

<https://www.halvorsen.blog>

SCADA Systems Overview

Supervisory Control and Data Acquisition

Hans-Petter Halvorsen



Table of Contents

- [Introduction](#)
- [SCADA Systems](#)
- [Air Heater](#)
- [Control System](#)
 - [Discrete PID Controller](#)
 - [Data Acquisition \(DAQ\)](#)
 - [Discrete Low-pass Filter](#)
- [Database Design](#)
- [SQL Server](#)
- [OPC UA](#)
- [Datalogging System](#)
 - [Read Data from OPC UA Server](#)
 - [Save Data to SQL Server](#)
- [Alarm System](#)
 - [ASP.NET Core Web Applications](#)
- [Cyber Security](#)

<https://www.halvorsen.blog>

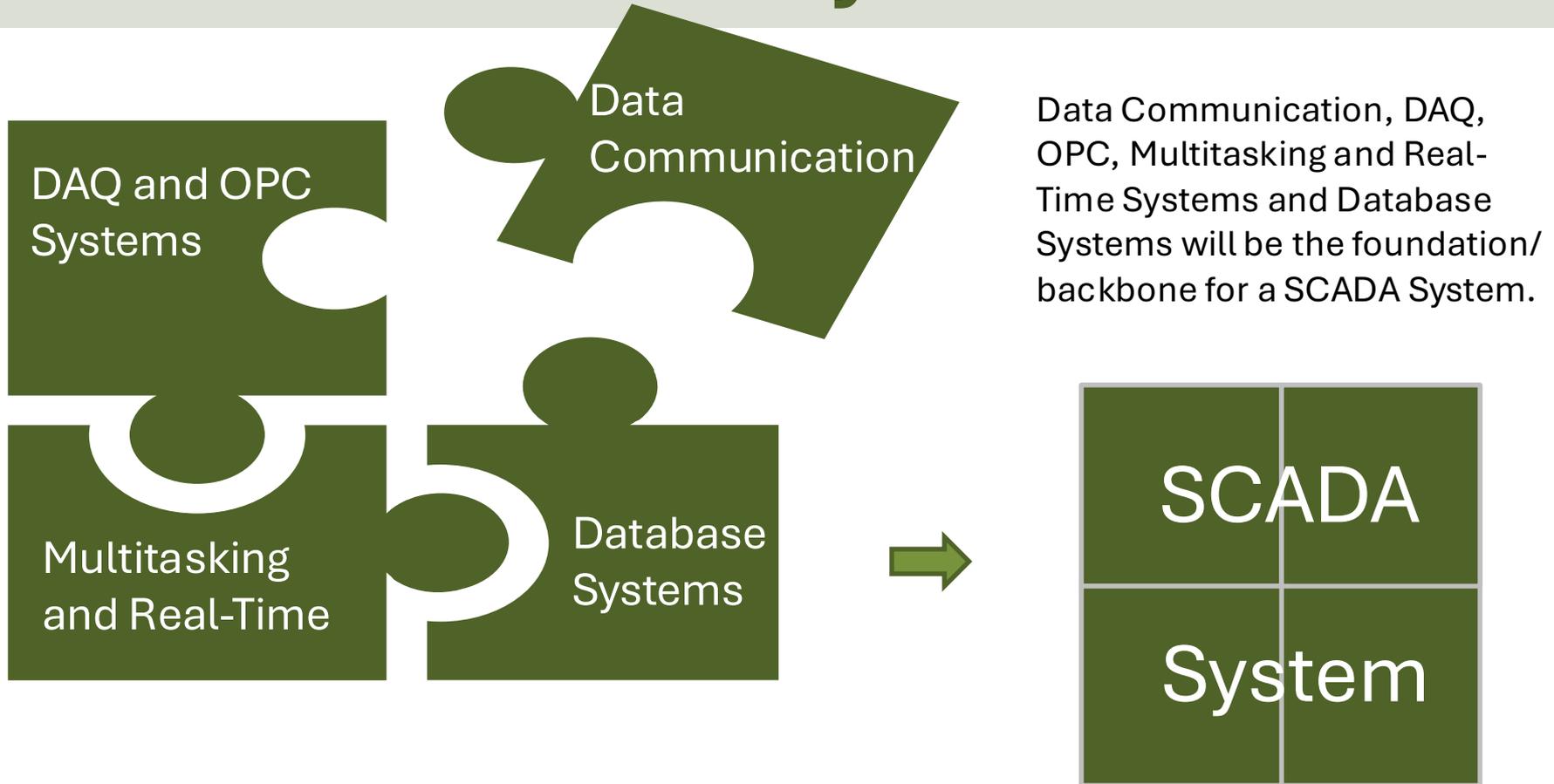
Introduction



Hans-Petter Halvorsen

[Table of Contents](#)

SCADA System



SCADA System Example

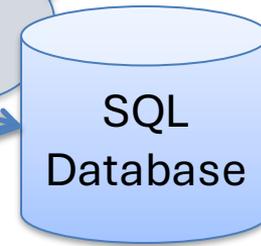
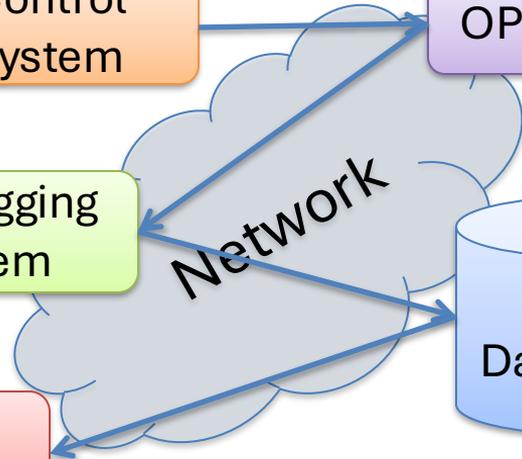
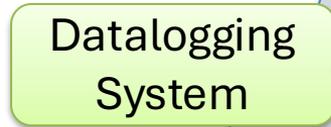
DAQ Device
(USB-6008)



Air Heater



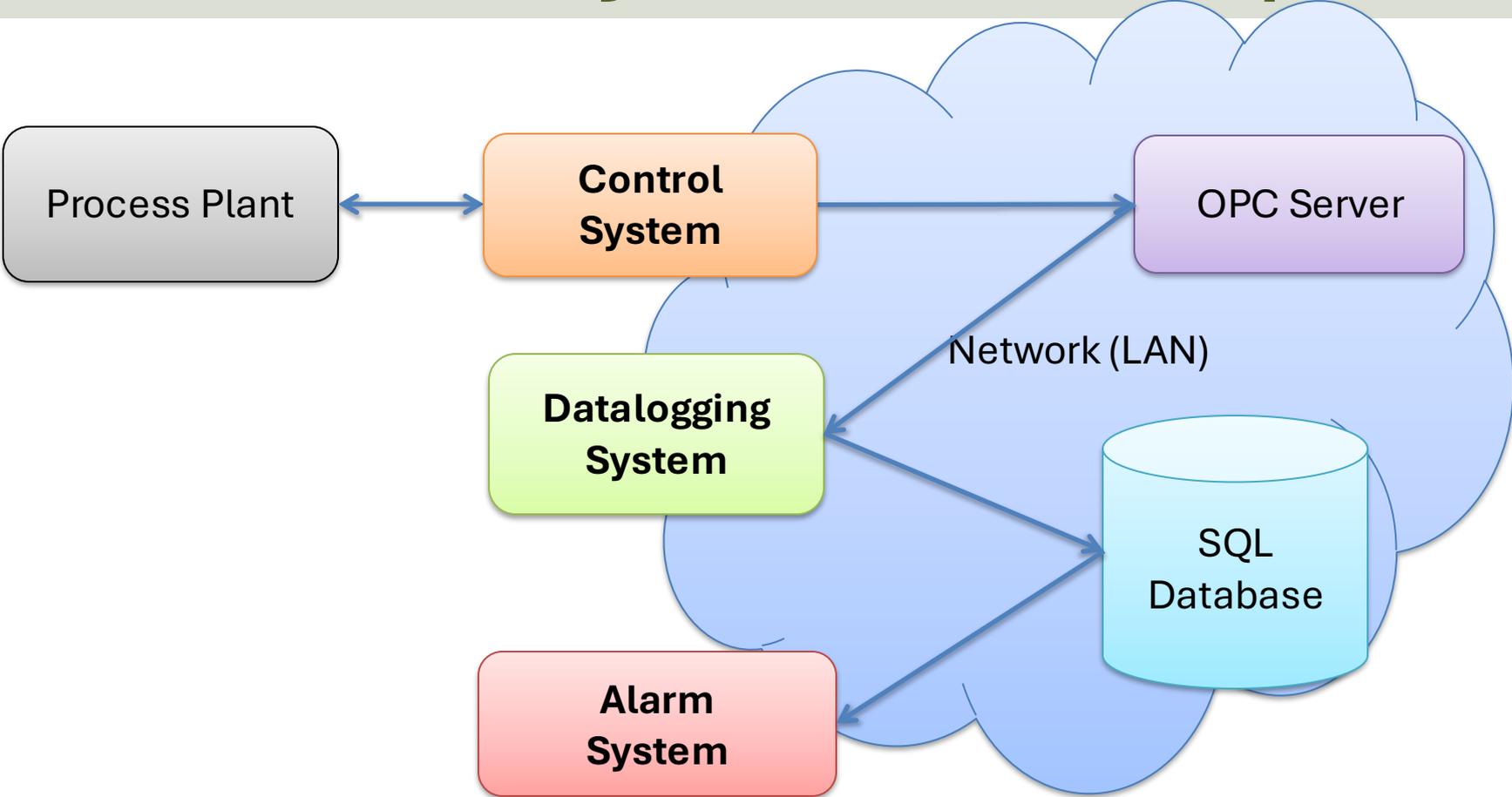
Simulator/Real Process



Microsoft
SQL Server

Alarm Trigger

SCADA System Example



<https://www.halvorsen.blog>

SCADA Systems

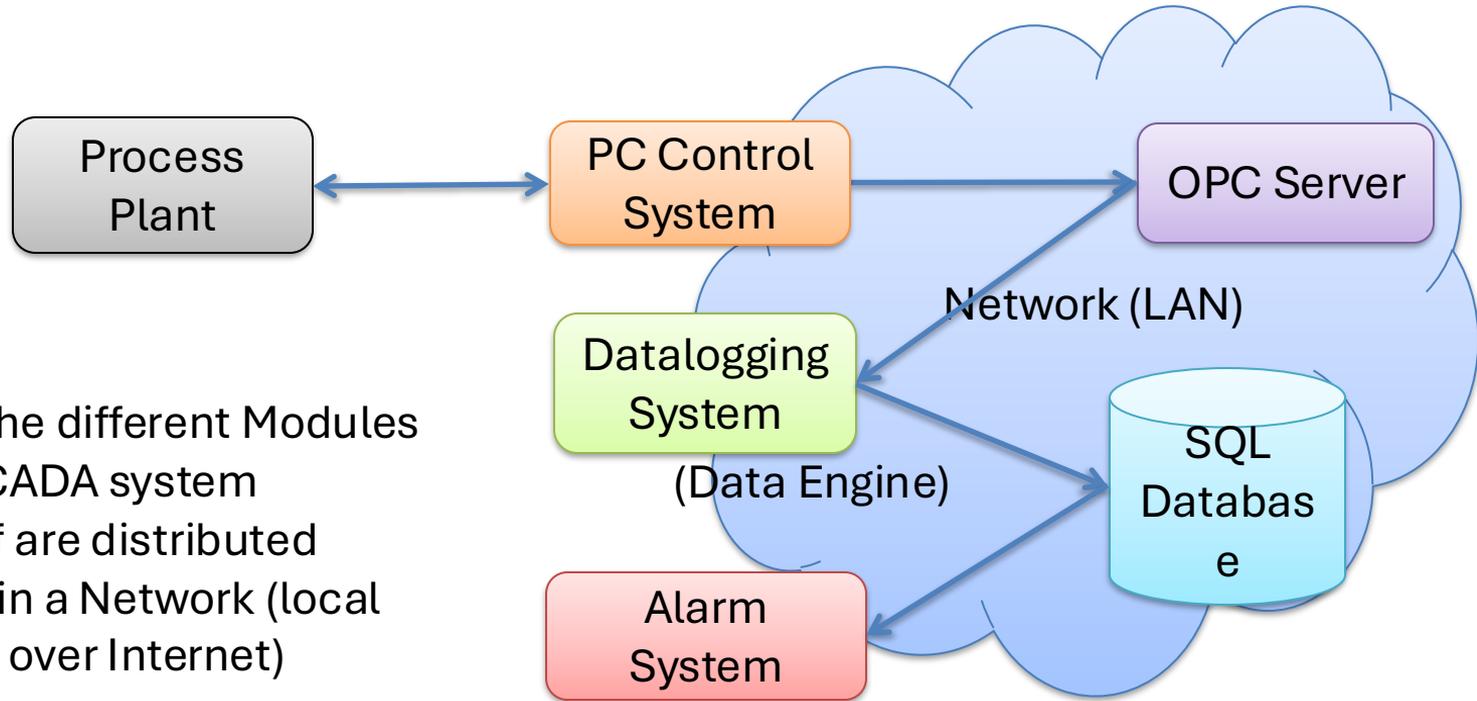
Supervisory Control and Data Acquisition



