KEYENCE

LK-G3000 providers

User's Guide

Version 1.0.1

December 21, 2021

NOTE:

[Revision History]

Version	Date	Description
1.0.0	2020-04-27	First edition
1.0.1	2021-12-21	Fixed false positives for errors.

[Operation Check Model]

Model	Version	Notes
LK-G3000	--	The dedicated cord OP-96368 (straight cord 2. 5m) and OP-26401(D-sub 9 pins must be used. Alternatively, connect OP-96369(D-sub 25 pins) to the PC and the device.

[Compatible models]

Model
LK-G3000
LK-G3000P
LK-G3000V
LK-G3000PV

Contents

1. Introduction

This manual is the user's guide for KEYENCE's laser displacement meter LK-G3000 series. Figure 1-1 shows the overall configuration of this provider and the device. LK-G3000 provider is referred to as KEYENCE laser displacement gauge LK-G3000 series. LK-G3000 provider uses RS-232C to perform serial communication.

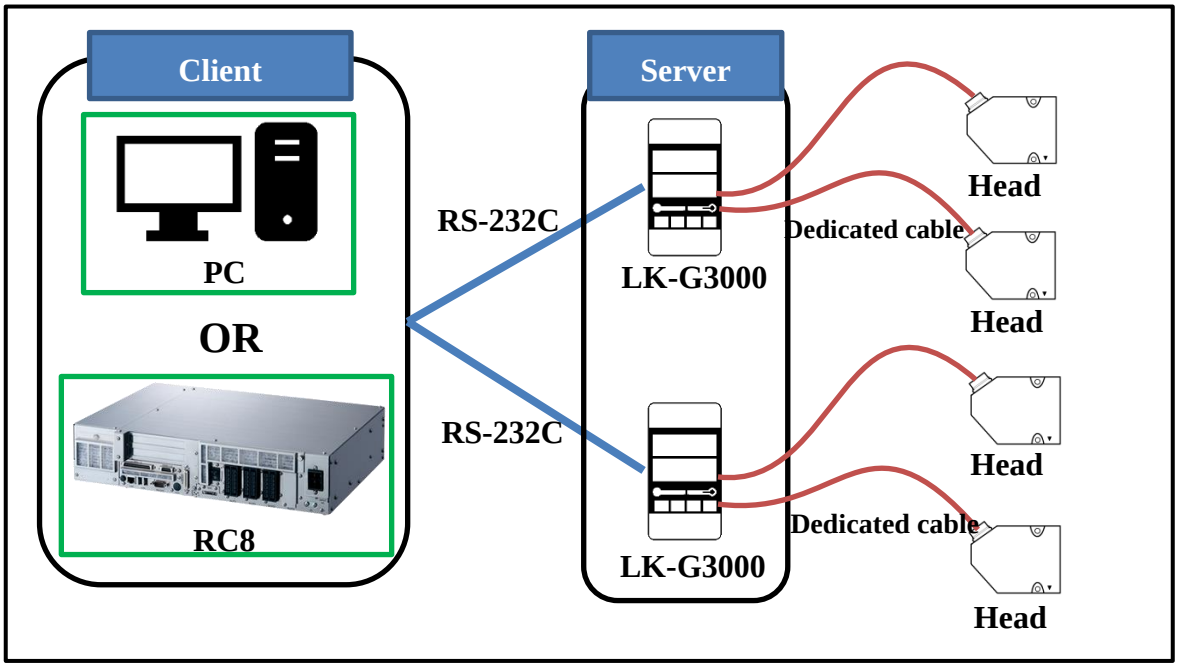


Figure 1-1 Configuration Diagram

Figure 1-2 shows the correspondence between this provider and each device.
(※An example. It does not represent everything.)

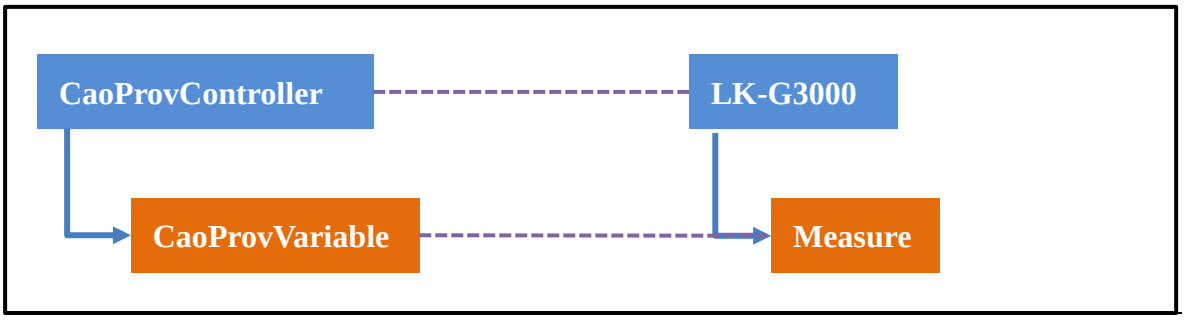


Figure 1-2 Provider configuration and device information

1.1. Informative sources

LK-G3000LkIF providers have been developed with reference to KEYENCE's User's Manual for High-Speed, High-Precision CCD Laser Displacement Meter LK-G Series 96M12274. This manual is referred to as LK-G3000 manual.

2. Setting Up Your Environment for Application Development

2.1. Connecting LK-G3000 to Client-PC

LK-G3000 providers connect using RS-232C communication with LK-G3000. Connect the dedicated cable (OP-96368 (straight cord 2. 5m) and OP-26401(D-sub 9 pin) or OP-96369(D-sub 25 pin) that is connected to the client PC to LK-G3000. Insert the dedicated cable into the slot in the red frame as shown in Figure 2-1 until it clicks into place.

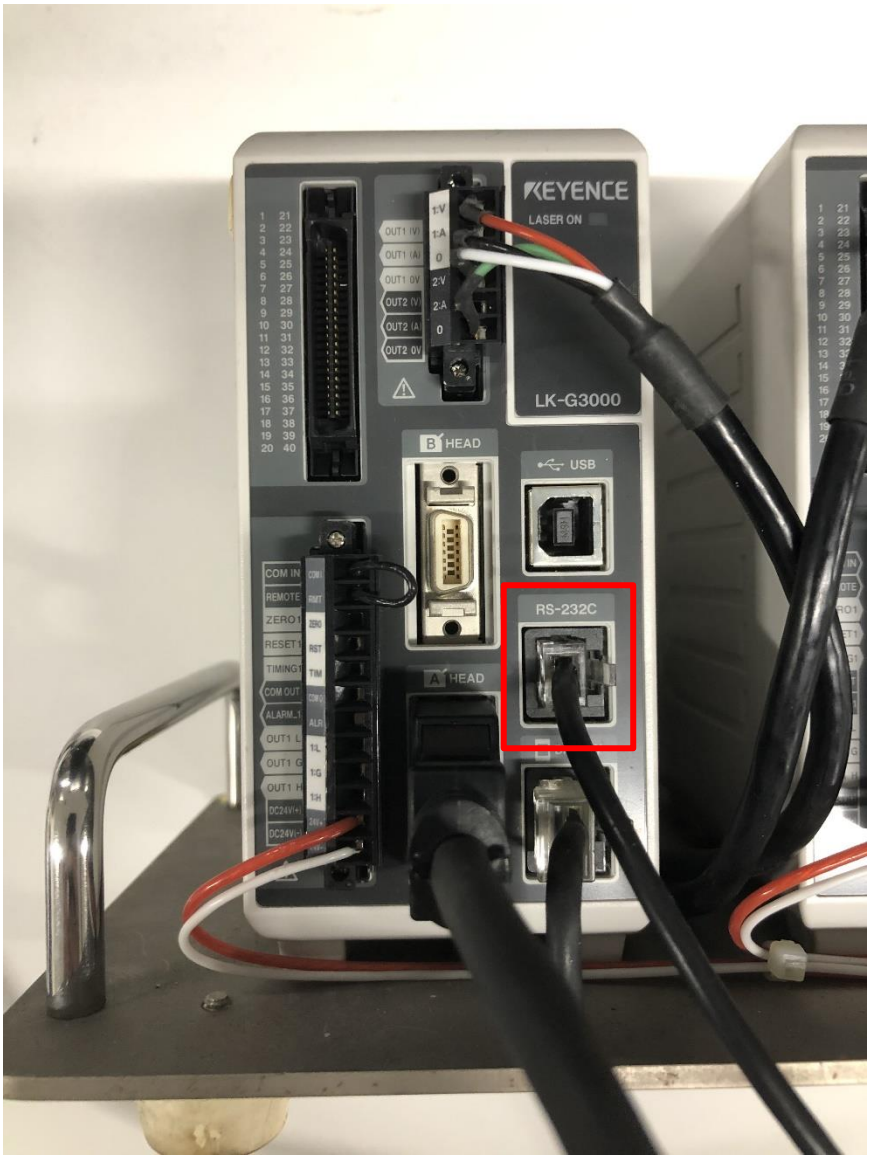


Figure 2-1 Actual LK-G3000 device

3. Command Reference

3.1. Method/Property List

Table 3-1 List of methods and properties

Category	Methods/Properties[1]		Function	See Also
CaoWorkspace				
	AddController	M	Connected to controller	P.9
CaoController				
	VariableNames	M	Get a list of variable names that can be connected	P.11
	Variables	P	Retrieving Variable Collections Holded by the Controller	P.11
	AddVariable	M	Adding Variable Objects	P.11
	Execute	M	Execute Extended Commands	P.11
CaoVariable				
	Value	P	Get/set value	P.12

3.2. Method properties

3.2.1. CaoWorkspace classes

3.2.1.1. AddController method

In CaoWorkspace, add a controller object. LK-G3000 providers connect to the appropriate LK-G3000 by referring to the parameters passed when AddController method is executed. The following are the specifics of AddController method:

SYNOPSIS

AddController

```
(
        "<controller name>",            // Controller name (optional)
        "CaoProv.KEYENCE.LK-G3000",     // Provider name (fixed)
        "<machine name>",               // Provider execution machine name (unused)
        "<Option>"                      // Option character string (mandatory)
)
```

Option

[1] M: Indicates methods, P: properties, and E: events, respectively.

The following is an optional specification for Option character string: Option character string is a comma (,) string consisting of the options listed below.

Option	Required	Description	Value Range	Default Value
Conn=	✓	Specifies the serial connection options for connecting to LK-G3000. Refer to 3.2.1.1.1 for details.	--	--
Timeout=	--	Specifies the number of milliseconds to wait for a response from sending a command to receiving data from LK-G3000.	1 - 4294967265	500

Usage example

```
Dim caoEng As CaoEngine    ' Engine
Dim caoWs As CaoWorkspace          ' Workspace
Dim caoCtrl As CaoController       ' Controller

' Generate CaoEngine
Set caoEng = new CaoEngine
' Generate CaoWorkspace
Set caoWs = caoEng.Workspaces.Item(0)
' Generate CaoController
Set caoCtrl = caoWs.AddController("LKG3000", _
                                  "CaoProv.KEYENCE.LK-G3000", _
                                  "", _
                                  "Conn=COM:1,Timeout=1000")
```

3.2.1.1.1. Conn Optional

The following is a Conn optional connection parameter string: Here, brackets ("[]") are optional, and the underlined part in the description of each parameter indicates the default value when no options are specified.

RS-232C

"Conn=COM:<COM Port>[:BaudRate>[:<Parity>:<DataBits>:<StopBits>[:Flow]]]"

<COM Port> : COM port number . '1' -COM1, '2' - COM2, ..

<BaudRate> : Baud rate. 9600, 19200, 38400, 57600, 115200

<Parity> : Parity . 'N'-NONE, 'E'-EVEN, 'O'-ODD

<DataBits> : Number of data bits. '7'-7bit, '8'-8bit

<StopBits> : No. of Stop Bits . '1'-1bit, '2'-2bit

<Flow> : Flow control. '0'-None, '1'-Xon/Xoff, '2'-Hardware control. It can be specified by taking OR.

