

# QuantumX-Treiber für DASyLab

---

## Inhalt

Inhalt.....	1
Systemvoraussetzungen .....	2
Dateien, die ins DASyLab-Verzeichnis kopiert werden .....	2
Besonderheiten zur Installation.....	3
Treiber-DLL in INI-Datei eintragen .....	3
Einträge in INI Datei .....	3
Freischaltung .....	3
QuantumX-Device-SN.....	4
QuantumX-SN .....	4
Debug-Flags .....	4
LogQuantumX .....	4
TraceDebug.....	4
TraceHBMApi .....	4
TraceDebugDuringAcquisition.....	4
TraceProcessData.....	4
Beispiel für die Einträge .....	5
Unterschiede zum alten Treiber .....	6
Technologie .....	6
Geräteauswahl, IP-Adresse, Port .....	6
Ethernet / Firewire.....	6
Auslesen von Informationen aus dem QuantumX .....	6
Abtastrate / Blockgröße .....	6
Kanalnamen .....	6
Lizensierung .....	6
Menu .....	6
Gerät gefunden, Modul-Einstellungen, CAN-Kanäle auswählen .....	7,8

## Systemvoraussetzungen

- Windows® XP, Vista™, 7 oder 8.x (32-Bit oder 64-Bit)
- DASyLab Version 13.  
Der Treiber wurde erfolgreich getestet mit DASyLab 10, 11 und 12.
- C++ Runtime VC2010 SP 1:  
Installieren via vcredist\_x86.exe Version 10.0.40291.1 (gleiche Version wie für den alten Treiber)
- QuantumX-Assistent 4.2 oder neuer zum Parametrieren der Messgeräte.  
Die QuantumX-Assistenten-Installation installiert auch die erforderlichen .NET-Frameworks.

## Dateien, die ins DASyLab-Verzeichnis kopiert werden müssen

Datei	Version	Datum
IMP40.dll	12.4	21.09.15
IMP40Gui.dll	12.4	21.09.15
IMP40.chm	12.3	26.08.15
HBM.QuantumX.dll	4.1.2.147	12.02.2015
HBM.DeviceComponents.dll	4.1.2.147	12.02.2015
HBM_QX_Framework.dll	4.1.1.18965	17.06.2014
HBM_Scan.dll	1.0.3.39	10.12.2014
HBM_Streaming.dll	1.2.0.71	10.12.2014
RegCOMAssembly32.bat	-	10.12.2013

Zur Deinstallation wird außerdem benötigt

Datei	Version	Datum
UnRegCOMAssembly32.bat	-	10.12.2013

## Besonderheiten zur Installation

Nach dem Kopieren der Dateien ins DASyLab-Programmverzeichnis müssen die ActiveX-Komponenten der QuantumX-API registriert werden. Dazu im DASyLab-Programmverzeichnis

# QuantumX-Treiber für DASyLab

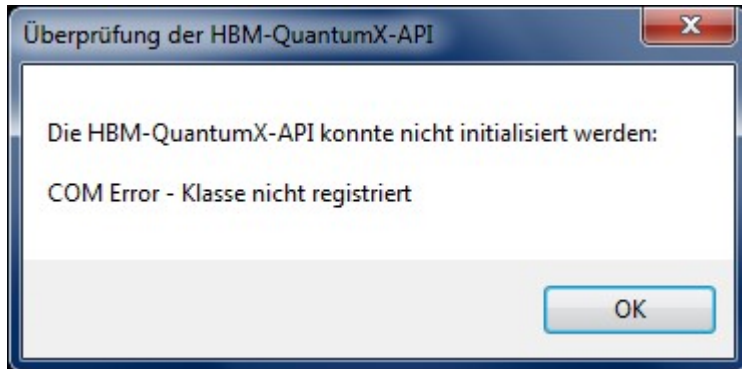
---

„RegCOMAssembly32.bat“

als Administrator – per Rechts-Klick - ausführen.

Dieser Aufruf registriert die COM-Server der HBM QuantumX-API.

Wenn diese nicht registriert sind, gibt der Treiber die Meldung



aus.

## Einträge in DASyLab.ini Datei:

Die DASyLab.ini finden Sie unter „Benutzer\Öffentlich\öffentliche Dokumente\DASyLab\13.0\ger“

### Treiber-DLL in INI-Datei eintragen

DASyLab durchsucht die Section [Extend] nach Einträgen für zusätzlich zu ladende DLLs. Einträge der Art „DLAB\_UX1.DLL“ bis „DLAB\_UX8.DLL“ sind Platzhaltereinträge.

