



EMP-S

運動控制函式庫

參考手冊

版本：V.1.01

日期：2021.07

<https://www.epcio.com.tw>



目 錄

I. MCCL 功能列表.....	2
II. MCCL 函式庫.....	9
A. 一般系統功能.....	9
B. 座標系統.....	16
C. 過行程保護.....	21
D. 直線、圓弧、圓與螺線運動(一般運動).....	25
E. 點對點運動.....	45
F. JOG 運動.....	55
G. 運動狀態檢視.....	57
H. 定位控制.....	62
I. 進階軌跡規劃.....	65
J. ETHERCAT 原點復歸.....	70
K. ETHERCAT 控制功能.....	73
L. 錯誤訊息代碼.....	85
III. 函式傳回值.....	86
IV. 運動控制函式庫初始設定.....	94



I. MCCL 功能列表

A. 一般系統功能

編號	函 式 名 稱	說 明
1	MCC_StartEcServer()	啟動 EcServer 應用程式
2	MCC_StartEcServerEx()	啟動 EcServer 應用程式
3	MCC_RtxInit()	建立與 EcServer 之通訊
4	MCC_RtxClose()	關閉且停止與 EcServer 之通訊
5	MCC_GetVersion()	讀取函式庫版本
6	MCC_CreateGroup()	新增運動群組
7	MCC_CreateGroupEx()	新增運動群組與設定運動命令佇列大小
8	MCC_CloseGroup()	關閉指定運動群組
9	MCC_CloseAllGroups()	關閉所有運動群組
10	MCC_SetMacParam()	設定機構參數
11	MCC_GetMacParam()	讀取機構參數
12	MCC_UpdateParam()	使系統反應更新後的參數
13	MCC_GetCmdQueueSize()	讀取運動命令佇列大小
14	MCC_InitSystem()	啟動運動控制函式庫
15	MCC_InitSystemEx()	啟動運動控制函式庫
16	MCC_CloseSystem()	結束運動控制函式庫
17	MCC_ResetMotion()	重置運動控制函式庫
18	MCC_SetSysMaxSpeed()	設定一般運動進給速度的上限
19	MCC_GetSysMaxSpeed()	讀取一般運動進給速度的上限
20	MCC_InitSimulation()	啟動運動控制函式庫模擬功能

B. 座標系統

編號	函 式 名 稱	說 明
1	MCC_SetAbsolute()	使用絕對座標型態
2	MCC_SetIncrease()	使用增量座標型態
3	MCC_GetCoordType()	讀取使用的座標型態
4	MCC_GetCurRefPos()	讀取各軸位置之直角座標值(不含補償)
5	MCC_GetCurPos()	讀取各軸位置之直角座標值(含補償)



6	MCC_GetPulsePos()	讀取各軸位置之馬達座標值(含補償)
7	MCC_GetENCValue()	讀取編碼器的計數值
8	MCC_DefineOrigin()	將目前位置視為原點
9	MCC_DefinePosHere()	將目前座標位置與實際機台位置對齊
10	MCC_DefinePos()	設定目前的座標位置

C. 過行程保護

編號	函 式 名 稱	說 明
1	MCC_EnableLimitSwitchCheck()	開啟硬體極限開關保護功能
2	MCC_DisableLimitSwitchCheck()	關閉硬體極限開關保護功能
3	MCC_SetOverTravelCheck()	設定軟體過行程保護功能
4	MCC_GetOverTravelCheck()	讀取軟體過行程保護的設定情形
5	MCC_GetLimitSwitchStatus()	讀取硬體極限開關的狀態

D. 直線、圓弧、圓、螺線運動(一般運動)

編號	函 式 名 稱	說 明
1	MCC_SetAccType()	設定加速型式
2	MCC_GetAccType()	讀取使用的加速型式
3	MCC_SetDecType()	設定減速型式
4	MCC_GetDecType()	讀取使用的減速型式
5	MCC_SetAccTime()	設定加速時間
6	MCC_GetAccTime()	讀取加速時間
7	MCC_SetDecTime()	設定減速時間
8	MCC_GetDecTime()	讀取減速時間
9	MCC_SetFeedSpeed()	設定進給速度
10	MCC_GetFeedSpeed()	讀取使用的進給速度
11	MCC_GetCurFeedSpeed()	讀取機台目前的進給速度
12	MCC_GetSpeed()	讀取目前各軸的速度
13	MCC_Line()	八軸同動直線運動
14	MCC_ArcXYZ()	X-Y-Z 空間任三點圓弧運動
15	MCC_ArcXY()	X-Y 平面圓弧運動
16	MCC_ArcYZ()	Y-Z 平面圓弧運動



17	MCC_ArcZX()	Z-X 平面圓弧運動
18	MCC_ArcXYZ_Aux()	X-Y-Z 空間任三點圓弧運動與輔助軸直線運動
19	MCC_ArcXY_Aux()	X-Y 平面圓弧與輔助軸直線運動
20	MCC_ArcYZ_Aux()	Y-Z 平面圓弧與輔助軸直線運動
21	MCC_ArcZX_Aux()	Z-X 平面圓弧與輔助軸直線運動
22	MCC_ArcThetaXY()	X-Y 平面圓弧運動(以旋轉角度為參數)
23	MCC_ArcThetaYZ()	Y-Z 平面圓弧運動(以旋轉角度為參數)
24	MCC_ArcThetaZX()	Z-X 平面圓弧運動(以旋轉角度為參數)
25	MCC_CircleXY()	X-Y 平面全圓運動
26	MCC_CircleYZ()	Y-Z 平面全圓運動
27	MCC_CircleZX()	Z-X 平面全圓運動
28	MCC_CircleXY_Aux()	X-Y 平面全圓與輔助軸直線運動
29	MCC_CircleYZ_Aux()	Y-Z 平面全圓與輔助軸直線運動
30	MCC_CircleZX_Aux()	Z-X 平面全圓與輔助軸直線運動
31	MCC_HelicalXY_Z()	在 X-Y 平面進行圓周運動的螺線運動
32	MCC_HelicalYZ_X()	在 Y-Z 平面進行圓周運動的螺線運動
33	MCC_HelicalZX_Y()	在 Z-X 平面進行圓周運動的螺線運動
34	MCC_HelicalXY_Z_Aux()	在 X-Y 平面進行圓周運動的螺線運動與輔助軸直線運動
35	MCC_HelicalYZ_X_Aux()	在 Y-Z 平面進行圓周運動的螺線運動與輔助軸直線運動
36	MCC_HelicalZX_Y_Aux()	在 Z-X 平面進行圓周運動的螺線運動與輔助軸直線運動

E. 點對點運動

編號	函 式 名 稱	說 明
1	MCC_SetPtPSpeed()	設定速度比例
2	MCC_GetPtPSpeed()	讀取使用的速度比例
3	MCC_PtP()	點對點運動
4	MCC_PtPX()	X 軸點對點運動
5	MCC_PtPY()	Y 軸點對點運動
6	MCC_PtPZ()	Z 軸點對點運動
7	MCC_PtPU()	U 軸點對點運動
8	MCC_PtPV()	V 軸點對點運動



9	MCC_PtPW()	W 軸點對點運動
10	MCC_PtPA()	A 軸點對點運動
11	MCC_PtPB()	B 軸點對點運動
12	MCC_SetPtPAccType()	設定點對點運動的各軸加速型式
13	MCC_GetPtPAccType()	讀取點對點運動的各軸加速型式
14	MCC_SetPtPDecType()	設定點對點運動的各軸減速型式
15	MCC_GetPtPDecType()	讀取點對點運動的各軸減速型式
16	MCC_SetPtPAccTime()	設定點對點運動的各軸加速時間
17	MCC_GetPtPAccTime()	讀取點對點運動的各軸加速時間
18	MCC_SetPtPDecTime()	設定點對點運動的各軸減速時間
19	MCC_GetPtPDecTime()	讀取點對點運動的各軸減速時間

F. JOG 運動

編號	函 式 名 稱	說 明
1	MCC_JogPulse()	微動(脈衝運動)
2	MCC_JogSpace()	吋動(單步運動)
3	MCC_JogConti()	連續吋動

G. 運動狀態檢視

編號	函 式 名 稱	說 明
1	MCC_GetMotionStatus()	讀取目前的運動狀態
2	MCC_GetCurCommand()	讀取執行中的運動命令之相關資訊
3	MCC_GetCommandCount()	讀取運動命令的庫存數目
4	MCC_ResetCommandIndex()	使運動命令編碼值歸零
5	MCC_GetCurPulseStockCount()	讀取目前軟體上的 Pulse 命令庫存數目
6	MCC_SetMaxPulseStockNum()	設定最大使用的 Pulse 命令庫存數目
7	MCC_GetMaxPulseStockNum()	讀取最大使用的 Pulse 命令庫存數目
8	MCC_GetErrorCode()	讀取現存錯誤的代碼
9	MCC_ClearError()	清除現存的錯誤狀態

H. 定位控制

編號	函 式 名 稱	說 明
1	MCC_SetCompParam()	設定齒輪齒隙、間隙補償參數
2	MCC_UpdateCompParam()	反應更新後的齒輪齒隙、間隙補償參數
3	MCC_SetMaxPulseSpeed()	設定各軸最大 Pulse 速度的上限
4	MCC_GetMaxPulseSpeed()	讀取各軸最大 Pulse 速度的上限
5	MCC_SetMaxPulseAcc()	設定各軸最大 Pulse 加速度的上限
6	MCC_GetMaxPulseAcc()	讀取各軸最大 Pulse 加速度的上限

I. 進階軌跡規劃

編號	函 式 名 稱	說 明
1	MCC_HoldMotion()	暫停運動
2	MCC_ContiMotion()	繼續運動
3	MCC_AbortMotionEx()	以設定的減速時間減速至停止，並放棄所有未執行之運動命令
4	MCC_EnableBlend()	開啟平滑運動功能
5	MCC_DisableBlend()	關閉平滑運動功能
6	MCC_CheckBlend()	檢查是否開啟平滑運動功能
7	MCC_DelayMotion()	設定運動延遲時間
8	MCC_CheckDelay()	檢查是否進入運動延遲狀態
9	MCC_TimeDelay()	設定運動延遲時間
10	MCC_OverrideSpeed()	設定一般運動與點對點運動的速度比例
11	MCC_OverrideSpeedEx()	設定一般運動與點對點運動的速度比例
12	MCC_GetOverrideRate()	讀取使用的一般運動與點對點運動速度比例

J. EtherCAT 原點復歸

編號	函 式 名 稱	說 明
1	MCC_EcatSetHomeAxis()	設定原點復歸的運動軸
2	MCC_EcatSetHomeMode()	設定各軸原點復歸模式
3	MCC_EcatSetHomeZeroSpeed()	設定原點復歸時尋找零點(Zero)的速



		度
4	MCC_EcatSetHomeSwitchSpeed()	設定原點復歸時尋找原點開關(Switch)的速度
5	MCC_EcatHome()	要求原點復歸運動
6	MCC_EcatGetGoHomeStatus()	確認原點復歸運動是否完成
7	MCC_EcatAbortHome()	停止原點復歸運動
8	MCC_GetHomeSensorStatus()	讀取 Home Sensor 接點的狀態

K. EtherCAT 系統功能

編號	函 式 名 稱	說 明
1	MCC_SetServoOn()	EtherCAT 伺服驅動器啟動
2	MCC_SetServoOff()	EtherCAT 伺服驅動器停止
3	MCC_EcatGetMasterErrorCode()	取得 EtherCAT 主站的錯誤碼(Error Code)
4	MCC_EcatClearMasterErrorCode()	清除 EtherCAT 主站的錯誤碼(Error Code)
5	MCC_EcatGetSlavePresent()	取得 EtherCAT 從站的連線狀態
6	MCC_EcatGetSlaveErrorCode()	取得 EtherCAT 從站的錯誤碼(Error Code)
7	MCC_EcatResetSlaveFault()	重置 EtherCAT 從站的錯誤(Fault)
8	MCC_EcatEnableWatchDog()	開啟軟體 Watch Dog 功能
9	MCC_EcatDisableWatchDog()	關閉軟體 Watch Dog 功能
10	MCC_EcatGetInput()	讀取指定 EtherCAT 輸出入控制卡的 16 點輸入訊號值
11	MCC_EcatGetOutput()	讀取指定 EtherCAT 輸出入控制卡的 16 點輸出訊號狀態值
12	MCC_EcatSetOutput()	設定指定 EtherCAT 輸出入控制卡的 16 點輸出訊號狀態值
13	MCC_EcatSetOutputEnqueue()	將 EtherCAT 輸出入控制卡的指定輸出接點之訊號狀態值存放至運動命令佇列中
14	MCC_EcatGetADCInput	讀取指定 EtherCAT 類比電壓輸入控制卡之輸入直流電壓值
15	MCC_EcatCoeSdoUpload()	對指定的 EtherCAT 從站下達 SDO 讀取指令
16	MCC_EcatCoeSdoDownload()	對指定的 EtherCAT 從站下達 SDO



		送指令
17	MCC_EcatGetMotorPosition()	讀取指定實體軸目前馬達的實際位置值
18	MCC_EcatGetMotorDemandPosition()	讀取指定實體軸目前馬達的命令位置值
19	MCC_EcatGetMotorVelocity()	讀取指定實體軸目前馬達的實際速度值
20	MCC_EcatGetMotorVelocityRaw()	讀取指定實體軸目前馬達的命令速度值
21	MCC_EcatGetMotorTorque()	讀取指定實體軸目前馬達的實際扭矩值
22	MCC_EcatGetMotorTorqueRaw()	讀取指定實體軸目前馬達的實際扭矩值
23	MCC_EcatGetMotorDemandVelocity()	讀取指定實體軸目前馬達的命令速度值
24	MCC_EcatGetMotorDemandTorque()	讀取指定實體軸目前馬達的命令扭矩值
25	MCC_EcatGetMotorRatedTorque()	讀取指定實體軸馬達的 Rated 扭矩值
26	MCC_EcatSetMotorTargetVelocity()	設定指定實體軸馬達的目標速度值
27	MCC_EcatSetMotorTargetTorque()	設定指定實體軸馬達的目標扭矩值
28	MCC_EcatLoadCFGFilePath()	設定 EtherCAT 組態設定檔的存放路徑
29	MCC_EcatSetESIPath()	設定 ESI 檔案的存放路徑
30	MCC_EcatLoadESIFileName()	設定 ESI 檔案的檔案名稱
31	MCC_EcatSetENIPath()	設定 ENI 檔案的存放路徑
32	MCC_EcatSetENIFileName()	設定 ENI 檔案的檔案名稱
33	MCC_EcatScanBus()	開始掃描主從站的拓樸連結狀況
34	MCC_EcatGetSlaveInfoByID()	取得 EtherCAT 從站的資訊
35	MCC_EcatClearSlaveInfo()	清除 EtherCAT 從站的資訊
36	MCC_EcatGenerateENI()	產生 ENI 檔案
37	MCC_EcatAutoGenerateENI()	自動產生 ENI 檔案

