CM-MISDK (C# version)

Reference Manual

[Rev.1.04]

KONICA MINOLTA

• Official application names used in this manual

Abbreviated in this manual as:	Official name
Windows 7	Microsoft® Windows® 7 Business Operating System
Windows 8.1	Microsoft® Windows® 8.1
Windows 10	Microsoft® Windows® 10
C#	Microsoft® Visual C#
C++	Microsoft® Visual C++
VB	Microsoft® Visual Basic .NET

• Trademarks

Microsoft and Windows are registered trademarks of Microsoft Corporation in the United States and/or other countries.

All other corporate and product names mentioned in this manual are properties of their respective owners.

• Notes on this manual

No part of this manual may be reproduced without prior permission.

The contents of this manual are subject to change without prior notice.

Notwithstanding the preceding Konica Minolta assumes no liability for any result obtained from the use of this manual.

CM-SDK (C# Version) Reference Manual

[Rev.1.04]

Contents

Introduction

The SDK is a tool for developing a PC application for instruments that measure object color. This manual describes how to use the SDK.

Application developers are assumed to be using C#, so the programming methods are described using C#. This SDK does not support multithreading.

1. System Environment

The following table lists the verified development environments.

Supported operating systems	Windows 7(x86), Windows 7(x64) Windows 8.1(x86), Windows 8.1(x64) Windows 10(x86), Windows 10(x64)
Operating environment	.NET Framework 4.5
Development environment	Visual Studio 2012 Visual Studio 2013 Visual Studio 2015 Visual Studio 2017 Visual Studio 2019
Development languages	VC++, VC#, VB.NET
Controllable instruments	• CM-M6 • CM-25cG • CM-26dG • CM-26d • CM-25d • CM-23d • CM-3630A • CM-36dG(CM-36dGV) • CM-36d

2. Installing/Uninstalling the SDK

2.1. Installation

Install this SDK according to the following procedure.

(1) Expand the contents of the provided “cm-misdk_verXXXrX.zip” file to the desired location on the PC.

(2) “cm-misdk_verXXXrX” will be displayed and the following folders will be present when the files are expanded.

No.	Folder name	Overview
1	SDK	CM-MISDK files
2	Manual	CM-MISDK Reference Manual
3	Driver	USB driver for instrument
4	SampleCode	Sample code files
5	License	License agreement

(3) To develop applications using the SDK, configure the appropriate settings so that the following files in the above SDK folder can be accessed from the development environment. For details, refer to “3. SDK Overview”.

No.	Folder name	Overview
1	CMMISDK_x86.dll	SDK DLL file (32-bit version)
2	CMMISDK_x64.dll	SDK DLL file (64-bit version)
3	CMMISDK.NET.dll	C# version DLL file

2.2. Uninstallation

Delete the cm-misdk_verXXXrX folder and manually copied folders.

3. SDK Overview

3.1 Function list

The following processing can be performed with the SDK.

Connect/disconnect		
	Connect	Connect to the instrument
	Disconnect	Ends the connection with the instrument.
	GetDeviceList	Obtains the list of connectible instruments.
	GetInstrumentInfo	Obtains the instrument information.
	GetSDKVersion	Obtains the SDK version.
Calibrate/measure		
	GetCalibrationStatus	Obtains the calibration status.
	PerformZeroCalibration	Executes zero calibration.
	PerformWhiteCalibration	Executes white calibration.
	PerformGlossCalibration	Executes gloss calibration.
	PerformUserCalibration	Executes user calibration.
	PerformMeasurement	Executes the measurement.
	PollingMeasurement	Judges if the measurement is complete.
	CancelMeasurement	Cancels measurement.
	ReadLatestData	Reads the latest data.
	ReadAllLatestData	Reads all of the latest data.
	SetWhiteCalibrationData	Sets the white calibration plate data.
	GetWhiteCalibrationData	Obtains the white calibration plate data.
	SetGlossCalibrationData	Sets the gloss calibration plate data.
	GetGlossCalibrationData	Obtains the gloss calibration plate data.
	SetUserCalibrationData	Sets the user calibration plate data.
	GetUserCalibrationData	Obtains the user calibration plate data.
	SetUserCalibrationEnable	This function enables and disables user calibration.
	GetUserCalibrationEnable	This function obtains the user calibration enabled or disabled state.
	SetTriggerMode	Sets the trigger mode.
	GetTriggerMode	Obtains the trigger mode.
	DeleteTriggerData	Deletes the trigger mode data.
	IsThereTriggerData	Obtains the availability of trigger mode data.
	GetZeroCalibrationDate	Obtains the zero calibration date and time.
	GetWhiteCalibrationDate	Obtains the white calibration date and time.
	GetGlossCalibrationDate	Obtains the gloss calibration date and time.
	GetUserCalibrationDate	Obtains the user calibration date and time.
	ClearUvAdjustInfo	Clears various data for fluorescence adjustment.
	SetProfileForUvAdjust	Sets the fluorescence adjustment profile.
	GetProfileForUvAdjust	Obtains the fluorescence adjustment profile.
	SetWiForUvAdjust	Sets the WI for fluorescence adjustment.
	GetWiForUvAdjust	Obtains the WI for fluorescence adjustment.
	SetTintForUvAdjust	Sets the Tint for fluorescence adjustment.
	GetTintForUvAdjust	Obtains the Tint for fluorescence adjustment.
	SetIsoBrightnessForUvAdjust	Sets the ISO brightness for fluorescence adjustment.
	GetIsoBrightnessForUvAdjust	Obtains the ISO brightness for fluorescence adjustment.
	SetGanzForUvAdjust	Sets the Ganz & Griesser for fluorescence adjustment.
	GetGanzForUvAdjust	Obtains the Ganz & Griesser for fluorescence adjustment.
	SetDataForUvAdjust	Sets the data for fluorescence adjustment.
	GetDataForUvAdjust	Obtains the data for fluorescence adjustment.

	PerformUvAdjust	Executes fluorescence adjustment and sets the coefficient.
	PerformUvAdjustUsingData	Executes fluorescence adjustment and sets the coefficient.
	ClearCoefForUvAdjust	Clears the fluorescence coefficient in the instrument.
	SetCoefForUvAdjust	Sets the fluorescence coefficient.
	GetCoefForUvAdjust	Obtains the fluorescence coefficient.
	SetOutputMinus	Enables and disables output of negative values.
	GetOutputMinus	Obtains the output of negative values enabled or disabled state.
Measurement Conditions		
	SetMeasurementArea	Sets the measurement area.
	GetMeasurementArea	Obtains the measurement area.
	SetMeasurementType	Sets the measurement type.
	GetMeasurementType	Obtains the measurement type.
	SetMeasurementAngle	Sets the measurement angle.
	GetMeasurementAngle	Obtains the measurement angle.
	SetTiltDetection	Sets tilt detection.
	GetTiltDetection	Obtains tilt detection.
	SetMeasurementMode	Sets the measurement mode.
	GetMeasurementMode	Obtains the measurement mode.
	SetSpecularComponent	Sets the specular component.
	GetSpecularComponent	Obtains the specular component.
	SetUv	Sets the UV condition.
	GetUv	Obtains the UV condition.
	SetAutoAverageTimes	Sets the number of times of automatic averaging.
	GetAutoAverageTimes	Obtains the number of times of automatic averaging.
	SetManualAverageTimes	Sets the number of times of manual averaging.
	GetManualAverageTimes	Obtains the number of times of manual averaging.
	SetManualAverageSaveMode	Sets the manual averaging save method.
	GetManualAverageSaveMode	Obtains the manual averaging save method.
	SetCondSMC	Sets the SMC conditions.
	GetCondSMC	Obtains the SMC conditions.
Display conditions		
	SetDisplayType	Sets the display type.
	GetDisplayType	Obtains the display type.
	SetObserverAndIlluminant	Sets the observation field and illuminant.
	GetObserverAndIlluminant	Obtains the observation field and illuminant.
	SetUserIlluminant	Sets the user illuminant data.
	GetUserIlluminant	Obtains the user illuminant data.
	SetColorSpace	Sets the color space.
	GetColorSpace	Obtains the color space.
	SetEquation	Sets the color difference equation.
	GetEquation	Obtains the color difference equation.
	SetCustomItem	Sets the custom items.
	GetCustomItem	Obtains the custom items.
	SetDisplayData	Sets the irradiation direction of data to display.
	GetDisplayData	Obtains the irradiation direction of data to display.
	SetUserEquation	Sets the user index.
	GetUserEquation	Obtains the user index.
Data		
	SetActiveTarget	Sets the active target.

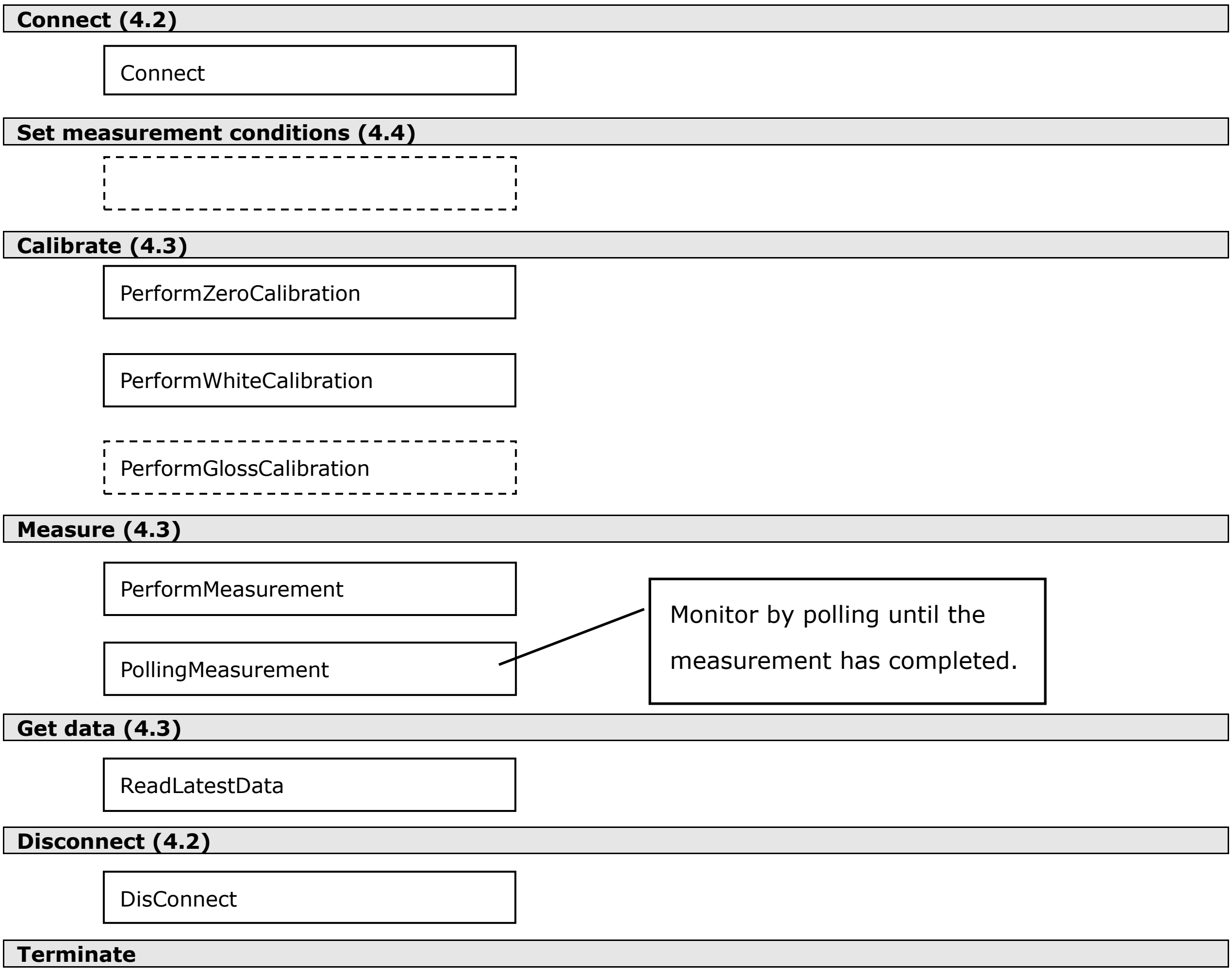
	GetActiveTarget	Obtains the active target.
	GetTargetNumberList	Obtains the list of saved target numbers.
	GetTargetNumberList2	Obtains the list of target numbers when the display filter is applied.
	DeleteTargetData	Deletes the target data.
	SetTargetData	Sets the target data.
	GetTargetData	Obtains the target data.
	GetAllTargetData	Obtains the target data.
	SetToleranceForTarget	Sets the target tolerance data.
	GetToleranceForTarget	Obtains the target tolerance data.
	SetParametricForTarget	Sets the parametric coefficient for a target color.
	GetParametricForTarget	Obtains the parametric coefficient for a target color.
	SetTargetFilter	Sets the target filter conditions.
	GetTargetFilter	Obtains the target filter conditions.
	SetTargetProtect	Sets target protection.
	GetTargetProtect	Obtains target protection.
	GetSampleCount	Obtains the number of saved measurement values.
	DeleteSampleData	Deletes measurement value data.
	GetSampleData	Obtains the measurement data.
	GetAllSampleData	Obtains the measurement data.
Others		
	SetActiveGroup	Sets the active group.
	GetActiveGroup	Obtains the active group.
	SetGroupName	Sets the group name.
	GetGroupName	Obtains the group name.
	SetMultipleGroupName	Sets multiple group names in batch.
	GetMultipleGroupName	Obtains multiple group names in batch.
	SetTolerance	Sets the default tolerance.
	GetTolerance	Obtains the default tolerance.
	SetParametric	Sets the default parametric coefficient.
	GetParametric	Obtains the default parametric coefficient.
	SetWarningLevel	Sets the warning level.
	GetWarningLevel	Obtains the warning level.
	SetInstrumentMode	Sets the instrument mode.
	GetInstrumentMode	Obtains the instrument mode.
	SetUserType	Sets the user type.
	GetUserType	Obtains the user type.
	SetAdminPassword	Sets the administrator password.
	GetAdminPassword	Obtains the administrator password.
	SetAutoPrint	Sets automatic printing.
	GetAutoPrint	Obtains the automatic printing setting.
	SetBrightness	Sets the brightness of the display.
	GetBrightness	Obtains the brightness of the display.
	SetScreenDirection	Sets the direction of the display.
	GetScreenDirection	Obtains the direction of the display.
	SetSound	Sets the sound.
	GetSound	Obtains the sound.
	SetCalibrationInterval	Sets the calibration interval.
	GetCalibrationInterval	Obtains the calibration interval.
	SetAnnualCalibration	Sets the periodical calibration notice.
	GetAnnualCalibration	Obtains the periodical calibration notice.
	SetDateTime	Sets the date and time.
	SetDateFormat	Sets the date format.
	GetDateFormat	Obtains the date format.
	SetLanguage	Sets the display language.

	GetLanguage	Obtains the display language.
	SetAutoPowerOff	Sets power saving mode.
	GetAutoPowerOff	Obtains power saving mode.
	ClearJobInfo	Clears job information.
	SetJobInfo	Sets job information.
	GetJobInfo	Obtains job information.
	GetJobStepType	Obtains the step type of the job.
	SetJobStepForOperation	Sets an operation step of the job.
	GetJobStepForOperation	Obtains an operation step of the job.
	SetJobStepForResult	Sets a result step of the job.
	GetJobStepForResult	Obtains a result step of the job.
	SetJobImage	Sets job images.
	GetJobImage	Obtains job images.
	ResetToFactorySetting	Resets the settings to factory default.
	SetFinderEnable	This function enables and disables functions.
	GetFinderImage	Obtains the finder image.
	GetMeasurementImage	Obtains the finder image at the time of measurement.
	GetDetectedMask	Obtains the detected measurement area.

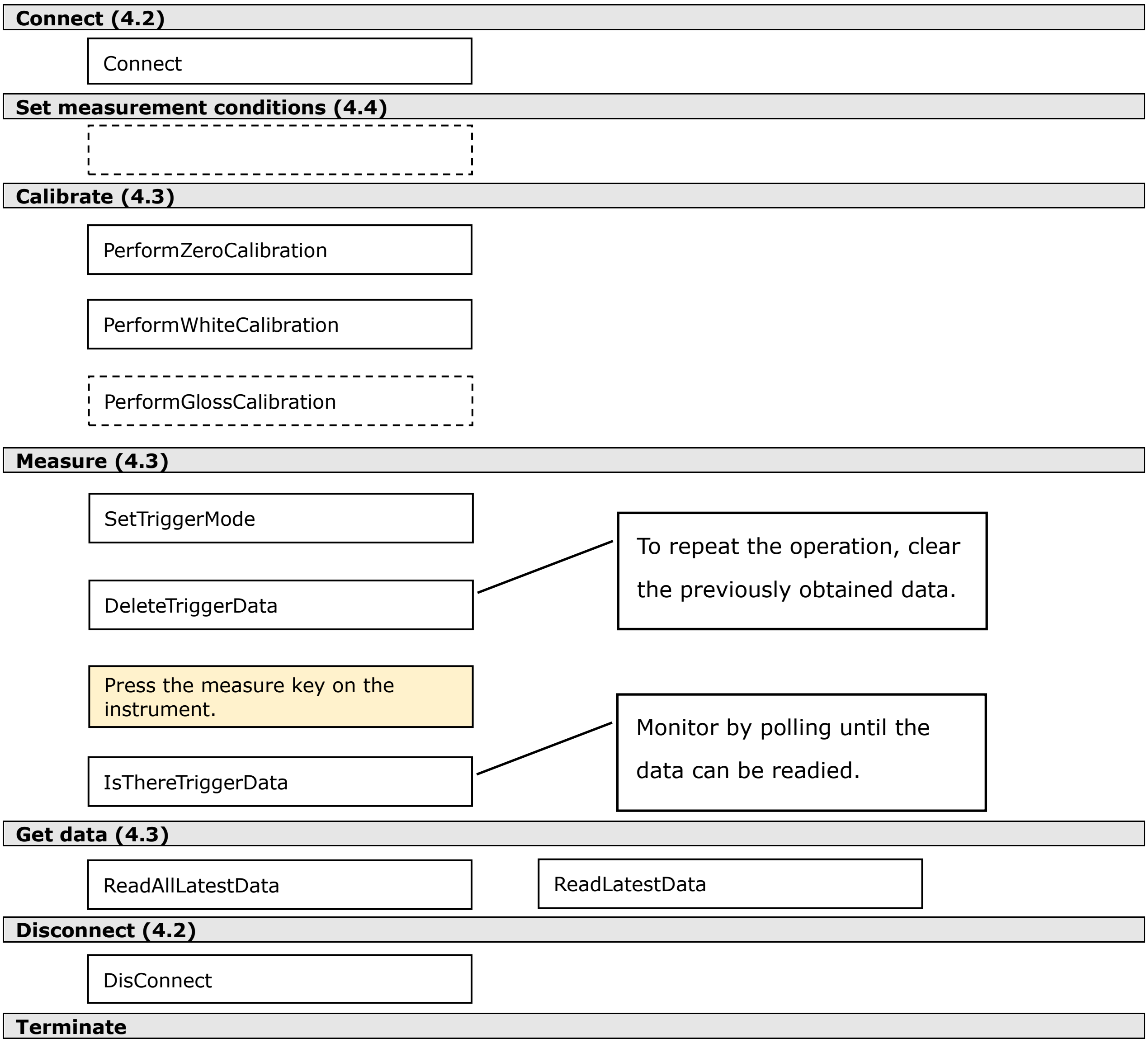
3.2 Basic processing flow

3.2.1 Measurements

3.2.1.1 Measurements using the SDK



3.2.1.2 Measurements using an instrument key



Connect (4.2)

Connect

Set measurement conditions (4.4)

Calibrate (4.3)

PerformZeroCalibration

PerformWhiteCalibration

PerformGlossCalibration

Measure (4.3)

SetTriggerMode

DeleteTriggerData

To repeat the operation, clear the previously obtained data.

Press the measure key on the instrument.

IsThereTriggerData

Monitor by polling until the data can be readied.

Get data (4.3)

ReadAllLatestData

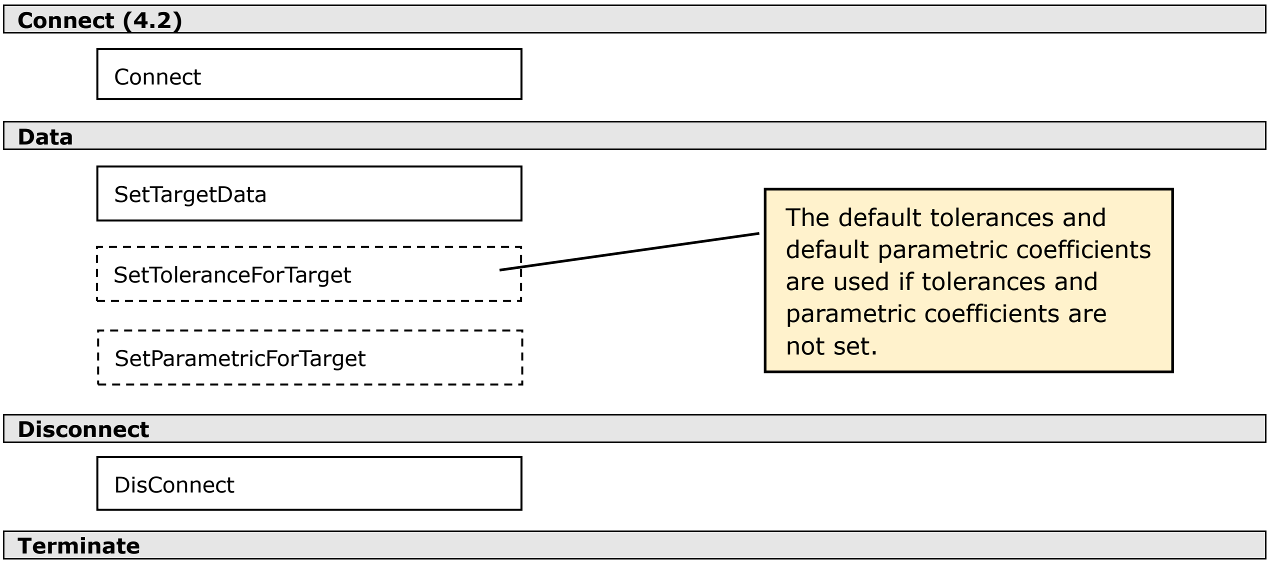
ReadLatestData

Disconnect (4.2)

DisConnect

Terminate

3.2.2 Target writing (when new)



Connect (4.2)

Connect

Data

SetTargetData

SetToleranceForTarget

SetParametricForTarget

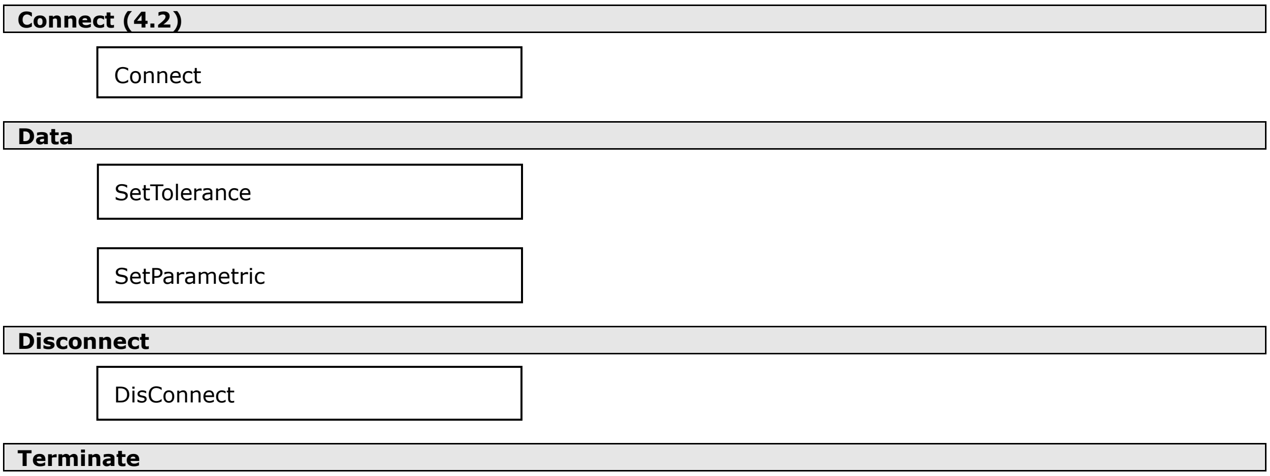
The default tolerances and default parametric coefficients are used if tolerances and parametric coefficients are not set.

