



支援 Win-GRAF 軟體的 機械控制 PLCopen 功能塊 使用手冊

V1.1, 2024/10

技術: Martin Kruschat; 翻譯編輯: Lynn Tang;

泓格科技研發部, 泓格科技 CO., LTD.

最後編修者: Lynn Tang;

V1.1, 12/2024

免責聲明

泓格科技股份有限公司對於因為應用本產品所造成的損害並不負任何法律上的責任。本公司保留有任何時間未經通知即可變更與修改本文件內容之權利。本文所含資訊如有變更，恕不予另行通知。

本公司盡可能地提供正確與可靠的資訊，但不保證此資訊的使用或其他團體在違反專利或權利下使用。此處包涵的技術或編輯錯誤、遺漏，概不負其法律責任。

版權宣告

版權所有© 2024 泓格科技股份有限公司保留所有權利。

商標&著作權

本文件提到的所有公司商標、商標名稱及產品名稱分別屬於該商標或名稱的擁有者所有。

技術支援

請連絡當地的經銷商或 e-mail 問題至 service@icpdas.com。

更多關於產品的訊息可參考官方網站 <https://www.icpdas.com>。

附錄 A

改版歷程	日期	詳細內容	編修者
1.1	30.12.2024	新增中文版使用手冊	Lynn Tang/ Mike Lee

術語

在本手冊中，除非特別標明，全手冊將使用以下術語進行說明。

術語	內容
Aborting、Buffered、Blending	緩衝模式的各種模式
管理系统 FB	將軸或軸組作為參數，軸狀態或軸組狀態不根據執行而變化的運動控制 FB
軸組變數	包含軸組相關的參數及數據，AXES_GROUP 型變數的介面
軸變數	包含軸相關的參數及數據，AXIS_*型變數的介面
軸	進行運動控制的物件
全域標籤	在專案內建立了多個程式數據時，對所有的程式數據均有效的標籤。 可以將運動系統內的全域標籤作為管理 CPU 模組內的模組標籤（全域標籤）進行公開
命令目前位置 set position	通過環形計數器化整的命令位置位址。受目前位置更改的影響
命令目前速度 set velocity	通過運動運算創建的當前的控制值
反饋位置 Actual position	對從屬設備返回的位置位址通過環形計數器進行了化整的位置。受當前位置更改的影響。
反饋速度 Actual velocity	實軸中將來自於從屬設備的反饋值轉換為軸單位系統的速度
指定位置 Commanded position	運動系統 FB 中定義的命令位置位址
指定速度 Commanded velocity	運動系統 FB 中定義的命令速度
實軸	已連接到網路上的從設備的軸
實編碼器軸	從從設備上連接的編碼器的當前位置創建命令位置的軸。用於單軸同步控制的主軸。
輸出變數	FB 的輸出參數
解除連接	數據連結異常時，停止數據連結的處理
多軸插補控制	直線插補及圓弧插補等多個軸關聯的動作控制。指定通過軸組進行協調的軸。
硬重啟	重新接通系統的電源或復位
緩衝記憶體	用於儲存設定值、監視值等數據的智慧功能模組記憶體
緩衝模式	多重啟動的別稱
緩衝	多重啟動中，運動控制 FB 進入待機狀態
緩衝 FB	進行多重啟動，處於執行等待狀態（Busy 為 TRUE）的運動控制 FB

目錄

1	產品概述.....	10
1.1	介紹	10
1.2	PLCOPEN 函式庫.....	11
1.2.1	Workbench PLCopen 函式庫.....	11
1.2.2	Runtime PLCopen 函式庫.....	12
1.3	範例程式	12
2	運動功能塊 (MFB) 定義和介面	13
2.1	功能塊實體	13
2.2	多個功能塊實體控制同一軸	13
2.3	運動控制的功能塊類型	13
2.4	功能塊觸發	14
2.4.1	執行“邊緣觸發 (Edge-Triggered)”的功能塊.....	14
2.4.2	啟用的功能塊.....	16
2.4.3	“Enable”輸入的時間範例.....	17
2.5	緩衝模式	18
3	單軸狀態圖.....	21
3.1	STANDSTILL 靜止.....	22
3.2	ERRORSTOP 錯誤停止.....	22
3.3	HOMING 歸位.....	23
3.4	DISCRETE MOTION 離散運動.....	23
3.5	CONTINUOUS MOTION 連續運動	23
3.6	STOPPING 停止.....	23
4	單軸運動功能塊.....	24
4.1	MC_POWER	24
4.2	MC_HOME.....	25
4.3	MC_STOP.....	27
4.4	MCV_HALT	30
4.5	MC_MOVEABSOLUTE.....	32
4.6	MC_MOVERELATIVE.....	34
4.7	MC_MOVEVELOCITY	36
4.8	MC_SETPOSITION.....	38
4.9	MC_READPARAMETER	39
4.9.1	參數表 Parameter Table	39
4.10	MC_READBOOLPARAMETER.....	41
4.11	MC_WRITEPARAMETER	42
4.12	MC_WRITEBOOLPARAMETER	43

4.13	MC_READDIGITALINPUT.....	44
4.14	MC_READDIGITALOUTPUT	46
4.15	MC_WRITEDIGITALOUTPUT.....	47
4.16	MC_READACTUALPOSITION	48
4.17	MC_READACTUALVELOCITY	49
4.18	MC_READSTATUS	50
4.19	MC_READMOTIONSTATE	51
4.20	MC_READAXISINFO	52
4.21	MC_READAXISERROR.....	54
4.22	MC_RESET.....	55
5	協調運動.....	56
5.1	功能塊概述	56
5.2	狀態圖	56
5.2.1	GroupDisabled.....	57
5.2.2	GroupStandby.....	58
5.2.3	GroupMoving	58
5.2.4	GroupStopping	58
5.2.5	GroupErrorStop	59
5.3	單軸與群組狀態圖的關係	59
5.4	運動的混合和緩衝	61
5.4.1	混合.....	61
5.4.2	緩衝模式.....	63
6	群組運動功能塊.....	64
6.1	MC_ADDAXISTOGROUP	64
6.2	MC_REMOVEAXISFROMGROUP	67
6.3	MC_UNGROUPALLAXES	68
6.4	MC_GROUPENABLE	69
6.5	MC_GROUPDISABLE.....	71
6.6	MC_GROUPREADACTUALPOSITION	72
6.7	MC_GROUPREADACTUALVELOCITY	73
6.8	MC_GROUPSTOP	74
6.9	MCV_GROUPHALT.....	77
6.10	MC_GROUPINTERRUPT.....	79
6.11	MC_GROUPCONTINUE.....	80
6.12	MC_GROUPREADSTATUS	81
6.13	MC_GROUPREADERROR	83
6.14	MC_GROUPRESET	84
6.15	MC_MOVELINEARABSOLUTE.....	85
6.16	MC_MOVELINEARRELATIVE.....	88
6.17	MC_MOVECIRCULARABSOLUTE	90
6.18	MC_MOVECIRCULARRELATIVE	97
6.19	MCV_GROUPMOVEINCPATH	102
6.19.1	Create Incremental CSV File	104
6.19.2	Download CSV File.....	108
7	ECAT-M801/E-M901 相關的 FUNCTION BLOCKS	109
7.1	EM_INITMASTER	109
7.2	MCV_AXISASSIGN.....	111
7.3	EM_READSDO	114
7.4	EM_WRITESDO	115
7.5	EM_READPDO	116

7.6	EM_WRITEPDO	117
7.7	EM_READLOCALDIALL	118
7.8	EM_READSERIALNo	119
8	ECAT UTILITY.....	120
8.1	安裝	120
8.1.1	軟體安裝.....	120
8.1.2	硬體安裝.....	120
8.2	ETHERCAT 網路配置.....	121
8.2.1	手動網路配置.....	124
8.3	運動控制測試	128
8.3.1	軸配置.....	128
8.3.2	手動軸控制設定.....	131
8.3.3	單軸控制.....	132
9	附錄.....	134
9.1	縮寫	134
9.2	錯誤代碼	134

圖目錄

圖 1-1：PLC 功能塊支援介紹.....	10
圖 1-2：以 “_I” 結尾的 PLCopen 名稱保留給 ECAT-M801/EMP-9xx8-xx 系列使用	12
圖 2-1：Execute 的訊號負緣不會停止 FB 執行	14
圖 2-2：PLCopen 定義的執行/完成行為.....	16
圖 2-3：範例 1 - 使用 “Enable” 輸入進行錯誤處理	17
圖 2-4：範例 2 - 使用 “Enable” 輸入進行錯誤處理	18
圖 3-1：ECAT-M801/M901 EtherCAT 主卡的狀態圖	21
圖 4-1：“MC_Stop” 時序圖	28
圖 4-2：“MC_Stop” 與 “MC_MoveVelocity” 結合的行為.....	29
圖 4-3：“MCV_Halt” 與 “MC_MoveVelocity” 結合的行為.....	31
圖 4-4：“MC_INPUT_REF”結構成員變數	45
圖 5-1：群組狀態圖	57
錯誤! 書籤的自我參照不正確。2：具有不同 "TransitionPara" 設置的輪廓曲線.....	62
圖 6-1：新增/移除軸到群組.....	70
圖 6-2：MC_GroupStop 時序圖	75
圖 6-3：“MC_GroupStop” 與 “MC_MoveLinearRelative” 結合的行為	76
圖 6-4：“MC_GroupHalt” 與 “MC_MoveLinearRelative” 結合的行為.....	78
圖 6-5：使用單個命令的路徑移動（ MC_MoveLinear、MC_MoveCircular ）	96
圖 6-6：創建和填充用於增量運動執行的 CSV 檔	104
圖 8-1：專案配置	122
圖 8-2：主站正在掃描 EtherCAT 網路以尋找線上從站	123
圖 8-3：OP 模式下的 Utility 程式.....	124
圖 8-4：手動配置 EtherCAT 網路	125
圖 8-5：手動 PDO 分配	127
圖 8-6：“Motion Info” 頁籤	132

表目錄

表 1：ECAT-M801 系列支援軸數	11
表 2：EMP-9xx8-xx 系列支援軸數	11
表 3：單軸 PLCopen 功能塊	14
表 4：支援的 MC_BUFFER_MODE	19
表 5：不同 “BufferMode” 的速度配置檔	20
表 6：MC_Home 的介面	25
表 7：“ParameterNumber”（PN）定義	40
表 8：對功能塊進行分組	56
表 9：群運動命令對單軸狀態的影響	60
表 10：過渡模式概述	61
表 11：“mcBlendingNext” 的過渡參數設置	62
表 12：緩衝模式概述	63
表 13：緩衝模式	63
表 14：“MC_MoveCircularAbsolute” 的迴圈模式	94
表 15：“MC_MoveCircularAbsolute” 的迴圈模式	101
表 16：EtherCAT 循環時間定義	110
表 17：CiA402 PDO 映射模式	127
表 18：“用戶定義” 基本 PDO 條目	128

1 產品概述

1.1 介紹

本手冊中描述的 PLCopen 功能塊只能與 EtherCAT 主站 PCIe 卡 ECAT-M801 和可編程自動化控制器 EMP-9xx8-xx 一起使用。EMP-9xx8-xx 內置了一張主站卡，該主站卡運行於專用處理器上。主運動控制引擎和路徑生成器被實現於主站卡中，這確保了實時性和確定性行為。標準 Windows 操作系統的 Win-GRAF 運行時環境並非實時，因此運行時環境需要依賴主站卡的實時特性來進行運動控制。運行時環境的 PLCopen 功能會直接調用 ECAT-M801/EMP-9xx8-xx 主站卡的運動功能，以控制每個軸的執行路徑。

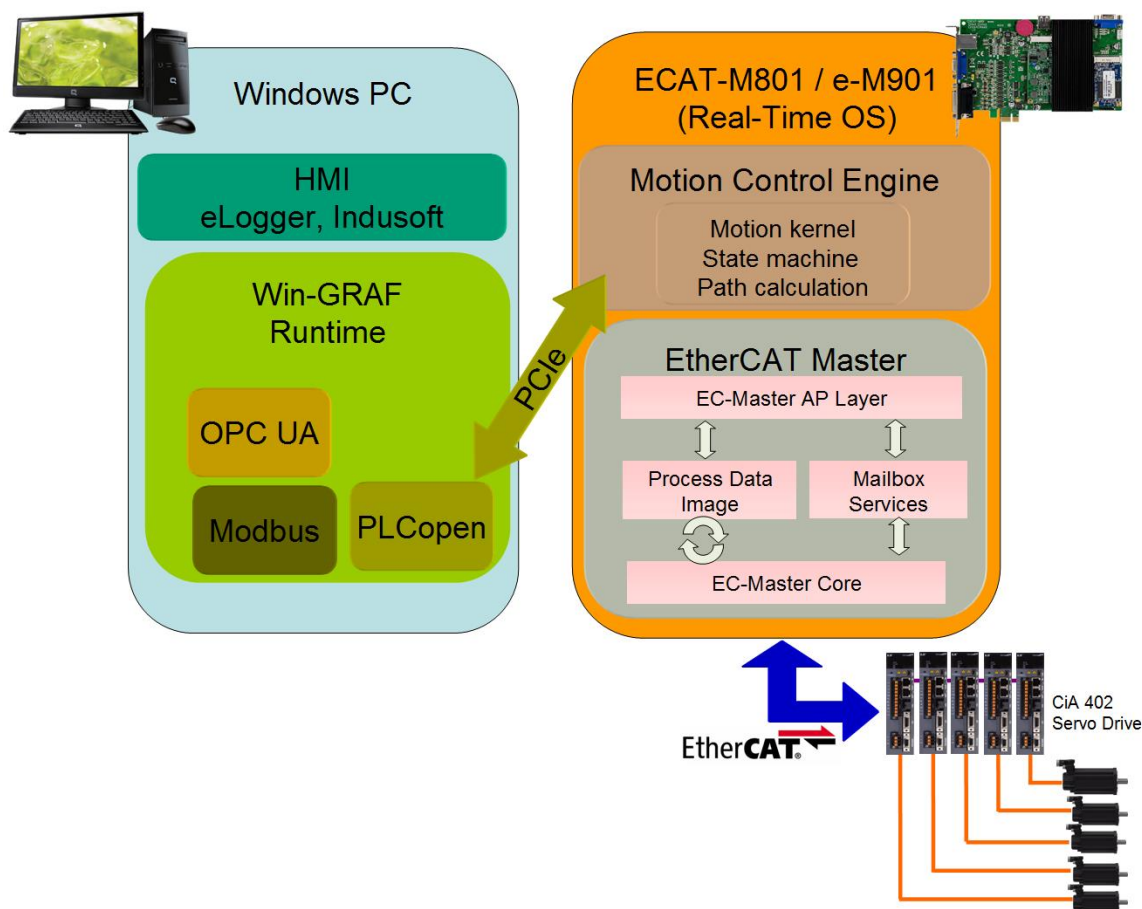


圖 1-1：PLC 功能塊支援介紹

ECAT-M801/EMP-9xx8-xx 主站卡系列支持不同數量的軸（見表 1/表 2）。這些軸可以是真實的（伺服或步進驅動）或虛擬的。Win-GRAF 運行時環境支持的軸數量受主站卡的限制。要獲取 ECAT-M801/EMP-9xx8-xx 系列的最新產品版本及支持的軸數，請訪問泓格科技網站，並搜索“ECAT-M801”或“EMP-9058”。

PCIe Card	支援軸數
ECAT-M801-8AX	8
ECAT-M801-16AX	16
ECAT-M801-32AX	32

表 1: ECAT-M801 系列支援軸數

EMP-9xx8-xx	支援軸數
EMP-9xx8-16	16
EMP-9xx8-32	32

表 2: EMP-9xx8-xx 系列支援軸數

1.2 PLCopen 函式庫

1.2.1 Workbench PLCopen 函式庫

PLCopen 函式庫隨著 Workbench 一起安裝。函式庫名稱為 'EtherCAT_PCI_Vxx'，後綴 'Vxx' 表示版本號。Workbench 函式庫可以在以下目錄中找到：

C:\Users\Public\Documents\Win-GRAF Workbench\Win-GRAF Wb 10.0\HwDef\ EtherCAT_PCI_V1a

未來泓格科技將延伸 PLCopen 運動控制系列產品。由於硬體限制，某些 PLCopen 功能塊的實現可能會因平台而異，例如：功能塊支援的輸入和輸出參數的數量可能會有所不同。因此，新增了功能塊後綴，以便使用者能夠清楚地識別一組 PLCopen 功能塊所支援的目標平台。對於 ECAT-M801/EMP-9xx8-xx 系列，每個 PLCopen 定義命令的末端都會加上“_I”（圖 1-2）。

注意:

以“_I”結尾的 PLCopen 名稱僅用於 ECAT-M801/EMP-9xx8-xx 系列主機。



MC_HOME_I
MC_MOVEABSOLUTE_I
MC_MOVECIRCABS_I
MC_MOVECIRCREL_I
MC_MOVELINEARABSOLUTE_I
MC_MOVELINEARRELATIVE_I
MC_MOVERELATIVE_I
MC_MOVEVELOCITY_I
MC_POWER_I

圖 1-2：以 “_I” 結尾的 PLCopen 名稱保留給 ECAT-M801/EMP-9xx8-xx 系列使用

1.2.2 Runtime PLCopen 函式庫

Runtime 運行時需要以下 PLCopen 函式庫：

- ECAT8K.dll
- Fb_EcatPlcopen.dll
- libecatdevice.dll

這些函式庫檔案應與 runtime 執行檔放於同一目錄中。Runtime 安裝程式時會預設將這些函式庫安裝到 runtime 執行檔目錄內。

1.3 範例程式

提供多種 PLC 程式語言的範例程式，示範如何初始化 EtherCAT 主站、設定軸配置以及呼叫 PLCopen 功能塊。

範例程式被放在底下的檔案夾路徑：

- C:\Users\Public\Documents\Win-GRAF Workbench\Win-GRAF Wb xx.x\ Projects\EMP-9000\PLCopen
- C:\Users\Public\Documents\Win-GRAF Workbench\Win-GRAF Wb xx.x\ Projects\Windows PC\PLCopen

範例程式在安裝 Win-GRAF workbench 時已同時被安裝。

2 運動功能塊 (MFB) 定義和介面

2.1 功能塊實體

應用程式中使用的每個運動功能塊 (MFB) 都有自己的實體記憶體。這些記憶體能儲存上一個週期的資料或在觸發執行後儲存該 MFB 輸入值。程式設計師是看不見內部記憶體資料的。實體記憶體的內部值從功能塊的一次執行到下一次執行都保持不變，以便保留功能塊執行的狀態。

2.2 多個功能塊實體控制同一軸

已啟用運動功能塊 (MFB) 的多個實體記憶體可以在單一應用程式中呼叫相同軸。如果兩個同類型的功能塊嘗試對同一軸進行操作，則它們將按照在應用程式邏輯執行期間遇到的順序進行處理。當遇到新的實體記憶體呼叫時，目前啟動中的功能塊實體將會終止，新的實體呼叫將會接管控制權。

2.3 運動控制的功能塊類型

運動功能和功能區塊分為兩種操作類型：管理和運動。

- 管理功能和功能塊不會造成軸運動，而運動功能與功能塊可控制軸的運動。

Administrative	Single Axis Motion
MC_Power	MC_Home
MC_ReadStatus	MC_Stop
MC_ReadAxisError	MCV_Halt
MC_ReadParameter	MC_MoveAbsolute
MC_ReadBoolParameter	MC_MoveRelative
MC_WriteParameter	MC_MoveVelocity
MC_WriteBoolParameter	
MC_ReadDigitalInput	
MC_WriteDigitalOutput	
MC_ReadActualPosition	
MC_ReadActualVelocity	
MC_ReadAxisInfo	
MC_ReadMotionState	
MC_SetPosition	
MC_Reset	

表 3：單軸 PLCopen 功能塊

2.4 功能塊觸發

PLCopen 與泓格科技定義的功能塊可透過 “**Enable**” 輸入（條件觸發）或 “**Execute**” 輸入（邊緣觸發）來啟用。

- 具有 “**Enable**” 輸入的功能塊，當 *Enable* 為真時便會套用輸入參數。
- 使用 “**Execute**” 輸入的功能塊，參數會在 “**Execute**” 輸入的訊號正緣被套用。

2.4.1 執行 “邊緣觸發 (Edge-Triggered)” 的功能塊

運動生成功能塊（如 MC_Stop 和 MC_Home）是邊緣觸發的，僅在 “**Execute**” 輸入的訊號正緣運作。僅當 “**Execute**” 輸入從 FALSE 更改為 TRUE 時，輸入參數才會被應用。要在 MFB 完成後再次重新啟動 MFB 執行，必須先將 “**Execute**” 輸入設定為 FALSE 並經過一個 PLC 週期，然後再使用 “**Execute**” 輸入的 TRUE 值呼叫 MFB。

輸入參數會在 “**Execute**” 輸入的訊號正緣時儲存在實例記憶體中。要修改一個或多個輸入參數，需要先將 “**Execute**” 設定為 FALSE 並經過一個週期，然後更改輸入參數，最後再次觸發功能塊。重要的是，Execute 輸入至少需要保持為 FALSE 一個週期。

“**Execute**” 輸入的訊號負緣不會停止或影響 MFB 的執行（圖 2-1）。MFB 啟動的操作將繼續執行，直到它完成或直到另一個 MFB 中斷或停止（例如 MC_Stop）操作。

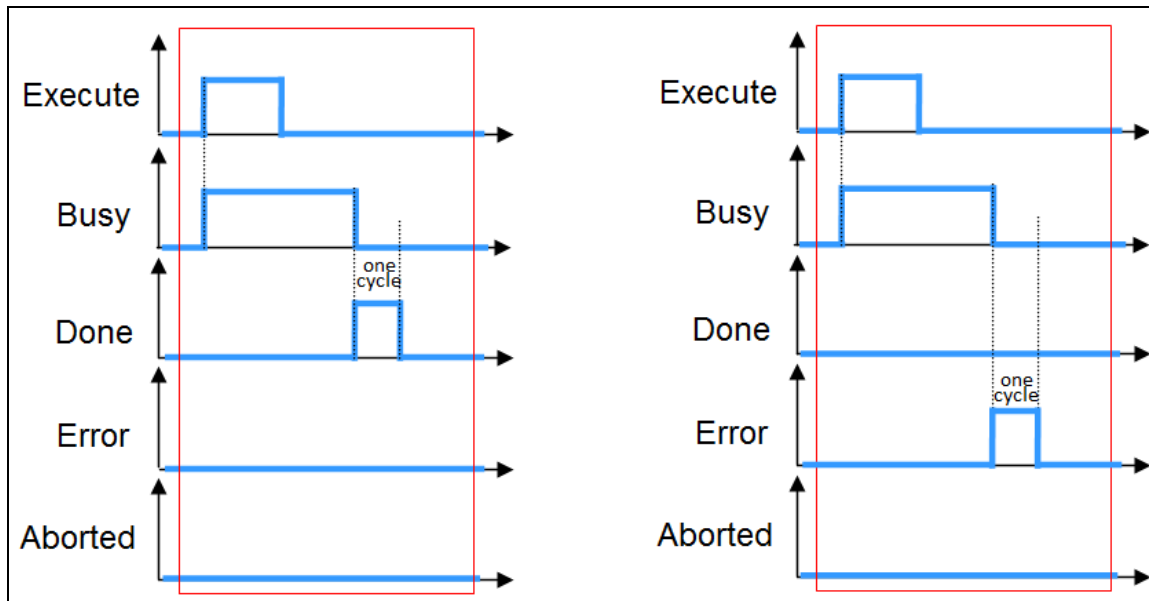


圖 2-1：Execute 的訊號負緣不會停止 FB 執行

2.4.1.1 已執行 (邊緣觸發) 功能塊的輸出

輸出 “Busy”、“Done”、“Error” 和 “CommandAborted” 是互斥的，這意味著其中只有一個可以是 TRUE。如果輸入 “Execute” 為 TRUE，則其中一個輸出必須為 TRUE。

“Busy” 輸出表示 MFB 的執行未完成，並且預計會生成新的輸出狀態（值）。一旦 MFB 執行完成，並且其他輸出之一（Done、CommandAborted、Error）從 FALSE 更改為 TRUE，“Busy” 輸出將重置為 FALSE。

“Execute” 的訊號負緣將重置以下輸出：

- Done
- Error
- ErrorID, 如果與錯誤相關聯
- CommandAborted

即使在功能模塊完成其動作之前，“Execute” 從 TRUE 變為 FALSE，輸出參數的值也會至少設置為 TRUE 一個週期。

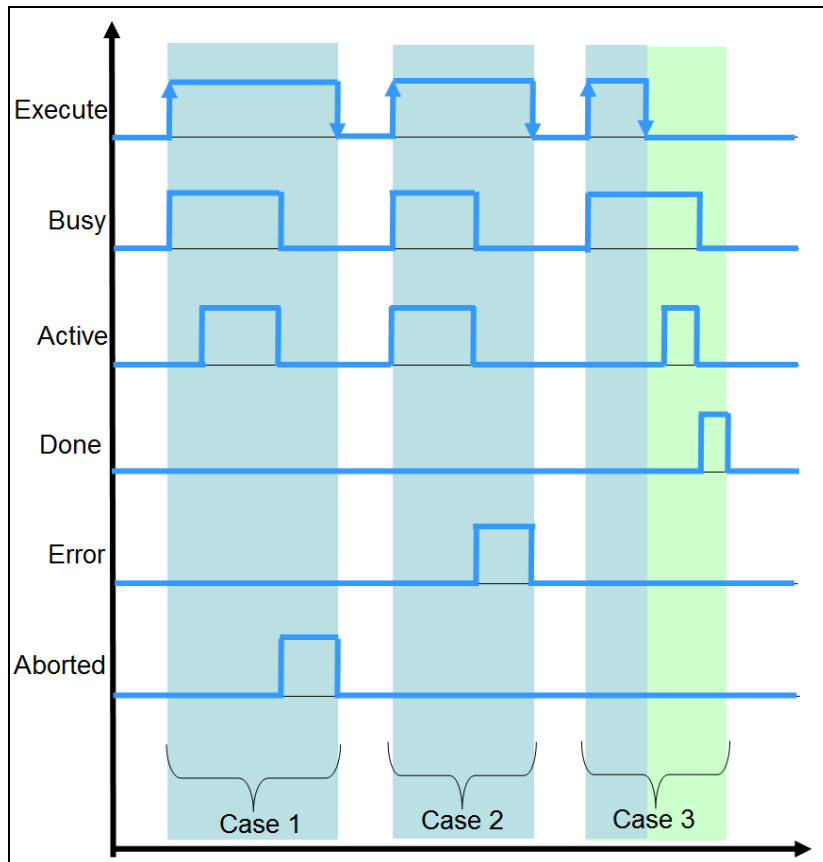


圖 2-2：PLCopen 定義的執行/完成行為

2.4.1.2 重新執行功能塊

如果正在執行的 MFB 的實例在完成其當前命令（“Busy”=TRUE）之前收到新的執行，則 MFB 將忽略執行觸發並繼續運行其當前命令，直到它完成或被中止（例如，由“MC_Stop” MFB 執行）。如果更改輸入值並在 MFB 執行時再次觸發“Execute”（“Busy”=TRUE），則“Execute”和所有其他輸入將被忽略。

2.4.2 啟用的功能塊

啟用的 MFB 只有在輸入參數“Enable”為 TRUE 時才會處於活動狀態。“Enable”參數用於執行迴圈操作，例如讀取每個週期中的當前編碼器位置（“MC_ReadActualPosition”）。與“Execute”相反，“Enable”輸入會使動作持續執行，只要啟用為 TRUE。

- “Enable”輸入：
 - “Enable”是級別敏感的，這意味著一旦“Enable”設定為 TRUE，MFB 將在每個任務週期中執行，直到設定為 FALSE。“Valid 有效”輸出表示 FB 輸出是否有效。
 - 輸入參數與“Enable”輸入的訊號正緣一起使用，並且可以連續修改。
 - 輸出“Valid”和“Error”是互斥的：一次只能有一個為 TRUE。
 - 當“Enable”輸入的訊號負緣到來時，“Valid”、“Busy”、“Error”和“ErrorID”輸出將儘快重置（為 FALSE）。
- 只要有效輸出值可用且“Enable”輸入為 TRUE，則“Valid”輸出為 TRUE。只要輸入“Enable”為 TRUE，相關輸出值就會在每個週期時間內更新。
- 具有輸出“Busy”參數的 MFB 表示它需要多個週期時間才能完成執行。“Busy”輸出表示 MFB 的執行尚未完成，預計在下一個週期中將生成新的輸出。“Busy”設定在“Enable”的訊號正緣，只要 MFB 正在執行任何操作，它就會保持原設定狀態。
- 如果發生 MFB 錯誤，則輸出無效（“Valid”設定為 FALSE）。當錯誤條件消失時，這些值將重新出現，並且將再次設定“Valid”輸出（“Valid”=TRUE）。

2.4.3 “Enable”輸入的時間範例

範例 1 (圖 2-1) :

- 案例 1：這個圖顯示為正常操作情況。一旦輸出參數 (例如 “MC_Power” 中的狀態) 有效， “Valid” 輸出將變為 TRUE。
- 案例 2：MFB 操作期間發生錯誤，而 “Enable” 為 TRUE。在錯誤發生的時間段內， “Error” 輸出設定為 TRUE， “Valid” 輸出設定為 FALSE。輸出 “Busy” 保持高電平。重置錯誤後，將恢復正常操作過程。清除錯誤後， “Valid” 輸出可能需要一些時間才能顯示 TRUE。

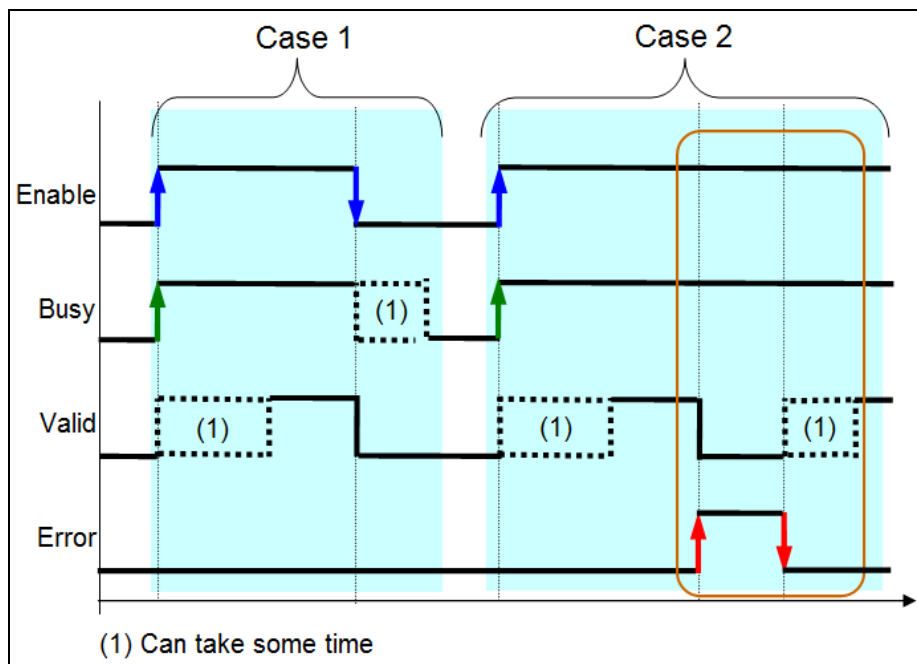


圖 2-3：範例 1 - 使用 “Enable” 輸入進行錯誤處理

範例 2 (圖 2-4) :

- 案例 2：此範例顯示無法自動清除的錯誤。如果輸出 “Busy” 和 “Valid” 為 FALSE，而 “Enable” 為 TRUE，則必須將 “Enable” 設定為 FALSE，然後再次設定為 TRUE，然後才能繼續運行功能塊。

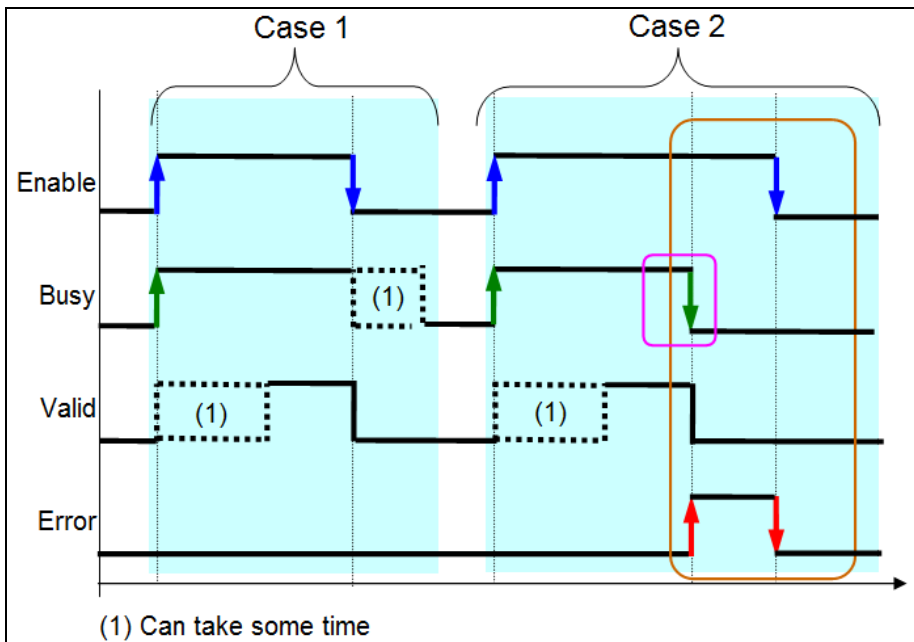


圖 2-4：範例 2 - 使用 “Enable” 輸入進行錯誤處理

2.5 緩衝模式

“BufferMode” 輸入用於控制多個功能塊的命令流程，並指定傳入命令是中斷另一個命令（非排隊模式、非緩衝模式），還是僅在上一個命令（排隊模式）之後執行以下命令。這些模式決定了 MFB 的運動動作何時啟動。

非排隊模式（非緩衝模式、中止模式）：

在非排隊模式下，觸發的 MFB 命令會導致正在執行的 MFB 終止。在這種情況下，終止的 MFB 設定 “CommandAborted” 輸出。“mcAborting” 命令是非緩衝命令，即使這會中斷另一個動作並清除命令緩衝區，也會立即執行。命令緩衝區存儲所有先前尚未執行的命令。

排隊模式（緩衝模式）：

緩衝模式下的命令會等到當前 MFB 執行結束，並將其 “Done” 輸出設定為 TRUE。換句話說，後續的 MFB 將等待，直到佇列中所有的 MFB 都完成。如果無限運動命令（“MC_MoveVelocity”）處於活動狀態，則無法使用緩衝命令。在這種情況下，必須先終止無限運動命令。ECAT-M801 的限制：在不停止的情況下，無法在離散運動和連續運動之間切換。

當 MFB 繁忙（“Busy” = TRUE）時，它無法重新觸發，並且其輸入參數的更改不會生效。因此，不可能在 MFB 忙碌時觸發相同的實例來向佇列添加命令，也不支持使用不同的參

數觸發相同的功能塊。對於每個新的佇列條目，需要另一個 MFB 的實例。

默認情況下，在執行另一個命令時，最多可以在緩衝區中排隊 100 個命令。佇列中可以存儲的最大命令數可以通過 FB “MC_WRITEPARAMETER” 設定，但無法設置超過 100 條命令。如果後續 MFB 遇到已滿緩衝區，則佇列將被拒絕並顯示錯誤（Error -1138：佇列已滿）。