Hans-Petter Halvorsen

[Table of Contents](#)

SCADA System Overview



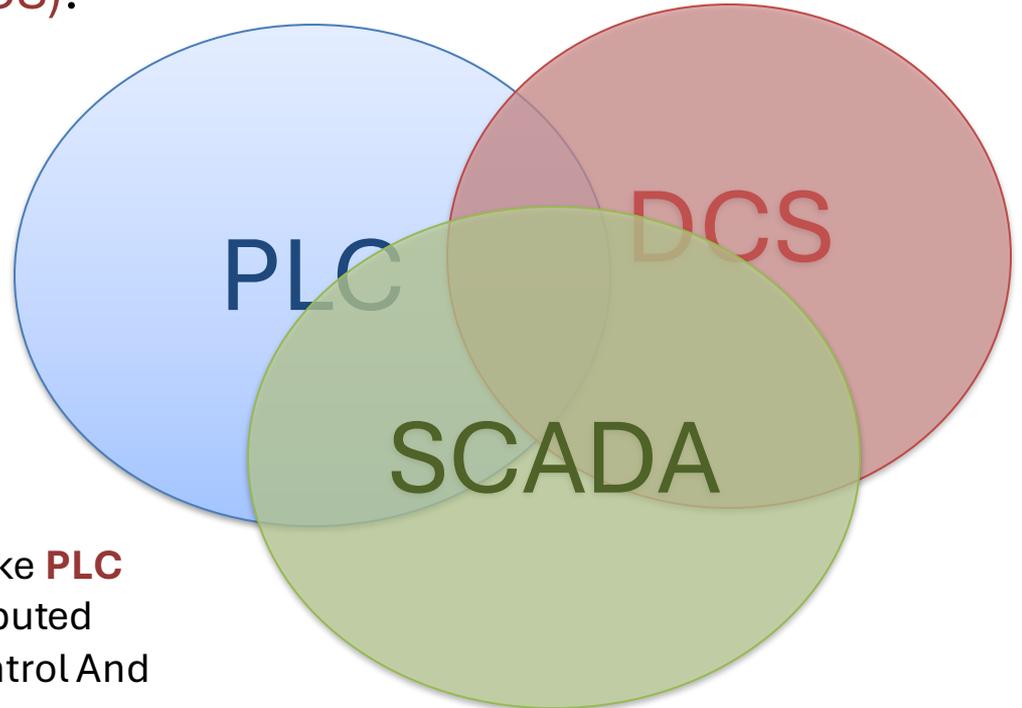
Typically, the different Modules that the SCADA system consists of are distributed physically in a Network (local network or over Internet)

SCADA Systems

SCADA (Supervisory Control And Data Acquisition) is a type of **Industrial Automation and Control System (IACS)**.

Industrial Automation and Control Systems (IACS) are computer systems that control and monitor industrial processes.

Industrial Automation and Control Systems, like **PLC** (Programmable Logic Controller), **DCS** (Distributed Control System) and **SCADA** (Supervisory Control And Data Acquisition) share many of the same features.



IACS Systems

Industrial Control Systems

Industrial Automation and Control Systems (IACS) are computer systems that control and monitor industrial processes.



Programmable Automation Controller (PAC) 4

cRIO

LabVIEW

1



5

I/O Module



Industrial PID Controller



DeltaV

PC based Control System/SCADA System (Supervisory Control And Data Acquisition)

PLC (Programmable Logic Controller)

3



Siemens PLC

2 Distributed Control Systems (DCS)



Controller I/O Modules

PC-based Control System

Industrial PID Controller



Embedded system with built-in PID algorithm, etc.

PID Control using PC and I/O Module

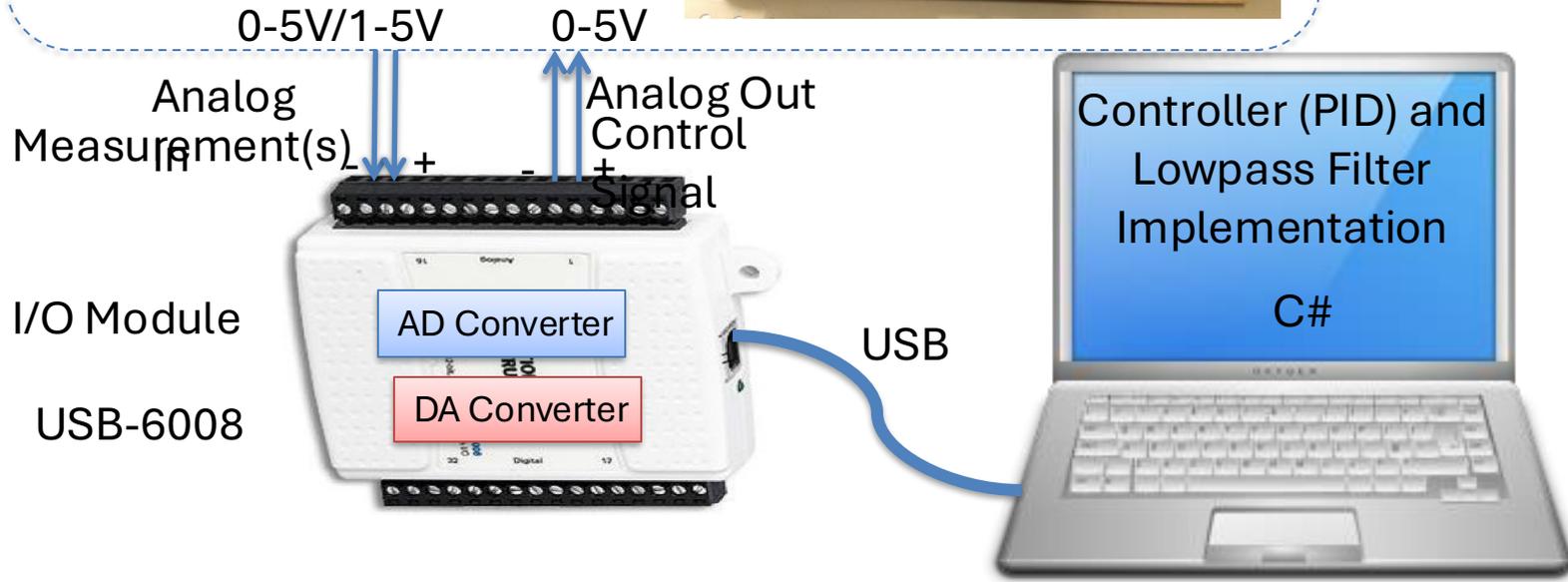


I/O Module

PC-based Control System Example



Process



PC-based Control System

PC with Control Application



USB-6008 DAQ



Air Heater Process



y

Process Value

Digital Signal

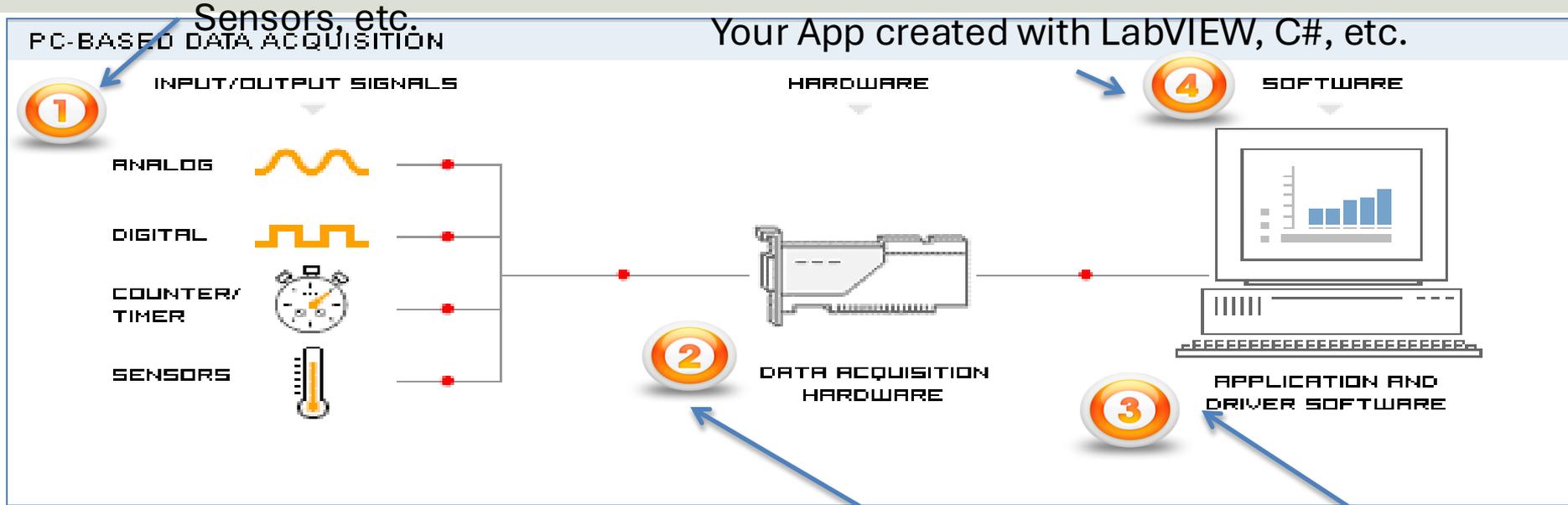


USB-6008 DAQ

Analog Measurement

Temperature

DAQ – Data Acquisition



A DAQ System consists of 4 parts:

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

NI USB 6008 DAQ Device

NI DAQmx Driver
or similar

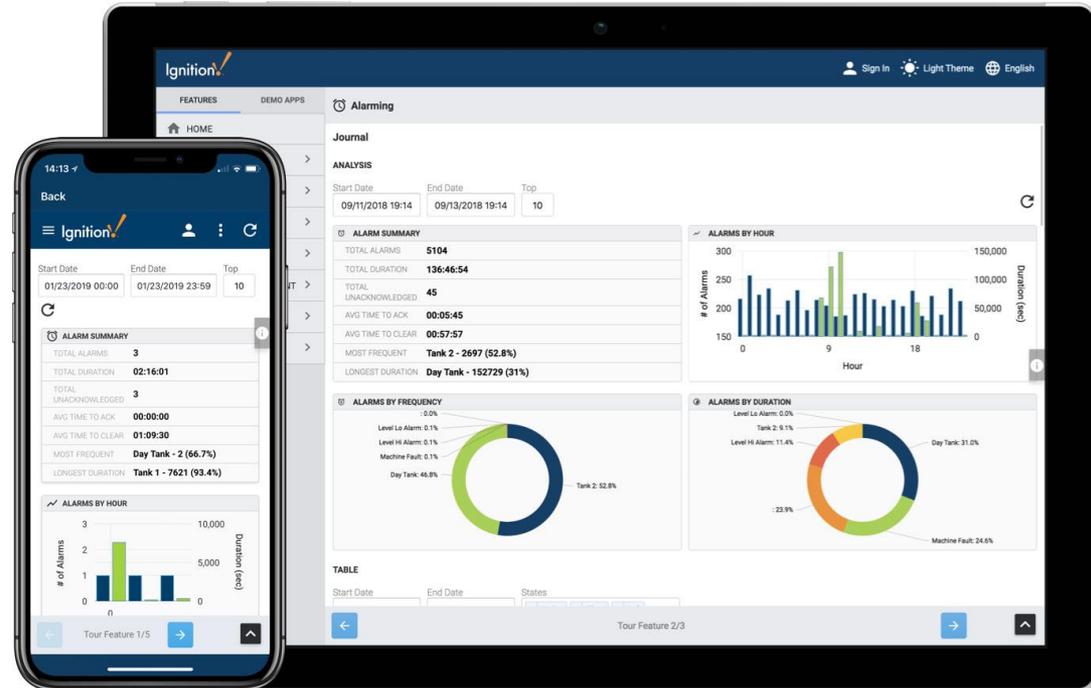
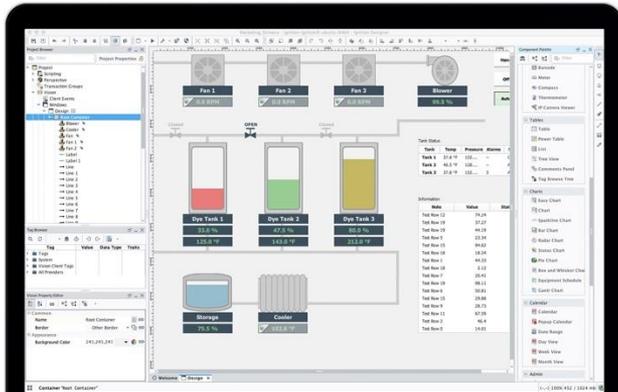
SCADA System

- The SCADA system typically contains different modules, such as:
 1. OPC Server
 2. A Database that stores all the necessary data
 - 3. Control System**
 - 4. Datalogging System**
 - 5. Alarm System**
- Note! They are typically implemented as separate applications because they should be able to run on different computers in a network (distributed system).

Ignition SCADA Software

- **Ignition** is an example of industrial SCADA software from “Inductive Automation”.
- You can download an unlimited trial version or the Ignition Maker Edition.
- <https://inductiveautomation.com>
- Feel free to integrate (or get ideas) your solution with existing industrial software like OPC software, PLC software or SCADA software like Ignition.

Ignition SCADA Software



<https://inductiveautomation.com>

<https://www.halvorsen.blog>

Air Heater

Hans-Petter Halvorsen



[Table of Contents](#)

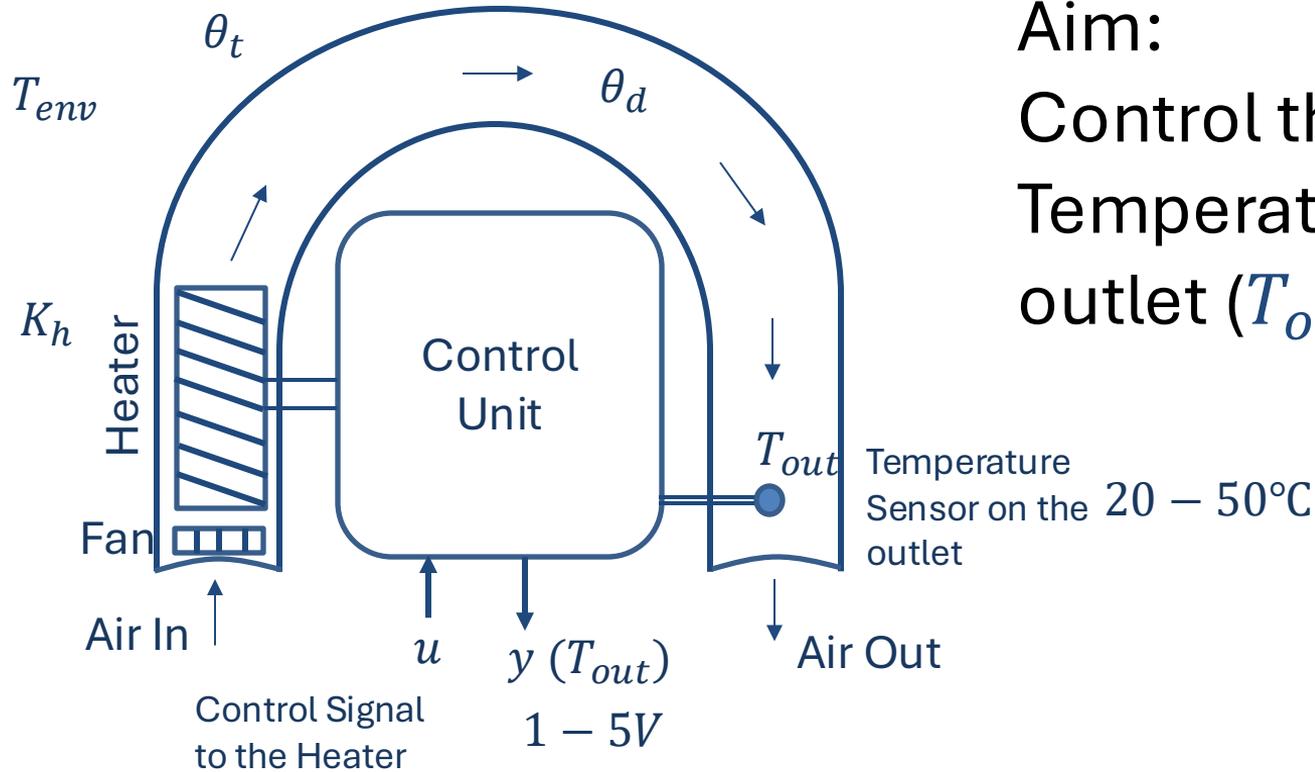
Air Heater



The Air Heater is a small-scale laboratory process suitable for learning about control systems

The purpose is to control the Temperature on the outflow of the Air Heater tube.

Air Heater



Aim:
Control the
Temperature on the
outlet (T_{out})

Air Heater Mathematical Model

$$\dot{T}_{out} = \frac{1}{\theta_t} \{-T_{out} + [K_h u(t - \theta_d) + T_{env}]\}$$

Example of Model
Parameters:

$$\theta_t = 22 \text{ sec}$$

$$\theta_d = 2 \text{ sec}$$

$$K_h = 3.5 \frac{^\circ\text{C}}{\text{V}}$$

$$T_{env} = 21.5 \text{ }^\circ\text{C}$$

Where:

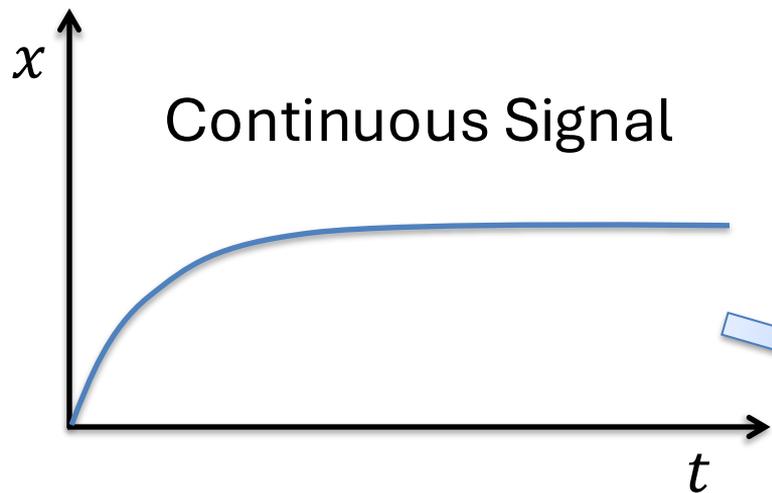
- T_{out} is the air temperature at the tube outlet
- $u [V]$ is the control signal to the heater
- $\theta_t [s]$ is the time-constant
- $K_h [deg\ C / V]$ is the heater gain
- $\theta_d [s]$ is the time-delay representing air transportation and sluggishness in the heater
- T_{env} is the environmental (room) temperature. It is the temperature in the outlet air of the air tube when the control signal to the heater has been set to zero for relatively long time (some minutes)

Use, e.g., these values:

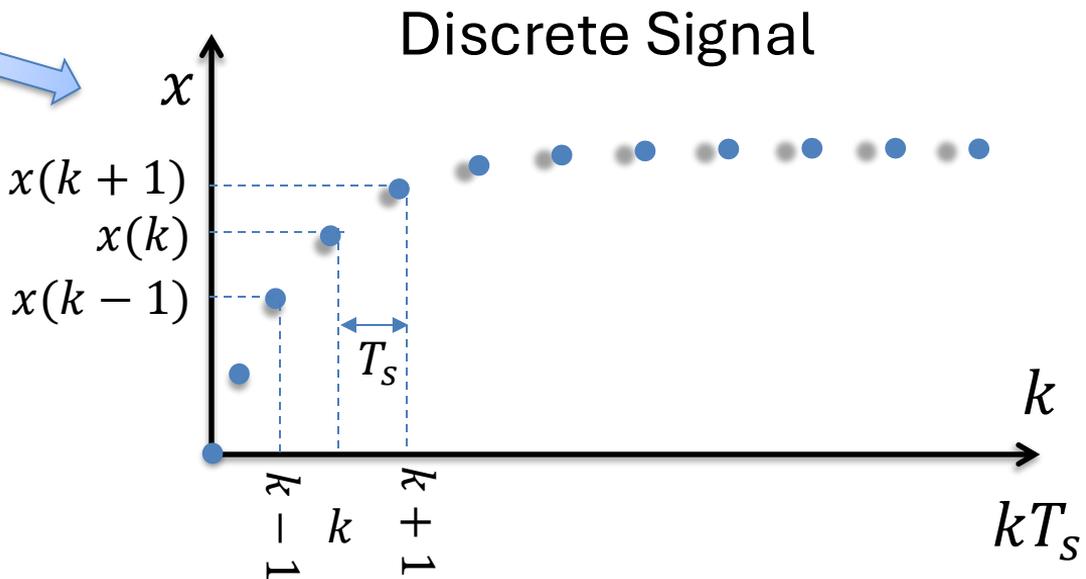
Air Heater Implementation

- The Air Heater is a standard 1.order process with time delay.
- In LabVIEW or C#, you can make a discrete version of the model and implement it. In LabVIEW you can, e.g., use a Formula Node.
- Discrete version: $T_{out}(k + 1) = \dots$
- Tip! The Time delay part of the system can be a bit “tricky“ to implement. Start by discarding/remove the Time delay part and implement and test it.
- Then later, try to implement and include the time delay part of the system. The time delay part can typically be implemented by creating an array/FIFO queue.

