TECH NOTE :: 3rd Party Device Integration into catmanAP or QuantumX Data Recorder CX22-W

Version: 2015-01-30
Author: Christof Salcher, HBM Germany
Status: public

Abstract
This Tech Note describes how to integrate a non HBM product / 3rd party device over RS232 into catmanAP and is based on an example using the scripting engine “EasyScript” which is part of the catmanAP package or the Data Recorder CX22 or CX22-W.

Detailed
EasyScript offers a scripting editor and engine. EasyScript is the programming language with which you can monitor and control catmanEasy/AP. With EasyScript you can extend catmanEasy/AP with your own functions. You can execute your own functions at certain times, e.g. before starting a DAQ job or you can evaluate the measured data and transfer the computed values to Excel. You can integrate complete programs or 3rd party devices. For this purpose you use auxiliary channels to be filled from the script running in a background task 5 – 10 times per second opening the specific port, writing and reading to that port according to the defined device protocol.

EasyScript is based on the VBA standard (Visual Basic for Applications), which is also used in the Office packages from Microsoft. VBA provides the general language elements: Variables, control flow, mathematics, etc. EasyScript makes the required objects available to you, enabling you to control the behaviour of catmanEasy/AP. Due to the open architecture of VBA however other programs, e.g. Excel or Word, can also be addressed as objects. With an EasyScript project written in VBA, you can either execute it completely like a normal program or you can also combine a collection of functions to form a script project and have each function executed by a different event which may be a certain point in time during a measurement or a keystroke or a click on a button.

The online help summarizes all the possibilities. Just start catman help and type in EasyScript.
CX22-W offers a RS232 interface which is rare in modern computers. This enables us to integrate 3rd party devices. Already fully integrated is a GPS sensor with its NMEA protocol.
Step by Step Workflow how to integrate a 3rd party device

1. Go to -> Additional Options -> Program functions -> Activate EasyScript

2. Start a “NEW DAQ project” or “Open DAQ project”
3. **Highlight** DAQ hardware you want to work with and **Connect**

![Device Manager](image1.png)

**Connection over Ethernet (TCP/IP)**

Select the modules with which you want to work in catmanEasy/AP in column “Connect”. Addresses of modules to be used in catmanEasy/AP may not be different from the computer address in places of the subnetmask having 255. Additionally you can change the name and flash the LED only on these modules.

For using modules with different addresses (Symbol “Connection not possible”) select the module and change the IP address.

**Connection over FireWire**

All FireWire modules found are automatically included in the DAQ project. Selection of singular modules is not possible.

4. **Go to screen DAQ channels -> Special -> New auxiliary channel**

![DAQ Channels](image2.png)
5. Create a name for the auxiliary channel, a physical unit, activate script and determine a channel for time difference calculation from drop down list, then **Edit script code**

![Image of EasyScript interface showing auxiliary channel setup](image)

6. **Edit Code and store it** (please find the sample code at the end of this TECH NOTE)

![Image of EasyScript code example](image)

```plaintext
Dim SampleCount As Integer
DimRetVal As Long
Dim CmdBuf As String
Dim NV As Double
Dim t As Integer

'Returns the number of samples to be delivered
EA_Daq.CurrentBlockSize(0,SampleCount)

ReDim NVBuf(SampleCount) As Double

'Your code to fill the sample array NVBuf goes here
RetVal = EA_Comm.RS232_OpenPort(COM1,9600,FARITY_EVEN,8,1,16000,0000)
If RetVal < 0 Then GoTo ErrHandler

CmdBuf$ = "Open"
RetVal = EA_Comm.RS232_WrtePort(COM1,CmdBuf$)
If RetVal < 0 Then GoTo ErrHandler

VarChar = "Start"
RetVal = EA_Comm.RS232_WrtePort(COM1,VarChar)
If RetVal < 0 Then GoTo ErrHandler

RetVal = EA_Comm.RS232_ReadPort(COM1,10,256,CmdBuf$)
If RetVal < 0 Then GoTo ErrHandler

RetVal = EA_Comm.RS232_ReadPort(COM1,10,256,RetVal$)
If RetVal < 0 Then GoTo ErrHandler

NV = Val(RvBuf$)
For i=1 To SampleCount
    NVBuf(i) = NV
Next

EA_Comm.RS232_ClosePort COM1
```
7. Start DAQ job

Sample Code RS232 Integration

Sub Main()
Dim SampleCount As Integer
Dim RetVal As Long
Dim RcvBuf As String
Dim Cmd As String
Dim MV As Double
Dim i As Integer

'Returns the number of samples to be delivered
EA_DAQ.CurrentBlockSize(0,SampleCount)
ReDim MVBuf(SampleCount) As Double

'Your code to fill the sample array MVBuf goes here
RetVal = EA_Comm.RS232_OpenPort(COM1,9600,PARITY_EVEN,8,1,16000,5000)
If RetVal < 0 Then GoTo ErrHandler
Cmd=Chr$(18)
RetVal = EA_Comm.RS232_WritePort (COM1,Cmd)
If RetVal < 0 Then GoTo ErrHandler
RetVal = EA_Comm.RS232_WritePort (COM1,"PCS1" & vbLf)
RetVal = EA_Comm.RS232_ReadPort (COM1,10,256, RcvBuf)
RetVal = EA_Comm.RS232_WritePort (COM1,"MSV?" & vbLf)
If RetVal < 0 Then GoTo ErrHandler
RetVal = EA_Comm.RS232_ReadPort (COM1,10,256, RcvBuf)
MV=Val(RcvBuf)
For i=1 To SampleCount
   MVBuf(i)=MV
Next
EA_Comm.RS232_ClosePort COM1

'******************************************************************************

'When done, pass the data to the auxiliary channel buffer
EA_DAQ.SetCurrentBlock 0, 1, SampleCount, MVBuf()

Exit Sub

ErrHandler:
MsgBox EA.LastErrorText

End Sub

Legal Disclaimer: TECH NOTEs are designed to provide a quick overview. TECH NOTEs are continuously improved and so change frequently. HBM assumes no liability for the correctness and/or completeness of the descriptions. We reserve the right to make changes to the features and/or the descriptions at any time without prior notice.