

SMX-15M5x

APIドキュメント

SMX-15M5x Series USB3.0 Camera API Documentation

Revision 1.0

Copyright © 2001-2014 Sumix Corporation

3532 Seagate Way, Suite 100,

Oceanside, CA, 92056

Tel.: (877)233-3385; Fax: (508) 300 5526

Email: camera@sumix.com

www.sumix.com

このドキュメントにある情報は予告なしに変更する場合があります。このドキュメントにあるソフトウェアは、ライセンス許諾に基づいて提供されるもので、当該使用許諾の定めに限って使用と複写が許可されます。

目次

1. データ構造 -----	1
1.1 struct TCameralInfo -----	1
1.2 struct TCameralInfoEx -----	1
1.3 enum Usb_Speed -----	1
1.4 enum SMX_StreamMode -----	2
1.5 enum STROBE_PARAMS -----	2
1.6 struct TFrameParams -----	2
1.7 struct _TStreamStats -----	3
1.8 struct _TFrameInfo -----	3
2. ファンクション -----	4
2.1 smx15m5X_OpenDevice -----	4
2.2 smx15m5X_OpenDeviceEx -----	5
2.3 smx15m5X_CloseDevice-----	5
2.4 smx15m5X_CheckId -----	6
2.5 smx15m5X_GetUsbSpeed -----	6
2.6 smx15m5X_DeviceReset-----	7
2.7 smx15m5X_SwitchFrontLED -----	7
2.8 smx15m5X_ReadFlash -----	8
2.9 smx15m5X_WriteFlash -----	9
2.10 smx15m5X_GetCameralInfo -----	9
2.11 smx15m5X_GetCameralInfoEx -----	10
2.12 smx15m5X_GetStreamMode -----	10
2.13 smx15m5X_SetStreamMode-----	11
2.14 smx15m5X_GetFrame16 -----	12
2.15 smx15m5X_GetFrameEx16 -----	12

2.16	smx15m5X_GetFrameReady16	13
2.17	smx15m5X_GetFrameReadyEx16	13
2.18	smx15m5X_GetFrame8	14
2.19	smx15m5X_GetFrameEx8	15
2.20	smx15m5X_GetFrameReady8	15
2.21	smx15m5X_GetFrameReadyEx8	16
2.22	smx15m5X_GetFrequency	16
2.23	smx15m5X_SetFrequency	17
2.24	smx15m5X_GetFrequencyD	18
2.25	smx15m5X_SetFrequencyD	18
2.26	smx15m5X_GetFrameParams	19
2.27	smx15m5X_SetFrameParams	20
2.28	smx15m5X_GetGain	20
2.29	smx15m5X_SetGain	21
2.30	smx15m5X_SetAllGain	22
2.31	smx15m5X_GetExposureMinMax	23
2.32	smx15m5X_GetExposureFramerateMax	23
2.33	smx15m5X_GetExposure	24
2.34	smx15m5X_SetExposure	24
2.35	smx15m5X_GetExposureMinMaxMs	25
2.36	smx15m5X_GetExposureFramerateMaxMs	26
2.37	smx15m5X_GetExposureMs	27
2.38	smx15m5X_SetExposureMs	27
2.39	smx15m5X_GetFrameRateMinMax	28
2.40	smx15m5X_GetFrameRate	29
2.41	smx15m5X_SetFrameRate	29

2.42	smx15m5X_SetMinFrameRate	30
2.43	smx15m5X_SetMaxFrameRate	30
2.44	smx15m5X_GetRowBlackLevel	31
2.45	smx15m5X_SetRowBlackLevel	32
2.46	smx15m5X_GetGlobalResetRelease	32
2.47	smx15m5X_SetGlobalResetRelease	33
2.48	smx15m5X_GetSnapshotParams	33
2.49	smx15m5X_SetSnapshotParams	34
2.50	smx15m5X_GetStrobeParams	35
2.51	smx15m5X_SetStrobeParams	35
2.52	smx15m5X_GetEventsState	36
2.53	smx15m5X_SetEventsState	37
2.54	smx15m5X_CheckEventsState	37

データ構造

1.1 struct TCameraInfo

ストラクチャのコンポーネントの説明:

SensorType - INT、センサータイプ: 1(カラー)または0(モノクロ)

MaxWidth - INT、最大フレーム幅

MaxHeight - INT、最大フレーム高さ

DeviceName[64] - CHAR、カメラ名

1.2 struct TCameraInfoEx

ストラクチャのコンポーネントの説明:

HWModelID - WORD、ハードウェアモデルID

HWVersion - WORD、ハードウェアバージョン

HWSerial - DWORD、カメラのシリアルナンバー

FWCVersion - WORD、ファームウェアバージョン#1

FWAVersion - WORD、ファームウェアバージョン#2

Reserved - BYTE、保留

1.3 enum Usb_Speed

Usb_UnsupportedSpeed=0

Usb_FullSpeed=1

Usb_HighSpeed=2

Usb_SuperSpeed=3

1.4 enum SMX_StreamMode

Camera_Stopped=0

Camera_Started=1

Camera_Snapshot=2

Camera_SpeedTest=3

1.5 enum STROBE_PARAMS

STROBE_FirstTrigger=0 - トリガー1 (シャッタースキャンの開始)

STROBE_StartOfExposure=1 - 露光の開始 (すべての行は同時に露光) off- VB
による設定

STROBE_ShutterWidth=2 - シャッター幅の終了 (内部シャッター幅カウン
ターの終了)

STROBE_SecondTrigger=3 - トリガー2 (読み出しスキャンの開始)

1.6 struct TFrameParams

ストラクチャのコンポーネントの説明:

StartX - ULONG、フルセンサー領域の左側からのピクセルの単位でのビュー
ポートオフセット。2の倍数の値に設定します。

StartY - ULONG、フルセンサー領域の上部からのピクセルの単位でのビュー
ポートオフセット。2の倍数の値に設定します。

Width - ULONG、ピクセルの単位でのビューポート幅。範囲: [4..2592]で、4の
倍数の値に設定します。

Height - ULONG、ピクセルの単位でのビューポート高さ。範囲: [2..1944]で、2の倍数の値に設定します。

Decimation - LONG、有効な値: [-4,-2,1..7]。1 - デシメーションなし、-4、-2 - ビニングメソッド、2..7 - 出力解像度削減のスキップメソッド

ColorDeep - ULONG、8 - モノクロ、24 - カラーセンサー

MirrorV - BOOLEAN、垂直鏡像反転フラグ。鏡像反転がない場合はFALSE

MirrorH - BOOLEAN、水平鏡像反転フラグ。鏡像反転がない場合はFALSE

1.7 struct _TStreamStats

ストラクチャのコンポーネントの説明:

CameraFrames - ULONG、カメラフレームの数

KernelFrames - ULONG、カーネルフレームの数

UserFrames - ULONG、ユーザーフレームの数

TotalResyncs - ULONG、トータル再同期の数

FrameResyncs - ULONG、フレーム再同期の数

1.8 struct _TFrameInfo

ストラクチャのコンポーネントの説明:

FrameParams - TFrameParams

Packets - USHORT

StreamStats - TStreamStats

Masks[4000] - UCHAR

ファンクション

2.1 smx15m5X_OpenDevice

SMX-15m5Xカメラを開きます。

シンタックス

```
extern HANDLE __stdcall smx15m5X_OpenDevice(ULONG DeviceID);
```

パラメーター:

DeviceID - ULONG変数で、開かれたカメラのインデックスを含みます。システムに接続された各カメラはこのインデックスによって特定されます。範囲:
[0..34]

返り値:

ファンクションが成功した場合はカメラへのハンドル。APIファンクションのその後の呼び出しにはこのハンドルを使用します。

ファンクションが失敗した場合はINVALID_HANDLE_VALUE (数値では-1に等しい)で、このDeviceIDのカメラがないことを意味します。

備考

このファンクションは、複数のカメラを使用するアプリケーションで使用されるカメラIDを示すのに使用することができます。最初に1台のカメラを接続するとDeviceIDは0になり、さらにカメラを接続するとDeviceIDは1つ増加します。

もう必要でなくなったらsmx15m5X_CloseDeviceファンクションを使用してデバイスハンドルを閉じます。

良いやり方は、ファンクションコールの前にデバイスを開いて、ファンクションコールの直後に閉じることです。そのようなやり方によって、プログラムの実行中にカメラを切断/接続しても再起動する必要がないプログラムを作成することができます。

リンク

[smx15m5X_OpenDeviceEx](#), [smx15m5X_CloseDevice](#)

2.2 smx15m5X_OpenDeviceEx

他のアプリケーションでカメラハンドルの使用を有効/無効にできるようにしてSMX-15m5Xカメラを開きます。

シンタックス

```
extern HANDLE __stdcall smx15m5X_OpenDeviceEx(ULONG DeviceID, BOOL Synchronous);
```

パラメーター:

DeviceID - ULONG変数で、開かれたカメラのインデックスを含みます。システムに接続された各カメラはこのインデックスによって特定されます。範囲: [0..34]

Synchronous - BOOL変数で、同期のステータスを含んでいます(TRUE - 有効、FALSE - 無効)。

返り値:

ファンクションが成功した場合はカメラへのハンドル。APIファンクションのその後の呼び出しにはこのハンドルを使用します。

ファンクションが失敗した場合はINVALID_HANDLE_VALUE (数値では-1に等しい)で、このDeviceIDのカメラがないことを意味します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_CloseDevice](#)

