EPCIO Series
Device Driver Library
Reference Manual

Version: V.4.00

Date: 2008.07

http://www.epcio.com.tw

Table of Contents

I. Introduction and Description

The EPCIO Series Device Driver Library can be used to drive control cards which are designed and developed with EPCIO ASIC. The library can be used to drive motion control cards having a PCI-Bus interface, such as EPCIO-4000, EPCIO-4005, EPCIO-6000, and EPCIO-6005. A user can link to different libraries according to practical needs. When working in a Windows 95/98, Windows NT, Windows 2000, Windows XP or Windows 7 environment, a dynamic link library (MEPCIOISADrv.DLL or MEPCIOPCIDrv.DLL) is provided. In fact, the functions are used in the same way in all operating systems. An EPCIO Series control board can be successfully driven by calling the desired function(s) as long as the corresponding header file (MEPCIODev.h for WINDOWS) is included.

The device driver library includes more than 100 functions for the user to call. These functions are divided into eight major groups and serve to drive different I/O functions of a control card respectively:

▲Bus Interface	for setting the interrupt and Reset functions
▲DDA Control Interface	for setting motion pulse output control
▲Encoder Counter Interface	for programming encoder input or general counter input
▲Remote Digital I/O Interface	for setting remote I/O control
▲ADC Control Interface	for setting analog-to-digital input control
▲Local I/O Control Interface:	for setting local I/O control
▲PCL Control Interface	for setting a hardware position closed-loop
▲DAC Control Interface	for setting digital-to-analog output control

In terms of use, the prototype declaration and data type declaration of each driver function are defined in the header file “MEpciodev.h”, and the constants, in the header file ”MEpcioIni.h”. The contents of these header files must be included when the EPCIO Series Device Driver Library (EDDL) is used.

The program examples are designed with the EPCIO Series Device Driver Library to demonstrate how each function is applied to the corresponding function module(s). These examples cover the program of pulse output (DDA), encoder counter input (ENC), analog voltage output (DAC), hardware closed-loop control (PCL), analog voltage input (ADC), local I/O control (LIO), remote I/O control (RIO), timer, and watchdog.

The installation program will copy associated file to a specified directory. All the user has to do is go through the installation steps.

II. EPCIO Device Driver Library

II.1. Interface Control

II.1.1 EPCIO4000_Init()

```
BOOL EPCIO4000_Init(    DDAISR      myEPCIO_DDA_ISR,
                        ENCISR      myEPCIO_ENC012_ISR,
                        ENCISR      myEPCIO_ENC345_ISR,
                        ENCISR      myEPCIO_ENC678_ISR,
                        RIOISR      myEPCIO_RIO0_ISR,
                        RIOISR      myEPCIO_RIO1_ISR,
                        ADCISR      myEPCIO_ADC_ISR,
                        LIOISR      myEPCIO_LIO_ISR,
                        PCLISR      myEPCIO_PCL_ISR,
                        WORD        card_index)
```

Parameters

myEPCIO_DDA_ISR Function pointer of the DDA interrupt service routine written by the user

myEPCIO_ENC012_ISR Function pointer of the ENC interrupt service routine written by the user for axes 1-3

myEPCIO_ENC345_ISR Function pointer of the ENC interrupt service routine written by the user for axes 4-6

myEPCIO_ENC678_ISR Function pointer of the ENC interrupt service routine written by the user for axes 7-9

myEPCIO_RIO0_ISR Function pointer of the interrupt service routine written by the user for RIO set 1

myEPCIO_RIO1_ISR Function pointer of the interrupt service routine written by the user for RIO set 2

myEPCIO_ADC_ISR Function pointer of the ADC interrupt service routine written by the user

myEPCIO_LIO_ISR Function pointer of the LIO interrupt service routine written by the user *myEPCIO_PCL_ISR* Function pointer of the PCL interrupt service routine written by the user

card_index The motion control card index, which ranges from 0 to 11, and is to be selected by the user. This index is used in the EPCIO Series Device Driver Library to identify motion control cards. Therefore, different indices must be selected for different motion control cards respectively. Due to the limited range of the index, a PC can use a maximum of only 12 EPCIO Series motion control cards at the same time.

Return Value	true	Initialization is successful.
	false	Initialization has failed.

Description Initialize an EPCIO-4000 control card and specify the customized interrupt service routines. If a routine name is given as NULL, a default service routine will be called.

Remark This function is applicable only to the EPCIO-4000 and EPCIO-4005 control cards.

II.1.2 EPCIO6000_Init()

```
BOOL EPCIO6000_Init(   DDAISR    myEPCIO_DDA_ISR,
                       ENCISR    myEPCIO_ENC012_ISR,
                       ENCISR    myEPCIO_ENC345_ISR,
                       ENCISR    myEPCIO_ENC678_ISR,
                       RIOISR    myEPCIO_RIO0_ISR,
                       RIOISR    myEPCIO_RIO1_ISR,
                       ADCISR    myEPCIO_ADC_ISR,
                       LIOISR    myEPCIO_LIO_ISR,
                       PCLISR    myEPCIO_PCL_ISR,
                       WORD      card_index)
```

Parameters

myEPCIO_DDA_ISR Function pointer of the DDA interrupt service routine written by the user

myEPCIO_ENC012_ISR Function pointer of the ENC interrupt service routine written by the user for axes 1-3

myEPCIO_ENC345_ISR Function pointer of the ENC interrupt service routine written by the user for axes 4-6 *myEPCIO_ENC678_ISR* Function pointer of the ENC interrupt service routine written by the user for axes 7-9

myEPCIO_RIO0_ISR Function pointer of the interrupt service routine written by the user for RIO set 1

myEPCIO_RIO1_ISR Function pointer of the interrupt service routine written by the user for RIO set 2

myEPCIO_ADC_ISR Function pointer of the ADC interrupt service routine written by the user

myEPCIO_LIO_ISR Function pointer of the LIO interrupt service routine written by the user

myEPCIO_PCL_ISR Function pointer of the PCL interrupt service routine written by the user

card_index The motion control card index, which ranges from 0 to 11, and is to be selected by the user. This index is used in the EPCIO Series Device Driver Library to identify motion control cards. Therefore, different indices must be selected for different motion control cards respectively. Due to the limited range of the index, a PC can use a maximum of only 12 EPCIO Series motion control cards at the same time.

Return Value true Initialization is successful.
false Initialization has failed.

Description Initialize an EPCIO-6000 control card and specify the customized interrupt service routines. If a routine name is given as NULL, a default service routine will be called.

Remark This function is applicable only to the EPCIO-6000 and EPCIO-6005 control cards.

II.1.3 EPCIO_SetISRFunction()

```
void   EPCIO_SetISRFunction(  RIOISR    myEPCIO_RIO0_ISR,
                              RIOISR    myEPCIO_RIO1_ISR,
                              ADCISR    myEPCIO_ADC_ISR,
                              LIOISR    myEPCIO_LIO_ISR,
                              PCLISR    myEPCIO_PCL_ISR,
                              WORD      card_index)
```

Parameters

myEPCIO_RIO0_ISR Function pointer of the interrupt service routine written by the user for RIO set 1

myEPCIO_RIO1_ISR Function pointer of the interrupt service routine written by the user for RIO set 2

myEPCIO_ADC_ISR Function pointer of the ADC interrupt service routine written by the user

myEPCIO_LIO_ISR Function pointer of the LIO interrupt service routine written by the user

myEPCIO_PCL_ISR Function pointer of the PCL interrupt service routine written by the user

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value None

Description Set interrupt service routines written by the user. This function, if used, must be used before EPCIO4000_Init(), or EPCIO6000_Init() is called.

Remark This function is applicable to all the EPCIO Series control cards.

II.1.4 EPCIO_Close()

BOOL EPCIO_Close(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value	true	Closing EPCIO modules is successful.
	false	Closing EPCIO modules has failed.

Description Disable EPCIO modules. This function disables all the functions of the specified EPCIO modules. If an interrupt function is set during initialization, the interrupt vector will be restored, too.

Remark This function is applicable to all the EPCIO Series control cards.

II.1.5 EPCIO_ResetModule()

BOOL EPCIO_ResetModule(WORD module_no, WORD card_index)

Parameters	***Module_no***	The number of the module to reset
	RESET_DDA	DDA Module
	RESET_ENC	Encoder Counter Channel 0 ~ 8
	RESET_RIO0	Remote I/O Set 0
	RESET_RIO1	Remote I/O Set 1
	RESET_ADC	ADC Module
	RESET_LIO	Local I/O Module
	RESET_PCL	PCL Module
	RESET_DAC	DAC Module

	RESET_PERI	Peripheral Module
	RESET_ALL	All Modules
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully reset.
	false	module_no is out of the range of correct value.
Description	Reset the specified EPCIO module(s). This function allows the user to reset an EPCIO module by a software setting. It is feasible to reset each EPCIO module independently or reset all the modules at once.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.1.6 EPCIO_SetIntPeriod()

BOOL EPCIO_SetIntPeriod(WORD period, WORD card_index)

Parameters	***period***	system clock number (1 ~ 255) (System Clock: 25ns)
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	Period is out of the range of correct value.
Description	Set the high or low active period (indicated by the number of system clock cycles) of an ISA/PCI Bus interrupt signal.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.1.7 EPCIO_SetIntMode()

BOOL EPCIO_SetIntMode(WORD mode, WORD card_index)

Parameters	***mode***	Interrupt mode
	INT_RISE_EDGE	Interrupt triggered by a rising edge
	INT_FALL_EDGE	Interrupt triggered by a falling edge
	INT_LEVEL_HIGH	Interrupt triggered by a high active level
	INT_LEVEL_LOW	Interrupt triggered by a low active level
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	The interrupt mode is out of the range of correct value.
Description	Set the trigger mode of an ISA/PCI Bus interrupt signal.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.2. DDA Control Interface

II.2.1 EPCIO_DDA_GetCurrentCmd()

BOOL EPCIO_DDA_GetCurrentCmd(WORD dda_ch_no, int *dda_cmd, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
dda_cmd Pulse command value to be acquired
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Data successfully acquired.
false The specified DDA channel is not in the setting range.

Description Acquire the value of the DDA pulse command which is currently executed in the specified DDA channel.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.2 EPCIO_DDA_CheckFIFOEmpty()

BOOL EPCIO_DDA_CheckFIFOEmpty(WORD dda_ch_no, WORD *flag, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)

flag To read back the pointer flag value of a FIFO
0 FIFO not empty
1 FIFO empty

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully checked.
false The specified DDA channel is not in the setting range.

Description Check whether the FIFO of the specified DDA channel is empty.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.3 EPCIO_DDA_CheckFIFOFull()

BOOL EPCIO_DDA_CheckFIFOFull(WORD dda_ch_no, WORD *flag, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)

flag To read back the pointer flag value of a FIFO
0 FIFO not full
1 FIFO full

Return Value true Successfully checked.
false The specified DDA channel is not in the setting range.

Description Check whether the FIFO of the specified DDA channel is full.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.4 EPCIO_DDA_GetStockCount()

BOOL EPCIO_DDA_GetStockCount(WORD dda_ch_no, WORD *count, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
count The number read back of the commands in stock
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Data successfully acquired.
false The specified DDA channel is not in the setting range.

