



EPCIO Series

驅動函式庫

參考手冊

版本：V.6.00

日期：2020.06

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



目 錄

1. EPCIO Series 驅動函式庫簡介	7
2. EPCIO Series 驅動函式庫功能	8
2.1. 運動控制卡初始化、中斷及重置功能(系統功能)	8
2.1.1 EPCIO4000_Init()	8
2.1.2 EPCIO6000_Init()	9
2.1.3 EPCIO6000e_Init()	10
2.1.4 EPCIO_SetISRFunction()	11
2.1.5 EPCIO_Close()	12
2.1.6 EPCIO_ResetModule()	12
2.1.7 EPCIO_GetIntSource()	13
2.2. 脈波輸出控制(DDA)	15
2.2.1 EPCIO_DDA_GetIntCondition()	15
2.2.2 EPCIO_DDA_GetCurrentCmd()	16
2.2.3 EPCIO_DDA_CheckFIFOEmpty()	16
2.2.4 EPCIO_DDA_CheckFIFOFull()	16
2.2.5 EPCIO_DDA_GetStockCount()	17
2.2.6 EPCIO_DDA_EnableOutABSwap()	17
2.2.7 EPCIO_DDA_DisableOutABSwap()	18
2.2.8 EPCIO_DDA_EnableOutAInverse()	18
2.2.9 EPCIO_DDA_DisableOutAInverse()	18
2.2.10 EPCIO_DDA_EnableOutBInverse()	19
2.2.11 EPCIO_DDA_DisableOutBInverse()	19
2.2.12 EPCIO_DDA_SetOutputFormat()	20
2.2.13 EPCIO_DDA_SetPulseWidth()	20
2.2.14 EPCIO_DDA_EnableStockInt()	21
2.2.15 EPCIO_DDA_DisableStockInt()	21
2.2.16 EPCIO_DDA_EnableCycleInt()	21
2.2.17 EPCIO_DDA_DisableCycleInt()	22
2.2.18 EPCIO_DDA_EnableOutputChannel()	22



2.2.19	EPCIO_DDA_DisableOutputChannel()	22
2.2.20	EPCIO_DDA_SetBitLength()	23
2.2.21	EPCIO_DDA_SetClockDivider()	23
2.2.22	EPCIO_DDA_SetTime()	24
2.2.23	EPCIO_DDA_GetTime()	25
2.2.24	EPCIO_DDA_SetMinStockNo()	25
2.2.25	EPCIO_DDA_SendPulse()	25
2.2.26	EPCIO_DDA_StartEngine()	26
2.2.27	EPCIO_DDA_StopEngine()	26
2.2.28	EPCIO_DDA_ShiftOutFIFOCmd()	27
2.2.29	EPCIO_DDA_EraseFIFOCmd()	27
2.2.30	EPCIO_DDA_EnableEmgcStop()	28
2.2.31	EPCIO_DDA_DisableEmgcStop()	28
2.2.32	EPCIO_DDA_GetOutputPulse()	28
2.2.33	EPCIO_DDA_SetPulseDivider()	29
2.2.34	EPCIO_DDA_ClearPulseDivider()	30
2.2.35	EPCIO_DDA_EnablePulseCounter()	30
2.2.36	EPCIO_DDA_DisablePulseCounter()	30
2.2.37	EPCIO_DDA_ClearPulseCounter()	31
2.3.	編碼器控制(ENC)	32
2.3.1	EPCIO_ENC012_GetIntCondition()	32
2.3.2	EPCIO_ENC345_GetIntCondition()	33
2.3.3	EPCIO_ENC678_GetIntCondition()	33
2.3.4	EPCIO_ENC_GetValue()	34
2.3.5	EPCIO_ENC_GetLatchValue()	34
2.3.6	EPCIO_ENC_GetIndexStatus()	34
2.3.7	EPCIO_ENC_SetCompValue()	35
2.3.8	EPCIO_ENC_EnableCompInt()	35
2.3.9	EPCIO_ENC_DisableCompInt()	36
2.3.10	EPCIO_ENC_SetInputRate()	36
2.3.11	EPCIO_ENC_GetInputRate()	37
2.3.12	EPCIO_ENC_SetInputType()	37



2.3.13	EPCIO_ENC_GetInputType()	38
2.3.14	EPCIO_ENC_EnableInAInverse()	38
2.3.15	EPCIO_ENC_DisableInAInverse()	39
2.3.16	EPCIO_ENC_EnableInBInverse()	39
2.3.17	EPCIO_ENC_DisableInBInverse()	39
2.3.18	EPCIO_ENC_EnableInCInverse()	40
2.3.19	EPCIO_ENC_DisableInCInverse()	40
2.3.20	EPCIO_ENC_EnableInABSwap()	40
2.3.21	EPCIO_ENC_DisableInABSwap()	41
2.3.22	EPCIO_ENC_SetTrigMode()	41
2.3.23	EPCIO_ENC_SetTrigSource()	42
2.3.24	EPCIO_ENC_EnableIndexInt()	42
2.3.25	EPCIO_ENC_DisableIndexInt()	43
2.3.26	EPCIO_ENC_SetFilterClock()	43
2.3.27	EPCIO_ENC_ClearCounter()	43
2.3.28	EPCIO_ENC_StartInput()	44
2.3.29	EPCIO_ENC_StopInput()	44
2.4.	近端輸出入接點控制(LIO)	45
2.4.1	EPCIO_LIO_GetIntCondition()	45
2.4.2	EPCIO_LIO_GetLDIIInput()	46
2.4.3	EPCIO_LIO_SetLDOOutput()	46
2.4.4	EPCIO_LIO_GetLDOOutput()	47
2.4.5	EPCIO_LIO_SetLDOOutputPoint()	47
2.4.6	EPCIO_LIO_EnableLDOOutput()	47
2.4.7	EPCIO_LIO_DisableLDOOutput()	48
2.4.8	EPCIO_LIO_SetLDIIIntType()	49
2.4.9	EPCIO_LIO_EnableLDIIInt()	49
2.4.10	EPCIO_LIO_DisableLDIIInt()	50
2.4.11	EPCIO_LIO_SetTimer()	50
2.4.12	EPCIO_LIO_EnableTimer()	51
2.4.13	EPCIO_LIO_DisableTimer()	51
2.4.14	EPCIO_LIO_EnableTimerInt()	51



2.4.15	EPCIO_LIO_DisableTimerInt()	52
2.4.16	EPCIO_LIO_SetWDogTimer()	52
2.4.17	EPCIO_LIO_EnableWDogTimer()	52
2.4.18	EPCIO_LIO_DisableWDogTimer()	53
2.4.19	EPCIO_LIO_SetWDogReset()	53
2.4.20	EPCIO_LIO_RefreshWDogTimer()	53
2.4.21	EPCIO_LIO_GetOverTravelUp()	54
2.4.22	EPCIO_LIO_GetOverTravelDown()	54
2.4.23	EPCIO_LIO_GetHomeSensor()	55
2.4.24	EPCIO_LIO_GetEmgcStopStatus()	56
2.4.25	EPCIO_LIO_ServoOff()	56
2.4.26	EPCIO_LIO_ServoOn()	56
2.4.27	EPCIO_LIO_EnablePrdy()	57
2.4.28	EPCIO_LIO_DisablePrdy()	57
2.4.29	EPCIO_LIO_EnablePulseDAC()	58
2.4.30	EPCIO_LIO_DisablePulseDAC()	58
2.5.	遠端輸出入接點控制(RIO)	59
2.5.1	EPCIO_RIO0_GetIntCondition()	59
2.5.2	EPCIO_RIO1_GetIntCondition()	60
2.5.3	EPCIO_RIO_GetInputValue()	60
2.5.4	EPCIO_RIO_SetOutputValue()	61
2.5.5	EPCIO_RIO_GetOutputValue()	62
2.5.6	EPCIO_RIO_GetTransStatus()	63
2.5.7	EPCIO_RIO_GetMasterStatus()	63
2.5.8	EPCIO_RIO_GetSlaveStatus()	64
2.5.9	EPCIO_RIO_GetSlaveFail()	64
2.5.10	EPCIO_RIO_SetClockDivider()	65
2.5.11	EPCIO_RIO_SetIntType()	65
2.5.12	EPCIO_RIO_EnableInputInt()	66
2.5.13	EPCIO_RIO_DisableInputInt()	67
2.5.14	EPCIO_RIO_SetTransError()	68
2.5.15	EPCIO_RIO_EnableSetControl()	68



2.5.16	EPCIO_RIO_DisableSetControl()	69
2.5.17	EPCIO_RIO_EnableSlaveControl()	69
2.5.18	EPCIO_RIO_DisableSlaveControl()	70
2.5.19	EPCIO_RIO_EnableTransInt()	71
2.5.20	EPCIO_RIO_DisableTransInt()	71
2.6.	類比轉數位輸入控制(ADC)	72
2.6.1	EPCIO_ADC_GetIntCondition()	72
2.6.2	EPCIO_ADC_GetWorkStatus()	73
2.6.3	EPCIO_ADC_GetInput()	73
2.6.4	EPCIO_ADC_SetCompMask()	73
2.6.5	EPCIO_ADC_SetCompValue()	74
2.6.6	EPCIO_ADC_SetCompType()	75
2.6.7	EPCIO_ADC_EnableCompInt()	75
2.6.8	EPCIO_ADC_DisableCompInt()	76
2.6.9	EPCIO_ADC_EnableTagInt()	76
2.6.10	EPCIO_ADC_DisableTagInt()	77
2.6.11	EPCIO_ADC_EnableConvInt()	77
2.6.12	EPCIO_ADC_DisableConvInt()	77
2.6.13	EPCIO_ADC_SetClockDivider()	77
2.6.14	EPCIO_ADC_SetConvType()	78
2.6.15	EPCIO_ADC_GetConvType()	78
2.6.16	EPCIO_ADC_EnableConvChannel()	79
2.6.17	EPCIO_ADC_DisableConvChannel()	79
2.6.18	EPCIO_ADC_SetConvMode()	79
2.6.19	EPCIO_ADC_SetTagChannel()	80
2.6.20	EPCIO_ADC_SetSingleChannel()	80
2.6.21	EPCIO_ADC_StartConv()	81
2.6.22	EPCIO_ADC_StopConv()	81
2.7.	數位轉類比輸出控制(DAC)	82
2.7.1	EPCIO_DAC_SetOutput()	82
2.7.2	EPCIO_DAC_GetOutput()	82
2.7.3	EPCIO_DAC_SetTrigOutput()	83



2.7.4	EPCIO_DAC_SetTrigSource()	83
2.7.5	EPCIO_DAC_SetClockDivider()	84
2.7.6	EPCIO_DAC_EnableTrigMode()	84
2.7.7	EPCIO_DAC_DisableTrigMode()	85
2.7.8	EPCIO_DAC_SetCmdSource()	85
2.7.9	EPCIO_DAC_StartConv()	86
2.7.10	EPCIO_DAC_StopConv()	86
2.8.	硬體位置閉迴路控制設定(PCL)	87
2.8.1	EPCIO_PCL_GetIntCondition()	87
2.8.2	EPCIO_PCL_GetErrorCounter()	87
2.8.3	EPCIO_PCL_SetScaleGain()	88
2.8.4	EPCIO_PCL_EnableOverflowInt()	88
2.8.5	EPCIO_PCL_DisableOverflowInt()	89
2.8.6	EPCIO_PCL_ClearErrorCounter()	89
2.8.7	EPCIO_PCL_EnableErrorCounter()	90
2.8.8	EPCIO_PCL_DisableErrorCounter()	90
2.8.9	EPCIO_PCL_StartControl()	90
2.8.10	EPCIO_PCL_StopControl()	91

1. EPCIO Series 驅動函式庫簡介

EPCIO Series 驅動函式庫可用來驅動利用 EPCIO ASIC 所設計開發的 PCI 與 PCIe 介面之控制板以及輸出入控制模組，PCI 介面之運動控制卡包括 EPCIO-4000、EPCIO-4005、EPCIO-6000 與 EPCIO-6005，PCIe 之運動控制卡包括 EPCIO-6000e 與 EPCIO-6005e，輸出入控制模組包括 EDIO-S003、EDIO-S005 與 EDIO-S003H。Windows XP、7、10 環境下提供靜態函式庫 (MEPCIOPCIDrv.Lib 或 MEPCIOPCIEdrv.lib)、動態連結函式庫 (MEPCIOPCIDrv.DLL 或 MEPCIOPCIEdrv.dll)，函式原型宣告及資料型態宣告的標頭檔 (MEPCIODev.h)，常數定義的標頭檔 (EPCIOIni.h)，使用者必須先引用這兩個標頭檔，呼叫相對應的功能函式，便可直接驅動 EPCIO Series 運動控制卡。

EPCIO Series 驅動函式庫共有超過 100 個函式可供使用者呼叫，依驅動運動控制卡上不同輸出或輸入功能區分成八大部份：

▲ Bus Interface	運動控制卡初始化、中斷及重置功能(系統功能)
▲ DDA Control Interface	脈波輸出控制(DDA)
▲ Encoder Counter Interface	編碼器控制(ENC)
▲ Local I/O Control Interface	近端輸出入點控制(LIO)
▲ Remote I/O Interface	遠端輸出入點控制(RIO)
▲ ADC Control Interface	類比轉數位輸入控制(ADC)
▲ DAC Control Interface	數位轉類比輸出控制(DAC)
▲ PCL Control Interface	硬體位置閉迴路控制(PCL)

範例程式乃使用 EPCIO Series 驅動函式庫所設計，主要針對各功能模組作一使用說明，包含 DDA 模組脈波輸出、ENC 模組編碼器輸入、LIO 近端輸出入點控制、計時器程式規劃、RIO 遠端輸出入點控制、ADC 類比電壓輸入、DAC 類比電壓輸出、PCL 硬體閉迴路控制等，詳細範例說明請參考” EPCIO Series 驅動函式庫範例手冊”。

2. EPCIO Series 驅動函式庫功能

2.1. 運動控制卡初始化、中斷及重置功能(系統功能)

2.1.1 EPCIO4000_Init()

```
BOOL EPCIO4000_Init(DDAISR    pfnDDAInterrupt,  
                    ENCISR    pfnENC0Interrupt,  
                    ENCISR    pfnENC1Interrupt,  
                    ENCISR    pfnENC2Interrupt,  
                    RIOISR    pfnRIO0Interrupt,  
                    RIOISR    pfnRIO1Interrupt,  
                    ADCISR    pfnADCInterrupt,  
                    LIOISR    pfnLIOInterrupt,  
                    PCLISR    pfnPCLInterrupt,  
                    WORD      wCardIndex)
```

Description 初始化 EPCIO-4000/4005 控制卡並指定使用者自訂中斷服務函式，當函式指標給定為 NULL 時，則呼叫內定的中斷服務函式。

Parameters

- pfnDDAInterrupt* 使用者自訂 DDA 中斷服務函式的函式指標
- pfnENC0Interrupt* 使用者自訂 ENC0(Channel 0 ~ 2)中斷服務函式的函式指標
- pfnENC1Interrupt* 使用者自訂 ENC1(Channel 3 ~ 5)中斷服務函式的函式指標
- pfnENC2Interrupt* 使用者自訂 ENC2(Channel 6 ~ 8)中斷服務函式的函式指標
- pfnRIO0Interrupt* 使用者自訂 RIO0 中斷服務函式的函式指標
- pfnRIO1Interrupt* 使用者自訂 RIO1 中斷服務函式的函式指標
- pfnADCInterrupt* 使用者自訂 ADC 中斷服務函式的函式指標
- pfnLIOInterrupt* 使用者自訂 LIO 中斷服務函式的函式指標
- pfnPCLInterrupt* 使用者自訂 PCL 中斷服務函式的函式指標
- wCardIndex* 運動控制卡編號，編號範圍 0 ~ 11，由使用者自行選定。

