

IT4xxxシリーズ  
リニアイメージ&エリアイメージ

専用 DLL 仕様書

# 改訂記録

改訂番号	改訂日	内容
Rev.1.0	Dec.2005	初版

- 1 . 本書の著作権はウェルコムデザイン株式会社に属します。
- 2 . 本書の内容は、改良のため、将来予告無く変更する場合があります。
- 3 . 本書の全部または一部を無断で複製することはできません。
- 4 . 本書内に記載している製品名等の固有名詞は、弊社または各社の商標または登録商標です。
- 5 . 本書内において、誤り・記載漏れなど、お気付きの点がありましたらご連絡ください。
- 6 . 本D L Lを運用した結果生じる影響について、弊社では一切責任を負いません。  
あらかじめご了承ください。

## 目 次

<b>1</b>	<b>はじめに</b>	<b>3</b>
1.1	機能概要	3
1.2	動作環境	3
1.3	提供ファイル	3
1.4	注意事項	4
<b>2</b>	<b>処理概要</b>	<b>5</b>
2.1	処理開始～終了の手順	5
2.2	送信処理	5
2.3	受信処理	6
2.4	イメージデータの受信について	6
<b>3</b>	<b>関数詳細説明</b>	<b>7</b>
3.1	ポートオープン [IT4xxx_PortOpen]	8
3.2	ポートクローズ [IT4xxx_PortClose]	10
3.3	タイムアウト設定 [IT4xxx_TimeoutSet]	11
3.4	コマンド送信 [IT4xxx_CmndSend]	13
3.5	データ送信 [IT4xxx_DataSend]	14
3.6	受信通知スレッド開始 [IT4xxx_RecvStart]	15
3.7	受信通知スレッド終了 [IT4xxx_RecvStop]	17
3.8	受信バイト数取得 [IT4xxx_RecvChk]	18
3.9	バーコードデータ受信 [IT4xxx_DataRecv]	19
3.10	コマンドレスポンス受信 [IT4xxx_ResponseRecv]	21
3.11	イメージデータ受信 [IT4xxx_ImageRecv]	23
3.12	イメージデータ受信状態取得 [IT4xxx_ImageProgress]	25
3.13	通信の中止 [IT4xxx_ComCancel]	26
3.14	ポートのクリア [IT4xxx_ComClear]	27

Blank Page

## 1 はじめに

本書は、弊社「ITシリーズ リニアイメージ&エリアイメージ」専用ライブラリ「IT4xxx.dll」の仕様について記載したものです。ITシリーズスキャナの手配の細は、「リニアイメージ&エリアイメージ取扱説明書」を参照ください。

### 1.1 機能概要

IT4xxx.dll(以下、本DLL)には、以下の関数があります。

- 通信ポートのオープン・クローズ
- 送信・受信機能
- 受信チェック機能
- キャンセル・バッファクリア

### 1.2 動作環境

対応OS : Windows 98/Me/NT/2000/XP  
動作確認済み開発環境 : Visual C++ Ver 6.0  
Visual Basic Ver6.0  
Visual Basic .NET

### 1.3 提供ファイル

【DLL】 Windows¥System32 または実行フォルダにコピーしてお使いください。

IT4xxx.dll IT シリーズスキャナ専用 DLL

※1 UTILIT4x-WIN をインストールすると、自動的にコピーされます。

※2 本DLLを使用したプログラムを配布する場合は、配布先のコンピュータにも、コピーしてください。

【VCで使用するファイル】 VCの開発フォルダにコピーしてお使いください。

IT4xxx.lib インポートライブラリ(明示的リンクの場合不要)

IT4xxx.h インクルードファイル

【VBで使用するファイル】 VBの開発フォルダにコピーしてお使いください。

IT4xxx.bas IT4xxx.dll をVB6で使用するための標準モジュール

IT4xxx.vb IT4xxx.dll をVB.NET で使用するための標準モジュール

【サンプル】 VB・VCのコーディングサンプルです

## 1. 4 注意事項

### 使用ライセンスと著作権について

- ・本DLLは、IT シリーズ専用ライブラリです。弊社よりご購入いただいた IT シリーズスキャナに使用する場合に限り、ライセンス料なしでご使用いただけます。
- ・本DLL及び本DLLに付属する全てのプログラム・サンプルファイル・ドキュメント等の著作権はウェルコムデザイン株式会社に属します。弊社よりご購入いただいたスキャナに使用するために本DLLを再配布する場合を除き、全部または一部を弊社に無断で複製することはできません。

