KEYENCE
GT2DLEP1 Provider

Version 1.0.1

User’s guide

July 11, 2017

Remarks:

This document uses the machine translation.

[Revision history]

Version	Date	Description
1.0.0	2017-3-16	First edition.
1.0.1	2017-7-11	Did error corrections. @PeekValue -> @PeakValue @JadgeDelay -> @JudgeDelay

[Operation confirmation model]

Model	Version	Note
GT2 Series		Configuration of Communication unit (DL-EP1) + Amplifier unit (GT2-71N etc.) + Sensor unit (GT2-H12etc.)

Contents

1. Introduction

KEYENCE GT2DLEP1 Provider (hereafter this is called GT2DLEP1 Provider) is ORiN2 CAO Provider that can access with the communication unit (DL-EP1) to a Contact displacement sensor (GT2 Series) made by KEYENCE.

This document describes the overview of GT2DLEP1 provider and implemented CAO interface (function specifications).

2. Outline of provider

2.1. Installation

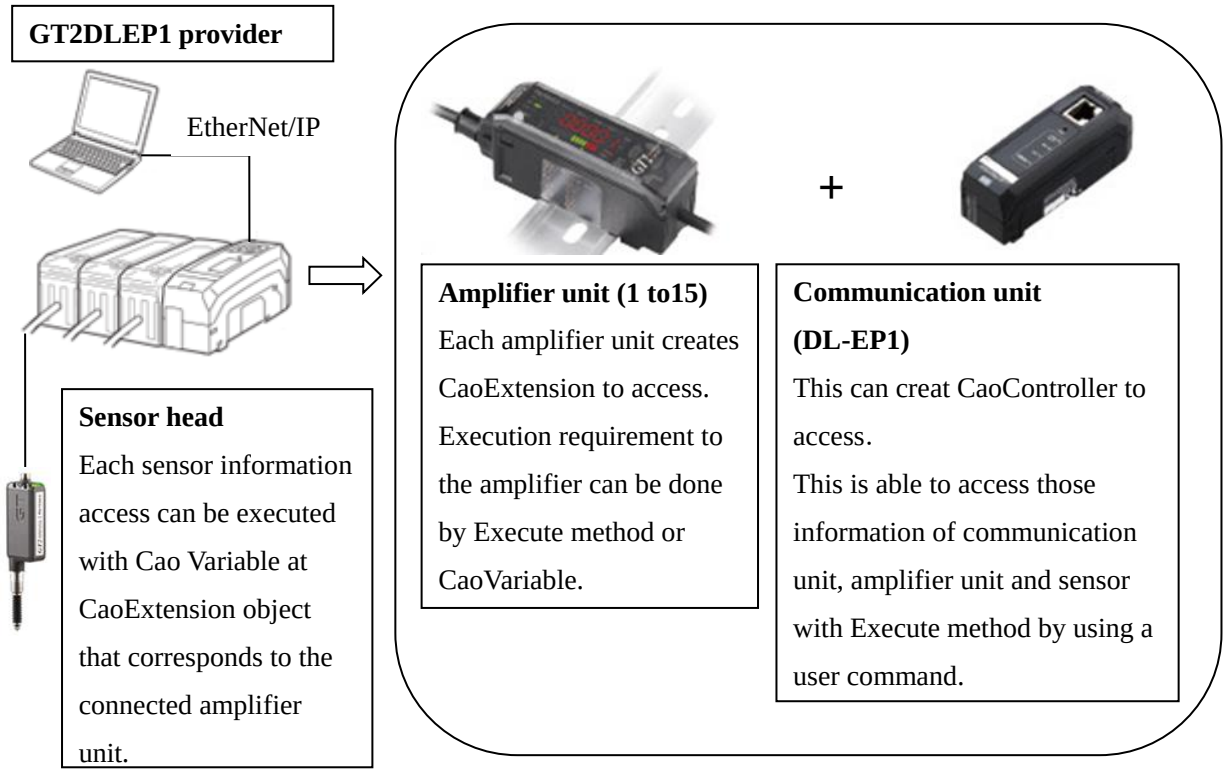
GT2DLEP1 provider module consists of the following DLL. You do not need to install it manually if it is installed by ORiN2 SDK. To install manually, use information of Table 2-1.