在 EPCIO 驅動函式庫中將利用此項編號來識別運動控制卡，因此不同的運動控制卡需選擇不同的編號。因編號範圍的限制，在同一部 PC 中，最多只能同時使用 12 片 EPCIO 運動控制卡

Return Value	TRUE	初始化 EPCIO-4000/4005 模組成功
	FALSE	初始化 EPCIO-4000/4005 模組失敗
Remark	本函式僅適用於 EPCIO-4000/4005 控制卡	

2.1.2 EPCIO6000_Init()

```
BOOL EPCIO6000_Init(DDAISR    pfnDDAInterrupt,  
                               ENCISR    pfnENC0Interrupt,  
                               ENCISR    pfnENC1Interrupt,  
                               ENCISR    pfnENC2Interrupt,  
                               RIOISR    pfnRIO0Interrupt,  
                               RIOISR    pfnRIO1Interrupt,  
                               ADCISR    pfnADCInterrupt,  
                               LIOISR    pfnLIOInterrupt,  
                               PCLISR    pfnPCLInterrupt,  
                               WORD      wCardIndex)
```

Description 初始化 EPCIO-6000/6005 控制卡並指定使用者自訂中斷服務函式，當函式指標給定為 NULL 時，則呼叫內定的中斷服務函式。

Parameters

- pfnDDAInterrupt* 使用者自訂 DDA 中斷服務函式的函式指標
- pfnENC0Interrupt* 使用者自訂 ENC0(Channel 0 ~ 2)中斷服務函式的函式指標
- pfnENC1Interrupt* 使用者自訂 ENC1(Channel 3 ~ 5)中斷服務函式的函式指標
- pfnENC2Interrupt* 使用者自訂 ENC2(Channel 6 ~ 8)中斷服務函式的函式指標
- pfnRIO0Interrupt* 使用者自訂 RIO0 中斷服務函式的函式指標
- pfnRIO1Interrupt* 使用者自訂 RIO1 中斷服務函式的函式指標



<i>pfnADCInterrupt</i>	使用者自訂 ADC 中斷服務函式的函式指標
<i>pfnLIOInterrupt</i>	使用者自訂 LIO 中斷服務函式的函式指標
<i>pfnPCLInterrupt</i>	使用者自訂 PCL 中斷服務函式的函式指標
<i>wCardIndex</i>	運動控制卡編號，編號範圍 0~11，由使用者自行選定。 在 EPCIO 驅動函式庫中將利用此項編號來識別運動控制卡，因此不同的運動控制卡需選擇不同的編號。因編號範圍的限制，在同一部 PC 中，最多只能同時使用 12 片 EPCIO 運動控制卡
Return Value	TRUE 初始化 EPCIO-6000/6005 模組成功 FALSE 初始化 EPCIO-6000/6005 模組失敗
Remark	本函式僅適用於 EPCIO-6000/6005 控制卡

2.1.3 EPCIO6000e_Init()

```
BOOL EPCIO6000e_Init(DDAISR pfnDDAInterrupt,
                     ENCISR pfnENC0Interrupt,
                     ENCISR pfnENC1Interrupt,
                     ENCISR pfnENC2Interrupt,
                     RIOISR pfnRIO0Interrupt,
                     RIOISR pfnRIO1Interrupt,
                     ADCISR pfnADCInterrupt,
                     LIOISR pfnLIOInterrupt,
                     PCLISR pfnPCLInterrupt,
                     WORD wCardIndex)
```

Description 初始化 EPCIO-6000e/6005e 控制卡並指定使用者自訂中斷服務函式，當函式指標給定為 NULL 時，則呼叫內定的中斷服務函式。

Parameters *pfnDDAInterrupt* 使用者自訂 DDA 中斷服務函式的函式指標
pfnENC0Interrupt 使用者自訂 ENC0(Channel 0~2)中斷服務函式的函式指標

<i>pfnENC1Interrupt</i>	使用者自訂 ENC1(Channel 3 ~ 5)中斷服務函式的函式指標
<i>pfnENC2Interrupt</i>	使用者自訂 ENC2(Channel 6 ~ 8)中斷服務函式的函式指標
<i>pfnRIO0Interrupt</i>	使用者自訂 RIO0 中斷服務函式的函式指標
<i>pfnRIO1Interrupt</i>	使用者自訂 RIO1 中斷服務函式的函式指標
<i>pfnADCInterrupt</i>	使用者自訂 ADC 中斷服務函式的函式指標
<i>pfnLIOInterrupt</i>	使用者自訂 LIO 中斷服務函式的函式指標
<i>pfnPCLInterrupt</i>	使用者自訂 PCL 中斷服務函式的函式指標
<i>wCardIndex</i>	運動控制卡編號，編號範圍 0 ~ 11，由使用者自行選定。

在 EPCIO 驅動函式庫中將利用此項編號來識別運動控制卡，因此不同的運動控制卡需選擇不同的編號。因編號範圍的限制，在同一部 PC 中，最多只能同時使用 12 片 EPCIO 運動控制卡

Return Value	TRUE	初始化 EPCIO-6000e/6005e 模組成功
	FALSE	初始化 EPCIO-6000e/6005e 模組失敗
Remark	本函式僅適用於 EPCIO-6000e/6005e 控制卡	

2.1.4 EPCIO_SetISRFunction()

```
void EPCIO_SetISRFunction(RIOISR     pfnRIO0Interrupt,  
                                      RIOISR     pfnRIO1Interrupt,  
                                      ADCISR     pfnADCInterrupt,  
                                      LIOISR     pfnLIOInterrupt,  
                                      PCLISR     pfnPCLInterrupt,  
                                      WORD     wCardIndex)
```

Description 設定使用者自訂的中斷服務函式，使用此函式必須使用於呼叫 EPCIO4000_Init()、EPCIO6000_Init()或 EPCIO6000e_Init()之前。

Parameters ***pfnRIO0Interrupt*** 使用者自訂 RIO0 中斷服務函式的函式指標
pfnRIO1Interrupt 使用者自訂 RIO1 中斷服務函式的函式指標



<i>pfnADCInterrupt</i>	使用者自訂 ADC 中斷服務函式的函式指標
<i>pfnLIOInterrupt</i>	使用者自訂 LIO 中斷服務函式的函式指標
<i>pfnPCLInterrupt</i>	使用者自訂 PCL 中斷服務函式的函式指標
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.1.5 EPCIO_Close()

BOOL EPCIO_Close(WORD wCardIndex)

Description 結束 EPCIO 模組，本函式會關閉 EPCIO 模組內所有功能，若初始化時有設定中斷功能，亦會還原中斷向量。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉 EPCIO 模組成功

FALSE 關閉 EPCIO 模組失敗

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.1.6 EPCIO_ResetModule()

BOOL EPCIO_ResetModule(WORD wModule, WORD wCardIndex)

Description 重置所選定的 EPCIO 模組，本函式為提供使用者以軟體設定方式重置 EPCIO 模組，EPCIO 可單獨重置各模組，亦可重置所有模組。

Parameters *wModule* 欲重置 EPCIO 模組的編號，可設定值有：

RESET_DDA DDA 模組

RESET_ENC ENC 模組

RESET_RIO0 Remote I/O Set 0 模組

RESET_RIO1 Remote I/O Set 1 模組

RESET_ADC ADC 模組

RESET_LIO Local I/O 模組

RESET_PCL PCL 模組



	<i>RESET_DAC</i>	DAC 模組
	<i>RESET_PERI</i>	Peripheral 模組
	<i>RESET_ALL</i>	所有模組
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	重置 EPCIO 模組成功
	FALSE	指定之 <i>wModule</i> 錯誤，不在參數設定的有效值內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.1.7 EPCIO_GetIntSource()

BOOL EPCIO_GetIntSource(EPCIOINT* pInterrupt, WORD wCardIndex)

Description EPCIO 除了在 DAC 模組之外，其它模組皆有不同的中斷訊號產生源，當硬體中斷產生時應先呼叫此函數判別中斷是由那一個模組所觸發，再至該模組所支援的中斷服務函式判斷中斷產生的源由。

Parameters *pInterrupt* 指向一 EPCIOINT 結構指標，用來存放中斷訊號之模組編號，相關定義如下：

```
typedef struct _EPCIO_INT
{
    BYTE    DDA;
    BYTE    ENC012;
    BYTE    ENC345;
    BYTE    ENC678;
    BYTE    RIO0;
    BYTE    RIO1;
    BYTE    ADC;
    BYTE    LIO;
    BYTE    PCL;
} EPCIOINT;
```

各狀態變數說明如下：

DDA DDA 模組觸發中斷



<i>ENC012</i>	ENC 0 ~ 2 模組觸發中斷
<i>ENC345</i>	ENC 3 ~ 5 模組觸發中斷
<i>ENC678</i>	ENC 6 ~ 8 模組觸發中斷
<i>RIO0</i>	RIO Set 0 模組觸發中斷
<i>RIO1</i>	RIO Set 1 模組觸發中斷
<i>ADC</i>	ADC 模組觸發中斷
<i>LIO</i>	LIO 模組觸發中斷
<i>PCL</i>	PCL 模組觸發中斷
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE 讀取成功
	FALSE 讀取失敗或各模組未觸發中斷
Remark	本函式適用於所有 EPCIO Series 運動控制卡
See Also	EPCIO_DDA_GetIntCondition() EPCIO_ENC012_GetIntCondition() EPCIO_ENC345_GetIntCondition() EPCIO_ENC678_GetIntCondition() EPCIO_RIO0_GetIntCondition() EPCIO_RIO1_GetIntCondition() EPCIO_ADC_GetIntCondition() EPCIO_LIO_GetIntCondition() EPCIO_PCL_GetIntCondition()



2.2. 脈波輸出控制(DDA)

2.2.1 EPCIO_DDA_GetIntCondition()

BOOL EPCIO_DDA_GetIntCondition(DDAINT* pInterrupt, WORD wCardIndex)

Description 讀取 DDA 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource()判斷是否為 DDA 所發生，若是則呼叫本函式讀取中斷發生源。

Parameters *pInterrupt* 指向一 DDAINT 結構指標，用來存放 DDA 中斷發生源訊號編碼；DDAINT 結構定義如下：

```
typedef struct _DDA_INT
{
    BYTE    FIFO0;
    BYTE    FIFO1;
    BYTE    FIFO2;
    BYTE    FIFO3;
    BYTE    FIFO4;
    BYTE    FIFO5;
    BYTE    CYCLE;
}DDAINT;
```

各狀態變數說明如下：

FIFO0 DDA Channel 0
~ *FIFO5* ~ DDA Channel 5 達到最小儲存筆數觸發中斷
CYCLE DDA 循環中斷觸發
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功
FALSE 讀取失敗或 DDA 未觸發中斷

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See also EPCIO_GetIntSource()

2.2.2 EPCIO_DDA_GetCurrentCmd()

BOOL EPCIO_DDA_GetCurrentCmd(WORD wChannel, int* pnCommand, WORD wCardIndex)

Description 讀取指定之 DDA Channel 目前正在執行之 DDA Pulse 命令。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
pnCommand 指向一 int 值，用來存放 Pulse 命令
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.3 EPCIO_DDA_CheckFIFOEmpty()

BOOL EPCIO_DDA_CheckFIFOEmpty(WORD wChannel, WORD* pwFlag, WORD wCardIndex)

Description 檢查目前指定之 DDA Channel 之 FIFO 是否為空的狀態。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
pwFlag 指向一 WORD 值，用來存放 FIFO 是否為空的狀態，1 表示 FIFO 為空，0 則否
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 檢查成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.4 EPCIO_DDA_CheckFIFOFull()

BOOL EPCIO_DDA_CheckFIFOFull(WORD wChannel, WORD* pwFlag, WORD wCardIndex)

Description 檢查目前指定之 DDA Channel 之 FIFO 是否已滿(Full)的狀態。

Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>pwFlag</i>	指向一 WORD 值，用來存放 FIFO 是否為滿的狀態，1 表示 FIFO 已滿，0 則否
Return Value	TRUE	檢查成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.5 EPCIO_DDA_GetStockCount()

BOOL EPCIO_DDA_GetStockCount(WORD wChannel, WORD* pwStockCount, WORD wCardIndex)

Description	讀取目前指定之 DDA Channel FIFO 中所儲存但尚未被消化並送出之命令筆數。	
Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>pwStockCount</i>	指向一 WORD 值，用來存放命令庫存數目
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.6 EPCIO_DDA_EnableOutABSwap()

BOOL EPCIO_DDA_EnableOutABSwap(WORD wChannel, WORD wCardIndex)

Description	設定 DDA Channel 脈波輸出格式時，脈波輸出腳 A 及 B 兩訊號線對調。預設狀態為不對調。	
Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	開啟成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.7 EPCIO_DDA_DisableOutABSwap()

BOOL EPCIO_DDA_DisableOutABSwap(WORD wChannel, WORD wCardIndex)

Description 設定 DDA Channel 脈波輸出格式時，脈波輸出腳 A 及 B 兩訊號線不對調。預設狀態為不對調。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.8 EPCIO_DDA_EnableOutAInverse()

BOOL EPCIO_DDA_EnableOutAInverse(WORD wChannel, WORD wCardIndex)

Description 設定 DDA Channel 脈波輸出格式時，脈波輸出腳 A 訊號線反相。預設狀態為不反相。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.9 EPCIO_DDA_DisableOutAInverse()

BOOL EPCIO_DDA_DisableOutAInverse(WORD wChannel, WORD wCardIndex)

Description 設定 DDA Channel 脈波輸出格式時，脈波輸出腳 A 訊號線不反相。預設狀態為不反相。

Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	關閉成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.10 EPCIO_DDA_EnableOutBIverse()

BOOL EPCIO_DDA_EnableOutBIverse(WORD wChannel, WORD wCardIndex)

Description 設定 DDA Channel 脈波輸出格式時，脈波輸出腳 B 訊號線反相。預設狀態為不反相。

Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	開啟成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.11 EPCIO_DDA_DisableOutBIverse()

BOOL EPCIO_DDA_DisableOutBIverse(WORD wChannel, WORD wCardIndex)

Description 設定 DDA Channel 脈波輸出格式時，脈波輸出腳 B 訊號線不反相。預設狀態為不反相。

Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	關閉成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.12 EPCIO_DDA_SetOutputFormat()

***BOOL EPCIO_DDA_Set_OutputFormat(WORD wChannel, WORD wFormat,
WORD wCardIndex)***

Description	設定 DDA Channel 脈波輸出格式。DDA Channel 的脈波輸出格式可經由軟體設定。	
Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wFormat</i>	DDA 脈波輸出格式，可設定值有：
	<i>DDA_FMT_NO</i>	禁止輸出
	<i>DDA_FMT_PD</i>	Pulse/Direction
	<i>DDA_FMT_CW</i>	CW/CCW
	<i>DDA_FMT_AB</i>	A/B Phase
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 或 <i>wFormat</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.13 EPCIO_DDA_SetPulseWidth()

