

diBar2D

WelHwComDLL

仕様書

改定記録		
改訂番号	改定日	内容
Rev.1.00	Aug.2016	初版

- 1.本書の内容に関しては、将来予告無しに変更することがあります。
- 2.本取扱説明書の全部又は一部を無断で複製することはできません。
- 3.本書内に記載されている製品名等の固有名詞は各社の商標又は登録商標です。
- 4.本書内において、万一誤り、記載漏れなどお気付きのことがありましたらご連絡ください。
- 5.運用した結果の影響について、責任を一切負いかねます。

目次

1. はじめに	4
1.1. 機能概要	5
1.2. 動作環境	5
1.3. 開発環境	5
1.4. 提供ファイル	5
1.5. 使用ライセンスと著作権について	5
2. 処理概要	6
2.1. 処理開始～終了の流れ	6
2.2. 送信処理	6
2.3. 受信処理	7
2.4. 画像データの受信について	7
3. 関数一覧	8
3.1. 関数詳細説明	9
3.1.1. ポートオープン	10
3.1.2. ポートクローズ	12
3.1.3. タイムアウト設定	13
3.1.4. コマンド送信	15
3.1.5. データ送信	16
3.1.6. 受信通知スレッド開始	17
3.1.7. 受信通知スレッド終了	19
3.1.8. 受信バイト数取得	20
3.1.9. バーコードデータ受信	21
3.1.10. コマンドレスポンス受信	23
3.1.11. 画像データ受信	25
3.1.12. 画像データ受信状態	27
4. 付録	28
4.1. 定数一覧	28

1.はじめに

この仕様書は、超小型固定式エリアイメージャ cubeQR、超小型エリアイメージャ slimQR シリーズ専用ライブラリ WelHwComDll (以下、本 DLL) の仕様について記載したものです。cubeQR、slimQR 本体の操作説明は『二次元コードリーダ ユーザーズガイド (対応機種 slimQR Series、cubeQR Series)』を参照してください。

1.1. 機能概要

本 DLL には、以下の機能があります。

- 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 8/8.1/10

1.3. 開発環境

動作確認済み開発環境: Visual Studio 2008

動作確認済み開発言語: Visual Basic.NET (以下、VB.NET)、Visual C++ (以下、VC)

