

目 錄

# 第一章 元件及常數

1.1	PLC 所使用的各種數值 · · · · ·	1-1
1.2	常數 K,H,E(10 進值/16 進值/實數) · · · · ·	1-5
1.3	文字字串 · · · · ·	1-6
1.4	位數元件(Kn□***)	1-7
1.5	資料暫存器的位元指定(D.□b)	1-8
1.6	BFM 的直接指定(U□\G□)	1-8
1.7	V,Z 修飾 · · · · ·	1-9

## 第二章 程式設計須知

2.1	如何閱讀指令	2-1
2.2	程式設計上的注意事項	2-4
2.3	輸入/輸出信號的再生動作	2-7
2.4	程式流程指令的相互關係	2-8
2.5	應用指令的表現方式及執行方式	2-10

第三章 基本指令

基本指令一覽	3-1
母線開始 a 接點【LD】	3-4
母線開始 b 接點【LDI】	3-5
輸出指令【OUT】	3-7
串接 a 接點【AND】	3-10
串接 b 接點【ANI】	3-11
並接 a 接點【OR】	3-13
並接 b 接點【ORI】	3-14
LD/AND/OR 上昇微分接點【LDP/ANDP/ORP】	3-17
LD/AND/OR 下降微分接點【LDF/ANDF/ORF】	3-18
兩個回路並接【ORB】	3-20
兩個回路串接【ANB】	3-21
分歧點開始/繼續/結束【MPS/MRD/MPP】	3-22
主控點開始/解除【MC/MCR】	3-26
反向輸出【INV】	3-29
上昇/下降微分輸出【PLS/PLF】	3-30
自保持指令/保持解除指令【SET/RST】	3-32

無處理【NOP】	3-35
程式結束【END】	3-36

## 第四章 程式流程指令

FNC00 有條件的跳躍【CJ】	4-2
FNC01 呼叫副程式【CALL】	4-6
FNC02 副程式結束返回【SRET】	4-6
FNC03 中斷插入副程式結束返回【IRET】	4-8
FNC04 中斷插入允許【EI】	4-8
FNC05 中斷插入禁止【DI】	4-9
FNC06 主程式結束【FEND】	4-14
FNC07 開門狗時間設定【WDT】	4-16
FNC08 迴圈開始【FOR】	4-18
FNC09 迴圈結束【NEXT】	4-18

## 第五章 傳送・比較

FNC10 比較【CMP】	5-2
FNC11 區域比較【ZCP】	5-4
FNC12 傳送【MOV】	5-6
FNC13 位數移動【SMOV】	5-8
FNC14 反相傳送【CML】	5-10
FNC15 複數傳送【BNOV】	5-12
FNC16 多點傳送【FMOV】	5-14
FNC17 資料交換【XCH】	5-15
FNC18 BIN→BCD 變換【BCD】	5-16
FNC19 BCD→BIN 變換【BIN】	5-17

## 第六章 四則・邏輯運算

FNC20 加算【ADD】	6-2
FNC21 減算【SUB】	6-3
FNC22 乘算【MUL】	6-5
FNC23 除算【DIV】	6-6
FNC24 加一【INC】	6-7
FNC25 減一【DEC】	6-8
FNC26 理論積【WAND】	6-9
FNC27 理論和【WOR】	6-10
FNC28 排它理論和【WXOR】	6-11
FNC29 補數【NEG】	6-12

## 第七章 位移・旋轉

FNC30	右旋轉【ROR】	7-2
FNC31	左旋轉【SUB】	7-3
FNC32	附 CY 的右旋轉【RCR】	7-4
FNC33	附 CY 的左旋轉【RCL】	7-5
FNC34	位元右移【SFTR】	7-6
FNC35	位元左移【SFTL】	7-7
FNC36	資料右移【WSFR】	7-10
FNC37	資料左移【WSFL】	7-11
FNC38	資料位移寫入【SFWR】	7-13
FNC39	資料位移讀出【SFRD】	7-14

## 第八章 資料處理

FNC40	區域復歸【ZRST】	8-2
FNC41	解碼【DECO】	8-4
FNC42	編碼【ENCO】	8-6
FNC43	ON 位元總數【SUM】	8-7
FNC44	ON 位元判定【BON】	8-8
FNC45	平均值【MEAN】	8-9
FNC46	警報點輸出【ANS】	8-10
FNC47	警報點復歸【ANR】	8-11
FNC48	BIN 開平方根【SQR】	8-13
FNC49	BIN 整數→2 進小數點值【FLT】	8-14

## 第九章 高速處理

FNC50	輸入/輸出強制再生【REF】	9-2
FNC51	變更輸入端反應時間【REFF】	9-4
FNC52	多點矩陣輸入【MTR】	9-6
FNC53	高速計數器比較 ON【HSCH】	9-9
FNC54	高速計數器比較 OFF【HSCR】	9-12
FNC55	高速計數器區域比較【HSZ】	9-14
FNC56	速度偵測【SPD】	9-22
FNC57	脈波輸出【PLSY】	9-24
FNC58	脈波寬度調整【PWM】	9-26
FNC59	脈波輸出附加減速【PLSR】	9-28

## 第十章 便利指令

FNC60	手動/自動運轉模態【IST】	10-2
-------	----------------	------

FNC61	多點比較【SER】	10-12
FNC62	絕對值凸輪控制【ABSD】	10-14
FNC63	相對值凸輪控制【INCD】	10-16
FNC64	交導計時器【TTMR】	10-18
FNC65	特殊計時器【STMR】	10-20
FNC66	單 ON 雙 OFF【ALT】	10-22
FNC67	傾斜信號【RAMP】	10-24
FNC68	圓盤控制【ROTC】	10-26
FNC69	資料排序【SORT】	10-29

## 第十一章 外部設定・顯示

FNC70	10 個按鍵鍵盤輸入【TKY】	11-2
FNC71	16 個按鍵鍵盤輸入【HKY】	11-4
FNC72	指撥開關輸出【DSW】	11-6
FNC73	7 段顯示器解碼輸出【SEGD】	11-8
FNC74	7 段顯示器掃描輸出【SEGL】	11-10
FNC75	箭頭操作盤【ARWS】	11-13
FNC76	ASCII 碼變換【ASC】	11-16
FNC77	ASCII 碼輸出【PR】	11-18
FNC78	特殊模組 BFM 資料讀出【FROM】	11-20
FNC79	特殊模組 BFM 資料寫出【TO】	11-21

## 第十二章 RS232・PID

FNC80	RS232 通信【RS】	12-2
FNC81	8 進位元傳送【PRUN】	12-9
FNC82	HEX→ASCII 碼變換【ASCI】	12-10
FNC83	ASCII→HEX 碼變換【HEX】	12-14
FNC84	總和檢查【CCD】	12-17
FNC87	RS232 通信 2【RS2】	12-20
FNC88	運算【PID】	12-22

## 第十三章 資料處理-2

FNC102	(V,Z)的內容儲存【ZPUSH】	13-2
FNC103	(V,Z)的內容回覆【ZPOP】	13-4

## 第十四章 小數點運算-1

FNC110	2 進小數點比較【ECMP】	14-2
FNC111	2 進小數點區域比較【EZCP】	14-4
FNC112	2 進小數點資料傳送【EMOV】	14-6

FNC116	2 進小數點→文字字串變換【ESTR】	14-8
FNC117	文字字串(ASCII 碼)→2 進小數點變換指令【EVAL】	14-14
FNC118	EBCD 2 進小數點→10 進小數點變換【EBCD】	14-18
FNC119	EBIN 10 進小數點→2 進小數點變換【EBIN】	14-19