Disconnect

DisConnect

Terminate

3.2.3 Default tolerance setting



Connect (4.2)

Connect

Data

SetTolerance

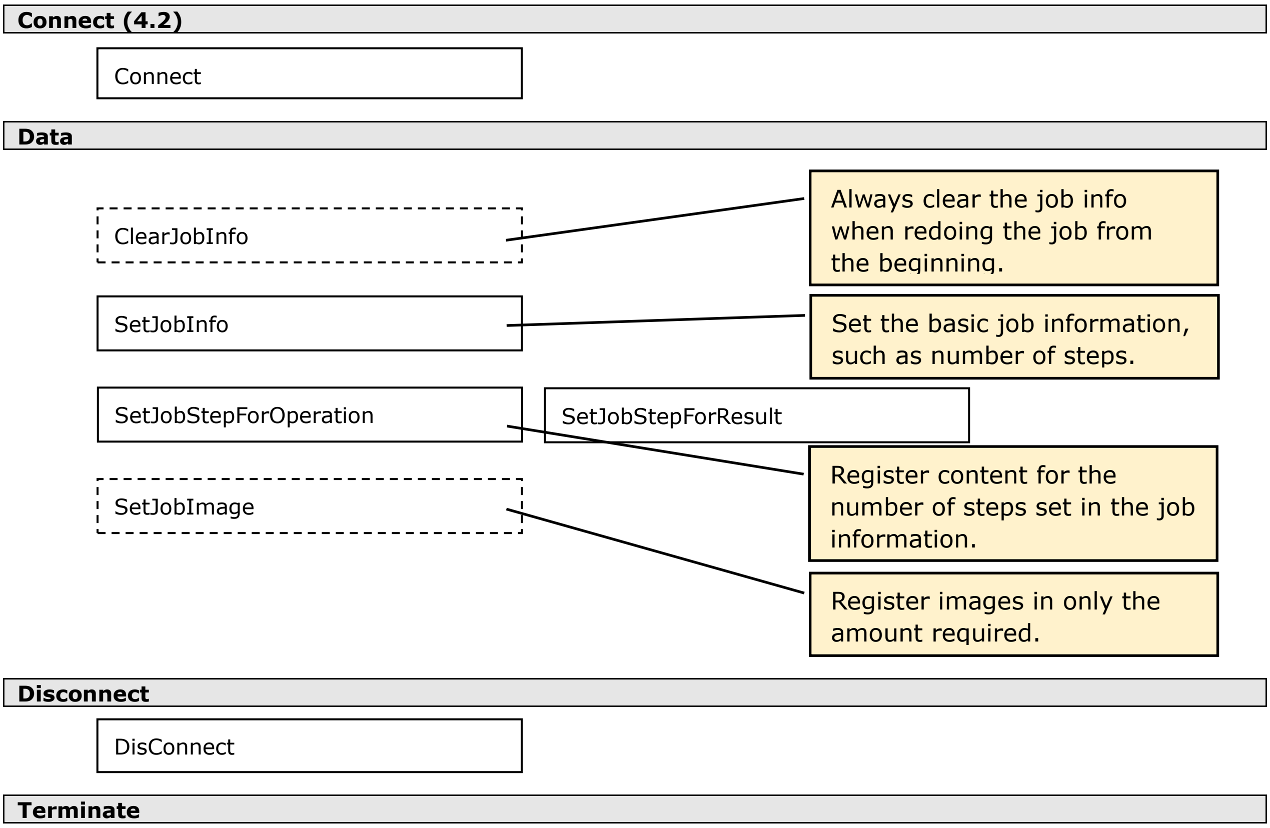
SetParametric

Disconnect

DisConnect

Terminate

3.2.4 Job name



Connect (4.2)

Connect

Data

ClearJobInfo

Always clear the job info when redoing the job from the beginning.

SetJobInfo

Set the basic job information, such as number of steps.

SetJobStepForOperation

SetJobStepForResult

Register content for the number of steps set in the job information.

SetJobImage

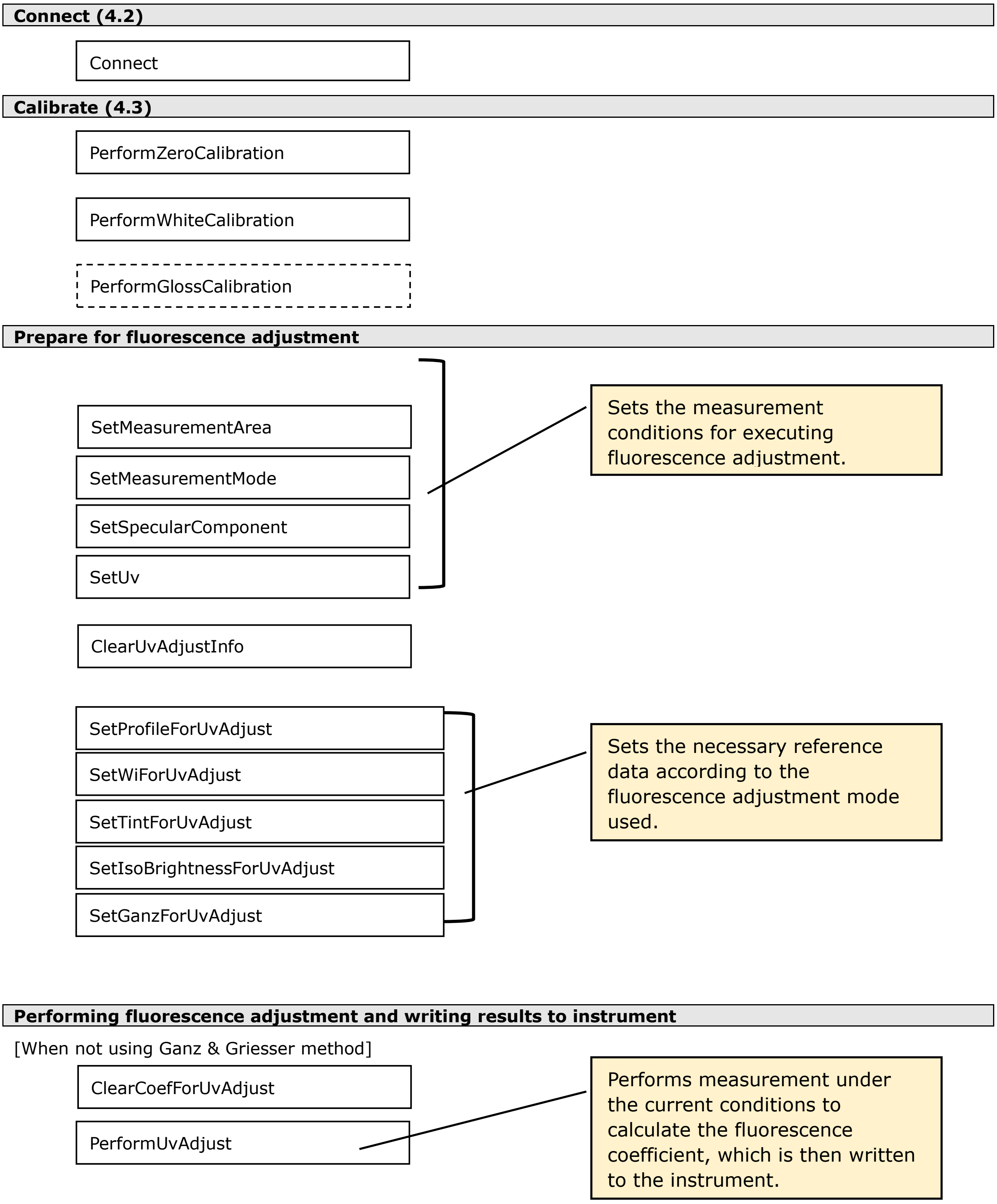
Register images in only the amount required.

Disconnect

DisConnect

Terminate

3.2.5 Fluorescence Adjustment



Connect (4.2)

Connect

Calibrate (4.3)

PerformZeroCalibration

PerformWhiteCalibration

PerformGlossCalibration

Prepare for fluorescence adjustment

SetMeasurementArea

SetMeasurementMode

SetSpecularComponent

SetUv

Sets the measurement conditions for executing fluorescence adjustment.

ClearUvAdjustInfo

SetProfileForUvAdjust

SetWiForUvAdjust

SetTintForUvAdjust

SetIsoBrightnessForUvAdjust

SetGanzForUvAdjust

Sets the necessary reference data according to the fluorescence adjustment mode used.

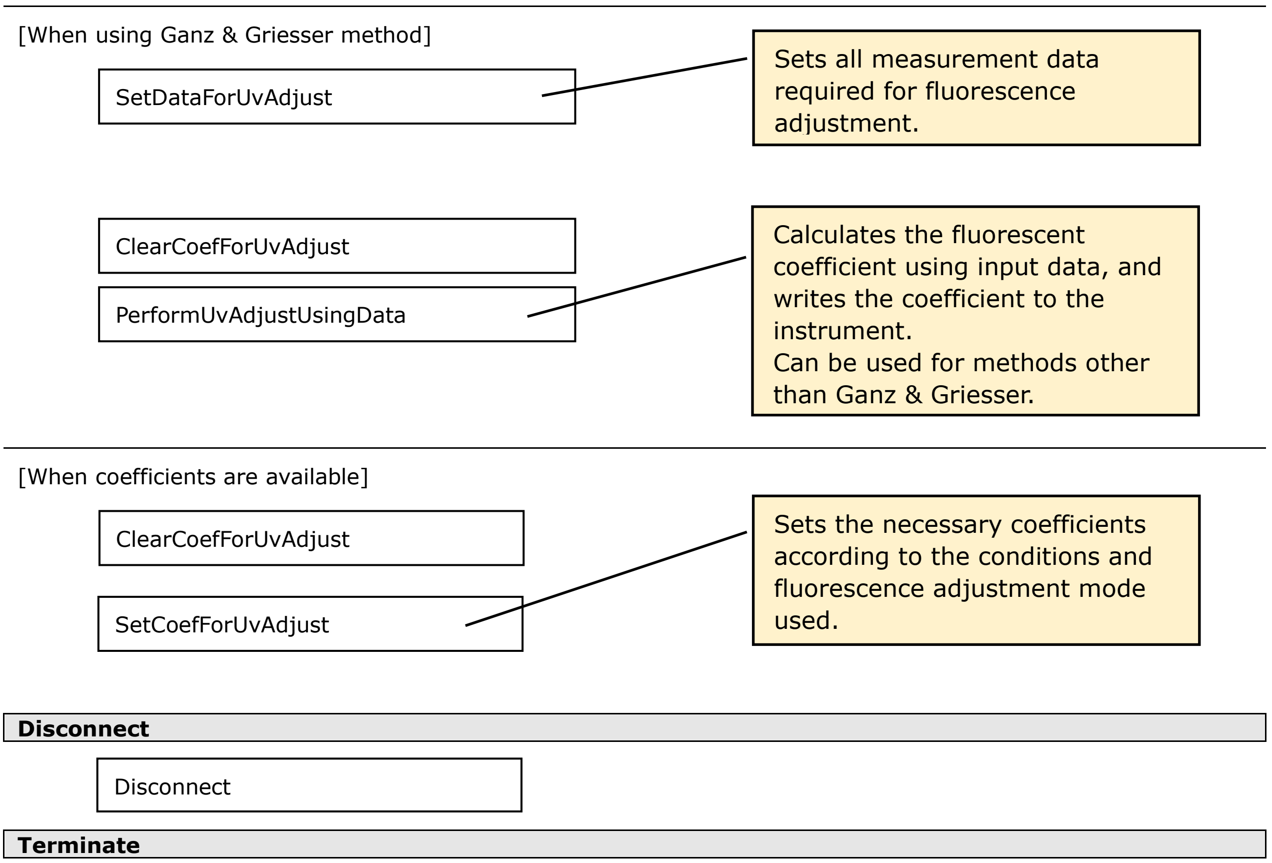
Performing fluorescence adjustment and writing results to instrument

[When not using Ganz & Griesser method]

ClearCoefForUvAdjust

PerformUvAdjust

Performs measurement under the current conditions to calculate the fluorescence coefficient, which is then written to the instrument.



[When using Ganz & Griesser method]

Sets all measurement data required for fluorescence adjustment.

Calculates the fluorescent coefficient using input data, and writes the coefficient to the instrument.
Can be used for methods other than Ganz & Griesser.

[When coefficients are available]

Sets the necessary coefficients according to the conditions and fluorescence adjustment mode used.

Disconnect

Terminate

3.3 How to create programs with the SDK

3.3.1 Using the SDK from a Development Environment

This section describes how to use the SDK with Visual Studio 2013 as an example.

(1) Create a C++ application project (referred to as "the project").

(2) Place "CMMISDK_***.dll" and "CMMISDK.NET.dll" in either of the following locations.

- Execution folder of the application that will be created
- Folder set as an environment path

(3) Add "CMMISDK.NET.dll" with "Add Reference".

(4) Create and build an application that uses the SDK.

3.3.2 Sample Code Overview

Three types of sample code have been prepared for this SDK.

(1) PerformMeasurement: Calibrate, measure, and get measurement data

(2) ReadSampleData: Get saved measurement data

(3) WriteTargetData: Write target data

Refer to the sample code for the specific implementation methods.

4. SDK Reference

4.1 Format of SDK functions

4.1.1 Format

The functions in the SDK are described using the following format.

Format:

Describes the format of the function.

Arguments:

Describes the arguments of the function.

Return Value:

Describes the return value that is returned when the function is used.

There are three types of return values.

Type	
Success	Returned when the processing was successful.
Error	Returned when the processing failed.
Warning	Returned when the processing was successful, but with restrictions.

Description:

Describes necessary information and precautions when using the function.

4.1.2 Return values

All methods return the following return value class.

ReturnMessage
Overview:
The class used as the return value of the methods.
Format:

```
class ReturnMessage
{
    Int32         errorCode;
    List<string>  errorMessage;
}
```

Variable:

Variable	Explanation
errorCode	Error code
errorMessage	Error message * If the error code indicates no error or if there is a warning, they are stored in this list.

Description:
Use the error code and message to identify the error.

4.2 Connect/disconnect

Connect: Connects the computer to the instrument connected to the specified virtual COM port.

Format:

ReturnMessage Connect(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Failed to connect to the instrument.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This method connects the computer to the instrument that is connected to the specified virtual COM port.
When 0 is specified for comPort (or when no argument is specified), GetDeviceList is automatically called and a connection is established with the instrument with the smallest COM port number.
Then specify 0 for comPort in the other methods to use the virtual COM port number of the instrument that was connected using Connect().

* When using a Bluetooth connection, you cannot set comPort to 0. You must specify comPort to make the connection.

DisConnect: Ends the communication with the instrument connected to the specified virtual COM port.

Format:

ReturnMessage DisConnect(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

If there was an error, turn off the power to the instrument and disconnect it.

GetDeviceList: Obtains the list of instruments connected to the PC.

Format:

ReturnMessage GetDeviceList(out Dictionary<Int32, string> deviceList, bool isAll=false)

Arguments:

<table>
<tr><th>Name</th><th>I/O</th><th>Explanation</th></tr>
<tr><td>deviceList</td><td>O</td><td>Instrument list<table><tr><td>key</td><td>COM port number</td></tr><tr><td>value</td><td>Model name (Unit number)</td></tr></table>E.g.) When a CM-25cG with unit number 1000001 is connected to COM3<table><tr><td>key</td><td>3</td></tr><tr><td>value</td><td>CM-25cG(1000001)</td></tr></table></td></tr>
<tr><td>isAll</td><td>I</td><td>Search target range<table><tr><td>false</td><td>Search USB only</td></tr><tr><td>true</td><td>Search both USB and Bluetooth</td></tr></table></td></tr>
</table>

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the PC.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This method obtains the list of instruments connected to the PC.
If isAll=true, the search will be performed on all available ports, so this processing will take some time.
To use USB only, search with isAll=false.

Note that instruments that have their COM ports opened using other software are not listed.

GetInstrumentInfo: Obtains the instrument information.

Format 1:

ReturnMessage GetInstrumentInfo(out InstrumentInfo info, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
instrumentInfo	O	Instrument information
comport	I	Virtual COM Port number

Format 2:

ReturnMessage GetInstrumentInfo(out InstrumentInfoEx info, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
instrumentInfo	O	Instrument information
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This method obtains the instrument information.
Format 1 and format 2 can be used with all supported instruments.

GetSDKVersion: Obtains the SDK version.

Format:

ReturnMessage GetSDKVersion(out string version)

Arguments:

Name	I/O	Explanation
version	O	Version (x.x.x.x)

Return Value:

Definition value	Explanation
KmSuccess	Completed normally

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function obtains the SDK version.

4.3 Calibration and measurement

GetCalibrationStatus: Obtains the calibration status.

Format:

ReturnMessage GetCalibrationStatus(out CalStatus calStatus, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
calStatus	O	Calibration status
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function gets the calibration status.
Execute measurements after checking the calibration status and performing calibration if necessary.

The calibration status is managed for each condition type (e.g. measurement area, specular component).
This method obtains the calibration status based on the conditions set to the instrument.

PerformZeroCalibration: Executes zero calibration.

Format:

ReturnMessage PerformZeroCalibration (Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.
KmErCalibration	Calibration was not executed in the correct procedure.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This command executes zero calibration.
No response is returned until zero calibration has completed.
If the zero calibration fails, the normal state immediately before the zero calibration is attempted is maintained.

Zero calibration does not need to be performed each time, but it should be performed when the measurement environment changes greatly and when the instrument has not be used for a long period of time.

PerformWhiteCalibration: Executes white calibration.

Format:

ReturnMessage PerformWhiteCalibration(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.
KmErCalibration	Calibration was not executed in the correct procedure.
KmErCalibrationRequired	Necessary calibration was not executed beforehand.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function executes white calibration. Check the IDs of the white calibration plates, and use the matching plate.
No response is returned until white calibration has completed.
If the white calibration fails, the normal state immediately before the white calibration is attempted is maintained.

Because the calibration status is managed for each condition type (e.g. measurement area, specular component), re-calibration may be required if any condition is changed.
Use GetCalibrationStatus to determine whether calibration is necessary.

If no white calibration plate data is set, use SetWhiteCalibrationData to set the data.

PerformGlossCalibration: Executes gloss calibration.

Format:

ReturnMessage PerformGlossCalibration(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErCalibration	Calibration was not executed in the correct procedure.
KmErCalibrationRequired	Necessary calibration was not executed beforehand.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	×	×	×	×	V1.0~	×	×

Description:

This function executes gloss calibration. Check the IDs of the gloss calibration plates, and use the matching plate.
No response is returned until gloss calibration has completed.
If the gloss calibration fails, the normal state immediately before the gloss calibration is attempted is maintained.

Because the calibration status is managed for each condition type (e.g. measurement area, specular component), re-calibration may be required if any condition is changed.
Use GetCalibrationStatus to determine whether calibration is necessary.

If no gloss calibration plate data is set, use SetGlossCalibrationData to set the data.

PerformUserCalibration: Executes user calibration.

Format:

ReturnMessage PerformUserCalibration(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErCalibration	Calibration was not executed in the correct procedure.
KmErCalibrationRequired	Necessary calibration was not executed beforehand.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	×	×	V1.0~	V1.0~	V1.0~

Description:

This function executes user calibration. Check the IDs of the user calibration plates, and use the matching plate.
No response is returned until user calibration has completed.
If the user calibration fails, the normal state immediately before the user calibration is attempted is maintained.

Because the calibration status is managed for each condition type (e.g. measurement area, specular component), re-calibration may be required if any condition is changed.
Use GetCalibrationStatus to determine whether calibration is necessary.

If no user calibration plate data is set, use SetUserCalibrationData to set the data.

*CM-36dG/CM-36d
If you conclude a license agreement and perform wavelength correction, use PerformWhiteCalibration before executing this API.

PerformMeasurement: Executes the measurement.

Format:

ReturnMessage PerformMeasurement (Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function executes the measurement.
This function returns a response when the measurement starts.
Use PollingMeasurement to determine the completion of the measurement, and after the measurement is complete, use ReadLatestData or ReadAllLatestData to get data.

When this method is completed successfully and measurement starts, the retained measurement data is cleared. For this reason, the previous data cannot be retrieved in case measurement fails.

PollingMeasurement: Obtains the measurement status.

Format:

ReturnMessage PollingMeasurement(out MeasStatus measStatus, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
measStatus	O	Measurement status
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.
KmErCalibrationRequired	Necessary calibration was not executed beforehand.
KmErTileDetection	The instrument could not measure correctly because it is tilted.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function gets the measurement status.

After verifying that the status has changed from 'measuring' to 'idling,' use ReadLatestData or ReadAllLatestData to retrieve data.

CancelMeasurement: Stops the measurement.

Format:

ReturnMessage CancelMeasurement (Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function stops the measurement.

It can be used when the number of times of automatic averaging is set to multiple times using SetAutoAverageTimes.

If this method is executed when a measurement is not being executed, it will be completed successfully.

ReadLatestData: Obtains the latest measurement data.

Format 1:

ReturnMessage ReadLatestData(DataForm dataForm, SpecData measurementData, Int32 comPort = 0)

Format 2:

ReturnMessage ReadLatestData(DataForm dataForm, ColorValue measurementData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
dataForm	I	Data type
measurementData	I/O	Measurement data
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function retrieves the latest measurement data.
The retrieved data includes the reflection rate and color value. For a color value, the color space must be selected in ColorValue.
After using PollingMeasurement and verifying that the status has changed from 'measuring' to 'idling,' use this method to retrieve the data.
* When gloss data is error, it is stored negative value.

* For Format 2, this function obtains the color values based on the conditions set to the instrument.(Excluding 36dG/36d/3630A)
However, color values cannot be obtained when using a user-prepared illuminant.

* For the 26dG/26d/25d/23d
Opacity measurement is standalone only. If the measurement mode is opacity, the instrument operates in the following states.

26dG	MeasModeColorAndGloss
26d	MeasModeColorOnly
25d	MeasModeColorOnly
23d	MeasModeColorOnly

ReadAllLatestData: Reads all of the latest data.

Format 1:

ReturnMessage ReadAllLatestData(out Dictionary<MeasDataType, List<double>> measurementData, Int32 comPort = 0)

Format 2:

ReturnMessage ReadAllLatestData(MeasDataColor measurementData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
measurementData	I/O	Measurement data
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

Reads all of the latest data.
The retrieved data includes the reflection rate and color value. For a color value, the color space must be selected in ColorValue.
After using PollingMeasurement and verifying that the status has changed from 'measuring' to 'idling,' use this method to retrieve the data.
Use of this function instead of ReadLatestData is recommended when collectively acquiring data because all measurement data can be acquired at once.
* When gloss data is error, it is stored negative value.

* For Format 2, this function obtains the color values based on the conditions set to the instrument. (Excluding 36dG/36d/3630A)
However, color values cannot be obtained when using a user-prepared illuminant.

* For the 26dG/26d/25d/23d
Opacity measurement is standalone only. If the measurement mode is opacity, the instrument operates in the following states.

26dG	MeasModeColorAndGloss
26d	MeasModeColorOnly
25d	MeasModeColorOnly
23d	MeasModeColorOnly

SetWhiteCalibrationData: Sets the white calibration data.

Format 1:

ReturnMessage SetWhiteCalibrationData(MeasArea area, DataForm dataForm, ColorCalData calData, Int32 comPort = 0)

Format 2:

ReturnMessage SetWhiteCalibrationData(CalDataType dataForm, ColorCalData calData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	I	Measurement area
dataForm	I	Data type
calData	I	Calibration data * The calibration plate ID will be overwritten by the ID that was last set.
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function sets the white calibration data.
White calibration is executed using the data set via this function.

Format 1 can be used only with the CM-25cG and CM-M6. Format 2 can be used with all supported instruments.

* When using the CM-M6 with format 1, enter the data for both irradiation directions LEFT and RIGHT.

The measurement area information is ignored.

GetWhiteCalibrationData: Obtains the white calibration data.

Format 1:

ReturnMessage GetWhiteCalibrationData(MeasArea area, DataForm dataForm, ColorCalData calData, Int32 comPort = 0)

Format 2:

ReturnMessage GetWhiteCalibrationData(CalDataType dataForm, ColorCalData calData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	I	Measurement area
dataForm	I	Data type
calData	O	Calibration data
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function obtains the white calibration data.

Format 1 can be used only with the CM-25cG and CM-M6. Format 2 can be used with all supported instruments.

* When using the CM-M6 with format 1, obtain the data for both irradiation directions LEFT and RIGHT. The measurement area information is ignored.

SetGlossCalibrationData: Sets the gloss calibration data.

Format:

ReturnMessage SetGlossCalibrationData(MeasArea area, GlossCalData calData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	I	Measurement area
calData	I	Calibration data
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	×	×	×	×	V1.0~	×	×

Description:

This function sets the gloss calibration data.
Gloss calibration is executed using the data set via this function.

GetGlossCalibrationData: Obtains the gloss calibration data.

Format:

ReturnMessage GetGlossCalibrationData(MeasArea area, GlossCalData calData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	I	Measurement area
calData	O	Calibration data
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	×	×	×	×	V1.0~	×	×

Description:

This function obtains the gloss calibration data.

SetUserCalibrationData: Sets the user calibration data.

Format 1:

ReturnMessage SetUserCalibrationData(MeasArea area, UserCalData calData, Int32 comPort = 0)

Format 2:

ReturnMessage SetUserCalibrationData(CalDataType dataForm, UserCalData2 calData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	I	Measurement area
dataForm	I	Data type
calData	I	Calibration data
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	×	×	V1.0~	V1.0~	V1.0~

Description:

This function sets the user calibration data.
User calibration is executed using the data set via this function.

Format 1 can be used only with the CM-25cG. Format 2 can be used with all supported instruments.

GetUserCalibrationData: Obtains the user calibration data.

Format 1:

ReturnMessage GetUserCalibrationData(MeasArea area, UserCalData calData, Int32 comPort = 0)

Format 2:

ReturnMessage GetUserCalibrationData(CalDataType dataForm, UserCalData2 calData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	I	Measurement area
dataForm	I	Data type
calData	O	Calibration data
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	×	×	V1.0~	V1.0~	V1.0~

Description:

This function obtains the user calibration data.

Format 1 can be used only with the CM-25cG. Format 2 can be used with all supported instruments.

SetUserCalibrationEnable: Enables and disables user calibration.

Format:

ReturnMessage SetUserCalibrationEnable(OnOff UserCalEnable, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
UserCalEnable	I	User calibration on/off
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	×	×	V1.0~	V1.0~	V1.0~

Description:

This function enables and disables user calibration.
If user calibration is turned on, user calibration can be used.

GetUserCalibrationEnable: Obtains the user calibration enabled or disabled state.

Format:

ReturnMessage GetUserCalibrationEnable (out OnOff UserCalEnable, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
UserCalEnable	O	User calibration on/off
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	×	×	V1.0~	V1.0~	V1.0~

Description:

This function obtains the user calibration enabled or disabled state.

SetTriggerMode: Sets the trigger mode.

Format:

ReturnMessage SetTriggerMode(OnOff triggerMode, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
triggerMode	I	Trigger mode
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×

Description:

This method sets the trigger mode.

If the trigger mode is turned on, the instrument key can be used as a measurement trigger.

To obtain the data, confirm the availability of the data with IsThereTriggerData, and then use ReadLatestData.

GetTriggerMode: Obtains the trigger mode.

Format:

ReturnMessage GetTriggerMode(out OnOff triggerMode, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
triggerMode	O	Trigger mode
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×

Description:

This method obtains the trigger mode.

DeleteTriggerData: Deletes the trigger mode data.

