

レーザーキャナライブラリ マニュアル

このマニュアルは、レーザーキャナライブラリの仕様について記載します。

ご注意

- このソフトウェアおよびマニュアルの、一部または全部を無断で使用、複製することはできません。
- このソフトウェアおよびマニュアルは、本製品の使用許諾契約書のもとでのみ使用することができます。
- このソフトウェアおよびマニュアルを運用した結果の影響については、一切の責任を負いかねますのでご了承ください。
- このソフトウェアの仕様、およびマニュアルに記載されている事柄は、将来予告なしに変更することがあります。
- このマニュアルの著作権はカシオ計算機株式会社に帰属します。
- 本書中に含まれている画面表示は、実際の画面とは若干異なる場合があります。予めご了承ください。

© 2016 カシオ計算機株式会社

Microsoft, MS, ActiveSync, Active Desktop, Outlook, Windows, Windows NT, および Windows ロゴは、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。Microsoft 社の製品は、OEM 各社に、Microsoft Corporation の 100%出資子会社である Microsoft Licensing, Inc.によりライセンス供与されています。

変更履歴

バージョン	変更日付	ページ	内容
1.00	2008.11		新規作成
1.01	2009.03	2	開発環境とプログラミング言語の対応表を訂正
		8	OBROpen 関数のパラメータ説明を追加
		43	OBRSetScanningTimeout 関数の解説の内容を追加
		47	OBRSetScanningType 関数の解説の内容を追加
		53	OBRStartScanning 関数のパラメータ説明を訂正
		67	OBRSetNW7OptionEx 関数のパラメータ説明を訂正
		69	OBRGetNW7Option 関数のパラメータ説明を訂正
		73	OBRSetWPCAddonOptionEx 関数のパラメータ説明を訂正
		75	OBRGetWPCAddonOption 関数のパラメータ説明を訂正
		79	OBRSetWPCOptionEx 関数のパラメータ説明を訂正
		81	OBRGetWPCOption 関数のパラメータ説明を訂正
		91	OBRSetUPCEOptionEx 関数のパラメータ説明を訂正
		101	OBRGetITFOption 関数のパラメータ説明を訂正
		105	OBRGetCode93Option 関数のパラメータ説明を訂正
		109	OBRGetCode128Option 関数のパラメータ説明を訂正
		113	OBRGetMSIOption 関数のパラメータ説明を訂正
115	OBRSetIATAOption 関数のパラメータ説明を訂正		
117	OBRGetIATAOption 関数のパラメータ説明を訂正		
1.02	2009.07	—	対象機種に DT-5300 を追加
1.03	2009.11	17	OBRSetSwingAngle 関数の注意を追加
		145	OBRSetGainControl 関数のパラメータおよび注意を追加
		158-171	設定ファイルの書式にパラメータを追加
1.04	2010.06	41	OBRSetCheckCounter 関数にリニアイメージモデルの記述を追加
		158	設定ファイルの書式にリニアイメージモデルの記述を追加
1.05	2011.01	—	対象機種に IT-300 および DT-X8 を追加
1.06	2011.09	55	OBRSetDecodeLevel 関数のパラメータを追加
1.07	2014.11	—	対象機種に IT-G500 を追加
		—	関数一覧表をデバイスライブラリ基本マニュアルへ移動
		8/41/61/65 /67/71/73/ 77/79/83/8 5/89/91/95 /99/103/10 7/111/115	IT-G500 の注意事項を追加

目次

1. 概要	1
2. 動作環境	2
3. 構造体	4
3.1 M_TBL	5
4. 関数	8
4.1 OBROpen	8
4.2 OBRClose	10
4.3 OBRSetLED	11
4.4 OBRGetLED	12
4.5 OBRSetBuzzer	13
4.6 OBRGetBuzzer	14
4.7 OBRSetVibrator	15
4.8 OBRGetVibrator	16
4.9 OBRSetSwingAngle	17
4.10 OBRGetSwingAngle	19
4.11 OBRSetFocus	20
4.12 OBRGetFocus	21
4.13 OBRSetScanningNotification	22
4.14 OBRGetScanningNotification	24
4.15 OBRGetLastEventStatus	25
4.16 OBRSetScanningKey	26
4.17 OBRGetScanningKey	27
4.18 OBRSetAllOptions	28
4.19 OBRGetAllOptions	29
4.20 OBRSetDefaultSymbology	30
4.21 OBRSetDefaultSymbologyEx	31
4.22 OBRSaveLogFile	32
4.23 OBRGetc	33
4.24 OBRGets	35
4.25 OBRGetStatus	37
4.26 OBRClearBuff	38
4.27 OBRSetBuffType	39
4.28 OBRGetBuffType	40
4.29 OBRSetCheckCounter	41
4.30 OBRGetCheckCounter	42
4.31 OBRSetScanningTimeout	43
4.32 OBRGetScanningTimeout	44
4.33 OBRSetSuffixChar	45
4.34 OBRGetSuffixChar	46
4.35 OBRSetScanningType	47
4.36 OBRGetScanningType	48
4.37 OBRSetScanningCounter	49
4.38 OBRGetScanningCounter	50
4.39 OBRSetMultiStepReading	51
4.40 OBRGetMultiStepReading	52

4.41	OBRStartScanning	53
4.42	OBRStopScanning	54
4.43	OBRSetDecodeLevel	55
4.44	OBRGetDecodeLevel	57
4.45	OBRSetScanningCode	58
4.46	OBRGetScanningCode	60
4.47	OBRSetCode39Option	61
4.48	OBRGetCode39Option	63
4.49	OBRSetNW7Option	65
4.50	OBRSetNW7OptionEx	67
4.51	OBRGetNW7Option	69
4.52	OBRSetWPCAddonOption	71
4.53	OBRSetWPCAddonOptionEx	73
4.54	OBRGetWPCAddonOption	75
4.55	OBRSetWPCOption	77
4.56	OBRSetWPCOptionEx	79
4.57	OBRGetWPCOption	81
4.58	OBRSetUPCEAddonOption	83
4.59	OBRSetUPCEAddonOptionEx	85
4.60	OBRGetUPCEAddonOption	87
4.61	OBRSetUPCEOption	89
4.62	OBRSetUPCEOptionEx	91
4.63	OBRGetUPCEOption	93
4.64	OBRSetIDFOption	95
4.65	OBRGetIDFOption	97
4.66	OBRSetITFOption	99
4.67	OBRGetITFOption	101
4.68	OBRSetCode93Option	103
4.69	OBRGetCode93Option	105
4.70	OBRSetCode128Option	107
4.71	OBRGetCode128Option	109
4.72	OBRSetMSIOption	111
4.73	OBRGetMSIOption	113
4.74	OBRSetIATAOption	115
4.75	OBRGetIATAOption	117
4.76	OBRSetRSS14Option	119
4.77	OBRGetRSS14Option	121
4.78	OBRSetRSSLimitedOption	123
4.79	OBRGetRSSLimitedOption	125
4.80	OBRSetRSSExpandedOption	127
4.81	OBRGetRSSExpandedOption	129
4.82	OBRSetRSS14StackedOption	131
4.83	OBRGetRSS14StackedOption	133
4.84	OBRSetRSSExpandedStackedOption	135
4.85	OBRGetRSSExpandedStackedOption	137
4.86	OBRSaveConfigFile	139
4.87	OBRLoadConfigFile	140
4.88	OBRSetFilter	141
4.89	OBRGetFilter	142
4.90	OBRSetFilterOnTimer	143

4.91	OBRGetFilterOnTimer	144
4.92	OBRSetGainControl	145
4.93	OBRGetGainControl	147
4.94	OBRSetBarWidthAdjustment	148
4.95	OBRGetBarWidthAdjustment	150
4.96	OBRSetMarginCheckRatio	151
4.97	OBRGetMarginCheckRatio	152
4.98	OBRSetDecodeLearningMode	153
4.99	OBRGetDecodeLearningMode	154
5.	プログラミング上の注意点	155
5.1	読み取り完了の通知について	155
5.2	設定ファイルの書式	158

1. 概要

レーザスキャナライブラリは、本体に搭載されているレーザスキャナを使用して、バーコードのスキャンやデコードを行う関数を提供します。

レーザスキャナクラスライブラリは、レーザスキャナライブラリを **.NET Compact Framework** アプリケーションから直接利用できるようにする、ラッパーライブラリです。

レーザスキャナライブラリを使用することにより、機種を意識することなく、アプリケーションのソースコード互換性を高めることができます。

レーザスキャナライブラリでは、機種を問わず、すべての関数を用意し、アプリケーションから見た「仮想マシン」としての振る舞いを提供します。

レーザスキャナライブラリの各関数は、アプリケーションからの要求に対して、対象のデバイス機能が制御できない場合は、「未サポートエラー」を返します。また搭載デバイスの機能差によって利用できないパラメータを設定した場合は、「パラメータエラー」を返します。

※ レーザスキャナライブラリは、アプリケーションのソースコード互換性の向上を目的としたライブラリであり、搭載デバイスの機能互換性を保障するものではありません。
「未サポートエラー」および「パラメータエラー」を正しく判定し、操作者に対して機能が未サポートである旨を通知する、あるいは処理そのものを無効としてください。

2. 動作環境

レーザスキャナライブラリの動作環境を以下に示します。

対象機種

DT-5200 / DT-X7 / DT-9800 / DT-5300 / IT-300 / DT-X8 / IT-G500 / DT-X100 / DT-X200

対象 OS

- Microsoft Windows CE 5.0
- Microsoft Windows CE 6.0
- Microsoft Windows Embedded Compact 7
- Microsoft WindowsMobile 6.5
- Microsoft WindowsMobile 6.5.3
- Microsoft Windows Embedded Handheld 6.5

開発環境とプログラミング言語

表 2-1

開発環境	Visual C++	Visual Basic, Visual C#
Microsoft embedded Visual C++ Version 4.0 + SP4	○	-
Microsoft Visual Studio.NET 2003 + SP1	×	○
Microsoft Visual Studio 2005 + SP1	○	○
Microsoft Visual Studio 2008	○	○

(○:利用可、×:利用不可、-:機能なし)

提供ファイル

表 2-2

ファイル	Visual C++	Visual Basic, Visual C#
OBReadLib.h	○	-
OBReadLib.lib	○	-
OBReadLib.dll	○	○
OBReadLibNet.dll (クラスライブラリ)	-	○

(○:必要、-:不要)

使用方法

Visual C++ の場合

- プログラムソース内に OBReadLib.h をインクルードし、リンカの依存ファイルとして OBReadLib.lib を指定してください
- OBReadLib.dll は本体に内蔵されています。

Visual Basic または Visual C# の場合

- OBReadLibNet.dll をプロジェクトの参照に追加してください。
- OBReadLib.dll は本体に内蔵されています。
- OBReadLibNet.dll を実行モジュールと同じフォルダにコピーしてください。

名前空間とクラス

クラスライブラリ OBReadLibNet.dll では、関数および定数の参照用として、下記のクラスが用意されています。

表 2-3

名前空間	クラス名	内容
CaLib	OBReadLibNet.Api	関数参照用クラス
	OBReadLibNet.Def	定数参照用クラス

クラス定義の詳細については、Microsoft Visual Studio で OBReadLibNet.dll を参照設定し、オブジェクトブラウザで確認してください。

3. 構造体

レーザスキャナライブラリで提供する構造体を以下に示します。

表 3-1. 構造体一覧

構造体名	内容	使用する関数
M_TBL	スキャナドライバの全ての設定情報を格納する構造体	OBRGetAllOptions OBRSetAllOptions

3.1 M_TBL

スキヤナドライバの全ての設定情報を格納します。

```
typedef struct m_tbl{
    DWORD      Code;
    BYTE       Cd39[6];
    BYTE       Nw7[6];
    BYTE       Wpcea[6];
    BYTE       Wpce[6];
    BYTE       Upcea[6];
    BYTE       Upce[6];
    BYTE       Idsf[6];
    BYTE       Itrf[6];
    BYTE       Cd93[6];
    BYTE       Cd128[6];
    BYTE       Msi[6];
    BYTE       Iata[6];
    BYTE       RSS-14[6];
    BYTE       RssLtd[6];
    BYTE       RssExp[6];
    BYTE       Rss14S[6];
    BYTE       RssExpS[6];
    BYTE       Resv[14][6];
    BYTE       Itrfs[2];
    BYTE       Nw7s[2];
    BYTE       Cd39s[2];
    BYTE       Type;
    BYTE       Gain;
    BYTE       Buzc;
    BYTE       Ledc;
    BYTE       Bufc;
    BYTE       Endc;
    BYTE       Mode;
    BYTE       Dumy;
    int        cmp_ct;
    int        rd_ct;
    int        scn_tm;
    DWORD      dwFilterMode;
    DWORD      dwFilterCnt;
    DWORD      dwLearning;
    DWORD      dwVibc;
    DWORD      dwScanningKey;
    DWORD      dwSwing;
    DWORD      dwFocus;
    DWORD      dwNotification;
    DWORD      dwDecodeLevel;
} M_TBL;
```

表 3-2 M_TBL 構造体メンバ変数一覧

メンバ	説明	同じ設定をセットできる関数
Code	読み取り有効/無効コード指定	OBRSetScanningCode
Cd39[6]	Code39 読み取り設定(※1)	OBRSetCode39Option
Nw7[6]	NW-7 読み取り設定(※1)	OBRSetNW7Option OBRSetNW7OptionEx
Wpcea[6]	WPC Addon 読み取り設定(※1)(※3)	OBRSetWPCAddonOption OBRSetWPCAddonOptionEx
Wpce[6]	WPC 読み取り設定(※1)(※3)	OBRSetWPCOption OBRSetWPCOptionEx
Upcea[6]	UPC-E Addon 読み取り設定(※1)(※4)	OBRSetUPCEAddonOption OBRSetUPCEAddonOptionEx
Upce[6]	UPC-E 読み取り設定(※1)(※4)	OBRSetUPCEOption OBRSetUPCEOptionEx
Idsf[6]	IDF 読み取り設定(※1)	OBRSetIDFOption
Itrf[6]	ITF 読み取り設定(※1)	OBRSetITFOption
Cd93[6]	Code93 読み取り設定(※1)	OBRSetCode93Option
Cd128[6]	Code128 読み取り設定(※1)	OBRSetCode128Option
Msi[6]	MSI 読み取り設定(※1)	OBRSetMSIOption
Iata[6]	IATA 読み取り設定(※1)	OBRSetIATAOption
Rss14[6]	RSS-14 読み取り設定(※1)(※2)	OBRSetRSS14Option
Rssltd[6]	RSS Limited 読み取り設定(※1)(※2)	OBRSetRSSLimitedOption
RssExp[6]	RSS Expanded 読み取り設定(※1)(※2)	OBRSetRSSExpandedOption
Rss14S[6]	RSS-14 Stacked 読み取り設定(※1)(※2)	OBRSetRSS14StackedOption
RssExpS[6]	RSS Expanded Stacked 読み取り設定(※1)(※2)	OBRSetRSSExpandedStackedOption
Resv[14][6]	リザーブ	—
Itrfs[2]	リザーブ	—
Nw7s[2]	リザーブ	—
Cd39s[2]	リザーブ	—
Type	読み取り方式設定	OBRSetScanningType
Gain	Gain 設定	OBRGetGainControl
Buzc	ブザー設定	OBRSetBuzzer
Ledc	LED 設定	OBRSetLED
Bufc	出力方式設定	OBRSetBuffType
Endc	終了コード設定	OBRSetSuffixChar
Mode	段数読み設定	OBRSetMultiStepReading
Dumy	リザーブ	—
cmp_ct	照合回数設定	OBRSetCheckCounter
rd_ct	読み取り回数設定	OBRSetScanningCounter
scn_tm	タイムアウト設定	OBRSetScanningTimeout
dwFilterMode	ノイズフィルタモードの設定	OBRSetFilter
dwFilterCnt	読み取り開始から、ノイズフィルタを開始するまでの時間の設定	OBRSetFilterOnTimer
dwLearning	デコーダ学習機能設定	OBRSetDecodeLearningMode
dwVibc	バイブレータ設定	OBRSetVibrator

dwScanningKey	スキャン開始キー設定	OBRSetScanningKey
dwSwing	振れ角モード設定	OBRSetSwingAngle
dwFocus	レーザフォーカス設定	OBRSetFocus
dwNotification	読み取り完了のイベント通知設定	OBRSetScanningNotification
dwDecodeLevel	デコードレベル設定	OBRSetDecodeLevel

※ 1 配列の内容は下記のとおりです。

- [0] リザーブ
- [1] 最小桁数
- [2] 最大桁数
- [3] 出力フォーマット
- [4] チェックデジット計算設定
- [5] チェックキャラクタ出力設定

※ 2 RSS コードは 2007 年 2 月に GS1 DataBar に名称変更されています。

- RSS-14・・・GS1 DataBar Omnidirectional
- RSS Limited・・・GS1 DataBar Limited
- RSS Expanded・・・GS1 DataBar Expanded
- RSS-14 Stacked・・・GS1 DataBar Stacked
- RSS Expanded Stacked・・・GS1 DataBar Expanded Stacked