II. MCCL 函式庫

A. 一般系統功能

1. BOOL MCC_StartEcServer(

int *nSleepSec*

)

Description 啟動 EcServer 應用程式(檔名: EcServer.rtss)，檔案存放位置須與執行檔同一目錄(若為 64-bit 系統，動態函式庫: EcMaster.rtdll、emlIRTL8169.rtdll 與 emlII8254x.rtdll 亦放置在此目錄)。

Parameters *nSleepSec* 啟動 EcServer 應用程式的延遲秒數，需大於等於 1

Return Value 0 失敗
1 成功

2. BOOL MCC_StartEcServerEx(

LPCTSTR *lpEcServerPath*,

int *nSleepSec*

)

Description 啟動 EcServer 應用程式。

Parameters *lpEcServerPath* 存放欲啟動 EcServer 應用程式的絕對路徑與檔案名稱，如: "C:\\EcServer.rtss"，(若為 64-bit 系統，動態函式庫: EcMaster.rtdll、emlIRTL8169.rtdll 與 emlII8254x.rtdll 亦放置在相同目錄)

nSleepSec 啟動 EcServer 應用程式的延遲秒數，需大於等於 1

Return Value 0 失敗
1 成功

3. int MCC_RtxInit(

int *nAxis*

)

Description	建立與 EcServer 之通訊。	
Parameters	<i>nAxis</i>	串接 EtherCAT 伺服驅動器的運動軸數
Return Value	0	成功
	2	RTX 初始化失敗

4. int MCC_RtxClose()

Description	關閉且停止與 EcServer 之通訊。	
Return Value	0	成功

5. void MCC_GetVersion(TCHAR* *strVersion*)

Description	讀取函式庫版本。	
Parameters	<i>strVersion</i>	指向一 TCHAR 值，用來存放函式庫版本

6. int MCC_CreateGroup(int *xMapToCh*,int *yMapToCh*,int *zMapToCh*,int *uMapToCh*, int *vMapToCh*,int *wMapToCh*,int *aMapToCh*,int *bMapToCh*)

Description	此函式用來建立一個新的運動群組。	
	在呼叫 MCCL 中與運動群組有關的函式（如：MCC_Line()）之前，必須先利用此函式或 MCC_CreateGroupEx() 建立群組，並得到新建立群組之編號，作為其傳入參數（之一）。	
	此函式必須在初始化 MCCL（MCC_InitSystem()）前始可呼叫；又，在第一次呼叫此函式前請先呼叫 MCC_CloseAllGroups()。注意，任兩運動軸，不可對應到同一實體輸出 Channel。	
Parameters	<i>xMapToCh</i> ~ <i>bMapToCh</i>	指定此 Group 中 X ~ B 軸所對應之實體軸的編號(0 ~ 7)
	若運動軸並不對應到實體軸，則須傳入 <i>AXIS_INVALID(-1)</i>	
Return Value	大於等於零	新建立之群組編號

小於零

失敗，傳回值的意義請參考 IV.函式傳回值

7. int MCC_CreateGroupEx(

```
int xMapToCh,int yMapToCh,int zMapToCh,int uMapToCh,  
int vMapToCh,int wMapToCh,int aMapToCh,int bMapToCh,  
int nMotionQueneSize  
)
```

Description

此函式用來建立一個新的運動群組及設定運動命令佇列大小。

在呼叫 MCCL 中與運動群組有關的函式（如：MCC_Line()）之前，必須先利用此函式或 MCC_CreateGroup()建立群組，並得到新建立群組之編號，作為其傳入參數（之一）。

此函式必須在初始化 MCCL（MCC_InitSystem()）前始可呼叫；又，在第一次呼叫此函式前請先呼叫 MCC_CloseAllGroups()。注意，任兩運動軸，不可對應到同一實體輸出 Channel。

Parameters

xMapToCh ~ 指定此 Group 中 X ~ B 軸所對應之實體軸的編號(0 ~ 7)
bMapToCh
nMotionQueneSize 自訂運動命令佇列大小

若運動軸並不對應到實體軸，則須傳入 *AXIS_INVALID(-1)*

Return Value

大於等於零 新建立之群組編號
小於零 失敗，傳回值的意義請參考 IV.函式傳回值

8. int MCC_CloseGroup(

```
int nGroupIndex  
)
```

Description

關閉指定的群組。

Parameters

nGroupIndex Group 編號

Return Value

0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

9. int MCC_CloseAllGroups()

Description 關閉系統中所有群組。在第一次呼叫 **MCC_CreateGroup()** 或 **MCC_CreateGroupEx()** 之前，請先呼叫此函式。

Return Value 0 成功
 非零 失敗，傳回值的意義請參考 IV.函式傳回值

10. int MCC_SetMacParam(SYS_MAC_PARAM* *pstMacParam*, DWORD *dwSlaveId*)

Description 設定各軸之機構參數。

Parameters *pstMacParam* 指向一 SYS_MAC_PARAM 結構，用來存放欲設定之機構參數

dwSlaveId EtherCAT 從站(實體軸)的編號

Return Value 0 成功
 非零 失敗，傳回值的意義請參考 IV.函式傳回值

11. int MCC_GetMacParam(SYS_MAC_PARAM* *pstMacParam*, DWORD *dwSlaveId*)

Description 讀取指定軸之機構參數內容。

Parameters *pstMacParam* 指向一 SYS_MAC_PARAM 結構，用來存放欲讀取之機構參數內容

dwSlaveId EtherCAT 從站(實體軸)的編號

Return Value 0 成功
 非零 失敗，傳回值的意義請參考 IV.函式傳回值

12. int MCC_UpdateParam()

Description 反應更新後的機構、編碼器與原點復歸參數。在呼叫過 **MCC_InitSystem()**

後，如果再次使用 `MCC_SetMacParam()` 變更相關的參數，則須使用此函式系統才會反應更新後的設定值；但須注意，呼叫此函式之結果與 `MCC_ResetMotion()` 相似，系統將回復到初始狀態。

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

13. `int MCC_GetCmdQueueSize(WORD wGroupIndex)`

Description	讀取運動命令佇列的大小。	
Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	運動命令佇列的大小(單位為運動命令)
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

14. `int MCC_InitSystem(int nInterpolateTime,)`

Description	啟動運動控制函式庫。 除了 <code>MCC_CreateGroup()</code> 、 <code>MCC_CreateGroupEx()</code> 、 <code>MCC_SetMacParam()</code> 與 <code>MCC_SetCompParam()</code> 之外，在使用 MCCL 其他函式之前，必須先呼叫此函式或 <code>MCC_InitSystemEx()</code> 。此函式僅須呼叫一次即可。	
Parameters	<i>nInterpolateTime</i>	插值時間，單位為 ms，設定範圍為 1 ~ 50 ms，較小的插值時間會使 MCCL 擁有較佳的運作性能，但須視系統能否負荷而定。
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

15. `int MCC_InitSystemEx(double dInterpolateTime,)`

Description	啟動運動控制函式庫。 除了 MCC_CreateGroup()、MCC_CreateGroupEx()、MCC_SetMacParam() 與 MCC_SetCompParam() 之外，在使用 MCCL 其他函式之前，必須先呼叫此函式或 MCC_InitSystem() 。此函式僅須呼叫一次即可。	
Parameters	<i>dInterpolateTime</i>	插值時間，單位為 ms，設定範圍為 1 ~ 50 ms，較小的插值時間會使 MCCL 擁有較佳的運作性能，但須視系統能否負荷而定。
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

16. int MCC_CloseSystem()

Description	關閉運動控制函式庫。	
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

17. int MCC_ResetMotion()

Description	重置運動控制函式庫。在使用此函式後，將清除錯誤狀態，並使直角座標值與馬達座標值歸零，最後系統會回復到如同呼叫 MCC_InitSystem() 過後的初始狀態。	
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

18. int MCC_SetSysMaxSpeed(

double *dfMaxSpeed*

)

Description	設定一般運動(直線、圓弧、圓與螺線)進給速度的上限，避免在使用 MCC_SetFeedSpeed() 時所設定的進給速度超出系統的工作範圍，使用單位為 User Unit/sec(*註)。	
Parameters	<i>dfMaxSpeed</i>	進給速度的上限
Return Value	0	成功

非零

失敗，傳回值的意義請參考 IV.函式傳回值

19. double MCC_GetSysMaxSpeed()

Description 讀取一般運動(直線、圓弧、圓與螺線)進給速度的上限，使用單位為 User Unit/sec(*註)。

Return Value 進給速度的上限

20. int MCC_InitSimulation(**int *nInterpolateTime*,****)**

Description 啟動運動控制函式庫模擬功能。

除了 MCC_CreateGroup()、MCC_CreateGroupEx()、MCC_SetMacParam() 與 MCC_SetCompParam()之外，在使用 MCCL 其他函式之前，必須先呼叫此函式，此函式僅具有軟體模擬功能，僅須呼叫一次即可。

Parameters *nInterpolateTime* 插值時間，單位為 ms，設定範圍為 1~50 ms，較小的插值時間會使 MCCL 擁有較佳的運作性能，但須視系統能否負荷而定。

Return Value 0 成功

非零 失敗，傳回值的意義請參考 IV.函式傳回值

*註：User Unit 即使用者在設定機構參數時，自行選定之長(角)度單位(例：*dfPitch*、*dfHighLimit*、*dfLowLimit*)。一經選定，MCCL 內部亦全部使用同樣長(角)度單位；以下以 UU 代表之。

B. 座標系統

1. int MCC_SetAbsolute(

WORD *wGroupIndex*

)

Description 使用絕對座標模式。

Parameters *wGroupIndex* Group 編號

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

2. int MCC_SetIncrease(

WORD *wGroupIndex*

)

Description 使用增量座標模式。

Parameters *wGroupIndex* Group 編號

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

3. int MCC_GetCoordType(

WORD *wGroupIndex*

)

Description 讀取使用的座標模式。

Parameters *wGroupIndex* Group 編號

Return Value 0 使用增量座標
1 使用絕對座標
其他 失敗，傳回值的意義請參考 IV.函式傳回值

4. int MCC_GetCurRefPos(

double* *pdfX*,

double* *pdfY*,



```
double* pdfZ,
double* pdfU,
double* pdfV,
double* pdfW,
double* pdfA,
double* pdfB,
WORD wGroupIndex
```

)

Description 讀取目前各軸位置的直角座標值(不含補償)。

Parameters	<i>pdfX ~ pdfB</i>	指向一 double 值，用來存放 X ~ B 各軸目前位置之直角座標值（不含補償）
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

5. int MCC_GetCurPos(

```
double* pdfX,
double* pdfY,
double* pdfZ,
double* pdfU,
double* pdfV,
double* pdfW,
double* pdfA,
double* pdfB,
WORD wGroupIndex
```

)

Description 讀取目前各軸位置的直角座標值(含補償)。

Parameters	<i>pdfX ~ pdfB</i>	指向一 double 值，用來存放 X ~ B 各軸目前位置之直角座標值（含補償）
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功

非零

失敗，傳回值的意義請參考 IV.函式傳回值

6. int MCC_GetPulsePos(

```
    long*  plX,  
    long*  plY,  
    long*  plZ,  
    long*  plU,  
    long*  plV,  
    long*  plW,  
    long*  plA,  
    long*  plB,  
    WORD   wGroupIndex  
)
```

Description 讀取目前各軸位置的馬達座標值(或稱為 *Pulse 座標值*，含補償)。

Parameters	<i>plX</i> ~ <i>plB</i>	指向一 long 值，用來存放 X ~ B 各軸目前位置的馬達座標值(含補償)
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

7. int MCC_GetENCValue(

```
    long*  plValue,  
    DWORD  dwSlaveId  
)
```

Description 讀取編碼器的計數值，請參閱該從站的 OD 碼 0x6064h 的說明。

Parameters	<i>plValue</i>	指向一 long 值，用來存放編碼器的計數值
	<i>dwSlaveId</i>	EtherCAT 從站(實體軸)的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

```
8. int MCC_DefineOrigin(
    WORD    wAxis,
    WORD    wGroupIndex
)
```

Description 使特定 Group 指定的運動軸之座標值歸零，使用此函式須在指定的 Group 為運動停止狀態。

Parameters *wAxis* 指定的運動軸編號 0 ~ 7 分別代表 X ~ B 軸
 wGroupIndex Group 編號

Return Value 0 成功
 非零 失敗，傳回值的意義請參考 IV. 函式傳回值

```
9. int MCC_DefinePosHere(
    WORD    wGroupIndex,
    DWORD   dwAxisMask
)
```

Description 修正目前的系統座標值使得與機台實際的位置相符合。

在某些情況下有可能使用手動方式移動機台，這時候機台實際的位置與運動控制函式庫中的系統座標值將產生差異，假使系統有安裝編碼器，則在呼叫此函式成功後將利用編碼器的計數值修正系統座標值，此時系統座標值將反應機台真實的位置。

Parameters *wGroupIndex* Group 編號
 dwAxisMask 指定欲發生作用的軸，指定參數可為

EMP_AXIS_X	X 軸
EMP_AXIS_Y	Y 軸
EMP_AXIS_Z	Z 軸
EMP_AXIS_U	U 軸
EMP_AXIS_V	V 軸
EMP_AXIS_W	W 軸
EMP_AXIS_A	A 軸
EMP_AXIS_B	B 軸



EMP_AXIS_ALL 全部運動軸

以上參數可自由組合，以作用在 X、Z 與 V 軸上為例：

(EMP_AXIS_X | EMP_AXIS_Z | EMP_AXIS_V)

