BARTEC

BT_LF
(Bartec Low Frequency DLL)

for Pocket PC

Version 20.01.2009

List of contents

General information

Language: Visual Studio 2005, VB.NET

Actual firmware for the Bartec-Board:

- btrw-hdx.v1.40.frm for everything TIRIS, UNIQUE32/64, ZOODIAC
- btrw-rw.v1.40.frm for Q5, HITAG-S(1), HITAG-II, EM4305
- btrw-ti.v1.40.frm for TITAN, UNIQUE32/64, ZOODIAC

Supported transponder types:

0 - EM 41xx (UNIQUE) - RO

1 - HITAG S - RW

2 - FDX-B ISO 11784/5-RO

3 - Q5 / 5557 -- RW

4 - HDX Transponder (TI)-RW

5 - HITAG 2 - RW

6 - EM 4305 - RW

7 - EM4450/4550 (TITAN)-RW

1.0 Property

1.1 Comport

Property **Comport**() As Integer

Summary:
Calls or stipulates the connection for the communication, including all available COM connections.

Parameter:
Comport no. as numerical value. 1 = Comport 1, 2 = Comport 2, etc.

1.2 Active

Property **Active**() As Boolean

Summary:
Opens a new serial connection.

Parameter:
True, in order to activate the interface, otherwise false. The standard value is false.

1.3 DTR

Property **DTR**() As Boolean

Summary:
Calls or stipulates a value, which activates the Data Terminal Ready (DTR) Signal during a serial communication.

Return values:
True, in order to activate Data Terminal Ready (DTR), otherwise false. The standard value is false.

1.4 RTS

Property **RTS**() As Boolean

Summary:
Calls or stipulates a value, which states whether the Request to Send (RTS) signal is activated during the serial communication.

Return values:
True, in order to activate Request to Transmit (RTS), otherwise false. The standard value is false.

1.5 TimeOut

Property **TimeOut**() As Integer

Summary:
Input of the delay value for the timeout of the reading/writing process. The standard value is 2000 milliseconds

Parameter:
Numerical value in milliseconds

1.6 ShutDownActive

Property **ShutDownActive**() As Boolean

Summary:
Activates the event "ReadyToShutDown". It is triggered off as soon as the Pocket PC has been connected to the Docking Station.

Parameter:
True, if the ShutDown is to be activated.

1.7 CDInfo

ReadOnly Property **CDInfo**() As Boolean

Summary:
Calls the status of the line for identifying the carrier for the connection.

Return values:
True, if the carrier is identified, otherwise false.

1.8 CTSInfo

ReadOnly Property **CTSInfo**() As Boolean

Summary:
Calls the status of the Clear-to-Send line.

Return values:
True, if the Clear-to-Send line is identified, otherwise false.

1.9 DSRInfo

ReadOnly Property **DSRInfo**() As Boolean

Summary:
Calls the status of the DSR (Data Set Ready)-Signal.

Return values:
True, if a Data Set Ready-Signal was sent to the connection, otherwise false.

1.10 DLLVersion

ReadOnly Property **DLLVersion**() As String

Summary:
Calls the version number of the DLL.

Return value:
Version number of the DLL

2. 0 Variables

2.1 Hitag2Password

Public **Hitag2Password** As String

Summary:
Parameter for the Hitag2Password, which is required for reading/writing.

Parameter:
8 characters hexadecimal

2.2 IsTagSelect

Public **IsTagSelect** As Boolean

Summary:
Calls the status of the selection of a transponder. This parameter is set automatically if the function SelectTag is carried out.

Parameter:
True, if a transponder is selected, false if not.

2.3 TitanPassword

Public **TitanPassword** As String

Summary:
Parameter for the Hitag2Password, which is required for reading/writing.

Parameter:
8 characters hexadecimal

3.0 Function

3.1 ASCIIToHex

Function **ASCIIToHex**(ByVal *Input* As **String**) As **String**

Summary:
Converts ASCII-characters into hexadecimal characters.

Parameter:
Input as ASCII-characters

Return values:
Hexadecimal characters

3.2 HexToASCII

Function **HexToASCII**(ByVal *Input* As **String**) As **String**

Summary:
Converts hexadecimal characters into ASCII characters.

Parameter:
Input as hexadecimal characters

Return values:
ASCII-characters as String

3.3 ChangeHitag2Password

Function **ChangeHitag2Password**(ByVal *OldPassword* As **String**, ByVal *NewPassword* As **String**) As **String**

Summary:
Change of the Hitag2 password. The Hitag2 is selected with the old password. After the selection the new password is written in the password block. The successful writing is confirmed as a return value with a "+" sign. A "- " sign is returned if it is unsuccessful.

