

**EX94421**

**Analog Input Card**

**Software Manual (V1.0)**

**旭蒙科技股份有限公司**

**TOPSCCC TECHNOLOGY CO., LTD.**

台北市內湖區陽光街 345 巷 12 號 5 樓

5F., No.12, Lane 345, Yang Guang St.

Neihu, 114 Taipei, Taiwan

TEL : +886-2-2799-9080

FAX : +886-2-2658-5042

<http://www.topscce.com>

E-mail : jane.w@topscce.com

## Correction record

Version	Record
1.0	

# Contents

1.	How to install the software of EX94421 .....	3
1.1	Install the PCI driver.....	3
2.	About the EX94421 software .....	4
2.1	What you need to get started.....	4
2.2	Software programming choices .....	4
3.	EX94421 Language support .....	5
3.1	Building applications with the EX94421 software library .....	5
	Applications tools .....	5
3.2	EX94421 Windows libraries .....	5
4.	Function format and language difference .....	6
4.1	Function format.....	6
4.2	Variable data types .....	7
4.3	Programming language considerations .....	8
5.	Software overview and dll function.....	10
5.1	Initialization and close .....	10
	EX94421_initial.....	10
	EX94421_close.....	10
	EX94421_info.....	10
5.2	Analog input .....	11
	EX94421_AD_config_set.....	11
	EX94421_AD_config_read .....	12
	EX94421_AD_value_read.....	12
	EX94421_AD_data_read.....	13
	EX94421_AD_integral_start .....	14
	EX94421_AD_integral_all_read .....	14
	EX94421_AD_integral_stop .....	15
5.3	Error conditions .....	16
6.	Flow chart of application implementation.....	17
6.1	EX94421 Flow chart of analog I/O application implementation.....	17
6.2	EX94421 Flow chart of analog I/O application with embedded integration function.....	18
7.	Dll list .....	19
8.	EX94421 Error codes summary.....	20
8.1	EX94421 Error codes table.....	20

# **1. How to install the software of EX94421**

## **1.1 Install the PCI driver**

The PCI card is a plug and play card, once you add a new card the on window system will detect while it is booting. Please follow the following steps to install your new card.

In Windows 2000/XP/Win7 system you should: (take Win XP as example)

1. Make sure the power is off
2. Plug in the interface card
3. Power on
4. A hardware install wizard will appear and tell you it finds a new PCI card
5. Tell the wizard the directory of the driver files (..\EX94421\software\Win2K\_up\driver or if you download from website please execute the self-unzip file EX94421\_driver.exe to get the file), then it will automatically setup the driver
6. After installation, power off
7. Power on, it's ready to use

## 2. **About the EX94421 software**

EX94421 software includes a set of dynamic link library (DLL) and system driver that you can utilize to control the I/O card's ports and points separately.

Your EX94421 software package includes setup driver, tutorial example and test program that help you how to setup and run appropriately, as well as an executable file which you can use to test each of the EX94421 functions within Windows' operation system environment.

### 2.1 What you need to get started

To set up and use your EX94421 software, you need the following:

- EX94421 software
- EX94421 hardware

Main board

Wiring board (Option)

### 2.2 Software programming choices

You have several options to choose from when you are programming EX94421 software. You can use Borland C/C++, Microsoft Visual C/C++, Microsoft Visual Basic, or any other Windows-based compiler that can call into Windows dynamic link libraries (DLLs) for use with the EX94421 software.

### 3. **EX94421 Language support**

The EX94421 software library is a DLL used with Windows 2000/XP/Win7. You can use these DLL with any Windows integrating development environment that can call Windows DLLs.

#### 3.1 Building applications with the EX94421 software library

The EX94421 function reference topic contains general information about building EX94421 applications, describes the nature of the EX94421 files used in building EX94421 applications, and explains the basics of making applications using the following tools:

##### **Applications tools**

- Borland C/C++
- Microsoft Visual C/C++
- Microsoft Visual Basic

If you are not using one of the tools listed, consult your development tool reference manual for details on creating applications that call DLLs.

#### 3.2 EX94421 Windows libraries

The EX94421 for Windows function library is a DLL called **EX94421.dll**. Since a DLL is used, EX94421 functions are not linked into the executable files of applications. Only the information about the EX94421 functions in the EX94421 import libraries is stored in the executable files.

