

CM-MISDK(C#版)

リファレンスマニュアル

[Rev.1.02]



KONICA MINOLTA

CM-MISDK(C#版) リファレンスマニュアル**●本書で使用しているアプリケーション名などの正式名称**

本文中の表記	正式名称
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

●商標について

Microsoft、Windows は、米国 Microsoft Corporation の米国およびその他の国における登録商標です。
その他、本書に記載の会社名、商品名は各社の登録商標または商標です。

●本書に関するご注意

本書の内容の一部または全部を無断で転載することは禁止されています。

本書の内容については、将来予告なしに変更することがあります。

本書の内容を運用した結果につきましては、上記にかかわらず責任を負いかねますので、あらかじめご了承ください。

CM-MISDK(C#版) リファレンスマニュアル

目 次

はじめに	12
1. システム環境	12
2. インストール・アンインストール方法	13
2.1 インストール	13
2.2 アンインストール	13
3. SDK 概要	14
3.1 機能一覧	14
3.2 基本的なフロー	19
3.2.1 測定	19
3.2.2 基準値書き込み(新規の場合)	21
3.2.3 デフォルト限界値設定	21
3.2.4 ジョブ設定	22
3.2.5 蛍光調整	23
3.3 SDK によるプログラム作成方法	25
3.3.1 開発環境からの使用方法	25
3.3.2 サンプルコード概要	25
4. SDK リファレンス	26
4.1 API のフォーマット	26
4.1.1 フォーマット	26
4.1.2 戻り値	27
4.2 接続・切断	28
Connect : 指定した仮想 COM ポートに繋がっている測色計に接続します	28
Disconnect : 指定した仮想 COM ポートに接続している測色計との通信を切断します	29
GetDeviceList : PC に繋がっている測色計のリストを取得します	30
GetInstrumentInfo : 機器情報を取得します	31
GetSDKVersion : SDK バージョンを取得します	32
4.3 校正・測定	33
GetCalibrationStatus : 校正状態を取得します	33
PerformZeroCalibration : ゼロ校正を実行します	34
PerformWhiteCalibration : 白色校正を実行します	35
PerformGlossCalibration : 光沢校正を実行します	36
PerformUserCalibration : ユーザー校正を実行します	37
PerformMeasurement : 測定を実行します	38
PollingMeasurement : 測定状態を取得します	39
CancelMeasurement : 測定を中止します	40

CM-MISDK(C#版) リファレンスマニュアル

ReadLatestData : 最新の測定データを取得します	41
ReadAllLatestData : 最新の測定データをすべて取得します	42
SetWhiteCalibrationData : 白色校正データを設定します	44
GetWhiteCalibrationData : 白色校正データを取得します	45
SetGlossCalibrationData : 光沢校正データを設定します	46
GetGlossCalibrationData : 光沢校正データを取得します	47
SetUserCalibrationData : ユーザー校正データを設定します	48
GetUserCalibrationData : ユーザー校正データを取得します	49
SetUserCalibrationEnable : ユーザー校正使用可否を設定します	50
GetUserCalibrationEnable : ユーザー校正使用可否を取得します	51
SetTriggerMode : 本体キー測定を設定します	52
GetTriggerMode : 本体キー測定を取得します	53
DeleteTriggerData : 本体キー測定データを削除します	54
IsThereTriggerData : 本体キー測定データの有無を取得します	55
GetZeroCalibrationDate : ゼロ校正日時を取得します	56
GetWhiteCalibrationDate : 白色校正日時を取得します	57
GetGlossCalibrationDate : 光沢校正日時を取得します	58
GetUserCalibrationDate : ユーザー校正日時を取得します	59
ClearUvAdjustInfo : 蛍光調整用の各種データをクリアします	60
SetProfileForUvAdjust : 蛍光調整用のプロファイルデータを設定します	61
GetProfileForUvAdjust : 蛍光調整用のプロファイルデータを取得します	62
SetWiForUvAdjust : 蛍光調整用の WI を設定します	63
GetWiForUvAdjust : 蛍光調整用の WI を取得します	64
SetTintForUvAdjust : 蛍光調整用の Tint を設定します	65
GetTintForUvAdjust : 蛍光調整用の Tint を取得します	66
SetIsoBrightnessForUvAdjust : 蛍光調整用の ISO ブライツネスを設定します	67
GetIsoBrightnessForUvAdjust : 蛍光調整用の ISO ブライツネスを取得します	68
SetGanzForUvAdjust : 蛍光調整用の Ganz & Griesser を設定します	69
GetGanzForUvAdjust : 蛍光調整用の Ganz & Griesser を取得します	70
SetDataForUvAdjust : 蛍光調整用のデータを設定します	71
GetDataForUvAdjust : 蛍光調整用のデータを取得します	72
PerformUvAdjust : 蛍光調整を実行し、係数を機器に設定します	73
PerformUvAdjustUsingData : 蛍光調整を実行し、係数を機器に設定します	74
ClearCoefForUvAdjust : 機器内の蛍光調整係数をクリアします	75
SetCoefForUvAdjust : 蛍光係数を設定します	76
GetCoefForUvAdjust : 蛍光係数を取得します	77

CM-MISDK(C#版) リファレンスマニュアル

4.4 測定条件	78
SetMeasurementArea : 測定径を設定します	78
GetMeasurementArea : 測定径を取得します	79
SetMeasurementType : 測定タイプを設定します	80
GetMeasurementType : 測定タイプを取得します	81
SetMeasurementAngle : 測定角度を設定します	82
GetMeasurementAngle : 測定角度を取得します	83
SetTiltDetection : 傾き検知を設定します	84
GetTiltDetection : 傾き検知の有効・無効を取得します	85
SetMeasurementMode : 測定モードを設定します	86
GetMeasurementMode : 測定モードを取得します	87
SetSpecularComponent : 正反射光処理を設定します	88
GetSpecularComponent : 正反射光処理を取得します	89
SetUv : UV 条件を設定します	90
GetUv : UV 条件を取得します	91
SetAutoAverageTimes : 自動平均回数を設定します	92
GetAutoAverageTimes : 自動平均回数を取得します	93
SetManualAverageTimes : 手動平均回数を設定します	94
GetManualAverageTimes : 手動平均回数を取得します	95
SetManualAverageSaveMode : 手動平均方法を設定します	96
GetManualAverageSaveMode : 手動平均保存方法を取得します	97
SetCondSMC : SMC 条件を設定します	98
GetCondSMC : SMC 条件を取得します	99
4.5 表示条件	100
SetDisplayType : 表示形式を設定します	100
GetDisplayType : 表示形式を取得します	101
SetObserverAndIlluminant : 観察視野・光源を設定します	102
GetObserverAndIlluminant : 観察視野・光源を取得します	103
SetUserIlluminant : ユーザー光源を設定します	104
GetUserIlluminant : ユーザー光源を取得します	105
SetColorSpace : 表色系を設定します	106
GetColorSpace : 表色系を取得します	107
SetEquation : 色差式を設定します	108
GetEquation : 色差式を取得します	109
SetCustomItem : カスタム表示項目を設定します	110
GetCustomItem : カスタム表示項目を取得します	111

CM-MISDK(C#版) リファレンスマニュアル

SetDisplayData : 表示するデータの照射方向を設定します	112
GetDisplayData : 表示するデータの照射方向を取得します	113
SetUserEquation : ユーザーインデックスを設定します	114
GetUserEquation : ユーザーインデックスを取得します	117
4.6 データ	118
SetActiveTarget : アクティブ基準色番号を設定します	118
GetActiveTarget : アクティブ基準色番号を取得します	119
GetTargetNumberList : 保存されている基準色番号のリストを取得します	120
GetTargetNumberList2 : 表示フィルター適用時の基準色番号のリストを取得します	121
DeleteTargetData : 基準色を削除します	122
SetTargetData : 基準色データを設定します	123
GetTargetData : 基準色データを取得します	124
GetAllTargetData : 基準色データを取得します	126
SetToleranceForTarget : 基準色の限界値を設定します	127
GetToleranceForTarget : 基準色の限界値を取得します	128
SetParametricForTarget : 基準色のパラメトリック係数を設定します	129
GetParametricForTarget : 基準色のパラメトリック係数を取得します	130
SetTargetFilter : 基準色フィルター条件を設定します	131
GetTargetFilter : 基準色フィルター条件を取得します	132
SetTargetProtect : 基準色保護を設定します	133
GetTargetProtect : 基準色保護を取得します	134
GetSampleCount : 測定値保存数を取得します	135
DeleteSampleData : 測定値を削除します	136
GetSampleData : 測定値データを取得します	137
GetAllSampleData : 測定値データを取得します	138
4.7 その他	139
SetActiveGroup : アクティブグループ番号を設定します	139
GetActiveGroup : アクティブグループ番号を取得します	140
SetGroupName : グループ名を設定します	141
GetGroupName : グループ名を取得します	142
SetMultipleGroupName : グループ名を複数まとめて設定します	143
GetMultipleGroupName : グループ名を複数まとめて取得します	144
SetTolerance : デフォルト限界値を設定します	145
GetTolerance : デフォルト限界値を取得します	146
SetParametric : デフォルトパラメトリック係数を設定します	147
GetParametric : デフォルトパラメトリック係数を取得します	148

CM-MISDK(C#版) リファレンスマニュアル

SetWarningLevel : 警告レベルを設定します	149
GetWarningLevel : 警告レベルを取得します	150
SetInstrumentMode : 機器モードを設定します	151
GetInstrumentMode : 機器モードを取得します	152
SetUserType : ユーザータイプを設定します	153
GetUserType : ユーザータイプを取得します	154
SetAdminPassword : 管理者パスワードを設定します	155
GetAdminPassword : 管理者パスワードを取得します	156
SetAutoPrint : 自動印刷を設定します	157
GetAutoPrint : 自動印刷を取得します	158
SetBrightness : 画面の明るさを設定します	159
GetBrightness : 画面の明るさを取得します	160
SetScreenDirection : 画面の表示向きを設定します	161
GetScreenDirection : 画面の表示向きを取得します	162
SetSound : ビープ音を設定します	163
GetSound : ビープ音を取得します	164
SetCalibrationInterval : 校正間隔を設定します	165
GetCalibrationInterval : 校正間隔を取得します	166
SetAnnualCalibration : 定期校正喚起を設定します	167
GetAnnualCalibration : 定期校正喚起を取得します	168
SetZeroCalibrationSkip : ゼロ校正スキップ可否を設定します	169
GetZeroCalibrationSkip : ゼロ校正スキップ可否を取得します	170
SetDateTime : 日時を設定します	171
SetDateFormat : 日付書式を設定します	172
GetDateFormat : 日付書式を取得します	173
SetLanguage : 表示言語を設定します	174
GetLanguage : 表示言語を取得します	175
SetAutoPowerOff : オートパワーオフが働く時間を設定します	176
GetAutoPowerOff : オートパワーオフが働く時間を取得します	177
ClearJobInfo : ジョブ情報をクリアします	178
SetJobInfo : ジョブ情報を設定します	179
GetJobInfo : ジョブ情報を取得します	180
GetJobStepType : ジョブのステップ種類を取得します	181
SetJobStepForOperation : ジョブの操作ステップを設定します	182
GetJobStepForOperation : ジョブの操作ステップを取得します	184
SetJobStepForResult : ジョブの結果ステップを設定します	185

CM-MISDK(C#版) リファレンスマニュアル

GetJobStepForResult : ジョブの結果ステップを取得します	186
SetJobImage : ジョブの画像を設定します	187
GetJobImage : ジョブの画像を取得します	188
ResetToFactorySetting : 工場出荷状態に戻します	189
5. 定義・構造体	190
5.1 型定義	190
5.2 クラス定義	191
Class DataForm(データフォームクラス)	191
Class SpecData(反射率クラス).....	191
Class MeasDataColor(測定データ色彩値クラス)	191
Class ColorCalData(測色用校正データクラス)	192
Class GlossCalData(光沢用校正データクラス)	192
Class UserCalData(ユーザー校正データクラス)	193
Class UserCalData2(ユーザー校正データクラス)	193
Class UvAdjustIndex(蛍光調整用のインデックスデータ)	194
Class UvAdjustCoef(蛍光調整係数).....	194
Class UvAdjustGG(Ganz & Griesser 蛍光調整データ)	195
Class MeasCondSMC(SMC 条件クラス)	195
Class ColorValue(各表色系の結果にアクセスするクラス)	196
Class TargetData(基準色データクラス).....	196
Class TargetDataPack(基準色データクラス)	197
Class ColorData(色彩値データクラス)	198
Class ToleranceData(限界値データクラス)	199
Class ToleranceParam(限界値データクラス)	199
Class ParamtricCoef(パラメトリック係数クラス)	200
Class SampleData(測定値データクラス)	200
Class SampleDataPack(測定値データクラス).....	201
Class InstrumentInfo(機器情報クラス).....	202
Class InstrumentInfoEx(機器情報クラス)	203
Class JobInfo(ジョブ情報)	203
Class JobStepOperation(ジョブ操作ステップ)	204
Class JobStepResult(ジョブ結果ステップ).....	205
Class JobImage(ジョブ画像)	206
5.3 値定義	207
CalStatus(校正状態).....	207
CalDataType(校正用データ種類).....	207

CM-MISDK(C#版) リファレンスマニュアル

MeasStatus(測定状態)	208
DataType(データ種類)	208
MeasDataType(測定データ種類)	209
IrradiationDirection (照射方向)	210
LightDirection (照射方向)	211
CondUvAdjust(蛍光調整条件)	211
UvAdjustDataType(蛍光係数用データ種類)	211
MeasType(測定タイプ)	211
MeasArea(測定径)	212
MeasAngle(測定角度)	212
MeasCondMode(測定モード)	212
MeasCondScie(正反射光処理)	213
MeasCondUv(UV 条件)	213
DataId(データ種類)	213
DataAttr(データ属性)	214
DataWarning(データ警告)	215
SaveMode(保存方法)	215
DisplayType(表示形式)	215
Observer(観察視野)	216
Illuminant(観察光源)	216
ColorSpace(表色系)	216
Equation(色差式)	217
CustomItem (カスタム項目)	217
DisplayData(表示データ)	219
FilterIndex(フィルター設定)	219
InstrumentMode(機器モード)	220
UserType(ユーザータイプ)	220
ScreenDirection(画面の表示向き)	220
DateFormat(日付書式)	220
Language(言語)	220
JobStepType(ジョブステップ種類)	221
OnOff(ON・OFF)	221
IsThereData(YES・NO)	221
DateType(日時種類)	221
ToleranceId(限界値 ID)	221
6. エラー/警告	224

CM-MISDK(C#版) リファレンスマニュアル

6.1 エラー一覧	224
6.2 警告一覧	226
付録 A. 使用できる文字コード	227
付録 B. デバイスドライバのインストール	228
自動インストール	228
手動インストール	228
付録 C. 機器・バージョンによる設定可能なパラメーター一覧	238
警告状態	238
校正状態	238
蛍光調整条件	239
蛍光係数用データ種類	239
測定径	240
測定タイプ	240
照射方向	241
測定角度	241
傾き検出	242
測定モード	242
正反射光処理	242
UV 条件	243
自動平均回数	244
手動平均回数	244
手動平均保存モード	244
SMC 設定	245
SMC 回数	245
表示形式	245
観察視野	246
観察光源	246
表色系	247
色差式	248
カスタム項目	249
表示する照射方向	252
基準色フィルター	252
基準色保護	253
限界値 ID	253
警告レベル	255
機器モード	256

CM-MISDK(C#版) リファレンスマニュアル

ユーザータイプ	256
自動印刷	256
画面明るさ	257
表示向き	257
ビープ音	257
校正喚起	258
ユーザー校正	258
定期校正連絡通知	258
ゼロ校正スキップ可否	259
日付書式	259
言語	259
省電力	260
ジョブ	260
日時種類	261

CM-MISDK(C#版) リファレンスマニュアル**はじめに**

本 SDK は物体色用機器の PC アプリケーションを開発するためのツールです。このマニュアルは本 SDK の使用方法を説明します。

アプリケーション開発者は、C#を使用することを想定し、プログラミングの方法はC#で説明します。また、本 SDK はマルチスレッドには対応していません。

1. システム環境

保証する環境は以下の通りです

対応 OS	Windows 7(x86)、Windows 7(x64) Windows 8.1(x86) 、Windows 8.1(x64) Windows 10(x86)、Windows 10(x64)
動作環境	.NET Framework 4.5
開発環境	Visual Studio 2012 Visual Studio 2013 Visual Studio 2015 Visual Studio 2017
開発言語	VC++、VC#、VB .NET
制御機種	・ CM-M6 ・ CM-25cG ・ CM-26dG ・ CM-26d ・ CM-25d ・ CM-23d

CM-MISDK(C#版) リファレンスマニュアル**2. インストール・アンインストール方法****2.1 インストール**

以下の手順に従い、本 SDK をインストールしてください。

- (1) 提供される「cm-misdk_verXXXrX.zip」を PC 上の任意の場所へ展開してください。
- (2) 展開が完了すると「cm-misdk_verXXXrX」が表示され、以下のフォルダが存在します。

番号	フォルダ名	概要
1	SDK	CM-MISDK 一式
2	Manual	CM-MISDK のリファレンスマニュアル
3	Driver	測色計の USB ドライバ
4	SampleCode	サンプルコード一式
5	License	使用許諾書

- (3) SDK を使用して開発するには、上記 SDK フォルダ内の下記ファイルを開発環境から参照できるように設定してください。詳細は、「3. SDK 概要」を参照してください。

番号	フォルダ名	概要
1	CMMISDK_x86.dll	SDK の dll ファイル(32bit 版)
2	CMMISDK_x64.dll	SDK の dll ファイル(64bit 版)
3	CMMISDK.NET.dll	C#版 dll ファイル

2.2 アンインストール

cm-misdk_verXXXrX フォルダと手作業でコピーしたフォルダ一式を削除してください。

CM-MISDK(C#版) リファレンスマニュアル**3. SDK 概要****3.1 機能一覧**

以下の処理を行うことができます。

接続・切断		
	Connect	機器と接続します
	Disconnect	機器との接続を解除します
	GetDeviceList	接続可能な機器のリストを取得します
	GetInstrumentInfo	機器情報を取得します
	GetSDKVersion	SDK バージョンを取得します
校正・測定		
	GetCalibrationStatus	校正状態を取得します
	PerformZeroCalibration	ゼロ校正を実行します
	PerformWhiteCalibration	白色校正を実行します
	PerformGlossCalibration	光沢校正を実行します
	PerformUserCalibration	ユーザー校正を実行します
	PerformMeasurement	測定を実行します
	PollingMeasurement	測定完了を判断します
	CancelMeasurement	測定をキャンセルします
	ReadLatestData	最新のデータを取得します
	ReadAllLatestData	最新のデータをすべて取得します
	SetWhiteCalibrationData	白色校正板データを設定します
	GetWhiteCalibrationData	白色校正板データを取得します
	SetGlossCalibrationData	光沢基準板データを設定します
	GetGlossCalibrationData	光沢基準板データを取得します
	SetUserCalibrationData	ユーザー校正板データを設定します
	GetUserCalibrationData	ユーザー校正板データを取得します
	SetUserCalibrationEnable	ユーザー校正使用可否を設定します。
	GetUserCalibrationEnable	ユーザー校正使用可否を取得します。
	SetTriggerMode	本体キー測定属性を設定します
	GetTriggerMode	本体キー測定属性を取得します
	DeleteTriggerData	本体キー測定データを削除します
	IsThereTriggerData	本体キー測定データ有無を取得します
	GetZeroCalibrationDate	ゼロ校正日時を取得します
	GetWhiteCalibrationDate	白色校正日時を取得します

CM-MISDK(C#版) リファレンスマニュアル

	GetGlossCalibrationDate	光沢校正日時を取得します
	GetUserCalibrationDate	ユーザー校正日時を取得します
	ClearUvAdjustInfo	蛍光調整用各種データをクリアします
	SetProfileForUvAdjust	蛍光調整用プロファイルを設定します
	GetProfileForUvAdjust	蛍光調整用プロファイルを取得します
	SetWiForUvAdjust	蛍光調整用 WI を設定します
	GetWiForUvAdjust	蛍光調整用 WI を取得します
	SetTintForUvAdjust	蛍光調整用 Tint を設定します
	GetTintForUvAdjust	蛍光調整用 Tint を取得します
	SetIsoBrightnessForUvAdjust	蛍光調整用 ISO ブライツネスを設定します
	GetIsoBrightnessForUvAdjust	蛍光調整用 ISO ブライツネスを取得します
	SetGanzForUvAdjust	蛍光調整用 Ganz & Griesser を設定します
	GetGanzForUvAdjust	蛍光調整用 Ganz & Griesser を取得します
	SetDataForUvAdjust	蛍光調整用データを設定します
	GetDataForUvAdjust	蛍光調整用データを取得します
	PerformUvAdjust	蛍光調整を実行し、係数を設定します
	PerformUvAdjustUsingData	蛍光調整を実行し、係数を設定します
	ClearCoefForUvAdjust	機器内の蛍光係数をクリアします
	SetCoefForUvAdjust	蛍光係数を設定します
	GetCoefForUvAdjust	蛍光係数を取得します
測定条件		
	SetMeasurementArea	測定径を設定します
	GetMeasurementArea	測定径を取得します
	SetMeasurementType	測定タイプを設定します
	GetMeasurementType	測定タイプを取得します
	SetMeasurementAngle	測定角度を設定します
	GetMeasurementAngle	測定角度を取得します
	SetTiltDetection	傾き検知を設定します
	GetTiltDetection	傾き検知を取得します
	SetMeasurementMode	測定モードを設定します
	GetMeasurementMode	測定モードを取得します
	SetSpecularComponent	正反射光処理を設定します
	GetSpecularComponent	正反射光処理を取得します
	SetUv	UV 条件を設定します
	GetUv	UV 条件を取得します
	SetAutoAverageTimes	自動平均回数を設定します

CM-MISDK(C#版) リファレンスマニュアル

	GetAutoAverageTimes	自動平均回数を取得します
	SetManualAverageTimes	手動平均回数を設定します
	GetManualAverageTimes	手動平均回数を取得します
	SetManualAverageSaveMode	手動平均保存方法を設定します
	GetManualAverageSaveMode	手動平均保存方法を取得します
	SetCondSMC	SMC 条件を設定します
	GetCondSMC	SMC 条件を取得します
表示条件		
	SetDisplayType	表示形式を設定します
	GetDisplayType	表示形式を取得します
	SetObserverAndIlluminant	観察視野・光源を設定します
	GetObserverAndIlluminant	観察視野・光源を取得します
	SetUserIlluminant	ユーザー光源データを設定します
	GetUserIlluminant	ユーザー光源データを取得します
	SetColorSpace	表色系を設定します
	GetColorSpace	表色系を取得します
	SetEquation	色差式を設定します
	GetEquation	色差式を取得します
	SetCustomItem	カスタム項目を設定します
	GetCustomItem	カスタム項目を取得します
	SetDisplayData	表示するデータの照射方向を設定します
	GetDisplayData	表示するデータの照射方向を取得します
	SetUserEquation	ユーザーインデックスを設定します
	GetUserEquation	ユーザーインデックスを取得します
データ		
	SetActiveTarget	アクティブ基準色を設定します
	GetActiveTarget	アクティブ基準色を取得します
	GetTargetNumberList	保存されている基準色番号のリストを取得します
	GetTargetNumberList2	表示フィルター適用時の基準色番号のリストを取得します
	DeleteTargetData	基準色データを削除します
	SetTargetData	基準色データを設定します
	GetTargetData	基準色データを取得します
	GetAllTargetData	基準色データを取得します
	SetToleranceForTarget	基準色の限界値データを設定します
	GetToleranceForTarget	基準色の限界値データを取得します

CM-MISDK(C#版) リファレンスマニュアル

	SetParametricForTarget	基準色のパラメトリック係数を設定します
	GetParametricForTarget	基準色のパラメトリック係数を取得します
	SetTargetFilter	基準色フィルター条件を設定します
	GetTargetFilter	基準色フィルター条件を取得します
	SetTargetProtect	基準色保護を設定します
	GetTargetProtect	基準色保護を取得します
	GetSampleCount	測定値保存数を取得します
	DeleteSampleData	測定値データを削除します
	GetSampleData	測定値データを取得します
	GetAllSampleData	測定値データを取得します
その他		
	SetActiveGroup	アクティブグループを設定します
	GetActiveGroup	アクティブグループを取得します
	SetGroupName	グループ名を設定します
	GetGroupName	グループ名を取得します
	SetMultipleGroupName	グループ名を複数まとめて設定します
	GetMultipleGroupName	グループ名を複数まとめて取得します
	SetTolerance	デフォルト限界値を設定します
	GetTolerance	デフォルト限界値を取得します
	SetParametric	デフォルトパラメトリック係数を設定します
	GetParametric	デフォルトパラメトリック係数を取得します
	SetWarningLevel	警告レベルを設定します
	GetWarningLevel	警告レベルを取得します
	SetInstrumentMode	機器モードを設定します
	GetInstrumentMode	機器モードを取得します
	SetUserType	ユーザータイプを設定します
	GetUserType	ユーザータイプを取得します
	SetAdminPassword	管理者パスワードを設定します
	GetAdminPassword	管理者パスワードを取得します
	SetAutoPrint	自動印刷を設定します
	GetAutoPrint	自動印刷を取得します
	SetBrightness	画面の明るさを設定します
	GetBrightness	画面の明るさを取得します
	SetScreenDirection	表示の向きを設定します
	GetScreenDirection	表示の向きを取得します
	SetSound	音量を設定します

CM-MISDK(C#版) リファレンスマニュアル

	GetSound	音量を取得します
	SetCalibrationInterval	校正間隔を設定します
	GetCalibrationInterval	校正間隔を取得します
	SetAnnualCalibration	定期校正喚起を設定します
	GetAnnualCalibration	定期校正喚起を取得します
	SetDateTime	日時を設定します
	SetDateFormat	日付書式を設定します
	GetDateFormat	日付書式を取得します
	SetLanguage	表示言語を設定します
	GetLanguage	表示言語を取得します
	SetAutoPowerOff	省電力モードを設定します
	GetAutoPowerOff	省電力モードを取得します
	ClearJobInfo	ジョブ情報をクリアします
	SetJobInfo	ジョブ情報を設定します
	GetJobInfo	ジョブ情報を取得します
	GetJobStepType	ジョブのステップ種類を取得します
	SetJobStepForOperation	ジョブの操作ステップを設定します
	GetJobStepForOperation	ジョブの操作ステップを取得します
	SetJobStepForResult	ジョブの結果ステップを設定します
	GetJobStepForResult	ジョブの結果ステップを取得します
	SetJobImage	ジョブの画像を設定します
	GetJobImage	ジョブの画像を取得します
	ResetToFactorySetting	工場出荷状態に戻します