Description Acquire the number of commands which are currently stored in the FIFO of the specified DDA channel but have yet to be executed and sent out.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.5 EPCIO_DDA_EnableOutABSwap()

BOOL EPCIO_DDA_EnableOutABSwap(WORD dda_ch_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false The specified DDA channel is not in the setting range.

Description Set the DDA channel output pulse format for swapping signals A and B of pulse output pins. The default setting is No Swapping.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.6 EPCIO_DDA_DisableOutABSwap()

BOOL EPCIO_DDA_DisableOutABSwap(WORD dda_ch_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified DDA channel is not in the setting range.

Description Set the DDA channel output pulse format for not swapping signals A and B of pulse output pins. The default setting is No Swapping.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.7 EPCIO_DDA_EnableOutAInverse()

BOOL EPCIO_DDA_EnableOutAInverse(WORD dda_ch_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false The specified DDA channel is not in the setting range.

Description Set the DDA channel output pulse format for inverting signal A of pulse output pin. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.8 EPCIO_DDA_DisableOutAInverse()

BOOL EPCIO_DDA_DisableOutAInverse(WORD dda_ch_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified DDA channel is not in the setting range.

Description Set the DDA channel output pulse format for not inverting signal A of pulse output pin. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.9 EPCIO_DDA_EnableOutBInverse()

BOOL EPCIO_DDA_EnableOutBInverse(WORD dda_ch_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false The specified DDA channel is not in the setting range.

Description Set the DDA channel output pulse format for inverting signal B of pulse output pin. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.10 EPCIO_DDA_DisableOutBInverse()

BOOL EPCIO_DDA_DisableOutBInverse(WORD dda_ch_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified DDA channel is not in the setting range.

Description Set the DDA channel output pulse format for not inverting signal B of pulse output pin. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.11 EPCIO_DDA_SetOutputFormat()

***BOOL EPCIO_DDA_Set_OutputFormat(WORD dda_ch_no, WORD format,
WORD card_index)***

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)

format DDA pulse output format, to be set as follows:

DDA_FMT_PD	Pulse/Direction output format (default)
DDA_FMT_CW	CW/CCW output format
DDA_FMT_AB	Phase A/Phase B output format
DDA_FMT_NO	Pulse output inhibited

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified DDA channel or output format is not in the corresponding setting range.

Description Specify the output pulse format. The pulse output format as DDA_FMT_PD, DDA_FMT_CW, or DDA_FMT_A/B.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.12 EPCIO_DDA_SetPulseWidth()

BOOL EPCIO_DDA_SetPulseWidth(WORD dda_ch_no, WORD clock_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
clock_no The width of pulses output from the DDA, ranges from 0 to 2047 system clock cycles (25ns).
An initial value of 0 means no pulse output.
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified DDA channel or pulse width is not in the corresponding setting range.

Description The width of pulses output from the DDA should be set according to servo drive requirements in order to adapt to servo drive specifications. With this function, the output pulse width can be set to the desired number (clock_no) of system clock cycles.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.13 EPCIO_DDA_EnableStockInt()

BOOL EPCIO_DDA_EnableStockInt(WORD FIFO_no, WORD card_index)

Parameters ***FIFO_no*** DDA FIFO channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false The specified DDA channel is not in the setting range.

Description Enable the DDA FIFO minimum stock interrupt function. A hardware interrupt request (IRQ) will be triggered if the number of commands left in the FIFO is equal to the preset minimum number.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.14 EPCIO_DDA_DisableStockInt()

BOOL EPCIO_DDA_DisableStockInt(WORD FIFO_no, WORD card_index)

Parameters ***FIFO_no*** DDA FIFO channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified DDA channel is not in the setting range.

Description Disable the DDA FIFO minimum stock interrupt function so that the corresponding hardware interrupt request (IRQ) will not be triggered.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.15 EPCIO_DDA_EnableCycleInt()

BOOL EPCIO_DDA_EnableCycleInt(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false Enabling has failed.

Description Enable the DDA cyclic interrupt function. The DDA will automatically trigger a hardware interrupt request (IRQ) at each fixed interval equal to the DDA time.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.16 EPCIO_DDA_DisableCycleInt()

BOOL EPCIO_DDA_DisableCycleInt(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.

false Disabling has failed.

Description Disable the DDA cyclic interrupt function so that hardware interrupt requests (IRQ) will not be triggered at each fixed interval equal to the DDA time.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.17 EPCIO_DDA_EnableOutputChannel()

BOOL EPCIO_DDA_EnableOutputChannel(WORD dda_ch_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false The specified DDA channel is not in the setting range.

Description Enable DDA control of the channel with the number dda_ch_no. This function serves to enable the DDA output function of a single axis, and yet the pulse output function will not work unless EPCIO_DDA_StartEngine() has been called at least once.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_DDA_StartEngine()

II.2.18 EPCIO_DDA_DisableOutputChannel()

BOOL EPCIO_DDA_DisableOutputChannel(WORD dda_ch_no, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.

false The specified DDA channel is not in the setting range.

Description Disable DDA control of the channel with the number dda_ch_no. Once this function is set, the specified DDA channel will stop pulse output immediately, and all the commands in the FIFO that are currently being executed but not yet completed will be removed.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.19 EPCIO_DDA_SetBitLength()

BOOL EPCIO_DDA_SetBitLength(WORD bitno, WORD card_index)

Parameters ***bitno*** The number of bits to be used in the DDA algorithm, to be set as follows:

DDA_LEN10	DDA engine bit length = 10 bits
DDA_LEN11	DDA engine bit length = 11 bits
DDA_LEN12	DDA engine bit length = 12 bits
DDA_LEN13	DDA engine bit length = 13 bits
DDA_LEN14	DDA engine bit length = 14 bits
DDA_LEN15	DDA engine bit length = 15 bits

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified bitno is not in the setting range.

Description Set the DDA engine bit length for use during DDA operation. The setting value is the maximum number of pulses (2^{bitno}) that can be output in each DDA time.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_DDA_SetClockDivider() EPCIO_DDA_SetTime()

II.2.20 EPCIO_DDA_SetClockDivider()

BOOL EPCIO_DDA_SetClockDivider(WORD divider, WORD card_index)

Parameters ***divider*** DDA clock divider value, to be set within the range of 0 to 4095. The default value, 0, means dividing by 1.

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified divider is not in the setting range.

Description Set the DDA clock for use during DDA engine operation, wherein the DDA clock frequency is System Clock (40MHz) divided by (divider + 1). This setting will affect the shortest time interval between two adjacent pulses.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_DDA_SetBitLength() EPCIO_DDA_SetTime()

II.2.21 EPCIO_DDA_SetTime()

BOOL EPCIO_DDA_SetTime(float ddatime, WORD length, WORD card_index)

Parameters ***ddatime*** DDA cycle time, whose setting range varies, depending on the user-given DDA engine bit length.

length The number of bits to be used in the DDA algorithm, to be set as follows:

DDA_LEN10 The setting range is 1≦ddatime≦105, in the unit of ms.

DDA_LEN11 The setting range is 1≦ddatime≦210, in the unit of ms.

DDA_LEN12 The setting range is 1≦ddatime≦420, in the unit of ms.

DDA_LEN13 The setting range is 1≦ddatime≦840, in the unit of ms.

DDA_LEN14 The setting range is 1≦ddatime≦1680, in the unit of ms.

DDA_LEN15 The setting range is 1≦ddatime≦3360, in the unit of ms.

If the user does not specify the length parameter, the default setting will be DDA_LEN15.

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified ddatime is not in the setting range.

Description Set the cycle time of DDA engine operation. The function converts the DDA time into the DDA clock divider value and a bit length, and writes them into hardware. Once DDA engine control is enabled, EPCIO ASIC reads one command from the FIFO at each fixed time interval in order for the DDA engine to convert the command value into pulses for output.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_DDA_SetBitLength() *EPCIO*_DDA_SetClockDivider()

II.2.22 EPCIO_DDA_SetMinStockNo()

BOOL EPCIO_DDA_SetMinStockNo(WORD stock_no, WORD card_index)

Parameters ***stock_no*** The minimum number of commands stored in a FIFO (1 ~ 63)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified stock_no is not in the setting range.

Description Set the interrupt-triggering minimum number of commands stored in DDA FIFOs. The setting of all DDA channels are the same. After the setting is completed, and the minimum stock interrupt function is enabled with EPCIO_EnableFIFOStockInt(), an interrupt will be generated when the number of commands left in the FIFO is equal to stock_no.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_EnableFIFOStockInt()

II.2.23 EPCIO_DDA_SendPulse()

BOOL EPCIO_DDA_SendPulse(WORD dda_ch_no, int pulse, WORD card_index)

Parameters ***dda_ch_no*** The specified DDA channel number (0 ~ 5)
pulse pulse command value
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Writing is successful.
false The specified DDA channel is not in the setting range.

Description Write a pulse command into the specified DDA FIFO. The maximum value that each pulse command can send out is related to the DDA bit length setting and the pulse width. When the DDA bit length is set to 10 bits, the maximum pulse command that can be send in each DDA time is $\pm 2^{10}$. By the same token, when the DDA bit length is set to N bits, the maximum pulse command that can be output in each DDA time is $\pm 2^{N}$. However, when the pulse width is greater than the time interval between two adjacent pulses (DDA time/2^{N}), each pulse will overlap with the previous one, and it is important to avoid such overlapping conditions.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_DDA_SetBitLength()
EPCIO_DDA_SetPulseWidth()
EPCIO_DDA_SetClockDivider()

II.2.24 EPCIO_DDA_StartEngine()

BOOL EPCIO_DDA_StartEngine(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false Setting has failed.

Description Start DDA engine operation.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.25 EPCIO_DDA_StopEngine()

BOOL EPCIO_DDA_StopEngine(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false Setting has failed.

Description Stop DDA engine operation.

Remark This function is applicable to all the EPCIO Series control cards.

II.2.26 EPCIO_DDA_ShiftOutFIFOCmd()

BOOL EPCIO_DDA_ShiftOutFIFOCmd(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false Setting has failed.

Description This function removes the next to-be-executed command in a DDA FIFO. In order to use this function to remove a command from the DDA FIFO, it is required that operation of the DDA channel is stopped in advance. Only when the channel has stopped operation can a command in the FIFO be removed. Channels in operation will not be affected.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_DDA_DisableOutputChannel()

II.2.27 EPCIO_DDA_EraseFIFOCmd()

BOOL EPCIO_DDA_EraseFIFOCmd(WORD ch, WORD erase, WORD card_index)

Parameters ***ch*** The specified DDA channel number (0 ~ 5)

erase The number of unexecuted commands in a FIFO that are to be erased

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully erased.

false Erasing has failed.

Description This function immediately removes tail commands in a DDA FIFO that have been stored but have not been executed yet. Up to 64 commands can be erased at a time. The commands being executed will not be affected.

Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.

