

## 2 携帯型

### 2-1 仕様

OS	WinCE 6.0
メインメモリー	256MB
フラッシュメモリー	4GB
SD カードスロット	32GB まで対応
マイク	内蔵/マイクロフォンジャック
バーコードスキャナ	5000LED Aimer/Bar CODE(UPC)
SP CODE スキャナ	1.3M Pixel /SP 2D code
USB	クライアント 1 (USB 2.0) HOST 1 (1.0)
バッテリー	1750mA (フル充電 8 時間使用)
充電	AC アダプター/USB 充電
サイズ	115(W)×63(D)×27(H) ※スキャナ部 40(H)
重さ	170g

音声コード以外に、バーコードを読ませる機能を考えており、バーコードの情報を音声化させる。

バーコードを音声コード化する事で、より音声コードが身近になり、広がり拍車をかけると考えられる。

#### その他の拡張機能

- ・デイジー機能 (デジタル録音図書)
- ・音声メモ
- ・MP3 プレイヤー
- ・色認識 (色を音声で案内)



## 2-2 携帯型機器：今後の展開

---

### 携帯型の商品化

音声コード以外にバーコードを読み取りその商品の名称やその商品に関する内容を音声にする事を検討。

1. 本機器でバーコードをスキャンし商品コードを取得する。
2. 取得した商品コードを機器内のプログラムで機器内にある商品データベース\*から商品名や付属情報を拾う
3. 本機器で音声出力し案内を行う。

\*商品データベースは JICFS/IFDB (Jan Item Code File Service/Integrated Flexible Data Base) を利用する。この JICFS/IFDB (JAN コード統合商品情報データベース)は財団法人流通システム開発センターが管理、サービスを行っており、JAN コードとこれに付随する商品情報を一元的に管理するデータベースサービスである。

JICFS/IFDB は、小売業における POS システムや EOS などの導入運用に必要な商品マスタや、小売業と卸売業間のオンライン受発注処理で利用される商品情報など、流通情報化において作成負荷が大きい商品マスタ情報を収集整備し、誰もが低コストで迅速に正確な商品情報を得られる事を目的とされたものである。

この JICFS/IFDB を利用しシステム化するには、流通システム開発センターの協力が必要になる。今後は、こことの調整も必要となる。

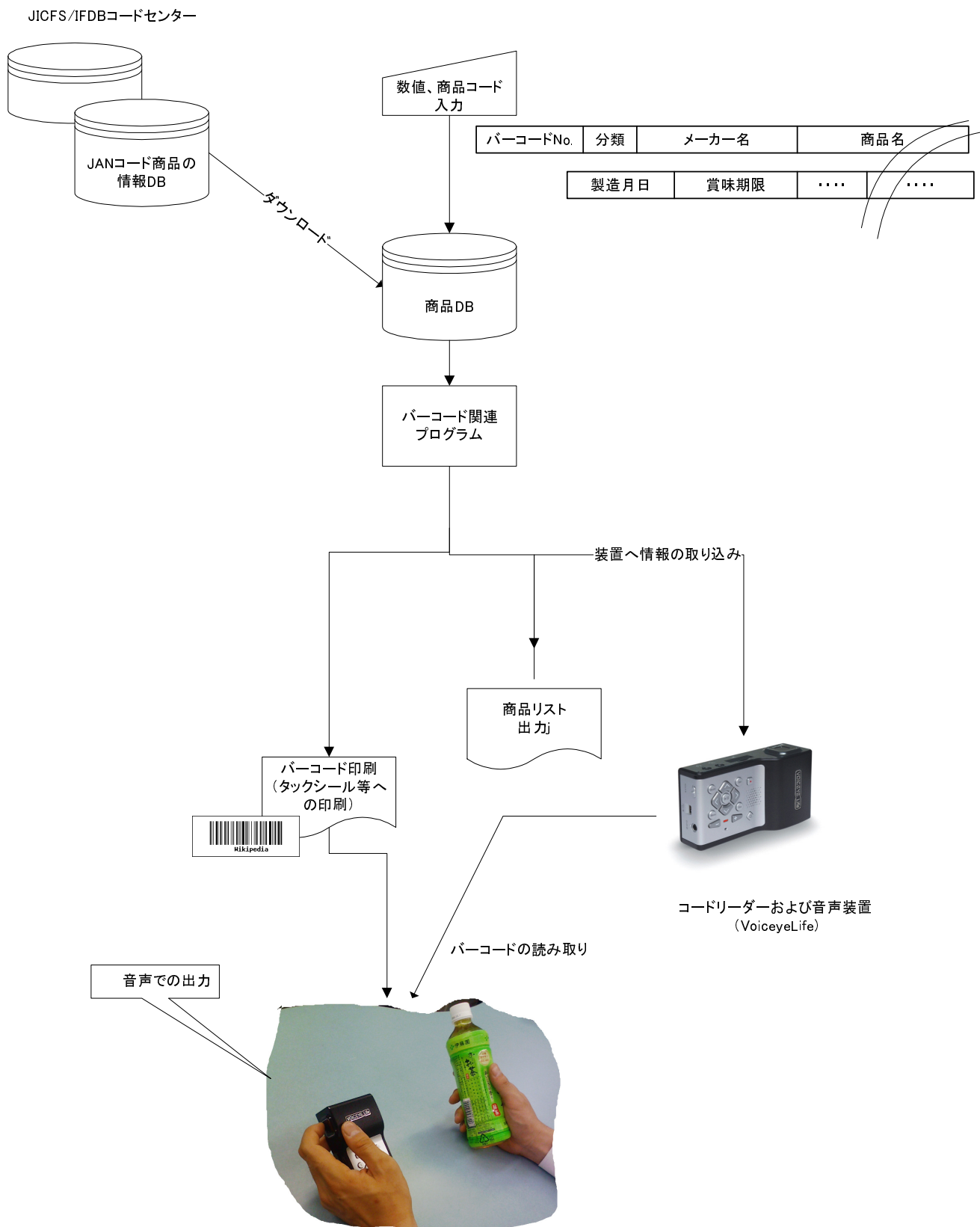
また、DB の組み込みや機器への情報組み込みや、個人個人が生活で利用する為の独自のコード作成・出力ツールの作成が必要となる。