CM-MISDK(C#版) リファレンスマニュアル3.2 基本的なフロー

3.2.1 測定

3.2.1.1 API を使用した測定

接続(4.2)

Connect

測定条件設定(4.4)

校正(4.3)

PerformZeroCalibration

PerformWhiteCalibration

PerformGlossCalibration

測定(4.3)

PerformMeasurement

PollingMeasurement

測定が完了するまで、ポーリング
して監視する

データ取得(4.3)

ReadAllLatestData

ReadLatestData

切断(4.2)

Disconnect

終了

3.2.1.2 本体キーを使用した測定

CM-MISDK(C#版) リファレンスマニュアル**接続(4.2)**

Connect

測定条件設定(4.4)**校正(4.3)**

PerformZeroCalibration

PerformWhiteCalibration

PerformGlossCalibration

測定(4.3)

SetTriggerMode

DeleteTriggerData

動作を繰り返す場合は、データ取得後にク
リアから行うこと

本体の測定キーを押す

IsThereTriggerData

データが準備できるまで、ポーリ
ングして監視する**データ取得(4.3)**

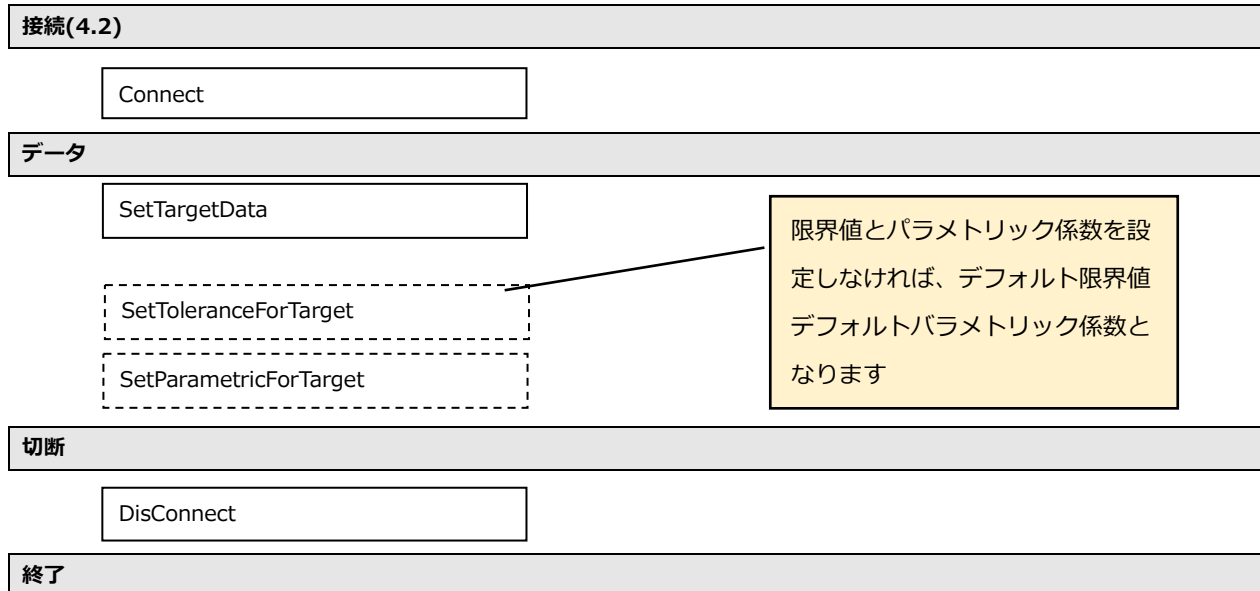
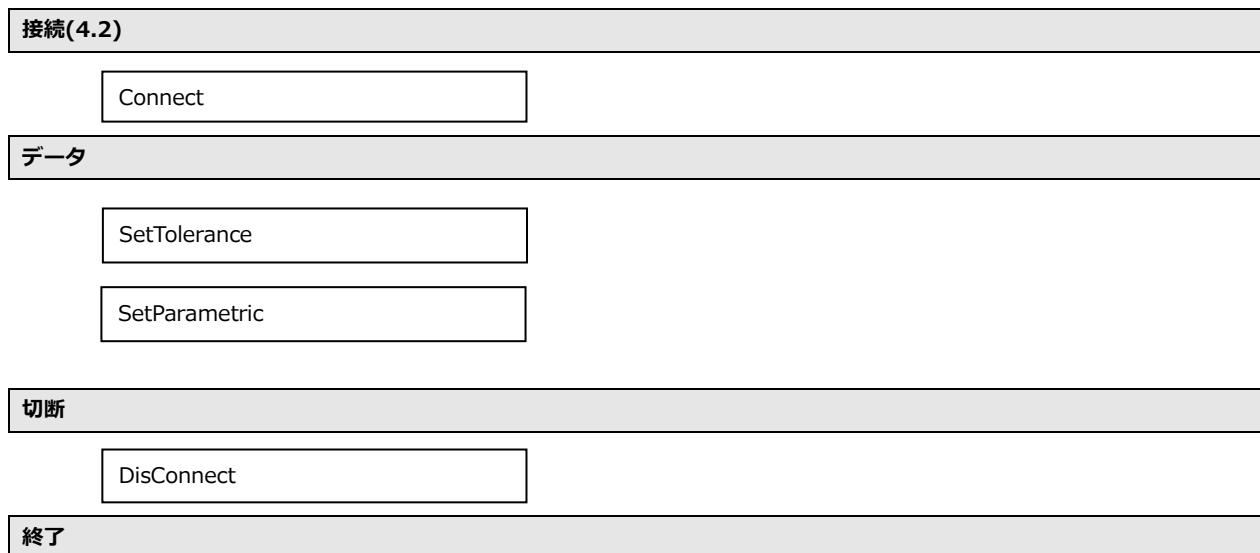
ReadAllLatestData

ReadLatestData

切断(4.2)

DisConnect

終了

CM-MISDK(C#版) リファレンスマニュアル**3.2.2 基準値書き込み(新規の場合)****3.2.3 デフォルト限界値設定**

CM-MISDK(C#版) リファレンスマニュアル**3.2.4 ジョブ設定****接続(4.2)**

Connect

データ

ClearJobInfo

最初からやり直す場合には、必ず
クリアしてください

SetJobInfo

ステップ数などジョブの基本情報
を設定します

SetJobStepForOperation

SetJobStepForResult

Info で設定したステップ数分の内
容を登録する

SetJobImage

必要な分だけ画像を登録する

切断

DisConnect

終了

CM-MISDK(C#版) リファレンスマニュアル

3.2.5 蛍光調整

接続(4.2)

Connect

校正(4.3)

PerformZeroCalibration

PerformWhiteCalibration

PerformGlossCalibration

蛍光調整事前準備

SetMeasurementType

SetMeasurementArea

SetMeasurementMode

SetSpecularComponent

SetUv

蛍光調整を実行するための測定条件を設定します

ClearUvAdjustInfo

SetProfileForUvAdjust

SetWiForUvAdjust

SetTintForUvAdjust

SetIsoBrightnessForUvAdjust

SetGanzForUvAdjust

使用する蛍光調整モードに応じて、必要な基準データを設定する

蛍光調整して機器へ書き込む

[Ganz & Griesser 以外の場合]

ClearCoefForUvAdjust

PerformUvAdjust

実行すると、現在の条件で測定を行い、蛍光係数を算出して機器に書き込む

CM-MISDK(C#版) リファレンスマニュアル

[Ganz & Griesser の場合]

SetDataForUvAdjust

蛍光調整に必要な測定データをすべて設定する

ClearCoefForUvAdjust

PerformUvAdjustUsingData

入力されたデータを使用して蛍光係数を算出して機器に書き込む
Ganz & Griesser 以外でも使用可能

[既に係数を保有している場合]

ClearCoefForUvAdjust

SetCoefForUvAdjust

使用する条件・蛍光調整モードに応じて、必要な係数を設定する

切断

Disconnect

終了

CM-MISDK(C#版) リファレンスマニュアル

3.3 SDK によるプログラム作成方法

3.3.1 開発環境からの使用方法

例として、Visual Studio 2013 での使用方法を記載します。

- (1) C++アプリケーションプロジェクト(以降、プロジェクトと記載)を作成します。
- (2) 「CMMISDK_***.dll」「CMMISDK.NET.dll」を以下のいずれかの場所に置いてください。
 - ・ 作成するアプリケーションの実行フォルダ
 - ・ 環境パスの通ったフォルダ
- (3) 「参照の追加」で「CMMISDK.NET.dll」を追加します。
- (4) API を使用したアプリケーションを作成してビルドします。

3.3.2 サンプルコード概要

サンプルコードでは、以下の 3 種類を準備しています・

- (1)校正・測定・測定データ取得(PerformMeasurement)
- (2)保存されている測定データの取得(ReadSampleData)
- (3)基準データの書き込み(WriteTargetData)

具体的な実装方法は、各サンプルコードを参照してください。

CM-MISDK(C#版) リファレンスマニュアル**4. SDK リファレンス****4.1 API のフォーマット****4.1.1 フォーマット**

各 API は以下のフォーマットで記載しています。

形式：

API の書式について説明しています。

引数：

API の引数について説明しています。

戻り値：

API を利用した際に返ってくる戻り値について説明しています。

戻り値の種別として以下の 3 種類があります。

種別	
正常	処理に成功した際に返ってきます
エラー	処理に失敗した際に返ってきます
警告	処理に成功したが、制約がある際に返ってきます

解説：

API を利用する際に必要な情報や注意事項などを説明しています。

CM-MISDK(C#版) リファレンスマニュアル

4.1.2 戻り値

すべての API は、以下の戻り値クラスを返します。

ReturnMessage

概要：

各 API の戻り値に用いるクラス

形式：

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

変数：

変数	説明
errorCode	エラーコード
errorMessage	エラーメッセージ ※エラーコードが正常の場合、警告が存在する場合はこのリストに格納されます

説明：

エラーコードとメッセージにより、エラーを特定します。

CM-MISDK(C#版) リファレンスマニュアル4.2 接続・切断

Connect : 指定した仮想 COM ポートに繋がっている測色計に接続します

形式 :

[ReturnMessage](#) Connect(Int32 comPort = 0)

引数 :

名称	I/O	説明
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	測定器との接続に失敗しました

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した仮想 COM ポートに繋がっている測色計に接続します。

comPort に 0 を指定(引数なしも同様)すると、自動で [GetDeviceList](#) を行い、一番小さい COM ポート番号に繋がっている測色計と接続を確立します。

以後、他の API で comPort に 0 を指定すると、Connect()した測色計の仮想 COM ポート番号で処理されます。

※Bluetooth 接続の場合は、comPort に 0 を指定できません。comPort を指定して接続する必要があります。

CM-MISDK(C#版) リファレンスマニュアル

Disconnect : 指定した仮想 COM ポートに接続している測色計との通信を切断します

形式 :

[ReturnMessage](#) Disconnect(Int32 comPort = 0)

引数 :

名称	I/O	説明
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

エラーの場合は、測色計の電源を OFF にして切断してください。

CM-MISDK(C#版) リファレンスマニュアル**GetDeviceList : PC に繋がっている測色計のリストを取得します****形式 :**[ReturnMessage](#) GetDeviceList(out Dictionary<Int32, string> deviceList, bool isAll=false)**引数 :**

名称	I/O	説明
deviceList	O	測色計リスト
		key COM ポート番号
		value 機種名(本体番号)
		例)COM3 に本体番号 1000001 の CM-25cG が接続されている場合
isAll	I	検索対象範囲
		false USB のみ検索する
		true USB・Bluetooth 両方を検索する

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	PC に繋がっている機器は存在しません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

PC に繋がっている測色計のリストを取得します。

isAll=true の場合は、使用可能な COM ポートに対して検索しますので、処理に時間が掛かります。

USB のみを使用する場合は、isAll=false で検索してください。

既に他のソフトで COM ポートをオープンしている測色計は、リストに存在しないので、ご注意ください。

CM-MISDK(C#版) リファレンスマニュアル**GetInstrumentInfo : 機器情報を取得します****形式 1 :**

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

引数 :

名称	I/O	説明
instrumentInfo	O	機器情報
comport	I	仮想 COM ポート番号

形式 2 :

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

引数 :

名称	I/O	説明
instrumentInfo	O	機器情報
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

機器情報を取得します。

形式 1 と形式 2 は、対応しているすべての機種で使用可能です。

CM-MISDK(C#版) リファレンスマニュアル**GetSDKVersion : SDK バージョンを取得します****形式 :**[ReturnMessage](#) GetSDKVersion(out string version)**引数 :**

名称	I/O	説明
version	O	バージョン(x.x.x.x)

戻り値 :

定義値	説明
KmSuccess	正常に完了しました

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

SDK バージョンを取得します。

CM-MISDK(C#版) リファレンスマニュアル4.3 校正・測定**GetCalibrationStatus : 校正状態を取得します**

形式 :

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

引数 :

名称	I/O	説明
calStatus	O	校正状態
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

校正状態を取得します。

校正状態を確認し、必要な校正を実施した上で、測定を実行してください。

校正状態は、条件の種類(測定径、正反射光処理など)に応じてそれぞれ管理しています。

本 API では、機器に設定されている条件で状態を取得します。

CM-MISDK(C#版) リファレンスマニュアル**PerformZeroCalibration : ゼロ校正を実行します****形式 :**[ReturnMessage](#) PerformZeroCalibration (Int32 comPort = 0)**引数 :**

名称	I/O	説明
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください
KmErCalibration	校正を正しい手順で実行していません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

ゼロ校正を実行します。

ゼロ校正が完了するまでは応答を返さないため、タイムアウトは 20 秒以上に設定してください。

ゼロ校正に失敗した場合は、実行する直前の正常状態を維持します。

ゼロ校正は毎回行う必要はありませんが、測定環境が大きく変化したときや長期間使用しなかった場合は、ゼロ校正を行ってください。

CM-MISDK(C#版) リファレンスマニュアル**PerformWhiteCalibration : 白色校正を実行します****形式 :**

[ReturnMessage](#) PerformWhiteCalibration(Int32 comPort = 0)

引数 :

名称	I/O	説明
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください
KmErCalibration	校正を正しい手順で実行していません
KmErCalibrationRequired	事前に必要な校正を実行していません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

白色校正を実行します。白色校正板の ID を確認し、一致するものを使用してください。

白色校正が完了するまでは応答を返さないため、タイムアウトは 20 秒以上に設定してください。

白色校正に失敗した場合は、実行する直前の正常状態を維持します。

校正状態は、条件の種類(測定径、正反射光処理など)に応じてそれぞれ管理しているため、条件を変更した場合には、再度校正が必要になることがあります。

[GetCalibrationStatus](#) を使用することで、校正の必要可否を判断できます。

白色校正板のデータを設定していない場合は、[SetWhiteCalibrationData](#) を使用して、データを設定してください。

CM-MISDK(C#版) リファレンスマニュアル**PerformGlossCalibration : 光沢校正を実行します****形式 :**[ReturnMessage](#) PerformGlossCalibration(Int32 comPort = 0)**引数 :**

名称	I/O	説明
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErCalibration	校正を正しい手順で実行していません
KmErCalibrationRequired	事前に必要な校正を実行していません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	×	×	×

解説 :

光沢校正を実行します。光沢基準板の ID を確認し、一致するものを使用してください。

光沢校正が完了するまでは応答を返さないため、タイムアウトは 20 秒以上に設定してください。

光沢校正に失敗した場合は、実行する直前の正常状態を維持します。

校正状態は、条件の種類(測定径、正反射光処理など)に応じてそれぞれ管理しているため、条件を変更した場合には、再度校正が必要になることがあります。

[GetCalibrationStatus](#) を使用することで、校正の必要可否を判断できます。

光沢基準板のデータを設定していない場合は、[SetGlossCalibrationData](#) を使用して、データを設定してください。

CM-MISDK(C#版) リファレンスマニュアル**PerformUserCalibration : ユーザー校正を実行します****形式 :**

[ReturnMessage](#) PerformUserCalibration(Int32 comPort = 0)

引数 :

名称	I/O	説明
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErCalibration	校正を正しい手順で実行していません
KmErCalibrationRequired	事前に必要な校正を実行していません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	×

解説 :

ユーザー校正を実行します。ユーザー校正板の ID を確認し、一致するものを使用してください。

ユーザー校正が完了するまでは応答を返さないため、タイムアウトは 20 秒以上に設定してください。

ユーザー校正に失敗した場合は、実行する直前の正常状態を維持します。

校正状態は、条件の種類(測定径、正反射光処理など)に応じてそれぞれ管理しているため、条件を変更した場合には、再度校正が必要になることがあります。

[GetCalibrationStatus](#) を使用することで、校正の必要可否を判断できます。

ユーザー校正板のデータを設定していない場合は、[SetUserCalibrationData](#) を使用して、データを設定してください。

CM-MISDK(C#版) リファレンスマニュアル**PerformMeasurement : 測定を実行します****形式 :**

[ReturnMessage](#) PerformMeasurement (Int32 comPort = 0)

引数 :

名称	I/O	説明
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

測定を実行します。

本 API は測定を開始した時点で応答を返します。

[PollingMeasurement](#) を使用して測定完了を判断し、測定完了後に [ReadLatestData](#) を使用してデータを取得してください。

本 API が正常に完了し、測定を開始した時点で保持している測定データはクリアされます。そのため、測定に失敗した場合には、以前のデータを取得することはできません。

CM-MISDK(C#版) リファレンスマニュアル**PollingMeasurement : 測定状態を取得します****形式 :**

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

引数 :

名称	I/O	説明
measStatus	O	測定状態
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください
KmErCalibrationRequired	事前に必要な校正を実行していません
KmErTileDetection	機器が傾いているため、正しく測定できませんでした

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

測定状態を取得します。

測定中から完了状態になったことを確認してから、[ReadLatestData](#) を使用してデータを取得してください。

CM-MISDK(C#版) リファレンスマニュアル**CancelMeasurement : 測定を中止します****形式 :**

[ReturnMessage](#) CancelMeasurement (Int32 comPort = 0)

引数 :

名称	I/O	説明
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

測定を中止します。

[SetAutoAverageTimes](#) で自動平均回数が複数回設定されている場合に、使用できます。

測定が実行されていない状態で本 API を実行した場合は、正常完了となります。

CM-MISDK(C#版) リファレンスマニュアル**ReadLatestData : 最新の測定データを取得します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
dataForm	I	データ種類
measurementData	I/O	測定データ
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

最新の測定データを取得します。

反射率または色彩値を取得できます。色彩値の場合は ColorValue で色彩系を選択する必要があります。

[PollingMeasurement](#) を使用して測定中から完了状態になったことを確認してから、本 API を使用してデータを取得してください。

※形式 2 は、機器に設定されている条件で色彩値を取得します。

ただし、ユーザー光源の場合は、色彩値を取得することはできません。

※26dG/26d/25d/23d の場合

オパシティ測定は単体のみです。測定モードがオパシティの場合、それぞれ以下の状態で動作します

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

CM-MISDK(C#版) リファレンスマニュアル**ReadAllLatestData : 最新の測定データをすべて取得します****形式 1 :**

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

形式 2 :

[ReturnMessage](#) ReadAllLatestData([MeasDataColor](#) measurementData, Int32 comPort = 0)

引数 :

名称	I/O	説明
measurementData	O	測定データ
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

最新の測定データをすべて取得します。反射率または色彩値を取得できます。

色彩値の場合は [MeasDataColor](#) で色彩系を選択する必要があります。

[PollingMeasurement](#) を使用して測定中から完了状態になったことを確認してから、本 API を使用してデータを取得してください。

すべての測定データを 1 回で取得できますので、データをまとめて取得する場合は [ReadLatestData](#) よりもこの API を使用することを推奨します。

※形式 2 は、機器に設定されている条件で色彩値を取得します(CM-36dG/CM-36d を除く)。

ただし、ユーザー光源の場合は、色彩値を取得することはできません。

※26dG/26d/25d/23d の場合

オパシティ測定は単体のみです。測定モードがオパシティの場合、それぞれ以下の状態で動作します

26dG	MeasModeColorAndGloss
26d	MeasModeColorOnly
25d	MeasModeColorOnly

CM-MISDK(C#版) リファレンスマニュアル

23d	MeasModeColorOnly
-----	-------------------

CM-MISDK(C#版) リファレンスマニュアル**SetWhiteCalibrationData : 白色校正データを設定します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
area	I	測定径
dataForm	I	データ種類
calData	I	校正データ ※校正板 ID は最後に設定したもので上書きされます
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

白色校正データを設定します。

本 API で設定したデータを使用して、白色校正を実行します。

形式 1 は CM-25-cG と CM-M6 のみで使用可能で、形式 2 は対応しているすべての機種で使用可能です。

※形式 1 で、CM-M6 を使用する際は、照射方向 LEFT と RIGHT それぞれに入力してください。

測定径の情報は無視されます。

CM-MISDK(C#版) リファレンスマニュアル**GetWhiteCalibrationData : 白色校正データを取得します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
area	I	測定径
dataForm	I	データ種類
calData	O	校正データ
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

白色校正データを取得します。

形式 1 は CM-25-cG と CM-M6 のみで使用可能で、形式 2 は対応しているすべての機種で使用可能です。

※形式 1 で、CM-M6 を使用する際は、照射方向 LEFT と RIGHT それぞれを取得してください。

測定径の情報は無視されます。

CM-MISDK(C#版) リファレンスマニュアル**SetGlossCalibrationData : 光沢校正データを設定します****形式 :**[ReturnMessage](#) SetGlossCalibrationData([MeasArea](#) area, [GlossCalData](#) calData, Int32 comPort = 0)**引数 :**

名称	I/O	説明
area	I	測定径
calData	I	校正データ
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	×	×	×

解説 :

光沢校正データを設定します。

本 API で設定したデータを使用して、光沢校正を実行します。

CM-MISDK(C#版) リファレンスマニュアル**GetGlossCalibrationData : 光沢校正データを取得します****形式 :**[ReturnMessage](#) GetGlossCalibrationData([MeasArea](#) area, [GlossCalData](#) calData, Int32 comPort = 0)**引数 :**

名称	I/O	説明
area	I	測定径
calData	O	校正データ
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	×	×	×

解説 :

光沢校正データを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetUserCalibrationData : ユーザー校正データを設定します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
area	I	測定径
dataForm	I	データ種類
calData	I	校正データ
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	×

解説 :

ユーザー校正データを設定します。

本 API で設定したデータを使用して、ユーザー校正を実行します。

形式 1 は CM-25-cG のみで使用可能で、形式 2 は対応しているすべての機種で使用可能です。

CM-MISDK(C#版) リファレンスマニュアル**GetUserCalibrationData : ユーザー校正データを取得します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
area	I	測定径
dataForm	I	データ種類
calData	O	校正データ
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	×

解説 :

ユーザー校正データを取得します。

形式 1 は CM-25-cG のみで使用可能で、形式 2 は対応しているすべての機種で使用可能です。

CM-MISDK(C#版) リファレンスマニュアル**SetUserCalibrationEnable : ユーザー校正使用可否を設定します****形式 :**

[ReturnMessage](#) SetUserCalibrationEnable([OnOff](#) UserCalEnable, Int32 comPort = 0)

引数 :

名称	I/O	説明
UserCalEnable	I	ユーザー校正使用可否
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	×

解説 :

ユーザー校正使用可否を設定します。

ON の場合には、ユーザー校正が使用できます。

CM-MISDK(C#版) リファレンスマニュアル**GetUserCalibrationEnable : ユーザー校正使用可否を取得します****形式 :**

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

引数 :

名称	I/O	説明
UserCalEnable	O	ユーザー校正使用可否
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	×

解説 :

ユーザー校正使用可否を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetTriggerMode : 本体キー測定を設定します****形式 :**

[ReturnMessage](#) SetTriggerMode([OnOff](#) triggerMode, Int32 comPort = 0)

引数 :

名称	I/O	説明
triggerMode	I	本体キー測定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

本体キー測定を設定します。

ON の場合には、本体キーを測定トリガーとして使用できます。

データ取得は、[IsThereTriggerData](#) を使用してデータ有無を確認し、[ReadLatestData](#) を使用することで取得してください。

CM-MISDK(C#版) リファレンスマニュアル**GetTriggerMode : 本体キー測定を取得します****概要 :**

本体キー測定を取得します。

形式 :

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

引数 :

名称	I/O	説明
triggerMode	O	本体キー測定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

本体キー測定を取得します。

CM-MISDK(C#版) リファレンスマニュアル**DeleteTriggerData : 本体キー測定データを削除します****形式 :**[ReturnMessage](#) DeleteTriggerData(Int32 comPort = 0)**引数 :**

名称	I/O	説明
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

トリガーモード中にトリガー測定したデータを削除します。

この API はデータの有無に関わらず、削除を実行します。

CM-MISDK(C#版) リファレンスマニュアル**IsThereTriggerData : 本体キー測定データの有無を取得します****形式 :**[ReturnMessage](#) IsThereTriggerData(out [IsThereData](#) IsThereData, Int32 comPort = 0)**引数 :**

名称	I/O	説明
IsThereData	O	データの有無
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

トリガーモード中にトリガー測定したデータの有無を返します。

CM-MISDK(C#版) リファレンスマニュアル**GetZeroCalibrationDate : ゼロ校正日時を取得します****形式 :**[ReturnMessage](#) GetZeroCalibrationDate([DateType](#) dateType, out DateTime date, Int32 comPort = 0)**引数 :**

名称	I/O	説明
dateType	I	日時種類
date	O	ゼロ校正日時
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErCalibrationRequired	事前に必要な校正を実行していません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

ゼロ校正日時を取得します。

校正を実行していない場合は、[KmErCalibrationRequired](#) を返します。

CM-MISDK(C#版) リファレンスマニュアル**GetWhiteCalibrationDate : 白色校正日時を取得します****形式 :**

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

引数 :

名称	I/O	説明
date	O	白色校正日時
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErCalibrationRequired	事前に必要な校正を実行していません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

白色校正日時を取得します。

校正を実行していない場合は、[KmErCalibrationRequired](#) を返します。

CM-MISDK(C#版) リファレンスマニュアル**GetGlossCalibrationDate : 光沢校正日時を取得します****形式 :**

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

引数 :

名称	I/O	説明
date	O	光沢校正日時
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErCalibrationRequired	事前に必要な校正を実行していません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	×	×	×

解説 :