II.2.28 EPCIO_DDA_EnableEmgcStop()

BOOL EPCIO_DDA_EnableEmgcStop(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false Enabling has failed.

Description Set DDA output emergency stop. This function can immediately stop the output of commands which are in the process of being output and which is in execution. Meanwhile, computation of commands continues in the EPCIO. Once the emergency stop function is canceled, commands will be output in the next DDA period simultaneously for all axes.

Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.

See also EPCIO_DDA_DisableEmgcStop()

II.2.29 EPCIO_DDA_DisableEmgcStop()

BOOL EPCIO_DDA_DisableEmgcStop(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting successfully canceled.

false Setting cancellation has failed.

Description Cancel the setting of DDA output emergency stop.
Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.
See also EPCIO_DDA_EnableEmgcStop()

II.2.30 EPCIO_DDA_GetOutputPulse()

BOOL EPCIO_DDA_GetOutputPulse(WORD ch, WORD card_index)

Parameters ***ch*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value true Data successfully acquired.
false Data acquisition has failed.
Description The actual number of pulses output from the DDA is recorded in the EPCIO register. With this function, the pulse values in the register (i.e., the logical coordinates rather than the actual coordinates) can be acquired.
Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.
See also EPCIO_DDA_EnablePulseCounter()
EPCIO_DDA_DisablePulseCounter()
EPCIO_DDA_ClearPulseCounter()

II.2.31 EPCIO_DDA_SetPulseDivider()

BOOL EPCIO_DDA_SetPulseDivider(WORD ch, WORD divider, WORD card_index)

Parameters ***ch*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.
divider The pulse frequency divider, to be set in the range of 0 ~ 255. When the divider is set to 0, the specified number of pulses in a FIFO command is divided by 1. The results of other settings can be deduced by analogy.
Return Value true Setting is successful.
false Setting has failed.
Description By setting the divider value, the actual number of output pulses is set to the specified number of pulses in a FIFO command divided by (divider + 1). This function helps provide smooth acceleration and deceleration via command-based frequency multiplication in software and output frequency division process in hardware.
Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.
See also EPCIO_DDA_ClearPulseDivider()

II.2.32 EPCIO_DDA_ClearPulseDivider()

BOOL EPCIO_DDA_ClearPulseDivider(WORD ch, WORD card_index)

Parameters ***ch*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully cleared.
false Clearing has failed.

Description Once EPCIO_DDA_SetPulseDivider() is called, the actual number of output pulses will be the specified number of pulses in a FIFO command divided by (divider + 1). If the specified number of pulses is not exactly divisible, a remainder will be generated in the divider. This function serves to clear the remainder in the divider.

Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.

See also EPCIO_DDA_SetPulseDivider()

II.2.33 EPCIO_DDA_EnablePulseCounter()

BOOL EPCIO_DDA_EnablePulseCounter(WORD ch, WORD card_index)

Parameters ***ch*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false Enabling has failed.

Description Enable the recording function of the EPCIO internal logical coordinate recorder.

Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.

See also EPCIO_DDA_GetOutputPulse()
EPCIO_DDA_DisablePulseCounter()
EPCIO_DDA_ClearPulseCounter()

II.2.34 EPCIO_DDA_DisablePulseCounter()

BOOL EPCIO_DDA_DisablePulseCounter(WORD ch, WORD card_index)

Parameters ***ch*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false Disabling has failed.

Description Disable the recording function of the EPCIO internal logical coordinate recorder.

Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.

See also EPCIO_DDA_GetOutputPulse()
EPCIO_DDA_EnablePulseCounter()
EPCIO_DDA_ClearPulseCounter()

II.2.35 EPCIO_DDA_ClearPulseCounter()

BOOL EPCIO_DDA_ClearPulseCounter(WORD ch, WORD card_index)

Parameters ***ch*** The specified DDA channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully cleared.
false Clearing has failed.

Description Clear the recorded values in the EPCIO internal logical coordinate recorder (i.e., to return the values to zero).

Remark This function is applicable to all the EPCIO Series control cards whose EPCIO-ASIC version no. is 6988-02 or above.

See also EPCIO_DDA_GetOutputPulse()
EPCIO_DDA_EnablePulseCounter()
EPCIO_DDA_DisablePulseCounter()

II.3. Encoder Counter Interface

II.3.1 EPCIO_ENC_GetValue()

BOOL EPCIO_ENC_GetValue(WORD enc_ch_no, long *counter, WORD card_index)

Parameters	***enc_ch_no***	Encoder channel number (0 ~ 8)
	counter	The encoder counter value to be acquired
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	The specified ENC channel is not in the setting range.
Description	Acquire the current count of an encoder counter channel.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.3.2 EPCIO_ENC_GetLatchValue()

BOOL EPCIO_ENC_GetLatchValue(WORD enc_ch_no, long *latch, WORD card_index)

Parameters	***enc_ch_no***	Encoder channel number (0 ~ 8)
	latch	The latched encoder counter value to be acquired
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	The specified ENC channel is not in the setting range.
Description	Acquire the latched encoder counter value due to meeting the specified interrupt conditions.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.3.3 EPCIO_ENC_GetIndexStatus()

BOOL EPCIO_ENC_GetIndexStatus(WORD enc_ch_no, WORD *status, WORD card_index)

Parameters	***enc_ch_no***	Encoder channel number (0 ~ 8)
	status	The encoder index status value to be acquired
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	The specified ENC channel is not in the setting range.
Description	Acquire the current HIGH/LOW status of the specified encoder counter index signal.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.3.4 EPCIO_ENC_SetCompValue()

BOOL EPCIO_ENC_SetCompValue(WORD enc_ch_no, long value, WORD card_index)

Parameters	***enc_ch_no***	Encoder channel number (0 ~ 8)
	value	A value set for the encoder counter comparator
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	The specified ENC channel is not in the setting range.
Description	Set a comparison value for the encoder counter. When the accumulated count of the counter is equal to the set comparison value, and if EPCIO_ENC_EnableCompInt() is enabled, a hardware interrupt signal will be generated. This trigger signal will trigger the DAC to rapidly send out a preset voltage command.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_ENC_EnableCompInt()	EPCIO_DAC_SetTrigSource()

II.3.5 EPCIO_ENC_EnableCompInt()

BOOL EPCIO_ENC_EnableCompInt(WORD enc_ch_no, WORD card_index)

Parameters	***enc_ch_no***	Encoder channel number (0 ~ 8)
Return Value	true	Successfully enabled.
	false	The specified ENC channel is not in the setting range.
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Remark	This function is applicable to all the EPCIO Series control cards.	
Description	Enable the interrupt of comparing the encoder count with the comparison value.	
See also	EPCIO_ENC_SetCompValue()	

II.3.6 EPCIO_ENC_DisableCompInt()

BOOL EPCIO_ENC_DisableCompInt(WORD enc_ch_no, WORD card_index)

Parameters	***enc_ch_no***	Encoder channel number (0 ~ 8)
Return Value	true	Successfully disabled.
	false	The specified ENC channel is not in the setting range.
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Description	Disable the interrupt of comparing the encoder count with the comparison value.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.3.7 EPCIO_ENC_SetInputRate()

BOOL EPCIO_ENC_SetInputRate(WORD enc_ch_no, WORD rate, WORD card_index)

Parameters	***enc_ch_no***	Encoder channel number (0 ~ 8)
	rate	Encoder multiplier
	ENC_RATE_X0	Multiplier to be 0 (inhibit)
	ENC_RATE_X1	Multiplier to be 1
	ENC_RATE_X2	Multiplier to be 2
	ENC_RATE_X4	Multiplier to be 4
	card_index	The index of the motion control card to be

controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified ENC channel or multiplier rate is not in the corresponding setting range.

Description Set the signal-decoding multiplier rate for the specified encoder counter channel. The encoder's decoding multiplier is valid only when the encoder input format is A/B phase. To use this function, the input format must be set to A/B phase with EPCIO_ENC_SetInputType().

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_ENC_SetInputType()

II.3.8 EPCIO_ENC_SetInputType()

BOOL EPCIO_ENC_SetInputType(WORD enc_ch_no, WORD type, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)

type Encoder input signal format

ENC_TYPE_AB	Input type is quadratic or A/B phase (default).
ENC_TYPE_CW	Input type is CW/CCW.
ENC_TYPE_PD	Input type is Pulse/Direction.
ENC_TYPE_NO	Input is inhibited.

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified ENC channel or type is not in the corresponding setting range.

Description Set the input signal format for the specified encoder counter channel. This function must match the hardware's actual signal settings. When the input signal is a motor encoder feedback signal, please refer to motor or servo drive settings; when a common MPG handwheel is used, set the input signal format to A/B phase (Default: A/B phase input).

Remark This function is applicable to all the EPCIO Series control cards.

II.3.9 EPCIO_ENC_EnableInAInverse()

BOOL EPCIO_ENC_EnableInAInverse(WORD enc_ch_no, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false The specified ENC channel is not in the setting range.

Description Invert the input signal inA of the specified encoder counter channel. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.3.10 EPCIO_ENC_DisableInAInverse()

BOOL EPCIO_ENC_DisableInAInverse(WORD enc_ch_no, WORD card_index)

Parameters *enc_ch_no* Encoder channel number (0 ~ 8)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified ENC channel is not in the setting range.

Description To not invert the input signal inA of the specified encoder counter channel. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.3.11 EPCIO_ENC_EnableInBInverse()

BOOL EPCIO_ENC_EnableInBInverse(WORD enc_ch_no, WORD card_index)

Parameters *enc_ch_no* Encoder channel number (0 ~ 8)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false Encoder counter channel selection error: not in the setting range.

Description Invert the input signal inB of the specified encoder counter channel. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.3.12 EPCIO_ENC_DisableInBInverse()

BOOL EPCIO__ENC_DisableInBInverse(WORD enc_ch_no, WORD card_index)

Parameters *enc_ch_no* Encoder channel number (0 ~ 8)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified ENC channel is not in the setting range.

Description To not invert the input signal inB of specified encoder counter channel. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.3.13 EPCIO_ENC_EnableInCInverse()

BOOL EPCIO_ENC_EnableInCInverse(WORD enc_ch_no, WORD card_index)

Parameters *enc_ch_no* Encoder channel number (0 ~ 8)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false The specified ENC channel is not in the setting range.

Description Invert the input signal inC of the specified encoder counter channel. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.3.14 EPCIO_ENC_DisableInCInverse()

BOOL EPCIO_ENC_DisableInCInverse(WORD enc_ch_no, WORD card_index)

Parameters *enc_ch_no* Encoder channel number (0 ~ 8)
card_index The index of the motion control card to be controlled.

The index ranges from 0 to 11.

Return Value true Successfully disabled.

false The specified ENC channel is not in the setting range.

Description To not invert the input signal inC of the specified encoder counter channel. The default setting is No Inverting.

Remark This function is applicable to all the EPCIO Series control cards.

II.3.15 EPCIO_ENC_EnableInABSwap()

BOOL EPCIO_ENC_EnableInABSwap(WORD enc_ch_no, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false The specified ENC channel is not in the setting range.

Description Swap the input signals inA and inB of the specified encoder channel (i.e., to perform signal swapping) before the signals enter the counter. The default setting is No Signal Swapping.

