
!@RSET		—			
!%RSET		—			
多位元設定	SETA	530	可將相當於所指定的位元數的連續位元設定為 ON。	2-31	
	@SETA				
多位元復歸	RSTA	531	可將相當於所指定的位元數的連續位元設定為 OFF。	2-31	
	@RSTA				
1 位元組	SETB	532	當輸入條件變為 ON 時, 可將您所指定的 CH 的指定接點設定為 ON。	2-33	
	@SETB				
	!SETB				
	!@SETB				
1 位元復歸	RSTB	533	當輸入條件變為 ON 時, 可將您所指定的 CH 的指定接點 OFF 並且進行復歸。	2-33	
	@RSTB				
	!RSTB				
	!@RSTB				
序列控制指令	END	END	001	表示某個程式結束。	2-37
	無功能	NOP	000	無任何功能的指令。	2-38
	互鎖	IL	002	當輸入條件 OFF 時, 可以讓 IL 指令 ~ ILC 指令之間的輸出互鎖。	2-39
	清除互鎖	ILC	003	表示互鎖範圍結束。	2-39
	多重互鎖 (微分旗標維持型)	MILH	517	當輸入條件 OFF 時, 可以讓 MILH 指令 ~ MILC 指令之間的輸出互鎖。	2-42
	多重互鎖 (非微分旗標維持型)	MILR	518	當輸入條件 OFF 時, 可以讓 MILR 指令 ~ MILC 指令之間的輸出互鎖。	2-42
	清除多重互鎖	MILC	519	表示利用互鎖編號相同的 MILH 或 MILR 指令來結束多重互鎖範圍。	2-42
	跳躍	JMP	004	當輸入條件 OFF 時, 就會直接跳躍至 JME 指令。	2-51
	跳躍結束 (Jump End)	JME	005	表示使用 JMP 或 CJP 指令時之跳躍結束位置。	2-51
	條件跳躍 (Jump)	CJP	510	當輸入條件 ON 時, 就會直接跳躍至 JME 指令。	2-51
	開始反覆動作	FOR	512	無條件依指定次數反覆執行 FOR ~ NEX 之間的程式後, 就會開始執行 NEXT 指令以後的指令。	2-54
	結束反覆動作	NEXT	513	表示 FOR-NEXT 迴圈結束。	2-54
	跳離迴圈 (Loop break)	BREAK	514	可中斷 FOR ~ NEXT 迴圈, 並且為 NEXT 前的所有指令執行 NOP 處理。	2-57

指令種類	指令名稱	指令記號	FUN No.	功能說明	頁碼
計時器 / 計數器指令	100 ms 計時器	TIM	—	表示以減法計算高速 ON 延遲的單位為 1 ms 的計時器動作。	2-64
		TIMX	550		2-64
	10 ms 計時器 (高速計時器)	TIMH	015	表示以減法計算高速 ON 延遲的單位為 10 ms (0.01 秒) 的計時器動作。	2-67
		TIMHX	551		2-67
	1 ms 計時器 (超高速計時器)	TMHH	540	表示以減法計算高速 ON 延遲的單位為 1 ms 的計時器動作。	2-70
		TMHHX	552		2-70
	累加計時器	TTIM	087	表示以累加計算高速 ON 延遲的單位為 1 ms 的計時器動作。	2-72
		TTIMX	555		2-72
	長時間計時器	TIML	542	以減法計算 ON 延遲單位為 100 ms 的計時器。	2-75
		TIMLX	553		2-75
	計數器	CNT	—	執行減法計數器的動作。	2-78
		CNTX	546		2-78
可逆計數器	CNTR	012	執行加法計數器的動作。利用加法計數器輸入訊號上升執行加法運算，再利用減法計數器輸入訊號上升，執行減法運算。	2-81	
	CNTRX	548		2-81	
復歸計時器 / 計數器	CNR/ @CNR	545	可復歸指定範圍內的計數器 / 計數器的 UP 旗標。	2-84	
	CNRX/ @CNRX	547			
資料比較指令	符號比較	=, >, <, <=, >, >=	300 ~ 328	比較 CH 資料或常數，當比較結果為 True (真) 時，就會連接至下一段以後的回路。	2-86
	時刻比較	LD, AND, OR+=DT	341	比較 2 種時間資料 (BCD 資料)，當比較結果為 True (真)。	2-89
		LD, AND, OR+>DT	342		
		LD, AND, OR+<DT	343		
		LD, AND, OR+<=DT	344		
		LD, AND, OR+>DT	345		
		LD, AND, OR+>=DT	346		
	不帶符號比較	CMP	020	利用不帶符號 BIN16 位元 (16 進位 4 位數) 來比較 CH 資料或常數，然後再將比較結果反映至條件旗標。	2-93
		!CMP			
	不帶符號倍精確度比較	CMPL	060	利用不帶符號 BIN32 位元 (16 進位 8 位數) 來比較相當於 2CH 的 CH 資料，然後再將比較結果反映至條件旗標。	2-93
	帶符號 BIN 比較	CPS	114	利用不帶符號 BIN16 位元 (以最高位元為符號位元的 16 進位 4 位數) 來比較 CH 資料或常數，然後再將比較結果反映至條件旗標。	2-95
		!CPS			
	帶符號 BIN 倍精確度比較	CPSL	115	利用帶符號倍精確度 BIN32 位元 (以最高位元為符號位元的 16 進位 8 位數) 來比較相當於 2CH 的 CH 資料或常數，然後再將比較結果反映至條件旗標。	2-95
	比較表一致	TCMP	085	針對比較資料 1CH 和相當於比較資料表 16CH 的資料比較其是否一致，接著再將 1 輸出至輸出 CH 所對應的位元。	2-98
@TCMP					
不帶符號資料表比較	BCMP	068	判斷比較資料的內容是否在 16 組比較資料的上下限範圍內，如果在範圍內時，就輸出 1 至輸出 CH 所對應的位元。	2-100	
	@BCMP				
區域比較	ZCP	088	利用不帶符號 BIN16 位元 (16 進位 4 位數) 來比較您所指定的 1 個 CH 資料或常數是否符合您所指定上限值及下限值之間的範圍，接著再將比較結果反映至條件旗標。	2-102	
倍精確度區域比較	ZCPL	116	利用不帶符號 BIN32 位元 (16 進位 8 位數) 來比較您所指定的 1 個倍精確度 CH 資料或常數是否符合您所指定上限值及下限值之間的範圍，接著再將比較結果反映至條件旗標。	2-102	

指令種類	指令名稱	指令記號	FUN No.	功能說明	頁碼
資料傳送指令	傳送	MOV	021	將 CH 資料或常數傳送至指定的 CH。	2-105
		@MOV			
		!MOV			
		!@MOV			
	傳送倍精確度	MOVL/ @MOVL	498	以 32 位元為單位，將相當於 2CH 的 CH 資料或常數輸出至傳送目的端的 CH。	2-105
	否定傳送	MVN/ @MVN	022	將 CH 資料或常數的位元反相資料傳送至您所指定的 CH。	2-105
	位元傳送	MOVB/ @MOVB	082	傳送指定的位元。	2-107
	位數傳送	MOVD/ @MOVD	083	傳送時以位數 (4 位元) 為單位。亦可傳送多位數。	2-109
	多位數傳送	XFRB @XFRB	062	傳送指定 CH 所指定的多位數。	2-111
	區塊傳送	XFER/ @XFER	070	可同時傳送多個連續的 CH 資料。	2-113
	區塊設定	BSET/ @BSET	071	將多個連續的 CH 皆設定為相同的資料	2-115
	資料交換	XCHG/ @XCHG	073	執行 CH 間的資料交換。	2-117
資料配置	DIST/ @DIST	080	以傳送資料作為傳送目的端的基準，然後再將資料傳送到位移 (Offset) 後的 CH。	2-119	
資料擷取	COLL/ @COLL	081	以傳送端為基準，將位移後的 CH 內容傳送到指定的 CH。	2-121	
資料移位指令	移位暫存器	SFT	010	可執行移位暫存器的動作。	2-123
	左右移位暫存器	SFTR/ @SFTR	084	執行可切換移位方向的移位暫存器動作。	2-125
	字元移位	WSFT/ @WSFT	016	表示以 CH 為單位的移位動作。	2-127
	向左移動 1 位元	ASL/ @ASL	025	將 CH 資料向左移動 1 位元。	2-129
	向右移動 1 位元	ASR/ @ASR	026	將 CH 資料向右移動 1 位元。	2-130
	附 CY 的 1 位元向左回轉	ROL/ @ROL	027	將 16 位元 CH 資料及進位 (CY) 旗標等向左回轉 1 位元。	2-131
	附 CY 的 1 位元向右回轉	ROR/ @ROR	028	將 16 位元 CH 資料及進位 (CY) 旗標等向右回轉 1 位元。	2-133
	向左移動 1 位數	SLD/ @SLD	074	以 1 位數 (4 位元) 為單位，將連續的 CH 資料內容向左移位。	2-135
	向右移動 1 位數	SRD/ @SRD	075	以 1 位數 (4 位元) 為單位，將連續的 CH 資料內容向右移位。	2-135
	向左移動 N 位元	NASL/ @NASL	580	依指定的位元數，將 16 位元的 CH 資料向左移動。	2-137
	向左倍精確度移動 N 位元	NSLL/ @NSLL	582	依指定的位元數，將 32 位元的 CH 資料向左移動。	2-137
	向右移動 N 位元	NASR/ @NASR	581	依指定的位元數，將 16 位元的 CH 資料向右移動。	2-140
	向左倍精確度移動 N 位元	NSRL/ @NSRL	583	依指定的位元數，將 32 位元的 CH 資料向右移動。	2-140
	加 / 減指令	BIN 加法運算	++ / @++	590	將 16 進位 4 位數的 1CH 資料加 1。
BIN 倍精確度遞增		++ L/ @++ L	591	將 16 進位 8 位數的 2CH 資料加 1。	2-143
BIN 減法運算		-- / @--	592	將 16 進位 4 位數的 1CH 資料減 1。	2-145
BIN 倍精確度遞減		-- L/ @-- L	593	將 16 進位 8 位數的 2CH 資料減 1。	2-145
BCD 遞增		++ B/ @++ B	594	將 BCD4 位數的 1CH 資料加 1。	2-147
BCD 倍精確度遞增		++ BL/ @++ BL	595	將 BCD8 位數的 2CH 資料加 1。	2-147
BCD 遞減		-- B/ @-- B	596	將 BCD4 位數的 1CH 資料減 1。	2-149
BCD 倍精確度遞減		-- BL/ @-- BL	597	將 BCD8 位數的 2CH 資料減 1。	2-149

指令種類	指令名稱	指令記號	FUN No.	功能說明	頁碼	
四則運算指令	帶符號·無 CY 的 BIN 加法運算	+ / @ +	400	利用帶符號的 16 進位 4 位數，將 CH 資料或常數相加。	2-151	
	帶符號·無 CY 的 BIN 倍精確度加法運算	+ L/ @ + L	401	利用帶符號的 16 進位 8 位數，將 2CH 資料或常數相加。	2-151	
	帶符號·CY 的 BIN 加法運算	+ C/ @ + C	402	利用帶符號 16 進位 4 位數，執行資料、常數及進位 (CY) 旗標等的加法演算。	2-153	
	帶符號·CY 的 BIN 倍精確度加法運算	+ CL/ @ + CL	403	利用帶符號 16 進位 8 位數，執行 2CH 資料、常數及進位 (CY) 旗標等的加法演算。	2-153	
	不含 CY 的 BCD 加法運算	+ B/ @ + B	404	利用 BCD4 位數，將 CH 資料或常數相加。	2-155	
	不含 CY 的 BCD 倍精確度加法運算	+ BL/ @ + BL	405	利用 BCD8 位數，將 2CH 的 CH 資料或常數相加。	2-155	
	含 CY 的 BCD 加法運算	+ BC/ @ + BC	406	利用帶符號的 BCD4 位數，將 CH 資料、常數以及進位 (CY) 旗標等相加。	2-157	
	含 CY 的 BCD 倍精確度加法運算	+ BCL/ @ + BCL	407	利用 BCD8 位數，將相當於 2CH 的 CH 資料、常數以及進位 (CY) 旗標等相加。	2-157	
	帶符號·無 CY 的 BIN 減法運算	- / @ -	410	利用帶符號的 16 進位 4 位數，將 CH 資料或常數相減。	2-159	
	帶符號·無 CY 的 BIN 倍精確度減法運算	- L/ @ - L	411	利用帶符號的 16 進位 8 位數，減去相當於 2CH 的 CH 資料或常數。	2-159	
	帶符號·CY 的 BIN 減法運算	- C/ @ - C	412	利用帶符號的 16 進位 4 位數，減去 CH 資料、常數以及進位 (CY) 旗標等。	2-162	
	帶符號·CY 的 BIN 倍精確度減法運算	- CL/ @CL	413	利用帶符號的 16 進位 8 位數，減去 2CH 的 CH 資料、常數以及進位 (CY) 旗標等。	2-162	
	不含 CY 的 BCD 減法運算	- B/ @ - B	414	利用 BCD4 位數，將 CH 資料或常數相減。	2-164	
	不含 CY 的 BCD 倍精確度減法運算	- BL/ @ - BL	415	利用 BCD8 位數，將 2CH 的 CH 資料或常數相減。	2-164	
	含 CY 的 BCD 減法運算	- BC/ @ - BC	416	利用 BCD4 位數減去 CH 資料、常數以及進位 (CY) 旗標等。	2-167	
	含 CY 的 BCD 倍精確度減法運算	- BCL/ @ - BCL	417	利用 BCD8 位數，將相當於 2CH 的 CH 資料、常數以及進位 (CY) 旗標等相減。	2-167	
	帶符號 BIN 乘法運算	* / @ *	420	利用帶符號的 16 進位 4 位數，將 CH 資料或常數相乘。	2-169	
	帶符號 BIN 倍精確度乘法運算	* L/ @ * L	421	利用帶符號的 16 進位 8 位數，乘以相當於 2CH 的 CH 資料或常數。	2-169	
	BCD 乘法運算	* B/ @ * B	424	利用 BCD4 位數，將 CH 資料或常數相乘。	2-171	
	BCD 倍精確度乘法運算	* BL/ @ * BL	425	利用 BCD8 位數，將相當於 2CH 的 CH 資料或常數相乘。	2-171	
	帶符號 BIN 除法運算	/ / @ /	430	利用帶符號的 16 進位 4 位數，將 CH 資料或常數相除。	2-173	
	帶符號 BIN 倍精確度除法運算	/ L / @ / L	431	利用帶符號的 16 進位 8 位數，除以相當於 2CH 的 CH 資料或常數。	2-173	
	BCD 除法運算	/ B / @ / B	434	利用 BCD4 位數，將 CH 資料或常數相除。	2-175	
	BCD 倍精確度除法運算	/ BL / @ / BL	435	利用 BCD8 位數，將 2CH 的 CH 資料或常數相除。	2-175	
	資料轉換指令	BCD → BIN 轉換	BIN/ @BIN	023	將 BCD 資料轉換為 BIN 資料。	2-177
		BCD → BIN 倍精確度轉換	BINL/ @BINL	058	將倍精確度 BCD 資料轉換為 BIN 資料。	2-177
		BIN → BCD 轉換	BCD/ @BCD	024	將 BIN 資料轉換為 BCD 資料。	2-179
		BIN → BCD 倍精確度轉換	BCDL/ @BCDL	059	倍長 BINBCD 換。	2-179
轉換 2 的補數		NEG/ @NEG	160	以 2 的補數表示 BIN16 位元資料。	2-181	
4 → 16/8 → 256 指令碼		MLPX/ @MLPX	076	讀取指定 CH 的指定位數或指定位元 (解碼為 4 → 168 或解碼為 8 → 256)，然後再將 1 輸出至指定 CH 的對應位元，或是將 0 輸出至 0。	2-183	
16 → 4/256 → 8 指令碼		DMPX/ @DMPX	077	讀取指定 CH 的 16 位元或 256 位元中已經 ON 的最上位元或最下位元 (解碼為 16 → 4 或解碼為 256 → 8)，然後再將結果輸出至指定 CH 的指定位數或指定位元。	2-188	
ASCII 碼轉換		ASC/ @ASC	086	將 16 位元資料的指定位數轉換為 ASCII 碼。	2-193	
ASCII → HEX 轉換		HEX/ @HEX	162	CH 資料的指定位數內容會被當作 8 位元的 ASCII 資料處理，並且以指定 CH 的指定位數所對應的字元輸出。	2-197	

指令種類	指令名稱	指令記號	FUN No.	功能說明	頁碼	
邏輯演算指令	字元邏輯積	ANDW/ @ANDW	034	求出以 CH 資料為單位的 CH 資料或是常數的邏輯積。	2-201	
	字元倍精確度邏輯積	ANDL/ @ANDL	610	求出 2CH 的 CH 資料、CH 資料或常數的邏輯積。	2-201	
	字元邏輯和	ORW/ @ORW	035	求出以 CH 資料為單位的 CH 資料或是常數的邏輯和。	2-203	
	字元倍精確度邏輯和	ORWL/ @ORWL	611	求出 2CH 的 CH 資料、CH 資料或常數的邏輯和。	2-203	
	字元互斥或 (Exclusive OR : XOR) 演算	XORW/ @XORW	036	求出以 CH 資料為單位的 CH 資料或是常數的 XOR 演算。	2-205	
	字元倍精確度 XOR	XORL/ @XORL	612	針對 2CH 的 CH 資料，進行 CH 資料或常數的 XOR 演算。	2-205	
	位元反相	COM/ @COM	029	將 CH 資料的位元反相。	2-207	
	位元倍精確度反相	COML/ @COML	614	將相當於 2CH 的 CH 資料中的各位元反相並輸出。	2-207	
特殊演算指令	數值轉換	APR/ @APR	069	執行 SIN、COS 及線性近似演算。	2-209	
	位元計數器	BCNT/ @BCNT	067	計算 CH 資料中數值為「1」的位元總數。	2-218	
浮動小數點轉換・演算指令	浮動小數點→轉換為 16 位元 BIN	FIX/ @FIX	450	將已指定的浮動小數點 32 位元資料轉換為帶符號的 BIN16 位元資料，然後再將結果輸出至指定的 CH。	2-223	
	浮動小數點→轉換為 32 位元 BIN	FIXL/ @FIXL	451	將已指定的浮動小數點 32 位元資料轉換為帶符號的 BIN32 位元資料，然後再將結果輸出至指定的 CH。	2-223	
	16 位元 BIN→轉換為 浮動小數點	FLT/ @FLT	452	將已指定的帶符號的 BIN16 位元資料轉換為浮動小數點 32 位元資料，然後再將結果輸出至指定的 CH。	2-225	
	32 位元 BIN→轉換為 浮動小數點	FLTTL/ @FLTTL	453	將已指定的帶符號的 BIN32 位元資料轉換為浮動小數點 32 位元資料，然後再將結果輸出至指定的 CH。	2-225	
	浮動小數點加法運算	+ F/ @ + F	454	為所指定的浮動小數點資料 (32 位元) 進行加法運算，然後再將結果輸出至指定的 CH。	2-227	
	浮動小數點減法運算	- F/ @ - F	455	為所指定的浮動小數點資料 (32 位元) 進行減法運算，然後再將結果輸出至指定的 CH。	2-227	
	浮動小數點乘法運算	* F/ @ * F	456	為所指定的浮動小數點資料 (32 位元) 進行乘法運算，然後再將結果輸出至指定的 CH。	2-227	
	浮動小數點除法運算	/ F/ @ / F	457	為所指定的浮動小數點資料 (32 位元) 進行除法運算，然後再將結果輸出至指定的 CH。	2-227	
	單精確度的浮動小數 點資料比較	=F		329	比較所指定的單精確度浮動小數點資料 (32 位元) 或常數後，如果比較結果為真，就會連接至下一段以後的回路區。	2-230
		<>F		330		
		<F		331		
		<=F		332		
		>F		333		
	浮動小數點 < 單精確 度 > →轉換為文字列	FSTR/ @FSTR	448	以小數點格式或指數格式來表示所指定的單精確度浮點資料 (32 位元)，然後再將該資料轉換為文字列資料 (ASCII 碼)。	2-233	
文字列→浮動小數點 < 單精確度 > 轉換	FVAL/ @FVAL	449	以指定的小數點格式或指數格式來表示文字列 (ASCII 碼)，然後再將該資料轉換為單精確度浮點資料 (32 位元)，並將結果輸出至指定的 CH。	2-237		
資料表執行資料處理時的指令	位元置換 (Byte swap)	SWAP/ @SWAP	637	置換指定資料表其資料所指定的 CH 數的上位元及下位元。	2-241	
	計算 FCS 值	FCS/ @FCS	180	計算指定資料表中的 FCS 值，接著將該值轉換為 ASCII 碼資料後再輸出。	2-243	
	附自動微調功能的 PID 演算	PIDAT	191	根據您所指定的參數，進行 PID 演算。	2-245	
	時間分割比例輸出	TPO	685	可輸入指定 CH 中的負載 (Duty) 比或操作量，並且根據指定的參數，將負載比轉換為時間分割比例輸出，然後再將脈衝輸出至指定的接點。	2-255	
	轉換 (Scaling)	SCL/ @SCL	194	根據所指定的一次函數，將不帶符號的 BIN 資料轉換 (Scaling) 為帶符號的 BCD 資料。	2-261	
	轉換 (Scaling) 2	SCL2/ @SCL2	486	根據所指定的附偏移量一次函數，將帶符號的 BIN 資料轉換 (Scaling) 為帶符號的 BCD 資料。	2-265	
	轉換 (Scaling) 3	SCL3/ @SCL3	487	根據所指定的附偏移量一次函數，將帶符號的 BCD 資料轉換 (Scaling) 為帶符號的 BIN 資料。	2-268	
	資料平均	AVG	195	以 BIN 格式來計算 CH 資料所指定週期次數的平均值。	2-271	
副程式的指令	呼叫副程式 (Subroutine call)	SBS/ @SBS	091	執行本程式，即可呼叫指定編號的副程式。	2-274	
	副程式進入	SBN	092	表示指定編號的副程式起始位置。	2-278	
	副程式返回	RET	093	表示副程式的結束。	2-278	

指令種類	指令名稱	指令記號	FUN No.	功能說明	頁碼
中斷控制指令	設定中斷遮罩	MSKS/ @MSKS	690	執行 I/O 中斷處理或是定時中斷處理的指定。	2-282
	中斷解除	CLI/ @CLI	691	指定為解除 / 維持 I/O 中斷輸入記憶，或是指定定時中斷的開始第一次中斷時間。	2-284
	禁止執行中斷任務	DI/ @DI	693	禁止執行所有的中斷任務 (斷電中斷任務除外)。	2-287
	禁止執行中斷任務的解除	EI	694	解除所有中斷任務 (斷電中斷任務除外) 的禁止執行。	2-288
高速計數器 / 脈衝輸出指令	動作模式控制	INI/ @INI	880	執行開始 / 停止和比較表進行比較、變更高速計數器的現在值、變更中斷輸入 (計數器模式) 的現在值、變更脈衝輸出的現在值 (以 0 確定原點)、或是停止脈衝輸出等其中的任一項動作。	2-289
	讀取脈衝現在值	PRV/ @PRV	881	讀取高速計數器的現在值，脈衝輸出的現在值、中斷輸入 (計數器模式) 的現在值。	2-292
	登錄比較資料表	CTBL/ @CTBL	882	針對高速計數器的現在值，進行目標值一致性比較或頻域比較。	2-297
	頻率設定	SPED/ @SPED	885	指定頻率並執行無加速度的脈衝輸出。	2-301
	脈衝量設定	PULS/ @PULS	886	設定脈衝輸出量。	2-304
	定位	PLS2/ @PLS2	887	指定脈衝頻率及加減速比率，以執行具有加減速功能的脈衝輸出 (加速比率 ≠ 減速比率)。	2-306
	頻率加減速控制	ACC/ @ACC	888	指定脈衝頻率及加減速比率，以執行具有加減速功能的脈衝輸出 (加速比率 = 減速比率)。	2-311
	原點搜尋	ORG/ @ORG	889	執行原點搜尋或原點復歸。	2-315
	PWM 輸出	PWM/ @PWM	891	輸出可調整負載比的脈衝。	2-318
製程步進控制指令	進入步進階梯區	SNXT	009	依配置位置不同，分為下列 3 種功能。 1) 進入步進階梯圖區 2) 進入下一個製程編號的步進控制 3) 進入結束步進階梯區的步進控制	2-321
	定義步進階梯區	STEP	008	本指令依是否指定欲配置的位置及製程編號，而有下列 2 種不同的功能。 1) 開始所指定的製程 2) 整個階梯式區的結束區指的就是 STEP 指令 (指定製程編號) ~ STEP 指令 (無製程編號) 之間的區域。	2-321
I/O 模組專用指令	I/O 更新狀態	IORF/ @IORF	097	將所指定的更新下位 CH 編號 ~ 所指定的更新上位編號之間的 I/OCH 資料進行更新。	2-331
	7 段解碼器	SDEC/ @SDEC	078	將 CH 資料的指定位數中各個 4 位元的內容 (0 ~ F) 轉換為 8 位元的 7 區段資料，然後再將結果輸出至指定 CH 以後的上位或下位的各 8 位元。	2-333
	數位開關	DSW	210	讀取連接至輸入裝置及輸出裝置的外部數位開關 (或指撥旋鈕開關) 設定值，然後再以 8 位數或 4 位數數值 (BCD 資料) 的格式儲存至指定 CH 以後的位置。	2-335
	矩陣輸入	MTR	213	由連接至輸入模組及輸出模組的外部 8 點 x 8 列接點 (矩陣) 依序讀取 64 點，然後再以 64 點 (4CH) 資料的格式，儲存在指定的 CH 以後的位置。	2-338
	7 段顯示	7SEG	214	將 4 位數或 8 位數的數值 (BCD 資料) 轉換為 7 段顯示器專用的資料，然後再輸出至指定 CH 以後的位置。	2-341
序列通訊指令	CPU 模組內建的 RS232C 通訊埠、序列選擇埠輸出	TXD/ @TXD	236	維持並且不轉換相當於指定位數的資料，接著利用 PLC 系統設定，指定為無程序模式下的啟動碼 / 結束碼，然後再利用 CPU 裝置內建的 RS-232C 通訊埠 (無程序模式) 或序列通訊埠裝置版本為 Ver. 1.2 以後的序列通訊埠 (無程序模式) 來傳送資料。	2-345
	CPU 模組內建的 RS232C 通訊埠、序列選擇埠輸入	RXD/ @RXD	235	利用 PLC 系統設定，指定為無程序模式下的啟動碼 / 結束碼，然後再利用 CPU 裝置內建的 RS-232C 通訊埠 (無程序模式) 或序列通訊埠裝置版本 Ver.1.2 以後的序列通訊埠 (無程序模式)，由指定的啟始 CH 編號開始，輸出相當於所指定的位元數的接收資料。	2-350
時鐘功能專用指令	日曆加法運算	CADD/ @CADD	730	在時刻資料中執行時間資料的加法運算。	2-356
	日曆減法運算	CSUB/ @CSUB	731	由時刻資料開始執行時間資料的減法運算。	2-356
	修正時鐘	DATE/ @DATE	735	根據所指定的時鐘資料來變更內部時鐘的數值。	2-360
故障診斷指令	運轉持續故障診斷	FAL/ @FAL	006	可登錄或解除使用者所定義的運轉持續異常。	2-362
	運轉停止故障診斷	FALS	007	登錄使用者所定義的停止運轉異常。	2-367
特殊指令	設定進位	STC/ @STC	040	將 CY 旗標設定為 ON。	2-372
	清除進位	CLC/ @CLC	041	將 CY 旗標設定為 OFF。	2-372
	設定週期監控時間	WDT/ @WDT	094	僅能針對執行本指令的週期，延長其週期監控時間。	2-373

2

各項指令說明

本章將針對 CP1E CPU 模組所適用的指令功能、運算元種類的内容及程式例等加以說明。

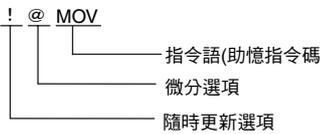
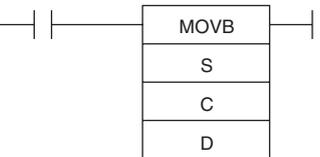
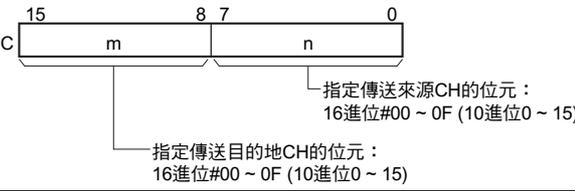
本章的閱讀方法	2-2
順序控制輸入指令	2-5
順序輸出指令	2-18
順序控制指令	2-35
計時器 / 計數器指令	2-58
資料比較指令	2-86
資料傳送指令	2-105
資料移位指令	2-123
加 / 減指令	2-143
四則運算指令	2-151
資料轉換指令	2-177
邏輯演算指令	2-201
特殊演算指令	2-209
浮動小數點轉換・演算指令	2-220
資料表執行資料處理時的指令	2-241
資料控制指令	2-245
副程式的指令	2-274
中斷控制指令	2-281
高速計數器 / 脈衝輸出指令	2-289
製程步進控制指令	2-320
I/O 模組專用指令	2-331
序列通訊指令	2-345
時鐘功能專用指令	2-356
故障診斷指令	2-362
特殊指令	2-372

本章的閱讀方法

指令記號的順序：依功能順序排列。

如欲直接查詢指令記號，請參閱「附錄 指令英文字母排序表」。

各項指令記號之說明項目如下：

項目	內容																																																																																						
指令名稱	表示指令記號的名稱。例：位元傳送																																																																																						
指令記號	指令記號。 例：MOVB																																																																																						
指令的各種組合	各種指令的各種組合的指令記號。 微分選項 @：將輸入上微分型指令作為指令。 %：將輸入下微分型指令作為指令。 立即更新選項 !：執行指令時，會針對已指定運算元種類的輸出輸入區域資料進行 I/O 更新。 																																																																																						
FUN No.	FUN No.。																																																																																						
功能	指令記號的功能。																																																																																						
符號	表示 CX-Programmer 中的階梯圖符號。 例： 																																																																																						
是否適用於特定區域	表示本指令是否適用在下列各種區域中。 工程步進程式 副程式 中斷任務程式 ○：可以使用。X：不可使用。 例： <table border="1" data-bbox="470 1400 1396 1467"> <thead> <tr> <th>區域</th> <th>工程步進程式</th> <th>副程式</th> <th>中斷任務程式</th> </tr> </thead> <tbody> <tr> <td>使用</td> <td>○</td> <td>○</td> <td>○</td> </tr> </tbody> </table>	區域	工程步進程式	副程式	中斷任務程式	使用	○	○	○																																																																														
區域	工程步進程式	副程式	中斷任務程式																																																																																				
使用	○	○	○																																																																																				
運算元種類的說明	表示運算元種類的內容、資料類型及大小。 表示僅針對控制資料等需要說明的運算元種類，說明其 CH/ 位元所代表的意義。 例： 																																																																																						
運算元種類	各運算元種類可指定的位元種類。元件所指為各種元件名稱。 S(來源運算元種類)、C(控制資料)、D(目的地運算元種類) 是否可以使用該項元件。 ○：可以使用。X：無法使用。 例： <table border="1" data-bbox="470 1915 1396 2116"> <thead> <tr> <th rowspan="2">元件</th> <th colspan="10">CH 位址</th> <th rowspan="2">常數</th> <th colspan="3">暫存器</th> <th rowspan="2">TK</th> <th rowspan="2">條件旗標</th> <th rowspan="2">時鐘脈衝</th> <th rowspan="2">TR</th> </tr> <tr> <th>CIO</th> <th>WR</th> <th>HR</th> <th>AR</th> <th>T</th> <th>C</th> <th>DM</th> <th>EM</th> <th>@DM @EM</th> <th>*DM *EM</th> <th>DR</th> <th>IR 直接</th> <th>IR 間接</th> </tr> </thead> <tbody> <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><td>-</td><td>-</td><td>-</td> </tr> <tr> <td>C</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><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><td></td><td></td><td></td> </tr> </tbody> </table>	元件	CH 位址										常數	暫存器			TK	條件旗標	時鐘脈衝	TR	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM	DR	IR 直接	IR 間接	S											○	○	-	○	-	-	-	C	○	○	○	○	○	○	○	○	○	○	-	○	-	○	-	-	-	D											-						
元件	CH 位址										常數	暫存器			TK	條件旗標					時鐘脈衝	TR																																																																	
	CIO	WR	HR	AR	T	C	DM	EM	@DM @EM	*DM *EM		DR	IR 直接	IR 間接																																																																									
S											○	○	-	○	-	-	-																																																																						
C	○	○	○	○	○	○	○	○	○	○	-	○	-	○	-	-	-																																																																						
D											-																																																																												

項目	內容												
相關的條件旗標	<p>本指令被執行時，相關的條件旗標 ON/OFF 變化的說明。 此處只針對有關的條件旗標作說明。 例：</p> <table border="1"> <thead> <tr> <th>名稱</th> <th>標籤</th> <th>內容</th> </tr> </thead> <tbody> <tr> <td>異常旗標</td> <td>P_ER</td> <td> <ul style="list-style-type: none"> • C的資料超出範圍時，ON。 • 除此之外，OFF。 </td> </tr> <tr> <td>= 旗標</td> <td>P_EQ</td> <td>OFF</td> </tr> <tr> <td>負旗標</td> <td>P_N</td> <td>OFF</td> </tr> </tbody> </table>	名稱	標籤	內容	異常旗標	P_ER	<ul style="list-style-type: none"> • C的資料超出範圍時，ON。 • 除此之外，OFF。 	= 旗標	P_EQ	OFF	負旗標	P_N	OFF
名稱	標籤	內容											
異常旗標	P_ER	<ul style="list-style-type: none"> • C的資料超出範圍時，ON。 • 除此之外，OFF。 											
= 旗標	P_EQ	OFF											
負旗標	P_N	OFF											
功能	顯示指令的功能。												
提示	基本功能外的補充說明。												
使用上的注意事項	使用時應注意的事項。												
程式例	程式舉例說明。												

關於常數的表現方式

本章將以下列方式來表現運算元種類常數。

- 在「運算元種類說明」及「資料內容」項目中
 - 當運算元種類以位元列表示時 (一般以 16 進位為輸入單位)

表示僅採用 16 進位表示。例)MOV 指令的運算元種類 S：表示僅以 #0000 ~ FFFF 作為常數範圍。(不過，週邊工具可附加前綴碼 & 並且以 10 進位作為輸入單位)
 - 當運算元種類以數字 (一般以 10 進位為輸入單位) 表示時 (含跳躍 (Jump) 指令)

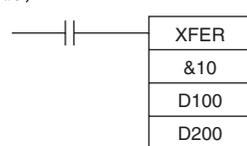
代表 16 進位及 10 進位表示法。例)XFER 指令的運算元種類 W：表示常數範圍為 #0000 ~ FFFF 及 10 進位 & 0 ~ 65535。
 - 當運算元種類以編號表示時 (跳躍 (Jump) 指令除外)

代表 10 進位表示法。例)SBS 指令的運算元種類 N：定範、0 ~ 1023 表示。

- 在「動作說明」項目中

表示在「動作說明」的階梯圖中使用 CX-Programmer 的輸入標記。

(註) 意思就是當運算元種類以數字表示時 (一般是以 10 進位為輸入單位)，可採用附加前綴碼 & 的 10 進位表示方式)



參考：利用週邊工具的常數輸入方法如下：

週邊工具	CX-Programmer
用來表示位元列 (一般以 16 進位為輸入單位) 的運算元種類	附加前綴碼 &，並且以 10 進位為輸入方式。
用來表示數值 (一般以 10 進位方式輸入) 的運算元種類	或是附加前綴碼 #，並以 16 進位方式輸入。(註)
用來表示編號的運算元種類 (除 Jump 指令外)	附加前綴碼 #，並以 10 進位方式輸入。(註)

(註) 表示 CX-Programmer 在輸入運算元種類時可輸入的資料範圍 (含前綴碼)。

關於條件旗標

本章將以 CX-Programmer 的條件標記來做說明。

名稱	CX-Programmer 中的名稱
異常旗標	P.ER
存取異常旗標	P.AER
進位旗標 (Carry flag)	P.CY
> 旗標	P.GT
= 旗標	P.EQ
< 旗標	P.LT
負旗標	P.N
溢位旗標 (Overflow flag)	P.OF
欠位旗標 (Underflow flag)	P.UF
≥ 旗標	P.GE
≠ 旗標	P.NE
≤ 旗標	P.LE
常 ON 旗標	P.On
常 OFF 旗標	P.Off

符號化指令

當 C/CV 系列中的部分指令記號被用於 CP1E CPU 模組時將變為具有相同功能的其他指令。

	C/CV 系列	CP1E 系列
資料比較	EQU	AND=
資料傳送	MOVQ	MOV
加 / 減	INC	++ B
	INCL	++ BL
	INCB	++
	INBL	++ L
	DEC	-- B
	DECL	-- BL
	DECB	--
	DCBL	-- L
四則運算	ADB	+ C
	ADBL	+ CL
	ADD	+ BC
	ADDL	+ BCL
	SBB	- C
	SBBL	- CL
	SUB	- BC
	SUBL	- BCL
	MBS	*
	MBSL	* L
	MUL	* B
	MULL	* BL
	DBS	/
	DBSL	/ L
DIV	/ B	
DIVL	/ BL	
中斷控制	INT	MSKS / CLI DI / EI

順序控制輸入指令

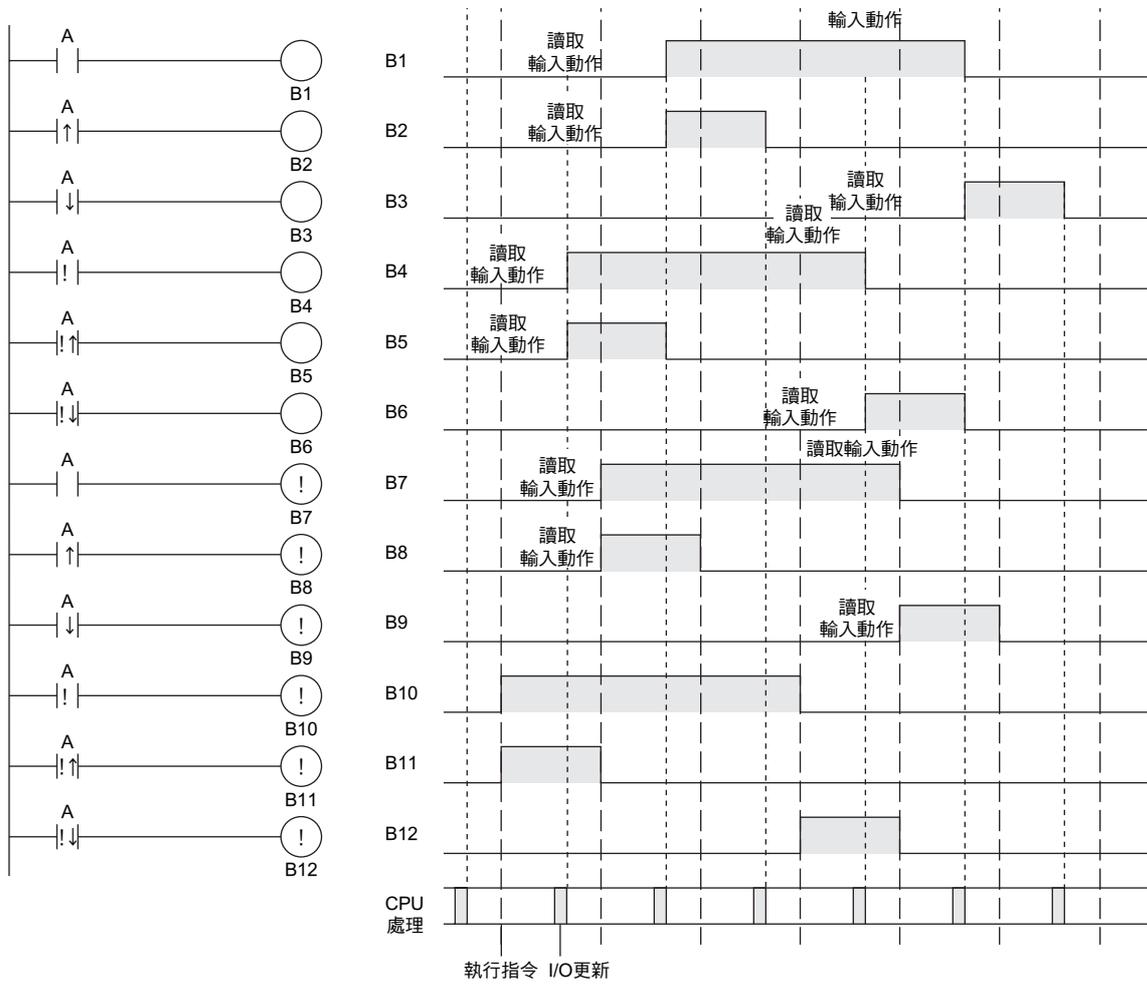
微分型指令及立即更新型指令

- LD/AND/OR 指令除了一般指令外，還包含微分型指令、立即更新型指令以及前述兩種指令之組合型指令。
 - LD NOT/AND NOT/OR NOT/OUT/OUT NOT 指令除了一般指令外，還包含立即更新型指令。
 - 一般指令、微分型指令、立即更新型指令以及微分立即更新型指令利用資料處理輸出輸入的時序各有所不同。
 - 一般指令及微分型指令係利用執行指令前藉由 I/O 更新處理所輸入的資料來執行指令，然後在下一次 I/O 更新處理時輸出執行結果，此種「I/O 更新」意謂著 CPU 內部記憶體與輸出輸入模組之間所進行的資料處理。
 - 立即更新型指令除了會執行上述的 I/O 更新外，輸入指令會在指令執行前，而輸出指令則會在指令執行後，僅針對已存取的 CH 與輸出輸入模組之間進行資料處理。接點的立即更新型指令可針對指定接點等相當於 8 位元的資料 (上層 8 位元或下層 8 位元) 同時進行更新。
- 立即更新指令並不適用於 CP1W 擴充模組。如果要對 CP1W 擴充模組執行指令時，請使用 I/O 更新指令 (IORF)。

指令型式	指令記號	功能	I/O 更新狀態
一般指令	LD/AND/OR LD NOT/AND NOT/ OR NOT	利用週期性更新的方式將指定接點的 ON/OFF 狀態讀入 CPU，並且將結果反映在下一個要執行的指令上。	週期更新
	OUT/OUT NOT	執行指令後，會在下一個週期性更新時輸出指定接點的 ON/OFF 狀態。	
上微分型指令	@LD/@AND/@OR	當指定接點的 ON 啟動 (OFF → ON) 時執行指令，而且持續 ON 動作 1 個週期。	週期更新
下微分型指令	%LD/%AND/%OR	當指定接點的 OFF 關閉 (ON → OFF) 時執行指令，而且持續 ON 動作 1 個週期。	
立即更新型指令	!LD!/AND!/OR/ !LD NOT!/AND NOT/ !OR NOT	將指定接點的輸入資料讀入 CPU 後即執行指令。	執行指令前
	!OUT!/OUT NOT	執行指令後，會將資料即時輸出至指定接點。	指令執行後
上微分・立即更新型指令	!@LD!/!@AND!/!@OR	將指定接點的輸入資料讀入 CPU，只要 ON 啟動 (OFF → ON) 立即執行指令，而且持續 ON 動作 1 個週期。	執行指令前
下微分・立即更新型指令	!%LD!/!%AND!/!%OR	將指定接點的輸入資料讀入 CPU，只要 OFF 關閉 (ON → OFF) 立即執行指令，而且持續 ON 動作 1 個週期。	

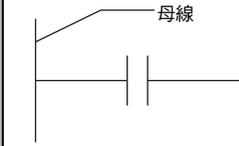
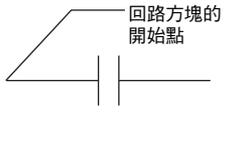
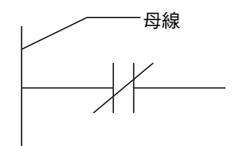
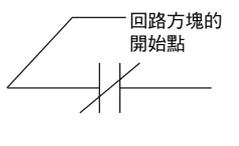
各輸出輸入指令的動作時序圖

LD 指令及 OUT 指令所組成的程式在各種指令動作上的差異，請參閱時序圖所示。



LD/LD NOT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
載入	LD	@LD, %LD, !LD, !@LD, !%LD	—	母線開始的第一個 a 接點。
LOAD NOT	LD NOT	@LD NOT, %LD NOT, !LD NOT, !@LD NOT, !%LD NOT	—	母線開始的第一個 b 接點。

符號	LD		LD NOT	
				

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
—	—	BOOL	—

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
LD	○	○	○	○	○	○	—	—	—	—	○	○	○
LD NOT	○	○	○	○	○	○	—	—	—	—	○	○	—

條件旗標的動作

無

功能

● LD

適用於連接母線的第一個 a 接點或是回路區塊的第一個 a 接點。

未指定立即更新時，本指令將會讀取 I/O 記憶體指定位元中的內容。若已經指定為立即更新時，則會讀取 CPU 內建輸入端子的實際接點狀態。

● LD NOT

適用於連接母線的第一個 b 接點或是回路區塊的第一個 b 接點。

未指定立即更新時，本指令會將 I/O 記憶體指定位元中的內容反相後再行讀取。若已經指定為立即更新時，則會將 CPU 內建輸入端子的實際接點狀態反相後再行讀取。

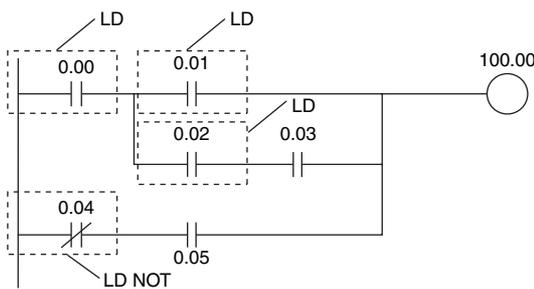
提示

- LD/LD NOT 指令是一種用來表示邏輯啟動的指令，並且適用於下列情況。
 - 直接連接母線時
 - 利用 AND LD 指令或 OR LD 指令來連接回路區塊時 (適用於回路區塊的前端) (AND LD 指令或 OR LD 指令可以讓以 LD/OD NOT 指令作為啟始的回路區塊串聯或是並聯連接。)
- 在輸出類指令中，只要該指令無法直接連接至母線，則輸入條件必須包含 1 個以上的 LD/LD NOT 指令；若未加入 LD/LD NOT 指令的話，週邊工具在執行程式檢查時，就會出現回路錯誤的情形。
- 利用 AND LD 指令或 OR LD 指令來連接回路區塊時，AND LD/OR LD 指令的總數必須等於 LD/LD NOT 指令的總數 -1，否則，有可能會出現回路錯誤的情形 (詳細內容請參閱 AND LD/OR LD 該節的相關說明)。

使用上的注意事項

- LD 指令可指定上微分或下微分 (@ 或 %) 動作，此時，只有在指定接點發生 OFF → ON 或是 ON → OFF 的變化時，才執行 ON 動作 1 個週期，週期結束後即變為 OFF。
- LD/LD NOT 指令可指定為立即更新 (!)，此時，本指令會在執行前，針對 CPU 模組內建輸入執行 IN 更新，並且直接讀取更新後的內容。
- LD 指令可將上微分、下微分及立即更新等動作互相搭配指定 (!@ 或 !%)，此時，本指令會在執行前，藉由 IN 更新方式由基本輸入模組直接讀取資料，只有當讀取到的內容發生 OFF → ON 或是 ON → OFF 的變化時，才會持續 ON 動作 1 個週期，週期結束後即變為 OFF。

程式例



指令	資料
LD	0.00
LD	0.01
LD	0.02
AND	0.03
OR LD	-
AND LD	-
LD NOT	0.04
AND	0.05
OR LD	-
OUT	100.00

AND/AND NOT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
And	AND	@AND, %AND, !AND, !@AND, !%AND	—	串接 a 接點。
AND/NOT	AND NOT	@AND NOT, %AND NOT, !AND NOT, !@AND NOT, !%AND NOT	—	串接 b 接點。

符號	AND	AND NOT
		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
—	—	BOOL	—

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
AND/ AND NOT	○	○	○	○	○	○	—	—	—	—	○	○	—

條件旗標的動作

無

功能

● AND

適用於以串聯連接的 a 接點，無法直接與母線連接。

不適用於回路區塊的前端。未指定立即更新時，本指令將會讀取 I/O 記憶體指定位元中的內容。若已經指定為立即更新時，則會讀取 CPU 內建輸入端子的實際接點狀態。

● AND NOT

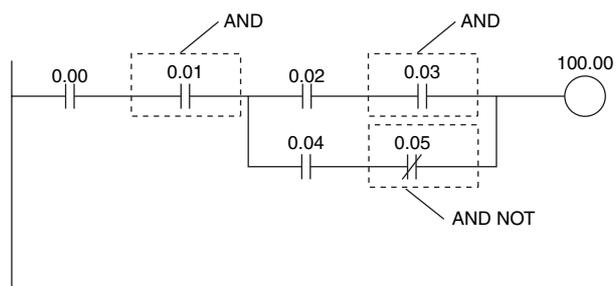
適用於以串聯連接的 b 接點，無法直接與母線連接。

不適用於回路區塊的前端。未指定立即更新時，本指令將會讀取 I/O 記憶體指定位元中的內容。若已經指定為立即更新時，則會將 CPU 內建輸入端子的實際接點狀態反相後再行讀取。

使用上的注意事項

- AND指令可指定上微分或下微分(@或%)動作,此時,只有在指定接點發生OFF → ON或是ON → OFF的變化時,才執行 ON 動作 1 個週期,週期結束後即變為 OFF。
- AND/AND NOT 指令可指定為立即更新(!),此時,本指令會在執行前,針對 CPU 模組內建輸入執行 IN 更新,並且直接讀取更新後的內容。
- AND 指令可將上微分、下微分及立即更新等動作互相搭配指定(!@ 或 !%),此時,本指令會在執行前,藉由 IN 更新方式由基本輸入模組直接讀取資料,只有當讀取到的內容發生 OFF → ON 或是 ON → OFF 的變化時,才會持續 ON 動作 1 個週期,週期結束後即變為 OFF。

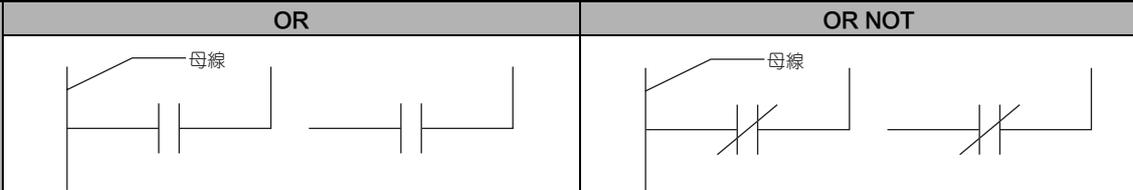
程式例



指令	資料
LD	0.00
AND	0.01
LD	0.02
AND	0.03
LD	0.04
AND NOT	0.05
OR LD	—
AND LD	—
OUT	100.00

OR/OR NOT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
OR	OR	@OR, %OR, !OR, !@OR, !%OR	—	並接 a 接點。
OR/NOT	OR NOT	@OR NOT, %OR NOT, !OR NOT, !@OR NOT, !%OR NOT	—	並接 b 接點。

符號	OR	OR NOT
		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
—	—	BOOL	—

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
OR/ OR NOT	○	○	○	○	○	○	—	—	—	—	○	○	—

條件旗標的動作

無

功能

● OR

適用於以並聯連接的 a 接點。

(可連接至母線或配置在回路區塊的前端) 以 LD/LD NOT 指令作為起始，而且由本指令前的回路之間的 OR (邏輯和) 的 a 接點所組成。

未指定立即更新時，本指令將會讀取 I/O 記憶體指定位元中的內容。若已經指定為立即更新時，則會讀取 CPU 內建輸入端子的實際接點狀態。

● OR NOT

適用於以並聯連接的 b 接點。

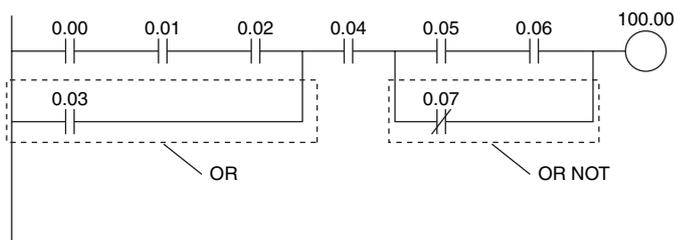
(可連接至母線或配置在回路區塊的前端) 以 LD/LD NOT 指令作為起始，而且由本指令前的回路之間的 OR (邏輯和) 的 b 接點所組成。

未指定立即更新時，本指令將會讀取 I/O 記憶體指定位元中的內容。若已經指定為立即更新時，則會讀取 CPU 內建輸入端子的實際接點狀態。

使用上的注意事項

- OR指令可指定上微分或下微分(@或%)動作,此時,只有在指定接點發生 OFF → ON 或是 ON → OFF 的變化時,才執行 ON 動作 1 個週期,週期結束後即變為 OFF。
- OR/OR NOT 指令可指定為立即更新(!),此時,本指令會在執行前,針對 CPU 模組內建輸入執行 IN 更新,並且直接讀取更新後的內容。
- OR 指令可將上微分、下微分及立即更新等動作互相搭配指定(!@或!%),此時,本指令會在執行前,藉由 IN 更新方式由基本輸入模組直接讀取資料,只有當讀取到的內容發生 OFF → ON 或是 ON → OFF 的變化時,才會持續 ON 動作 1 個週期,週期結束後即變為 OFF。

程式例



指令	資料
LD	0.00
AND	0.01
AND	0.02
OR	0.03
AND	0.04
LD	0.05
AND	0.06
OR NOT	0.07
AND LD	—
OUT	100.00

AND LD/OR LD

指令名稱	指令記號	指令的各種組合	Fun No.	功能
AND LOAD	AND LD	—	—	兩個回路串接。
OR LOAD	OR LD	—	—	兩個回路並接。

符號	AND LD	OR LD
		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

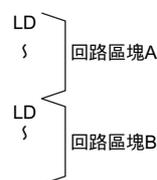
條件旗標的動作

無

功能

● AND LD

可將本指令前的回路區塊彼此以串聯連接。
所謂回路區塊就是以 LD/LD NOT 指令作為起始，直到下一個 LD/LD NOT 指令前的所有回路。



AND LD……將回路區塊A及回路區塊B以串聯連接。

● OR LD

可將本指令前的回路區塊彼此以並聯連接。
所謂回路區塊就是以 LD/LD NOT 指令作為起始，直到下一個 LD/LD NOT 指令前的所有回路。



AND LD……將回路區塊A及回路區塊B以串聯連接。

提示

● AND LD

- 如果要將3個以上的回路區塊以串聯連接時，可以利用本指令將2個回路區塊串聯連接，接著再將下一個回路區塊以串聯方式連接，如此就能夠依序連接完成。而且，只要在3個以上的回路區塊後方持續本指令的配置方式，即可將所有的回路區塊以串聯連接。

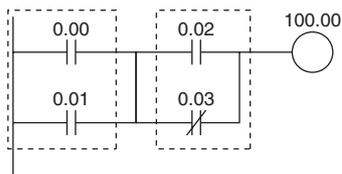
● OR LD

- 如果要將3個以上的回路區塊以串聯連接時，可以利用本指令將2個回路區塊串聯連接，接著再將下一個回路區塊以串聯方式連接，如此就能夠依序連接完成。而且，只要在3個以上的回路區塊後方持續本指令的配置方式，即可將所有的回路區塊以並聯連接。

使用上的注意事項

- 利用AND LD指令或OR LD指令來連接回路區塊時，AND LD/OR LD指令的總數必須等於LD/LD NOT指令的總數-1，否則，就會出現回路錯誤的情形。

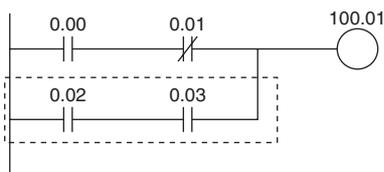
● AND LD



指令	資料
LD	0.00
OR	0.01
LD	0.02
OR NOT	0.03
AND LD	—
OUT	100.00

←另一個區塊開始。

● OR LD

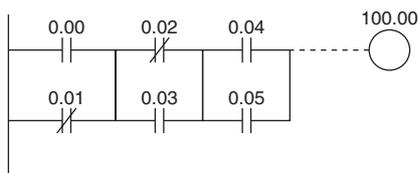


指令	資料
LD	0.00
AND NOT	0.01
LD	0.02
AND	0.03
OR LD	—
OUT	100.01

←另一個區塊開始。

程式例

● AND LD



指令碼①

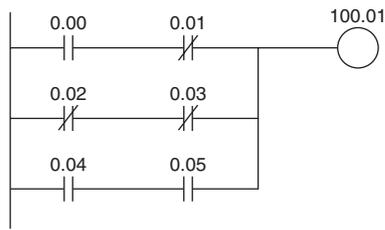
指令	資料
LD	0.00
OR NOT	0.01
LD NOT	0.02
OR	0.03
AND LD	—
LD	0.04
OR	0.05
AND LD	—
:	:
OUT	100.00

指令碼②

指令	資料
LD	0.00
OR NOT	0.01
LD NOT	0.02
OR	0.03
LD	0.04
OR	0.05
:	:
AND LD	—
AND LD	—
:	:
OUT	100.00

- AND LD 指令可持續使用無限次，不過如果利用②的方法來編寫程式時，AND LD 的總數將變成 AND LD 指令前的 (LD 及 LD NOT 總數) -1。
- 使用②的方法時，LD、LD NOT 在 AND LD 前的數量必須 ≤ 8 次。
- 若數量 ≥ 9 次時，必須利用①的方法來編寫程式。
- 一旦數量 ≥ 9 次時，就會造成週邊工具在進行程式檢查時發生回路異常。

● OR LD



指令碼①

指令	資料
LD	0.00
AND NOT	0.01
LD NOT	0.02
AND NOT	0.03
OR LD	—
LD	0.04
AND	0.05
OR LD	—
:	:
OUT	100.01

指令碼②

指令	資料
LD	0.00
AND NOT	0.01
LD NOT	0.02
AND NOT	0.03
LD	0.04
AND	0.05
:	:
OR LD	—
OR LD	—
:	:
OUT	100.01

- OR LD 指令可持續使用無限次，不過如果利用②的方法來編寫程式時，OR LD 的總數將變成 OR LD 指令前的 (LD 及 LD NOT 總數) -1。
- 使用②的方法時，LD、LD NOT 在 OR LD 前的數量必須 ≤ 8 次。
- 若數量 ≥ 9 次時，必須利用①的方法來編寫程式。
- 若因為使用②的方法使得數量 ≥ 9 次時，就會造成週邊工具在進程式檢查時發生回路錯誤。

NOT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
NOT	NOT	—	520	將輸入條件反相。

符號	NOT
	

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

條件旗標的動作

無

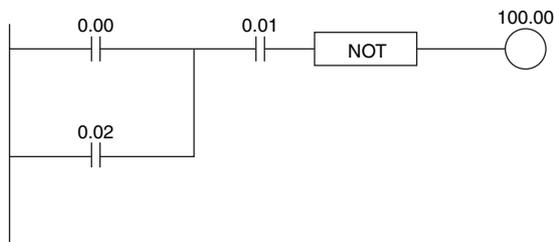
功能

可將輸入條件反相，並連接至下一段指令。

使用上的注意事項

- 請在本指令的最後一段使用輸出類型指令。
- 本指令不適用於回路的最末段。

程式例



可將0.00、0.01、0.02等邏輯條件的演算結果反轉，並輸出至100.00。

0.00	0.01	0.02	100.00
1	1	1	0
1	1	0	0
1	0	1	1
0	1	1	0
1	0	0	1
0	1	0	1
0	0	1	1
0	0	0	1

UP/DOWN

指令名稱	指令記號	指令的各種組合	Fun No.	功能
輸入條件上微分	UP	—	521	當輸入訊號啟動 (OFF → ON) 時，將執行 ON 動作 1 個週期，並且連接至下一段指令。
輸入條件下微分	DOWN	—	522	當輸入訊號啟動 (ON → OFF) 時，將執行 ON 動作 1 個週期，並且連接至下一段指令。

符號	UP	DOWN
		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

條件旗標的動作

無

功能

● UP

輸入條件上微分指令。

● DOWN

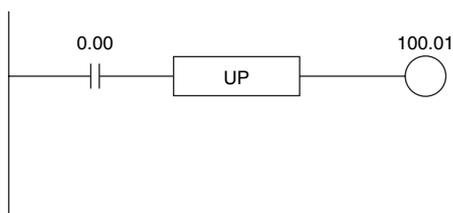
輸入條件下微分指令。

使用上的注意事項

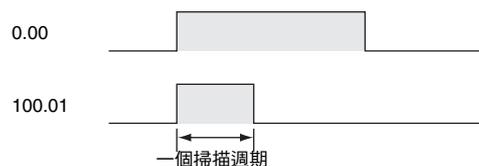
- 請在本指令的最後一段使用輸出類指令。
- 本指令不適用於回路的最末段。
- 如果在 IL-ILC 之間、JMP/CJP-JME 指令之間、副程式之間使用本指令時，有可能會因為指令的執行條件造成動作不穩定，此點需特別注意。
- 請注意，如果在相同週期、相同副程式中多次執行本指令，將因此造成本指令動作不穩定。
- 副程式無法在輸入為 OFF 的程序下執行，因此，如果要在副程式執行本指令的話，必須特別注意。詳細內容請參閱副程式指令 (SBS) 該節。

程式例

● UP



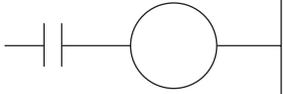
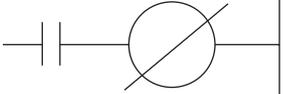
當 0.00 由 OFF → ON 時，只有 1 個掃描週期的 100.01 會變為 ON。



順序輸出指令

OUT/OUT NOT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
輸出	OUT	!OUT	—	繼電器輸出指令。
輸出反相	OUT NOT	!OUT NOT	—	反相輸出指令。

符號	OUT	OUT NOT
		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
—	—	BOOL	—

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
OUT/ OUT NOT	○	○	○	○	—	—	—	—	—	—	—	—	○

條件旗標的動作

無

功能

● OUT

若未指定立即更新時，本指令會將輸入條件 (POWER FLOW) 的內容寫入 I/O 記憶體所指定的位元。

若已經指定為立即更新時，本指令會將輸入條件的內容連同 I/O 記憶體所指定的位元，寫入 CPU 模組內建輸出的實際輸出接點。

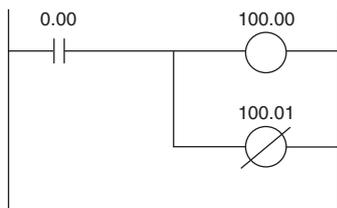
● OUT NOT

若未指定立即更新時，本指令會將輸入條件 (POWER FLOW) 的內容反相後，寫入 I/O 記憶體所指定的位元。

若已經指定為立即更新時，本指令會將輸入條件的內容連同 I/O 記憶體所指定的位元，寫入 CPU 模組內建輸出的實際輸出接點。

提示

- 可指定為立即更新 (!OUT!/OUT NOT)。此時，會在執行本指令後，將先前的輸入條件 (POWER FLOW) 的內容寫入 I/O 記憶體所指定的位元，同時還會對 CPU 模組內建輸出執行 OUT 更新。
- SET/RESET 指令與 OUT 指令的不同點。
OUT 指令所指定的元件編號於輸入條件 ON 的時候 ON、OFF 的時候 OFF。
SET 指令指定的元件編號於輸入條件 ON 的時候 ON、輸入條件 OFF 時，輸出仍繼續保持 ON 的狀態。
RSET 指令指定的元件編號於輸入條件 ON 的時候輸出被復歸成 OFF、輸入條件 OFF 時，輸出仍繼續保持 OFF 的狀態。

程式例

指令	資料
LD	0.00
OUT	100.00
OUT NOT	100.01

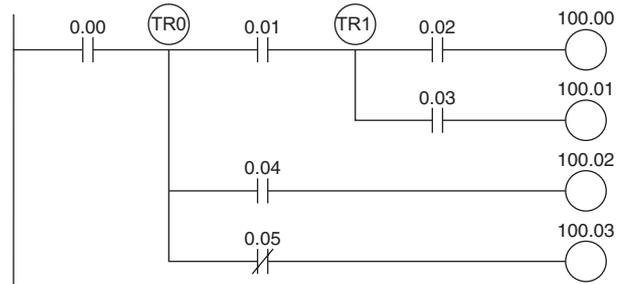
TR

指令名稱	指令記號	指令的各種組合	Fun No.	功能
暫時記憶繼電器	TR	—	—	於指令碼當中，TR 被用來當成分支點的信號記憶。

功能

TR (暫時記憶繼電器) 適用於指令欄程式需要暫時記憶回路執行時的 ON/OFF 狀態。

根據階梯圖，軟體端會自動進行處理，因此不使用本指令。為了讓接下來的說明更簡單易懂，我們將利用階梯圖來說明。



繼電器編號

指令	資料
LD	0.00
OUT	TR0
AND	0.01
OUT	TR1
AND	0.02
OUT	100.00
LD	TR1
AND	0.03
OUT	100.01
LD	TR0
AND	0.04
OUT	100.02
LD	TR0
AND NOT	100.00
OUT	100.03

暫時記憶繼電器	TR0 ~ 15
---------	----------

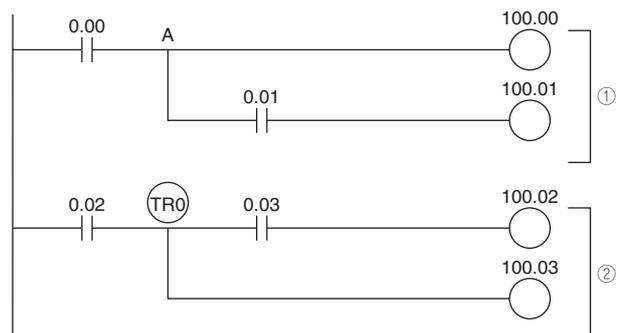
● TR0 ~ 15 的使用方法

- TR0 ~ 15 不適用於 LD、OUT 指令以外的指令。
- TR0 ~ 15 的繼電器編號使用順序並未設定任何限制。
- TR 不需要的回路與重要的回路。

在範例①中，A 點的 ON/OFF 狀態和輸出 100.00 相同，因此可以接在 OUT100.00 後面，繼續編寫 AND0.01、OUT100.01 的程式，而且不需要 TR。

在範例②中，分歧點的內容和 100.02 的輸出內容不一定要相同，因此必須利用 TR 來接收訊號。

如果將②的回路更換為①時，程式的步驟數將會減少。



KEEP

指令名稱	指令記號	指令的各種組合	Fun No.	功能
KEEP	KEEP	!KEEP	011	動作 ON 並保持。



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
R	繼電器編號	BOOL	—

● 運算元種類

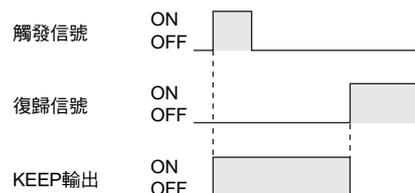
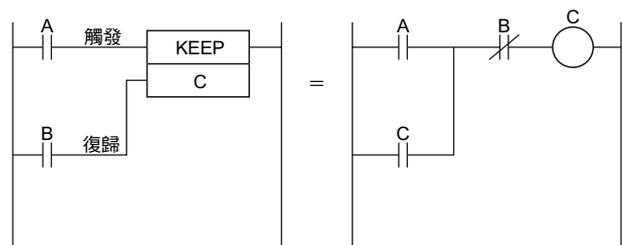
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
R	○	○	○	○	—	—	—	—	—	—	—	—	○

條件旗標的動作

無

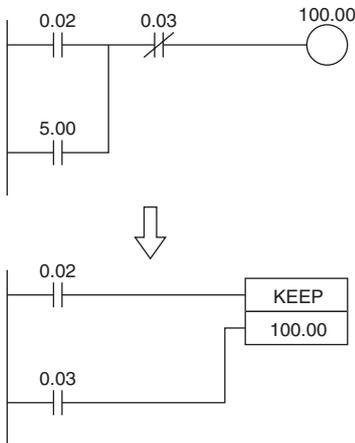
功能

當觸發信號（輸入條件）變為 ON 時，本指令可保持 R 所指定的繼電器 ON 狀態。當復歸信號 ON 時，裝置就會進入 OFF 狀態。

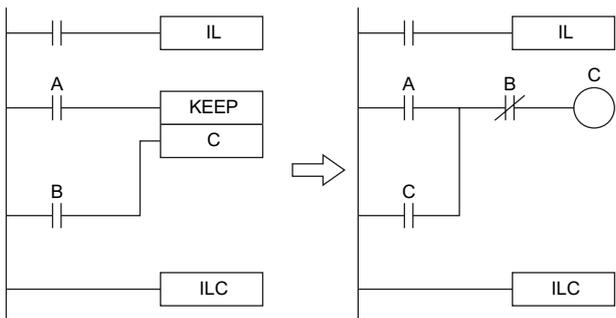


提示

- 當一個外部輸出位元已指定給 !KEEP(O11) 指令中的 R 時，那麼在 !KEEP(O11) 指令執行時，R 的任何變化將被更新並且立即反映到輸出位元。(C200H 群組 2 的多點輸入輸出模組及遠端 I/O 模組的模組除外)
- 使用保持回路的 KEEP 指令，即可將回路更換如下：



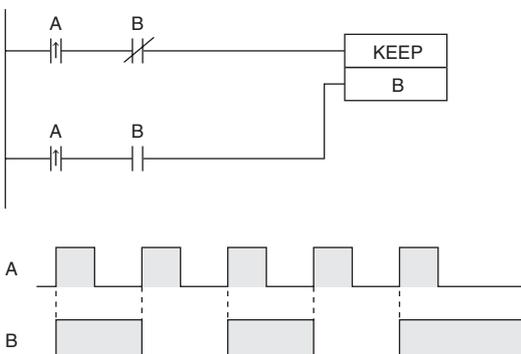
不過，當上述回路出現在 IL-ILC 之間且 IL 條件 OFF 時，使用輸出繼電器的回路 (上方的回路) 其輸出 100.00 就會變為 OFF，而使用 KEEP 指令的回路則會保持原來的狀態，此點需特別注意。



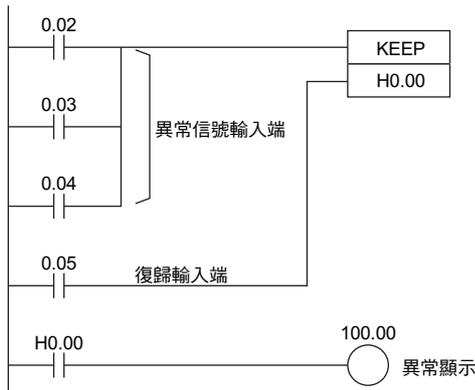
IL 條件 OFF 時，C 的狀態被保持

IL 條件 OFF 時，C 的狀態變成 OFF

- 使用 KEEP 指令，所製作的單 ON/ 雙 OFF 回路 (Flip-Flop)。



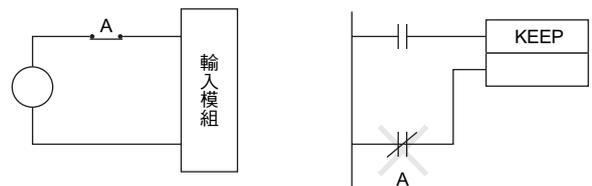
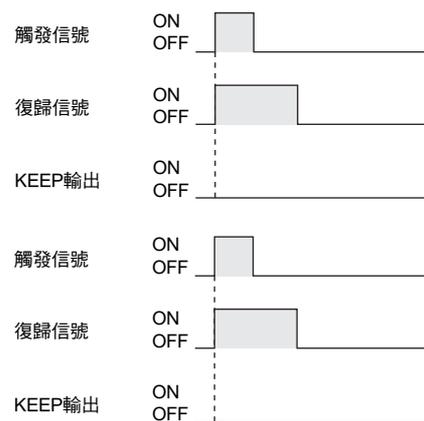
- KEEP 指令的輸出，如果指定保持繼電器的話，停電時，PLC 可將停電前的 ON/OFF 狀態加以記憶。
- < 解決停電問題之異常顯示回路範例 >



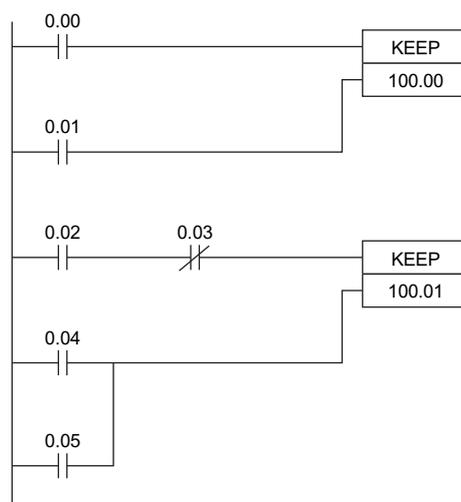
- 當 I/O 記憶體保持旗標變為 ON，並且利用 PLC 系統設定方式將 I/O 記憶體保持旗標設定為保持狀態時，(由於輸出輸入繼電器在停電時仍然會保持斷電前的狀態) 此時只要利用 KEEP 指令來驅動輸出入繼電器，就能夠和保持記憶體一樣，即使停電時也能記憶即將停電前的狀態。另外，需要特別注意的一點就是當 PLC 系統設定完成後，本指令就會從下次開啟電源開始執行動作。

使用上的注意事項

- 當觸發信號(輸入條件)及復歸信號同時 ON 時，系統將優先處理復歸信號。
- 當復歸信號條件 ON 時，將無法同時接受復歸信號(輸入條件)。
- 請勿直接使用 PLC 外部信號的 b 接點來當成 KEEP 指令的復歸端信號。當 PLC 的 AC 電源斷電或瞬時停電時，PLC 的內部電源並不會立即 OFF，反而是輸入模組的電源先 OFF 使得輸入信號 OFF。其結果，導致 b 接點的外部信號 ON、復歸信號 ON、KEEP 指令的輸出變成 OFF。



程式例



當0.00為ON時，100.00將保持ON狀態。

當0.01為ON時，100.00就會被OFF。

當0.02為ON時，0.03為OFF時，100.01將保持ON狀態。

當0.04或0.05為ON時，100.01將保持OFF狀態。

指令	資料
LD	0.00
LD	0.01
KEEP(011)	100.00
LD	0.02
AND NOT	0.03
LD	0.04
OR	0.05
KEEP(011)	100.01

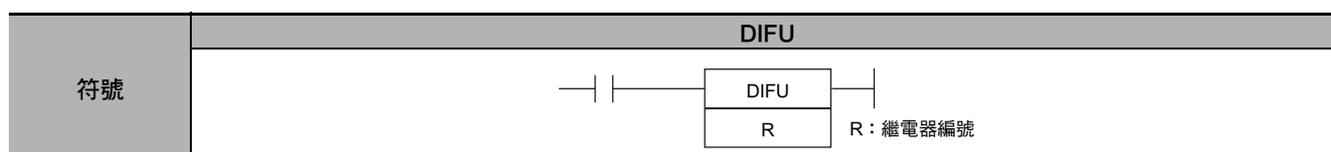
(註) KEEP 指令的輸入順序將依階梯圖及指令欄而異。

階梯圖：觸發端→KEEP 指令→復歸端。

指令記號：觸發端→復歸端→KEEP 指令。

DIFU

指令名稱	指令記號	指令的各種組合	Fun No.	功能
上微分	DIFU	!DIFU	013	輸入條件由 OFF → ON 變化時，指令所指定的元件編號 ON 一次掃描週期



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
R	繼電器編號	BOOL	—

● 運算元種類

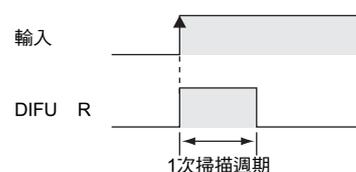
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
R	○	○	○	○	—	—	—	—	—	—	—	—	—

條件旗標的動作

無

功能

當輸入訊號上升 (OFF → ON) 時，R 所指定的接點將會持續 ON 動作 1 次掃描週期，週期結束後，只要執行本指令就會變為 OFF。



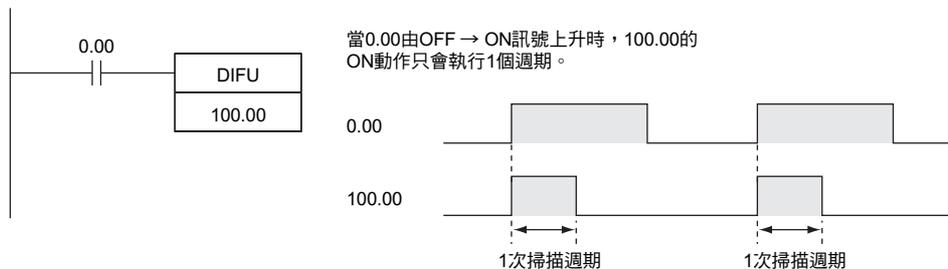
提示

- 不想透過內部補助繼電器來產生上微分信號，希望直接輸出上微分信號時，請使用 UP 指令。
- DIFU 指令可作為立即更新型指令 (!DIFU)，立即更新型指令時，R 必須指定輸出端的編號。所以，R 指定輸出端編號的話，R=ON 時，CPU 執行輸出更新動作。(R=ON 時，輸出繼電器也是 ON 一次掃描週期)

使用上的注意事項

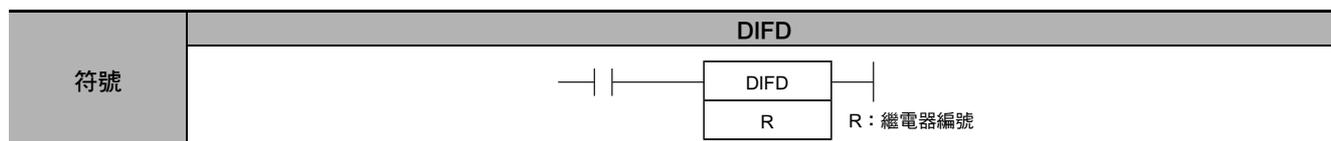
- 於 IL-ILC 之間、JMP-JME 之間或者是副程式裡面使用本指令的話，輸出結果並不一定會正確的隨著輸入條件的變化而變化，此點請注意。
- 副程式於主指式不呼叫時，副程式內容呈現“不執行”狀態，因此，於副程式中使用到本指令時，請注意。
- 同一次掃描週期內，多次呼叫同一個副程式時，本指令的輸出動作並不一定會正確。

程式例



DIFD

指令名稱	指令記號	指令的各種組合	Fun No.	功能
下微分	DIFD	!DIFD	014	當輸入訊號變化 ON → OFF 時，您所指定的接點將會持續 ON 動作一次掃描時間。



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
R	繼電器編號	BOOL	—

● 運算元種類

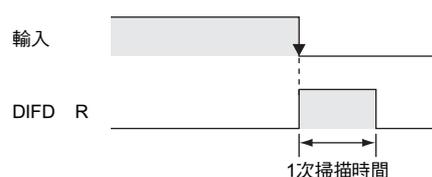
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
R	○	○	○	○	—	—	—	—	—	—	—	—	—

條件旗標的動作

無

功能

輸入條件由 ON → OFF 下降變化時，R 指定的元件編號 ON、於下次掃描執行至本指令時 OFF，也就是說，R 指定的元件編號 ON 一次掃描週期。



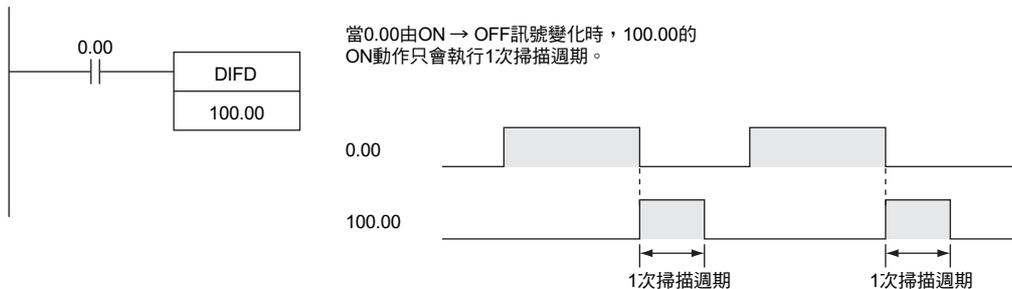
提示

- 不想透過內部補助繼電器來產生下微分信號，希望直接輸出下微分信號時，請使用 DOWN 指令。
- DIFD 指令可作為立即更新型指令 (!DIFD)，立即更新型指令時，R 必須指定輸出端的編號。所以，R 指定輸出端編號的話，R=ON 時，CPU 執行輸出更新動作。(R=ON 時，輸出繼電器也是 ON 一次掃描週期)

使用上的注意事項

- 於 IL-ILC 之間、JMP-JME 之間或者是副程式裡面使用本指令的話，輸出結果並不一定會正確的隨著輸入條件的變化而變化，此點請注意。
- 副程式於主指式不呼叫時，副程式內容呈現“不執行”狀態，因此，於副程式中使用到本指令時，請注意。
- 同一次掃描週期內，多次呼叫同一個副程式時，本指令的輸出動作並不一定會正確。

程式例



SET/RSET

指令名稱	指令記號	指令的各種組合	Fun No.	功能
強制 ON	SET	@SET, %SET, !SET, !@SET, !%SET	—	當輸入條件 ON 時，您所指定的接點就會被設定為 ON，接下來，無論輸入條件是否為 ON/OFF，您所指定的接點將保持 ON 狀態。
強制 OFF	RSET	@RSET, %RSET, !RSET, !@RSET, !%RSET	—	當輸入條件 ON 時，您所指定的接點就會被復歸為 OFF。接下來，無論輸入條件是否為 ON/OFF，您所指定的接點將保持 OFF 狀態。

符號	SET	RSET
	 R：繼電器編號	 R：繼電器編號

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
R	繼電器編號	BOOL	—

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
R	○	○	○	○	—	—	—	—	—	—	—	—	—

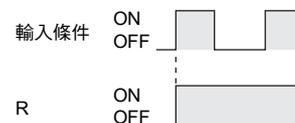
條件旗標的動作

無

功能

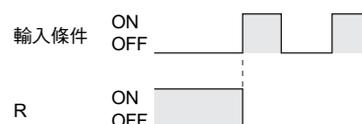
● SET

當輸入條件 ON 時，R 所指定的接點就會被設定為 ON。此時，無論輸入條件為 OFF 或是 ON，您所指定的接點 R 將保持 ON 狀態，如果要設定為 OFF 時，請使用 RSET 指令。



● RSET

當輸入條件 ON 時，R 所指定的接點就會被 OFF。此時，無論輸入條件為 OFF 或是 ON，您所指定的接點 R 將保持 OFF 狀態，使用 SET 指令，即可設定為 ON。

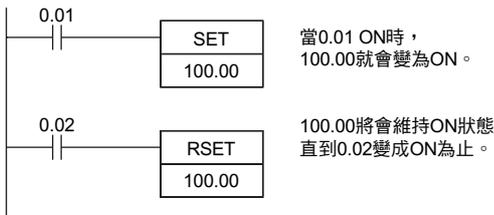
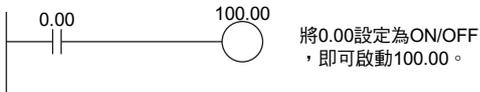


提示

- SET/RESET 指令與 OUT 指令之差異。

SET/RSET 指令只有在輸入條件 ON 時，才會將指定接點 ON/OFF，當輸入條件變為 OFF 時，指定接點的 ON/OFF 狀態不會改變。

OUT 指令會在輸入條件 ON 時，將指定接點 ON，當輸入條件 OFF 時，就會將指定接點 OFF。



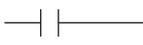
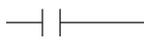
- 使用 KEEP 指令時必須同時指定觸發端及復歸端，但是，SET/RSET 指令卻可單獨使用，此外，也可以對同一個輸出點使用多個 SET 指令或 RSET 指令。
- SET/RSET 指令可作為立即更新型指令 (!SET!/RSET)，立即更新型指令時，R 必須指定輸出端的編號。所以，R 指定輸出端編號的話，R=ON 或 OFF(指令被執行)時，CPU 執行輸出更新動作。

使用時的注意事項

- CJ2 CPU 模組的 SET/RSET 指令可指定資料暫存器 (DM)、擴充暫存器 (EM) 的位元當成輸出的元件編號。CJ2 以外的 CPU 模組則不支援此項功能，但是可使用 SETB/RSETB 指令來達到相同的功能。
- SET/RSET 指令被使用於 IL-ILE/JMP-JME 回路當中時，IL 條件或 JMP 條件 OFF 的話，本指令所指定輸出點的 ON/OFF 狀態沒有變化。

SETA/RSTA

指令名稱	指令記號	指令的各種組合	Fun No.	功能
多位元強制 ON	SETA	@SETA	530	指定連續的多個位元 ON。
多位元強制 OFF	RSTA	@RSTA	531	指定連續的多個位元 OFF。

符號	SETA	RSTA							
	 <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>SETA</td></tr> <tr><td>D</td></tr> <tr><td>N1</td></tr> <tr><td>N2</td></tr> </table>	SETA	D	N1	N2	 <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>RSTA</td></tr> <tr><td>D</td></tr> <tr><td>N1</td></tr> <tr><td>N2</td></tr> </table>	RSTA	D	N1
SETA									
D									
N1									
N2									
RSTA									
D									
N1									
N2									
	D：設定CH編號 N1：開始位元位置 N2：位元數	D：設定CH編號 N1：開位元位置 N2：位元數							

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D	設定 CH 編號	UINT	可調整
N1	開始位元位置	UINT	1
N2	位元數	UINT	1

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
D	○	○	○	○	○	○	○	○	○	—	—	—	—
N1,N2	○	○	○	○	○	○	○	○	○	○	—	—	—

條件旗標的動作

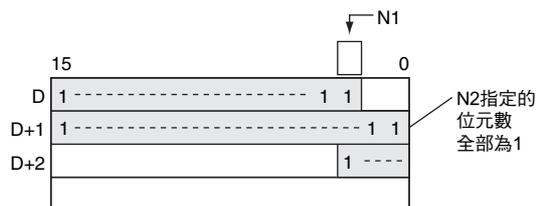
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 N1 的資料以 10 進位表示時，超過 0 ~ 15 或是以 16 進位表示時，超過 #0000 ~ 000F 的範圍時，就會變為 ON 否則為 OFF

功能

● SETA

輸入條件 ON 時，D 所指定的 CH 編號、從 N1 指定的位元編號開始，共 N2 個位元的內容全部為 1。範圍外的位元內容不變。當 N2 的內容為 0 時，全部位元內容沒有變化。

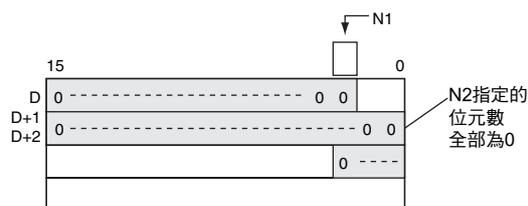
使用本指令來強制多點位元 ON，強制多點位元 OFF 時，不只可以使用 RSTA 指令，亦可使用其他的指令來執行。



● RSTA

輸入條件 ON 時，D 所指定的 CH 編號、從 N1 指定的位元編號開始，共 N2 個位元的內容全部為 0。範圍外的位元內容不變。當 N2 的內容為 0 時，全部位元內容沒有變化。

使用本指令來強制多點位元 OFF，強制多點位元 ON 時，不只可以使用 SETA 指令，亦可使用其他的指令來執行。



提示

● SETA

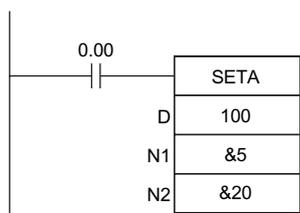
在 SETA (多位元強制 ON) 指令中，您也可以針對以資料記憶體的 CH 為處理單位的運算元種類，將指定範圍內的位元區全部設定為 ON。

● RSTA

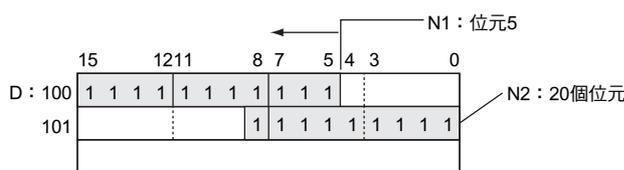
在 RSTA (多位元強制 OFF) 指令中，您也可以針對以資料記憶體的 CH 為處理單位的運算元種類，將指定範圍內的位元區全部設定為 OFF。

程式例

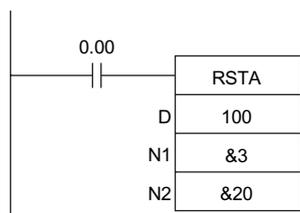
● SETA



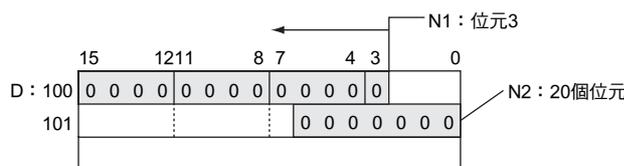
當 0.00=ON 時，100CH 內的位元 5 開始算的 20 (16 進制#14) 個位元的內容強制為 1。



● RSTA

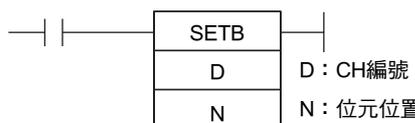
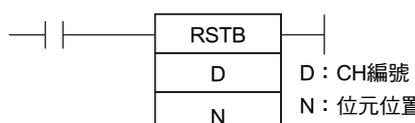


當 0.00=ON 時，100CH 內的位元 3 開始算的 20 (16 進制#14) 個位元的內容強制為 0。



SETB/RSTB

指令名稱	指令記號	指令的各種組合	Fun No.	功能
1 位元組	SETB	@SETB, !SETB, !@SETB	532	指定 CH 當中的 1 個位元 ON。
1 位元復歸	RSTB	@RSTB, !RSTB, !@RSTB	533	指定 CH 當中的 1 個位元 OFF。

符號	SETB	RSTB
	 <p>D : CH編號 N : 位元位置</p>	 <p>D : CH編號 N : 位元位置</p>

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D	CH 位址	UINT	1
N	位元位置	UINT	1

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
D	○	○	○	○	○	○	○	○	○	—	—	—	—
N	○	○	○	○	○	○	○	○	○	○	—	—	—

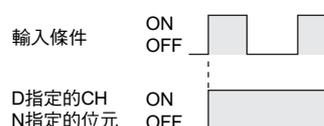
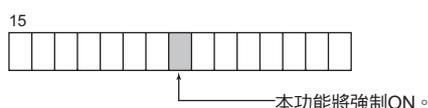
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 N 以 10 進位表示時為 0 ~ 15 或是以 16 進位表示時為 #0000 ~ 000F 以外的數值時，本旗標就會 ON 否則為 OFF

功能

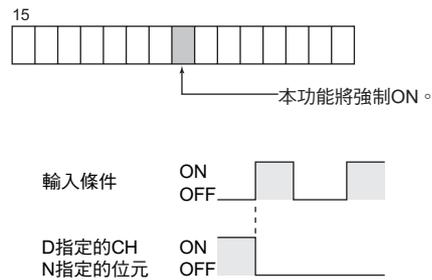
● SETB

當輸入條件變為 ON 時，D 所指定的 CH 的位元位址 N 就會被設定為 ON。即使輸入條件變為 OFF，D 所指定的 CH 的位元位址 N 仍然不會發生變化。



● RSTB

當輸入條件變為 ON 時，D 所指定的 CH 的位元位址 N 就會被 OFF。即使輸入條件變為 OFF 或是 ON，指定接點 D 所指定的 CH 的位元位址 N 仍然會保持 OFF 狀態。使用 SETB 指令，即可設定為 ON。



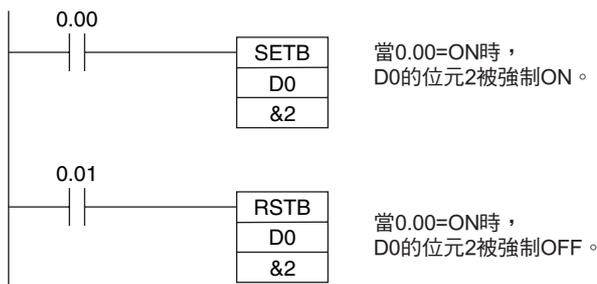
提示

- SETB/RSTB 指令與 SET/RST 指令之差異。
本功能對於 CIO、內部輔助繼電器 (W)、保持繼電器 (H)、特殊輔助繼電器 (A) 而言為相同的功能。
SETB/RSTB 指令可針對 SET/RST 指令所無法指定的資料記憶體 (DM) 的指定位元進行強制 ON/ 強制 OFF。
- KEEP 指令中的觸發信號及復歸信號必須在相同的位置敘述，而 SETB/RSTB 指令則可分別敘述，而且，還可以將多個 SETB/RSTB 指令用在相同位址的輸出繼電器。

使用上的注意事項

- SETB/RSTB 指令不需配對使用 (例如: 如果要復歸 SETB 指令所設定的位元時，並不一定非要使用 RSTB 指令才能達成)。
- 本頁所提到的 SETB/RSTB 指令無法執行計時器、計數器強制 ON/ 強制 OFF 動作。
- 在 IL-ILC/JMP-JME 指令區內使用 SETB/RSTB 指令，或是當輸入條件 OFF 時，您所指定的輸出接點狀態將不會發生任何變化。
- 變為立即更新型指令 (!SETB!/RSTB) 後，即可在 R 必須指定輸出端的編號。
- 利用 !SETB (或是 !RSTB 指令) 將 D CH 的位元位址 N 指定為 CPU 模組內建輸出時，這時候已經被設定為 ON (或 OFF) 的 D CH 位元位址 N 就會隨時 (執行指令時) 執行 OUT 更新動作。已經被設定為 ON (或 OFF) 的 D CH 位元位址 N 會和平常一樣，持續 ON 的狀態，直到執行 RSTB 指令 (或 SETB 指令) 為止。

程式例



順序控制指令

互鎖 (INTERLOCK) 指令

● 互鎖指令的種類

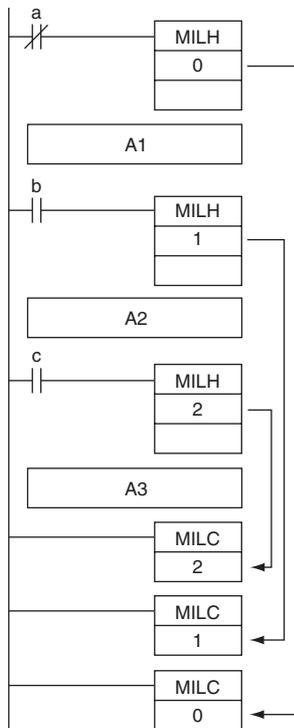
可執行互鎖輸出的指令包含下列幾種：

- 互鎖 (IL-ILC) 指令
- 多重互鎖 (MILH-MILC 或 MILR-MILC) 指令
 - 多重互鎖 (微分旗標保持型) (MILH-MILC) 指令
 - 多重互鎖 (微分旗標非保持型) (MILR-MILC) 指令

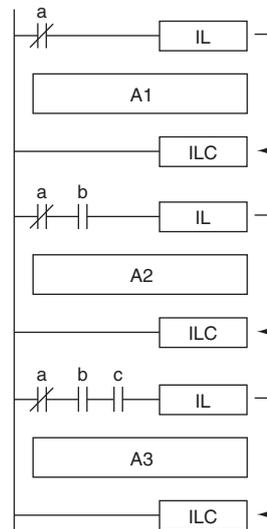
● 互鎖 (IL-ILC) 指令和多重互鎖 (MILH-MILC 或 MILR-MILC) 指令之相異點

互鎖 (IL-ILC) 指令無法執行巢狀模式，相對地，多重互鎖 (MILH-MILC 或 MILR-MILC) 指令卻能夠執行巢狀模式，因此，只要參閱下圖即可發現，相較於 IL-ILC 指令，多重互鎖指令在指令使用上更簡單。

MILH-MILC 指令



利用 IL-ILC 指令時



● 多重互鎖(微分旗標保持型) (MILH)指令與多重互鎖(微分旗標非保持型) (MILR)指令之相異點

當 MILH 指令與 MILR 指令在 MILH-MILC 指令、MILR-MILC 指令之間，有存在微分指令 (附 DIFU/DIFD/@/% 指令) 時，此時微分指令的動作將有所不同。

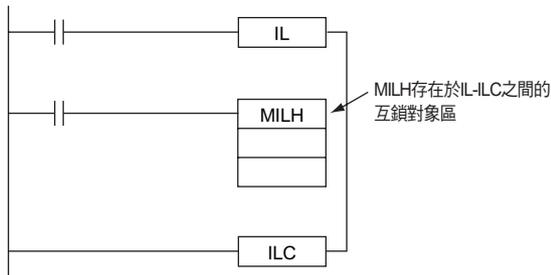
又，MILH 指令在微分指令的動作上和 IL 指令相同。

詳細內容請參閱 MILH/MILR/MILC 指令說明中的「MILH 指令與 MILR 指令之相異點」該節。

● 使用限制

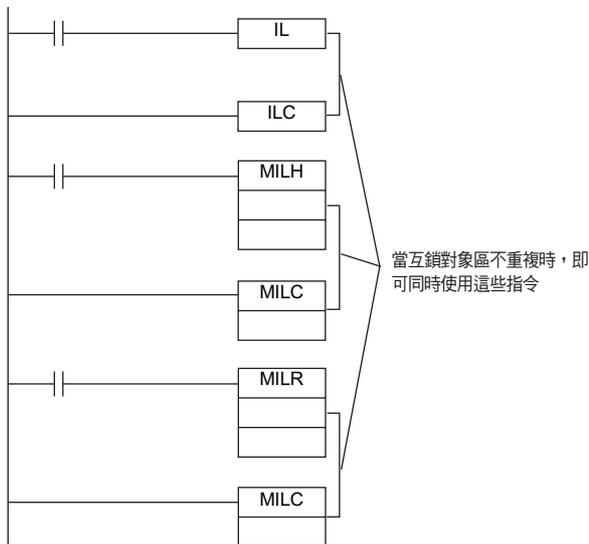
IL-ILC 指令、MILH-MILC 指令及 MILR-MILC 指令不得互相搭配使用，否則，將無法保證回路是否正常動作。又，關於互相搭配時所產生的動作，請參閱 MILH/MILR/MILC 指令的相關說明項目。

例) IL-ILC 指令區塊不得配置 MILH 指令。



(註)只要 IL-ILC 指令、MILH-MILC 指令、MILR-MILC 指令的每個互鎖對象區不重複，即可同時使用前述指令。

例) 可在不重複 IL-ILC、MILH-MILC、MILR-MILC 的條件下進行配置。



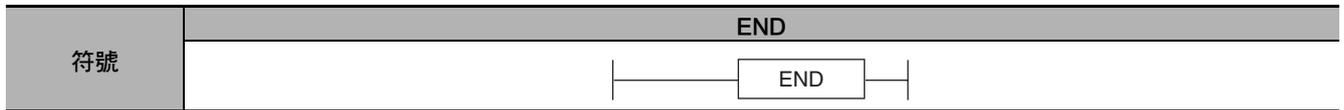
● 互鎖指令與跳躍指令之相異點

IL-ILC、MILH-MILC、MILR-MILC 指令與 JMP-JME 指令的相異之處如下：

指令	執行指令	所有指令輸出	OUT、OUTB、OUT NOT 指令所指定的繼電器	計時器類指令 (TTIM/TTIMX、MTIM/MTIMX 除外)
IL-ILC 指令 MILH-MILC 指令 MILR-MILC 指令	不執行所有指令 (除 OUT、OUT NOT、計時器類指令以外的)	所有指令輸出將保持先前的狀態 (除 OUT、OUT NOT、計時器類指令以外的)	OFF	復歸
JMP-JME 命令	不執行所有指令	所有指令輸出將保持先前的狀態	保持先前的狀態	啟動中的計時器 (只有 TIM/TIMX、TIMH/TIMHX、TMHH/TMHH 指令) 保持計時動作，即使在 JMP 條件 OFF 時，也會進行現在值更新處理，因此會持續計時。

END

指令名稱	指令記號	指令的各種組合	Fun No.	功能
END	END	—	001	表示某個程式結束。



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	—	—	○

條件旗標的動作

無

功能

執行本指令，即可結束該程式在週期內的執行動作，因此，END 指令以後的指令將不會被執行。

使用上的注意事項

- 請務必在每個程式最後加上 END 指令。當 END 指令不存在時，就會發生程式錯誤。

NOP

指令名稱	指令記號	指令的各種組合	Fun No.	功能
無處理	NOP	—	000	無任何功能的指令。

符號	NOP (不會顯示在階梯圖上。)

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

條件旗標的動作

無

功能

本指令不具備任何功能，本指令只有在“語句表”的模態下才被使用。

提示

- 於“語句表”的模態下，希望程式回路編號的開始點為整數位址時 (10、50、100、1000 等)，插入 NOP 指令做區隔。

IL/ILC

指令名稱	指令記號	指令的各種組合	Fun No.	功能
互鎖	IL	—	002	互鎖回路的開始。
清除互鎖	ILC	—	003	互鎖回路的結束。

符號	IL	ILC

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	—	○	○

條件旗標的動作

無

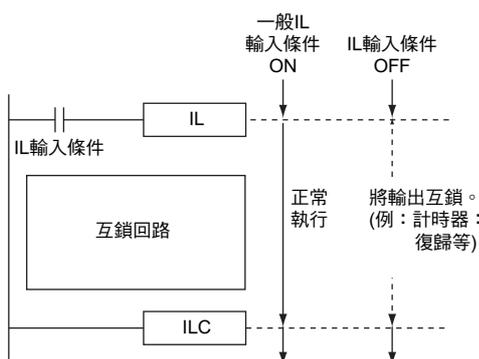
功能

當 IL 指令的輸入條件 OFF 時，可以讓 IL 指令 ~ ILC 指令之間的所有指令輸出互鎖。當 IL 指令的輸入條件 ON 時，可以讓 IL 指令 ~ ILC 指令之間的所有指令正常執行。

互鎖 (IL) 狀態下的所有指令輸出如下：

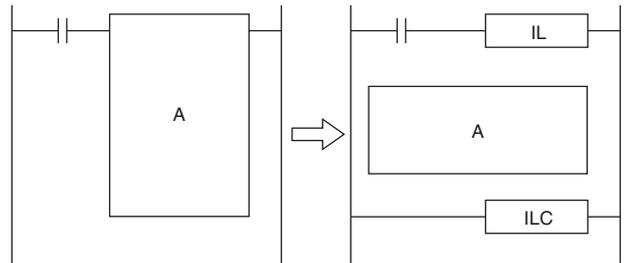
各命令出力	狀態	
OUT、OUT NOT 指令所指定的繼電器	全部 OFF	
計時器類指令 (100 ms 計時器 TIM/TIMX、10 ms 計時器 TIMH/TIMHX、1 ms 計時器 TMHH/TMHHX、長時間計時器 TIML/TIMLX 指令)	現在值	計時器設定值 (復歸)
	時間到旗標	OFF (復歸)
其他指令 (註) 所指定的繼電器、CH	保持先前的狀態 (指令本身不會執行任何動作)	

(註) 表示 TTIM/TTIMX、SET、RSET、CNT/CNTX、CNTR/CNTRX、SFT、KEEP 等所有其他的指令。



提示

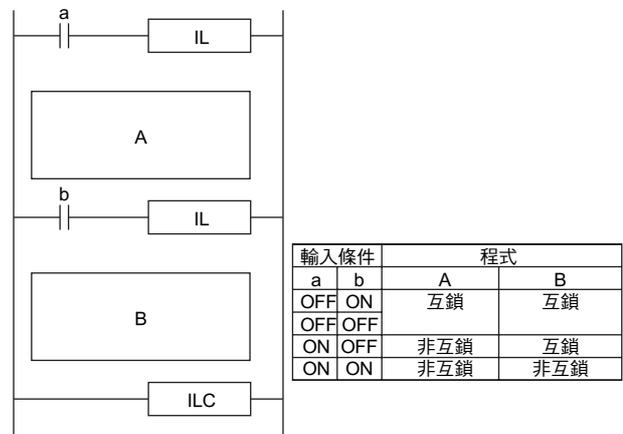
- 使用 IL-ILC 指令，即可在程式中完成更高效率的回路切換。
- 執行 IL 指令時，也能夠在 IL 指令前，使用 SET 指令，如此就能夠將您希望保持 ON 的指令區預先設定為 ON。在相同的輸入條件下，執行多項處理動作時，只要分別將 IL 指令及 ILC 指令放在這些動作的正前方及正後方，即可精簡程式部數。



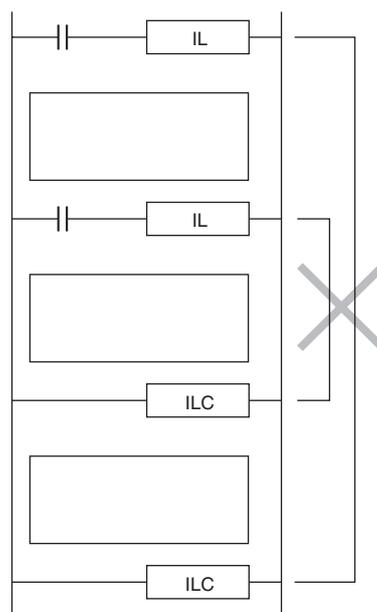
使用上的注意事項

- 即使系統因為 IL 指令而進入互鎖狀態，IL ~ ILC 之間程式仍然會在內部繼續執行，因此無法縮短週期時間。

- IL 指令和 ILC 指令必須一對一使用。否則 (當 IL 指令及 ILC 指令之間存在 IL 指令時)，就會在程式檢查時出現 IL-ILC 錯誤，不過，這時候程式仍然會如右圖所示繼續執行動作。



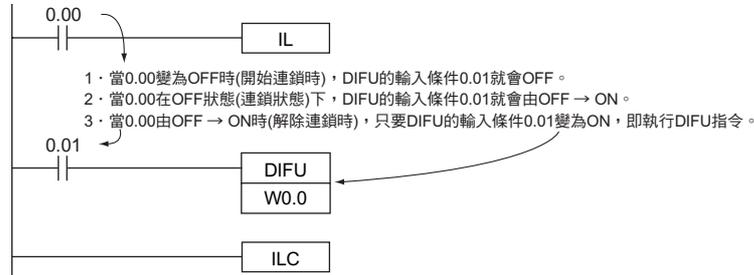
- IL-ILC 指令不可使用巢狀結構。欲使用巢狀結構時，請使用 MLH-MILC 或 MILR-MILC 指令。



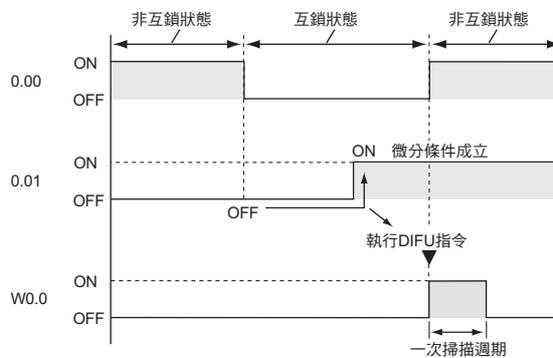
• 微分指令所對應的動作

當 IL-ILC 指令之間存在微分指令 (附加 DIFU/DIFD/@/% 指令) 時，一旦微分指令的輸入條件在開始互鎖到解除的這段期間發生變化，使得微分條件成立時，只要互鎖解除，微分指令就會開始執行。

例：執行上微分 (DIFU) 指令時，一旦開始互鎖的輸入條件 OFF 而且解除互鎖的輸入條件 OFF，上微分 (DIFU) 指令會在解除互鎖時上升並且執行動作。

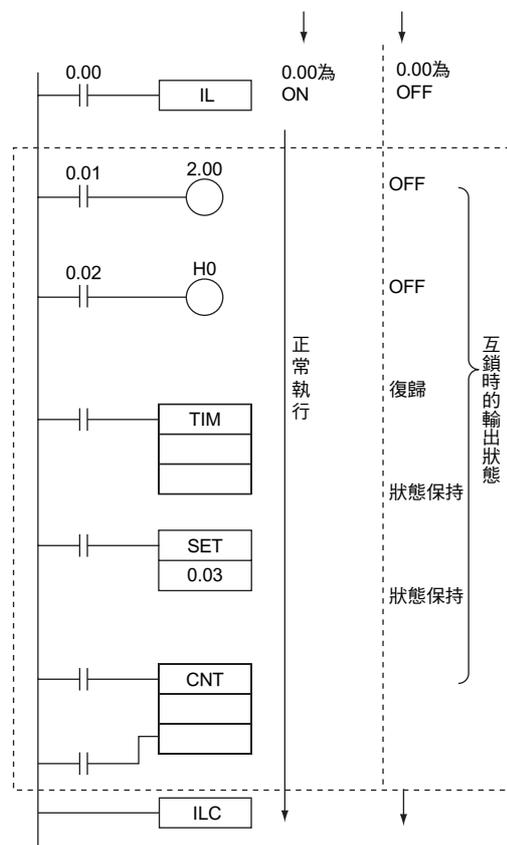


參考：IL 指令在微分指令的動作上，和 MILH 指令的動作相同。



程式例

當 0.00 變為 OFF，就會將 IL ~ ILC 之間的輸出互鎖。
當 0.00 變為 ON 時，IL ~ ILC 之間的指令將正常執行。



MILH/MILR/MILC

指令名稱	指令記號	指令的各種組合	Fun No.	功能
多重互鎖 (微分旗標保持型)	MILH	—	517	當輸入條件 OFF 時，可以讓 MILH 指令 ~ MILC 指令之間的輸出互鎖。
多重互鎖 (微分旗標非保持型)	MILR	—	518	當輸入條件 OFF 時，可以讓 MILR 指令 ~ MILC 指令之間的輸出互鎖。
多重互鎖結束	MILC	—	519	互鎖回路結束。

符號	MILH	MILR	MILC

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	—	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
N	互鎖編號	—	1
D	互鎖狀態輸出位元	BOOL	—

N：互鎖編號

0 ~ 15

請將成對的 MILH (或 MILR) 指令及 MILC 指令的 N (互鎖編號) 設定為相同的號碼。

(註) N (互鎖編號) 的使用順序並無限制。

D：互鎖狀態輸出位元

- 非互鎖狀態下 ON。
- 互鎖狀態下 OFF。

利用 MILH (或 MILR) 指令執行互鎖時，只要強制設定此位元，即可轉換為非互鎖 (IL) 狀態，相反地，如果在非互鎖狀態下強制設定此位元，亦可轉換回互鎖 (IL) 狀態。

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
N	—	—	—	—	—	—	—	—	—	○	—	—	—
D	○	○	○	○	—	—	—	—	—	—	—	—	—

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF

功能

當互鎖編號 N 的 MILH (或 MILR) 指令的輸入條件 OFF 時，即可將 MILH (或 MILR) 指令到相同互鎖編號 N 的 MILC 指令等所有的指令輸出互鎖。

當互鎖編號 N 的 MILH (或 MILR) 指令的輸入條件 ON 時，MILH (或 MILR) 指令到相同互鎖編號 N 的 MILC 指令等所有的指令將正常執行動作。

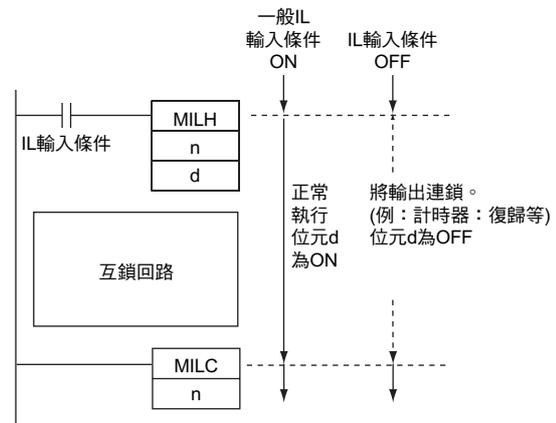
● 互鎖 (IL) 狀態

互鎖 (IL) 狀態下的所有指令輸出如下：

各命令出力	狀態	
OUT、OUT NOT 指令所指定的繼電器	全部 OFF	
計時器類指令 (100 ms 計時器 TIM/TIMX、10 ms 計時器 TIMH/TIMHX、1 ms 計時器 TMHH/TMHHX、長時間計時器 TIML/TIMLX 指令)	現在值	計時器設定值 (復歸)
	時間到旗標	OFF (復歸)
其他指令 (註) 所指定的繼電器、CH	保持先前的狀態 (指令本身不會執行任何動作)	

(註) 表示 TTIM/TTIMX、SET、RSET、CNT/CNTX、CNTR/CNTRX、SFT、KEEP 等所有其他的指令。

又，MILH 指令 (或 MILR 指令) 的第 2 運算元種類 D (互鎖狀態輸出位元) 會在互鎖狀態下輸出 OFF，以及非互鎖狀態下輸出 ON。因此，只要監控此位元，即可監控每個互鎖編號的互鎖 (IL) 狀態。



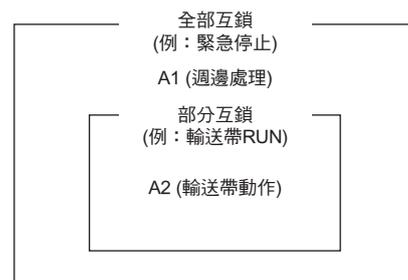
● 巢狀結構

所謂的巢狀結構是指 MILH (MILR)-MILC 指令間再使用 MILH (MILR)-MILC 指令。

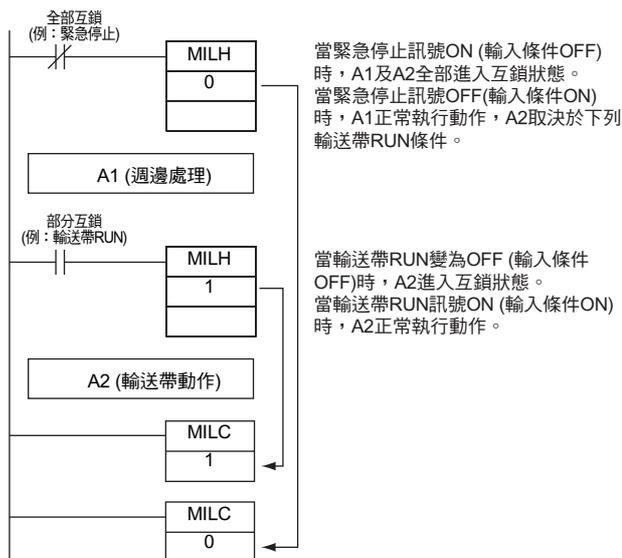
MILH (MILR)-MILC 指令間的巢狀結構加入 N 編號作區別 (例：MILH0-MILH1-MILC1-MILC0) 最多 16 層。

巢狀結構用途如下：

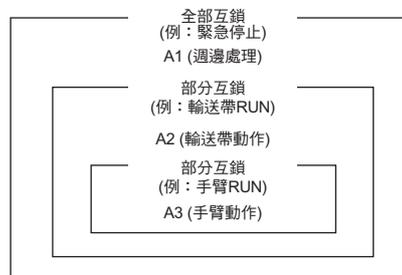
例 1) 與全體回路作互鎖、與部分回路作互鎖各自獨立時。(巢狀結構：1 層)



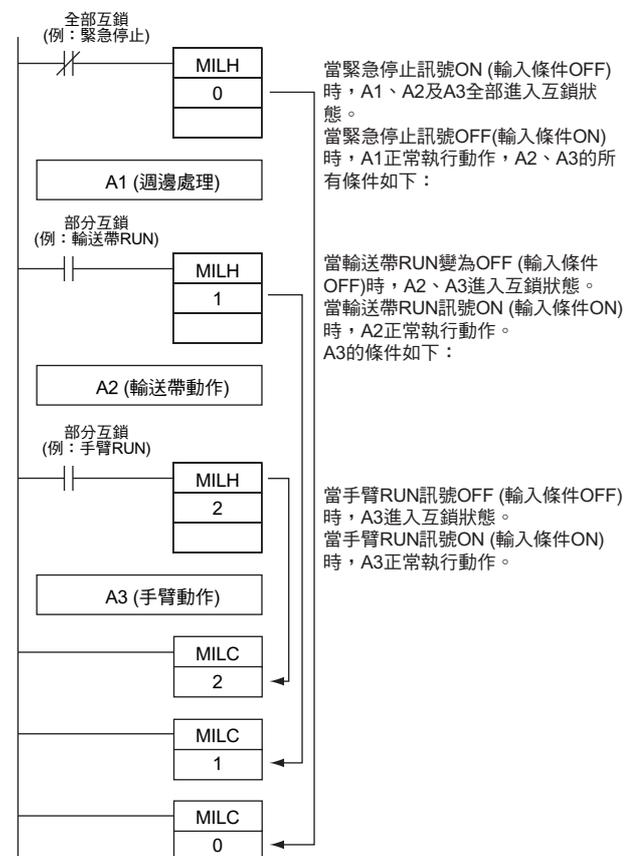
- 緊急停止鍵 ON 時，將 A1、A2 互鎖
- 輸送帶 RUN 訊號 OFF 時，將 A2 互鎖



例 2) 與全體回路作互鎖又有需要部分回路作互鎖。(巢狀結構：2 層)



- 緊急停止鍵 ON 時，將 A1、A2、A3 互鎖
- 輸送帶 RUN 訊號 OFF 時，將 A2、A3 互鎖
- 手臂 RUN 訊號 OFF 時，將 A3 互鎖



● MILH 指令與 MILR 指令之相異點

當 MILC 指令前的所有指令中存在微分動作 (附加 DIFU/DIFD/@/% 指令) 時, MILH 指令和 MILR 指令的動作將有所不同。

使用 MILR 指令時, 即使在互鎖狀態下微分條件成立, 所成立的條件仍然會被取消, 互鎖解除後, 也不會執行微分指令。

相對地, 使用 MILH 指令時, 只要因為互鎖開始及解除時的數值使得微分條件成立, 那麼在互鎖解除 (附加 DIFU/DIFD/@/% 指令) 後, 就會開始執行微分指令。

所有指令	由於輸入條件在互鎖開始及解除時發生變化, 使得微分指令 (附加 DIFU/DIFD/@/% 指令) 中的微分條件成立時
MILH (多重互鎖 (微分旗標保持型)) 指令	互鎖解除後, 即開始執行微分指令。
MILR (多重互鎖 (微分旗標非保持型)) 指令	互鎖解除後, 即停止執行微分指令。

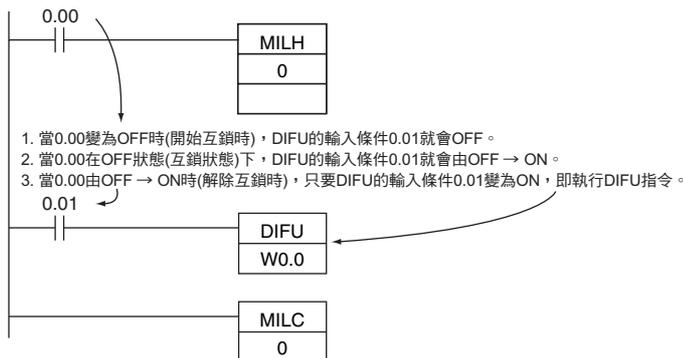
● MILH 指令中出現微分指令時之對應動作

當 MILH-MILC 指令之間存在微分指令 (附加 DIFU/DIFD/@/% 指令) 時, 只要因為輸入條件在開始互鎖時的數值到解除互鎖時的數值範圍內, 因而使得微分條件成立, 那麼當互鎖解除時就會開始執行微分指令。同樣地, 如果微分條件在互鎖開始或解除時的同一個時間點成立時, 就會開始執行微分指令。

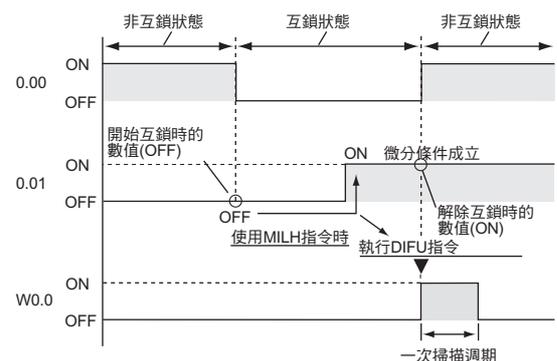
不過, 即使微分條件在互鎖狀態下成立, 如果因為程式的各項條件, 或是因為解除互鎖而造成微分條件不成立時, 此時將不會執行微分指令。

例: 執行上微分 (DIFU) 指令時, 一旦開始互鎖的輸入條件 OFF 而且解除互鎖的輸入條件 ON, 上微分 (DIFU) 指令會在解除互鎖時上升並且執行動作。

(又, IL 指令在微分指令的動作上和 MILH 指令相同。)



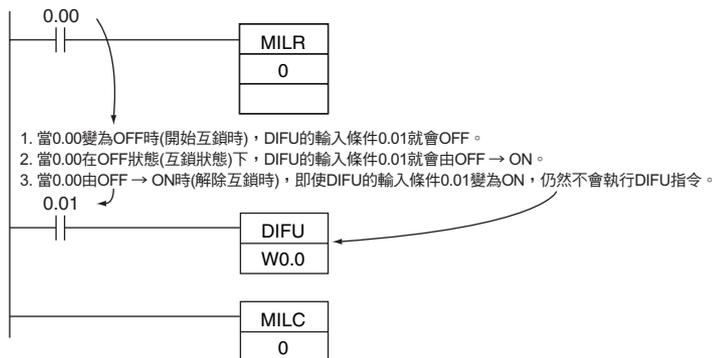
時序圖



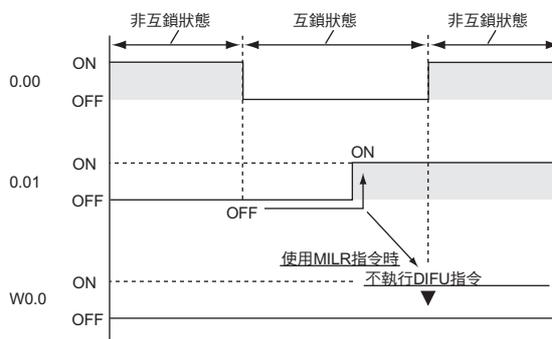
● MILR 指令中出現微分指令時之對應動作

當 MILR-MILC 指令之間存在微分指令 (附加 DIFU/DIFD/@/% 指令) 時, 即使因為輸入條件在開始互鎖時的數值到解除互鎖時的數值範圍內, 使得微分條件成立, 互鎖解除時仍不會執行微分指令。同樣地, 即使微分條件在互鎖開始或解除時的同一個時間點成立時, 也不會執行微分指令。

例：使用上微分 (DIFU) 指令時，只要開始互鎖時的輸入條件 OFF，而且解除互鎖時的輸入條件 ON，當互鎖解除時，將不會執行上微分 (DIFU) 指令。



時序圖

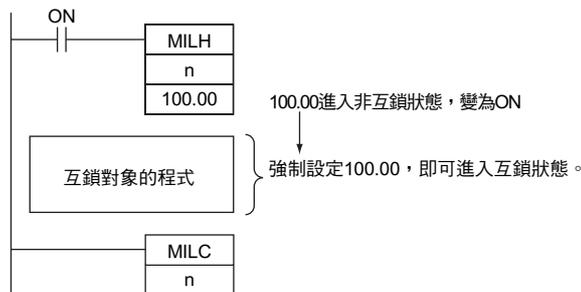
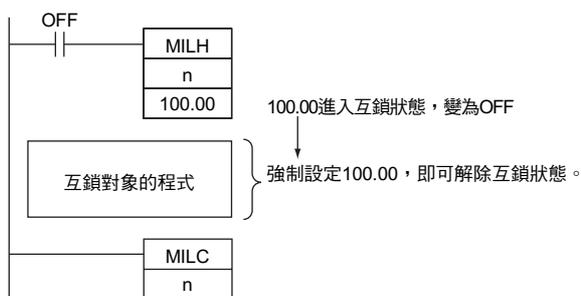


● 使用周邊裝置的強制 ON/OFF 操作來變更互鎖 / 非互鎖狀態

可使用周邊裝置對 MILH 指令第 2 個運算元 D 所指定的暫時記憶繼電器執行強制 ON/OFF 的操作，藉以變更 MILH-MILC 指令間的互鎖或非互鎖狀態。