Table 2-1 GT2DLEP1 provider

File name	CaoProvKEYENCEGT2DLEP1.dll
ProgID	CaoProv.KEYENCE.GT2DLEP1
Registration	regsvr32 CaoProvKEYENCEGT2DLEP1.dll
Deregistration	regsvr32/u CaoProvKEYENCEGT2DLEP1.dll

2.2. Outline

GT2DLEP1 provider accesses to DE-EP1 by using CIP protocol.
This provider creates a CaoExtension object per Amplifier unit, connected to DE-EP1 and can access to the information of the sensor by Execute method or by reading/writing to CaoVariable.



3. Method and Properties

3.1. CaoWorkspace::AddController method

GT2DLEP1 provider establishes a connection by referring to the communication connection parameter at the time of AddController.

Syntax AddController(<bstrCtrlName:BSTR>,<bstrProvName:BSTR>,
<bstrPCName:BSTR>,<bstrOption:BSTR>))

bstrCtrlName : [in] Controller name
bstrProvName : [in] Provider name. Fixed to " CaoProv.KEYENCE.GT2DLEP1"
bstrPcName : [in] Computer name where provider runs
bstrOption : [in] Option strings

The following table shows a list of option character strings.

Table 3-1 Option character string of CaoWorkspace: : AddController

Option ([1])	Description
Conn=< Connection parameter>	Required. Communication format and connection parameter. (See 3.1.1)
MyIP[=<Self IP address>]	Self IP address (for several NICs) (Default : not specified)
ConnTimeout [=<Timeout period>]	Timeout period for TCP connection. (Millisecond) (Default : 3000)
Timeout[=<Timeout period>]	Timeout period at the receiving and sending. (Millisecond) (Default : 3000)

3.1.1. Conn option

Specify the communication format and the connection parameters.

The following shows connection parameter strings for Conn option.

Ethernet/IP device

"Conn=ETH:<Dest IP Address>[:<Dest Port No>[:<Src IP Address>[:<Src Port No>]]]"

< Dest IP Address > : TCP/IP connection destination IP address.
Example : "127.0.0.1", "192.168.0.1"

<Dest Port No> : TCP/IP connection port number.

[1] Items enclosed with square brackets ("[]") are omittable. Underlined part shows the default value when the option is not specified.

Example : 44818, 5006, 5007

<Src IP Address> : Self IP address. (for several NICs) ([2])

<Src Port No> : Self port number. (for several NICs)

3.2. CaoControllere::Execute method

Execute method of CaoController class executes a command. Specify a command name for the first argument and command parameter for the second argument. For details and usage of each command, refer to Chapter 4.1.

Syntax [<vntRet:VARIANT> =] Execute(<bstrCmd:BSTR > [,<vntParam:VARIANT>])

< bstrCmd> : [in] Command name

< vntParam> : [in] Parameter

3.3. CaoController::AddVariable method

AddVariable method of CaoController class is a method to create a variable object. For a list of variables available, see Chapter 5.1.

Syntax AddVariable(<bstrVariableName:VT_BSTR>[,<bstrOption:VT_BSTR>])

<bstrVariableName> : [in] Variable name

<bstrOption> : [in] Option string

[2] An error occurs if self IP address is specified both in the Conn option and in the MyIP option. When you specify an IP address, be sure to specify it in one option only.

3.4. CaoController::AddExtension method

AddExtension method of CaoController class is a method to create an Extension object per amplifier unit that is connected to DL-EP1.

Syntax AddExtension(<bstrChannelNo:VT_BSTR>)

< bstrChannelNo>	:	[in] ID number of amplifier unit. Specify it with " CH<ID number >" (ID number: 1to15)

3.5. CaoExtension::Execute method

Execute method of CaoExtension class executes a command. Specify a command name for the first argument and parameter for the second argument. For details and usage of each command, refer to Chapter 4.2.