問題点として「バーコードの位置が視覚障害者はわかるか？」という点がある。

当然、位置を認識できない。但し、缶飲料等は側面に縦向きに印刷されているので缶の側面にバーコードリーダーを向け、缶を回転させれば、読み取りが可能となる。

他製品については、今後、バーコード位置の統一化を図る様、働きかける必要があるが、ゆっくりとこの機器を商品に向け動かして行けば、読み取る事ができる。

# 1次元バーコード（商品コード）生成の流れ



## 1 SP コードの形状

### 1-1 各部の名称

SP コードの形状は、以下に示す図のようになっている。また、以下に SP コードを構成する各部の名称を示す。

- ① セル  
二次元コードを形成する最小のドットを「セル」と呼ぶ。
- ② ユニット  
セルの 11×11 単位を「ユニット」と呼ぶ。
- ③ データマトリクス  
SP コード内部のデータ格納領域を「データマトリクス」と呼ぶ。
- ④ 見出し線  
SP コードの四方を取り囲む凸型の部分を、「見出し線 (アラインメント)」と呼ぶ。見出し線の太さの幅は 1 セルで構成されている。

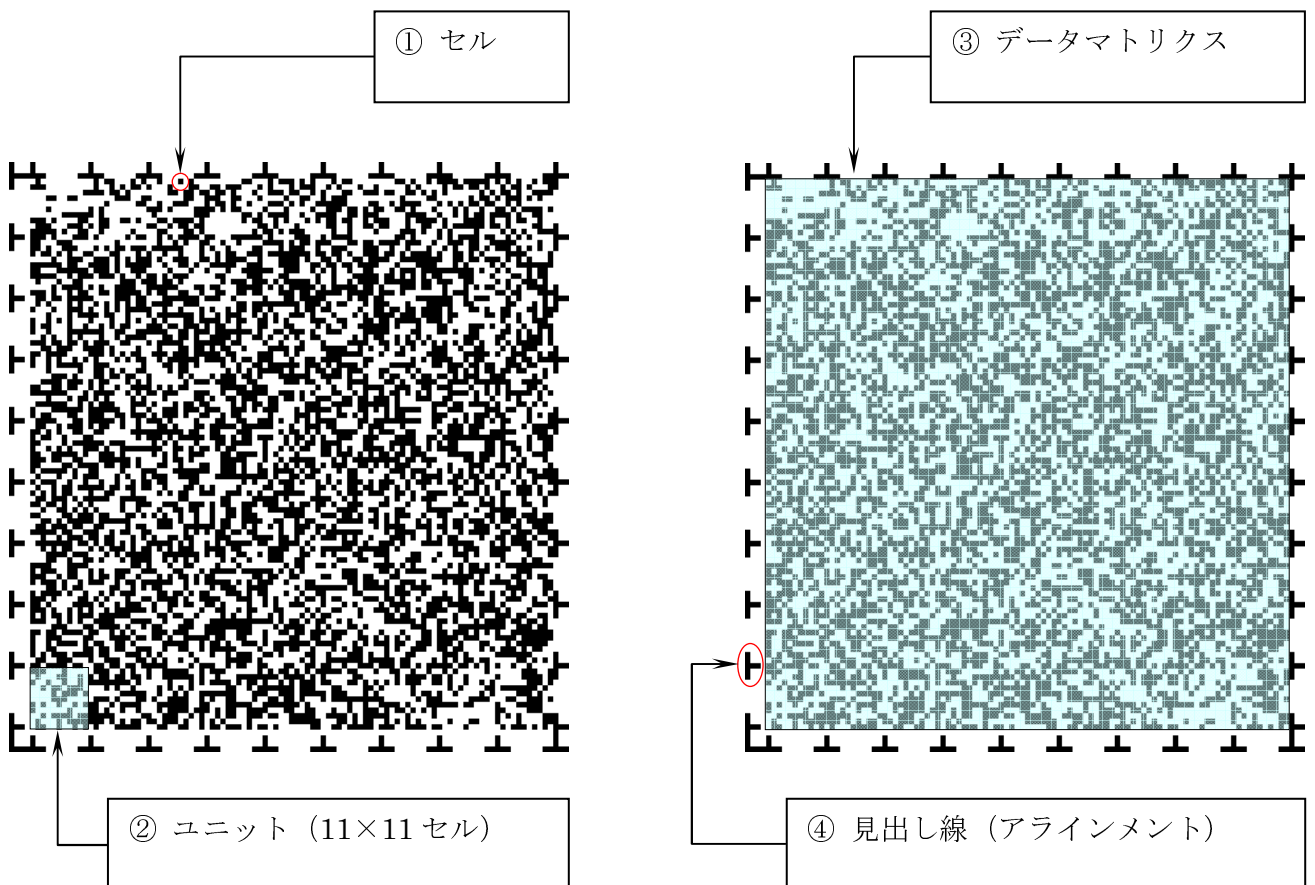


図2 SP コードの各部の名称

左図と右図は同じ M サイズの SP コード (M サイズについては「1-2 サイズとデータ容量」参照のこと)。図は説明のために拡大して表示。

## 1-2 サイズとデータ容量

SP コードには、XS・S・M・L の 4 段階のサイズがあり、それぞれの大きさとデータ容量は下表のようになっている。

表 1 SP コードのサイズとデータ容量

サイズ	セル数	ユニット数	印刷時の大きさ (mm)	誤り訂正	データ容量 (全角文字数)
XS	40×40	3×3	6.8×6.8	強	41
				中	48
				弱	51
S	73×73	6×6	12.4×12.4	強	250
				中	298
				弱	329
M	106×106	9×9	17.9×17.9	強	651
				中	768
				弱	840
L	117×117	10×10	19.8×19.8	強	793
				中	921
				弱	1027

### セル数

見出し線を含めたセル数。