Continuous vs. Discrete Systems



A computer can only deal with discrete signals



T_s - Sampling Interval

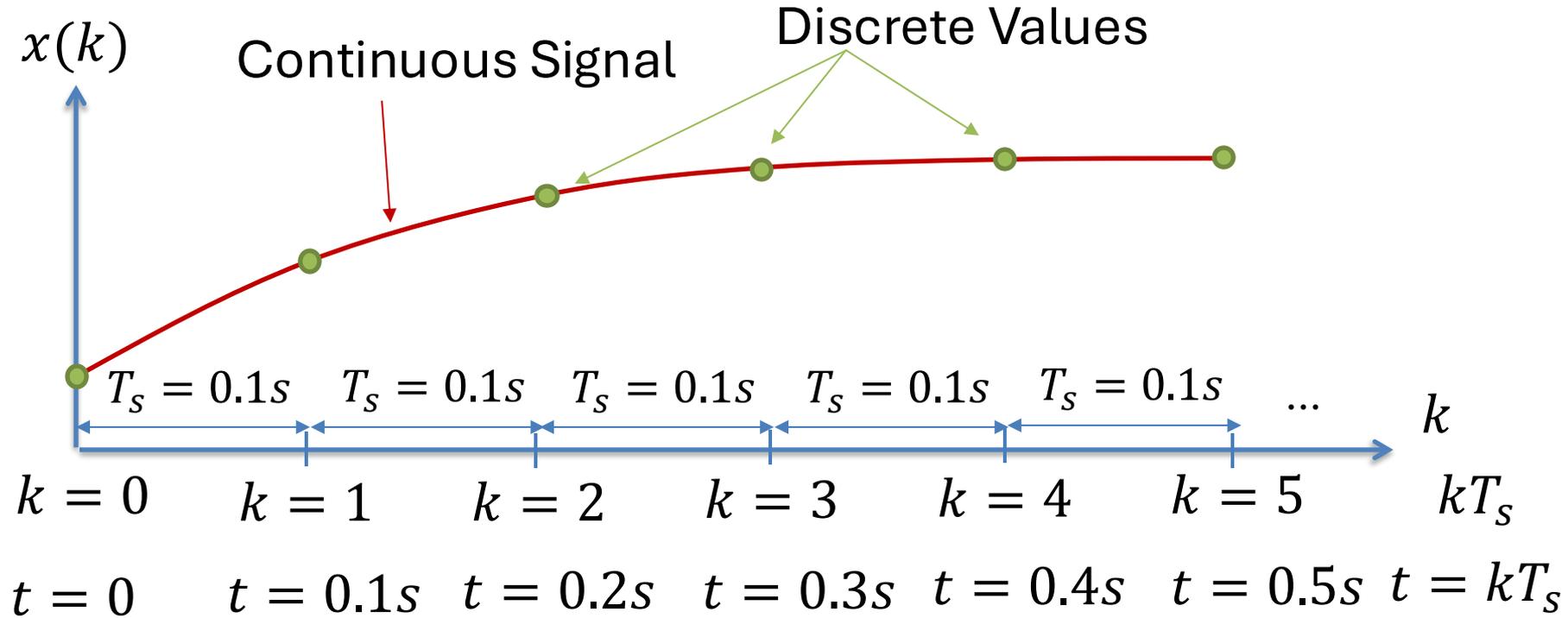
$x(k-1)$ - Previous Value

$x(k)$ - Current Value

$x(k+1)$ - Next Value

Continuous vs. Discrete Systems

In this Example we have used Sampling Interval $T_s = 0.1s$



Discretization

Continuous Model:

$$\dot{T}_{out} = \frac{1}{\theta_t} \{-T_{out} + [K_h u(t - \theta_d) + T_{env}]\}$$

We can use e.g., the Euler Approximation in order to find the discrete Model:

$$\dot{x} \approx \frac{x(k+1) - x(k)}{T_s}$$

T_s - Sampling Time $x(k)$ - Present value
 $x(k+1)$ - Next (future) value

The discrete Model will then be on the form:

$$x(k+1) = x(k) + \dots$$

We can then implement the discrete model in C#

Finding Model Parameters using “Trial and Error”

You may use, e.g., the following Parameters as a starting point, but since every Air Heater is unique, you may want to adjust these parameters. The “Trial and Error Method” may be an easy way to find the Parameters for your Process.

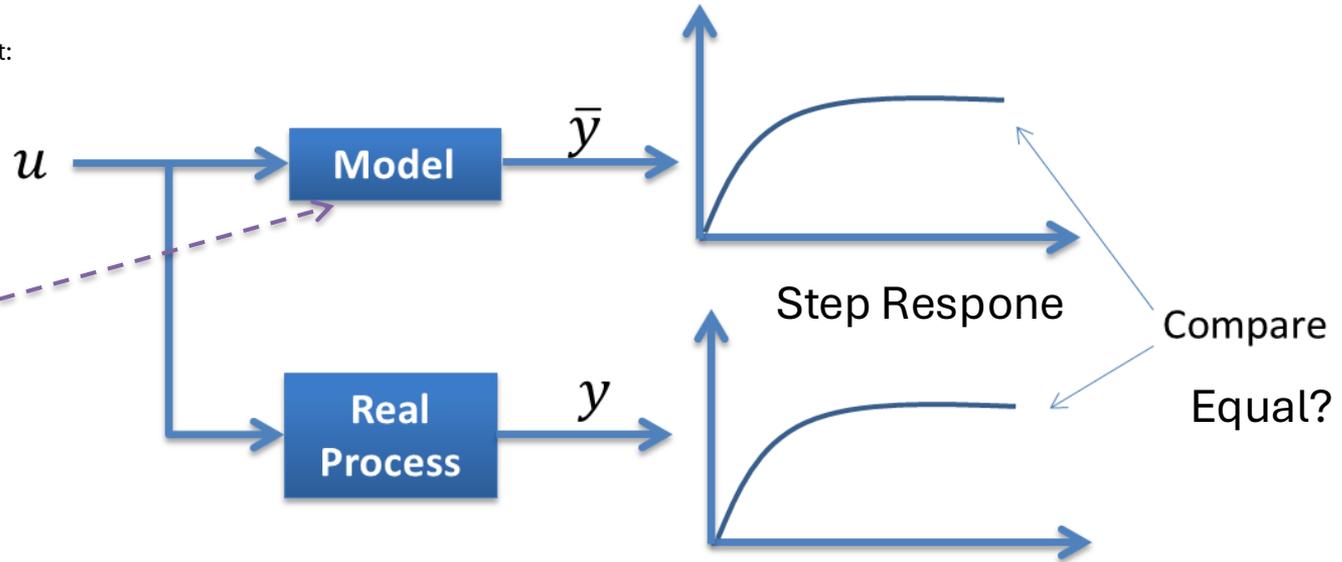
These values can be a good starting point:

$$\theta_t = 22 \text{ sec}$$

$$\theta_d = 2 \text{ sec}$$

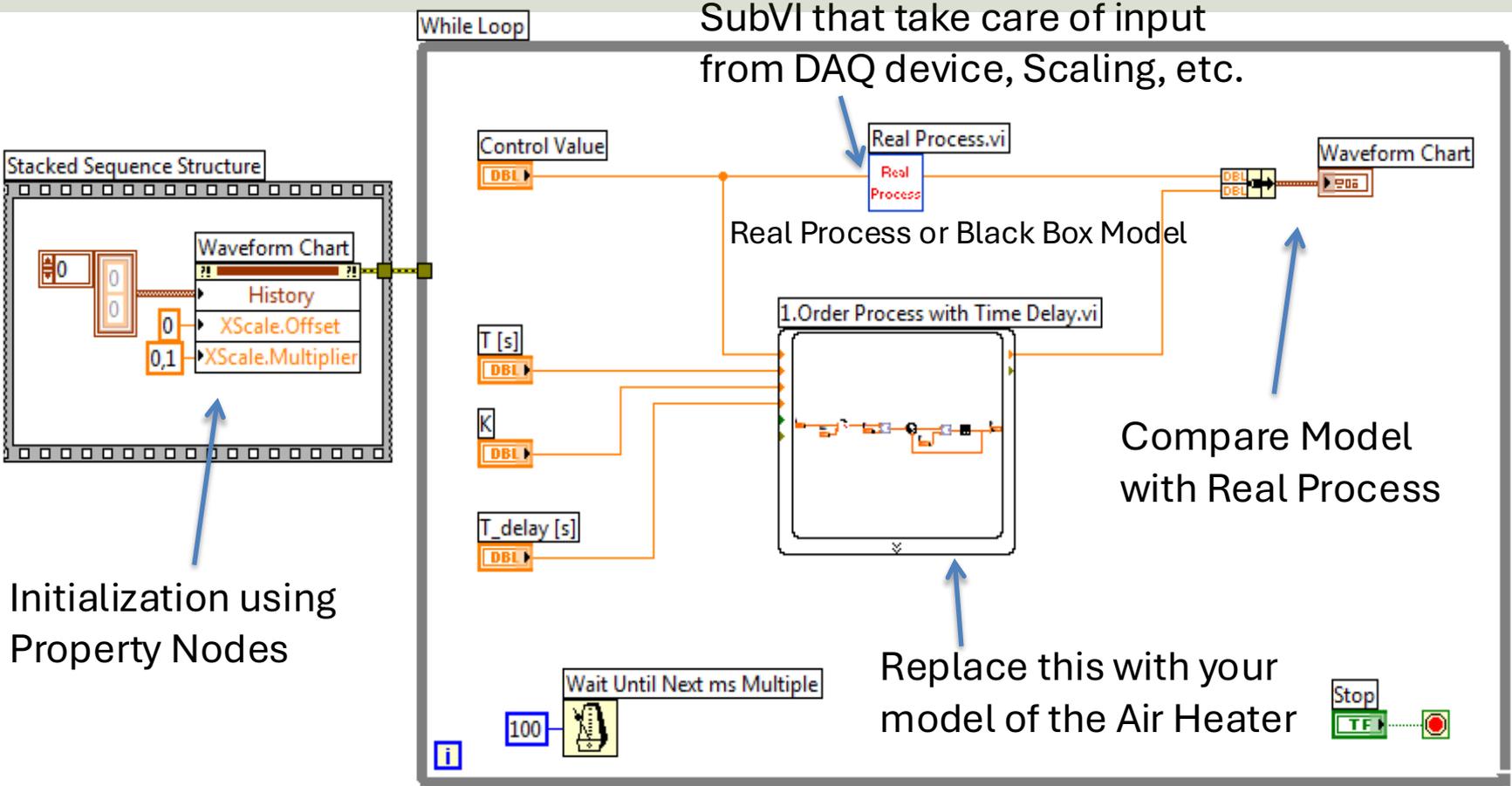
$$K_h = 3.5 \frac{\text{°C}}{\text{V}}$$

$$T_{env} = 21.5 \text{ °C}$$



Procedure: You run the Model and the Real Process in Parallel. Adjust the Model Parameters until the output of the Model and the Real Process is “equal”.