### 本書の表記について

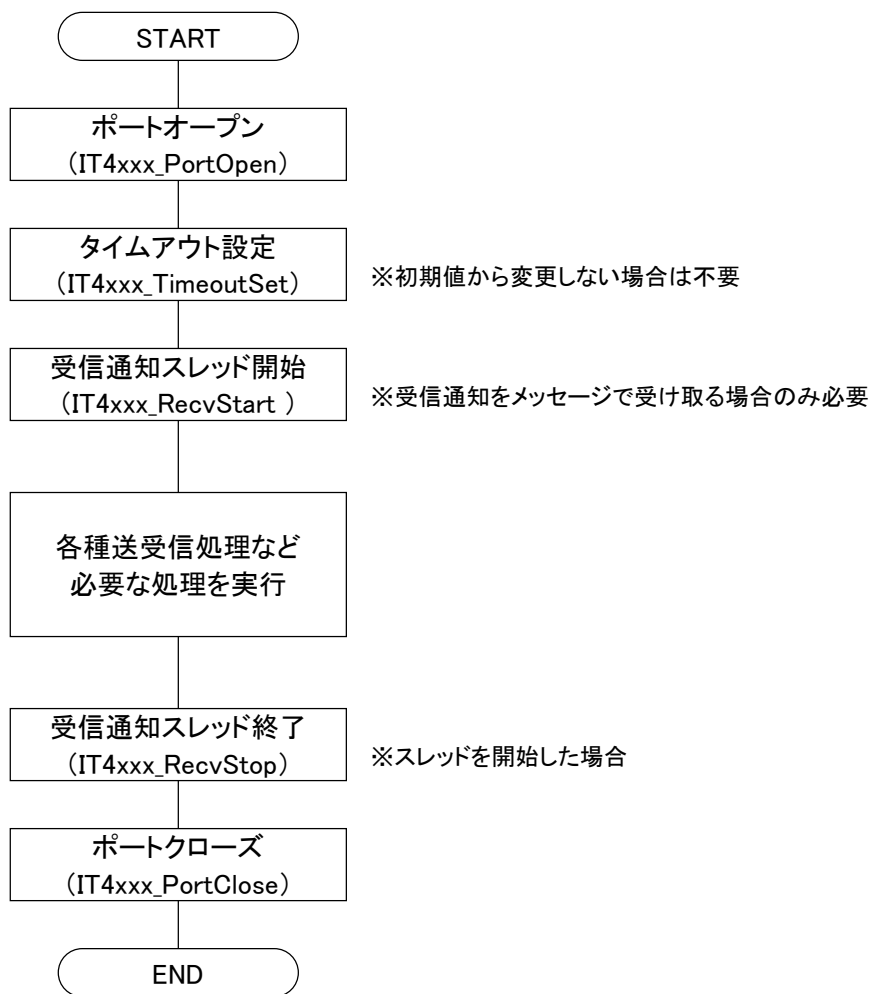
- ・本書の中では、以下の略称を使用する場合があります。
  - DLL — ライブラリ／Dynamic Link Library
  - PC — パーソナルコンピュータ
  - VB — Visual Basic
  - VC — Visual C++
- ・16進数は、xxH と表記しています。

## 2 処理概要

ここでは、処理の概要を説明します。各関数の詳細は次章を参照してください。

### 2.1 処理開始～終了の手順

処理開始から終了までの手順は以下の通りです。



### 2.2 送信処理

送信処理関数をコールしてください。

コマンドの送信・・・IT4xxx\_CmndSend

データの送信・・・IT4xxx\_DataSend

## 2. 3 受信処理

データを受信する関数は、下記の3種類です。

IT4xxx\_DataRecv      ... バーコードデータを受信します。

IT4xxx\_ResponseRecv      ... コマンドレスポンスを受信します。

IT4xxx\_ImageRecv      ... イメージデータを受信します。

読み取りデータを受信する場合は、IT4xxx\_DataRecv を使用してください。

IMGSHIP、IMGBOX コマンド送信後は、IT4xxx\_ImageRecv を、それ以外のコマンド送信後は、IT4xxx\_ResponseRecv をコールしてください。(シリアルトリガコマンドはコマンドレスポンスなし)

データを受信するには、下記の3つの方法があります。

### 【受信チェックを行わず受信関数をコールする】

対応する受信関数をコールしてください。データを受信しなかった場合は、IT4xxx\_ERR\_TIMEOUT が返されますので、必要な処理(再受信、エラー処理など)を行ってください。

### 【一定の時間ごとに受信バッファをチェックする】

タイマーを使用して、一定時間ごとに、受信バイト数取得関数(IT4xxx\_RecvChk)をコールします。受信していた場合には、対応する受信関数をコールしてください。