### 印刷時の大きさ

SpeechioSymbol 関数（「2-3 シンボル作成関数：SpeechioSymbol」参照）にて作成された BMP (Bitmap) 画像を、600dpi にて印刷出力したときの大きさ。

### 誤り訂正

誤り訂正には、強・中・弱の 3 種類があり、SP コード作成時の SpeechioSymbol 関数で指定できる（「2-2 エンコード関数：SpeechioEncode」参照）。

### データ容量

目安としての数字であり、テキストデータの内容によって圧縮率が異なるため、文字数は一定ではない。

## 2 ライブラリ (Speechio.dll)

SP コードは、専用のライブラリ「Speechio.dll」を使用して、Microsoft Windows 環境 (Windows XP 日本語版 以降) で容易に作成することができる。本章では、API 関数の仕様説明のみを行い、dll 内の関数の呼び出し方法など、Windows プログラミングに関する説明は、使用例、およびサンプルプログラムにとどめるものとする。

SP コードの作成は、SpeechioEncode 関数によるテキストデータのエンコードと、SpeechioSymbol 関数によるエンコードデータのシンボル化の順で行う。SpeechioEncode 関数では入力テキストデータがエンコードされ、データが bit\_string へ格納される。ここで、bit\_string は"0"と"1"からなる文字配列であり、SP コードの左上から右下へのセルをあらわす中間データである。SpeechioSymbol 関数では、この bit\_string と SP コードの縦横のセル数から、SP コードの画像データが作成され、作成された画像データはクリップボードへ転送される。

### 2-1 入力仕様

入力文字コードは、シフト JIS コードとし、制御コード (0x01~0x1f, 0x7f) は基本的に含めないものとする (「4 入力テキストの整形処理」参照)。シフト JIS コードには、半角 ASCII 文字 (7bit Ascii, 0x20~0x7e)、半角カナ文字 (JIS X 0201 カタカナ, 0xa1~0xdf) を含む。