2.3 smx15m5X_CloseDevice

SMX-15m5Xカメラを閉じます。

シンタックス

```
extern BOOL __stdcall smx15m5X_CloseDevice(HANDLE H);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_OpenDeviceEx](#)

2.4 smx15m5X_CheckId

カメラIDを取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_CheckId(HANDLE H, CHAR* CheckId, size_t* length);
```

パラメーター:

H - HANDLE変数で、以前に[smx15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#)

2.5 smx15m5X_GetUsbSpeed

カメラのUSB速度を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetUsbSpeed(HANDLE H, PUsb_Speed  
UsbSpeed, PULONG RealSpeedMBpS);
```

パラメーター:

H - HANDLE変数で、以前に[smx15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

UsbSpeed - カメラUSB速度のステータスを受け取る *Usb_Speed* 列挙タイプへのポインター。

RealSpeedMBpS - メガバイト/秒の単位で実際のUSB速度を受け取る *ULONG* 変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#)

2.6 smx15m5X_DeviceReset

カメラのファームウェアをリセットします。

シンタックス

```
extern BOOL __stdcall smx15m5X_DeviceReset(HANDLE H);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

このファンクションを呼び出した後にカメラで作業を続けるにはsmx15m5X_OpenDeviceを使用して新規ハンドルを開く必要があります。

リンク

[smx15m5X_OpenDevice](#)

2.7 smx15m5X_SwitchFrontLED

カメラロジックボード前面のLEDのオン/オフを切り換えます。

シンタックス

```
extern BOOL __stdcall smx15m5X_SwitchFrontLED(HANDLE H, UCHAR Mode);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

Mode - UCHAR変数で、LEDのステータスを含んでいます: ON(1)またはOFF(0)

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

このファンクションは、カバーを開けないでカメラのテストをするのに使用されます。

リンク

[smx15m5X_OpenDevice](#)

2.8 smx15m5X_ReadFlash

カメラのフラッシュメモリから指定されたオフセットで指定されたサイズのデータを取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_ReadFlash(HANDLE H, USHORT Offset,  
PUCHAR buffer, USHORT size);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

Offset - USHORT変数で、オフセット値を示します。値の範囲: [0, 32767]

buffer - データを含むUCHAR変数へのポインター。

size - USHORT変数で、読み取り可能なデータのサイズを含んでいます。値の範囲: [1, 32768]

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_WriteFlash](#)

2.9 smx15m5X_WriteFlash

カメラのフラッシュメモリに指定されたオフセットで指定されたサイズのデータを書き込みます。

シンタックス

```
extern BOOL __stdcall smx15m5X_WriteFlash(HANDLE H, USHORT Offset,  
PUCHAR buffer, USHORT size);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

Offset - USHORT変数で、カメラのフラッシュメモリのオフセット値を示します。値の範囲: [0, 32767]

buffer - 書き込まれたデータを受け取るUCHAR変数へのポインター。

size - USHORT変数で、書き込まれた値のサイズを含んでいます。値の範囲: [1, 32768]

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_ReadSensorReg](#)

2.10 smx15m5X_GetCameraInfo

[struct TCameraInfo](#) ストラクチャからカメラ情報を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetCameraInfo(HANDLE H, PCameraInfo  
CameraInfo);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

CameraInfo - データを含む[struct TCameraInfo](#) ストラクチャへのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

[リンク](#)

[smx15m5X_OpenDevice](#), [smx15m5X_GetCameraInfoEx](#), [struct TCameraInfo structure](#)

2.11 smx15m5X_GetCameraInfoEx

[struct TCameraInfoEx](#) ストラクチャからカメラ情報を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetCameraInfoEx(HANDLE H,  
PCameraInfoEx CameraInfoEx);
```

パラメーター:

H - HANDLE変数で、以前に[smx15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

CameraInfoEx - データを含む[struct TCameraInfoEx](#)ストラクチャへのポインター。

返回值:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

[リンク](#)

[smx15m5X_OpenDevice](#), [smx15m5X_GetCameraInfo](#), [struct TCameraInfo structure](#)

2.12 smx15m5X_GetStreamMode

カメラストリームのモードを取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetStreamMode(HANDLE H,  
SMX_StreamMode* StreamMode);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

StreamMode - カメラストリームモードのステータスを受け取るSMX_StreamMode列挙へのポインタ(0: カメラストリームはOFF、1: カメラストリームはON、2: カメラはスナップショットモード、3: カメラはスピードテストモード)

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_SetStreamMode](#), [enum SMX_StreamMode](#)

2.13 smx15m5X_SetStreamMode

カメラストリームのモードを設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetStreamMode(HANDLE H, SMX_StreamMode StreamMode);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

StreamMode - カメラストリームモードのステータスを含むSMX_StreamMode列挙へのポインタ(0: カメラストリームはOFF、1: カメラストリームはON、2: カメラはスナップショットモード、3: カメラはスピードテストモード)

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetStreamMode](#), [enum SMX_StreamMode](#)

2.14 smx15m5X_GetFrame16

ユーザーが指定したメモリバッファ(16ビット/ピクセル)に現在のビデオフレームを取り込みます。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrame16(HANDLE H, PVOID  
Buffer, size_t length);
```