### 【データ受信時に、メッセージを受け取って処理する】

受信通知スレッド開始関数(IT4xxx\_RecvStart)をコールし、メッセージを受け取るウィンドウのハンドルと、メッセージ番号を指定します。指定したメッセージを受け取った際に、対応する受信関数をコールします。

## 2. 4 イメージデータの受信について

イメージングコマンドには、次の3種類があります。

IMGSNIP コマンド...画像の撮影。撮影したデータはスキャナのメモリに保持されます。

IMGSHIP コマンド...IMGSNIP で撮影したイメージの送信。メモリに保持されているデータが PC に送信されます。

IMGBOX コマンド...サインキャプチャ。指定したサインエリアの画像データが PC に送信されます。(トリガを引いたときの画像がスキャナに自動的に保持されています。)

IMGSNIP コマンドでは、撮影を行うだけで、データの送信は行われません。IMGSNIP コマンド発行後はコマンドレスポンス受信(IT4xxx\_ResponseRecv)をコールしてください。

IMGSHIP、IMGBOX コマンドを受け付けると、受信が終了するまで、スキャナはデータを送信し続けます。送信されるデータを処理しないと PC の動作が不定になる場合がありますので、これらのコマンドを発行した後は、必ず、イメージデータ受信関数(IT4xxx\_ImageRecv)を実行してください。



### 3 関数詳細説明

次ページより、本DLLの各関数の仕様について、詳細を説明します。

各関数の説明は以下の構成になっています。

#### X. XX 機能名称[関数名]

(VC) VCの呼び出し形式

(VB) VBの呼び出し形式 (Visual Basic Ver6 の形式で記載しています)

#### 【引 数】

関数で使用する引数について説明しています。

(例)

引 数	I/O	説 明
Port	I	ポート番号

引数の意味 および  
設定する値の説明

#### 引数の種別

I - DLL 側で値を参照します  
O - DLL 側で値を設定します  
IO - DLL 側で値を参照し、  
値を設定します

#### 【戻り値】 戻り値の意味

(例)

定 義	意 味
IT4xxx_NORMAL_END	正常終了

戻り値の定義名または値です。  
できる限り定義名を使用してください。  
実際の値についてはインクルードファイル等を  
参照してください

戻り値の見方・意味について  
説明しています

#### 【機能詳細】

関数の機能の詳細説明です。

#### 【注意事項】

関数を使用する際の注意事項を記載しています。

#### 【コーディング例】

VCおよびVBでのコーディングの一例です。

VBで2行以上の文に必要な、' 'は省略しています。

### 3. 1 ポートオープン[IT4xxx\_PortOpen]

(VC) IT4xxx\_PortOpen( long Port, long Baud, long DataBit, long StopBit, long Parity, long FlowCtrl );

(VB) Function IT4xxx\_PortOpen(ByVal Port As Long, ByVal baud As Long, \_  
ByVal DataBit As Long, ByVal StopBit As Long, ByVal parity As Long, \_  
ByVal FlowCtrl As Long ) As Long

#### 【引 数】

引 数	I/O	説 明
Port	I	ポート番号(1～256)
Baud	I	ボーレート IT4xxx_BAUD_300            300Bps IT4xxx_BAUD_600            600Bps IT4xxx_BAUD_1200          1200Bps IT4xxx_BAUD_2400          2400Bps IT4xxx_BAUD_4800          4800Bps IT4xxx_BAUD_9600          9600Bps IT4xxx_BAUD_19200         19200Bps IT4xxx_BAUD_38400         38400Bps IT4xxx_BAUD_57600         57600Bps IT4xxx_BAUD_115200        115200Bps
DataBit	I	データビット IT4xxx_DATABIT_7 - 7ビット IT4xxx_DATABIT_8 - 8ビット
StopBit	I	ストップビット IT4xxx_STOPBIT_1 - 1ビット IT4xxx_STOPBIT_2 - 2ビット
Parity	I	パリティ IT4xxx_PARITY_NONE - なし IT4xxx_PARITY_EVEN - 偶数パリティ IT4xxx_PARITY_ODD - 奇数パリティ
FlowCtrl	I	ハンドシェイク IT4xxx_FLOW_NONE - なし IT4xxx_FLOW_XONXOFF - XON/XOFF IT4xxx_FLOW_RTSCTS - RTS/CTS