3.2.2. CaoController classes

3.2.2.1. VariableNames method

Gets a list of variable names that can be connected. The variable name obtained by this method can be used as the first argument of AddVariable method described later.AddVariable method

Usage example

```
' Get variable name list
Dim variableNames As Variant
VariableNames = caoCtrl.variableNames
```

3.2.2.2. Variables Properties

Gets a collection of variables that the controller holds.

Usage example

```
' Variable Collection Retrieval
Dim variables As CaoVariables
Set variables = caoCtrl.variables
' Variable acquisition
Dim variable As CaoVariable
Set variable = variables.Item(0)
```

3.2.2.3. AddVariable method

Adds a variable object to CaoController. Only those listed in 3.4. Variable List can be used for variable names. AddVariable is specifi ed as follows.

SYNOPSIS

AddVariable

```
(
        "<variable name>",          // Variable name
        "<Option>"                  // Option character string (optional)
)
```

3.2.2.4. Execute method

Execute CaoController extended command. The extended commands that can be specified with Execute can be used only as shown in 3.3. Extended Commands List. Execute is specifi ed as follows.

SYNOPSIS

Execute

```
(
        "<extension command name>",             // Extended command name
        "<Option string>"           // Option character string (optional)
```

)

3.2.3. CaoVariable classes

3.2.3.1. Value Properties

Gets/sets the data from the connected LK-G3000. The behavior depends on the variable name. For details, refer to section 3.4, Variable List.

3.3. Extended command list

Defines the list of extended commands that can be used. The usage examples are described in detail for each command.

Table 3-2 List of extended commands

Command	Description	See Also
Mode change command		
SetMode	Sets the operation mode of the main unit.	P.14
Measurement and Control-Related Commands		
GetCalcData	Gets the measured value.	P.15
SetTiming	Sets ON/OFF of the timing.	P.16
SetZero	Sets the autozero ON/OFF.	P.17
SetReset	Set the reset.	P.17
SetPanelLock	Sets the panel lock.	P.17
SetProgramNo	Switches the program number.	P.18
GetProgramNo	Gets the program number.	P.18
GetFigureData	Get statistics.	P.18
ClearFigureData	Clears the statistics.	P.20
StartDataStorage	Starts data storage accumulation.	P.20
StopDataStorage	Stop accumulating data storage.	P.20
ClearDataStorage	Clears the accumulated data in the data storage.	P.21
GetDataStorageData	Retrieves accumulated data for data storage.	P.21
GetDataStorageStatus	Acquires the accumulated state of data storage.	P.21
Configuration-Related Commands		
Panel Display-Related Commands		

Command		Description	See Also
	SetPanel	Switches the panel display.	P.22
	GetPanel	Gets the panel display.	P.22
Commands Related to Tolerance Setting			
	SetTolerance	Sets the tolerance.	P.22
	GetTolerance	Gets the tolerance.	P.23
Head setting-related commands			
	SetAbleMode	Sets ABLE tuning mode.	P.24
	GetAbleMode	Gets ABLE tuning mode.	P.25
	SetAbleMinMax	Sets ABLE control area.	P.25
	GetAbleMinMax	Gets ABLE control scope.	P.25
	SetMeasureMode	Set the measurement mode.	P.26
	GetMeasureMode	Used to acquire the measurement mode.	P.27
	SetNumAlarm	Set the number of alarm processing times.	P.27
	SetAlarmLevel	Sets the alarm level.	P.28
	GetAlarm	Get alarm information.	P.28
	StartABLE	Starts ABLE tuning.	P.29
	StopABLE	Finish ABLE tuning.	P.30
	CancelABLE	Cancels ABLE tuning.	P.30
	SetReflectionMode	Sets the installation mode.	P.30
	GetReflectionMode	Gets the installation mode.	P.31
OUT setting-related commands			
	SetCalcMethod	Sets the calculation method.	P.31
	GetCalcMethod	Gets the calculation method.	P.35
	SetScaling	Set the scaling.	P.36
	GetScaling	Get the scaling.	P.37
	SetFilter	Configure the filter.	P.37
	GetFilter	Get the filter.	P.39
	SetTriggerMode	Sets or queries the trigger mode.	P.39
	GetTriggerMode	Gets the trigger mode.	P.40

Command		Description	See Also
	SetOffset	Set the offset.	P.40
	GetOffset	Gets the offset.	P.41
	SetAnalogScaling	Sets the analog output scaling.	P.41
	GetAnalogScaling	Obtains the analog output scaling.	P.42
	SetCalcMode	Sets the measurement mode.	P.43
	GetCalcMode	Gets the measurement mode.	P.44
	SetDisplayUnit	Set the minimum display unit.	P.44
	GetDisplayUnit	Gets the smallest display unit.	P.44
	SetAnalogThrough	Sets the analog slew.	P.45
	GetAnalogThrough	Acquires an analog through.	P.45
Common setting related commands			
	SetDataStorage	Set the target OUT, number of accumulated points, and accumulation period for data storage.	P.46
	GetDataStorage	Retrieves the target OUT, number of accumulated points, and accumulation period for data storage.	P.47
	SetSamplingCycle	Sets the sampling period.	P.47
	GetSamplingCycle	Acquires the sampling period.	P.48
	SetMutualInterPrev	Set mutual interference prevention.	P.48
	GetMutualInterPrev	Acquires mutual interference prevention.	P.48
	SetTimingSync	Sets the timing synchronization.	P.48
	GetTimingSync	Gets the timing synchronization.	P.49
	SetToleCompOutputFormat	Sets the judgment output format.	P.49
	GetToleCompOutputFormat	Acquires the judgment output format.	P.50
	SetStorobeTime	Sets the strobe time.	P.50
	GetStorobeTime	Gets the strobe time.	P.51

3.3.1. Mode change command

3.3.1.1. SetMode Commands

Changes the operating mode of LK-G3000.3.3.2. When executing measurement/control related commands, set the operation mode of LK-G3000 body to "Normal mode". When executing 3.3.3. setting content related commands, set the operation mode of LK-G3000 body to "Communication mode". The following arguments are

provided.

Item	Type Description	
Argument	VT_I4	Specifies the operation mode. Specify one of the following values. ● 0-Normal mode ● 1-Communication mode

Usage example

```
' Execute SetMode
Call caoCtrl.Execute("SetMode", 0)
```

3.3.2. Measurement and Control-Related Commands

3.3.2.1. GetCalcData Commands

Gets the measured value. The following are the arguments and return values: The return value varies depending on the specified OUT number.

Item	Type Description	
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 0 - OUT1 + OUT2 ● 1 - OUT1 ● 2 - OUT2

- Return value when OUT number "0" is specified as argument:

Item	Type Description		
Return Value	VT_ARRAY | VT_VARIANT		
	0	VT_R4 Or VT_EMPTY	A measure of OUT1. The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.
	1	VT_R4 Or VT_EMPTY	A measure of OUT2. The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.

● Return value when OUT number "1" or "2" is specified in the argument:

Item	Type Description	
Return Value	VT_R4 Or VT_EMPTY	Measured value of the specified OUT number. The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.

Table 3-3 Measurement details

Data Type	Acquired value	Status of the measured value
VT_R4	Normal value	Normal
VT_R4	1. #INF	+ RANGE OVER
VT_R4	-1.#INF	- Range over
VT_EMPTY	--	Judgement standby status

Usage example

```
' Execute GetCalcData
Dim values As Variant
Values = caoCtrl.Execute("GetCalcData", 0)

If Not IsEmpty(values) Then
    If Not IsEmpty(values(0)) Then
        ' OUT1 readings
        Dim value1 As Single
        Value1 = values(0)
    End If
    If Not IsEmpty(values(1)) Then
        ' OUT2 readings
        Dim value2 As Single
        Value2 = values(1)
    End If
End If
```

3.3.2.2. SetTiming Commands

Sets the timing input. When the timing input is set, the measured value at the time of setting is retained. The function at the time of timing input differs according to the measurement mode. For more information on timing input, refer to the timing chart in "Chapter 3 FUNCTION SETTING-SETTING UP MEASUREMENT-USING THE HOLD FUNCTION (MEASUREMENT MODE)" in LK-G3000 manual. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	0	VT_I4	Specify the OUT number to be set. Specify one of the following: ● 0 - OUT1 + OUT2 ● 1 - OUT1 ● 2 - OUT2
	1	VT_BOOL	Specify timing input. Specify ON if TRUE, OFF if FALSE.

Usage example

```
' Execute SetTiming
Dim param As Variant
Param = Array(0, True)
Call caoCtrl.Execute("SetTiming", param)
```

3.3.2.3. SetZero Commands

Sets the auto zero. When Auto Zero is set to ON, the measured value during measurement is set to zero. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	0	VT_I4	Specify the OUT number to be set. Specify one of the following: ● 0 - OUT1 + OUT2 ● 1 - OUT1 ● 2 - OUT2
	1	VT_BOOL	Specify auto zero. Specify ON if TRUE, OFF if FALSE.

Usage example

```
' Execute SetZero
Dim param As Variant
Param = Array(0, True)
Call caoCtrl.Execute("SetZero ", param)
```

3.3.2.4. SetReset Commands

Sets the reset input and resets the measured value of the specified OUT number. The following arguments are provided.

Item	Type Description	
Argument	VT_I4	Specify the OUT number to be set. Specify one of the following: ● 0 - OUT1 + OUT2 ● 1 - OUT1 ● 2- OUT2

Usage example

```
' Execute SetReset
Call caoCtrl.Execute("SetReset", 0)
```

3.3.2.5. SetPanelLock Commands

Locks the keystrokes on the display panel. By locking the display panel, you can prevent accidental operation even if you accidentally touch the operation key. The following arguments are provided.

Item	Type Description	
Argument	VT_BOOL	Specify panel lock. Specify ON if TRUE, OFF if FALSE.

Usage example

```
' Execute SetPanelLock
Call caoCtrl.Execute("SetPanelLock", true)
```

3.3.2.6. SetProgramNo Commands

Switches the program number. The following arguments are provided.

Item	Type Description	
Argument	VT_I4	Specify the program number to be set. You can specify a value from 0 to 7.

Usage example

```
' Execute SetProgramNo
Call caoCtrl.Execute("SetProgramNo ", 0)
```

3.3.2.7. GetProgramNo Commands

Gets the current program number. The return values are shown below.

Item	Type Description	
Return Value	VT_I4	Gets the program number.

Usage example

```
' Execute GetProgramNo
Dim value As Integer
Value = caoCtrl.Execute("GetProgramNo")
```

3.3.2.8. GetFigureData Commands

Acquires the statistical results of the measured values that have been statistically processed. The data subject to statistical processing is the data held in each measurement mode. The following are the arguments and return values:

Item	Type Description		
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2	
Return Value	VT_ARRAY | VT_VARIANT		
	0	VT_R4 Or VT_EMPTY	Mean measured . The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.

Item	Type	Description	
	1	VT_R4 Or VT_EMPTY	Maximum measurement . The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.
	2	VT_R4 Or VT_EMPTY	Minimum measured value . The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.
	3	VT_R4 Or VT_EMPTY	Maximum-Minimum. The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.
	4	VT_R4 Or VT_EMPTY	Standard deviation. The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.
	5	VT_I4	Total number of data.
	6	VT_I4	Number of High judgement data. The total number of data items that have been determined by High.
	7	VT_I4	Number of Go judgment data. The total number of measured values that have been determined by Go.
	8	VT_I4	Number of low judgement data. The total number of measured values that have been judged to be low.