※ 3 WPC Addon または WPC のチェックデジット計算設定を無効に設定する場合、
WPC Addon、WPC 双方のチェックデジット計算設定を無効に設定する必要があります。

※ 4 UPC-E Addon または UPC-E のチェックデジット計算設定を無効に設定する場合、
UPC-E Addon、UPC-E 双方のチェックデジット計算設定を無効に設定する必要があります。

4. 関数

4.1 OBROpen

読み取り許可状態を設定します。

```
[C++]
int OBROpen(
    HWND hWnd,
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBROpen( _
    ByVal hWnd As IntPtr, _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBROpen(
    IntPtr hWnd,
    Int32 dwMode
);
```

解説

レーザスキャナドライバを、読み取り許可状態 (Open 状態) にします。この状態は **OBRClose** 関数を実行するまで有効です。Open 状態時に読み取り開始キーを押すと、スキャンを開始します。レーザスキャナドライバが Open 状態のときは、動作設定や読み取り設定の変更は行えません。設定を変更するには、1 度 **OBRClose** 関数を呼び出して Close 状態にしてから行ってください。Open 状態の時は、読み取り開始キーはレーザスキャナドライバにより占有されるため、ユーザは使用することができません。

パラメータ

hWnd

アプリケーションウィンドウハンドルを指定します。デコードデータの出力方式を「キーバッファ出力」方式に設定した場合、指定されたウィンドウハンドルに出力します。ウィンドウハンドルを持たないアプリケーションで呼び出す場合、出力モードを「キーバッファ出力」方式以外のモードに設定し、このパラメータに **NULL** を指定してください。出力方式は **OBRSetBuffType** 関数により設定することができます。

dwMode

各コードの読み取り許可・禁止を設定します。以下の値の論理和で指定します。**0** を指定すると、現在セットされている設定のままで Open 状態にします。**0** 以外の値を設定した場合、一度全ての設定をデフォルトの設定にセットしてから、指定された値の設定を反映します。

OBR_CD39	: Code39 読み取り許可
OBR_NW_7	: NW-7 読み取り許可
OBR_WPCA	: WPC Addon 読み取り許可
OBR_WPC	: WPC 読み取り許可
OBR_UPEA	: UPC-E Addon 読み取り許可

OBR_UPE	: UPC-E 読み取り許可
OBR_IDF	: Industrial 2 of 5 読み取り許可
OBR_ITF	: Interleaved 2 of 5 読み取り許可
OBR_CD93	: Code93 読み取り許可
OBR_CD128	: Code128 読み取り許可
OBR_MSI	: MSI 読み取り許可
OBR_IATA	: IATA 読み取り許可
OBR_RSS14	: RSS-14 読み取り許可
OBR_RSSLTD	: RSS Limited 読み取り許可
OBR_RSSEXP	: RSS Expanded 読み取り許可
OBR_RSS14S	: RSS-14 Stacked 読み取り許可
OBR_RSSEXPS	: RSS Expanded Stacked 読み取り許可)
OBR_ALL	: 全コードに対して、読み取り許可
OBR_CHK_ON	: 全コードに対して、チェックデジット計算有効(※1)
OBR_OUT_ON	: 全コードに対して、チェックキャラクタ出力有効(※2)

※ 1 チェックデジット計算に対応しているコードのみ有効となります。

※ 2 チェックキャラクタ出力に対応しているコードのみ有効となります。

※ RSS コードは 2007 年 2 月に GS1 DataBar に名称変更されています。

RSS-14・・・GS1 DataBar Omnidirectional

RSS Limited・・・GS1 DataBar Limited

RSS Expanded・・・GS1 DataBar Expanded

RSS-14 Stacked・・・GS1 DataBar Stacked

RSS Expanded Stacked・・・GS1 DataBar Expanded Stacked

※ Code128 を読取許可に設定すると、GS1-128 コードも読取可能となります。

GS1-128 は 2006 年 1 月に旧名称 EAN-128 より名称変更されています。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー
OBR_ERROR_HOTKEY	: スキャンキー登録エラー
	Device Emulator では発生しません
OBR_ERROR_INVALID_ACCESS	: カメラとの同時使用競合エラー
	Device Emulator では発生しません
OBR_PRM	: パラメータエラー
	Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、RSS コードの有効/無効設定と RSS Stacked コードの有効/無効設定は常に同一設定となります。そのため、本関数の引数において、RSS コードの有効/無効設定と RSS Stacked コードの有効/無効設定で異なる設定を指定した場合、有効設定を優先します。

4.2 OBRClose

読み取り禁止状態を設定します。

```
[C++]  
int OBRClose()
```

```
[Visual Basic]  
Public Shared Function OBRClose() As Int32
```

```
[C#]  
public static Int32 OBRClose()
```

解説

レーザスキャナドライバを読み取り禁止状態 (Close 状態) にします。

パラメータ

なし

戻り値

OBR_OK	: 正常終了
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.3 OBRSetLED

LED による通知方法を設定します。

```
[C++]
int OBRSetLED(
    BYTE byMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetLED( _
    ByVal byMode As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetLED(
    Byte byMode
);
```

解説

読み取り完了時に、LED による通知を行うかどうかを設定します。

パラメータ

byMode

LED による通知方法を、以下の値で指定します。

- OBR_LEDOFF : LED 制御なし
- OBR_LEDON : LED 制御あり
- OBR_LEDEROF : LED 制御あり(読み取りエラー時は制御なし)

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.4 OBRGetLED

LED による通知方法を取得します。

```
[C++]
int OBRGetLED(
    BYTE *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetLED( _
    ByRef pMode As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetLED(
    ref Byte pMode
);
```

解説

読み取り完了時に、LED による通知を行うかどうかを取得します。

パラメータ

pMode

LED による通知方法の値を取得します。取得する値は **OBRSetLED** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.5 OBRSetBuzzer

ブザーによる通知方法を設定します。

```
[C++]
int OBRSetBuzzer (
    BYTE byMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetBuzzer ( _
    ByVal byMode As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetBuzzer (
    Byte byMode
);
```

解説

読み取り成功時に、ブザーにより通知をおこなうかどうかを設定します。

パラメータ

byMode

ブザーによる通知方法を、以下の値で指定します。

OBR_BUZOFF : ブザー通知 OFF
OBR_BUZON : ブザー通知 ON

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
OBR_PRM : パラメータエラー
FUNCTION_UNSUPPORTED : 未サポートエラー

4.6 OBRGetBuzzer

ブザーによる通知方法を取得します。

```
[C++]
int OBRGetBuzzer (
    BYTE *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetBuzzer ( _
    ByRef pMode As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetBuzzer (
    ref Byte pMode
);
```

解説

読み取り成功時に、ブザーによる通知を行うかどうかを取得します。取得する値は関数を参照してください。

パラメータ

pMode

ブザーによる通知方法の値を取得します。取得する値は **OBRSetBuzzer** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.7 OBRSetVibrator

バイブレータによる通知方法を設定します。

```
[C++]
int OBRSetVibrator(
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetVibrator( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetVibrator(
    Int32 dwMode
);
```

解説

読み取り成功時に、バイブレータにより通知をおこなうかどうかを設定します。

パラメータ

dwMode

バイブレータによる通知方法を、以下の値で指定します。

- OBR_VIBOFF : バイブレータ通知 OFF
- OBR_VIBON : バイブレータ通知 ON

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.8 OBRGetVibrator

バイブレータ による通知方法を取得します。

```
[C++]
int OBRGetVibrator(
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetVibrator( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetVibrator(
    ref Int32 pMode
);
```

解説

読み取り成功時に、バイブレータによる通知を行うかどうかを取得します。

パラメータ

pMode

バイブレータによる通知方法の値を取得します。取得する値は **OBRSetVibrator** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.9 OBRSetSwingAngle

レーザ振れ角制御モードを設定します。

```
[C++]
int OBRSetSwingAngle(
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetSwingAngle( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetSwingAngle(
    Int32 dwMode
);
```

解説

レーザ振れ角制御モードを設定します。各振れ角モードでのレーザの照射幅は、OBR キャリブレーションにより設定することができます。OBR キャリブレーションにより設定を行わなかった場合は、工場出荷時に設定された照射幅となります。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetSwingAngle 関数を実行することにより、設定値を確認することができます。

パラメータ

dwMode

レーザ振れ角制御モードを、以下の値で指定します。

OBR_SWING_OFF	: 振れ角制御なし
OBR_SWING_WIDE	: 振れ角大
OBR_SWING_MIDDLE	: 振れ角中
OBR_SWING_NARROW	: 振れ角小

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

補足

OBR キャリブレーションは、レーザスキャナ設定ユーティリティから行うことができます。キャリブレーションを行うためには、専用のバーイメージが必要となります。

注意

レーザー振れ角制御モード(OBRSetSwingAngle 関数)において振れ角大/振れ角中/振れ角小を設定し、ゲインコントロール設定(OBRSetGainControl 関数)においてゲインオートを設定している場合は、バーコードの印字品質のより読取精度が低下することがあります。

振れ角大/振れ角中/振れ角小を設定する場合は、ゲインコントロールをゲインモード 0~3 に設定してください。

4.10 OBRGetSwingAngle

レーザ振れ角制御モードを取得します。

```
[C++]
int OBRGetSwingAngle(
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetSwingAngle( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetSwingAngle(
    ref Int32 pMode
);
```

解説

現在設定されているレーザ振れ角制御モードを取得します。

パラメータ

pMode

レーザ振れ角制御モードの値を取得します。取得する値は **OBRSetSwingAngle** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.11 OBRSetFocus

レーザフォーカスモードを設定します。

```
[C++]
int OBRSetFocus(
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetFocus( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetFocus(
    Int32 dwMode
);
```

解説

レーザフォーカスモードを設定します。レーザフォーカスとは、バーコード読み取り成功時に、読み取ったバーコードをレーザで指し示す機能です。レーザが複数のバーコードにあたってしまい、読み取ったときにどのバーコードを読み取ったのかわかりにくい、といった場合に有効な機能です。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、**OBRGetFocus** 関数を実行することにより、設定値を確認することができます。

パラメータ

dwMode

レーザフォーカスモードを以下の値で指定します。

OBR_FOCUSOFF : レーザフォーカス OFF
OBR_FOCUSON : レーザフォーカス ON

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
OBR_PRM : パラメータエラー
FUNCTION_UNSUPPORTED : 未サポートエラー

4.12 OBRGetFocus

現在設定されているレーザーフォーカスモードを取得します。

```
[C++]
int OBRGetFocus(
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetFocus( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetFocus(
    ref Int32 pMode
);
```

解説

現在設定されているレーザーフォーカスモードを取得します。

パラメータ

pMode

レーザーフォーカスモードを取得します。取得する値は **OBRSetFocus** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.13 OBRSetScanningNotification

読み取り完了通知を設定します。

```
[C++]
int OBRSetScanningNotification(
    DWORD dwMode,
    HWND hWnd
)
```

```
[Visual Basic]
Public Shared Function OBRSetScanningNotification( _
    ByVal dwMode As Int32, _
    ByVal hWnd As IntPtr _
) As Int32
```

```
[C#]
public static Int32 OBRSetScanningNotification(
    Int32 dwMode,
    IntPtr hWnd
);
```

解説

レーザスキャナによる読み取りの完了通知を、イベントで行うかウィンドウメッセージで行うかを設定します。

■ウィンドウメッセージ通知

WM_OBR_SCANNING(WM_USER +0x530)のウィンドウメッセージを、指定されているウィンドウハンドルに対して送ります。

■イベント通知

読み取り完了時に発行されるイベントは“OBRScanningEvent”という名前のイベントになります。

WindowsCE では、名前は Unicode で表されるため、プログラム上では TEXT(“OBRScanningEvent”) と指定します。

詳細は、「読み取り完了の通知について」を参照して下さい。

パラメータ

dwMode

読み取り完了時の通知方法を、以下の値で指定します。

- OBR_MESSAGE : ウィンドウメッセージ通知
- OBR_EVENT : イベント通知

hWnd

dwMode に OBR_MESSAGE を指定した場合の、ウィンドウメッセージ送り先のハンドルを指定します。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UN SUPPORT	: 未サポートエラー

4.14 OBRGetScanningNotification

読み取り完了通知を取得します。

```
[C++]
int OBRGetScanningNotification(
    DWORD *pMode,
    HWND *hWnd
)
```

```
[Visual Basic]
Public Shared Function OBRGetScanningNotification( _
    ByRef pMode As Int32, _
    ByRef hWnd As IntPtr _
) As Int32
```

```
[C#]
public static Int32 OBRGetScanningNotification(
    ref Int32 pMode,
    ref IntPtr hWnd
);
```

解説

レーザスキャナによる読み取りの完了通知を、イベントで行うか、ウィンドウメッセージで行うかの設定を取得します。

パラメータ

pMode

読み取り完了時の通知方法を取得します。取得する値は `OBRSetScanningNotification` 関数を参照してください。

hWnd

`pMode` に `OBR_MESSAGE` が設定されている場合の、ウィンドウメッセージ送り先のハンドルを取得します。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー
	Device Emulator では発生しません
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.15 OBRGetLastEventStatus

イベントステータスを取得します。

```
[C++]
int OBRGetLastEventStatus(
    DWORD *pStatus
)
```

```
[Visual Basic]
Public Shared Function OBRGetLastEventStatus( _
    ByRef pStatus As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetLastEventStatus(
    ref Int32 pStatus
);
```

解説

読み取り完了時の通知をイベントにより取得する場合、イベント発行後に読み取りの完了要因を取得します。詳細は、「読み取り完了の通知について」を参照して下さい。

パラメータ

pStatus

イベントの要因を取得します。

- OBR_NONE : イベント要因なし
- OBR_SUCCESS : 読み取り成功
- OBR_TIMEOUT : タイムアウト
- OBR_BUFFEROVER : OBR バッファ許容量オーバーエラー

戻り値

- OBR_OK : 正常終了
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.16 OBRSetScanningKey

読み取り開始キーを設定します。

```
[C++]
int OBRSetScanningKey(
    DWORD dwKey
)
```

```
[Visual Basic]
Public Shared Function OBRSetScanningKey( _
    ByVal dwKey As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetScanningKey(
    Int32 dwKey
);
```

解説

読み取り開始キーの設定をセットします。OBROpen 関数により Open 状態にした後で、本関数により設定されている開始キーを押すと、スキャンを開始します。

パラメータ

dwKey

読み取り開始キーを以下の値の論理和で指定します。

OBR_TRIGGERKEY_L	: トリガキーL
OBR_TRIGGERKEY_R	: トリガキーR
OBR_CURSORKEY_L	: カーソルキー左
OBR_CURSORKEY_R	: カーソルキー右
OBR_CURSORKEY_U	: カーソルキー上
OBR_CURSORKEY_D	: カーソルキー下
OBR_GUNTRIGGER	: ガントリガキー(DT-5200 のみ)
OBR_GUNGRIP	: ガントリガキー(DT-5200 のみ)
OBR_CENTERTRIGGER	: センターキー

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.17 OBRGetScanningKey

読み取り開始キーを取得します。

```
[C++]
int OBRGetScanningKey(
    DWORD *pKey
)
```

```
[Visual Basic]
Public Shared Function OBRGetScanningKey( _
    ByRef pKey As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetScanningKey(
    ref Int32 pKey
);
```

解説

現在設定されている読み取り開始キーの設定を取得します。OBROpen 関数により Open 状態にした後で、本関数により設定されている開始キーを押すと、スキャンを開始します。この関数により読み取り開始キーに設定されたキーは、Open 中レーザスキャナドライバにより占有されるため、ユーザは使用することができなくなります。OBRClose 関数により Close することで、再び使用可能となります。

パラメータ

pKey

読み取り開始キーを取得します。取得する値は OBRSetScanningKey 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.18 OBRSetAllOptions

レーザスキャナドライバの全ての設定を一括設定します。

```
[C++]
int OBRSetAllOptions(
    M_TBL *pModeTable
)
```

```
[Visual Basic]
Public Shared Function OBRSetAllOptions( _
    ByVal pModeTable As C:OBReadLibNet+M_TBL _
) As Int32
```

```
[C#]
public static Int32 OBRSetAllOptions(
    C:OBReadLibNet+M_TBL pModeTable
);
```

解説

レーザスキャナドライバの全ての設定を一括設定します。

パラメータ

pModeTable

スキャナドライバ設定構造体を指定します。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

補足

M_TBL 構造体については、「M_TBL」を参照して下さい。

4.19 OBRGetAllOptions

レーザスキャナドライバの全ての設定を一括して取得します。

```
[C++]
int OBRGetAllOptions(
    M_TBL *pModeTable
)
```

```
[Visual Basic]
Public Shared Function OBRGetAllOptions( _
    ByVal pModeTable As C:OBReadLibNet+M_TBL _
) As Int32
```

```
[C#]
public static Int32 OBRGetAllOptions(
    C:OBReadLibNet+M_TBL pModeTable
);
```

解説

レーザスキャナドライバの全ての設定を一括して取得します。

パラメータ

pModeTable

取得したスキャナドライバ設定を格納する構造体へのポインタ。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