Syntax [<vntRet:VARIANT> =] Execute(<bstrCmd:BSTR > [,<vntParam:VARIANT>])

< bstrCmd>	:	[in] Command name
< vntParam>	:	[in] Parameter

3.6. CaoExtension::AddVariable method

AddVariable method of CaoExtension class is a method to create a variable object. For a list of variables available, see Chapter 5.2.

Syntax AddVariable(<bstrVariableName:VT_BSTR>[,<bstrOption:VT_BSTR>])

<bstrVariableName>	:	[in] Variable name
<bstrOption>	:	[in] Option string

3.7. CaoController::get_ExtensionNames property

Obtain an Extension name list described in.

3.8. CaoController::get_VariableNames property

Obtain a variable name list described in.

3.9. CaoExtension::get_VariableNames property

Obtain a variable name list described in.

3.10. CaoVariable::put_ID property

Set an ID of CaoVariable. Setting value is set by the arguments specified.

This can be used to specify the ID when accessing to some variables.

3.11. CaoVariable::get_ID property

Obtain an ID of CaoVariable.

3.12. CaoVariable::put_Value property

Set the specified data by the variable name. Setting value is set by the arguments specified.

3.13. CaoVariable::get_Value property

Obtain the specified data by the variable name.

4. Command list

4.1. CaoController class

Table 4-1 Command list of CaoController class

Command name	Function	
Raw	Send a user command.	P11

4.1.1. CaoController::Execute("Raw") command

Execute sending a user command.

Syntax Raw (<vntDataArray>)

<vntDataArray> : [in] Command data
(VT_UI1 | VT_ARRAY)
[0] Service code
[1] Class ID
[2] Instance ID
[3] Attribute ID(Low)
[4] Attribute ID(High)
[n] Service data

- When Attribute ID is 8 Bit, specify 0x00 to High side.
- Later of element 5 can be used as a service data.
- Specify the service data per 8Bit from Low side.
- When the service data is odd-Byte, High side gets NULL (0x00) padding internally.

Return value : [out] Response data
(VT_UI1 | VT_ARRAY)

- No response data will be returned by VT_EMPTY.
- Response data will be stored per 8 Bit from Low side.

4.2. CaoExtention class

Table 4-2 Command list of CaoExtention class

Command name	Function	
Preset	Execute Preset.	P12
PresetReset	Execute PresetReset.	P12
Reset	Execute Reset.	P12
InitReset	Execute Initial Reset.	P13
ErrClear	Execute Error Clear.	P13

4.2.1. CaoExtention::Execute(“Preset”) command

Execute Preset execution requirement.

Syntax Preset ()

Argument : None

Return value : None

4.2.2. CaoExtention::Execute(“PresetReset”) command

Execute PresetReset execution requirement.

Syntax PresetResett ()

Argument : None

Return value : None

4.2.3. CaoExtention::Execute(“Reset”) command

Execute Reset execution requirement.

Syntax Reset ()

Argument : None

Return value : None

4.2.4. CaoExtention::Execute(“InitReset”) command

Execute InitReset execution requirement.

Syntax InitReset ()

Argument : None

Return value : None

4.2.5. CaoExtention::Execute(“ErrClear”) command

Execute ErrClear execution requirement.

Syntax ErrClear ()

Argument : None

Return value : None

5. Variable list

5.1. CaoController class

5.1.1. System variable

Table 5-1 CaoController class system variable list

Variable name	Data type	Description	Attribute	
			get	put
@MAKER_NAME	VT_BSTR	KEYENCE is returned.	✓	—
@VERSION	VT_BSTR	Return the provider version information.	✓	—
@LAST_RESPONSE_STATUS	VT_I4 | VT_ARRAY	Detailed information of the latest error report. (VT_I4 | VT_ARRAY) [0] General status [1] Additional status (0 is set when nothing to be added)	✓	—