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

10. int MCC_DefinePos(

WORD *wAxis*,
double *dfPos*,
WORD *wGroupIndex*

)

Description 設定目前的系統座標值。

在某些情況下有可能機台需要重新開機，這時候編碼器位置將被歸零，當運動控制函式庫重新初始化後，此時機台實際的位置與運動控制函式庫中的系統座標值將產生差異，假使系統有安裝絕對值編碼器，則在呼叫此函式成功後將利用絕對值編碼器的計數值修正系統座標值，此時系統座標值將反應機台真實的位置。

Parameters	<i>wAxis</i>	指定的運動軸編號 0 ~ 7 分別代表 X ~ B 軸
	<i>dfPos</i>	設定的系統座標值
	<i>wGroupIndex</i>	Group 編號

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

C. 過行程保護

1. int MCC_EnableLimitSwitchCheck(

int *nMode*

)

Description

開啟極限開關保護功能，機構參數 SYS_MAC_PARAM 中的 *wOverTravelUpSensorMode*、*wOverTravelDownSensorMode* 須與實際極限開關配線方式和驅動器參數設定的一致(Normal Open 或 Normal Close)，如此才能正確執行此項功能。

當功能開啟時，在碰觸到該軸運動方向的極限開關時(例如往正方向移動且觸到正向極限開關，或往負方向移動且碰觸到負向極限開關)，將會停止輸出 Group 的運動命令(並產生錯誤記錄)。

MCC_EnableLimitSwitchCheck() 通常會與 MCC_GetErrorCode() 搭配使用，利用不斷呼叫 MCC_GetErrorCode() 可獲知系統是否因碰觸到極限開關而產生錯誤記錄(錯誤代碼 0xF701 ~ 0xF708 分別代表 X ~ B 軸碰觸極限開關，請參考 III. 錯誤訊息代碼)；當發現碰觸極限開關之錯誤時，一般作法可能是：在螢幕上顯示訊息告知操作員，然後在程式中呼叫 MCC_ClearError() 清除錯誤記錄，則此時系統可再往反方向退出極限開關。

Parameters

nMode

硬體極限開關保護模式，可設定為：

- 0 只要碰觸到極限開關，皆會停止輸出該軸之運動命令
- 1 只有在碰觸到該軸運動方向的極限開關時(例如往正方向移動且觸到正向極限開關，或往負方向移動且碰觸到負向極限開關)，才會停止輸出該軸之運動命令
- 2 只要碰觸到極限開關，皆會停止輸出該軸之運動命令，並且產生錯誤記錄
- 3 只有在碰觸到該軸運動方向的極限開關時(例如往

正方向移動且觸到正向極限開關，或往負方向移動且碰觸到負向極限開關)，才會停止輸出該軸之運動命令，並且產生錯誤記錄

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

2. int MCC_DisableLimitSwitchCheck()

Description 關閉極限開關保護功能。

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

3. int MCC_SetOverTravelCheck(

```
int  nOTCheck0,  
int  nOTCheck1,  
int  nOTCheck2,  
int  nOTCheck3,  
int  nOTCheck4,  
int  nOTCheck5,  
int  nOTCheck6,  
int  nOTCheck7,  
WORD wGroupIndex
```

)

Description 設定軟體過行程保護功能。

Parameters	<i>nCheckOT0</i> ~ <i>nCheckOT7</i>	設定對應各運動軸(X ~ B 軸)軟體過行程保護的功能；1 表示開啟，0 則表示關閉
	<i>wGroupIndex</i>	Group 編號

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

4. int MCC_GetOverTravelCheck(

```
int*  pnOTChk0,
```

```
int*   pnOTChk1,  
int*   pnOTChk2,  
int*   pnOTChk3,  
int*   pnOTChk4,  
int*   pnOTChk5,  
int*   pnOTChk6,  
int*   pnOTChk7,  
WORD   wGroupIndex
```

)

Description	讀取軟體過行程保護的設定情形。	
Parameters	<i>pnOTChk0</i> ~	指向一 int 值，用來存放目前各運動軸(X ~ B 軸)軟體
	<i>pnOTChk7</i>	過行程保護的設定情形，1 表示開啟，0 則表示關閉
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

5. int MCC_GetLimitSwitchStatus(

```
WORD*   pwStatus,  
WORD     wUpDown,  
DWORD    dwSlaveId
```

)

Description	讀取極限開關的狀態，使用此函式前須確認機構參數 SYS_MAC_PARAM 中的 <i>wOverTravelUpSensorMode</i> 、 <i>wOverTravelDownSensorMode</i> 須與實際極限開關配線方式和驅動器參數設定的一致(Normal Open 或 Normal Close)，如此才能正確執行此項功能。	
Parameters	<i>pwStatus</i>	指向一 WORD 值，用來存放欲讀取之極限開關的狀態
	<i>wUpDown</i>	0 讀取負向極限開關(OD 碼 0x60FDh 的 bit 0) 1 讀取正向極限開關(OD 碼 0x60FDh 的 bit 1)
	<i>dwSlaveId</i>	EtherCAT 從站(實體軸)的編號
Return Value	0	成功



非零

失敗，傳回值的意義請參考 IV.函式傳回值

D. 直線、圓弧、圓與螺線運動(一般運動)

```
1. int MCC_SetAccType(  
    char  cAccType,  
    WORD  wGroupIndex  
)
```

Description 設定在進行一般運動時的加速型式。

Parameters *cAccType* 加速型式，可設定為：

 'T' 使用梯型加速曲線

 'S' 使用 S 型加速曲線

wGroupIndex Group 編號

Return Value 0 成功

 非零 失敗，傳回值的意義請參考 IV.函式傳回值

```
2. int MCC_GetAccType(  
    WORD  wGroupIndex  
)
```

Description 讀取在進行一般運動時使用的加速型式。

Parameters *wGroupIndex* Group 編號

Return Value 0 目前使用梯型加速曲線

 1 目前使用 S 型加速曲線

```
3. int MCC_SetDecType(  
    char  cDecType,  
    WORD  wGroupIndex  
)
```

Description 設定在進行一般運動時的減速型式。

Parameters *cDecType* 減速型式，可設定為：

 'T' 使用梯型減速曲線

 'S' 使用 S 型減速曲線

	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

**4. int MCC_GetDecType(
 WORD *wGroupIndex*
)**

Description	讀取在進行一般運動時使用的減速型式。	
Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	0	目前使用梯型減速曲線
	1	目前使用 S 型減速曲線

**5. int MCC_SetAccTime(
 double *dfAccTime*,
 WORD *wGroupIndex*
)**

Description	設定在進行一般運動時加速到穩定速度所需的時間。	
Parameters	<i>dfAccTime</i>	要求的加速時間，必須大於 0，單位為 ms。
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

**6. double MCC_GetAccTime(
 WORD *wGroupIndex*
)**

Description	讀取在進行一般運動時加速到穩定速度所需的時間。	
Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	進行一般運動時加速到穩定速度所需的時間，單位為 ms。	

**7. int MCC_SetDecTime(
 double *dfDecTime*,**

WORD *wGroupIndex*

)

Description	設定在進行一般運動由穩定速度減速至停止運動所需的時間。	
Parameters	<i>dDecTime</i>	要求的減速時間，必須大於 0，單位為 ms。
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

8. double MCC_GetDecTime(

WORD *wGroupIndex*

)

Description	讀取在進行一般運動時由穩定速度減速至停止運動所需的時間。	
Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	進行一般運動時由穩定速度減速至停止運動所需的時間，單位為 ms。	

9. double MCC_SetFeedSpeed(

double *dfFeedSpeed,*

WORD *wGroupIndex*

)

Description	設定在進行一般運動時的進給速度，單位為 UU/sec；此值不可為 0。但一般運動實際的進給速度(可使用 MCC_GetCurFeedSpeed() 獲得)須參考是否曾使用 MCC_OverrideSpeed() 設定進給速度強制比例，例如最後一次設定進給速度倍強制比例使用 MCC_OverrideSpeed(150)，此時呼叫 MCC_SetFeedSpeed(10)，則一般運動實際使用的進給速度為 $10 \times 150\% = 15$ 。	
Parameters	<i>dfFeedSpeed</i>	要求的進給速度
	<i>wGroupIndex</i>	Group 編號
Return Value	實際設定的進給速度	

10. double MCC_GetFeedSpeed(

WORD *wGroupIndex*

)

Description	讀取在進行一般運動時所設定的進給速度。利用此函式所獲得的進給速度並未加入使用 <code>MCC_OverrideSpeed()</code> 後對實際進給速度的影響，而只是單純傳回 <code>MCC_SetFeedSpeed()</code> 時所使用的設定值。
Parameters	<i>wGroupIndex</i> Group 編號
Return Value	目前設定的進給速度

11. double MCC_GetCurFeedSpeed(

WORD *wGroupIndex*

)

Description	讀取機台目前實際的進給速度。
Parameters	<i>wGroupIndex</i> Group 編號
Return Value	機台目前實際的進給速度

12. int MCC_GetSpeed(

double* *pdfVel0*,

double* *pdfVel1*,

double* *pdfVel2*,

double* *pdfVel3*,

double* *pdfVel4*,

double* *pdfVel5*,

double* *pdfVel6*,

double* *pdfVel7*,

WORD *wGroupIndex*

)

Description	讀取各軸目前的進給速度值。
Parameters	<i>pdfVel0</i> ~ 指向一 double 值，用來存放欲讀取之目前各軸 <i>pdfVel7</i> (X ~ B 軸)的進給速度值 <i>wGroupIndex</i> Group 編號
Return Value	0 成功

非零

失敗，傳回值的意義請參考 IV.函式傳回值

13. int MCC_Line(

```
double  dfX0,  
double  dfX1,  
double  dfX2,  
double  dfX3,  
double  dfX4,  
double  dfX5,  
double  dfX6,  
double  dfX7,  
WORD    wGroupIndex,  
DWORD   dwAxisMask  
)
```

Description	從目前位置以直線運動方式移動到指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。		
Parameters	<i>dfX0 ~ dfX7</i>	目的點的座標值	
	<i>wGroupIndex</i>	Group 編號	
	<i>dwAxisMask</i>	指定欲發生作用的軸，指定參數可為	
	EMP_AXIS_X	X 軸	
	EMP_AXIS_Y	Y 軸	
	EMP_AXIS_Z	Z 軸	
	EMP_AXIS_U	U 軸	
	EMP_AXIS_V	V 軸	
	EMP_AXIS_W	W 軸	
	EMP_AXIS_A	A 軸	
	EMP_AXIS_B	B 軸	
	EMP_AXIS_ALL	全部運動軸	
	以上參數可自由組合，以作用在 X、Z 與 V 軸上為例：		
	(EMP_AXIS_X EMP_AXIS_Z EMP_AXIS_V)		
Return Value	大於或等於零	MCCL 給予此運動命令的編碼	

小於零

失敗，傳回值的意義請參考 IV.函式傳回值

14. int MCC_ArcXYZ(

```
double  dfRX0,  
double  dfRX1,  
double  dfRX2,  
double  dfX0,  
double  dfX1,  
double  dfX2,  
WORD    wGroupIndex  
)
```

Description	在 X-Y-Z 軸所構成的空間上，以圓弧運動的方式從目前位置經過指定的參考點到目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfRX0</i> ~ <i>dfRX2</i>	參考點的 X-Y-Z 軸座標值
	<i>dfX0</i> ~ <i>dfX2</i>	目的點的 X-Y-Z 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

15. int MCC_ArcXY(

```
double  dfRX0,  
double  dfRX1,  
double  dfX0,  
double  dfX1,  
WORD    wGroupIndex  
)
```

Description	在 X-Y 軸所構成的平面上，以圓弧運動的方式從目前位置經過指定的參考點到目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfRX0</i> , <i>dfRX1</i>	參考點的 X-Y 軸座標值
	<i>dfX0</i> , <i>dfX1</i>	目的點的 X-Y 軸座標值
	<i>wGroupIndex</i>	Group 編號

Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

```
16. int MCC_ArcYZ(  
    double  dfRX1,  
    double  dfRX2,  
    double  dfX1,  
    double  dfX2,  
    WORD    wGroupIndex  
)
```

Description	在 Y-Z 軸所構成的平面上，以圓弧運動的方式從目前位置經過指定的參考點到目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfRX1, dfRX2</i>	參考點的 Y-Z 軸座標值
	<i>dfX1, dfX2</i>	目的點的 Y-Z 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

```
17. int MCC_ArcZX(  
    double  dfRX2,  
    double  dfRX0,  
    double  dfX2,  
    double  dfX0,  
    WORD    wGroupIndex  
)
```

Description	在 Z-X 軸所構成的平面上，以圓弧運動的方式從目前位置經過指定的參考點到目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfRX2, dfRX0</i>	參考點的 Z-X 軸座標值
	<i>dfX2, dfX0</i>	目的點的 Z-X 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼

小於零

失敗，傳回值的意義請參考 IV.函式傳回值

18. int MCC_ArcXYZ_Aux(

```
double  dfRX0,  
double  dfRX1,  
double  dfRX2,  
double  dfX0,  
double  dfX1,  
double  dfX2,  
double  dfX3,  
double  dfX4,  
double  dfX5,  
double  dfX6,  
double  dfX7,  
WORD    wGroupIndex  
)
```

Description	在 X-Y-Z 軸所構成的空間上，以圓弧運動的方式從目前位置經過指定的參考點到目的點，且 U、V、W、A、B 軸以同動方式進行直線運動。成功呼叫此函式將增加運動命令的庫存數目。		
Parameters	<i>dfRX0</i> ~ <i>dfRX2</i>	參考點的 X-Y-Z 軸座標值	
	<i>dfX0</i> ~ <i>dfX2</i>	目的點的 X-Y-Z 軸座標值	
	<i>dfX3</i> ~ <i>dfX7</i>	目的點的 U-V-W-A-B 軸座標值	
	<i>wGroupIndex</i>	Group 編號	
Return Value	大於或等於零	MCCL 給予此運動命令的編碼	
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值	