1.4. 提供ファイル

【 本 DLL 本体 】

WelHwCom.dll : slimQR、cubeQR 専用ライブラリ

【 インクルードファイル (VC) 】

WelHwCom.lib : インポートライブラリ (明示的リンクの場合は不要)
WelHwCom.h : インクルードファイル

【 インクルードファイル (VB.NET) 】

WelHwComDll.vb : 本 DLL 関数を VB.NET で使用する為の標準モジュール

1.5. 使用ライセンスと著作権について

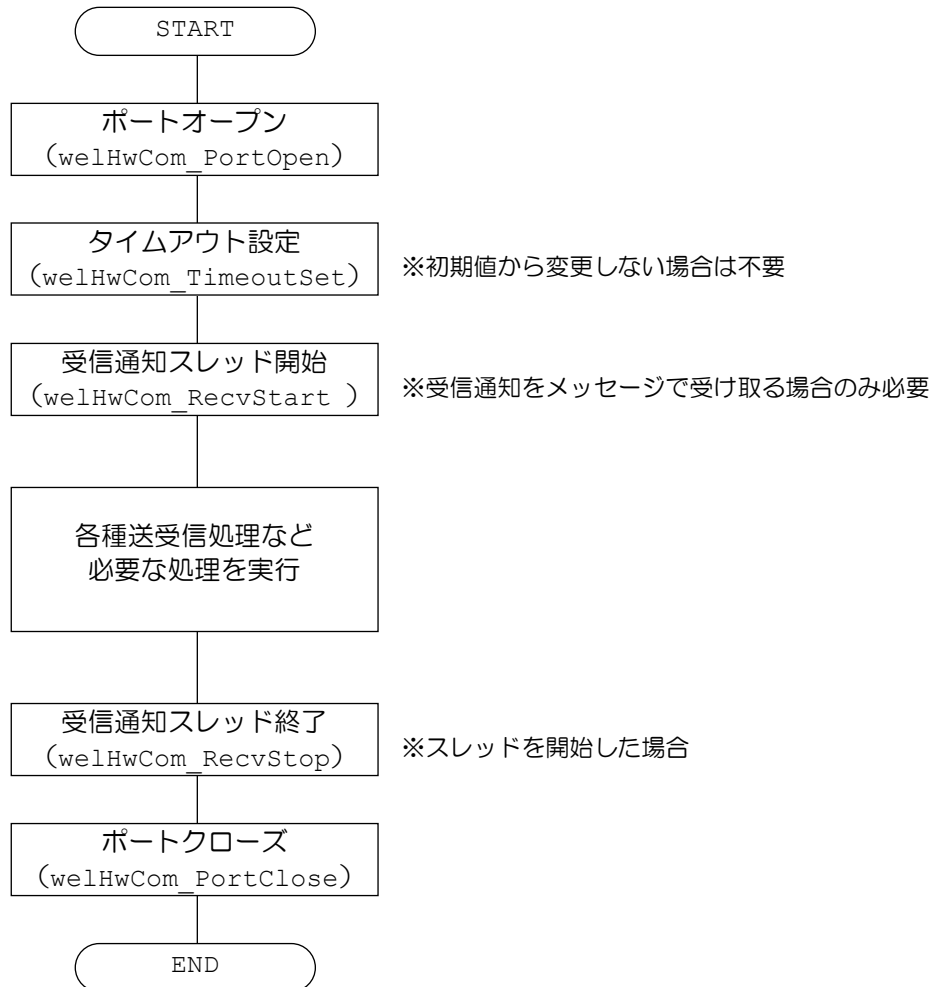
- 本 DLL は、ウェルコムデザイン株式会社 (以下、弊社) から購入したイメージャ (slimQR、cubeQR) と PC 間の通信用途で使用する場合に限り、ライセンス料なしでご利用頂けます。
- 本 DLL 及び付属する全てのファイル・ドキュメント等の著作権は、弊社に属します。
- 上記使用目的以外で、全部または一部を弊社に無断で複製する事は出来ません。

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～終了の流れ

処理開始から終了までの流れは以下の通りです。



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 ・・・ WelHwCom_CmndSend
データの送信 ・・・ WelHwCom_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。

WelHwCom_DataRecv	・・・	バーコードデータを受信します。
WelHwCom_ResponseRecv	・・・	コマンドレスポンスを受信します。
WelHwCom_ImageRecv	・・・	画像データを受信します。

読取りデータを受信する場合は、WelHwCom_DataRecv を使用します。

IMGSHP コマンド送信後は、WelHwCom_ImageRecv を、それ以外のコマンド送信後は WelHwCom_ResponseRecv を使用します。

シリアルトリガコマンドはレスポンスがありません。

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、WELHWCORE_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(WelHwCom_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(WelHwCom_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.画像データの受信について

イメージングコマンドには、次の2種類があります。

IMGSNP コマンド	・・・	画像の撮影。撮影したデータはイメージャのメモリに保持されます。
IMGSHP コマンド	・・・	IMGSNP 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。

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

イメージャは IMGSHP コマンドを受け付けると、受信が終了するまで、画像データを送信し続けます。送信される画像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、画像データ受信関数(WelHwCom_ImageRecv)を実行してください。

3. 関数一覧

本 DLL の関数一覧を以下に示します。

No	機 能	関数名 (VC)	関数名 (VB.NET)
1	ポートオープン	WelHwCom_PortOpen	WelHwCom_PortOpen
2	ポートクローズ	WelHwCom_PortClose	WelHwCom_PortClose
3	タイムアウト設定	WelHwCom_TimeoutSet	WelHwCom_TimeoutSet
4	コマンド送信	WelHwCom_CmndSend	vbWelHwCom_CmndSend
5	データ送信	WelHwCom_DataSend	vbWelHwCom_DataSend
6	受信通知スレッド開始	WelHwCom_RecvStart	WelHwCom_RecvStart
7	受信通知スレッド終了	WelHwCom_RecvStop	WelHwCom_RecvStop
8	受信バイト数取得	WelHwCom_RecvChk	WelHwCom_RecvChk
9	バーコードデータ受信	WelHwCom_DataRecv	vbWelHwCom_DataRecv
10	コマンドレスポンス受信	WelHwCom_ResponseRecv	vbWelHwCom_ResponseRecv
11	画像データ受信	WelHwCom_ImageRecv	vbWelHwCom_ImageRecv
12	画像データ受信状態取得	WelHwCom_ImageProgress	WelHwCom_ImageProgress
13	通信の中止	WelHwCom_ComCancel	WelHwCom_ComCancel
14	ポートのクリア	WelHwCom_ComClear	WelHwCom_ComClear