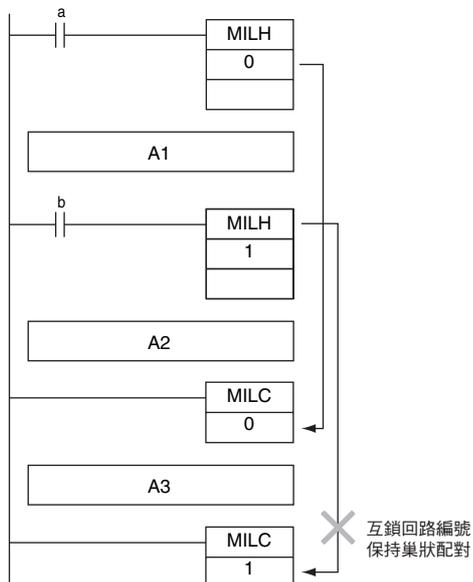
強制設定：進入非互鎖 (IL) 狀態。

強制復歸：進入互鎖 (IL) 狀態。



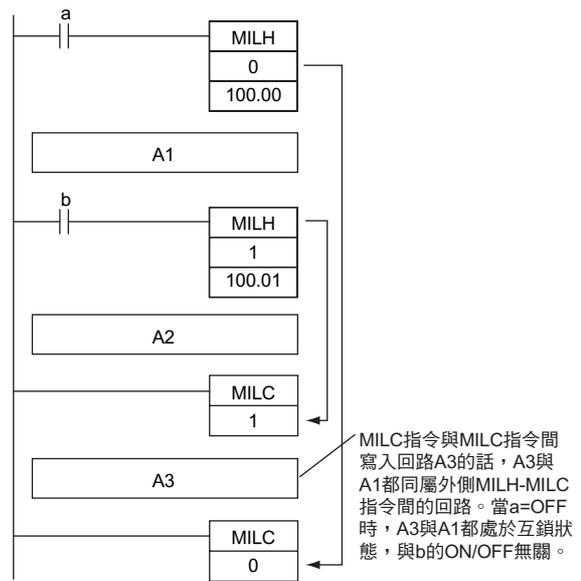
提示

- 儘管 MILH (或 MILR) 指令的輸入條件 OFF 時，MILH (MILR)-MILC 指令間的回路不執行，但是，MILH (MILR)-MILC 指令間的程式仍然是處於被執行的狀態，因此，整體的掃描時間不會因而縮短。
- MILH (MILR)-MILC 指令必須配對使用，有巢狀結構時，MILH (MILR) 指令的編號也必須遵守巢狀結構，不可互相跨越，如下圖所示。

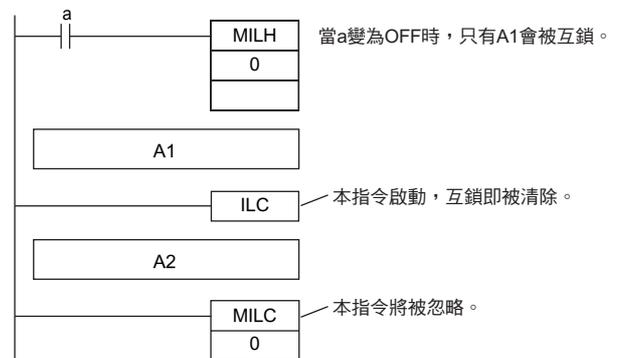


輸入條件		程式		
a	b	A1	A2	A3
OFF	ON	互鎖	互鎖	非互鎖
	OFF			
ON	OFF	非互鎖	互鎖	互鎖
	ON	非互鎖	非互鎖	非互鎖

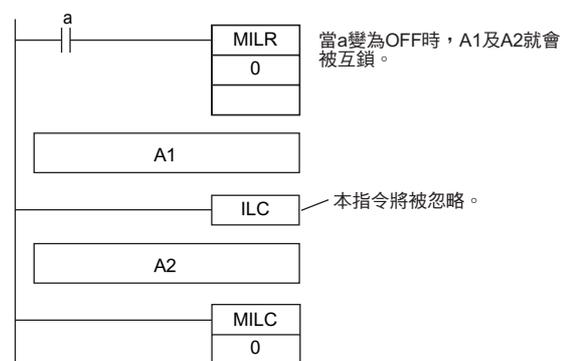
- MILC指令與MILC指令間存在回路的話，該回路照樣可以執行。



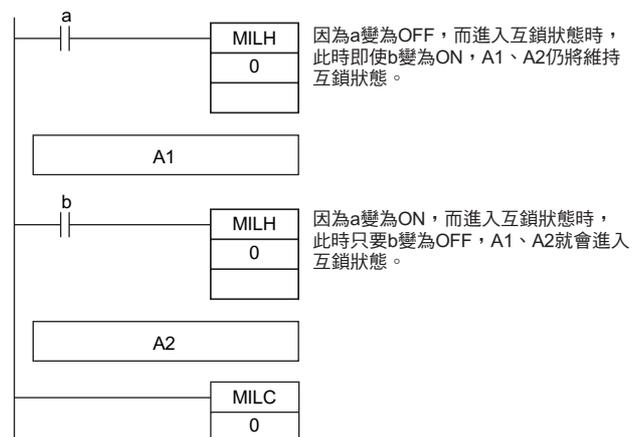
- 當MILH指令－MILC指令之間存在ILC指令時，MILH指令－ILC指令之間的指令區即為無效。



- 當MILR指令－MILC指令之間存在ILC指令時，ILC指令就會被忽略，而且MILR指令－MILC指令之間的指令將成為無效。

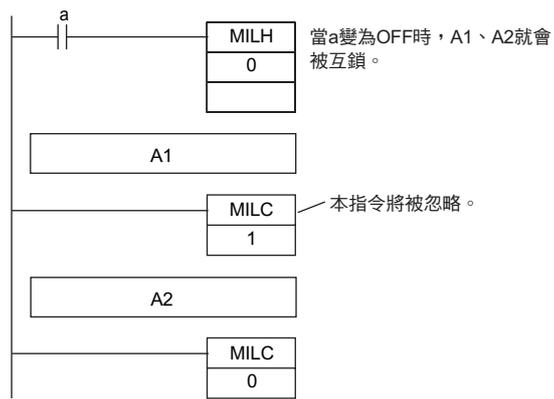


- 當MILH指令－MILC指令之間存在著互鎖編號相同的MILH(或MILR)指令時：
 - 利用外側的MILH指令而進入互鎖狀態時，此時內側的MILH(或MILR)指令將無法啟動互鎖。
 - 利用外側的MILH指令而進入非互鎖狀態時，此時只要利用內側的MILH(或MILR)指令即可啟動互鎖。

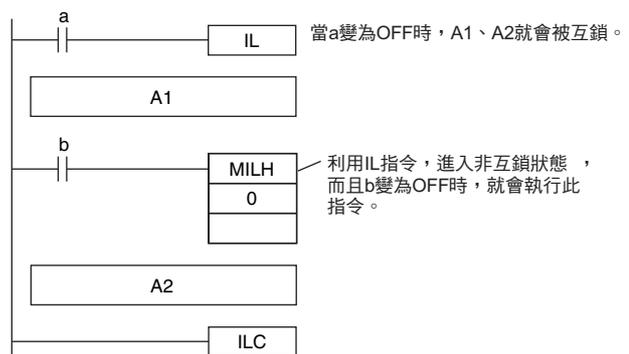


(註) 當MILR指令－MILC指令之間存在著MILH(或MILR)指令時亦同。

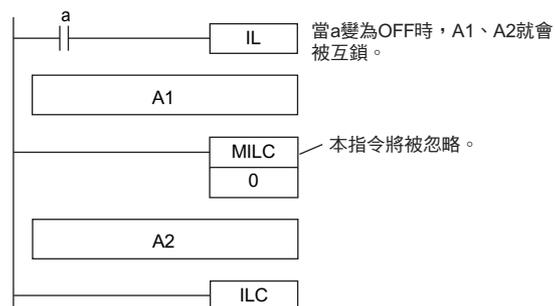
- 當 MILH (或 MILR) 指令 – MILC 指令之間, 存在互鎖編號相異的 MILC 指令時, 互鎖編號不同的 MILC 指令就會被忽略。



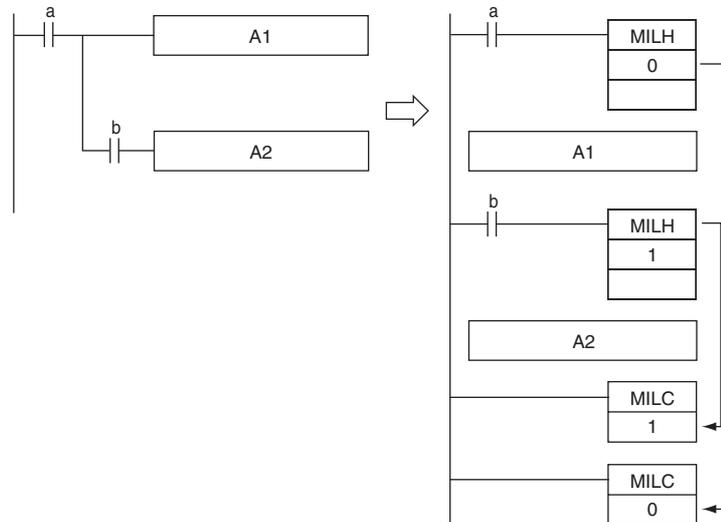
- 當 IL 指令 – ILC 指令之間存在著 MILH 指令時, MILH 指令會自行判斷互鎖狀態是否因 IL 指令而造成, 而且不會在互鎖狀態下執行任何動作。此時, IL-ILC 指令之間的區塊即為互鎖對象。又, 利用 IL 指令進入非互鎖狀態時, 只要 b 變為 OFF, 就會執行 MILH 指令, 而且 MILH-ILC 之間的區塊將進入互鎖狀態。



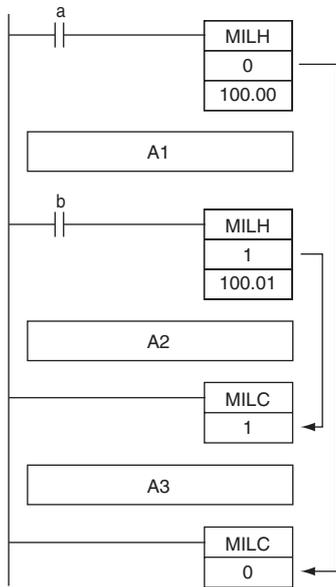
- 當 IL 指令 – ILC 指令之間存在著 MILC 指令時, MILC 指令將會被忽略, 而且 IL-ILC 指令之間的指令區就會變為互鎖對象。



- 利用 MILH (或 MILR) – MILC 指令, 即可在程式內進行高效率的回路切換。如果希望依多項不同的條件來切換處理動作時, 請將 MILH (或 MILR) 指令放在各項處理動作前, MILC 指令放在最後。

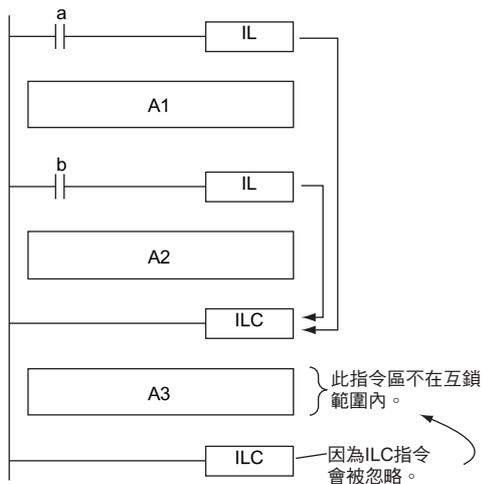


- 不同於 IL-ILC 指令，MILH (或 MILR) – MILC 指令可執行巢狀結構，因此即使階梯程式的敘述相同，仍然會出現下列不同的動作。
- MILH-MILC 指令



輸入條件		程式		
a	b	A1	A2	A3
OFF	ON	互鎖	互鎖	互鎖
	OFF			
ON	OFF	非互鎖	互鎖	非互鎖
ON	ON	非互鎖	非互鎖	非互鎖

- IL-ILC 指令

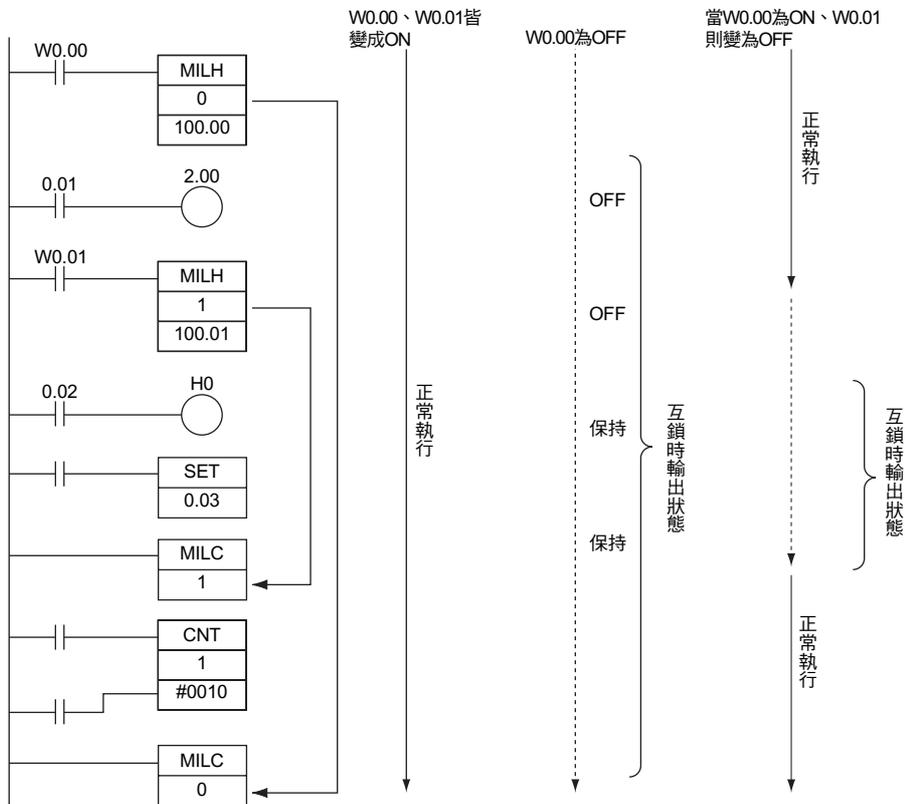


輸入條件		程式		
a	b	A1	A2	A3
OFF	ON	互鎖	互鎖	超出互鎖範圍，因此進入非互鎖狀態
	OFF			
ON	OFF	非互鎖	互鎖	
ON	ON	非互鎖	非互鎖	

- 即使 MILH (或 MILR) 指令正在執行，也能在 MILH (或 MILR) 指令前加上 SET 指令，預先將希望保持 ON 的指令區設定為 ON。

程式例

當 W0.00、W0.01 同時 ON 時，MILH (互鎖編號 0) ~ MILC (互鎖編號 0) 之間的指令將正常執行。
 當 W0.00 為 OFF 時，MILH (互鎖編號 0) ~ MILC (互鎖編號 0) 之間的指令就會被互鎖，
 當 W0.00 變成 ON、W0.01 變成 OFF 時，MILH (互鎖編號 1) ~ MILC (互鎖編號 1) 之間的指令就會被互鎖，除此之外，將正常執行動作。



JMP/CJP/JME

指令名稱	指令記號	指令的各種組合	Fun No.	功能
跳躍	JMP	—	004	當輸入條件 OFF 時，就會直接跳躍至 JME 指令。
條件跳躍 (Jump)	CJP	—	510	當輸入條件 ON 時，就會直接跳躍至 JME 指令。
跳躍結束 (Jump End)	JME	—	005	表示使用 JMP 或 CJP 指令時之跳躍結束位置。

符號	JMP	JME

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	—	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
N	跳躍編號	UINT	1

N：10 進位 & 0 ~ 127 或 16 進位 #0000 ~ 007F

JME 指令只能將 N 指定為常數

● 運算元種類

內容		CH 位址						間接 DM		常數	條件旗標	時鐘脈衝	TR
		CIO	WR	HR	AR	T	C	DM	@DM				
JMP/CJP	N	○	○	○	○	○	○	○	○	○	—	—	—
JME	N	—	—	—	—	—	—	—	—	○	—	—	—

條件旗標的動作

● JMP/CJP

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 N 的資料以 10 進位表示時超過 & 0 ~ 127 或是 16 進位表示時超過 #0000 ~ 07F 的範圍時，就會變成 ON 當所對應的跳躍編號的 JME 指令不存在，本旗標就會 ON 否則為 OFF

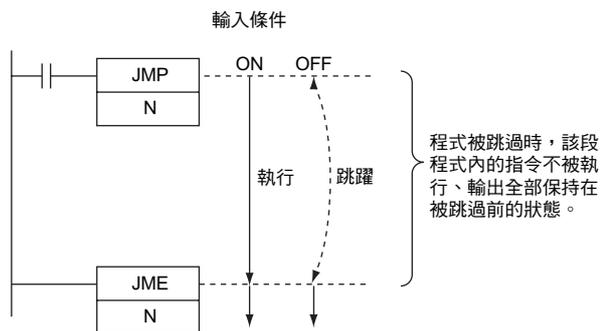
● JME

無

功能

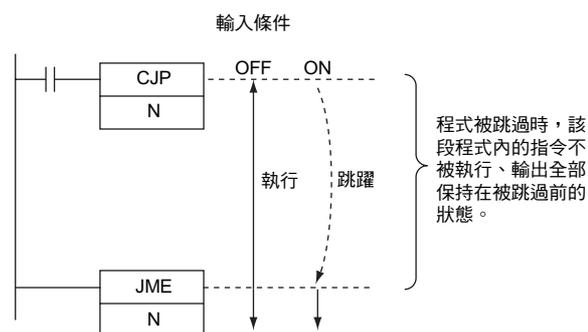
● JMP

只要 JMP 指令的輸入條件 OFF，就會跳躍至附有 N 所指定的跳躍編號的 JME 指令，而且一旦輸入條件 ON，就會執行下一個指令以後的指令。輸入條件 OFF 時，則不會執行 JMP-JME 之間的指令，因此，輸出將保持原來的狀態。不過，進入程式區塊時，就會依不同的輸入條件執行跳躍。



● CJP

當輸入條件 ON 時，就會跳躍至附有 N 跳躍編號的 JME 指令，當輸入條件 OFF 時，則會執行下一個指令以後的指令。



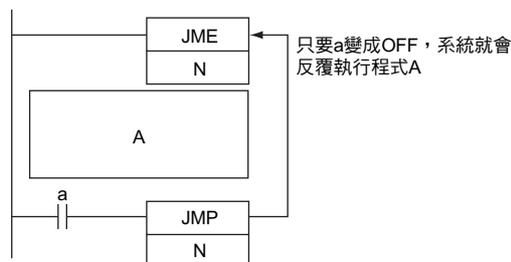
提示

- 使用 JMP/CJP 指令時，只要跳躍條件成立，就會直接跳躍至 JME 指令，此時不會執行 JMP/CJP-JME 之間的指令，因此不會增加這些指令的執行時間，如此便能夠縮短掃描週期。
- 跳躍指令功能比較表

跳躍指令	JMP-JME	CJP-JME
跳躍的輸入條件	OFF	ON
使用數量	共 128 個	
跳躍時是否處理指令	不執行	
跳躍時的執行時間	無	
跳躍時是否輸出指令	保持原先的狀態	
跳躍時，將更新啟動中的計時器現在值	持續更新	
程式區塊內的處理動作	無條件跳躍	ON 時跳躍

使用上的注意事項

- 跳躍時，所有的指令輸出 (繼電器、CH) 將保持原先的狀態。不過，利用 TIM/TIMX 指令 /TIMH/TIMHX 指令 /TMHH/TMHHX 指令來啟動的計時器，即使在執行指令以外的時間，也會進行現在值更新處理，因此會持續計時。
- 當附有相同編號的 JME 指令超過 2 個時，程式位址較小的 JME 指令就會啟動，此時，位址較大的 JME 指令會被忽略。
- CJP 指令會在輸入條件 ON 時跳躍，並且和 JME 指令執行相反的動作。
- 跳躍至程式位址較小的指令時，只要 JMP (CJP) 的輸入條件進入 OFF (ON)，系統就會反覆執行 JMP (CJP)-JME 之間的指令。當 JMP (CJP) 的輸入條件 ON (OFF) 時，就會結束反覆執行動作。又，此時只要 JMP (CJP) 的輸入條件不變成 ON，不需要執行 END 指令，也會出現超出週期時間的情形，此點需特別注意。

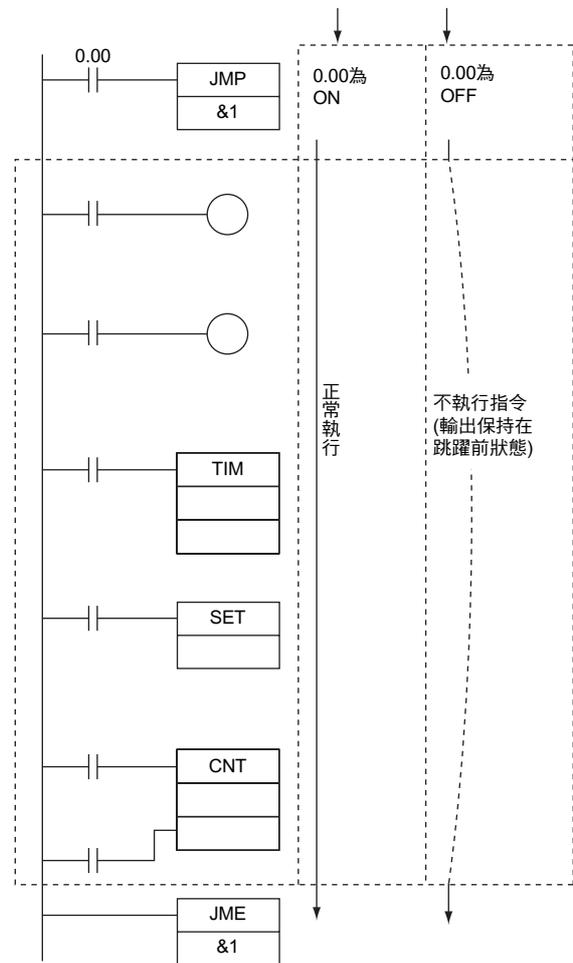


- 在 JMP/CJP-JME 指令區使用微分指令 (附加 DIFU/DIFD/@/% 指令) 時，某些輸入條件將造成動作不穩定，此點請特別注意。

程式例

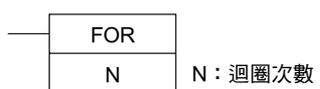
0.00=OFF 時，JMP-JME & 1 之間的指令不被執行，輸出保持在跳躍之前的狀態。

0.00=ON 時，JMP-JME & 1 之間的指令正常的被執行。



FOR/NEXT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
迴圈開始	FOR	—	512	無條件依指定次數反覆執行 FOR ~ NEXT 之間的程式後，就會開始執行 NEXT 指令以後的指令。
迴圈結束	NEXT	—	513	表示 FOR-NEXT 迴圈結束。

符號	FOR	NEXT
		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	—	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
N	迴圈次數	UINT	1

N：迴圈次數

10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
N	○	○	○	○	○	○	○	○	○	○	—	—	—

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 一旦執行時超過巢狀結構的最大數 (15 層)，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	OFF
負旗標	P_N	OFF

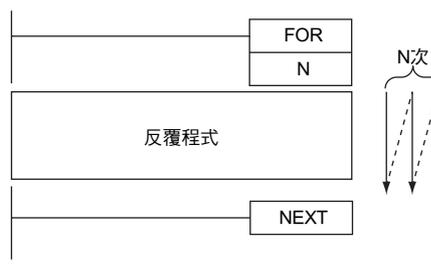
功能

無條件在 N 次反覆執行 FOR ~ NEXT 之間的程式後，就會開始執行 NEXT 指令以後的指令。

欲中斷反覆動作時，請使用 BREAK 指令。

將 N 指令為 0 時，即可讓 FOR ~ NEXT 之間被當成 NOP 指令來執行的指令執行 NOP 處理。

可利用較小的程式容量來處理資料表的資料等。



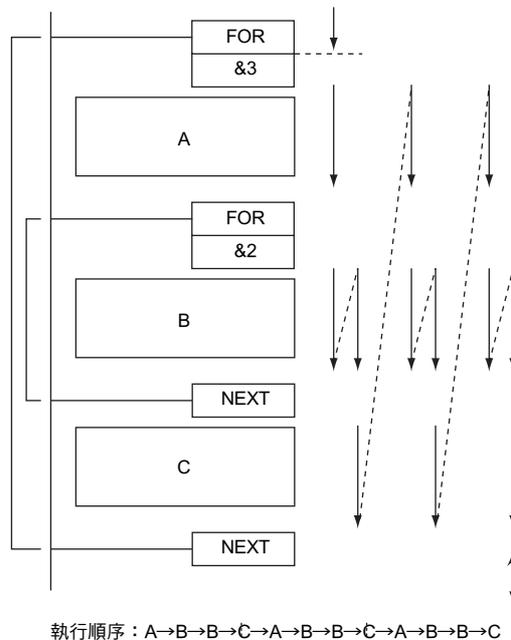
提示

若不希望在無條件狀況下，而是某個輸入條件成立時才反覆程式時，請使用下列 2 種方法。

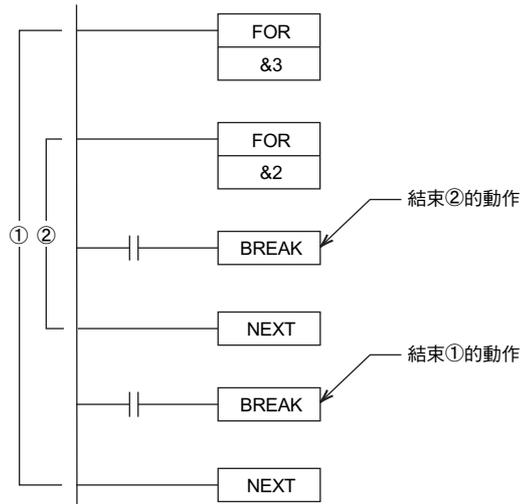
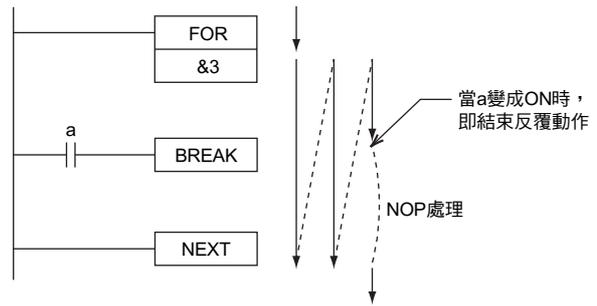
- 1) 使用 FOR/NEXT 指令，將 N 指定為最大反覆次數，並且執行反覆迴圈。只要輸入條件在 N 次執行反覆動作前成立，即可執行 BREAK 指令並且強制結束反覆迴圈。
- 2) 使用跳躍指令 (JME-JMP 指令等)，並且將 JMP 指令配置在 JMP 指令後方。當某項輸入條件成立前，將反覆由 JMP 指令 (程式位址較小者) 跳躍至 JME 指令的動作，不過，系統會在輸入條件成立前，無限制地執行反覆動作，因此有可能發生超出週期時間的情形，此點需特別注意。

使用上的注意事項

- FOR 指令和 NEXT 指令會被敘述在相同的 Task 中，當您所敘述的 Task 相異時，將不會執行反覆動作。
- 請注意，如果將微分指令用在 FOR ~ NEXT 之間的指令區時，該指令只會動作一次，不會依指定的迴圈次數動作。
 - UP 指令、DOWN 指令
 - DIFU 指令、DIFD 指令
 - 上微分型指令 (包含指令的各種組合「@」的指令)
 - 下微分型指令 (包含指令的各種組合「%」的指令)
- FOR ~ NEXT 的巢狀結構 (例如：FORn ~ FORm ~ NEXT ~ NEXT) 最多為 15 層。

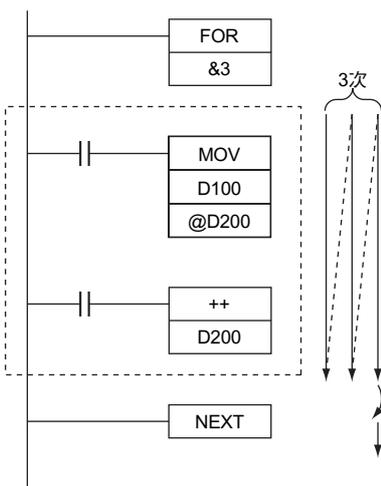


- FOR~NEXT迴圈執行途中欲跳脫時，請使用BREAK指令，自巢狀結構的 FOR~NEXT 迴圈跳脫時，每一層 FOR~NEXT 迴圈請使用一個 BREAK 指令。
- 當BREAK指令被執行時，自BREAK到NEXT指令之間的程式以 NOP 來處理。

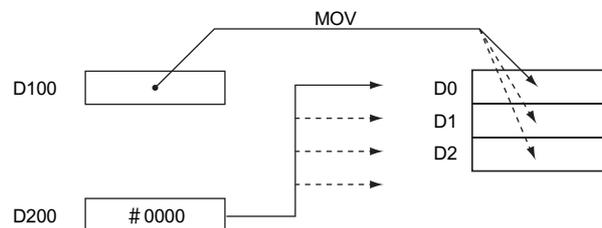


- 在 FOR ~ NEXT 之間執行跳躍指令 (JMP 等) 時，請勿跳躍至 FOR ~ NEXT 以外的指令。
- FOR ~ NEXT 之間不得使用下列指令。
 - 階梯 (SXNT) 指令 /STEP 指令

程式例



將D100的內容傳送至D200的內容所間接指定的位址。
 利用++ (遞增)指令，設定為D200的內容+1，
 並且反覆執行3次。



BREAK

指令名稱	指令記號	指令的各種組合	Fun No.	功能
迴圈跳脫 (Loop break)	BREAK	—	514	可中斷 FOR ~ NEXT 迴圈，並且為 NEXT 前的所有指令執行 NOP 處理。

符號	BREAK

是否適用於特定區域

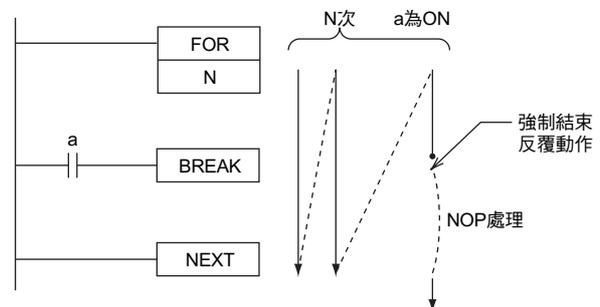
區域	工程步進程式	副程式	中斷任務程式
使用	—	○	○

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	OFF
負旗標	P_N	OFF

功能

配置於 FOR ~ NEXT 指令之間的程式區，輸入條件 ON 時，強制結束 FOR-NEXT 迴圈 (反覆處理)，並且為 NEXT 前的所有指令執行 NOP 處理。



使用上的注意事項

- BREAK 指令僅適用於 1 個巢狀結構，如果要結束多個巢狀結構時，請依巢狀結構數執行 BREAK 指令。
- BREAK 指令僅適用於 FOR ~ NEXT 之間的指令。

計時器 / 計數器指令

計時器 / 計數器的現在值更新方式

● 概要

計時器 / 計數器指令現在值的計時 / 計數方式分成「BCD 方式」及「BIN 方式」兩種。

方式	內容	設定範圍	設定值
BCD 方式	計時器設定值以 BCD 方式來設定。	0 ~ 9,999	#0000 ~ 9999
BIN 方式	計時器設定值以 BIN 方式來設定。	0 ~ 65,535	&& 0 ~ 65535 或 #0000 ~ FFFF

計時器 / 計數器指令的設定值一般為常數外，也可以指定 CH(間接方式) 來取代常數。

設定值指定 CH 時，現在值的計時 / 計數方式也是根據「BCD 方式」或「BIN 方式」的設定來做為計時 / 計數的依據。

● 應用指令

指令分類	指令名稱	指令記號	
		使用 BCD 方式	使用 BIN 方式
計時器 / 計數器指令	100 ms 計時器 (計時器)	TIM	TIMX(550)
	10 ms 計時器 (高速計時器)	TIMH(015)	TIMHX(551)
	1 ms 計時器 (超高速計時器)	TMHH(540)	TMHHX(552)
	累加計時器 (100 ms)	TTIM(087)	TTIMX(555)
	長時間計時器 (100 ms)	TIML(542)	TIMLX(553)
	計數器	CNT	CNTX(546)
	正反計數器	CNTR(012)	CNTRX(548)
	復歸計時器 / 計數器	CNR(545)	CNRX(547)

● 現在值計時 / 計數的設定方法

BIN/BCD 方式可同時存在於同一個專案中，不需要利用 CX-Programmer 的 PLC 內容設定，特別設定更新方式。

● 基本功能一覽表

指令名稱	指令記號	計時	單位	最大設定值	計時器點數 / 指令	計時器編號	時間到旗標的更新時序	計時器現在值更新時序	復歸時	
									時間到旗標	現在值
100ms 計時器	TIM	減法	100ms	999.9 秒	1 點	使用	執行指令時	僅在執行指令時	OFF	設定值
	TIMX			6553.5 秒						
10ms 計時器	TIMH	減法	10ms	99.99 秒	1 點	使用	執行指令時	僅在執行指令時	OFF	設定值
	TIMHX			655.35 秒						
1ms 計時器	TMHH	減法	1ms	9.999 秒	1 點	使用	執行指令時	僅在執行指令時	OFF	設定值
	TMHHX			65.535 秒						
累加計時器	TTIM	累加	100ms	999.9 秒	1 點	使用	執行指令時	僅在執行指令時	OFF	0
	TTIMX			6553.5 秒						
長時間計時器	TIML	減法	100ms	115 日	1 點	不能使用	執行指令時	僅在執行指令時	OFF	設定值
	TIMLX			49710 日						

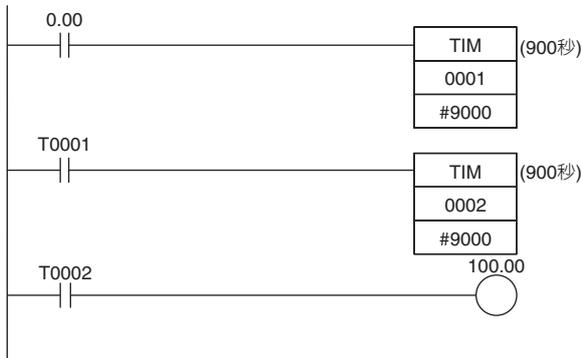
● 各種條件之動作一覽表

指令名稱	指令記號	各種條件					強制 ON 時		強制 OFF 時	
		動作模式變更時	斷電或復歸時	CNR/CNRX 指令執行時	利用 JMP-JME 指令跳躍時	利用 IL-ILC 指令互鎖時	時間到旗標	現在值	時間到旗標	現在值
100ms 計時器	TIM	現在值 = 0 時間到旗標 = OFF	現在值 = 0 時間到旗標 = OFF	現在值 9999 時間到旗標 = OFF	啟動中更新	復歸 (現在值 = 設定值、時間到旗標 = OFF)	ON	0	OFF	設定值
	TIMX			現在值 FFFF 時間到旗標 = OFF						
10ms 計時器	TIMH	現在值 = 0 時間到旗標 = OFF	現在值 = 0 時間到旗標 = OFF	現在值 9999 時間到旗標 = OFF	啟動中更新	復歸 (現在值 = 設定值、時間到旗標 = OFF)	ON	0	OFF	設定值
	TIMHX			現在值 FFFF 時間到旗標 = OFF						
1ms 計時器	TMHH	現在值 = 0 時間到旗標 = OFF	現在值 = 0 時間到旗標 = OFF	現在值 9999 時間到旗標 = OFF	啟動中更新	復歸 (現在值 = 設定值、時間到旗標 = OFF)	ON	0	OFF	設定值
	TMHHX			現在值 FFFF 時間到旗標 = OFF						
累加計時器	TTIM	現在值 = 0 時間到旗標 = OFF	現在值 = 0 時間到旗標 = OFF	現在值 9999 時間到旗標 = OFF	保持	保持	ON	0	OFF	0
	TTIMX			現在值 FFFF 時間到旗標 = OFF						
長時間計時器	TIML	—	—	不能使用	保持	復歸 (現在值 = 設定值、時間到旗標 = OFF)	—	—	—	—
	TIMLX									

● 計時器 / 計數器指令時之程式範例

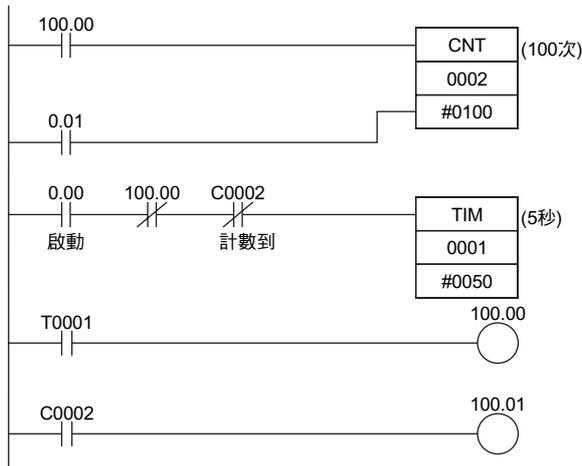
(1) 長時間計時器

① TIM + TIM (例 30 分鐘延遲回路)



指令	資料
LD	0.00
TIM	1
	#9000
LD	T0001
TIM	2
	#9000
LD	T0002
OUT	100.00

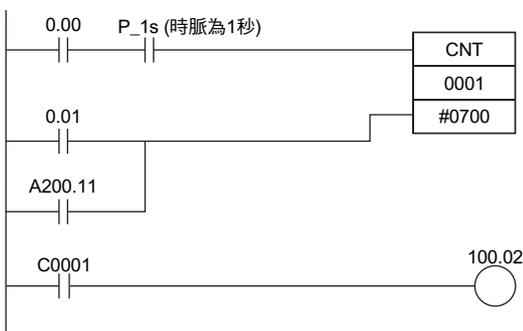
② TIM + CNT (例 500 秒延遲回路)



指令	資料
LD	100.00
LD	0.01
CNT	2
	#0100
LD	0.00
AND NOT	100.00
ANT NOT	C0002
TIM	1
	#0050
LD	T0001
OUT	100.00
LD	C0002
OUT	100.01

- 利用 TIM1 每隔 5 秒產生 1 次脈衝，然後再利用 CNT2 計算每隔 5 秒所產生的脈衝數。
- 計時器時間為(計時器+週期時間) x 計數器數值。例如 500 秒計時器。
- 計數器的現在值即使在電源 OFF 後，仍然保持原來的資料。

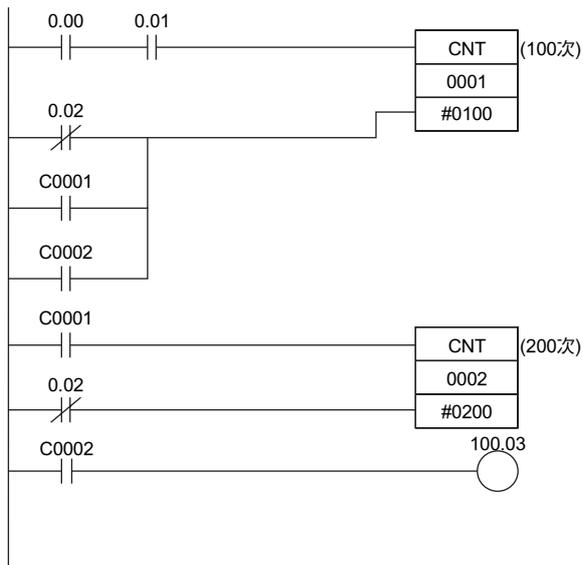
③ 時鐘脈衝 + 計數器 (例如：700 秒延遲回路)



指令	資料
LD	0.00
AND	1s
LD NOT	0.01
CNT	1
	#0700
LD	C0001
OUT	100.02

- 將內部時鐘脈衝與計數器互相搭配，即可建構長時間計時器。
- 將 CNT1 復歸信號中的特殊輔助繼電器 A200.11 (開始運轉時持續 ON 動作 1 個掃描時間) 設定為 OR 時，一旦程式開始執行，就會再次由設定值 (0700) 開始計數。

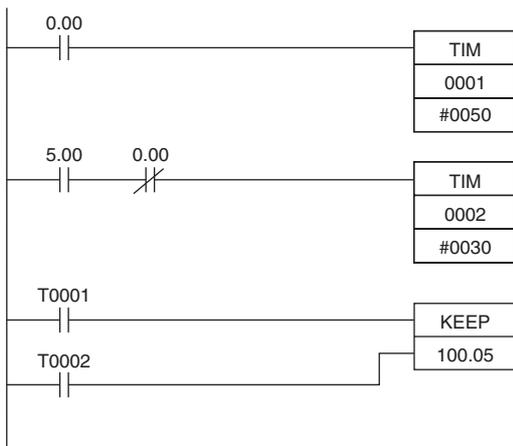
(2) 多位數計數器 (例如 20,000 次)



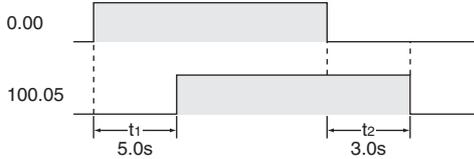
指令	資料
LD	0.00
AND	0.01
LD NOT	0.02
OR	C0001
OR	C0002
CNT	1
	#0100
LD	C0001
LD NOT	0.02
CNT	2
	#0200
LD	C0002
OUT	100.03

- 若希望計數能超過 9999 次以上時，只要
在多段編寫計數器程式即可。

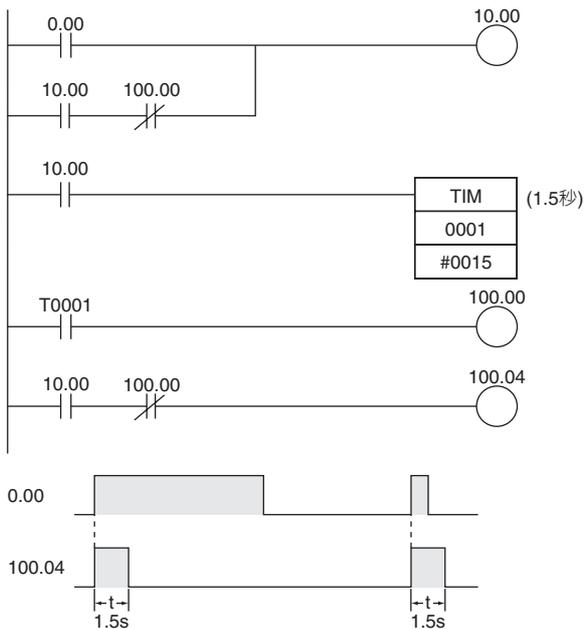
(3) ON/OFF 延遲 (Delay) 回路



指令	資料
LD	0.00
TIM	1
	#0050
LD	5.00
AND NOT	0.00
TIM	2
	#0030
LD	T0001
LD	T0002
KEEP (011)	100.05



(4) 單次觸發回路

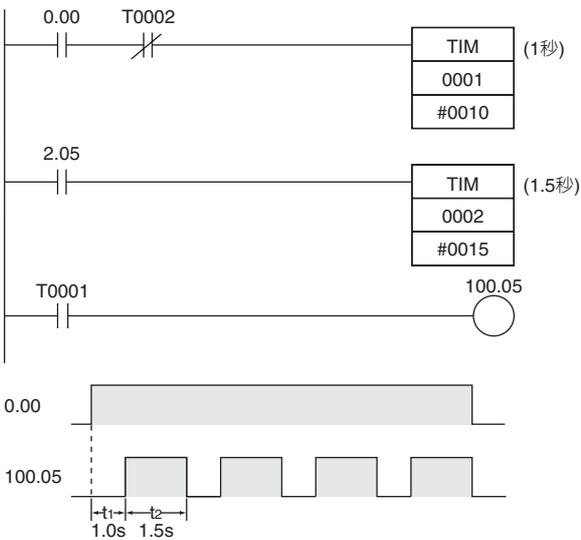


指令	資料
LD	0.00
LD	10.00
AND NOT	100.00
OR LD	—
OUT	10.00
LD	10.00
TIM	1
	#0015
LD	T0001
OUT	100.00
LD	10.00
AND NOT	100.00
OUT	100.04

- 當單次觸發的輸入 ON 時，只會在 TIM1 的設定時間輸出。

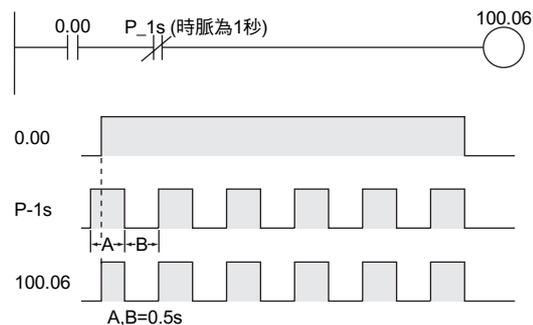
(5) 閃爍回路

① 使用 2 個 TIM 之範例



指令	資料
LD	0.00
AND NOT	T0002
TIM	1
	#0010
LD	2.05
TIM	2
	#0015
LD	T0001
OUT	100.05

② 脈衝使用範例



指令	資料
LD	0.00
AND	1s
OUT	100.06

- 利用內部時鐘脈衝 (0.1 秒、0.2 秒、1 秒)，即可更輕鬆編寫閃爍回路。

● 計時器的復歸方法

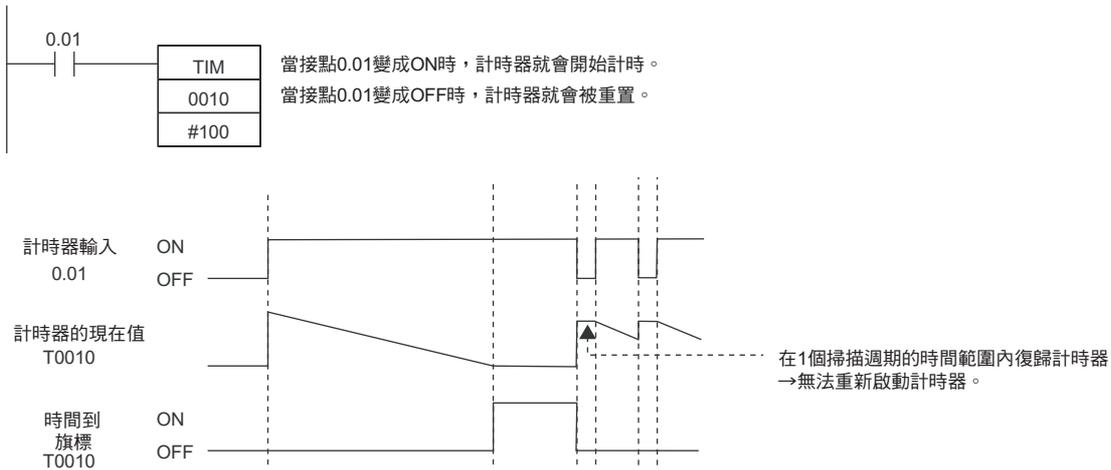
計時器指令的復歸方法有以下 2 種：

1. 將計時器指令的輸入 OFF

將計時器指令的輸入 OFF 後，計時器就會被復歸。

將計時器指令再次 ON 時，計時器就會開始計時。

在 1 掃描週期的時間範圍內復歸計時器→無法重新啟動計時器。

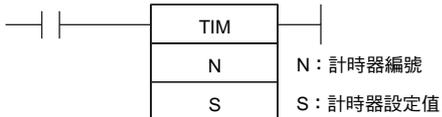
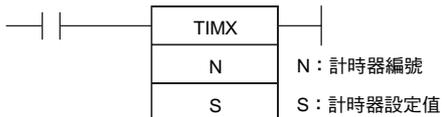


2. 使用計時器 / 計數器復歸指令

執行計時器 / 計數器復歸指令 (CNR/CNRX)，即可復歸您所指定的計時器。

TIM/TIMX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
100 ms 計時器	TIM/TIMX	—	550	0.1 秒為單位，減算型計時器。

符號	TIM	TIMX
	BCD 方式  N：計時器編號 S：計時器設定值	BIN 方式  N：計時器編號 S：計時器設定值

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

運算元種類的說明

運算元種類	內容	資料型態		容量
		TTIM	TTIMX	
N	計時器編號	TIMER	TIMER	1
S	計時器設定值	WORD	UINT	1

N：計時器編號

10 進位 0 ~ 255

S：計時器設定值 (100 ms 單位)

TIM (BCD)：#0000~9999

TIMX (BIN)：10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

● 運算元種類

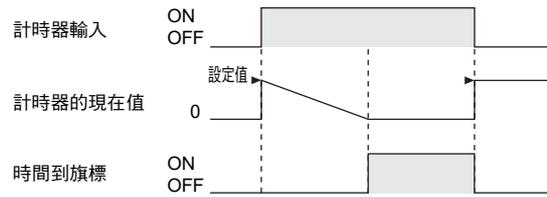
內容	CH 位址					間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	@DM	* DM				
N	—	—	—	—	○	—	—	—	—	—	—
S	○	○	○	○	○	○	○	○	○	—	—

條件旗標的動作

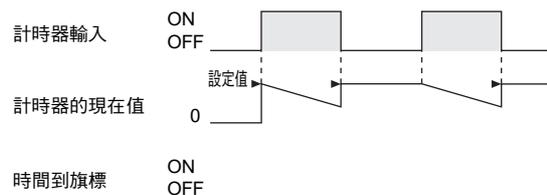
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 使用 BCD 方式且 S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF

功能

- 當計時器輸入 OFF 時，N 所指定編號的計時器就會被復歸 (將設定值 S 代入計時器現在值，並且將時間到旗標設定為 OFF)
- 當計時器輸入由 OFF → ON 時，即可啟動計時器，並且開始計時器現在值的減法計算，在計時器輸入 ON 的期間，將執行計時器現在值的更新，一旦計時器現在值變成 0 時，時間到旗標就會變成 ON (計時器時間截止)。
- 時間到後，將繼續保持計時器現在值及時間到旗標的狀態。如果要重新啟動計時器，請先將計時器輸入由 OFF → ON，或是 (利用 MOV 指令等) 將計時器現在值變更為 0 以外的數值。
- 設定時間如下。
 - 使用 TIM 時 (BCD) : 0 ~ 999.9 秒
 - 使用 TIMX 時 (BIN) : 0 ~ 6553.5 秒
- 計時器的精確度為 -0.01 ~ 0 秒。



(當計時器輸入在計時器時間截止前變成 OFF 時)



提示

- TIM/TIMX 指令所使用的計時器現在值及時間到旗標會在下列時間點進行更新。

更新的時間點	內容
執行各項指令時	<ul style="list-style-type: none"> 每次執行指令時就會更新現在值 當現在值 = 0 時，時間到旗標就會 ON，否則時間到旗標將會 OFF

- 斷電時，計時器會被復歸 (計時器現在值 = 設定值、時間到旗標 OFF)。如果希望斷電時也能保持計時器現在值，請依照下列方法將內部時鐘脈衝與計時器互相搭配。



- 當計時器設定值為 #0000 時，就會在指令執行時發生時間截止的情形。

使用上的注意事項

- 計時器編號可和其他計時器指令共用。利用這一類的指令同時啟動編號相同的計時器時，有可能會造成錯誤動作，此點需特別注意。同時啟動編號相同的計時器時，程式檢查過程中就會出現「線圈重複使用」的訊息，又，只要將條件設定為不同時啟動，即可使用編號相同的計時器。
- 當週期時間超過 4s 時，計時器將無法正常動作，因此，當您使用計時器指令時，週期時間必須小於 4s。
- 一旦出現下列情況，計時器就會被復歸或保持原先狀態。(復歸時、現在值 = 設定值、時間到旗標 = OFF。)

	運轉模式切換時 (Program ↔ Run 或 Monitor) 註 1	斷電後復歸時	執行 CNR/CNRX (計時器 / 計數器復歸指令) (註 2)	利用 IL-ILC 指令互鎖時	利用 JMP-JME 指令跳躍時
現在值	0	0	採用 BCD 方式時 : #9999 採用 BIN 方式時 : #FFFF	設定值	保持先前的狀態
時間到旗標	OFF	OFF	OFF	OFF	保持先前的狀態

(註) 1 當 I/O 記憶體保持旗標 (A500.12) 被設定為 1 (ON) 時，即使動作模式變更仍然會保持先前的狀態。

2 TIM/TIMX 指令執行時，會進行設定值的設定。

- 利用 IL-ILC 指令來執行互鎖動作時，系統將會被復歸，此時現在值 = 設定值，而且時間到旗標 = OFF。
- 利用 JMP/CJP-JME 指令進行跳躍時，目前啟動中的計時器現在值將不會被更新。
- 執行強制設定時，時間到旗標 = ON、現在值 = 0，強制復歸時，時間到旗標 = OFF、現在值 = 設定值。
- 只有在指令執行時才能更新時間到旗標，因此如果要利用使用者程式來讀取時間到旗標時，某些時間點可能會發生相當於 1 個掃描時間的延遲，此點請特別注意。
- 利用線上編輯完成計時器指令的覆寫後，請務必將時間到旗標復歸，否則，將造成覆寫後的計時器無法正常動作。

程式例

當計時器輸入 0.00 由 OFF → ON 時，就會由設定值開始進行計時器現在值的減法運算。

當計時器現在值 = 0 時，時間到旗標 T0000 就會變成 ON。

當計時器輸入 0.00 變成 OFF 時，計時器現在值被復歸成設定值，時間到旗標 T0000 將變成 OFF。



TIMH/TIMHX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
10 ms 計時器 (高速計時器)	TIMH	—	015	0.01 秒為單位，減算型計時器。
	TIMHX	—	551	

符號	TIMH	TIMHX
	BCD 方式	BIN 方式

N：計時器編號
S：計時器設定值

N：計時器編號
S：計時器設定值

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

運算元種類的說明

運算元種類	內容	資料型態		容量
		TIMH	TIMHX	
N	計時器編號	TIMER	TIMER	1
S	計時器設定值	WORD	UINT	1

N：計時器編號

10 進位 0 ~ 255

S：計時器設定值 (100 ms 單位)

TIMH (BCD)：#0000~9999

TIMHX (BIN)：10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

● 運算元種類

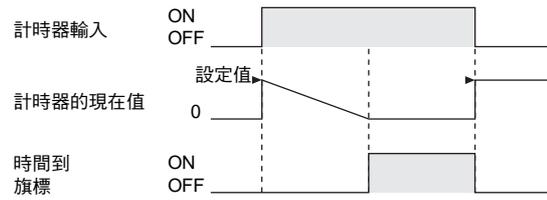
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
N	—	—	—	—	○	—	—	—	—	—	—	—	—
S	○	○	○	○	○	○	○	○	○	○	—	—	—

條件旗標的動作

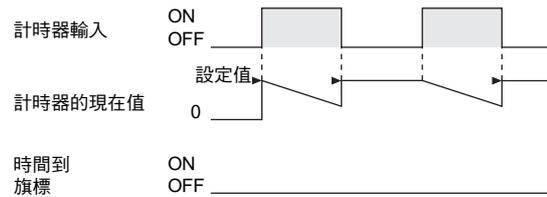
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 利用 BCD 方式且 S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF

功能

- 當計時器輸入 OFF 時，N 所指定編號的計時器就會被復歸 (將設定值 S 代入計時器現在值，並且將計時器時間到旗標設定為 OFF)
- 當計時器輸入由 OFF → ON 時，即可啟動計時器，並且開始計時器現在值的減法計算，在計時器輸入 ON 的期間，將執行計時器現在值的更新，一旦計時器現在值變成 0 時，計時器時間到旗標就會變成 ON (時間到旗標)。
- 計時器時間截止後，將繼續保持計時器現在值及時間到旗標的狀態。如果要重新啟動計時器，請先將計時器輸入由 OFF → ON，或是 (利用 MOV 指令等) 將計時器現在值變更為 0 以外的數值。
- 設定時間如下。
 - 使用 BCD 方式 0 ~ 99.99 秒
 - 使用 BIN 方式 0 ~ 655.35 秒
- 計時器的精確度為 -0.01 ~ 0 秒。



(當計時器輸入在計時器時間截止前變成OFF時)



提示

TIMH/TIMHX 指令所使用的計時器現在值及時間到旗標會在下列時間點進行更新。

更新的時間點	內容
執行各項指令時	<ul style="list-style-type: none"> 每次執行指令時就會更新現在值 當現在值 = 0 時，時間到旗標就會 ON，否則時間到旗標將會 OFF

使用上的注意事項

- 計時器編號可和其他計時器指令共用。利用這一類的指令同時啟動編號相同的計時器時，有可能會造成錯誤動作，此點需特別注意。同時啟動編號相同的計時器時，程式檢查過程中就會出現「線圈重複使用」的訊息，又，只要將條件設定為不同時啟動，即可使用編號相同的計時器。
- 當週期時間超過 4s 時，計時器將無法正常動作，因此，當您使用計時器指令時，週期時間必須小於 4s。
- 一旦出現下列情況，計時器就會被復歸或保持原先狀態。(復歸時，計時器現在值 = 設定值、時間到旗標 = OFF)

	運轉狀態切換時 (Program ↔ Run 或 Monitor) 註 1	斷電後復歸時	執行 CNR/CNRX (計時器 / 計數器復歸指令) (註 2)	利用 IL-ILC 指令互鎖時	利用 JMP-JME 指令跳躍時
現在值	0	0	採用 BCD 方式時：#9999 採用 BIN 方式時：#FFFF	設定值	保持先前的狀態
時間到旗標	OFF	OFF	OFF	OFF	保持先前的狀態

(註) 1 當 I/O 記憶體保持旗標 (A500.12) 被設定為 1 (ON) 時，即使動作模式變更仍然會保持先前的狀態。

2 TIMH/TIMHX 指令執行時，會進行設定值的設定。

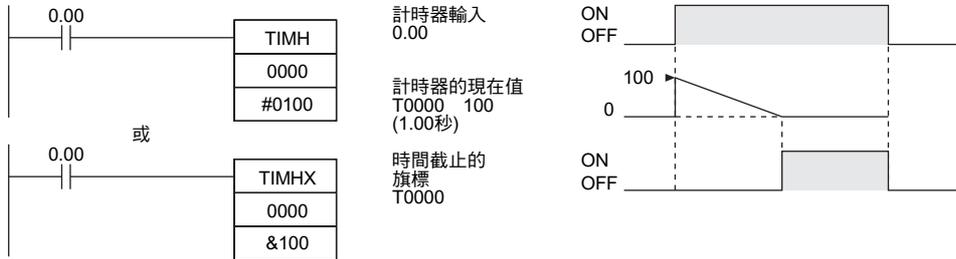
- 利用 JMP/CJP-JME 指令進行跳躍時，目前啟動中的計時器現在值將不會被更新。
- 利用 IL-ILC 指令來執行互鎖動作時，系統將會被復歸，此時現在值 = 設定值，而且時間到旗標 = OFF。
- 執行強制設定時，時間到旗標 = ON、現在值 = 0，強制復歸時，時間到旗標 = OFF、現在值 = 設定值。
- 只有在指令執行時才能更新時間到旗標，因此如果要利用使用者程式來讀取時間到旗標時，某些時間點可能會發生相當於 1 個掃描時間的延遲，此點請特別注意。
- 利用線上編輯完成計時器指令的覆寫後，請務必將時間到旗標復歸，否則，將造成覆寫後的計時器無法正常動作。

程式例

當計時器輸入 0.00 由 OFF → ON 時，就會由設定值開始進行計時器現在值的減法運算。

當 1 秒後的計時器現在值 = 0 時，時間到旗標 T0000 就會變成 ON。

當計時器輸入 0.00 變成 OFF 時，計時器現在值被復歸成設定值，時間到旗標 T0000 將變成 OFF。



TMHH/TMHHX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
1 ms 計時器 (超高速計時器)	TMHH	—	540	0.01 秒為單位，減算型計時器。
	TMHHX	—	552	

符號	TMHH	TMHHX
	BCD 方式	

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

運算元種類的說明

運算元種類	內容	資料型態		容量
		TMHH	TMHHX	
N	計時器編號	TIMER	TIMER	1
S	計時器設定值	WORD	UINT	1

N：計時器編號

10 進位 0 ~ 15

S：計時器設定值

TMHH (BCD)：#0000~9999

TMHHX (BIN)：10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
N	—	—	—	—	○	—	—	—	—	—	—	—	—
S	○	○	○	○	○	○	○	○	○	○	—	—	—

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 利用 BCD 方式且 S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF

功能

- 當計時器輸入 OFF 時，N 所指定編號的計時器就會被復歸 (將設定值 S 代入計時器現在值，並且將計時器時間到旗標設定為 OFF)
- 當計時器輸入由 OFF → ON 時，即可啟動計時器，並且開始計時器現在值的減法計算，在計時器輸入 ON 的期間，將執行計時器現在值的更新，一旦計時器現在值變成 0 時，計時器時間到旗標就會變成 ON (計時器時間截止)。
- 計時器時間截止後，將繼續保持計時器現在值及時間到旗標的狀態。如果要重新啟動計時器，請先將計時器輸入由 OFF → ON，或是 (利用 MOV 指令等) 將計時器現在值變更為 0 以外的數值。
- 設定時間如下。
 - 使用 BCD 方式 0 ~ 9.999 秒
 - 使用 BIN 方式 0 ~ 65.535 秒
- 計時器的精確度為 -0.001 ~ 0 秒。

提示

TMHH/TMHHX 指令所使用的計時器現在值及時間到旗標會在下列時間點進行更新。

更新的時間點	內容
執行各項指令時	<ul style="list-style-type: none"> 每次執行指令時就會更新現在值 當現在值 = 0 時，時間到旗標就會 ON，否則時間到旗標將會 OFF

使用上的注意事項

- 計時器編號可和其他計時器指令共用。利用這一類的指令同時啟動編號相同的計時器時，有可能會造成錯誤動作，此點需特別注意。同時啟動編號相同的計時器時，程式檢查過程中就會出現「線圈重複使用」的訊息，又，只要將條件設定為不同時啟動，即可使用編號相同的計時器。
- 只有在指令執行時才能更新時間到旗標，因此如果要利用使用者程式來讀取時間到旗標時，某些時間點可能會發生相當於 1 週期的延遲，此點請特別注意。請特別注意。
- TMHH/TMHHX 指令無法在待機狀態下更新任務的現在值。
- 一旦出現下列情況，計時器就會被復歸或保持原先狀態。(復歸時，計時器現在值 = 設定值、時間到旗標 = OFF)

	運轉模式切換時 (Program ↔ Run 或 Monitor) 註 1	斷電後復歸時	執行 CNR/CNRX (計時器 / 計數器復歸指令) (註 2)	利用 IL-ILC 指令互 鎖時	利用 JMP-JME 指 令跳躍時
現在值	0	0	採用 BCD 方式時：#9999 採用 BIN 方式時：#FFFF	設定值	保持先前的狀態
時間到旗標	OFF	OFF	OFF	OFF	保持先前的狀態

(註) 1 當 I/O 記憶體保持旗標 (A500.12) 被設定為 1 (ON) 時，即使動作模式變更仍然會保持先前的狀態。

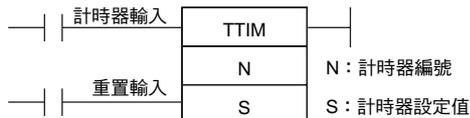
2 TMHH/TMHHX 指令執行時，會進行設定值的設定。

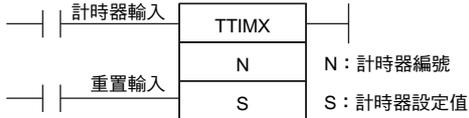
- 利用 JMP/CJP-JME 指令進行跳躍時，目前啟動中的計時器現在值將不會被更新。
- 利用 IL-ILC 指令來執行互鎖動作時，系統將會被復歸，此時現在值 = 設定值，而且時間到旗標 = OFF。
- 執行強制設定時，時間到旗標 = ON、現在 = 0，強制復歸時，時間到旗標 = OFF、現在值 = 設定值。
- 「ON-LINE 程式編輯」的狀態下欲變更計時器指令時 (TIM 指令 ↔ TIMH 指令 ↔ TMHH 指令)，請先強制該時間到旗標 OFF，否則，變更後的計時器無法正常動作。

TTIM/TTIMX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
累加計時器	TTIM	—	087	0.01 秒為單位，減算型計時器。
	TTIMX	—	555	

符號	TTIM	TTIMX
	採用 BCD 方式時	採用 BIN 方式時





是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

運算元種類的說明

運算元種類	內容	資料型態		容量
		TTIM	TTIMX	
N	計時器編號	TIMER	TIMER	1
S	計時器設定值	WORD	UINT	1

N：計時器編號

10 進位 0 ~ 255

S：計時器設定值

TTIM (BCD)：#0000~9999

TTIMX(BIN)：10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

● 運算元種類

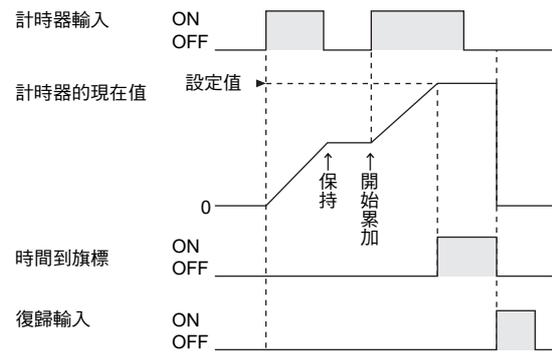
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
N	—	—	—	—	○	—	—	—	—	—	—	—	—
S	○	○	○	○		○	○	○	○	○			

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 使用 BCD 方式且 S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF

功能

- 當計時器輸入 ON 時，就會開始以加法 (累加) 方式來計算現在值；當計時器輸入 OFF 時，就會停止累加，並且保持現在值，當計時器輸入再次 ON 時，將開始累加，一旦計時器現在值到達設定值時，時間到旗標就會變成 ON。
- 計時器時間截止後，將繼續保持計時器現在值及時間到旗標的狀態。如果要重新啟動系統 (利用 MOV 指令等) 時，計時器現在值必須小於設定值，或者是利用復歸信號 ON 或 CNR/CNRX 指令來復歸計時器。
- 設定時間如下。
 - 使用 BCD 方式 0 ~ 999.9 秒
 - 使用 BIN 方式 0 ~ 6553.5 秒
- 計時器的精確度為 -0.01 ~ 0 秒。



提示

- 一般的 TIM/TIMX 指令係利用減法計數的方式來執行動作，因此會顯示截止計算時間結束前的剩餘時間，相對地，TTIM/TTIMX (累加計時器) 指令的經過時間為現在值，因此會直接使用現在值來演算，或是輸出顯示值。

使用上的注意事項

- 計時器編號可和其他計時器指令共用。利用這一類的指令同時啟動編號相同的計時器時，有可能會造成錯誤動作，此點需特別注意。同時啟動編號相同的計時器時，程式檢查過程中就會出現「線圈重複使用」的訊息，又，只要將條件設定為不同時啟動，即可使用編號相同的計時器。
- 一旦出現下列情況，計時器就會被復歸或保持原先狀態。(時、現在 = 0、= OFF。)

	運轉模式切換時 (Program ↔ Run 或 Monitor) 註 1	斷電後復歸時	執行 CNR/CNRX (計時器 / 計數器復歸指令) (註 2)	利用 IL-ILC 指令互鎖時	利用 JMP-JME 指令跳躍時
現在值	0	0	採用 BCD 方式時：#9999 採用 BIN 方式時：#FFFF	保持先前的狀態	保持先前的狀態
時間到旗標	OFF	OFF	OFF	保持先前的狀態	保持先前的狀態

(註) 1 當 I/O 記憶體保持旗標 (A500.12) 被設定為 1 (ON) 時，即使動作模式變更仍然會保持先前的狀態。

2 TIM/TIMX 指令執行時，會進行設定值的設定。

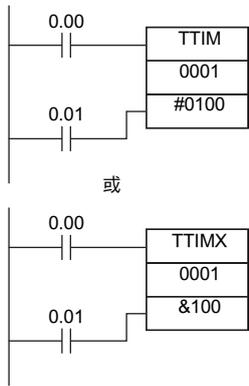
- 利用 IL-ILC 指令來執行互鎖動作時，系統將保持現在值，並且被復歸，請注意，此時有可能會因此造成誤差發生。
- 利用 JMP-JME 指令來跳躍時，系統將保持現在值。請注意，此時有可能會因此造成誤差發生。
- 執行強制設定時，時間到旗標 = ON、現在值 = 0，強制復歸時，時間到旗標 = OFF、現在值 = 0。又，強制設定與強制復歸輸入優先於計時器復歸輸入。
- 只有執行指令時才能累加現在值，因此當週期時間超過 100 ms 以上時 (因為累加單位為 100 ms)，系統將無法正常動作，此點需特別注意。
- 只有在指令執行時才能更新時間到旗標，因此如果要利用使用者程式來讀取時間到旗標時，某些時間點可能會發生相當於 1 個掃描週期的延遲，此點請特別注意。

程式例

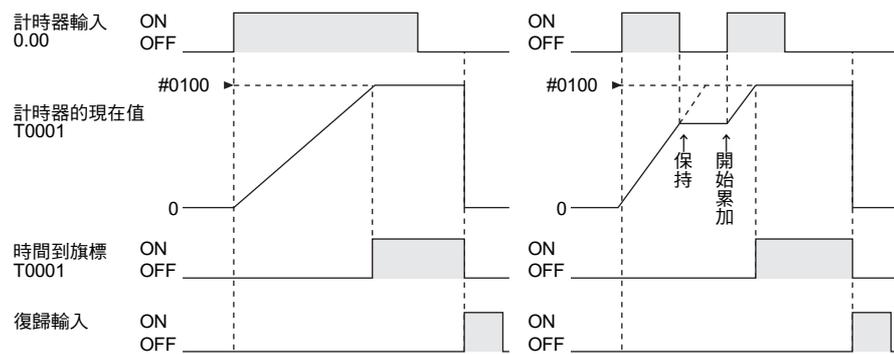
當計時器輸入 0.00 變成 ON 時，就會開始計時器現在值的累加計數，而且當計時器現在值 = 計時器設定值時，時間到旗標 T0001 將變成 ON。

當復歸信號變成 ON 時，計數器現在值將變為 0、時間到旗標 T0001 變為 OFF (通常使用時會將復歸信號設定為 ON 後，再將計時器輸入 ON)。

如果計時器輸入在到達設定值前變成 OFF，就會停止累加，並且保持計時器現在值，當計時器輸入再次 ON 時，才會開始累加動作。

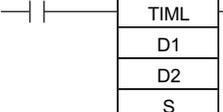
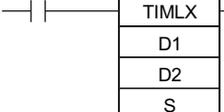


一般的動作



TIML/TIMLX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
長時間計時器	TIML	—	542	0.01 秒為單位，減算式延遲計時器。
	TIMLX	—	553	

符號	TIML	TIMLX
	採用 BCD 方式時  D1：時間到旗標的CH編號 D2：現在值輸出下位CH編號 S：計時器設定值下位CH編號	採用 BIN 方式時  D1：時間到旗標的CH編號 D2：現在值輸出下位CH編號 S：計時器設定值下位CH編號

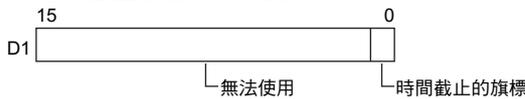
是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

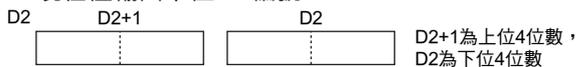
運算元種類的說明

運算元種類	內容	資料型態		容量
		TIML	TIMLX	
D1	時間到旗標的 CH 編號	WORD	UINT	1
D2	現在值輸出下位 CH 編號	DWORD	UDINT	2
S	計時器設定值下位 CH 編號	DWORD	UDINT	2

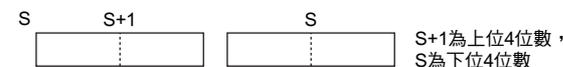
D1：時間截止旗標的CH編號



D2：現在值輸出下位CH編號



S：計時器設定值下位CH編號



D2、S 的範圍

- 採用 BCD 方式時：
BCD#00000000 ~ 99999999
 - 採用 BIN 方式時：
10 進位 & 00000000 ~ 4294967294
或 16 進位 #00000000 ~ FFFFFFFF
- (註) D2+1、D2、S+1 和 S 必須屬於相同的運算元種類。

● 運算元種類

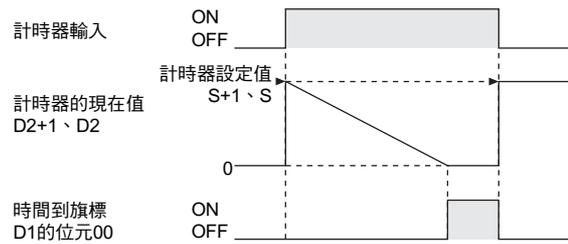
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D1,D2	○	○	○	○	—	—	○	○	○	—	—	—	
S	○	○	○	○	○	○	○	○	○	○	—	—	

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 使用 BCD 方式且 D2 的資料非 BCD 時，本旗標就會 ON 使用 BCD 方式且 S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF

功能

- 當計時器輸入 OFF 時，就會將計時器復歸 (將設定值 S+1、S 代入計時器現在值 D2+1、D2，並且將時間到旗標設定為 OFF)。
- 當計時器輸入由 OFF → ON 時，即可啟動計時器，並且開始計時器現在值 D2+1、D2 的減法計算。在計時器輸入 ON 的期間，將執行計時器現在值的更新，一旦計時器現在值變成 0 時，計時器時間到旗標就會變成 ON (計時器時間截止)。
- 計時器時間截止後，將繼續保持計時器現在值及時間到旗標的狀態。如果要重新啟動計時器，請先將計時器輸入由 OFF → ON，或是 (利用 MOV 指令等) 將計時器現在值 D2+1、D2 變更為 0 以外的數值。
- 計時器的精確度為 -0.01 ~ 0 秒。
- 時間設定的最大值 (以秒為單位) 如下：
 - 使用 BCD 方式 115 日
 - 使用 BIN 方式 4971 日

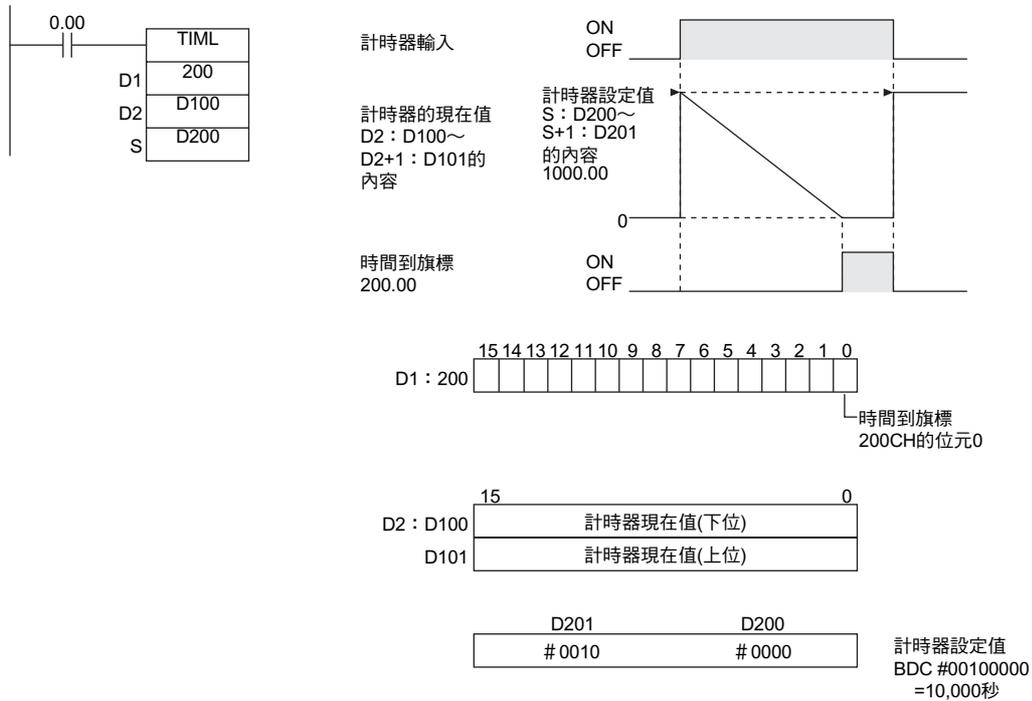


使用上的注意事項

- 不必指定計時器編號。
- 時間到旗標可被強制 ON 或強制 OFF，但是，計時現在值沒有變化。
- 只有執行指令時才能更新現在值，因此當週期時間超過 100 ms 以上時 (因為更新單位為 100 ms)，系統將無法正常動作，此點需特別注意。
- 在指令執行時才能更新時間到旗標，因此如果要利用使用者程式來讀取時間到旗標時，某些時間點可能會發生相當於 1 個掃描週期的延遲，此點請特別注意。
- 利用 IL-ILC 指令來執行互鎖動作時，系統將會被復歸，此時現在值 = 設定值，而且時間到旗標 = OFF。
- 利用 JMP-JME 指令來跳躍時，系統將保持現在值。請注意，此時有可能會因此造成誤差發生。
- 利用長時間計時器指令來使用 D1、D2、D2+1 時，將無法同時執行其他的指令，否則將無法保證時間截止功能正常動作，此點需特別注意。

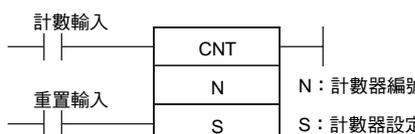
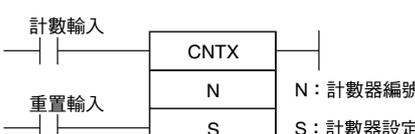
程式例

當計時器輸入 0.00 變成 ON，計時器現在值 (D101、D100) 就會變為計時器設定值 (D201、D200)，並且開始執行減法計算。當計時器現在值 = 0 時，時間到旗標 200.00 就會變成 ON。當計時器輸入 0.00 變成 OFF，時間到旗標 200.00 將變為 OFF。



CNT/CNTX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
計數器	CNT/CNTX	—	546	執行減法計數器的動作。

符號	CNT	CNTX
	採用 BCD 方式時 	採用 BIN 方式時 

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量
		CNT	CNTX	
N	カウンタ番号	COUNTER	COUNTER	1
S	カウンタ設定値	WORD	UINT	1

N：計數器編號

10 進位 0 ~ 255

S：計數器設定

CNT (BCD)：#0000 ~ 9999

CNTX (BIN)：10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

● 運算元種類

內容	CH 位址						間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM				
N	—	—	—	—	—	○	—	—	—	—	—	—
S	○	○	○	○	○	○	○	○	○	○	—	—

條件旗標的動作

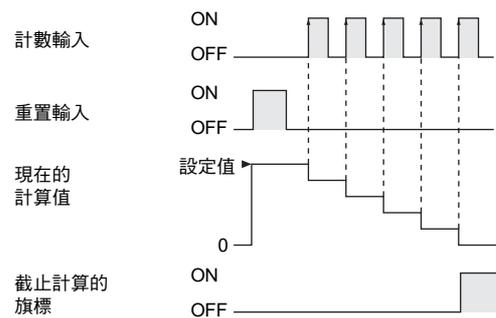
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 使用 BCD 方式且 S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF

功能

每當計數器輸入訊號上升時，就會以減法計算計數器的現在值，當計數器現在值 = 0 時，截止計算 (Count up) 旗標就會 ON。截止計算完成後，如果不將復歸信號 ON 或是利用 CNR/CNRX 指令來復歸計數器，將無法重新啟動系統。當復歸信號 ON 時，系統就會被復歸 (現在值 = 設定值、截止計算旗標 = OFF)，此時計數器輸入將被關閉。

設定值如下。

- 使用 BCD 方式 0~9999 次
- 使用 BIN 方式 0~65535 次



提示

- 斷電時仍然會保持計數器現在值，因此，如果希望現在值可以在電源 ON 時由設定值開始計數 (不希望由斷電前夕的計數器現在值開始計數) 時，請在計數器復歸信號中輸入 1 週期 ON 旗標 A200.11 (特殊輔助繼電器: 只會在開始運轉時的 1 週期 ON) (註) 1 利用電容器來備份 CP1E CPU 模組時，只要電源 OFF 時間超過下列時間，計數器現在值、截止計算旗標就會出現不穩定的情形。

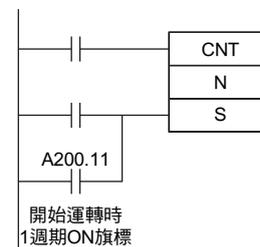
E 型 CPU 模組

9 小時 (60°C)、50 小時 (25°C)

N/NA 型 CPU 模組

7 小時 (60°C)、40 小時 (25°C)

- 2 利用 PLC 系統設定為「保持記憶體歸零」後，每次只要電源 ON 時，計時器現在值和截止計算時間到旗標就會被清除，此時，資料記憶體 (D)、保持記憶體 (H) 也會同時被清除。
- 3 CP1E CPU 模組 N/NA 型 (CP1E-N/NA□D□-□型) 可安裝電池，安裝電池後，即可在電源 OFF 時，保持計數器現在值、截止計算旗標。



使用上的注意事項

- 利用計數器指令、可逆計數器指令即可共用計數器編號。利用這一類的指令同時啟動編號相同的計數器時，有可能會造成錯誤動作，此點需特別注意。同時啟動編號相同的計時器時，程式檢查過程中就會出現「線圈重複使用」的訊息，又，只要將條件設定為不同時啟動，即可使用編號相同的計時器。

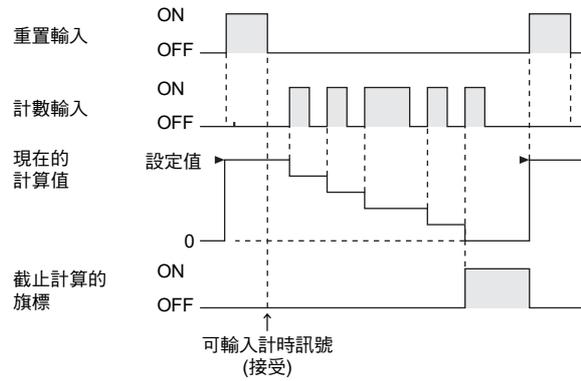
計數器現在值、截止計算旗標的更新時間

	更新的時間點
現在的計算值	當計數器輸入訊號上升時
截止計算的旗標	每次執行指令時 (當現在值為 0 時 ON，否則就會 OFF)

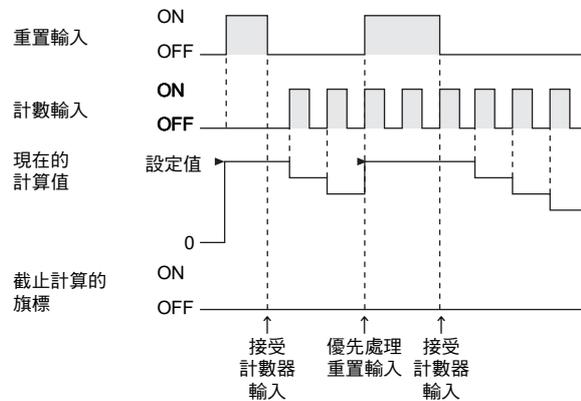
強制設定 / 復歸時的計數器現在值、截止計算旗標的狀態

	強制設定時	強制復歸時
現在的計算值	0	設定值
截止計算的旗標	ON	OFF

- 請依下圖所示，在計數器輸入由 OFF → ON 開始前，請避避先將復歸信號由 OFF → ON 後再進行復歸。而且請在數器輸入由 OFF → ON 前，將復歸信號由 ON → OFF。當復歸信號 ON 時，將無法接受計數器輸入訊號。



- 當復歸信號和計數器輸入同時 ON 時，將優先處理復歸信號，此時計數器就會被復歸 (計數器現在值 = 設定值，截止時間旗標 = OFF)。



- 利用線上編輯來新增計時器時，必須先將計時器復歸然後再使用復歸功能。當計數器還保留現在值之前的數值時，將造成程式變更後的計數器無法正常動作。

CNTR/CNTRX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
可逆計數器	CNTR	—	012	執行加減法計數器的動作。利用加法計數器輸入訊號上升執行加法運算，再利用減法計數器輸入訊號上升，執行減法運算。
	CNTRX	—	548	

符號	CNTR			CNTRX		
	BCD 方式			BIN 方式		
	—	加法計數器	CNTR	—	加法計數器	CNTRX
	—	減法計數器		—	減法計數器	
	—	重置輸入		—	重置輸入	
			N : 計數器編號			N : 計數器編號
			S : 計數器設定			S : 計數器設定

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量
		CNTR	CNTRX	
N	計數器編號	COUNTER	COUNTER	1
S	計數器設定	WORD	UINT	1

N : 計數器編號

10 進位 0 ~ 255

S : 計數器設定

CNTR (BCD) : #0000 ~ 9999

CNTRX (BIN) : 10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

● 運算元種類

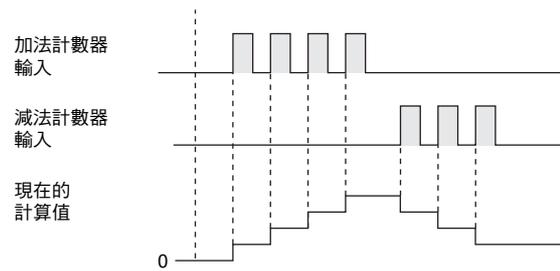
內容	CH 位址						間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM				
N	—	—	—	—	—	○	—	—	—	—	—	—
S	○	○	○	○	○	○	○	○	○	○	—	—

條件旗標的動作

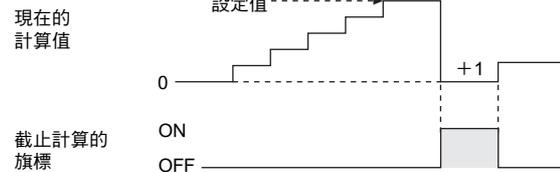
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 使用 BCD 方式且 S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF

功能

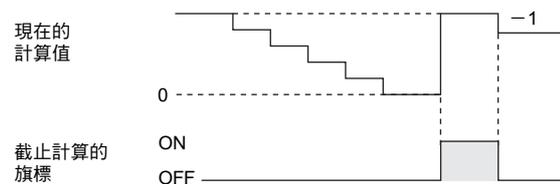
利用加法計數器輸入訊號上升執行加法運算，再利用減法計數器輸入訊號上升，執行減法運算。利用加法運算，當現在值由設定值開始進位為 0 時，截止計數旗標將變成 ON，由 0 加到 1 時則變為 OFF，利用減法運算，當現在值由 0 開始退位時，本旗標將變成 ON，由設定值開始減去 1 時則變為 OFF。



截止計算旗標的動作
・利用加法運算時



・利用減法運算時

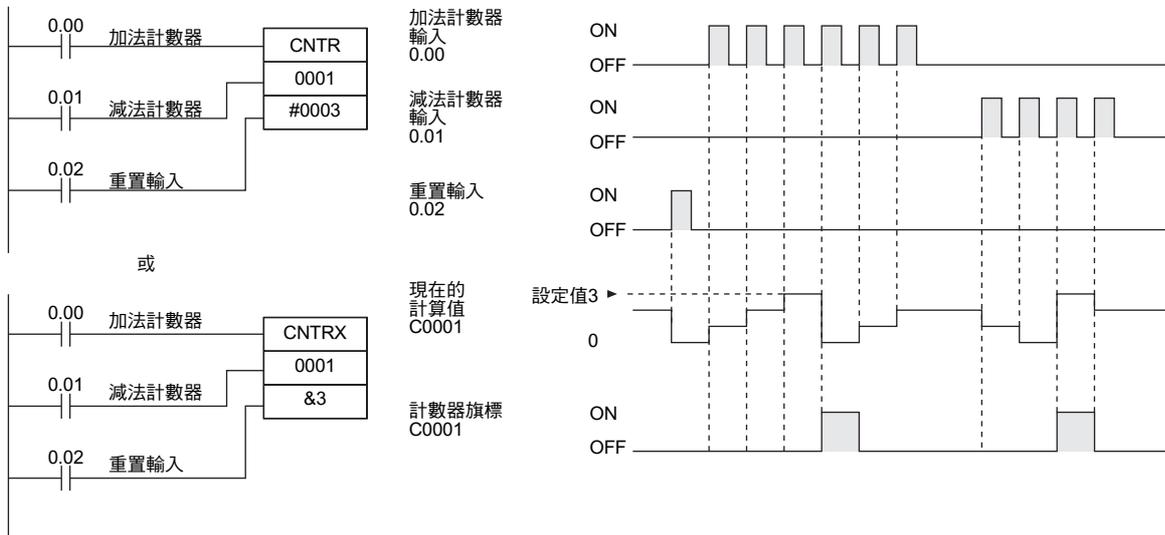


使用上的注意事項

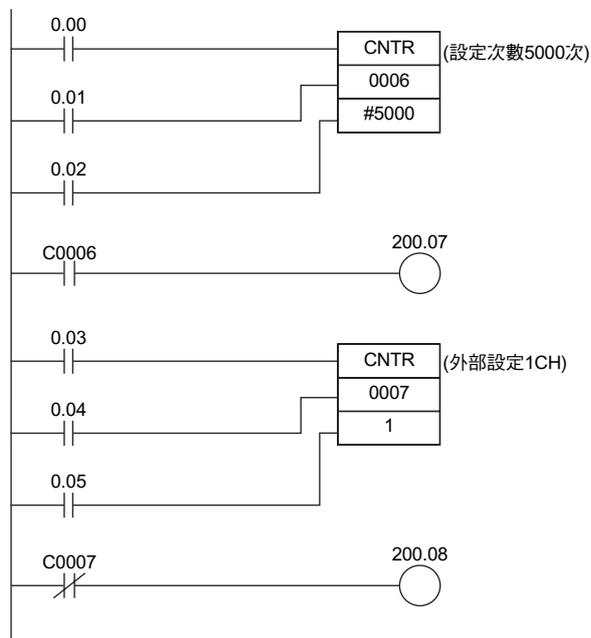
- 利用計數器指令、可逆計數器指令、計數器等待指令，即可共用計數器編號。利用這一類的指令同時啟動編號相同的計數器時，有可能會造成錯誤動作，此點需特別注意。同時啟動編號相同的計時器時，程式檢查過程中就會出現「線圈重複使用」的訊息，又，只要將條件設定為不同時啟動，即可使用編號相同的計時器。
- 當加法、減法等 2 種輸入同時上升時，將不會執行計數動作。當復歸信號 ON 時，現在值 = 0，此時計數器輸入將被關閉。截止計算旗標會在進位 (加法運算) 及退位減法運算時 ON，除此之外皆為 OFF。
- CNTR/CNTRX 指令在階梯圖及指令欄中的輸入順序會出現下列差異。
 - 階梯圖：
 - 加法輸入 → CNTR/CNTRX 指令 → 減法輸入 → 復歸信號
 - ニモニック：
 - 加法輸入 → 減法輸入 → 復歸信號 → CNTR/CNTRX 指令

程式例

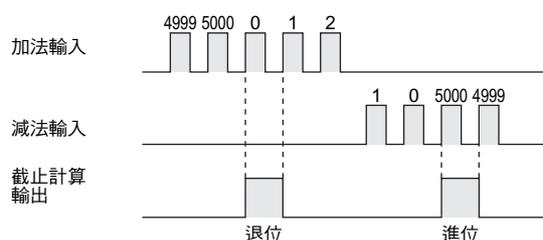
當復歸信號 0.02 變成 ON 時，計數器現在值將變為 0，每當加法計數器輸入 0.00 由 OFF → ON 時，計數器現在值就會 +1。當計數器現在值變成設定值 3 的狀態時，加法計數器輸入 0.00 會由 OFF → ON，此時，計數器現在值會變成 0，同時計數器旗標也會變為 ON。每當減法計數器輸入 0.01 由 OFF → ON 時，計數器現在值就會 -1。當計數器現在值變成設定值 0 的狀態時，接下來加法計數器輸入 0.01 會由 OFF → ON，此時，計數器現在值會變成設定值 3，同時計數器旗標也會變為 ON。



當加法、減法計數器輸入 (OFF → ON) 的訊號上升時，就會執行 1 次計數動作，當上述 2 種輸入同時 ON 時，皆不會執行計數。此外，當復歸信號 ON 時，現在值將變成 0，此時無法接受計數器輸入。當輸入繼電器 0.03 變成 ON 時，設定值將取決於 1CH 的內容，CNTR0007。



指令	資料
LD	0.00
LD	0.01
LD	0.02
CNTR (012)	0006
	#5000
LD	C0006
OUT	200.07
LD	0.03
LD	0.04
LD	0.05
CNTR (012)	0007
	1
LD NOT	C0007
OUT	200.08



CNR/CNRX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
復歸計時器 / 計數器	CNR	@CNR	545	可復歸指定範圍內的計數器 / 計數器的時間到旗標。
	CNRX	@CNRX	547	

符號	CNR	CNRX
	採用 BCD 方式時	採用 BIN 方式時

D1: 計時器/計數器編號1
D2: 計時器/計數器編號2

D1: 計時器/計數器編號1
D2: 計時器/計數器編號2

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D1	計時器 / 計數器編號 1	TIMER/COUNTER ¹	可調整
D2	計時器 / 計數器編號 2	TIMER/COUNTER ¹	可調整

? 1 只有當 D1 和 D2 為相同變數時才會啟動

D1: 計時器 / 計數器編號 1

T000 ~ T255 或 C000 ~ C255

D2: 計時器 / 計數器編號 2

T000 ~ T255 或 C000 ~ C255

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D1,D2	-	-	-	-	○	○	-	-	-	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 D1 和 D2 非相同區域時 ON 否則為 OFF

功能

復歸編號 D1 的計時器 / 計數器 ~ 編號 D2 的計時器 / 計數器等所有的時間到旗標。將現在值同時設定為最大值 (使用 BCD 方式時: #9999、使用 BIN 方式時: #FFFF)。(執行編號為 D1 ~ D2 的計時器 / 計數器指令時, 設定值會被設定為現在值。)

使用上的注意事項

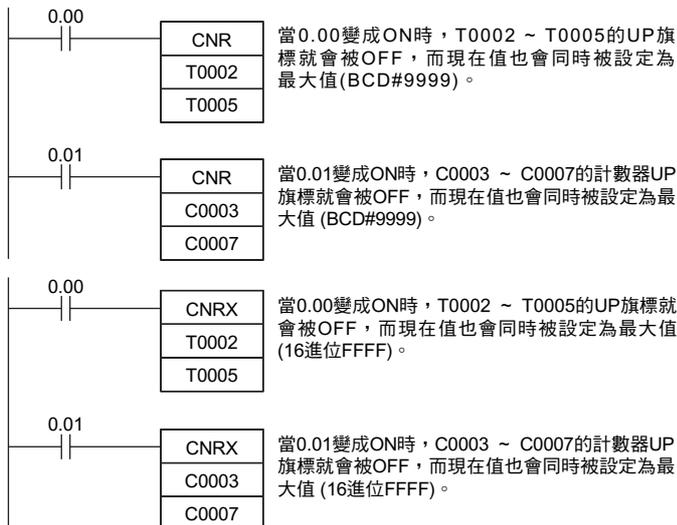
- 會被復歸的計時器 / 計數器如下：

對象指令		執行 CNR 指令時的動作	對象指令		執行 CNRX 指令時的動作
BCD	TIM (100 ms 計時器)	將現在值設定為最大值 (BCD#9999) 並且將時間到旗標 OFF。	BIN	TIMX (100 ms 計時器)	將現在值設定為最大值 (16 進位 #FFFF) 並且將時間到旗標 OFF。
	TIMH (10 ms 計時器)			TIMHX (10 ms 計時器)	
	TMHH (1 ms 計時器)			TMHHX (1 ms 計時器)	
	TTIM (累加計時器)			TTIMX (累加計時器)	
	CNT (計數器)			CNTX (計數器)	
	CNTR (可逆計數器)			CNTRX (可逆計數器)	

不過，IML/TIMLX (長時間計時器) 指令不在復歸範圍內。

- 本指令並不會對指令進行復歸，而是將使用該指令的計時器 / 計數器現在值設定為最大值，然後再將時間到旗標復歸，此種作法和針對指令進行復歸時的動作相異，請特別注意。(例：針對 TIM/TIMX 指令進行復歸時，即可讓現在值 = 設定值，以及時間到旗標 = OFF，相對地如果利用 CNR/CNRX 指令進行復歸，將變成現在值 = 最大值，時間到旗標 = OFF。
- 將 D1、D2 指定為 D1 > D2 時，只有編號為 D1 的計時器 / 計數器時間到旗標會被復歸。

程式例



資料比較指令

=, <>, <, <=, >, >=

指令名稱	指令記號	指令的各種組合	Fun No.	功能
符號比較	=, <>, <, <=, >, >=	—	300 ~ 328	比較 CH 資料或常數，當比較結果為 True (真) 時，就會連接至下一段以後的回路，亦可比較資料格式 (不帶符號、帶符號)、資料長度 (字元、倍精確度) 是否一致。

符號	=, <>, <, <=, >, >=		
	LD (LOAD) 連接型	AND (串接) 連接型	OR (並聯) 連接型

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態				容量	
		不帶符號	倍精確度・不帶符號	附帶符號	倍精確度・附帶符號	字元	倍精確度
S1	比較資料 1	UINT	UDINT	INT	DINT	1	2
S2	比較資料 2	UINT	UDINT	INT	DINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1	○	○	○	○	○	○	○	○	○	○	—	—	—
S2	○	○	○	○	○	○	○	○	○	○	—	—	—

條件旗標的動作

名稱	標籤	內容	
		資料長度：字元時	資料長度：倍長場合
異常旗標	P_ER	無任何變化	無任何變化
> 旗標	P_GT	<ul style="list-style-type: none"> 當比較結果為 S1 > S2 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 S1+1、S1 > S2+1、S2 時，本旗標 ON 否則為 OFF
≥ 旗標	P_GE	<ul style="list-style-type: none"> 當比較結果為 S1 ≥ S2 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 S1+1、S1 ≥ S2+1、S2 時，本旗標 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當比較結果為 S1 = S2 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 S1+1、S1 = S2+1、S2 時，本旗標 ON 否則為 OFF
≠ 旗標	P_NE	<ul style="list-style-type: none"> 當比較結果為 S1 ≠ S2 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 S1+1、S1 ≠ S2+1、S2 時，本旗標 ON 否則為 OFF

名稱	標籤	內容	
		資料長度：字元時	資料長度：倍長場合
< 旗標	P_LT	<ul style="list-style-type: none"> 當比較結果為 $S1 < S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1、S1 < S2+1、S2$ 時，本旗標 ON 否則為 OFF
\leq 旗標	P_LE	<ul style="list-style-type: none"> 當比較結果為 $S1 \leq S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1、S1 \leq S2+1、S2$ 時，本旗標 ON 否則為 OFF
負旗標	P_N	無任何變化	無任何變化

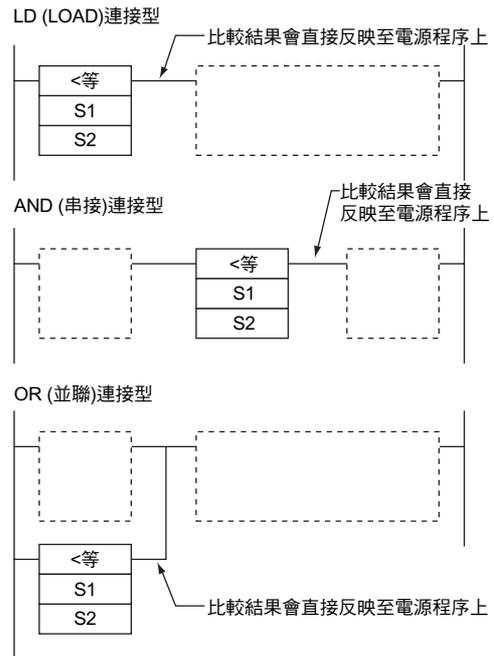
功能

利用帶符號或不帶符號表示法來比較 S2 和 S2 時，一旦比較結果為 True (真)，即連接下一段以後的回路，和 LD、AND、OR 指令的處理方法一樣，在各指令之後接著編寫其他指令。

LCD 型時：可直接連接母線。

AND 型時：無法直接與母線連接。

OR 型時：可直接連接母線。



● 符號及選項

將符號及選項互相組合後，即可利用 72 種指令欄來表示比較指令。

符號 (階梯程式中無 LD、AND、OR)	選項 (資料格式)	選項 (資料長度)
LD/AND/OR =、LD/AND/OR <> LD/AND/OR <、LD/AND/OR <= LD/AND/OR >、LD/AND/OR >=	無 (不帶符號) S (附帶符號)	無 (字元) L (倍精確度)

功能	資料形式資料長度	指令記號	名稱	FUN No.
當 $S1 = S2$ 時為 True (真) (ON)	不帶符號·字元型	LD/AND/OR =	一致	300
	不帶符號·倍精確度型	LD/AND/OR =L	倍精確度·一致	301
	附帶符號·字元型	LD/AND/OR =S	帶符號·一致	302
	附帶符號·倍精確度型	LD/AND/OR =SL	帶符號倍精確度·一致	303
當 $S1 \neq S2$ 時為 True (真) (ON)	不帶符號·字元型	LD/AND/OR <>	不一致	305
	不帶符號·倍精確度型	LD/AND/OR <>L	倍精確度·不一致	306
	附帶符號·字元型	LD/AND/OR <>S	帶符號·不一致	307
	附帶符號·倍精確度型	LD/AND/OR <>SL	帶符號倍精確度·不一致	308
當 $S1 < S2$ 時為 True (真) (ON)	不帶符號·字元型	LD/AND/OR <	小於	310
	不帶符號·倍精確度型	LD/AND/OR <L	倍精確度·小於	311
	附帶符號·字元型	LD/AND/OR <S	帶符號·小於	312
	附帶符號·倍精確度型	LD/AND/OR <SL	帶符號倍精確度·小於	313
當 $S1 \leq S2$ 時為 True (真) (ON)	不帶符號·字元型	LD/AND/OR <=	以下	315
	不帶符號·倍精確度型	LD/AND/OR <=L	倍精確度·以下	316
	附帶符號·字元型	LD/AND/OR <=S	帶符號·以下	317
	附帶符號·倍精確度型	LD/AND/OR <=SL	帶符號倍精確度·以下	318

功能	資料形式資料長度	指令記號	名稱	FUN No.
當 $S1 > S2$ 時為 True (真) (ON)	不帶符號・字元型	LD/AND/OR >	超過	320
	不帶符號・倍精確度型	LD/AND/OR >L	倍精確度・超過	321
	附帶符號・字元型	LD/AND/OR >S	帶符號・超過	322
	附帶符號・倍精確度型	LD/AND/OR >SL	帶符號倍精確度・超過	323
當 $S1 \geq S2$ 時為 True (真) (ON)	不帶符號・字元型	LD/AND/OR >=	以上	325
	不帶符號・倍精確度型	LD/AND/OR >=L	倍精確度・以上	326
	附帶符號・字元型	LD/AND/OR >=S	帶符號・以上	327
	附帶符號・倍精確度型	LD/AND/OR >=SL	帶符號倍精確度・以上	328

- 帶符號比較指令 (不帶 S 選項) 可處理不帶符號資料 (10 進位表示時為 & 0 ~ &65535 或是 16 進位表示時為 #0000 ~ FFFF) 及 BCD 資料。
- 帶符號比較指令 (帶 S 選項) 可處理帶符號資料 (10 進位表示時為 32768 ~ +32767 或是 16 進位表示時為 #0000 ~ FFFF)。

提示

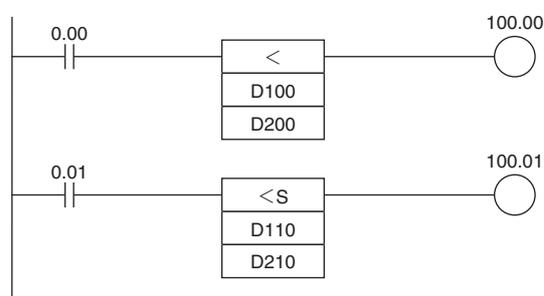
- 不同於 CMP 指令或 CMPL 指令，比較結果會被當作輸入條件直接反映至下一段的回路，由於不需要讀取條件旗標，因此程式敘述會更簡潔。

使用上的注意事項

- 請在本指令的最後一段附加輸出類指令 (OUT 類指令及下一段連接型指令以外的應用指令)。
- 本指令不適用於回路的最末段。

程式例

- 使用 AND 連接型的 < 指令、< S 指令時



- < 指令的動作
當 0.00 變成 ON 時，就會以不帶符號 BIN4 位數來比較資料記憶體 D100 及 D200 的資料。當比較結果為 (D100 的資料) < (D200 的資料) 時，就會連接至下一段以後的回路，此時輸出繼電器 100.00 將變成 ON。
當比較結果並非 (D100 的資料) < (D200 的資料) 時，則不會連接至下一段的回路。
- < S 指令的動作
當 0.01 變成 ON 時，就會利用帶符號 BIN4 位數來比較 D110 及 D210 的資料。當比較結果為 (D110 的資料) < (D210 的資料) 時，就會連接至下一段以後的回路，此時輸出繼電器 100.01 將變成 ON。
當比較結果並非 (D110 的資料) < (D210 的資料) 時，則不會連接至下一段的回路。

=DT , <> DT , < DT , <= DT , > DT , >= DT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
時刻比較	=DT	-	341	比較 2 種時間資料 (BCD 資料)，當比較結果為 True (真)
	<> DT		342	
	< DT		343	
	<= DT		344	
	> DT		345	
	>= DT		346	

符號	=DT , <> DT , < DT , <= DT , > DT , >= DT		
	LD (LOAD) 連接型	AND (串接) 連接型	OR (並聯) 連接型

助憶指令碼

C
S1
S2

C: 控制資料
S1: 現在時刻資料下位CH編號
S2: 比較時刻資料下位CH編號

助憶指令碼

C
S1
S2

C: 控制資料
S1: 現在時刻資料下位CH編號
S2: 比較時刻資料下位CH編號

助憶指令碼

C
S1
S2

C: 控制資料
S1: 現在時刻資料下位CH編號
S2: 比較時刻資料下位CH編號

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

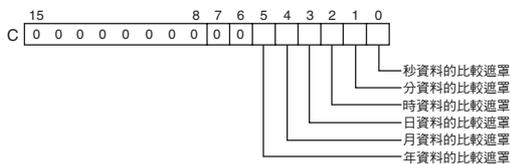
運算元種類的說明

運算元種類	內容	資料型態	容量
C	控制資料	WORD	1
S1	現在時刻資料下位 CH 編號	WORD	3
S2	比較時刻資料下位 CH 編號	WORD	3

C: 控制資料

利用位元 00 ~ 05 分別指定年 / 月 / 日 / 時 / 分 / 秒中的哪一項要當作比較遮罩 (排除在比較範圍外)，當全部項目都被遮罩 (位元 00 ~ 05 皆為 1) 時，將不執行指令，而且不會連接至下一段以後的回路。

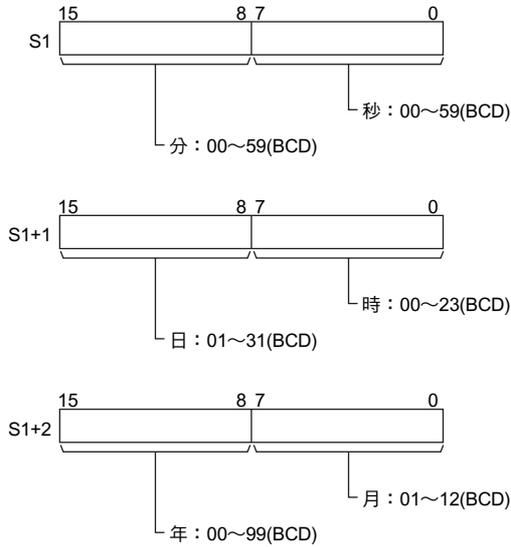
此時，異常旗標將變成 ON。



S1：現在時刻資料下位 CH 編號

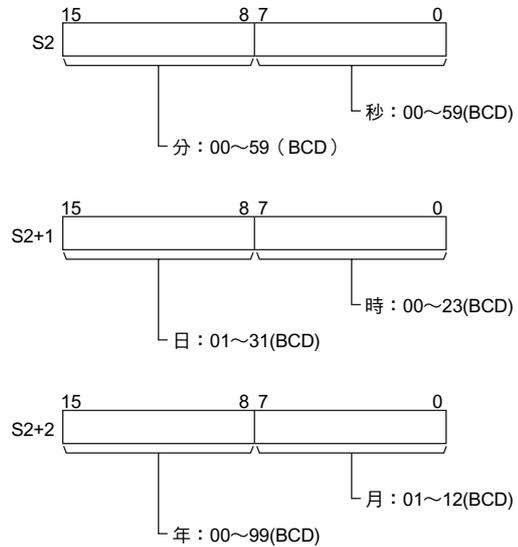
現在時刻 (年 / 月 / 日 / 時 / 分 / 秒) 資料將會被儲存在下列的 S1 ~ S1+2 中。

利用 CPU 模組的內部時間資料做為比較基準時，將以 S1 = A351CH 來指定 CPU 模組的內部時間資料 (A351 ~ 353CH)。



S2：比較時刻資料下位 CH 編號

比較時刻 (年 / 月 / 日 / 時 / 分 / 秒) 資料將會被儲存在下列的 S1 ~ S1+2 中。



註：年的形式為西元數字的後面 2 位數。又，如果以 97 年和 98 年為分界點，98 年表示西元 1998 年，97 則代表西元 2097 年。

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C	○	○	○	○	○	○	○	-	-	○	-	-	-
S1,S2	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 全部被遮罩時，本旗標 ON 否則為 OFF
> 旗標	P_GT	<ul style="list-style-type: none"> 當比較結果為 S1 > S2 時，本旗標 ON 否則為 OFF
≥ 旗標	P_GE	<ul style="list-style-type: none"> 當比較結果為 S1 ≥ S2 時，本旗標 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當比較結果為 S1 = S2 時，本旗標 ON 否則為 OFF
≠ 旗標	P_NE	<ul style="list-style-type: none"> 當比較結果為 S1 ≠ S2 時，本旗標 ON 否則為 OFF
< 旗標	P_LT	<ul style="list-style-type: none"> 當比較結果為 S1 < S2 時，本旗標 ON 否則為 OFF
≤ 旗標	P_LE	<ul style="list-style-type: none"> 當比較結果為 S1 ≤ S2 時，本旗標 ON 否則為 OFF

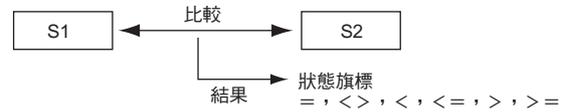
功能

C 所指定為 0 (比較對象) 的時刻項目 (年 / 月 / 日 / 時 / 分 / 秒) 會利用 BCD 方式來比較 S1 和 S2，當結果為 True (真) 時，就會連接至下一段以後的回路，並且將結果反映至條件旗標 (=、<>、<、<=、>、>=)，而時刻比較指令會以 18 種指令欄來表示。

C 所指定為 1 (比較遮罩) 的項目將不會執行比較動作。

又，當指令執行後，各項條件旗標將根據下列比較結果 ON/OFF。

比較結果	=	<>	<	<=	>	>=
S1=S2	ON	OFF	OFF	ON	OFF	ON
S1>S2	OFF	ON	OFF	OFF	ON	ON
S1<S2	OFF	ON	ON	ON	OFF	OFF

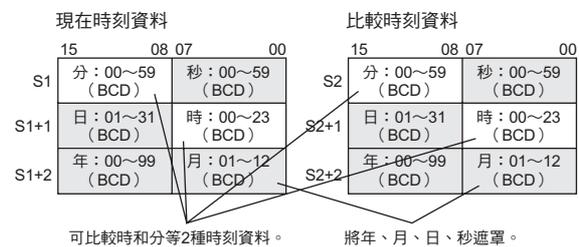


● 時刻資料的比較遮罩

所謂時刻資料的比較遮罩就是一種可以將比較對象以外的時刻資料遮罩 (隱藏) 的功能。利用此種功能，即可針對 C 所指定為 0 (比較對象) 的「年 / 月 / 日 / 時 / 分 / 秒」等 6 種時刻資料，個別進行比較。

例：當 C = #39 (以 2 進位表示時為 111001，因此年：1、月：1、日：1、時：0、分：0、秒：1) 時，只有時、分資料會被當作比較對象。(0 = 比較對象、1 = 遮罩)

如此一來，就能夠執行像是每天 a 時 b 分 ON 的行事曆計時器功能。



提示

- 傳統的資料比較指令在比較資料時是以 16 位元為單位，使用時刻比較指令，即可以 8 位元為單位來限定並比較時刻資料。
- CPU 模組的內部時刻資料會以 BCD 的形式儲存於下列特殊輔助繼電器中。

比較

通道 (CH)	位址		內容
	位元		
A351CH	00 ~ 07	秒 (00 ~ 59) (BCD)	
	08 ~ 15	分 (00 ~ 59) (BCD)	
A352CH	00 ~ 07	時 (00 ~ 23) (BCD)	
	08 ~ 15	日 (01 ~ 31) (BCD)	
A353CH	00 ~ 07	月 (01 ~ 12) (BCD)	
	08 ~ 15	年 (00 ~ 99) (BCD)	

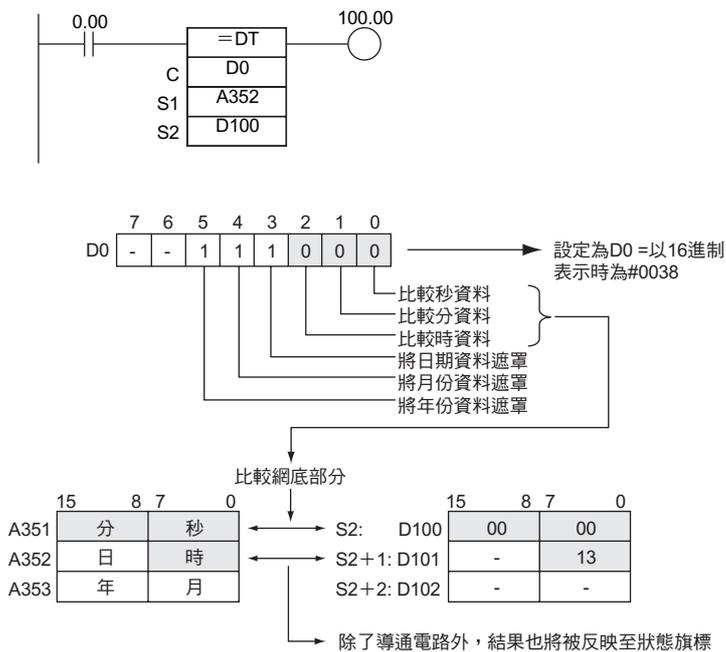
使用上的注意事項

- 請在本指令的最後一段附加輸出類指令 (OUT 類指令及下一段連接型指令以外的應用指令)。
- 本指令不適用於回路的最末段。
- CP1E CPU 模組 E 型 (CP1E-E □□□□ - 型) 未配備時間功能。
CPU 模組的內部時間資料已經被預設為 01 年 01 月 01 日 01 時 01 分 01 秒。

程式例

當 0.00 變成 ON 且現在時刻為 13 時 0 分 0 秒時，100.00 就會 ON。

以時 / 分 / 秒來比較 CPU 模組內建時鐘 A351 ~ A352 的現在時刻與 D100 ~ D102 的設定時刻。



CMP/CMPL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
不帶符號比較	CMP	!CMP	020	利用不帶符號 BIN16 位元 (16 進位 4 位數) 來比較 CH 資料或常數，然後再將比較結果反映至條件旗標。
不帶符號倍精確度比較	CMPL	—	060	利用不帶符號 BIN32 位元 (16 進位 8 位數) 來比較相當於 2CH 的 CH 資料，然後再將比較結果反映至條件旗標。

符號	CMP		CMPL						
		<table border="1"> <tr><td>CMP</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> </table> <p>S1：比較資料1 S2：比較資料2</p>	CMP	S1	S2		<table border="1"> <tr><td>CMPL</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> </table> <p>S1：比較資料1下位CH編號 S2：比較資料2下位CH編號</p>	CMPL	S1
CMP									
S1									
S2									
CMPL									
S1									
S2									

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		CMP	CMPL	CMP	CMPL
S1	CMP：比較資料 1 CMPL：比較資料 1 下位 CH 編號	UINT	UDINT	1	2
S2	CMP：比較資料 2 CMPL：比較資料 2 下位 CH 編號	UINT	UDINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—

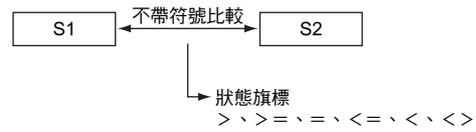
條件旗標的動作

名稱	標籤	內容	
		CMP	CMPL
異常旗標	P_ER	無任何變化	無任何變化
> 旗標	P_GT	<ul style="list-style-type: none"> 當比較結果為 $S1 > S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1 > S2+1$、$S1 > S2+1$、$S2$ 時，本旗標 ON 否則為 OFF
≥ 旗標	P_GE	<ul style="list-style-type: none"> 當比較結果為 $S1 \geq S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1 \geq S2+1$、$S1 \geq S2+1$、$S2$ 時，本旗標 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當比較結果為 $S1 = S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1 = S2+1$、$S1 = S2+1$、$S2$ 時，本旗標 ON 否則為 OFF
≠ 旗標	P_NE	<ul style="list-style-type: none"> 當比較結果為 $S1 \neq S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1 \neq S2+1$、$S1 \neq S2+1$、$S2$ 時，本旗標 ON 否則為 OFF
< 旗標	P_LT	<ul style="list-style-type: none"> 當比較結果為 $S1 < S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1 < S2+1$、$S1 < S2+1$、$S2$ 時，本旗標 ON 否則為 OFF
≤ 旗標	P_LE	<ul style="list-style-type: none"> 當比較結果為 $S1 \leq S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1 \leq S2+1$、$S1 \leq S2+1$、$S2$ 時，本旗標 ON 否則為 OFF
負旗標	P_N	無任何變化	無任何變化

- 執行 CMP 指令後，>、>=、=、<=、<、<> 的條件旗標將變成 ON/OFF。

比較結果	>	>=	=	<=	<	<>
S1 > S2	ON	ON	OFF	OFF	OFF	ON
S1 = S2	OFF	ON	ON	ON	OFF	OFF
S1 < S2	OFF	OFF	OFF	ON	ON	ON

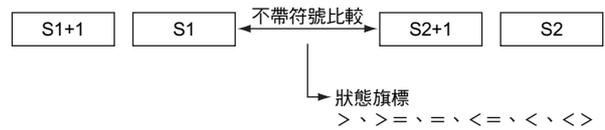
(註) - 表示依情況而異。



- 執行 CMPL 指令後，>、>=、=、<=、<、<> 的條件旗標將變成 ON/OFF。

比較結果	>	>=	=	<=	<	<>
S1+1, S1 > S2+1, S2	ON	ON	OFF	OFF	OFF	ON
S1+1, S1 = S2+1, S2	OFF	ON	ON	ON	OFF	OFF
S1+1, S1 < S2+1, S2	OFF	OFF	OFF	ON	ON	ON

(註) - 表示依情況而異。



功能

● CMP

利用不帶符號 BIN (16 進位 4 位數) 來比較 S1 和 S2，然後再將比較結果反映至條件旗標 (>、>=、=、<>、<、<=)。

● CMPL

利用不帶符號 BIN (16 進位 8 位數) 來比較 S1 和 S2，然後再將比較結果反映至條件旗標 (>、>=、=、<>、<、<=)。

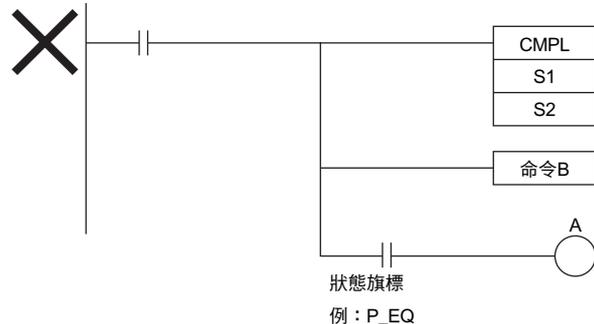
使用上的注意事項

- 執行本指令後，比較結果會被反映至條件旗標，因此請依照範例所示，利用和 CMP (CMPL) 指令輸入條件相同的輸出分歧來輸入結果。



當S1和S2一致時，P_EQ旗標將變成ON，而A也會變為ON。

- 將條件旗標配置在其他指令之後時，條件旗標將依其他指令的執行結果而改變，因此務必將條件旗標配置在 CMP (CMPL) 指令的正後方。



狀態旗標將完全接受指令B的執行結果。

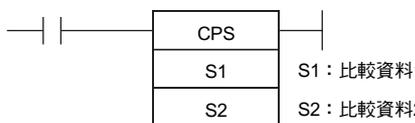
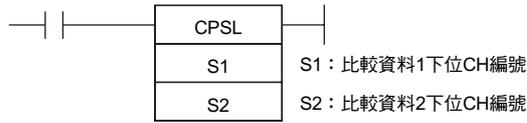
- CMP 指令可以被當作立即更新指令 !CMP 使用。

使用立即更新型指令 !CMP，即可將 CPU 模組內建輸入指定為 S1 或 S2 (或是 S1 及 S2)。

執行指令時，S1 或 S2 的數值會被 IN 更新，並且比較其數值。

CPS/CPSL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
帶符號 BIN 比較	CPS	!CPS	114	利用不帶符號 BIN16 位元 (以最上位位元為符號位元的 16 進位 4 位數) 來比較 CH 資料或常數，然後再將比較結果反映至條件旗標。
帶符號 BIN 倍精確度比較	CPSL	—	115	利用帶符號倍精確度 BIN32 位元 (以最上位位元為符號位元的 16 進位 8 位數) 來比較相當於 2CH 的 CH 資料或常數，然後再將比較結果反映至條件旗標。

符號	CPS	CPSL
		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		CPS	CPSL	CPS	CPSL
S1	CPS：比較資料 1 CPSL：比較資料 1 下位 CH 編號	INT	DINT	1	2
S2	CPS：比較資料 2 CPSL：比較資料 2 下位 CH 編號	INT	DINT	1	2

● 運算元種類

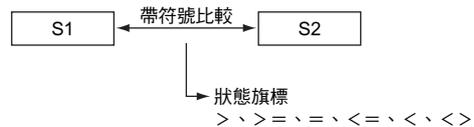
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—

條件旗標的動作

名稱	標籤	內容	
		CPS	CPSL
異常旗標	P_ER	無任何變化	無任何變化
> 旗標	P_GT	<ul style="list-style-type: none"> 當比較結果為 $S1 > S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1、S1 > S2+1、S2$ 時，本旗標 ON 否則為 OFF
\geq 旗標	P_GE	<ul style="list-style-type: none"> 當比較結果為 $S1 \geq S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1、S1 \geq S2+1、S2$ 時，本旗標 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當比較結果為 $S1 = S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1、S1 = S2+1、S2$ 時，本旗標 ON 否則為 OFF
\neq 旗標	P_NE	<ul style="list-style-type: none"> 當比較結果為 $S1 \neq S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1、S1 \neq S2+1、S2$ 時，本旗標 ON 否則為 OFF
< 旗標	P_LT	<ul style="list-style-type: none"> 當比較結果為 $S1 < S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1、S1 < S2+1、S2$ 時，本旗標 ON 否則為 OFF
\leq 旗標	P_LE	<ul style="list-style-type: none"> 當比較結果為 $S1 \leq S2$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S1+1、S1 \leq S2+1、S2$ 時，本旗標 ON 否則為 OFF
負旗標	P_N	無任何變化	無任何變化

● 執行 CPS 指令後，>、>=、=、<=、<、<> 的條件旗標將變成 ON/OFF。

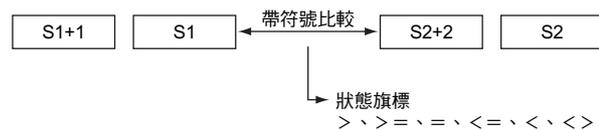
比較結果	>	>=	=	<=	<	<>
$S1 > S2$	ON	ON	OFF	OFF	OFF	ON
$S1 = S2$	OFF	ON	ON	ON	OFF	OFF
$S1 < S2$	OFF	OFF	OFF	ON	ON	ON



(註) - 表示依情況而異。比較資料 1、2 可指定的範圍為帶符號 BIN (以 10 進位表示時為 -32768 ~ 32767 或是以 16 進位表示時為 #8000 ~ 7FFF)。