光沢校正日時を取得します。

校正を実行していない場合は、[KmErCalibrationRequired](#) を返します。

CM-MISDK(C#版) リファレンスマニュアル**GetUserCalibrationDate : ユーザー校正日時を取得します****形式 :**[ReturnMessage](#) GetUserCalibrationDate(out DateTime date, Int32 comPort = 0)**引数 :**

名称	I/O	説明
date	O	ユーザー校正日時
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErCalibrationRequired	事前に必要な校正を実行していません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	×

解説 :

ユーザー校正日時を取得します。

校正を実行していない場合は、[KmErCalibrationRequired](#) を返します。

CM-MISDK(C#版) リファレンスマニュアル**ClearUvAdjustInfo : 蛍光調整用の各種データをクリアします****形式 :**[ReturnMessage](#) ClearUvAdjustInfo(Int32 comPort = 0)**引数 :**

名称	I/O	説明
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します**解説 :**

蛍光調整用の各種データをクリアします。

プロファイル、WI、Tint、ISO ブライツネス、Ganz&Griesser、および蛍光係数・蛍光補正値の情報が該当します。

機器内の蛍光係数・蛍光補正値をクリアする場合は、[ClearCoefForUvAdjust](#) を使用してください。

CM-MISDK(C#版) リファレンスマニュアル**SetProfileForUvAdjust : 蛍光調整用のプロファイルデータを設定します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明		
uvtype	I	設定するデータの種類		
data	I	蛍光調整用プロファイルデータ <table><tr><td>値範囲</td><td>0.01~200.00</td></tr></table> ※ GetInstrumentInfo で取得した DataSize の個数	値範囲	0.01~200.00
値範囲	0.01~200.00			
comport	I	仮想 COM ポート番号		

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用のプロファイルデータを設定します。

形式 1 では、設定可能なすべてのデータ種類に、同じ値が設定されます。

CM-MISDK(C#版) リファレンスマニュアル**GetProfileForUvAdjust : 蛍光調整用のプロファイルデータを取得します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	取得するデータの種類
data	O	蛍光調整用プロファイルデータ ※ GetInstrumentInfo で取得した DataSize の個数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用のプロファイルデータを取得します。

形式 1 の場合は、データ種類の先頭のデータを返します。

CM-MISDK(C#版) リファレンスマニュアル**SetWiForUvAdjust : 蛍光調整用の WI を設定します****形式 1 :**

[ReturnMessage](#) SetWiForUvAdjust([UvAdjustIndex](#) data, Int32 comPort = 0)

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	設定するデータの種類
data	I	蛍光調整用 WI
		値範囲 40.00~250.00
		許容値範囲 0.20~3.00
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用の WI を設定します。

形式 1 では、設定可能なすべてのデータ種類に、同じ値が設定されます。

CM-MISDK(C#版) リファレンスマニュアル**GetWiForUvAdjust : 蛍光調整用の WI を取得します****形式 1 :**

[ReturnMessage](#) GetWiForUvAdjust([UvAdjustIndex](#) data, Int32 comPort = 0)

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	取得するデータの種類
data	O	蛍光調整用 WI
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用の WI を取得します。

形式 1 の場合は、データ種類の先頭のデータを返します。

CM-MISDK(C#版) リファレンスマニュアル**SetTintForUvAdjust : 蛍光調整用の Tint を設定します****形式 1 :**

[ReturnMessage](#) SetTintForUvAdjust([UvAdjustIndex](#) data, Int32 comPort = 0)

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	設定するデータの種類
data	I	蛍光調整用 Tint
		値範囲 -6.00~6.00
		許容値範囲 0.05~0.30
Comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用の Tint を設定します。

形式 1 では、設定可能なすべてのデータ種類に、同じ値が設定されます。

CM-MISDK(C#版) リファレンスマニュアル**GetTintForUvAdjust : 蛍光調整用の Tint を取得します****形式 1 :**

[ReturnMessage](#) GetTintForUvAdjust([UvAdjustIndex](#) data, Int32 comPort = 0)

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	取得するデータの種類
data	O	蛍光調整用 Tint
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用の Tint を取得します。

形式 1 の場合は、データ種類の先頭のデータを返します。

CM-MISDK(C#版) リファレンスマニュアル**SetIsoBrightnessForUvAdjust : 蛍光調整用の ISO ブライツネスを設定します****形式 1 :**

[ReturnMessage](#) SetIsoBrightnessForUvAdjust([UvAdjustIndex](#) data, Int32 comPort = 0)

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	設定するデータの種類
data	I	蛍光調整用 ISO ブライツネス
		値範囲 40.00~250.00
		許容値範囲 0.50~3.00
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用の ISO ブライツネスを設定します。

形式 1 では、設定可能なすべてのデータ種類に、同じ値が設定されます。

CM-MISDK(C#版) リファレンスマニュアル**GetIsoBrightnessForUvAdjust : 蛍光調整用の ISO ブライツネスを取得します****形式 1 :**

[ReturnMessage](#) GetIsoBrightnessForUvAdjust([UvAdjustIndex](#) data, Int32 comPort = 0)

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	取得するデータの種類
data	O	蛍光調整用 ISO ブライツネス
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用の ISO ブライツネスを取得します。

形式 1 の場合は、データ種類の先頭のデータを返します。

CM-MISDK(C#版) リファレンスマニュアル**SetGanzForUvAdjust : 蛍光調整用の Ganz&Griesser を設定します****形式 1 :**

[ReturnMessage](#) SetGanzForUvAdjust([UvAdjustGG](#) data, Int32 comPort = 0)

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	設定するデータの種類
data	I	蛍光調整用 Ganz & Griesser
		WI 範囲 40.00~250.00
		Tint 範囲 -6.00~6.00
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用の Ganz&Griesser を設定します。

形式 1 では、設定可能なすべてのデータ種類に、同じ値が設定されます。

CM-MISDK(C#版) リファレンスマニュアル**GetGanzForUvAdjust : 蛍光調整用の Ganz&Griesser を取得します****形式 1 :**

[ReturnMessage](#) GetGanzForUvAdjust([UvAdjustGG](#) data, Int32 comPort = 0)

形式 2 :

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

引数 :

名称	I/O	説明
uvtype	I	取得するデータの種類
data	O	蛍光調整用 Ganz&Griesser
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用の Ganz&Griesser を取得します。

形式 1 の場合は、データ種類の先頭のデータを返します。

CM-MISDK(C#版) リファレンスマニュアル**SetDataForUvAdjust : 蛍光調整用のデータを設定します****形式 :**

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

引数 :

名称	I/O	説明		
dataType	I	蛍光調整用データ種類		
num	I	番号 ※Ganz & Griesser 以外の場合は 0 のみ ※Ganz & Griesser の場合は、0~3 または 0~4		
dataFull	I	UV フル測定データ <table><tr><td>値範囲</td><td>0.00~300.00</td></tr></table> ※ GetInstrumentInfo で取得した DataSize の個数	値範囲	0.00~300.00
値範囲	0.00~300.00			
dataCut	I	UV カット測定データ <table><tr><td>値範囲</td><td>0.00~300.00</td></tr></table> ※ GetInstrumentInfo で取得した DataSize の個数	値範囲	0.00~300.00
値範囲	0.00~300.00			
Comport	I	仮想 COM ポート番号		

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用のデータを設定します。

Ganz & Griesser の場合は、必ず本 API を使用してデータを設定する必要があります。

CM-MISDK(C#版) リファレンスマニュアル**GetDataForUvAdjust : 蛍光調整用のデータを取得します****形式 :**

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

引数 :

名称	I/O	説明
dataType	I	蛍光調整用データ種類
num	I	番号 ※Ganz & Griesser 以外の場合は 0 のみ ※Ganz & Griesser の場合は、0~3 または 0~4
dataFull	O	UV フル測定データ ※ GetInstrumentInfo で取得した DataSize の個数
dataCut	O	UV カット測定データ ※ GetInstrumentInfo で取得した DataSize の個数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整用のデータを取得します。

CM-MISDK(C#版) リファレンスマニュアル**PerformUvAdjust : 蛍光調整を実行し、係数を機器に設定します****形式 :**

[ReturnMessage](#) PerformUvAdjust([CondUvAdjust](#) cond, Int32 comPort = 0)

引数 :

名称	I/O	説明
cond	I	蛍光調整用の条件
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません(必要なデータが揃っていません)
KmErUvAdjust	測定試料に蛍光が含まれていません
KmErCalculateCoef	蛍光係数を算出できません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

蛍光調整を実行します。

本 API では測定を行い、指定した条件で蛍光係数を算出し、係数を機器に反映します。

条件に UVADJ_GG を指定している場合は、[KmErInvalidParameter](#) を返します。

Ganz & Griesser で蛍光調整を行う場合は、[PerformUvAdjustUsingData](#) を使用してください。

CM-MISDK(C#版) リファレンスマニュアル**PerformUvAdjustUsingData : 蛍光調整を実行し、係数を機器に設定します****形式 :**[ReturnMessage](#) PerformUvAdjustUsingData([CondUvAdjust](#) cond, Int32 comPort = 0)**引数 :**

名称	I/O	説明
cond	I	蛍光調整用の条件
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません(必要なデータが揃っていません)
KmErUvAdjust	測定試料に蛍光が含まれていません
KmErCalculateCoef	蛍光係数を算出できません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します**解説 :**

蛍光調整を実行します。

本 API では [SetDataForUvAdjust](#) のデータを使用し、指定した条件で蛍光係数を算出し、係数を機器に反映します。

CM-MISDK(C#版) リファレンスマニュアル**ClearCoefForUvAdjust : 機器内の蛍光調整係数をクリアします****形式 :**[ReturnMessage](#) ClearCoefForUvAdjust(Int32 comPort = 0)**引数 :**

名称	I/O	説明
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します**解説 :**

機器内に保存された蛍光調整係数をクリアします。

CM-MISDK(C#版) リファレンスマニュアル**SetCoefForUvAdjust : 蛍光係数を設定します****形式 :**

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

引数 :

名称	I/O	説明
area	I	測定径
uvtype	I	設定するデータの種類
cond	I	蛍光調整用の条件
coef	I	蛍光調整係数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmWarning	処理が正常に完了しました(警告を含んでいます)
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

機器内に蛍光係数を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetCoefForUvAdjust : 蛍光係数を取得します****形式 :**

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

引数 :

名称	I/O	説明
area	I	測定径
uvtype	I	取得するデータの種類
cond	O	蛍光調整用の条件
coef	O	蛍光調整係数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmWarning	処理が正常に完了しました(警告を含んでいます)
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

※[機種・バージョン](#)依存します

解説 :

機器内の蛍光係数を取得します。

CM-MISDK(C#版) リファレンスマニュアル4.4 測定条件**SetMeasurementArea : 測定径を設定します**

形式 :

[ReturnMessage](#) SetMeasurementArea([MeasArea](#) area, Int32 comPort = 0)

引数 :

名称	I/O	説明
area	I	測定径
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	×	×	×	×

解説 :

測定径を設定します。

条件を変更すると、再度校正が必要になることがあります。

[GetCalibrationStatus](#) を使用することで、校正の必要可否を判断できます。

CM-MISDK(C#版) リファレンスマニュアル**GetMeasurementArea : 測定径を取得します****形式 :**

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

引数 :

名称	I/O	説明
area	O	測定径
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	×	×

解説 :

測定径を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetMeasurementType : 測定タイプを設定します****形式 :**

[ReturnMessage](#) GetMeasurementType([MeasType](#) inType, Int32 comPort = 0)

引数 :

名称	I/O	説明
inType	I	測定タイプ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	×	×	×	×

解説 :

測定タイプを設定します。

条件を変更すると、再度校正が必要になることがあります。

[GetCalibrationStatus](#) を使用することで、校正の必要可否を判断できます。

CM-MISDK(C#版) リファレンスマニュアル**GetMeasurementType : 測定タイプを取得します****形式 :**

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

引数 :

名称	I/O	説明
outType	O	測定タイプ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	×	×	×	×

解説 :

測定タイプを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetMeasurementAngle : 測定角度を設定します****形式 :**[ReturnMessage](#) SetMeasurementAngle([MeasAngle](#) angle, Int32 comPort = 0)**引数 :**

名称	I/O	説明
angle	I	測定角度
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	○	×	×	×	×

解説 :

測定角度を設定します。

角度は 1 コ以上であれば、任意の組み合わせで設定できます。

CM-MISDK(C#版) リファレンスマニュアル**GetMeasurementAngle : 測定角度を取得します****形式 :**

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

引数 :

名称	I/O	説明
angle	O	測定角度
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	○	×	×	×	×

解説 :

測定角度を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetTiltDetection : 傾き検知を設定します****形式 :**

[ReturnMessage](#) SetTiltDetection([OnOff](#) tilt, Int32 comPort = 0)

引数 :

名称	I/O	説明
tilt	I	傾き検知設定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	○	×	×	×	×

解説 :

傾き検知を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetTiltDetection : 傾き検知の有効・無効を取得します****形式 :**[ReturnMessage](#) GetTiltDetection(out [OnOff](#) tilt, Int32 comPort = 0)**引数 :**

名称	I/O	説明
tilt	O	傾き検知設定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	○	×	×	×	×

解説 :

傾き検知の有効・無効を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetMeasurementMode : 測定モードを設定します****形式 :**[ReturnMessage](#) SetMeasurementMode ([MeasCondMode](#) Mode, Int32 comPort = 0)**引数 :**

名称	I/O	説明
Mode	I	測定モード
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

測定モードを設定します。

条件を変更すると、再度校正が必要になることがあります。

[GetCalibrationStatus](#) を使用することで、校正の必要可否を判断できます。

CM-MISDK(C#版) リファレンスマニュアル**GetMeasurementMode : 測定モードを取得します****形式 :**

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

引数 :

名称	I/O	説明
Mode	O	測定モード
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

測定モードを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetSpecularComponent : 正反射光処理を設定します**

形式 :

[ReturnMessage](#) SetSpecularComponent([MeasCondScie](#) scie, Int32 comPort = 0)

引数 :

名称	I/O	説明
scie	I	正反射光処理
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	○

解説 :

正反射光処理を設定します。

条件を変更すると、再度校正が必要になることがあります。

[GetCalibrationStatus](#) を使用することで、校正の必要可否を判断できます。

CM-MISDK(C#版) リファレンスマニュアル**GetSpecularComponent : 正反射光処理を取得します****形式 :**[ReturnMessage](#) GetSpecularComponent(out [MeasCondScie](#) scie, Int32 comPort = 0)**引数 :**

名称	I/O	説明
scie	O	正反射光処理
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	○

解説 :

正反射光処理を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetUv : UV 条件を設定します****形式 :**[ReturnMessage](#) SetUv([MeasCondUv](#) uv, Int32 comPort = 0)**引数 :**

名称	I/O	説明
uv	I	UV 条件
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

解説 :

UV 条件を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetUv : UV 条件を取得します****形式 :**[ReturnMessage](#) GetUv(out [MeasCondUv](#) uv, Int32 comPort = 0)**引数 :**

名称	I/O	説明
uv	O	UV 条件
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	×	×

解説 :

UV 条件を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetAutoAverageTimes : 自動平均回数を設定します****形式 :**[ReturnMessage](#) SetAutoAverageTimes(Int32 times, Int32 comPort = 0)**引数 :**

名称	I/O	説明
times	I	自動平均回数(1~10 回)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

自動平均回数を設定します。

[PerformMeasurement](#) では、本条件で設定した回数分の測定を実行します。

CM-MISDK(C#版) リファレンスマニュアル**GetAutoAverageTimes : 自動平均回数を取得します****形式 :**[ReturnMessage](#) GetAutoAverageTimes(out Int32 times, Int32 comPort = 0)**引数 :**

名称	I/O	説明
times	O	自動平均回数(1~10 回)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

自動平均回数を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetManualAverageTimes : 手動平均回数を設定します****形式 :**[ReturnMessage](#) SetManualAverageTimes(Int32 times, Int32 comPort = 0)**引数 :**

名称	I/O	説明
times	I	手動平均回数 ※設定範囲は、機種・バージョン依存します。 付録 C を参照してください。
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

手動平均回数を設定します。

単体測定時に使用し、通信使用時には適用されません。

CM-MISDK(C#版) リファレンスマニュアル**GetManualAverageTimes : 手動平均回数を取得します****形式 :**[ReturnMessage](#) GetManualAverageTimes(out Int32 times, Int32 comPort = 0)**引数 :**

名称	I/O	説明
times	O	手動平均回数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

手動平均回数を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetManualAverageSaveMode : 手動平均方法を設定します****形式 :**[ReturnMessage](#) SetManualAverageSaveMode([SaveMode](#) mode, Int32 comPort = 0)**引数 :**

名称	I/O	説明
mode	I	手動平均保存方法
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

手動平均保存方法を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetManualAverageSaveMode : 手動平均保存方法を取得します****形式 :**

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

引数 :

名称	I/O	説明
mode	O	手動平均保存方法
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

手動平均保存方法を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetCondSMC : SMC 条件を設定します****形式 :**

[ReturnMessage](#) SetCondSMC([MeasCondSMC](#) smc, Int32 comPort = 0)

引数 :

名称	I/O	説明
smc	I	SMC 条件
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

SMC 条件を設定します。

単体測定時に使用し、通信使用時には適用されません。

CM-MISDK(C#版) リファレンスマニュアル**GetCondSMC : SMC 条件を取得します****形式 :**

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

引数 :

名称	I/O	説明
smc	O	SMC 条件
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

SMC 条件を取得します。

CM-MISDK(C#版) リファレンスマニュアル4.5 表示条件**SetDisplayType : 表示形式を設定します****形式 :**[ReturnMessage](#) SetDisplayType([DisplayType](#) displayType, Int32 comPort = 0)**引数 :**

名称	I/O	説明
displayType	I	表示形式
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

表示形式を設定します。

表示形式は、任意の組み合わせで設定できます。

機種によって設定できる項目が異なりますので、注意してください。

CM-MISDK(C#版) リファレンスマニュアル**GetDisplayType : 表示形式を取得します****形式 :**[ReturnMessage](#) GetDisplayType(out [DisplayType](#) displayType, Int32 comPort = 0)**引数 :**

名称	I/O	説明
displayType	O	表示形式
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

表示形式を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetObserverAndIlluminant : 観察視野・光源を設定します**

形式 :

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

引数 :

名称	I/O	説明
num	I	視野・光源番号(0~1)
obs	I	観察視野
ill	I	観察光源
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

観察視野・光源を設定します。

視野・光源番号が 1 の場合に、光源に「なし」を指定することで、1 つのみ使用することができます。

機種によって設定できる項目が異なりますので、ご注意ください。

ユーザー光源を使用する場合は、事前に [SetUserIlluminant](#) を使用して、光源データを登録してください。

CM-MISDK(C#版) リファレンスマニュアル**GetObserverAndIlluminant : 観察視野・光源を取得します****形式 :**

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

引数 :

名称	I/O	説明
num	I	視野・光源番号(0~1)
obs	O	観察視野
ill	O	観察光源
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

観察視野・光源を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetUserIlluminant : ユーザー光源を設定します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
illData	I	光源データ 360~780nm、5nm ピッチで 85 コのデータ
name	I	名前(10 文字以内)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	×

解説 :

ユーザー光源を設定します。

光源データは、360~780nm(5nm ピッチ)の 85 コのデータを設定してください。

名前を設定できる機種で形式 1 を使用した場合は、「User」を設定します。

名前を設定できない機種で形式 2 を使用した場合は、名前の設定を無視します。

CM-MISDK(C#版) リファレンスマニュアル**GetUserIlluminant : ユーザー光源を取得します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
illData	O	光源データ 360~780nm、5nm ピッチで 85 コのデータ
name	O	名前
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	×

解説 :

ユーザー光源を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetColorSpace : 表色系を設定します****形式 :**

[ReturnMessage](#) SetColorSpace([ColorSpace](#) colorSpace, Int32 comPort = 0)

引数 :

名称	I/O	説明
colorSpace	I	表色系
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

表色系を設定します。

機種によって、設定できる項目が異なりますので、ご注意ください。

CM-MISDK(C#版) リファレンスマニュアル**GetColorSpace : 表色系を取得します****形式 :**

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

引数 :

名称	I/O	説明
colorSpace	O	表色系
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

表色系を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetEquation : 色差式を設定します****形式 :**

[ReturnMessage](#) SetEquation([Equation](#) equation, Int32 comPort = 0)

引数 :

名称	I/O	説明
equation	I	色差式
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

色差式を設定します。

機種によって、設定できる項目が異なりますので、ご注意ください。

CM-MISDK(C#版) リファレンスマニュアル**GetEquation : 色差式を取得します****形式 :**

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

引数 :

名称	I/O	説明
equation	O	色差式
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

色差式を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetCustomItem : カスタム表示項目を設定します****形式 :**[ReturnMessage](#) SetCustomItem (Int32 customNum, [CustomItem](#) customItem, Int32 comPort = 0)**引数 :**

名称	I/O	説明
customNum	I	カスタム表示番号(0~13 の 14 コ)
customItem	I	カスタム表示項目
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

カスタム表示項目を設定します。

カスタム項目を表示するためには、[SetDisplayType](#) を使用し、「カスタム」を有効にしてください。

CM-MISDK(C#版) リファレンスマニュアル**GetCustomItem : カスタム表示項目を取得します****形式 :**[ReturnMessage](#) GetCustomItem(Int32 customNum, out [CustomItem](#) customItem, Int32 comPort = 0)**引数 :**

名称	I/O	説明
customNum	I	カスタム表示番号(0~13 の 14 コ)
customItem	O	カスタム表示項目
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

カスタム表示項目を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetDisplayData : 表示するデータの照射方向を設定します****形式 :**

[ReturnMessage](#) SetDisplayData([DisplayData](#) data, Int32 comPort = 0)

引数 :

名称	I/O	説明
data	I	表示するデータ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	○	×	×	×	×

解説 :

表示するデータの照射方向を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetDisplayData : 表示するデータの照射方向を取得します****形式 :**[ReturnMessage](#) GetDisplayData (out [DisplayData](#) data, Int32 comPort = 0)**引数 :**

名称	I/O	説明
data	O	表示しているデータ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	○	×	×	×	×

解説 :

表示しているデータの照射方向を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetUserEquation : ユーザーインデックスを設定します**

形式 :

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

引数 :

名称	I/O	説明
num	I	ユーザーインデックス番号(0~2)
data	I	ユーザーインデックス
name	I	ユーザークラス
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

ユーザーインデックスを設定します。

ユーザーインデックスに使用できる表色値は、[SetColorSpace](#)、[SetEquation](#)、[SetCustomItem](#) で選択されている項目のみです。ユーザーインデックスの例 : ΔE^*ab 相当 $\text{SQRT}(\text{POW}([DL]) + \text{POW}([DA]) + \text{POW}([DB]))$

ユーザーインデックスには、以下の変数を使用できます。

[L]	L*	[DL]	ΔL^*	[MI]	MI
[A]	a*	[DA]	Δa^*	[WIE]	WI(E313-73)
[B]	b*	[DB]	Δb^*	[DWIE]	$\Delta WI(E313-73)$
[C]	C*	[DC]	ΔC^*	[WIC]	WI(CIE)
[H]	h	[DH]	ΔH^*	[DWIC]	$\Delta WI(CIE)$

CM-MISDK(C#版) リファレンスマニュアル

[HL]	L(Hunter)	[DHL]	ΔL	[TINT]	Tint
[HA]	a(Hunter)	[DHA]	Δa	[DTINT]	$\Delta Tint$
[HB]	b(Hunter)	[DHB]	Δb	[YIE]	YI(E313-73)
[X]	X	[DX]	ΔX	[DYIE]	$\Delta YI(E313-73)$
[Y]	Y	[DY]	ΔY	[YID]	YI(D1925)
[Z]	Z	[DZ]	ΔZ	[DYID]	$\Delta YI(D1925)$
[SX]	x	[DSX]	Δx	[BISO]	B
[SY]	y	[DSY]	Δy	[DBISO]	ΔB
[GU]	GU	[DGU]	ΔGU		
[DE]	ΔE^{*ab}	[DE94]	ΔE^{*94}	[DEH]	$\Delta E(\text{Hunter})$
[CMC]	CMC	[DE00]	$\Delta E00$		

※CM-26dG/26d/25d/23d で、項目を SCI と SCE で限定する場合は、変数にそれぞれ「I」と「E」を付けてください。

例えば、 $L^{*}(SCI)+L^{*}(SCE)$ を行う場合は、 $[LI]+[LE]$ とします。

ユーザーインデックスに使用できる演算子・関数は以下の通りです。

+	$[A]+[B]$	$A+B$
-	$[A]-[B]$	$A-B$
*	$[A]*[B]$	$A \times B$
/	$[A]/[B]$	$A \div B$
POW	POW([A])	A の 2 乗
SQRT	SQRT([A])	A の平方根
ABS	ABS([A])	A の絶対値
SIN	SIN([A])	A(degree)の正弦
COS	COS([A])	A(degree)の余弦
TAN	TAN([A])	A(degree)の正接
ASIN	ASIN([A])	逆正弦(degree)
ACOS	ACOS([A])	逆余弦(degree)
ATAN	ATAN([A])	逆正接(degree)
LOG	LOG([A])	常用対数
LN	LN([A])	自然対数
EXP	EXP([A])	指数関数
POW2	POW2([A],[B])	べき乗関数(A の B 乗)

ユーザークラスの入力フォーマット

```
CLASS(n, "str1", d1, "str2", d2, "str3", d3, "str4", d4, "str5", ...)
```

CM-MISDK(C#版) リファレンスマニュアル

n : クラス分けするための閾値の数を示します。(クラス数-1)となり、50 以内で指定できます。

“str1”,d1 : ユーザークラスの判別結果が d1 以上であるとき、機器の結果欄に“str1”と表示します。

d1 は 20 文字以下の数値設定が可能ですが、演算上の有効数字は 5 桁です。数値の大きい閾値の順に左から入力し、すべての閾値以上に分類されない場合に表示するクラスも必ず設定してください。

また、入力する文字(CLASS()を含む)が合計で半角 200 文字以内になるようにしてください。

ユーザークラスを使用しない場合は、ユーザークラスを未入力にします。

小数点には“.”、パラメータの区切りには“,”を使用してください。

ユーザークラスの入力フォーマット

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

ユーザーインデックスの結果を以下のように 5 分類します。

ユーザーインデックス結果	クラス
4 以上	A
3 以上	B
2 以上	C
1 以上	D
1 より小さい	E

CM-MISDK(C#版) リファレンスマニュアル**GetUserEquation : ユーザーインデックスを取得します****形式 1 :**

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

引数 :

名称	I/O	説明
num	I	ユーザーインデックス番号(0~2)
data	O	ユーザーインデックス
name	O	ユーザークラス
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