Remark This function is applicable to all the EPCIO Series control cards.

II.3.16 EPCIO_ENC_DisableInABSwap()

BOOL EPCIO_ENC_DisableInABSwap(WORD enc_ch_no, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.

false The specified ENC channel is not in the setting range.

Description To not swap the input signals inA and inB of the specified encoder counter channel (i.e., to not perform signal swapping) before the signals enter the counter. The default setting is No Signal Swapping.

Remark This function is applicable to all the EPCIO Series control cards.

II.3.17 EPCIO_ENC_SetTrigMode()

BOOL EPCIO_ENC_SetTrigMode(WORD enc_ch_no, WORD mode, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)

mode Encoder counter latch trigger mode

ENC_TRIG_FIRST When the trigger condition is met for the first time, the count is immediately latched and will no longer change.

ENC_TRIG_LAST When the trigger condition is met, the count is immediately latched, and the latched count is updated whenever the trigger condition is met again.

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified ENC channel is not in the setting range.

Description Set the encoder counter latch trigger mode for the specified encoder counter channel. This function must be used in combination with

EPCIO_ENC_SetTrigSource() for setting the sources of trigger signals so that, when a trigger signal occurs, the count of the specified encoder channel will be latched according to the specified trigger mode.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_ENC_ SetTrigSource()

II.3.18 EPCIO_ENC_SetTrigSource()

BOOL EPCIO_ENC_SetTrigSource(WORD enc_ch_no, WORD source, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)

source Encoder latch condition. There are a total of 15 trigger sources which can be selected for the counter latch condition. The setting can be the union of a number of sources.

ENC_TRIG_NO	No trigger source selected
ENC_TRIG_INDEX0	Index signal of encoder channel 0
ENC_TRIG_INDEX1	Index signal of encoder channel 1
ENC_TRIG_INDEX2	Index signal of encoder channel 2
ENC_TRIG_INDEX3	Index signal of encoder channel 3
ENC_TRIG_INDEX4	Index signal of encoder channel 4
ENC_TRIG_INDEX5	Index signal of encoder channel 5
ENC_TRIG_INDEX6	Index signal of encoder channel 6
ENC_TRIG_INDEX7	Index signal of encoder channel 7
ENC_TRIG_INDEX8	Index signal of encoder channel 8
ENC_TRIG_LIO0	Local DI 0 INT
ENC_TRIG_LIO1	Local DI 1 INT
ENC_TRIG_RDI0	Remote I/O Set 0 Slave 0 DI 0 INT
ENC_TRIG_RDI1	Remote I/O Set 0 Salve 0 DI 1 INT
ENC_TRIG_ADC0	ADC channel 0 comparator INT
ENC_TRIG_ADC1	ADC channel 1 comparator INT

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified ENC channel is not in the setting range.

Description Set the trigger sources for the specified encoder counter channel. The specified trigger sources will trigger the latch of the encoder counter's value. This function must be used in combination with EPCIO_ENC_SetTrigMode().

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_ENC_SetTrigMode()

II.3.19 EPCIO_ENC_EnableIndexInt()

BOOL EPCIO_ENC_EnableIndexInt(WORD enc_ch_no, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false The specified ENC channel is not in the setting range.
Description Enable the interrupt function of encoder index.
Remark This function is applicable to all the EPCIO Series control cards.

II.3.20 EPCIO_ENC_DisableIndexInt()

BOOL EPCIO_ENC_DisableIndexInt(WORD enc_ch_no, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value true Successfully disabled.
false The specified ENC channel is not in the setting range.
Description Disable the interrupt function of encoder index.
Remark This function is applicable to all the EPCIO Series control cards.

II.3.21 EPCIO_ENC_SetFilterClock()

BOOL EPCIO_ENC_SetFilterClock(WORD divider, WORD card_index)

Parameters ***divider*** Sampling clock divider value of encoder filter (0 ~ 255)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value true Setting is successful.
false The specified divider value is not in the setting range.
Description Enable the encoder filtering and sampling function and set the sampling rate. The sampling rate is System Clock (40MHz) divided by 2 × (divider + 1). Once the sampling rate is set, the input signal must be identical (High or Low) in three consecutive samples to be considered as a valid input. The default divider value is 0.
Remark This function is applicable to all the EPCIO Series control cards.

II.3.22 EPCIO_ENC_ClearCounter()

BOOL EPCIO_ENC_ClearCounter(WORD enc_ch_no, WORD card_index)

Parameters ***enc_ch_no*** Encoder channel number (0 ~ 8)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value true Successfully cleared.
false The specified ENC channel is not in the setting range.
Description Clear the counter value of the specified encoder channel.
Remark This function is applicable to all the EPCIO Series control cards.

II.3.23 EPCIO_ENC_StartInput()

BOOL EPCIO_ENC_StartInput(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value true Successfully started.
false Starting has failed.
Description Enable the encoder counting function.
Remark This function is applicable to all the EPCIO Series control cards.

II.3.24 EPCIO_ENC_StopInput()

BOOL EPCIO_ENC_StopInput(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully stopped.
	false	Stopping has failed.
Description	Disable the encoder counting function.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.4. Remote Digital I/O

II.4.1 EPCIO_RIO_GetInputValue()

BOOL EPCIO_RIO_GetInputValue(WORD set, WORD slave, WORD port, WORD *value,
WORD card_index)

Parameters	***set***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	Slave	Slave number selection in the selected set
	RIO_SLAVE0	Remote I/O slave 0 in the selected set
	RIO_SLAVE1	Remote I/O slave 1 in the selected set
	RIO_SLAVE2	Remote I/O slave 2 in the selected set
	Port	Digital input port number selection in the selected slave
	RIO_PORT0	DI 0 ~ DI 15 in the selected slave
	RIO_PORT1	DI 16 ~ DI 31 in the selected slave
	RIO_PORT2	DI 32 ~ DI 47 in the selected slave
	RIO_PORT3	DI 48 ~ DI 63 in the selected slave
	value	Variable name of the read-back digital input data
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	The specified parameter set, slave, or port is not in the corresponding setting range.
Description	Acquire the current signal status of the digital input of the specified remote I/O port.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.4.2 EPCIO_RIO_SetOutputValue()

BOOL EPCIO_RIO_SetOutputValue(WORD set, WORD slave, WORD port , WORD value,
WORD card_index)

Parameters	***set***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	slave	Slave number selection in the selected set
	RIO_SLAVE0	Remote I/O slave 0 in the selected set
	RIO_SLAVE1	Remote I/O slave 1 in the selected set

RIO_SLAVE2 Remote I/O slave 2 in the selected set

port Digital output port number selection in the selected slave

RIO_PORT0	DI 0 ~ DI 15 in the selected slave
RIO_PORT1	DI 16 ~ DI 31 in the selected slave
RIO_PORT2	DI 32 ~ DI 47 in the selected slave
RIO_PORT3	DI 48 ~ DI 63 in the selected slave

value 16-bit output data

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified parameter set, slave, or port is not in the corresponding setting range.

Description Set the status value of the 16-bit digital output signal of the specified port of the specified slave in the specified set.

Remark This function is applicable to all the EPCIO Series control cards.

II.4.3 EPCIO_RIO_GetTransStatus()

BOOL EPCIO_RIO_GetTransStatus(WORD set_no, WORD *status, WORD card_index)

Parameters ***set_no*** Remote I/O set number selection

status RIO master/slave data receiving status

1	Specified RIO set is working.
0	Specified RIO set has stopped working.

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Data successfully acquired.

false The specified parameter set is not in the setting range.

Description Acquire the current status of RIO transmission. If transmission has stopped, EPCIO_RIO_GetMasterStatus() and EPCIO_RIO_Get SlaveStatus() will be called to determine whether the error occurs at the master or slave.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_RIO_GetMasterStatus() EPCIO_RIO_GetSlaveStatus()

II.4.4 EPCIO_RIO_GetMasterStatus()

BOOL EPCIO_RIO_GetMasterStatus(WORD set_no, WORD *status, WORD card_index)

Parameters ***set_no*** Remote I/O set number selection

RIO_SET0	Remote I/O Set 0
RIO_SET1	Remote I/O Set 1

status RIO master data receiving status

0	RIO master data receiving OK
1	RIO master data receiving FAIL

	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	The specified parameter set is not in the setting range.
Description	Acquire the current status of RIO master data transmission to slave. If a transmission error occurs, EPCIO_RIO_GetSlaveFail() will be called to locate the slave where the error takes place.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_RIO_GetSlaveStatus() EPCIO_RIO_GetSlaveFail()	

II.4.5 EPCIO_RIO_GetSlaveStatus()

BOOL EPCIO_RIO_GetSlaveStatus(WORD set_no, WORD *status, WORD card_index)

Parameters	***set_no***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	status	RIO slave data receiving status
	0	RIO slave data receiving OK
	1	RIO slave data receiving FAIL
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	The specified parameter set is not in the setting range.
Description	Acquire the current status of RIO slave reception of master data. If a reception error occurs, EPCIO_RIO_GetSlaveFail() will be called to locate the slave where the error takes place.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_RIO_GetMasterStatus() EPCIO_RIO_GetSlaveFail()	

II.4.6 EPCIO_RIO_SetClockDivider()

BOOL EPCIO_RIO_SetClockDivider(WORD set_no, WORD divider, WORD card_index)

Parameters	***set_no***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	divider	Remote I/O clock divider (0 ~ 255)
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	The specified parameter set_no is not in the setting range.
Description	Set the clock of remote I/O data transmission. The transmission frequency is System Clock (40MHz) divided by 2 × (divider + 1). The default divider is 0.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.4.7 EPCIO_RIO_SetIntType()

BOOL EPCIO_RIO_SetIntType(WORD set, WORD slave, WORD input, WORD type, WORD card_index)

Parameters	***set***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	slave	Remote slave number selection
	RIO_SLAVE0	Remote I/O slave 0 in the selected set
	RIO_SLAVE1	Remote I/O slave 1 in the selected set
	RIO_SLAVE2	Remote I/O slave 2 in the selected set
	input	Slave DI number
	RIO_DI0	Remote I/O input 0 in the selected slave
	RIO_DI1	Remote I/O input 1 in the selected slave
	RIO_DI2	Remote I/O input 2 in the selected slave
	RIO_DI3	Remote I/O input 3 in the selected slave
	type	interrupt trigger type
	RIO_INT_RISE	Rising edge trigger
	RIO_INT_FALL	Falling edge trigger
	RIO_INT_LEVEL	Level change trigger
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified parameter set, slave, input, or type is not in the corresponding setting range.