如果最後一個命令在非排隊模式（“mcAborting”）下啟動，則該命令將變為活動狀態並中斷正在運行的命令並清除所有排隊的命令。

每個支援緩衝功能的 MFB 都有一個 “Active” 輸出，當功能塊控制軸時設定該輸出。觸發 “Execute” 輸入後，無法更改 MFB 的輸入參數。

支援的 “BufferModes”：

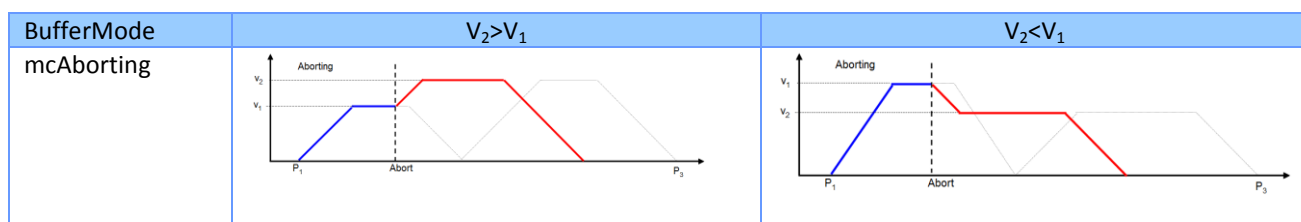
- Aborting（中斷模式）：無命令緩衝。命令會立即執行，並中斷任何可能正在運行的命令。
- Buffered（緩衝模式）：命令會在軸上無其他命令運行時執行。之前的運動會繼續，直到完全停止為止。後續命令從靜止狀態開始執行。
- BlendingNext（混合模式）：命令會在軸上無其他命令運行時執行。與緩衝模式對比，軸不會在前一目標位置停下，而是以上一條命令的速度通過該位置。

No	MC_BUFFER_MODE	描述
0	mcAborting	立即啟動FB（預設模式）
1	mcBuffered	目前運動完成後啟動 FB
4	mcBlendingNext	該速度與第二個 FB 的速度混合

表 4：支援的 MC_BUFFER_MODE

例子：

第一個 MFB 將軸從位置 P1 移動到 P2，第二個 MFB 則從 P2 移動到 P3。當軸以恆定速度向 P2 移動時，觸發第二個 MFB。不同速度曲線的參考點始終為 P2。“BufferMode” 用於指定 P2 處的速度為 v1 或 v2。



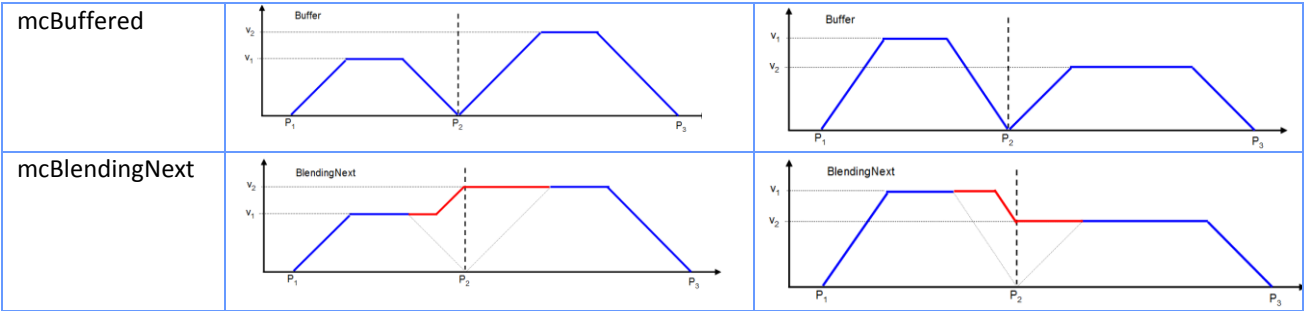


表 5：不同 “BufferMode” 的速度配置檔

3 單軸狀態圖

狀態圖定義了單一軸的行為。如圖 3-1，軸始終處於下列定義狀態之一。在下圖中，上面列出的運動命令將軸轉換為相對應的運動狀態。如果軸已經處於相對應的運動狀態，該運動命令仍然可以執行。帶有實線的箭頭表示發出運動命令時的狀態轉換；帶有虛線的箭頭表示軸的命令結束時發生的狀態轉換。

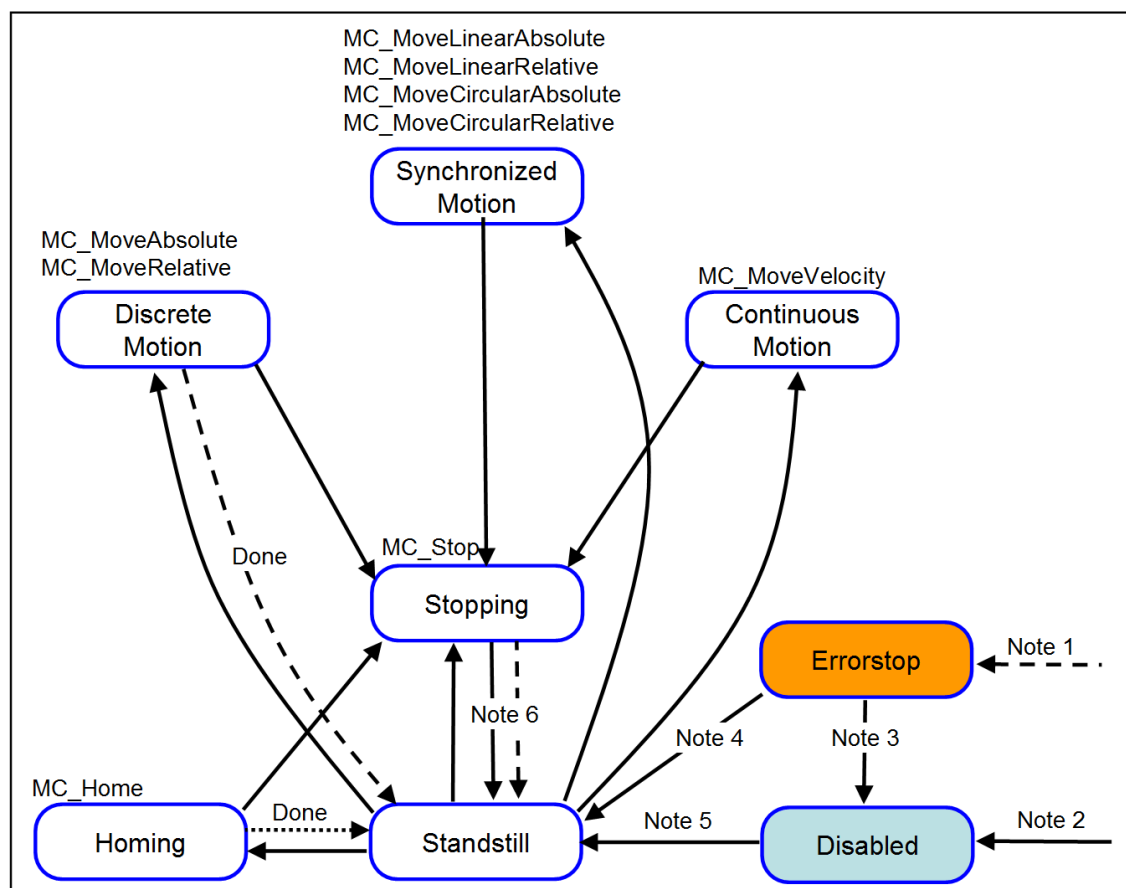


圖 3-1：ECAT-M801/M901 EtherCAT 主卡的狀態圖

備註:

- 附註 1: 軸可以從除了“Disabled”狀態之外的任何其他狀態進入“Errorstop”狀態。
- 附註 2: 通過停用“MC_Power”，軸可以直接進入“Disabled”狀態：
 - i. 成功初始化 EtherCAT 主站（“EM_InitMaster”）和軸分配（“EM_AxisAssign”）時，當“MC_Power.Enable” = FALSE 且“MC_Power.Status” = FALSE。

ii. 如果 “MC_Power.Enable” 設定為 FALSE，則從 “ErrorStop” 以外的任何狀態。

- 附註 3: 如果軸處於 “Errorstop” 狀態，則需要先透過執行 “MC_Reset” 切換到 “Standstill” 狀態，然後再停用電源（“MC_Power.Enable” = TRUE）。
- 附註 4: 執行 “MC_Reset” 將狀態從 “Errorstop” 更改為 “Standstill”。
- 附註 5: 要從 “Disabled” 狀態切換到 “Standstill” 狀態，必須將 “MC_Power.Enable” 設定為 TRUE。當 “MC_Power.Status” 變為 TRUE，就會進入 “Standstill” 狀態。
- 附註 6: 當 “MC_Stop.Done” = TRUE 和 “MC_Stop.Execute” = FALSE，軸就會從三種運動狀態（離散運動、連續運動、歸位）中的任何一種變為 “Standstill” 狀態。

3.1 Standstill 靜止

在 “Standstill” 狀態下，軸處於靜止狀態並準備執行運動。軸 “MC_Power” 已啟用，其 “Status” 輸出設定為 TRUE。可以從此狀態執行以下任何運動命令：

- MC_Home, MC_MoveAbsolute, MC_MoveRelative, MC_MoveVelocity
- 軸每次到達其命令目標位置後都會進入 “Standstill” 狀態。

3.2 Errorstop 錯誤停止

如果發生軸錯誤，軸將進入 “Errorstop” 狀態。可以通過呼叫 “MC_ReadAxisError” MFB 來確定錯誤類型。

可能的軸錯誤包括：

- 軸觸發硬體限位開關
- 軸達到軟體限制
- 軸驅動錯誤（例如扭矩過大）
- EtherCAT 主站遇到通訊錯誤等。

在任何狀態下都可能發生錯誤。一旦軸處於 “Errorstop” 狀態，軸運動就會停止，並且無法發出新的運動命令。退出 “Errorstop” 狀態的唯一方法是糾正錯誤的原因並發送 “MC_Reset” 命令，將軸轉換為 “Standstill”。

例如，軸被一個限位開關停止：

1. 首先使用 “MC_Reset” 重置軸錯誤並切換到待機狀態。

2. 然後將軸從限位開關向相反的方向移動（例如，通過呼叫 `MC_MoveRelative`）。

3.3 Homing 歸位

此狀態表示軸正在執行歸位搜尋。進入此狀態的唯一方法是在“*Standstill*”時使用“*MC_Home*”功能塊。只有“*MC_Stop*”命令可以中斷“*Homing*”狀態。

3.4 Discrete Motion 離散運動

在此狀態下，軸正在執行點對點運動。當軸到達其命令的目標位置時，運動完成。一旦到達該位置並且緩衝區中沒有待執行的命令，軸就會返回到“*Standstill*”狀態。支援以下離散

MFB：

- `MC_MoveAbsolute`,
- `MC_MoveRelative`

3.5 Continous Motion 連續運動

此狀態表示移動沒有明確的終點。軸繼續移動，直到“*MC_Stop*”或“*MCV_Halt*”FB使軸停止或發生軸錯誤（例如，觸發限位開關）。目前僅支援“*MC_MoveVelocity*”。

3.6 Stopping 停止

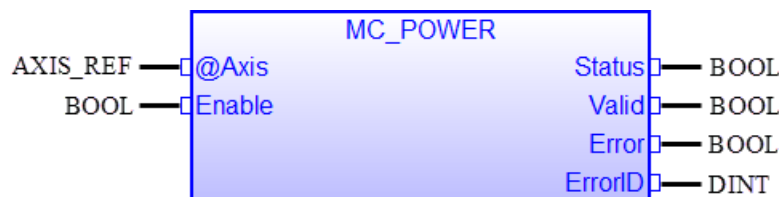
“*MC_Stop*”停止正在運行的離散或連續運動命令。儲存在命令緩衝區中的命令將被中止和刪除。一旦設定“*MC_Stop.Execute*” = TRUE 後，軸將立即進入“*Stopping*”狀態。軸將保持此狀態，直到“*MC_Stop.Execute*”設定為 FALSE。當軸處於此狀態時，可以呼叫其他 MFB，但不會執行它們。當軸停止時，“*MC_Stop.Done*”將輸出 TRUE。只有在“*MC_Stop.Done*”為 TRUE 且“*MC_Stop.Execute*”設定為 FALSE 後，狀態才會轉換為“*Standstill*”。

4 單軸運動功能塊

本章介紹用於單軸運動控制的 Win-GRAF PLCopen 函式庫。

4.1 MC_Power

要執行運動，必須通過“MC_Power”功能塊啟用軸。只要“MC_Power”處於活動狀態（“Status” = TRUE），可以執行運動命令。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Enable	BOOL	只要“Enable”為 TRUE，伺服就會被啟用。
VAR_OUTPUT	Status	BOOL	指示伺服驅動器是已設定為開啟（ON）還是關閉（OFF）。
	Valid	BOOL	如果為 TRUE，則狀態輸出有效。
	Error	BOOL	功能模塊內部發生錯誤。
	ErrorID	DINT	錯誤識別

備註:

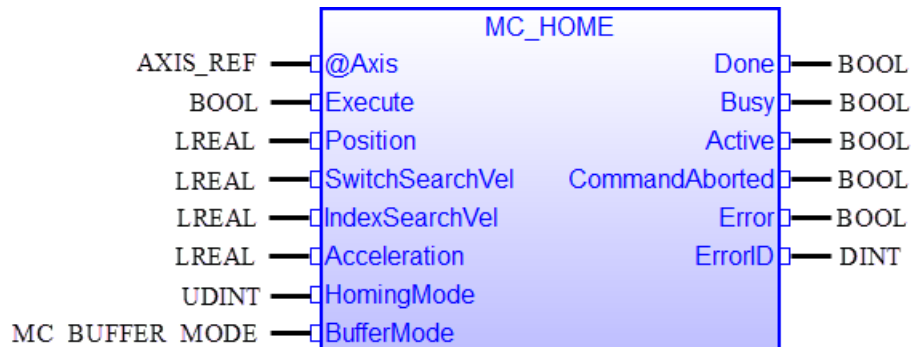
- “Enable”輸入後使能啟用驅動器中的功率等級而非 FB 本身的功率等級
- 若是“MC_Power”與“Enable”一起使用並設定為 TRUE，當狀態在“Disabled”時，軸狀態會變為“Standstill”。
- 當電力中斷(或運轉時中斷)，軸狀態會轉入“ErrorStop”狀態。舉個電力中斷的例子說明：伺服驅動器在運動執行過程中關閉或伺服驅動器因扭矩過大而進入錯誤狀態。

透過將“MC_Power.Enable”設定為 TRUE，軸才能夠從“ErrorStop”以外的任何狀態切換電源。如果軸處於“ErrorStop”狀態，則必須使用“MC_Reset”將狀態更改為“Dsiabled”之前清除錯誤。

- 每個軸只能發出一個“MC_Power” FB。如果多個“MC_Power”實例控制同一軸，則會造成錯誤。

4.2 MC_Home

此 MFB 命令軸執行'search home'序列。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	在訊號正緣開始動作
	Position	LREAL	編碼器在歸零搜尋過程完成後，將被設定為此數值絕對值 (Unit: user unit)
	SwitchSearchVel	LREAL	搜尋歸位零點過程中的速度(Unit: user unit)
	IndexSearchVel	LREAL	搜尋 Z 相訊號時的速度 (Unit: user unit)
	Acceleration	LREAL	歸零加速度 (Unit: user unit/s ²)
	HomingMode	UDINT	根據 CiA402 定義的歸零模式。歸零模式由製造商設定
	BufferMode	MC_BUFFER_MODE	保留（不支援）
VAR_OUTPUT	Done	BOOL	已達到零速度。狀態機已轉換為靜止狀態
	Busy	BOOL	功能模塊正在執行中
	Active	BOOL	命令正在執行中（與 '忙碌' 相同）
	CommandAborted	BOOL	命令被另一個命令中止
	Error	BOOL	功能模塊內部發生錯誤
	ErrorID	DINT	錯誤識別

表 6：MC_Home 的介面

備註:

“MC_Home” 只能在軸處於 “Standstill” 狀態時執行，否則 FB 將輸出錯誤 ID，表示軸狀態無效

- “MC_Home” 不支援中止 (aborting) 和緩衝 (buffer) 模式。當 “MC_Home” 命令正在執行時，它不能被任何其他運動命令 (例如，另一個 “MC_Home” 或任何其他 “MC_Move” 功能塊) 中止
- 歸位 (homing) 模式：自動化 CAN (CiA) 402 定義了多達 37 種歸位模式。請參閱支持歸位模式的 EtherCAT 伺服驅動器製造商的使用手冊。

4.3 MC_Stop

該 MFB 命令受控運動停止並將軸傳輸到 “Stopping” 狀態，它會中止任何正在進行的 MFB 執行。當軸處於 “Stoppin” 狀態時，其他 MFB 無法在同一軸上執行任何運動。當軸達到零速度後，“Done” 輸出設定為 TRUE。只要 “Execute” 為 TRUE，軸就會保持 “Stopping” 狀態。一旦設定了 “Done” 且 “Execute” 為 FALSE，軸進入 “Standstill” 狀態。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	在訊號正緣開始動作
	HardStop	BOOL	<ul style="list-style-type: none"> FALSE: 使用加速時間作為減速時間 TRUE: 停止，不進行減速停止
VAR_OUTPUT	Done	BOOL	已達到零速度。狀態機已轉換為靜止狀態
	Busy	BOOL	功能模塊正在執行中
	CommandAborted	BOOL	命令因切斷伺服電源而被中止
	Error	BOOL	功能模塊內部發生錯誤
	ErrorID	DINT	錯誤識別

備註:

- 該 MFB 主要用於緊急停止目的或運動異常情況
 - 只要 “Execute” 為 TRUE，軸就會保持 “Stopping” 狀態，並且不會執行任何其他運動命令。
 - “MC_Stop.HardStop” = FALSE 的減速時間由當前正在運行的運動 MFB 決定，例如 (“MC_MoveAbsolut”、 “MC_MoveRelative”)。緩衝區中第一個 MFB 的 “AccDecTime” 輸入設定所有後續運動命令的加速和減速時間。
 - 下面顯示了帶有 “MC_Stop” 狀態轉換的時序圖。
 - 在情況 1 中，在重置 “Execute” (FALSE) 之前設置 “Done” 輸出 (TRUE)。
 - 在情況 2 中，在 “Done” 變為 true 之前，“Execute” 被重置。
- 軸仍處於 “Stopping” 狀態

- 當 “Execute” 設定為 (TRUE)
- 直到將 “Execute” 設定為 FALSE 之後， “Done” 變為 TRUE。

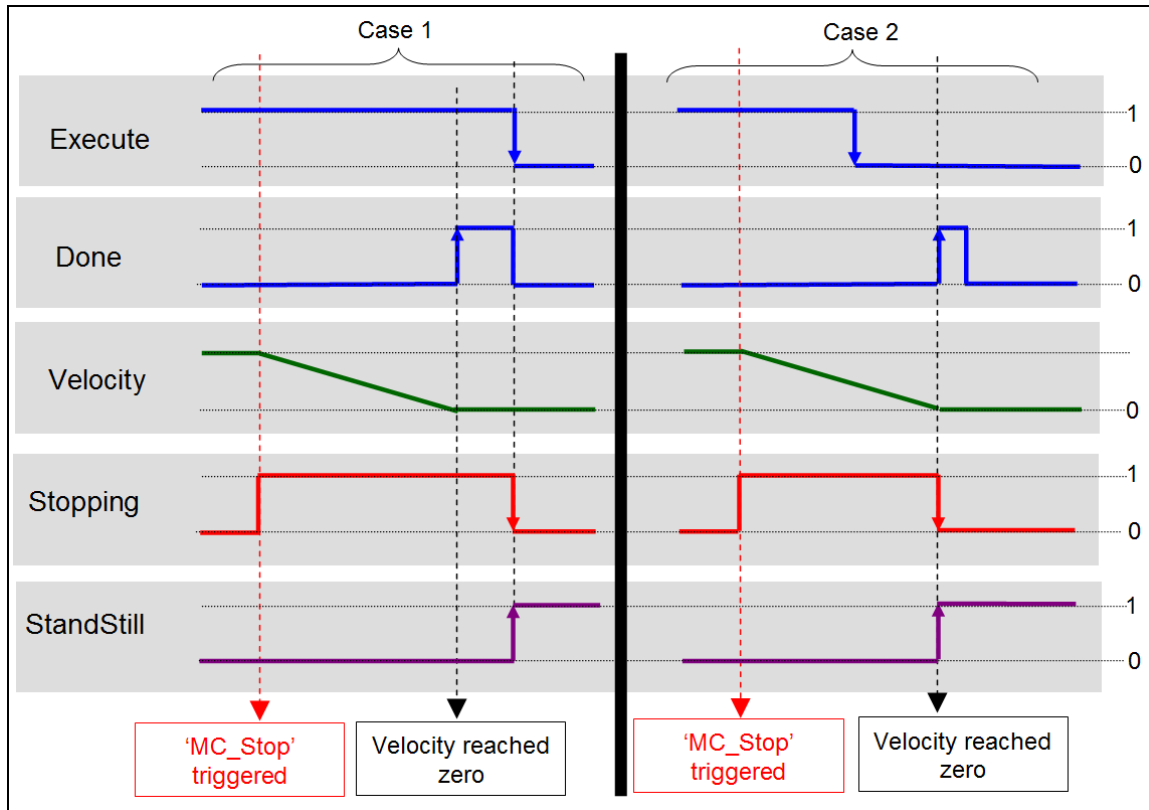


圖 4-1：“MC_Stop” 時序圖

- 下面的範例 (圖 4-2) 顯示了與 “MC_MoveVelocity” 結合使用的行為。
 - a) 軸運動以 “MC_Stop” 進行減速。
 - b) 只要 “MC_Stop.Execute” 為 TRUE 時，軸拒絕運動命令。“MC_MoveVelocity” 會報告一個錯誤，指示已啟用的 “MC_Stop” 命令。返回的錯誤是功能塊錯誤，而不是軸錯誤，這意味著狀態保持在 “Stopping” 狀態，不會更改為 “ErrorStop”。在第三個 “Exe_1” 訊號正緣，軸開始下一個運動。

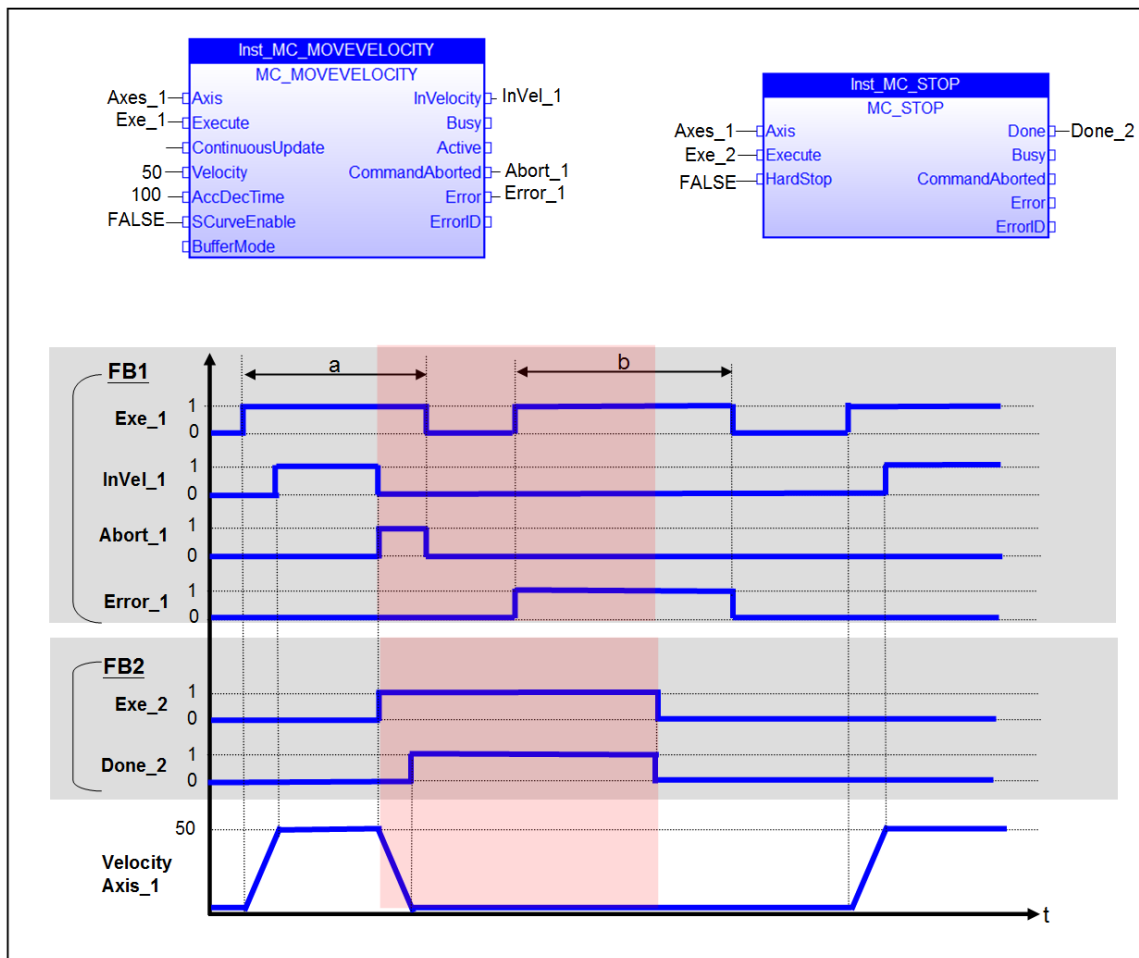
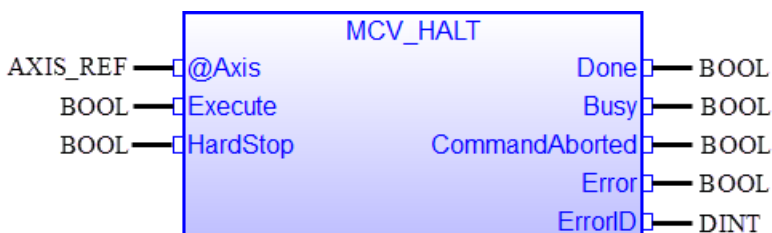


圖 4-2：“MC_Stop”與“MC_MoveVelocity”結合的行為

4.4 MCV_Halt

該 MFB 將軸切換到 “Standstill” 狀態。它命令軸進行運動停止，並在停止後轉換到 “Standstill” 狀態。該 MFB 首先將正在運動的軸設置為 “Stopping” 狀態；當速度降至零後，軸的狀態會自動切換為 “Standstill”。設置了 “Done” 輸出後，狀態將轉換為 “Standstill”。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	在訊號正緣開始動作
	HardStop	BOOL	<ul style="list-style-type: none"> FALSE: 使用加速時間作為減速時間 TRUE: 停止，不進行減速停止
VAR_OUTPUT	Done	BOOL	已達到零速度。狀態機已轉換為靜止狀態
	Busy	BOOL	軸仍在減速，尚未達到靜止狀態
	CommandAborted	BOOL	命令被另一個命令中止
	Error	BOOL	功能模塊內部發生錯誤
	ErrorID	DINT	錯誤識別

備註:

- “MCV_Halt” 不會被緩衝並立即生效。
- 以緩衝或混合模式發送的運動命令將從 FIFO 記憶體中刪除。
- 當 “MCV_Halt” 命令繁忙時，無法執行軸的其他運動命令（參見圖 4-3 中的 b）。
- 一旦 “MCV_Halt.Done” 為 TRUE，軸即可執行下一個運動命令。一旦 “MCV_Halt.Done” = TRUE 至少一個週期，就可以執行新的運動命令，即使 “MCV_Halt.Execute” 仍然是 TRUE（參見圖 4-3 中的 c）。與此相反，“MC_Stop” 僅在 “MC_Stop.Done” = TRUE 和 “MC_Stop.Execute” = FALSE 時，才允許執行下一個運動命令。
- 在軸減速期間，無法執行另一個運動指令（參見圖 4-3 中的 b）。當此 MFB 處於忙碌狀態時，無法被其他 MFB 中斷。
- “MCV_Halt” 不支援緩衝和混合模式。在另一個 “MCV_Halt” 控制同一軸時多次呼叫此 MFB 不會緩衝 MFB。

- 軸的減速度是前一個移動命令設置的加速度值（HardStop = FALSE）或零（HardStop = TRUE）。

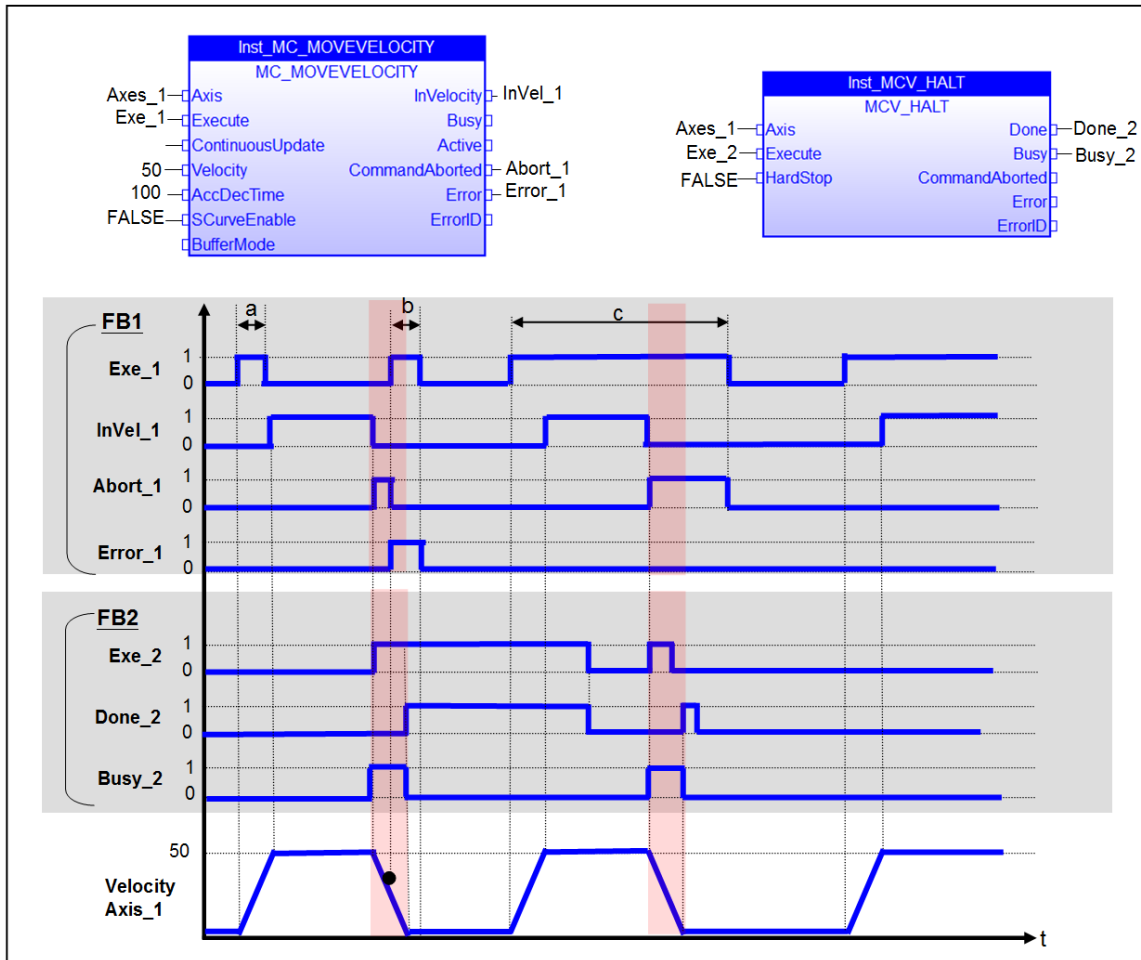
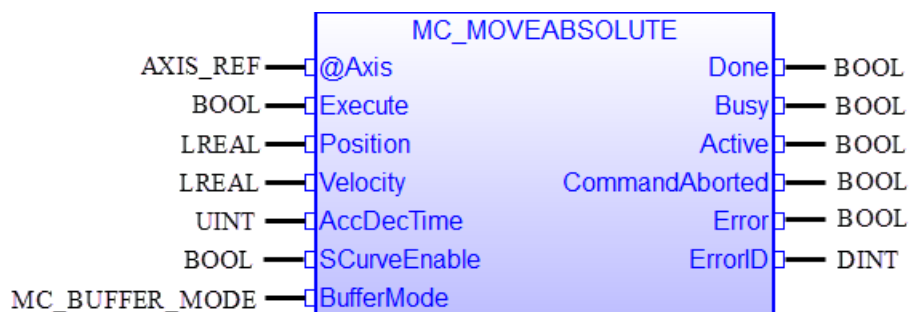


圖 4-3：“MCV_Halt”與“MC_MoveVelocity”結合的行為

4.5 MC_MoveAbsolute

移動到指定的絕對位置。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	在訊號正緣開始動作
	Position	LREAL	運動的命令位置 (Unit: user unit)
	Velocity	LREAL	最大速度的值 (Unit: user unit)。如果移動距離小於加速和減速距離的總和，則不一定能達到該速度
	AccDecTime	UINT	達到“速度”所需的加速和減速時間 (ms)
	SCurveEnable	BOOL	速度曲線： <ul style="list-style-type: none"> FALSE：梯形曲線 (T-curve) TRUE：正弦曲線 (S-curve)
	BufferMode	MC_BUFFER_MODE	定義 MFB 緩衝行為
VAR_OUTPUT	Done	BOOL	命令位置已達到
	Busy	BOOL	功能模塊 (MFB) 尚未完成，仍在處理命令
	Active	BOOL	功能模塊 (MFB) 已控制該軸
	CommandAborted	BOOL	命令被另一個命令中止
	Error	BOOL	功能模塊內部發生錯誤
	ErrorID	DINT	錯誤識別

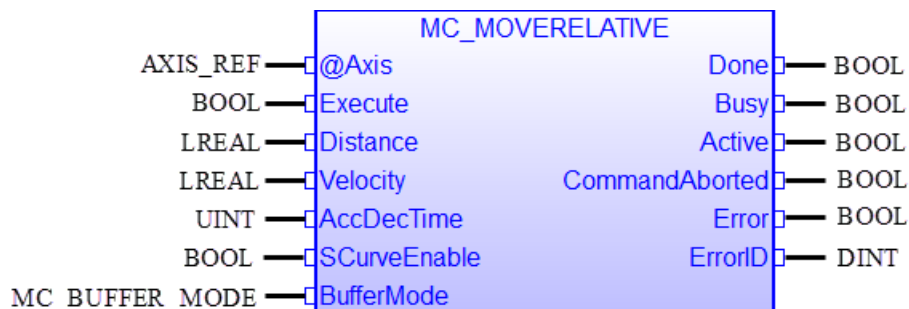
備註:

- 當此 MFB 處於活動狀態時，軸處於 “Discrete Motion” 狀態。
- 當達到軸的絕對位置時，“Done” 輸出將設置為 TRUE，軸狀態變為 “Standstill” 或開始執行另一個移動命令。
 - 如果沒有待處理的進一步操作，則此 MFB 以速度為零完成。
 - 在 “mcBuffered” 模式下：當運動完成後，軸返回到 “Standstill” 狀態。
- 此 MFB 不能與 “MC_MoveVelocity” 命令一起使用。

- 圖表：參見 PLCopen 技術規格書：“運動控制功能塊（2.0 版）”第 1 部分；圖 18，第 36/141 頁。

4.6 MC_MoveRelative

根據當前位置移動指定的距離。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	在訊號正緣開始動作
	Distance	LREAL	運動的相對距離 (Unit: user unit)
	Velocity	LREAL	最大速度的值 (Unit: user unit)。如果移動距離小於加速和減速距離的總和，則不一定能達到該速度
	AccDecTime	UINT	達到“Velocity”所需的加速和減速時間 (ms)
	SCurveEnable	BOOL	速度曲線： <ul style="list-style-type: none"> FALSE：梯形曲線 (T-curve) TRUE：正弦曲線 (S-curve)
	BufferMode	MC_BUFFER_MODE	定義 MFB 緩衝行為
VAR_OUTPUT	Done	BOOL	命令位置已達到
	Busy	BOOL	功能模塊 (MFB) 尚未完成，仍在處理命令
	Active	BOOL	功能模塊 (MFB) 已控制該軸
	CommandAborted	BOOL	命令被另一個命令中止
	Error	BOOL	功能模塊內部發生錯誤
	ErrorID	DINT	錯誤識別

