BarcodeStudio DLL x64

外部インターフェース設計仕様書

NECソリューションイノベータ株式会社

目次

[1. 概要 3](#_Toc487708299)

[1.1. モジュール概要 3](#_Toc487708300)

[1.2. モジュール構成 3](#_Toc487708301)

[1.3. 位置付け 3](#_Toc487708302)

[2. 外部インターフェース 4](#_Toc487708303)

[2.1. 処理の流れ 4](#_Toc487708304)

[2.2. 各インターフェース仕様 6](#_Toc487708305)

[2.2.1. 初期化処理 6](#_Toc487708306)

[2.2.2. 終了処理 7](#_Toc487708307)

[2.2.3. バーコードの設定 8](#_Toc487708308)

[2.2.4. 現在のバーコードの設定値の取得 10](#_Toc487708309)

[2.2.5. バーコードのコード設定 11](#_Toc487708310)

[2.2.6. 現在設定されているバーコードのコードの取得 12](#_Toc487708311)

[2.2.7. バーコードの描画 13](#_Toc487708312)

[2.3. データ構造 15](#_Toc487708313)

[2.3.1. BS\_INFO構造体 15](#_Toc487708314)

[3. デフォルト値 20](#_Toc487708315)

[4. 各バーコードのタイプごとのサンプルコード 22](#_Toc487708316)

[5. エラー一覧 23](#_Toc487708317)

[5.1. BS\_SetParam()のエラー 23](#_Toc487708318)

[5.2. BS\_DrawBarcode()、BS\_DrawBarcodeEx()のエラー 24](#_Toc487708319)

[6. 特記事項 25](#_Toc487708320)

# 概要

## モジュール概要

BarcodeStudio DLL x64(以下、BarcodeStudio DLL)は、連携する64ビットアプリケーションから呼び出され、指定されたデバイスコンテキストにバーコードを描画するDLLである。

## モジュール構成

モジュール構成は単一のDLLファイルとし、ファイル名は「BarStdo\_x64.dll」とする。

## 位置付け

本モジュールの位置付けを以下に示す。

連携するAP

BarcodeStudio DLL

DLL Export

によるI/F

# 外部インターフェース

## 処理の流れ

＜イベントトレース図＞

連携AP BarcodeStudio DLL

DLLの

初期化

LoadLibrary()

DLLMain

BS\_Open()

DLLの初期化、及び、使用する内部バッファ領域の確保／保存

戻り値：バーコードの設定のIndex

BS\_SetParam()

バーコードの

設定

バーコードの設定を引数で受け取り、DLL内のバッファの設定を更新

戻り値：成功=0、失敗=エラー値

BS\_SetCode()

戻り値：TRUE / FALSE

コードの設定を引数で受け取り、DLL内のバッファの設定を更新

コードの設定

BS\_GetParam()

戻り値：TRUE / FALSE

DLL内のバッファの設定を、引数で受け取ったバッファにコピー

現在のバーコードの設定値 / 描画領域サイズの取得

BS\_GetCode(ポインタNULL)

現在のコードの取得

(必要な場合のみ)

戻り値：コードを格納するのに必要なバッファのサイズ

コードを格納するバッファのサイズを計算

BS\_GetCode(ポインタあり)

戻り値：コピーしたBYTE数

DLL内で保存してあるコードの文字列を、引数で受け取ったバッファにコピー

BS\_DrawBarcode()

BS\_DrawBarcodeEx()

BS\_Close()

戻り値：TRUE / FALSE

DLLの終了処理、及び、内部バッファ領域開放

DLLの終了処理

バーコードの描画

描画先のDC、位置を引数で受け取り、バーコードを描画

戻り値：TRUE / FALSE

DLLMain

FreeLibrary()

＜補足＞

* BS\_SetParam()を複数回実行すると、エンジン内のバッファ(バーコード設定値)はその都度上書きされる。（BS\_SetCode()も同様）

【例】

BS\_SetParam(①)→BS\_SetParam(②)→BS\_GetParam()の順に行った場合、BS\_GetParam()では②の設定が取得される。