● 執行 CPSL 指令後，>、>=、=、<=、<、<> 的條件旗標將變成 ON/OFF。

比較結果	>	>=	=	<=	<	<>
$S1+1, S1 > S2+1, S2$	ON	ON	OFF	OFF	OFF	ON
$S1+1, S1 = S2+1, S2$	OFF	ON	ON	ON	OFF	OFF
$S1+1, S1 < S2+1, S2$	OFF	OFF	OFF	ON	ON	ON



(註) - 表示依情況而異。比較資料 1、2 可指定的範圍為帶符號 BIN (以 10 進位表示時為 -2147483648 ~ 2147483647 或是以 16 進位表示時為 #80000000 ~ 7FFFFFFF)。

功能

● CPS

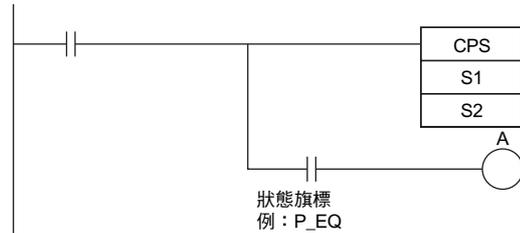
利用帶符號 BIN (以最上位位元為符號位元為的 16 進位 4 位數) 來比較 S1 和 S2，然後再將結果反映至條件旗標 (>、>=、=、<=、<、<>)。

● CPSL

利用帶符號 BIN 倍精確度 (以最上位位元為符號位元為的 16 進位 8 位數) 來比較 S1 和 S2，然後再將結果反映至條件旗標 (>、>=、=、<=、<、<>)。

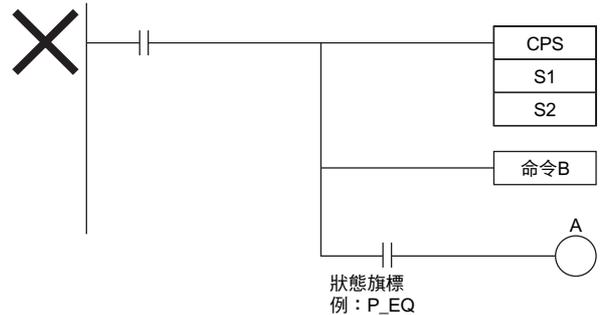
使用上的注意事項

- 執行本指令後，比較結果會被反映至條件旗標，因此請依照範例所示，利用和 CPS 指令輸入條件相同的輸出分歧來輸入結果。



當S1和S2一致時，P_EQ旗標將變成ON，而A也會變為ON。

- 將條件旗標配置在其他指令之後時，條件旗標將依其他指令的執行結果而改變，因此務必將條件旗標配置在 CPS 指令的正後方。



狀態旗標將完全接受指令B的執行結果。

- CPS 指令可當作立即更新型指令 (ICPS) 使用。

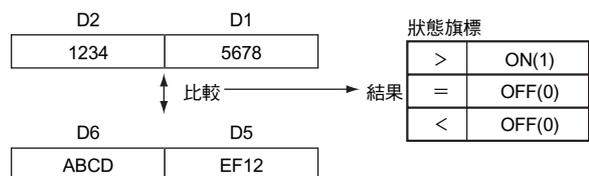
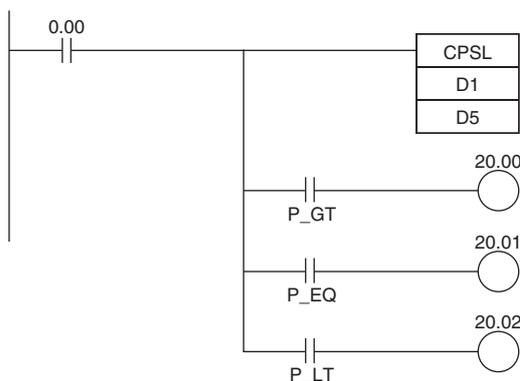
使用立即更新型指令 (ICPS)，即可將 CPU 模組內建輸入指定為 S1 或 S2 (或是 S1 及 S2)。執行指令時，S1 或 S2 的數值會被 IN 更新，並且比較其數值。

程式例

當 0.00 變成 ON 時，就會利用帶符號 BIN 來比較資料記憶體 D2、D1 的資料內容及 D6、D5 的資料內容。當比較結果為 D2、D1 的資料內容較大時，條件旗標 > 就會變成 ON，此時輸出繼電器 20.00 也會 ON。

當 D2、D1 的資料內容和 D6、D5 的資料內容相同時，= 就會變成 ON，此時 20.01 也會 ON。

當 D6、D5 的資料內容大於 D2、D1 的資料內容時，< 就會變成 ON。



TCMP

指令名稱	指令記號	指令的各種組合	Fun No.	功能
比較表一致	TCMP	@TCMP	085	針對比較資料 1CH 和相當於比較資料表 16CH 的資料比較其是否一致，接著再將 1 輸出至輸出 CH 所對應的位元。

符號	TCMP	
		<p>S: 比較資料</p> <p>T: 比較資料表下位CH編號</p> <p>D: 比較結果輸出CH編號</p>

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

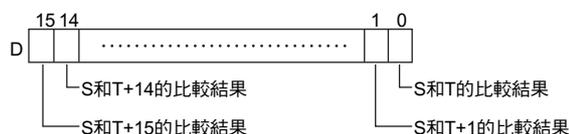
運算元種類的說明

運算元種類	內容	資料型態	容量
S	比較資料	WORD	1
T	比較資料表下位 CH 編號	WORD	16
D	比較結果輸出 CH 編號	UINT	1

T：比較資料表下位 CH 編號

T	比較資料0
T+1	比較資料1
~	~
T+15	比較資料15

D：比較結果輸出 CH 編號



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	-	-	-
T,D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
=旗標	P_EQ	<ul style="list-style-type: none"> 當比較結果為 0 時 (和比較資料表比較後的結果以及 S 資料皆不一致時), 本旗標就會 ON 否則為 OFF

功能

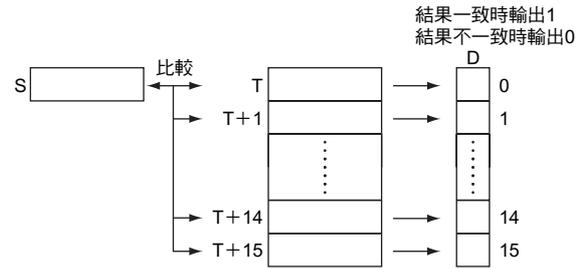
分別比較 S 所指定的 1CH(字元) 和 T ~ T+15 所指定的 16CH(字元) 資料，當比較結果一致時就將 1 輸出至 D 所對應的位元，如果不一致時則輸出 0。

比較 S 和 T，只要比較結果一致，就將 1 輸出至 DCH 的位元 0，如果不一致時則輸出 0。

比較 S 和 T+1，只要比較結果一致，就將 1 輸出至 DCH 的位元 0，如果不一致時則輸出 1。

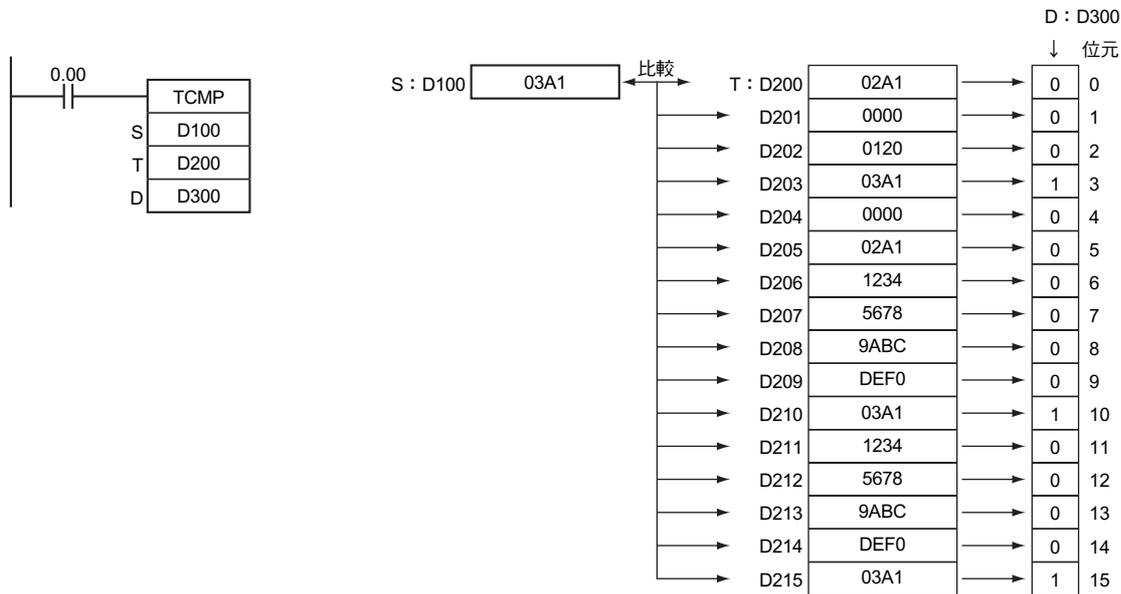
以下相同。

比較 S 和 T+15，只要比較結果一致，就將 1 輸出至 DCH 的位元 0，如果不一致時則輸出 15。



程式例

當0.00變成ON時，比較D100和D200 ~ D215，當比較結果一致時，將1儲存至D300的位元0~15，不一致時則輸出0。



BCMP

指令名稱	指令記號	指令的各種組合	Fun No.	功能
不帶符號資料表比較	BCMP	@BCMP	068	判斷比較資料的內容是否在 16 組比較資料的上下限範圍內，如果在範圍內時，就輸出 1 至輸出 CH 所對應的位元。

符號	BCMP	
		S : 比較資料 T : 比較資料表下位CH編號 D : 比較結果輸出CH編號

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

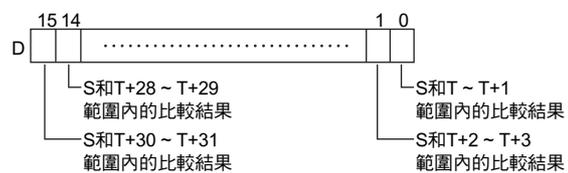
運算元種類的說明

運算元種類	內容	資料型態	容量
S	比較資料	WORD	1
T	比較資料表下位 CH 編號	WORD	32
D	比較結果輸出 CH 編號	UINT	1

T : 比較資料表下位 CH 編號

T	下限值0
T+1	上限值0
T+2	下限值1
T+3	上限值1
~	~
T+30	下限值15
T+31	上限值15

D : 比較結果輸出 CH 編號



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	-	-	-
T,D	○	○	○	○	○	○	○	○	○	○	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當比較結果為 0 時 (和比較資料表比較後的結果以及 S 資料皆超出範圍時)，本旗標就會 ON 否則為 OFF

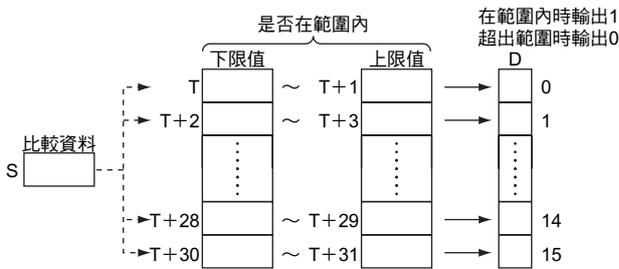
功能

將 T 所指定的 32 字元資料當作 16 組上限值、下限值資料，然後比較 S 是否符合每一組所設定的範圍 (包含上限值、下限值)。當比較結果在範圍內 (包含一致的情況) 時，就會輸出 1 至 D 所對應的位元，超出範圍 (不包含一致) 時，則輸出 0。

T、T+2、...、T+28、T+30 為下限值。

T+1、T+3、...、T+29、T+31 為上限值。

比較 S 是否在 T (下限值) ~ T+1 (上限值)、T+2 (下限值) ~ T+3 (上限值)、... 等範圍內，接著再將結果輸出至 DCH 的位元 0、1、... 等。



例：

當 $T \leq S \leq T+1$ 時，就會輸出 1 至 DCH 的位元 0。

當 $T+2 \leq S \leq T+3$ 時，就會輸出 1 至 DCH 的位元 1。

以下相同。

當 $S < T$ 、 $T+1 < S$ 時，就會輸出 0 至 DCH 的位元 0。

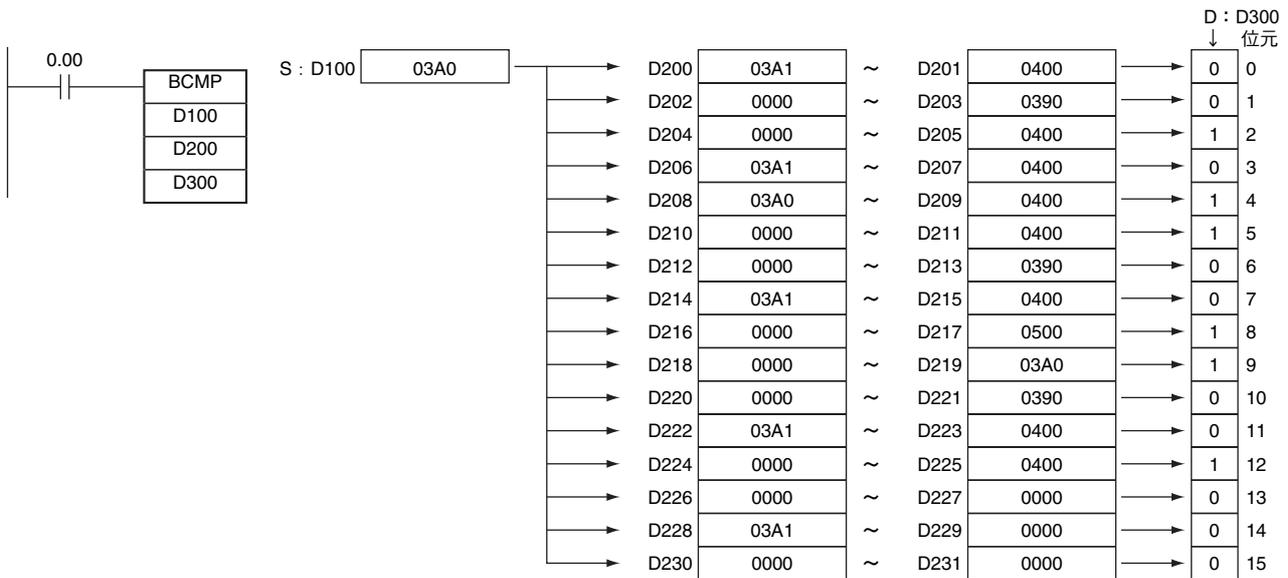
當 $S < T+2$ 、 $T+3 < S$ 時，就會輸出 1 至 DCH 的位元 0。

以下相同。

(註) 當下限值和上限值相反 ($T > T+1$ 等) 時，就會輸出 0 (超出範圍) 至 D 所對應的位元，而不會產生錯誤。

程式例

當 0.00 變成 ON 時，只要 D100 的內容是在以 D200 的內容為下限值，以及以 D201 的內容為上限值所構成的範圍內時，就會將 1 儲存至 D300 的位元，超出範圍時，則儲存 0。同樣地，當 D100 的內容在 D202 ~ D203、... D214 ~ D215 的範圍內時，就會將 1 儲存至 D300 的位元 1 ... 15，超出範圍時則儲存 0。



ZCP/ZCPL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
區域比較	ZCP	—	088	利用不帶符號 BIN16 位元 (16 進位 4 位數) 來比較您所指定的 1 個 CH 資料或常數是否符合您所指定上限值及下限值之間的範圍，接著再將比較結果反映至條件旗標。
倍精確度區域比較	ZCPL	—	116	利用不帶符號 BIN32 位元 (16 進位 8 位數) 來比較您所指定的 1 個倍精確度 CH 資料或常數是否符合您所指定上限值及下限值之間的範圍，接著再將比較結果反映至條件旗標。

符號	ZCP			ZCPL			
		ZCP	S	S : 比較資料(1CH資料)		ZCPL	S
	T1	T1	T1 : 下限值		T1	T1	T1 : 下限值下位CH編號
	T2	T2	T2 : 上限值		T2	T2	T2 : 上限值下位CH編號

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		ZCP	ZCPL	ZCP	ZCPL
S	ZCP : 比較資料 (1CH 資料) ZCPL : 比較資料 (2CH 資料)	UINT	UDINT	1	2
T1	ZCP : 下限值 ZCPL : 下限值下位 CH 編號	UINT	UDINT	1	2
T2	ZCP : 上限值 ZCPL : 上限值上位 CH 編號	UINT	UDINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,T1,T2	○	○	○	○	○	○	○	○	○	○	—	—	—

條件旗標的動作

名稱	標籤	內容	
		ZCP	ZCPL
異常旗標	P_ER	設定為 T1 > T2 時，本旗標就會 ON	設定為 T1+1、T1 > T2+1、T2 時，本旗標就會 ON
> 旗標	P_GT	<ul style="list-style-type: none"> 當比較結果為 S > T2 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 S+1、S > T2+1、T2 時，本旗標 ON 否則為 OFF
≥ 旗標	P_GE	無任何變化	無任何變化
= 旗標	P_EQ	<ul style="list-style-type: none"> 當比較結果為 T1ST2 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 T1+1、T1S+1、ST2+1、T2 時，本旗標 ON 否則為 OFF
≠ 旗標	P_NE	無任何變化	無任何變化

名稱	標籤	內容	
		ZCP	ZCPL
< 旗標	P_LT	<ul style="list-style-type: none"> 當比較結果為 $S < T1$ 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當比較結果為 $S+1、S < T1+1、T1$ 時，本旗標 ON 否則為 OFF
\leq 旗標	P_LE	無任何變化	無任何變化
負旗標	P_N	無任何變化	無任何變化

功能

● ZCP

利用不帶符號 BIN (16 進位 4 位數) 來比較 S 是否符合 $T1 \sim T2$ 的範圍 ($T1 \leq S \leq T2$)，然後再將結果反映至條件旗標 (>、=、<)。

ZCP 指令執行後，>、=、< 等各種條件旗標就會 ON/OFF (>=、<=、<> 旗標則不會 ON/OFF)。如右圖所示，當 $S > T2$ 時，> 旗標將變為 ON。

當 $T1 \leq S \leq T2$ 時，= 旗標就會 ON。當 $S < T1$ 時，< 旗標將變為 ON。

● ZCPL

利用不帶符號 BIN (16 進位 8 位數) 來比較 S 是否符合 $T1 \sim T2$ 的範圍 ($T1 \leq S \leq T2$)，然後再將結果反映至條件旗標 (>、=、<)。

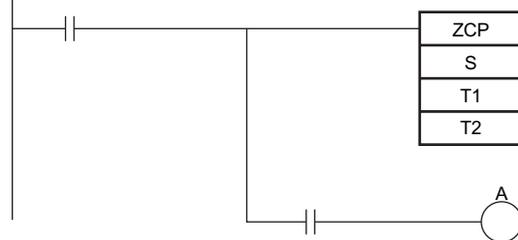
ZCPL 指令執行後，>、=、< 等各種條件旗標就會 ON/OFF (>=、<=、<> 旗標則不會 ON/OFF)。如右圖所示，當 $S+1、S > T2+1、T2$ 時，> 旗標將變為 ON。當 $T1+1、T1 \leq S+1、S \leq T2+1、T2$ 時，= 旗標將變為 ON。當 $S+1、S < T1+1、T1$ 時，< 旗標將變為 ON。

比較結果	>	=	<
$S > T2$	ON	OFF	OFF
$S = T2$	OFF	ON	
$T1 < S < T2$			
$S = T1$			
$S < T1$		OFF	ON

比較結果	>	=	<
$S+1, S > T2+1, T2$	ON	OFF	OFF
$S+1, S = T2+1, T2$	OFF	ON	
$T1+1, T1 < S+1, S < T2+1, T2$			
$S+1, S = T1+1, T1$			
$S+1, S < T1+1, T1$		OFF	ON

使用上的注意事項

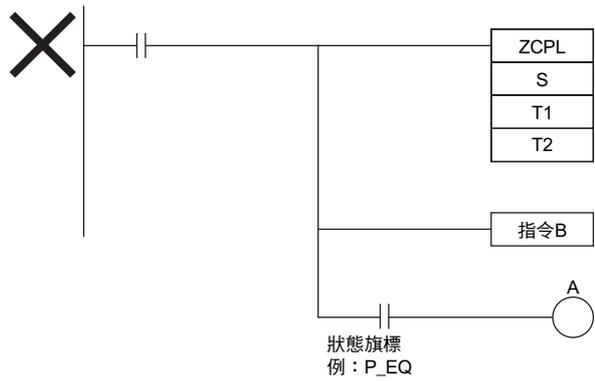
- 執行本指令後，比較結果會被反映至條件旗標，因此請依照範例所示，利用和 ZCP (ZCPL) 指令輸入條件相同的輸出分歧來輸入結果。



狀態旗標
例：P_EQ

當 $T1 \leq S \leq T2$ 時，P_EQ 旗標將變為 ON，此時 A 也會 ON。

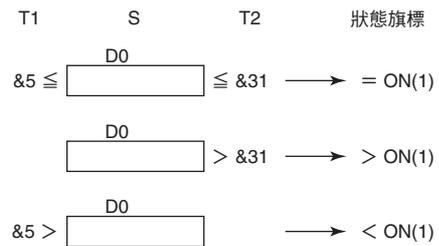
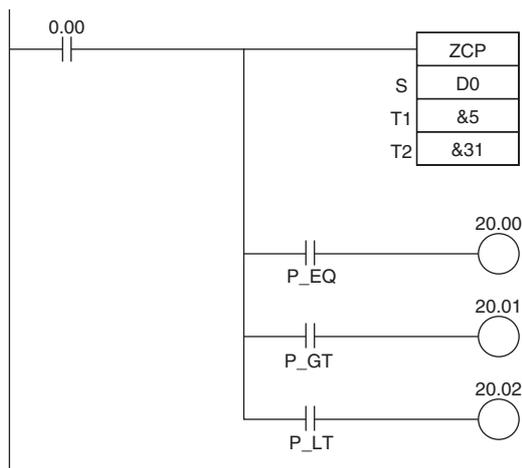
- 將條件旗標配置在其他指令之後時，條件旗標將依其他指令的執行結果而改變，因此務必將條件旗標配置在 ZCP (ZCPL) 指令的正後方。



狀態旗標將完全接受指令B的執行結果。

程式例

- 當輸入繼電器 0.00 變成 ON 時，就會比較 D0 的內容是否在 10 進位表示時為 5 ~ 31 或是 16 進位表示時為 #0005 ~ 001F 的範圍內。
- 當 D0 的內容 > 31 時，20.01 會變為 ON，當 $5 \leq D0$ 的內容 ≤ 31 時，20.00 會變為 ON，如果是 D0 的內容 < 5 時，20.02 將變為 ON。



資料傳送指令

MOV/MOVL/MVN

指令名稱	指令記號	指令的各種組合	Fun No.	功能
傳送	MOV	@MOV, !MOV, !@MOV	021	將 CH 資料或常數傳送至指定的 CH。
傳送倍精確度	MOVL	@MOVL	498	以 32 位元為單位，將相當於 2CH 的 CH 資料或常數輸出至傳送目的端的 CH。
否定傳送	MVN	@MVN	022	將 CH 資料或常數的位元反相資料傳送至您所指定的 CH。

符號	MOV	MOVL
	MVN	

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		MOV/MVN	MOVL	MOV/MVN	MOVL
S	MOV/MVN：傳送資料 MOVL：傳送資料下位 CH 編號	WORD	DWORD	1	2
D	MOV/MVN：傳送目的端的 CH 編號 MOVL：傳送目的端下位 CH 編號	WORD	DWORD	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

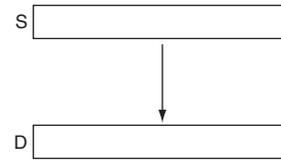
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當您所傳送的資料 (D) 為 0 時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 當您所傳送的資料 (D) 最上位元為 1 時，本旗標就會 ON 否則為 OFF

功能

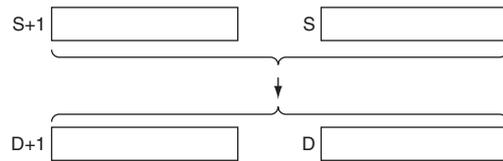
● MOV

將 D 傳送至 S，當 S 為常數時，本指令可用來設定資料。



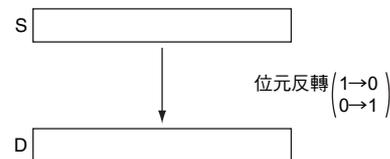
● MOVL

將 S 當作倍精確度資料，並傳送至 D+1、D。當 S、S+1 為常數時，本指令可用來設定資料。



● MVN

將 S 的 16 位元反相，並且傳送至 D。



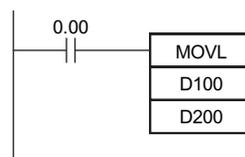
使用上的注意事項

- MOV 指令可當作立即更新型指令 (!MOV) 使用。
- 使用立即更新型指令 (!MOV) 時，即可將 CPU 模組內建的輸入繼電器區指定為 S，而且，還可以為 D 指定 CPU 模組內建的輸出繼電器區。
為 S 指定 CPU 模組內建的輸入時，只要指令開始執行，即可執行 S 值的 IN 更新，並且將更新後的數值傳送至 D。為 D 指定 CPU 模組的內建輸出時，只要指令開始執行，S 值就會被傳送至 D，並且即時進行 OUT 更新。本指令可同時對 S 執行 IN 更新以及對 D 執行 OUT 更新。

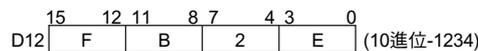
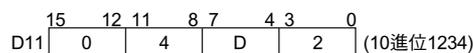
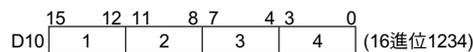
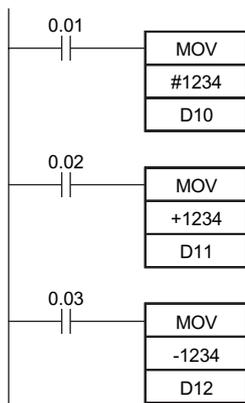
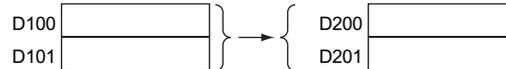
程式例



當 0.00 變成 ON 時，就會將 100CH 傳送至 D100。



當 0.00 變成 ON 時，就會將 D100 ~ D101 傳送至 D200 ~ D201。



MOVB

指令名稱	指令記號	指令的各種組合	Fun No.	功能
位元傳送	MOVB	@MOVB	082	傳送指定的位元。

符號	MOVB	
		S：傳送端CH編號 C：控制資料 D：傳送目的端的CH編號

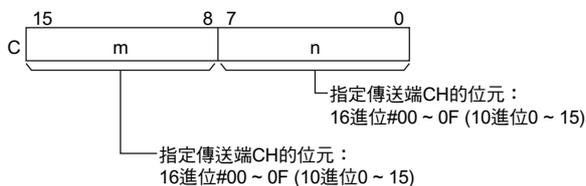
是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
S	傳送端 CH 編號	WORD	1
C	控制資料	UINT	1
D	傳送目的端的 CH 編號	WORD	1

C：控制資料



● 運算元種類

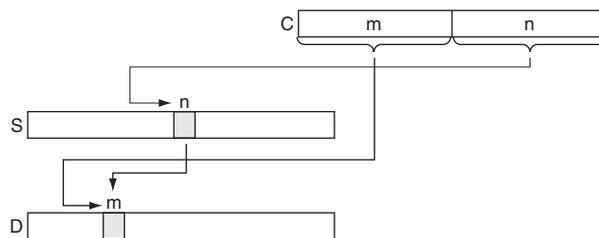
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S, C	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 C 的資料超過範圍時變為 ON 否則為 OFF

功能

將 S 所指定的位元位置 (C 的 n) 內容 (0/1) 傳送至 D 所指定的位元位置 (C 的 m)。



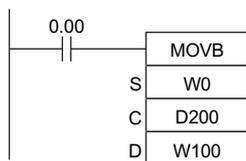
提示

- 只要在 S 和 D 指定相同的 CH，即可將本指令用來變更位元位置等。

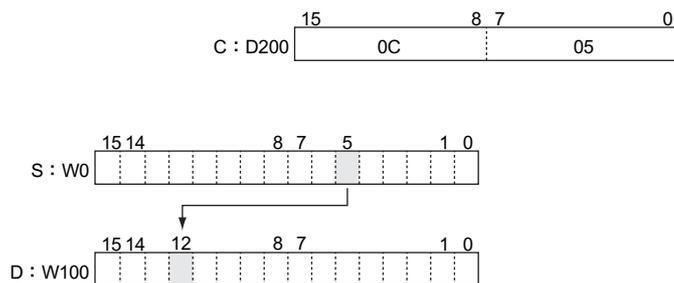
使用上的注意事項

- 除了已傳送的位元外，傳送目的端的 CH 資料將不會改變。

程式例



當0.00變成ON時，只要控制資料(C)出現下列內容，就會將W0的位元5傳送至W100的位元12。



MOVD

指令名稱	指令記號	指令的各種組合	Fun No.	功能
位數傳送	MOVD	@MOVD	083	傳送時以位數 (4 位元) 為單位，亦可傳送多位數。

符號	MOVD	
		S : 傳送端CH編號 C : 控制資料 D : 傳送目的端的CH編號

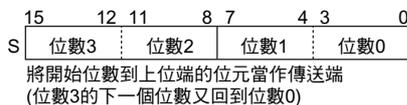
是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

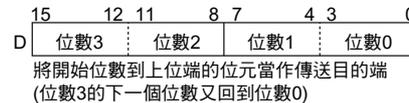
運算元種類的說明

運算元種類	內容	資料型態	容量
S	傳送端 CH 編號	WORD	1
C	控制資料	UINT	1
D	傳送目的端的 CH 編號	UINT	1

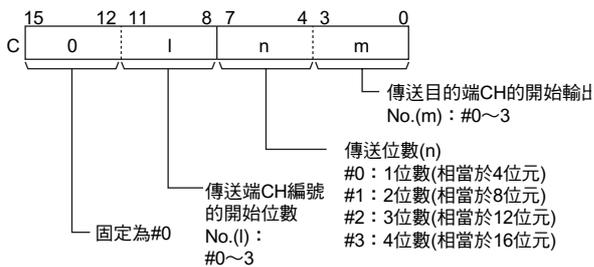
S：傳送端 CH 編號



D：傳送目的端的 CH 編號



C：控制資料



● 運算元種類

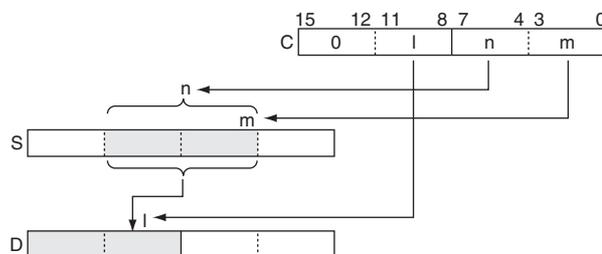
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S,C	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 C 的資料超過範圍時變為 ON 否則為 OFF

功能

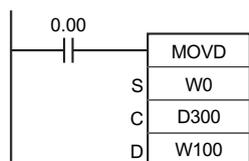
以 1 位數為 4 位元，將 S 所指定傳送的開始位元 (C 的 m) ~ 指定傳送位數 (C 的 n) 的內容傳送至 D 指定的開始輸出位數 (C 的 l) 之後的位數。



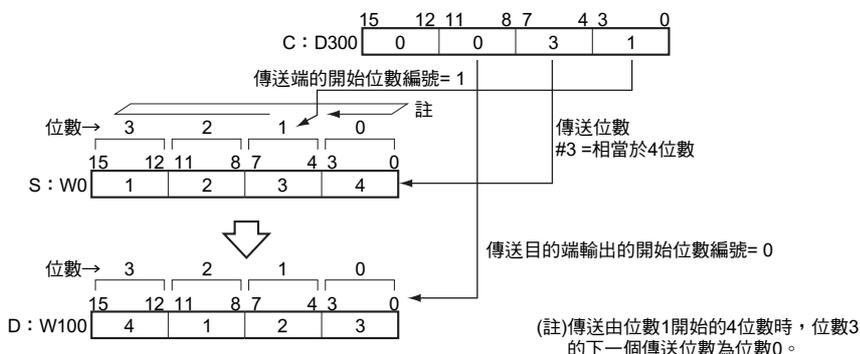
使用上的注意事項

- 此時除了已傳送位數外，傳送目的端的 CH 資料將不會發生任何變化。
- 傳送多位數時，由傳送端 CH 的最上位位數衍生出來的位數會被傳送到相同 CH 的最下位位數端。

程式例



當 0.00 變為 ON 時，只要控制資料 (C) 出現下列內容，就會將 W0 的位數 1 ~ 上位端的 4 位數資料，傳送到 W100 的位數 0 ~ 上位端的 4 位數。



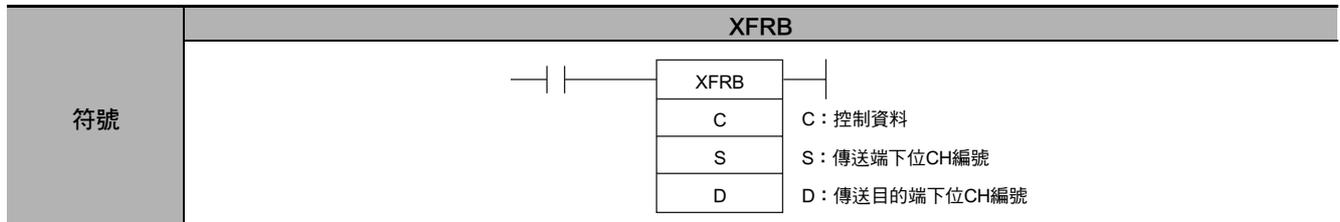
● 多位數 (DIGIT) 傳送範例

傳送多位數時，必須將傳送端的開始未數編號及傳送目的端的開始輸出位數編號指定為下位位數端。



XFRB

指令名稱	指令記號	指令的各種組合	Fun No.	功能
多位數傳送	XFRB	@XFRB	062	傳送指定 CH 所指定的多位數。



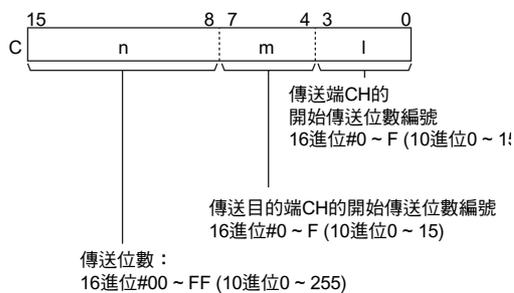
是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

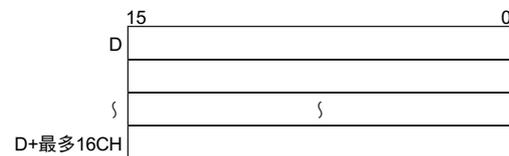
運算元種類的說明

運算元種類	內容	資料型態	容量
C	控制資料	UINT	1
S	傳送端下位 CH 編號	WORD	可調整
D	傳送目的端下位 CH 編號	WORD	可調整

C：控制資料

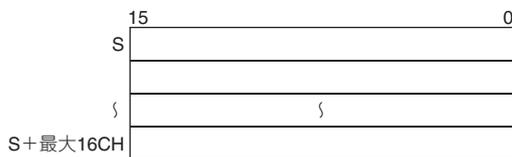


D：傳送目的端下位 CH 編號



(註) S ~ S+ 等最多 16CH 及 D ~ D+ 等最多 16CH，其中每個 CH 的運算元種類必須相同。

S：轉送元下位 CH 番号



● 運算元種類

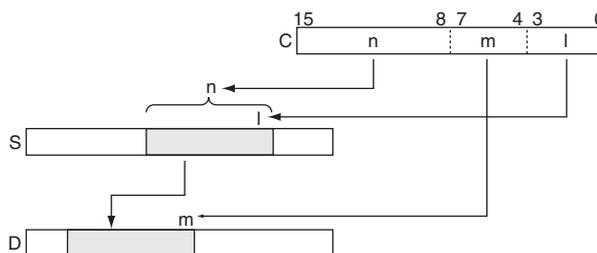
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C	○	○	○	○	○	○	○	○	○	○	-	-	-
S,D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF

功能

將 S 所指定的傳送端下位 CH 編號中指定的開始位元位置 (C 的 1) ~ 指定位元數 (C 的 n) 的資料傳送到 D 所指定的傳送目的端下位 CH 編號所指定的開始位元以後 (C 的 m) 的位置。



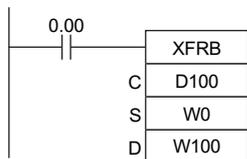
提示

- 利用 1 個指令即可傳送跨多個 CH 的資料，也就是最多相當於 255 位元的資料。
- 利用傳送跨多個 CH 資料的方式，即可壓縮資料，並且有效運用資料區。(特別是在處理定位等位置資料時將更輕鬆。)
- 即使傳送端之間互相重疊時，也能正常傳送，因此，只要搭配 ANDW 指令，甚至還能夠將 m 位元移到 n 位元，也就是取代移位 (Shift) 指令。
- 可將傳送端和傳送目的端的資料區域指定為重疊。

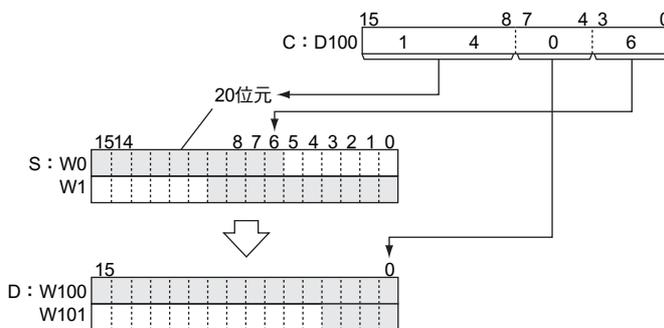
使用上的注意事項

- 傳送端、傳送目的端的資料範圍不得超過區域的最大範圍。
- 當傳送位元數 (C 的 n) 為 0 時，無法執行傳送動作。
- 除了已傳送的位元外，傳送目的端 CH 的內容將不會改變。

程式例



當 0.00 變成 ON 時，只要控制資料 (C) 出現下列內容，就會將 W0 的位元 1 ~ 上位端的 40 位元，傳送到 W100 的位元 0 ~ 上位端的 20 位元。



XFER

指令名稱	指令記號	指令的各種組合	Fun No.	功能
區塊傳送	XFER	@XFER	070	可同時傳送多個連續的 CH 資料。

符號	XFER	
		<p>W : 傳送的CH數</p> <p>S : 送元下位CH番</p> <p>D : 送先下位CH番</p>

是否適用於特定區域

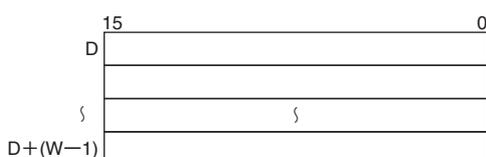
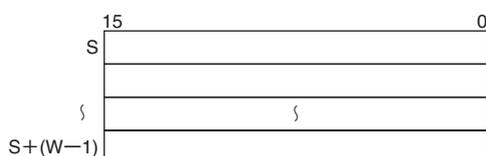
區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
W	傳送的 CH 數	UINT	1
S	傳送端下位 CH 編號	WORD	可調整
D	傳送目的端下位 CH 編號	WORD	可調整

W : 傳送的 CH 數

10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF



● 運算元種類

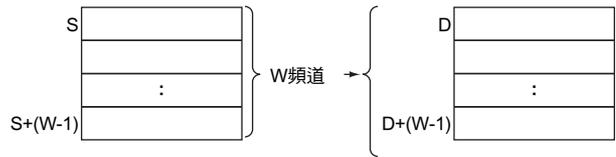
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
W	○	○	○	○	○	○	○	○	○	○	-	-	-
S,D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF

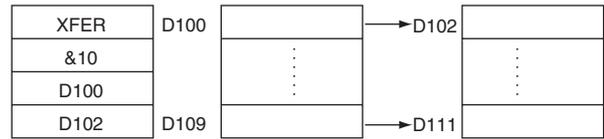
功能

將 S 所指定的傳送端下位 CH 編號 ~ 相當於 W 所指定的資料數 (10 進位或 16 進位) 傳送到 D 所指定的傳送目的端下位 CH 編號以後的 CH。



提示

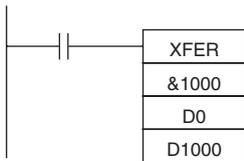
- 將 S 和 D 指定為相同的運算元種類，亦可使用 XFER 指令來移動 (Shift) 資料。



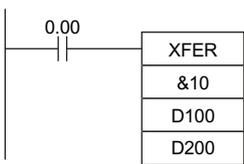
- 可將傳送端和傳送目的端的資料區域指定為重疊 (字元移位的動作)。

使用上的注意事項

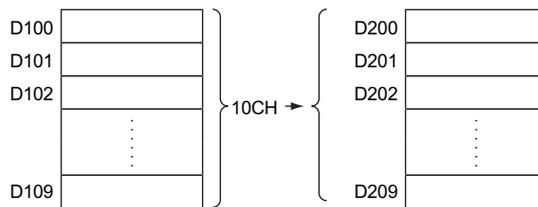
- 傳送端、傳送目的端的 CH 不得超過資料區。
- 利用區塊傳送方式來傳送多個 CH 時，將增加指令執行的時間，即使在本指令執行時發生配置訊號，配置任務仍然會在本指令執行完成後才開始啟動。
另外，一旦在本指令執行時發生斷電，就會中途停止區塊傳送動作。



程式例



當0.00變成ON時，就會將D100 ~ D109的10CH傳送到D200 ~ D209。



BSET

指令名稱	指令記號	指令的各種組合	Fun No.	功能
區塊設定	BSET	@BSET	071	將多個連續的 CH 皆設定為相同的資料。

符號	BSET	
		<p>S：傳送資料</p> <p>D1：傳送目的端下位CH編</p> <p>D2：傳送目的端上位CH編</p>

是否適用於特定區域

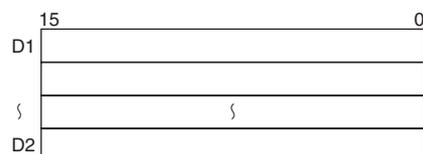
區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
S	傳送資料	WORD	1
D1	傳送目的端下位 CH 編號	WORD	可調整
D2	傳送目的端上位 CH 編號	WORD	可調整

D1：傳送目的端下位 CH 編號

D2：傳送目的端上位 CH 編號



(註) D1 ~ D2 的運算元種類必須相同。

● 運算元種類

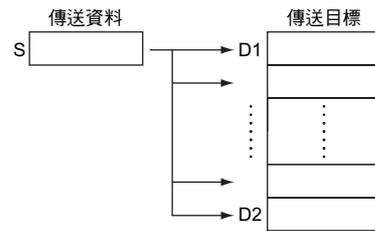
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	-	-	-
D1,D2	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 D1 > D2 時，本旗標 ON 否則為 OFF

功能

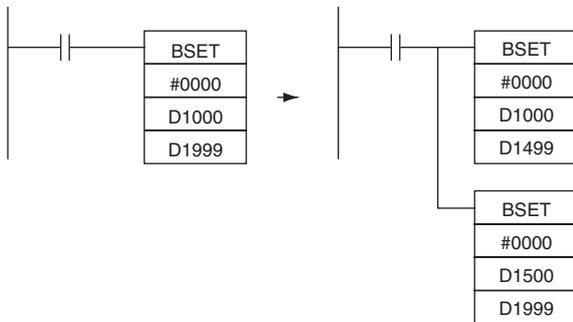
將 S 輸出至 D1 所指定的傳送目的端下位 CH 編號 ~ D2 所指定的傳送目的端上位 CH 編號。



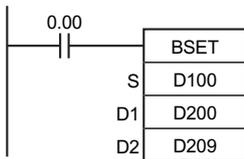
使用上的注意事項

利用區塊設定方式來傳送多個 CH 時，將增加指令執行的時間，即使在本指令執行時發生配置訊號，配置任務仍然會在本指令執行完成後才開始啟動。另外，一旦在本指令執行時發生斷電，就會中途停止區塊傳送動作。只要分割指令，即可避免上述問題發生。

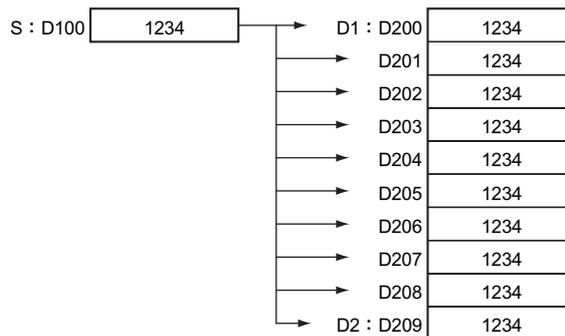
例)



程式例

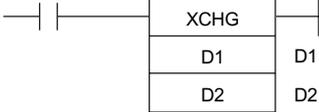


當0.00變成ON時，就會將D100的內容傳送至D200 ~ D209的10CH。



XCHG

指令名稱	指令記號	指令的各種組合	Fun No.	功能
資料交換	XCHG	@XCHG	073	執行 CH 間的資料交換。

符號	XCHG	
		D1

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D1	交換的 CH 編號 1	WORD	1
D2	交換的 CH 編號 2	WORD	1

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D1,D2	○	○	○	○	○	○	○	○	○	-	-	-	-

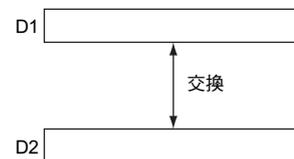
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	無任何變化
= 旗標	P_EQ	無任何變化
負旗標	P_N	無任何變化

功能

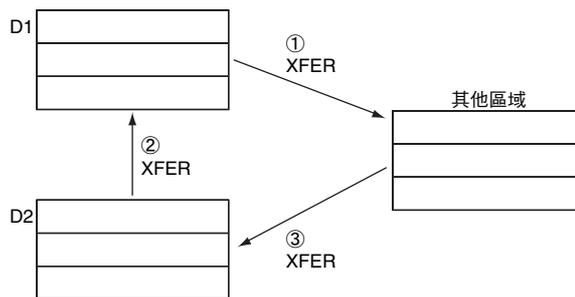
● XCHG

交換 D1 和 D2 的內容。

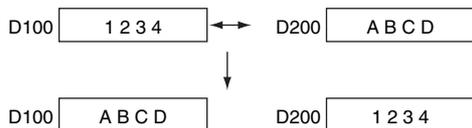
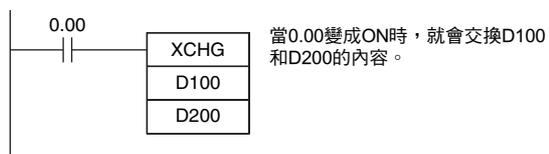


提示

同時交換 3CH 以上的 CH 資料時，請依照下圖所示，先透過其他區域，並使用 XFER (區塊傳送) 指令進行資料交換。



程式例



DIST

指令名稱	指令記號	指令的各種組合	Fun No.	功能
資料配置	DIST	@DIST	080	以傳送資料作為傳送目的端的基準，然後再將資料傳送到位移 (Offset) 後的 CH。

符號	DIST	
		<p>S1: 傳送資料</p> <p>D: 傳送目的端基準CH編</p> <p>S2: 位移(Offset)資料</p>

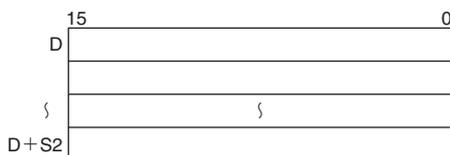
是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
S1	傳送資料	WORD	1
D	傳送目的端基準 CH 編號	WORD	1
S2	位移 (Offset) 資料	UINT	1

D：傳送目的端基準 CH 編號



S2：位移 (Offset) 資料

10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

(註) D ~ D+S2 的運算元種類必須相同。

● 運算元種類

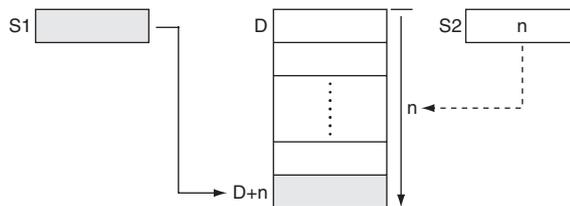
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1										○			
D	○	○	○	○	○	○	○	○	○	—			
S2										○			

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當傳送資料變成 0 時，本旗標 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 當傳送資料的最上位元變成 1 時，本旗標 ON 否則為 OFF

功能

將 S1 傳送到 D 所指定的傳送目的端基準 CH 編號 ~ 相當 S1 → D+S2
於 S2 所指定的位移資料移位後的位址。



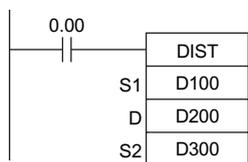
提示

- 改變位移資料 (S2) 的內容後，即可利用 DIST 指令，將資料傳送 (配置) 到任意位置。

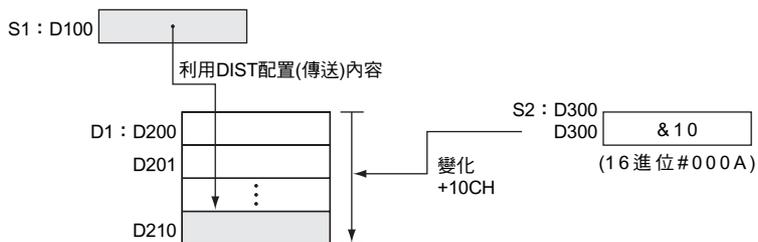
使用上的注意事項

- 位移資料 (S2) 的內容不得超過傳送目的端的區域範圍。

程式例



當 0.00 變成 ON 時，就會將 D100 的內容傳送到 D300 的內容 (例如：&10 (16 進位表示時為 #000A)) 加上 D200 後所得到的位址 (D210) (改變 D300 的內容後，即可將 D100 的內容配置到任何位址)。



COLL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
資料擷取	COLL	@COLL	081	以傳送端為基準，將位移後的 CH 內容傳送到指定的 CH。

符號	COLL	
		<p>S1: 傳送端基準CH編號</p> <p>S2: 位移(Offset)資料</p> <p>D: 傳送目的端的CH編號</p>

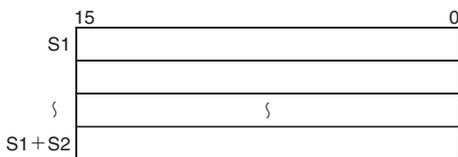
是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
S1	傳送端基準 CH 編號	WORD	1
S2	位移 (Offset) 資料	WORD	1
D	傳送目的端的 CH 編號	WORD	1

S1：傳送端基準 CH 編號



S2：位移 (Offset) 資料

10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF

(註) D ~ D+S 的運算元種類必須相同。

● 運算元種類

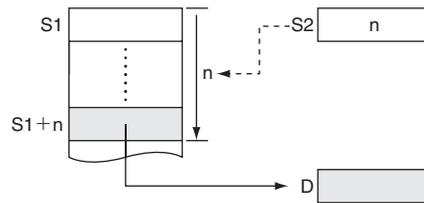
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1										—			
S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D										—			

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當傳送資料變成 0 時，本旗標 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 當傳送資料的最上位元變成 1 時，本旗標 ON 否則為 OFF

功能

將 S1 所指定的傳送端基準 CH ~ 相當於 S2 所指定的位移資料移位後的位址資料傳送至 D。



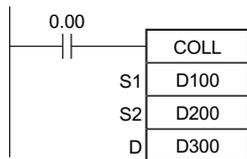
提示

- 改變位移資料 (S2) 的內容後，即可利用 COLL 指令，在任意位置擷取 (取出) 資料。

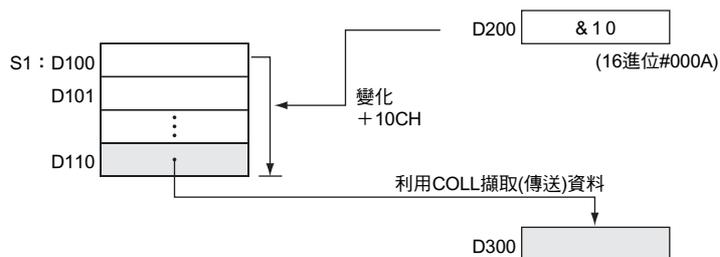
使用上的注意事項

- 位移資料 (S2) 的內容不得超過傳送端的區域範圍。

程式例



當0.00變成ON時，就會將D200的內容(例如：&10 (16進位表示時為#000A))加上D100後所得到的位址(D110)內容傳送到D300 (改變D200的內容後，即可由任意位址擷取資料)



資料移位指令

SFT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
移位暫存器	SFT	—	010	可執行移位暫存器的動作。



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D1	移位下位 CH 編號	UINT	可調整
D2	移上位 CH 編號	UINT	可調整

● 運算元種類

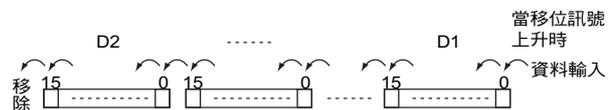
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D1,D2	○	○	○	—	—	—	—	—	—	—	—	—	—

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF

功能

當移位訊號輸入上升 (OFF → ON) 時，會將 D1 ~ D2 全部向左 (最下位元 → 最上位元) 移動 1 位元，然後再將資料輸入的 ON/OFF 內容反映至最下位元。



使用上的注意事項

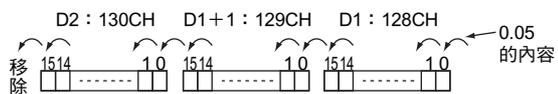
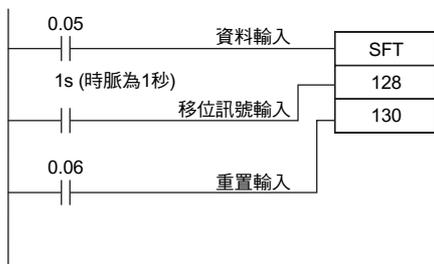
- 請勿使用多個 D1、D2 範圍重疊的 SFT 指令。(一旦使用將造成結果不穩定。)
- D1、D2 的運算元種類必須相同。
- 由移位範圍溢出的位元內容就會被移除。
- 當復歸信號 ON 時，就會復歸 (= 0) D1 所指定的移位下位 CH 編號 ~ D2 所指定的移上位 CH 編號，復歸信號在順序上較其他輸入優先。
- 設定移位範圍時，基本上以 $D1 \leq D2$ 為準，不過，即使指定為 $D1 > D2$ ，也只會將 D1 移動 1 個 CH (字元)，而不會造成錯誤發生。

程式例

● 超過 16 位元的移位暫存器範例

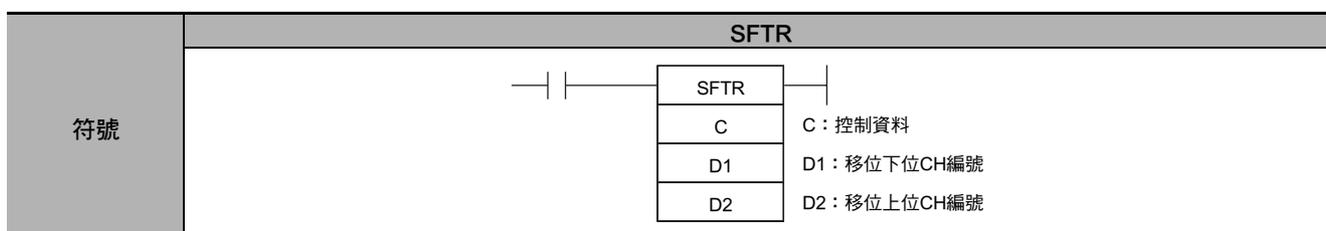
此種移位暫存器使用 128 ~ 130CH 其中的 48 個位元。

以時鐘脈衝 1s 輸入移位訊號時，每隔 1 秒輸入繼電器 0.05 的內容就會被移位至 128.00 ~ 130.15。



SFTR

指令名稱	指令記號	指令的各種組合	Fun No.	功能
左右移位暫存器	SFTR	@SFTR	084	執行可切換移位方向的移位暫存器動作。



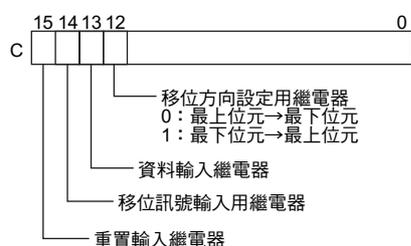
是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
C	控制資料	UINT	1
D1	移位下位 CH 編號	UINT	可調整
D2	移至上位 CH 編號	UINT	可調整

C：控制資料



(註) D1、D2 的運算元種類必須相同。

● 運算元種類

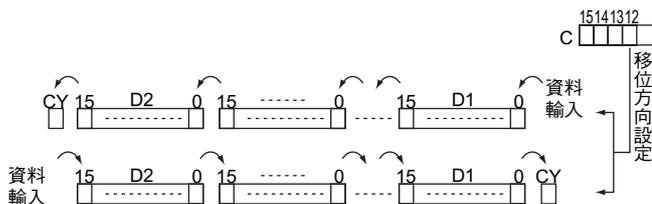
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C,D1,D2	○	○	○	○	○	○	○	○	○	—	—	—	—

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 D1 > D2 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當 1 被移位至進位 (CY) 旗標時，本旗標就會 ON 當 0 被移位至進位 (CY) 旗標時，本旗標就會 OFF 當復歸信號變成 1 時，本旗標 OFF

功能

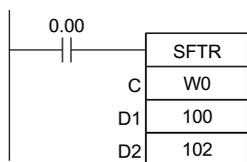
當移位訊號輸入用繼電器 (C 的位元 14) ON 時，就會將 D1 ~ D2 朝移位方向設定用繼電器 (C 的位元 12) 所指定的方向移動 1 位元，然後再將資料輸入用繼電器 (C 的位元 13) 的 ON/OFF 內容反映至最下位元或最上位元。此時，溢出移位範圍的位元內容將會被反映至進位旗標 (CY)。



(註)

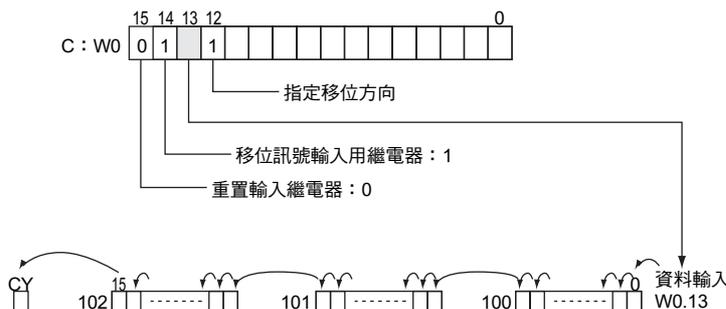
- 上述移位動作為復歸信號繼電器 (C 的位元 15) OFF 時的條件。
- 當復歸信號繼電器 (C 的位元 15) ON 時，D1 ~ D2 全部會被復歸 (= 0)。

程式例



• 移位動作

在0.00變成ON的狀態下，只要一位訊號輸入用繼電器W0.14 ON，就會將100 ~ 102CH共3CH朝W0.12所指定的方向 (例1：最下位元→最上位元的方向)移動1位元，然後在將資料輸入計電器W0.13的內容設定至最下位元100.00中。溢出的102.15內容會被設定到CY (進位旗標)中。

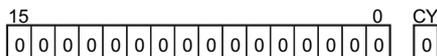


• 重置動作

重置輸入繼電器W0.15 ON時
在0.00 ON的狀態下，只要W0.14變成ON，就會將100 ~ 102CH等共3CH以及CY旗標全部重置為OFF。

● 控制資料的內容及動作

(1)重置時...當重置輸入繼電器(C的位元15) ON時



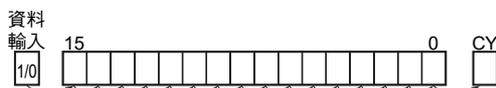
D1 ~ D2CH所有位元的內容及CY旗標將變成0，此時無法再接受其他的輸入。

(2)向左移位時(最上位元←最下位元)...當移位方向設定用繼電器(C的位元12) ON時



當移位訊號輸入用繼電器(C的位元14) ON時，就會根據每個位元的內容，將資料輸入繼電器(C的位元13)的內容(1/0)，依位元別分別朝上位元端移位至D1CH的位元0，而D2CH的位元15內容則會被移位至CY旗標。

(3)向右移位時(最上位元→最下位元)...當移位方向設定用繼電器(C的位元12) OFF時



當移位訊號輸入用繼電器(C的位元14) ON時，就會根據每個位元的內容，將資料輸入繼電器(C的位元13)的內容(1/0)，依位元別分別朝下位元端移位至D2CH的位元15，而D1CH的位元0內容則會被移位至CY旗標。

WSFT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
字元移位	WSFT	@WSFT	016	表示以 CH 為單位的移位動作。

符號	WSFT								
		<table border="1"> <tr> <td>WSFT</td> <td>S：移位資料編號</td> </tr> <tr> <td>S</td> <td>D1：移下位CH編號</td> </tr> <tr> <td>D1</td> <td>D2：移上位CH編號</td> </tr> <tr> <td>D2</td> <td></td> </tr> </table>	WSFT	S：移位資料編號	S	D1：移下位CH編號	D1	D2：移上位CH編號	D2
WSFT	S：移位資料編號								
S	D1：移下位CH編號								
D1	D2：移上位CH編號								
D2									

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
S	移位資料編號	WORD	1
D1	移下位 CH 編號	UINT	可調整
D2	移上位 CH 編號	UINT	可調整

● 運算元種類

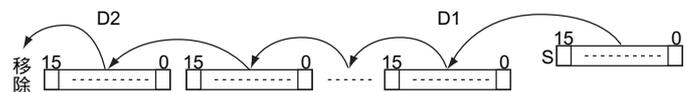
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	-	-	-
D1,D2	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 D1 > D2 時，本旗標 ON 否則為 OFF

功能

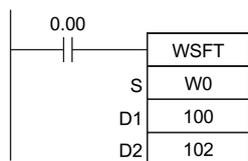
依字元別將 D1 ~ D2 移位至上位 CH，然後再將 S 所指定的資料輸出至最下位 CH (D1)。此時，原先的最上位 CH (D2) 資料就會被移除。



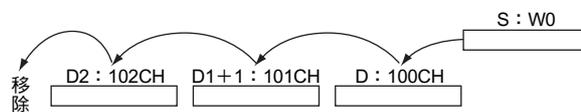
使用上的注意事項

- D1、D2 的運算元種類必須相同。
- 移位的資料較多時，所需的指令執行時間也較長。因此，一旦在本指令執行時發生斷電，就會中途停止移位的執行動作，此點需特別注意。

程式例

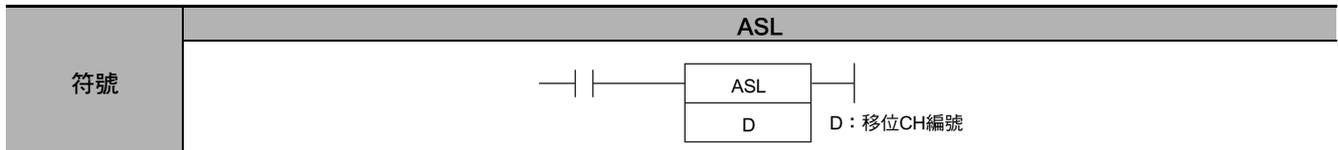


當0.00變成ON時，就會依字元別(CH)將100 ~ 102CH移位至上位端。
100CH中的W0內容會被儲存起來，而102CH的內容則會被移除。



ASL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
向左移動 1 位元	ASL	@ASL	025	將 CH 資料向左移動 1 位元。



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D	移位 CH 編號	UINT	1

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	-	-	-	-

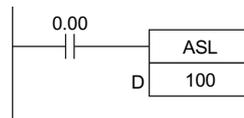
條件旗標的動作

名稱	標籤	內容
異常旗標 = 旗標	P_ER P_EQ	OFF • 當移位結果為 0 時，本旗標 ON • 否則為 OFF
進位旗標 (Carry flag)	P_CY	• 當 1 被移位至進位 (CY) 旗標時，本旗標就會 ON • 否則為 OFF
負旗標	P_N	• 根據移位結果，最上位元變成 1 時，本旗標就會 ON • 否則為 OFF

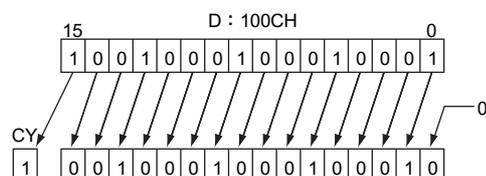
功能

● ASL

將 D 向左 (最下位元→最上位元) 移動 1 位元，而 0 會被設定在最下位元中。最上位元會被移位至進位旗標 (CY) 中。

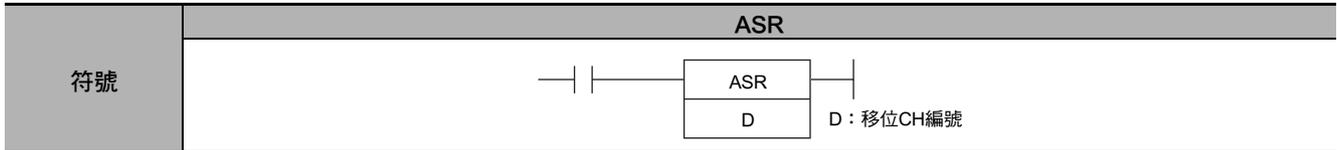


當 0.00 變成 ON 時，就會將 100CH 向左移動 1 位元。此時 0 會被設定至 100.00，而 100.15 的內容也會被設定在 CY 旗標中。



ASR

指令名稱	指令記號	指令的各種組合	Fun No.	功能
向右移動 1 位元	ASR	@ASR	026	將 CH 資料向右移動 1 位元。



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D	移位 CH 編號	UINT	1

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	-	-	-	-

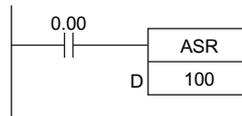
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當移位結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當 1 被移位至進位 (CY) 旗標時，本旗標就會 ON 否則為 OFF
負旗標	P_N	OFF

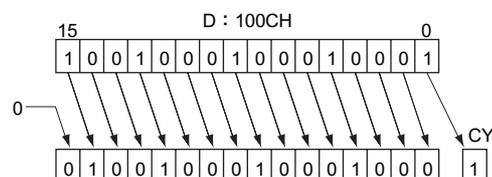
功能

● ASR

將 D 向右 (最上位元→最下位元) 移動 1 位元，而 0 會被設定在最上位元中。最下位元會被移位至進位旗標 (CY) 中。

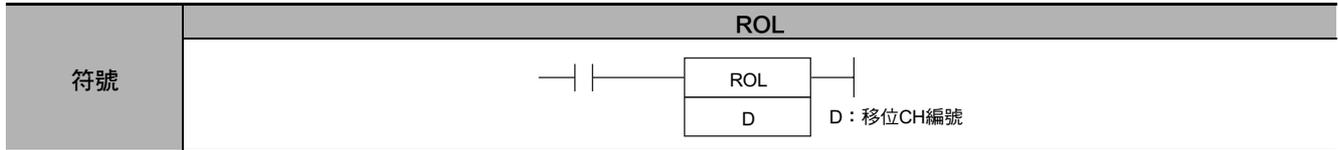


當 0.00 變成 ON 時，就會將 100CH 向右移動 1 位元。此時 0 會被設定至 100.00，而 100.15 的內容也會被設定在 CY 旗標中。



ROL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
附 CY 的 1 位元向左回轉	ROL	@ROL	027	將 16 位元 CH 資料及進位 (CY) 旗標等向左回轉 1 位元。



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D	移位 CH 編號	UINT	1

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	-	-	-	-

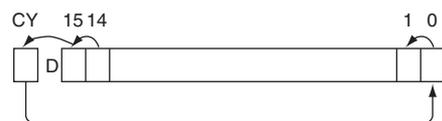
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當移位結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當 1 被移位至進位 (CY) 旗標時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據移位結果，最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

● ROL

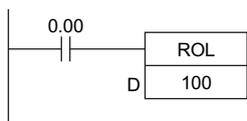
將 D 及進位旗標 (CY) 等向左 (最下位元) → 最上位元) 向左回轉 1 位元。



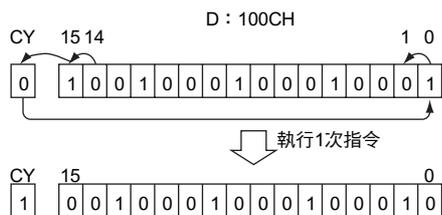
提示

- 將設定進位 (STC)、清除進位 (CLC) 指令用在本指令前，即可將進位 (CY) 旗標的內容設定為 1、0。

程式例

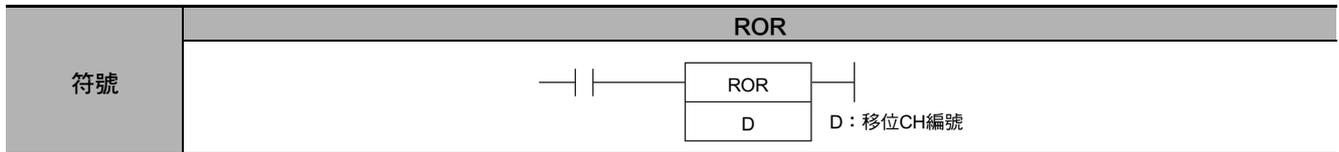


當0.00變成ON時，就會100CH及CY旗標等向左回轉1位元。
此時100.15的內容會被設定在CY旗標中，而CY旗標的內容則會被設定至100.00。



ROR

指令名稱	指令記號	指令的各種組合	Fun No.	功能
附 CY 的 1 位元向右回轉	ROR	@ROR	028	將 16 位元 CH 資料及進位 (CY) 旗標等向右回轉 1 位元。



是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D	移位 CH 編號	UINT	1

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	—	—	—	—

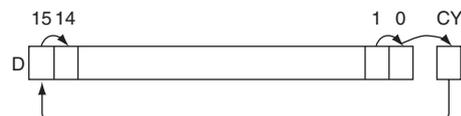
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當移位結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當 1 被移位至進位 (CY) 旗標時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據移位結果，最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

● ROR

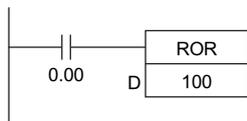
將 D 及進位旗標 (CY) 等向右 (最上位元) → 最下位元) 向右回轉 1 位元。



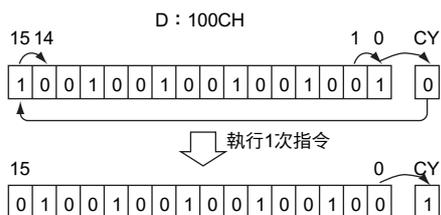
提示

- 將設定進位 (STC)、清除進位 (CLC) 指令用在本指令前，即可將進位 (CY) 旗標的內容設定為 1、0。

程式例

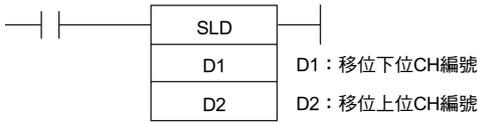
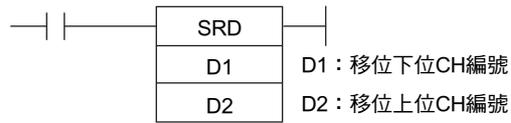


當0.00變成ON時，就會100CH及CY旗標等向右回轉1位元。
此時100.00的內容會被設定在CY旗標中，而CY旗標的內容則會被設定至100.15。



SLD/SRD

指令名稱	指令記號	指令的各種組合	Fun No.	功能
向左移動 1 位數	SLD	@SLD	074	以 1 位數 (4 位元) 為單位，將連續的 CH 資料內容向左移位。
向右移動 1 位數	SRD	@SRD	075	以 1 位數 (4 位元) 為單位，將連續的 CH 資料內容向右移位。

符號	SLD	SRD
	 <p>D1: 移位下位CH編號 D2: 移上位CH編號</p>	 <p>D1: 移位下位CH編號 D2: 移上位CH編號</p>

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
D1	移位下位 CH 編號	UINT	可調整
D2	移上位 CH 編號	UINT	可調整

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D1,D2	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 D1 > D2 時，本旗標 ON 否則為 OFF

功能

● SLD

以位數 (4 位元) 為單位，將 D1 ~ D2 的範圍移動至上位位數端。此時，最下位位數 (D1 的位元 3-0) 會被加上 0，原先最上位位數 (D2 的位元 15-12) 的資料將會被移除。

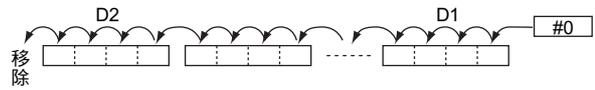
● SRD

以位數 (4 位元) 為單位，將 D1 ~ D2 的範圍移動至下位位數端。此時，最上位位數 (D2 的位元 15-12) 會被加上 0，原先最下位位數 (D1 的位元 3-0) 的資料將會被移除。

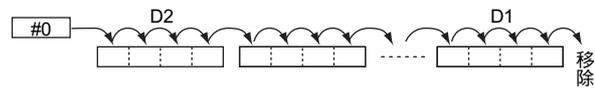
使用上的注意事項

- D1、D2 的運算元種類必須相同。
- 移位的資料較多時，所需的指令執行時間也較長。因此，一旦在本指令執行時發生斷電，就會中途停止移位的執行動作，此點需特別注意。

• SLD

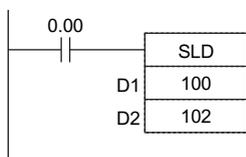


• SRD

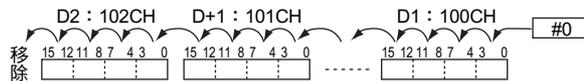


程式例

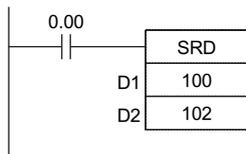
• SLD



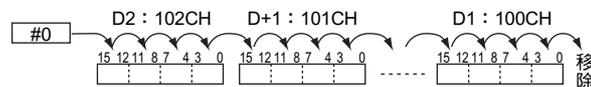
當0.00變成ON時，就會將100 ~ 102的每個位數(4位元)朝上位端移動。
當#0被設定在100CH的位元0 ~ 3時，102CH的位元12 ~ 15就會被移除。



• SRD



當0.00變成ON時，就會將100 ~ 102的每個位數(4位元)朝下位端移動。
當#0被設定在102CH的位元12 ~ 15時，100CH的位元0 ~ 3就會被移除。



NASL/NSLL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
向左移動 N 位元	NASL	@NASL	580	依指定的位元數，將 16 位元的 CH 資料向左移動。
向左倍精確度移動 N 位元	NSLL	@NSLL	582	依指定的位元數，將 32 位元的 CH 資料向左移動。

符號	NASL	NSLL
	<p>D：移位CH編號 C：控制資料</p>	<p>D：移位下位CH編號 C：控制資料</p>

是否適用於特定區域

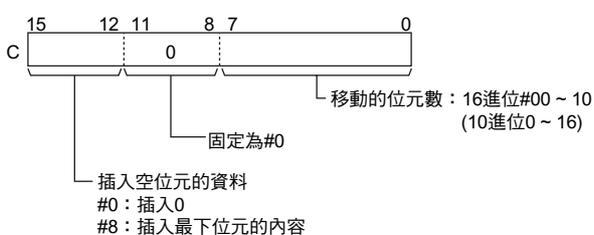
區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

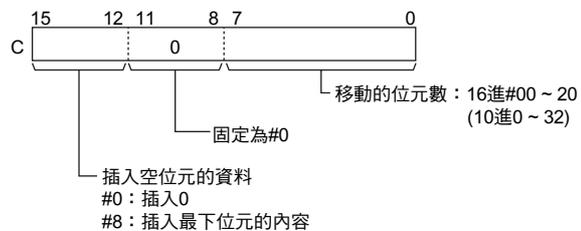
運算元種類	內容	資料型態		容量	
		NASL	NSLL	NASL	NSLL
D	NASL：移位 CH 編號 NSLL：移位下位 CH 編號	UINT	UDINT	1	2
C	控制資料	UINT	UINT	1	1

C：控制資料

• NASL



• NSLL



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	—	—	—	—
C	○	○	○	○	○	○	○	○	○	○	—	—	—

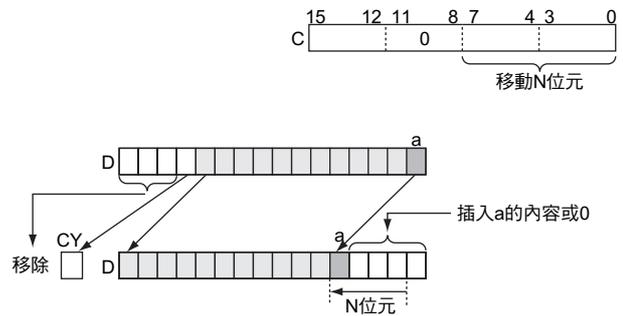
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當控制資料 C (移動的位元數) 超出範圍時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當移位結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當 1 被移位至進位 (CY) 旗標時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據移位結果，最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

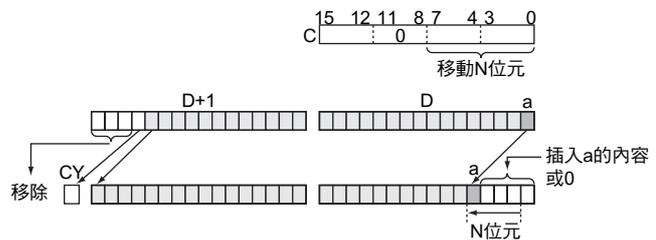
● NASL

依 C 所指定的位元數 (BIN)，將 D 向左移動 (最下位元→最上位元)。此時，您所指定的 CH 的最下位元～相當於移動位元數的位元就會被插入符合空位元的插入資料中。



● NSLL

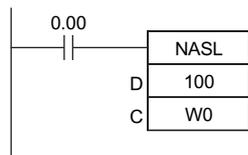
以 D 作為倍精確度資料，將 C 所指定的位元數 (BIN) 全部向左移動 (最下位元→最上位元)。此時，您所指定的 CH 的最下位元～相當於移動位元數的位元就會被插入符合空位元的插入資料中。



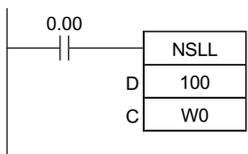
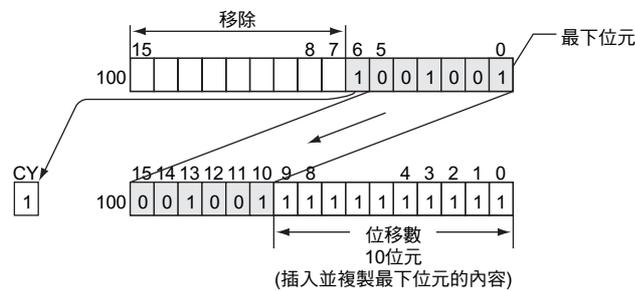
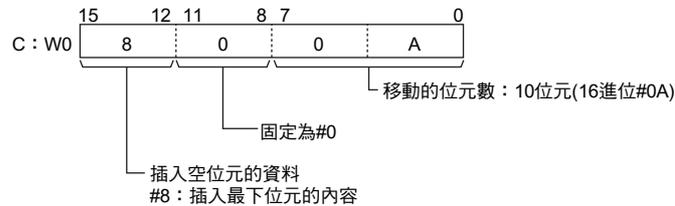
使用上的注意事項

- 當位元溢出於指定的 CH 時，最後的位元內容會被移位至進位旗標 (CY)，除此之外皆會被移除。
- 當移動位元數 (C) 為 0 時，將不會執行移位動作，此時，將依照指定 CH 的資料，將每個旗標 ON/OFF。
-

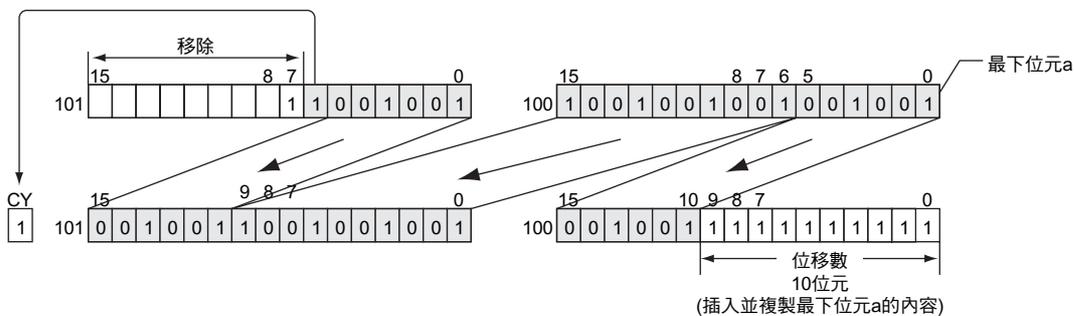
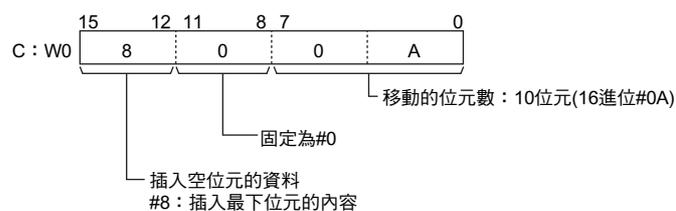
程式例



當 0.00 變成 ON 時，就會將 100CH 的資料向左移動 (最下位元→最上位元) 相當於 W0 (控制資料) 位元 0~7 所指定的位元數 (例：10 位元)。
100CH 位元 0 的內容會被插入並複製到空位元中。
當位元溢出時，最下位元會被設定至 CY，其他的位元則會被移除。



當 0.00 變成 ON 時，就會將 100CH ~ 101CH 的 2CH 資料向左移動 (最下位元→最上位元) 相當於 W0 (控制資料) 位元 0~7 所指定的位元數 (例：10 位元)。
100CH 位元 0 的內容會被插入並複製到空位元中。
當位元溢出時，最下位元會被設定至 CY，其他的位元則會被移除。



NASR/NSRL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
向右移動 N 位元	NASR	@NASR	581	依指定的位元數，將 16 位元的 CH 資料向右移動。
向左倍精確度移動 N 位元	NSRL	@NSRL	583	依指定的位元數，將 32 位元的 CH 資料向右移動。

符號	NASR	NSRL

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

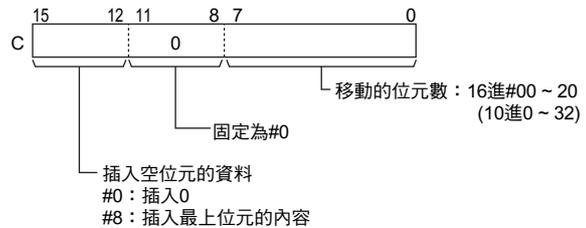
運算元種類	內容	資料型態		容量	
		NASR	NSRL	NASR	NSRL
D	NASR: 移位 CH 編號 NSRL: 移位下位 CH 編號	UINT	UDINT	1	2
C	控制資料	UINT	UINT	1	1

C: 控制資料

• NASR



• NSRL



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	—	—	—	—
C	○	○	○	○	○	○	○	○	○	○	—	—	—

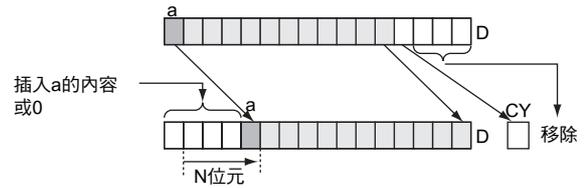
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當控制資料 C (移動的位元數) 超出範圍時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當移位結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當 1 被移位至進位 (CY) 旗標時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據移位結果，最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

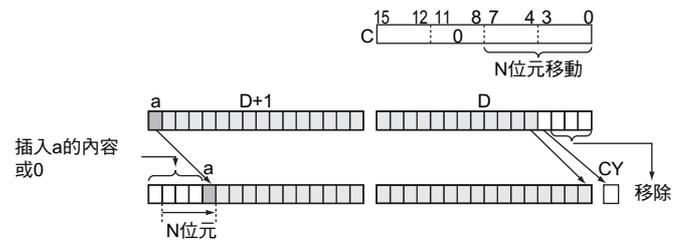
● NASR

依 C 所指定的位元數 (BIN)，將 D 全部向右移動 (最上位元→最下位元)。此時，您所指定的 CH 的最上位元～相當於移動位元數的位元就會被插入符合空位元的插入資料中。



● NSRL

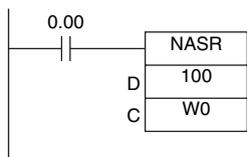
以 D 作為倍精確度資料，將 C 所指定的位元數 (BIN) 全部向右移動 (最上位元→最下位元)。此時，您所指定的 CH 的最上位元～相當於移動位元數的位元就會被插入符合空位元的插入資料中。



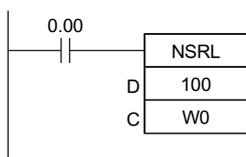
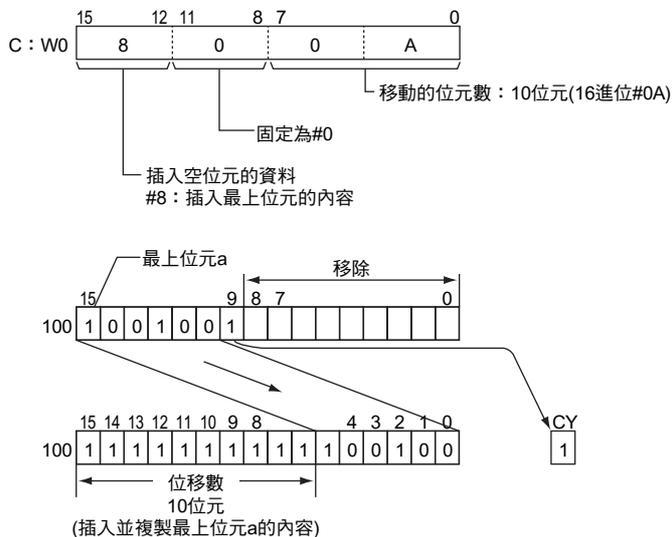
使用上的注意事項

- 當位元溢出於指定的 CH 時，最後的位元內容會被移位至進位旗標 (CY)，除此之外皆會被移除。
- 當移動位元數 (C) 為 0 時，將不會執行移位動作，此時，將依照指定 CH 的資料，將每個旗標 ON/OFF。

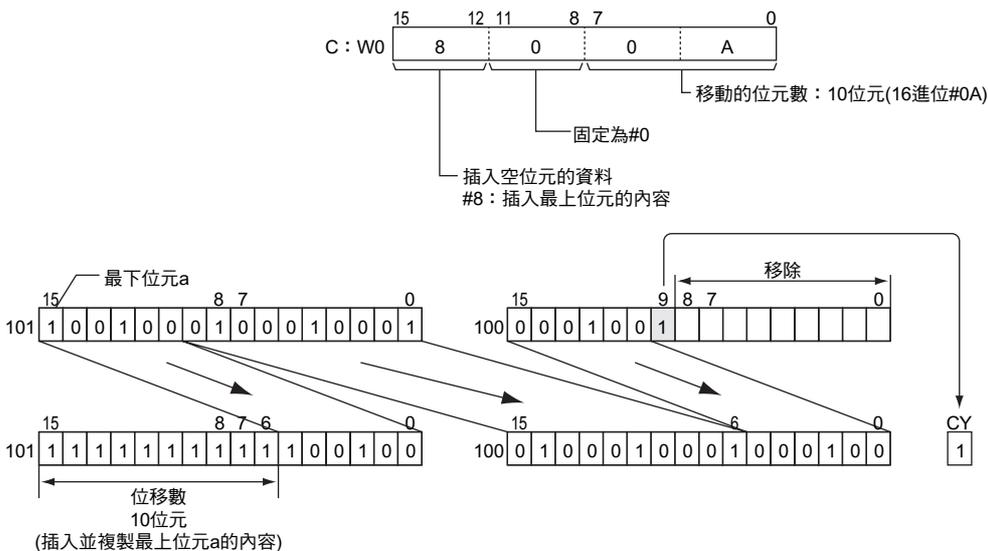
程式例



當 0.00 變成 ON 時，就會將 100CH 的資料向右移動 (最上位元→最下位元) 相當於 W0 (控制資料) 位元 0 ~ 7 所指定的位元數 (例：10 位元)。100CH 位元 15 的內容會被插入並複製到空位元中。當位元溢出時，最後的位元會被設定至 CY，其他的位元則會被移除。



當 0.00 變成 ON 時，就會將 100CH ~ 101CH 的 2CH 資料向右移動 (最上位元→最下位元) 相當於 W0 (控制資料) 位元 0 ~ 7 所指定的位元數 (例：10 位元)。101CH 位元 15 的內容會被插入空位元中。當位元溢出時，最後的位元會被設定至 CY，其他的位元則會被移除。



加 / 減指令

++/++L

指令名稱	指令記號	指令的各種組合	Fun No.	功能
BIN 加法運算	++	@ ++	590	將 16 進位 4 位數的 1CH 資料加 1。
BIN 倍精確度遞增	++L	@ ++L	591	將 16 進位 8 位數的 2CH 資料加 1。

Ω □	++	++L

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		++	++L	++	++L
D	++: 資料的 CH 編號 ++L: 控制資料的下位 CH 編號	UINT	UDINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標 = 旗標	P_ER	OFF
	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據演算結果，當最上位元為 1 時，本旗標 ON 否則為 OFF

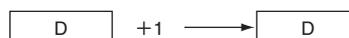
功能

● ++

為 D 所指定的資料進行 BIN 演算 (+1)。

若使用 ++ 時，只要輸入條件 ON (變成 OFF 前)，就會在每個週期加 1。

若使用 @++ 時，只會在輸入條件上升 (僅限於 1 週期) 時才會加 1。

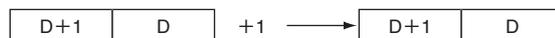


● **++L**

將 D 所指定資料當作倍精確度資料，並且進行 BIN 演算 (+1)。

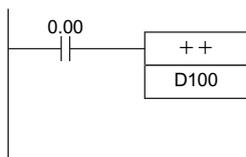
若使用 ++L 時，只要輸入條件 ON (變成 OFF 前)，就會在每個週期加 1。

若使用 @++L 時，只會在輸入條件上升 (僅限於 1 週期) 時才會加 1。

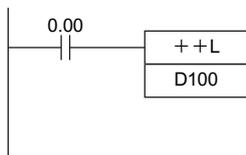


程式例

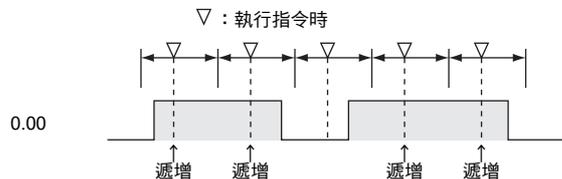
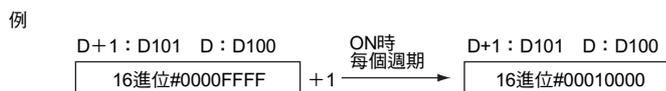
● 使用 ++、++L 指令時



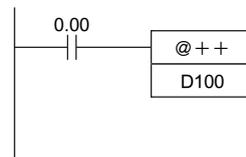
當 0.00 變成 ON 時，每個週期都會將 D100 的資料內容加 1 (每個週期都會加 1 直到 0.00 變成 OFF 為止)。



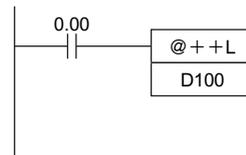
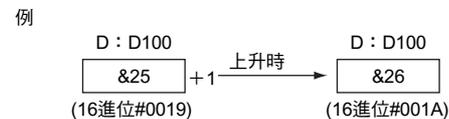
當 0.00 變成 ON 時，每個週期都會將 D101、D100 的資料內容加 1 (每個週期都會加 1 直到 0.00 變成 OFF 為止)。



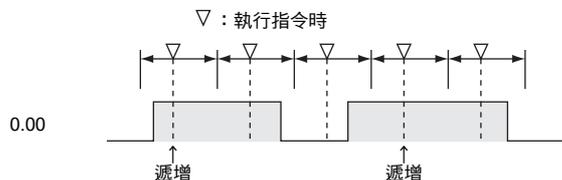
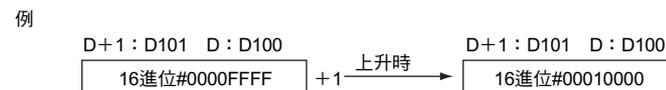
● 使用 @++、@++L 指令時



只有在 0.00 的訊號由 OFF 上升為 ON 時，D100 的資料內容才會被加 1。



只有在 0.00 的訊號由 OFF 上升為 ON 時，D101、D100 的資料內容才會被加 1。



/ L

指令名稱	指令記號	指令的各種組合	Fun No.	功能
BIN 減法運算	--	@ --	592	將 16 進位 4 位數的 1CH 資料減 1。
BIN 倍精確度遞減	-- L	@ -- L	593	將 16 進位 8 位數的 2CH 資料減 1。

Ω □	□ □		□ □ Λ	

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		--	-- L	--	-- L
D	-- : 資料的 CH 編號 -- L : 控制資料的下位 CH 編號	UINT	UDINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

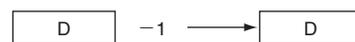
名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據演算結果，當最上位元為 1 時，本旗標 ON 否則為 OFF

功能

● --

為 D 所指定的資料進行 BIN 演算 (-1)。

使用 -- 時，只要輸入條件 ON (變成 OFF 前)，就會在每個週期減 1。若使用 @ -- 時，只會在輸入條件上升 (僅限於 1 週期) 時才會減 1。

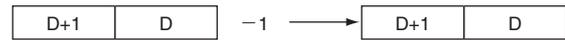


● --- L

將 D 所指定資料當作倍精確度資料，並且進行 BIN 演算 (-1)。

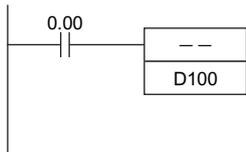
使用 --- L 時，只要輸入條件 ON (變成 OFF 前)，就會在每個週期減 1。

若使用 @ --- L 時，只會在輸入條件上升 (僅限於 1 週期) 時才會減 1。

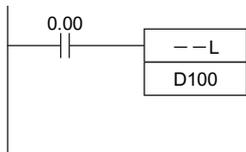
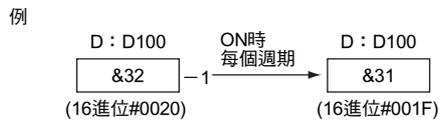


程式例

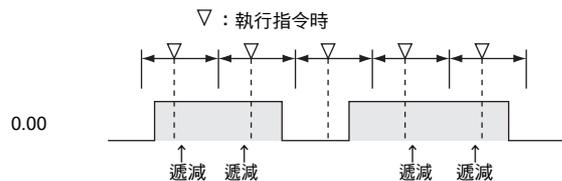
● 使用 ---、--- L 指令時



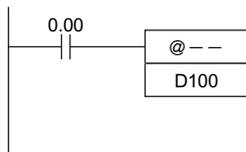
當 0.00 變成 ON 時，每個週期都會將 D100 的資料內容減 1 (每個週期都會減 1 直到 0.00 變成 OFF 為止)



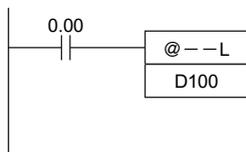
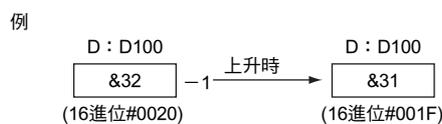
當 0.00 變成 ON 時，每個週期都會將 D101、D100 的資料內容減 1 (每個週期都會減 1 直到 0.00 變成 OFF 為止)



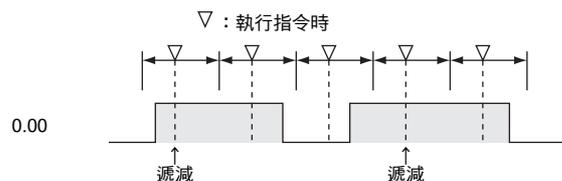
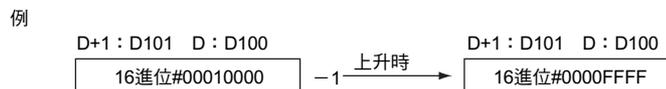
● 使用 @ ---、@ --- L 指令時



只有在 0.00 的訊號由 OFF 上升為 ON 時，D100 的資料內容從 1 開始減算。



只有在 0.00 的訊號由 OFF 上升為 ON 時，D101、D100 的資料內容從 1 開始減算。



++B/++BL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
BCD 遞增	++B	@++B	594	將 BCD4 位數的 1CH 資料加 1。
BCD 倍精確度遞增	++BL	@++BL	595	將 BCD8 位數的 2CH 資料加 1。

符號	++B	++BL

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		++B	++BL	++B	++BL
D	++B: 資料的 CH 編號 ++BL: 控制資料的下位 CH 編號	WORD	DWORD	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 D 的資料非 BCD 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF

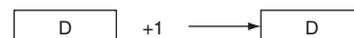
功能

● ++B

為 D 所指定的資料進行 BCD 演算 (+1)。

若使用 ++B 時，只要輸入條件 ON (變成 OFF 前)，就會在每個週期加 1。

若使用 @++B 時，只會在輸入條件上升 (僅限於 1 週期) 時才會加 1。



● ++BL

將 D 所指定資料當作倍精確度資料，並且進行 BCD 演算 (+1)。

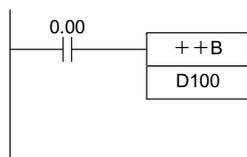
若使用 ++BL 時，只要輸入條件 ON (變成 OFF 前)，就會在每個週期加 1。

若使用 @++BL 時，只會在輸入條件上升 (僅限於 1 週期) 時才會加 1。



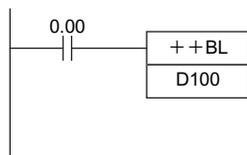
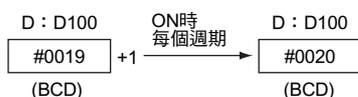
程式例

● 使用 ++B、++BL 指令時



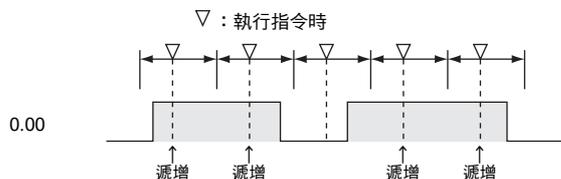
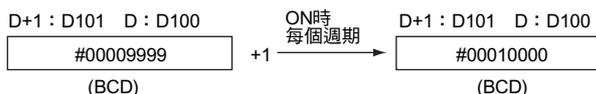
當 0.00 變成 ON 時，每個週期都會將 D100 的資料內容加 1 (每個週期都會加 1 直到 0.00 變成 OFF 為止)。

例

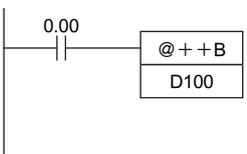


當 0.00 變成 ON 時，每個週期都會將 D101、D100 的資料內容加 1 (每個週期都會加 1 直到 0.00 變成 OFF 為止)。

例

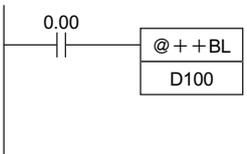
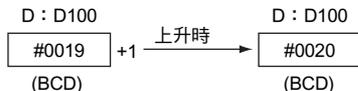


● 使用 @++B、@++BL 指令時



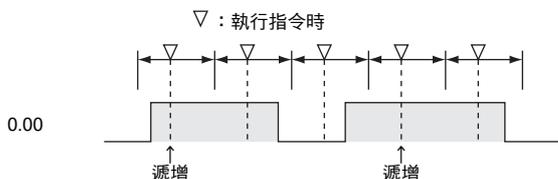
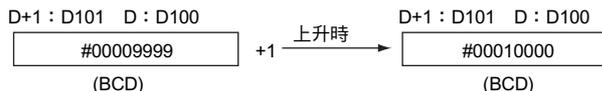
只有在 0.00 的訊號由 OFF 上升為 ON 時，D100 的資料內容才會被加 1。

例



只有在 0.00 的訊號由 OFF 上升為 ON 時，D101、D100 的資料內容才會被加 1。

例



— B / — BL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
BCD 遞減	— B	@ — B	596	將 BCD4 位數的 1CH 資料減 1。
BCD 倍精確度遞減	— BL	@ — BL	597	將 BCD8 位數的 2CH 資料減 1。

Ω □	— B	— BL

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		— B	— BL	— B	— BL
D	— B : 資料的 CH 編號 — BL : 資料的下位 CH 編號	WORD	DWORD	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	—	—	—	—

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 D 的資料非 BCD 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF

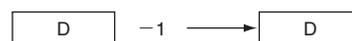
功能

● — B

為 D 所指定的資料進行 BCD 演算 (-1)。

使用 — B 時，只要輸入條件 ON (變成 OFF 前)，就會在每個週期減 1。

若使用 @ — B 時，只會在輸入條件上升 (僅限於 1 週期) 時才會減 1。

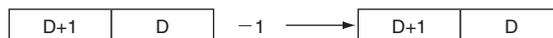


● **-- BL**

將 D 所指定資料當作倍精確度資料，並且進行 BCD 演算 (-1)。

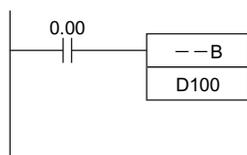
使用 -- BL 時，只要輸入條件 ON (變成 OFF 前)，就會在每個週期減 1。

若使用 @ -- BL 時，只會在輸入條件上升 (僅限於 1 週期) 時才會減 1。



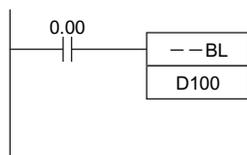
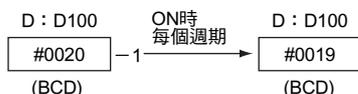
程式例

● **使用 -- B、-- BL 指令時**



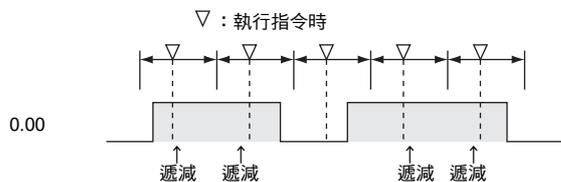
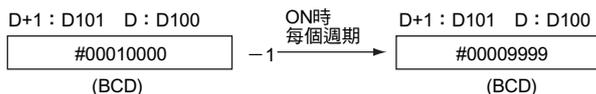
當 0.00 變成 ON 時，每個週期都會將 D100 的資料內容減 1 (每個週期都會減 1 直到 0.00 變成 OFF 為止)

例

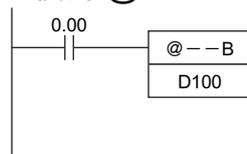


當 0.00 變成 ON 時，每個週期都會將 D101、D100 的資料內容減 1 (每個週期都會減 1 直到 0.00 變成 OFF 為止)

例

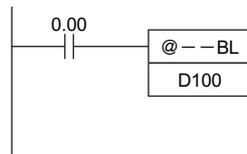
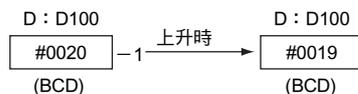


● **@ -- B、@ -- BL 指令時**



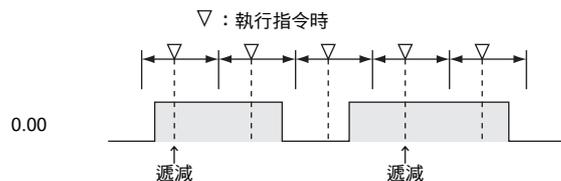
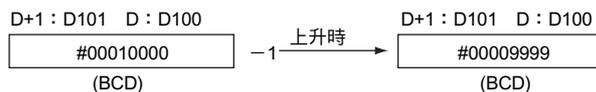
只有在 0.00 的訊號由 OFF 上升為 ON 時，D100 的資料內容從 1 開始減算。

例



只有在 0.00 的訊號由 OFF 上升為 ON 時，D101、D100 的資料內容從 1 開始減算。

例



四則運算指令

+ / + L

指令名稱	指令記號	指令的各種組合	Fun No.	功能
帶符號・無 CY 的 BIN 加法運算	+	@ +	400	利用帶符號的 16 進位 4 位數，將 CH 資料或常數相加。
帶符號・無 CY 的 BIN 倍精確度加法運算	+ L	@ + L	401	利用帶符號的 16 進位 8 位數，將 2CH 資料或常數相加。

Ω □	+			+ L		
		+	S1 S2 D	S1: 被加資料 S2: 加算資料 D: 演算結果的輸出CH編號		+L S1 S2 D

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		+	+ L	+	+ L
S1	+ : 被加資料 +L : 被加資料的下位 CH 編號	INT	DINT	1	2
S2	+ : 加算資料 +L : 加算資料的下位 CH 編號	INT	DINT	1	2
D	+ : 演算結果的輸出 CH 編號 +L : 演算結果的輸出下位 CH 編號	INT	DINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

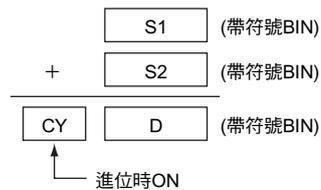
條件旗標的動作

名稱	標籤	內容	
		+	+ L
異常旗標	P_ER	OFF	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF
溢位旗標 (Overflow flag)	P_OF	<ul style="list-style-type: none"> 當正數 + 正數的結果在負數的範圍 (以 16 進位表示時為 #8000 ~ FFFF) 內時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當正數 + 正數的結果在負數的範圍 (以 16 進位表示時為 #80000000 ~ FFFFFFFF) 內時，本旗標就會 ON 否則為 OFF
欠位旗標 (Underflow flag)	P_UF	<ul style="list-style-type: none"> 當負數 + 負數的結果在正數的範圍 (以 16 進位表示時為 #0000 ~ 7FFF) 內時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當負數 + 負數的結果在正數的範圍 (以 16 進位表示時為 #00000000 ~ 7FFFFFFF) 內時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據演算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 根據演算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

● +

為 S1 所指定的資料及 S2 所指定的資料執行 BIN 加法運算，然後再將結果輸出至 D。

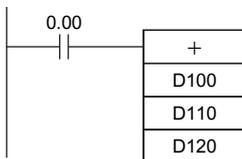


● + L

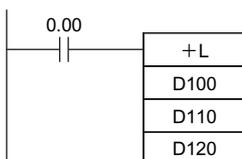
將 S1 所指定資料及 S2 所指定的資料當作倍精確度資料，執行 BIN 加法演算，然後再將結果輸出至 D。



程式例



當 0.00 變成 ON 時，利用帶符號 BIN 4 位數，將 D100 及 D110 相加後，然後再將結果輸出至 D120。



當 0.00 變成 ON 時，利用帶符號 BIN 8 位數，將 D101、D100 的 2 頻道資料及 D111、D110 的 2 頻道資料相加後，然後再將結果輸出至 D121、D120。

+ C / + CL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
帶符號·CY 的 BIN 加法運算	+ C	@ + C	402	利用帶符號 16 進位 4 位數，執行資料、常數及進位 (CY) 旗標等的加法演算。
帶符號·CY 的 BIN 倍精確度加法運算	+ CL	@ + CL	403	利用帶符號 16 進位 8 位數，執行 2CH 資料、常數及進位 (CY) 旗標等的加法演算。

符號	+ C			+ CL		
	<p>S1：被加資料 S2：加算資料 D：演算結果的輸出CH編號</p>	<p>S1：被加資料的下位CH編號 S2：加算資料的下位CH編號 D：演算結果的輸出下位CH編號</p>				

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		+ C	+ CL	+ C	+ CL
S1	+C：被加資料 +CL：被加資料的下位 CH 編號	INT	DINT	1	2
S2	+C：加算資料 +CL：加算資料的下位 CH 編號	INT	DINT	1	2
D	+C：演算結果的輸出 CH 編號 +CL：演算結果的輸出下位 CH 編號	INT	DINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR	
	CIO	WR	HR	AR	T	C	DM	@DM	* DM					
S1,S2	○	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	○	—	—	—	—

條件旗標的動作

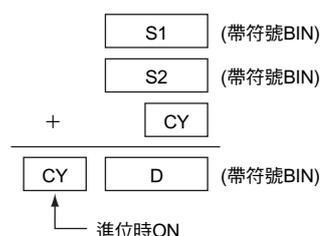
名稱	標籤	內容	
		+ C	+ CL
異常旗標	P_ER	OFF	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF
溢位旗標 (Overflow flag)	P_OF	<ul style="list-style-type: none"> 當正數 + 正數 + 進位旗標的結果符合負數的範圍 (以 16 進位表示時為 #8000 ~ FFFF) 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當正數 + 正數 + 進位旗標的結果符合負數的範圍 (以 16 進位表示時為 #80000000 ~ FFFFFFFF) 時，本旗標就會 ON 否則為 OFF
欠位旗標 (Underflow flag)	P_UF	<ul style="list-style-type: none"> 當負數 + 負數 + 進位旗標的結果在正數的範圍 (以 16 進位表示時為 #0000 ~ 7FFF) 內時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當負數 + 負數 + 進位旗標的結果在正數的範圍 (以 16 進位表示時為 #00000000 ~ 7FFFFFFF) 內時，本旗標就會 ON 否則為 OFF

名稱	標籤	內容	
		+ C	+ CL
負旗標	P_N	<ul style="list-style-type: none"> 根據演算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 根據演算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

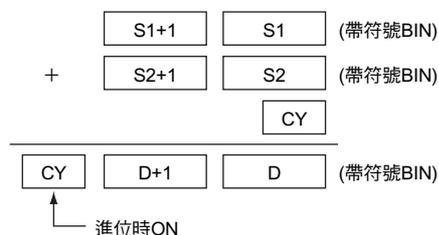
● + C

利用 BIN 加法運算，將 S1 所指定的資料、S2 所指定的資料以及進位 (CY) 旗標等相加，然後再將結果輸出至 D。



● + CL

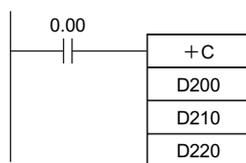
將 S1 所指定的資料、S2 所指定的資料當作倍精確度資料，並且利用 BIN 加法運算將前述資料以及進位 (CY) 旗標等相加，然後再將結果輸出至 D+1、D。



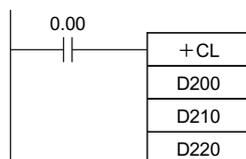
提示

- 如果要清除進位 (CY) 旗標，必須執行 CLC (清除進位) 指令。

程式例



當0.00變成ON時，利用帶符號BIN4位數將 D200、D210及CY旗標相加，然後再將結果輸出至D220。



當0.00變成ON時，利用帶符號BIN8位數將 D201、D200和D211、D210以及CY旗標相加，然後再將結果輸出至D221、D220。

+ B / + BL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
不含 CY 的 BCD 加法運算	+ B	@ + B	404	利用 BCD4 位數，將 CH 資料或常數相加。
不含 CY 的 BCD 倍精確度加法運算	+ BL	@ + BL	405	利用 BCD8 位數，將 2CH 的 CH 資料或常數相加。

符號	+B		+BL								
		<table border="1"> <tr><td>+B</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：被加資料 S2：加算資料 D：演算結果的輸出CH編號</p>	+B	S1	S2	D		<table border="1"> <tr><td>+BL</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：被加資料的下位CH編號 S2：加算資料的下位CH編號 D：演算結果的輸出下位CH編號</p>	+BL	S1	S2
+B											
S1											
S2											
D											
+BL											
S1											
S2											
D											

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		+ B	+ BL	+ B	+ BL
S1	+B：被加資料 +BL：被加資料的下位 CH 編號	WORD	DWORD	1	2
S2	+B：加算資料 +BL：加算資料的下位 CH 編號	WORD	DWORD	1	2
D	+B：演算結果的輸出 CH 編號 +BL：演算結果的輸出下位 CH 編號	WORD	DWORD	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

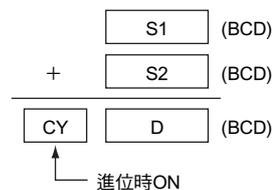
條件旗標的動作

名稱	標籤	內容	
		+ B	+ BL
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S1 的資料非 BCD 時，本旗標就會 ON 當 S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當 S1+1、S1 的資料非 BCD 時，本旗標就會 ON 當 S2+1、S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF

功能

● + B

為 S1 所指定的資料及 S2 所指定的資料執行 BCD 加法運算，然後再將結果輸出至 D。

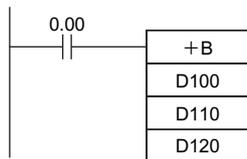


● + BL

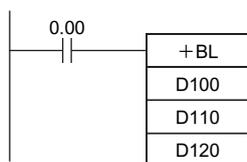
將 S1 所指定資料及 S2 所指定的資料當作倍精確度資料，執行 BCD 加法演算，然後再將結果輸出至 D+1、D。



程式例



當0.00變成ON時，利用BCD4位數將D100及D110相加，然後再將結果輸出至D120。



當0.00變成ON時，利用BCD8位數將D101、D100及D111、D110相加，然後再將結果輸出至D121、D120。

+ BC/ + BCL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
含 CY 的 BCD 加法運算	+ BC	@ + BC	406	利用帶符號的 BCD4 位數，將 CH 資料、常數以及進位 (CY) 旗標等相加。
含 CY 的 BCD 倍精確度加法運算	+ BCL	@ + BCL	407	利用 BCD8 位數，將相當於 2CH 的 CH 資料、常數以及進位 (CY) 旗標等相加。

符號	+ BC			+ BCL									
			<table border="1"> <tr><td>+BC</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1: 被加資料 S2: 加算資料 D: 演算結果的輸出CH編號</p>	+BC	S1	S2	D		<table border="1"> <tr><td>+BCL</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1: 被加資料的下位CH編號 S2: 加算資料的下位CH編號 D: 演算結果的輸出下位CH編號</p>	+BCL	S1	S2	D
+BC													
S1													
S2													
D													
+BCL													
S1													
S2													
D													

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		+ BC	+ BCL	+ BC	+ BCL
S1	被加資料	WORD	DWORD	1	2
S2	加算資料	WORD	DWORD	1	2
D	+BC: 演算結果的輸出 CH 編號 +BCL: 演算結果的輸出下位 CH 編號	WORD	DWORD	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	○	-	-	-

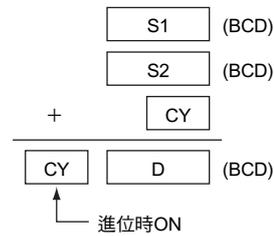
條件旗標的動作

名稱	標籤	內容	
		+ BC	+ BCL
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S1 的資料非 BCD 時，本旗標就會 ON 當 S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當 S1+1、S1 的資料非 BCD 時，本旗標就會 ON 當 S2+1、S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當加法運算造成進位時，本旗標就會 ON 否則為 OFF

功能

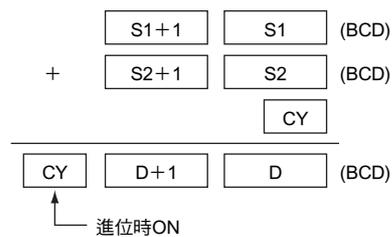
● + BC

利用 BCD 加法運算，將 S1 所指定的資料、S2 所指定的資料以及進位 (CY) 旗標相加，然後再將結果輸出至 D。



● + BCL

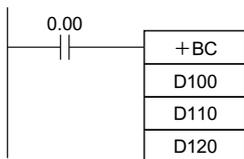
將 S1 所指定的資料、S2 所指定的資料當作倍精確度資料，並且利用 BCD 加法運算將前述資料以及進位 (CY) 旗標等相加，然後再將結果輸出至 D+1、D。



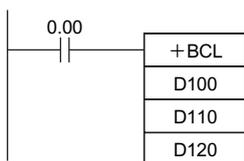
提示

- 如果要清除進位 (CY) 旗標，必須執行 CLC (清除進位) 指令。

程式例



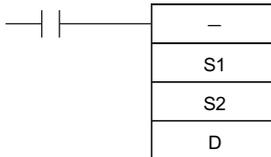
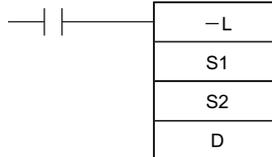
當0.00變成ON時，以BCD4位數將D100、D110及CY旗標相加，然後再將結果輸出至D120。



當0.00變成ON時，以BCD8位數將D101、D100和D111、D110以及CY旗標相加，然後再將結果輸出至D121、D120。

- / - L

指令名稱	指令記號	指令的各種組合	Fun No.	功能
帶符號・無 CY 的 BIN 減法運算	-	@ -	410	利用帶符號的 16 進位 4 位數，將 CH 資料或常數相減。
帶符號・無 CY 的 BIN 倍精確度減法運算	- L	@ - L	411	利用帶符號的 16 進位 8 位數，減去相當於 2CH 的 CH 資料或常數。

符號	-			- L		
	 <p>S1: 被減資料 S2: 相減的資料 D: 演算結果的輸出 CH 編號</p>	 <p>S1: 被減資料的下位 CH 編號 S2: 相減資料的下位 CH 編號 D: 演算結果的輸出下位 CH 編號</p>				

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		-	- L	-	- L
S1	- : 被減資料 - L : 被減資料的下位 CH 編號	INT	DINT	1	2
S2	- : 相減的資料 - L : 相減資料的下位 CH 編號	INT	DINT	1	2
D	- : 演算結果的輸出 CH 編號 - L : 演算結果的輸出下位 CH 編號	INT	DINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	○	-	-	-

條件旗標的動作

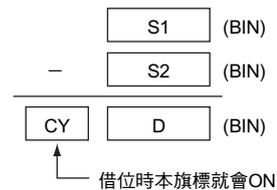
名稱	標籤	內容	
		-	- L
異常旗標	P_ER	OFF	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF
溢位旗標 (Overflow flag)	P_OF	<ul style="list-style-type: none"> 當正數-負數的結果符合負數的範圍(以 16 進位表示時為 #8000 ~ FFFF) 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當正數-負數的結果符合負數的範圍(以 16 進位表示時為 #80000000 ~ FFFFFFFF) 時，本旗標 ON 否則為 OFF
欠位旗標 (Underflow flag)	P_UF	<ul style="list-style-type: none"> 當負數+正數的結果在正數的範圍(以 16 進位表示時為 #0000 ~ 7FFF) 內時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當負數+正數的結果在正數的範圍(以 16 進位表示時為 #00000000 ~ 7FFFFFFF) 內時，本旗標就會 ON 否則為 OFF

名稱	標籤	內容	
		-	- L
負旗標	P_N	<ul style="list-style-type: none"> 根據運算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 根據運算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

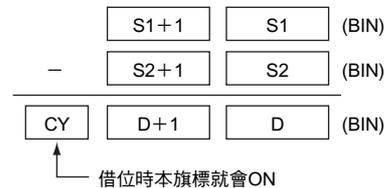
● -

為 S1 所指定的資料及 S2 所指定的資料執行 BIN 減法運算，然後再將結果輸出至 D。當結果為負數時，就會將 2 的補數輸出至 D。



● - L

將 S1 所指定資料及 S2 所指定的資料當作倍精確度資料，執行 BIN 減法演算，然後再將結果輸出至 D+1、D。當結果為負數時，就會將 2 的補數輸出至 D+1、D。



提示

● 何謂 2 的補數

就是以 1 減去 2 進位的每個位數，然後再將運算結果加 1 後所得到的數值。

例) 以 2 的補數來表示 2 進位的 1101 時，即為 1111 (16 進位表示時為 #F) - 1101 (16 進位表示時為 #D) + 1 (16 進位表示時為 #1) = 0011 (16 進位表示時為 #3)。

例) 以 16 進位 4 位數來說，如果要以 2 的補數來表示 16 進位的 #3039 時，即為 16 進位表示時的 #FFFF - 16 進位表示時的 #3039 + 16 進位表示時的 #0001 = 16 進位表示時的 #CFC7，因此使用 16 進位 4 位數時，對於 16 進位 a 而言，2 的補數就是 16 進位表示時的 #FFFF - 16 進位表示時的 a + 16 進位表示時的 #0001 = 16 進位表示時的 b。

如果要以 2 的補數所表示的 16 進位 b 來計算以真數表示的 16 進位 a 時，16 進位表示時的 a = 16 進位表示時的 #10000 - 16 進位表示時的 b。

例) 如果要以 2 的補數所表示的 16 進位 #CFC7 來計算以真數表示時，16 進位表示時的 #10000 - 16 進位表示時的 CFC7 = 16 進位表示時的 #3039。

數值範例1)

	當作帶符號資料時	當作不帶符號資料時
16進位#FFFF →	- 1	65535
-) 16進位#0001 →	-) +1	-) 00001
16進位#FFFE →	-2*1	65534*2

* 1：此時負數旗標會變成ON，結果(以16進制表示時為 #FFFE)將出現負數(2的補數)，並且以-2表示。

* 2：進位旗標OFF，結果(以16進制表示時為#FFFE)將出現不帶符號資料的正數，也就是65534。

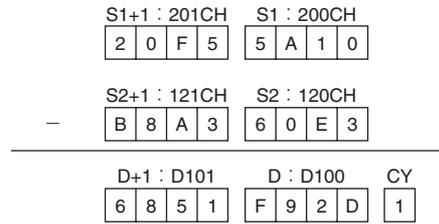
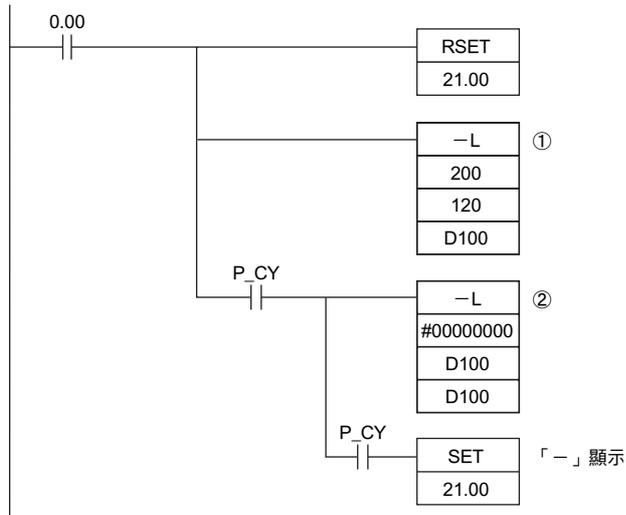
數值範例2)

	當作帶符號資料時	當作不帶符號資料時
16進位#FFFD →	- 3	65533
-) 16進位#FFFF →	-) -1	-) 65535
16進位#FFFE →	-2*3	65534*4

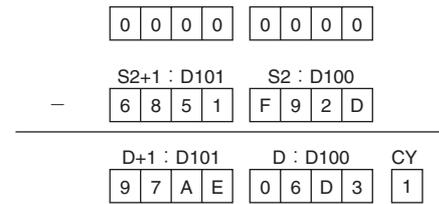
* 3：此時負數旗標會變成ON，結果(以16進制表示時為 #FFFE)將出現負數(2的補數)，並且以-2表示。

* 4：此時進位訊號變成ON，因此結果將出現(以16進制表示時為#FFFE)負數(2的補數)，轉換為真數時為-2。

(例) 16 進位：20F55A10 - B8A360E3 = -97AE06D3 ①的減法運算

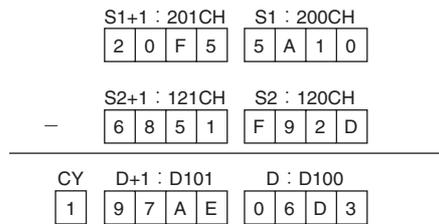


②的減法運算...CY 旗標為 1，因此將補數修改為真數。



根據 201、200CH 的資料，為 121、120CH 執行 BCD8 位數的減法運算，結果會被輸出至 D101、D100 的 8 位數。當結果為負數時，就會執行程式②，也就是 #00000000 - 補數 = 真數，接著會再次將真數輸出至 D101、D100。