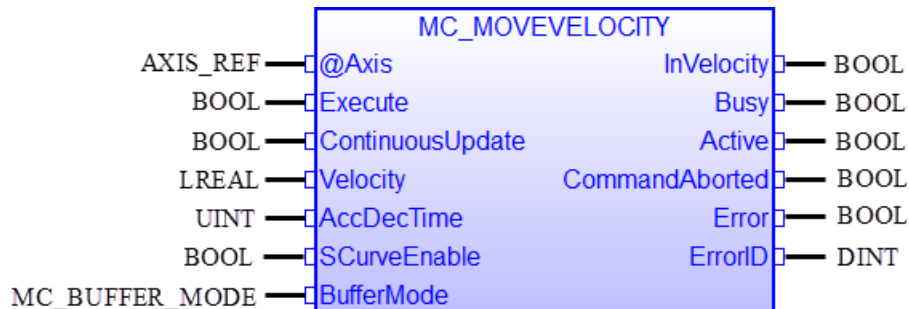
備註:

- 此 MFB 命令軸從當前位置移動指定的相對距離。
- 當此 MFB 處於活動狀態時，軸處於“Discrete Motion”狀態。
- 當軸達到絕對位置時，“Done”輸出將設置為 TRUE，軸的狀態變為“Standstill”或開始執行另一個運動命令。

- 如果沒有待處理的進一步操作，則此 MFB 以速度為零完成。
- 在 “*mcBuffered*” 模式下：當運動完成後，軸返回到 “*Standstill*” 狀態。
- 此 FB 不能與 “*MC_MoveVelocity*” 命令一起使用。
- 圖表：參見 PLCopen：“運動控制功能塊（2.0 版）” 第 1 部分；圖 19，第 38/141 頁。

4.7 MC_MoveVelocity

以指定的速度移動軸。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	在訊號正緣開始動作
	ContinuousUpdate	BOOL	決定在觸發“Execute”之後是否可以更新“Velocity”輸入： <ul style="list-style-type: none"> TRUE：新目標速度值將在每個週期中從“Velocity”讀取 FALSE：速度將無法更改
	Velocity	LREAL	命令速度的值 (Unit: user unit) <ul style="list-style-type: none"> 正值：向正方向移動 負值：向負方向移動
	AccDecTime	UINT	達到“速度”所需的加速和減速時間 (ms)
	SCurveEnable	BOOL	速度曲線： <ul style="list-style-type: none"> FALSE：梯形曲線 (T-curve) TRUE：正弦曲線 (S-curve)
	BufferMode	MC_BUFFER_MODE	僅支援“mcAborting”模式
VAR_OUTPUT	InVelocity	BOOL	表示已達到命令的“Velocity”
	Busy	BOOL	功能模塊 (MFB) 尚未完成，仍在處理命令
	Active	BOOL	功能模塊 (MFB) 已控制該軸
	CommandAborted	BOOL	命令被另一個命令中止
	Error	BOOL	功能模塊內部發生錯誤
	ErrorID	DINT	錯誤識別

備註:

- 此 MFB 命令軸移動至指定的速度。

- 要停止命令，必須用 “MC_STOP” 或 “MCV_HALT” 中斷此 MFB。
- “Velocity” 的值決定了速度和方向。使用正值表示正方向移動，使用負值表示負方向移動。
- 當此 MFB 處於活動狀態時，軸處於 “Continuous Motion” 狀態。
- 當軸達到命令的 “Velocity” 時，“InVelocity” 將設置為 TRUE。
- “MC_MoveAbsolute” 或 “MC_MoveRelative” 命令不能中止此 MFB。

4.8 MC_SetPosition

為當前編碼器位置和實際命令位置分配位置值。編碼器和命令位置都將設置為相同的值。在位置值更改期間不會發生任何移動。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	在訊號正緣開始動作
	Position	LREAL	指定設定的位置 (Unit: user unit)
	Relative	BOOL	指定相對位置或絕對位置。 <ul style="list-style-type: none">• TRUE：相對位置• FALSE：絕對位置
VAR_OUTPUT	Done	BOOL	當位置變更完成時為真
	Busy	BOOL	MFB 正在忙碌執行中
	Error	BOOL	MFB 內部發生錯誤
	ErrorID	DINT	錯誤識別 <ul style="list-style-type: none">• ECAT_ERR_MC_INVALID_AXIS_STATE:<ul style="list-style-type: none">- 當該軸的運動指令正在執行時，無法設置位置- 如果同一軸的另一個“MC_SetPosition”實例正在執行，則新的“MC_SetPosition”指令將無法執行。

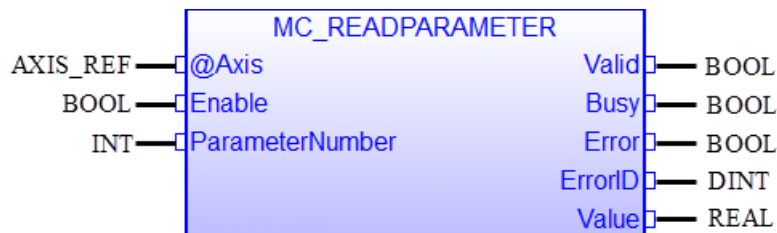
備註:

- “Relative” 是指在執行時將 “Position” 添加到軸的實際位置值中。這會導致按指定距離重新校準。
- “Absolute” 表示軸的實際位置值設置為 “Position” 輸入參數指定的值。
- “MC_SetPosition” 將伺服軸的當前位置和命令位置更改為指定的目標位置。
- 在執行運動命令時，不能呼叫 “MC_SetPosition”。
- 在內部執行時，“MC_SetPosition” 使用 CiA402 定義的歸位方法 37。如果伺服驅動器不支援

歸位方法 37，則 MFB 將失敗。“MC_SetPosition” 成功完成後，“MC_AxisReadInfo” 將顯示 “IsHome” 為真。如果伺服驅動器不支援歸位方法 37，則改用 “MC_Home” 並選擇歸位方法 35。

4.9 MC_ReadParameter

此 MFB 返回軸或 EtherCAT 主卡 ECAT-M801/e-M901 的配置參數值。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Enable	BOOL	當輸入為 TRUE 時，每個週期都會讀取參數值
	ParameterNumber	INT	表 7 中顯示的參數編號 (PN)
VAR_OUTPUT	Valid	BOOL	當輸出參數 “Value” 有效時為真
	Busy	BOOL	功能區塊正在忙於處理請求
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	Value	REAL	指定參數的值

備註:

- 配置參數由 “Axis” 和 “ParameterNumber” (PN) 輸入參數標識。PN 在表中定義 (表 7)。
- 只有 PN 表中指示為 REAL 的數據類型才能使用 “MC_ReadParameter” FB 讀取。
- PLCopen 定義用於設置或讀取最常見的軸參數，獨立的功能塊。為 PN 表中未列出的軸參數提供了單獨的功能塊。

4.9.1 參數表 Parameter Table

參數表是用於配置各個軸的參數清單。某些參數與 EtherCAT 主卡 (ECAT-M801/e-M901) 的配置相關。在初始化階段，如果 PLC 應用程式未設置軸和硬體，它們將自動設置為預設參數。應

用程式可以讀取表中“R”指示的參數以確認設置。“W”表示可以設置參數值。

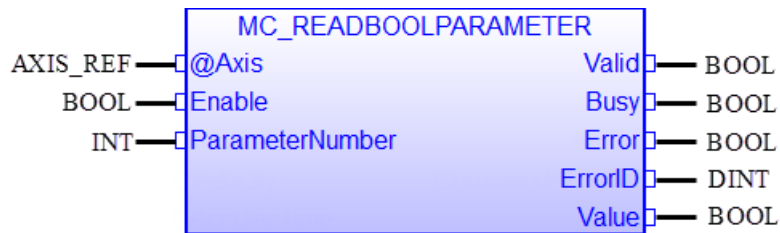
PN	名稱	資料類型	R/W	說明
2	SWLimitPos	REAL	R/W	正向軟體極限位置
3	SWLimitNeg	REAL	R/W	負向軟體極限位置
4	EnableLimitPos	BOOL	R/W	啟用正向軟體極限
5	EnableLimitNeg	BOOL	R/W	啟用負向軟體極限
1000	Pulse/unit	REAL	R/W	指定軸的每單位脈衝數
1001	Timeout	REAL	W	ECAT-M801/ECAT-M901 的響應超時
1002	Heartbeat time	REAL	W	PLC 應用程序必須在設置的時間間隔內向 ECAT-M801/ECAT-M901 發送一個命令，以表示 PLC 仍在運行。如果在時間間隔內未收到命令，則運動將自動停止
1003	Command buffer size	REAL	W	存儲在緩衝區中的 MFB 命令數量
1004	SWLimitDec	REAL	R/W	不支持 軟體極限減速值。一旦達到軟體限制，這個減速值將生效。
1005	LimitSwitchDec	REAL	R/W	不支持 極限開關減速值。此值決定硬件極限開關被觸發時，軸的減速度。 必須通過 SDO 功能塊設置。並非所有 CiA402 驅動器都支持此設置。

表 7：“ParameterNumber” (PN) 定義

有些參數設置會立即生效，有些參數設置會在軸執行時被拒絕。通常，最好在軸處於靜止狀態時設置命令。軸的執行應該在設置動作完成後再開始。

4.10 MC_ReadBoolParameter

此 MFB 傳回資料為 BOOL 類型特定參數的值。



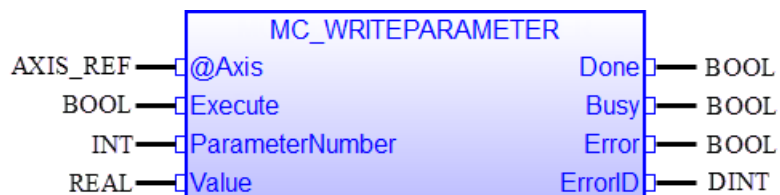
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Enable	BOOL	當輸入為 TRUE 時，每個週期都會讀取參數值
	ParameterNumber	INT	表 7 中顯示的參數編號 (PN)
VAR_OUTPUT	Valid	BOOL	當輸出參數 “Value” 有效時為真
	Busy	BOOL	功能區塊正在忙於處理請求
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	Value	BOOL	指定參數的值

備註:

- 配置參數由 “Axis” 和 “ParameterNumber” (PN) 輸入參數標識。PN 在表中定義 (表 7)。
- 只有 PN 表中 BOOL 指示的數據類型才能使用 “MC_ReadBoolParameter” MFB 存取。

4.11 MC_WriteParameter

此 MFB 設定軸參數的值。



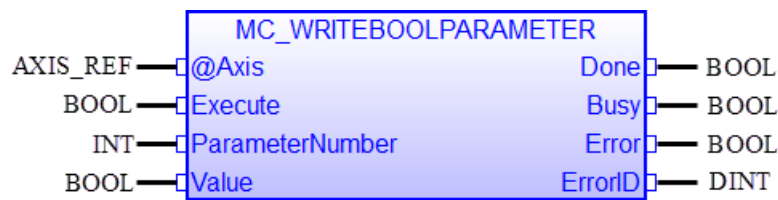
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	正緣觸發寫入命令執行
	ParameterNumber	INT	參數編號 (PN) 顯示在表 7
	Value	REAL	要分配給參數 (PN) 的新值
VAR_OUTPUT	Done	BOOL	表示輸入的 “Value” 已成功寫入到參數中
	Busy	BOOL	功能區塊正在忙於執行寫入命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別

備註:

- 配置參數由 “Axis” 和 “ParameterNumber” (PN) 輸入參數標識。 PN 在表中定義 (表 7)。
- 只有 PN 表中 REAL 指示的數據類型才能使用 “MC_WriteParameter” FB 存取。
- PLCopen 定義用於設置或讀取最常見的軸參數，獨立的功能塊。為 PN 表中未列出的軸參數提供了單獨的功能塊。

4.12 MC_WriteBoolParameter

此 MFB 修改所選 BOOL 類型的參數值。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸
VAR_INPUT	Execute	BOOL	正緣觸發寫入命令執行
	ParameterNumber	INT	參數編號 (PN) 顯示在表 7
	Value	BOOL	要分配給參數 (PN) 的新值
VAR_OUTPUT	Done	BOOL	表示輸入的 “Value” 已成功寫入到參數中
	Busy	BOOL	功能區塊正在忙於執行寫入命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別

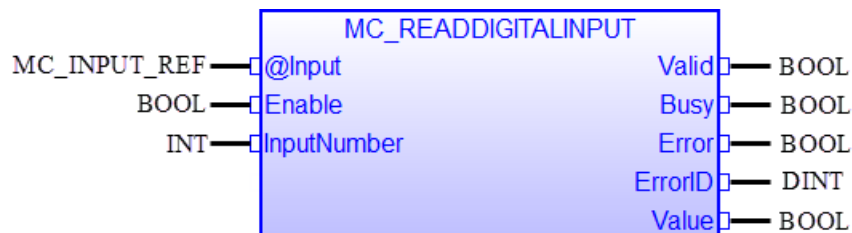
備註:

- 配置參數由 “Axis” 和 “ParameterNumber” (PN) 輸入參數標識。PN 在表中定義 (表 7)。
- 只有 PN 表中 BOOL 指示的數據類型可以使用 “MC_WriteBoolParameter” FB 進行修改。
- PLCopen 定義用於設置或讀取最常見的軸參數，獨立的功能塊。為 PN 表中未列出的軸參數提供了單獨的功能塊。

4.13 MC_ReadDigitalInput

此 FB 傳回數位輸入的值。可以存取三種不同類型的 DI 模組：

- ECAT-M801 卡的本地 DI
- EtherCAT DI 從站
- 遠端 EtherCAT 伺服驅動器 DI



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Input	MC_INPUT_REF	參考數位輸入 (DI)
VAR_INPUT	Enable	BOOL	啟用時，持續讀取數位輸入通道
	InputNumber	INT	通道號碼
VAR_OUTPUT	Valid	BOOL	輸出參數 “Value” 有效時為 TRUE
	Busy	BOOL	功能塊正在忙於執行讀取命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	Value	BOOL	選擇的輸入信號數值

備註:

- 如果 DI 信號寬度短於 PLC 週期時間，則可能會發生 FB 無法檢測到數位信號的情況。短暫的數字輸入信號可能會在 FB 下一個週期中讀取通道之前就結束。
- 數位輸入數據結構 “MC_INPUT_REF” 定義如下（圖 4-4）：
 - 數字輸入（DI）結構 “DiType”：
 - 0: 讀取遠端 EtherCAT 從站 DI
 - 1: 讀取軸 DI，它是伺服驅動器的一部分
 - 2: 讀取 ECAT-M801 卡的本地 DI
 - “CardNo”: ECAT-M801 的卡號。如果 “DiType” 為 0 或 1，則 FB 將忽略此變數
 - “SlaveN”: EtherCAT DI 從站的編號。僅當 “DiType” 為 0 時才有效。
 - “AxisNo”: 分配給伺服驅動器的軸號。僅對 “DiType” 1 有效。

Name	Type
DiRef	lib:MC_INPUT_REF

Variables		Properties
Name	Value	Type
DiRef		lib:MC_INPUT_REF
.CardNo		USINT
.DiType		USINT
.SlaveNo		UINT
.AxisNo		UINT

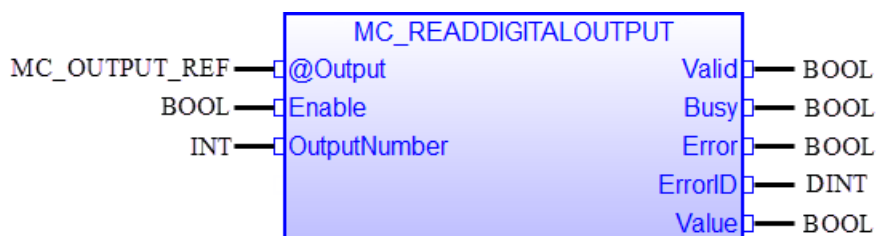
Blocks	Spylist	Public variables	Defi
--------	---------	------------------	------

圖 4-4：“MC_INPUT_REF”結構成員變數

4.14 MC_ReadDigitalOutput

此 FB 傳回數位輸出通道的值。可以存取兩種類型的 DO 模組：

- ECAT-M801 卡的本地 DO
- EtherCAT DO 從站。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Output	MC_OUTPUT_REF	參考數位輸出 (DO) 結構
VAR_INPUT	Enable	BOOL	啟用時，持續讀取數位輸入通道
	OutputNumber	INT	通道號碼
VAR_OUTPUT	Valid	BOOL	輸出參數 "Value" 有效時為 TRUE
	Busy	BOOL	功能塊正在忙於執行讀取命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	Value	BOOL	選擇的輸入信號數值

備註:

- “MC_ReadDigitalOutput” 的操作與 “MC_ReadDigitalInput” 相似。
- 數位輸出數據結構 “MC_OUTPUT_REF” 定義如下：
 - 數字輸出 (DO) 結構 “DoType”：
 - 0: 讀取遠端 EtherCAT 從站 DO
 - 1: 讀取軸 DO，它是伺服驅動器的一部分 (不支援)。
 - 2: 讀取 ECAT-M801 卡的本地 DO
 - “CardNo”: ECAT-M801 的卡號。如果 “DoType” 為 0 或 1，則 FB 將忽略此變數
 - “SlaveNo”: EtherCAT DO 從站的編號。僅當 “DoType” 為 0 時才有效。

4.15 MC_WriteDigitalOutput

此 FB 將值寫入選定的數位輸出通道。可以存取兩種類型的 DO 模組：

- ECAT-M801 卡的本地 DO
- EtherCAT DO 從站。



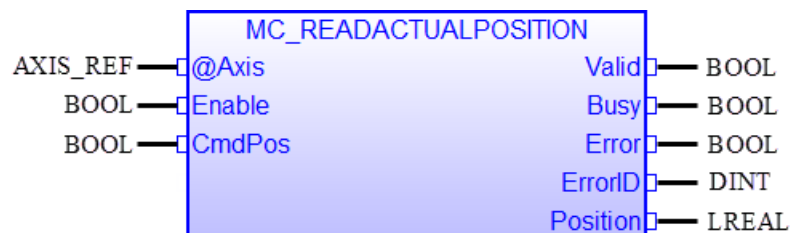
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Output	MC_OUTPUT_REF	參考數位輸出 (DO) 結構
VAR_INPUT	Execute	BOOL	正緣觸發執行 DO 寫入命令
	OutputNumber	INT	通道號碼
	Value	BOOL	所選數位輸出通道的值
VAR_OUTPUT	Done	BOOL	數位輸出通道已設定為 TRUE
	Busy	BOOL	功能塊正在忙於執行寫入命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別

備註:

- 數位輸出數據結構 “MC_OUTPUT_REF” 定義如下：
 - 數位輸出（DO）結構 “DoType”：
 - 0: 讀取遠端 EtherCAT 從站 DO
 - 1: 讀取伺服驅動器的軸 DO（不支援）。
 - 2: 讀取 ECAT-M801 卡的本地 DO
 - “CardNo”: ECAT-M801 的卡號。如果 “DoType” 為 0 或 1，則 FB 將忽略此變數
 - “SlaveNo”: EtherCAT DO 從站的編號。僅當 “DoType” 為 0 時才有效。

4.16 MC_ReadActualPosition

此 MFB 返回步進器/伺服驅動器的實際編碼器位置或 ECAT-M801/e-M901 卡內運動控制引擎的當前命令位置。



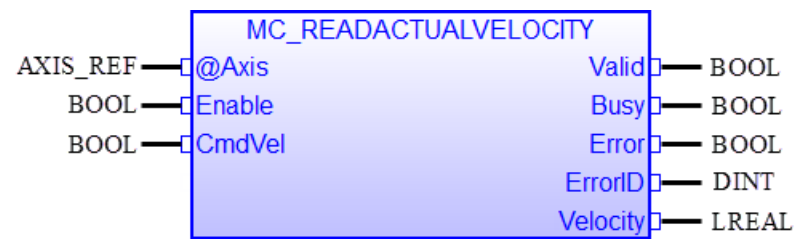
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸結構
VAR_INPUT	Enable	BOOL	在啟用的情況下，持續讀取當前位置。
	CmdPos	BOOL	FALSE: 讀取編碼器位置 (CiA402) TRUE: 讀取命令位置
VAR_OUTPUT	Valid	BOOL	當輸出參數 “Position” 有效時，為 TRUE。
	Busy	BOOL	功能塊正在忙於執行讀取命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	Position	LREAL	當前的編碼器位置或命令位置

備註:

- 為了降低運行時的反應時間，建議每個軸只使用一個 “MC_ReadActualPosition” 實例。

4.17 MC_ReadActualVelocity

該 MFB 返回步進器/伺服驅動器的實際速度或 ECAT-M801/e-M901 卡中之運動控制引擎的命令速度。



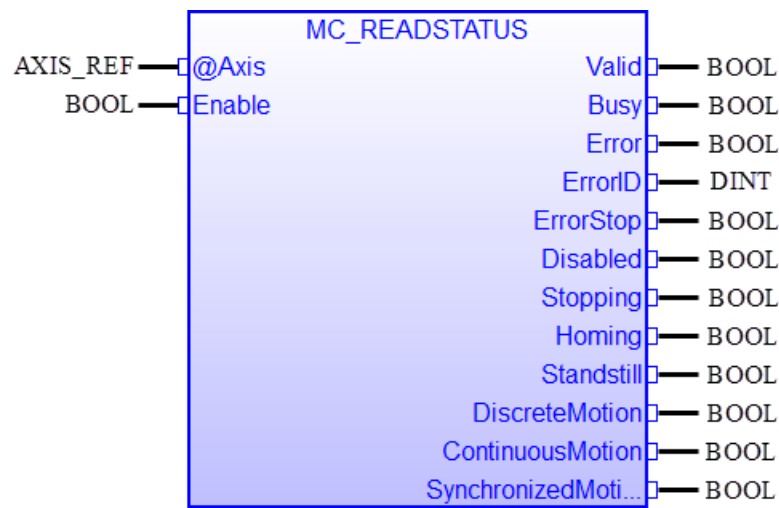
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸結構
VAR_INPUT	Enable	BOOL	在啟用時，持續讀取當前速度
	CmdVel	BOOL	FALSE: 讀取實際速度 (CiA402) TRUE: 讀取命令速度
VAR_OUTPUT	Valid	BOOL	當輸出參數 “Velocity” 有效時，為 TRUE
	Busy	BOOL	功能塊正在忙於執行讀取命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	Velocity	LREAL	當前的實際速度或命令速度

備註:

- 為了降低運行時的反應時間，建議每個軸只使用一個 “MC_ReadActualVelocity” 實例。

4.18 MC_ReadStatus

此 MFB 返回所選軸的狀態圖狀態。



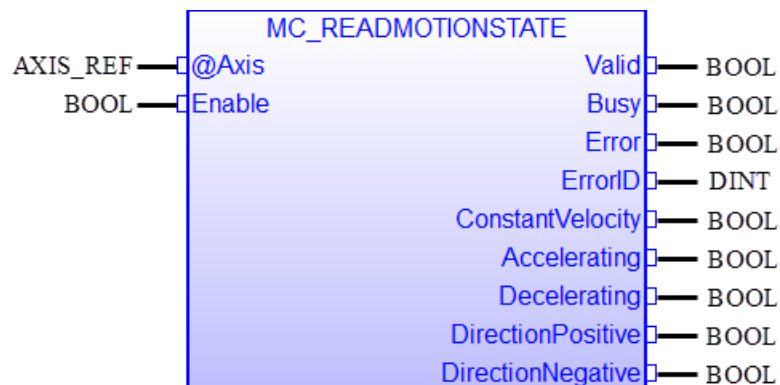
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸結構
VAR_INPUT	Enable	BOOL	當啟用時，持續讀取狀態圖的狀態
VAR_OUTPUT	Valid	BOOL	當所有輸出參數有效時，為 TRUE
	Busy	BOOL	功能塊正在忙於執行讀取命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	ErrorStop	BOOL	請參閱 PLCopen 狀態圖
	Disabled	BOOL	請參閱 PLCopen 狀態圖
	Stopping	BOOL	請參閱 PLCopen 狀態圖
	Homing	BOOL	請參閱 PLCopen 狀態圖
	Standstill	BOOL	請參閱 PLCopen 狀態圖
	DiscreteMotion	BOOL	請參閱 PLCopen 狀態圖
	ContinuousMotion	BOOL	請參閱 PLCopen 狀態圖
	SynchronizedMotion	BOOL	請參閱 PLCopen 狀態圖

備註:

-

4.19 MC_ReadMotionState

此 MFB 返回軸運動的狀態。



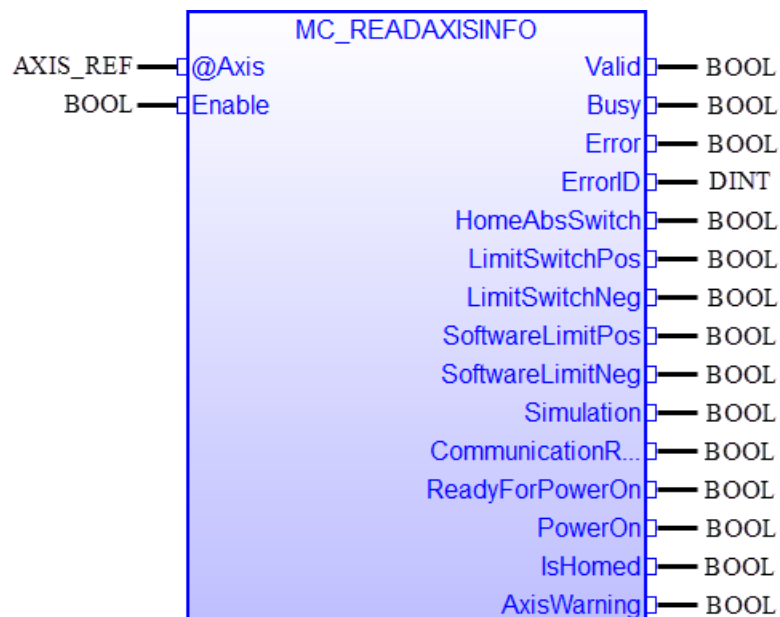
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸結構
VAR_INPUT	Enable	BOOL	在啟用時，持續讀取運動狀態
VAR_OUTPUT	Valid	BOOL	當所有輸出參數有效時，為 TRUE
	Busy	BOOL	功能塊正在忙於執行讀取命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	ConstantVelocity	BOOL	軸速度為恆定
	Accelerating	BOOL	軸正在加速
	Decelerating	BOOL	軸正在減速
	DirectionPositive	BOOL	軸正在朝正方向移動
	DirectionNegative	BOOL	軸正在朝負方向移動

備註:

- 當軸運動狀態的所有輸出都有效時，輸出 “Valid” 變為 TRUE。
- 如果 “Enable” 為 FALSE，則所有輸出數據均無效（“Valid” = FALSE）

4.20 MC_ReadAxisInfo

此 MFB 用於讀取軸上的各種狀態資訊。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸結構
VAR_INPUT	Enable	BOOL	在每個PLC週期中，當啟用時，持續讀取軸的資訊
VAR_OUTPUT	Valid	BOOL	當所有輸出參數有效時，為 TRUE
	Busy	BOOL	功能塊正在忙於執行讀取命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	HomeAbsSwitch	BOOL	當原點開關（數位輸入）啟動時，為 TRUE
	LimitSwitchPos	BOOL	當正向硬體極限開關（數位輸入）啟動時，為 TRUE
	LimitSwitchNeg	BOOL	當負向硬體極限開關（數位輸入）啟動時，為 TRUE
	SoftwareLimitPos	BOOL	當軸已達到正向軟體極限時，為 TRUE
	SoftwareLimitNeg	BOOL	當軸已達到負向軟體極限時，為 TRUE
	Simulation	BOOL	當軸處於模擬模式（例如，馬達被模擬）時，為 TRUE
	CommunicationReady	BOOL	當 EtherCAT 主機初始化並設置為 OP 模式時，為 TRUE
	ReadyForPowerOn	BOOL	當驅動器準備好啟用時，MC_Power 現在可以啟用
	PowerOn	BOOL	如果為 TRUE，則表示 MC_Power 已啟用，並且電源階段已開啟
	IsHomed	BOOL	如果為 TRUE，則表示 MC_Home 已成功執行且軸已知

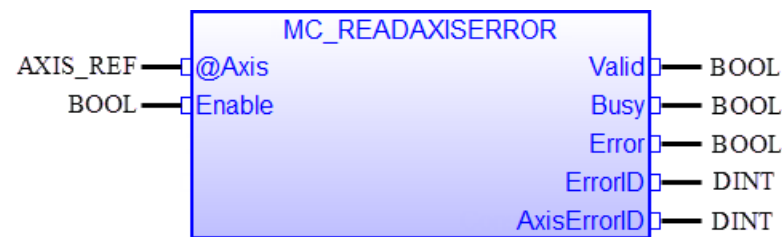
		絕對參考位置。
AxisWarning	BOOL	軸上存在警告

備註:

- 當軸運動狀態的所有輸出都有效時，輸出 “Valid” 變為 TRUE。
- 如果 “Enable” 為 FALSE，則所有輸出數據均無效（ “Valid” = FALSE ）
- 只要輸入 “Enable” 設置為 TRUE，MFB 就會讀取每個 POU 週期中的軸資訊。
- “CommunicationReady” 參數表示 EtherCAT 網絡中是否有任何軸處於離線狀態。一旦與任何一個軸的通訊失敗，所有軸的 “CommunicationReady” 輸出參數將設置為 TRUE。因此，它不表示單個軸的通訊狀態，而是表示所有軸的通訊狀態。

4.21 MC_ReadAxisError

MFB 讀取導致突然停止或阻止軸到達目標位置的錯誤。

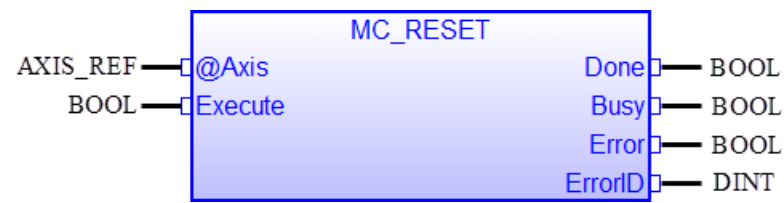


輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸結構
VAR_INPUT	Enable	BOOL	在啟用時，持續讀取當前的軸錯誤狀態
VAR_OUTPUT	Valid	BOOL	當輸出參數 “AxisErrorID” 有效時，為 TRUE
	Busy	BOOL	功能塊正在忙於執行讀取命令
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別
	AxisErrorID	DINT	軸錯誤的值。請參見軸錯誤表

備註:

4.22 MC_Reset

如果軸處於 “ErrorStop” 狀態，則必須呼叫此 MFB，以清除和重置軸錯誤。成功的重置執行會將軸的狀態從 “ErrorStop” 轉換為 “Standstill”（參見狀態圖）。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	Axis	AXIS_REF	參考軸結構
VAR_INPUT	Execute	BOOL	重置所有與軸相關的內部錯誤
VAR_OUTPUT	Done	BOOL	TRUE：所有軸錯誤已被重置，並且軸狀態已轉換為 “Standstill” 或 “Disabled” 狀態
	Busy	BOOL	功能區塊正在忙於重置軸錯誤
	Error	BOOL	功能區塊內部發生錯誤
	ErrorID	DINT	錯誤識別

備註:

5 協調運動

5.1 功能塊概述

下表概述了受支援的多軸 MFB。它們分為管理和運動相關類別。管理 MFB 不直接控制軸的移動，而是讀取分組狀態並將軸分配給群組。

Administrative	Group Motion
MC_AddAxisToGroup	MC_GroupStop
MC_RemoveAxisFromGroup	MCV_GroupHalt
MC_UngroupAllAxes	MC_GroupInterrupt
MC_GroupEnable	MC_GroupContinue
MC_GroupDisable	MC_MoveLinearAbsolute
MC_GroupReadActualPosition	MC_MoveLinearrelative
MC_GroupReadActualVelocity	MC_MoveCircularAbsolute
MC_GroupReadStatus	MC_MoveCircularRelative
MC_GroupReadError	
MC_GroupReset	

表 8：對功能塊進行分組

5.2 狀態圖

狀態圖（圖 5-1）描述了群組軸的狀態。它允許在特定狀態下執行哪些組 MFB。群組狀態不會影響管理 MFB，這些 MFB 會返回當前群組狀態、位置、速度等。當軸處於群組狀態時，單軸狀態圖也會對每個軸處於活動狀態。

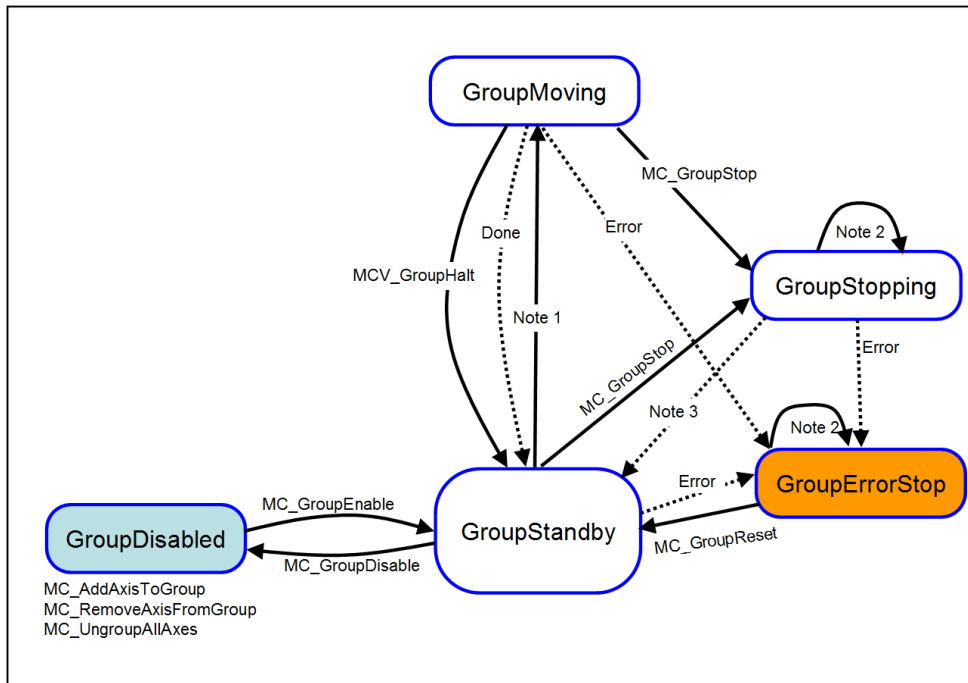


圖 5-1：群組狀態圖

群組狀態圖（圖 5-1）的備註：

狀態轉換的注意事項: 連續線是命令轉換；虛線是自動轉換。

- 註 1: 適用於所有群組運動 MFB
- 註 2: 在 “GroupErrorStop” 和 “GroupStopping” 狀態下，可以呼叫所有 MFB，但群組運動 MFB 將無法執行並返回錯誤。首先，必須通過分別呼叫 “MC_GroupReset” 和 “MC_GroupStop” 將狀態轉換為 “GroupStandby”，然後 MFB 才能控制群組。
- 註 3: 當 “MC_GroupStop.Done” 為 TRUE 且 “MC_GroupStop.Execute” 已設置為 FALSE 時，群組會從 “GroupStopping” 狀態自動轉換為 “GroupStandby”。
- 註 4（圖中未顯示）： “MC_GroupDisable” 可以在所有狀態下執行，並將狀態更改為 “GroupDisabled”。

下面更詳細地介紹了群組狀態和轉換：

5.2.1 GroupDisabled

- 這是分配軸號（ “MCV_AxisAssign” ）後的群組初始狀態，用於為每個伺服/步進從站分配一個軸號，以便於識別。軸分配後，所有組都是空的，需要（ “MC_AddAxisToGroup” ）將軸添加至群組中。要控制群組，必須透過 “MC_GroupEnable” 啟用它，會把狀態由 “GroupDisabled” 轉換為 “GroupStandby” 狀態