* 1つのデバイスコンテキストに複数のバーコードを描画する場合、描画するバーコードの数だけBS\_Open()を呼ぶ事を推奨する。
* バーコードを描画する領域サイズは、BS\_GetParam()またはBS\_SetParam()で取得するが、バーコードのスタイルと、コードを共にセットした後でなければ正しい値とならない。以下に2通りの取得方法を示す。

BS\_SetParam()→BS\_SetCode()→BS\_GetParam() ：イベントトレース図の方法

BS\_SetCode()→BS\_SetParam()：コードをセットした後ならBS\_SetParam()で取得できる描画領域サイズも有効（推奨）。

## 各インターフェース仕様

### 初期化処理

1. 概略

初期化を行う。DLL内部で使用するバッファ領域の確保／保存を行う。

1. 関数定義

DllExport HANDLE BS\_Open(

void

)

1. 引数説明

引数なし。

1. 戻り値

バーコードの設定のIndex。各インターフェースを呼び出す際は、本Indexをパラメータとして指定する。

1. 注意事項

BarcodeStudio DLLの各インターフェースを使用する場合は、最初に必ず本インターフェースを呼び出す必要がある。

### 終了処理

1. 概略

終了処理。DLL内部で使用したバッファ領域開放などを行う。

1. 関数定義

DllExport BOOL BS\_Close (

HANDLE hBSIndex

)

1. 引数説明

|  |  |  |
| --- | --- | --- |
| No. | 引数名 | 説明 |
| 1 | hBSIndex | BS\_Open()の戻り値のIndexを指定する。 |

1. 戻り値

処理が正常終了した場合はTRUE 、そうでない時はFALSE を返す。

1. 注意事項

BarcodeStudio DLLの各インターフェースを使用した場合は、最後に必ず本インターフェースを呼び出す必要がある。

### バーコードの設定

1. 概略

バーコードの諸設定を引数で受け取り、DLL内で保存しているバッファの設定を更新する。

1. 関数定義

DllExport DWORD BS\_SetParam (

HANDLE hBSIndex,

BS\_INFO\* lpBS\_Info,

SIZE\* lpSize

)

1. 引数説明

|  |  |  |
| --- | --- | --- |
| No. | 引数名 | 説明 |
| 1 | hBSIndex | BS\_Open()の戻り値のIndexを指定する。 |
| 2 | lpBS\_Info | BS\_INFO構造体へのポインタ。バーコードの諸設定が保存されているBS\_INFO構造体のポインタを指定する。  （構造体の内容は「2.3.1 BS\_INFO構造体」を参照） |
| 3 | lpSize | バーコードの描画領域を格納するSIZE構造体へのポインタ。DLLが本パラメータにサイズを代入して返す。  サイズの単位は0.01mm(0.16mmの場合は16という値がSIZE構造体に入る)。 |

1. 戻り値

処理が正常終了した場合は0を返す。DLL内でエラーが発生した場合は、該当するエラー値を返す。

1. 注意事項

lpBS\_Info、lpSizeのバッファ領域は、連携AP側で管理（確保/開放）をする。（BarcodeStudio DLL内で使用するバッファとは異なる）

サイズ自動設定使用時のCODE128、GS1-128（EAN-128)でドット単位の描画を行う場合、指定したDPI値と画面のDPI値が一致しない限り、lpSizeの値と画面上の描画サイズは一致しない。

### 現在のバーコードの設定値の取得

1. 概略

現在のバーコードの設定を、引数で受け取った構造体のポインタに反映する。

1. 関数定義

DllExport BOOL BS\_GetParam(

HANDLE hBSIndex,

BS\_INFO\* lpBS\_Info,

SIZE\* lpSize

)