補足

M_TBL 構造体については、「M_TBL」を参照して下さい。

4.20 OBRSetDefaultSymbology

レーザスキャナドライバの全ての設定値をデフォルトに設定します。

```
[C++]  
int OBRSetDefaultSymbology ()
```

```
[Visual Basic]  
Public Shared Function OBRSetDefaultSymbology () As Int32
```

```
[C#]  
public static Int32 OBRSetDefaultSymbology ()
```

解説

レーザスキャナドライバの全ての設定値をデフォルトに設定します。設定ファイルが存在する状態で、OBRLoadConfigFile 関数を呼び出していれば、設定ファイルに保存されている値が設定されます。

パラメータ

なし

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UN SUPPORT	: 未サポートエラー

4.21 OBRSetDefaultSymbologyEx

全設定値を工場出荷状態に戻します。

```
[C++]  
int OBRSetDefaultSymbologyEx ( )
```

```
[Visual Basic]  
Public Shared Function OBRSetDefaultSymbologyEx () As Int32
```

```
[C#]  
public static Int32 OBRSetDefaultSymbologyEx ()
```

解説

レーザスキャナドライバの全ての設定値を工場出荷状態に戻します。
Device Emulator では、常に FUNCTION_UN SUPPORT を返します。

パラメータ

なし

戻り値

OBR_OK : 正常終了
FUNCTION_UN SUPPORT : 未サポートエラー

4.22 OBRSaveLogFile

ドライバログ情報を出力します。

```
[C++]  
int OBRSaveLogFile()
```

```
[Visual Basic]  
Public Shared Function OBRSaveLogFile() As Int32
```

```
[C#]  
public static Int32 OBRSaveLogFile()
```

解説

レーザスキャナドライバによりメモリに保存されたスキャナのログ情報を、所定のファイルに保存します。

パラメータ

なし

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

補足

ドライバログ、ルートディレクトリに以下のファイル名で出力します。

スキャナ制御関連	: OBRLog.dat
デコーダ関連	: DecodeLog.dat

本関数を呼び出すには、OBROpen 関数を呼び出してレーザスキャナドライバを Open 状態にする必要があります。

4.23 OBRGetc

OBR バッファから 1 文字読み出します。

```
[C++]
int OBRGetc(
    DWORD *dwRcd
)
```

```
[Visual Basic]
Public Shared Function OBRGetc( _
    ByRef dwRcd As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetc(
    ref Int32 dwRcd
);
```

解説

レーザスキャナドライバ内のデコードデータ格納バッファから 1 文字読み出します。
Device Emulator では、I/O Simulator で指定したコードを読み出します。

パラメータ

dwRcd

読み出したデコードデータの種類を取得します。

OBR_NONDT	: データなし
OBR_CD39	: CODE39
OBR_NW_7	: NW-7
OBR_WPCA	: WPC(UPC-E 以外)addon
OBR_WPC	: WPC(UPC-E 以外)
OBR_UPEA	: UPC-E addon
OBR_UPE	: UPC-E
OBR_IDF	: Industrial 2of5
OBR_ITF	: Interleaved 2of5
OBR_CD93	: CODE93
OBR_CD128	: CODE128
OBR_MSI	: MSI
OBR_IATA	: IATA
OBR_RSS14	: RSS-14 または RSS-14 Stacked
OBR_RSSLTD	: RSS Limited
OBR_RSSEXP	: RSS Expanded または RSS Expanded Stacked

※ RSS-14 と RSS-14 Stacked のコード種類を表す値は、同じになります。

※ RSS Expanded と RSS Expanded Stacked のコード種類を表す値は、同じになります。

※ RSS コードは 2007 年 2 月に GS1 DataBar に名称変更されています。

RSS-14・・・GS1 DataBar Omnidirectional

RSS Limited・・・GS1 DataBar Limited

RSS Expanded・・・GS1 DataBar Expanded

RSS-14 Stacked・・・GS1 DataBar Stacked
RSS Expanded Stacked・・・GS1 DataBar Expanded Stacked
※ GS1-128 コード読取時にも OBR_CD128 が取得されます。
GS1-128 は 2006 年 1 月に旧名称 EAN-128 より名称変更されています。

戻り値

OBR データ(1 文字)

または、

OBR_NOT_DEVICE : OBR(スキャナ部)非常駐状態
Device Emulator では発生しません

FUNCTION_UNSUPPORTED : 未サポートエラー

補足

レーザスキャナドライバ未オープン状態にて本関数を実行する場合、異常終了することがあります。

4.24 OBRGets

OBR バッファから 1 コード分のデコードデータを読み出します。

```
[C++]
int OBRGets(
    BYTE *pDecodeData,
    DWORD *pCode,
    BYTE *pLength
)
```

```
[Visual Basic]
Public Shared Function OBRGets( _
    ByVal pDecodeData As Byte(), _
    ByRef pCode As Int32, _
    ByRef pLength As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGets(
    Byte[] pDecodeData,
    ref Int32 pCode,
    ref Byte pLength
);
```

解説

レーザスキャナドライバ内のデコードデータ格納バッファから 1 コード分のデコードデータを読み出します。

Device Emulator では、I/O Simulator で指定したコードを読み出します。

パラメータ

pDecodeData

読み出したデコードデータを取得します。

pCode

読み出したデコードデータの種類を取得します。

OBR_NONDT	: データなし
OBR_CD39	: CODE39
OBR_NW_7	: NW-7
OBR_WPCA	: WPC(UPC-E 以外)addon
OBR_WPC	: WPC(UPC-E 以外)
OBR_UPEA	: UPC-E addon
OBR_UPE	: UPC-E
OBR_IDF	: Industrial 2of5
OBR_ITF	: Interleaved 2of5
OBR_CD93	: CODE93
OBR_CD128	: CODE128
OBR_MSI	: MSI

- OBR_IATA : IATA
- OBR_RSS14 : RSS-14 または RSS-14 Stacked
- OBR_RSSLTD : RSS Limited
- OBR_RSSEXP : RSS Expanded または RSS Expanded Stacked
- ※ RSS-14 と RSS-14 Stacked のコード種類を表す値は、同じになります。
- ※ RSS Expanded と RSS Expanded Stacked のコード種類を表す値は、同じになります。
- ※ RSS コードは 2007 年 2 月に GS1 DataBar に名称変更されています。
- RSS-14・・・GS1 DataBar Omnidirectional
- RSS Limited・・・GS1 DataBar Limited
- RSS Expanded・・・GS1 DataBar Expanded
- RSS-14 Stacked・・・GS1 DataBar Stacked
- RSS Expanded Stacked・・・GS1 DataBar Expanded Stacked
- ※ GS1-128 コード読取時にも OBR_CD128 が取得されます。
- GS1-128 は 2006 年 1 月に旧名称 EAN-128 より名称変更されています。

pLength

読み出したデコードデータの長さを取得します。

戻り値

- OBR_OK : 正常終了
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- FUNCTION_UNSUPPORTED : 未サポートエラー

補足

この関数で取得したデコードデータを WindowsCE 上において表示するには、Unicode に変換する必要があります。

OBR バッファにデータがない場合、pCode に OBR_NONDT が格納されます。

4.25 OBRGetStatus

OBR バッファ状態を取得します。

```
[C++]
int OBRGetStatus(
    DWORD *pSize,
    BYTE *pNumber
)
```

```
[Visual Basic]
Public Shared Function OBRGetStatus( _
    ByRef pSize As Int32, _
    ByRef pNumber As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetStatus(
    ref Int32 pSize,
    ref Byte pNumber
);
```

解説

レーザスキャナドライバ内のデコードデータ格納バッファの状態を取得します。

パラメータ

pSize

取得したバッファ内の残りバイト数(残り段数×1データの最大長)を取得します。

pNumber

取得したバッファ内の残り段数を取得します。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.26 OBRClearBuff

OBR バッファをクリアします。

```
[C++]  
int OBRClearBuff()
```

```
[Visual Basic]  
Public Shared Function OBRClearBuff() As Int32
```

```
[C#]  
public static Int32 OBRClearBuff()
```

解説

レーザスキャナドライバ内のデコードデータ格納バッファをクリアします。

パラメータ

なし

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.27 OBRSetBuffType

デコードデータ出力方式を設定します。

```
[C++]
int OBRSetBuffType(
    BYTE byType
)
```

```
[Visual Basic]
Public Shared Function OBRSetBuffType( _
    ByVal byType As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetBuffType(
    Byte byType
);
```

解説

デコードデータの出力方式を設定します。

パラメータ

byType

出力方式設定を、以下の値で指定します。

- OBR_BUFOBR : OBR バッファへ出力
- OBR_STOFF : 指定されたウィンドウハンドルへキーメッセージとして出力
- OBR_ASTOFF : クリップボードを経由してフォーカスのある場所へ出力

戻り値

- OBR_OK : 正常終了
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

補足

キーメッセージの送り先のウィンドウハンドルは、OBROpen 関数により指定します。

4.28 OBRGetBuffType

デコードデータ出力方式を取得します。

```
[C++]
int OBRGetBuffType(
    BYTE *pType
)
```

```
[Visual Basic]
Public Shared Function OBRGetBuffType( _
    ByRef pType As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetBuffType(
    ref Byte pType
);
```

解説

デコードデータの出力方式を取得します。

パラメータ

pType

出力方式を取得します。取得する値は `OBRSetBuffType` 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.29 OBRSetCheckCounter

デコードデータの照合回数を設定します。

```
[C++]
int OBRSetCheckCounter (
    int nCounter
)
```

```
[Visual Basic]
Public Shared Function OBRSetCheckCounter ( _
    ByVal nCounter As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetCheckCounter (
    Int32 nCounter
);
```

解説

デコードデータの照合回数を設定します。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetCheckCounter 関数を実行することにより、設定値を確認することができます。

パラメータ

nCounter

デコードデータの照合回数を指定します。

■ レーザスキャナモデル

1～9 の範囲で指定します。

指定する値の詳細については、各機種ソフトウェアマニュアルを参照してください。

■ リニアイメージャモデル

0～9 の範囲で指定します。デフォルトは 0 回です。

0 を指定した場合は、照合回数が自動になります。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.30 OBRGetCheckCounter

デコードデータの照合回数設定を取得します。

```
[C++]
int OBRGetCheckCounter (
    int *pCounter
)
```

```
[Visual Basic]
Public Shared Function OBRGetCheckCounter ( _
    ByRef pCounter As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetCheckCounter (
    ref Int32 pCounter
);
```

解説

デコードデータの照合回数設定を取得します。

パラメータ

pCounter

デコードデータの照合回数を取得します。取得する値は `OBRSetCheckCounter` 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.31 OBRSetScanningTimeout

読み取りタイムアウト時間を設定します。

```
[C++]
int OBRSetScanningTimeout(
    int nTimeout
)
```

```
[Visual Basic]
Public Shared Function OBRSetScanningTimeout( _
    ByVal nTimeout As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetScanningTimeout(
    Int32 nTimeout
);
```

解説

読み取りタイムアウト時間を設定します。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetScanningTimeout 関数を実行することにより、設定値を確認することができます。

パラメータ

nTimeout

読み取りタイムアウト時間を 1～9 秒の範囲で指定します。デフォルトは 3 秒です。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.32 OBRGetScanningTimeout

読み取りタイムアウト時間設定を取得します。

```
[C++]
int OBRGetScanningTimeout(
    int *pTimeout
)
```

```
[Visual Basic]
Public Shared Function OBRGetScanningTimeout( _
    ByRef pTimeout As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetScanningTimeout(
    ref Int32 pTimeout
);
```

解説

読み取りタイムアウト時間設定を取得します。

パラメータ

pTimeout

読み取りタイムアウト時間を取得します。取得する値は `OBRSetScanningTimeout` 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.33 OBRSetSuffixChar

デコードデータの末尾制御コードを設定します。

```
[C++]
int OBRSetSuffixChar (
    BYTE bySuffix
)
```

```
[Visual Basic]
Public Shared Function OBRSetSuffixChar ( _
    ByVal bySuffix As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetSuffixChar (
    Byte bySuffix
);
```

解説

デコードデータの末尾に、追加して出力する制御コードの設定をセットします。

パラメータ

bySuffix

追加出力する制御コードを、以下の値で指定します。

OBR_ENDCR	: CR コード(0Dh)
OBR_ENDLF	: LF コード(0Ah)
OBR_ENDCL	: CR コード+LF コード(0D0Ah)
OBR_ENDTAB	: HT コード(09h)
OBR_ENDNULL	: 終了コードなし

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.34 OBRGetSuffixChar

デコードデータの末尾制御コードを取得します。

```
[C++]
int OBRGetSuffixChar (
    BYTE *pSuffix
)
```

```
[Visual Basic]
Public Shared Function OBRGetSuffixChar ( _
    ByRef pSuffix As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetSuffixChar (
    ref Byte pSuffix
);
```

解説

デコードデータの末尾に、追加して出力する制御コードの設定を取得します。

パラメータ

pSuffix

追加出力する制御コードを取得します。取得する値は **OBRSetSuffixChar** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.35 OBRSetScanningType

読み取り方式を設定します。

```
[C++]
int OBRSetScanningType(
    BYTE byType
)
```

```
[Visual Basic]
Public Shared Function OBRSetScanningType( _
    ByVal byType As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetScanningType(
    Byte byType
);
```

解説

スキヤナの読み取り方式を設定します。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetScanningType 関数を実行することにより、設定値を確認することができます。

パラメータ

byType

読み取り方式を、以下の値で指定します。

- OBR_TYPE0 : 単発読み・キー離しチェックなし
- OBR_TYPE1 : 連続読み・キー離しチェックあり

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキヤナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.36 OBRGetScanningType

読み取り方式を取得します。

```
[C++]
int OBRGetScanningType(
    BYTE *pType
)
```

```
[Visual Basic]
Public Shared Function OBRGetScanningType( _
    ByRef pType As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetScanningType(
    ref Byte pType
);
```

解説

スキヤナの読み取り方式を取得します。

パラメータ

pType

読み取り方式を取得します。取得する値は `OBRSetScanningType` 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキヤナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.37 OBRSetScanningCounter

連続読み取り回数を設定します。

```
[C++]
int OBRSetScanningCounter (
    int nCounter
)
```

```
[Visual Basic]
Public Shared Function OBRSetScanningCounter ( _
    ByVal nCounter As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetScanningCounter (
    Int32 nCounter
);
```

解説

読み取り方式が連続読みの場合の、連続して読み取る回数を設定します。

パラメータ

nCounter

連続読み取り回数を 1～9 回の範囲で指定します。デフォルトは 1 回です。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.38 OBRGetScanningCounter

連続読み取り回数を取得します。

```
[C++]
int OBRGetScanningCounter (
    int *pCounter
)
```

```
[Visual Basic]
Public Shared Function OBRGetScanningCounter ( _
    ByRef pCounter As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetScanningCounter (
    ref Int32 pCounter
);
```

解説

読み取り方式が連続読みの場合の、連続して読み取る回数を取得します。

パラメータ

pCounter

連続読み取り回数を取得します。取得する値は `OBRSetScanningCounter` 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.39 OBRSetMultiStepReading

段数読みを設定します。

```
[C++]
int OBRSetMultiStepReading(
    BYTE byMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetMultiStepReading( _
    ByVal byMode As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetMultiStepReading(
    Byte byMode
);
```

解説

段数読みを設定します。段数読みの方式は以下のとおりです。

Device Emulator では、複数のコードを **I/O Simulator** から取得することができません。

■通常読み

OBROpen 関数を実行して **Open** 状態にしてから、**OBRClose** 関数を実行して **Close** 状態にするまでの間、読み取れるバーコードの数に制限がありません。連続読みを行う場合、1回前の読み取り結果と重複する場合は読み取りません。

■段数読み

OBROpen 関数を実行して **Open** 状態にしてから、**OBRClose** 関数を実行して **Close** 状態にするまでの間、指定した数のバーコードしか読み取りません。読み取りを行うバーコードの数は、**OBRSetScanningCounter** 関数により指定します。連続読みを行う場合、**Open** 状態になってから読み取ってきた結果と重複する場合は読み取りません。

パラメータ

byMode

段数読みの方式を、以下の値で指定します。

OBR_NORM : 通常読み
OBR_DANN : 段数読み

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
OBR_PRM : パラメータエラー
FUNCTION_UNSUPPORTED : 未サポートエラー

4.40 OBRGetMultiStepReading

段数読みを取得します。

```
[C++]
int OBRGetMultiStepReading(
    BYTE *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetMultiStepReading( _
    ByRef pMode As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetMultiStepReading(
    ref Byte pMode
);
```

解説

段数読みを取得します。

パラメータ

pMode

段数読みの方式を取得します。取得する値は `OBRSetMultiStepReading` 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.41 OBRStartScanning

連続読み取りを開始します。

```
[C++]
int OBRStartScanning(
    DWORD dwTimeout
)
```

```
[Visual Basic]
Public Shared Function OBRStartScanning( _
    ByVal dwTimeout As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRStartScanning(
    Int32 dwTimeout
);
```

解説

レーザの点灯を開始し、連続読み取りを開始します。スキャン実行中は、**OBRStopScanning** 関数によりスキャンを停止することができます。本関数を呼び出すには、**OBROpen** 関数を呼び出してレーザスキャナドライバを **Open** 状態にする必要があります。読み取り回数に制限はありません。連続読み取り実行中に電源 OFF した場合、次回電源 ON 時に自動的にレーザが点灯するため、注意が必要となります。