- 只有當其所有軸都處於 “Standstill” 狀態時，才能啟用該群組。
 - i. 如果其中一個軸處於 “ErrorStop” 狀態，則必須解決並清除該錯誤（“MC_Reset”）。
 - ii. 如果其中一個軸處於 “Stopping” 狀態，則透過將 “MC_Stop.Execute” 設置為 FALSE 來釋放 “MC_Stop” 命令。
- 如果組在 “GroupStandby” 狀態時遇到錯誤或 “MC_GroupStop” 命令處於活動狀態，則其狀態將立即轉換為 “GroupErrorStop” 或 “GroupStopping”。
- 一旦啟用了群組，單軸運動命令就無法控制群組內的軸，也就是不能發出 “MC_Stop”、“MC_MoveAbsolute” 等單軸運動 FB。單軸管理 FB 仍允許讀取軸狀態。
- 不支援將歸位（Homing）作為群組命令。停用該群組以執行單軸歸位零點搜尋。

5.2.2 GroupStandby

- 啟用群組後，它會轉換為 “GroupStandby”。群組內所有軸都處於靜止狀態，並準備執行組運動命令。
- 如果群組內任一軸出現錯誤，狀態將更改為 “GroupErrorStop”，只能透過執行 “MC_ResetGroup” 來離開。
- “MC_GroupStop” 將狀態更改為 “GroupStopping”。
- 在 “GroupStandby” 中，群組內各軸的狀態維持在 “Standstill”。

5.2.3 GroupMoving

- 下列五個運動命令會轉換狀態至 “GroupMoving”：
 - i. MC_MoveLinearAbsolute
 - ii. MC_MoveLinearRelative
 - iii. MC_MoveCircAbs
 - iv. MC_MoveCircRel
 - v. MC_GroupMoveIncPath
- 群組將保持此狀態，直到命令完成（“Done” = TRUE）、遇到錯誤（例如，觸發了硬體極限）或發出停止命令 “MC_GroupStop” 或 “MCV_Halt”。
- 群組運動命令會一直讓單軸狀態更改為 “SynchronizedMotion”。

5.2.4 GroupStopping

- 這種狀態由 “MC_GroupStop” FB 控制。“MC_GroupStop.Execute” = TRUE 觸發群組停止命令並將狀態更改為 “GroupStopping”。群組在 “Execute” 為 TRUE 時會保持此狀態。

“MC_GroupStop.Done” = TRUE 表示所有軸都處於靜止狀態。一旦滿足兩個條件

“MC_GroupStop.Done” = TRUE 和 “MC_GroupStop.Execute” = FALSE，該組就會自動切換到狀態 “GroupStandby”。

- 當群組軸停止，單軸狀態將從 “SynchronizedMotion” 轉換為 “Standstill”。

5.2.5 GroupErrorStop

- 當群組中一個軸觸發硬體極限開關、超過軟體極限或遇到伺服驅動器錯誤時，會發生組錯誤停止。錯誤的 EtherCAT 連接也可能導致錯誤停止。在大多數情況下，單軸 “ErrorStop” 會導致 “GroupErrorStop”，因為單軸錯誤會影響整個群組。
- 解決錯誤原因後，必須通過 “MC_GroupReset” 清除錯誤。該命令不僅可以清除群組錯誤，還可以清除單軸錯誤。例如：當群組中的單軸達到軟體極限後，軸和群組狀態將變為錯誤停止。執行 “MC_GroupReset” 將同時清除單軸和群組錯誤。

5.3 單軸與群組狀態圖的關係

一旦通過 “MC_GroupEnable” 啟用群組，就無法發出單軸運動命令，例如

“MC_MoveAbsolute”。單軸 MFB 錯誤輸出將設置為 TRUE，以指示該命令尚未被接受。單軸命令不會影響群組狀態及其運動，例如：如果呼叫單軸移動命令，則移動中的群組將繼續而不會中斷。

如果群組處於禁用狀態，則單軸運動命令可以控制群組中的每個軸。在單軸命令控制了群組中的某個軸時，群組無法啟用以進行群組命令控制，直到單軸命令完成並停止為止。

欲啟用一個群組，所有群組成員軸都必須在 “Standstill” 狀態。

單軸與其群組之間交互的一般規則：

- 如果群組中至少有一個軸在移動，則該組群處於 “GroupMoving” 狀態
- 如果所有軸都處於 “Standstill” 狀態，則該群組處於 “GroupStandby”、“GroupDisabled” 或 “GroupErrorStop” 狀態
- 如果一個軸位於 “ErrorStop” 中，則整個群組位於 “GroupErrorStop” 狀態
- 如果已啟用群組，則無法發出單軸 “MC_Home” 的命令
- 如果已啟用群組，則無法發出單軸 “MC_Stop” 的命令

- 如果已啟用群組，則無法從群組中刪除單軸。

群組與其中的單軸之間交互的一般規則：

- 若群組被群組移動命令控制，所有群組內的單軸皆會呈現 “*SynchronizedMotion*” 狀態
- 如果群組處於 “*GroupStandby*” 狀態，則單軸的狀態將處於 “*StandStill*” 狀態。

群組運動命令對單軸狀態的影響概述：

命令	組狀態	軸狀態
MC_MoveLinear... MC_MoveCircular...	GroupMoving	SynchronizedMotion
MC_GroupStop	GroupStopping / GroupStandby	SynchronizedMotion / StandStill
MC_GroupReset	GroupErrorStop / GroupStandby	One or more axis in ErrorStop

表 9：群運動命令對單軸狀態的影響

5.4 運動的混合和緩衝

5.4.1 混合

對於需要在兩個不同的命令移動之間連續運動的應用程式，需要進行混合。對於單軸，始終達到每個連續運動命令的結束位置；第一個命令的命令結束位置會成為下一個命令的開始位置。

但是，在混合插值運動中，路徑不會通過原始軌跡中的所有點。混合修改了原始路徑，以實現無尖角的平滑軌跡。以這樣的方式修改第一個插值命令的結束位置和第二個命令的開始位置，以防止軸停止運動。

目前僅支援混合模式“*TMStartAccNext*” (表 10)。此模式在減速階段決定下一個命令控制軸群組的時間。拐角輪廓由“*TransitionPara*”定義，它指定了下一個命令控制的程式加速時間的百分比。

No.	MC_TRANSITION_MODE	描述
0	TMNone	不插入過渡曲線
1	TMStartVelocity	不支援
2	TMConstVelocity	不支援
3	TMConerDistance	不支援
4	TMConerDeviation	不支援
5 - 9	Reserved by PLCopen	不支援
10	TMStartAccNext	設定混合百分比

表 10：過渡模式概述

例如，軌跡被指定為由多段路徑組成的序列，其中每段路徑都由一個單獨的命令控制。計算每個路徑段的速度曲線，其中每個速度曲線被劃分為混合進入區域、混合退出區域和剩餘區域。執行每段路徑時，其速度曲線的混合進入區域僅與前一個速度曲線的混合退出區域重疊。“*TransitionPara*”決定了在減速曲線上下一段路徑的混合進入區域的開始時間 (表 11)。

- “*TransitionPara*” = 0

不發生混合，群組軸將在每段結束時減速為零，然後再加速到下一段路徑的目標速度。

- “*TransitionPara*” = 50

混合進入和混合退出區域的重疊在路徑段的減速時間經過 50% 後開始。

- “*TransitionPara*” = 100

當路徑段開始減速時，混合進入和混合退出區域的重疊立即開始。

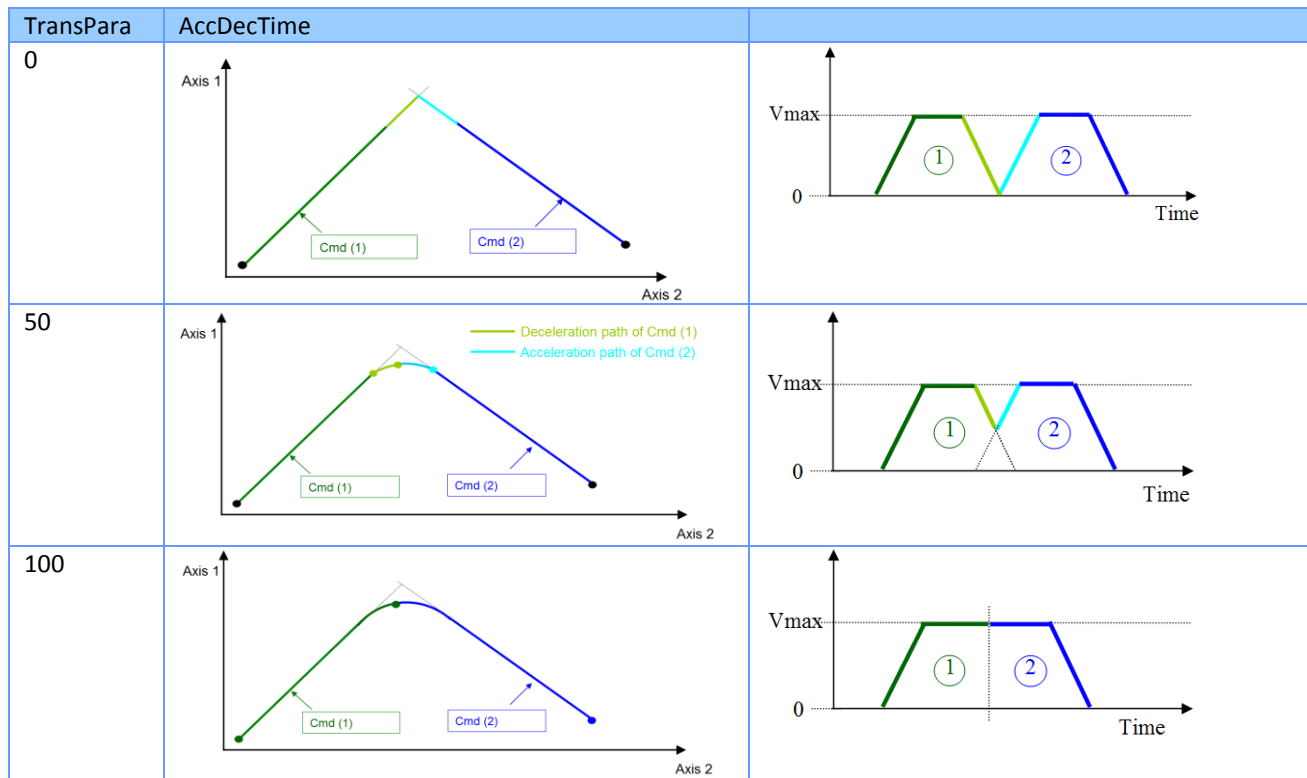


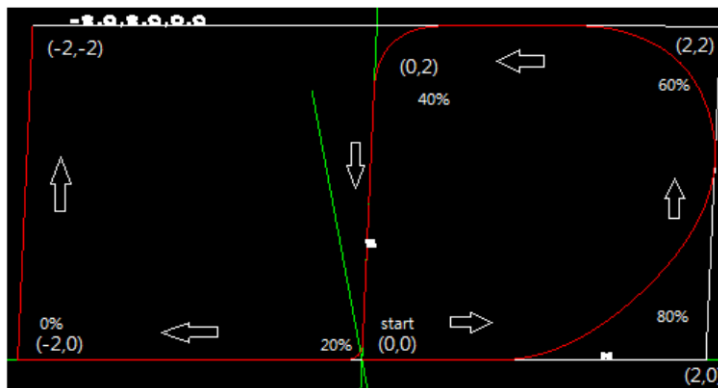
表 11：“mcBlendingNext”的過渡參數設置

注意：

- 每個運動命令的加速和減速時間相等。
- 加速和減速時間只能在第一段路徑運動開始前設置。運動開始後，時間無法修改，後續所有路徑將引用相同的時間設置。

例:

錯誤! 書籤的自我參照不正確。2 顯示由於不同的“TransitionPara”設置（百分比值）而導致的原始路徑偏差。



錯誤! 書籤的自我參照不正確。2：具有不同“TransitionPara”設置的輪廓曲線

5.4.2 緩衝模式

對於群組運動，使用與單軸 MFB 相同的緩衝模式 (表 12).

No.	MC_BUFFER_MODE	描述
0	mcAborting	立即啟動 MFB
1	mcBuffered	緩衝區中的所有命令完成後啟動 MFB
2	mcBlendingLow	不支援
3	mcBlendingPrevious	不支援
4	mcBlendingNext	該速度與第二個 FB 的速度混合
5	mcBlendingHigh	不支援

表 12：緩衝模式概述

支援的緩衝模式及其與過渡模式和參數的關係如 表 13. 速度時間圖中每個帶圓圈的數字代表一個新的群組命令。

如果將 “BufferMode” 設置為 “mcAborting”或 “mcBuffered”，則 “TransitionMode” 設置將被忽略。

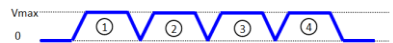
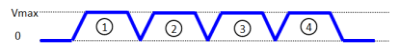

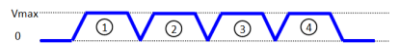

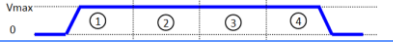
BufferMode	TransitionMode	TransPara	AccDecTime	Velocity-Time
mcBuffered	TMNone	0 ... 100	1...999	
	TMStartAccNext	0 ... 100	1...999	
mcBlendingNext	TMNone	0 ... 100	1...999	
	TMStartAccNext	0	1..999	
	TMStartAccNext	50	1..999	
	TMStartAccNext	100	1..999	

表 13：緩衝模式

6 群組運動功能塊

本章介紹 Win-GRAF PLCopen 函式庫，用於線性和圓弧插值等群組運動控制功能塊。

6.1 MC_AddAxisToGroup

此 MFB 將一個軸添加到群組中。最多可將 32 個軸添加到一個群組中。ECAT-M801/M901 最多支援 8 個群組。只有當軸處於禁用狀態（“GroupStandby”）時，才能在群組中添加或刪除軸。將軸添加到群組不會自動啟用該群組。在使用群組運動命令之前，請在分配所有軸之後啟用群組（“MC_GroupEnable”）。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
	Axis	AXIS_REF	參考要添加的軸
VAR_INPUT	Execute	BOOL	在正緣開始動作
VAR_OUTPUT	Done	BOOL	當軸成功添加到群組中時，為 TRUE
	Busy	BOOL	功能塊正在執行中
	Error	BOOL	在將軸添加到群組時發生錯誤
	ErrorID	DINT	錯誤識別

備註:

- 如果軸已成功添加到群組中，或者在 MFB 執行之前已經在群組中，則 MFB 將設置為“Done”。
- 當群組處於“GroupDisabled”狀態時，只能呼叫此 MFB。
- 一旦觸發“MCV_AXISASSIGN.Execute”，所有軸將自動從所有群組中刪除。
- 可以將同一軸分配給多個群組，但在同時共享該軸的群組中，僅能啟用其中一個群組。

例:

以下範例顯示了使用 “MC_AddAxisToGroup” 的 PLC 程式的節錄：

VAR

```
Inst_MC_ADDAXISTOGROUP : ARRAY [0 .. 5] OF MC_AddAxisToGroup ;
```

```
AxesGroup : ARRAY [0 .. 2] OF lib:AXES_GROUP_REF ;
```

```
flgAddToGroup : BOOL := TRUE ;
```

```
AxisRef : ARRAY [0 .. 5] OF lib:AXIS_REF ;
```

END_VAR

```
//=====
```

```
//Add to Group:
```

```
//=====
```

```
AxesGroup[0].GroupNo := 0;
```

```
AxesGroup[0].CardNo := AxisRef[0].CardNo;
```

```
AxesGroup[1].GroupNo := 1;
```

```
AxesGroup[1].CardNo := AxisRef[1].CardNo;
```

```
AxesGroup[2].GroupNo := 2;
```

```
AxesGroup[2].CardNo := AxisRef[2].CardNo;
```

```
//Add axis 0 to group 0:
```

```
Inst_MC_ADDAXISTOGROUP[0]( AxesGroup[0](*lib:AXES_GROUP_REF*), AxisRef[0](*lib:AXIS_REF*),  
flgAddToGroup(*BOOL*) );
```

```
//Add axis 1 to group 0 after axis 0 has been added:
```

```
if Inst_MC_ADDAXISTOGROUP[0].Done = TRUE then
```

```
    Inst_MC_ADDAXISTOGROUP[1]( AxesGroup[0](*lib:AXES_GROUP_REF*), AxisRef[1](*lib:AXIS_REF*),  
    flgAddToGroup(*BOOL*) );
```

```
end_if;
```

```
//Add axis 2 to group 1:
```

```
if Inst_MC_ADDAXISTOGROUP[1].Done = TRUE then
```

```
    Inst_MC_ADDAXISTOGROUP[2]( AxesGroup[1](*lib:AXES_GROUP_REF*), AxisRef[2](*lib:AXIS_REF*),  
    flgAddToGroup(*BOOL*) );
```

```
end_if;
```

```

//Add axis 3 to group 1 after axis 2 has been added::
if Inst_MC_ADDAXISTOGROUP[2].Done = TRUE then
    Inst_MC_ADDAXISTOGROUP[3]( AxesGroup[1](*lib:AXES_GROUP_REF*), AxisRef[3](*lib:AXIS_REF*),
    flgAddToGroup(*BOOL*) );
end_if;

//Add axis 4 to group 2:
if Inst_MC_ADDAXISTOGROUP[3].Done = TRUE then
    Inst_MC_ADDAXISTOGROUP[4]( AxesGroup[2](*lib:AXES_GROUP_REF*), AxisRef[4](*lib:AXIS_REF*),
    flgAddToGroup(*BOOL*) );
end_if;

//Add axis 5 to group 2 after axis 4 has been added::
if Inst_MC_ADDAXISTOGROUP[4].Done = TRUE then
    Inst_MC_ADDAXISTOGROUP[5]( AxesGroup[2](*lib:AXES_GROUP_REF*), AxisRef[5](*lib:AXIS_REF*),
    flgAddToGroup(*BOOL*) );
end_if;

```

6.2 MC_RemoveAxisFromGroup

此 MFB 會從群組中刪除一個軸。只能在群組處於（“GroupDisabled”）時，才能呼叫此 MFB。



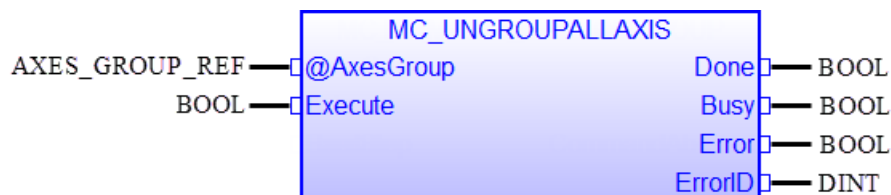
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
	Axis	AXIS_REF	參考要移除的軸
VAR_INPUT	Execute	BOOL	在正緣觸發時移除軸
VAR_OUTPUT	Done	BOOL	當軸成功從群組中移除時，為 TRUE
	Busy	BOOL	TRUE - 功能區塊正在忙於執行。當 Busy 變為 FALSE 時，功能區塊準備接受新的指令
	Error	BOOL	在將軸移除群組時發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 如果軸不在群組中，則 MFB 返回 “Done” = TRUE。

6.3 MC_UngroupAllAxes

此 MFB 會從群組中刪除所有軸。只能在群組處於（“GroupDisabled”）時，才能呼叫此 MFB。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	在正緣觸發時移除所有軸
VAR_OUTPUT	Done	BOOL	當所有軸成功從群組中移除時，為 TRUE
	Busy	BOOL	TRUE - 功能區塊正在忙於執行。當 Busy 變為 FALSE 時，功能區塊準備接受新的指令
	Error	BOOL	在將軸移除群組時發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

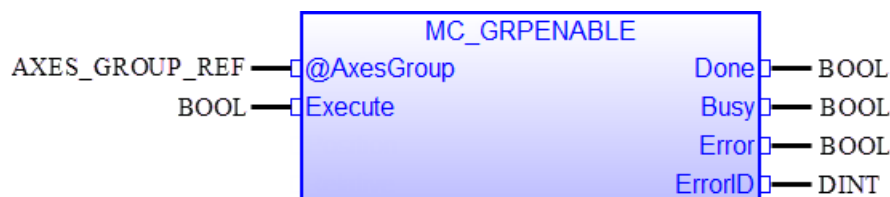
備註:

- 如果沒有軸在群組內，MFB 將回傳 “Done” = TRUE。這意味著如果群組在呼叫此 FB 之前已經為空，MFB 也會回傳 “Done” = TRUE。

6.4 MC_GroupEnable

此 MFB 將群組的狀態從 “GroupDisabled” 更改為 “GroupStandby”。在呼叫群組移動 MFB 之前，必須啟用群組。

可以同時啟用多個群組，但不能啟用包含同一軸的多個群組。如果同一軸屬於不同的群組，則必須先禁用帶有該軸的當前活動群組，然後再啟用包含相同軸的下一個群組。群組成員軸都必須處於 “Standstill” 狀態，否則無法啟用該群組。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	在正緣觸發時啟用群組
VAR_OUTPUT	Done	BOOL	當組別成功啟用時，為 TRUE
	Busy	BOOL	TRUE - 功能區塊正在忙於執行。當 Busy 變為 FALSE 時，功能區塊準備接受新的指令。
	Error	BOOL	在將軸啟用群組時發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 在以下情況下，無法啟用群組：
 - 如果未通過 “MC_AddAxisToGrp” 將軸添加到群組（群組為空）
 - 如果要啟用的群組的軸屬於其他已啟用的群組
 - 如果群組的某個軸正在執行單軸運動命令（例如，軸由單軸命令控制）。
 - 如果群組的某個軸處於 “Disabled”、“Stopping” 或 “ErrorStop” 狀態。
- 注意：
 - 當軸已屬於啟用的群組時，則無法執行單軸運動命令。如果在軸分配給啟用的群組時呼叫 “MC_Stop”、“MC_MoveAbsolute”、“MC_MoveRelative” 等命令，將回傳錯誤。
 - 當軸屬於啟用的群組時，仍然可以呼叫單軸管理 MFB。
- 如果在啟用群組時 “MC_GrpStop.Execute” 為 TRUE，則該群組將切換到 “GroupStopping” 狀

態。

- 函數 “MC_AddAxisToGrp” 、 “MC_RemoveAxisfromGrp” 或 “MC_UngroupAllAxis” 不能在已啟用的群組上執行。

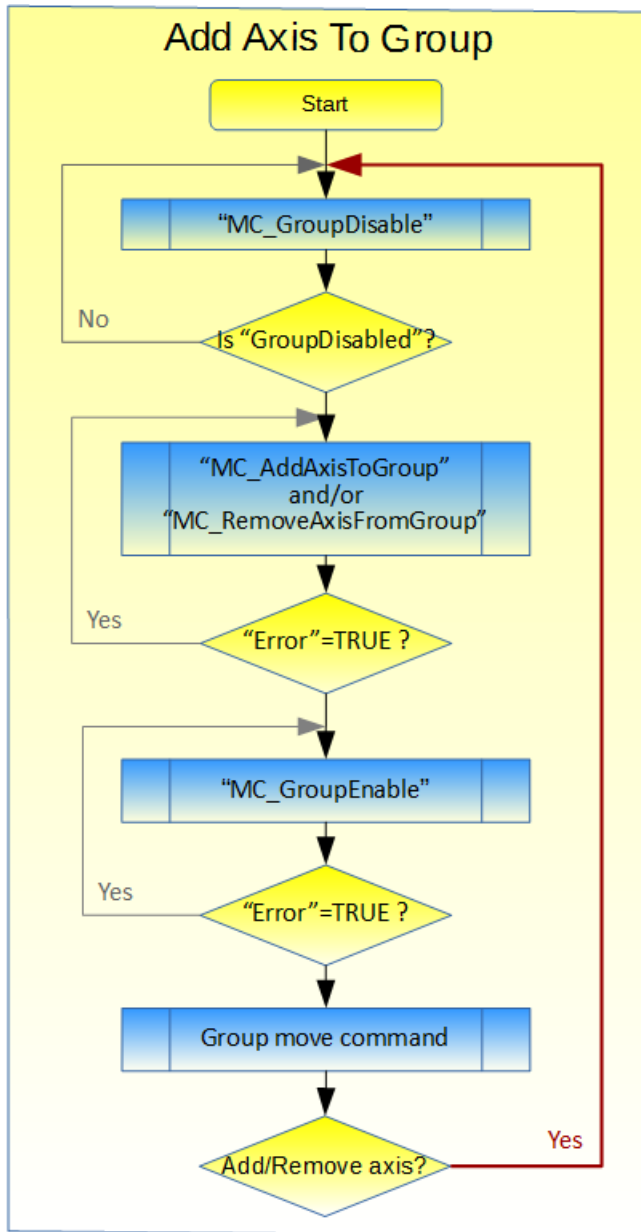
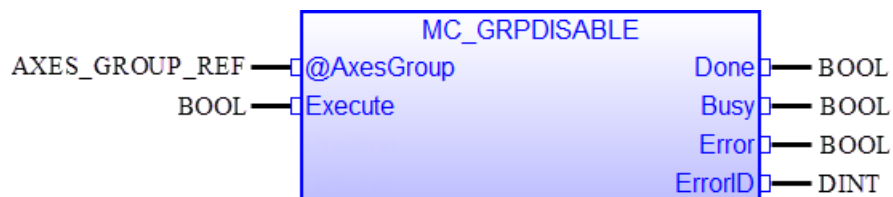


圖 6-1：新增/移除軸到群組

6.5 MC_GroupDisable

此 MFB 將群組的狀態更改為 “GroupDisabled”。在執行 “MC_GrpDisable” 之前，應先通過呼叫 “MC_GrpHalt” 或 “MC_GrpStop” 來停止正在運動的群組，否則將導致立即停止。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	在正緣觸發時禁用群組
VAR_OUTPUT	Done	BOOL	當組別成功禁用時，為 TRUE
	Busy	BOOL	TRUE - 功能區塊正在忙於執行。當 Busy 變為 FALSE 時，功能區塊準備接受新的指令。
	Error	BOOL	在將軸禁用群組時發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 透過這個 MFB，正在運動中的群組會立刻停止。突然停止可能會超過允許的伺服驅動器減速度限制。根據驅動器硬體的不同，這可能會導致功率峰值和運行時錯誤。為防止突然停止，首先必須通過執行 “MC_GrpHalt” 或 “MC_GrpStop” 將運動的群組減速至靜止狀態，然後再禁用該群組。
- 在 “GroupMoving”、 “GroupStopping”、 “GroupErrorStop” 和 “GroupStandby” 中，可以禁用該群組。
- 禁用群組時，將在內部執行以下操作：
 - 群組錯誤 “GroupErrorStop” 將自動重置
 - 即使 “MC_GrpStop.Execute” 仍處於活動狀態，停止狀態 “GroupStopping” 也將被清除。當 “MC_GrpStop.Execute” 為 TRUE 時，該群組在啟用時返回到 “GroupStopping”。

6.6 MC_GroupReadActualPosition

該 MFB 為群組中的每個軸回傳步進/伺服驅動器的實際編碼器位置或 ECAT-M801/e-M901 卡內運動控制引擎的當前命令位置。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Enable	BOOL	<ul style="list-style-type: none">TRUE：持續獲取群組軸的位置信息
	CmdPos	BOOL	<ul style="list-style-type: none">FALSE：讀取實際位置TRUE：讀取命令位置
VAR_OUTPUT	Valid	BOOL	<ul style="list-style-type: none">TRUE：表示位置輸出值有效
	Busy	BOOL	<ul style="list-style-type: none">TRUE：功能區塊正在忙於獲取位置信息
	Error	BOOL	<ul style="list-style-type: none">TRUE：在讀取軸的位置時發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼
	Position	ARRAY[1..N] OF LREAL	群組的當前或命令位置值 注意： 雖然該參數顯示在功能區塊的輸入端，但它實際上作為輸出參數

備註:

6.7 MC_GroupReadActualVelocity

該 MFB 為群組中的每個軸回傳步進/伺服驅動器的實際速度或 ECAT-M801/e-M901 卡內運動控制引擎的指令速度。

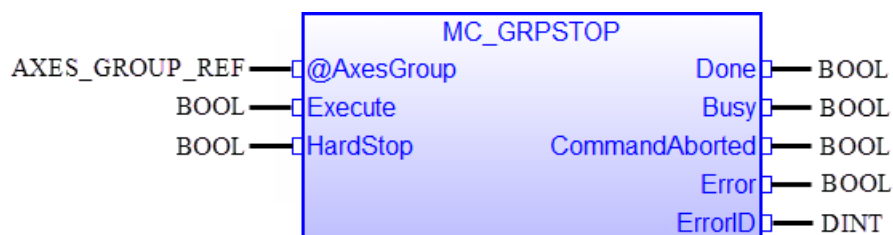


輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Enable	BOOL	• TRUE：持續獲取群組軸的速度
	CmdPos	BOOL	• FALSE：讀取實際速度 • TRUE：讀取命令速度
VAR_OUTPUT	Valid	BOOL	• TRUE：表示速度輸出值有效
	Busy	BOOL	• TRUE：功能區塊正在忙於獲取速度值
	Error	BOOL	• TRUE：在讀取速度時發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼
	Velocity	ARRAY[1..N] OF LREAL	群組的當前或命令速度值 注意： 雖然該參數顯示在功能區塊的輸入端，但它實際上作為輸出參數

備註:

6.8 MC_GroupStop

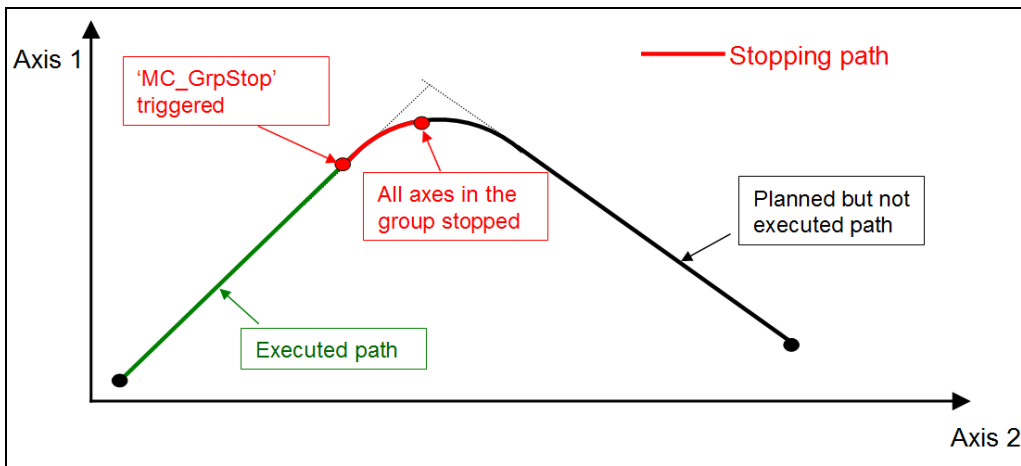
此 MFB 停止群組的所有軸，並將群組轉移到 “GroupStopping” 狀態。支援兩種類型的停止行為：在不減速的情況下突然停止或透過控制減速到零來停止。當群組處於 “GroupStopping” 狀態時，其他 MFB 無法移動任何群組軸（請參閱群組狀態圖）。一旦速度降到零， “MC_GrpStop.Done” 輸出將設置為 TRUE。只有在速度降到零之後，將 “MC_GrpStop.Execute” 輸入設置為 FALSE，才能再次移動群組軸。只要 “MC_GrpStop.Execute” 為 TRUE 或群組中的某個軸仍在移動，群組就會保持 “GroupStopping” 狀態。當 “MC_GrpStop.Done” 為 TRUE 且 “MC_GrpStop.Execute” 為 FALSE 時，群組狀態將轉換為 “GroupStandBy”。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	<ul style="list-style-type: none"> TRUE：立即停止（Hardstop=TRUE）或通過減速停止（Hardstop=FALSE）
	HardStop	BOOL	<ul style="list-style-type: none"> FALSE：在 MC_MoveXXX 指令設置的減速時間內減速至停止 TRUE：無減速停止
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE：群組軸已達到零速度
	Busy	BOOL	<ul style="list-style-type: none"> TRUE：功能區塊正在忙於減速軸組
	CommandAborted	BOOL	<ul style="list-style-type: none"> 命令因禁用群組中一個或多個軸的 MC_Power 而中止，狀態變更為 “GroupDisabled”
	Error	BOOL	<ul style="list-style-type: none"> TRUE：停止過程中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 對於 “MC_GrpStop.HardStop” = FALSE 在減速階段，群組軸保持在路徑上。因此，可能會發生群組內的單個軸可能會在停止階段加速，以保持在路徑軌跡上的情況。



- “MC_GrpStop.HardStop” = FALSE 的減速時間由目前執行的 MFB 設置（例如 “MC_MoveLinearAbsolute” , “MC_MoveLinearRelative” ）。緩衝區中第一個 MFB 的 “AccDecTime” 輸入以設定所有後續運動命令的加速和減速時間。
- 下面顯示了帶有 “MC_GroupStop” 狀態轉換的時序圖。
 - 在情況 1 中，在重置 “Execute”（FALSE）之前設置 “Done” 輸出（TRUE）。
 - 在情況 2 中，“Execute” 被重置為 FALSE，然後 “Done” 變為 TRUE。
 該群組停留在 “GroupStopping” 狀態
 - 當 “Execute” 設定為 (TRUE)
 - 在 “Execute” 已被設為 FALSE 之後，直到 “Done” 為 TRUE

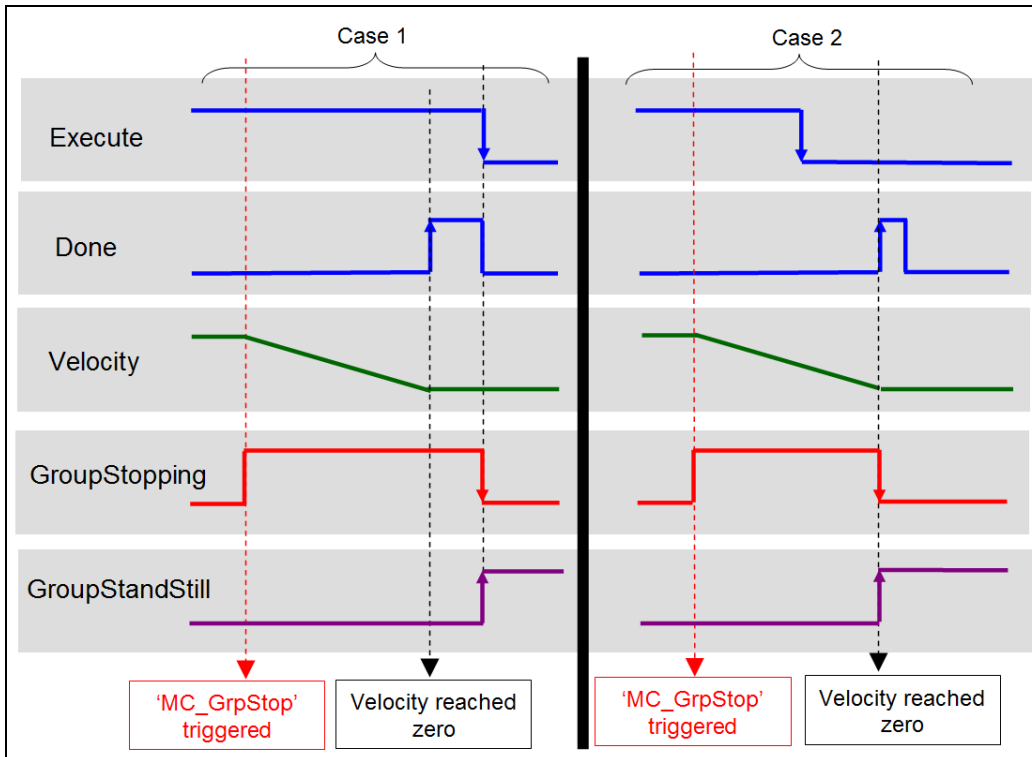


圖 6-2：MC_GroupStop 時序圖

- 以下範例顯示了與 “MC_MoveLinearRelative” 結合使用的行為。

- 線性運動中的群組軸用 “MC_GroupStop” 進行減速。群組在原始路徑上停止。
- 只要 “MC_GroupStop.Execute” 為 TRUE，群組軸會拒絕運動指令。