***BOOL EPCIO_DDA_SetPulseWidth(WORD wChannel, WORD wWidth,
WORD wCardIndex)***

Description	DDA 輸出之脈波寬度可根據不同的驅動器需求進行設定輸出寬度的長短，以符合驅動器規格。利用此函式可將脈波輸出寬度設定為 <i>wWidth</i> 個的週期寬度。	
Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wWidth</i>	DDA 所脈波輸出的寬度，設定值為 0 ~ 2047 個 System Clock (25ns)。預設值為 0，表示無脈波輸出
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 或 <i>wWidth</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.14 EPCIO_DDA_EnableStockInt()

BOOL EPCIO_DDA_EnableStockInt(WORD wChannel, WORD wCardIndex)

Description 開啟 DDA FIFO 最小儲存筆數中斷功能。當 DDA Engine 在固定時間自 DDA FIFO 中讀取命令時，若 FIFO 中剩餘之命令筆數等於所設定的最小庫存筆數時，將觸發硬體 IRQ 中斷。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.15 EPCIO_DDA_DisableStockInt()

BOOL EPCIO_DDA_DisableStockInt(WORD wChannel, WORD wCardIndex)

Description 關閉 DDA FIFO 最小儲存數目中斷功能，不再觸發硬體 IRQ 中斷。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.16 EPCIO_DDA_EnableCycleInt()

BOOL EPCIO_DDA_EnableCycleInt(WORD wCardIndex)

Description 開啟 DDA 循環中斷功能，DDA 將在每隔 DDA 時間的固定週期自動觸發硬體 IRQ 中斷。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功
FALSE 開啟不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.17 EPCIO_DDA_DisableCycleInt()

BOOL EPCIO_DDA_DisableCycleInt(WORD wCardIndex)

Description 關閉 DDA 循環中斷功能，不再以每隔 DDA 時間固定週期產生硬體 IRQ 中斷觸發。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功

FALSE 關閉不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.18 EPCIO_DDA_EnableOutputChannel()

BOOL EPCIO_DDA_EnableOutputChannel(WORD wChannel, WORD wCardIndex)

Description 啟動指定 Channel 的 DDA 脈波控制功能。本函式為開啟單軸 DDA 脈波輸出功能，但至少必需呼叫過一次 EPCIO_DDA_StartEngine()，脈波輸出功能才有效。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功

FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_DDA_StartEngine()

2.2.19 EPCIO_DDA_DisableOutputChannel()

BOOL EPCIO_DDA_DisableOutputChannel(WORD wChannel, WORD wCardIndex)

Description 停止指定 Channel 的 DDA 脈波控制功能。設定本函式後指定的 DDA Channel 立即停止脈波輸出，同時 FIFO 內正在執行但尚未執行完之命令將被移除。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)

	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	關閉成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.20 EPCIO_DDA_SetBitLength()

BOOL EPCIO_DDA_SetBitLength(WORD wDDALength, WORD wCardIndex)

Description 設定 DDA Engine 運作時所使用的數位運算法則的 bit 數目，此設定值將會影響每個 DDA 時間所能送出的最大脈波數 ($2^{wDDALength}$)。

Parameters	<i>wDDALength</i>	DDA 數位運算法則所使用的 bit 數目
	<i>DDA_LEN10</i>	10 bits
	<i>DDA_LEN11</i>	11 bits
	<i>DDA_LEN12</i>	12 bits
	<i>DDA_LEN13</i>	13 bits
	<i>DDA_LEN14</i>	14 bits
	<i>DDA_LEN15</i>	15 bits
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Return Value	TRUE	設定成功
	FALSE	指定之 <i>wDDALength</i> 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_DDA_SetClockDivider() EPCIO_DDA_SetTime()

2.2.21 EPCIO_DDA_SetClockDivider()

BOOL EPCIO_DDA_SetClockDivider(WORD wDivider, WORD wCardIndex)

Description 設定 DDA Engine 運作時，所使用的工作時脈(DDA Clock)為 System Clock (40MHz) 除以 ($wDivider + 1$)，*wDivider* 內定設定值為 0 表示除 1。本設定將影響最小兩相鄰 Pulses 間的時間。

Parameters	<i>wDivider</i>	DDA 內時脈除頻值(0 ~ 4095)
------------	------------------------	----------------------

	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wDivider</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_DDA_SetBitLength()	EPCIO_DDA_SetTime()

2.2.22 EPCIO_DDA_SetTime()

BOOL EPCIO_DDA_SetTime(float fDDATime, WORD wDDALength, WORD wCardIndex)

Description 設定 DDA Engine 運作時的週期時間。此函式會將 DDA 週期時間轉換成為 DDA 內時脈除頻值及 bit 長度並寫入硬體，當啟動 DDA Engine 控制功能後，EPCIO ASIC 將會固定時間自 FIFO 中讀取一筆命令，並經由 DDA Engine 轉成脈波輸出。

Parameters ***fDDATime*** DDA 週期時間，設定範圍會由使用者所給定 *wDDALength* 不同而有所不同

wDDALength DDA 數位運算法則所使用的 bit 數目，可設定值有：

DDA_LEN10 (1 ≤ *fDDATime* ≤ 105) ms

DDA_LEN11 (1 ≤ *fDDATime* ≤ 210) ms

DDA_LEN12 (1 ≤ *fDDATime* ≤ 420) ms

DDA_LEN13 (1 ≤ *fDDATime* ≤ 840) ms

DDA_LEN14 (1 ≤ *fDDATime* ≤ 1680) ms

DDA_LEN15 (1 ≤ *fDDATime* ≤ 3360) ms

當使用者不指定 *wDDALength* 時，預設值為 *DDA_LEN15*。

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *fDDATime* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_DDA_SetBitLength() EPCIO_DDA_SetClockDivider()

2.2.23 EPCIO_DDA_GetTime()

BOOL EPCIO_DDA_GetTime(float* pfDDATime, WORD wCardIndex)

Description 讀取 DDA Engine 運作時的週期時間。

Parameters *pfDDATime* DDA 週期時間，單位為 ms
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功
FALSE 讀取失敗

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_DDA_SetBitLength() EPCIO_DDA_SetClockDivider()

2.2.24 EPCIO_DDA_SetMinStockNo()

BOOL EPCIO_DDA_SetMinStockNo(WORD wStockNumber, WORD wCardIndex)

Description 設定 DDA FIFO 中觸發中斷功能之最小儲存命令筆數，所有 DDA Channel 均使用相同的設定。設定後且經由 EPCIO_EnableFIFOStockInt() 啟動 FIFO 最小儲存筆數檢查中斷功能，則在 FIFO 中剩餘的儲存筆數等於 *wStockNumber* 時將觸發中斷產生。

Parameters *wStockNumber* 最小 FIFO 儲存命令筆數(1 ~ 63)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wStockNumber* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_EnableFIFOStockInt()

2.2.25 EPCIO_DDA_SendPulse()

BOOL EPCIO_DDA_SendPulse(WORD wChannel, int nPulse, WORD wCardIndex)

Description 將 Pulse 命令寫入指定之 DDA FIFO；每一筆 Pulse 命令所能送出的最大值與設定的 DDA bit 數目相關，且與 Pulse 寬度相關，當設定 DDA 為 10 bits 時，則

每個 DDA 時間可接受最大的 Pulse 命令 = $\pm 2^{10}$ ，以此類推當設定 DDA 為 N bits 時，每個 DDA 時間最大輸出 Pulse 命令為 $\pm 2^N$ 。但當 Pulse 寬度大於兩相鄰 Pulses 間距時(DDA 時間 / 2^N)，則前一 Pulse 將會蓋到下一 Pulse，必須特別注意以避免此種情況發生。

Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>nPulse</i>	Pulse 命令
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	寫入成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_DDA_SetBitLength()	EPCIO_DDA_SetPulseWidth()
	EPCIO_DDA_SetClockDivider()	

2.2.26 EPCIO_DDA_StartEngine()

BOOL EPCIO_DDA_StartEngine(WORD wCardIndex)

Description	啟動 DDA Engine 運作功能	
Parameters	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	設定不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.2.27 EPCIO_DDA_StopEngine()

BOOL EPCIO_DDA_StopEngine(WORD wCardIndex)

Description	停止 DDA Engine 運作功能。	
Parameters	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	設定不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.2.28 EPCIO_DDA_ShiftOutFIFOCmd()

BOOL EPCIO_DDA_ShiftOutFIFOCmd(WORD wCardIndex)

Description 本函式可移除 DDA FIFO 內下一筆待執行的命令。使用本函式移除 DDA FIFO 中的命令時必須先停止 DDA Channel 運作功能，只有已停止的 Channel，FIFO 中的命令才可被移除，執行中的 Channel 將不受影響。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 設定不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_DDA_DisableOutputChannel()

2.2.29 EPCIO_DDA_EraseFIFOCmd()

BOOL EPCIO_DDA_EraseFIFOCmd(WORD wChannel, WORD wEraseNumber, WORD wCardIndex)

Description 本函式可移除 DDA FIFO 內已設定但未執行的命令。本函式可於命令輸出中途刪除 FIFO 內未執行之命令，最多一次可刪除 64 筆命令，但正執行中之該筆命令不受影響。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
wEraseNumber 欲刪除 FIFO 內之未執行命令筆數
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 刪除成功
FALSE 刪除不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。



2.2.30 EPCIO_DDA_EnableEmgcStop()

BOOL EPCIO_DDA_EnableEmgcStop(WORD wCardIndex)

Description 設定 DDA 輸出緊急停止。本函式可於命令輸出中途停止命令輸出，執行中之該筆命令將立即停止輸出，但 EPCIO 內部仍會同步執行命令之計算，取消緊急停止功能後，輸出命令將於下一個 DDA 周期同步與其他軸開始命令輸出。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功

FALSE 設定不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。

See Also EPCIO_DDA_DisableEmgcStop()

2.2.31 EPCIO_DDA_DisableEmgcStop()

BOOL EPCIO_DDA_DisableEmgcStop(WORD wCardIndex)

Description 取消設定 DDA 輸出緊急停止。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 取消設定成功

FALSE 取消設定不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。

See Also EPCIO_DDA_EnableEmgcStop()

2.2.32 EPCIO_DDA_GetOutputPulse()

***BOOL EPCIO_DDA_GetOutputPulse(WORD wChannel, long* plPulse,
WORD wCardIndex)***

Description DDA 實際脈波輸出值數會記錄在 EPCIO 內部暫存器，透過本函式可讀取暫存



器內的脈波值，即邏輯座標值(非實際座標)。

Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>plPulse</i>	指向一 long 值，用來存放 DDA 實際脈波輸出值
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	讀取不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。	
See Also	EPCIO_DDA_EnablePulseCounter() EPCIO_DDA_DisablePulseCounter() EPCIO_DDA_ClearPulseCounter()	

2.2.33 EPCIO_DDA_SetPulseDivider()

BOOL EPCIO_DDA_SetPulseDivider(WORD wChannel, WORD wDivider, WORD wCardIndex)

Description 經由設定 *wDivider* 值，可使實際脈波輸出數為所設定之 FIFO 命令除以 ($wDivider + 1$)，此功能可經由軟體命令倍頻及硬體輸出除頻得到加減速過程命令更均勻之目的。

Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
	<i>wDivider</i>	脈波除頻值(0 ~ 255)，當 <i>wDivider</i> 設定為 0 時，表示 Pulse 除以 1，依此類推
Return Value	TRUE	設定成功
	FALSE	設定不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。	
See Also	EPCIO_DDA_ClearPulseDivider()	

2.2.34 EPCIO_DDA_ClearPulseDivider()

BOOL EPCIO_DDA_ClearPulseDivider(WORD wChannel, WORD wCardIndex)

Description 當呼叫 EPCIO_DDA_SetPulseDivider() 後，則實際脈波輸出數為所設定之 FIFO 命令除以 ($wDivider + 1$)，當命令無法整除時，除法器內部將產生餘數，此函式可清除除法器內之餘數。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 清除成功
FALSE 清除不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。

See Also EPCIO_DDA_SetPulseDivider()

2.2.35 EPCIO_DDA_EnablePulseCounter()

BOOL EPCIO_DDA_EnablePulseCounter(WORD wChannel, WORD wCardIndex)

Description 開啟 EPCIO 內部邏輯座標紀錄器之記錄功能。

Parameters *wChannel* DDA Channel 編號(0 ~ 5)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 設定不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。

See Also EPCIO_DDA_GetOutputPulse() EPCIO_DDA_DisablePulseCounter()
EPCIO_DDA_ClearPulseCounter()

2.2.36 EPCIO_DDA_DisablePulseCounter()

BOOL EPCIO_DDA_DisablePulseCounter(WORD wChannel, WORD wCardIndex)

Description 關閉 EPCIO 內部邏輯座標紀錄器之記錄功能。

Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	設定不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。	
See Also	EPCIO_DDA_GetOutputPulse() EPCIO_DDA_ClearPulseCounter()	EPCIO_DDA_EnablePulseCounter()

2.2.37 EPCIO_DDA_ClearPulseCounter()

BOOL EPCIO_DDA_ClearPulseCounter(WORD wChannel, WORD wCardIndex)

Description	清除 EPCIO 內部邏輯座標紀錄器之值為零。	
Parameters	<i>wChannel</i>	DDA Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	清除成功
	FALSE	清除不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡中使用 EPCIO-ASIC 編號 6988-02 以上版本。	
See Also	EPCIO_DDA_GetOutputPulse() EPCIO_DDA_DisablePulseCounter()	EPCIO_DDA_EnablePulseCounter()

2.3. 編碼器控制(ENC)

2.3.1 EPCIO_ENC012_GetIntCondition()

BOOL EPCIO_ENC012_GetIntCondition(ENCINT* psInterrupt, WORD wCardIndex)

Description 讀取編碼器 Channel 0~2 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource()判斷是否為編碼器 Channel 0~2 所發生，若是則呼叫本函式讀取中斷發生源。

Parameters *psInterrupt* 指向一 ENCINT 結構指標，用來存放 ENC012 中斷發生源訊號編碼；ENCINT 結構定義如下：

```
typedef struct _ENC_INT
{
    BYTE    COMP0;
    BYTE    COMP1;
    BYTE    COMP2;
    BYTE    INDEX0;
    BYTE    INDEX1;
    BYTE    INDEX2;
}ENCINT;
```

各狀態變數說明如下：

<i>COMP0</i>	編碼器 Channel 0
~ <i>COMP2</i>	~ 編碼器 Channel 2 比較器觸發中斷
<i>INDEX0</i>	編碼器 Channel 0
~ <i>INDEX2</i>	~ 編碼器 Channel 2 INDEX 觸發中斷
<i>wCardIndex</i>	欲控制的運動控制卡編號(0~11)

Return Value TRUE 讀取成功
FALSE 讀取失敗或編碼器 Channel 0~2 未觸發中斷

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See also EPCIO_GetIntSource()

2.3.2 EPCIO_ENC345_GetIntCondition()

BOOL EPCIO_ENC345_GetIntCondition(ENCINT* psInterrupt, WORD wCardIndex)

Description 讀取編碼器 Channel 3 ~ 5 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource() 判斷是否為編碼器 Channel 3 ~ 5 所發生，若是則呼叫本函式讀取中斷發生源。

Parameters *psInterrupt* 指向一 ENCINT 結構指標，用來存放 ENC345 中斷發生源訊號編碼；ENCINT 結構定義與各狀態變數說明請參考函式 EPCIO_ENC123_GetIntCondition()

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功

FALSE 讀取失敗或編碼器 Channel 3 ~ 5 未觸發中斷

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See also EPCIO_GetIntSource()