パラメータ

dwTimeout

読み取りタイムアウト時間を、1~300 秒、または **OBR_INFINITE**(タイムアウトなし)で指定します。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_ERROR_SCANNING	: スキャン中エラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

補足

1 つのバーコードを読み取ってから、バーコードを読み取ることなく **dwTimeout** で指定した時間を経過するとスキャンを停止します。

4.42 OBRStopScanning

連続読み取りを停止します。

```
[C++]  
int OBRStopScanning()
```

```
[Visual Basic]  
Public Shared Function OBRStopScanning() As Int32
```

```
[C#]  
public static Int32 OBRStopScanning()
```

解説

OBRStartScanning 関数によりスキャンを実行中している最中に呼び出すことにより、スキャンを停止させます。本関数を呼び出すには、OBROpen 関数を呼び出してレーザスキャナドライバを Open 状態にする必要があります。

パラメータ

なし

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UN SUPPORT	: 未サポートエラー

4.43 OBRSetDecodeLevel

デコードレベルを設定します。

```
[C++]
int OBRSetDecodeLevel (
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetDecodeLevel ( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetDecodeLevel (
    Int32 dwMode
);
```

解説

デコードレベルを設定します。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetDecodeLevel 関数を実行することにより、設定値を確認することができます。

デコードレベル設定とは、バーの比率が正しくない不良バーコードに対する読取精度を切り替える設定です。デコードレベルには以下の 2 段階があります。

■標準

標準のデコードロジックを使って読み取りを行います。

■高度

最初に標準のデコードロジックを使って読み取りを行い、読み取りに失敗した場合、追加のデコードロジックを使って読み取りを行います。EAN (JAN)、UPC-A、Code39、Code128 に対する読み取り率が向上します。その他のコードについては、読み取り率が上がることはありません。また、高度に設定した場合、誤読をする可能性が高くなります。

読み取り対象のバーコードが、標準では読み取りできない場合にのみ高度に設定することを推奨します。

■堅実

デコードロジックにおいて、バーの比率チェックを厳しくした状態で読み取りを行うことにより、スキヤンの正確性を高めることができます。これにより、Code128、EAN (JAN) の誤読を回避することが可能となります。(上記以外のコードについては、標準モードになります。)

パラメータ

dwMode

デコードレベルを、以下の値で指定します。

- OBR_DECODELEVEL_NORMAL : 標準(DT-5200)
- OBR_DECODELEVEL_HIGH : 高度(DT-X7 / DT-5300 / IT-300 / DT-X8 / IT-G500 / DT-X100 / DT-X200)
- OBR_DECODELEVEL_STEADY : 堅実(DT-5300 / IT-300 / DT-X8)

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、デコードレベルは高度固定です。その他のデコードレベルを指定しても、高度で動作します。

4.44 OBRGetDecodeLevel

デコードレベルを取得します。

```
[C++]
int OBRGetDecodeLevel(
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetDecodeLevel( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetDecodeLevel(
    ref Int32 pMode
);
```

解説

デコードレベルを取得します。

パラメータ

pMode

デコードレベルを取得します。取得する値は `OBRSetDecodeLevel` 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.45 OBRSetScanningCode

読み取り許可・禁止状態を設定します。

```
[C++]
int OBRSetScanningCode(
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetScanningCode( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetScanningCode(
    Int32 dwMode
);
```

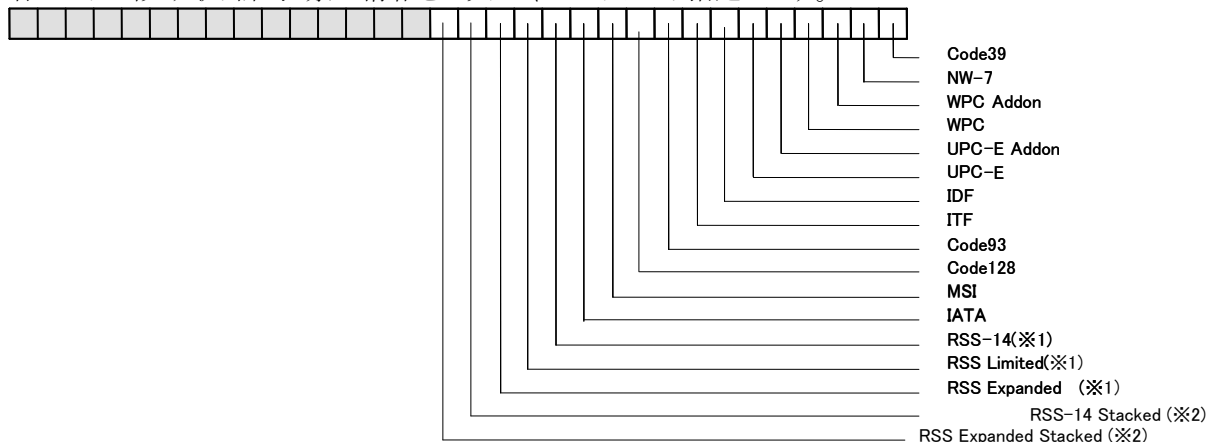
解説

各コードの読み取り許可・禁止設定を設定します。

パラメータ

dwMode

各コードの読み取り許可・禁止情報をビットフィールドにより指定します。



OBR_CD39	: Code39 有効
OBR_NW_7	: NW-7 有効
OBR_WPCA	: WPC Addon 有効
OBR_WPC	: WPC 有効
OBR_UPEA	: UPC-E Addon 有効
OBR_UPE	: UPC-E 有効
OBR_IDF	: Industrial 2 of 5 有効
OBR_ITF	: Interleaved 2 of 5 有効
OBR_CD93	: Code93 有効
OBR_CD128	: Code128 有効
OBR_MSI	: MSI 有効

OBR_IATA	: IATA 有効
OBR_RSS14	: RSS-14 有効
OBR_RSSLTD	: RSS Limited 有効
OBR_RSSEXP	: RSS Expanded 有効
OBR_RSS14S	: RSS-14 Stacked 読み取り許可
OBR_RSSEXPS	: RSS Expanded Stacked 読み取り許可
OBR_ALL	: 全コード有効

複数種類のバーコードの読み取りを有効にする場合は、各コードに対応する値の論理和を指定します。

※ RSS コードは 2007 年 2 月に GS1 DataBar に名称変更されています。

RSS-14・・・GS1 DataBar Omnidirectional

RSS Limited・・・GS1 DataBar Limited

RSS Expanded・・・GS1 DataBar Expanded

RSS-14 Stacked・・・GS1 DataBar Stacked

RSS Expanded Stacked・・・GS1 DataBar Expanded Stacked

※ Code128 を読取許可に設定すると、GS1-128 コードも読取可能となります。

GS1-128 は 2006 年 1 月に旧名称 EAN-128 より名称変更されています。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

補足

OBROpen 関数でも各コードの読み取り許可設定の指定が行えます。OBROpen 関数を呼び出す際に、読み取りコードの指定を行った場合、OBROpen 関数により指定される設定に更新されます。

4.46 OBRGetScanningCode

読み取り許可・禁止状態を取得します。

```
[C++]
int OBRGetScanningCode(
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetScanningCode( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetScanningCode(
    ref Int32 pMode
);
```

解説

各コードの読み取り許可・禁止設定を取得します。

パラメータ

pMode

各コードの読み取り許可・禁止状態を取得します。取得する値は `OBRSetScanningCode` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー Device Emulator では発生しません
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.47 OBRSetCode39Option

Code39 コードのデコードオプションを設定します。

```
[C++]
int OBRSetCode39Option(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetCode39Option( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetCode39Option(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

Code39 コードのデコードオプションを設定します。

パラメータ

byEnable

Code39 コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : Code39 の読み取り有効
- OBR_CODE_DISABLE : Code39 の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 2 桁です。

最小桁数より小さな Code39 データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 52 桁です。

IT-G500/DT-X100/DT-X200 では、最大桁数に 48 以上 52 以下の値を指定した場合、最大桁数は 48 となり、本関数は正常終了します。

最大桁数より大きな Code39 データのデコードは行いません。

byOutFormat

出力フォーマットを、以下の値で指定します。

- OBR_39SON : Start/Stop コードあり
- OBR_39SOFF : Start/Stop コードなし
- OBR_39ASON : Full ASCII、Start/Stop コードあり
- OBR_39ASOFF : Full ASCII、Start/Stop コードなし

byCheckDigit

チェックデジットを、以下の値で指定します。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタ出力を、以下の値で指定します。

- OBR_CHKKOFF : チェックキャラクタ出力なし
- OBR_CHKKON : チェックキャラクタ出力あり

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.48 OBRGetCode39Option

Code39 コードのデコードオプションを取得します。

```
[C++]
int OBRGetCode39Option(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetCode39Option( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetCode39Option(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

Code39 コードのデコードオプションを取得します。

パラメータ

pEnable

Code39 コード読み取り許可を取得します。取得する値は OBRSetCode39Option 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetCode39Option 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetCode39Option 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetCode39Option` 関数を参照してください。

pCheckDigit

チェックデジット設定を取得します。取得する値は `OBRSetCode39Option` 関数を参照してください。

pCheckChar

チェックキャラクタ出力を取得します。取得する値は `OBRSetCode39Option` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー <code>Device Emulator</code> では発生しません
<code>FUNCTION_UN SUPPORT</code>	: 未サポートエラー

4.49 OBRSetNW7Option

NW-7 コードのデコードオプションを設定します。

```
[C++]
int OBRSetNW7Option(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat
)
```

```
[Visual Basic]
Public Shared Function OBRSetNW7Option( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetNW7Option(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat
);
```

解説

NW-7 コードのデコードオプションを設定します。

パラメータ

byEnable

NW-7 コードの読み取り許可設定を、以下の値で指定します。

- OBR_CODE_ENABLE : NW-7 の読み取り有効
- OBR_CODE_DISABLE : NW-7 の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 2 桁です。
最小桁数より小さな NW-7 データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 63 桁です。
IT-G500/DT-X100/DT-X200 では、最大桁数に 50 以上 63 以下の値を指定した場合、最大桁数は 50 となり、本関数は正常終了します。

byOutFormat

出力フォーマットを、以下の値で指定します。

OBR_NWSON : Start/Stop コードあり
OBR_NWSOFF : Start/Stop コードなし

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
OBR_PRM : パラメータエラー
FUNCTION_UNSUPPORTED : 未サポートエラー

4.50 OBRSetNW7OptionEx

NW-7 コードのデコードオプションを設定します。

```
[C++]
int OBRSetNW7Option(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetNW7Option( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetNW7Option(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

NW-7 コードのデコードオプションを設定します。

Device Emulator では、常に FUNCTION_UNSUPPORTED を返します。

パラメータ

byEnable

NW-7 コードの読み取り許可設定を、以下の値で指定します。

- OBR_CODE_ENABLE : NW-7 の読み取り有効
- OBR_CODE_DISABLE : NW-7 の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 2 桁です。

最小桁数より小さな NW-7 データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 63 桁です。

IT-G500/DT-X100/DT-X200 では、最大桁数に 50 以上 63 以下の値を指定した場合、最大桁数は 50 となり、本関数は正常終了します。

byOutFormat

出力フォーマットを、以下の値で指定します。

OBR_NWSON : Start/Stop コードあり

OBR_NWSOFF : Start/Stop コードなし

byCheckDigit

チェックデジット設定を、以下の値で指定します。

OBR_CHKDOFF : チェックデジット計算なし

OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタ出力を、以下の値で指定します。

OBR_CHKKOFF : チェックキャラクタ出力なし

OBR_CHKKON : チェックキャラクタ出力あり

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
OBR_PRM : パラメータエラー
FUNCTION_UNSupport : 未サポートエラー

4.51 OBRGetNW7Option

NW-7 コードのデコードオプションを取得します。

```
[C++]
int OBRGetNW7Option(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetNW7Option( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetNW7Option(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

NW-7 コードのデコードオプションを取得します。

パラメータ

pEnable

NW-7 コードの読み取り許可を取得します。取得する値は OBRSetNW7OptionEx 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetNW7OptionEx 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetNW7OptionEx 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetNW7OptionEx` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetNW7OptionEx` 関数を参照してください。

pCheckChar

チェックキャラクタ出力を取得します。取得する値は `OBRSetNW7OptionEx` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー <code>Device Emulator</code> では発生しません
<code>FUNCTION_UN SUPPORT</code>	: 未サポートエラー

4.52 OBRSetWPCAddonOption

WPC Addon コードのデコードオプションを設定します。

```
[C++]
int OBRSetWPCAddonOption(
    BYTE byEnable,
    BYTE byCheckDigit
)
```

```
[Visual Basic]
Public Shared Function OBRSetWPCAddonOption( _
    ByVal byEnable As Byte, _
    ByVal byCheckDigit As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetWPCAddonOption(
    Byte byEnable,
    Byte byCheckDigit
);
```

解説

WPC Addon コードのデコードオプションを設定します。

パラメータ

byEnable

WPC Addon コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : WPC Addon の読み取り有効
- OBR_CODE_DISABLE : WPC Addon の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byCheckDigit

チェックデジット設定を、以下の値で指定します。

チェックデジット計算なしを設定する場合は、WPC のチェックデジット設定についても、チェックデジット計算なしを設定してください。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、UPC-B Addon を読み取ることはできません。また、チェックデジット計算は無効に設定することはできません。

4.53 OBRSetWPCAddonOptionEx

WPC Addon コードのデコードオプションを設定します。

```
[C++]
int OBRSetWPCAddonOptionEx(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetWPCAddonOptionEx( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetWPCAddonOptionEx(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

WPC Addon コードのデコードオプションを設定します。

パラメータ

byEnable

WPC Addon コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : WPC Addon の読み取り有効
- OBR_CODE_DISABLE : WPC Addon の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。WPC Addon コードの読み取り有効最小桁数は、10 桁固定で変更できません。必ず 10 を指定します。

byMaxLength

最大桁数を指定します。WPC Addon コードの読み取り有効最大桁数は、18 桁固定で変更できません。必ず 18 を指定します。

byOutFormat

出力フォーマットを、以下の値で指定します。

OBR_NUMSYSON	: UPC-A の先頭の 0 出力あり
OBR_NUMSYSOFF	: UPC-A の先頭の 0 出力なし
OBR_WPCGTIN	: UPC-A, EAN8(JAN8), EAN13(JAN13)を GTIN フォーマット(14 桁)出力

byCheckDigit

チェックデジット設定を指定。

チェックデジット計算なしを設定する場合は、WPC のチェックデジット設定についても、チェックデジット計算なしを設定してください。

OBR_CHKDOFF	: チェックデジット計算なし
OBR_CHKDON	: チェックデジット計算あり

byCheckChar

チェックキャラクタを指定します。WPC Addon コードのチェックデジット計算の設定は常に有効となるため、必ず OBR_CHKKON を指定します。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UN SUPPORT	: 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、UPC-B Addon を読み取ることはできません。また、チェックデジット計算は無効に設定することはできません。

4.54 OBRGetWPCAddonOption

WPC Addon コードのデコードオプションを取得します。

```
[C++]
int OBRGetWPCAddonOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetWPCAddonOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetWPCAddonOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

WPC Addon コードのデコードオプションを取得します。

パラメータ

pEnable

WPC Addon コードの読み取り許可を取得します。取得する値は OBRSetWPCAddonOptionEx 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetWPCAddonOptionEx 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetWPCAddonOptionEx 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetWPCAddonOptionEx` 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は `OBRSetWPCAddonOptionEx` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー Device Emulator では発生しません
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.55 OBRSetWPCOption

WPC コードのデコードオプションを設定します。

```
[C++]
int OBRSetWPCOption(
    BYTE byEnable,
    BYTE byCheckDigit
)
```

```
[Visual Basic]
Public Shared Function OBRSetWPCOption( _
    ByVal byEnable As Byte, _
    ByVal byCheckDigit As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetWPCOption(
    Byte byEnable,
    Byte byCheckDigit
);
```

解説

WPC コードのデコードオプションを設定します。

パラメータ

byEnable

WPC コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : WPC の読み取り有効
- OBR_CODE_DISABLE : WPC の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byCheckDigit

チェックデジット設定を、以下の値で指定します。

チェックデジット計算なしを設定する場合は、WPC Addon のチェックデジット設定についても、チェックデジット計算なしを設定してください。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、UPC-B Addon を読み取ることはできません。また、チェックデジット計算は無効に設定することはできません。