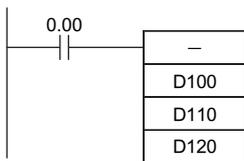
減法運算的最後結果



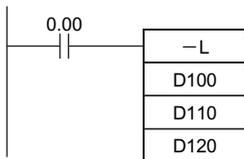
此時 CY 旗標會變成 ON，因此實際數值將顯示為 -97AE06D3。

當 D101、D100 的內容顯示為負數時，該內容就會被輸出至 21.00。

程式例



當 0.00 變成 ON 時，就會以 16 進制的 4 位數減去 D100 ~ D110 的數值，然後再將結果輸出至 D120。



當 0.00 變成 ON 時，利用 16 進制 8 位數減去 D101、D100 ~ D110、D110，然後再將結果輸出至 D121、D120。

當減法運算的結果為負數時 (S1 < S2 或 S1+1、S1 < S2+1、S2)，就會以 2 的補數來輸出運算結果，此時，CY 旗標將變成 ON。

如果要將 2 的補數轉換為真數時，程式必須以 CY 旗標為輸入條件，然後再用 0 減去減法運算的結果。此時，結果將以負數表示，因此可用來表示 CY 旗標等。

— C/ — CL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
帶符號·CY 的 BIN 減法運算	— C	@ — C	412	利用帶符號的 16 進位 4 位數，減去 CH 資料、常數以及進位 (CY) 旗標等。
帶符號·CY 的 BIN 倍精確度減法運算	— CL	@ — CL	413	利用帶符號的 16 進位 8 位數，減去 2CH 的 CH 資料、常數以及進位 (CY) 旗標等。

符號	— C			— CL		
	<p>S1：被減資料 S2：相減的資料 D：演算結果的輸出CH編號</p>	<p>S1：被減資料的下位CH編號 S2：相減資料的下位CH編號 D：演算結果的輸出下位CH編號</p>				

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		— C	— CL	— C	— CL
S1	— C：被減資料 — CL：被減資料的下位 CH 編號	INT	DINT	1	2
S2	— C：相減的資料 — CL：相減資料的下位 CH 編號	INT	DINT	1	2
D	— C：演算結果的輸出 CH 編號 — CL：演算結果的輸出下位 CH 編號	INT	DINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	○	—	—	—

條件旗標的動作

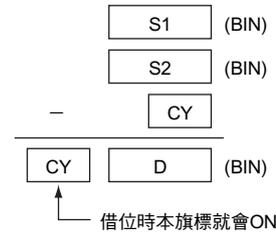
名稱	標籤	內容	
		— C	— CL
異常旗標	P_ER	OFF	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF
溢位旗標 (Overflow flag)	P_OF	<ul style="list-style-type: none"> 當正數-負數-進位旗標的結果符合負數的範圍 (以 16 進位表示時為 #8000 ~ FFFF) 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當正數-負數-進位旗標的結果符合負數的範圍 (以 16 進位表示時為 #80000000 ~ FFFFFFFF) 時，本旗標 ON 否則為 OFF
欠位旗標 (Underflow flag)	P_UF	<ul style="list-style-type: none"> 當負數+正數-進位旗標的結果在正數的範圍 (以 16 進位表示時為 #0000 ~ 7FFF) 內時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當負數+正數-進位旗標的結果在正數的範圍 (以 16 進位表示時為 #00000000 ~ 7FFFFFFF) 內時，本旗標就會 ON 否則為 OFF

名稱	標籤	內容	
		- C	- CL
負旗標	P_N	<ul style="list-style-type: none"> 根據演算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 根據演算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

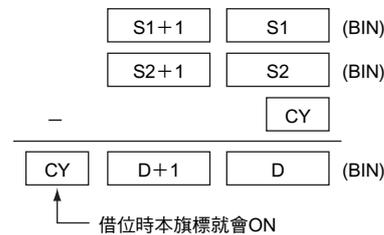
● - C

利用 BIN 減法運算，將 S1 所指定的資料、S2 所指定的資料以及進位 (CY) 旗標等相減，然後再將結果輸出至 D。當結果為負數時，就會將 2 的補數輸出至 D。



● - CL

將 S1 所指定的資料、S2 所指定的資料當作倍精確度資料，並且利用 BIN 減法運算將前述資料以及進位 (CY) 旗標等相減，然後再將結果輸出至 D+1、D。當結果為負數時，就會將 2 的補數輸出至 D+1、D。



提示

- 如果要清除進位 (CY) 旗標，必須執行 CLC (清除進位) 指令。
- 何謂 2 的補數

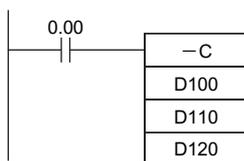
就是以 1 減去 2 進位的每個位數，然後再將運算結果加 1 後所得到的數值。

例) 以 2 的補數來表示 2 進位的 1101 時，即為 1111 (16 進位表示時為 #F) - 1101 (16 進位表示時為 #D) + 1 (16 進位表示時為 #1) = 0011 (16 進位表示時為 #3)。

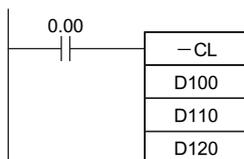
例) 以 16 進位 4 位數來說，如果要以 2 的補數來表示 16 進位的 #3039 時，即為 16 進位表示時的 #FFFF - 16 進位表示時的 #3039 + 16 進位表示時的 #0001 = 16 進位表示時的 #CFC7，因此使用 16 進位 4 位數時，對於 16 進位 a 而言，2 的補數就是 16 進位表示時的 #FFFF - 16 進位表示時的 a + 16 進位表示時的 #0001 = 16 進位表示時的 b。如果要以 2 的補數所表示的 16 進位 b 來計算以真數表示的 16 進位 a 時，16 進位表示時的 a = 16 進位表示時的 #10000 - 16 進位表示時的 b。

例) 如果要以 2 的補數所表示的 16 進位 #CFC7 來計算以真數表示時，16 進位表示時的 #10000 - 16 進位表示時的 CFC7 = 16 進位表示時的 #3039。

程式例



當 0.00 變成 ON 時，利用 16 進位 4 位數減去 D100 ~ D110 及 CY 旗標，然後再將結果輸出至 D120。



當 0.00 變成 ON 時，利用 16 進位 8 位數減去 D101、D100 ~ D111、D110 及 CY 旗標，然後再將結果輸出至 D121、D120。

當減法運算的結果為負數時 (S1 < S2 或 S1+1、S1 < S2+1、S2)，就會以 2 的補數來輸出運算結果，此時，CY 旗標將變成 ON。

如果要將 2 的補數轉換為真數時，程式必須以 CY 旗標為輸入條件，然後再用 0 減去減法運算的結果。此時，結果將以負數表示，因此可用來表示 CY 旗標等。

— B/ — BL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
不含 CY 的 BCD 減法運算	— B	@ — B	414	利用 BCD4 位數，將 CH 資料或常數相減。
不含 CY 的 BCD 倍精確度減法運算	— BL	@ — BL	415	利用 BCD8 位數，將 2CH 的 CH 資料或常數相減。

符號	— B		— BL								
		<table border="1"> <tr><td>— B</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：被減算 S2：相減的資料 D：演算結果的輸出CH編號</p>	— B	S1	S2	D		<table border="1"> <tr><td>— BL</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：被減資料的下位CH編號 S2：相減資料的下位CH編號 D：演算結果的輸出下位CH編號</p>	— BL	S1	S2
— B											
S1											
S2											
D											
— BL											
S1											
S2											
D											

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		— B	— BL	— B	— BL
S1	— B：被減資料 — BL：被減資料的下位 CH 編號	WORD	DWORD	1	2
S2	— B：相減的資料 — BL：減法運算資料的 CH 番	WORD	DWORD	1	2
D	— B：演算結果的輸出 CH 編號 — BL：演算結果的輸出下位 CH 編號	WORD	DWORD	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

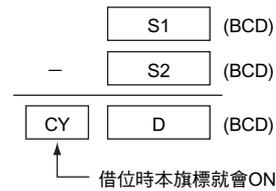
條件旗標的動作

名稱	標籤	內容	
		— B	— BL
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S1 的資料非 BCD 時，本旗標就會 ON 當 S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當 S1+1、S1 的資料非 BCD 時，本旗標就會 ON 當 S2+1、S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF

功能

● - B

為 S1 所指定的資料及 S2 所指定的資料執行 BCD 減法運算，然後再將結果輸出至 D。當結果為負數時，就會將 10 的補數輸出至 D。



● - BL

將 S1 所指定資料及 S2 所指定的資料當作倍精確度資料，執行 BCD 減法演算，然後再將結果輸出至 D+1、D。當結果為負數時，就會將 10 的補數輸出至 D+1、D。



提示

• 何謂 10 的補數

就是以 9 減去每個位數，然後再將運算結果加 1 後所得到的數值。

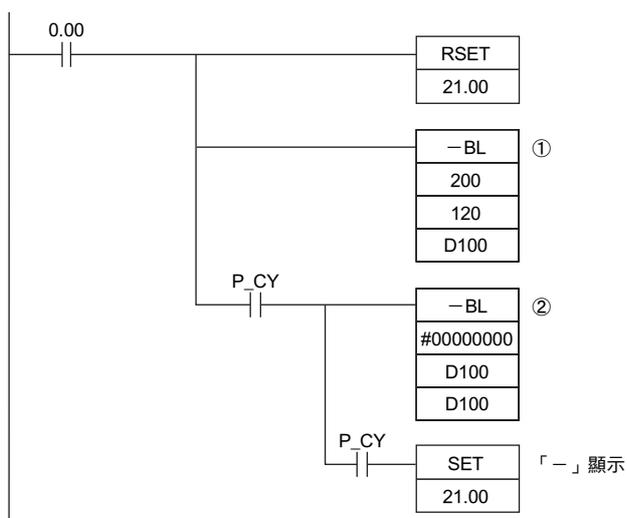
例) 對於 7556 來說，10 的補數就是 $9999 - 7556 + 1 = 2444$ 。

因此，以 4 位數表示時，對於 A 來說 10 的補數就是 $9999 - A + 1 = B$ 。

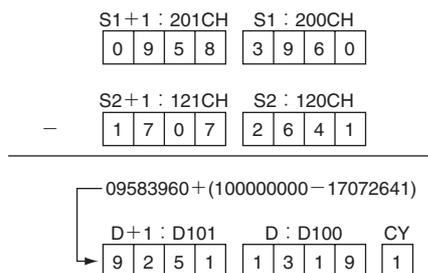
如果要以 10 的補數所表示的 B 來計算以真數表示的 A 時， $A = 10000 - B$ 。

例) 如果要以 10 的補數所表示的 2444 來計算以真數表示時， $10000 - 2444 = 7556$ 。

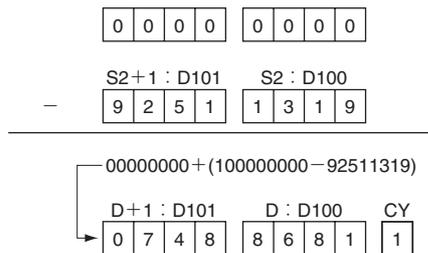
(例)BCD : 9,583,960 - 17,072,641 = - 7,488,681 ①的減法運算



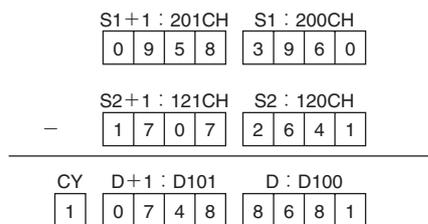
根據 201、200CH 的資料，為 121、120CH 執行 BCD8 位數的減法運算，結果會被輸出至 D101、D100 的 8 位數。當結果為負數時，就會執行程式②，也就是 #00000000 - 補數 = 真數，接著會再次將真數輸出至 D101、D100。



②的減法運算...CY 旗標為 1，因此將補數修改為真數。

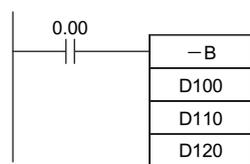


減法運算的最後結果

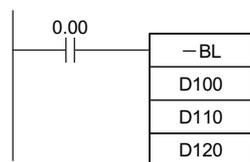


此時 CY 旗標會變成 ON，因此實際數值將顯示為 - 7、488、681。
當 D101、D100 的內容顯示為負數時，該內容就會被輸出至 21.00。

程式例



當 0.00 變成 ON 時，利用 BCD4 位數將 D100~D110 相加，然後再將結果輸出至 D120。



當 0.00 變成 ON 時，利用 BCD8 位數將 D101、D100~D111、D110 相加，然後再將結果輸出至 D121、D120。

當減法運算的結果為負數時 (S1 < S2 或 S1+1、S1 < S2+1、S2)，就會以 10 的補數來輸出運算結果，此時，CY 旗標將變成 ON。
如果要將 10 的補數轉換為真數時，程式必須以 CY 旗標為輸入條件，然後再用 0 減去減法運算的結果。此時，結果將以負數表示，因此可用來表示 CY 旗標等。

- BC/ - BCL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
含 CY 的 BCD 減法運算	- BC	@ - BC	416	利用 BCD4 位數減去 CH 資料、常數以及進位 (CY) 旗標等。
含 CY 的 BCD 倍精確度減法運算	- BCL	@ - BCL	417	利用 BCD8 位數，將相當於 2CH 的 CH 資料、常數以及進位 (CY) 旗標等相減。

符號	- BC			- BCL		
		S1	S2	D	S1	S2
	S1：被減資料 S2：相減的資料 D：演算結果的輸出CH編號			S1：被減資料的下位CH編號 S2：相減資料的下位CH編號 D：演算結果的輸出下位CH編號		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		- BC	- BCL	- BC	- BCL
S1	- BC：被減資料 - BCL：被減資料的下位 CH 編號	WORD	DWORD	1	2
S2	- BC：相減的資料 - BCL：相減資料的下位 CH 編號	WORD	DWORD	1	2
D	- BC：演算結果的輸出 CH 編號 - BCL：演算結果的輸出下位 CH 編號	WORD	DWORD	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

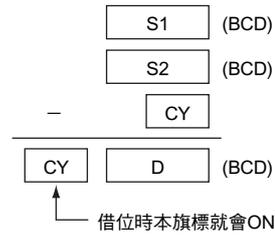
條件旗標的動作

名稱	標籤	內容	
		- BC	- BCL
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S1 的資料非 BCD 時，本旗標就會 ON 當 S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當 S1+1、S1 的資料非 BCD 時，本旗標就會 ON 當 S2+1、S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當減法運算造成借位時，本旗標就會 ON 否則為 OFF

功能

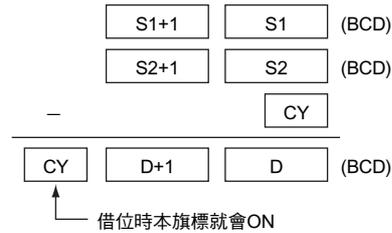
● - BC

利用 BCD 減法運算，將 S1 所指定的資料、S2 所指定的資料以及進位 (CY) 旗標相加，然後再將結果輸出至 D。當結果為負數時，就會將 10 的補數輸出至 D。



● - BCL

將 S1 所指定的資料、S2 所指定的資料當作倍精確度資料，並且利用 BIN 減法運算將前述資料以及進位 (CY) 旗標等相減，然後再將結果輸出至 D+1、D。當結果為負數時，就會將 10 的補數輸出至 D+1、D。



提示

- 如果要清除進位 (CY) 旗標，必須執行 CLC (清除進位) 指令。
- 何謂 10 的補數

就是以 9 減去每個位數，然後再將運算結果加 1 後所得到的數值。

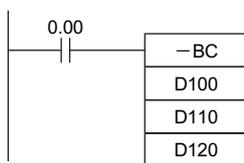
例) 對於 7556 來說，10 的補數就是 $9999 - 7556 + 1 = 2444$ 。

因此，以 4 位數表示時，對於 A 來說 10 的補數就是 $9999 - A + 1 = B$ 。

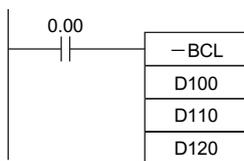
如果要以 10 的補數所表示的 B 來計算以真數表示的 A 時， $A = 10000 - B$ 。

例) 如果要以 10 的補數所表示的 2444 來計算以真數表示時， $10000 - 2444 = 7556$ 。

程式例



當 0.00 變成 ON 時，以 BCD4 位數將 D100~D110 及 CY 旗標相減，然後再將結果輸出至 D120。



當 0.00 變成 ON 時，利用 BCD8 位數減去 D101、D100 ~ D111、D110 及 CY 旗標，然後再將結果輸出至 D121、D120。

當減法運算的結果為負數時 ($S1 < S2$ 或 $S1+1、S1 < S2+1、S2$)，就會以 10 的補數來輸出運算結果，此時，CY 旗標將變成 ON。

如果要將 10 的補數轉換為真數時，程式必須以 CY 旗標為輸入條件，然後再用 0 減去減法運算的結果。此時，結果將以負數表示，因此可用來表示 CY 旗標等。

* / * L

指令名稱	指令記號	指令的各種組合	Fun No.	功能
帶符號 BIN 乘法運算	*	@ *	420	利用帶符號的 16 進位 4 位數，將 CH 資料或常數相乘。
帶符號 BIN 倍精確度乘法運算	* L	@ * L	421	利用帶符號的 16 進位 8 位數，乘以相當於 2CH 的 CH 資料或常數。

符號	*			* L									
		<table border="1"> <tr><td>*</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table>	*	S1	S2	D	S1：被乘資料 S2：相乘資料 D：演算結果的輸出下位CH編號		<table border="1"> <tr><td>* L</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table>	* L	S1	S2	D
*													
S1													
S2													
D													
* L													
S1													
S2													
D													

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		*	* L	*	* L
S1	*：被乘資料 * L：被乘資料的下位 CH 編號	INT	DINT	1	2
S2	*：相乘資料 * L：相乘資料的下位 CH 編號	INT	DINT	1	2
D	演算結果的輸出下位 CH 編號	DINT	LINT	2	4

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	○	—	—	—

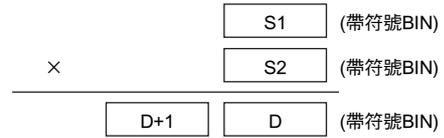
條件旗標的動作

名稱	標籤	內容	
		*	* L
異常旗標	P_ER	OFF	OFF
=旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據演算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 根據演算結果，當最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能

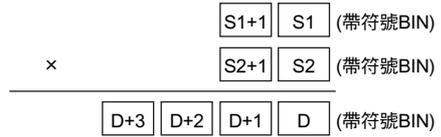
● *

利用帶符號 BIN 乘法運算，將 S1 所指定的資料及 S2 所指定的資料相乘，然後再將結果輸出至 D+1、D。

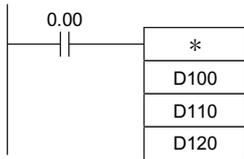


● *L

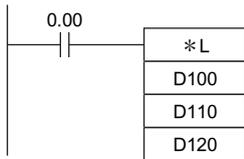
將 S1 所指定的資料及 S2 所指定的資料當作倍精確度資料，利用帶符號的 BIN 乘法運算，將前述資料相乘，然後再將結果輸出至 D+3、D+2、D+1、D。



程式例



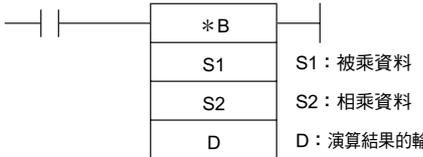
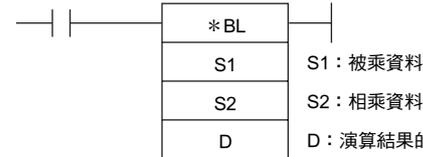
當0.00變成ON時，就會以16進位的4位數乘以D100及D110的數值，然後再將結果輸出至D121、D120。



當0.00變成ON時，利用帶符號的16進位8位數，將D101、D100以及D111、D110相乘，然後再將結果輸出至D123、D122、D121、D120。

* B / * BL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
BCD 乘法運算	* B	@ * B	424	利用 BCD4 位數，將 CH 資料或常數相乘。
BCD 倍精確度乘法運算	* BL	@ * BL	425	利用 BCD8 位數，將相當於 2CH 的 CH 資料或常數相乘。

符號	* B	* BL
	 <p>S1：被乘資料 S2：相乘資料 D：演算結果的輸出下位CH編號</p>	 <p>S1：被乘資料的下位CH編號 S2：相乘資料的下位CH編號 D：演算結果的輸出下位CH編號</p>

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		* B	* BL	* B	* BL
S1	* B：被乘資料 * BL：被乘資料的下位 CH 編號	WORD	DWORD	1	2
S2	* B：相乘資料 * BL：相乘資料的下位 CH 編號	WORD	DWORD	1	2
D	演算結果的輸出下位 CH 編號	DWORD	LWORD	2	4

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

條件旗標的動作

名稱	標籤	內容	
		* B	* BL
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S1 的資料非 BCD 時，本旗標就會 ON 當 S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當 S1+1、S1 的資料非 BCD 時，本旗標就會 ON 當 S2+1、S2 的資料非 BCD 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF

功能

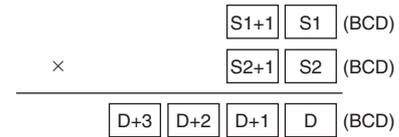
● * B

為 S1 所指定的資料及 S2 所指定的資料執行 BCD 乘法運算，然後再將結果輸出至 D+1、D。

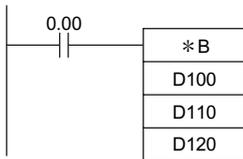


● * BL

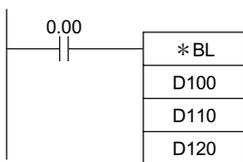
將 S1 所指定的資料及 S2 所指定的資料當作倍長度資料，利用 BCD 乘法運算，將前述資料相乘，然後再將結果輸出至 D+3、D+2、D+1、D。



程式例



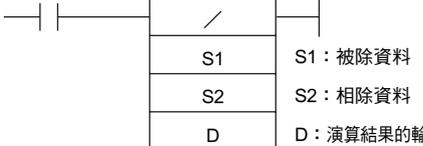
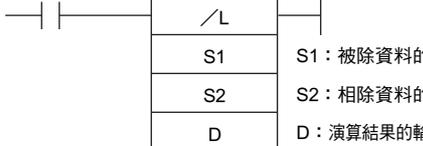
當0.00變成ON時，利用BCD4位數將D100及D110相乘，然後再將結果輸出至D121、D120。



當0.00變成ON時，利用BCD8位數將D101、D100及D111、D110相乘，然後再將結果輸出至D123、D122、D121、D120。

／, ／L

指令名稱	指令記號	指令的各種組合	Fun No.	功能
帶符號 BIN 除法運算	／	@ ／	430	利用帶符號的 16 進位 4 位數，將 CH 資料或常數相除。
帶符號 BIN 倍精確度除法運算	／L	@ ／L	431	利用帶符號的 16 進位 8 位數，除以相當於 2CH 的 CH 資料或常數。

符號	／		／L	
	 <p>S1：被除資料 S2：相除資料 D：演算結果的輸出下位CH編號</p>	 <p>S1：被除資料的下位CH編號 S2：相除資料的下位CH編號 D：演算結果的輸出下位CH編號</p>		

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		／	／L	／	／L
S1	／：被除資料 ／L：被除資料的下位 CH 編號	INT	DINT	1	2
S2	／：相除資料 ／L：相除資料的下位 CH 編號	INT	DINT	1	2
D	演算結果的輸出下位 CH 編號	DWORD	LWORD	2	4

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	○	—	—	—

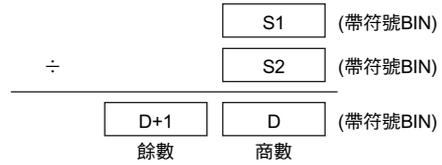
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當相除資料變成 0 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當運算結果所得到的商為 0 時，本旗標就會 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 在運算結果的商中，只要最上位元變成 1 時，本旗標就會 ON 否則為 OFF

功能



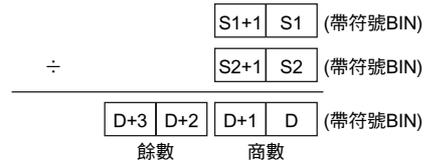
以帶符號 BIN 資料來說，其意思就是先計算 $S1 \div S2$ 的結果，然後再將商數 (16 位元) 輸出至 D，餘數 (16 位元) 則會被輸出至 D。



(註) 一旦以 16 進位表示時的 #8000 ÷ 16 進位表示時的 #FFFF 時，就會較不穩定。

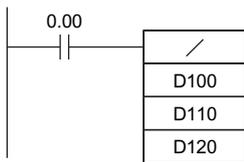


以帶符號 BIN 資料來說，其意思就是先計算 $S1 \div S2$ 的結果，然後再將商數 (32 位元) 輸出至 D+1、D，餘數 (32 位元) 則會被輸出至 D+3、D+2。

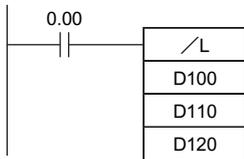


(註) 一旦以 16 進位表示時的 #80000000 ÷ 16 進位表示時的 #FFFFFFF 時，就會較不穩定。

程式例



當 0.00 變成 ON 時，利用 16 進位 4 位數將 D110 除以 D100，然後再將運算結果的商輸出至 D120，而餘數則輸出至 D121。



當 0.00 變成 ON 時，利用 16 進位 8 位數將 D101、D100 除以 D111、D110，然後再將運算結果的商輸出至 D121、D120，而餘數則輸出至 D123、D122。

／ B , ／ BL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
BCD 除法運算	／ B	@ ／ B	434	利用 BCD4 位數，將 CH 資料或常數相除。
BCD 倍精確度除法運算	／ BL	@ ／ BL	435	利用 BCD8 位數，將 2CH 的 CH 資料或常數相除。

符號	／ B	／ BL
	<p>S1：被除資料 S2：相除資料 D：演算結果的輸出下位CH編號</p>	<p>S1：被除資料的下位CH編號 S2：相除資料的下位CH編號 D：演算結果的輸出下位CH編號</p>

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		／ B	／ BL	／ B	／ BL
S1	／ B：被除資料 ／ BL：被除資料的下位 CH 編號	WORD	DWORD	1	2
S2	／ B：相除資料 ／ BL：相除資料的下位 CH 編號	WORD	DWORD	1	2
D	演算結果的輸出下位 CH 編號	DWORD	LWORD	2	4

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

條件旗標的動作

名稱	標籤	內容	
		／ B	／ BL
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S1 的資料非 BCD 時，本旗標就會 ON 當 S2 的資料非 BCD 時，本旗標就會 ON 當相除資料變成 0 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當 S1+1、S1 的資料非 BCD 時，本旗標就會 ON 當 S2+1、S2 的資料非 BCD 時，本旗標就會 ON 當相除資料變成 0 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當運算結果所得到的商為 0 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當運算結果所得到的商為 0 時，本旗標就會 ON 否則為 OFF

功能

● /B

以 BCD 資料 (16 位元) 來說，就是先計算 $S1 \div S2$ 的結果，然後再將商數 (16 位元) 傳送至 D，而餘數 (16 位元) 則會被輸出至 D+1。

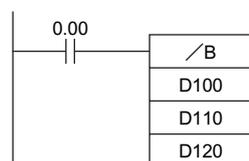


● /BL

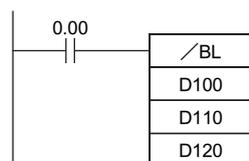
以 BCD 資料來說，其意思就是先計算 $S1 \div S2$ 的結果，然後再將商數 (32 位元) 輸出至 D+1、D，餘數 (32 位元) 則會被輸出至 D+3、D+2。



程式例



當 0.00 變成 ON 時，利用 BCD 4 位數將 D110 除以 D100，然後再將運算結果的商輸出至 D120，而餘數則輸出至 D121。



當 0.00 變成 ON 時，利用 BCD 8 位數將 D101、D100 除以 D111、D110，然後再將運算結果的商輸出至 D121、D120，而餘數則輸出至 D123、D122。

資料轉換指令

BIN/BINL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
BCD → BIN 轉換	BIN	@BIN	023	將 BCD 資料轉換為 BIN 資料。
BCD → BIN 倍精確度轉換	BINL	@BINL	058	將倍精確度 BCD 資料轉換為 BIN 資料。

符號	BIN		BINL						
		<table border="1"> <tr><td>BIN</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> </table> <p>S：轉換資料的CH編號 D：輸出轉換結果的CH編號</p>	BIN	S	D		<table border="1"> <tr><td>BINL</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> </table> <p>S：轉換資料的下位CH編號 D：輸出轉換結果的下位CH編號</p>	BINL	S
BIN									
S									
D									
BINL									
S									
D									

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		BIN	BINL	BIN	BINL
S	BIN：轉換資料的 CH 編號 BINL：轉換資料的下位 CH 編號	WORD	DWORD	1	2
D	BIN：輸出轉換結果的 CH 編號 BINL：輸出轉換結果的下位 CH 編號	UINT	UDINT	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S,D	○	○	○	○	○	○	○	○	○	—	—	—	—

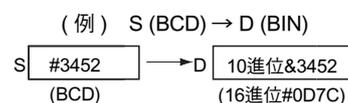
條件旗標的動作

名稱	標籤	內容	
		BIN	BINL
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當 S+1、S 的資料非 BCD 時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當轉換結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當轉換結果為 0 時，本旗標 ON 否則為 OFF
負旗標	P_N	OFF	OFF

功能

● BIN

利用 BIN 來轉換 S 所指定的 BCD 資料，然後再將結果輸出至 D。

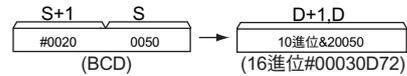


● BINL

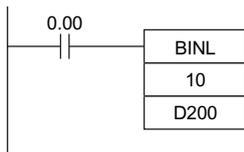
將 S+1、S 所指定的 BCD 資料當作倍精確度資料，並進行 BIN 轉換，接著將結果輸出至 D+1、D。



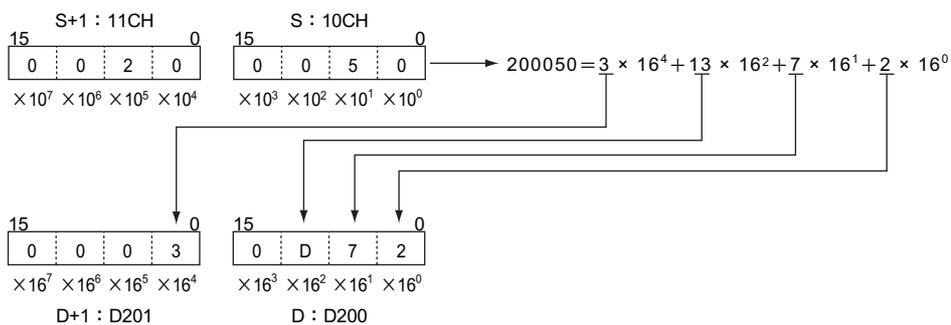
(例) S + 1、S (BCD) → D + 1、D (BIN)



程式例

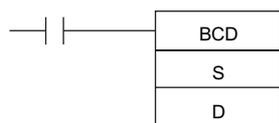
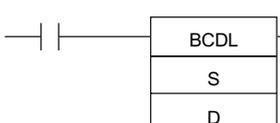


當0.00變成ON時，將11CH、10CH的BCD8位數資料轉換為BIN32位元資料，然後再將結果輸出至資料記憶體D201、D200。



BCD/BCDL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
BIN → BCD 轉換	BCD	@BCD	024	將 BIN 資料轉換為 BCD 資料。
BIN → BCD 倍精確度轉換	BCDL	@BCDL	059	將倍精確度 BIN 資料轉換為 BCD 資料。

符號	BCD	BCDL
	 <p>S : 轉換資料的CH編號 D : 輸出轉換結果的CH編號</p>	 <p>S : 轉換資料的下位CH編號 D : 輸出轉換結果的下位CH編號</p>

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		BCD	BCDL	BCD	BCDL
S	BCD : 轉換資料的 CH 編號 BCDL : 轉換資料的下位 CH 編號	UINT	UDINT	1	2
D	BCD : 輸出轉換結果的 CH 編號 BCDL : 輸出轉換結果的下位 CH 編號	WORD	DWORD	1	2

S : 轉換資料的 CH 編號

- BCD
10 進位表示時為 &0 ~ 9999 或是 16 進位表示時為 #0000 ~ 270F (轉換後 BCD#0000 ~ 9999 的數值)
- BCDL
10 進位表示時為 &0 ~ 99999999 或是 16 進位表示時為 #00000000 ~ 05F5E0FF (轉換後 BCD#00000000 ~ 99999999 的數值)

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S,D	○	○	○	○	○	○	○	○	○	-	-	-	-

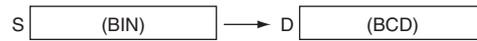
條件旗標的動作

名稱	標籤	內容	
		BCD	BCDL
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S 的資料超過 10 進位表示時的 &0 ~ 9999 或是 16 進位表示時的 #0000 ~ 270F 等範圍時，本旗標就會 ON 否則為 OFF 	<ul style="list-style-type: none"> 當 S+1\ S 的資料超過 10 進位表示時的 &0 ~ 99999999 或是 16 進位表示時的 #00000000 ~ 05F5E0FF 等範圍時，本旗標就會 ON 否則為 OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當轉換結果為 0 時，本旗標 ON 否則為 OFF 	<ul style="list-style-type: none"> 當轉換結果為 0 時，本旗標 ON 否則為 OFF

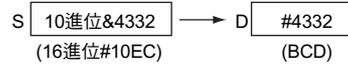
功能

● BCD

將 S 所指定的 BIN 資料轉換為 BCD，然後再將結果輸出至 D。



(例) S (BIN) → D (BCD)

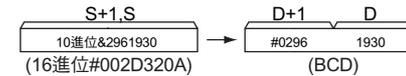


● BCDL

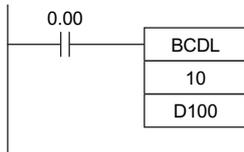
將 S+1、S 所指定的 BIN 資料當作倍精確度資料，並進行 BCD 轉換，接著將結果輸出至 D+1、D。



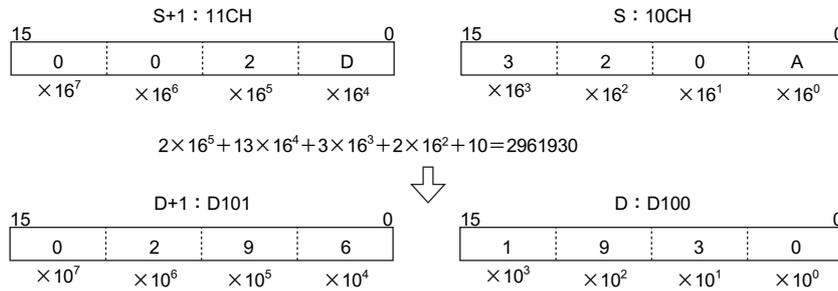
(例) S + 1、S (BIN) → D + 1、D (BCD)



程式例



當0.00變成ON時，將11CH、10CH的BIN32位元資料轉換為BCD8位數資料，然後再將結果輸出至資料記憶體D101、D100。



NEG

指令名稱	指令記號	指令的各種組合	Fun No.	功能
轉換 2 的補數	NEG	@NEG	160	以 2 的補數表示 BIN16 位元資料。

符號	NEG	
		S : 轉換資料的CH編號 D : 輸出轉換結果的CH編號

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
		NEG	NEG
S	NEG : 轉換資料的 CH 編號	WORD	1
D	NEG : 輸出轉換結果的 CH 編號	UINT	1

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> 當轉換結果為 0 時，本旗標 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據轉換結果，當最上位元為 1 時，本旗標就會 ON 否則為 OFF

功能

● NEG

將 S 所指定的資料進行位元反相並 +1，然後再將結果輸出至 D。

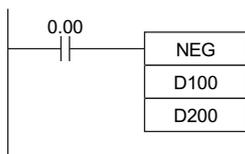
$$\frac{\text{2的補數}}{\text{(位元反轉後+1)}} \rightarrow (D)$$

(註) 16 進位表示時的 #8000 其轉換結果以 16 進位表示時為 #8000。

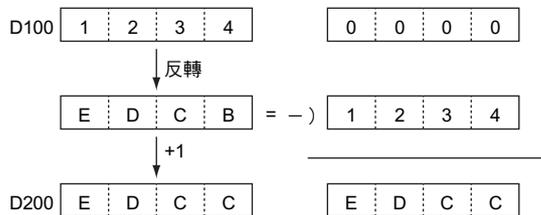
提示

- 將位元反相後 +1 就和以 0 減去 S 的內容這項操作動作相同。

程式例



當0.00變成ON時，就會以2的補數來表示D100的內容，然後再將結果輸出至D200。



MLPX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
4 → 16/8 → 256 指令碼	MLPX	@MLPX	076	讀取指定 CH 的指定位數或是指定位元 (4 → 16 或 8 → 256 指令碼)，然後再將 1 輸出至指定 CH 所對應的位元，而 0 則輸出至其他位元。

符號	MLPX						
		<table border="1"> <tr> <td>S</td> <td>S：轉換資料的CH編號</td> </tr> <tr> <td>K</td> <td>K：控制器資料(指定位數)</td> </tr> <tr> <td>D</td> <td>D：輸出轉換結果的下位CH編號</td> </tr> </table>	S	S：轉換資料的CH編號	K	K：控制器資料(指定位數)	D
S	S：轉換資料的CH編號						
K	K：控制器資料(指定位數)						
D	D：輸出轉換結果的下位CH編號						

是否適用於特定區域

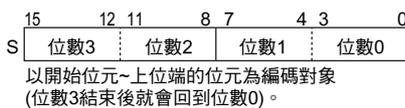
區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
S	轉換資料的 CH 編號	UINT	1
K	控制器資料 (指定位數)	UINT	1
D	輸出轉換結果的下位 CH 編號	UINT	可調整

● 使用 4 → 16 指令碼時

S：轉換資料的 CH 編號

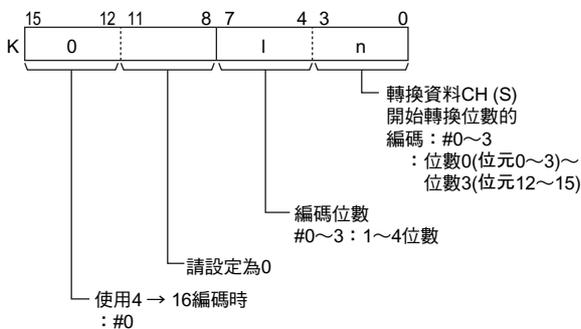


D：輸出轉換結果的下位 CH 編號

- D：第 1 個指令碼位數的指令碼結果
- D+1：第 2 個指令碼位數的指令碼結果
- D+2：第 3 個指令碼位數的指令碼結果
- D+3：第 4 個指令碼位數的指令碼結果

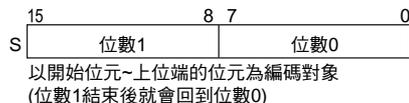
(註) D~最大 D+3 的運算元種類必須相同。

K：控制資料



● 使用 8 → 256 指令碼時

S：轉換資料的 CH 編號



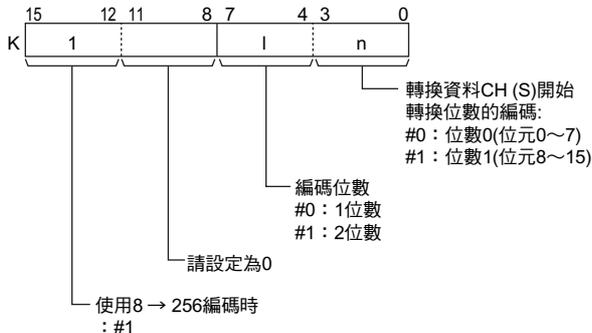
D：輸出轉換結果的下位 CH 編號

D+15 ~ D：第 1 個指令碼位數的指令碼結果

D+31 ~ D+16：第 2 個指令碼位數的指令碼結果

(註) D~ D+31 的運算元種類必須相同。

K：控制資料



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S										—			
K	○	○	○	○	○	○	○	○	○	○	—	—	—
D										—			

條件旗標的動作

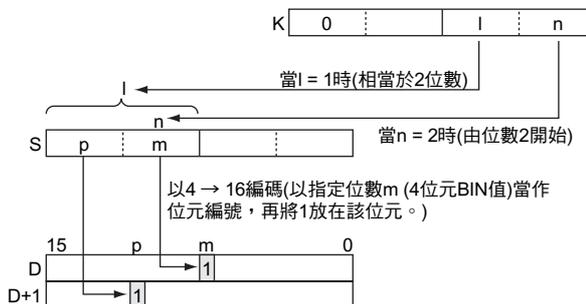
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 K 的資料超過範圍時變為 ON 否則為 OFF

功能

依轉換種類 (K) 不同，可指定為 4 → 16 或是 8 → 256 指令碼。

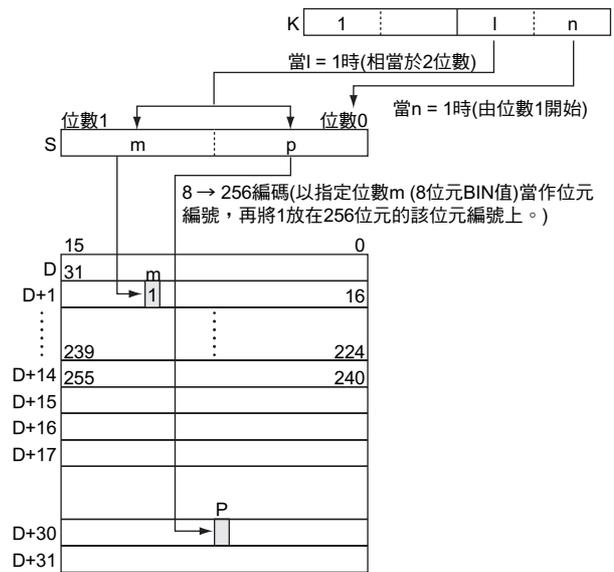
● 指定為 4 → 16 指令碼時

在 S 所指定的資料中，將 K 所指定的 (由位數 N 到 I+1 個部分的) 每個位數 (4 位元) 內容 (16 進位表示時為 #0 ~ F) 當作位元位置 (0-15)，然後再將 1 輸出至 D 所指定的每個 CH (16 位元) 以後的相關位元，而 0 則輸出至其他的位元。



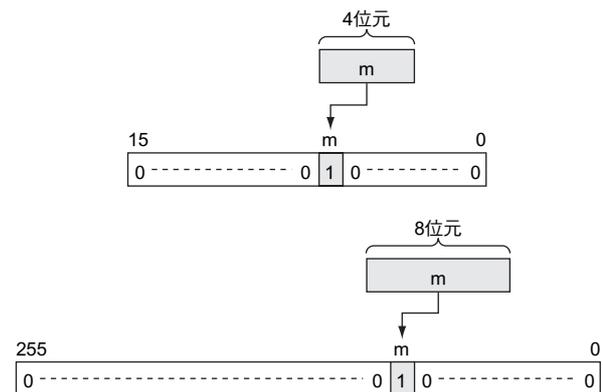
● 指定為 8 → 256 指令碼時

在 S 所指定的資料中，將 K 所指定的 (由位數 n 到 l+1 位數的) 每個位數 (8 位元) 內容 (16 進位表示時為 #00 ~ FF) 當作位元位置 (0~255)，然後再將 1 輸出至 D 所指定的每個 CH (16 位元) 以後的相關位元，而 255 則輸出至其他的位元。



提示

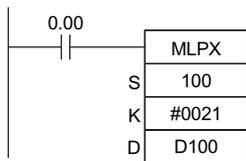
- 所謂 4 → 16 指令碼，就是如右圖所示，將 4 位元 BIN 值當作位元編號，然後再將 1 放在 16 位元中的該位元編號上，其他編號則當作 0。
- 所謂 8 → 256 指令碼，就是如右圖所示，將 8 位元 BIN 值當作位元編號，然後再將 1 放在 256 位元中的該位元編號上，其他編號則當作 0。



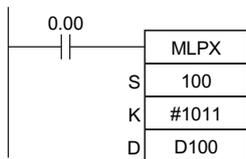
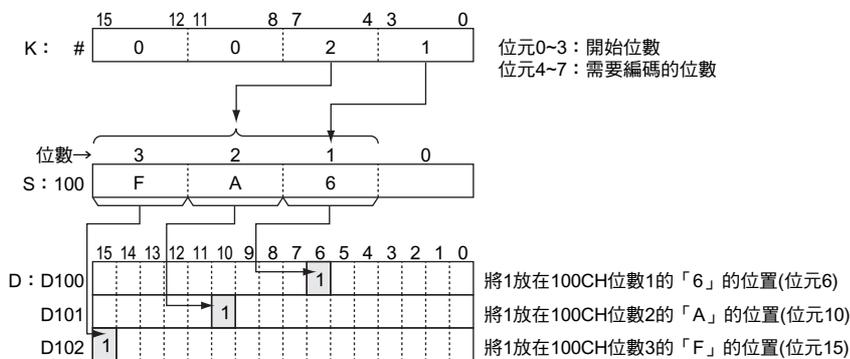
使用上的注意事項

- 指定為 4 → 16 指令碼時
利用指令碼位數 (K) 來指定多個位數時，轉換對象將依 S 的開始轉換位數 (K 的 n) ~ 上位位數端 (位數 3 結束後就會回到位數 0) 的順序，而轉換結果則依 K ~ 上位 CH 端 (每個位數為 1CH) 的順序加以儲存。
- 指定為 8 → 256 指令碼時
利用指令碼位數 (K 的 n) 來指定多個位數時，轉換對象將依 S 的開始轉換位數 (K 的 n) ~ 上位位數端 (位數 1 結束後就會回到位數 0) 的順序，而轉換結果則依 K ~ 上位 CH 端 (每個位數為 16CH) 的順序加以儲存。

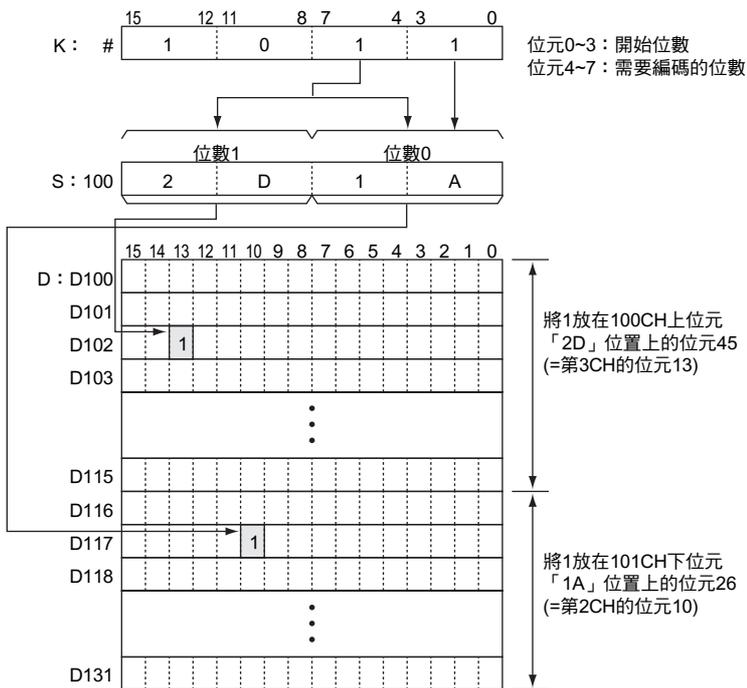
程式例



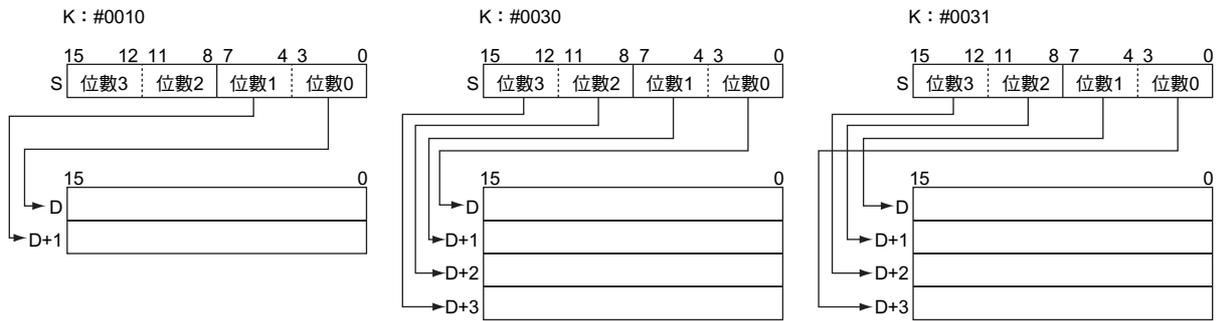
• 4 → 16編碼
當0.00變成ON時，就會將100CH第1位數開始的3位數(根據K的內容)數值(16進位)當作位元編號，然後再將1放在D100~D102的3CH所對應的位元上。



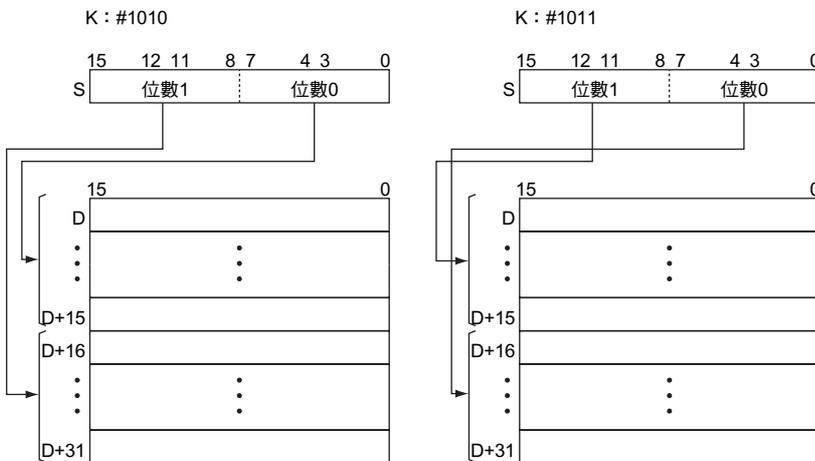
• 8 → 256編碼
當0.00變成ON時，就會將100CH的上位元開始的2位元(根據K的內容)數值(16進位)當作位元編號，然後再將1放在D100 ~ D115及D116 ~ D131各256位元所對應的位元上(依下位→上位的順序16CH會被當作連續的256位元)。



- 多位數的指令碼範例
- 使用 4 → 16 指令碼的範例



- 使用 8 → 256 指令碼的範例：



DMPX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
16 → 4/256 → 8 指令碼	DMPX	@DMPX	077	讀取指定 CH 的 16 位元或是 256 位元中已經啟動的最上位元或最下位元 (使用 16 → 4 指令碼或 256 → 8 指令碼)，然後再將讀取結果輸出至指定 CH 的指定位數或是指定位元。

符號	DMPX	
		S：轉換資料的下位CH編號 D：輸出轉換結果的CH編號 K：控制器資料(指定位數)

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
S	轉換資料的下位 CH 編號	UINT	可調整
D	輸出轉換結果的 CH 編號	UINT	1
K	控制器資料 (指定位數)	UINT	1

● 16 → 4 指令碼器時

S：轉換資料的下位 CH 編號

S：第一個指令碼位數的指令碼對象

S+1：第二個指令碼位數的指令碼對象

S+2：第三個指令碼位數的指令碼對象

S+3：第四個指令碼位數的指令碼對象

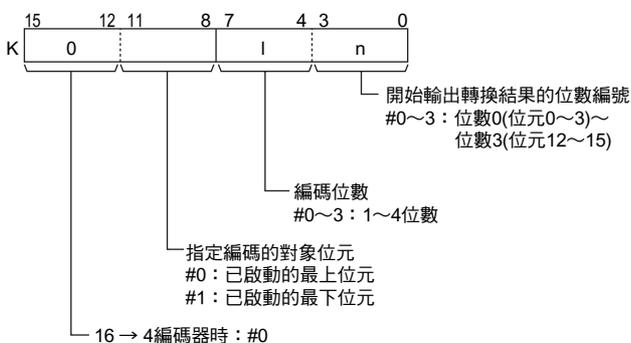
D：輸出轉換結果的 CH 編號



將 S ~ S+3 的所有編碼結果儲存在開始位元 ~ 上位元之間 (位數 3 結束後就會回到位數 0)

(註) S ~ 最大 S+3 的運算元種類必須相同。

K：控制資料



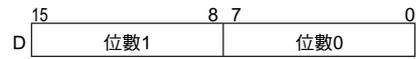
● 256 → 8 指令碼器時

S：轉換資料的下位 CH 編號

S+15 ~ S：第一個指令碼位數的指令碼對象

S+31 ~ S+16：第二個指令碼位數的指令碼對象

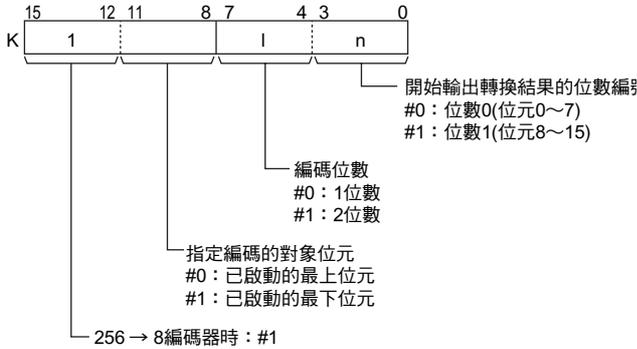
D：輸出轉換結果的 CH 編號



將 S ~ S+15、S+16 ~ S+31 的所有編碼結果儲存在開始位數~上位之間(位數1結束後就會回到位數0)

(註) S~ 最大 S+31 的運算元種類必須相同。

K：控制資料



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S,D	○	○	○	○	○	○	○	○	○	—	—	—	—
K	○									○			

條件旗標的動作

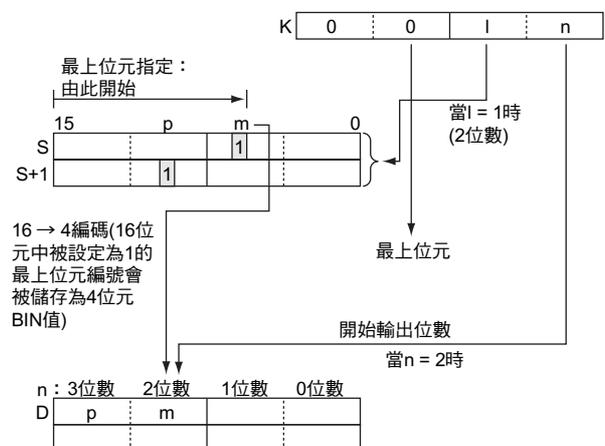
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S 所指定的轉換資料中有任何一個 CH 出現 0 時 (不存在可指令碼的位元), 本旗標就會 ON 當 K 的資料超過範圍時變為 ON 否則為 OFF

功能

依轉換種類 (K) 不同, 可指定為 16 → 4 指令碼或 256 → 8 指令碼。

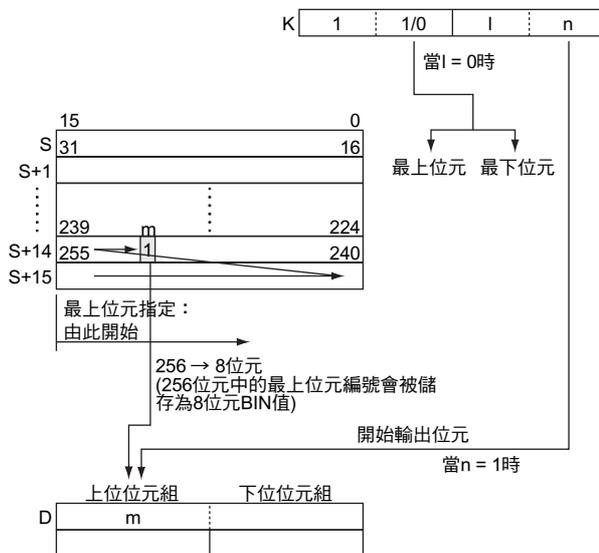
● 16 → 4 指令碼器指定時

由 S 開始到 K 的 l 所指定數值的每個 16 位元 (1CH) 中, 讀取所要指令碼位元 (在已經 ON 的位元中的最上位元或最下位) 的位元編號 (0 ~ 15), 然後再將位元編號 (16 進位表示時為 #0 ~ F) 輸出至 K 的 n 所指定的 D 的開始輸出位數 (4 位元) 以後的位數。



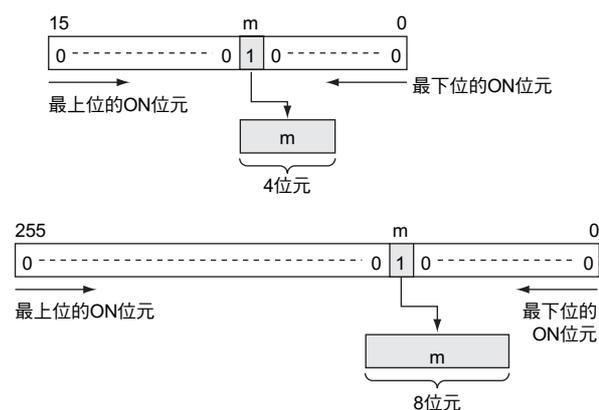
● 256 → 8 指令碼器指定時

由 S 開始到 K 的 I 所指定數值的每個 256 位元 (16CH) 中，讀取所要指令碼位元 (在已經 ON 的位元中的最上位元或最下位) 的位元編號 (0 ~ 255)，然後再將位元編號 (16 進位表示時為 #00 ~ FF) 輸出至 K 的 n 所指定的 D 的開始輸出位數 (8 位元) 以後的位數。



提示

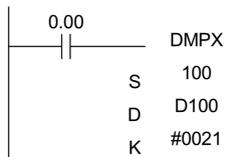
- 所謂 16 → 4 指令碼就是如右圖所示，將 16 位元中被設定為 1 的最上位或最下位元編號 (m) 轉換成 4 位元 BIN 值。
- 所謂 256 → 8 指令碼就是如右圖所示，將 256 位元中被設定為 1 的最上位或最下位元編號 (m) 轉換成 8 位元 BIN 值。



使用上的注意事項

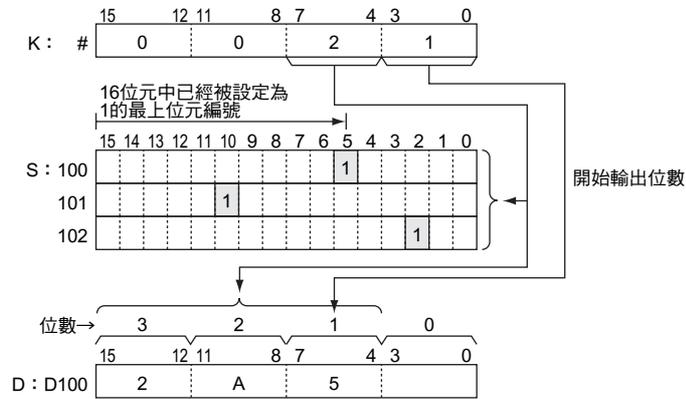
- 16 → 4 指令碼器指定時
除輸出對象外，其他資料皆不會改變。利用指令碼位數編號 (K) 來指定多位數轉換時，轉換對象的順序為 S ~ 上位 CH 端 (每個位數為 1CH)，而轉換結果則會依 D 的開始輸出位數 (K) ~ 上位位數端 (位數 3 結束後就會回到位數 0) 的順序被儲存起來。
- 256 → 8 指令碼器指定時
除輸出對象外，其他資料皆不會改變。利用指令碼位數編號 (K) 來指定多位數轉換時，轉換對象的順序為 S ~ 上位 CH 端 (每個位數為 16CH)，而轉換結果則會依 D 的開始輸出位數 (K) ~ 上位位數端 (位數 1 結束後就會回到位數 0) 的順序被儲存起來。

程式例



• 16 → 4編碼器

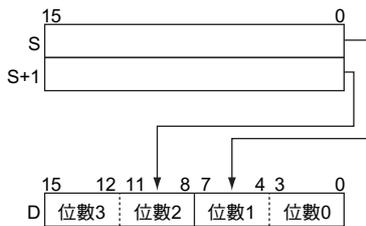
當0.00變成ON時，在100CH ~ 102CH共3CH的16位元中，其中已經被設定為1的最上位元編號會以16進位的方式儲存在D100的1 ~ 3位數。



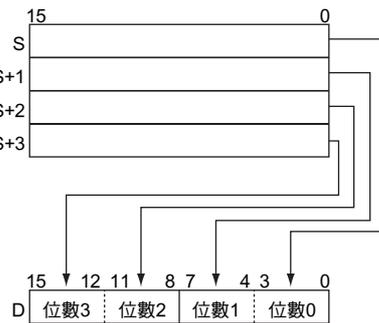
● 多位數的指令碼範例

• 16 → 4 指令碼範例

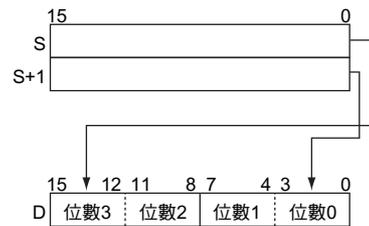
K : #0011



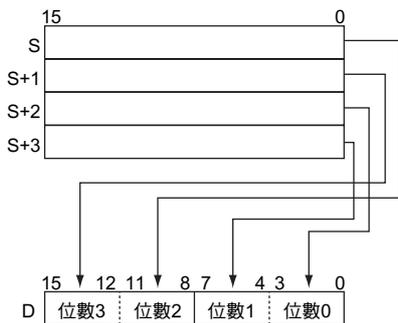
K : #0030



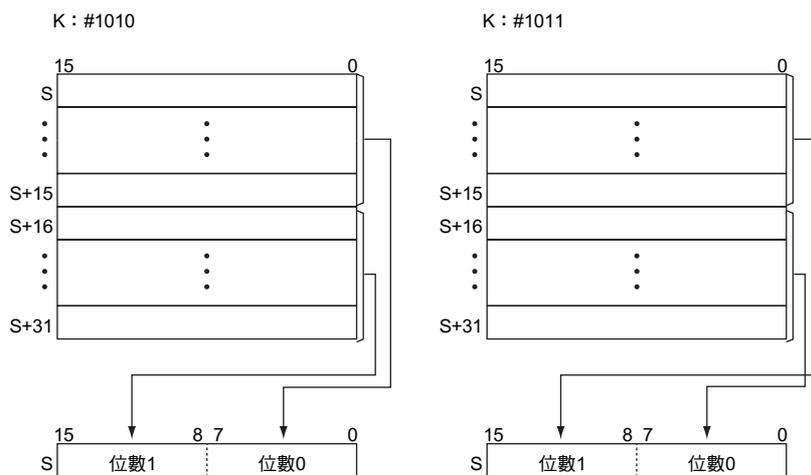
K : #0013



K : #0032



• 256 → 8 指令碼範例



當轉換資料中包含 0 時，也希望能夠對其他資料進行指令碼的話，就必須分割多個 DMPX 指令。

(例) 使用指定 4 位數的 16 → 4 指令碼方式時

DMPX D0 D100 #0030

↓

DMPX D0 D100 #0000

DMPX D1 D100 #0001

DMPX D2 D100 #0002

DMPX D3 D100 #0003

ASC

指令名稱	指令記號	指令的各種組合	Fun No.	功能
ASCII 碼轉換	ASC	@ASC	086	將 16 位元資料的指定位數轉換為 ASCII 碼。

符號	ASC	
		<p>S: 轉換資料的CH編號</p> <p>K: 指定位數的資料</p> <p>D: 輸出轉換結果的下位CH編號</p>

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

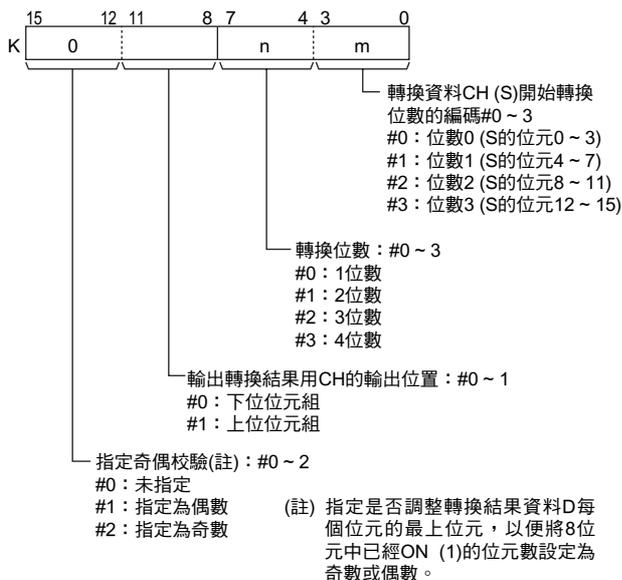
運算元種類的說明

運算元種類	內容	資料型態	容量
S	轉換資料的 CH 編號	UINT	1
K	指定位數的資料	UINT	1
D	輸出轉換結果的下位 CH 編號	UINT	可調整

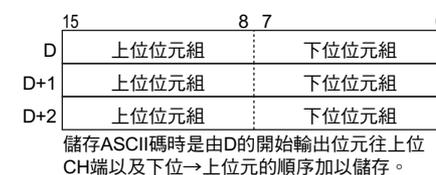
S：轉換資料的 CH 編號



K：指定位數的資料



D：輸出轉換結果的下位 CH 編號



(註) D~ 最大 D+2 的運算元種類必須相同。

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S										—			
K	○	○	○	○	○	○	○	○	○	○	—	—	—
D										—			

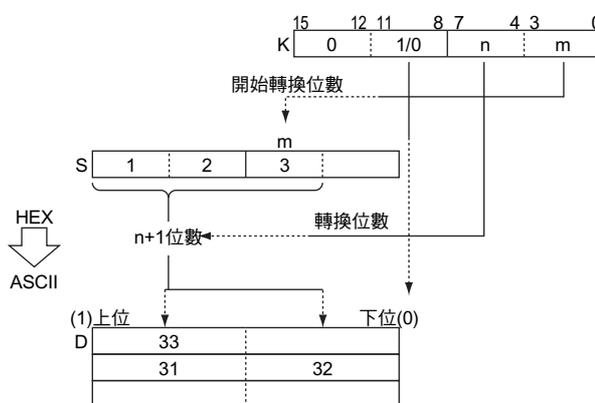
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 K 的資料超過範圍時變為 ON 否則為 OFF

功能

以 S 作為 4 位數的 HEX 資料，將開始轉換位數編號 (K 的 m) 及轉換位數 (K 的 n) 所指定的位數資料 (16 進位表示時為 #0 ~ F) 轉換為 8 位元的 ASCII 碼資料 (「0」(16 進位表示時為 #30) ~ 「9」(16 進位表示時為 #39)、「A」(16 進位表示時為 #41) ~ 「F」(16 進位表示時為 #46))，然後再將結果輸出至 D、K 所指定的輸出位置 (由上位或下位開始儲存)。

又，ASCII 碼資料的最上位元可指定奇偶校驗 (K 的位元 12 ~ 15)，因此可轉換為奇同位元或偶同位元 (可將 8 位元中的某 1 個位元數調整為奇數或偶數)。



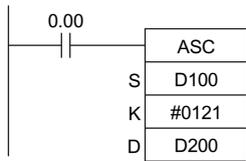
提示

- 所謂同位元 (Parity bit) 就是為了檢測資料傳送時的錯誤所附加在資料中的位元。附加該位元後，資料中的某個個數就會經常被當作奇數或偶數傳送，只要接收端的某個個數並非奇數或偶數時，即可發現傳送時所發生的錯誤。

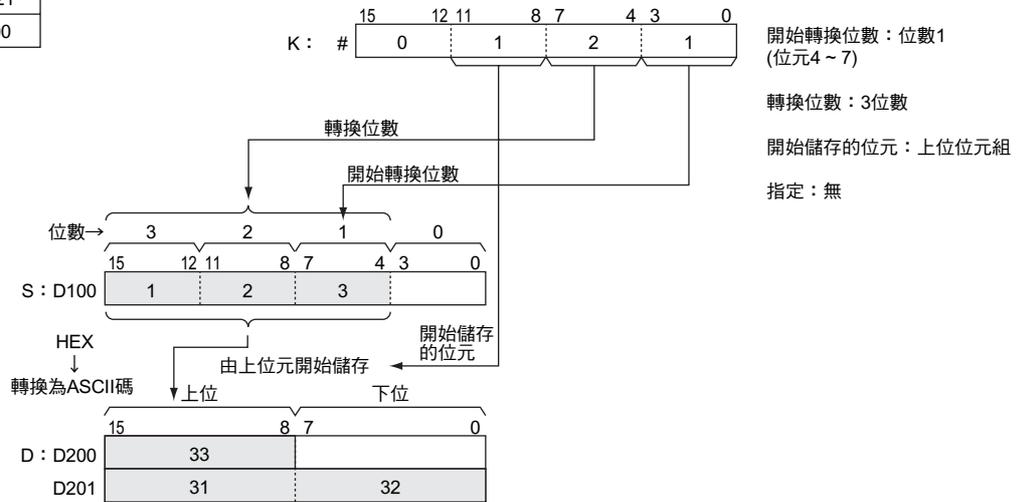
使用上的注意事項

- 利用轉換位數 (K) 來指定多位數轉換時，所要轉換的位數將依開始位數朝上位位數端的順序 (位數 3 結束後就會回到位數 0) 進行轉換，而轉換結果則依 D 的輸出位置朝上位 CH 端 (以 8 位元為單位) 的順序被儲存起來。
- 在輸出轉換結果 CH 的資料中，非輸出對象的位置資料將保持不變。
- 轉換多位數時，必須注意 D+1、D+2CH 是否超出區域。

程式例



當0.00變成ON時，D100就會被當作4位數的HEX (16進位)資料，並且根據控制資料的內容，將位數1 ~ 3的3位數轉換為ASCII碼，並且儲存在D200上位元以後的3位元。



● ASCII 換例

轉換資料的位數內容					轉換的輸出資料								
數值	位元內容				數值 ?	(MSB) 位元內容 (LSB)							
0	0	0	0	0	#30	*	0	1	1	0	0	0	0
1	0	0	0	1	#31	*	0	1	1	0	0	0	1
2	0	0	1	0	#32	*	0	1	1	0	0	1	0
3	0	0	1	1	#33	*	0	1	1	0	0	1	1
4	0	1	0	0	#34	*	0	1	1	0	1	0	0
5	0	1	0	1	#35	*	0	1	1	0	1	0	1
6	0	1	1	0	#36	*	0	1	1	0	1	1	0
7	0	1	1	1	#37	*	0	1	1	0	1	1	1
8	1	0	0	0	#38	*	0	1	1	1	0	0	0
9	1	0	0	1	#39	*	0	1	1	1	0	0	1
A	1	0	1	0	#41	*	1	0	0	0	0	0	1
B	1	0	1	1	#42	*	1	0	0	0	0	1	0
C	1	1	0	0	#43	*	1	0	0	0	0	1	1
D	1	1	0	1	#44	*	1	0	0	0	1	0	0
E	1	1	1	0	#45	*	1	0	0	0	1	0	1
F	1	1	1	1	#46	*	1	0	0	0	1	1	0

* 依同位元 --- 奇偶校驗指定方式而異。

● 指定奇偶校驗及同位元的內容

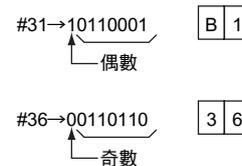
- 指定奇偶校驗 0...未指定奇偶校驗

同位元的內容將輸出 0。

- 指定奇偶校驗 1...指定偶數同位元

根據同位元的內容，當其他 7 位元中已經 ON 的位元數為奇數時，本旗標就會變成 ON，如為偶數時，本旗標 OFF，此時，8 位元中已經 ON 的位元總數將變成偶數。

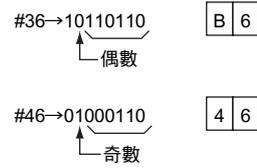
例：



• 指定奇偶校驗 2...指定奇數同位元

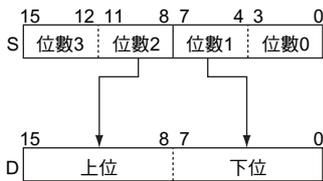
根據同位元的內容，當其他 7 位元中已經 ON 的位元數為偶數時，本旗標就會變成 ON，如為奇數時，本旗標 OFF，此時，8 位元中已經 ON 的位元總數將變成奇數。

例：

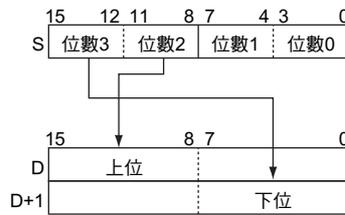


● 多位數的 ASCII 碼轉換範例

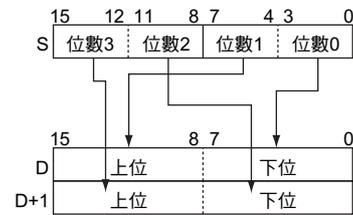
K : #0011



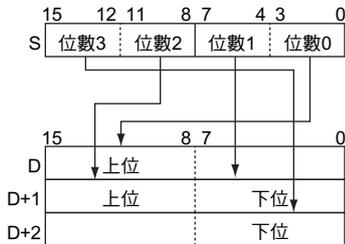
K : #0112



K : #0030



K : #0130



HEX

指令名稱	指令記號	指令的各種組合	Fun No.	功能
ASCII → HEX 轉換	HEX	@HEX	162	CH 資料的指定位數內容會被當作 8 位元的 ASCII 資料處理，並且以指定 CH 的指定位數所對應的字元輸出。

符號	HEX	
		S：轉換資料的下位CH編號 C：控制資料 D：輸出轉換結果的CH編號

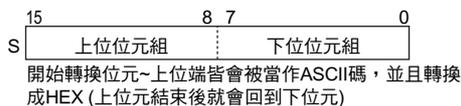
是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態	容量
S	轉換資料的下位 CH 編號	UINT	可調整
C	控制資料	UINT	1
D	輸出轉換結果的 CH 編號	UINT	1

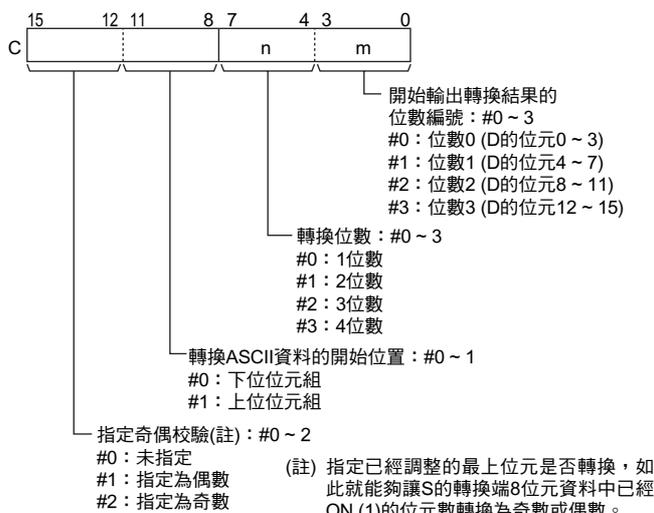
S：轉換資料的下位 CH 編號



D：輸出轉換結果的 CH 編號



C：控制資料



● 運算元種類

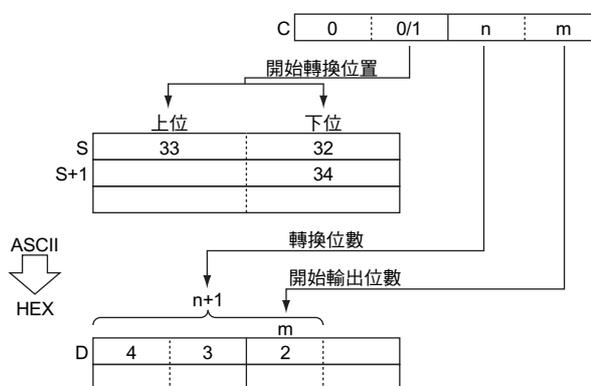
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S										—			
K	○	○	○	○	○	○	○	○	○	○	—	—	—
D										—			

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> 當 S 的 ASCII 碼資料發生奇偶校驗錯誤時，本旗標就會 ON 當 S 為無法轉換的 ASCII 碼資料時，本旗標就會 ON 當 C 的資料超過範圍時變為 ON 否則為 OFF

功能

將 S 當作上位 8 位元、下位 8 位元的 ASCII 碼資料 (「0」(16 進位表示時為 #30) - 「9」(16 進位表示時為 #39)、「A」(16 進位表示時為 #41) - 「F」(16 進 #46))，並轉換為 HEX 資料，然後再將結果輸出至 D。將開始轉換位置所指定的資料轉換為 1 位數 (4 位元) 的 HEX 資料 (16 進位表示時為 #0-F)，然後再將結果輸出至 D 所指定開始輸出的位數 (C)。利用奇指定偶校驗 (C) 的方式，將 ASCII 碼資料的最上位元當作奇同位元或偶同位元，即可開始轉換資料。



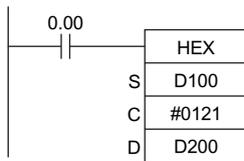
提示

- 所謂同位元 (Parity bit) 就是為了檢測資料傳送時的錯誤所附加在資料中的位元。附加該位元後，資料中的某個個數就會經常被當作奇數或偶數傳送，只要接收端的某個個數並非奇數或偶數時，即可發現傳送時所發生的錯誤。

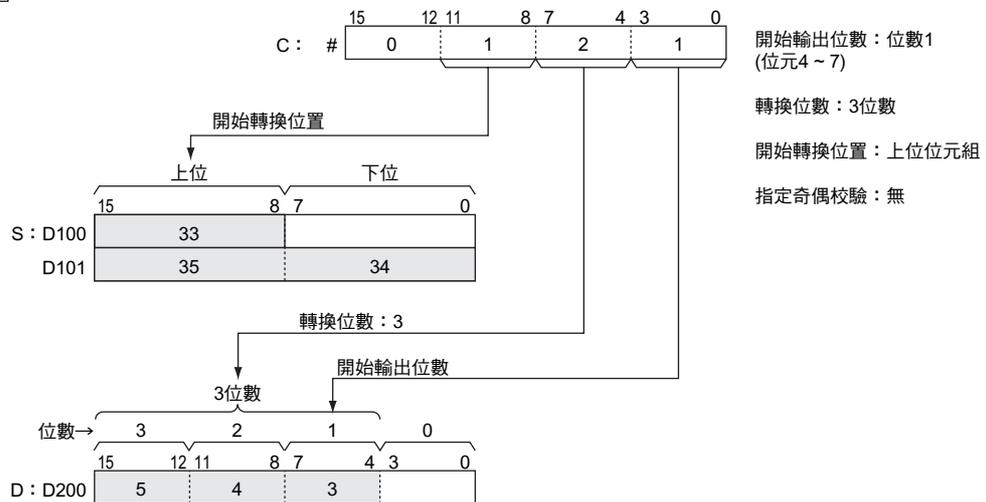
使用上的注意事項

- 利用轉換位數 (C) 來指定多位數轉換時，轉換對象會依 S 開始轉換位置往高位 CH 端的順序進行轉換，而轉換結果則會依 D 的開始輸出位數 (C) 往高位位數端的順序 (位數 3 結束後就會回到位數) 被儲存起來。
- 在輸出轉換結果 CH 的資料中，非輸出對象的位置資料將保持不變 (和先前一樣)。
- S 所指定的 ASCII 碼資料必須符合下述範圍。
 - 16 進位表示時的 #30 (ASCII 碼為 0) ~ 16 進位表示時的 #39 (ASCII 碼為 9)
 - 16 進位表示時的 #41 (ASCII 碼為 A) ~ 16 進位表示時的 #46 (ASCII 碼為 F) (同位元除外)

程式例



當0.00變成ON時，就會將D100～D101當作ASCII碼資料，並且根據#0121，將D100上位元開始的3位元(3個ASCII碼)轉換為HEX (16進位)資料，然後再將資料儲存在D200的位數1～3的3個位數中。



● 指定奇偶校驗

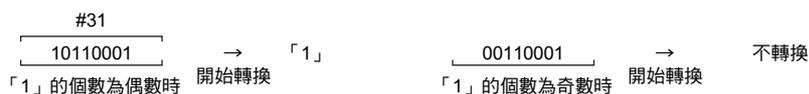
0: 未指定

只有當轉換資料的同位元為0時才會開始轉換，否則，異常旗標就會 ON，此時將不會執行轉換動作。

1: 指定為偶數

只有當轉換資料 (8 位元) 的「1」的個數為偶數時才會開始轉換。當「1」的個數為奇數時，異常旗標就會 ON，此時將不會執行轉換動作。

例：



2: 指定為奇數

只有當轉換資料 (8 位元) 的「1」的個數為奇數時才會開始轉換。當「1」的個數為偶數時，異常旗標就會 ON，此時將不會執行轉換動作。

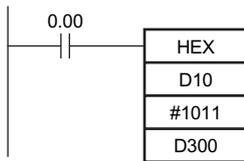
例：



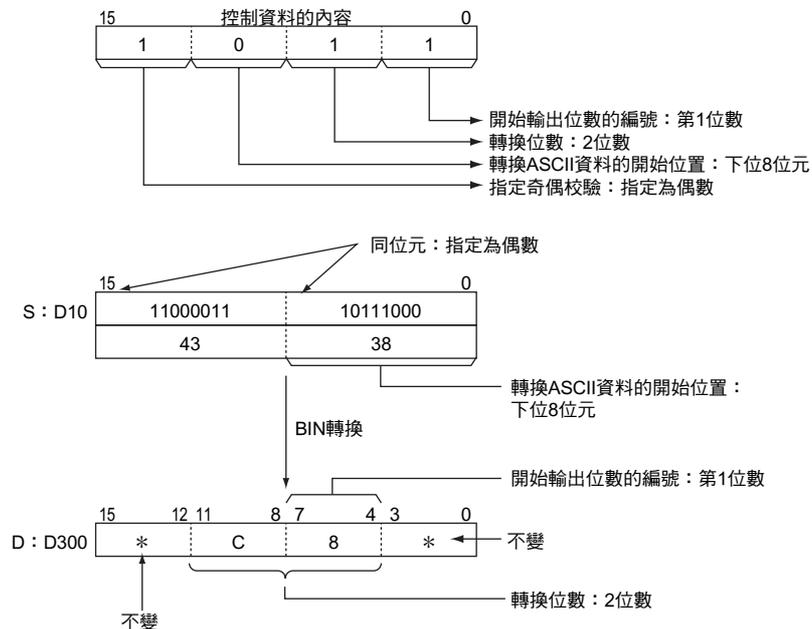
轉換範例

ASCII 碼	轉換資料								轉換結果 (Hex 資料)				
	(MSB)	位元內容						(LSB)	數值	位元內容			
\$30	*	0	1	1	0	0	0	0	0	0	0	0	0
\$31	*	0	1	1	0	0	0	1	1	0	0	0	1
\$32	*	0	1	1	0	0	1	0	2	0	0	1	0
\$33	*	0	1	1	0	0	1	1	3	0	0	1	1
\$34	*	0	1	1	0	1	0	0	4	0	1	0	0
\$35	*	0	1	1	0	1	0	1	5	0	1	0	1
\$36	*	0	1	1	0	1	1	0	6	0	1	1	0
\$37	*	0	1	1	0	1	1	1	7	0	1	1	1
\$38	*	0	1	1	1	0	0	0	8	1	0	0	0
\$39	*	0	1	1	1	0	0	1	9	1	0	0	1
\$41	*	1	0	0	0	0	0	1	A	1	0	1	0
\$42	*	1	0	0	0	0	1	0	B	1	0	1	1
\$43	*	1	0	0	0	0	1	1	C	1	1	0	0
\$44	*	1	0	0	0	1	0	0	D	1	1	0	1
\$45	*	1	0	0	0	1	0	1	E	1	1	1	0
\$46	*	1	0	0	0	1	1	0	F	1	1	1	1

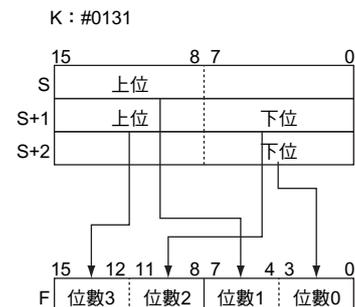
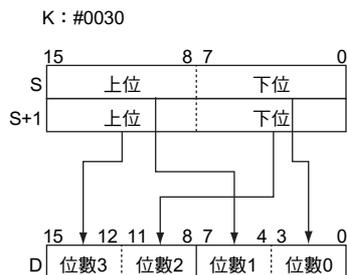
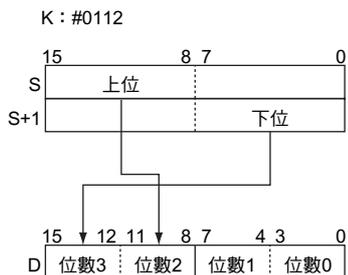
* 依同位元 ... 奇偶校驗指定方式而異。



當0.00變成ON時，就會依照控制資料的內容，將資料記憶體D10的ASCII碼資料轉換為BIN資料，然後再將結果輸出至D300。



● 多位元由 ASCII 轉換為 HEX 資料的範例



邏輯演算指令

ANDW/ANDL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
字元邏輯積	ANDW	@ANDW	034	求出以 CH 資料為單位的 CH 資料或是常數的邏輯積。
字元倍精確度邏輯積	ANDL	@ANDL	610	求出 2CH 的 CH 資料、CH 資料或常數的邏輯積。

符號	ANDW	ANDL							
	 <table border="1" style="margin-left: 20px;"> <tr><td>ANDW</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：演算資料1 S2：演算資料2 D：演算結果的輸出CH編號</p>	ANDW	S1	S2	D	 <table border="1" style="margin-left: 20px;"> <tr><td>ANDL</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：演算資料1下位CH編號 S2：演算資料2下位CH編號 D：演算結果的輸出下位CH編號</p>	ANDL	S1	S2
ANDW									
S1									
S2									
D									
ANDL									
S1									
S2									
D									

是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

運算元種類的說明

運算元種類	內容	資料型態		容量	
		ANDW	ANDL	ANDW	ANDL
S1	ANDW：演算資料 1 ANDL：演算資料 1 下位 CH 編號	WORD	DWORD	1	2
S2	ANDW：演算資料 2 ANDL：演算資料 2 下位 CH 編號	WORD	DWORD	1	2
D	演算結果的輸出 CH 編號	WORD	DWORD	1	2

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
=旗標	P_EQ	<ul style="list-style-type: none"> 當演算結果為 0 時，本旗標 ON 否則為 OFF
負旗標	P_N	<ul style="list-style-type: none"> 根據演算結果，當最上位元為 1 時，本旗標就會 ON 否則為 OFF

功能

● ANDW

求出 S1 和 S2 所指定的資料的邏輯積，然後再將結果輸出至 D。

S1 · S2 → D

S1	S2	D
1	1	1
1	0	0
0	1	0
0	0	0

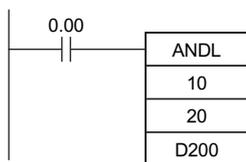
● ANDL

將 S1 和 S2 所指定的資料當作倍精確度資料，求出其邏輯積，然後再將結果輸出至 +1、D。

(S1+1, S1) · (S2+1, S2) → (D+1, D)

S1+1, S1	S2+1, S2	D+1, D
1	1	1
1	0	0
0	1	0
0	0	0

程式例



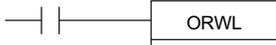
當0.00變成ON時，分別求出11、10CH及21、20CH所對應位元的邏輯積，然後再將結果輸出至資料記憶體D201、D200所對應的位元。