19. int MCC_ArcXY_Aux(

```
double  dfRX0,  
double  dfRX1,  
double  dfX0,  
double  dfX1,
```

```
double dfX3,  
double dfX4,  
double dfX5,  
double dfX6,  
double dfX7,  
WORD wGroupIndex
```

)

Description	在 X-Y 軸所構成的平面上，以圓弧運動的方式從目前位置經過指定的參考點到目的點，且 U、V、W、A、B 軸以同動方式進行直線運動。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfRX0, dfRX1</i>	參考點的 X-Y 軸座標值
	<i>dfX0, dfX1</i>	目的點的 X-Y 軸座標值
	<i>dfX3 ~ dfX7</i>	目的點的 U-V-W-A-B 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

20. int MCC_ArcYZ_Aux(

```
double dfRX1,  
double dfRX2,  
double dfX1,  
double dfX2,  
double dfX3,  
double dfX4,  
double dfX5,  
double dfX6,  
double dfX7,  
WORD wGroupIndex
```

)

Description	在 Y-Z 軸所構成的平面上，以圓弧運動的方式從目前位置經過指定的參考點到目的點，且 U、V、W、A、B 軸以同動方式進行直線運動。成功	
-------------	--	--

呼叫此函式將增加運動命令的庫存數目。

Parameters	<i>dfRX1, dfRX2</i>	參考點的 Y-Z 軸座標值
	<i>dfX1, dfX2</i>	目的點的 Y-Z 軸座標值
	<i>dfX3 ~ dfX7</i>	目的點的 U-V-W-A-B 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

21. int MCC_ArcZX_Aux(

```
double dfRX2,  
double dfRX0,  
double dfX2,  
double dfX0,  
double dfX3,  
double dfX4,  
double dfX5,  
double dfX6,  
double dfX7,  
WORD wGroupIndex  
)
```

Description 在 Z-X 軸所構成的平面上，以圓弧運動的方式從目前位置經過指定的參考點到目的點，且 U、V、W、A、B 軸以同動方式進行直線運動。成功呼叫此函式將增加運動命令的庫存數目。

Parameters	<i>dfRX2, dfRX0</i>	參考點的 Z-X 軸座標值
	<i>dfX2, dfX0</i>	目的點的 Z-X 軸座標值
	<i>dfX3 ~ dfX7</i>	目的點的 U-V-W-A-B 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

22. int MCC_ArcThetaXY(

double *dfCX*,
double *dfCY*,
double *dfTheta*,
WORD *wGroupIndex*
)

Description	在 X-Y 軸所構成的平面上，以指定的圓心與移動角度進行圓弧運動。移動角度為負則進行順時針運動，移動角度為正則進行逆時針運動。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCX, dfCY</i>	指定的圓心座標值
	<i>dfTheta</i>	移動角度
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

23. int MCC_ArcThetaYZ(

double *dfCY*,
double *dfCZ*,
double *dfTheta*,
WORD *wGroupIndex*
)

Description	在 Y-Z 軸所構成的平面上，以指定的圓心與移動角度進行圓弧運動。移動角度為負則進行順時針運動，移動角度為正則進行逆時針運動。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCY, dfCZ</i>	指定的圓心座標值
	<i>dfTheta</i>	移動角度
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

24. int MCC_ArcThetaZX(

double *dfCZ*,
double *dfCX*,
double *dfTheta*,
WORD *wGroupIndex*
)

Description	在 Z-X 軸所構成的平面上，以指定的圓心與移動角度進行圓弧運動。移動角度為負則進行順時針運動，移動角度為正則進行逆時針運動。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCZ, dfCX</i>	指定的圓心座標值
	<i>dfTheta</i>	移動角度
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

25. int MCC_CircleXY(

double *dfCX*,
double *dfCY*,
BYTE *byCirDir*,
WORD *wGroupIndex*
)

Description	在 X-Y 軸所構成的平面上，由目前位置執行一個以指定點為圓心座標的全圓軌跡運動。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCX, dfCY</i>	圓心的 X-Y 軸座標值
	<i>byCirDir</i>	運動方向，0 為順時針運動，1 為逆時針運動
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

26. int MCC_CircleYZ(

double *dfCY*,

	double <i>dfCZ</i> ,
	BYTE <i>byCirDir</i> ,
	WORD <i>wGroupIndex</i>
)
Description	在 Y-Z 軸所構成的平面上，由目前位置執行一個以指定點為圓心座標的全圓軌跡運動。成功呼叫此函式將增加運動命令的庫存數目。
Parameters	<i>dfCY, dfCZ</i> 圓心的 Y-Z 軸座標值
	<i>byCirDir</i> 運動方向，0 為順時針運動，1 為逆時針運動
	<i>wGroupIndex</i> Group 編號
Return Value	大於或等於零 MCCL 給予此運動命令的編碼
	小於零 失敗，傳回值的意義請參考 IV.函式傳回值

27. int MCC_CircleZX(

	double <i>dfCZ</i> ,
	double <i>dfCX</i> ,
	BYTE <i>byCirDir</i> ,
	WORD <i>wGroupIndex</i>
)
Description	在 Z-X 軸所構成的平面上，由目前位置執行一個以指定點為圓心座標的全圓軌跡運動。成功呼叫此函式將增加運動命令的庫存數目。
Parameters	<i>dfCZ, dfCX</i> 圓心的 Z-X 軸座標值
	<i>byCirDir</i> 運動方向，0 為順時針運動，1 為逆時針運動
	<i>wGroupIndex</i> Group 編號
Return Value	大於或等於零 MCCL 給予此運動命令的編碼
	小於零 失敗，傳回值的意義請參考 IV.函式傳回值

28. int MCC_CircleXY_Aux(

double <i>dfCX</i> ,
double <i>dfCY</i> ,
double <i>dfU</i> ,
double <i>dfV</i> ,

```
double  dfW,  
double  dfA,  
double  dfB,  
BYTE    byCirDir,  
WORD    wGroupIndex
```

```
)
```

Description	在 X-Y 軸所構成的平面上，由目前位置執行一個以指定點為圓心座標的全圓軌跡運動，且 U、V、W、A、B 軸以同動方式進行直線運動。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCX, dfCY</i>	圓心的 X-Y 軸座標值
	<i>dfU ~ dfB</i>	目的點的 U-V-W-A-B 軸座標值
	<i>byCirDir</i>	運動方向，0 為順時針運動，1 為逆時針運動
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

29. int MCC_CircleYZ_Aux(

```
double  dfCY,  
double  dfCZ,  
double  dfU,  
double  dfV,  
double  dfW,  
double  dfA,  
double  dfB,  
BYTE    byCirDir,  
WORD    wGroupIndex
```

```
)
```

Description	在 Y-Z 軸所構成的平面上，由目前位置執行一個以指定點為圓心座標的全圓軌跡運動，且 U、V、W、A、B 軸以同動方式進行直線運動。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCY, dfCZ</i>	圓心的 Y-Z 軸座標值

	<i>dfU ~ dfB</i>	目的點的 U-V-W-A-B 軸座標值
	<i>byCirDir</i>	運動方向，0 為順時針運動，1 為逆時針運動
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

30. int MCC_CircleZX_Aux(

```
double dfCZ,  
double dfCX,  
double dfU,  
double dfV,  
double dfW,  
double dfA,  
double dfB,  
BYTE byCirDir,  
WORD wGroupIndex  
)
```

Description	在 Z-X 軸所構成的平面上，由目前位置執行一個以指定點為圓心座標的全圓軌跡運動，且 U、V、W、A、B 軸以同動方式進行直線運動。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCZ, dfCX</i>	圓心的 Z-X 軸座標值
	<i>dfU ~ dfB</i>	目的點的 U-V-W-A-B 軸座標值
	<i>byCirDir</i>	運動方向，0 為順時針運動，1 為逆時針運動
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

31. int MCC_HelicalXY_Z(

```
double dfCX,  
double dfCY,  
double dfPitch,
```

	double <i>dfTheta</i> ,
	WORD <i>wGroupIndex</i>
)
Description	由目前位置執行螺線運動，此運動在 X-Y 平面進行圓周運動，可利用 MCC_SetFeedSpeed() 設定此圓周運動的速度。使用此函式必須指定 X-Y 平面上圓周運動的圓心座標值，圓周半徑由目前位置與圓心座標值決定，並須規劃圓周運動移動的角度值，成功呼叫此函式將增加運動命令的庫存數目。
Parameters	<i>dfCX, dfCY</i> 圓周運動圓心的 X-Y 軸座標值
	<i>dfPitch</i> 在 X-Y 平面進行一個整圓運動後，Z 軸所移動的距離，此值必須大於 0
	<i>dfTheta</i> 圓周運動所移動的角度與運動方向，正值為順時針運動，負值為逆時針運動
	<i>wGroupIndex</i> Group 編號
Return Value	大於或等於零 MCCL 給予此運動命令的編碼
	小於零 失敗，傳回值的意義請參考 IV. 函式傳回值

32. int MCC_HelicalYZ_X(

double *dfCY*,
double *dfCZ*,
double *dfPitch*,
double *dfTheta*,
WORD *wGroupIndex*

)

Description	由目前位置執行螺線運動，此運動在 Y-Z 平面進行圓周運動，可利用 MCC_SetFeedSpeed() 設定此圓周運動的速度。使用此函式必須指定 Y-Z 平面上圓周運動的圓心座標值，圓周半徑由目前位置與圓心座標值決定，並須規劃圓周運動移動的角度值。成功呼叫此函式將增加運動命令的庫存數目。
Parameters	<i>dfCY, dfCZ</i> 圓周運動圓心的 Y-Z 軸座標值



	<i>dfPitch</i>	在 Y-Z 平面進行一個整圓運動後，X 軸所移動的距離，此值必須大於 0
	<i>dfTheta</i>	圓周運動所移動的角度與運動方向，正值為順時針運動，負值為逆時針運動
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

33. int MCC_HelicalZX_Y(

double *dfCZ*,
 double *dfCX*,
 double *dfPitch*,
 double *dfTheta*,
 WORD *wGroupIndex*

)

Description	由目前位置執行螺線運動，此運動在 Z-X 平面進行圓周運動，可利用 MCC_SetFeedSpeed() 設定此圓周運動的速度。使用此函式必須指定 Z-X 平面上圓周運動的圓心座標值，圓周半徑由目前位置與圓心座標值決定，並須規劃圓周運動移動的角度值。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCZ, dfCX</i>	圓周運動圓心的 Z-X 軸座標值
	<i>dfPitch</i>	在 Z-X 平面進行一個整圓運動後，Y 軸所移動的距離，此值必須大於 0
	<i>dfTheta</i>	圓周運動所移動的角度與運動方向，正值為順時針運動，負值為逆時針運動
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

34. int MCC_HelicalXY_Z_Aux(

```
double dfCX,  
double dfCY,  
double dfPitch,  
double dfU,  
double dfV,  
double dfW,  
double dfA,  
double dfB,  
double dfTheta,  
WORD wGroupIndex
```

```
)
```

Description 由目前位置執行螺線運動，此運動在 X-Y 平面進行圓周運動，且 U、V、W、A、B 軸以同動方式進行直線運動，可利用 MCC_SetFeedSpeed() 設定此圓周運動的速度。使用此函式必須指定 X-Y 平面上圓周運動的圓心座標值，圓周半徑由目前位置與圓心座標值決定，並須規劃圓周運動移動的角度值。成功呼叫此函式將增加運動命令的庫存數目。

Parameters	<i>dfCX, dfCY</i>	圓周運動圓心的 X-Y 軸座標值
	<i>dfPitch</i>	在 X-Y 平面進行一個整圓運動後，Z 軸所移動的距離，此值必須大於 0
	<i>dfU ~ dfB</i>	目的點的 U-V-W-A-B 軸座標值
	<i>dfTheta</i>	圓周運動所移動的角度與運動方向，正值為順時針運動，負值為逆時針運動
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於 0	MCCL 給予此運動命令的編碼
	小於 0	失敗，傳回值的意義請參考 IV. 函式傳回值

35. int MCC_HelicalYZ_X_Aux(

```
double dfCY,  
double dfCZ,  
double dfPitch,
```

double *dfU*,
double *dfV*,
double *dfW*,
double *dfA*,
double *dfB*,
double *dfTheta*,
WORD *wGroupIndex*

)

Description	由目前位置執行螺線運動，此運動在 Y-Z 平面進行圓周運動，且 U、V、W、A、B 軸以同動方式進行直線運動，可利用 MCC_SetFeedSpeed() 設定此圓周運動的速度。使用此函式必須指定 Y-Z 平面上圓周運動的圓心座標值，圓周半徑由目前位置與圓心座標值決定，並須規劃圓周運動移動的角度值。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCY, dfCZ</i>	圓周運動圓心的 Y-Z 軸座標值
	<i>dfPitch</i>	在 Y-Z 平面進行一個整圓運動後，X 軸所移動的距離，此值必須大於 0
	<i>dfU ~ dfB</i>	目的點的 U-V-W-A-B 軸座標值
	<i>dfTheta</i>	圓周運動所移動的角度與運動方向，正值為順時針運動，負值為逆時針運動
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV. 函式傳回值

36. int MCC_HelicalZX_Y_Aux(

double *dfCZ*,
double *dfCX*,
double *dfPitch*,
double *dfU*,
double *dfV*,
double *dfW*,
double *dfA*,

double *dfB*,
double *dfTheta*,
WORD *wGroupIndex*

)

Description	由目前位置執行螺線運動，此運動在 Z-X 平面進行圓周運動，且 U、V、W、A、B 軸以同動方式進行直線運動，可利用 MCC_SetFeedSpeed() 設定此圓周運動的速度。使用此函式必須指定 Z-X 平面上圓周運動的圓心座標值，圓周半徑由目前位置與圓心座標值決定，並須規劃圓周運動移動的角度值。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfCZ, dfCX</i>	圓周運動圓心的 Z-X 軸座標值
	<i>dfPitch</i>	在 Z-X 平面進行一個整圓運動後，Y 軸所移動的距離，此值必須大於 0
	<i>dfU ~ dfB</i>	目的點的 U-V-W-A-B 軸座標值
	<i>dfTheta</i>	圓周運動所移動的角度與運動方向，正值為順時針運動，負值為逆時針運動
Return Value	<i>wGroupIndex</i>	Group 編號
	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV. 函式傳回值