Usage example

```
' Execute GetFigureData
Dim values As Variant
Values = caoCtrl.Execute("GetFigureData", 1)

If Not IsEmpty(values) Then
    If Not IsEmpty(values(0)) Then
        ' Mean
        Dim average As Single
        Average = values(0)
    End If
    If Not IsEmpty(values(1)) Then
        ' Maximum value
        Dim maxValue As Single
        MaxValue = values(1)
    End If
    If Not IsEmpty(values(2)) Then
        ' Minimum value
        Dim minValue As Single
        MinValue = values(2)
    End If
    If Not IsEmpty(values(3)) Then
```

```
        ' Maximum-Minimum
        Dim difValue As Single
        DifValue = values(3)
    End If
    If Not IsEmpty(values(4)) Then
        ' Standard deviation
        Dim stndDev As Single
        StndDev = values(4)
    End If

    ' Total number of data
    Dim totalDataCnt As Long
    TotalDataCnt = values(5)

    ' High judgement data count
    Dim highDataCnt As Long
    HighDataCnt = values(6)

    ' Number of Go judgment data
    Dim goDataCnt As Long
    GoDataCnt = values(7)

    'Number of low judgement data
    Dim lowDataCnt As Long
    LowDataCnt = values(8)

End If
```

3.3.2.9. ClearFigureData Commands

Clears the statistics. The following arguments are provided.

Item	Type	Description
Argument	VT_I4	Specify the OUT number to be cleared. Specify one of the following: ● 0 - OUT1 + OUT2 ● 1 - OUT1 ● 2 - OUT2

Usage example

```
' Execute ClearFigureData
Call caoCtrl.Execute("ClearFigureData", 0)
```

3.3.2.10. StartDataStorage Commands

Starts accumulating data in data storage.

3.3.2.11. StopDataStorage Commands

Stops accumulating data in data storage.

3.3.2.12. ClearDataStorage Commands

Clears the accumulated data in the data storage.

3.3.2.13. GetDataStorageData Commands

Retrieves accumulated data for data storage. When this command is executed, the response wait time must be set in 3.2.1.1. AddContoller method according to the number of accumulated data. For the setting of the number of data items to be accumulated, refer to 3.3.3.5.1. SetDataStorage Commands. The following are the arguments and return values:

Item	Type Description		
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2	
Return Value	VT_EMPTY	When there is no accumulated data	
	Or		
	VT_ARRAY | VT_VARIANT		
	I	VT_R4 Or VT_EMPTY	The measured value. The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.

※ i-number of accumulated data items acquired

Usage example

4.2.1.6

3.3.2.14. GetDataStorageStatus Commands

Acquires the accumulated state of data storage. The return values are shown below.

Item	Type Description		
Return Value	VT_ARRAY | VT_VARIANT		
	0	VT_BOOL	Acquires whether or not accumulation is in progress. ● TRUE-accumulating ● FALSE-stopped
	1	VT_I4	Number of data items stored in OUT1 readings
	2	VT_I4	Number of data items stored in OUT2 readings

Usage example

4.2.1.5

3.3.3. Configuration-Related Commands

3.3.3.1. Panel Display-Related Commands

3.3.3.1.1. SetPanel Commands

Switches the panel display. The following arguments are provided.

Item	Type	Description
Argument	VT_I4	Specify the OUT number to be displayed. Specify one of the following: ● 0 - OUT1+OUT2 ● 1 - OUT1 ● 2 - OUT2

Usage example

```
' Execute SetPanel
Call caoCtrl.Execute("SetPanel", 0)
```

3.3.3.1.2. GetPanel Commands

Gets the currently displayed panel. The return values are shown below.

Item	Type	Description
Return Value	VT_I4	Gets the displayed OUT number. Refer to 3.3.3.1.1 for details of the value.

Usage example

```
' Execute GetPanel
Dim outNo As Integer
OutNo = caoCtrl.Execute("GetPanel")
```

3.3.3.2. Commands Related to Tolerance Setting

3.3.3.2.1. SetTolerance Commands

Sets the judgment value (tolerance judgment value) of the tolerance range. When the upper limit value is exceeded (HI), when the lower limit value is exceeded (LO), and when the upper limit value is exceeded (GO), the judgment is made in three levels, and the display and output can be performed. If the measured value is raised or lowered near the tolerance judgment value, the judgment output may repeat ON/OFF. When hysteresis is set, the width of the detected value and the restored value of tolerance judgment can be set, so that such a situation can be prevented. For details on the displayed information and hysteresis, refer to "Chapter 2-Setting Tolerance Determination Values" in LK-G3000 manuals. The following arguments are provided.

Item	Type Description
Argument	VT_ARRAY | VT_I4

Item	Type Description		
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_I4	Specifies the upper tolerance value. Specify a value from-999999 to 999999.
	2	VT_I4	Specifies the lower tolerance value. Specify a value from-999999 to 999999.
	3	VT_I4	Specifies the tolerance hysteresis. Specify a value from 0 to 999999.

※ Specify so that "Tolerance high limit value-(tolerance low limit value) > tolerance hysteresis" is satisfied. An example is shown below.

Good example-> Tolerance high limit = 1000, Tolerance low limit = 100, Tolerance hysteresis = 0

Poor Example-> Tolerance Upper Limit =-1000, Tolerance Lower Limit = 100, Tolerance Hysteresis = 0

Usage example

```
' Execute SetTolerance
Dim param As Variant
Param = Array(1, 1000, 100, 0)
Call caoCtrl.Execute("SetTolerance", param)
```

3.3.3.2.2. GetTolerance Commands

Acquires the judgment value (tolerance judgment value) of the tolerance range. The following are the arguments and return values:

Item	Type Description		
Argument	VT_I4		Specify the OUT number to be displayed. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_ARRAY | VT_I4		
	0	VT_I4	Gets the upper tolerance value.
	1	VT_I4	Gets the lower tolerance value.
	2	VT_I4	Obtains the tolerance hysteresis.

Usage example

```
' Execute GetTolerance
Dim tolerance As Variant
Tolerance = caoCtrl.Execute("GetTolerance", 1)
If Not IsEmpty(tolerance) Then
```

```
    ' Tolerance upper limit
    Dim maxLimit As Integer
    MaxLimit = tolerance(0)
    ' Lower tolerance limit
    Dim minLimit As Integer
    MinLimit = tolerance(1)
    ' Tolerance hysteresis
    Dim hysterisis As Integer
    Hysterisis = tolerance(2)
End If
```

3.3.3.3. Head setting-related commands

This command sets/obtains the function related to sensing for stable detection.

3.3.3.3.1. SetAbleMode Commands

Set ABLE tuning mode. ABLE function automatically adjusts the appropriate amount of light and sensitivity to the surface condition (color, gloss, material) of the object. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
	1	VT_I4	Specify ABLE tuning mode. For details on ABLE tuning mode, see Table3-4. Specify one of the following:Table 3-4 ABLE Tuning Mode Detailed ● Automatic ● 1-Documentation

Table 3-4 ABLE Tuning Mode Detailed

Mode	Function
Automatic	Automatically adjust the light to an appropriate amount. Normally, select here.
Documentation	Adjust the light intensity and sensitivity by limiting the adjustment range to any range from 1 to 99. Select this if the reflectance of the object changes significantly at a quick cycle, or if you want to detect only the object.

Usage example

```
' Execute SetAbleMode
Dim param As Variant
Param = Array(1, 0)
Call caoCtrl.Execute("SetAbleMode", param)
```

3.3.3.3.2. GetAbleMode Commands

Gets ABLE tuning mode. The following are the arguments and return values:

Item	Type	Description
Argument	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
Return Value	VT_I4	Gets ABLE tuning mode. Refer to 3.3.3.3.1 for details of the value.

Usage example

```
' Execute GetAbleMode
Dim mode As Integer
Mode = caoCtrl.Execute("GetAbleMode", 1)
```

3.3.3.3.3. SetAbleMinMax Commands

Sets ABLE control area. The following arguments are provided.

Item		Type	Description
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
	1	VT_I4	Specify ABLE upper limit. Specify a value from 1 to 99.
	2	VT_I4	Specifies the lower ABLE limit. Specify a value from 1 to 99.

※ If a setting that does not satisfy "ABLE max.-ABLE min. >= 0" is set, an out-of-range error is returned from the device.

Usage example

```
' Execute SetAbleMinMax
Dim param As Variant
Param = Array(1, 10, 8)
Call caoCtrl.Execute("SetAbleMinMax", param)
```

3.3.3.3.4. GetAbleMinMax Commands

Gets ABLE control scope. The following are the arguments and return values:

Item	Type	Description

Item	Type Description		
Argument	VT_I4		Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
Return Value	VT_ARRAY | VT_I4		
	0	VT_I4	Get ABLE min.
	1	VT_I4	Get ABLE max.

Usage example

```
' Execute GetAbleMinMax
Dim value As Variant
Value = caoCtrl.Execute("GetAbleMinMax", 1)
If Not IsEmpty(value) Then
        ' ABLE max
        Dim maxAble As Integer
        MaxAble = value(0)
        ' ABLE min.
        Dim minAble As Integer
        MinAble = value(1)
End If
```

3.3.3.3.5. SetMeasureMode Commands

Set the measurement mode according to the object to be measured. By specifying what the object to be measured is, stable detection is performed. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
	1	VT_I4	Specifies the measurement mode. Refer to Table 3-5 for details on each measurement mode. Specify one of the following: ● 0-Standard ● 1-Translucent body ● 2-transparency ● 3-Transparency 2 ● 4-Multiple reflector

Table 3-5 Measurement Mode Details

Mode	Function

Mode	Function
Standard	Normally, this setting is used.
Translucent body	Corresponds to objects such as semi-transparent plastic that can be used to soak light.
Transparency	Used to measure transparency variations and thickness. This is used when the reflectivity of multiple planes of the transparency is the same party.
Transparency 2	This is used when reflectivity of multiple surfaces (up to four surfaces) such as the front and back surfaces of a transparent body is different, etc.
Multiple reflector	Used for bending measurement of terminals of ICs and connectors, etc.

Usage example

```
' Execute SetMeasureMode
Dim param As Variant
Param = Array(1, 0)
Call caoCtrl.Execute("SetMeasureMode", param)
```

3.3.3.3.6. GetMeasureMode Commands

Gets the measurement mode. The following are the arguments and return values:

Item	Type	Description
Argument	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
Return Value	VT_I4	Gets the measurement mode. Refer to 3.3.3.3.5 for details of the value.

Usage example

```
' Execute GetMeasureMode
Dim mode As Integer
Mode = caoCtrl.Execute("GetMeasureMode", 1)
```

3.3.3.3.7. SetNumAlarm Commands

Set the number of alarm processing operations. The following arguments are provided.

Item	Type		Description
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B

Item	Type	Description	
	1	VT_I4	Specify the number of times the alarm is processed. Specify a value from 0 to 999.

Usage example

```
' Execute SetNumAlarm
Dim param As Variant
Param = Array(1, 5)
Call caoCtrl.Execute("SetNumAlarm", param)
```

3.3.3.3.8. SetAlarmLevel Commands

Sets the alarm level. The following arguments are provided.

Item	Type	Description	
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
	1	VT_I4	Specify the alarm level. Specify a value from 0 to 9. The larger the value, the more likely it is to be an alarm.

Usage example

```
' Execute SetAlarmLevel
Dim param As Variant
Param = Array(1, 3)
Call caoCtrl.Execute("SetAlarmLevel", param)
```

3.3.3.3.9. GetAlarm Commands

Get alarm information. The following are the arguments and return values:

Item	Type	Description	
Argument	VT_I4		Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
Return Value	VT_ARRAY | VT_I4		
	0	VT_I4	Gets the alarm level.
	1	VT_I4	Acquires the number of alarm processing operations.

Usage example

```
' Execute GetAlarm
Dim alarmvalue As Variant
Alarmvalue = caoCtrl.Execute("GetAlarm", 1)
If Not IsEmpty(alarmvalue) Then
        ' Alarm level
        Dim alarmLevel As Integer
        AlarmLevel = alarmvalue(0)
        ' Number of alarm processes
        Dim numAlarm As Integer
        NumAlarm = alarmvalue(1)
End If
```

3.3.3.3.10. StartABLE Commands

Starts ABLE tuning. ABLE tuning function optimizes ABLE tuning by allowing the target head to actually measure the object. Perform ABLE tuning referring to Fig. 3-1 ABLE Tuning Flow. When ABLE tuning is executed, ABLE tuning mode is set manually, and ABLE upper limit and ABLE lower limit are set to the adjusted values. The following arguments are provided.