## 第十五章 小數點運算-2

FNC120	2 進小數點加算【EADD】	15-2
FNC121	2 進小數點減算【ESUB】	15-3
FNC122	2 進小數點乘算【EMUL】	15-4
FNC123	2 進小數點除算【EDIV】	15-5
FNC124	2 進小數點指數運算【EXP】	15-6
FNC125	2 進小數點自然對數運算【LOGE】	15-8
FNC126	2 進小數點常用對數運算【LOG10】	15-10
FNC127	2 進小數點開平方根【ESQR】	15-12
FNC128	2 進小數點補數(符號反轉) 【ENEG】	15-13
FNC129	2 進小數點→BIN 整數變換【INT】	15-14

## 第十六章 小數點運算-3

FNC130	2 進小數點 SIN 運算【SIN】	16-2
FNC131	2 進小數點 COS 運算【COS】	16-4
FNC132	2 進小數點 TAN 運算【TAN】	16-5
FNC133	2 進小數點 $\sin^{-1}$ 運算【ASIN】	16-6
FNC134	2 進小數點指數運算 $\cos^{-1}$ 運算【ACOS】	16-8
FNC135	2 進小數點 $\tan^{-1}$ 運算【ATAN】	16-10
FNC136	2 進小數點角度→弧度變換【RAD】	16-12
FNC137	2 進小數點弧度→角度變換【DEG】	16-14

## 第十七章 資料處理-3

FNC140	資料加總【WSUM】	17-2
FNC141	字元分離【WTOB】	17-4
FNC142	字元結合【BTOM】	17-6
FNC143	4 位元結合【UNI】	17-10
FNC144	4 位元分離【DIS】	17-12
FNC147	上下 8 位元互換【SWAP】	17-14
FNC149	資料排序 2【SORT2】	17-15

## **第十八章 定位控制**

FNC150	近點搜尋原點復歸【DSZR】	18-2
FNC151	中斷插入 1 段速定位【DVIT】	18-4
FNC152	資料表單定位控制【TBL】	18-6
FNC155	絕對位置的讀出【ABS】	18-7
FNC156	原點復歸【ZRN】	18-8
FNC157	變速輸出【PLSV】	18-10
FNC158	相對距離定位控制【DRV1】	18-12
FNC159	絕對位置定位控制【DRV1】	18-14

## **第十九章 萬年曆時鐘**

FNC160	萬年曆資料比較【TCMP】	19-2
FNC161	萬年曆資料區域比較【TZCP】	19-4
FNC162	萬年曆資料加算【TADD】	19-6
FNC163	萬年曆資料減算【TSUB】	19-8
FNC164	萬年曆→秒的變換【HTOS】	19-10
FNC165	秒→萬年曆的變換【STOH】	19-13
FNC166	萬年曆資料讀出【TRD】	19-15
FNC167	萬年曆資料寫入【TWR】	19-16
FNC169	HOUR 小時的測量【HOUR】	19-18

## **第二十章 絕對位置的編寫**

FNC170	BIN→絕對位置變換【GRY】	20-2
FNC171	絕對位置→BIN 變換【GBIN】	20-3
FNC176	類比讀出【RD3A】	20-4
FNC177	類比寫入【WR3A】	20-5

## **第二十一章 其他指令**

FNC182	元件註解的讀出【COMRD】	21-2
FNC184	亂數產生【RND】	21-4
FNC186	時鐘脈衝產生【DUTY】	21-5
FNC188	運算【CRC】	21-8
FNC189	高速計數器現在值傳送【HCMOV】	21-12

## **第二十二章 區塊資料處理**

FNC192	區塊資料加算【BK+】	22-2
FNC193	區塊資料減算【BK-】	22-5
FNC194	區塊資料比較【BKCMP=】	22-8

## 第二十三章 文字字串處理

FNC200	BIN→文字字串的變換【STR】	23-2
FNC201	文字字串→BIN 變換【VAL】	23-6
FNC202	文字字串結合【\$T】	23-12
FNC203	文字字串長度判定【LEN】	23-14
FNC204	右取文字【RIGHT】	23-16
FNC205	左取文字【LEFT】	23-18
FNC206	任意位置文字取出【MIDR】	23-20
FNC207	任意位置文字替換【MIDW】	23-22
FNC208	文字搜尋【INSTR】	23-26
FNC209	文字字串傳送【\$MOV】	23-28

第二十四章 資料表單

FNC210	資料表單中的資料刪除【FDEL】	24-2
FNC211	資料表單中的資料插入【FINS】	24-4
FNC212	資料表單中的後入先出【POP】	24-6
FNC213	16 位元資料右移【SFR】	24-10
FNC214	16 位元資料左移【SFL】	24-12

## 第二十五章 接點型態比較

FNC224	母線開始(S1)=(S2)時導通【LD=】	25-2
FNC225	母線開始(S1)>(S2)時導通【LD>】	25-2
FNC226	母線開始(S1)<(S2)時導通【LD<】	25-2
FNC228	母線開始(S1)≠(S2)時導通【LD<>】	25-2
FNC229	母線開始(S1)≤(S2)時導通【LD≤】	25-2
FNC230	母線開始(S1)≥(S2)時導通【LD≥】	25-2
FNC232	串接指令(S1)=(S2)時導通【AND=】	25-4
FNC233	串接指令(S1)>(S2)時導通【AND>】	25-4
FNC234	串接指令(S1)<(S2)時導通【AND<】	25-4
FNC236	串接指令(S1)≠(S2)時導通【AND<>】	25-4
FNC237	串接指令(S1)≤(S2)時導通【AND≤】	25-4
FNC238	串接指令(S1)≥(S2)時導通【AND≥】	25-4
FNC240	並接指令(S1)=(S2)時導通【OR=】	25-6

FNC241	並接指令(S1)>(S2)時導通【OR>】	25-6
FNC242	並接指令(S1)<(S2)時導通【OR<】	25-6
FNC244	並接指令(S1)≠(S2)時導通【OR<>】	25-6
FNC245	並接指令(S1)≤(S2)時導通【OR≤】	25-6
FNC246	並接指令(S1)≥(S2)時導通【OR≥】	25-6

## 第二十六章 資料控制

FNC256	上下極限值控制【LIMIT】	26-2
FNC257	不感帶控制【BAND】	26-6
FNC258	資料區域控制【ZONE】	26-10
FNC259	尺規【SCL】	26-14

## 第二十七章 資料表單處理

FNC260	10 進 ASC II 碼→BIN 值變換【DABIN】	27-2
FNC261	BIN 值→10 進 ASC II 碼變換【BINDA】	27-5
FNC269	尺規 2【SCL2】	27-8

## 第二十八章 變頻器的通訊傳輸

FNC270	變頻器的運轉監視【IVCK】	28-2
FNC271	變頻器的運轉控制【IVDR】	28-4
FNC272	變頻器的參數讀出【IVRD】	28-6
FNC273	變頻器的參數寫入【IVWR】	28-8
FNC274	變頻器的多個參數寫入【IVBWR】	28-10

## 第二十八章 資料傳輸 3

FNC278	BFM 分割讀出【RBFM】	28-14
FNC279	BFM 分割寫入【IVBWR】	28-18

## 第二十九章 高速處理

FNC280	HSCT 高速計數器資料表單比較	29-2
--------	------------------	------

## 第三十章 擴充檔案暫存器

FNC290	擴充檔案暫存器的讀出【LOADR】	30-2
FNC291	擴充檔案暫存器的寫入【SAVER】	30-4
FNC292	擴充暫存器的格式化【INITR】	30-11
FNC293	登錄至擴充暫存器【LOGR】	30-14
FNC294	擴充檔案暫存器的清除與寫入【RWER】	30-18
FNC295	擴充檔案暫存器的格式化【INITR】	30-22

## 第三十一章 PLC 異常的解讀方法及錯誤一覽表

31.1	PLC 動作情況顯示用 LED 及相關顯示顏色 · · · · ·	31-1
31.2	使用階梯圖編輯軟體 GPPW 來監視異常現象 · · · · ·	31-4
31.2.1	錯誤發生時的表現方式 · · · · ·	31-5
31.3	顯示錯誤編號的特殊輔助繼電器 · · · · ·	31-5
31.3.1	錯誤偵測元件(M8060~/D8060) · · · · ·	31-5
31.3.2	停電檢出時間(D8008) · · · · ·	31-6
31.3.3	錯誤顯示用各特殊元件相互間的顯示關係 · · · · ·	31-6
31.4	錯誤一覽表及處置 · · · · ·	31-7

第一章 指令所使用的元件及常數

本章是針對指令內來源運算元及目的地運算元可指定的元件及常數做說明。



## 1.1 PLC 所使用的各種數值(8 進值/10 進值/16 進值/實數)

FX3U 的內部依據控制目的共使用 5 種數值執行運算工作，各種數值的任務及功能如下所示。

### 1.1.1 數值的種類

### 1. 10 進值(DEC: Decimal Number K 值)

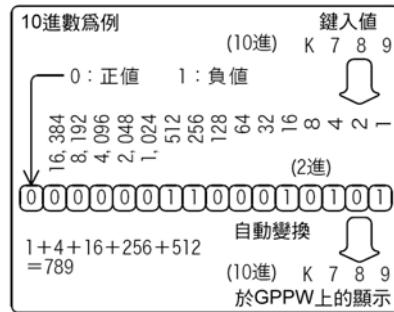
- 當成計時器(T)及計數器(C)的設定值。(K 常數)
  - 當成 M、T、C、S 的元件號碼來使用。(元件號碼)
  - 置於應用指令的運算元當中來使用。例：MOV K1 D0。(K 常數)

## 2. 16 進值(HEX: Hexadecimal Number H 值)

- 置於應用指令的運算元當中來使用，例：MOV H0A1B D0。(H 常數)

### 3. 2 進值(BIN: Binary Number BIN 值)

- 使用者在程式設計時，將 10 進數或 16 進數打入至程式中當成計時器(T)、計數器(C)的設定值或暫存器的內容，實際上，PLC 會將這些數值轉換成 2 進數於內部執行運算動作，但是，使用者使用周邊裝置作數值監視時，PLC 內部的 2 進數又會自動以 10 進數或 16 進數的格式顯示於畫面上，如右圖。



#### 4. 8 進值(OCT: Octal Number O 值)

- FX3U 的輸入端及輸出端號碼採用 8 進數作編號。  
因此, I/O 編號的順序為「0~7、10~17…70~77、100~107」, 8、9 及 8□、9□ 均是不存在的號碼。

## 5. BCD(BCD : Binary Code Decimal BCD 值)

- 以  $8(2^3)$ 、 $4(2^2)$ 、 $2(2^1)$ 、 $1(2^0)$  共 4 個位元去組合 10 進值的一位數，以 16 位元去組合 10 進數的 4 位數，此種位元組合型態為 BCD 碼，它專門用來接受指撥開關的輸入值及將運算值送出至 7 段顯示器作顯示。

## 6. 實數(浮點運算值)

- FX3U 可執行浮點運算，所謂的浮點運算指的是小數點的 $+$ 、 $-$ 、 $\times$ 、 $\div$ 、 $\sqrt{\phantom{x}}$ 等運算，任何一個數值要執行浮點運算前均必須轉換成實數(浮點運算值)方可進行。

# 第一章 指令所使用的元件及常數

## 1.1.2 數值的變換

各種數值及位元變換如下表所示。

10進數(DEC)	8進數(OCT)	16進數HEX	2進數(BIN)		BCD	
0	0	00	0000	0000	0000	0000
1	1	01	0000	0001	0000	0001
2	2	02	0000	0010	0000	0010
3	3	03	0000	0011	0000	0011
4	4	04	0000	0100	0000	0100
5	5	05	0000	0101	0000	0101
6	6	06	0000	0110	0000	0110
7	7	07	0000	0111	0000	0111
8	10	08	0000	1000	0000	1000
9	11	09	0000	1001	0000	1001
10	12	0A	0000	1010	0001	0000
11	13	0B	0000	1011	0001	0001
12	14	0C	0000	1100	0001	0010
13	15	0D	0000	1101	0001	0011
14	16	0E	0000	1110	0001	0100
15	17	0F	0000	1111	0001	0101
16	20	10	0001	0000	0001	0110
:	:	:	:	:	:	:
99	143	63	0110	0011	1001	1001
:	:	:	:	:	:	:

### 主要的用途

10進數(DEC)	8進數(OCT)	16進數HEX	2進數(BIN)	BCD
常數K值,除X/Y以外的元件編號	輸入X/輸出Y的元件編號	常數H值	PLC內部處理的數值	BCD指撥開關及編碼後的7段顯示器

# 第一章 指令所使用的元件及常數

## 1.5 資料暫存器的位元指定(D□. b)

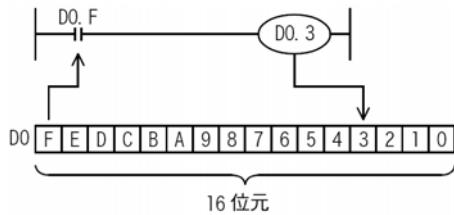
資料暫存器(D)的位元可獨立出來當成一般位元使用。

資料暫存器(D)的位元是以 16 進編號(0~F)來指定。

例：D0.0 代表的是 D0 的 b0。

資料暫存器(D)的位元不可再使用 V,Z 作修飾。

不只資料暫存器(D)，特殊暫存器(SD)也可做這樣的指定。



## 1.6 BFM 的直接指定(U□\G□)

應用指定可直接指定特殊模組的 BFM 作資料讀出或寫入的動作。

U 後面的□請寫入特殊模組的編號、G 後面的□請寫入 BFM 的位址編號。

例：U0\G0 代表的是特殊模組 0 號機的 BFM#0。

(U□\G□)不可再使用 V,Z 作修飾。

可指定的範圍如下所示。

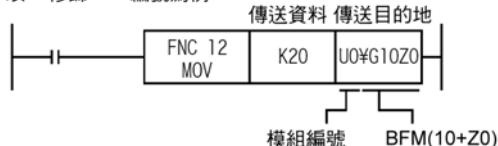
特殊模組編號：0~7

BFM 的位址編號：0~32767

以MOV指令為例



以Z0修飾BFM編號為例



FNC 150 DSZR	16位元指令 指令符號 DSZR 執行條件 佔9個位址 DSZR 連續執行	32位元指令 指令符號 - 執行條件 -
<b>近點搜尋</b>		
<b>原點復歸</b>		

元件類別	內容	資料格式
<b>S<sub>1</sub>*</b>	近點信號(DOG)所連接的輸入點編號	位元
<b>S<sub>2</sub>*</b>	零點信號所連接的輸入點編號	
<b>D<sub>1</sub>*</b>	脈波輸出的輸出點編號	
<b>D<sub>2</sub>*</b>	回轉方向的輸出點編號	

元件類別	位元元件							字元元件								其他								
	使用者							指定位數				使用者		特殊模組		間接指定		常數	實數	文字	指標			
	X	Y	M	T	C	S	D□.b	KnX	KnY	KnM	KnS	T	C	D	R	U□￥G□	V	Z	修飾	K	H	E	"□"	P
<b>S<sub>1</sub>*</b>	●	●	●	●			▲1												●					
<b>S<sub>2</sub>*</b>	▲2																		●					
<b>D<sub>1</sub>*</b>	▲3																		●					
<b>D<sub>2</sub>*</b>	▲4	●	●				▲1												●					

▲ 1 : D□.b 不可使用間接指定暫存器(V,Z)作修飾。

▲ 2 : 請指定 X0~X7。

▲ 3 : 請指定主機的輸出端(電晶體輸出)Y0,Y1,Y2 或者是高速輸出模組※1 的 Y0,Y1,Y2※2,Y3※2

※ 1 : FX3UC-32MT-LT 不可連接是高速輸出模組。

※ 2 : 若是使用高速輸出模組的 Y2,Y3 時，必須再使用第 2 台高速輸出模組。

注意：

- 繼電器輸出的 FX3U 使用本指令時，必須連接高速輸出模組作輸出。

- 高速輸出模組的輸出為差動信號。

▲ 4 : 本指令的輸出必須是電晶體輸出。

使用 FX3U+高速輸出模組的時候，輸出編號的指定如下表所示。

高速輸出模組的排列位置	脈波輸出	回轉方向輸出
第 1 台	D1 • =Y0	D2 • =Y4
	D1 • =Y1	D2 • =Y5
第 2 台	D1 • =Y2	D2 • =Y6
	D1 • =Y3	D2 • =Y7

**功能及動作****16 位元運算(DSZR)**

本指令被執行時，FX3U 執行原點復歸的動作，原點復歸到達機械原點時，現在值被當成機械原點的位置，本指令與 ZRN 指令(FNC156)的不同點如下：

- 具有近點(DOG)搜尋功能。
- 具有近點(DOG)及零點搜尋功能，但是不具零點次數計算功能。

**RUN 當中程式寫入的注意事項**

DSZR 指令(FNC150)執行中(脈波輸出中)，請避免'RUN 當中程式寫入'動作，此舉，即有可能造成 脈波減速停止

FNC 158  
DRV1

16位元指令 指令符號 執行條件  
佔9個位址 DRV1 連續執行

32位元指令 指令符號 執行條件  
佔17個位址 DDRV1 連續執行

### 相對距離 定位控制

元件類別	內容	資料格式
S <sub>1*</sub>	指定定位用的相對距離值 ※1	BIN 16/32 位元
S <sub>2*</sub>	指定輸出的脈波速度 ※2	
D <sub>1*</sub>	脈波輸出的輸出點編號	
D <sub>2*</sub>	回轉方向的輸出點編號	位元

※1. 設定範圍：16 位元運算時，-32,768~32,767。(0 除外)

32 位元運算時，-999,999~999,999。(0 除外)

※2. 設定範圍：16 位元運算時，10~32,767(Hz)

32 位元運算時，如下表所示。

脈波輸出		設定範圍
FX3U	高速輸出模組	10~200,000(Hz)
FX3UC	主機	10~100,000(Hz)

元件類別	位元元件								字元元件								其他						
	使用者				指定位數				使用者				特殊模組		間接指定		常數	實數	文字	指標			
X	Y	M	T	C	S	D□.b	KnX	KnY	KnM	KnS	T	C	D	R	U□¥G□	V	Z	修飾	K	H	E	"□"	P
S <sub>1*</sub>							●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	
S <sub>2*</sub>							●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	
D <sub>1*</sub>	▲ 1																			●			
D <sub>2*</sub>	▲ 2	●		●	●	▲3													●				

▲ 1 : D□.b 不可使用間接指定暫存器(V,Z)作修飾。

▲ 2 : 請指定主機的輸出端(電晶體輸出)Y0,Y1,Y2 或者是高速輸出模組※1 的 Y0,Y1,Y2※2,Y3※2

※ 1 : FX3UC-32MT-LT 不可連接是高速輸出模組。

※ 2 : 若是使用高速輸出模組的 Y2,Y3 時，必須再使用第 2 台高速輸出模組。

注意：

- 繼電器輸出的 FX3U 使用本指令時，必須連接高速輸出模組作輸出。
- 高速輸出模組的輸出為差動信號。

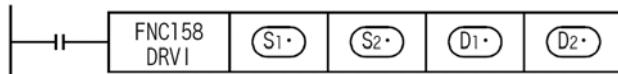
▲ 3 : 本指令的輸出必須是電晶體輸出。

使用 FX3U+高速輸出模組的時候，輸出編號的指定如下表所示。

高速輸出模組的排列位置	脈波輸出	回轉方向輸出
第 1 台	D1 • =Y0	D2 • =Y4
	D1 • =Y1	D2 • =Y5
第 2 台	D1 • =Y2	D2 • =Y6
	D1 • =Y3	D2 • =Y7

**功能及動作****16 位元運算(DRVI)**

本指令被執行時，FX3U 以  $S2 \bullet$  所指定的速度值執行  $S1 \bullet$  所指定的距離(相對距離)後定位停止

**RUN 當中程式寫入的注意事項**

DRVI 指令(FNC158)執行中(脈波輸出中)，請避免'RUN 當中程式寫入'動作，此舉，即有可能造成脈波減速停止。



16位元指令	指令符號	執行條件
佔3個位址	RND RNDP	■ 連續執行 □ 執行一次

32位元指令	指令符號	執行條件
		-

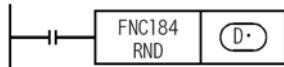
元件類別	內容	資料格式
D•	存放亂數的元件帶頭編號	BIN 16 位元

元件類別	位元元件						字元元件								其他									
	使用者				指定位數				使用者		特殊模組		間接指定		常數	實數	文字	指標						
	X	Y	M	T	C	S	D <b>□</b> b	KnX	KnY	KnM	KnS	T	C	D	R	U <b>□</b> YG <b>□</b>	V	Z	修飾	K	H	E	" <b>□</b> "	P
D•									●	●	●	●	●	●	●	●			●					

## 功能及動作

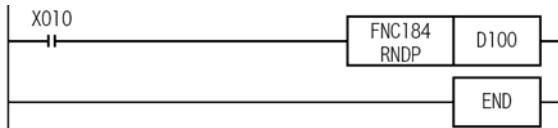
## 1. 16 位元運算(RND/RNDP)

產生 0~32767 之間的亂數並暫存於 D• 所指定的元件當中。



## 程式例

X10=ON 的時候，亂數被暫存於 D100 當中。



FNC 186 DUTY	16位元指令 指令符號 佔7個位址 DUTY	執行條件 連續執行	32位元指令 指令符號 —	執行條件
<b>時鐘脈衝的產生</b>				

元件類別	內容	資料格式
n1	ON 的掃描次數	BIN 16 位元
n2	OFF 的掃描次數	
D•	使用者自定的時鐘脈波輸出	位元

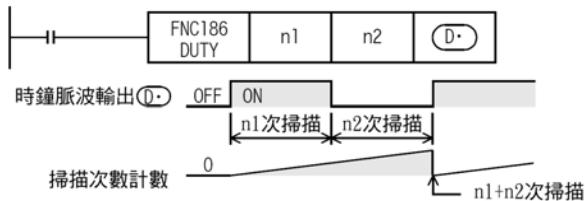
元件類別	位元元件								字元元件								其他							
	使用者				指定位數				使用者				特殊模組		間接指定		常數	實數	文字	指標				
	X	Y	M	T	C	S	D□b	KnX	KnY	KnM	KnS	T	C	D	R	U□¥G□	V	Z	修飾	K	H	E	"□"	P
n1												●	●	●	●					●	●			
n2												●	●	●	●					●	●			
D•			▲																●					

▲：請指定 M8330~M8334

### 功能及動作

#### 1. 16 位元運算(DUTY/DUTYP)

- (1) DUTY 指令被執行的時候，D•所指定的特 M 以 n1 設定的掃描次數 ON、n2 設定的掃描次數 OFF 的狀態輸出。



- (2) 時鐘脈衝輸出 D•的特 M 請指定 M8330~M8334。

- (3) 時鐘脈衝輸出 D•所經過的掃描次數計數被存放在 D8330~D8334 當中。當指令的條件接點=ON、而且 D8330~D8334 的內容=(n1+n2)的時候，D8330~D8334 的內容被復歸為 0。

**FNC 194**  
D BKCMP = P  
**區塊資料比較等於**

**16位元指令 指令符號 執行條件**  
佔9個位址 BKCMP= 連續執行  
BKCMP=P 執行一次

**32位元指令 指令符號 執行條件**  
佔17個位址 DBKCMP= 連續執行  
DBKCMP=P 執行一次

**FNC 195**  
D BKCMP > P  
**區塊資料比較大於**

**16位元指令 指令符號 執行條件**  
佔9個位址 BKCMP= 連續執行  
BKCMP=P 執行一次

**32位元指令 指令符號 執行條件**  
佔17個位址 DBKCMP= 連續執行  
DBKCMP=P 執行一次

**FNC 196**  
D BKCMP < P  
**區塊資料比較小於**

**16位元指令 指令符號 執行條件**  
佔9個位址 BKCMP> 連續執行  
BKCMP>P 執行一次

**32位元指令 指令符號 執行條件**  
佔17個位址 DBKCMP> 連續執行  
DBKCMP>P 執行一次

**FNC 197**  
D BKCMP<> P  
**區塊資料比較不等於**

**16位元指令 指令符號 執行條件**  
佔9個位址 BKCMP<> 連續執行  
BKCMP<>P 執行一次

**32位元指令 指令符號 執行條件**  
佔17個位址 DBKCMP<> 連續執行  
DBKCMP<>P 執行一次

**FNC 198**  
D BKCMP<= P  
**區塊資料比較小於或等於**

**16位元指令 指令符號 執行條件**  
佔9個位址 BKCMP<= 連續執行  
BKCMP<=P 執行一次

**32位元指令 指令符號 執行條件**  
佔17個位址 DBKCMP<= 連續執行  
DBKCMP<=P 執行一次

**FNC 199**  
D BKCMP>= P  
**區塊資料比較大於或等於**

**16位元指令 指令符號 執行條件**  
佔9個位址 BKCMP>= 連續執行  
BKCMP>=P 執行一次

**32位元指令 指令符號 執行條件**  
佔17個位址 DBKCMP>= 連續執行  
DBKCMP>=P 執行一次

元件類別	內容														資料格式		
S <sub>1</sub> *	被比較資料的元件帶頭編號														BIN 16/32 位元		
S <sub>2</sub> *	比較資料的元件帶頭編號																
D*	比較結果的元件帶頭編號														位元		
n	比較的筆數														BIN 16/32 位元		

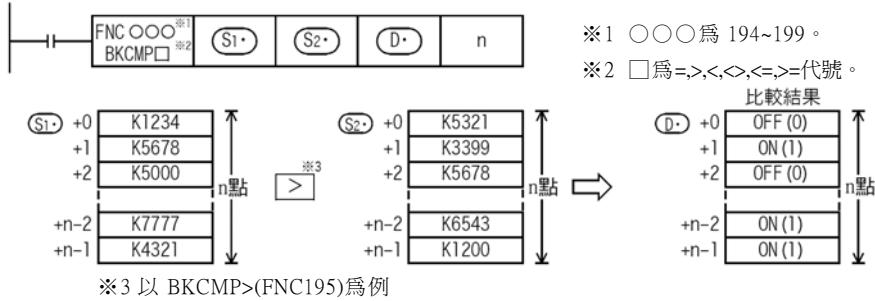
元件類別	位元元件								字元元件								其他						
	使用者				指定位數				使用者				特殊模組		間接指定		常數		實數	文字	指標		
X	Y	M	T	C	S	D□.b	KnX	KnY	KnM	KnS	T	C	D	R	U□￥G□	V	Z	修飾	K	H	E	"□"	P
S <sub>1</sub> *											●	●	●	●				●	●	●			
S <sub>2</sub> *											●	●	●	●				●					
D*	●	●			●	▲												●					
n																		●	●				

▲ : D□.b 不可再使用間接指定暫存器(V,Z)作修飾。

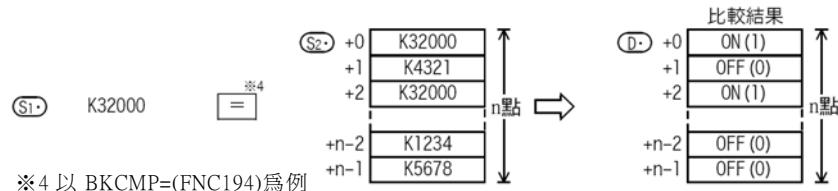
## 功能及動作

## 1. 16 位元運算(BKCMP=,&gt;,&lt;,&gt;,&lt;=,&gt;=/ BKCMP =P,&gt;P,&lt;P,&gt;P,&lt;=P,&gt;=P)

(1)  $S_1 \cdot$  指定被比較的帶頭元件編號， $S_2 \cdot$  指定比較的帶頭元件編號，n 指定比較的資料組數，比較結果反應至  $D \cdot$  所指定的 n 個輸出線圈上。



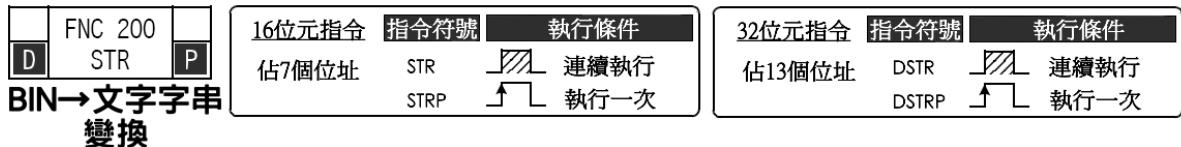
(2)  $S_1 \cdot$  可直接指定常數。



(3) 每個比較指令邏輯如下表所示。

指令	比較結果ON的條件	比較結果OFF的條件
BKCMP= (FNC194)	$S_1 \cdot = S_2 \cdot$	$S_1 \cdot \neq S_2 \cdot$
BKCMP> (FNC195)	$S_1 \cdot > S_2 \cdot$	$S_1 \cdot \leq S_2 \cdot$
BKCMP< (FNC196)	$S_1 \cdot < S_2 \cdot$	$S_1 \cdot \geq S_2 \cdot$
BKCMP<> (FNC197)	$S_1 \cdot \neq S_2 \cdot$	$S_1 \cdot = S_2 \cdot$
BKCMP<= (FNC198)	$S_1 \cdot \leq S_2 \cdot$	$S_1 \cdot > S_2 \cdot$
BKCMP>= (FNC199)	$S_1 \cdot \geq S_2 \cdot$	$S_1 \cdot < S_2 \cdot$

(4) 開始的 n 點比較結果都 ON 的時候，M8090=ON。

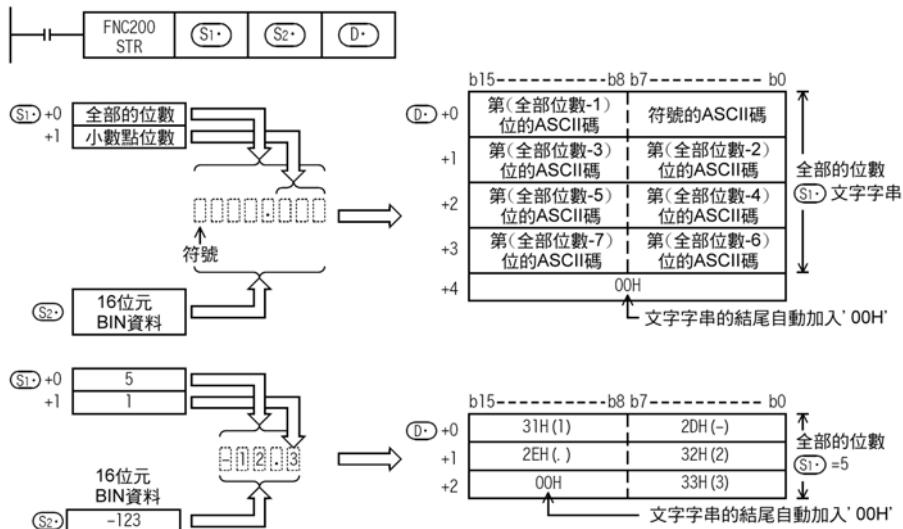


元件類別	內容	資料格式
S <sub>1</sub>	文字字串的全部位數及小數點位數	BIN 16 位元
S <sub>2</sub>	要變換成文字字串的 BIN 值	BIN 16/32 位元
D	變換結果的元件帶頭編號	文字字串

元件類別	位元元件								字元元件										其他							
	使用者								指定位數				使用者				特殊模組		間接指定		常數	實數	文字	指標		
	X	Y	M	T	C	S	D	b	KnX	KnY	KnM	KnS	T	C	D	R	U	¥	G	V	Z	修飾	K	H	E	"□"
S <sub>1</sub>													●	●	●	●					●					
S <sub>2</sub>									●	●	●	●	●	●	●	●		●		●	●	●	●	●	●	
D													●	●	●	●					●					

**功能及動作****1. 16 位元運算(STR/STRP)**

- S<sub>2</sub> • 指定的 BIN 值(16 位元)被變換成文字字串暫存於 D • 指定的元件當中，文字字串的格式(全部的位數及小數點位數)由 S<sub>1</sub> • 來指定。





**16位元指令 指令符號 執行條件**  
佔7個位址 VAL 連續執行  
VALP 執行一次

**32位元指令 指令符號 執行條件**  
佔13個位址 DVAL 連續執行  
DVALP 執行一次

### 文字字串→BIN 變換

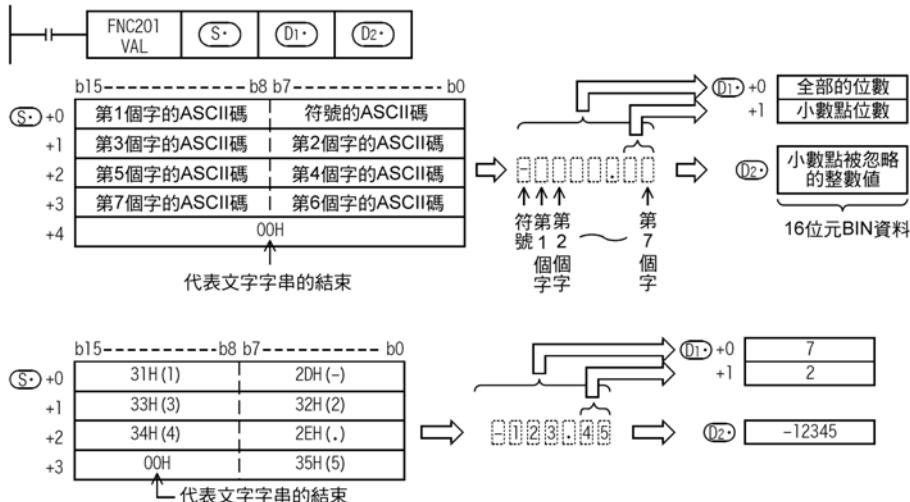
元件類別	內容	資料格式
S•	存放文字字串的元件編號	文字字串
D1•	變換結果的全部位數及小數點位數	BIN 16 位元
D2•	變換結果的 BIN 資料元件帶頭編號	BIN 16/32 位元

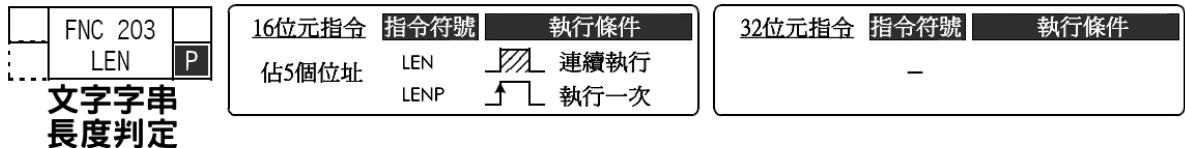
元件類別	位元元件								字元元件								其他							
	使用者				指定位數				使用者				特殊模組		間接指定		常數	實數	文字	指標				
	X	Y	M	T	C	S	D□·b	KnX	KnY	KnM	KnS	T	C	D	R	U□¥G□	V	Z	修飾	K	H	E	"□"	P
S•												●	●	●	●				●					
D1•												●	●	●	●				●					
D2•								●	●	●	●	●	●	●	●			●						

### 功能及動作

#### 1. 16 位元運算(VAL/VALP)

- 指定的文字字串(到'00H'為止)被變換成 BIN 值(16 位元)，變換結果的文字字串的格式被暫存於 D1•(全部的位數), D1•+1 (小數點位數)指定的元件當中、BIN 資料暫存於 D2•指定的元件當中。





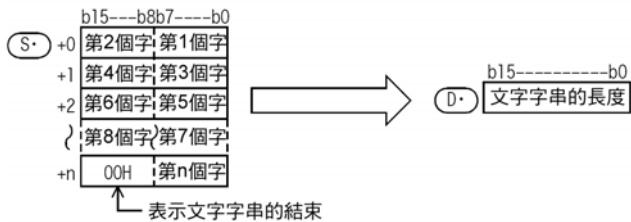
元件類別	內容												資料格式			
S•	存放文字字串的元件帶頭編號編號												文字字串			
D•	長度判定結果的元件帶頭編號												BIN 16 位元			

元件類別	位元元件						字元元件									其他											
	使用者						指定位數			使用者			特殊模組		間接指定			常數	實數	文字	指標						
	X	Y	M	T	C	S	D	□	b	KnX	KnY	KnM	KnS	T	C	D	R	U	□	¥G	V	Z	修飾	K	H	E	"□"
S•										●	●	●	●	●	●	●	●	●	●	●			●				
D•										●	●	●	●	●	●	●	●	●	●	●			●				

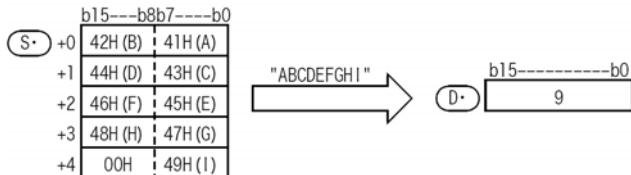
### 功能及動作

#### 1. 16 位元運算(LEN/LENP)

- 計算 S• 指定的文字字串的長度(字數)，計算結果暫存於 D• 指定的元件當中。



假設 S• 的內容為'ABCDEFGHI'時，D• 的內容為 9。



FNC 204 RIGHT 右取文字	16位元指令 指令符號 <b>P</b> 佔7個位址 RIGHT  連續執行 RIGHTP  執行一次	32位元指令 指令符號 <b>P</b> — —
--------------------------	---	--------------------------------

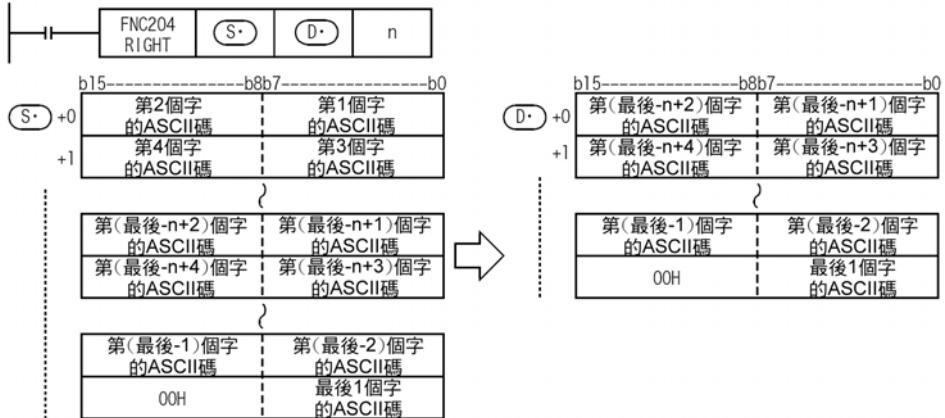
元件類別	內容												資料格式	
S•	取字對象的文字字串或存放文字字串的元件帶頭編號												文字字串	
D•	存放取出文字的元件帶頭編號													
n	取出的文字字數												BIN 16 位元	

元件類別	位元元件						字元元件								其他											
	使用者						指定位數				使用者		特殊模組		間接指定		常數	實數	文字	指標						
	X	Y	M	T	C	S	D	b	KnX	KnY	KnM	KnS	T	C	D	R	U	¥	G	V	Z	修飾	K	H	E	"□"
S•									●	●	●	●	●	●	●	●	●	●	●	●	●	●				
D•									●	●	●	●	●	●	●	●	●	●	●	●	●	●				
n																	●	●				●	●			

## 功能及動作

## 1. 16 位元運算(RIGHT/RIGHTP)

- 從 **S•** 指定的文字字串右邊開始算，取出 **n** 個文字並暫存於 **D•** 所指定的元件當中。



 左取文字	<b>16位元指令</b> 指令符號 <b>LEFT</b> 執行條件 佔7個位址 LEFT LEFTP	<b>32位元指令</b> 指令符號 <b>-</b> 執行條件 -
---	---	---

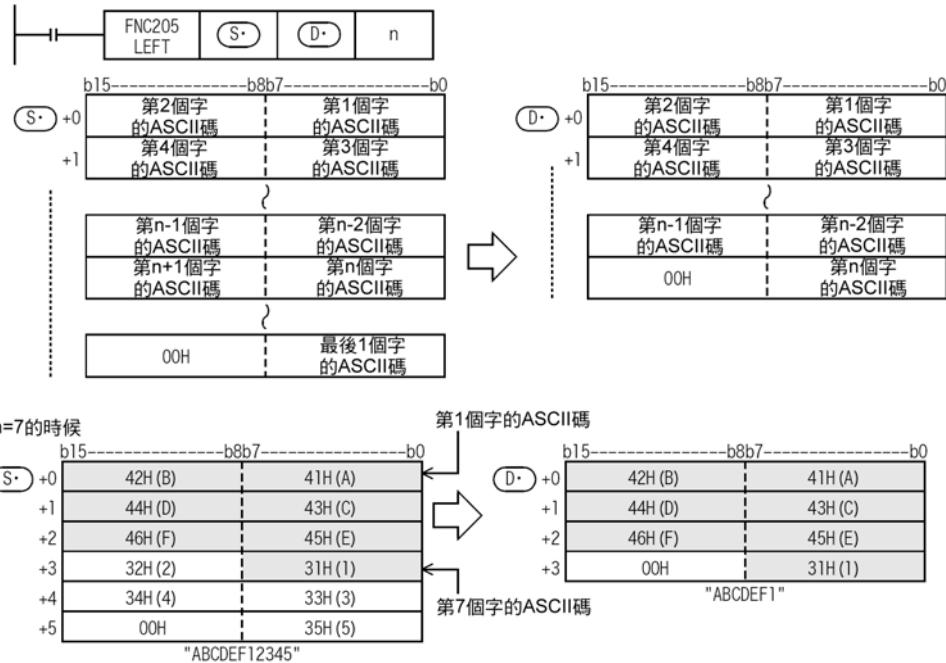
元件類別	內容												資料格式	
S·	取字對象的文字字串或存放文字字串的元件帶頭編號												文字字串	
D·	存放取出文字的元件帶頭編號												BIN 16 位元	
n	取出的文字字數												BIN 16 位元	

元件類別	位元元件						字元元件								其他									
	使用者				指定位數				使用者				特殊模組		間接指定		常數	實數	文字	指標				
	X	Y	M	T	C	S	D□b	KnX	KnY	KnM	KnS	T	C	D	R	U□¥G□	V	Z	修飾	K	H	E	"□"	P
S·								●	●	●	●	●	●	●	●				●					
D·									●	●	●	●	●	●	●				●					
n																				●	●			

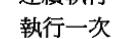
## 功能及動作

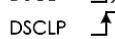
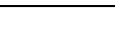
## 1. 16 位元運算(LEFT/LEFTP)

- 從 **S·** 指定的文字字串左邊開始算，取出 **n** 個文字並暫存於 **D·** 所指定的元件當中。





**16位元指令 指令符號**  **執行條件**  
 佔7個位址 SCL  連續執行  
 SCLP  執行一次

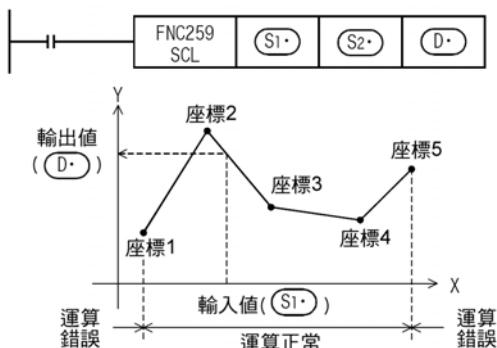
**32位元指令 指令符號**  **執行條件**  
 佔13個位址 DSCL  連續執行  
 DSCLP  執行一次

元件類別	內容												資料格式					
	輸入值(尺規 X 座標值)												BIN 16/32 位元					
	尺規資料表單的元件帶頭編號																	
	輸出值(尺規 Y 座標值)																	

元件類別	位元元件								字元元件								其他						
	使用者				指定位數				使用者				特殊模組		間接指定		常數	實數	文字	指標			
	X	Y	M	T	C	S	D□b	KnX	KnY	KnM	KnS	T	C	D	R	U□¥G□	V	Z	修飾	K	H	E	"□"
								●	●	●	●	●	●	●	●				●	●	●		
																			●				
								●	●	●	●	●	●	●	●				●				

