https://www.halvorsen.blog

DAQ and OPC Overview

Hans-Petter Halvorsen

Table of Contents

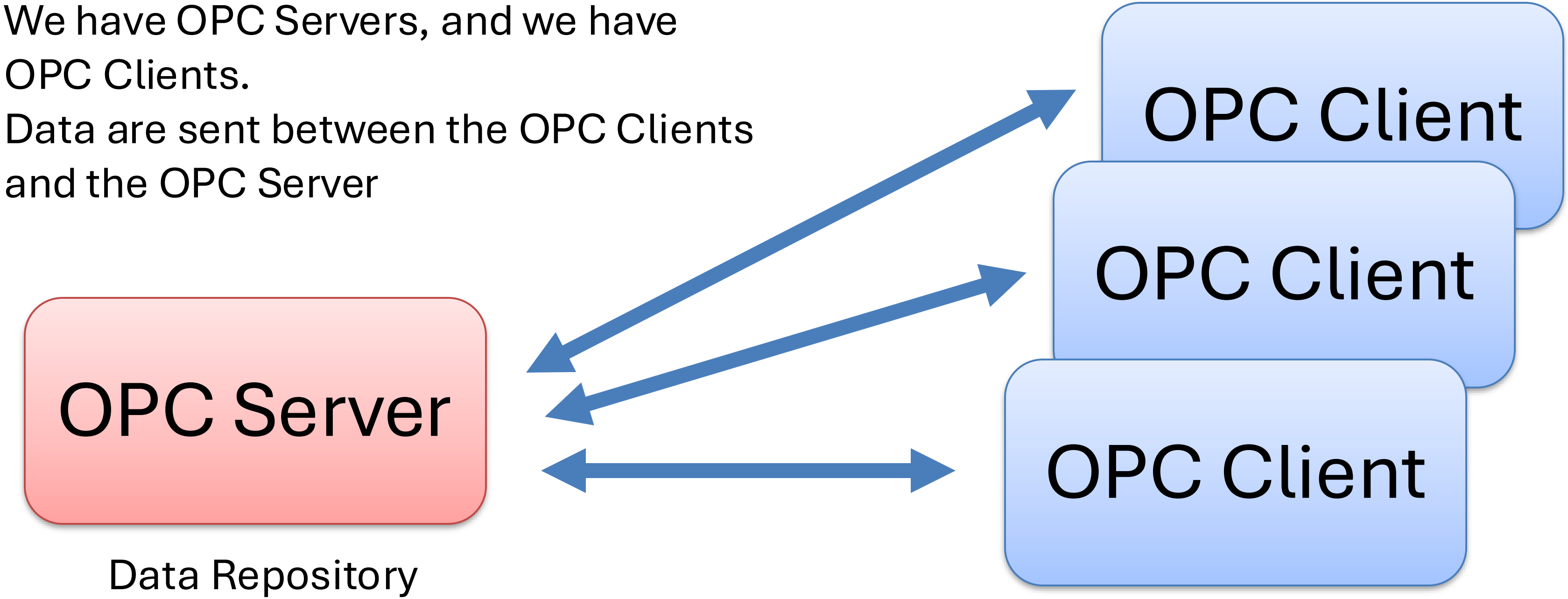
Introduction

What is OPC?

- OPC - “Open Process Control”/“Open Platform Communications”
- A standard that defines the communication of data between devices from different manufactures
- Requires an **OPC server** that communicates with the **OPC clients**
- OPC allows “plug-and-play”, gives benefits as reduces installation time and the opportunity to choose products from different manufactures
- Different standards: “Real-time” data (**OPC DA**), Historical data (**OPC HDA**), Alarm & Event data (**OPC AE**), etc.

Basic OPC concept

We have OPC Servers, and we have OPC Clients.
Data are sent between the OPC Clients and the OPC Server



Data Repository

Send Data (Write) to OPC Server
or Retrieve Data (Read) from OPC Server

OPC Specifications

OPC DA (Data Access)

The most common OPC specification is OPC DA, which is used to read and write “real-time” data. When vendors refer to OPC generically, they typically mean OPC DA.

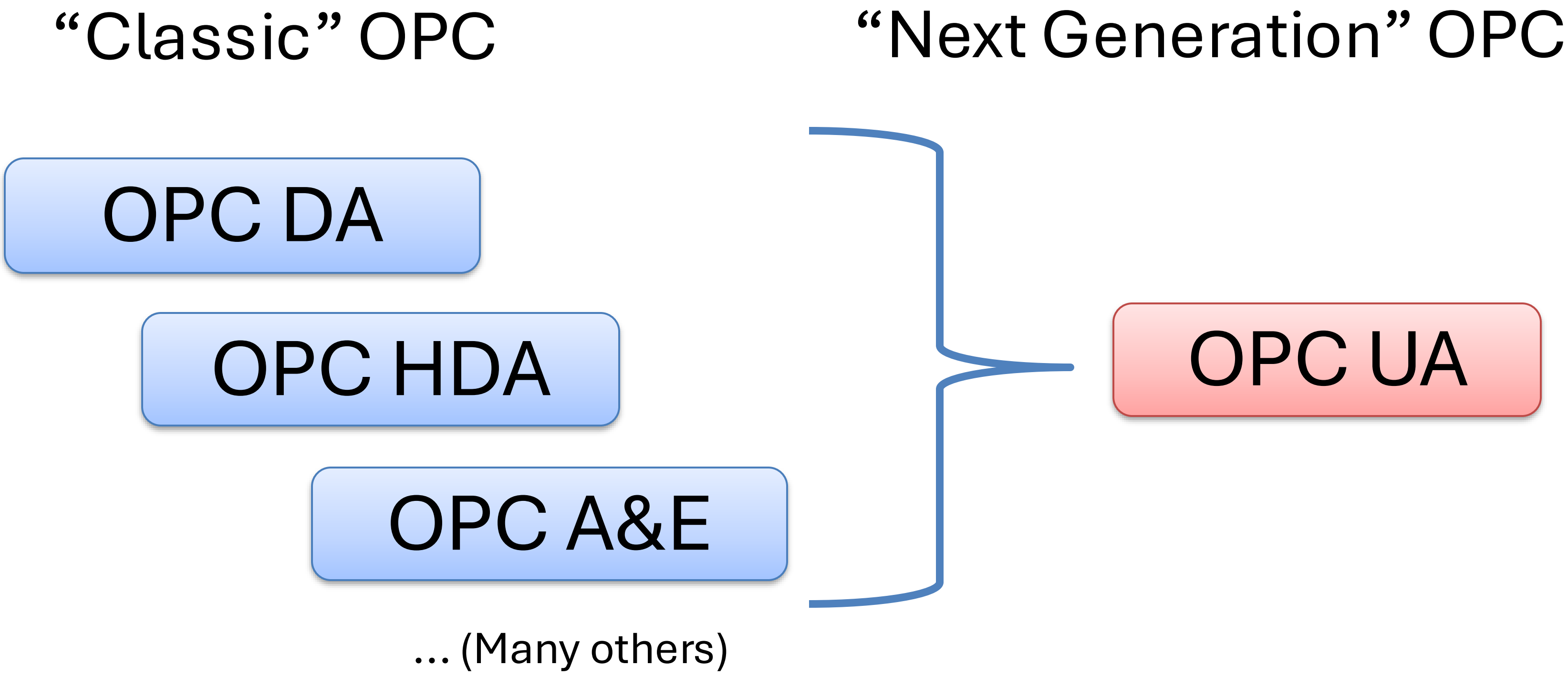
- OPC HDA (Historical Data Access)
- OPC A & E (Alarms & Events)
- … (many others)

These OPC specification are based on the OLE, COM, and DCOM technologies developed by Microsoft for the Microsoft Windows operating system family. This makes it complicated to make it work in a modern Network! Typically, you need a Tunneller Software in order to share the OPC data in a network (between OPC Servers and Clients)

OPC UA (Unified Architecture)

OPC UA eliminating the need to use a Microsoft Windows based platform of earlier OPC versions. OPC UA combines the functionality of the existing OPC interfaces with new technologies such as XML and Web Services (HTTP, SOAP)

“Next Generation” OPC



“Classic” OPC

“Next Generation” OPC

OPC DA

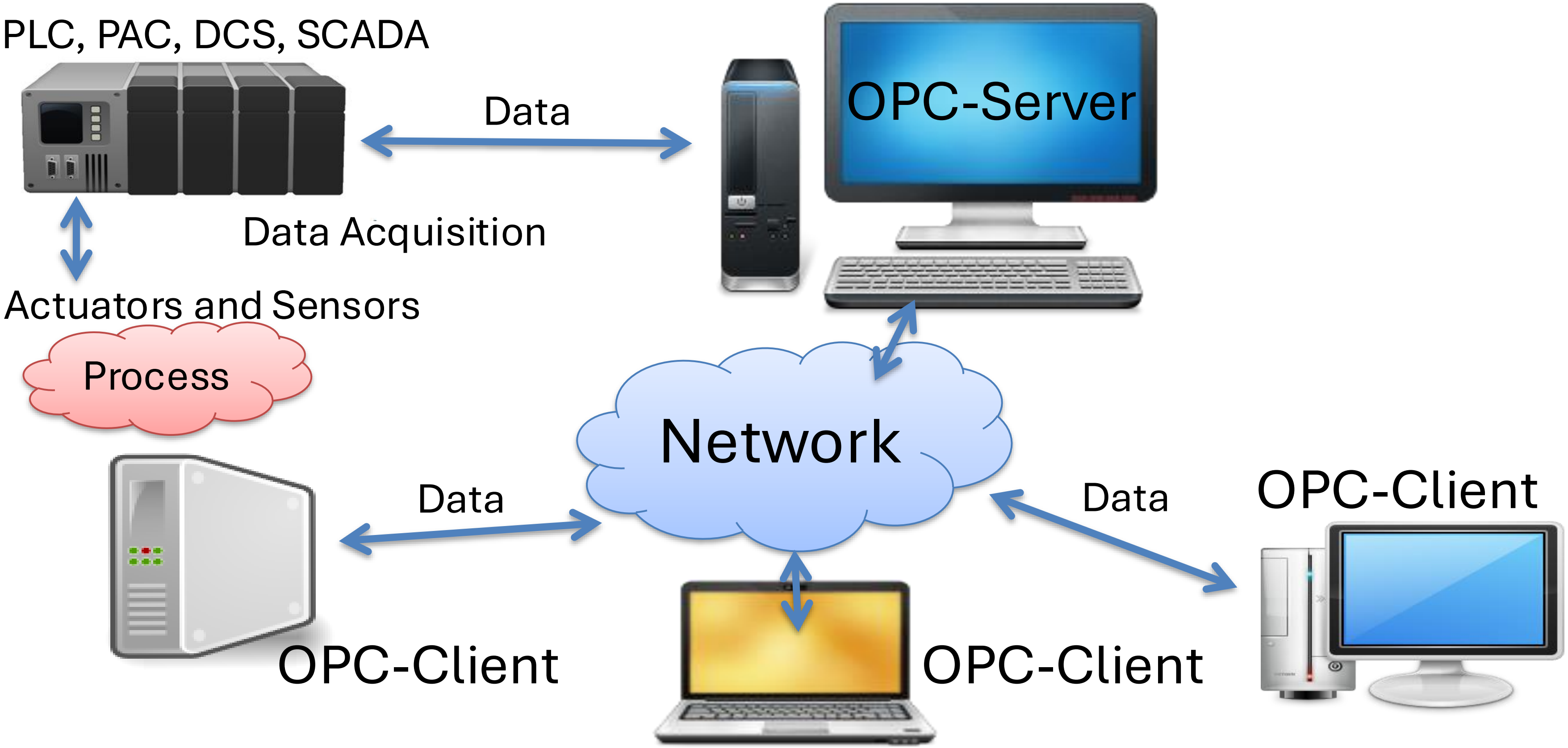
OPC HDA

OPC A&E

… (Many others)

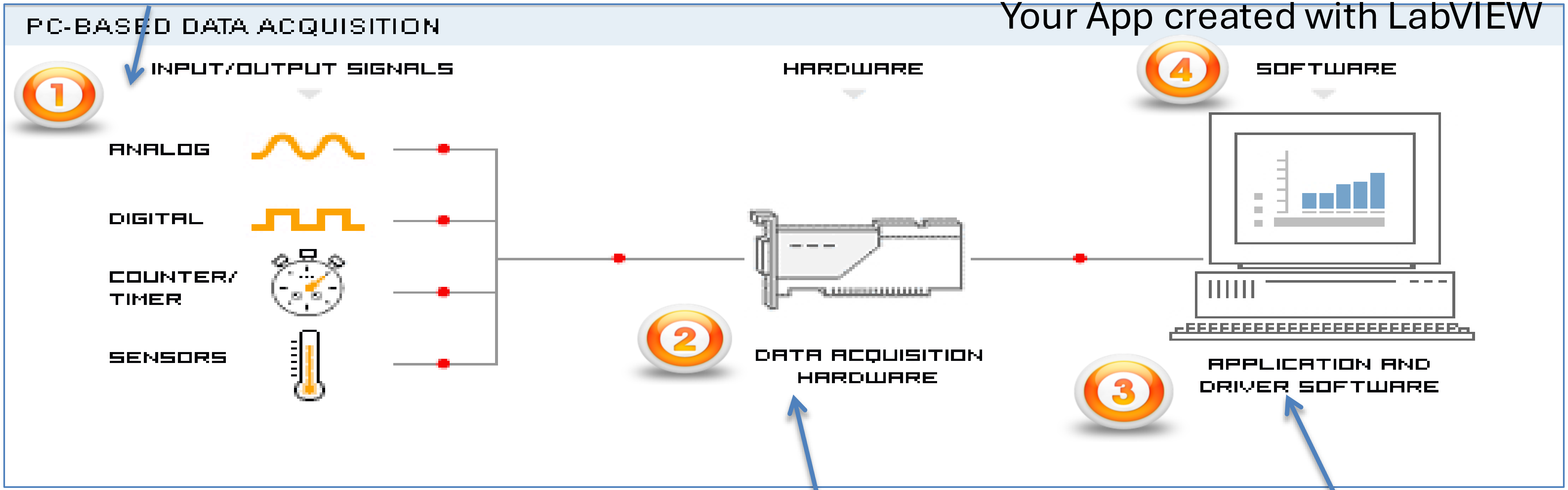
OPC UA

Typical OPC Scenario



Data Acquisition (DAQ)

Sensors, etc.



Your App created with LabVIEW

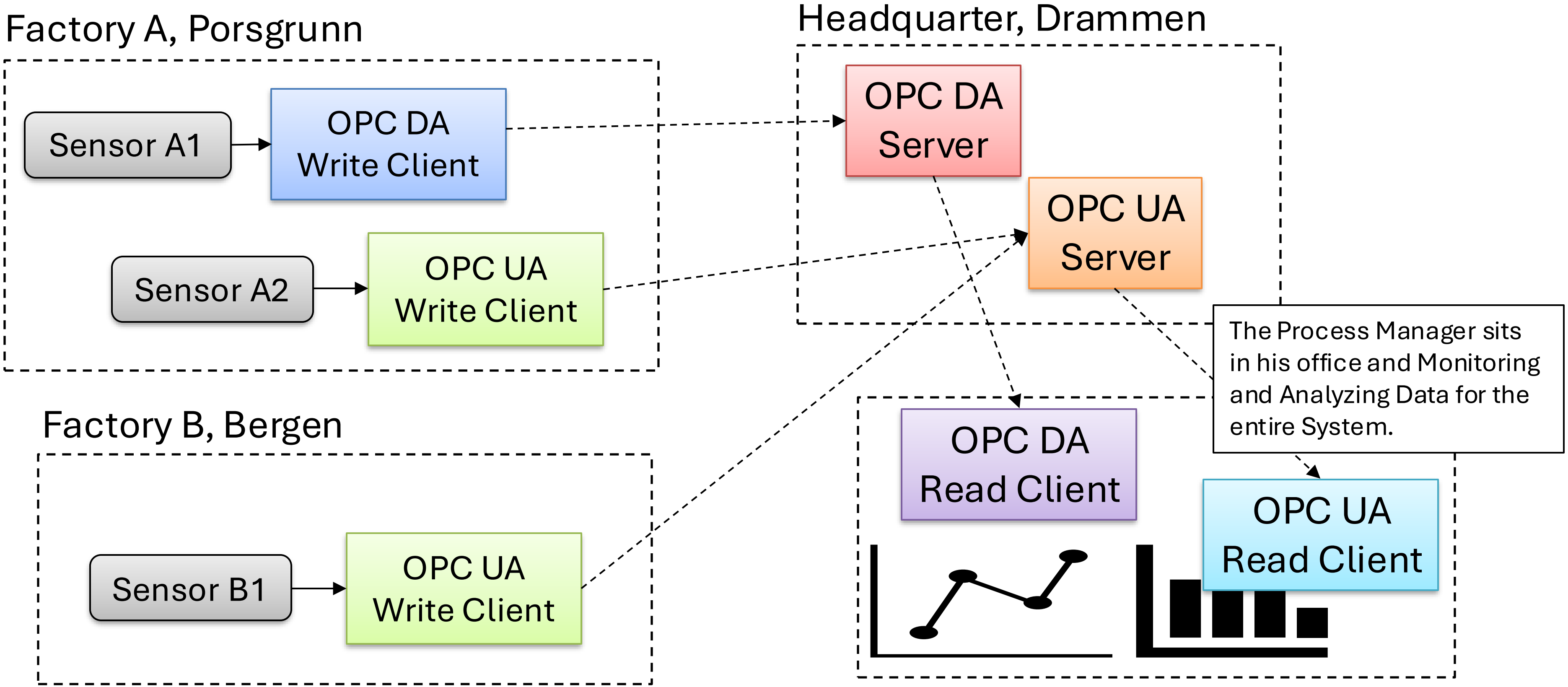
NI TC-01 Thermocouple
or USB-6008 Device

NI DAQmx Driver

<u>A DAQ System consists of 4 parts:</u>

1. Physical input/output signals, sensors
2. DAQ device/hardware
3. Driver software
4. Your software application (Application software)

Use Case Scenario for DAQ and OPC System



The Process Manager sits in his office and Monitoring and Analyzing Data for the entire System.

Office Building, OSLO

A typical factory can include both OPC DA and OPC UA parts

DAQ with TC-01 Thermocouple

Hans-Petter Halvorsen

Table of Contents

TC-01 Thermocouple Sensor

TC-01 Thermocouple Temperature Sensor is made by NI, the same company that develop LabVIEW

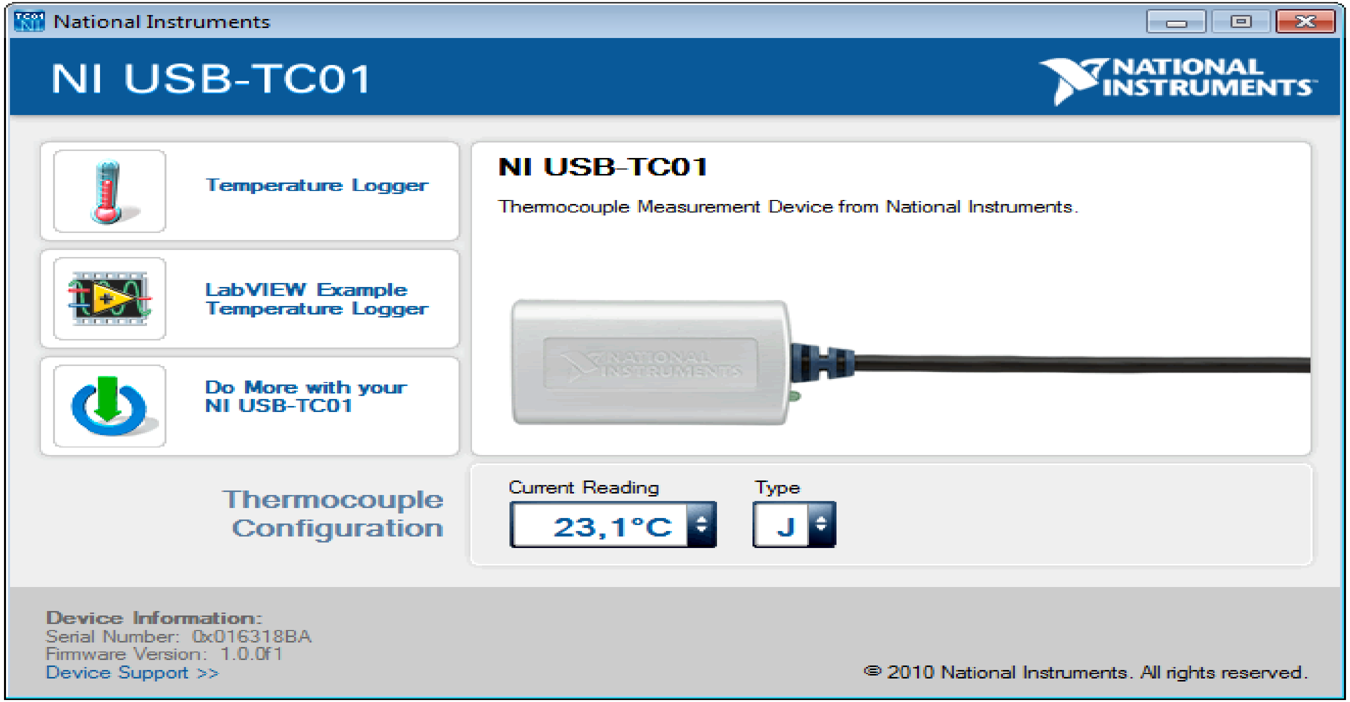
Sample Rate: 4 Samples/S

https://www.ni.com/en-no/support/model.usb-tc01.html

Datasheet: https://www.ni.com/pdf/manuals/374918b.pdf

Getting Started with TC-01

The following window should pop up automatically when you plug in your NI USB-TC01 device in your USB port (if not, select “**TC01Launcher.exe**”):



National Instruments

NI USB-TC01

NATIONAL INSTRUMENTS
Temperature Logger
LabVIEW Example Temperature Logger
Do More with your NI USB-TC01
Thermocouple Configuration

NI USB-TC01

Thermocouple Measurement Device from National Instruments.

Current Reading 23,1°C

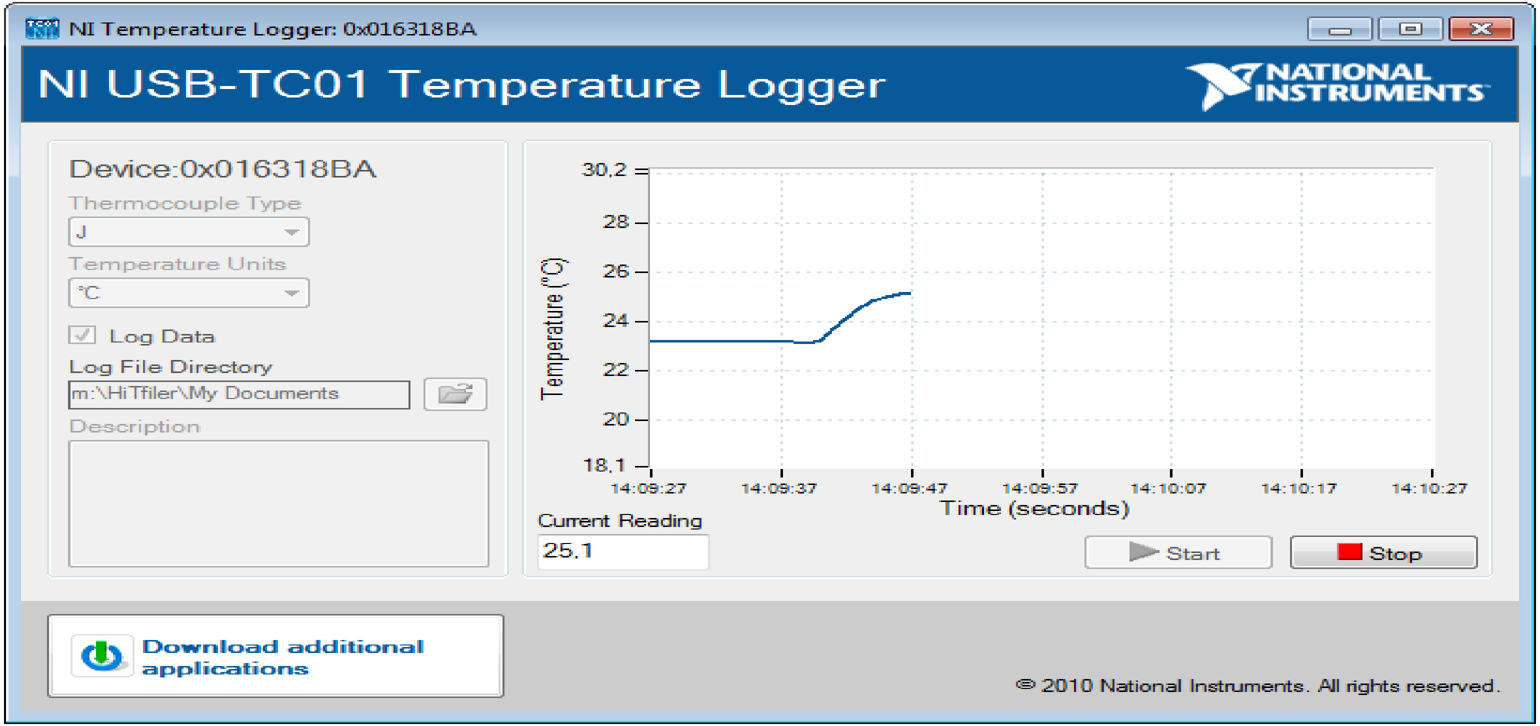
Type J

Device Information:
Serial Number: 0x016318BA
Firmware Version: 1.0.0f1
Device Support >>

© 2010 National Instruments. All rights reserved.

Built-in Temperature Logger

The TC-01 comes with a built-in Temperature Logger (No Driver or programming needed):



NI Temperature Logger: 0x016318BA

NI USB-TC01 Temperature Logger

Device:0x016318BA

Thermocouple Type: J

Temperature Units: °C

☑ Log Data

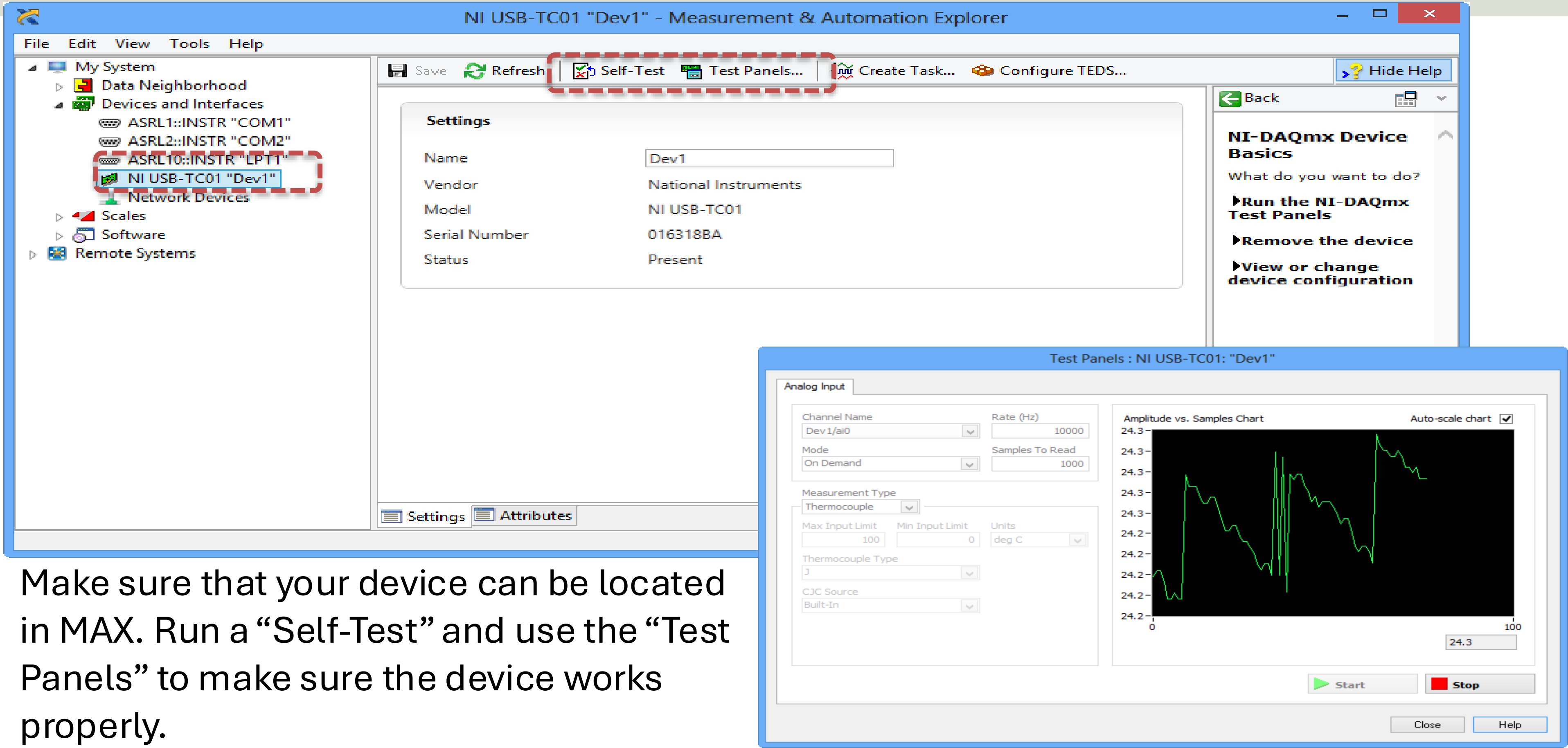
Log File Directory: m:\HiTfiler\My Documents

Description

Current Reading: 25.1

© 2010 National Instruments. All rights reserved.