2.3.3 EPCIO_ENC678_GetIntCondition()

BOOL EPCIO_ENC678_GetIntCondition(ENCINT* psInterrupt, WORD wCardIndex)

Description 讀取編碼器 Channel 6 ~ 8 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource() 判斷是否為編碼器 Channel 6 ~ 8 所發生，若是則呼叫本函式讀取中斷發生源。

Parameters *psInterrupt* 指向一 ENCINT 結構指標，用來存放 ENC678 中斷發生源訊號編碼；ENCINT 結構定義與各狀態變數說明請參考函式 EPCIO_ENC123_GetIntCondition()

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功

FALSE 讀取失敗或編碼器 Channel 6 ~ 8 未觸發中斷

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See also EPCIO_GetIntSource()

2.3.4 EPCIO_ENC_GetValue()

BOOL EPCIO_ENC_GetValue(WORD wChannel, long* plCountValue, WORD wCardIndex)

Description 讀取編碼器的計數值。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
plCountValue 指向一 long 值，用來存放編碼器的計數值
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.5 EPCIO_ENC_GetLatchValue()

BOOL EPCIO_ENC_GetLatchValue(WORD wChannel, long* plCountValue, WORD wCardIndex)

Description 讀取 閘鎖(Latch)暫存器內的紀錄值。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
plCountValue 指向一 long 值，用來存放閘鎖(Latch)暫存器的紀錄值
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.6 EPCIO_ENC_GetIndexStatus()

BOOL EPCIO_ENC_GetIndexStatus(WORD wChannel, WORD* pwStatus, WORD wCardIndex)

Description 讀取目前位置是否位於 INDEX 訊號的輸入點上。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
pwStatus 指向一 WORD 值，用來存放 INDEX 訊號的輸入狀態；

		1 代表目前位置在 INDEX 上，0 則否
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.3.7 EPCIO_ENC_SetCompValue()

BOOL EPCIO_ENC_SetCompValue(WORD wChannel, long lCompValue, WORD wCardIndex)

Description	設定編碼器的比較值，當計數值等於設定比較值時，搭配 EPCIO_ENC_EnableCompInt() 可產生一硬體中斷觸發訊號。此觸發訊號可用來觸發 DAC 快速送出一預先設定好的電壓命令。	
Parameters	<i>wChannel</i>	編碼器 Channel 編號(0 ~ 8)
	<i>lCompValue</i>	編碼器的比較值
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_ENC_EnableCompInt()	EPCIO_DAC_SetTrigSource()

2.3.8 EPCIO_ENC_EnableCompInt()

BOOL EPCIO_ENC_EnableCompInt(WORD wChannel, WORD wCardIndex)

Description	開啟編碼器的計數值等於比較值時觸發編碼器中斷觸發功能。	
Parameters	<i>wChannel</i>	編碼器 Channel 編號(0 ~ 8)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_ENC_SetCompValue()

2.3.9 EPCIO_ENC_DisableCompInt()

BOOL EPCIO_ENC_DisableCompInt(WORD wChannel, WORD wCardIndex)

Description 關閉編碼器的計數值等於比較值時觸發編碼器中斷觸發功能。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)

Return Value TRUE 設定成功

FALSE 指定之 *wChannel* 不在設定範圍內

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.10 EPCIO_ENC_SetInputRate()

BOOL EPCIO_ENC_SetInputRate(WORD wChannel, WORD wRate, WORD wCardIndex)

Description 設定編碼器的回授倍率。編碼器的回授倍率必須在輸入格式為 A/B Phase 時方為有效。本函式必須搭配 EPCIO_ENC_SetInputType() 設定為 *ENC_TYPE_AB*。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)

wRate 編碼器的回授倍率，可設定值有：

ENC_RATE_X0 0

ENC_RATE_X1 1

ENC_RATE_X2 2

ENC_RATE_X4 4

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功

FALSE 指定之 *wChannel* 或 *wRate* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_ENC_SetInputType()

2.3.11 EPCIO_ENC_GetInputRate()

BOOL EPCIO_ENC_GetInputRate(WORD wChannel, WORD* pwRate, WORD wCardIndex)

Description 讀取編碼器的回授倍率。編碼器的回授倍率必須在輸入格式為 A/B Phase 時方為有效。

Parameters

<i>wChannel</i>	編碼器 Channel 編號(0 ~ 8)
<i>pwRate</i>	指向一 WORD 值，用來存放編碼器的回授倍率
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Return Value

TRUE	設定成功
FALSE	指定之 <i>wChannel</i> 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_ENC_GetInputType()

2.3.12 EPCIO_ENC_SetInputType()

BOOL EPCIO_ENC_SetInputType(WORD wChannel, WORD wType, WORD wCardIndex)

Description 設定編碼器的訊號輸入型態，此函式必須搭配硬體實際訊號設定，當輸入訊號為馬達編碼器回授訊號時，請參考馬達或驅動器設定；若接一般手搖輪時請設定為 A/B Phase。

Parameters

<i>wChannel</i>	編碼器 Channel 編號(0 ~ 8)
<i>wType</i>	編碼器輸入訊號格式，可設定值有：
<i>ENC_TYPE_AB</i>	A/B Phase(預設)
<i>ENC_TYPE_CW</i>	CW/CCW
<i>ENC_TYPE_PD</i>	Pulse/Direction
<i>ENC_TYPE_NO</i>	禁止輸入
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Return Value

TRUE	設定成功
FALSE	指定之 <i>wChannel</i> 或 <i>wType</i> 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_ENC_SetInputRate()

2.3.13 EPCIO_ENC_GetInputType()

BOOL EPCIO_ENC_GetInputType(WORD wChannel, WORD* pwType, WORD wCardIndex)

Description 讀取編碼器的訊號輸入型態。

Parameters **wChannel** 編碼器 Channel 編號(0 ~ 8)
pwType 指向一 WORD 值，用來存放編碼器輸入訊號格式
0 A/B Phase
1 CW/CCW
2 Pulse/Direction
3 禁止輸入
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 或 *wType* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_ENC_GetInputRate()

2.3.14 EPCIO_ENC_EnableInAIinverse()

BOOL EPCIO_ENC_EnableInAIinverse(WORD wChannel, WORD wCardIndex)

Description 設定編碼器輸入訊號之 inA 腳位反相。預設狀態為無反相。

Parameters **wChannel** 編碼器 Channel 編號(0 ~ 8)
wCardIndex 控制的運動控制卡之編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.15 EPCIO_ENC_DisableInAInverse()

BOOL EPCIO_ENC_DisableInAInverse(WORD wChannel, WORD wCardIndex)

Description 設定編碼器輸入訊號之 inA 腳位無反相。預設狀態為無反相。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
wCardIndex 控制的運動控制卡之編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.16 EPCIO_ENC_EnableInBInverse()

BOOL EPCIO_ENC_EnableInBInverse(WORD wChannel, WORD wCardIndex)

Description 設定編碼器輸入訊號之 inB 腳位反相。預設狀態為無反相。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
wCardIndex 控制的運動控制卡之編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.17 EPCIO_ENC_DisableInBInverse()

BOOL EPCIO_ENC_DisableInBInverse(WORD wChannel, WORD wCardIndex)

Description 設定編碼器輸入訊號之 inB 腳位無反相。預設狀態為無反相。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
wCardIndex 控制的運動控制卡之編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.18 EPCIO_ENC_EnableInCInverse()

BOOL EPCIO_ENC_EnableInCInverse(WORD wChannel, WORD wCardIndex)

Description 設定編碼器輸入訊號之 inC 腳位反相。預設狀態為無反相。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
wCardIndex 控制的運動控制卡之編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.19 EPCIO_ENC_DisableInCInverse()

BOOL EPCIO_ENC_DisableInCInverse(WORD wChannel, WORD wCardIndex)

Description 設定編碼器輸入訊號之 inC 腳位無反相。預設狀態為無反相。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
wCardIndex 控制的運動控制卡之編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.20 EPCIO_ENC_EnableInABSwap()

BOOL EPCIO_ENC_EnableInABSwap(WORD wChannel, WORD wCardIndex)

Description 設定編碼器輸入訊號之 inA 及 inB 腳位，在訊號進入編碼器前需經過訊號交換處理。預設狀態為不需經過訊號交換處理。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
wCardIndex 控制的運動控制卡之編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.21 EPCIO_ENC_DisableInABSwap()

BOOL EPCIO_ENC_DisableInABSwap(WORD wChannel, WORD wCardIndex)

Description 關閉編碼器輸入訊號之 inA 及 inB 腳位，在訊號進入編碼器前不需經過訊號交換處理。預設狀態為不需經過訊號交換處理。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.3.22 EPCIO_ENC_SetTrigMode()

BOOL EPCIO_ENC_SetTrigMode(WORD wChannel, WORD wMode, WORD wCardIndex)

Description 設定觸發閘鎖(Latch)編碼器計數值動作的方式。本函式必須搭配 EPCIO_ENC_SetTrigSource()設定觸發訊號源。

Parameters *wChannel* 編碼器 Channel 編號(0 ~ 8)
wMode 觸發閘鎖編碼器計數值動作的方式
ENC_TRIG_FIRST 第一次滿足觸發條件時，計數值即被閘鎖並不再變動
ENC_TRIG_LAST 當觸發條件滿足時即閘鎖新的計數值，次數不限
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_ENC_SetTrigSource()

2.3.23 EPCIO_ENC_SetTrigSource()

**BOOL EPCIO_ENC_SetTrigSource(WORD wChannel, WORD wTrigSource,
WORD wCardIndex)**

Description 設定編碼器觸發訊號來源，此觸發訊號來源可用來門鎖(Latch)編碼器計數值動作，可同時設定多個訊號來源的聯集。本函式必須搭配 EPCIO_ENC_SetTrigMode() 使用。

Parameters

wChannel	編碼器 Channel 編號(0 ~ 8)
wTrigSource	訊號來源，共有 15 種觸發來源：
ENC_TRIG_NO	沒有選擇任何觸發訊號源
ENC_TRIG_INDEX0	編碼器 Channel 0
~ ENC_TRIG_INDEX8	~ 編碼器 Channel 8 的 INDEX 訊號
ENC_TRIG_LIO0	發生近端輸入接點的 DI 0 中斷
ENC_TRIG_LIO1	發生近端輸入接點的 DI 1 中斷
ENC_TRIG_RDI0	發生遠端輸入接點 Set 0 的 DI 0 中斷
ENC_TRIG_RDI1	發生遠端輸入接點 Set 0 的 DI 1 中斷
ENC_TRIG_ADC0	發生 ADC Channel 0 的比較條件成立
ENC_TRIG_ADC1	發生 ADC Channel 1 的比較條件成立
wCardIndex	欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 wChannel 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_ENC_SetTrigMode()

2.3.24 EPCIO_ENC_EnableIndexInt()

BOOL EPCIO_ENC_EnableIndexInt(WORD wChannel, WORD wCardIndex)

Description 開啟編碼器 INDEX 訊號觸發編碼器中斷觸發功能。

Parameters

wChannel	編碼器 Channel 編號(0 ~ 8)
wCardIndex	欲控制的運動控制卡編號(0 ~ 11)



Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.3.25 EPCIO_ENC_DisableIndexInt()

BOOL EPCIO_ENC_DisableIndexInt(WORD wChannel, WORD wCardIndex)

Description	關閉編碼器 INDEX 訊號觸發編碼器中斷觸發功能。	
Parameters	<i>wChannel</i>	編碼器 Channel 編號(0 ~ 8)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.3.26 EPCIO_ENC_SetFilterClock()

BOOL EPCIO_ENC_SetFilterClock(WORD wClock, WORD wCardIndex)

Description	開啟編碼器濾波取樣功能並設定取樣頻率，取樣頻率為 System Clock (40MHz) 除以 (<i>wClock</i> + 1)。當取樣頻率設定完成後則輸入訊號必須滿足連續 3 個以上取樣值為 High 或 Low，輸入值才為有效。 <i>wClock</i> 預設值為 0。	
Parameters	<i>wClock</i>	編碼器濾波取樣除頻值(0 ~ 255)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wClock</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.3.27 EPCIO_ENC_ClearCounter()

BOOL EPCIO_ENC_ClearCounter(WORD wChannel, WORD wCardIndex)



Description	使編碼器的計數值歸零。	
Parameters	<i>wChannel</i>	編碼器 Channel 編號(0 ~ 8)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.3.28 EPCIO_ENC_StartInput()

BOOL EPCIO_ENC_StartInput(WORD wCardIndex)

Description	啟動編碼器記錄功能。	
Parameters	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	設定不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.3.29 EPCIO_ENC_StopInput()

BOOL EPCIO_ENC_StopInput(WORD wCardIndex)

Description	關閉編碼器記錄功能。	
Parameters	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	設定不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.4. 近端輸出入接點控制(LIO)

2.4.1 EPCIO_LIO_GetIntCondition()

BOOL EPCIO_LIO_GetIntCondition(LIOINT* psInterrupt, WORD wCardIndex)

Description 讀取 LIO 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource()判斷是否為 LIO 所發生，若是則呼叫本函式讀取中斷發生源。

Parameters *psInterrupt* 指向一 LIOINT 結構指標，用來存放 LIO 中斷發生源訊號編碼；LIOINT 結構定義如下：

```
typedef struct _LIO_INT
{
    BYTE    LDI0;
    BYTE    LDI1;
    BYTE    LDI2;
    BYTE    LDI3;
    BYTE    LDI4;
    BYTE    LDI5;
    BYTE    LDI6;
    BYTE    LDI7;
    BYTE    DFI0;
    BYTE    DFI1;
    BYTE    DFI2;
    BYTE    DFI3;
    BYTE    DFI4;
    BYTE    DFI5;
    BYTE    DFI6;
    BYTE    TIMER;
}LIOINT;
```

各狀態變數說明如下：

LDI0 ~ LDI7 近端輸入接點 LDI 0 ~ LDI 7 觸發中斷之狀態

	<i>DFI0 ~ DFI6</i>	近端雙功能接點 DFI 0 ~ DFI 6 觸發中斷之狀態
	<i>Timer</i>	Timer 計時器觸發中斷之狀態
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	讀取失敗或 LIO 未觸發中斷
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See also	EPCIO_GetIntSource()	

2.4.2 EPCIO_LIO_GetLDIInput()

BOOL EPCIO_LIO_GetLDIInput(DWORD* pdwInput, WORD wCardIndex)

Description 讀取近端輸入接點 LDI 0 ~ LDI 27 的訊號值。

Parameters *pdwInput* 近端輸入接點狀態，bit 0 ~ bit 27 表示為 LDI 0 ~ LDI 27
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功
FALSE 讀取不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.4.3 EPCIO_LIO_SetLDOOutput()

BOOL EPCIO_LIO_SetLDOOutput(DWORD dwOutput, WORD wCardIndex)

Description 設定近端輸出接點 LDO 0 ~ LDO 27 的訊號狀態值，此函式需先規劃 LIO 輸出功能。

Parameters *dwOutput* 近端輸出接點狀態，bit 0 ~ bit 27 表示為 LDO 0 ~ LDO 27
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 寫入成功
FALSE 寫入不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_EnableLDOOutput()

2.4.4 EPCIO_LIO_GetLDOOutput()

BOOL EPCIO_LIO_GetLDOOutput(DWORD* dwOutput, WORD wCardIndex)

Description 讀取近端輸出接點 LDO 0 ~ LDO 27 的訊號狀態值。

Parameters ***dwOutput*** 指向一 DWORD 值，用來存放近端輸出接點狀態，bit 0 ~ bit 27 表示為 LDO 0 ~ LDO 27