S1 : 10CH S1+1 : 11CH		S2 : 20CH S2+1 : 21CH		D : D200 D+1 : D201	
10.00	0	20.00	1	00	0
10.01	1	20.01	1	01	1
10.02	0	20.02	0	02	0
10.03	1	20.03	0	03	0
10.04	0	20.04	1	04	0
~~~~~		~~~~~		~~~~~	
11.13	1	21.13	1	13	1
11.14	1	21.14	0	14	0
11.15	0	21.15	0	15	0

(註) 符號代表邏輯積。

# ORW/ORWL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
字元邏輯和	ORW	@ORW	035	求出以 CH 資料為單位的 CH 資料或是常數的邏輯和。
字元倍精確度邏輯和	ORWL	@ORWL	611	求出 2CH 的 CH 資料、CH 資料或常數的邏輯和。

符號	ORW	ORWL							
	 <table border="1" style="margin-left: 20px;"> <tr><td>ORW</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：演算資料1 S2：演算資料2 D：演算結果的輸出CH編號</p>	ORW	S1	S2	D	 <table border="1" style="margin-left: 20px;"> <tr><td>ORWL</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：演算資料1下位CH編號 S2：演算資料2下位CH編號 D：演算結果的輸出下位CH編號</p>	ORWL	S1	S2
ORW									
S1									
S2									
D									
ORWL									
S1									
S2									
D									

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態		容量	
		ORW	ORWL	ORW	ORWL
S1	ORW：演算資料 1 ORWL：演算資料 1 下位 CH 編號	WORD	DWORD	1	2
S2	ORW：演算資料 2 ORWL：演算資料 2 下位 CH 編號	WORD	DWORD	1	2
D	演算結果的輸出 CH 編號	WORD	DWORD	1	2

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
=旗標	P_EQ	<ul style="list-style-type: none"> <li>當演算結果為 0 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
負旗標	P_N	<ul style="list-style-type: none"> <li>根據演算結果，當最上位元為 1 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

### ● ORW

求出 S1 和 S2 所指定的資料的邏輯和，然後再將結果輸出至 D。

S1+S2→D

S1	S2	D
1	1	1
1	0	1
0	1	1
0	0	0

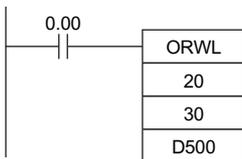
### ● ORWL

將 S1 和 S2 所指定的資料當作倍精確度資料，求出其邏輯和，然後再將結果輸出至 +1、D。

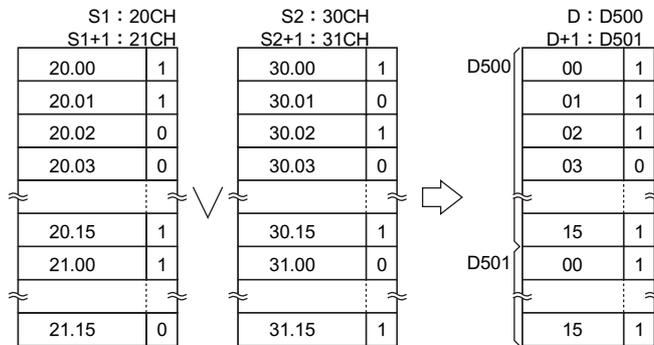
(S1+1, S1)+(S2+1, S2)→(D+1, D)

S1+1, S1	S2+1, S2	D+1, D
1	1	1
1	0	1
0	1	1
0	0	0

## 程式例



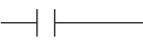
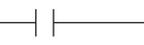
當0.00變成ON時，分別求出21、20CH及31、30CH所對應位元的邏輯和，然後再將結果輸出至資料記憶體D501、D500所對應的位元。



(註) 符號代表邏輯和。

# XORW/XORL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
字元互斥或(Exclusive OR : XOR) 演算	XORW	@XORW	036	求出以 CH 資料為單位的 CH 資料或是常數的 XOR 演算。
字元倍精確度 XOR	XORL	@XORL	612	針對 2CH 的 CH 資料，進行 CH 資料或常數的 XOR 演算。

符號	XORW	XORL							
	 <table border="1" style="margin-left: 20px;"> <tr><td>XORW</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1 : 演算資料1 S2 : 演算資料2 D : 演算結果的輸出CH編號</p>	XORW	S1	S2	D	 <table border="1" style="margin-left: 20px;"> <tr><td>XORL</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1 : 演算資料1下位CH編號 S2 : 演算資料2下位CH編號 D : 演算結果的輸出下位CH編號</p>	XORL	S1	S2
XORW									
S1									
S2									
D									
XORL									
S1									
S2									
D									

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態		容量	
		XORW	XORL	XORW	XORL
S1	XORW : 演算資料 1 XORL : 演算資料 1 下位 CH 編號	WORD	DWORD	1	2
S2	XORW : 演算資料 2 XORL : 演算資料 2 下位 CH 編號	WORD	DWORD	1	2
D	XORW : 演算結果的輸出 CH 編號 XORL : 演算結果的輸出下位 CH 編號	WORD	DWORD	1	2

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當演算結果為 0 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
負旗標	P_N	<ul style="list-style-type: none"> <li>根據演算結果，當最上位元為 1 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

功能

● XORW

求出 S1 和 S2 所指定的資料的 XOR，然後再將結果輸出至 D。

$$S1 \cdot \overline{S2} + \overline{S1} \cdot S2 \rightarrow D$$

S1	S2	D
1	1	0
1	0	1
0	1	1
0	0	0

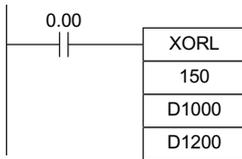
● XORL

將 S1 和 S2 所指定的資料當作倍精確度資料，求出其 XOR，然後再將結果輸出至 +1、D。

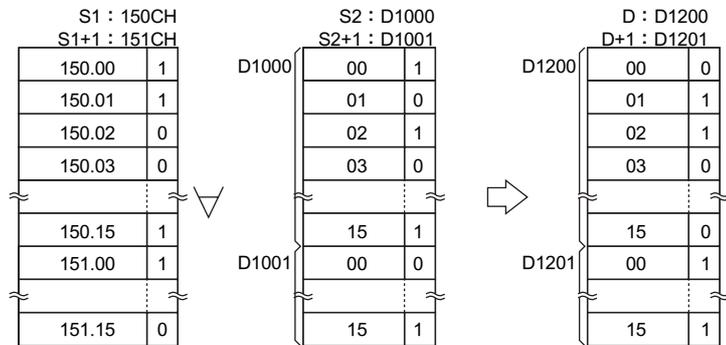
$$(S1+1, S1) \cdot \overline{(S2+1, S2)} + \overline{(S1+1, S1)} \cdot (S2+1, S2) \rightarrow (D+1, D)$$

S1+1, S1	S2+1, S2	D+1, D
1	1	0
1	0	1
0	1	1
0	0	0

程式例



當0.00變成ON時，針對151、150CH及資料記憶體D1001、D1000所對應的位元進行互斥或(XOR)演算，然後再將結果輸出至資料記憶體D1201、D1200所對應的位元。



(註)▽符號代表XOR。

# COM/COML

指令名稱	指令記號	指令的各種組合	Fun No.	功能
位元反相	COM	@COM	029	將 CH 資料進行位元反相。
位元倍精確度反相	COML	@COML	614	將 2CH 的 CH 資料之各位元進行反相輸出。

符號	COM	COML

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態		容量	
		COM	COML	COM	COML
D	COM: 反相資料的 CH 編號 COML: 反相資料的下位 CH 編號	WORD	DWORD	1	2

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
D	○	○	○	○	○	○	○	○	○	○	-	-	-

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
=旗標	P_EQ	<ul style="list-style-type: none"> <li>當演算結果為 0 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
負旗標	P_N	<ul style="list-style-type: none"> <li>根據演算結果，當最上位元為 1 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

### ● COM

將 D 所指定資料的每個位元反相。

$$\overline{D} \rightarrow D \quad \left( \begin{array}{l} 1 \rightarrow 0 \\ 0 \rightarrow 1 \end{array} \right)$$

(註) 請注意：當您使用 COM 指令時，只要輸入變成 ON，每個週期就會執行一次本指令。

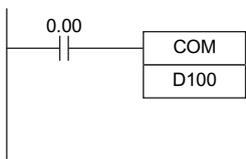
### ● COML

將 D 所指定的資料當作倍精確度資料，然後再將每個位元反相。

$$\overline{(D+1, D)} \rightarrow (D+1, D)$$

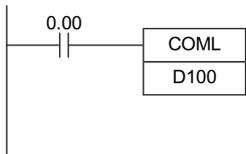
(註) 請注意：當您使用 COML 指令時，只要輸入變成 ON，每個週期就會執行一次本指令。

## 程式例



當 0.00 變成 ON 時，就將 D100 的所有位元反轉。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D100	1	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
↓																
D100	0	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0



當 0.00 變成 ON 時，就將 D101 及 D100 的所有位元反轉。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D101	1	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	D101	1	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
↓																	↓																
D100	0	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	D100	0	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0

# 特殊演算指令

## APR

指令名稱	指令記號	指令的各種組合	Fun No.	功能
數值轉換	APR	@APR	069	以線性近似 (linear approximation) 法演算。

符號	APR					
		<table border="1"> <tr><td>APR</td></tr> <tr><td>C</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> </table>	APR	C	S	D
APR						
C						
S						
D						

### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 運算元種類的說明

運算元種類	內容	資料型態	容量
C	控制資料	UINT	可調整
S	數值轉換後的輸入資料	WORD	1
D	數值轉換輸出的 CH 編號	WORD	1

#### ● 指定為 SIN 計算時

運算元種類	值	資料範圍
C	&0 或 #0000	—
S	BCD#0000 ~ 0900	0° ~ 90°
D	BCD#0000 ~ 9999 BCD#9999	0.0000 ~ 0.9999 1.0000

- 執行 SIN 計算 (C = 0) 時：  
根據 S 所指定的角度資料 (單位為  $\times 10^{-1}$  的 BCD 資料) 0000 ~ 0900 (0° ~ 90°) 計算其所對應的 SIN，然後再將結果當作小數點後 4 位數的 BCD 資料 0000 ~ 9999 (0.0000 ~ 0.9999) 輸出至 D。  
小數點後的第 5 位數以後全部捨去。

#### ● COS 計算指定時

運算元種類	值	資料範圍
C	&1 或 #0001	—
S	BCD#0000 ~ 0900	0° ~ 90°
D	BCD#0000 ~ 999 BCD#9999	0.0000 ~ 0.9999 1.0000

- 指定為 COS 計算 (C = 1) 時：  
根據 S 所指定的角度資料 (單位為  $\times 10^{-1}$  的 BCD 資料) 0000 ~ 0900 (0° ~ 90°) 計算其所對應的 COS，然後再將結果當作小數點後 4 位數的 BCD 資料 0000 ~ 9999 (0.0000 ~ 0.9999) 輸出至 D。  
小數點後的第 5 位數以後全部捨去。

(註) SIN 90° 及 COS 0° 所對應的演算結果為 1.0000，因此輸出 9999 (= 0.9999)。

## ● 指定以線性近似法演算時

運算元種類	值	資料範圍
C	CH 編號	—
S	不帶符號的 BCD 16 位元資料	#0000 ~ 9999
	不帶符號的 BIN 16 位元資料	&0000 ~ 65535
	帶符號的 BIN 16 位元資料	— 32768 ~ 32767
	帶符號的 BIN 32 位元資料	— 2147483648 ~ 2147483647
	單精確度的浮動小數點資料	— ∞、 — $3.402823 \times 10^{38}$ ~ — $1.175494 \times 10^{-38}$ 、 + $1.175494 \times 10^{-38}$ ~ + $3.402823 \times 10^{38}$ 、 + ∞
D	不帶符號的 BCD 16 位元資料	#0000 ~ 9999
	不帶符號的 BIN 16 位元資料	&0000 ~ 65535
	帶符號的 BIN 16 位元資料	— 32768 ~ 32767
	帶符號的 BIN 32 位元資料	— 2147483648 ~ 2147483647
	單精確度的浮動小數點資料	— ∞、 — $3.402823 \times 10^{38}$ ~ — $1.175494 \times 10^{-38}$ 、 + $1.175494 \times 10^{-38}$ ~ + $3.402823 \times 10^{38}$ 、 + ∞

## 線性近似法演算

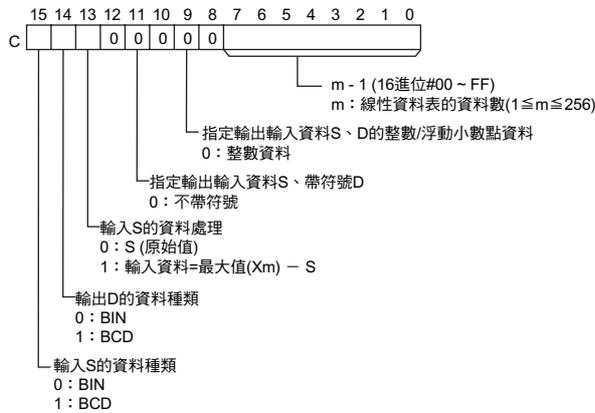
如果有 CH 位址輸入 C 時，C 就會被指定為以線性近似法演算。

針對以輸入 C 的 CH 位址作為起點的線性資料表，指定資料數或是輸出輸入資料的種類 (BCD/BIN) 等。

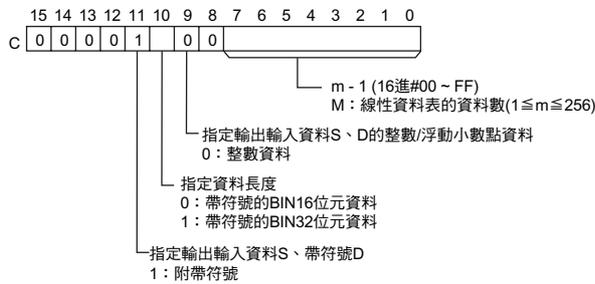
適用於輸出輸入資料的資料種類有下列 5 種：

- 不帶符號的 BCD 16 位元資料
- 不帶符號的 BIN 16 位元資料
- 帶符號的 BIN 16 位元資料
- 帶符號的 BIN 32 位元資料
- 單精確度的浮動小數點資料

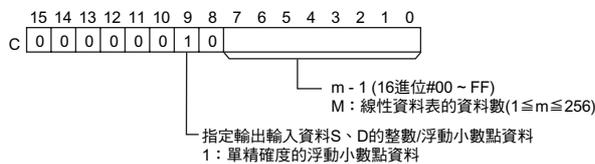
· 指定不帶符號的整數資料(BIN或BCD)時



· 指定帶符號的整數資料(BIN)時



· 指定單精確度的浮動小數點資料時



· 當資料為BIN 16位元資料、帶符號/不帶符號的BIN 16位元資料時

C+1	X0(*1)
C+2	Y0
C+3	X1
C+4	Y1
C+5	X2
C+6	Y2
~	~
~	~
~	~
C+(2m+1)	Xm
C+(2m+2)	Ym

*1: 指定輸出輸入資料S、帶符號的D後，即可儲存指定為不帶符號時(當C的位元11=0時)的Xm(資料資料表中的X資料最大值)

當資料為帶符號的BIN 16位元資料時

C+1	X0(下位16位元)
C+2	X0(上位16位元)
C+3	Y0(下位16位元)
C+4	Y0(上位16位元)
C+5	X1(下位16位元)
C+6	X1(上位16位元)
C+7	Y1(下位16位元)
C+8	Y1(上位16位元)
~	~
~	~
C+(4n+1)	Xn(下位16位元)
C+(4n+2)	Xn(上位16位元)
C+(4n+3)	Yn(下位16位元)
C+(4n+4)	Yn(上位16位元)
~	~
~	~
C+(4m+1)	Xm(下位16位元)
C+(4m+2)	Xm(上位16位元)
C+(4m+3)	Ym(下位16位元)
C+(4m+4)	Ym(上位16位元)

若為單精確度浮動小數點資料時

C+1	X0(下位16位元)
C+2	X0(上位16位元)
C+3	Y0(下位16位元)
C+4	Y0(上位16位元)
C+5	X1(下位16位元)
C+6	X1(上位16位元)
C+7	Y1(下位16位元)
C+8	Y1(上位16位元)
~	~
~	~
C+(4n+1)	Xn(下位16位元)
C+(4n+2)	Xn(上位16位元)
C+(4n+3)	Yn(下位16位元)
C+(4n+4)	Yn(上位16位元)
~	~
~	~
C+(4m+1)	Xm(下位16位元)
C+(4m+2)	Xm(上位16位元)
C+(4m+3)	Ym(下位16位元)
C+(4m+4)	Ym(上位16位元)

(註)假設X1 < X2 < Xm...

無論控制資料的輸出輸入指定為何，線性資料(Xm、Ym)皆會被儲存為BIN資料。

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C,S	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	○	-	-	-

條件旗標的動作

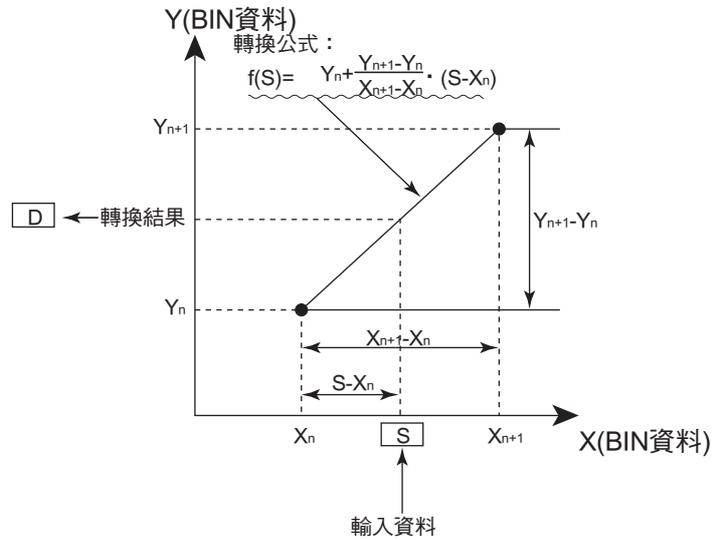
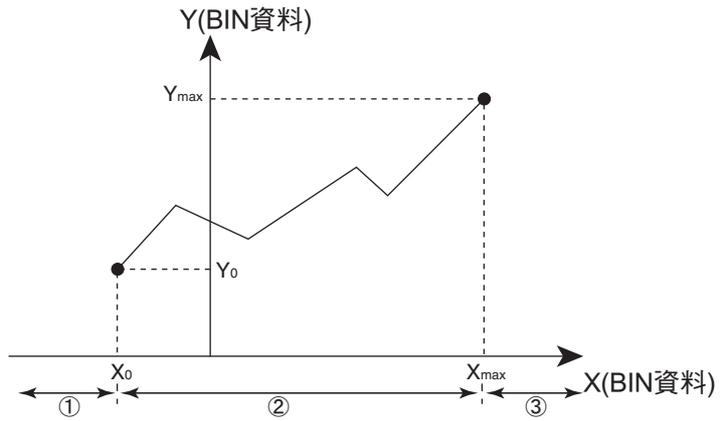
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 C 的資料超出 0 ~ 1 的範圍時，本旗標就會 ON( 指定為常數時 )</li> <li>當 C 的線性資料不符合 $X1 \leq X2 \leq \dots \leq Xm$ 的條件時，本旗標就會 ON( 指定以線性近似法演算時 )</li> <li>當 S 的資料非 BCD 時，本旗標就會 ON( 指定以線性近似法演算且輸入種類為 BCD 時 )</li> <li>當 S 的資料被指定為 1 字元的常數時( 指定以線性近似法演算而且指定輸出輸入資料S、D 為浮動小數點資料時 )</li> <li>當 S 的資料非 0000 ~ 0900 的 BCD 資料時，本旗標就會 ON( 指定為 SIN/COS 演算時 )</li> <li>否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當轉換結果為 0 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
負旗標	P_N	<ul style="list-style-type: none"> <li>當 D 的資料的最上位元為 1 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

功能

- 指定以線性近似法演算時(指定為C = CH) :

根據下列轉換公式，利用 C 所指定的線性資料 (Xn、Yn)，為 S 所指定的輸入資料進行近似演算，然後再將結果輸出至 D 所指定的 CH。

- ①當  $S < X_0$  時 轉換結果 =  $Y_0$
  - ②當  $X_0 \leq S \leq X_{max}$  時，假設  $X_n < S < X_{n+1}$ ，  
則轉換結果 =  $Y_n + \{ \frac{Y_{n+1} - Y_n}{X_{n+1} - X_n} \} \times \{ \text{輸入資料 } S - X_n \}$
  - ③當  $X_{max} < S$  時 轉換結果 =  $Y_{max}$
- 又，線性資料表最多可儲存 256 個資料。



使用各類型的資料時，必須先執行下列設定。

### ● 當資料為不帶符號的 BCD 16 位元資料時

BCD 資料適用於輸入資料、輸出資料的其中任一項或兩者皆適用。

而且，您還可以選擇「直接使用 S 所指定的值」或是「 $X_m - S$ 」（由線性資料 X 的最大值 -S 所指定的數值）作為輸入資料。

設定項目	線性資料起始 CH 的位元	設定值
指定輸入資料 S 的 BIN/BCD	15	0 : BIN 1 : BCD
指定輸出資料 D 的 BIN/BCD	14	0 : BIN 1 : BCD
輸入資料 S 的資料處理	13	0 : 原始值 1 : 輸入資料 = $X_m - S$
指定輸出輸入資料 S、帶符號 D	11	0 : 不帶符號
當資料為輸出輸入資料 S、帶符號 D 時指定其資料長度	10	無效 (資料長度固定為 16 位元)
指定輸出輸入資料 S、D 的整數 / 浮動小數點資料	09	0 : 整數資料

### ● 當資料為不帶符號的 BIN 16 位元資料時

BIN 資料適用於輸入資料、輸出資料的其中任一項或兩者皆適用。

而且，您還可以選擇「直接使用 S 所指定的值」或是「 $X_m - S$ 」（由線性資料 X 的最大值 -S 所指定的數值）作為輸入資料。

設定項目	線性資料起始 CH 的位元	設定值
輸入資料 S 的資料種類	15	0 : BIN 1 : BCD
輸出資料 D 的資料種類	14	0 : BIN 1 : BCD
輸入資料 S 的資料處理	13	0 : 原始值 1 : 輸入資料 = $X_m - S$
指定輸出輸入資料 S、帶符號 D	11	0 : 不帶符號
當資料為輸出輸入資料 S、帶符號 D 時指定其資料長度	10	無效 (資料長度固定為 16 位元)
指定輸出輸入資料 S、D 的整數 / 浮動小數點資料	09	0 : 整數資料

### ● 當資料為帶符號的 BIN 16 位元資料時

設定項目	線性資料起始 CH 的位元	設定值
輸入資料 S 的資料種類	15	0 : BIN
輸出資料 D 的資料種類	14	0 : BIN
輸入資料 S 的資料處理	13	0
指定輸出輸入資料 S、帶符號 D	11	1 : 帶符號
當資料為輸出輸入資料 S、帶符號 D 時指定其資料長度	10	0 : 帶符號 (BIN16 位元資料)
指定輸出輸入資料 S、D 的整數 / 浮動小數點資料	09	0 : 整數資料

### ● 當資料為帶符號的 BIN 32 位元資料時

設定項目	線性資料起始 CH 的位元	設定值
輸入資料 S 的資料種類	15	0 : BIN
輸出資料 D 的資料種類	14	0 : BIN
輸入資料 S 的資料處理	13	0
指定輸出輸入資料 S、帶符號 D	11	1 : 帶符號
當資料為輸出輸入資料 S、帶符號 D 時指定其資料長度	10	1 : 帶符號 (BIN32 位元資料)
指定輸出輸入資料 S、D 的整數 / 浮動小數點資料	09	0 : 整數資料

(註) 利用「當資料為輸出輸入資料 S、帶符號 D 時指定其資料長度」的方式來指定帶符號的 BIN32 位元資料，同時以 16 常數作為 S 時，會將輸入資料先轉換 (符號延伸) 為 32 位元資料後再進行計算。

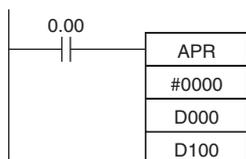
● 若為單精度浮動小數點資料時

設定項目	線性資料起始 CH 的位元	設定值
輸入資料 S 的資料種類	15	0
輸出資料 D 的資料種類	14	0
輸入資料 S 的資料處理	13	0
指定輸出輸入資料 S、帶符號 D	11	0
當資料為輸出輸入資料 S、帶符號 D 時指定其資料長度	10	0
指定輸出輸入資料 S、D 的整數 / 浮動小數點資料	09	1：浮動小數點資料

(註) 利用「指定輸出輸入資料 S、D 的整數 / 浮動小數點資料」的方式來指定單精度浮動小數點資料時，無法將輸入資料 S 指定為常數。

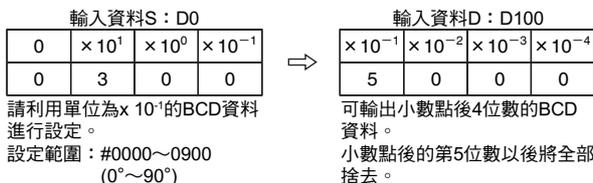
程式例

● 執行 SIN 計算 (C = 0)

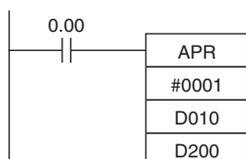


APR 指令的動作

- 請將控制資料 C 設定為 &0 或 #0000。
- (例)  $\sin 30^\circ = 0.5000$

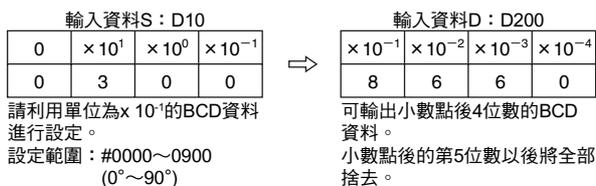


● 執行 COS 計算 (C = 1)



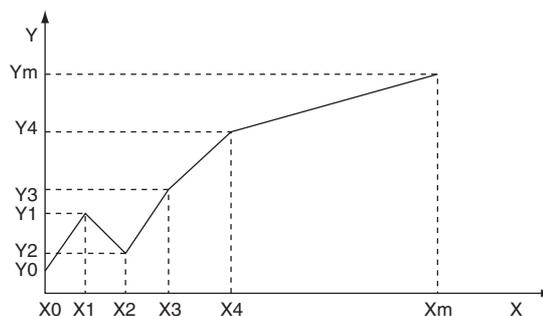
APR 指令的動作

- 請將控制資料 C 設定為 &1 或 #0001。
- (例)  $\cos 30^\circ = 0.8660$

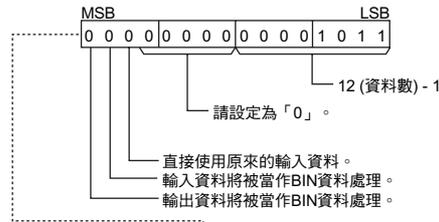
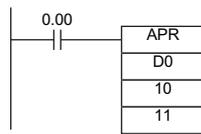
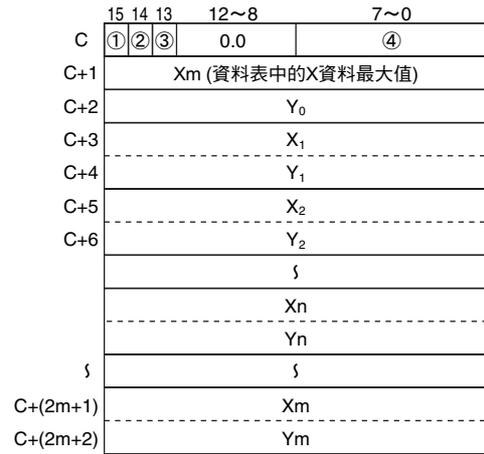


● 以線性近似法演算 (指定為 C : CH 時)

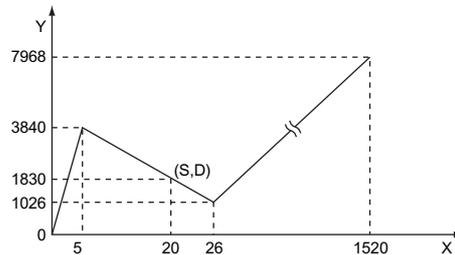
- 當資料為不帶符號的 BCD/BIN16 位元資料時  
只要在控制資料 C 所設定的 CH 以後的位置先設定好線性資料，即可根據該資料計算 SCH 輸入資料所對應的近似值，並且將結果輸出至 DCH。



- $Y_n = f(X_n)$      $Y_0 = f(X_0)$
- 請設定為  $X_{n-1} < X_n$
- 所有的  $X_n$ 、 $Y_n$  皆必須以 BIN 資料輸入。



D0	000B	
D1	&1520	..... X12
D2	&0	..... Y0
D3	&5	..... X1
D4	&3840	..... Y1
D5	&26	..... X2
D6	&1026	..... Y2
~	~	
D25	&1520	..... X12
D26	&7968	..... Y12

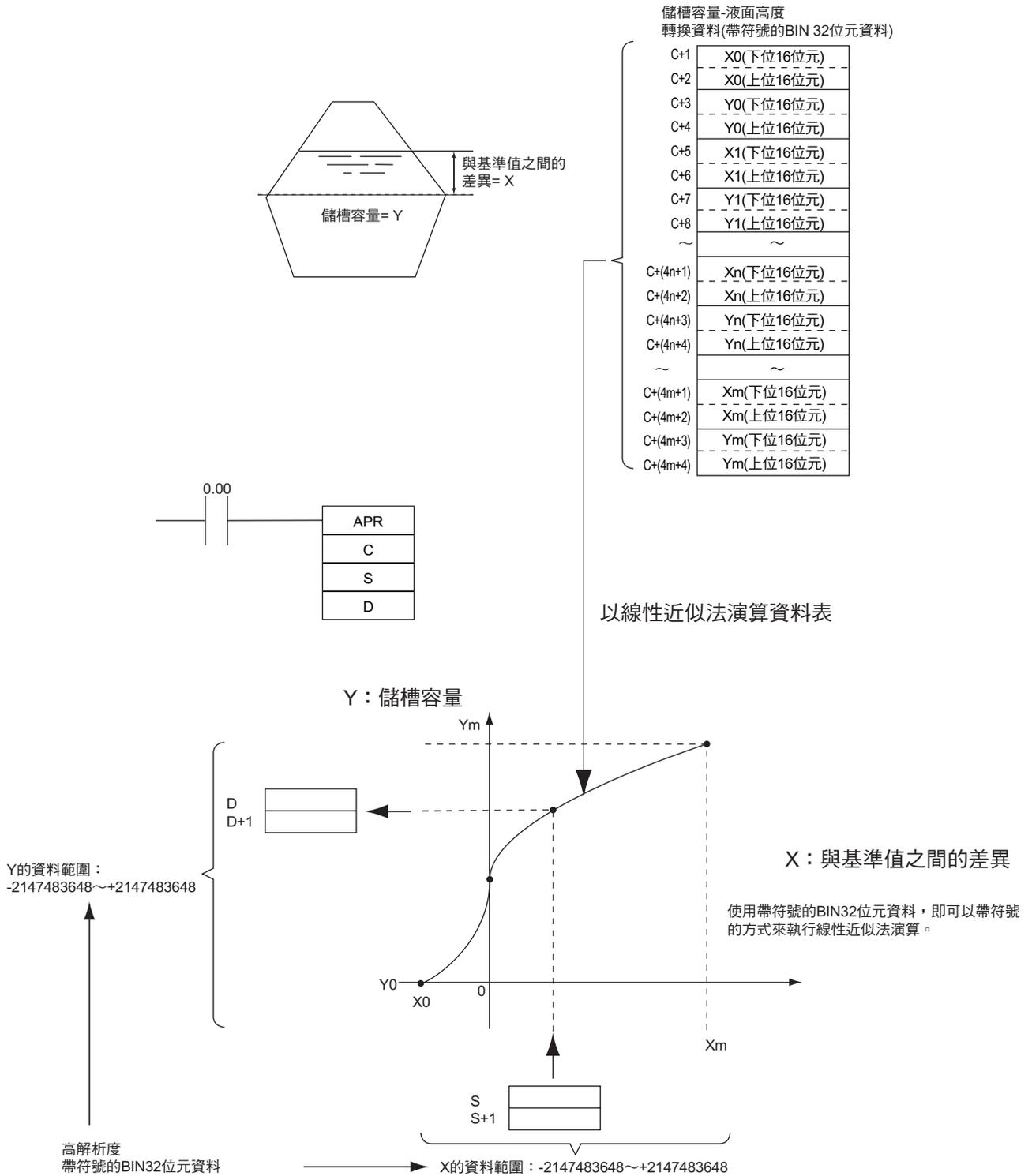


(計算公式)

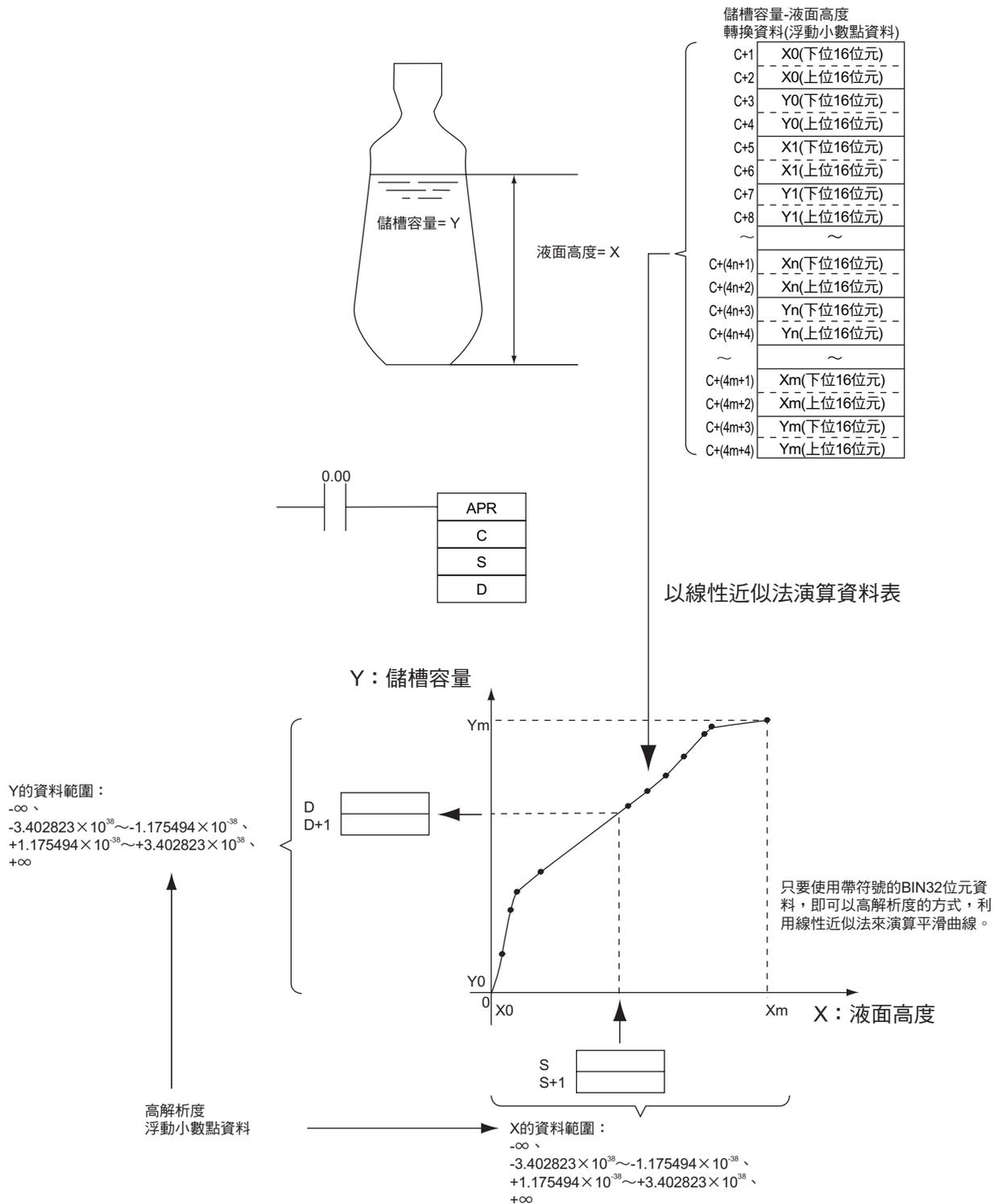
$$Y = 3840 + \frac{1026 - 3840}{26 - 5} \times (20 - 5)$$

$$= 1830$$

- 當資料為帶符號的 BIN16/32 位元資料時  
依儲槽的形狀決定液面高度→依儲槽容量而異



- 若為浮動小數點資料時  
依儲槽的形狀決定液面高度→依儲槽容量而異



# BCNT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
位元計數器	BCNT	@BCNT	067	計算 CH 資料中數值為「1」的位元總數。

符號	BCNT	
		W : 計算CH數 S : 計算下位CH編號 D : 輸出計算結果的CH編號

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
W	計算 CH 數	UINT	1
S	計算下位 CH 編號	UINT	可調整
D	輸出計算結果的 CH 編號	UINT	1

W：計算 CH 數

10 進位 & 1 ~ 65535 或 16 進位 #0001 ~ FFFF

### ● 運算元種類

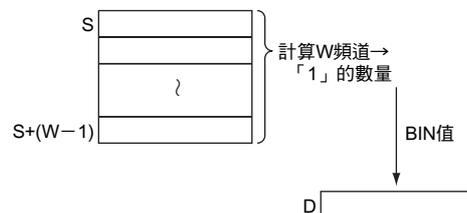
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
W	○	○	○	○	○	○	○	○	○	○	—	—	—
S,D	○	○	○	○	○	○	○	○	○	—	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 W 的資料為 0 時，本旗標就會 ON，或是當計算結果 D 的資料超過 65535 (以 16 進位表示時為 #FFFF) 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當計算結果 (D 的資料) 為 0 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

針對 S 所指定的計算下位 CH 編號 ~ 指定 CH 數 (W) 的資料，計算數值為「1」的位元總數，然後再以 BIN 值的格式，將結果輸出至 D。



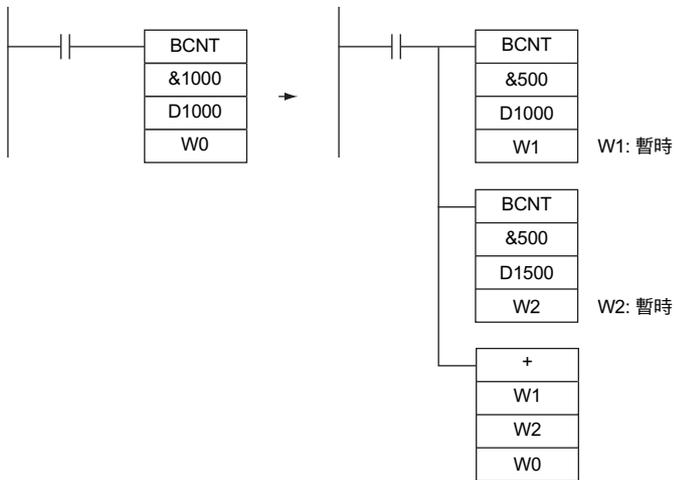
## 使用上的注意事項

如果需要計算位元的 CH 數量較多時，執行指令的時間就會愈長。

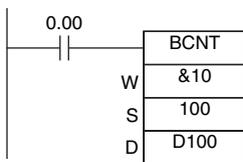
即使在本指令執行時發生配置訊號，配置任務仍然會在本指令執行完成後才開始啟動。

可藉由分割命令來避免上述問題發生。

例)



## 程式例



當0.00變成ON時，就會計算100CH ~ 10CH的資料中數值為「1」的數量，然後再以BIN值的格式將結果儲存至D100。



# 浮動小數點轉換・演算指令

## ● 資料格式

所謂浮動小數點資料就是利用符號、假數、指數等來表示實數的一種資料。利用浮動小數點的形式來表示任意資料時，將出現下列結果。

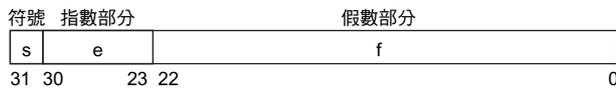
$$\text{實數值} = (-1)^s 2^{e-127} (1.f)$$

s：符號

e：指數

f：假數

浮動小數點資料的格式符合 IEEE754 所規格的單精確度，單精確度資料係以 32 位元表示，其資料格式如下：



資料	位元數	內容
s：符號	1	0：正數、1：負數
e：指數	8	指數部分 e 為數值 0 ~ 255。e 減去 127 的數值即為實際的指數值，也就是 -127 ~ 128。當 e = 0、e = 255 時，表示特殊數。
f：假數	23	表示以 2 進位表示時浮動小數點的假數部分。以正規化表示時為 $2.0 > 1.f \geq 1.0$ 。

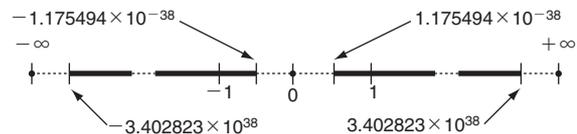
## ● 有效位數

浮動小數點資料的有效位數為 7 位數。

## ● 浮動小數點資料的表示方式

### ● 可表示的數值範圍

- $-\infty$
- $-3.402823 \times 10^{38} \leq \text{值} \leq -1.175494 \times 10^{-38}$
- 0
- $+1.175494 \times 10^{-38} \leq \text{值} \leq +3.402823 \times 10^{38}$
- $+\infty$
- 非數



### ● 特殊數

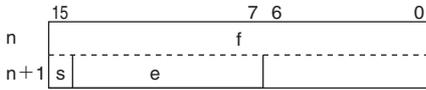
非數、 $\pm\infty$ 、0 的格式定義如下：

- |             |                     |
|-------------|---------------------|
| • 非數        | e = 255、f ≠ 0       |
| • $+\infty$ | e = 255、f = 0、s = 0 |
| • $-\infty$ | e = 255、f = 0、s = 1 |
| • 0         | e = 0、f = 0         |

所謂「非數」就是無法被當作浮動小數點資料的數值。不過，浮動小數點及演算指令的結果並不會變成非數。

### ● 當資料格式為浮點時的儲存方法

進入週邊工具 CX-Programmer 的 I/O 記憶體編輯畫面，並且將資料格式指定為「浮點」後，輸入完成的小數點資料就會自動地被轉換為符合上述 IEEE754 規格的資料格式，並且儲存在 I/O 記憶體中。而且，以符合上述 IEEE754 規格的資料格式儲存完成的資料將自動被小數點資料監控。因此，使用者不需要特別注意資料的資料格式是否符合前述的 IEEE754 規格。唯一需要注意的就是浮動小數點資料是否佔用 2CH。



### ● 參考

以浮動小數點數來表示的數值種類

指數部分 (e) \ 假數部分 (f)	0	非 0 而且非所有位元 1 (255)	所有位元 1 (255)
0	0	正規化數值	無限大
0 以外	非正規化數值		非數

(註) 非正規化數值會在正規化數值所無法表示的範圍內，以絕對值較小的浮動小數點數來表示，相較於正規化數值，其有效位數將變得更少。因此，當演算結果為非正規化數值，或是中途出現非正規化數值時，並不保證最後所得到的是有效位數，此點請特別注意。

#### (1) 正規化數

以一般的實數值來表現。

符號部分將以 0 (正數) 或 1 (負數) 來表示數值的符號。

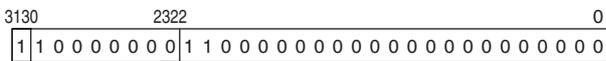
指數部分則取數值 1 ~ 254。實際的指數就是該數值減去 127 後所得到的數值，其範圍為 -126 ~ 127。

假數部分則取數值 0 ~ (2²³ - 1)。實際的假數是將 2²³ 位元假設為 1，並且當作該數值後面附加了小數點。

以正規化數值來表現的數值就是

$$(-1)^{(\text{符號部分})} \times 2^{(\text{指數部分}) - 127} \times (1 + \langle \text{假數部分} \rangle \times 2^{-23})$$

例



符號：-

指數：128 - 127 = 1

假數：1 + (2²² + 2²¹) × 2⁻²³ = 1 + (2⁻¹ + 2⁻²) = 1 + 0.75 = 1.75

值：- 1.75 × 2¹ = - 3.5

#### (2) 非正規化數值

以絕對值較小的實數值來表現。

符號部分將以 0 (正數) 或 1 (負數) 來表示數值的符號。

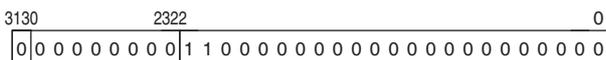
指數部分為 0，實際的指數為 -126。

假數部分為 1 ~ 2²³ - 1，實際的假數是假設 2²³ 位元為 0，並且當作該數值後面附加了小數點。

以非正規化數值來表現的數值就是

$$(-1)^{(\text{符號部分})} \times 2^{-126} \times (\langle \text{假數部分} \rangle \times 2^{-23})$$

例



符號：+

指數：- 126

假數：0 + (2²² + 2²¹) × 2⁻²³ = 0 + (2⁻¹ + 2⁻²) = 0 + 0.75 = 0.75

值：0.75 × 2⁻¹²⁶

### (3) 0

符號部分為 0 (正數) 或 1 (負數), 分別以 +0.0、-0.0 來表示。

指數與假數部分皆為 0。

將 +0.0、-0.0 皆當作一個數值, 並且以 0.0 來表示。各種演算功能依 0 的符號而異, 詳細內容請參閱「浮動小數點的四則運算」

### (4) 無限大

符號部分為 0 (正數) 或 1 (負數), 分別以  $+\infty$ 、 $-\infty$  來表示。

指數部分為 255 ( $2^8 - 1$ )。

假數部分為 0。

### (5) 非數

演算後將出現「0.0/0.0」、「 $\infty/\infty$ 」、「 $\infty - \infty$ 」等無法對應至數值或無限大之結果。

指數部分為 255 ( $2^8 - 1$ )。

假數部分為 0 以外的數值。

(註) 本手冊並未規定非數的符號及假數欄(0 以外)的數值。

## ● 浮點的四則運算

### (1) 結果數值的捨去方法

當浮動小數點數經過四則演算後所產生的正確數值超過內部所表現的假數有效數字時, 請依照下列規則來捨去數值。

① 根據內部的表現方式, 結果數值會朝其所近似的 2 個浮動小數點數中較靠近的一方捨去。

② 當結果數值刚好在其所近似的 2 個浮動小數點數的中央時, 將朝假數最後位數為 0 的方向捨去。

### (2) 溢位 (Overflow)、欠位 (Underflow)、無效演算時的處理方法

執行時一旦發生溢位、欠位或無效演算, 將採取下列處理方式。

① 溢位時, 將依結果的符號變為正或負無限大。

② 欠位時, 將依結果的符號變為正值或負值的零。

③ 無效演算時, 一旦符號被加上負無限大、符號被減去正無限大、或是乘上 0 和無限大, 就會發生以 0 除以 0, 或是以無限大除以無限大的情形。

發生上述情況時, 結果將變成非數字 (NaN)。

④ 將浮動小數點數點被轉換為整數時, 就會發生溢位的情形, 此時將無法保證一定會產生結果數值。

### (3) 演算特殊值時的相關注意事項

演算特殊值 (0、無限大、非數字) 時的相關注意事項如下:

① 正 0 和負 0 的和為正 0。

② 相同符號的 0 相減時, 結果為正 0。

③ 演算時, 只要運算元種類的任一方或是兩方皆為非數字 (NaN), 異常旗標就會 ON, 此時將不會執行動作。

④ 執行比較演算時, 正 0 和負 0 將會被視為相等的數值。

⑤ 執行比較演算時, 只要運算元種類的任一方或是兩方皆為非數字 (NaN), 異常旗標就會 ON, 此時將不會執行動作。

## ● 浮動小數點資料的演算結果

### (1) 溢位 (Overflow)

當演算結果的絕對值大於浮動小數點資料所能表現的最大值時, 就會發生溢位的情形, 此時將出現下列演算結果。

• 當演算結果為正時:  $+\infty$

• 當演算結果為負時:  $-\infty$

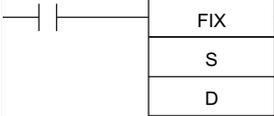
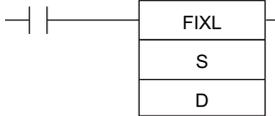
### (2) 當演算結果為 0 時

當演算結果的假數部分 f 及指數部分 e 兩者皆為 0 時, 演算結果就會變成 0。又, 當演算結果的絕對值大於浮動小數點資料所能表現的最大值時, 就會發生溢位的情形, 這時候, 演算結果也是 0。

(3) 又, 當演算結果為 0 時, 只有演算結果的假數部分 f 和指數部分 e 皆為 0 時, 已經 ON 的條件旗標 = 旗標才會變成 ON。

# FIX/FIXL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
浮動小數點→轉換為 16 位元 BIN	FIX	@FIX	450	將已指定的浮動小數點 32 位元資料轉換為帶符號的 BIN16 位元資料，然後再將結果輸出至指定的 CH。
浮動小數點→轉換為 32 位元 BIN	FIXL	@FIXL	451	將已指定的浮動小數點 32 位元資料轉換為帶符號的 BIN32 位元資料，然後再將結果輸出至指定的 CH。

符號	FIX	FIXL
	 <p>S：浮動小數點資料的下位CH編號 D：輸出轉換結果的CH編號</p>	 <p>S：浮動小數點資料的下位CH編號 D：輸出轉換結果的下位CH編號</p>

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態		容量	
		FIX	FIXL	FIX	FIXL
S	浮動小數點資料的下位 CH 編號	REAL	REAL	2	2
D	輸出轉換結果的 CH 編號	INT	DINT	1	2

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

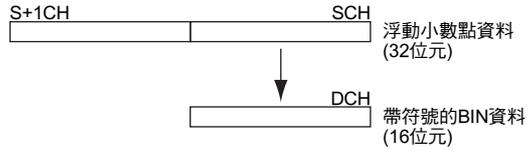
## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>FIX</li> <li>當 S+1、S 的資料超出 -32768 ~ +32767 時，本旗標 ON</li> <li>FIXL</li> <li>當 S+1、S 的資料超出 -2147483648 ~ +2147483647 時，本旗標 ON</li> <li>當 S+1、S 的資料為非數字 (NaN) 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當轉換結果為 0 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
負旗標	P_N	<ul style="list-style-type: none"> <li>根據轉換結果，當最上位元為 1 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

### ● FIX

將S所指定的單精確度浮動小數點資料(32位元:IEEE754)的整數部分轉換為帶符號的 BIN (16 位元) 資料，然後再將結果輸出至 D。



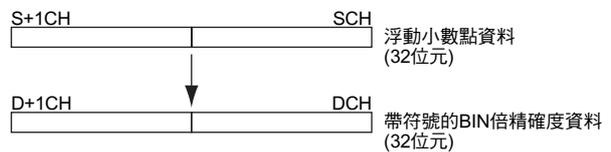
將浮動小數點資料的整數部分轉換為帶符號的 BIN 資料，然後再將結果輸出至所指定的 CH。小數點無條件捨去。

例：

浮動小數點資料		轉換結果 (帶符號的 BIN16 位元)
3.5	→	3
- 3.5	→	- 3

### ● FIXL

將 S 所指定的單精確度浮動小數點資料 (32 位元: IEEE754) 的整數部分轉換為帶符號的 BIN (32 位元) 資料，然後再將結果輸出至 D+1、D。



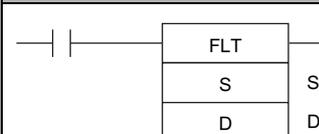
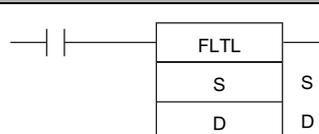
將浮動小數點資料的整數部分轉換為帶符號的 BIN 倍精確度資料，然後再將結果輸出至所指定的 CH。小數點無條件捨去。

例：

浮動小數點資料		轉換結果 (帶符號的 BIN32 位元)
2147483640.5	→	2147483640
- 2147483640.5	→	- 2147483640

# FLT/FLTL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
16 位元 BIN → 轉換為浮動小數點	FLT	@FLT	452	將已指定的帶符號的 BIN16 位元資料轉換為浮動小數點 32 位元資料，然後再將結果輸出至指定的 CH。
32 位元 BIN → 轉換為浮動小數點	FLTL	@FLTL	453	將已指定的帶符號的 BIN32 位元資料轉換為浮動小數點 32 位元資料，然後再將結果輸出至指定的 CH。

符號	FLT	FLTL
	 <p>S : BIN資料 D : 輸出轉換結果的下位CH編號</p>	 <p>S : BIN倍精確度資料的下位CH編號 D : 輸出轉換結果的下位CH編號</p>

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態		容量	
		FLT	FLTL	FLT	FLTL
S	BIN 資料	INT	DINT	1	2
D	輸出轉換結果的下位 CH 編號	REAL	REAL	2	2

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

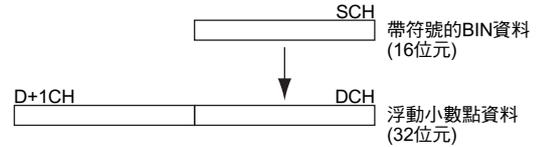
## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當轉換結果(浮動小數點資料)的指數部分和假數部分皆為0(浮動小數點資料中的0)時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
負旗標	P_N	<ul style="list-style-type: none"> <li>當轉換結果為負數時，本旗標 ON</li> <li>否則為 OFF</li> </ul>

## 功能

### ● FLT

將 S 所指定的帶符號 BIN 資料 (16 位元) 轉換為單精確度浮動小數點資料 (32 位元：IEEE754)，然後再將結果輸出至 D+1、D。當浮點資料出現小數點以後的位數時，小數點後第 1 位將變成 0。



S 可指定為 -32768 ~ 32767 範圍內的 BIN 資料。

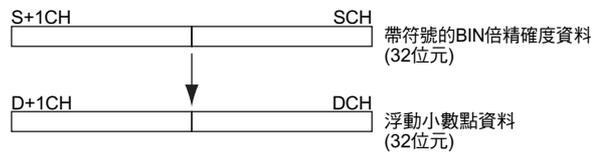
又，如果要轉換為 -32768 ~ 32767 範圍以外的 BIN 資料時，請使用 FLTL 指令。

例：

帶符號的 BIN16 位元		→	浮動小數點資料
3		→	3.0
- 3		→	- 3.0

### ● FLTL

將 S 所指定的帶符號 BIN 資料 (32 位元) 轉換為單精確度浮動小數點資料 (32 位元：IEEE754)，然後再將結果輸出至 D+1、D。當浮動小數點資料出現小數點以後的位數時，小數點後第 1 位將變成 0。



• S 可指定為 -2147483648 ~ 2147483647 範圍內的 BIN 倍精確度資料。

• 浮點的有效位數為 24 位元，因此，一旦 FLTL 指令所轉換的數值超過 16777215 (24 位元 ~ 最大值) 時，轉換結果就會出現誤差。

例：

帶符號的 BIN32 位元		→	浮動小數點資料
16777215		→	16777215.0
- 16777215		→	- 16777215.0

# + F, - F, * F, / F

指令名稱	指令記號	指令的各種組合	Fun No.	功能
浮動小數點加法運算	+ F	@ + F	454	為所指定的浮動小數點資料 (32 位元) 進行加法運算，然後再將結果輸出至指定的 CH。
浮動小數點減法運算	- F	@ - F	455	為所指定的浮動小數點資料 (32 位元) 進行減法運算，然後再將結果輸出至指定的 CH。
浮動小數點乘法運算	* F	@ * F	456	為所指定的浮動小數點資料 (32 位元) 進行乘法運算，然後再將結果輸出至指定的 CH。
浮動小數點除法運算	/ F	@ / F	457	為所指定的浮動小數點資料 (32 位元) 進行除法運算，然後再將結果輸出至指定的 CH。

符號	+ F		- F									
		<table border="1"> <tr><td>+F</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：被加數的浮動小數點資料下位CH編號 S2：加數浮動小數點資料的下位CH編號 D：演算結果的輸出下位CH編號</p>	+F	S1	S2	D		<table border="1"> <tr><td>-F</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：被減數的浮動小數點資料下位CH編號 S2：減數浮動小數點資料下位CH編號 D：演算結果的輸出下位CH編號</p>	-F	S1	S2	D
	+F											
	S1											
S2												
D												
-F												
S1												
S2												
D												
* F		/ F										
	<table border="1"> <tr><td>*F</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：被乘數的浮動小數點資料下位CH編號 S2：乘數浮動小數點資料下位CH編號 D：演算結果的輸出下位CH編號</p>	*F	S1	S2	D		<table border="1"> <tr><td>/F</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> <tr><td>D</td></tr> </table> <p>S1：被除數的浮動小數點資料下位CH編號 S2：除數浮動小數點資料下位CH編號 D：演算結果的輸出下位CH編號</p>	/F	S1	S2	D	
*F												
S1												
S2												
D												
/F												
S1												
S2												
D												

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S1	+ F：被加數的浮動小數點資料下位 CH 編號 - F：被減數的浮動小數點資料下位 CH 編號 * F：被乘數的浮動小數點資料下位 CH 編號 / F：被除數的浮動小數點資料下位 CH 編號	REAL	2
S2	+ F：加數浮動小數點資料的下位 CH 編號 - F：減數浮動小數點資料下位 CH 編號 * F：乘數浮動小數點資料下位 CH 編號 / F：除數浮動小數點資料下位 CH 編號	REAL	2
D	運算結果的下位 CH 編號	REAL	2

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	○	-	-	-

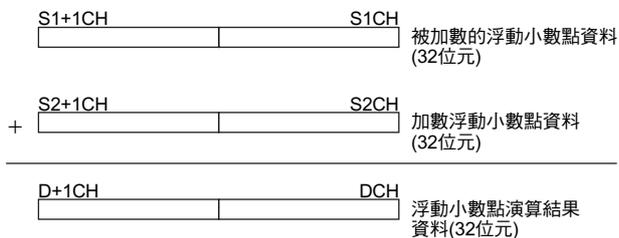
## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>• + F</li> <li>• 當被加數資料或加數資料為非數字 (NaN) 時，本旗標就會 ON</li> <li>• 嘗試將 $+\infty$ 和 $-\infty$ 進行加法運算時，本旗標就會 ON</li> <li>• - F</li> <li>• 當被減數資料或減數資料為非數字 (NaN) 時，本旗標就會 ON</li> <li>• 嘗試將 $+\infty$ 和 $+\infty$ 或者是 $-\infty$ 和 $-\infty$ 進行減法運算時，本旗標就會 ON</li> <li>• * F</li> <li>• 當被乘數資料或乘數資料為非數字 (NaN) 時，本旗標就會 ON</li> <li>• 嘗試將 $+\infty$ 和 $+\infty$ 或者是 $-\infty$ 和 $-\infty$ 進行乘法運算時，本旗標就會 ON</li> <li>• / F</li> <li>• 當被除數資料或除數資料為非數字 (NaN) 時，本旗標就會 ON</li> <li>• 當被除數資料和除數資料皆為 0 或 $\pm\infty$ 時，本旗標就會 ON</li> <li>• 否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>• 當演算結果(浮動小數點資料)的指數部分和假數部分皆為 0 (浮動小數點資料的 0) 時，本旗標就會 ON</li> <li>• 否則為 OFF</li> </ul>
溢位旗標 (Overflow flag)	P_OF	<ul style="list-style-type: none"> <li>• 當演算結果的絕對值大於浮動小數點資料所能表現的最大值時，本旗標就會 ON</li> <li>• 否則為 OFF</li> </ul>
欠位旗標 (Underflow flag)	P_UF	<ul style="list-style-type: none"> <li>• 當演算結果的絕對值小於浮動小數點資料所能表現的最小值時，本旗標就會 ON</li> <li>• 否則為 OFF</li> </ul>
負旗標	P_N	<ul style="list-style-type: none"> <li>• 當演算結果為負數時，本旗標就會 ON</li> <li>• 否則為 OFF</li> </ul>

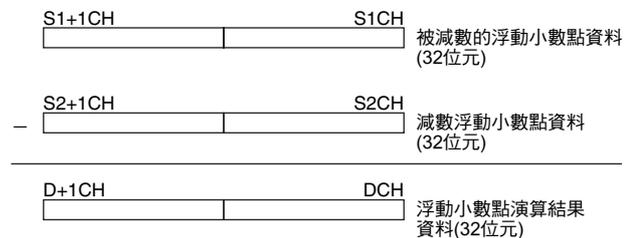
## 功能

將 S1 所指定的資料及 S2 所指定的資料當作單精確度浮動小數點資料 (32 位元：IEEE754)，進行加法運算 (+F)、減法運算 (-F)、乘法運算 (*F)、除法運算 (/F)，然後再將結果輸出至 D+1、D。

## • + F



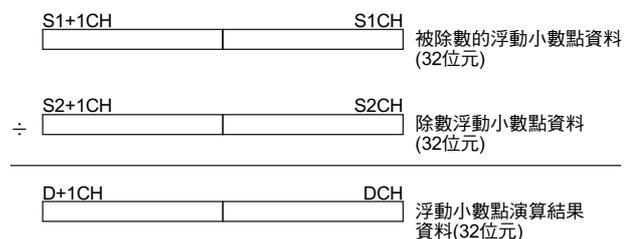
## • - F



## • * F



## • / F



- 當演算結果的絕對值大於浮動小數點資料所能表現的最大值時，溢位旗標就會 ON，此時，將輸出  $\pm\infty$  作為演算結果。
- 當演算結果的絕對值小於浮動小數點資料所能表現的最小值時，欠位旗標就會轉為 ON。此時，將輸出浮動小數點資料的 0 作為演算結果。

## ● 演算規則

當所要演算的浮動小數點資料組合不同時，將輸出下表所示的各種結果。

### ● 浮動小數點加法運算 (+F)

		被加資料				
		0	數值	+∞	-∞	非數
加算資料	0	0	數值	+∞	-∞	ER
	數值	數值	* 1	+∞	-∞	
	+∞	+∞	+∞	+∞	ER	
	-∞	-∞	-∞	ER	-∞	
	非數					
						ER

* 1 此時將輸出 0(包含欠位的結果 UF 為 0 的情況)、數值、+∞或是 -∞等任一種結果。

ER：當異常旗標 ON 時，將不會執行任何指令。

### ● 浮動小數點加法運算 (-F)

		被減資料				
		0	數值	+∞	-∞	非數
減算資料	0	0	數值	+∞	-∞	ER
	數值	數值	* 1	+∞	-∞	
	+∞	-∞	-∞	ER	-∞	
	-∞	+∞	+∞	+∞	ER	
	非數					
						ER

* 1 此時將輸出 0(包含欠位的結果 UF 為 0 的情況)、數值、+∞或是 -∞等任一種結果。

ER：當異常旗標 ON 時，將不會執行任何指令。

### ● 浮動小數點乘法運算 (*F)

		被乘資料				
		0	數值	+∞	-∞	非數
乘算資料	0	0	0	ER	ER	ER
	數值	0	* 1	+/-∞	+/-∞	
	+∞	ER	+/-∞	+∞	-∞	
	-∞	ER	+/-∞	-∞	+∞	
	非數					
						ER

* 1 此時將輸出 0(包含欠位的結果 UF 為 0 的情況)、數值、+∞或是 -∞等任一種結果。

ER：當異常旗標 ON 時，將不會執行任何指令。

### ● 浮動小數點除法運算 (/F)

		被除資料				
		0	數值	+∞	-∞	非數
除算資料	0	ER	+/-∞	+∞	-∞	ER
	數值	0	* 2	+/-∞	+/-∞	
	+∞	0	0* 1	ER	ER	
	-∞	0	0* 1	ER	ER	
	非數					
						ER

? 1 將出現欠位，數值為 0 的結果。

* 2 此時將輸出 0(包含欠位的結果 UF 為 0 的情況)、數值、+∞或是 -∞等任一種結果。

ER：當異常旗標 ON 時，將不會執行任何指令。

# =F , <>F , <F , <=F , >F , >=F

指令名稱	指令記號	指令的各種組合	Fun No.	功能
單精確度的浮動小數點資料比較	=F	—	329	比較所指定的單精確度浮動小數點資料 (32 位元) 或常數後，如果比較結果為真，就會連接至下一段以後的回路區。
	<>F		330	
	<F		331	
	<=F		332	
	>F		333	
	>=F		334	



## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S1	比較資料 1	REAL	2
S2	比較資料 2	REAL	2

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S1+1、S1 或是 S2+1、S2 為非數字 (NaN) 時，本旗標就會 ON</li> <li>當 S1+1、S1 或是 S2+1、S2 為 +∞時，本旗標就會 ON</li> <li>當 S1+1、S1 或是 S2+1、S2 為 -∞時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
> 旗標	P_GT	<ul style="list-style-type: none"> <li>當比較結果為 S1+1、S1 &gt; S2+1、S2 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
≥ 旗標	P_GE	<ul style="list-style-type: none"> <li>當比較結果為 S1+1、S1 ≥ S2+1、S2 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當比較結果為 S1+1、S1 = S2+1、S2 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
≠ 旗標	P_NE	<ul style="list-style-type: none"> <li>當比較結果為 S1+1、S1 ≠ S2+1、S2 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
< 旗標	P_LT	<ul style="list-style-type: none"> <li>當比較結果為 S1+1、S1 &lt; S2+1、S2 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
≤ 旗標	P_LE	<ul style="list-style-type: none"> <li>當比較結果為 S1+1、S1 ≤ S2+1、S2 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
負旗標	P_N	無任何變化

## 功能

當 S1 所指定的資料和 S2 所指定的資料被當作單精確度浮動小數點資料 (32 位元：IEEE754) 比較，且比較結果為真時，就會被連接至下一段以後的回路。

指定比較資料 S2、S2 前，必須先將儲存資料用的 CH 起始位址輸入至運算元種類 S1、S2。

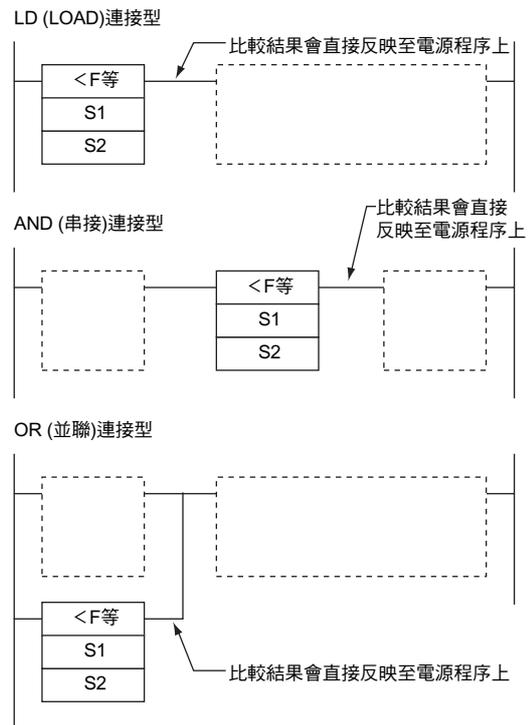
又，如果要將常數用於比較資料時，必須將 16 進位的 8 位數數值直接輸入 S1、S2。

和 LD、AND、OR 指令的處理方法一樣，在各指令之後接著編寫其他指令。

LCD 型時：可直接連接母線。

AND 型時：無法直接與母線連接。

OR 型時：可直接連接母線。



## ● 符號及選項

將符號及選項互相組合後，即可利用 18 種指令欄來表示比較指令。

符號 (階梯程式中無 LD、AND、OR)	選項 (資料類型)
LD=、AND=、OR=、LD<>、AND<>、OR<>、LD<、AND<、OR<、LD<=、AND<=、OR<=、LD>、AND>、OR>、LD>=、AND>=、OR>=	F (單精確度浮動小數點)

功能	指令記號	名稱	Fun No.
當 S1 = S2 時為 True (真) (ON)	LD = F	LD 型・浮動小數點資料一致	329
	AND = F	AND 型・浮動小數點資料一致	329
	OR = F	OR 型・浮動小數點資料一致	329
當 S1 ≠ S2 時為 True (真) (ON)	LD <> F	LD 型・浮動小數點資料不一致	330
	AND <> F	AND 型・浮動小數點資料不一致	330
	OR <> F	OR 型・浮動小數點資料不一致	330
當 S1 < S2 時為 True (真) (ON)	LD < F	LD 型・浮動小數點資料小於	331
	AND < F	AND 型・浮動小數點資料小於	331
	OR < F	OR 型・浮動小數點資料小於	331
當 S1 ≤ S2 時為 True (真) (ON)	LD ≤ F	LD 型・浮動小數點資料以下	332
	AND ≤ F	AND 型・浮動小數點資料以下	332
	OR ≤ F	OR 型・浮動小數點資料以下	332
當 S1 > S2 時為 True (真) (ON)	LD > F	LD 型・浮動小數點資料超過	333
	AND > F	AND 型・浮動小數點資料超過	333
	OR > F	OR 型・浮動小數點資料超過	333
當 S1 ≥ S2 時為 True (真) (ON)	LD ≥ F	LD 型・浮動小數點資料以上	334
	AND ≥ F	AND 型・浮動小數點資料以上	334
	OR ≥ F	OR 型・浮動小數點資料以上	334

### 使用上的注意事項

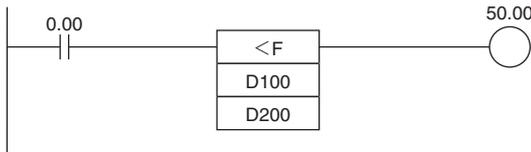
- 請在本指令的最後一段附加輸出類指令 (OUT 類指令及下一段連接型指令以外的應用指令)。
- 本指令不適用於回路的最末段。

### 程式例

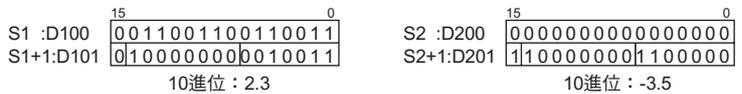
當 0.00 變成 ON 時，就會開始比較儲存在資料記憶體 D100、101 中的浮動小數點資料以及儲存在 D200、201 中的浮動小數點資料。

當比較結果為 (D100、101 的資料) < (D200、201 的資料) 時，就會連接至下一段以後的回路，此時輸出繼電器 50.00 將變成 ON。

當比較結果並非 (D100、101 的資料) < (D200、201 的資料) 時，則不會連接至下一段的回路。



單精確度浮動小數點資料之比較(<F)



↓ 2.3 > -3.5

無法連接下一段以後的電路



↓ 4294967296 < 5566555656

連接於下一段以後的電路

# FSTR

指令名稱	指令記號	指令的各種組合	Fun No.	功能
浮動小數點 < 單精確度 > → 文字列轉換	FSTR	@FSTR	448	利用小數點或是指數格式來表示所指定的單精確度浮動小數點資料 (32 位元)，然後再將該資料轉換為文字列資料 (ASCII 碼)

符號	FSTR	
		S: 浮動小數點資料的下位CH編號 C: 控制資料 D: 轉換結果的輸出目的地CH編號

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	浮動小數點資料的下位 CH 編號	REAL	2
C	控制資料	UINT	3
D	轉換結果的輸出目的地 CH 編號	UINT	可調整

### C: 控制資料

C	文字列的表示格式	&0: 小數點格式 &1: 指數格式
C+1	所有位數	10進位&2 ~ &24 (16進位#0002 ~ 0018) (註)
C+2	小數部分位數	10進位&0 ~ &7 (16進位#0000 ~ 0007) (註)

(註) 所有位數、小數部分的位數各有其限制條件。

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	—	—	—
C,D	○	○	○	○	○	○	○	○	○	○	—	—	—

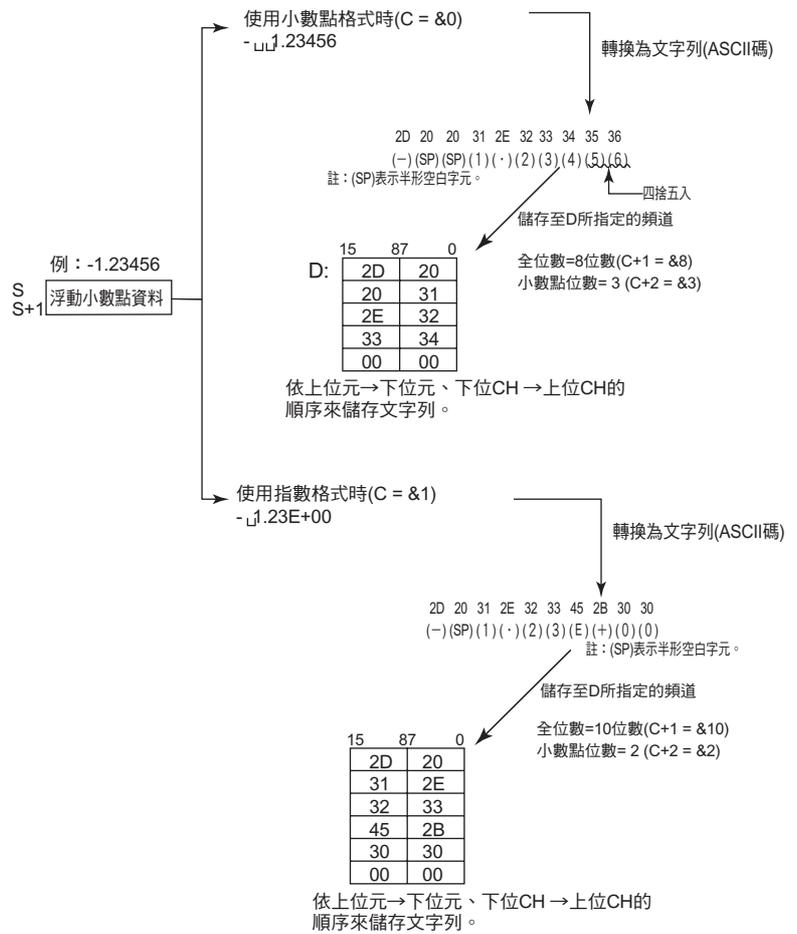
## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S+1、S 的資料為非數字 (NaN) 時，本旗標 ON</li> <li>當 S+1、S 的資料為 +/- ∞ 時，本旗標 ON</li> <li>當 C 所指定的文字列被指定為 0000、0001 以外的表示格式時，本旗標就會 ON</li> <li>當 C+1 的所有位數超出範圍 (請參閱「轉換後的文字列位數限制」) 時，本旗標就會 ON</li> <li>當 C+2 所指定的小數部分位數超出範圍 (請參閱「轉換後的文字列位數限制」) 時，本旗標就會 ON</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當轉換結果為「0」時，本旗標 ON</li> <li>否則為 OFF</li> </ul>

### 功能

根據 C、C+2 的內容，以小數點或指數格式來表示 S 所指定的單精確度浮動小數點資料 (32 位元 :IEEE754)，接著將該資料轉換為文字列 (ASCII 碼)，然後再將結果輸出至 D 所指定的 CH。

- 利用 C (轉換格式) 來指定要以小數點格式 / 指數格式的其中任一種來表示 S+1、S 的浮動小數點資料。
    - 小數點格式
      - 利用整數部分及小數部分來表現實數的一種格式。
      - 例) 124.56
    - 指數格式
      - 利用整數部分、小數部分及指數部分來表現實數的一種格式。
      - 例) 1.2456E-2 (1.2456 × 10⁻²)
  - 利用 C+1 (所有位數) 來指定轉換後的文字列數 (包含符號、數值、小數點、半形空白)。
  - 利用 C+2 (小數部分位數) 來指定轉換後的文字列的小數部分位數 (字元數)。
- D 以後的儲存順序 → D 的下位元 → D+1 的上位元 → D+1 的下位元 → ... (以下相同)。

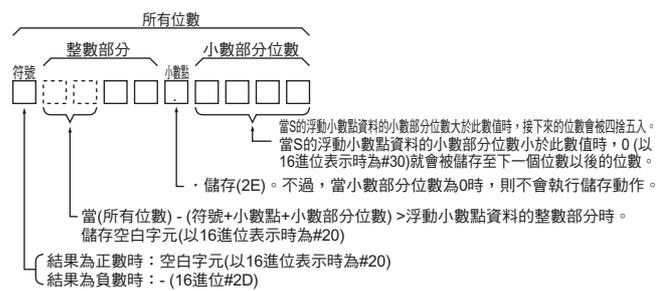


### ● 如何儲存轉換後的文字列

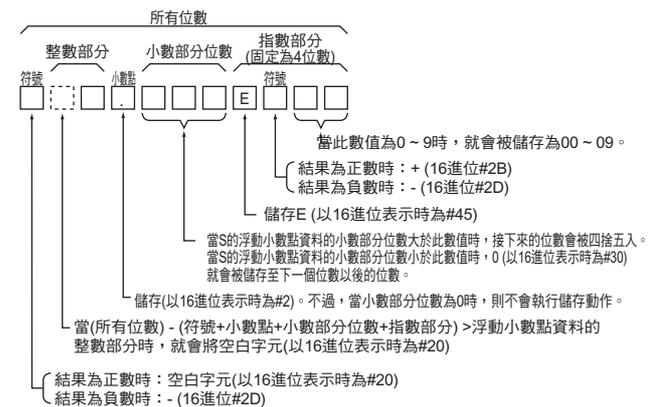
依右圖所示，轉換後的文字列會被儲存至 D。

又，使用小數點和指數格式時，其文字列的儲存方法將各有不同。

使用小數點格式時(C = 0)



使用指數格式時(C = 16進位#1)



(註) 轉換後的文字列在經過最後處理後(結束碼)將儲存為下列格式。

- 當所有位數為奇數時：「00Hex」將會被儲存在最後一個文字列後方。
- 當所有位數為偶數時：「0000Hex」將會被儲存在最後一個文字列後方。

### ● 轉換後的文字列位數限制

轉換後的文字列位數(字元數)有其限制條件。又，一旦不符合限制條件時，異常旗標就會 ON。

• 對於所有位數(所有字元數)的限制

- 1) 當轉換後的文字列為小數點格式(C = 0)時
  - 當小數部分的位數為 0 時： $2 \leq \text{所有位數} \leq 24$
  - 當小數部分的位數不為 0 時： $(\text{小數部分位數} + 3) \leq \text{所有位數} \leq 24$
- 2) 當轉換後的文字列為指數格式(C = 1)時
  - 當小數部分的位數為 0 時： $6 \leq \text{所有位數} \leq 24$
  - 當小數部分的位數不為 0 時： $(\text{小數部分位數} + 7) \leq \text{所有位數} \leq 24$

• 對於整數部分位數(整數部分的字元數)的限制

- 1) 當轉換後的文字列為小數點格式(C = 0)時
  - 當小數部分的位數為 0 時： $1 \leq \text{整數部分位數} \leq 24 - 1$
  - 當小數部分的位數不為 0 時： $1 \leq \text{整數部分位數} \leq 24 - \text{小數部分位數} - 2$
- 2) 當轉換後的文字列為指數格式(C = 1)時
  - 1 (固定)

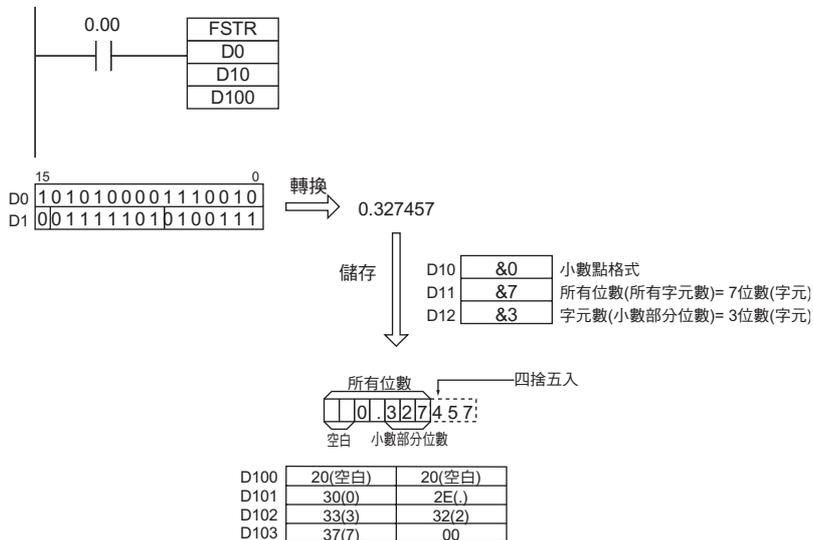
• 對於小數部分位數的限制

- 1) 當轉換後的文字列為小數點格式(C = 0)時
  - 小數部分位數  $\leq 7$  且  $(\text{小數部分的位數}) \leq (\text{所有位數} - 3)$
- 2) 當轉換後的文字列為指數格式(C = 1)時
  - 小數部分位數  $\leq 7$  且  $(\text{小數部分的位數}) \leq (\text{所有位數} - 7)$

### 程式例

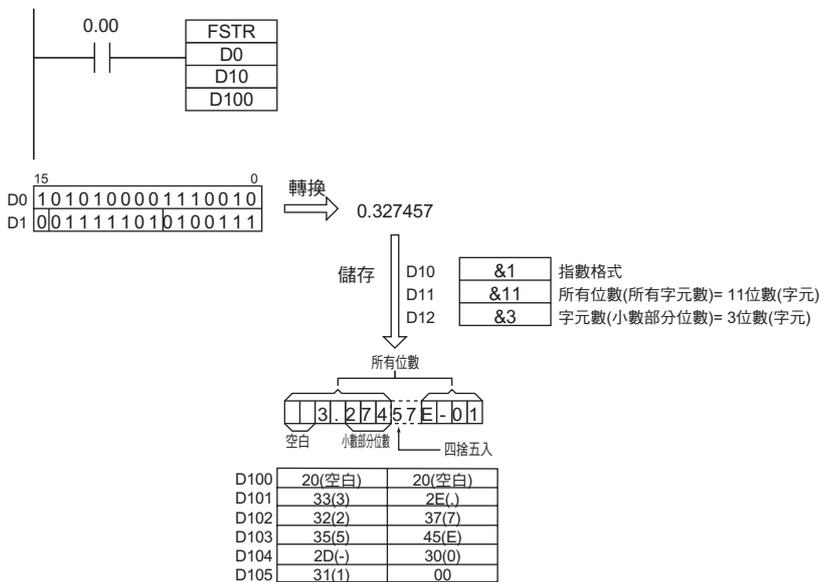
#### ● 將浮動小數點資料轉換為小數點格式的文字列

當 0.00 變成 ON 時，就會將儲存在 D1、D0 的浮動小數點資料轉換為小數點格式的文字列，然後再根據儲存在 D10 以後的控制資料內容 ( 小數點格式、所有位數 = 7 位數、小數部分位數 = 3 位數 )，將轉換後的文字列儲存在 D100 以後的位置。



#### ● 將浮動小數點資料轉換為指數格式的文字列

當 0.00 變成 ON 時，就會將儲存在 D0 的浮動小數點資料轉換為指數格式的文字列，然後再根據儲存在 D10 以後的控制資料內容 ( 指數格式、所有位數 = 11 位數、小數部分位數 = 3 位數 )，將轉換後的文字列儲存在 D100 以後的位置。



# FVAL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
文字列→浮動小數點 <單精確度>轉換	FVAL	@FVAL	449	將表示為指定的小數點格式或指數格式的文字列 (ASCII 碼) 轉換為單精確度浮動小數點資料 (32 位元)，然後再將結果輸出至所指定的 CH。

符號	FVAL	
		<p>S: 文字列資料的儲存頻道編號</p> <p>D: 輸出轉換結果的頻道編號</p>

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	文字列資料的儲存 CH 編號	UINT	可調整
D	輸出轉換結果的 CH 編號	REAL	2

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	-	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S 所指定的文字列資料的整數部分、小數部分出現 #30 ~ 39 (字元 0 ~ 9) 以外的字元時，本旗標就會 ON</li> <li>當 S 所指定的文字列資料的指數部分出現「#45 (字元 E)、#2B (字元 +)」、「#45 (字元 E)、#2D (字元 -)」以外的字元時，本旗標就會 ON</li> <li>當 S 所指定的文字列出現多個指數部分時，本旗標就會 ON</li> <li>當轉換後的資料變成 +/- ∞ 時，本旗標就會 ON</li> <li>當 S 所指定的文字列資料的字元數為 0 時，本旗標就會 ON</li> <li>在 S 所指定的文字列資料中，只要有 25 個以下的字元未出現 #00，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當轉換結果為「0」時，本旗標 ON</li> <li>否則為 OFF</li> </ul>

## 功能

將儲存在 S 所指定的 CH 的文字列資料 (ASCII 碼) 轉換為單精確度浮動小數點資料 (32 位元：IEEE754)，然後再將結果輸出至 D 所指定的 CH。

需要轉換的文字列資料就是以下列任一種格式來表現的 ASCII 碼。

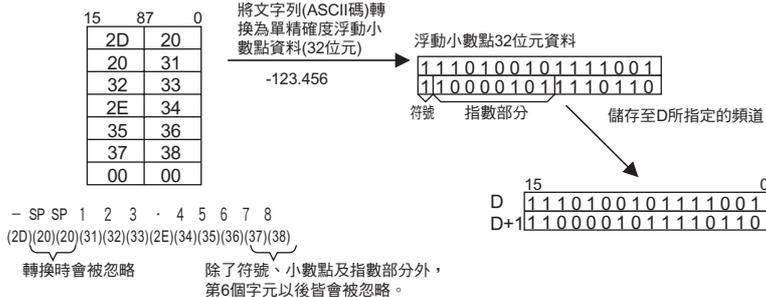
除了符號、小數點、指數部分外，本功能的有效範圍為 6 個字元以下。第 6 個字元以後的字元皆被視為無效。

- 小數點格式  
利用整數部分及小數部分來表現實數的一種格式。  
例) 124.56
- 指數格式  
利用整數部分、小數部分及指數部分來表現實數的一種格式。  
例) 1.2456E-2 ( $1.2456 \times 10^{-2}$ )

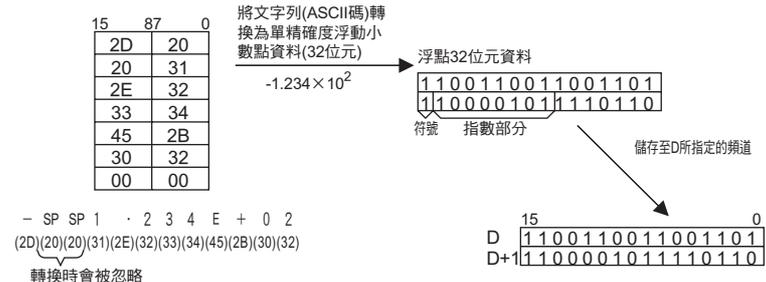
又，系統會自動判別所儲存的文字列資料格式。

S 以後的文字列儲存順序為 S 的上位元 → S 的下位元 → S+1 的下位元 → S+1 的下位元 → ... (以下相同)。

使用小數點格式時



使用指數格式時



● 如何儲存文字列

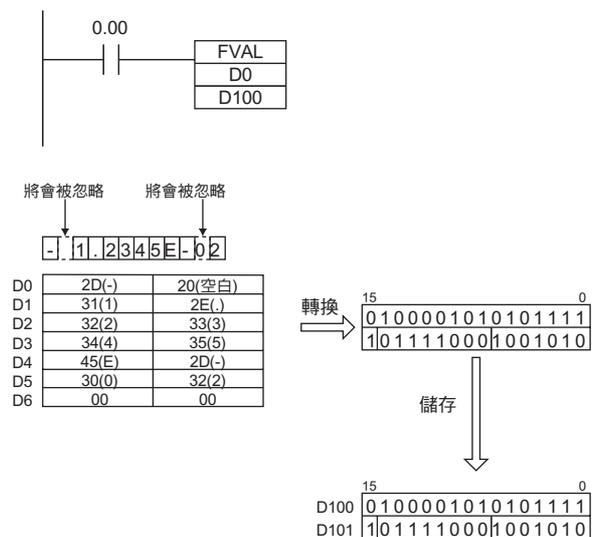
文字列的轉換方式如下表所示。

又，小數點格式和指數格式的轉換條件不同。



● 將指數格式的文字列轉換為浮動小數點資料時

當 0.00 變成 ON 時，就會將儲存在 D0 的指數格式的文字列轉換為浮點小數資料，然後再將轉換後的浮動小數點資料儲存在 D100 以後的位置。



# 資料表執行資料處理時的指令

## SWAP

指令名稱	指令記號	指令的各種組合	Fun No.	功能
位元置換 (Byte swap)	SWAP	@SWAP	637	置換指定資料表其資料所指定的 CH 數的上位元及下位元。

符號	SWAP	
		<p>S : 指定資料表長度的資料</p> <p>D : 資料表的下位CH編號</p>

### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 運算元種類的說明

運算元種類	內容	資料型態	容量
S	指定資料表長度的資料	UINT	1
D	資料表的下位 CH 編號	UINT	可調整

S : 指定資料表長度的資料

10 進位 & 1 ~ 65535 或 16 進位 #0001 ~ FFFF

D : 資料表的下位 CH 編號



(註) D ~ D+(S - 1) 的運算元種類必須相同。

### ● 運算元種類

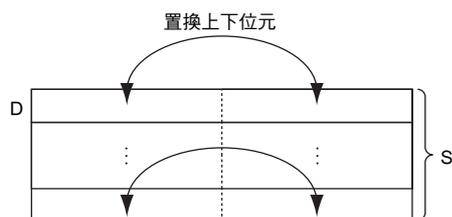
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	○	-	-	-
D	○	○	○	○	○	○	○	○	○	-	-	-	-

### 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S 的資料為 0 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

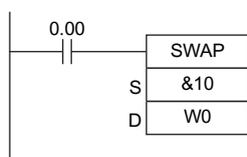
將 D 所指定的資料表其資料起始 CH~ 指定資料表長度 (S) 的資料表資料分別置換為資料 (16 位元) 的上位 8 位元及下位 8 位元，然後再將結果輸出至 D。



## 提示

- 當文字列 (ASCII 碼) 的儲存順序相反時，即可使用本功能。

## 程式例



當0.00變成ON時，針對W0CH ~ W9CH的10CH分別置換每個資料的上位元及下位元。

	15	8	7	0		15	8	7	0
W0	4	1	4	2	⇒	4	2	4	1
W1	4	3	4	4		4	4	4	3
W2	4	5	4	6		4	6	4	5
⋮						⋮			
W9	3	0	3	1		3	1	3	0

# FCS

指令名稱	指令記號	指令的各種組合	Fun No.	功能
計算 FCS 值	FCS	@FCS	180	計算指定資料表中的 FCS 值，接著將該值轉換為 ASCII 碼資料後再輸出。

符號	FCS					
		<table border="1"> <tr><td>FCS</td></tr> <tr><td>C</td></tr> <tr><td>S</td></tr> <tr><td>D</td></tr> </table>	FCS	C	S	D
FCS						
C						
S						
D						

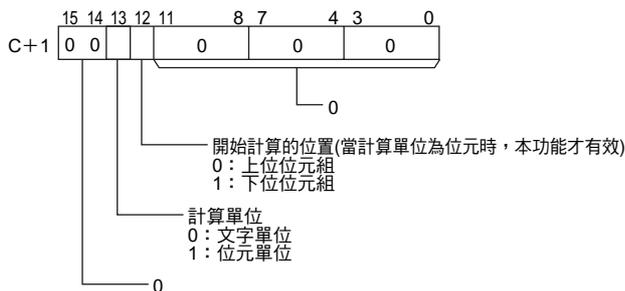
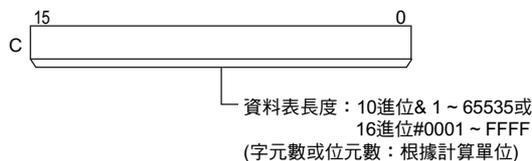
## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

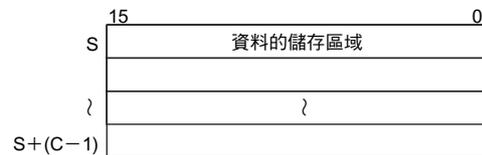
## 運算元種類的說明

運算元種類	內容	資料型態	容量
C	控制資料	UDINT	2
S	資料表的下位 CH 編號	UINT	可調整
D	儲存 FCS 值的起始 CH 編號	UINT	可調整

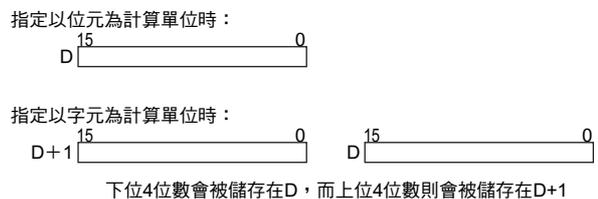
### C：控制資料



### S：資料表的下位 CH 編號



### D：儲存 FCS 值的起始 CH 編號



(註) C ~ C+1 及 S ~ S+(C 的資料 - 1) 的運算元種類皆必須相同。

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C	○	○	○	○	○	○	○	○	○	○	—	—	—
S,D	○	○	○	○	○	○	○	○	○	—	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 C 的資料表長度資料 (位元 00~15) 超出 10 進位的 &amp;1 ~ 65535 或是 16 進位的 #0001 ~ FFFF 的範圍時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

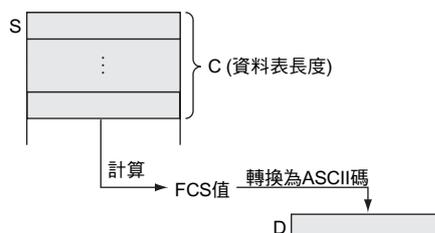
## 功能

將 S 所指定的資料表下位 CH 編號 ~C 所指定的資料表長度視為資料表資料，利用 C+1 所指定的計算單位 (以字元或位元為單位) 來演算 FCS 值，並且轉換為 ASCII 碼資料，

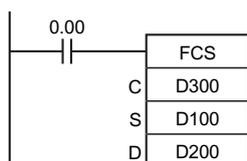
- 指定以位元為計算單位時：將輸出 D。
- 指定以字元為計算單位時：將輸出 D+1、D。

指定以位元作為計算單位後，必須利用 C+1 來指定開始計算位置 (上位元或下位元)。

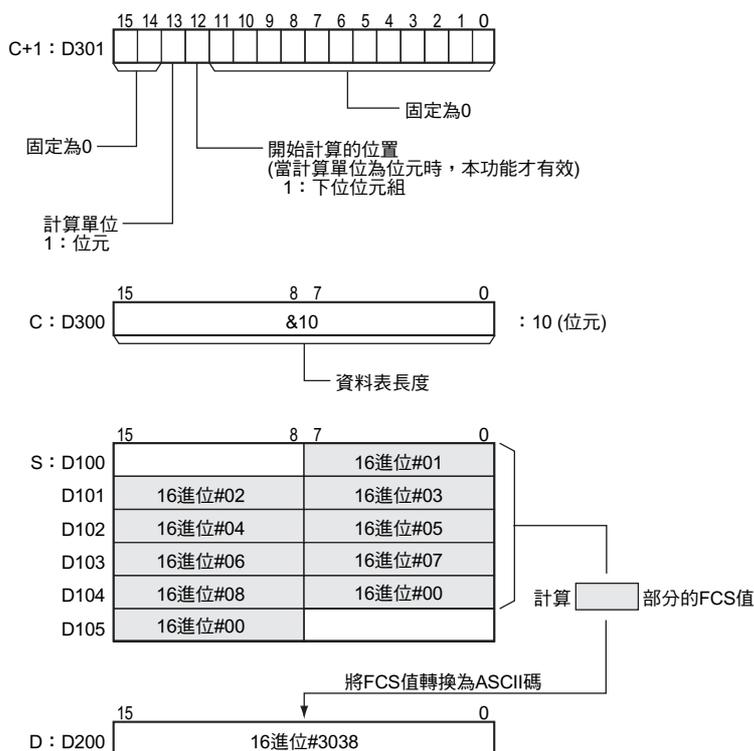
資料表長度 (C) 的單位 (字元或位元) 係根據計算單位 (C+1) 而定。



## 程式例



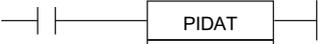
當0.00變成ON時，D100的下位元~ 300所指定的位元數資料將會被當作BIN資料，計算其FCS值，然後再將結果儲存至D200。



# 資料控制指令

## PIDAT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
附自動微調功能的 PID 演算	PIDAT	—	191	根據您所指定的參數，進行 PID 演算。

符號	附自動微調功能的 PID 演算						
		<table border="1"> <tr> <td>S</td> <td>S：輸入測定值的CH編號</td> </tr> <tr> <td>C</td> <td>C：儲存PID參數的下位CH編號</td> </tr> <tr> <td>D</td> <td>D：輸出操作量的CH編號</td> </tr> </table>	S	S：輸入測定值的CH編號	C	C：儲存PID參數的下位CH編號	D
S	S：輸入測定值的CH編號						
C	C：儲存PID參數的下位CH編號						
D	D：輸出操作量的CH編號						

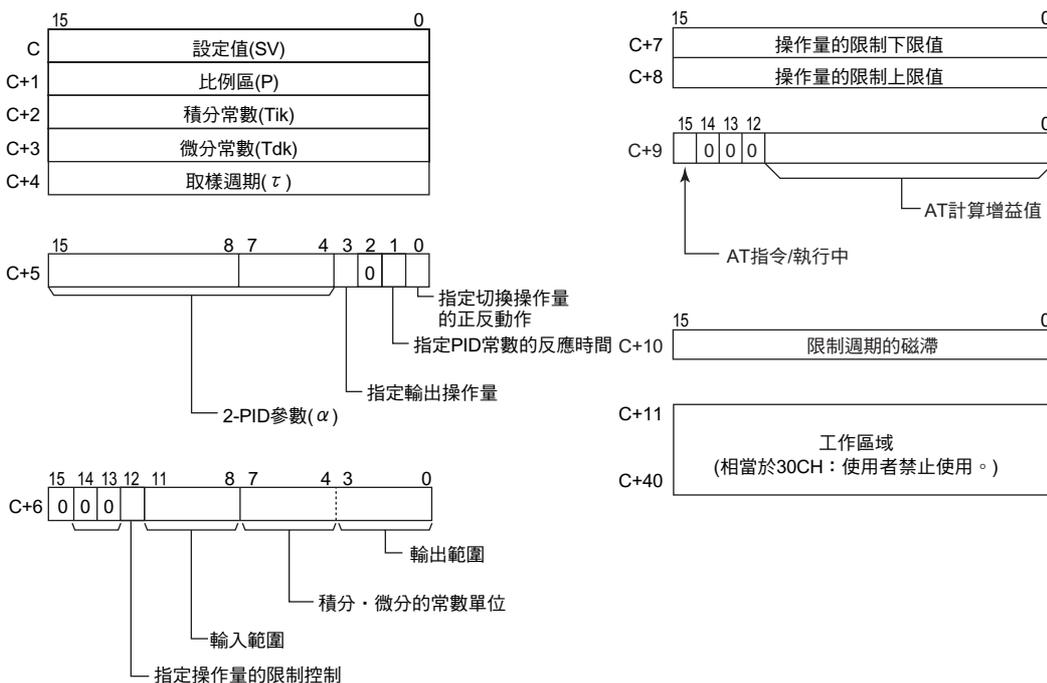
### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

### 運算元種類的說明

運算元種類	內容	資料型態	容量
S	輸入測定值的 CH 編號	UINT	1
C	儲存 PID 參數的下位 CH 編號	WORD	41
D	輸出操作量的 CH 編號	UINT	1

#### C：儲存 PID 參數的下位 CH 編號



## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S,C,D	○	○	○	○	○	○	○	○	○	—	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 C 的資料超過範圍時變為 ON</li> <li>當實際的取樣週期超過您所設定的取樣週期的 2 倍時，本旗標就會 ON</li> <li>如果再執行自動微調時發生異常，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
> 旗標	P_GT	<ul style="list-style-type: none"> <li>當 PID 所演算的操作量超過操作輸出的限制上限值時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
< 旗標	P_LT	<ul style="list-style-type: none"> <li>當 PID 所演算的操作量超過操作輸出的限制下限值時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> <li>執行 PID 演算時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

根據 C 所指定的參數 ( 設定值、PID 常數等 )，以 S 作為測定值輸入資料執行 PID 演算 ( 目標值濾波器型 2 靈活度 PID 演算 )，然後再將操作量輸出至 D。

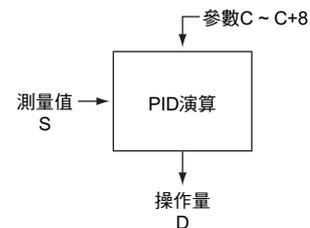
當輸入條件上升 (OFF → ON) 時，就開始讀取參數，一旦超出正常範圍，異常旗標就會變成 ON。

只要在正常範圍內，操作量就會被當作初始值並且執行 PID 處理。

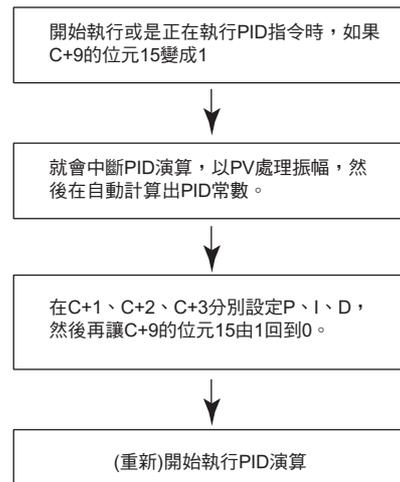
當輸入條件 ON 時，就會將每個指定的取樣週期當作輸入資料，並進行演算。

當輸入條件 OFF 時，就會停止 PID 演算，D 的操作量將保持當時的數值，如需變更時，請利用階梯圖程式或是以手動方式來變更數值。

- 當某個週期的 AT 指令 / 執行過程中 (C+9 的位元 15) 變成 1 (ON) 時，( 檢查每個週期的位元 )，就會開始執行 PID 常數的自動微調 (AT) ( 又，AT 執行時，並不會造成 SV 變更 ) AT 需符合限制週期法之規定。強制改變控制量 ( 最大操作量 ← → 最小操作量 )，並根據觀察結果，自動演算 P、I、d 常數，然後再分別儲存至 C+1、C+2、C+3。同時，將 AT 指令 / 執行中 (C+9 的位元 15) 當作 1 (ON) ~ 0 (OFF)，接著再根據自動演算後的 PID 常數 (C+1、C+2、C+3) 數值，開始進行 PID 演算。
- 開始執行 PIDAT 指令時，如果 C+9 的位元 15 變成 1：先執行 AT，然後再利用計算出來的 PID 常數，執行 PID 演算。
- 開始執行 PIDAT 指令時，如果 C+9 的位元 15 被當作 0~1：就會中斷以使用者預先儲存好的 PID 常數所執行的 PID 演算，開始執行 AT 後，則會利用計算出來的 PID 常數，( 重新 ) 開始 PID 演算。



## 執行 AT 的步驟



(註)1 執行 AT 時，如果因為 C+9 的位元 15 由 1 → 0 而中斷 AT 時，就會利用 AT 開始執行前的 PID 參數，開始 PID 演算。  
2 發生 AT 執行錯誤時，同樣會利用 AT 開始執行前的 PID 參數，開始 PID 演算。1)、2) 皆會以中斷 AT 時的內容做為 P、I、D 的有效演算參數。

- 指定輸入範圍 (C+6 的位元 8 ~ 11) 後，即可針對 S 的測量值輸入的 16 位元中指定有效的位元數。例如，將輸入範圍指定為 12 位元 (#4) 時，16 進位的 #0000 ~ 0FFF 範圍內的數值皆會被當作有效的測量值。當輸入的數值大於輸入範圍所指定的位元時，就會利用輸入範圍的最大值來修正，此時，異常旗標將不會變成 ON。
- 設定值的範圍同樣必須符合輸入範圍。
- 測量值及設定值就是 16 進位表示時的 #0000 ~ 輸入範圍最大值的不帶符號 BIN 值。
- 指定輸入範圍 (C+6 的位元 0 ~ 3) 後，即可針對操作量的 16 個位元，指定有效的位元數。例如，將輸出範圍指定為 12 位元 (#4) 後，16 進位的 #0000 ~ 0FFF 範圍內的數值即可被當作操作量輸出。
- 只有在執行比例動作時，才能將測量值 = 設定值時的操作量指定為 0% 或是 50% 的其中任一種。
- 可指定比例動作的正、反兩種動作。
- 可指定操作量的上下限。
- 取樣週期可指定為以 10 ms 為單位 (0.01 ~ 99.99s)。不過，實際在進行 PID 演算時，則是根據取樣週期和 PIDAT 指令執行時 (每個週期) 的組合方式，來決定是否執行動作。
- 如果在執行 PID 演算時，變更比例區 (P)、積分長數 (Tik)、微分常數 (Tdk)，即可指定為是 / 否依取樣週期別，將變更內容反映在 PID 演算中。細部設定取決於 PID 常數反映時間的指定內容 (C+5 的位元 1)。

## 提示

- PIDAT 指令會將輸入條件上升當作 STOP → RUN 來執行。當輸入條件上升時，就會將 C+11 ~ C+40 初始化 (清除)，只要下一個週期以後的輸入條件保持 ON 的狀態，即開始執行 PIDAT 指令。因此，當常 ON 旗標 (ON) 被當作 PIDAT 指令的輸入條件時，請另行設定開始運轉時 C+11 ~ C+40 的初始化 (清除) 處理動作。

## 使用上的注意事項

- 多個 PIDAT 指令無法共用同一個 PID 參數的儲存 CH。即使多個 PIDAT 要使用相同的參數時，也必須指定其他的 CH。
- 以手動方式變更 PID 常數時，PID 常數反映時間的指定內容 (C+5 的位元 1) 就會被當作 1 (依週期別將 C+1、C+2、C+3 的數值反映至 PID 演算中)，又，此種指定方式只有在執行 AT 後，希望以手動方式調整 PID 常數時有效。
- 在 PID 參數 (C ~ C+40) 中，可以讓輸入條件保持 ON 狀態，而且能夠變更的數值為下列參數。如果要變更其他的數值時，必須讓輸入條件上升，也就是 OFF → ON。
  - C 的設定值 (SV) (僅限於執行 PID 演算時，執行 AT 時將不會反映變更後的數值)
  - 指定 PID 常數的反映時間
  - 當 PID 常數的反映時間 (C+5 的位元 1) 為 1 (依週期別反映 C+1、C+2、C+3 的數值) 時 C+1、C+2、C+3 的 P、I、D 常數
  - C+9 位元 15 的 AT 指令 / 執行中的位元
  - C+9 位元 0 ~ 14 的 AT 計算增益值及 C+10 的限制週期磁滯 (不過，無論前述哪一項，皆會在開始執行 AT 且反映時間為 1 時開始讀取)

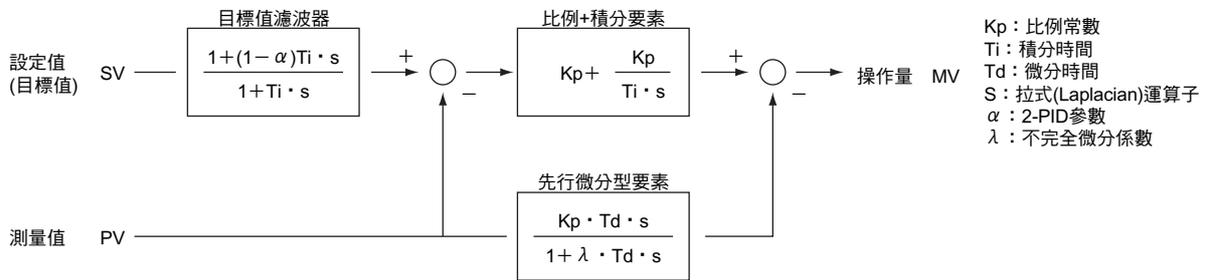
## 功能規格

項目		規格	
PID 演算方式		—	目標值濾波器型 2 自由型 PID 方式 (正向動作 / 反相動作)
PID 控制的群組數		—	無限制 (1 個指令控制 1 個群組)
取樣控制		$\tau$	0.01 ~ 99.99s
PID 常數	比例區	P	0.1 ~ 999.9%
	積分常數	Tik	1 ~ 8191、9999 (如果是取樣週期的倍數或 9999 時，則不會執行積分動作)
	微分常數	Tdk	0 ~ 8191 (如果是取樣週期的倍數或 0 時，則不會執行微分動作)
設定數		SV	0 ~ 65535 (不過，必須小於輸入範圍的最大值，設定才有效)
設定值		PV	0 ~ 65535 (不過，必須小於輸入範圍的最大值，設定才有效)
操作值		MV	0 ~ 65535 (不過，必須小於輸出範圍的最大值，設定才有效)

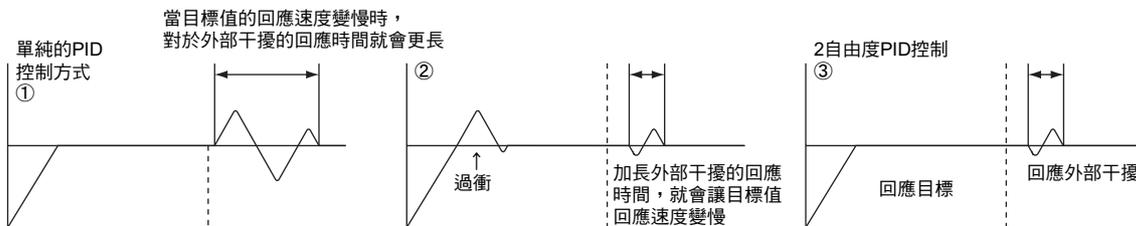
## 演算方式

PID 控制在執行演算時，係採用目標值濾波器型 2 自由型 PID 方式。

### 目標值濾波器型 2 自由型 PID 方式的方塊圖



- 執行單純的 PID 控制時，只要避免過衝 (Overshoot) 發生，就會造成對於外部干擾的穩定時間變慢 (①)，反之，如果提早對於外部干擾的穩定時間，就會出現過衝，此時對於目標值的回應速度就會變慢 (②)。相對地，執行 2 自由度 PID 控制時，不會發生過衝，而且不但對於目標值的回應速度快，還能夠更快達成對於外部干擾的穩定性 (③)。



## PID 參數的設定內容

控制資料	項目	內容	設定範圍	是否可在保持輸入條件 ON 的情況下變更
C	設定值 (SV)	控制對象的目標值。	輸入範圍的位元數所對應的 BIN 資料 (0 ~ 指定的輸入範圍最大值)	可 (不過, 執行 AT 時無法變更)
C+1	比例區 (P)	用來表示比例控制範圍 / 所有控制範圍的 P 控制用參數。	10 進位 &1 ~ 9999 或 16 進位 #0001 ~ 270F (0.1% 單位、0.1 ~ 999.9%)	當輸入條件 ON 時, 只要 C+5 的位元 1 變成 1, 即可變更
C+2	積分常數 (Tik)	用來表示積分動作效果大小的常數。當此數值愈大時, 積分效果就愈弱。	10 進位表示時的 &1 ~ 8191 或是 16 進位表示時的 #0001 ~ 1FFF10 進位表示時的 &9999 或是 16 進位表示時的 #270F (設定為無積分動作) 當積分・微分常數的單位被指定為「1」時: 1 ~ 8191 倍; 若指定為「9」時: 0.1 ~ 819.1s (註 1)	
C+3	微分常數 (Tdk)	用來表示微分動作大小的常數。當此數值愈大時, 微分效果就愈強。	10 進位表示時的 &1 ~ 8191 或是 16 進位表示時的 #0001 ~ 1FFF10 進位表示時的 &0 或是 16 進位表示時的 #0000 (設定為無微分動作) 當積分・微分常數的單位被指定為「1」時: 1 ~ 8191 倍; 若指定為「9」時: 0.1 ~ 819.1s (註 1)	
C+4	取樣週期 ( $\tau$ )	設定執行 PID 演算的週期。	10 進位 &1 ~ 9999 或 16 進位 #0001 ~ 270F (10 ms 單位、0.01 ~ 99.99s)	不可
C+5 的位元 4 ~ 15	2-PID 參數 ( $\alpha$ )	輸入濾波器的係數。一般會設定為 0.65。當數值愈趨近 0 時, 表示濾波器愈無法發揮效果。	16 進位 #000: 當 $\alpha = 0.65$ (16 進位的 3 位數) 16 進位表示時為 #100 ~ 163 時, 表示下位 2 位數的數值將變成 $\alpha = 0.00 \sim 0.99$ 。(註 2)	
C+5 的位元 3	指定輸出操作量	指定測量值 = 設定值時的操作量。 (註)本設定只有在「無積分動作」時才有效。	0、1 0: 輸出 0% 1: 輸出 50%	
C+5 的位元 1	指定 PID 常數的反應時間	指定 P (比例區)、Tik (積分常數)、Tdk (微分常數) 等各種參數所要被反映在 PID 演算中的時間。	0、1 0: 只有在輸入條件上升時 1: 輸入條件上升時以及每個取樣週期	可
C+5 的位元 0	指定切換操作量的正反相動作	用來決定比例動作方向的參數。	0、1 0: 反相動作 1: 正向動作	不可
C+6 的位元 12	指定操作量的限制控制	指定是否針對操作量執行限制控制。	0、1 0: 無效 (不執行限制控制) 1: 有效 (執行限制控制) (註 3)	
C+6 的位元 8 ~ 11	輸入輸入範圍	輸入資料的位元數。	(16 進位 1 位數) 0: 8 位元 1: 9 位元 2: 10 位元 3: 11 位元 4: 12 位元 5: 13 位元 6: 14 位元 7: 15 位元 8: 16 位元	
C+6 的位元 4 ~ 7	指定積分 / 微分的單位	指定積分常數 / 微分常數的時間單位。	9 (16 進位 1 位數) 1: 將取樣週期倍數指定的積分・微分時間指定為取樣週期所指定的倍數時間。 9: 以 100 ms 為單位, 指定時間指定積分・微分時間。(註 1)	
C+6 的位元 0 ~ 3	輸出範圍	輸出資料的位元數。	設定範圍和輸入範圍相同。	
C+7	操作量的限制下限值	將操作量當作執行限制控制時的限制下限值。	10 進位 &0 ~ 65535 或 16 進位 #0000 ~ FFFF (註 3)	
C+8	操作量的限制上限值	將操作量當作執行限制控制時的限制上限值。	10 進位 &0 ~ 65535 或 16 進位 #0000 ~ FFFF (註 3)	
C+9 的位元 15	AT 指令 / 執行中	可同時當作 PID 常數的 AT (自動微調) 執行指令及 AT 執行中旗標使用。 ・執行 AT 時, 會被設定為 1 (執行 PIDAT 指令時同樣有效)。 ・當 AT 結束時, 就會自動回到 0。 (註)如果在執行 AT 時, 將數值由 1 設定為 0, 一旦 AT 中斷動作, 就會利用 AT 開始執行前的 PID 參數, 開始進行 PID 演算。不過, P、I、D 參數的數值只有在中斷時有效。	0、1 ・以 AT 執行指令來說 0 → 1: 指示執行 AT (執行 PIDAT 指令以及數值變成 1 時, 即指定執行 AT) 1 → 0: 指示中斷 AT 或是 AT 結束時, 數值就會自動改變 ・執行 AT 時的旗標 0: AT 非執行中 1: AT 執行中	可

控制資料	項目	內容	設定範圍	是否可在保持輸入條件 ON 的情況下變更
C+9 的位元 0 ~ 11	AT 計算增益值	需要以使用者定義的方式來調整 AT 進行 PID 調整的計算結果對於自動儲存值的貢獻度時，請設定本增益值，通常會使用預設值。 ·若重視穩定性，則需加大本數值。 ·若重視回應性，則需縮小本數值。	10 進位 &0 或 16 進位 #0000 : 1.00 (預設值) 10 進位 &1 ~ 1000 或 16 進位 #0001 ~ 03E8 : 0.01 ~ 10.00 (0.01 單位)	可 (不過，反映時間點為開始執行 AT 時)
C+10	限制週期的磁滯	執行 SV 時，必須設定啟動限制週期時的磁滯，預設值為執行反相動作時，利用 SV - 0.20% 的磁滯將 MV 啟動。此時將造成 PV 不穩定，因此一旦無法啟動正常的限制週期時，就必須加大此數值。不過，如果數值設得過大，反而會造成 AT 精確度不佳。	10 進位 &0 或 16 進位 #0000 : 0.20% (預設值) 10 進位 &1 ~ 1000 或 16 進位 #0001 ~ 03E8 : 0.01 ~ 10.00% (0.01 位) 16 進位 #FFFF : 0.00% (註)輸入範圍所對應的 %	

(註) 1 如果要將積分、微分常數的指定單位用來指定時間 (「9」) 時，積分時間・微分時間必須設定在取樣週期 1 ~ 8191 倍的時間範圍內。  