Item	Type	Description
Argument	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B

1. **Measure the actual object as shown in the illustration below.**
2. **Start ABLE tuning.**
3. **Move the object slowly.**
4. **Exits ABLE tuning mode.**

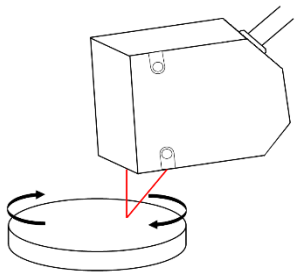


Fig. 3-1 ABLE Tuning Flow

Usage example

```
' Execute StartABLE
Call caoCtrl.Execute("StartABLE", 1)
```

3.3.3.3.11. StopABLE Commands

Finish ABLE tuning.

Usage example

```
' Execute StopABLE
Call caoCtrl.Execute("StopABLE")
```

3.3.3.3.12. CancelABLE Commands

Cancels ABLE tuning.

Usage example

```
' Execute CancelABLE
Call caoCtrl.Execute("CancelABLE")
```

3.3.3.3.13. SetReflectionMode Commands

Sets the installation mode. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
	1	VT_I4	Specify the installation mode. Refer to Table 3-6 for details on the installation mode. Specify one of the following:Table 3-6 Installation Mode Details ● 0-Diffuse reflection ● 1-Regular reflection

Table 3-6 Installation Mode Details

Mode	Function
Diffuse reflection	Sets the diffuse reflection. Typically, you select this setting.
Regular reflection	Sets the regular reflection. Select this when the object to be measured is a mirror or glass.

Usage example

```
' Execute SetReflectionMode
Dim param As Variant
Param = Array(1, 0)
Call caoCtrl.Execute("SetReflectionMode", param)
```

3.3.3.3.14. GetReflectionMode Commands

Gets the installation mode. The following are the arguments and return values:

Item	Type Description	
Argument	VT_I4	Specify the head number. Specify one of the following: ● 1 - HEAD-A ● 2 - HEAD-B
Return Value	VT_I4	Gets the installation mode. Refer to 3.3.3.3.13 for details of the value.

Usage example

```
' Execute GetReflectionMode
Dim mode As Integer
Mode = caoCtrl.Execute("GetReflectionMode", 1)
```

3.3.3.4. OUT setting-related commands

This command sets/gets functions related to data processing.

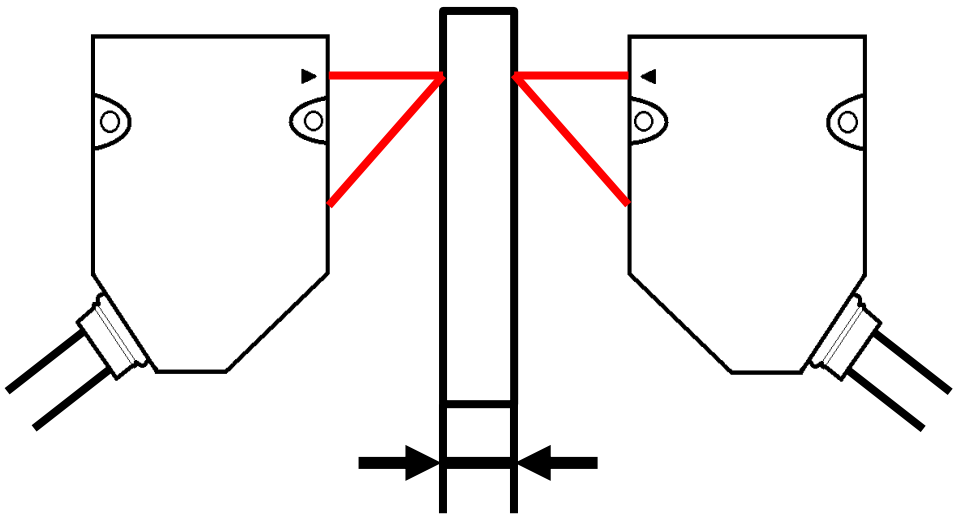
3.3.3.4.1. SetCalcMethod Commands

Sets the calculation method between heads. Surface displacement, thickness, and level difference can be measured by calculating the data obtained by head setting of Head A or Head B according to the object to be measured. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_I4	Specifies the operation method. See Table 3-7 for details on the calculation method. Specify one of the following: ● 0-Head A ● 1-Head B ● 2-Head A + Head B ● 3-Head A to Head B ● 4-Head A transparency ● 5-Head B transparency

Item	Type Description		
	2	VT_I4	Specify the measurement target. Refer to Table 3-8 for details on the measurement targets. Specify one of the following: ● 0-Peak 1 ● 1-Peak 2 ● 2-Peak 3 ● 3-Peak 4 ● 4-Peak 1-Peak 2 ● 5-Peak 1-Peak 3 ● 6-Peak 1-Peak 4 ● 7-Peak 2-Peak 3 ● 8-Peak 2-Peak 4 ● 9-Peak 3-Peak 4

Table 3-7 Operation method details

Operation	Function
Head A	Surface displacement measurement of head A or head B
Head B	
Head A + Head B	Thickness determination using head A and head B  Thickness judgment

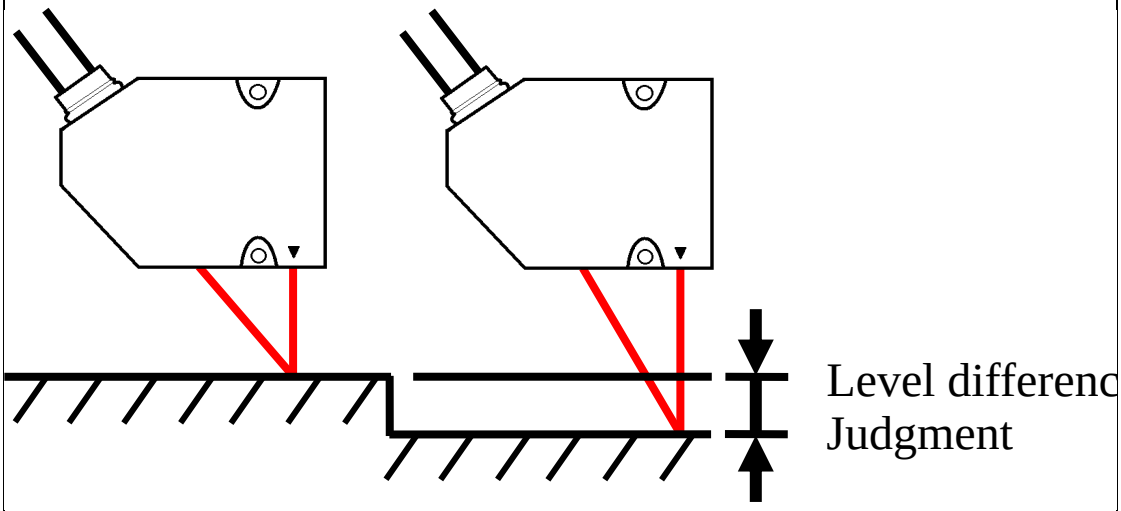
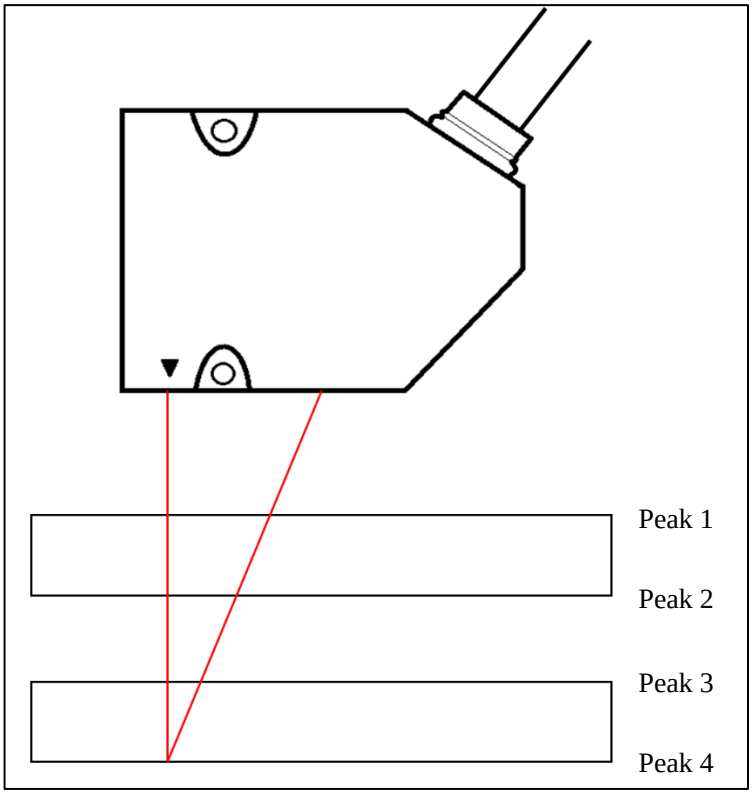
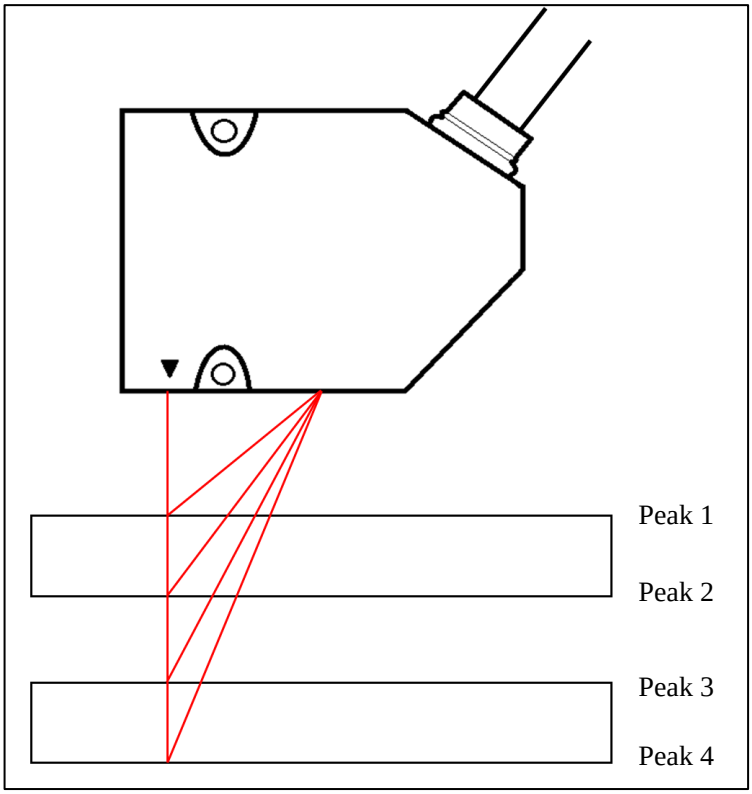
Operation	Function
Head A to Head B	Thickness determination using head A and head B 
Head A transparency	Measure the displacement and thickness of the transparent body. The selection of the target plane is specified by the measurement target.
Head B transparency	

Table 3-8 Measurement target details

Measuring surface	Function
Peak 1	1 Displacement measurement of reflecting surface
Peak 2	
Peak 3	
Peak 4	

<table>
<tr><th>Measuring surface</th><th>Function</th></tr>
<tr><td></td><td> 
</td></tr>
<tr><td>Peak 1-2</td><td rowspan="5">2 Measure the reflective surface and select Peak 1-2 to measure the thickness of the first glass. You can also measure the spacing between the first and second glass by selecting Peak 2-3.</td></tr>
<tr><td>Peak 1-3</td></tr>
<tr><td>Peak 1-4</td></tr>
<tr><td>Peak 2-3</td></tr>
<tr><td>Peak 2-4</td></tr>
</table>

Measuring surface	Function
Peak 3-4	

Usage example

```
' Execute SetCalcMethod
Dim param As Variant
Param = Array(1, 0, 0)
Call caoCtrl.Execute("SetCalcMethod", param)
```

3.3.3.4.2. GetCalcMethod Commands

Gets the calculation method. The following are the arguments and return values:

Item	Type		Description
Argument	VT_I4		Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_ARRAY | VT_I4		
	0	VT_I4	Gets the calculation method. Refer to 3.3.3.4.1 for details of the value.
	1	VT_I4	Gets the measurement object. Refer to 3.3.3.4.2 for details of the value.