ユーザーインデックスを取得します。

CM-MISDK(C#版) リファレンスマニュアル4.6 データ**SetActiveTarget : アクティブ基準色番号を設定します**

形式 :

[ReturnMessage](#) SetActiveTarget(Int32 num, Int32 comPort = 0)

引数 :

名称	I/O	説明
num	I	基準色番号
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

アクティブ基準色番号を設定します。

アクティブ基準色は、設定以降の測定時に関連付けされる基準色番号として使用します。

CM-MISDK(C#版) リファレンスマニュアル**GetActiveTarget : アクティブ基準色番号を取得します****形式 :**[ReturnMessage](#) GetActiveTarget(out Int32 num, Int32 comPort = 0)**引数 :**

名称	I/O	説明
num	O	基準色番号
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

アクティブ基準色番号を取得します。

CM-MISDK(C#版) リファレンスマニュアル**GetTargetNumberList : 保存されている基準色番号のリストを取得します****形式 :**[ReturnMessage](#) GetTargetNumberList (out List<Int32> targetlist, Int32 comPort = 0)**引数 :**

名称	I/O	説明
targetlist	O	登録されている基準色番号のリスト
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

登録されている基準色番号のリストを取得します。

この API は保存されているすべての基準色データの基準色番号を返します。

CM-MISDK(C#版) リファレンスマニュアル**GetTargetNumberList2 : 表示フィルター適用時の基準色番号のリストを取得します****形式 :**[ReturnMessage](#) GetTargetNumberList2 (out List<Int32> targetlist, Int32 comPort = 0)**引数 :**

名称	I/O	説明
targetlist	O	登録されている基準色番号のリスト
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

表示フィルターを適用したときの基準色番号のリストを取得します。

CM-MISDK(C#版) リファレンスマニュアル**DeleteTargetData : 基準色を削除します****形式 :**[ReturnMessage](#) DeleteTargetData(Int32 num, Int32 comPort = 0)**引数 :**

名称	I/O	説明
num	I	基準色番号 ※-1 を指定した場合は、全削除します
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の基準色を削除します。

番号として-1 を指定した場合は、全削除となります。

CM-MISDK(C#版) リファレンスマニュアル**SetTargetData : 基準色データを設定します****形式 1 :**

[ReturnMessage](#) SetTargetData(Int32 num, [TargetData](#) target, Int32 comPort = 0)

形式 2 :

[ReturnMessage](#) SetTargetData(Int32 num, [TargetDataPack](#) target, Int32 comPort = 0)

引数 :

名称	I/O	説明
num	I	基準色番号
target	I	データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の基準色データを設定します。

既に基準色が存在する場合には、上書きされます。

[SetTargetProtect](#) でデータ保護が有効になっている場合は、設定できません。

形式 1 は CM-25-cG と CM-M6 のみで使用可能で、形式 2 は対応しているすべての機種で使用可能です。

※CM-M6 の場合は 6 角度をまとめて設定する必要があります。

色彩値データをやり取りする場合は、形式 2 をご使用ください。

※オパシティでは、色彩値の登録はできません。

CM-MISDK(C#版) リファレンスマニュアル**GetTargetData : 基準色データを取得します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
num	I	基準色番号
target	O	データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErNotSupported	色彩値データ取得に対応していません (形式 2 の API を使用してください)

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の基準色データを取得します。

形式 1 は CM-25cG と CM-M6 のみで使用可能で、形式 2 は対応しているすべての機種で使用可能です。

形式 2 は保存されているすべてのデータを 1 回で取得できますので、こちらの API を使用することを推奨します。

CM-M6 についても [GetAllTargetData](#) より早くすべてのデータを取得できます。

形式 1 の場合 :

- ・ CM-25cG の場合
ColorValue クラスの ColorSpace を入力してください。
- ・ CM-M6 の場合
ColorValue クラスの ColorSpace を入力してください。

CM-MISDK(C#版) リファレンスマニュアル

DataForm クラスの DataType と IrradiationDirection を入力してください。

CM-MISDK(C#版) リファレンスマニュアル**GetAllTargetData : 基準色データを取得します****形式 :**[ReturnMessage](#) GetAllTargetData(Int32 num, [TargetData](#) target, Int32 comPort = 0)**引数 :**

名称	I/O	説明
num	I	基準色番号
target	O	データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	○	×	×	×	×

解説 :

指定した番号の基準色データを取得します。

※この API は CM-M6 専用です。

ColorValue クラスの ColorSpace を入力してください。

取得した演算値は ColorValue の AllAngleValue に格納されます。

DataForm クラスの IrradiationDirection を入力してください。

[GetTargetData](#) の形式 2 は、保存されているすべてのデータを 1 回で取得できますので、そちらの API を使用することを推奨します。本 API よりも短い時間ですべてのデータを取得できます。

CM-MISDK(C#版) リファレンスマニュアル**SetToleranceForTarget : 基準色の限界値を設定します****形式 :**

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

引数 :

名称	I/O	説明
num	I	基準色番号(1~[GetTargeNumberList で取得した数])
numObsIll	I	視野・光源番号(0~1)
dataForm	I	データ種類
tolerance	I	限界値データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の基準色限界値を設定します。

[SetTargetProtect](#) でデータ保護が有効になっている場合は、設定できません。

※キーで指定した Tolerance 以外は変更されません

CM-MISDK(C#版) リファレンスマニュアル**GetToleranceForTarget : 基準色の限界値を取得します****形式 :**

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

引数 :

名称	I/O	説明
num	I	基準色番号(1~[GetTargeNumberList で取得した数])
numObsIll	I	視野・光源番号(0~1)
dataForm	I	データ種類
tolerance	O	限界値データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の基準色限界値を取得します。

※ToleranceData クラスには取得したい限界値の Key と空の ToleranceParam を入力してください

CM-MISDK(C#版) リファレンスマニュアル**SetParametricForTarget : 基準色のパラメトリック係数を設定します****形式 :**

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

引数 :

名称	I/O	説明
num	I	基準色番号(1~[GetTargeNumberList で取得した数])
dataType	I	データ種類
parametric	I	パラメトリック係数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の基準色パラメトリック係数を設定します。

[SetTargetProtect](#) でデータ保護が有効になっている場合は、設定できません。

CM-MISDK(C#版) リファレンスマニュアル**GetParametricForTarget : 基準色のパラメトリック係数を取得します****形式 :**

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

引数 :

名称	I/O	説明
num	I	基準色番号(1~[GetTargeNumberList で取得した数])
dataType	I	データ種類
parametric	O	パラメトリック係数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください
KmErNoData	データが存在しません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の基準色パラメトリック係数を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetTargetFilter : 基準色フィルター条件を設定します****形式 :**[ReturnMessage](#) SetTargetFilter([FilterIndex](#) index, List<Int32> numGroup, Int32 comPort = 0)**引数 :**

名称	I/O	説明
index	I	フィルター属性
numGroup	I	グループ番号(0~50) ※フィルター属性が「グループ」の場合のみ使用します。それ以外の場合は 0 を指定してください
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

基準色フィルター条件を設定します。

フィルターを OFF 以外にすると、基準色は条件と一致する番号のデータのみ表示されます。

CM-MISDK(C#版) リファレンスマニュアル**GetTargetFilter : 基準色フィルター条件を取得します****形式 :**[ReturnMessage](#) GetTargetFilter(out [FilterIndex](#) index, out List<Int32> numGroup, Int32 comPort = 0)**引数 :**

名称	I/O	説明
index	O	フィルター属性
numGroup	O	グループ番号(0~50) ※フィルター属性が「グループ」の場合のみ取得できます。それ以外 の場合は0となります。
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

基準色フィルター条件を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetTargetProtect : 基準色保護を設定します****形式 :**[ReturnMessage](#) SetTargetProtect([OnOff](#) protect, Int32 comPort = 0)**引数 :**

名称	I/O	説明
protect	I	保護設定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

基準色保護を設定します。

基準色保護を ON にした場合には、新規測定(保存)のみ操作可能です。

CM-MISDK(C#版) リファレンスマニュアル**GetTargetProtect : 基準色保護を取得します****形式 :**[ReturnMessage](#) GetTargetProtect(out [OnOff](#) protect, Int32 comPort = 0)**引数 :**

名称	I/O	説明
protect	O	保護設定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

基準色保護を取得します。

CM-MISDK(C#版) リファレンスマニュアル**GetSampleCount : 測定値保存数を取得します****形式 :**

[ReturnMessage](#) GetSampleCount(out Int32 count, Int32 comPort = 0)

引数 :

名称	I/O	説明
count	O	測定値数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

測定値数を取得します。

CM-MISDK(C#版) リファレンスマニュアル**DeleteSampleData : 測定値を削除します****形式 :**

[ReturnMessage](#) DeleteSampleData(Int32 num, Int32 comPort = 0)

引数 :

名称	I/O	説明
num	I	測定値番号 ※-1 を指定した場合は、全削除します
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の測定値を削除します。

番号として-1 を指定した場合は、全削除となります。

CM-MISDK(C#版) リファレンスマニュアル**GetSampleData : 測定値データを取得します****形式 1 :**

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

形式 2 :

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

引数 :

名称	I/O	説明
num	I	測定値番号
sample	O	データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

指定した番号の測定値データを取得します。

形式 1 は CM-25-cG と CM-M6 のみで使用可能で、形式 2 は対応しているすべての機種で使用可能です。

形式 2 は保存されているすべてのデータを 1 回で取得できますので、こちらの API を使用することを推奨します。

CM-M6 についても GetAllSampleData より早くすべてのデータを取得できます。

形式 1 の場合

- ・ CM-25 c G の場合

ColorValue クラスの ColorSpace を入力してください。

- ・ CM-M6 の場合

ColorValue クラスの ColorSpace を入力してください。

DataForm クラスの DataType と IrradiationDirection を入力してください。

CM-MISDK(C#版) リファレンスマニュアル**GetAllSampleData : 測定値データを取得します****形式 :**

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

引数 :

名称	I/O	説明
num	I	測定値番号
sample	O	データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	○	×	×	×	×

解説 :

指定した番号の測定値データを取得します。

※この API は CM-M6 専用です。

ColorValue クラスの ColorSpace を入力してください。

取得した演算値は SampleData クラス内の ColorValue クラスの AllAngleValue に格納されます。

取得した分光反射率は SampleData クラスの AllAngleData に格納されます。

DataForm クラスの IrradiationDirection を入力してください。

CM-MISDK(C#版) リファレンスマニュアル4.7 その他**SetActiveGroup : アクティブグループ番号を設定します**

形式 :

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

引数 :

名称	I/O	説明
numGroup	I	グループ番号(0~50)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

アクティブグループ番号を設定します。

アクティブグループ番号は、設定以降の測定時に関連付けされるグループ番号として使用します。

設定しない場合は 0 を入力してください。

CM-MISDK(C#版) リファレンスマニュアル**GetActiveGroup : アクティブグループ番号を取得します****形式 :**[ReturnMessage](#) GetActiveGroup(out List<Int32> numGroup, Int32 comPort = 0)**引数 :**

名称	I/O	説明
numGroup	O	グループ番号(0~50)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

アクティブグループ番号を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetGroupName : グループ名を設定します****形式 :**[ReturnMessage](#) SetGroupName(Int32 numGroup, string name, Int32 comPort = 0)**引数 :**

名称	I/O	説明
numGroup	I	グループ番号(1~50)
name	I	グループ名
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

指定した番号のグループ名を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetGroupName : グループ名を取得します****形式 :**[ReturnMessage](#) GetGroupName(Int32 numGroup, out string name, Int32 comPort = 0)**引数 :**

名称	I/O	説明
numGroup	I	グループ番号(1~50)
name	O	グループ名
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

指定した番号のグループ名を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetMultipleGroupName : グループ名を複数まとめて設定します****形式 :**[ReturnMessage](#) SetMultipleGroupName(Int32 groupList, List<string> name, Int32 comPort = 0)**引数 :**

名称	I/O	説明	
groupList	I	グループリスト (0~4)	
		0	グループ番号 1~10
		1	グループ番号 11~20
		2	グループ番号 21~30
		3	グループ番号 31~40
		4	グループ番号 41~50
name	I	グループ名(1~10 個)	
comport	I	仮想 COM ポート番号	

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

この API はグループ名をまとめて設定できます。(グループリスト 1 つに設定できるグループ名は 10 個まで)

CM-MISDK(C#版) リファレンスマニュアル**GetMultipleGroupName : グループ名を複数まとめて取得します****形式 :**[ReturnMessage](#) GetMultipleGroupName(Int32 groupList, out List<string> name, Int32 comPort = 0)**引数 :**

名称	I/O	説明	
groupList	I	グループリスト (0~4)	
		0	グループ番号 1~10
		1	グループ番号 11~20
		2	グループ番号 21~30
		3	グループ番号 31~40
		4	グループ番号 41~50
name	O	グループ名(1~10 個)	
comport	I	仮想 COM ポート番号	

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	×	○	○	○	○

解説 :

この API はグループリストに登録されたグループ名をまとめて取得できます。

CM-MISDK(C#版) リファレンスマニュアル**SetTolerance : デフォルト限界値を設定します****形式 :**

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

引数 :

名称	I/O	説明
numObsIll	I	視野・光源番号(0~1)
dataForm	I	データ種類
tolerance	I	限界値データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

デフォルト限界値を設定します。

基準色の新規保存の場合に、この限界値を割り当てます。

※キーで指定された Tolerance 以外は変更されません

CM-MISDK(C#版) リファレンスマニュアル**GetTolerance : デフォルト限界値を取得します****形式 :**

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

引数 :

名称	I/O	説明
numObsIll	I	視野・光源番号(0~1)
dataForm	I	データ種類
tolerance	O	限界値データ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

デフォルト限界値を取得します。

※ToleranceData クラスには取得したい限界値の Key と空の ToleranceParam を入力してください

CM-MISDK(C#版) リファレンスマニュアル**SetParametric : デフォルトパラメトリック係数を設定します****形式 :**

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

引数 :

名称	I/O	説明
dataForm	I	データ種類
Parametric	I	パラメトリック係数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

デフォルトパラメトリック係数を設定します。

基準色の新規保存の場合に、この限界値を割り当てます。

CM-MISDK(C#版) リファレンスマニュアル**GetParametric : デフォルトパラメトリック係数を取得します****形式 :**

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

引数 :

名称	I/O	説明
dataForm	I	データ種類
parametric	O	パラメトリック係数
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

デフォルトパラメトリック係数を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetWarningLevel : 警告レベルを設定します****形式 :**[ReturnMessage](#) SetWarningLevel(Int32 level, Int32 comPort = 0)**引数 :**

名称	I/O	説明
level	I	警告レベル(0~100%)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

警告レベルを設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetWarningLevel : 警告レベルを取得します****形式 :**[ReturnMessage](#) GetWarningLevel(out Int32 level, Int32 comPort = 0)**引数 :**

名称	I/O	説明
level	I	警告レベル(0~100%)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

警告レベルを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetInstrumentMode : 機器モードを設定します****形式 :**[ReturnMessage](#) SetInstrumentMode([InstrumentMode](#) mode, Int32 comPort = 0)**引数 :**

名称	I/O	説明
mode	I	機器モード
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	○

解説 :

機器モードを設定します。

機器の電源を OFF すると、設定はノーマルとなります。

CM-MISDK(C#版) リファレンスマニュアル**GetInstrumentMode : 機器モードを取得します****形式 :**

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

引数 :

名称	I/O	説明
mode	O	機器モード
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	○

解説 :

機器モードを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetUserType : ユーザータイプを設定します****形式 :**[ReturnMessage](#) SetUserType([UserType](#) userType, Int32 comPort = 0)**引数 :**

名称	I/O	説明
userType	I	ユーザータイプ
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

ユーザータイプを設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetUserType : ユーザータイプを取得します****形式 :**[ReturnMessage](#) GetUserType(out [UserType](#) userType, Int32 comPort = 0)**引数 :**

名称	I/O	説明
userType	O	ユーザータイプ
comPort	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

ユーザータイプを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetAdminPassword : 管理者パスワードを設定します****形式 :**[ReturnMessage](#) SetAdminPassword(string adminPass, Int32 comPort = 0)**引数 :**

名称	I/O	説明
adminPass	I	管理者パスワード(数字のみ 4~8 桁)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	○

解説 :

管理者パスワードを設定します。

パスワードは、単体使用時に作業員から管理者へ設定を変更する場合に使用します。

CM-MISDK(C#版) リファレンスマニュアル**GetAdminPassword : 管理者パスワードを取得します****形式 :**[ReturnMessage](#) GetAdminPassword(out string adminPass, Int32 comPort = 0)**引数 :**

名称	I/O	説明
adminPass	O	管理者パスワード(数字のみ 4~8 桁)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	○

解説 :

管理者パスワードを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetAutoPrint : 自動印刷を設定します****形式 :**

[ReturnMessage](#) SetAutoPrint([OnOff](#) autoPrint, Int32 comPort = 0)

引数 :

名称	I/O	説明
autoPrint	I	自動印刷設定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

自動印刷を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetAutoPrint : 自動印刷を取得します****形式 :**[ReturnMessage](#) GetAutoPrint(out [OnOff](#) autoPrint, Int32 comPort = 0)**引数 :**

名称	I/O	説明
autoPrint	O	自動印刷設定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

自動印刷を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetBrightness : 画面の明るさを設定します****形式 :**

[ReturnMessage](#) SetBrightness(Int32 brightness, Int32 comPort = 0)

引数 :

名称	I/O	説明
brightness	I	画面明るさ(0~4)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

画面の明るさを設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetBrightness : 画面の明るさを取得します****形式 :**

[ReturnMessage](#) GetBrightness(out Int32 brightness, Int32 comPort = 0)

引数 :

名称	I/O	説明
brightness	O	画面明るさ(0~4)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

画面の明るさを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetScreenDirection : 画面の表示向きを設定します****形式 :**

[ReturnMessage](#) SetScreenDirection([ScreenDirection](#) direction, Int32 comPort = 0)

引数 :

名称	I/O	説明
direction	I	画面の表示向き
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	○	○	○	○	○

解説 :

画面の表示向きを設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetScreenDirection : 画面の表示向きを取得します****形式 :**

[ReturnMessage](#) GetScreenDirection(out [ScreenDirection](#) direction, Int32 comPort = 0)

引数 :

名称	I/O	説明
direction	O	画面の表示向き
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	○	○	○	○	○

解説 :

画面の表示向きを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetSound : ビープ音を設定します****形式 :**

[ReturnMessage](#) SetSound([OnOff](#) sound, Int32 comPort = 0)

引数 :

名称	I/O	説明
sound	I	音設定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

ビープ音を設定します。

ON の場合は、校正・測定の完了時、エラー時に音を鳴らします。

CM-MISDK(C#版) リファレンスマニュアル**GetSound : ビープ音を取得します****形式 :**

[ReturnMessage](#) GetSound(out [OnOff](#) sound, Int32 comPort = 0)

引数 :

名称	I/O	説明
sound	O	音設定
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

ビープ音を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetCalibrationInterval : 校正間隔を設定します****形式 :**[ReturnMessage](#) SetCalibrationInterval(Int32 calInterval, Int32 comPort = 0)**引数 :**

名称	I/O	説明
calInterval	I	校正間隔 1~24(1h ピッチ)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

校正間隔を設定します。

校正間隔を経過すると、測定可能ですが、校正を推奨する警告状態となります。

CM-MISDK(C#版) リファレンスマニュアル**GetCalibrationInterval : 校正間隔を取得します****形式 :**

[ReturnMessage](#) GetCalibrationInterval(out Int32 calInterval, Int32 comPort = 0)

引数 :

名称	I/O	説明
calInterval	O	校正間隔 1~24(1h ピッチ)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

校正間隔を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetAnnualCalibration : 定期校正喚起を設定します****形式 :**[ReturnMessage](#) SetAnnualCalibration([OnOff](#) annualCal, Int32 comPort = 0)**引数 :**

名称	I/O	説明
annualCal	I	定期校正喚起
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

定期校正喚起を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetAnnualCalibration : 定期校正喚起を取得します****形式 :**

[ReturnMessage](#) GetAnnualCalibration(out [OnOff](#) annualCal, Int32 comPort = 0)

引数 :

名称	I/O	説明
annualCal	O	定期校正喚起
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

定期校正喚起を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetZeroCalibrationSkip : ゼロ校正スキップ可否を設定します****形式 :**[ReturnMessage](#) SetZeroCalibrationSkip([OnOff](#) skip, Int32 comPort = 0)**引数 :**

名称	I/O	説明
skip	I	ゼロ校正スキップ可否
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	○

解説 :

ゼロ校正スキップ可否を設定します。

ON の場合は機器起動時にゼロ校正をスキップできますが、OFF の場合は必ずゼロ校正を行う必要があります。

CM-MISDK(C#版) リファレンスマニュアル**GetZeroCalibrationSkip : ゼロ校正スキップ可否を取得します****形式 :**[ReturnMessage](#) GetZeroCalibrationSkip(out [OnOff](#) skip, Int32 comPort = 0)**引数 :**

名称	I/O	説明
skip	O	ゼロ校正スキップ可否
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	○

解説 :

ゼロ校正スキップ可否を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetDateTime : 日時を設定します****形式 :**[ReturnMessage](#) SetDateTime(DateTime date, Int32 comPort = 0)**引数 :**

名称	I/O	説明
date	I	日時
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

日時を設定します。

CM-MISDK(C#版) リファレンスマニュアル**SetDateFormat : 日付書式を設定します****形式 :**

[ReturnMessage](#) SetDateFormat([DateFormat](#) dateFormat, Int32 comPort = 0)

引数 :

名称	I/O	説明
dateFormat	I	日付書式
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

日付書式を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetDateFormat : 日付書式を取得します****形式 :**[ReturnMessage](#) GetDateFormat(out [DateFormat](#) dateFormat, Int32 comPort = 0)**引数 :**

名称	I/O	説明
dateFormat	O	日付書式
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

日付書式を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetLanguage : 表示言語を設定します****形式 :**

[ReturnMessage](#) SetLanguage([Language](#) language, Int32 comPort = 0)

引数 :

名称	I/O	説明
language	I	表示言語
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

表示言語を設定します。

CM-MISDK(C#版) リファレンスマニュアル**GetLanguage : 表示言語を取得します****形式 :**

[ReturnMessage](#) GetLanguage(out [Language](#) language, Int32 comPort = 0)

引数 :

名称	I/O	説明
language	O	表示言語
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

表示言語を取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetAutoPowerOff : オートパワーオフが働く時間を設定します****形式 :**[ReturnMessage](#) SetAutoPowerOff(Int32 time, Int32 comPort = 0)**引数 :**

名称	I/O	説明
time	I	0～60 分
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
△	△	○	○	○	○

※[機種・バージョン](#)依存します**解説 :**

オートパワーオフが働く時間を設定します。

0 を設定した場合はオートパワーオフ機能を Off にします。

CM-MISDK(C#版) リファレンスマニュアル**GetAutoPowerOff : オートパワーオフが働く時間を取得します****形式 :**[ReturnMessage](#) GetAutoPowerOff(out Int32 time, Int32 comPort = 0)**引数 :**

名称	I/O	説明
time	O	0~60 分
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
△	△	○	○	○	○

※[機種・バージョン](#)依存します**解説 :**

機器に設定されているオートパワーオフが働く時間を取得します。

CM-MISDK(C#版) リファレンスマニュアル**ClearJobInfo : ジョブ情報をクリアします****形式 :**

[ReturnMessage](#) ClearJobInfo(Int32 jobNum, Int32 comPort = 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブ情報をクリアします。

指定したジョブ番号に登録されているステップ情報、画像情報をクリアします。

CM-MISDK(C#版) リファレンスマニュアル**SetJobInfo : ジョブ情報を設定します****形式 :**

[ReturnMessage](#) SetJobInfo(Int32 jobNum, [JobInfo](#) info, Int32 comPort = 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
info	I	ジョブ情報
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブ情報を設定します。

ここで指定した数分のステップを [SetJobStepForOperation](#) と [SetJobStepForResult](#) を使用して設定してください。

手順は、「[3.2 基本的なフロー](#)」を参照してください。

機器に登録後は、適切に設定されているか試行した上でジョブを運用してください。

CM-MISDK(C#版) リファレンスマニュアル**GetJobInfo : ジョブ情報を取得します****形式 :**

[ReturnMessage](#) GetJobInfo(Int32 jobNum, out [JobInfo](#) info, Int32 comPort = 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
info	O	ジョブ情報
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブ情報を取得します。

CM-MISDK(C#版) リファレンスマニュアル**GetJobStepType : ジョブのステップ種類を取得します****形式 :**

[ReturnMessage](#) GetJobStepType(Int32 jobNum, Int32 stepNum, out [JobStepType](#) stepType, Int32 comPort
= 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
stepNum	I	ステップ番号(0~19)
stepType	O	ステップ種類
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブのステップ種類を取得します。

ジョブの種類が操作ステップの場合は [GetJobStepForOperation](#) を、結果ステップの場合は [GetJobStepForResult](#) を使用してください。

CM-MISDK(C#版) リファレンスマニュアル**SetJobStepForOperation : ジョブの操作ステップを設定します****形式 :**

[ReturnMessage](#) SetJobStepForOperation(Int32 jobNum, Int32 stepNum, [JobStepOperation](#) jobStep, Int32 comPort = 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
stepNum	I	ステップ番号(0~19)
jobStep	I	ステップ内容(操作ステップ)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブの操作ステップを設定します。

[SetJobInfo](#) で指定したステップ数分のステップ(操作ステップまたは結果ステップ)を設定する必要があります。

手順は、「[3.2 基本的なフロー](#)」を参照してください。

各機種で設定が必要な項目は以下の通りです。「-」のところは入力を見捨てます。

	26dG	26d	25d
meas_type	-	-	-
meas_mode	○	-	-
meas_area	○	○	-
meas_angle	-	-	-
meas_ldirection	-	-	-
meas_scie	○	○	○

CM-MISDK(C#版) リファレンスマニュアル

meas_uv	○	○	-
---------	---	---	---

CM-MISDK(C#版) リファレンスマニュアル**GetJobStepForOperation : ジョブの操作ステップを取得します****形式 :**

[ReturnMessage](#) GetJobStepForOperation(Int32 jobNum, Int32 stepNum, out [JobStepOperation](#) jobStep, Int32 comPort = 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
stepNum	I	ステップ番号(0~19)
jobStep	O	ステップ内容(操作ステップ)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブの操作ステップを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetJobStepForResult : ジョブの結果ステップを設定します****形式 :**

[ReturnMessage](#) SetJobStepForResult(Int32 jobNum, Int32 stepNum, [JobStepResult](#) jobStep, Int32 comPort = 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
stepNum	I	ステップ番号(0~19)
jobStep	I	ステップ内容(結果ステップ)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブの結果ステップを設定します。