Measurement & Automation Explorer (MAX)



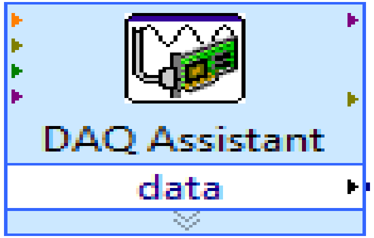
Settings

Name	Dev1
Vendor	National Instruments
Model	NI USB-TC01
Serial Number	016318BA
Status	Present

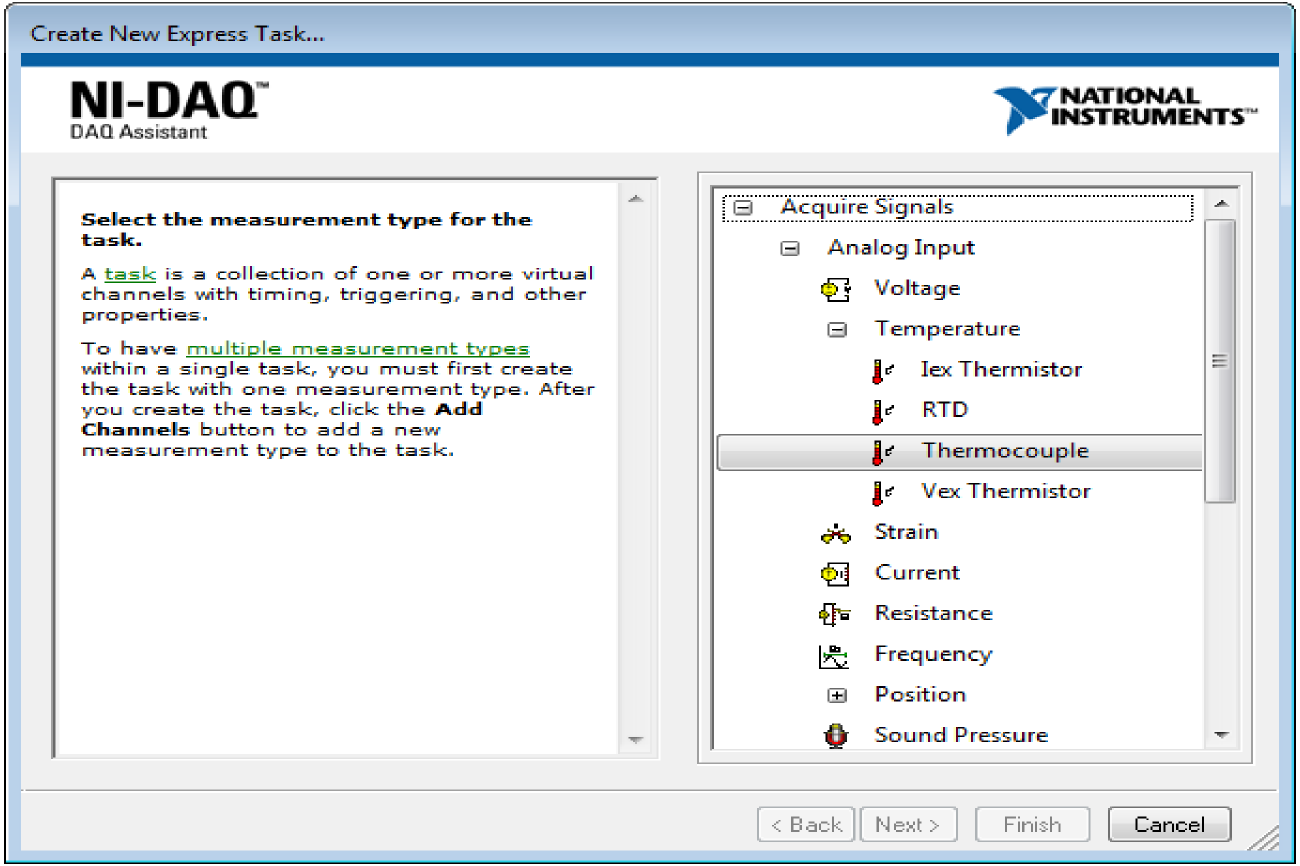
Channel Name: Dev1/ai0
Rate (Hz): 10000
Mode: On Demand
Samples To Read: 1000
Measurement Type: Thermocouple
Max Input Limit: 100
Min Input Limit: 0
Units: deg C
Thermocouple Type: J
CJC Source: Built-In

Make sure that your device can be located in MAX. Run a “Self-Test” and use the “Test Panels” to make sure the device works properly.

LabVIEW DAQ Assistant



When you place the **DAQ Assistant** on the Block Diagram, a Wizard automatically pops up where you configure what you want to do, i.e., if you want to Read or Write Data, Analog or Digital signals, which channel you want to use, etc.

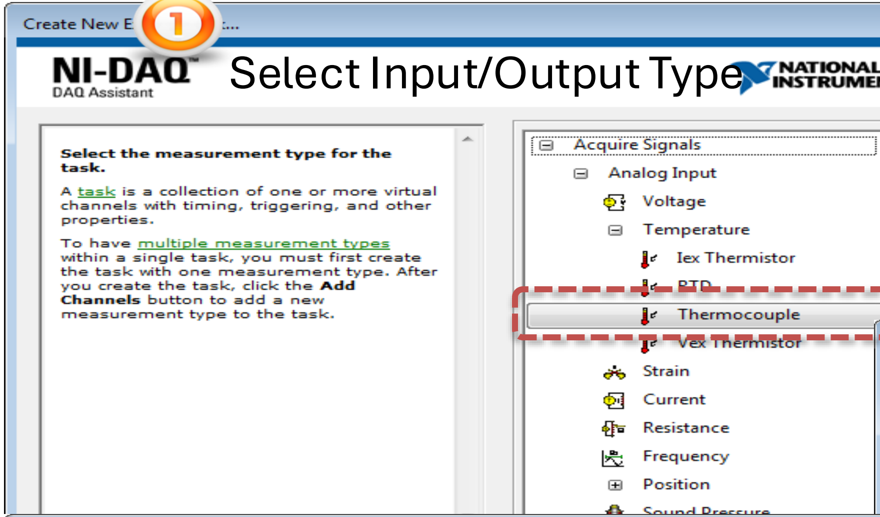


Select the measurement type for the task.

A task is a collection of one or more virtual channels with timing, triggering, and other properties.

To have multiple measurement types within a single task, you must first create the task with one measurement type. After you create the task, click the **Add Channels** button to add a new measurement type to the task.

LabVIEW DAQ Assistant



1 Select Input/Output Type

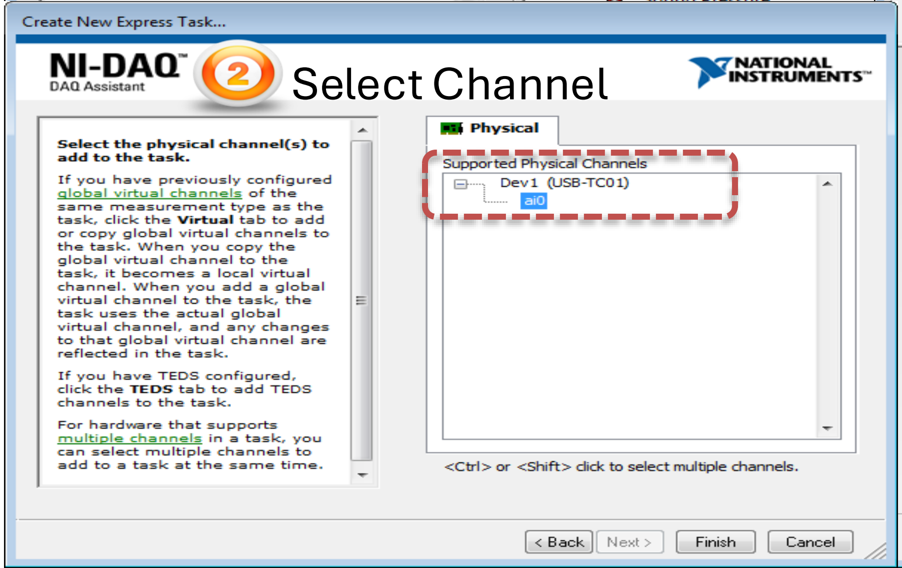
Create New E ...

NI-DAQ™ DAQ Assistant

Select the measurement type for the task.

A task is a collection of one or more virtual channels with timing, triggering, and other properties.

To have multiple measurement types within a single task, you must first create the task with one measurement type. After you create the task, click the **Add Channels** button to add a new measurement type to the task.



2 Select Channel

Create New Express Task...

NI-DAQ™ DAQ Assistant

Select the physical channel(s) to add to the task.

If you have previously configured global virtual channels of the same measurement type as the task, click the **Virtual** tab to add or copy global virtual channels to the task. When you copy the global virtual channel to the task, it becomes a local virtual channel. When you add a global virtual channel to the task, the task uses the actual global virtual channel, and any changes to that global virtual channel are reflected in the task.

If you have TEDS configured, click the **TEDS** tab to add TEDS channels to the task.

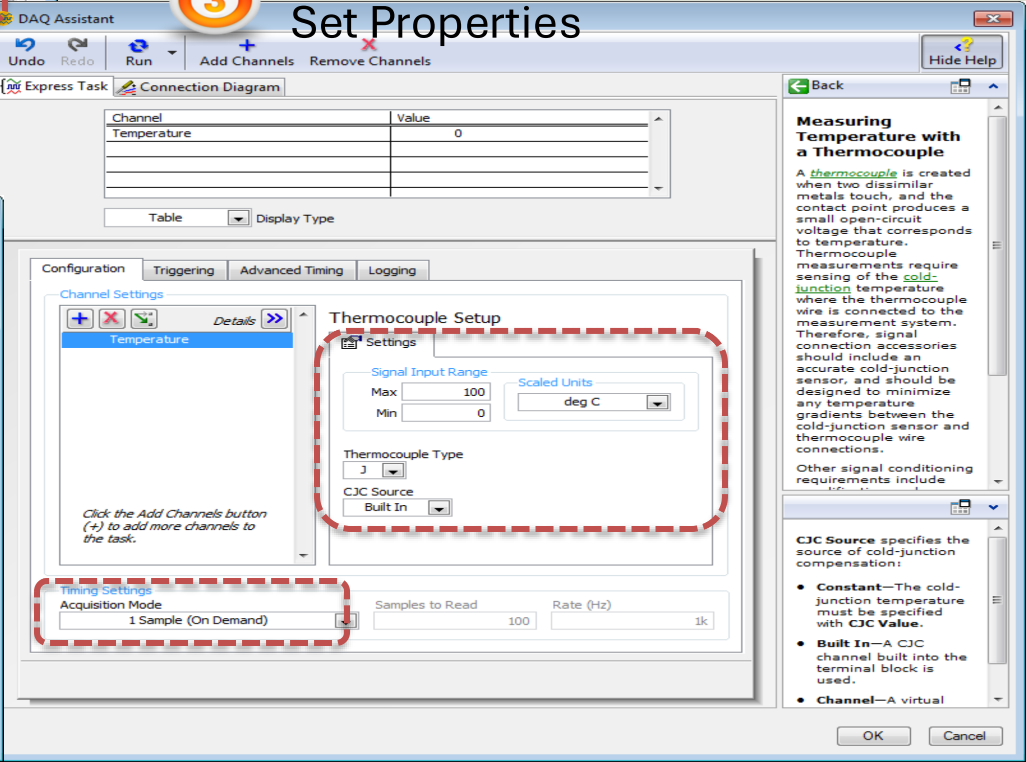
For hardware that supports multiple channels in a task, you can select multiple channels to add to a task at the same time.

Supported Physical Channels

Dev1 (USB-TC01)

ai0

<Ctrl> or <Shift> click to select multiple channels.



3 Set Properties

DAQ Assistant

Channel	Value
Temperature	0

Display Type: Table

Channel Settings: Temperature

Click the Add Channels button (+) to add more channels to the task.

Thermocouple Setup

Settings

Signal Input Range: Max 100, Min 0

Scaled Units: deg C

Thermocouple Type: J

CJC Source: Built In

Timing Settings

Acquisition Mode: 1 Sample (On Demand)

Samples to Read: 100

Rate (Hz): 1k

Measuring Temperature with a Thermocouple

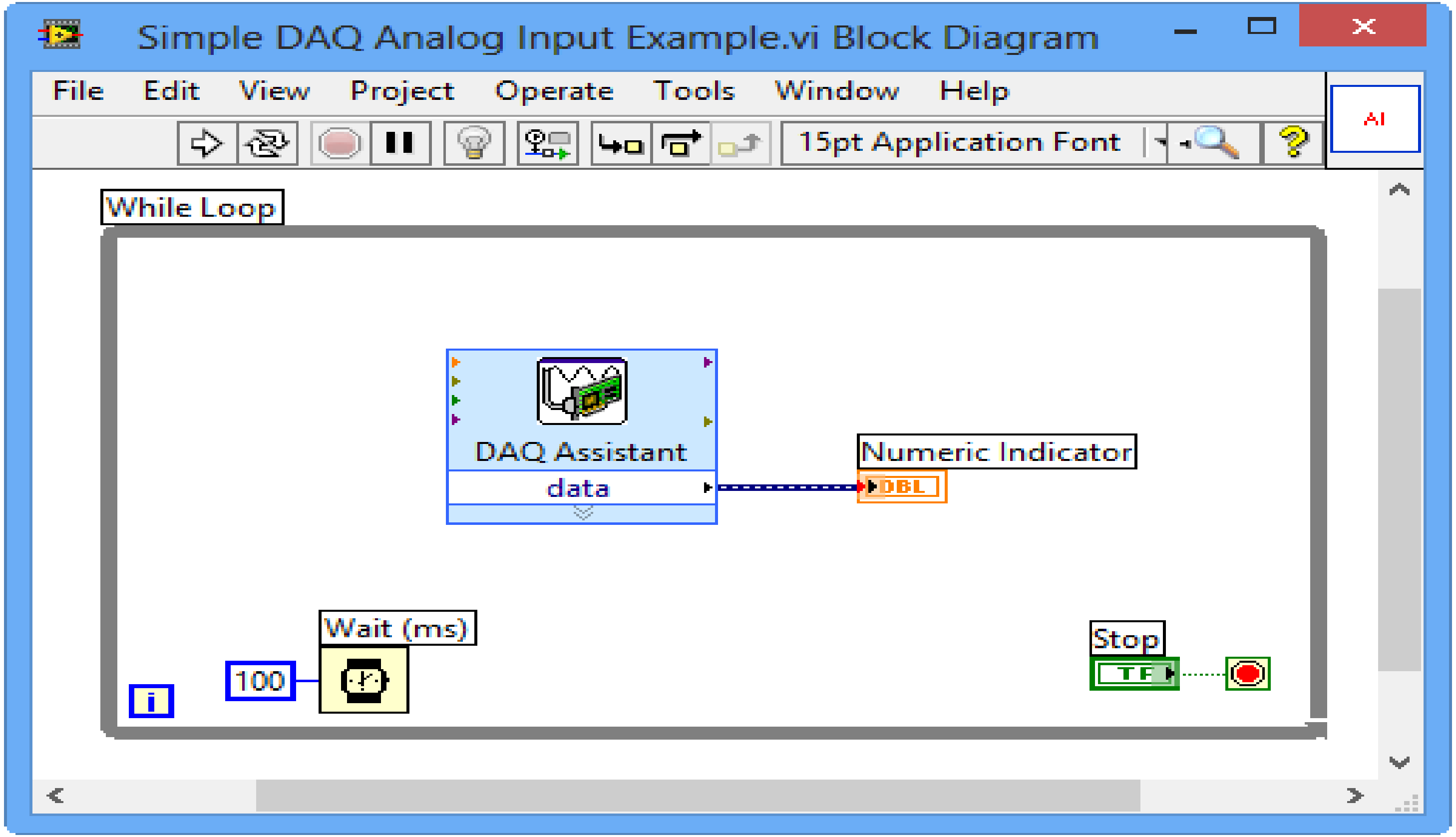
A thermocouple is created when two dissimilar metals touch, and the contact point produces a small open-circuit voltage that corresponds to temperature. Thermocouple measurements require sensing of the cold-junction temperature where the thermocouple wire is connected to the measurement system. Therefore, signal connection accessories should include an accurate cold-junction sensor, and should be designed to minimize any temperature gradients between the cold-junction sensor and thermocouple wire connections.

Other signal conditioning requirements include

CJC Source specifies the source of cold-junction compensation:

- **Constant**—The cold-junction temperature must be specified with **CJC Value**.
- **Built In**—A CJC channel built into the terminal block is used.
- **Channel**—A virtual

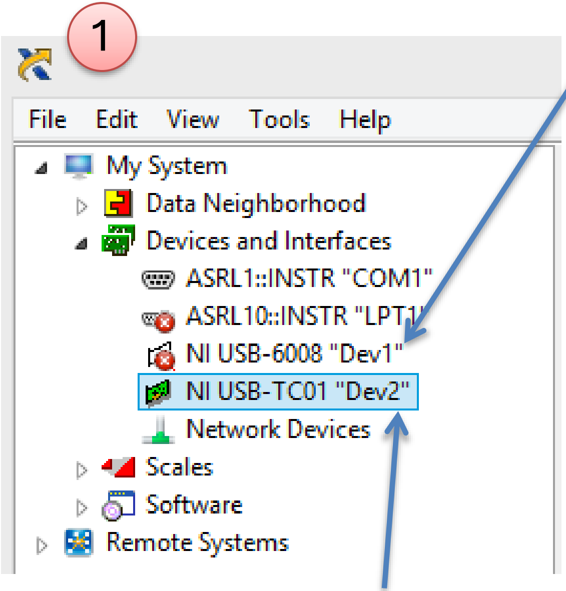
Read Data from TC-01 Device



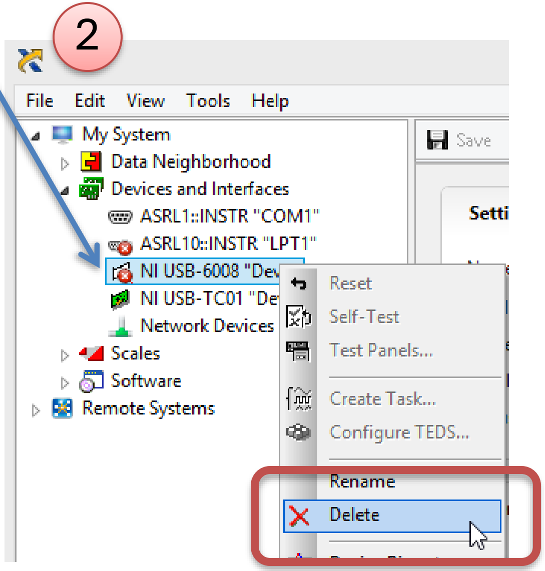
Not working after you got a new Device?

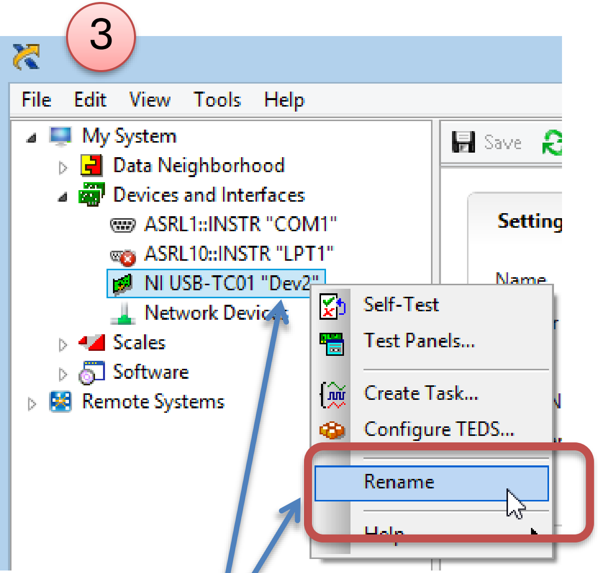
Solution, Alt 1: Open **MAX** (Measurement & Automation Explorer) in order to Fix-it!

Delete Old Device



New Device



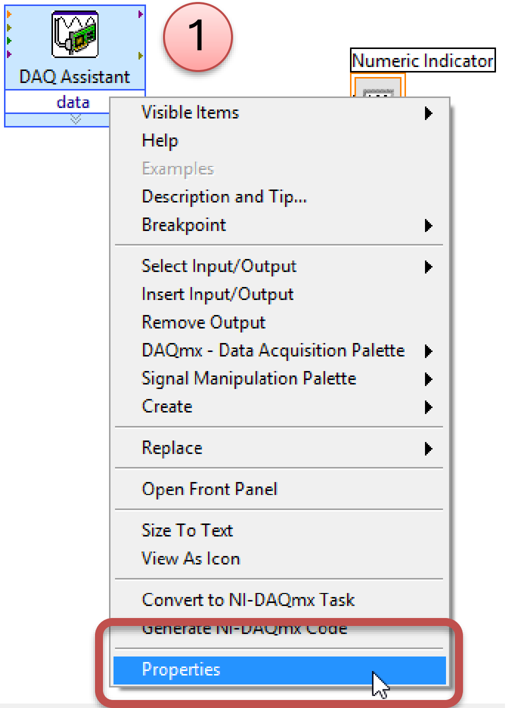


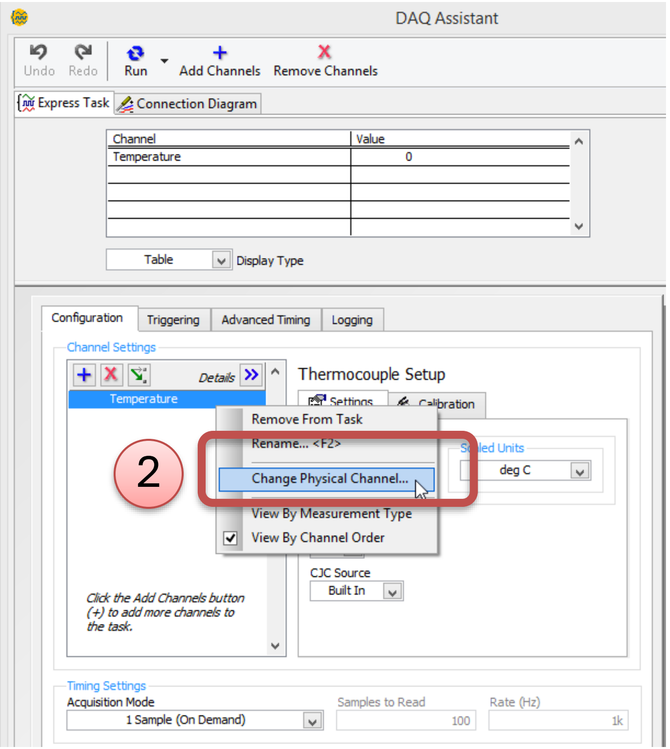
Rename the New Device with the same Name as the Old one

Not working after you got a new Device?

Solution, Alt 2: Change the Settings in the DAQ Assistant in your LabVIEW Applica

Right-click and select “Change Physical Channel”





Channel	Value
Temperature	0

Display Type: Table

Temperature

Scaled Units: deg C

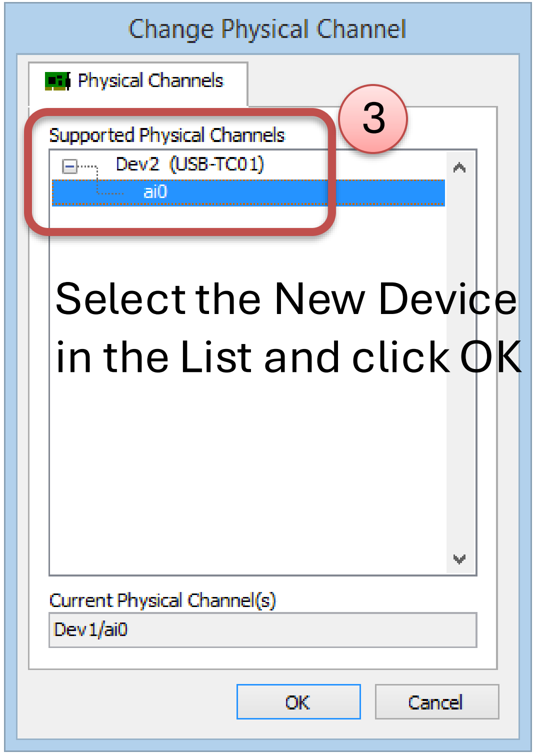
CJC Source: Built In

Click the Add Channels button (+) to add more channels to the task.

Acquisition Mode: 1 Sample (On Demand)

Samples to Read: 100

Rate (Hz): 1k



Supported Physical Channels

Dev2 (USB-TC01)
ai0

Current Physical Channel(s): Dev1/ai0

Select the New Device in the List and click OK

DAQ with USB-6008

Table of Contents

Hans-Petter Halvorsen

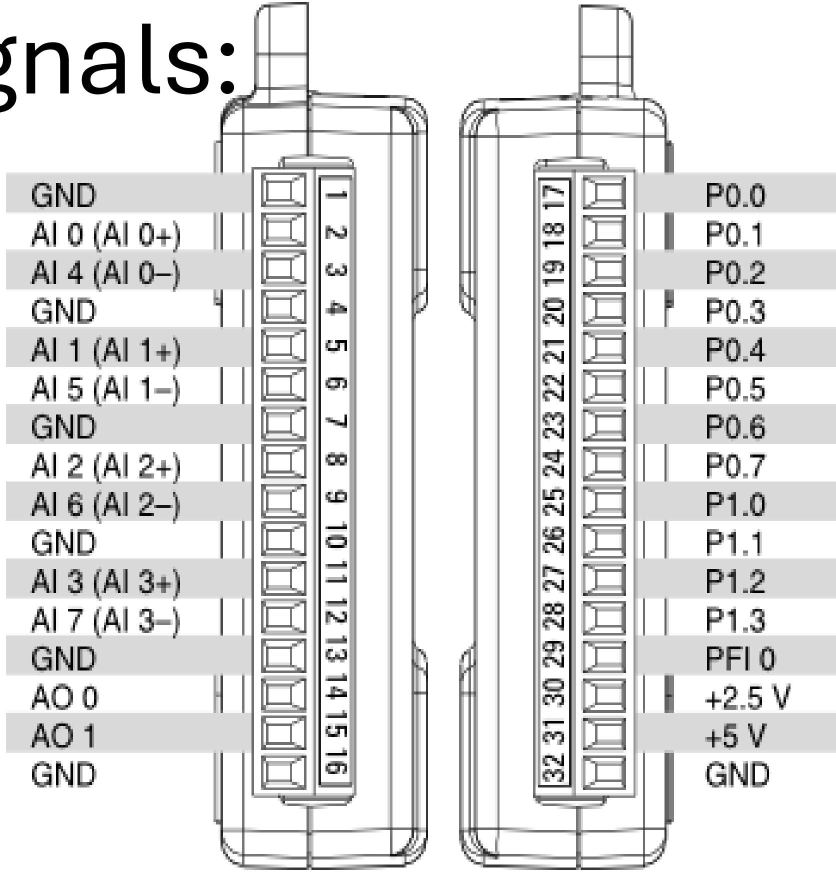
USB-6008

- USB-6008 is a DAQ Device from NI
- Can be used within LabVIEW
- NI-DAQmx Driver
- It has Analog and Digital Inputs and Outputs

USB-6008

4 different types of Signals:

- AO – Analog Output
- AI – Analog Input
- DO – Digital Output
- DI – Digital Input



GND	1	17	P0.0
AI 0 (AI 0+)	2	18	P0.1
AI 4 (AI 0–)	3	19	P0.2
GND	4	20	P0.3
AI 1 (AI 1+)	5	21	P0.4
AI 5 (AI 1–)	6	22	P0.5
GND	7	23	P0.6
AI 2 (AI 2+)	8	24	P0.7
AI 6 (AI 2–)	9	25	P1.0
GND	10	26	P1.1
AI 3 (AI 3+)	11	27	P1.2
AI 7 (AI 3–)	12	28	P1.3
GND	13	29	PFI 0
AO 0	14	30	+2.5 V
AO 1	15	31	+5 V
GND	16	32	GND

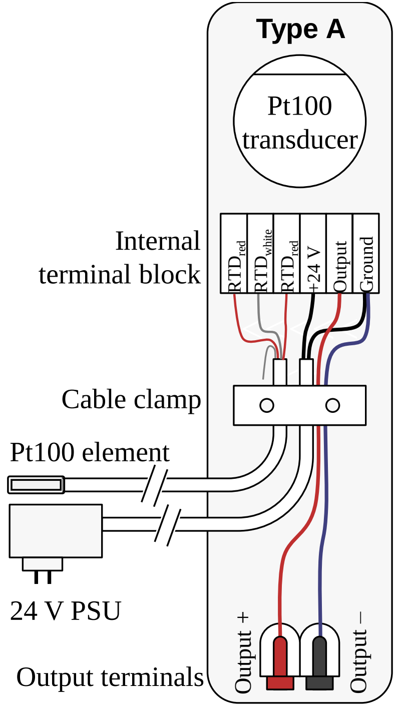
Temperature Sensors

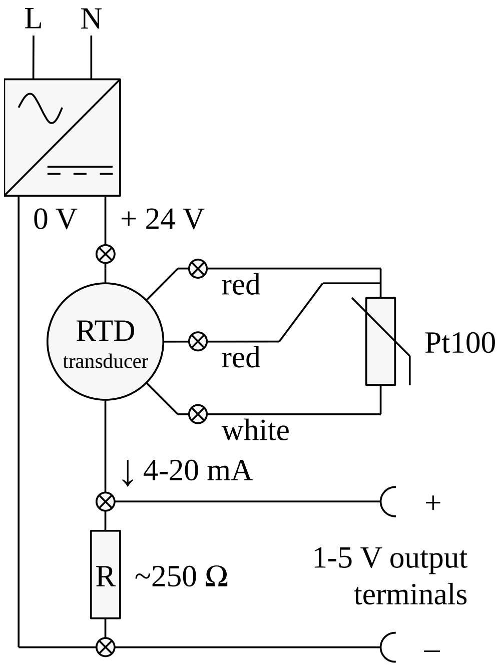
In the Laboratory we have different types of Temperature Sensors that we can connect to the USB-6008 DAQ device:

- PT-100
  - A Pt100 element is a RTD that uses platinum (Pt) as the resistor element. A Pt100 element is calibrated so that a temperature of $0°C$ yields a resistance of exactly 100Ω.
- TMP36
  - It provides a voltage output that is linearly proportional to the Celsius temperature.
- Thermistor
  - A thermistor is an electronic component that changes resistance to temperature - so-called Resistance Temperature Detectors (RTD).