Format:

ReturnMessage DeleteTriggerData(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×

Description:

This method deletes data measured by the trigger during trigger mode.
This method executes delete regardless of whether or not there is data.

IsThereTriggerData: Obtains the availability of trigger mode data.

Format:

ReturnMessage IsThereTriggerData(out IsThereData IsThereData, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
IsThereData	O	Availability of data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×

Description:

This method returns whether or not there is data measured by the trigger during trigger mode.

GetZeroCalibrationDate: Obtains the zero calibration date and time.

Format:

ReturnMessage GetZeroCalibrationDate(DateType dateType, out DateTime date, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
dateType	I	Date/time type
date	O	Zero calibration date/time
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErCalibrationRequired	Necessary calibration was not executed beforehand.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function obtains the zero calibration date and time.
If calibration was not executed, this function returns KmErCalibrationRequired.

GetWhiteCalibrationDate: Obtains the white calibration date and time.

Format:

ReturnMessage GetWhiteCalibrationDate(out DateTime date, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
date	O	White calibration date/time
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErCalibrationRequired	Necessary calibration was not executed beforehand.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function obtains the white calibration date and time.
If calibration was not executed, this function returns KmErCalibrationRequired.

GetGlossCalibrationDate: Obtains the gloss calibration date and time.

Format:

ReturnMessage GetGlossCalibrationDate(out DateTime date, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
date	O	Gloss calibration date/time
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErCalibrationRequired	Necessary calibration was not executed beforehand.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	×	×	×	×	V1.0~	×	×

Description:

This function obtains the gloss calibration date and time.
If calibration was not executed, this function returns KmErCalibrationRequired.

GetUserCalibrationDate: Obtains the user calibration date and time.

Format:

ReturnMessage GetUserCalibrationDate(out DateTime date, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
date	O	User calibration date/time
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErCalibrationRequired	Necessary calibration was not executed beforehand.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	×	×	V1.0~	V1.0~	V1.0~

Description:

This function obtains the user calibration date and time.
If calibration was not executed, this function returns KmErCalibrationRequired.

ClearUvAdjustInfo: Clears various data for fluorescence adjustment.

Format:

ReturnMessage ClearUvAdjustInfo(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function clears various data for fluorescence adjustment.
Such data includes profile, WI, Tint, ISO brightness, Ganz & Griesser, and fluorescence coefficient/correction value information.

To clear fluorescence coefficient/correction value information in the instrument, use ClearCoefForUvAdjust.

SetProfileForUvAdjust: Sets the fluorescence adjustment profile data.

Format 1:

ReturnMessage SetProfileForUvAdjust(List<double> data, Int32 comPort = 0)

Format 2:

ReturnMessage SetProfileForUvAdjust(UvAdjustDataType uvtype, List<double> data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be set
data	I	Fluorescence adjustment profile data Value range: 0.01 to 200.00 *the DataSize number obtained by GetInstrumentInfo is used.
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function sets the fluorescence adjustment profile data.

For Format 1, this function sets the same value for all data types that can be set.

GetProfileForUvAdjust: Obtains the fluorescence adjustment profile data.

Format 1:

ReturnMessage GetProfileForUvAdjust(out List<double> data, Int32 comPort = 0)

Format 2:

ReturnMessage GetProfileForUvAdjust(UvAdjustDataType uvtype, out List<double> data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be get
data	O	Fluorescence adjustment profile data *the DataSize number obtained by GetInstrumentInfo is used.
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function obtains the fluorescence adjustment profile data.

For Format 1, this function returns the first data for the data type.

SetWiForUvAdjust: Sets the WI for fluorescence adjustment.

Format 1:

ReturnMessage SetWiForUvAdjust(UvAdjustIndex data, Int32 comPort = 0)

Format 2:

ReturnMessage SetWiForUvAdjust(UvAdjustDataType uvtype, UvAdjustIndex data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be set
data	I	WI for fluorescence adjustment Value range: 40.00 to 250.00 Tolerance range: 0.20 to 3.00
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function sets the WI for fluorescence adjustment.

For Format 1, this function sets the same value for all data types that can be set.

GetWiForUvAdjust: Obtains the WI for fluorescence adjustment.

Format 1:

ReturnMessage GetWiForUvAdjust(UvAdjustIndex data, Int32 comPort = 0)

Format 2:

ReturnMessage GetWiForUvAdjust(UvAdjustDataType uvtype, UvAdjustIndex data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be get
data	O	WI for fluorescence adjustment
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function obtains the WI for fluorescence adjustment.

For Format 1, this function returns the first data for the data type.

SetTintForUvAdjust: Sets the Tint for fluorescence adjustment.

Format 1:

ReturnMessage SetTintForUvAdjust(UvAdjustIndex data, Int32 comPort = 0)

Format 2:

ReturnMessage SetTintForUvAdjust(UvAdjustDataType uvtype, UvAdjustIndex data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be set
Data	I	Tint for fluorescence adjustment Value range: -6.00 to 6.00 Tolerance range: 0.05 to 0.30
Comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function sets the Tint for fluorescence adjustment.

For Format 1, this function sets the same value for all data types that can be set.

GetTintForUvAdjust: Obtains the Tint for fluorescence adjustment.

Format 1:

ReturnMessage GetTintForUvAdjust(UvAdjustIndex data, Int32 comPort = 0)

Format 2:

ReturnMessage GetTintForUvAdjust(UvAdjustDataType uvtype, UvAdjustIndex data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be get
data	O	Tint for fluorescence adjustment
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function obtains the Tint for fluorescence adjustment.

For Format 1, this function returns the first data for the data type.

SetIsoBrightnessForUvAdjust: Sets the ISO brightness for fluorescence adjustment.

Format 1:

ReturnMessage SetIsoBrightnessForUvAdjust(UvAdjustIndex data, Int32 comPort = 0)

Format 2:

ReturnMessage SetIsoBrightnessForUvAdjust(UvAdjustDataType uvtype, UvAdjustIndex data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be set
data	I	ISO brightness for fluorescence adjustment Value range: 40.00 to 250.00 Tolerance range: 0.50 to 3.00
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function sets the ISO brightness for fluorescence adjustment.

For Format 1, this function sets the same value for all data types that can be set.

GetIsoBrightnessForUvAdjust:Obtains the ISO brightness for fluorescence adjustment.

Format 1:

ReturnMessage GetIsoBrightnessForUvAdjust(UvAdjustIndex data, Int32 comPort = 0)

Format 2:

ReturnMessage GetIsoBrightnessForUvAdjust(UvAdjustDataType uvtype, UvAdjustIndex data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be get
data	O	ISO brightness for fluorescence adjustment
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function obtains the ISO brightness for fluorescence adjustment.

For Format 1, this function returns the first data for the data type.

SetGanzForUvAdjust: Sets the Ganz & Griesser for fluorescence adjustment.

Format 1:

ReturnMessage SetGanzForUvAdjust(UvAdjustGG data, Int32 comPort = 0)

Format 2:

ReturnMessage SetGanzForUvAdjust(UvAdjustDataType uvtype, UvAdjustGG data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be set
data	I	Ganz & Griesser for fluorescence adjustment WI range: 40.00 to 250.00 Tint range: -6.00 to 6.00
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function sets the Ganz & Griesser for fluorescence adjustment.

For Format 1, this function sets the same value for all data types that can be set.

GetGanzForUvAdjust: Obtains the Ganz & Griesser for fluorescence adjustment.

Format 1:

ReturnMessage GetGanzForUvAdjust(UvAdjustGG data, Int32 comPort = 0)

Format 2:

ReturnMessage GetGanzForUvAdjust(UvAdjustDataType uvtype, UvAdjustGG data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uvtype	I	Type of data to be get
data	O	Ganz & Griesser for fluorescence adjustment
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function obtains the Ganz & Griesser for fluorescence adjustment.

For Format 1, this function returns the first data for the data type.

SetDataForUvAdjust: Sets the data for fluorescence adjustment.

Format:

ReturnMessage SetDataForUvAdjust(UvAdjustDataType datatype, Int32 num, List<double> dataFull, List<double> dataCut, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
dataType	I	Data types for fluorescence adjustment
num	I	No. * When not using Ganz & Griesser: "0" only * When using Ganz & Griesser: "0" to "3" or "0" to "4"
dataFull	I	UV full measurement data Value range: 0.00 to 300.00 *the DataSize number obtained by GetInstrumentInfo is used.
dataCut	I	UV cut measurement data Value range: 0.00 to 300.00 *the DataSize number obtained by GetInstrumentInfo is used.
Comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function sets the data for fluorescence adjustment.

When using the Ganz & Griesser method, the data must be set by using this API.

GetDataForUvAdjust: Obtains the data for fluorescence adjustment.

Format:

ReturnMessage GetDataForUvAdjust(UvAdjustDataType datatype, Int32 num, out List<double> dataFull, out List<double> dataCut, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
dataType	I	Data types for fluorescence adjustment
num	I	No. * When not using Ganz & Griesser: "0" only * When using Ganz & Griesser: "0" to "3" or "0" to "4"
dataFull	O	UV full measurement data *the DataSize number obtained by GetInstrumentInfo is used.
dataCut	O	UV cut measurement data *the DataSize number obtained by GetInstrumentInfo is used.
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function obtains the data for fluorescence adjustment.

PerformUvAdjust: Executes fluorescence adjustment and sets the coefficient to the instrument.

Format:

ReturnMessage PerformUvAdjust(CondUvAdjust cond, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
cond	I	Conditions for fluorescence adjustment
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	The data does not exist (the required data is not available).
KmErUvAdjust	The measurement sample does not contain fluorescence.
KmErCalculateCoef	The fluorescence coefficient cannot be calculated.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function executes fluorescence adjustment.
This API executes measurement, calculates the fluorescence coefficient under the specified conditions, and then writes the data to the instrument.

If UVADJ_GG is specified in the conditions, KmErInvalidParameter will be returned.
If fluorescence adjustment is performed using the Ganz & Griesser method, PerformUvAdjustUsingData should be used.

PerformUvAdjustUsingData: Executes fluorescence adjustment and sets the coefficient to the instrument.

Format:

ReturnMessage PerformUvAdjustUsingData(CondUvAdjust cond, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
cond	I	Conditions for fluorescence adjustment
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	The data does not exist (the required data is not available).
KmErUvAdjust	The measurement sample does not contain fluorescence.
KmErCalculateCoef	The fluorescence coefficient cannot be calculated.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function executes fluorescence adjustment.

This API uses the data from SetDataForUvAdjust, calculates the fluorescence coefficient under the specified conditions, and then writes the data to the instrument.

ClearCoefForUvAdjust: Clears the fluorescence adjustment coefficient in the instrument.

Format:

ReturnMessage ClearCoefForUvAdjust(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function clears the fluorescence adjustment coefficient saved in the instrument.

SetCoefForUvAdjust: Sets the fluorescence coefficient.

Format:

ReturnMessage SetCoefForUvAdjust(MeasArea area, UvAdjustDataType uvtype, CondUvAdjust cond, UvAdjustCoef coef, Int32 comPort = 0)

引数：

Name	I/O	Explanation
area	I	Measurement area
uvtype	I	Type of data to be set
cond	I	Conditions for fluorescence adjustment
coef	I	Fluorescence adjustment coefficient
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmWarning	The processing was completed normally (there was a warning).
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This function sets the fluorescence coefficient in the instrument.

GetCoefForUvAdjust: Obtains the fluorescence coefficient.

Format:

ReturnMessage GetCoefForUvAdjust(MeasArea area, UvAdjustDataType uvtype, out CondUvAdjust cond, UvAdjustCoef coef, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	I	Measurement area
uvtype	I	Type of data to be get
cond	O	Conditions for fluorescence adjustment
coef	O	Fluorescence adjustment coefficient
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmWarning	The processing was completed normally (there was a warning).
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.1~	×	×	×	V1.0~	×	V1.0~

Description:

This method obtains the fluorescence coefficient in the instrument.

SetOutputMinus: Enables and disables output of negative values.

Format:

ReturnMessage SetOutputMinus(OnOff enable, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
enable	I	Enable/disable output of negative values
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	V1.0~	×

Description:

This function sets whether or not to output negative reflectance values.
If OutputMinus is enabled, the instrument will output reflectance/transmittance values of 0% or less.
If OutputMinus is disabled, the instrument will output reflectance/transmittance of 0% or less as 0.01.
This function can be used when calculating strength and absorbance.

GetOutputMinus: Obtains the output of negative values enabled or disabled state.

Format:

ReturnMessage GetOutputMinus(out OnOff enable, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
enable	O	Enable/disable output of negative values
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	V1.0~	×

Description:

This function obtains the output of negative values enabled or disabled state.

4.4 Measurement conditions

SetMeasurementArea: Sets the measurement area.

Format:

ReturnMessage SetMeasurementArea(MeasArea area, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	I	Measurement area
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	V1.0~	V1.0~

Description:

This function sets the measurement area.

Calibration may be required again when conditions are changed.
Use GetCalibrationStatus to determine whether calibration is necessary.

* For the 36dG
Regarding gloss measurement, if the measurement area is SAV, it is set to SAV.
If the measurement area is not SAV, it is set to MAV.

* When this API sets AREA_AUTO
Use SetFinderEnable to enable the camera.
Use GetDetectedMask to get the measurement area.

GetMeasurementArea: Obtains the measurement area.

Format:

ReturnMessage GetMeasurementArea(out MeasArea area, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	O	Measurement area
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	×	×	×	V1.0~	V1.0~	V1.0~

Description:

This function obtains the measurement area.

SetMeasurementType: Sets the measurement type.

Format:

ReturnMessage GetMeasurementType(MeasType inType, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
inType	I	Measurement type
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified method.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	×	×

Description:

This method sets the measurement type.

Calibration may be required again when conditions are changed.
Use GetCalibrationStatus to determine whether calibration is necessary.
When using transmitted, use SetMeasurementMode to set MeasModeColorOnly.

GetMeasurementType: Obtains the measurement type.

Format:

ReturnMessage GetMeasurementType(out MeasType outType, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
outType	O	Measurement type
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified method.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	×	×

Description:

This method obtains the measurement type.

SetMeasurementAngle: Sets the measurement angle.

Format:

ReturnMessage SetMeasurementAngle(MeasAngle angle, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
angle	I	Measurement angle
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	V1.0~	×	×	×

Description:

This function sets the measurement angle.
If at least one angle is specified, the angle(s) can be set with any combination.

GetMeasurementAngle: Obtains the measurement angle.

Format:

ReturnMessage GetMeasurementAngle(out MeasAngle angle, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
angle	O	Measurement angle
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	V1.0~	×	×	×

Description:

This function obtains the measurement angle.

SetTiltDetection: Sets tilt detection.

Format:

ReturnMessage SetTiltDetection(OnOff tilt, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
tilt	I	Tilt detection setting
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	V1.0~	×	×	×

Description:

This method sets tilt detection.

GetTiltDetection: Obtains whether tilt detection is enabled or disabled.

Format:

ReturnMessage GetTiltDetection(out OnOff tilt, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
tilt	O	Tilt detection setting
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	V1.0~	×	×	×

Description:

This function obtains whether tilt detection is enabled or disabled.

SetMeasurementMode: Sets the measurement mode.

Format:

ReturnMessage SetMeasurementMode (MeasCondMode Mode, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
Mode	I	Measurement mode
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	V1.0~	×	×

Description:

This function sets the measurement mode.

Calibration may be required again when conditions are changed.
Use GetCalibrationStatus to determine whether calibration is necessary.

GetMeasurementMode: Obtains the measurement mode.

Format:

ReturnMessage GetMeasurementMode (out MeasCondMode Mode, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
Mode	O	Measurement mode
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	V1.0~	×	×

Description:

This function obtains the measurement mode.

SetSpecularComponent: Sets the specular component.

Format:

ReturnMessage SetSpecularComponent(MeasCondScie scie, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
scie	I	Specular component
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	×	V1.0~	V1.0~	×

Description:

This function sets the specular component.

Calibration may be required again when conditions are changed.
Use GetCalibrationStatus to determine whether calibration is necessary.

GetSpecularComponent: Obtains the specular component.

Format:

ReturnMessage GetSpecularComponent(out MeasCondScie scie, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
scie	O	Specular component
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	×	V1.0~	V1.0~	×

Description:

This function obtains the specular component.

SetUv: Sets the UV condition.

Format:

ReturnMessage SetUv(MeasCondUv uv, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uv	I	UV condition
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	×	×	×	V1.0~	V1.0~	V1.0~

Description:

This function sets the UV condition.

GetUv: Obtains the UV condition.

Format:

ReturnMessage GetUv(out MeasCondUv uv, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
uv	O	UV condition
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	×	×	×	V1.0~	V1.0~	V1.0~

Description:

This function obtains the UV condition.

SetAutoAverageTimes: Sets the number of times of automatic averaging.

Format:

ReturnMessage SetAutoAverageTimes(Int32 times, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
times	I	Number of times of automatic averaging (1-10)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function sets the number of times of automatic averaging.
PerformMeasurement function executes measurements for the number of times specified in this condition.

GetAutoAverageTimes: Obtains the number of times of automatic averaging.

Format:

ReturnMessage GetAutoAverageTimes(out Int32 times, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
times	O	Number of times of automatic averaging (1-10)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function obtains the number of times of automatic averaging.

SetManualAverageTimes: Sets the number of times of manual averaging.

Format:

ReturnMessage SetManualAverageTimes(Int32 times, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
times	I	Manual average count * The range of the setting is dependent on the instrument and version. Refer to Appendix C.
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the number of times of manual averaging.

This condition is only applied to standalone measurements, and not applied when communication is used.

GetManualAverageTimes: Obtains the number of times of manual averaging.

Format:

ReturnMessage GetManualAverageTimes(out Int32 times, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
times	O	Manual average count
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the number of times of manual averaging.

SetManualAverageSaveMode: Sets the manual averaging method.

Format:

ReturnMessage SetManualAverageSaveMode(SaveMode mode, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
mode	I	Manual averaging save method
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the manual averaging save method.

GetManualAverageSaveMode: Obtains the manual averaging save method.

Format:

ReturnMessage GetManualAverageSaveMode(out SaveMode mode, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
mode	O	Manual averaging save method
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the manual averaging save method.

SetCondSMC: Sets the SMC conditions.

Format:

ReturnMessage SetCondSMC(MeasCondSMC smc, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
smc	I	SMC conditions
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function sets the SMC conditions.

This condition is only applied to standalone measurements, and not applied when communication is used.

GetCondSMC: Obtains the SMC conditions.

Format:

ReturnMessage GetCondSMC(out MeasCondSMC smc, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
smc	O	SMC conditions
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function obtains the SMC conditions.

4.5 Display conditions

SetDisplayType: Sets the display type.

Format:

ReturnMessage SetDisplayType(DisplayType displayType, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
displayType	I	Display Type
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the display type.
The display type can be set with any combination.
Note that settable items are different depending on the model.

GetDisplayType: Obtains the display type.

Format:

ReturnMessage GetDisplayType(out DisplayType displayType, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
displayType	O	Display Type
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the display type.

SetObserverAndIlluminant: Sets the observation field and illuminant.

Format:

ReturnMessage SetObserverAndIlluminant(Int32 num, Observer obs, Illuminant ill, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Observation field / illuminant number (0-1)
obs	I	Observer
ill	I	Illuminant
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the observation field and illuminant.
Setting the illuminant to 'None' when the observation field / illuminant number is 1 enables the use of one observation field and illuminant set.
Note that settable items are different depending on the model.

When using a user illuminant, use SetUserIlluminant to register illuminant data.

GetObserverAndIlluminant: Obtains the observation field and illuminant.

Format:

ReturnMessage GetObserverAndIlluminant(Int32 num, out Observer obs, out Illuminant ill, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Observation field / illuminant number (0-1)
obs	O	Observer
ill	O	Illuminant
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the observation field and illuminant.

SetUserIlluminant: Sets the user illuminant.

Format 1:

ReturnMessage SetUserIlluminant(List<double> illData, Int32 comPort = 0)

Format 2:

ReturnMessage SetUserIlluminant(List<double> illData, string name, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
illData	I	Illuminant data 85 items of data between 360 to 780 nm at 5-nm pitch
name	I	Name (10 characters max.)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	×	V1.1~	×	×	×

Description:

This function sets the user illuminant.
Set the illuminant data as 85 items of data between 360 to 780 nm (5-nm pitch).

Set "User" when using Format 1 on a model that allows the name to be set.
Ignore the setting when using Format 2 on a model that does not allow the name to be set.

GetUserIlluminant: Obtains the user illuminant.

Format 1:

ReturnMessage GetUserIlluminant(List<double> illData, Int32 comPort = 0)

Format 2:

ReturnMessage GetUserIlluminant(List<double> illData, out string name, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
illData	O	Illuminant data 85 items of data between 360 to 780 nm at 5-nm pitch
name	O	Name
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	×	V1.1~	×	×	×

Description:

This function obtains the user illuminant.

SetColorSpace: Sets the color space.

Format:

ReturnMessage SetColorSpace(ColorSpace colorSpace, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
colorSpace	I	Color space
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the color space.
Note that settable items are different depending on the model.

GetColorSpace: Obtains the color space.

Format:

ReturnMessage GetColorSpace(out ColorSpace colorSpace, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
colorSpace	O	Color space
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the color space.

SetEquation: Sets the color difference equation.

Format:

ReturnMessage SetEquation(Equation equation, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
equation	I	Color difference equation
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the color difference equation.
Note that settable items are different depending on the model.

GetEquation: Obtains the color difference equation.

Format:

ReturnMessage GetEquation(out Equation equation, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
equation	O	Color difference equation
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the color difference equation.

SetCustomItem: Sets the custom display items.

Format:

ReturnMessage SetCustomItem (Int32 customNum, CustomItem customItem, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
customNum	I	Custom display number (14 numbers, 0-13)
customItem	I	Custom display item
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function sets the custom display items.
To display custom items, use SetDisplayType to enable 'custom.'

GetCustomItem: Obtains the custom display items.

Format:

ReturnMessage GetCustomItem(Int32 customNum, out CustomItem customItem, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
customNum	I	Custom display number (14 numbers, 0-13)
customItem	O	Custom display item
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function obtains the custom display items.

SetDisplayData: Sets the irradiation direction of data to display.

Format:

ReturnMessage SetDisplayData(DisplayData data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
data	I	Data to display
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	V1.1~	×	×	×

Description:

This method sets the irradiation direction of data to display.

GetDisplayData: Obtains the irradiation direction of data to display.

Format:

ReturnMessage GetDisplayData (out DisplayData data, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
data	O	Data being displayed
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	V1.1~	×	×	×

Description:

This method obtains the irradiation direction of data being displayed.

SetUserEquation: Sets the user index.

Format1:

ReturnMessage SetUserEquation(Int32 num, string data, string name, Int32 comPort = 0)

Format2:

ReturnMessage SetUserEquation(Int32 num, string data, string name, string index_name, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	User index number (0-2)
data	I	User index
name	I	User classes
index_name	I	User index name
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function sets the user index.

The color space values that can be used for a user index are only the items that are selected in SetColorSpace, SetEquation, and SetCustomItem.

Sample user index: Equivalent to ΔE*ab
SQRT(POW([DL])+POW([DA])+POW([DB]))

The following variables can be used for a user index.

[L]	L*	[DL]	ΔL*	[MI]	MI
[A]	a*	[DA]	Δa*	[WIE]	WI(E313-73)
[B]	b*	[DB]	Δb*	[DWIE]	ΔWI(E313-73)
[C]	C*	[DC]	ΔC*	[WIC]	WI(CIE)
[H]	h	[DH]	ΔH*	[DWIC]	ΔWI(CIE)
[HL]	L(Hunter)	[DHL]	ΔL	[TINT]	Tint
[HA]	a(Hunter)	[DHA]	Δa	[DTINT]	ΔTint
[HB]	b(Hunter)	[DHB]	Δb	[YIE]	YI(E313-73)
[X]	X	[DX]	ΔX	[DYIE]	ΔYI(E313-73)
[Y]	Y	[DY]	ΔY	[YID]	YI(D1925)
[Z]	Z	[DZ]	ΔZ	[DYID]	ΔYI(D1925)
[SX]	x	[DSX]	Δx	[BISO]	B
[SY]	y	[DSY]	Δy	[DBISO	ΔB
[GU]	GU	[DGU]	ΔGU		
[DE]	ΔE*ab	[DE94]	ΔE*94	[DEH]	ΔE(Hunter)
[CMC]	CMC	[DE00]	ΔE00		

* For the CM-26dG/26d/25d/23d, when items are limited by SCI and SCE, add "I" and "E" to the variables.
E.g.) To calculate L*(SCI)+L*(SCE), set [LI]+[LE].

The operators and functions that can be used for a user index are as follows:

+	[A]+[B]	A+B

-	[A]-[B]	A-B
*	[A]*[B]	A×B
/	[A]/[B]	A÷B
POW	POW([A])	Square of A
SQRT	SQRT([A])	Square root of A
ABS	ABS([A])	Absolute value of A
SIN	SIN([A])	Sine of A (degree)
COS	COS([A])	Cosine of A (degree)
TAN	TAN([A])	Tangent of A (degree)
ASIN	ASIN([A])	Arcsine (degree)
ACOS	ACOS([A])	Arccosine (degree)
ATAN	ATAN([A])	Arctangent (degree)
LOG	LOG([A])	Common logarithm
LN	LN([A])	Natural logarithm
EXP	EXP([A])	Exponential function
POW2	POW2([A],[B])	Power function (A to the power of B)

Input format of the user classes

```
CLASS(n, “str1”, d1, “str2”, d2, “str3”, d3, “str4”, d4, “str5”, …)
```

n: Indicates the number of threshold values that separate the classes. (Number of classes - 1). Specify this as a number within 50.

“str1”,d1: When the judgment result of the user classes is d1 or higher, “str1” is displayed in the results field of the instrument.

d1 can be numeric value setting of 20 or fewer digits, but the effective digits in the calculation are 5 digits. Enter the threshold values from the left in order of the largest values, and always set a class to be displayed if a value is not classified into a threshold value greater than or equal to all of the threshold values.

The total length of characters that are entered (including CLASS()) must be within 200 single-byte characters.

Leave blank if the user classes will not be used.