Description Set the remote I/O digital input interrupt trigger mode to “rising edge trigger”, “falling edge trigger”, or “level change trigger”. Once this function is set, EPCIO_RIO_EnableInputInt() must be called to enable the interrupt function.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_RIO_EnableInputInt()

II.4.8 EPCIO_RIO_EnableInputInt()

BOOL EPCIO_RIO_EnableInputInt(WORD set, WORD slave, WORD input, WORD card_index)

Parameters	***set***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	slave	Remote slave number selection in the selected set
	RIO_SLAVE0	Remote I/O slave 0 in the selected set
	RIO_SLAVE1	Remote I/O slave 1 in the selected set
	RIO_SLAVE2	Remote I/O slave 2 in the selected set

	input	Slave DI number
	RIO_DI0	Remote I/O input 0 in the selected slave
	RIO_DI1	Remote I/O input 1 in the selected slave
	RIO_DI2	Remote I/O input 2 in the selected slave
	RIO_DI3	Remote I/O input 3 in the selected slave

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false The specified parameter set, slave, or input is not in the corresponding setting range.

Description Each remote I/O set has three slaves, and the first four digital inputs (DI0, DI1, DI2, DI3) of each slave can trigger an interrupt. This function is used to enable the interrupt function of DI0 ~ DI3.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_RIO_SetIntType()

II.4.9 EPCIO_RIO_DisableInputInt()

BOOL EPCIO_RIO_DisableInputInt(WORD set, WORD slave, WORD input, WORD card_index)

Parameters	***set***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	slave	Remote slave number selection
	RIO_SLAVE0	Remote I/O slave 0 in the selected set
	RIO_SLAVE1	Remote I/O slave 1 in the selected set
	RIO_SLAVE2	Remote I/O slave 2 in the selected set
	input	Slave DI number
	RIO_DI0	Remote I/O input 0 in the selected slave
	RIO_DI1	Remote I/O input 1 in the selected slave
	RIO_DI2	Remote I/O input 2 in the selected slave
	RIO_DI3	Remote I/O input 3 in the selected slave

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.

false The specified parameter set, slave, or input is not in the corresponding setting range.

Description Each remote I/O set has three slaves, and the first four digital inputs (DI0, DI1, DI2, DI3) of each slave can trigger an interrupt. This function is used to disable the interrupt function of DI0 ~ DI3.

Remark This function is applicable to all the EPCIO Series control cards.

II.4.10 EPCIO_RIO_SetTransError()

BOOL EPCIO_RIO_SetTransError(WORD time, WORD card_index)

Parameters ***time*** Maximum number of times of retransmission when an

error occurs (0 ~ 15)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified parameter time is not in the setting range.

Description Set the maximum number of times for which retransmission can be carried out when an remote I/O transmission error occurs.

Remark This function is applicable to all the EPCIO Series control cards.

II.4.11 EPCIO_RIO_EnableSetControl()

BOOL EPCIO_RIO_EnableSetControl(WORD set_no, WORD card_index)

Parameters ***set_no*** Remote I/O set number selection
RIO_SET0 Remote I/O Set 0
RIO_SET1 Remote I/O Set 1

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false The specified parameter set_no is not in the setting range.

Description Enable control of the specified remote I/O set. Each slave in the set can be enabled by calling EPCIO_RIO_EnableSlaveControl.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_RIO_EnableSlaveControl()
EPCIO_RIO_DisbaleSlaveControl()

II.4.12 EPCIO_RIO_DisableSetControl()

BOOL EPCIO_RIO_DisableSetControl(WORD set_no, WORD card_index)

Parameters ***set_no*** Remote I/O set number selection
RIO_SET0 Remote I/O Set 0
RIO_SET1 Remote I/O Set 1

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified parameter set_no is not in the setting range.

Description Disable control of the specified remote I/O set. All the slaves in the set will be disabled, too.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_RIO_EnableSlaveControl(),
EPCIO_RIO_DisableSlaveControl()

II.4.13 EPCIO_RIO_EnableSlaveControl()

BOOL EPCIO_RIO_EnableSlaveControl(WORD set, WORD slave, WORD card_index)

Parameters ***set*** Remote I/O set number selection
RIO_SET0 Remote I/O Set 0

	RIO_SET1	Remote I/O Set 1
	slave	Remote I/O slave selection
	RIO_SLAVE0	Remote I/O slave 0 in the selected set
	RIO_SLAVE1	Remote I/O slave 1 in the selected set
	RIO_SLAVE2	Remote I/O slave 2 in the selected set
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	The specified parameter set or slave is not in the corresponding setting range.
Description	Enable the specified remote I/O slave. Once the slave is enabled, EPCIO_RIO_EnableSetControl() is required to enable the set to which the slave belongs, allowing the I/O module to begin transmitting and receiving.	
Remark	This function is applicable to all the EPCIO Series control cards.	

See also	EPCIO_RIO_EnableSetControl()
	EPCIO_RIO__DisabaleSetControl()	

II.4.14 EPCIO_RIO_DisableSlaveControl()

BOOL EPCIO_RIO_DisableSlaveControl(WORD set, WORD slave, WORD card_index)

Parameters	***set***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	slave	Slave number selection in the selected RIO set
	RIO_SLAVE0	Remote I/O slave 0 in the selected set
	RIO_SLAVE1	Remote I/O slave 1 in the selected set
	RIO_SLAVE2	Remote I/O slave 2 in the selected set
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully disabled.
	false	The specified parameter set or slave is not in the corresponding setting range.
Description	Disable the specified remote I/O slave. This function can disable a single specified slave.	
Remark	This function is applicable to all the EPCIO Series control cards.	

See also	EPCIO_RIO_EnableSetControl()
	EPCIO_RIO_DisbaleSetControl()	

II.4.15 EPCIO_RIO_EnableTransInt()

BOOL EPCIO_RIO_EnableTransInt(WORD set, WORD card_index)

Parameters	***set***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1

	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	The specified parameter set is not in the setting range.
Description	Enable the remote I/O "transmission error" interrupt function.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.4.16 EPCIO_RIO_DisableTransInt()

BOOL EPCIO_RIO_DisableTransInt(WORD set_no, WORD card_index)

Parameters	***set_no***	Remote I/O set number selection
	RIO_SET0	Remote I/O Set 0
	RIO_SET1	Remote I/O Set 1
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully disabled.
	false	The specified parameter set is not in the setting range.
Description	Disable the remote I/O "transmission error" interrupt function.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.5. ADC I/O Control

II.5.1 EPCIO_ADC_GetWorkStatus()

BOOL EPCIO_ADC_GetWorkStatus(WORD *status, WORD card_index)

Parameters	***status***	ADC work status
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Status successfully acquired.
	false	Status acquisition has failed.
Description	Acquire current ADC work status.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.	

II.5.2 EPCIO_ADC_GetInput()

BOOL EPCIO_ADC_GetInput(WORD channel, float *value, WORD card_index)

Parameters	***channel***	ADC channel selection (0 ~ 7)
	value	DC voltage input value of the specified ADC channel
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	The specified parameter channel is not in the setting range.
Description	Acquire the DC voltage input of the specified ADC channel. If the ADC is specified as "unipolar," EPCIO-4000 and EPCIO-6000 rms	

voltage input is 0 to 10 V. If the ADC is set to “bipolar,” EPCIO-4000 and EPCIO-6000 rms voltage input is 0 to ±5 V.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

II.5.3 EPCIO_ADC_SetCompMask()

BOOL EPCIO_ADC_SetCompMask(WORD mask, WORD card_index)

Parameters ***mask*** The number of bits masked by ADC comparator

ADC_MASK_NO	ADC comparator masks no bit.
ADC_MASK_BIT1	ADC comparator masks one bit.
ADC_MASK_BIT2	ADC comparator masks two bits.
ADC_MASK_BIT3	ADC comparator masks three bits.

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified parameter mask is not in the setting range.

Description Set the minimum number of bits that will be masked from comparison when the input voltage value is compared with the preset comparison value in the ADC comparison mode. This function reduces the sensitivity of the comparator to prevent interruptions due to input voltage vibrations. Once this function is set, EPCIO_ADC_SetCompType() and EPCIO_ADC_EnableCompInt() must be called in order to generate an ADC interrupt signal when the trigger condition, involving a specified type of comparison between the ADC channel input voltage and a comparison value, is met.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

See also EPCIO_ADC_SetCompType() EPCIO_ADC_EnableCompInt() EPCIO_ADC_SetConvMode()

II.5.4 EPCIO_ADC_SetCompValue()

BOOL EPCIO_ADC_SetCompValue(WORD channel, float value, WORD card_index)

Parameters ***channel*** ADC channel selection (0 ~ 7)
value ADC channel voltage comparison value (0 ~ 10 V)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified parameter channel is not in the setting range.

Description Set the ADC channel input voltage comparison value for use in the bipolar mode. This function does not support voltage comparison in the unipolar mode. Once this function is set, EPCIO_ADC_SetCompType() and EPCIO_ADC_EnableCompInt() must be called in order to generate an ADC interrupt signal when the trigger condition, involving a specified type of comparison between the ADC channel input voltage and the comparison value, is met. The trigger signal can trigger the DAC module to output a preset voltage. The first two ADC trigger signals can also be used to trigger the ENC counter latch function.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

See also EPCIO_ADC_SetCompType(), EPCIO_ADC_EnableCompInt(), EPCIO_ADC_Setconv_mode(), EPCIO_DAC_SetTrigSource(), EPCIO_ENC_SetTrigSource().

II.5.5 EPCIO_ADC_SetCompType()

BOOL EPCIO_ADC_SetCompType(WORD channel, WORD type, WORD card_index)

Parameters ***channel*** ADC channel selection (0 ~ 7)

type ADC channel comparison type

ADC_COMP_RISE The ADC input voltage passes the comparison value while increasing.

ADC_COMP_FALL The ADC input voltage passes the comparison value while decreasing.

ADC_COMP_LEVEL The ADC input voltage passes the comparison value while being changed.

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified parameter channel is not in the setting range.

Description Set the type of ADC channel voltage comparison. After calling this function, EPCIO_ADC_EnableCompInt() must also be called so that satisfaction of the comparison condition triggers a hardware interrupt signal. The trigger signal can trigger the DAC module to output a preset voltage. The first two ADC trigger signals can also be used to trigger the ENC counter latch function.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

See also EPCIO_ADC_SetCompValue()
EPCIO_ADC_EnableCompInt()
EPCIO_DAC_SetTrigSource() EPCIO_ENC_SetTrigSource()

II.5.6 EPCIO_ADC_EnableCompInt()

BOOL EPCIO_ADC_EnableCompInt(WORD channel, WORD card_index)

Parameters ***channel*** ADC channel selection (0 ~ 7)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false The specified parameter channel is not in the setting range.

Description Enable the comparison-triggered interrupt function of the ADC. The trigger signal can trigger the DAC module to output a preset voltage.

The first two ADC trigger signals can also be used to trigger the ENC counter latch function.

Remark — This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

See also — EPCIO_ADC_SetCompValue() EPCIO_ADC_EnableCompType()
EPCIO_DAC_SetTrigSource() EPCIO_ENC_SetTrigSource()

II.5.7 EPCIO_ADC_DisableCompInt()

BOOL EPCIO_ADC_DisableCompInt(WORD channel, WORD card_index)

Parameters — ***channel*** ADC channel selection (0 ~ 7)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value — true Successfully disabled.
false The specified parameter channel is not in the setting range.

Description — Disable the comparison-triggered interrupt function of the ADC.