### 2-2 エンコード関数 : SpeechioEncode

```
short __stdcall SpeechioEncode(  
    char data_type[],  
    char cell_type[],  
    char recover_level[],  
    char copyright[],  
    short data_size,  
    unsigned char data_code[],  
    char path_name[],  
    unsigned char bit_string[]  
);
```

機能

入力テキストデータをエンコードする。

引数

data_type	変換対象データの種類 ="T" (日本語テキスト)
cell_type	作成する SP コードのサイズ ="s" (XS) /="S" (S) /="m" (M) /="M" (L)
recover_level	誤り訂正の強さ ="S" (強) /="N" (中) /="P" (弱)
copyright	未使用
data_size	変換対象データのバイト数
data_code	変換対象データ
path_name	未使用
bit_string	エンコード後のデータを返すバッファ

戻り値

>0            正常終了 : bit\_string へエンコードデータが格納される  
 =0            エラー : エンコードに失敗  
 <0            データ容量オーバー : = (オーバーしたバイト数) × -1

補足

- ・エンコードデータが格納される bit\_string は、呼び出し側であらかじめメモリ領域を確保しておく必要がある。
- ・bit\_string は、SP コードの左上から右下方向へのセルを表す中間データで、"0" (0x30) と "1" (0x31) からなる配列として生成される。"0"が SP コードの白のセルを、"1"が黒のセルを表す。

## 2-3 シンボル作成関数 : SpeechioSymbol

---

```
short __stdcall SpeechioSymbol(
    short col,
    short row,
    unsigned char symbol_data[]
);
```

機能

エンコードデータからの SP コード画像を作成 (シンボル化) する。

引数

col            シンボルの横方向のセル数  
               =40 (XS) /=73 (S) /=106 (M) /=117 (L)  
 row           シンボルの縦方向のセル数  
               =40 (XS) /=73 (S) /=106 (M) /=117 (L)  
 symbol\_data   SpeechioEncode 関数で作成した bit\_string を指定

戻り値

=0            正常終了  
 <0            エラー

補足

- ・成功時は SP コードの画像データが、BMP 形式でクリップボードへ転送される。
- ・作成される画像は、SP コードの 1 セルが 4×4 ドットで構成される。

## 2-4 誤り訂正

誤り訂正は、コードの読み取りに問題がなければ、標準の「中」を推奨する。コード印刷面が汚れる環境などで使用される場合は、「強」とする必要がある。ただし、この場合は搭載できるデータ容量が小さくなるので注意する必要がある（1-2 サイズとデータ容量：表 1 参照）。

## 2-5 関数の使用例

エンコード関数・シンボル作成関数について、簡単な使用例を以下に示す。

```
/*
 * エンコード関数・シンボル作成関数の使用例
 * "Speechio.lib"ファイルへリンクする。"Speechio.dll"はパスの通った
 * フォルダ（システムフォルダかプログラムのあるフォルダ）へ配置する。
 */
#include <stdio.h>
#include <string.h>
#include "Speechio.h" /* 関数プロトタイプ宣言、SP_MAX_BIT_SIZE を定義したヘッダ */

int main(void) {
    unsigned char text[] = "SP コードの作成例です。"; /* 入力テキストデータ */
    unsigned char bit_str[SP_MAX_BIT_SIZE]; /* bit_string バッファ */
    short rc; /* 関数の戻り値用変数 */

    /* エンコード処理 */
    rc = SpeechioEncode("T", "m", "N", "", (short)strlen(text), text, "", bit_str);
    if(rc <= 0) {
        if(rc == 0)
            printf("エンコード処理に失敗しました\n");
        else if(rc < 0)
            printf("%hd バイトオーバーしました\n", -rc);
        return 1;
    }

    /* SP コード画像作成処理 */
    rc = SpeechioSymbol(106, 106, bit_str);
    if(rc < 0) {
        printf("画像作成処理に失敗しました\n");
        return 1;
    }

    printf("SP コード画像データがクリップボードに転送されました\n");

    return 0;
}
```



### 3 SP コード画像の印刷

音声コード読取機器では、台座上に固定された CCD カメラで画像を取り込むため、PC 上で作成した SP コードは、紙面上で適切な大きさと位置となるよう印刷する必要がある。

#### 3-1 画像の大きさ

「SpeechioSymbol 関数」にて作成される BMP 画像は、SP コードの 1 セルが 4×4 ドットで構成されている。音声コード読取機器は、この画像を 600dpi の解像度で印刷した紙面上の SP コードを、精度良く読み取れるよう設計されている（600dpi 以上のレーザープリンタで出力したものと想定）。この場合、紙面上に印刷される SP コードの大きさは、下表のように算出される。