Use “.” for the decimal point and “,” as the separator between parameters.

Input format of the user classes

```
CLASS(4, “A”, 4, “B”, 3, “C”, 2, “D”, 1, “E”)
```

The result of the user index is split into 5 classes.

User index result	Class
4 or higher	A
3 or higher	B
2 or higher	C
1 or higher	D
Less than 1	E

GetUserEquation: Obtains the user index.

Format 1:

ReturnMessage GetUserEquation(Int32 num, out string data, out string name, Int32 comPort = 0)

Format 2:

ReturnMessage GetUserEquation(Int32 num, out string data, out string name, out string index_name, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	User index number (0-2)
data	O	User index
name	O	User classes
index_name	O	User index name
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function obtains the user index.

4.6 Data

SetActiveTarget: Sets the active target number.

Format:

ReturnMessage SetActiveTarget(Int32 num, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Target number
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the active target number.

The active target is used as a target number associated for measurements after setting.

GetActiveTarget: Obtains the active target number.

Format:

ReturnMessage GetActiveTarget(out Int32 num, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	O	Target number
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the active target number.

GetTargetNumberList: Obtains the list of saved target numbers.

Format:

ReturnMessage GetTargetNumberList (out List<Int32> targetlist, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
targetlist	O	List of registered target numbers
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method obtains the list of registered target numbers.
It returns the target numbers of all saved target color data.

GetTargetNumberList2: This function obtains the list of target numbers when the display filter is applied.

Format:

ReturnMessage GetTargetNumberList2 (out List<Int32> targetlist, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
targetlist	O	List of registered target numbers
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This method obtains the list of target numbers when the display filter was applied.

DeleteTargetData: Deletes the target.

Format:

ReturnMessage DeleteTargetData(Int32 num, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Target number * All targets are cleared when '-1' is specified.
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function deletes the target for the specified number.
All targets are cleared when '-1' is specified as the number.

SetTargetData: Sets the target data.

Format 1:
ReturnMessage SetTargetData(Int32 num, TargetData target, Int32 comPort = 0)
Format 2:
ReturnMessage SetTargetData(Int32 num, TargetDataPack target, Int32 comPort = 0)
Arguments:

Name	I/O	Explanation
num	I	Target number
target	I	Data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.
Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:
This method sets the target data for the specified number.
If the target already exists, it is overwritten.

If data protection is enabled with SetTargetProtect, the target data cannot be set.

Format 1 can be used only with the CM-25cG and CM-M6. Format 2 can be used with all supported instruments.
* For the CM-M6, six angles must be set as a group.

To exchange color value data, use format 2.
* With opacity, the color value cannot be registered.

GetTargetData: Obtains the target data.

Format 1:

ReturnMessage GetTargetData(Int32 num, TargetData target, Int32 comPort = 0)

Format 2:

ReturnMessage GetTargetData(Int32 num, TargetDataPack target, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Target number
target	O	Data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErNotSupported	Obtaining color value data is not supported. (Use the format 2 method.)

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method obtains the target data for the specified number.

Format 1 can be used only with the CM-25cG and CM-M6. Format 2 can be used with all supported instruments.
Format 2 can obtain all saved data with one call, so using this method is recommended.
Even for the CM-M6, this method can obtain all data faster than GetAllTargetData.

Format 1:

- For the CM-25cG:

  Enter ColorSpace of the ColorValue class.

- For the CM-M6:

  Enter ColorSpace of the ColorValue class.
  Enter DataType and IrradiationDirection of the DataForm class.

GetAllTargetData: Obtains the target data.

Format:

ReturnMessage GetAllTargetData(Int32 num, TargetData target, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Target number
target	O	Data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	V1.0~	×	×	×

Description:

This method obtains the target data for the specified number.

* This method is for the CM-M6 only.

Enter ColorSpace of the ColorValue class.

The calculated values that were obtained are stored in AllAngleValue of ColorValue.

Enter IrradiationDirection of the DataForm class.

Format 2 of GetTargetData can obtain all saved data with one call, so using that method is recommended. That method can obtain all data in a shorter amount of time than this method.

SetToleranceForTarget: Sets the target tolerance.

Format:

ReturnMessage SetToleranceForTarget(Int32 num, Int32 numObsIll, DataForm dataForm, ToleranceData tolerance, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Target number (1-[Number obtained using GetTargetNumberList])
numObsIll	I	Observation field / illuminant number (0-1)
dataForm	I	Data type
tolerance	I	Tolerance data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method sets the target tolerance for the specified number.

If data protection is enabled with SetTargetProtect, the target data cannot be set.

* Only Tolerance specified by the key is changed.

GetToleranceForTarget: Obtains the target tolerance.

Format:

ReturnMessage GetToleranceForTarget(Int32 num, Int32 numObsIll, DataForm dataForm, ToleranceData tolerance, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Target number (1-[Number obtained using GetTargetNumberList])
numObsIll	I	Observation field / illuminant number (0-1)
dataForm	I	Data type
tolerance	O	Tolerance data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method obtains the target tolerance for the specified number.

* Enter Key for the tolerance to obtain and a blank ToleranceParam in the ToleranceData class.

SetParametricForTarget: Sets the parametric coefficient for a target color.

Format:

ReturnMessage SetParametricForTarget(Int32 num, DataForm dataForm, ParametricCoef parametric, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Target number (1-[Number obtained using GetTargetNumberList])
dataType	I	Data type
parametric	I	Parametric coefficient
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method sets the target parametric coefficient for the specified number.

If data protection is enabled with SetTargetProtect, the target data cannot be set.

GetParametricForTarget: Obtains the parametric coefficient for a target color.

Format:

ReturnMessage GetParametricForTarget(Int32 num, DataForm dataForm, ParametricCoef parametric, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Target number (1-[Number obtained using GetTargetNumberList])
dataType	I	Data type
parametric	O	Parametric coefficient
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.
KmErNoData	No data

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method obtains the target parametric coefficient for the specified number.

SetTargetFilter: Sets the target filter conditions.

Format:

ReturnMessage SetTargetFilter(FilterIndex index, List<Int32> numGroup, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
index	I	Filter index
numGroup	I	Group number (0-50) * This item is only used when the filter index is 'group.' Otherwise, set this to 0.
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function sets the target filter conditions.

If the filter is not set to OFF, only the target data with the number that matches the condition is displayed.

GetTargetFilter: Obtains the target filter conditions.

Format:

ReturnMessage GetTargetFilter(out FilterIndex index, out List<Int32> numGroup, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
index	O	Filter index
numGroup	O	Group number (0-50) * This item is only returned when the filter index is 'group.' Otherwise, it will be 0.
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function obtains the target filter conditions.

SetTargetProtect: Sets target protection.

Format:

ReturnMessage SetTargetProtect(OnOff protect, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
protect	I	Protection ON/OFF
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets target protection.
When target protection is turned on, only new measurement (save) operations are allowed.

GetTargetProtect: Obtains target protection.

Format:

ReturnMessage GetTargetProtect(out OnOff protect, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
protect	O	Protection ON/OFF
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the target protection setting.

GetSampleCount: Obtains the number of saved measurement values.

Format:

ReturnMessage GetSampleCount(out Int32 count, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
count	O	Number of measurement data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method obtains the number of measurement data.

DeleteSampleData: Deletes a measurement value.

Format:

ReturnMessage DeleteSampleData(Int32 num, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Measurement value number * All targets are cleared when '-1' is specified.
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function deletes the measurement value for the specified number.
All targets are cleared when '-1' is specified as the number.

GetSampleData: Obtains the measurement data.

Format 1:

ReturnMessage GetSampleData(Int32 num, SampleData sample, Int32 comPort = 0)

Format 2:

ReturnMessage GetSampleData(Int32 num, SampleDataPack sample, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Measurement value number
sample	O	Data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method obtains the measurement data for the specified number.

Format 1 can be used only with the CM-25cG and CM-M6. Format 2 can be used with all supported instruments.
Format 2 can obtain all saved data with one call, so using this method is recommended.
Even for the CM-M6, this method can obtain all data faster than GetAllSampleData.

Format 1:

- For the CM-25cG:

  Enter ColorSpace of the ColorValue class.

- For the CM-M6:

  Enter ColorSpace of the ColorValue class.
  Enter DataType and IrradiationDirection of the DataForm class.

GetAllSampleData: Obtains the measurement data.

Format:

ReturnMessage GetAllSampleData(Int32 num, SampleData sample, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
num	I	Measurement value number
sample	O	Data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	V1.0~	×	×	×

Description:

This method obtains the measurement data for the specified number.

* This method is for the CM-M6 only.

Enter ColorSpace of the ColorValue class.

The calculated values that were obtained are stored in AllAngleValue of the ColorValue class inside the SampleData class.

The spectral reflectance that was obtained are stored in AllAngleData of the SampleData class.

Enter IrradiationDirection of the DataForm class.

4.7 Others

SetActiveGroup: Sets the active group number.

Format:

ReturnMessage SetActiveGroup(List<Int32> numGroup, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
numGroup	I	Group number (0-50)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function sets the active group number.
The active group number is used as a group number associated for measurements after setting.
Enter 0 if nothing will be set.

GetActiveGroup: Obtains the active group number.

Format:

ReturnMessage GetActiveGroup(out List<Int32> numGroup, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
numGroup	O	Group number (0-50)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function obtains the active group number.

SetGroupName: Sets the group name.

Format:

ReturnMessage SetGroupName(Int32 numGroup, string name, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
numGroup	I	Group number (1-50)
name	I	Group name
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function sets the group name for the specified number.

GetGroupName: Obtains the group name.

Format:

ReturnMessage GetGroupName(Int32 numGroup, out string name, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
numGroup	I	Group number (1-50)
name	O	Group name
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function obtains the group name corresponding to the specified number.

SetMultipleGroupName: Sets multiple group names in batch.

Format:

ReturnMessage SetMultipleGroupName(Int32 groupList, List<string> name, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
groupList	I	Group list (0 to 4) 0 Group number 1 to 10 1 Group number 11 to 20 2 Group number 21 to 30 3 Group number 31 to 40 4 Group number 41 to 50
name	I	Group name (1 to 10)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This method sets group names in batch. (Up to 10 group names can be set for one group list.)

GetMultipleGroupName: Obtains multiple group names in batch.

Format:

ReturnMessage GetMultipleGroupName(Int32 groupList, out List<string> name, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
groupList	I	Group list (0 to 4) 0: Group number 1 to 10 1: Group number 11 to 20 2: Group number 21 to 30 3: Group number 31 to 40 4: Group number 41 to 50
name	O	Group name (1 to 10)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This method obtains the group names registered to the group list in batch.

SetTolerance: Sets the default tolerance.

Format:

ReturnMessage SetTolerance(Int32 numObsIll, DataForm dataForm, ToleranceData tolerance, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
numObsIll	I	Observation field / illuminant number (0-1)
dataForm	I	Data type
tolerance	I	Tolerance data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method sets the default tolerance.
This tolerance is assigned when saving a new target.
* Only Tolerance specified by the key is changed.

GetTolerance: Obtains the default tolerance.

Format:

ReturnMessage GetTolerance(Int32 numObsIll, DataForm dataForm ToleranceData tolerance, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
numObsIll	I	Observation field / illuminant number (0-1)
dataForm	I	Data type
tolerance	O	Tolerance data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method obtains the default tolerance.
* Enter Key for the tolerance to obtain and a blank ToleranceParam in the ToleranceData class.

SetParametric: Sets the default parametric coefficient.

Format:

ReturnMessage SetParametric(DataForm dataForm, ParametricCoef parametric, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
dataForm	I	Data type
Parametric	I	Parametric coefficient
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method sets the default parametric coefficient.
This tolerance is assigned when saving a new target.

GetParametric: Obtains the default parametric coefficient.

Format:

ReturnMessage GetParametric(DataForm dataForm, ParametricCoef parametric, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
dataForm	I	Data type
parametric	O	Parametric coefficient
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This method obtains the default parametric coefficient.

SetWarningLevel: Sets the warning level.

Format:

ReturnMessage SetWarningLevel(Int32 level, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
level	I	Warning level (0 to 100%)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the warning level.

GetWarningLevel: Obtains the warning level.

Format:

ReturnMessage GetWarningLevel(out Int32 level, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
level	I	Warning level (0 to 100%)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the warning level.

SetInstrumentMode: Sets the instrument mode.

Format:

ReturnMessage SetInstrumentMode(InstrumentMode mode, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
mode	I	Instrument mode
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function sets the instrument mode.

When the power supply to the instrument is turned off, the setting reverts normal mode.

GetInstrumentMode: Obtains the instrument mode.

Format:

ReturnMessage GetInstrumentMode(out InstrumentMode mode, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
mode	O	Instrument mode
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function obtains the instrument mode.

SetUserType: Sets the user type.

Format:

ReturnMessage SetUserType(UserType userType, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
userType	I	User type
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the user type.

GetUserType: Obtains the user type.

Format:

ReturnMessage GetUserType(out UserType userType, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
userType	O	User type
comPort	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the user type.

SetAdminPassword: Sets the administrator password.

Format:

ReturnMessage SetAdminPassword(string adminPass, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
adminPass	I	Administrator password (numbers only, 4 to 8 digits)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function sets the administrator password.

The password is used to change the settings from operator to administrator when the instrument is used standalone.

GetAdminPassword: Obtains the administrator password.

Format:

ReturnMessage GetAdminPassword(out string adminPass, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
adminPass	O	Administrator password (numbers only, 4 to 8 digits)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function obtains the administrator password.

SetAutoPrint: Sets automatic printing.

Format:

ReturnMessage SetAutoPrint(OnOff autoPrint, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
autoPrint	I	Automatic printing ON/OFF
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets automatic printing.

GetAutoPrint: Obtains the automatic printing setting.

Format:

ReturnMessage GetAutoPrint(out OnOff autoPrint, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
autoPrint	O	Automatic printing ON/OFF
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the automatic printing setting.

SetBrightness: Sets the brightness of the display.

Format:

ReturnMessage SetBrightness(Int32 brightness, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
brightness	I	Display brightness (0 to 4)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the brightness of the display.

GetBrightness: Obtains the brightness of the display.

Format:

ReturnMessage GetBrightness(out Int32 brightness, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
brightness	O	Display brightness (0 to 4)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the brightness of the display.

SetScreenDirection: Sets the display direction of the screen.

Format:

ReturnMessage SetScreenDirection(ScreenDirection direction, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
direction	I	Display direction of screen
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the display direction of the screen.

GetScreenDirection: Obtains the display direction of the screen.

Format:

ReturnMessage GetScreenDirection(out ScreenDirection direction, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
direction	O	Display direction of screen
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the display direction of the screen.

SetSound: Sets the beep.

Format:

ReturnMessage SetSound(OnOff sound, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
sound	I	Sound ON/OFF
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the beep.

When the sound is turned on, the instrument will generate a sound when calibration or measurement is completed or ends in an error.

GetSound: Obtains the beep.

Format:

ReturnMessage GetSound(out OnOff sound, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
sound	O	Sound ON/OFF
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the beep.

SetCalibrationInterval: Sets the calibration interval.

Format:

ReturnMessage SetCalibrationInterval(Int32 calInterval, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
calInterval	I	Calibration interval (1-24) (1-hr pitch)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function sets the calibration interval.

Note that, although it is possible to run a measurement after the calibration interval ends, the instrument is in an alert state where calibration is recommended.

GetCalibrationInterval: Obtains the calibration interval.

Format:

ReturnMessage GetCalibrationInterval(out Int32 calInterval, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
calInterval	O	Calibration interval (1-24) (1-hr pitch)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function obtains the calibration interval.

SetAnnualCalibration: Sets the periodical calibration notice.

Format:

ReturnMessage SetAnnualCalibration(OnOff annualCal, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
annualCal	I	Periodical calibration notice ON/OFF
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function sets the periodical calibration notice.

GetAnnualCalibration: Obtains the periodical calibration notice.

Format:

ReturnMessage GetAnnualCalibration(out OnOff annualCal, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
annualCal	O	Periodical calibration notice ON/OFF
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Description:

This function obtains the periodical calibration notice setting.

SetZeroCalibrationSkip: Sets whether or not to skip zero calibration.

Format:

ReturnMessage SetZeroCalibrationSkip(OnOff skip, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
skip	I	Skip zero calibration on/off
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function sets whether or not to skip zero calibration.

If skip zero calibration is turned on, zero calibration can be skipped when the instrument starts. If turned off, zero calibration must always be performed.

GetZeroCalibrationSkip: Obtains whether or not zero calibration can be skipped.

Format:

ReturnMessage GetZeroCalibrationSkip(out OnOff skip, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
skip	O	Skip zero calibration on/off
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	V1.0~	×	×	×	×

Description:

This function obtains whether skip zero calibration is on or off.

SetDateTime: Sets the date and time.

Format:

ReturnMessage SetDateTime(DateTime date, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
date	I	Date/time
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the date and time.

SetDateFormat: Sets the date format.

Format:

ReturnMessage SetDateFormat(DateFormat dateFormat, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
dateFormat	I	Date format
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the date format.

GetDateFormat: Obtains the date format.

Format:

ReturnMessage GetDateFormat(out DateFormat dateFormat, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
dateFormat	O	Date format
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the date format.

SetLanguage: Sets the display language.

Format:

ReturnMessage SetLanguage(Language language, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
language	I	Display language
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function sets the display language.

GetLanguage: Obtains the display language.

Format:

ReturnMessage GetLanguage(out Language language, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
language	O	Display language
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	V1.0~	×	×	×

Description:

This function obtains the display language.

SetAutoPowerOff: Sets the time until auto power off functions.

Format:

ReturnMessage SetAutoPowerOff(Int32 time, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
time	I	0 to 60 minutes
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.1~	V1.1~	V1.0~	V1.0~	V1.0~	V1.1~	×	×	×

Description:

This method sets the time until auto power off functions.
If set to 0, the auto power off function is turned off.

GetAutoPowerOff: Obtains the time until auto power off functions.

Format:

ReturnMessage GetAutoPowerOff(out Int32 time, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
time	O	0 to 60 minutes
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.1~	V1.1~	V1.0~	V1.0~	V1.0~	V1.1~	×	×	×

Description:

This method obtains the time until auto power off functions that has been set on the instrument.

ClearJobInfo: Clears job information.

Format:

ReturnMessage ClearJobInfo(Int32 jobNum, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function clears job information.
It clears the step information and image information registered to the specified jog number.

SetJobInfo: Sets job information.

Format:

ReturnMessage SetJobInfo(Int32 jobNum, JobInfo info, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
info	I	Job information
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function sets job information.

Use SetJobStepForOperation and SetJobStepForResult to set the number of steps specified here.

Refer to “3.2 Basic processing flow” for the procedure.

After the job information is registered to the instrument, perform trial operation to determine if the job has been configured appropriately before putting the job into operation.

GetJobInfo: Obtains job information.

Format:

ReturnMessage GetJobInfo(Int32 jobNum, out JobInfo info, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
info	O	Job information
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function obtains job information.

GetJobStepType: Obtains the step type of the job.

Format:

ReturnMessage GetJobStepType(Int32 jobNum, Int32 stepNum, out JobStepType stepType, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
stepNum	I	Step number (0 to 19)
stepType	O	Step type
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function obtains the step type of the job.

If the job type is operation step, use GetJobStepForOperation. If the job type is result step, use GetJobStepForResult.

SetJobStepForOperation: Sets an operation step of the job.

Format:

ReturnMessage SetJobStepForOperation(Int32 jobNum, Int32 stepNum, JobStepOperation jobStep, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
stepNum	I	Step number (0 to 19)
jobStep	I	Step content (operation step)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function sets an operation step of the job.

The steps (operation steps or result steps) must be set in the amount of the number of steps specified by SetJobInfo.

Refer to "3.2 Basic processing flow" for the procedure.

The following table gives the items that must be set for each instrument. Entered information is ignored for cells indicated by "-".

	26dG	26d	25d
meas_type	-	-	-
meas_mode	✓	-	-
meas_area	✓	✓	-
meas_angle	-	-	-
meas_ldirection	-	-	-
meas_scie	✓	✓	✓
meas_uv	✓	✓	-

GetJobStepForOperation: Obtains an operation step of the job.

Format:

ReturnMessage GetJobStepForOperation(Int32 jobNum, Int32 stepNum, out JobStepOperation jobStep, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
stepNum	I	Step number (0 to 19)
jobStep	O	Step content (operation step)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function obtains an operation step of the job.

SetJobStepForResult: Sets a result step of the job.

Format:

ReturnMessage SetJobStepForResult(Int32 jobNum, Int32 stepNum, JobStepResult jobStep, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
stepNum	I	Step number (0 to 19)
jobStep	I	Step content (result step)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function sets a result step of the job.

The steps (operation steps or result steps) must be set in the amount of the number of steps specified by SetJobInfo.

Refer to "3.2 Basic processing flow" for the procedure.

GetJobStepForResult: Obtains a result step of the job.

Format:

ReturnMessage GetJobStepForResult(Int32 jobNum, Int32 stepNum, out JobStepResult jobStep, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
stepNum	I	Step number (0 to 19)
jobStep	O	Step content (result step)
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function obtains a result step of the job.

SetJobImage: Sets job images.

Format:

ReturnMessage SetJobImage(Int32 jobNum, Int32 imageNum, JobImage jobImage, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
imageNum	I	Image number (0 to 9)
jobImage	I	Image
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function sets job images.
To set an image, job information must be set with SetJobInfo.
Refer to "3.2 Basic processing flow" for the procedure.
This function directly writes to the flash memory on the instrument.

Up to 10 images can be set for one job.
The resolutions of images that can be registered on each instrument are as follows. From the top left, register data left-aligned in the amount of the size.

	26dG	26d	25d
Resolution	W: 240 H: 128	W: 240 H: 128	W: 240 H: 128

GetJobImage: Obtains job images.

Format:

ReturnMessage GetJobImage(Int32 jobNum, Int32 imageNum, out JobImage jobImage, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
jobNum	I	Job number (0 to 4)
imageNum	I	Image number (0 to 9)
jobImage	O	Image
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	The processing was completed normally.
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Failed to connect to the instrument. Or, connection is not established using Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	V1.1~	V1.0~	V1.0~	×	×	×	×	×

Description:

This function obtains job images.

ResetToFactorySetting: Resets the settings to factory default.

Format:

ReturnMessage ResetToFactorySetting(Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
V1.0~	V1.1~	V1.0~	V1.0~	V1.0~	×	V1.0~	V1.0~	V1.0~

Description:

This method resets the settings to factory default.
The measurements and targets are all cleared, and the settings are returned to the initial values.

SetFinderEnable: Enables or disables the finder function.

Format:

ReturnMessage SetFinderEnable(OnOff enable, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
enable	I	Enable/disable finder
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErInvalidParameter	The specified parameter is incorrect.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	V1.0~	V1.0~

Description:

This function enables or disables the finder.
When set to ON, images can be obtained using GetFinderImage when camera operation starts.

Immediately after starting the camera and after measurement, it may take some time for the image to stabilize depending on the illumination light source and the measurement target.

GetFinderImage: Obtains the finder image.

Format:

ReturnMessage GetFinderImage(ImageInfo image_info, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
image	O	Image data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	V1.0~	V1.0~

Description:

This function obtains the finder image for one measurement.
To obtain the finder image, operation using SetFinderEnable must be enabled.

Obtained data includes width and height information.
The data is arranged in a Z pattern in order from the top left. Obtain the data for the required size.

GetMeasurementImage: Obtains the finder image at the time of measurement.

Format:

ReturnMessage GetMeasurementImage(ImageInfo image_info, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
image_info	O	Image data
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	V1.0~	V1.0~

Description:

This function obtains the finder image at the time of measurement.
To obtain the finder image, operation using SetFinderEnable must be enabled.

Obtained data includes width and height information.
The data is arranged in a Z pattern in order from the top left. Obtain the data for the required size.

GetDetectedMask: Obtains the detected measurement area.

Format:

ReturnMessage GetDetectedMask(out MeasArea area, Int32 comPort = 0)

Arguments:

Name	I/O	Explanation
area	O	Detected measurement area
comport	I	Virtual COM Port number

Return Value:

Definition value	Explanation
KmSuccess	Completed normally
KmErNoConnect	No instrument is connected to the specified virtual COM port.
KmErCannotCommand	The current instrument does not support the specified function.
KmErConnectFailed	Execute Connect.

* Refer to the List of errors for errors not listed above.

Supported Instruments:

25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
×	×	×	×	×	×	V1.0~	V1.0~	×

Description:

This function obtains the detected measurement area.
When SetMeasurementArea is AREA_AUTO, use this API.

5. Definitions/Structures

5.1 Type definitions

Definition	Byte	Description	C/C++	VB.NET	VBA (VB6)
int8_km	1	Signed byte type	char	SByte	(Byte)
uint8_km	1	Byte type	unsigned char	Byte	Byte
int16_km	2	Short integer type	short	Short	Integer
uint16_km	2	Short integer type (unsigned)	unsigned short	UShort	(Integer)
int32_km	4	Integer type	long(int)	Integer	Long
uint32_km	4	Integer type (unsigned)	unsigned long	UInteger	(Long)
int64_km	8	Long integer type	long long	Long	
uint64_km	8	Long integer type (unsigned)	unsigned long long	ULong	
float32_km	4	Single precision floating point type	float	Single	Single
float64_km	8	Double precision floating point type	double	Double	Double

5.2 Class definition

Class DataForm (Data form class)

Format:

```
class DataForm
{
    DataType                DataType;
    IrradiationDirection    IrradiationDirection;
}
```

Variable:

Variable	Explanation
DataType	Data type
IrradiationDirection	Irradiation direction

Class SpecData (Reflectance class)

Format:

```
class SpecData
{
        List<double>    Spec;
        double          Gloss;
}
```

Variable:

Variable	Explanation
Spec	Reflectance *the DataSize number obtained by GetInstrumentInfo is used.
Gloss	Gloss value

Class MeasDataColor (measurement color data class)

Format:

```
class MeasDataColor
{
        ColorSpace                                    color;
        Dictionary<MeasDataType, List<double>>        data;
}
```

Variable:

Variable	Explanation
color	Color space
data	Measurement data

Class ColorCalData (Color calibration data class)

Format:

```
class ColorCalData
{
        List<double>    Data;
        Int32           PlateId;
}
```

Variable:

Variable	Explanation
Data	Calibration data *the DataSize number obtained by GetInstrumentInfo is used.
PlateId	Calibration plate ID

Class GlossCalData (Gloss calibration data class)

Format:

```
class GlossCalData
{
     double          Data;
     Int32           PlateId;
}
```

Variable:

Variable	Explanation
Data	Calibration data
PlateId	Calibration plate ID

Class UserCalData (User calibration data class)

Format:

```
class UserCalData
{
     List<double>    Data;
     Int             PlateId;
}
```

Variable:

Variable	Explanation
Data	Calibration data *the DataSize number obtained by GetInstrumentInfo is used.
PlateId	Calibration plate ID

Class UserCalData2 (User calibration data class)

Format:

```
class UserCalData2
{
     List<double>    Data;
     string          PlateId;
}
```

Variable:

Variable	Explanation
Data	Calibration data *the DataSize number obtained by GetInstrumentInfo is used.
PlateId	Calibration plate ID (8 characters) * If the user ID is blank, it will be treated as if there is no data.