Import libraries contain information about their DLL-exported functions. They indicate the presence and location of the DLL routines. Depending on the development tools you are using, you can make your compiler and linker aware of the DLL functions through import libraries or through function declarations.

Refer to **Table 1** to determine to which files you need to link and which to include in your development to use the EX94421 functions in EX94421.dll.

<b>Header Files and Import Libraries for Different Development Environments</b>		
Development Environment	Header File	Import Library
Microsoft C/C++	EX94421.h	EX94421vc.lib
Borland C/C++	EX94421.h	EX94421bc.lib
Microsoft Visual Basic	EX94421.bas	

**Table 1**

## 4. **Function format and language difference**

### 4.1 Function format

Every EX94421 function is consist of the following format:

Status = function\_name (parameter 1, parameter 2, ... parameter n)

Each function returns a value in the **Status** global variable that indicates the success or failure of the function. A returned **Status** equal to zero that indicates the function executed successfully. A non-zero status indicates failure that the function did not execute successfully because of an error, or executed with an error.

**Note** : **Status** is a 32-bit unsigned integer.

The first parameter to almost every EX94421 function is the parameter **CardID** which is located the driver of EX94421 board you want to use those given operation. The **CardID** is assigned by DIP/ROTARY SW. You can utilize multiple devices with different card CardID within one application; to do so, simply pass the appropriate **CardID** to each function.

**Note:** **CardID** is set by DIP/ROTARY SW (**0x0-0xF**)

## 4.2 Variable data types

Every function description has a parameter table that lists the data types for each parameter. The following sections describe the notation used in those parameter tables and throughout the manual for variable data types.

Primary Type Names					
Name	Description	Range	C/C++	Visual BASIC	Pascal (Borland Delphi)
u8	8-bit ASCII character	0 to 255	char	Not supported by BASIC. For functions that require character arrays, use string types instead.	Byte
i16	16-bit signed integer	-32,768 to 32,767	short	Integer (for example: deviceNum%)	SmallInt
u16	16-bit unsigned integer	0 to 65,535	unsigned short for 32-bit compilers	Not supported by BASIC. For functions that require unsigned integers, use the signed integer type instead. See the i16 description.	Word
i32	32-bit signed integer	-2,147,483,648 to 2,147,483,647	long	Long (for example: count&)	LongInt
u32	32-bit unsigned integer	0 to 4,294,967,295	unsigned long	Not supported by BASIC. For functions that require unsigned long integers, use the signed long integer type instead. See the i32 description.	Cardinal (in 32-bit operating systems). Refer to the i32 description.
f32	32-bit single-precision floating-point value	-3.402823E+38 to 3.402823E+38	float	Single (for example: num!)	Single
f64	64-bit double-precision floating-point value	-1.797683134862315E+308 to 1.797683134862315E+308	double	Double (for example: voltage Number)	Double

**Table 2**



### 4.3 Programming language considerations

Apart from the data type differences, there are a few language-dependent considerations you need to be aware of when you use the EX94421 API. Read the following sections that apply to your programming language.

**Note:** Be sure to include the declaration functions of EX94421 prototypes by including the appropriate EX94421 header file in your source code. Refer to Building Applications with the EX94421 Software Library for the header file appropriate to your compiler.

#### 4.3.1 C/C++

For C or C++ programmers, parameters listed as Input/Output parameters or Output parameters are pass-by-reference parameters, which means a pointer points to the destination variable should be passed into the function. For example, the Read AD function has the following format:

```
Status = EX94421_AD_data_read(CardID, channel, *voltage_data);
```

where **CardID** and **channel** are input parameters, and **voltage\_data** is an output parameter. Consider the following example:

```
u8 CardID, channel;  
i16 voltage_data,  
u32 Status;  
Status = EX94421_AD_data_read (CardID, channel, voltage_data);
```

#### 4.3.2 Visual basic

The file EX94421.bas contains definitions for constants required for obtaining AIO Card information and declared functions and variable as global variables. You should use these constants symbols in the EX94421.bas, do not use the numerical values.

In Visual Basic, you can add the entire EX94421.bas file into your project. Then you can use any of the constants defined in this file and call these constants in any module of your program. To add the EX94421.bas file for your project in Visual Basic 4.0, go to the **File** menu and select the **Add File...** option. Select EX94421.bas, which is browsed in the EX94421 \ api directory. Then, select **Open** to add the file to the project.