表 2 SP コードの印刷時の大きさ

サイズ	セル数	600dpi 換算での大きさ (mm)
XS	40× 40	$40 \times 4 \times 25.4 \div 600 = 6.77$
S	73× 73	$73 \times 4 \times 25.4 \div 600 = 12.36$
M	106× 106	$106 \times 4 \times 25.4 \div 600 = 17.95$
L	117× 117	$117 \times 4 \times 25.4 \div 600 = 19.81$

「SpeechioSymbol 関数」では画像データをクリップボードへ転送するが、これを避けたい場合には、エンコードデータ (bit\_string) に対して「SpeechioSymbol 関数」と同等の処理を独自で行う必要がある。解像度を大きくした場合などでも、印刷出力時の SP コードの大きさを、表 2 の値と等しくしないと、音声コード読取機器での読み取り精度に問題が生じる可能性があるため注意が必要である。

#### 3-2 画像の位置と余白

音声コード読取機器では、SP コードの印刷された紙の隅を差し込んで読み取らせるため、適切な位置にコードが印刷されるよう SP コード画像を配置する必要がある。

基本的に SP コードは印刷物の右下に配置するものとし、コードのサイズによらず、コードの中心位置が印刷物の右端から 25mm、下端から 25mm となるように配置する。また、他の文字や図形が SP コード画像と重ならないよう注意が必要である。音声コード読取機器で SP コード画像を識別するためには、コードの周囲に最低 4mm の余白が必要である（図 3 参照）。

印刷物のレイアウトや形状から、右下に配置できない場合は、SP コードを印刷物の左下や右上に配置しても問題はない。この場合、音声コード読取機器では SP コード画像の方向を自動的に識別するため、回転に関しては考慮する必要はないが、印刷物の隅を差し込んで音声コード読取機器で読み取り可能な位置に、コードを配置するように注意が必要である（図 4 参照）。