Usage example

```
' Execute GetCalcMethod
Dim value As Variant
Value = caoCtrl.Execute("GetCalcMethod", 1)
If Not IsEmpty(value) Then
        ' Calculation method
        Dim method As Integer
        Method = value(0)
        ' Measurement object
        Dim target As Integer
        Target = value(1)
End If
```

3.3.3.4.3. SetScaling Commands

Set the scaling. By setting the scaling, the displayed value relative to the measured value can be calibrated arbitrarily. Calibration sets the values to be displayed for any two points. For each OUT1,OUT2, head A and head B can be calibrated. For more information about scaling, see LK-G3000 documentation. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the OUT number and head number. Specify one of the following: ● 1-HEAD-A of OUT1 ● 2-HEAD-B of OUT1 ● 3-HEAD-A of OUT2 ● 4-HEAD-B of OUT2
	1	VT_I4	Specify the first measurement value. Specify a value from-999999 to 999999.
	2	VT_I4	Specify the 1st display value. Specify a value from-999999 to 999999.
	3	VT_I4	Specify the second measurement value. Specify a value from-999999 to 999999.
	4	VT_I4	Specify the second display value. Specify a value from-999999 to 999999.

※ The configuration fails if the following conditions are not met:

(1) Input value 1-Input value 2 ≠ 0

(2) (2nd point display value-1st point display value)/(2nd point measurement value-1st point measurement value) < 10

Usage example

```
' Execute SetScaling
Dim param As Variant
```

```
Param = Array(1, 500, 10, 1000, 1000)
Call caoCtrl.Execute("SetScaling", param)
```

3.3.3.4.4. GetScaling Commands

Get the scaling. The following are the arguments and return values:

Item	Type Description		
Argument	VT_I4	Specify the OUT number and head number. Specify one of the following: ● 1-HEAD-A of OUT1 ● 2-HEAD-B of OUT1 ● 3-HEAD-A of OUT2 ● 4-HEAD-B of OUT2	
Return Value	VT_ARRAY | VT_I4		
	0	VT_I4	Gets the first measured value.
	1	VT_I4	Gets the 1st point display value.
	2	VT_I4	Acquires the second measurement value.
	3	VT_I4	Retrieves the second displayed value.

Usage example

```
' Execute GetScaling
Dim value As Variant
Value = caoCtrl.Execute("GetScaling", 1)
If Not IsEmpty(value) Then
        ': Measured value at the first point
        Dim inputValue1 As Integer
        InputValue1 = value(0)
        ' 1st point display value
        Dim outputValue1 As Integer
        OutputValue1 = value(1)
        ' 2nd measured value
        Dim inputValue2 As Integer
        InputValue2 = value(2)
        ' 2nd display value
        Dim outputValue2 As Integer
        OutputValue2 = value(3)
End If
```

3.3.3.4.5. SetFilter Commands

Configure the filter. By applying filters, you can make stable measurements. For more information on filtering, refer to Chapter 3-Setting Measurement Output Conditions-Filtering for Stable Measurement in LK-G3000 Manual. The following arguments are provided.

Item	Type Description
Argument	VT_ARRAY | VT_I4

Item	Type	Description	
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_I4	Specifies the filter mode. Specify one of the following: ● 0-Moving average ● 1-Low-pass filter ● 2-High-pass filter
	2	VT_I4	The contents specified depend on the above filter mode. Specify one of the following from the specified filter mode. [When the filter mode is moving average (0)] Specify the number of averages. Specify one of the following: ● 0-Once ● 1-4 times ● 2-16 times ● 3-64 times ● 4-256 times ● 5-1024 times ● 6-4096 times ● 7-16384 times ● 8-65536 times ● 9-262144 times [When the filter mode is low-pass filter (1) or high-pass filter (2)] Specifies the cutoff frequency. Specify one of the following: ● 0 - 1000Hz ● 1 - 300Hz ● 2 - 100Hz ● 3 - 30Hz ● 4 - 10Hz ● 5 - 3Hz ● 6 - 1Hz ● 7 - 0.3Hz ● 8 - 0.1Hz

Usage example

```
' Execute SetFilter
Dim param As Variant
```

```
Param = Array(1, 0, 0)
Call caoCtrl.Execute("SetFilterMode", param)
```

3.3.3.4.6. GetFilter Commands

Get filter information. The following are the arguments and return values:

Item	Type	Description
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_ARRAY | VT_I4	
	0 VT_I4	Gets the filter mode. Refer to 3.3.3.4.5 for details of the value.
	1 VT_I4	The contents obtained depend on the above filter mode. Refer to 3.3.3.4.5 for details of the value.

Usage example

```
' Execute GetFilter
Dim value As Variant
Value = caoCtrl.Execute("GetFilterMode", 1)
If Not IsEmpty(value) Then
        ' Filter mode
        Dim filterMode As Integer
        FilterMode = value(0)
        ' Data according to the filter mode
        Dim filterValue As Integer
        FilterValue = value(1)
End If
```

3.3.3.4.7. SetTriggerMode Commands

Sets or queries the trigger mode. The following arguments are provided.

Item	Type		Description
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_I4	Specify the trigger mode. For details on the trigger mode, see Table 3-9. Specify one of the following: ● 0-External trigger 1 ● 1-External trigger 2

Table 3-9 Trigger mode details

Mode	Function		
	Standard	Peak Hold/Bottom Hold/Peak to Peak Hold/Average Hold	Sample Hold
Trigger 1	Holds the internal measurement value while the timing input is rising (ON).	The sampling period is defined as the timing input from the rising edge to the next rising edge.	Holds the internal measurement value when the timing input is started.
Trigger 2		The sampling period is defined as the period from the falling (OFF) edge of the timing input to the next rising edge.	When the timing input starts, the data of the set average number of times is sampled from that time, and the fixed internal measurement value is held.

Usage example

```
' Execute SetTriggerMode
Dim param As Variant
Param = Array(1, 0)
Call caoCtrl.Execute("SetTriggerMode", param)
```

3.3.3.4.8. GetTriggerMode Commands

Gets the trigger mode. The following are the arguments and return values:

Item	Type	Description
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_I4	Gets the trigger mode. See 3.3.3.4.7 for more information on the values.

Usage example

```
' Execute GetTriggerMode
Dim mode As Integer
Mode = caoCtrl.Execute("GetTriggerMode", 1)
```

3.3.3.4.9. SetOffset Commands

Set the offset. By setting the offset, you can add or subtract any value from the displayed value. If an offset is set, the offset value can be displayed when auto zero is executed. The offset value is set for the measured value after measurement mode processing and auto zero processing. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_I4	Specify the offset. Specify a value from-999999 to 999999.

Usage example

```
' Execute SetOffset
Dim param As Variant
Param = Array(1, 10)
Call caoCtrl.Execute("SetOffset", param)
```

3.3.3.4.10. GetOffset Commands

Gets the offset. The following are the arguments and return values:

Item	Type Description	
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_I4	Gets the offset.

Usage example

```
' Execute GetOffset
Dim offset As Integer
Offset = caoCtrl.Execute("GetOffset", 1)
```

3.3.3.4.11. SetAnalogScaling Commands

Sets the analog output scaling. For more information on analog output scaling, refer to "Chapter 3-Setting the Output Conditions of Measured Values-Scaling the Analog Output" in LK-G3000 manuals. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_I4	Specify the display value of the first point. Specify a value from-999999 to 999999.

Item	Type Description		
	2	VT_I4	Specifies the output voltage value of the first point in mV units. Specify a value from-10500 to 10500.
	3	VT_I4	Specify the display value of the second point. Specify a value from-999999 to 999999.
	4	VT_I4	Specify the output voltage value of the second point in mV units. Specify a value from-10500 to 10500.

※ The configuration fails if the following conditions are not met:

(1) Display value of the first point-Display value of the second point ≠ 0

(2) (2nd point output voltage value-1st point output voltage value)/(2nd point display value-1st point display value) ≦ 10

Usage example

```
' Execute SetAnalogScaling
Dim param As Variant
Param = Array(1, 500, 10, 1000, 1000)
Call caoCtrl.Execute("SetAnalogScaling", param)
```

3.3.3.4.12. GetAnalogScaling Commands

Obtains the analog output scaling. The following are the arguments and return values:

Item	Type Description		
Argument	VT_I4		Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_ARRAY | VT_I4		
	0	VT_I4	Gets the display value of the first point.
	1	VT_I4	Acquires the output voltage value of the first point.
	2	VT_I4	Acquires the displayed value of the second point.
	3	VT_I4	Acquires the output voltage value of the second point.

Usage example

```
' Execute GetAnalogScaling
Dim value As Variant
Value = caoCtrl.Execute("GetAnalogScaling", 1)
If Not IsEmpty(value) Then
        ' 1st point display value
        Dim inputValue1 As Integer
        InputValue1 = value(0)
        ' 1st output voltage
        Dim outputValue1 As Integer
        OutputValue1 = value(1)
```

```
        ' 2nd display value
        Dim inputValue2 As Integer
        InputValue2 = value(2)
        ' Second output voltage
        Dim outputValue2 As Integer
        OutputValue2 = value(3)
End If
```

3.3.3.4.13. SetCalcMode Commands

Sets the measurement mode. For more information on measurement modes, refer to "Chapter 3-Setting Measurement Output Conditions-Using the Hold Function (Measurement Mode)" in LK-G3000 Manual. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_I4	Specifies the measurement mode. For details on the measurement modes that can be specified, see Table 3-10. Specify one of the following: ● 0-Standard ● 1-Peak Hold ● 2-Bottom Hold ● 3-Peak to Peak Hold ● 4-Sample Hold ● 5-Average Hold

Table 3-10 Measurement Mode Details

Mode	Function
Standard	Measured results can be displayed and output at any time.
Peak Hold	The maximum value within the sampling period can be measured.
Bottom Hold	The minimum value within the sampling period can be measured.
Peak to Peak Hold	You can measure the "maximum value-minimum value" within the sampling period.
Sample Hold	The value at the moment the timing input is turned ON can be measured.
Average Hold	You can measure the average value within the sampling period.

Usage example

```
' Execute SetCalcMode
Dim param As Variant
Param = Array(1, 0)
```

```
Call caoCtrl.Execute("SetCalcMode", param)
```

3.3.3.4.14. GetCalcMode Commands

Gets the measurement mode. The following are the arguments and return values:

Item	Type	Description
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_I4	Gets the measurement mode. Refer to 3.3.3.4.13 for details of the value.

Usage example

```
' Execute GetCalcMode
Dim mode As Integer
Mode = caoCtrl.Execute("GetCalcMode", 1)
```

3.3.3.4.15. SetDisplayUnit Commands

Set the minimum display unit to be displayed on the panel. The following arguments are provided.

Item			Type Description
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_I4	Specifies the smallest display unit. Specify one of the following: ● 0 - 0.01mm ● 1 - 0.001mm ● 2 - 0.0001mm ● 3 - 0.00001mm ● 4 - 0.01μm ● 5 - 0.001μm

Usage example

```
' Execute SetDisplayUnit
Dim param As Variant
Param = Array(1, 0)
Call caoCtrl.Execute("SetDisplayUnit", param)
```

3.3.3.4.16. GetDisplayUnit Commands

Gets the smallest display unit. The following are the arguments and return values:

Item	Type Description	
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_I4	Gets the smallest display unit. Refer to 3.3.3.4.15 for details of the value.

Usage example

```
' Execute GetDisplayUnit
Dim unit As Integer
Unit = caoCtrl.Execute("GetDisplayUnit", 1)
```

3.3.3.4.17. SetAnalogThrough Commands

Sets the analog through. If the analog slew is set to ON while the measurement value is being held in measurement mode, the internal measurement value before holding is output in analog. The following arguments are provided.