[SetJobInfo](#) で指定したステップ数分のステップ(操作ステップまたは結果ステップ)を設定する必要があります。

手順は、「[3.2 基本的なフロー](#)」を参照してください。

CM-MISDK(C#版) リファレンスマニュアル**GetJobStepForResult : ジョブの結果ステップを取得します****形式 :**

[ReturnMessage](#) GetJobStepForResult(Int32 jobNum, Int32 stepNum, out [JobStepResult](#) jobStep, Int32 comPort = 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
stepNum	I	ステップ番号(0~19)
jobStep	O	ステップ内容(結果ステップ)
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブの結果ステップを取得します。

CM-MISDK(C#版) リファレンスマニュアル**SetJobImage : ジョブの画像を設定します****形式 :**[ReturnMessage](#) SetJobImage(Int32 jobNum, Int32 imageNum, [JobImage](#) jobImage, Int32 comPort = 0)**引数 :**

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
imageNum	I	画像番号(0~9)
jobImage	I	画像
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブの画像を設定します。

画像を設定するためには、[SetJobInfo](#) でジョブ情報を設定する必要があります。手順は、「[3.2 基本的なフロー](#)」を参照してください。

この API では機器のフラッシュメモリに直接書き込みを行います。

画像は 1 つのジョブに、10 コまで設定できます。

各機種で登録可能な画像解像度は以下の通りです。左上からサイズ分のデータを左詰めで登録してください。

	26dG	26d	25d
解像度	W : 240 H : 128	W : 240 H : 128	W : 240 H : 128

CM-MISDK(C#版) リファレンスマニュアル**GetJobImage : ジョブの画像を取得します****形式 :**

[ReturnMessage](#) GetJobImage(Int32 jobNum, Int32 imageNum, out [JobImage](#) jobImage, Int32 comPort = 0)

引数 :

名称	I/O	説明
jobNum	I	ジョブ番号(0~4)
imageNum	I	画像番号(0~9)
jobImage	O	画像
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	処理が正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErInvalidParameter	指定したパラメータが正しくありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	測定器との接続に失敗しました。または Connect をされていません

※上記以外は、[エラー一覧](#)を参照してください。

対応機種 :

25cG	M6	26dG	26d	25d	23d
×	×	○	○	○	×

解説 :

ジョブの画像を取得します。

CM-MISDK(C#版) リファレンスマニュアル**ResetToFactorySetting : 工場出荷状態に戻します****形式 :**[ReturnMessage](#) ResetToFactorySetting(Int32 comPort = 0)**引数 :**

名称	I/O	説明
comport	I	仮想 COM ポート番号

戻り値 :

定義値	説明
KmSuccess	正常に完了しました
KmErNoConnect	指定した仮想 COM ポート番号に接続している測定器がありません
KmErCannotCommand	指定された API は現在の機種では対応していません
KmErConnectFailed	Connect を実行してください

※上記以外は、[エラー一覧](#)を参照してください。**対応機種 :**

25cG	M6	26dG	26d	25d	23d
○	○	○	○	○	○

解説 :

工場出荷状態に戻します。

測定値、基準色を全削除し、設定値を初期状態にします。

CM-MISDK(C#版) リファレンスマニュアル**5. 定義・構造体****5.1 型定義**

定義	Byte	内容	C/C++	VB.NET	VBA(VB6)
int8_km	1	符号付バイト型	char	SByte	(Byte)
uint8_km	1	バイト型	unsigned char	Byte	Byte
int16_km	2	短整数型	short	Short	Integer
uint16_km	2	短整数型(符号なし)	unsigned short	UShort	(Integer)
int32_km	4	整数型	long(int)	Integer	Long
uint32_km	4	整数型(符号なし)	unsigned long	UInteger	(Long)
int64_km	8	長整数型	long long	Long	
uint64_km	8	長整数型(符号なし)	unsigned long long	ULong	
float32_km	4	単精度浮動小数点型	float	Single	Single
float64_km	8	倍精度浮動小数点型	double	Double	Double

CM-MISDK(C#版) リファレンスマニュアル5.2 クラス定義**Class DataForm(データフォームクラス)**

形式 :

```
class DataForm
{
    DataType        DataType;
    IrradiationDirection IrradiationDirection;
}
```

変数 :

変数	説明
DataType	データ種類
IrradiationDirection	照射方向

Class SpecData(反射率クラス)

形式 :

```
class SpecData
{
    List<double> Spec;
    double Gloss;
}
```

変数 :

変数	説明
Spec	反射率 ■ CM-25CG/CM-26dG/CM-26d の場合 <div>39 コ(360~740nm/10nm ピッチ)</div> ■ CM-M6/CM-25d/CM-23d の場合 <div>31 コ(400~700nm/10nm ピッチ)</div>
Gloss	光沢値

Class MeasDataColor(測定データ色彩値クラス)

形式 :

```
class MeasDataColor
```

CM-MISDK(C#版) リファレンスマニュアル

```
{  
    ColorSpace                color;  
    Dictionary<MeasDataType, List<double>> data;  
}
```

変数：

変数	説明
color	色彩値を算出する表色系 ※COLOR_MUNSELL_C 以外を指定できます
data	色彩値データ

Class ColorCalData(測色用校正データクラス)

形式：

```
class ColorCalData  
{  
    List<double>    Data;  
    Int32           PlateId;  
}
```

変数：

変数	説明
Data	校正データ ■ CM-25cG/CM-26dG/CM-26d の場合 39 コ(360~740nm/10nm ピッチ) ■ CM-M6/CM-25d/CM-23d の場合 31 コ(400~700nm/10nm ピッチ)
PlateId	校正板 ID

Class GlossCalData(光沢用校正データクラス)

形式：

```
class GlossCalData  
{  
    double          Data;  
    Int32           PlateId;  
}
```


CM-MISDK(C#版) リファレンスマニュアル**変数 :**

変数	説明
Data	校正データ
PlateId	校正板 ID

Class UserCalData(ユーザー校正データクラス)**形式 :**

```
class UserCalData
{
    List<double>    Data;
    Int            PlateId;
}
```

変数 :

変数	説明
Data	校正データ <div> <div>■ CM-25cG/CM-26dG/CM-26d の場合</div> <div>39 コ(360~740nm/10nm ピッチ)</div> </div> <div> <div>■ CM-25d/CM-23d の場合</div> <div>31 コ(400~700nm/10nm ピッチ)</div> </div>
PlateId	校正板 ID

Class UserCalData2(ユーザー校正データクラス)**形式 :**

```
class UserCalData
{
    List<double>    Data;
    string          PlateId;
}
```

変数 :

変数	説明
Data	校正データ <div>■ CM-25cG/CM-26dG/CM-26d の場合</div>

CM-MISDK(C#版) リファレンスマニュアル

	<div>39 コ(360~740nm/10nm ピッチ)</div> <div>■ CM-25d/CM-23d の場合</div> <div>31 コ(400~700nm/10nm ピッチ)</div>
PlateId	校正板 ID(8 文字) ※ユーザーID を空白文字にすると、データなしと同じ扱いとなります。

Class UvAdjustIndex(蛍光調整用のインデックスデータ)

形式 :

```
class UvAdjustIndex
{
    double    value;
    double    tolerance;
}
```

変数 :

変数	説明
value	値
tolerance	許容幅

Class UvAdjustCoef(蛍光調整係数)

形式 :

```
class UvAdjustCoef
{
    List<double>    coefficient;
    List<double>    correction;
    List<double>    param;
    DateTime        date;
}
```

変数 :

変数	説明
coefficient	蛍光係数 ※ GetInstrumentInfo で取得した DataSize の個数
correction	蛍光補正值 ※ GetInstrumentInfo で取得した DataSize の個数
param	パラメータ

CM-MISDK(C#版) リファレンスマニュアル

	※Ganz & Griesser の場合のみで、P・Q・C・m・n・k・dWdS の順番で格納されます
date	蛍光調整日時

Class UvAdjustGG(Ganz&Griesser 蛍光調整データ)**形式 :**

```

class UvAdjustGG
{
    Int32          count;
    List<double>   WI;
    List<double>   Tint;
}

```

変数 :

変数	説明
count	使用するサンプル数(4 または 5)
WI	使用するサンプル数分の WI 値
Tint	使用するサンプル数分の Tint 値

Class MeasCondSMC(SMC 条件クラス)**形式 :**

```

class MeasCondSMC
{
    OnOff          enable;
    Int32           times;
    double          threshold;
}

```

変数 :

変数	説明
enable	機能の有効・無効
times	平均回数(3~10 回)
threshold	閾値レベル(0.01~9.99)

CM-MISDK(C#版) リファレンスマニュアル**Class ColorValue(各表色系の結果にアクセスするクラス)****形式：**

```

class ColorValue
{
    string                Id;
    ColorSpace           ColorModeId;
    DateTime              Date;
    Dictionary<string, double> Value;
    Dictionary<string, string> CharValue;
    List<Dictionary<string, double>> AllAngleValue;
}

```

変数：

変数	説明
Id	ID
ColorModeId	表色系 ID ※COLOR_MUNSELL_C 以外を指定できます
Date	登録・測定日時
Value	表色値
CharValue	※未使用
AllAngleValue	表色値（複数の角度のデータを取得する際に使用してください。）

Class TargetData(基準色データクラス)**形式：**

```

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;
}

```

CM-MISDK(C#版) リファレンスマニュアル

変数 :

変数	説明
DateTime	測定(登録)日時
Group	グループ番号
Name	名前
Data	反射率
Gloss	光沢値
ColorData	表色値
Mode	測定モード
Area	測定経
dataForm	データ種類

Class TargetDataPack(基準色データクラス)

形式 :

```

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;
}

```

変数 :

変数	説明
----	----

CM-MISDK(C#版) リファレンスマニュアル

date	測定(登録)日時
group_list	グループ番号リスト
meas_type	測定タイプ
meas_mode	測定モード
meas_area	測定径 ※25d、23d のデータの場合は、AREA_MAV をを設定してください
meas_angle	測定角度
l_direction	照明方向
meas_scie	正反射光処理
meas_uv	UV 条件 ※25d、23d のデータの場合は、UV_CUT400 を設定してください
warning	警告
diagnosis	診断
data_attr	データ属性
data	反射率データ ※data_attr が DATAATTR_SPEC の場合のみ使用します ※測定モード、測定角度、照明方向、正反射光処理によって必要となるデータ数が異なります。ご注意ください。
data_color	色彩値データ ※data_attr が DATAATTR_SPEC 以外の場合に使用します ※光沢値は data_color.data1、data_color.data2 の両方に指定してください ※測定モード、測定角度、照明方向、正反射光処理によって必要となるデータ数が異なります。ご注意ください。
name	名前

Class ColorData(色彩値データクラス)

形式:

```

class ColorData
{
    Observer                obs1;
    Observer                obs2;
    Illuminant              ill1;
    Illuminant              ill2;
    Dictionary<DataId, List<double>> data1;
    Dictionary<DataId, List<double>> data2;

```

CM-MISDK(C#版) リファレンスマニュアル

}

変数 :

変数	説明
obs1	観察視野 1
obs2	観察視野 2
ill1	観察光源 1
ill2	観察光源 2
data1	観察視野 1・光源 1 の色彩値データ
data2	観察視野 2・光源 2 の色彩値データ

Class ToleranceData(限界値データクラス)

形式 :

```
class ToleranceData
{
    Dictionary<Int32, ToleranceParam> Tolerance;
}
```

変数 :

変数	説明								
Tolerance	限界値データ Key 限界値番号(ToleranceId を参照) ToleranceParam <table> <tr> <td>Upper_enable</td><td>上限有効・無効</td></tr> <tr> <td>Upper_value</td><td>上限値</td></tr> <tr> <td>Lower_enable</td><td>下限有効・無効</td></tr> <tr> <td>Lower_value</td><td>下限値</td></tr> </table>	Upper_enable	上限有効・無効	Upper_value	上限値	Lower_enable	下限有効・無効	Lower_value	下限値
Upper_enable	上限有効・無効								
Upper_value	上限値								
Lower_enable	下限有効・無効								
Lower_value	下限値								

※機器により設定可能なパラメータの種類、範囲が異なる場合があります。機器本体のマニュアルを参照ください。

※ Δx 、 Δy には 100 倍した値を入力してください。取得した Δx 、 Δy の値は 100 倍されています。

Class ToleranceParam(限界値データクラス)

形式 :

```
class ToleranceData
{
```

CM-MISDK(C#版) リファレンスマニュアル

```

byte           Upper_enable;
double         Upper_value;
byte           Lower_enable;
double         Lower_value;
}

```

変数：

変数	説明
Upper_enable	上限有効・無効(0：無効、1：有効)
Upper_value	上限値
Lower_enable	下限有効・無効(0：無効、1：有効)
Lower_value	下限値

※機器により設定可能なパラメータの種類、範囲がことなる場合があります。機器本体のマニュアルを参照ください。

Class ParamtricCoef(パラメトリック係数クラス)**形式：**

```

class ParametricCoef
{
    List<double>  CMC;
    List<double>  DeltaE94;
    List<double>  DeltaE00;
}

```

変数：

変数	説明
CMC	CMC 用 ※係数は l、c の順番となります
DeltaE94	ΔE^*94 用 ※係数は l、c、h の順番となります
DeltaE00	$\Delta E00$ 用 ※係数は l、c、h の順番となります

※機器により設定可能なパラメータの種類、範囲が異なる場合があります。機器本体のマニュアルを参照ください。

Class SampleData(測定値データクラス)**形式：**

```

class SampleData

```


CM-MISDK(C#版) リファレンスマニュアル

```

{
    DateTime                                Date;
    string                                  Name
    SpecData                                Data;
    Dictionary< IrradiationDirection, List<List<double>>> AllAngleData;
    ColorValue                              ColorData;
    MeasCondMode                            Mode;
    MeasArea                                Area;
    DataForm                                DataForm;
    Int32                                  relation_target;
}

```

変数：

変数	説明
Date	測定(登録)日時
Name	名前
Data	反射率、光沢値
AllAngleData	反射率(複数の角度のデータを取得する際に使用してください。)
ColorData	表色値
Mode	測定モード
Area	測定経
DataForm	データ種類
relation_target	関連基準色

Class SampleDataPack(測定値データクラス)

形式：

```

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;
}

```

CM-MISDK(C#版) リファレンスマニュアル

```

        Int32                diagnosis;
        Int32                relation_target;
        Dictionary<DataId, List<double>> data;
        string               name;
    }

```

変数 :

変数	説明
date	測定(登録)日時
meas_type	測定タイプ
meas_mode	測定モード
meas_area	測定径
meas_angle	測定角度
l_direction	照明方向
meas_scie	正反射光処理
meas_uv	UV 条件
warning	警告
diagnosis	診断
relation_target	関連基準色番号
data	データ ※測定モード、測定角度、照明方向、正反射光処理によって必要となるデータ数が異なります。ご注意ください。
name	名前

Class InstrumentInfo(機器情報クラス)**形式 :**

```

class InstrumentInfo
{
    string    Name;
    Int32    SerialNum;
    string    Version;
}

```

変数 :

変数	説明
Name	機器名
SerialNum	本体番号

CM-MISDK(C#版) リファレンスマニュアル

Version	バージョン
---------	-------

Class InstrumentInfoEx(機器情報クラス)

形式 :

```
class InstrumentInfoEx
{
    Int32    SerialNum;
    Int32    WaveLengthStart;
    Int32    WaveLengthEnd;
    Int32    WaveLengthPitch;
    Int32    DataSize;
    string    Version;
    string    Name;
}
```

変数 :

変数	説明
SerialNum	本体番号
WaveLengthStart	反射率 波長範囲 : 最短波長
WaveLengthEnd	反射率 波長範囲 : 最長波長
WaveLengthPitch	反射率 波長範囲 : 波長ピッチ
DataSize	反射率 データ数(最短波長、最長波長、波長ピッチで決まります)
Version	バージョン
Name	機器名

Class JobInfo(ジョブ情報)

形式 :

```
class JobInfo
{
    Int32    step_count;
    OnOff    step_loop;
    string    name;
}
```

変数 :

変数	説明
----	----

CM-MISDK(C#版) リファレンスマニュアル

step_count	ジョブに登録するステップ数(1~20)
step_loop	ジョブの繰り返し可否
name	名前

Class JobStepOperation(ジョブ操作ステップ)

形式：

```

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;
}

```

変数：

変数	説明
image_num	画像番号(0~10) ※10 は「画像なし」と扱います
meas_type	測定タイプ
meas_mode	測定モード
meas_area	測定径
meas_angle	測定角度
meas_ldirection	照射方向
meas_scie	正反射光処理

CM-MISDK(C#版) リファレンスマニュアル

meas_uv	UV 条件
auto_ave_times	自動平均回数(1~10)
manu_ave_times	手動平均回数(1~30)
relation_target	関連基準色番号 ※オパシティ属性のデータは設定できません
enable_meas	測定ボタン使用可否
eable_prev	戻るボタン表示可否
enable_next	進むボタン表示可否
enable_end	終了ボタン使用可否
name	名前
comment	コメント

Class JobStepResult(ジョブ結果ステップ)**形式 :**

```

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;
}

```

変数 :

変数	説明
meas_scie	正反射光処理
obs1	観察視野 1
obs2	観察視野 2
ill1	観察光源 1
ill2	観察光源 2
custom_item	カスタム項目

CM-MISDK(C#版) リファレンスマニュアル

enable_meas	測定ボタン使用可否
eable_prev	戻るボタン表示可否
enable_next	進むボタン表示可否
enable_end	終了ボタン使用可否

Class JobImage(ジョブ画像)**形式：**

```
class JobImage
{
    Int32      width;
    Int32      height;
    List<Int32> data;
    string     name;
}
```

変数：

変数	説明
width	幅(240 固定)
height	高さ(128 固定)
data	画像データ(左上から Z 順序で配置されます) データは 1 要素当たり 1 ピクセルで、RGB それぞれ 8 ビットとし、右詰で BGR の順序で格納します
name	画像名

CM-MISDK(C#版) リファレンスマニュアル5.3 値定義**CalStatus(校正状態)**

	値	説明
StatusZero	0	ゼロ校正が必要な状態
StatusWhite	1	白色校正が必要な状態
StatusGloss	2	光沢校正が必要な状態
StatusMeasure	3	測定可能な状態
StatusMeasureWrn	4	測定可能な状態(校正推奨)
StatusUser	5	ユーザー校正が必要な状態

※[機種・バージョン](#)依存します

CalDataType(校正用データ種類)

■CM-25cG の場合

	値	説明
CALTYPE_MAV	0	MAV
CALTYPE_SAV	1	SAV

■CM-26dG/CM-26d の場合

	値	説明
CALTYPE_MAV_SCI	0	MAV・SCI
CALTYPE_MAV_SCE	1	MAV・SCE
CALTYPE_SAV_SCI	2	SAV・SCI
CALTYPE_SAV_SCE	3	SAV・SCE

■CM-25d/CM-23d の場合

	値	説明
CALTYPE_MAV_SCI	0	MAV・SCI
CALTYPE_MAV_SCE	1	MAV・SCE

■CM-M6 の場合

	値	説明
CALTYPE_L_ANGLE_M15	0	左 -15°
CALTYPE_L_ANGLE_15	1	左 15°
CALTYPE_L_ANGLE_25	2	左 25°
CALTYPE_L_ANGLE_45	3	左 45°
CALTYPE_L_ANGLE_75	4	左 75°

CM-MISDK(C#版) リファレンスマニュアル

CALTYPE_L_ANGLE_110	5	左 110°
CALTYPE_R_ANGLE_M15	6	右 -15°
CALTYPE_R_ANGLE_15	7	右 15°
CALTYPE_R_ANGLE_25	8	右 25°
CALTYPE_R_ANGLE_45	9	右 45°
CALTYPE_R_ANGLE_75	10	右 75°
CALTYPE_R_ANGLE_110	11	右 110°

MeasStatus(測定状態)

	値	説明
Idling	0	完了(待機)状態
Measuring	1	測定中

DataType(データ種類)

■CM-25cG の場合

	値	説明
NO_PARAM	0xFF	NO_PARAMETER

■CM-M6 の場合

	値	説明
ANGLE_M15	3	-15°
ANGLE_15	4	15°
ANGLE_25	5	25°
ANGLE_45	6	45°
ANGLE_75	7	75°
ANGLE_110	8	110°

■CM-26dG/CM-26d の場合

	値	説明
SC_SCI	0	SCI(UV 条件 : 出力数が 1 コの場合のみ)
SC_SCE	1	SCE(UV 条件 : 出力数が 1 コの場合のみ)
SC_SCI_UVFULL	2	SCI(UV100%)
SC_SCE_UVFULL	3	SCE(UV100%)
SC_SCI_UVCUT	4	SCI(UV カット)
SC_SCE_UVCUT	5	SCE(UV カット)
SC_SCI_UVADJ	6	SCI(UV 調整)

CM-MISDK(C#版) リファレンスマニュアル

SC_SCE_UVADJ	7	SCE(UV 調整)
DATATYPE_BACKWHITE	10	オパシティ白バック
DATATYPE_BACKBLACK	11	オパシティ黒バック

■CM-25d/CM-23d の場合

	値	説明
SC_SCI	0	SCI
SC_SCE	1	SCE
DATATYPE_BACKWHITE	10	オパシティ白バック
DATATYPE_BACKBLACK	11	オパシティ黒バック

※個数記載がないものは、[GetInstrumentInfo](#) で取得した DataSize の個数

MeasDataType(測定データ種類)

■CM-25cG の場合

	値	説明
DTYPE_GLOSS	0	GU(1 コ)
DTYPE_SPEC	1	分光データ

■CM-26dG の場合

	値	説明
DTYPE_GLOSS	0	GU(1 コ)
DTYPE_SCI	1	SCI(UV 条件 : 出力数が 1 コの場合のみ)
DTYPE_SCE	2	SCE(UV 条件 : 出力数が 1 コの場合のみ)
DTYPE_SCI_UVFULL	10	SCI(UV100%)
DTYPE_SCE_UVFULL	11	SCE(UV100%)
DTYPE_SCI_UVCUT	12	SCI(UV カット)
DTYPE_SCE_UVCUT	13	SCE(UV カット)
DTYPE_SCI_UVADJ	14	SCI(UV 調整)
DTYPE_SCE_UVADJ	15	SCE(UV 調整)

■CM-26d の場合

	値	説明
DTYPE_SCI	1	SCI(UV 条件 : 出力数が 1 コの場合のみ)
DTYPE_SCE	2	SCE(UV 条件 : 出力数が 1 コの場合のみ)
DTYPE_SCI_UVFULL	10	SCI(UV100%)
DTYPE_SCE_UVFULL	11	SCE(UV100%)
DTYPE_SCI_UVCUT	12	SCI(UV カット)
DTYPE_SCE_UVCUT	13	SCE(UV カット)

CM-MISDK(C#版) リファレンスマニュアル

DTYPE_SCI_UVADJ	14	SCI(UV 調整)
DTYPE_SCE_UVADJ	15	SCE(UV 調整)

■CM-25d/CM-23d の場合

	値	説明
DTYPE_SCI	1	SCI
DTYPE_SCE	2	SCE

■CM-M6 の場合

	値	説明
DTYPE_L_ANGLE_M15	0	左 -15°
DTYPE_L_ANGLE_15	1	左 15°
DTYPE_L_ANGLE_25	2	左 25°
DTYPE_L_ANGLE_45	3	左 45°
DTYPE_L_ANGLE_75	4	左 75°
DTYPE_L_ANGLE_110	5	左 110°
DTYPE_R_ANGLE_M15	6	右 -15°
DTYPE_R_ANGLE_15	7	右 15°
DTYPE_R_ANGLE_25	8	右 25°
DTYPE_R_ANGLE_45	9	右 45°
DTYPE_R_ANGLE_75	10	右 75°
DTYPE_R_ANGLE_110	11	右 110°
DTYPE_DP_ANGLE_M15	12	ダブルパス -15°
DTYPE_DP_ANGLE_15	13	ダブルパス 15°
DTYPE_DP_ANGLE_25	14	ダブルパス 25°
DTYPE_DP_ANGLE_45	15	ダブルパス 45°
DTYPE_DP_ANGLE_75	16	ダブルパス 75°
DTYPE_DP_ANGLE_110	17	ダブルパス 110°

※個数記載がないものは、[GetInstrumentInfo](#) で取得した DataSize の個数

IrradiationDirection (照射方向)

■CM-25cG/26dG/26d/25d/23d の場合

	値	説明
NO_PARAM	0xFF	No parameter

■CM-M6 の場合

	値	説明
LEFT_DIRECTION	0	LEFT

CM-MISDK(C#版) リファレンスマニュアル

RIGHT_DIRECTION	1	RIGHT
AVERAGE	2	Double Path

LightDirection (照射方向)

	値	説明
LDIRECTION_NONE	0	なし
LDIRECTION_L	0x01	LEFT
LDIRECTION_R	0x02	RIGHT
LDIRECTION_DP	0x04	Double Path

※[機種・バージョン](#)依存します**CondUvAdjust(蛍光調整条件)**

	値	説明
UVADJ_PROFILE	0	プロファイル
UVADJ_WI	1	WI
UVADJ_TINT	2	Tint
UVADJ_WITINT	3	WI&Tint
UVADJ_BRIGHTNESS	4	ISO ブライツネス
UVADJ_GG	5	Ganz & Griesser
UVADJ_NONE	-1	条件なし

※[機種・バージョン](#)依存します**UvAdjustDataType(蛍光係数用データ種類)**

	値	説明
UVADJ_DATATYPE_SCI	0	SCI
UVADJ_DATATYPE_SCE	1	SCE
UVADJ_DATATYPE_NONE	-1	条件なし

※[機種・バージョン](#)依存します**MeasType(測定タイプ)**

	値	説明
MEASTYPE_REF	0	反射

CM-MISDK(C#版) リファレンスマニュアル

MEASTYPE_TRA	1	透過
MEASTYPE_NONE	-1	条件なし

※[機種・バージョン](#)依存します**MeasArea(測定径)**

	値	説明
AREA_MAV	0	MAV
AREA_SAV	1	SAV
AREA_LAV	2	LAV
AREA_LMAV	3	LMAV
AREA_NONE	-1	径固定

※[機種・バージョン](#)依存します**MeasAngle(測定角度)**

	値	説明
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	条件なし

※[機種・バージョン](#)依存します**MeasCondMode(測定モード)**

	値	説明
MeasModeColorAndGloss	0	測色値と光沢値の取得
MeasModeColorOnly	1	測色値のみ取得
MeasModeGlossOnly	2	光沢値のみ取得
MeasModeOpacity	3	オパシティ