PT-100

In the Laboratory we have a PT-100 device with Power Supply and PT-100 transducer:





In the laboratory we have 2 different Transducers/Transmitters:

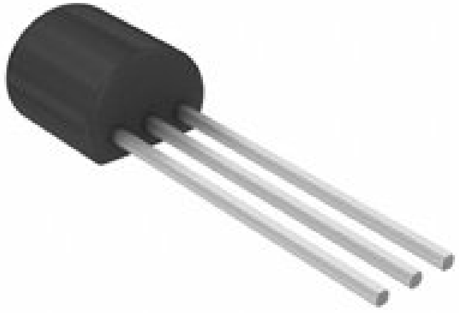
$$0 - 50°\text{C} \rightarrow 4 - 20mA$$

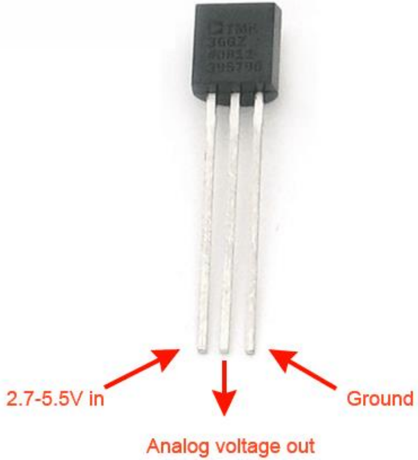
$$0 - 1 + 0°\text{C} \rightarrow 4 - 20mA$$

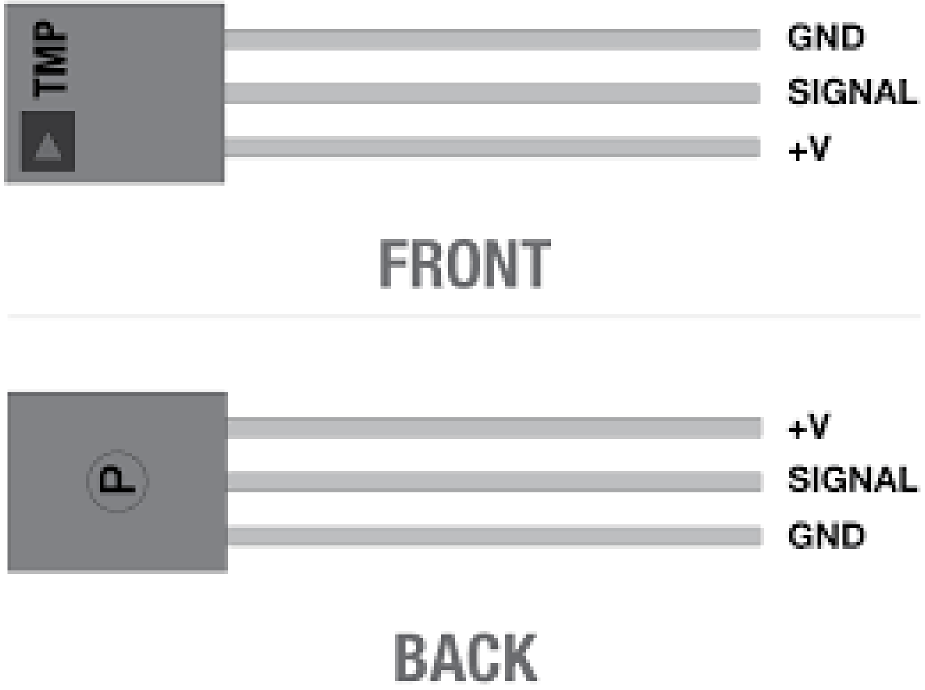
You must scale the output to a temperature value in degree Celsius

The PT-100 device can easily be connected to the USB-6008 DAQ device with Banana cable

TMP36

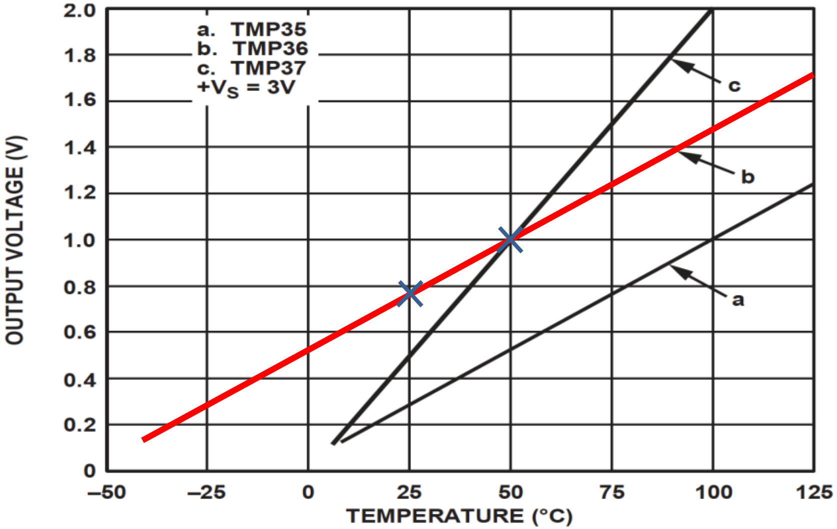






TMP is a small, low-cost temperature sensor and cost about $1 (you can buy it “everywhere

Linear Scaling



Convert form Voltage (V) to degrees Celsius

From the Datasheet we have:

$$(x_1, y_1) = (0.75V, 25°C)$$
$$(x_2, y_2) = (1V, 50°C)$$

There is a linear relationship between Voltage and degrees Celsius:

$$y = ax + b$$

We can find a and b using the following known formula:

$$y - y_1 = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1)$$

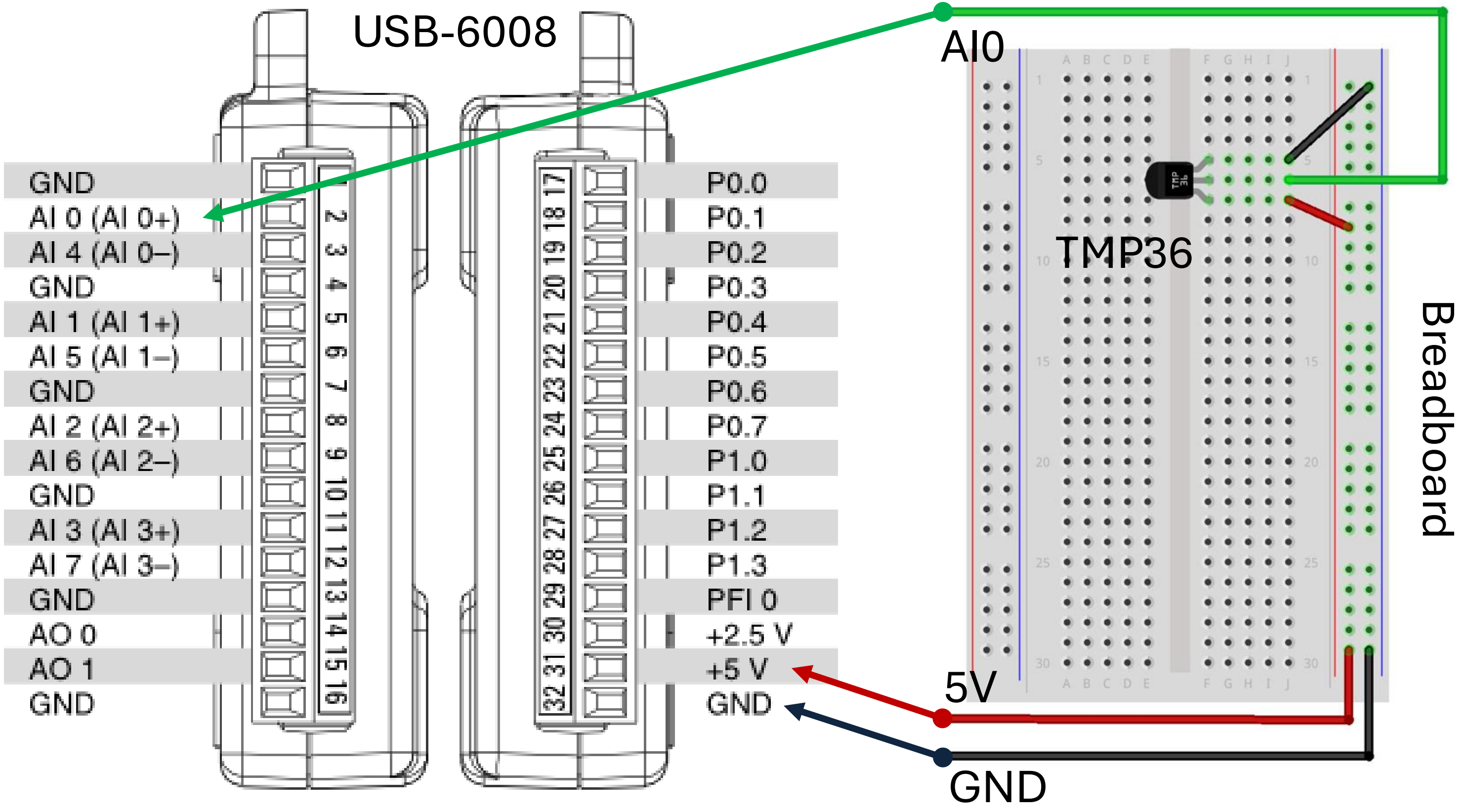
This gives:

$$y - 25 = \frac{50 - 25}{1 - 0.75}(x - 0.75)$$

Then we get the following formula:

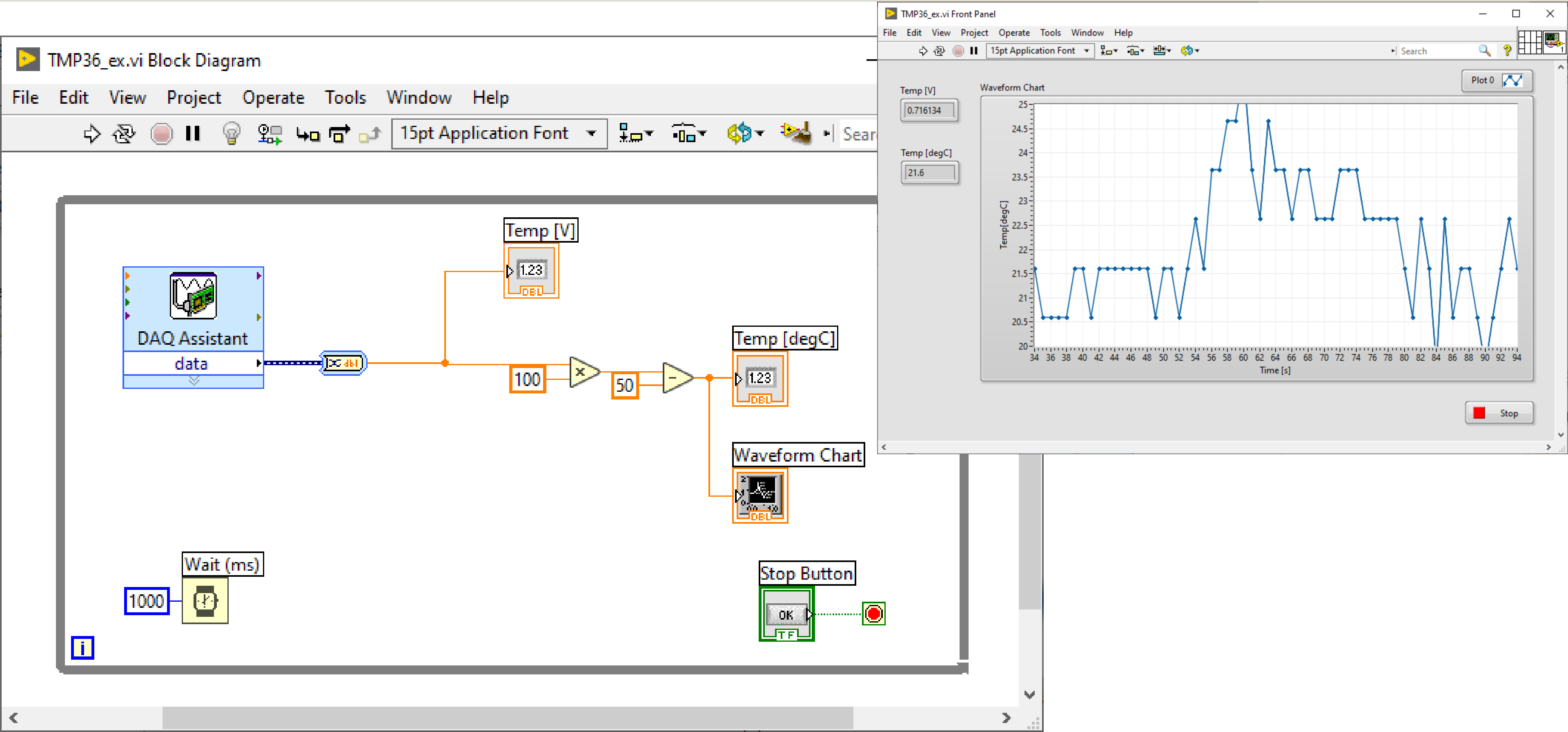
$$y = 100x - 50$$

Wiring



USB-6008
GND
AI 0 (AI 0+)
AI 4 (AI 0–)
GND
AI 1 (AI 1+)
AI 5 (AI 1–)
GND
AI 2 (AI 2+)
AI 6 (AI 2–)
GND
AI 3 (AI 3+)
AI 7 (AI 3–)
GND
AO 0
AO 1
GND
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
P0.0
P0.1
P0.2
P0.3
P0.4
P0.5
P0.6
P0.7
P1.0
P1.1
P1.2
P1.3
PFI 0
+2.5 V
+5 V
GND
AI0
TMP36
Breadboard
5V
GND

Plotting Example

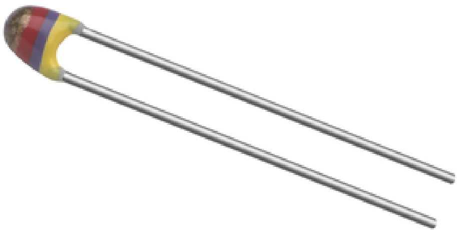


Temp [V] 0.716134

Temp [degC] 21.6

Thermistor

A thermistor is an electronic component that changes resistance to temperature - so-called Resistance Temperature Detectors (RTD). It is often used as a temperature sensor.



Our Thermistor is a so-called NTC (Negative Temperature Coefficient). In a NTC Thermistor, resistance decreases as the temperature rises.

There is a **non-linear relationship** between resistance and excitement. To find the temperature we can use the following equation (**Steinhart-Hart equation**):

[Wikipedia]

$$\frac{1}{T} = A + B\ln(R) + C(\ln(R))^3$$

where A, B, C are constants given below

$A = 0.001129148, B = 0.000234125 \; and \; C = 8.76741E-08$

Steinhart-Hart Equation

To find the Temperature we can use Steinhart-Hart Equation:

$$\frac{1}{T_K} = A + B\ln(R) + C(\ln(R))^3$$

This gives:

$$T_K = \frac{1}{A + B\ln(R) + C(\ln(R))^3}$$

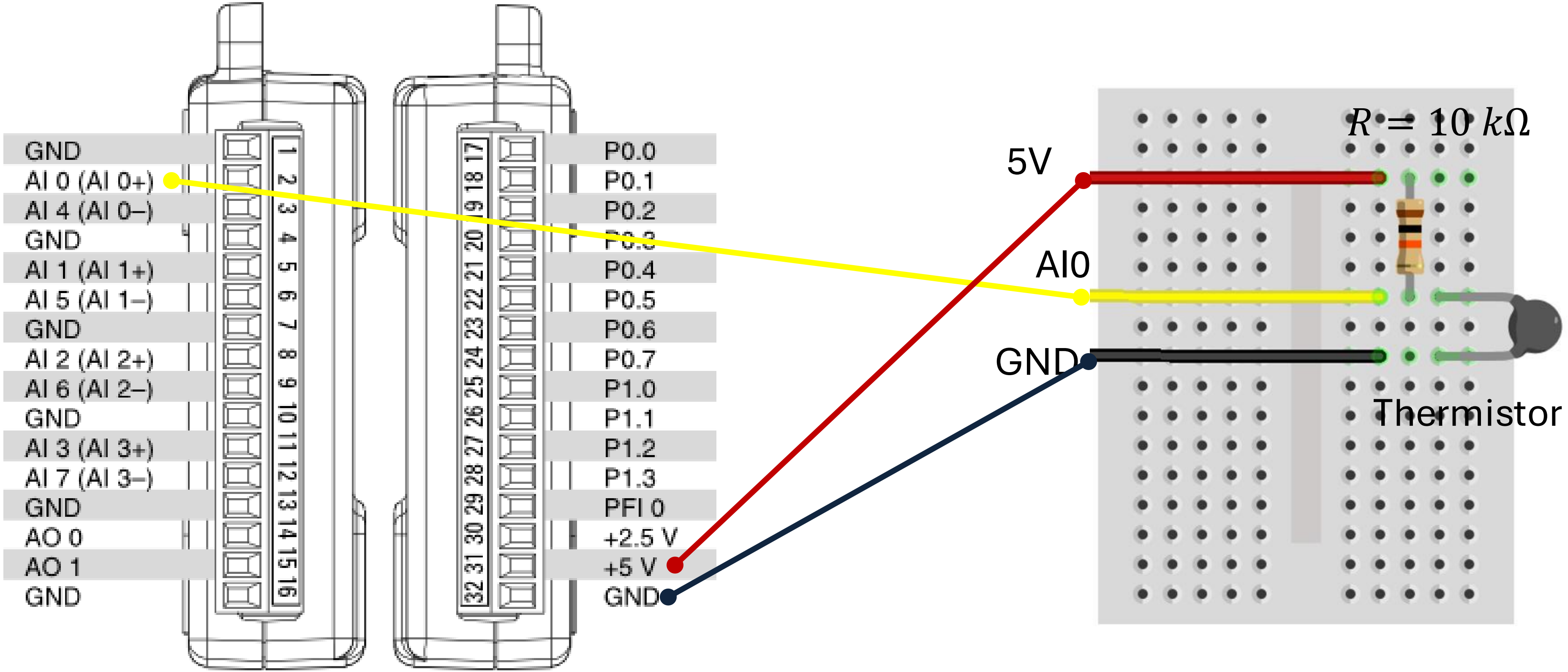
Where the Temperature T_K is in Kelvin
$A, B\ and\ C$ are constants

$$A = 0.001129148$$
$$B = 0.000234125$$
$$C = 0.0000000876741$$

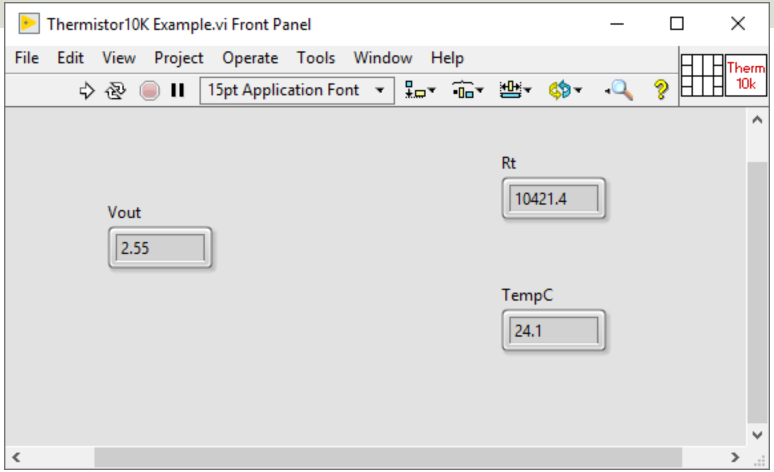
The Temperature in degrees Celsius will then be:

$$T_C = T_K - 273.15$$

Wiring


GND
AI 0 (AI 0+)
AI 4 (AI 0–)
GND
AI 1 (AI 1+)
AI 5 (AI 1–)
GND
AI 2 (AI 2+)
AI 6 (AI 2–)
GND
AI 3 (AI 3+)
AI 7 (AI 3–)
GND
AO 0
AO 1
GND
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
P0.0
P0.1
P0.2
P0.3
P0.4
P0.5
P0.6
P0.7
P1.0
P1.1
P1.2
P1.3
PFI 0
+2.5 V
+5 V
GND
5V
AI0
GND
$R = 10\ k\Omega$
Thermistor

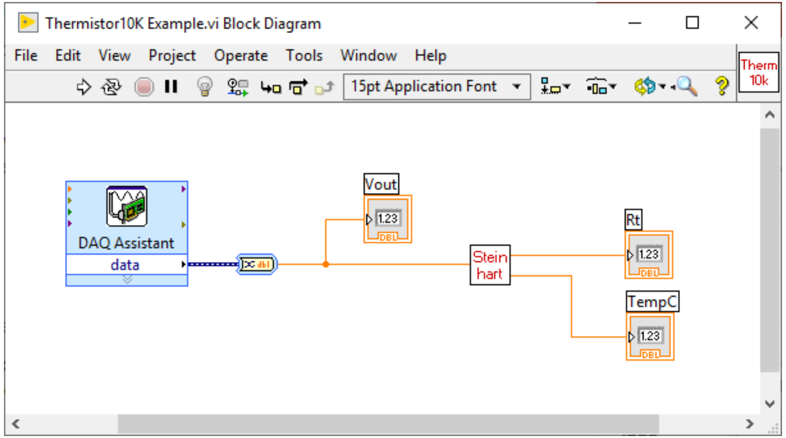
LabVIEW Example

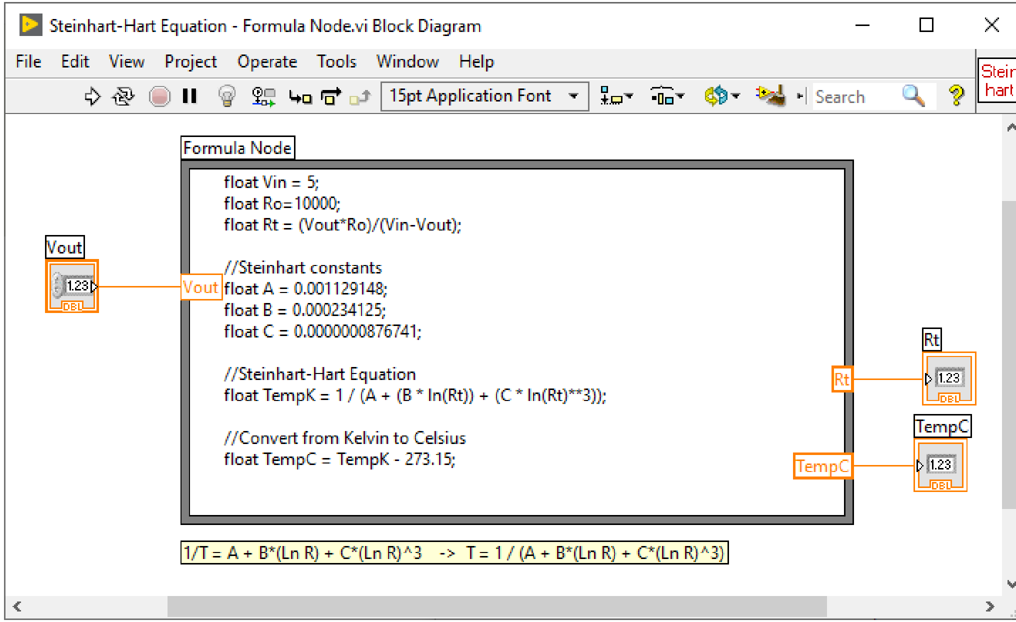

Thermistor10K Example.vi Front Panel
File Edit View Project Operate Tools Window Help
15pt Application Font
Therm 10k

Vout 2.55

Rt 10421.4

TempC 24.1


Thermistor10K Example.vi Block Diagram
File Edit View Project Operate Tools Window Help
15pt Application Font
Therm 10k
DAQ Assistant
data
Vout
1.23
DBL
Stein hart
Rt
TempC


Steinhart-Hart Equation - Formula Node.vi Block Diagram
File Edit View Project Operate Tools Window Help
15pt Application Font
Search
Stein hart
Formula Node
Vout
1.23
DBL
Rt
TempC

```
float Vin = 5;
float Ro=10000;
float Rt = (Vout*Ro)/(Vin-Vout);

//Steinhart constants
float A = 0.001129148;
float B = 0.000234125;
float C = 0.0000000876741;

//Steinhart-Hart Equation
float TempK = 1 / (A + (B * ln(Rt)) + (C * ln(Rt)**3));

//Convert from Kelvin to Celsius
float TempC = TempK - 273.15;
```

1/T = A + B*(Ln R) + C*(Ln R)^3 -> T = 1 / (A + B*(Ln R) + C*(Ln R)^3)

OPC DA

Hans-Petter Halvorsen

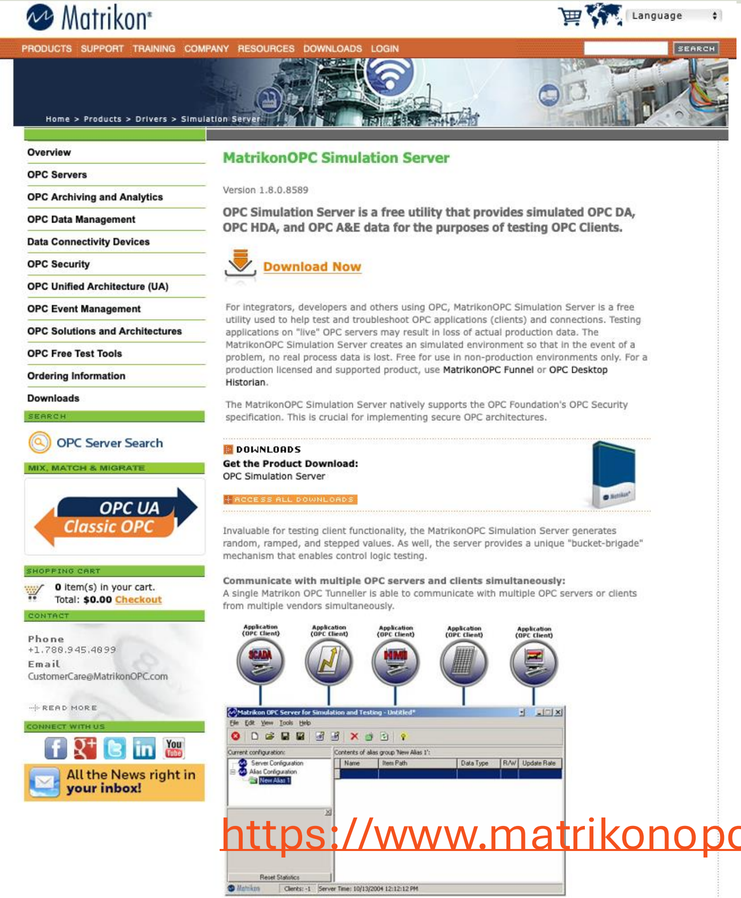
MatrikonOPC Simulation Server

Hans-Petter Halvorsen

Table of Contents

MatrikonOPC Simulation Server

MatrikonOPC Simulation Server is a free utility that provides Simulated OPC DA, OPC HDA, and OPC A&E Data for the Purposes of Testing OPC Clients.



MatrikonOPC Simulation Server

Version 1.8.0.8589

OPC Simulation Server is a free utility that provides simulated OPC DA, OPC HDA, and OPC A&E data for the purposes of testing OPC Clients.

Download Now

For integrators, developers and others using OPC, MatrikonOPC Simulation Server is a free utility used to help test and troubleshoot OPC applications (clients) and connections. Testing applications on "live" OPC servers may result in loss of actual production data. The MatrikonOPC Simulation Server creates an simulated environment so that in the event of a problem, no real process data is lost. Free for use in non-production environments only. For a production licensed and supported product, use MatrikonOPC Funnel or OPC Desktop Historian.

The MatrikonOPC Simulation Server natively supports the OPC Foundation's OPC Security specification. This is crucial for implementing secure OPC architectures.

DOWNLOADS

Get the Product Download:
OPC Simulation Server

ACCESS ALL DOWNLOADS

Invaluable for testing client functionality, the MatrikonOPC Simulation Server generates random, ramped, and stepped values. As well, the server provides a unique "bucket-brigade" mechanism that enables control logic testing.

Communicate with multiple OPC servers and clients simultaneously:
A single Matrikon OPC Tunneller is able to communicate with multiple OPC servers or clients from multiple vendors simultaneously.

