

Using Hart in Excel

Software Documentation

Walter Borst, April 2010

Getting Started

The worksheet which is used is very simple.

	A	B	C	D	E
1	Test	Day	Month	Year	ComPort
2					2
3	Old				
4	New	5	3	2010	
5	Stored				
6					

Double click the file UsingHartDLL.xls in the example path .\Examples\HartDLL\Excel. Excel opens and appears with a button in the upper left corner. Enter the com port number of the port you have connected a device to. Press the button and the Visual Basic Editor will appear because the program was stopped at a 'breakpoint'.

```
'Connect to device with address 0
hSrv = BHDrv_ConnectByAddr(hDrv, 0, DRV_WAIT, 2)
If hSrv <> INVALID_SRV_HANDLE _
Then
    BHDrv_FetchConnection hSrv, strConnection
    If strConnection.byError = SRV_SUCCESSFUL _
    Then
        'Read tag, descriptor and date
        hSrv = BHDrv_DoCommand
            (hDrv, 13, DRV_WAIT, byReqData(0), 0, 0, ByVal strConnection.sUniqueID)
        If hSrv <> INVALID_SRV_HANDLE _
        Then
            BHDrv_FetchConfirmation hSrv, strConfirmation
            If strConfirmation.byError = SRV_SUCCESSFUL _
            Then
                Range("B3") = Format(BHDrv_PickInt8(18, ByVal strConfirmation.sData), "0")
                Range("C3") = Format(BHDrv_PickInt8(19, ByVal strConfirmation.sData), "0")
                Range("D3") = Format(BHDrv_PickInt8(20, ByVal strConfirmation.sData) + 1900, "0")
                Stop
```

Code Snippet 1: Getting Started



<http://borst-automation.com>
info@borst-automation.de

Download: <http://borst-automation.com/downloads/UsingHartInExcel.zip>

Hart® is a registered trademark of the Hart Communication Foundation
 Windows® is a registered trademark of Microsoft Corporation

Coding Details



While the module HartTest is containing the little test program the module HartInterface contains the necessary structures and functions declarations. The following is an example of the declaration

of one of the functions in the DLL.

```
Public Declare Function BHDrv_DoCommand Lib "BaHartDrv70.dll" _
  (ByVal hDrv As Long, _
  ByVal byCmd As Byte, _
  ByVal byQOS As Byte, _
  pstrRequestData As Any, _
  ByVal byReqLen As Byte, _
  ByVal lAppKey As Long, _
  pstrUniqueID As Any _
  ) As Long
```

Code Snippet 2: Declaration of a HartDLL Function

Adjust Working Directory

The first sequence is used to adjust the current directory to the path with the DLL.

```
'Set working directory relative to xls path to
'allocate the directory with the DLL
sDir = ActiveWorkbook.Path
sDrive = Left$(sDir, 2)
ChDrive sDrive
ChDir sDir
ChDir "..\..\..\\"
```

Code Snippet 3: Setting the Working Directory

Set License Key

Then the license key is registered.

```
'Set the license information
BHDrv_ValidateLicense ByVal "12345678-ABCD-9012-DDDD-1234567TRIAL"
```

Code Snippet 4: Registering the License

Open Port

For opening the com port the number of the port is taken from the excel sheet.

```
'Open Com from Cell E2
'Configuration will be default
iComPort = Range("E2")
hDrv = BHDrv_OpenChannel(iComPort)
```

Code Snippet 5: Opening a Communication Channel

Connect

The connection is established to retrieve the unique identifier from the device and store it into the structure strConnection. The unique identifier is stored in the application program to allow also the operation with multiplexer devices.

```
'Connect to device with address 0
hSrv = BHDrv_ConnectByAddr(hDrv, 0, DRV_WAIT, 2)
If hSrv <> INVALID_SRV_HANDLE _
Then
  BHDrv_FetchConnection hSrv, strConnection
```

Code Snippet 6: Connecting to the Hart Device

Read Date

If the connection was O.K. command 13 is send to get tag, descriptor and date.

```
'Read tag descriptor date
hSrv = BHDrv_DoCommand(hDrv, 13, DRV_WAIT, byReqData(0), 0, 0, ByVal strConnection.sUniqueID)
If hSrv <> INVALID_SRV_HANDLE _
Then
  BHDrv_FetchConfirmation hSrv, strConfirmation
  If strConfirmation.byError = SRV_SUCCESSFUL _
  Then
    Range("B3") = Format(BHDrv_PickInt8(18, ByVal strConfirmation.sData), "0")
    Range("C3") = Format(BHDrv_PickInt8(19, ByVal strConfirmation.sData), "0")
    Range("D3") = Format(BHDrv_PickInt8(20, ByVal strConfirmation.sData) + 1900, "0")
  Stop
```

Code Snippet 7: Sending a Command and Retrieving the Confirmation

The data from the response frame is stored in cells B3, C3 and D3 of the worksheet.

Write Date

Prepare Request Frame

Firstly the response data is copied to the byte stream for the request.

```
'Copy the response to the request
For e = 0 To 17
  byReqData(e) = BHDrv_PickInt8(e, ByVal strConfirmation.sData)
Next e
```

Code Snippet 8: Copying Response Bytes

Then day, month and year are picked from the excel worksheet and are inserted into the request data byte array.

```
'Set day
byReqData(18) = Range("B4")
'Set month
byReqData(19) = Range("C4")
'Set Year
byReqData(20) = Range("D4") - 1900
```

Code Snippet 9: Insert New Date

Send Command 18

Finally command 18 is sent.

```
'Send command 18
hSrv = BHDrv_DoCommand(hDrv, 18, DRV_WAIT, byReqData(0), 21, 0, ByVal strConnection.sUniqueID)
```

Code Snippet 10: Perform Command 18

Read Back Date

As a last sequence the date is read back and stored in the cells B5, C5 and D5.

```
'Read back tag descriptor date
hSrv = BHDrv_DoCommand(hDrv, 13, DRV_WAIT, byReqData(0), 0, 0, ByVal strConnection.sUniqueID)
If hSrv <> INVALID_SRV_HANDLE _
Then
  BHDrv_FetchConfirmation hSrv, strConfirmation
  If strConfirmation.byError = SRV_SUCCESSFUL _
  Then
    Range("B5") = Format(BHDrv_PickInt8(18, ByVal strConfirmation.sData), "0")
    Range("C5") = Format(BHDrv_PickInt8(19, ByVal strConfirmation.sData), "0")
    Range("D5") = Format(BHDrv_PickInt8(20, ByVal strConfirmation.sData) + 1900, "0")
```

Code Snippet 11: Send Command 13 and Display the Result