※[機種・バージョン](#)依存します

CM-MISDK(C#版) リファレンスマニュアル**MeasCondScie(正反射光処理)**

	値	説明
SC_SCI	0	SCI
SC_SCE	1	SCE
SC_SCIE	2	SCI+SCE
SC_NONE	-1	条件なし

※[機種・バージョン](#)依存します**MeasCondUv(UV 条件)**

	値	説明
UV_100	0	UV100%
UV_CUT400	1	UV400nm カット
UV_CUT420	2	UV420nm カット
UV_CUT400N	3	UV 調整 400nm カット Normal 発光
UV_CUT400L	4	UV 調整 400nm カット Low 発光
UV_CUT420N	5	UV 調整 420nm カット Normal 発光
UV_CUT420L	6	UV 調整 420nm カット Low 発光
UV_100_CUT400	7	UV100%+UV400nm カット
UV_100_CUT420	8	UV100%+UV420nm カット
UV_100_CUT400N	9	UV100%+UV400nm カット+UV 調整 400nm カット Normal 発光
UV_100_CUT400L	10	UV100%+UV400nm カット+UV 調整 400nm カット Low 発光
UV_100_CUT420N	11	UV100%+UV420nm カット+UV 調整 420nm カット Normal 発光
UV_100_CUT420L	12	UV100%+UV420nm カット+UV 調整 420nm カット Low 発光
UV_NONE	-1	条件なし

※[機種・バージョン](#)依存します**DataId(データ種類)**

■CM-25cG の場合

	値	説明
DATAID_GLOSS	0	GU(1 コ)
DATAID_SPEC	1	分光データ

■CM-26dG の場合

	値	説明
--	---	----

CM-MISDK(C#版) リファレンスマニュアル

DATAID_GLOSS	0	GU(1 コ)
DATAID_SCI	1	SCI
DATAID_SCE	2	SCE
DATAID_BACKWHITE	3	オパシティ白バック
DATAID_BACKBLACK	4	オパシティ黒バック

■CM-26d/CM-25d/CM-23d の場合

	値	説明
DATAID_SCI	1	SCI
DATAID_SCE	2	SCE
DATAID_BACKWHITE	3	オパシティ白バック
DATAID_BACKBLACK	4	オパシティ黒バック

■CM-M6 の場合

	値	説明
DATAID_L_ANGLE_M15	0	左 -15°
DATAID_L_ANGLE_15	1	左 15°
DATAID_L_ANGLE_25	2	左 25°
DATAID_L_ANGLE_45	3	左 45°
DATAID_L_ANGLE_75	4	左 75°
DATAID_L_ANGLE_110	5	左 110°
DATAID_R_ANGLE_M15	6	右 -15°
DATAID_R_ANGLE_15	7	右 15°
DATAID_R_ANGLE_25	8	右 25°
DATAID_R_ANGLE_45	9	右 45°
DATAID_R_ANGLE_75	10	右 75°
DATAID_R_ANGLE_110	11	右 110°
DATAID_DP_ANGLE_M15	12	ダブルパス -15°
DATAID_DP_ANGLE_15	13	ダブルパス 15°
DATAID_DP_ANGLE_25	14	ダブルパス 25°
DATAID_DP_ANGLE_45	15	ダブルパス 45°
DATAID_DP_ANGLE_75	16	ダブルパス 75°
DATAID_DP_ANGLE_110	17	ダブルパス 110°

※個数記載がないものは、[GetInstrumentInfo](#) で取得した DataSize の個数

DataAttr(データ属性)

	値	説明
--	---	----

CM-MISDK(C#版) リファレンスマニュアル

DATAATTR_SPEC	0	分光データ
DATAATTR_LAB	1	L*a*b*
DATAATTR_HLAB	2	ハンターLa b
DATAATTR_XYZ	3	XYZ

DataWarning(データ警告)

	値	説明
KmWrBattery	0x01	電池電圧が低下しています
KmWrCalibration	0x02	校正後の時間変化が大きいのので、再校正が必要です
KmWrPreAnnualCalibraton	0x04	定期校正の時期が近づいています
KmWrAnnualCalibraton	0x08	定期校正が必要です
KmWrLampForColor	0x10	反射率が保証範囲外です
KmWrOutOfColorRange	0x20	色測定用の光源光量が低下しています
KmWrOutOfGlossRange	0x40	光沢値が保証範囲外です
KmWrLampForGloss	0x80	光沢測定用の光源光量が低下しています

SaveMode(保存方法)

	値	説明
SAVEMODE_AUTO	0	自動保存
SAVEMODE_MANUAL	1	手動保存

DisplayType(表示形式)

	値	説明
DISPTYPE_ABS	0x001	絶対値
DISPTYPE_DIF	0x002	色差
DISPTYPE_ABSDIF	0x004	絶対値&色差
DISPTYPE_CUSTOM	0x008	カスタム
DISPTYPE_GRAPH_ABS	0x010	絶対値グラフ
DISPTYPE_GRAPH_DIF	0x020	色差グラフ
DISPTYPE_GRAPH_REF	0x040	分光グラフ
DISPTYPE_PASS_FAIL	0x080	合否判定
DISPTYPE_MI	0x100	MI
DISPTYPE_GRAPH_LINE	0x200	折れ線グラフ

CM-MISDK(C#版) リファレンスマニュアル

DISPTYPE_AUDI2000_EC	0x400	Audi2000 ΔEc
DISPTYPE_AUDI2000_EP	0x800	Audi2000 ΔEp

※[機種・バージョン](#)依存します

Observer(観察視野)

	値	説明
Deg02	0	2°
Deg10	1	10°

Illuminant(観察光源)

	値	説明
ILL_None	0	なし
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	ユーザー

※[機種・バージョン](#)依存します

ColorSpace(表色系)

	値	説明
COLOR_LAB	0	L*a*b*
COLOR_LCH	1	L*C*h
COLOR_HLAB	2	ハンターLa b

CM-MISDK(C#版) リファレンスマニュアル

COLOR_XYX	3	Yxy
COLOR_XYZ	4	XYZ
COLOR_MUNSELL_C	5	マンセル(C)
COLOR_DEFAULT(※)	255	演算しない

※COLOR_DEFAULT は、ColorValue クラスの ColorModeId で使用し、これを使用した場合は色彩値出力しません

※[機種・バージョン](#)依存します

Equation(色差式)

	値	説明
EQUATION_DE1976	0	ΔE^*_{ab}
EQUATION_CMC	1	CMC
EQUATION_DE1994	2	ΔE^*_{94}
EQUATION_DE2000	3	ΔE_{00}
EQUATION_DEH	4	$\Delta E(\text{Hunter})$
EQUATION_DEP	5	$\Delta E_p(\text{DIN6175})$
EQUATION_DEC	6	$\Delta E_c(\text{DIN6175})$
EQUATION_DE99o	7	$\Delta E_{99o}(\text{DIN99o})$

※[機種・バージョン](#)依存します

CustomItem (カスタム項目)

	値	説明
NONE	0	なし
L	1	L^*
a	2	a^*
b	3	b^*
C	4	C^*
h	5	h
HL	6	$L(\text{Hunter})$
Ha	7	$a(\text{Hunter})$
Hb	8	$b(\text{Hunter})$
X	9	X
Y	10	Y
Z	11	Z
x	12	x

CM-MISDK(C#版) リファレンスマニュアル

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°グロス
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	$\Delta L(\text{Hunter})$
DELTA_Ha	-7	$\Delta a(\text{Hunter})$
DELTA_Hb	-8	$\Delta b(\text{Hunter})$
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	$\Delta WI(E313-73)$
DELTA_WI_CIE	-15	$\Delta WI(CIE)$
DELTA_Tint_CIE	-16	$\Delta Tint(CIE)$

CM-MISDK(C#版) リファレンスマニュアル

DELTA_YI_E313	-17	Δ YI(E313)
DELTA_YI_D1925	-18	Δ YI(D1925)
DELTA_B_ISO	-19	Δ B(ISO)
DELTA_GU	-20	Δ GU
MI	-21	MI
DELTA_E_ab	-22	Δ E*ab
CMC	-23	CMC
DELTA_E_94	-24	Δ E*94
DELTA_E_00	-25	Δ E00
DELTA_HE	-26	Δ E(Hunter)
DELTA_E_99o	-27	Δ E99o
Strength_XYZ	-28	StrengthXYZ
Strength_X	-29	StrengthX
Strength_Y	-30	StrengthY
Strength_Z	-31	StrengthZ
GreyScale	-32	GreyScale
DELTA_WI_GANZ	-33	Δ WI(Ganz)
DELTA_TINT_GANZ	-34	Δ Tint(Ganz)

※[機種・バージョン](#)依存します**DisplayData(表示データ)**

	値	説明
AVERAGE	0	AVERAGE(DP)
LEFT	1	LEFT

※[機種・バージョン](#)依存します**FilterIndex(フィルター設定)**

	値	説明
OFF	0	OFF
FILTER_SAVE	1	保存されているデータのみ表示します
FILTER_GROUP	2	指定されたグループ番号に該当するデータのみ表示します

※[機種・バージョン](#)依存します

CM-MISDK(C#版) リファレンスマニュアル**InstrumentMode(機器モード)**

	値	説明
INSTRUMENTMODE_NORMAL	0	ノーマルモード
INSTRUMENTMODE_SIMPLE	1	シンプルモード

※[機種・バージョン](#)依存します**UserType(ユーザータイプ)**

	値	説明
USERTYPE_ADMIN	0	管理者
USERTYPE_WORKER	1	作業者

※[機種・バージョン](#)依存します**ScreenDirection(画面の表示向き)**

	値	説明
SCREENDIR_0	0	回転なし
SCREENDIR_180	1	180°回転

※[機種・バージョン](#)依存します**DateFormat(日付書式)**

	値	説明
YYMMDD	0	YYYY/MM/DD
MMDDYY	1	MM/DD/YYYY
DDMMYY	2	DD/MM/YYYY

Language(言語)

	値	説明
LANGUAGE_ENGLISH	0	英語
LANGUAGE_JAPANESE	1	日本語
LANGUAGE_GERMAN	2	ドイツ語
LANGUAGE_FRENCH	3	フランス語
LANGUAGE_SPANISH	4	スペイン語

CM-MISDK(C#版) リファレンスマニュアル

LANGUAGE_ITALIAN	5	イタリア語
LANGUAGE_CHINESE_S	6	中国語(簡体)
LANGUAGE_PORTUGUESE	7	ポルトガル語
LANGUAGE_RUSSIAN	8	ロシア語
LANGUAGE_POLISH	9	ポーランド語
LANGUAGE_TURKISH	10	トルコ語

JobStepType(ジョブステップ種類)

	値	説明
JOB_STEPTYPE_OPERATION	0	操作ステップ
JOB_STEPTYPE_RESULT	1	結果ステップ

OnOff(ON・OFF)

	値	説明
OFF	0	OFF
ON	1	ON

IsThereData(YES・NO)

	値	説明
NO	0	NO
YES	1	YES

DateType(日時種類)

	値	説明
DATETYPE_COLOR	0	色
DATETYPE_GLOSS	1	光沢

ToleranceId(限界値 ID)

	値	説明
TOLERANCE_ID_L	-1	ΔL^*
TOLERANCE_ID_A	-2	Δa^*

CM-MISDK(C#版) リファレンスマニュアル

TOLERANCE_ID_B	-3	Δb^*
TOLERANCE_ID_C	-4	ΔC^*
TOLERANCE_ID_H	-5	ΔH^*
TOLERANCE_ID_HL	-6	$\Delta L(\text{Hunter})$
TOLERANCE_ID_HA	-7	$\Delta a(\text{Hunter})$
TOLERANCE_ID_HB	-8	$\Delta b(\text{Hunter})$
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	$\Delta WI(E313-73)$
TOLERANCE_ID_WI_C	-15	$\Delta WI(CIE)$
TOLERANCE_ID_TINT_C	-16	$\Delta Tint(CIE)$
TOLERANCE_ID_YI_E	-17	$\Delta YI(E313)$
TOLERANCE_ID_YI_D	-18	$\Delta YI(D1925)$
TOLERANCE_ID_B_ISO	-19	$\Delta B(ISO)$
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	$\Delta E00$
TOLERANCE_ID_DEH	-26	$\Delta E(\text{Hunter})$
TOLERANCE_ID_DEP_DIN6175	-27	$\Delta Ep(DIN6175)$
TOLERANCE_ID_DEC_DIN6175	-28	$\Delta Ec(DIN6175)$
TOLERANCE_ID_FF	-29	ΔFF
TOLERANCE_ID_DE99O	-30	$\Delta E99o$
TOLERANCE_ID_DEC_AUDI2000	-31	$\Delta Ec(\text{Audi2000})$
TOLERANCE_ID_MDEC_AUDI2000	-32	ΔEc 平均(Audi2000)
TOLERANCE_ID_DECM_AUDI2000	-33	ΔEc 最大(Audi2000)
TOLERANCE_ID_DEP_AUDI2000	-34	$\Delta Ep(\text{Audi2000})$
TOLERANCE_ID_MDEP_AUDI2000	-35	ΔEp 平均(Audi2000)
TOLERANCE_ID_DEPM_AUDI2000	-36	ΔEp 最大(Audi2000)
TOLERANCE_ID_DSTRENGTH_XYZ	-37	$\Delta StrengthXYZ$
TOLERANCE_ID_DSTRENGTH_X	-38	$\Delta strengthX$

CM-MISDK(C#版) リファレンスマニュアル

TOLERANCE_ID_DSTRENGTH_Y	-39	Δ strengthY
TOLERANCE_ID_DSTRENGTH_Z	-40	Δ strengthZ
TOLERANCE_ID_DOPACITY	-41	オパシティ差
TOLERANCE_ID_DGRAYSCALE	-42	グレースケール差
TOLERANCE_ID_WI_G	-43	Δ WI(Ganz)
TOLERANCE_ID_TINT_G	-44	Δ Tint(Ganz)

※[機種・バージョン](#)依存します

CM-MISDK(C#版) リファレンスマニュアル**6. エラー/警告****6.1 エラー一覧**

エラーID	値		
KmSuccess	0	内容	処理が正常に完了しました
		対応	-
KmErNoConnect	10	内容	指定した仮想 COM ポート番号に接続している測定器がありません
		対応	下記の可能性があります ・機器の電源を ON していますか？ ・ケーブルを正しく接続していますか？ ・COM ポート番号は合っていますか？ ・他のソフトで既に制御していませんか？
		注意	短い測定間隔で連続測定を行うと機器との接続が切れる可能性があります その場合、機器の電源を入れ直してください
KmErInvalidParameter	25	内容	指定したパラメータが正しくありません
		対応	・入力範囲をチェックし、適切な値を入力してください
KmErCannotCommand	30	内容	指定された API は現在の機種では対応していません
		対応	-
KmErNoData	45	内容	データが存在しません
		対応	・事前に必要なデータを登録する必要があります
KmErDataProtected	46	内容	データが保護されています
		対応	・データ保護を解除してから操作してください
KmErOutOfRangeValue	50	内容	機器で測定できる範囲外の値です
		対応	-
KmErConnectFailed	100	内容	測定器との接続に失敗しました または、Connect をしていません
		対応	・Connect で接続してから処理を行ってください
KmErDevice	110	内容	機器内のデバイスに異常があります
		対応	機器を再起動して解消するかをご確認ください 解消しない場合は、弊社サービスへお問い合わせください
KmErAd	111	内容	機器内の A/D 変換デバイスが動作不良です

CM-MISDK(C#版) リファレンスマニュアル

		対応	※機器を再起動しても解消しない場合は、弊社サービスにお問い合わせください
KmErCharge	112	内容	機器内の発光回路充電が動作不良です
		対応	※機器を再起動しても解消しない場合は、弊社サービスにお問い合わせください
KmErFlash	113	内容	機器内の発光回路が動作不良です
		対応	※機器を再起動しても解消しない場合は、弊社サービスにお問い合わせください
KmErFinder	114	内容	ファインダーが開いたままなので操作できません
		対応	・ファインダーを閉じてから操作してください ※ファインダーを閉じていてもエラーが出る場合は、弊社サービスにお問い合わせください
KmErCalculation	115	内容	必要な情報が不足しているため、演算できません
		対応	※機器を再起動しても解消しない場合は、弊社サービスにお問い合わせください
KmErCalibration	120	内容	校正を正しい手順で実行していません
		対応	・ゼロ校正時にゼロ校正ボックスを使用していますか？ ・白色校正時に白色校正板を使用していますか？ ・光沢校正時に光沢基準板を使用していますか？
KmErCalibrationRequired	130	内容	事前に必要な校正を実行していません
		対応	・白色校正前にゼロ校正を完了している必要があります ・光沢校正前に白色校正を完了している必要があります ・測定前に光沢校正を完了している必要があります
KmErNoCalibrationData	131	内容	校正データが設定されていません
		対応	・校正データを設定してください
KmErTiltDetection	140	内容	機器が傾いています
		対応	・機器を試料に正しく設置する必要があります
KmErNotUse	170	内容	他の設定との組み合わせにより、この設定を使用できません
		対応	・他の設定を変更することで解消します
KmErDisagreeCond	171	内容	条件があわないため、設定できません
		対応	・同じモードのデータを関連付ける必要があります 例えば、色 + 光沢のデータに、オパシティのデータを関連付けることはできません
KmErUvAdjust	172	内容	測定試料に蛍光が含まれていません
		対応	適切な試料を測定してください

CM-MISDK(C#版) リファレンスマニュアル

KmErBattery	180	内容	電池電圧が低下しています
		対応	・電池を充電するか、電源に接続する必要があります ※充電しても解消しない場合は、弊社サービスにお問い合わせください
KmErMemory	181	内容	機器内のメモリの読み書きが動作不良です
		対応	※機器を再起動しても解消しない場合は、弊社サービスにお問い合わせください
KmErMotor	182	内容	機器内のモーターが動作不良です
		対応	※機器を再起動しても解消しない場合は、弊社サービスにお問い合わせください
KmErNotSupported	190	内容	機器として機能に対応していますが、使用することができません
		対応	・使用している形式では対応していません。異なる形式の API を使用してください ・ SDK のバージョンが古くてデータが使用できません。バージョンアップを行ってください。
KmErCalculateColor	195	内容	反射率から色彩値を算出できません
		対応	
KmErCalculateCoef	196	内容	蛍光係数を算出できません
		対応	許容幅を見直してください
KmEr	200	内容	予期せぬエラーが発生しました
		対応	機器を再起動して解消するかをご確認ください

6.2 警告一覧

ワーニング ID	内容
WR1	(電池)電池電圧が低下しています
WR2	(時間)校正後の時間変化が大きいため、再校正をお勧めします
WR3	(時間)定期校正の時期が近づいています
WR4	(時間)定期校正が必要です。定期校正を行ってください。
WR5	(信頼性)測色用照明光源の光量が低下しています
WR6	(信頼性)反射率が保証範囲外です
WR7	(信頼性)光沢値が保証範囲外です
WR8	(信頼性)光沢用照明光源の光量が低下しています

CM-MISDK(C#版) リファレンスマニュアル**付録 A. 使用できる文字コード**

名前・コメントとして使用できる文字コードは、以下の通りです。

	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	

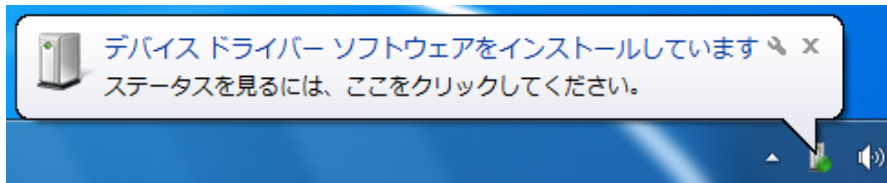
CM-MISDK(C#版) リファレンスマニュアル

付録 B. デバイスドライバーのインストール

測定器を USB 接続するには、あらかじめデバイスドライバーをインストールする必要があります。

初めに、PC に測定器を接続し、測定器の電源を ON にします。

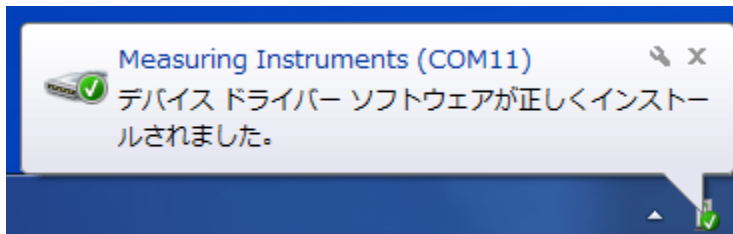
自動でドライバーインストールが開始され、Windows7 の場合、画面右下タスクバーに「デバイスドライバーソフトウェアをインストールしています」とポップアップが表示されます。



また、Windows10 によってインストールされるデバイスドライバは正しく動作しません。このため、下記のインストール手順に従ってデバイスドライバファイル"KMMIUSB.INF"を手動でインストールしてください。

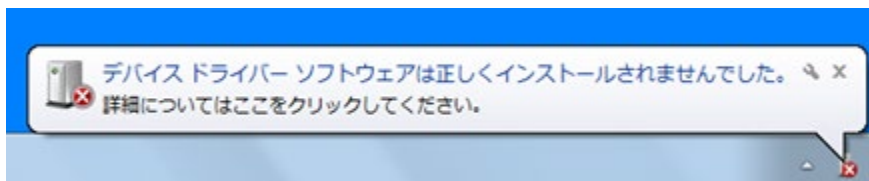
自動インストール

自動インストールに成功した場合は、インストール終了です。



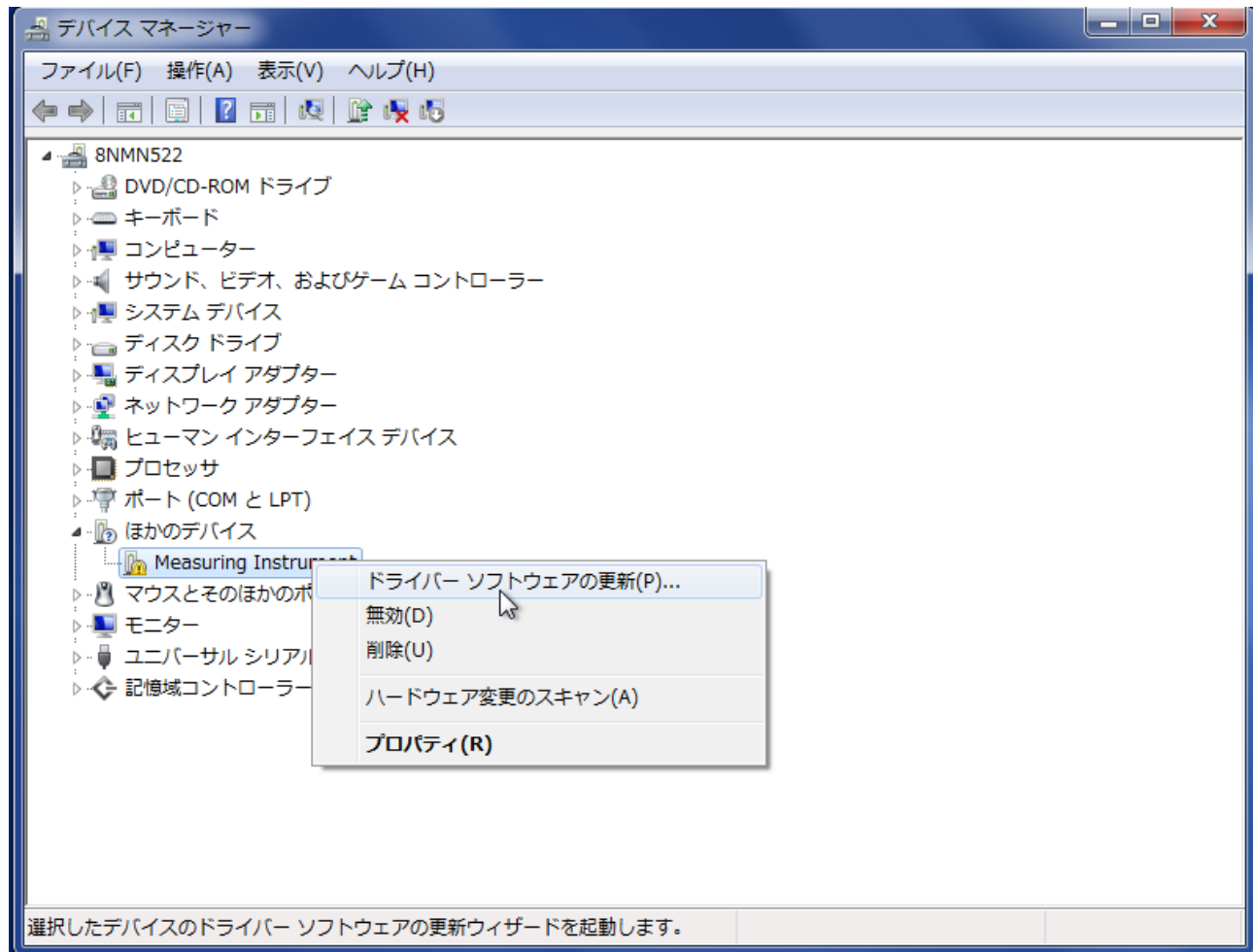
手動インストール

自動インストールに失敗した場合は、以下の手順で手動インストールを実行してください。



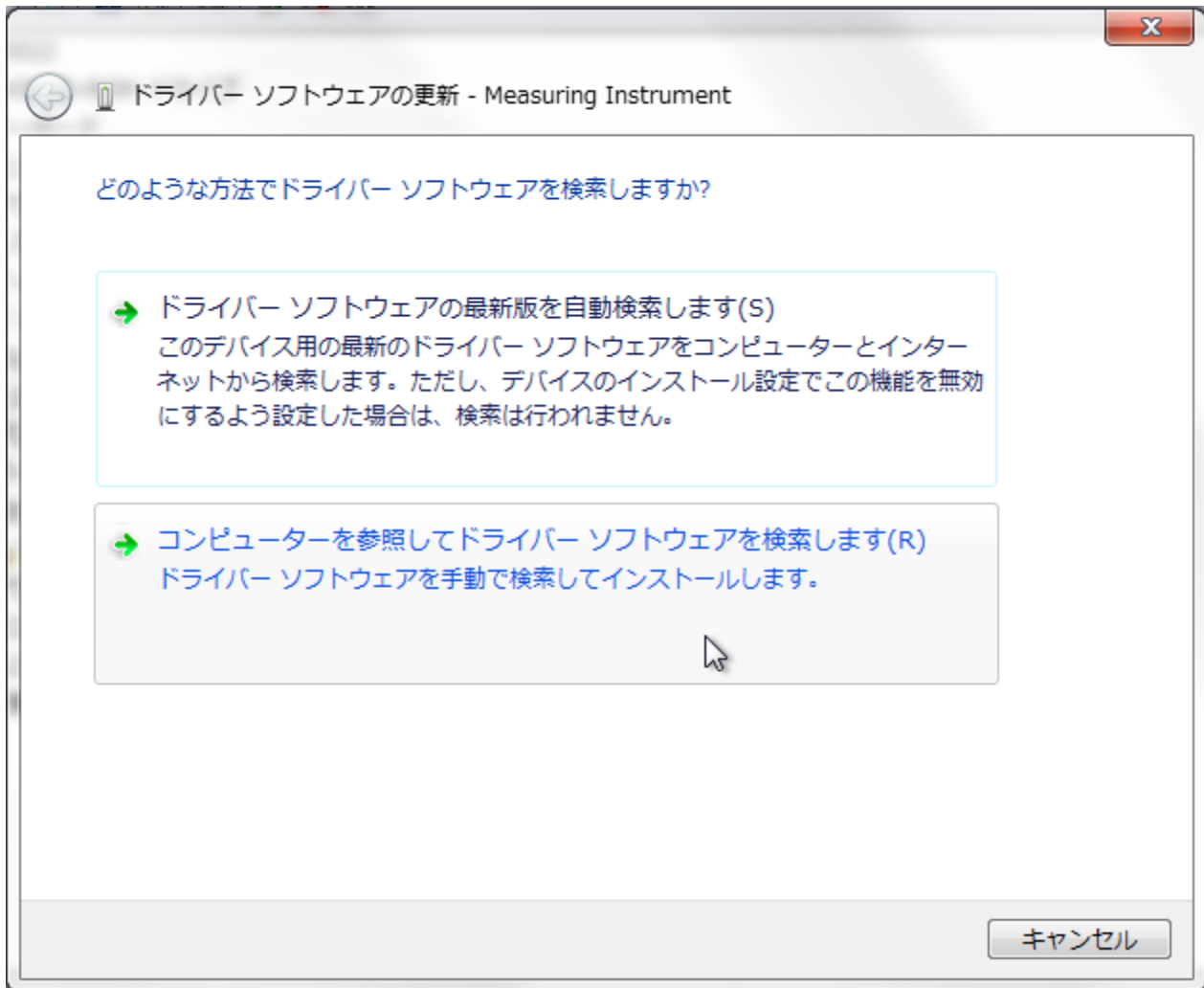
CM-MISDK(C#版) リファレンスマニュアル

デバイスマネージャーを開き、[ほかのデバイス]-[Measuring Instruments]を右クリックし、ドライバーソフトウェアの更新をクリックします。ドライバーインストールが失敗している状態では、下図のように[Measuring Instruments]に警告マークが付いています。