3.1.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。

3.1.n.機能名称

(VC) VC の呼び出し形式

(VB.NET) VB の呼び出し形式

【 機能詳細 】

関数の機能詳細について説明します。

【 引 数 】

関数で使用する引数について説明します。

(例)

引 数	I/O	説 明
Port	I	ポート番号

引数の意味、及び
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
I/O - DLL 側で値を参照し、
値を設定します

【 戻り値 】

戻り値について説明します。

(例)

定 義	意 味
WELHWCOM_NORMAL_END	正常終了

戻り値の見方・意味に
ついて説明しています。

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

【 注意事項 】

関数を使用する際の注意事項を記載します。

3.1.1.ポートオープン

```
(VC) long WelHwCom_PortOpen(long Port, long Baud, long DataBit, long StopBit,
                             long Parity, long FlowCtrl);
```

```
(VB.NET) Function WelHwCom_PortOpen (ByVal Port As Integer, ByVal baud As Integer,
                                       ByVal DataBit As Integer, ByVal StopBit As Integer,
                                       ByVal parity As Integer, ByVal FlowCtrl As Integer) As Integer
```

【機能詳細】

指定された設定で、ポートをオープンします。オープン後、通信状態をチェックするために、イメージャ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。エラーの場合、0 未満の値が返ります。

【引数】

引 数	I/O	説 明
Port	I	ポート番号 (1~256)
Baud	I	ボーレート WELHWCORM_BAUD_300 300bps WELHWCORM_BAUD_600 600bps WELHWCORM_BAUD_1200 1200bps WELHWCORM_BAUD_2400 2400bps WELHWCORM_BAUD_4800 4800bps WELHWCORM_BAUD_9600 9600bps WELHWCORM_BAUD_19200 19200bps WELHWCORM_BAUD_38400 38400bps WELHWCORM_BAUD_57600 57600bps WELHWCORM_BAUD_115200 115200bps
DataBit	I	データビット WELHWCORM_DATABIT_7 7 ビット WELHWCORM_DATABIT_8 8 ビット
StopBit	I	ストップビット WELHWCORM_STOPBIT_1 1 ビット WELHWCORM_STOPBIT_2 2 ビット
Parity	I	パリティ WELHWCORM_PARITY_NONE なし WELHWCORM_PARITY_EVEN 奇数 WELHWCORM_PARITY_ODD 偶数
FlowCtrl	I	フロー制御 WELHWCORM_FLOW_NONE なし WELHWCORM_FLOW_XONXOFF XON/XOFF WELHWCORM_FLOW_RTSCCTS RTS/CTS

【 戻り値 】

定 義	意 味
0 以上	通信ポート ID (以降の関数で使用)
WELHWCOR_ERR_NOTOPEN	ポートオープンエラー
WELHWCOR_ERR_TIMEOUT?	タイムアウトエラー (コマンドに対する応答がない)
WELHWCOR_ERR_CANCEL?	受信キャンセル
WELHWCOR_ERR_PARAMETER	関数パラメータエラー
WELHWCOR_ERR_COM	通信エラー
WELHWCOR_ERR_FULL	受信バッファフル

【 コーディング例 】

(VC)

```
long ComId ;
if ((ComId=WelHwCom_PortOpen(5,WELHWCOR_BAUD_9600,WELHWCOR_DATABIT_8,
    WELHWCOR_STOPBIT_1,WELHWCOR_PARITY_NONE,WELHWCOR_FLOW_RTSCTS)) < 0)
    MessageBox(hDlg, "ポートがオープンできません", "VCTest", MB_OK);
```

(VB.NET)

```
Dim ComId As Integer
ComId = WelHwCom_PortOpen(5,WELHWCOR_BAUD_9600,WELHWCOR_DATABIT_8,
    WELHWCOR_STOPBIT_1, WELHWCOR_PARITY_NONE, WELHWCOR_FLOW_RTSCTS)
If ComId < 0 Then
    MessageBox.Show("ポートがオープンできません")
End If
```

3.1.2.ポートクローズ

```
(VC) long WelHwCom_PortClose(long ComId);
(VB.NET) Function WelHwCom_PortClose(ByVal ComId As Integer) As Integer
```