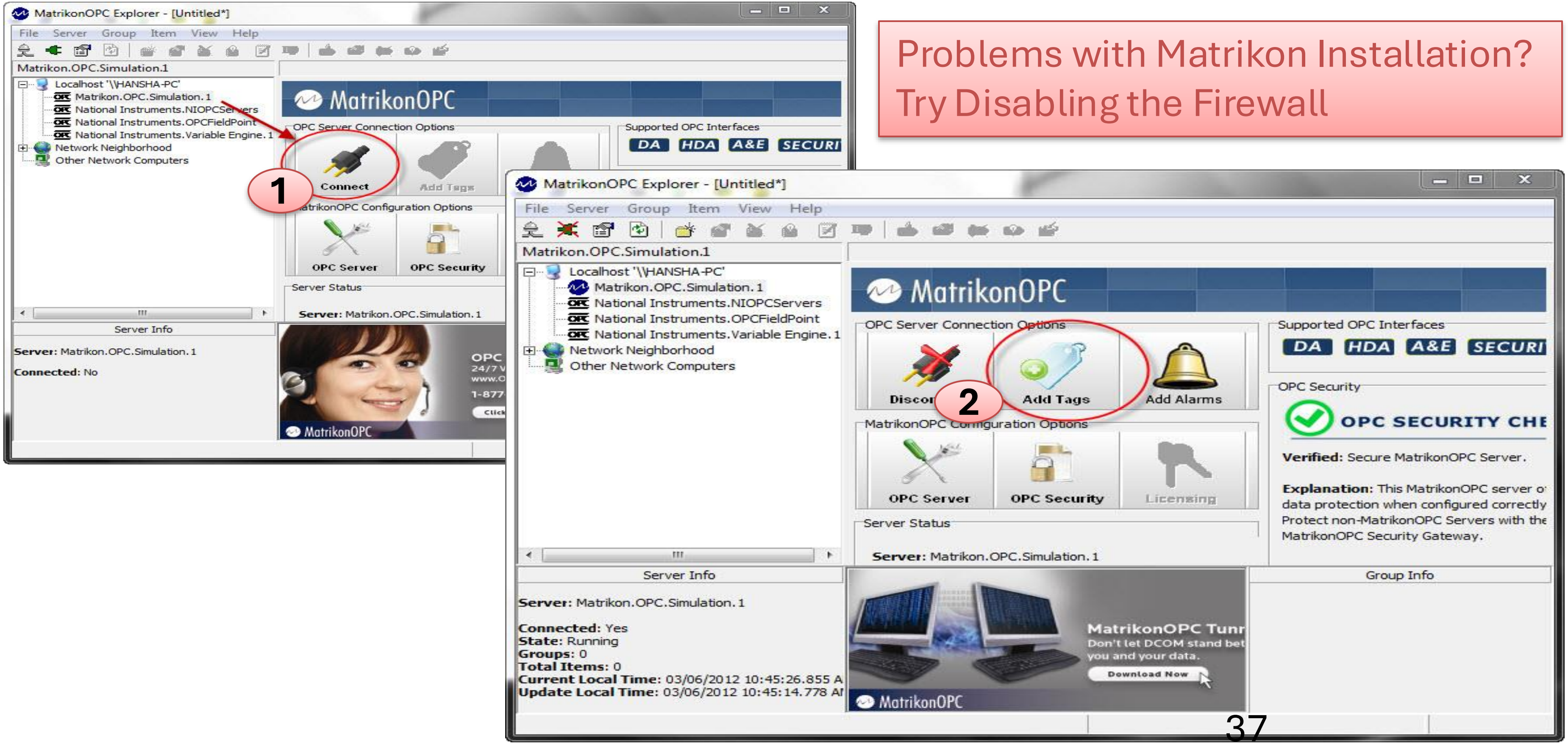
0 item(s) in your cart.
Total: **$0.00** Checkout

Phone
+1.780.945.4099
Email
CustomerCare@MatrikonOPC.com

https://www.matrikonopc.com/products/opc-drivers/opc-simulation-server.aspx

Matrikon OPC Explorer – Connect to Server

Problems with Matrikon Installation?
Try Disabling the Firewall


MatrikonOPC Explorer - [Untitled*]
File Server Group Item View Help
Matrikon.OPC.Simulation.1
Localhost '\\HANSHA-PC'
Matrikon.OPC.Simulation.1
National Instruments.NIOPCServers
National Instruments.OPCFieldPoint
National Instruments.Variable Engine.1
Network Neighborhood
Other Network Computers
MatrikonOPC
OPC Server Connection Options
Connect
Add Tags
1
MatrikonOPC Configuration Options
OPC Server
OPC Security
Supported OPC Interfaces
DA HDA A&E SECURI
Server Status
Server Info
MatrikonOPC Explorer - [Untitled*]
Discor
2
Add Tags
Add Alarms
Licensing
OPC Security
OPC SECURITY CHE
Group Info
MatrikonOPC Tunr
Don't let DCOM stand bet you and your data.
Download Now

Server: Matrikon.OPC.Simulation.1

Connected: No

Server: Matrikon.OPC.Simulation.1

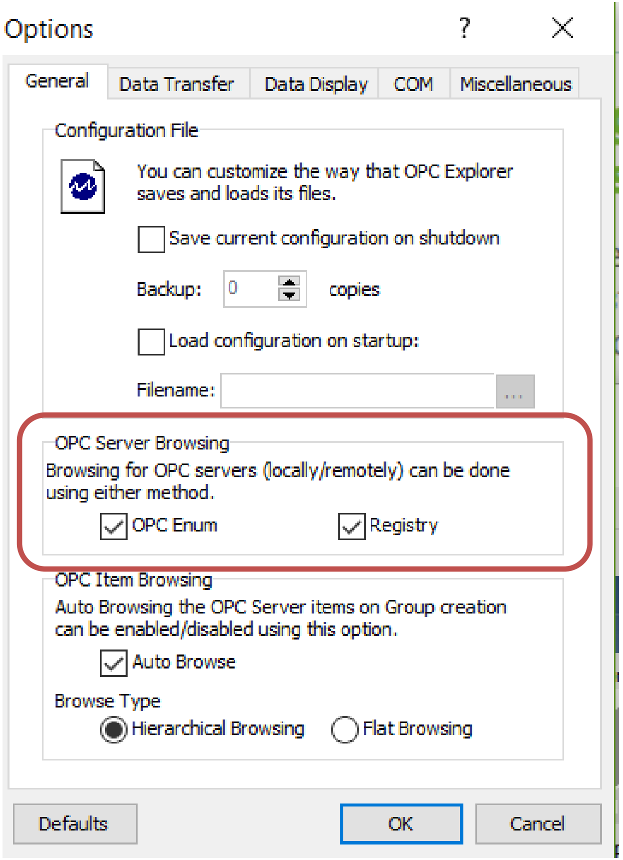
Connected: Yes
State: Running
Groups: 0
Total Items: 0
Current Local Time: 03/06/2012 10:45:26.855 A
Update Local Time: 03/06/2012 10:45:14.778 AI

Verified: Secure MatrikonOPC Server.

Explanation: This MatrikonOPC server o data protection when configured correctly Protect non-MatrikonOPC Servers with the MatrikonOPC Security Gateway.

MatrikonOPC Explorer Troubleshooting

- **Problem**: “When starting MatrikonOPC Explorer, I get an error indicating there are no servers installed”.
- **Solution**:
  - In OPC Explorer select View ->Options from the menu bar.
  - On the General Tab select both “OPCEnum” and “Registry” as the Browse Methods.
  - Exit OPC Explorer and restart.
  - Upon restarting, you should see a listing of locally registered OPC servers.
  - If this still does not work, remove OPCEnum as a browse method and restart.



Configuration File

You can customize the way that OPC Explorer saves and loads its files.

☐ Save current configuration on shutdown

Backup: 0 copies

☐ Load configuration on startup:

Filename:

OPC Server Browsing

Browsing for OPC servers (locally/remotely) can be done using either method.

☑ OPC Enum ☑ Registry

OPC Item Browsing

Auto Browsing the OPC Server items on Group creation can be enabled/disabled using this option.

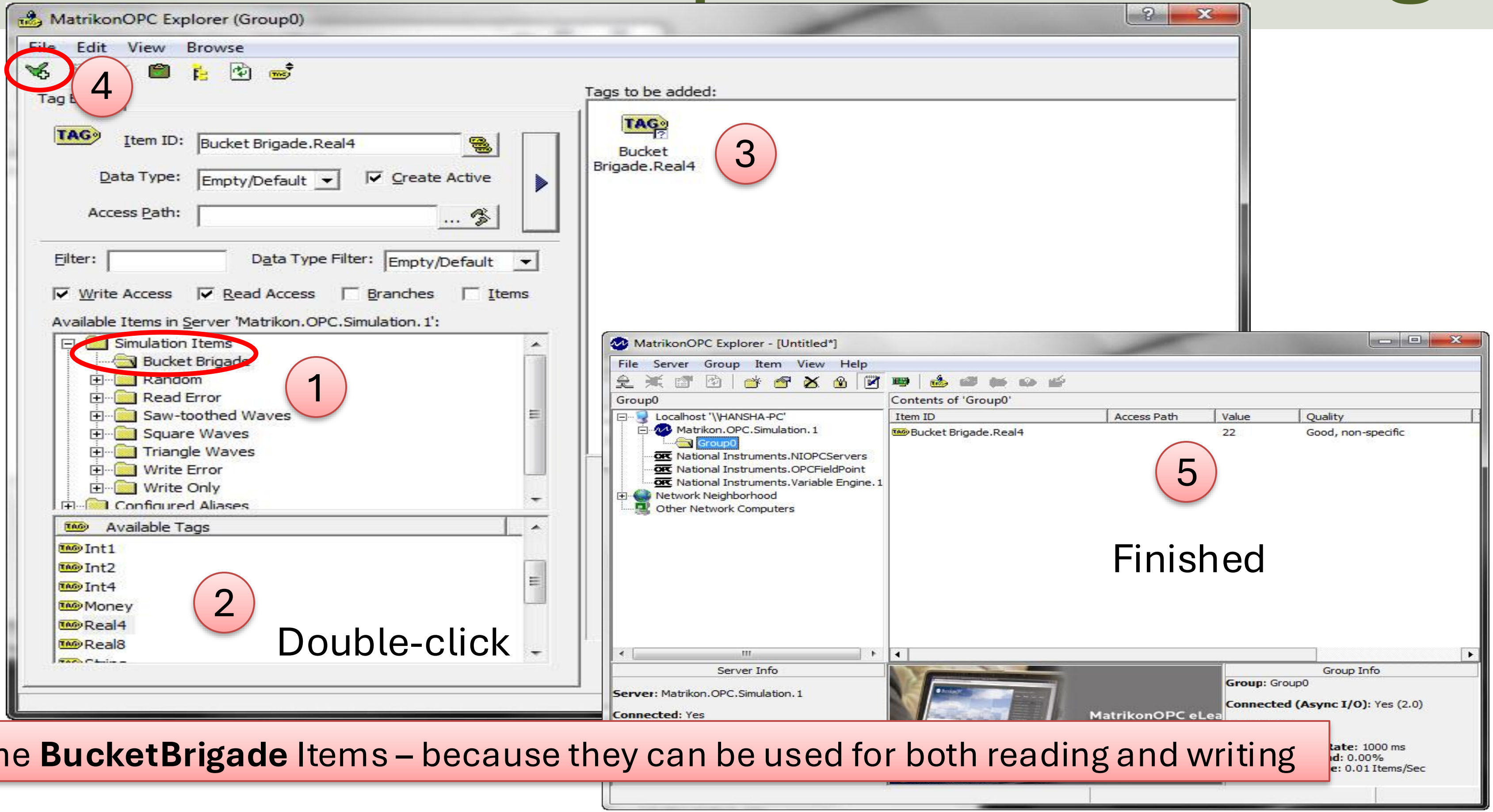
☑ Auto Browse

Browse Type

● Hierarchical Browsing ○ Flat Browsing

Matrikon OPC Explorer - Add Tag

Add Tags


MatrikonOPC Explorer (Group0)
File Edit View Browse
4
Tag E
1
2
3
5
MatrikonOPC Explorer - [Untitled*]
File Server Group Item View Help
Server Info
Group Info
MatrikonOPC eLea

Item ID: Bucket Brigade.Real4

Data Type: Empty/Default ☑ Create Active

Access Path:

Filter: Data Type Filter: Empty/Default

☑ Write Access ☑ Read Access □ Branches □ Items

Available Items in Server 'Matrikon.OPC.Simulation.1':

- Simulation Items
  - Bucket Brigade
  - Random
  - Read Error
  - Saw-toothed Waves
  - Square Waves
  - Triangle Waves
  - Write Error
  - Write Only
- Configured Aliases

Available Tags

- Int1
- Int2
- Int4
- Money
- Real4
- Real8

Double-click

Tags to be added:

Bucket Brigade.Real4

Group0

- Localhost '\\HANSHA-PC'
  - Matrikon.OPC.Simulation.1
    - Group0
  - National Instruments.NIOPCServers
  - National Instruments.OPCFieldPoint
  - National Instruments.Variable Engine.1
- Network Neighborhood
- Other Network Computers

Contents of 'Group0'

Item ID	Access Path	Value	Quality
Bucket Brigade.Real4		22	Good, non-specific

Finished

Server: Matrikon.OPC.Simulation.1

Connected: Yes

Group: Group0

Connected (Async I/O): Yes (2.0)

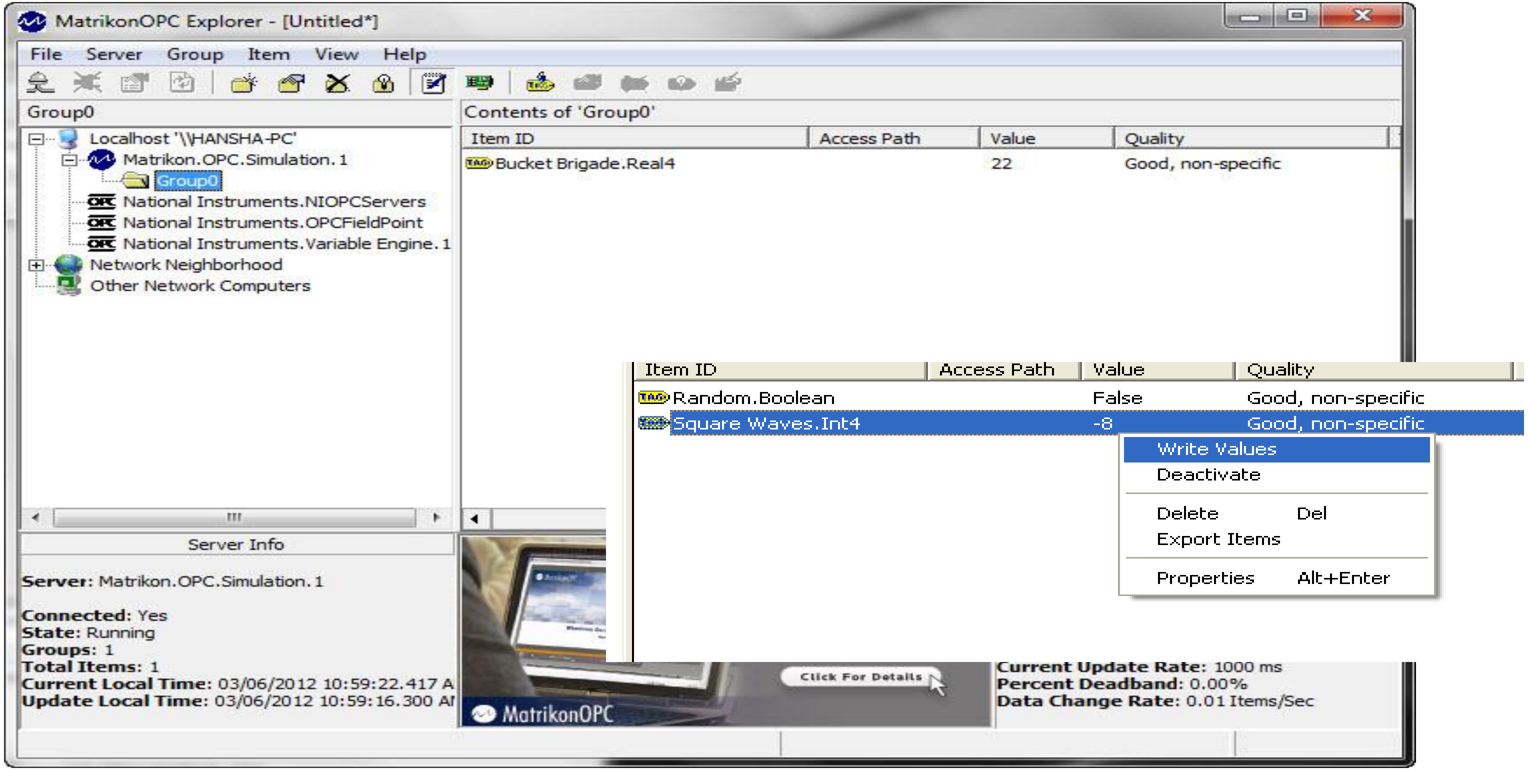
ate: 1000 ms

d: 0.00%

e: 0.01 Items/Sec

Tip! Use the **BucketBrigade** Items – because they can be used for both reading and writing

MatrikonOPC Explorer (OPC Client)



Item ID	Access Path	Value	Quality
Bucket Brigade.Real4		22	Good, non-specific

Item ID	Access Path	Value	Quality
Random.Boolean		False	Good, non-specific
Square Waves.Int4		-8	Good, non-specific

Server: Matrikon.OPC.Simulation.1

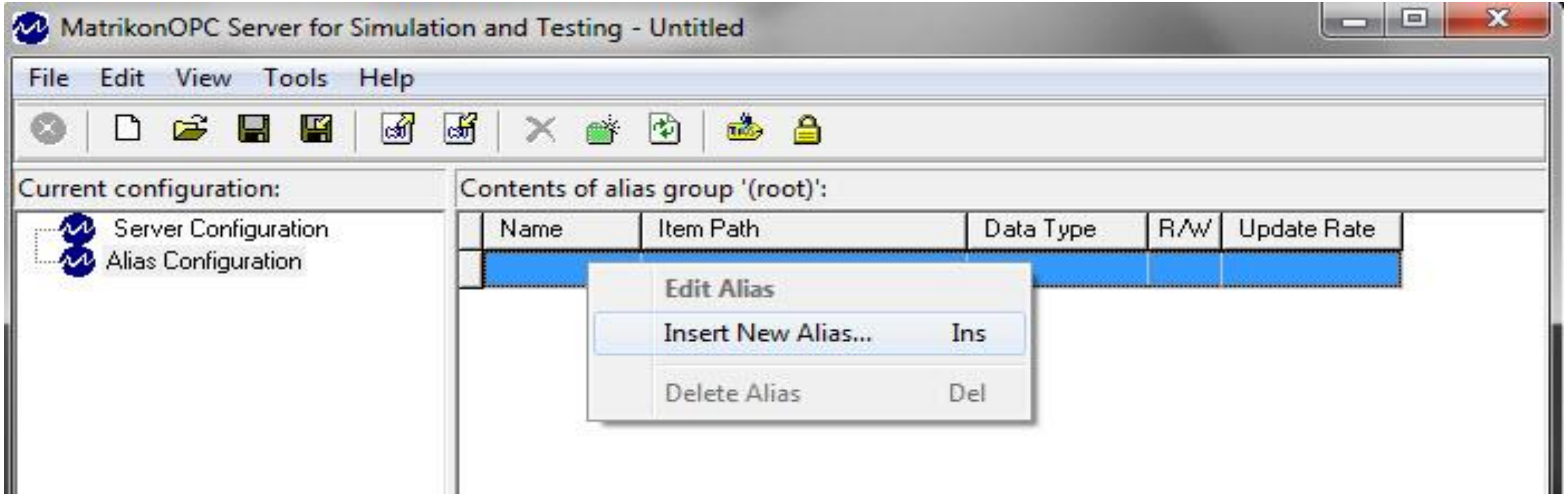
Connected: Yes
State: Running
Groups: 1
Total Items: 1
Current Local Time: 03/06/2012 10:59:22.417 A
Update Local Time: 03/06/2012 10:59:16.300 AI

Current Update Rate: 1000 ms
Percent Deadband: 0.00%
Data Change Rate: 0.01 Items/Sec

The MatrikonOPC Explorer is useful for testing. You can use it for writing and reading OPC Tags

Aliases

In the “Matrikon OPCServer for Simulation” you can create Aliases. Aliases is handy when you want to describe your OPC items using more realistic names.



Tip: You can create an alias called, e.g., “Temperature” which you can use instead of the real OPC Tag Name

https://www.halvorsen.blog

OPC DA in LabVIEW

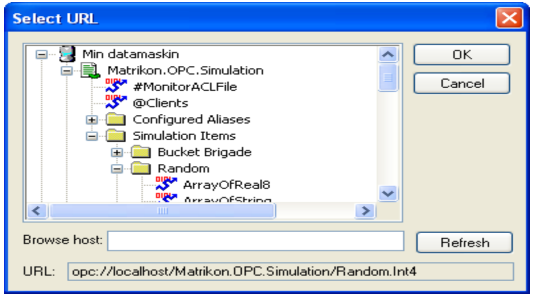
Hans-Petter Halvorsen

Table of Contents

OPC DA in LabVIEW

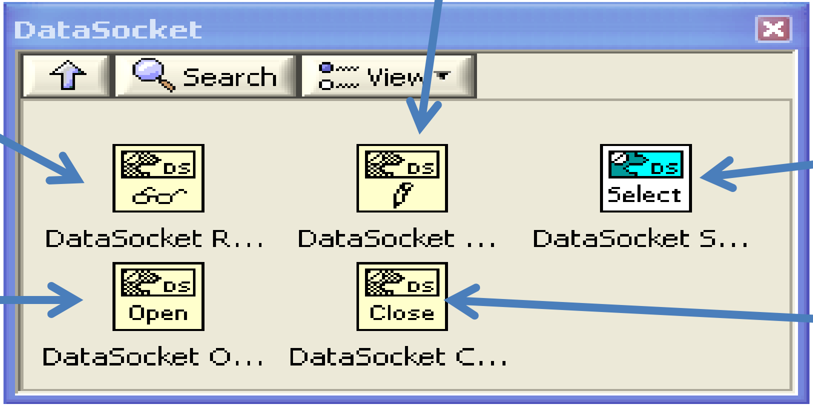
You can use LabVIEW as an OPC client by connecting to an OPC server through a **DataSocket** connection.

The **DataSocket** palette in LabVIEW:



Browse host:

URL: opc://localhost/Matrikon.OPC.Simulation/Random.Int4



Write Data to OPC

Read Data from OPC

Browse OPC Servers and OPC Items

Open Connection to OPC Server

Close Connection to OPC Server

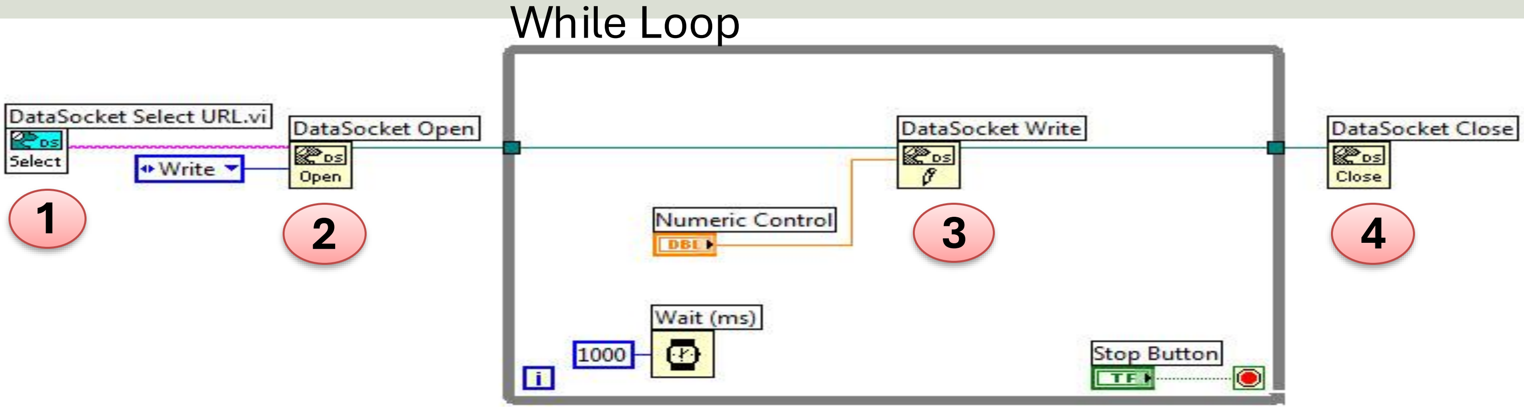
Note! Make sure to use LabVIEW 32bit version (even if you have 64bit operating system) because the DataSocket feature is only supported by the 32bit version of LabVIEW.

LabVIEW OPC DA - Write

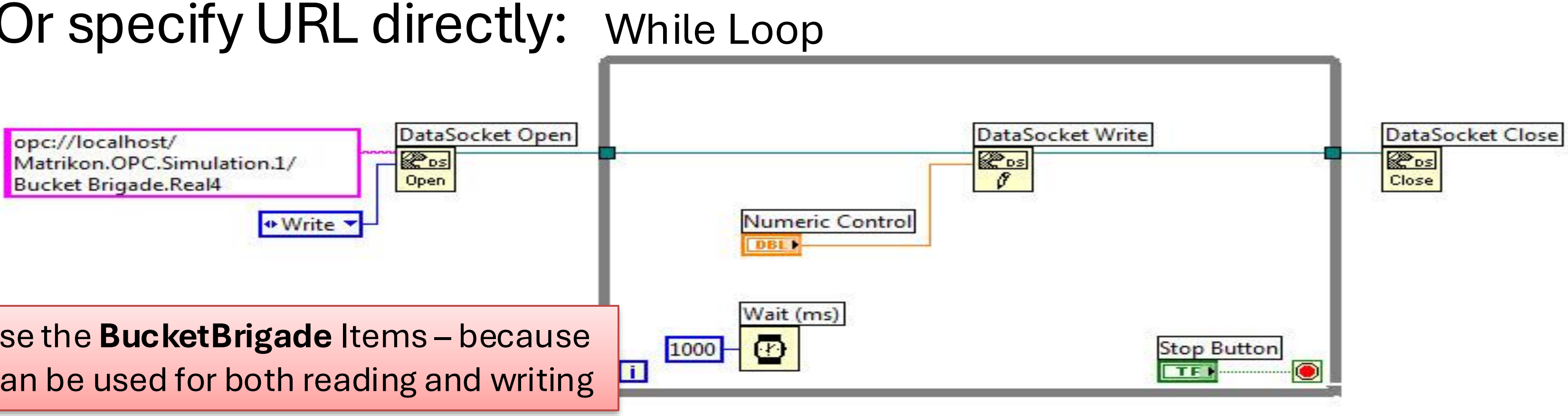
Hans-Petter Halvorsen

Table of Contents

LabVIEW OPC DA - Write



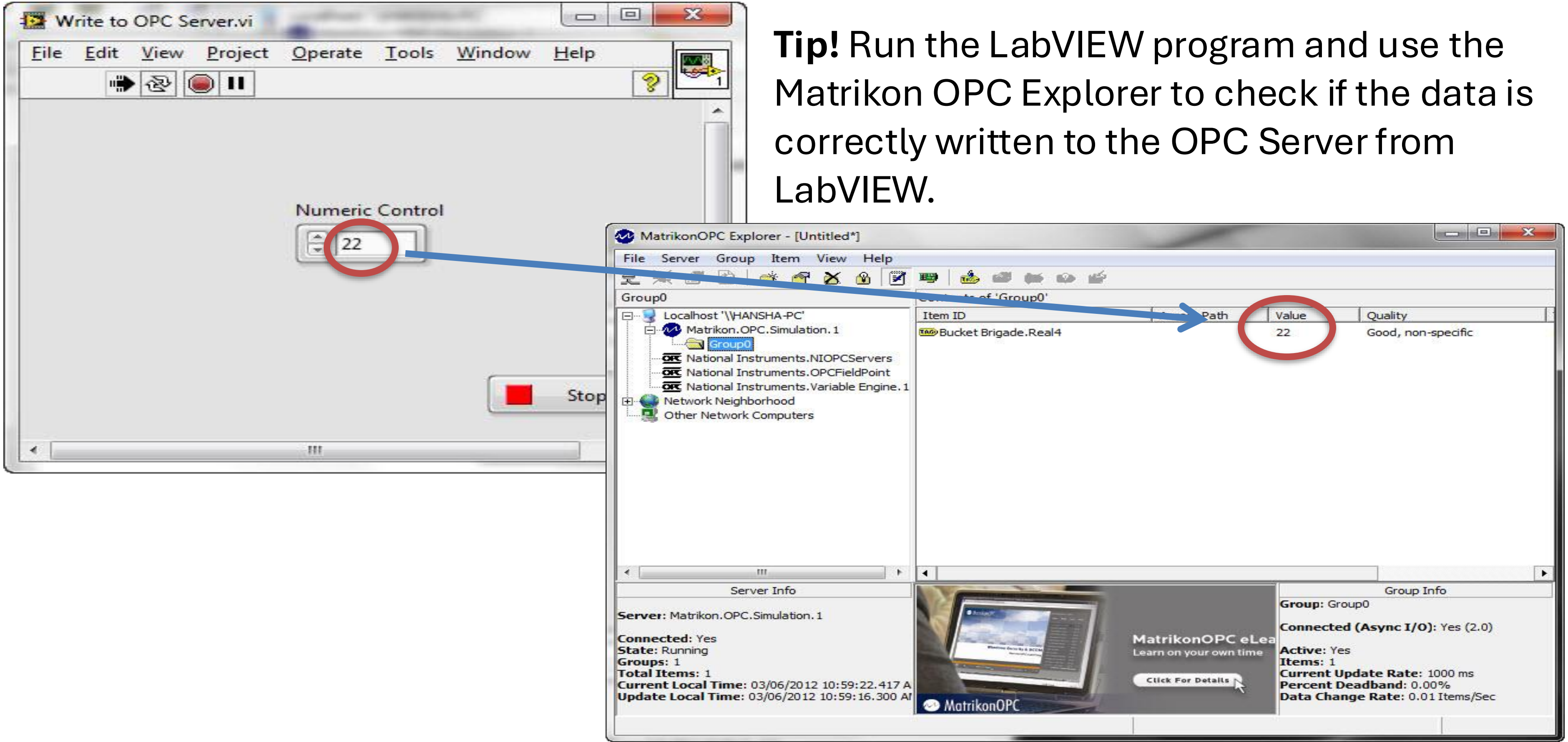
Or specify URL directly:



Tip! Use the **BucketBrigade** Items – because they can be used for both reading and writing

Use OPC Explorer to Check Communication

Tip! Run the LabVIEW program and use the Matrikon OPC Explorer to check if the data is correctly written to the OPC Server from LabVIEW.


