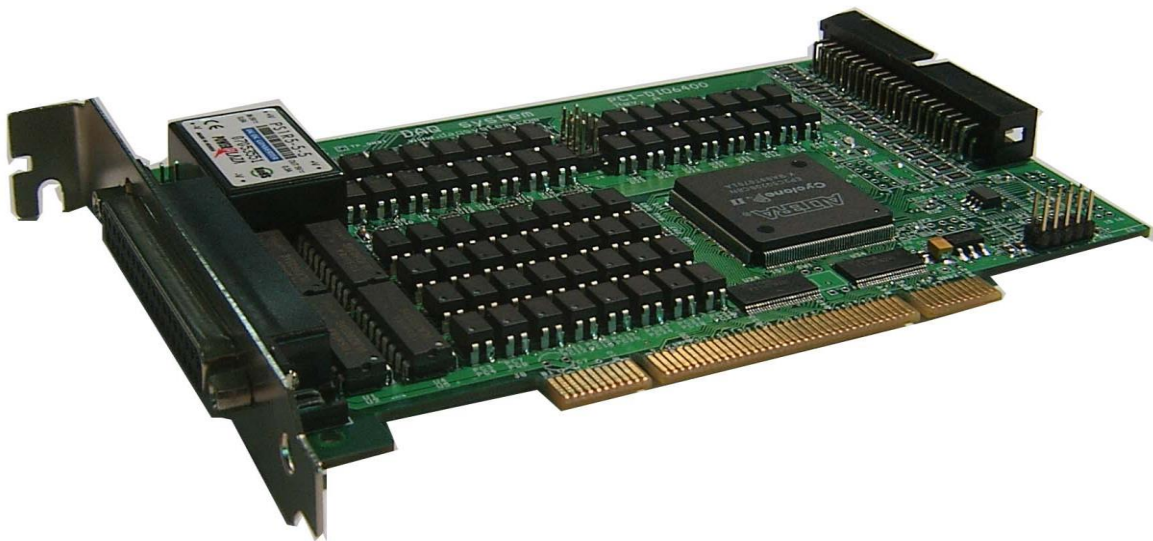


# (c)PCI(e)-DIO64xx Series API Programming (Rev 1.0)



**Windows, Windows2000, Windows NT, Windows XP and Windows 7** are trademarks of **Microsoft**. We acknowledge that the trademarks or service names of all other organizations mentioned in this document as their own property.

Information furnished by DAQ system is believed to be accurate and reliable. However, no responsibility is assumed by DAQ system for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or copyrights of DAQ system.

The information in this document is subject to change without notice and no part of this document may be copied or reproduced without the prior written consent.

Copyrights © 2012 DAQ system Co., LTD. All rights reserved.

## API

This chapter explains API (Application Programming Interface) to use a (c)PCI(e)-DIO6400.

Currently it supports API as follows.

```
////////////////////////////////////  
// (c) 2007 DAQ system  
////////////////////////////////////  
#define MAX_MODEL 5 // Maximum number of Device Model  
#define MAX_DEVICE 5 // Maximum number of Device to find  
  
#define PCI_DIO6400 0  
#define PCI_DIO6401 1  
#define PCI_DIO6402 2  
#define PCI_DIO6403 3  
#define PCI_DIO6404 4  
  
// Board level API functions  
extern "C" __declspec(dllimport) BOOL __stdcall OpenDAQDevice(int nModel, int nBoard);  
extern "C" __declspec(dllimport) BOOL __stdcall ResetBoard(int nModel, int nBoard);  
extern "C" __declspec(dllimport) BOOL __stdcall CloseDAQDevice(int nModel, int nBoard);  
  
// digital input/output  
extern "C" __declspec(dllimport) BOOL __stdcall DIN_Read(int nModel, int nBoard, int nReg, DWORD *dwVal);  
extern "C" __declspec(dllimport) BOOL __stdcall DOUT_Read(int nModel, int nBoard, int nReg, DWORD *dwVal);  
extern "C" __declspec(dllimport) BOOL __stdcall DOUT_Write(int nModel, int nBoard, int nReg, DWORD dwVal);  
  
extern "C" __declspec(dllimport) BOOL __stdcall Set_LED_Mode(int nModel, int nBoard, int nReg, int nMode);
```

## Board Level API Functions

### Overview

BOOL	OpenDAQDevice (int nModel, int nBoard)
BOOL	ResetBoard (int nModel, int nBoard)
BOOL	CloseDAQDevice (int nModel, int nBoard)

## OpenDAQDevice

### **BOOL            OpenDAQDevice (int nModel, int nBoard)**

It opens a device. You may call this function at the very first time you run the program and some suspicious operation.