4.56 OBRSetWPCOptionEx

WPC コードのデコードオプションを設定します。

```
[C++]
int OBRSetWPCOptionEx(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetWPCOptionEx( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetWPCOptionEx(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

WPC コードのデコードオプションを設定します。

パラメータ

byEnable

WPC コードの読み取り許可、以下の値で指定します。

- OBR_CODE_ENABLE : WPC の読み取り有効
- OBR_CODE_DISABLE : WPC の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。WPC コードの読み取り有効最小桁数は、8 桁固定で変更できません。必ず 8 を指定します。

byMaxLength

最大桁数を指定します。WPC コードの読み取り有効最小桁数は、13 桁固定で変更できません。必ず 13 を指定します。

byOutFormat

出力フォーマットを、以下の値で指定します。

- OBR_NUMSYSON : UPC-A の先頭の 0 出力あり
- OBR_NUMSYSOFF : UPC-A の先頭の 0 出力なし
- OBR_WPCGTIN : UPC A, EAN8(JAN8), EAN13(JAN13)を GTIN フォーマット(14 桁)出力

byCheckDigit

チェックデジット設定を、以下の値で指定します。

チェックデジット計算なしを設定する場合は、WPC Addon のチェックデジット設定についても、チェックデジット計算なしを設定してください。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタ設定を指定します。WPC コードのチェックデジット計算の設定は常に有効となるため、必ず OBR_CHKON を指定します。

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、UPC-B Addon を読み取ることはできません。また、チェックデジット計算は無効に設定することはできません。

4.57 OBRGetWPCOption

WPC コードのデコードオプションを取得します。

```
[C++]
int OBRGetWPCOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetWPCOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetWPCOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

WPC コードのデコードオプションを取得します。

パラメータ

pEnable

WPC コードの読み取り許可を取得します。取得する値は `OBRSetWPCOptionEx` 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は `OBRSetWPCOptionEx` 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は `OBRSetWPCOptionEx` 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetWPCOptionEx` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetWPCOptionEx` 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は `OBRSetWPCOptionEx` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー <code>Device Emulator</code> では発生しません
<code>FUNCTION_UN SUPPORT</code>	: 未サポートエラー

4.58 OBRSetUPCEAddonOption

UPC-E Addon コードのデコードオプションを設定します。

```
[C++]
int OBRSetUPCEAddonOption(
    BYTE byEnable,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetUPCEAddonOption( _
    ByVal byEnable As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetUPCEAddonOption(
    Byte byEnable,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

UPC-E Addon コードのデコードオプションを設定します。

パラメータ

byEnable

UPC-E Addon コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : UPC-E Addon の読み取り有効
- OBR_CODE_DISABLE : UPC-E Addon の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byCheckDigit

チェックデジットを、以下の値で指定します。

チェックデジット計算なしを設定する場合は、UPC-E のチェックデジット設定についても、チェックデジット計算なしを設定してください。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタを、以下の値で指定します。

- OBR_CHKKOFF : チェックキャラクタ出力なし
- OBR_CHKKON : チェックキャラクタ出力あり

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UN SUPPORT	: 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、チェックデジット計算を無効に設定することはできません。
OBR_CHKDOFF を指定しても OBR_CHKDON になります。

4.59 OBRSetUPCEAddonOptionEx

UPC-E Addon コードのデコードオプションを設定します。

```
[C++]
int OBRSetUPCEAddonOptionEx(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetUPCEAddonOptionEx( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetUPCEAddonOptionEx(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

UPC-E Addon コードのデコードオプションを設定します。

パラメータ

byEnable

UPC-E Addon コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : UPC-E Addon の読み取り有効
- OBR_CODE_DISABLE : UPC-E Addon の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。UPC-E Addon コードの読み取り有効最小桁数は、9 桁固定で変更できません。必ず 9 を指定します。

byMaxLength

最大桁数を指定します。UPC-E Addon コードの読み取り有効最小桁数は、12 桁固定で変更できません。必ず 12 を指定します。

byOutFormat

出力フォーマットを、以下の値で指定します。

- OBR_NUMSYSON : 先頭の 0 出力あり
- OBR_NUMSYSOFF : 先頭の 0 出力なし

byCheckDigit

チェックデジットを、以下の値で指定します。

チェックデジット計算なしを設定する場合は、UPC-E のチェックデジット設定についても、チェックデジット計算なしを設定してください。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタを、以下の値で指定します。

- OBR_CHKKOFF : チェックキャラクタ出力なし
- OBR_CHKKON : チェックキャラクタ出力あり

戻り値

- OBR_OK : 正常終了
- OBR_PRM : パラメータエラー
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、チェックデジット計算を無効に設定することはできません。OBR_CHKDOFF を指定しても OBR_CHKDON になります。

4.60 OBRGetUPCEAddonOption

UPC-E Addon コードのデコードオプションを取得します。

```
[C++]
int OBRGetUPCEAddonOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetUPCEAddonOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetUPCEAddonOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

UPC-E Addon コードのデコードオプションを取得します。

パラメータ

pEnable

UPC-E コードの読み取り許可を取得します。取得する値は OBRSetUPCEAddonOptionEx 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetUPCEAddonOptionEx 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetUPCEAddonOptionEx 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は OBRSetUPCEAddonOptionEx 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は OBRSetUPCEAddonOptionEx 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は OBRSetUPCEAddonOptionEx 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
FUNCTION_UN SUPPORT	: 未サポートエラー

4.61 OBRSetUPCEOption

UPC-E コードのデコードオプションを設定します。

```
[C++]
int OBRSetUPCEOption(
    BYTE byEnable,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetUPCEOption( _
    ByVal byEnable As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetUPCEOption(
    Byte byEnable,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

UPC-E コードのデコードオプションを設定します。

パラメータ

byEnable

UPC-E コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : UPC-E の読み取り有効
- OBR_CODE_DISABLE : UPC-E の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byCheckDigit

チェックデジット設定を、以下の値で指定します。

チェックデジット計算なしを設定する場合は、UPC-E Addon のチェックデジット設定についても、チェックデジット計算なしを設定してください。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタ設定を、以下の値で指定します。

- OBR_CHKKOFF : チェックキャラクタ出力なし
- OBR_CHKKON : チェックキャラクタ出力あり

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UN SUPPORT	: 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、チェックデジット計算を無効に設定することはできません。
OBR_CHKDOFF を指定しても OBR_CHKDON になります。

4.62 OBRSetUPCEOptionEx

UPC-E コードの読み取り設定をセットします。

```
[C++]
int OBRSetUPCEOptionEx(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetUPCEOptionEx( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetUPCEOptionEx(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

UPC-E コードの読み取り設定をセットします。

パラメータ

byEnable

UPC-E コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : UPC-E の読み取り有効
- OBR_CODE_DISABLE : UPC-E の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。UPC-E コードの読み取り有効最小桁数は、7 桁固定で変更できません。必ず 7 を指定します。

byMaxLength

最大桁数を指定します。UPC-E コードの読み取り有効最小桁数は、7 桁固定で変更できません。必ず 7 を指定します。

byOutFormat

出力フォーマットを、以下の値で指定します。

- OBR_NUMSYSON : 先頭の 0 出力あり
- OBR_NUMSYSOFF : 先頭の 0 出力なし
- OBR_UPCEGTIN : GTIN フォーマット(14 桁)で出力

byCheckDigit

チェックデジットを、以下の値で指定します。

チェックデジット計算なしを設定する場合は、UPC-E Addon のチェックデジット設定についても、チェックデジット計算なしを設定してください。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタ設定を、以下の値で指定します。

- OBR_CHKKOFF : チェックキャラクタ出力なし
- OBR_CHKKON : チェックキャラクタ出力あり

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、チェックデジット計算を無効に設定することはできません。OBR_CHKDOFF を指定しても OBR_CHKDON になります。

4.63 OBRGetUPCEOption

UPC-E コードのデコードオプションを取得します。

```
[C++]
int OBRGetUPCEOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetUPCEOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetUPCEOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

UPC-E コードのデコードオプションを取得します。

パラメータ

pEnable

UPC-E コードの読み取り許可を取得します。取得する値は OBRSetUPCEOptionEx 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetUPCEOptionEx 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetUPCEOptionEx 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetUPCEOptionEx` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetUPCEOptionEx` 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は `OBRSetUPCEOptionEx` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー <code>Device Emulator</code> では発生しません
<code>FUNCTION_UN SUPPORT</code>	: 未サポートエラー

4.64 OBRSetIDFOption

IDF(Industrial 2 of 5)コードのデコードオプションを設定します。

```
[C++]
int OBRSetIDFOption(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetIDFOption( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetIDFOption(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

IDF(Industrial 2 of 5)コードのデコードオプションを設定します。

パラメータ

byEnable

Industrial 2 of 5 コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : Industrial 2 of 5 の読み取り有効
- OBR_CODE_DISABLE : Industrial 2 of 5 の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定。デフォルトは 2 桁です。

最小桁数より小さな Industrial 2 of 5 データのデコードは行いません。

byMaxLength

最大桁数を指定。デフォルトは 67 桁です。

最大桁数より大きな Industrial 2 of 5 データのデコードは行いません。

IT-G500/DT-X100/DT-X200 では、最大桁数に 48 以上 67 以下の値を指定した場合、最大桁数は

48 となり、本関数は正常終了します。

byCheckDigit

チェックデジットを、以下の値で指定します。

OBR_CHKDOFF : チェックデジット計算なし
OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタを、以下の値で指定します。

OBR_CHKKOFF : チェックキャラクタ出力なし
OBR_CHKKON : チェックキャラクタ出力あり

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
OBR_PRM : パラメータエラー
FUNCTION_UN SUPPORT : 未サポートエラー

4.65 OBRGetIDFOption

IDF(Industrial 2 of 5)コードのデコードオプションを取得します。

```
[C++]
int OBRGetIDFOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetIDFOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetIDFOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

IDF(Industrial 2 of 5)コードのデコードオプションを取得します。

パラメータ

pEnable

IDF コードの読み取り許可を取得します。取得する値は `OBRSetIDFOption` 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は `OBRSetIDFOption` 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は `OBRSetIDFOption` 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetIDFOption` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は OBRSetIDFOption 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は OBRSetIDFOption 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.66 OBRSetITFOption

Interleaved 2 of 5(ITF)コードのデコードオプションを設定します。

```
[C++]
int OBRSetITFOption(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetITFOption( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetITFOption(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

Interleaved 2 of 5(ITF)コードのデコードオプションを設定します。

パラメータ

byEnable

Interleaved 2 of 5 コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : Interleaved 2 of 5 の読み取り有効
- OBR_CODE_DISABLE : Interleaved 2 of 5 の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 4 桁です。

最小桁数より小さな Interleaved 2 of 5 データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 94 桁です。

最大桁数より大きな Interleaved 2 of 5 データのデコードは行いません。

IT-G500/DT-X100/DT-X200 では、最大桁数に 60 以上 94 以下の値を指定した場合、最大桁数は

60 となり、本関数は正常終了します。

byCheckDigit

チェックデジットを、以下の値で指定します。

OBR_CHKDOFF : チェックデジット計算なし
OBR_CHKDON : チェックデジット計算あり

byCheckChar

チェックキャラクタを、以下の値で指定します。

OBR_CHKKOFF : チェックキャラクタ出力なし
OBR_CHKKON : チェックキャラクタ出力あり

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
OBR_PRM : パラメータエラー
FUNCTION_UN SUPPORT : 未サポートエラー

4.67 OBRGetITFOption

ITF(Interleaved 2 of 5)コードのデコードオプションを取得します。

```
[C++]
int OBRGetITFOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetITFOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetITFOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

ITF(Interleaved 2 of 5)コードのデコードオプションを取得します。

パラメータ

pEnable

ITF コードの読み取り許可を取得します。取得する値は `OBRSetITFOption` 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は `OBRSetITFOption` 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は `OBRSetITFOption` 関数を参照してください。

pOutFormat

NULL を取得します。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetITFOption` 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は `OBRSetITFOption` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー Device Emulator では発生しません
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.68 OBRSetCode93Option

Code93 コードのデコードオプションを設定します。

```
[C++]
int OBRSetCode93Option(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byCheckDigit,
)
```

```
[Visual Basic]
Public Shared Function OBRSetCode93Option( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byCheckDigit As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetCode93Option(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byCheckDigit
);
```

解説

Code93 コードのデコードオプションを設定します。

パラメータ

byEnable

Code 93 コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : Code 93 の読み取り有効
- OBR_CODE_DISABLE : Code 93 の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 3 桁です。

最小桁数より小さな Code 93 データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 70 桁です。

最大桁数より大きな Code 93 データのデコードは行いません。

IT-G500/DT-X100/DT-X200 では、最大桁数に 60 以上 70 以下の値を指定した場合、最大桁数は 60 となり、本関数は正常終了します。

byCheckDigit

チェックデジット設定を、以下の値で指定します。

- OBR_CHKDOFF : チェックデジット計算なし
- OBR_CHKDON : チェックデジット計算あり

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、チェックデジット計算を無効に設定することはできません。
OBR_CHKDOFF を指定しても OBR_CHKDON になります。

4.69 OBRGetCode93Option

Code93 コードのデコードオプションを取得します。

```
[C++]
int OBRGetCode93Option(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetCode93Option( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetCode93Option(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

Code93 コードのデコードオプションを取得します。

パラメータ

pEnable

Code93 コードの読み取り許可を取得します。取得する値は OBRSetCode93Option 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetCode93Option 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetCode93Option 関数を参照してください。

pOutFormat

NULL を取得します。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetCode93Option` 関数を参照してください。

pCheckChar

NULL を取得します。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー <code>Device Emulator</code> では発生しません
<code>FUNCTION_UN SUPPORT</code>	: 未サポートエラー

4.70 OBRSetCode128Option

Code128 コードのデコードオプションを設定します。

```
[C++]
int OBRSetCode128Option(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
)
```

```
[Visual Basic]
Public Shared Function OBRSetCode128Option( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetCode128Option(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit
);
```

解説

Code128 コードのデコードオプションを設定します。

パラメータ

byEnable

Code128 コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : Code128 の読み取り有効
- OBR_CODE_DISABLE : Code128 の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 2 桁です。

最小桁数より小さな Code128 データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 98 桁です。

最大桁数より大きな Code128 データのデコードは行いません。

byOutFormat

出力フォーマットを、以下の値で指定します。

OBR_128AON	: ASCII 変換出力
OBR_128AOFF	: ASCII 変換なし出力
OBR_128EAN	: EAN-128 のみ出力
OBR_128AIM	: EAN-128 のみ出力、コード ID あり
OBR_128FNC	: EAN-128 のみ出力、Fnc→GS 変換

出力フォーマットには、上記以外に下記の値の論理和を指定することができます。

OBR_128AIML	: EAN-128 読み取り時、コード ID 付加
OBR_128FNCL	: EAN-128 読み取り時、Fnc→GS 変換
OBR_128CAOF	: Code128 読み取り時、ASCII 変換なし
OBR_128EAOF	: EAN-128 読み取り時、ASCII 変換なし
OBR_128FNC2	: Code128/EAN-128 読み取り時、FNC2 連結出力あり
OBR_128FNC4	: Code128/EAN-128 読み取り時、FNC4 拡張 ASCII 変換あり

※ EAN-128 コードは 2006 年 1 月に GS1-128 に名称変更されています。

byCheckDigit

チェックデジットを、以下の値で指定します。

OBR_CHKDOFF	: チェックデジット計算なし
OBR_CHKDON	: チェックデジット計算あり

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

注意

Code128 はチェックデジット計算ありに設定してください。

特別な理由によりチェックデジット計算なしに設定する場合は、印字品質や条件によって誤読する可能性があるため、動作確認を十分に行ってください。

IT-G500/DT-X100/DT-X200 では、チェックデジット計算を無効に設定することはできません。

OBR_CHKDOFF を指定しても OBR_CHKDON になります。また、ASCII 変換なしを設定することはできません。OBR_128AOFF を指定しても OBR_128AON になります。

4.71 OBRGetCode128Option

Code128 コードのデコードオプションを取得します。

```
[C++]
int OBRGetCode128Option(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetCode128Option( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetCode128Option(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

パラメータ

pEnable

Code128 コードの読み取り許可を取得します。取得する値は OBRSetCode128Option 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetCode128Option 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetCode128Option 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は OBRSetCode128Option 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetCode128Option` 関数を参照してください。

pCheckChar

NULL を取得します。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー Device Emulator では発生しません
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.72 OBRSetMSIOption

MSI コードのデコードオプションを設定します。

```
[C++]
int OBRSetMSIOption(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetMSIOption( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetMSIOption(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

MSI コードのデコードオプションを設定します。

パラメータ

byEnable

MSI コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : MSI の読み取り有効
- OBR_CODE_DISABLE : MSI の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 1 桁です。

最小桁数より小さな MSI データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 57 桁です。

最大桁数より大きな MSI データのデコードは行いません。

IT-G500/DT-X100/DT-X200 では、最大桁数に 50 以上 57 以下の値を指定した場合、最大桁数は

50 となり、本関数は正常終了します。

byCheckDigit

チェックデジットを、以下の値で指定します。

OBR_CHKDOFF : チェックデジット計算なし
OBR_CHKDON : チェックデジット 1 桁 mod10
OBR_CDMSEV : チェックデジット 2 桁 1st:mod11 2nd:mod10
OBR_CDMSTN : チェックデジット 2 桁 1st:mod10 2nd:mod10

byCheckChar

チェックキャラクタを、以下の値で指定します。

OBR_CHKKOFF : チェックキャラクタ出力なし
OBR_CHKKON : チェックキャラクタ出力あり

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
OBR_PRM : パラメータエラー
FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、OBR_CHKDOFF、OBR_CDMSEV を設定することはできません。指定しても、OBR_CHKDON として動作します。

4.73 OBRGetMSIOption

MSI コードの読み取り設定を取得します。