Class UvAdjustIndex (Index data for fluorescence adjustment)

Format:

```
class UvAdjustIndex
{
   double    value;
   double    tolerance;
 }
```

Variable:

Variable	Explanation
value	Value
tolerance	Allowable width

Class UvAdjustCoef (Fluorescence adjustment coefficient)

Format:

```
class UvAdjustCoef
{
   List<double>    coefficient;
   List<double>    correction;
   List<double>    param;
   DateTime        date;
 }
```

Variable:

Variable	Explanation
coefficient	Fluorescence coefficient *the DataSize number obtained by GetInstrumentInfo is used.
correction	Fluorescence correction value *the DataSize number obtained by GetInstrumentInfo is used.
param	Parameter * Stored in the order of P, Q, C, m, n, k, and dWdS only when using the Ganz & Griesser method.
date	Fluorescence adjustment date/time

Class UvAdjustGG (Ganz & Griesser fluorescence adjustment data)

Format:

```
class UvAdjustGG
{
   Int32           count;
   List<double>    WI;
   List<double>    Tint;
 }
```

Variable:

Variable	Explanation
count	Number of samples to be used
WI	WI value for the number of samples used
Tint	Tint value for the number of samples used

Class MeasCondSMC (SMC conditions class)

Format:

```
class MeasCondSMC
{
     OnOff            enable;
     Int32             times;
     double             threshold;
}
```

Variable:

Variable	Explanation
enable	Enable or disable function
times	Number of times to average (3 to 10 times)
threshold	Threshold value level (0.01 to 9.99)

Class ColorValue (Class for accessing color space results)

Format:

```
class ColorValue
{
     string                      Id;
     ColorSpace                 ColorModeId;
     DateTime                    Date;
```

```
        Dictionary<string, double>   Value;
        Dictionary<string, string>    CharValue;
        List<Dictionary<string, double>>   AllAngleValue;
    }
```

Variable:

Variable	Explanation
Id	ID
ColorModeId	Color space ID * A value other than COLOR_MUNSELL_C can be specified.
Date	Registration/measurement date/time
Value	Color space value
CharValue	* Unused
AllAngleValue	Color space value (Use when obtaining data of multiple angles.)

Class TargetData (Target data class)

Format:

```
    class TargetData
    {
        DateTime                                                Date;
        List<Int32>                                             Group;
        string                                                  Name;
        Dictionary< IrradiationDirection, List<List<double>>>   Data;
        double                                                  Gloss;
        ColorValue                                              ColorData;
        MeasCondMode                                            Mode;
        MeasArea                                                Area;
        DataForm                                                DataForm;
    }
```

Variable:

Variable	Explanation
DateTime	Measurement (registration) date/time
Group	Group number
Name	Name
Data	Reflectance
Gloss	Gloss value
ColorData	Color space value
Mode	Measurement mode
Area	Measurement area
dataForm	Data type

Class TargetDataPack (Target data class)

Format:

```
    class TargetDataPack
    {
        DateTime                                   date;
        List<Int32>                                group_list;
        MeasType                                   meas_type;
        MeasCondMode                               meas_mode;
        MeasArea                                   meas_area;
        MeasAngle                                  meas_angle;
        LightDirection                             l_direction;
        MeasCondScie                               meas_scie;
        MeasCondUv                                 meas_uv;
        DataWarning                                warning;
        Int32                                      diagnosis;
        DataAttr                                    data_attr;
        Int32                                       warning_level;
        Dictionary<DataId, List<double>>            data;
        ColorData                                   data_color;
        string                                      name;
    }
```

Variable:

Variable	Explanation
date	Measurement (registration) date/time
group_list	Group number list
meas_type	Measurement type
meas_mode	Measurement mode
meas_area	Measurement area * For 25d and 23d data, set AREA_MAV for mea_area.
meas_angle	Measurement angle
l_direction	Irradiation direction
meas_scie	Specular component
meas_uv	UV condition * For 25d and 23d data, set UV_CUT400 for meas_uv.
warning	Warning
diagnosis	Diagnosis
data_attr	Data attribute
data	Reflectance data * Use only if data_attr is DATAATTR_SPEC. * The amount of required data will depend on the measurement mode, measurement angle, irradiation direction, and specular component. Be careful regarding the amount of required data.
data_color	Color value data * Use only if data_attr is a value other than DATAATTR_SPEC. * Specify gloss values for both data_color.data1 and data_color.data2. * The amount of required data will depend on the measurement mode, measurement angle, irradiation direction, and specular component. Be careful regarding the amount of required data.
name	Name

Class ColorData (Color data class)

Format:

```
    class ColorData
    {
        Observer                            obs1;
        Observer                            obs2;
        Illuminant                          ill1;
        Illuminant                          ill2;
        Dictionary<DataId, List<double>>    data1;
        Dictionary<DataId, List<double>>    data2;
    }
```

Variable:

Variable	Explanation
obs1	Observer 1
obs2	Observer 2
ill1	Illuminant 1
ill2	Illuminant 2
data1	Observer 1/illuminant 1 color data
data2	Observer 2/illuminant 2 color data

Class ToleranceData (Tolerance data class)

Format:

```
class ToleranceData
{
        Dictionary<Int32, ToleranceParam>   Tolerance;
}
```

Variable:

<table>
<tr><th>Variable</th><th>Explanation</th></tr>
<tr><td>Tolerance</td><td>Tolerance data
Key   Tolerance number (refer to ToleranceId)

ToleranceParam
<table>
<tr><td>Upper_enable</td><td>Upper limit enable/disable</td></tr>
<tr><td>Upper_value</td><td>Upper limit</td></tr>
<tr><td>Lower_enable</td><td>Lower limit enable/disable</td></tr>
<tr><td>Lower_value</td><td>Lower limit</td></tr>
</table></td></tr>
</table>

* The types and ranges of parameters that can be set may depend on the instrument. Refer to the manual of the instrument.

* Enter a value multiplied by 100 for Δx and Δy. Obtained values of Δx and Δy have been multiplied by 100.

Class ToleranceParam (Tolerance data class)

Format:

```
class ToleranceData
{
        byte                    Upper_enable;
        double                  Upper_value;
        byte                    Lower_enable;
        double                  Lower_value;
}
```

Variable:

Variable	Explanation
Upper_enable	Upper limit enable/disable (0: disable, 1: enable)
Upper_value	Upper limit
Lower_enable	Lower limit enable/disable (0: disable, 1: enable)
Lower_value	Lower limit

* The types and ranges of parameters that can be set may depend on the instrument. Refer to the manual of the instrument.

Class ParamtricCoef (Parametric coefficient class)

Format:

```
class ParametricCoef
{
        List<double>   CMC;
        List<double>   DeltaE94;
        List<double>   DeltaE00;
}
```

Variable:

Variable	Explanation
CMC	For CMC * The coefficients have the following order: l, c.
DeltaE94	For ΔE*94

	* The coefficients have the following order: l, c, h.
DeltaE00	For ΔE00 * The coefficients have the following order: l, c, h.

* The types and ranges of parameters that can be set may depend on the instrument. Refer to the manual of the instrument.

Class SampleData (Measurement value data class)

Format:

```
class SampleData
{
        DateTime                                                   Date;
        string                                                     Name
        SpecData                                                   Data;
        Dictionary< IrradiationDirection, List<List<double>>>   AllAngleData;
        ColorValue                                                 ColorData;
        MeasCondMode                                               Mode;
        MeasArea                                                   Area;
        DataForm                                                   DataForm;
        Int32                                                      relation_target;
}
```

Variable:

Variable	Explanation
Date	Measurement (registration) date/time
Name	Name
Data	Reflectance, gloss value
AllAngleData	Reflectance (Use when obtaining data of multiple angles.)
ColorData	Color space value
Mode	Measurement mode
Area	Measurement area
DataForm	Data type
relation_target	Related target

Class SampleDataPack (Measurement value data class)

Format:

```
class SampleDataPack
{
        DateTime                                                   date;
        MeasType                                                   meas_type;
        MeasCondMode                                               meas_mode;
        MeasArea                                                   meas_area;
        MeasAngle                                                  meas_angle;
        LightDirection                                             l_direction;
        MeasCondScie                                               meas_scie;
        MeasCondUv                                                 meas_uv;
        DataWarning                                                warning;
        Int32                                                      diagnosis;
        Int32                                                      relation_target;
        Dictionary<DataId, List<double>>                           data;
        string                                                     name;
}
```

Variable:

Variable	Explanation
date	Measurement (registration) date/time
group_list	Group number list
meas_type	Measurement type

meas_mode	Measurement mode
meas_area	Measurement area
meas_angle	Measurement angle
l_direction	Irradiation direction
meas_scie	Specular component
meas_uv	UV condition
warning	Warning
diagnosis	Diagnosis
relation_target	Related target number
data	Data * The amount of required data will depend on the measurement mode, measurement angle, irradiation direction, and specular component. Be careful regarding the amount of required data.
name	Name

Class InstrumentInfo (Instrument information class)

Format:

```
class InstrumentInfo
{
     string       Name;
     Int32        SerialNum;
     string       Version;
}
```

Variable:

Variable	Explanation
Name	Instrument name
SerialNum	Unit number
Version	Version

Class InstrumentInfoEx (Instrument information class)

Format:

```
class InstrumentInfoEx
{
     Int32        SerialNum;
     Int32        WaveLengthStart;
     Int32        WaveLengthEnd;
     Int32        WaveLengthPitch;
     Int32        DataSize;
     string       Version;
     string       Name;
}
```

Variable:

Variable	Explanation
SerialNum	Unit number
WaveLengthStart	Reflectance Wavelength range: Minimum wavelength
WaveLengthEnd	Reflectance Wavelength range: Maximum wavelength
WaveLengthPitch	Reflectance Wavelength range: Wavelength pitch
DataSize	Reflectance Size of data (determined by minimum wavelength, maximum wavelength, and wavelength pitch)
Version	Version
Name	Instrument name

Class JobInfo (Job information)

Format:

```
class JobInfo
```

```
{
    Int32          step_count;
    OnOff          step_loop;
    string         name;
 }
```

Variable:

Variable	Explanation
step_count	Number of steps to register for job (1 to 20)
step_loop	Repeat job on/off
name	Name

Class JobStepOperation (Job operation step)

Format:

```
class JobStepOperation
{
    Int32                   image_num;
    MeasType                meas_type;
    MeasCondMode            meas_mode;
    MeasArea                meas_area;
    MeasAngle               meas_angle;
    LightDirection          meas_ldirection;
    MeasCondScie            meas_scie;
    MeasCondUv              meas_uv;
    Int32                   auto_ave_times;
    Int32                   manu_ave_times;
    Int32                   relation_target;
    OnOff                   enable_meas;
    OnOff                   enable_prev;
    OnOff                   enable_next;
    OnOff                   enable_end;
    String                  name;
    string                  comment;
 }
```

Variable:

Variable	Explanation
image_num	Image number (0 to 10) * 10 is handled as “No image”.
meas_type	Measurement type
meas_mode	Measurement mode
meas_area	Measurement area
meas_angle	Measurement angle
meas_ldirection	Irradiation direction
meas_scie	Specular component
meas_uv	UV condition
auto_ave_times	Auto average count (1 to 10)
manu_ave_times	Manual average count (1 to 30)
relation_target	Related target number * Opacity attribute data cannot be set.
enable_meas	Measurement button enable/disable
eable_prev	Display previous button on/off
enable_next	Display next button on/off
enable_end	End button enable/disable
name	Name
comment	Comment

Class JobStepResult (Job result step)

Format:

```
class JobStepResult
{
    MeasCondScie          meas_scie;
    Observer              obs1;
    Observer              obs2;
    Illuminant            ill1;
    Illuminant            ill2;
    List<CustomItem>      custom_item;
    OnOff                 enable_meas;
    OnOff                 enable_prev;
    OnOff                 enable_next;
    OnOff                 enable_end;
 }
```

Variable:

Variable	Explanation
meas_scie	Specular component
obs1	Observer 1
obs2	Observer 2
ill1	Illuminant 1
ill2	Illuminant 2
custom_item	Custom items
enable_meas	Measurement button enable/disable
eable_prev	Display previous button on/off
enable_next	Display next button on/off
enable_end	End button enable/disable

Class JobImage (Job image)

Format:

```
class JobImage
{
    Int32           width;
   Int32            height;
    List<Int32>   data;
    string         name;
}
```

Variable:

Variable	Explanation
width	Width (240 fixed)
height	Height (128 fixed)
data	Image data (arranged in Z order from top left) The data is stored as 1 pixel per element. R, G, and B are each 8 bits. The data is stored right-aligned in BGR order.
name	Image name

Class ImageInfo (image)

Format:

```
class JobImage
{
    Int32           width;
   Int32            height;
    List<Int32>   data;
   }
```

Variable:

Variable	Explanation
width	Width (240 fixed)
height	Height (128 fixed)
data	Image data (arranged in Z order from top left) The data is stored as 1 pixel per element. R, G, and B are each 8 bits. The data is stored right-aligned in BGR order.

5.3 Value definition

CalStatus (Calibration status)

	Value	Explanation
StatusZero	0	Zero calibration is required
StatusWhite	1	White calibration is required
StatusGloss	2	Gloss calibration is required
StatusMeasure	3	Measurement is possible
StatusMeasureWrn	4	Measurement is possible (calibration recommended)
StatusUser	5	User calibration is required

* Dependent on instrument and version.

CalDataType (Calibration Data Type)

■ For the CM-25cG

	Value	Explanation
CALTYPE_MAV	0	MAV
CALTYPE_SAV	1	SAV

■ For the CM-26dG/CM-26d

	Value	Explanation
CALTYPE_MAV_SCI	0	MAV/SCI
CALTYPE_MAV_SCE	1	MAV/SCE
CALTYPE_SAV_SCI	2	SAV/SCI
CALTYPE_SAV_SCE	3	SAV/SCE

■ For CM-25d/CM-23d

	Value	Explanation
CALTYPE_MAV_SCI	0	MAV/SCI
CALTYPE_MAV_SCE	1	MAV/SCE

■ For the CM-M6

	Value	Explanation
CALTYPE_L_ANGLE_M15	0	Left -15°
CALTYPE_L_ANGLE_15	1	Left 15°
CALTYPE_L_ANGLE_25	2	Left 25°
CALTYPE_L_ANGLE_45	3	Left 45°
CALTYPE_L_ANGLE_75	4	Left 75°
CALTYPE_L_ANGLE_110	5	Left 110°
CALTYPE_R_ANGLE_M15	6	Right -15°
CALTYPE_R_ANGLE_15	7	Right 15°
CALTYPE_R_ANGLE_25	8	Right 25°
CALTYPE_R_ANGLE_45	9	Right 45°
CALTYPE_R_ANGLE_75	10	Right 75°
CALTYPE_R_ANGLE_110	11	Right 110°

■ For the CM-36dG

	Value	Explanation
CALTYPE_MAV_SCI	0	MAV/SCI
CALTYPE_MAV_SCE	1	MAV/SCE
CALTYPE_SAV_SCI	2	SAV/SCI
CALTYPE_SAV_SCE	3	SAV/SCE
CALTYPE_LAV_SCI	4	LAV/SCI
CALTYPE_LAV_SCE	5	LAV/SCE
CALTYPE_LMAV_SCI	6	LMAV/SCI
CALTYPE_LMAV_SCE	7	LMAV/SCE
CALTYPE_TRA	8	Transmitted(LAV fixed)

■ For the CM-36d

	Value	Explanation
CALTYPE_MAV_SCI	0	MAV/SCI
CALTYPE_MAV_SCE	1	MAV/SCE
CALTYPE_SAV_SCI	2	SAV/SCI
CALTYPE_SAV_SCE	3	SAV/SCE
CALTYPE_LAV_SCI	4	LAV/SCI
CALTYPE_LAV_SCE	5	LAV/SCE

■ For the CM-3630A

	Value	Explanation
CALTYPE_MAV	0	MAV
CALTYPE_LAV	2	LAV

MeasStatus (Measurement status)

	Value	Explanation
Idling	0	Idling state
Measuring	1	Measuring

DataType (Data type)

■ For the CM-25cG

	Value	Explanation
NO_PARAM	0xFF	NO_PARAMETER

■ For the CM-M6

	Value	Explanation
ANGLE_M15	3	-15°
ANGLE_15	4	15°
ANGLE_25	5	25°
ANGLE_45	6	45°
ANGLE_75	7	75°
ANGLE_110	8	110°

■ For the CM-26dG/CM-26d

	Value	Explanation
SC_SCI	0	SCI (UV condition: Only when number of outputs is 1)
SC_SCE	1	SCE (UV condition: Only when number of outputs is 1)
SC_SCI_UVFULL	2	SCI (UV100%)
SC_SCE_UVFULL	3	SCE (UV100%)
SC_SCI_UVCUT	4	SCI (UV cut)
SC_SCE_UVCUT	5	SCE (UV cut)
SC_SCI_UVADJ	6	SCI (UV adjustment)
SC_SCE_UVADJ	7	SCE (UV adjustment)
DATATYPE_BACKWHITE	10	Opacity white back
DATATYPE_BACKBLACK	11	Opacity black back

■ For the CM-25d/CM-23d

	Value	Explanation
SC_SCI	0	SCI
SC_SCE	1	SCE
DATATYPE_BACKWHITE	10	Opacity white back
DATATYPE_BACKBLACK	11	Opacity black back

■ For the CM-36dG

	Value	Explanation

SC_SCI	0	SCI (UV condition: Only when number of outputs is 1)
SC_SCE	1	SCE (UV condition: Only when number of outputs is 1)
SC_SCI_UVFULL	2	SCI (UV100%)
SC_SCE_UVFULL	3	SCE (UV100%)
SC_SCI_UVCUT	4	SCI (UV cut)
SC_SCE_UVCUT	5	SCE (UV cut)
SC_SCI_UVADJ	6	SCI (UV adjustment)
SC_SCE_UVADJ	7	SCE (UV adjustment)
DATATYPE_TRA	8	Transmitted(LAV fixed)

■ For the CM-36d

	Value	Explanation
SC_SCI	0	SCI
SC_SCE	1	SCE

■ For the CM-3630A

	Value	Explanation
NO_PARAM	0xFF	NO_PARAMETER
SC_UVFULL	2	(UV100%)
SC_UVCUT	4	(UV cut)
SC_UVADJ	6	(UV adjustment)

* If no number is indicated, the DataSize number obtained by GetInstrumentInfo is used.

MeasDataType (Measurement data type)

■ For the CM-25cG

	Value	Explanation
DTYPE_GLOSS	0	GU (1 item)
DTYPE_SPEC	1	Spectral data

■ For the CM-26dG/CM-36dG

	Value	Explanation
DTYPE_GLOSS	0	GU (1 item)
DTYPE_SCI	1	SCI (UV condition: Only when number of outputs is 1)
DTYPE_SCE	2	SCE (UV condition: Only when number of outputs is 1)
DTYPE_SCI_UVFULL	10	SCI (UV100%)
DTYPE_SCE_UVFULL	11	SCE (UV100%)
DTYPE_SCI_UVCUT	12	SCI (UV cut)
DTYPE_SCE_UVCUT	13	SCE (UV cut)
DTYPE_SCI_UVADJ	14	SCI (UV adjustment)
DTYPE_SCE_UVADJ	15	SCE (UV adjustment)

■ For the CM-26d

	Value	Explanation
DTYPE_SCI	1	SCI (UV condition: Only when number of outputs is 1)
DTYPE_SCE	2	SCE (UV condition: Only when number of outputs is 1)
DTYPE_SCI_UVFULL	10	SCI (UV100%)
DTYPE_SCE_UVFULL	11	SCE (UV100%)
DTYPE_SCI_UVCUT	12	SCI (UV cut)
DTYPE_SCE_UVCUT	13	SCE (UV cut)
DTYPE_SCI_UVADJ	14	SCI (UV adjustment)
DTYPE_SCE_UVADJ	15	SCE (UV adjustment)

■ For the CM-25d/CM-23d/CM-36d

	Value	Explanation
DTYPE_SCI	1	SCI
DTYPE_SCE	2	SCE

■ For the CM-M6

	Value	Explanation
DTYPE_L_ANGLE_M15	0	Left -15°
DTYPE_L_ANGLE_15	1	Left 15°
DTYPE_L_ANGLE_25	2	Left 25°
DTYPE_L_ANGLE_45	3	Left 45°
DTYPE_L_ANGLE_75	4	Left 75°
DTYPE_L_ANGLE_110	5	Left 110°
DTYPE_R_ANGLE_M15	6	Right -15°
DTYPE_R_ANGLE_15	7	Right 15°
DTYPE_R_ANGLE_25	8	Right 25°
DTYPE_R_ANGLE_45	9	Right 45°
DTYPE_R_ANGLE_75	10	Right 75°
DTYPE_R_ANGLE_110	11	Right 110°
DTYPE_DP_ANGLE_M15	12	Double path -15°
DTYPE_DP_ANGLE_15	13	Double path 15°
DTYPE_DP_ANGLE_25	14	Double path 25°
DTYPE_DP_ANGLE_45	15	Double path 45°
DTYPE_DP_ANGLE_75	16	Double path 75°
DTYPE_DP_ANGLE_110	17	Double path 110°

■ For the CM-3630A

	Value	Explanation
DTYPE_SPEC	1	(UV condition: Only when number of outputs is 1)
DTYPE_SPEC_UVFULL	10	(UV100%)
DTYPE_SPEC_UVCUT	12	(UV cut)
DTYPE_SPEC_UVADJ	14	(UV adjustment)

* If no number is indicated, the DataSize number obtained by GetInstrumentInfo is used.

IrradiationDirection (Irradiation Direction)

■ For the CM-25cG/26dG/26d/25d/23d/36dG/36d/3630A

	Value	Explanation
NO_PARAM	0xFF	No parameter

■ For the CM-M6

	Value	Explanation
LEFT_DIRECTION	0	LEFT
RIGHT_DIRECTION	1	RIGHT
AVERAGE	2	Double Path

LightDirection (Irradiation Direction**)**

	Value	Explanation
LDIRECTION_NONE	0	None
LDIRECTION_L	0x01	LEFT
LDIRECTION_R	0x02	RIGHT
LDIRECTION_DP	0x04	Double Path

* Dependent on instrument and version.

CondUvAdjust (Fluorescence adjustment conditions)

	Value	Explanation
UVADJ_PROFILE	0	Profile
UVADJ_WI	1	WI

UVADJ_TINT	2	Tint
UVADJ_WITINT	3	WI & Tint
UVADJ_BRIGHTNESS	4	ISO brightness
UVADJ_GG	5	Ganz & Griesser
UVADJ_NONE	-1	No condition

* Dependent on instrument and version.

UvAdjustDataType (Fluorescence coefficient data type)

	Value	Explanation
UVADJ_DATATYPE_SCI	0	SCI
UVADJ_DATATYPE_SCE	1	SCE
UVADJ_DATATYPE_NONE	-1	No condition

* Dependent on instrument and version.

MeasType (Measurement Type)

	Value	Explanation
MEASTYPE_REF	0	Reflected
MEASTYPE_TRA	1	Transmitted
MEASTYPE_NONE	-1	No condition

* Dependent on instrument and version.

MeasArea (Measurement area)

	Value	Explanation
AREA_MAV	0	MAV
AREA_SAV	1	SAV
AREA_LAV	2	LAV
AREA_LMAV	3	LMAV
AREA_AUTO	10	Detected mask
AREA_NONE	-1	Area fixed
AREA_MOVING	-99	Moving measurement area

* Dependent on instrument and version.

MeasAngle (Measurement angle)

	Value	Explanation
MEAS_ANGLE_M15	0x01	-15°
MEAS_ANGLE_15	0x02	15°
MEAS_ANGLE_25	0x04	25°
MEAS_ANGLE_45	0x08	45°
MEAS_ANGLE_75	0x10	75°
MEAS_ANGLE_110	0x20	110°
MEAS_ANGLE_NONE	-1	No condition

* Dependent on instrument and version.

MeasCondMode (Measurement Mode)

	Value	Explanation
MeasModeColorAndGloss	0	Obtain measured color and gloss
MeasModeColorOnly	1	Obtain measured color only
MeasModeGlossOnly	2	Obtain gloss only

MeasModeOpacity	3	Opacity

* Dependent on instrument and version.

MeasCondScie (Specular Component)

	Value	Explanation
SC_SCI	0	SCI
SC_SCE	1	SCE
SC_SCIE	2	SCI+SCE
SC_NONE	-1	No condition

* Dependent on instrument and version.

MeasCondUv (UV Condition)

	Value	Explanation
UV_100	0	UV100%
UV_CUT400	1	UV Cut 400 nm
UV_CUT400N	3	UV adjustment Cut 400 nm, normal illumination
UV_100_CUT400	7	UV 100% + UV cut 400 nm
UV_100_CUT420	8	UV 100% + UV cut 420 nm
UV_100_CUT400LOW	15	UV 100% + UV cut 400 nm, low illumination
UV_100_CUT420LOW	16	UV 100% + UV cut 420 nm, low illumination
UV_100_CUT400N	9	UV 100% + UV cut 400 nm + UV adjustment cut 400 nm, normal illumination
UV_100_CUT400L	10	UV 100% + UV cut 400 nm + UV adjustment cut 400 nm, low illumination
UV_100_CUT420N	11	UV 100% + UV cut 420 nm + UV adjustment cut 420 nm, normal illumination
UV_100_CUT420L	12	UV 100% + UV cut 420 nm + UV adjustment cut 420 nm, low illumination
UV_NONE	-1	No condition

* Dependent on instrument and version.