## 【戻り値】

通信ポート ID またはオープン結果が返されます。

定 義	意 味
0以上	通信ポート ID (以降の関数で使用)
IT4xxx_ERR_NOTOPEN	オープンエラー
IT4xxx_ERR_TIMEOUT	受信タイムアウト(コマンドに対する応答がない)
IT4xxx_ERR_CANCEL	受信キャンセル
IT4xxx_ERR_PARAMETER	関数パラメータエラー
IT4xxx_ERR_COM	通信エラー
IT4xxx_ERR_FULL	受信バッファフル

## 【機能詳細】

指定された設定で、ポートのオープンを行います。

オープン後、通信状態をチェックするために、スキャナにテスト用のコマンドを送信し、レスポンスを受信します。正常にオープンできた場合は、以降の関数で使用する、通信ポート ID (0以上) が返されます。値が<0の場合はエラーです。

## 【コーディング例】

```
(VC) long      ComId ;
      if((ComId=IT4xxx_PortOpen(5,IT4xxx_BAUD_9600, IT4xxx_DATABIT_8, IT4xxx_STOPBIT_1,
                                IT4xxx_PARITY_NONE, IT4xxx_FLOW_RTSCTS)) < 0 )
          MessageBox ( hDlg, "ポートがオープンできません", "VCTest", MB_OK );

(VB) Dim ComId As Long
      ComId = IT4xxx_PortOpen(5, IT4xxx_BAUD_9600, IT4xxx_DATABIT_8, IT4xxx_STOPBIT_1,
                              IT4xxx_PARITY_NONE, IT4xxx_FLOW_RTSCTS)

      If ComId < 0 Then
          MsgBox "ポートオープンエラー"
      End If
```

### 3. 2 ポートクローズ [IT4xxx\_PortClose]

(VC) long IT4xxx\_PortClose( long ComId ) ;

(VB) Function IT4xxx\_PortClose Lib "IT4xxx.dll" (ByVal ComId As Long) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)

#### 【戻り値】

定 義	意 味
IT4xxx_NORMAL_END	正常終了
IT4xxx_ERR_NOTOPEN	オープンされていない
IT4xxx_ERR_PARAMETER	関数パラメータエラー
IT4xxx_ERR_IMGRCV	イメージデータ処理中

#### 【機能詳細】

ポートをクローズします。

#### 【注意事項】

イメージデータの送受信処理中 (バッファクリアを含む) はクローズできません。

スキャナがデータを送信している状態で、ポートをクローズすると、コンピュータの動作が不安定になる場合があります。

#### 【コーディング例】

```
(VC) if( IT4xxx_PortClose( ComId ) != IT4xxx_NORMAL_END )
    return ;
```

```
(VB) if IT4xxx_PortClos( ComId ) <> IT4xxx_NORMAL_END THEN
    Exit Sub
End If
```

### 3.3 タイムアウト設定 [IT4xxx\_TimeoutSet]

(VC) void IT4xxx\_TimeoutSet( long ComId, long Wait, long Interval );

(VB) Sub IT4xxx\_TimeoutSet (ByVal ComId As Long, ByVal Wait As Long, ByVal Interval As Long)

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)
Wait	I	受信タイムアウト時間 (ms) 0を指定した場合は、タイムアウトなし 0未満の数値を指定した場合は、初期値 (10000ms) 受信関数をコールしてから、最初の文字を受信するまでのタイムアウトを設定します。 全ての受信関数に作用します。
Interval	I	インターバルタイムアウト時間 (ms) 0を指定した場合は、タイムアウトなし 0未満の数値を指定した場合は、初期値 (300ms) 次の文字を受信するまでのタイムアウトを設定します。 バーコードデータ受信関数に作用します。

#### 【戻り値】

なし

#### 【機能詳細】

全受信関数 (IT4xxx\_DataRecv、IT4xxx\_ResponseRecv、IT4xxx\_ImageRecv) の受信タイムアウト時間、およびバーコードデータ受信関数 (IT4xxx\_DataRecv) のインターバルタイムアウト時間を設定します。いずれも10ms単位の値に切り上げられます。

#### 【注意事項】

タイムアウト時間は、実行時の PC の負荷などによって、指定した時間よりも多少長くなる場合があります。

【コーディング例】

(VC) long ComId ;

```
if( (ComId = IT4xxx_PortOpen( 5, IT4xxx_BAUD_9600, IT4xxx_DATABIT_8, IT4xxx_STOPBIT_1,  
    IT4xxx_PARITY_NONE, IT4xxx_FLOW_RTSCTS, FALSE ) ) >= 0 ) {  
    IT4xxx_TimeoutSet( ComId, -1, 100 ) ;  
}
```