E. 點對點運動

```
1. double MCC_SetPtPSpeed(  
    double dfRatio,  
    WORD wGroupIndex  
)
```

Description	<p>設定進行點對點運動時的速度比例。</p> <p>進行點對點運動時各軸的速度(UU/sec) 等於</p> $((\text{馬達最大轉速}/60) \times \text{導螺桿間隙值}/\text{齒輪箱減速比}) \times (\text{速度比例}/100)$ <p>其中</p> <p>馬達最大轉速(<i>wRPM</i>)、齒輪箱減速比(<i>dfPitch</i>)、齒輪減速比(<i>dfGearRatio</i>)皆定義在機構參數中。</p> <p>因此</p> <p>速度比例的計算可以利用：$(\text{要求的速度}) / (\text{馬達最大轉速能帶動機台的最大安全速度}) \times 100$。</p> <p>注意：點對點運動實際操作時的進給速度須參考是否曾使用 <code>MCC_OverrideSpeed()</code> 設定運動進給速度倍率而定。</p>	
Parameters	<i>dfRatio</i>	速度比例，其值必須大於 0 且小於等於 100
	<i>wGroupIndex</i>	Group 編號
Return Value	實際設定的速度比例	

```
2. double MCC_GetPtPSpeed(  
    WORD wGroupIndex  
)
```

Description	讀取進行點對點運動時使用的速度比例。	
Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	進行點對點運動時使用的速度比例
	小於零	失敗，傳回值意義請參考 IV.函式傳回值

```
3. int MCC_PtP(
```

```
double  dfX0,
double  dfX1,
double  dfX2,
double  dfX3,
double  dfX4,
double  dfX5,
double  dfX6,
double  dfX7,
WORD    wGroupIndex,
DWORD   dwAxisMask
```

)

Description 使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。

Parameters *dfX0, dfX1, dfX2* 目的點的 X-Y-Z 軸座標值
 dfX3, dfX4, dfX5 目的點的 U-V-W 軸座標值
 dfX6, dfX7 目的點的 A-B 軸座標值
 wGroupIndex Group 編號
 dwAxisMask 指定欲發生作用的軸，指定參數可為

EMP_AXIS_X X 軸

EMP_AXIS_Y Y 軸

EMP_AXIS_Z Z 軸

EMP_AXIS_U U 軸

EMP_AXIS_V V 軸

EMP_AXIS_W W 軸

EMP_AXIS_A A 軸

EMP_AXIS_B B 軸

EMP_AXIS_ALL 全部運動軸

以上參數可自由組合，以作用在 X、Z 與 V 軸上為例：

(EMP_AXIS_X | EMP_AXIS_Z | EMP_AXIS_V)

Return Value 大於或等於零 MCCL 給予此運動命令的編碼

小於零

失敗，傳回值的意義請參考 IV.函式傳回值

```
4. int MCC_PtPX(  
    double dfX,  
    WORD wGroupIndex  
)
```

Description	使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfX</i>	目的點的 X 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

```
5. int MCC_PtPY(  
    double dfY,  
    WORD wGroupIndex  
)
```

Description	使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfY</i>	目的點的 Y 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

```
6. int MCC_PtPZ(  
    double dfZ,  
    WORD wGroupIndex  
)
```

Description	使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfZ</i>	目的點的 Z 軸座標值

	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

7. **int** MCC_PtPU(
 double *dfU*,
 WORD *wGroupIndex*
)

Description	使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfU</i>	目的點的 U 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

8. **int** MCC_PtPV(
 double *dfV*,
 WORD *wGroupIndex*
)

Description	使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfV</i>	目的點的 V 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

9. **int** MCC_PtPW(
 double *dfW*,
 WORD *wGroupIndex*
)

Description	使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfW</i>	目的點的 W 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

```
10. int MCC_PtPA(  
    double dfA,  
    WORD wGroupIndex  
)
```

Description	使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfA</i>	目的點的 A 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

```
11. int MCC_PtPB(  
    double dfB,  
    WORD wGroupIndex  
)
```

Description	使用點對點運動的方式，從目前位置以設定的進給速度比例移動至指定的目的點。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dfB</i>	目的點的 B 軸座標值
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

12. int MCC_SetPtPAccType(

```
char    cAccType0,  
char    cAccType1,  
char    cAccType2,  
char    cAccType3,  
char    cAccType4,  
char    cAccType5,  
char    cAccType6,  
char    cAccType7,  
WORD    wGroupIndex
```

)

Description 設定在進行點對點運動時各軸的加速型式。

Parameters	<i>cAccType0</i> ~ <i>cAccType7</i>	各軸的加速型式，可設定為：
		'T' 使用梯型加速曲線
		'S' 使用 S 型加速曲線
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

13. int MCC_GetPtPAccType(

```
char*    pcAccType0,  
char*    pcAccType1,  
char*    pcAccType2,  
char*    pcAccType3,  
char*    pcAccType4,  
char*    pcAccType5,  
char*    pcAccType6,  
char*    pcAccType7,  
WORD    wGroupIndex
```

)

Description 讀取在進行點對點運動時各軸的加速型式。

Parameters *pcAccType0* ~ 指向一 char 值，用來存放各軸的加速型式：



	<i>pcAccType7</i>	0	使用梯型加速曲線
		1	使用 S 型加速曲線
	<i>wGroupIndex</i>	Group 編號	
Return Value	0	成功	
	非零	失敗，傳回值的意義請參考 IV.函式傳回值	

14. int MCC_SetPtPDecType(

```

char  cDecType0,
char  cDecType1,
char  cDecType2,
char  cDecType3,
char  cDecType4,
char  cDecType5,
char  cDecType6,
char  cDecType7,
WORD  wGroupIndex
)

```

Description	設定在進行點對點運動時各軸的減速型式。		
Parameters	<i>cDecType0</i> ~	各軸的減速型式，'T'表示使用梯型減速曲線，'S' 表示	
	<i>cDecType7</i>	使用 S 型減速曲線	
	<i>wGroupIndex</i>	Group 編號	
Return Value	0	成功	
	非零	失敗，傳回值的意義請參考 IV.函式傳回值	

15. int MCC_GetPtPDecType(

```

char*  pcDecType0,
char*  pcDecType1,
char*  pcDecType2,
char*  pcDecType3,
char*  pcDecType4,
char*  pcDecType5,

```

```
char* pcDecType6,  
char* pcDecType7,  
WORD wGroupIndex
```

```
)
```

Description 讀取在進行點對點運動時各軸的減速型式。

Parameters *pcDecType0* ~ 指向一 char 值，用來存放各軸的減速型式；0 表示梯
pcDecType7 型減速曲線，1 表示 S 型減速曲線
wGroupIndex Group 編號

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

16. int MCC_SetPtPAccTime(

```
double dfAccTime0,  
double dfAccTime1,  
double dfAccTime2,  
double dfAccTime3,  
double dfAccTime4,  
double dfAccTime5,  
double dfAccTime6,  
double dfAccTime7,  
WORD wGroupIndex
```

```
)
```

Description 設定在進行點對點運動時各軸加速到穩定速度所需的時間。

Parameters *dfAccTime0* ~ 各軸的加速時間，必須大於 0，單位為 ms
dfAccTime7
wGroupIndex Group 編號

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

17. int MCC_GetPtPAccTime(

```
double* pdfAccTime0,
```

```
double* pdfAccTime1,  
double* pdfAccTime2,  
double* pdfAccTime3,  
double* pdfAccTime4,  
double* pdfAccTime5,  
double* pdfAccTime6,  
double* pdfAccTime7,  
WORD wGroupIndex
```

)

Description	讀取在進行點對點運動時各軸加速到穩定速度所需的時間。	
Parameters	pdfAccTime0 ~	指向一 double 值，用來存放各軸的加速時間，單位
	pdfAccTime7	為 ms
	wGroupIndex	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

18. int MCC_SetPtPDecTime(

```
double dfDecTime0,  
double dfDecTime1,  
double dfDecTime2,  
double dfDecTime3,  
double dfDecTime4,  
double dfDecTime5,  
double dfDecTime6,  
double dfDecTime7,  
WORD wGroupIndex
```

)

Description	設定在進行點對點運動時各軸由穩定速度減速到停止運動所需的時間。	
Parameters	dfDecTime0 ~	各軸的減速時間，必須大於 0，單位為 ms
	dfDecTime7	
	wGroupIndex	Group 編號



Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

19. int MCC_GetPtPDecTime(
 double* *pdfDecTime0*,
 double* *pdfDecTime1*,
 double* *pdfDecTime2*,
 double* *pdfDecTime3*,
 double* *pdfDecTime4*,
 double* *pdfDecTime5*,
 double* *pdfDecTime6*,
 double* *pdfDecTime7*,
 WORD *wGroupIndex*
)

Description	讀取在進行點對點運動時各軸由穩定速度減速到停止運動所需的時間	
Parameters	<i>pdfDecTime0</i> ~ <i>pdfDecTime7</i>	指向一 double 值，用來存放各軸的減速時間，單位為 ms
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

F. JOG 運動

1. int MCC_JogPulse(

```
    int    nPulse,  
    char   cAxis,  
    WORD   wGroupIndex  
)
```

Description 微動(脈衝運動)。在其他運動命令皆已執行完成後(此時呼叫 MCC_GetMotionStatus()所獲得的傳回值應為 GMS_STOP),依照指定的位移量(Pulse 數)及方向帶動特定軸。

此函式為手動程式的細調模式且需處在運動停止狀態時呼叫才有效。脈衝運動並沒有包含加減速的動作,因此給定的位移量不宜過大,避免機台過度震動。

Parameters	<i>nPulse</i>	位移量,單位為 pulse,可給定範圍為-2048~2048
	<i>cAxis</i>	要求進行脈衝運動的運動軸編號(0~7 代表 X~B 軸)
	<i>wGroupIndex</i>	Group 編號

Return Value	0	成功
	非零	失敗,傳回值的意義請參考 IV.函式傳回值

2. int MCC_JogSpace(

```
    double  dfOffset,  
    int     nRatio,  
    char    cAxis,  
    WORD    wGroupIndex  
)
```

Description 吋動(單步運動)。在其他運動命令皆已執行完成後(此時呼叫 MCC_GetMotionStatus()所獲得的傳回值應為 GMS_STOP),依照指定的位移量(增量)及速度比例(與點對點運動的速度比例意義相同)帶動特定軸。成功呼叫此函式將增加運動命令的庫存數目。



Parameters	<i>dfOffset</i>	位移量，單位為 UU
	<i>nRatio</i>	速度比例，其值必須大於 0 且小於等於 100
	<i>cAxis</i>	要求進行脈衝運動的運動軸編號(0~7 代表 X~B 軸)
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

```

3. int MCC_JogConti(
    int  nDir,
    double dfRatio,
    char  cAxis,
    WORD  wGroupIndex
)

```

Description 連續吋動。在其他運動命令皆已執行完成後(此時呼叫 MCC_GetMotionStatus())所獲得的傳回值應為 GMS_STOP)，依照指定的方向及速度比例(與點對點運動的速度比例意義相同)帶動特定軸，並移動到有效工作區間的邊界才停止(機構參數定義了有效工作區間的範圍)。成功呼叫此函式將增加運動命令的庫存數目。

Parameters	<i>nDir</i>	連續吋動的方向，可設定為：
	1	往正方向運動
	-1	往負方向運動
	<i>dfRatio</i>	速度比例，其值必須大於 0 且小於等於 100
	<i>cAxis</i>	要求進行脈衝運動的運動軸編號(0~7 代表 X~B 軸)
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

G. 運動狀態檢視

1. int MCC_GetMotionStatus(WORD *wGroupIndex*)

Description	檢視系統目前的運動狀態。	
Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	0	處於運動狀態(GMS_RUNNING),代表尚有運動命令未執行完成
	1	處於停止狀態(GMS_STOP),已無庫存運動命令
	2	處於暫停狀態(GMS_HOLD),可能因使用者呼叫MCC_HoldMotion
	3	處於延遲狀態(GMS_DELAYING),可能因使用者呼叫MCC_DelayMotion
	其他	失敗,傳回值的意義請參考 IV.函式傳回值

2. int MCC_GetCurCommand(COMMAND_INFO* *pstCurCmdInfo*, WORD *wGroupIndex*)

Description	讀取執行中的運動命令之相關資訊,包括運動命令之類型、運動命令編碼、要求的進給速度與目的點位置等。	
Parameters	<i>pstCurCmdInfo</i>	指向一 COMMAND_INFO 結構,用來存放執行中之運動命令內容,定義如下:

```
typedef struct _COMMAND_INFO
{
    int      nType;
    int      nCommandIndex;
    double   dfFeedSpeed;
    double   dfPos[MAX_AXIS_NUM];
} COMMAND_INFO;
```



其中

nType：運動命令類型

0	點對點運動
1	直線運動
2	順時針圓弧、圓運動
3	逆時針圓弧、圓運動
4	順時針螺線運動
5	逆時針螺線運動
6	運動延遲命令
7	開啟平滑運動
8	關閉平滑運動

nCommandIndex：運動命令編碼

dfFeedSpeed：

一般運動	規劃的進給速度
點對點運動	規劃的速度比例
運動延遲	目前剩餘的延遲時間(單位:ms)

dfPos：目的點的絕對位置座標

	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

3. int MCC_GetCommandCount(

int* ***pnCmdCount***,
WORD ***wGroupIndex***

)

Description	讀取運動命令佇列中尚未執行的運動命令之庫存數目。 關於哪些函式呼叫後會增加庫存命令，請參考”EMP-S 運動控制函式庫使用手冊 2.3 函式庫操作特性”。	
Parameters	<i>pnCmdCount</i>	指向一 int 值，用來存放運動命令庫存數目
	<i>wGroupIndex</i>	Group 編號

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

4. int MCC_ResetCommandIndex(

WORD *wGroupIndex*

)

Description 使運動命令編碼值歸零。運動命令編碼相當於 MCCL 給於每一筆運動命令的識別資料，利用此函式可以運動命令編碼值從 0 開始計數。

Parameters *wGroupIndex* Group 編號

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

5. int MCC_GetCurPulseStockCount(

WORD* *pwStockCount*,

DWORD *dwSlaveId*

)