Remark — This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

II.5.8 EPCIO_ADC_EnableTagInt()

BOOL EPCIO_ADC_EnableTagInt(WORD card_index)

Parameters — ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value — true Successfully enabled.
false Enabling has failed.

Description — Enable the interrupt-triggering function of a tag channel.

Remark — This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

II.5.9 EPCIO_ADC_DisableTagInt()

BOOL EPCIO_ADC_DisableTagInt(WORD card_index)

Parameters — ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value — true Successfully disabled.
false Disabling has failed.

Description — Disable the interrupt-triggering function of a tag channel.

Remark — This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

II.5.10 EPCIO_ADC_EnableConvInt()

BOOL EPCIO_ADC_EnableConvInt(WORD card_index)

Parameters — ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value — true Successfully enabled.
false Enabling has failed.

Description — Enable the function by which an interrupt is triggered upon completion of voltage conversion in any channel.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

II.5.11 EPCIO_ADC_DisableConvInt()

BOOL EPCIO_ADC_DisableConvInt(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false Disabling has failed.

Description Disable the function by which an interrupt is triggered upon completion of voltage conversion in any channel.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

II.5.12 EPCIO_ADC_SetClockDivider()

BOOL EPCIO_ADC_SetClockDivider(WORD divider, WORD card_index)

Parameters ***divider*** ADC serial interface clock divider (0 ~ 255)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified parameter divider is not in the setting range.

Description Set the ADC serial interface clock. The ADC clock is System Clock (40MHz) divided by 4 × (divider + 1). The default divider is 0.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

II.5.13 EPCIO_ADC_SetConvType()

BOOL EPCIO_ADC_SetConvType(WORD channel, WORD type, WORD card_index)

Parameters ***channel*** ADC channel number (0 ~ 7)
type Conversion mode setting
ADC_TYPE_BIP bipolar converter type
ADC_TYPE_UNI unipolar converter type

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified parameter channel or type is not in the corresponding setting range.

Description Set the ADC channel voltage conversion mode as bipolar or unipolar.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.

II.5.14 EPCIO_ADC_EnableConvChannel()

BOOL EPCIO_ADC_EnableConvChannel(WORD channel, WORD card_index)

Parameters ***channel*** ADC channel number (0 ~ 7)

	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	The specified parameter channel is not in the setting range.
Description	Enable the voltage input function of the specified ADC channel. The conversion channel set in this function must use the Free Run mode. EPCIO_ADC_StartConv() must be called after the channel is set, in order to initiate analog-to-digital conversion.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.	
See also	EPCIO_ADC_StartConv() EPCIO_ADC_SetConvMode()	

II.5.15 EPCIO_ADC_DisableConvChannel()

BOOL EPCIO_ADC_DisableConvChannel(WORD channel, WORD card_index)

Parameters	***channel***	ADC channel number (0 ~ 7)
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully disabled.
	false	The specified parameter channel is not in the setting range.
Description	Disable the voltage input function of the specified ADC channel.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.	

II.5.16 EPCIO_ADC_SetConvMode()

BOOL EPCIO_ADC_SetConvMode(WORD mode, WORD card_index)

Parameters	***mode***	ADC conversion mode selection
	ADC_MODE_SINGLE	ADC single conversion
	ADC_MODE_FREE	ADC free running conversion
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	The specified parameter mode is not in the setting range.
Description	Set the ADC voltage conversion mode to single or free running mode.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.	
See also	EPCIO_ADC_SetSingleChannel(), EPCIO_ADC_EnableConvChannel()	

II.5.17 EPCIO_ADC_SetTagChannel()

BOOL EPCIO_ADC_SetTagChannel(WORD channel, WORD card_index)

Parameters	***channel***	ADC channel number (0 ~ 7)
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.

	false	The specified parameter channel is not in the setting range.
Description	Set an ADC channel as “tag channel”. Used in combination with EPCIO_ADC_EnableTagInt(), an interrupt signal will be generated when the “tag channel” voltage completes conversion.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.	
See also	EPCIO_ADC_EnableTagInt()	

II.5.18 EPCIO_ADC_SetSingleChannel()

BOOL EPCIO_ADC_SetSingleChannel(WORD channel, WORD card_index)

Parameters	***channel***	ADC channel number (0 ~ 7)
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	The specified parameter channel is not in the setting range.
Description	Set a specified channel to be converted. When this function is used in combination with EPCIO_ADC_SetConvMode() for setting the conversion mode to single mode, calling EPCIO_ADC_StartConv() will trigger the selected channel to directly convert input voltage to digital value once. Conversion will not occur again once it is finished; the user must call EPCIO_ADC_StartConv() again for another conversion. EPCIO_ADC_GetWorkStatus() can be used to check the conversion progress.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.	

See also	EPCIO_ADC_StartConv() EPCIO_ADC_SetConvMode()
	EPCIO_ADC_GetWorkStatus()	

II.5.19 EPCIO_ADC_StartConv()

BOOL EPCIO_ADC_StartConv(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully started.
	false	Starting has failed.
Description	Start analog voltage conversion in the conversion-enabled ADC channel. This function must be used in combination with EPCIO_ADC_EnableConvChannel().	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.	
See also	EPCIO_ADC_EnableConvChannel()	

II.5.20 EPCIO_ADC_StopConv()

BOOL EPCIO_ADC_StopConv(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully stopped.

	false	Stopping has failed.
Description	Stop analog voltage conversion in all ADC channels.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards, which have an ADC element.	

II.6. Local I/O Control

II.6.1 EPCIO_LIO_GetLDIInput()

BOOL EPCIO_LIO_GetLDIInput(DWORD *input, WORD card_index)

Parameters	***input***	Local I/O input status value
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	Data acquisition has failed.
Description	Acquire the digital input signal values of local LDI0 ~ LDI27. There are 28 local I/O connections, which can be programmed for input or output. Bit0 ~ bit27 correspond to LDI0 ~ LDI27 respectively while bit28 ~ bit31 are not input signal values. When the connections are programmed for output, the values read back represent their output statuses.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.6.2 EPCIO_LIO_SetLDOOutput()

BOOL EPCIO_LIO_SetLDOOutput(DWORD value, WORD card_index)

Parameters	***value***	Signal statuses of LIO digital outputs DO0 ~ DO27
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	Setting has failed.
Description	Set a digital output signal status of LIO outputs LDO0 ~ LDO27. The LIO output function must be programmed before this function is used.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_LIO_EnableLDOOutput()	

II.6.3 EPCIO_LIO_EnableLDOOutput()

BOOL EPCIO_LIO_EnableLDOOutput(WORD port, WORD card_index)

Parameters	***port***	The local I/O digital output port to be enabled (each port consists of 4 I/O connections)
	LIO_OUT_EN0	Port 0 is LDO0 ~ LDO3
	LIO_OUT_EN1	Port 1 is LDO4 ~ LDO7
	LIO_OUT_EN2	Port 2 is LDO8 ~ LDO11
	LIO_OUT_EN3	Port 3 is LDO12 ~ LDO15
	LIO_OUT_EN4	Port 4 is LDO16 ~ LDO19
	LIO_OUT_EN5	Port 5 is LDO20 ~ LDO23
	LIO_OUT_EN6	Port 6 is LDO24 ~ LDO27
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.

	false The specified parameter port is not in the setting range.
Description	Each local I/O digital output port includes 4 I/O connections. The 28 inputs/outputs are divided into Port 0 ~ Port 6. This function can be used to enable the output function of the specified port. The default output statuses of all the ports are Disable.
Remark	This function is applicable to all the EPCIO Series control cards.

II.6.4 EPCIO_LIO_DisableLDOOutput()

BOOL EPCIO_LIO_DisableLDOOutput(WORD port, WORD card_index)

Parameters

port The local I/O digital output port to be disabled (each port consists of 4 I/O connections)

LIO_OUT_EN0	Port 0 is LDO0 ~ LDO3
LIO_OUT_EN1	Port 1 is LDO4 ~ LDO7
LIO_OUT_EN2	Port 2 is LDO8 ~ LDO11
LIO_OUT_EN3	Port 3 is LDO12 ~ LDO15
LIO_OUT_EN4	Port 4 is LDO16 ~ LDO19
LIO_OUT_EN5	Port 5 is LDO20 ~ LDO23
LIO_OUT_EN6	Port 6 is LDO24 ~ LDO27

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value	true Successfully disabled.
	false The specified parameter port is not in the setting range.
Description	Disable the output function of the specified port.
Remark	This function is applicable to all the EPCIO Series control cards.

II.6.5 EPCIO_LIO_SetLDIIntType()

BOOL EPCIO_LIO_SetLDIIntType(WORD inputno, WORD type, WORD card_index)

Parameters

inputno The number of the LDI digital input (LDI0 ~ LDI7) to be set to be able to trigger an interrupt

type	Interrupt trigger type
LIO_INT_RISE	Rising edge trigger (default)
LIO_INT_FALL	Falling edge trigger
LIO_INT_LEVEL	Level change trigger

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value	true Setting is successful.
	false The specified parameter inputno is not in the setting range.
Description	Set the local digital input in LIO that will be able to trigger an interrupt. The interrupt trigger type can be rising edge trigger, falling edge trigger, or level change trigger. Once this function is set, EPCIO_LIO_EnableLDIInt() must also be set.
Remark	This function is applicable to all the EPCIO Series control cards.
See also	EPCIO_LIO_EnableLDIInt()

II.6.6 EPCIO_LIO_EnableLDIInt()

BOOL EPCIO_LIO_EnableLDIInt(WORD point, WORD card_index)

Parameters	***point***	LIO interrupt input	
	LIO_LDI0	Local digital input 0 interrupt	Axis 0_ OT+
	LIO_LDI1	Local digital input 1 interrupt	Axis 1_ OT+
	LIO_LDI2	Local digital input 2 interrupt	Axis 2_ OT+
	LIO_LDI3	Local digital input 3 interrupt	Axis 3_ OT+
	LIO_LDI4	Local digital input 4 interrupt	Axis 4_ OT+
	LIO_LDI5	Local digital input 5 interrupt	Axis 5_ OT+
	LIO_LDI6	Local digital input 6 interrupt	Axis 1_ OT-
	LIO_LDI7	Local digital input 7 interrupt	

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false The specified parameter point is not in the setting range.

Description Enable the interrupt function of LIO inputs. Before calling this function, EPCIO_LIO_SetLDIIntType() must be called to set the interrupt trigger type.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_LIO_SetLDIIntType()

II.6.7 EPCIO_LIO_DisableLDIInt()

BOOL EPCIO_LIO_DisableLDIInt(WORD point, WORD card_index)

Parameters	***point***	LIO interrupt input
	LIO_LDI0	Local digital input 0 interrupt
	LIO_LDI1	Local digital input 1 interrupt
	LIO_LDI2	Local digital input 2 interrupt
	LIO_LDI3	Local digital input 3 interrupt
	LIO_LDI4	Local digital input 4 interrupt
	LIO_LDI5	Local digital input 5 interrupt
	LIO_LDI6	Local digital input 6 interrupt
	LIO_LDI7	Local digital input 7 interrupt

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.

false The specified parameter point is not in the setting range.