【 機能詳細 】

ポートをクローズします。

【 引 数 】

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

【 戻り値 】

定 義	意 味
WELHWCOM_NORMAL_END	正常終了
WELHWCOM_ERR_NOTOPEN	ポートはオープンされていない
WELHWCOM_ERR_PARAMETER	関数パラメータエラー
WELHWCOM_ERR_IMGRCV	イメージデータ処理中

【 注意事項 】

画像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。
 イメージャ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 (WelHwCom_PortClose(ComId) != WELHWCOM_NORMAL_END)
    return ;

(VB.NET)
If WelHwCom_PortClos(ComId) <> WELHWCOM_NORMAL_END Then
    Exit Sub
End If
```

3.1.3. タイムアウト設定

```
(VC) void WelHwCom_TimeoutSet(long ComId, long Wait, long Interval);
(VB.NET) Sub WelHwCom_TimeoutSet(ByVal ComId As Integer, ByVal Wait As Integer,
                                   ByVal Interval As Integer)
```

【 機能詳細 】

全受信関数 (WelHwCom_DataRecv、WelHwCom_ResponseRecv、WelHwCom_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 (WelHwCom_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。

【 引 数 】

引 数	I/O	説 明
ComId	I	通信ポート ID (WelHwCom_PortOpen の戻り値)
Wait	I	受信タイムアウト時間 (ms) 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。 0 を指定した場合は、タイムアウトなしとなります。 0 未満の数値を指定した場合は、初期値 (10000ms) となります。
Interval	I	インターバルタイムアウト時間 (ms) 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。 0 を指定した場合は、タイムアウトなしとなります。 0 未満の数値を指定した場合は、初期値 (300ms) となります。

【 戻り値 】

なし

【 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 = WelHwCom_PortOpen(5, WELHWCOT_BAUD_9600, WELHWCOT_DATABIT_8,
    WELHWCOT_STOPBIT_1, WELHWCOT_PARITY_NONE, WELHWCOT_FLOW_RTSCOT)) >= 0) {
    WelHwCom_TimeoutSet(ComId, -1, 100) ;
}
```

(VB.NET)

```
Dim ComId As Integer
ComId = WelHwCom_PortOpen(5, WELHWCOT_BAUD_9600, WELHWCOT_DATABIT_8,
    WELHWCOT_STOPBIT_1, WELHWCOT_PARITY_NONE, WELHWCOT_FLOW_RTSCOT)
If ComId >= 0 Then
    Call IT4xxx_TimeoutSet(ComId, -1, 100)
End If
```

3.1.4. コマンド送信

```
(VC) long WelHwCom_CmndSend(long ComId, unsigned char *Cmnd, long SendCnt);
(VB.NET) Function vbWelHwCom_CmndSend(ByVal ComId As Integer,
                                         ByVal Cmnd As String) As Integer
```

【 機能詳細 】

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。

Cmnd で指定された文字列の前に、プリフィックス (<SYN>M<CR>) を付加します。

また、Cmnd の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。

【 引 数 】

引 数	I/O	説 明
ComId	I	通信ポート ID (WelHwCom_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
WELHWCOR_NORMAL_END	正常終了
WELHWCOR_ERR_NOTOPEN	ポートはオープンされていない
WELHWCOR_ERR_TIMEOUT	タイムアウトエラー
WELHWCOR_ERR_PARAMETER	関数パラメータエラー
WELHWCOR_ERR_CANCEL	ユーザーによる中断

【 コーディング例 】

(VC)

```
ret = WelHwCom_CmndSend(ComId, senddata, strlen(senddata)) ;
if(ret != WELHWCOR_NORMAL_END)
    MessageBox(hDlg, "送信エラーが発生しました", NULL, MB_OK) ;
```

(VB.NET)

```
ret = vbWelHwCom_CmndSend(ComId, SerialCmnd.Text)
If ret <> WELHWCOR_NORMAL_END Then
    MessageBox.Show("送信エラーが発生しました")
End If
```

3.1.5. データ送信

```
(VC) long WelHwCom_DataSend(long ComId, unsigned char *WriteData, long WriteCnt);
(VB.NET) Function vbWelHwCom_DataSend(ByVal ComId As Integer,
                                         ByVal WriteData As String) As Integer
```

【 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。

【 引 数 】

引 数	I/O	説 明
ComId	I	通信ポート ID (WelHwCom_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
WELHWCOM_NORMAL_END	正常終了
WELHWCOM_ERR_NOTOPEN	ポートはオープンされていない
WELHWCOM_ERR_TIMEOUT	タイムアウトエラー
WELHWCOM_ERR_PARAMETER	関数パラメータエラー
WELHWCOM_ERR_CANCEL	ユーザーによる中断