1. 引数説明

|  |  |  |
| --- | --- | --- |
| No. | 引数名 | 説明 |
| 1 | hBSIndex | BS\_Open()の戻り値のIndexを指定する。 |
| 2 | lpBS\_Info | BS\_INFO構造体へのポインタ。DLLが、本構造体の各設定項目を、現在のバーコードの設定に更新する。  （構造体の内容は「2.3.1 BS\_INFO構造体」を参照） |
| 3 | lpSize | バーコードの描画領域を格納するSIZE構造体へのポインタ。DLLが本パラメータにサイズを代入して返す。  サイズの単位は0.01mm(0.16mmの場合は16という値がSIZE構造体に入る)。 |

1. 戻り値

処理が成功した場合はTRUE、そうでない時はFALSEを返す。

1. 注意事項

lpBS\_Info、lpSizeのバッファ領域は、連携AP側で管理（確保/開放）をする。（BarcodeStudio DLL内で使用するバッファとは異なる）

サイズ自動設定使用時のCODE128、GS1-128（EAN-128)でドット単位の描画を行う場合、指定したDPI値と画面のDPI値が一致しない限り、lpSizeの値と画面上の描画サイズは一致しない。

### バーコードのコード設定

1. 概略

バーコードのコードの文字列のポインタを引数で受け取り、DLL内で保存しているバッファの設定を更新する。

1. 関数定義

DllExport BOOL BS\_SetCode(

HANDLE hBSIndex,

LPSTR lpszCode,

DWORD dwSize

)

1. 引数説明

|  |  |  |
| --- | --- | --- |
| No. | 引数名 | 説明 |
| 1 | hBSIndex | BS\_Open()の戻り値のIndexを指定する。 |
| 2 | lpszCode | バーコードのコードの文字列が保存されているバッファのポインタを指定する。 |
| 3 | dwSize | lpszCodeの示すバッファのサイズを指定する。 |

1. 戻り値

処理が正常終了した場合はTRUE、失敗した場合はFALSEを返す。

1. 注意事項

lpszCodeのバッファ領域は、連携AP側で管理（確保/開放）をする。（BarcodeStudio DLLで使用するバッファとは異なる）

指定できる文字列は最大255文字（終端のNULLを含めた256バイト）までとし、それ以上の文字列バッファを渡した場合は、255文字以降は無視される。

### 現在設定されているバーコードのコードの取得

1. 概略

引数で受け取ったポインタのバッファに、DLL内で保存しているバーコードのコードの文字列をコピーする。

1. 関数定義

DllExport BOOL BS\_GetCode(

HANDLE hBSIndex,

LPSTR lpszCode,

DWORD dwSize

)

1. 引数説明

|  |  |  |
| --- | --- | --- |
| No. | 引数名 | 説明 |
| 1 | hBSIndex | BS\_Open()の戻り値のIndexを指定する。 |
| 2 | lpszCode | バーコードのコードの文字列を格納するバッファのポインタを指定する。NULLが指定された場合は、バッファに必要な領域のサイズを戻り値で返す。 |
| 3 | dwSize | lpszCodeの示すバッファのサイズを指定する。lpszCodeにNULLを指定する場合は、本パラメータに値を設定する必要はない。 |

1. 戻り値

lpszCodeにNULLを指定した場合は、バーコードのコードの文字列を格納するのに必要なバッファのサイズをBYTE単位で返す。

lpszCodeにポインタが設定されている場合は、コピーしたBYTE数を返す。

1. 注意事項

lpszCodeのバッファ領域は、連携AP側で管理（確保/開放）をする。（BarcodeStudio DLL内で使用するバッファとは異なる）

lpszCodeには最大で256バイトの領域が必要となる。

### バーコードの描画

1. 概略

引数で受け取ったデバイスコンテキストに対し、現在のバーコードの設定でバーコードを描画する。

1. 関数定義

DllExport BOOL BS\_DrawBarcode(

HANDLE hBSIndex,

HDC hDC,

POINT\* pPoint,

DWORD\* lpdwBSErr,

DWORD\* lpdwBSLastErr

)

DllExport BOOL BS\_DrawBarcodeEx(

HANDLE hBSIndex,

CDC\* pDC,

POINT\* pPoint,

DWORD\* lpdwBSErr,

DWORD\* lpdwBSLastErr

)