(VB) Dim ComId As Long

```
ComId = IT4xxx_PortOpen(5, IT4xxx_BAUD_9600, IT4xxx_DATABIT_8, IT4xxx_STOPBIT_1,  
    IT4xxx_PARITY_NONE, IT4xxx_FLOW_RTSCTS, True)  
If ComId >= 0 Then  
    IT4xxx_TimeoutSet ComId, -1, 100  
End If
```

### 3. 4 コマンド送信 [IT4xxx\_CmndSend]

(VC) long IT4xxx\_CmndSend( long ComId, unsigned char \*Cmnd, long SendCnt );

(VB) Function vbIT4xxx\_CmndSend(ByVal ComId As Long, ByVal Cmnd As String) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)
Cmnd	I	送信するコマンド
SendCnt	I	送信するバイト数

#### 【戻り値】

定 義	意 味
IT4xxx_NORMAL_END	正常終了
IT4xxx_ERR_NOTOPEN	ポートがオープンされていない
IT4xxx_ERR_TIMEOUT	タイムアウト
IT4xxx_ERR_PARAMETER	パラメータエラー
IT4xxx_ERR_CANCEL	ユーザーによる中断

#### 【機能詳細】

Cmnd で指定された文字列の前に、プリフィックス(<SYN>M<CR>)を付加、最後がメモリフィールド(!または.)でない場合は、「.」を付加して送信します。

メニューコマンドおよびイメージングコマンドの送信に使用します。

#### 【コーディング例】

(VC) ret = IT4xxx\_CmndSend( ComId, senddata, strlen(senddata) );

if( ret != IT4xxx\_NORMAL\_END )

MessageBox ( hDlg, "送信エラーが発生しました", NULL, MB\_OK );

(VB) ret = vbIT4xxx\_CmndSend(ComId, SerialCmnd.Text)

If ret <> IT4xxx\_NORMAL\_END Then

MsgBox "送信エラーが発生しました"

End If

### 3. 5 データ送信 [IT4xxx\_DataSend]

(VC) long IT4xxx\_DataSend( long ComId, unsigned char \*WriteData, long WriteCnt) ;

(VB) Function vbIT4xxx\_DataSend(ByVal ComId As Long, ByVal WriteData As String) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)
WriteData	I	送信するデータ
WriteCnt	I	送信するバイト数

#### 【戻り値】

定 義	意 味
IT4xxx_NORMAL_END	正常終了
IT4xxx_ERR_NOTOPEN	ポートがオープンされていない
IT4xxx_ERR_TIMEOUT	タイムアウト
IT4xxx_ERR_PARAMETER	パラメータエラー
IT4xxx_ERR_CANCEL	ユーザーによる中断

#### 【機能詳細】

指定されたバイト数分、データを送信します。シリアルトリガコマンドの送信などに使用します。

#### 【注意事項】

メニューコマンドやイメージングコマンドを送信する場合は、先頭にプリフィックス(<SYN>M<CR>)、最後にメモリフィールド(!または.)が必要です。

#### 【コーディング例】

(VC) int ret ;

```
ret = IT4xxx_DataSend( ComId, "¥x16T¥x0D", 3 ) ;
```

(VB) Dim trcmd As String

```
trcmd = Chr$(&H16) + "T" + Chr$(&HD)
```

```
vbIT4xxx_DataSend ComId, trcmd
```



### 3. 6 受信通知スレッド開始 [IT4xxx\_RecvStart]

(VC) long IT4xxx\_RecvStart ( long ComId, HWND Wnd, long MsgNo );

(VB) Function IT4xxx\_RecvStart (ByVal ComId As Long, ByVal Wnd As Long,  
ByVal MsgNo As Long) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)
Wnd	I	メッセージを受信するウィンドウハンドル
MsgNo	I	データを受信した際に送信されるメッセージの番号 (注意事項参照)

#### 【戻り値】

定 義	意 味
IT4xxx_ERR_NOTOPEN	ポートがオープンされていない
IT4xxx_ERR_PARAMETER	パラメータエラー
IT4xxx_ERR_OTHER	スレッドの作成・開始に失敗した

#### 【機能詳細】

受信通知スレッドを開始します。

#### 【注意事項】

メッセージ番号には、下記の値を使用することをお勧めします。

(VC) 他のメッセージと重複しないよう、WM\_APP+X (X は任意の数値) を指定してください

(VB) 通常のイベントとして処理できるよう、既存のメッセージ番号を指定してください。

例) ダブルクリックイベントで処理する場合 = WM\_LBUTTONDOWNBLCLK (203H)