Item	Type Description		
Argument	VT_ARRAY | VT_VARIANT		
	0	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
	1	VT_BOOL	Specifies the analog through. ● TRUE - ON ● FALSE - OFF

Usage example

```
' Execute SetAnalogThrough
Dim param As Variant
Param = Array(1, false)
Call caoCtrl.Execute("SettAnalogThrough", param)
```

3.3.3.4.18. GetAnalogThrough Commands

Acquires an analog through. The following are the arguments and return values:

Item	Type Description	
Argument	VT_I4	Specify the OUT number. Specify one of the following: ● 1 - OUT1 ● 2 - OUT2
Return Value	VT_BOOL	Acquires an analog through.

Usage example

```
' Execute GetAnalogThrough
Dim value As Variant
Value = caoCtrl.Execute("GetAnalogThrough", 1)
```

3.3.3.5. Common setting related commands

This command sets/obtains the common functions related to the head setting and OUT setting.

3.3.3.5.1. SetDataStorage Commands

Set the target OUT, number of accumulated points, and accumulation period to accumulate data storage. The following arguments are provided.

Item	Type	Description	
Argument	VT_ARRAY | VT_I4		
	0	VT_I4	Specify the target OUT. Specify one of the following: ● 0-No target OUT ● 1 - OUT1 ● 2 - OUT2 ● 3-OUT1 and OUT2
	1	VT_I4	Specify the number of points to be accumulated. Specify a value from 1 to 65536.
	2	VT_I4	Specify the accumulation period. Specify one of the following: ● 0-Sampling cycle × 1 ● 1-Sampling period x 2 ● 2-Sampling cycle × 5 ● 3-Sampling period x 10 ● 4-Sampling period x 20 ● 5-Sampling cycle × 50 ● 6-Sampling period × 100 ● 7-Sampling period × 200 ● 8-Sampling period × 500 ● 9-Sampling period × 1000

Usage example

```
' Execute SetDataStorage
Dim param As Variant
Param = Array(1, 100, 0)
Call caoCtrl.Execute("SetDataStorage", param)
```

3.3.3.5.2. GetDataStorage Commands

Retrieves the target OUT, number of accumulated points, and accumulation period for data storage. The following are the arguments and return values:

Item	Type Description		
Return Value	VT_ARRAY | VT_I4		
	0	VT_I4	Gets the target OUT. Refer to 3.3.3.5.1 for details of the value.
	1	VT_I4	Retrieves the accumulated score.
	2	VT_I4	Acquires the accumulation period. Refer to 3.3.3.5.1 for details of the value.

Usage example

```
' Execute GetDataStorage
Dim value As Variant
Value = caoCtrl.Execute("GetDataStorage")
If Not IsEmpty(value) Then
        ' Target OUT
        Dim out As Integer
        Out = value(0)
        ' Accumulated score
        Dim dataCnt As Integer
        DataCnt = value(1)
        ' Storage cycle
        Dim cycle As Integer
        Cycle = value(2)
End If
```

3.3.3.5.3. SetSamplingCycle Commands

Sets the sampling period. The following arguments are provided.

Item	Type Description	
Argument	VT_I4	Specify the sampling period. Specify one of the following: ● 0 - 20μs ● 1 - 50μs ● 2 - 100μs ● 3 - 200μs ● 4 - 500μs ● 5 - 1000μs

Usage example

```
' Execute SetSamplingCycle
Call caoCtrl.Execute("SetSamplingCycle", 0)
```

3.3.3.5.4. GetSamplingCycle Commands

Acquires the sampling period. The return values are shown below.

Item	Type	Description
Return Value	VT_I4	Acquires the sampling period. Refer to 3.3.3.5.3 for details of the value.

Usage example

```
' Execute GetSamplingCycle
Dim value As Integer
Value = caoCtrl.Execute("GetSamplingCycle")
```

3.3.3.5.5. SetMutualInterPrev Commands

Configure mutual interference prevention. If mutual interference prevention is set to ON, the two heads will flash alternately and will not receive interference from the other head. The following arguments are provided.

Item	Type	Description
Argument	VT_BOOL	Specify mutual interference prevention. ● TRUE - ON ● FALSE - OFF

Usage example

```
' Execute SetMutualInterPrev
Call caoCtrl.Execute("SetMutualInterPrev", true)
```

3.3.3.5.6. GetMutualInterPrev Commands

Acquires mutual interference prevention. The return values are shown below.

Item	Type	Description
Return Value	VT_BOOL	Acquires mutual interference prevention.

Usage example

```
' Execute GetMutualInterPrev
Dim value As Variant
Value = caoCtrl.Execute("GetMutualInterPrev")
```

3.3.3.5.7. SetTimingSync Commands

Sets how OUT1 and OUT2 timings are controlled. The following arguments are provided.

Item	Type	Description

Argument	VT_I4	Specify the timing synchronization. See Table 3-11 for more information about timing synchronization. Specify one of the following: ● 0-Asynchronous ● 1-Synchronization

Table 3-11 Details of timing synchronization

Timing Synchronization	Function
Asynchronous	Controls OUT1 and OUT2 asynchronously. Assign independent inputpins to each OUT1,OUT2. ● OUT1 : 8 of the 12-pin terminal block ● OUT2 : Expansion connector number 8
Synchronization	Synchronize and control OUT1 and OUT2. No. 8 of the 12-pole terminal block corresponds, and No. 6 of the expansion connector is disabled.

※ For input terminals, refer to "Chapter 4 Input Terminals" in LK-G3000 manuals.

Usage example

```
' Execute SetTimingSync
Call caoCtrl.Execute("SetTimingSync", 0)
```

3.3.3.5.8. GetTimingSync Commands

Gets the timing synchronization. The return values are shown below.

Item	Type	Description
Return Value	VT_I4	Gets the timing synchronization. See 3.3.3.5.7 for more information on the values.

Usage example

```
' Execute GetTimingSync
Dim value As Integer
Value = caoCtrl.Execute("GetTimingSync")
```

3.3.3.5.9. SetToleCompOutputFormat Commands

Set the output form of tolerance judgment output. The following arguments are provided.

Item	Type	Description

Argument	VT_I4	Specifies the judgment output format. Refer to Table 3-12 for details on the judgment output format. Specify one of the following: ● 0-Normal ● 1-Hold ● 2-off-delay

Table 3-12 Details of the judgment output mode

Timing Synchronization	Function
Normal	Output according to tolerance judgment.
Hold	Holds the output that is turned ON. Releases the hold by resetting the measured value.
Off delay	An off delay of 60ms is applied to the normal output. Releases the hold by resetting the measured value.

Usage example

```
' Execute SetToleCompOutputFormat
Call caoCtrl.Execute("SetToleCompOutputFormat", 0)
```

3.3.3.5.10. GetToleCompOutputFormat Commands

Acquires the output format of tolerance judgment output. The return values are shown below.

Item	Type	Description
Return Value	VT_I4	Acquires the judgment output format. See 3.3.3.5.9 for more information on the values.

Usage example

```
' Execute GetToleCompOutputFormat
Dim value As Integer
Value = caoCtrl.Execute("GetToleCompOutputFormat")
```

3.3.3.5.11. SetStorobeTime Commands

Sets the time (one-shot output time) at which the strobe output turns ON. The following arguments are provided.

Item	Type	Description

Argument	VT_I4	Specifies the strobe time. Specify one of the following: ● 0 - 2ms ● 1 - 5ms ● 2 - 10ms ● 3 - 20ms

Usage example

```
' Execute SetStorobeTime
Call caoCtrl.Execute("SetStorobeTime", 0)
```

3.3.3.5.12. GetStorobeTime Commands

Used to acquire the time (one-shot output time) during which the strobe output turns ON. The return values are shown below.

Item	Type Description	
Return Value	VT_I4	Gets the strobe time. Refer to 3.3.3.5.11 for details of the value.

Usage example

```
' Execute GetStorobeTime
Dim value As Integer
Value = caoCtrl.Execute("GetStorobeTime")
```

3.4. Variable list

Defines a list of variables that can be used in each class. Variables refer to objects of CaoVariable classes.

3.4.1. CaoController class-variable

Variable name	Description	Value		See Also
		Get	Put	
@MAKER_NAME	Obtain the manufacturer's name.	✓	-	P.51
@VERSION	Get the DLL version.	✓	-	P.52
@CALCDATA	Obtains the measured OUT1 and OUT2 during measurement.	✓	✓	P.52

3.4.1.1. @MAKER_NAME

Obtain the manufacturer's name.

Data Type

Type Description	
VT_BSTR	Obtain the manufacturer's name.

Usage example

```
' Add Variable
Dim var As CaoVariable
Var = caoCtrl.AddVariable("@MAKER_NAME")
' Acquisition of Values
Dim name As String
Name = var.value
```

3.4.1.2. @VERSION

Gets the DLL version.

Data Type

Type Description	
VT_BSTR	Get the DLL version. *.*.*

Usage example

```
' Add Variable
Dim var As CaoVariable
Var = caoCtrl.AddVariable("@VERSION", "")
' Acquisition of Values
Dim version As String
Version = var.value
```

3.4.1.3. @CALCDATA

Gets OUT1 and OUT2 readings being measured. If you use this variable to obtain a value, you must switch the operation mode of LK-G3000 body to normal mode.

Data Type

Type Description		
VT_ARRAY | VT_VARIANT		
0	VT_R4 Or VT_EMPTY	A measure of OUT1. The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.
1	VT_R4 Or VT_EMPTY	A measure of OUT2. The value obtained depends on the state of the measured value. See Table 3-3 for more information on the measurements that are taken.

Usage example

```
' Add Variable
Dim var As CaoVariable
Set var = caoCtrl.AddVariable("@CALCDATA")
```

```
' Retrieving Values
Dim calcvalues As Variant
Calcvalues = caoVrl.value

If Not IsEmpty(calcvalues) Then
        If Not IsEmpty(calcvalues(0)) Then
                ' OUT1 readings
                Dim value1 As Single
                Value1 = calcvalues(0)
        End If
        If Not IsEmpty(calcvalues(1)) Then
                ' OUT2 readings
                Dim value2 As Single
                Value2 = calcvalues(1)
        End If
End If
```

4. Programming by LK-G3000 providers

With LK-G3000 providers, you can connect LK-G3000 to the client computer as follows:

- Creating a CaoEngine
- Creating a CaoWorkspace
- Creating a CaoController

After you connect to LK-G3000, you can access CaoController by using its Execute method or by creating a CaoVariabe object.

4.1. Sample-programming to obtain OUT1 and OUT2 measurements

This example shows a sample program that reads OUT1 and OUT2 readings. Table 4-1 describes the requirements of the sample program, and Figure 4-1 describes the flow of the sample program.

Table 4-1 Sample program requirements for acquiring measurements

Requirements	Description
Host	Connect via RS-232C communication
	The host port number is 1.
Process Description	Read OUT1 and OUT2 readings from LK-G3000.