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功

FALSE 讀取不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_EnableLDOOutput()

2.4.5 EPCIO_LIO_SetLDOOutputPoint()

BOOL EPCIO_LIO_SetLDOOutputPoint(WORD wValue, WORD wPoint, WORD wCardIndex)

Description 設定指定近端輸出接點的訊號狀態值，此函式需先規劃 LIO 輸出功能。

Parameters ***wValue*** 近端輸出接點狀態

wPoint 近端輸出接點編號

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 寫入成功

FALSE 寫入不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_EnableLDOOutput()

2.4.6 EPCIO_LIO_EnableLDOOutput()

BOOL EPCIO_LIO_EnableLDOOutput(WORD wPort, WORD wCardIndex)

Description 單獨啟動特定近端輸出訊號功能，輸出狀態會以 4 點為一個 Port，28 個輸出

點可分為 Port 0 ~ Port 6，所有 Port 的預設輸出狀態為關閉(Disable)。

Parameters	wPort	近端訊號輸出 Port (每 4 點為一個 Port)，可設定值有：
	<i>LIO_OUT_EN0</i>	Port 0 為 LDO 0 ~ LDO 3
	<i>LIO_OUT_EN1</i>	Port 1 為 LDO 4 ~ LDO 7
	<i>LIO_OUT_EN2</i>	Port 2 為 LDO 8 ~ LDO 11
	<i>LIO_OUT_EN3</i>	Port 3 為 LDO 12 ~ LDO 15
	<i>LIO_OUT_EN4</i>	Port 4 為 LDO 16 ~ LDO 19
	<i>LIO_OUT_EN5</i>	Port 5 為 LDO 20 ~ LDO 23
	<i>LIO_OUT_EN6</i>	Port 6 為 LDO 24 ~ LDO 27
	wCardIndex	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	開啟成功
	FALSE	指定之 <i>wPort</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.4.7 EPCIO_LIO_DisableLDOOutput()

BOOL EPCIO_LIO_DisableLDOOutput(WORD wPort, WORD wCardIndex)

Description 單獨關閉特定近端輸出訊號功能。

Parameters	wPort	近端訊號輸出 Port (每 4 點為一個 Port)，可設定值有：
	<i>LIO_OUT_EN0</i>	Port 0 為 LDO 0 ~ LDO 3
	<i>LIO_OUT_EN1</i>	Port 1 為 LDO 4 ~ LDO 7
	<i>LIO_OUT_EN2</i>	Port 2 為 LDO 8 ~ LDO 11
	<i>LIO_OUT_EN3</i>	Port 3 為 LDO 12 ~ LDO 15
	<i>LIO_OUT_EN4</i>	Port 4 為 LDO 16 ~ LDO 19
	<i>LIO_OUT_EN5</i>	Port 5 為 LDO 20 ~ LDO 23
	<i>LIO_OUT_EN6</i>	Port 6 為 LDO 24 ~ LDO 27
	wCardIndex	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	關閉成功
	FALSE	指定之 <i>wPort</i> 不在設定範圍內



Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.4.8 EPCIO_LIO_SetLDIIntType()

BOOL EPCIO_LIO_SetLDIIntType(WORD wPoint, WORD wType, WORD wCardIndex)

Description 設定 LIO 中具有中斷型態之近端輸入接點與中斷觸發型態，本函式設定完後必須再呼叫 EPCIO_LIO_EnableLDIInt() 設定觸發中斷條件。

Parameters **wPoint** 近端輸入接點編號，可設定值有：
LIO_LDI0 近端輸入接點 0 (LDI 0)
 ~ *LIO_LDI7* ~ 近端輸入接點 7 (LDI 7)
wType 中斷觸發型態，可設定值有：
LIO_INT_RISE 上緣觸發(預設值)
LIO_INT_FALL 下緣觸發
LIO_INT_LEVEL 轉態觸發
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
 FALSE 指定之 *wPoint* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_EnableLDIInt()

2.4.9 EPCIO_LIO_EnableLDIInt()

BOOL EPCIO_LIO_EnableLDIInt(WORD wPoint, WORD wCardIndex)

Description 開啟近端輸入接點訊號觸發中斷條件，呼叫本函式前必須先呼叫 EPCIO_LIO_SetLDIIntType() 設定中斷觸發型態。

Parameters **wPoint** 近端輸入接點編號，可設定值有：
LIO_LDI0 近端輸入接點 0 (LDI 0)
 ~ *LIO_LDI7* ~ 近端輸入接點 7 (LDI 7)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value	TRUE	設定成功
	FALSE	指定之 <i>wPoint</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_LIO_SetLDIIntType()	

2.4.10 EPCIO_LIO_DisableLDIInt()

BOOL EPCIO_LIO_DisableLDIInt(WORD wPoint, WORD wCardIndex)

Description	關閉近端輸入接點訊號觸發中斷條件。	
Parameters	<i>wPoint</i>	近端輸入接點編號，可設定值有：
	<i>LIO_LDI0</i>	近端輸入接點 0 (LDI 0)
	~ <i>LIO_LDI7</i>	~ 近端輸入接點 7 (LDI 7)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wPoint</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.4.11 EPCIO_LIO_SetTimer()

BOOL EPCIO_LIO_SetTimer(DWORD dwTime, WORD wCardIndex)

Description	設定計時器的計時週期，配合呼叫 EPCIO_LIO_EnableTimer() 及 EPCIO_LIO_EnableTimerInt()，計時終了可觸發 Timer 中斷功能。Timer 計時時間之單位為 System Clock。	
Parameters	<i>dwTime</i>	24-bit 計時器，設定範圍 (0 ~ 2 ²⁴ -1)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>dwTime</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_LIO_EnableTimer()	EPCIO_LIO_EnableTimerInt()

2.4.12 EPCIO_LIO_EnableTimer()

BOOL EPCIO_LIO_EnableTimer(WORD wCardIndex)

Description 開啟 Timer 計時功能，呼叫本函式前請先呼叫 EPCIO_LIO_SetTimer() 設定 Timer 計時器值。

Parameters **wCardIndex** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功

FALSE 開啟不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_SetTimer()

2.4.13 EPCIO_LIO_DisableTimer()

BOOL EPCIO_LIO_DisableTimer(WORD wCardIndex)

Description 關閉 Timer 計時功能。

Parameters **wCardIndex** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功

FALSE 關閉不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.4.14 EPCIO_LIO_EnableTimerInt()

BOOL EPCIO_LIO_EnableTimerInt(WORD wCardIndex)

Description 開啟 Timer 中斷觸發功能，呼叫本函式前請先呼叫 EPCIO_LIO_SetTimer() 設定 Timer 計時器值以及呼叫 EPCIO_LIO_EnableTimer() 開啟 Timer 計時功能。

Parameters **wCardIndex** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功

FALSE 開啟不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_SetTimer() EPCIO_LIO_EnableTimer()

2.4.15 EPCIO_LIO_DisableTimerInt()

BOOL EPCIO_LIO_DisableTimerInt(WORD wCardIndex)

Description 關閉 Timer 中斷觸發功能。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功

FALSE 關閉不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.4.16 EPCIO_LIO_SetWDogTimer()

BOOL EPCIO_LIO_SetWDogTimer(WORD wTime, WORD wCardIndex)

Description 設定 Watch Dog 計時器比較值。Watch Dog 計時器比較值的 Time Base 為使用計時器之計時時間，請呼叫 EPCIO_LIO_SetTimer() 設定計時功能。呼叫 EPCIO_LIO_EnableWDogTimer() 可以啟動 Watch Dog 功能。

Parameters *wTime* Watch Dog 計時器比較值

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功

FALSE 設定不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_SetTimer() EPCIO_LIO_EnableWDogTimer()

2.4.17 EPCIO_LIO_EnableWDogTimer()

BOOL EPCIO_LIO_EnableWDogTimer(WORD wCardIndex)

Description 開啟 Watch Dog 計時器功能，必須先呼叫 EPCIO_LIO_SetWDogTimer() 設定 Watch Dog 計時器比較值。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功

	FALSE	開啟不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_LIO_SetWDogTimer()	

2.4.18 EPCIO_LIO_DisableWDogTimer()

BOOL EPCIO_LIO_DisableWDogTimer(WORD wCardIndex)

Description 關閉 Watch Dog Timer 功能。

Parameters ***wCardIndex*** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功

FALSE 關閉不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.4.19 EPCIO_LIO_SetWDogReset()

BOOL EPCIO_LIO_SetWDogReset(DWORD dwTime, WORD wCardIndex)

Description 設定 Watch Dog 計時器 Reset 訊號持續時間，Watch Dog 計時器計時終了後將觸發硬體 Reset 訊號，Reset 維持時間可透過本函式規劃。設定單位為 System Clock(40MHz)。

Parameters ***dwTime*** 設定 24-bit 的 Reset 暫存器，設定範圍(0 ~ $2^{24}-1$)，單位為 25ns

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功

FALSE 指定之 *dwTime* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_EnableWDogTimer()

2.4.20 EPCIO_LIO_RefreshWDogTimer()

BOOL EPCIO_LIO_RefreshWDogTimer(WORD wCardIndex)



Description 更新 Watch Dog 計時器時間。當使用者開啟 Watch Dog 功能後，必須在固定時間內呼叫本函式清除 Watch Dog 計時器內容。否則一旦計時終了後將發生硬體 Reset 動作。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功

FALSE 設定不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.4.21 EPCIO_LIO_GetOverTravelUp()

BOOL EPCIO_LIO_GetOverTravelUp(WORD wPoint, WORD* pwOT, WORD wCardIndex)

Description 讀取設定軸是否超過正方向之行程極限。若機台有可能發生撞機的危險，使用者應立即作緊急處理。本函式無法利用中斷方式讀取，檢查方式必須透過 CPU 查看方式。

Parameters *wPoint* 正方向行程溢位狀態編號，可設定值有：

LIO_OT0 第 0 軸行程溢位輸入點

~ *LIO_OT5* ~ 第 5 軸行程溢位輸入點

pwOT 指向一 WORD 值，用來存放正行程溢位狀態值，0 代表無溢位；1 代表發生溢位

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功

FALSE 讀取不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.4.22 EPCIO_LIO_GetOverTravelDown()

BOOL EPCIO_LIO_GetOverTravelDown(WORD wPoint, WORD* pwOT, WORD wCardIndex)

Description 讀取設定軸是否超過負方向之行程極限。若機台有可能發生撞機的危險，使用



者應立即作緊急處理。本函式無法利用中斷方式讀取，檢查方式必須透過 CPU 查看方式。

Parameters	wPoint	負方向行程溢位狀態編號，可設定值有：
	<i>LIO_OT0</i>	第 0 軸行程溢位輸入點
	~ <i>LIO_OT5</i>	~ 第 5 軸行程溢位輸入點
	pwOT	指向一 WORD 值，用來存放負行程溢位狀態值，0 代表無溢位；1 代表發生溢位
	wCardIndex	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	讀取不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.4.23 EPCIO_LIO_GetHomeSensor()

BOOL EPCIO_LIO_GetHomeSensor(WORD wPoint, WORD* pdwHome, WORD wCardIndex)

Description	讀取設定軸 HOME 點狀態。本函式無法利用中斷方式讀取，檢查方式必須透過 CPU 查看方式。	
Parameters	wPoint	HOME 點編號，可設定值有：
	<i>LIO_HOME0</i>	第 0 軸 HOME 點輸入點
	~ <i>LIO_HOME5</i>	~ 第 5 軸 HOME 點輸入點
	pdwHome	指向一 WORD 值，用來存放 HOME 點狀態值；0 代表無觸發；1 代表 HOME 點觸發
	wCardIndex	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	讀取不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.4.24 EPCIO_LIO_GetEmgcStopStatus()

BOOL EPCIO_LIO_GetEmgcStopStatus(WORD* pwStatus, WORD wCardIndex)

Description 讀取緊急停止開關點狀態。本函式無法利用中斷方式讀取，檢查方式必須透過 CPU 查看方式。

Parameters *pwStatus* 指向一 WORD 值，用來存放緊急停止開關輸入狀態值；0 代表無觸發；1 代表觸發緊急停止開關

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功
FALSE 讀取不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.4.25 EPCIO_LIO_ServoOff()

BOOL EPCIO_LIO_ServoOff(WORD wPoint, WORD wCardIndex)

Description 開啟指定軸之禁止輸入接點功能。本接點可連接馬達驅動器的禁止輸入接點，當呼叫本函式設定後，指定軸將不再接受位置命令或速度命令之輸入。當呼叫初始化函式設定後，內定狀態為開啟禁止輸入功能。

Parameters *wPoint* 禁止輸入開關接點編號，可設定值有：

LIO_INH0 第 0 軸禁止輸入開關接點
~ *LIO_INH5* ~ 第 5 軸禁止輸入開關接點

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功
FALSE 開啟不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See also EPCIO_LIO_ServoOn()

2.4.26 EPCIO_LIO_ServoOn()

BOOL EPCIO_LIO_ServoOn(WORD wPoint, WORD wCardIndex)

Description	關閉指定軸之禁止輸入接點功能。本接點可連接馬達驅動器的禁止輸入接點，當呼叫本函式設定後，指定軸將可接受來自於 EPCIO Series 控制卡的位置命令或速度命令之輸入。當呼叫初始化函式設定後，內定狀態為開啟禁止輸入功能。
Parameters	wPoint 禁止輸入開關接點編號，可設定值有： LIO_INH0 第 0 軸禁止輸入開關接點 ~ LIO_INH5 ~ 第 5 軸禁止輸入開關接點 wCardIndex 欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE 關閉成功 FALSE 關閉不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡
See Also	EPCIO_LIO_ServoOff()

2.4.27 EPCIO_LIO_EnablePrdy()

BOOL EPCIO_LIO_EnablePrdy(WORD wCardIndex)

Description	開啟 Position Ready 輸出接點功能。本輸出點可連接電源開關控制接點，呼叫本函式設定後，控制接點將被導通。當呼叫初始化函式設定後，內定狀態為關閉 Position Ready 輸出功能。
Parameters	wCardIndex 欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE 開啟成功 FALSE 開啟不成功
Remark	本函式適用於所有 EPCIO Series 運動控制卡
See Also	EPCIO_LIO_DisablePrdy()

2.4.28 EPCIO_LIO_DisablePrdy()

BOOL EPCIO_LIO_DisablePrdy(WORD wCardIndex)

Description	關閉 Position Ready 輸出接點功能。本輸出點可連接電源開關控制接點，呼叫本函式設定後，控制接點將被開路。當呼叫初始化函式設定後，內定狀態為關
-------------	---

閉 Position Ready 輸出功能。

Parameters **wCardIndex** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功

FALSE 關閉不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_EnablePrdy()

2.4.29 EPCIO_LIO_EnablePulseDAC()

BOOL EPCIO_LIO_EnablePulseDAC(WORD wCardIndex)

Description 開啟 EPCIO Series 模組上位置命令及電壓命令輸出功能。本函式設定後，輸出功能將被開啟。當呼叫初始化函式設定後，內定狀態為關閉輸出功能。

Parameters **wCardIndex** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功

FALSE 開啟不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_LIO_DisablePulseDAC()

2.4.30 EPCIO_LIO_DisablePulseDAC()

BOOL EPCIO_LIO_DisablePulseDAC(WORD wCardIndex)

Description 關閉 EPCIO Series 模組上位置命令及電壓命令輸出功能。本函式設定後，輸出功能將被關閉。當呼叫初始化函式設定後，內定狀態為關閉輸出功能。