Write to OPC Server.vi
File Edit View Project Operate Tools Window Help
Stop
MatrikonOPC Explorer - [Untitled*]
File Server Group Item View Help
Group0
Localhost '\\HANSHA-PC'
Matrikon.OPC.Simulation.1
Group0
National Instruments.NIOPCServers
National Instruments.OPCFieldPoint
National Instruments.Variable Engine.1
Network Neighborhood
Other Network Computers
MatrikonOPC eLea
Learn on your own time
Click For Details
MatrikonOPC

Numeric Control: 22

Item ID	Path	Value	Quality
Bucket Brigade.Real4		22	Good, non-specific

Server Info

Server: Matrikon.OPC.Simulation.1

Connected: Yes
State: Running
Groups: 1
Total Items: 1
Current Local Time: 03/06/2012 10:59:22.417 A
Update Local Time: 03/06/2012 10:59:16.300 AI

Group Info

Group: Group0

Connected (Async I/O): Yes (2.0)

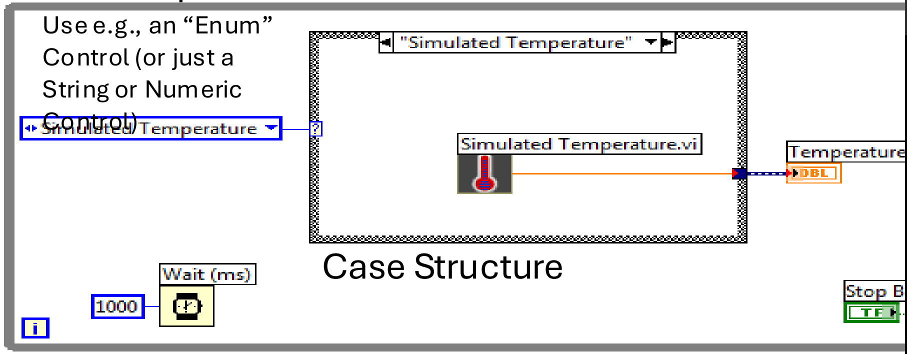
Active: Yes
Items: 1
Current Update Rate: 1000 ms
Percent Deadband: 0.00%
Data Change Rate: 0.01 Items/Sec

Temperature Simulator Example

❗If you do not have the TC01 device available, you can create and use a simple “Temperature Simulator” instead

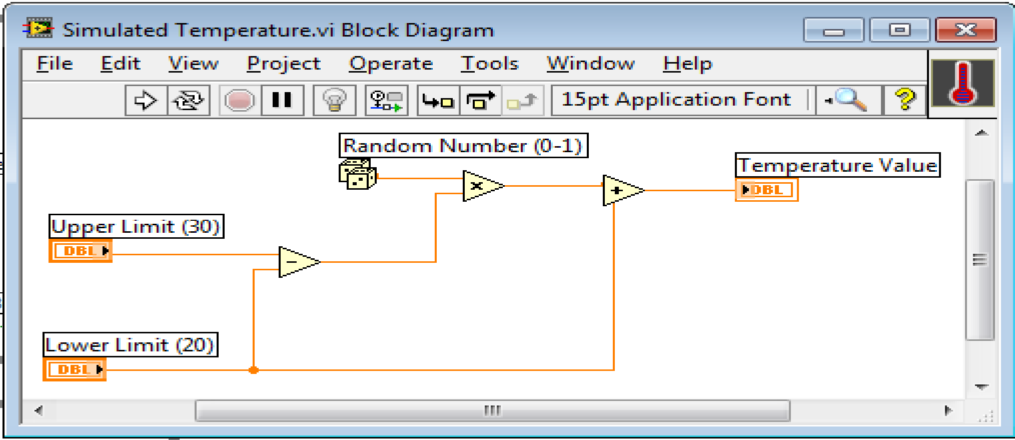
A simple SubVI that simulates a Temperature value using a Random Generator:

While Loop

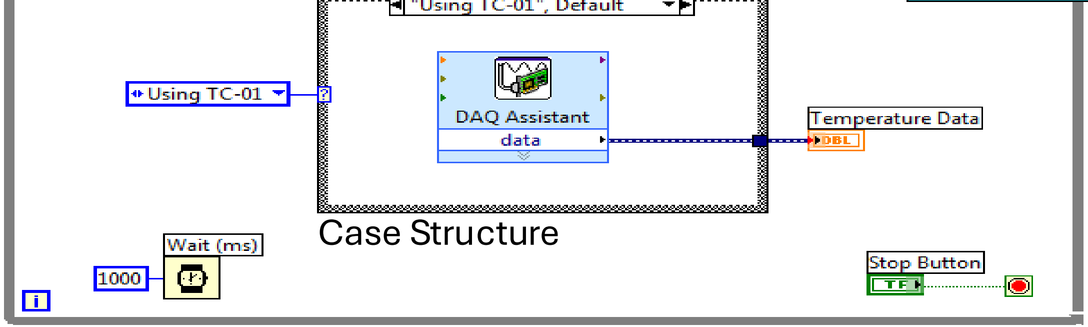


Use e.g., an “Enum” Control (or just a String or Numeric Control)

Case Structure



While Loop



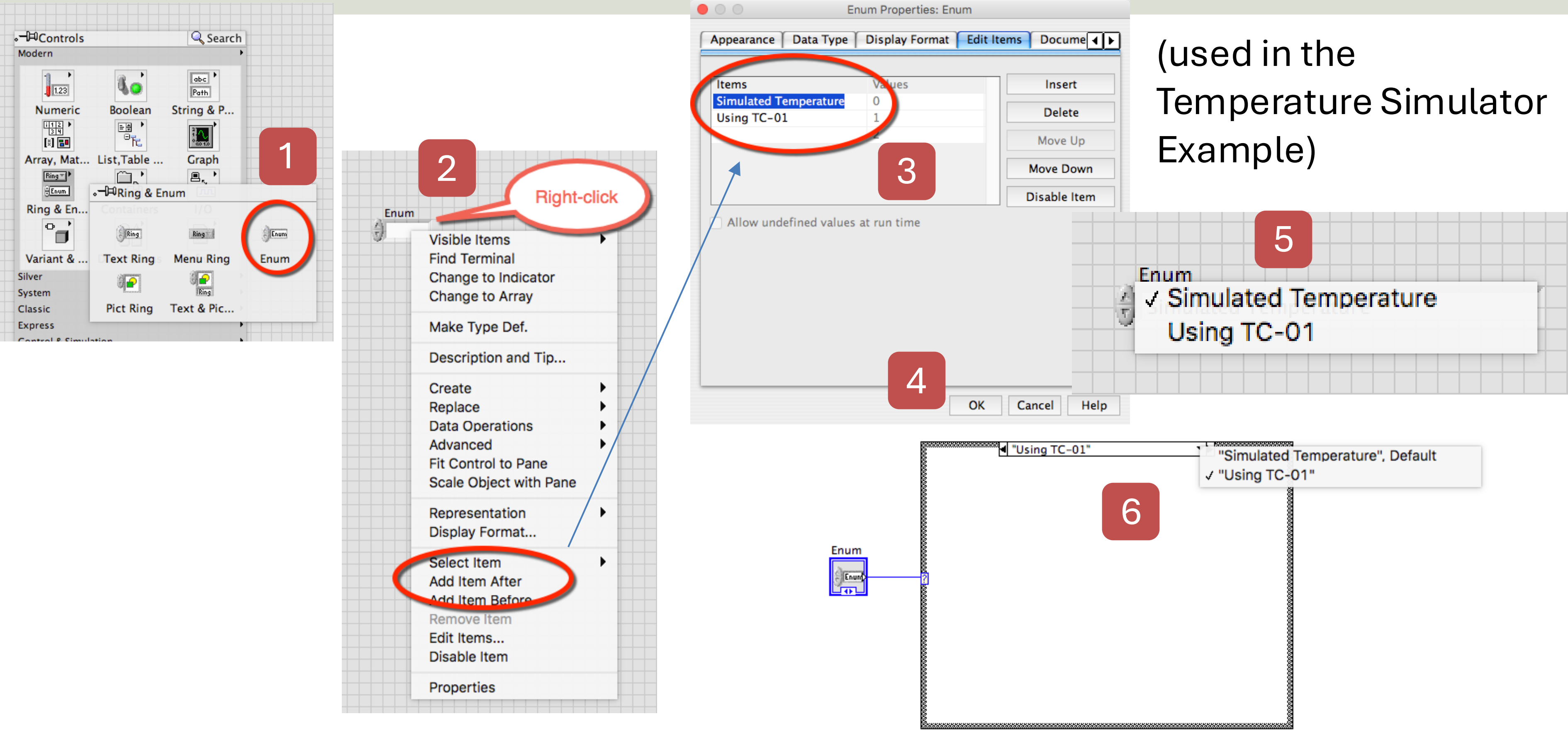
Case Structure

In this way you can easily switch between the real Temperature sensor (TC-01) and the Simulator.

Here you just see a simple example - feel free to create a more realistic Temperature Simulator

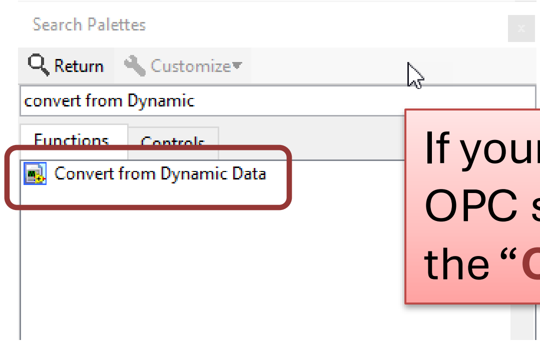
How to create an “Enum” in LabVIEW

(used in the Temperature Simulator Example)



Items	Values
Simulated Temperature	0
Using TC-01	1

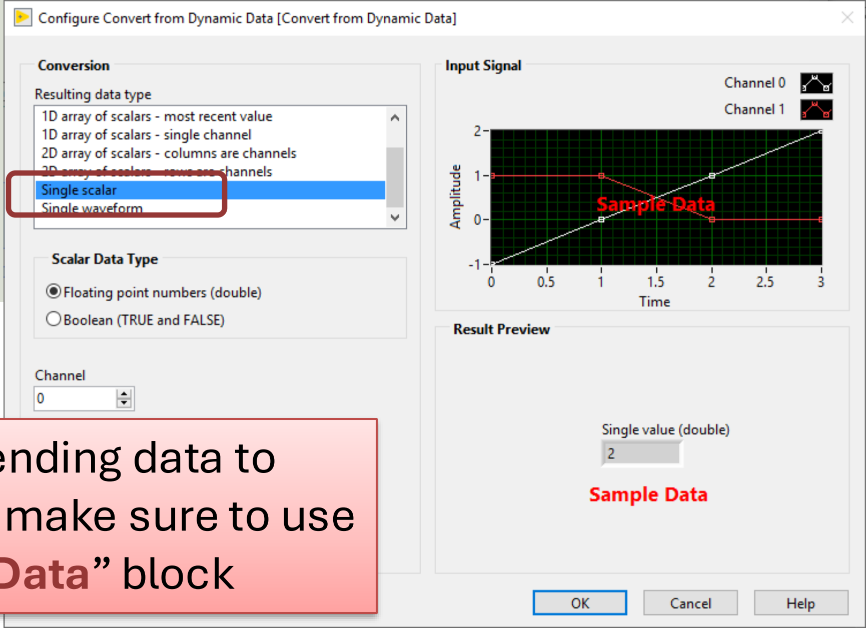
Convert from Dynamic Data



Search Palettes

convert from Dynamic

Convert from Dynamic Data



Configure Convert from Dynamic Data [Convert from Dynamic Data]

Conversion

Resulting data type

- 1D array of scalars - most recent value
- 1D array of scalars - single channel
- 2D array of scalars - columns are channels
- 2D array of scalars - rows are channels
- Single scalar
- Single waveform

Scalar Data Type

● Floating point numbers (double)

○ Boolean (TRUE and FALSE)

Channel: 0

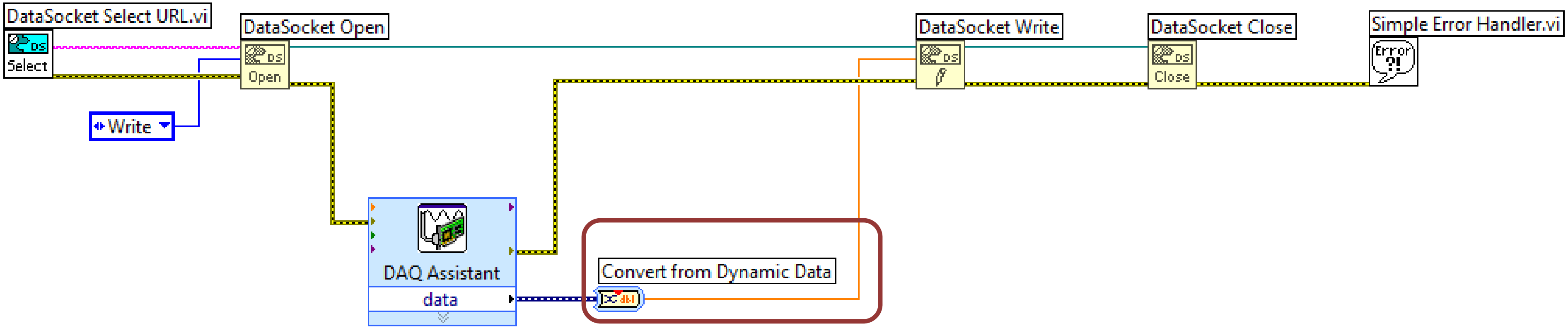
Input Signal

Result Preview

Single value (double): 2

Sample Data

If your Program crash when sending data to OPC server from DAQ device, make sure to use the "**Convert from Dynamic Data**" block



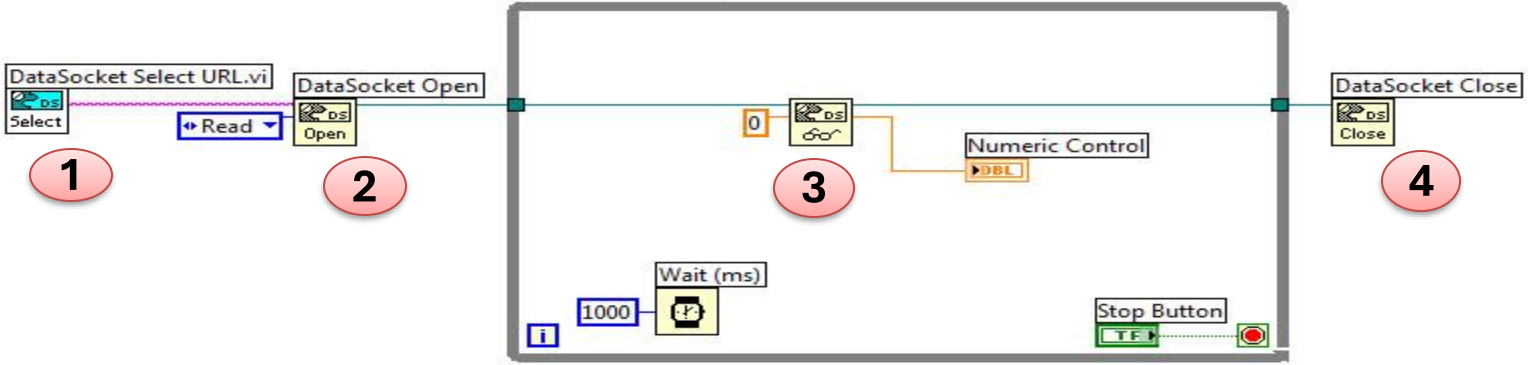
LabVIEW OPC DA - Read

Hans-Petter Halvorsen

Table of Contents

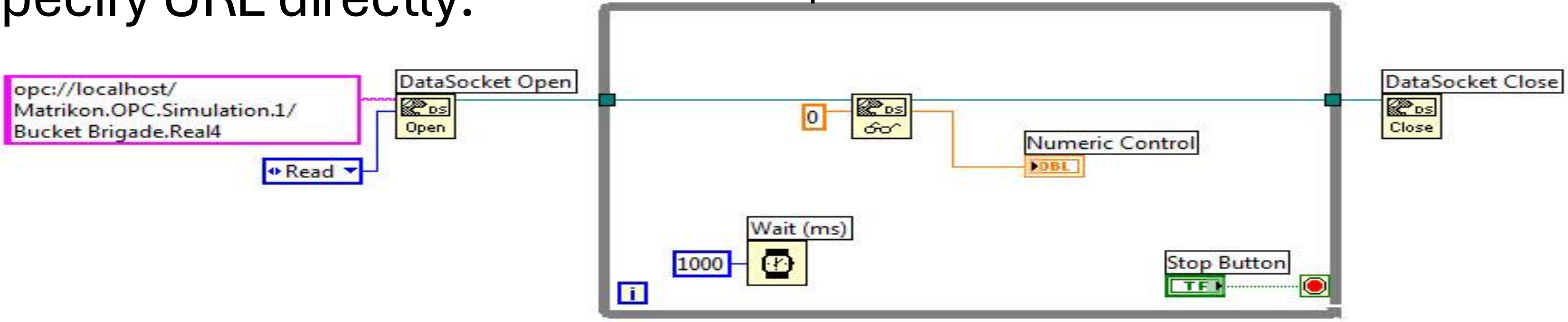
Read from OPC Server using LabVIEW

While Loop



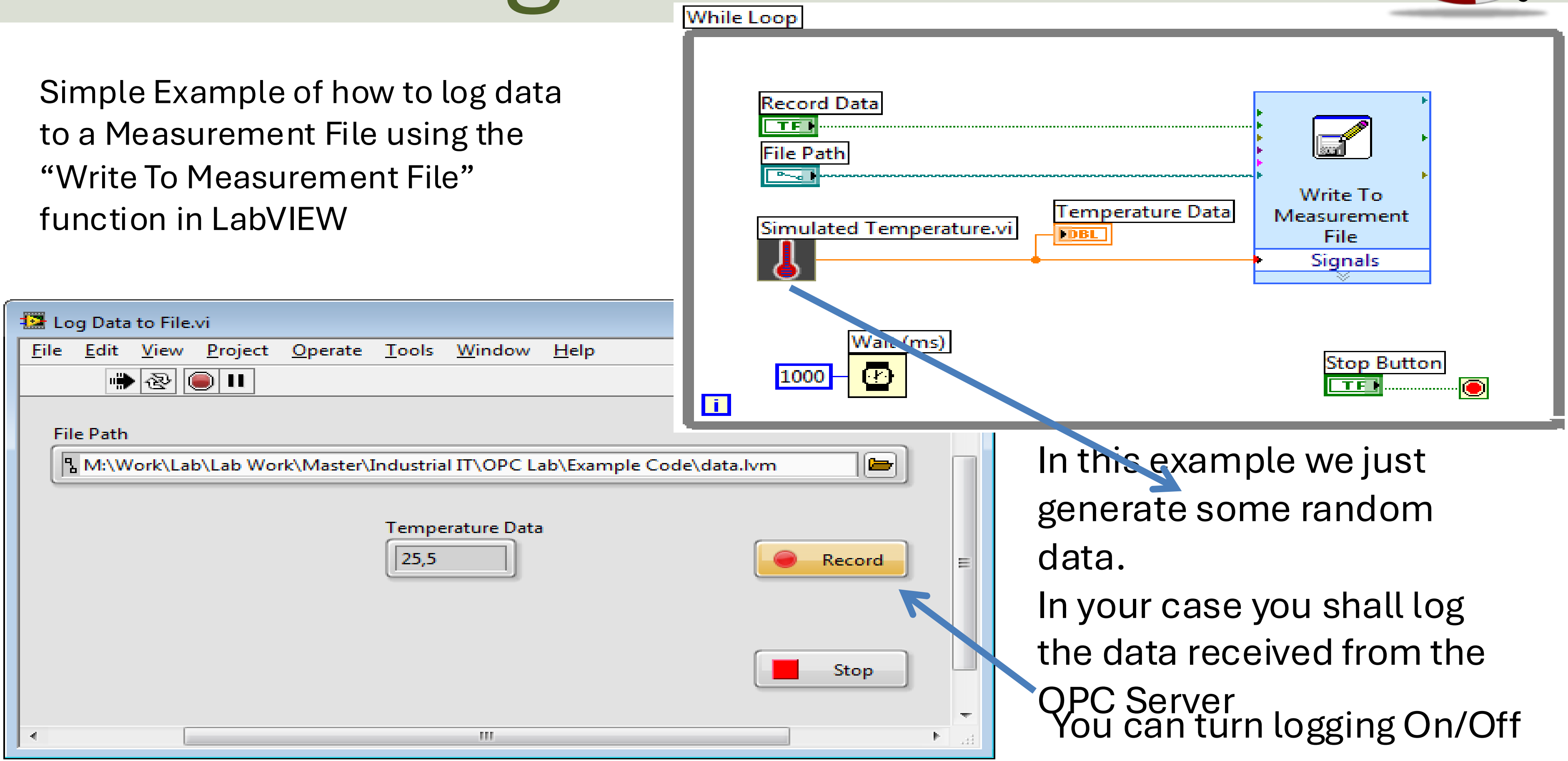
Or specify URL directly:

While Loop



Log Data to File

Simple Example of how to log data to a Measurement File using the “Write To Measurement File” function in LabVIEW



File Path
M:\Work\Lab\Lab Work\Master\Industrial IT\OPC Lab\Example Code\data.lvm

Temperature Data
25,5

Record

Stop

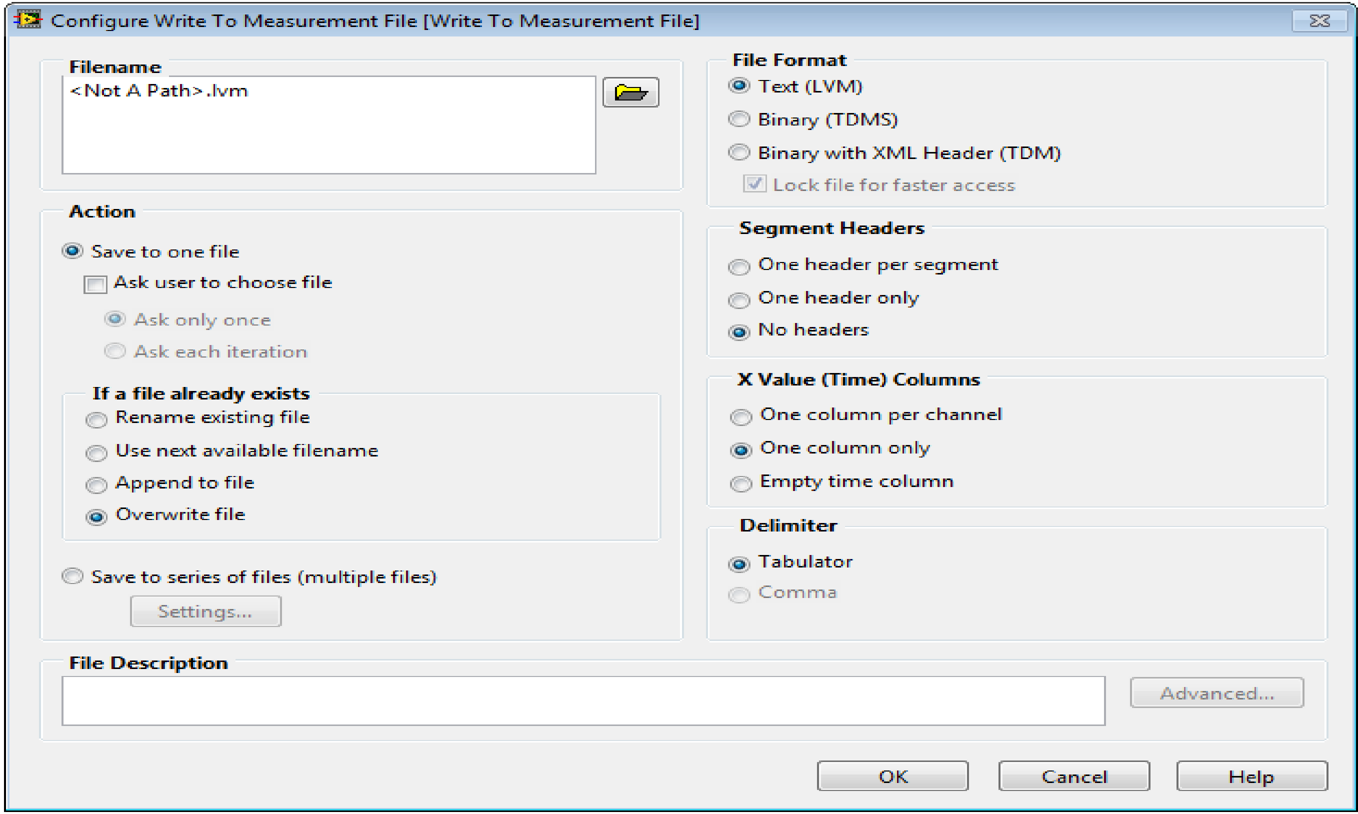
In this example we just generate some random data.

In your case you shall log the data received from the OPC Server

You can turn logging On/Off

Log Data to File - Properties

Recommended Settings in the **Properties** Window (Right-click on the “Write To Measurement” File icon):



Configure Write To Measurement File [Write To Measurement File]

Filename

<Not A Path>.lvm

Action

● Save to one file

□ Ask user to choose file

● Ask only once

○ Ask each iteration

If a file already exists

○ Rename existing file

○ Use next available filename

○ Append to file

● Overwrite file

○ Save to series of files (multiple files)

Settings...

File Format

● Text (LVM)

○ Binary (TDMS)

○ Binary with XML Header (TDM)

☑ Lock file for faster access

Segment Headers

○ One header per segment

○ One header only

● No headers

X Value (Time) Columns

○ One column per channel

● One column only

○ Empty time column

Delimiter

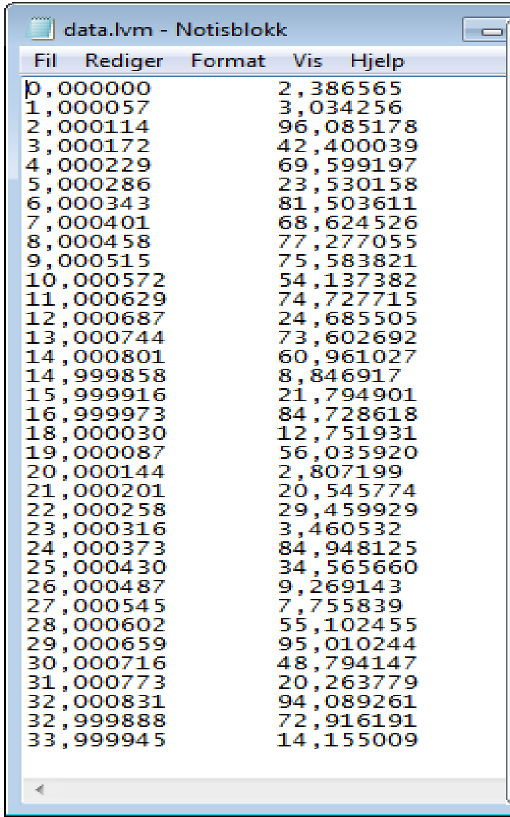
● Tabulator

○ Comma

File Description

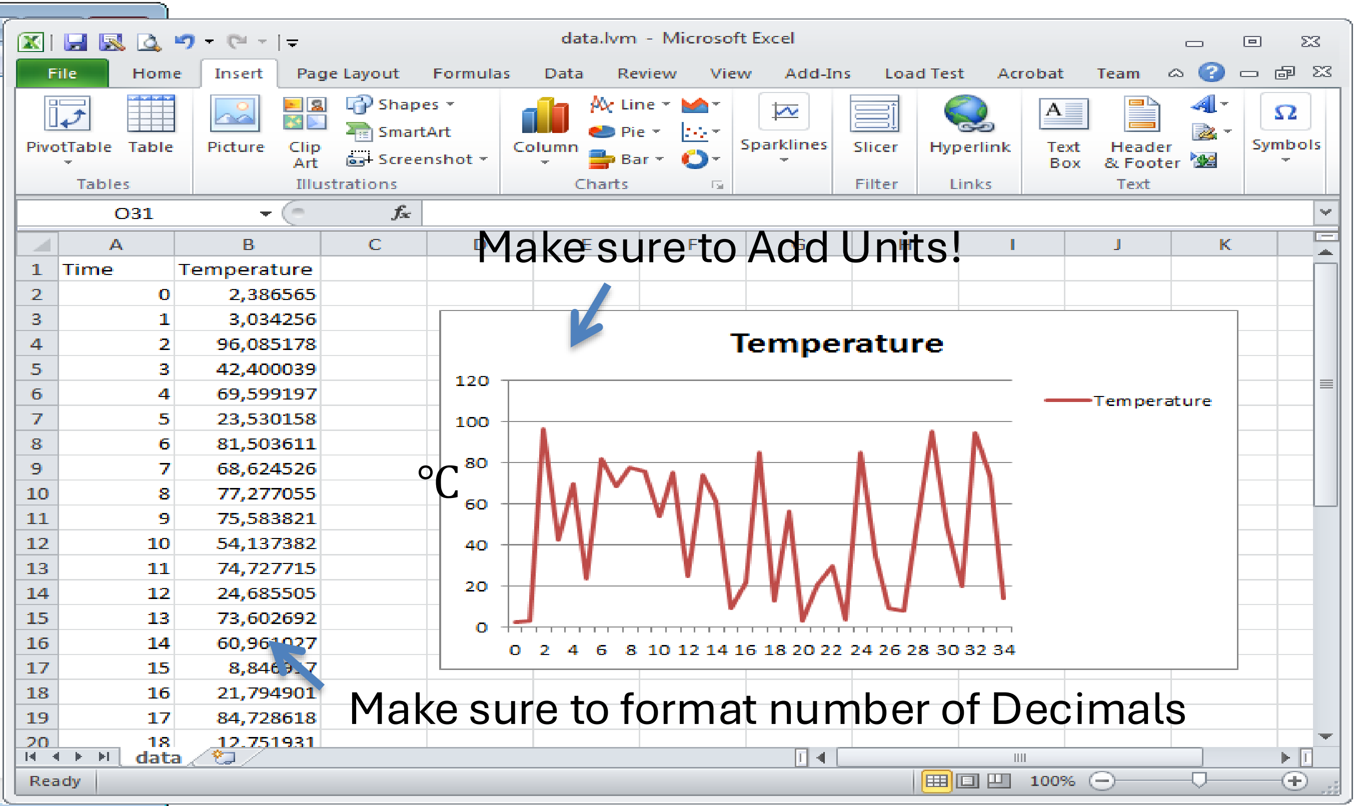
Measurement File – Data Visualization

Open the File with Logged Data in e.g., **Notepad**:



```
0,000000	2,386565
1,000057	3,034256
2,000114	96,085178
3,000172	42,400039
4,000229	69,599197
5,000286	23,530158
6,000343	81,503611
7,000401	68,624526
8,000458	77,277055
9,000515	75,583821
10,000572	54,137382
11,000629	74,727715
12,000687	24,685505
13,000744	73,602692
14,000801	60,961027
14,999858	8,846917
15,999916	21,794901
16,999973	84,728618
18,000030	12,751931
19,000087	56,035920
20,000144	2,807199
21,000201	20,545774
22,000258	29,459929
23,000316	3,460532
24,000373	84,948125
25,000430	34,565660
26,000487	9,269143
27,000545	7,755839
28,000602	55,102455
29,000659	95,010244
30,000716	48,794147
31,000773	20,263779
32,000831	94,089261
32,999888	72,916191
33,999945	14,155009
```

Here we see an example where we have opened the File with Logged Data in **MS Excel** and created a Chart



	A	B
1	Time	Temperature
2	0	2,386565
3	1	3,034256
4	2	96,085178
5	3	42,400039
6	4	69,599197
7	5	23,530158
8	6	81,503611
9	7	68,624526
10	8	77,277055
11	9	75,583821
12	10	54,137382
13	11	74,727715
14	12	24,685505
15	13	73,602692
16	14	60,961027
17	15	8,846917
18	16	21,794901
19	17	84,728618
20	18	12,751931

Make sure to Add Units!