“MC_MoveLinearRelative” 報告一個錯誤，指示已啟用的 “MC_GroupStop” 命令。返回的錯誤是功能塊，而不是軸錯誤，這意味著狀態保持在 “GroupStopping” 中，不會更改為 “GroupErrorStop”。在第 3 個 “Exe_1” 正緣，群組開始下一個運動。

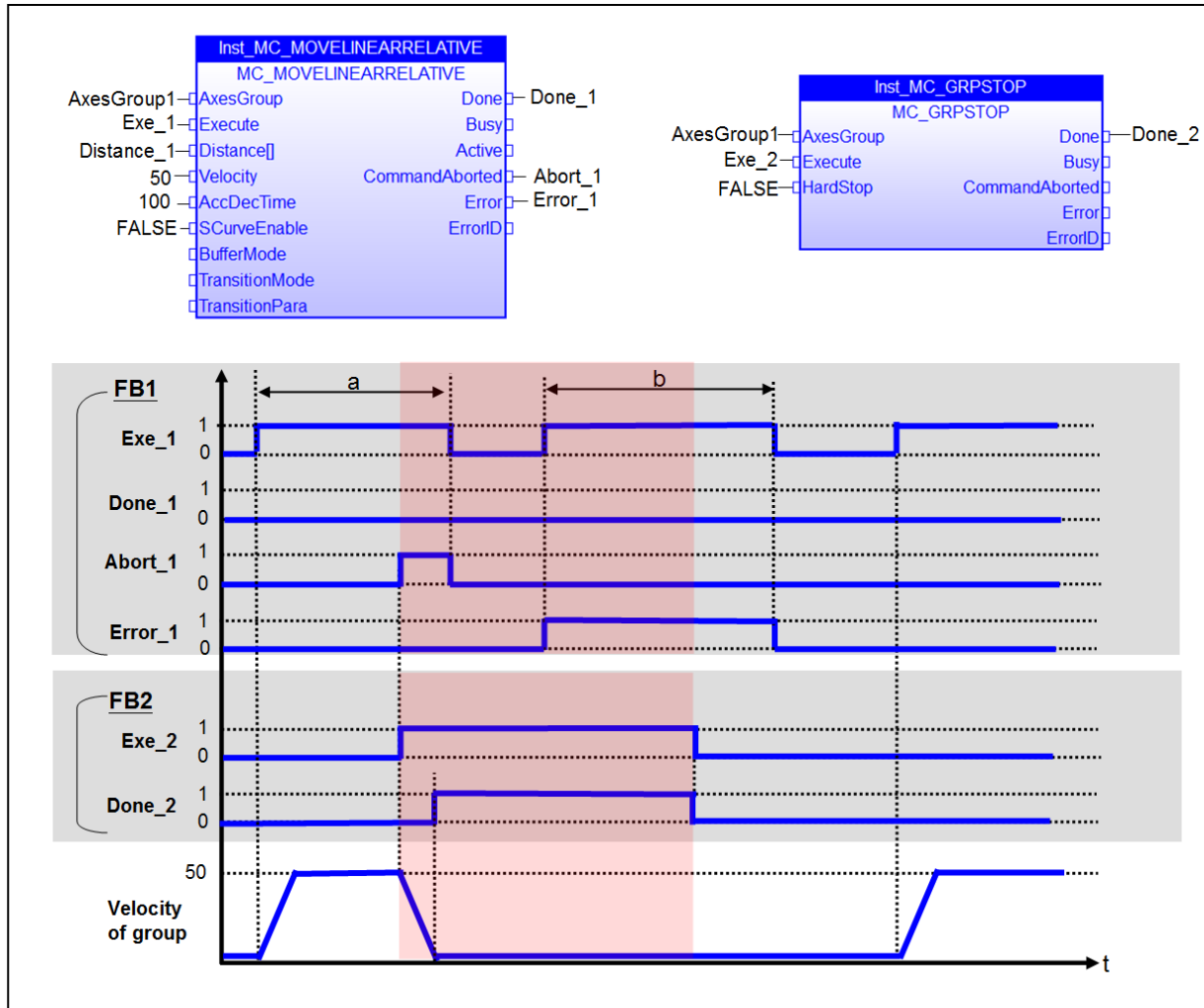
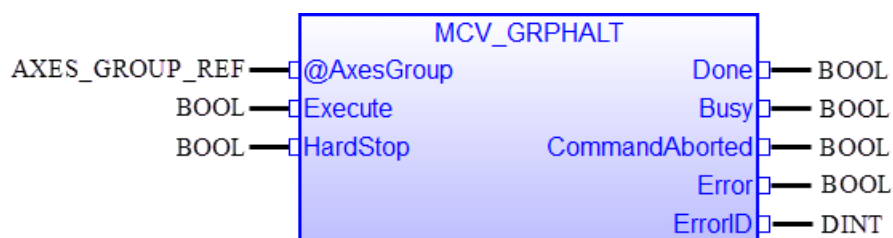


圖 6-3：“MC_GroupStop” 與 “MC_MoveLinearRelative” 結合的行為

6.9 MCV_GroupHalt

此 MFB 命令執行控制的運動停止，中止任何正在進行的運動命令，並清除任何緩衝命令。

“AxesGroup” 將轉換為狀態 “GroupStopping”，直到所有群組成員軸的速度都為零。當 “Done” 輸出為 TRUE 後，狀態將更改為 “GroupStandby”。“MC_GrpStop” 和 “MC_GrpHalt” 的區別在於，一旦速度達到零，“MC_GrpHalt” 會自動將組狀態轉換為 “GroupStandby”，即使 “Execute” 仍然是 TRUE。“MC_GrpStop” 會一直等到 “Done” 變為 TRUE，輸入 “Execute” 變為 FALSE，然後才會切換到 “GroupStandBy” 狀態。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	<ul style="list-style-type: none"> TRUE：立即停止（Hardstop=TRUE）或通過減速停止（Hardstop=FALSE）
	HardStop	BOOL	<ul style="list-style-type: none"> FALSE：在 MC_MoveXXX 指令設置的減速時間內減速至停止 TRUE：無減速停止
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE：群組軸已達到零速度
	Busy	BOOL	<ul style="list-style-type: none"> TRUE：功能區塊正在忙於減速軸組
	CommandAborted	BOOL	<ul style="list-style-type: none"> 命令因禁用群組中一個或多個軸的 MC_Power 而中止，狀態變更為 “GroupDisabled”
	Error	BOOL	<ul style="list-style-type: none"> TRUE：停止過程中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 在減速階段，群組軸保持在路徑上，由於路徑需求，群組中的單個軸可能會在此過程加速。
- “MCV_GroupHalt” 不支援緩衝模式。“MCV_GroupHalt” 將立即執行，並且緩衝區中當前的所

有運動命令都將被中止。

- 在所有軸都達到靜止狀態後，群組狀態會自動從 “GroupStopping” 切換到 “GroupStandby” 。一旦狀態切換到 “GroupStandby” ，就可以執行下一個運動命令。
- 即使 “MCV_GroupHalt.Execute” 為 TRUE，則組狀態為 “GroupStandBy”，也可以發送下一個群組運動命令。

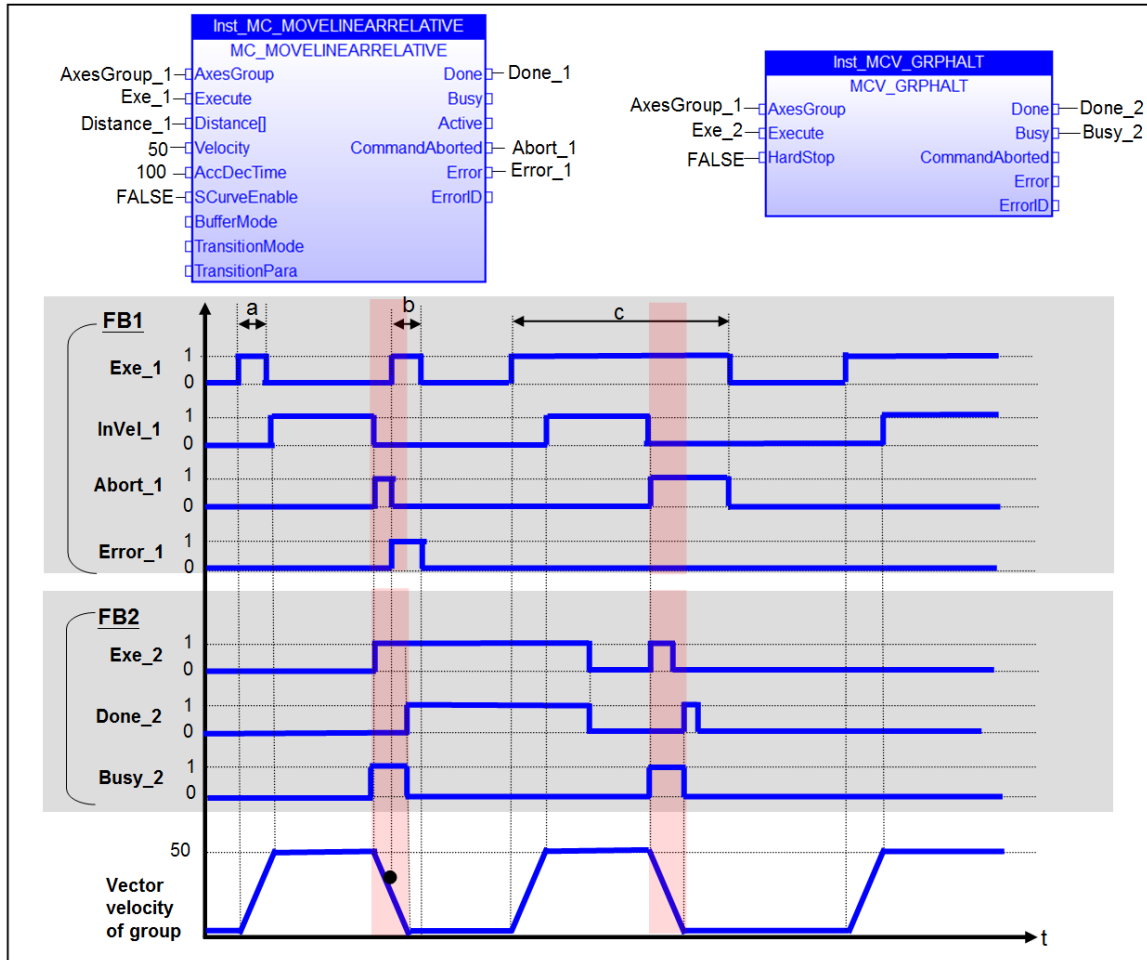


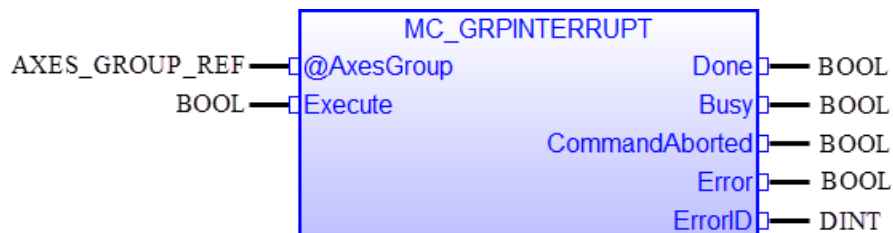
圖 6-4：“MC_GroupHalt” 與 “MC_MoveLinearRelative” 結合的行為

圖 6-4 顯示了 “MC_GrpHalt” 與 “MC_MoveLinearRelative” 結合使用的行為範例：

- 當 “MCV_GrpHalt” 命令繁忙時，無法執行其他群組運動命令。
- 一旦 “MCV_GrpHalt.Done” 至少在一個週期內為 TRUE，就可以執行新的移動命令；即使 “MCV_GrpHalt.Execute” 仍然是 TRUE。
- 與之對比，當 “MC_GroupStop.Done” = TRUE 和 “MC_GroupStop.Execute” = FALSE 時，“MC_GroupStop” 才允許執行下一個運動命令。

6.10 MC_GroupInterrupt

此 MFB 中斷運動命令，沿命令軌跡進行受控停止。在觸發 “MC_GroupContinue.Execute” 之前，不允許中斷的移動命令繼續並到達其目標位置。中斷的 MFB 命令不會被中止，並且在觸發 “MC_GroupContinue.Execute” 後繼續執行。此 MFB 不會更改群組狀態，這意味著即使在所有成員軸都已停止並設置了 “Done” 輸出為 TRUE， “AxesGroup” 狀態仍保持原始狀態。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	正緣觸發 MFB 命令
VAR_OUTPUT	Done	BOOL	TRUE：軸組已達到零速度
	Busy	BOOL	TRUE：功能區塊正在忙於減速軸組
	CommandAborted	BOOL	MFB 被另一個命令中止
	Error	BOOL	TRUE：停止過程中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 減速度由 MFB 決定，MFB 將軸從靜止加速。
- 在減速階段，群組軸保持在命令軌跡上。
- 當速度達到零時，設置 “Done” 輸出為 TRUE。
- 中斷的 MFB 命令不會被中止，因此輸出 “CommandAborted” 不會變為 TRUE。“Busy” 輸出保持為 TRUE，“Active” 設置為 FALSE。
- 該群組保持中斷狀態，直到群組速度達到零並且執行 “MC_GrpContinue”。

6.11 MC_GroupContinue

此 MFB 將群組軸的控制權轉移回已被 “MC_GroupInterrupt” 中斷的原始 MFB。



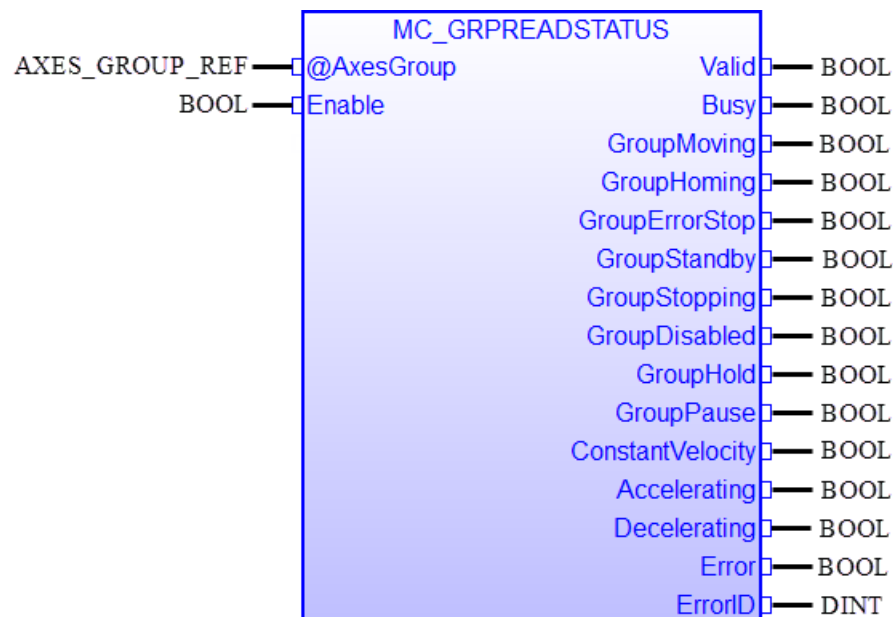
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	正緣觸發功能塊（FB）的執行
VAR_OUTPUT	Done	BOOL	TRUE：被中斷的 MFB 恢復對群組軸的控制
	Busy	BOOL	TRUE：功能區塊正在忙於將控制權轉回被中斷的 MFB
	CommandAborted	BOOL	MFB 被另一個命令中止
	Error	BOOL	TRUE：發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 一旦原始 MFB 再次完全控群組
 - “MC_GroupContinue.Done” 設置為 TRUE
 - 原始 MFB 的 “Active” 輸出變為 TRUE
- 在原始 MFB 可以收回控制權之前，群組軸速度必須為零。
- 減速度由 MFB 確定，MFB 將軸從靜止加速。
- 在減速階段，群組軸保持在指令軌跡上。
- 當速度達到零時，設置 “Done” 輸出為 TRUE。

6.12 MC_GroupReadStatus

此 MFB 返回群組的狀態。

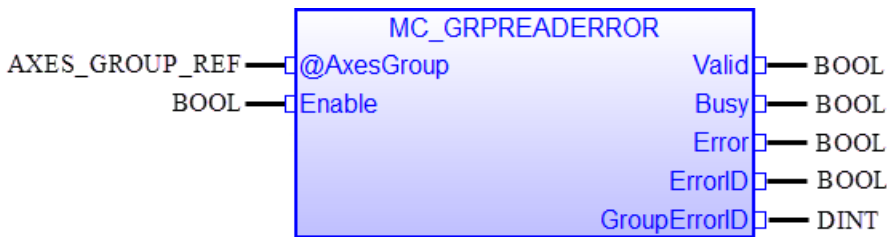


輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Enable	BOOL	TRUE：持續獲取群組的狀態
VAR_OUTPUT	Valid	BOOL	TRUE：狀態輸出有效
	Busy	BOOL	TRUE：功能區塊正在忙於獲取群組的狀態
	GroupMoving	BOOL	請參閱組別狀態圖
	GroupHoming	BOOL	請參閱組別狀態圖
	GroupErrorStop	BOOL	請參閱組別狀態圖
	GroupStandby	BOOL	請參閱組別狀態圖
	GroupStopping	BOOL	請參閱組別狀態圖
	GroupDisabled	BOOL	請參閱組別狀態圖
	GroupHold	BOOL	保留
	GroupPause	BOOL	保留
	ConstantVelocity	BOOL	沿命令路徑以恒定速度運動
	Accelerating	BOOL	群組正在沿命令路徑加速
	Decelerating	BOOL	群組正在沿命令路徑減速
	Error	BOOL	TRUE：在讀取群組狀態時發生錯誤
	ErrorID	DINT	錯誤識別

			命令特定錯誤代碼
--	--	--	----------

6.13 MC_GroupReadError

此 MFB 傳回軸組的錯誤代碼。此錯誤與功能塊錯誤輸出參數無關（例如，無效的參數化）。如果發生組錯誤，組軸將停止，一旦所有軸的速度為零，組狀態將轉換為 “GroupErrorStop”。



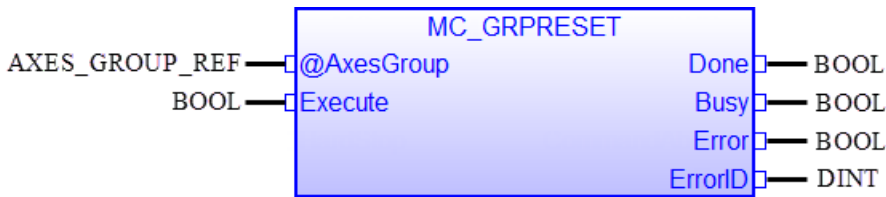
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Enable	BOOL	TRUE：持續讀取群組的 "GroupErrorID"
VAR_OUTPUT	Valid	BOOL	TRUE：輸出有效
	Busy	BOOL	TRUE：功能區塊正在忙於讀取 "GroupErrorID"
	Error	BOOL	TRUE：讀取 "GroupErrorID" 時發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼
	GroupErrorID	DINT	群組錯誤的值。請參見附錄中的錯誤表

備註:

- 如果 “GroupErrorID” 輸出為零，則群組沒有錯誤。
- 當輸出 “Valid” 為 TRUE 時，“GroupErrorID” 輸出才有效。
- 例如，群組錯誤可以是：
 - 超出軟體極限
 - 硬體極限開關已被觸發
 - 驅動器緊急狀態啟動
 - 通訊錯誤：無法與遠端伺服驅動器進行週期性數據交換；EtherCAT 主站切換到 SafeOP 模式等。

6.14 MC_GroupReset

此 MFB 重置群組中的所有軸錯誤，並在清除錯誤後從狀態 “GroupErrorStop” 轉換到 “GroupStandby”。如果錯誤無法重置（例如 EtherCAT 主站和從站之間的通訊錯誤），則群組狀態將保持 “GroupErrorStop”。



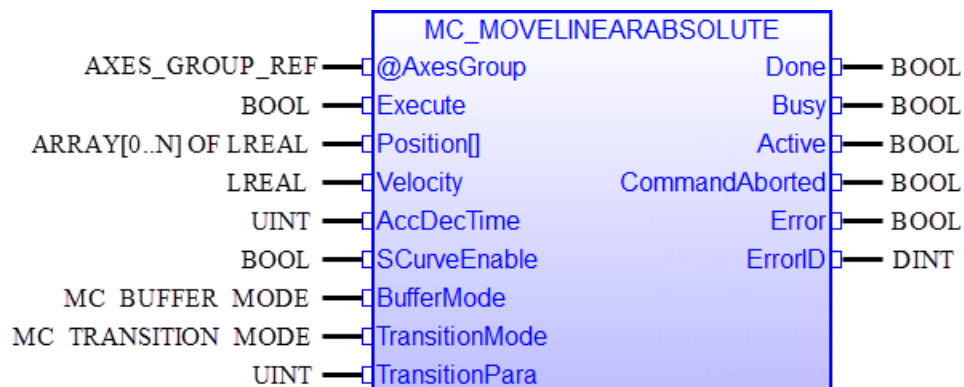
輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Enable	BOOL	TRUE：持續讀取群組的 "GroupErrorID"
VAR_OUTPUT	Done	BOOL	TRUE：群組內的所有軸已成功重置
	Busy	BOOL	TRUE：正在忙於重置群組內的所有軸
	Error	BOOL	TRUE：在重置軸過程中發生了錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 此 MFB 會重置此群組中的所有軸，類似於單軸 FB “MC_Reset” 的操作。

6.15 MC_MoveLinearAbsolute

此 MFB 指令執行線性插值運動，從當前位置移動到指定的絕對位置。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	正緣時開始運動
	Position[]	ARRAY[0..15] OF LREAL	<ul style="list-style-type: none"> 陣列 [0..15] 包含每個維度的絕對終點位置 最多支援 16 個軸 (N≤16) 如果軸組包含 4 個軸，則陣列 [0..3] 必須至少包含 4 個絕對目標位置
	Velocity	LREAL	路徑的最大速度 [u/s]，始終為正值
	AccDecTime	UINT	加速時間 [ms]：從起始速度加速至最大速度所需的時間 減速時間 [ms]：從最大速度減速至零速度所需的時間
	SCurveEnable	BOOL	加速/減速曲線類型： <ul style="list-style-type: none"> FALSE：梯形曲線 (T-curve) TRUE：S 型曲線 (S-curve)
	BufferMode	MC_BUFFER_MODE	參考緩衝模式定義
	TransitionMode	MC_TRANSITION_MODE	參考過渡模式定義
	TransitionPara	UINT	值範圍：0 到 100
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE：指令已成功到達目標位置並完成執行。
	Busy	BOOL	<ul style="list-style-type: none"> TRUE：功能塊正在等待執行或執行中。
	Active	BOOL	<ul style="list-style-type: none"> TRUE：功能塊已控制群組軸並正在執

			行操作
	CommandAborted	BOOL	MFB 被另一個命令中止
	Error	BOOL	TRUE：在移動過程中發生了錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 當此 MFB 處於活動狀態時，該群組處於 “GroupMoving” 狀態。
- 當到達絕對位置時，“Done” 輸出將設置為 TRUE，並且群組狀態更改為 “GroupStandby” 或開始執行另一個運動命令。
 - 如果沒有進一步的待處理操作，則此 MFB 以速度 0 完成。
 - 在 “mcBuffered” 模式下：當運動完成時，軸返回到 “GroupStandby” 狀態。
- 如果要移動的距離短於加速和減速所需的距離，則不會達到輸入 “Velocity”。
- 如果群組已在移動，則將忽略 “AccDecTime”。僅當群組從速度零開始（狀態 “GroupStandby”）時，才會應用 “AccDecTime”。
- 支援的緩衝模式及其與過渡模式和參數的關係如表 13 所示。速度時間圖表中的每個圓圈數字都表示一個新的群組命令。
- “TransitionPara”：設置群組的混合百分比。混合將會使從一個命令到另一個命令之間的過程更平滑，但會產生角點誤差。在 “mcBlendingNext” 模式下
 - “100” 混合百分比表示從上一個運動指令的減速開始混合下一個運動指令。
 - “0” 混合百分比表示沒有混合部分；並且行為類似於 “mcBuffered” 命令模式。
 圖表中的紅線（錯誤！書籤的自我參照不正確。2）顯示了在 “mcBlendingNext” 模式下具有不同 “TransitionPara” 設置的插值路徑：
- 當此 MFB 處於活動狀態時，組軸處於 “SynchronizedMotion” 狀態。
- 時序圖：參見 PLCopen：“用於運動控制的功能塊（第 4 部分）” 圖 15，第 57/119 頁。

例:

VAR

```

    flgAddToGroup : BOOL := TRUE ;
    flgMoveLinear : BOOL := FALSE ;
    AxesGroup: lib:AXES_GROUP_REF := [UINT#1,UINT#1];
    dfPosition : ARRAY [0 .. 5] OF LREAL := LREAL#10000,LREAL#20000,
        LREAL#30000,LREAL#40000,LREAL#50000,LREAL#60000 ;
    dfVectorVelocity : LREAL := LREAL#1000 ;
  
```

```

AccDecTime : UINT := UINT#999 ;
wTransitionPara : UINT := UINT#50 ;
Inst_MC_MoveLinearAbsolute : MC_MoveLinearAbsolute ;

MovAbsDone: BOOL ;
MovAbsBusy: BOOL ;
MovAbsActive: BOOL ;
MovAbsAborted: BOOL ;
MovAbsError: BOOL ;
MovAbsErrID: BOOL ;
END_VAR

(* Initialize the ECAT-M8000 *)
//...
(* Axis Assignment *)
//...

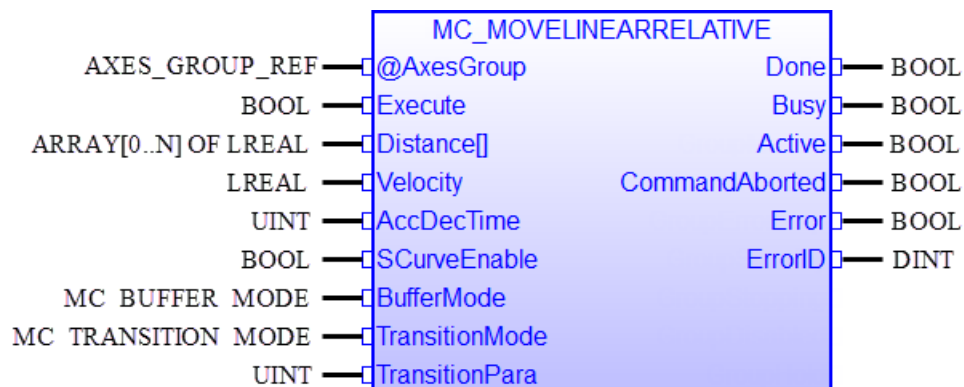
(* Execute an instance of MC_MOVELINEARABSOLUTE *)
flgMoveLinear := TRUE ;
Inst_MC_MOVELINEARABSOLUTE( AxesGroup(*lib:AXES_GROUP_REF*),
    flgMoveLinear(*BOOL*),
    dfPosition(*LREAL*),
    dfVectorVelocity(*LREAL*),
    AccDecTime(*UINT*),
    FALSE(*SCurveEnable BOOL*),
    MC_BUFFER_MODE#mcBuffered,
    MC_TRANSITION_MODE#TMStartAccNext,
    wTransitionPara(*UINT*) );

(* Get function block output status: *)
MovAbsDone := Inst_MC_MoveLinearAbsolute.Done;
MovAbsBusy := Inst_MC_MoveLinearAbsolute.Busy;
MovAbsActive := Inst_MC_MoveLinearAbsolute.Active;
MovAbsAborted := Inst_MC_MoveLinearAbsolute.CommandAborted;
MovAbsError := Inst_MC_MoveLinearAbsolute.Error;
MovAbsErrID := Inst_MC_MoveLinearAbsolute.ErrorID;

```

6.16 MC_MoveLinearRelative

此 MFB 執行線性插值運動命令，並將組軸移動到距當前位置的相對距離。參數 “Distance[]” 包含組中每個軸從當前位置到結束位置的線性路徑。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	正緣時開始運動
	Distance[]	ARRAY[0..N] OF LREAL	<ul style="list-style-type: none"> 陣列 [0..N] 包含每個維度的絕對終點位置 最多支援 16 個軸 (N<16) 如果軸組包含 4 個軸，則陣列 [0..3] 必須至少包含 4 個絕對目標位置
	Velocity	LREAL	路徑的最大速度 [u/s]，始終為正值
	AccDecTime	UINT	加速時間 [ms]：從起始速度加速至最大速度所需的時間 減速時間 [ms]：從最大速度減速至零速度所需的時間
	SCurveEnable	BOOL	加速/減速曲線類型： <ul style="list-style-type: none"> FALSE：梯形曲線 (T-curve) TRUE：S 型曲線 (S-curve)
	BufferMode	MC_BUFFER_MODE	參考緩衝模式定義
	TransitionMode	MC_TRANSITION_MODE	參考過渡模式定義
	TransitionPara	UINT	值範圍：0 到 100
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE：指令已成功到達目標位置並完成執行。
	Busy	BOOL	<ul style="list-style-type: none"> TRUE：功能塊正在等待執行或執行中。

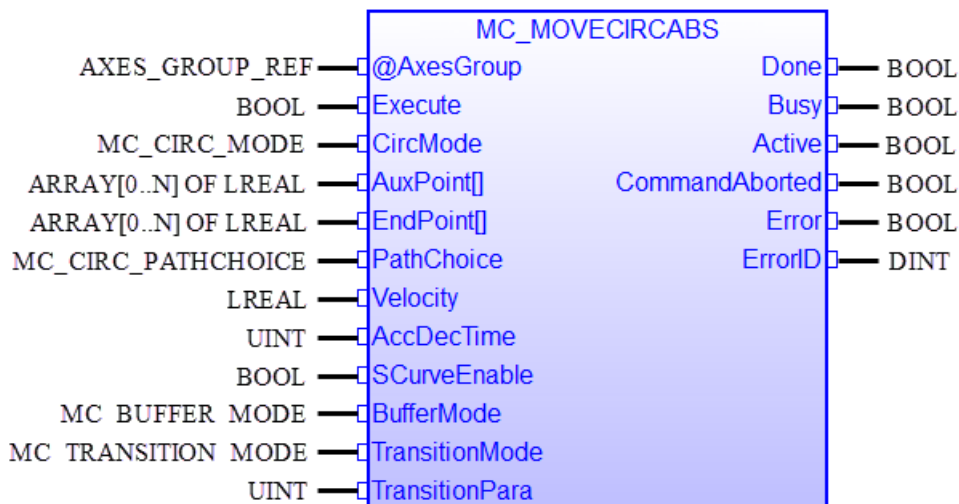
	Active	BOOL	<ul style="list-style-type: none"> TRUE：功能塊已控制群組軸並正在執行操作
	CommandAborted	BOOL	MFB 被另一個命令中止
	Error	BOOL	<ul style="list-style-type: none"> TRUE：在移動過程中發生了錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 當此 MFB 處於活動狀態時，該群組處於 “GroupMoving” 狀態。
- 距離移動完成後，如果沒有進一步的待處理運動命令，則群組將更改為 “GroupStandby” 狀態。“Done” 輸出將設置為 TRUE。
- 如果要移動的距離短於加速和減速所需的距離，則不會達到輸入 “Velocity”。
- 如果群組已在移動，則將忽略 “AccDecTime”。僅當群組從速度零開始（狀態 “GroupStandby”）時，才會應用 “AccDecTime”。
- 支援的緩衝模式及其與過渡模式和參數的關係如表 13 所示。速度時間圖表中的每個圓圈數字都表示一個新的群組命令。
- 'TransitionPara' 設置群組的混合百分比（表 11）。混合將會使從一個命令到另一個命令之間的過程更平滑；但是，它將偏離原始路徑。
- 時序圖：參見 PLCopen：“用於運動控制的功能塊（第 4 部分）” 圖 16，第 60/119 頁。

6.17 MC_MoveCircularAbsolute

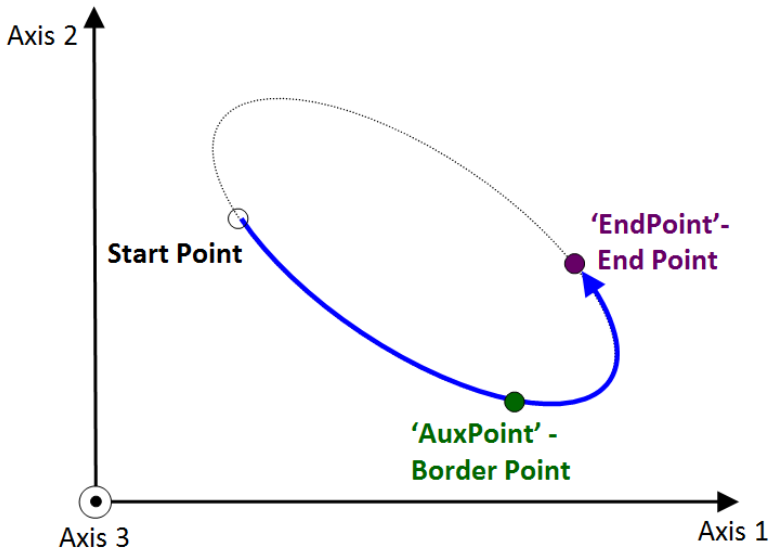
此 MFB 命令從當前位置開始進行插值圓周運動。終點和輔助點都定義為座標系中的絕對位置。

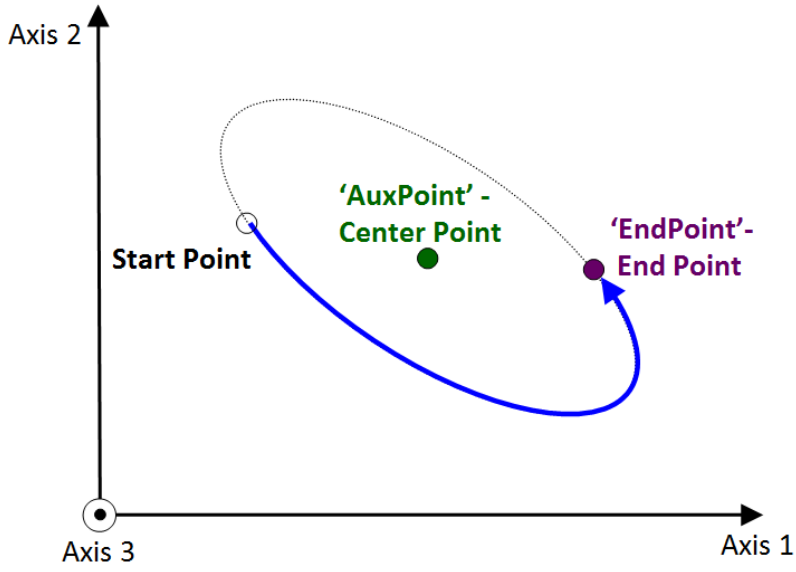


輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	正緣時開始運動
	CircMode	MC_CIRC_MODE	這個列舉類型具有以下元素： <ul style="list-style-type: none"> BORDER、CENTER、RADIUS、ANGLE 它們決定了“AuxPoint”輸入參數的用途
	AuxPoint[]	ARRAY[0..N] OF LREAL	<ul style="list-style-type: none"> 參數行為取決於“CircMode”（見表 14） 數組 [0..2] 包含絕對位置 最多支持 3 軸的軸組（N<3） 如果軸組有三個軸，則數組 [0..2] 必須至少包含三個絕對位置（每個軸一個）
	EndPoint[]	ARRAY[0..N] OF LREAL	<ul style="list-style-type: none"> 參數行為取決於“CircMode”（見表 14）。 數組 [0..3] 包含結束位置 最多支持 3 軸的軸組
	PathChoice	MC_CIRC_PATHCHOICE	圓形的方向： <ul style="list-style-type: none"> 順時針或逆時針
	Velocity	LREAL	路徑的最大速度 [u/s]。始終為正數
	AccDecTime	UINT	<ul style="list-style-type: none"> 加速和減速時間。兩個參數設置為相同的值