5.1.2. User variable

Table 5-2 CaoController class user variable list

Variable name	Data type	Description	Attribute	
			get	put
None			—	—

5.2. CaoExtention class

5.2.1. System variable

Table 5-3 CaoExtention class system variable list

Variable name	Data type	Description	Attribute	
			get	put
@CriteriaValue	VT_R8	Criteria value (P.V. value) -199.9999 to 199.9999 (the initial value: 0.0000)	✓	—
@CalcValue	VT_R8	Calculation value -199.9999 to 199.9999 (the initial value: 0.0000) ※ If calculation function is not used, errors of reading will occur.	✓	—
@RawValue	VT_R8	Raw value (R.V.value) -199.9999 to 199.9999(the initial value: 0.0000)	✓	—
@PeakValue	VT_R8	Peak hold value in P-P mode -199.9999 to 199.9999 (the initial value: 0.0000) ※ If detecting mode is other than peak to peak mode, errors of reading will occur.	✓	—
@BottomValue	VT_R8	Bottom hold value in P-P mode -199.9999 to 199.9999 (the initial value: 0.0000) ※ If detecting mode is other than peak to peak mode, errors of reading will occur.	✓	—
@ControlOutput	VT_I4	Control output 0x00 to 0x1F (the initial value: 0x00) bit0 : HIGH output bit1 : LOW output bit2 : GO output bit3 : HH output bit4 : LL output	✓	—

@ErrStatu	VT_I4	Sensor amplifier with errors 0x00 to 0xFF bit0 : Error of Over current (ErC) bit1 : Error of Head (ErH) bit2 : Error of EEPROM (ErE) bit3 : Error of checking Hook (Er.chK) bit4 : Self-timing delay (Er.dLY) bit5 : Error of Unit number (Er.Unit) bit6 : Error of Calculation (Er.CAL) bit7 : Error of dedicated calculation mode (Er.noH)	✓	—
@CalcRawValue	VT_R8	R.V.value which was the base of the calculation result. -199.9999 to 199.9999 (the initial value: 0.0000) ※ Specify an ID device with ID properties (ID= 1 to 15) ※ If calculation function is not used, errors of reading will occur	✓	—
@Preset	VT_I4	Preset execution requirement R 0 to2 : Value written lastly W 0→1 : Execute Preset 0→2 : Execute PresetReset	✓	✓
@Reset	VT_I4	Reset execution requirement R 0to1 : Value written lastly W 0→1 : Execute Reset	✓	✓
@InitReset	VT_I4	Initial Reset execution requirement R 0to1 : Value written lastly W 0→1 : Execute Initial Reset	✓	✓
@ErrClear	VT_I4	Error Clear execution requirement R 0to1 : Value written lastly W 0→1 : Execute Error Clear	✓	✓
@Bank	VT_I4	In the state of Bank selecting 0 to 3 (the initial value: 0)	✓	✓

@Timing	VT_I4	In the state of Timing 0 to 1 (the initial value: 0) Entering Timing: OFF R 0 : Or Now measuring by self-timing Entering Timing: ON 1 : Or Now not measuring by self-timing W 0 : Change to now measuring 1 : Change to now not measuring	✓	✓
@KeyLock	VT_I4	Key Lock 0 to 2 (the initial value: 0) 0 : Unlock 1 : Full Key Lock 2 : Key Lock	✓	✓
@BarDisp	VT_I4	Bar display mode 0 to 1 (the initial value: 0) 0 : Bar display mode 1 : OK/NG display mode	✓	✓
@Bank_HH	VT_R8	Setting value of HH side of Bank* -199.9999 to 199.9999 (the initial value: 7.0000) ※ Specify a Bank No. with ID properties (ID= 0 to 3)	✓	✓
@Bank_HIGH	VT_R8	Setting value of High side of Bank* -199.9999 to 199.9999 (the initial value: 5.0000) ※ Specify a Bank No. with ID properties (ID= 0 to 3)	✓	✓
@Bank_LOW	VT_R8	Setting value of Low side of Bank* -199.9999 to 199.9999 (the initial value: 1.0000) ※ Specify a Bank No. with ID properties (ID= 0 to 3)	✓	✓
@Bank_LL	VT_R8	Setting value of LL side of Bank* -199.9999 to 199.9999 (the initial value: -1.0000) ※ Specify a Bank No. with ID properties (ID= 0 to 3)	✓	✓
@Bank_Preset	VT_R8	Setting value of Preset side of Bank* -199.9999 to 199.9999 (the initial value: 0.0000) ※ Specify a Bank No. with ID properties (ID= 0 to 3)	✓	✓