Parameter:
OldPassword as 8 characters hex. e.g: "00000000"

NewPassword as 8 characters hex. e.g.: "11111111"

Return values:
"+" sign as OK

"-" sign as error

Example:

```
Private Sub ChangePasswordHitag2()
          Dim Res As String
          Res = BTLF_DLL.ChangeHitag2Password(“00000000”,”11111111”)
    If Res Like "*+*" Then
      MsgBox("OK")
    ElseIf Res Like "*- 4*" Then
        MsgBox("No Transponder selected")
    Else
        MsgBox("[" & Res & "]Error")
    End If
End Sub
```

3.4 ChangeTitanPassword

Function **ChangeTitanPassword**(ByVal *OldPassword* As **String**, ByVal *NewPassword* As **String**) As **String**

Summary:

Change of the Titan Password. The Titan is selected with the old password. After the selection the new password is written in the password block. The successful writing is confirmed as a return value with a “+“ sign. A “- “ sign is returned if it is unsuccessful.

Parameter:

OldPassword as 8 characters hex. e.g: “00000000“

NewPassword as 8 characters hex. e.g.: “11111111“

Return values:

“+“ characters as OK

“-“ characters as error

Example:

```
Private Sub ChangePasswordTitan()
          Dim Res As String
    Res = BTLF_DLL.ChangeTitanPassword(“00000000“,“11111111“)
    If Res Like "*+*" Then
      MsgBox("OK")
    ElseIf Res Like "*- 4*" Then
       MsgBox("No Transponder selected")
    Else
       MsgBox("[" & Res & "]Error")
    End If
End Sub
```

3.5 GetVersion

Function **GetVersion()** As String

Summary:
Calls the actual version number of the firmware from the reader.

Return values:
Actual firmware which is loaded in the Reader. e.g: BARTEC-btrw-rw.[bartec].v1.40

3.6 SelectTag

Function **SelectTag**(ByVal *TagType* As Integer) As String

Summary:
Selects a transponder. With the successful reading the UID of the transponder is handed over as 8 character hexadecimal. In case of an error an error code is returned. At the same time the parameter "IsTagSelect" is set.

The password „TitanPassword" must be set in advance with the Titan transponder.

Parameter:
TagType:

0 - EM 41xx (UNIQUE) -RO
1 - HITAG S - RW
2 - FDX-B ISO 11784/5-RO
3 - Q5 / 5557 -- RW
4 - HDX Transponder (TI)-RW
5 - HITAG 2 - RW
6 - EM 4305 - RW
7 - EM4450/4550 (TITAN)-RW

Return values:

UID
(as 8 character hexadecimal.)

"- 1" "- 2" "- 3" "- 4" "- 5" or "- 99" for general reader error
3 characters, minus sign + one space + one number (dependent on the transponder type)

„Error: 54"
Transponder type is not supported in the actual firmware.

Example: Selection of a Hitag S Transponder

```
Private Sub SelectTransponder()
        Dim UIDData As String
        If TagType = 7 Then                                  // Only with Titan
        BTLF_DLL.TitanPassword = "00000000" // Only with Titan
          End If                                             // Only with Titan
        UIDData = BTLF_DLL.SelectTag(1)
        if (UIDData Like "*- 1*") Or (UIDData Like "*- 2*") Or (UIDData Like "*-  3*") Or (UIDData Like "*- 4*") Or
        (UIDData Like "*- 5*") Then
    MsgBox("[" & UIDData & " ] No Transponder selected")
    ElseIf UIDData = "Error: 54" Then
      MsgBox("[54] Not available in this firmware")
    ElseIf UIDData = "- 99" Then
      MsgBox("[- 99] Reader-Error")
    Else
      MsgBox("TAG-No. :" & UIDData)
    End If
End Sub
```

3.7 ReadTagData

Function **ReadTagData**(ByVal *TagType* As Integer, ByVal *Block* As String) As String

Summary:
Reading of data from a transponder. The transponder type and the block address have to be handed over as parameters. Beforehand the transponder has to be selected with the function „SelectTransponder“. The UID is issued when selecting a transponder and the parameter “IsTagSelect“ is set.

After the successful reading the read block is handed over as 8 character String as return value. Data cannot be read from every transponder.