【コーディング例】

```
(VC) #define   RECVMSG               (WM_APP+10)

      if( IT4xxx_RecvStart( ComId, hDlg, RECVMSG ) != IT4xxx_NORMAL_END ){
          MessageBox ( hDlg, "スレッドが作成できません", "VCTest", MB_OK );
          IT4xxx_PortClose( ComId );
          return ;
      }

(VB) Private Const WM_LBUTTONDBLCLK = &H203
      if IT4xxx_RecvStart(ComId,SerialCmnd.hWnd,WM_LBUTTONDBLCLK)
          <> IT4xxx_NORMAL_END Then
              IT4xxx_PortClose ComId
              MsgBox "スレッド作成エラー"
              Exit Sub
      End If
```

### 3. 7 受信通知スレッド終了 [IT4xxx\_RecvStop]

(VC) void IT4xxx\_RecvStop ( long ComId );

(VB) Sub IT4xxx\_RecvStop (ByVal ComId As Long)

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)

#### 【戻り値】

なし

#### 【機能詳細】

受信通知スレッドを終了します。

#### 【コーディング例】

(VC) IT4xxx\_RecvStop( ComId );

(VB) IT4xxx\_RecvStop ComId

### 3. 8 受信バイト数取得 [IT4xxx\_RecvChk]

(VC) long IT4xxx\_RecvChk( long ComId );

(VB) Function IT4xxx\_RecvChk (ByVal ComId As Long) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)

#### 【戻り値】

ポートに届いているデータのバイト数が返されます。

定 義	意 味
0	受信なし
1 以上	通信ポートに届いているデータのバイト数

#### 【機能詳細】

指定した通信ポートに届いているデータの、バイト数を取得します。

ポートがオープンされていない場合や、データを受信していない場合は0が返されます。

#### 【コーディング例】

(VC) unsigned char readbuf[1024]

long recvsize ;

if( vbIT4xxx\_RecvChk(ComId) > 0)

recvsize = IT4xxx\_DataRecv(ComId, readbuf, sizeof(readbuf), NULL, NULL)

(VB) Dim readbuf As String

Dim recvsize As Long

If IT4xxx\_RecvChk(ComId) > 0 Then

recvsize = vbIT4xxx\_DataRecv(ComId, readbuf, 1024, "", "")

End

### 3. 9 バーコードデータ受信 [IT4xxx\_DataRecv]

(VC) long IT4xxx\_DataRecv( long ComId, unsigned char \*RcvBuffer, long BufSize,  
unsigned char \*PreFix, unsigned char \*PostFix );

(VB) Function vbIT4xxx\_DataRecv(ByVal ComId As Long, ByRef RcvBuffer As String,  
ByVal BufSize As Long, ByVal PreFix As String, ByVal PostFix As String) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)
RcvBuffer	O	受信データ格納領域
BufSize	I	受信データ格納領域のバイト数
PreFix	I	プリフィックス
PostFix	I	ポストフィックス

#### 【戻り値】

受信バイト数またはエラー番号が返されます

定 義	意 味
1 以上	受信バイト数(正常終了)
IT4xxx_ERR_NOTOPEN	ポートがオープンされていない
IT4xxx_ERR_COM	通信エラー
IT4xxx_ERR_TIMEOUT	タイムアウト(データを1文字も受信しなかった場合)
IT4xxx_ERR_FULL	受信バッファフル
IT4xxx_ERR_CANCEL	ユーザーによる中断
IT4xxx_ERR_PARAMETER	パラメータエラー

#### 【機能詳細】

バーコードデータを受信します。

プリフィックスが指定されている場合、プリフィックス受信までのデータは無視します。

ポストフィックスが指定されている場合、ポストフィックスの受信で、関数を終了します。

また、データ受信後、次の文字の受信までにインターバルタイムアウトで指定した時間を経過した場合、関数を終了します。

受信したデータを RcvBuffer に格納し、受信バイト数を返します。NULL コードは付加しません。

【注意事項】

当関数コール後、受信タイムアウトで指定した時間を経過してもデータを全く受信しなかった場合に、タイムアウトエラー (IT4xxx\_TIMEOUT) となります。

タイムアウト時間の指定方法は、IT4xxx\_TimeoutSet を参照してください。

【コーディング例】

(VC) #define RECVBUFSIZE 1024

long recvcnt ;

unsigned char recvdata[RECVBUFSIZE] ;

recvcnt = IT4xxx\_DataRecv( ComId, recvdata, RECVBUFSIZE, NULL, NULL ) ;

(VB) Dim readbuf As String

Dim recvsize As Long

recvsize = vbIT4xxx\_DataRecv(ComId, readbuf, 1024, "", "")