1. 引数説明

|  |  |  |
| --- | --- | --- |
| No. | 引数名 | 説明 |
| 1 | hBSIndex | BS\_Open()の戻り値のIndexを指定する。 |
| 2 | hDC | バーコードを描画するデバイスコンテキストを指定する。 |
| 3 | pDC | バーコードを描画するデバイスコンテキストを持つ、MFCのCDCオブジェクトのポインタを指定する。 |
| 4 | pPoint | POINT構造体へのポインタ。バーコードを描画する領域の、左上の座標を設定する。座標の単位は0.01mm(5mmの場合は500という値をPOINT構造体に設定する)。 |
| 5 | pdwBSErr | DWORD型の数値へのポインタ。DLLでエラーが発生した場合、DLL側が該当するエラー値を設定して返す。 |
| 6 | lpdwBSLastErr | DWORD型の数値へのポインタ。DLLでエラーが発生した場合、DLL側でGetLastError()APIで取得した値を設定して返す。 |

1. 戻り値

処理が正常終了した場合はTRUE 、そうでない時はFALSE を返す。

1. 注意事項

pPoint、lpdwBSErr、lpdwBSLastErrのバッファ領域は、連携AP側で管理（確保/開放）をする。

スケーリングを変更するときは既定のマッピングモード(MM\_TEXTを除く)の値を元に、Viewportを使用すること。SetWindowExt()、ScaleWindowExt()等の設定は反映されない。また、スケーリングの変更により向きを変化させないこと。

BS\_DrawBarcode()とBS\_DrawBarcodeEx()は、ほぼ同じ動作をするが、MFCアプリケーションではBS\_DrawBarcodeEx()を使用することを推奨する。