【 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、先頭にプリフィックス (<SYN>M<CR>) が、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 = WelHwCom_DataSend(ComId, "¥x16T¥x0D", 3) ;

(VB.NET)
Dim trcmd As String = Chr(&H16) + "T" + Chr(&HD)
Call vbWelHwCom_DataSend(ComId, trcmd)
```


3.1.6. 受信通知スレッド開始

```
(VC) long WelHwCom_RecvStart(long ComId, HWND Wnd, long MsgNo);
(VB.NET) Function WelHwCom_RecvStart(ByVal ComId As Integer, ByVal Wnd As Integer,
                                     ByVal MsgNo As Integer) As Integer
```

【 機能詳細 】

受信通知スレッドを開始します。

【 引 数 】

引 数	I/O	説 明
ComId	I	通信ポート ID (WelHwCom_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
WELHWCOM_NORMAL_END	スレッドは正常に起動された
WELHWCOM_ERR_NOTOPEN	ポートはオープンされていない
WELHWCOM_ERR_PARAMETER	関数パラメータエラー
WELHWCOM_ERR_OTHER	スレッドの作成・開始に失敗した

【 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(WelHwCom_RecvStart( ComId, hDlg, RECVMSG ) != WELHWCOR_NORMAL_END){
    MessageBox (hDlg, "スレッドの作成・開始に失敗しました", "VCTest", MB_OK );
    WelHwCom_PortClose(ComId) ;
}
```

(VB.NET)

```
Private Const WM_LBUTTONDBLCLK As Integer = &H203
If (WelHwCom_RecvStart(ComId,SerialCmnd.hWnd,WM_LBUTTONDBLCLK)
    <> WELHWCOR_NORMAL_END) Then
    MessageBox.Show("スレッドの作成・開始に失敗しました")
    Call WelHwCom_PortClose(ComId)
End If
```

3.1.7. 受信通知スレッド終了

```
(VC) void WelHwCom_RecvStop(long ComId);
(VB.NET) Sub WelHwCom_RecvStop(ByVal ComId As Integer)
```

【 機能詳細 】

受信通知スレッドを終了します。

【 引 数 】

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

【 戻り値 】

なし。

【 コーディング例 】

```
(VC)
WelHwCom_RecvStop(ComId) ;

(VB.NET)
Call WelHwCom_RecvStop(ComId)
```

3.1.8.受信バイト数取得

(VC) `long WelHwCom_RecvChk(long ComId);`

(VB.NET) `Function WelHwCom_RecvChk(ByVal ComId As Integer) As Integer`

【 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。

【 引 数 】

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

【 戻り値 】

指定された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

【 コーディング例 】

(VC)

```
unsigned char readbuf[1024];
```

```
long recvsize ;
```

```
if (WelHwCom_RecvChk(ComId) > 0)
```

```
    recvsize = WelHwCom_DataRecv(ComId, readbuf, sizeof(readbuf), NULL, NULL)
```

(VB.NET)

```
Dim readbuf As String
```

```
Dim recvsize As Integer
```

```
If (WelHwCom_RecvChk(ComId) > 0) Then
```

```
    recvsize = WelHwCom_DataRecv(ComId, readbuf, 1024, "", "")
```

```
End If
```

3.1.9.バーコードデータ受信

```
(VC) long WelHwCom_DataRecv(long ComId, unsigned char *RcvBuffer, long BufSize,
    unsigned char *PreFix, unsigned char *PostFix) ;
```

```
(VB.NET) Function vbWelHwCom_DataRecv(ByVal ComId As Integer,
    ByRef RcvBuffer As String, ByVal BufSize As Integer,
    ByVal PreFix As String, ByVal PostFix As String) As Integer
```

【 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されます。

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は、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コードは付加されません。

【 引 数 】

引 数	I/O	説 明
ComId	I	通信ポート ID (WelHwCom_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)
WELHWCOR_ERR_NOTOPEN	ポートはオープンされていない
WELHWCOR_ERR_COM	通信エラー
WELHWCOR_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)
WELHWCOR_ERR_FULL	受信バッファフル
WELHWCOR_ERR_CANCEL	ユーザーによる中断
WELHWCOR_ERR_PARAMETER	関数パラメータエラー