DataId (Data Type)

■ For the CM-25cG

	Value	Explanation
DATAID_GLOSS	0	GU (1 item)
DATAID_SPEC	1	Spectral data

■ For the CM-26dG

	Value	Explanation
DATAID_GLOSS	0	GU (1 item)
DATAID_SCI	1	SCI
DATAID_SCE	2	SCE
DATAID_BACKWHITE	3	Opacity white back
DATAID_BACKBLACK	4	Opacity black back

■ For the CM-26d/CM-25d/CM-23d

	Value	Explanation
DATAID_SCI	1	SCI
DATAID_SCE	2	SCE
DATAID_BACKWHITE	3	Opacity white back
DATAID_BACKBLACK	4	Opacity black back

■ For the CM-M6

	Value	Explanation
DATAID_L_ANGLE_M15	0	Left -15°
DATAID_L_ANGLE_15	1	Left 15°
DATAID_L_ANGLE_25	2	Left 25°
DATAID_L_ANGLE_45	3	Left 45°
DATAID_L_ANGLE_75	4	Left 75°
DATAID_L_ANGLE_110	5	Left 110°
DATAID_R_ANGLE_M15	6	Right -15°
DATAID_R_ANGLE_15	7	Right 15°
DATAID_R_ANGLE_25	8	Right 25°
DATAID_R_ANGLE_45	9	Right 45°
DATAID_R_ANGLE_75	10	Right 75°
DATAID_R_ANGLE_110	11	Right 110°
DATAID_DP_ANGLE_M15	12	Double path -15°
DATAID_DP_ANGLE_15	13	Double path 15°
DATAID_DP_ANGLE_25	14	Double path 25°
DATAID_DP_ANGLE_45	15	Double path 45°
DATAID_DP_ANGLE_75	16	Double path 75°
DATAID_DP_ANGLE_110	17	Double path 110°

* If no number is indicated, the DataSize number obtained by GetInstrumentInfo is used.

DataAttr (Data Attribute)

	Value	Explanation
DATAATTR_SPEC	0	Spectral data
DATAATTR_LAB	1	L*a*b*
DATAATTR_HLAB	2	Hunter Laboratory
DATAATTR_XYZ	3	XYZ

DataWarning (Data Warning)

	Value	Explanation
KmWrBattery	0x01	Low battery voltage.
KmWrCalibration	0x02	Recalibration required. It has been a long time since calibration.
KmWrPreAnnualCalibraton	0x04	Periodical calibration required soon.
KmWrAnnualCalibraton	0x08	Periodical calibration required.
KmWrLampForColor	0x10	Reflectance outside range of guaranteed performance.
KmWrOutOfColorRange	0x20	Low illuminant intensity for color measurement.
KmWrOutOfGlossRange	0x40	Gloss outside range of guaranteed performance.
KmWrLampForGloss	0x80	Low illuminant intensity for gloss measurement.

SaveMode (Save Method)

	Value	Explanation
SAVEMODE_AUTO	0	Auto save
SAVEMODE_MANUAL	1	Manual save

DisplayType (Display type)

	Value	Explanation
DISPTYPE_ABS	0x001	Absolute values
DISPTYPE_DIF	0x002	Color difference
DISPTYPE_ABSDIF	0x004	Absolute value & color difference
DISPTYPE_CUSTOM	0x008	Custom
DISPTYPE_GRAPH_ABS	0x010	Absolute value graph

DISPTYPE_GRAPH_DIF	0x020	Color difference graph
DISPTYPE_GRAPH_REF	0x040	Spectral graph
DISPTYPE_PASS_FAIL	0x080	Pass/fail judgment
DISPTYPE_MI	0x100	MI
DISPTYPE_GRAPH_LINE	0x200	Line graph
DISPTYPE_AUDI2000_EC	0x400	Audi2000 ΔEc
DISPTYPE_AUDI2000_EP	0x800	Audi2000 ΔEp

* Dependent on instrument and version.

Observer (Observation field)

	Value	Explanation
Deg02	0	2°
Deg10	1	10°

Illuminant (Observation illuminant)

	Value	Explanation
ILL_None	0	None
ILL_A	1	A
ILL_C	2	C
ILL_D50	3	D50
ILL_D65	4	D65
ILL_ID50	5	ID50
ILL_ID65	6	ID65
ILL_F2	7	F2
ILL_F6	8	F6
ILL_F7	9	F7
ILL_F8	10	F8
ILL_F10	11	F10
ILL_F11	12	F11
ILL_F12	13	F12
ILL_USER	14	User

* Dependent on instrument and version.

ColorSpace (Color space)

	Value	Explanation
COLOR_LAB	0	L*a*b*
COLOR_LCH	1	L*C*h
COLOR_HLAB	2	Hunter Laboratory
COLOR_YXY	3	Yxy
COLOR_XYZ	4	XYZ
COLOR_MUNSELL_C	5	Munsell (C)
COLOR_DEFAULT(*)	255	No calculation

* COLOR_DEFAULT is used in ColorModeId of the ColorValue class. Color is not output when this is used.

* Dependent on instrument and version.

Equation (Color difference equation)

	Value	Explanation
EQUATION_DE1976	0	ΔE*ab
EQUATION_CMC	1	CMC
EQUATION_DE1994	2	ΔE*94

EQUATION_DE2000	3	ΔE00
EQUATION_DEH	4	ΔE(Hunter)
EQUATION_DEP	5	ΔEp(DIN6175)
EQUATION_DEC	6	ΔEc(DIN6175)
EQUATION_DE99o	7	ΔE99o(DIN99o)

* Dependent on instrument and version.

CustomIndex (Custom Items)

	Value	Explanation
NONE	0	None
L	1	L*
a	2	a*
b	3	b*
C	4	C*
h	5	h
HL	6	L(Hunter)
Ha	7	a(Hunter)
Hb	8	b(Hunter)
X	9	X
Y	10	Y
Z	11	Z
x	12	x
y	13	y
MslH	14	H
MslV	15	V
MslC	16	C
WI_E313_73	17	WI(E313-73)
WI_CIE	18	WI(CIE)
Tint_CIE	19	Tint(CIE)
YI_E313	20	YI(E313)
YI_D1925	21	YI(D1925)
B_ISO	22	B(ISO)
GU	23	GU
USER_E1	24	UserE1
USER_C1	25	UserC1
USER_E2	26	UserE2
USER_C2	27	UserC2
USER_E3	28	UserE3
USER_C3	29	UserC3
Gloss8	30	8° gloss
WI_GANZ	31	WI(Ganz)
TINT_GANZ	32	Tint(Ganz)
DELTA_L	-1	ΔL*
DELTA_a	-2	Δa*
DELTA_b	-3	Δb*
DELTA_C	-4	ΔC*
DELTA_h	-5	ΔH*
DELTA_HL	-6	ΔL(Hunter)
DELTA_Ha	-7	Δa(Hunter)
DELTA_Hb	-8	Δb(Hunter)
DELTA_X	-9	ΔX
DELTA_Y	-10	ΔY
DELTA_Z	-11	ΔZ
DELTA_x	-12	Δx
DELTA_y	-13	Δy

DELTA_WI_E313_73	-14	ΔWI(E313-73)
DELTA_WI_CIE	-15	ΔWI(CIE)
DELTA_Tint_CIE	-16	Δtint(CIE)
DELTA_YI_E313	-17	ΔYI(E313)
DELTA_YI_D1925	-18	ΔYI(D1925)
DELTA_B_ISO	-19	ΔB(ISO)
DELTA_GU	-20	ΔGU
MI	-21	MI
DELTA_E_ab	-22	ΔE*ab
CMC	-23	CMC
DELTA_E_94	-24	ΔE*94
DELTA_E_00	-25	ΔE00
DELTA_HE	-26	ΔE(Hunter)
DELTA_E_99o	-27	ΔE99o
Strength_XYZ	-28	StrengthXYZ
Strength_X	-29	StrengthX
Strength_Y	-30	StrengthY
Strength_Z	-31	StrengthZ
GreyScale	-32	GreyScale
DELTA_WI_GANZ	-33	ΔWI(Ganz)
DELTA_TINT_GANZ	-34	ΔTint(Ganz)

* Dependent on instrument and version.

DisplayData (Display Data)

	Value	Explanation
AVERAGE	0	AVERAGE(DP)
LEFT	1	LEFT

* Dependent on instrument and version.

FilterIndex (Filter settings)

	Value	Explanation
OFF	0	OFF
FILTER_SAVE	1	Displays only the saved data
FILTER_GROUP	2	Displays only the data corresponding to the specified group number

* Dependent on instrument and version.

InstrumentMode (Instrument Mode)

	Value	Explanation
INSTRUMENTMODE_NORMAL	0	Normal mode
INSTRUMENTMODE_SIMPLE	1	Simple mode

* Dependent on instrument and version.

UserType (User Type)

	Value	Explanation
USERTYPE_ADMIN	0	Administrator
USERTYPE_WORKER	1	Operator

* Dependent on instrument and version.

ScreenDirection (Display Direction of Screen)

	Value	Explanation
SCREENDIR_0	0	Not rotated
SCREENDIR_180	1	Rotated 180°

* Dependent on instrument and version.

DateFormat (Date format)

	Value	Explanation
YYMMDD	0	YYYY/MM/DD
MMDDYY	1	MM/DD/YYYY
DDMMYY	2	DD/MM/YYYY

Language (Language)

	Value	Explanation
LANGUAGE_ENGLISH	0	English
LANGUAGE_JAPANESE	1	Japanese
LANGUAGE_GERMAN	2	German
LANGUAGE_FRENCH	3	French
LANGUAGE_SPANISH	4	Spanish
LANGUAGE_ITALIAN	5	Italian
LANGUAGE_CHINESE_S	6	Chinese (simplified)
LANGUAGE_PORTUGUESE	7	Portuguese
LANGUAGE_RUSSIAN	8	Russian
LANGUAGE_POLISH	9	Polish
LANGUAGE_TURKISH	10	Turkish

JobStepType (Job Step Type)

	Value	Explanation
JOB_STEPTYPE_OPERATION	0	Operation step
JOB_STEPTYPE_RESULT	1	Result step

OnOff (ON/OFF)

	Value	Explanation
OFF	0	OFF
ON	1	ON

IsThereData (YES/NO)

	Value	Explanation
NO	0	NO
YES	1	YES

DateType (Date/Time Type)

	Value	Explanation
DATETYPE_COLOR	0	Color
DATETYPE_GLOSS	1	Gloss

ToleranceId (Tolerance ID)

	Value	Explanation
TOLERANCE_ID_L	-1	ΔL*
TOLERANCE_ID_A	-2	Δa*
TOLERANCE_ID_B	-3	Δb*
TOLERANCE_ID_C	-4	ΔC*
TOLERANCE_ID_H	-5	ΔH*
TOLERANCE_ID_HL	-6	ΔL(Hunter)
TOLERANCE_ID_HA	-7	Δa(Hunter)
TOLERANCE_ID_HB	-8	Δb(Hunter)
TOLERANCE_ID_X	-9	ΔX
TOLERANCE_ID_Y	-10	ΔY
TOLERANCE_ID_Z	-11	ΔZ
TOLERANCE_ID_SX	-12	Δx
TOLERANCE_ID_SY	-13	Δy
TOLERANCE_ID_WI_E	-14	ΔWI(E313-73)
TOLERANCE_ID_WI_C	-15	ΔWI(CIE)
TOLERANCE_ID_TINT_C	-16	ΔTint(CIE)
TOLERANCE_ID_YI_E	-17	ΔYI(E313)
TOLERANCE_ID_YI_D	-18	ΔYI(D1925)
TOLERANCE_ID_B_ISO	-19	ΔB(ISO)
TOLERANCE_ID_GU	-20	ΔGU
TOLERANCE_ID_MI	-21	MI
TOLERANCE_ID_DE	-22	ΔE*ab
TOLERANCE_ID_CMC	-23	CMC
TOLERANCE_ID_DE94	-24	ΔE*94
TOLERANCE_ID_DE00	-25	ΔE00
TOLERANCE_ID_DEH	-26	ΔE(Hunter)
TOLERANCE_ID_DEP_DIN6175	-27	ΔEp(DIN6175)
TOLERANCE_ID_DEC_DIN6175	-28	ΔEc(DIN6175)
TOLERANCE_ID_FF	-29	ΔFF
TOLERANCE_ID_DE99O	-30	ΔE99o
TOLERANCE_ID_DEC_AUDI2000	-31	ΔEc(Audi2000)
TOLERANCE_ID_MDEC_AUDI2000	-32	ΔEc average (Audi2000)
TOLERANCE_ID_DECM_AUDI2000	-33	ΔEc maximum (Audi2000)
TOLERANCE_ID_DEP_AUDI2000	-34	ΔEp(Audi2000)
TOLERANCE_ID_MDEP_AUDI2000	-35	ΔEp average (Audi2000)
TOLERANCE_ID_DEPM_AUDI2000	-36	ΔEp maximum (Audi2000)
TOLERANCE_ID_DSTRENGTH_XYZ	-37	ΔStrengthXYZ
TOLERANCE_ID_DSTRENGTH_X	-38	ΔstrengthX
TOLERANCE_ID_DSTRENGTH_Y	-39	ΔstrengthY
TOLERANCE_ID_DSTRENGTH_Z	-40	ΔstrengthZ
TOLERANCE_ID_DOPACITY	-41	Opacity difference
TOLERANCE_ID_DGRAYSCALE	-42	Grayscale difference
TOLERANCE_ID_WI_G	-43	ΔWI(Ganz)
TOLERANCE_ID_TINT_G	-44	ΔTint(Ganz)

* Dependent on instrument and version.

6. Errors/Warnings

6.1 List of errors

Error ID	Value		
KmSuccess	0	Description	The processing was completed normally.
		Action	-
KmErNoConnect	10	Description	No instrument is connected to the specified virtual COM port.
		Action	Check the following: • Is the instrument powered on? • Is the cable correctly connected? • Is the COM port number correct? • No other software is controlling the instrument?
		Attention	The instrument may become disconnected if continuous measurements are performed at a short measurement interval. In this case, turn off the power to the instrument and then turn it on again.
KmErInvalidParameter	25	Description	The specified parameter is incorrect.
		Action	• Check the input range and enter an appropriate value.
KmErCannotCommand	30	Description	The current instrument does not support the specified function.
		Action	-
KmErNoData	45	Description	No data
		Action	• The necessary data must be registered beforehand.
KmErDataProtected	46	Description	The data is protected.
		Action	• Cancel data protection before performing the operation.
KmErOutOfRangeValue	50	Description	The value is outside the range that can be measured by the instrument.
		Action	-
KmErConnectFailed	100	Description	Failed to connect to the instrument. Or, connection is not established using Connect.
		Action	• Perform operation after establishing the connection using Connect.
KmErDevice	110	Description	A device in the instrument is malfunctioning.
		Action	Check again after restarting the instrument. If this problem persists, contact Customer Service.
KmErAd	111	Description	The A/D converter in the instrument has malfunctioned.
		Action	* If this problem persists even after the instrument is restarted, contact Customer Service.
KmErCharge	112	Description	Charging of the light emission circuit in the instrument has malfunctioned.
		Action	* If this problem persists even after the instrument is restarted, contact Customer Service.
KmErFlash	113	Description	The light emission circuit in the instrument has malfunctioned.
		Action	* If this problem persists even after the instrument is restarted, contact Customer

			Service.
KmErFinder	114	Description	Operation is not possible because the finder is open.
		Action	• Close the finder before performing the operation. * If this error occurs even when the finder is closed, contact Customer Service.
KmErCalculation	115	Description	The calculation cannot be performed because the required information is lacking.
		Action	* If this problem persists even after the instrument is restarted, contact Customer Service.
KmErCamera	116	Description	The camera for obtaining the finder image is not in operation.
		Action	Enable operation with SetFinderEnable.
KmErMeasuring	117	Description	The camera cannot be used because instrument is measuring.
		Action	Wait until measurement is completed.
KmErMotorDriving	118	Description	This API cannot be used because the motor is being driven.
		Action	Wait until the motor drive is completed.
KmErCalibration	120	Description	Calibration was not executed in the correct procedure.
		Action	• Is the zero calibration box used for zero calibration? • Is the white calibration plate used for white calibration? • Is the gloss calibration plate used for gloss calibration? • Does the target mask agree with the measurement area?
KmErCalibrationRequired	130	Description	Necessary calibration was not executed beforehand.
		Action	・Zero calibration must be completed before performing white calibration. ・White calibration must be completed before performing measurements. ・Gloss calibration must be completed before performing measurements.
KmErNoCalibrationData	131	Description	Calibration data is not set.
		Action	• Set calibration data.
KmErTiltDetection	140	Description	The instrument is tilted.
		Action	• Install the instrument correctly for the sample.
KmErNotUse	170	Description	This setting cannot be used due to its combination with another setting.
		Action	• Change the other setting to solve this problem. • For the CM-36dG, when the UV condition is UV adjustment, the coefficient may not be set
KmErDisagreeCond	171	Description	This cannot be set because the conditions do not agree.
		Action	• Data of the same mode must be associated. For example, opacity data cannot be associated with color+gloss data.

KmErUvAdjust	172	Description	The measurement sample does not contain fluorescence.
		Action	Be sure to measure an appropriate sample.
KmErBattery	180	Description	Low battery voltage.
		Action	• Charge the battery or connect the instrument to a power supply. * If this problem persists even after charging, contact Customer Service.
KmErMemory	181	Description	Reading or writing the memory in the instrument has failed.
		Action	* If this problem persists even after the instrument is restarted, contact Customer Service.
KmErMotor	182	Description	The motor in the instrument has malfunctioned.
		Action	* If this problem persists even after the instrument is restarted, contact Customer Service.
KmErNotSupported	190	Description	The instrument supports the function, but the function cannot be used.
		Action	• The format being used is not supported. Use another format of the function. • The SDK version is old and the data cannot be used. Upgrade the SDK to a more recent version.
KmErCalculateColor	195	Description	A color value cannot be calculated from reflectance.
		Action	
KmErCalculateCoef	196	Description	The fluorescence coefficient cannot be calculated.
		Action	Check the tolerance.
KmEr	200	Description	An unexpected error has occurred.
		Action	Check again after restarting the instrument.

6.2 List of warnings

Warning ID	Description
WR1	(Battery) Low battery voltage.
WR2	(Time) Recalibration required. It has been a long time since calibration.
WR3	(Time) Periodical calibration required soon.
WR4	(Time) Periodical calibration required. Perform periodical calibration.
WR5	(Reliability) Low light intensity in illuminant for color measurements.
WR6	(Reliability) Reflectance outside range of guaranteed performance.
WR7	(Reliability) Gloss outside range of guaranteed performance.
WR8	(Reliability) Low light intensity in illuminant for gloss measurements.
WR9	The deadline for wavelength correction is approaching.
WR10	The wavelength correction deadline has passed.
WR11	Low light intensity in illuminatin for wavelength correction.
WR12	Lack of light intensity in illuminatin for wavelength correction.

	* If this problem persists even after the instrument is restarted, contact Customer Service.
WR13	Temperture outside range of guranteed wavelength correction performance. The correction accuracy has decreased.
WR14	Correction amount outside range of guranteed wavelength correction perfomance. The correction accuracy has decreased. * If this problem persists even after the instrument is restarted, contact Customer Service.

Appendix A. Available character codes

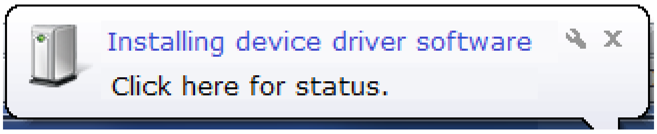
The character codes that can be used for names and comments are listed below.

	00	10	20	30	40	50	60	70
0			(sp)	0	@	P	`	p
1			!	1	A	Q	a	q
2			“	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			’	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C				<	L	¥	l	|
D			-	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

Appendix B. Installing the device driver

The device driver for the instrument must be installed in advance to connect the instrument to the PC via USB.

First connect the instrument to the PC, and then turn on the power supply to the instrument.
The driver installation will begin automatically. For Windows 7, the "Installing device driver software" popup will be displayed on the taskbar at the bottom right of the screen.

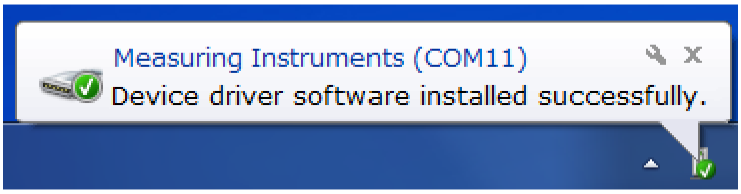


Installing device driver software
Click here for status.

The device driver that is installed may not function correctly due to Windows 10. For this reason, manually install the "KMMIUSB.INF" device driver according to the following installation procedure.

Automatic installation

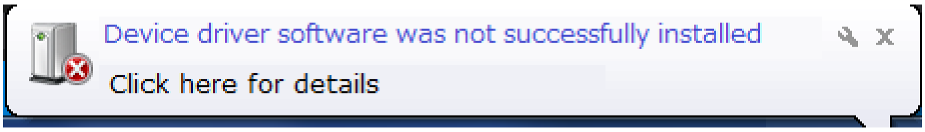
The installation is finished if the automatic installation was successful.



Measuring Instruments (COM11)
Device driver software installed successfully.

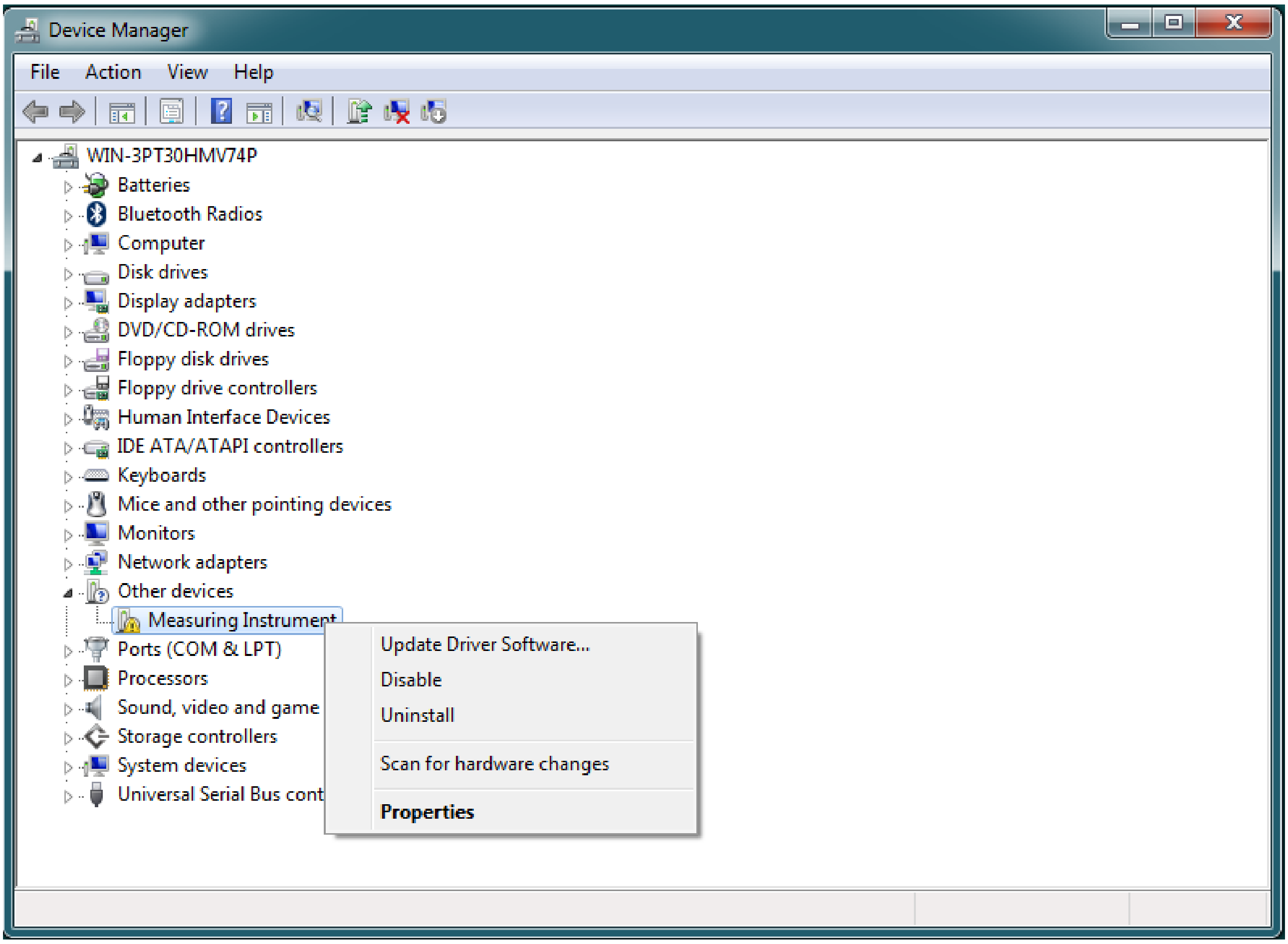
Manual installation

If the automatic installation has failed, use the following procedure to perform manual installation.