To add the EX94421.bas file to your project in Visual Basic 5.0 and 6.0, go to the **Project** menu and select **Add Module**. Click on the Existing tab page. **Select** EX94421.bas, which is in the EX94421 \ api directory. Then, select **Open** to add the file to the project.

#### 4.3.3 Borland C++ builder

To use Borland C++ builder as development tool, you should generate a .lib file from the .dll file by implib.exe.

```
implib EX94421bc.lib EX94421.dll
```

Then add the **EX94421bc.lib** to your project and add

```
#include "EX94421.h" to main program.
```

Now you may use the dll functions in your program. For example, the Read AD function has the following format:

```
Status = EX94421_AD_data_read(CardID, channel, *voltage_data);
```

where **CardID** and **channel** are input parameters, and **voltage\_data** is an output parameter. Consider the following example:

```
u8 CardID, channel;
```

```
i16 voltage_data,
```

```
u32 Status;
```

```
Status = EX94421_AD_data_read (CardID, channel, voltage_data);
```

## 5. Software overview and dll function

These topics describe the features and functionality of the EX94421 boards and briefly describes the EX94421 functions.

### 5.1 Initialization and close

You need to initialize system resource each time you start to run your application.

*EX94421\_initial( )* will do.

Once you want to close your application, call

*EX94421\_close( )* to release all the resource.

If you want to know the physical address assigned by OS. use

*EX94421\_info( )* to get the address.

#### ● EX94421\_initial

**Format :** u32 Status =EX94421\_initial (void)

**Purpose:** Initial the EX94421 resource when start the Windows applications.

#### ● EX94421\_close

**Format :** u32 Status =EX94421\_close (void);

**Purpose:** Release the EX94421 resource when close the Windows applications.

#### ● EX94421\_info

**Format :** u32 status =EX94421\_info(u8 CardID,u16 \*address)

**Purpose:** Read the physical I/O address assigned by O.S..

**Parameters:**

**Input:**

Name	Type	Description
CardID	u8	assigned by DIP/ROTARY SW

**Output:**

Name	Type	Description
address	u16	physical I/O address assigned by OS

## 5.2 Analog input

The EX94421 now is a 16 bit AD cards. You must configure the input range of the specific channel by:

*EX94421\_AD\_config\_set( )* and read back the configuration for verification by:

*EX94421\_AD\_config\_read( )*

To read the input voltage value by:

*EX94421\_AD\_value\_read( )*, it can be also read data by

*EX94421\_AD\_data\_read( )*

The EX94421 hardware only provide the AD conversion data on the fly, in noisy environment the conversion result maybe contaminated by noise, to use the integral of signals will eliminate the high frequency noise. The dll has provide build in software integration functions; to start the function by:

*EX94421\_AD\_integral\_start( )* and read the integration data by

*EX94421\_AD\_integral\_all\_read( )*, if you want to stop the integration function don't forget to release the resource and stop integration by:

*EX94421\_AD\_integral\_stop( )*

### ● EX94421 AD config set

**Format :** u32 status = EX94421\_AD\_config\_set(u8 CardID,u8 channel,u8 mode)

**Purpose:** Set A/D config.

**Parameters:**

**Input:**

Name	Type	Description
CardID	u8	assigned by DIP/ROTARY SW
channel	u8	A/D channel number 0~7: EX94421, 8 channels AD
mode	u8	scale range: 0: 0V ~ 5V 1: -5V ~ +5V 2: 0V ~ 10V 3: -10V ~ +10V 255 : AD stop operation.

### ● EX94421 AD config read

**Format :** u32 status = EX94421\_AD\_config\_read(u8 CardID,u8 channel,u8 \*mode)

**Purpose:** Read A/D configuration.

**Parameters:**

**Input:**

Name	Type	Description
CardID	u8	assigned by DIP/ROTARY SW
channel	u8	A/D channel number 0~7: EX94421, 8 channels AD

**Output:**

Name	Type	Description
mode	u8	scale range: 0: 0V ~ 5V 1: -5V ~ +5V (Default) 2: 0V ~ 10V 3: -10V ~ +10V 255 : AD stop operation.