Parameters **wCardIndex** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功

FALSE 關閉不成功

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.5. 遠端輸出入接點控制(RIO)

2.5.1 EPCIO_RIO0_GetIntCondition()

BOOL EPCIO_RIO0_GetIntCondition(RIOINT* psInterrupt, WORD wCardIndex)

Description 讀取 Remote I/O Set 0 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource() 判斷是否為 Remote I/O Set 0 所發生，若是則呼叫本函式讀取中斷發生源。

Parameters *psInterrupt* 指向一 RIOINT 結構指標，用來存放 Remote I/O Set 0 中斷發生源訊號編碼；RIOINT 結構定義如下：

```
typedef struct _RIO_INT
{
    BYTE    S0DI0;
    BYTE    S0DI1;
    BYTE    S0DI2;
    BYTE    S0DI3;
    BYTE    S1DI0;
    BYTE    S1DI1;
    BYTE    S1DI2;
    BYTE    S1DI3;
    BYTE    S2DI0;
    BYTE    S2DI1;
    BYTE    S2DI2;
    BYTE    S2DI3;
    BYTE    FAIL;
}RIOINT;
```

各狀態變數說明如下：

<i>S0DI0 ~ S0DI3</i>	Remote I/O Slave 0 的輸入接點 DI 0 ~ DI 3 觸發中斷
<i>S1DI0 ~ S1DI3</i>	Remote I/O Slave 1 的輸入接點 DI 0 ~ DI 3 觸發中斷
<i>S2DI0 ~ S2DI3</i>	Remote I/O Slave 2 的輸入接點 DI 0 ~ DI 3 觸發中斷
<i>FAIL</i>	Remote I/O Set 資料傳輸狀態
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Return Value	TRUE	讀取成功
	FALSE	讀取失敗或 RIO Set 0 未觸發中斷
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See also	EPCIO_GetIntSource()	

2.5.2 EPCIO_RIO1_GetIntCondition()

BOOL EPCIO_RIO1_GetIntCondition(RIOINT* psInterrupt, WORD wCardIndex)

Description	讀取 Remote I/O Set 1 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource() 判斷是否為 Remote I/O Set 1 所發生，若是則呼叫本函式讀取中斷發生源。	
Parameters	<i>psInterrupt</i>	指向一 RIOINT 結構指標，用來存放 Remote I/O Set 1 中斷發生源訊號編碼；RIOINT 結構定義與各狀態變數說明請參考函式 EPCIO_RIO0_GetIntCondition()
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	讀取失敗或 RIO Set 1 未觸發中斷
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See also	EPCIO_GetIntSource()	

2.5.3 EPCIO_RIO_GetInputValue()

BOOL EPCIO_RIO_GetInputValue(WORD wSet, WORD wSlave, WORD wPort, WORD* pwInput, WORD wCardIndex)

Description	讀取指定 Remote I/O 的 16 點輸入訊號狀態值	
Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	<i>wSlave</i>	Remote I/O Slave 編號
	<i>RIO_SLAVE0</i>	Remote I/O Slave 0



	<i>RIO_SLAVE1</i>	Remote I/O Slave 1
	<i>RIO_SLAVE2</i>	Remote I/O Slave 2
	<i>wPort</i>	將 Remote I/O 64 個輸入點分為四組(Port)，每組包含 16 個輸入點
	<i>RIO_PORT0</i>	Remote I/O 輸入點 0 ~ 15(DI 0 ~ DI 15)
	<i>RIO_PORT1</i>	Remote I/O 輸入點 16 ~ 31(DI 16 ~ DI 31)
	<i>RIO_PORT2</i>	Remote I/O 輸入點 32 ~ 47(DI 32 ~ DI 47)
	<i>RIO_PORT3</i>	Remote I/O 輸入點 48 ~ 63(DI 48 ~ DI 63)
	<i>pwInput</i>	指向一 WORD 值，用來存放 16 個輸入點的訊號狀態
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wSet</i> , <i>wSlave</i> 或 <i>wPort</i> 不在設定範圍內
Remark	本函式適用所有 EPCIO Series 運動控制卡	

2.5.4 EPCIO_RIO_SetOutputValue()

BOOL EPCIO_RIO_SetOutputValue(WORD wSet, WORD wSlave, WORD wPort, WORD wOutput, WORD wCardIndex)

Description 設定指定 Remote I/O 的 16 點輸出訊號狀態值。

Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	<i>wSlave</i>	Remote I/O Slave 編號
	<i>RIO_SLAVE0</i>	Remote I/O Slave 0
	<i>RIO_SLAVE1</i>	Remote I/O Slave 1
	<i>RIO_SLAVE2</i>	Remote I/O Slave 2
	<i>wPort</i>	將 Remote I/O 64 個輸出點分為四組(Port)，每組包含 16 個輸入點
	<i>RIO_PORT0</i>	Remote I/O 輸出點 0 ~ 15(DO 0 ~ DO 15)
	<i>RIO_PORT1</i>	Remote I/O 輸出點 16 ~ 31(DO 16 ~ DO 31)
	<i>RIO_PORT2</i>	Remote I/O 輸出點 32 ~ 47(DO 32 ~ DO 47)



<i>RIO_PORT3</i>	Remote I/O 輸出點 48 ~ 63(DO 48 ~ DO 63)
<i>wOutput</i>	16 個輸出點的訊號狀態
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value TRUE	設定成功
FALSE	指定之 <i>wSet</i> , <i>wSlave</i> 或 <i>wPort</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡

2.5.5 EPCIO_RIO_GetOutputValue()

BOOL EPCIO_RIO_GetOutputValue(WORD wSet, WORD wSlave, WORD wPort, WORD* wOutput, WORD wCardIndex)

Description 讀取指定 Remote I/O 的 16 點輸出訊號狀態值。

Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	<i>wSlave</i>	Remote I/O Slave 編號
	<i>RIO_SLAVE0</i>	Remote I/O Slave 0
	<i>RIO_SLAVE1</i>	Remote I/O Slave 1
	<i>RIO_SLAVE2</i>	Remote I/O Slave 2
	<i>wPort</i>	將 Remote I/O 64 個輸出點分為四組(Port)，每組包含 16 個輸出點
	<i>RIO_PORT0</i>	Remote I/O 輸出點 0 ~ 15(DO 0 ~ DO 15)
	<i>RIO_PORT1</i>	Remote I/O 輸出點 16 ~ 31(DO 16 ~ DO 31)
	<i>RIO_PORT2</i>	Remote I/O 輸出點 32 ~ 47(DO 32 ~ DO 47)
	<i>RIO_PORT3</i>	Remote I/O 輸出點 48 ~ 63(DO 48 ~ DO 63)
	<i>wOutput</i>	指向一 WORD 值，用來存放 16 個輸出點的訊號狀態
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value TRUE		讀取成功
FALSE		指定之 <i>wSet</i> , <i>wSlave</i> 或 <i>wPort</i> 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.5.6 EPCIO_RIO_GetTransStatus()

BOOL EPCIO_RIO_GetTransStatus(WORD wSet, WORD* pwStatus, WORD wCardIndex)

Description 讀取目前 Remote I/O Master 端是否收到來自 Slave 資料傳輸的狀態。若傳輸已停止，則呼叫 EPCIO_RIO_GetMasterStatus()及 EPCIO_RIO_GetSlaveStatus()分辨產生錯誤的為 Master 端或 Slave 端。

Parameters *wSet* Remote I/O Set 編號
RIO_SET0 Remote I/O Set 0
RIO_SET1 Remote I/O Set 1
pwStatus 指向一 WORD 值，用來存放 Remote I/O 資料傳輸狀態；
1 代表 Remote I/O Master 端有接收到來自 Slave 端的訊號正常，0 則否
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取狀態成功
FALSE 指定之 *wSet* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

See Also EPCIO_RIO_GetMasterStatus() EPCIO_RIO_GetSlaveStatus()

2.5.7 EPCIO_RIO_GetMasterStatus()

BOOL EPCIO_RIO_GetMasterStatus(WORD wSet, WORD* pwStatus, WORD wCardIndex)

Description 讀取目前 RIO Master 端傳送資料到 Slave 的狀態。若傳送發生錯誤，再由呼叫 EPCIO_RIO_GetSlaveFail()分辨產生錯誤的 Slave 編號。

Parameters *wSet* Remote I/O Set 編號
RIO_SET0 Remote I/O Set 0
RIO_SET1 Remote I/O Set 1
pwStatus 指向一 WORD 值，用來存放 Remote I/O 資料傳輸狀態；
1 代表 Remote I/O Master 端有接收到來自 Slave 端的訊號

		正常，0 則否
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取狀態成功
	FALSE	指定之 <i>wSet</i> 不在設定範圍內
Remark	本函式適用所有 EPCIO Series 運動控制卡	
See Also	EPCIO_RIO_GetSlaveStatus()	EPCIO_RIO_GetSlaveFail()

2.5.8 EPCIO_RIO_GetSlaveStatus()

BOOL EPCIO_RIO_GetSlaveStatus(WORD wSet, WORD* pwStatus, WORD wCardIndex)

Description 讀取目前 RIO Slave 端接收 Master 資料的狀態，若接收發生錯誤，再由呼叫 EPCIO_RIO_GetSlaveFail() 分辨產生錯誤的 Slave 編號。

Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	<i>pwStatus</i>	指向一 WORD 值，用來存放 Remote I/O 資料傳輸狀態； 1 代表 Remote I/O Master 端有接收到來自 Slave 端的訊號 正常，0 則否
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取狀態成功
	FALSE	指定之 <i>wSet</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_RIO_GetMasterStatus()	EPCIO_RIO_GetSlaveFail()

2.5.9 EPCIO_RIO_GetSlaveFail()

BOOL EPCIO_RIO_GetSlaveFail(WORD wSet, WORD* pwStatus, WORD wCardIndex)

Description 讀取目前 RIO Slave 端接收 Master 資料發生錯誤時的 Slave 編號。

Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0

<i>RIO_SET1</i>	Remote I/O Set 1
<i>pwStatus</i>	指向一 WORD 值，用來存放傳輸資料錯誤的 Slave 編號： 1：Remote I/O Slave 0 2：Remote I/O Slave 1 3：Remote I/O Slave 2
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE 讀取狀態成功 FALSE 指定之 <i>wSet</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡
See Also	EPCIO_RIO_GetMasterStatus() EPCIO_RIO_GetSlaveFail()

2.5.10 EPCIO_RIO_SetClockDivider()

BOOL EPCIO_RIO_SetClockDivider(WORD wSet, WORD wDivider, WORD wCardIndex)

Description 設定 Remote I/O 傳輸資料的傳輸時脈。傳輸時脈為 System Clock (40MHz)除以 $2 \times (wDivider + 1)$ ，*wDivider* 內定設定值為 0。

Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	<i>wDivider</i>	Remote I/O 時脈除頻值(0 ~ 255)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 指定之 *wSet* 不在設定範圍內

Remark 本函式適用所有 EPCIO Series 運動控制卡

2.5.11 EPCIO_RIO_SetIntType()

BOOL EPCIO_RIO_SetIntType(WORD set, WORD wSlave, WORD wPoint, WORD wType, WORD wCardIndex)

Description	設定 Remote I/O 輸入點中斷訊號觸發型態。本函式設定完成後，必須呼叫 EPCIO_RIO_EnableInputInt()，開啟中斷功能。	
Parameters	wSet	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	wSlave	Remote I/O Slave 編號
	<i>RIO_SLAVE0</i>	Remote I/O Slave 0
	<i>RIO_SLAVE1</i>	Remote I/O Slave 1
	<i>RIO_SLAVE2</i>	Remote I/O Slave 2
	wPoint	Remote I/O Slave 輸入點
	<i>RIO_DI0</i>	Remote I/O Slave 輸入點 0(DI 0)
	<i>RIO_DI1</i>	Remote I/O Slave 輸入點 1(DI 1)
	<i>RIO_DI2</i>	Remote I/O Slave 輸入點 2(DI 2)
	<i>RIO_DI3</i>	Remote I/O Slave 輸入點 2(DI 3)
	wType	觸發型態，可設定值有：
	<i>RIO_INT_RISE</i>	上緣觸發
	<i>RIO_INT_FALL</i>	下緣觸發
	<i>RIO_INT_LEVEL</i>	轉態觸發
	wCardIndex	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wSet</i> , <i>wSlave</i> , <i>wPoint</i> 或 <i>wType</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_RIO_EnableInputInt()	

2.5.12 EPCIO_RIO_EnableInputInt()

**BOOL EPCIO_RIO_EnableInputInt(WORD set, WORD wSlave, WORD wPoint,
WORD wCardIndex)**

Description Remote I/O 每一 Set 具有 3 組 Slaves，每一組 Slave 的前四點輸入點(DI 0、DI

1、DI 2、DI 3)可引發中斷。此函式用來啟動 DI 0 ~ DI 3 中斷功能。

Parameters	wSet	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	wSlave	Remote I/O Slave 編號
	<i>RIO_SLAVE0</i>	Remote I/O Slave 0
	<i>RIO_SLAVE1</i>	Remote I/O Slave 1
	<i>RIO_SLAVE2</i>	Remote I/O Slave 2
	wPoint	Remote I/O 輸入點
	<i>RIO_DI0</i>	Remote I/O Slave 輸入點 0(DI 0)
	<i>RIO_DI1</i>	Remote I/O Slave 輸入點 1(DI 1)
	<i>RIO_DI2</i>	Remote I/O Slave 輸入點 2(DI 2)
	<i>RIO_DI3</i>	Remote I/O Slave 輸入點 3(DI 3)
	wCardIndex	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wSet</i> , <i>wSlave</i> 或 <i>wPoint</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_RIO_SetIntType()	

2.5.13 EPCIO_RIO_DisableInputInt()

BOOL EPCIO_RIO_DisableInputInt(WORD set, WORD wSlave, WORD wPoint, WORD wCardIndex)

Description Remote I/O 每一 Set 具有 3 組 Slaves，每一組 Slave 的前四點輸入點(DI 0、DI 1、DI 2、DI 3)可引發中斷。此函式用來關閉 DI 0 ~ DI 3 中斷功能。

Parameters	wSet	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	wSlave	Remote I/O Slave 編號
	<i>RIO_SLAVE0</i>	Remote I/O Slave 0

	<i>RIO_SLAVE1</i>	Remote I/O Slave 1
	<i>RIO_SLAVE2</i>	Remote I/O Slave 2
	<i>wPoint</i>	Remote I/O Slave 輸入點
	<i>RIO_DI0</i>	Remote I/O Slave 輸入點 0(DI 0)
	<i>RIO_DI1</i>	Remote I/O Slave 輸入點 1(DI 1)
	<i>RIO_DI2</i>	Remote I/O Slave 輸入點 2(DI 2)
	<i>RIO_DI3</i>	Remote I/O Slave 輸入點 3(DI 3)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之參 <i>wSet</i> , <i>wSlave</i> 或 <i>wPoint</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.5.14 EPCIO_RIO_SetTransError()

BOOL EPCIO_RIO_SetTransError(WORD wSet, WORD wErrorTime, WORD wCardIndex)

Description	設定 Remote I/O 傳輸錯誤時系統可自動重送次數，當設定為 0 時，系統將於錯誤發生後會自動重傳最多 16 次。	
Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	<i>wErrorTime</i>	資料傳輸錯誤重送次數(設定範圍：0 ~ 15)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wErrorTime</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	