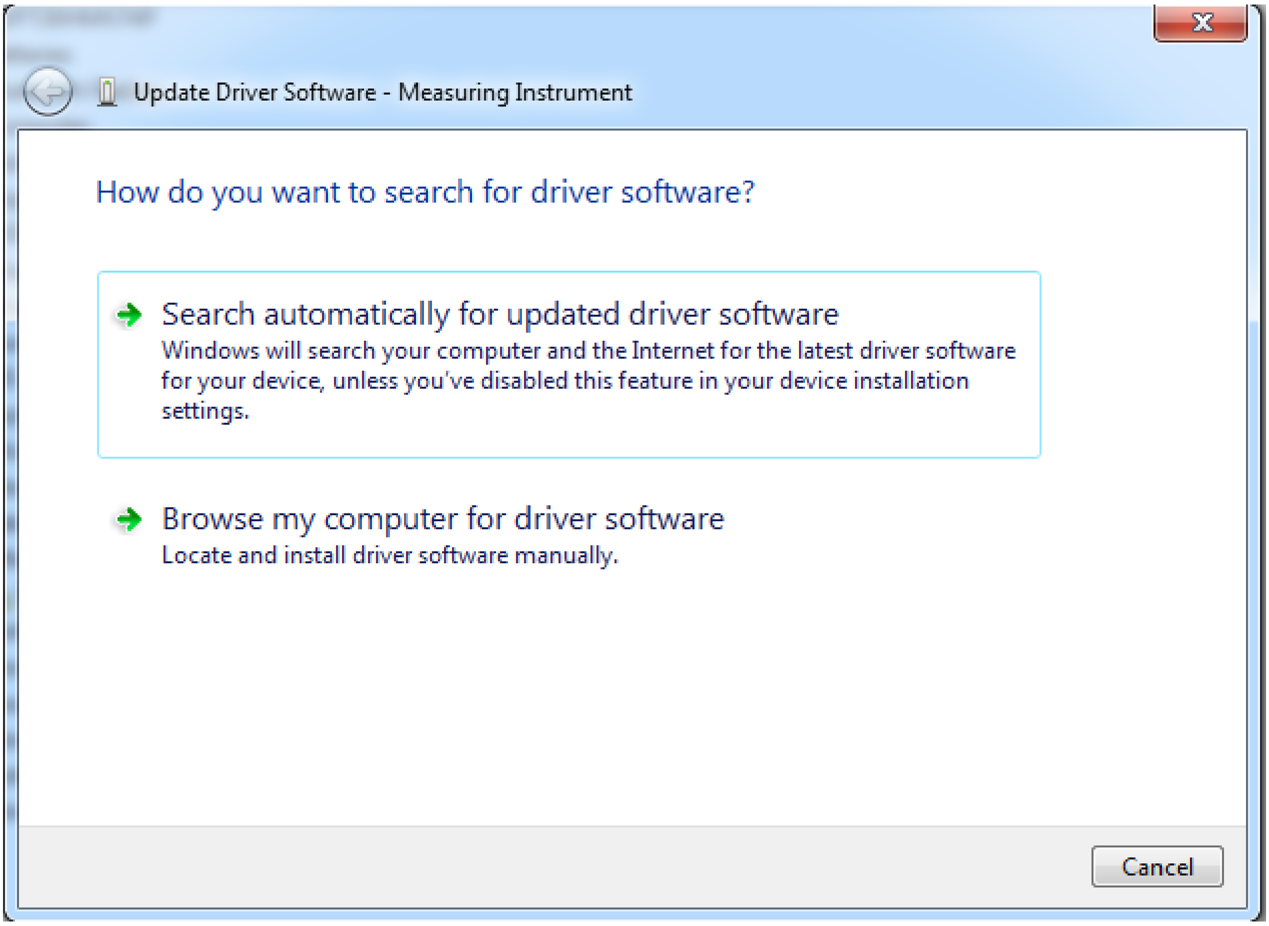


Device driver software was not successfully installed
Click here for details

Open Device Manager, right-click [Other devices]-[Measuring Instruments], and then click [Update Driver Software]. As shown in the following screenshot, the warning symbol will be added to [Measuring Instruments] if the driver installation has failed.



Click [Browse my computer for driver software].



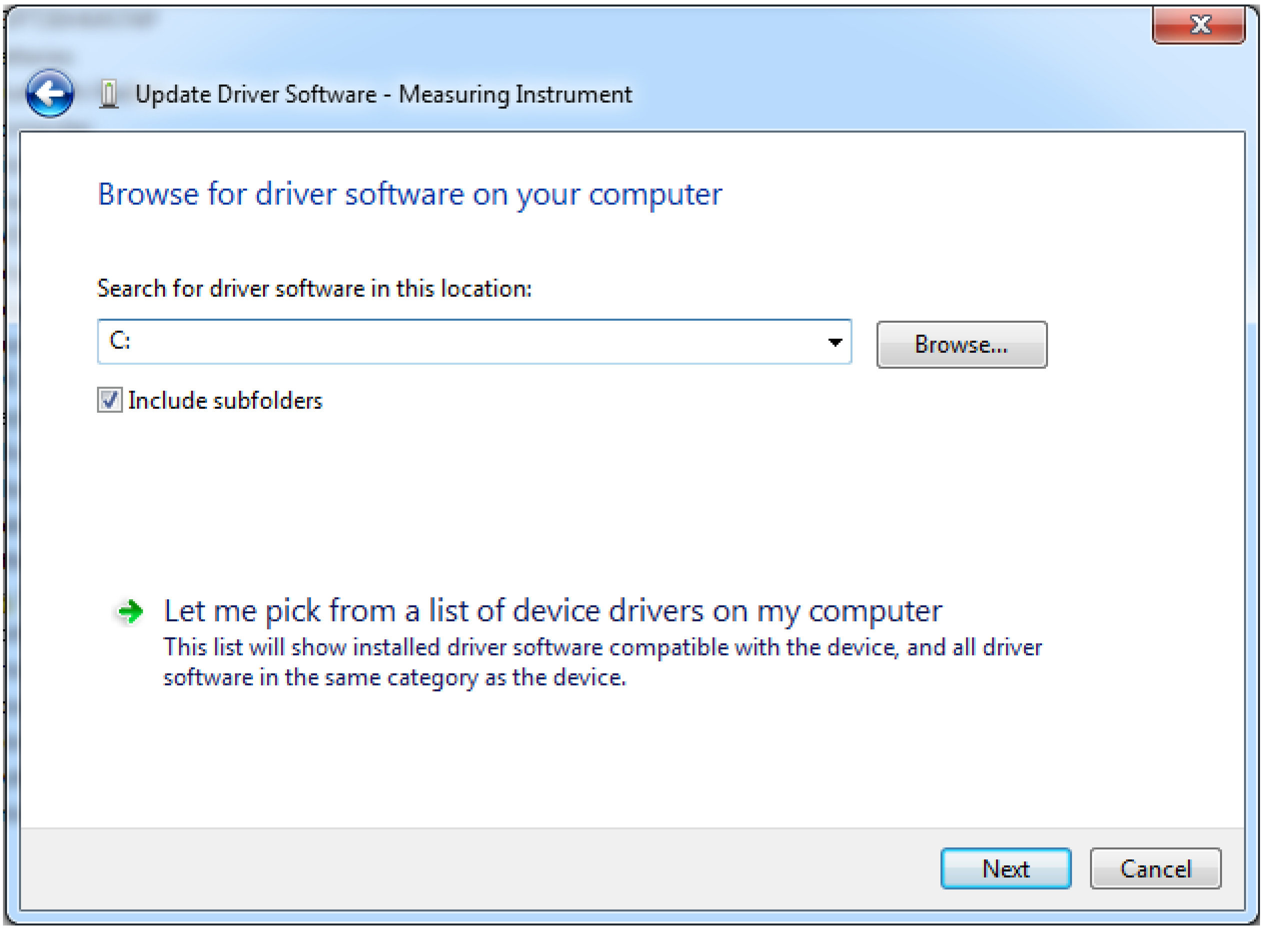
Update Driver Software - Measuring Instrument

How do you want to search for driver software?

➔ Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

➔ Browse my computer for driver software
Locate and install driver software manually.

Click [Let me pick from a list of device drivers on my computer].



Update Driver Software - Measuring Instrument

Browse for driver software on your computer

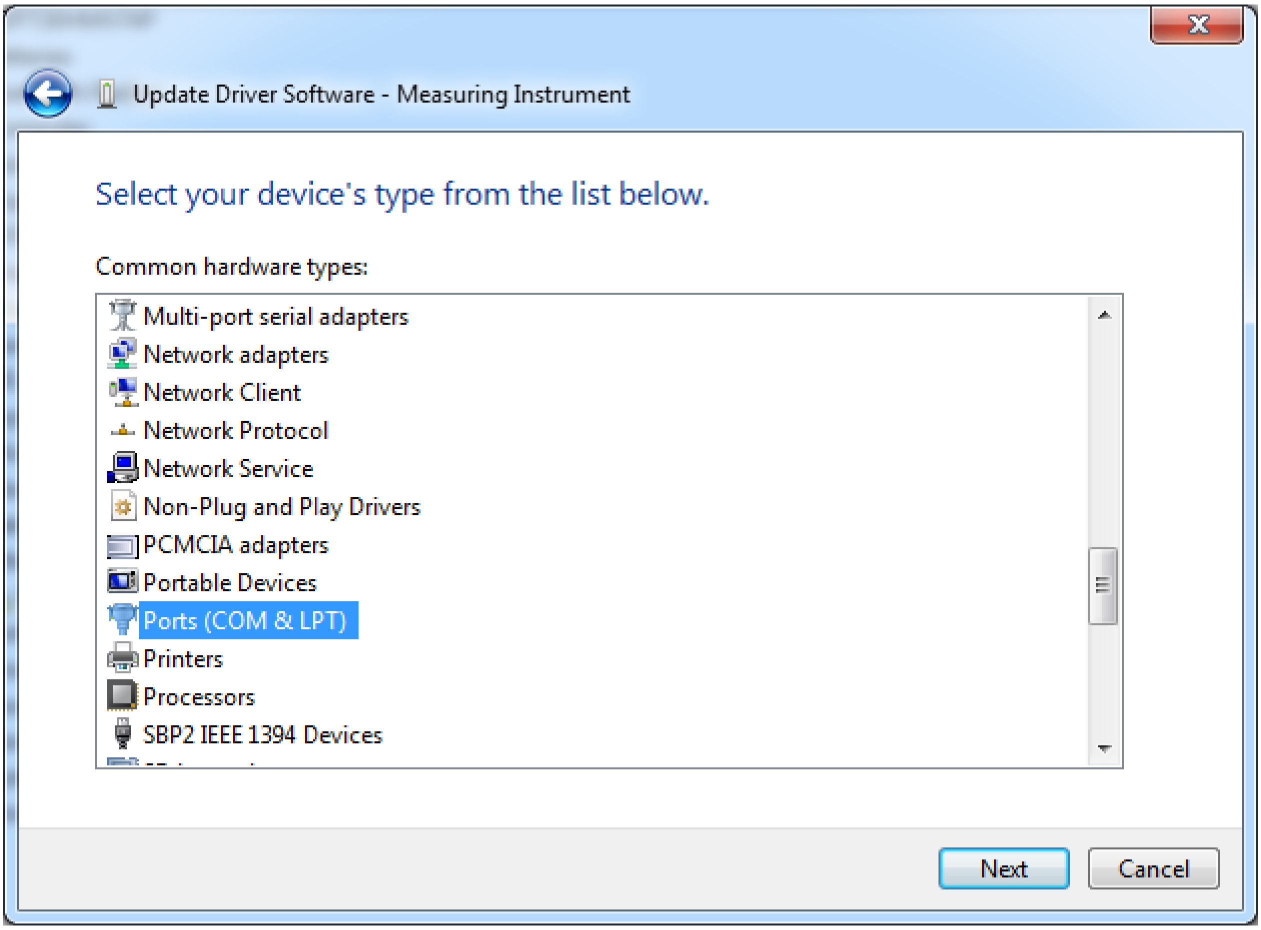
Search for driver software in this location:

C:

☑ Include subfolders

Let me pick from a list of device drivers on my computer
This list will show installed driver software compatible with the device, and all driver software in the same category as the device.

From the list of common hardware types, click [Ports (COM & LPT)], and then click [Next].



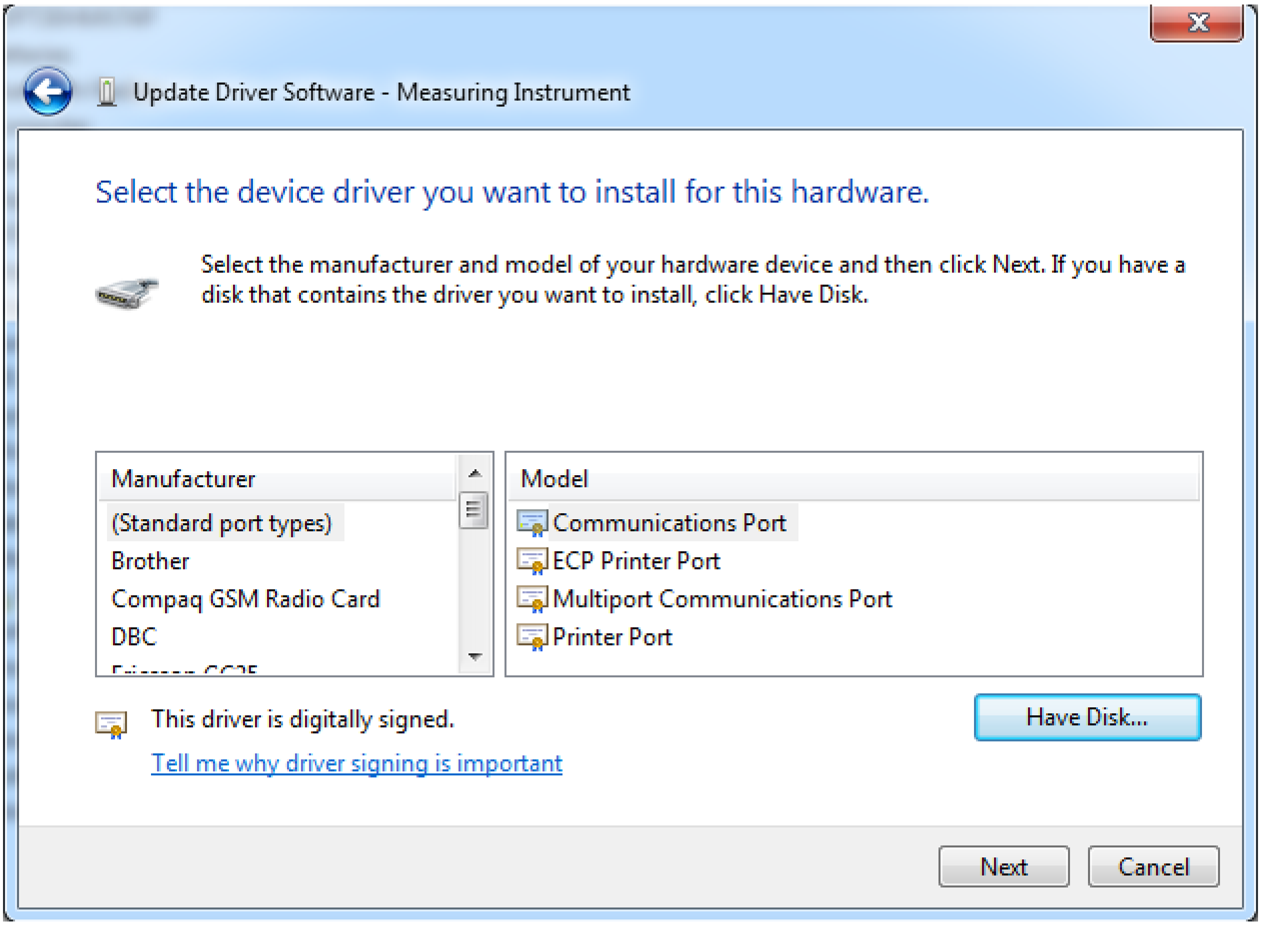
Update Driver Software - Measuring Instrument

Select your device's type from the list below.

Common hardware types:

- Multi-port serial adapters
- Network adapters
- Network Client
- Network Protocol
- Network Service
- Non-Plug and Play Drivers
- PCMCIA adapters
- Portable Devices
- Ports (COM & LPT)
- Printers
- Processors
- SBP2 IEEE 1394 Devices

Click [Have Disk].



Update Driver Software - Measuring Instrument

Select the device driver you want to install for this hardware.

Select the manufacturer and model of your hardware device and then click Next. If you have a disk that contains the driver you want to install, click Have Disk.

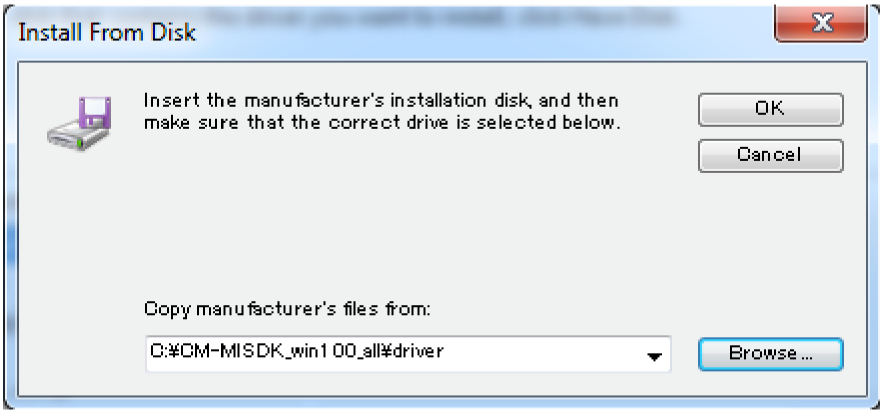
Manufacturer
(Standard port types)
Brother
Compaq GSM Radio Card
DBC

Model
Communications Port
ECP Printer Port
Multiport Communications Port
Printer Port

This driver is digitally signed.

Tell me why driver signing is important

Click [Browse].



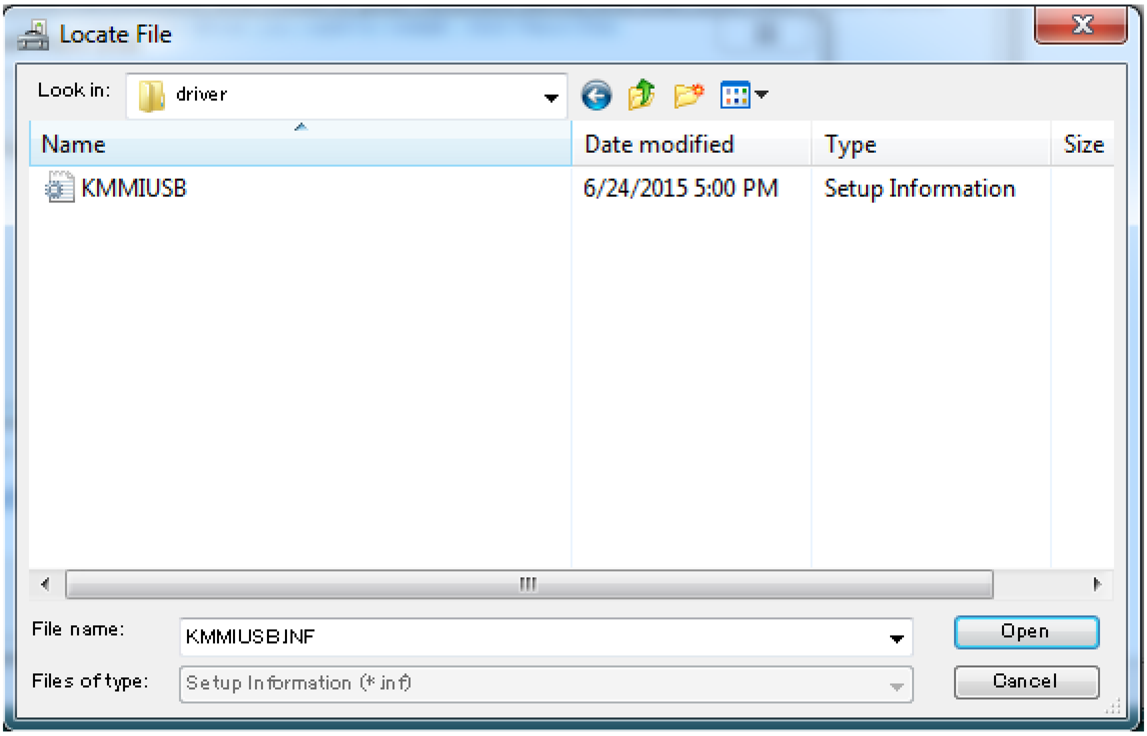
Install From Disk

Insert the manufacturer's installation disk, and then make sure that the correct drive is selected below.

Copy manufacturer's files from:

C:¥CM-MISDK_win1 00_all¥driver

Select the “KMMIUSB.INF” file in “cm-misdk_verXXXrX/driver/”, and then click [Open].



Locate File

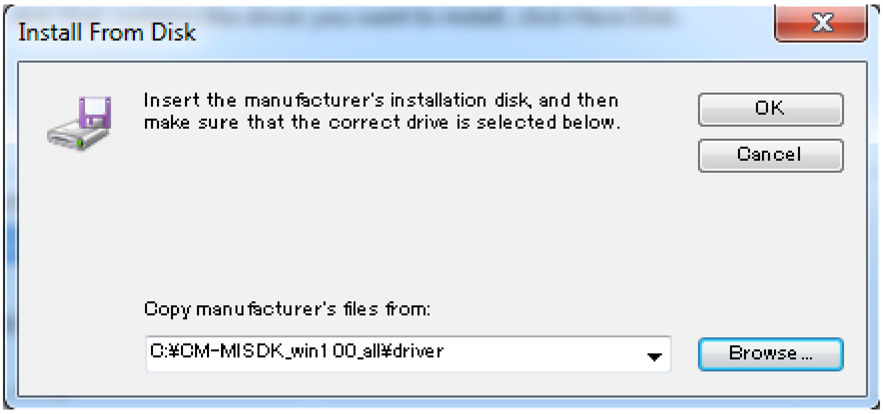
Look in: driver

Name	Date modified	Type	Size
KMMIUSB	6/24/2015 5:00 PM	Setup Information	

File name: KMMIUSB.INF

Files of type: Setup Information (*.inf)

Click [OK].



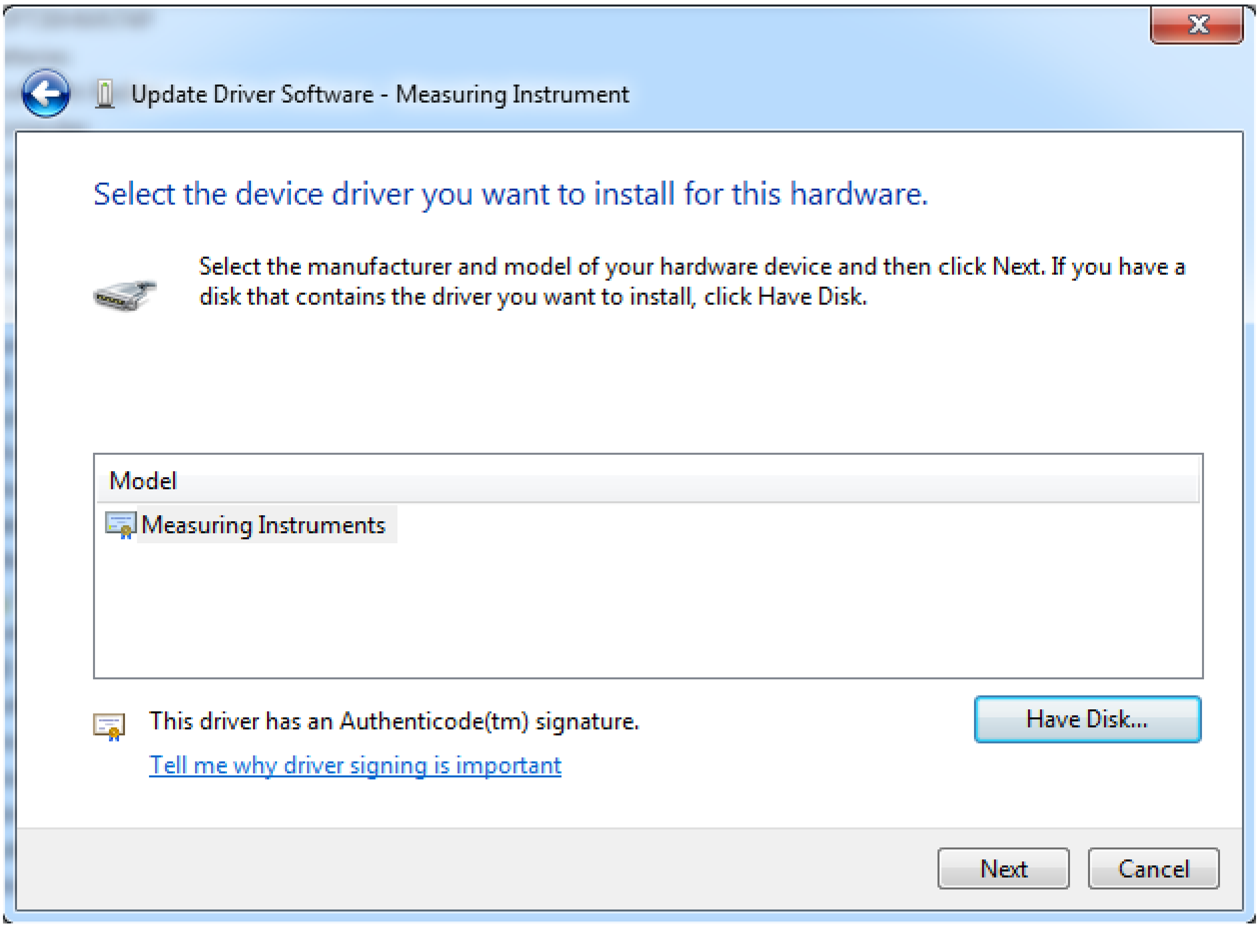
Install From Disk

Insert the manufacturer's installation disk, and then make sure that the correct drive is selected below.

Copy manufacturer's files from:

C:¥CM-MISDK_win100_all¥driver

Click [Next].



Update Driver Software - Measuring Instrument

Select the device driver you want to install for this hardware.