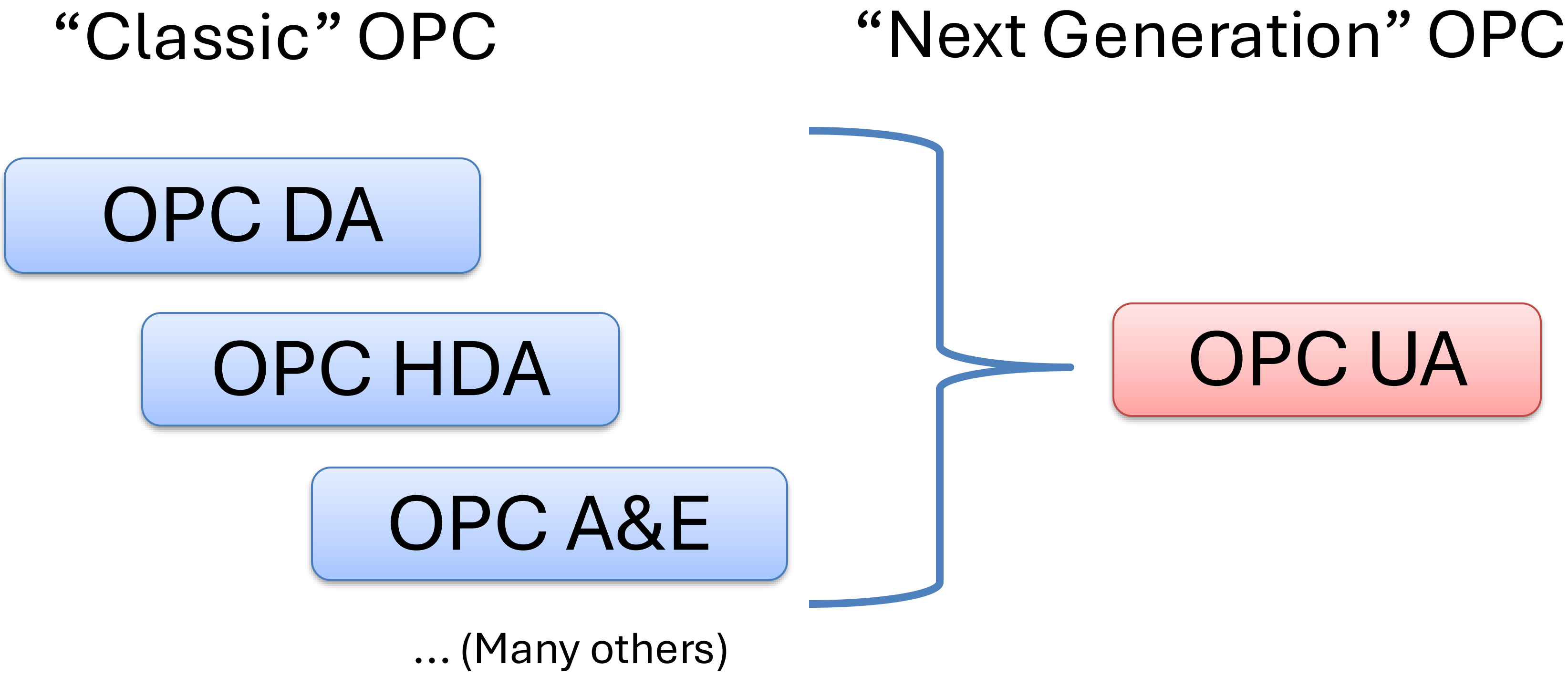
Make sure to format number of Decimals

OPC UA

OPC Unified Architecture

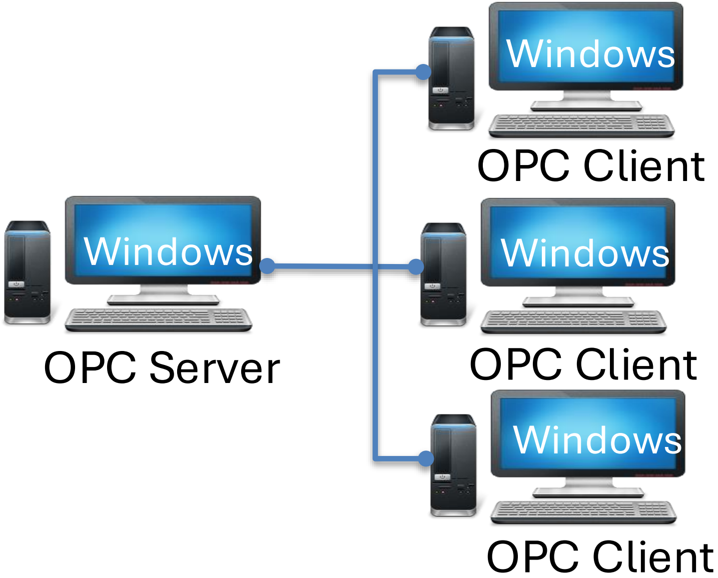
Hans-Petter Halvorsen

“Next Generation” OPC

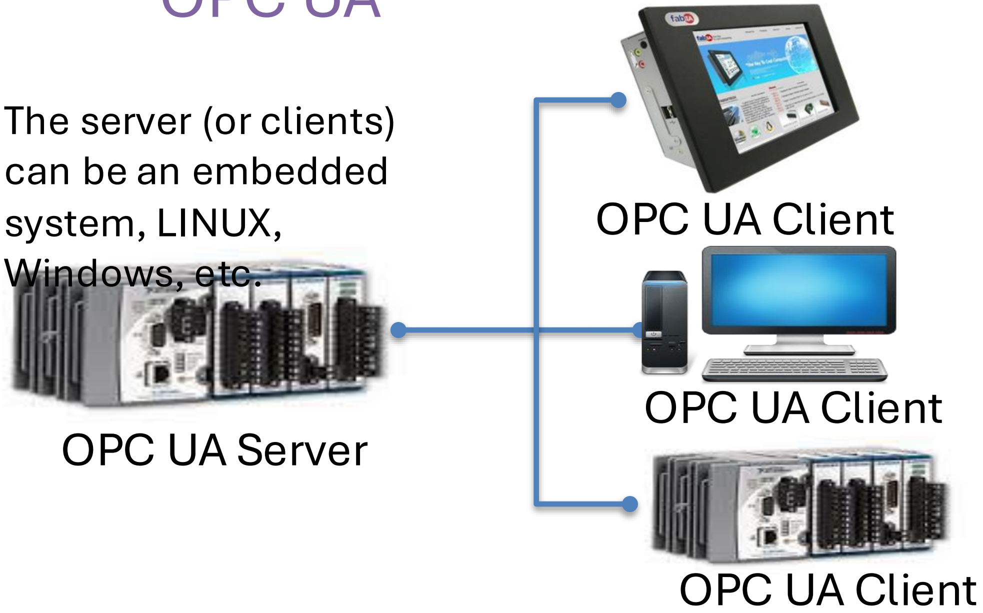


Classic OPC vs. OPC UA

Classic OPC (DCOM)



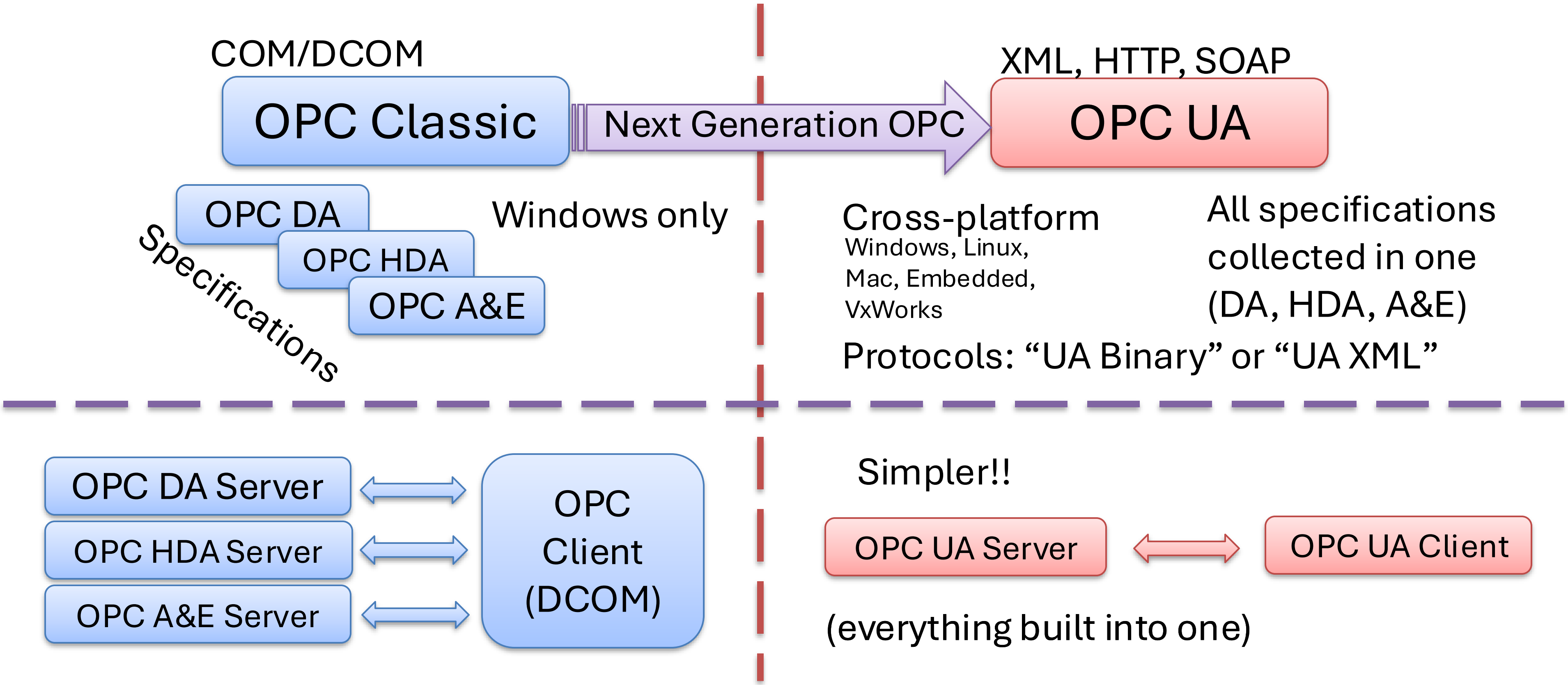
OPC UA


The server (or clients) can be an embedded system, LINUX, Windows, etc.

Classic OPC requires a Microsoft Windows operating system to implement COM/DCOM server functionality. By utilizing SOA and Web Services, OPC UA is a platform-independent system that eliminates the previous dependency on a Windows operating system. By utilizing SOAP/XML over HTTP, OPC UA can deploy on a variety of embedded systems regardless of whether the system is a general-purpose operating system, such as Windows, or a deterministic real-time operating system.

http://www.ni.com/white-paper/13843/en/

Next Generation OPC



COM/DCOM

OPC Classic

Next Generation OPC

XML, HTTP, SOAP

OPC UA

OPC DA

OPC HDA

OPC A&E

Specifications

Windows only

Cross-platform

Windows, Linux,
Mac, Embedded,
VxWorks

All specifications collected in one (DA, HDA, A&E)

Protocols: “UA Binary” or “UA XML”

OPC DA Server

OPC HDA Server

OPC A&E Server

OPC Client (DCOM)

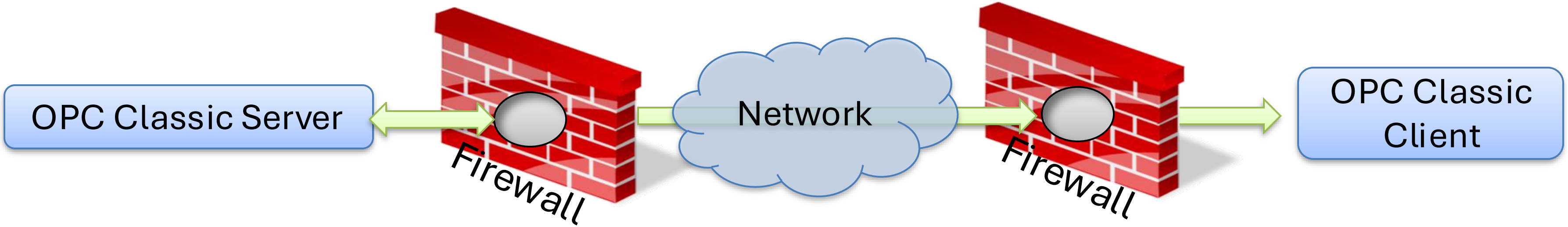
Simpler!!

OPC UA Server

OPC UA Client

(everything built into one)

Firewall



OPC Classic Server

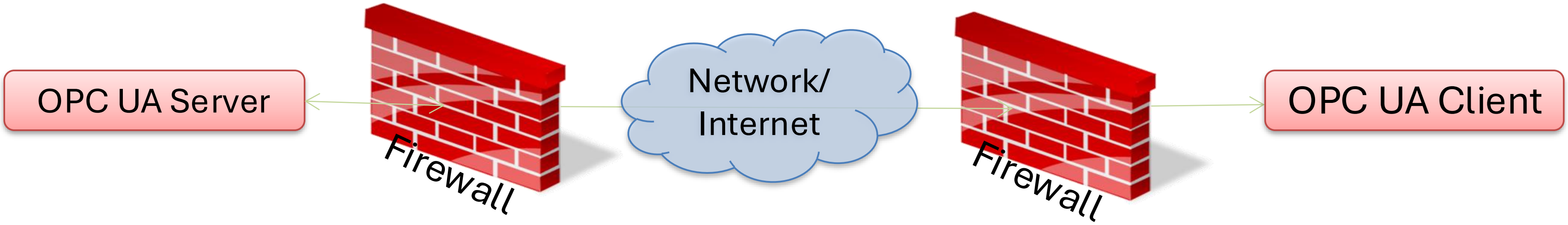
Firewall

Network

Firewall

OPC Classic Client

To open DCOM through firewalls demanded a large hole in the firewall!
Impossible to route over Internet!



OPC UA Server

Firewall

Network/ Internet

Firewall

OPC UA Client

No hole in firewall (UA XML) or just a simple needlestick (UA Binary) is necessary
Easy to route over Internet!

OPC UA (Unified Architecture)

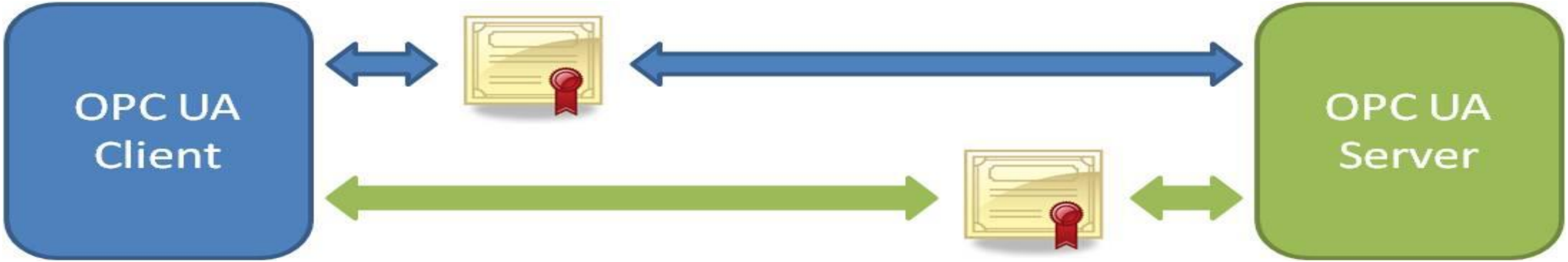
- OPC UA solves problems with standard/classic OPC
  - Works only on Windows
  - Cumbersome to use OPC in a network due to COM/DCOM
- OPC UA eliminating the need to use a Microsoft Windows based platform of earlier OPC versions.
- OPC UA combines the functionality of the existing OPC interfaces with new technologies such as XML and Web Services (HTTP, SOAP)
- Cross-platform
- No dedicated OPC Server is no longer necessary because the server can run on an embedded system

OPC UA Protocols

- OPC UA supports two protocols.
  - **“UA Binary”** protocol opc.tcp://Server
    This uses a simple binary protocol
  - **“UA XML”** protocol http://Server
    This used open standards like XML, SOAP (-> Web Service)
- This is visible to application programmers only via changes to the URL.
- Otherwise OPC UA works completely transparent to the API.

OPC UA Security

One of the most important benefits of eliminating the reliance on COM/DCOM technology is the expanded security features.



[Figure: http://www.ni.com/white-paper/13843/en/]

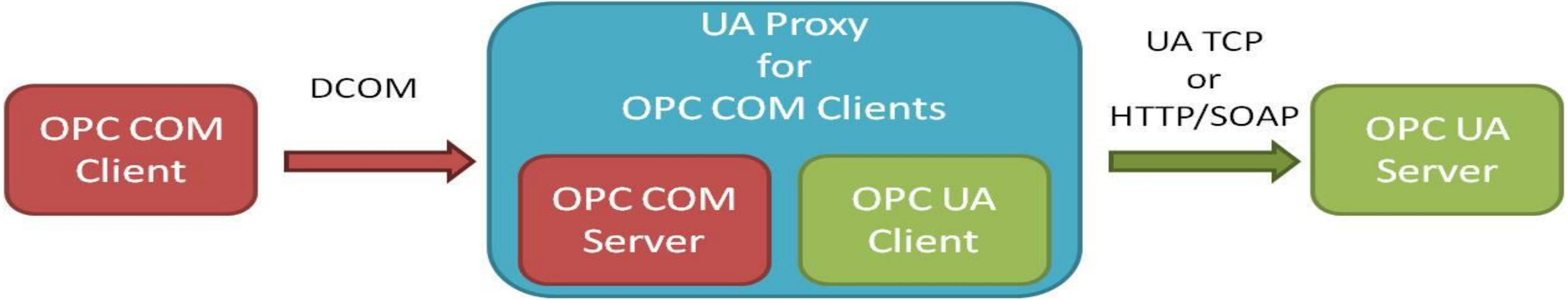
- OPC UA requires handshaking between clients and servers using X.509 Web standard certificates for authentication before they are able to talk with one another.
- To communicate between the server and client, the user can choose from three kinds of messaging modes: None, Sign, Sign and Encrypt.
- OPC UA can communicate through any standard HTTP or UA TCP port. Through this standardization, OPC UA can connect securely over a VPN and through firewalls to allow seamless, remote client-to-server connectivity.

http://www.ni.com/white-paper/13843/en/

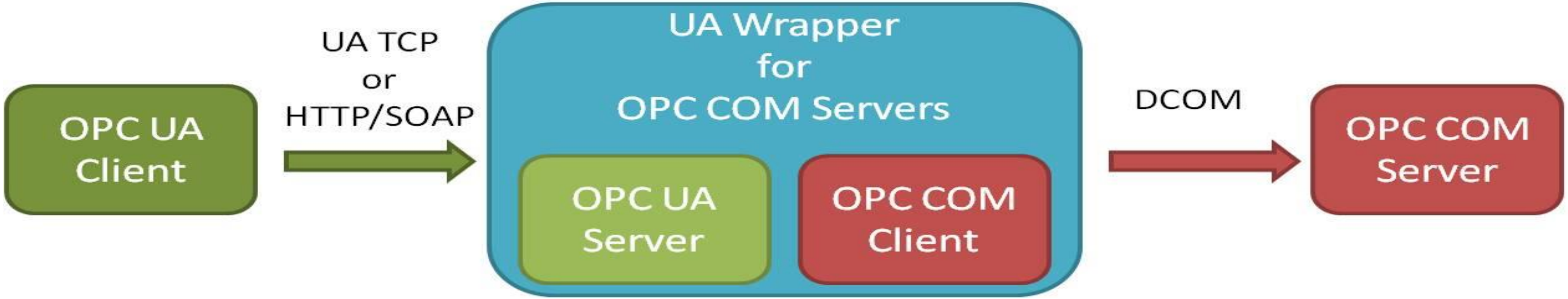
Classic OPC and OPC UA Compatibility

Because of the shift in data communication technology, the OPC UA protocol is **not** inherently backwards compatible with Classic OPC data access (DA) models!

Classic OPC COM-based Clients require a UA Proxy to communicate with UA Servers:



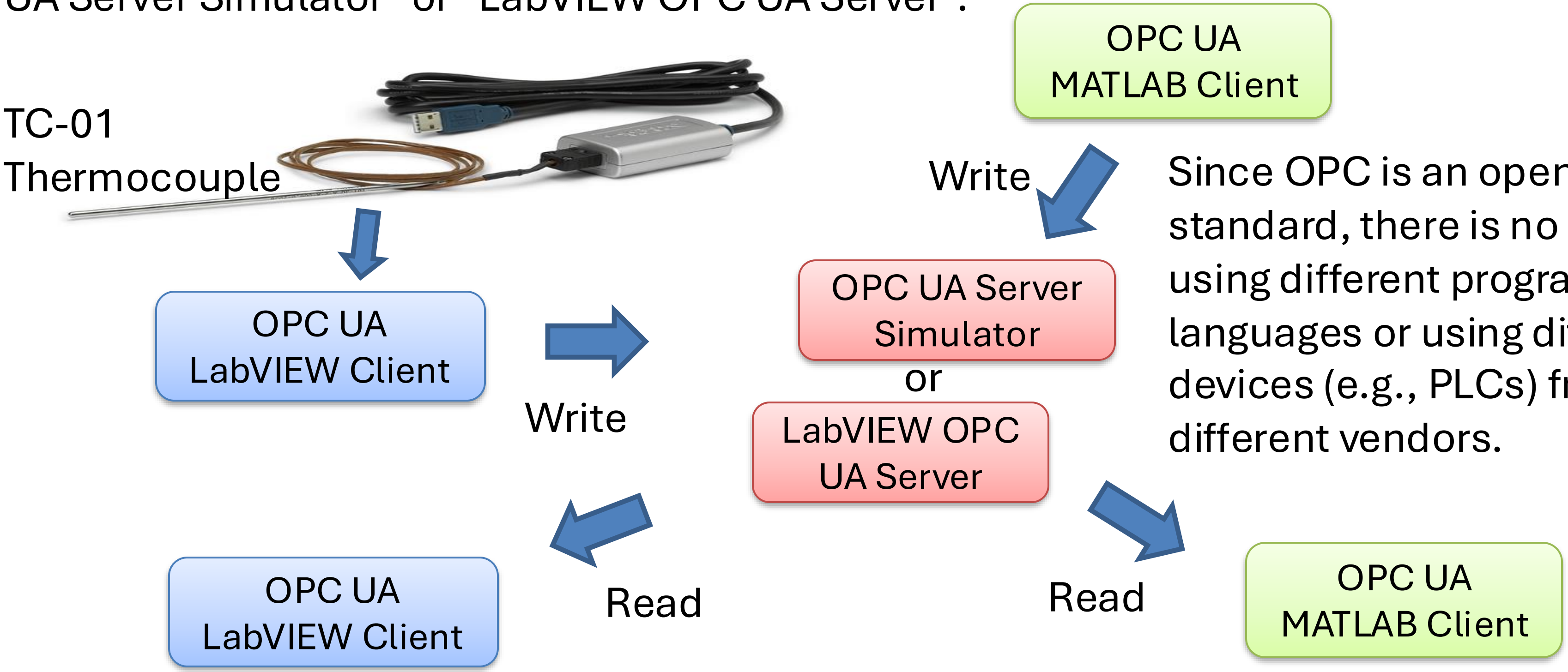
Classic OPC COM-based Servers require UA Wrappers to interact with UA Clients:



http://www.ni.com/white-paper/13843/en/

OPC UA Scenario

This OPC UA Scenario shows multiple OPC UA Clients made with different Programming Languages where some Write Data and others Read Data from an OPC Server, e.g., “OPC UA Server Simulator” or “LabVIEW OPC UA Server”.



TC-01
Thermocouple

OPC UA
LabVIEW Client

Write

OPC UA
MATLAB Client

Write

OPC UA Server
Simulator

or

LabVIEW OPC
UA Server

Read

OPC UA
LabVIEW Client

Read

OPC UA
MATLAB Client

Since OPC is an open standard, there is no problem using different programming languages or using different devices (e.g., PLCs) from different vendors.

OPC UA Server Simulator

Free OPC UA Simulation Server from Integration Objects

Hans-Petter Halvorsen

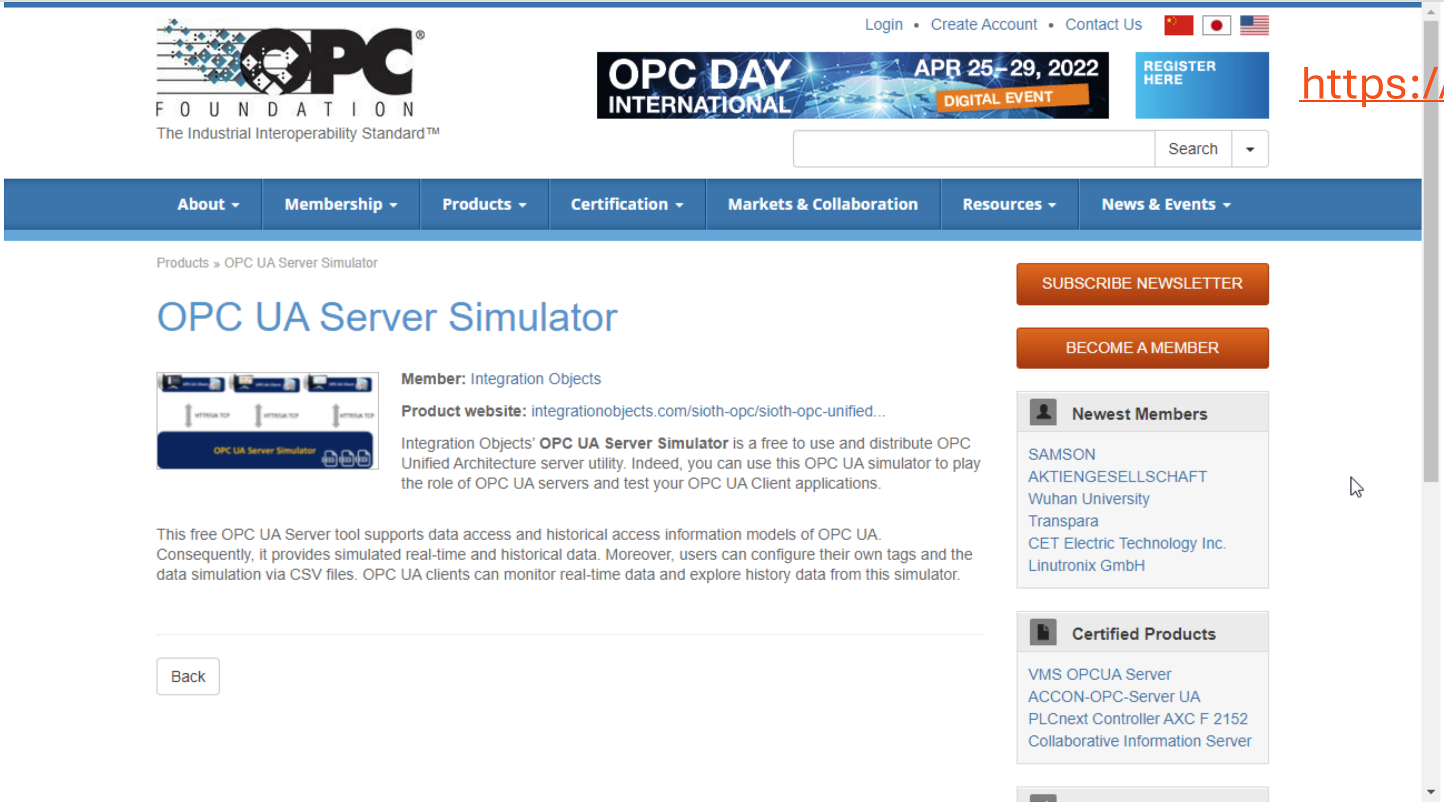
Table of Contents

OPC UA Server Simulator

- This free OPC UA Server tool supports data access and historical access information models of OPC UA.
- Consequently, it provides simulated real-time and historical data.
- Moreover, users can configure their own tags and the data simulation via CSV files.
- OPC UA clients can monitor real-time data and explore history data from this simulator.
- https://opcfoundation.org/products/view/opc-ua-server-simulator

OPC UA Server Simulator

https://opcfoundation.org



OPC UA Server Simulator

Member: Integration Objects

Product website: integrationobjects.com/sioth-opc/sioth-opc-unified...

Integration Objects' **OPC UA Server Simulator** is a free to use and distribute OPC Unified Architecture server utility. Indeed, you can use this OPC UA simulator to play the role of OPC UA servers and test your OPC UA Client applications.

This free OPC UA Server tool supports data access and historical access information models of OPC UA. Consequently, it provides simulated real-time and historical data. Moreover, users can configure their own tags and the data simulation via CSV files. OPC UA clients can monitor real-time data and explore history data from this simulator.