@CalcMode	VT_I4	Calculation mode/ Calculation setting 0 to 27 (the initial value: 0) Tens place (Calculation mode) 0 : Do not use calculation function 1 : Use calculation function 2 : Dedicated calculation mode Ones place (Calculation setting) 0 : Maximum value 1 : Minimum value 2 : Flatness 3 : Average 4 : Reference difference 5 : Twist 6 : Warpage 7 : Thickness	✓	✓
@DetectingMode	VT_I4	Detecting mode 0 to 4 (the initial value: 0) 0 : Standard 1 : NG hold 2 : Peak hold 3 : Bottom hold 4 : Peak to peak	✓	✓
@HoldUpdate	VT_I4	Hold Update 0 to 1 (the initial value: 0) 0 : Timing 1 : Regular updated	✓	✓

@ResponseTime	VT_I4	Response time 0 to 5 (the initial value: 3) If the connected sensor head is GT2-H***, A*** 0 : HSP(3.0ms) 1 : 5ms 2 : 10ms 3 : 100ms 4 : 500ms 5 : 1000ms If the connected sensor head is GT2-P*** 0 : HSP(12ms) 1 : 20ms 2 : 40ms 3 : 400ms 4 : 2000ms 5 : 4000ms	✓	✓
@TimingCategory	VT_I4	Timing type 0 to 2 (the initial value: 0) 0 : External timing input 1 : Rising edge self-timing 2 : Falling edge self-timing	✓	✓
@TimingLevel	VT_R8	Self-timing level -199.9999 to 199.9999 (the initial value: 0.5000)	✓	✓
@TimingDelay	VT_I4	Self-timing delay type 0 to 1 (the initial value: 0) 0 : Static hold 1 : Delay timer	✓	✓
@DelayTime	VT_I4	Specified delay time by user 0 to 9999 (the initial value: 1000)	✓	✓
@JudgeDelay	VT_I4	Indicates the static hold delay stable determination. 0 to 1 (the initial value: 0) 0 : Default 1 : User	✓	✓
@HoldRange	VT_R8	Indicates the static hold delay stable range. 0.0000 to 199.9999 (the initial value: 0.0100)	✓	✓

@MeasureDirect	VT_I4	Measurement in/decrease direction 0 to 1 (the initial value: 0) 0 : Normal 1 : Invert	✓	✓
@LeverRatio	VT_R8	Lever ratio 0.1 to 100.0 (the initial value: 1.0)	✓	✓
@Output	VT_I4	Output style 0 to 1 (the initial value: 0) 0 : N.O. 1 : N.C.	✓	✓
@DisplayDigit	VT_I4	Display digit number 0 to 3 (the initial value: 0) 0 : 0.0001 1 : 0.001 2 : 0.01 3 : ___0.1	✓	✓
@Hysteresis	VT_R8	Hysteresis 0.0000 to 199.9999 (the initial value: 0.0030)	✓	✓
@AllInput	VT_I4	Setting of all input 0 to 1 (the initial value: 0) 0 : Individual input 1 : Batch input	✓	✓
@SpecialOutput	VT_I4	Setting of special output 0 to 5 (the initial value: 0) 0 : Not used 1 : 5-output 2 : Limit output 3 : Limit output: user setting 4 : All GO 5 : All limit output	✓	✓
@Limit_HH	VT_R8	Limit output: setting of judgment point at HH side -199.9999 to 199.9999 (the initial value: 0.5000)	✓	✓
@Limit_LL	VT_R8	Limit output: setting of judgment point at LL side -199.9999 to 199.9999 (the initial value: 0.5000)	✓	✓