Einer dieser Platzhaltereinträge muss durch „IMP40.DLL“ ersetzt werden.

Beispiel:

Der erste freie Eintrag sei zum Beispiel „DLL5=DLAB\_UX5.DLL“

Er muss abgeändert werden nach:

„DLL5=IMP40.DLL“

Der Treiber berücksichtigt Einträge in DASyLab.ini in Section [IMP40].

### Freischaltung

Die Lizenzierung bezieht sich auf ein QuantumX-Gerät.

# QuantumX-Treiber für DASYLab

---

## QuantumX-Device-SN

Seriennummer des QuantumX in Hex.

Beispiel:

QuantumX-Device-SN=9E5001CE1

## QuantumX-SN

Freischalt-Seriennummer des Treibers. Diese bekommen Sie mit Ihrer Bestellung ausgehändigt.

## Debug-Flags

Zu Debug-Zwecken können vom Treiber verschiedene Ausgaben in den Trace geschrieben werden.

Die Ausgabe wird von Einträgen in der INI-Datei gesteuert.

Trace-Ausgaben kann man z.B. mit DebugView von SysInternals ( <https://download.sysinternals.com/files/DebugView.zip> ) sichtbar machen und als Logfile abspeichern.

## LogQuantumX

- 1: Einschalten von HBM-Logging (QXSystem.UseTraceLog = 1). Die Logs werden in die Datei „%APPDATA%\HBM\QuantumX\_API\QX\_API.log“ geschrieben
- 0: HBM-Logging bleibt ausgeschaltet.

## TraceDebug

- 1: Einschalten von Trace-Ausgaben.  
Die meisten Funktionsaufrufe und einige Status-Informationen werden in den Trace geschrieben
- 0: Diese Ausgaben werden nicht in den Trace geschrieben

## TraceHBMApi

- 1: Einschalten von Trace-Ausgaben über Aufrufe der HBM QuantumX-API.
- 0: Diese Ausgaben werden nicht in den Trace geschrieben

## TraceDebugDuringAcquisition

Sollte nur bei niedrigen Abtastraten verwendet werden.

- 1: Einschalten von Trace-Ausgaben.  
Funktionsaufrufe und einige Status-Informationen werden auch während einer Messung in den Trace geschrieben
- 0: Diese Ausgaben werden nicht in den Trace geschrieben

## TraceProcessData

Sollte nur bei niedrigen Abtastraten verwendet werden.

- 1: Einschalten von Trace-Ausgaben über das Abarbeiten von Messdaten während einer Messung
- 0: Diese Ausgaben werden nicht in den Trace geschrieben

# QuantumX-Treiber für DASyLab

---

## Beispiel für die Einträge in der DASyLab.ini:

[IMP40]

LogQuantumX=0

TraceDebug=1

TraceHBMApi=1

TraceDebugDuringAcquisition=1

TraceProcessData=1

QuantumX-Device-SN=9E5001CE1

QuantumX-SN=127490086344

## Unterschiede zum alten Treiber

### Technologie

Es wird nun statt des CATMAN-ActiveX die QuantumX-API 4.1 verwendet.

Es wird nun eine Firmwareversion im QuantumX ab oder neuer als 4.x benötigt.

### Geräteauswahl, IP-Adresse, Port

Die zu verwenden QuantumX können nun aus einer Liste aller angeschlossenen Geräte ausgewählt werden.

Die manuelle Eingabe von IP-Adresse und Port entfällt, ist aber möglich

### Ethernet / Firewire

Der Zugriff via Firewire geschieht mit „IP-over-Firewire“.

Die Auswahl „IP-Adresse oder Firewire entfällt.

## Auslesen von Informationen aus dem QuantumX

### Abtastrate / Blockgröße

Die Kanal-Abtastraten werden aus dem QuantumX ausgelesen und kanalspezifisch - also auch auf Wunsch - unterschiedlich übergeben. Die Blockgrößen pro Kanal werden berechnet aus einer neuen Blockgröße in s.

### Kanalnamen

Gerätebezeichnung, Kanalnamen, CAN-Kanalnamen sowie Einheiten werden aus dem QuantumX ausgelesen und an DASyLab übergeben.