“Trial and Error” Example in LabVIEW



<https://www.halvorsen.blog>

Control System



Hans-Petter Halvorsen

[Table of Contents](#)

Control System



Computer



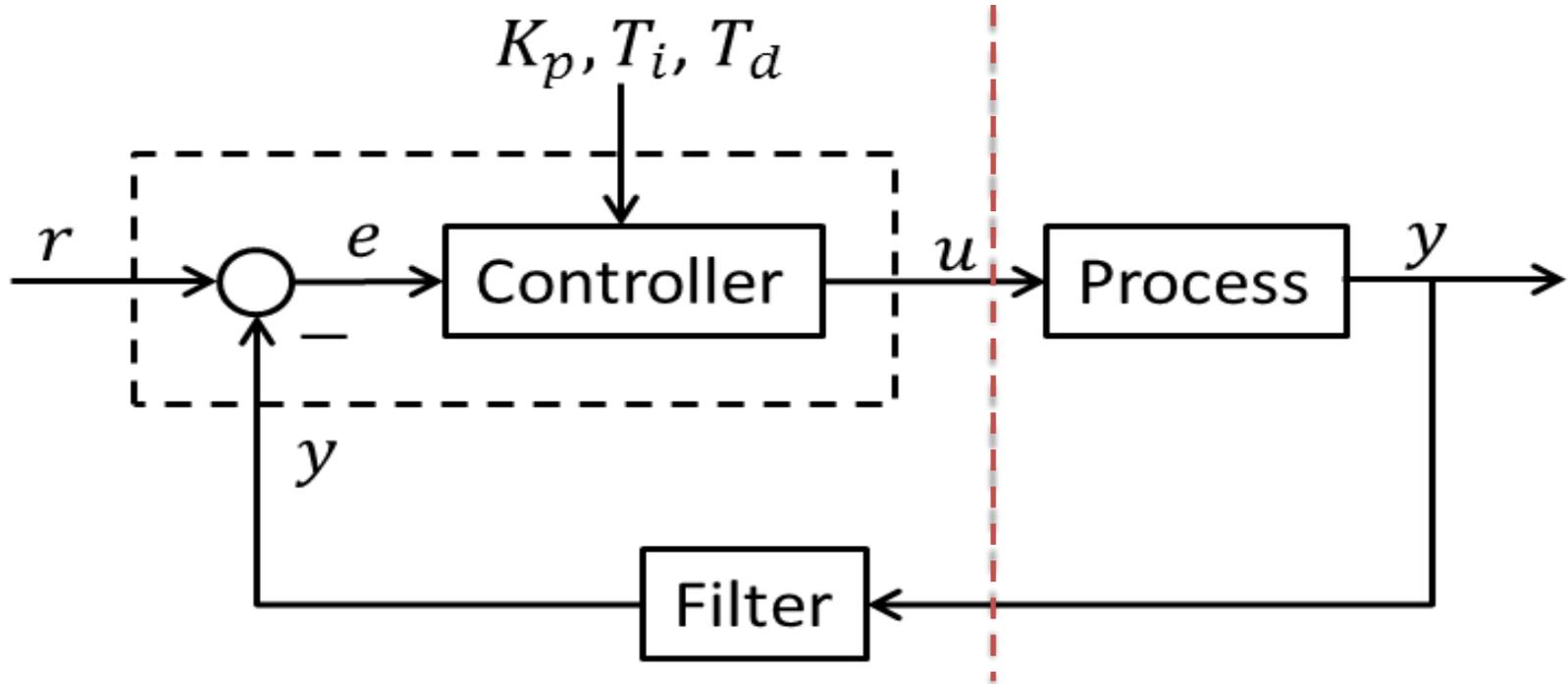
I/O Module



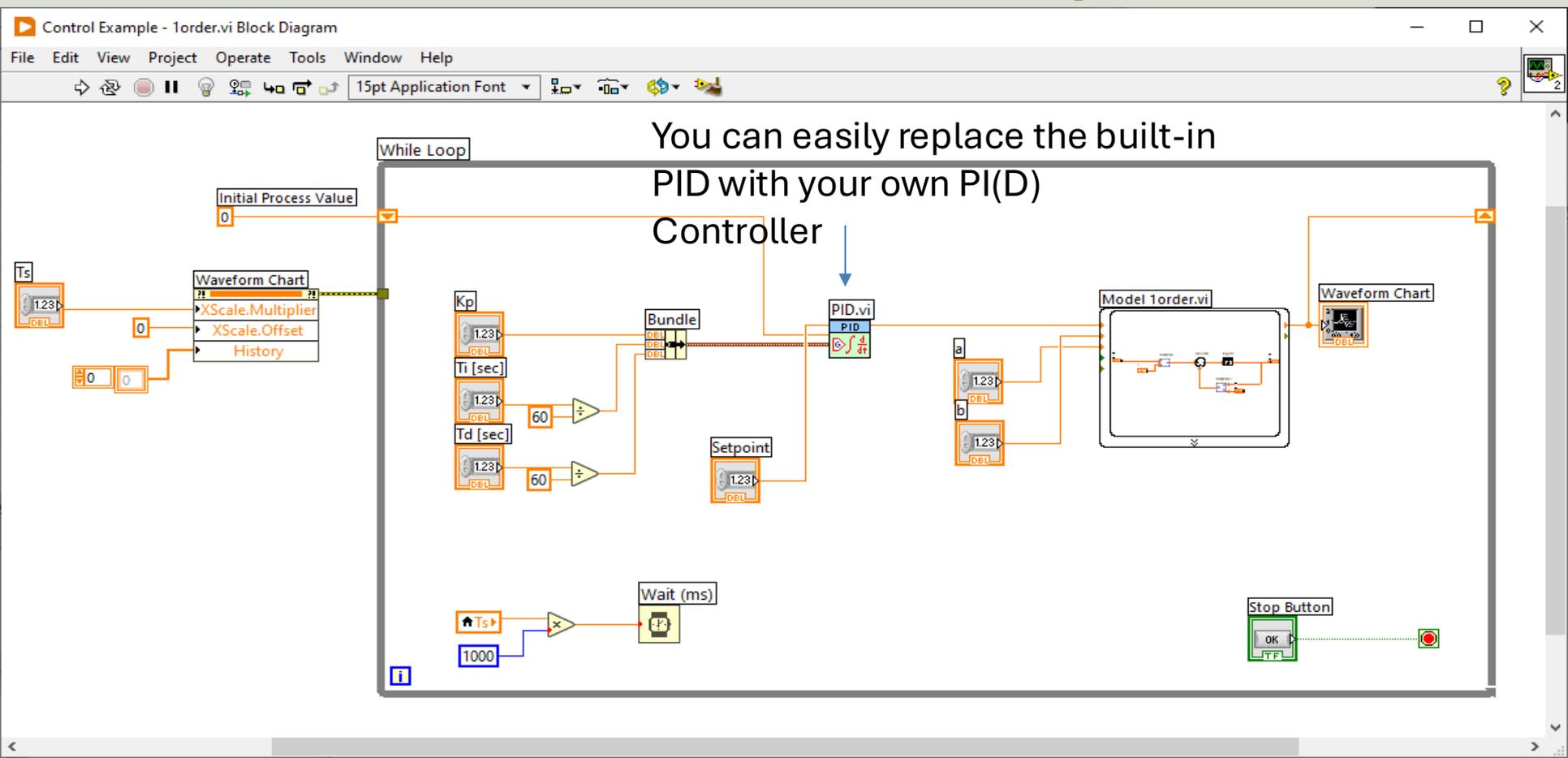
Air Heater

Control System Example

While the real process is continuous, normally the Controller and the Filter is implemented in a computer.



LabVIEW Control System



Control System in C#

1  Timer Select the “Timer” component in the

2 Initialization: Toolbox

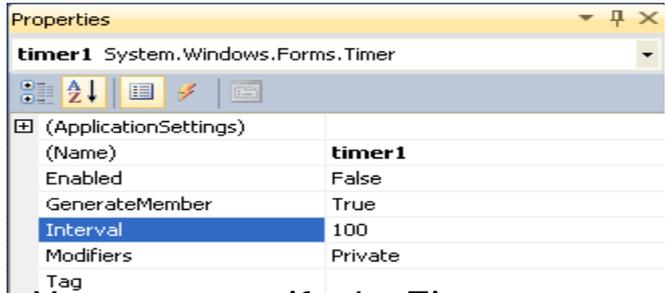
```
public Form1()
{
    InitializeComponent();
    timer1.Start();
}
```

4 Timer Event:

```
private void timer1_Tick(object sender, EventArgs e)
{
    ... //DAQ
    ... //Scaling
    ... //Control
    ... //Plot Data
    ... //Write to OPC
}
```

In Visual Studio you may want to use a Timer instead of a While Loop in order to read values at specific intervals.

Properties: 3

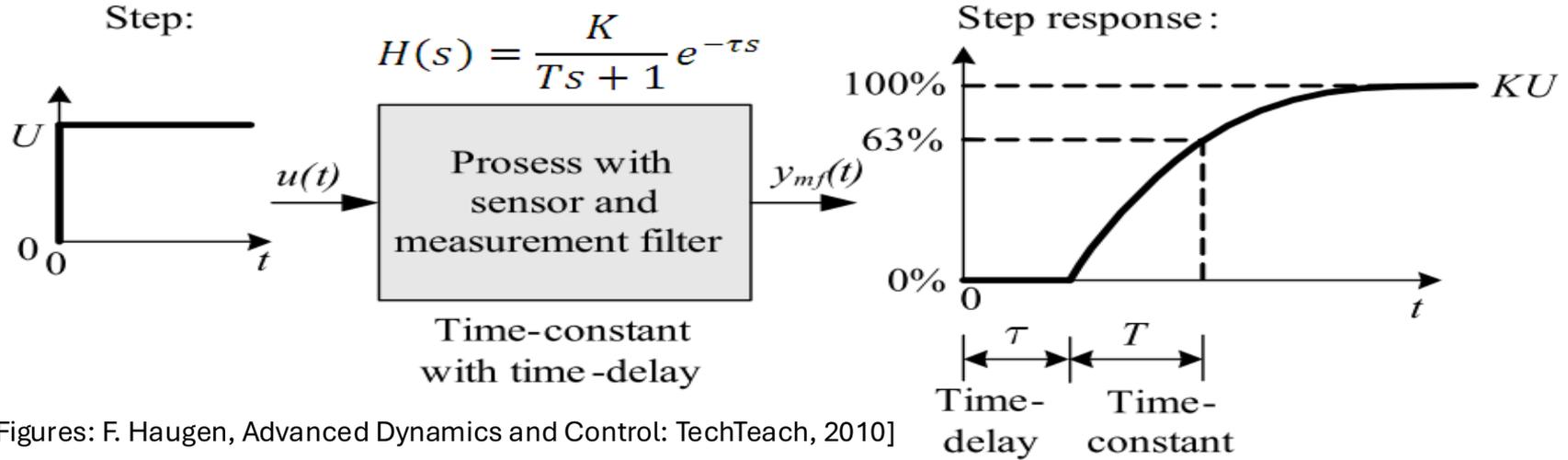


The screenshot shows the Properties window for a timer component named 'timer1' of type 'System.Windows.Forms.Timer'. The 'Interval' property is highlighted in blue and set to 100. Other visible properties include 'Enabled' (False), 'GenerateMember' (True), 'Modifiers' (Private), and 'Tag'.

Properties	
timer1 System.Windows.Forms.Timer	
[ApplicationSettings]	
(Name)	timer1
Enabled	False
GenerateMember	True
Interval	100
Modifiers	Private
Tag	

You may specify the Timer Interval in the Properties Window

PID Tuning with Skogestad



[Figures: F. Hagen, Advanced Dynamics and Control: TechTeach, 2010]

We can set, e.g., $T_c=10$ sec.
and $c=1.5$.

You may use other values if
these values give a poor result.

Process type	$H_{psf}(s)$ (process)	K_p	T_i	T_d
Integrator + delay	$\frac{K}{s} e^{-\tau s}$	$\frac{1}{K(T_C + \tau)}$	$c(T_C + \tau)$	0
Time-constant + delay	$\frac{K}{Ts+1} e^{-\tau s}$	$\frac{1}{K(T_C + \tau)}$	$\min [T, c(T_C + \tau)]$	0
Integr + time-const + del.	$\frac{K}{(Ts+1)s} e^{-\tau s}$	$\frac{1}{K(T_C + \tau)}$	$c(T_C + \tau)$	T
Two time-const + delay	$\frac{K}{(T_1s+1)(T_2s+1)} e^{-\tau s}$	$\frac{1}{K(T_C + \tau)}$	$\min [T_1, c(T_C + \tau)]$	T_2
Double integrator + delay	$\frac{K}{s^2} e^{-\tau s}$	$\frac{1}{4K(T_C + \tau)^2}$	$4(T_C + \tau)$	$4(T_C + \tau)$

Table 1: Skogestad's formulas for PI(D) tuning.

<https://www.halvorsen.blog>

Discrete PI(D) Controller

Hans-Petter Halvorsen



[Table of Contents](#)

Discrete PI Controller Example

Continuous PI Controller:

$$u(t) = u_0 + K_p e(t) + \frac{K_p}{T_i} \int_0^t e d\tau$$



$$\dot{u} = \dot{u}_0 + K_p \dot{e} + \frac{K_p}{T_i} e$$

We use the Euler Backward method:

$$\dot{x} = \frac{x_k - x_{k-1}}{T_s}$$

$$\frac{u_k - u_{k-1}}{T_s} = \frac{u_{0,k} - u_{0,k-1}}{T_s} + K_p \frac{e_k - e_{k-1}}{T_s} + \frac{K_p}{T_i} e_k$$



$$u_k = u_{k-1} + u_{0,k} - u_{0,k-1} + K_p(e_k - e_{k-1}) + \frac{K_p}{T_i} T_s e_k$$

We may set:

$$\Delta u_k = u_k - u_{k-1}$$

This gives the following discrete PI algorithm:

$$e_k = r_k - y_k$$

$$\Delta u_k = u_{0,k} - u_{0,k-1} + K_p(e_k - e_{k-1}) + \frac{K_p}{T_i} T_s e_k$$

$$u_k = u_{k-1} + \Delta u_k$$

This algorithm can be easily implemented in a Programming language

Discrete PI Controller Example

```
class PidController
{
    public double r;
    public double Kp;
    public double Ti;
    public double Ts;

    public double PiController(double y)
    {
        double e;
        double u;

        e = r - y;
        u = ...;
        return u;
    }
}
```

Note! This is just a basic Example

<https://www.halvorsen.blog>

Data Acquisition

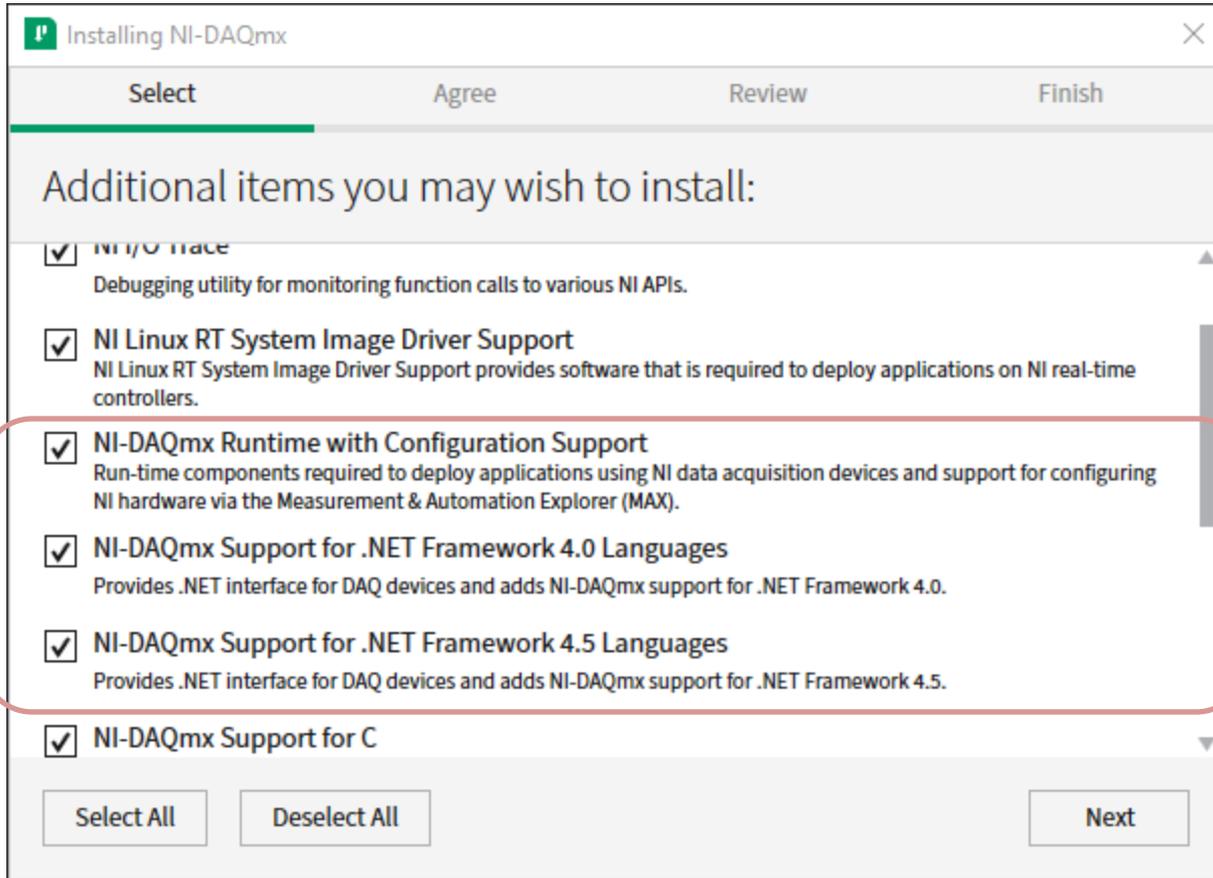
DAQ



Hans-Petter Halvorsen

[Table of Contents](#)

NI-DAQmx Installation



Make sure to add support for Visual Studio/.NET during installation of the NI-DAQmx software

Add DAQ Reference

You need to add the reference **NationalInstruments.DAQmx.dll** by right-clicking in the Solution Explorer and select "Add Reference". This dll is installed by the NI-DAQmx driver and are typically installed within `C:/Program Files (x86)/National Instruments/..`

The screenshot shows the 'Add Reference' dialog box in Visual Studio. The 'Browse' tab is active, displaying a list of files. The file 'NationalInstruments.DAQmx.dll' is selected, and its path is shown as 'C:\Program Files (x86)\National Instru...'. The file details on the right indicate it was created by National Instruments and has a file version of 23.3.45.49311.

Name	Path
<input checked="" type="checkbox"/> NationalInstruments.DAQ... COMMLib.dll	C:\Program Files (x86)\National Instru... C:\Users\hansha\OneDrive\Programmin...

Name: NationalInstruments.DAQmx.dll
Created by: National Instruments
File Version: 23.3.45.49311

NationalInstruments.DAQmx.dll

Note! NI-DAQmx is so far not supported for .NET 5 or higher, so you need to use the “Windows Forms App (.NET Framework)” Template

C:\Program Files (x86)\National Instruments\MeasurementStudioVS2012\DotNET\Assemblies\Current

Browse... OK Cancel

DAQ in C# with DAQmx

```
double ReadDaqData()  
{  
    Task analogInTask = new Task();  
  
    AIChannel myAIChannel;  
  
    myAIChannel = analogInTask.AIChannels.CreateVoltageChannel(  
        "dev1/ai0",  
        "myAIChannel",  
        AITerminalConfiguration.Differential,  
        0,  
        5,  
        AIVoltageUnits.Volts  
    );  
  
    AnalogSingleChannelReader reader = new  
        AnalogSingleChannelReader(analogInTask.Stream);  
  
    double analogDataIn = reader.ReadSingleSample();  
    return analogDataIn;  
}
```

Analog In Example

Note! The physical wiring on the DAQ device can either be “Differential” or “RSE”. Make sure your code and the wiring uses the same configuration.

DAQ in C# with DAQmx

```
WriteDagData(double analogDataOut)
{
    Task analogOutTask = new Task();

    AOChannel myAOChannel;

    myAOChannel = analogOutTask.AOChannels.CreateVoltageChannel (
        "dev1/ao0",
        "myAOChannel",
        0,
        5,
        AOVoltageUnits.Volts
    );

    AnalogSingleChannelWriter writer = new
        AnalogSingleChannelWriter(analogOutTask.Stream);

    writer.WriteSingleSample(true, analogDataOut);
}
```

Analog Out Example

<https://www.halvorsen.blog>

Discrete Lowpass Filter

Hans-Petter Halvorsen



[Table of Contents](#)

Discrete Lowpass Filter Example

Lowpass Filter Transfer function:

$$H(s) = \frac{y(s)}{u(s)} = \frac{1}{T_f s + 1}$$

Inverse Laplace the differential Equation:

$$T_f \dot{y} + y = u$$

We use the Euler Backward method:

$$\dot{x} = \frac{x_k - x_{k-1}}{T_s}$$

This gives:

$$T_f \frac{y_k - y_{k-1}}{T_s} + y_k = u_k$$

$$y_k = \frac{T_f}{T_f + T_s} y_{k-1} + \frac{T_s}{T_f + T_s} u_k$$

We define:

$$\frac{T_s}{T_f + T_s} \equiv a$$

This gives:

$$y_k = (1 - a)y_{k-1} + au_k$$

Filter output

Noisy input signal

$$T_s \leq \frac{T_f}{5}$$

This algorithm can be easily implemented in a Programming language

Discrete Lowpass Filter Example

```
class Filter
{
    public double yk;
    public double Ts;
    public double Tf;

    public double LowPassFilter(double yFromDaq)
    {
        double a;
        double yFiltered;

        a = Ts / (Ts + Tf);
        yFiltered = (1 - a) * yk + a * yFromDaq;
        yk = yFiltered;
        return yFiltered;
    }
}
```

Note! This is just a simple Example

<https://www.halvorsen.blog>

Database Design



Hans-Petter Halvorsen

[Table of Contents](#)

Database Examples

Tag Configuration:

The TAG table(s) could e.g., have the following columns:

- TagId (int, Primary Key, Identity)
- TagName (varchar)
- ItemId (varchar) (OPC)
- ItemUrl (varchar) (OPC)
- Description (varchar)
- etc.

Alarm Configuration & Alarm Data:

Important fields in an alarm handling system could be:

- AlarmId
- Activation Time
- Acknowledge Time and Person
- Type of Alarm
- Which Tag
- Alarm Limits
- Textual Description
- etc.

Here are some examples of functionality of the SCADA system and information that typically could be stored in the Database.

Tag Data:

Create one or more tables used for logging the Tag Values into the Database. Example of information:

- Value
- Timestamp
- Status (e.g., “Active”, “Not Active”)
- Quality (e.g., “Good”, “Bad”)
- etc.

Database Examples

The alarm system will be responsible for the warnings and the alarms in a monitoring and control system. An alarm system contains different **Alarms** and **Warnings** like:

- Timeout; no input from a sensor or another computer system within a specific amount of time,
- High High (HH) or Low Low (LL) alarm; a critical alarm condition,
- High (H) or Low alarm (L)
- I/O device errors
- System device errors
- etc.

Make sure your Alarm tables and system can handle some of these kinds of alarms and warnings.