Description 讀取目前軟體上的 Pulse 命令庫存數目。在運動過程中，利用此函式所獲得的 Pulse 命令庫存數目不應小於 60，如此才能保有穩定的運動控制性能。

Parameters *pwStockCount* 指向一 WORD 值，用來存放 Pulse 命令庫存數目
dwSlaveId EtherCAT 從站(實體軸)的編號

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

6. int MCC_SetMaxPulseStockNum(

int *nMaxStockNum*,

)

Description 設定軟體 FIFO 的使用個數。軟體 FIFO 的使用個數應該搭配作業系統的即時性能一起評估，設定越少的使用個數，作業系統要具備的即時性能越強，設定越多的使用個數，作業系統對即時性能的容忍性越大，對運動控制性能的穩定性越高；如不能達到運動控制性能的穩定性的要求時，請設定較多的使用個數。在呼叫 MCC_InitSystem()後，使用此函式可設定軟體 FIFO 的使用個數。

Parameters *nMaxStockNum* 軟體 FIFO 的最大使用個數，設定範圍為 1 ~ 60

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

7. int MCC_GetMaxPulseStockNum()

Description 讀取軟體 FIFO 的最大使用個數。

Return Value 軟體 FIFO 的最大使用個數

8. int MCC_GetErrorCode(

WORD *wGroupIndex*

)

Description 讀取目前錯誤記錄，用來檢查系統運作時是否發生錯誤。
系統運作中應隨時(例如每 100 ms)呼叫此函式確認系統目前工作正常，若發現錯誤記錄產生，則須採取相對應之錯誤回復(Error Recovery)處理。

Parameters *wGroupIndex* Group 編號

Return Value 0 無錯誤
其它 錯誤代碼(請參考 IV.錯誤訊息代碼)

9. int MCC_ClearError(

WORD *wGroupIndex*

)



Description	在系統運作發生錯誤後，若已排除這些錯誤，必須使用此函式清除系統內的錯誤記錄，否則系統仍無法正常運作。	
Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

H. 定位控制

```
1. int MCC_SetCompParam(  
    SYS_COMP_PARAM* pstCompParam,  
    DWORD dwSlaveId  
)
```

Description	設定齒輪齒隙、間隙補償參數，使用者可先設定補償參數的內容，再利用此函式將補償參數傳入，最後呼叫 MCC_UpdateCompParam()，補償參數的內容必須涵蓋機台全部的工作行程，以避免產生不正常的動作。更詳細的說明請參考” EMP-S 運動控制函式庫使用手冊 2.7.1 齒輪齒隙、背隙補償”。	
Parameters	<i>pstCompParam</i>	指向一 SYS_COMP_PARAM 結構，內含欲設定之補償參數
	<i>dwSlaveId</i>	EtherCAT 從站(實體軸)的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

```
2. int MCC_UpdateCompParam()
```

Description	反應更新後的齒輪齒隙、間隙補償參數。在呼叫過 MCC_SetCompParam()後，須執行此函式系統才會反應新的設定值。	
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

```
3. int MCC_SetMaxPulseSpeed(  
    int nPulse0,  
    int nPulse1,  
    int nPulse2,  
    int nPulse3,  
    int nPulse4,  
    int nPulse5,
```

```
    int  nPulse6,  
    int  nPulse7  
)
```

Description 設定各軸(X ~ B 軸)最大 Pulse 速度的上限。最大 Pulse 速度的上限用來限制一插值單位時間(1 ms)內，各軸能送出的最大 Pulse 數，避免機台的速度超出工作範圍。

Parameters *nPulse0* 各軸最大 Pulse 速度的上限，設定範圍為 1~1000000
~ *nPulse7* ，需視機台特性與插值時間而定

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

4. int MCC_GetMaxPulseSpeed(

```
    int*  pnSpeed0,  
    int*  pnSpeed1,  
    int*  pnSpeed2,  
    int*  pnSpeed3,  
    int*  pnSpeed4,  
    int*  pnSpeed5,  
    int*  pnSpeed6,  
    int*  pnSpeed7  
)
```

Description 讀取各軸(X ~ B 軸)最大 Pulse 速度的上限。

Parameters *pnSpeed0* 指向一 int 值，用來存放各軸最大 Pulse 速度的
~ *pnSpeed7* 上限

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

5. int MCC_SetMaxPulseAcc(

```
    int  nPulseAcc0,  
    int  nPulseAcc1,  
    int  nPulseAcc2,
```



```
int  nPulseAcc3,  
int  nPulseAcc4,  
int  nPulseAcc5,  
int  nPulseAcc6,  
int  nPulseAcc7
```

```
)
```

Description	設定各軸(X~B軸)最大 Pulse 加速度的上限。最大 Pulse 加速度的上限用來限制任連續兩插值時間(1 ms)中，各軸所送出 Pulse 數最大的差異量，可避免機台的加(減)速度超出工作範圍。	
Parameters	<i>nPulseAcc0</i> ~ <i>nPulseAcc7</i>	各軸最大 Pulse 加速度上限，設定範圍為 1~1000000，需視機台特性與插值時間而定
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

6. int MCC_GetMaxPulseAcc(

```
int*  pnPulseAcc0,  
int*  pnPulseAcc1,  
int*  pnPulseAcc2,  
int*  pnPulseAcc3,  
int*  pnPulseAcc4,  
int*  pnPulseAcc5,  
int*  pnPulseAcc6,  
int*  pnPulseAcc7
```

```
)
```

Description	讀取各軸(X~B軸)最大 Pulse 加速度的上限。	
Parameters	<i>pnPulseAcc0</i> ~ <i>pnPulseAcc7</i>	指向一 int 值，存放各軸最大 Pulse 加速度的上限
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

I. 進階軌跡規劃

1. int MCC_HoldMotion(WORD *wGroupIndex*)

Description 暫時停止運動，但必須在運動進行中使用此函式才有意義。呼叫此函式後將減速至停止運動，在減速至停止前，若呼叫 MCC_GetMotionStatus() 所獲得的傳回值仍為 GMS_RUNNING，必須等到運動完全停止後，才會得到 GMS_HOLD 的傳回值。

Parameters *wGroupIndex* Group 編號

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

2. int MCC_ContiMotion(WORD *wGroupIndex*)

Description 繼續執行未完成的運動命令，但必須在運動暫停狀態使用此函式才有意義。

Parameters *wGroupIndex* Group 編號

Return Value 0 成功
非零 失敗，傳回值的意義請參考 IV.函式傳回值

3. int MCC_AbortMotionEx(double *dfDecTime*, WORD *wGroupIndex*)

Description 以設定的減速時間，減速至停止並拋棄後續全部運動命令。呼叫此函式後將減速至停止運動，在減速至停止前，若呼叫 MCC_GetMotionStatus() 所獲得的傳回值仍為 GMS_RUNNING，必須等到運動完全停止後，才會得到 GMS_STOP 的傳回值。注意，**在使用此函式**

之後，必須等到系統進入 **GMS_STOP** 狀態，始可下達後續之運動命令，否則將得到 **ABORT_NOT_FINISH_ERR(-15)** 的回傳值。

Parameters	<i>dfDecTime</i>	要求減速的時間
	<i>wGroupIndex</i>	Group 編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

4. **int MCC_EnableBlend(** **WORD *wGroupIndex*** **)**

Description 開啟平滑運動功能。呼叫此函式後，以連續路徑方式進行軌跡規劃。
成功呼叫此函式將增加運動命令的庫存數目。

Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

5. **int MCC_DisableBlend(** **WORD *wGroupIndex*** **)**

Description 關閉平滑運動功能。成功呼叫此函式將增加運動命令的庫存數目。

Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

6. **int MCC_CheckBlend(** **WORD *wGroupIndex*** **)**

Description 檢查是否開啟平滑運動功能。

Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	0	已開啟平滑運動功能

1	未開啟平滑運動功能
其他	失敗，傳回值的意義請參考 IV.函式傳回值

**7. int MCC_DelayMotion(
 DWORD *dwTime*,
 WORD *wGroupIndex*
)**

Description	設定運動延遲時間，強迫延遲執行下一個運動命令。成功呼叫此函式將增加運動命令的庫存數目。	
Parameters	<i>dwTime</i>	延遲時間，單位為 ms
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

**8. int MCC_CheckDelay(
 WORD *wGroupIndex*
)**

Description	檢查目前是否進入運動延遲狀態（此時若呼叫 MCC_GetMotionStatus()將得到 GMS_DELAYING 的回傳值）。	
Parameters	<i>wGroupIndex</i>	Group 編號
Return Value	0	未處於運動延遲狀態
	1	處於運動延遲狀態
	其他	失敗，傳回值的意義請參考 IV.函式傳回值

**9. int MCC_TimeDelay(
 DWORD *dwTime*
)**

Description	設定運動延遲時間，強迫延遲執行下一個運動命令。	
Parameters	<i>dwTime</i>	延遲時間，單位為 ms
Return Value	0	成功

非零

失敗，傳回值的意義請參考 IV.函式傳回值

10. double MCC_OverrideSpeed(

double *dfRate*,

WORD *wGroupIndex*

)

Description	設定一般運動與點對點運動的速度強制比例，使用此項函式將即時變動一般運動與點對點運動的進給速度。	
Parameters	<i>dfRate</i>	速度強制比例，為原來進給速度的多少百分比再乘以 100，也就是說一般運動新的進給速度將等於 $(dfFeedSpeed \times dfRate / 100)$ ， <i>dfFeedSpeed</i> 為原來使用 <i>MCC_SetFeedSpeed()</i> 所設定的進給速度，而點對點運動速度等於 $(速度比例 \times dfRate / 100)$ 。 設定值必須大於 0；若更新後的進給速度超過 <i>MCC_SetSysMaxSpeed()</i> 的設定值，則新的進給速度將等於 <i>MCC_SetSysMaxSpeed()</i> 的設定值。
	<i>wGroupIndex</i>	Group 編號
Return Value	大於零	實際設定的速度強制比例
	其他	失敗，傳回值的意義請參考 IV.函式傳回值

11. double MCC_OverrideSpeedEx(

double *dfRate*,

BOOL *bInstant*,

WORD *wGroupIndex*

)

Description	設定一般運動與點對點運動的速度強制比例，使用此項函式將即時變動一般運動的進給速度。	
Parameters	<i>dfRate</i>	速度強制比例，為原來進給速度的多少百分比再乘以 100，也就是說一般運動新的進給速度將等於 $(dfFeedSpeed \times dfRate / 100)$ ， <i>dfFeedSpeed</i> 為原來使用



MCC_SetFeedSpeed()所設定的進給速度。

設定值必須大於 0；若更新後的進給速度超過 MCC_SetSysMaxSpeed() 的設定值，則新的進給速度將等於 MCC_SetSysMaxSpeed() 的設定值。

bInstant 更新速度時機，1 代表立即更新速度，0 代表下一筆命令才更新速度

wGroupIndex Group 編號

Return Value 大於零 實際設定的速度強制比例
其他 失敗，傳回值的意義請參考 IV.函式傳回值

12. double MCC_GetOverrideRate(

WORD *wGroupIndex*

)

Description 讀取一般運動目前使用的速度強制比例。

Parameters *wGroupIndex* Group 編號

Return Value 大於零 一般運動目前使用的速度強制比例
其他 失敗，傳回值的意義請參考 IV.函式傳回值

J. EtherCAT原點復歸

1. int MCC_EcatSetHomeAxis(

BYTE *byAxisX*, BYTE *byAxisY*, BYTE *byAxisZ*, BYTE *byAxisU*,
BYTE *byAxisV*, BYTE *byAxisW*, BYTE *byAxisA*, BYTE *byAxisB*,
BYTE *byAxisX1*, BYTE *byAxisY1*, BYTE *byAxisZ1*, BYTE *byAxisU1*,
BYTE *byAxisV1*, BYTE *byAxisW1*, BYTE *byAxisA1*, BYTE *byAxisB1*
)

Description 設定原點復歸的運動軸。

Parameters *byAxisX*~ 設定對應各運動軸執行原點復歸的功能，
byAxisB1 1 表示該運動軸執行原點復歸，0 則否

Return Value 1 成功
0 失敗

2. int MCC_EcatSetHomeMode(

int *nMode*,
DWORD *dwSlaveId*
)

Description 設定原點復歸模式。

Parameters *nMode* 原點復歸模式，請依照 Home Sensor、實際配線與搭配的 EtherCAT 伺服驅動器而定(請參考 OD 碼 0x6098h)
dwSlaveId EtherCAT 從站(運動軸)的編號

Return Value 1 成功
0 失敗

3. int MCC_EcatSetHomeZeroSpeed(

int *nZeroSpeed*,
DWORD *dwSlaveId*
)

Description 設定原點復歸時尋找零點(Zero)的速度。

Parameters	<i>nZeroSpeed</i>	尋找零點(Zero)的速度
	<i>dwSlaveId</i>	EtherCAT 從站(運動軸)的編號
Return Value	1	成功
	0	失敗

4. **int MCC_EcatSetHomeSwitchSpeed(** **int *nSwitchSpeed*,** **DWORD *dwSlaveId*** **)**

Description	設定原點復歸時尋找原點開關(Switch)的速度。	
Parameters	<i>nSwitchSpeed</i>	尋找原點開關(Switch)的速度
	<i>dwSlaveId</i>	EtherCAT 從站(運動軸)的編號
Return Value	1	成功
	0	失敗

5. **int MCC_EcatHome()**

Description	執行原點復歸運動。使用此函式時，可配合呼叫 <code>MCC_EcatGetGoHomeStatus()</code> ，來檢查原點復歸是否完成。當完成原點復歸的運動後各軸的直角座標值將被設定為零。	
Return Value	1	成功
	0	失敗

6. **BOOL MCC_GetGoHomeStatus()**

Description	在呼叫 <code>MCC_EcatHome()</code> 後，使用此函式可檢視是否已完成原點復歸動作。	
Return Value	1	完成原點復歸動作
	0	尚未完成原點復歸動作