Description Disable the interrupt function of LIO inputs. Inhibit the interrupt trigger condition of the local digital input capable of triggering an interrupt.

Remark This function is applicable to all the EPCIO Series control cards.

II.6.8 EPCIO_LIO_SetTimer()

BOOL EPCIO_LIO_SetTimer(DWORD value, WORD card_index)

Parameters ***value*** The content of a 24-bit timer, to be set in the range of 0

	~ 2^{24}.	
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	The specified parameter value is out of the setting range.
Description	Set an LIO timer value. When the time expires, an LIO timer interrupt signal will be trigger. Keep in mind that EPCIO_LIO_EnableTimer() and EPCIO_LIO_EnableTimerInt() must be called. The timer's timing unit is System Clock period.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_LIO_EnableTimer() EPCIO_LIO_EnableTimerInt()	

II.6.9 EPCIO_LIO_EnableTimer()

BOOL EPCIO_LIO_EnableTimer(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	Enabling has failed.
Description	Enable a timer. Set the timer period before calling this function.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_LIO_SetTimer()	

II.6.10 EPCIO_LIO_DisableTimer()

BOOL EPCIO_LIO_DisableTimer(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully disabled.
	false	Disabling has failed.
Description	Disable a timer.	

II.6.11 EPCIO_LIO_EnableTimerInt()

BOOL EPCIO_LIO_EnableTimerInt(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	Enabling has failed.
Description	Enable the interrupt-triggering function of a timer. Set and enable the timer before calling this function.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_LIO_SetTimer(), EPCIO_LIO_EnableTimer()	

II.6.12 EPCIO_LIO_DisableTimerInt()

BOOL EPCIO_LIO_DisableTimerInt(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully disabled.

	false	Disabling has failed.
Description	Disable the interrupt-triggering function of a timer.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.6.13 EPCIO_LIO_SetWDogTimer()

BOOL EPCIO_LIO_SetWDogTimer(WORD value, WORD card_index)

Parameters	***value***	Watchdog timer value
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	Setting has failed.
Description	Set a comparison value for a watchdog timer. The time base of the watchdog timer is the LIO timer period, so EPCIO_LIO_SetTimer() must be called. To enable the watchdog function, call EPCIO_LIO_EnableWDogTimer().	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_LIO_SetTimer() EPCIO_LIO_EnableWDogTimer()	

II.6.14 EPCIO_LIO_EnableWDogTimer()

BOOL EPCIO_LIO_EnableWDogTimer(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	Enabling has failed.
Description	Enable a watchdog timer. Before calling this function, EPCIO_LIO_SetWDogTimer() must be set.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_LIO_SetWDogTimer()	

II.6.15 EPCIO_LIO_DisableWDogTimer()

BOOL EPCIO_LIO_DisableWDogTimer(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully disabled.
	false	Disabling has failed.
Description	Disable a watchdog timer.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.6.16 EPCIO_LIO_SetWDogReset()

BOOL EPCIO_LIO_SetWDogReset(DWORD value, WORD card_index)

Parameters	***value***	The setting of a 24-bit Reset register, to be set in the range of 0 ~ 2^{24}
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Setting is successful.
	false	The specified parameter value is not in the setting range.

Description Set the Reset signal duration of a watchdog timer. Reset will be triggered upon timeout of the watchdog timer. The duration of Reset can be programmed with this function, in units of System Clock.

Remark This function is applicable to all the EPCIO Series control cards.

II.6.17 EPCIO_LIO_RefreshWDogTimer()

BOOL EPCIO_LIO_RefreshWDogTimer(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully refreshed.
false Refreshing has failed.

Description Refresh the time of a watchdog timer. Once the watchdog function is enabled, this function must be called within a fixed time period in order to refresh the content of the watchdog timer. Otherwise, Reset will be triggered upon timeout.

Remark This function is applicable to all the EPCIO Series control cards.

II.6.18 EPCIO_LIO_GetOverTravelUp()

BOOL EPCIO_LIO_GetOverTravelUp(WORD point, WORD *overtravel, WORD card_index)

Parameters

point	Forward-direction over-travel status number
LIO_OT0	Axis-0 over-travel input
LIO_OT1	Axis-1 over-travel input
LIO_OT2	Axis-2 over-travel input
LIO_OT3	Axis-3 over-travel input
LIO_OT4	Axis-4 over-travel input
LIO_OT5	Axis-5 over-travel input

overtravel	Forward-direction over-travel status value
0	No over-travel
1	Over-travel

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Data successfully acquired.
false Data acquisition has failed.

Description Check whether the forward-direction travel limit of the specified axis is exceeded. If yes, the machine is prone to collision, which is dangerous, and the user should take necessary measures immediately. This function acquire data by polling registers' data.

Remark This function is applicable to all the EPCIO Series control cards.

II.6.19 EPCIO_LIO_GetOverTravelDown()

BOOL EPCIO_LIO_GetOverTravelDown(WORD point, WORD *overtravel, WORD card_index)

Parameters

point	Reverse-direction over-travel status number
LIO_OT0	Axis-0 over-travel input

	LIO_OT1	Axis-1 over-travel input
	LIO_OT2	Axis-2 over-travel input
	LIO_OT3	Axis-3 over-travel input
	LIO_OT4	Axis-4 over-travel input
	LIO_OT5	Axis-5 over-travel input
	overtravel	Reverse-direction over-travel status value
	0	No over-travel
	1	Over-travel
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	Data acquisition has failed.
Description	Check whether the reverse-direction travel limit of the specified axis is exceeded. If yes, the machine is prone to collision, which is dangerous, and the user should take necessary measures immediately. This function acquire data by polling registers' data.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.6.20 EPCIO_LIO_GetHomeSensor()

BOOL EPCIO_LIO_GetHomeSensor(WORD point, WORD *home, WORD card_index)

Parameters	***point***	HOME sensor number
	LIO_HOME0	Axis-0 HOME sensor input
	LIO_HOME1	Axis-1 HOME sensor input
	LIO_HOME2	Axis-2 HOME sensor input
	LIO_HOME3	Axis-3 HOME sensor input
	LIO_HOME4	Axis-4 HOME sensor input
	LIO_HOME5	Axis-5 HOME sensor input
	home	HOME sensor status value
	0	Not triggered
	1	HOME sensor triggered
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	Data acquisition has failed.
Description	Acquire the HOME sensor status of the specified axis. This function does not acquire data via an interrupt, but by checking registers' data.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.6.21 EPCIO_LIO_GetEmgcStopStatus()

BOOL EPCIO_LIO_GetEmgcStopStatus(WORD *estop, WORD card_index)

Parameters	***estop***	Status value of emergency stop switch input
	0	Not triggered

	1	Emergency stop switch triggered
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Data successfully acquired.
	false	Data acquisition has failed.
Description	Acquire the status of an emergency stop switch. This function does not acquire data via an interrupt, but by checking registers' data.	
Remark	This function is applicable to all the EPCIO Series control cards.	

II.6.22 EPCIO_LIO_ServoOff()

BOOL EPCIO_LIO_ServoOff(WORD channel, WORD card_index)

Parameters	***channel***	The number of an ServoON switch
	LIO_INH0	Axis-0 ServoON switch
	LIO_INH1	Axis-1 ServoON switch
	LIO_INH2	Axis-2 ServoON switch
	LIO_INH3	Axis-3 ServoON switch
	LIO_INH4	Axis-4 ServoON switch
	LIO_INH5	Axis-5 ServoON switch
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	Enabling has failed.
Description	Servo off the servo drive for specified axis. The connection terminal can be connected with the ServoON connection terminal of the motor drive. Once this function is called, the specified axis can no longer receive position or velocity commands. After the initialization function is called, ServoON is disabled by default.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_LIO_ServoOn()	

II.6.23 EPCIO_LIO_ServoOn()

BOOL EPCIO_LIO_ServoOn(WORD channel, WORD card_index)

Parameters	***channel***	The number of an ServoON switch
	LIO_INH0	Axis-0 ServoON switch
	LIO_INH1	Axis-1 ServoON switch
	LIO_INH2	Axis-2 ServoON switch
	LIO_INH3	Axis-3 ServoON switch
	LIO_INH4	Axis-4 ServoON switch
	LIO_INH5	Axis-5 ServoON switch
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	Enabling has failed.
Description	Servo on the servo drive for specified axis. The connection terminal can be connected with the ServoON connection terminal of the	

motor drive. Once this function is called, the specified axis can receive position or velocity commands from an EPCIO Series control card. After the initialization function is called, ServoON is disabled by default.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_LIO_ServoOff()

II.6.24 EPCIO_LIO_EnablePrdy()

BOOL EPCIO_LIO_EnablePrdy(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false Enabling has failed.

Description Enable Position Ready output. The output connection terminal can be connected with the connection terminal for power switch control. Once this function is called, the connection terminal is closed-circuited. After the initialization function is called and set, Position Ready output is disabled by default.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_LIO_DisablePrdy()

II.6.25 EPCIO_LIO_DisablePrdy()

BOOL EPCIO_LIO_DisablePrdy(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.

false Disabling has failed.

Description Disable Position Ready output. The output connection terminal can be connected with the connection terminal for power switch control. Once this function is called, the connection terminal is open-circuited. After the initialization function is called and set, Position Ready output is disabled by default.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_LIO_EnablePrdy()

II.6.26 EPCIO_LIO_EnablePulseDAC()

BOOL EPCIO_LIO_EnablePulseDAC(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

false Enabling has failed.

Description Enable position command and voltage command output from EPCIO Series modules. Once this function is set, the output function is enabled. After the initialization function is called and set, the output is disabled by default.

Remark This function is applicable to all the EPCIO Series control cards.

See also EPCIO_LIO_DisablePulseDAC()

II.6.27 EPCIO_LIO_DisablePulseDAC()

BOOL EPCIO_LIO_DisablePulseDAC(WORD card_index)

Parameters	***card_index***	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully disabled.
	false	Disabling has failed.
Description	Disable position command and voltage command output from EPCIO Series modules. Once this function is set, the output function is disabled. After the initialization function is called and set, the output is disabled by default.	
Remark	This function is applicable to all the EPCIO Series control cards.	
See also	EPCIO_LIO_EnablePulseDAC()	

II.7. PCL Control

II.7.1 EPCIO_PCL_GetErrorCounter()

BOOL EPCIO_PCL_GetErrorCounter(WORD channel, int *error, WORD card_index)

Parameters ***channel*** Error counter channel number (0 ~ 5)

error Error counter value

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Data successfully acquired.

false The specified parameter *channel* is not in the setting range.

Description Acquire the position error value (the difference of position command and encoder feedback) of the specified axis. Before calling this function, EPCIO_PCL_EnablErrorCounter() must be activated.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

See also EPCIO_PCL_EnableErrorCounter()

II.7.2 EPCIO_PCL_SetScaleGain()

BOOL EPCIO_PCL_SetScaleGain(WORD channel, WORD pgain, int sgain, WORD card_index)

Parameters ***channel*** PCL channel number (0 ~ 5)

pgain closed-loop proportion gain (0 ~ 127)

sgain closed-loop scaling gain (-7 ~ 7)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified parameter *channel*, *pgain*, or *sgain* is not in the corresponding setting range.