@PresetData	VT_I4	Select Preset Data 0 to 1 (the initial value: 0) 0 : R.V. value 1 : P.V. value	✓	✓
@PresetMem	VT_I4	Preset Memory 0 to 1 (the initial value: 0) 0 : YES 1 : NO	✓	✓
@PresetPoint	VT_I4	Preset Point 0 to 1 (the initial value: 0) 0 : Common to all the Banks 1 : Savr for each Bank	✓	✓
@EcoMode	VT_I4	Power-saving function (ECO) 0 to 2 (the initial value: 0) 0 : OFF 1 : HALF 2 : ALL	✓	✓
@CheckHook	VT_I4	Function of checking Hook 0 to 2 (the initial value: 0) 0 : OFF 1 : ON 2 : USER	✓	✓
@CheckHookPos	VT_R8	Motion point of checking Hook -199.9999 to 199.9999 (the initial value: 0.5000)	✓	✓
@BunchSetting	VT_I4	Bunch settings 0 to 1 (the initial value: 0) 0 : Individually 1 : At a time ※ This variable can be used by GT2-100 series only	✓	✓
@AnalogSetting	VT_I4	Analog range setting 0 to 1 (the initial value: 0) 0 : Default 1 : Free range setting ※ This variable can be used by GT2-71MC* only	✓	✓

@FreeRange_Hi	VT_R8	Free range setting (Hi side) -199.9999 to 199.9999 (the initial value: 12.0000) ※ This variable can be used by GT2-71MC* only	✓	✓
@FreeRange_Lo	VT_R8	Free range setting (Lo side) -199.9999 to 199.9999 (the initial value: 0.0000) ※ This variable can be used by GT2-71MC* only	✓	✓
@LAST_RESPONSE_STATUS	VT_I4 | VT_ARRAY	Detailed information of the latest error report. (VT_I4 | VT_ARRAY) [0] General status [1] Additional status (0 is set when nothing to be added)	✓	—

5.2.2. User variable

Table 5-4 CaoController class user variable list

Variable name	Data type	Description	Attribute	
			get	put
None			—	—

6. Error code

In GT2DLEP1 provider, the following original error codes are defined.

For ORiN2 common errors, refer to the error code section of "ORiN2 programming guide".

Table 6-1 Original error code

Error name	Error No.	Description
CIP response error (Format Error)	0x80100000	Return this error code if a response packet of CIP communication is unexpected abnormal format.
CIP response error (Error Status Code)	0x80100001	Return this error code if a response packet of CIP communication is abnormal status.
CIP response error (Error General Status)	0x80100002	Return this error code if a response packet of CIP communication is an error. When received this error, detailed error code can be obtained by reading out the variable system, @LAST_RESPONSE_STATUS.

Table 6-2 Detailed error code

General status	Additional status	Description
0x00	0x0000	Normal end.
0x05	0x0000	Instance ID is out of range.
0x08	0x0000	This service does not correspond to the specified Instance ID or cannot execute.
0x09	0x0000	Written data is out of range.
0x0C	0x0000	Motion command was failed to execute. Make sure that the sensor amplifier is able to execute the motion command.
0x0E	0x0000	・ You are trying to write to Attribute ID which is unable to be written in or which is not ready to be written in. ・ Written data is out of range.
0x10	0x0000	You are trying to read from Attribute ID which is unable to be read or which is not ready to be read.
0x13	0x0000	Service data size is smaller than the prescribed data. The prescribed data size will be stored as the additional status.
0x14	0x0000	Attribute ID is out of range.

0x16	0x0000	Sensor amplifier corresponding to the specified instance ID is not connected.
0x1F	0xC350	Although the specified ID is supported by this service, the specified attribute ID is not available.
	0xC351	This mode cannot write the specified setting value in the parameter.
	0xC352	Communication initialization of this robot is in progress.
0xFE	0x0000	System error.

7. Sample program

Sample code (C#) is shown below.