### ● EX94421 AD value read

**Format :** u32 status = EX94421\_AD\_value\_read(u8 CardID,u8 channel,  
f32 \*voltage\_value)

**Purpose:** Read voltage value with pre-calibration data.

**Parameters:**

**Input:**

Name	Type	Description
CardID	u8	assigned by DIP/ROTARY SW
channel	u8	A/D channel number 0~7: EX94421, 8 channels AD

**Output:**

Name	Type	Description
voltage_value	f32	Voltage value based on the AD converted and calibrated data. Say if the AD scale range is set at 0~5V then the voltage value returned will be in the 0~5 range.

- **EX94421 AD data read**

**Format :** u32 status = EX94421\_AD\_data\_read(u8 CardID,u8 channel,  
u16 \*voltage\_data)

**Purpose:** Read voltage value with pre-calibration data.

**Parameters:**

**Input:**

Name	Type	Description
CardID	u8	assigned by DIP/ROTARY SW
channel	u8	A/D channel number 0~7: EX94421, 8 channels AD

**Output:**

Name	Type	Description
voltage_data	u16	Voltage value based on the AD converted data. If unipolar voltage: 0~5V or 0~10V is 0~65535 If bipolar voltage, please take the data as 2's complement, change to i16 first then -5V~+5V or -10V~+10V is -32768 ~ +32767

### ● EX94421 AD integral start

**Format :** u32 status = EX94421\_AD\_integral\_start(u8 CardID,u8 mode)

**Purpose:** start AD conversion with integral constant.

**Parameters:**

**Input:**

Name	Type	Description
CardID	u8	assigned by jumper setting
mode	u8	0: immediately access, no integration 1: integration time 100ms 2: integration time 200ms 3: integration time 300ms 4: integration time 400ms 5: integration time 500ms 6: integration time 600ms 7: integration time 700ms 8: integration time 800ms 9: integration time 900ms 10: integration time 1s

### ● EX94421 AD integral all read

**Format :** u32 status = EX94421\_AD\_integral\_all\_read(u8 CardID,i16 data[8])

**Purpose:** read one port integral result of AD conversion data.

**Parameters:**

**Input:**

Name	Type	Description
CardID	u8	assigned by jumper setting

**Output:**

Name	Type	Description
data[8]	u16	data[0]: Channel 0 AD data ... data[7]: Channel 7AD data If unipolar voltage: 0~5V or 0~10V is 0~65535 If bipolar voltage, please take the data as 2's complement, change to i16 first then -5V~+5V or -10V~+10V is -32768 ~ +32767

**Note:**

To read all channels in integral

Start integral mode by EX94421\_AD\_integral\_start.

Read all channels by **EX94421\_AD\_integral\_all\_read**.

Stop AD integration function by **EX94421\_AD\_integral\_stop**.