パラメーター:

H - HANDLE変数で、以前にsmx15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

Buffer - PVOIDで、ビデオフレームを受け取るバッファを示します。

length - バッファサイズを含むsize_t変数。

返り値:

成功したらTRUE

失敗したらFALSE

リンク

[smx15m5X_OpenDevice](#)

2.15 smx15m5X_GetFrameEx16

ユーザーが指定したメモリバッファ(16ビット/ピクセル)にフレームパラメータの情報と共に現在のビデオフレームを取り込みます(struct_TFrameInfoストラクチャを参照)。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameEx16(HANDLE H, PVOID  
Buffer, size_t length);
```

パラメーター:

H - HANDLE変数で、以前にsmx-15m5X_OpenDeviceファンクションによって返されたカメラへのハンドルを含んでいます。

Buffer - PVOIDで、ビデオフレームを受け取るバッファを示します。

length - バッファサイズを含むsize_t変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

[リンク](#)

[smx15m5X_OpenDevice](#)

2.16 smx15m5X_GetFrameReady16

ユーザーが指定したメモリバッファ(16ビット/ピクセル)内に最後に用意ができたビデオフレームを取り込みます。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameReady16(HANDLE H, PVOID Buffer, size_t length);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Buffer - PVOIDで、ビデオフレームを受け取るバッファを示します。

length - バッファサイズを含むsize_t変数。

返回值:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

[リンク](#)

[smx15m5X_OpenDevice](#)

2.17 smx15m5X_GetFrameReadyEx16

ユーザーが指定したメモリバッファ(16ビット/ピクセル)にフレームパラメーターの情報と共に最後に用意ができたビデオフレームを取り込みます([struct _TFrameInfo](#)ストラクチャを参照)。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameReadyEx16(HANDLE H, PVOID Buffer, size_t length);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Buffer - PVOIDで、ビデオフレームを受け取るバッファーを示します。

length - バッファーサイズを含むsize_t変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [struct _TFrameInfo](#) structure

2.18 smx15m5X_GetFrame8

ユーザーが指定したメモリバッファー(1ピクセルあたりの8ビット)に現在のビデオフレームを取り込みます。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrame8(HANDLE H, PVOID Buffer, size_t length);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Buffer - PVOIDで、ビデオフレームを受け取るバッファーを示します。

length - バッファーサイズを含むsize_t変数。

返り値:

成功したら *TRUE*

失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#)

2.19 smx15m5X_GetFrameEx8

ユーザーが指定したメモリバッファ(8ビット/ピクセル)にフレームパラメータの情報と共に現在のビデオフレームを取り込みます([struct _TFrameInfo](#)ストラクチャを参照)。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameEx8(HANDLE H, PVOID  
Buffer, size_t length);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Buffer - PVOIDで、ビデオフレームを受け取るバッファを示します。

length - バッファサイズを含むsize_t変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [struct _TFrameInfo](#) structure

2.20 smx15m5X_GetFrameReady8

ユーザーが指定したメモリバッファ(8ビット/ピクセル)内に最後に用意ができたビデオフレームを取り込みます。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameReady8(HANDLE H, PVOID  
Buffer, size_t length);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#)

2.21 smx15m5X_GetFrameReadyEx8

ユーザーが指定したメモリバッファ(8ビット/ピクセル)にフレームパラメータの情報と共に最後に用意ができたビデオフレームを取り込みます([struct _TFramelInfo](#)ストラクチャを参照)。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameReadyEx8(HANDLE H, PVOID Buffer, size_t length);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Buffer - PVOIDで、ビデオフレームを受け取るバッファを示します。

length - バッファサイズを含むsize_t変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [struct _TFramelInfo](#) structure

2.22 smx15m5X_GetFrequency

MHzの単位で現在の周波数を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrequency(HANDLE H, PULONG FrequencyMHz);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

pFrequencyMHz - 周波数値を受け取るULONGLONG変数へのポインター。

返回值:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

有効な周波数範囲: [6..96] MHz

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_SetFrequency](#)

2.23 smx15m5X_SetFrequency

MHzの単位でカメラの周波数を設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetFrequency(HANDLE H, ULONGLONG FrequencyMHz);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

FrequencyMHz - 周波数値を含むULONGLONG変数。

返回值:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

有効な周波数範囲: [6..96] MHz

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetFrequency](#), [smx15m5X_GetFrequencyD](#), [smx15m5X_SetFrequencyD](#)

2.24 smx15m5X_GetFrequencyD

MHzの単位でダブル変数として現在の周波数を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrequencyD(HANDLE H, double*  
Frequen- cyMHz);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

pFrequencyMHz - 周波数値を受け取るダブル変数へのポインタ。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

有効な周波数範囲: [6..96] MHz

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_SetFrequency](#), [smx15m5X_GetFrequency](#),