7.1. Connection and creating objects

```
// Connect to the communication unit and create an object which accesses to the amplifier
// unit or sensor head.
private void Form1_Load(object sender, EventArgs e)
        {
            try
            {
                // Create CAO engine
                this.eng = new CCaoEngine();

                // Connect to the communication unit
                this.ctrl = this.eng.Workspaces[0].AddController(
                                        "Sample",
                                        "CaoProv.KEYENCE.GT2DLEP1",
                                        string.Empty,
                                        "CONN=ETH:192.168.0.20");

                // Create Amplifier unit object
                this.ext = this.ctrl.AddExtension("CH1", string.Empty);

                // Create a variable object
                this.wVal = this.ext.AddVariable("@Bank_HIGH", string.Empty);
                this.rVal = this.ext.AddVariable("@CriteriaValue", string.Empty);
            }
            catch (Exception ex)
            {
                MessageBox.Show(this, ex.Message);
            }
        }
```

7.2. Reset Sensor ID=1

```
//  Reset Sensor ID=1
private void Command1_Click(object sender, EventArgs e)
        {
            try
            {
                // Require Reset
                this.ext.Execute("Reset", null);
            }
            catch (Exception ex)
            {
                MessageBox.Show(this, ex.Message);
            }
        }
```

7.3. Sensor ID=1: Set the setting value of High side at Bank 2

```
//  Set the bank No.
private void Command2_Click(object sender, EventArgs e)
        {
            try
            {
                // Set the bank No.
                this.wVal.ID = 2;

                // Set the value
                double val;
                double.TryParse(this.textBox1.Text, out val);
                this.wVal.Value = val;
            }
            catch (Exception ex)
            {
                MessageBox.Show(this, ex.Message);
            }
        }
```

7.4. Sensor ID=1: Obtain the criteria value (P.V. value)

```
//  Obtain Sensor ID=1 criteria value (P.V. value)
private void Command3_Click(object sender, EventArgs e)
        {
            try
            {
                // Obtain the value
                this.textBox2.Text = ((double)(this.rVal.Value)).ToString();
            }
            catch (Exception ex)
            {
                MessageBox.Show(this, ex.Message);
            }
        }
```

7.5. Sensor ID=1: Obtain the criteria value (Raw command)

```
private void Command4_Click(object sender, EventArgs e)
        {
            try
            {
                // Obtain Sensor ID=1 criteria value (P.V. value)
                byte[] param = new byte[5];
                param[0] = 0x0E;    // Service code        : Read parameter
                param[1] = 0x67;    // Class ID               : DL Object class
                param[2] = 0x01;    // Instance ID         : Sensor ID=1
                param[3] = 0x25;    // Attribute ID(Low)   : Criteria value (P.V. value)
(0x0325)
                param[4] = 0x03;    // Attribute ID(High)
                object val = this.ctrl.Execute("Raw", param);
                string str = string.Empty;
                foreach (byte b in (byte[])val)
                {
                    str += " ";
```

```
            str += b.ToString();
        }
        this.textBox3.Text = str;
    }

    catch (Exception ex)
    {
        MessageBox.Show(this, ex.Message);
    }
}
```

7.6. Obtain the communication unit status (Raw command)

```
private void Command5_Click(object sender, EventArgs e)
{
    try
    {
        // Obtain the communication unit status
        byte[] param = new byte[5];
        param[0] = 0x0E;    // Service code         : Read parameter
        param[1] = 0x67;    // Class ID              : DL Object class
        param[2] = 0x00;    // Instance ID         : Communication unit=0
        param[3] = 0x64;    // Attribute ID (Low)  : Status (0x64)
        param[4] = 0x00;    // Attribute ID (High)
        object val = this.ctrl.Execute("Raw", param);
        string str = string.Empty;
        foreach (byte b in (byte[])val)
        {
            str += " ";
            str += b.ToString();
        }
        this.textBox4.Text = str;
    }
    catch (Exception ex)
    {
        MessageBox.Show(this, ex.Message);
    }
}
```