Quick start guide

iX Detect VNC Client Connection

SER0055 – iX example project

1 Function and area of use

This document shows how to check whether a VNC Client is connected to the X2 panel.
The time for the cyclic check is adjustable inside the script module.

2 About this document

This quick start document should not be considered as a complete manual. It is an aid to be able to startup a normal application quickly and easily.

Use the following items in order to obtain a stable application:

In this document we have used following software and hardware

- iX Developer 2.40 SP5 / SP6
- X2 base/pro/marine/control/extreme series

For further information refer to

- iX Developer Reference Manual (MAxx831)
- iX Developer User's Guide (MAxx832)
- Beijer Electronics knowledge database, HelpOnline

This document and other Startup documents can be obtained from our homepage.
Please use the address *support.europe@beijerelectronics.com* for feedback about our Startup documents.

3 Table of Contents

4 The iX example project

The cyclic check whether a VNC Client is connected is done inside the script module SCM_VNCDetection.

4.1 The VNC server

In the System menu you can activate the VNC/Remote Access Server.
Optionally specify credentials and/or eenable "Notify when connection is active".
This option shows that a VNC Client connection is active in a small PopUp – but you don´t get this information bound to an iX Tag.
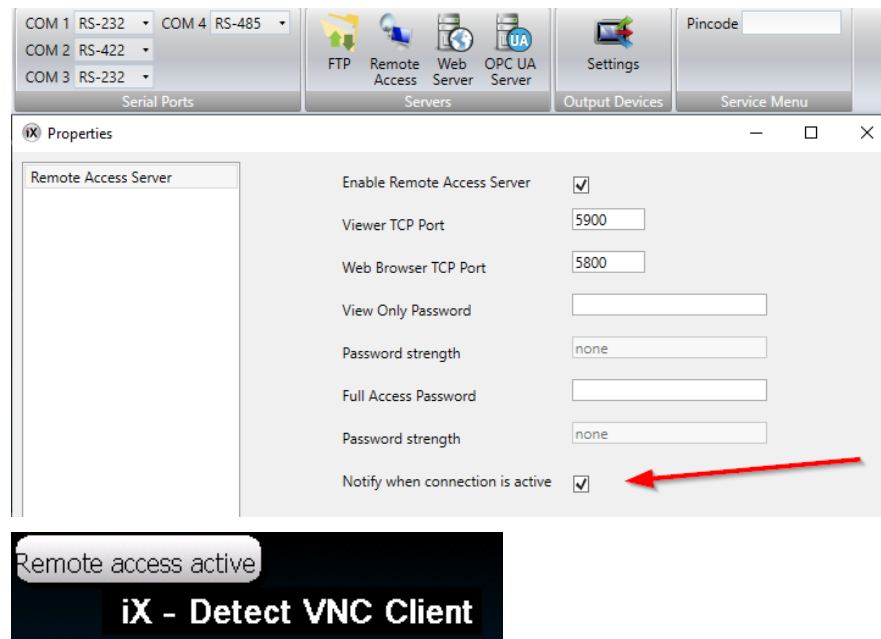
4.2 The Referenced Assembly OpenNETCF.Net.dll

OpenNETCF is committed to open source projects to help the mobile and embedded development community in their projects whether it be enterprise development or commercial development. All projects are under either the OpenNETCF Shared Source License or the MIT X11 license.

In "Project" → "Referenced Assemblies" the "OpenNETCF.Net.dll" is inserted.
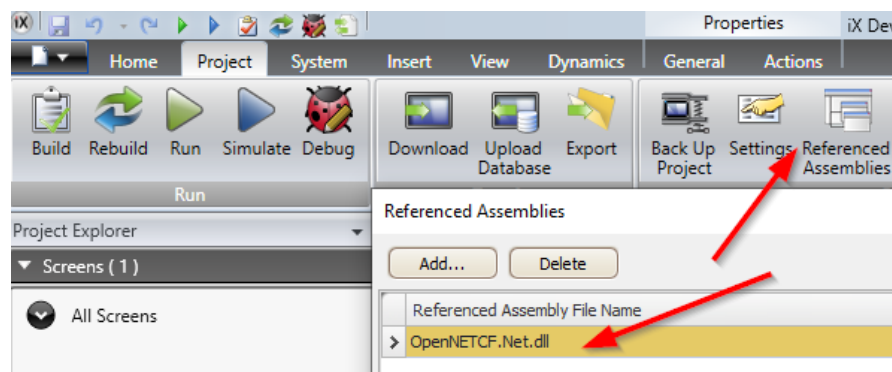
4.3   The Script Module