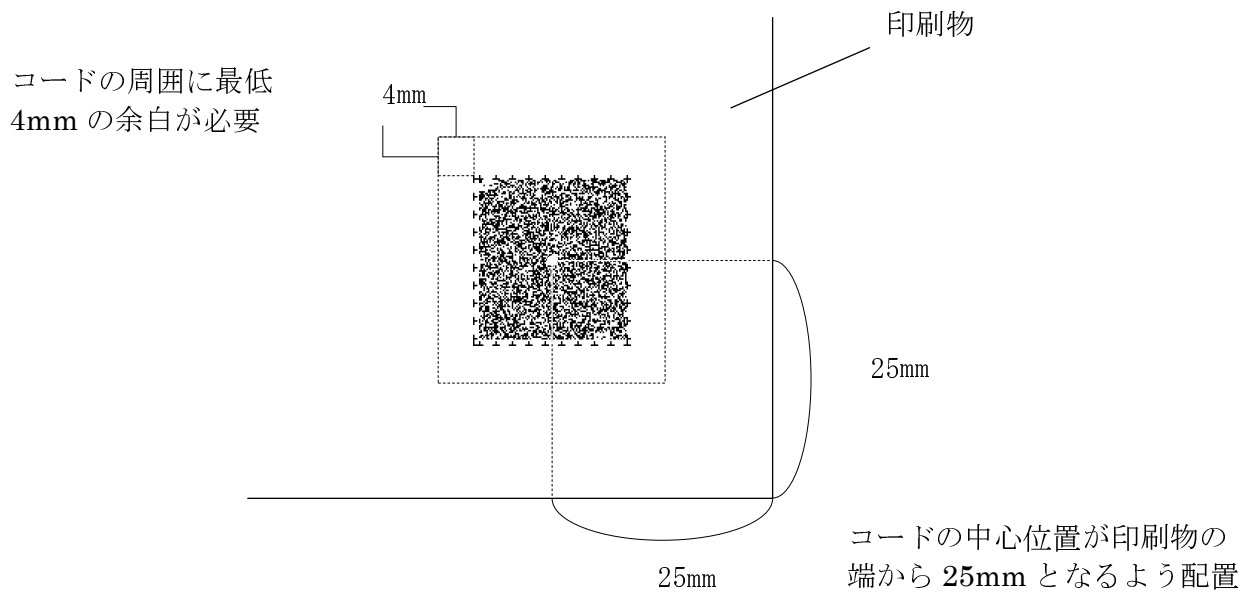


図3 SPコードの位置と余白（右下に配置）

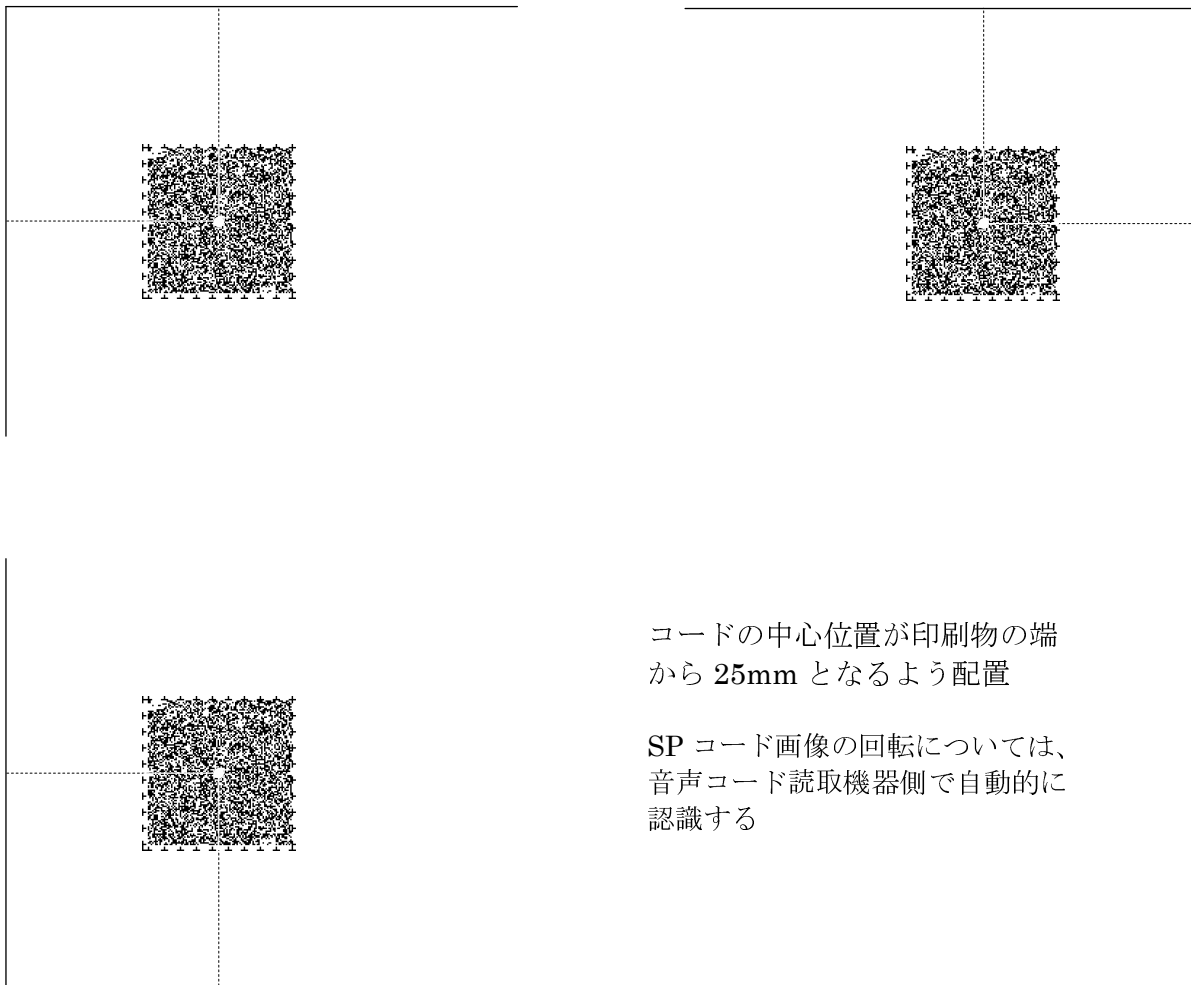


図4 SPコードの配置例

## 4 入力テキストの整形処理

音声コード読取機器では、SP コード内のテキストを音声合成するため、その中に制御コードなどが含まれていると音声合成プログラム内で不具合が発生する可能性がある。このため、エンコード前にテキストデータ内の制御コードを削除するなどの整形処理が必要となる。

また、音声コード読取機器では、特定の制御コードによる男女の声質の切り替えや、漢字の読み方を指定することなどが可能であり、入力テキストに対して追加情報を付与することができる。

### 4-1 制御コード等の削除

音声コード読取機器内部での音声合成処理の不具合を避けるため、エンコード前に入力テキストデータ内の制御コードを削除しておく必要がある。

ここでいう制御コードとは、シフト JIS コード内で 0x01~0x1f、0x7f (16 進数表記) の文字コードのことである。ただし、タブ (0x09) ・改行 (0x0a) ・復帰 (0x0d) は除く。これは、音声コード読取機器内部にて、タブを読点に変換し、改行を一度 0x0d0a に統一してから句点に変換しているためである。