2.5.15 EPCIO_RIO_EnableSetControl()

BOOL EPCIO_RIO_EnableSetControl(WORD wSet, WORD wCardIndex)

Description	開啟指定之 Remote I/O Set 之控制功能，而該 Set 之各個 Slave 功能需再呼叫	
-------------	--	--

EPCIO_RIO_EnableSlaveControl() 才會開啟控制功能。

Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wSet</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_RIO_EnableSlaveControl()	EPCIO_RIO_DisableSlaveControl()

2.5.16 EPCIO_RIO_DisableSetControl()

BOOL EPCIO_RIO_DisableSetControl(WORD wSet, WORD wCardIndex)

Description 關閉指定之 Remote I/O Set 之控制功能，該 Set 之所有 Slaves 功能也會一併關閉。

Parameters	<i>wSet</i>	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	關閉成功
	FALSE	指定之 <i>wSet</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_RIO_EnableSlaveControl()	EPCIO_RIO_DisableSlaveControl()

2.5.17 EPCIO_RIO_EnableSlaveControl()

BOOL EPCIO_RIO_EnableSlaveControl(WORD set, WORD wSlave, WORD wCardIndex)

Description 開啟指定的 Remote I/O Slave 功能。Slave 功能開啟後，必須再呼叫 EPCIO_RIO_EnableSetControl() 開啟 Set 功能，Remote I/O 模組才開始做傳送與接收。

Parameters	wSet	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	wSlave	Remote I/O Slave 編號
	<i>RIO_SLAVE0</i>	Remote I/O Slave 0
	<i>RIO_SLAVE1</i>	Remote I/O Slave 1
	<i>RIO_SLAVE2</i>	Remote I/O Slave 2
	wCardIndex	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	開啟成功
	FALSE	指定之 <i>wSet</i> 或 <i>wSlave</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_RIO_EnableSetControl()	EPCIO_RIO_DisableSetControl()

2.5.18 EPCIO_RIO_DisableSlaveControl()

BOOL EPCIO_RIO_DisableSlaveControl(WORD wSet, WORD wSlave, WORD wCardIndex)

Description 關閉指定的 Remote I/O Slave 功能。本函式可單獨關閉指定的 Slave 功能。

Parameters	wSet	Remote I/O Set 編號
	<i>RIO_SET0</i>	Remote I/O Set 0
	<i>RIO_SET1</i>	Remote I/O Set 1
	wSlave	Remote I/O Slave 編號
	<i>RIO_SLAVE0</i>	Remote I/O Slave 0
	<i>RIO_SLAVE1</i>	Remote I/O Slave 1
	<i>RIO_SLAVE2</i>	Remote I/O Slave 2
	wCardIndex	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	關閉成功
	FALSE	指定之 <i>wSet</i> 或 <i>wSlave</i> 不在設定範圍內
Remark	本函式適用於所有 EPCIO Series 運動控制卡	
See Also	EPCIO_RIO_EnableSetControl()	EPCIO_RIO_DisableSetControl()



2.5.19 EPCIO_RIO_EnableTransInt()

BOOL EPCIO_RIO_EnableTransInt(WORD wSet, WORD wCardIndex)

Description 開啟 Remote I/O 通訊失敗中斷功能。

Parameters ***wSet*** Remote I/O Set 編號
RIO_SET0 Remote I/O Set 0
RIO_SET1 Remote I/O Set 1
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功
FALSE 指定之 *wSet* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.5.20 EPCIO_RIO_DisableTransInt()

BOOL EPCIO_RIO_DisableTransInt(WORD wSet, WORD wCardIndex)

Description 關閉 Remote I/O 通訊失敗中斷功能。

Parameters ***wSet*** Remote I/O Set 編號
RIO_SET0 Remote I/O Set 0
RIO_SET1 Remote I/O Set 1
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功
FALSE 指定之 *wSet* 不在設定範圍內

Remark 本函式適用於所有 EPCIO Series 運動控制卡

2.6. 類比轉數位輸入控制(ADC)

2.6.1 EPCIO_ADC_GetIntCondition()

BOOL EPCIO_ADC_GetIntCondition(ADCINT* psInterrupt, WORD wCardIndex)

Description 讀取 ADC 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource()判斷是否為 ADC 所發生，若是則呼叫本函式讀取中斷發生源。

Parameters *psInterrupt* 指向一 ADCINT 結構指標，用來存放 ADC 中斷發生源訊號編碼；RIOINT 結構定義如下：

```
typedef struct _ADC_INT
{
    BYTE    COMP0;
    BYTE    COMP1;
    BYTE    COMP2;
    BYTE    COMP3;
    BYTE    COMP4;
    BYTE    COMP5;
    BYTE    COMP6;
    BYTE    COMP7;
    BYTE    CONV;
    BYTE    TAG
}ADCINT;
```

各狀態變數說明如下：

COMP0 ADC Channel 0
~ *COMP7* ~ ADC Channel 7 比較器觸發中斷
CONV ADC 任意 Channel 完成電壓轉換
TAG ADC 標籤 Channel 完成電壓轉換(在同一時間止允許一個
特定 Channel 被貼上標籤)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功

	FALSE	讀取失敗或 ADC 未觸發中斷
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	
See also	EPCIO_GetIntSource()	

2.6.2 EPCIO_ADC_GetWorkStatus()

BOOL EPCIO_ADC_GetWorkStatus(WORD* pwStatus, WORD wCardIndex)

Description 讀取目前 ADC 的工作狀態，確認轉換過程是否完成。

Parameters *pwStatus* 指向一 WORD 值，用來存放 ADC 工作狀態

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功

FALSE 讀取不成功

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

2.6.3 EPCIO_ADC_GetInput()

BOOL EPCIO_ADC_GetInput(WORD wChannel, float* pfVoltage, WORD wCardIndex)

Description 讀取指定的 ADC Channel 輸入之直流電壓值。單極性(Unipolar)轉換型式時輸入電壓的有效值為 0 ~ 10 V；雙極性(Bipolar)轉換型式時輸入電壓的有效值為 -5 ~ 5 V。

Parameters *wChannel* ADC Channel 編號(0 ~ 7)

pfVoltage 指向一 float 值，用來存放 ADC Channel 直流電壓輸入值

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功

FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

2.6.4 EPCIO_ADC_SetCompMask()

BOOL EPCIO_ADC_SetCompMask(WORD wMask, WORD wCardIndex)

Description	設定 ADC 比較模式下輸入電壓值與設定之比較值比較時，遮蔽最小幾個 bit 不作比較動作，本函式可降低比較器的靈敏度，避免因輸入電壓跳動造成中斷持續發生。設定本函式後須再搭配呼叫 EPCIO_ADC_SetCompType() 及 EPCIO_ADC_EnableCompInt()，則當該 ADC Channel 輸入電壓與比較型式條件成立後，可產生 ADC 中斷觸發訊號。	
Parameters	wMask	ADC 比較器的遮蔽位元 ADC_MASK_NO ADC 比較器不遮蔽 ADC_MASK_BIT1 ADC 比較器遮蔽 1 bit ADC_MASK_BIT2 ADC 比較器遮蔽 2 bits ADC_MASK_BIT3 ADC 比較器遮蔽 3 bits wCardIndex 欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 wMask 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	
See Also	EPCIO_ADC_SetCompType()	EPCIO_ADC_EnableCompInt() EPCIO_ADC_SetConvMode()

2.6.5 EPCIO_ADC_SetCompValue()

BOOL EPCIO_ADC_SetCompValue(WORD wChannel, float fVoltage, WORD wCardIndex)

Description	設定 ADC Channel 的輸入電壓比較值。設定本函式後必須再呼叫 EPCIO_ADC_SetCompType() 及 EPCIO_ADC_EnableCompInt()，此時當該 ADC Channel 輸入電壓與比較型式條件成立後，可產生 ADC 中斷觸發訊號。此觸發訊號可用來觸發 DAC 模組輸出一預先規劃之電壓值，同時前兩組 ADC 觸發訊號亦可用來觸發閘鎖(Latch)編碼器的計數值功能。	
Parameters	wChannel	ADC Channel 編號(0 ~ 7)
	fVoltage	ADC Channel 直流電壓輸入值，電壓轉換型式若為單極性(Unipolar)，設定範圍為 0 ~ 10 V；若為雙極性(Bipolar)則設定範圍為-5 ~ 5 V

	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	
See Also	EPCIO_ADC_SetCompType()	EPCIO_ADC_EnableCompInt()
	EPCIO_ADC_SetConvMode()	EPCIO_DAC_SetTrigSource()
	EPCIO_ENC_SetTrigSource()	

2.6.6 EPCIO_ADC_SetCompType()

BOOL EPCIO_ADC_SetCompType(WORD wChannel, WORD wType, WORD wCardIndex)

Description 設定 ADC Channel 電壓比較型式，當比較條件成立便會觸發 ADC 硬體中斷訊號。呼叫本函式後必須呼叫 EPCIO_ADC_EnableCompInt()，則此觸發訊號可用來觸發 DAC 模組輸出一預先規劃之電壓值，同時前兩組 ADC Channel 觸發訊號亦可用來觸發閘鎖(Latch)編碼器的計數值功能。

Parameters

<i>wChannel</i>	ADC Channel 編號(0 ~ 7)
<i>wType</i>	ADC Channel 比較型式，可設定值有：
<i>ADC_COMP_RISE</i>	ADC 輸入電壓由小到大，並通過比較值
<i>ADC_COMP_FALL</i>	ADC 輸入電壓由大到小，並通過比較值
<i>ADC_COMP_LEVEL</i>	ADC 輸入電壓值改變，並通過比較值
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功

FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

See Also EPCIO_ADC_SetCompValue() EPCIO_ADC_EnableCompInt()
EPCIO_DAC_SetTrigSource() EPCIO_ENC_SetTrigSource()

2.6.7 EPCIO_ADC_EnableCompInt()

BOOL EPCIO_ADC_EnableCompInt(WORD wChannel, WORD wCardIndex)

Description	開啟 ADC 產生比較值中斷觸發功能。此觸發訊號可用來觸發 DAC 模組輸出一預先設定之電壓值，同時前兩組 ADC 觸發訊號亦可用來觸發門鎖(Latch)編碼器的計數值功能。	
Parameters	<i>wChannel</i>	ADC Channel 編號(0 ~ 7)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	開啟成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	
See Also	EPCIO_ADC_SetCompValue()	EPCIO_ADC_EnableCompType()
	EPCIO_DAC_SetTrigSource()	EPCIO_ENC_SetTrigSource()

2.6.8 EPCIO_ADC_DisableCompInt()

BOOL EPCIO_ADC_DisableCompInt(WORD wChannel, WORD wCardIndex)

Description	關閉 ADC 產生比較值中斷觸發功能。	
Parameters	<i>wChannel</i>	ADC Channel 編號(0 ~ 7)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	關閉成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	

2.6.9 EPCIO_ADC_EnableTagInt()

BOOL EPCIO_ADC_EnableTagInt(WORD wCardIndex)

Description	開啟標籤 Channel 產生中斷觸發功能。	
Parameters	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	開啟成功
	FALSE	開啟不成功
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	

2.6.10 EPCIO_ADC_DisableTagInt()

BOOL EPCIO_ADC_DisableTagInt(WORD wCardIndex)

Description 關閉標籤 Channel 產生中斷觸發功能。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功

FALSE 關閉不成功

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

2.6.11 EPCIO_ADC_EnableConvInt()

BOOL EPCIO_ADC_EnableConvInt(WORD wCardIndex)

Description 開啟任一 Channel 轉換完成後觸發產生中斷功能。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功

FALSE 開啟不成功

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

2.6.12 EPCIO_ADC_DisableConvInt()

BOOL EPCIO_ADC_DisableConvInt(WORD wCardIndex)

Description 關閉任一 Channel 轉換完成後產生中斷觸發功能。

Parameters *wCardIndex* 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功

FALSE 關閉不成功

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

2.6.13 EPCIO_ADC_SetClockDivider()

BOOL EPCIO_ADC_SetClockDivider(WORD wDivider, WORD wCardIndex)

Description	設定 ADC 串列介面的傳輸時脈。ADC 的傳輸時脈為 System Clock (40Mhz) 除以 $4 \times (wDivider + 1)$ ， <i>wDivider</i> 內定設定值為 0。
Parameters	<i>wDivider</i> ADC 串列介面的時脈除頻值(0 ~ 255) <i>wCardIndex</i> 欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE 設定成功 FALSE 指定之 <i>wDivider</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

2.6.14 EPCIO_ADC_SetConvType()

BOOL EPCIO_ADC_SetConvType(WORD wChannel, WORD wType, WORD wCardIndex)

Description	設定 ADC Channel 電壓轉換型式為雙極性(Bipolar)轉換型式或單極性(Unipolar)轉換型式。
Parameters	<i>wChannel</i> ADC Channel 編號(0 ~ 7) <i>wType</i> ADC 電壓轉換型式，可設定值有： <i>ADC_TYPE_BIP</i> 雙極性(Bipolar)轉換型式 <i>ADC_TYPE_UNI</i> 單極性(Unipolar)轉換型式 <i>wCardIndex</i> 欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE 設定成功 FALSE 指定之 <i>wChannel</i> , <i>wType</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

2.6.15 EPCIO_ADC_GetConvType()

BOOL EPCIO_ADC_GetConvType(WORD wChannel, WORD *pwType, WORD wCardIndex)

Description	讀取目前 ADC Channel 電壓轉換型式。
Parameters	<i>wChannel</i> ADC Channel 編號(0 ~ 7) <i>pwType</i> 指向一 WORD 值，用來存放 ADC 電壓轉換型式 <i>wCardIndex</i> 欲控制的運動控制卡編號(0 ~ 11)

Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> , <i>wType</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	

2.6.16 EPCIO_ADC_EnableConvChannel()

BOOL EPCIO_ADC_EnableConvChannel(WORD wChannel, WORD wCardIndex)

Description 開啟 ADC Channel 輸入電壓類比轉數位功能。本函式設定的轉換 Channel 必須在連續電壓轉換 (Free Run) 模式下才有效，設定完成後必須呼叫 EPCIO_ADC_StartConv() 啟動 ADC 轉換功能。

Parameters *wChannel* ADC Channel 編號(0 ~ 7)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

See Also EPCIO_ADC_StartConv() EPCIO_ADC_SetConvMode()

2.6.17 EPCIO_ADC_DisableConvChannel()

BOOL EPCIO_ADC_DisableConvChannel(WORD wChannel, WORD wCardIndex)

Description 關閉 ADC Channel 輸入電壓類比轉數位功能。

Parameters *wChannel* ADC Channel 編號(0 ~ 7)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡

2.6.18 EPCIO_ADC_SetConvMode()

BOOL EPCIO_ADC_SetConvMode(WORD wMode, WORD wCardIndex)

Description	設定 ADC 電壓轉換型態。	
Parameters	<i>wMode</i>	ADC 電壓轉換型態，可設定值有： <i>ADC_MODE_SINGLE</i> 單次電壓轉換(Single Run) <i>ADC_MODE_FREE</i> 連續電壓轉換(Free Run)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wMode</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	
See Also	EPCIO_ADC_SetSingleChannel()	EPCIO_ADC_EnableConvChannel()

2.6.19 EPCIO_ADC_SetTagChannel()

BOOL EPCIO_ADC_SetTagChannel(WORD wChannel, WORD wCardIndex)

Description	設定某一 ADC Channel 為 Tag Channel，呼叫 EPCIO_ADC_EnableTagInt()可以在 Tag Channel 轉換完成時可觸發中斷訊號。	
Parameters	<i>wChannel</i>	ADC Channel 編號(0 ~ 7)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	
See Also	EPCIO_ADC_EnableTagInt()	