【 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は、タイムアウトエラー (WELHWCOR_TIMEOUT) となります。

タイムアウト時間の指定方法は、WelHwCom_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 = WelHwCom_DataRecv(ComId, recvdata, RECVBUFSIZE, NULL, NULL) ;
```

(VB.NET)

```
Dim readbuf As String
```

```
Dim recvsize As Integer
```

```
recvsize = WelHwCom_DataRecv(ComId, readbuf, 1024, "", "")
```

3.1.10. コマンドレスポンス受信

```
(VC) long WelHwCom_ResponseRecv(long ComId, unsigned char *RcvBuffer,
                                long BufSize) ;
(VB.NET) Function vbWelHwCom_ResponseRecv(ByVal ComId As Integer,
                                           ByRef RcvBuffer As String, ByVal BufSize As Integer) As Integer
```

【 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。

【 引 数 】

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

【 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 (レスポンス) バイト数 (正常終了)
WELHWCOM_ERR_NOTOPEN	ポートはオープンされていない
WELHWCOM_ERR_COM	通信エラー
WELHWCOM_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)
WELHWCOM_ERR_FULL	受信バッファフル
WELHWCOM_ERR_CANCEL	ユーザーによる中断
WELHWCOM_ERR_PARAMETER	関数パラメータエラー

【 注意事項 】

イメージ受信コマンド (IMGSHF) 以外のコマンドを送信した後に、コールしてください。

IMGSHF 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。

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は、タイムアウトエラー (WELHWCOM_ERR_TIMEOUT) となります。受信タイムアウト時間の指定方法は、WelHwCom_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 = WelHwCom_ResponseRecv(ComId, recvdata, RECVBUFSIZE) ;
```

(VB.NET)

```
Dim readbuf As String  
Dim recvsize As Integer  
recvsize = WelHwCom_ResponseRecv(ComId, readbuf, 1024)
```


3.1.11.画像データ受信

```
(VC) long WelHwCom_ImageRecv(long ComId, unsigned char *Filename,
                             unsigned char *RcvBuffer, long BufSize, long Protocol) ;
(VB.NET) Function vbWelHwCom_ImageRecv(ByVal ComId As Integer,
                                       ByVal Filename As String, ByRef RcvBuffer As String,
                                       ByVal BufSize As Integer, ByVal Protocol As Integer) As Integer
```

【 機能詳細 】

画像データを受信し、指定されたファイル名で保存します。画像を受信した後、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。

【 引 数 】

引 数	I/O	説 明
ComId	I	通信ポート ID(WelHwCom_PortOpen の戻り値)
Filename	I	保存画像ファイル名
RcvBuffer	O	受信データ(レスポンス) 格納領域
BufSize	I	受信データ(レスポンス) 格納領域のバイト数
Protocol	I	受信プロトコル WELHWCOM_PROTOCOL_2P を指定してください。 上記以外のプロトコルは指定できません。

【 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)
WELHWCOM_ERR_NOTOPEN	ポートはオープンされていない
WELHWCOM_ERR_COM	通信エラー
WELHWCOM_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)
WELHWCOM_ERR_FILE	ファイルアクセスエラー
WELHWCOM_ERR_FULL	受信バッファフル
WELHWCOM_ERR_CANCEL	ユーザーによる中断
WELHWCOM_ERR_PARAMETER	関数パラメータエラー

【 注意事項 】

WELHWCOM_PROTOCOL_2P(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。IMGSHF 送信時には「2P」を指定してください。

MGSHF コマンドを送信した後は、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は、タイムアウトエラー(WELHWCOM_TIMEOUT)となります。受信タイムアウト時間の指定方法は、WelHwCom_TimeoutSet を参照してください。

画像受信中に WelHwCom_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。

画像受信中に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 = WelHwCom_ImageRecv(ComId, "RecvImage.jpg", recvdata,
                             RECVBUFSIZE , WELHWCOT_PROTOCOL_2P) ;
```

(VB.NET)

```
Dim readbuf As String
Dim recvsize As Integer
recvsize = vbWelHwCom_ImageRecv(ComId, "RecvImage.jpg", readbuf,
                                1024, WELHWCOT_PROTOCOL_2P)
```

3.1.12.画像データ受信状態

