

バーコードTrueType
フォントExpert2

JANコード作成モジュール

取扱説明書

新潟キヤノテック株式会社

取扱説明書目次

はじめに	2
お願いとお断り	2
商標について	2
モジュール概要	3
対応 OS	3
モジュール形式	3
モジュール名称	3
対応フォント名	3
API リファレンス	4
関数リファレンス	4
(1) CN.MakeJanCode	4
(2) パラメータ	4
(3) 戻り値	5
(4) 備考	5
使用例	6
Cソースでの使用例	6
Visual Basic®での使用例	7

■はじめに

このたびは、「バーコード TrueType フォント Expert2」をお買い求め頂き、誠にありがとうございます。

「JANコード作成モジュール」は、「バーコード TrueType フォント Expert2」に対応した JAN コードシンボルを作成するための DLL であり、ソフトウェア開発者向けのモジュールです。

ソフトウェアに本 DLL を組み込むことにより、「バーコード TrueType フォント Expert2」に対応した JAN コードシンボルの作成が可能となります。

なお、JAN コードシンボルを表示するためには、「バーコード TrueType フォント Expert2」に同梱されている TrueType フォント「CN-JANCODE」が必要です。

■お願いとお断り

1. 本書の内容の一部または全部を無断転載することは、禁止されています。
2. 本書の内容は、将来予告なしに変更する場合があります。
3. 本書の内容は、万全を期して制作しております。万一、ご不審な点や誤り、あるいは記載もれなど、お気付きのことがありましたらご連絡ください。
4. 運用した結果の影響につきましては、上記 3 項にかかわらず、一切の責任を負いかねます。

■商標について

- Microsoft、Windows、Windows NT、Visual Basic 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。
- その他、本書に記載の製品名、サービス名などは、一般に各社の登録商標または商標です。

■モジュール概要

○対応 OS

Microsoft® Windows® 95 日本語版
Microsoft Windows 98/98 Second Edition 日本語版
Microsoft Windows Millennium Edition 日本語版
Microsoft Windows NT® 4.0 Workstation/Server 日本語版
Microsoft Windows 2000 Professional/Server 日本語版
Microsoft Windows XP Professional/Home Edition 日本語版

○モジュール形式

32ビットDLL (Dynamic Link Library)

○モジュール名称

CN_JanCv.dll

○対応フォント名

バーコード TrueType フォント Expert2
[CN-JANCODE]

■ API リファレンス

○関数リファレンス

(1) CN_MakeJanCode

```
int CN_MakeJanCode (
    const char    *lpText,        // Input
    int           nCD,            // Input
    int           nKeta,          // Input
    int           nSuji,          // Input
    char          *lpJanCode      // Output
);
```

入力されたJANコード(JAN標準13桁またはJAN短縮8桁)から「バーコードTrueTypeフォントExpert2」(CN-JANCODE)に対応したJANコードシンボルを作成します。

(2) パラメータ

lpText 入力文字列

入力のJANコード文字列へのポインタを指定します。

nCD チェックデジットの自動付加の有無

入力のJANコード文字列にチェックデジットが付加されているかどうかを指定します。

0: チェックデジットあり

1: チェックデジットなし

チェックデジットがないので、入力文字列は12桁あるいは7桁となります。

この時、本関数内で自動的にチェックデジットを計算し付加します。

nKeta JANコード桁数

入力のJANコード文字列の桁数を指定します。

13: JAN標準13桁

8: JAN短縮8桁

nSuji 数字表記の有無

作成するJANコードシンボルの数字表記の有無を指定します。

0: 数字表記なし

1: 数字表記あり

lpJanCode 変換後の JAN コードシンボル

変換後の JAN コードシンボルの格納先のポインタを指定します。

JAN 標準 13 桁の場合は 14 バイト以上の領域、JAN 短縮 8 桁の場合は 9 バイト以上の領域が必要です。

戻り値が 0 及び 1 の場合に変換後の文字列がセットされますが、それ以外の場合はセットされません。

(3) 戻り値

変換時のエラーコードを返します。

- 0 : 正常終了。
- 1 : 入力文字列の桁数が不正。
- 1 : パラメータが不正であり、変換不可。
 - ・ lpText または lpJanCode が NULL の場合。
 - ・ nCD、nKeta、nSuji の値が定義外の場合。
- 2 : 入力文字列に不正な文字が含まれており、変換不可。

(4) 備考

戻り値が 1 の場合、以下の規則に従って入力文字列を変換し、返します。

- 入力文字列が変換対象桁数を超える場合は、先頭から対象桁数までを有効とします。
- 入力文字列が変換桁数より少ない場合は、先頭に不足桁数分の "0" を追加して対象桁数の文字列を作成します。

入力文字列の変換対象桁数は以下のように決定されます。

- nCD が「0」で、nKeta が「13」の場合、対象桁数は 13 桁。
- nCD が「0」で、nKeta が「8」の場合、対象桁数は 8 桁。
- nCD が「1」で、nKeta が「13」の場合、対象桁数は 12 桁。
- nCD が「1」で、nKeta が「8」の場合、対象桁数は 7 桁。

■使用例

○ C ソースでの使用例

```
typedef int ( WINAPI *PROC_MAKEJANCODE )( const char *, int, int, int, char * );

PROC_MAKEJANCODE   MakeJanCodeProc;
HINSTANCE           hLib;
char                JanCode[14];
int                 result;

hLib = LoadLibrary( "CN_JanCv.dll" );

if ( hLib ) {
    MakeJanCodeProc = (PROC_MAKEJANCODE)GetProcAddress( hLib, "CN_MakeJanCode" );

    if( MakeJanCodeProc ) {

        /* JAN コード作成関数コール */
        result = MakeJanCodeProc( "490120492525", 1, 13, 0, JanCode);

        if ( result != 0 ) {
            /* JAN コード作成エラー */
        }
    }

    FreeLibrary( hLib );
}
}
```

○ Visual Basic® での使用例

```
Public Declare Function CN_MakeJanCode Lib "CN_JanCv" _
    ( ByVal lpText As String, _
      ByVal nCD As Long, _
      ByVal nKeta As Long, _
      ByVal nSuji As Long, _
      ByVal lpJanCode As String _
    ) As Long

'Private Sub Command1_Click()

    Dim keta As Long
    Dim calc As Long
    Dim suji As Long
    Dim ret As Long

    Dim Text As String * 128
    Dim Jan As String * 128

    Text = "490123456789"
    calc = 1
    keta = 13
    suji = 1

    ret = CN_MakeJanCode(Text, calc, keta, suji, Jan)
    If (ret < 0) Then
        'error
    End If

End Sub
```