### 3. 10 コマンドレスポンス受信 [IT4xxx\_ResponseRecv]

(VC) long IT4xxx\_ResponseRecv( long ComId, unsigned char \*RcvBuffer, long BufSize );

(VB) Function vbIT4xxx\_ResponseRecv(ByVal ComId As Long, ByRef RcvBuffer As String,  
ByVal BufSize As Long) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)
RcvBuffer	O	レスポンス格納領域
BufSize	I	レスポンス格納領域のバイト数

#### 【戻り値】

レスポンスのバイト数またはエラー番号が返されます

定 義	意 味
1 以上	レスポンスのバイト数 (正常終了)
IT4xxx_ERR_NOTOPEN	ポートがオープンされていない
IT4xxx_ERR_COM	通信エラー
IT4xxx_ERR_TIMEOUT	タイムアウト (データを1文字も受信しなかった場合)
IT4xxx_ERR_FULL	受信バッファフル
IT4xxx_ERR_CANCEL	ユーザーによる中断
IT4xxx_ERR_PARAMETER	パラメータエラー

#### 【機能詳細】

コマンドレスポンスを受信します。

メモリフィールド (!または.) 文字を受信すると、関数を終了します。

また、データ受信後、次の文字の受信までに規定の時間 (1000ms) を経過した場合にも、関数を終了します。

受信したレスポンスを RcvBuffer に格納し、受信バイト数を返します。NULL コードは付加しません。

#### 【注意事項】

- ・イメージ受信コマンド (IMGSHIP、IMGBOX) 以外のコマンドを送信した後に、コールしてください。
- ・IMGSNP コマンドで、トリガ待ち (1T) を指定した場合、正常実行のレスポンス (ACK) は、トリガが押された後に返されますのでご注意ください。
- ・当関数コール後、受信タイムアウトで指定した時間を経過してもデータを全く受信しなかった場合に、タイムアウトエラー (IT4xxx\_TIMEOUT) となります。受信タイムアウト時間の指定方法は、IT4xxx\_TimeoutSet を参照してください。

【コーディング例】

(VC) #define RECVBUFSIZE 1024

long recvnt ;

unsigned char recvdata[RECVBUFSIZE] ;

recvnt = IT4xxx\_ResponseRecv( ComId, recvdata, RECVBUFSIZE ) ;

(VB) Dim readbuf As String

Dim recvsize As Long

recvsize = vbIT4xxx\_ResponseRecv(ComId, readbuf, 1024)



### 3. 11 イメージデータ受信 [IT4xxx\_ImageRecv]

(VC) long IT4xxx\_ImageRecv( long ComId, unsigned char \*Filename,

unsigned char \*RcvBuffer, long BufSize, long Protocol );

(VB) Function vbIT4xxx\_ImageRecv(ByVal ComId As Long, ByVal Filename As String,

ByRef RcvBuffer As String, ByVal BufSize As Long, ByVal Protocol As Long) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)
Filename	I	イメージ保存ファイル名
RcvBuffer	O	レスポンス格納領域
BufSize	I	レスポンス格納領域のバイト数
Protocol	I	受信プロトコル IT4xxx_PROTOCOL_2P - 「2P プロトコル無し」で受信 ※上記以外のプロトコルは指定できません。

#### 【戻り値】

レスポンスのバイト数またはエラー番号が返されます

定 義	意 味
1 以上	レスポンスのバイト数 (正常終了)
IT4xxx_ERR_NOTOPEN	ポートがオープンされていない
IT4xxx_ERR_COM	通信エラー
IT4xxx_ERR_TIMEOUT	タイムアウト (データを1文字も受信しなかった場合)
IT4xxx_ERR_FULL	レスポンス格納領域バッファフル
IT4xxx_ERR_CANCEL	ユーザーによる中断
IT4xxx_ERR_PARAMETER	パラメータエラー

#### 【機能詳細】

イメージデータを受信し、指定されたファイルに保存します。イメージを受信した後、コマンドレスポンスを受信し、メモリフィールド (!または.) 文字を受信すると、関数を終了します。