**Parameters:**

nModel : It writes down the PCI-DIO6400 model number. (0 ~ 3)

nBoard : It informs a board number at currently equipped system.

The board number set up by DIP switch.

**Return Value:**

If the function succeeds, it returns the number of boards which were detected.

If the function fails, the return value is -1, it means there is no device in the system.

## ResetBoard

### **BOOL            ResetBoard (int nModel, int nBoard)**

It initializes a device at currently equipped system (PC).

**Parameters:**

nModel : It writes down the PCI-DIO6400 model number. (0 ~ 3)

nBoard : It informs a board number at currently equipped system.

The board number set up by DIP switch.

**Return Value:**

If the function fail to reset, it returns "FALSE".

If the function succeed to reset, it returns "TRUE".

## CloseDAQDevice

### **BOOL            CloseDAQDevice (int nModel, int nBoard)**

The CloseDAQDevice function closes all opened devices (boards). If use of device is finished, it can certainly close a device for making it other programs so as usable.

**Parameters:**

nModel : It writes down the PCI-DIO6400 model number. (0 ~ 3)

nBoard : It informs a board number at currently equipped system.

The board number set up by DIP switch.

**Return Value:**

If the function fail to close, it returns "FALSE".

If the function succeed to close, it returns "TRUE".

## DIO(Digital Input Output) API Functions

### Overview

BOOL	DIN_Read (int nModel, int nBoard, int nReg, DWORD *dwVal)
BOOL	DOUT_Write (int nModel, int nBoard, int nReg, DWORD dwVal)
BOOL	DOUT_Read (int nModel, int nBoard, int nReg, DWORD *dwVal)
BOOL	Set_LED_Mode (int nModel, int nBoard, int nReg, int nMode)

### DIN\_Read

**BOOL**            **DIN\_Read (int nModel, int nBoard, int nReg, DWORD \*dwVal)**

It reads 32bit or 64bit Digital Input value. (for PCI-DIO6400 or PCI-DIO6401)

**Parameters:**

nModel : It writes down the PCI-DIO6400 model number. (0 ~ 3)

nBoard : It informs a board number at currently equipped system.

The board number set up by DIP switch.

nReg : It ready for future.

It is not used currently, but you must setup '0'.

\*wVal : It is a pointer of currently value to get from input port.

**Return Value:**

If the function call fails, it returns "FALSE".

If the function call succeeds, it returns "TRUE".

### DOUT\_Write

**BOOL**            **DOUT\_Write (int nModel, int nBoard, int nReg, DWORD dwVal)**

It prints out 32bit or 64bit Digital value at output port. (for PCI-DIO6400 or PCI-DIO6402)

**Parameters:**

nModel : It writes down the PCI-DIO6400 model number. (0 ~ 3)

nBoard : It informs a board number at currently equipped system.

The board number set up by DIP switch.

nReg : : It ready for future.

It is not used currently, but you must setup '0'.

wVal : The value to record to an output port.

**Return Value:**

If the function call fails, it returns "FALSE".

If the function call succeeds, it returns "TRUE".

## DOUT\_Read

**BOOL**            **DOUT\_Read (int nModel, int nBoard, int nReg, DWORD \*dwVal)**

It reads a currently output value with 32 or 64bits. (for PCI-DIO6400 or PCI-DIO6402)

**Parameters:**

nModel : It writes down the PCI-DIO6400 model number. (0 ~ 3)

nBoard : It informs a board number at currently equipped system.

The board number set up by DIP switch.

nReg : : It ready for future.

It is not used currently, but you must setup '0'.

\*dwVal : It is a pointer of currently value to get from output port.

**Return Value:**

If the function call fails, it returns "FALSE".

If the function call succeeds, it returns "TRUE".

## Set\_LED\_Mode

**BOOL**            **Set\_LED\_Mode (int nModel, int nBoard, int nReg, int nMode)**

It marks a current value of input/output to LED on a front panel. It can set up to display what kind of value. (for cPCI-DIO6400)

**Parameters:**

nModel : It writes down the PCI-DIO6400 model number. (0 ~ 3)

nBoard : It informs a board number at currently equipped system.

The board number set up by DIP switch.

nReg : : It ready for future.

It is not used currently, but you must setup '0'.

nMode : It sets up a value to be displayed to LED.

0 : Input from 15 to 0

1 : Input from 31 to 16

2 : Output from 15 to 0

3 : Output from 31 to 16.

(Notice) It is not defined the above value, don't use it.

**Return Value:**

If the function call fails, it returns "FALSE".

If the function call succeeds, it returns "TRUE".