2 設定為 000 時，2-PID 參數就會變成 0.65，通常請設定為 000。  
3 將操作量限制控制的指定設定為有效 (「1」) 時，必須依下列條件來設定各項數值。  
0000 ≤ 操作量限制下限值 ≤ 操作量限制上限值 ≤ 指定輸出範圍的最大值

### 取樣週期和週期時間之間的關係

取樣週期可指定為以 10ms 為單位 (0.01 ~ 99.99s)。不過，實際在進行 PID 演算時，則依取樣週期及 PID 指令執行時 (週期別) 的組合而異，取樣週期和週期時間的關係如下：

1) 當取樣週期 < 週期時間時

會在每個週期時間執行 PID 演算，而不會依週期別執行 PID 演算。

2) 當取樣週期 ≥ 週期時間時

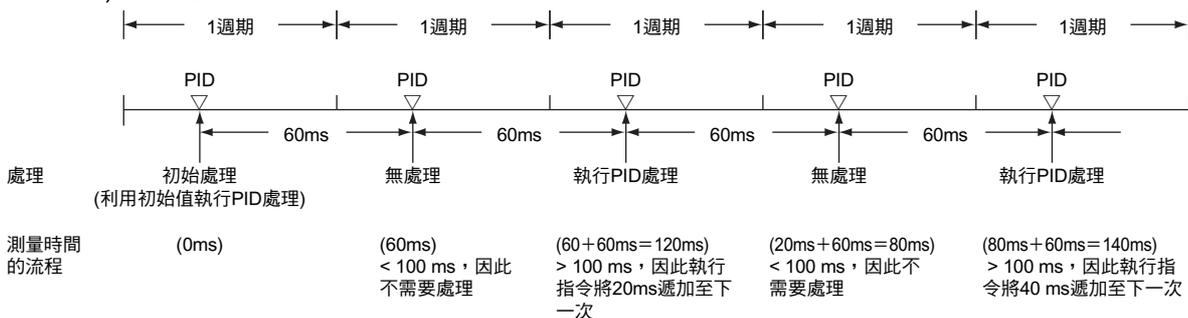
不執行每個週期的 PID 演算，而是當週期時間 (PID 指令 -PID 指令間隔的時間) 的累計值 ≥ 取樣週期時，才會執行 PID 指令。累計值超過的部分 (週期時間的累計值 - 取樣週期) 就會被遞加到下一個累計值。

例如，當取樣週期為 100 ms 且取樣時間固定為 60 ms 時，由於執行初始動作後的第一次週期為 60 ms < 100 ms，因此不會執行 PID 指令。

第 2 次為 60 ms + 60 ms > 100 ms，所以會開始執行 PID 指令。(超過部分 120 ms - 100 ms = 20 ms 會被遞加至下一次週期)。

第 3 次會遞加前一次超出的部分，並且以 20 ms + 60 ms 來計算，此時結果為 80 ms < 100 ms，因此不會執行 PID 指令。

第 4 次為 80 ms + 60 ms > 100 ms，因此會執行 PID 指令。(超過部分 140 ms - 100 ms = 40 ms 會被遞加至下一次週期) 以下相同。

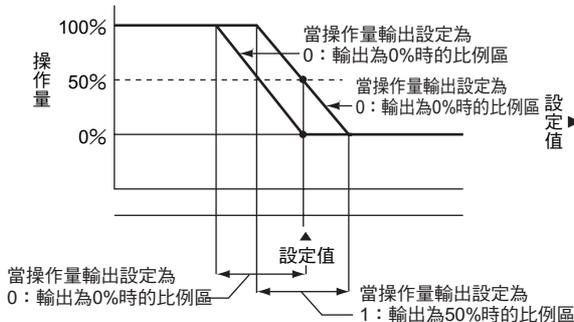


## PID 控制

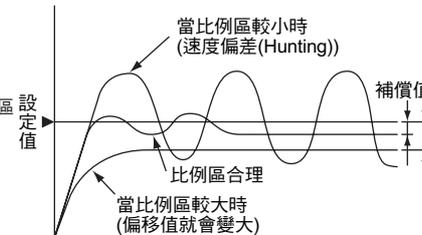
### ● 比例動作 (P)

具有可對應至設定值的比例區，其中，操作量 (控制量) 和偏差具有比例關係的動作就稱之為「比例動作」。當現在值小於比例區時，操作量為 100%。比例區內的操作量會和偏差呈比例關係縮小，而且當設定值和現在值一致 (偏差為 0) 時，操作量將依照操作量輸出的指定參數，變為 0% 或是 50%。(執行反相動作時) 比例區會以相對於整個輸入範圍的百分比的方式來表示。當比例動作發生偏移 (殘留偏差) 時，只要將比例區縮小，即可減少偏移的情形，不過當數值過小時，反而會造成速度偏差 (Hunting) 的情形。

執行比例動作(反相動作)



調整比例區



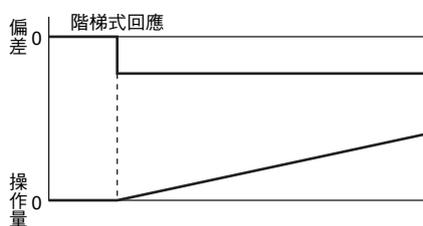
### ● 積分動作 (I)

執行比例動作時會發生偏移的情形。這時候，只要將比例動作搭配積分動作，經過一段時間後，就能夠減少偏移，並且達到測量值和設定值一致的目標。

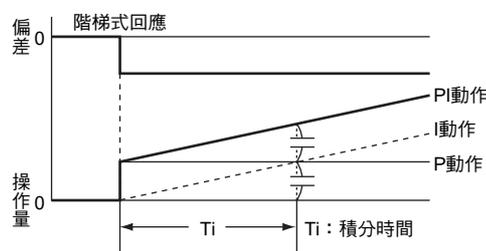
#### ● 積分時間

用來表示積分動作強弱的單位，如圖所示，也就是相對於階梯形偏差，積分操作量和比例動作達到相同操作量所需要的時間。積分時間愈短，愈需要利用積分動作進行修正，不過，如果時間過短時，就會造成修正動作過大，因而導致速度偏差 (Hunting) 發生。

積分動作



PI動作與積分時間



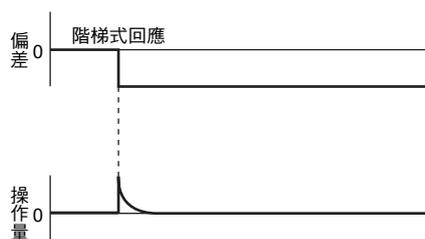
### ● 微分動作 (D)

比例動作或積分動作都是針對控制結果所做的修正動作，因此無論如何都會造成回應時間變長。為了改善此項缺點，一旦遇到劇烈的外部干擾時，微分動作的作用就是提供較大的操作量，以便讓系統回到原先的狀態。接著，再根據和偏差所產生的傾斜 (微分係數) 呈等比關係的操作量來執行修正動作。

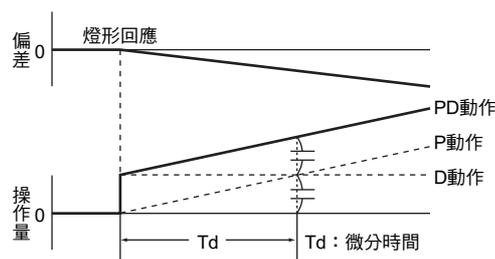
#### ● 微分時間

用來表示積分動作強弱的單位，如圖所示，也就是相對於階梯形偏差，積分操作量和比例動作達到相同操作量所需要的時間。微分時間愈短，愈需要利用微分動作進行修正。

微分時間



PD動作及微分時間

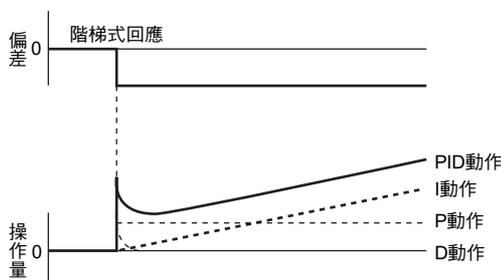


## PID 動作

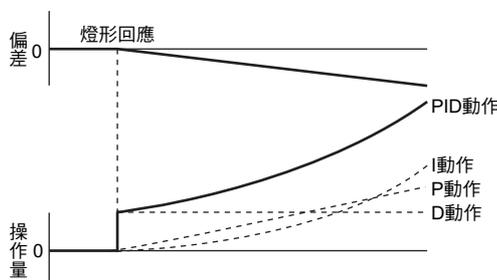
PID 動作就是將比例動作 (P)、積分動作 (I) 及微分動作 (D) 加以組合後的動作，即使控制對象會發生浪費時間，也能產生絕佳的控制結果。藉由比例動作可執行無速度偏差 (Hunting) 的控制動作，積分動作可自動修正偏移，而微分動作則能夠提高對於外部干擾的回應速度。

以下將介紹遇到階梯形偏差、燈形偏差時 PID 動作在操作量上的變化。

PID動作的階梯形回應



PID動作的燈形回應

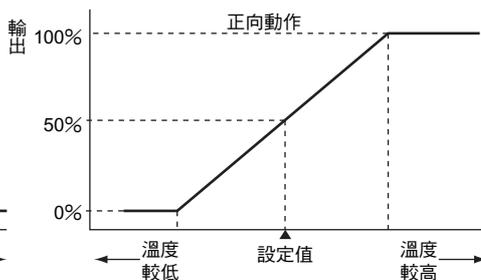
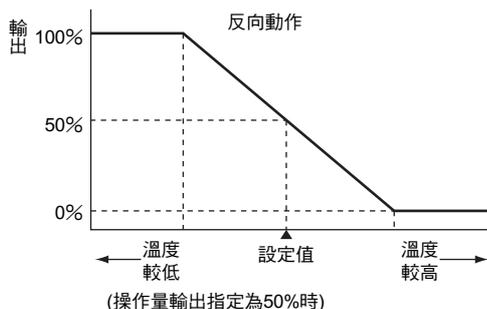


## 動作方向

執行 PID 控制時，請選擇下列 2 種控制方向中的任一種。

反相動作：當測量值小於設定值時，即增加操作量。(加熱)

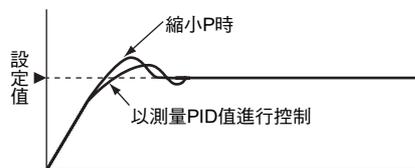
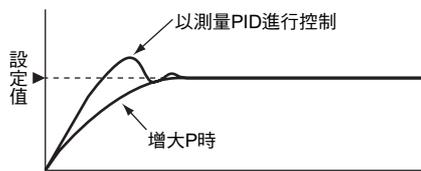
正向動作：當測量值大於設定值時，即增加操作量。(冷卻)



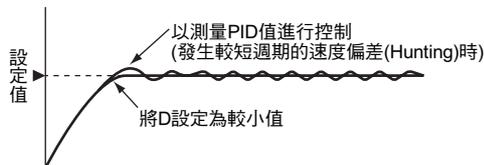
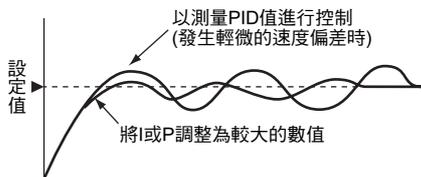
## PID 參數調整

以下為 PID 參數與控制狀態之間的一般關係，提供您做為使用 PID 指令時的參考。

- 即使進入穩定狀態前需要一些時間 (穩定時間)，也不希望發生過衝的情形時，請增大比例區。
- 即使發生過衝，仍希望能夠盡快進入穩定的控制狀態時，請縮小比例區 (P)，不過，比例區過小時，反而會造成速度偏差 (Hunting)。

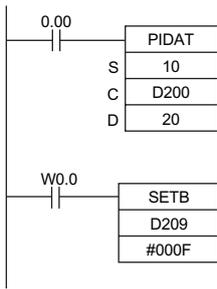


- 一旦發生輕微的速度偏差 (Hunting) 或是因為反覆過衝、下衝而造成收斂的情形時，有可能是因為積分動作過強的緣故。此時只要加長積分時間 (I)，或是加大比例區 (P)，即可減少速度偏差 (Hunting) 發生。
- 發生較短週期的速度偏差 (Hunting) 時，有可能是控制類元件的回應過快，造成微分動作太強所造成的，此時，請縮短微分時間 (D)。

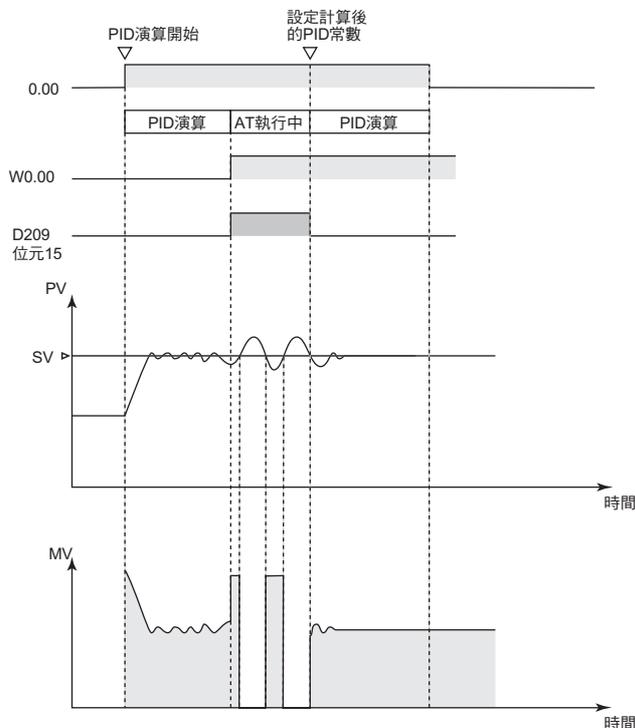
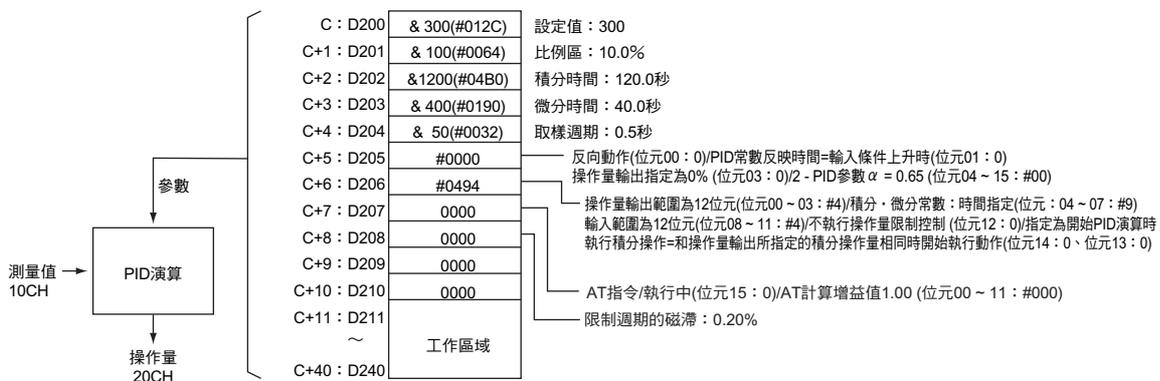


程式例

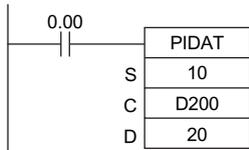
● 執行 PID 指令後，如需利用其他條件來執行 AT 時



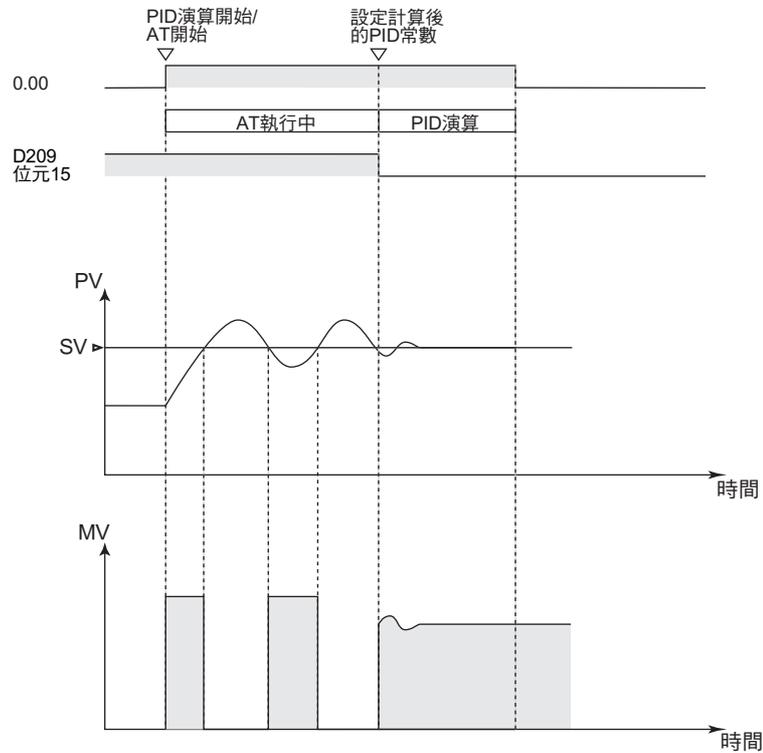
- 當 0.00 上升 (OFF → ON) 時，將根據 D200 ~ D208 所設定的下列參數來執行 D211 ~ D240 工作區域的初始化 (清除) 動作。完成初始化後，就會開始執行 PID 演算，並且將操作量輸出至 20CH。
- 當 0.00 變成 ON 時，將根據 D200 ~ D210 所設定的參數，並且依取樣週期的間隔來執行 PID 演算，然後再將操作量輸出至 20CH。
- 如果在 0.00 上升後才變更比例區 (P)、積分常數 (Tik)、微分常數 (Tdk) 等各項參數，結果將不會被反映在 PID 演算中。
- 當 W0.0 由 ON → OFF 時，就會利用 SETB 指令，將 D209 (C+9) 的位元設定為 1 (ON)，並且開始執行 AT。當 AT 執行完畢後，就會將計算出來的 P、I、D 常數分別設定在 C+1、C+2、C+3 中，接著再根據各項 PID 常數進行 PID 演算。



● 執行 PID 指令同時立刻執行 AT 時

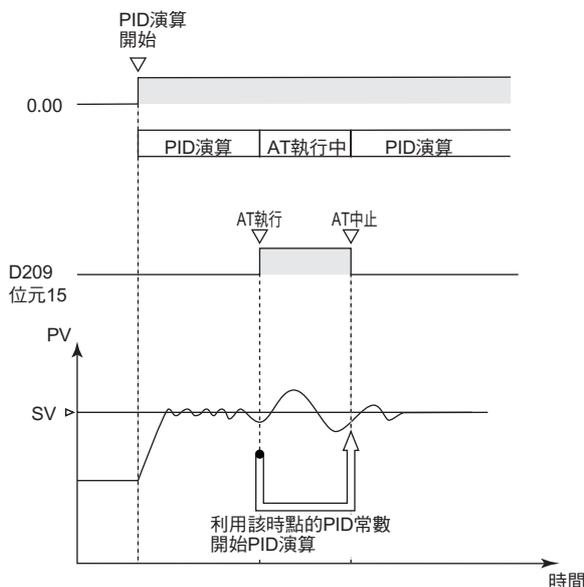


- 當 0.00 變成 ON 或是 D209 (C+9) 的位元 15 變成 1 (ON) 時，就會先執行 AT。當 AT 執行完畢後，就會將計算出來的 P、I、D 常數分別設定在 C+1、C+2、C+3 中，接著再根據各項 PID 常數進行 PID 演算。



● 在 AT 執行過程中中斷 AT 動作

執行 AT 時，如果將 09 (C+9) 的位元 15 由 1 (ON) 變更為 0 (OFF)，將造成 AT 動作中斷，此時，系統將利用開始執行 AT 前的 PID 常數重新開始 PID 演算。



# TPO

指令名稱	指令記號	指令的各種組合	Fun No.	功能
時間分割比例輸出	TPO	—	685	可輸入指定 CH 中的負載 (Duty) 比或操作量，並且根據指定的參數，將負載比轉換為時間分割比例輸出，然後再將脈衝輸出至指定的接點。

符號	TPO	
		<p>S：儲存輸入負載比或操作量的CH編號</p> <p>C：儲存參數的下位CH編號</p> <p>R：輸出脈衝的繼電器編號</p>

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	儲存輸入負載比或操作量的 CH 編號	UINT	1
C	儲存參數的下位 CH 編號	WORD	7
R	輸出脈衝的繼電器編號	BOOL	—

S：儲存輸入負載比或操作量的 CH 編號。

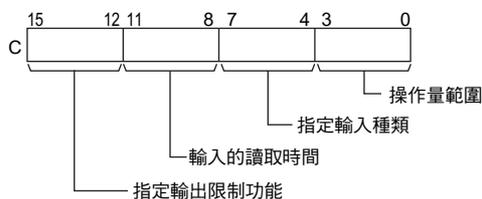
指定用來儲存輸入負載比或操作量的 CH。

- 輸入負載比：10 進位 & 0 ~ 10000 或 16 進位 #0000 ~ 2710 (0.00 ~ 100.00%)
- 輸入操作量^{*1}：10 進位表示時的 & 0 ~ 最大 65535 或是 16 進位表示時的 #0000 ~ 最大 FFFF (利用 C 的位元 00 ~ 03，指定可當作有效位數的操作量範圍，該範圍必須和 PID 演算指令的操作量輸出範圍一致。)

* 1 將 S 當作輸入操作量時，必須利用附自動微調功能的 PID 演算指令 (PIDAT) 來指定用來儲存操作量的 CH 編號。

C：儲存參數的下位 CH 編號

利用 C 的位元 04 ~ 07 (指定輸入種類)，來指定此編號要用來當作輸入負載比、或是輸入操作量的其中任一種資料來使用 (當 C 的位元 04 ~ 07 為 0 時，就當作負載比，如果 C 的位元 04 ~ 07 為 1 時，則當作操作量使用)



(註) 詳情請參閱後述的參數設定內容。

R：輸出脈衝的繼電器編號

指定脈衝輸出的輸出目的接點。

通常會指定電晶體輸出模組的配置接點，然後再將 SSR (固態繼電器) 連接至電晶體輸出模組。

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S					○	○	○	○	○	○			
C	○	○	○	○									
R					-	-	-	-	-	-			

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S 超出範圍時，本旗標就會 ON (註) 範圍將依輸入種類而改變</li> <li>當 C 超出範圍時，本旗標就會 ON (註) 操作範圍僅適用於輸入種類指定為操作量時</li> <li>當控制週期 (C+1) 超出範圍時，本旗標就會 ON</li> <li>輸出限制功能啟動，且輸出限制下限值 (C+2)、輸出限制上限值當 (C+3) 超出範圍時，本旗標就會 ON</li> <li>輸出限制功能啟動，而且不適用於輸出限制下限值 (C+2) ≤ 輸出限制上限值 (C+3) 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

功能

輸入 S 所指定的 CH 編號中的負載 (Duty) 比或操作量，並且利用 C~C+3 所指定的參數，將負載比轉換為時間分割比例輸出 *1，接著再將脈衝輸出至 R 所指定的接點。

* 1 所謂「時間分割比例輸出」就是根據輸入值 (S) 大小，按照比例來調整 ON、OFF 時間比的一種輸出方式。改變 ON 和 OFF 時間比的週期就稱為「控制週期」，可利用 C+1 來指定。

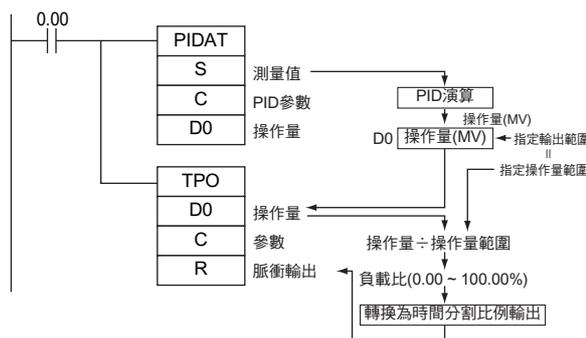
例) 當控制週期 = 1s、輸入值為 50% 時，會在 0.5s 的期間 ON，0.5s 的期間 OFF，當輸入值為 80% 時，則會在 0.8s 的期間 ON，0.2s 的期間 OFF。

一般來說，可以和附自動微調功能的 PID 演算指令 (PIDAT) 搭配使用，而操作量輸出結果 (D) 則可利用本指令的 S 來指定。

而且只要為 R 指定電晶體輸出模組的配置接點，並且將 SSR (固態繼電器) 連接至該電晶體輸出模組，即可執行加熱器的時間分割比例控制 (將 ON 和 OFF 依時間比進行比例控制)。

● PID 演算指令的搭配範例

搭配 PID 演算指令使用時，系統會根據輸入操作量，將操作範圍除以操作量後所得到的數值當作負載比，接著再將該負載比轉換為時間分割比例輸出，並輸出脈衝。

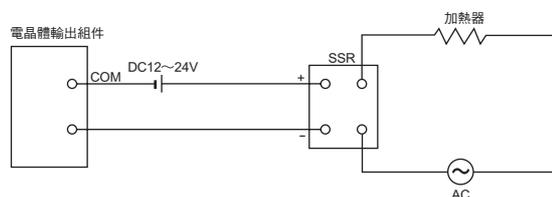


此時，必須先將 PID 演算指令的輸出範圍和本指令的操作量範圍設定為一致。

例) 當輸出範圍 = 操作量範圍被指定為 12 位元 (16 進位表示時的 #0000 ~ 0FFF) 時，利用 PID 演算指令以 16 進位的 #0FFF 除以操作量後所得到的數值，就會被當作負載比，本指令會將該數值轉換為時間分割比例輸出。

● 外部配線範例

電晶體輸出模組和 SSR 的連接方式如下：



## 參數設定內容

	項目	內容	設定範圍	是否可在保持輸入條件 ON 的情況下變更
C 的位元 0 ~ 3	操作量範圍	輸入資料的位元數。	(16 進位 1 位數) 0 : 8 位元 1 : 9 位元 2 : 10 位元 3 : 11 位元 4 : 12 位元 5 : 13 位元 6 : 14 位元 7 : 15 位元 8 : 16 位元	可
C 的位元 4 ~ 7	指定輸入種類	選擇負載比或操作量作為輸入 (S) 的資料。	0 : 負載比 (註) S 值的範圍 : 負載比 : 10 進位 & 0 ~ 10000 或 16 進位 #0000 ~ 2710 (0 ~ 100.00%) 1 : 操作量 (註) S 值的範圍 : 10 進位 & 0 ~ 65535 或 16 進位 #0000 ~ FFFF。 上限是以操作範圍 (C 的位元 0 ~ 3) 為基準	可
C 的位元 8 ~ 11	指定輸入讀取時間	指定輸入的讀取時間。	0 : 每個控制週期 1 : 下方優先 2 : 上方優先 3 : 常時	
C 的位元 12 ~ 15	指定輸出限制功能	指定輸出限制功能啟動 / 關閉。	0 : 無效 (不執行限制控制) 1 : 有效 (執行限制控制)	可
C+1	控制週期	控制週期 (改變 ON 和 OFF 時間比的週期)	10 進位 &100 ~ 9999 或 16 進位 #0064 ~ 270F (1.00 ~ 99.99 秒) (註) 例如, 如果是 1 秒時, 請設定為 10 進位表示時的 &100 而非 &1, 此點需特別注意。	可
C+2	輸出限制的下限值	將輸出當作執行限制控制時的限制下限值。	10 進位 & 0 ~ 10000 或 16 進位 #0000 ~ 2710 (0 ~ 100.00%)	可
C+3	輸出限制的上限值	將輸出當作執行限制控制時的限制上限值。	10 進位 & 0 ~ 10000 或 16 進位 #0000 ~ 2710 (0 ~ 100.00%)	可
C+4	工作區域	系統所使用的區域。	無法使用	—
C+5		使用者無法使用此 CH。		
C+6				

(註) 將輸出限制控制的指定設定為有效 (「1」) 時, 必須依下列條件來設定各項數值。  
10 進位的 &0 ≤ 輸出限制下限值 ≤ 輸出限制的上限值 ≤ 10 進位的 &10000

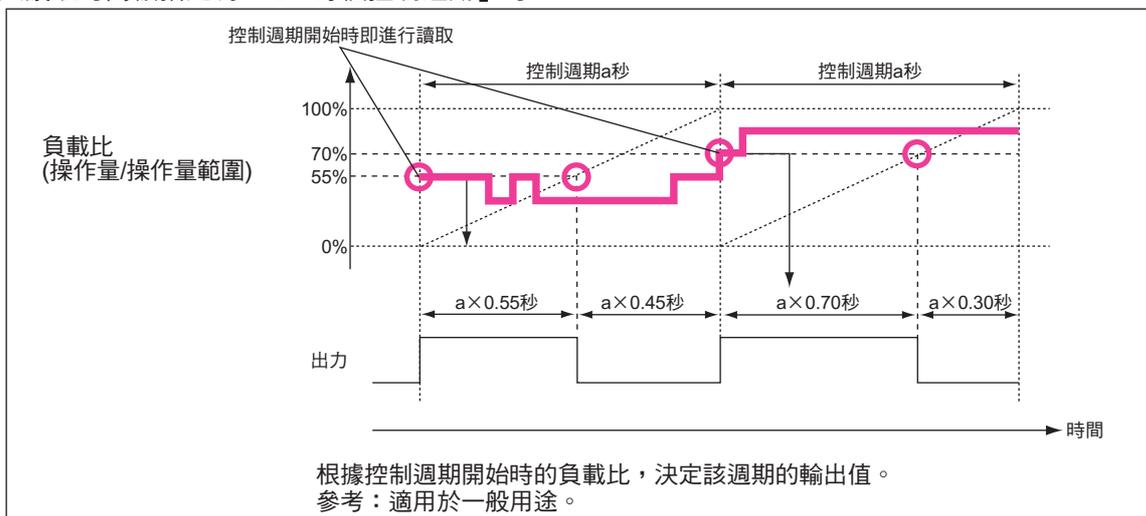
## 功能說明

- 輸入條件 ON 時，執行本指令。
- 開始執行指令時，將根據負載比，將輸出接點 (R) 設定為 ON/OFF。
- 系統會在每次執行指令時即時讀取參數 (C ~ C+3)，如需變更某一項參數時 (請避免破壞每個參數之間的一致性)，請同時變更其他的參數。
- 指令會在輸出 (R) ON/OFF 時執行，又，輸出 (R) ON/OFF 的時間精確度最大為 10 ms。
- 當輸入條件 OFF 時，就會中斷指令的執行動作，此時，經過時間會被復歸，而控制週期也會被初始化。
- 利用指定輸入種類 (C 的位元 04 ~ 07) 的方式，選擇以負載比或是操作量作為輸入 (S) 的資料。
- 當輸入 (S) 的資料被設定為操作量時，就會以操作量 (S) ÷ 操作量範圍 (根據 C 的位元 0 ~ 3) 的公式來計算負載比。
- 欲指定輸入 (S) 的讀取時間時，請利用輸入讀取時間 (C 的位元 8 ~ 11) 指定功能。

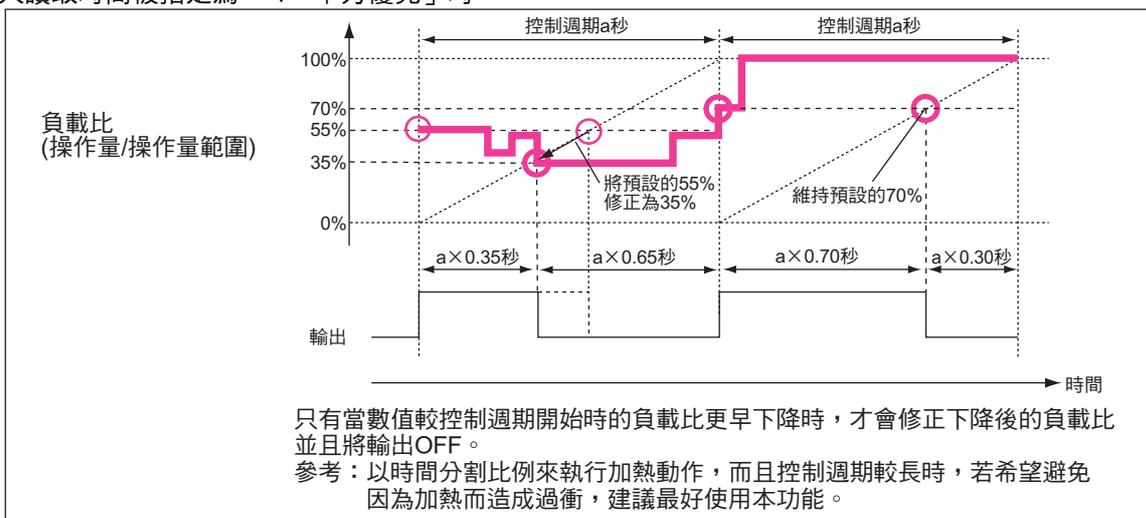
指定為「0：每個控制週期」時	會在控制週期開始的時點讀取負載比輸入，週期過程中所發生的負載比變化皆會被忽略。
指定為「1：下方優先」時	當控制週期開始時的負載比低於負載比輸入值時，將以下降的數值為優先，向下修正 ON 時間。
指定為「2：上方優先」時	當負載比輸入值高於控制週期開始時的負載比時，將以上升的數值為優先，向上修正 ON 時間。
指定為「3：常時」時	在每次執行指令時讀取負載比，並且在控制週期過程中反覆執行 ON/OFF。

請參閱下圖說明。

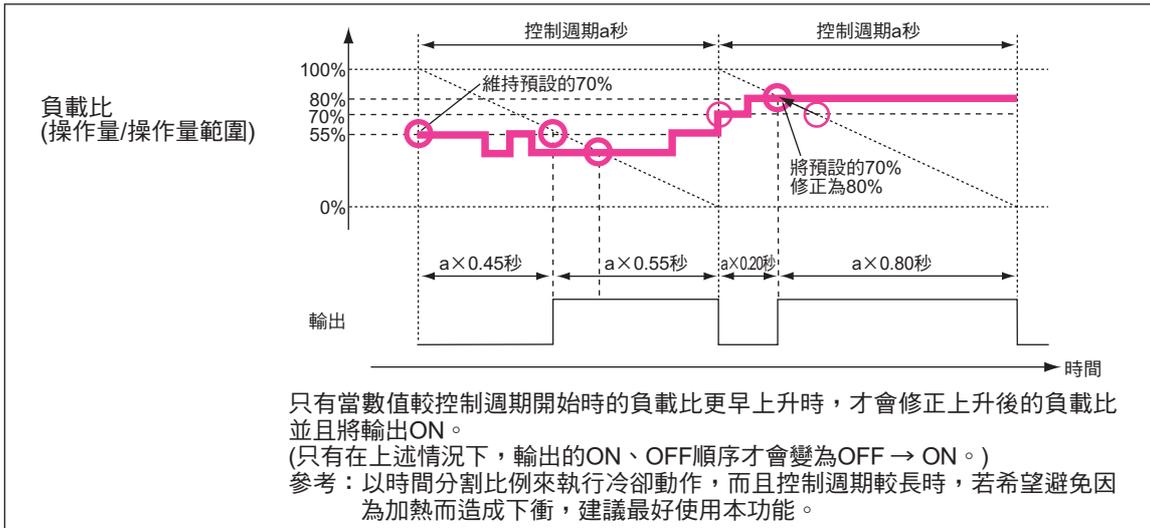
- 當輸入讀取時間被指定為「0：每個控制週期」時



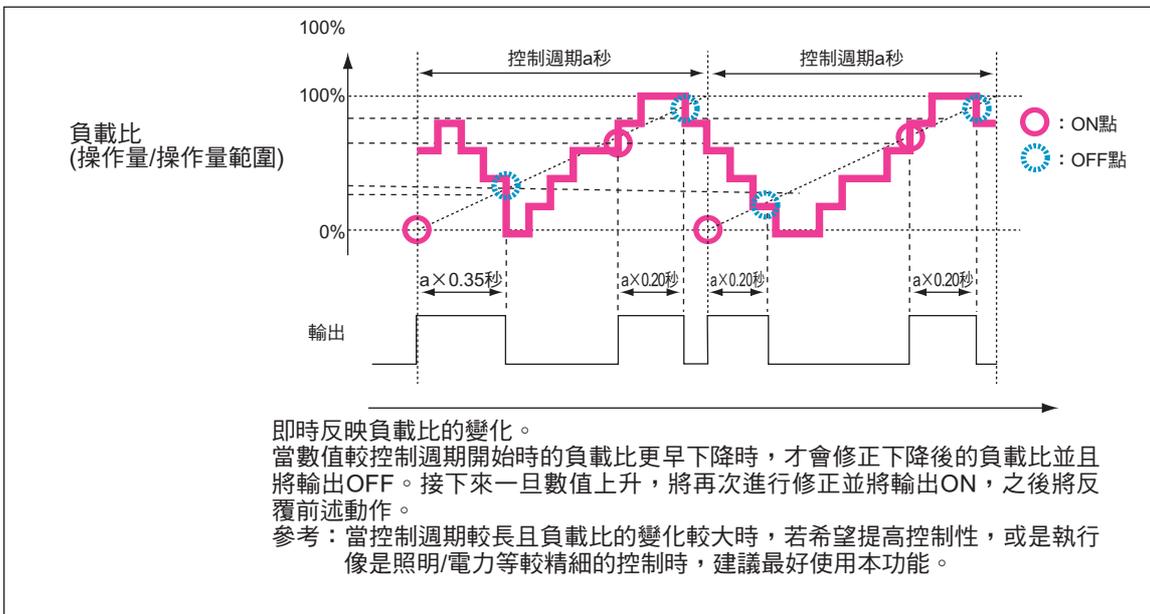
- 當輸入讀取時間被指定為「1：下方優先」時



- 當輸入讀取時間被指定為「2：上方優先」時



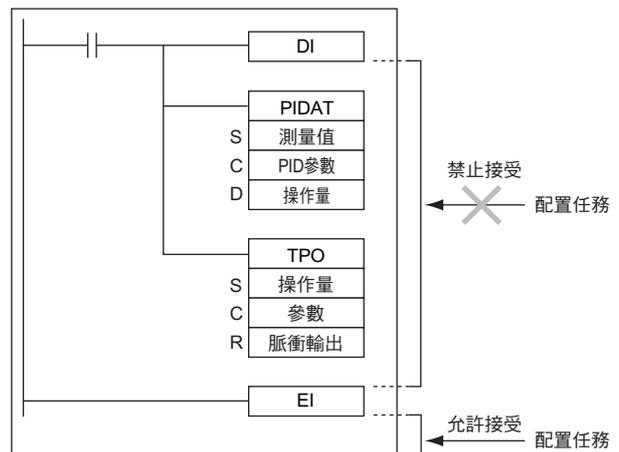
- 當輸入讀取時間被指定為「3：常時」時



- 啟動輸出限制功能(C的位元12~15)後，一旦輸出超出輸出限制的下限值(C+2)及輸出限制的上限值(C+3)範圍，就會限制輸出(讓輸出飽和)。

## 使用上的注意事項

- 在執行週期任務的過程中，將本指令搭配PIDAT指令使用，同時還使用配置任務時，請在PIDAT指令及本指令之前執行DI(禁止執行配置任務)，以禁止配置任務的執行。一旦在PIDAT指令和本指令之間加入配置，有可能會造成控制週期偏差的情形。

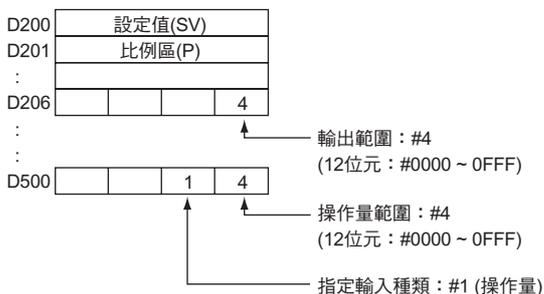
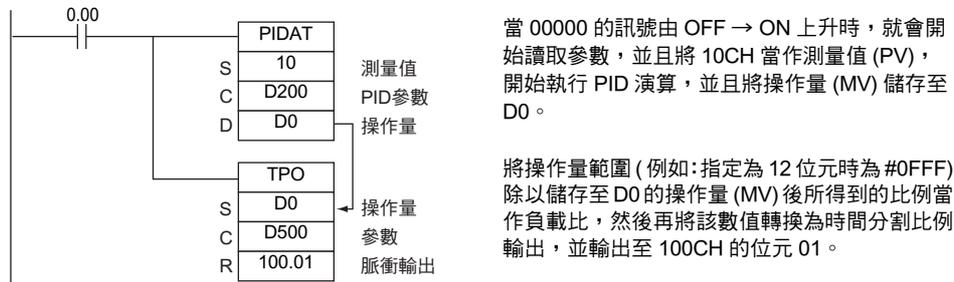


### 程式例

#### ● 搭配 PIDAT 指令使用時

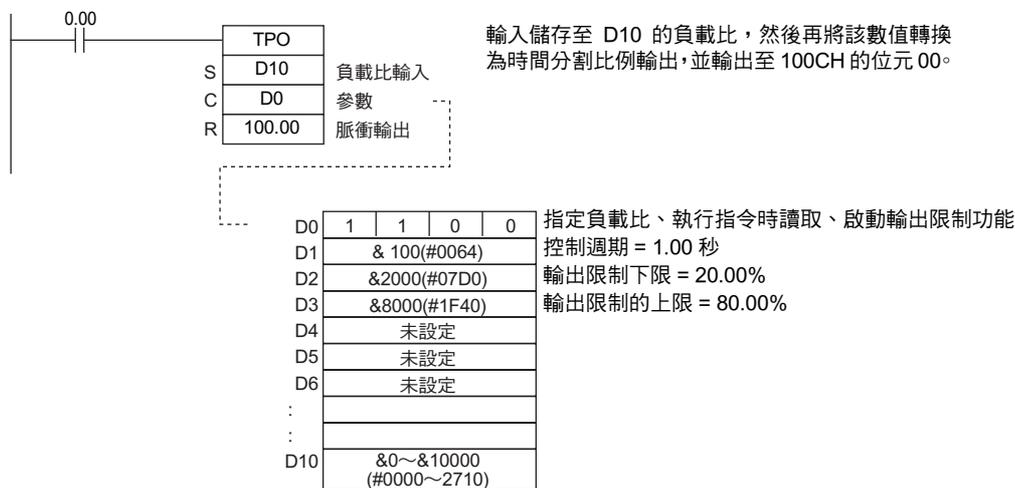
當 0.00 變成 ON 時，就會輸入 PIDAT 演算指令所輸出的輸出操作量 (儲存至 D0)，接著根據該操作量來計算負載比 (操作量 ÷ 操作量範圍)，並將該數值轉換為時間分割比例輸出，然後輸出脈衝至 100CH 位元 01。

又，將電晶體輸出模組配置到 100CH，然後再將 SSR (固態繼電器) 連接至位元 01 的端子，以執行加熱器控制。



#### ● 單獨使用本指令時

當 0.00 變成 ON 時，就會輸入 D10 的負載比，然後再將該數值轉換為時間分割比例輸出，並輸出脈衝至 100CH 的位元 00。以控制週期 = 1 秒、輸出限制功能啟動 (下限 20.00%、上限 80.00%) 時為例：



# SCL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
轉換 (Scaling)	SCL	@ SCL	194	根據所指定的一次函數，將不帶符號的 BIN 資料轉換 (Scaling) 為帶符號的 BCD 資料。

符號	SCL	
		S : 欲轉換的CH編號 C : 儲存參數的下位CH編號 D : 儲存轉換結果用的CH編號

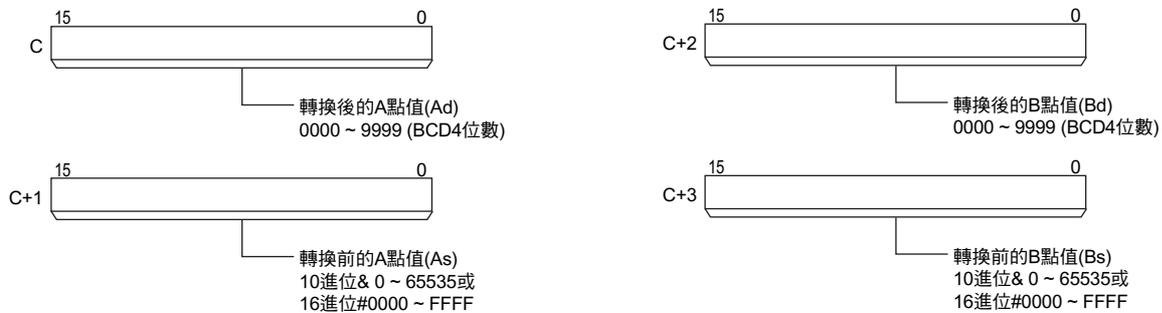
## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	欲轉換的 CH 編號	UINT	1
C	儲存參數的下位 CH 編號	LWORD	4
D	儲存轉換結果用的 CH 編號	WORD	1

### C : 儲存參數的下位 CH 編號



(註) C ~ C+3 的運算元種類必須相同。

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S,C,D	○	○	○	○	○	○	○	○	○	—	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P.ER	<ul style="list-style-type: none"> <li>當 Ad (C) 和 Bd (C+2) 的值非 BCD 時，本旗標就會 ON</li> <li>當 As (C+1) 和 Bs (C+3) 的數值相等時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
= 旗標	P.EQ	<ul style="list-style-type: none"> <li>當轉換結果為 0 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>

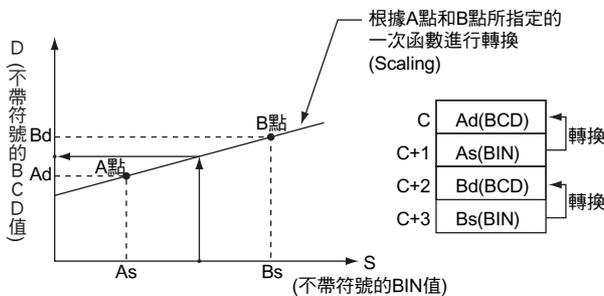
## 功能

根據由 C 所指定參數 (A 點和 B 點此 2 點在執行各種轉換前後的數值) 構成的一次函數，將 S 所指定的不帶符號 BIN 資料轉換為不帶符號的 BCD 資料，然後再將結果輸出至 D。

轉換公式：
$$D = Bd - \frac{(Bd - Ad)}{(Bs - As) \text{的BCD轉換值}} \times (Bs - S) \text{的BCD轉換值}$$

A 點和 B 點除了正斜率外，也有可能是負斜率，因此，也能夠進行反相轉換。

- 將轉換後的資料小數點以下全部四捨五入。
- 當轉換結果小於 0000 時，就輸出 0000，如果大於 9999，就輸出 9999。

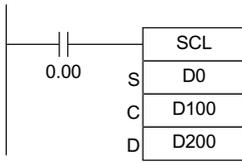


## 提示

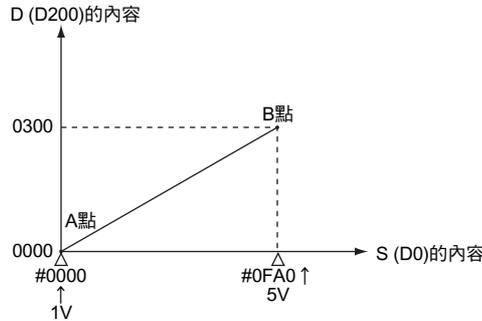
- 如果要將類比輸入模組所產生的類比訊號轉換結果轉換為使用者所定義的刻度單位時，建議最好使用本指令。(例：將 1 ~ 5V 所對應的 10 進位的 &0 ~ 4000 或 16 進位的 #0000 ~ 0FA0 轉換為 50 ~ 200°C)
- 本指令可將不帶符號的 BIN 轉換 (Scaling) 為不帶符號的 BCD。  
因此，如果原先的資料 S 包含負數時，就必須重新編寫程式，才能夠提高該數值 (加上最大的負值) 後再進行轉換。又，負數無法被輸出至轉換結果 D。當轉換 (Scaling) 結果為負數時，則會輸出 0。

### 程式例

將類比訊號 1 ~ 5V 所對應的 16 進位的 #0000 ~ 0FA0 數值儲存至 D0 時，該數值將會被轉換 (Scaling) 為 0 ~ 300 的 BCD 值。



當 0.00 變為 ON 時，將根據 A 點 (#0000 → 0 (BCD)) 和 B 點 (#0FA0 → 300 (BCD)) 所構成的一次函數來轉換類比輸入模組所輸出的 D0 值，然後再將結果儲存至 D200。



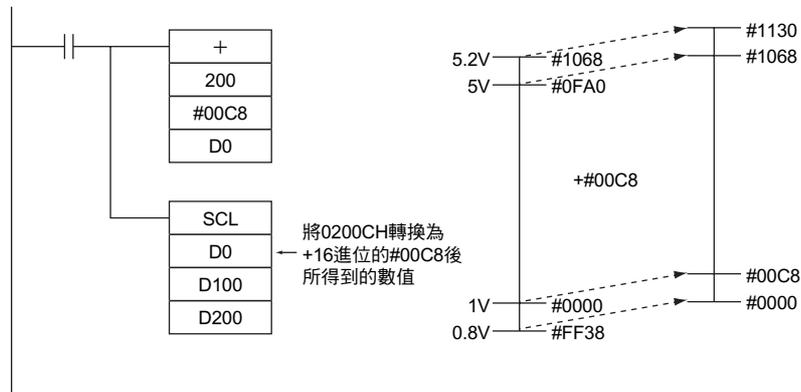
C : D100	0 0 0 0	Ad(BCD)
C+1 : D101	0 0 0 0	As(BIN)
C+2 : D102	0 3 0 0	Bd(BCD)
C+3 : D103	0 F A 0	Bs(BIN)

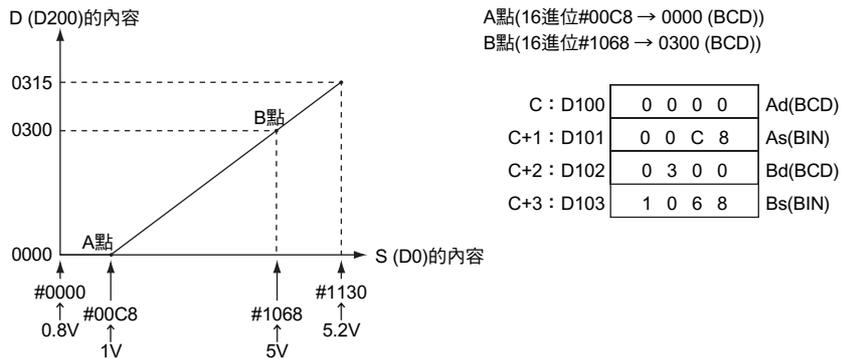
#### 參考

類比輸入模組所產生的實際變更值將會以對比 0.8V ~ 5.2V 的 16 進位 #FF38 ~ 1068 數值的方式加以儲存。

不過，這時候 SCL 指令會將 S 值當作 16 進位 #0000 ~ FFFF 的不帶符號 16 進位數值處理，因此對於小於 1V (#0000) 的帶符號 16 進位數值 #FF38 ~ FFFF 將無法正確執行轉換 (Scaling)。

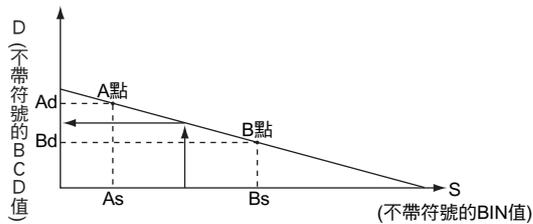
所以，實際利用 SCL 指令來轉換類比輸入模組所產生的轉換值時，會先將 16 進位 #00C8 和所有數值相加，以便讓 16 進位的 #FF38 變成 #0000，然後再執行 SCL 指令。



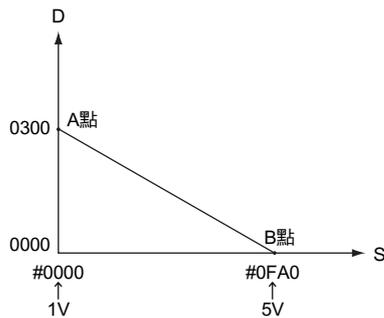


不過，使用 SCL 指令時，0 ~ 9999 的不帶符號 BCD 值將會被儲存在 D 值中。這時候，就必須對 0.8 ~ 5.2V，進行原始值為 -15 ~ +315 的轉換，不過當轉換結果為負數時，就會儲存為 0(BCD)，因此如上圖所示，一旦數值為 16 進位的 #0000 ~ #00C8 時，轉換結果就會變成 0(BCD)。

參考：反相轉換時，也有可能出現  $As < Bs$ 、 $Ad > Bd$  的關係。



例如，對應至 1 ~ 5V (16進位的#0000 ~ 0FA0)時，亦可轉換為0300 ~ 0000的BCD數值(反向轉換)。



# SCL2

指令名稱	指令記號	指令的各種組合	Fun No.	功能
轉換 (Scaling) 2	SCL2	@ SCL2	486	根據所指定的附偏移量一次函數，將帶符號的 BIN 資料轉換 (Scaling) 為帶符號的 BCD 資料。

符號	SCL2	
		S：欲轉換的CH編號 C：儲存參數的下位CH編號 D：儲存轉換結果用的CH編號

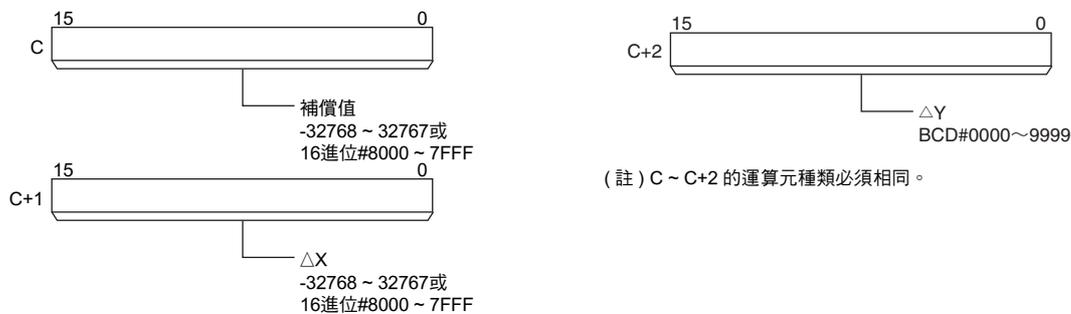
## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	欲轉換的 CH 編號	INT	1
C	儲存參數的下位 CH 編號	WORD	3
D	儲存轉換結果用的 CH 編號	WORD	1

### C：儲存參數的下位 CH 編號



## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S,C,D	○	○	○	○	○	○	○	○	○	—	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 $\Delta X$ (C+1) 的資料為 0 時，本旗標就會 ON</li> <li>當 $\Delta Y$ (C+2) 的資料非 BCD 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當轉換結果為 0 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
進位旗標 (Carry flag)	P_CY	<ul style="list-style-type: none"> <li>當轉換結果為負數時，本旗標 ON</li> <li>否則為 OFF</li> </ul>

## 功能

根據由 C 所指定參數 ( 斜率和偏移量 ) 所構成的一次函數，將 S 所指定的帶符號 BIN 資料轉換為帶符號的 BCD 資料 (BCD 資料為絕對值，利用 CY 旗標判別正負，旗標 ON 時：負數，OFF 時：正數)，然後再將結果輸出至 D。

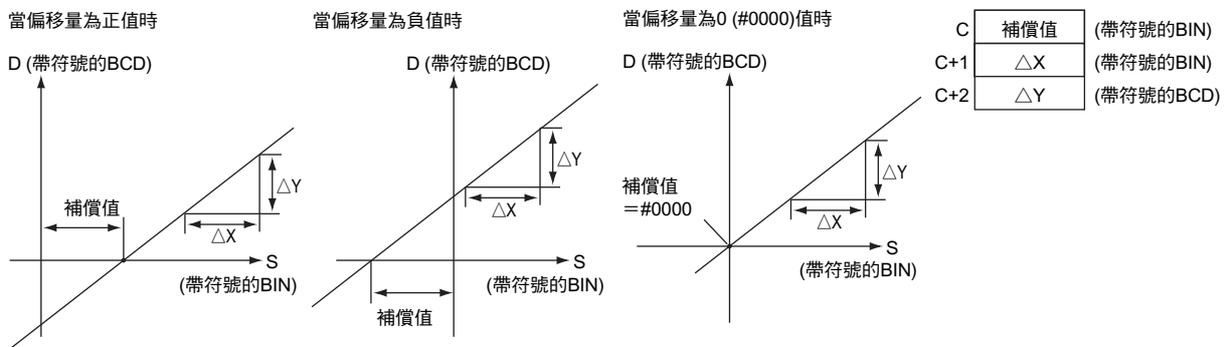
轉換公式：
$$D = \frac{\Delta Y}{\Delta X \text{的BCD轉換值}} \times \{(S \text{的BCD轉換值}) - (\text{偏移量的BCD轉換值})\}$$

(註)  $\frac{\Delta Y}{\Delta X}$  表示斜率。

偏移量有可能是正數、0 或是負數。

斜率也同樣可能是正數、0 或是負數，因此，可執行反相轉換。

- 將轉換後的資料小數點以下全部四捨五入。
- D (BCD 資料) 代表絕對值，並且以進位旗標來表示正負數。因此轉換結果的輸出範圍為 -9999 ~ 9999。
- 當轉換結果超出上限 (9999) 時，就會輸出 9999，如果超出下限時，則會輸出 -9999。



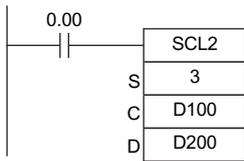
## 提示

- 如果要將類比輸入模組所產生的類比訊號轉換結果轉換為使用者所定義的刻度單位時，建議最好使用本指令。(例：將 1 ~ 5V 所對應的 10 進位的 &0 ~ 4000 或 16 進位的 #0000 ~ 0FA0 轉換為 -100 ~ 200°C)
- 本指令可將帶符號的 BIN 轉換 (Scaling) 為帶符號的 BCD。
- 因此，即使原始資料 S 出現負數時，也能夠持續進行轉換 (Scaling)。
- 而且，也可以利用轉換結果 D 和進位旗標，將負數輸出至轉換結果。

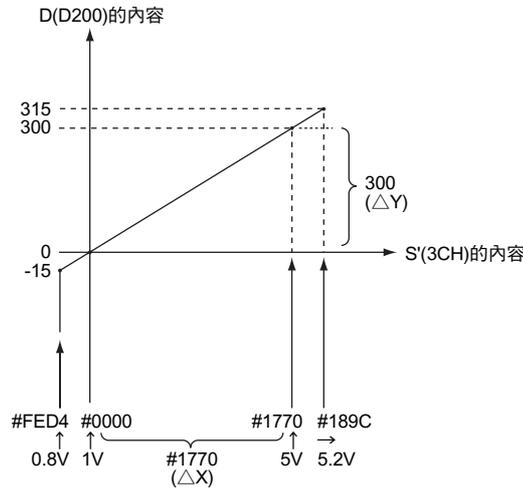
## 程式例

## ● 將類比訊號 1 ~ 5V 轉換為 0 ~ 300 時

當對應類比訊號 1 ~ 5V 的帶符號 16 進位數值 #0000 ~ #1770 被儲存至 3CH 時，該數值就會被轉換 (Scaling) 為 0 ~ 300 的 BCD 值。



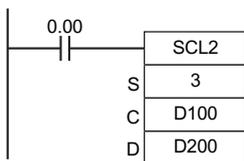
當 0.00 變成 ON 時，就會根據由偏移量 = #0000  $\Delta X = 16$  進位的 #1770  $\Delta Y = \text{BCD}\#0300$  所構成的一次函數進行轉換 (Scaling)，將類比輸入模組所輸出的 3CH 值儲存至 D200。



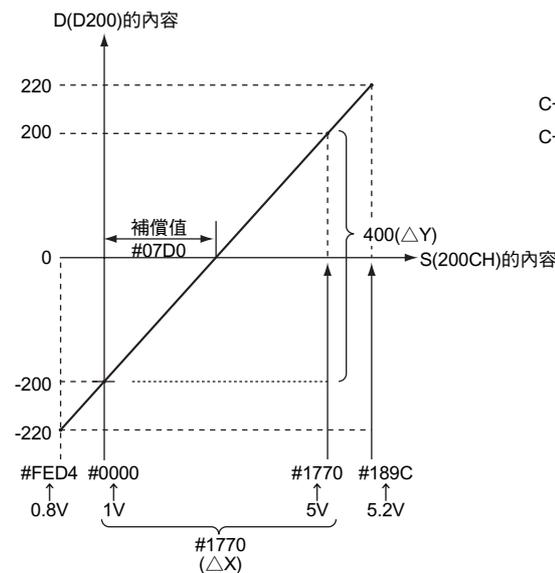
C : D100	0 0 0 0	補償值
C+1 : D101	1 7 7 0	$\Delta X$
C+2 : D102	0 3 0 0	$\Delta Y$

## ● 將類比訊號 1 ~ 5V 轉換為 -200 ~ +200 時

當對應類比訊號 1 ~ 5V 的帶符號 BIN 值也就是 10 進位的 &0 ~ 6000 或是 16 進位的 #0000 ~ 1770 被儲存至 3CH 時，該數值就會被轉換 (Scaling) 為 -0200 ~ 0200 的 BCD 值。SCL3



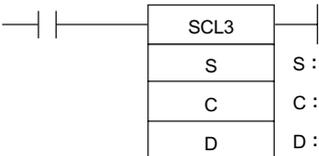
當 0.00 變成 ON 時，就會根據由偏移量 = 16 進位的 #07D0  $\Delta X = 16$  進位的 #1770  $\Delta Y = \text{BCD}\#0400$  所構成的一次函數進行轉換 (Scaling)，將類比輸入模組所輸出的 3CH 值儲存至 D200。



C : D100	0 7 D 0	補償值
C+1 : D101	1 7 7 0	$\Delta X$
C+2 : D102	0 4 0 0	$\Delta Y$

# SCL3

指令名稱	指令記號	指令的各種組合	Fun No.	功能
轉換 (Scaling) 3	SCL3	@ SCL3	487	根據所指定的附偏移量一次函數，將帶符號的 BCD 資料轉換 (Scaling) 為帶符號的 BIN 資料。

符號	SCL3	
		S : 欲轉換的CH編號
	D : 儲存轉換結果用的CH編號	

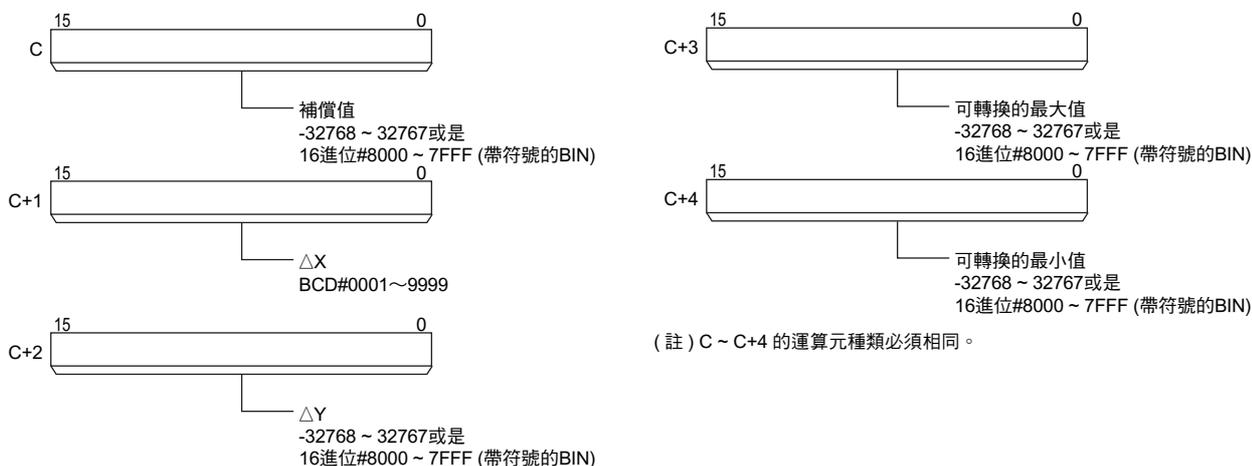
## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	欲轉換的 CH 編號	WORD	1
C	儲存參數的下位 CH 編號	WORD	5
D	儲存轉換結果用的 CH 編號	INT	1

### C : 儲存參數的下位 CH 編號



### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S,C,D	○	○	○	○	○	○	○	○	○	○	-	-	-

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S 的資料不是 BCD 時，本旗標 ON</li> <li>當 $\Delta X (C+1)$ 的資料為 0001 ~ 9999 的 BCD 以外的數值時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>當轉換結果為 0 時，本旗標 ON</li> <li>否則為 OFF</li> </ul>
負旗標	P_N	<ul style="list-style-type: none"> <li>根據轉換結果，當最上位元為 1 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

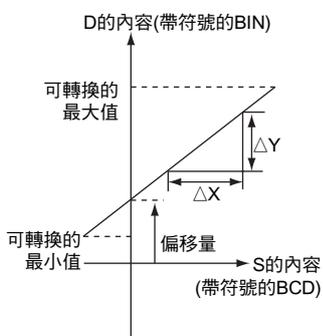
根據由 C 所指定參數所構成的一次函數，將 S 所指定的帶符號 BCD 資料 (BCD 資料為絕對值，利用 CY 旗標進行正負判別，當旗標 ON 時：負數、OFF 時：正數) 轉換為帶符號的 BIN 資料，然後再將結果輸出至 D。請利用 STC 指令 (040)/CLC 指令 (041) 將 CY 旗標設定為 ON/OFF。

轉換公式：
$$D = \frac{\Delta Y}{\Delta X \text{的BIN轉換值}} \times (\text{S的BIN轉換值}) + (\text{偏移量})$$

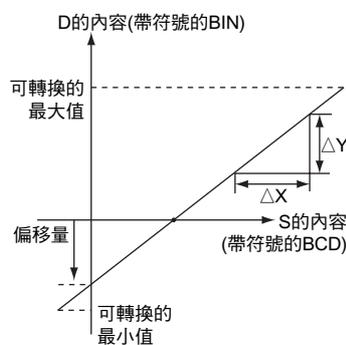
(註)  $\frac{\Delta Y}{\Delta X}$  表示斜率。

- 偏移量有可能是正數、0 或是負數。
- 斜率也同樣可能是正數、0 或是負數，因此，可執行反相轉換。
- S 的 BCD 資料代表絕對值，而且利用執行指令時的進位旗標 (CY) 來判別正負，因此，您所要轉換的資料必須符合 -9999 ~ 9999 的範圍。
- 將轉換後的資料小數點以下全部四捨五入。
- 當轉換結果超出可轉換的最大值 (C+3) 時，就會輸出可轉換的最大值，一旦可轉換的最小值 (C+4) 時，就會輸出可轉換的最小值。

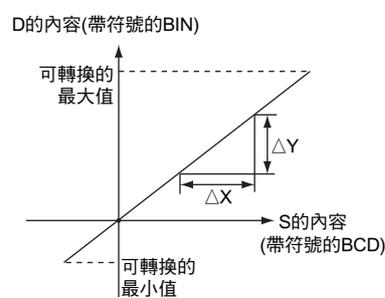
當偏移量為正值時



當偏移量為負值時



當偏移量為0值時

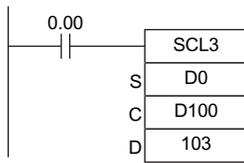


## 提示

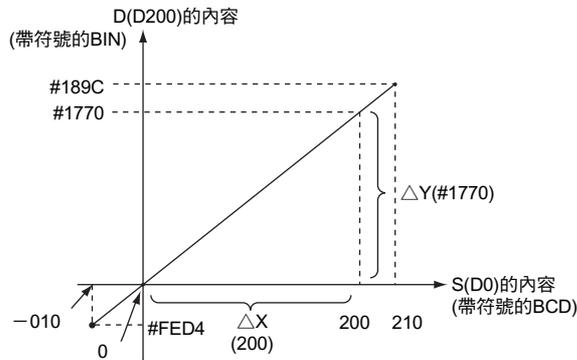
- 當您希望將使用者定義的刻度轉換為類比輸出模組專用的帶符號 BIN 資料時，請使用本指令 (例如：本指令可將 0 ~ 200°C 轉換為 10 進位的 &O ~ 6000 或是 16 進位的 #0000 ~ 1770，然後再將類比輸出訊號 1 ~ 5V 輸出至類比輸出模組)

### 程式例

將 0 ~ 200 的數值轉換為類比訊號 (例如：1 ~ 5V) 時，帶符號的 BCD 值 0 ~ 200 就會被轉換 (Scaling) 為類比輸出模組專用的資料，也就是帶符號的 BIN 值 0 ~ 1770。



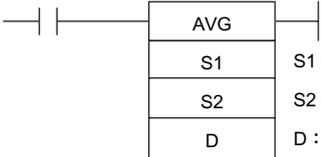
當 0.00 變為 ON 時，就會根據偏移量 = #0000、 $\Delta X = \text{BCD}\#0200$ 、 $\Delta Y = 16$  進位的 #1770 所構成的一次函數來轉換 D0 的數值，然後再將結果儲存至 103CH。



C : D100	0000	偏移量
C+1 : D101	0200	$\Delta X$
C+2 : D102	1770	$\Delta Y$
C+3 : D103	189C	可轉換的最大值
C+4 : D104	FED4	可轉換的最小值

# AVG

指令名稱	指令記號	指令的各種組合	Fun No.	功能
資料平均化	AVG	—	195	由 BIN 計算出 CH 資料的指定週期次數之平均值。

符號	AVG	
		<p>S1：輸入現在值的CH編號(對象CH)</p> <p>S2：計算平均值的週期次數</p> <p>D：儲存平均值的下位CH編號</p>

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

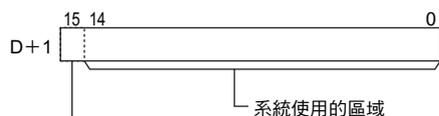
運算元種類	內容	資料型態	容量
S1	輸入現在值的 CH 編號 (對象 CH)	UINT	1
S2	計算平均值的週期次數	UINT	1
D	儲存平均值的下位 CH 編號	UINT	可調整

S2：計算平均值的週期次數

10 進位 &1 ~ 64 或 16 進位 #0001 ~ 0040

D：儲存平均值的下位 CH 編號

D+1：作業資料 (禁止讀寫資料)



啟動平均值的旗標

0：關閉(當AVG指令開始執行後的掃描次數 < S2值時)

1：開啟(當AVG指令開始執行後的掃描次數 ≥ S2值時)

D+2：過去值1

D+(S的N值)+1：過去值N

(註) D ~ D+(S的N值)+1 的運算元種類必須相同。

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S1,S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D	○	○	○	○	○	○	○	○	○	○	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P.ER	<ul style="list-style-type: none"> <li>當 S2 的資料為 0 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

依指定的週期次數 (S2) 及儲存指標 (Pointer) (D+1 的位元 00 ~ 07) 更新 S1 所指定的不帶符號 BIN 資料 (10 進位的 &0 ~ 65535 或是 16 進位的 #0000 ~ FFFF)，同時，將該資料當作過去值，依序儲存在 D+2 以後的位置。

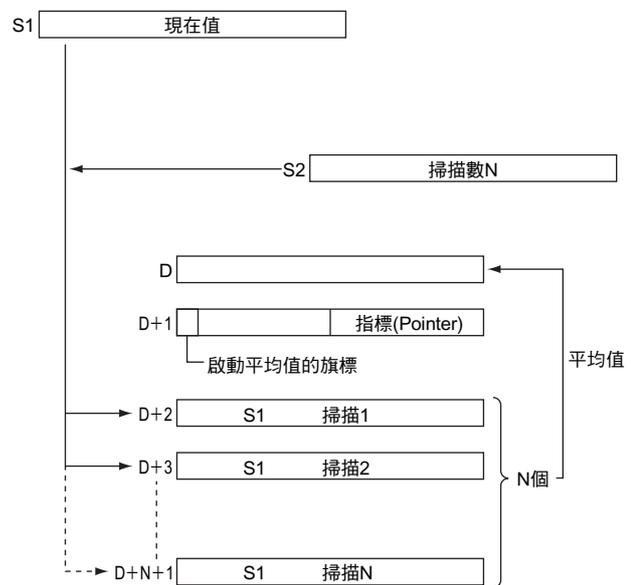
此時，S1 的資料將直接被輸出至 D，並且將啟動平均值的旗標 (D+1 的位元 15) 設定為 0 (OFF)。當所指定的週期次數的過去值 S1 被儲存在 D+2 以後的位置時，本指令就會開始演算過去值的平均值，然後再將結果以不帶符號的 BIN 資料的方式輸出至 D。此時，啟動平均值的旗標 (D+1 的位元 15) 將被設定為 1 (ON)。

接下來的每次掃描將根據掃描 S2 的最新資料來計算平均值，並且將結果輸出至 D。

可指定的週期次數 (S2) 最大為 64。

一旦所指定的次數超過 64 時，就會以 64 來執行動作當過去值的儲存指標到達 S2-1 時，就會重新由 0 開始儲存。

平均值小數點以後的資料皆會被四捨五入。

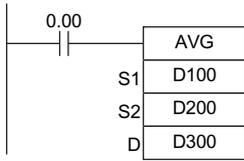


## 使用上的注意事項

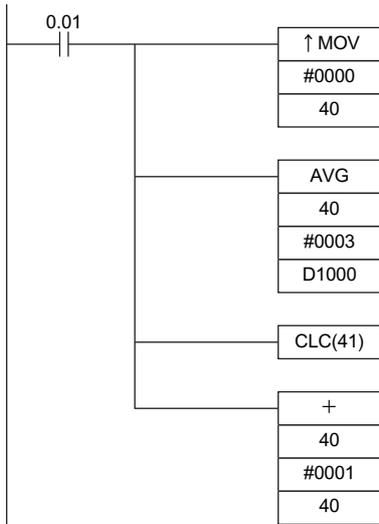
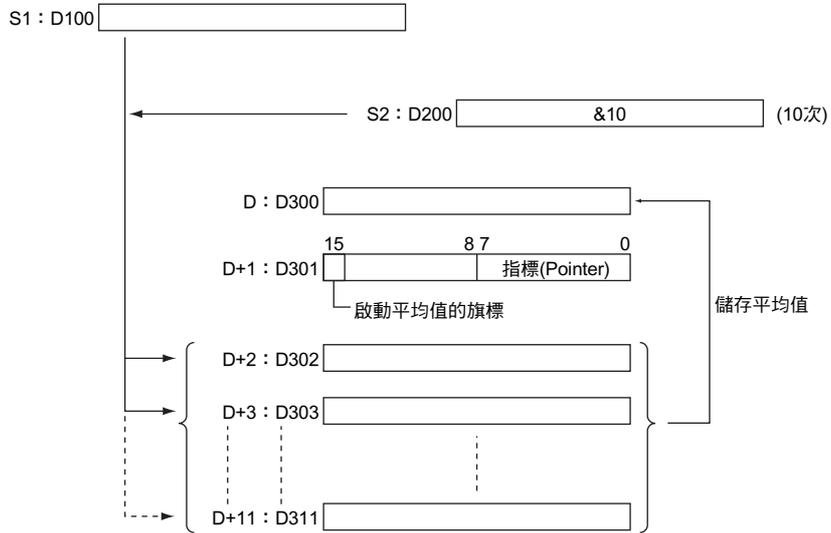
當第一次輸入條件上升時，就會利用本指令將作業資料 (D+1) 初始化 (清除為 0)。

不過，一旦開始運轉第 1 週期時，就無法在指令執行時，利用本指令將作業資料 (D+1) 初始化了。因此，如果希望從開始運轉的第 1 週期起就執行本指令時，請利用程式來清除 D+1。

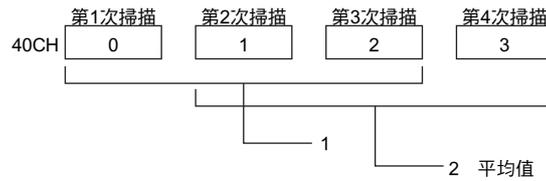
程式例



當 0.00 變成 ON 時，就會依掃描過的 D200 內容 (例如：&10 也就是 10 次)，將 D100 的內容依序儲存在 D302 ~ D311 的 10CH 中，此時，10CH 的平均值會被儲存在 D300，而 D301 的位元 15 會被設定為 ON。



- 將輸入 0.01 設定為 ON 時，每掃描一次，就會由 0 開始將 40CH 的內容依序加 1。
- 第 1、2 次掃描時，AVG 指令會將 40CH 的內容儲存在 D1002、D1003，同時也會更新 D1001。(為 40CH 的內容執行加法運算，目的僅在於確認 AVG 指令的執行結果有何變化，並無特別的意義)。
- 從第 3 次掃描開始，AVG 指令將計算最新的平均值，並且將結果儲存至 D1000。

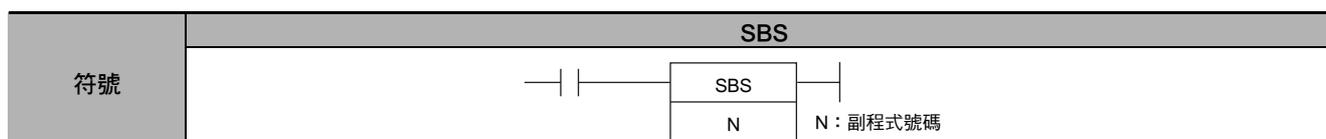


D1000	0	1	1	2	平均值 作業資料 過去讀取資料 的儲存區域
D1001	1	2	8000	8001	
D1002	0	0	0	3	
D1003	-	1	1	1	
D1004	-	-	2	2	

# 副程式的指令

## SBS

指令名稱	指令記號	指令的各種組合	Fun No.	功能
呼叫副程式 (Subroutine call)	SBS	@SBS	091	執行本程式，即可呼叫指定編號的副程式。



### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 運算元種類的說明

運算元種類	內容	資料型態	容量
N	副程式號碼	—	1

N：副程式號碼

10 進位 0 ~ 127

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
N	—	—	—	—	—	—	—	—	—	○	—	—	—

### 組合使用指令

SBN (Subroutine entry：副程式進入) 指令及 RET (Subroutine return：副程式返回)

### 條件旗標的動作

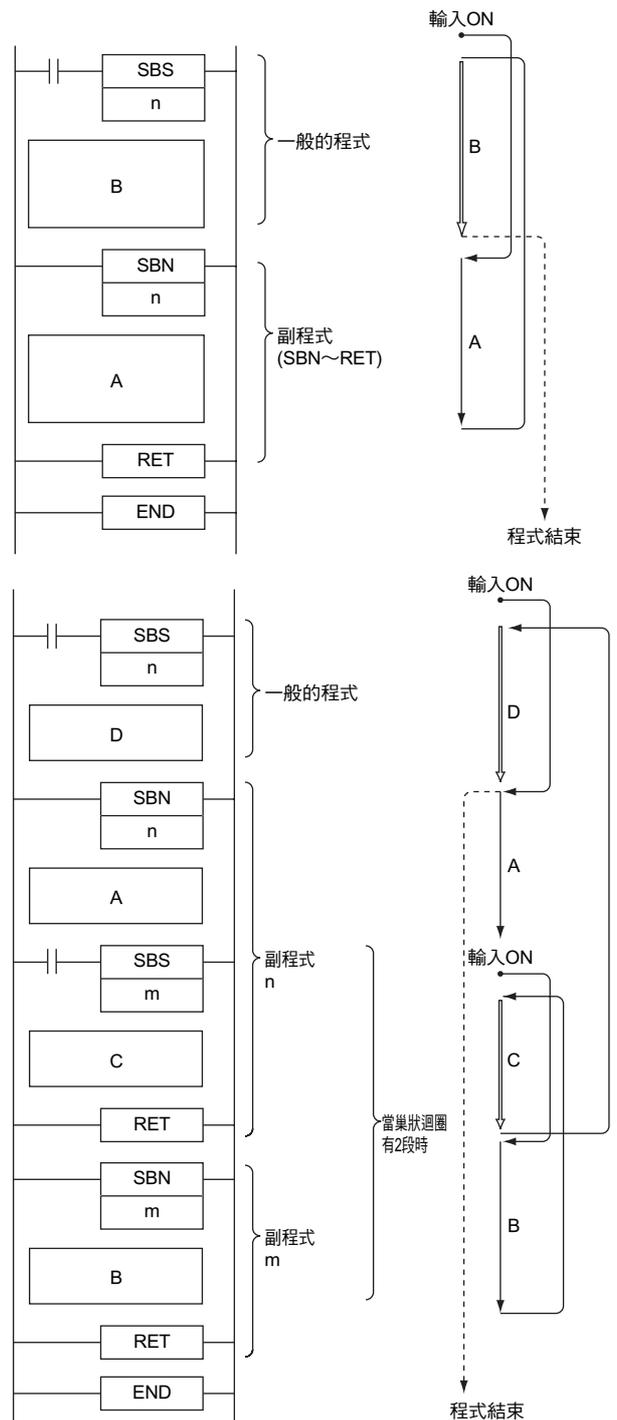
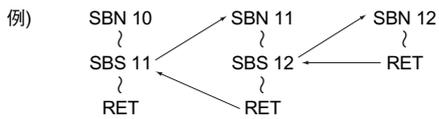
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>• 巢狀結構數超過 16 段時，本旗標就會 ON</li> <li>• 當您所指定的副程式不存在時，本旗標就會 ON</li> <li>• 由副程式中呼叫同一個副程式時，本旗標就會 ON</li> <li>• 呼叫執行中的副程式時，本旗標就會 ON</li> <li>• 您所指定的副程式並未定義在本任務中時，本旗標就會 ON</li> <li>• 否則為 OFF</li> </ul>

## 功能

呼叫 N 所指定編號的副程式中的 (SBN 指令 ~ RET 指令之間的區域) 程式，當副程式執行完成後，就會回到本指令的下一項指令，您可多次呼叫同一個副程式。

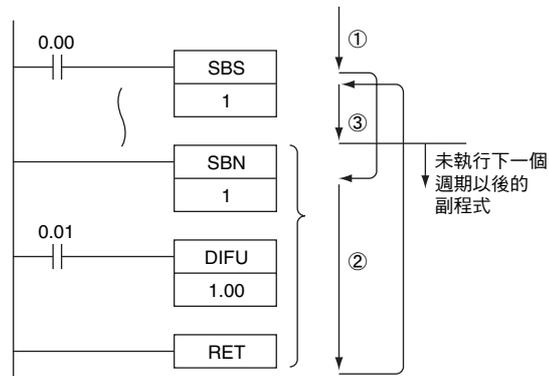
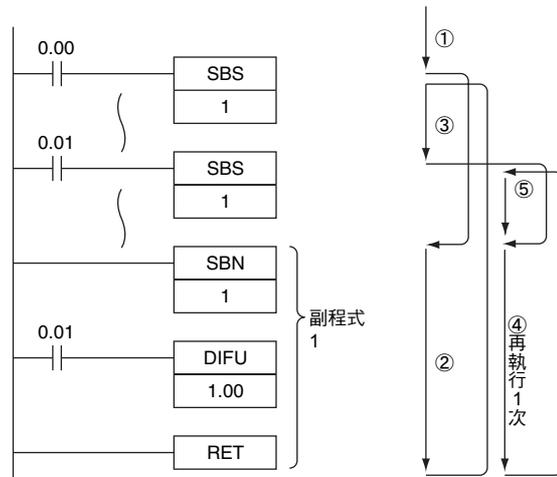
副程式的巢狀結構最多 16 段。

所謂「巢狀結構」指的就是副程式 (SBN ~ RET) 當中還包含了一個副程式呼叫的狀態。



## 使用上的注意事項

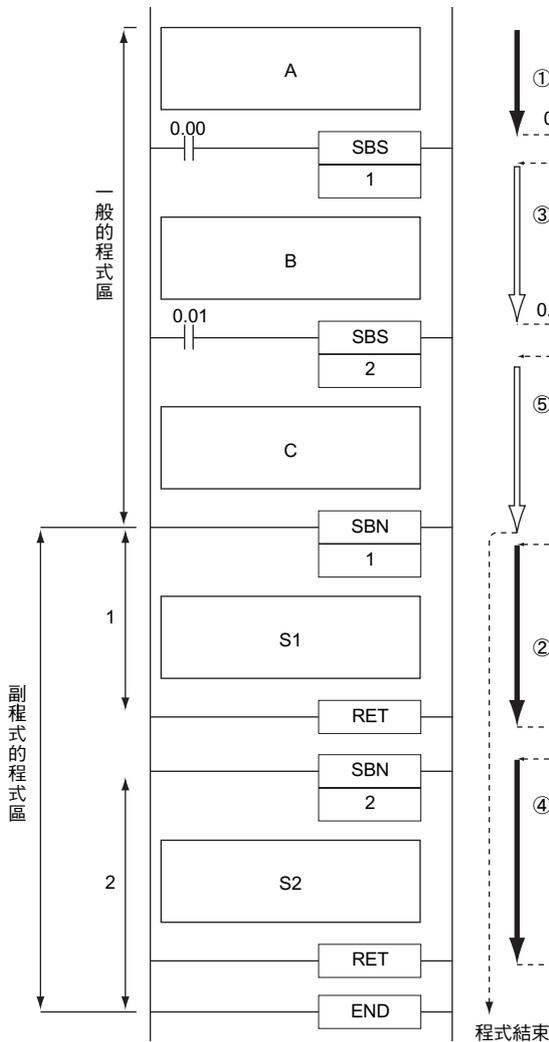
- 副程式編號就是針對某一個副程式進行配置的唯一編號，因此請勿重複使用該編號。
- 如果要在副程式中使用微分型指令 (UP、DOWN、DIFU、DIFD、指令的各種組合 @、附 % 指令) 時，需注意下列事項。
  - 在同一個週期中多次執行相同的副程式時，將造成副程式中的輸入微分型指令 (DIFU、DIFD、指令的各種組合 @、附 % 指令) 動作出現不穩定的情形，此點需特別注意。
  - 下列範例就是利用將輸入繼電器 0.00 設定為 ON 的方式來執行副程式 1，當輸入繼電器由 OFF → ON 時，就會利用 DIFU 指令，讓 1.00 變成 ON。不過，當同一個週期中的輸入繼電器 0.01 被設定為 ON 時，就會再次執行副程式 1，因此這時候 DIFU 指令不會檢測出 0.01 訊號上升，反而會讓 1.00 變為 OFF。
  - 反之，如果先執行副程式中的微分指令 (DIFU、DIFD 指令)，而且在輸出 ON 的狀態下，並未由下次開始呼叫相同的副程式時，微分指令 (DIFU、DIFD 指令) 的輸出將持續 ON，而且不會變為 OFF，此點請特別注意。
  - 下列範例就是利用將 0.00 變成 ON 的方式，以執行副程式 1，一旦 0.01 由 OFF → ON 時，就會利用 DIFU 指令將 1.00 設定為 ON。如果接下來的週期，0.00 保持 OFF 狀態且未執行副程式 1 時，DIFU 指令的輸出 1.00 也將保持 ON 狀態。
  - 以 IL-ILC 指令來執行互鎖時，SBS 指令將執行 NOP 處理動作。



程式例

● 出現 2 個副程式呼叫指令時

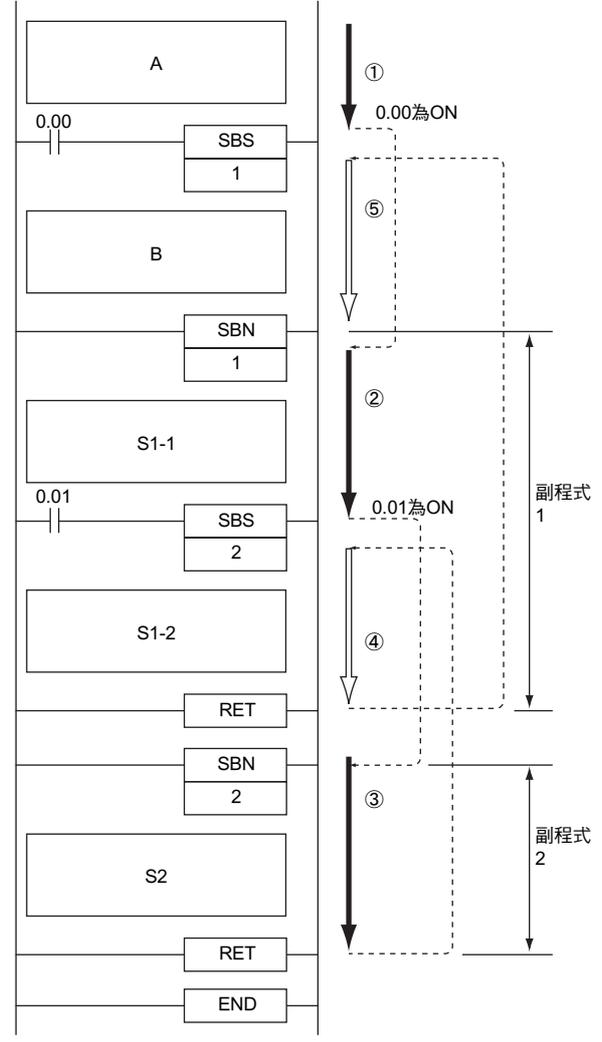
• 未設定巢狀結構時



當0.00變成ON時，一旦副程式編號1的副程式執行完成後，就會回到SBS1指令的下一項指令。  
當0.01變成ON時，一旦副程式編號2的副程式執行完成後，就會回到SBS2指令的下一項指令。

0.00	0.01	執行順序
ON	ON	A→S1→B→S2→C
ON	OFF	A→S1→B→C
OFF	ON	A→B→S2→C
OFF	OFF	A→B→C

• 巢狀結構 2 段時



當0.00變成ON時，就會執行副程式編號1的副程式，如果0.01變成ON時，則會在執行副程式編號2的副程式後，回到SBS2的下一項指令，繼續執行到SBN2的前一項指令結束，接著就會回到SBS1的下一項指令。此時，將持續執行SBN1的前一項指令，並且結束程式。

0.00	0.01	執行順序
ON	ON	A→S1-1→S2→S1-2→B
ON	OFF	A→S1-1→S1-2→B
OFF	ON	A→B
OFF	OFF	A→B

# SBN/RET

指令名稱	指令記號	指令的各種組合	Fun No.	功能
副程式進入	SBN	—	092	表示指定編號的副程式起始位置。
副程式返回	RET	—	093	表示副程式結束。

符號	SBN	RET

## 是否適用於特定區域

### ● SBN

區域	工程步進程式	副程式	中斷任務程式
使用	—	—	○

### ● RET

區域	工程步進程式	副程式	中斷任務程式
使用	—	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
		SBN	
N	副程式編號	—	1

### ● SBN

N：副程式編號

10 進位 0 ~ 127

### ● 運算元種類

內容		CH 位址						間接 DM		常數	條件旗標	時鐘脈衝	TR	
		CIO	WR	HR	AR	T	C	DM	@DM					* DM
SBS	N	—	—	—	—	—	—	—	—	—	○	—	—	—

## 組合使用指令

SBS ( 呼叫副程式 ) 指令

## 條件旗標的動作

### ● SBS/ RET

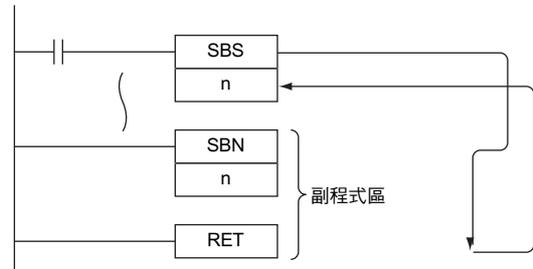
無

## 功能

### ● SBN

表示副程式的開頭。

第一個 SBN 指令以後的區域就稱之為副程式，除了 SBS 指令外，其他指令皆無法執行副程式的任何動作。

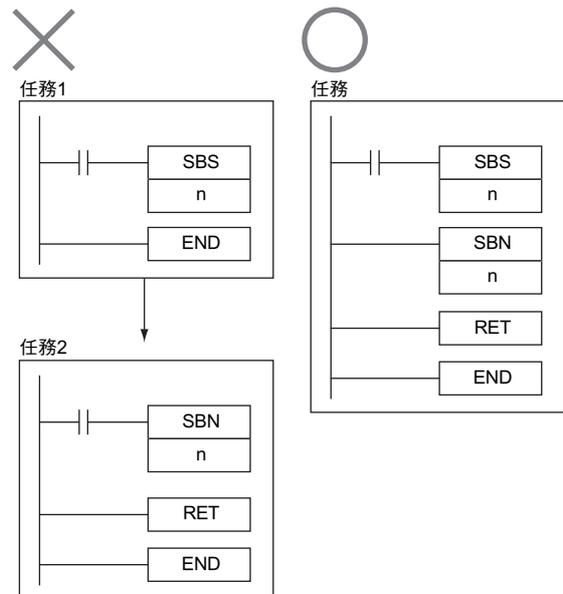


### ● RET

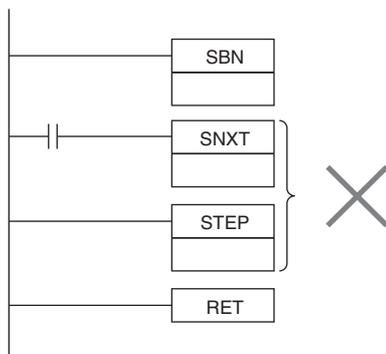
當副程式執行完成後，就會回到呼叫來源也就是 SBS 指令的下一個指令。

## 使用上的注意事項

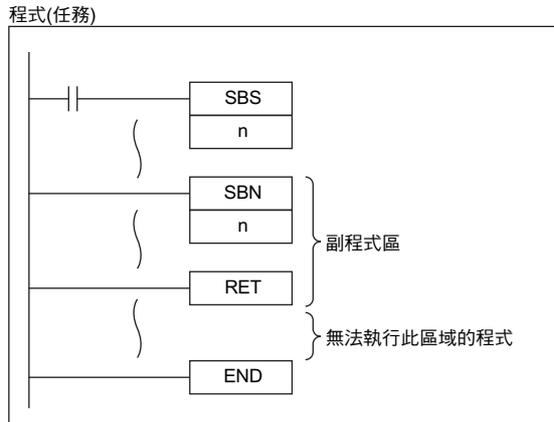
- 請將副程式(SBN ~ RET)中編號相同的SBS指令配置在同一個程式(任務)中；否則，將無法執行副程式。



- 請注意：工程步進指令 (STEP 指令、SNXT 指令) 無法在副程式中使用。



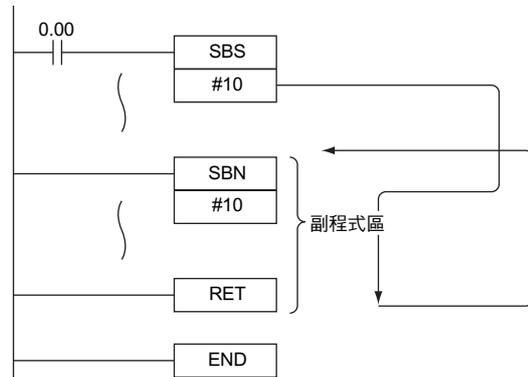
- 請將副程式 (SBN ~ RET) 配置於任務中所配置的程式結尾 (END 指令) 的前面或是一般程式的後面。一般程式如果被配置在副程式 (SBN ~ RET) 的後面時，將使得一般程式無法執行，並且發生程式無效的情形。



- (註)  
利用週邊程式來輸入 N 資料的方法如下：
- 使用 CX-Programmer 時：輸入 0 ~ 127

### 程式例

當 0.00 變成 ON 時，執行完副程式編號 10 的副程式 (SBN ~ RET) 後，就會回到 SBS 的下一個指令。



# 中斷控制指令

CP1E CPU 模組具有下列中斷功能。

中斷種類	中斷條件	設定方法
I/O 中斷	當 CPU 輸入的中斷輸入訊號上升 / 下降時	利用 MSKS 指令，指定要設定為啟動中斷的中斷輸入。
定時中斷	定時 (時間間隔固定)	利用 MSKS 指令來指定定時中斷時間。

## 指令概述

### ● 設定中斷遮罩：MSKS 指令

當 I/O 中斷任務及定時中斷任務執行時，就會開始執行初始狀態的遮罩 (禁止) 動作，只要執行 MSKS 指令，即可進行同意 / 禁止接受 I/O 中斷任務的控制，或是設定定時中斷任務的定時中斷時間。

### ● 中斷解除：CLI 指令

執行解除 / 保持 I/O 中斷輸入記憶的指定，指定定時中斷的第一次開始中斷時間，或是指定為解除 / 保持高速計數器中斷的記憶。

### ● 禁止執行中斷任務：DI 指令

禁止執行所有的中斷任務。

### ● 禁止執行中斷任務的解除：EI 指令

解除禁止執行所有的中斷任務。

## 使用中斷任務時的注意事項

### ● 所有中斷任務的共通事項

- 同時發生多個中斷要因時，需符合下列要求。

I/O 中斷 > 定時中斷

(註) A > B 表示 A 的優先順序高於 B，而且在相同的中斷等級下，以任務編號較小者為優先。

### ● I/O 中斷

- I/O 中斷任務的輸入對象為 CPU1E CPU 模組內建輸入所傳送的中斷輸入。
- 請使用 CPU 模組的 0ch02bit ~ 0ch07bi 作為中斷輸入，若使用其他的輸入，將造成 I/O 中斷任務無法啟動。
- 解除中斷任務時，解除前所檢測到的中斷輸入將會全部被清除
- 本指令對於 I/O 中斷輸入的記憶數並無特別限制，不過，不得重複記憶相同編號的 I/O 中斷輸入，否則之後的中斷將會被忽略。而且，I/O 中斷輸入將持續記憶直到執行結束為止，因此即使執行時再次出現相同編號的中斷輸入，仍然會被忽略。

### ● 定時中斷

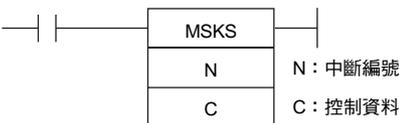
- 定時中斷任務的執行時間必須短於定時中斷的設定時間。
- 定時中斷的時間單位固定為 0.1 ms。

## 相關特殊輔助繼電器

名稱	標籤	內容
中斷任務最大值處理時間	A440	中斷任務的最大處理時間會被儲存為單位 0.1 ms 的 BIN 資料，開始運轉時即被清除。
最大處理時間中斷任務編號	A441	最大處理時間的中斷任務編號會被儲存為 BIN 資料格式，任務 0 ~ 15 (16 進位的 0 ~ FF) 所對應的格式為 #8000 ~ 800F。 開始運轉後一旦發生第一次中斷，A441GH 的位元 15 就會變成 1 (ON)。接著，在所發生的中斷任務中的最大處理時間將被儲存為 16 進位的下位 2 位數格式。開始運轉時即被清除。

# MSKS

指令名稱	指令記號	指令的各種組合	Fun No.	功能
設定中斷遮罩	MSKS	@ MSKS	690	執行 I/O 中斷處理或是定時中斷處理的指定。

符號	MSKS	
		N : 中斷編號

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
N	中斷編號	—	1
C	控制資料	UINT	1

### (1) 發生 I/O 中斷任務時

資料	資料內容	
N	設定禁止 / 同意輸入中斷	指定檢測 I/O 中斷任務的 Edge
	I/O 中斷編號	I/O 中斷編號
	102 : 輸入中斷 2 (中斷任務 No. 2) 103 : 輸入中斷 3 (中斷任務 No. 3) 104 : 輸入中斷 4 (中斷任務 No. 4) 105 : 輸入中斷 5 (中斷任務 No. 5) 106 : 輸入中斷 6 (中斷任務 No. 6) 107 : 輸入中斷 7 (中斷任務 No. 7)	112 : 輸入中斷 2 (中斷任務 No. 2) 113 : 輸入中斷 3 (中斷任務 No. 3) 114 : 輸入中斷 4 (中斷任務 No. 4) 115 : 輸入中斷 5 (中斷任務 No. 5) 116 : 輸入中斷 6 (中斷任務 No. 6) 117 : 輸入中斷 7 (中斷任務 No. 7)
C	中斷遮罩資料	輸入中斷 Edge 資料
	10 進位 &0 (16 進位 #0000) : 同意中斷 (直接模式) 10 進位 &1 (16 進位 #0001) : 禁止中斷	10 進位 &0 (16 進位 #0000) : 檢測輸入訊號上升 10 進位 &1 (16 進位 #0001) : 檢測輸入訊號下降

(註) 1 變更 Edge 檢測時，變更前所檢測到的中斷輸入將會全部被清除。

2 CP1E10 點輸出輸入型無法使用輸入中斷 6、7。

### (2) 利用指定復歸啟動來設定禁止 / 同意定時中斷任務的計時器中斷時

資料	資料內容	
N	定時中斷編號	
	4 或 14 : 定時中斷 0 (中斷任務 No. 1)	
C	定時中斷單位時間 (PLC 系統設定)	定時中斷設定時間
	全設定共通	10 進位 &0 (16 進位 #0000) : 禁止中斷 (停止內部計時器)
	0.1ms	10 進位 &10 ~ 9999 (16 進位 #000A ~ 270F) : 同意中斷 (復歸內部計時器的數值後，就會以 1.0 ~ 999.9 ms 的中斷間隔時間啟動內部計時器) (註) 禁止使用 10 進位的 &1 ~ 10 (16 進位的 #0001 ~ 000A)，否則就會發生指令錯誤。

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
N	—	—	—	—	—	—	—	—	—	○	—	—	—
C	○	○	○	○	○	○	○	○	○	○	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 N 的資料超出指定範圍時，本旗標就會 ON</li> <li>I/O 中斷指定：當 C 的內容超出指定範圍時，本旗標就會 ON</li> <li>定時中斷指定：以 0.1 ms 為單位時，只要資料超出 10 進位的 &amp;10 ~ 9999 (16 進 #000A ~ 270F) 的範圍，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

依 N 的值指定要針對哪個中斷任務進行何種設定。

### (1) I/O 中斷的輸入中斷 (N = 102 ~ 107)

- 利用 C 的位元，將 N 所指定的輸入中斷設定為同意 / 禁止中斷。如此一來，就能夠控制每個 I/O 中斷任務是否能夠執行。
- 當中斷輸入被指定為同意中斷時，指定前所檢測到的中斷要因將全部被清除。

### (2) I/O 中斷的 Edge 檢測 (N = 112 ~ 117)

- 利用 C 的位元，將 N 所指定的輸入中斷指定為 Edge 檢測 ( 檢測輸入中斷訊號上升 / 下降的其中任一種 ) 。
- 指定 Edge 檢測可以和同意 / 禁止輸入中斷的處理搭配使用。又，若未執行 MSKS 指令來指定訊號上升 / 下降時，根據初始設定，系統將自動進行檢測上升訊號的動作。
- 指定為檢測 Edge 時，指定前所檢測出來的中斷要因將全部被清除。

### (3) 復歸並啟動定時中斷任務 (N = 4 或 14)

- 將 N 所指定的定時中斷任務的中斷間隔設定為 C 所指定的時間，同時還會復歸並啟動內部計時器的現在值。如此，便能夠保證執行本指令後的開始中斷時間。

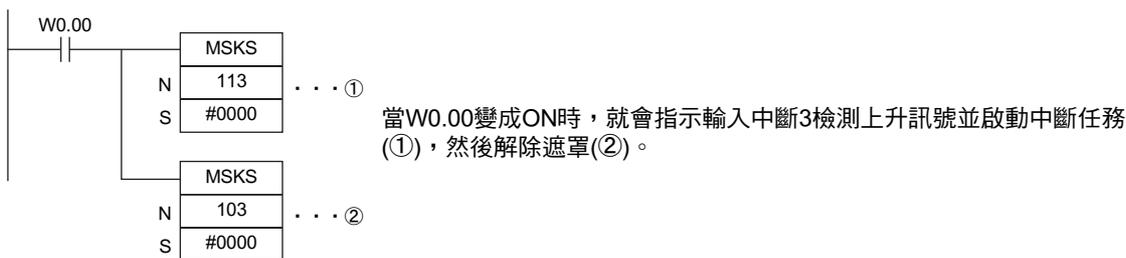
## 使用上的注意事項

- 定時中斷任務的執行時間必須短於定時中斷的設定時間。
- 執行定時中斷時，無法同時變更訂時中斷設定時間。請先利用 MSKS 指令設定為禁止中斷 ( 停止內部計時器 )，然後再變更定時中斷的設定時間。

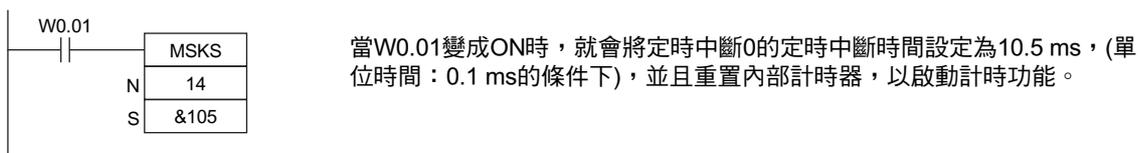
## 程式例

### 動作說明

- 指定為輸入中斷時 ( 直接模式 )



- 指定為定時中斷時 ( 指定為復歸啟動 )



# CLI

指令名稱	指令記號	指令的各種組合	Fun No.	功能
中斷解除	CLI	@CLI	691	指定中斷輸入記憶的解除 / 保持，或是指定定時中斷的第一次開始配置的時間。

符號	CLI	

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
N	中斷編號	—	1
C	控制資料	UINT	1

### (1) 指定為解除 / 保持 I/O 中斷任務的輸入中斷要因記憶時

資料	資料內容
N	I/O 中斷編號
	102：輸入中斷 2 (中斷任務 No. 2)
	103：輸入中斷 3 (中斷任務 No. 3)
	104：輸入中斷 4 (中斷任務 No. 4)
	105：輸入中斷 5 (中斷任務 No. 5)
	106：輸入中斷 6 (中斷任務 No. 6)
C	要因記憶資料
	10 進位 &0 (16 進位 #0000)：解除記憶
	10 進位 &1 (16 進位 #0001)：保持記憶

(註) CP1E 10 點輸出輸入型無法使用輸入中斷 6、7。

### (2) 指定定時中斷任務第一次中斷的開始時間時

資料	資料內容
N	定時中斷編號
	4：定時中斷 0 (中斷任務 No. 1)
C	定時中斷單位時間 (PLC 系統設定)
	0.1ms

## (3) 指定為解除 / 保持高速計數器中斷要因的記憶時合

資料	資料內容
N	高速計數器的中斷編號 10：高速計數器輸入 0 11：高速計數器輸入 1 12：高速計數器輸入 2 13：高速計數器輸入 3 14：高速計數器輸入 4 15：高速計數器輸入 5
C	要因記憶資料 10 進位 &0 (16 進位 #0000)：解除記憶 10 進位 &1 (16 進位 #0001)：保持記憶

(註) CP1E 10 點輸入型無法使用高速計數器輸入 5。

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
N	—	—	—	—	—	—	—	—	—	○	—	—	—
C	○	○	○	○	○	○	○	○	○	○	—	—	—

## 條件旗標的動作

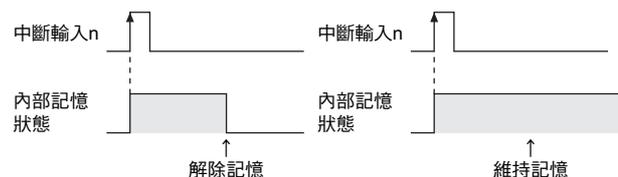
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>N 的資料不在指定範圍內時，ON</li> <li>當 C 的資料超出 0 ~ 1 的範圍時，本旗標就會 ON (指定為 I/O 中斷或是高速計數器中斷時)</li> <li>當 C 的資料超出 10 進位的 &amp;10 ~ 9999 或 16 進位的 #000A ~ 270F 的範圍時，本旗標就會 ON (指定為定時中斷時)</li> <li>否則為 OFF</li> </ul>

## 功能

本指令將依 N 的數值不同，執行解除 / 保持輸入中斷要因的記憶、設定定時中斷開始第一次中斷的時間，或是指定解除 / 保持高速計數器中斷要因的記憶等。

## (1) 輸入中斷 (N = 102 ~ 107)

利用 C 的位元，為 N 所指定的輸入中斷解除 / 保持輸入中斷要因的記憶。

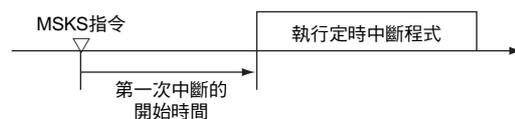


執行中斷任務時，一旦系統接收到其他 I/O 中斷任務的中斷輸入，就會先將輸入中斷編號記憶在系統內部 (條件為不執行 CLI (解除中斷) 指令)，執行時，將由已經被記憶的輸入中斷編號的 I/O 中斷任務中，由編號較小者開始依序執行。

因此，若希望在中斷任務執行時，將已經記憶的輸入中斷設定為無效時，請利用 CLI (解除中斷) 指令來解除已經記憶在內部的輸入中斷編號。

## (2) 定時中斷 (N = 4)

針對 N 所指定的定時中斷任務，將 C 的值指定為開始第一次中斷的時間。

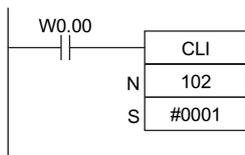


## (3) 高速計數器中斷 (N = 10 ~ 15)

針對 N 所指定的內建高速計數器中斷 (比較目標值一致性)，解除其中斷要因的記憶或是保持記憶。

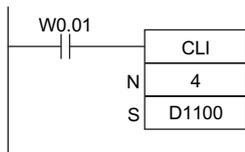
### 程式例

- 指定為輸入中斷時

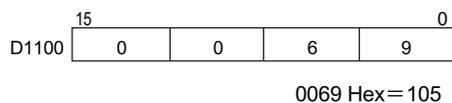


當W0.00變成ON時，就會解除輸入中斷2的中斷要因記憶。

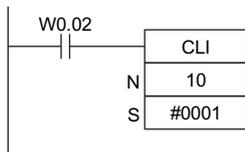
- 指定為定時中斷時



當W0.01變成ON時，就會將定時中斷0的第一次開始中斷時間設定為10.5 ms。



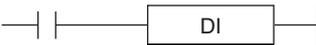
- 指定為高速計數器中斷時



當W0.02變成ON時，就會解除高速計數器中斷0的中斷要因記憶。

# DI

指令名稱	指令記號	指令的各種組合	Fun No.	功能
禁止執行中斷任務	DI	@DI	693	禁止執行所有的中斷任務 (斷電中斷任務除外)。

符號	DI
	

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>執行中斷任務時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

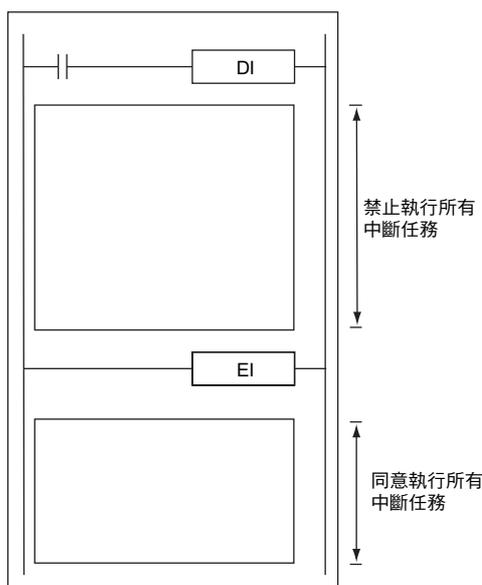
## 功能

適用於執行週期任務的期間，執行本指令可禁止所有中斷任務 (I/O 中斷任務、定時中斷任務) 的執行。若希望在禁止執行中斷任務的解除指令 (EI) 執行前，暫停中斷任務的執行，請使用本指令。

## 使用上的注意事項

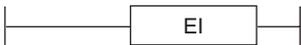
- 進入禁止狀態時，請利用 EI (禁止執行中斷任務的解除) 指令來解除禁止狀態。
- 本指令無法在中斷任務中執行。

## 程式例



# EI

指令名稱	指令記號	指令的各種組合	Fun No.	功能
禁止執行中斷任務的解除	EI	—	694	解除所有中斷任務 ( 斷電中斷任務除外 ) 的禁止執行。

符號	EI
	

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>執行中斷任務時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

- 適用於執行週期任務的期間，可解除 DI ( 禁止執行中斷任務 ) 指令對於所有中斷任務 ( I/O 中斷任務、定時中斷任務 ) 的禁止執行。

## 使用上的注意事項

- 本指令不需要輸入條件 ( 電源流程 )。
- 本指令係利用 DI ( 禁止執行中斷任務的解除 ) 指令來解除禁止狀態，對於那些未利用 MSKS ( 中斷遮罩設定 ) 指令來設定執行 I/O 中斷 ( 可接受 ) 或執行定時中斷處理的中斷任務編號，將無法利用本指令來執行任何處理。
- 本指令無法在中斷任務中執行。

# 高速計數器 / 脈衝輸出指令

## INI

指令名稱	指令記號	指令的各種組合	Fun No.	功能
動作模式控制	INI	@ INI	880	執行開始 / 停止和比較表進行比較、變更高速計數器的現在值、變更中斷輸入 (計數器模式) 的現在值、變更脈衝輸出的現在值 (以 0 確定原點)、或是停止脈衝輸出等其中的一項動作。

符號	INI	
		<p>C1: 指定連接埠</p> <p>C2: 控制資料</p> <p>S: 儲存變更資料的下位CH編號</p>

### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 運算元種類的說明

運算元種類	內容	資料型態	容量
C1	指定連接埠	WORD	1
C2	控制資料	UINT	1
S	儲存變更資料的下位 CH 編號	DWORD	2

#### C1：指定連接埠

指定連接埠 (16 進位)	
#0000	脈衝輸出 0
#0001	脈衝輸出 1
#0010	高速計數器輸入 0
#0011	高速計數器輸入 1
#0012	高速計數器輸入 2
#0013	高速計數器輸入 3
#0014	高速計數器輸入 4
#0015	高速計數器輸入 5
#1000	PWM 輸出 0

(註) CP1E 10 點輸入型無法使用高速計數器輸入 5。

#### C2：控制資料

控制資料 (16 進位)	
#0000	開始比較
#0001	停止比較
#0002	變更現在值
#0003	停止脈衝輸出

S：儲存變更資料的下位 CH 編號

指定為變更現在值 (C2 = #0002) 時，就會儲存變更資料。

指定為變更現在值以外的選項時，將不會使用此運算元種類。



發生脈衝輸出時  
-2147.483648 ~ 2147.483647或16進位#80000000 ~ 7FFFFFFF  
發生高速計數器輸入時  
-2147.483648 ~ 2147.483647或16進位#80000000 ~ 7FFFFFFF  
10進&0 ~ 4 294.96729516進#00000000 ~ FFFFFFFF

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
C1,C2	-	-	-	-	-	-	-	-	-	○	-	-	-
S	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>超出 C1、C2、S 所指定的範圍時，本旗標就會 ON</li> <li>當 C1 和 C2 不支援時，本旗標就會 ON</li> <li>未登錄比較表即指定為開始比較時，本旗標就會 ON</li> <li>正在執行脈衝輸出的連接埠被指定為變更現在值時，本旗標就會 ON</li> <li>未被設定為高速計數器的連接埠被指定為變更高速計數器現在值時，本旗標就會 ON</li> <li>CTBL 指令執行時發生中斷，或是在中斷任務中，執行高速計數器輸入指定的 INI 指令時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

功能

針對 C1 所指定的連接埠進行 C2 所指定的控制。

可指定的 C1、C2 組合如下：

C1 (指定連接埠)	C2 (控制資料)			
	開始比較 (#0000)	停止比較 (#0001)	變更現在值 (#0002)	停止脈衝輸出 (#0003)
脈衝輸出 (#0000、#0001)	×	×	○	○
高速計數器輸入 (#0010 ~ #0015)	○	○	○	×
PWM 輸出 (#1000)	×	×	×	○

● 開始比較 (C2 = #0000)

利用比較表登錄 (CTBL) 指令，開始比較已登錄的比較表和高速計數器現在值。

(註) 如果未登錄比較表時，本指令就會出現錯誤，並且無法執行動作。

● 停止比較 (C2 = #0001)

利用比較表登錄 (CTBL) 指令，停止比較已登錄的比較表和高速計數器現在值。

## ● 變更現在值 (C2 = #0002)

控制對象		控制內容	可變更的範圍
脈衝輸出 (C1 = #0000、#0001)		變更脈衝輸出的現在值。為 S+1、S 設定所要變更的數值。 (註) 本指令只能在脈衝輸出停止時執行，一旦在脈衝輸出時執行，就會發生錯誤。	-2147483648 ~ 2147483647 或 16 進位 #80000000 ~ 7FFFFFFF
高速計數器輸入 (C1 = #0010 ~ #0015)	線性模式 下	變更高速計數器的現在值。為 S+1、S 設定所要變更的數值。 (註) 當所指定的連接埠並未被設定為高速計數器時，本指令就會發生錯誤。	-2147483648 ~ 2147483647 或 16 進位 #80000000 ~ 7FFFFFFF
	相位差輸入 / 加法減法 運算脈衝輸入 / 脈衝 + 方向輸入 加法運算脈衝輸入		10 進位 & 0 ~ 4294967295 或 16 進位 #00000000 ~ FFFFFFFF
連結模式下			10 進位 & 0 ~ 4294967295 或 16 進位 #00000000 ~ FFFFFFFF

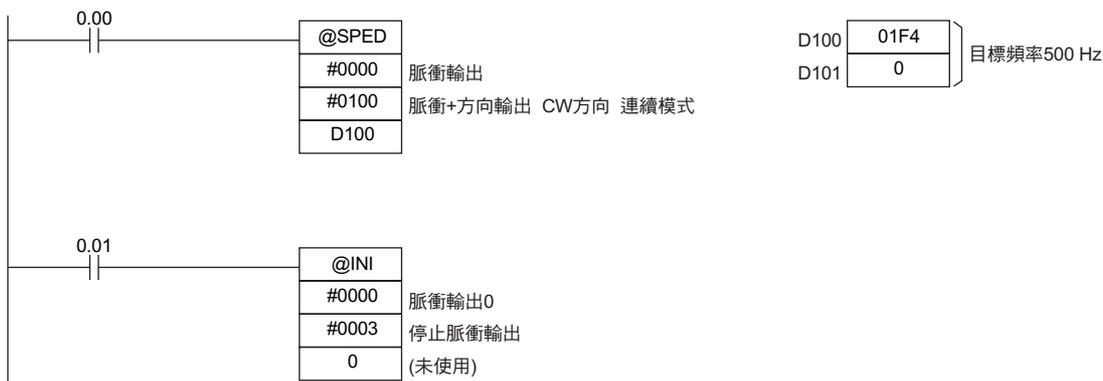
## ● 停止脈衝輸出 (C1 = #0000、#0001、#1000、C2 = #0003)

停止所指定的連接埠的脈衝輸出。(立刻停止)

又，如果在脈衝輸出停止時執行本指令，脈衝量設定就會被清除。

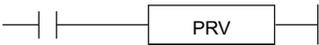
## 程式例

當 0.00 由 OFF → ON 時，就會利用 SPED 指令，在連續模式下由脈衝輸出 0 開始輸出 500Hz 的脈衝。當 0.01 由 OFF → ON 時，就會利用 INI 指令來停止脈衝輸出。



# PRV

指令名稱	指令記號	指令的各種組合	Fun No.	功能
讀取脈衝現在值	PRV	@PRV	881	讀取高速計數器的現在值，脈衝輸出的現在值、中斷輸入 (計數器模式) 的現在值。

符號	PRV					
		<table border="1"> <tr><td>PRV</td></tr> <tr><td>C1</td></tr> <tr><td>C2</td></tr> <tr><td>D</td></tr> </table>	PRV	C1	C2	D
PRV						
C1						
C2						
D						

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
C1	指定連接埠	—	1
C2	控制資料	—	1
D	儲存現在值的下位 CH 編號	WORD	可調整

### C1：指定連接埠

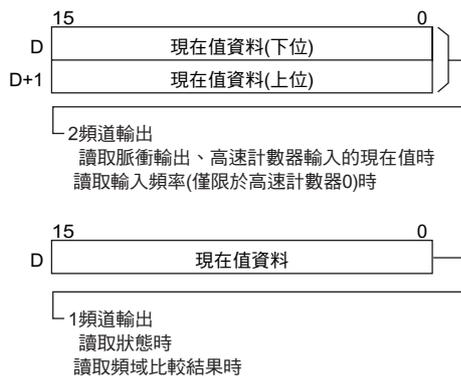
指定連接埠 (16 進位)	
#0000	脈衝輸出 0
#0001	脈衝輸出 1
#0010	高速計數器輸入 0
#0011	高速計數器輸入 1
#0012	高速計數器輸入 2
#0013	高速計數器輸入 3
#0014	高速計數器輸入 4
#0015	高速計數器輸入 5
#1000	PWM 輸出 0

(註) CP1E 10 點輸入型無法使用高速計數器輸入 5。

### C2：控制資料

控制資料 (16 進位)	
#0000	讀取現在值
#0001	讀取狀態
#0002	讀取頻域比較結果
#00 □ 3	當 C1 = #0000 或 #0001 時：讀取脈衝輸出 0 或 1 的頻率 #0003 當 C1 = #0010 時：讀取高速計數器輸入 0 的頻率 #0013：10 ms 取樣方式 #0023：100 ms 取樣方式 #0033：1s 取樣方式

## D：儲存現在值的下位 CH 編號



## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C1,C2	—	—	—	—	—	—	—	—	—	○	—	—	—
D	○	○	○	○	○	○	○	○	○	—	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>超出 C1、C2 所指定的範圍時，本旗標就會 ON</li> <li>當 C1 和 C2 不支援時，本旗標就會 ON</li> <li>在未執行頻域比較動作的狀態下，指定為讀取頻域比較結果時，本旗標就會 ON</li> <li>將高速計數器 0 以外的連接埠指定為讀取輸入頻率時，本旗標就會 ON</li> <li>當連接埠未被設定為高速計數器，即執行本指令時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

針對 C1 所指定的連接埠，讀取 C2 所指定的資料。

可指定的 C1、C2 組合如下：

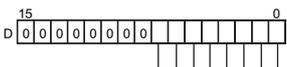
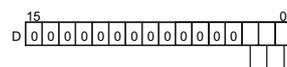
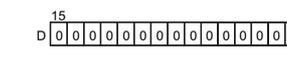
C1 (指定連接埠)	C2 (控制資料)		
	讀取現在值 (#0000)	讀取狀態 (#0001)	讀取頻域比較結果 (#0002)
脈衝輸出 (#0000、#0001)	○	○	×
高速計數器輸入 (#0010 ~ #0015)	○	○	○
PWM 輸出 (#1000)	×	○	×

C1 (指定連接埠)	讀取頻率 (#00 □ 3)			
	一般方式 (#0003)	10 ms 取樣方式 (#0013)	100 ms 取樣方式 (#0023)	1s 取樣方式 (#0033)
脈衝輸出 (#0000、#0001)	○	×	×	×
高速計數器輸入 (#0000、#0001)	×	○* (僅限於高速計數器 0)	○* (僅限於高速計數器 0)	○* (僅限於高速計數器 0)
PWM 輸出 (#1000)	×	×	×	×

● 讀取現在值 (C2 = #0000)

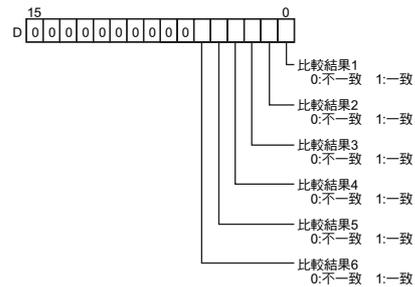
對象	內容	讀取結果的範圍
脈衝輸出 (C1 = #0000、#0001)	讀取脈衝輸出的現在值，並將結果儲存至 D+1、D。	-2147483648 ~ 2147483647 或 16 進位 #80000000 ~ 7FFFFFFF
高速計數器輸入 (C1 = #0010 ~ #0015)	線性模式下	讀取高速計數器的現在值，並將結果儲存至 D+1、D。
	連結模式下	10 進位 & 0 ~ 4294967295 或 16 進位 #00000000 ~ FFFFFFFF

● 讀取狀態 (C2 = #0001)

對象	內容	讀取結果的範圍
脈衝輸出	讀取脈衝輸出的狀態，並將結果儲存至 D。	 <ul style="list-style-type: none"> <li>脈衝輸出狀態 0: 定速中 1: 加速減速中</li> <li>現在值溢位(Overflow)/ 欠位(Underflow) 0: 正常 1: 發生中</li> <li>設定脈衝輸出量 0: 未設定 1: 已設定</li> <li>脈衝輸出完成 0: 輸出未完成 1: 輸出完成</li> <li>脈衝輸出中 0: 停止中 1: 輸出中</li> <li>無原點旗標 0: 原點確定狀態 1: 原點未確定狀態</li> <li>原點停止旗標 0: 未停止在原點 1: 原點停止中</li> <li>脈衝輸出停止的異常旗標 0: 無異常 1: 停止異常發生中</li> </ul>
高速計數器輸入	讀取高速計數器的狀態，並將結果儲存至 D。	 <ul style="list-style-type: none"> <li>動作比較 0: 停止中 1: 執行中</li> <li>現在值溢位(Overflow)/ 欠位(Underflow) 0: 正常 1: 發生中</li> <li>計數方向 0: 減算計數中 1: 加算計數中</li> </ul>
PWM 輸出	讀取 PWM 輸出的狀態，並將結果儲存至 D。	 <ul style="list-style-type: none"> <li>脈衝輸出中 0: 停止中 1: 輸出中</li> </ul>

### ● 讀取頻域比較結果 (C2 = #0002)

利用頻域比較型來比較高速計數器時，將會讀取 PRV 指令執行時的比較結果，然後再將結果儲存至 D。



### ● 讀取脈衝輸出或高速計數器頻率 (C2 = #00 □ 3)

讀取脈衝輸出 0 或 1 所輸出的脈衝頻率 (Hz)、或是輸入至高速計數器 0 的脈衝頻率 (Hz)，然後再將結果儲存至 D+1、D。

當 C1 = #0000 或 #0001 時：讀取脈衝輸出 0 或 1 的頻率

轉換結果 = 10 進位的 &0 ~ 100000 或 16 進位的 #00000000 ~ 000186A0

C1 = #0010 場合：讀取高速計數器輸入 0 的頻率

當計數器輸入方式為 4 倍相位差以外的方式時：

轉換結果 = 10 進位的 &0 ~ 100000 或 16 進位的 #00000000 ~ 000186A0

(註) 當輸入頻率超過 100 kHz 時，輸出的數值就會被固定為最大值 (10 進位的 &100000)。

計數輸入方式為 4 倍相位差時：

轉換結果 = 10 進位的 &0 ~ 200000 或 16 進位的 #00000000 ~ 00030D40

(註) 當輸入頻率超過 200 kHz 時，輸出的數值就會被固定為最大值 (10 進位的 &200000)。

### ● 脈衝頻率計算方式

測量固定時間 ( 取樣時間 ) 內的計數脈衝並計算頻率。依取樣時間的時間寬度不同，可選擇的種類包含下列 3 種。利用 C2 下方的第 2 位數來指定。

- 10 ms 取樣方式 ( 每隔 10 ms 計算脈衝數的方式 ) :  
C2 = 16 進位的 #0013 ( 請參閱：1 kHz 可到最大 10% 的誤差 )
- 100 ms 取樣方式 ( 每隔 100 ms 計算脈衝數的方式 ) :  
C2 = 16 進位的 #0023 ( 請參閱：1 kHz 可到最大 1% 的誤差 )
- 1s 取樣方式 ( 每隔 1s 計算脈衝數的方式 ) :  
C2 = 16 進位的 #0033 ( 請參閱：1 kHz 可到最大 0.1% 的誤差 )

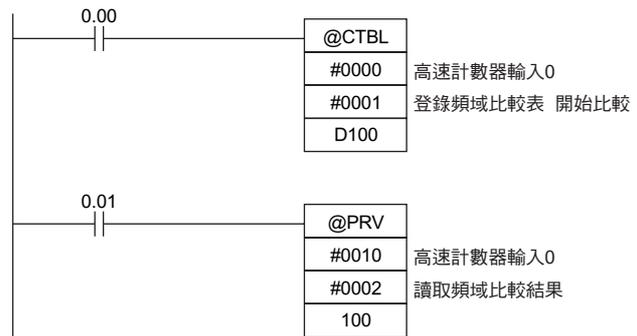
### 使用上的注意事項

- 指定 C1 = 0010 HEx ( 高速計數器輸入 0 )，同時還指定 C2 = 0013 Hex、0023 Hex 或 0033 HEx ( 讀取頻率 ) 時，只要將計數值復歸，就會造成取樣時間內所讀取到的資料不確定的情形，此點需特別注意。。

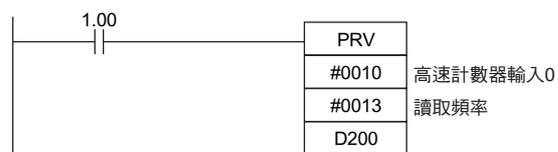
## 程式例

當 0.00 由 OFF → ON 時，就會利用 CTBL 指令，對高速計數器輸入 0 登錄頻域比較表，並且開始進行比較。

當 0.01 由 OFF → ON 時，就會利用 PRV 指令，將該時點的頻域比較結果讀入 0100。

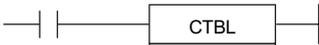


當 1.00 變成 ON 時，就會利用 PRV 指令，讀取該時點被輸入高速計數器輸入 0 的脈衝頻率，然後再以 16 進制的方式來輸出 D201、D200。



# CTBL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
登錄比較資料表	CTBL	@CTBL	882	針對高速計數器的現在值，進行目標值一致性比較或頻域比較。

符號	CTBL								
		<table border="1"> <tr> <td>CTBL</td> <td></td> </tr> <tr> <td>C1</td> <td>C1：指定連接埠</td> </tr> <tr> <td>C2</td> <td>C2：控制資料</td> </tr> <tr> <td>S</td> <td>S：比較資料表下位CH編號</td> </tr> </table>	CTBL		C1	C1：指定連接埠	C2	C2：控制資料	S
CTBL									
C1	C1：指定連接埠								
C2	C2：控制資料								
S	S：比較資料表下位CH編號								

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
C1	指定連接埠	—	1
C2	控制資料	—	1
S	比較資料表下位 CH 編號	LWORD	可調整

### C1：指定連接埠

指定連接埠 (16 進位)	
#0000	高速計數器輸入 0
#0001	高速計數器輸入 1
#0002	高速計數器輸入 2
#0003	高速計數器輸入 3
#0004	高速計數器輸入 4
#0005	高速計數器輸入 5

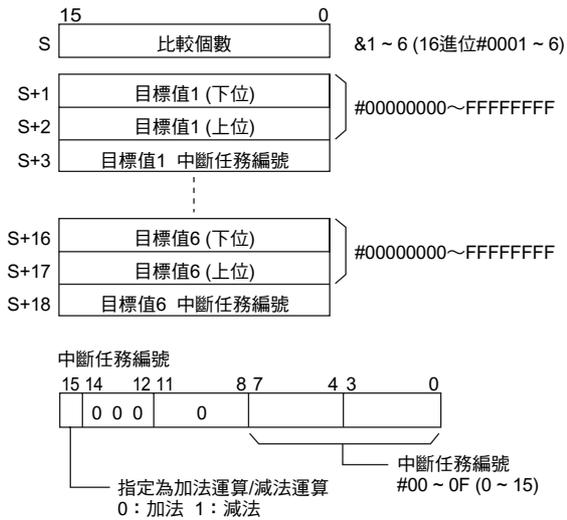
(註) CP1E 10 點輸入型無法使用高速計數器輸入 5。

### C2：控制資料

控制資料 (16 進位)	
#0000	登錄目標值一致比較表且開始比較
#0001	登錄比較資料表且開始比較
#0002	僅登錄目標值一致比較表
#0003	僅登錄頻域比較表

### S：比較資料表下位 CH 編號

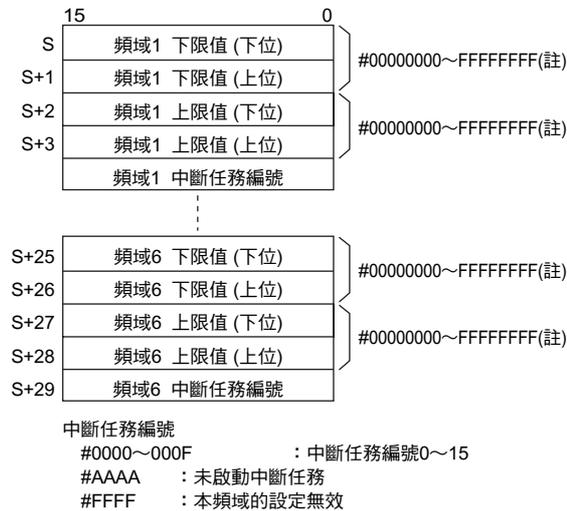
- 指定為目標值一致比較表時  
可依 S 的比較個數不同，調整 4 ~ 19 的 CH 長度。



● 指定為頻域比較表時

請務必指定 6 個頻域，並且將 30CH 設定為固定長度。

當設定值小於 6 個時，就會將中斷任務編號指定為 16 進位的 #FFFF。



(註)設定頻域的上限值及下限值時，需符合上限值 ≥ 下限值的條件。

● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C1,C2	-	-	-	-	-	-	-	-	-	○	-	-	-
S	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>超出 C1、C2 所指定的範圍時，本旗標就會 ON</li> <li>進行目標值一致的比較時，只要比較個數被設定為 0 個，本旗標就會 ON</li> <li>進行目標值不一致的比較時，只要比較個數超過 6 個，本旗標就會 ON</li> <li>進行頻域比較時，只要將上限值和下限值設定為反相，本旗標就會 ON</li> <li>進行頻域比較時，只要將所有的頻域設定值設定為無效，本旗標就會 ON</li> <li>高速計數器被設定為加法運算脈衝模式，可是卻指定執行減法運算時作為一致條件的話，本旗標就會 ON</li> <li>當高速計數器被設定為加法運算脈衝模式以及線性模式時，一旦在進行目標一致性比較時，設定多個相同的目標值，本旗標就會 ON</li> <li>設定為高速計數器的連結模式時，只要所指定的數值超過連結最大值並執行指令，本旗標就會 ON</li> <li>當連接埠未被設定為高速計數器，即執行本指令時，本旗標就會 ON</li> <li>執行比較動作時，只要利用不同的比較方法來執行指令，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

根據 C2 所指定的方式，對 C1 所指定的連接埠登錄可用來比較高速計數器現在值的資料表，然後再開始進行比較。

將曾經登錄過的比較表登錄為不同的資料表，或是進入「程式」模式前，本指令皆有效。

執行 CTBL 指令後，就會利用所指定的條件開始進行比較動作，

因此，基本上來說，本指令適用於輸入微分型(附@)或是持續ON動作1週期的輸入條件。(利用C2指定為開始比較時)

(註)一旦比較表中已經針對未被登錄為程式的中斷任務編號加以指定時，就會發生程式錯誤(運轉停止異常)的情形。

### ● 登錄比較資料表 (C2 = #0002、#0003)

僅登錄可用來和高速計數器現在值互相比較的資料表。

此時，只要執行 INI 指令即可開始進行比較。

### ● 登錄比較資料並開始比較 (C2 = #0000、#0001)

登錄可用來和高速計數器現在值互相比較的資料表，並且開始進行比較。

### ● 停止比較

如果要停止比較動作時，無論是利用 CTBL 指令開始進行比較、或是利用 INI 指令開始進行比較等任一場合皆須使用 INI 指令。

### ● 目標值一致性比較

當高速計數器現在值和資料表的目標值一致時，就會執行已指定的中斷任務。

- 亦可使用在比較多個中斷任務編號相同的中斷任務。

- 可指定加法運算計數時和減法計數時的一致性作為一致條件。

當資料表內的中斷任務編號最上位元被指定為 0，就執行加法運算，若指定為 1 時則執行減法運算。

- 比較資料表最多可登錄 6 個目標值。又，請將啟始 CH 所指定的個數設定在資料表中 (資料表的資料長度:可調整)。

- 目標值比較的對象為登錄於資料表中的所有數值。

(註)1在資料表中指定多個相同的目標值時，就會發生錯誤。

2當高速計數器被設定為加法運算脈衝模式時，一旦符合指定為減法運算時的一致條件時，就會發生錯誤。

3當計數方向(加法運算/減法運算)出現和目標值不一致的狀態時，將無法和該方向所對應的下一個目標值一致。設定目標值時，該數值不需要對應至計數值變化的峰值或谷值。

### ● 頻域比較

當高速計數器的現在值在上限值和下限值之間的範圍時，就會開始執行所指定的中斷任務。

- 亦可使用在比較多個中斷任務編號相同的中斷任務。

- 比較表可登錄 6 個頻域 (上限值和下限值的組合)，頻域可重覆指定。

- 當頻域小於 6 個時，只要將中斷任務編號指定為 16 進位的 #FFFF，即可忽略該頻域的設定值 (資料表的資料長度:固定(30CH))。

- 只有在條件一致的訊號上升時才會執行中斷任務。

執行 1 週期的比較時，一旦出現多個條件一致訊號上升的頻域，就會優先執行靠近資料表表頭的頻域，其他頻域則會在下一個週期以後執行。又，若不需要啟動中斷任務時，請將中斷任務編號指定為 16 進位的 #AAAA。

此時，只要參照執行 PRV 指令 (讀取頻域比較結果) 或頻域比較條件一致性圖表，即可確認比較結果。

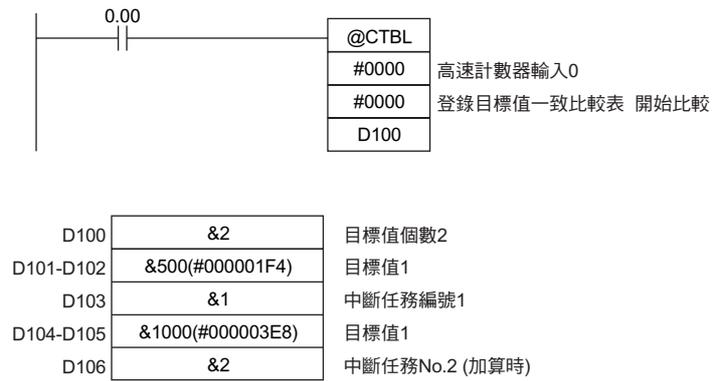
(註)1當上限值和下限值被指定為反相時，就會發生錯誤。

## 程式例

在 0.00 由 OFF → ON 時，藉由 CTBL 命令對高速計數輸入 0 登錄目標值一致比較表並開始進行比較。

當高速計數器的現在值朝加法運算方向計數，並且到達 500 時，此時數值和目標值 1 一致，因此會執行中斷任務 No.1。

繼續執行加法運算計數，並且到達 1000 時，此時數值和目標值 2 一致，因此會執行中斷任務 No. 2。



# SPED

指令名稱	指令記號	指令的各種組合	Fun No.	功能
頻率設定	SPED	@SPED	885	指定頻率並執行無加減速的脈衝輸出。

符號	SPED								
		<table border="1"> <tr> <td>SPED</td> <td></td> </tr> <tr> <td>C1</td> <td>C1：指定連接埠</td> </tr> <tr> <td>C2</td> <td>C2：輸出模式</td> </tr> <tr> <td>S</td> <td>S：目標頻率的下位CH編號</td> </tr> </table>	SPED		C1	C1：指定連接埠	C2	C2：輸出模式	S
SPED									
C1	C1：指定連接埠								
C2	C2：輸出模式								
S	S：目標頻率的下位CH編號								

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

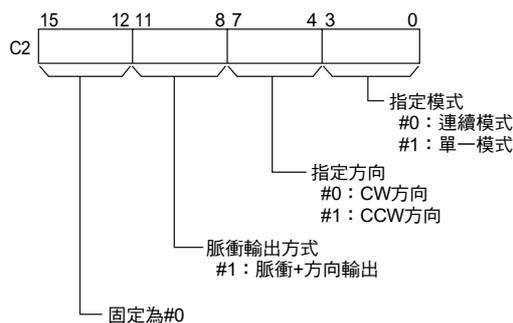
## 運算元種類的說明

運算元種類	內容	資料型態	容量
C1	指定連接埠	UINT	1
C2	輸出模式	WORD	1
S	目標頻率的下位 CH 編號	UDINT	2

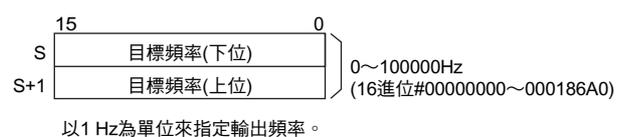
### C1：指定連接埠

指定連接埠 (16 進位)	
#0000	脈衝輸出 0
#0001	脈衝輸出 1

### C2：指定模式



### S：目標頻率的下位 CH 編號



## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C1,C2	—	—	—	—	—	—	—	—	—	○	—	—	—
S	○	○	○	○	○	○	○	○	○	—	—	—	—

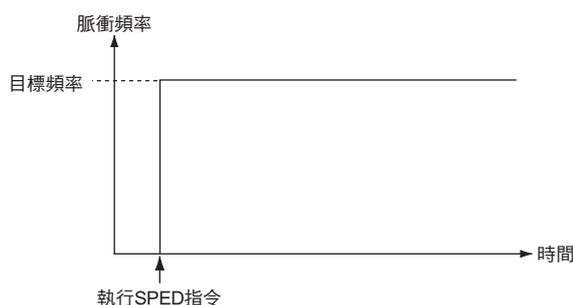
## 條件旗標的動作

名稱	標籤	內容
異常旗標	P.ER	<ul style="list-style-type: none"> <li>超出 C1、C2、S 所指定的範圍時，本旗標就會 ON</li> <li>當本指令被用在已經利用 PLS2、ORG 指令進行脈衝輸出的連接埠時，本旗標就會 ON</li> <li>變更本指令的指定方式為單一 / 連續模式，並且用在已經利用 SPED、ACC 指令進行脈衝輸出的連接埠時，本旗標就會 ON</li> <li>在週期執行任務中執行用來控制脈衝輸出的指令時，一旦出現中斷訊號，並且在中斷任務中執行本指令，異常旗標就會 ON</li> <li>在原點未確定的狀態下，即指定單一模式 / 絕對脈衝指定以執行本指令時，異常旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

執行本指令，即可依 S 所指定的「目標頻率」以及 C2 所指定的方式，由 C1 所指定的連接埠開始執行脈衝輸出。

執行 1 次 SPED 指令後，就會依指定的條件開始執行脈衝輸出。因此，基本上來說，本指令適用於輸入微分型 (附 @) 或是持續 ON 動作 1 週期的輸入條件。



在單一模式下，事先輸出 PULS 指令所設定的脈衝量時，系統就會自動停止脈衝輸出。

在連續模式下，系統將持續進行脈衝輸出直到執行停止脈衝輸出動作為止。

在單獨模式的脈衝輸出狀態下，變更為連續模式，或是在連續模式的脈衝輸出狀態下，變更為單獨模式，就會發生錯誤，並且無法執行動作。

(註) SPED 指令僅適用於 CP1E CPU 模組 N/NA 型電晶體輸出型。

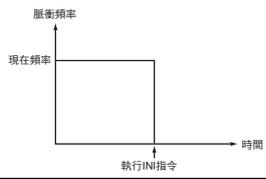
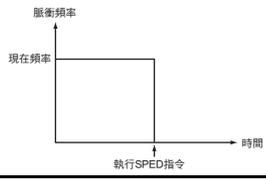
E 型或繼電器輸出型就會執行 NOP 處理。

### ● 連續模式 (速度控制)

持續脈衝輸出直到執行停止脈衝輸出為止。

(註) 進入「程式」模式後，就會立刻停止衝輸出。

操作	動作內容	使用範例	頻率變化	說明	使用步驟
					指令記號
開始脈衝輸出	輸出速度指定	當速度 (頻率) 呈階梯形變化時		執行指定頻率的脈衝輸出。	SPED (連續)
變更設定	變更階梯形速度	希望在運轉時變更速度		將脈衝輸出中的頻率變更為階梯形 (朝上或朝下)。	SPED (連續) ↓ SPED (連續)

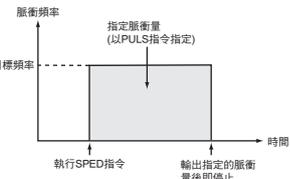
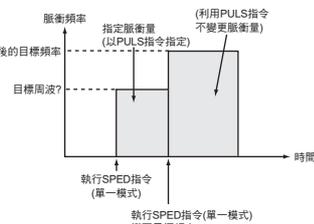
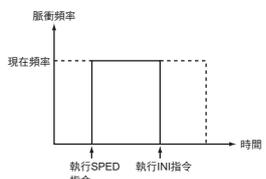
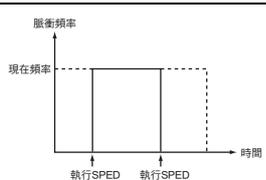
操作	動作內容	使用範例	頻率變化	說明	使用步驟
					指令記號
停止脈衝輸出	停止脈衝輸出	立即停止		立刻停止脈衝輸出。	SPED (連續) ↓ INI
	停止脈衝輸出	立即停止		立刻停止脈衝輸出。	SPED (連續) ↓ SPED (連續、目標頻率 0 Hz)

### ● 單一模式 (定位)

只有以 PULS 指令設定的脈衝輸出量才會執行脈衝輸出。

(註)

- 轉換為「程式」模式後，將會立即停止脈衝輸出。
- 停止脈衝輸出後，如果要再次執行輸出，就必須利用 PULS 指令來設定脈衝輸出量。
- 請先利用 PULS 指令來設定脈衝輸出量。即使在未設定脈衝輸出量的狀態下執行 SPED 指令，系統仍然不會輸出脈衝。
- 利用 PULS 指令，以絕對脈衝指定的方式來設定移動脈衝量時，由 SPED 指令的運算元種類所指定的方向 (CW 方向 /CCW 方向) 就會被忽略。

操作	動作內容	使用範例	頻率變化	說明	使用步驟
					指令記號
開始脈衝輸出	輸出速度指定	執行未加減速的定位時		以指定脈衝輸出的頻率開始輸出指定的脈衝量時，就會立刻停止動作。	PULS ↓ SPED (單一)
變更設定	變更階梯形速度	希望在運轉時將速度變更為階梯形時		定位時會執行 SPED 指令，並且將脈衝輸出時的頻率變更為階梯形 (朝上或朝下)，此時，目標位置 (脈衝量) 不變。	PULS ↓ SPED (單一) ↓ SPED (單一)
停止脈衝輸出	停止脈衝輸出 (不保持脈衝輸出量)	立即停止		立刻停止脈衝輸出。此時，目前的脈衝輸出量會被清除。	PULS ↓ SPED (單一) ↓ INI
	停止脈衝輸出 (不保持脈衝輸出量)	立即停止		立刻停止脈衝輸出。此時，目前的脈衝輸出量會被清除。	PULS ↓ SPED (單一) ↓ SPED (單一、目標頻率 0 Hz)

# PULS

指令名稱	指令記號	指令的各種組合	Fun No.	功能
脈衝量設定	PULS	@PULS	886	設定脈衝輸出量。

符號	PULS	
		C1：指定連接埠 C2：控制資料 S：設定脈衝輸出量的下位CH編號

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
C1	指定連接埠	—	1
C2	控制資料	—	1
S	設定脈衝輸出量的下位 CH 編號	DINT	2

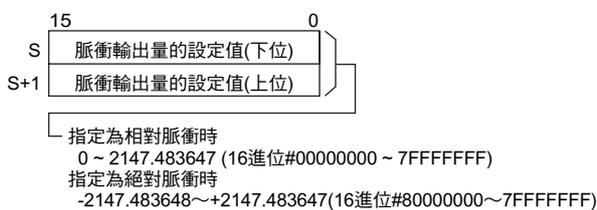
### C1：指定連接埠

指定連接埠 (16 進位)	
#0000	脈衝輸出 0
#0001	脈衝輸出 1

### C2：控制資料

控制資料 (16 進位)	
#0000	指定相對脈衝
#0001	指定絕對脈衝

### S：設定脈衝輸出量的下位 CH 編號



指定為相對脈衝和絕對脈衝時，實際輸出的絕對脈衝量如下：

- 指定為相對脈衝時  
移動的脈衝量 = 脈衝輸出量的設定值
- 指定為絕對脈衝時  
移動的脈衝量 = 脈衝輸出量的設定值 - 現在值

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR	
	CIO	WR	HR	AR	T	C	DM	@DM	* DM					
C1,C2	—	—	—	—	—	—	—	—	—	—	○	—	—	—