また、記号などが音声コード読取機器で意図した読み方とならない場合は、あらかじめ削除しておくことが望ましい。

### 4-2 読みの指定

音声コード読取機器においてテキストデータを音声出力した際に、目的の読み方で音声出力されない単語がある場合は、その単語の読み方を指定することができる。変換対象のテキストデータに以下の形式で読みを入力する。

(単語:ヨミ)

単語           読み方を指定したい単語 (全角)  
ヨミ           単語の読み方を半角カタカナで入力  
括弧とコロンは、半角の()と:を使用する。

例 株式会社(廣濟堂:コウサイドウ)は

「カブシキガイシャコウサイドウカ」と音声出力される。

単語の後に半角のコロンを、続けて読みを半角カタカナで入力し、単語の始めから読みの終わりまでを半角の丸カッコで囲む。ただし、単語は全角でなくてはならない。

半角の英数字については、上記の方法で読み方を指定することはできない。この場合、英単語の発音をカタカナで、数字を漢数字で入力するなどの工夫が必要となる。

### 4-3 音声コード読取機器での声質の切り替え

音声コード読取機器では、テキストデータの先頭に特定の制御コードを付加することで、音声出力に関する制御を行うことが可能である。また、音声コード読取機器では文単位にテキストデータを分割して音声出力を行うため、文単位に音声の制御を行うことができる。例えば、一文目を男性の声で、二文目を女性の声で音声出力することなどができる。音声出力に関する設定の種類と制御コードは、下表の通りである。

表 3 音声出力用の制御コード

音質	制御コード	種類・レベル	備考
声質	^V0	男声 (デフォルト)	^V=0x16
	^V1	女声	
高さ	^H0	レベル 0 (最低)	^H=0x08
	^H1	レベル 1	
	^H2	レベル 2	
	^H3	レベル 3 (男性のデフォルト)	
	^H4	レベル 4 (女性のデフォルト)	
	^H5	レベル 5	
	^H6	レベル 6	
	^H7	レベル 7 (最高)	
大きさ	^P0	レベル 0 (最小)	^P=0x10
	^P1	レベル 1	
	^P2	レベル 2	
	^P3	レベル 3	
	^P4	レベル 4 (デフォルト)	
	^P5	レベル 5	
	^P6	レベル 6	
	^P7	レベル 7 (最大)	

※ 「制御コード+数字」の形式で構成されている。

文頭に制御コードが付加されていない場合は、前文と同じ設定内容で音声出力される。このため、文毎に男女の声を変更する場合は、以下のように制御コードを付加すればよい。

^V0^H3^P4 この文は男性の声で音声出力されます。

^V1^H4 この文は女性の声で音声出力されます。

^V0^H3 この文は再び男性の声で音声出力されます。

制御コードを何も指定しない場合は、表 3 内のデフォルトで設定された声質で音声出力される。つまり、^V0^H3^P4 を指定した場合と同様で、男声・高さレベル 3・大きさレベル 4 の声質となる。

また、制御コードは必ず文の先頭に付加しなければならない。ここで、文とは、音声コード読取機器で判定されるものであり、音声コード読取機器では表 4 の文字を文末として判定する。このため、入力テキストデータの文末を判定して、文単位に分割し、前文と異なる制御コードのみを付加するようにする。

表 4 音声コード読取機器で文末として判定される文字

。)	。) )	? )	? )	! )	! )	。
。]	。) ]	? ]	? ]	! ]	! ]	。
。}	。) }	? }	? }	! }	! }	?
。>	。) >	? >	? >	! >	! >	!
。》	。) 》	? 》	? 》	! 》	! 》	!
。↓	。) ↓	? ↓	? ↓	! ↓	! ↓	CR+LF
。』	。) 』	? 』	? 』	! 』	! 』	
。】	。) 】	? 】	? 】	! 】	! 】	
。)	。) )	? )	? )	! )	! )	
。]	。) ]	? ]	? ]	! ]	! ]	
。}	。) }	? }	? }	! }	! }	
。↓	。) ↓	? ↓	? ↓	! ↓	! ↓	
。”	。) ”	? ”	? ”	! ”	! ”	
。’	。) ’	? ’	? ’	! ’	! ’	