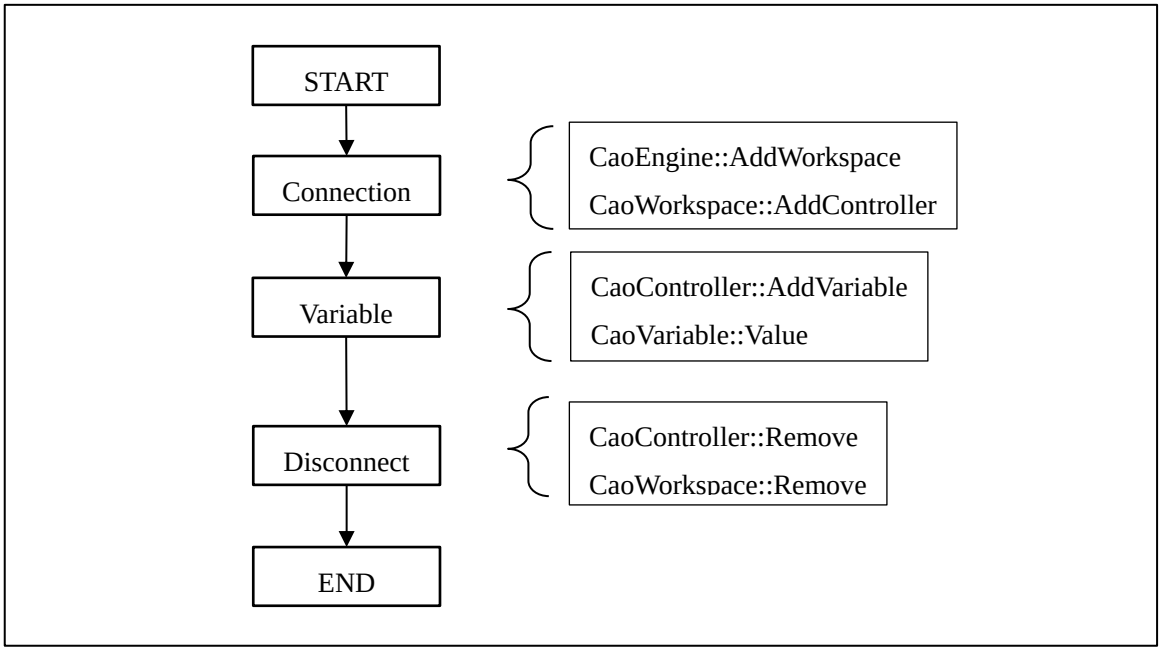


Figure 4-1 Flow of reading OUT1 and OUT2 readings

Specific codes are given in the following sections.

4.1.1. Sample program

The following is an overview of the sample program.

Sample **GetCalcDataSample.cs**

```
' Object
Dim caoEng As CaoEngine
Dim caoWs As CaoWorkspace
Dim caoCtrl As CaoController
Dim caoVrl As CaoVariable
Private Sub Main()
    ' Connection
    Call Connect

    ' Execute this command if the operating mode of the device is not Normal mode.
    'Call caoCtrl.Execute("SetMode", 0)

    ' Retrieving Values
    Dim values As Variant
    Values = caoVrl.value
    If Not IsEmpty(values) Then
        ' OUT1 readings
        If Not IsEmpty(values(0)) Then
            Dim value1 As Single
            Value1 = values(0)
        End If
        ' OUT2 readings
        If Not IsEmpty(values(1)) Then
            Dim value2 As Single
            Value2 = values(1)
        End If
    End If
    ' Post-processing
    Call DisConnect
End Sub

' Preparation method
Private Sub Connect()
    ' Generate CaoEngine
    Set caoEng = New CaoEngine
    ' Generate CaoWorkspace
    Set caoWs = caoEng.AddWorkspace("Workspace", "")
    ' Generate CaoController
    Set caoCtrl = caoWs.AddController("LKG3000", _
                                      "CaoProv.KEYENCE.LK-G3000", _
                                      "", _
                                      "Conn=COM:1,Timeout=1000")
    ' Generate CaoVariable
    Set caoVrl = caoCtrl.AddVariable("@CALCDATA")
End Sub
' Post-processing method
Private Sub DisConnect()
    ' Remove CaoVariable from CaoController
    Call caoCtrl.variables.Remove(caoVrl.Index)
    ' Clear CaoVariable
    Set caoVrl = Nothing
    ' Remove CaoController from CaoWorkspace
    Call caoWs.Controllers.Remove(caoCtrl.Index)
```

```
    ' Clear CaoController
    Set caoCtrl = Nothing
    ' Remove CaoWorkspace from CaoEngine
    Call caoEng.Workspaces.Remove(caoWs.Index)
    ' Clear CaoWorkspace
    Set caoWs = Nothing
    ' Clear CaoEngine
    Set caoEng = Nothing
End Sub
```

4.1.1.1. Connection

To connect to LK-G3000, perform the following steps:

(1) Prepare a variable to hold the object. The objects required to connect to the controller are CaoEngine object, CaoWorkspace object, and CaoController object. CaoWorkpace object does not need to have a variable to obtain CaoController object from CaoWorkspaces. You will also need a CaoVariable for accessing the variable. The following is a sample code for VB6.

```
    ' Variables for CaoEngine Objects
    Dim caoEng As CaoEngine
    ' Variables for CaoWorkspace Objects
    Dim caoWs As CaoWorkspace
    ' Variables for CaoController Objects
    Dim caoCtrl As CaoController
    ' Variables for CaoVariable Objects
    Dim caoVrl As CaoVariable
```

(2) Creates a CaoEngine object. CaoEngine object is generated using the New keyword.

```
    ' Generate CaoEngine
    Set caoEng = New CaoEngine
```

(3) Gets or generates a CaoWorkspace object. When you create a CaoEngine object, it defaults to one CaoWorkspaces object and one object. The following is a sample code/default CaoWorkspace for creating a new CaoWorkspace.

```
    ' Generate CaoWorkspace
    Set caoWs = caoEng.AddWorkspace("Workspace", "")
```

(4) Generate a CaoController object. To generate a CaoController object, set the provider name to use and the parameters to use. For LK-G3000 providers, optionally specify the COM port to connect to and how long to wait for a response from the device. The following is a code example:

```
    ' Generate CaoController
    Set caoCtrl = caoWs.AddController("LKG3000", _
                                      "CaoProv.KEYENCE.LK-G3000", _
                                      "", _
                                      "Conn=COM:1,Timeout=1000")
```

(5) Creates a CaoVariable. Create a CaoVariable for the variable you want to retrieve. The following are

examples of codes that generate variable objects that access OUT1 and OUT2 measurements.

```
' Generate CaoVariable
Set caoVrl = caoCtrl.AddVariable("@CALCDATA")
```

4.1.1.2. Changing the operation mode of the main unit

When the operation mode of LK-G3000 main unit is "communication mode", the measured data cannot be acquired. The operating mode must be changed by SetMode Commands. To change the operating mode, execute CaoController object's Execute method.

```
' Execute this command if the operating mode of the device is not Normal mode.
Call caoCtrl.Execute("SetMode", 0)
```

4.1.1.3. Get OUT1 and OUT2 readings

To get OUT1 and OUT2 metrics, see Value Properties for CaoVariable Objects. Variables must be provided for each measurement. The following is a code example:

```
' Retrieving Values
Dim values As Variant
Values = caoVrl.value

If Not IsEmpty(values) Then
    ' OUT1 readings
    If Not IsEmpty(values(0)) Then
            Dim value1 As Single
            Value1 = values(0)
    End If
    ' OUT2 readings
    If Not IsEmpty(values(1)) Then
            Dim value2 As Single
            Value2 = values(1)
    End If
End If
```

4.1.1.4. Disconnect

To disconnect from the controller, you can erase the generated objects and delete the objects that you want to erase from the collection class that manages the objects. The following is a code example:

```
' Remove CaoController from CaoWorkspace
Call caoWs.Controllers.Remove(caoCtrl.Index)
' Clear CaoController
Set caoCtrl = Nothing
' Remove CaoWorkspace from CaoEngine
Call caoEng.Workspaces.Remove(caoWs.Index)
' Clear CaoWorkspace
Set caoWs = Nothing
' Clear CaoEngine
Set caoEng = Nothing
```

4.2. Sample programming for obtaining stored data in a data storage

The following example shows a sample program that stores data in data storage and reads the accumulated data in data storage. Table 4-2 describes the requirements of the sample program, and Figure 4-1 describes the flow of the sample program.

Table 4-2 Sample program requirements for acquiring accumulated data

Requirements	Description
Host	Connect via RS-232C communication
	The host port number is 1.
Process Description	Starts data accumulation in data storage.
	Finish data accumulation in data storage.
	Acquires the accumulated state of data storage.
	Acquires the data storage accumulated data.
	Clear data storage accumulated data.

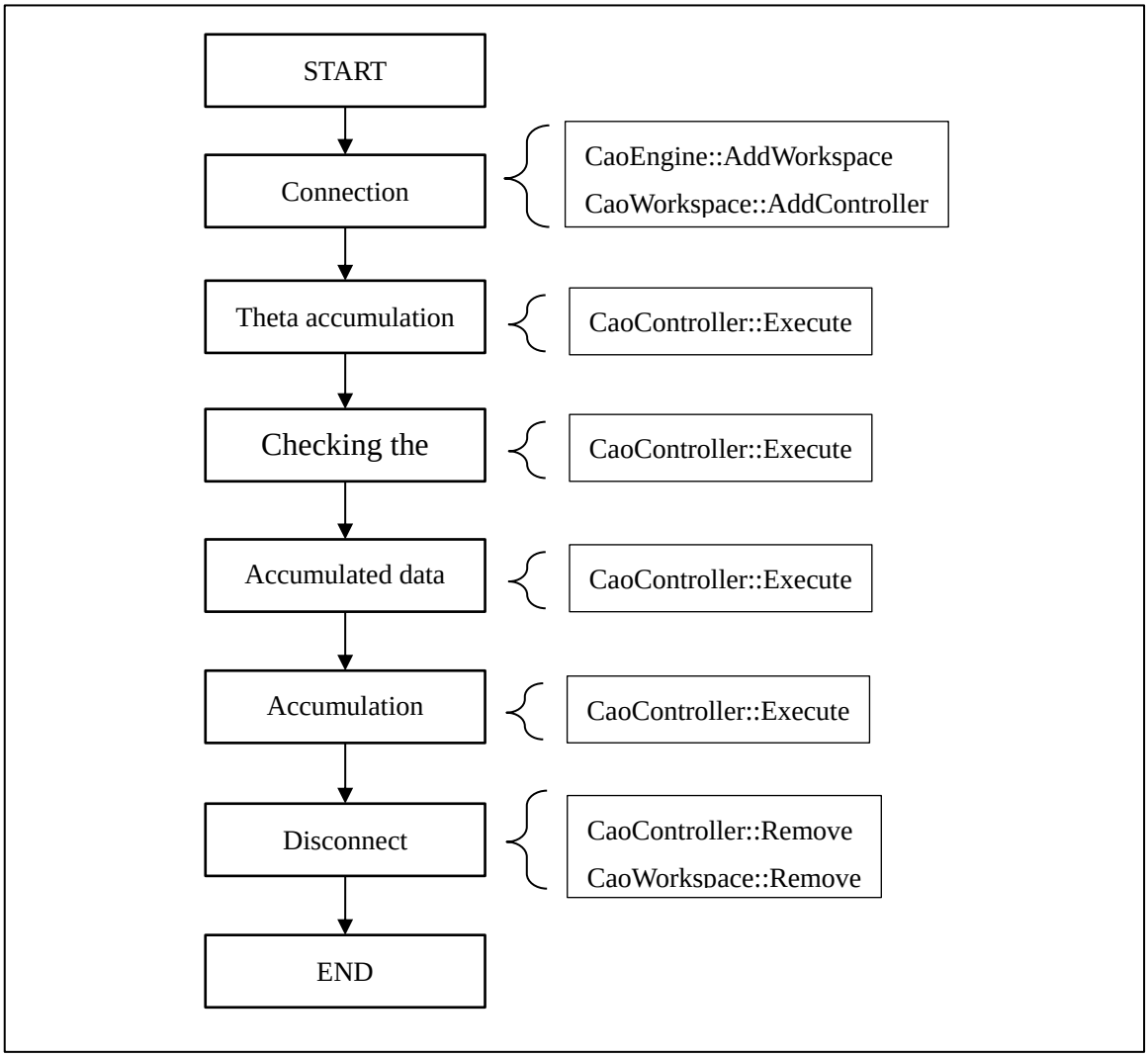


Figure 4-2 Flow of acquiring accumulated data for data storage

Specific codes are given in the following sections.

4.2.1. Sample program

The following is an overview of the sample program.