7. BOOL MCC_EcatAbortHome()

Description 當呼叫 MCC_EcatHome()後，使用此函式可停止原點復歸的動作。

Return Value	1	成功
	0	失敗

8. int MCC_GetHomeSensorStatus(

WORD* *pwStatus*,

DWORD *dwSlaveId*

)

Description 讀取 Home Sensor 的狀態，使用此函式前須先正確定義 Home Sensor 的配線方式(Normal Open 或 Normal Close)與該從站的參數設定一致。

Parameters	<i>pwStatus</i>	指向一 WORD 值，用來存放欲讀取 Home Sensor 的狀態，請參考該從站的 OD 碼 0x60FDh 的 bit 2
------------	-----------------	---

	<i>dwSlaveId</i>	EtherCAT 從站(運動軸)的編號
--	------------------	---------------------

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

EtherCAT 原點復歸相關函式使用順序：

1. MCC_EcatSetHomeAxis()
2. MCC_EcatSetHomeMode()
3. MCC_EcatSetHomeZeroSpeed()
4. MCC_EcatSetHomeSwitchSpeed()
5. MCC_EcatHome()

可以使用 MCC_GetEcatGoHomeStatus() 取得目前原點復歸的狀態，可以使用 MCC_EcatAbortHome() 停止原點復歸的動作。

K. EtherCAT控制功能

1. int MCC_SetServoOn(

DWORD *dwSlaveId*

)

Description	開啟伺服系統。	
Parameters	<i>dwSlaveId</i>	EtherCAT 從站(實體軸)的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

2. int MCC_SetServoOff(

DWORD *dwSlaveId*

)

Description	關閉伺服系統。	
Parameters	<i>dwSlaveId</i>	EtherCAT 從站(實體軸)的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

3. DWORD MCC_EcatGetMasterErrorCode()

Description	取得 EtherCAT 主站的錯誤碼(Error Code)。	
Return Value	0x229	EtherCAT 從站處於錯誤(Error)狀態
	0x252	EtherCAT 從站處於離線狀態
	0x253	EtherCAT 從站處於 Recover 狀態

4. DWORD MCC_EcatClearMasterErrorCode()

Description	清除取得 EtherCAT 主站的錯誤碼(Error Code)。	
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

5. WORD MCC_EcatGetSlaveErrorCode(

	WORD* <i>pwType</i> ,	
	WORD* <i>pwCode</i> ,	
	DWORD <i>dwSlaveId</i>	
)	
Description	取得 EtherCAT 從站的錯誤碼(Error Code)，請參閱該從站的 OD 碼 0x6041h 與 0x603Fh 的說明。	
Parameters	<i>pwType</i>	錯誤碼層級
		0 無錯誤碼
		1 Fault 層級(OD 碼 0x6041h 的 bit 3)
		2 Warning 層級(OD 碼 0x6041h 的 bit 7)
	<i>pwCode</i>	錯誤碼(OD 碼 0x603Fh)
	<i>dwSlaveId</i>	EtherCAT 從站的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

6. **WORD** MCC_EcatResetSlaveFault(

DWORD *dwSlaveId*

)

Description	重置 EtherCAT 從站的錯誤(Fault reset)，請參閱該從站的 OD 碼 0x6040h 的 bit 7 的說明。	
Parameters	<i>dwSlaveId</i>	EtherCAT 從站的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

7. **DWORD** MCC_EcatGetSlavePresent(

DWORD *dwSlaveId*,

BOOL* *pbStatus*

)

Description	取得 EtherCAT 從站的連線狀態。	
Parameters	<i>dwSlaveId</i>	EtherCAT 從站的編號

	<i>pbStatus</i>	EtherCAT 從站的連線狀態；1 表示該從站處於連線狀態，0 則否
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

8. BOOL MCC_EcatEnableWatchDog(

int *nExpireTime*

)

Description 開啟軟體 Watch Dog 功能，必須在 MCC_RTXInit()之後呼叫。EcServer 會判斷人機介面的程式在 *nExpireTime* 時間內無回應後，自動終止 EcServer。

Parameters *nExpireTime* Watch Dog 的倒數計時時間，單位為 ms

Return Value 1 成功

0 失敗，MCC_RTXInit()尚未呼叫

9. BOOL MCC_EcatDisableWatchDog()

Description 關閉軟體 Watch Dog 功能，必須在 MCC_EcatEnableWatchDog()之後呼叫。

Return Value 1 成功

0 失敗，未呼叫 MCC_EcatEnableWatchDog()

10. int MCC_EcatGetInput(

DWORD *dwSlaveId*,

DWORD* *pdwInData*

)

Description 讀取 EtherCAT 輸出入控制卡(ECATS-IO32)的 16 點輸入訊號值。

Parameters *dwSlaveId* EtherCAT 從站的編號

pdwInData 指向一 DWORD 值，用來存放 16 點的輸出訊號狀態(bit 0 ~ bit 15 分別代表第 0 點到第 15 點的狀態)

Return Value 1 成功

0 失敗

11. int MCC_EcatGetOutput(

DWORD *dwSlaveId*,

DWORD* *pdwOutData*

)

Description 讀取 EtherCAT 輸出入控制卡(ECATS-IO32)的 16 點輸出訊號值。

Parameters *dwSlaveId* EtherCAT 從站的編號
pdwOutData 指向一 DWORD 值，用來存放 16 點的輸出訊號狀態(bit 0 ~ bit 15 分別代表第 0 點到第 15 點的狀態)

Return Value 1 成功
0 失敗

12. int MCC_EcatSetOutput(

DWORD *dwSlaveId*,

DWORD *dwOutData*

)

Description 設定 EtherCAT 輸出入控制卡(ECATS-IO32)的 16 點輸出訊號值。

Parameters *dwSlaveId* EtherCAT 從站的編號
dwOutData 指定 16 點的輸出訊號狀態(bit 0 ~ bit 15 分別代表第 0 點到第 15 點的狀態)

Return Value 1 成功
0 失敗

13. int MCC_EcatSetOutputEnqueue(

DWORD *dwSlaveId*,

DWORD *dwOutData*,

int *nType*,

WORD *wGroupIndex*

)

Description 設定 EtherCAT 輸出入控制卡(ECATS-IO32)的 16 點輸出訊號值存放至運動命令佇列(Queue)中。成功呼叫此函式將增加運動命令的庫存數目。

Parameters	<i>dwSlaveId</i>	EtherCAT 從站的編號
	<i>dwOutData</i>	指定 16 點輸出訊號狀態(bit 0 ~ bit15 分別代表第 0 點第 15 點的狀態)，可同時輸出
	<i>nType</i>	命令與命令間的輸出時機
	<i>wGroupIndex</i>	Group 編號
Return Value	大於或等於零	MCCL 給予此運動命令的編碼
	小於零	失敗，傳回值的意義請參考 IV.函式傳回值

14. int MCC_EcatGetADCInput(

DWORD *dwSlaveId*,
WORD *wChannel*,
float* *pfInput*

)

Description	讀取指定 EtherCAT 類比電壓輸入控制卡 (ECATS-ADC) 的輸入直流電壓值。	
Parameters	<i>dwSlaveId</i>	EtherCAT 從站(ECATS-ADC)的編號
	<i>wChannel</i>	ADC Channel 的編號(0 ~ 7)
	<i>pfInput</i>	指向一 float 值，用來存放 ADC Channel 直流電壓輸入值，輸入電壓的有效值為-10V ~ +10V
Return Value	1	成功
	0	失敗

15. int MCC_EcatCoeSdoUpload(

DWORD *dwSlaveId*,
WORD *wObIndex*,
BYTE *byObSubIndex*,

	BYTE* <i>pbyData</i> ,	
	DWORD <i>dwDataLen</i> ,	
	DWORD* <i>pdwOutDataLen</i>	
)	
Description	透過標準的 CoE 通訊指令，對指定的從站下達 CANopen 協議的 SDO(Service Data Objects)讀取指令。	
Parameters	<i>dwSlaveId</i>	EtherCAT 從站的編號
	<i>wObIndex</i>	CANopen 物件字典的索引值(Index)
	<i>byObSubIndex</i>	CANopen 物件字典的子索引值(Sub Index)
	<i>pbyData</i>	指向一 BYTE 值，用來存放欲讀取訊息的資料指標
	<i>dwDataLen</i>	欲讀取訊息的資料量大小，單位為 byte
	<i>pdwOutDataLen</i>	指向一 DWORD 值，用來存放實際讀取訊息的資料長度
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

16. int MCC_EcatCoeSdoDownload(

DWORD *dwSlaveId*,
WORD *wObIndex*,
BYTE *byObSubIndex*,
BYTE* *pbyData*,
DWORD *dwDataLen*

)

Description	透過標準的 CoE 通訊指令，對指定的從站下達 CANopen 協議的 SDO(Service Data Objects)傳送指令。	
Parameters	<i>dwSlaveId</i>	EtherCAT 從站的編號
	<i>wObIndex</i>	CANopen 物件字典的索引值(Index)
	<i>byObSubIndex</i>	CANopen 物件字典的子索引值(Sub Index)
	<i>pbyData</i>	指向一 BYTE 值，用來存放欲傳送訊息的資料指標
	<i>dwDataLen</i>	欲傳送訊息的資料量大小，單位為 byte

Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

17. int MCC_EcatGetMotorPositionRaw(

double* *pdfPosition*,

DWORD *dwSlaveId*

)

Description 讀取指定實體軸目前馬達位置值，請參閱該從站的 OD 碼 0x6064 的說明。

Parameters *pdfPosition* 指向一 double 值，用來存放馬達位置值

dwSlaveId EtherCAT 從站的編號

Return Value 0 成功

非零 失敗，傳回值的意義請參考 IV.函式傳回值

18. int MCC_EcatGetMotorDemandPositionRaw(

double* *pdfPosition*,

DWORD *dwSlaveId*

)

Description 讀取指定實體軸目前馬達命令位置值，請參閱該從站的 OD 碼 0x6062 的說明。

Parameters *pdfPosition* 指向一 double 值，用來存放馬達命令位置值

dwSlaveId EtherCAT 從站的編號

Return Value 0 成功

非零 失敗，傳回值的意義請參考 IV.函式傳回值

19. int MCC_EcatGetMotorVelocityRaw(

int* *pnSpeed*,

DWORD *dwSlaveId*

)

Description 讀取指定實體軸目前馬達實際速度值，請參閱該從站的 OD 碼 0x606C 的說明。

Parameters	<i>pnSpeed</i>	指向一 int 值，用來存放馬達實際速度值
	<i>dwSlaveId</i>	EtherCAT 從站的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

20. int MCC_EcatGetMotorTorqueRaw(

int* *pnTorque*,
DWORD *dwSlaveId*
)

Description	讀取指定實體軸目前馬達實際扭矩值，請參閱該從站的 OD 碼 0x6077h 的說明。	
Parameters	<i>pnTorque</i>	指向一 int 值，用來存放馬達實際扭矩值
	<i>dwSlaveId</i>	EtherCAT 從站的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

21. int MCC_EcatGetMotorDemandVelocityRaw(

double* *pdfSpeed*,
DWORD *dwSlaveId*
)

Description	讀取指定實體軸目前馬達命令速度值，請參閱該從站的 OD 碼 0x606B 的說明。	
Parameters	<i>pdfSpeed</i>	指向一 double 值，用來存放馬達命令速度值
	<i>dwSlaveId</i>	EtherCAT 從站的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

22. int MCC_EcatGetMotorDemandTorqueRaw(

double* *pdfTorque*,
DWORD *dwSlaveId*
)

Description	讀取指定實體軸目前馬達的命令扭矩值，請參閱該從站的 OD 碼 0x6074 的說明	
Parameters	<i>pdfTorque</i>	指向一 double 值，用來存放馬達命令扭矩值
	<i>dwSlaveId</i>	EtherCAT 從站的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

23. int MCC_EcatGetMotorRatedTorque(

DWORD* *pdwTorque*,

DWORD *dwSlaveId*

)

Description	讀取指定實體軸馬達的額定扭矩值，請參閱該從站的 OD 碼 0x6076h 的說明。	
Parameters	<i>pdwTorque</i>	指向一 WORD 值，用來存放馬達的額定扭矩值
	<i>dwSlaveId</i>	EtherCAT 從站的編號
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值

24. int MCC_EcatLoadCFGFilePath(

char* *pCfgFilePath*

)

Description	設定 EtherCAT 組態設定檔的存放目錄路徑與檔名。	
Parameters	<i>pCfgFilePath</i>	指向一 char*，用來存放檔案路徑與檔名
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值 2. EtherCAT Preset 函式傳回值與說明

25. int MCC_EcatSetESIPath(

char* *pEsiPath*

)

Description	設定 ESI 檔案的存放目錄路徑。	
Parameters	<i>pEsiPath</i>	指向一 char*，用來存放目錄路徑
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值 2. EtherCAT Preset 函式傳回值與說明

26. int MCC_EcatLoadESIFilename(

char* *pEsiFileName*

)

Description	設定 ESI 檔案的檔案名稱(含副檔名)。	
Parameters	<i>pEsiFilePath</i>	指向一 char*，用來存放檔案名稱
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值 2. EtherCAT Preset 函式傳回值與說明

27. int MCC_EcatSetENIPath(

char* *pEniPath*

)

Description	設定 ENI 檔案的存放目錄路徑。	
Parameters	<i>pEniPath</i>	指向一 char*，用來存放目錄路徑
Return Value	0	成功
	非零	失敗，傳回值的意義請參考 IV.函式傳回值 2. EtherCAT Preset 函式傳回值與說明

28. int MCC_EcatSetENIFilename(

char* *pEniFileName*

)

Description	設定 ENI 檔案的檔案名稱(不含副檔名)。	
Parameters	<i>pEniFileName</i>	指向一 char*，用來存放檔案名稱
Return Value	0	成功

非零 失敗，傳回值的意義請參考 **IV.函式傳回值 2. EtherCAT**
Preset 函式傳回值與說明

29. int MCC_EcatScanBus()

Description 開始掃描 EtherCAT 網路拓模狀態。

Return Value 0 成功
非零 失敗，傳回值的意義請參考 **IV.函式傳回值 2. EtherCAT**
Preset 函式傳回值與說明

30. int MCC_EcatGetSlaveInfoByID(SlaveInfoPre* *pstSlaveInfoPre*, DWORD *dwSlaveId*)