Newest Members

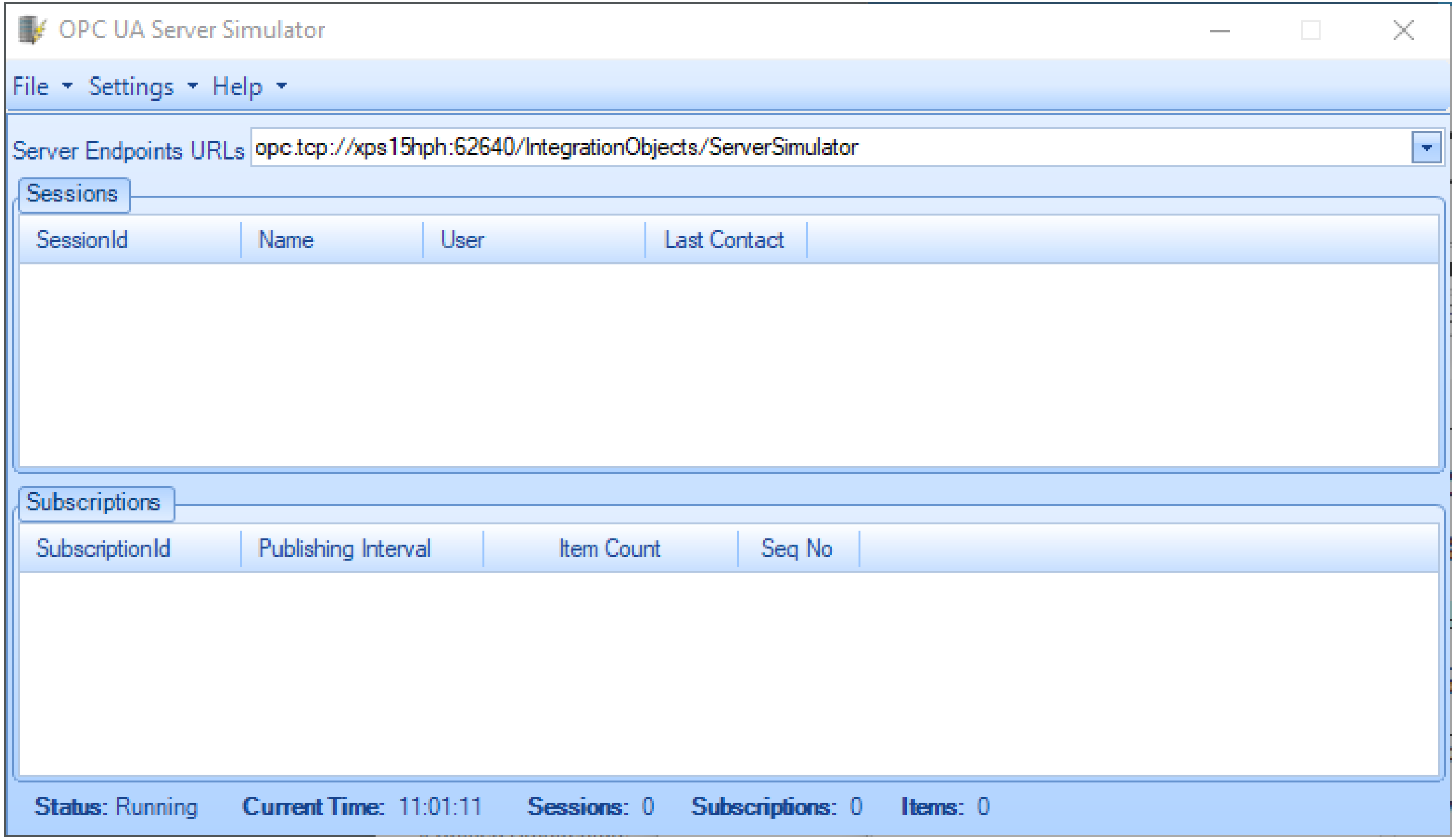
SAMSON AKTIENGESELLSCHAFT
Wuhan University
Transpara
CET Electric Technology Inc.
Linutronix GmbH

Certified Products

VMS OPCUA Server
ACCON-OPC-Server UA
PLCnext Controller AXC F 2152
Collaborative Information Server

https://opcfoundation.org/products/view/opc-ua-server-simulator

OPC UA Server Simulator



Server Endpoints URLs opc.tcp://xps15hph:62640/IntegrationObjects/ServerSimulator

Sessions

SessionId	Name	User	Last Contact

Subscriptions

SubscriptionId	Publishing Interval	Item Count	Seq No

Status: Running **Current Time:** 11:01:11 **Sessions:** 0 **Subscriptions:** 0 **Items:** 0

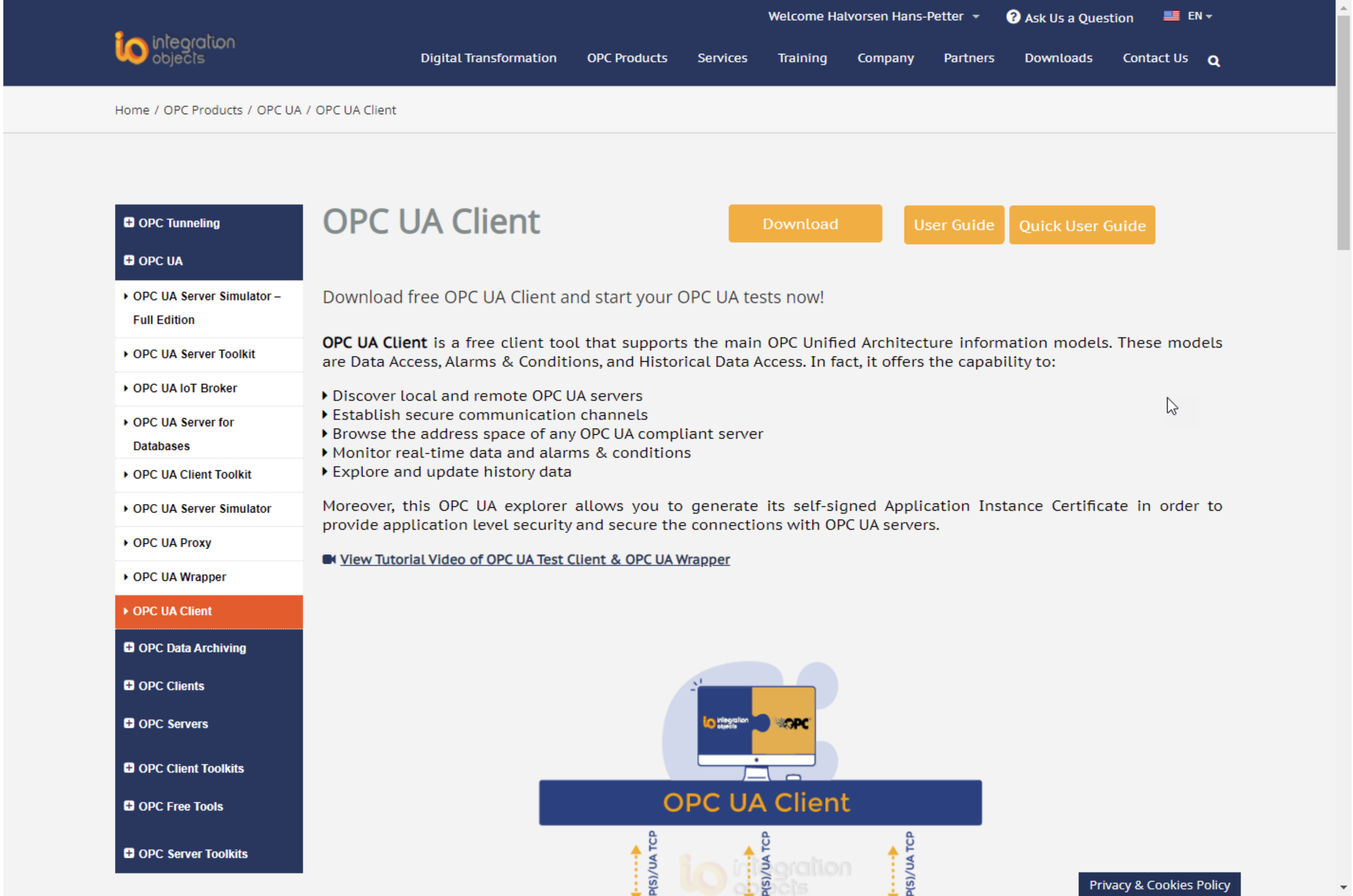
OPC UA Server Simulator

The OPC UA Server Simulator uses 2 CSV simulation files:

- “**AddressSpace.csv**” used to build the address space of the OPC UA Server.
- “**ValueSpace.csv**” used to simulate the data values of the OPC UA items.
- Those two files are located at the following path: X:\Program Files (x86)\Integration Objects\Integration Objects' OPC UA Server Simulator\OPC UA Server Simulator\DATA

“OPC UA Client” Tool

- “OPC UA Client” is a free client tool that supports the main OPC Unified Architecture information models.
- These models are Data Access, Alarms & Conditions, and Historical Data Access
- https://integrationobjects.com/sioth-opc/sioth-opc-unified-architecture/opc-ua-client/



Welcome Halvorsen Hans-Petter ▾ Ask Us a Question EN ▾

Digital Transformation | OPC Products | Services | Training | Company | Partners | Downloads | Contact Us

Home / OPC Products / OPC UA / OPC UA Client

- OPC Tunneling
- OPC UA
  - OPC UA Server Simulator – Full Edition
  - OPC UA Server Toolkit
  - OPC UA IoT Broker
  - OPC UA Server for Databases
  - OPC UA Client Toolkit
  - OPC UA Server Simulator
  - OPC UA Proxy
  - OPC UA Wrapper
  - OPC UA Client
- OPC Data Archiving
- OPC Clients
- OPC Servers
- OPC Client Toolkits
- OPC Free Tools
- OPC Server Toolkits

OPC UA Client

Download | User Guide | Quick User Guide

Download free OPC UA Client and start your OPC UA tests now!

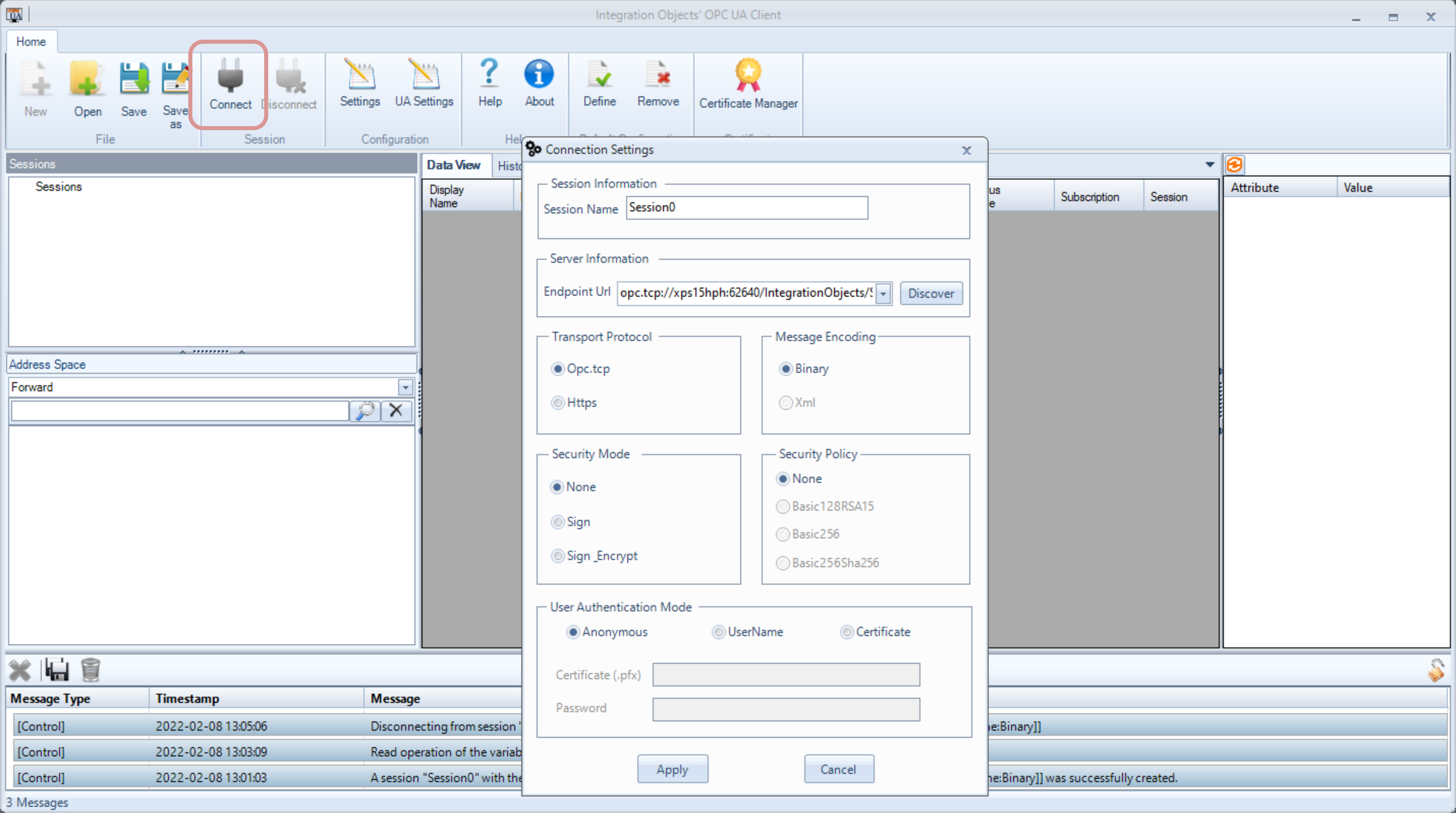
OPC UA Client is a free client tool that supports the main OPC Unified Architecture information models. These models are Data Access, Alarms & Conditions, and Historical Data Access. In fact, it offers the capability to:

- Discover local and remote OPC UA servers
- Establish secure communication channels
- Browse the address space of any OPC UA compliant server
- Monitor real-time data and alarms & conditions
- Explore and update history data

Moreover, this OPC UA explorer allows you to generate its self-signed Application Instance Certificate in order to provide application level security and secure the connections with OPC UA servers.

View Tutorial Video of OPC UA Test Client & OPC UA Wrapper

Privacy & Cookies Policy



Connection Settings

Session Information

Session Name: Session0

Server Information

Endpoint Url: opc.tcp://xps15hph:62640/IntegrationObjects/!

Transport Protocol
● Opc.tcp
○ Https

Message Encoding
● Binary
○ Xml

Security Mode
● None
○ Sign
○ Sign_Encrypt

Security Policy
● None
○ Basic128RSA15
○ Basic256
○ Basic256Sha256

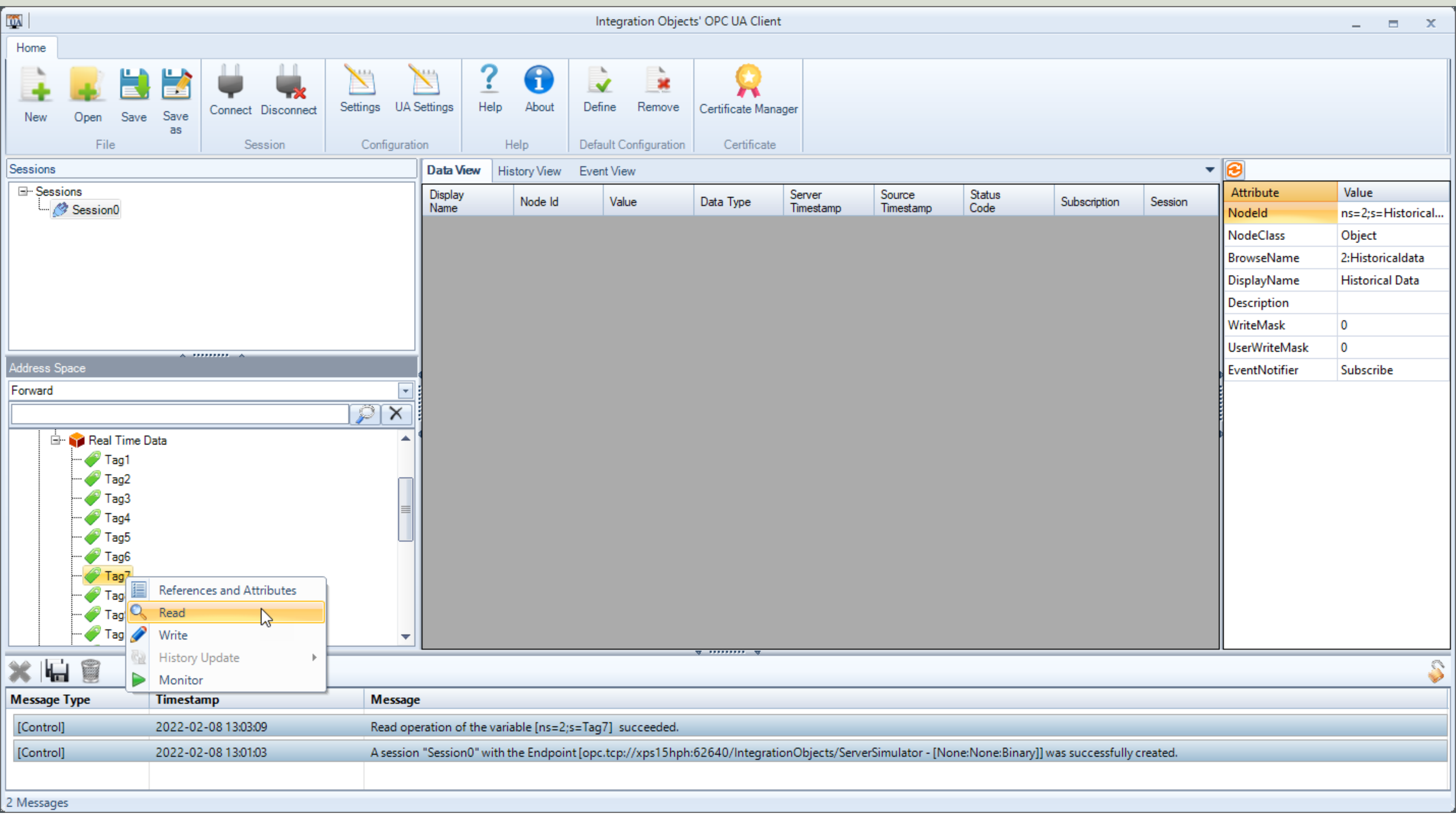
User Authentication Mode
● Anonymous ○ UserName ○ Certificate

Certificate (.pfx)

Password

Message Type	Timestamp	Message
[Control]	2022-02-08 13:05:06	Disconnecting from session ' … e:Binary]]
[Control]	2022-02-08 13:03:09	Read operation of the variab
[Control]	2022-02-08 13:01:03	A session "Session0" with the … e:Binary]] was successfully created.

3 Messages



Attribute	Value
NodeId	ns=2;s=Historical...
NodeClass	Object
BrowseName	2:Historicaldata
DisplayName	Historical Data
Description	
WriteMask	0
UserWriteMask	0
EventNotifier	Subscribe

Message Type	Timestamp	Message
[Control]	2022-02-08 13:03:09	Read operation of the variable [ns=2;s=Tag7] succeeded.
[Control]	2022-02-08 13:01:03	A session "Session0" with the Endpoint [opc.tcp://xps15hph:62640/IntegrationObjects/ServerSimulator - [None:None:Binary]] was successfully created.

2 Messages

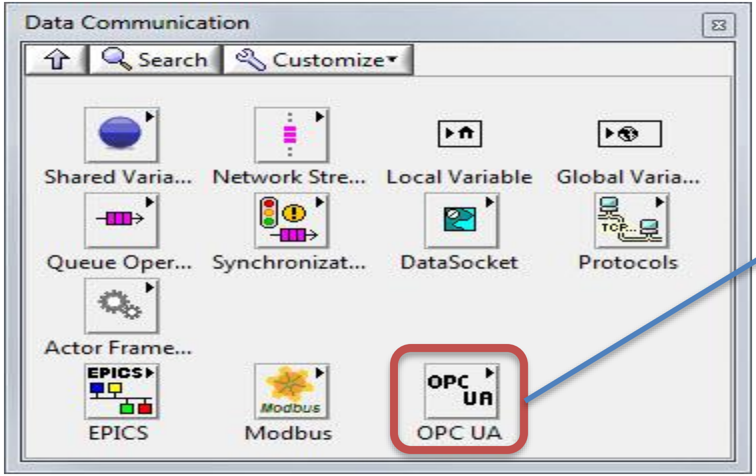
OPC UA in LabVIEW

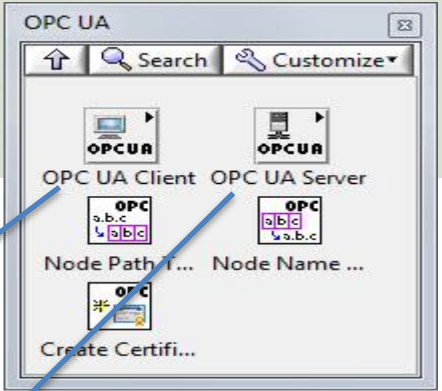
Hans-Petter Halvorsen

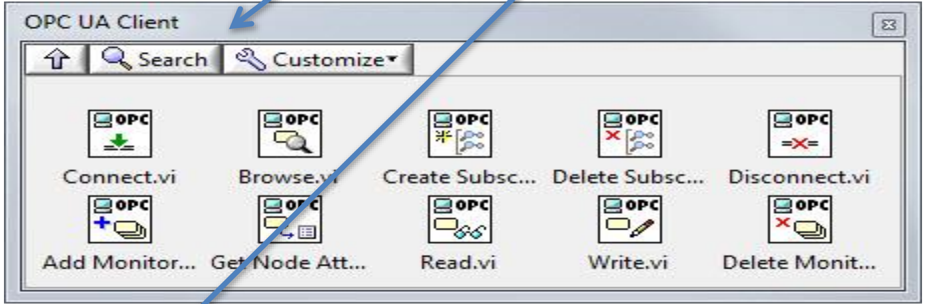
Table of Contents

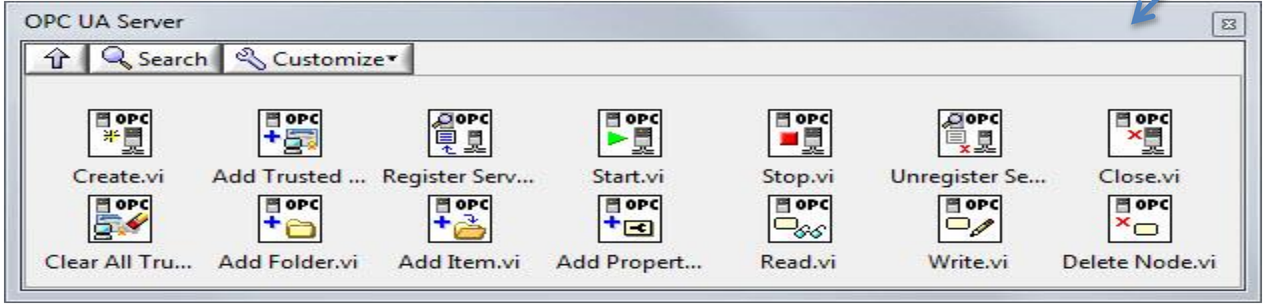
OPC UA in LabVIEW

http://zone.ni.com/reference/en-XX/help/371618J-1/TOC9.htm







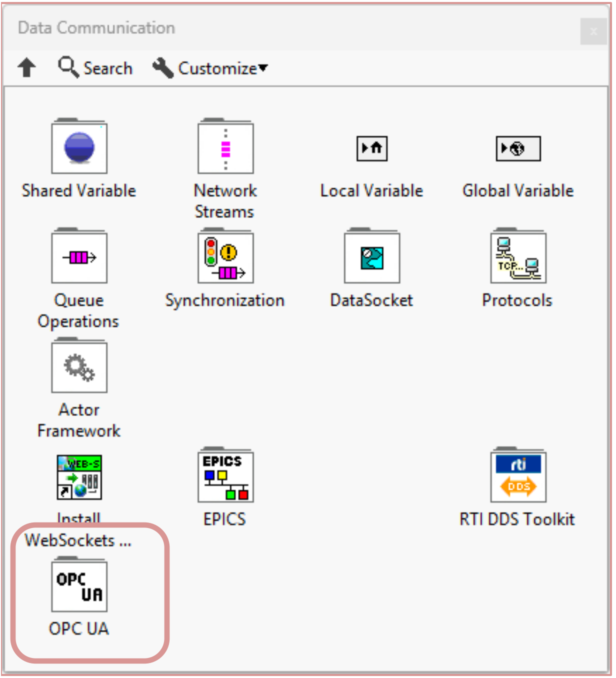


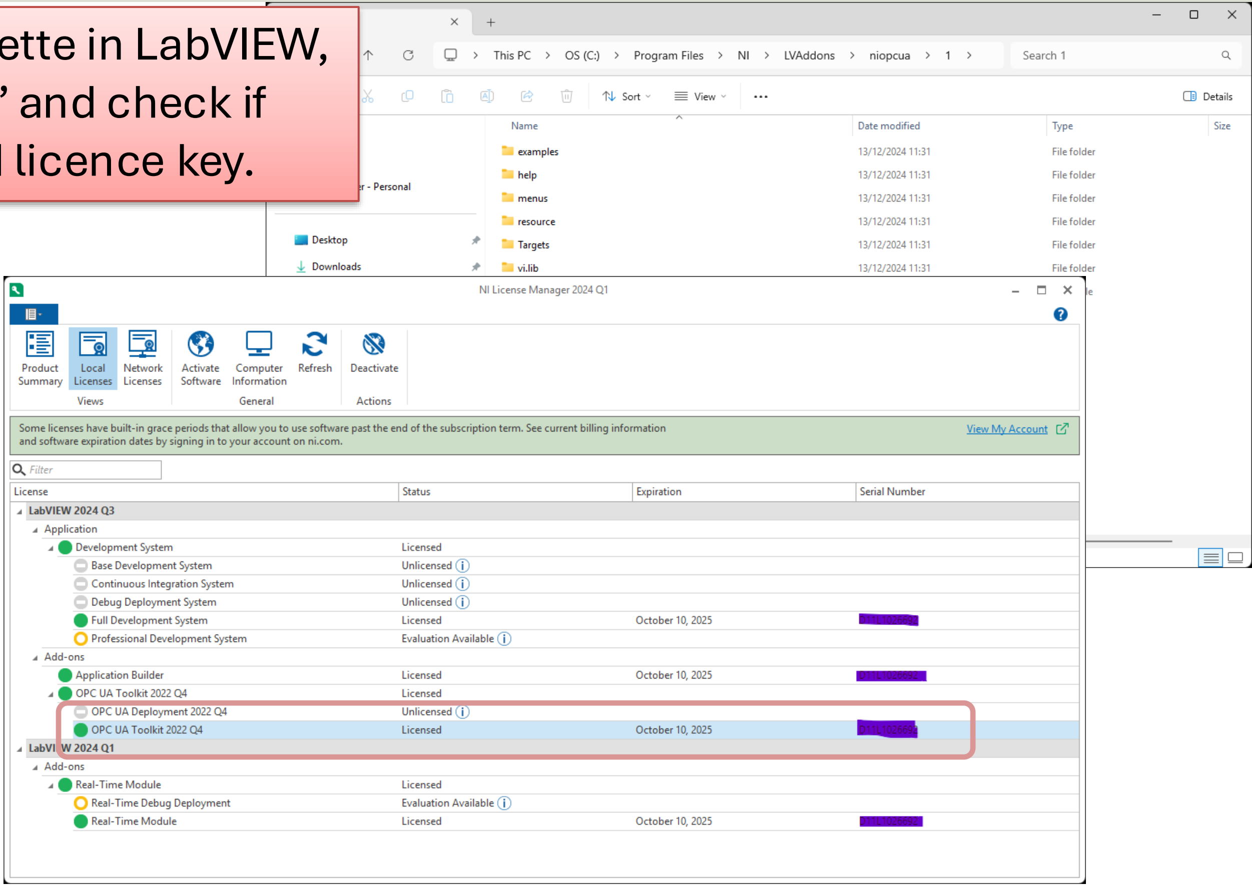
Note! You need to install the **LabVIEW OPC UA Toolkit**

https://zone.ni.com/reference/en-XX/help/376230B-01/

LabVIEW OPC UA Toolkit

If you don’t find the OPC UA palette in LabVIEW, open the “**NI License Manager**” and check if the “OPC UA Toolkit” has a valid licence key.



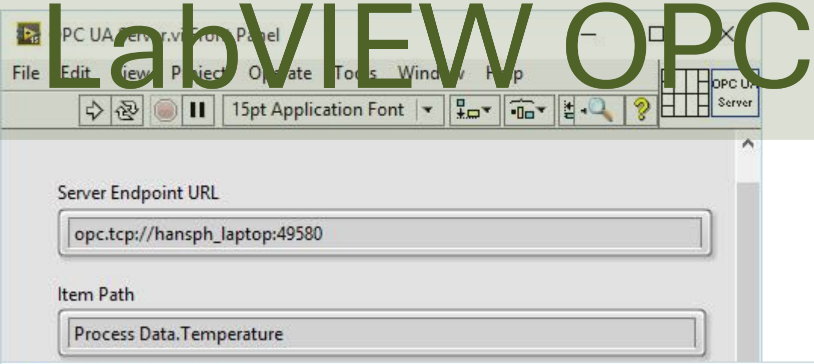


Name	Date modified	Type	Size
examples	13/12/2024 11:31	File folder	
help	13/12/2024 11:31	File folder	
menus	13/12/2024 11:31	File folder	
resource	13/12/2024 11:31	File folder	
Targets	13/12/2024 11:31	File folder	
vi.lib	13/12/2024 11:31	File folder	

Some licenses have built-in grace periods that allow you to use software past the end of the subscription term. See current billing information and software expiration dates by signing in to your account on ni.com.