- **EX94421 AD integral stop**

**Format :** u32 status = EX94421\_AD\_integral\_stop(u8 CardID)

**Purpose:** stop AD integral conversion.

**Parameters:**

**Input:**

Name	Type	Description
CardID	u8	assigned by jumper setting

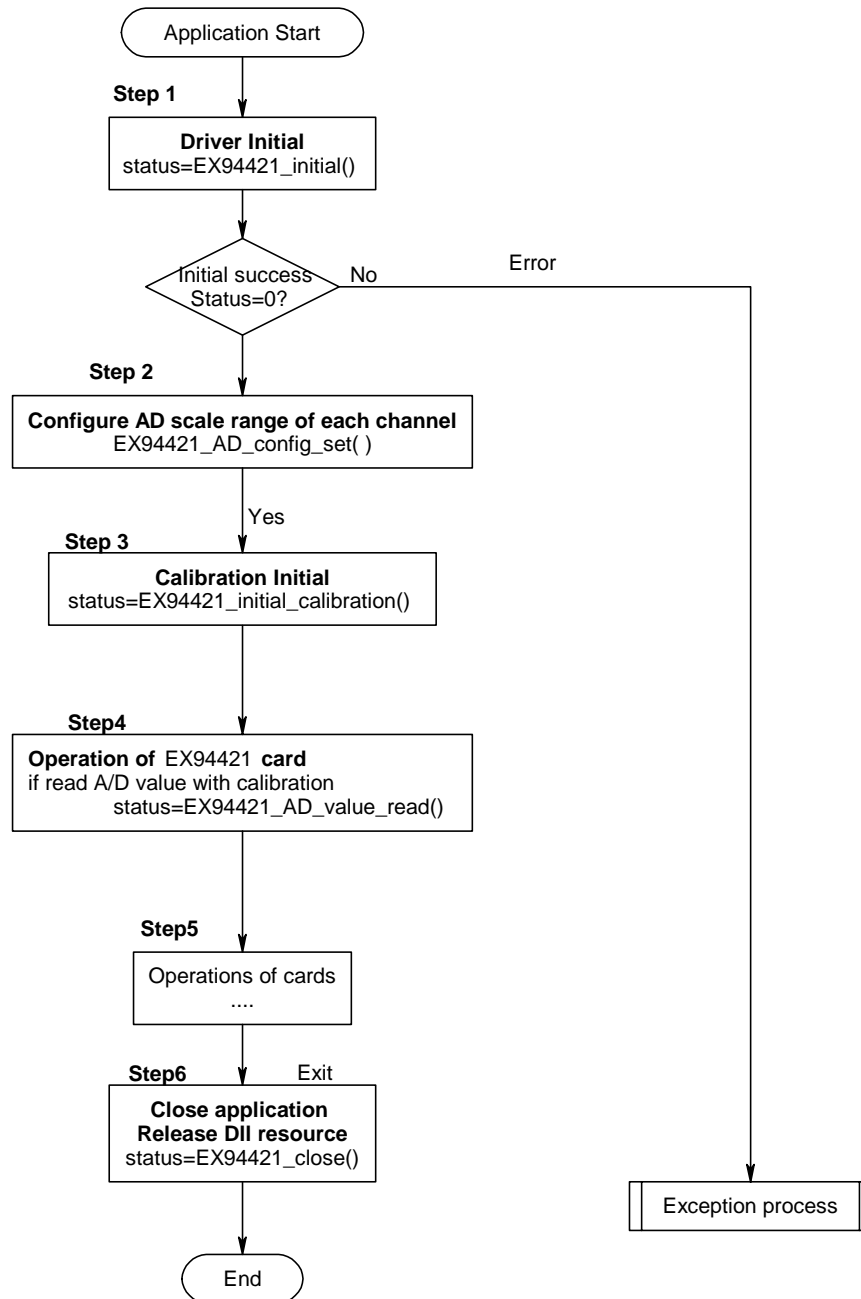


### 5.3 Error conditions

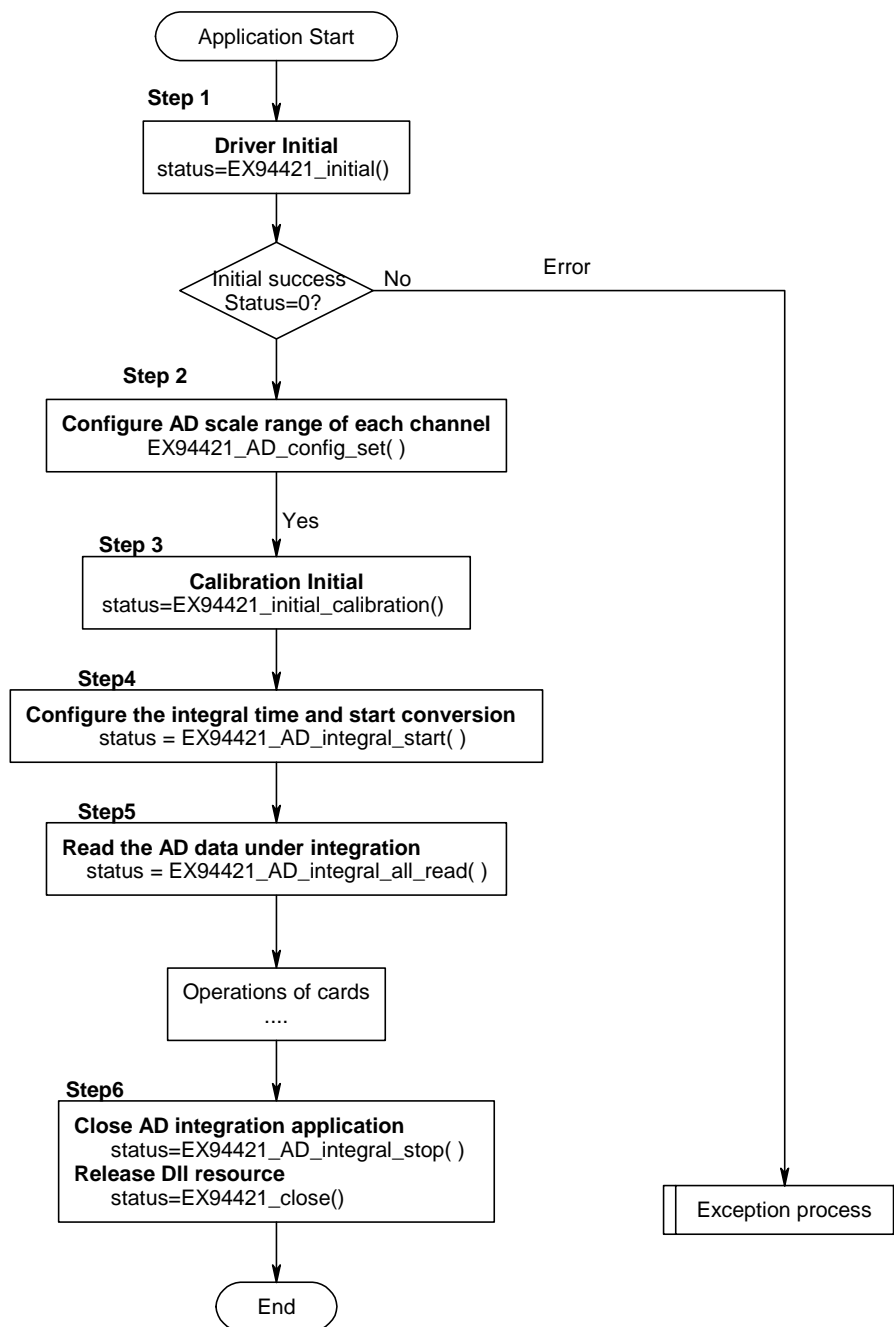
These error types may indicate an internal hardware problem on the board. Error Codes summary contains a detailed listing of the error status returned by EX94421 functions.

## 6. Flow chart of application implementation

### 6.1 EX94421 Flow chart of analog I/O application implementation



## 6.2 EX94421 Flow chart of analog I/O application with embedded integration function



## 7. Dll list

	Function Name	Description
1.	EX94421_initial( )	Card initial.
2.	EX94421_close( )	Card Close.
3.	EX94421_info( )	Read Card Address.
4.	EX94421_AD_config_set( )	Set AD config.
5.	EX94421_AD_config_read( )	Read AD config.
6.	EX94421_AD_value_read( )	Read AD value.
7.	EX94421_AD_data_read( )	Read AD data value.
8.	EX94421_AD_integral_start( )	start AD conversion with integral constant
9.	EX94421_AD_integral_all_read( )	read port integral result of AD conversion data