```
[C++]
int OBRGetMSIOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetMSIOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetMSIOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

MSI コードの読み取り設定を取得します。

パラメータ

pEnable

MSI コードの読み取り許可を取得します。取得する値は `OBRSetMSIOption` 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は `OBRSetMSIOption` 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は `OBRSetMSIOption` 関数を参照してください。

pOutFormat

NULL を取得します。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetMSIOption` 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は `OBRSetMSIOption` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー Device Emulator では発生しません
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.74 OBRSetIATAOption

IATA コードのデコードオプションを設定します。

```
[C++]
int OBRSetIATAOption(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byCheckDigit,
)
```

```
[Visual Basic]
Public Shared Function OBRSetIATAOption( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byCheckDigit As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetIATAOption(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byCheckDigit
);
```

解説

IATA コードのデコードオプションを設定します。

パラメータ

bEnable

IATA コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : IATA の読み取り有効
- OBR_CODE_DISABLE : IATA の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 4 桁です。

最小桁数より小さな IATA データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 65 桁です。

最大桁数より大きな IATA データのデコードは行いません。

IT-G500/DT-X100/DT-X200 では、最大桁数に 50 以上 65 以下の値を指定した場合、最大桁数は 50 となり、本関数は正常終了します。

byCheckDigit

チェックデジットを、以下の値で指定します。

- OBR_CHKDOFF : チェックデジット計算なし(読取桁数 1-40)
- OBR_CHKDON : 最終キャラクタ以外を対象(読取桁数 2-40)
- OBR_CDMSEV : クーポン番号/数値部を対象(読取桁数 15-17)
- OBR_CDMSTN : 数値部を対象(読取桁数 15-17)
- OBR_CHKDON4 : モジュラス 10 のチェックデジットによりチェックを行います
(DT-5200 以外)

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.75 OBRGetIATAOption

IATA コードのデコードオプションを取得します。

```
[C++]
int OBRGetIATAOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetIATAOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetIATAOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

IATA コードのデコードオプションを取得します。

パラメータ

pEnable

IATA コードの読み取り許可を取得します。取得する値は OBRSetIATAOption 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetIATAOption 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetIATAOption 関数を参照してください。

pOutFormat

NULL を取得します。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetIATAOption` 関数を参照してください。

pCheckChar

NULL を取得します。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー Device Emulator では発生しません
<code>FUNCTION_UN SUPPORT</code>	: 未サポートエラー

4.76 OBRSetRSS14Option

RSS-14 コードのデコードオプションを設定します。

```
[C++]
int OBRSetRSS14Option(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetRSS14Option( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetRSS14Option(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

RSS-14 コードのデコードオプションを設定します。

※ RSS-14 コードは 2007 年 2 月に GS1 DataBar Omnidirectional に名称変更されています。

パラメータ

byEnable

RSS-14 コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : RSS-14 の読み取り有効
- OBR_CODE_DISABLE : RSS-14 の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。RSS-14 コードの読み取り有効最小桁数は 14 桁固定で変更できません。必ず 14 を指定します。

byMaxLength

最大桁数を指定します。RSS-14 コードの読み取り有効最大桁数は、14 桁固定で変更できません。必ず 14 を指定します。

byOutFormat

出力フォーマットを、以下の値で指定します。

- OBR_RS14NORM : 標準出力
- OBR_RS14AIOF : アプリケーション識別子出力なし(先頭の“01”を出力しない)

byCheckDigit

チェックデジットを指定します。RSS-14 コードのチェックデジットは常に有効となるため、必ず OBR_CHKDON を指定します。

byCheckChar

チェックキャラクタ出力を指定します。RSS-14 コードのチェックキャラクタ出力は常に有効となるため、必ず OBR_CHKON を指定します。

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.77 OBRGetRSS14Option

RSS-14 コードのデコードオプションを取得します。

```
[C++]
int OBRGetRSS14Option(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetRSS14Option( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetRSS14Option(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

RSS-14 コードのデコードオプションを取得します。

※ RSS-14 コードは 2007 年 2 月に GS1 DataBar Omnidirectional に名称変更されています。

パラメータ

pEnable

RSS-14 コードの読み取り許可を取得します。取得する値は OBRSetRSS14Option 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetRSS14Option 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetRSS14Option 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetRSS14Option` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetRSS14Option` 関数を参照してください。

pCheckChar

チェックキャラクタ出力を取得します。取得する値は `OBRSetRSS14Option` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー Device Emulator では発生しません
<code>FUNCTION_UN SUPPORT</code>	: 未サポートエラー

4.78 OBRSetRSSLimitedOption

RSS Limited コードのデコードオプションを設定します。

```
[C++]
int OBRSetRSSLimitedOption(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetRSSLimitedOption( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetRSSLimitedOption(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

RSS Limited コードのデコードオプションを設定します。

※ RSS Limited コードは 2007 年 2 月に GS1 DataBar Limited に名称変更されています。

パラメータ

byEnable

RSS Limited コードの読み取り許可を以下の値で指定します。

- OBR_CODE_ENABLE : RSS Limited の読み取り有効
- OBR_CODE_DISABLE : RSS Limited の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。RSS Limited コードの読み取り有効最小桁数は、14 桁固定で変更できません。必ず NULL を指定します。

byMaxLength

最大桁数を指定します。RSS Limited コードの読み取り有効最大桁数は、14 桁固定で変更できません。必ず NULL を指定します。

byOutFormat

出力フォーマットを以下の値で指定します。

- OBR_RSLMNORM : 標準出力
- OBR_RSLMAIOF : アプリケーション識別子出力なし(先頭の“01”を出力しない)

byCheckDigit

チェックデジットを指定します。RSS Limited コードのチェックデジット計算の設定は常に有効となるため、必ず OBR_CHKDON を指定します。

byCheckChar

チェックキャラクタを指定します。RSS Limited コードのチェックキャラクタ出力の設定は常に有効となるため、必ず OBR_CHKKON を指定します。

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.79 OBRGetRSSLimitedOption

RSS Limited コードのデコードオプションを取得します。

```
[C++]
int OBRGetRSSLimitedOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetRSSLimitedOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetRSSLimitedOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

RSS Limited コードのデコードオプションを取得します。

※ RSS Limited コードは 2007 年 2 月に GS1 DataBar Limited に名称変更されています。

パラメータ

pEnable

RSS Limited コードの読み取り許可読み取り許可を取得します。取得する値は OBRSetRSSLimitedOption 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetRSSLimitedOption 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetRSSLimitedOption 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetRSSLimitedOption` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetRSSLimitedOption` 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は `OBRSetRSSLimitedOption` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー <code>Device Emulator</code> では発生しません
<code>FUNCTION_UNSupport</code>	: 未サポートエラー

4.80 OBRSetRSSEExpandedOption

RSS Expanded コードのデコードオプションを設定します。

```
[C++]
int OBRSetRSSEExpandedOption(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetRSSEExpandedOption( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetRSSEExpandedOption(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

RSS Expanded コードのデコードオプションを設定します。

※ RSS Expanded コードは 2007 年 2 月に GS1 DataBar Expanded に名称変更されています。

パラメータ

byEnable

RSS Expanded コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : RSS Expanded の読み取り有効
- OBR_CODE_DISABLE : RSS Expanded の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 1 桁です。

最小桁数より小さな RSS Expanded データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 74 桁です。
最大桁数より大きな RSS Expanded データのデコードは行いません。

byOutFormat

出力フォーマットを指定します。
RSS Expanded コードには出力フォーマット設定がないため、必ず NULL を指定します。

byCheckDigit

チェックデジット計算の設定を指定します。
RSS Expanded コードのチェックデジット計算の設定は常に有効となるため、必ず OBR_CHKDON を指定します。

byCheckChar

チェックキャラクタ出力の設定を指定します。
RSS Expanded コードのチェックキャラクタ出力の設定は常に有効となるため、必ず OBR_CHKON を指定します。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.81 OBRGetRSSEExpandedOption

RSS Expanded コードのデコードオプションを取得します。

```
[C++]
int OBRGetRSSEExpandedOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetRSSEExpandedOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetRSSEExpandedOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

RSS Expanded コードのデコードオプションを取得します。

※ RSS Expanded コードは 2007 年 2 月に GS1 DataBar Expanded に名称変更されています。

パラメータ

pEnable

RSS Expanded コードの読み取り許可を取得します。取得する値は OBRSetRSSEExpandedOption 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetRSSEExpandedOption 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は OBRSetRSSEExpandedOption 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetRSSEExpandedOption` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetRSSEExpandedOption` 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は `OBRSetRSSEExpandedOption` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー <code>Device Emulator</code> では発生しません
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.82 OBRSetRSS14StackedOption

RSS-14 Stacked コードのデコードオプションを設定します。

```
[C++]
int OBRSetRSS14StackedOption(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetRSS14StackedOption( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetRSS14StackedOption(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

RSS-14 Stacked コードのデコードオプションを設定します。
Device Emulator では、常に FUNCTION_UNSUPPORTED を返します。

※ RSS-14 Stacked コードは 2007 年 2 月に GS1 DataBar Stacked に名称変更されています。

パラメータ

byEnable

RSS-14 Stacked コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : RSS-14 Stacked の読み取り有効
- OBR_CODE_DISABLE : RSS-14 Stacked の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。RSS-14 Stacked コードの読み取り有効最小桁数は 14 桁固定で変更できま

せん。必ず 14 を指定します。

byMaxLength

最大桁数を指定します。RSS-14 Stacked コードの読み取り有効最大桁数は、14 桁固定で変更できません。必ず 14 を指定します。

byOutFormat

出力フォーマットを、以下の値で指定します。

- OBR_RS14NORM : 標準出力
- OBR_RS14AIOF : アプリケーション識別子出力なし(先頭の“01”を出力しない)

byCheckDigit

チェックデジットを指定します。RSS-14 Stacked コードのチェックデジットは常に有効となるため、必ず OBR_CHKDON を指定します。

byCheckChar

チェックキャラクタ出力を指定します。RSS-14 Stacked コードのチェックキャラクタ出力は常に有効となるため、必ず OBR_CHKKON を指定します。

戻り値

- OBR_OK : 正常終了
- OBR_PON : オープン済み
- OBR_NOT_DEVICE : スキャナドライバエラー
- OBR_PRM : パラメータエラー
- FUNCTION_UNSUPPORTED : 未サポートエラー

4.83 OBRGetRSS14StackedOption

RSS-14 Stacked コードのデコードオプションを取得します。

```
[C++]
int OBRGetRSS14StackedOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetRSS14StackedOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetRSS14StackedOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

RSS-14 Stacked コードのデコードオプションを取得します。
Device Emulator では、常に `FUNCTION_UNSUPPORTED` を返します。

※ RSS-14 Stacked コードは 2007 年 2 月に GS1 DataBar Stacked に名称変更されています。

パラメータ

pEnable

RSS-14 Stacked コードの読み取り許可を取得します。取得する値は `OBRSetRSS14StackedOption` 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は `OBRSetRSS14StackedOption` 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は `OBRSetRSS14StackedOption` 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetRSS14StackedOption` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetRSS14StackedOption` 関数を参照してください。

pCheckChar

チェックキャラクタ出力を取得します。取得する値は `OBRSetRSS14StackedOption` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.84 OBRSetRSSExpandedStackedOption

RSS Expanded Stacked コードのデコードオプションを設定します。

```
[C++]
int OBRSetRSSExpandedStackedOption(
    BYTE byEnable,
    BYTE byMinLength,
    BYTE byMaxLength,
    BYTE byOutFormat,
    BYTE byCheckDigit,
    BYTE byCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRSetRSSExpandedStackedOption( _
    ByVal byEnable As Byte, _
    ByVal byMinLength As Byte, _
    ByVal byMaxLength As Byte, _
    ByVal byOutFormat As Byte, _
    ByVal byCheckDigit As Byte, _
    ByVal byCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRSetRSSExpandedStackedOption(
    Byte byEnable,
    Byte byMinLength,
    Byte byMaxLength,
    Byte byOutFormat,
    Byte byCheckDigit,
    Byte byCheckChar
);
```

解説

RSS Expanded Stacked コードのデコードオプションを設定します。
Device Emulator では、常に FUNCTION_UNSUPPORTED を返します。

※ RSS Expanded Stacked コードは 2007 年 2 月に GS1 DataBar Expanded Stacked に名称変更されています。

パラメータ

byEnable

RSS Expanded Stacked コードの読み取り許可を、以下の値で指定します。

- OBR_CODE_ENABLE : RSS Expanded Stacked の読み取り有効
- OBR_CODE_DISABLE : RSS Expanded Stacked の読み取り無効
- OBR_CODE_IGNORE : 現在の設定継続

byMinLength

最小桁数を指定します。デフォルトは 1 桁です。
最小桁数より小さな RSS Expanded Stacked データのデコードは行いません。

byMaxLength

最大桁数を指定します。デフォルトは 74 桁です。
最大桁数より大きな RSS Expanded Stacked データのデコードは行いません。

byOutFormat

出力フォーマットを指定します。
RSS Expanded Stacked コードには出力フォーマット設定がないため、必ず NULL を指定します。

byCheckDigit

チェックデジット計算の設定を指定します。
RSS Expanded Stacked コードのチェックデジット計算の設定は常に有効となるため、必ず OBR_CHKDON を指定します。

byCheckChar

チェックキャラクタ出力の設定を指定します。
RSS Expanded Stacked コードのチェックキャラクタ出力の設定は常に有効となるため、必ず OBR_CHKON を指定します。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.85 OBRGetRSSExpandedStackedOption

RSS Expanded Stacked コードのデコードオプションを取得します。

```
[C++]
int OBRGetRSSExpandedStackedOption(
    BOOL *pEnable,
    BYTE *pMinLength,
    BYTE *pMaxLength,
    BYTE *pOutFormat,
    BYTE *pCheckDigit,
    BYTE *pCheckChar
)
```

```
[Visual Basic]
Public Shared Function OBRGetRSSExpandedStackedOption( _
    ByRef pEnable As Boolean, _
    ByRef pMinLength As Byte, _
    ByRef pMaxLength As Byte, _
    ByRef pOutFormat As Byte, _
    ByRef pCheckDigit As Byte, _
    ByRef pCheckChar As Byte _
) As Int32
```

```
[C#]
public static Int32 OBRGetRSSExpandedStackedOption(
    ref Boolean pEnable,
    ref Byte pMinLength,
    ref Byte pMaxLength,
    ref Byte pOutFormat,
    ref Byte pCheckDigit,
    ref Byte pCheckChar
);
```

解説

RSS Expanded Stacked コードのデコードオプションを取得します。
Device Emulator では、常に FUNCTION_UNSUPPORTED を返します。

※ RSS Expanded Stacked コードは 2007 年 2 月に GS1 DataBar Expanded Stacked に名称変更されています。

パラメータ

pEnable

RSS Expanded Stacked コードの読み取り許可を取得します。取得する値は OBRSetRSSExpandedStackedOption 関数を参照してください。

pMinLength

最小桁数を取得します。取得する値は OBRSetRSSExpandedStackedOption 関数を参照してください。

pMaxLength

最大桁数を取得します。取得する値は `OBRSetRSSEExpandedStackedOption` 関数を参照してください。

pOutFormat

出力フォーマットを取得します。取得する値は `OBRSetRSSEExpandedStackedOption` 関数を参照してください。

pCheckDigit

チェックデジットを取得します。取得する値は `OBRSetRSSEExpandedStackedOption` 関数を参照してください。

pCheckChar

チェックキャラクタを取得します。取得する値は `OBRSetRSSEExpandedStackedOption` 関数を参照してください。

戻り値

<code>OBR_OK</code>	: 正常終了
<code>OBR_NOT_DEVICE</code>	: スキャナドライバエラー
<code>FUNCTION_UNSUPPORTED</code>	: 未サポートエラー

4.86 OBRSaveConfigFile

レーザスキャナドライバの全ての設定を保存します。

```
[C++]  
int OBRSaveConfigFile()
```

```
[Visual Basic]  
Public Shared Function OBRSaveConfigFile() As Int32
```

```
[C#]  
public static Int32 OBRSaveConfigFile()
```

解説

レーザスキャナドライバの全ての設定を、所定のファイルに保存します。

パラメータ

なし

戻り値

OBR_OK	: 正常終了
OBR_PRM	: パラメータエラー
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

補足

ファイル名は「OBRDRV.ini」となります。以下のフォルダに保存されます。
“FlashDisk¥System Settings¥OBRDRV.ini”

Device Emulator では、常に“¥FlashDisk¥System Settings¥OBRDRV.ini”に保存します。

設定ファイルについては、「設定ファイルの書式」を参照してください。

4.87 OBRLoadConfigFile

レーザスキャナドライバの全てのデフォルト設定を変更します。

```
[C++]  
int OBRLoadConfigFile()
```

```
[Visual Basic]  
Public Shared Function OBRLoadConfigFile() As Int32
```

```
[C#]  
public static Int32 OBRLoadConfigFile()
```

解説

所定のファイルに保存される設定を読み出して、レーザスキャナドライバのデフォルト設定を変更します。OBRLoadConfigFile 関数を呼び出しただけでは、デフォルト設定が変更されるだけなので、設定ファイルに記述される設定をレーザスキャナドライバに反映させるためには、OBRLoadConfigFile 関数を呼び出した後に、OBRSetDefaultSymbology 関数を呼び出す必要があります。

パラメータ

なし

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー
FUNCTION_UNSupport	: 未サポートエラー

補足

設定ファイルについては、「設定ファイルの書式」を参照してください。

4.88 OBRSetFilter

ノイズフィルタモードを設定します。

```
[C++]
int OBRSetFilter(
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetFilter( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetFilter(
    Int32 dwMode
);
```

解説

ノイズフィルタモードを設定します。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetFilter 関数を実行することにより、設定値を確認することができます。

パラメータ

dwMode

ノイズフィルタモードを、以下の値で指定します。

OBR_NOFILTER	: ノイズフィルタ無効
OBR_SOFTFILTER	: ソフトウェアフィルタ有効
OBR_HARDFILTER	: ハードウェアフィルタ有効
OBR_SOFTHARDFILTER	: ソフトウェアフィルタ&ハードウェアフィルタ有効

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、ノイズフィルタモードはノイズフィルタ無効固定です。その他のフィルタモードを指定しても、ノイズフィルタ無効で動作します。

4.89 OBRGetFilter

ノイズフィルタモードを取得します。

```
[C++]
int OBRGetFilter(
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetFilter( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetFilter(
    ref Int32 pMode
);
```

解説

ノイズフィルタモードを取得します。

パラメータ

pMode

ノイズフィルタモードを取得します。取得する値は **OBRSetFilter** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
OBR_NOT_DEVICE_DECODE	: Decode 常駐失敗
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.90 OBRSetFilterOnTimer

ノイズフィルタ開始時間を設定します。

```
[C++]
int OBRSetFilterOnTimer (
    int nTime
)
```

```
[Visual Basic]
Public Shared Function OBRSetFilterOnTimer ( _
    ByVal nTime As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetFilterOnTimer (
    Int32 nTime
);
```

解説

デコードを開始してから、ノイズフィルタを開始するまでの時間を設定します。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetFilterOnTimer 関数を実行することにより、設定値を確認することができます。

パラメータ

nTime

デコード開始からノイズフィルタを有効にするまでの時間を、1～8 秒の範囲で指定します。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.91 OBRGetFilterOnTimer

ノイズフィルタ開始時間を取得します。

```
[C++]
int OBRGetFilterOnTimer (
    int *pTime
)
```

```
[Visual Basic]
Public Shared Function OBRGetFilterOnTimer ( _
    ByRef pTime As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetFilterOnTimer (
    ref Int32 pTime
);
```

解説

デコードを開始してから、ノイズフィルタを開始するまでの時間を取得します。

パラメータ

pTime

デコード開始からノイズフィルタを有効にするまでの時間を取得します。取得する値は OBRSetFilterOnTimer 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.92 OBRSetGainControl

ゲインコントロールの設定をします。

```
[C++]
int OBRSetGainControl (
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetGainControl ( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetGainControl (
    Int32 dwMode
);
```

解説

ゲインコントロールの設定をします。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetGainControl 関数を実行することにより、設定値を確認することができます。

パラメータ

dwMode

設定するゲインコントロール値を指定します。

OBR_GAIN_MODE0	: ゲインモード 0
OBR_GAIN_MODE1	: ゲインモード 1
OBR_GAIN_MODE2	: ゲインモード 2
OBR_GAIN_MODE3	: ゲインモード 3
OBR_GAIN_AUTO	: ゲインオート(DT-5200 以外)

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

注意

レーザ振れ角制御モード(OBRSetSwingAngle 関数)において振れ角大/振れ角中/振れ角小を設定し、ゲインコントロール設定(OBRSetGainControl 関数)においてゲインオートを設定している場合は、バーコードの印字品質のより読取精度が低下することがあります。

ゲインオートを設定する場合は、レーザ振れ角制御モードを振れ角制御なしに設定してください。

IT-G500/DT-X100/DT-X200 では、ゲインコントロールはゲインモード 0 固定です。その他のゲインコ

ントロールを指定しても、ゲインモード **0** で動作します。

4.93 OBRGetGainControl

ゲインコントロールの取得をします。

```
[C++]
int OBRGetGainControl (
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetGainControl ( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetGainControl (
    ref Int32 pMode
);
```

解説

ゲインコントロールの取得をします。

パラメータ

pMode

取得する値は OBRSetGainControl 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.94 OBRSetBarWidthAdjustment

読取バーコードのバーの幅にかける補正値を設定します。

```
[C++]
int OBRSetBarWidthAdjustment (
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetBarWidthAdjustment ( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetBarWidthAdjustment (
    Int32 dwMode
);
```

解説

読取バーコードのバーの幅にかける補正値を設定します。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetBarWidthAdjustment 関数を実行することにより、設定値を確認することができます。

パラメータ

dwMode

バーコードのバーの幅にかける補正値の設定を指定。デフォルト設定は「補正なし」。

OBR_BAR_NORMAL	: 補正なし
OBR_BAR_BLACK0	: 黒バーを細らせる
OBR_BAR_BLACK1	: 黒バーを大きく細らせる
OBR_BAR_WHITE0	: 白バーを細らせる
OBR_BAR_WHITE1	: 白バーを大きく細らせる
OBR_BAR_BLACK2	: 黒バーを太らせる
OBR_BAR_BLACK3	: 黒バーを大きく太らせる
OBR_BAR_WHITE2	: 白バーを太らせる
OBR_BAR_WHITE3	: 白バーを大きく太らせる

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、補正値は補正なし固定です。その他の補正値を指定しても、補正

なしで動作します。

4.95 OBRGetBarWidthAdjustment

読取バーコードのバーの幅にかける補正値を取得します。

```
[C++]
int OBRGetBarWidthAdjustment (
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetBarWidthAdjustment ( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetBarWidthAdjustment (
    ref Int32 pMode
);
```

解説

読取バーコードのバーの幅にかける補正値を設定します。

パラメータ

pMode

バーコードのバーの幅にかける補正値の設定を格納する **DWORD** 型変数へのポインタ。
取得する値は **OBRSetBarWidthAdjustment** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.96 OBRSetMarginCheckRatio

読取バーコードのマージンチェックの倍率の設定を行います。

```
[C++]
int OBRSetMarginCheckRatio (
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetMarginCheckRatio ( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetMarginCheckRatio (
    Int32 dwMode
);
```

解説

読取バーコードのマージンチェックの倍率の設定を行います。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetMarginCheckRatio 関数を実行することにより、設定値を確認することができます。

パラメータ

dwMode

バーコードのマージンチェックの倍率の設定を指定します。

OBR_MARGIN_WIDE	: マージンチェック倍率 大
OBR_MARGIN_MIDDLE	: マージンチェック倍率 中
OBR_MARGIN_NARROW	: マージンチェック倍率 小
OBR_MARGIN_MINIMUM	: マージンチェック倍率 最小

戻り値

OBR_OK	: 正常終了
OBR_PON	: オープン済み
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
OBR_PRM	: パラメータエラー
FUNCTION_UNSUPPORTED	: 未サポートエラー

注意

デフォルトは OBR_MARGIN_WIDE です。マージンが狭いコードを読む場合は、小さい倍率を指定してください。

4.97 OBRGetMarginCheckRatio

読取バーコードのマージンチェックの倍率の設定を取得します。

```
[C++]
int OBRGetMarginCheckRatio (
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetMarginCheckRatio ( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetMarginCheckRatio (
    ref Int32 pMode
);
```

解説

読取バーコードのマージンチェックの倍率の設定を取得します。

パラメータ

pMode

バーコードのマージンチェックの倍率の設定を格納する **DWORD** 型変数へのポインタ。
取得する値は **OBRSetMarginCheckRatio** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

4.98 OBRSetDecodeLearningMode

スキャナ学習機能の設定をセットします。

Device Emulator では、設定値を内部変数として格納するため、何も動作しませんが、OBRGetDecodeLearningMode 関数を実行することにより、設定値を確認することができます。

```
[C++]
int OBRSetDecodeLearningMode (
    DWORD dwMode
)
```

```
[Visual Basic]
Public Shared Function OBRSetDecodeLearningMode ( _
    ByVal dwMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRSetDecodeLearningMode (
    Int32 dwMode
);
```

解説

スキャナ学習機能の設定をセットします。

パラメータ

dwMode

スキャナ学習機能の設定を指定。デフォルト設定はスキャナ学習機能無効。

OBR_LEARNING_OFF : スキャナ学習機能無効
OBR_LEARNING_ON : スキャナ学習機能有効

戻り値

OBR_OK : 正常終了
OBR_PON : オープン済み
OBR_NOT_DEVICE : スキャナドライバエラー
Device Emulator では発生しません
OBR_PRM : パラメータエラー
FUNCTION_UNSUPPORTED : 未サポートエラー

注意

IT-G500/DT-X100/DT-X200 では、スキャナ学習機能はスキャナ学習機能無効固定です。その他のスキャナ学習機能を指定しても、ノイズフィルタ無効で動作します。

4.99 OBRGetDecodeLearningMode

スキャナ学習機能の設定を取得します。

```
[C++]
int OBRGetDecodeLearningMode (
    DWORD *pMode
)
```

```
[Visual Basic]
Public Shared Function OBRGetDecodeLearningMode ( _
    ByRef pMode As Int32 _
) As Int32
```

```
[C#]
public static Int32 OBRGetDecodeLearningMode (
    ref Int32 pMode
);
```

解説

スキャナ学習機能の設定を取得します。

パラメータ

pMode

スキャナ学習機能の設定を格納する **DWORD** 型変数へのポインタ。
取得する値は **OBRSetDecodeLearningMode** 関数を参照してください。

戻り値

OBR_OK	: 正常終了
OBR_NOT_DEVICE	: スキャナドライバエラー Device Emulator では発生しません
FUNCTION_UNSUPPORTED	: 未サポートエラー

5. プログラミング上の注意点

5.1 読み取り完了の通知について

本体に組み込まれているレーザスキャナドライバはデータを取得した際に、メッセージまたはイベントを発行します。(初期設定はメッセージによる通知のみの設定となっています)

メッセージによる通知は、WM_OBR_SCANNING (WM_USER + 0x530) のメッセージが発行されます。

また、メッセージの wParam には完了条件をあらわす以下の値が格納されます。

OBR_SUCCESS	: 正常終了
OBR_TIMEOUT	: タイムアウト発生
OBR_BUFFEROVER	: OBR バッファ出力エラー

OBRSetScanningNotification 関数を使用してイベントによる通知の設定を有効にすると、名前付きイベントにより通知が行われます。名前付きイベントは「OBRScanningEvent」という名称になっています。

また、イベント発生後に OBRGetLastEventStatus 関数を呼び出すことにより、イベントの発生要因を取得することができます。取得できる要因は、メッセージ通知の wParam で表される種類と同じものになります。

■メッセージによる通知方法に設定した場合

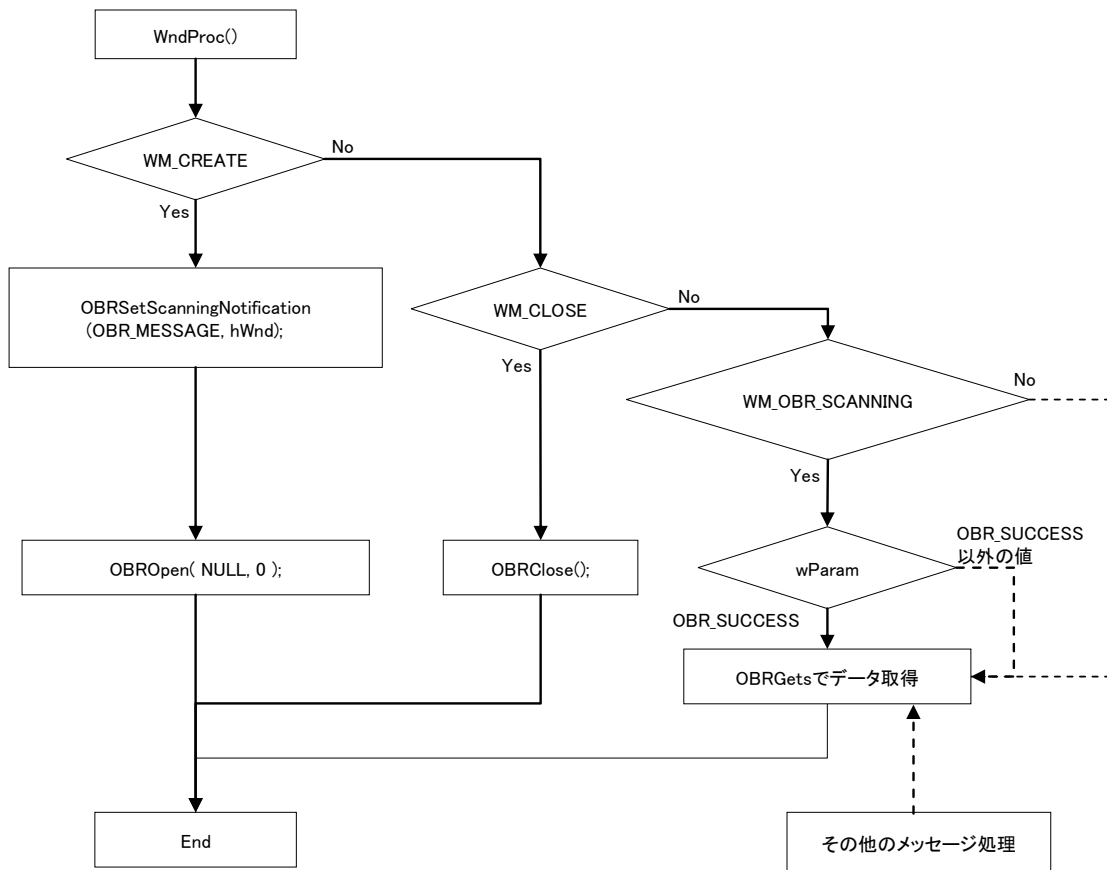


図 5.1 フローチャート(メッセージによる通知方法)

1. OBRSetScanningNotification 関数により、読み取り完了の通知方法をメッセージ出力に設定します。このとき、メッセージ送り先のウィンドウハンドルも指定します。
2. OBROpen 関数によりレーザスキャナを Open 状態にした後、読み取り通知のメッセージが発行されるまで待ちます。
3. メッセージを受け取った後、メッセージの wParam の値を確認します。
4. 読み取りに成功していることが確認できたら、各出力先に出力されたデータを処理します。
5. 読み取りを終了する場合は、OBRClose 関数を実行して下さい。

■ イベントによる通知方法に設定した場合

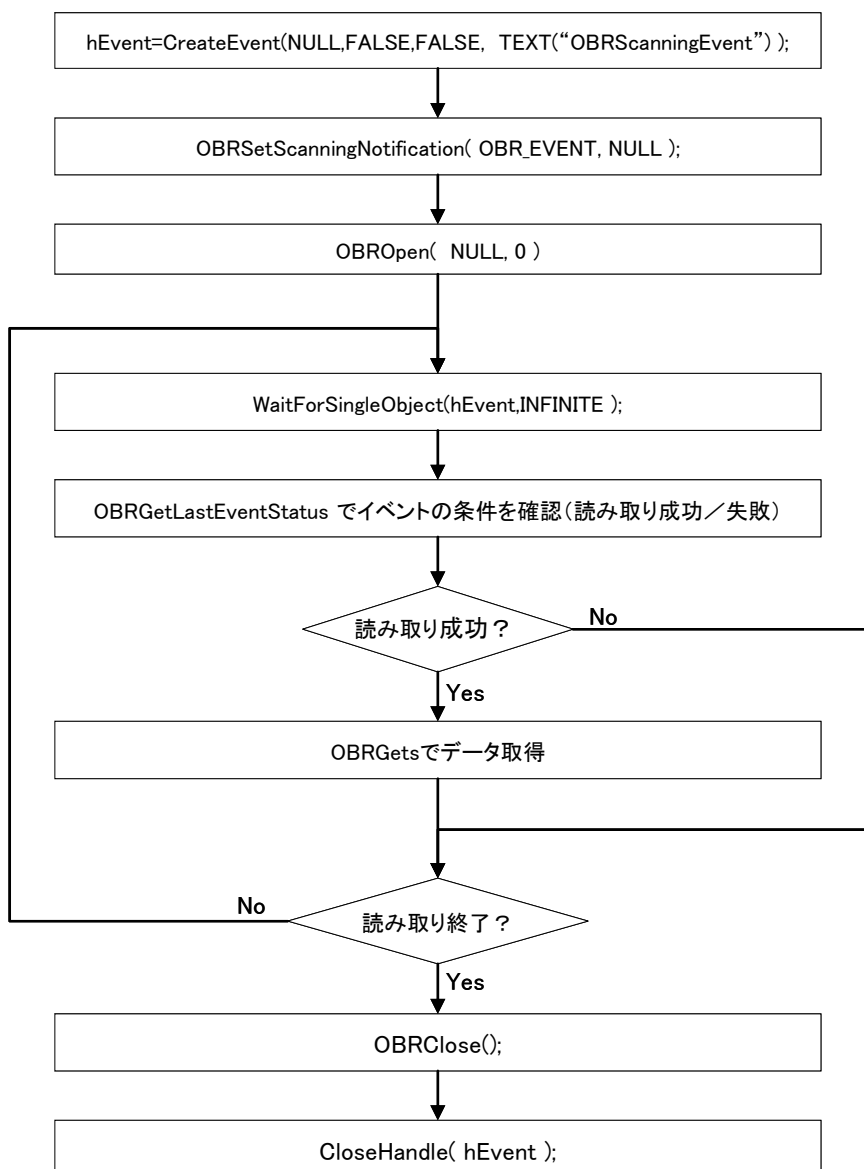


図 5.2 フローチャート(イベントによる通知方法)

1. 名前つきイベントの初期化を行います。
2. `OBRSetScanningNotification` 関数により、読み取り完了の通知方法をイベント出力に設定します。
3. `OBROpen` 関数によりレーザスキャナを `Open` 状態にした後、`WaitForSingleObject` 等で名前付きイベントが発行されるまで待ちます。
4. 名前付きイベントを受け取った後、`OBRGetLastEventStatus` 関数を呼び出し、イベントが発行された条件を確認します。
5. 読み取りに成功していることが確認できたら、各出力先に出力されたデータを処理します。
6. 読み取りを終了する場合は、`OBRClose` 関数を実行して下さい。

5.2 設定ファイルの書式

設定ファイルはテキスト形式で保存されます。保存される設定項目は、以下のとおりです。
 設定ファイルをエディタ等で変更した場合、OBRLoadConfigFile 関数を呼び出した後、
 OBRSetDefaultSymbology 関数を呼び出すことで、ドライバに設定内容を反映させることができます。
 設定ファイルによる指定と個々のコードに対するライブラリ関数による指定では、常に後に行った操作が優先されます。

表 5-1 設定項目一覧

項目	説明	値	
[Readable Bar Code]			
READCODE	各コードの読み取り許可／禁止 (複数のコードを同時に有効にする場合、これらの値の合計を指定します)	1	Code39 読み取り有効
		2	NW7 読み取り有効
		4	WPC Addon 読み取り有効
		8	WPC 読み取り有効
		16	UPC-E Addon 読み取り有効
		32	UPC-E 読み取り有効
		64	IDF 読み取り有効
		128	ITF 読み取り有効
		256	Code93 読み取り有効
		512	Code128 読み取り有効
		1024	MSI 読み取り有効
		2048	IATA 読み取り有効
		4096	RSS-14 有効
		8192	RSS Limited 有効
		16384	RSS Expanded 有効
32768	RSS-14 Stacked 有効(△2)		
65536	RSS Expanded Stacked 有効(△2)		
[CODE39 Option]			
CODE39_MIN	Code39 読み取り有効最小桁数	1～52	
CODE39_MAX	Code39 読み取り有効最大桁数	1～52	
CODE39_OUTFORMAT	Code39 出力フォーマット指定	0	Start/Stop コードあり
		1	Start/Stop コードなし
		2	Full ASCII 変換あり Start/Stop コードあり
		3	Full ASCII 変換あり Start/Stop コードなし
CODE39_CHKD	Code39 チェックデジット計算指定	0	無効
		1	有効
CODE39_CHKCH	Code39 チェックキャラクタ出力指定	0	無効
		1	有効
[NW7 Option]			
NW7_MIN	NW-7 読み取り有効最小桁数	1～63	
NW7_MAX	NW-7 読み取り有効最大桁数	1～63	
NW7_OUTFORMAT	NW-7 出力フォーマット指定	0	Start/Stop コードあり

		1	Start/Stop コードなし
NW7_CHKD	NW-7 チェックデジット計算指定	0	無効
		1	有効(DT-5200 以外)
NW7_CHKCH	NW-7 チェックキャラクタ出力指定	0	無効
		1	有効
[WPC Addon Option]			
WPCADDON_MIN	WPC Addon 読み取り有効最小桁数	10 (固定)	
WPCADDON_MAX	WPC Addon 読み取り有効最大桁数	18 (固定)	
WPCADDON_OUTFOR MAT	WPC Addon 出力フォーマット指定	0	先頭の 0 出力あり
		1	先頭の 0 出力なし
WPCADDON_CHKD	WPC Addon チェックデジット計算指定	0	無効
		1	有効
WPCADDON_CHKCH	WPC Addon チェックキャラクタ出力指定	0	無効 (固定)
[WPC Option]			
WPC_MIN	WPC 読み取り有効最小桁数	8 (固定)	
WPC_MAX	WPC 読み取り有効最大桁数	13 (固定)	
WPC_OUTFORMAT	WPC 出力フォーマット指定	0	先頭の 0 出力あり
		1	先頭の 0 出力なし
		14	GTIN 出力
WPC_CHKD	WPC チェックデジット計算指定	0	無効
		1	有効
WPC_CHKCH	WPC チェックキャラクタ出力指定	0	無効 (固定)
[UPCE Addon Option]			
UPCEADDON_MIN	UPC-E 読み取り有効最小桁数	9 (固定)	
UPCEADDON_MAX	UPC-E 読み取り有効最大桁数	12 (固定)	
UPCEADDON_OUTFOR MAT	UPC-E 出力フォーマット指定	0	先頭の 0 出力あり
		1	先頭の 0 出力なし
UPCEADDON_CHKD	UPC-E チェックデジット計算指定	0	無効
		1	有効
UPCEADDON_CHKCH	UPC-E チェックキャラクタ出力指定	0	無効
		1	有効
[UPCE Option]			
UPCE_MIN	UPC-E Addon 読み取り有効最小桁数	7 (固定)	
UPCE_MAX	UPC-E Addon 読み取り有効最大桁数	7 (固定)	
UPCE_OUTFORMAT	UPC-E Addon 出力フォーマット指定	0	先頭の 0 出力あり
		1	先頭の 0 出力なし
		2	GTIN 出力
UPCE_CHKD	UPC-E Addon チェックデジット計算指定	0	無効
		1	有効
UPCE_CHKCH	UPC-E Addon チェックキャラクタ出力指 定	0	無効
		1	有効
[IDF Option]			
IDF_MIN	IDF 読み取り有効最小桁数	2~67	
IDF_MAX	IDF 読み取り有効最大桁数	2~67	
IDF_OUTFORMAT	IDF 出力フォーマット指定	0	出力フォーマットなし(固定)
IDF_CHKD	IDF チェックデジット計算指定	0	無効
		1	有効

IDF_CHKCH	IDF チェックキャラクタ出力指定	0	無効
		1	有効
[ITF Option]			
ITF_MIN	ITF 読み取り有効最小桁数	2~94	
ITF_MAX	ITF 読み取り有効最大桁数	2~94	
ITF_OUTFORMAT	ITF 出力フォーマット指定	0	出力フォーマットなし(固定)
ITF_CHKD	ITF チェックデジット計算指定	0	無効
		1	有効
ITF_CHKCH	ITF チェックキャラクタ出力指定	0	無効
		1	有効
[CODE93 Option]			
CODE93_MIN	Code93 読み取り有効最小桁数	1~70	
CODE93_MAX	Code93 読み取り有効最大桁数	1~70	
CODE93_OUTFORMAT	Code93 出力フォーマット指定	0	出力フォーマットなし(固定)
CODE93_CHKD	Code93 チェックデジット計算指定	0	無効
		1	有効
CODE93_CHKCH	Code93 チェックキャラクタ出力指定	0	無効 (固定)
[CODE128 Option]			
CODE128_MIN	Code128 読み取り有効最小桁数	1~98	
CODE128_MAX	Code128 読み取り有効最大桁数	1~98	
CODE128_OUTFORMAT	Code128 出力フォーマット指定	0	Full ASCII 変換あり
		1	Full ASCII 変換なし
		2	EAN-128 のみ出力
		6	EAN-128 のみ出力 コード ID 出力
		10	EAN-128 のみ出力 Fnc→GS 変換
	Code128 出力フォーマット指定 (複数のフォーマットを同時に指定する 場合、これらの値の合計を指定します) (△)	2	EAN-128 のみ出力
		4	EAN-128 コード ID 付加
		8	EAN-128 Fnc→GS 変換
		16	Code128 Full ASCII 変換なし
		32	EAN-128 Full ASCII 変換なし
64	Code128/EAN-128 FNC2 連結出力有効		
128	Code128/EAN-128 FNC4 拡張 ASCII 変換有効		
CODE128_CHKD	Code128 チェックデジット計算指定	0	無効
		1	有効
CODE128_CHKCH	Code128 チェックキャラクタ出力指定	0	無効 (固定)
[MSI Option]			
MSI_MIN	MSI 読み取り有効最小桁数	1~57	
MSI_MAX	MSI 読み取り有効最大桁数	1~57	
MSI_OUTFORMAT	MSI 出力フォーマット指定	0	出力フォーマットなし
MSI_CHKD	MSI チェックデジット計算指定	0	チェックデジット計算なし

		1	1桁、mod 10
		2	2桁 1st: mod 11/2nd: mod 10
		3	2桁 1st: mod 10/2nd: mod 10
MSI_CHKCH	MSI チェックキャラクタ出力指定	0	無効
		1	有効
[IATA Option]			
IATA_MIN	IATA 読み取り有効最小桁数	1~65	
IATA_MAX	IATA 読み取り有効最大桁数	1~65	
IATA_OUTFORMAT	IATA 出力フォーマット指定	0	無効 (固定)
IATA_CHKD	IATA チェックデジット計算指定	0	無効
		1	末尾以外を計算
		2	クーポン No+データ計算
		3	データ部のみ計算
		4	モジュラス 10(DT-X7 のみ)
IATA_CHKCH	IATA チェックキャラクタ出力指定	0	無効 (固定)
[RSS-14 Option] (DT-5200、DT-X7 のみ)			
RSS14_MIN	RSS-14 読み取り有効最小桁数	14 (固定)	
RSS14_MAX	RSS-14 読み取り有効最大桁数	14 (固定)	
RSS14_OUTFORMAT	RSS-14 出力フォーマット指定	0	標準出力
		1	先頭の“01”出力なし
RSS14_CHKD	RSS-14 チェックデジット計算指定	1	有効 (固定)
RSS14_CHKCH	RSS-14 チェックキャラクタ出力指定	1	有効 (固定)
[RSS Limited Option] (DT-5200、DT-X7 のみ)			
RSSLTD_MIN	RSS Limited 読み取り有効最小桁数	14 (固定)	
RSSLTD_MAX	RSS Limited 読み取り有効最大桁数	14 (固定)	
RSSLTD_OUTFORMAT	RSS Limited 出力フォーマット指定	0	標準出力
		1	先頭の“01”出力なし
RSSLTD_CHKD	RSS Limited チェックデジット計算指定	1	有効 (固定)
RSSLTD_CHKCH	RSS Limited チェックキャラクタ出力指定	1	有効 (固定)
[RSS Expanded Option] (DT-5200、DT-X7 のみ)			
RSSEXP_MIN	RSS Expanded 読み取り有効最小桁数	1~74	
RSSEXP_MAX	RSS Expanded 読み取り有効最大桁数	1~74	
RSSEXP_OUTFORMAT	RSS Expanded 出力フォーマット指定	0	出力フォーマットなし (固定)
RSSEXP_CHKD	RSS Expanded チェックデジット計算指定	1	有効 (固定)
RSSEXP_CHKCH	RSS Expanded チェックキャラクタ出力指定	1	有効 (固定)
[RSS 14 Stacked Option] (DT-5200、DT-X7 のみ)			
RSS14S_MIN	RSS 14 Stacked 読み取り有効最小桁数	14(固定)	
RSS14S_MAX	RSS 14 Stacked 読み取り有効最大桁数	14(固定)	
RSS14S_OUTFORMAT	RSS 14 Stacked 出力フォーマット指定	0	標準出力

T		1	先頭の“01”出力なし
RSS14S_CHKD	RSS 14 Stacked チェックデジット計算指定	1	有効(固定)
RSS14S_CHKCH	RSS 14 Stacked チェックキャラクタ出力指定	1	有効(固定)
[RSS Expanded Stacked Option] (DT-5200、DT-X7 のみ)			
RSSEXP_MIN	RSS Expanded 読み取り有効最小桁数	1~74	
RSSEXP_MAX	RSS Expanded 読み取り有効最大桁数	1~74	
RSSEXP_OUTFORM AT	RSS Expanded 出力フォーマット指定	0	出力フォーマットなし (固定)
RSSEXP_CHKD	RSS Expanded チェックデジット計算指定	1	有効 (固定)
RSSEXP_CHKCH	RSS Expanded チェックキャラクタ出力指定	1	有効 (固定)
[Read Mode Option]			
READMODE	読み取り方式設定を指定	0	単発読み
		1	連続読み
[Buzzer Control Option]			
BUZZER	ブザー設定の指定	0	無効
		1	有効
[Led Control Option]			
LEDCTRL	LED 設定の指定	0	無効
		1	有効
		2	有効、読み取り失敗時は無効
[OutBuff Control Option]			
OUTBUFF	出力方式設定の指定	0	OBR バッファ出力
		2	キー出力
		3	クリップボード出力
		4	キーボードイベント出力
[Suff Char Option]			
ENDCODE	終了コードの指定	0	<CR>
		1	<LF>
		2	<CR> + <LF>
		3	<TAB>
		4	終了コードなし
[Multi Step Reading Option]			
MULTISTEP	段数読みの指定	0	通常読み
		1	段数読み
[Comp Counter Option]			
CMPCNT	照合回数を指定	1~9(レーザスキャナモデル) 0~9(リニアイメージャモデル)	
[Read Counter Option]			
READCNT	連続読み方式の時の、読み取り回数を指定	1~9	
[Scan Time Option]			
SCANTIME	タイムアウト時間(秒)を指定	1~9	

FILTERMODE	ノイズキャンセルフィルタのモードを指定	0	フィルタ無
		1	ソフトフィルタ有
		2	ハードフィルタ有
		3	ソフト/ハードフィルタ有
FILTERCNT	読み取り開始から、ノイズキャンセルフィルタを開始するまでの時間(秒)を指定	1~8	
[Vibrator Control Option]			
VIBRATOR	バイブレータ設定の指定	0	無効
		1	有効
[Scanning Key Option]			
KEY	読み取り開始キーの設定を指定 (複数のキーを同時に読み取り開始キーに設定する場合、これらの値の合計を指定します)	1	トリガキーL
		2	トリガキーR
		4	マルチキー
		8	カーソルキーL
		10	カーソルキーR
		32	カーソルキーUp
		64	カーソルキーDown
		128	ガントリガー(DT-5200のみ)
		256	センタートリガー
[Laser Swing Option]			
SWING	振れ角制御モードの指定	0	制御なし
		1	大
		2	中
		3	小
[Laser Focus Option]			
FOCUS	レーザフォーカス設定の指定	0	無効
		1	有効
[Notification Option]			
NOTIFICATION	読み取り完了の通知方法の指定	1	ウィンドウメッセージによる通知
		2	Eventによる通知
[Decode Level Option]			
LEVEL	デコードレベル設定の指定	0	標準
		1	高度
Gain	レーザゲイン設定の指定	0	ゲインモード0
		1	ゲインモード1
		2	ゲインモード2
		3	ゲインモード3
		128	ゲインオート
[Decode Customise Option] (△)			
BARWIDTH	バーの太り・細りの補正値を指定	0	補正なし
		1	黒を細く補正
		2	黒を細く補正(大)
		3	白を細く補正
		4	白を細く補正(大)
		5	黒を太く補正

		6	黒を太く補正(大)
		7	白を太く補正
		8	白を太く補正(大)
MARGINCHECK	Right/Left マージン閾値を指定	0	マージン閾値大
		1	マージン閾値中
		2	マージン閾値小
		3	マージン閾値最小
[Learning Decode Option] (△)			
LEARINING	デコーダ学習機能設定の指定	0	学習機能無効
		1	学習機能有効

△ パッチ対応 DT5200M50 シリーズの場合には、サービスパック(バージョン 1.02)または「MoDevDT5200.101.CAB+LaserDT5200.101.CAB」をインストールする必要があります。

△2 パッチ対応 DT5200M50 シリーズの場合には、サービスパック(バージョン 1.02)または「MoDevDT5200.103.CAB+LaserDT5200.102.CAB」をインストールする必要があります。

※ RSS コードは 2007 年 2 月に GS1 DataBar に名称変更されています。

RSS-14・・・GS1 DataBar Omnidirectional

RSS Limited・・・GS1 DataBar Limited

RSS Expanded・・・GS1 DataBar Expanded

RSS-14 Stacked・・・GS1 DataBar Stacked

RSS Expanded Stacked・・・GS1 DataBar Expanded Stacked

※ EAN-128 コードは 2006 年 1 月に GS1-128 に名称変更されています。

カシオ計算機お問い合わせ窓口

製品に関する最新情報

- 製品サポートサイト（カシオペア・ハンディターミナル）

<http://casio.jp/support/ht/>

カシオ計算機株式会社

〒151-8543 東京都渋谷区本町 1-6-2

TEL 03-5334-4638(代)