License	Status	Expiration	Serial Number
LabVIEW 2024 Q3			
Application			
● Development System	Licensed		
○ Base Development System	Unlicensed ⓘ		
○ Continuous Integration System	Unlicensed ⓘ		
○ Debug Deployment System	Unlicensed ⓘ		
● Full Development System	Licensed	October 10, 2025	[redacted]
○ Professional Development System	Evaluation Available ⓘ		
Add-ons			
● Application Builder	Licensed	October 10, 2025	[redacted]
● OPC UA Toolkit 2022 Q4	Licensed		
○ OPC UA Deployment 2022 Q4	Unlicensed ⓘ		
● OPC UA Toolkit 2022 Q4	Licensed	October 10, 2025	[redacted]
LabVIEW 2024 Q1			
Add-ons			
● Real-Time Module	Licensed		
○ Real-Time Debug Deployment	Evaluation Available ⓘ		
● Real-Time Module	Licensed	October 10, 2025	[redacted]

LabVIEW OPC UA Server Example

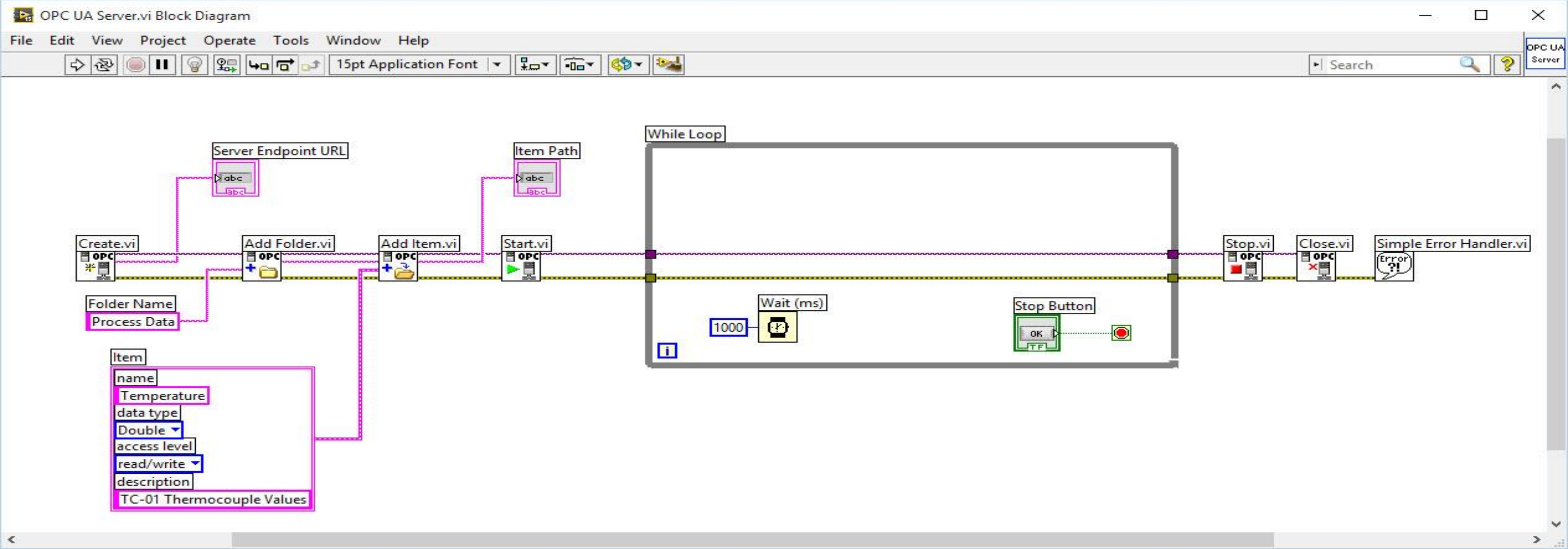

OPC UA Server.vi Front Panel
File Edit View Project Operate Tools Window Help
15pt Application Font
OPC UA Server

Server Endpoint URL

opc.tcp://hansph_laptop:49580

Item Path

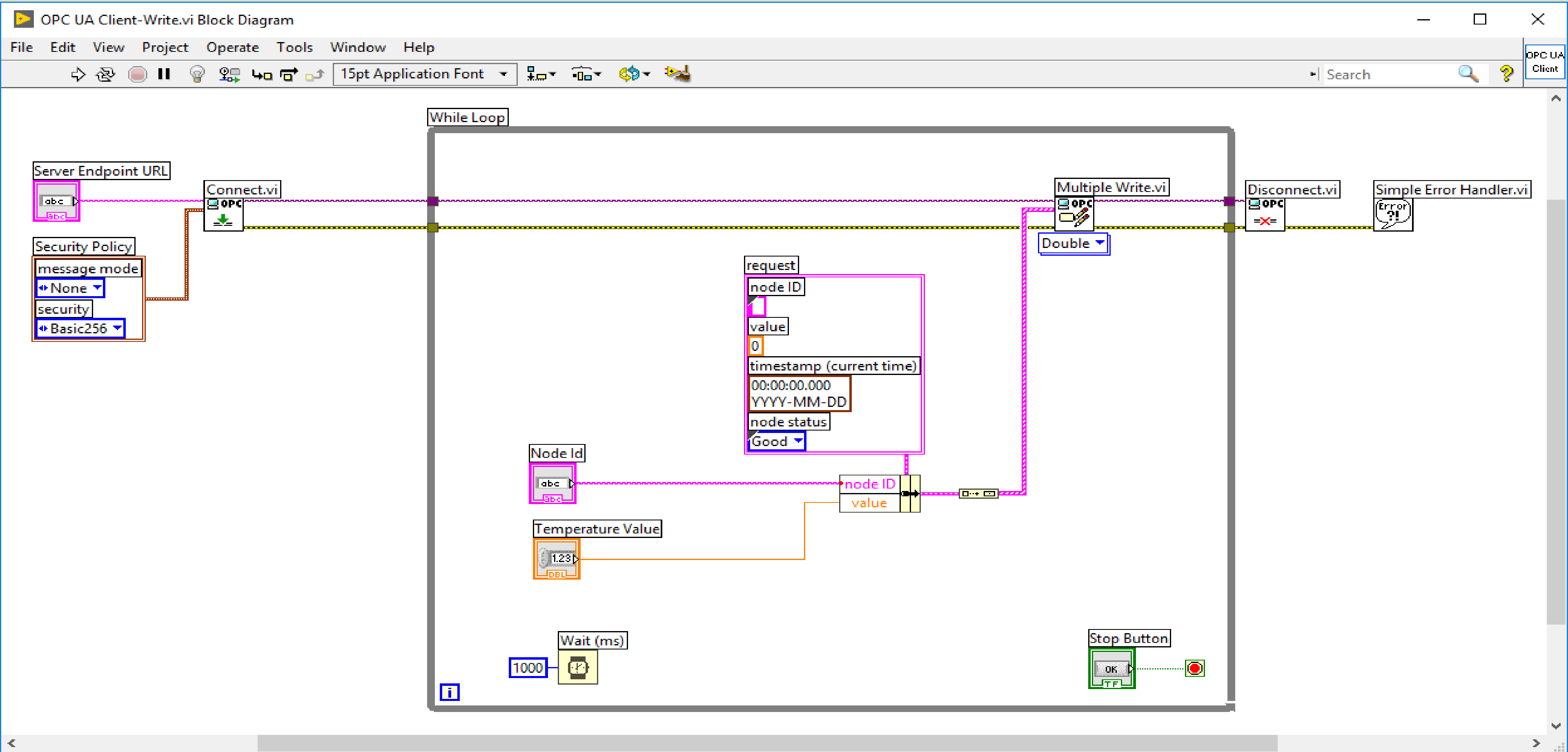
Process Data.Temperature



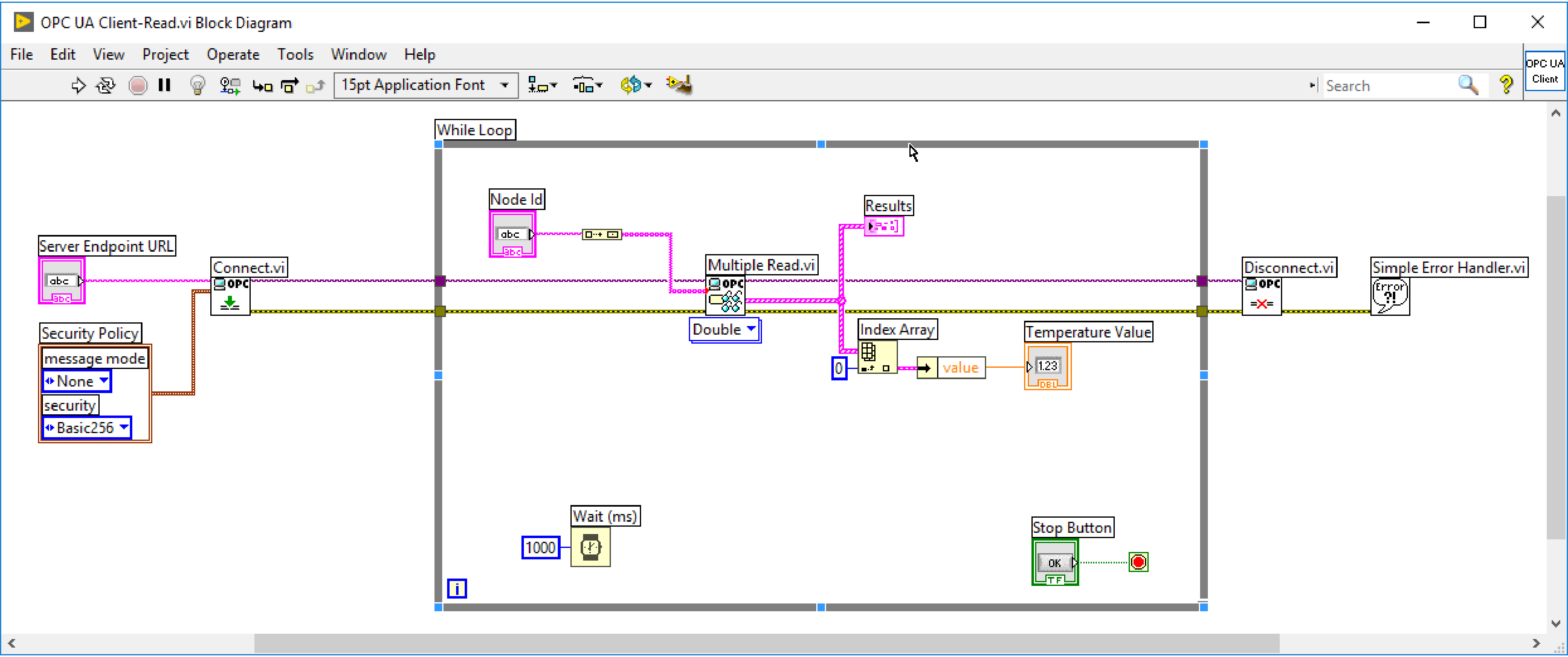
Folder Name: Process Data

Item
- name: Temperature
- data type: Double
- access level: read/write
- description: TC-01 Thermocouple Values

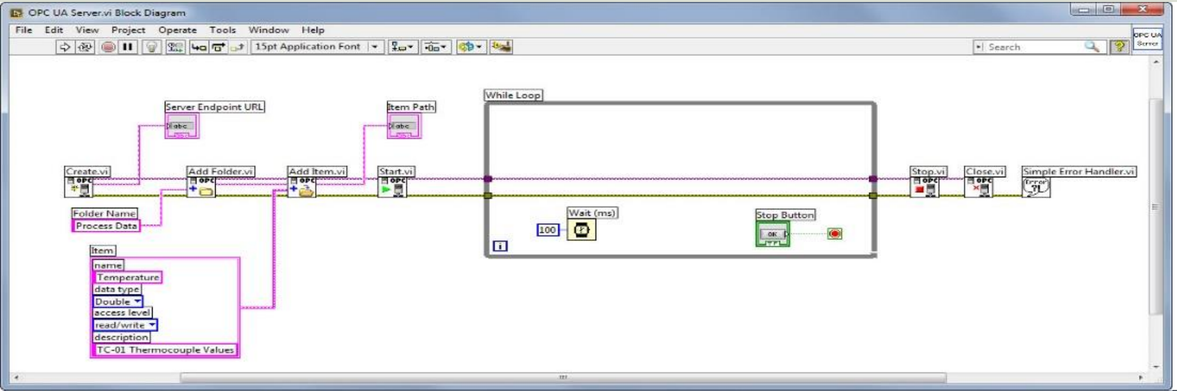
LabVIEW OPC UA Client - Write



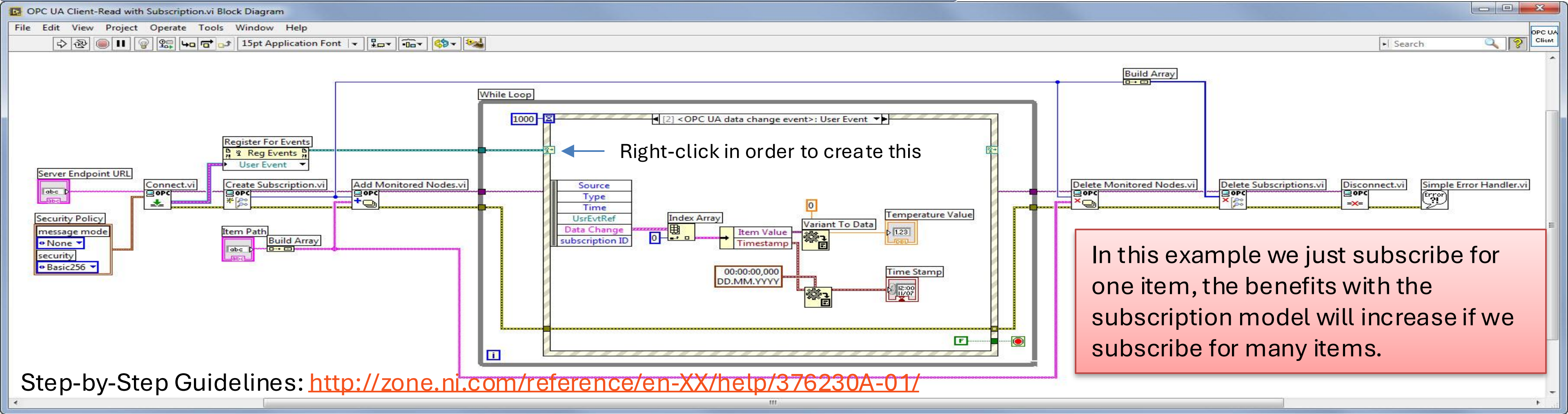
LabVIEW OPC UA Client - Read



OPC UA Client with Subscription



This is a more complex example where you read data on the client only when the value on the server is changed.



Right-click in order to create this

In this example we just subscribe for one item, the benefits with the subscription model will increase if we subscribe for many items.

Step-by-Step Guidelines: http://zone.ni.com/reference/en-XX/help/376230A-01/

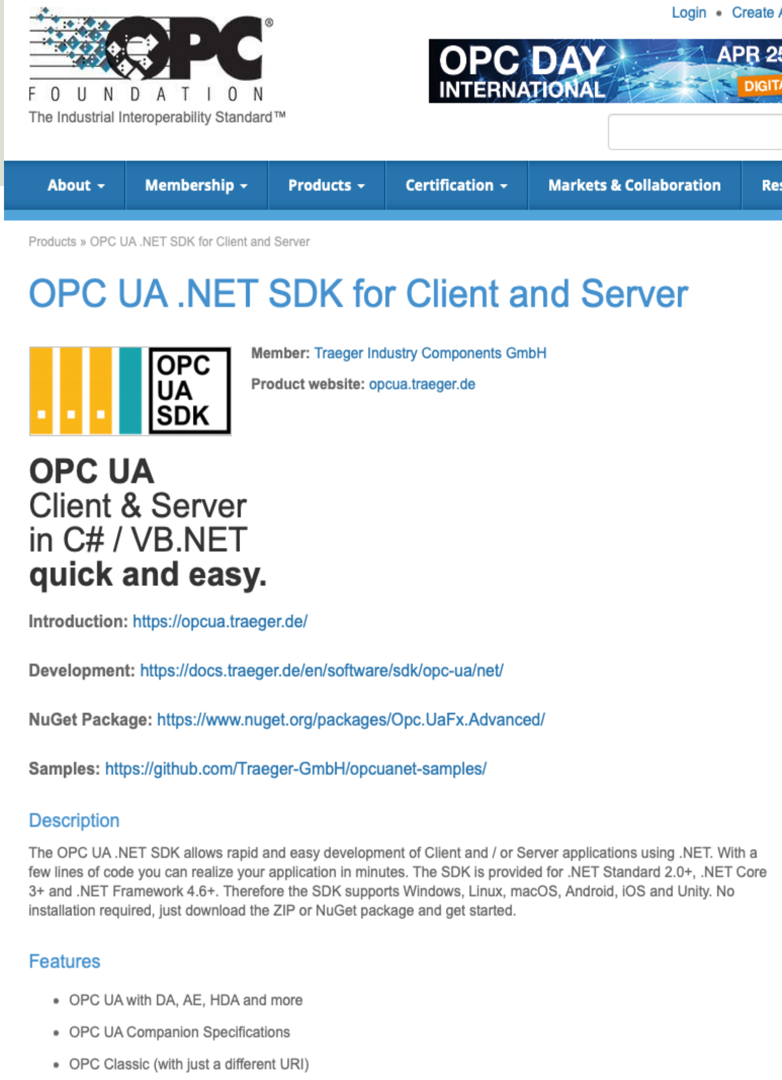
OPC UA in Visual Studio/C#

Hans-Petter Halvorsen

Table of Contents

OPC UA .NET SDK

- The “OPC UA .NET SDK” comes with an evaluation license which can be used unlimited for each application run for 30 minutes
- It comes in a **NuGet** Package you can install and use in your Visual Studio Project
- https://opcfoundation.org/products/view/opc-ua-net-sdk-for-client-and-server



Products » OPC UA .NET SDK for Client and Server

OPC UA .NET SDK for Client and Server

Member: Traeger Industry Components GmbH

Product website: opcua.traeger.de

OPC UA
Client & Server
in C# / VB.NET
quick and easy.

Introduction: https://opcua.traeger.de/

Development: https://docs.traeger.de/en/software/sdk/opc-ua/net/

NuGet Package: https://www.nuget.org/packages/Opc.UaFx.Advanced/

Samples: https://github.com/Traeger-GmbH/opcuanet-samples/

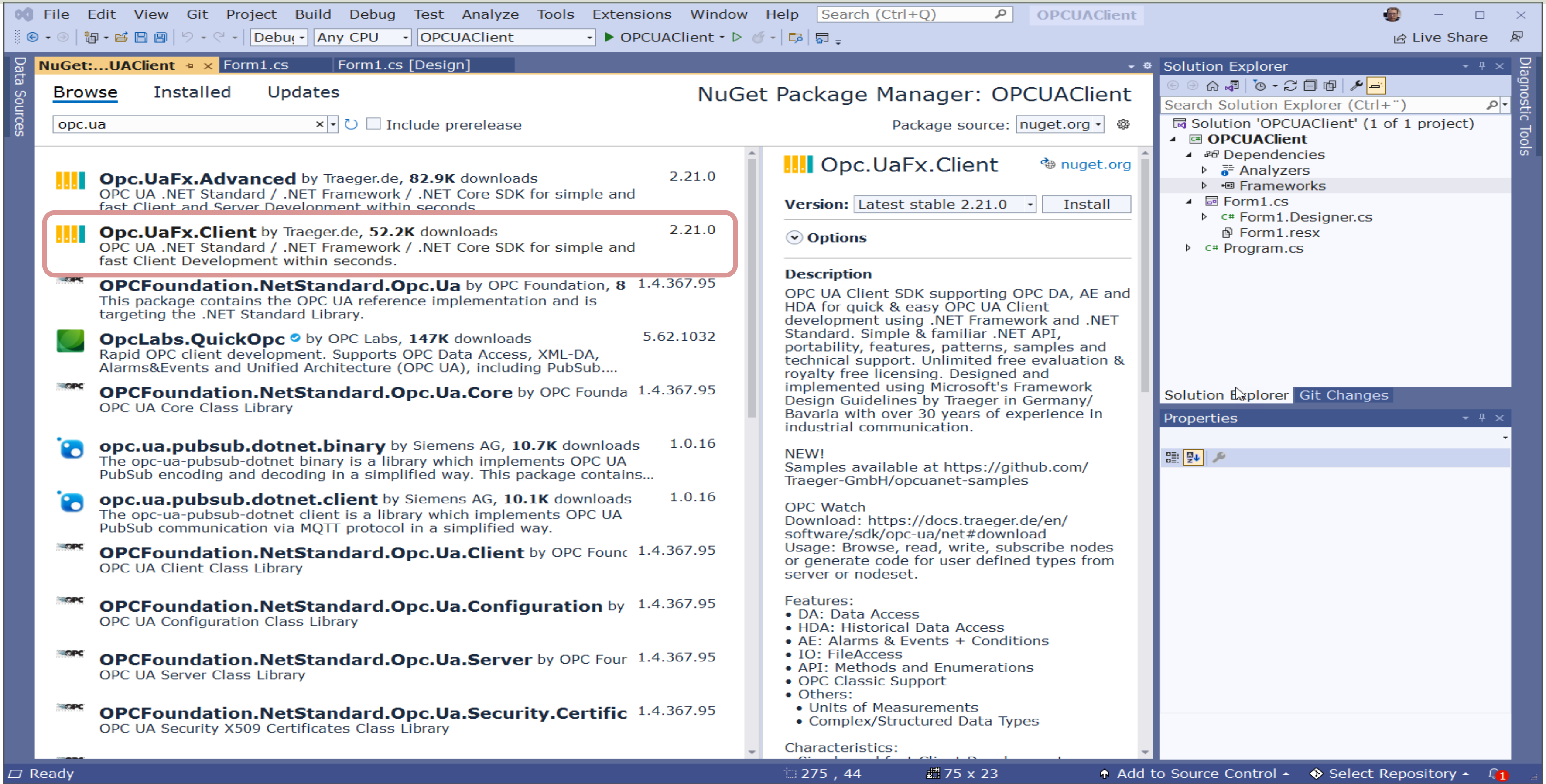
Description

The OPC UA .NET SDK allows rapid and easy development of Client and / or Server applications using .NET. With a few lines of code you can realize your application in minutes. The SDK is provided for .NET Standard 2.0+, .NET Core 3+ and .NET Framework 4.6+. Therefore the SDK supports Windows, Linux, macOS, Android, iOS and Unity. No installation required, just download the ZIP or NuGet package and get started.

Features

- OPC UA with DA, AE, HDA and more
- OPC UA Companion Specifications
- OPC Classic (with just a different URI)

NuGet Package



NuGet Package Manager: OPCUAClient

opc.ua ☐ Include prerelease

Package source: nuget.org

Opc.UaFx.Advanced by Traeger.de, **82.9K** downloads 2.21.0
OPC UA .NET Standard / .NET Framework / .NET Core SDK for simple and fast Client and Server Development within seconds

Opc.UaFx.Client by Traeger.de, **52.2K** downloads 2.21.0
OPC UA .NET Standard / .NET Framework / .NET Core SDK for simple and fast Client Development within seconds.

OPCFoundation.NetStandard.Opc.Ua by OPC Foundation, 8 1.4.367.95
This package contains the OPC UA reference implementation and is targeting the .NET Standard Library.

OpcLabs.QuickOpc by OPC Labs, **147K** downloads 5.62.1032
Rapid OPC client development. Supports OPC Data Access, XML-DA, Alarms&Events and Unified Architecture (OPC UA), including PubSub....

OPCFoundation.NetStandard.Opc.Ua.Core by OPC Founda 1.4.367.95
OPC UA Core Class Library

opc.ua.pubsub.dotnet.binary by Siemens AG, **10.7K** downloads 1.0.16
The opc-ua-pubsub-dotnet binary is a library which implements OPC UA PubSub encoding and decoding in a simplified way. This package contains...

opc.ua.pubsub.dotnet.client by Siemens AG, **10.1K** downloads 1.0.16
The opc-ua-pubsub-dotnet client is a library which implements OPC UA PubSub communication via MQTT protocol in a simplified way.

OPCFoundation.NetStandard.Opc.Ua.Client by OPC Founc 1.4.367.95
OPC UA Client Class Library

OPCFoundation.NetStandard.Opc.Ua.Configuration by 1.4.367.95
OPC UA Configuration Class Library

OPCFoundation.NetStandard.Opc.Ua.Server by OPC Four 1.4.367.95
OPC UA Server Class Library

OPCFoundation.NetStandard.Opc.Ua.Security.Certific 1.4.367.95
OPC UA Security X509 Certificates Class Library

Opc.UaFx.Client nuget.org

Version: Latest stable 2.21.0 Install

Options

Description

OPC UA Client SDK supporting OPC DA, AE and HDA for quick & easy OPC UA Client development using .NET Framework and .NET Standard. Simple & familiar .NET API, portability, features, patterns, samples and technical support. Unlimited free evaluation & royalty free licensing. Designed and implemented using Microsoft's Framework Design Guidelines by Traeger in Germany/ Bavaria with over 30 years of experience in industrial communication.

NEW!
Samples available at https://github.com/Traeger-GmbH/opcuanet-samples

OPC Watch
Download: https://docs.traeger.de/en/software/sdk/opc-ua/net#download
Usage: Browse, read, write, subscribe nodes or generate code for user defined types from server or nodeset.

Features:
- DA: Data Access
- HDA: Historical Data Access
- AE: Alarms & Events + Conditions
- IO: FileAccess
- API: Methods and Enumerations
- OPC Classic Support
- Others:
  - Units of Measurements
  - Complex/Structured Data Types

Characteristics:

Solution Explorer

Search Solution Explorer (Ctrl+")

Solution 'OPCUAClient' (1 of 1 project)
- **OPCUAClient**
  - Dependencies
    - Analyzers
    - Frameworks
  - Form1.cs
    - Form1.Designer.cs
    - Form1.resx
  - Program.cs

Solution Explorer Git Changes

Properties

OPC UA Write Example

```
private void btnOpcWrite_Click(object sender, EventArgs e)
{
        string opcUrl = "opc.tcp://localhost:62640/";
        var tagName = "ns=2;s=Tag7";

        var client = new OpcClient(opcUrl);
        client.Connect();

        double temperature;
        temperature = Convert.ToDouble(txtOpcDataWrite.Text);

        client.WriteNode(tagName, temperature);

        client.Disconnect();
}
```

OPC UA Read Example

```
private void btnOpcRead_Click(object sender, EventArgs e)
{
    string opcUrl = "opc.tcp://localhost:62640/";
    var tagName = "ns=2;s=Tag7";

    var client = new OpcClient(opcUrl);
    client.Connect();

    var temperature = client.ReadNode(tagName);
    txtOpcDataRead.Text = temperature.ToString();

    client.Disconnect();
}
```

OPC UA in MATLAB

Hans-Petter Halvorsen

Table of Contents

MATLAB OPC UA - Write

1. Locate Your OPC UA Server
   ```
   serverList = opcuaserverinfo('localhost’)
   ```
2. Create an OPC UA Client
   ```
   uaClient = opcua('localhost’, port)
   ```
3. Connect to the Server
   ```
   connect(uaClient)
   ```
4. Browse OPC UA Server Namespace
   ```
   serverNodes = browseNamespace(uaClient)
   ```
5. Write Current Values to the OPC UA Server
   ```
   newValue = 22.5
   writeValue(uaClient, serverNodes, newValue);
   ```
6. Disconnect
   ```
   disconnect(uaClient)
   ```

MATLAB OPC UA - Read

1. Locate Your OPC UA Server
   ```
   serverList = opcuaserverinfo('localhost’)
   ```
2. Create an OPC UA Client
   ```
   uaClient = opcua('localhost’, port)
   ```
3. Connect to the Server
   ```
   connect(uaClient)
   ```
4. Browse OPC UA Server Namespace
   ```
   serverNodes = browseNamespace(uaClient)
   ```
5. Read Current Values from the OPC UA Server
   ```
   [val,ts,qual] =
   readValue(uaClient,serverNodes)
   ```
6. Disconnect
   ```
   disconnect(uaClient)
   ```

https://www.halvorsen.blog

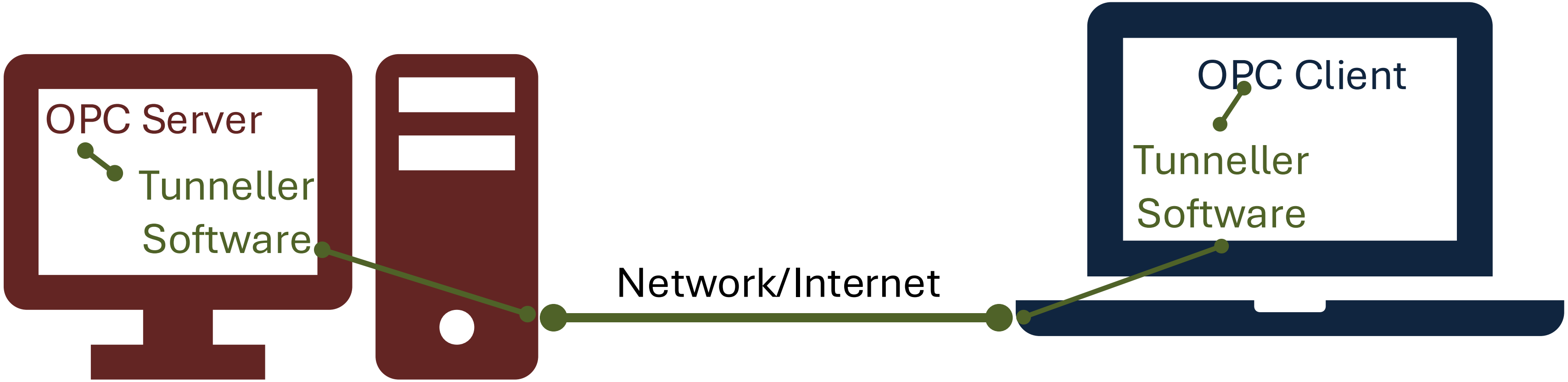
OPC in Network and Tunneling

Hans-Petter Halvorsen

Table of Contents

OPC Tunneller

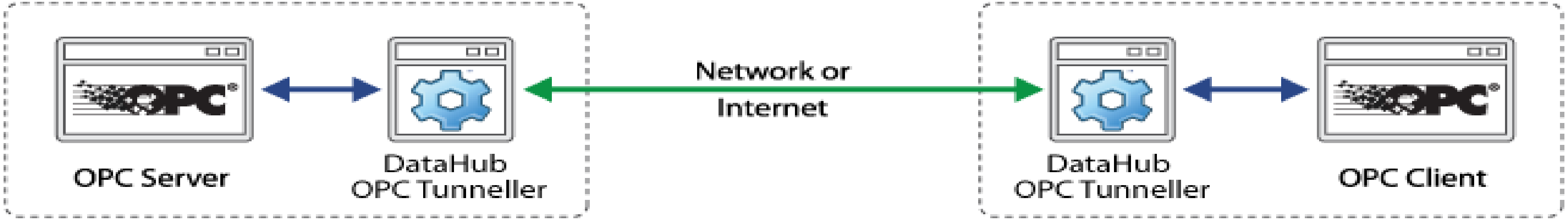
Problem: Sending OPC Data between 2 (or more) Computers in a Network, or even worse, over Internet. OPC DA uses COM/DCOM. This makes it complicated to make it work in a modern Network



Solution: Use OPC Tunneller Software that makes an open tunnel between the 2 network nodes. The goal of OPC tunneling is to eliminate DCOM, i.e., replacing the DCOM networking protocol with TCP.

OPC DA in Network

- OPC DA uses COM/DCOM -> Complicated to make it work in a modern Network!!
- Solution: Use an **OPC Tunneller Software**, e.g.:
  - OPC Tunneller from MatrikonOPC (30 days free trial)
  - Cogent DataHub Tunnelling Software (Trial software works only 1 hour, then needs to be restarted)



Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no

Web: https://www.halvorsen.blog