10.	EX94421_AD_integral_stop( )	stop AD integral conversion

## 8. EX94421 Error codes summary

### 8.1 EX94421 Error codes table

Error Code	Symbolic Name	Description
0	DRV_NO_ERROR	Success, No error.
2	DRV_INIT_ERROR	Driver initial error
3	DRV_UNLOCK_ERROR	Security unlock failure
4	DRV_LOCK_COUNTER_ERROR	Dead lock, unlock failure more than 10 times
5	SDRV_SET_SECURITY_ERROR	Password overwrite error
100	DEVICE_RW_ERROR	Device Read/Write error
101	DRV_NO_CARD	No EX94421 card on the system.
102	DRV_DUPLICATE_ID	EX94421 CardID duplicate error.
104	DRV_PAR_ERROR	Bad parameter or illegal parameter
300	DIO_ID_ERROR	Function input parameter error. CardID setting error, CardID doesn't match the DIP/ROTARY SW setting
301	AIO_MODE_ERROR	Mode parameter error. Parameter out of range.
302	AIO_CHANNEL_ERROR	Channel parameter error. Parameter out of range.
305	AIO_CONVERSION_ERROR	Conversion time over. Maybe no hardware or bad hardware.
306	AIO_CONVERSION_BUSY	A/D is busy in conversion
400	AIO_PORT_ERROR	Port parameter error. Parameter out of range.
401	AIO_STATE_ERROR	State parameter error. Parameter out of range.
402	AIO_POINT_ERROR	Point parameter error. Parameter out of range.