			<ul style="list-style-type: none"> 加速時間是達到最大速度所需的時間 減速時間是從最大速度減速到零所需的時間
	SCurveEnable	BOOL	加速/減速曲線 <ul style="list-style-type: none"> FALSE: 梯形曲線 (T-曲線) TRUE: S-曲線
	BufferMode	MC_BUFFER_MODE	參考緩衝模式定義
	TransitionMode	MC_TRANSITION_MODE	參考過渡模式定義
	TransitionPara	UINT	值範圍：0 到 100
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE：指令已成功到達目標位置並完成執行。
	Busy	BOOL	<ul style="list-style-type: none"> TRUE：功能塊正在等待執行或執行中。
	Active	BOOL	<ul style="list-style-type: none"> TRUE：功能塊已控制群組軸並正在執行操作
	CommandAborted	BOOL	MFB 被另一個命令中止
	Error	BOOL	<ul style="list-style-type: none"> TRUE：在移動過程中發生了錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

No.	MC_CIRC_MODE	描述
0	BORDER	<p>在圓上定義一個點，該點是從起點到終點的路徑中經過。</p> <ul style="list-style-type: none"> “AuxPoint” (輔助點) : <ul style="list-style-type: none"> 圓上的絕對位置點 (數組 [0..2]) : <ul style="list-style-type: none"> 對於有兩個軸的群組，索引 0 和 1 表示 2D 圓邊界上的絕對點。 對於有三個軸的群組，索引 0、1 和 2 表示 3D 圓邊界上的絕對點。 支援的最大軸數為 3 個軸 (N<3)。 “EndPoint” (終點) : <ul style="list-style-type: none"> 定義圓弦的絕對終點 (數組 [0..2]) : <ul style="list-style-type: none"> 對於有兩個軸的群組，索引 0 和 1 表示 2D 圓的絕對終點。 對於有三個軸的群組，索引 0、1 和 2 表示 3D 圓的絕對終點。 如果無法計算同時包含 “AuxPoint” 和 “EndPoint” 的圓形路徑，MFB 將輸出錯誤。

		
1	CENTER	<p>定義圓的中心點和結束位置。中心點決定了連接起點位置和終點位置的圓弧半徑。</p> <ul style="list-style-type: none"> • “AuxPoint”（輔助點）： <ul style="list-style-type: none"> - 圓心的絕對位置（陣列 [0..2]）： <ul style="list-style-type: none"> ▪ 對於具有兩個軸的群組，索引 0 和 1 表示 2D 圓的絕對中心點。 ▪ 對於具有三個軸的群組，索引 0、1 和 2 表示 3D 圓的絕對中心點。 - 支援的群組最多包含 3 個軸（$N < 3$）。 • “EndPoint”（終點）： <ul style="list-style-type: none"> - 定義圓弦的絕對終點位置（陣列 [0..2]）： <ul style="list-style-type: none"> ▪ 對於具有兩個軸的群組，索引 0 和 1 表示 2D 圓的絕對終點位置。 ▪ 對於具有三個軸的群組，索引 0、1 和 2 表示 3D 圓的絕對終點位置。 - 如果無法使用中心點 “AuxPoint” 和終點 “EndPoint” 計算圓弧路徑，MFB 將輸出錯誤。

		
2	RADIUS	不支援
3	ANGLE	<p>定義圓弧區域的中心點、角度和法向量（適用於 3D）。</p> <p>終點由 MFB 在內部計算。</p> <ul style="list-style-type: none"> • “AuxPoint”（輔助點）： <ul style="list-style-type: none"> - 圓心的絕對位置（陣列 [0..2]）： <ul style="list-style-type: none"> ▪ 對於具有兩個軸的群組，索引 0 和 1 表示 2D 圓的絕對中心點。 ▪ 對於具有三個軸的群組，索引 0、1 和 2 表示 3D 圓的絕對中心點。 - 支援的群組最多包含 3 個軸（$N < 3$）。 • “EndPoint”（終點）： <ul style="list-style-type: none"> - 定義法向量和角度（陣列 [0..3]）。 <ul style="list-style-type: none"> ▪ 法向量：索引 0 到 2 定義了 3D 圓的法向量。如果群組有三個軸需要定義法向量，否則陣列索引 0 到 2 的值將被忽略。 ▪ 角度：陣列的索引 3 設定起點和終點之間的角度（右手法則）（單位：度）。 - 如果無法根據中心點 “AuxPoint” 和角度 “EndPoint[3]” 計算圓弧路徑，MFB 將輸出錯誤。

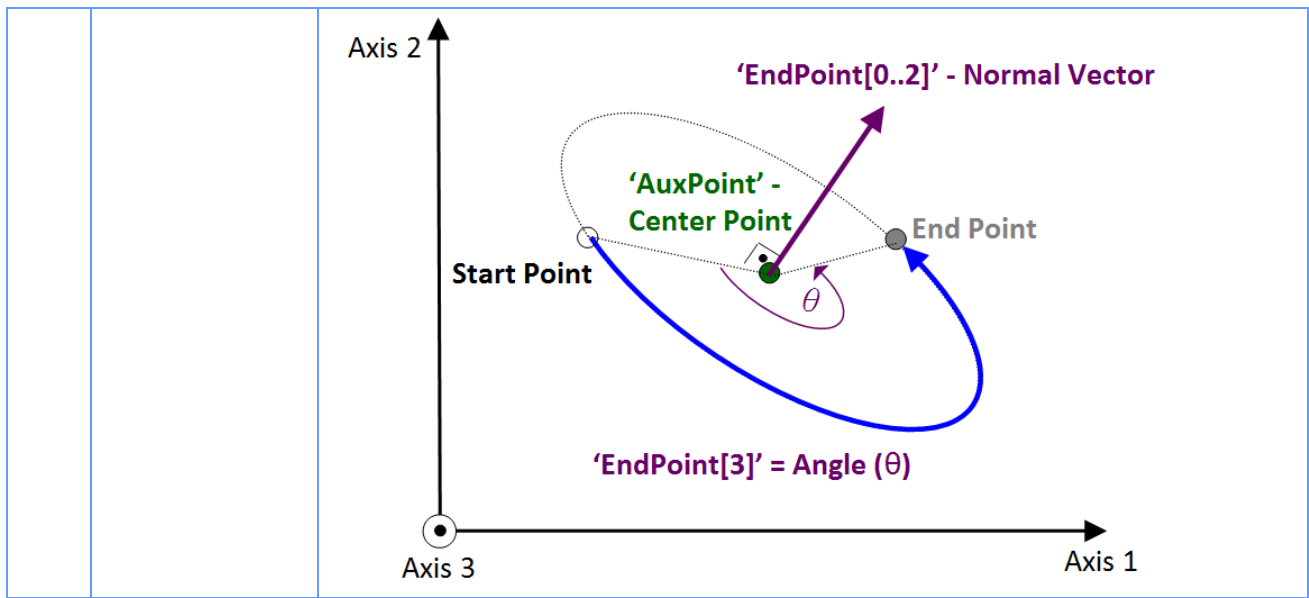
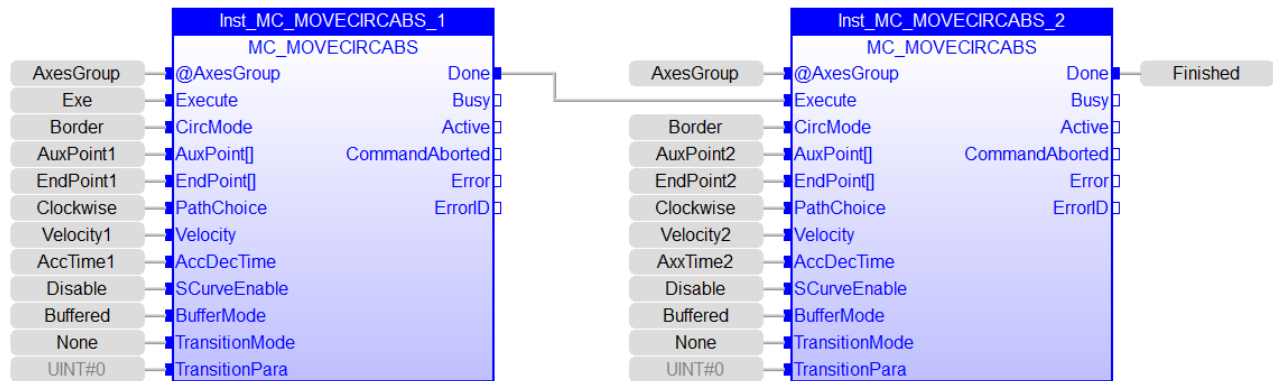


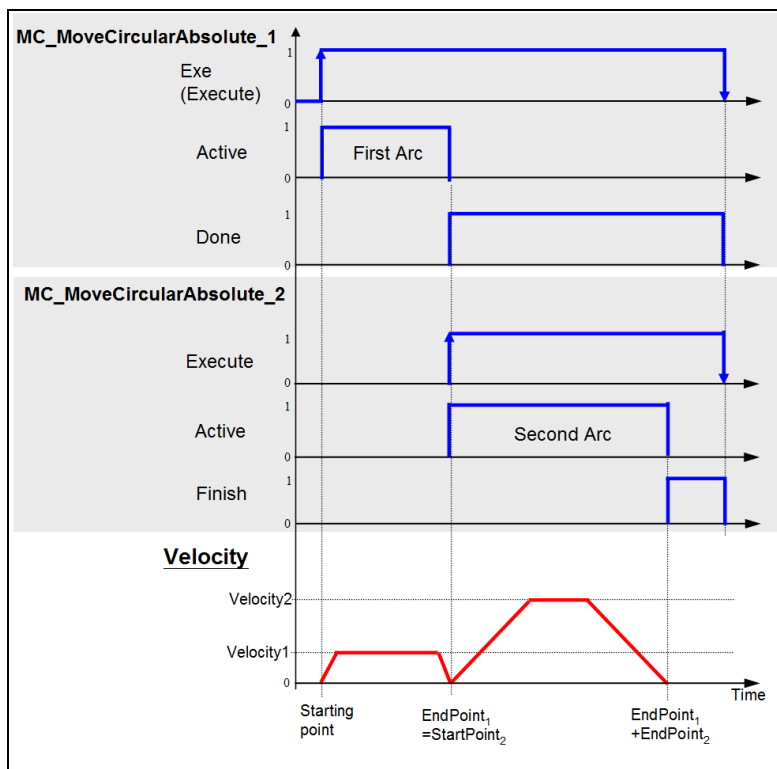
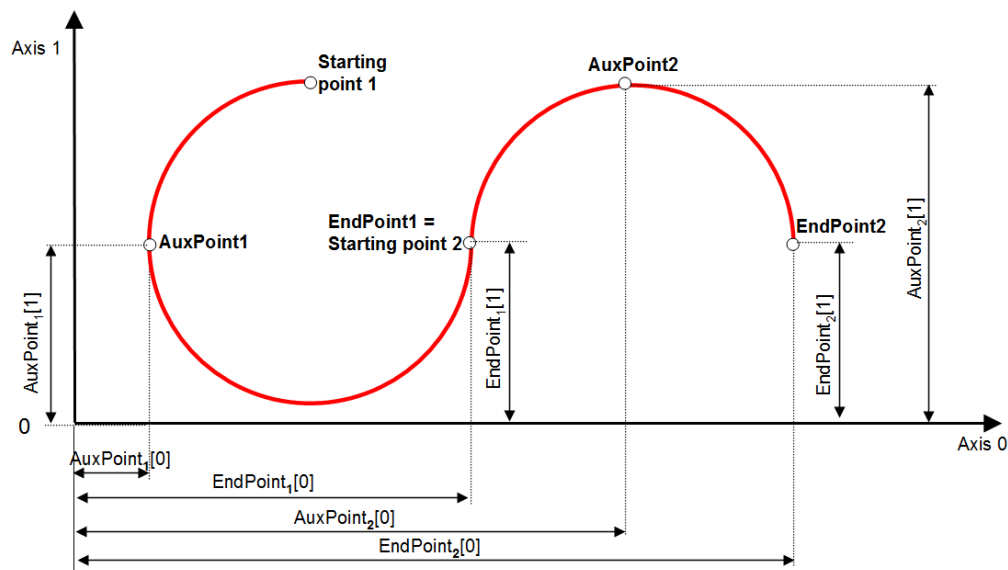
表 14：“MC_MoveCircularAbsolute”的圓形模式

例：

範例 1：

兩個“MC_MoveCircularAbsolute” MFB 的序列範例：





範例 2 :

帶有 “MC_MoveLinear” 和 “MC_MoveCircular” 的路徑運動：

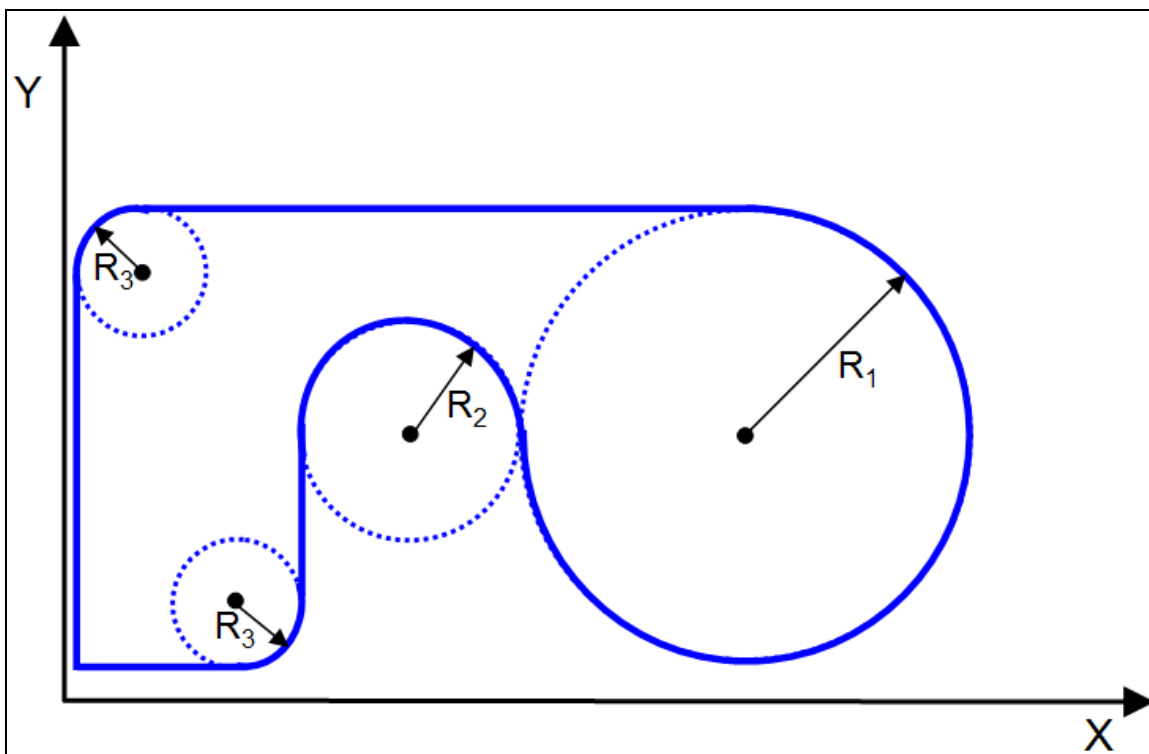
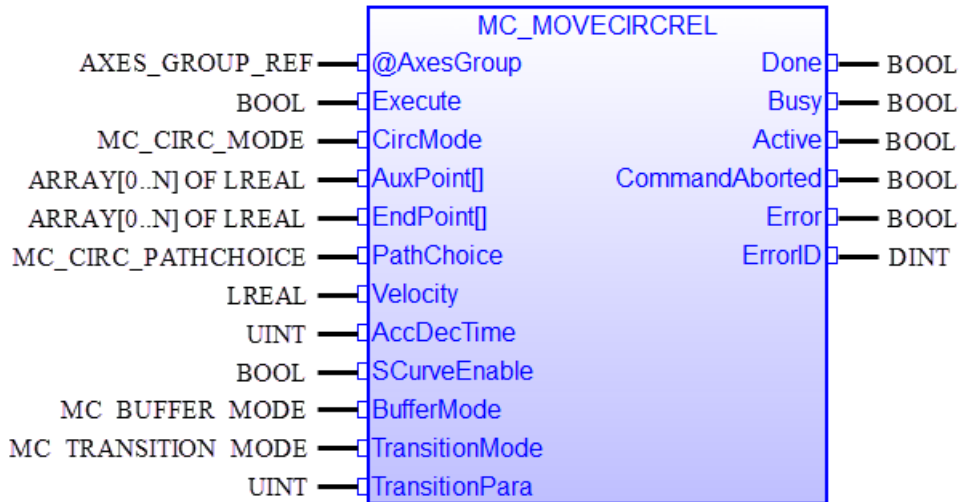


圖 6-5：使用單一命令的路徑運動（ MC_MoveLinear、MC_MoveCircular ）

6.18 MC_MoveCircularRelative

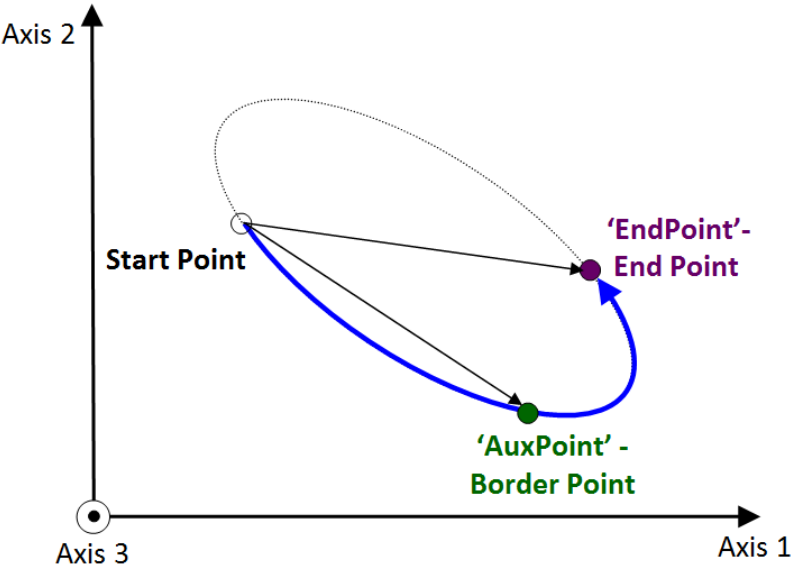
此 MFB 從實際位置開始執行圓周運動。為了讓 MFB 計算出圓周路徑，除了終點外，還必須指定輔助位置。終點和輔助點均定義為笛卡爾座標系中與起點的相對位置。

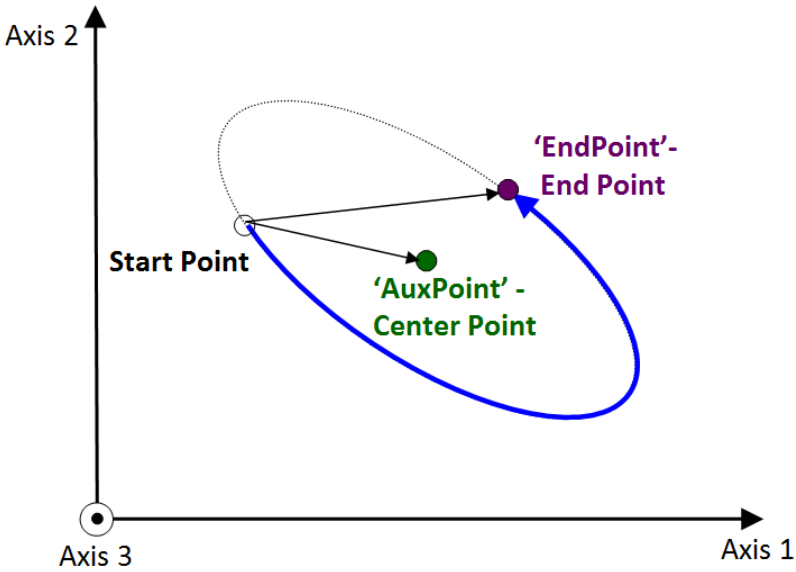


輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	在正緣開始運動
	CircMode	MC_CIRC_MODE	這個列舉類型具有以下元素： <ul style="list-style-type: none"> BORDER、CENTER、RADIUS、ANGLE 它們決定了“AuxPoint”輸入參數的用途
	AuxPoint[]	ARRAY[0..N] OF LREAL	<ul style="list-style-type: none"> 參數行為取決於“CircMode”（請參見表 15） 相對位置的陣列 [0..2] 每個群組最多支援 3 個軸（$N < 3$） 如果群組包含三個軸，則陣列 [0..2] 必須包含三個相對位置（每個軸一個）
	EndPoint[]	ARRAY[0..N] OF LREAL	<ul style="list-style-type: none"> 參數行為取決於“CircMode”（請參見表 15） 相對終點位置的陣列 [0..3] 每個群組最多支援 3 個軸
	PathChoice	MC_CIRC_PATHCHOICE	圓形的方向： <ul style="list-style-type: none"> 順時針或逆時針
	Velocity	LREAL	路徑的最大速度 [u/s]。始終為正數

	AccDecTime	UINT	<ul style="list-style-type: none"> 加速和減速時間。兩個參數設置為相同的值 加速時間是達到最大速度所需的時間 減速時間是從最大速度減速到零所需的時間
	SCurveEnable	BOOL	加速/減速曲線 <ul style="list-style-type: none"> FALSE: 梯形曲線 (T-曲線) TRUE: S-曲線
	BufferMode	MC_BUFFER_MODE	參考緩衝模式定義
	TransitionMode	MC_TRANSITION_MODE	參考過渡模式定義
	TransitionPara	UINT	值範圍：0 到 100
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE：指令已成功到達目標位置並完成執行。
	Busy	BOOL	<ul style="list-style-type: none"> TRUE：功能塊正在等待執行或執行中。
	Active	BOOL	<ul style="list-style-type: none"> TRUE：功能塊已控制群組軸並正在執行操作
	CommandAborted	BOOL	MFB 被另一個命令中止
	Error	BOOL	<ul style="list-style-type: none"> TRUE：在移動過程中發生了錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

No.	MC_CIRC_MODE	描述
0	BORDER	<p>定義圓上的中間點（邊界點），該中間點在從起點到終點的路徑上。</p> <p>FB 在內部使用起點、中間點和終點計算圓形路徑。只有小於 360° 的圓形路徑才有效。</p> <p>中間點（“AuxPoint”）和終點（“EndPoint”）都是相對於起點定義的。</p> <ul style="list-style-type: none"> “AuxPoint”（輔助點）： <ul style="list-style-type: none"> 指定圓形路徑上中間點（陣列[0..2]）的相對位置，通過該位置接近終點。它是相對於起點指定的。 <ul style="list-style-type: none"> 對於具有兩個軸的群組，索引 0 和 1 表示 2D 圓形路徑上該點的相對位置。每個索引表示一個軸的相對位置。 對於具有三個軸的群組，索引 0、1 和 2 表示 3D 圓形路徑上該點的相對位置。 支援的群組最多包含 3 個軸（$N < 3$）。 中間位置必須恰好位於圓形路徑上，不能進行近似。 “EndPoint”（終點）： <ul style="list-style-type: none"> 定義圓弦終點（陣列 [0..2]）的相對位置。該位置相對於起點來指定。

		<ul style="list-style-type: none"> 對於具有兩個軸的群組，索引 0 和 1 表示 2D 圖形的相對終點位置。 對於具有三個軸的群組，索引 0、1 和 2 表示 3D 圖形的相對終點位置。 <p>- 如果無法計算 “AuxPoint” 和 “EndPoint” 點所在的圓形路徑，則 MFB 輸出錯誤。</p> 
1	CENTER	<p>定義圓的中心點和終點位置。中心點決定了連接起點和終點的弧的半徑。中心點 (“AuxPoint”) 和終點 (“EndPoint”) 都相對於起點來定義。</p> <ul style="list-style-type: none"> “AuxPoint” (輔助點) : <ul style="list-style-type: none"> 圓心點的相對位置 (陣列 [0..2]) 。它是相對於起點指定的。 <ul style="list-style-type: none"> 對於具有兩個軸的群組，索引 0 和 1 表示 2D 圓形路徑上該點的相對位置。每個索引表示一個軸的相對位置。 對於具有三個軸的群組，索引 0、1 和 2 表示 3D 圓形路徑上該點的相對位置。 支援的群組最多包含 3 個軸 ($N < 3$) “EndPoint” (終點) : <ul style="list-style-type: none"> 定義圓弦的絕對終點位置 (陣列 [0..2]) 。 <ul style="list-style-type: none"> 對於具有兩個軸的群組，索引 0 和 1 表示 2D 圓形的相對終點位置。 對於具有三個軸的群組，索引 0、1 和 2 表示 3D 圓形的相對終點位置。 如果無法計算包含相對中心點 “AuxPoint” 和終點 “EndPoint” 的圓形路徑，MFB 輸出錯誤。

		
2	RADIUS	不支援
3	ANGLE	<p>定義圓弧區域的圓心、角度和法向量（適用於 3D）。終點由 FB 內部計算。 中心點（“AuxPoint”）是相對於起點定義的。</p> <ul style="list-style-type: none"> • “AuxPoint”（輔助點）： <ul style="list-style-type: none"> - 圓心點的相對位置（陣列 [0..2]）。它是相對於起點指定的。 <ul style="list-style-type: none"> ▪ 對於具有兩個軸的群組，索引 0 和 1 表示 2D 圓形路徑上該點的相對位置。每個索引表示一個軸的相對位置。 ▪ 對於具有三個軸的群組，索引 0、1 和 2 表示 3D 圓形路徑上該點的相對位置。 - 支援的群組最多包含 3 個軸（$N < 3$） • “EndPoint”（終點）： <ul style="list-style-type: none"> - 定義法向量和角度（陣列 [0..3]）。 <ul style="list-style-type: none"> ▪ 法向量：索引 0 到 2 定義了 3D 圓的法向量。如果群組有三個軸需要定義法向量，否則陣列索引 0 到 2 的值將被忽略。 ▪ 角度：陣列的索引 3 設定起點和終點之間的角度（右手法則）（單位：度）。 - 如果無法根據中心點 “AuxPoint” 和角度 “EndPoint[3]” 計算圓弧路徑，MFB 將輸出錯誤。

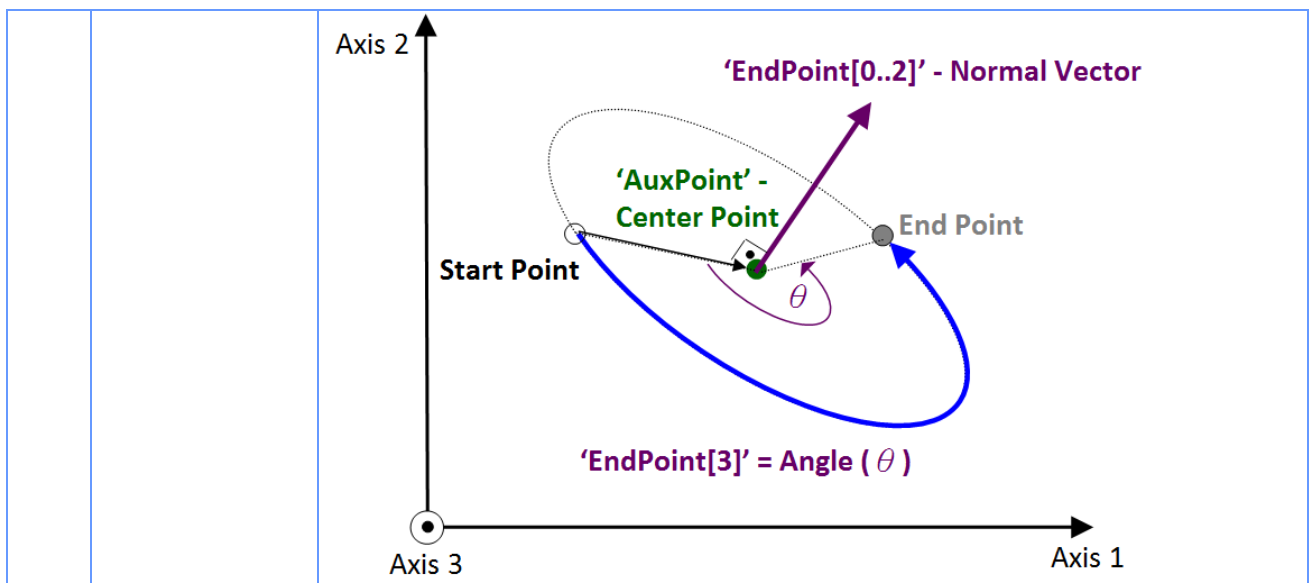


表 15：“MC_MoveCircularAbsolute”的圓形模式

6.19 MCV_GroupMoveIncPath

此 MFB 命令群組軸根據 “PathID” 中指定的增量路徑運動。“PathID” 是指 ECAT-M801/e-M9000 內部的 CSV 檔案的控制碼，該文件存儲了每個群組軸的絕對距離序列。在每個間隔中，MFB 從 CSV 檔案中讀取下一個絕對位置，並執行從當前位置到清單的下一個絕對位置的增量插值線性運動。所有軸在間隔開始時同步開始運動，並在間隔結束時一起到達絕對位置。輸入參數 “Interval” 決定間隔時間，從而決定軸移動到下一個絕對位置的速度。群組中最多支援 16 個軸。



輸入/輸出	名稱	類型	註解
VAR_IN_OUT	AxesGroup	AXIS_GROUP_REF	參考群組軸
VAR_INPUT	Execute	BOOL	在正緣開始運動
	PathID	UINT	存儲於 ECAT-M801/e-M901 設備中的 CSV 檔案的控制碼 該控制數據通過功能塊 (FB) 'MCV_IncPathLoadFileCsv' 的輸入參數 'PathID' 來設置
	Interval	UINT	在 CSV 檔案中的增量位置命令執行的間隔時間
VAR_OUTPUT	Done	BOOL	CSV 檔案中存儲的所有目標位置命令已執行完畢
	Busy	BOOL	<ul style="list-style-type: none"> TRUE：功能塊 (FB) 忙於等待執行或正在執行中
	Active	BOOL	<ul style="list-style-type: none"> TRUE：功能塊已控制群組，並且正在執行中
	CommandAborted	BOOL	命令已被另一個命令中止
	Error	BOOL	<ul style="list-style-type: none"> TRUE：在執行過程中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

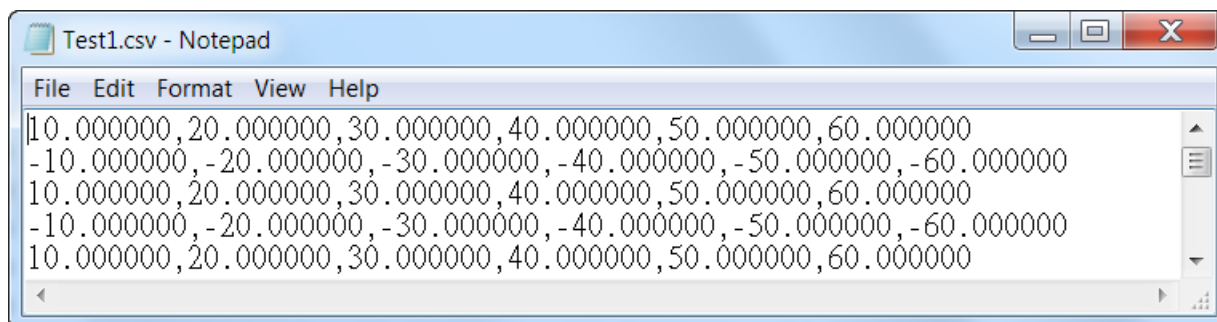
- “Interval” 設定執行 CSV 檔案中增量絕對命令列表的速度。在每個間隔結束時，所有軸同時

到達 CSV 檔案設置的增量絕對位置。

- 間隔是指 EtherCAT 主站週期。這意味著如果間隔設置為 1，則在每個主週期中，將從 csv 檔中讀取並執行新的增量命令。
- 例如：
 - 1：每個週期時間讀取運動的位置陣列數據（預設設定）
 - 2：每兩個週期時間讀取運動的位置陣列數據。

6.19.1 Create Incremental CSV File

CSV 檔包含群組中每個軸的絕對位置列表。每一行代表該組在某個週期內的絕對目標坐標。運動引擎將從 CSV 檔案的首行開始，並在每個迴圈中按順序讀取一行的數據，以獲取該組的絕對目標位置。每行的第一個值表示組中第一個軸的絕對目標位置；第二個值表示第二個軸的位置，以此類推。組的絕對位置用逗號分隔，但行中的值以換行符結尾。



CSV 檔可以由使用者手動創建，也可以通過在 PLC 程式中呼叫以下函數方式創建“MCV_CsvFile_Open”、“MCV_CsvFile_AddPoint”、“MCV_CsvFile_Close”（圖 6-6）。

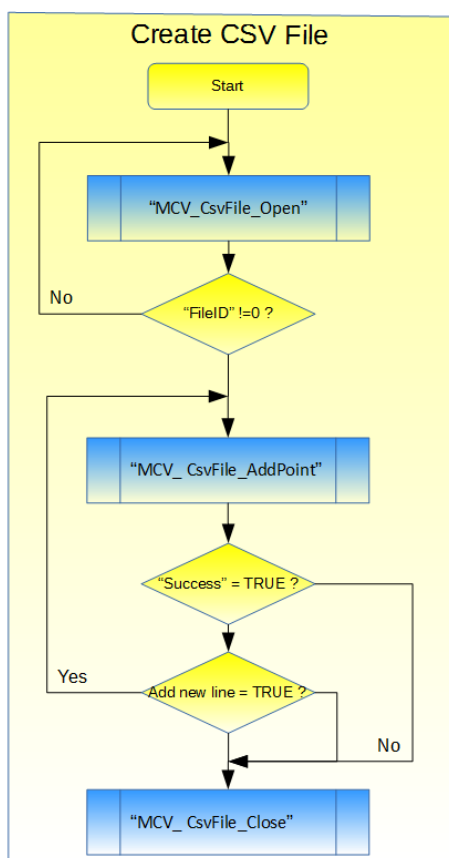
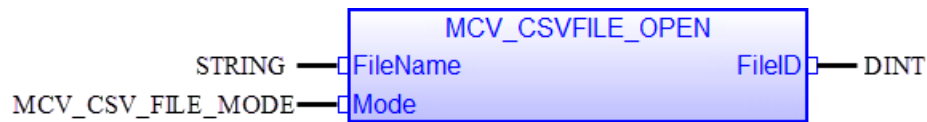


圖 6-6：創建和填充用於增量運動執行的 CSV 檔

6.19.1.1 MCV_CsvFile_Open

呼叫此函數可以打開絕對位置數據序列 CSV 檔案。如果該檔案不存在，將創建一個具有該名稱的新檔案，並將其添加到目錄“./Motion/CSV”中。



輸入/輸出	名稱	類型	註解
VAR_INPUT	File	STRING	<ul style="list-style-type: none">檔案名稱不包含目錄。請勿在檔案名稱中添加任何目錄名稱。檔案將保存到目錄：./Motion/CSV
	Mode	MC_CSV_FILE_MOD	設置檔案保存模式
VAR_OUTPUT	FileID	DINT	<ul style="list-style-type: none">0：檔案打開失敗非 0：檔案標識號

備註:

- 列舉“MCV_CSV_FILE_MODE”定義如下：
 - “fRead”：
 - 讀取模式。如果檔案不存在或無法找到，則函數呼叫將失敗。此模式無法寫入數據。
 - “fClearWrite”：
 - 寫入模式。打開一個空檔案進行寫入。如果檔案已經存在，則清除其內容。
 - “fWriteAppend”：
 - 在檔末尾打開以進行寫入（附加模式），在將新數據寫入檔案之前，不刪除檔案結束（EOF）標記。如果檔案不存在，則創建檔案。
 - “fReadWrite”：
 - 打開用於讀取和寫入模式。檔案必須已存在。
 - “fClearReadWrite”：
 - 讀取和寫入模式，打開一個空檔案進行讀取和寫入。如果檔案存在，則清除其內容。