Parameter:
TygTyp

Supported transponder types:

1 - HITAG S - RW

3 - Q5 / 5557 -- RW

4 -HDX Transponder (TI)-RW

5 - HITAG 2 - RW

6 - EM 4305 - RW

7 - EM4450/4550 (TITAN)-RW

Block as address

Return values:

8 characters hexadecimal with successful reading.

"- 1" or "- 2" for errors

"- 1 " transponder has not been selected

"- 2 " error during reading

Example: Reading of data from a Hitag S, Block 10.

```
Private Sub ReadData()
        Dim UIDData As String
    Dim TAGData As String
    UIDData = BTLF_DLL.SelectTag(1)
    If BTLF_DLL.IsTagSelect And (UIDData <> "") Then
        TAGData = BTLF_DLL.ReadTagData(1, "10")
      If (TAGData = "- 1") Then
        MsgBox("No Transponder select")
      ElseIf (TAGData = "- 2") Then
        MsgBox("Read Error")
      Else
        txtData.Text = fc.RFIDData
        MsgBox("Read OK")
      End If
    End If
End Sub
```

3.8 WriteTagData

Function **WriteTagData**(ByVal *TagType* As **Integer**, ByVal *Block* As **String**, ByVal *Data* As **String**) As **String**

Summary:

Writing of data on a transponder. The transponder type block address and the data have to be handed over as parameters. Beforehand the transponder has to be selected with the function „SelectTag". The UID is issued when selecting a transponder and the parameter "IsTagSelect" is set.

After the successful reading the read block is handed over as 8 character String as return value. Data cannot be written in every transponder.

Parameter:

TagTyp

Supported transponder types:

1 - HITAG S - RW

3 - Q5 / 5557 -- RW

4 -HDX Transponder (TI)-RW

5 - HITAG 2 - RW

6 - EM 4305 - RW

7 - EM4450/4550 (TITAN)-RW

Block: as address

Data: 8 characters hexadecimal

Return values:

8 characters hexadecimal with successful reading.
"- 1" "- 2" "-3" for errors
"- 1 " transponder has not been selected
"- 2 " error during reading
"- 3 " false block length

Example: Writing of data in a Hitag S, Block 10 with 8 hexadecimal characters "11111111"

```
Private Sub WriteData()
        Dim UIDData As String
    Dim Result As String
    UIDData = BTLF_DLL.SelectTag(1)
    If BTLF_DLL.IsTagSelect And (UIDData <> "") Then
        Result = BTLF_DLL.WriteTagData(1, "10", "11111111")
      If (Result = "- 1") Then
        MsgBox("No Transponder select")
      ElseIf (Result = "- 2") Then
        MsgBox("Write Error")
            ElseIf (Result = "- 3") Then
              MsgBox("Block length Error")
            ElseIf (Result = "+") Then
        MsgBox("Write OK")
      End If
    End If
End Sub
```

3.9 SendDataToReader

Function **SendDataToReader**(ByVal *Command* As **String**) As **String**

Summary:

Additional *Low Level* function for transmitting commands to the reader. The reader will react to the commands immediately and return the results immediately.

Parameter:

Command as String

Return values:

ASCII and hexadecimal characters

Example: Query of the firmware version in the reader

```
Private Sub SendData()
    Dim Result As String
    Result = BTLF_DLL.SendDataToReader("sv")
    If Result <> "" Then
      MsgBox("Version" & Result)
    End If
End Sub
```

4.0 Event

4.1 ReadyToShutDown

Public Event **ReadyToShutDown**(ByVal *Sender* As **Object**, ByVal *ReaderDown* As **Boolean**)

Summary:
The event is triggered off when “ShutDownActive” has been activated and the Pocket PC has been connected to the Docking Station. Through this event you have the possibility e.g to end the software with the deactivation of the reader.

Parameter:
ReaderDown = True, if the electricity supply for the reader has been interrupted

4.2 TLMData

public Event **TLMData**(ByVal *Sender* As **Object**, ByVal *Direction* As **String**, ByVal *Data* As **String**)

Summary:
The event is always triggered off if data is transferred between the reader and the application.

Parameter:
Direction:
“>“ = Data which are sent to the reader.
“<“ = Data which come from the reader.
Data:
Data from the data transfer.

5.0 Errorcode

5.1 SelectTransponder

- 1 No Transponder selected
- 2 No Transponder selected
- 3 No Transponder selected
- 4 No Transponder selected
- 5 No Transponder selected

Error: 54 Not available in this firmware
- 99 Reader Error

5.2 ReadData

- 1 No Transponder selected
- 2 Read Error

5.3 WriteData

- 1 No Transponder selected
- 2 Read Error
- 3 Block length Error