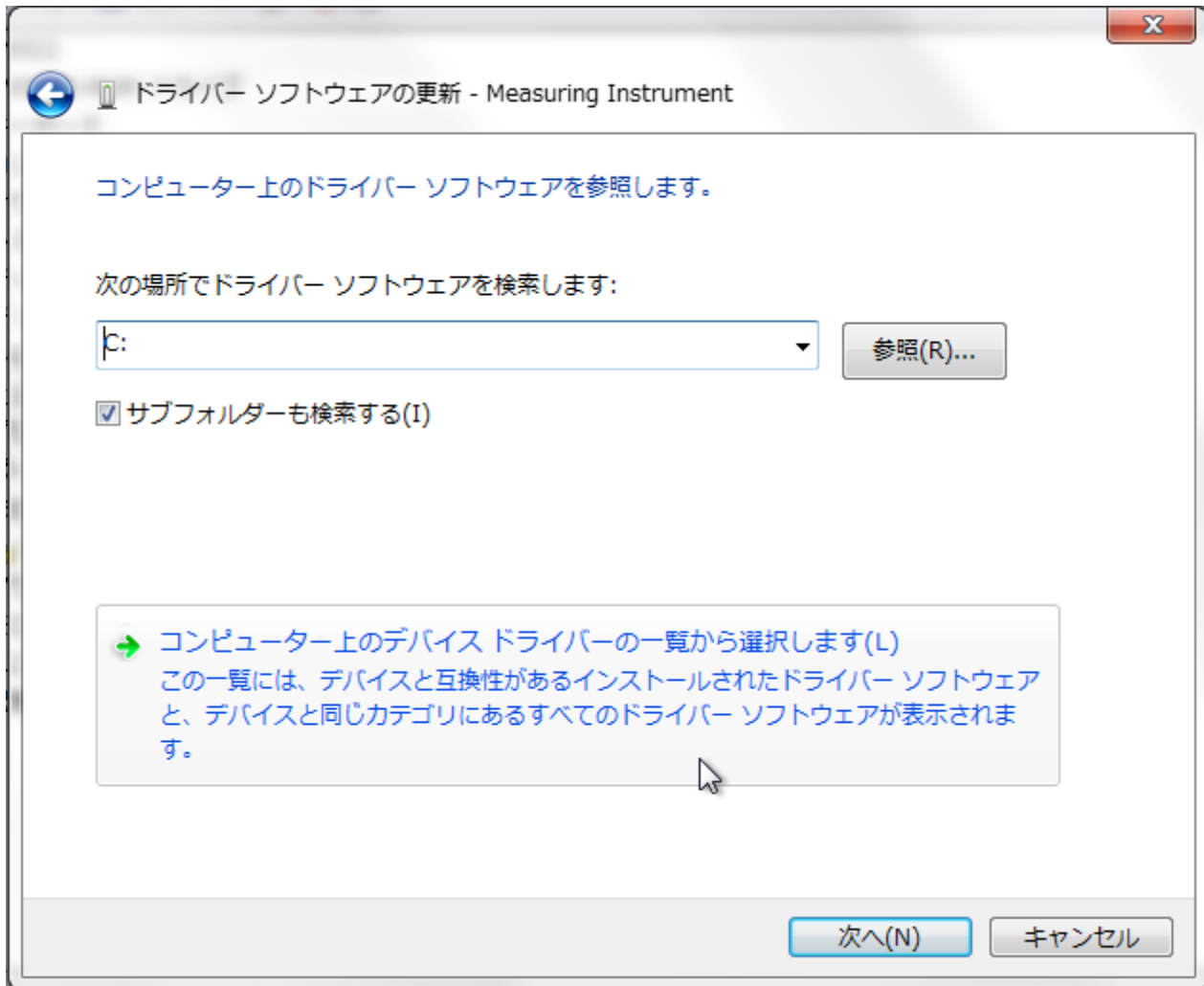
CM-MISDK(C#版) リファレンスマニュアル

「コンピューターを参照してドライバーソフトウェアを検索します」をクリックします。



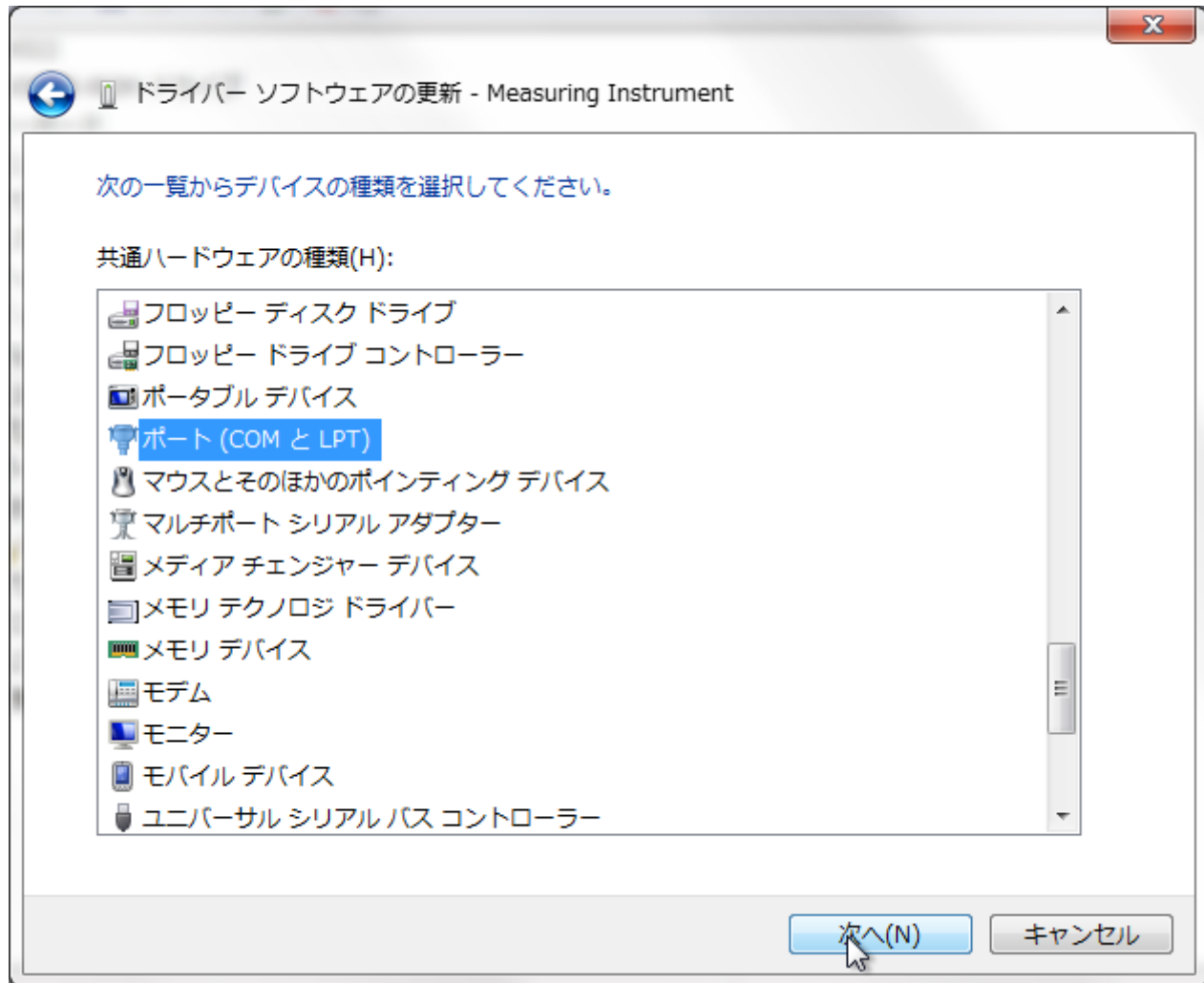
CM-MISDK(C#版) リファレンスマニュアル

「コンピューター上のデバイスドライバーの一覧から選択します」をクリックします。



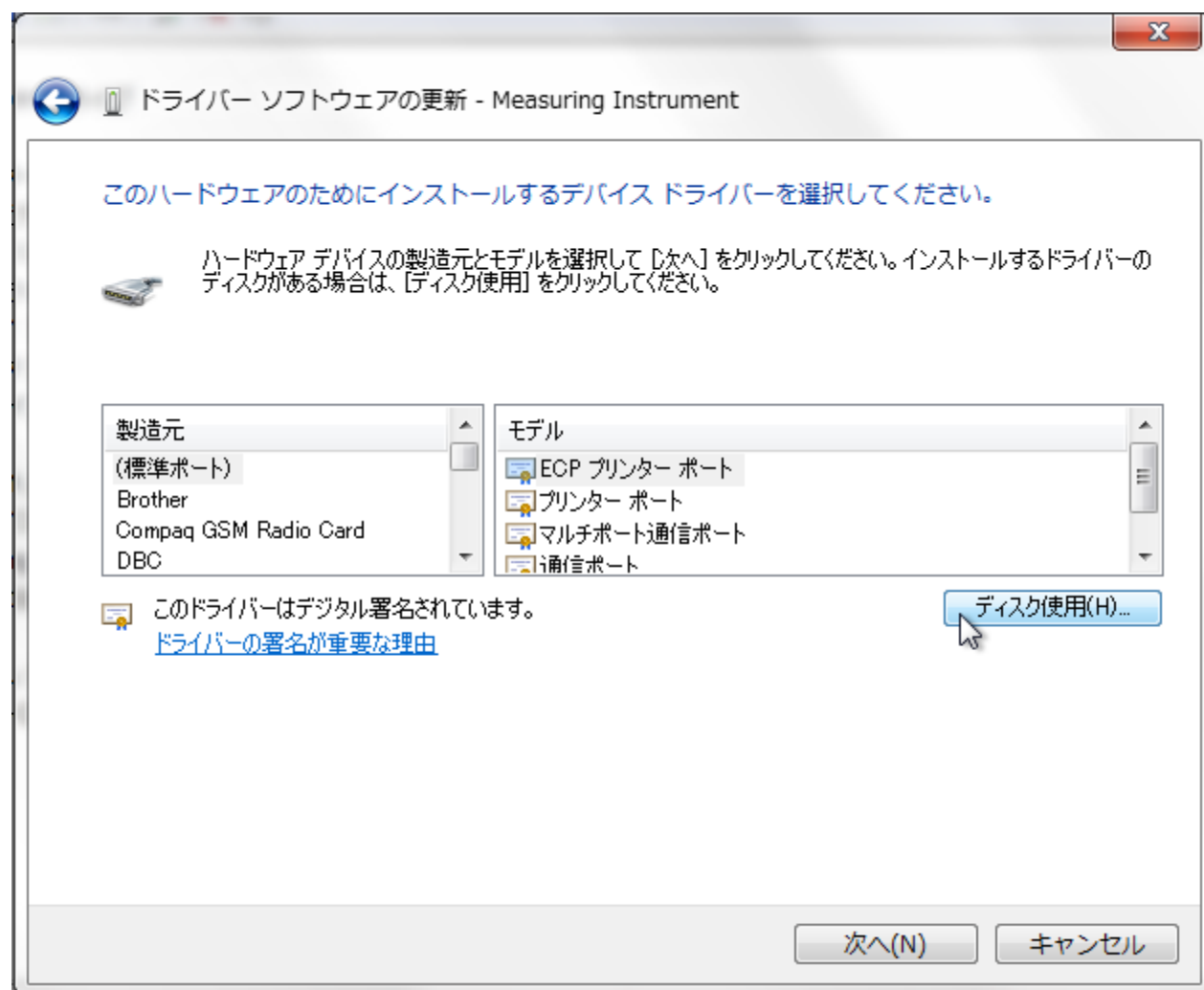
CM-MISDK(C#版) リファレンスマニュアル

共通ハードウェアの種類一覧から、「ポート(COM と LPT)」をクリックし、「次へ」をクリックします。



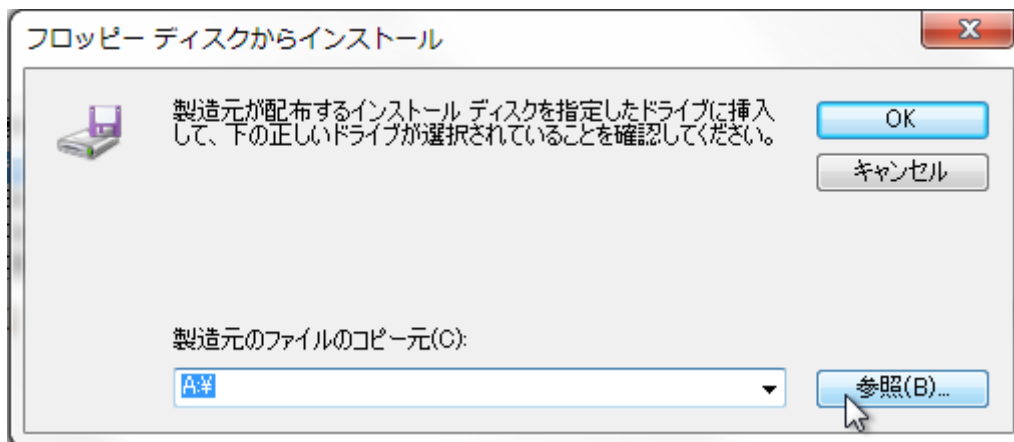
CM-MISDK(C#版) リファレンスマニュアル

ディスク使用をクリックします。

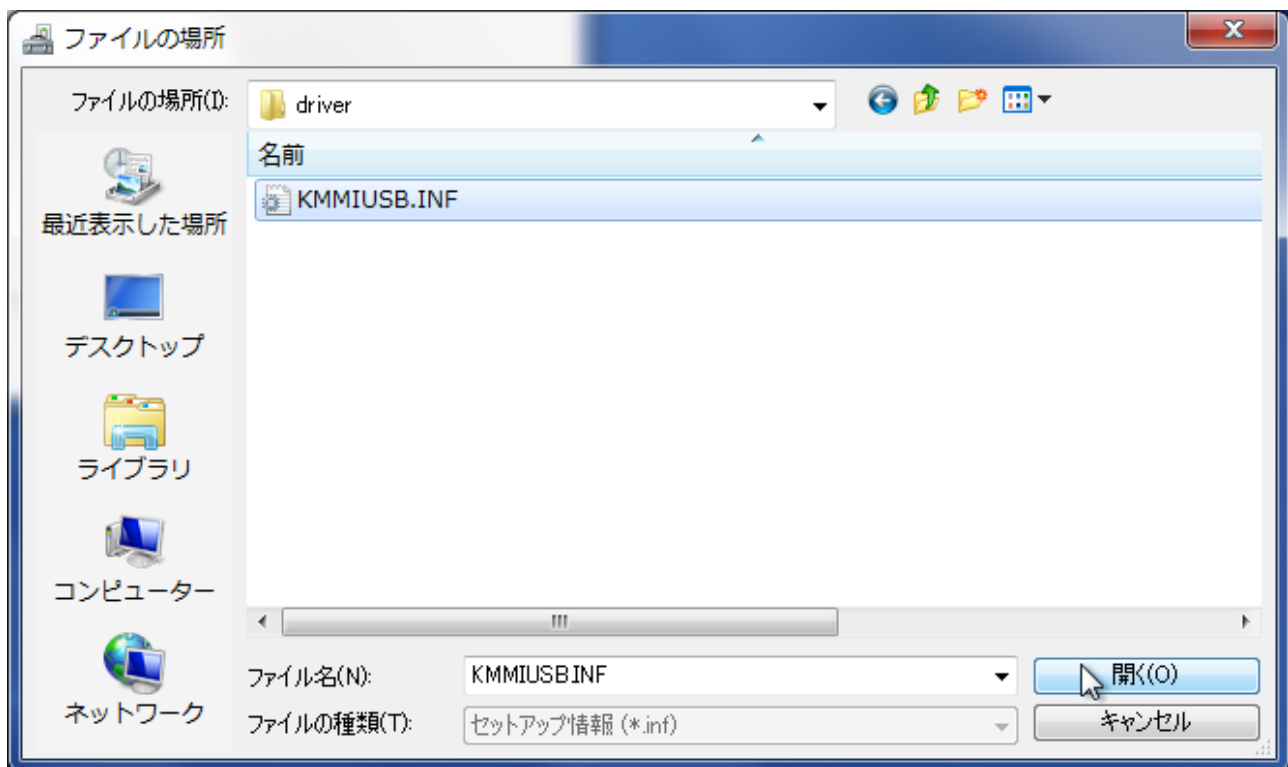


CM-MISDK(C#版) リファレンスマニュアル

参照をクリックし、

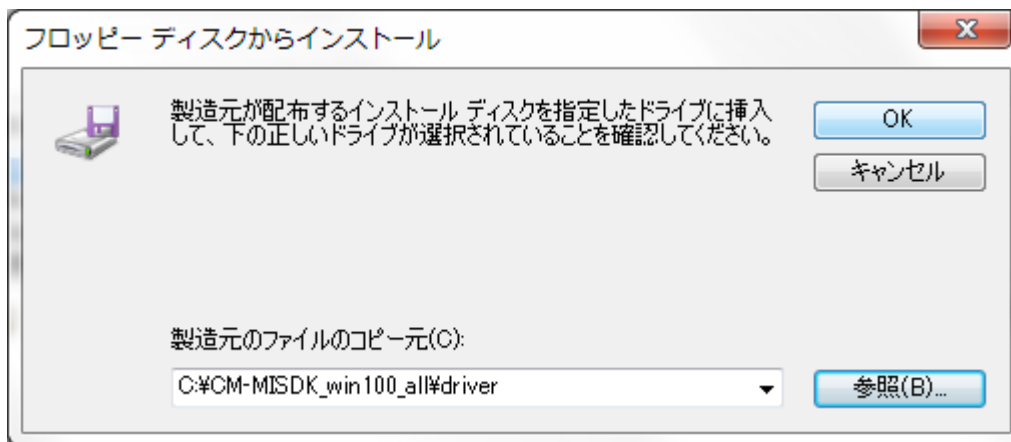


CM-MISDK_win100_all/driver/にある「KMMIUSB.INF」ファイルを選択し、「開く」をクリックします。

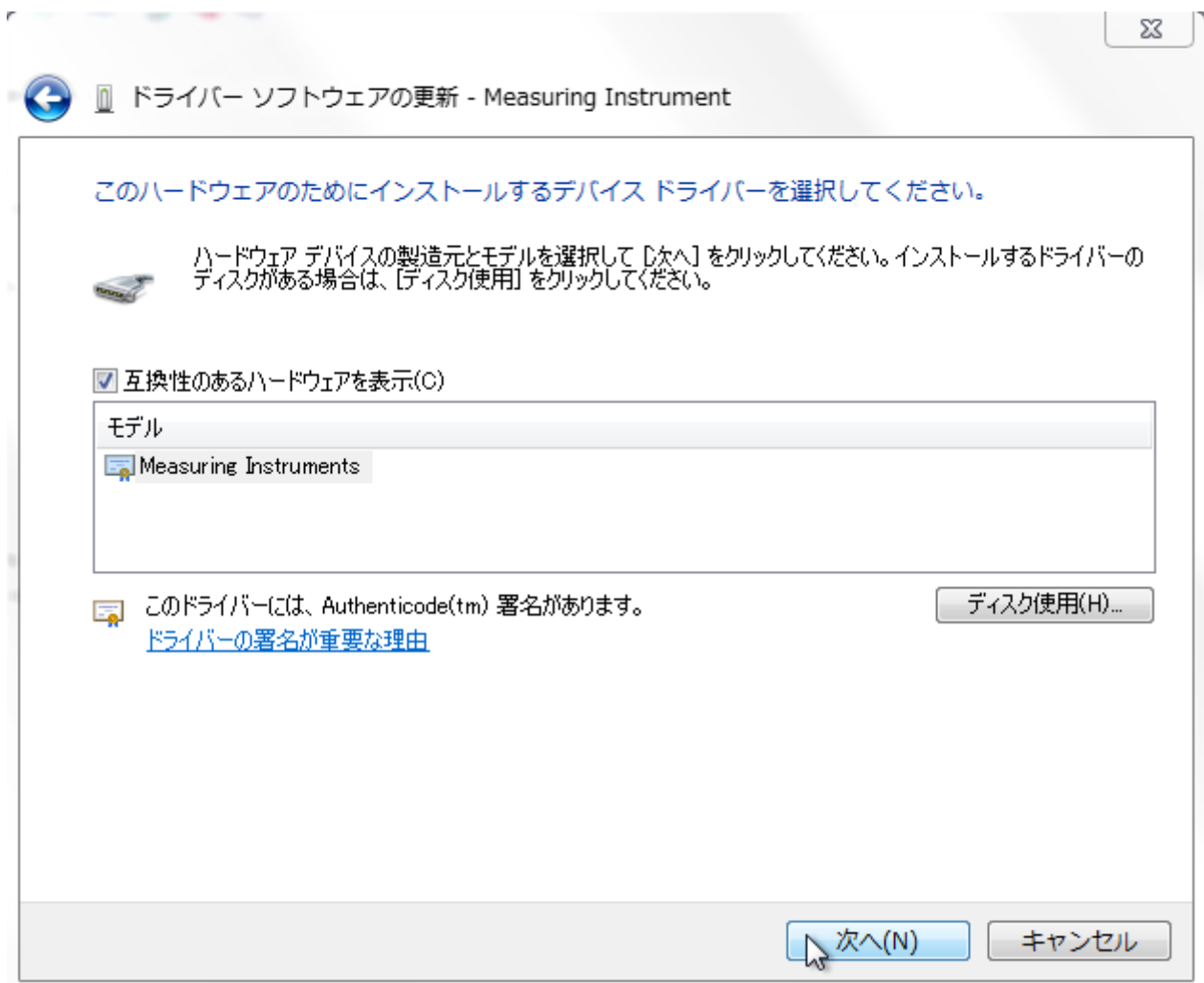


CM-MISDK(C#版) リファレンスマニュアル

「OK」をクリックします。

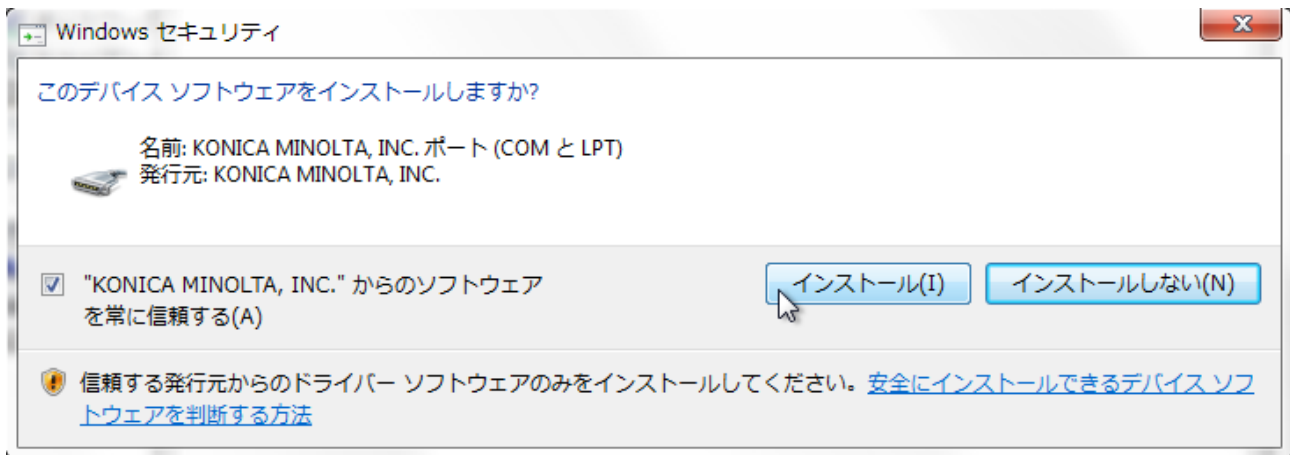


「次へ」をクリックします。

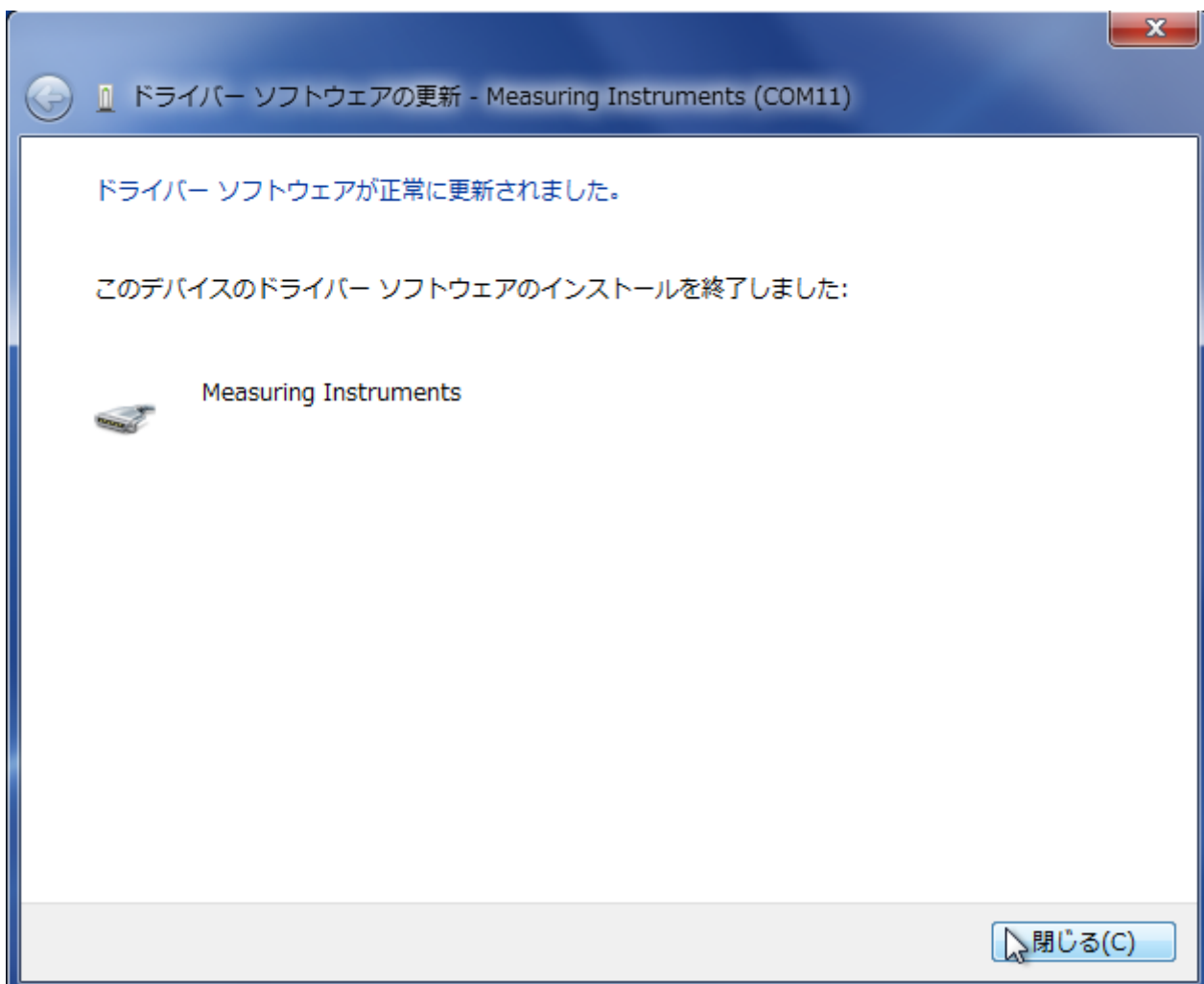


CM-MISDK(C#版) リファレンスマニュアル

「インストール」をクリックします。



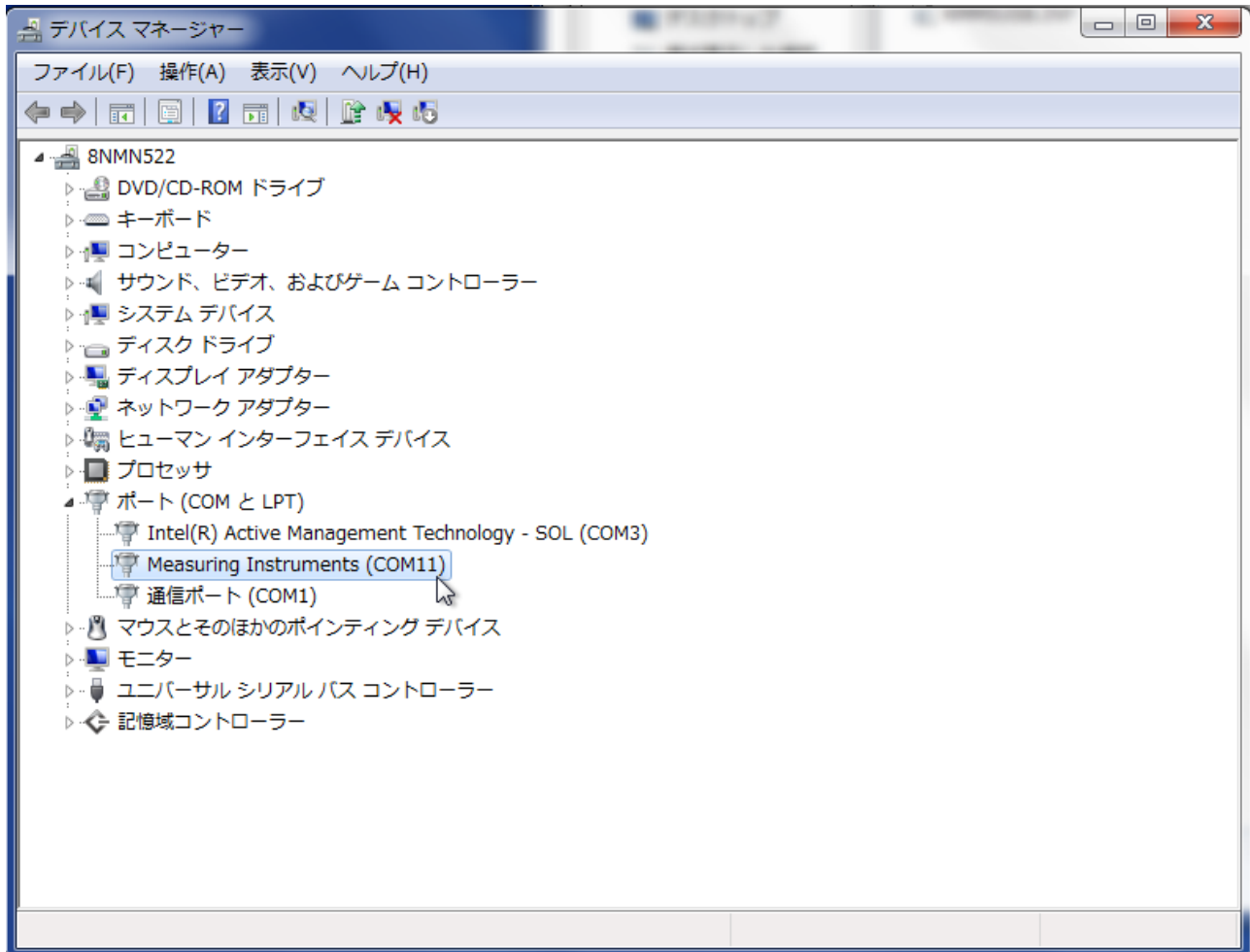
インストール終了を確認したら、閉じるをクリックします。



CM-MISDK(C#版) リファレンスマニュアル

Measuring Instruments に警告マークが表示されておらず、続けて COM 番号が表示されていることを確認したら右上の×でウィンドウを閉じます。

(下図では COM11 となっていますが、この COM 番号は PC によって異なります。)



以上で、ドライバインストールは終了です。

CM-MISDK(C#版) リファレンスマニュアル**付録 C. 機器・バージョンによる設定可能なパラメータ一覧**

機器・バージョンにより、設定可能なパラメータが異なります。それぞれ下記の一覧を参照してください。

警告状態

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
WR1	○	○	○	○	○	○	○	○	○	○
WR2	○	○	○	○	○	○	○	○	○	○
WR3	○	○	○	○	○	○	○	○	○	○
WR4	○	○	○	○	○	○	○	○	○	○
WR5	○	○	○	○	○	○	○	○	○	○
WR6	○	○	○	○	○	○	○	○	○	○
WR7	○	○	×	×	○	○	×	×	×	×
WR8	○	○	×	×	○	○	×	×	×	×

校正状態

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
StatusZero	○	○	○	○	○	○	○	○	○	○
StatusWhite	○	○	○	○	○	○	○	○	○	○
StatusGloss	○	○	×	×	○	○	×	×	×	×

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
StatusMeasure	○	○	○	○	○	○	○	○	○	○
StatusMeasureWrn	○	○	○	○	○	○	○	○	○	○
StatusUser	○	○	×	×	○	○	○	○	○	○

蛍光調整条件

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
UVADJ_PROFILE						○		○		
UVADJ_WI						○		○		
UVADJ_TINT						×		×		
UVADJ_WITINT						○		○		
UVADJ_BRIGHTNESS						○		○		
UVADJ_GG						○		○		

蛍光係数用データ種類

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
UVADJ_DATATYPE_SCI						○		○		
UVADJ_DATATYPE_SCE						○		○		

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
UVADJ_DATATYPE_NONE						×		×		

測定径

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
AREA_MAV	○	○			○	○	○	○		
AREA_SAV	○	○			○	○	○	○		
AREA_LAV	×	×			×	×	×	×		
AREA_LMAV	×	×			×	×	×	×		

測定タイプ

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
MEASTYPE_REF										
MEASTYPE_TRA										

CM-MISDK(C#版) リファレンスマニュアル**照射方向**

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
LDIRECTION_NONE			○	○						
LDIRECTION_L			○	○						
LDIRECTION_R			○	○						
LDIRECTION_DP			○	○						

測定角度

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
MEAS_ANGLE_M15			○	○						
MEAS_ANGLE_15			○	○						
MEAS_ANGLE_25			○	○						
MEAS_ANGLE_45			○	○						
MEAS_ANGLE_75			○	○						
MEAS_ANGLE_110			○	○						

CM-MISDK(C#版) リファレンスマニュアル**傾き検出**

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
OFF			○	○						
ON			○	○						

測定モード

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
MeasModeColorAndGloss	○	○			○	○	×	×	×	×
MeasModeColorOnly	○	○			○	○	○	○	○	○
MeasModeGlossOnly	○	○			○	○	×	×	×	×
MeasModeOpacity	×	×			○	○	○	○	○	○

正反射光処理

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
SC_SCI					○	○	○	○	○	○
SC_SCE					○	○	○	○	○	○
SC_SCIE					○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル**UV 条件**

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
UV_100					○	○	○	○		
UV_CUT400					○	○	○	○		
UV_CUT420					×	×	×	×		
UV_CUT400N					×	△(*)	×	△(*)		
UV_CUT400L					×	×	×	×		
UV_CUT420N					×	×	×	×		
UV_CUT420L					×	×	×	×		
UV_100_CUT400					×	×	×	×		
UV_100_CUT420					×	×	×	×		
UV_100_CUT400N					×	○	×	○		
UV_100_CUT400L					×	×	×	×		
UV_100_CUT420N					×	×	×	×		
UV_100_CUT420L					×	×	×	×		

(*) : データプロパティ、ジョブの場合に使用します

CM-MISDK(C#版) リファレンスマニュアル**自動平均回数**

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
最小	1	1	1	1	1	1	1	1	1	1
最大	10	10	10	10	10	10	10	10	10	10

手動平均回数

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
最小	1	1	1	1	1	1	1	1	1	1
最大	30	30	10	10	30	30	30	30	30	30

手動平均保存モード

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
SAVEMODE_AUTO	○	○	○	○	○	○	○	○	○	○
SAVEMODE_MANUAL	○	○	○	○	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル**SMC 設定**

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
OFF					○	○	○	○	○	
ON					○	○	○	○	○	

SMC 回数

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
最小					3	3	3	3	3	
最大					10	10	10	10	10	

表示形式

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
DISPTYPE_ABS	○	○	○	○	○	○	○	○	○	○
DISPTYPE_DIF	○	○	○	○	○	○	○	○	○	○
DISPTYPE_ABSDIF	○	○	×	×	○	○	○	○	○	○
DISPTYPE_CUSTOM	○	○	×	×	○	○	○	○	○	○
DISPTYPE_GRAPH_ABS	○	○	○	○	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
DISPTYPE_GRAPH_DIF	○	○	○	○	○	○	○	○	○	○
DISPTYPE_GRAPH_REF	○	○	×	×	○	○	○	○	○	○
DISPTYPE_PASS_FAIL	○	○	○	○	○	○	○	○	○	○
DISPTYPE_MI	×	×	○	○	×	×	×	×	×	×
DISPTYPE_GRAPH_LINE	×	×	○	○	×	×	×	×	×	×
DISPTYPE_AUDI2000_EC	×	×	×	○	×	×	×	×	×	×
DISPTYPE_AUDI2000_EP	×	×	×	○	×	×	×	×	×	×

観察視野

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
Deg02	○	○	○	○	○	○	○	○	○	○
Deg10	○	○	○	○	○	○	○	○	○	○

観察光源

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
ILL_None	△(*)	△(*)	△(*)	△(*)	△(*)	△(*)	△(*)	△(*)	△(*)	△(*)
ILL_A	○	○	○	○	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
ILL_C	○	○	○	○	○	○	○	○	○	○
ILL_D50	○	○	○	○	○	○	○	○	○	○
ILL_D65	○	○	○	○	○	○	○	○	○	○
ILL_ID50	○	○	×	×	○	○	○	○	○	○
ILL_ID65	○	○	×	×	○	○	○	○	○	○
ILL_F2	○	○	○	○	○	○	○	○	○	○
ILL_F6	○	○	○	○	○	○	○	○	○	○
ILL_F7	○	○	○	○	○	○	○	○	○	○
ILL_F8	○	○	○	○	○	○	○	○	○	○
ILL_F10	○	○	○	○	○	○	○	○	○	○
ILL_F11	○	○	○	○	○	○	○	○	○	○
ILL_F12	○	○	○	○	○	○	○	○	○	○
ILL_USER1	○	○	×	○	○	○	○	○	○	×

(*)第 2 光源のみ設定可能

表色系

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
COLOR_LAB	○	○	○	○	○	○	○	○	○	○
COLOR_LCH	○	○	○	○	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
COLOR_HLAB	○	○	×	×	○	○	○	○	○	×
COLOR_YXY	○	○	×	×	○	○	○	○	○	○
COLOR_XYZ	○	○	×	×	○	○	○	○	○	○
COLOR_MUNSELL_C	○	○	×	×	○	○	○	○	○	○

色差式

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
EQUATION_DE1976	○	○	○	○	○	○	○	○	○	○
EQUATION_CMC	○	○	○	○	○	○	○	○	○	○
EQUATION_DE1994	○	○	○	○	○	○	○	○	○	○
EQUATION_DE2000	○	○	○	○	○	○	○	○	○	○
EQUATION_DEH	○	○	×	×	○	○	○	○	○	×
EQUATION_DEP	×	×	○	○	×	×	×	×	×	×
EQUATION_DEC	×	×	○	○	×	×	×	×	×	×
EQUATION_DE99o	×	○	×	○	○	○	○	○	○	×

CM-MISDK(C#版) リファレンスマニュアル

カスタム項目

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
なし	○	○			○	○	○	○	○	○
L*	○	○			○	○	○	○	○	○
a*	○	○			○	○	○	○	○	○
b*	○	○			○	○	○	○	○	○
C*	○	○			○	○	○	○	○	○