6.19.1.2 MCV_CsvFile_AddPoint

此函數將新群組目標位置添加到 CSV 檔案中。添加到檔案中的每個新位置必須呼叫此函數。

輸入陣列 “Point[]” 的每個元素都必須包含群組中每個軸的下一個絕對目標位置的值：第一個陣列元素 [0] 的值設置組中第一個軸的絕對目標位置，第二個元素 [1] 設置第二個軸的絕對目標位置，以此類推。請務必確保陣列包含的元素數量與群組中的軸數完全相同。一個群組中最多支援 16 個軸。



輸入/輸出	名稱	類型	註解
VAR_INPUT	FileID	DINT	由 “MCV_CsvFile_Open” 返回的檔案標識號
	Point[]	ARRAY[0..N] OF LREAL	<ul style="list-style-type: none">絕對位置陣列每個位置代表該群組軸的下一個絕對目標位置陣列的元素數量不得多於或少於該群組的軸數量每個群組最多可包含 16 個軸，因此陣列長度不得超過 16 個元素
VAR_OUTPUT	Success	BOOL	<ul style="list-style-type: none">TRUE：陣列數據已成功保存到檔案中FALSE：保存陣列數據失敗

6.19.1.3 MCV_CsvFile_Close

此函數在將所有軸位置完畢後關閉 CSV 檔案。需要先關閉該檔案，然後才能被其他功能塊讀取。



輸入/輸出	名稱	類型	註解
VAR_INPUT	FileID	DINT	由 “MCV_CsvFile_Open” 返回的檔案標識號
VAR_OUTPUT	Success	BOOL	<ul style="list-style-type: none">TRUE: 檔案已成功關閉FALSE: 檔案關閉失敗

例:

```
VAR
    flgOpen : BOOL := FALSE ;
```

```

FileMode : MCV_CSV_FILE_MODE := fClearWrite ;
FileID : DINT ;
flgRet : BOOL ;
dfPoint : ARRAY [0 .. 5] OF LREAL ;
dfVal : LREAL := LREAL#0 ;

```

```
END_VAR
```

```

//=====
// Create a CSV file and fill it with incremental absolute
// position data
//=====
IF flgOpen = TRUE then

    //-----
    //Create or open a CSV file in the directory ".\Motion\CSV\":
    //-----
    FileID := MCV_CSVFILE_OPEN('Test1.CSV'(*STRING*), FileMode );

    //-----
    //Write the following data to the file
    //-----
    for i:=1 to 100 do
        if dfVal>LREAL#0.0 then
            dfVal := LREAL#-10.0;
        else
            dfVal := LREAL#10.0;
        end_if;

        for j:=0 to 5 do
            dfPoint[j] := dfVal * ANY_TO_LREAL(j+1);
        end_for;

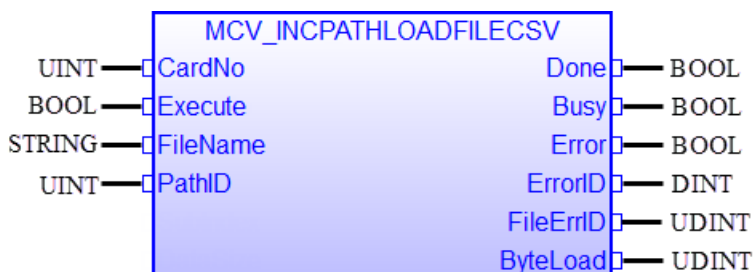
        Ret := MCV_CSVFILE_ADDPOINT( FileID(*DINT*), dfPoint(*LREAL*) );
    end_for;

    //-----
    //Close the file:
    //-----
    flgRet := MCV_CSVFILE_CLOSE( FileID(*DINT*) );
    flgOpen := FALSE;
end_if;

```

6.19.2 Download CSV File

MFB “MCV_IncPathLoadFileCsv” 將 CSV 檔案下載到 ECAT-M801/e-M901 中，並將其存儲在非易失性記憶體中。根據 CSV 檔案大小，可能需要多個任務週期才能下載所有數據。



輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	<ul style="list-style-type: none"> ECAT-M801/e-M901 卡號 需要通過 ECAT-M801/e-M901 上的 DIP 開關設置卡號
	Execute	BOOL	啟動 CSV 檔案下載，正緣觸發 CSV 檔案的下載
	FileName	STRING	CSV 檔案名稱，位於 ./Motion/CSV 目錄中的檔案名稱
	PathID	UINT	<ul style="list-style-type: none"> 檔案控制碼： 此數字表示已下載檔案的控制碼。任何非零的 UNIT 值都可以分配給 ECAT-M801/e-M901 中的 CSV 檔案 用於運動執行的命令： 命令 “MC_GroupMoveIncPath” 需要此控制碼以確定運動執行時使用的 CSV 檔案
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE：檔案已成功下載
	Busy	BOOL	<ul style="list-style-type: none"> TRUE：功能塊 (FB) 仍在忙於下載檔案
	Error	BOOL	<ul style="list-style-type: none"> TRUE：下載過程中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼
	FileErrID	UDINT	<ul style="list-style-type: none"> 檔案錯誤編號：由操作系統返回的特殊檔案錯誤，錯誤 ID 由操作系統或 “stdio.h” 庫定義。
	ByteLoad	UDINT	已下載的字節數：此值應與檔案大小（以字節為單位）相同

7 ECAT-M801/e-M901 相關的 Function Blocks

本章中介紹的 FB 是用於特定設備的。它們的主要目的是初始化主機 (PC 或 PAC) 和 ECAT-M801/e-M901 卡之間的通訊，初始化 EtherCAT 網路並將 EtherCAT 主站設置為操作模式。此外，必須先完成從站到軸的分配，然後 PLCopen 定義的 FB 才能控制和讀取任何軸。

7.1 EM_InitMaster

此 FB 建立主機 (PC 或 PAC) 與 ECAT-M801/e-M901 卡之間的通信；通過確保配置檔案中列出的從站 (由 EtherCAT Utility 創建) 在 EtherCAT 網路中處於在線狀態，檢查 EtherCAT 網路設置是否正確；並將 EtherCAT 主站設置為操作模式 (OP)。



輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	<ul style="list-style-type: none">ECAT-M801/e-M901 卡號卡號需要通過 ECAT-M801/e-M901 卡上的 DIP 開關設置
	Execute	BOOL	在正緣將主控設置為 OP 模式
	NetworkInfo	UINT	網絡信息檔案號碼 (由 EtherCAT 工具配置)
	CycleTime	EM_CYCLE_TIME	週期時間 (表 16)
	WcErrCnt	UDINT	工作計數器錯誤次數
VAR_OUTPUT	Done	BOOL	TRUE：主控處於 OP 模式 FALSE：初始化 EtherCAT 網絡失敗，主控不在 OP 模式
	Busy	BOOL	TRUE：功能塊仍在忙於下載檔案
	Error	BOOL	TRUE：下載過程中發生錯誤
	ErrorID	DINT	錯誤識別

輸入/輸出	名稱	類型	註解
			命令特定錯誤代碼

備註:

- 在呼叫 “EM_InitMaster” FB 之前，必須至少將一個網路資訊文件預載入到 M801/e-M901 卡中。必須使用 ECAT Utility 建立並下載配置檔。配置檔的目的是檢查實際的 EtherCAT 網路從站設置是否與配置的網路相同，並映射過程數據（PDO）。
- “EM_InitMaster” 功能塊的執行需要幾個任務週期才能完成，因為初始化和配置網路中的每個從站以及將 EtherCAT 主站切換到 OP 模式需要一些時間。
- 當主站處於 OP 模式後，可以呼叫本章中的其他 FB。
- 列舉 EM_CYCLE_TIME 定義如下：

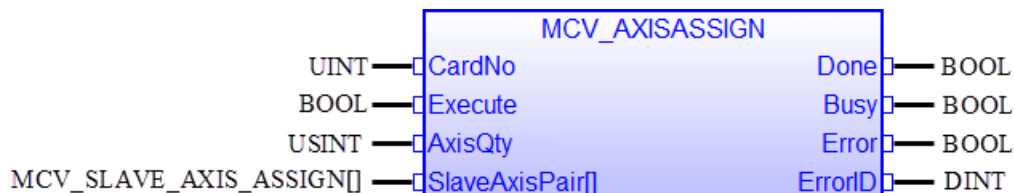
No.	EM_CYCLE_TIME	描述
0	CYCLE_TIME_1MS	1 millisecond cycle time
1	CYCLE_TIME_2MS	2 millisecond cycle time
2	CYCLE_TIME_3MS	3 millisecond cycle time
3	CYCLE_TIME_4MS	4 millisecond cycle time
4	CYCLE_TIME_5MS	5 millisecond cycle time
5	CYCLE_TIME_6MS	6 millisecond cycle time
6	CYCLE_TIME_7MS	7 millisecond cycle time
7	CYCLE_TIME_8MS	8 millisecond cycle time
8	CYCLE_TIME_9MS	9 millisecond cycle time
9	CYCLE_TIME_10MS	10 millisecond cycle time
10	CYCLE_TIME_11MS	11 millisecond cycle time
11	CYCLE_TIME_12MS	12 millisecond cycle time
12	CYCLE_TIME_13MS	13 millisecond cycle time
13	CYCLE_TIME_14MS	14 millisecond cycle time
14	CYCLE_TIME_15MS	15 millisecond cycle time

表 16：EtherCAT 循環時間定義

- 在 OP 模式下，主站在每個週期中訪問所有從站。如果其中一個從站在 OP 模式下與 EtherCAT 網路物理斷開，則主站將在每個週期中嘗試訪問從站。一旦從站再次連接到網路，master 將自動再次初始化它並進入正常的數據交換模式。

7.2 MCV_AxisAssign

該 MFB 為 EtherCAT 網路中的步進/伺服驅動從站分配一個軸號，以便 PLCopen 定義的 MFB 可以訪問它。這個 MFB 基本上是做軸 - 從站映射。通常 EtherCAT 從站只控制一個軸，但市場上的一些步進/伺服從站控制多個軸（例如泓格科技的 ECAT-2094S）。在這種情況下，還必須設置 “MCV_SLAVE_AXIS_ASSIGN” 結構的子軸編號 “SubAxisNo”。



輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	<ul style="list-style-type: none"> ECAT-M801/e-M901 卡號 卡號需要通過 ECAT-M801/e-M901 卡上的 DIP 開關設置
	Execute	BOOL	在正緣時，指定的伺服/步進驅動從站將映射到軸號
	AxisQty	USINT	要映射的軸數量
	SlaveAxisPair[]	MCV_SLAVE_AXIS_ASSIGN	從站-軸對映陣列
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE: 軸映射成功 FALSE: 軸分配失敗
	Busy	BOOL	<ul style="list-style-type: none"> TRUE: 功能塊仍在忙於軸分配
	Error	BOOL	<ul style="list-style-type: none"> TRUE: 映射過程中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 軸分配可能需要幾個週期時間，因為一旦映射過程完成，EtherCAT 主站將自動重置軸並嘗試清除所有軸錯誤。
- 當 “MCV_AXISASSIGN.Execute” 被觸發，軸將從所有群組中刪除。成功執行 “MCV_AXISASSIGN” 後，所有群組都將清空，並且必須再次將軸添加到群組中。
- 虛擬軸：MFB 支援虛擬軸分配。在這種情況下，從站編號必須設置為 65535。
- “MCV_SLAVE_AXIS_ASSIGN” 結構的定義：

```

STRUCT
    SlaveNo : UINT;

```

```
AxisNo : UINT;
SubAxisNo: UINT;
END_STRUCT
```

- 當從站控制多個軸時，必須設置“SubAxisNo”參數，否則必須將其設置為零。
- 範例 1：

從站 4、5 和 6 分別控制一個伺服驅動器。此外，還分配兩個虛擬軸

No.	EM_CYCLE_TIME	描述
SlaveAxisPair[0]	SlaveNo = 4; AxisNo = 0; SubAxisNo = 0;	將從站 4 映射到軸號 0
SlaveAxisPair[1]	SlaveNo = 5; AxisNo = 1; SubAxisNo = 0;	將從站 5 映射到軸號 1
SlaveAxisPair[2]	SlaveNo = 6; AxisNo = 2; SubAxisNo = 0;	將從站 6 映射到軸號 2
SlaveAxisPair[3]	SlaveNo = 65535; AxisNo = 3; SubAxisNo = 0;	將虛擬從站映射到軸號 3
SlaveAxisPair[4]	SlaveNo = 65535; AxisNo = 4; SubAxisNo = 0;	將虛擬從站映射到軸號 4

備註:

- 由於從站每個只控制一個軸，因此必須將“SubAxisNo”參數設置為零。
- 在此範例中，映射了三個軸，因此將軸數量設置為 3（“AxisQty” = 3）。
- 範例 2：

從站 3 控制 4 軸，從站 4 控制一個伺服驅動器，從站 5 控制 2 個軸。

No.	EM_CYCLE_TIME	描述
SlaveAxisPair[0]	SlaveNo = 3; AxisNo = 0; SubAxisNo = 0;	將從站 3 的第一軸（SubAxisNo=0）映射到軸號 0
SlaveAxisPair[1]	SlaveNo = 3; AxisNo = 1; SubAxisNo = 1;	將從站 3 的第二軸（SubAxisNo=1）映射到軸號 1
SlaveAxisPair[2]	SlaveNo = 3; AxisNo = 2; SubAxisNo = 2;	將從站 3 的第三軸（SubAxisNo=2）映射到軸號 2
SlaveAxisPair[3]	SlaveNo = 3; AxisNo = 3; SubAxisNo = 3;	將從站 3 的第四軸（SubAxisNo=3）映射到軸號 3
SlaveAxisPair[4]	SlaveNo = 4; AxisNo = 4; SubAxisNo = 0;	將從站 4 映射到軸號 4
SlaveAxisPair[5]	SlaveNo = 5; AxisNo = 5; SubAxisNo = 0;	將從站 5 的第一軸（SubAxisNo=0）映射到軸號 5

SlaveAxisPair[6]	SlaveNo = 5; AxisNo = 6; SubAxisNo = 1;	將從站 5 的第二軸 (SubAxisNo=1) 映射到軸號 6
------------------	---	------------------------------------

備註:

- 在此範例中，映射了 7 個軸，因此設置 'AxisQty'= 7 。

例：

VAR

SlaveAxisPair : ARRAY [0 .. 9] OF lib:MCV_SLAVE_AXIS_ASSIGN ;

flgAxisAssign : BOOL := FALSE ;

diAxisQty : DINT ;

Inst_MCV_AXISASSIGN : MCV_AxisAssign ;

CardNo : UINT := UINT#1 ;

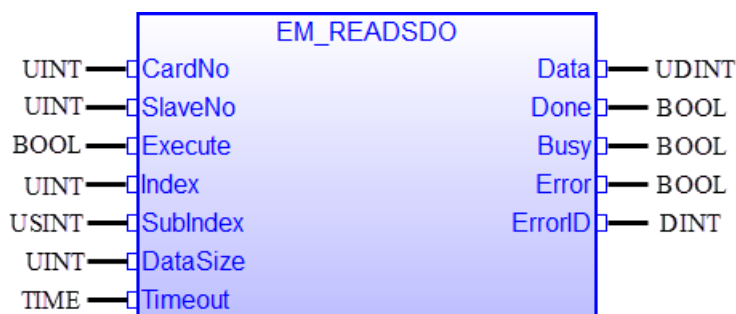
END_VAR

```
//=====
//Assign each axis number to a servo driver slave:
//=====
IF (flgAxisAssign = TRUE) THEN
  diAxisQty := 6;
  FOR i:=0 TO diAxisQty-1 DO
    SlaveAxisPair[i].SlaveNo := any_to_uint(i);
    SlaveAxisPair[i].AxisNo := any_to_uint(i);
    SlaveAxisPair[i].SubAxisNo := 0;
  END_FOR;
END_IF;

Inst_MCV_AXISASSIGN( CardNo, flgAxisAssign, any_to_usint(diAxisQty), SlaveAxisPair );
```

7.3 EM_ReadSdo

該 FB 從 EtherCAT 從站讀取服務數據物件（SDO）數據。這要求從站有一個 mailbox 並支援 CoE 協定。“Index” 和 “Subindex” 輸入參數選擇要讀取的物件。SDO 通訊是非週期性，其優先順序低於 PDO 通訊，因此響應時間較長。每次最多可以讀取 4 個位元組。



輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	<ul style="list-style-type: none"> ECAT-M801/e-M901 卡號 卡號需要通過 ECAT-M801/e-M901 卡上的 DIP 開關設置
	SlaveNo	UINT	EtherCAT 從站地址
	Execute	BOOL	在正緣讀取 SDO
	Index	UINT	<ul style="list-style-type: none"> 要讀取對象的 Index（十進制值） 請參考從站設備手冊中具有讀取權限的可用 Index
	SubIndex	USINT	<ul style="list-style-type: none"> 要讀取對象的 Sub-index（十進制值） 請參考從站設備手冊中具有讀取權限的可用 Sub-index
	DataSize	UINT	<ul style="list-style-type: none"> 要讀取的位元組數。支持的最大數據大小為 4 位元組
	Timeout	TIME	FB 執行所允許的最長時間（毫秒）
VAR_OUTPUT	Data	UDINT	讀取的 SDO 數據
	Done	BOOL	<ul style="list-style-type: none"> TRUE: SDO 數據已讀取完成
	Busy	BOOL	<ul style="list-style-type: none"> TRUE: 正在等待從站響應幀
	Error	BOOL	<ul style="list-style-type: none"> TRUE: SDO 讀取過程中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 僅可在成功執行 “EM_InitMaster” 後呼叫。

7.4 EM_WriteSdo

此 FB 用於在 slave 的對象字典中寫入特定對象。這要求從站有一個 mailbox 並支援 CoE 協定。
“Index” 和 “Subindex” 輸入參數選擇要寫入的對象。SDO 通訊是非週期性，其優先順序低於 PDO 通訊，因此響應時間較長。最多可寫入四個位元組長度的 SDO 數據類型。



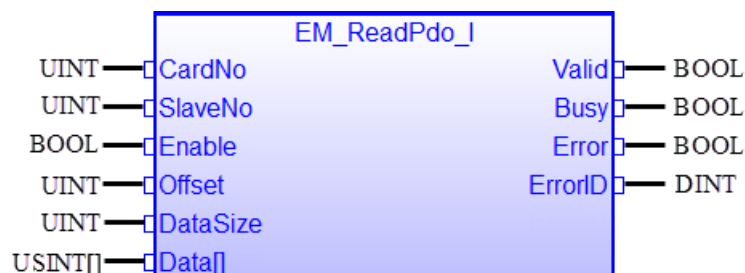
輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	<ul style="list-style-type: none"> ECAT-M801/e-M901 卡號 卡號需要通過 ECAT-M801/e-M901 卡上的 DIP 開關設置
	SlaveNo	UINT	EtherCAT 從站地址
	Execute	BOOL	在正緣寫入 SDO
	Index	UINT	<ul style="list-style-type: none"> 要寫入對象的 Index (十進制值) 請參考從站設備手冊中具有寫入權限的可用 Index
	SubIndex	USINT	<ul style="list-style-type: none"> 要寫入對象的 Sub-index (十進制值) 請參考從站設備手冊中具有寫入權限的可用 Sub-index
	DataSize	UINT	要寫入的字節數。支持的最大數據大小為 4 字節
	Data	UDINT	要寫入 SDO 的數據
	Timeout	TIME	FB 執行所允許的最長時間 (毫秒)
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE: SDO 數據已成功寫入
	Busy	BOOL	<ul style="list-style-type: none"> TRUE: 正在等待從站響應
	Error	BOOL	<ul style="list-style-type: none"> TRUE: SDO 寫入操作中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 僅可在成功執行 “EM_InitMaster” 後呼叫。

7.5 EM_ReadPdo

此 FB 讀取從 EtherCAT 從站接收到的過程數據對象 (PDO)。這需要 EtherCAT 主站處於 DC 模式。



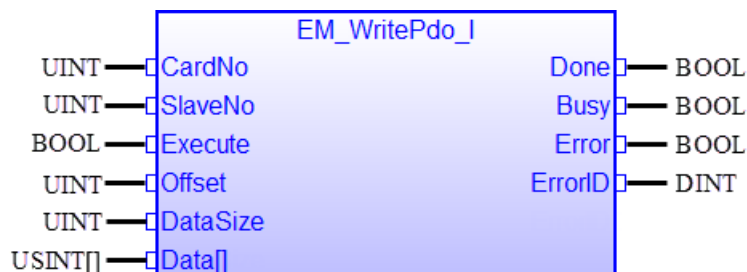
輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	<ul style="list-style-type: none"> ECAT-M801/e-M901 卡號 卡號需要通過 ECAT-M801/e-M901 卡上的 DIP 開關設置
	SlaveNo	UINT	EtherCAT 從站地址
	Enable	BOOL	<ul style="list-style-type: none"> TRUE: 在每個週期讀取 PDO
	Offset	UINT	<ul style="list-style-type: none"> 從從站 PDO 偏移的起始位置開始讀取
	DataSize	UINT	<ul style="list-style-type: none"> 要讀取的位元組數。支持的最大數據大小為 512 位元組
VAR_OUTPUT	Data[]	USINT[]	從主站接收到的 PDO 數據幀中讀取的數據數組 注意: 雖然參數顯示在功能塊的輸入側，但它實際上作為輸出使用
	Valid	BOOL	<ul style="list-style-type: none"> TRUE: 輸出數據 "Data[]" 有效
	Busy	BOOL	<ul style="list-style-type: none"> TRUE: 正在讀取從站接收到的 PDO 幀
	Error	BOOL	<ul style="list-style-type: none"> TRUE: PDO 讀取過程中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 僅可在成功執行 “EM_InitMaster” 後呼叫。

7.6 EM_WritePdo

此 FB 直接將過程數據寫入由 EtherCAT 主站傳送到指定從站的數據幀中。PDO 數據通訊是週期性的，具有最高優先權。



輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	<ul style="list-style-type: none"> ECAT-M801/e-M901 卡號 卡號需要通過 ECAT-M801/e-M901 卡上的 DIP 開關設置
	SlaveNo	UINT	EtherCAT 從站地址
	Execute	BOOL	在正緣寫入 PDO
	Offset	UINT	<ul style="list-style-type: none"> 過程數據的偏移位置 主站發送給從站的過程數據長度在配置階段設置，並在 OP 模式下固定。偏移位置允許用戶在分配特定從站的過程數據幀任意的的位置輸入/修改數據。 過程數據對象結構（例如條目的數量及其偏移和大小）由網絡配置檔案設置。
	DataSize	USINT	<ul style="list-style-type: none"> 要寫入的位元組數。 支持的最大數據大小為 512 位元組
	Data[]	USINT[]	<ul style="list-style-type: none"> 要寫入 PDO 的 USINT 陣列數據
VAR_OUTPUT	Done	BOOL	<ul style="list-style-type: none"> TRUE: PDO 數據已成功寫入
	Busy	BOOL	<ul style="list-style-type: none"> TRUE: 正在寫入 PDO
	Error	BOOL	<ul style="list-style-type: none"> TRUE: PDO 寫入操作中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

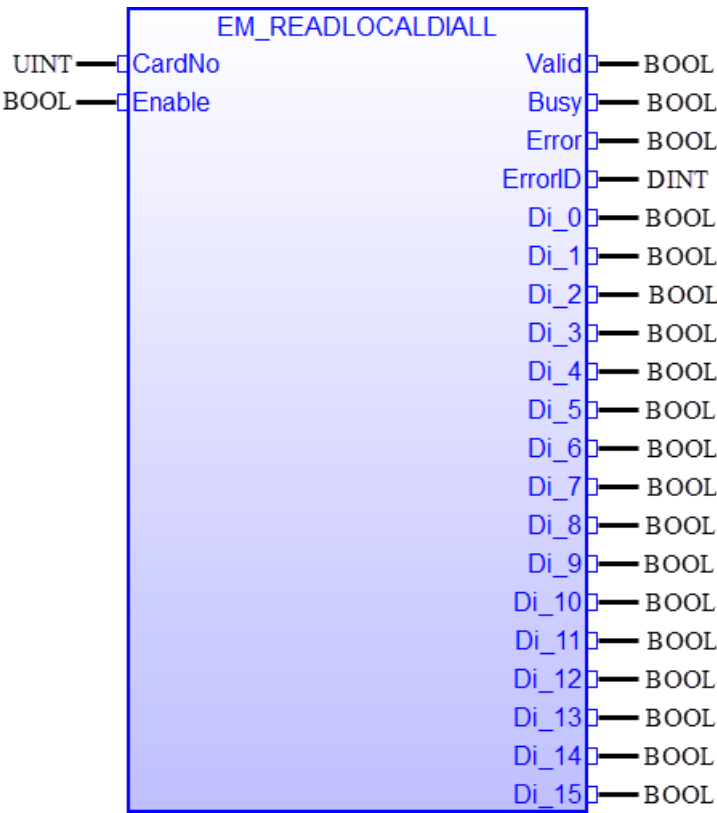
備註:

- 僅可在成功執行 “EM_InitMaster” 後呼叫。

7.7 EM_ReadLocalDiAll

此 FB 讀取 ECAT-M801/ e-M901 的本地數位輸入通道 (DI) 。

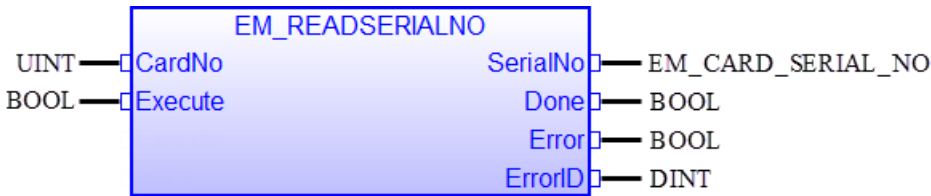
- ECAT-M801 具有 13 個板載數位輸入通道 (DI)
- e-M901 有 8 個板載數位輸入通道 (DI)



輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	• ECAT-M801/e-M901 卡號
	Enable	BOOL	• 在啟用狀態下持續讀取狀態圖的狀態
VAR_OUTPUT	Valid	BOOL	• TRUE: 數字輸入狀態有效 • FALSE: 讀取數字輸入狀態失敗
	Busy	BOOL	• TRUE: 正在忙於讀取 DI 通道
	Error	BOOL	• TRUE: 讀取操作中發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼
	Di_0 - Di_15	BOOL	• DI 狀態 • 對於 ECAT-M801，只有通道 Di_0 到 Di_12 有效 • 對於 e-M901，只有通道 Di_0 到 Di_7 有效

7.8 EM_ReadSerialNo

此 FB 讀取 ECAT-M801/e-M901 PCIe 卡的序號。



輸入/輸出	名稱	類型	註解
VAR_INPUT	CardNo	UINT	<ul style="list-style-type: none">ECAT-M801/e-M901 卡號卡號需要通過 ECAT-M801/e-M901 卡上的 DIP 開關設置
	Execute	BOOL	在正緣讀取 SDO
VAR_OUTPUT	SerialNo	EM_CARD_SERIAL_NO	ECAT-M801/e-M901 PCIe 卡的序號
	Done	BOOL	<ul style="list-style-type: none">TRUE: 已完成讀取序號
	Busy	BOOL	<ul style="list-style-type: none">TRUE: 正在忙於讀取
	Error	BOOL	<ul style="list-style-type: none">TRUE: 發生錯誤
	ErrorID	DINT	錯誤識別 命令特定錯誤代碼

備註:

- 結構 “EM_CARD_SERIAL_NO” 定義如下：

```
STRUCT
    Number : ARRAY[0..7] OF USINT;
END_STRUCT
```

將 8 個位元組的序號長度寫入 8 個元素的 USINT 陣列中。

8 ECAT Utility

ECAT Utility 是一款配置工具，用於建立網路設定檔、初始化和監控 EtherCAT 網路，並支援簡單的運動控制功能以便進行測試。此實用程式由泓格科技專門為 ECAT-M801 和 EMP9K (EMP-9xx8-xx) EtherCAT 主站開發，只能與這兩個主站一起使用。主伺服器和從伺服器必須處於連線狀態，工具程式才能正常運行。它掃描 EtherCAT 拓撲結構並在樹視圖中列出目前在線上的所有從站。

在本手冊的其餘部分，縮寫 EMP9K 用於表示 EMP-9xx8-xx。

8.1 安裝

ECAT Utility 已預先安裝在 EMP9K 平台上，無需額外安裝。

以下安裝過程僅適用於 Win-GRAF PC 運行時：

ECAT Utility 的安裝文件和手冊需要從泓格科技網站 “<https://www.icpdas.com/>” 下載。在搜尋框中輸入名稱 “ECAT-M801” 或 “EMP-9” 進入產品網站，接著點擊 “檔案中心” 按鈕：

- 從 “使用者手冊” 類別下載軟體和硬體手冊
- 下載 “ECAT Master Driver” 類別中列出的安裝檔。

8.1.1 軟體安裝

請依照《EtherCAT 主站軟體手冊》中所述的安裝說明安裝主機與 PCI 通訊和 ECAT-Utility 所需的驅動程式。

8.1.2 硬體安裝

《EtherCAT 主站硬體手冊》描述如何設定卡片 ID 編號、安裝 PCIe 卡、連接外部連接以及將裝置連接到 EtherCAT 從站。預設情況下，卡片 ID 設置為零。如果僅使用一張 ECAT-M801 卡，則無需設定卡片 ID。

8.2 EtherCAT 網路配置

本章介紹了設定 EtherCAT 網路的過程以及將伺服/步進控制從站整合到網路中所需的 ECAT-Utility 基本功能，使 PLCopen 定義的功能對軸進行命令控制。

僅針對 PC-Runtime 的準備工作：

在啟動 ECAT-Utility 之前，請確保 ECAT-M801 卡已插入 PCI 插槽並且 EtherCAT 從站已經連線。

步驟 1: 透過板載 DIP 開關設定 ECAT-M801/e-M901 的卡片 ID。預設卡 ID 為零。如果只使用一張 EtherCAT 主卡，則無需設定 ID。

步驟 2: 將 ECAT-M801 插入 PCI 插槽並確保所有從站都已連線。

針對 EMP9K 和 PC-Runtime 的準備工作:

第一步，需要掃描 EtherCAT 網路中的所有從站。

步驟 1: 將 EtherCAT 網路中每個從站的 ESI 檔新增到 ECAT-Utility 的 ESI 目錄中：

- ECAT-M801 設備：“C:\icpdas\Ecat-M801\ESI”
ESI 檔案由 EtherCAT 從站的製造商或經銷商提供。

步驟 2: 執行 ECAT Utility。

“ECAT_Utility.exe”可以在以下目錄中找到：

- ECAT-M801 設備：“C:\icpdas\Ecat-M801\Utility”

步驟 3: 彈出視窗要求您進行以下配置（圖 8-1）：

1. **Mdevice Card No.**：設定您要存取之卡 ID 的“MDevice Numbe”。預設情況下，ID 號碼設定為零。ECAT-M801 設備的 ID 透過指撥開關設定。對於 EMP-9000 系列，ID 固定為零。“MDevice”代表 EtherCAT 主站。
2. **Addressing Method**：選擇站點尋址方式。支援兩個選項：
 - **POSITION**：啟動時，EtherCAT 主站會根據每個從站在網路中的節點位置自動分配一個位址。最靠近主站的從屬節點被分配位址 0，並且每個後續從站節點的位址加 1。只有主站設定位址，從站不能更改它。切換網路中的節點位置會導致主站在下次啟動時根據新的從站偏移位置分配新的位址。因此，完成網路配置後，不能更改節點位置，否則應用程式將存取錯誤的從站。

- **ALIAS (ID)**：工作站別名是指定工作站的值（範圍從 0 到 65535）。使用者可以透過主站或直接設定從站的撥碼（DIP）或旋轉位址開關來為每個從站分配一個固定位址。優點是主站始終可以存取從站，即使其在網路中的站位置發生變化。如果選擇此選項，則需要為每個從站設定別名。確保為每個從站分配網路中唯一的別名值。

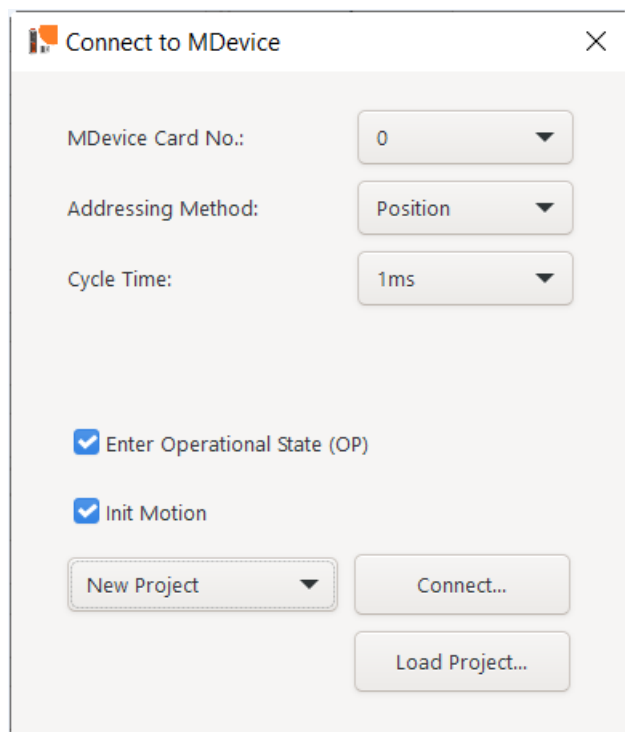


圖 8-1：專案配置

3. **Cycle Time**: 設置 EtherCAT 主站的週期時間。EtherCAT 主站將以設定的時間間隔與 EtherCAT 網路中的所有從站交換數據。
4. **Enter Operational State (OP)**: 如果啟用，該 utility 程式將在完成所有運動配置後自動將 EtherCAT 主站設定為運行狀態。
5. **Init Motion**: 如果啟用，則 Utility 將在每次 Utility 啟動後自動執行所有運動初始化。
6. **Select Project**: 透過此實用程式進行的設定將作為專案檔案保存在本機上。有兩個選項：
 - 可以新增專案。從下拉框中選擇 “New Project”。
 - 可以打開現有 Utility 專案。從下拉框中選擇一個專案，或使用 “Load Project” 按鈕打開現有專案。

7. **Connect...:** 點擊 “Connect...” 按鈕連接到 EtherCAT 主站和運動控制引擎。

- 對於新專案，Utility 程式將自動掃描 EtherCAT 網路並執行必要的網路和運動配置。（圖 8-2）
- 對於現有專案，Utility 程式使用專案檔案來配置 EtherCAT 網路和運動引擎。

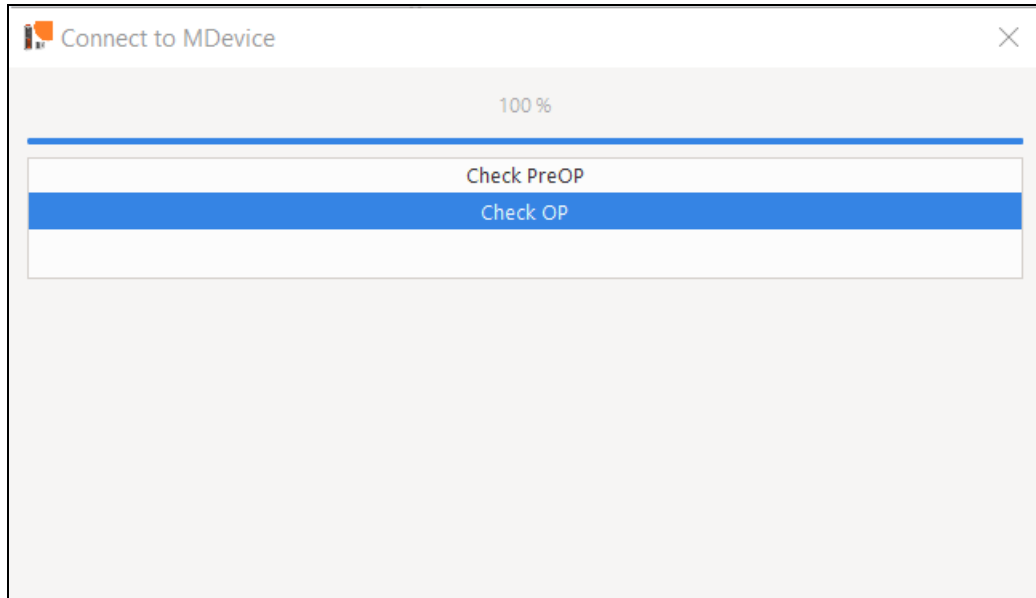


圖 8-2：主站正在掃描 EtherCAT 網路以尋找線上從站

掃描過程順利完成後，如果啟用了 “Enter Operational State (OP)” 和 “Init Motion”（圖 8-1），則 Utility 程式會自動執行以下步驟（圖 8-3）