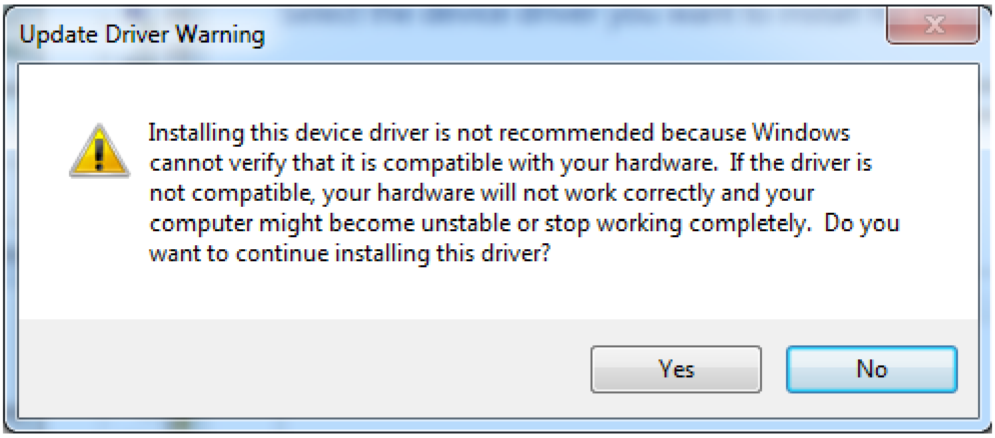
Select the manufacturer and model of your hardware device and then click Next. If you have a disk that contains the driver you want to install, click Have Disk.

Model
Measuring Instruments

This driver has an Authenticode(tm) signature.

Tell me why driver signing is important

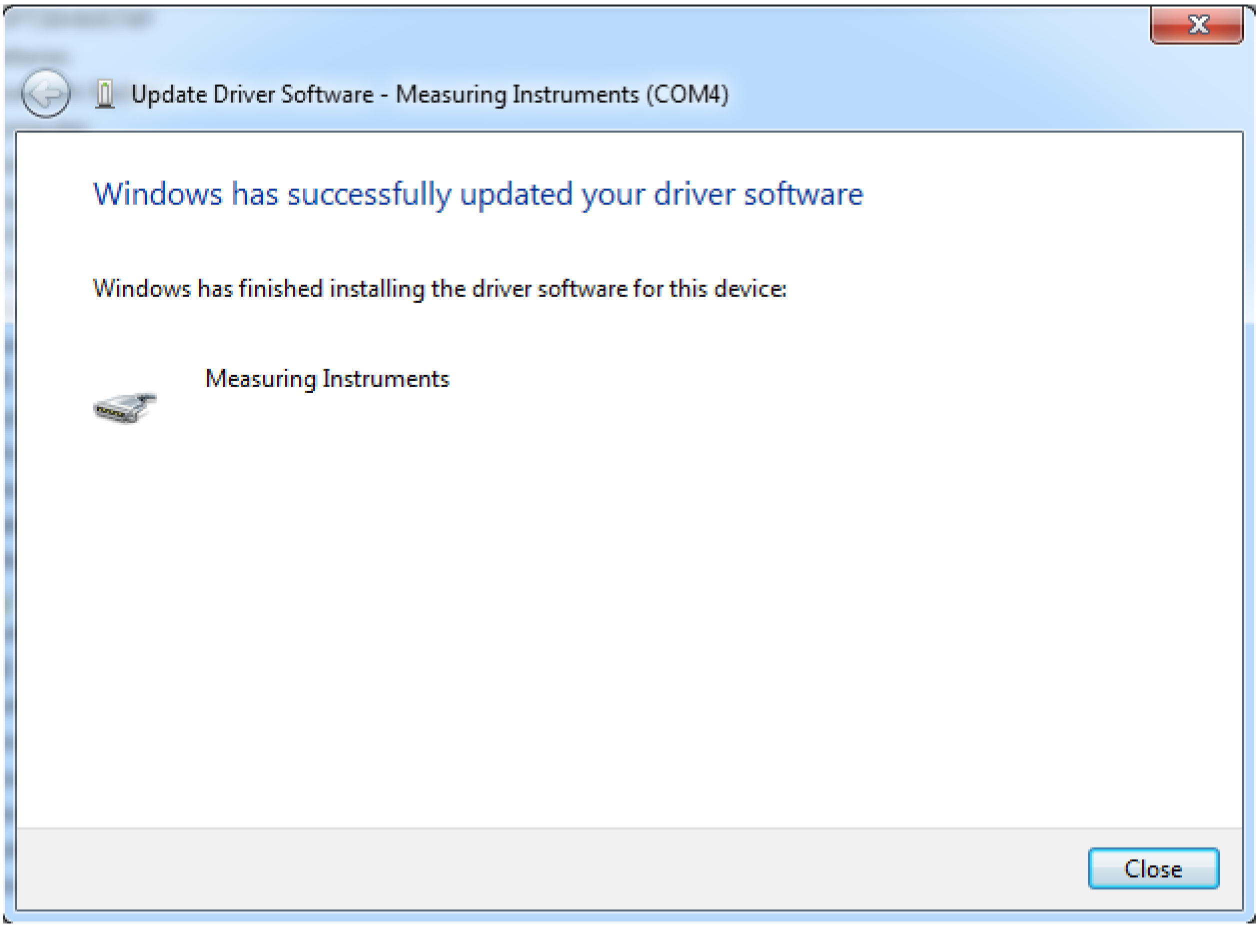
Click [Yes].



Update Driver Warning

Installing this device driver is not recommended because Windows cannot verify that it is compatible with your hardware. If the driver is not compatible, your hardware will not work correctly and your computer might become unstable or stop working completely. Do you want to continue installing this driver?

Click [Close] when you have confirmed that the installation has finished.



Update Driver Software - Measuring Instruments (COM4)

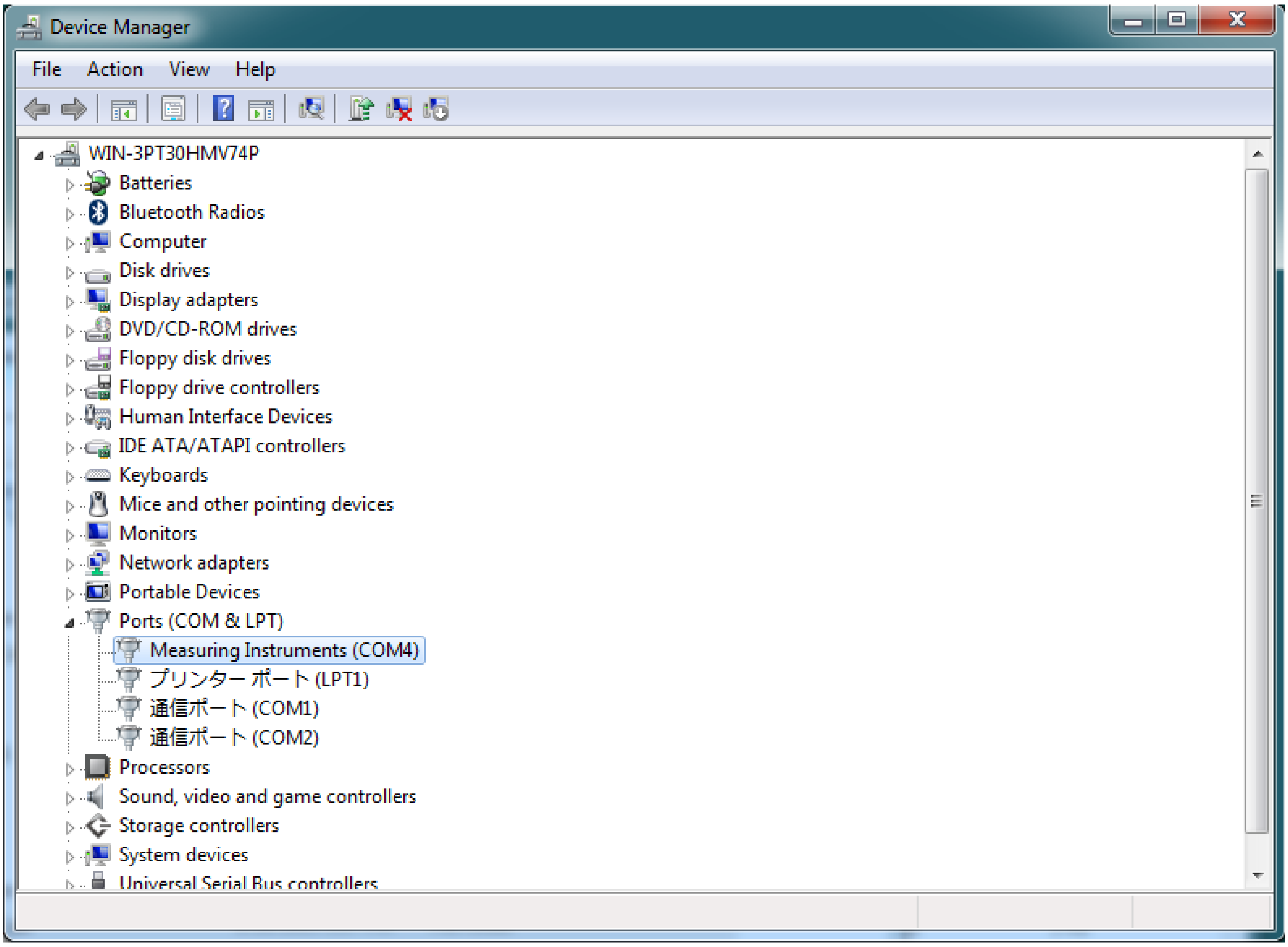
Windows has successfully updated your driver software

Windows has finished installing the driver software for this device:

Measuring Instruments

The warning symbol on [Measuring Instruments] should no longer be displayed. Next, confirm that the COM number is displayed, and then close the window by clicking x at the top right.

(COM11 is shown in the following screenshot, but the actual COM number will depend on the PC.)



This concludes installation of the driver.

Appendix C. List of parameters settable by instrument and version

Parameters that can be set on the instrument will depend on the instrument itself and its version. Refer to the following lists.

Warning status

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
WR1	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	×	×	×
WR2	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
WR3	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
WR4	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
WR5	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
WR6	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
WR7	V1.0～	V1.0～	×	×	×	×	V1.0～	×	×
WR8	V1.0～	V1.0～	×	×	×	×	V1.0～	×	×
WR9	×	×	×	×	×	×	V1.0～	V1.0～	×
WR10	×	×	×	×	×	×	V1.0～	V1.0～	×
WR11	×	×	×	×	×	×	V1.0～	V1.0～	×
WR12	×	×	×	×	×	×	V1.0～	V1.0～	×
WR13	×	×	×	×	×	×	V1.0～	V1.0～	×
WR14	×	×	×	×	×	×	V1.0～	V1.0～	×

Calibration status

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
StatusZero	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
StatusWhite	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
StatusGloss	V1.0～	V1.0～	×	×	×	×	V1.0～	×	×
StatusMeasure	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
StatusMeasureWrn	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～
StatusUser	V1.0～	V1.0～	V1.0～	V1.0～	V1.0～	×	V1.0～	V1.0～	V1.0～

Fluorescence adjustment conditions

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
UVADJ_PROFILE		V1.1～	V1.1～				V1.1～		V1.1～
UVADJ_WI		V1.1～	V1.1～				V1.1～		V1.1～
UVADJ_TINT		×	×				V1.1～		V1.1～
UVADJ_WITINT		V1.1～	V1.1～				V1.1～		V1.1～
UVADJ_BRIGHTNESS		V1.1～	V1.1～				V1.1～		V1.1～
UVADJ_GG		V1.1～	V1.1～				V1.1～		V1.1～

Fluorescence coefficient data type

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
UVADJ_DATATYPE_SCI		V1.1～	V1.1～				V1.0～		×
UVADJ_DATATYPE_SCE		V1.1～	V1.1～				V1.0～		×
UVADJ_DATATYPE_NONE		×	×				×		V1.0～

Measurement type

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
MEASTYPE_REF							V1.0～		
MEASTYPE_TRA							V1.0～		

Measurement area

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
AREA_MAV	V1.0～	V1.0～	V1.0～				V1.0～	V1.0～	V1.0～

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
AREA_SAV	V1.0~	V1.0~	V1.0~	▨	▨	▨	V1.0~	V1.0~	×
AREA_LAV	×	×	×	▨	▨	▨	V1.0~	V1.0~	V1.0~
AREA_LMAV	×	×	×	▨	▨	▨	V1.0~	×	×
AREA_AUTO	×	×	×	▨	▨	▨	V1.0~	V1.0~	×

Irradiation direction

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
LDIRECTION_NONE	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
LDIRECTION_L	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
LDIRECTION_R	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
LDIRECTION_DP	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨

Measurement angle

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
MEAS_ANGLE_M15	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
MEAS_ANGLE_15	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
MEAS_ANGLE_25	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
MEAS_ANGLE_45	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
MEAS_ANGLE_75	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
MEAS_ANGLE_110	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨

Tilt detection

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
OFF	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨
ON	▨	▨	▨	▨	▨	V1.0~	▨	▨	▨

Measurement mode

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
MeasModeColorAndGloss	V1.0~	V1.0~	V1.0~	×	×	▨	V1.0~	▨	▨
MeasModeColorOnly	V1.0~	V1.0~	V1.0~	V1.0	V1.0~	▨	V1.0~	▨	▨
MeasModeGlossOnly	V1.0~	V1.0~	V1.0~	×	×	▨	×	▨	▨
MeasModeOpacity	×	V1.0~	V1.0~	V1.0	V1.0~	▨	×	▨	▨

Specular component

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
SC_SCI	▨	V1.0~	V1.0~	V1.0~	V1.0~	▨	V1.0~	V1.0~	▨
SC_SCE	▨	V1.0~	V1.0~	V1.0~	V1.0~	▨	V1.0~	V1.0~	▨
SC_SCIE	▨	V1.0~	V1.0~	V1.0~	V1.0~	▨	V1.0~	V1.0~	▨

UV condition

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
UV_100	▨	V1.0~	V1.0~	▨	▨	▨	V1.0~	V1.0~	V1.0~
UV_CUT400	▨	V1.0~	V1.0~	▨	▨	▨	×	×	×

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
UV_CUT400N	▨	V1.1~(*)	V1.1~(*)	▨	▨	▨	×	×	×
UV_100_CUT400	▨	×	×	▨	▨	▨	V1.0~	V1.0~	V1.0~
UV_100_CUT420	▨	×	×	▨	▨	▨	V1.0~	V1.0~	V1.0~
UV_CUT400LOW	▨	×	×	▨	▨	▨	V1.0~	V1.0~	V1.0~
UV_CUT420LOW	▨	×	×	▨	▨	▨	V1.0~	V1.0~	V1.0~
UV_100_CUT400N	▨	V1.1~	V1.1~	▨	▨	▨	V1.0~	V1.0~	V1.0~
UV_100_CUT400L	▨	×	×	▨	▨	▨	V1.0~	V1.0~	V1.0~
UV_100_CUT420N	▨	×	×	▨	▨	▨	V1.0~	V1.0~	V1.0~
UV_100_CUT420L	▨	×	×	▨	▨	▨	V1.0~	V1.0~	V1.0~

(*): * Used for data properties and jobs.

Auto average count

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
Minimum	1	1	1	1	1	1	1	1	1
Maximum	10	10	10	10	10	10	10	10	10

Manual average count

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
Minimum	1	1	1	1	1	1	▨	▨	▨
Maximum	30	30	30	30	30	10	▨	▨	▨

Manual averaging save mode

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
SAVEMODE_AUTO	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	▨	▨	▨
SAVEMODE_MANUAL	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	▨	▨	▨

SMC setting

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
OFF	▨	V1.0~	V1.0~	V1.0~	▨	▨	▨	▨	▨
ON	▨	V1.0~	V1.0~	V1.0~	▨	▨	▨	▨	▨

SMC number of times

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
Minimum	▨	3	3	3	▨	▨	▨	▨	▨
Maximum	▨	10	10	10	▨	▨	▨	▨	▨

Display type

	25cG	26dG	26d	25d	23d	M6
DISPTYPE_ABS	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
DISPTYPE_DIF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
DISPTYPE_ABSDIF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×
DISPTYPE_CUSTOM	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×
DISPTYPE_GRAPH_ABS	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
DISPTYPE_GRAPH_DIF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

	25cG	26dG	26d	25d	23d	M6
DISPTYPE_GRAPH_REF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×
DISPTYPE_PASS_FAIL	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
DISPTYPE_MI	×	×	×	×	×	V1.0~
DISPTYPE_GRAPH_LINE	×	×	×	×	×	V1.0~
DISPTYPE_AUDI2000_EC	×	×	×	×	×	V1.1~
DISPTYPE_AUDI2000_EP	×	×	×	×	×	V1.1~

Observer

	25cG	26dG	26d	25d	23d	M6
Deg02	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
Deg10	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Illuminant

	25cG	26dG	26d	25d	23d	M6
ILL_None	V1.0~ (*)	V1.0~ (*)	V1.0~ (*)	V1.0~ (*)	V1.0~ (*)	V1.0~ (*)
ILL_A	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_C	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_D50	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_D65	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_ID50	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×
ILL_ID65	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×
ILL_F2	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_F6	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_F7	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_F8	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_F10	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_F11	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_F12	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ILL_USER1	V1.0~	V1.0~	V1.0~	V1.0~	×	V1.0~

(*) Only the second illuminant can be set.

Color space

	25cG	26dG	26d	25d	23d	M6
COLOR_LAB	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
COLOR_LCH	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
COLOR_HLAB	V1.0~	V1.0~	V1.0~	V1.0~	×	×
COLOR_YXY	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×
COLOR_XYZ	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×
COLOR_MUNSELL_C	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	×

Color equation

	25cG	26dG	26d	25d	23d	M6
EQUATION_DE1976	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
EQUATION_CMC	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

	25cG	26dG	26d	25d	23d	M6
EQUATION_DE1994	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
EQUATION_DE2000	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
EQUATION_DEH	V1.0~	V1.0~	V1.0~	V1.0~	×	×
EQUATION_DEP	×	×	×	×	×	V1.0~
EQUATION_DEC	×	×	×	×	×	V1.0~
EQUATION_DE99o	V1.1~	V1.0~	V1.0~	V1.0~	×	V1.1~

Custom items

	25cG	26dG	26d	25d	23d	M6
None	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
L*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
a*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
b*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
C*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
h	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
L(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	
a(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	
b(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	
X	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
Y	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
Z	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
x	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
y	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
H	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
V	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
C	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
WI(E313-73)	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
WI(CIE)	V1.0~	V1.0~	V1.0~	V1.0~	×	
Tint(CIE)	V1.0~	V1.0~	V1.0~	V1.0~	×	
YI(E313)	V1.0~	V1.0~	V1.0~	V1.0~	×	
YI(D1925)	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
B(ISO)	V1.0~	V1.0~	V1.0~	V1.0~	×	
GU	V1.0~	V1.0~	×	×	×	
UserE1	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
UserC1	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
UserE2	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
UserC2	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
UserE3	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
UserC3	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
8° gloss	×	×	V1.0~	V1.0~	×	
WI(Ganz)	×	V1.1~	V1.1~	×	×	
Tint(Ganz)	×	V1.1~	V1.1~	×	×	
ΔL*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
Δa*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
Δb*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	

	25cG	26dG	26d	25d	23d	M6
ΔC*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔH*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔL(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	
Δa(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	
Δb(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	
ΔX	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔY	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔZ	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
Δx	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
Δy	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔWI(E313-73)	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔWI(CIE)	V1.0~	V1.0~	V1.0~	V1.0~	×	
ΔTint(CIE)	V1.0~	V1.0~	V1.0~	V1.0~	×	
ΔYI(E313)	V1.0~	V1.0~	V1.0~	V1.0~	×	
ΔYI(D1925)	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔB(ISO)	V1.0~	V1.0~	V1.0~	V1.0~	×	
ΔGU	V1.0~	V1.0~	×	×	×	
MI	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔE*ab	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
CMC	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔE*94	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔE00	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
ΔE(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	
ΔE99o	V1.1~	V1.0~	V1.0~	V1.0~	×	
StrengthXYZ	×	V1.0~	V1.0~	V1.0~	×	
StrengthX	×	V1.0~	V1.0~	V1.0~	×	
StrengthY	×	V1.0~	V1.0~	V1.0~	×	
StrengthZ	×	V1.0~	V1.0~	V1.0~	×	
GreyScale	×	V1.0~	V1.0~	V1.0~	×	
ΔWI(Ganz)	×	V1.1~	V1.1~	×	×	
ΔTint(Ganz)	×	V1.1~	V1.1~	×	×	

Irradiation direction to display

	25cG	26dG	26d	25d	23d	M6
DIRECTION_DP						V1.1~
DIRECTION_L						V1.1~

Target filter

	25cG	26dG	26d	25d	23d	M6
FILTER_OFF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
FILTER_SAVE	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	
FILTER_GROUP	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	

Target protection

	25cG	26dG	26d	25d	23d	M6
OFF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ON	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Tolerance ID

	25cG	26dG	26d	25d	23d	M6
ΔL*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
Δa*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
Δb*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
ΔC*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
ΔH*	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
ΔL(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	×
Δa(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	×
Δb(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	×
ΔX	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	×
ΔY	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	×
ΔZ	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	×
Δx	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	×
Δy	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	×
ΔWI(E313-73)	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	×
ΔWI(CIE)	V1.0~	V1.0~	V1.0~	V1.0~	×	×
ΔTint(CIE)	V1.0~	V1.0~	V1.0~	V1.0~	×	×
ΔYI(E313)	V1.0~	V1.0~	V1.0~	V1.0~	×	×
ΔYI(D1925)	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	×
ΔB(ISO)	V1.0~	V1.0~	V1.0~	V1.0~	×	×
ΔGU	V1.0~	V1.0~	×	×	×	×
MI	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
ΔE*ab	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
CMC	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
ΔE*94	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
ΔE00	V1.0~	V1.0~	V1.0~	V1.0~	V1.0	V1.0~
ΔE(Hunter)	V1.0~	V1.0~	V1.0~	V1.0~	×	×
ΔEp(DIN6175)	×	×	×	×	×	V1.0~
ΔEc(DIN6175)	×	×	×	×	×	V1.0~
ΔFF	×	×	×	×	×	V1.0~
ΔE99o	V1.1~	V1.0~	V1.0~	V1.0~	×	V1.1~
ΔEc(Audi2000)	×	×	×	×	×	V1.1~
ΔEc average (Audi2000)	×	×	×	×	×	V1.1~
ΔEc maximum (Audi2000)	×	×	×	×	×	V1.1~
ΔEp(Audi2000)	×	×	×	×	×	V1.1~
ΔEp average (Audi2000)	×	×	×	×	×	V1.1~
ΔEp maximum (Audi2000)	×	×	×	×	×	V1.1~
ΔStrengthXYZ	×	V1.0~	V1.0~	V1.0~	×	×
ΔstrengthX	×	V1.0~	V1.0~	V1.0~	×	×

	25cG	26dG	26d	25d	23d	M6
ΔstrengthY	×	V1.0~	V1.0~	V1.0~	×	×
ΔstrengthZ	×	V1.0~	V1.0~	V1.0~	×	×
Opacity difference	×	V1.0~	V1.0~	V1.0~	×	×
Grayscale difference	×	V1.0~	V1.0~	V1.0~	×	×
ΔWI(Ganz)	×	V1.1~	V1.1~	×	×	×
ΔTint(Ganz)	×	V1.1~	V1.1~	×	×	×

Warning level

	25cG	26dG	26d	25d	23d	M6
Minimum	0	0	0	0	0	0
Maximum	100	100	100	100	100	100

Instrument mode

	25cG	26dG	26d	25d	23d	M6
INSTRUMENTMODE_NORMAL	▨	V1.0~	V1.0~	V1.0~	V1.0~	▨
INSTRUMENTMODE_SIMPLE	▨	V1.0~	V1.0~	V1.0~	V1.0~	▨

User type

	25cG	26dG	26d	25d	23d	M6
USERTYPE_ADMIN	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
USERTYPE_WORKER	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Automatic printing

	25cG	26dG	26d	25d	23d	M6
OFF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ON	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Display brightness

	25cG	26dG	26d	25d	23d	M6
Minimum	0	0	0	0	0	0
Maximum	4	4	4	4	4	4

Display direction

	25cG	26dG	26d	25d	23d	M6
SCREENDIR_0	▨	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
SCREENDIR_180	▨	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Sound

	25cG	26dG	26d	25d	23d	M6
OFF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ON	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Calibration interval

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
Minimum	1	1	1	1	1	1	1	1	1
Maximum	24	24	24	24	24	24	24	24	24

User calibration

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
OFF	V1.0~	V1.0~	V1.0~	V1.0~	▨	▨	V1.0~	V1.0~	V1.0~
ON	V1.0~	V1.0~	V1.0~	V1.0~	▨	▨	V1.0~	V1.0~	V1.0~

Periodical calibration notification

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
OFF	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
ON	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Skip zero calibration on/off

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
OFF	▨	V1.0~	V1.0~	V1.0~	V1.0~	▨	▨	▨	▨
ON	▨	V1.0~	V1.0~	V1.0~	V1.0~	▨	▨	▨	▨

Date format

	25cG	26dG	26d	25d	23d	M6
DF_YYYYMMDD	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
DF_MMDDYYYY	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
DF_DDMMYYYY	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Language

	25cG	26dG	26d	25d	23d	M6
LANGUAGE_ENGLISH	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_JAPANESE	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_GERMAN	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_FRENCH	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_SPANISH	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_ITALIAN	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_CHINESE_S	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_PORTUGUESE	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_RUSSIAN	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_POLISH	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~
LANGUAGE_TURKISH	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~	V1.0~

Power savings

	25cG	26dG	26d	25d	23d	M6
	V1.1~	V1.1~	V1.0~	V1.0~	V1.0~	V1.1~
Minimum	0	0	0	0	0	0
Maximum	60	60	60	60	60	60

Job

	25cG	26dG	26d	25d	23d	M6
Job name (size)		20	20	20		
Data name (size)		30	30	30		
Comment (size)		100	100	100		

Date/time type

	25cG	26dG	26d	25d	23d	M6	36dG	36d	3630A
DATETYPE_COLOR	V1.0~	V1.0~					V1.0~		
DATETYPE_GLOSS	V1.0~	V1.0~					V1.0~		

KONICA MINOLTA