S	○	○	○	○	○	○	○	○	○	○	○	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>超出 C1、C2、S 所指定的範圍時，本旗標就會 ON</li> <li>當本指令被用在已經輸出脈衝的連接埠時，本旗標就會 ON</li> <li>在週期執行任務中執行用來控制脈衝輸出的指令時，一旦出現中斷訊號，並且在中斷任務中執行本指令，異常旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

針對 C1 所指定的連接埠，設定 C2、S 所指定的方式 / 脈衝輸出量。

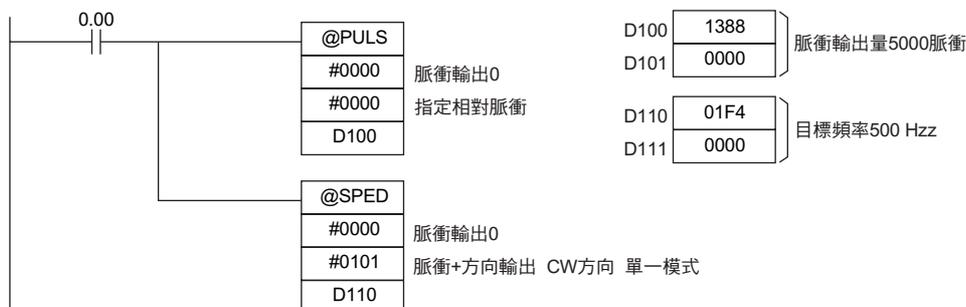
只要在單一模式下執行頻率設定 (SPED) 指令或頻率加減速控制 (ACC) 指令，即可輸出 PULS 指令所設定的脈衝輸出量。

(註)

- 如果在脈衝輸出時執行 PULS 指令，就會發生錯誤，此時無法重新設定脈衝輸出量。
- 因此，基本上來說，本指令只能在輸入微分型 (附 @) 或是持續 ON 動作 1 週期的輸入條件下使用。
- 執行本指令後，即使利用動作模式控制 (INI) 指令來變更脈衝輸出的現在值，也無法變更計算出來的移動脈衝量。
- 利用絕對脈衝指定的方式來設定移動脈衝量時，則 SPED 指令及 ACC 指令的運算元種類所指定的方向 (CW 方向 / CCW 方向) 皆會被忽略。
- 亦可指定為變更至超出脈衝輸出量的現在值範圍 (-2147483648 ~ 2147483647)。
- PLUS 指令僅適用於 CP1E CPU 模組 N/NA 型電晶體輸出型。  
E 型或繼電器輸出型就會執行 NOP 處理。

## 程式例

當 0.00 由 OFF → ON 時，就會利用 PULS 指令，以相對脈衝指定的方式，將脈衝輸出 0 的脈衝輸出量設定為 5000 脈衝。同時，還會利用 SPED 指令，以脈衝 + 方向方式、CW 方向、單一模式，開始輸出目標頻率 500Hz 的脈衝。



# PLS2

指令名稱	指令記號	指令的各種組合	Fun No.	功能
定位	PLS2	@PLS2	887	指定脈衝頻率及加減速比率，以執行具有加減速功能的脈衝輸出 ( 加速比率 ≠ 減速比率 )。

符號	PLS2						
		<table border="1"> <tr><td>PLS2</td></tr> <tr><td>C1</td></tr> <tr><td>C2</td></tr> <tr><td>S1</td></tr> <tr><td>S2</td></tr> </table>	PLS2	C1	C2	S1	S2
PLS2							
C1							
C2							
S1							
S2							

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

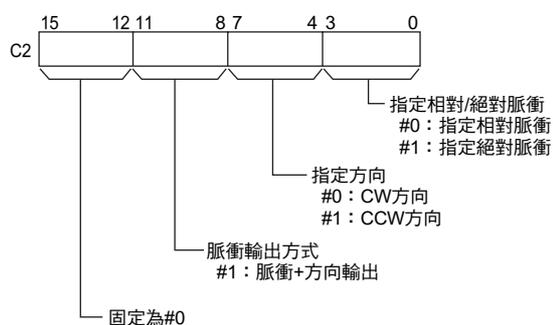
## 運算元種類的說明

運算元種類	內容	資料型態	容量
C1	指定連接埠	—	1
C2	控制資料	—	1
S1	設定資料表的下位 CH 編號	WORD	6
S2	啟動頻率的下位 CH 編號	UDINT	2

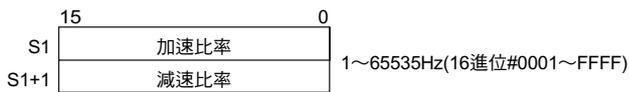
### C1：指定連接埠

指定連接埠 (16 進位)	
#0000	脈衝輸出 0
#0001	脈衝輸出 1

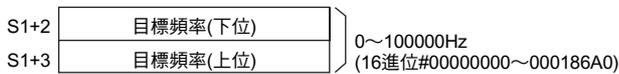
### C2：控制資料



## S1：設定資料表的下位 CH 編號



以1 Hz為單位，依不同的脈衝控制週期(4 ms)，分別指定其頻率增量。



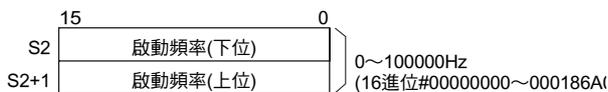
以1 Hz為單位，指定加速、減速後的頻率。



指定為相對脈衝和絕對脈衝時，實際輸出的絕對脈衝量如下：

- 指定為相對脈衝時  
移動的脈衝量 = 脈衝輸出量的設定值
- 指定為絕對脈衝時  
移動的脈衝量 = 脈衝輸出量的設定值 - 現在值

## S2：啟動頻率的下位 CH 編號



以1 Hz為單位，指定啟動時的頻率。

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C1,C2	-	-	-	-	-	-	-	-	-	○	-	-	-
S1	○	○	○	○	○	○	○	○	○	-	-	-	-
S2	○	○	○	○	○	○	○	○	○	○	-	-	-

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>超出 C1、C2、S1、S2 所指定的範圍時，本旗標就會 ON</li> <li>當本指令被用在已經利用 SPED、ORG 指令進行脈衝輸出的連接埠時，本旗標就會 ON</li> <li>在週期執行任務中執行用來控制脈衝輸出的指令時，一旦出現中斷訊號，並且在中斷任務中執行本指令，異常旗標就會 ON</li> <li>在原點未確定的狀態下，以絕對脈衝指定的方式來執行本指令時，異常旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

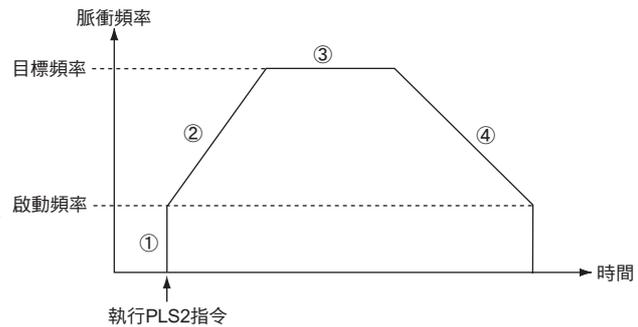
從 C1 上所被指定的埠開始利用 C2 上所被指定的方式，並以 S2 上所被指定的啟動頻率開始進行脈衝輸出 (右圖①)。

根據 S1 所指定的加速比率，增加不同脈衝控制週期 (4 ms) 的頻率，直到到達 S1 所指定的目標頻率為止 (右圖②)。

到達目標頻率後，即停止加速，並且以等速方式持續進行脈衝輸出 (右圖③)。

一旦到達由 S1 所指定的脈衝輸出量和減速比率所求出的減速點 (讓頻率減少的时间點)，就會依不同的脈衝控制週期 (4 ms) 分別減少其頻率，並且在到達啟動頻率後停止輸出 (右圖④)。

執行 1 次 PLS2 指令後，就會依指定的條件開始執行脈衝輸出。因此，基本上來說，本指令適用於輸入微分型 (附 @) 或是持續 ON 動作 1 週期的輸入條件。



PLS2 指令只能執行定位動作。

唯有利用 ACC 指令 (單一或連續模式下) 執行脈衝輸出時 (除了等速外，加速或減速狀態皆適用)，才能執行 PLS2 指令。(*1)(*2)。相反地，亦可利用 ACC 指令 (單一或連續模式下) 執行脈衝輸出時 (除了等速外，加速或減速狀態皆適用)，來執行 PLS2 指令。

- * 1 以 ACC 指令 (連續模式) 執行速度控制時，只要指定和 ACC 指令相同的目標頻率並執行 PLS2 指令，亦可執行中斷固定尺寸前進。此時，PLS2 指令無法執行加速動作，如果將加速比率設定為 0，異常旗標就會 ON，而且無法執行 PLS2 指令，請將加速比率設定為 0 以外的數值。
- * 2 如果在脈衝輸出停止前 ~ 停止時 1 個週期內執行 PLS2 指令，必須等到脈衝輸出停止後的下一個週期才會重新開始進行脈衝輸出。不過，若利用 INI 指令來停止脈衝輸出時，停止後 1 個週期以內的脈衝輸出指令皆會被視為無效。當脈衝輸出時，必須等到繼電器 OFF 後再執行本指令。
- * 3 PLS2 指令僅適用於 CP1E CPU 模組 N/NA 型電晶體輸出型。E 型或繼電器輸出型就會執行 NOP 處理。

### ● 單一模式 (定位)

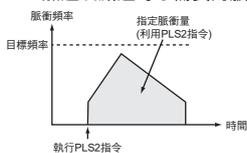
(註) 進入「程式」模式後，就會立刻停止衝輸出。

操作	動作內容	使用範例	頻率變化	說明	使用步驟
					指令記號
開始脈衝輸出	細部的梯形控制	進行梯形加減速的定位時 (加速比率和減速比率分別定位，且需要啟動速度) 可變更定位中的脈衝量		依固定比率，將頻率加速，並且依固定比率減速。一旦輸出指定的脈衝量，就會停止動作 (*1)。 (註) 可變更定位 (脈衝輸出) 中的目標位置 (脈衝量)。	PLS2

操作	動作內容	使用範例	頻率變化	說明	使用步驟
					指令記號
變更設定	變更附斜率的速度 (加速比率 ≠ 減速比率)	變更定位中的目標速度 (頻率) (加速比率和減速比率分別變更)		可在定位時執行 PLS2 指令，並且變更加速比率、減速比率、目標頻率。 (註) 如果要刻意不改變目標位置的話，必須利用絕對指定的方式指定為和原來相同的位置，以當作 PLS2 指令的運算元種類。	PLS2 ↓ PLS2 PULS ↓ ACC (單一) ↓ PLS2
	變更目標位置	變更定位中的目標位置 (多重啟動)		可在定位時執行 PLS2 指令，並且變更目標位置 (脈衝量)、加速比率、減速比率、目標頻率。 (註) 變更後若無法保留等速區，就會發生錯誤，此時執行動作會被忽略，且持續原來的動作。	PLS2 ↓ PLS2 PULS ↓ ACC (單一) ↓ PLS2
	變更附目標位置 + 斜率的速度	變更定位中的目標位置、目標速度 (頻率) (多重啟動)		可在定位時執行 PLS2 指令，並且變更目標位置 (脈衝量)、加速比率、減速比率、目標頻率。 (註) 變更後若無法保留等速區，就會發生錯誤，此時執行動作會被忽略，且持續原來的動作。	PULS ↓ ACC (單一) ↓ PLS2 PLS2 ↓ PLS2
		變更定位中的加減速 (多重啟動)		可在定位時 (加速或減速時) 執行 PLS2 指令，並且變更加速比率、減速比率。	PULS ↓ ACC (單一) ↓ PLS2 PLS2 ↓ PLS2
	變更方向	變更定位中的方向		以絕對脈衝指定的方式來定位時，可執行 PLS2 指令，並指定絕對位置，以變更方向。	PLS2 ↓ PLS2 PULS ↓ ACC (單一) ↓ PLS2
停止脈衝輸出	停止脈衝輸出 (不保持脈衝輸出量)	立即停止		立刻停止脈衝輸出。此時，目前的脈衝輸出量會被清除。	PLS2 ↓ INI
	停止附斜率脈衝輸出 (不保持脈衝輸出量)	停止減速		減速並停止脈衝輸出。	PLS2 ↓ ACC(單一、目標頻率 0 Hz)

* 1 三角控制

加速及減速時才需要的脈衝量，即使不足以到達目標頻率 (不會發生錯誤)，仍然會自動縮短加減速時間，並且執行僅適用於加減速的三角形控制。

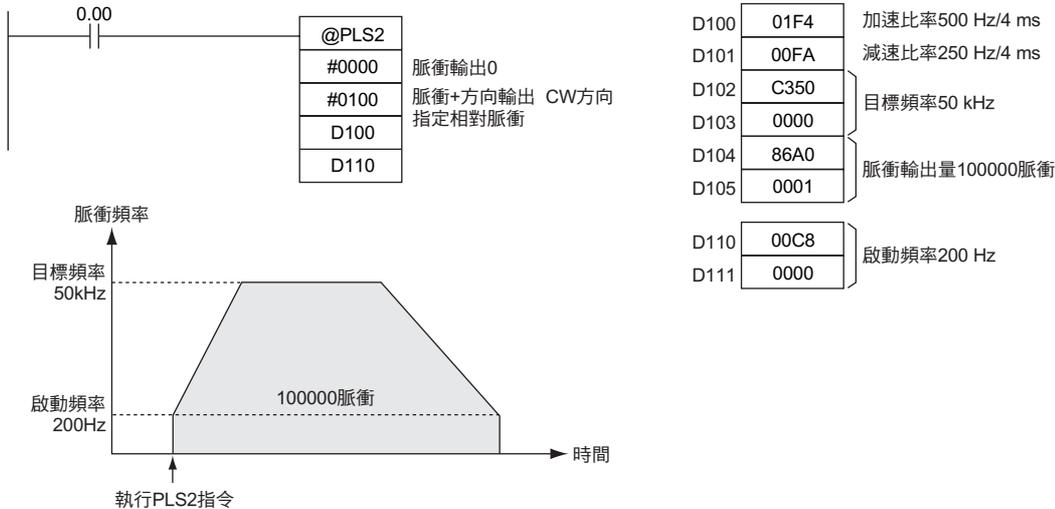


● 連續模式 (速度控制) → 單一模式 (定位)

使用範例	頻率變化	說明	使用步驟 指令記號
變更為速度控制時以固定量前進的定位方式		利用 ACC 指令，在速度控制狀態下執行 PLS2 指令，並且變更為定位。	ACC (連續) ↓ PLS2
中斷固定前進			

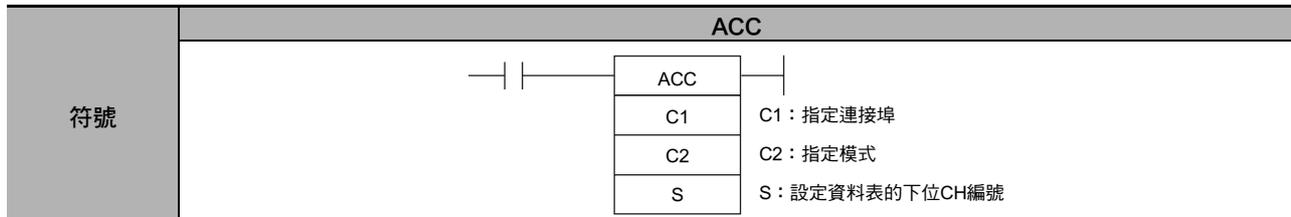
程式例

當 0.00 由 OFF → ON 時，就會利用 PLS2 指令，以相對脈衝指定的方式，由脈衝輸出 0 開始輸出 10000 脈衝。以 500 Hz/4 ms 的加速比率，由啟動頻率 200 Hz 開始加速直到到達目標頻率 50 kHz 為止。接著，再依 250 Hz/4 ms 的減速比例，由減速點開始進行減速，一旦減速到啟動頻率 200 Hz 後，即停止脈衝輸出。



# ACC

指令名稱	指令記號	指令的各種組合	Fun No.	功能
頻率加減速控制	ACC	@ACC	888	指定脈衝頻率及加減速比率，以執行具有加減速功能的脈衝輸出 (加速比率 = 減速比率)。



## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

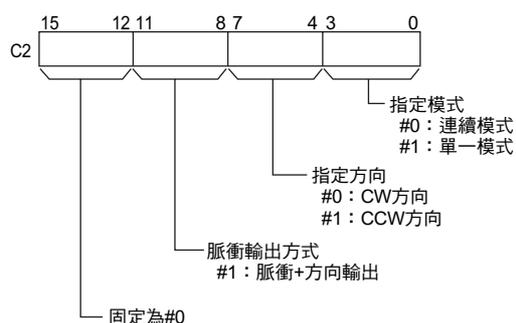
## 運算元種類的說明

運算元種類	內容	資料型態	容量
C1	指定連接埠	—	1
C2	指定模式	—	1
S	設定資料表的下位 CH 編號	WORD	3

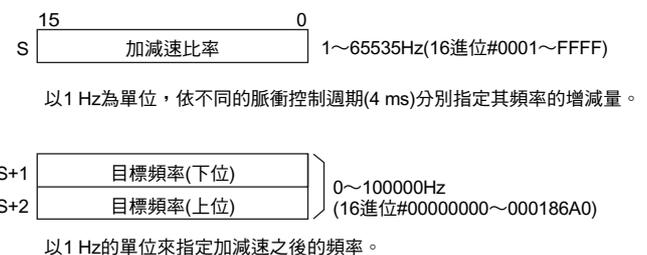
### C1：指定連接埠

指定連接埠 (16 進位)	
#0000	脈衝輸出 0
#0001	脈衝輸出 1

### C2：指定模式



### S：設定資料表的下位 CH 編號



## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
C1,C2	—	—	—	—	—	—	—	—	—	○	—	—	—
S	○	○	○	○	○	○	○	○	○	—	—	—	—

## 條件旗標的動作

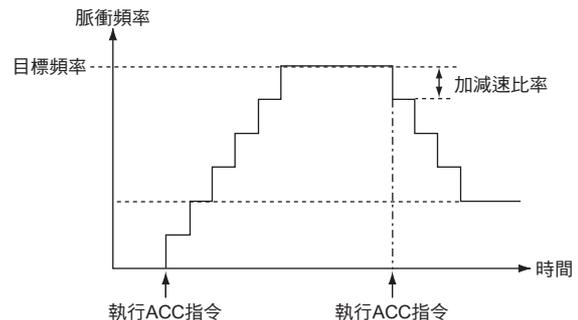
名稱	標籤	內容
異常旗標	P.ER	<ul style="list-style-type: none"> <li>超出 C1、C2、S 所指定的範圍時，本旗標就會 ON</li> <li>當本指令被用在已經利用 ORG 指令進行脈衝輸出的連接埠時，本旗標就會 ON</li> <li>變更本指令的指定方式為單一 / 連續模式，並且用在已經利用 SPED、ACC、PLS2 指令進行脈衝輸出的連接埠時，本旗標就會 ON</li> <li>在週期執行任務中執行用來控制脈衝輸出的指令時，一旦出現中斷訊號，並且在中斷任務中執行本指令，異常旗標就會 ON</li> <li>在原點未確定的狀態下，即指定單一模式 / 絕對脈衝指定以執行本指令時，異常旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

從 C1 上所被指定的埠開始利用 C2 上所被指定的方式，並以 S 上所被指定的「目標頻率」、「加減速比例」開始進行脈衝輸出。

又，根據 S 所指定的加減速比率，依不同的脈衝控制週期 (4 ms) 分別進行頻率的加減速，直到到達 S+2、S+1 所指定的目標頻率為止。

執行 1 次 ACC 指令後，就會依指定的條件開始執行脈衝輸出。因此，基本上來說，本指令適用於輸入微分型 ( 附 @ ) 或是持續 ON 動作 1 週期的輸入條件。



在單一模式下，事先輸出 PULS 指令所設定的脈衝量時，系統就會自動停止脈衝輸出。

在連續模式下，系統將持續進行脈衝輸出直到執行停止脈衝輸出動作為止。

在單獨模式的脈衝輸出狀態下，變更為連續模式，或是在連續模式的脈衝輸出狀態下，變更為單獨模式，就會發生錯誤，並且無法執行動作。

在依本 ACC 命令 ( 單獨或連續模式 ) 而執行脈衝輸出中 ( 等速、加速或減速中均可 ) 時，可執行 PLS2 命令 (*1)。另外，相反地，在依 PLS2 命令而執行的脈衝輸出中 ( 等速、加速或減速中均可 ) 時，亦可執行 PLS2 命令或本 ACC 命令 ( 單獨模式 )。

以 0 Hz 的目標頻率來執行 ACC 指令 ( 單一或連續模式 )，即使在脈衝輸出停止前執行 ACC 指令或 PLS2 指令，目標頻率仍然不會改變，而脈衝輸出則會保持停止狀態。ACC 指令或 PLS2 指令必須在脈衝輸出停止後執行，不得在脈衝輸出停止前執行。

- * 1 以 ACC 指令 ( 連續模式 ) 執行速度控制時，只要指定和 ACC 指令相同的目標頻率並執行 PLS2 指令，亦可執行中斷固定尺寸前進。此時，PLS2 指令無法執行加速動作，如果將加速比率設定為 0，異常旗標就會 ON，而且無法執行 PLS2 指令，請將加速比率設定為 0 以外的數值。
- * 2 如果在脈衝輸出停止前 ~ 停止時 1 個週期內 ( 脈衝輸出中繼電器為 ON ) 執行了 ACC 命令或 PLS2 指令時，必須等到脈衝輸出停止後的下一個週期才會重新開始進行脈衝輸出。  
不過，若利用 INI 指令來停止脈衝輸出時，停止後 1 個週期以內的脈衝輸出指令皆會被視為無效。當脈衝輸出時，必須等到繼電器 OFF 後再執行本指令。又，即使利用目標頻率 0 Hz 的 ACC 指令或 SPED 指令來停止以 SPED 指令輸出的脈衝輸出，其處理方式仍然和以 INI 指令來停止脈衝輸出時一樣，因此，必須等到脈衝輸出中繼電器 OFF，再執行脈衝輸出指令。
- * 3 ACC 指令僅適用於 CP1E CPU 模組 N/NA 型電晶體輸出型。  
E 型或繼電器輸出型就會執行 NOP 處理。

### ● 連續模式 ( 速度控制 )

持續脈衝輸出直到執行停止脈衝輸出為止。

( 註 ) 進入「程式」模式後，就會立刻停止衝輸出。

操作	動作內容	使用範例	頻率變化	說明	使用步驟 指令記號
開始脈衝輸出	加速度 / 速度 指定輸出	以固定比率進行 速度 ( 頻率 ) 加速時	<p>脈衝頻率</p> <p>目標頻率</p> <p>加減速比率</p> <p>時間</p> <p>執行 ACC 指令</p>	以固定比率來執行可改變 頻率的脈衝輸出。	ACC ( 連續 )

操作	動作內容	使用範例	頻率變化	說明	使用步驟
					指令記號
變更設定	變更附斜率的速度	希望運轉狀態下速度能緩慢變化時		可依固定比率，由目前頻率開始進行頻率的加速或減速。	ACC (連續) 或 SPED (連續) ↓ ACC (連續)
		希望以連續傾斜的方向來改變速度時		可在加速或減速時，變更加速比率或減速比率。	ACC (連續) ↓ ACC (連續)
		希望能一面改變速度同時停止動作時		可在減速時變更減速比率。 (註) 當目標頻率被設定為 0 Hz 時，減速比率動作中的值仍然有效。	ACC (連續) ↓ ACC (連續) ↓ ACC (連續、目標頻率 0 Hz)
停止脈衝輸出	停止脈衝輸出	立即停止		立刻停止脈衝輸出。	ACC (連續) ↓ INI
	停止附斜率脈衝輸出	停止減速		減速並停止脈衝輸出。 (註) 當第 2 次 ACC 指令的目標頻率到達 0Hz 時，就會依第 1 次 ACC 指令的減速比率進行減速。	ACC (連續) ↓ ACC (連續、目標頻率 0 Hz)

● 單一模式 (定位)

只有以 PULS 指令設定的脈衝輸出量才會執行脈衝輸出。

到達由設定的脈衝輸出量和加減速比率所求出的減速點 (讓頻率減少的时间點) 後，就會每隔 4 ms，針對減速比率所設定的頻率部分減少其頻率，依不同的脈衝控制週期 (4 ms) 分別減少其頻率，一旦到達脈衝輸出量時，就會停止脈衝輸出量。

(註)

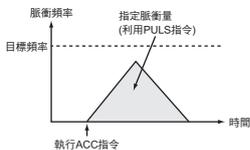
- 進入「程式」模式後，就會立刻停止衝輸出。
- 停止脈衝輸出後，如果要再次執行輸出，就必須利用 PULS 指令來設定脈衝輸出量。
- 請先利用 PULS 指令來設定脈衝輸出量。即使在未設定脈衝輸出量的狀態下執行 ACC 指令，系統仍然不會輸出脈衝。
- 利用 PULS 指令，以絕對脈衝指定的方式來設定移動脈衝量時，由 ACC 指令的運算元種類所指定的方向 (CW 方向 / CCW 方向) 就會被忽略。

操作	動作內容	使用範例	頻率變化	說明	使用步驟
					指令記號
開始脈衝輸出	單純的梯形控制	進行梯形加速的定時 (加速比率與減速比率一致。無起動速度)。在定位中無法改變脈衝量。		可依固定的比率進行頻率加速或減速，一旦輸出指定的脈衝量時，就會立刻停止動作 (*1)。 (註) 正在定位 (脈衝輸出) 中的目標位置 (脈衝量) 無法進行變更。	PULS ↓ ACC (單一)

操作	動作內容	使用範例	頻率變化	說明	使用步驟 指令記號
變更設定	變更具斜率的 速度 (加速比率 = 減速比率)	改變正在定位 中的目標速度 (頻率) (加速 比率與減速比 率相同)		可在定位時執行 ACC 指令， 並且變更加減速率、目標 頻率。此時，目標位置 (脈 衝量) 不變。	PULS ↓ ACC (單一) 或 SPED (單一) ↓ ACC (單一) PLS2 ↓ ACC (單一)
停止脈衝輸出	停止脈衝輸出 (不保持脈衝輸出量)	立即停止		減速並停止脈衝輸出。 (註) 執行 ACC 指令時的加減 速率，動作中的值仍保持有 效狀態。因此，利用 SPED 指令來啟動時，加減速率 將變為無效，因此會立刻停 止動作。	PULS ↓ ACC (單一) ↓ INI
	停止附斜率脈衝輸出 (不保持脈衝輸出量)	停止減速			PULS ↓ ACC (單一) または SPED (單一) ↓ ACC(單一、目 標頻率 0 Hz) PLS2 ↓ ACC(單一、目 標頻率 0 Hz)

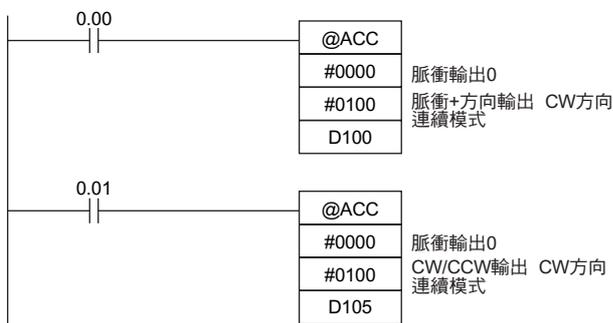
* 1 三角控制

加速及減速時才需要的脈衝量，即使不足以到達目標頻率 (不會發生錯誤)，仍然會自動縮短加減速時間，並且執行僅適用於加減速的三角形控制。

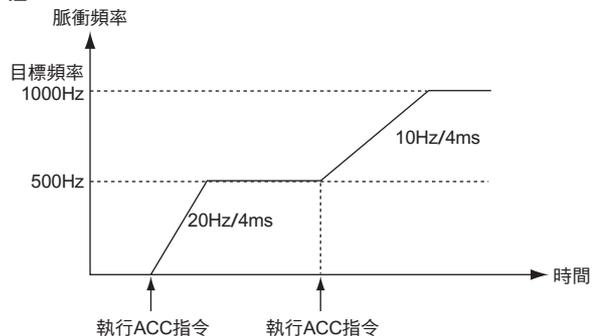


程式例

當 0.00 由 OFF → ON 時，就會利用 ACC 指令，依脈衝 + 方向方式、CW 方向、連續模式，由脈衝輸出 0 的連接埠開始輸出加減速率為 20 Hz、目標頻率為 500 Hz 的脈衝。接著，當 0.01 由 OFF → ON 時，就會再次利用 ACC 指令，將加減速率變更為 10 Hz，目標頻率變更為 1000 Hz。

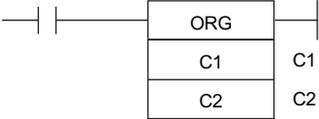


D100	0014	} 加速比率 20 Hz
D101	01F4	
D102	0000	
D105	000A	} 加速比率 10 Hz
D106	03EB	
D107	0000	



# ORG

指令名稱	指令記號	指令的各種組合	Fun No.	功能
原點搜尋	ORG	@ORG	889	執行原點搜尋或原點復歸。

符號	ORG	
		C1 : 指定連接埠

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

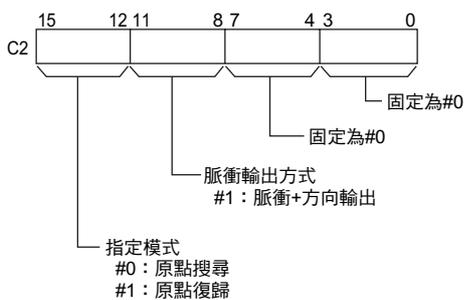
## 運算元種類的說明

運算元種類	內容	資料型態	容量
C1	指定連接埠	—	1
C2	控制資料	—	1

### C1：指定連接埠

指定連接埠 (16 進位)	
#0000	脈衝輸出 0
#0001	脈衝輸出 1

### C2：控制資料



### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C1,C2	—	—	—	—	—	—	—	—	—	○	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P.ER	<ul style="list-style-type: none"> <li>超出 C1、C2 所指定的範圍時，本旗標就會 ON</li> <li>當本指令被用在已經利用 SPED、ACC、PLS2 指令進行脈衝輸出的連接埠時，本旗標就會 ON</li> <li>在週期執行任務中執行用來控制脈衝輸出的指令時，一旦出現中斷訊號，並且在中斷任務中執行本指令，異常旗標就會 ON</li> <li>當 PLC 系統所設定的原點定位相關參數超出範圍時，本旗標就會 ON</li> <li>當 PLC 系統所設定的原點搜尋相關速度符合原點搜尋高速速度 $\leq$ 原點搜尋最近點速度或是原點搜尋最近點速度 $\leq$ 原點搜尋的啟動速度等條件時，本旗標就會 ON</li> <li>在未確定原點的狀態下，指定原點復歸時，除了 ON 以外，本旗標皆為 OFF</li> </ul>

## 功能

利用 C 所指定的方式，由 C 所指定的連接埠輸出脈衝，並且執行原點搜尋或原點復歸。執行 ORG 指令時，必須利用 PLC 系統設定方式，事先設定好下列參數。

原點搜尋	原點復歸
<ul style="list-style-type: none"> <li>原點搜尋功能的使用設定</li> <li>選擇動作模式</li> <li>指定原點搜尋動作</li> <li>指定原點檢測方法</li> <li>原點搜尋方向指定</li> <li>原點搜尋 / 原點復歸的啟動速度</li> <li>高速原點搜尋時之速度</li> <li>原點搜尋最近點速度</li> <li>原點搜尋的加速比率</li> <li>原點搜尋的減速比率</li> <li>原點修正資料</li> <li>原點最近點輸入的訊號種類</li> <li>原點輸入的訊號種類</li> <li>界限輸入訊號種類</li> <li>定位監控時間</li> </ul>	<ul style="list-style-type: none"> <li>原點復歸的目標速度</li> <li>原點搜尋 / 原點復歸的啟動速度</li> <li>原點復歸的加速比率</li> <li>原點復歸的減速比率</li> </ul>

執行 1 次 ORG 指令後，就會依所指定的條件開始原點定位動作。因此，基本上來說，本指令適用於輸入微分型 (附 @) 或是持續 ON 動作 1 週期的輸入條件。

(註) ORG 指令僅適用於 CP1E CPU 模組 N/NA 型電晶體輸出型。  
E 型或繼電器輸出型就會執行 NOP 處理。

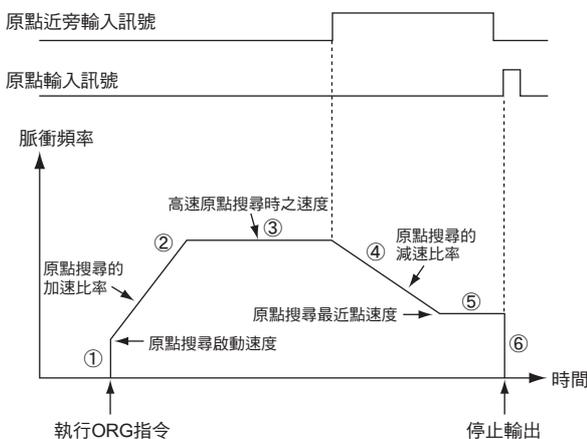
### ● 原點搜尋 (C2 位元 12 ~ 15 = 0)

依所指定的輸出方式及原點搜尋啟動速度，開始執行脈衝輸出。(右圖①)

根據原點搜尋的加速比率，持續加速直到到達原點搜尋的高速速度為止(右圖②)，接著以等速動作直到原點最近點輸入訊號變成 ON 為止。(右圖③)

當原點最近點輸入訊號變成 ON 時，就會根據原點搜尋減速比率，持續減速直到到達原點搜尋最近點速度為止(右圖④)，接著以等速動作直到原點輸入訊號變成 ON 為止。(右圖⑤) 當原點輸入訊號 ON 時，即停止脈衝輸出。(右圖⑥)

原點搜尋完成後，偏差計數器復歸輸出就會 ON。  
上述動作依動作模式或原點檢測方法等參數而異。

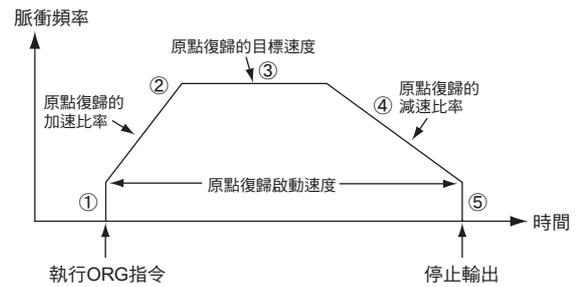


### ● 原點復歸 (C2 位元 12 ~ 15 = 1)

依所指定的輸出方式及原點復歸的啟動速度，開始進行脈衝輸出。(右圖①)

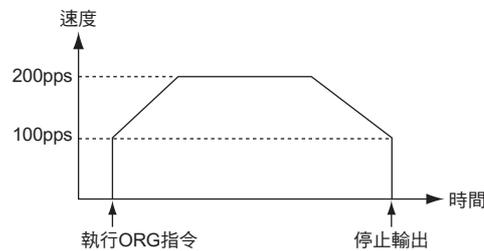
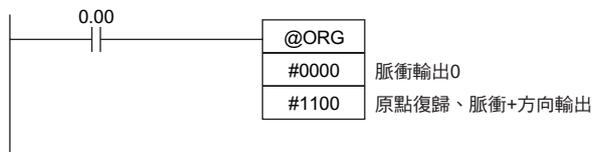
依原點復歸的加速比率，持續加速直到到達原點復歸的目標速度為止(右圖②)，接著將以等速進行動作。(右圖③)

到達由原點位置到達前所需的脈衝輸出量及原點復歸減速比率所求出的減速點(讓頻率減少的时间點)後，就會開始減速(右圖④)，接著持續減速，一旦到達原點復歸啟動速度後(到達原點後)，就會停止脈衝輸出。(右圖⑤)



### 程式例

當 0.00 由 OFF → ON 時，就會利用 ORG 指令，以輸出脈衝 + 方向方式輸出脈衝至脈衝輸出 0，並進行原點復歸。根據 PLC 系統的設定，啟動速度必須為 100 pps，目標速度為 200 pps，加速比率和減速比率為 50 Hz/4 ms。

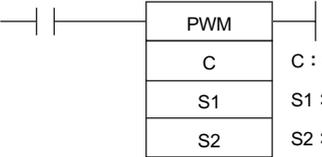


### 設定 PLC 系統

脈衝輸出 0 的原點搜尋 / 原點復歸啟動速度	100pps
脈衝輸出 0 的原點復歸目標速度	200pps
脈衝輸出 0 的原點復歸加速比率	50Hz/4ms
脈衝輸出 0 的原點復歸減速比率	50Hz/4ms

# PWM

指令名稱	指令記號	指令的各種組合	Fun No.	功能
PWM 輸出	PWM	@PWM	891	輸出可調整負載比的脈衝。

符號	PWM	
		C : 指定連接埠 S1 : 指定頻率 S2 : 指定負載比

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
C	指定連接埠	—	1
S1	指定頻率	—	1
S2	指定負載比	—	1

### C：指定連接埠

指定連接埠 (16 進位)	
#1000	脈衝輸出 0 (負載比單位為 0.1%，頻率單位為 0.1 Hz)
#1100	脈衝輸出 0 (負載比單位為 0.1%，頻率單位為 1 Hz)

### S1：指定頻率

2.0 Hz ~ 6553.5 Hz (指定以 0.1 Hz 為單位時)：10 進位 & 20 ~ 65535 或 16 進位 #0014 ~ FFFF  
 2 Hz ~ 32,000 Hz (指定以 1 Hz 為單位時)：10 進位 & 2 ~ 32000 或 16 進位 #0002 ~ 7D00

### S2：指定負載比

0.0 ~ 100.0% (指定以 0.1% 為單位時)：10 進位 & 0 ~ 1000 或 16 進位 #0000 ~ 03E8  
 以百分比來指定負載比 (頻率週期所對應的 ON 時間比率)。

## ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
C	—	—	—	—	—	—	—	—	—	○	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>超出 C、S1、S2 所指定的範圍時，本旗標就會 ON</li> <li>當本指令被用在已經利用 ORG 指令進行脈衝輸出的連接埠時，本旗標就會 ON</li> <li>在週期執行任務中執行用來控制脈衝輸出的指令時，一旦出現中斷訊號，並且在中斷任務中執行本指令，異常旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

依 S1 所指定的頻率，由 C 所指定的連接埠輸出 S2 所指定的負載比脈衝。利用 PWM 指令來輸出可調整負載比的脈衝輸出時，只要變更負載比的指定並執行 PWM 指令，不需要停止脈衝輸出，即可變更負載比。不過，頻率變更將會被忽略，而且無法執行動作。

另外，執行一次 PWM 指令後，即會以所指定的條件來開始進行脈衝輸出。因此，基本上來說，本指令適用於輸入微分型 (附 @) 或是持續 ON 動作 1 週期的輸入條件。

請利用下列方法來停止脈衝輸出：

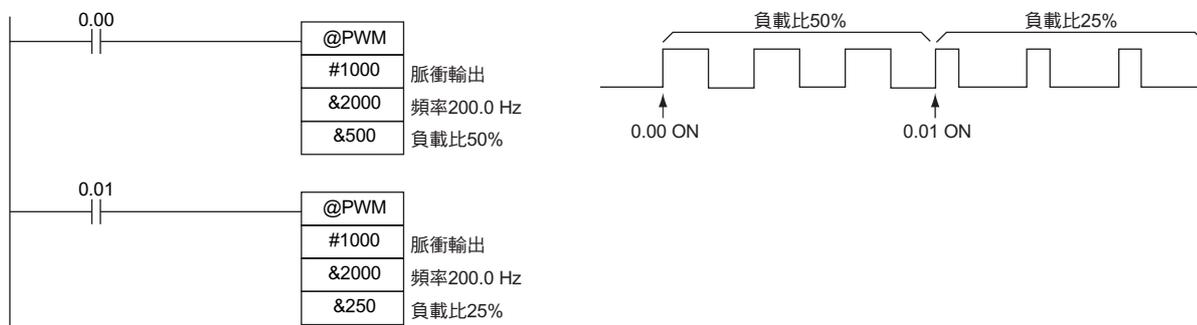
- (1) 執行 INI 指令 (C2 = #0003：停止脈衝輸出)
- (2) 進入「程式」模式

(註) PWM 指令僅適用於 CP1E CPU 模組 N/NA 型電晶體輸出型。  
E 型或繼電器輸出型就會執行 NOP 處理。

## 程式例

當 0.00 由 OFF → ON 時，就會利用 PWM 指令，以頻率 200Hz、負載比 50% 的條件，開始輸出脈衝至脈衝輸出 0。

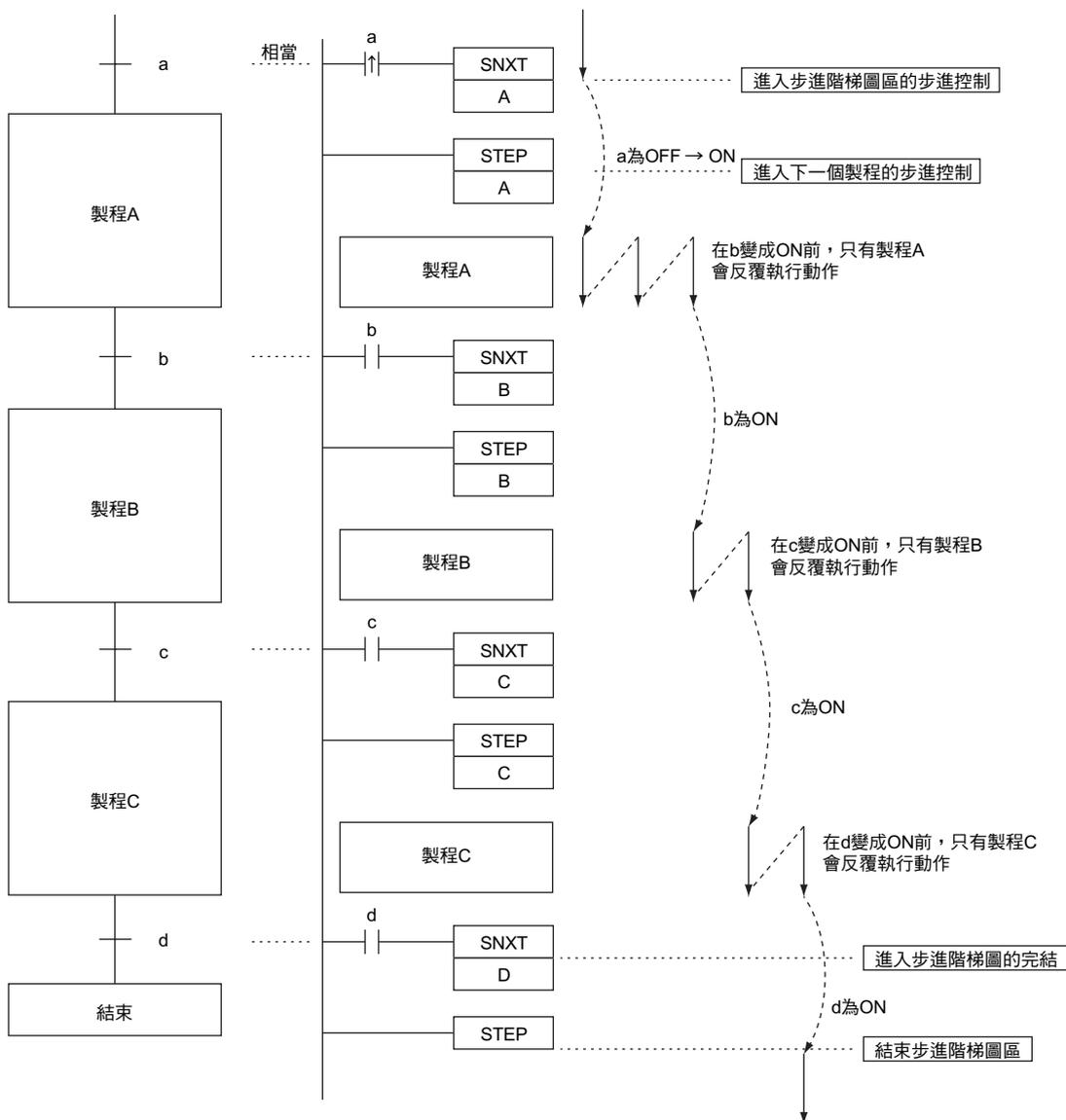
當 0.01 由 OFF → ON 後，負載比就會變更為 25%。



# 製程步進控制指令

使用 CP1E 系列時，只要將 SNXT( 步進階梯圖的步進 ) 指令和 STEP( 步進階梯圖的區域 ) 指令互相搭配，即可編寫製程步進程式。

指令	功能	製程圖
SNXT (步進階梯圖的步進)	執行進入下一個製程的步進控制。	
STEP (步進階梯圖區)	表示製程開始。在進入下一個製程的步進條件成立前，只有該製程的程式會反覆執行動作。	



(註) A、B、C、D的製程步進編號使用內部輔助繼電器WR。

# SNXT/STEP

指令名稱	指令記號	指令的各種組合	Fun No.	功能
前往步進階梯區	SNXT	—	009	執行步進階梯圖區的製程步進控制。
定義步進階梯區	STEP	—	008	定義步進階梯圖區的製程啟始及結束。

符號	SNXT	STEP
		<p>步進階梯圖的啟始指令 (指定製程編號時)</p>  <p>步進階梯圖區結束指令 (不指定製程編號時)</p> 

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	—	—

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	製程編號	—	1

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	—	○	—	—	—	—	—	—	—	—	—	—	—

## 條件旗標的動作

名稱	標籤	內容	
		SNXT	STEP
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S 所指定的資料超出內部輔助繼電器 WR 區的範圍時，本旗標就會 ON</li> <li>將本指定用在中斷任務時，異常旗標就會 ON</li> <li>否則為 OFF</li> </ul>	<ul style="list-style-type: none"> <li>當 S 的資料超出 WR 區時，本旗標就會 ON</li> <li>將本指定用在中斷任務時，異常旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 功能

### ● SNXT

依配置位置不同，分為下列 3 種功能。

- 1) 進入步進階梯圖區的步進控制
- 2) 進入下一個製程編號的步進控制
- 3) 進入結束步進階梯區的步進控制

步進階梯區就是 STEP 指令 ( 指定製程編號 ) ~ STEP 指令 ( 無製程編號 ) 之間的區域。

- 進入步進階梯圖區的步進控制

( 當本指令被放在整個步進階梯圖區的前面時 )

當製程編號 S 所指定的繼電器由 OFF → ON 時，就會進入製程 S (STEP 指令以後的製程) 的步進控制。

輸入條件需設定為上升微分型。( 因為如果將 SNXT 指令配置在步進階梯圖區以外的位置時，本指令將和 SET 指令執行相同的動作 )

- 進入下一個製程的步進控制

( 置於步進階梯區的全體當中時 )

將前一個製程編號 ( 繼電器 ) 由 ON → OFF，然後再將下一個製程的製程編號 ( 繼電器 ) S 由 OFF → ON，即可進入製程 S (STEP 指令以後的製程) 的步進控制。

- 進入結束步進階梯區的步進控制

( 置於步進階梯區的全體最後時 )

將前一個製程編號設定為 ON → OFF。S 所指定的製程編號為假體繼電器 ( 不過這時候繼電器會變為 ON，因此指定時，需指定為 ON 也不會出現問題的繼電器。 )

### ● STEP

本指令依是否指定欲配置的位置及製程編號，而有下列 2 種不同的功能。

- 1) 開始所指定的製程
- 2) 結束整個步進階梯區

- 開始所指定的製程

( 將本指令配置於各製程的前面，並且指定為 S 時 )

利用 SNXT 指令，將製程編號 S 由 OFF 上升為 ON 時，就會由本指令的下一個指令開始執行，同時，也會將 A200.12 ( 步進階梯圖 ON 1 週期旗標 ) 設定為 ON。

下一個週期開始，將反覆執行目前的製程，直到改變下一個製程的條件成立為止 ( 利用 SNXT 指令將下一個製程編號由 OFF → ON 為止 )。

利用 SNXT 指令，將下一個製程編號由 OFF 上升為 ON 時，本指令所指定的製程編號 S 就會被復歸 (ON → OFF)，而目前製程編號 S 的製程就會進入 IL 狀態 ( 互鎖中 )。

依 STEP 指令所指定的製程編號 S 的 ON/OFF 狀態不同，製程編號 S 在製程內的指令及輸出變化如下：當製程編號 S 由 ON 被復歸為 OFF 時，就會進入互鎖狀態 ( 請參閱表 1 )。

製程編號 ( 繼電器 ) 的狀態	動作
ON	以一般的動作來執行製程中的指令。
ON → OFF	讓製程內所使用的繼電器及指令進入表 1 的 IL ( 互鎖 ) 狀態。
OFF	為製程中的指令執行 NOP 處理。

表 1：IL ( 互鎖 ) 狀態

所有指令輸出		狀態
OUT、OUT NOT 指令所指定的繼電器		全部 OFF
計時器類指令 (100 ms 計時器 TIM/TIMX、10 ms 計時器 TIMH/TIMHX、1 ms 計時器 TMHH/TMHHX、長時間計時器 TIML/TIMLX 指令)	現在值	#0000 ( 復歸 )
	時間到旗標	OFF ( 復歸 )
其他指令 ( 註 ) 所指定的繼電器、CH		保持先前的狀態 ( 指令本身不會執行任何動作 )

( 註 ) 表示 TTIM/TTIMX、SET、RSET、CNT/CNTX、CNTR/CNTRX、SFT、KEEP 等所有其他的指令。

本指令必須被插入各製程的前方。

如果放在步進階梯區的啟始位置時，表示開始步進階梯區。

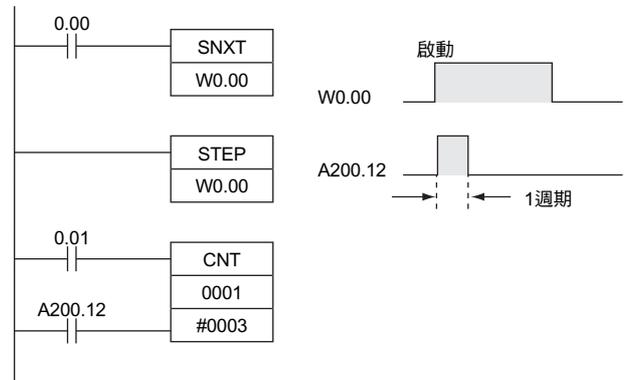
- 結束整個步進階梯區

(將本指令配置於整個步進階梯區，並且未指定 S 時)

利用 SNXT 指令，將 SNXT 指令前的製程編號由 ON 下降為 OFF 時，整個步進階梯圖區就會結束動作。

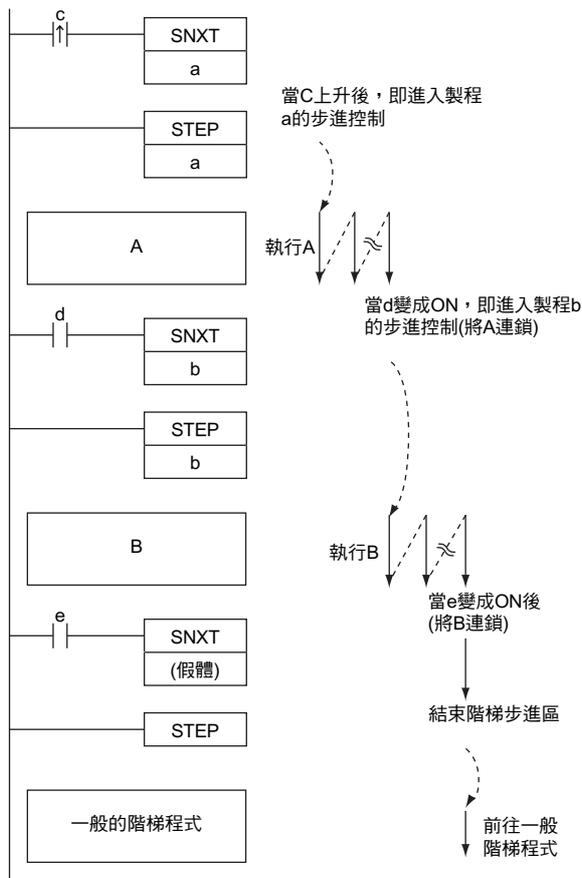
## 提示

本產品配備 1 個只會在製程開始時的 1 個週期內 ON 1 次的特殊輔助繼電器「步進階梯圖 ON 1 週期旗標」(A200.12)。請使用此旗標，執行進入製程時的初始化處理等。



## 相關特殊輔助繼電器

名稱	位址	內容
步進階梯圖 ON 1 週期旗標	A200.12	利用 STEP 指令來啟動某一個製程時，1 個週期就會變成 1 (ON)，本指令可在製程啟動時進行計數器初始化時使用。



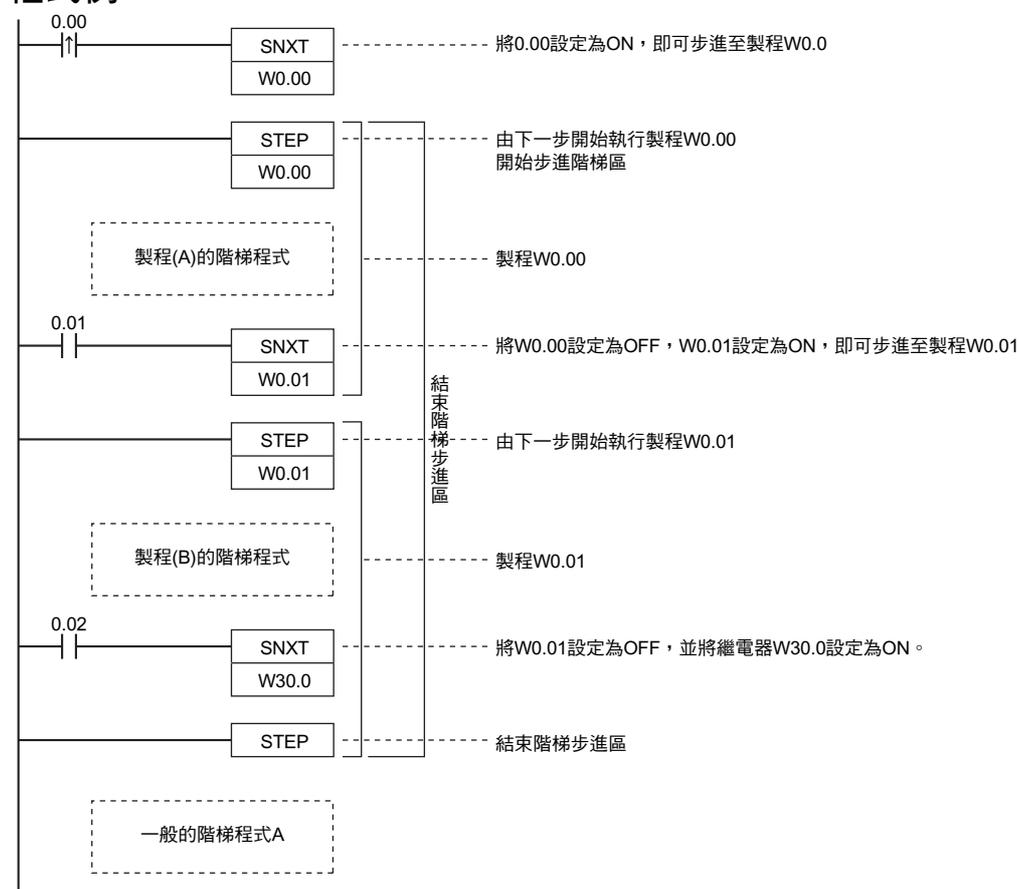
## 使用上的注意事項

- 請注意使用 SNXT/STEP 指令時，可指定於製程編號 S 的運算元種類只有內部輔助繼電器 WR。
- 使用 SNXT/STEP 指令時，在 S 所指定製程編號的位址不得和一般階梯回路所使用的位址重複，重複使用時，就會發生線圈重複使用錯誤。
- 當製程中出現 SBS (呼叫副程式) 指令時，即使製程編號由 ON → OFF，副程式中的輸出也不會進入 IL(互鎖狀態)，此點需特別注意。
- SNXT 指令只會在輸入訊號上升 (OFF → ON) 1 次時執行。
- 請將 SNXT 指令插入最後一個製程的結尾，此時，製程編號會被當作 繼電器編號，並且以製程編號的形式，當作未使用的內部輔助繼電器 WR 使用，指定製程編號所使用的繼電器後，即可啟動該製程。
- 本指令不適用在副程式中、中斷任務中、或是方塊程式區中。
- 在同一個週期中，無法同時啟動 2 個以上的階梯步進區。

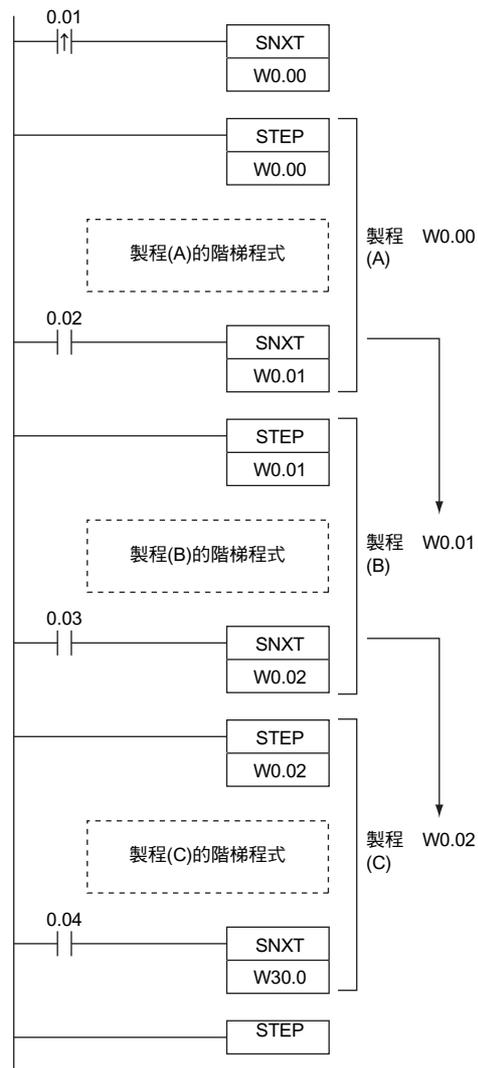
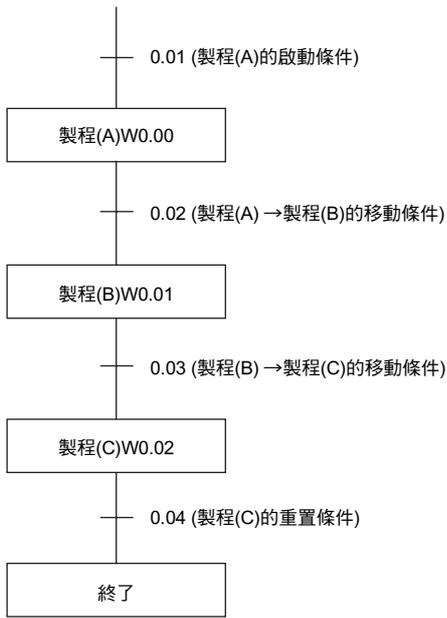
不適用於步進階梯程式區的指令如下：

依功能分類	指令記號	指令
序列控制	END	END
	IL	互鎖
	ILC	清除互鎖
	JMP	跳躍
	JME	跳躍結束 (Jump End)
	CJP	條件跳躍 (Jump)
副程式	SBN	副程式進入
	RET	副程式返回

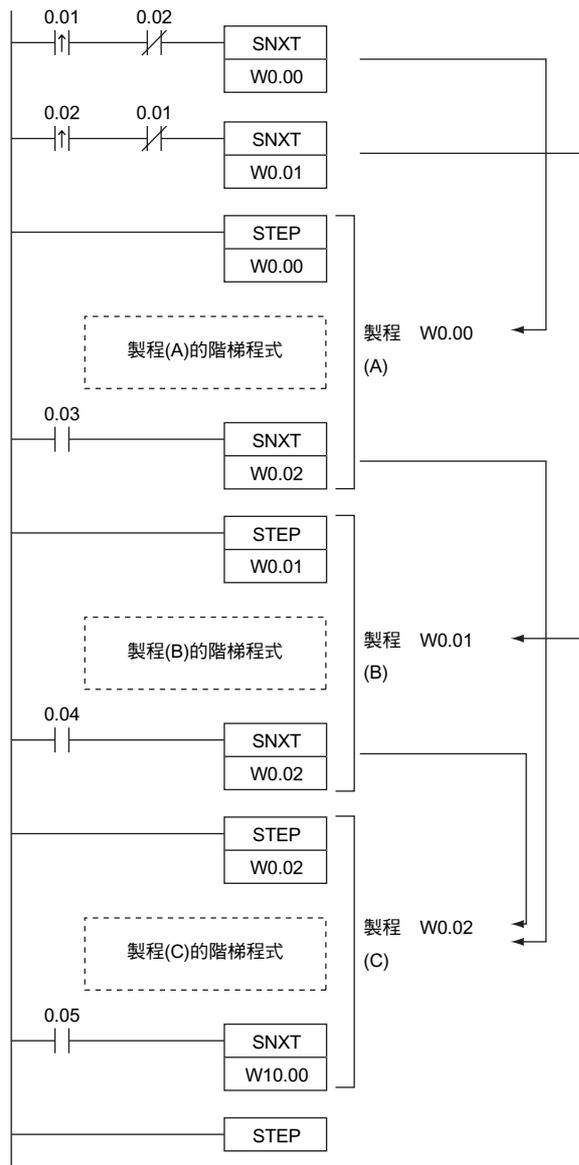
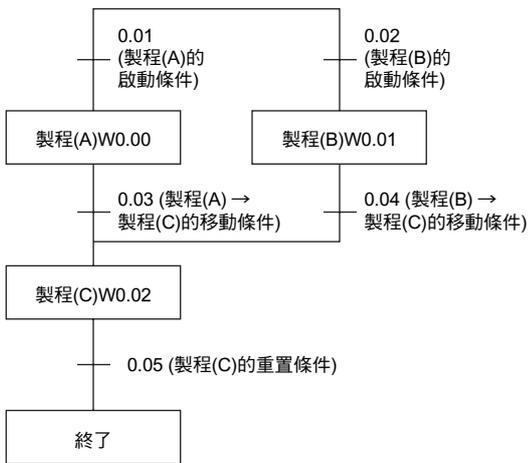
## 程式例



(1) 順序控制



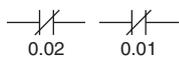
(2) 選擇分歧控制



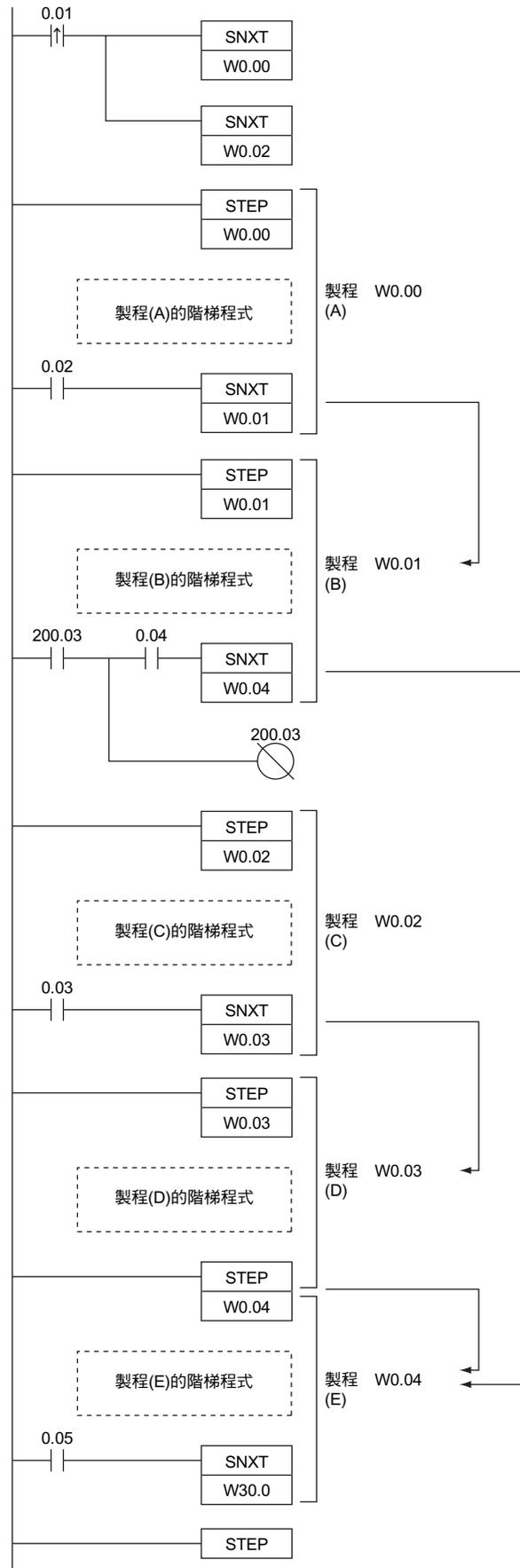
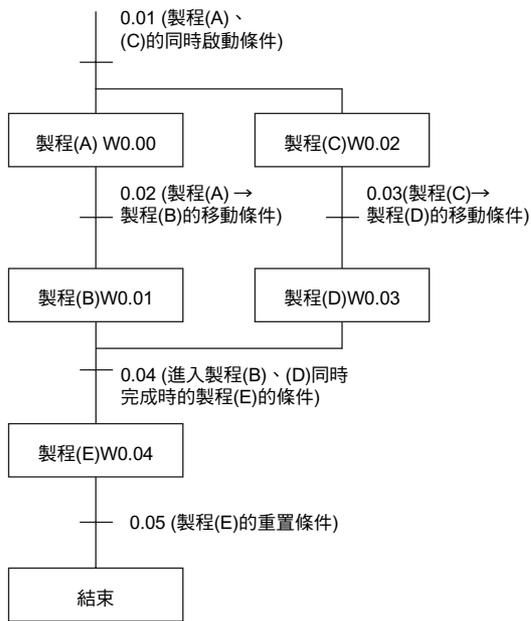
參考

- 如上述範例中 SNXT 的 W0.02 所示，利用選擇分歧來啟動下一個製程時，就會進入線圈重複使用狀態，不過，若利用 CX-Programmer 來執行程式檢查時，則不會出現重複使用錯誤。
- 只有將一般階梯回路中所使用的繼電器編號用在步進階梯指令時，才會造成步進階梯回路中發生重複使用錯誤。
- 上述程式不得同時執行製程 A、B 的動作。
- 若希望同時執行製程 A、B 的動作時，

請刪除

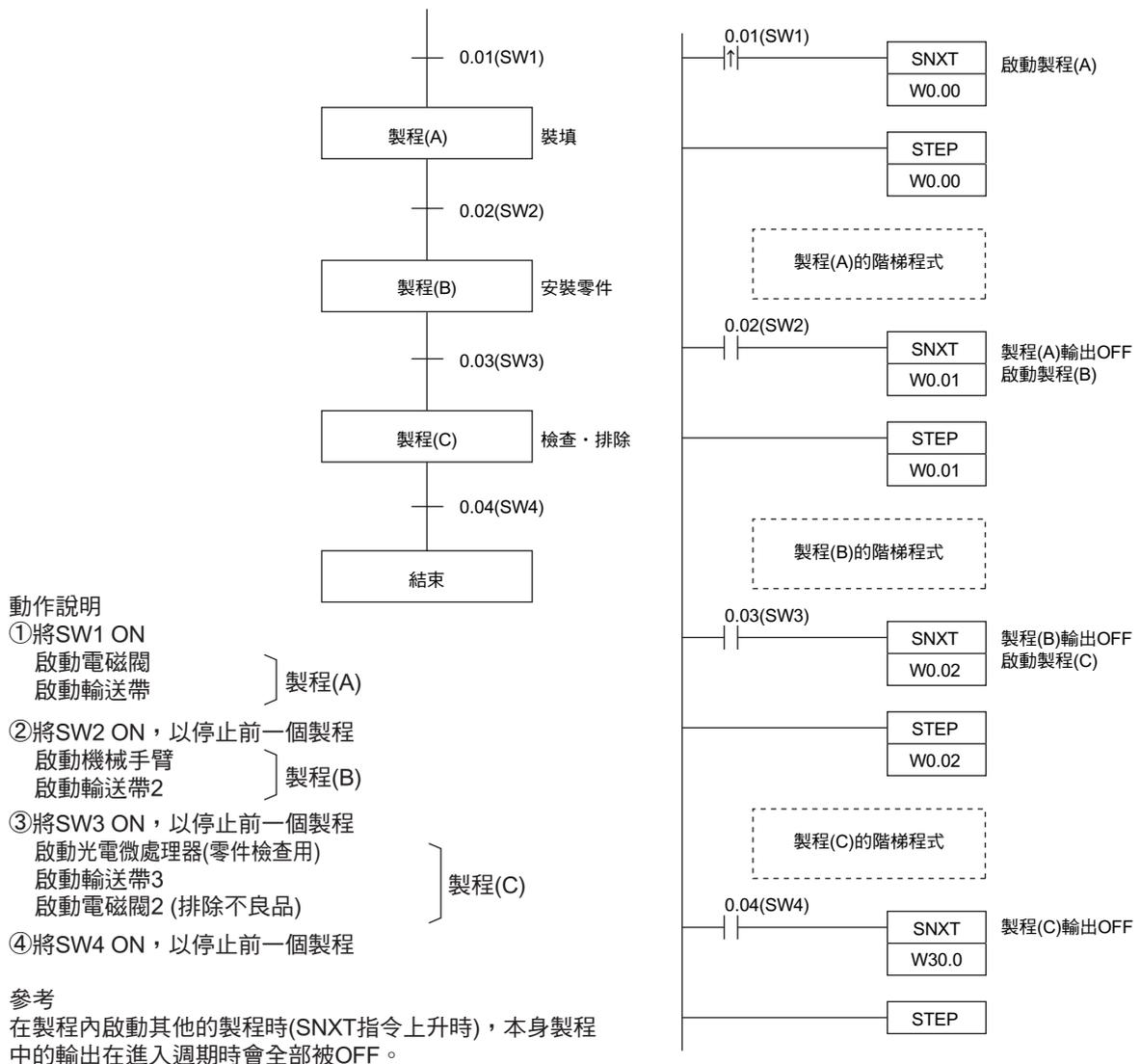
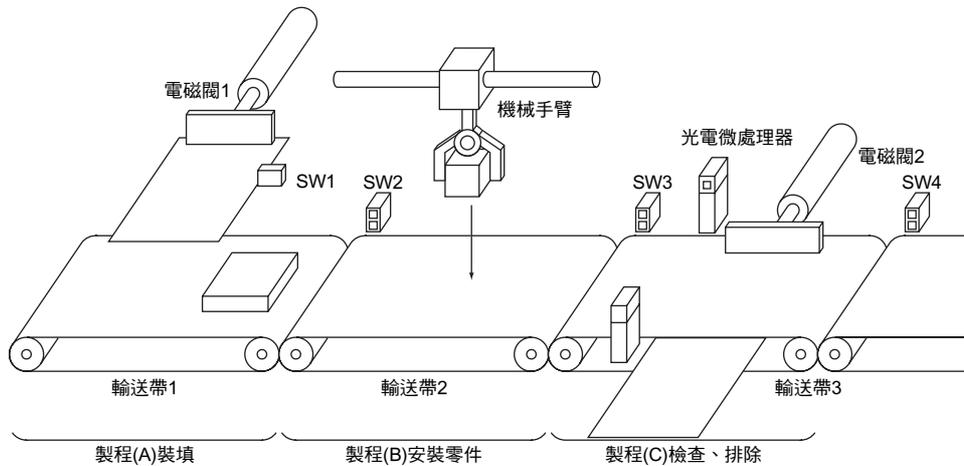


(3) 並聯分歧合流控制



## 應用範例

### (1) 順序控制



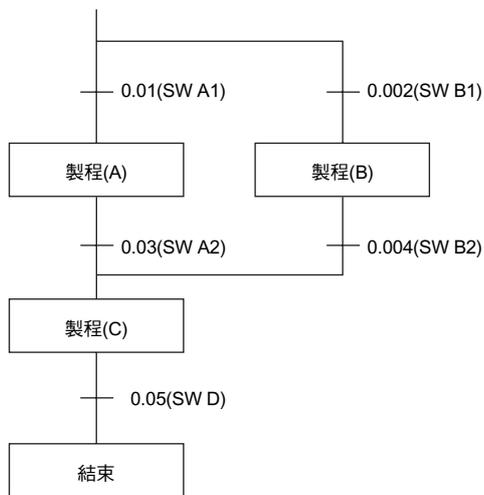
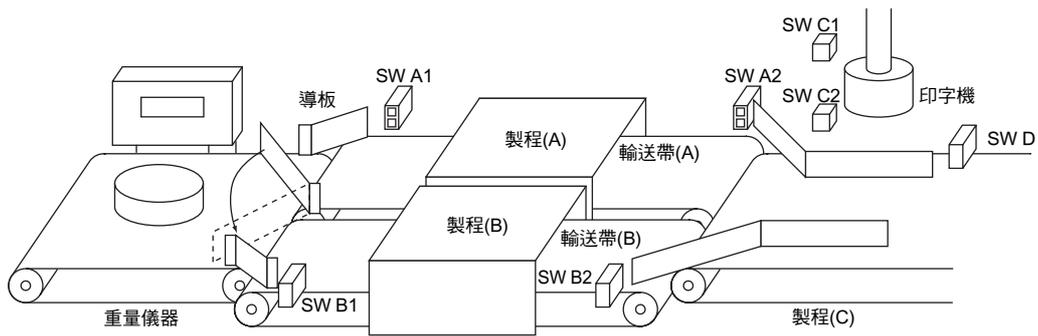
#### 動作說明

- ①將SW1 ON  
 啟動電磁閥  
 啟動傳送帶  
 } 製程(A)
- ②將SW2 ON，以停止前一個製程  
 啟動機械手臂  
 啟動傳送帶2  
 } 製程(B)
- ③將SW3 ON，以停止前一個製程  
 啟動光電微處理器(零件檢查用)  
 啟動傳送帶3  
 啟動電磁閥2(排除不良品)  
 } 製程(C)
- ④將SW4 ON，以停止前一個製程

#### 參考

在製程內啟動其他的製程時(SNXT指令上升時)，本身製程中的輸出在進入週期時會全部被OFF。

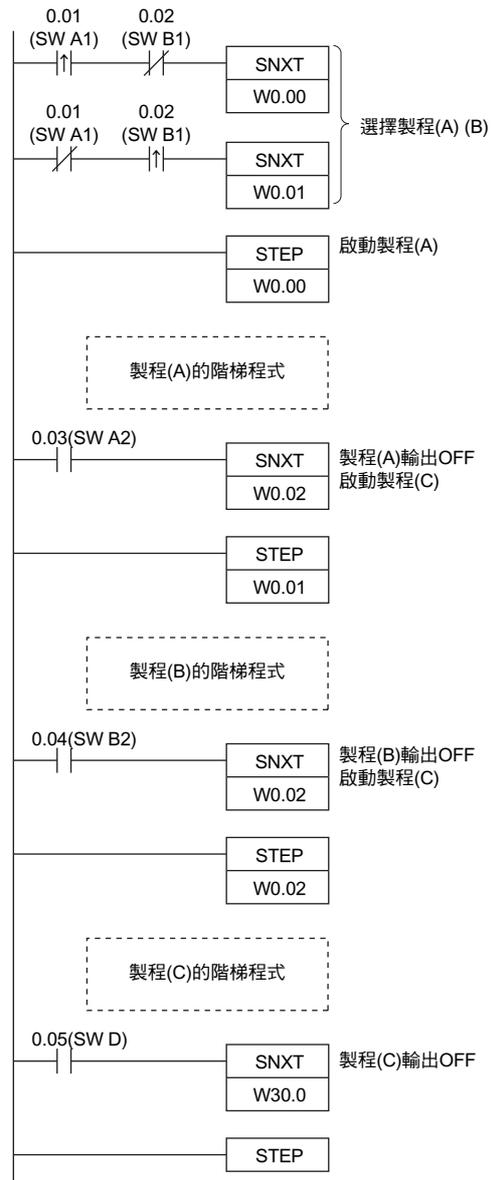
## (2) 選擇分歧控制



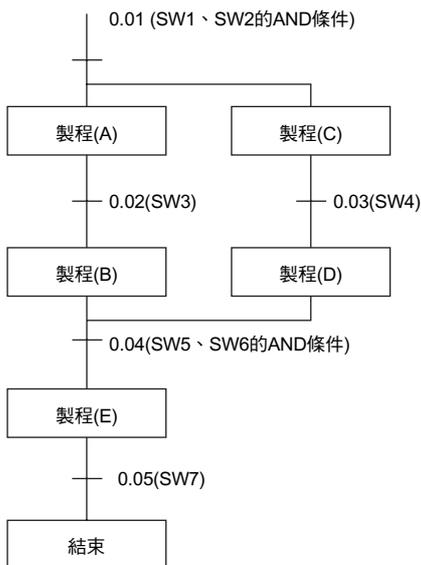
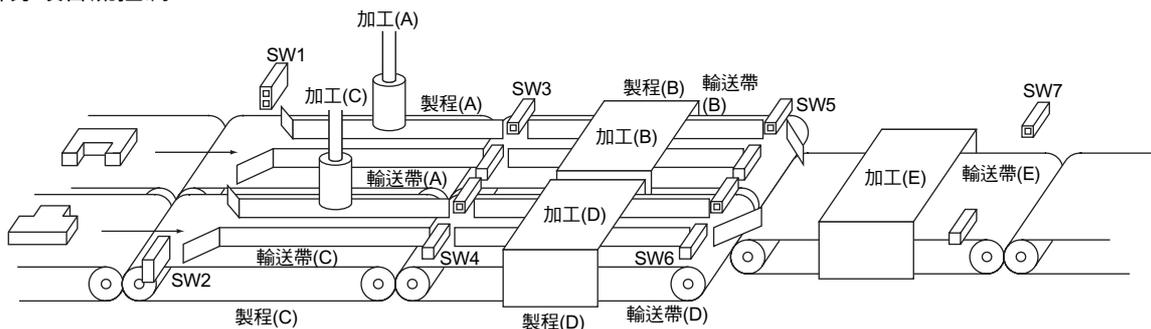
## 動作說明

依重量不同，以導板來分送產品。

- ① SWA1 ON  
即啟動輸送帶(A)  
啟動加工機器(A) } 製程(A)
- ② SWB1 ON  
即啟動輸送帶(B)  
啟動加工機器(B) } 製程(B)
- ③ SWA2 ON即停止製程A  
啟動印字機(down)  
SWC2 ON即UP } 製程(C)
- ④ SWB2 ON即停止製程B  
啟動印字機(down)  
SWC2 ON即UP } 製程(C)
- ⑤ SWD ON，即停止印字機動作



(3) 並聯分歧合流控制



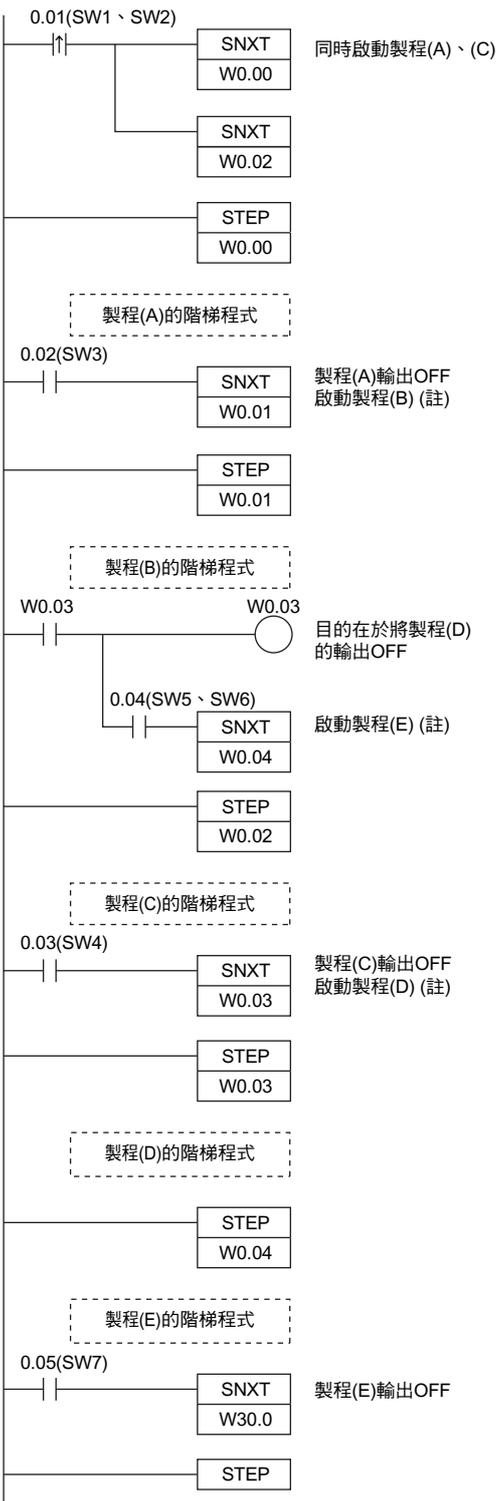
動作說明

- ①將SW1、SW2 ON  
即啟動輸送帶(A)  
啟動加工(A) } 製程(A)  
即啟動輸送帶(C)  
啟動加工(C) } 製程(C)
- ②將SW3 ON  
製程(A)停止  
即啟動輸送帶(B)  
啟動加工(B) } 製程(B)
- ③將SW4 ON  
製程(C)停止  
即啟動輸送帶(D)  
啟動加工(D) } 製程(D)
- ④將SW5、SW6 ON  
製程(B)停止  
製程(D)停止  
即啟動輸送帶(E)  
啟動加工(E) } 製程(E)
- ⑤將SW7 ON  
製程(E)停止

(註) 當製程(B)、製程(D)啟動且SW、SW變成ON時，就會判斷為結束製程(B)、(D)。執行SNXT W0.04，即可將製程(B)的輸出OFF，並且將W0.03 OFF。判斷為STEP W0.03由ON下降為OFF，並且將製程(D)的輸出OFF。

參考

當STEP指令由ON下降為OFF時，本身製程的所有輸出就會被OFF。



# I/O 模組專用指令

## IORF

指令名稱	指令記號	指令的各種組合	Fun No.	功能
I/O 更新狀態	IORF	@IORF	097	更新所指定的 I/OCH 資料。

符號	IORF	
		D1
	D2	D2：更新的上位CH編號

### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 運算元種類的說明

運算元種類	內容	資料型態	容量
D1	更新的下位 CH 編號	—	可調整
D2	更新的上位 CH 編號	—	可調整

D1：更新的下位 CH 編號

CIO 區域中 CP1W 擴充 I/O 模組的輸出輸入繼電器區 (001 ~ 099CH、101 ~ 199CH)

D2：更新的上位 CH 編號

CIO 區域中 CP1W 擴充 I/O 模組的輸出輸入繼電器區 (001 ~ 099CH、101 ~ 199CH)

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
D1,D2	○	—	—	—	—	—	—	—	—	—	—	—	—

### 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 D1 &gt; D2 時，本旗標 ON</li> <li>D1 和 D2 的位址位於不同區域時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

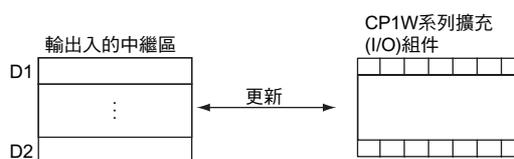
## IORF 指令可更新的模組

裝置位置	CH	可否更新 ○：可以 ×：不支援
30、40 點輸出輸入型	CP1E CPU 模組內建的輸出輸入：0、1、100、101CH	×
	CP1W 擴充 (I/O) 模組：2 ~ 99CH、102 ~ 199CH	○
60 點輸入輸出型	CP1E CPU 模組內建的輸出輸入：0、1、2、100、101、102CH	×
	CP1W 擴充 (I/O) 模組：3 ~ 99CH、103 ~ 199CH	○
NA20 點輸入輸出型	CP1E CPU 模組內建的輸出輸入：0、100CH	×
	CP1E CPU 模組內建的類比輸出輸入：90、91、190CH	○
	CP1W 擴充 (I/O) 模組：1 ~ 99CH、101 ~ 199CH	○

(註) CP1E CPU 模組內建的輸出輸入區無法利用 IORF 指令進行更新。  
請利用立即更新 (I) 的指定方式，為 CCP1E CPU 模組內建的輸出輸入區進行更新。

## 功能

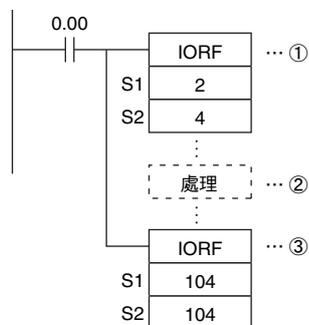
為 D1 所指定的更新下位 CH 編號 ~ D2 所指定的更新上位 CH 編號進行 I/OCH 資料的更新。  
更新對象為 CP1W 擴充 (I/O) 模組的輸出輸入繼電器區及 CP1E CPU 模組內建的類比區的對象範圍。



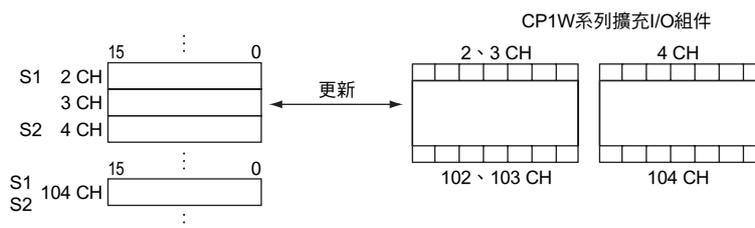
## 使用上的注意事項

- 本指令可在中斷任務中使用 ( 如此一來，即可利用中斷進行特定輸出輸入的高速回應處理 )。
- 在更新的對象範圍 (D1 ~ D2) 中，不對未安裝模組的 CH，進行任何處理，僅針對已安裝模組的模組進行更新處理。
- 利用 IORF 指令進行 I/O 更新處理時，一旦發生 I/O 匯流排異常，就會中斷 I/O 更新動作。

## 程式例



當輸出輸入繼電器區的更新0.00變成ON時，就利用①來更新2 ~ 4 CH共3 CH (36點)的輸入更新區，然後再利用②執行必要的處理，以及③更新104 CH共1 CH (8點)的輸出繼電器區。



# SDEC

指令名稱	指令記號	指令的各種組合	Fun No.	功能
7 段解碼器	SDEC	@SDEC	078	將 CH 資料的指定位數中各個 4 位元的內容 (0 ~ F) 轉換為 8 位元的 7 區段資料，然後再將結果輸出至指定 CH 以後的上位或下位的各 8 位元。

符號	SDEC	
		S : 轉換資料的CH編號 K : 指定位數的資料 D : 輸出轉換結果的下位CH編號

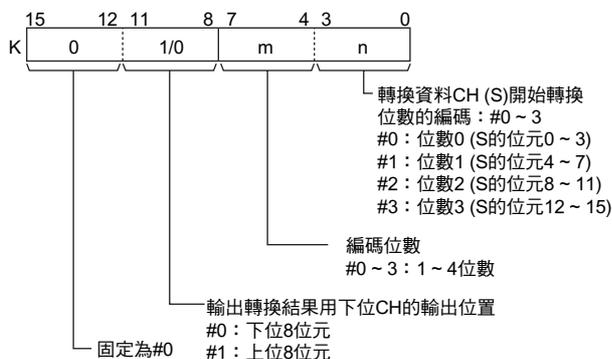
## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	轉換資料的 CH 編號	UINT	1
K	指定位數的資料	UINT	1
D	輸出轉換結果的下位 CH 編號	UINT	可調整

### K：指定位數的資料



### ● 運算元種類

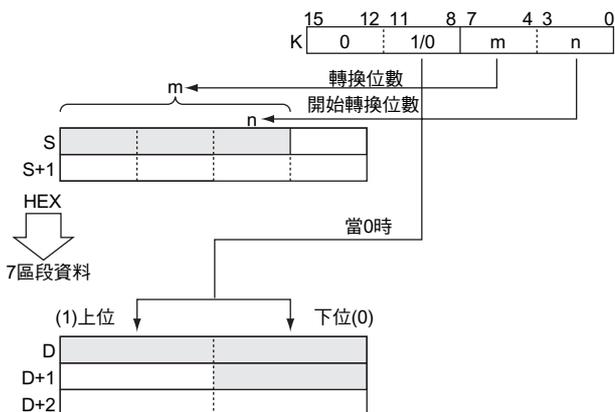
內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S										—			
K	○	○	○	○	○	○	○	○	○	○	—	—	—
D										—			

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 K 的資料超過範圍時變為 ON</li> <li>否則為 OFF</li> </ul>

### 功能

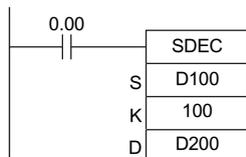
將 S 所指定的資料當作 4 位數的 16 進位資料 (#0000 ~ FFFF)，然後利用開始轉換的位數編號 (K)，將所指定的位數資料 (#0 ~ F) 轉換為 7 區段資料，並將結果輸出至 D 指定的輸出位置 (K)。



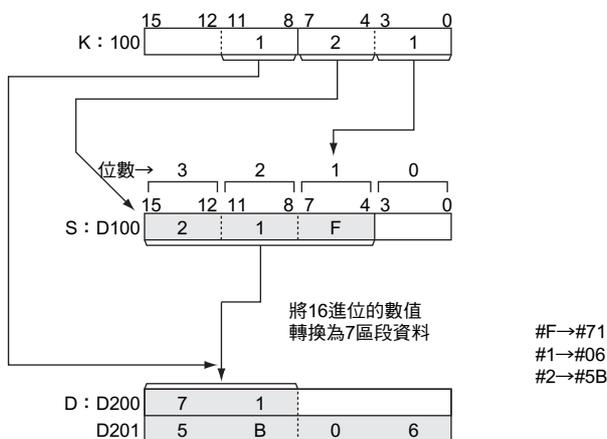
### 使用上？注意

- 利用轉換位數 (K) 指定為轉換多個位數時，轉換的對象位數將變成由開始位數朝上位位數端的順序 (位數 3 結束後就會回到位數 0)。
- 儲存轉換結果時，將依 D 的輸出位置朝上位 CH 端 (以 8 位元為單位) 的順序加以儲存。在輸出轉換結果的 CH 資料中，非輸出對象的位置資料將不會發生任何改變。

### 程式例



當 0.00 變成 ON 時，D100 的 1CH 就會被當作 4 位數的 16 進位 (#0 ~ F) 資料，並且由 100CH (位數指定資料) 的位元 0 ~ 3 所指定的位數位置 (例：位數 1) 開始將位元 4 ~ 7 所指定的位數轉換為 7 區段資料，然後再由 D200 所指定的位元位置 (利用 100CH 的位元 8 ~ 11 指定) 開始儲存。



### ● 轉換資料的內容和解碼輸出之間的關係

當位數 (4 位元) 的內容 0 ~ F 被轉換為 8 位元的 7 區段資料時，其結果如下表所示：

原始轉換資料		轉換結果資料 (8 位元)						7 段顯示	
數值	位元內容	g	f	e	d	c	b	a	16 進位
0	0 0 0 0	0	0	0	0	0	0	1 1 1 1	3F
1	0 0 0 1	0	0	0	0	0	0	1 1 1 0	06
2	0 0 1 0	0	1	0	0	1	1	0 1 1 1	5B
3	0 0 1 1	0	1	0	1	0	1	1 1 1 1	4F
4	0 1 0 0	0	1	1	0	0	1	1 1 1 0	66
5	0 1 0 1	0	1	1	0	1	1	0 1 1 1	6D
6	0 1 1 0	0	1	1	1	1	1	0 1 1 1	7D
7	0 1 1 1	0	1	1	1	0	1	1 1 1 1	27
8	1 0 0 0	0	1	1	1	1	1	1 1 1 1	7F
9	1 0 0 1	0	1	1	1	0	1	1 1 1 1	6F
A	1 0 1 0	0	1	1	0	1	1	1 1 1 1	77
B	1 0 1 1	0	1	1	1	1	1	0 1 1 1	7C
C	1 1 0 0	0	0	1	1	1	0	0 1 1 1	39
D	1 1 0 1	0	0	1	1	0	1	1 1 1 0	5E
E	1 1 1 0	0	1	1	1	1	0	0 1 1 1	79
F	1 1 1 1	0	1	1	1	0	0	1 1 1 1	71

LSB

a

b

c

d

e

f

g

0

MSB

# DSW

指令名稱	指令記號	指令的各種組合	Fun No.	功能
數位開關	DSW	—	210	讀取連接至輸出輸入的外部數位開關 (或指撥旋鈕開關) 的設定值，然後再以 8 位數或 4 位數數值 (BCD) 的形式，儲存在指定的 CH 以後的位置。

符號	DSW	
		DSW
	I	I：輸入資料線(D0~3)的CH編號
	O	O：輸出控制資料(CS/RD)的CH編號
	D	D：儲存資料的啟始CH編號
	C1	C1：指定位數的啟始CH編號
	C2	C2：工作區的啟始CH編號

## 是否適用於特定區域

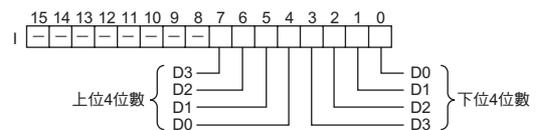
區域	工程步進程式	副程式	中斷任務程式
使用	○	○	—

## 運算元種類的說明

運算元種類	內容	資料型態	容量
I	輸入資料線 (D0 ~ 3) 的 CH 編號	UINT	1
O	輸出控制資料 (CS/RD) 的 CH 編號	UINT	1
D	儲存資料的啟始 CH 編號	WORD	可調整
C1	指定位數	UINT	1
C2	工作區的啟始 CH 編號	WORD	1

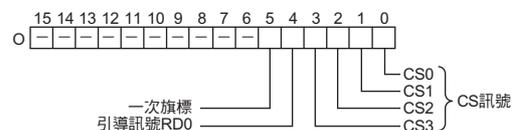
### I：輸入資料線 (D0 ~ 3) 的 CH 編號

通常會指定為輸入繼電器的輸入模組配置 CH，並且將數位開關的資料線 D0 ~ D3 連接至該輸入模組。



### O：輸出控制資料 (CS/RD) 的 CH 編號

通常會指定為輸出繼電器的輸出模組配置 CH，並且將數位開關的控制訊號 (CS 訊號即 RD 訊號) 連接至該輸出模組。



### D：儲存資料的啟始 CH 編號

指定用來儲存外部數位開關設定值的啟始 CH 編號。



(註) C1 = &1 (16進位#0001)：僅限讀取位數為8位數時

C1：指定位數

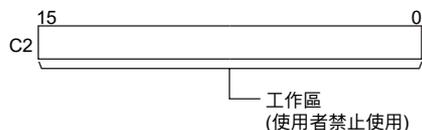
指定由外部數位開關所讀取的位數

#0000：讀取 4 位數

#0001：讀取 8 位數

C2：工作區的啟始 CH 編號

指定為工作區，本指令將此一 CH 用於系統用途，禁止當作其他用途使用。



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
I,O,D	○	○	○	○	○	○	○	○	○	—	—	—	—
C1	—	—	—	—	—	—	—	—	—	○			
C2	○	○	○	○	○	○	○	○	—	—			

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF

功能

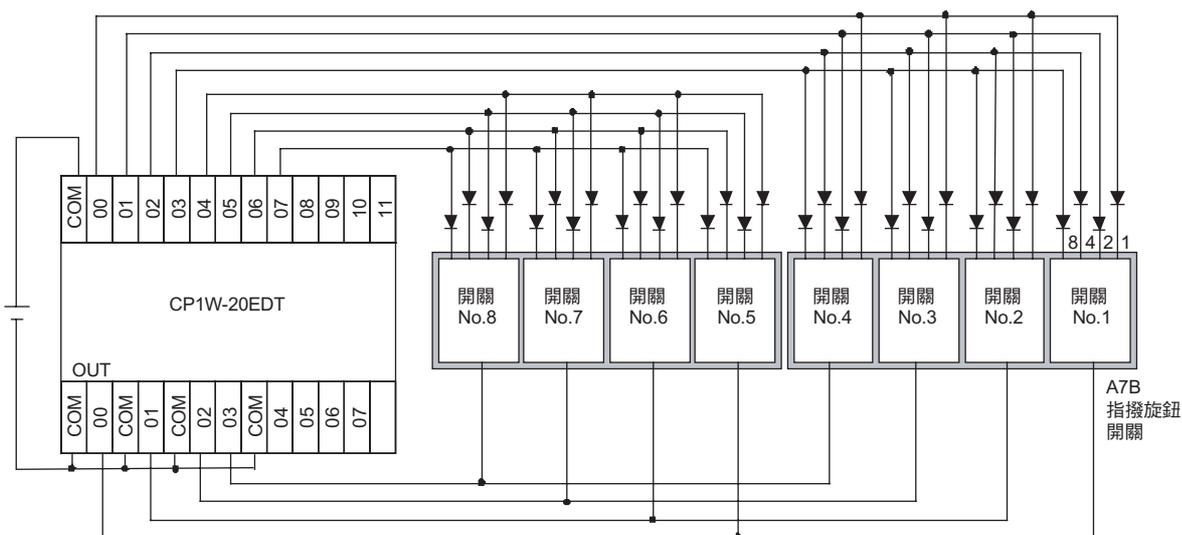
將控制訊號輸出至 O 所指定的 CH 位元 00 ~ 04，然後再由 I 所指定 CH 的外部數關的資料線 ( 下位 4 位數 D0 ~ 3、上位 4 位數 D0 ~ 3) 開始讀取 C1 所指定的位數 (8 位數或 4 位數)，並將結果儲存於 D 以後的位置 ( 當指定位數為 4 位數時，就將結果儲存至 DCH，如果是 8 位數時，則儲存於 D+1CH)。本指令會以 16 週期 1 次的頻率，由 CPU 模組讀取外部數位開關 ( 或指撥旋鈕開關 ) 的 4 位數或 8 位數設定值。

又，如果每隔 16 週期就會出現 1 次控制訊號，那麼一次控制旗標 (OCH 位元 05) 就會執行 ON 動作 1 週期。

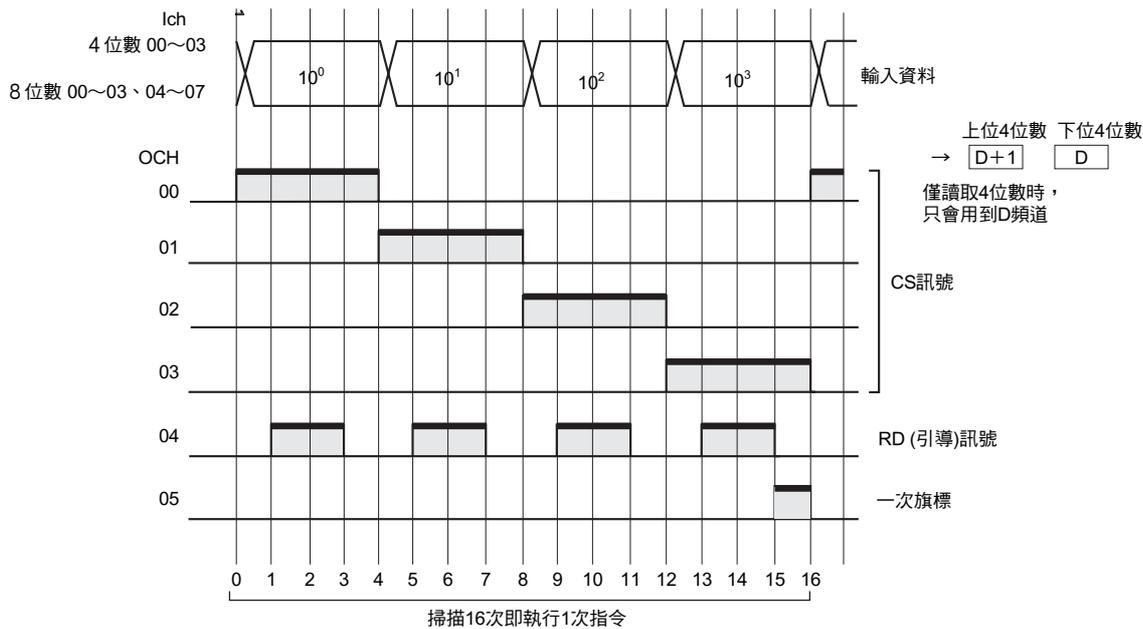
● 與外部的連接方法

請依下圖所示，將外部的數位開關或指撥旋鈕開關連接至輸入模組的接點 0 ~ 7，以及輸出模組的接點 0 ~ 4。

例) 指撥旋鈕開關



## ● 時序圖

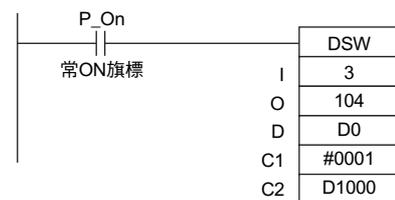


## 使用上的注意事項

- 請勿利用其他指令來讀寫 C2 的工作區 1CH。  
以其他指令進行讀寫時，將造成本指令無法正常動作，此點需特別注意。  
如果是開始運轉的第 1 週期，本指令將不會在執行時將工作區的 1CH 初始化。因此，如果希望從開始運轉的第 1 週期起就執行本指令時，請利用程式來清除工作區。
- 執行 1 次指令後，如果未和連接外部數位開關(或是指撥旋鈕開關)的輸入模組/輸出模組進行 I/O 更新，將導致系統無法正常動作。  
因此，當您在設定數位資料線輸入 CH 編號所對應的輸入模組輸入時間常數時，其數值必須小於指撥旋鈕開關。
- 本指令會在 16 週期內讀取 4 位數或 8 位數的資料，接著，將再次回到啟始位置讀取資料。
- 當本指令開始執行時(無論指令停止執行時的狀態為何)，將隨時由第一個週期開始讀取資料。

## 程式例

由連接至 3CH 和 104CH 的數位開關隨時讀取 8 位數數值，然後再將結果儲存至 D0 ~ D3。  
D1000 為工作區。



# MTR

指令名稱	指令記號	指令的各種組合	Fun No.	功能
矩陣輸入	MTR	—	213	由連接至輸入模組及輸出模組的外部 8 點 x 8 列接點 (矩陣) 依序讀取 64 點，然後再以 64 點 (4CH) 資料的格式，儲存在指定的 CH 以後的位置。

符號	MTR						
		<table border="1"> <tr><td>MTR</td></tr> <tr><td>I</td></tr> <tr><td>O</td></tr> <tr><td>D</td></tr> <tr><td>C</td></tr> </table>	MTR	I	O	D	C
MTR							
I							
O							
D							
C							

## 是否適用於特定區域

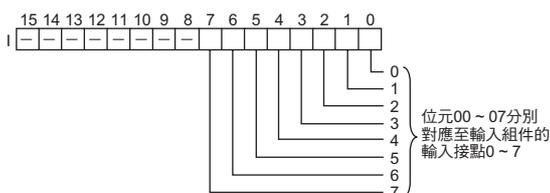
區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
I	輸入資料的 CH 編號	UINT	1
O	輸出資料選擇訊號的 CH 編號	UINT	1
D	儲存輸入資料的啟始 CH 編號	ULINT	4
C	工作區的啟始 CH 編號	WORD	1

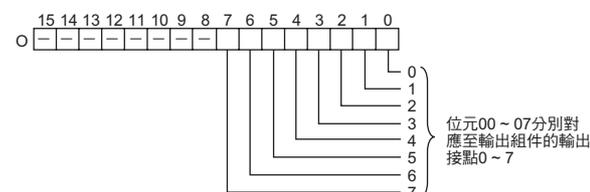
### I：輸入資料的 CH 編號

通常會指定為輸入繼電器的輸入模組配置 CH，並且將 8 點輸入訊號連接至該輸入模組。



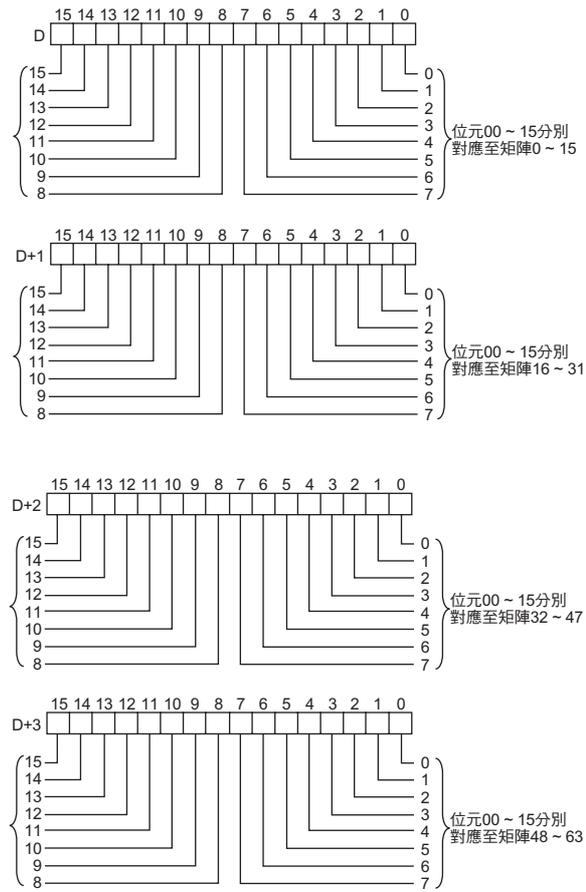
### O：輸出資料選擇訊號的 CH 編號

通常會指定為輸出繼電器的輸出模組配置 CH，並且將 8 點資料選擇訊號連接至該輸出模組。



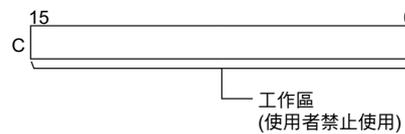
**D：**儲存輸入資料的啟始 CH 編號

指定用來儲存以 8 點 x 8 列接點 (點陣) 為起始的 16 點資料 (4CH) 的啟始 CH 編號。



**C：**工作區的啟始 CH 編號

指定可用來當作工作區使用的 CH 編號。本指令將此一 CH 用於系統用途，禁止當作其他用途使用。



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
I.O.D.C	○	○	○	○	○	○	○	○	○	-	-	-	-

條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	OFF

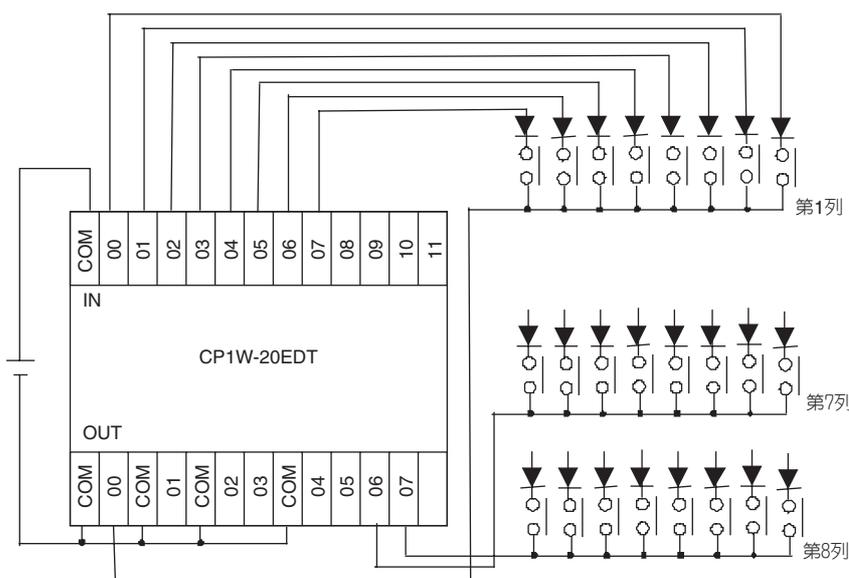
功能

將選擇訊號輸出至 O 所指定的 CH 位元 00 ~ 07，然後再由 I 所指定 CH 的位元 00 ~ 07 開始依序讀取資料，並將結果以 16 點資料 (4CH) 的格式，儲存在 D 以後的位置。本指令會以 24 週期 1 次的頻率，由 CPU 模組讀取 1 個矩陣狀態。

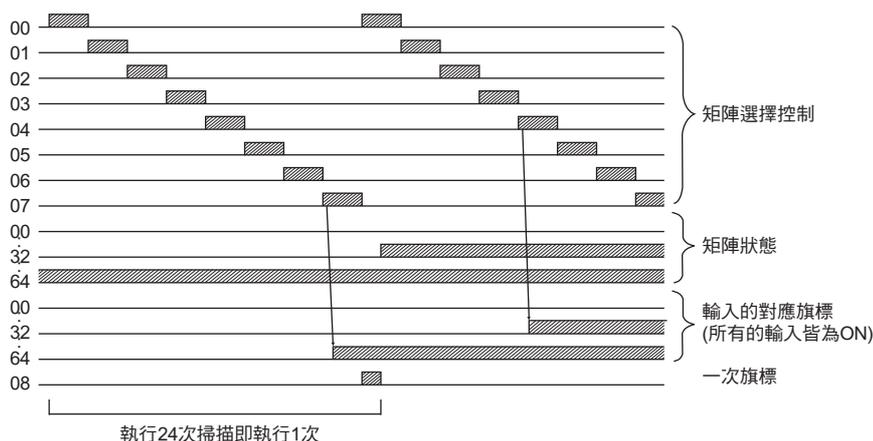
又，每隔 24 週期出現 1 次資料選擇用訊號時，按鍵掃描一次旗標 (OCH 位元 08) 就會持續 ON 動作 1 週期。

● 與外部的連接方法

請參右圖所示，將外部 8 點 x 8 列的矩陣連接至輸入模組的接點 0~7，以及輸出模組的接點 0~7。



● 時序圖



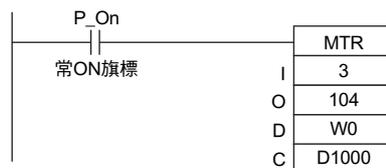
使用上的注意事項

- 請勿使用其他指令來讀寫 C 所指定的工作區的 1CH。以其他指令進行讀寫時，將造成本指令無法正常動作，此點需特別注意。
- 如果是開始運轉的第 1 週期，本指令將不會在執行時將工作區的 1CH 初始化。因此，如果希望從開始運轉的第 1 週期起就執行本指令時，請利用程式來清除工作區。
- 執行 1 次指令後，如果未和連接外部矩陣的輸入模組 / 輸出模組進行 I/O 更新，將導致系統無法正常動作。
- 因此，當您在設定數位資料輸入 CH 編號所對應的輸入模組輸入時間常數時，其數值必須小於指撥旋鈕開關。
- 開始執行本指令時 (無論停止執行指令時的狀態為何)，就會隨時由第一個週期開始讀取資料。

程式例

隨時由連接至 3CH 和 104CH 的 8 點 x 8 烈的接點 (矩陣) 開始讀取 64 點資料，然後再將結果儲存至 W0 ~ W3CH。

又，右圖所示為 D (工作區) 被當作 D1000 時的範例。



# 7SEG

指令名稱	指令記號	指令的各種組合	Fun No.	功能
7 段顯示	7SEG	—	214	將 4 位數或 8 位數的數值 (BCD 資料) 轉換為 7 段顯示器專用的資料，然後再輸出至指定 CH 以後的位置。

符號	7SEG										
		<table border="1"> <tr> <td>7SEG</td> <td></td> </tr> <tr> <td>S</td> <td>S：儲存顯示資料的啟始CH編號</td> </tr> <tr> <td>O</td> <td>O：儲存輸出資料/輸出栓鎖的CH編號</td> </tr> <tr> <td>C</td> <td>C：顯示位數・輸出邏輯的選擇資料</td> </tr> <tr> <td>D</td> <td>D：工作區的啟始CH編號</td> </tr> </table>	7SEG		S	S：儲存顯示資料的啟始CH編號	O	O：儲存輸出資料/輸出栓鎖的CH編號	C	C：顯示位數・輸出邏輯的選擇資料	D
7SEG											
S	S：儲存顯示資料的啟始CH編號										
O	O：儲存輸出資料/輸出栓鎖的CH編號										
C	C：顯示位數・輸出邏輯的選擇資料										
D	D：工作區的啟始CH編號										

## 是否適用於特定區域

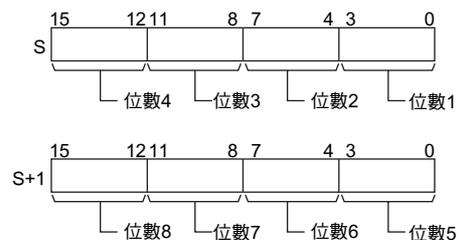
區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	儲存顯示資料的啟始 CH 編號	WORD	可調整
O	儲存輸出資料 / 輸出栓鎖的 CH 編號	UINT	1
C	顯示位數・輸出邏輯的選擇資料	僅限於 #+10 進位資料	1
D	工作區的啟始 CH 編號	WORD	1

### S：儲存顯示資料的啟始 CH 編號

指定可用來儲存 7 段顯示器專用資料的啟始 CH 編號。

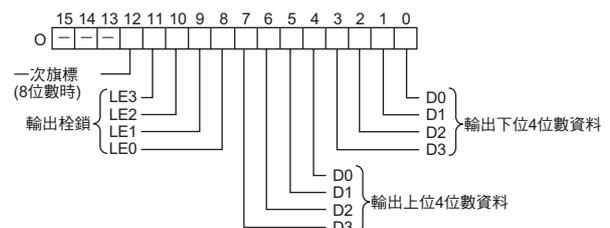
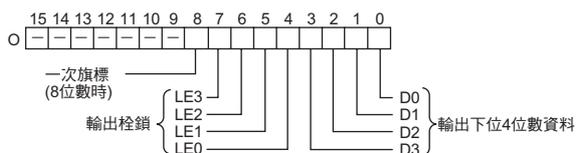


### O：儲存輸出資料 / 輸出栓鎖的 CH 編號

通常會指定為輸出繼電器的輸出模組配置 CH，並且將 7 段顯示器連接至該輸出模組。

指定為 4 位數時：

指定為 8 位數時：



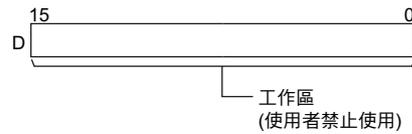
**C：顯示位數・輸出邏輯的選擇資料**

選擇顯示位數為 4 位數或 8 位數，以及 7 段顯示器的資料輸出端和使用輸出模組端之間的邏輯（輸出電晶體的負邏輯 (NPN)/ 正邏輯 (PNP) 是否相同，請指定為右列 #0000 ~ 0007 的其中任一項。

顯示位數	7 段顯示器的資料輸入和輸出模組的邏輯	7 段顯示器的栓鎖輸入和輸出模組的邏輯	C (顯示位數・輸出邏輯的選擇資料)
4 位數 (1 組 4 位數)	相同	相同	#0000
		相異	#0001
	相異	相同	#0002
		相異	#0003
8 位數 (2 組 4 位數)	相同	相同	#0004
		相異	#0005
	相異	相同	#0006
		相異	#0007

**D：工作區的啟始 CH 編號**

指定可用來當作工作區使用的 CH 編號。本指令將此一 CH 用於系統用途，禁止當作其他用途使用。



● **運算元種類**

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S.O	○	○	○	○	○	○	○	○	○	—	—	—	—
C	—	—	—	—	—	—	—	—	—	○			
D	○	○	○	○	○	○	○	○	○	—			

**條件旗標的動作**

名稱	標籤	內容
異常旗標	P_ER	OFF

**功能**

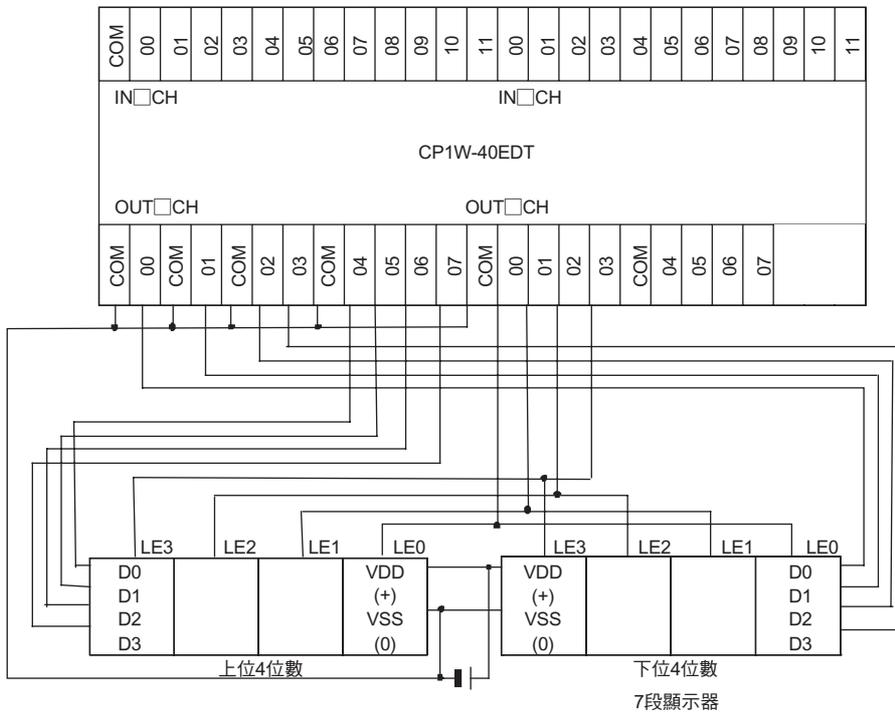
根據 C 所指定的顯示位數及輸出邏輯，將 S 所指定的 4 位數 (SCH) 或 8 位數 (S、S+1CH) 的 BCD 資料轉換為 7 段顯示器專用的資料 (下位 4 位數 D0 ~ 3、上位 4 位數 D0 ~ 3、栓鎖輸出訊號 LE0 ~ 3)，然後再將結果輸出至 O。

本指令會以 12 週期 1 次的頻率，由 CPU 模組輸出 7 段顯示器專用的資料。

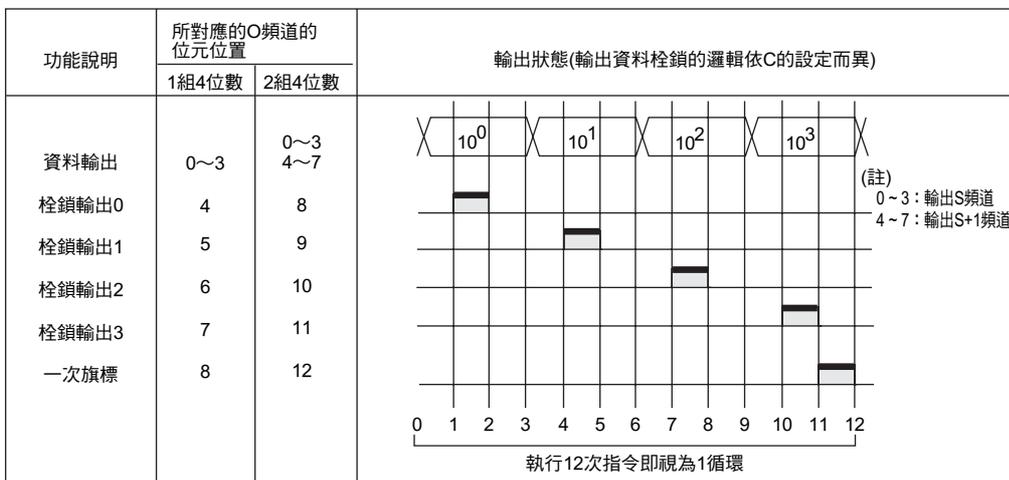
又，每隔 12 週期出現 1 次栓鎖輸出訊號時，一次旗標 (指定為 4 位數時：OCH 位元 08，指定為 8 位數時：OCH 位元 12) 將持續 ON 動作 1 週期。

### ● 與外部的連接方法

請依下圖所示，將 7 段顯示器連接至輸出模組。此圖為 8 位數顯示時之範例，執行位數顯示時，資料輸出 (D0 ~ D3) 會被連接至輸出接點 0 ~ 3，並且將栓鎖輸出 (LE0 ~ LE3) 連接至輸出接點 4 ~ 7。指定為 8 位數顯示時，當資料顯示 1 次，就會將 ON 輸出傳送到輸出接點 12，若指定為 4 位數顯示時，則會將 ON 輸出傳送至輸出接點 8。



### ● 時序圖



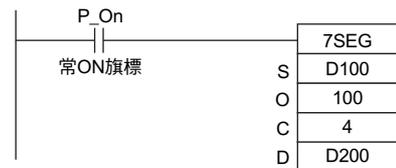
## 使用上的注意事項

- 請勿使用其他指令來讀寫 D 所指定的工作區的 1CH。
  - 以其他指令進行讀寫時，將造成本指令無法正常動作，此點需特別注意。
  - 如果是開始運轉的第 1 週期，本指令將不會在執行時將工作區的 1CH 初始化。因此，如果希望從開始運轉的第 1 週期起就執行本指令時，請利用程式來清除工作區。
- 本指令會在 12 週期內輸出 4 位數或 8 位數的資料。接著，會再次回到啟始位置，進行資料輸出。
- 開始執行本指令時 ( 無論停止執行指令時的狀態為何 )，就會隨時由第一個週期開始輸出資料。
- 即使所連接的 7 段顯示器位數小於 4 位數或 8 位數，系統仍然會輸出相當於 4 位數或 8 位數的資料。

## 程式例

讓 D100 中的 BCD8 位數數值隨時顯示在連接至 100CH 的 7 段顯示器上。

8 位數 (2 組 4 位數)、7 段顯示器的資料輸入和輸出模組的邏輯相同，而且 7 段顯示器的栓鎖輸入和輸出模組的邏輯相同，因此會變為 C (顯示位數 · 輸出邏輯的選擇資料) = #0004。又，右圖所示為 D (工作區) 被當作 D200 時的範例。



# 序列通訊指令

## TXD

指令名稱	指令記號	指令的各種組合	Fun No.	功能
CPU 模組內建的 RS232C 通訊埠、序列選擇埠輸出	TXD	@TXD	236	在不轉換且保持指定位元原始資料的條件下，根據 PLC 系統設定，在無程序模式下指定為啟動碼 / 結束碼，然後再由 CPU 模組內建的 RS-232C 通訊埠或序列選擇埠來傳送資料。

符號	TXD	
		<p>S：傳送資料的啟始CH編號</p> <p>C：控制資料</p> <p>N：傳送位元數 10進位&amp;0 ~ 256 (16進位#0000 ~ 0100)</p>

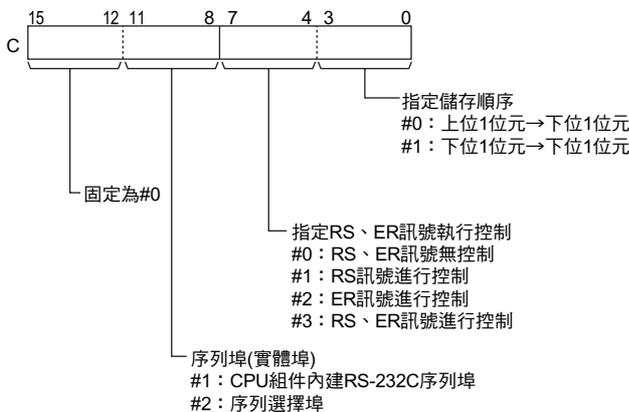
### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 運算元種類的說明

運算元種類	內容	資料型態	容量
S	傳送資料的啟始 CH 編號	UINT	可調整
C	控制資料	UINT	1
N	傳送位元數 10 進位 &0 ~ &256 (16 進位 #0000 ~ 0100)	UINT	1

#### C：控制資料



#### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S	○	○	○	○	○	○	○	○	○	—	—	—	—
C,N	○	○	○	○	○	○	○	○	○	○	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P.ER	<ul style="list-style-type: none"> <li>當 PLC 系統設定為無程式模式以外的模式時，本旗標就會 ON</li> <li>當 C 的資料超過範圍時變為 ON</li> <li>當 N 的資料以 10 進位表示時超過 0 ~ 256 或是 16 進位表示時超過 #0000 ~ 0100 的範圍內時，就會變成 ON</li> <li>當傳送準備完成旗標 (CPU 模組內建的連接埠：A392.05、序列選擇埠：A392.13) 變成 OFF，並且嘗試傳送時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 相關特殊輔助繼電器

- CPU 模組內建 RS-232C 序列埠時

名稱	標籤	內容
RS-232 序列埠準備傳送旗標	A392.05	可在無程序模式下執行傳送時，就會變成 1 (ON)。

- 使用序列選擇埠時

名稱	標籤	內容
序列選擇埠傳送準備完成旗標	A392.13	可在無程序模式下執行傳送時，就會變成 1 (ON)。

## PLC 系統相關設定



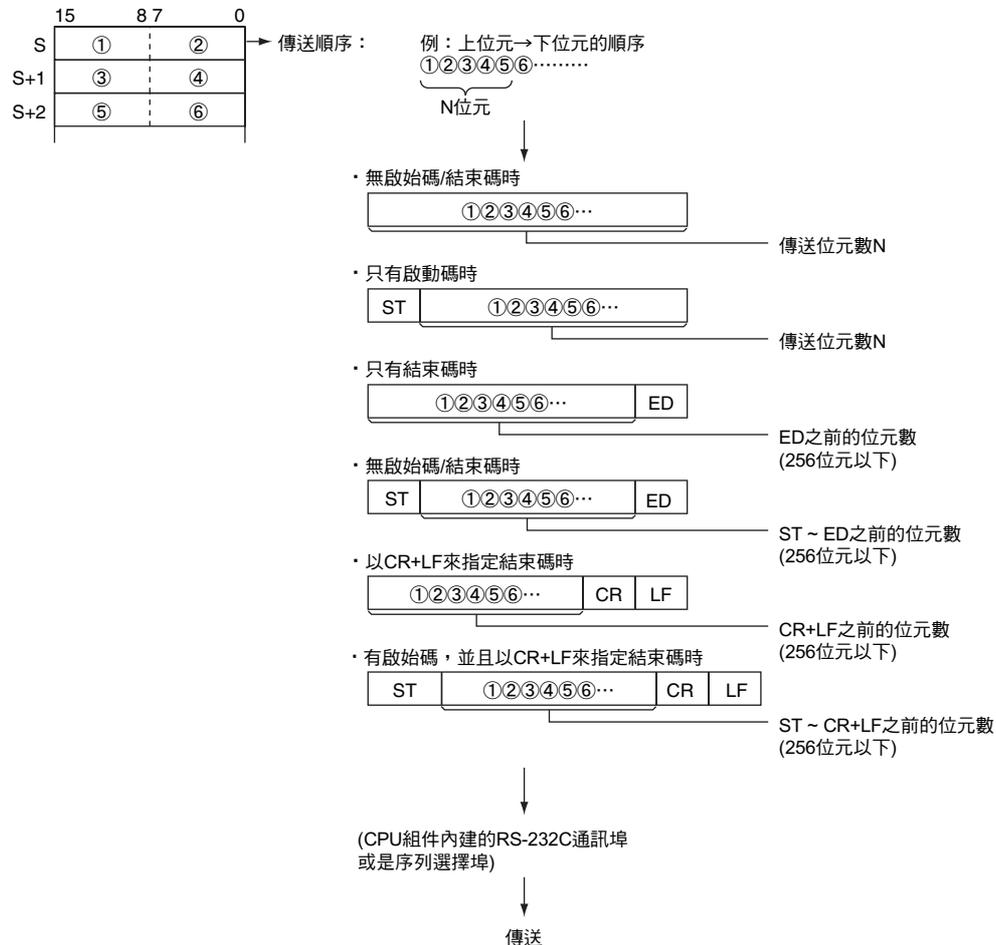
## 功能

- 在不轉換 S 所指定的傳送資料啟始 CH 編號 ~ N 所指定的傳送位元數資料的條件下，將資料輸出至 C 的位元 8 ~ 11 所指定的 CPU 模組內建的 RS-232 通訊埠 (無程序模式) 或是序列選擇埠的序列埠 (無程序模式)。
- 利用 PLC 系統設定的方式，將傳送訊息的訊框格式設定如下，根據本設定指定為啟始碼及結束碼時，將會附加所指定的啟始及結束碼後再傳送出去。
  - 是否已經設定啟始碼 (如果答案為「是」，啟始碼即為 #00 ~ FF)
  - 是否設定結束碼 (如果答案為「是」，啟始碼即為 #00 ~ FF 或 CR + LF)
- 根據 C 的位元 0 ~ 3 所指定的儲存順序來傳送資料。
- 利用 C 的位元 4 ~ 7 所指定的 RS、ER 訊號的控制指令將會被反映為下列方式。
  - 指定 RS 訊號的控制後，S 的位元 15 的內容就會被反映至 RS。
  - 指定 ER 訊號的控制後，S 的位元 15 的內容就會被反映至 ER。
  - 指定為控制 RS 訊號及 ER 訊號時，S 的位元 15 內容將會被反映至 RS 訊號，而 S 的位元 14 內容則會被反映至 ER 訊號。

當 RS、ER 訊號的控制指定被設定為 1 ~ 3Hex 時，無論傳送準備完成旗標 (CPU 模組內建通訊埠：A392.05、序列選擇埠：A392.13) 的狀態為何，皆會執行指令。

- 可傳送的位元數最多為 259 位元 (傳送資料區：最多為 256 位元，啟始碼：1 位元、結束碼區：CR+LF 所指定的 2 位元)。
- 請將不含啟始碼及結束碼的傳送資料區大小，指定至 N。

### ● 啟始碼 / 結束碼設定及傳送資料



### 提示

- 當某些對象裝置以 TXD 指令來傳送資料時，有可能需要保留傳送間隔。此時，請設定傳送延遲時間，並調整傳送間隔。

### 使用上的注意事項

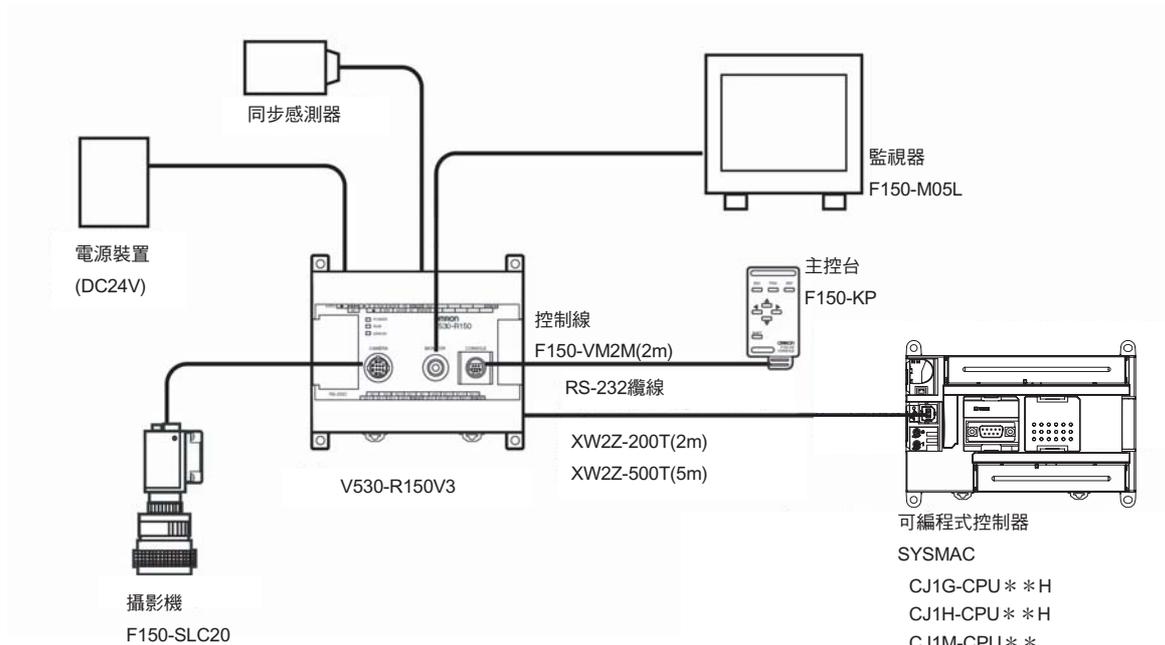
- 本指令僅適用於在無程序模式下所設定的 CPU 模組內建的 RS-232C 通訊埠或是序列選擇埠。
- 只有當特殊輔助繼電器的傳送準備完成旗標 (CPU 模組內建的連接埠：A392.05、序列選擇埠：A392.13) ON 時，才能進行傳送。
- 當 N 被指定為 0 時，將無法執行傳送動作。

## 程式例

### ● 傳送資料時

利用和通用外埠裝置之間的通訊為例，將資料傳送到 2 維讀碼機 (V530-R150V3 型)。

#### ● 硬體架構



本範例係連接至 CPU 模組內建的 RS-232 通訊埠。  
請在一開始就先設定好讀碼機的讀取條件設定。

#### ● 通訊設定

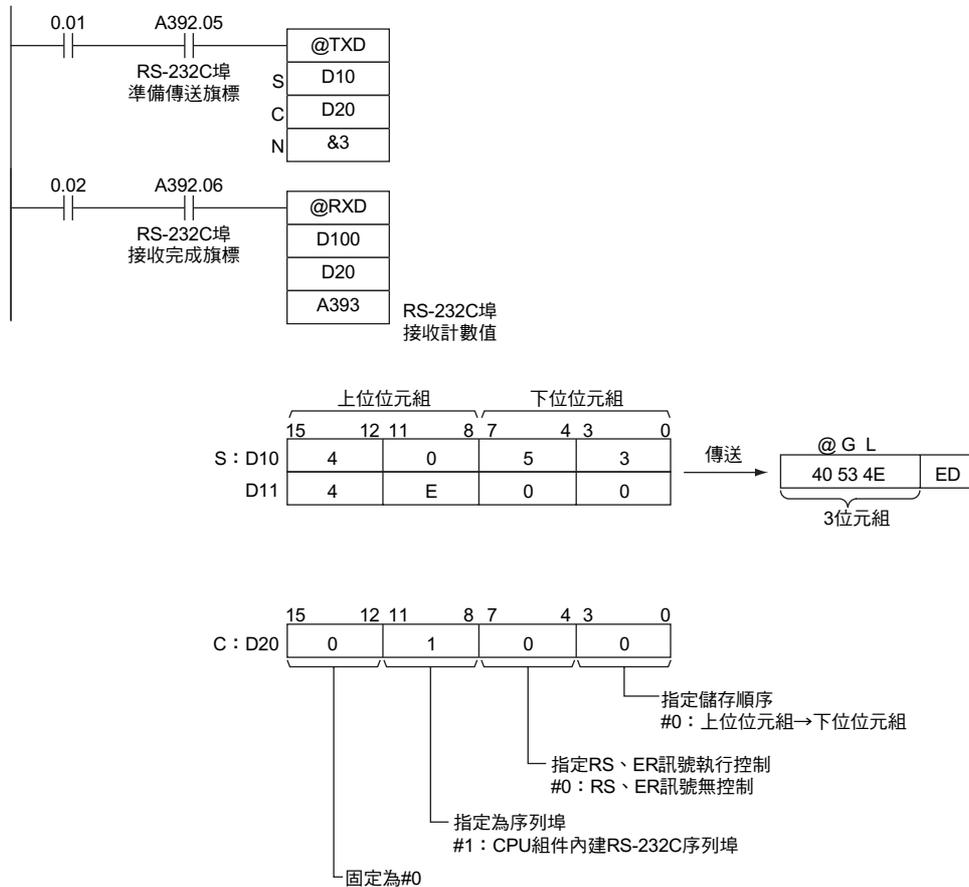
設定讀碼機時，請依照下列出廠時的初始值來設定。

通訊模式	無程序模式
通訊速度	38400bps
資料位元長度	8 位元
同位元檢查	無
停止位元	1 位元
啟始碼	無
結束碼	#000D (CR)

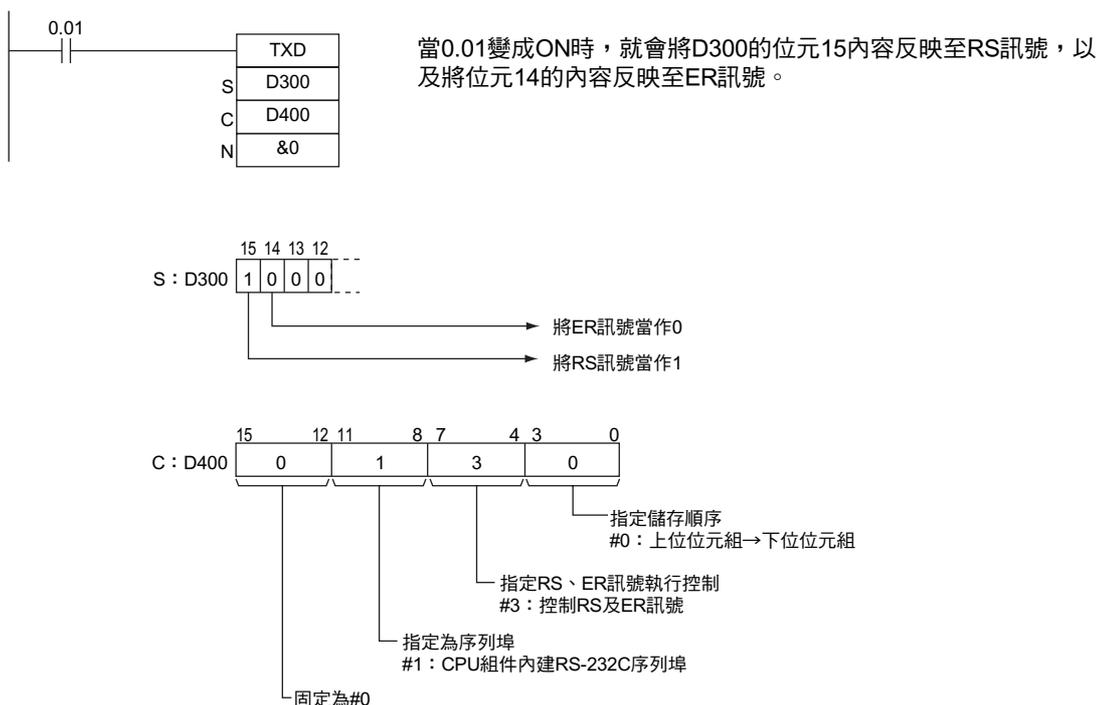
請利用 PLC 系統設定，將 PLC 端的通訊設定為相同的設定值。  
此時，設定值必須是僅指定結束碼。

● 程式範例

當 RS-232C 通訊埠的傳送準備完成旗標 A392.05 變成 ON 時，只要將 0.01 設定為 ON，即可在不轉換 D10 上位元開始的 3 位元的條件下，將 RS-232C 送出的觸發輸出「通訊讀取指令」@GL 傳送到連接至 CPU 模組內建 RS-232C 通訊埠的讀碼機。



● 執行訊號控制時



# RXD

指令名稱	指令記號	指令的各種組合	Fun No.	功能
CPU 模組內建的 RS232C 通訊埠、序列選擇埠輸入	RXD	@RXD	235	利用 PLC 系統設定的方式，指定為無程序模式下的啟始碼 / 結束碼，然後再由 CPU 模組內建的 RS-232 通訊埠或序列選擇埠，將接收到的資料由指定的啟始 CH 編號輸出指定的位元數。

符號	RXD						
	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>D</td> <td>D : 儲存接收資料的啟始CH編號</td> </tr> <tr> <td>C</td> <td>C : 控制資料</td> </tr> <tr> <td>N</td> <td>N : 儲存位元數 10進位&amp;0 ~ 256 (16進位#0000 ~ 0100)</td> </tr> </table>	D	D : 儲存接收資料的啟始CH編號	C	C : 控制資料	N	N : 儲存位元數 10進位&0 ~ 256 (16進位#0000 ~ 0100)
D	D : 儲存接收資料的啟始CH編號						
C	C : 控制資料						
N	N : 儲存位元數 10進位&0 ~ 256 (16進位#0000 ~ 0100)						

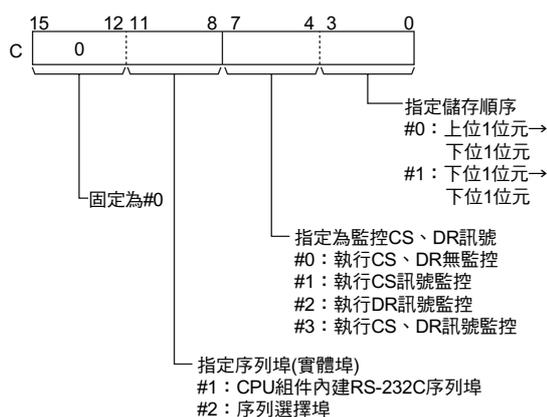
## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
D	儲存接收資料的啟始 CH 編號	UINT	可調整
C	控制資料	UINT	1
N	儲存位元數 10 進位 &0 ~ 256 (16 進位 #0000 ~ 0100)	UINT	1

### C : 控制資料



### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	○	○	○	○	○	○	○	○	○	-	-	-	-
C,N	○	○	○	○	○	○	○	○	○	○	-	-	-

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 PLC 系統設定為無程式模式以外的模式時，本旗標就會 ON</li> <li>當 C 的資料超過範圍時變為 ON</li> <li>當 N 的資料以 10 進位表示時超過 &amp; 0 ~ 256 或是 16 進位表示時超過 #0000 ~ 0100 的範圍內時，就會變成 ON</li> <li>否則為 OFF</li> </ul>

## 相關特殊輔助繼電器

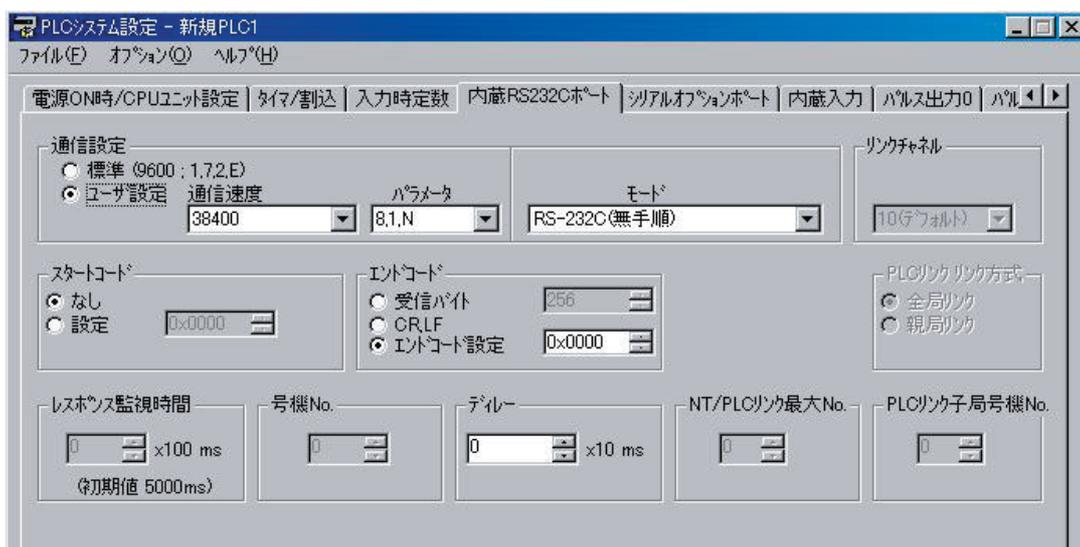
## ● 使用 CPU 模組內建 RS-232C 序列埠時

名稱	標籤	內容
RS-232C 通訊埠接收完成旗標	A392.06	在無程序模式下完成接收動作，本旗標就會變為 1 (ON)。 <ul style="list-style-type: none"> <li>指定接收位元數時：接收所指定的位元數時，本旗標就會 ON</li> <li>指定結束碼時：接收結束碼或 256 位元時，本旗標就會 ON</li> </ul>
RS-232C 通訊埠接收超過旗標	A392.07	在無程序模式下，一旦接收時超過接收的資料數，本旗標就會變成 1 (ON)。 <ul style="list-style-type: none"> <li>指定接收位元數時：接收完成後，只要在執行 RXD 指令前接收資料，本旗標就會 ON</li> <li>指定結束碼時：只要在接收結束碼 ~ 執行 RXD 指令的這段期間接收資料，本旗標就會 ON</li> <li>未接收結束碼時，只要收到 256 位元，且第 257 個位元非結束碼時，本旗標就會 ON</li> </ul>
RS-232C 通訊埠接收計數值	A393.00 ~ 393.15	在無程序模式下，以 16 進位來顯示所接收到的資料位元數 (0 ~ 256)。

## ● 使用序列選擇埠時

名稱	標籤	內容
序列選擇埠接收完成旗標	A392.14	在無程序模式下完成接收動作，本旗標就會變為 1 (ON)。 <ul style="list-style-type: none"> <li>指定接收位元數時：接收所指定的位元數時，本旗標就會 ON</li> <li>指定結束碼時：接收結束碼或 256 位元時，本旗標就會 ON</li> </ul>
序列選擇埠接收超過旗標	A392.15	在無程序模式下，一旦接收時超過接收的資料數，本旗標就會變成 1 (ON)。 <ul style="list-style-type: none"> <li>指定接收位元數時：接收完成後，只要在 RXDU 指令讀取資料前收到下一個資料，本旗標就會 ON</li> <li>未接收結束碼時，只要收到 256 位元，且第 257 個位元非結束碼時，本旗標就會 ON</li> </ul>
序列選擇埠接收計數值	A394.00 ~ 394.15	在無程序模式下，序列埠以 16 進位來顯示所接收到的資料位元數 (0 ~ 256)。

## PLC 系統相關設定



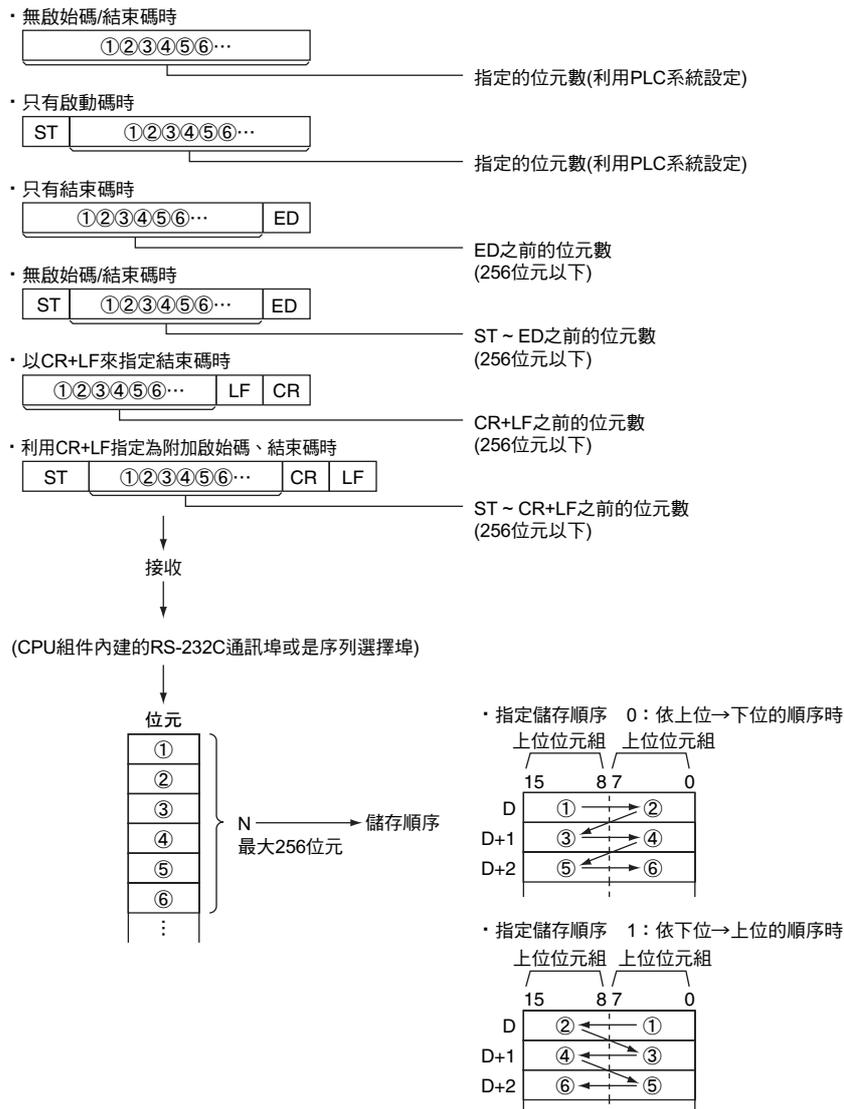
### 功能

- 利用 C 的位元 8 ~ 11 所指定的 CPU 模組內建 RS-232C 通訊埠(無程序模式下)或序列選擇埠的序列埠(無程序模式下)，將接收完成的資料由 D 所指定的儲存接收資料的啟始 CH 編號開始儲存 N 所指定的儲存位元數。當接收完成的資料不存在 N 所指定的儲存位元數時，就會儲存已存在的接收完成資料。
- 利用 PLC 系統設定的方式，將接收訊息的訊框格式設定如下。
  - 1) 是否已經設定啟始碼 ( 如果答案為 「是」，啟始碼即為 #00 ~ FF)
  - 2) 是否設定結束碼 ( 指定為 「是」時，結束碼將以 #00 ~ FF 或 CR+LF 表示，指定為 「否」時，接收資料數將以 #00 ~ FF : 1 ~ 256、#00 表示 256)
- 根據 C 的位元 0 ~ 3 所指定的儲存順序，將接收資料儲存至 D 以後的位置。
- 當接收完成旗標 ON 時
  - 1) 利用 PLC 系統設定的方式來接收所指定的接收位元數時，接收完成旗標^(*1) 就會 ON。此時，接收計數值^(*2) 和 PLC 系統設定時所指定的無程序模式下的接收資料數相等。
  - 2) 利用 PLC 系統設定來指定結束碼時，一旦接收到結束碼，接收完成旗標^(*1) 就會 ON。
  - 3) 接收 256 位元的資料時，接收完成旗標^(*1) 就會 ON。  
在接收完成旗標^(*1)ON 的狀態下，執行無程序指令並接收資料時，接收超過旗標^(*3) 就會 ON。
- 執行本 RXD 指令，由接收緩衝區開始將資料儲存於 D 時，接收完成旗標^(*1) 就會 OFF，而接收計數器^(*2) 則會變為 0。又，當接收超過旗標 ON 時，本旗標就會 OFF。
- 即使 RS-232C 通訊埠重新啟動旗標^(*4) 設定為 ON^(*1)，接收完成旗標^(*1) 仍然會變成 OFF，而接收計數值^(*2) 則變為 0。此時，當接收超過旗標 ON 時，本旗標就會 OFF。
- 利用 C 的位元 4 ~ 7 所指定的 CS、DR 訊號的監控指令將會被反映為下列方式。
  - 1) 指定為監控 CS 訊號時，CS 訊號的狀態就會被輸出至 D 所指定 CH 的位元 15。
  - 2) 指定為監控 DR 訊號時，CS 訊號的狀態就會被輸出至 D 所指定 CH 的位元 15。
  - 3) 指定為監控 CS 訊號及 DR 訊號時，CS 訊號的狀態就會被輸出至 D 所指定的 CH 位元 15，而 DR 訊號的狀態則會被輸出至 D 所指定 CH 的位元 14。
- 當 CS、DR 訊號的監控指定被設定為 #1 ~ 3 時，無論接收完成旗標^(*1) 的狀態為何，皆會讀取資料。指定為監控 CS 訊號或 DR 訊號時，將不會執行接收資料的儲存動作。
- 可接收的位元數最多為 259 位元(接收資料區: 最多為 256 位元, 啟始碼: 1 位元、結束碼區: CR+LF 所指定的 2 位元)。
- 請將不含啟始碼及結束碼的接收資料區大小，指定至 N。

### 相關特殊輔助繼電器 / 中斷繼電器的位址

* 1 接收完成旗標	內建 RS-232C 選擇埠	A392CH 位元 06 A392CH 位元 14
* 2 接收計數值	內建 RS-232C 選擇埠	A393CH A394CH
* 3 接收超過旗標	內建 RS-232C 選擇埠	A392CH 位元 07 A356CH 位元 15
* 4 選擇埠重新啟動旗標	內建 RS-232C 選擇埠	A526CH 位元 00 A526CH 位元 01

## ● 啟始碼 / 結束碼設定及接收資料



### 提示

- 序列埠所對應的 RXD 指令會在執行 RXD 指令後，清除接收緩衝區，因此，您無法利用多個 RXD 指令來執行分割接收，此點需特別注意。

### 使用上的注意事項

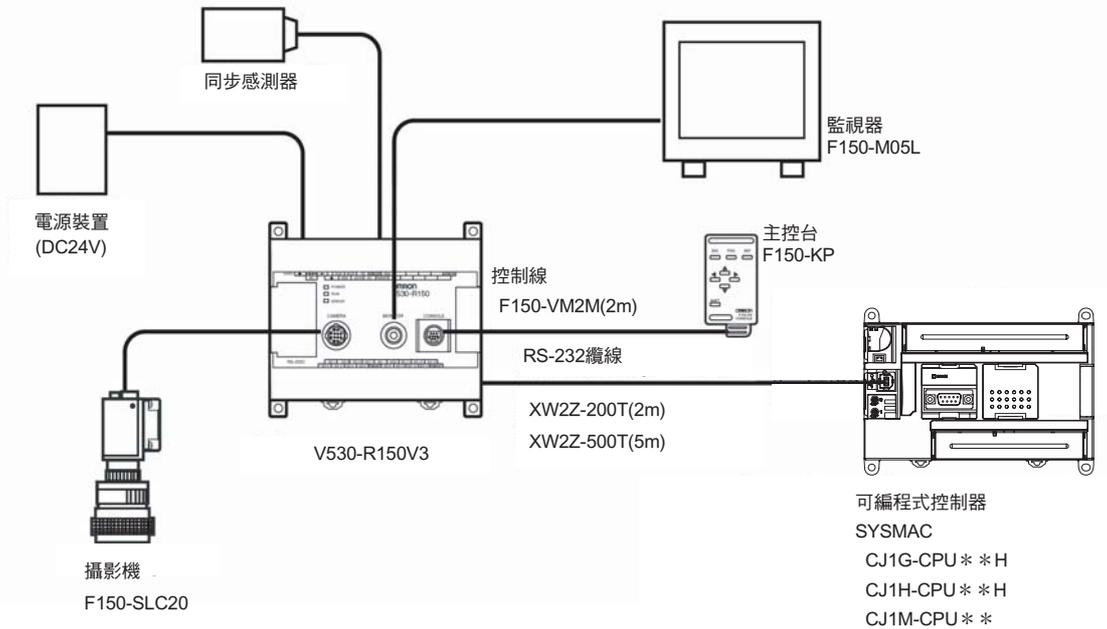
- 本指令僅適用於無程序模式下所設定的 CPU 模組內建的 RS-232C 通訊埠、或序列選擇埠的序列埠。
- 當接收完成旗標 (CPU 模組內建的 RS-232C:A392.06、序列選擇埠:A392.14) 變成 1 (ON) 時，請設定為執行本指令 (由接收緩衝區) 並接收資料。
- 接收資料時，若未利用 RXD 指令來讀取資料，則無法接收下一個資料。當接收完成旗標 ON，就會利用 RXD 指令來讀取接收資料直到下一次接收為止。
- 當 N 被指定為 0 時，D 以後的位置將無法讀取資料。此時，接收完成旗標^(*1) 和接收超過旗標^(*3) 會變成 OFF，而接收計數值^(*2) 則會變成 0。

### 程式例

#### ● 接收資料時

利用和通用外埠裝置之間的通訊為例，從 2 維讀碼機接收資料 (V530-R150V3 型)。

#### ● 硬體架構



本範例係連接至 CPU 模組內建的 RS-232 通訊埠。  
請在一開始就先設定好讀碼機的讀取條件設定。

#### ● 通訊設定

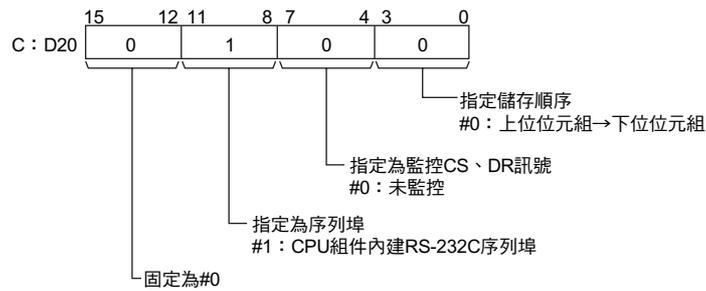
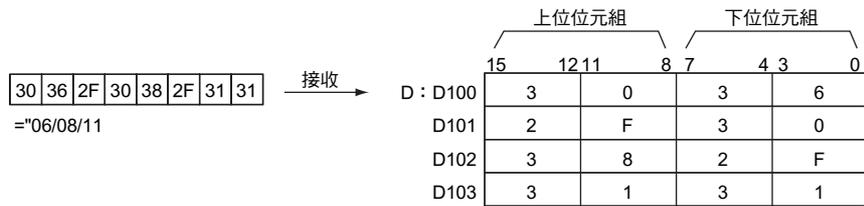
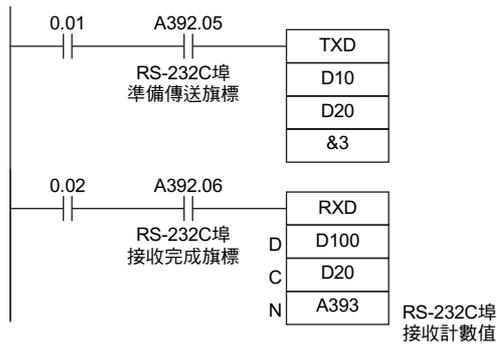
設定讀碼機時，請依照下列出廠時的初始值來設定。

通訊模式	無程序模式
通訊速度	38400bps
資料位元長度	8 位元
同位元檢查	無
停止位元	1 位元
啟始碼	無
結束碼	#000D (CR)

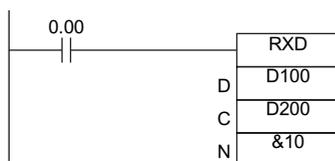
請利用 PLC 系統設定，將 PLC 的通訊設定為相同的設定值。  
此時，設定值必須是僅指定結束碼。

#### ● 程式範例

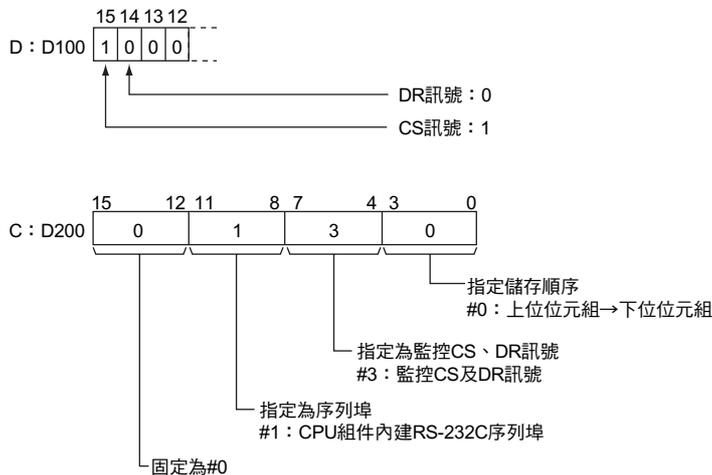
當 RS-232C 通訊埠接收完成旗標 A392.06 變成 ON 時，只要將 0.02 設定為 ON，就會將連接至 CPU 模組內建的 RS-232C 通訊埠的讀碼機所接收到的讀取結果，儲存至 D100 的上位元，所儲存的内容為 RS-232C 通訊埠的接收計數值 A393CH 的位元數。



● 執行訊號監控時



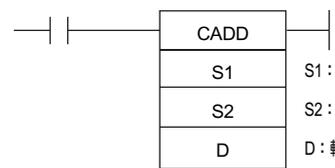
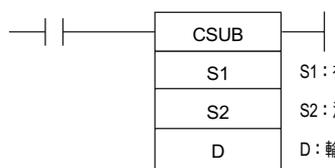
當0.00變成ON時，就會將CS訊號反映至D100的位元15，而DR訊號則會被反映至位元14。



# 時鐘功能專用指令

## CADD/CSUB

指令名稱	指令記號	指令的各種組合	Fun No.	功能
日曆加法運算	CADD	@CADD	730	在時刻資料中執行時間資料的加法運算。
日曆減法運算	CSUB	@CSUB	731	由時刻資料開始執行時間資料的減法運算。

符號	CADD	CSUB
	 <p>S1：被加資料(時刻)的下位CH編號 S2：加算資料(時間)的下位CH編號 D：輸出演算結果(時刻)的下位CH編號</p>	 <p>S1：被減資料(時刻)的下位CH編號 S2：減算資料(時間)的下位CH編號 D：輸出演算結果(時刻)的下位CH編號</p>

### 是否適用於特定區域

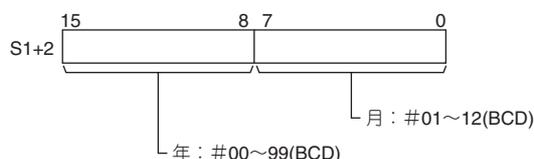
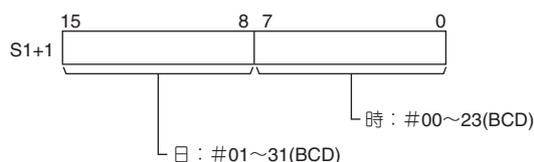
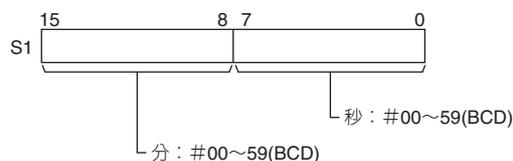
區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 運算元種類的說明

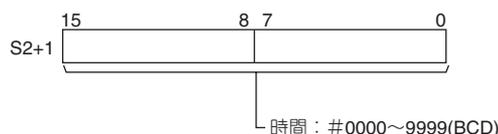
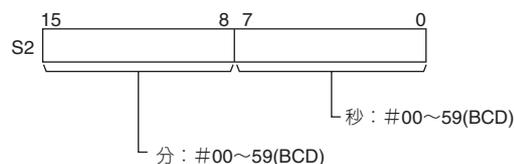
運算元種類	內容	資料型態	容量
S1	CADD：被加資料(時刻)的下位CH編號 CSUB：被減資料(時刻)的下位CH編號	WORD	3
S2	CADD：加算資料(時間)的下位CH編號 CSUB：減算資料(時間)的下位CH編號	DWORD	2
D	輸出演算結果(時刻)的下位CH編號	WORD	3

#### • CADD

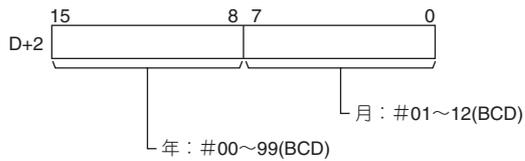
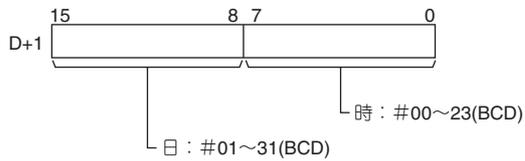
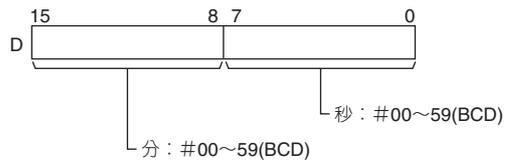
S1：被加算データ(時刻)下位CH番号



S2：加算資料(時間)的下位CH編號

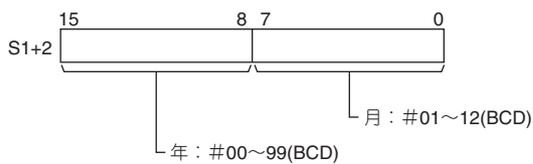
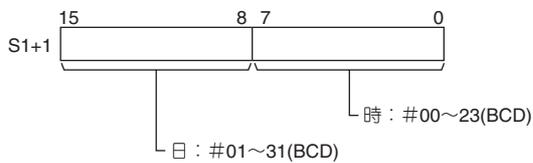
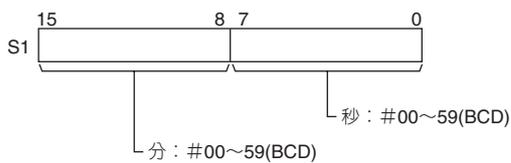


D：輸出演算結果 (時刻) 的下位 CH 編號

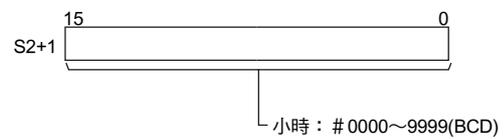
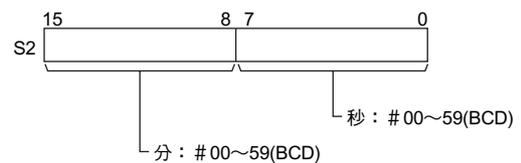


• CSUB

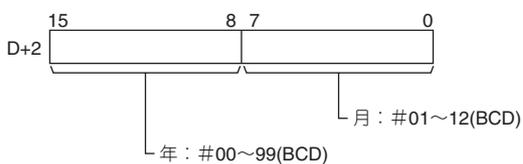
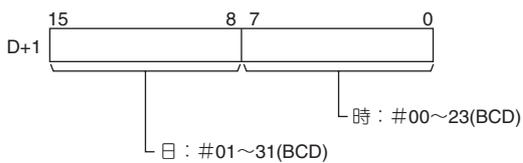
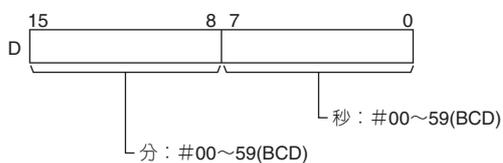
S1：被減資料 (時刻) 的下位 CH 編號



S2：減算資料 (小時) 的下位 CH 編號



D：輸出演算結果 (時刻) 的下位 CH 編號



● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM				
S1										—			
S2	○	○	○	○	○	○	○	○	○	○	—	—	—
D										—			

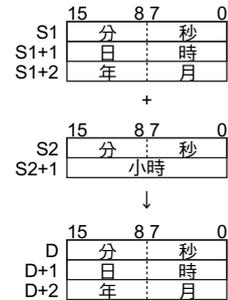
條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S1 的時刻資料超出範圍時，本旗標就會 ON</li> <li>當 S2 的時間資料超出範圍時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>
= 旗標	P_EQ	<ul style="list-style-type: none"> <li>利用 CSUB 指令讓演算結果變成 0 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

功能

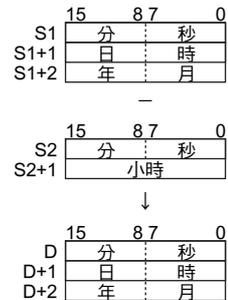
● CADD

針對 S1 所指定的時刻資料 (年 / 月 / 日 / 時 / 分 / 秒) 和 S2 所指定的時間資料 (小時 / 分 / 秒) 執行加法運算，然後再將結果當作時刻資料 (年 / 月 / 日 / 時 / 分) 輸出至 D。



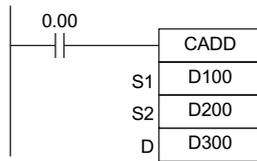
● CSUB

從 S1 所指定的時刻資料 (年 / 月 / 日 / 時 / 分 / 秒) 到 S2 所指定的時間資料 (小時 / 分 / 秒) 執行減法運算，然後再將結果當作時刻資料 (年 / 月 / 日 / 時 / 分) 輸出至 D。



## 程式例

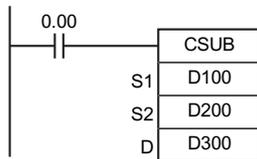
### ● CADD



當0.00變成ON時，就會針對D100 ~ D102的時刻資料(年月日時分秒)和D200 ~ D201的小時資料(小時/分秒)執行加法運算，並且儲存D300 ~ D302的時刻資料(年月日時分秒)。

S1 : D100	15	8	7	0	
D101	30	20			30分20秒
D102	10	18			10日18時
D102	99	12			99年12月
+					
S2 : D200	15	8	7	0	
D201	10	15			10分15秒
D201	06	00			600小時
↓					
D : D300	15	8	7	0	
D301	40	35			40分35秒
D301	04	18			4日18時
D302	00	01			2000年1月

### ● CSUB



當0.00變成ON時，就會針對從D100 ~ D102的時刻資料(年月日時分秒)到D200 ~ D201的小時資料(小時/分秒)執行減法運算，並且儲存D300 ~ D302的時刻資料(年月日時分秒)。

S1 : D100	15	8	7	0	
D101	30	20			30分20秒
D102	10	18			10日18時
D102	98	07			98年7月
-					
S2 : D200	15	8	7	0	
D201	10	15			10分15秒
D201	00	50			50小時
↓					
D : D300	15	8	7	0	
D301	20	05			20分5秒
D301	08	16			8日16時
D302	98	07			98年7月

# DATE

指令名稱	指令記號	指令的各種組合	Fun No.	功能
修正時鐘	DATE	@DATE	735	根據所指定的時鐘資料來變更內部時鐘的數值。

符號	DATE	
		S：時鐘資料的下位CH編號

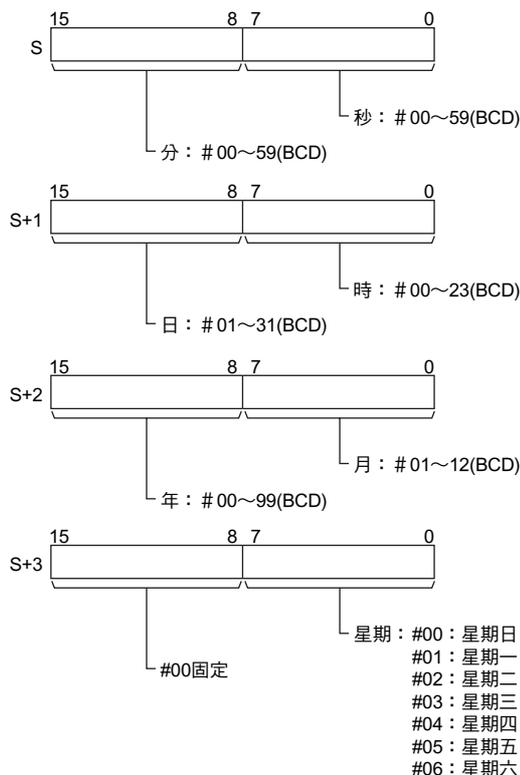
## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	時鐘資料的下位 CH 編號	LWORD	4

S：時鐘資料的下位 CH 編號



(註) S ~ S+3 必須屬於相同的運算元種類。

## ● 運算元種類

內容	CH 位址								間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	*DM					
S1	○	○	○	○	○	○	○	○	○	○	—	—	—	—

## 條件旗標的動作

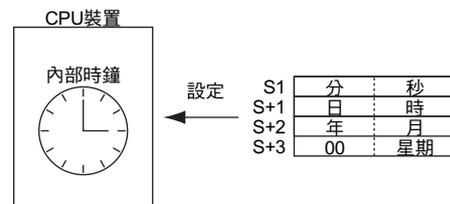
名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 S+3、S+2、S+1、S 的資料超過範圍時變為 ON</li> <li>否則為 OFF</li> <li>利用 CP1E-E □□□□-□來執行 DATE 指令時，本旗標就會 ON</li> </ul>

## 相關特殊輔助繼電器

名稱	位址	內容
時鐘資料	A351 ~ A354CH	A351.00 ~ A351.07：秒 (00 ~ 59) (BCD) A351.08 ~ A351.15：分 (00 ~ 59) (BCD) A352.00 ~ A352.07：時 (00 ~ 23) (BCD) A352.08 ~ A352.15：日 (01 ~ 31) (BCD) A353.00 ~ A353.07：月 (01 ~ 12) (BCD) A353.08 ~ A353.15：年 (00 ~ 99) (BCD) A354.00 ~ A354.07：星期 (00 ~ 06) (BCD) 00：星期日、01：星期一、02：星期二、03：星期三、04：星期四、05：星期五、 06：星期六

## 功能

根據 S ~ S+3 所指定的時鐘資料 (4CH) 來變更內部時鐘的數值，然後再直接反映至特殊輔助繼電器的時鐘資料區 (A351 ~ 354CH)。



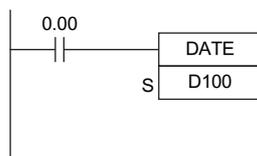
## 提示

- 時鐘資料亦可利用其他週邊工具 ( 及寫入 FINS 指令的時間資訊 (#0702) ) 進行設定。

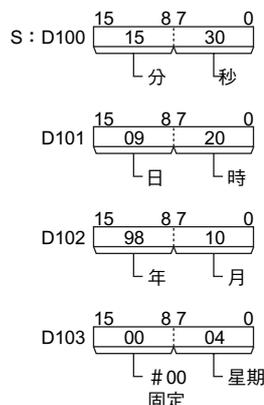
## 使用上的注意事項

- 即使所指定日期時間為日曆上所沒有日期時，也不會發生錯誤。(例：11月31日等)
- 利用 E 型 CPU 模組 (CP1E-E □□□□-□型) 來執行本指令時，異常旗標就會 ON，此時將無法執行指令。E 型 CPU 模組在出廠時已經將 A351 ~ A354 固定為 01 年 01 月 01 日 01 時 01 分 01 秒星期日。

## 程式例



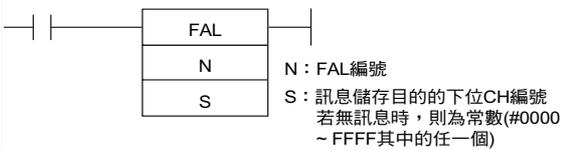
當 0.00 變成 ON 時，就會將內部時鐘設定為 1998 年 10 月 9 日 20 時 15 分 30 秒星期四。



# 故障診斷指令

## FAL

指令名稱	指令記號	指令的各種組合	Fun No.	功能
運轉持續故障診斷	FAL	@FAL	006	可登錄或解除使用者所定義的運轉持續異常。

符號	ΦΛΛ	
	登錄 / 解除使用者所定義的運轉持續異常時 	利用系統來登錄運轉持續異常時 

### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 運算元種類的說明

運算元種類	內容	資料型態	容量
N	FAL 號碼	僅限於常數	1
S	如果是儲存對象的下位 CH 編號或時無訊息時, 則為常數 / 所發生的故障碼	WORD	可調整

#### ● 登錄 / 解除使用者所定義的持續運轉異常時

(註) 運算元種類 N 的內容和特殊輔助繼電器 A529CH (發生系統異常的 FAL/FALS 編號) 中的數值必須相異。

	登錄運轉持續異常	解除運轉持續異常
N	1 ~ 511 : FAL 編號 (和 FALS 編號共用)	0
S	指定 CH : S ~ S+7 的 16 個字元 (ASCII 碼) 會被顯示在週邊工具上 #0000 ~ FFFF : 無訊息	#FFFF : 全部解除 & 1 ~ 511 (10 進位) #0001 ~ 01FF (16 進位) : 欲解除的 FAL 編號 上述以外的數值或指定 CH : 解除 1 個重要性最高的持續運轉異常

#### ● 利用系統來登錄運轉持續異常時

(註) 運算元種類 N 的內容和特殊輔助繼電器 A529CH (發生系統異常的 FAL/FALS 編號) 中的數值必須相同。