1. 建立網路設定檔案並將其儲存在 EtherCAT 主站中的網路資訊檔案編號 0 下（tempupload）。
2. 為連接的步進和伺服驅動器產生運動設定檔案（auto.motcfg）。
3. 在使用者介面的清單中顯示所有偵測到的從站。（此處偵測到 6 台松下的 MADLN05BE 伺服驅動器。）
4. 顯示網路中的從站數量（此處為 6）
5. 將 EtherCAT 狀態機切換到 OP 模式。

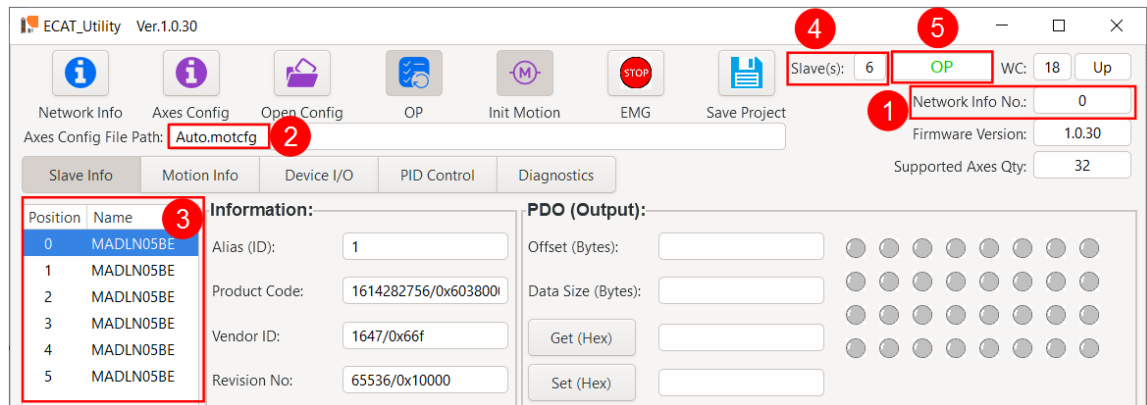


圖 8-3 : OP 模式下的 Utility 程式

現在使用者可以透過實用程式直接操作和控制每個從站。例如，可以透過 “Motion Info” 標籤發送運動命令，或者可以透過 “Device I/O” 標籤監視和控制數位和類比輸入和輸出。

8.2.1 手動網路配置

Utility 程式允許您手動設定網路設置。如果 “Enter Operational State (OP) ” (圖 8-1) 已停用或者您想更改自動產生的配置，則需要執行以下步驟。

- 步驟 1:** 掃描 EtherCAT 網路並以樹狀檢視列出所有線上從站：
點擊網路資訊 ⓘ 按鈕打開 Network Configuration 視窗。

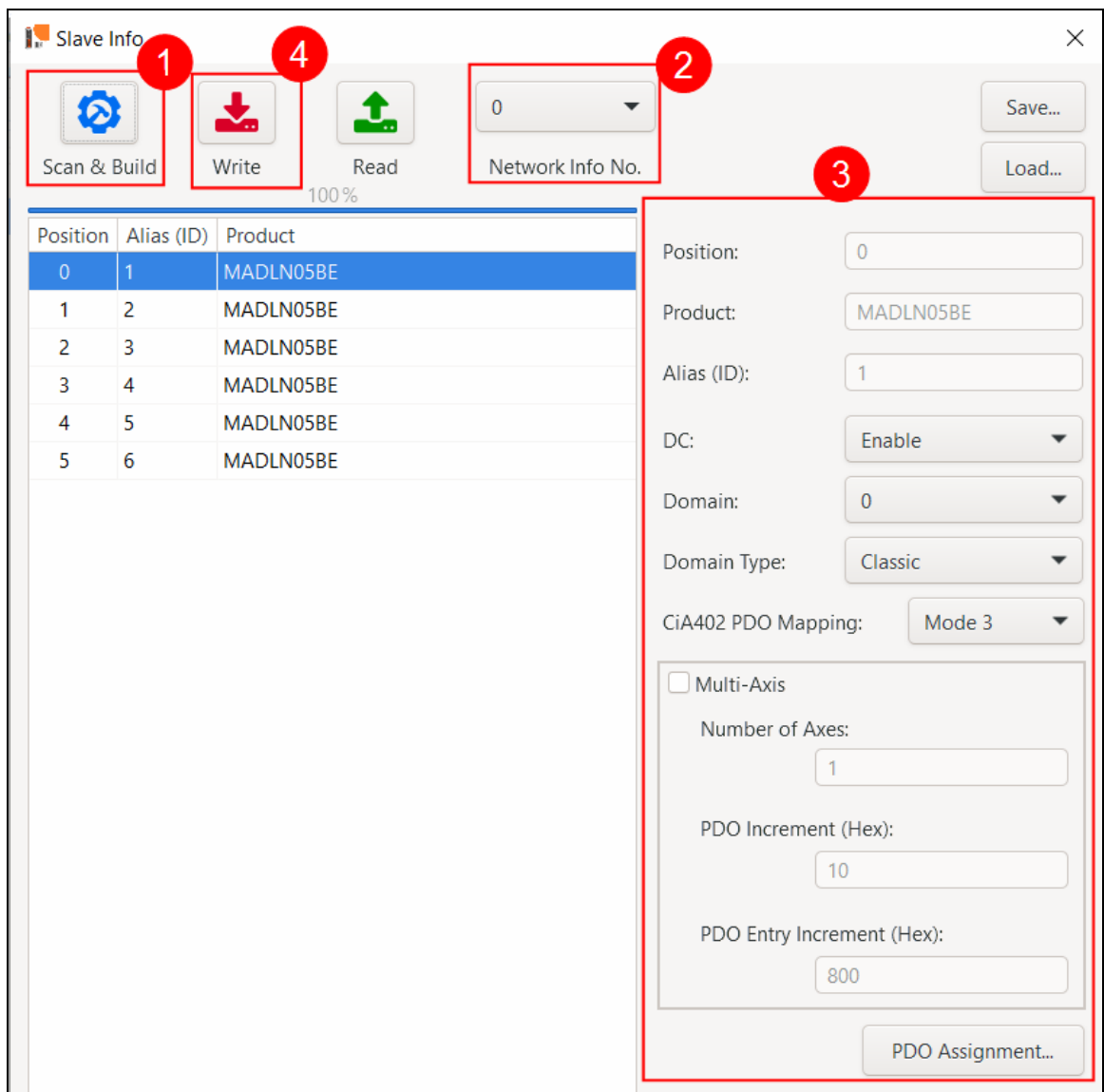


圖 8-4：手動配置 EtherCAT 網路

步驟 2: 建立網路配置檔案

1. 按一下“Scan & Build”  按鈕建立網路配置檔案。

EtherCAT 主站線上掃描網路中的所有站點，並根據其在網路中的節點位置將它們列在樹狀視圖中。最靠近主卡的 slave 節點列在頂部，第二個最近的 slave 節點列在樹狀檢視清單中的第二個位置，以此類推。子項目“Position”在樹狀圖中的值表示節點在網路中的位置。位置值越小，站點節點離主節點越近。

2. 設定網路資訊 ID 號碼。ECAT-M801/EMP9K 可以儲存多個網路資訊檔案。因此，使用 ID 號碼來識別文件非常重要。
您的 PLC 應用程式必須告訴主站使用哪個網路資訊檔案進行初始化和網路配置。

這是透過將功能區塊 “EM_InitMaster” 的輸入參數 “NetworkInfo” 指派給網路資訊檔案的 ID 號碼之一來完成的。在啟動階段，主站會根據分配的網路資訊檔案初始化並配置 EtherCAT 網路。

3. 設定和配置:

僅介紹運動控制相關配置。詳細說明請參閱 ECAT-Utility 手冊。

建立項目:

- i) **DC** : 如果從站支援 DC 通訊並符合系統的主站 DC 週期設置，則啟用 DC 模式。主站 DC 週期時間設定必須由 FB “EM_InitMaster” 透過 PLC 應用程式中的 “CycleTime” 輸入參數進行設定。如果 DC 週期時間快於從站支援的最小週期時間，請選擇以下選項之一：
 - 選項 1：不要啟用所選從站的 DC 模式，否則主機將無法進入操作（OP）模式
 - 選項 2：將 “EM_InitMaster” 的 “CycleTime” 輸入參數設置為等於或大於網路中最慢從站支援的週期時間的值。
- ii) **CiA 402 PDO 映射** :
CiA 402 物件字典定義了許多過程數據對象（PDO）。根據 CiA402 規範，並非所有物件都是必須的，因此某些 CiA402 從站僅支援基本 PDO。

使用預定義的 PDO 映射:

為了簡化數據映射過程，提供了已映射的 PDO 列表（模式 0、模式 1 等）（表 17）。您可以直接選擇其中一種模式。在選擇模式之前，必須確保 CiA402 從站支援該模式的所有 PDO 條目，否則主站不會進入運行模式或工作異常。

Mode	RxPDO		TxPDO	
Mode 0	6040	Controlword	6041	Statusword
	6060	Modes of operation	603F	Error code
	607A	Target Position	6061	Modes of operation display
	60FF	Target Velocity	6064	Position actual value
	60B8	Touch probe function	606C	Velocity actual value
	60B1	Velocity offset	60FD	Digital inputs
Mode1	6040	Controlword	6041	Statusword
	6060	Modes of operation	603F	Error code
	607A	Target Position	6061	Modes of operation display
			6064	Position actual value

Mode	RxPDO		TxPDO	
Mode2			606C	Velocity actual value
			60FD	Digital inputs
	6040	Controlword	6041	Statusword
	6060	Modes of operation	603F	Error code
	607A	Target Position	6061	Modes of operation display
	60FF	Target Velocity	6064	Position actual value
	6071	Target Torque	606C	Velocity actual value
	60B8	Touch probe function	60FD	Digital inputs
	60B0	Position offset	6077	Torque actual value
	60B1	Velocity offset		
	60B2	Torque offset		
Mode3	6040	Controlword	6041	Statusword
	6060	Modes of operation	603F	Error code
	607A	Target Position	6061	Modes of operation display
	60FF	Target Velocity	6064	Position actual value
			606C	Velocity actual value
			60FD	Digital inputs

表 17：CiA402 PDO 映射模式

使用者自定義的 PDO 映射:

使用者可以從下拉清單中選擇 “User Define” 來手動進行流程資料映射。這種方法很容易出錯，並且需要對映射過程有很好的理解。使用者定義的映射必須包括表 18 中所列的 CiA 402 物件。

映射過程：選擇 “User Define” 後，按一下 “PDO Assignment...” 按鈕開啟 PDO 設定視窗（圖 8-5）。從 “Unassigned PDOs” 清單中選擇一個 PDO，並將其移至 “Assigned PDOs” 清單中。每個 PDO 的物件項目顯示在 “PDO List” 中。點擊 “Insert” 按鈕以建立新的 PDO 或將新的 CiA402 項目新增至現有 PDO。關閉 “PDO Assignment” 以套用 PDO 配置。

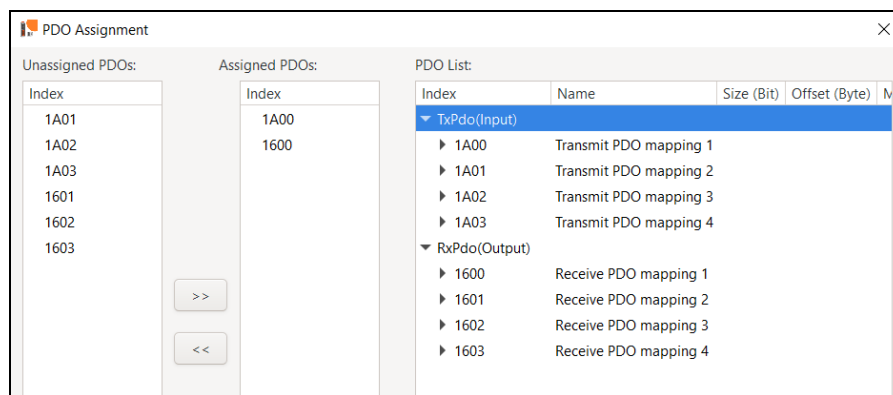


圖 8-5：手動 PDO 分配


RxPDO		TxPDO	
6040	Controlword	6041	Statusword
6060	Modes of operation	6061	Modes of operation display
607A	Target Position	6064	Position actual value
		60FD	Digital inputs

表 18：“用戶定義”基本 PDO 條目

iii) 多軸設定:

大多數 CiA 402 伺服/步進從站只控制一個軸。如果 CiA 402 從站控制多個軸，請勾選 “*Multi-axis*” 選框，但不要為四軸 ECAT-2094S 設備勾選該框，因為它不是 CiA 402 從站。

設置軸數。查閱從站用戶手冊以確定 PDO 和 PDO 條目遞增方式。

4. 點擊 “Write”  按鈕，將網路資訊檔下載到 ECAT-M801/EMP2K Mdevice。網路配置現已完成。

在 PLC 應用程式中，呼叫 “*EM_InitMaster*” 功能塊並將 “*NetworkInfo*” 輸入參數設定為 “*Network Info No*” 文字欄位中指定的編號。

8.3 運動控制測試

本章介紹如何設定 ECAT Utility 來直接控制單軸或群組。支援基本運動命令，如原點搜尋、點動、線性/圓周運動。

8.3.1 軸配置

如果啟用了 “*Operational State (OP)*” (圖 8-1)，則 ECAT Utility 在啟動期間會自動掃描和配置 EtherCAT 網路，並將 EtherCAT 狀態機切換到 OP 模式。如果啟用了 “*Init Motion*” (圖 8-1) 之前沒有進行軸配置，則 Utility 將自動進行基本運動配置。網路上的每個步進器/伺服從站現在都可以使用預設軸設置直接從實用程式進行控制。

如果預設值不符合您的要求，本節將介紹如何為每個軸設定參數。例如為每個步進/伺服從站分配軸號、設定脈衝解析度、原點搜尋方法...等。請務必記住，本章中所做的設置僅用於 ECAT-Utility，對 PLC 邏輯控制應用程式沒有影響。

以下步驟介紹如何建立新的運動配置檔案或修改 Utility 自動完成的配置設定。

步驟 1: 建立軸設定檔：

點擊 “Axis Config” 按鈕打開 Axis 配置對話框。

The Motion Config dialog box contains a table of axis configurations and a set of parameters for the selected axis.

AxisNo	SlaveNo	SubAxisNo
0	0	0
1	1	0
2	2	0
3	3	0
4	4	0
5	5	0

Parameters for the selected axis (0):

- Axis No: 0
- Slave No: 0
- Sub Axis: 0
- PPU: 1
- Home Search:
 - Method: 37
 - Speed (Search Switch): 10000
 - Speed (Search Z-Phase): 1000
 - Acceleration: 100000
- DI Active Level: 1
- Acceleration Time: 100
- Encoder PPR: 0
- Motor PPR: 51200

Default


- 單擊 “New file” 按鈕建立新檔案
- 透過右鍵單擊清單視圖並選擇 “addAxis” 或 “insertAxis”，將新軸新增/插入到清單中。

AxisNo	SlaveNo	SubAxisNo
0	0	0
1	1	0
2	2	0
3	3	0
4	4	0
5	5	0


add Axis
insert Axis
delete Axis

- 設置軸屬性:
- 從清單檢視中點選一個軸，接著設定從右側視窗中列出的所有參數。例如：設定伺服/步進驅動器的軸數、每單位脈衝數（PPU）解析度、回原點速度等。
- 將配置檔案以“*.motcfg”格式保存到本機 PC。到此步驟便完成軸配置過程。


步驟 2: 將軸設定檔載入到 ECAT Utility 記憶體中：

點擊“Open Config” 按鈕，然後選擇在上一步驟中建立的 Axis 配置檔“.motcfg”。

步驟 3: 將 EtherCAT 主站設置為 OP 模式：

點選“Start/Stop Operation” 按鈕，並等待主站成功進入 OP 模式。等待時間取決於網路中的從站數以及每個從站所需的配置時間。視窗頂端的狀態列顯示主站是否已進入 OP 模式。視窗底部的 info 欄位顯示主站無法進入 OP 模式時的錯誤類型。錯誤類型可以參閱 EtherCAT 主站用戶手冊“EtherCAT_Master_software_manual_tc.pdf”的使用者手冊。




步驟 4: 使用配置數據初始化軸：

點擊“Init Motion” 按鈕將 ECAT Utility 中的運動結構初始化為配置檔案中的數據。

步驟 5: ECAT Utility 現在可以直接控制每個軸。按兩下“Motion Info”選項卡，顯示所有軸的當前狀態，並顯示 Utility 支援的所有運動命令。

8.3.2 手動軸控制設定

每次啟動 ECAT Utility 程式後，都必須採取以下步驟來直接訪問每個伺服/步進器。如果選擇了“Enter Operational State”和“Init Motion”（圖 8-1），則 Utility 會自動執行這些步驟。

1. 通過點擊工具列上的“Start/Stop Operation”按鈕將主站設置為操作模式。主站將使用存儲在 ECAT-M801/EMP9K 設備中的網路資訊檔案編號，如第 8.2.1 章所述。如果主站找不到網路資訊檔，或者在網路中找不到檔案中列出的從站，則主站將不會進入操作模式（OP）。
2. 將軸配置檔案載入到 ECAT-Utility 記憶體中。此檔案基本上為每個伺服/步進電機從屬分配一個軸編號，Utility 程式可以通過該軸編號訪問它。
 - 按兩下“Open Config”並選擇之前創建的“.motcfg”檔。
3. 初始化運動引擎並控制軸。
 - 點擊“Init Motion”按鈕
 - 點選“Motion Info”選項（圖 8-6）以顯示每個軸的目前狀態
 - 點選視窗下半部分的“Axis”選項以顯示所有單軸命令的視窗。

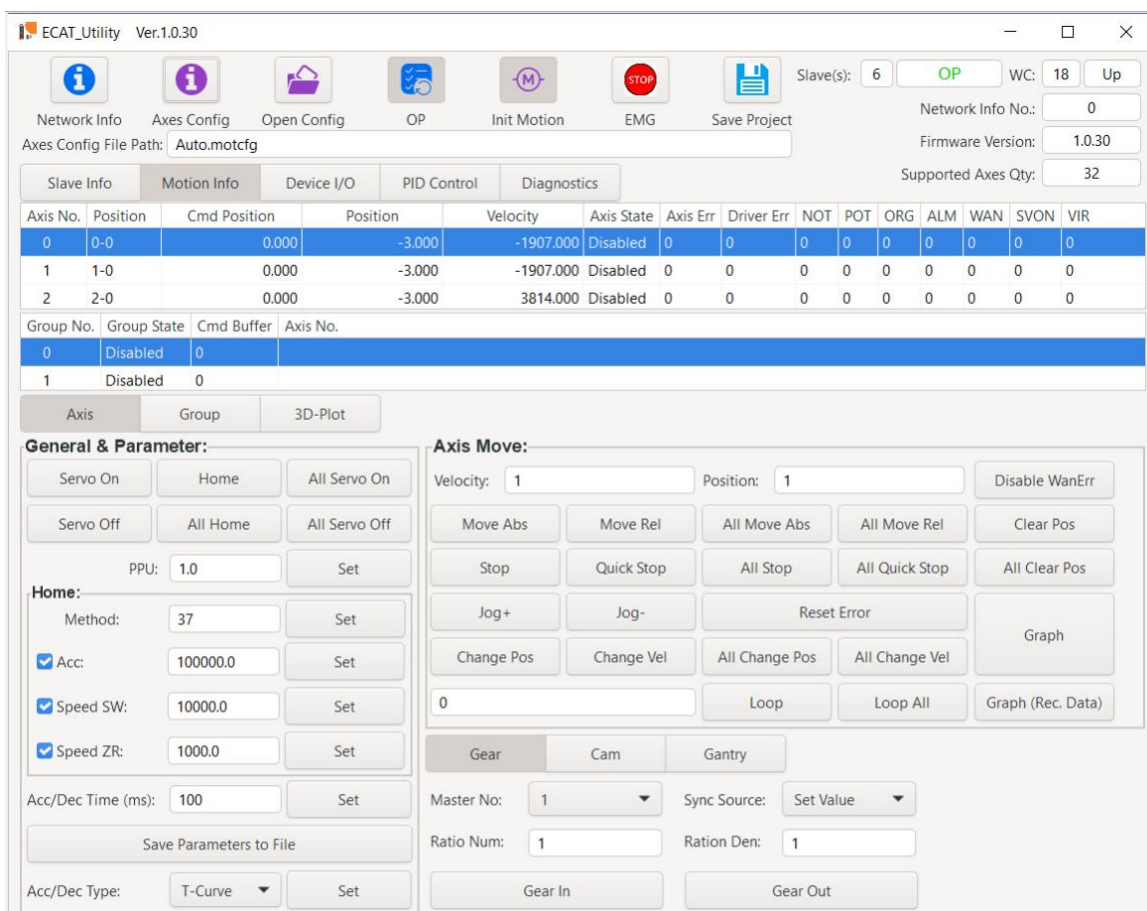


圖 8-6：“Motion Info” 頁籤


8.3.3 單軸控制

單軸參數設置：

1. 通過點擊單軸運動資訊面板（“Motion Info”）中的軸編號來選擇軸。當軸線以藍色背景色顯示時，表示該軸已被選中。
2. 基本運動參數已經在“*.motcfg*”檔案中設置，但可以更改這些設置。歸位搜尋、加/減速時間和曲線、脈衝解析度等參數都可以隨時調整。每個設置都點擊“Set”按鈕進行確認。請務必注意，這些設置不會保存到“*.motcfg*”檔案中。
3. 在執行歸位搜尋以外的任何行動命令之前，必須設置“Velocity”和“Position”參數。位置值可以是相對位置或絕對位置，具體取決於運動命令。

運動控制：

1. 點擊“Servo ON/OFF”按鈕以啟用或禁用驅動器。對於虛擬軸，無需此操作即可啟動運動控制。

2. 歸位搜尋：點擊 **“Home”** 按鈕開始此軸的歸位。
3. 絕對距離移動：在 **“Position”** 字段中設置絕對位置，然後點擊 **“MoveAbs”** 按鈕以命令所選軸移動到新位置。
4. 相對距離移動：在 **“Position”** 字段中設置相對位置，然後點擊 **“MoveRel”** 按鈕以驅動軸移動設定的距離。
5. 停止：點擊 **“Stop”** 或 **“QuickStop”** 按鈕停止軸。
 - **“QuickStop”** 命令將立即停止軸而不減速。
 - **“Stop”** 命令會根據 **“Acc/Dec Time”**（加/減速時間）值控制減速停止。
6. 點動操作（Jog）：以設定的 **“Velocity”** 速度開始運動。點擊 **“Jog+”** 或 **“Jog-”** 按鈕。按住滑鼠左鍵時，運動持續進行。鬆開滑鼠按鈕後，移動會減速並停止。
7. 緊急停止：點擊 **“EMG”**  按鈕立即停止所有軸的運動。

有關單軸和多軸命令的更多資訊，請參閱手冊 **“EtherCAT_Master_software_manual_tc.pdf”**。

9 附錄

9.1 縮寫

縮寫	描述
FB	功能塊
MFB	運動功能塊
CoE	基於 EtherCAT 的 CAN 應用協定
DC	分散式時鐘
ENI	EtherCAT 網路資訊
ESC	EtherCAT 從站控制器
OP	EtherCAT 主站或從站狀態 “Operational”
PreOP	EtherCAT 設備狀態 “Pre-Operational”
SafeOP	EtherCAT 設備狀態 “Safe-Operational”
WKC	工作計數器
EMP9K	EMP-9xx8-xx 系列 (例如:EMP-9058-16、EMP-9258-32、EMP-9098-16 等)

9.2 錯誤代碼

錯誤ID	錯誤代碼	描述
ECAT_ERR_REQUEST_MASTER	-1001	請求 master 資料失敗
ECAT_ERR_ETHERNET_LINK_DOWN	-1002	乙太網路連接異常
ECAT_ERR_SLAVES_STATE	-1003	從站模組未處於 OPERATIONAL 狀態
ECAT_ERR_WORKING_COUNTER	-1004	工作計數器數值異常
ECAT_ERR_SLAVE_CNT_EXCEEDED	-1005	連接的從站數量超過最大支援從站數量
ECAT_ERR_CREATE_DOMAIN	-1006	無法建立網域資料
ECAT_ERR_ALLOCATE_SLAVE_DATA	-1007	無法分配從站資料
ECAT_ERR_CONFIG_SLAVE	-1008	無法配置從站
ECAT_ERR_NETWORK_MISMATCH	-1009	目前連接的從站資訊與配置的從站資訊不相符
ECAT_ERR_MASTER_ACTIVATE	-1010	啟動 master 失敗
ECAT_ERR_GET_PROCESS_DATA	-1011	無法取得網域流程資料

ECAT_ERR_CONFIG_CYCLIC_TASK	-1012	循環任務配置失敗
ECAT_ERR_RUN_CYCLIC_TASK	-1013	無法運行循環任務
ECAT_ERR_INVALID_SLAVE_TYPE	-1014	無效的從站類型
ECAT_ERR_SAME_SLAVE_NO	-1015	相同的從屬編號
ECAT_ERR_INVALID_SLAVE_NO	-1016	無效的從屬編號
ECAT_ERR_INVALID_PARAM	-1017	參數無效
ECAT_ERR_INVALID_DATA_SIZE	-1018	數據大小無效
ECAT_ERR_SDO_REQUEST_BUSY	-1019	正在處理 SDO 請求
ECAT_ERR_SDO_REQUEST_ERROR	-1020	SDO 請求處理失敗
ECAT_ERR_ALLOCATE_PDO_QUEUE	-1021	無法分配 PDO 佇列數據
ECAT_ERR_INVALID_OFFSET	-1022	數據偏移量無效
ECAT_ERR_INIT_MOTION	-1023	初始化動作失敗
ECAT_ERR_GET_SLAVE_INFO	-1024	無法獲取 slave 資訊
ECAT_ERR_OPEN_FILE	-1025	無法打開檔案
ECAT_ERR_WRITE_FILE	-1026	無法將數據寫入檔案
ECAT_ERR_READ_FILE	-1027	無法從檔案中讀取資料
ECAT_ERR_FUNC_NOT_SUPPORT	-1028	不支援函數
ECAT_ERR_INVALID_CHANNEL	-1029	頻道參數無效
ECAT_ERR_EMG_HAPPENED	-1030	緊急情況發生
ECAT_ERR_INVALID_PID_NO	-1031	PID 編號無效
ECAT_ERR_TIMER_NOT_ACTIVATED	-1032	計時器未啟動
ECAT_ERR_ALL_EVENT_CREATE	-1033	創建的所有事件
ECAT_ERR_EVENT_NOT_CREATE	-1034	未創建事件
ECAT_ERR_INVALID_EVENTID	-1035	無效的事件ID
ECAT_ERR_INVALID_FILTER_TYPE	-1036	無效的篩選器類型
ECAT_ERR_SLAVES_ALIAS	-1037	重複別名或別名 == 0
ECAT_ERR_SLAVES_ALIAS_NOT_EXIST	-1038	別名不存在
ECAT_ERR_OPTASK	-1039	主伺服器處於 OPERATIONAL 狀態
ECAT_ERR_MC_NOT_ENABLE_DC	-1100	未設定啟用DC
ECAT_ERR_MC_TIME_OUT	-1101	呼叫動作功能超時
ECAT_ERR_MC_AXIS_CNT_EXCEEDED	-1102	初始化的軸數超過最大支撐軸數
ECAT_ERR_MC_NOT_INITIALIZED	-1103	Motion 未初始化
ECAT_ERR_MC_INVALID_AXIS_NO	-1104	軸編號無效
ECAT_ERR_MC_NOT_AXIS_SERVO_ON	-1105	軸伺服未開啟
ECAT_ERR_MC_INVALID_AXIS_STATE	-1106	無效的軸狀態
ECAT_ERR_MC_DRIVE_FAULT	-1107	驅動器故障
ECAT_ERR_MC_DRIVE_WARNING	-1108	驅動器告警
ECAT_ERR_MC_INVALID_PARAM	-1109	無效的運動參數

ECAT_ERR_MC_HOMING	-1110	歸位故障
ECAT_ERR_MC_LIMIT_ACTIVE	-1111	限位開關處於活動狀態
ECAT_ERR_MC_INVALID_ACC_TIME	-1112	加速時間無效
ECAT_ERR_MC_INVALID_GROUP_NO	-1113	群組號無效
ECAT_ERR_MC_INVALID_GROUP_STATE	-1114	無效的群組狀態
ECAT_ERR_MC_AXIS_WAS_IN_GROUP	-1115	Axis 已在群組中
ECAT_ERR_MC_AXIS_IN_OTHER_GROUP	-1116	Axis 已在其他群組中
ECAT_ERR_MC_GROUP_CMD_ALLOCATE	-1117	無法呼叫群組命令
ECAT_ERR_MC_GROUP_CMD_BUFFER_OVERFLOW	-1118	群組命令緩衝區已滿
ECAT_ERR_MC_INVALID_AXIS_SYNC_MODE	-1119	無效的軸同步模式
ECAT_ERR_MC_INVALID_PROFILE_NO	-1120	配置檔案編號無效
ECAT_ERR_MC_INVALID_GROUP_MOVE_CMD	-1121	無效的群組命令
ECAT_ERR_MC_GROUP_CMD_MODE_NOT_SUPPORT	-1122	該功能不支援當前組命令模式
ECAT_ERR_MC_INVALID_ACC_DEC_TYPE	-1123	加速類型參數無效
ECAT_ERR_MC_INVALID_VEL	-1124	速度參數無效
ECAT_ERR_MC_INVALID_ANGLE	-1125	角度參數無效
ECAT_ERR_MC_INVALID_RADIUS	-1126	半徑參數無效
ECAT_ERR_MC_INVALID_END_POS	-1127	無效的結束位置 參數
ECAT_ERR_MC_INVALID_ECAM_TABLE_NO	-1128	無效的 E-CAM 表 數字
ECAT_ERR_MC_INVALID_NORMAL_VECTOR	-1129	無效的法線向量 參數
ECAT_ERR_MC_NOT_SETUP	-1130	未設置
ECAT_ERR_MC_GREATER_THAN_MAX_RODLENGTH	-1131	計算值大於最大伸長長度
ECAT_ERR_MC_LESS_THAN_RODLENGTH	-1132	計算值小於伸長長度
ECAT_ERR_MC_GREATER_THAN_RECORD_COUNT	-1133	超過最大記錄計數
ECAT_ERR_MC_SOFTWARE_LIMIT_ACTIVATE	-1134	軟體限制處於活動狀態
ECAT_ERR_MC_GANTRY_POS_EXCESSIVE_DEVIATION	-1135	龍門控制位置偏差過大
ECAT_ERR_MC_GROUP_NO_NOT_SUPPORT	-1136	不支援群組號
ECAT_ERR_MC_INVALID_MOVE_CMD	-1137	無效的移動命令
ECAT_ERR_MC_QUEUE_IS_FULL	-1138	佇列已滿
ECAT_ERR_MC_COORDINATE_TRANS_ON	-1139	座標轉換處於活動狀態
ECAT_ERR_IPC_INVALID_DEVICE_NO	-1201	設備編號無效
ECAT_ERR_IPC_DEVICE_IS_OPEN	-1202	設備已開啟
ECAT_ERR_IPC_DEVICE_NOT_OPEN	-1203	設備未開啟

ECAT_ERR_IPC_CREATE_HANDLE	-1204	無法建立 IPC 處理
ECAT_ERR_IPC_BUSY	-1205	IPC 忙碌中
ECAT_ERR_IPC_TIME_OUT	-1206	IPC 逾時
ECAT_ERR_IPC_INVALID_CMD	-1207	IPC 命令無效
ECAT_ERR_IPC_WRITE_SHM	-1208	無法將數據寫入分片記憶體
ECAT_ERR_IPC_READ_SHM	-1209	無法從分片記憶體中讀取數據
ECAT_ERR_IPC_RUN_DOWN_UP_LOAD	-1210	無法處理下載/上傳數據
ECAT_ERR_IPC_INVALID_SHM	-1211	分片記憶體無效
ECAT_ERR_IPC_DEVICE_NOT_READY	-1212	設備未就緒
ECAT_ERR_DRV_GET_INFO	-1301	無法獲取驅動程式資訊
ECAT_ERR_DRV_CREATE_HANDLE	-1302	無法建立驅動程式處理
ECAT_ERR_DRV_IOCTL	-1303	呼叫驅動程式 IO 控制錯誤
ECAT_ERR_DRV_DEVICE_NOT_FOUND	-1304	未找到設備
ERR_MC_DISABLED	-2001	單軸被禁用，因此無法移動
ERR_MC_STOPPING	-2004	單軸處於停止狀態，無法移動
ERR_MC_HOMING	-2005	單軸處於歸位狀態，不能被另一個移動命令打斷
ERR_MC_GS_DISABLED	-2011	群組已禁用，因此無法執行和移動命令
ERR_MC_GS_ERRORSTOP	-2013	群組處於 “GroupErrorStop” 狀態，並且不會執行任何運動命令
ERR_INPUT_PARA	-2021	MFB 輸入參數值不正確或不受支援
FB_ERR_AXIS_REF_INVALID	-2022	卡號或軸號超出支援範圍
ERR_MULTI_INSTANCE_CALL	-2024	已啟動同一 MFB 的多個實例。 對於某些 MFB，一次只有一個實例可以控制卡或/和軸。 例如 “EM_InitMaster”：只有一個實例可以控制 ECAT-M801 主卡。如果兩個 “EM_InitMaster” 的實例試圖控制，將產生錯誤。
ERR_GRP_AXIS_NOT_IN_GROUP	-2030	未向群組添加任何軸
ERR_GRP_AXIS_MISMATCH	-2031	檢測到內部群組分配錯誤
ERR_ARRAY_OVERFLOW	-2035	MFB 的輸入陣列的大小太小。增加陣列大小 (元素) 以解決此錯誤。 例如：陣列元素的數量與群組中的軸的數量不對應

ERR_AXES_GROUP_REF_INVALID	-2036	卡號或群組號碼超出範圍
ERR_CARD_NO_MISMATCH	-2037	群組和軸的卡號不相同
ERR_POWER_ON_MULTI_AXIS	-2039	<ul style="list-style-type: none"> 多個 “MC_Power” 實例試圖控制同一軸，根據 PLCopen (第 1 部分，第 29/14 頁)，這是不允許的 同一個 “MC_Power” 實例試圖控制多個軸
ERR_FILE	-2041	檔案操作錯誤
ERR_FILE_DIRECTORY_NOT_FOUND	-2042	找不到檔案目錄
ERR_PATH_ID_VALUE_NOT_SUPPORTED	-2043	檔案路徑ID不正確
ERR_STRING_LENGTH	-2044	字串太短或太長
ERR_INTERVAL_VALUE_NOT_SUPPORTED	-2045	不支援間隔值
ERR_INC_MOVE_FILE_EXECUTING	-2046	無法執行增量路徑檔案，因為它正在由另一個 MFB 執行。
ERR_INVALID_DATA_SIZE	-2049	不支援資料大小
ERR_BUFFER_MODE_NOT_SUPPORTED	-2102	不支援命令緩衝區模式
ERR_TRANSITION_MODE_NOT_SUPPORTED	-2104	不支援組過渡模式
ERR_CIRC_ANGLE_VALUE_MISSING	-2105	缺少圓弧命令的角度值
ERR_CIRC_MODE_NOT_SUPPORTED	-2106	不支援 Circular 命令
ERR_CIRC_TOO_MANY_AXES_IN_GROUP	-2107	圓弧命令中軸數太多
ERR_CIRC_DIMENSION	-2108	圓弧運動僅支援 2D 和 3D 尺寸