### Lizensierung

Wenn „zur Lizenzierung verwenden“ angewählt wird, wird zur Eingabe der Lizenznummer aufgefordert. Diese wird dann in die INI-Datei eingetragen

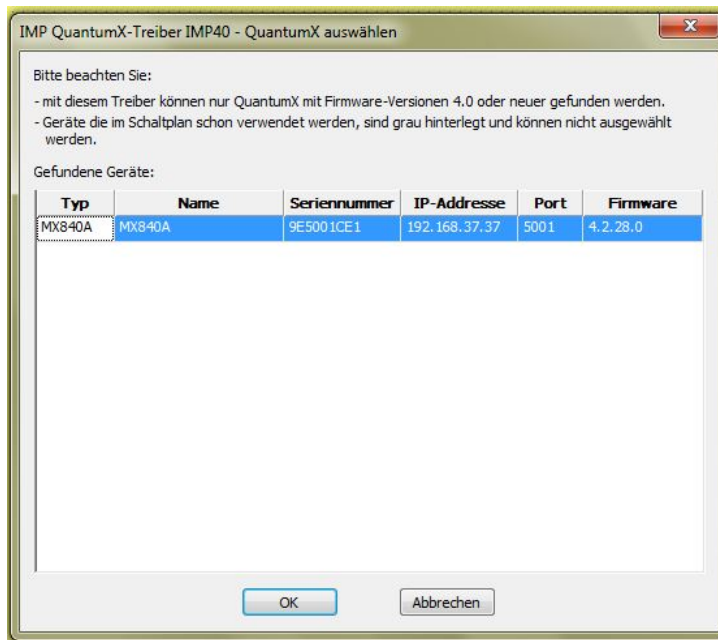
Siehe Abschnitt „Freischaltung“.

### Menu

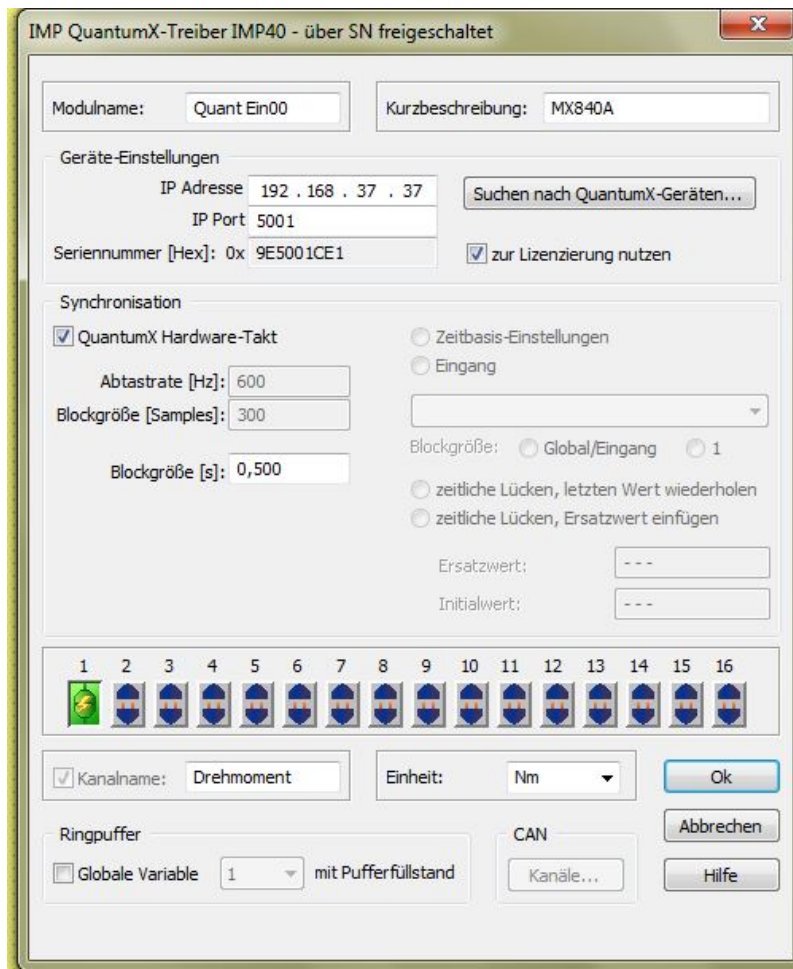
Die Menus heißen nunmehr „IMP40...“ statt „IMP...“

# QuantumX-Treiber für DASyLab

Gerät gefunden:



Modul-Einstellungen:



## CAN-Bus-Auswahl:

