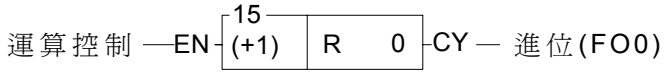
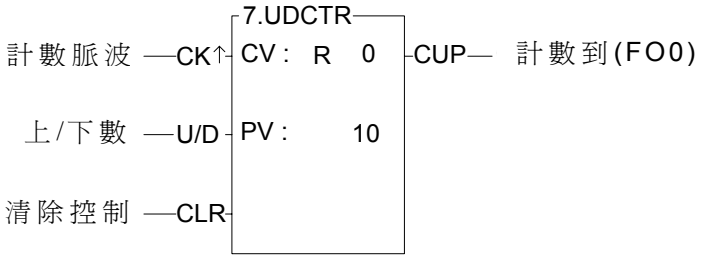
第 5 章：應用指令說明

5.1 應用指令之通則

FBs 系列 PLC 之應用指令可分爲輸入控制、指令號碼名稱、運算元及功能輸出四部分。而各指令之輸入控制、運算元、及功能輸出之數目不盡相同(請參閱各指令說明)。在 FP-07C 程式書寫器上除常用之 T、C、SET、RST 四指令及 SFC 指令有對應之專用按鍵，可直接按鍵輸入外，其他之應用指令均需以指令號碼輸入，不能以指令名稱輸入。如下例：

階　　梯　　圖	FP-07C 簡碼指令
例 1：單輸入指令	
	FUN 15 D: R 0
例 2：多輸入指令	
	FUN 7 CV: R 0 PV: 10

註：在本手冊之簡碼指令欄位中，凡有實線方框框住之字樣（如上例 D:、CV:、Pr:等）係 FP-07C 為方便您輸入而自動顯示之運算元名稱導引字，非使用者所鍵入者。

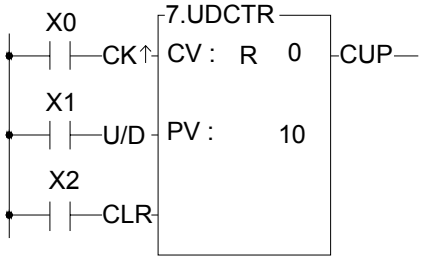
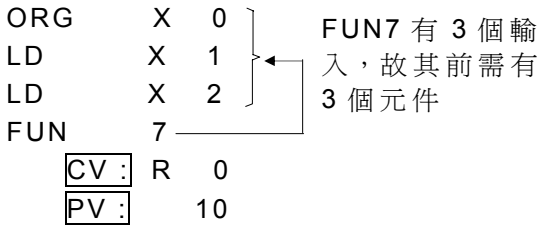
5.1.1 輸入控制

FBs-PLC 除 7 個無輸入控制之應用指令外，其他應用指令至少有一個輸入控制，最多爲四個。應用指令係依輸入控制信號之組合以決定該指令是否執行，以及執行何種運算。在 PRO-LADDER 套裝軟體上及階梯圖程式印出時，所有之應用指令符號之輸入控制及功能輸出端子上均有加註英文註解簡寫，以註明該端子是何種功能控制或輸出，以利於記憶和閱讀，如上圖例 2 第一個輸入標註〝CK↑〞，表示計數脈波 Clock 由 0→1（升緣）時，該計數器才計數一次，第二個輸入標註〝U/D〞斜線上方 U 表上數 Up，下方 D 表示下數 Down，若此輸入爲 1 則當計數脈波 CK↑來時，該計數器值會加 1，反之若爲 0 則減 1，第三個輸入標示〝CLR〞，表示清除 Clear，即當此輸入爲 1 時，該計數器之計數值會被清爲 0。其他應用指令之輸入控制註解請參閱各指令說明。

註：無輸入控制指令係指該指令需直接接於母線，不能串接輸入控制元件，亦無功能輸出。該指令本身單獨形成一個網路。有 MCE、SKPE、LBL、RTS、RTI、FOR、NEXT 等 7 個無輸入控制指令，請參閱第 6 及 7 章各該指令之說明。

所有應用指令之各〝輸入控制〞均應有元件連接，否則會出現語法錯誤。如下圖例 3，FUN7 爲三輸入之應用指令，在 FUN7 指令前面之三個元件（ORG X0, LD X1, LD X2）分別對應到 FUN7 之第一個輸入 CK↑，第二個輸入 U/D 和第三個輸入 CLR。

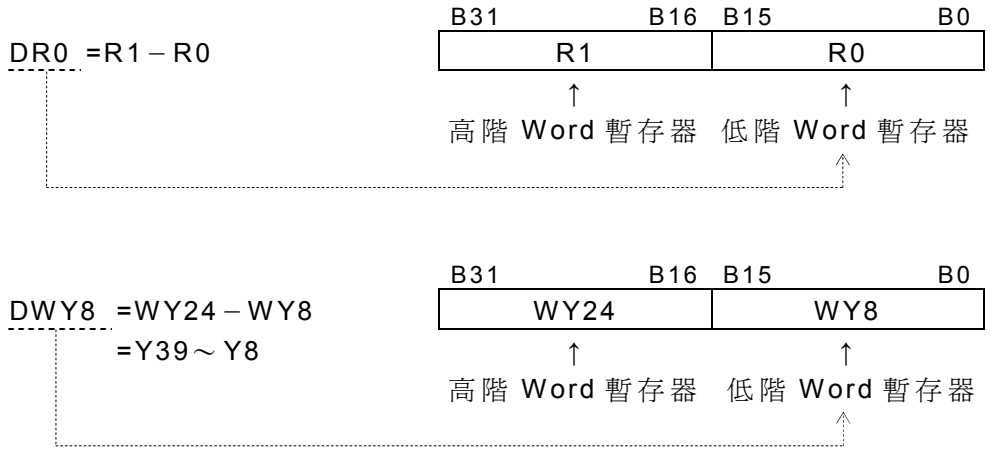
例 3：

階　梯　圖	FP-07 簡碼指令
	

5.1.2 指令號碼與衍生指令

FP-07C 除前述 9 個指令以專用按鍵輸入外，其他之應用指令均需以〝指令號碼〞來輸入，在〝指令號碼〞後，尙可加上 D、P 或 DP 等尾碼，而衍生出另外三種不同之指令，茲敘述如下：

D: 表示 Double Word，雙字元組（32 位元）之意。在 FBs-PLC 中之暫存器均以字元組 WORD（16 位元）爲基本單位，即所有 R、T、C 暫存器（C200～C255 除外）均爲 16 位元長度，例如 R0、R1、T0 …… 等。若需 32 位元長度之暫存器，則必須由兩個連續之 16 位元暫存器合併起來而形成如 R1-R0、R3-R2、 …… 等，針對這種連續兩個 16 位元暫存器組成之雙字元組暫存器，我們以該雙字元組暫存器之低階暫存器號碼（如 R1-R0 取 R0，R3-R2 取 R2）加上 D 表示之（如 DR0 表示 R1-R0，DR2 表示 R3-R2），例如您在監視模式(MON)下鍵入如下之 DR0 或 DWY8，將會顯示 32 位元（R1-R0，或 WY24-WY8）長度之數值。

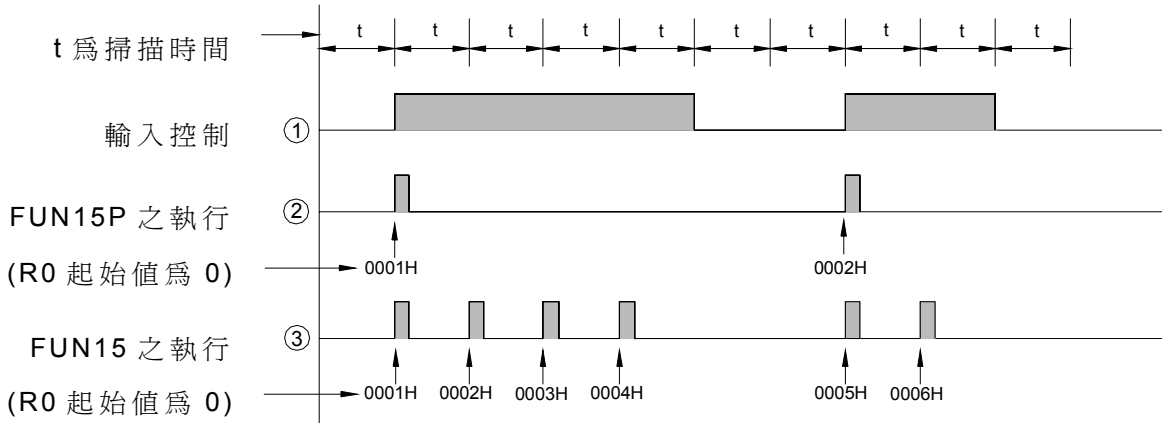


註：在階梯圖或簡碼指令之表示，為方便區別 16 位元或 32 位元指令，我們在〝指令號碼〞後面加 D 尾碼以表示 32 位元指令，其來源或目的運算元的長度當然也是 32 位元。但在運算元（如 S，D ……）欄上只標示 Double Word 之低階暫存器號碼，如 6-6 頁例 4 中〝被加數〞Sa：R0 因該指令 FUN11D，有 D 尾碼，故所有來源或目的運算元均無冠上 D 字，亦即 Sa 為 DR0=R1-R0，Sb 為 DR2=R3-R2 …· 但非來源或目的運算元如指標 Pr，數值 N，長度 L ……等則無論 D 或非 D 指令均固定只有 16 位元長度，請特別注意。在 16 位元指令因其運算元長度只有一個 Word，也正是運算元欄上所標示之暫存器號碼，如例 1 中之 D=R0。

P：表示 Pulse（脈沖）模式運作，也就是每當輸入控制由 0→1 瞬間（升緣↑）該指令即執行一次，如上圖例 1 若指令號碼加上 P 碼（即 FUN 15P），則只有在輸入控制信號之升緣（0→1）時 FUN15P 才執行一次。若指令號碼後無 P 尾碼，則爲連續執行模式，即只輸入控制爲 1，PLC 每次掃描到該指令均會執行一次，一直至輸入控制變爲 0 爲止。（所有脈沖輸入端均標有〝↑〞符號，如 CK↑、EN↑、TG↑ ……）。在本手冊之應用指令說明中之輸入敘述有如下之敘述例：

● 當運算控制〝EN〞=1 或〝EN↑〞（P 指令）由 0→1 時，……

前者即表非 P 指令（連續模式）之執行條件，後者即爲 P 指令（脈沖模式）之執行條件。下列波形圖爲上節範例 1（FUN15）工作在 P 模式和非 P 模式下其執行結果(R0)之比較。



DP：表該指令爲 32 位元指令，且爲脈沖模式運作。

註：實際控制應用上大部分之應用指令都可使用 P 指令，在程式設計時請儘可能使用 P 指令，以節省程式執行時間。

5.1.3 運算元

運算元爲指令運算時之參考或寫入之對象。可分爲只供參考，內容不會因指令運算而改變之來源運算元（Source，簡稱 S）及用來儲存運算結果之目的運算元（Destination 簡稱 D）兩大類。以下就 FBs 系列 PLC 應用指令中，主要之運算元名稱及性質作說明，並將可當運算元之接點、線圈或暫存器之類別範圍分述如下：

■ 主要運算元名稱及性質：

簡寫	名　　稱	說　　明
S	來源運算元 (Source)	S 爲指令運算中之資料讀取、參考的對象，其內容不會因運算而改變，若不只一個以上，則以註腳區分，如 Sa、Sb。
D	目的運算元 (Destination)	用以存放指令運算結果之區域，其原始資料在運算後會破壞，只有能寫入之線圈或暫存器才能當目的運算元。
L	長　度 (Length)	用以表示一連串資料或列表(Table)之長度（範圍），可爲常數或變數。
N	數　值 (Number)	用以指定次數，個數（如第 N 個位元）等之固定數字，若不只一個，則以註腳區分如 Na、Nb、Ns、Nd 等。
Pr	指　標 (Point)	用以指定一串資料或列表中之某個資料或暫存器，通常 Pr 值爲可變，故不能爲常數或輸入暫存器。
CV	現在值	用於 T、C 中，只能爲可寫入之暫存器。
PV	設定值	用於 T、C 中，只供參考比較用。
T	列　表 (Table)	列表是一連續暫存器組合之稱，其運作單位係以字元組或雙字元組爲單位，若不只一個，則以註腳區分，如 Ta、Tb、Ts、Td 等。
M	矩　陣 (Matrix)	矩陣亦是一連續暫存器組合之稱，但是其運作單位是以位元爲單位。若不只一個則以 Ms、Md、Ma、Mb 等表示。

除上述主要運算元外，尚有用以指定特定用途之運算元，如 Fr 表頻率、ST 表堆疊、QU 表示 QUEUE ····· 等，請參閱各指令之說明。

■ 運算元類別與範圍：應用指令之運算元類別有 a.單點(數位)　b.暫存器　c.常數　三種

a.單點（數位）運算元：

在應用指令中，有單點運算元者（即其運算元只影響某一單點者）僅有 SET、RST、DIFU、DIFD、TOGG 五個指令，而且只能對 Y△△△（外部輸出）、M△△△△（內部及特殊）、S△△△（步進）三類型之繼電器運作。下表爲可當此五指令之單點運算元之種類及範圍，細部解釋請參閱此五指令之說明。

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

〝○〞符號表 D 可用該類別之線圈當運算元。在 SM 欄位中〝○〞上方標有〝*〞符號，表示在 SM 中禁止寫入之特殊繼電器不得當作 D 運算元，請參考 2-3 頁〝特殊繼電器明細〞。

b.暫存器運算元：

應用指令中之運算元主要爲暫存器運算元。暫存器運算元又分爲兩類，一爲原本就以 Word 或 2 Words 爲單位之暫存器(R、T、C)。另一則爲由 16 或 32 個單點(X、Y、M、S)組成 Word 或 2 Words 之暫存器(WX、WY、WM、WS)。下表爲在本手冊中用以表示各指令之運算元所能容許之暫存器類別及其範圍之範例：

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

〝○〞符號表示可以以該類別之暫存器當運算元。在 SR 和 D 交會之欄位中，〝○〞符號上方標有〝*〞符號，表示 D 運算元若爲特殊暫存器 SR 時，應扣除不可寫入之暫存器，請參閱 2-7 頁〞特殊暫存器明細〞。

※R5000～R8071 不是規劃爲唯讀暫存器時，可當一般暫存器使用（可讀寫）

註 1：凡有 W 開頭之暫存器（WX、WY、WM、WS）表示此暫存器係由 16 個單點組成 Word 之暫存器。例如 WX0 表示由 X0（位元 0）～X15（位元 15）組成之暫存器，WY144 表示由 Y144（位元 0）～Y159（位元 15）所組成之暫存器。但注意單點之號碼必須為 8 之倍數（如 0、8、16、24 …… 等）才允許。

註 2：表中最後一個暫存器（Word），不能當 32 位元運算元，因 32 位元運算元需有連續兩個 Word 的長度才行。

註 3：TMR（T0～T255）和 CTR（C0～C255）為計時器和計數器專用之暫存器，雖亦可當一般暫存器使用，但會造成系統複雜，除錯困難，因此除 T 或 C 指令外，其他指令應避免寫入 TMR 或 CTR。

註 4:T0～T255 和 C0～C199 均為 16 位元長度，而 C200～C255 限定為 32 位元長度，故不能當 16 位元運算元。

註 5：暫存器運算元除如上述直接以暫存器號碼（位址）來指定外，對於 R0～R8071 範圍內之暫存器運算元尚可結合指標暫存器 V、Z 或 P0~P9 來作間接定址指定。利用指標暫存器（XR）作間接定址之說明請參考下節（5.2 節）之範例。

c.常數運算元：

在 16 位元中之常數範圍最大爲-32768～32767，32 位元之範圍爲-2147483648～2147483647，而某些指令只能爲正常數，因此我們以下列敘述表示 16 或 32 位元之常數範圍。

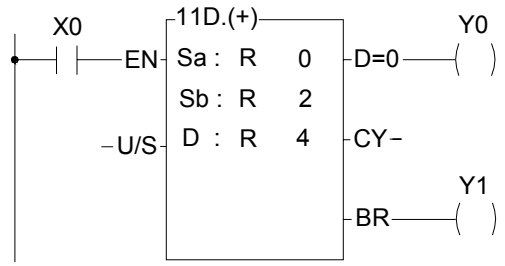
常　數　類　別	範　　圍
16 位元正負數	-32768～32767
16 位元正數	0～32767
32 位元正負數	-2147483648～2147483647
32 位元正數	0～2147483647
16／32 位元正負數	-32768～32767 或 -2147483648～2147483647
16／32 位元正數	0～32767 或 0～2147483647

此外有某些特定之運算元長度大小不一（如長度 L、位元數 …… N 等）將於各該運算元之欄位上直接標示範圍，請參考個別之指令說明。

5.1.4 功能輸出(FO)

簡稱 FO(Function Output)爲應用指令運算結果或狀態之輸出，如同〝輸入控制〞一樣。在 WinProladder 及程式印出之階梯圖應用指令上，其功能輸出上亦有英文註解說明該 FO 爲何種功能，如上圖例 1 之 CY，例 2、例 3 之 CUP 及下圖例 4 之 D=0、CY、BR 均是。功能輸出 FO 最多只有 4 個（即 FO0～FO3），其編號順序是由上而下，第一個 FO 爲 FO0，第二個爲 FO1，最後一個爲 FO3。FO 狀態之取出必須用 FO 指令（在 FP-07C 程式書寫器上有 FO 專用按鍵），不使用之功能輸出可空著不接任何元件，如下圖例 4 之 FO1(CY)即是。

例 4：

階　梯　圖	簡碼指令
	ORG X 0 FUN 11D Sa: R 0 Sb: R 2 D : R 4 FO 0 OUT Y 0 FO 2 OUT Y 1

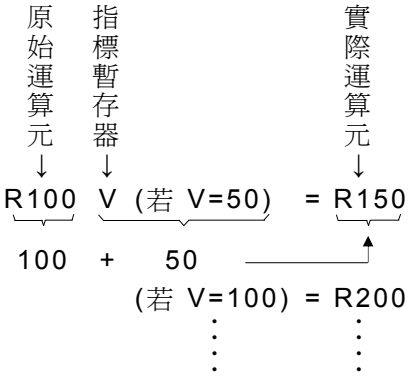
當 M1919=0 時，FO 狀態只有在該指令被執行時才會更新，然後就一直保持至該指令下一次被執行時（記憶保持），始由新產出之 FO 狀態所更新。

當 M1919=1 時，應用指令不執行時，FO 狀態清除為 0（無記憶保持）。

5.2 利用指標暫存器（XR）作間接定址

在 FBs-PLC 應用指令中，有些運算元可以結合指標暫存器（V 或 Z）而作間接定址之指定（在每個指令說明欄之運算元敘述中會註明）但能夠結合指標暫存器作間接位址指定之運算元只限定爲 R0～R8071 範圍內之暫存器（其他運算元如單點、常數、D0～D3071 等均不能作間接位址指定）。

指標暫存器 XR 共有 12 個（V、Z、P0~P9），實際上在 FBs-PLC 之 V 暫存器就是特殊暫存器（R3840～R4167）中之 R4164，而 Z 暫存器則爲 R4165。運算元結合指標暫存器作間接定址之表示方式是原運算元後緊接 V 或 Z 來表示：



如上圖示，只要變更 V 之值即可變更運算元之位址，利用此功能結合 FBs-PLC 之應用指令，您可以極簡易之指令，達成功能強大、極具效率之控制應用，如下圖程式例，您只須以一個區塊搬移指令（BT_M）即達成諸如停車管理系統之動態區塊資料顯示。

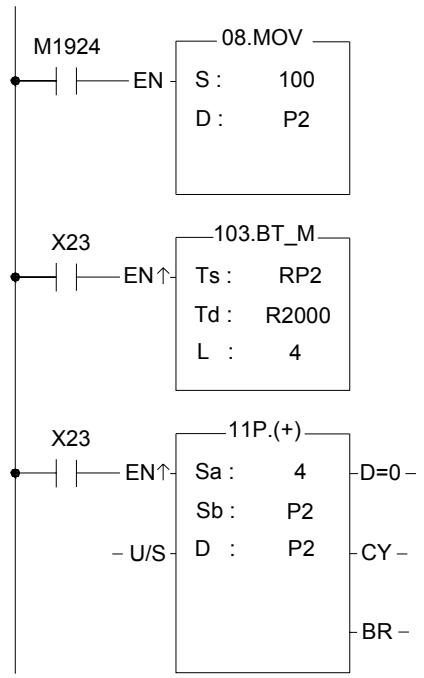
指標暫存器 P0~P9 應用說明：

● 在間接定址應用中，RXXXX 暫存器可以結合指標暫存器 V、Z 和 P0～P9 作間接定址應用；DXXXX 暫存器不可以結合指標暫存器 V、Z 作間接定址應用，但可以結合 P0～P9 作間接定址應用。

● 當 RXXXX 暫存器要結合 V、Z 作間接定址應用時，例如 R0 要結合 V、Z 做間接定址應用，則所輸入之格式爲 R0V(當 V=100 時，則指向 R100)或 R0Z(當 Z=500 時，則指向 R500)；而欲結合 P0～P9 作間接定址應用時，則所輸入之格式爲 RPn (n 爲 0～9)或爲 RPmPn (m,n 爲 0～9)，例如 RP5 (若 P5 內容爲 100，則指向 R100) 或 RP0P1(若 P0 內容爲 100，P1 內容爲 50，則指向 R150)。

● 當 DXXXX 暫存器要結合 P0～P9 作間接定址應用時，則所輸入之格式爲 DPn (n 爲 0～9)或爲 DPmPn (m,n 爲 0～9)，例如 DP3 (若 P3 內容爲 10，則指向 D10) 或 DP4P5(若 P4 內容爲 100，P5 內容爲 1，則指向 D101)。

● P0～P9 指標暫存器可同時結合運用，例如 P2=20、P3=30，當 RXXXX 或 DXXXX 暫存器一次結合兩個指標暫存器時，RP2P3 就會指向 R50，DP2P3 就會指向 D50；也就是說兩個指標值之間的關係是相加的。

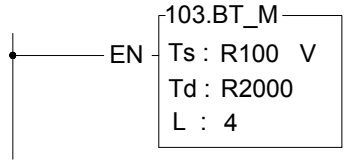


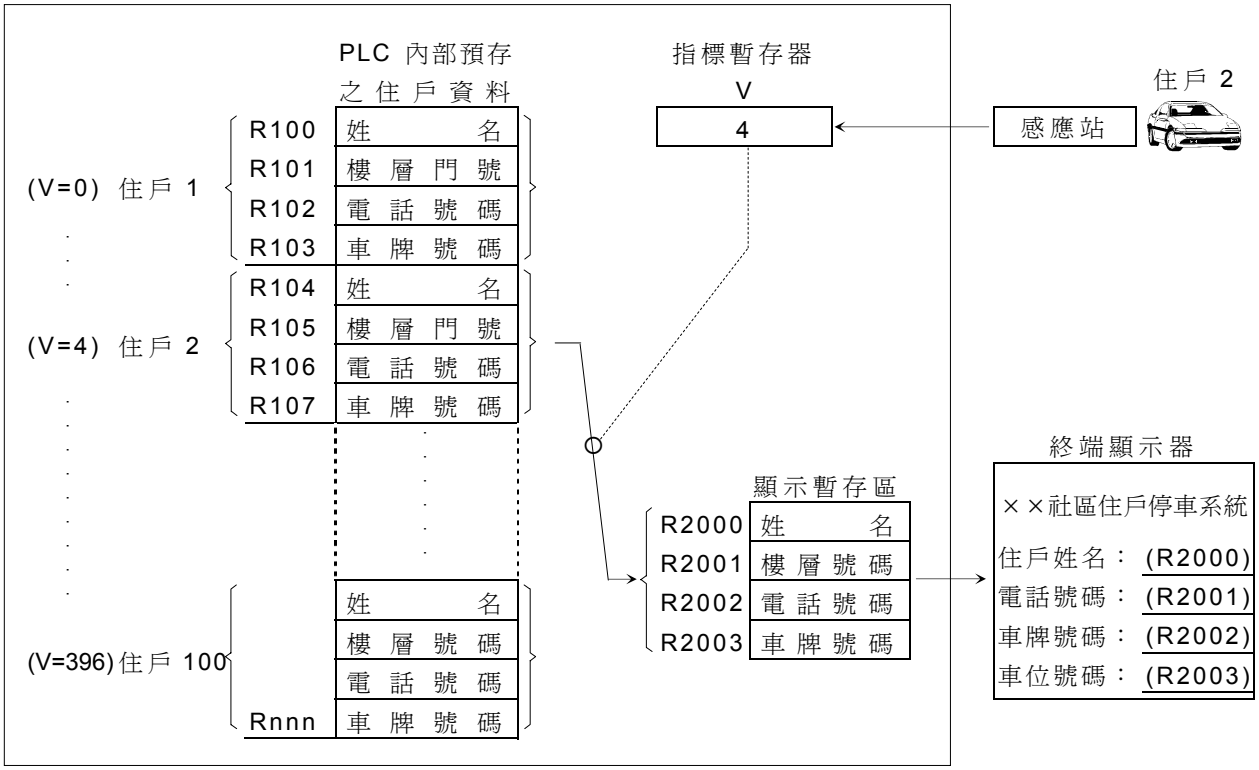
1. 開機時 M1924 起始脈波將 100 搬入指標暫存器 P2。

2. 當 X23 由 0→1 時，Fun103 將由 R100(因爲 P2=100)開始，一次 4 個暫存器的長度，依序搬到 R2000。也就是說第一次將 R100~R103 搬到 R2000~R2003，第二次將 R104~R107 搬到 R2000~R2003，第三次將 R108~R111 搬到 R2000~R2003 依此類推。

3. Fun11 用來將指標每次增加 4 個 word 用，亦即 X23 每"ON"一次，P2 之指標值便加上 4。

間接定位程式範例：

階　梯　圖	簡碼指令
	ORG SHORT FUN 103 Ts : R100V Td : R2000 L : 4



程式說明：上例假設某社區住戶之自動化停車場管理系統，共有 100 個住戶停車位，每個住戶均有 1 組基本資料，分別爲住戶姓名，電話號碼，車牌號碼，停車位號碼等 4 個，每組資料如上圖示佔用連續 4 個 PLC 內部暫存器，共佔用 R100～R499 等 400 個暫存器，每個住戶均有一不同卡號之卡片，以供進出口門禁及停車場進出之感應通行使用，其卡號爲 0，4，……，396 等 100 種，在 PLC 感應到卡號後，將之存入指標暫存器〝V〞，而在管理員處之終端機（LCD 或 CRT）顯示資料則固定由 PLC 內部之 R2001～R2003 來抓取並顯示，如本例感應到住戶 2 之卡片，其值=4，因此 V 暫存器=4，PLC 立即將 R104～R107 之資料搬移至顯示暫存區（R2000～R2003），因此管理員處之終端機可在感應到住戶 2 之卡片之同時，立即將其資料顯示在終端機上。

⚠ 警告

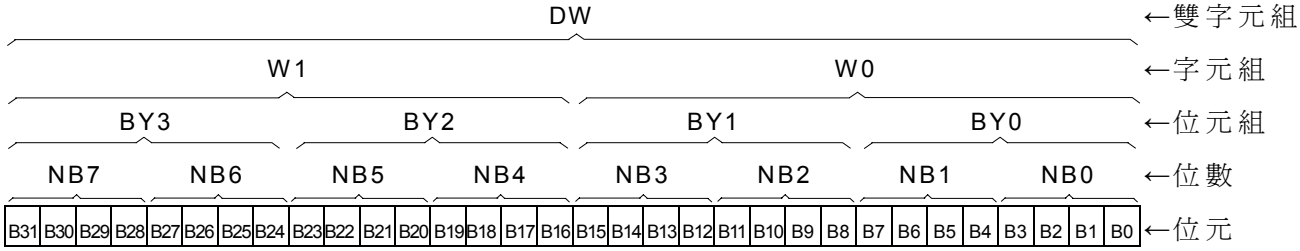
1. 運用指標暫存器作間接定址之應用雖然功能強大、彈性方便，但相對地 V、Z 內容值之任意變化可能對正常資料區之錯誤寫入有著極大之殺傷力，因此使用者在使用時應特別小心。
2. 在間接定址所能定址之資料暫存器（R0～R8071）範圍中，暫存器 R3840～R4167 等之 328 個暫存器（即 IR，OR，SR）爲系統或 I/O 用之重要暫存器，任意對此等暫存器之寫入，將可能使系統或 I/O 錯誤，造成重大之災害。鑑於 V、Z 值對暫存器位址變化之靈活變化，使用者可能不易察覺或掌握，因此 FBs-PLC 對間接定址之寫入動作會自動檢查寫入目的（Destination）是否在上述之 R3840～R4067 範圍內，若是則不執行該寫入動作，並將〝間接定址不合法寫入〞旗標 M1969 設爲 1。若應用上確實需要對 R3840～R4067 之暫存器作寫入，請使用非間接定址之指令來執行。

5.3 數目系統

5.3.1 二進制數值及其術語

二進制(Binary)爲數位計算機之基本數目系統，PLC 是由數位計算機所構成，自然亦採用二進制，爲便於表示及掌握二進數值，首先需了解如下之術語：

- 位元：（Bit 簡寫 B，如 B0，B1 ‧‧‧‧‧‧‧‧‧ ）位元爲二進制數值之最基本單位，其狀態非 1 即 0。
- 位數：（Nibble 簡寫 NB，如 NB0，NB1 ‧‧‧‧‧‧‧‧‧ ）由連續的 4 個位元所組成（如 B3～B0）可用以表示一個位數之 10 進制數字 0～9 或 16 進制之 0～F。
- 位元組：（Byte 簡寫 BY，如 BY0，BY1， ‧‧‧‧‧‧‧‧‧ ）是由連續之兩個位數所組成（亦即 8 個位元，例如 B7～B0）。可表示 16 進制之兩個位數值 00～FF。
- 字元組：（Word 簡寫 W，如 W0，W1， ‧‧‧‧‧‧‧‧‧ ）是由連續之兩個位元所組成（亦即 16 個位元例如 B15～B0）可表示 16 進制之 4 個位數值 0000～FFFF。
- 雙字元組：（Double word 簡寫 DW，如 DW0，DW1 ‧‧‧‧‧‧‧‧‧ ）是由連續之兩個字位元組所組成（亦即 32 個位元，例如 B31～B0）可表示 16 進制之 8 個位數值 00000000～FFFFFFFF。



- 浮點數：（Floating Point Number）也是由連續之兩個字位元組所組成。浮點數所能表示的最大範圍爲$\pm(1.8*10^{-38}\sim 3.4*10^{38})$，有關詳細之格式說明請參考 5.3.6 節。

5.3.2 FBs-PLC 之數碼

FBs-PLC 內部之數值運算或儲存全部採用二進值（Binary），因此外界輸入 PLC 內部之數值必須轉換成二進碼 PLC 才能處理，同樣地自 PLC 內部取出之數值結果亦均爲二進值，因此無論 FP-07C 或 WinProladder 其所有數目最終均須化成二進制才能輸入 PLC。但因二進值極難輸入和閱讀，因此 FP-0C7 或 WinProladder 在人機界面（數值輸入或顯示）部分均提供使用者以人們熟悉之 10 進值（Decimal）或 16 進制（Hexadecimal）來輸入或顯示，但實際上之數值處理全部都以二進碼來進行。

> 註：若您的數值輸入或顯示不是透過 FP-07C 或 WinProladder（例如以指撥開關或 7 段顯示器透過 I/O 點而輸入 PLC 或自 PLC 取出），那麼您就得自己藉著階梯圖程式指令來處理二進制和 10 進制間之轉換，使您雖不透過 FP-07C 或 WinProladder 亦能以 10 進制來輸入及得到 10 進制之輸出顯示，請參考 FUN20(BIN→BCD)和 FUN21(BCD→BIN)之說明。

5.3.3 數值之範圍

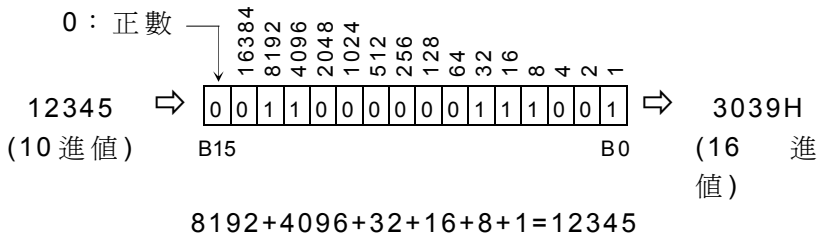
如前述 FBs-PLC 內部全部採用二進制（BCD 值只是爲適合人們習慣，而由二進值轉成適合人們閱讀之顯示用數碼而已）。在 PLC 之數值有 16 位元、32 位元以及浮點數三種數值，分別能表示如下之範圍。

16 位元	－32768～32767
32 位元	－2147483648～2147483647
浮點數	$\pm(1.8*10^{-38}\sim 3.4*10^{38})$

5.3.4 數值之表示 (初學者請略過本節)

以下各節將敘述 16 位元及 32 位元數值之表示方式與格式。以供使用者能深入了解數值之運算過程及結果而能應付各種複雜之應用需求。

無論是 16 位元或 32 位元之數值，均以其最高位元 MSB（16 位元之 B15，32 位元之 B31）表示該數值之正負（0：正數，1：負數），剩下之位元（B14～B0 或 B30～B0）才真正用以表示數值大小，茲以 16 位元爲例說明如下：（32 位元之作法亦同，只是長度倍增而已）。

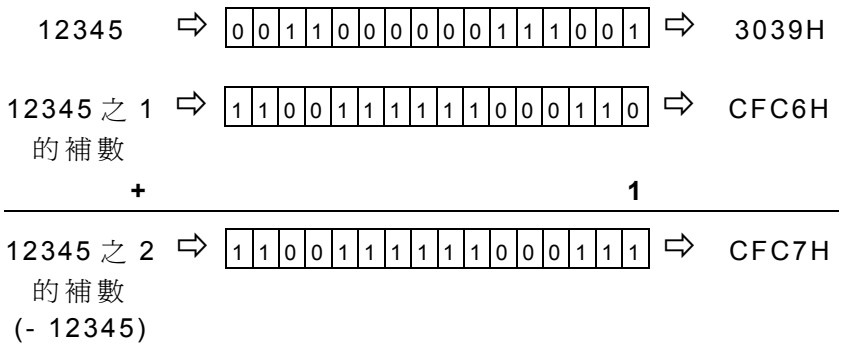


8192+4096+32+16+8+1=12345

如上例，無論 16 位元或 32 位元，其二進制之位元由最低位元 LSB（B0）開始，B0 代表 1，B1 代表 2，B2 代表 4，B3 代表 8，‥‥‥ 餘此類推，而其數值則爲所有爲 1 之位元所代表數值之總和。

5.3.5 負數之表示及取得 (初學者請略過本節)

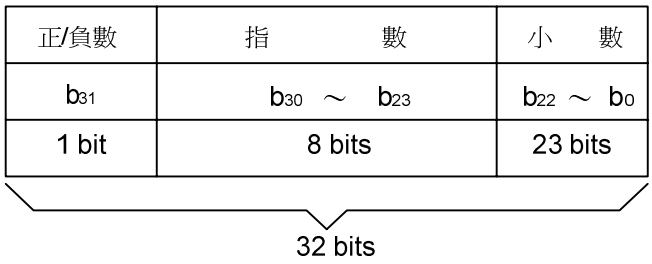
如前述當 MSB 爲 1，則此數爲負數，FBs-PLC 之負數係以〝2 的補數〞(2'S Complement) 來表示。所謂 2 的補數，係將等值正數之所有位元（B15～B0 或 B31～B0）倒相（爲 1 之位元變 0，爲 0 之位元變 1，亦即所謂 1 的補數），然後再加上 1 即變成 2 的補數，茲以上例正數 12345，取其 2 的補數（即-12345）爲例說明如下：



5.3.6 浮點數之表示 (初學者請略過本節)

永宏 FBs浮點數格式同 IEEE-754 所制定之標準。儲存之格式共佔用 32 個位元(雙字元組) 其說明如下所示:

浮點數 = (正/負)數 + 指數 + 小數



▲ 若(正/負)位元之值爲 0，表示此浮點數爲正值，反之如果若(正/負)位元之值爲 1，代表此浮點數爲負值。

▲ 指數表示法爲超 127 法，舉例來說，若指數之值爲 128 就代表 1 次方，指數之值爲 129 就代表 2 次方 ····· 依此類推。若想表示指數爲負之值，則 126 就是-1 次方，125 爲 -2 次方 ····· 依此類推。

▲ 小數位數有 23bits，用來存放小數點以下之數字，正規化格式要求小數點之前一個位數，必須爲 1，且不必儲存(稱爲隱藏位元)。

● 整數轉換浮點數之規則爲如下:

$N = (-1)^{S} * 2^{(E -127)} * (1.M)$ $\quad 0 < E < 255$

轉換範例 1:

$1 = (-1)^{0} * 2^{(01111111)} * (1.000 \cdots\cdots 0)$

此範例中(正/負)數位元爲 0，指數部分爲 127(超 127 法)=01111111，隱藏位元爲 1，而小數位數則全部爲 0，因此經過轉換後的浮點數表示法如下所示:

Code(1) =

0	0	1	1	1	1	1	1	1	0	0	0	0	0……0	0	0
s	e	e	e	e	e	e	e	e	m	m	m	m	m m…… m	m	m

= 3F800000H

轉換範例 2 :

$0.5 = (-1)^{0} * 2^{(01111110)} * (1.000 \cdots\cdots 0)$

此範例中(正/負)數位元爲 0，指數部分以超 127 法表示爲 126=01111110(因指數部分爲-1，126-127=-1，故指數部份在超 127 法中要轉換成 126)，隱藏位元爲 1，而小數位數則全部爲 0，因此經過轉換後的浮點數表示法如下所示:

Code(0.5) =

0	0	1	1	1	1	1	1	0	0	0	0	0	0……0	0	0
s	e	e	e	e	e	e	e	e	m	m	m	m	m m…… m	m	m

= 3F000000H

轉換範例 3 :

$-500.125 = (-1)^{1} * 2^{(10000111)} * (1.11110100001000000000000)$

此範例中(正/負)數位元爲 1，指數部分以超 127 法表示爲 135=10000111 (因指數部分爲 8，135-127=8，故指數部份在超 127 法中要轉換成 135)，隱藏位元部分爲 1，而小數位數則爲 11110100001000000000000，因此經過轉換後的浮點數表示法如下所示:

Code(-500.125) =

1	1	0	0	0	0	1	1	1	1	1	1	1	0	1	0	0	0	0	1	0	0	0	00……0	0
s	e	e	e	e	e	e	e	e	m	m	m	m	m	m	m	m	m	m	m	m	m	m	mm……m	m

= C3FA1000H

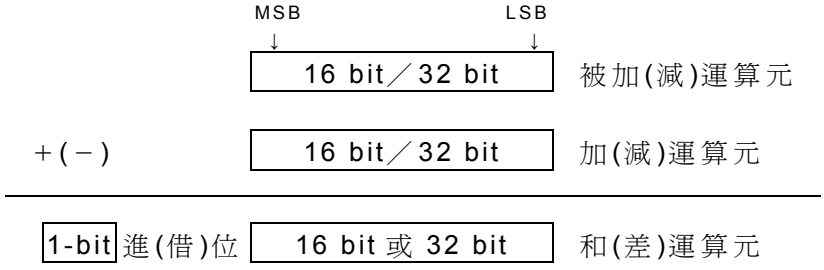
5.4 運算元遞增(減)之溢位與欠位 (初學者請略過本節)

16 位元及 32 位元運算元所能表示之數位極限正數最上限爲 32767 和 2147483647，而負數之最下限爲-32768 和-2147483648。當對一運算元作遞增或遞減動作（例如計數器上/下數或使暫存器值+1 或－1 時，若結果使其數值超過該運算元正數之最上限，則稱之爲溢位（Overflow：OVF），溢位之結果會使數值循環至負數之最下限（例如 16 位元上限 32767 再加 1，變成-32768）。若遞增(減)結果使其數值小於負數最下限，則稱之欠位（Underflow：UDF），而欠位之結果會使數值循環至正數之最上限（例如-32768 若再減 1 變成 32767），如下表所示。在 FBs-PLC 之遞增指令之功能輸出（FO）均有溢位或欠位之旗號輸出，可供您作串聯（Cascade）應用而獲得超過 16 或 32 位元之運算結果。

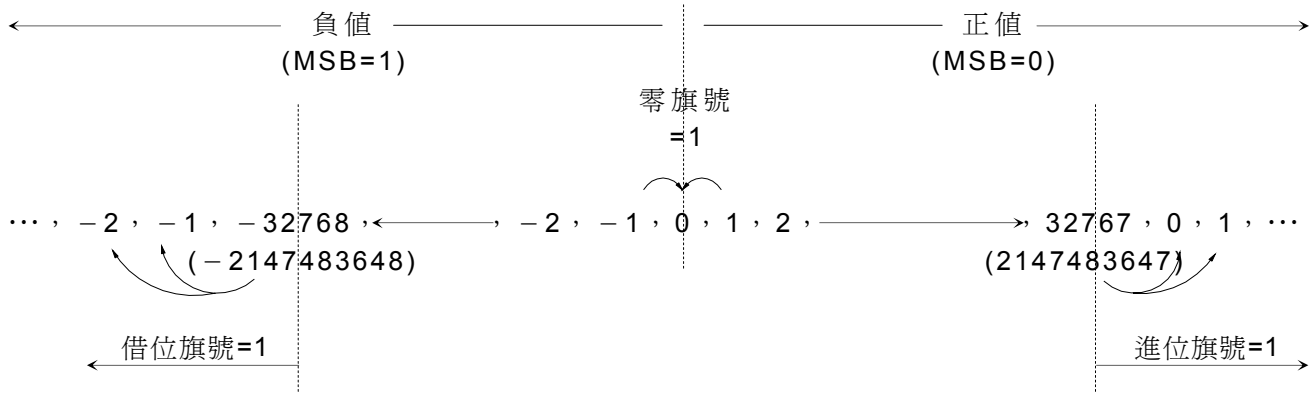
遞增(減)結果 / 溢/欠位	16 bit 運算元	32 bit 運算元
遞增	OVF=1：−32767 → −32768 → 32767 → 32766 → 32765	OVF=1：−2147483646 → −2147483647 → −2147483648 → 2147483647 → 2147483646
遞減	UDF=1：−32767 → −32768 → 32767 → 32766 → 32765	UDF=1：−2147483647 → −2147483648 → 2147483647 → 2147483646 → 2147483645