また、データ受信後、次の文字の受信までに規定の時間 (1000ms) を経過した場合も、関数を終了します。

受信したレスポンスを RcvBuffer に格納し、レスポンスのバイト数を返します。レスポンスには、NULL コードは付加しません。

【注意事項】

- ・「2P プロトコル無し(USB のデフォルト)」以外のプロトコルでは使用できません。  
IMGSHp、IMGBOX 送信時には「2P」を指定してください。
- ・IMGSHp、IMGBOX コマンドを送信した後に、必ずコールしてください。
- ・当関数コール後、受信タイムアウトで指定した時間を経過してもデータを全く受信しなかった場合に、タイムアウトエラー(IT4xxx\_TIMEOUT)となります。受信タイムアウト時間の指定方法は、IT4xxx\_TimeoutSet を参照してください。
- ・イメージ受信中に IT4xxx\_ComCancel 関数で受信処理を中断することはできますが、スキャナからのデータ送信を中止することはできません。送られてくるデータをクリアするために、受信と同程度の時間を要します。
- ・イメージ受信時、ポートのクローズはできません。

【コーディング例】

(VC) #define RECVBUFSIZE 1024

```
long recvcnt ;  
unsigned char recvdata[RECVBUFSIZE] ;  
recvcnt = IT4xxx_ImageRecv( ComId, "RecvImage.jpg", recvdata,  
                           RECVBUFSIZE , IT4xxx_PROTOCOL_2P) ;
```

(VB) Dim readbuf As String

```
Dim recvsize As Long  
recvsize = vbIT4xxx_ImageRecv(ComId, "RecvImage.jpg", readbuf,  
                               1024, IT4xxx_PROTOCOL_2P)
```

### 3. 12 イメージデータ受信状態取得 [IT4xxx\_ImageProgress]

(VC) long IT4xxx\_ImageProgress( long ComId, long \*ImgSize, long \*ImgRecv );

(VB) Function IT4xxx\_ImageProgress (ByVal ComId As Long, ByRef ImgSize As Long,  
ByRef ImgRecv As Long) As Long

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)
ImgSize	O	イメージデータのサイズ (バイト数)
ImgRecv	O	受信済みバイト数

#### 【戻り値】

定 義	意 味
IT4xxx_ERR_NOTOPEN	ポートがオープンされていない
IT4xxx_IMG_NOTEXECUTE	実行していない
IT4xxx_IMG_RECEIVING	イメージデータ受信中
IT4xxx_IMG_BUFFERCLEAR	受信バッファクリア中

#### 【機能詳細】

イメージデータの受信状況を取得します。

ImgSize および ImgRecv の値は、関数の戻り値によって異なります。

IT4xxx_ERR_NOTOPEN	...	0
IT4xxx_IMG_NOTEXECUTE	...	前回実行したイメージデータの情報
IT4xxx_IMG_RECEIVING	...	現在の受信状況
IT4xxx_IMG_BUFFERCLEAR	...	実行を中断した時点での受信状況

#### 【コーディング例】

```
(VC) long ret ;
      long filesize,recvsize ;
      ret = IT4xxx_ImageProgress( ComId, &filesize, &recvsize) ;
```

```
(VB) Dim ret As Long
      Dim filesize As Long
      Dim recvsize As Long
      ret = IT4xxx_ImageProgress(ComId, filesize, recvsize)
```

### 3. 13 通信の中止 [IT4xxx\_ComCancel]

(VC) void IT4xxx\_ComCancel( long ComId );

(VB) Sub IT4xxx\_ComCancel (ByVal ComId As Long)

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)

#### 【戻り値】

なし

#### 【機能詳細】

送受信処理を中断します。

実行中の送受信関数は、バッファのクリアを行ったのち、IT4xxx\_ERR\_CANCEL を返します。

#### 【コーディング例】

(VC) IT4xxx\_ComCancel(ComId);

(VB) IT4xxx\_ComCancel ComId

### 3. 14 ポートのクリア [IT4xxx\_ComClear]

(VC) void IT4xxx\_ComClear( long ComId );

(VB) Sub IT4xxx\_ComClear (ByVal ComId As Long)

#### 【引 数】

引 数	I/O	説 明
ComId	I	通信ポート ID (IT4xxx_PortOpen の戻り値)

#### 【戻り値】

なし

#### 【機能詳細】

指定したポートのデータをクリアします。

データが途切れる(20ms以上のインターバル)まで、クリアを続けます。

#### 【注意事項】

イメージデータなどの場合、クリアには受信と同程度の時間がかかります。

#### 【コーディング例】

(VC) IT4xxx\_ComClear(ComId);

(VB) IT4xxx\_ComClear ComId





## ウェルコムデザイン株式会社

本社 〒651-2242  
神戸市西区井吹台東町1丁目1番1 西神南センタービル5F  
Phone. 078-993-6010(代) Fax. 078-993-6020

東京営業所 〒110-0005  
東京都台東区上野1-2-5 黒門町ビル2F  
Phone. 03-3836-8411(代) Fax. 03-3836-9412