[smx15m5X_GetFrequencyD](#)

2.25 smx15m5X_SetFrequencyD

MHzの単位でカメラ周波数を設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetFrequencyD(HANDLE H, double FrequencyMHz);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

FrequencyMHz - 周波数値を含むダブル変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら **FALSE**

備考

有効な周波数範囲: [6..96] MHz

このファンクションは、ダブルタイプの周波数を変換して、**ULONG**として周波数を設定します。周波数を整数でない値に設定した場合は、小数点以下の値が欠落します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetFrequency](#), [smx15m5X_SetFrequency](#), [smx15m5X_GetFrequencyD](#)

2.26 smx15m5X_GetFrameParams

フレームパラメーター(開始X、開始Yの位置、幅、高さ、デシメーション、カラー深度、垂直および水平鏡像反転)を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameParams(HANDLE H,  
TFrameParams* FrameParams);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

FrameParams - データを含む[struct TFrameParams](#)ストラクチャへの32ビットポインター。

返回值:

ファンクションが成功したら **TRUE**

ファンクションが失敗したら **FALSE**

リンク

[smx15m5X_OpenDevice](#), [struct TFrameParamsstructure](#), [smx15m5X_SetFrameParams](#)

2.27 smx15m5X_SetFrameParams

フレームパラメーター(開始X、開始Yの位置、幅、高さ、デシメーション、カラー深度、垂直および水平鏡像反転)を設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetFrameParams(HANDLE H,  
TFrameParams* FrameParams);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

FrameParams - データを含む[struct TFrameParams](#)ストラクチャへの32ビットポインター。

返り値:

成功したら *TRUE*

失敗したら *FALSE*

備考

入力パラメーターの少なくとも1つが無効であれば、ファンクションは実行されないで(*FALSE*を返します)、すべてのパラメーターは変化しません。

現在のフレームパラメーターをチェックするには[smx15m5X_GetFrameParams](#)ファンクションを使用します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetFrameParams](#)

2.28 smx15m5X_GetGain

4つのチャンネル(G1、R、G2、B)のゲイン値を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetGain(HANDLE H, PLONG G1, PLONG  
R, PLONG G2, PLONG B);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

G1 - Green1 (G1)チャンネルのゲインを受け取るLONG変数へのポインター。

R - Red (R)チャンネルのゲインを受け取るLONG変数へのポインター。

G2 - Green2 (G2)チャンネルのゲインを受け取るLONG変数へのポインター。

B - Blue (B)チャンネルのゲインを受け取るLONG変数へのポインター。

返回值:

ファンクションが成功したら **TRUE**

ファンクションが失敗したら **FALSE**

備考

値の範囲: [0..160]

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_SetGain](#), [smx15m5X_SetAllGain](#)

2.29 smx15m5X_SetGain

4つのカラーチャンネル(G1、R、G2、B)のゲイン値を設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetGain(HANDLE H, LONG G1, LONG R,  
LONG G2, LONG B);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

G1 - Green1 (G1)チャンネルのゲインを含むLONG変数。

R - Red (R)チャンネルのゲインを含むLONG変数。

G2 - Green2 (G2)チャンネルのゲインを含むLONG変数。

B - Blue (B)チャンネルのゲインを含むLONG変数。

返回值:

ファンクションが成功したら **TRUE**

ファンクションが失敗したら **FALSE**

備考

値の範囲: すべてのチャンネル: [0..160]

範囲外のゲインを設定しようとする、ファンクションは値0を設定します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetGain](#), [smx15m5X_SetAllGain](#)

2.30 smx15m5X_SetAllGain

すべてのチャンネルに同じゲインを設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetAllGain(HANDLE H, LONG Gain);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Gain - ゲイン値を含むLONG変数

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

ファンクション[smx15m5X_SetAllGain](#)はすべてのチャンネル(G1、R、G2、B)のゲインを同じ値に設定して、トータルゲイン値と同じにします。

値の範囲: [0..160]

範囲外のゲインを設定しようとする、ファンクションは値0を設定します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetGain](#), [smx15m5X_SetGain](#)

2.31 smx15m5X_GetExposureMinMax

行の露光の値の範囲を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetExposureMinMax(HANDLE H, PULONG  
ExpMin, PULONG ExpMax);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

ExpMin - 行の最小露光値を受け取るULONG変数へのポインター。

ExpMax - 行の最大露光値を受け取るULONG変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetExposureFramerateMax](#), [smx-15m5X_GetExposure](#), [smx15m5X_SetExposure](#)

2.32 smx15m5X_GetExposureFramerateMax

現在の周波数とビューポートサイズで可能な最大のフレームレートを維持して設定できる露光の最大値を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetExposureFramerateMax(HANDLE H,  
PULONG ExpMax);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

ExpMax - 現在の周波数とビューポートサイズで可能な最大のフレームレートを維持して設定できる露光の最大値を取得する **ULONG** 変数へのポインター。

返り値:

ファンクションが成功したら **TRUE**

ファンクションが失敗したら **FALSE**

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetExposureMinMax](#), [smx15m5X_GetExposure](#), [smx15m5X_SetExposure](#)

2.33 smx15m5X_GetExposure

行の現在の露光値を読み出します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetExposure(HANDLE H, PULONG Exposure);
```