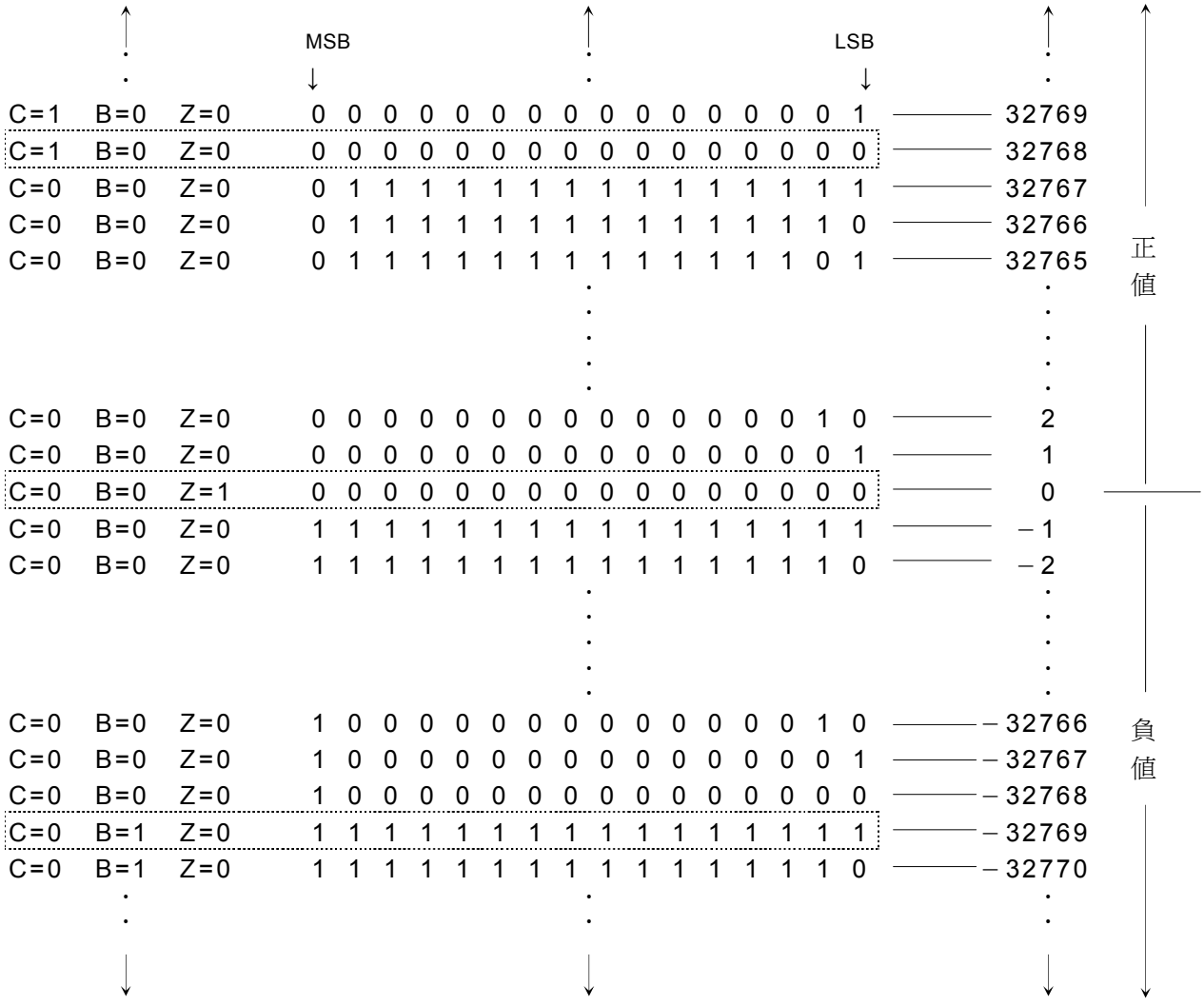
5.5 加/減運算之進位與借位

溢/欠位之發生係針對單一運算元之遞增/減致使該運算元之值超出其所能表示之正/負值極限時，產生溢/欠位旗號。而進/借位則不同於溢/欠位，首先其必有兩個運算元作加(減)運算，而得到和（差）結果與進/借位旗號。因被加（減）、加（減）及和（差）之位數(bit數)均一樣（16 bit 或 32bit），因此相加（減）之結果將可能造成和（差）之值超出 16 或 32 bit，因此需以進（借）位旗號配合和（差）運算元來表示真正之數值。而進位旗號發生在加（減）結果超出該和（差）運算元所能表示之正值最大極限（32767 或 2147483647）時，而借位則發生在加（減）結果超出該和（差）運算元所能表示之負值最大極限（−32768 或 −2147483648）時，因此加（減）運算後之真正結果爲進（借）位再加上和（差）運算元之值，FBs-PLC 之加減指令之功能輸出（FO）均有進位與借位旗號輸出，可供您獲得真正之結果。



因 FBs-PLC 之數值運算均採用 2 的補數，因此加（減）運算所得之和（差）值之負值之表示將不同於我們一般習慣之負值表示方式。運算結果爲負值時，其和（差）運算元將永遠不可能出現 0 值。進位旗號代表正值 32768(2147483648)，借位旗號則代表負值 －32768(－2147483648)。





C = Carry　　B = Borrow　　Z = Zero