An Alarm System use different Alarm Devices, such as, e.g.,

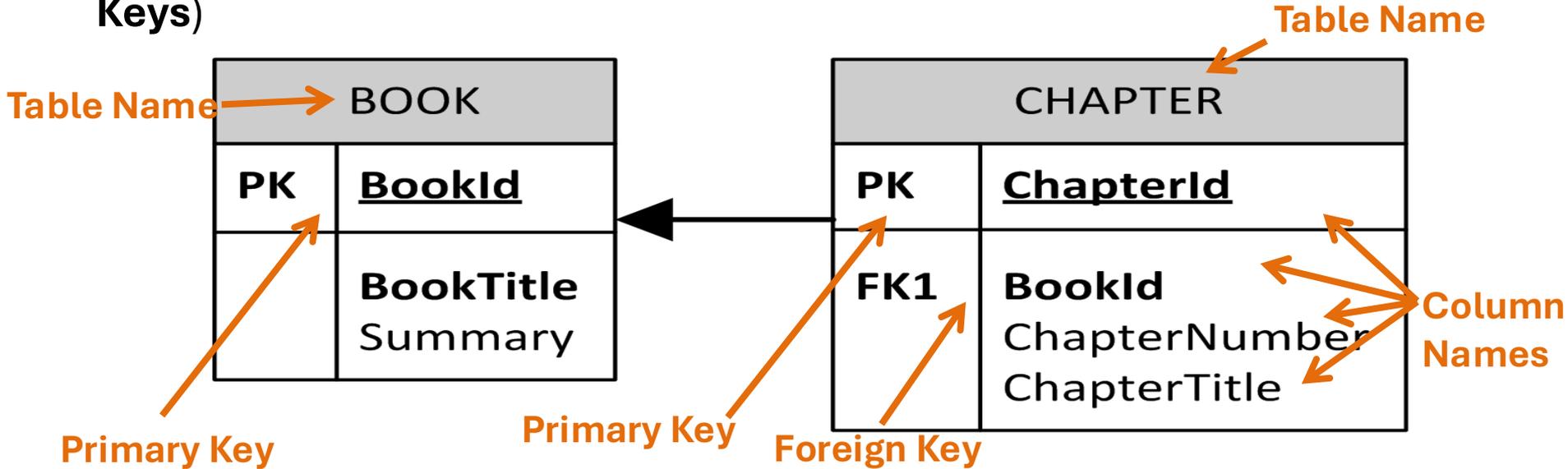
- Screen; display the alarms
- Keyboard; alarm operations
- Horn; indicate an active alarm, or security alarm
- Lamp; indicate an active alarm, or an active alarm by blinking and an acknowledge alarm by a steady light
- Printer; logging of the alarm states
- SMS
- E-mail
- Etc.

Make use of one or more of these alarm devices in your Alarm Handling and Management System.

Database Design – ERD

ER Diagram (Entity-Relationship Diagram, ERD)

- Used for Design and Modeling of Databases.
- Specify Tables and **relationship** between them (**Primary Keys** and **Foreign Keys**)



Relational Database. In a relational database all the tables have one or more relation with each other using Primary Keys (PK) and Foreign Keys (FK). Note! You can only have one PK in a table, but you may have several FK's.

Database - “Best Practice”

- **Tables:** Use upper case and singular form in table names – not plural, e.g., “STUDENT” (not “students”)
- **Columns:** Use Pascal notation, e.g., “StudentId”
- **Primary Key:**
 - If the table name is “COURSE”, name the Primary Key column “CourseId”, etc.
 - “Always” use Integer and Identity(1,1) for Primary Keys. Use UNIQUE constraint for other columns that needs to be unique, e.g. “RoomNumber”
- Specify **Required** Columns (NOT NULL) – i.e., which columns that need to have data or not
- Standardize on few/these **Data Types:** *int, float, varchar(x), datetime, bit*
- Use English for table and column names
- Avoid abbreviations! (Use “RoomNumber” – not “RoomNo” “RoomNr”)

<https://www.halvorsen.blog>

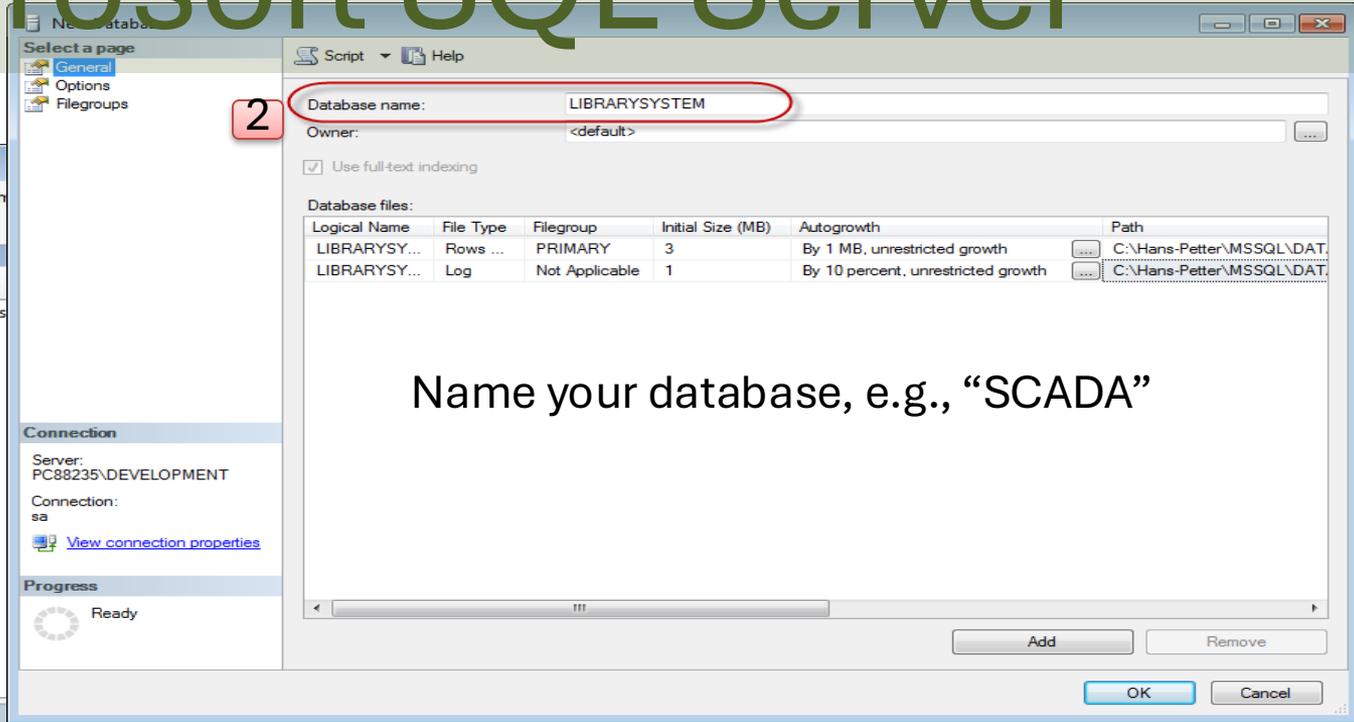
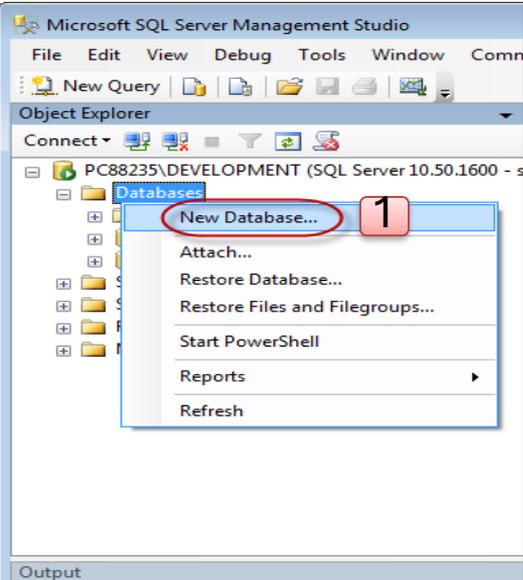
SQL Server



Hans-Petter Halvorsen

[Table of Contents](#)

Microsoft SQL Server



How-To Create a New Database

Microsoft SQL Server

The screenshot shows the Microsoft SQL Server Management Studio interface. The 'Object Explorer' on the left shows a tree view of the server hierarchy. The 'Query Editor' in the center contains a SQL query. The 'Results' pane at the bottom displays the output of the query. The 'Properties' pane on the right shows connection details.

3 Microsoft SQL Server Management Studio

3 New Query

1 Your SQL Server

2 Your Database

4 Write your Query here

5 The result from your Query

```
select * from SCHOOL
```

	SchoolId	SchoolName	Description	Address	Phone	PostCode	PostAddress
1	1	TUC	The best school	Telemark	NULL	NULL	NULL
2	2	MIT	OK School	USA	NULL	NULL	NULL
3	3	NTNU	The second best school	Trondheim	NULL	NULL	NULL
4	4	University of Oslo	The third best school	Oslo	NULL	NULL	NULL

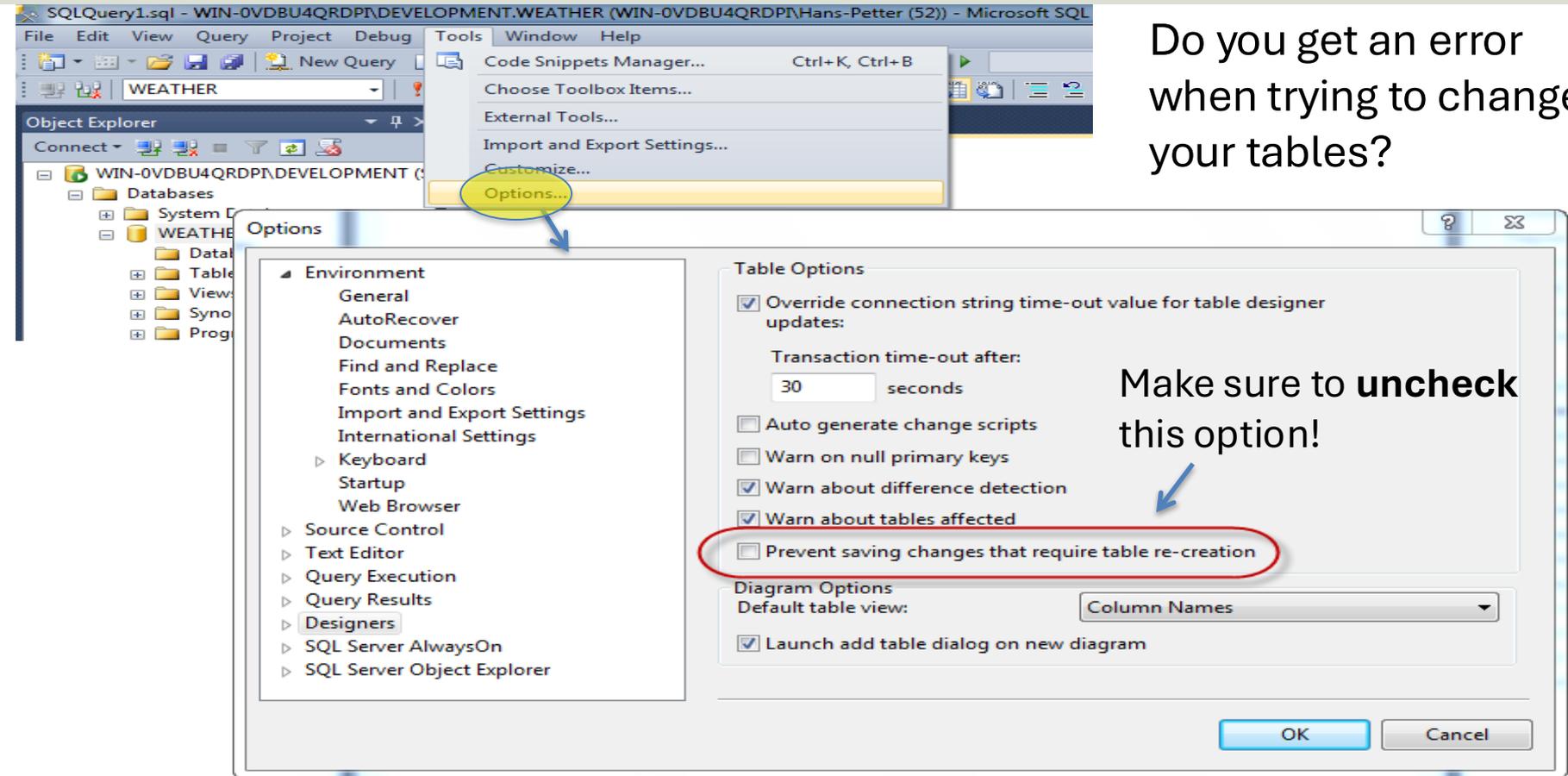
Query executed successfully. PC88235\DEVELOPMENT (10.50 ... sa (52) SCHOOL 00:00:00 4 rows

Properties
Current connection parameters
Aggregate Status
Connection
Connection Details

Microsoft SQL Server

Do you get an error when trying to change your tables?