パラメーター:

H - **HANDLE** 変数で、以前に [smx-15m5X_OpenDevice](#) ファンクションによって返されたカメラへのハンドルを含んでいます。

Exposure - 行の現在の露光値を受け取る **ULONG** 変数へのポインター。

返り値:

ファンクションが成功したら **TRUE**

ファンクションが失敗したら **FALSE**

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetExposureMinMax](#), [smx15m5X_GetExposureFramerateMax](#), [smx15m5X_SetExposure](#)

2.34 smx15m5X_SetExposure

行の露光値を設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetExposure(HANDLE H, ULONG Exposure);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Exposure - 行の露光値を含むULONG変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

値の範囲: [1..1048575]

範囲外の露光を設定しようとすると、ファンクションは最も近い値を設定します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetExposureMinMax](#), [smx15m5X_GetExposureFramerateMax](#), [smx15m5X_GetExposure](#)

2.35 smx15m5X_ GetExposureMinMaxMs

現在の周波数とビューポートサイズでミリ秒の単位で露光値の範囲を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetExposureMinMaxMs(HANDLE H,  
double* Exp- Min, double* ExpMax);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

ExpMin - ミリ秒の単位で最小露光値を受け取るダブル変数へのポインター。

ExpMax - ミリ秒の単位で最大露光値を受け取るダブル変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetExposureFramerateMaxMs](#), [smx-15m5X_GetExposureMs](#), [smx15m5X_SetExposureMs](#)

2.36 smx15m5X_GetExposureFramerateMaxMs

現在の周波数とビューポートサイズで可能な最大のフレームレートを維持して設定できるミリ秒の単位での最大露光値を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetExposureFramerateMaxMs(HANDLE H,  
double* Exposure);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Exposure - 現在の周波数とビューポートサイズで可能な最大のフレームレートを維持して設定できるミリ秒の単位での最大露光値を受け取るダブル変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetExposureFramerateMaxMs](#), [smx-15m5X_GetExposureMs](#), [smx15m5X_SetExposureMs](#)

2.37 smx15m5X_GetExposureMs

ミリ秒の単位で現在の露光値を読み出します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetExposureMs(HANDLE H, double* Exposure);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Exposure - ミリ秒の単位で現在の露光値を受け取るダブル変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetExposureMinMaxMs](#), [smx15m5X_GetExposureFramerateMaxMs](#), [smx15m5X_SetExposureMs](#)

2.38 smx15m5X_SetExposureMs

ミリ秒の単位でカメラ露光を設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetExposureMs(HANDLE H, double Exposure, double* ExposureSet);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Exposure - ミリ秒の単位での露光値を含むダブル変数。

ExposureSet - ミリ秒の単位での設定露光値を受け取るダブル変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら **FALSE**

備考

露光(ms)の値の範囲を取得するには [smx15m5X_GetExposureMinMaxMs](#) ファンクションを使用します。

範囲外の露光を設定しようとする、ファンクションは最も近い値を設定します。ファンクションは指定された値を設定することができず、その値よりも小さい次の可能な値が設定されます。設定された値をチェックするには [smx15m5X_GetExposureMs](#) ファンクションを使用します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetExposureMinMaxMs](#), [smx15m5X_GetExposureFramerateMaxMs](#), [smx15m5X_GetExposureMs](#)

2.39 smx15m5X_GetFrameRateMinMax

現在のカメラパラメーターでのフレームレート(フレーム/秒)の値の範囲を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameRateMinMax(HANDLE H, double* FrameRateMin, double* FrameRateMax);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#) ファンクションによって返されたカメラへのハンドルを含んでいます。

FrameRateMin - 最小フレームレート値を受け取るダブル変数へのポインター。

FrameRateMax - 最大フレームレート値を受け取るダブル変数へのポインター。

返り値:

ファンクションが成功したら **TRUE**

ファンクションが失敗したら **FALSE**

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetFrameRate](#), [smx15m5X_SetFrameRate](#), [smx15m5X_SetMinFrameRate](#), [smx15m5X_SetMaxFrameRate](#)

2.40 smx15m5X_GetFrameRate

現在のフレームレートを取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetFrameRate( HANDLE H, double* FrameRate);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

FrameRate - 現在のフレームレートを受け取るダブル変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetFrameRateMinMax](#), [smx15m5X_SetFrameRate](#), [smx15m5X_SetMinFrameRate](#), [smx15m5X_SetMaxFrameRate](#)

2.41 smx15m5X_SetFrameRate

カメラのフレームレートを設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetFrameRate( HANDLE H, double FrameRate);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

FrameRate - フレームレート値を含むダブル変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

現在のフレームサイズと周波数でのフレームレートの値の範囲を取得するには [smx-15m5X_GetFrameRateMinMax](#) フังก์ションを使用します。

範囲外のフレームレートを設定しようとする、ファンクションは最も近い値を設定します。ファンクションは指定された値を正確に設定しない場合があるので、設定されたフレームレートをチェックするには [smx15m5X_GetFrameRate](#) フังก์ションを使用します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetFrameRateMinMax](#), [smx15m5X_GetFrameRate](#), [smx15m5X_SetMinFrameRate](#), [smx15m5X_SetMaxFrameRate](#)

2.42 smx15m5X_SetMinFrameRate

現在の周波数とビューポートサイズでの最小のフレームレートを設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetMinFrameRate(HANDLE H);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

設定されたフレームレートをチェックするには[smx15m5X_GetFrameRate](#)ファンクションを使用します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetFrameRateMinMax](#), [smx15m5X_GetFrameRate](#), [smx15m5X_SetFrameRate](#), [smx15m5X_SetMaxFrameRate](#)

2.43 smx15m5X_SetMaxFrameRate

現在の周波数とビューポートサイズでの最大のフレームレートを設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetMaxFrameRate(HANDLE H);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

設定されたフレームレートをチェックするには[smx15m5X_GetFrameRate](#)ファンクションを使用します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetFrameRateMinMax](#), [smx15m5X_GetFrameRate](#), [smx15m5X_SetFrameRate](#), [smx15m5X_SetMinFrameRate](#)

2.44 smx15m5X_GetRowBlackLevel

生のブラックレベルの較正值を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetRowBlackLevel(HANDLE H, PUSHORT Level);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Level - 生のブラックレベルの較正值を受け取るUSHORT変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetRowBlackLevel](#)

2.45 smx15m5X_SetRowBlackLevel

生のブラックレベルの較正值を設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetRowBlackLevel(HANDLE H, USHORT Level);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Level - 生のブラックレベルの較正值を含むUSHORT変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

生のブラックレベルの値の範囲: [0..4095]

設定された生のブラックレベルの値をチェックするには
[smx15m5X_GetRowBlackLevel](#)ファンクションを使用します。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetRowBlackLevel](#)

2.46 smx15m5X_GetGlobalResetRelease

グローバルリセットリリースシャッターモードのステータス(有効または無効)を取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetGlobalResetRelease(HANDLE H, BOOL* Level);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Level - グローバルリセットリリースシャッターモードのステータス(ONまたはOFF)を受け取るBOOL変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_SetGlobalResetRelease](#)

2.47 smx15m5X_SetGlobalResetRelease

グローバルリセットリリースシャッターモードを設定/キャンセルします。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetGlobalResetRelease(HANDLE H, BOOL Level);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Level - グローバルリセットリリースシャッターモードのステータス(ONまたはOFF)を含むBOOL変数

返回值:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

備考

グローバルリセットリリースの現在のステータスをチェックするには[smx15m5X_GetGlobalResetRelease](#)ファンクションを使用します。

グローバルリセットシャッターモードはスナップショットモードでのみ利用可能です。

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetGlobalResetRelease](#), [smx15m5X_SetStreamMode](#)

2.48 smx15m5X_GetSnapshotParams

カメラのスナップショットパラメーターを取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetSnapshotParams(HANDLE H,  
PULONG Time- outMs, PBOOLEAN Inverted);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

TimeoutMs - ミリ秒の単位で外部シグナルの待ち時間を受け取る ULONG変数へのポインター。

Inverted - 入力シグナルの極性(反転または非反転)を受け取る BOOLEAN変数へのポインター。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_SetSnapshotParams](#)

2.49 smx15m5X_SetSnapshotParams

カメラのスナップショットパラメーターを設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetSnapshotParams(HANDLE H, ULONG  
TimeoutMs, BOOLEAN Inverted);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