Inside the script module the following Namespace is inserted additionally to enable TCP socket info.

using OpenNETCF.Net.NetworkInformation;

Every iX script module brings a "Created" method with it, which is perfect to initialize stuff as it is worked on exactly once at the start of the iX project.

```
// The method is invoked when the script module object is created.
void SCM_VNCDetection_Created(System.Object sender, System.EventArgs e)
{
        m_Timer= new Timer();
        m_Timer.Enabled = true;
        m_Timer.Interval = 2000;              //ms - adjust time for cyclic check here.
        m_Timer.Tick += OnTimerTick;
}

// Cycle
private void OnTimerTick(System.Object sender, System.EventArgs e)
{
        CheckIfVNCPortisOpen(5900);          // This is the VNC port to check (default is 5900)
}


// Check if VNC port is open, gets port, sets the VncActive tag
public void CheckIfVNCPortisOpen(int port)
{
        bool foundPort = false;
        IPGlobalProperties ipProperties = IPGlobalProperties.GetIPGlobalProperties();
        TcpConnectionInformation[] tcpConnections = ipProperties.GetActiveTcpConnections();
        foreach (TcpConnectionInformation info in tcpConnections)
              if( (byteswap(info.LocalEndPoint.Port) == port) &&
                  (info.State.ToString() == "Established") )
                    {
// Set tag if port connection is established
                            Globals.Tags.VNCActive.Value = true;
                            foundPort = true;
                    }

        if(!foundPort)
// If no port is found, reset tag
              Globals.Tags.VNCActive.Value = false;
}
```

Normal.dotm, 2020-07-01

Beijer
ELECTRONICS

5 Adding the VNC Client Detection

Implementation

1. Import the script module "SCM_VNCDetection" , see example project (iX_Detect_VNC_Client_V1_0_0).

2. Create the Tag "VNCActive" with Datatyp BOOL

Tag			Controllers	
Name	Data T...	Access Right	Data T...	Controller 1
> VNCActive ···	BOOL	ReadWrite	DEFAULT	

3. Add the OpenNETCF.Net.dll to your referenced assemblies (see 4.2).

4. Optionally import Screen1 of the example project (iX_Detect_VNC_Client_V1_0_0).

5. Adapt the screen to your needs.

6. Transfer the application.

7. Run the application.

5.1 Import the project parts

Follow the steps to add the enclosed screen and the script module to your iX project:

1. Unpack the enclosed example ZIP-file to a temporary folder.

2. Start iX Developer and load your project.

3. In the Project Explorer, right-click in the lower left corner (1. in the picture)

4. In the list, select Import... (2. in the picture)

5. Navigate to the temporary folder, where you unpacked the ZIP-file and select SCM_VNCDetection.neo, click [Open].

   Select and add the OpenNETCF.Net.dll . It is located in the example project -folder iX_Detect_Internet_Connection_V1_0_0\ReferencedAssembly.

6. Select Screen1.neoxaml, click [Open].

7. Done!
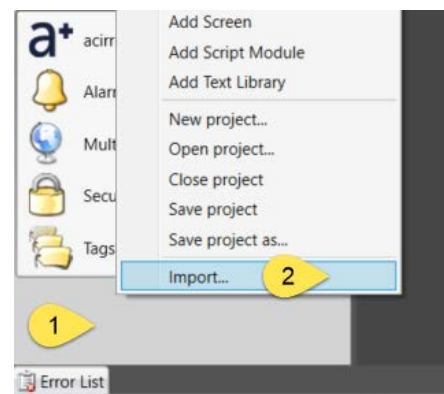
6 About Beijer Electronics

Beijer Electronics is a multinational, cross-industry innovator that connects people and technologies to optimize processes for business-critical applications. Our offer includes operator communication, automation solutions, digitalization, display solutions and support. As experts in user-friendly software, hardware and services for the Industrial Internet of Things, we empower you to meet your challenges through leading-edge solutions. Beijer Electronics is a Beijer Group company.
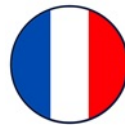
Beijer Group has a sale over 1.6 billion SEK in 2019 and is listed on the NASDAQ OMX Nordic Stockholm Small Cap list under the ticker BELE. www.beijergroup.com

China
Shanghai

Denmark
Roskilde

France
Champlan

Germany
Nürtingen

Italy
Salsomaggiore

Norway
Lier

South Korea
Geumcheon-gu

Sweden HQ
Malmö

Taiwan
Taipei City

Turkey
Istanbul

United Kingdom
Nottingham

Usa
Salt Lake City

6.1 Contact us

Global offices and distributors