## データ構造

### BS\_INFO構造体

1. 概略

バーコードの各種設定。

1. 構造定義

＜BS\_INFO＞

|  |  |  |  |
| --- | --- | --- | --- |
| サイズ | 型 | シンボル | 意味 |
| 2 | WORD | wFlag1 | 動作フラグ。各フラグの値と意味は、以下の形式となっている。   |  |  | | --- | --- | | 値 | 意味 | | 0x0001 | HRC（数字/文字）を付加する。 | | 0x0002 | HRC（数字/文字）をBoldで描画する。 | | 0x0004 | HRC（数字/文字）をItalicで描画する。 | | 0x0008 | JAN/ITFでチェックサムの照合を行う。本フラグを立てた場合は、lpszCodeにチェックサムも入力する必要がある。  本フラグを立てない場合は、チェックサムが自動的に付加される。 | | 0x0010 | JANのガードバーを長くする | | 0x0020 | TFのベアラーバーを付加する。 | | 0x0040 | Code39、NW7でスタート/エンドキャラクタを付加する。 | | 0x0080 | GS1-128（EAN-128)でHRC（数字/文字）の識別子に自動的に”()”を付加する。 | | 0x0100 | サイズ自動設定を使用する。  使用する場合は同じ構造体のパラメータwDPIで設定された値で描画する。  サイズ自動設定を使用する場合、wWidthの設定は無視される。 | |
| 2 | WORD | wFlag2 | 動作フラグ。各フラグの値と意味は、以下の形式となっている。   |  |  | | --- | --- | | 値 | 意味 | | 0x0001 | GS1-128のHRCを"標準料金代理収納ガイドライン"に合わせる。 | | 0x0002 | 印刷優先の描画（ドット単位の描画）を行う。（印刷時に使用する）  BS\_INFO.wDPIの項目の解像度で描画されるため、大きなバーコードが表示される。  このフラグを立てない場合は、画面解像度で表示する。  この項目は以下の条件のときに動作する。   * wFlag2の0ｘ0010フラグを立てた場合 * wFlag1 0x0100フラグが立っており、かつwTypeがCode128　or　GS1-128（EAN-128)の場合 | | 0x0004 | GS1-128（EAN-128)で標準料金代理収納のHRC表示のとき、44桁目のチェックデジットを計算する | | 0x0008 | Code128、GS1-128（EAN-128)のバーの幅を補正する。補正はサイズ自動設定が有効なときにのみ行われる。  補正量はiBarCorrectionW、iBarCorrectionBを使用する。ただし補正後のバーの幅は1未満にはならない（1残る）。 | | 0x0010 | 解像度を意識した補正を使用する。 | |
| 2 | WORD | wType | バーコードのタイプ。タイプ毎の値は、以下のようになっている。   |  |  | | --- | --- | | 値 | 意味 | | 0 | JAN標準13桁 | | 1 | JAN短縮8桁 | | 2 | ITF(Interleaved 2of5)標準 | | 3 | ITF(Interleaved 2of5)拡張 | | 4 | ITF(Interleaved 2of5)アドオン | | 5 | Industrial 2of5 | | 6 | NW7 | | 7 | CODE39 | | 8 | CODE128 | | 9 | 郵政コード | | 10 | GS1-128（EAN-128) | |
| 2 | WORD | wHeight | バーコードのバーの高さを指定する。0.01mm単位で指定する。（12.00mmの場合、1200と指定する）  BS\_INFO.wFlag2 0x0010のフラグがたっている場合、入力値はBS\_INFO.wDPIで指定した解像度に補正される。  最小：1.00mm　最大300.00mm |
| 2 | WORD | wWidth | バーコードのバーの幅を指定する。0.01mm単位で指定する。ITFなどでは細バーの幅となる。（1.50mmの場合、150と指定する）  BS\_INFO.wFlag2 0x0010のフラグがたっている場合、入力値はBS\_INFO.wDPIで指定した解像度に  補正される。  最小：0.10mm　最大10.00mm |
| 2 | WORD | wRotation | バーコードを回転して描画する。設定する値は以下のようになっている。   |  |  | | --- | --- | | 値 | 意味 | | 0 | 回転角0度 | | 1 | 回転角90度 | | 2 | 回転角180度 | | 3 | 回転角270度 | |
| 2 | WORD | wQuietZone | 左右のクワイエットゾーンを指定する。0.01mm単位で指定する。（12.00mmの場合、1200と指定する）  0を指定すると、自動的に最適な幅で描画する。  BS\_INFO.wFlag2 0x0010のフラグがたっている場合、入力値はBS\_INFO.wDPIで指定した解像度に補正される。  最小：0.01mm　最大300.00mm |
| 2 | WORD | wRatio | ITF、NW7、CODE39の場合の、細バーと太バーの比率を指定する。0.1単位で指定する。（2.5の場合、25と指定する）  最小：2.0　最大：3.0 |
| 2 | WORD | wStartCode128 | Code128、GS1-128（EAN-128)でスタートコードを指定して描画する。設定値は以下のようになっている。  ただし、自動以外を指定し、且つ入力されたバーコードのコードが表現できない場合はエラーが返る。このとき、コードは自動的に変換されバーコードが表示される。   |  |  | | --- | --- | | 値 | 意味 | | 0 | 自動。スタートコードはBarcodeStudioで自動的に決定される。 | | 1 | CodeSet Aで描画する。 | | 2 | CodeSet Bで描画する。 | | 3 | CodeSet Cで描画する。 | |
| 2 | WORD | wHRCHeight | HRC（数字/文字）の高さをポイント数で指定する。  0を指定すると、自動的に最適な高さで描画する。  最小：1Point　最大50Point |
| 2 | WORD | wHRCWidth | HRC（数字/文字）の幅をポイント数で指定する。  0を指定すると、自動的に最適な幅で描画する。  最小：1Point　最大50Point |
| 256 | Char[256] | szHRCFontName | HRC（数字/文字）の書体名を指定する。（例：”ＭＳゴシック”等） |
| 4 | DWORD | dwBarColor | バーコードのバーの色を以下のように指定する。  0x00BBGGRR  RR：赤色成分を0x00～0xFFの範囲で指定  GG：緑色成分を0x00～0xFFの範囲で指定  BB：青色成分を0x00～0xFFの範囲で指定 |
| 4 | DWORD | dwBackColor | バーコードの背景色を指定する。  指定方法はdwBarColorと同様。 |
| 4 | DWORD | dwHRCColor | バーコードのHRCの色を指定する。  指定方法はdwBarColorと同様。 |
| 2 | WORD | wDPI | プリンタのDPIを指定する。  wWidth、wHeightの補正に使用する。   |  |  | | --- | --- | | 値 | 意味 | | 0 | 96 dpi | | 1 | 160 dpi | | 2 | 180 dpi | | 3 | 200 dpi | | 4 | 240 dpi | | 5 | 300 dpi | | 6 | 360 dpi | | 7 | 400 dpi | | 8 | 600 dpi | | 9 | 720 dpi | | 10 | 800 dpi | | 11 | 1200 dpi(8dot) | | 12 | 1440 dpi | | 13 | 1600 dpi | | 14 | 1200dpi(9dot) | |
| 1 | char | cESCChar | Code128、GS1-128（EAN-128)の制御コード入力に使用する文字を指定する。 |
| 2 | WORD | wHRCMargin | バーコードとHRCの間のマージンを指定 |
| 4 | int | iBarCorrectionW | 白バーの補正値。Code128、GS1-128（EAN-128)のときのみ有効。  最小：-2 　最大：2 |
| 4 | int | iBarCorrectionB | 黒バーの補正値。Code128、GS1-128（EAN-128)のときのみ有効。  最小：-2 　最大：2 |

# デフォルト値

BarcodeStudio DLLのインターフェース関数BS\_Open()が呼ばれたときのデフォルト値は以下のようになっている。

＜BS\_INFO＞

|  |  |
| --- | --- |
| シンボル | 値 |
| wFlag1 | 以下の値が設定されている。   |  |  | | --- | --- | | 値 | 意味 | | 0x0001 | HRC（数字/文字）を付加する。 | | 0x0010 | JANのガードバーを長くする | | 0x0020 | ITFのベアラーバーを付加する。 | | 0x0040 | Code39、NW7でスタート/エンドキャラクタを付加する。 | | 0x0080 | GS1-128（EAN-128)でHRC（数字/文字）の識別子に自動的に”()”を付加する。 | | 0x0100 | サイズ自動設定を使用する。 | |
| wFlag2 | 以下の値が設定されている。   |  |  | | --- | --- | | 値 | 意味 | | 0x0001 | GS1-128（EAN-128)のHRCを"標準料金代理収納ガイドライン"に合わせる。 | | 0x0002 | 印刷優先の描画を行う。 | | 0x0004 | GS1-128（EAN-128)で標準料金代理収納のHRC表示のとき、44桁目のチェックデジットを計算する。 | |
| wType | 以下の値が設定されている。   |  |  | | --- | --- | | 値 | 意味 | | 0 | JAN標準13桁 | |
| lpszCode | “”(文字列が設定されていない場合、各バーコードのサンプルコードが表示される。) |
| wHeight | 0 |
| wWidth | 0 |
| wRotation | 以下の値が設定されている。   |  |  | | --- | --- | | 値 | 意味 | | 0 | 回転角0度 | |
| wQuietZone | 0 |
| wRatio | 25 |
| wStartCode128 | 以下の値が設定されている。   |  |  | | --- | --- | | 値 | 意味 | | 0 | 自動 | |
| wHRCHeight | 0 |
| wHRCWidth | 0 |
| lpszHRCFontName | ”ＭＳゴシック” |
| dwBarColor | 0x00000000 |
| dwBackColor | 0x00FFFFFF |
| dwHRCColor | 0x00000000 |
| wDPI | 以下の値が設定されている。   |  |  | | --- | --- | | 値 | 意味 | | 8 | 600dpi | |
| cESCChar | ‘@’ |
| wHRCMargin | 0 |
| iBarCorrectionW | 0 |
| iBarCorrectionB | 0 |

# 各バーコードのタイプごとのサンプルコード

|  |  |  |
| --- | --- | --- |
| No. | バーコードタイプ | サンプルコード |
| 1 | JAN標準 JIS X 0501 | 4912345678904 |
| 2 | JAN短縮 JIS X 0501 | 49123494 |
| 3 | ITF標準 JIS X 0502 | 14901234567891 |
| 4 | ITF拡張 JIS X 0502 | 0104901234567893 |
| 5 | ITFアドオン JIS X 0502 | 042178 |
| 6 | Ind2of5 | 1234567890 |
| 7 | NW-7 | 1234567890 |
| 8 | CODE39 | CODE-39 12340 |
| 9 | CODE128 | CODE-128 12340 |
| 10 | 郵便 某公文書 | 10000131-3-2-503  (東京都千代田区霞が関1丁目3番2号 郵政省本庁503号室) |
| 11 | GS1-128（EAN-128) | 91912345012345678901234567890101033101230003 |

# エラー一覧

各インターフェースで返されるエラーの一覧を示す。

## BS\_SetParam()のエラー

|  |  |  |  |
| --- | --- | --- | --- |
| No. | 戻り値として返されるエラーフラグ | 値 | 意味 |
| 1 | BS\_ERR\_BSINFO\_NULL | 0x0001 | 引数BS\_INFO\* がNULL。 |
| 2 | BS\_ERR\_TYPE | 0x0002 | 範囲外：BS\_INFO.wType |
| 3 | BS\_ERR\_HEIGHT | 0x0004 | 範囲外：BS\_INFO.wHeight |
| 4 | BS\_ERR\_WIDTH | 0x0008 | 範囲外：BS\_INFO.wWidth |
| 5 | BS\_ERR\_ROTATION | 0x0010 | 範囲外：BS\_INFO.wRotation |
| 6 | BS\_ERR\_QUIETZONE | 0x0020 | 範囲外：BS\_INFO.wQuietZode |
| 7 | BS\_ERR\_RATIO | 0x0040 | 範囲外：BS\_INFO.wRatio |
| 8 | BS\_ERR\_STARTCODE128 | 0x0080 | BS\_INFO.wStartCode128 |
| 9 | BS\_ERR\_HRCHEIGHT | 0x0100 | 範囲外：BS\_INFO.wHRCHeight |
| 10 | BS\_ERR\_HRCWIDTH | 0x0200 | 範囲外：BS\_INFO.wHRCWidth |
| 11 | BS\_ERR\_LPSIZE\_NULL | 0x0400 | 引数lpSizeがNULL |
| 12 | BS\_ERR\_DPI | 0x0800 | 範囲外：BS\_INFO.wDPI |
| 13 | BS\_ERR\_ESCCHAR | 0x1000 | 範囲外：BS\_INFO.iBarCorrectionB |
| 14 | BS\_ERR\_HRCMARGIN | 0x2000 | 範囲外：BS\_INFO.wHRCMargin |
| 15 | BS\_ERR\_INDEX | 0x8000 | Indexが不正。 |
| 16 | BS\_ERR\_BARCORRECTIONW | 0x00010000 | 範囲外：BS\_INFO.iBarCorrectionW |
| 17 | BS\_ERR\_BARCORRECTIONB | 0x00020000 | 範囲外：BS\_INFO.iBarCorrectionB |

## BS\_DrawBarcode()、BS\_DrawBarcodeEx()のエラー

|  |  |  |  |
| --- | --- | --- | --- |
| No. | LpdwBSErrに返されるエラーフラグ | 値 | 意味 |
| 1 | BS\_ERR\_CODECHAR | 0x0001 | コードに使用できない文字が含まれている。 |
| 2 | BS\_ERR\_CODELENGTH | 0x0002 | コードの長さが不正。 |
| 3 | BS\_ERR\_EAN128AI | 0x0004 | GS1-128（EAN-128)でHRC（数字/文字）の識別子の定義とコードが一致しない |
| 4 | BS\_ERR\_CHECKSUM | 0x0008 | Check Sum エラー |
| 5 | BS\_WAR\_CODECHANGED | 0x0100 | 指定されたスタートコードで表示できない。  文字があったため、コード変更を行った。 |
| 6 | BS\_ERR\_INDEX | 0x8000 | Indexが不正。 |

# 特記事項

* MFCアプリケーションでの使用時の注意（DrawBarcodeEx関数を使用する場合）

DLLファイルはMFCアプリケーションでのMFCのリンク方法に合わせ、以下の2種類のDLLがある。

1. 「共有DLLでMFCを使用」するもの
2. 「MFCのスタティックライブラリを使用」するもの

DLLを使用するプロジェクトの設定に合わせて、DLLを選択する必要がある。

また、デバッグ環境においては「共有DLLでMFCを使用」のものを使用すること。

* Code128、GS1-128（EAN-128)においてFNC1/FNC2/FNC3/FNC4/SHIFT/DEL/CODEA/CODEB/CODECを入力する場合、BS\_INFO.cESCCharで指定した記号を使用する。(※1 ※2)

以下にデフォルト値”@”を使用した場合の例を示す。

|  |  |  |
| --- | --- | --- |
| FNC1 | @1 |  |
| FNC2 | @2 |  |
| FNC3 | @3 |  |
| FNC4 | @4 |  |
| SHIFT | @S |  |
| DEL | @D |  |
| CODEA | @A |  |
| CODEB | @B |  |
| CODEC | @C |  |
| @ | @@ | BS\_INFO.cESCCharに指定した文字の入力 |

※1 制御コードの入力にはヘッダーファイルに定義されている数値を使用することも

可能。

※2 BS\_INFO.cESCCharには英数字を除く記号のみ指定可能。

以下はGS1-128のHRCに”(“,”)”を追加するときに使用する。コードには変換されず、HRCに表示される。

TypeがGS1-128で、GS1-128の識別子に”()”を付加するフラグが立っていないときのみ動作する。フラグがたっていない場合、標準料金代理収納のフラグが立っている場合、CODE128の場合は無視される。

またBS\_INFO.cESCCharに”(“,”)”を指定した場合は、出力するコードとして”(“, “)” 自体が入力できなくなる。コードに”(“,“)”を使用する場合はBS\_INFO.cESCCharに他の記号を設定すること。

( @(

) @)

また、コードセットAのNUL、SOHなどの制御コードの入力は、CODEA、SHIFTなどを使用してコードセットAを指定し、同じ数値に変換されるコードセットBの文字を入力することで行う。キャラクタと数値の関係についてはJIS X 0504の表1参照。

以下の例では”m“が”CR”に変換される。

|  |  |
| --- | --- |
| BS\_SetCode()で指定した文字列 | 作成されたバーコードの読み取り結果 |
| (CODEA)123m456 | 123(CR)456 |
| (CODEB)abc(SHIFT)mxyz | abc(CR)xyz |

* 解像度による入力値の補正、及び印刷時の設定について

BS\_INFO.wFlag2に0x0010のフラグがたっている場合、

BS\_INFO.wWidth、BS\_INFO.wHeightの入力値はプリンタ解像度（BS\_INFO.wDPI）によって自動的に補正され、補正された値によって描画される。

補正は以下の式によって行われる。

[入力された値]　/ ( 25.4 / [解像度] ) ※小数を四捨五入する

【例】BS\_INFO.wWidth = 25(0.25mm)、プリンタ解像度600 の場合

0.25 / (25.4 / 600) = 5.905511811023622047244094488189… ≒ 6

以上から、入力値25は600dpiのプリンタでは6ドットとして描画される。

印刷時にはBS\_INFO.wFlag2の2bit目のフラグ(0x0002)を立てる必要がある。このフラグを立てることで、プリンタの解像度に合わせて印刷される。

フラグを立てない場合は、PCでの画面表示用に使用する。この設定で印刷した場合、プリンタの解像度に合わせた描画はされないため、印刷精度は低下する。

* 64ビットアプリケーションから32ビットのBarcodeStudio DLLを呼び出した場合、アプリケーションエラーとなる。アプリケーションとDLLのアーキテクチャーを合わせる必要がある。