**功能及動作****1. 16 位元運算(SCL/SCLP)**

 指定尺規座標資料表單的元件帶頭號碼， 指定輸入值(資料表單區域的 X 座標值)， 的內容顯示輸出值(對照資料表單之後所求得的 Y 座標值)。

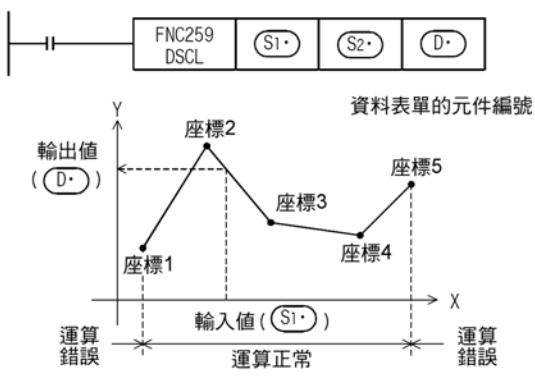


資料表單

設定項目		資料表單的元件編號
座標點數(如左圖為5點)		
座標1	x座標	 +1
	y座標	 +2
座標2	x座標	 +3
	y座標	 +4
座標3	x座標	 +5
	y座標	 +6
座標4	x座標	 +7
	y座標	 +8
座標5	x座標	 +9
	y座標	 +10

## 2. 32 位元運算(DSCL/DSCLP)

( $S2 \bullet +1, S2 \bullet$ ) 指定尺規座標資料表單的元件帶頭號碼，( $S1 \bullet +1, S1 \bullet$ ) 指定輸入值(資料表單區域的 X 座標值)，( $D \bullet +1, D \bullet$ ) 的內容顯示輸出值(對照資料表單之後所求得的 Y 座標值)。

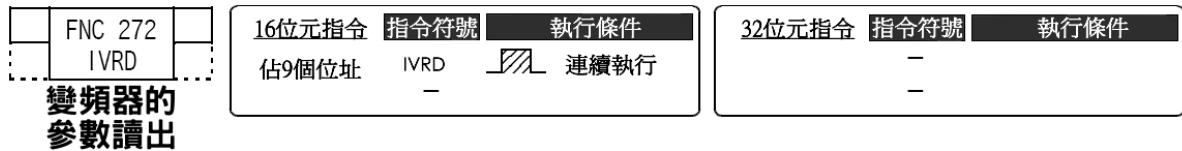


資料表單	
設定項目	資料表單的元件編號
座標點數(如左圖為5點)	[ $S2 \bullet +1, S2 \bullet$ ]
座標1	x座標 [ $S2 \bullet +3, S2 \bullet +2$ ]
	y座標 [ $S2 \bullet +5, S2 \bullet +4$ ]
座標2	x座標 [ $S2 \bullet +7, S2 \bullet +6$ ]
	y座標 [ $S2 \bullet +9, S2 \bullet +8$ ]
座標3	x座標 [ $S2 \bullet +11, S2 \bullet +10$ ]
	y座標 [ $S2 \bullet +13, S2 \bullet +12$ ]
座標4	x座標 [ $S2 \bullet +15, S2 \bullet +14$ ]
	y座標 [ $S2 \bullet +17, S2 \bullet +16$ ]
座標5	x座標 [ $S2 \bullet +19, S2 \bullet +18$ ]
	y座標 [ $S2 \bullet +21, S2 \bullet +20$ ]

## 3. 尺規對照用的資料表單

$S2 \bullet$  指定尺規座標資料表單的元件帶頭號碼， $S2 \bullet$  指定座標點數，  
 $(S2 \bullet +1) \sim (S2 \bullet +2n)$  設定資料表單內各點的座標值(X,Y)

設定項目		資料表單的元件編號	
		16 位元演算	32 位元演算
座標點數	$S2 \bullet$	[ $S2 \bullet +1, S2 \bullet$ ]	
座標1	x座標	$S2 \bullet +1$	[ $S2 \bullet +3, S2 \bullet +2$ ]
	y座標	$S2 \bullet +2$	[ $S2 \bullet +5, S2 \bullet +4$ ]
座標2	x座標	$S2 \bullet +3$	[ $S2 \bullet +7, S2 \bullet +6$ ]
	y座標	$S2 \bullet +4$	[ $S2 \bullet +9, S2 \bullet +8$ ]
{	{	{	{
座標n	x座標	$S2 \bullet +2n-1$	[ $S2 \bullet +4n-1, S2 \bullet +4n-2$ ]
	y座標	$S2 \bullet +2n$	[ $S2 \bullet +4n+1, S2 \bullet +4n$ ]

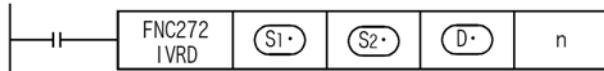


元件類別	內容	資料格式
S1*	變頻器的站號(K0~K31)	BIN 16 位元
S2*	變頻器的參數編號	
D•	存放讀出值的元件編號	
n	使用 ch(K1:ch1, K2:ch2)	

元件類別	位元元件								字元元件								其他						
	使用者				指定位數				使用者				特殊模組		間接指定		常數	實數	文字	指標			
	X	Y	M	T	C	S	D□·b	KnX	KnY	KnM	KnS	T	C	D	R	U□¥G□	V	Z	修飾	K	H	E	"□"
S1*													●	●	●				●	●	●		
S2*													●	●	●				●	●	●		
D•													●	●	●				●				
n																				●	●		

**功能及動作****1. 16 位元運算(IVDR)**

n 指定與變頻器連接的通信埠編號，以 S1• 指定的變頻器站號、S2• 指定的變頻器參數編號，將變頻器的參數內容讀出並暫存於 D• 指定的元件當中。



※ 1. 連接對象：三菱電機變頻器 FREQROL-F700※3/A700※3 /V500/F500/A500/E500/S500(必須附通信供能的機種)系列。

※ 2. 與 FREQROL-F700 通信的 FX3U/FX3UC 機種必須是 Ver.2.00 以上的機種。



16位元指令 指令符號 執行條件  
佔9個位址 IVWR 連續執行

32位元指令 指令符號 執行條件  
— —

### 變頻器的參數寫入

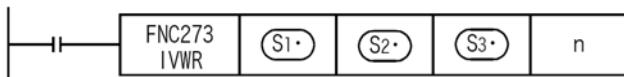
元件類別	內容	資料格式
S <sub>1</sub> *	變頻器的站號(K0~K31)	BIN 16 位元
S <sub>2</sub> *	變頻器的參數編號	
S <sub>3</sub> *	寫入至變頻器參數的設定值	
n	使用 ch(K1:ch1, K2:ch2)	

元件類別	位元元件								字元元件								其他								
	使用者								指定位數				使用者				特殊模組		間接指定			常數	實數	文字	指標
	X	Y	M	T	C	S	D□b	KnX	KnY	KnM	KnS	T	C	D	R	U□¥G□	V	Z	修飾	K	H	E	"□"	P	
S <sub>1</sub> *															●	●	●			●	●	●			
S <sub>2</sub> *															●	●	●			●	●	●			
S <sub>3</sub> *															●	●	●			●	●	●			
n																				●	●				

### 功能及動作

#### 1. 16 位元運算(IVWR)

n 指定與變頻器連接的通信埠編號，以 S<sub>1</sub>• 指定的變頻器站號、S<sub>2</sub>• 指定的變頻器參數編號，將變頻器的參數內容讀出並暫存於 S<sub>3</sub>• 指定的元件當中。



※ 1. 連接對象：三菱電機變頻器 FREQROL-F700※3/A700※3/V500/F500/A500/E500/S500(必須附通信供能的機種)系列。

※ 2. 與 FREQROL-F700 通信的 FX3U/FX3UC 機種必須是 Ver.2.00 以上的機種。