h	○	○			○	○	○	○	○	○
L(Hunter)	○	○			○	○	○	○	○	×
a(Hunter)	○	○			○	○	○	○	○	×
b(Hunter)	○	○			○	○	○	○	○	×
X	○	○			○	○	○	○	○	○
Y	○	○			○	○	○	○	○	○
Z	○	○			○	○	○	○	○	○
x	○	○			○	○	○	○	○	○
y	○	○			○	○	○	○	○	○
H	○	○			○	○	○	○	○	○
V	○	○			○	○	○	○	○	○
C	○	○			○	○	○	○	○	○
WI(E313-73)	○	○			○	○	○	○	○	○
WI(CIE)	○	○			○	○	○	○	○	×

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
Tint(CIE)	○	○			○	○	○	○	○	×
YI(E313)	○	○			○	○	○	○	○	×
YI(D1925)	○	○			○	○	○	○	○	○
B(ISO)	○	○			○	○	○	○	○	×
GU	○	○			○	○	×	×	×	×
UserE1	○	○			○	○	○	○	○	○
UserC1	○	○			○	○	○	○	○	○
UserE2	○	○			○	○	○	○	○	○
UserC2	○	○			○	○	○	○	○	○
UserE3	○	○			○	○	○	○	○	○
UserC3	○	○			○	○	○	○	○	○
8°グロス	×	×			×	×	○	○	○	×
WI(Ganz)	×	×			×	○	×	○	×	×
Tint(Ganz)	×	×			×	○	×	○	×	×
ΔL^*	○	○			○	○	○	○	○	○
Δa^*	○	○			○	○	○	○	○	○
Δb^*	○	○			○	○	○	○	○	○
ΔC^*	○	○			○	○	○	○	○	○
ΔH^*	○	○			○	○	○	○	○	○
ΔL (Hunter)	○	○			○	○	○	○	○	×
Δa (Hunter)	○	○			○	○	○	○	○	×

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
$\Delta b(\text{Hunter})$	○	○			○	○	○	○	○	×
ΔX	○	○			○	○	○	○	○	○
ΔY	○	○			○	○	○	○	○	○
ΔZ	○	○			○	○	○	○	○	○
Δx	○	○			○	○	○	○	○	○
Δy	○	○			○	○	○	○	○	○
$\Delta WI(\text{E313-73})$	○	○			○	○	○	○	○	○
$\Delta WI(\text{CIE})$	○	○			○	○	○	○	○	×
$\Delta Tint(\text{CIE})$	○	○			○	○	○	○	○	×
$\Delta YI(\text{E313})$	○	○			○	○	○	○	○	×
$\Delta YI(\text{D1925})$	○	○			○	○	○	○	○	○
$\Delta B(\text{ISO})$	○	○			○	○	○	○	○	×
ΔGU	○	○			○	○	×	×	×	×
MI	○	○			○	○	○	○	○	○
ΔE^*ab	○	○			○	○	○	○	○	○
CMC	○	○			○	○	○	○	○	○
ΔE^*94	○	○			○	○	○	○	○	○
$\Delta E00$	○	○			○	○	○	○	○	○
$\Delta E(\text{Hunter})$	○	○			○	○	○	○	○	×
$\Delta E99o$	×	○			○	○	○	○	○	×
StrengthXYZ	×	×			○	○	○	○	○	×

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
StrengthX	×	×			○	○	○	○	○	×
StrengthY	×	×			○	○	○	○	○	×
StrengthZ	×	×			○	○	○	○	○	×
GreyScale	×	×			○	○	○	○	○	×
ΔWI(Ganz)	×	×			×	○	×	○	×	×
ΔTint(Ganz)	×	×			×	○	×	○	×	×

表示する照射方向

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
DIRECTION_DP				○						
DIRECTION_L				○						

基準色フィルター

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
FILTER_OFF	○	○			○	○	○	○	○	○
FILTER_SAVE	○	○			○	○	○	○	○	○
FILTER_GROUP	○	○			○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル**基準色保護**

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
OFF	○	○	○	○	○	○	○	○	○	○
ON	○	○	○	○	○	○	○	○	○	○

限界値 ID

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
ΔL^*	○	○	○	○	○	○	○	○	○	○
Δa^*	○	○	○	○	○	○	○	○	○	○
Δb^*	○	○	○	○	○	○	○	○	○	○
ΔC^*	○	○	○	○	○	○	○	○	○	○
ΔH^*	○	○	○	○	○	○	○	○	○	○
$\Delta L(\text{Hunter})$	○	○	×	×	○	○	○	○	○	×
$\Delta a(\text{Hunter})$	○	○	×	×	○	○	○	○	○	×
$\Delta b(\text{Hunter})$	○	○	×	×	○	○	○	○	○	×
ΔX	○	○	×	×	○	○	○	○	○	○
ΔY	○	○	×	×	○	○	○	○	○	○
ΔZ	○	○	×	×	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
Δx	○	○	×	×	○	○	○	○	○	○
Δy	○	○	×	×	○	○	○	○	○	○
$\Delta WI(E313-73)$	○	○	×	×	○	○	○	○	○	○
$\Delta WI(CIE)$	○	○	×	×	○	○	○	○	○	×
$\Delta Tint(CIE)$	○	○	×	×	○	○	○	○	○	×
$\Delta YI(E313)$	○	○	×	×	○	○	○	○	○	×
$\Delta YI(D1925)$	○	○	×	×	○	○	○	○	○	○
$\Delta B(ISO)$	○	○	×	×	○	○	○	○	○	×
ΔGU	○	○	×	×	○	○	×	×	×	×
MI	○	○	○	○	○	○	○	○	○	○
ΔE^*ab	○	○	○	○	○	○	○	○	○	○
CMC	○	○	○	○	○	○	○	○	○	○
ΔE^*94	○	○	○	○	○	○	○	○	○	○
$\Delta E00$	○	○	○	○	○	○	○	○	○	○
$\Delta E(Hunter)$	○	○	×	×	○	○	○	○	○	×
$\Delta Ep(DIN6175)$	×	×	○	○	×	×	×	×	×	×
$\Delta Ec(DIN6175)$	×	×	○	○	×	×	×	×	×	×
ΔFF	×	×	○	○	×	×	×	×	×	×
$\Delta E99o$	×	○	×	○	○	○	○	○	○	×
$\Delta Ec(Audi2000)$	×	×	×	○	×	×	×	×	×	×
ΔEc 平均(Audi2000)	×	×	×	○	×	×	×	×	×	×

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
ΔEc 最大(Audi2000)	×	×	×	○	×	×	×	×	×	×
ΔEp(Audi2000)	×	×	×	○	×	×	×	×	×	×
ΔEp 平均(Audi2000)	×	×	×	○	×	×	×	×	×	×
ΔEp 最大(Audi2000)	×	×	×	○	×	×	×	×	×	×
ΔStrengthXYZ	×	×	×	×	○	○	○	○	○	×
ΔstrengthX	×	×	×	×	○	○	○	○	○	×
ΔstrengthY	×	×	×	×	○	○	○	○	○	×
ΔstrengthZ	×	×	×	×	○	○	○	○	○	×
オパシティ差	×	×	×	×	○	○	○	○	○	○
グレースケール差	×	×	×	×	○	○	○	○	○	×
ΔWI(Ganz)	×	×	×	×	×	○	×	○	×	×
ΔTint(Ganz)	×	×	×	×	×	○	×	○	×	×

警告レベル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
最小	0	0	0	0	0	0	0	0	0	0
最大	100	100	100	100	100	100	100	100	100	100

CM-MISDK(C#版) リファレンスマニュアル**機器モード**

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
INSTRUMENTMODE_NORMAL					○	○	○	○	○	○
INSTRUMENTMODE_SIMPLE					○	○	○	○	○	○

ユーザータイプ

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
USERTYPE_ADMIN	○	○	○	○	○	○	○	○	○	○
USERTYPE_WORKER	○	○	○	○	○	○	○	○	○	○

自動印刷

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
OFF	○	○	○	○	○	○	○	○	○	○
ON	○	○	○	○	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル

画面明るさ

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
最小	0	0	0	0	0	0	0	0	0	0
最大	4	4	4	4	4	4	4	4	4	4

表示向き

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
SCREENDIR_0			○	○	○	○	○	○	○	○
SCREENDIR_180			○	○	○	○	○	○	○	○

ビープ音

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
OFF	○	○	○	○	○	○	○	○	○	○
ON	○	○	○	○	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル**校正喚起**

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
最小	1	1	1	1	1	1	1	1	1	1
最大	24	24	24	24	24	24	24	24	24	24

ユーザー校正

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
OFF	○	○			○	○	○	○	○	
ON	○	○			○	○	○	○	○	

定期校正連絡通知

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
OFF	○	○	○	○	○	○	○	○	○	○
ON	○	○	○	○	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル

ゼロ校正スキップ可否

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
OFF					○	○	○	○	○	○
ON					○	○	○	○	○	○

日付書式

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
DF_YYYYMMDD	○	○	○	○	○	○	○	○	○	○
DF_MMDDYYYY	○	○	○	○	○	○	○	○	○	○
DF_DDMMYYYY	○	○	○	○	○	○	○	○	○	○

言語

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
LANGUAGE_ENGLISH	○	○	○	○	○	○	○	○	○	○
LANGUAGE_JAPANESE	○	○	○	○	○	○	○	○	○	○
LANGUAGE_GERMAN	○	○	○	○	○	○	○	○	○	○
LANGUAGE_FRENCH	○	○	○	○	○	○	○	○	○	○

CM-MISDK(C#版) リファレンスマニュアル

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
LANGUAGE_SPANISH	○	○	○	○	○	○	○	○	○	○
LANGUAGE_ITALIAN	○	○	○	○	○	○	○	○	○	○
LANGUAGE_CHINESE_S	○	○	○	○	○	○	○	○	○	○
LANGUAGE_PORTUGUESE	○	○	○	○	○	○	○	○	○	○
LANGUAGE_RUSSIAN	○	○	○	○	○	○	○	○	○	○
LANGUAGE_POLISH	○	○	○	○	○	○	○	○	○	○
LANGUAGE_TURKISH	○	○	○	○	○	○	○	○	○	○

省電力

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
最小		0		0	0	0	0	0	0	0
最大		60		60	60	60	60	60	60	60

ジョブ

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
ジョブ名(サイズ)					20	20	20	20	20	
データ名(サイズ)					30	30	30	30	30	

CM-MISDK(C#版) リファレンスマニュアル

コメント(サイズ)					100	100	100	100	100	
-----------	--	--	--	--	-----	-----	-----	-----	-----	--

日時種類

	25cG 1.0x	25cG 1.1x	M6 1.0x	M6 1.1x	26dG 1.0x	26dG 1.1x	26d 1.0x	26d 1.1x	25d	23d
DATETYPE_COLOR	○	○			○	○				
DATETYPE_GLOSS	○	○			○	○				

CM-MISDK(C#版) リファレンスマニュアル



KONICA MINOLTA