Sample	DataStorageSample.cs

```
' When using Sleep function
Private Declare Sub Sleep Lib "kernel32" (ByVal ms As Long)

' Object
Dim caoEng As CaoEngine
Dim caoWs As CaoWorkspace
Dim caoCtrl As CaoController
Dim caoVrl As CaoVariable
Private Sub Main()
        ' Connection
        Call Connect
```

```
        ' Execute this command if the operating mode of the device is not communication mode.
        Call caoCtrl.Execute("SetMode", 1)

        ' Configuring Data Storage
        Dim param As Variant
        Param = Array(1, 10, 1)
        Call caoCtrl.Execute("SetDataStorage", param)

        ' Execute this command if the operating mode of the device is not Normal mode.
        Call caoCtrl.Execute("SetMode", 0)
        ' Starts data accumulation in data storage
        Call caoCtrl.Execute("StartDataStorage")
        ' Wait for 0.5 seconds
        Sleep 500

        ' Finish data accumulation in data storage
        Call caoCtrl.Execute("StopDataStorage")
        ' Acquires the accumulated state of data storage.
        Dim statusvalue As Variant
        Statusvalue = caoCtrl.Execute("GetDataStorageStatus")
        If Not IsEmpty(statusvalue) Then
                ' Indicates whether or not data is being accumulated.
                Dim isAccumulation As BooleanisAccumulation = statusvalue(0)
                'Number of items stored in OUT1
                Dim dataCnt1 As Integer
                DataCnt1 = statusvalue(1)
                'Number of items stored in OUT2
                Dim dataCnt2 As Integer
                DataCnt2 = statusvalue(2)
        End If

        ' Acquire accumulated data in data storage
        Dim datavalue As Variant
        Datavalue = caoCtrl.Execute("GetDataStorageData", 1)
        If Not IsEmpty(datavalue) Then
                Dim i As Integer
                For i = LBound(datavalue) To (UBound(datavalue) - 1)
                        If Not IsEmpty(datavalue(i)) Then
                                Dim accumulationData As Single
                                AccumulationData = datavalue(i)
                        End If
                Next i
        End If

        ' Clear accumulated data in data storage
        Call caoCtrl.Execute("ClearDataStorage")

        ' Disconnect
        Call DisConnect

End Sub

' Connection method
Private Sub Connect()
        ' Generate CaoEngine
```

```
        Set caoEng = New CaoEngine
        ' Generate CaoWorkspace
        Set caoWs = caoEng.AddWorkspace("Workspace", "")
        ' Generate CaoController
        Set caoCtrl = caoWs.AddController("LKG3000", _
                                          "CaoProv.KEYENCE.LK-G3000", _
                                          "", _
                                          "Conn=COM:1,Timeout=1000")
End Sub

' Disconnect method
Private Sub DisConnect()
        ' Remove CaoController from CaoWorkspace
        Call caoWs.Controllers.Remove(caoCtrl.Index)
        ' Clear CaoController
        Set caoCtrl = Nothing
        ' Remove CaoWorkspace from CaoEngine
        Call caoEng.Workspaces.Remove(caoWs.Index)
        ' Clear CaoWorkspace
        Set caoWs = Nothing
        ' Clear CaoEngine
        Set caoEng = Nothing
End Sub
```

4.2.1.1. Connection

To connect to LK-G3000, perform the following steps:

(1) Prepare a variable to hold the object. The objects required to connect to the controller are CaoEngine object, CaoWorkspace object, and CaoController object. CaoWorkpace object does not need to have a variable to obtain CaoController object from CaoWorkspaces. You will also need a CaoVariable for accessing the variable. The following is a sample code for VB6.

```
' Variables for CaoEngine Objects
Dim caoEng As CaoEngine
' Variables for CaoWorkspace Objects
Dim caoWs As CaoWorkspace
' Variables for CaoController Objects
Dim caoCtrl As CaoController
' Variables for CaoVariable Objects
Dim caoVrl As CaoVariable
```

(2) Creates a CaoEngine object. CaoEngine object is generated using the New keyword.

```
' Generate CaoEngine
Set caoEng = New CaoEngine
```

(3) Gets or generates a CaoWorkspace object. When you create a CaoEngine object, it defaults to one CaoWorkspaces object and one object. The following is a sample code/default CaoWorkspace for creating a new CaoWorkspace.

```
' Generate CaoWorkspace
```

```
Set caoWs = caoEng.AddWorkspace("Workspace", "")
```

(4) Generate a CaoController object. To generate a CaoController object, set the provider name to use and the parameters to use. For LK-G3000 providers, optionally specify the COM port to connect to and how long to wait for a response from the device. The following is a code example:

```
' Generate CaoController
Set caoCtrl = caoWs.AddController("LKG3000", _
                                  "CaoProv.KEYENCE.LK-G3000", _
                                  "", _
                                  "Conn=COM:1,Timeout=1000")
```

4.2.1.2. Configuring Data Storage

If required, configure LK-G3000 unit's data-storage settings. In the sample program, the target OUT is set to "OUT1", the data accumulation count is set to "10", and the accumulation cycle is set to "Sampling cycle × 2". The sampling period is 200μs, which is the default of LK-G3000. For more information about the commands, see 3.3.3.5.1. SetDataStorage Commands. The following is a code example:

```
' Execute this command if the operating mode of the device is not communication mode.
Call caoCtrl.Execute("SetMode", 1)

' Configuring Data Storage
Dim param As Variant
Param = Array(1, 10, 1)
Call caoCtrl.Execute("SetDataStorage", param)
```

4.2.1.3. Start of data accumulation to data storage

Starts accumulation in data storage. 4.2.1.2. The measured value of the target OUT specified in the data storage setting is accumulated for the specified number of data accumulations. If the target OUT is none, this command fails. The following is a code example:

```
' Execute this command if the operating mode of the device is not Normal mode.
Call caoCtrl.Execute("SetMode", 0)

' Starts data accumulation in data storage
Call caoCtrl.Execute("StartDataStorage")
```

4.2.1.4. End of data accumulation in data storage

Ends accumulation in data storage. If the accumulated number specified in 4.2.1.2 "Data storage setting" is accumulated, the accumulation ends automatically. The following is a code example:

```
' Finish data accumulation in data storage
Call caoCtrl.Execute("StopDataStorage")
```

4.2.1.5. Acquires the accumulated state of data storage.

To get the accumulated state of the data storage, refer to Execute method of CaoCaoController object. The

following is a code example:

```
' Acquires the accumulated state of data storage.
Dim statusvalue As Variant
Statusvalue = caoCtrl.Execute("GetDataStorageStatus")
If Not IsEmpty(statusvalue) Then
    ' Indicates whether or not data is being accumulated.
    Dim isAccumulation As Boolean
    IsAccumulation = statusvalue(0)
    'Number of items stored in OUT1
    Dim dataCnt1 As Integer
    DataCnt1 = statusvalue(1)
    'Number of items stored in OUT2
    Dim dataCnt2 As Integer
    DataCnt2 = statusvalue(2)
End If
```

4.2.1.6. Acquire accumulated data in data storage

To retrieve accumulated data for data storage, refer to Execute method of CaoCaoController object. The following is a code example:

```
' Acquire accumulated data in data storage
Dim datavalue As Variant
Datavalue = caoCtrl.Execute("GetDataStorageData", 1)
If Not IsEmpty(datavalue) Then
    Dim i As Integer
    For i = LBound(datavalue) To (UBound(datavalue) - 1)
        If Not IsEmpty(datavalue(i)) Then
            Dim accumulationData As Single
            AccumulationData = datavalue(i)
        End If
    Next i
End If
```

4.2.1.7. Clear accumulated data in data storage

To clear the accumulated data in the data storage, refer to Execute method of CaoCaoController object. The following is a code example:

```
' Clear accumulated data in data storage
Call caoCtrl.Execute("ClearDataStorage")
```

4.2.1.8. Disconnect

To disconnect from the controller, you can erase the generated objects and delete the objects that you want to erase from the collection class that manages the objects. The following is a code example:

```
' Remove CaoController from CaoWorkspace
Call caoWs.Controllers.Remove(caoCtrl.Index)
' Clear CaoController
Set caoCtrl = Nothing
' Remove CaoWorkspace from CaoEngine
Call caoEng.Workspaces.Remove(caoWs.Index)
```

```
' Clear CaoWorkspace
Set caoWs = Nothing
' Clear CaoEngine
Set caoEng = Nothing
```

5. LK-G3000 Provider Error Codes

This provider has the following unique error codes masked with the 0x8011****. (Refer to Table 5-1 Unique Error Codes Table.)

For information about common ORiN2 errors, see the Error Codes section of ORiN2 Programming Guide (Link).C:¥ORiN2¥CAO¥Doc¥ORiN2_ProgrammersGuide_en.pdf

Table 5-1 Unique Error Codes

Error Number	Description
0x80110001	There is no Conn optional specification for AddContoller.
0x80110002	Conn option-specification method during AddContoller is incorrect.
0x80110003	Conn option-value at AddContoller is out of scope.
0x80110004	Timeout optional specification during AddContoller is incorrect.
0x80110005	Timeout option-value at AddContoller is out of scope.
0x80110006	The argument when the extended command is executed does not match the command specification.
0x80110007	An unexpected data was received from the device when the extended command was executed.

This provider also masks the error code from the device with "0x8010****" and returns it. The following device error codes exist: (Refer to Table 5-2 Device Error Codes)

Table 5-2 Device Error Codes

Error Number	Description
0x80100000	Command error.
0x80100001	Status error.
0x80100020	Command length error.
0x80100021	Parameter number error.
0x80100022	Parameter out of range error.
0x80100088	Timeout error
0x80100099	Other.

6. Appendix

Appendix A. Communication protocol command correspondence table

CaoControllre::Execute method	Device Communication Protocol Commands
SetMode (Normal mode set)	R0
SetMode (when communication is set)	Q0
GetCalcData	Ma[2]
SetTiming	Tp[3]
SetZero (when set to ON)	Va
SetZero (when set to OFF)	Wa
SetReset	VR
SetPanelLock	KL
SetProgramNo	PW
GetProgramNo	PR
GetFigureData	DO
ClearFigureData	DQ
StartDataStorage	AS
StopDataStorage	AP
ClearDataStorage	AQ
GetDataStorageData	AO
GetDataStorageStatus	AN
SetPanel	DC
GetPanel	DR
SetTolerance	SW,LM
GetTolerance	SR,LM
SetAbleMode	SW,HA,M
GetAbleMode	SR,HA,M
SetAbleMinMax	SW,HA,R
GetAbleMinMax	SR,HA,R
SetMeasureMode	SW,HB

[2] a : OUT specification
[3] p : ON/OFF specification

GetMeasureMode	SR,HB
SetNumAlarm	SW,HC,N
SetAlarmLevel	SW,HC,L
GetAlarm	SR,HC
StartABLE	SW,HD,S
StopABLE	SW,HD,P
CancelABLE	SW,HD,C
SetReflectionMode	SW,HE
GetReflectionMode	SR,HE
SetCalcMethod	SW,OA
GetCalcMethod	SR,OA
SetScaling	SW,OB
GetScaling	SR,OB
SetFilter	SW,OC
GetFilter	SR,OC
SetTriggerMode	SW,OE,M
GetTriggerMode	SR,OE,M
SetOffset	SW,OF
GetOffset	SR,OF
SetAnalogScaling	SW,OH
GetAnalogScaling	SR,OH
SetCalcMode	SW,OD
GetCalcMode	SR,OD
SetDisplayUnit	SW,OG
GetDisplayUnit	SR,OG
SetAnalogThrough	SW,OI
GetAnalogThrough	SR,OI
SetDataStorage	SW,CI
GetDataStorage	SR,CI
SetSamplingCycle	SW,CA
GetSamplingCycle	SR,CA
SetMutualInterPrev	SW,CB
GetMutualInterPrev	SR,CB
SetTimingSync	SW,CC
GetTimingSync	SR,CC
SetToleCompOutputFormat	SW,CD

GetToleCompOutputFormat	SR,CD
SetStorobeTime	SW,CE
GetStorobeTime	SR,CE

CaoVariable::Value Properties	Device Communication Protocol Commands
@CALCDATA::get	M0