TimeoutMs - ミリ秒の単位で外部シグナルの待ち時間を含む ULONG変数。

Inverted - 入力シグナルの極性(反転または非反転)を含む BOOLEAN変数。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

[リンク](#)

[smx15m5X_OpenDevice](#), [smx15m5X_GetSnapshotParams](#)

2.50 smx15m5X_GetStrobeParams

カメラの出力シグナルのパラメーターを取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetStrobeParams(HANDLE H,  
PSTROBE_PARAMS StrobeStart, PSTROBE_PARAMS StrobeEnd,  
PBOOLEAN Inverted, PBOOLEAN Enabled);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

StrobeStart - ストロボパルスの開始を受け取るSTROBE_PARAMS列挙へのポインター。

StrobeEnd - ストロボパルスの終了を受け取るSTROBE_PARAMS列挙へのポインター。

Inverted - 出力シグナルの極性(反転または非反転)を受け取るBOOLEAN変数へのポインター。

Enabled - ストロボモードのステータス(有効または無効)を受け取るBOOLEAN変数へのポインター。FALSEの場合、汎用出力モードが有効になります。

返り値:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

[リンク](#)

[smx15m5X_OpenDevice](#), [smx15m5X_SetStrobeParams](#), [enum STROBE_PARAMS](#)

2.51 smx15m5X_SetStrobeParams

カメラの出力シグナルのパラメーターを設定します。

シンタックス


```
extern BOOL __stdcall smx15m5X_SetStrobeParams(HANDLE H,  
STROBE_PARAMS StrobeStart, STROBE_PARAMS StrobeEnd, BOOLEAN  
Inverted, BOOLEAN Enabled);
```

パラメーター:

H - HANDLE変数で、以前に**smx15m5X_OpenDevice**ファンクションによって返されたカメラへのハンドルを含んでいます。

StrobeStart - ストロボパルスの開始を含む**STROBE_PARAMS**列挙。

StrobeEnd - ストロボパルスの終了を含む**STROBE_PARAMS**列挙。

Inverted - 出力シグナルの極性(反転または非反転)を含む**BOOLEAN**変数。

Enabled - ストロボモードのステータス(有効または無効)を含む**BOOLEAN**変数。FALSEの場合、汎用出力モードが有効になります。

戻り値:

ファンクションが成功したら **TRUE**

ファンクションが失敗したら **FALSE**

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetStrobeParams](#), [enum STROBE_PARAMS](#)

2.52 smx15m5X_GetEventsState

カメラのソフトウェアイベントビットを取得します。

シンタックス

```
extern BOOL __stdcall smx15m5X_GetEventsState(HANDLE H, PCHAR Bit);
```

パラメーター:

H - HANDLE変数で、以前に**smx-15m5X_OpenDevice**ファンクションによって返されたカメラへのハンドルを含んでいます。

Bit - ソフトウェアイベントビット(1または0)を受け取る**UCHAR**変数へのポインター。

戻り値:

ファンクションが成功したら **TRUE**

ファンクションが失敗したら **FALSE**

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_SetEventsState](#), [smx15m5X_CheckEventsState](#)

2.53 smx15m5X_SetEventsState

カメラのソフトウェアイベントビットを設定します。

シンタックス

```
extern BOOL __stdcall smx15m5X_SetEventsState(HANDLE H, UCHAR  
Value, UCHAR Mask);
```

パラメーター:

H - HANDLE変数で、以前に[smx-15m5X_OpenDevice](#)ファンクションによって返されたカメラへのハンドルを含んでいます。

Value - イベント値を含むUCHAR変数。

Mask - マスク値を含むUCHAR変数。

返回值:

ファンクションが成功したら *TRUE*

ファンクションが失敗したら *FALSE*

リンク

[smx15m5X_OpenDevice](#), [smx15m5X_GetEventsState](#), [smx15m5X_CheckEventsState](#)

2.54 smx15m5X_CheckEventsState

フレームのソフトウェアイベントビットをチェックします。

シンタックス

```
extern LONG __stdcall smx15m5X_CheckEventsState(PFrameInfo FrameInfo,  
UCHAR Value, UCHAR Mask);
```

パラメーター:

FrameInfo - [struct _TFrameInfo](#)ストラクチャへのポインター。

Value - イベント値を含むUCHAR変数。

Mask - マスク値を含むUCHAR変数。

返回值:

ファンクションは、ソフトウェアイベントステートビットが同じステートにあるラインの最初のナンバーを返します。

ファンクションがソフトウェアイベントステートビットを検出しなかった場合は、返回值は-1になります。

リンク

[smx15m5X_OpenDevice](#), [struct _TFramelInfo](#) structure, [smx15m5X_GetEventsState](#), [smx15m5X_SetEventsState](#)