2.6.20 EPCIO_ADC_SetSingleChannel()

BOOL EPCIO_ADC_SetSingleChannel(WORD wChannel, WORD wCardIndex)

Description	設定某一 ADC Channel 為 Single Channel，呼叫 EPCIO_ADC_SetConvMode()將轉換模式設定為單次轉換 (Single Run)，則在呼叫 EPCIO_ADC_StartConv()後，此 Channel 會將電壓值轉換一次，轉換完成後即不再進行轉換，使用者必須再呼叫 EPCIO_ADC_StartConv()才會進行下一個單次的轉換。轉換期間可經	
-------------	--	--

由呼叫 EPCIO_ADC_GetWorkStatus() 確認轉換過程是否完成。

Parameters	<i>wChannel</i>	ADC Channel 編號(0 ~ 7)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	
See Also	EPCIO_ADC_StartConv()	EPCIO_ADC_SetConvMode()
	EPCIO_ADC_GetWorkStatus()	

2.6.21 EPCIO_ADC_StartConv()

BOOL EPCIO_ADC_StartConv(WORD wCardIndex)

Description 開始進行 ADC Channel 類比電壓值轉換功能，本函式必須配合呼叫 EPCIO_ADC_EnableConvChannel()。

Parameters	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	設定不成功
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	
See Also	EPCIO_ADC_EnableConvChannel()	

2.6.22 EPCIO_ADC_StopConv()

BOOL EPCIO_ADC_StopConv(WORD wCardIndex)

Description 停止進行所有 ADC Channel 類比電壓值轉換功能。

Parameters	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	設定不成功
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 具 ADC 元件之運動控制卡	

2.7. 數位轉類比輸出控制(DAC)

2.7.1 EPCIO_DAC_SetOutput()

BOOL EPCIO_DAC_SetOutput(WORD wChannel, float fVoltage, WORD wCardIndex)

Description 設定 DAC Channel 輸出電壓值。DAC 的輸出電壓可呼叫 EPCIO_DAC_SetCmdSource() 設定為軟體直接規劃模式。設定完成後便可呼叫本函式利用 DAC 直接輸出電壓。

Parameters

<i>wChannel</i>	DAC Channel 編號(0 ~ 7)
<i>fVoltage</i>	類比輸出電壓 (-10 ~ 10 V)
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Return Value

TRUE	設定成功
FALSE	指定之 <i>wChannel</i> 或 <i>fVoltage</i> 不在設定範圍內

Remark 本函式僅適用於 EPCIO-6000、EPCIO-6000 及 EPCIO-6000e 之運動控制卡

See Also EPCIO_DAC_SetCmdSource()

2.7.2 EPCIO_DAC_GetOutput()

BOOL EPCIO_DAC_GetOutput(WORD wChannel, float* fVoltage, WORD wCardIndex)

Description 讀取 DAC Channel 輸出電壓值。

Parameters

<i>wChannel</i>	DAC Channel 編號(0 ~ 7)
<i>fVoltage</i>	指向一 float 值，用來存取類比輸出電壓
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)

Return Value

TRUE	設定成功
FALSE	指定之 <i>wChannel</i> 不在設定範圍內

Remark 本函式僅適用於 EPCIO-6000、EPCIO-6000 及 EPCIO-6000e 之運動控制卡

See Also EPCIO_DAC_SetCmdSource()

2.7.3 EPCIO_DAC_SetTrigOutput()

BOOL EPCIO_DAC_SetTrigOutput(WORD wChannel, float fVoltage, WORD wCardIndex)

Description 設定 DAC Channel 在滿足觸發條件時立即輸出的電壓值。當 DAC 規劃為軟體命令模式時，可預先設定一硬體觸發電壓命令在 DAC 模組內，當觸發條件成立時則可由硬體立即把預先設定的命令輸出。

Parameters

- wChannel** DAC Channel 編號(0 ~ 7)
- fVoltage** 類比輸出電壓 (-10 ~ 10 V)
- wCardIndex** 欲控制的運動控制卡編號(0 ~ 11)

Return Value

- TRUE 設定成功
- FALSE 指定之 wChannel 或 fVoltage 不在設定範圍內

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡

See Also EPCIO_DAC_SetCmdSource() EPCIO_DAC_SetTrigSource()
EPCIO_DAC_EnableTrigMode()

2.7.4 EPCIO_DAC_SetTrigSource()

BOOL EPCIO_DAC_SetTrigSource(WORD wChannel, DWORD dwTrigSource, WORD wCardIndex)

Description 設定 DAC Channel 在符合中斷條件時，立即輸出電壓值之功能。每一 DAC Channel 可設定搭配多種中斷條件。設定完本函式後必須再呼叫 EPCIO_DAC_EnableTrigMode() 開啟中斷觸發模式。本功能只能在 DAC 設定為軟體命令模式下使用，請參考 EPCIO_DAC_SetCmdSource()。

Parameters

- wChannel** DAC Channel 編號(0 ~ 7)
- dwTrigSource** DAC 硬體觸發來源，以 bit 表示共 32 個中斷觸發源，可同時設定多種觸發源，可設定值有：

- DAC_TRIG_ENC0** 編碼器 Channel 0
- ~ DAC_TRIG_ENC7** ~ 編碼器 Channel 7 特定計數值
- DAC_TRIG_ADC0** ADC Channel 0
- ~ DAC_TRIG_ADC7** ~ ADC Channel 7 特定電壓輸入值

<i>DAC_TRIG_LDI0</i>	近端輸入接點 LDI 0 訊號輸入
<i>~DAC_TRIG_LDI3</i>	~ 近端輸入接點 LDI 3 訊號輸入
<i>DAC_TRIG_DFI0</i>	雙功能輸入接點 DFI 0 訊號輸入
<i>~DAC_TRIG_DFI3</i>	~ 雙功能輸入接點 DFI 3 訊號輸入
<i>DAC_TRIG_R0DI0</i>	遠端輸入接點 Set 0 的 DI 0 訊號輸入
<i>~DAC_TRIG_R0DI3</i>	~ 遠端輸入接點 Set 0 的 DI 3 訊號輸入
<i>DAC_TRIG_R1DI0</i>	遠端輸入接點 Set 1 的 DI 0 訊號輸入
<i>~DAC_TRIG_R1DI3</i>	~ 遠端輸入接點 Set 1 的 DI 3 訊號輸入
<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE 設定成功
	FALSE 指定之 <i>wChannel</i> 或 <i>dwTrigSource</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡
See Also	EPCIO_DAC_SetCmdSource() EPCIO_DAC_EnableTrigMode()

2.7.5 EPCIO_DAC_SetClockDivider()

BOOL EPCIO_DAC_SetClockDivider(WORD wDivider, WORD wCardIndex)

Description	設定 DAC 串列介面的傳輸時脈。DAC 的傳輸時脈為 System Clock (40MHz) 除以 $4 \times (wDivider + 1)$ ， <i>wDivider</i> 內定設定值為 0。
Parameters	<i>wDivider</i> DAC 傳輸時脈除頻值(0 ~ 255)
	<i>wCardIndex</i> 欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE 設定成功
	FALSE 指定之 <i>wDivider</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡

2.7.6 EPCIO_DAC_EnableTrigMode()

BOOL EPCIO_DAC_EnableTrigMode(WORD wChannel, WORD wCardIndex)

Description	啟動 DAC Channel 搭配中斷產生觸發立即輸出之模式。開啟觸發模式前請先設
-------------	---

定觸發源。

Parameters	<i>wChannel</i>	DAC Channel 編號(0 ~ 7)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡	
See Also	EPCIO_DAC_SetTrigSource()	

2.7.7 EPCIO_DAC_DisableTrigMode()

BOOL EPCIO_DAC_DisableTrigMode(WORD wChannel, WORD wCardIndex)

Description	關閉 DAC Channel 搭配中斷產生觸發立即輸出之模式。	
Parameters	<i>wChannel</i>	DAC Channel 編號(0 ~ 7)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡	

2.7.8 EPCIO_DAC_SetCmdSource()

BOOL EPCIO_DAC_SetCmdSource(WORD wChannel, WORD wType, WORD wCardIndex)

Description	設定 DAC Channel 輸出之命令來源為軟體規劃或由硬體閉迴路(PCL)輸入。當命令來源為 PCL 時則命令來自 PCL 內部的位置誤差值；當設定為軟體規劃模式時可呼叫 EPCIO_DAC_SetOutput()、EPCIO_DAC_SetTrigOutput() 設定觸發模式時的輸出電壓。	
Parameters	<i>wChannel</i>	DAC Channel 編號(0 ~ 7)
	<i>wType</i>	DAC 輸出之命令來源
	<i>DAC_CMD_SOFT</i>	軟體規劃
	<i>DAC_CMD_PCL</i>	硬體閉迴路內部的位置誤差值(PCL Error Counter)

	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 或 <i>wType</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡	
See Also	EPCIO_DAC_SetOutput()	EPCIO_DAC_SetTrig_Output()

2.7.9 EPCIO_DAC_StartConv()

BOOL EPCIO_DAC_StartConv(WORD wCardIndex)

Description 啟動 DAC Channel 進行電壓輸出轉換。

Parameters ***wCardIndex*** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 啟動成功

FALSE 啟動不成功

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡

2.7.10 EPCIO_DAC_StopConv()

BOOL EPCIO_DAC_StopConv(WORD wCardIndex)

Description 停止 DAC Channel 進行電壓輸出轉換。

Parameters ***wCardIndex*** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 停止成功

FALSE 停止未成功

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡

2.8. 硬體位置閉迴路控制設定(PCL)

2.8.1 EPCIO_PCL_GetIntCondition()

BOOL EPCIO_PCL_GetIntCondition(PCLINT* psInterrupt, WORD wCardIndex)

Description 讀取 PCL 中斷訊號之源由，並清除中斷 Latch 值，等待下一次中斷發生。當硬體中斷發生後，經由呼叫 EPCIO_GetIntSource()判斷是否為 PCL 所發生，若是則呼叫本函式讀取中斷發生源。

Parameters *psInterrupt* 指向一 PCLINT 結構指標，用來存放 PCL 中斷發生源訊號編碼；PCLINT 結構定義如下：

```
typedef struct _PCL_INT{  
    BYTE    OV0;  
    BYTE    OV1;  
    BYTE    OV2;  
    BYTE    OV3;  
    BYTE    OV4;  
    BYTE    OV5;  
}PCLINT;
```

各狀態變數說明如下：

OV0 ~ OV5 PCL Channel 0 ~ 5 命令誤差計數值溢位觸發中斷

wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 讀取成功

FALSE 讀取失敗或 PCL 未觸發中斷

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡

See also EPCIO_GetIntSource()

2.8.2 EPCIO_PCL_GetErrorCounter()

***BOOL EPCIO_PCL_GetErrorCounter(WORD wChannel, int* pnErrorCount,
WORD wCardIndex)***

Description	讀取各軸位置命令輸出與編碼器回授命令之命令誤差計數值(Error Counter)， 呼叫本函式時必須先呼叫 EPCIO_PCL_EnableErrorCounter()。	
Parameters	<i>wChannel</i>	PCL Channel 編號(0 ~ 5)
	<i>pnErrorCount</i>	Error counter 值
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	讀取成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡	
See Also	EPCIO_PCL_EnableErrorCounter()	

2.8.3 EPCIO_PCL_SetScaleGain()

BOOL EPCIO_PCL_SetScaleGain(WORD wChannel, WORD wPGain, int nSGain, WORD wCardIndex)

Description	設定閉迴路控制軸之增益值。閉迴路增益值可藉由設定比例項(Kp1)及倍率項(Kp2)組成，增益值 = $Kp1 \times Kp2 / 16$ 。其中 Kp1 為 <i>wPGain</i> ，Kp2 則為 2^{nSGain} 。	
Parameters	<i>wChannel</i>	PCL Channel 編號(0 ~ 5)
	<i>wPGain</i>	比例增益(P Gain) 0 ~ 127
	<i>nSGain</i>	倍率增益(S Gain) -7 ~ 7
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	設定成功
	FALSE	指定之 <i>wChannel</i> 、 <i>wPGain</i> 或 <i>nSGain</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡	

2.8.4 EPCIO_PCL_EnableOverflowInt()

BOOL EPCIO_PCL_EnableOverflowInt(WORD wChannel, WORD wCardIndex)

Description	開啟指定 Channel 的命令誤差計數值(Error Counter)溢位時產生中斷觸發之功能。當位置命令與編碼器實際位置的誤差量超過所能容許範圍時，因而產生生命	
-------------	--	--

令誤差計數值溢位中斷通知，並自動輸出 DAC 電壓值為 0 V。

Parameters	<i>wChannel</i>	PCL Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	開啟成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡	

2.8.5 EPCIO_PCL_DisableOverflowInt()

BOOL EPCIO_PCL_DisableOverflowInt(WORD wChannel, WORD wCardIndex)

Description 關閉指定 Channel 發生命令誤差計數值溢位時產生中斷觸發之功能。

Parameters	<i>wChannel</i>	PCL Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	關閉成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡	

2.8.6 EPCIO_PCL_ClearErrorCounter()

BOOL EPCIO_PCL_ClearErrorCounter(WORD wChannel, WORD wCardIndex)

Description 清除命令誤差計數值以及溢位狀態。

Parameters	<i>wChannel</i>	PCL Channel 編號(0 ~ 5)
	<i>wCardIndex</i>	欲控制的運動控制卡編號(0 ~ 11)
Return Value	TRUE	清除成功
	FALSE	指定之 <i>wChannel</i> 不在設定範圍內
Remark	本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡	

2.8.7 EPCIO_PCL_EnableErrorCounter()

BOOL EPCIO_PCL_EnableErrorCounter(WORD wChannel, WORD wCardIndex)

Description 開啟閉迴路命令誤差計數功能。本函式設定完成後，必須呼叫 EPCIO_PCL_StartControl() 啟動硬體閉迴路計數功能。

Parameters **wChannel** PCL Channel 編號(0 ~ 5)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 開啟成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡

See Also EPCIO_PCL_StartControl()

2.8.8 EPCIO_PCL_DisableErrorCounter()

BOOL EPCIO_PCL_DisableErrorCounter(WORD wChannel, WORD wCardIndex)

Description 關閉指定 Channel 之閉迴路命令誤差計數功能。

Parameters **wChannel** PCL Channel 編號(0 ~ 5)
wCardIndex 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 關閉成功
FALSE 指定之 *wChannel* 不在設定範圍內

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡

2.8.9 EPCIO_PCL_StartControl()

BOOL EPCIO_PCL_StartControl(WORD wCardIndex)

Description 啟動 PCL 控制功能，呼叫本函式前請先呼叫 EPCIO_PCL_EnableErrorCounter() 啟動命令誤差計數功能。

Parameters **wCardIndex** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功
FALSE 設定成功



Remark 本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡

See Also EPCIO_PCL_EnableErrorCounter()

2.8.10 EPCIO_PCL_StopControl()

BOOL EPCIO_PCL_StopControl(WORD wCardIndex)

Description 關閉 PCL 控制功能，本函式關閉所有 Channel 之命令誤差計數功能。

Parameters ***wCardIndex*** 欲控制的運動控制卡編號(0 ~ 11)

Return Value TRUE 設定成功

FALSE 設定不成功

Remark 本函式僅適用於 EPCIO-4000/6000/6000e 之運動控制卡