```
(VC) long WelHwCom_ImageProgress(long ComId, long *ImgSize, long *ImgRecv) ;
(VB.NET) Function WelHwCom_ImageProgress(ByVal ComId As Integer,
      ByRef ImgSize As Integer, ByRef ImgRecv As Integer) As Integer
```

【 機能詳細 】

イメージデータの受信状況を取得します。

【 引 数 】

引 数	I/O	説 明
ComId	I	通信ポート ID (WelHwCom_PortOpen の戻り値)
RcvBuffer	O	画像データのサイズ(バイト)
BufSize	O	受信済みバイト数

【 戻り値 】

定 義	意 味
WELHWCOM_ERR_NOTOPEN	ポートはオープンされていない
WELHWCOM_IMG_NOTEXECUTE	実行していない
WELHWCOM_IMG_RECEIVING	画像データ受信中
WELHWCOM_IMG_BUFFERCLEAR	受信バッファクリア中

【 注意事項 】

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

WELHWCOM_ERR_NOTOPEN	・・・ 0
WELHWCOM_IMG_NOTEXECUTE	・・・ 前回実行した画像データの情報
WELHWCOM_IMG_RECEIVING	・・・ 現在の受信状況
WELHWCOM_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 = WelHwCom_ImageProgress(ComId, &filesize, &recvsize) ;
```

```
(VB.NET)
Dim ret As Integer
Dim filesize As Integer
Dim recvsize As Integer
recvsize = WelHwCom_ImageProgress(ComId, filesize, recvsize)
```

4. 付録

4.1. 定数一覧

本 DLL で使用する定数の一覧です。

定 義	値	説 明
WELHWCOT_NORMAL_END	0	正常終了
WELHWCOT_ERR_NOTOPEN	-1	ポートはオープンされていない
WELHWCOT_ERR_FILE	-2	ファイルアクセスエラー
WELHWCOT_ERR_TIMEOUT	-3	送信中
WELHWCOT_ERR_CANCEL	-4	キャンセルされた
WELHWCOT_ERR_PARAMETER	-5	実行されていない
WELHWCOT_ERR_COM	-6	通信エラー
WELHWCOT_ERR_FULL	-7	受信 (送信) バッファフル
WELHWCOT_ERR_IMGRCV	-8	画像データ受信中
WELHWCOT_ERR_OTHER	-99	その他通信エラー
WELHWCOT_IMG_NOTEXECUTE	0	実行されていない
WELHWCOT_IMG_RECEIVING	1	画像データ受信中
WELHWCOT_IMG_BUFFCLEAR	2	受信バッファクリア中
WELHWCOT_BAUD_300	0	ボーレート：300bps
WELHWCOT_BAUD_600	1	ボーレート：600bps
WELHWCOT_BAUD_1200	2	ボーレート：1200bps
WELHWCOT_BAUD_2400	3	ボーレート：2400bps
WELHWCOT_BAUD_4800	4	ボーレート：4800bps
WELHWCOT_BAUD_9600	5	ボーレート：9600bps
WELHWCOT_BAUD_19200	6	ボーレート：19200bps
WELHWCOT_BAUD_38400	7	ボーレート：38400bps
WELHWCOT_BAUD_57600	8	ボーレート：57600bps
WELHWCOT_BAUD_115200	9	ボーレート：115200bps
WELHWCOT_PARITY_NONE	0	パリティ：なし
WELHWCOT_PARITY_EVEN	1	パリティ：偶数
WELHWCOT_PARITY_ODD	2	パリティ：奇数
WELHWCOT_STOPBIT_1	0	ストップビット：1
WELHWCOT_STOPBIT_2	1	ストップビット：2
WELHWCOT_DATABIT_7	0	データビット：7
WELHWCOT_DATABIT_8	1	データビット：8
WELHWCOT_FLOW_NONE	0	フロー制御：なし
WELHWCOT_FLOW_XONXOFF	1	フロー制御：XON/OFF
WELHWCOT_FLOW_RTSCS	2	フロー制御：RTS/CTS
WELHWCOT_PROTOCOL_2P	2	送信プロトコル：USB



ウェルコムデザイン株式会社

U R L : www.e-welcom.com
e-mail: welcom@e-welcom.com

本 社 〒651-2242 神戸市西区井吹台東町1-1-1 西神南センタービル
TEL. 078-993-6010 (代) FAX. 078-993-6020
東 京 〒113-0034 東京都文京区湯島 3-14-9 湯島ビル
TEL. 03-3836-9411 (代) FAX. 03-3836-9412