	登錄運轉持續異常
N	1 ~ 511 : FAL 編號 (和 FALS 編號共用)
S	發生的故障碼 (請參閱後述)
S + 1	設定異常內容 (請參閱後述)

#### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
N	—	—	—	—	—	—	—	—	—	○	—	—	—
S	○	○	○	○	○	○	○	○	○	○	○	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 N 的資料超出 0 ~ 511 的範圍時，本旗標就會 ON</li> <li>當系統發生持續運轉異常時，只要所儲存的異常碼 ( 停止異常、持續異常 ) 無法對應至所發生的故障碼 / 儲存異常內容的下位 CH 編號時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 相關特殊輔助繼電器

### ● 使用者定義的運轉持續相關異常

名稱	位址	內容
FAL 異常旗標	A402.15	執行 FAL 指令時，相關位元就會變成 1 (ON)。
執行 FAL 編號	A360.01 ~ A391.15	執行 AL 指令時，本旗標就會變成 1 (ON)。 A360.01 ~ A391.15 分別對應至 FAL1 ~ 511。

### ● 系統發生運轉持續相關異常

名稱	位址	內容
發生系統異常的 FAL/FALS 編號	A529CH	若希望執行 FAL 指令或 FALS 指令，刻意讓系統發生異常時，必須先設定虛擬使用的 FAL/FALS 編號。 &1 ~ 511 或 #0001 ~ 01FF

### ● 使用者定義 / 系統共通的運轉持續相關異常

名稱	位址	內容
異常記錄儲存區	A100 ~ 199CH	發生異常時 ( 執行 FAL 指令 )，系統會儲存 20 個最新的故障碼、異常發生時刻。
故障代碼	A400CH	發生異常時 ( 執行 FAL 指令 )，所對應的故障碼將會被儲存起來。 #4101 ~ 42FF 分別對應至 FAL1 ~ 511 (16 進位的 #001 ~ 1FF)。 ( 註 ) 當異常同時發生時，將儲存重要度最高的異常碼。

## 解除運轉持續異常

### ● 解除使用者所定義的運轉持續異常

將 N 所指定的 FAL 編號設定為 0 後，即可解除運轉持續異常。  
實際設定依 S 所指定的資料而有不同。

S	內容
10 進位 & 1 ~ 511 或 16 進位 #0001 ~ 01FF	可解除該編號的 FAL 異常。
16 進位 #FFFF	可解除所有的持續運轉異常 ( 包含系統的持續運轉異常 )。
16 進位的 #0200 ~ FFFE 及指定 CH	針對已發生的持續運轉異常中 ( 包含系統的持續運轉異常 )，解除其中重要性最高的一項異常。發生多個 FAL 異常時，則會優先解除編號最新的 FAL。

### ● 解除因系統造成的持續運轉異常

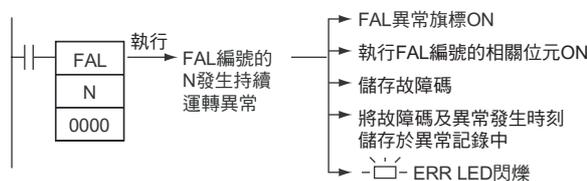
要解除刻意產生的因系統造成的持續運轉異常時，可使用下列 2 種方法①將 PLC 本體的電源由 OFF → ON，或是② ( 在 PLC 本體電源 ON 的狀態下 ) 和該異常實際發生時採取相同的處理方法。

## 功能

### ● 登錄使用者所定義的運轉持續異常

FAL/FALS 編號	1 ~ 511 (16 進位 #001 ~ 1FF)
FAL 故障代碼	16 進位 #4101 ~ #42FF
特殊輔助繼電器的執行 FAL 編號	A360.01~A391.15

當 N 所指定的資料為 1~511，而且和特殊輔助繼電器 A529CH ( 系統發生異常的 FAL/FALS 編號 ) 中的數值不一致時，一旦輸入條件成立，就會視為 FAL 編號 N 發生異常 ( 運轉持續異常 )，並且執行下列動作。



- 1) 將特殊輔助繼電器的 FAL 異常旗標 (A402.15) 設定為 ON ( 持續運轉 )。

- 2) 當 N 所指定的資料變為 1 ~ 511 時，特殊輔助繼電器的執行 FAL 編號 (A360 ~ A391) 的相關位元就會 ON。  
(A360.01 ~ A391.15：對應至 FAL 編號的 1 ~ 511)
- 3) 將特殊輔助繼電器的故障碼 (A400) 設定為故障碼 (可對應至 FAL 編號 1~ 511 (10 進位並顯示為故障碼 #4101 ~ 42FF))。
- 4) 將故障碼及異常發生時刻儲存至特殊輔助繼電器的異常記錄儲存區 (A100 ~ 199) 中。

(註)不過，若利用 PLC 系統設定並檢查出「未登錄 FAL 至異常記錄中」，則故障碼及異常發生時刻不會被儲存至異常記錄儲存區。

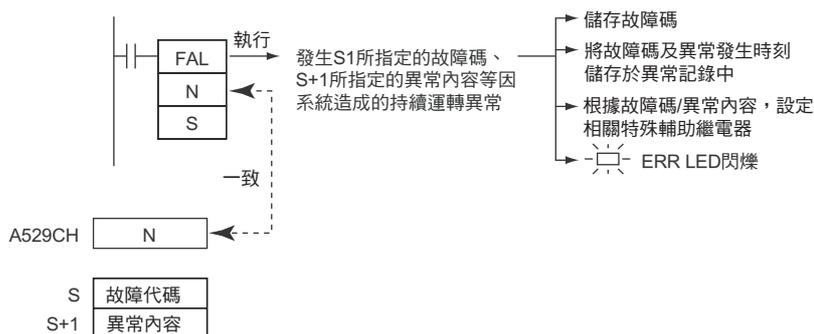
5) CPU 模組的 ERR LED 開始閃爍，並持續進行運轉。

6) 利用 S 將訊息指定為 CH，即可登錄訊息 (顯示在週邊工具上)。

(註)當重要度高於本指令所登錄的異常 (因系統造成停止運轉異常、包含因執行 FAL 編號造成的停止運轉異常) 同時發生時，該異常碼就會被設定至故障碼 (A400) 中。

### ● 登錄因系統造成的持續運轉異常

當 N 所指定的資料為 1 ~ 511 而且和 A529CH (發生系統異常的 FAL/FALS 編號) 中的數值一致時，一旦輸入條件成立，系統就會刻意產生 S、S+1 所指定的故障碼及異常內容等因系統造成的持續運轉異常，同時還會執行下列動作。



- 1) 將故障碼設定為特殊輔助繼電器的故障碼。
- 2) 將故障碼及異常發生時刻儲存至特殊輔助繼電器的異常記錄儲存區 (A100 ~ 199) 中。
- 3) 根據故障碼 / 異常內容，設定相關特殊輔助繼電器。
- 4) CPU 模組的 ERR LED 開始閃爍，並持續進行運轉。

參考：利用 FAL 指令執行「因系統產生持續運轉異常」的動作可在程式偵錯 (Debug) 時使用。例如，利用刻意產生異常的方式，即可確認 PT (可程式化終端機) 等的 MMI 是否顯示錯誤訊息。

- (註) 1 刻意產生因系統造成的持續運轉 / 停止異常時，特殊輔助繼電器 A529CH (發生系統異常的 FAL/FALS 編號) 中的數值即為虛擬使用的 FAL/FALS 編號 (FAL、FALS 指令共用)。(此數值為虛擬數值，因此不會被反映至執行的 FAL 編號或故障碼中)。
- 刻意產生多個因系統造成的持續運轉 / 停止異常時，請將 A529CH 中的相同數值指定為 N，並且利用相異的 S、S+1 內容來執行多個 FAL/FALS 指令。
- 2 當重要度高於本指令刻意發生的異常 (因系統造成停止運轉異常、包含因執行 FAL 編號造成的停止運轉異常) 同時發生時，該異常碼就會被設定至故障碼 (A400) 中。
  - 3 如果要解除以本指令所刻意產生的異常時，請先將 PLC 本體的電源 OFF，然後再重新啟動 (ON)。此時，和 PLC 電源 ON 狀態下，實際發生相關異常時的處理方法相同。
- 詳細內容請參閱「SYSMAC CP1E CPU 模組使用手冊 硬體篇」或「SYSMAC CP1E CPU 模組使用手冊 軟體篇」。

### ● S、S+1 的指定方法

異常名稱	S (16 進位)	S+1
PLC 系統設定異常	#009B	PLC 系統設定的異常位置 #0000~FFFF
內建的類比訊號異常	#008A	16 進位 #0000
選擇埠異常	#00D1	選擇埠插槽編號 16 進位 #0001
電池異常	#00F7	16 進位 #0000

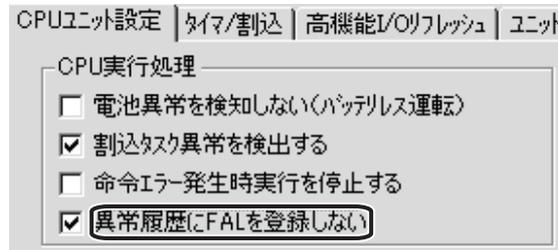
### ● 執行 FAL 指令時是否登錄異常記錄

執行 FAL 指令時，系統會刻意產生使用者所定義的持續運轉異常，不過您也可以選擇不登錄異常 (故障碼及異常發生時刻) 至特殊輔助繼電器的異常記錄儲存區 (A100 ~ 199CH) 中 (註)。

(註) 此時，該異常會被反映為特殊輔助繼電器 A402.15 的 FAL 異常旗標、A360 ~ 391CH 的執行 FAL 編號及 A400CH 的故障碼。

此功能只能儲存因系統造成的異常至異常記錄時使用。例如，以常用 FAL 指令的應用程式來說，可防止偵錯時，異常記錄儲存區因 FAL 指令所造成的異常而滿溢的情形。  
請利用下列 PLC 系統設定的方法來設定。

- PLC 系統設定時 ( 使用 CX-Programmer 時 )



(註) 利用上述 PLC 系統設定的方式，設定為「未登錄至異常記錄」時，下列異常仍然會被登錄至異常記錄中。

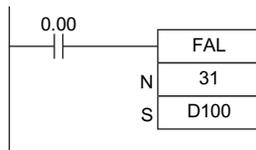
- 因執行 FALS 指令所造成的停止運轉異常
- 因系統所造成的持續運轉異常
- 因系統所造成的停止運轉異常
- 執行 FALS 指令所刻意產生的系統持續運轉異常
- 執行 FALS 指令所刻意產生的系統停止運轉異常

### ● 發生使用者所定義的持續運轉異常時的訊息顯示

- 若希望執行 FAL 指令時，同時讓訊息顯示在週邊工具上，請將 CH 編號設定至 S。( 如果不需要訊息時，請將任一常數 #0000 ~ FFFF 設定至 S。)
- 執行 FAL 指令後，就會登錄訊息。登錄訊息後，即可顯示訊息。
- 利用 ASCII 碼即可將相當於 16 個字元設定至 S ~ S+7CH 共 8CH 中。
- 訊息的顯示順序為上位元→下位元。
- 訊息的分隔符號為 NUL (#00)，除了 NUL 外，最多可顯示 16 個字元。
- 執行 FAL 指令後，只要變更新來設定訊息的 CH 內容，訊息就會根據變更內容而改變。

### 程式例

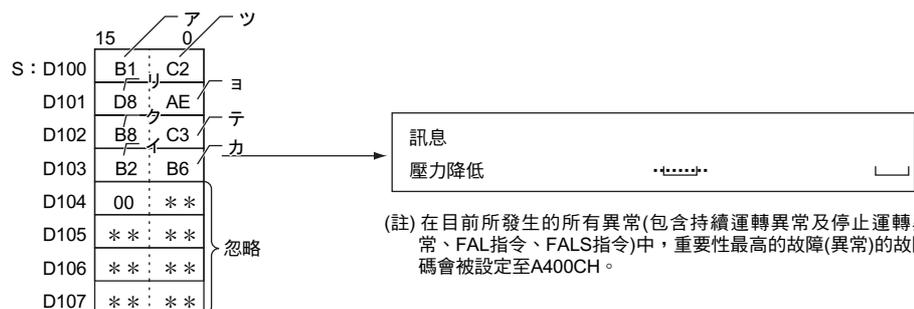
#### ● 登錄時 ( 設定 N : 001 ~ 511 )



當 0.00 變成 ON 時，系統就會判斷為發生 FAL 編號 31 的持續運轉故障(異常)，並執行下列處理。

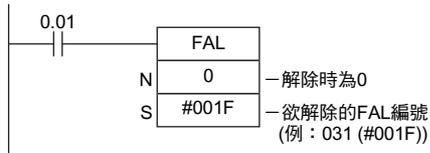
- 1) 將特殊輔助繼電器的 FAL 異常旗標 A402.15 設定為 ON
- 2) 將 FAL 編號所對應的特殊輔助繼電器 A361CH 的位元 15 設定為 ON
- 3) 將故障碼 #411F 設定為特殊輔助繼電器故障碼 A400 (當此故障的重要性最高時) (註)
- 4) 將故障碼及異常發生時刻儲存至特殊輔助繼電器的異常記錄儲存區(A100 ~ 199)中
- 5) CPU 組件 ERR LED 開始閃爍
- 6) D100 ~ D107 共 8CH 的資料會被視為 16 個字元的 ASCII 碼，並且顯示在週邊工具上。  
(例：壓力降低)

(註) 若不需要訊息時，請在 S 設定常數(#0000 ~ FFFF 的其中一個)。



● 解除時 (N：設定為 0)

· 欲解除已發生的故障時(S：設定欲解除的故障碼)



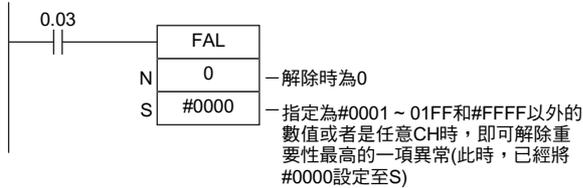
當0.01變成ON時，就會解除FAL編號31的故障(異常)。  
A361CH的位元15將變為OFF。  
A402.15變為OFF。

· 欲解除所有的持續運轉異常時(包含因FAL指令及系統自行診斷造成的異常) (設定為S：#FFFF)



當0.02變成ON時，就會同時解除所有的持續運轉異常  
A360.01 ~ A391.15的所有位元皆變為OFF。  
A402.15變為OFF。

· 欲解除目前所發生的持續運轉異常(包含因FAL指令及系統自行診斷造成的異常)中，重要性最高的1項異常時(S：設定為上述以外的數值)

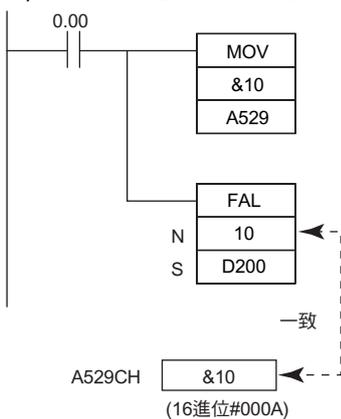


當0.03變成ON時，就會解除目前所發生的持續運轉異常中，重要性最高的一項異常。  
A400CH的故障碼將被重置。  
當FAL指令所造成的故障(異常)為重要性最高者時，A360.01 ~ A391.15所對應的位元就會OFF，而A402.15則會變為OFF。

● 欲產生因系統造成的持續運轉異常時

當 0.00 變成 ON 時，就會產生選擇埠異常。使用 10 作為虛擬的 FAL 編號 (預先將 &10 (16 進位的 #000A) 儲存在 A529CH 中)，同時執行下列處理動作。

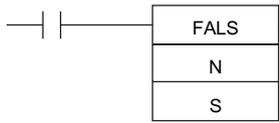
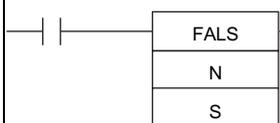
- 1) 將故障碼 #00D1 設定為特殊輔助繼電器故障碼 A400 (當此故障為重要性最高者時)
- 2) 將故障碼及異常發生時刻儲存至特殊輔助繼電器的異常記錄儲存區 (A100 ~ 199) 中
- 3) 特殊輔助繼電器 A315.13 (選擇埠異常旗標) 變為 1 (ON)
- 4) CPU 模組 ERR LED 開始閃爍
- 5) 發生「選擇埠異常」



S : D200 #00D1 故障代碼：#00D1 (選擇埠設定異常)  
D201 #0001 異常的插槽編號：1 (16進位#0001)

# FALS

指令名稱	指令記號	指令的各種組合	Fun No.	功能
運轉停止故障診斷	FALS	—	007	登錄使用者所定義的停止運轉異常。

符號	FALS	
	登錄 / 解除使用者所定義的運轉持續異常時  <p>N : FALS編號 S : 訊息儲存目的的下位CH編號 無訊息時為常數 (#0000 ~ FFFF的其中一個)</p>	利用系統來登錄運轉持續異常時  <p>N : FALS編號 S : 發生的故障碼/異常內容 儲存的下位CH編號</p>

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
N	FALS 編號	僅限於常數	1
S	如果是儲存對象的下位 CH 編號或時無訊息時，則為常數 / 所發生的故障碼	WORD	可調整

### ● 登錄使用者所定義的運轉停止異常時

(註) 運算元種類 N 的內容和特殊輔助繼電器 A529CH (發生系統異常的 FAL/FALS 編號) 中的數值必須相異。

登錄運轉停止異常	
N	1 ~ 511 : FALS 編號 (和 FAL 編號共用)
S	指定 CH : S ~ S+7 的 16 個字元 (ASCII 碼) 會被顯示在週邊工具上 #0000 ~ FFFF : 無訊息

### ● 利用系統來登錄運轉停止異常時

(註) 運算元種類 N 的內容和特殊輔助繼電器 A529CH (發生系統異常的 FAL/FALS 編號) 中的數值必須相同。

登錄運轉停止異常	
N	1 ~ 511 : FALS 編號 (和 FAL 編號共用)
S	發生的故障碼 (請參閱後述)
S + 1	設定異常內容 (請參閱後述)

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
N	—	—	—	—	—	—	—	—	—	○	—	—	—
S	○	○	○	○	○	○	○	○	○	○	—	—	—

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當 N 的資料超出 1 ~ 511 的範圍時，本旗標就會 ON</li> <li>當系統發生停止運轉異常時，只要所儲存的異常碼 ( 停止異常、持續異常 ) 無法對應至所發生的故障碼 / 儲存異常內容的下位 CH 編號時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## 相關特殊輔助繼電器

### ● 使用者定義的運轉持續相關異常

名稱	位址	內容
FAL 異常旗標	A401.06	執行 FALS 指令時，本旗標就會變成 1 (ON)。

### ● 因系統所造成的停止運轉異常相關

名稱	位址	內容
發生系統異常的 FAL/FALS 編號	A529CH	若希望執行 FAL 指令或 FALS 指令，刻意讓系統發生異常時，必須先設定虛擬使用的 FAL/FALS 編號。 10 進位 &1 ~ 511 或 16 進位 #0001 ~ 01FF

### ● 使用者定義 / 系統共通的運轉停止相關異常

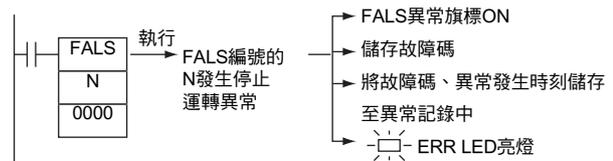
名稱	位址	內容
異常記錄儲存區	A100 ~ 199CH	發生異常時 ( 包括執行 FALS 指令 )，系統會儲存 20 個最新的故障碼、異常發生時刻。
故障代碼	A400CH	發生異常時 ( 包括執行 FALS 指令 )，所對應的故障碼將會被儲存起來。 16 進位的 #C101 ~ C2FF 可對應至 FALS1 ~ 511 (16 進位的 #001 ~ 1FF)。 ( 註 ) 當異常同時發生時，將儲存重要度最高的異常碼。

## 功能

### ● 登錄使用者所定義的運轉停止異常

FAL/FALS 編號	1 ~ 511 (16 進位 #001 ~ 1FF)
FAL 故障代碼	16 進位 #C101 ~ C2FF

當 FALS 編號 N 和特殊輔助繼電器 A529CH ( 發生系統異常的 FAL/FALS 編號 ) 中的數值不一致時，一旦輸入條件成立，就會被視為發生 FALS 編號 N，也就是使用者定義的停止運轉異常 ( 讓運轉停止的異常 )，並執行下列動作。

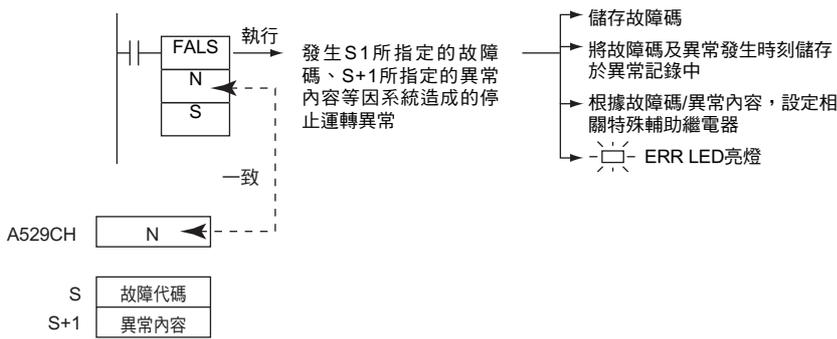


- 1) 將特殊輔助繼電器的 FALS 異常旗標 (A401.06) 設定為 ON ( 停止運轉 )。
- 2) 特殊輔助繼電器的故障碼 (A400) 會被對應至故障碼 (FALS 編號 1 ~ 511 (16 進位的 #001 ~ 1FF) 並設定為 16 進位的故障碼的 #C101 ~ C2FF) ( 不過，若同時發生其他故障時，只會儲存重要性最高者 )。
- 3) 將故障碼及異常發生時刻儲存至特殊輔助繼電器的異常記錄儲存區 (A100 ~ 199) 中。
- 4) CPU 模組的 ERR LED 會亮燈，並停止運轉。
- 5) 利用 S 將訊息指定為 CH，即可登錄訊息 ( 顯示在週邊工具上 )。

- ( 註 )
- 1 若同時發生重要性高於本指令所登錄的異常 ( 包含因系統造成的停止運轉異常 ) 時，該異常碼就會被設定至故障碼 (A400) 中。
  - 2 訊息的分隔符號為 NUL (#00)，除了 NUL 外，最多可顯示 16 個字元。
  - 3 當 3N 的資料超出 1 ~ 511 的範圍時，就會發生錯誤，而異常旗標會變成 ON。
  - 4 登錄使用者所定義的停止運轉異常後，I/O 記憶體及輸出模組的輸出接點狀態將變化如下：

		I/O 記憶體	輸出模組的輸出接點
I/O 記憶體保持旗標 (A500.12)	ON	保持	OFF
	OFF	保持	OFF

## ● 登錄因系統造成的停止運轉異常



當 FALS 編號 N 和特殊輔助繼電器 A529CH (發生系統異常的 FAL/FALS 編號) 中的數值一致時，一旦輸入條件成立，就會刻意產生 S 所指定的故障碼及 S+1 所指定的異常內容等因系統造成停止運轉異常。同時還會執行下列動作。

- 1) 將故障碼設定為特殊輔助繼電器的故障碼。
- 2) 將故障碼及異常發生時刻儲存至特殊輔助繼電器的異常記錄儲存區 (A100 ~ 199) 中。
- 3) 根據故障碼 / 異常內容，設定相關特殊輔助繼電器。
- 4) CPU 模組的 ERR LED 會亮燈，並停止運轉。

- (註) 1 刻意產生系統異常時，特殊輔助繼電器 A529CH (發生系統異常的 FAL/FALS 編號) 中的數值即為虛擬使用的 FAL/FALS 編號 (FAL、FALS 指令共用) (此數值為虛擬數值，因此不會被反映至故障碼)。
- 刻意產生多個系統異常時，請在 N 指定和 A529CH 中相同的數值，並且利用相異的 S、S+1 內容來執行 FAL/FALS 指令。
- 2 當重要度高於本指令刻意發生的異常 (包含系統停止運轉異常、因執行 FAL 編號造成的停止運轉異常) 同時發生時，該異常碼就會被設定至故障碼 (A400) 中。
  - 3 如果要解除以本指令所刻意產生的異常時，請先將 PLC 本體的電源 OFF，然後再重新啟動 (ON)。此時，和 PLC 電源 ON 狀態下，實際發生相關異常時的處理方法相同。
  - 4 詳細內容請參閱「SYSMAC CP1E CPU 模組使用手冊 硬體篇」或「SYSMAC CP1E CPU 模組使用手冊 軟體篇」。
- 4 刻意產生因系統造成的停止運轉異常時，I/O 記憶體及輸出模組的輸出接點狀如下 (和實際因系統造成的停止運轉異常一樣)。

		I/O 記憶體	輸出模組的輸出接點
I/O 記憶體保持旗標 (A500.12)	ON	保持	OFF
	OFF	清除	OFF

## ● S、S+1 的指定方法

異常名稱	S	S+1
	設定故障碼 (16 進位)	設定異常內容
記憶體異常	#80F1	位元 00 ~ 09：記憶體發生異常的位置 (如下) 位元 00：使用者程式 位元 01：IO 記憶體 位元 04：設定 PLC 系統 位元 2 ~ 3、5 ~ 15：無效
I/O 匯流排異常	#80CA	CP1W 系列的擴充 I/O 模組、擴充模組異常 #0A0A
I/O 件數過量	#80E1	位元 13 ~ 15：異常要因 位元 00 ~ 12：詳細要因 • CP1W 擴充 I/O 模組的 CH 數超過 位元 13~15：010 位元 00 ~ 12：全部 0
程式錯誤	#80F0	位元 08 ~ 15：異常要因 位元 15：UM 超過旗標 位元 14：不當指令旗標、不正確物件旗標 位元 13：微分超過旗標 位元 12：任務異常旗標 位元 11：無 END 旗標 位元 10：不當區域存取異常旗標 位元 09：DM 間接 BCD 異常旗標 位元 08：指令處理異常旗標 位元 00 ~ 07：無效
超過週期時間	#809F	16 進位 #0000

### ● 發生使用者定義的停止運轉異常時的訊息顯示

- 若希望執行 FALS 指令時，同時讓訊息顯示在週邊工具上，請將 CH 編號設定至 S。( 如果不需要訊息時，請將任一個常數 #0000 ~ FFFF 設定至 S。)
- 執行 FALS 指令後，就會登錄訊息。登錄訊息後，即可顯示訊息
- 利用 ASCII 碼即可將相當於 16 個字元設定至 S ~ S+7CH 共 8CH 中。
- 訊息的顯示順序為上位元→下位元。
- 訊息的分隔符號為 NUL (#00)，除了 NUL 外，最多可顯示 16 個字元。
- 執行 FALS 指令後，只要變更新來設定訊息的 CH 內容，訊息就會根據變更內容而改變。

### ● 解除因系統造成的停止運轉異常

要解除刻意產生的因系統造成的持續運轉異常時，可使用下列 2 種方法 ① 將 PLC 本體的電源由 OFF → ON，或是 ② ( 在 PLC 本體電源 ON 的狀態下 ) 和該異常實際發生時採取相同的處理方法。

### ● 解除 FALS 指令的方法

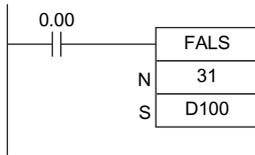
欲解除 FALS 指令時，請先排除異常原因，然後再 ① 利用週邊工具，執行「讀取 / 解除異常」操作動作 ② 將電源由 OFF 設定為 ON。

### 使用上的注意事項

- 登錄因系統造成的運轉停止異常時，如果 I/O 記憶體保持旗標 OFF，I/O 記憶體就會被清除，此點需特別注意。

## 程式例

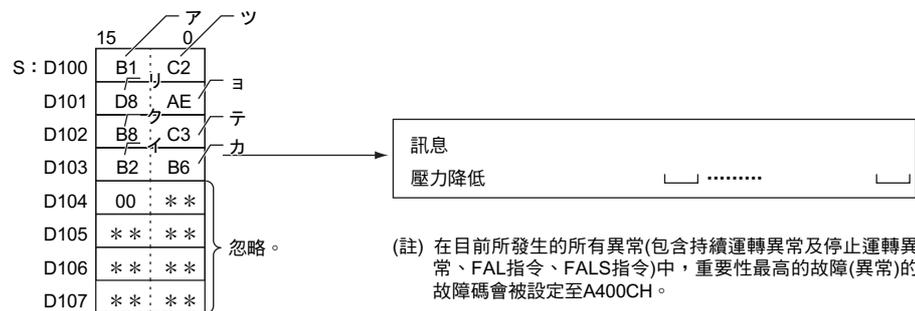
## ● 欲產生使用者定義的停止運轉異常時



當0.00變成ON時，系統就會判斷為發生FALS編號031的停止運轉故障(異常)，並執行下列處理。

- 1) 將特殊輔助繼電器的FALS異常旗標A401.06設定為ON
- 2) 將故障碼#C11F設定為特殊輔助繼電器故障碼A400 (當此故障為重要性最高者時)(註)
- 3) 將故障碼及異常發生時刻儲存至特殊輔助繼電器的異常記錄儲存區(A100 ~ 199)中
- 4) CPU組件ERR LED開始亮燈
- 5) D100 ~ D107共8CH的資料會被視為16個字元的ASCII碼，並且顯示在週邊工具上。(例：壓力降低)

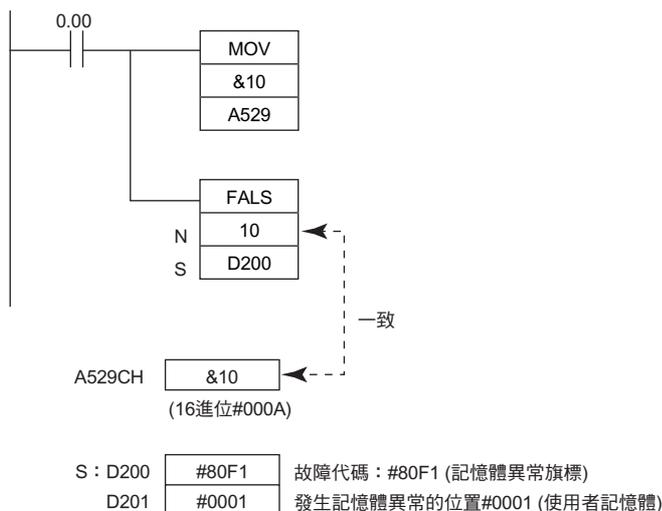
(註) 若不需要訊息時，請在S設定常數(#0000 ~ FFFF的其中任一個)。



## ● 欲產生因系統造成的停止運轉異常時

當 0.00 變成 ON 時，就會產生記憶體異常 (使用者程式異常)。使用 10 作為虛擬的 FAL 編號 (預先將 &10 (16 進位的 #80F1) 儲存在 A529CH 中)。同時執行下列處理動作。

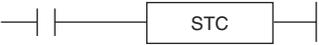
- 1) 將故障碼 #80F1 設定為特殊輔助繼電器故障碼 A400 (當此故障的重要性最高時)
- 2) 將故障碼及異常發生時刻儲存至特殊輔助繼電器的異常記錄儲存區 (A100 ~ 199) 中
- 3) 特殊輔助繼電器 A401.15 (記憶體異常旗標) 變成 1 (ON)。
- 4) CPU 模組的 ERR LED 會亮燈，並停止運轉。
- 5) 發生「記憶體異常」。



# 特殊指令

## STC/CLC

指令名稱	指令記號	指令的各種組合	Fun No.	功能
設定進位	STC	@STC	040	將 CY 旗標設定為 ON。
清除進位	CLC	@CLC	041	將 CY 旗標設定為 OFF。

符號	STC	CLC
		

### 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

### 條件旗標的動作

名稱	標籤	內容	
		STC	CLC
進位旗標 (Carry flag)	P_CY	ON	OFF

### 功能

#### ● STC

當輸入條件 ON 時，就會將條件旗標中的進位旗標設定為 1 (ON)。不過，利用本指令將進位旗標設定為 ON 後，仍然可以利用其他指令將進位旗標 OFF/ON。

ROL、ROR 指令可將進位旗標等旋轉位移，因此請使用 STC 指令來操作進位旗標。

#### ● CLC

當輸入條件 ON 時，就會將條件旗標中的進位旗標設定為 0 (OFF)。不過，利用本指令將進位旗標設定為 OFF 後，仍然可以利用其他指令將進位旗標 ON/OFF。+C、+CL、+BC、+BCL、-C、-CL、-BC、-BCL 指令可演算進位旗標等，為了避免受到先前指令所影響，請使用 CLC 指令。

ROL、ROR 指令可將進位旗標等旋轉位移，因此請使用 CLC 指令來操作進位旗標。

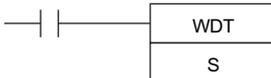
### 提示

- 加法運轉 / 減法運算指令包含下列可演算的指令，但不包含進位旗標，基本上來說，執行加法運算 / 減法運算時，可使用這一類指令。

+、+L、+B、+BL、-、-L、-B、-BL、

# WDT

指令名稱	指令記號	指令的各種組合	Fun No.	功能
設定週期監控時間	WDT	@WDT	094	僅能針對執行本指令的週期，延長其週期監控時間。

符號	WDT	
		S: 延長週期監控時間資料

## 是否適用於特定區域

區域	工程步進程式	副程式	中斷任務程式
使用	○	○	○

## 運算元種類的說明

運算元種類	內容	資料型態	容量
S	延長週期監控時間資料	僅限於常數	1

S: 延長週期監控時間資料

10 進位 &0 ~ 0100 或 16 進位 #0000 ~ 0064

### ● 運算元種類

內容	CH 位址							間接 DM		常數	條件旗標	時鐘脈衝	TR
	CIO	WR	HR	AR	T	C	DM	@DM	* DM				
S	-	-	-	-	-	-	-	-	-	○	-	-	-

## 條件旗標的動作

名稱	標籤	內容
異常旗標	P_ER	<ul style="list-style-type: none"> <li>當週期監控時間超過 1s 時，本旗標就會 ON</li> <li>否則為 OFF</li> </ul>

## PLC 系統相關設定

- PLC 系統設定時 (使用 CX-Programmer 時)



- 設定 PLC 系統

名稱	內容	設定
設定週期監控時間	當週期時間超過所設定的監控時間時，就會登錄週期時間超過旗標 ( 運轉停止異常 )。	0：初始值 (1,000 ms) 1：設定任意時間
	設定週期監控時間 ( 僅適用於上述設定被設定為 1：任意時間 )	#0001~0FA0： 10 ~ 1,000 ms (10ms 單位)

(註)

- 利用 PLC 系統設定，即可在 10 ~ 1,000 ms (10 ms 單位) 的範圍內設定週期監控時間，初始值為 1000 ms (1s)。
- 本指令可在同一個週期中重複使用，此時，每次的延長時間將會被累計。不過，累計值不得超過 1,000 ms (1s)。
- 欲執行本指令時，只要累計值已經到達 1,000 ms (1s)，就無法再執行本指令。

## 相關特殊輔助繼電器

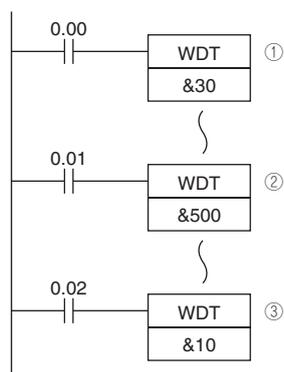
名稱	位址	內容
週期時間超過旗標	A401.08	當週期時間的現在值超過 PLC 系統所設定的「週期監控時間」時，本旗標就會變成 1 (ON)。CPU 模組停止運轉。
週期時間的最大值	A262 ~ A263	週期時間的最大值將會以 BIN32 的格式，依週期別儲存。
週期時間的現在值	A264 ~ A265	週期時間的現在值將會以 BIN32 的格式，依週期別儲存。

## 功能

只有在本指令的執行週期期間，才會利用 PLC 系統設定，將 S 所指定的資料 x 10 ms (0 ~ 1,000 ms) 延長至所指定的週期監控時間中。

欲暫時增加處理內容並延長週期時間時，利用本指令即可避免發生週期時間超過。

## 程式例



### WDT 指令的動作

以上為週期監控時間設定值被設定為 500 ms 時的範例。

- 當輸入繼電器 0.00 變成 ON 時，即利用①的 WDT 指令，將週期監控時間延長相當於 300 ms (數值為 30，因此可求出 30 x 10 ms)。因此，該時點的週期監控時間累計值為 800 ms (500 ms + 300 ms)。
- 當輸入繼電器 0.01 變成 ON 時，即利用②的 WDT 指令，將週期監控時間再延長 500 ms (數值為 50，因此可求出 50 x 10 ms)。不過，這時候累計值將變成 1,300 ms，因此超過週期監控時間上下值 1000 ms 的部分，亦即 300 ms 會被忽略，結果就是由該時點開始延長 200 ms。
- 當輸入繼電器 0.02 變成 ON 時，即利用③的 WDT 指令，將週期監控時間再延長 10 ms (數值為 10，因此可求出 10 x 10 ms)。不過，由於累計值已經到達 1,000 ms，因此不會執行③的 WDT 指令。

# 3

## 指令處理時間 /STEP 數

3

本章將說明 CP1E CPU 裝置各種指令的處理時間。

---

3-1CP1E 系列指令處理時間 /STEP 數 .....	3-2
--------------------------------	-----

## 3-1 CP1E 系列指令處理時間 /STEP 數

以下將介紹各種指令記號的處理時間。

在 1 個階梯程式 ( 註 ) 中各種指令記號的處理時間總和即為週期中的「演算處理時間」。

( 註 ) 正確來說，就是執行週期任務以及被配置為中斷條件成立的中斷任務之程式。

指令與的處理時間依指令執行時的條件而異。

而且，不執行指令時，仍會佔用處理時間，此點需特別注意。

同時還代表指令長度 (STEP)。亦即包含運算元種類的各種指令佔用多少階梯程式區的意思。STEP ( 步數 ) 依指令記號及表現格式而異。又，請注意 STEP 並不代表指令記號的數量。

( 註 ) 1 除了部分指令記號外，執行微分時必須附加 ( 基本指令為「↑↓」、應用指令為「@%」) 的條件。此時的處理時間需加上下列時間。

( 單位：μ 秒 )

條件	CP1 CPU 模組
	CPU □□
附加微分 ( ↑ ↓ ) 時	+4.0
附加微分 ( @ % ) 時	+2.5

2 不執行指令時的處理時間基準如下：

CP1 CPU 模組
CPU □□
1.4

## 順序控制指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
LOAD	LD	-	1	1.19	-
	!LD	-	2	10.26	-
LOAD NOT	LD NOT	-	1	1.19	-
	!LD NOT	-	2	10.26	-
AND	AND	-	1	1.19	-
	!AND	-	2	10.26	-
AND/NOT	AND NOT	-	1	1.19	-
	!AND NOT	-	2	10.26	-
OR	OR	-	1	1.29	-
	!OR	-	2	10.36	-
OR/NOT	OR NOT	-	1	1.29	-
	!OR NOT	-	2	10.36	-
AND/LOAD	AND LD	-	1	0.60	-
OR/LOAD	OR LD	-	1	0.60	-
NOT	NOT	520	1	0.80	-
P. F. 上升微分	UP	521	3	4.92	-
P. F. 下降微分	DOWN	522	4	5.69	-

## 時序輸出指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
輸出	OUT	-	1	1.61	-
	!OUT	-	2	38.06	-
輸出否定	OUT NOT	-	1	1.61	-
	!OUT NOT	-	2	38.06	-
KEEP	KEEP	011	1	4.72	-
上升微分	DIFU	013	2	4.12	-
下降微分	DIFD	014	2	4.19	-
設定	SET	-	1	2.69	-
	!SET	-	2	39.12	-
復歸	RSET	-	1	2.69	指定 CH
	!RSET	-	2	39.12	-
多位元設定	SETA	530	4	17.60	1 bit 設定
				253.5	1000 bit 設定
多位元復歸	RSTA	531	4	17.60	1 bit 設定
				249.5	1000 bit 設定
1 位元組	SETB	532	2	16.60	-
	!SETB		3	54.60	-
1 位元復歸	RSTB	534	2	16.60	-
	!RSTB		3	54.60	-

## 序列控制指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
END	END	001	1	4.6	-
無功能	NOP	000	1	1.2	-
互鎖	IL	002	1	4.3	-
清除互鎖	ILC	003	1	4.3	-
多重互鎖 (微分旗標維持型)	MILH	517	3	19.4	在 IL 狀態下執行時
				19.4	在非 IL 狀態下執行時，則不執行 IL
				21.5	在非 IL 狀態下執行時，將會執行 IL
多重互鎖 (非微分旗標維持型)	MILR	518	3	19.4	在 IL 狀態下執行時
				19.4	在非 IL 狀態下執行時，則不執行 IL
				21.5	在非 IL 狀態下執行時，將會執行 IL
清除多重互鎖	MILC	519	2	8.9	解除 IL 時除外
				8.9	IL 解除時
跳躍	JMP	004	2	6.1	-
跳躍結束 (Jump End)	JME	005	2	6.2	-
條件跳躍 (Jump)	CJP	510	2	10.1	JMP 條件成立時
開始反覆動作	FOR	512	2	9.7	指定常數
跳離迴圈 (Loop break)	BREAK	514	1	4.1	-
結束反覆動作	NEXT	513	1	5.8	持續迴圈時
				5.7	結束迴圈時

## 計時器 / 計數器指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
計時器	TIM	-	3	11.6	-
	TIMX	550			持續迴圈時
計數器	CNT	-	3	11.5	-
	CNTX	546			持續迴圈時
高速計時器	TIMH	015	3	11.6	-
	TIMHX	551			持續迴圈時
超高速計時器	TMHH	540	3	10.8	-
	TMHHX	552			-
累加計時器	TTIM	087	3	22.7	-
				17.4	復歸時
				15.0	IL 時
	TTIMX	555	3	22.2	-
				17.4	復歸時
				15.2	IL 時
長時間計時器	TIML	542	4	24.3	-
				20.4	IL 時
	TIMLX	553	4	24.5	-
				22.2	IL 時
可逆計數器	CNTR	012	3	26.2	-
	CNTRX	548		25.4	-
復歸計時器 / 計數器	CNR	545	3	19.0	1 CH 復歸
				659.0	256 CH 復歸
	CNRX	547	3	19.0	1 CH 復歸
				659.0	256 CH 復歸

## 資料比較指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
符號比較 (不帶符號)	LD,AND,OR+ =	300	4	9.3	-
	LD,AND,OR+ <>	305			
	LD,AND,OR+ <	310			
	LD,AND,OR+ < =	315			
	LD,AND,OR+ >	320			
	LD,AND,OR+ > =	325			
符號比較 (倍精確度・不帶符號)	LD,AND,OR+ =+L	301	4	10.8	-
	LD,AND,OR+ <> +L	306			
	LD,AND,OR+ < +L	311			
	LD,AND,OR+ < =+L	316			
	LD,AND,OR+ > +L	321			
	LD,AND,OR+ > =+L	326			
符號比較 (附帶符號)	LD,AND,OR+ =+S	302	4	9.4	-
	LD,AND,OR+ <> +S	307			
	LD,AND,OR+ < +S	312			
	LD,AND,OR+ < =+S	317			
	LD,AND,OR+ > +S	322			
	LD,AND,OR+ > =+S	327			
符號比較 (倍精確度・附帶符號)	LD,AND,OR+ =+SL	303	4	10.9	-
	LD,AND,OR+ <> +SL	308			
	LD,AND,OR+ < +SL	313			
	LD,AND,OR+ < =+SL	318			
	LD,AND,OR+ > +SL	323			
	LD,AND,OR+ > =+SL	328			
時刻比較	=DT	341	4	14.5	-
	<>DT	342	4	14.5	-
	<DT	343	4	14.4	-
	<=DT	344	4	14.4	-
	>DT	345	4	14.6	-
	>=DT	346	4	14.6	-
比較	CMP	020	3	8.1	-
	!CMP	020	7	49.1	-
倍精確度比較	CMPL	060	3	9.5	-
帶符號 BIN 比較	CPS	114	3	8.1	-
	!CPS	114	7	49.1	-
帶符號 BIN 倍精確度比較	CPSL	115	3	9.5	-
比較表一致	TCMP	085	4	61.1	-
不帶符號資料表比較	BCMP	068	4	107.6	-
區域比較	ZCP	088	3	17.8	-
倍精確度區域比較	ZCPL	116	3	20.1	-

## 資料傳送指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
傳送	MOV	021	3	8.0	-
	!MOV	021	7	57.7	-
傳送倍精確度	MOVL	498	3	8.9	-
否定傳送	MVN	022	3	13.7	-
位元傳送	MOVB	082	4	21.4	-
位數傳送	MOVD	083	4	22.4	-
多位數傳送	XFRB	062		26.4	1 CH 傳送
				137.3	255 bit 傳送
區塊傳送	XFER	070	4	24.2	1 CH 傳送
				3747.7	1000 CH 傳送
區塊設定	BSET	071	4	21.3	1 CH 設定
				2074.4	1000 CH 設定
資料交換	XCHG	073	3	19.2	-
資料配置	DIST	080	4	20.8	-
資料擷取	COLL	081	4	20.6	-

## 資料移位指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
移位暫存器	SFT	010	3	14.1	1 CH 位移
				1076.0	290 CH 位移
左右移位暫存器	SFTR	084	4	18.0	1 CH 位移
				3784.4	1000 CH 位移
字元移位	WSFT	016	4	25.8	1 CH 位移
				3783.9	1000 CH 位移
向左移動 1 位元	ASL	025	2	13.0	-
向右移動 1 位元	ASR	026	2	13.0	-
附 CY 的 1 位元向左回轉	ROL	027	2	13.3	-
附 CY 的 1 位元向右回轉	ROR	028	2	13.5	-
向左移動 1 位數	SLD	074	3	21.8	1 CH 位移
				3778.3	1000 CH 位移
向右移動 1 位數	SRD	075	3	22.2	1 CH 位移
				3778.6	1000 CH 位移
向左移動 N 位元	NASL	580	3	19.5	-
向左倍精確度移動 N 位元	NSLL	582	3	20.8	-
向右移動 N 位元	NASR	581	3	19.6	-
向左倍精確度移動 N 位元	NSRL	583	3	21.0	-

## 加 / 減指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
BIN 加法運算	++	590	2	12.3	-
BIN 倍精確度遞增	++L	591	2	13.5	-
BIN 減法運算	--	592	2	12.3	-
BIN 倍精確度遞減	--L	593	2	13.6	-
BCD 遞增	++B	594	2	13.2	-
BCD 倍精確度遞增	++BL	595	2	14.4	-
BCD 遞減	--B	596	2	13.2	-
BCD 倍精確度遞減	--BL	597	2	14.5	-

## 四則運算指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
帶符號·無 CY 的 BIN 加法運算	+	400	4	11.5	-
帶符號·無 CY 的 BIN 倍精確度加法運算	+L	401	4	13.0	-
帶符號·CY 的 BIN 加法運算	+C	402	4	11.7	-
帶符號·CY 的 BIN 倍精確度加法運算	+CL	403	4	13.2	-
不含 CY 的 BCD 加法運算	+B	404	4	20.6	-
不含 CY 的 BCD 倍精確度加法運算	+BL	405	4	22.9	-
含 CY 的 BCD 加法運算	+BC	406	4	20.8	-
含 CY 的 BCD 倍精確度加法運算	+BCL	407	4	23.1	-
帶符號·無 CY 的 BIN 減法運算	-	410	4	11.6	-
帶符號·無 CY 的 BIN 倍精確度減法運算	-L	411	4	13.2	-
帶符號·CY 的 BIN 減法運算	-C	412	4	11.7	-
帶符號·CY 的 BIN 倍精確度減法運算	-CL	413	4	13.3	-
不含 CY 的 BCD 減法運算	-B	414	4	20.3	-
不含 CY 的 BCD 倍精確度減法運算	-BL	415	4	23.6	-
含 CY 的 BCD 減法運算	-BC	416	4	20.5	-
含 CY 的 BCD 倍精確度減法運算	-BCL	417	4	23.8	-
帶符號 BIN 乘法運算	*	420	4	18.4	-
帶符號 BIN 倍精確度乘法運算	* L	421	4	23.9	-
BCD 乘法運算	* B	424	4	22.0	-
BCD 倍精確度乘法運算	* BL	425	4	33.2	-

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
帶符號 BIN 除法運算	/	430	4	19.8	-
帶符號 BIN 倍精確度除法運算	/L	431	4	25.8	-
BCD 除法運算	/B	434	4	23.2	-
BCD 倍精確度除法運算	/BL	435	4	33.0	-

## 資料轉換指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
BCD → BIN 轉換	BIN	023	3	15.1	-
BCD → BIN 倍精確度轉換	BINL	058	3	16.7	-
BIN → BCD 轉換	BCD	024	3	15.1	-
BIN → BCD 倍精確度轉換	BCDL	059	3	17.3	-
轉換 2 的補數	NEG	160	3	14.3	-
4 → 16/8 → 256 指令碼	MLPX	076	4	19.6	解碼由 4 → 16 時 / 位數為 1 位數
				31.0	解碼由 4 → 16 時 / 位數為 4 位數
				79.4	解碼由 8 → 256 時 / 位數為 1 位數
				138.2	解碼由 8 → 256 時 / 位數為 4 位數
16 → 4/256 → 8 指令碼	DMPX	077	4	32.5	解碼由 16 → 4 時 / 位數為 1 位數
				63.0	解碼由 16 → 4 時 / 位數為 4 位數
				68.0	解碼由 256 → 8 時 / 位數為 1 位數
				112.3	解碼由 256 → 8 時 / 位數為 2 位數
ASCII 碼轉換	ASC	086	4	22.8	1 位數 ASCII 碼轉換
				24.7	4 位數 ASCII 碼轉換
ASCII → HEX 轉換	HEX	162	4	18.4	轉換 1 位數

## 邏輯演算指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
字元邏輯積	ANDW	034	4	18.6	-
字元倍精確度邏輯積	ANDL	610	4	20.4	-
字元邏輯和	ORW	035	4	18.6	-
字元倍精確度邏輯和	ORWL	611	4	20.4	-
字元互斥或 (Exclusive OR: XOR) 演算	XORW	036	4	18.6	-
字元倍精確度 XOR	XORL	612	4	20.4	-
位元反相	COM	029	2	12.4	-
位元倍精確度反相	COML	614	2	13.6	-

## 特殊演算指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
數值轉換	APR	069	4	34.2	指定 SIN、COS 計算
				25.9	指定線性近似計算
位元計數器	BCNT	067	4	19.5	1 CH 計數

## 浮動小數點轉換・演算指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
浮動小數點→轉換為 16 位元 BIN	FIX	450	3	15.9	-
浮動小數點→轉換為 32 位元 BIN	FIXL	451	3	16.2	-
16 位元 BIN →轉換為浮動小數點	FLT	452	3	16.2	-
32 位元 BIN →轉換為浮動小數點	FLTL	453	3	17.1	-
浮動小數點加法運算	+F	454	4	24.1	-
浮動小數點減法運算	-F	455	4	25.2	-
浮動小數點除法運算	/F	457	4	25.0	-
浮動小數點乘法運算	* F	456	4	24.4	-
單精確度的浮動小數點資料比較	LD、AND、OR+ = F	329	3	11.6	-
	LD、AND、OR+ <> F	330			
	LD、AND、OR+ < F	331			
	LD、AND、OR+ <= F	332			
	LD、AND、OR+ > F	333			
	LD、AND、OR+ >= F	334			
浮動小數點 < 單精確度 > →轉換為文字列	FSTR	448	4	56.8	-
文字列→浮動小數點 < 單精確度 > 轉換	FVAL	449	3	42.9	-

## 資料表執行資料處理時的指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
位元置換	SWAP	637	3	16.8	取代 1 CH
				6250.0	取代 1000 CH
計算 FCS 值	FCS	180	4	24.1	資料長度為 1 字元
				2710.0	資料長度為 1000 字元

## 資料控制指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
附自動微調功能的 PID 演算	PIDAT	191	4	316.0	PID 演算時 第一次執行本指令
				270.0	PID 演算時 取樣時
				228.0	PID 演算時 非取樣時
				275.5	AT 時 第一次執行本指令
				276.0	AT 時 取樣時
時間分割比例輸出	TPO	685	4	5.8	PF=OFF
				40.8	指定負載比為 PF = ON or 關閉輸出限制
				43.4	指定操作量為 PF = ON & 啟動輸出限制
轉換	SCL	194	4	24.8	-
轉換 2	SCL2	486	4	20.2	-
轉換 3	SCL3	487	4	26.4	-
資料平均	AVG	195	4	24.2	1 次平均
				225.5	64 次平均

## 副程式的指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
呼叫副程式	SBS	091	2	6.6	-
副程式進入	SBN	092	2	2.6	-
副程式返回	RET	093	1	3.1	-

## 中斷控制指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
設定中斷遮罩	MSKS	690	3	15.1	設定
				15.1	復歸
中斷解除	CLI	691	3	14.9	設定
				18.0	復歸
禁止執行中斷任務	DI	693	1	8.5	-
禁止執行中斷任務的解除	EI	694	1	8.9	-

## 高速計數器 / 脈衝輸出指令

指令名稱	指令記號	FUN No.	指令長度 (STEP) * 1	執行時 (μs)	執行條件
動作模式控制	INI	880	4	46.0	開始高速計數值比較
				31.8	停止高速計數值比較
				48.7	變更脈衝輸出現在值
				35.2	變更高速脈衝輸出現在值
				27.2	停止脈衝輸出
				13.0	停止 PWM 輸出
讀取脈衝現在值	PRV	881	4	40.0	讀取脈衝輸出現在值
				35.0	讀取高速脈衝現在值
				37.2	讀取脈衝輸出狀態
				32.6	讀取高速脈衝輸出狀態
				24.5	讀取 PWM 狀態
				36.5	讀取高速計數值的頻域比較結果
登錄比較資料表	CTBL	882	4	69.3	登錄目標一致比較表 & 開始比較 (1 點)
				116.3	登錄目標一致比較表 & 開始比較 (6 點)
				126.6	登錄頻域比較表 & 開始比較
				46.3	僅限於登錄目標一致比較表 (1 點)
				93.3	僅限於登錄目標一致比較表 (6 點)
				122.5	僅登錄頻域比較表
頻率設定	SPED	885	4	69.2	連續模式
				74.0	單一模式
脈衝量設定	PULS	886	4	44.1	-
定位	PLS2	887	5	97.6	-
頻率加減速控制	ACC	888	4	75.6	連續模式
				82.8	單一模式
原點搜尋	ORG	889	3	52.2	原點搜尋
				126.8	原點復歸
PWM 輸出	PWM	891	4	28.9	-

## 製程步進控制指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
定義步進階梯區	STEP	008	2	10.5	步進繼電器 ON 時
				10.4	步進繼電器 OFF 時
進入步進階梯區	SNXT	009	2	9.6	-

## I/O 模組專用指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
I/O 更新狀態	IORF	097	3	170.7	對象裝置 (CP1W 系列擴充裝置) 1 CH IN
				146.6	對象裝置 (CP1W 系列擴充裝置) 1 CH OUT
				1725.8	對象裝置 (CP1W 系列擴充裝置) 12 CH IN
				1359.9	對象裝置 (CP1W 系列擴充裝置) 12 CH OUT
7 段解碼器	SDEC	078	4	21.9	-
矩陣輸入	MTR	213	5	31.6	資料輸入值 : 00
				31.6	資料輸入值 : FF
7 段顯示	7SEG	214	5	27.1	指定為 4 位數時
				30.8	指定為 8 位數時

## 序列通訊指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
序列埠輸出	TXD	236	4	25.0	傳送的位元數為 1 位元
				25.0	傳送的位元數為 256 位元
序列埠輸入	RXD	235	4	39.2	儲存的位元數為 1 位元
				256.6	儲存的位元數為 256 位元

## 時鐘功能指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
日曆加法運算	CADD	730	4	56.6	-
日曆減法運算	CSUB	731	4	55.1	-
修正時鐘	DATE	735	2	29.9	-

## 故障診斷指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
運轉持續故障診斷	FAL	006	3	55.6	登錄異常
				79.6	解除異常 (最優先)
				61.6	解除異常 (所有異常)
				60.0	解除異常 (個別)
運轉停止故障診斷	FALS	007	3	-	-

## 特殊指令

指令名稱	指令記號	FUN No.	指令長度 (STEP)	執行時 (μs)	執行條件
設定進位	STC	040	1	32.6	-
清除進位	CLC	041	1	3.9	-
設定週期監控時間	WDT	094	2	11.7	-

# 4

## 監控及計算週期時間的方法

本章將針對程式中所能夠使用的 CP1E CPU 裝置的週期時間監控方法及計算方法加以說明。

4

---

4-1 監控週期時間 .....	4-2
4-1-1 監控週期時間方法 .....	4-2
4-2 週期時間的計算方法 .....	4-3
4-2-1 PLC 的動作流程 .....	4-3
4-2-2 週期時間計算概要 .....	4-4
4-2-3 各裝置的 I/O 更新時間 .....	4-5
4-2-4 週期時間的計算範例 .....	4-6
4-2-5 執行線上編輯時的週期延遲時間 .....	4-6

## 4-1 監控週期時間

### 4-1-1 監控週期時間方法

使用 CX-Programmer，即可在連線狀態下，監控週期時間的平均值、最大值或最小值。

#### 監控平均值

在連線狀態下，除非 CPU 裝置處於「程式」模式，否則週期時間的平均值皆會顯示在狀態列上。

PLC_Name(ネット:0ノード:0) - モニタモード **0.4 ms** モニター同期

#### 監控最大值、最小值

請利用 [PLC] 選單，依序選擇 [PLC 設定]-[PLC 資訊]-[週期時間] 選。  
畫面上將顯示 [PLC 週期時間] 的對話框。

The screenshot shows a dialog box titled "PLCのサイクルタイム - 新規PLC1". It is divided into two main sections: "サイクルタイム" and "実行時間".

- サイクルタイム (Cycle Time):**
  - 平均 (Average): 0.4 ms
  - 最大 (Maximum): 0.8 ms
  - 最小 (Minimum): 0.4 ms
  - There is a button labeled "リセット(R)" (Reset).
- 実行時間 (Execution Time):**
  - 開始マーク番号(S) (Start Mark Number): [Input field]
  - 終了マーク番号(I) (End Mark Number): [Input field]
  - 時間 (Time): [Input field]
  - Buttons: "測定(M)" (Measure), "中止(A)" (Stop), and "閉じる" (Close).

由上而下分別是週期時間的平均值、最大值、最小值。  
點擊 [復歸] 鍵，即可再次測量並顯示週期時間。



#### 參考

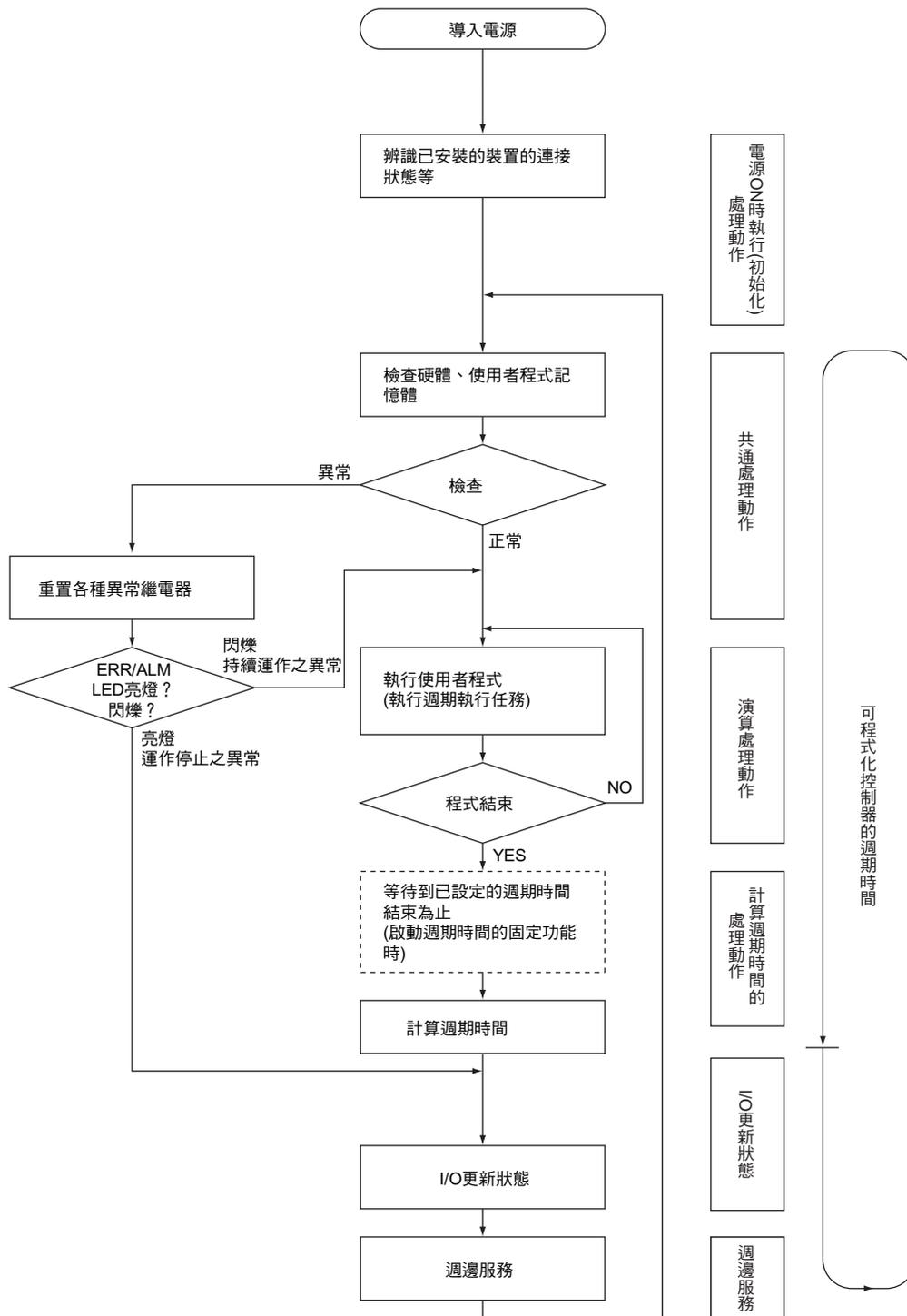
週期時間的現在值及週期時間的最大值會被儲存在以下的特殊輔助繼電器中。

- 週期時間的現在值 (0.1 ms 單位) : A264/A265CH (A264 : 下位、A265 上位)
- 週期時間的最大值 (0.1 ms 單位) : A262/A263CH (A262 : 下位、A263 上位)

## 4-2 週期時間的計算方法

### 4-2-1 PLC 的動作流程

CPU 裝置會將圖中的共通處理 ~ 週邊服務之間的處理動作當作 1 個週期，並且反覆進行週期動作。



## 4-2-2 週期時間計算概要

週期時間依下列要素而改變：

- 整個階梯條件 (週期執行任務及執行條件成立的中斷任務中) 的指令種類及數量
- CP 系列擴充 I/O 裝置和擴充裝置的種類及連接數量
- PLC 系統所設定的「固定週期時間」
- 是否使用平行埠 (USB 連接埠)、序列埠



### 使用上的注意事項

將動作模式由「監控」模式切換至「運轉」模式時，有可能造成週期時間延長 10 ms 的情形 (不過，不會導致週期時間超過)。

週期時間為執行下列各處理時的處理時間總和。

週期時間 = (1) + (2) + (3) + (4) + (5)

#### (1) 共通處理動作

處理內容	處理時間及變動要因
I/O 匯流排檢查 檢查使用者記憶體、檢查電池異常等	固定為 0.4 ms

#### (2) 演算處理動作

處理內容	處理時間及變動要因
處理使用者程式 (執行指令)。指令記號的執行時間總和。	指令執行時間的總和

#### (3) 固定週期時間、計算處理

處理內容	處理時間及變動要因
利用 PLC 系統設定，將週期時間指定為固定時間時，必須等到到達所指定的週期時間為止。計算週期時間。	不執行固定處理時的時間趨近於 0。 固定處理所需的追加時間 指定為固定週期時間 - 實際的週期時間 ((1)+(2)+(4)+(5))

#### (4) I/O 更新狀態

處理內容	處理時間及變動要因
CPU 裝置內建的輸出輸入、CPU 裝置內建的類比輸出輸入 (僅適用於 NA 型)、CP 系列擴充 I/O 裝置、擴充裝置。	依裝置別，OUT (CPU 裝置→裝置) 後依 IN (裝置→CPU 裝置) 的順序進行更新。 各裝置的 I/O 更新時間 x 使用台數

#### (5) 週邊服務

處理內容	處理時間及變動要因
平行埠 (USB 埠) 服務	此項服務的執行時間為 (3) 所計算出來的前一個週期時間的 8%。
序列埠服務 (內建 RS-232C 埠、序列選擇埠)	

## 4-2-3 各裝置的 I/O 更新時間

## ● CP1E NA 型 CPU 裝置內建的類比輸出輸入的更新時間

裝置名稱	型式	I/O 更新時間
CP1E CPU 模組 NA 型	CP1E-NA □□ D □ - □	0.5ms

(註) 此時間為內建的類比輸出輸入的 I/O 更新時間，內建的輸出輸入的更新時間已經包含在共通處理的時間中。  
不使用內建的類比輸出輸入功能時，仍然會加上此時間計算。

## ● CP 系列擴充 (I/O) 裝置的更新時間

裝置名稱	型式	每裝置的 I/O 更新時間	
擴充 I/O 裝置	8 點輸入型	CP1W-8ED	0.14ms
	8 點輸出型	CP1W-8ER	0.06ms
		CP1W-8ET	
		CP1W-8ET1	
	16 點輸出型	CP1W-16ER	0.17ms
		CP1W-16ET	
		CP1W-16ET1	
	20 點輸入輸出型	CP1W-20EDR1	0.20ms
		CP1W-20EDT	
		CP1W-20EDT1	
	32 點輸出型	CP1W-32ER	0.33ms
		CP1W-32ET	
CP1W-32ET1			
40 點輸入輸出型	CP1W-40EDR	0.45ms	
	CP1W-40EDT		
	CP1W-40EDT1		
擴充裝置	類比輸入裝置	CP1W-AD041	0.72ms
	類比輸出裝置	CP1W-DA041	0.33ms
	類比輸入輸出裝置	CP1W-MAD11	0.36ms
	溫度感測器裝置	CP1W-TS001	0.30ms
		CP1W-TS002	0.57ms
		CP1W-TS101	0.30ms
		CP1W-TS102	0.57ms
CompoBus/SI/O 連接裝置	CP1W-SRT21	0.20ms	



## 參考

對應至 CPU 裝置所內建的輸出輸入的更新時間已經包含在共通處理時間中。

### 4-2-4 週期時間的計算範例

接下來將說明連接至 CP1E CPU 模組的 I/O 模組時，其週期時間的計算方法。

#### ● 條件

項目	內容	
CP1 CPU 模組	40 點輸出單元 CP1W-40EDR	1 台
階梯圖程式	5K STEP	LD 指令 2.5K STEP、 OUT 指令 2.5K STEP
連接平行埠 (USB 埠)	有或無	
固定週期時間處理	無	
連接序列埠	無	
其他週邊服務	無	

#### ● 計算範例

處理名稱	算式	處理時間	
		將週邊工具連接至 平行埠 (USB 埠) 時	將週邊工具連接至 平行埠 (USB 埠) 時
(1) 共通處理動作	—	0.4ms	0.4ms
(2) 演算處理動作	$1.19 \mu s \times 2,500 + 1.61 \mu s \times 2,500$	7.0ms	7.0ms
(3) 固定週期時間、計算處理	( 固定週期時間處理 )	0ms	0ms
(4) I/O 更新狀態	0.45ms	0.45ms	0.45ms
(5) 週邊服務	( 僅連接平行埠 (USB 埠) )	0.2ms	0ms
週期時間	(1) + (2) + (3) + (4) + (5)	8.15ms	7.85ms

### 4-2-5 執行線上編輯時的週期延遲時間

運轉時 ( 監控模式下 )，若利用 CX-Programmer 進行線上編輯，並執行已編輯程式的寫入動作時，CPU 模組的將會暫時中斷運轉，並且在中斷過程中寫入程式。利用 CPU 模組的程式寫入處理，以延長週期時間。又，CPU 模組處理中斷運轉狀態下，亦不會執行中斷任務。

若程式的大小為 8K STEP 時，週期延遲時間如下所示。

CPU 裝置	執行線上編輯時的週期延遲時間
CP1 CPU 模組	最大 16 ms，一般約 6 ms ( 不過，程式大小為 8K Step )

執行線上編輯時，週期時間將因延長時間而變長，而且會造成時間中斷任務延遲執行，此點需特別注意。

# 附

## 附錄

附

---

指令英文字母排序表 .....	附 -2
-----------------	------

# 指令英文字母排序表

## A

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
ACC	頻率控制	888	@ACC	無	無	2-311
AND	AND	無	@AND	%AND	!AND	2-9
AND LD	AND/LOAD	無	無	無	無	2-13
AND NOT	AND/NOT	無	無	無	!AND NOT	2-9
AND<	AND 型・小於	310	無	無	無	2-86
AND<=	AND 型・以下	315	無	無	無	2-86
AND<=F	AND 型・單精確度浮動小數點・以下	332	無	無	無	2-230
AND<=DT	AND 型・時刻・以下	344	無	無	無	2-89
AND<=L	AND 型・倍精確度・以下	316	無	無	無	2-86
AND<=S	AND 型・帶符號・以下	317	無	無	無	2-86
AND<=SL	AND 型・帶符號倍精確度・以下	318	無	無	無	2-86
AND<>	AND 型・不一致	305	無	無	無	2-86
AND<>DT	AND 型・時刻・不一致	342	無	無	無	2-89
AND<>F	AND 型・單精確度浮動小數點・不一致	330	無	無	無	2-230
AND<>L	AND 型・倍精確度・不一致	306	無	無	無	2-86
AND<>S	AND 型・帶符號・不一致	307	無	無	無	2-86
AND<>SL	AND 型・帶符號倍精確度・不一致	308	無	無	無	2-86
AND<DT	AND 型・時刻・小於	343	無	無	無	2-89
AND<F	AND 型・單精確度浮動小數點・小於	331	無	無	無	2-230
AND<L	AND 型・倍精確度・小於	311	無	無	無	2-86
AND<S	AND 型・帶符號・小於	312	無	無	無	2-86
AND<SL	AND 型・帶符號倍精確度・小於	313	無	無	無	2-86
AND=	AND 型・一致	300	無	無	無	2-86
AND=DT	AND 型・時刻・一致	341	無	無	無	2-89
AND=F	AND 型・單精確度浮動小數點・一致	329	無	無	無	2-230
AND=L	AND 型・倍精確度・一致	301	無	無	無	2-86
AND=S	AND 型・帶符號・一致	302	無	無	無	2-86
AND=SL	AND 型・帶符號倍精確度・一致	303	無	無	無	2-86

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
AND>	AND 型・超過	320	無	無	無	2-86
AND>=	AND 型・以上	325	無	無	無	2-86
AND>=DT	AND 型・時刻・以上	346	無	無	無	2-89
AND>=F	AND 型・單精確度浮動小數點・以上	334	無	無	無	2-230
AND>=L	AND 型・倍精確度・以上	326	無	無	無	2-86
AND>=S	AND 型・帶符號・以上	327	無	無	無	2-86
AND>=SL	AND 型・帶符號倍精確度・以上	328	無	無	無	2-86
AND>DT	AND 型・時刻・超過	345	無	無	無	2-89
AND>F	AND 型・單精確度浮動小數點・超過	333	無	無	無	2-230
AND>L	AND 型・倍精確度・超過	321	無	無	無	2-86
AND>S	AND 型・帶符號・超過	322	無	無	無	2-86
AND>SL	AND 型・帶符號倍精確度・超過	323	無	無	無	2-86
ANDL	字元倍精確度邏輯積	610	@ANDL	無	無	2-201
ANDW	字元邏輯積	034	@ANDW	無	無	2-201
APR	數值轉換	069	@APR	無	無	2-209
ASC	ASCII 碼轉換	086	@ASC	無	無	2-193
ASL	向左移動 1 位元	025	@ASL	無	無	2-129
ASR	向右移動 1 位元	026	@ASR	無	無	2-130
AVG	資料平均	195	無	無	無	2-271

## B

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
BCD	BIN → BCD 轉換	024	@BCD	無	無	2-179
BCDL	BIN → BCD 倍精確度轉換	059	@BCDL	無	無	2-179
BCMP	不帶符號資料表比較	068	@BCMP	無	無	2-100
BCNT	位元計數器	067	@BCNT	無	無	2-218
BIN	BCD → BIN 轉換	023	@BIN	無	無	2-177
BINL	BCD → BIN 倍精確度轉換	058	@BINL	無	無	2-177
BREAK	跳出迴圈	514	無	無	無	2-57
BSET	區塊設定	071	@BSET	無	無	2-115

## C

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
CADD	日曆加法運算	730	@CADD	無	無	2-356
CJP	條件跳躍 (Jump)	510	無	無	無	2-51
CLC	清除進位	041	@CLC	無	無	2-372
CLI	中斷解除	691	@CLI	無	無	2-284
CMP	符號比較	020	無	無	ICMP	2-93
CMPL	符號倍精確度比較	060	無	無	無	2-93
CNR	復歸計時器 / 計數器	545	@CNR	無	無	2-84
CNRX	復歸計時器 / 計數器	547	@CNRX	無	無	2-84
CNT	計數器	無	無	無	無	2-78
CNTR	可逆計數器	012	無	無	無	2-81
CNTRX	可逆計數器	548	無	無	無	2-81
CNTX	計數器	546	無	無	無	2-78
COLL	資料擷取	081	@COLL	無	無	2-121
COM	位元反相	029	@COM	無	無	2-207
COML	位元倍精確度反相	614	@COML	無	無	2-207
CPS	帶符號 BIN 比較	114	無	無	ICPS	2-95
CPSL	帶符號 BIN 倍精確度比較	115	無	無	無	2-95
CSUB	日曆減法運算	731	@CSUB	無	無	2-356
CTBL	登錄比較資料表	882	@CTBL	無	無	2-297

## D

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
DATE	修正時鐘	735	@DATE	無	無	2-360
DI	禁止執行中斷任務	693	@DI	無	無	2-287
DIFD	下微分	014	無	無	!DIFD	2-27
DIFU	上微分	013	無	無	!DIFU	2-25
DIST	資料配置	080	@DIST	無	無	2-119
DMPX	16 → 4/256 → 8 指令碼	077	@DMPX	無	無	2-188
DOWN	P. F. 下微分	522	無	無	無	2-17
DSW	數位開關	210	無	無	無	2-335

## E

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
EI	禁止執行中斷任務的解除	694	無	無	無	2-288
END	END	001	無	無	無	2-37

## F

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
FAL	運轉持續故障診斷	006	@FAL	無	無	2-362
FALS	運轉停止故障診斷	007	無	無	無	2-367
FCS	計算 FCS 值	180	@FCS	無	無	2-243
FIX	浮動小數點 →轉換為 16 位元 BIN	450	@FIX	無	無	2-223
FIXL	浮動小數點 →轉換為 32 位元 BIN	451	@FIXL	無	無	2-223
FLT	16 位元 BIN →轉換為浮動小數點	452	@FLT	無	無	2-225
FLTL	32 位元 BIN →轉換為浮動小數點	453	@FLTL	無	無	2-225
FOR	開始反覆動作	512	無	無	無	2-54
FSTR	浮動小數點 < 單精確度 > →轉換為文字列	448	@FSTR	無	無	2-233
FVAL	文字列 → 浮動小數點 < 單精確度 > 轉換	449	@FVAL	無	無	2-237

## H

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
HEX	ASCII → HEX 轉換	162	@HEX	無	無	2-197

## I

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
IL	互鎖	002	無	無	無	2-39
ILC	清除互鎖	003	無	無	無	2-39
INI	動作模式控制	880	@INI	無	無	2-289
IORF	I/O 更新狀態	097	@IORF	無	無	2-331

## J

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
JME	跳躍結束 (Jump End)	005	無	無	無	2-51
JMP	跳躍	004	無	無	無	2-51

## K

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
KEEP	KEEP	011	無	無	!KEEP	2-21

## L

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
LD	LOAD	無	@LD	%LD	!LD	2-7
LD=DT	LD 型・時刻・一致	341	無	無	無	2-89
LD=S	LD 型・帶符號・一致	302	無	無	無	2-86
LD NOT	LOAD NOT	無	無	無	!LD NOT	2-7
LD<	LD 型・小於	310	無	無	無	2-86
LD<=	LD 型・以下	315	無	無	無	2-86
LD<=DT	LD 型・時刻・以下	344	無	無	無	2-89
LD<=F	LD 型・單精度浮動小數點・以下	332	無	無	無	2-230
LD<=L	LD 型・倍精度・以下	316	無	無	無	2-86
LD<=S	LD 型・帶符號・以下	317	無	無	無	2-86
LD<=SL	LD 型・帶符號倍精度・以下	318	無	無	無	2-86
LD<>	LD 型・不一致	305	無	無	無	2-86
LD<>DT	LD 型・時刻・不一致	342	無	無	無	2-89
LD<>F	LD 型・單精度浮動小數點・不一致	330	無	無	無	2-230
LD<>L	LD 型・倍精度・不一致	306	無	無	無	2-86
LD<>S	LD 型・帶符號・不一致	307	無	無	無	2-86
LD<>SL	LD 型・帶符號倍精度・不一致	308	無	無	無	2-86
LD<DT	LD 型・時刻・小於	343	無	無	無	2-89
LD<F	LD 型・單精度浮動小數點・小於	331	無	無	無	2-230
LD<L	LD 型・倍精度・小於	311	無	無	無	2-86
LD<S	LD 型・帶符號・小於	312	無	無	無	2-86
LD<SL	LD 型・帶符號倍精度・小於	313	無	無	無	2-86
LD=	LD 型・一致	300	無	無	無	2-86
LD=F	LD 型・單精度浮動小數點・一致	329	無	無	無	2-230
LD=L	LD 型・倍精度・一致	301	無	無	無	2-86
LD=SL	LD 型・帶符號倍精度・一致	303	無	無	無	2-86
LD>	LD 型・超過	320	無	無	無	2-86
LD>=	LD 型・以上	325	無	無	無	2-86
LD>=DT	LD 型・時刻・以上	346	無	無	無	2-89
LD>=F	LD 型・單精度浮動小數點・以上	334	無	無	無	2-230

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
LD>=L	LD 型・倍精度・以上	326	無	無	無	2-86
LD>=S	LD 型・帶符號・以上	327	無	無	無	2-86
LD>=SL	LD 型・帶符號倍精度・以上	328	無	無	無	2-86
LD>DT	LD 型・時刻・超過	345	無	無	無	2-89
LD>F	LD 型・單精度浮動小數點・超過	333	無	無	無	2-230
LD>L	LD 型・倍精度・超過	321	無	無	無	2-86
LD>S	LD 型・帶符號・超過	322	無	無	無	2-86
LD>SL	LD 型・帶符號倍精度・超過	323	無	無	無	2-86

## M

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
MILC	清除多重互鎖	519	無	無	無	2-42
MILH	多重互鎖(微分旗標維持型)	517	無	無	無	2-42
MILR	多重互鎖(非微分旗標維持型)	518	無	無	無	2-42
MLPX	4 → 16/8 → 256 指令碼	076	@MLPX	無	無	2-183
MOV	傳送	021	@MOV	無	!MOV	2-105
MOVB	位元傳送	082	@MOVB	無	無	2-107
MOVD	位數傳送	083	@MOVD	無	無	2-109
MOVL	傳送倍精度	498	@MOVL	無	無	2-105
MSKS	設定中斷遮罩	690	@MSKS	無	無	2-282
MTR	矩陣輸入	213	無	無	無	2-338
MVN	否定傳送	022	@MVN	無	無	2-105

## N

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
NASL	向左移動 N 位元	580	@NASL	無	無	2-137
NASR	向右移動 N 位元	581	@NASR	無	無	2-140
NEG	轉換2的補數	160	@NEG	無	無	2-181
NEXT	結束反覆動作	513	無	無	無	2-54
NOP	無功能	000	無	無	無	2-38
NOT	NOT	520	無	無	無	2-16
NSLL	向左倍精度移動 N 位元	582	@NSLL	無	無	2-137
NSRL	向左倍精度移動 N 位元	583	@NSRL	無	無	2-140

## O

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
OR	OR	無	@OR	%OR	IOR	2-11
ORG	原點搜尋	889	@ORG	無	無	2-315
OR LD	OR/LOAD	無	無	無	無	2-13
OR NOT	OR/NOT	無	無	無	IOR NOT	2-11
OR<	OR型·小於	310	無	無	無	2-86
OR<=	OR型·以下	315	無	無	無	2-86
OR<=DT	OR型·時刻·以下	344	無	無	無	2-89
OR<=F	OR型·單精確度浮動小數點·以下	332	無	無	無	2-230
OR<=L	OR型·倍精確度·以下	316	無	無	無	2-86
OR<=S	OR型·帶符號·以下	317	無	無	無	2-86
OR<=SL	OR型·帶符號倍精確度·以下	318	無	無	無	2-86
OR<>	OR型·不一致	305	無	無	無	2-86
OR<>DT	OR型·時刻·不一致	342	無	無	無	2-89
OR<>F	OR型·單精確度浮動小數點·不一致	330	無	無	無	2-230
OR<>L	OR型·倍精確度·不一致	306	無	無	無	2-86
OR<>S	OR型·帶符號·不一致	307	無	無	無	2-86
OR<>SL	OR型·帶符號倍精確度·不一致	308	無	無	無	2-86
OR<DT	OR型·時刻·小於	343	無	無	無	2-89
OR<F	OR型·單精確度浮動小數點·小於	331	無	無	無	2-230
OR<L	OR型·倍精確度·小於	311	無	無	無	2-86
OR<S	OR型·帶符號·小於	312	無	無	無	2-86
OR<SL	OR型·帶符號倍精確度·小於	313	無	無	無	2-86
OR=	OR型·一致	300	無	無	無	2-86
OR=DT	OR型·時刻·一致	341	無	無	無	2-89
OR=F	OR型·單精確度浮動小數點·一致	329	無	無	無	2-230
OR=L	OR型·倍精確度·一致	301	無	無	無	2-86
OR=S	OR型·帶符號·一致	302	無	無	無	2-86
OR=SL	OR型·帶符號倍精確度·一致	303	無	無	無	2-86
OR>	OR型·超過	320	無	無	無	2-86
OR>=	OR型·以上	325	無	無	無	2-86
OR>=DT	OR型·時刻·以上	346	無	無	無	2-89

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
OR>=F	OR型精確度浮動小以上	334	無	無	無	2-230
OR>=L	OR型·倍精確度·以上	326	無	無	無	2-86
OR>=S	OR型·帶符號·以上	327	無	無	無	2-86
OR>=SL	OR型·帶符號倍精確度·以上	328	無	無	無	2-86
OR>DT	OR型·時刻·超過	345	無	無	無	2-89
OR>F	OR型·單精確度浮動小數點·超過	333	無	無	無	2-230
OR>L	OR型·倍精確度·超過	321	無	無	無	2-86
OR>S	OR型·帶符號·超過	322	無	無	無	2-86
OR>SL	OR型·帶符號倍精確度·超過	323	無	無	無	2-86
ORW	字元邏輯和	035	@ORW	無	無	2-203
ORWL	字元倍精確度邏輯和	611	@ORWL	無	無	2-203
OUT	輸出	無	無	無	IOUT	2-18
OUT NOT	輸出否定	無	無	無	IOUT NOT	2-18

## P

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
PIDAT	附自動微調功能的 PID 演算	191	無	無	無	2-245
PLS2	定位	887	@PLS2	無	無	2-306
PRV	輸出脈衝現在值	881	@PRV	無	無	2-292
PULS	脈衝量設定	886	@PULS	無	無	2-304
PWM	PWM 輸出	891	@PWM	無	無	2-318

## R

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
RET	副程式返回	093	無	無	無	2-278
ROL	附CY的1位元向左回轉	027	@ROL	無	無	2-131
ROR	附CY的1位元向右回轉	028	@ROR	無	無	2-133
RSET	復歸	無	@RSET	%RSET	!RSET	2-29
RSTA	多位元復歸	531	@RSTA	無	無	2-31
RSTB	1位元復歸	533	@RSTB	無	!RSTB	2-33
RXD	序列埠輸入	235	@RXD	無	無	2-350

## S

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
SBN	副程式進入	092	無	無	無	2-278
SBS	呼叫副程式	091	@SBS	無	無	2-274
SCL	轉換	194	@SCL	無	無	2-261
SCL2	轉換 2	486	@SCL2	無	無	2-265
SCL3	轉換 3	487	@SCL3	無	無	2-267
SDEC	7 段解碼器	078	@SDEC	無	無	2-333
SET	設定	無	@SET	%SET	!SET	2-29
SETA	多位元設定	530	@SETA	無	無	2-31
SETB	1 位元組	532	@SETB	無	!SETB	2-33
SFT	移位暫存器	010	無	無	無	2-123
SFTR	左右移位暫存器	084	@SFTR	無	無	2-125
SLD	向左移動1位數	074	@SLD	無	無	2-135
SNXT	進入步進階梯區	009	無	無	無	2-321
SPED	頻率設定	885	@SPED	無	無	2-301
SRD	向右移動1位數	075	@SRD	無	無	2-135
STC	設定進位	040	@STC	無	無	2-372
STEP	定義步進階梯區	008	無	無	無	2-321
SWAP	位元置換	637	@SWAP	無	無	2-241

## T

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
TCMP	比較表一致	085	@TCMP	無	無	2-98
TIM	100 ms 計時器	無	無	無	無	2-64
TIMH	10 ms 計時器	015	無	無	無	2-67
TIMHX	10 ms 計時器	551	無	無	無	2-67
TIML	長時間計時器	542	無	無	無	2-75
TIMLX	長時間計時器	553	無	無	無	2-75
TIMX	100 ms 計時器	550	無	無	無	2-64
TMHH	1 ms 計時器	540	無	無	無	2-70
TMHXX	1 ms 計時器	552	無	無	無	2-70
TPO	時間分割比例輸出	685	無	無	無	2-255
TR	暫時記憶繼電器	無	無	無	無	2-20
TTIM	累加計時器	087	無	無	無	2-72
TTIMX	累加計時器	555	無	無	無	2-72
TXD	序列埠輸出	236	@TXD	無	無	2-345

## U

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
UP	P. F. 上微分	521	無	無	無	2-17

## W

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
WDT	設定週期監控時間	094	@WDT	無	無	2-373
WSFT	字元移位	016	@WSFT	無	無	2-127

## X

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
XCHG	資料交換	073	@XCHG	無	無	2-117
XFER	區塊傳送	070	@XFER	無	無	2-113
XFRB	多位數傳送	062	@XFRB	無	無	2-111
XORL	字元倍精確度 XOR	612	@XORL	無	無	2-205
XORW	字元互斥或 (Exclusive OR:XOR) 演算	036	@XORW	無	無	2-205

## Z

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
ZCP	區域比較	088	無	無	無	2-102
ZCPL	倍精確度區域比較	116	無	無	無	2-102

## 其他符號

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
7SEG	7 段顯示	214	無	無	無	2-341
+	帶符號·無 CY 的 BIN 加法運算	400	@ +	無	無	2-151
++	BIN 加法運算	590	@ ++	無	無	2-143
++B	BCD 遞增	594	@ ++ B	無	無	2-147
++BL	BCD 倍精確度遞增	595	@ ++ BL	無	無	2-147
++L	BIN 倍精確度遞增	591	@ ++ L	無	無	2-143
+B	不含 CY 的 BCD 加法運算	404	@ + B	無	無	2-155
+BC	含 CY 的 BCD 加法運算	406	@ + BC	無	無	2-157
+BCL	含 CY 的 BCD 倍精確度加法運算	407	@ + BCL	無	無	2-157
+BL	不含 CY 的 BCD 倍精確度加法運算	405	@ + BL	無	無	2-155

指令記號	指令名稱	FUN No.	啟動為上升	啟動為下降	指定為立即更新	頁碼
+ C	帶符號·CY的BIN加法運算	402	@ + C	無	無	2-153
+ CL	帶符號·CY的BIN倍精確度加法運算	403	@ + CL	無	無	2-153
+ F	浮動小數點加法運算	454	@ + F	無	無	2-227
+ L	帶符號·無CY的BIN倍精確度加法運算	401	@ + L	無	無	2-151
—	帶符號·無CY的BIN減法運算	410	@—	無	無	2-159
— —	BIN減法運算	592	@— —	無	無	2-145
— — B	BCD遞減	596	@— — B	無	無	2-149
— — BL	BCD倍精確度遞減	597	@— — BL	無	無	2-149
— — L	BIN倍精確度遞減	593	@— — L	無	無	2-145
— B	不含CY的BCD減法運算	414	@— B	無	無	2-164
— BC	含CY的BCD減法運算	416	@— BC	無	無	2-167
— BCL	含CY的BCD倍精確度減法運算	417	@— BCL	無	無	2-167
— BL	不含CY的BCD倍精確度減法運算	415	@— BL	無	無	2-164
— C	帶符號·CY的BIN減法運算	412	@— C	無	無	2-162
— CL	帶符號·CY的BIN倍精確度減法運算	413	@— CL	無	無	2-162
— F	浮動小數點減法運算	455	@— F	無	無	2-227
— L	帶符號·無CY的BIN倍精確度減法運算	411	@— L	無	無	2-159
*	帶符號BIN乘法運算	420	@ *	無	無	2-169
* B	BCD乘法運算	424	@ * B	無	無	2-171
* BL	BCD倍精確度乘法運算	425	@ * BL	無	無	2-171
* F	浮動小數點乘法運算	456	@ * F	無	無	2-227
* L	帶符號BIN倍精確度乘法運算	421	@ * L	無	無	2-169
/	帶符號BIN除法運算	430	@ /	無	無	2-173
/ B	BCD除法運算	434	@ / B	無	無	2-175
/ BL	BCD倍精確度除法運算	435	@ / BL	無	無	2-175
F	浮動小數點除法運算	457	@ / F	無	無	2-227
/ L	帶符號BIN倍精確度除法運算	431	@ / L	無	無	2-173

## ASCII碼一覽表

		上位4位元															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
下位4位元	0			Sp	0	@	P	'	p				—	タ	ミ		
	1			!	1	A	Q	a	q				。	ア	チ	ム	
	2			"	2	B	R	b	r				「	イ	ツ	メ	
	3			#	3	C	S	c	s				」	ウ	テ	モ	
	4			\$	4	D	T	d	t				、	エ	ト	ヤ	
	5			%	5	E	U	e	u				・	オ	ナ	ユ	
	6			&	6	F	V	f	v				ヲ	カ	ニ	ヨ	
	7			'	7	G	W	g	w				ア	キ	ヌ	ラ	
	8			(	8	H	X	h	x				イ	ク	ネ	リ	
	9			)	9	I	Y	i	y				ウ	ケ	ノ	ル	
	A			*	:	J	Z	j	z				エ	コ	ハ	レ	
	B			+	;	K	[	k	{				オ	サ	ヒ	ロ	
	C			,	<	L	¥	l					ヤ	シ	フ	ワ	
	D			—	=	M	]	m	}				ユ	ス	ヘ	ン	
	E			.	>	N	^	n	~				ヨ	セ	ホ	〃	
	F			/	?	O	_	o					ツ	ソ	マ	°	



# 使用手冊修訂記錄

使用手冊的修訂符號將會刊印在位於封面/封底左下方Man. No.的後方。

**Man. No. SBCA-356C**

↑ 修訂符號

修訂符號	修訂年月	修訂理由/修訂頁數
A	2009年3月	初版
B	2009年6月	誤記修正
C	2010年1月	· 新增10/14/60點、類比內建20點輸出輸入CPU裝置 · 誤記修正



各位OMRON產品愛用者

## 選購時的注意事項

首先感謝您平時對OMRON產品的支持與愛護。  
各位根據型錄購買本公司控制器產品(以下稱為「本公司產品」時，敬請確認以下內容。

### 1. 保固內容：

#### 保固期間

本公司的產品保固期間為購買產品後亦或是將產品交貨至指定地點後一年內。

#### 保固範圍

上述保固期間中，若產品因本公司責任發生故障者，將於原購買地點提供免費的維修服務或更換代替品。

但下列故障原因不在保固範圍內：

- a) 不在本目錄或規格書內所規定之條件、環境使用下所造成的故障
- b) 非產品本身原因所造成的故障
- c) 非經由本公司所進行的改裝或維修所造成的故障
- d) 未依照原本設計之使用方式所造成的故障
- e) 出貨時之科技水準所無法預測之原因所造成的故障
- f) 其它天災、災害等不可抗力所造成的故障

此外，上述保固僅限於本公司產品本身，因產品故障所導致之相關損失並不包含在本保固範圍內。

### 2. 責任限制

關於因本公司產品所引發之一切特別損害、間接損害、消極損害(應得利益之喪失)，本公司不負任何責任。

關於本公司之可程式化產品，針對非經本公司技術人員所執行之程式或因其所造成之結果，本公司不負任何責任。

### 3. 選購時，應符合用途條件

將本公司商品與其他搭配使用時，請確認是否符合顧客所需之規格、法規或限制等。

此外，請顧客自行確認目前所使用的系統、機械或是裝置是

否適用於本公司商品。

再者，請顧客自行確認本公司商品是否符合目前所使用的系統、機械或是裝置。

如未確認是否符合或適用時，本公司無須對本公司商品的適用性負責。

使用於以下用途時，敬請洽詢本公司業務人員後根據規格書等進行確認，同時注意安全設施，例如使用的額定電壓、性能要盡量低於限制範圍以策安全；或是採用在發生故障時可將危險程度降至最小的安全回路等。

- a) 用於戶外、會遭受潛在化學污染、電力會遭受妨礙的用途、或是在本型錄未記載的條件或環境下使用。
- b) 核能控制設備、焚燒設備、鐵路、航空、車輛設備、醫療機器、娛樂用途機械設備、安全裝置以及遵照政府機構或個別業界規定的設備。
- c) 危及生命或財產的系統、機械、裝置。
- d) 瓦斯、水/供電系統，或是系統穩定性有特殊要求的設備。
- e) 其他符合a)~d)、需要高度安全性的用途。

當顧客將本公司商品使用於可能嚴重危害生命、財產等用途時，敬請務必事先確認系統整體有危險告示、並採用備援設計等可確保安全性，以及本公司產品針對整體設備的特定用途上的配電與設置適當。

由於本型錄所記載的應用程式範例屬於參考性質，如需直接採用時，使用前請先確認機械、裝置的功能與安全性。敬請顧客務必以正確的方法來使用本公司產品，並了解使用時的禁止事項與注意事項，以免不當的使用而造成他人意外的損失。

### 4. 規格變更

本型錄所記載的規格以及附屬品，可能會在必要時、進行改良時或其他事由而變更。敬請洽詢本公司或特約店之營業人員，以確認本公司商品的實際規格。

# 台灣歐姆龍股份有限公司

OMRON 產品技術客服中心



008-0186-3102

【產業自動化】

產品技術諮詢服務

· 服務時間 ·

週一 ~ 週五

8:15~12:00/13:00~17:00

· FAX諮詢專線 ·

002-86-21-50504618

· E-mail諮詢 ·

<http://www.omron.com.tw>

<http://www.omron.com.tw>

■ 台北總公司：台北市復興北路363號6樓(弘雅大樓)

電話：02-2715-3331 傳真：02-2712-6712

■ 新竹事業所：新竹市民生路46號1F

電話：03-535-7330 傳真：03-535-7511

■ 台中事業所：台中市港路一段345號27樓之3(中港高峰大樓)

電話：04-2325-0834 傳真：04-2325-0734

■ 台南事業所：台南市民生路二段 307 號 22 樓之 1 (台南運河大樓)

電話：06-226-2208 傳真：06-226-1751

特約店

註：規格可能改變，恕不另行通知，最終以產品說明書為準。