Description 取得指定 EtherCAT 從站的資訊。

Parameters *pstSlaveInfoPre* 指向一 SlaveInfoPre 結構，用來存放 EtherCAT 從站的資訊
dwSlaveId EtherCAT 從站的編號

Return Value 0 成功
非零 失敗，傳回值的意義請參考 **IV.函式傳回值 2. EtherCAT**
Preset 函式傳回值與說明

31. int MCC_EcatClearSlaveInfo()

Description 清除 EtherCAT 從站的資訊。

Return Value 0 成功
非零 失敗，傳回值的意義請參考 **IV.函式傳回值 2. EtherCAT**
Preset 函式傳回值與說明

32. int MCC_EcatGenerateENI()

Description 由使用者依照一定程序產生自訂 ENI 檔案。

Return Value 0 成功



非零

失敗，傳回值的意義請參考 **IV.函式傳回值 2. EtherCAT**
Preset 函式傳回值與說明

33. int MCC_EcatAutoGenerateENI(

 BOOL* *bEnable*,

)

Description 由系統自動產生 ENI 檔案(C:\MccMotion\ENIFile.xml)。此函式
 必須在 MCC_InitSystem()之前先呼叫。

Parameters *bEnable* true
 false

Return Value 0 成功
 非零 失敗，傳回值的意義請參考 **IV.函式傳回值 2. EtherCAT**
 Preset 函式傳回值與說明

III. 錯誤訊息代碼

錯誤代碼	說明
0xF101	尚未初使化運動控制函式庫
0xF102	緊急停止
0xF104	在使用圓弧運動命令時給定的參數不合理
0xF203	進給速度太快，超過每個插值時間內允許輸出的 Pulse 數
0xF204	進給加速度太快，超過每個插值時間內允許輸出的 Pulse 增量數
0xF301	X 軸座標值超出機構參數設定的工作範圍
0xF302	Y 軸座標值超出機構參數設定的工作範圍
0xF303	Z 軸座標值超出機構參數設定的工作範圍
0xF304	U 軸座標值超出機構參數設定的工作範圍
0xF305	V 軸座標值超出機構參數設定的工作範圍
0xF306	W 軸座標值超出機構參數設定的工作範圍
0xF307	A 軸座標值超出機構參數設定的工作範圍
0xF308	B 軸座標值超出機構參數設定的工作範圍
0xF401	運動命令在執行間發生錯誤
0xF501	定位確認錯誤
0xF701	X 軸碰觸硬體極限開關
0xF702	Y 軸碰觸硬體極限開關
0xF703	Z 軸碰觸硬體極限開關
0xF704	U 軸碰觸硬體極限開關
0xF705	V 軸碰觸硬體極限開關
0xF706	W 軸碰觸硬體極限開關
0xF707	A 軸碰觸硬體極限開關
0xF708	B 軸碰觸硬體極限開關

IV. 函式傳回值

1. 一般函式傳回值與說明

傳 回 值 定 義	數 值	說 明
NO_ERR	0	函式呼叫成功
INITIAL_MOTION_ERR	-1	系統尚未啟動，請再次呼叫 MCC_InitSystem()
COMMAND_BUFFER_FULL_ERR	-2	運動命令佇列已滿，此時無法接受此筆命令。
COMMAND_NOTACCEPTED_ERR	-3	系統處於忙碌狀態，此時無法接受此筆命令。
COMMAND_NOTFINISHED_ERR	-4	執行中的運動命令尚未完成，此時無法接受此筆命令。
PARAMETER_ERR	-5	呼叫函式時所傳入的參數格式錯誤
GROUP_PARAMETER_ERR	-6	Group 參數給定錯誤，所指定為無效的 Group
FEED_RATE_ERR	-7	進給速度未設定或設定錯誤，請重新呼叫 MCC_SetFeedSpeed()
VOLTAGE_COMMAND_NOTCALLED_ERR	-9	因系統或此運動軸使用 V Command 操作模式，限制使用此函式
HOME_COMMAND_NOTCALLED_ERR	-10	目前並不在原點復歸模式下
HOLD_ILLEGAL_ERR	-11	不適當時機發出暫停(Hold)命令
CONTI_ILLEGAL_ERR	-12	不適當時機發出繼續(Continue)命令
ABORT_ILLEGAL_ERR	-13	不適當時機使用棄置(Abort)命令
RUN_TIME_ERR	-14	執行時期產生錯誤，利用呼叫 MCC_GetErrorCode()所獲得的錯誤訊息代碼可瞭解錯誤的內容
ABORT_NOT_FINISH_ERR	-15	命令棄置動作尚未完成
GROUP_RAN_OUT_ERR	-16	已無多餘 Group 可使用



2. EtherCAT Preset 函式傳回值與說明

傳 回 值 定 義	數 值	說 明
PRESET_NO_ERROR	0	函式呼叫成功
PRESET_ESI_NOTFOUND_ERR	-1	沒有找到 ESI 檔
PRESET_SLAVE_NOTFOUND_ERR	-2	Scan Bus 沒有找到任何 slave
PRESET_ESI_TOSLAVE_ERR	-3	ESI 檔案與 Scan 的 slave 不匹配
PRESET_ENI_FOLDER_NOT_FOUND_ERR	-4	未找到 ENI 產生的路徑
PRESET_SLAVEINFO_NOT_GET	-5	Slave 資訊無法取得
PRESET_ADAPTER_ERR	-6	EC_Config 當中選擇的網路卡錯誤
PRESET_ERR_TYPE_SDO_ERROR	-7	SDO 傳送錯誤
PRESET_ERR_FOE_INPUT_FILE_ERROR	-8	FOE 輸入的檔名錯誤
PRESET_ERR_ENI_FILE_NOT_FOUND	-9	找不到 ENI 檔案
PRESET_ERR_ENI_TYPE_ERROR	-10	ENI 當中 slave 的種類錯誤
PRESET_ERR_ENI_DRIVE_COUNT_ERROR	-11	ENI 中驅動器數量錯誤
PRESET_TOO_ESIFILE	-12	ESI 檔案數量太多
PRESET_CFG_FILE_NOT_FOUND_ERR	-13	找不到 EC_Config.xml
PRESET_ERR_SCANBUS_ERROR	-14	Scan Bus 發生錯誤

3. EtherCAT 函式傳回值與說明

函式傳回值	說 明
0x98110001	Feature not supported
0x98110002	Invalid Index
0x98110003	Invalid Offset
0x98110004	Cancel
0x98110005	Invalid Size
0x98110006	Invalid Data
0x98110007	Not ready
0x98110008	Busy
0x98110009	Cannot queue acyclic ecat command
0x9811000A	No Memory left
0x9811000B	Invalid Parameter
0x9811000C	Not Found
0x9811000D	Duplicated fixed address detected
0x9811000E	Invalid State
0x9811000F	Cannot add slave to timer list
0x98110010	Timeout
0x98110011	Open Failed
0x98110012	Send Failed
0x98110013	Insert Mailbox error
0x98110014	Invalid Command
0x98110015	Unknown Mailbox Protocol Command
0x98110016	Access Denied
0x9811001A	Product Key Invalid
0x9811001B	Wrong configuration format
0x9811001C	Feature disabled
0x9811001E	Bus Config Mismatch
0x9811001F	Error reading config file



0x98110021	Cyclic commands are missing
0x98110022	AL_STATUS register read missing in XML file for at least one state
0x98110023	Fatal internal McSm
0x98110024	Slave error
0x98110025	Frame lost, IDX mismatch
0x98110026	At least one EtherCAT command is missing in the received frame
0x98110028	IOCTL_EC_IOCTL_DC_LATCH_REQ_LTIMVALS invalid in DCL auto read mode
0x98110029	Auto increment address increment mismatch
0x9811002A	Slave in invalid state, e.g. not in OP (API not callable in this state)
0x9811002B	Station address lost (or slave missing) - FPRD to AL_STATUS failed
0x9811002C	Too many cyclic commands in XML configuration file
0x9811002D	Ethernet link cable disconnected
0x9811002E	Master core not accessible
0x9811002F	COE mbox send: working counter
0x98110030	COE mbox receive: working counter
0x98110031	No mailbox support
0x98110032	CoE protocol not supported
0x98110038	Evaluation Time limit reached
0x98110070	Master configuration not found
0x98110071	Command error while EEPROM upload
0x98110072	Command error while EEPROM download
0x98110073	Cyclic command wrong size (too long)
0x98110076	Port Close failed
0x98110077	Port Close failed
0x9811010e	Command not executed (slave not present on bus)
0x98110110	Command error while EEPROM reload
0x98110111	Command error while Reset Slave Controller

0x98110112	Cannot open system driver
0x9811011E	Bus configuration not detected, Topology changed
0x98110123	VoE mailbox send: working counter
0x98110124	EEPROM assignment failed
0x98110125	Error mailbox received
0x98110126	Redundancy line break
0x98110127	Invalid EtherCAT cmd in cyclic frame with redundancy
0x98110128	<PreviousPort>-tag is missing
0x98110129	DC is enabled and DC cyclic commands are missing (e.g. access to 0x900)
0x98110130	Data link (DL) status interrupt because of changed topology
0x98110131	Pass Through Server is not running
0x98110132	ADS adapter (Pass Through Server) is running
0x98110133	Could not start the Pass Through Server
0x98110134	The Pass Through Server could not bind the IP address with a socket
0x98110135	The Pass Through Server is running but not enabled
0x98110136	The Link Layer mode is not supported by the Pass Through Server
0x98110137	No VoE mailbox received
0x98110138	SYNC out unit of reference clock is disabled
0x98110139	Reference clock not found!
0x9811013B	Mailbox command working counter error
0x9811013C	AoE: Protocol not supported
0x9811016E	AoE: Invalid NetID
0x9811016F	Error: Maximum number of bus slave has been exceeded
0x98110170	Mailbox error: Syntax of 6 octet Mailbox header is wrong
0x98110171	Mailbox error: The Mailbox protocol is not supported
0x98110172	Mailbox error: Field contains wrong value
0x98110173	Mailbox error: The mailbox protocol header of the mailbox protocol is wrong



0x98110174	Mailbox error: The mailbox protocol header of the mailbox protocol is wrong
0x98110175	Mailbox error: Length of received mailbox data is too short
0x98110176	Mailbox error: Mailbox protocol can not be processed because of limited resources
0x98110177	Mailbox error: The length of data is inconsistent
0x98110178	Slaves with DC configured present on bus before reference clock
0x9811017B	Line crossed

4. SDO 函式傳回值與說明

函式傳回值	說 明
0x98110040	Toggle bit not alternated
0x98110041	Protocol timed out
0x98110042	Client/server command specifier not valid or unknown
0x98110043	Invalid block size (block mode only)
0x98110044	Invalid sequence number (block mode only)
0x98110045	CRC error (block mode only)
0x98110046	Out of memory
0x98110047	Unsupported access to an object
0x98110048	Attempt to read a write only object
0x98110049	Attempt to write a read only object
0x98130004	Sub index cannot be written, SIO must be 0 for write access
0x98130005	Complete access not supported for objects of cvariable length suach as ENUM object types
0x98130006	Object length exceeds mailbox size
0x98130007	Object mapped to RxPDO, SDO Download blocked
0x9811004A	Object does not exist in the object dictionary
0x9811004B	Object cannot be mapped to the PDO
0x9811004C	The number and length of the objects to be mapped would exceed PDO length
0x9811004D	General parameter incompatibility reason
0x9811004E	General internal incompatibility in the device
0x9811004F	Access failed due to an hardware error
0x98110050	Data type does not match, length of service parameter does not match
0x98110051	Data type does not match, length of service parameter too high
0x98110052	Data type does not match, length of service parameter too low
0x98110053	Sub index does not exist
0x98110054	Value range of parameter exceeded (only for write access)



0x98110055	Value of parameter written too high
0x98110056	Value of parameter written too low
0x98110057	Maximum value is less than minimum value
0x98110058	General error
0x98110059	Data cannot be transferred or stored to the application
0x9811005A	Data cannot be transferred or stored to the application because of local control
0x9811005B	Data cannot be transferred or stored to the application because
0x9811005C	Object dictionary dynamic generation fails or no object dictionary is present (e.g. object dictionary is generated from file and generation fails because of a file error)
0x9811005D	Unknown code

V. 運動控制函式庫初始設定

下表所列出的為呼叫 MCC_InitSystem()後，MCCL 的初始設定，初始設定若無法滿足使用者的需要，可呼叫相關函式加以更改。

初始設定內容	初 始 設 定	相 關 函 式
運動命令佇列大小	10000 筆命令	MCC_CreateGroupEx() MCC_GetCmdQueueSize()
機台允許的的最大進給速度	100	MCC_SetSysMaxSpeed() MCC_GetSysMaxSpeed()
系統座標型態	絕對座標	MCC_SetAbsolute() MCC_SetIncrease() MCC_GetCoordType()
各軸允許的最大 Pulse 加速度	1000000	MCC_SetMaxPulseAcc() MCC_GetMaxPulseAcc()
各軸允許的最大 Pulse 速度	1000000	MCC_SetMaxPulseSpeed() MCC_GetMaxPulseSpeed()
軟體過行程檢查	未開啟	MCC_SetOverTravelCheck() MCC_GetOverTravelCheck
硬體極限開關檢查	未開啟	MCC_EnableLimitSwitchCheck() MCC_DisableLimitSwitchCheck()
進行直線、圓弧、圓與螺線運動時使用的加、減速型式	S 型曲線	MCC_SetAccType() MCC_GetAccType() MCC_SetDecType() MCC_GetDecType()
進行直線、圓弧、圓與螺線運動時使用的加、減速時間	300 ms	MCC_SetAccTime() MCC_GetAccTime() MCC_SetDecTime() MCC_GetDecTime()
進行直線、圓弧、圓與螺線運動時使用的進給速度	1	MCC_SetFeedSpeed() MCC_GetFeedSpeed()
進行點對點運動時各軸使用的速度比例	1	MCC_SetPtPSpeed() MCC_GetPtPSpeed()
平滑運動功能	未開啟	MCC_EnableBlend() MCC_DisableBlend()