Description Set the scaling gain value of a closed-loop controlled axis. The closed-loop gain value is composed of a proportional term (Kp1) and a scaling factor term (Kp2), i.e., gain = Kp1 × Kp2 / 16, where Kp1 is pgain and Kp2 is 2^{-sgain}.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

II.7.3 EPCIO_PCL_EnableOverflowInt()

BOOL EPCIO_PCL_EnableOverflowInt(WORD channel, WORD card_index)

Parameters ***channel*** PCL channel number (0 ~ 5)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.

	false	The specified parameter *channel* is not in the setting range.
Description	Enable the function by which an interrupt is triggered by overflow of the error counter of the specified channel. When the difference of the position command and the encoder position exceeds the range the error counter allows, the error counter generates an overflow interrupt notification and automatically outputs a 0V voltage from DAC.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.	

II.7.4 EPCIO_PCL_DisableOverflowInt()

BOOL EPCIO_PCL_DisableOverflowInt(WORD channel, WORD card_index)

Parameters	***channel***	Error counter channel number (0 ~ 5)
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully disabled.
	false	The specified parameter *channel* is not in the setting range.
Description	Disable the function by which an interrupt is triggered by overflow of the error counter of the specified channel.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.	

II.7.5 EPCIO_PCL_ClearCounter()

BOOL EPCIO_PCL_ClearCounter(WORD channel, WORD card_index)

Parameters	***channel***	Error counter channel number (0 ~ 5)
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully cleared.
	false	The specified parameter *channel* is not in the setting range.
Description	Clear the count and overflow status of the error counter.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.	

II.7.6 EPCIO_PCL_EnableErrorCounter()

BOOL EPCIO_PCL_EnableErrorCounter(WORD channel, WORD card_index)

Parameters	***channel***	Error counter channel number (0 ~ 5)
	card_index	The index of the motion control card to be controlled. The index ranges from 0 to 11.
Return Value	true	Successfully enabled.
	false	The specified parameter *channel* is not in the setting range.
Description	Enable the error counting function of closed-loop command. Once this function is enabled, EPCIO_PCL_ StartControl() must be called to enable the hardware closed-loop counting function.	
Remark	This function is applicable only to the EPCIO-4000, and EPCIO-	

6000 control cards.

See also EPCIO_PCL_StartControl()

II.7.7 EPCIO_PCL_DisableErrorCounter()

BOOL EPCIO_PCL_DisableErrorCounter(WORD channel, WORD card_index)

Parameters ***channel*** Error counter channel number (0 ~ 5)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified parameter *channel* is not in the setting range.

Description Disable the error counting function of closed-loop command on the specified channel.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

II.7.8 EPCIO_PCL_StartControl()

BOOL EPCIO_PCL_StartControl(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully started.
false Starting has failed.

Description Enable PCL control. Before calling this function, EPCIO_PCL_EnableErrorCounter() must be called to enable the error counting function.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

See also EPCIO_PCL_EnableErrorCounter()

II.7.9 EPCIO_PCL_StopControl()

BOOL EPCIO_PCL_StopControl(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully stopped.
false Stopping has failed.

Description Disable PCL control. This function disables the error counting function in all channels.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

II.8. DAC I/O Control

II.8.1 EPCIO_DAC_SetOutput()

BOOL EPCIO_DAC_SetOutput(WORD channel, float voltage, WORD card_index)

Parameters ***channel*** DAC channel number (0 ~ 7)

voltage Analog output voltage (-10V ~ 10V)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified parameter *channel* or *voltage* is not in the corresponding setting range.

Description Set the output voltage value of a DAC channel. EPCIO_DAC_SetCmdSource() must be called in advance to set the source mode of DAC to *DAC_CMD_SOFT*.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

See also EPCIO_DAC_SetCmdSource()

II.8.2 EPCIO_DAC_SetTrigOutput()

BOOL EPCIO_DAC_SetTrigOutput(WORD channel, float voltage, WORD card_index)

Parameters ***channel*** DAC channel number (0 ~ 7)

voltage Analog output voltage (-10V ~ 10V)

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified parameter *channel* or *voltage* is not in the corresponding setting range.

Description Set the voltage value that a DAC channel will immediately output upon satisfaction of the trigger condition. When the DAC is programmed to be “software command mode”, a hardware-triggered voltage value can be set and modified before the trigger condition is reached. If the trigger condition is reached, then the voltage value will be sent out from the hardware.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

See also EPCIO_DAC_SetCmdSource() EPCIO_DAC_SetTrigSource() EPCIO_DAC_EnableTrigMode()

II.8.3 EPCIO_DAC_SetTrigSource()

BOOL EPCIO_DAC_SetTrigSource(WORD channel, DWORD source, WORD card_index)

Parameters ***channel*** DAC channel number (0 ~ 7)

source DAC hardware trigger source(s), expressed in bits, to be selected from a total of 32 interrupt trigger sources. Plural trigger sources can be set at the same time. The eligible sources are defined with constants as follows:

DAC_TRIG_ENC0	Encoder counter channel 0 comparator interrupt
DAC_TRIG_ENC1	Encoder counter channel 1 comparator interrupt
DAC_TRIG_ENC2	Encoder counter channel 2 comparator interrupt
DAC_TRIG_ENC3	Encoder counter channel 3 comparator interrupt
DAC_TRIG_ENC4	Encoder counter channel 4 comparator interrupt
DAC_TRIG_ENC5	Encoder counter channel 5 comparator interrupt
DAC_TRIG_ENC6	Encoder counter channel 6 comparator interrupt
DAC_TRIG_ENC7	Encoder counter channel 7 comparator interrupt
DAC_TRIG_ADC0	ADC channel 0 comparator interrupt
DAC_TRIG_ADC1	ADC channel 1 comparator interrupt
DAC_TRIG_ADC2	ADC channel 2 comparator interrupt
DAC_TRIG_ADC3	ADC channel 3 comparator interrupt
DAC_TRIG_ADC4	ADC channel 4 comparator interrupt
DAC_TRIG_ADC5	ADC channel 5 comparator interrupt
DAC_TRIG_ADC6	ADC channel 6 comparator interrupt
DAC_TRIG_ADC7	ADC channel 7 comparator interrupt
DAC_TRIG_LDI0	Local I/O LDI0 input interrupt
DAC_TRIG_LDI1	Local I/O LDI1 input interrupt
DAC_TRIG_LDI2	Local I/O LDI2 input interrupt
DAC_TRIG_LDI3	Local I/O LDI3 input interrupt
DAC_TRIG_DFI0	Local double function DFI0 input interrupt
DAC_TRIG_DFI1	Local double function DFI1 input interrupt
DAC_TRIG_DFI2	Local double function DFI2 input interrupt
DAC_TRIG_DFI3	Local double function DFI3 input interrupt
DAC_TRIG_R0DI0	Remote I/O Set 0 Slave 0 DI0 input interrupt
DAC_TRIG_R0DI1	Remote I/O Set 0 Slave 0 DI1 input interrupt
DAC_TRIG_R0DI2	Remote I/O Set 0 Slave 0 DI2 input interrupt
DAC_TRIG_R0DI3	Remote I/O Set 0 Slave 0 DI3 input interrupt
DAC_TRIG_R1DI0	Remote I/O Set 1 Slave 0 DI0 input interrupt
DAC_TRIG_R1DI1	Remote I/O Set 1 Slave 0 DI1 input interrupt
DAC_TRIG_R1DI2	Remote I/O Set 1 Slave 0 DI2 input interrupt
DAC_TRIG_R1DI3	Remote I/O Set 1 Slave 0 DI3 input interrupt

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified parameter *channel* or *source* is not in the corresponding setting range.

Description Set the function by which a DAC channel immediately outputs voltage when the interrupt condition is met. Several interrupt conditions can be set for each DAC channel. Once this function is set, EPCIO_DAC_EnableTrigMode() must also be set to enable the trigger mode. This function is valid only when the DAC is in the software command mode. To set the command source, please refer to EPCIO_DAC_SetCmdSource().

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

See also EPCIO_DAC_SetCmdSource()
EPCIO_DAC_EnableTrigMode()

II.8.4 EPCIO_DAC_SetClockDivider()

BOOL EPCIO_DAC_SetClockDivider(WORD divider, WORD card_index)

Parameters ***divider*** DAC transmission clock divider (0 ~ 255 clock)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.
false The specified parameter *divider* is not in the setting range.

Description Set the DAC serial interface transmission clock frequency. The DAC transmission clock frequency is System Clock (40MHz) divided by 2 × (divider + 1). The default divider is 0.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

II.8.5 EPCIO_DAC_EnableTrigMode()

BOOL EPCIO_DAC_EnableTrigMode(WORD channel, WORD card_index)

Parameters ***channel*** DAC channel number (0 ~ 7)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully enabled.
false The specified parameter *channel* is not in the setting range.

Description Enable a DAC channel’s trigger mode so that an interrupt can trigger immediately voltage output from the DAC channel. Set the trigger source(s) before enabling the trigger mode.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

See also EPCIO_DAC_SetTrigSource()

II.8.6 EPCIO_DAC_DisableTrigMode()

BOOL EPCIO_DAC_DisableTrigMode(WORD channel, WORD card_index)

Parameters ***channel*** DAC channel number (0 ~ 7)
card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully disabled.
false The specified parameter *channel* is not in the setting

range.

Description Disable a DAC channel's trigger mode so that an interrupt will not trigger voltage output from the DAC channel.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

II.8.7 EPCIO_DAC_SetCmdSource()

BOOL EPCIO_DAC_SetCmdSource(WORD channel, WORD source, WORD card_index)

Parameters

channel	DAC channel number (0 ~ 7)
source	DAC command source
DAC_CMD_SOFT	DAC output buffer
DAC_CMD_PCL	PCL closed-loop output

card_index The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Setting is successful.

false The specified parameter *channel* or *source* is not in the corresponding setting range.

Description Set the source of DAC channel output commands from either software programming or hardware PCL closed-loop mode. When the command source is PCL, the commands are generated by the position errors in the PCL. When the command source is set to software programming mode, EPCIO_DAC_SetOutput() can be called to set the output voltage and EPCIO_DAC_SetTrigOutput() also can be called to set the trigger output voltage.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

See also EPCIO_DAC_SetOutput(), EPCIO_DAC_SetTrig_Output()

II.8.8 EPCIO_DAC_StartConv()

BOOL EPCIO_DAC_StartConv(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully started.

false Starting has failed.

Description Start output voltage conversion in the specified DAC channel. Analog/digital conversion starts as soon as this function is set.

Remark This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.

II.8.9 EPCIO_DAC_StopConv()

BOOL EPCIO_DAC_StopConv(WORD card_index)

Parameters ***card_index*** The index of the motion control card to be controlled. The index ranges from 0 to 11.

Return Value true Successfully stopped.

false Stopping has failed.

Description Stop output voltage conversion in the specified DAC channel.

Remark	This function is applicable only to the EPCIO-4000, and EPCIO-6000 control cards.