※1 連続する文末文字は、一つの文末文字として認識される。

※2 CR=0x0d、LF=0x0a をあらかわす。

## 5 その他条件

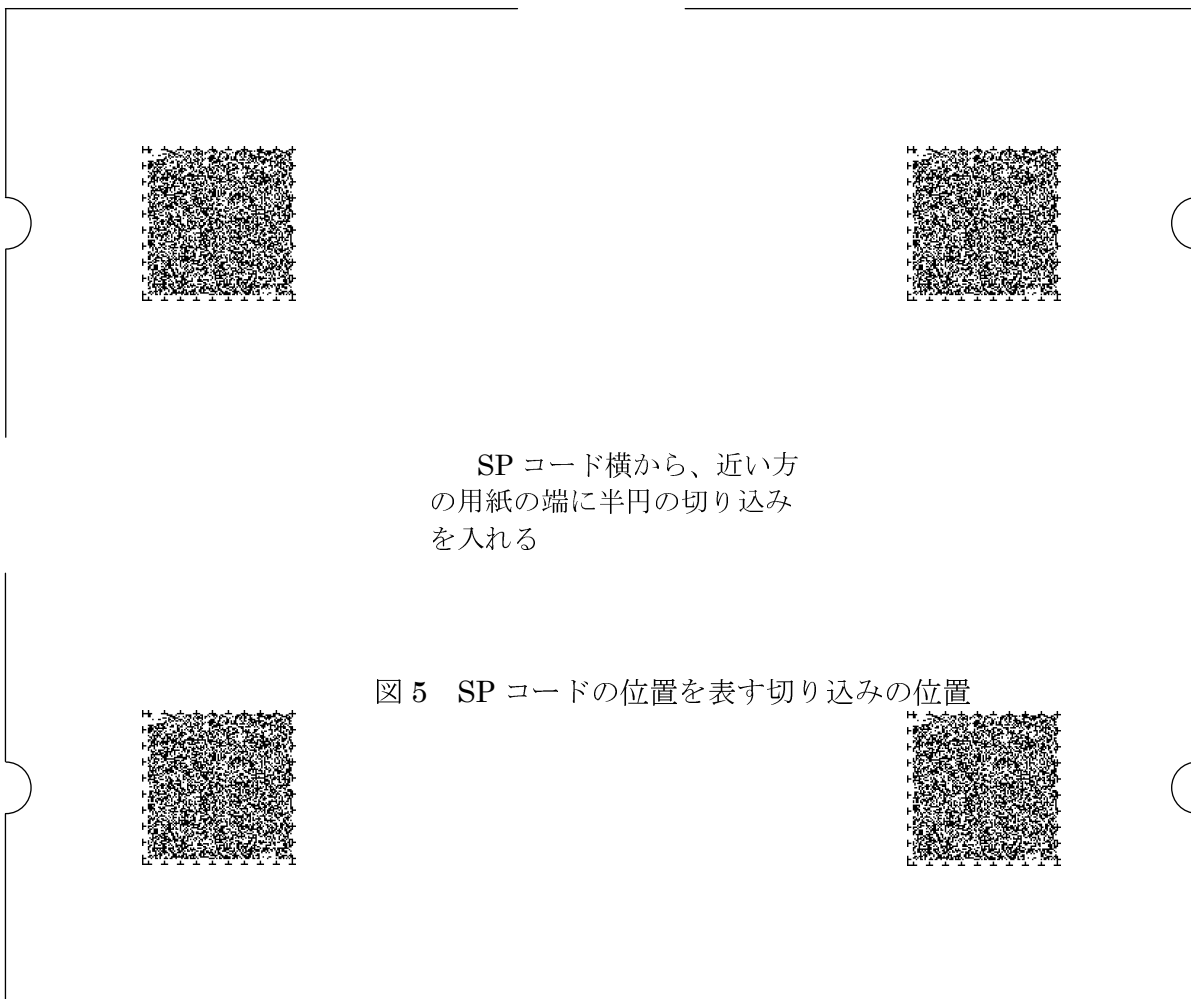
---

コード生成時の用紙の条件や用紙の加工（切り込み）、印刷条件等を説明する。

### 5-1 コードの位置を表す切り込みについて

---

印刷した SP コードの横には、視覚障害者が触覚によりコードの位置を把握できるよう、印刷物に半円の切り込みを入れることを推奨している（下図参照）。切り込みは、市販の穴あけパンチを利用するなどして、印刷物を加工する程度で構わない。



### 5-2 用紙の指定について

---

SP コードを印刷する用紙は、一般的なコピー用紙や再生紙等での印刷を推奨する。音声コード読取機器では CCD カメラによって、SP コードを撮影・解析するため、光沢紙のように光を反射する用紙、地の色が濃い色上質紙、凹凸のある加工紙では、場合によっては読み取りに支障をきたす可能性がある。