Make sure to **uncheck** this option!



<https://www.halvorsen.blog>

OPC UA



Hans-Petter Halvorsen

[Table of Contents](#)

<https://www.halvorsen.blog>

OPC UA in LabVIEW

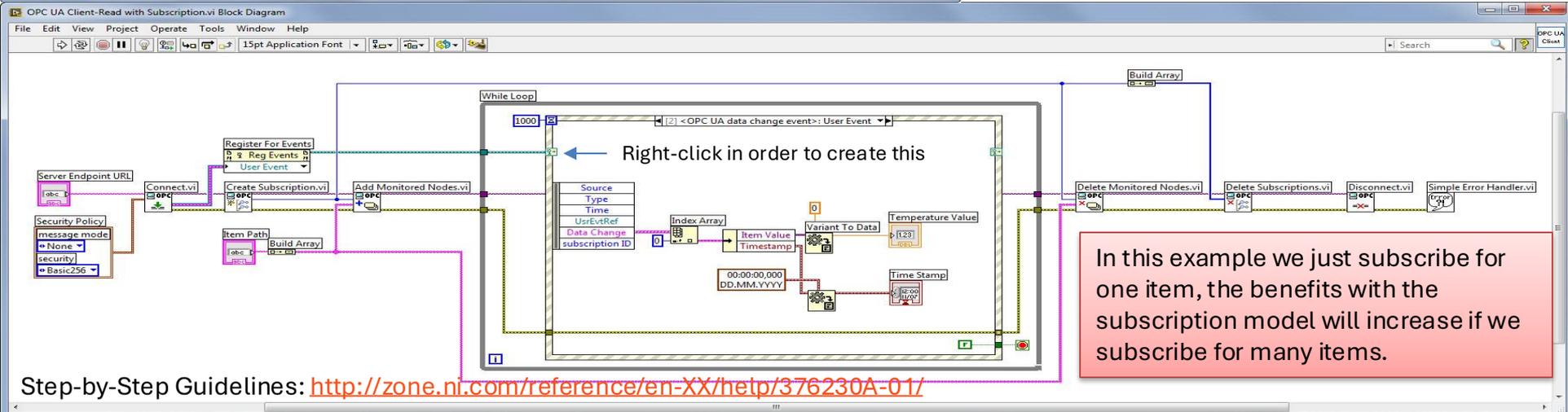
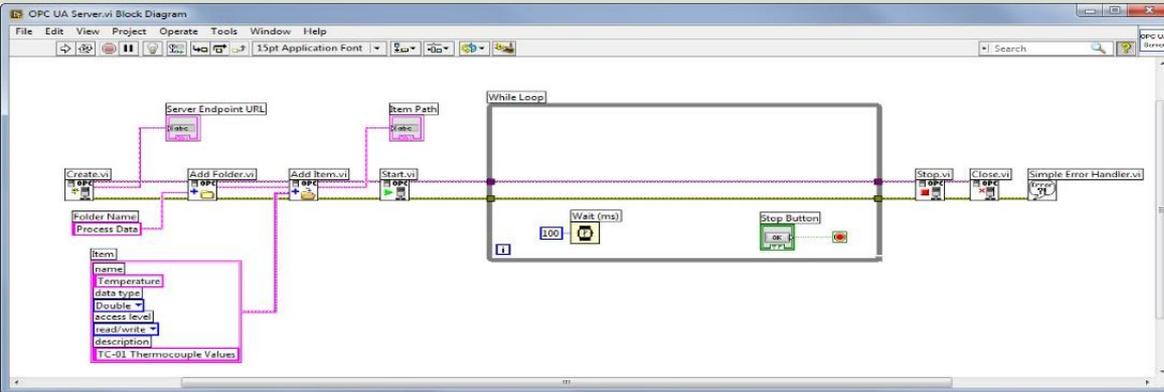
[Table of Contents](#)

Hans-Petter Halvorsen



OPC UA Client with Subscription

Here you read data on the client only when the value on the server is changed.



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

<https://www.halvorsen.blog>

OPC UA .NET SDK

& OPC UA Server Simulator



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>

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/opcuanel-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)

OPC UA .NET SDK Installation

The screenshot shows the Visual Studio IDE with the NuGet Package Manager open for the 'OPCUAClient' project. The 'Browse' tab is selected, showing a list of packages. The package 'Opc.UaFx.Client' by Traeger.de is highlighted with a red box. The right pane displays the details for this package, including its version (2.21.0), description, and features.

NuGet Package Manager: OPCUAClient

Package source: nuget.org

Search: Include prerelease

Package Name	Author	Downloads	Version
Opc.UaFx.Advanced	Traeger.de	82.9K	2.21.0
Opc.UaFx.Client	Traeger.de	52.2K	2.21.0
OPCFoundation.NetStandard.Opc.Ua	OPC Foundation	8	1.4.367.95
OpcLabs.QuickOpc	OPC Labs	147K	5.62.1032
OPCFoundation.NetStandard.Opc.Ua.Core	OPC Founda	1.4.367.95	1.4.367.95
opc.ua.pubsub.dotnet.binary	Siemens AG	10.7K	1.0.16
opc.ua.pubsub.dotnet.client	Siemens AG	10.1K	1.0.16
OPCFoundation.NetStandard.Opc.Ua.Client	OPC Founc	1.4.367.95	1.4.367.95
OPCFoundation.NetStandard.Opc.Ua.Configuration	OPC UA Configuration	1.4.367.95	1.4.367.95
OPCFoundation.NetStandard.Opc.Ua.Server	OPC Four	1.4.367.95	1.4.367.95
OPCFoundation.NetStandard.Opc.Ua.Security.Certific	OPC UA Security X509	1.4.367.95	1.4.367.95

Opc.UaFx.Client nuget.org

Version:

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/opcuonet-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:

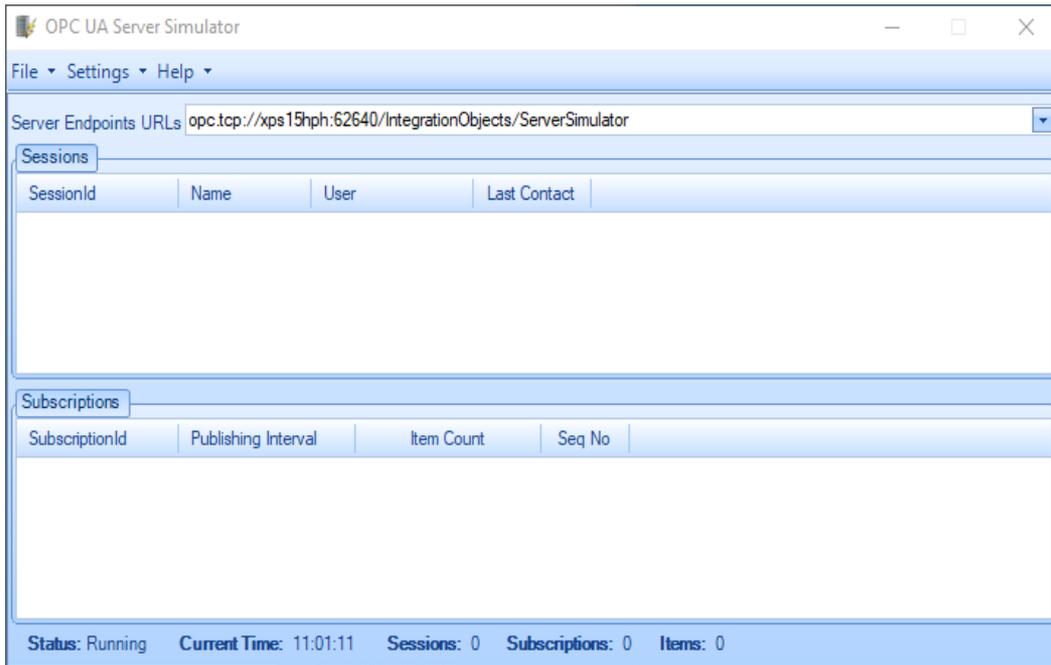
Solution Explorer

Solution 'OPCUAClient' (1 of 1 project)

- OPCUAClient
 - Dependencies
 - Analyzers
 - Frameworks
 - Form1.cs
 - Form1.Designer.cs
 - Form1.resx
 - Program.cs

Properties

OPC UA Server Simulator



- This free OPC UA Server tool supports data access and historical access information models of OPC UA.
- It provides simulated real-time and historical data.
- It is possible to configure your 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>

<https://www.halvorsen.blog>

Write Data to OPC UA Server

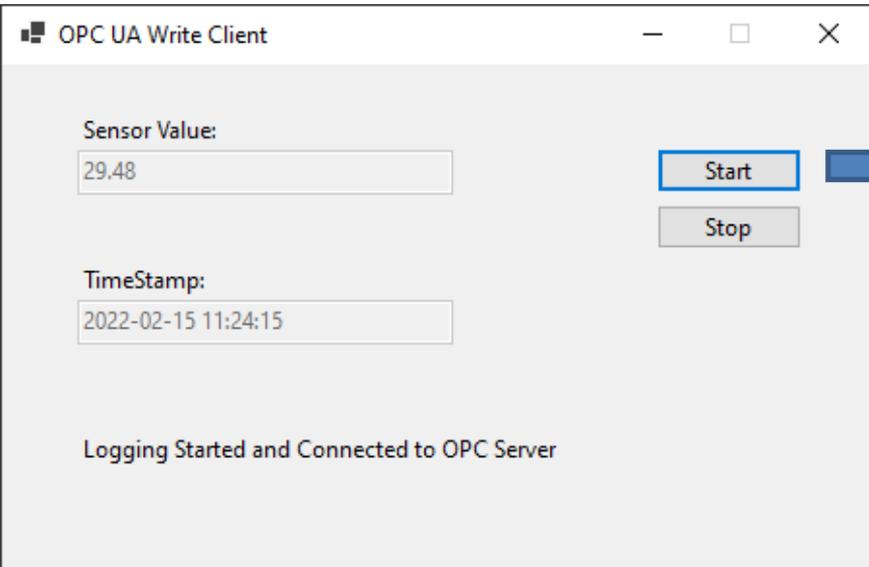


Hans-Petter Halvorsen

[Table of Contents](#)

OPC UA .NET SDK Example

OPC UA Write C# App



This is just a very simple Example.
You need to implement “OPC Write”
within your “Control Application”

OPC UA Server Simulator

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

SessionId	Name	User	Last Contact
OPCUAWrite	Anonymous	ns=3;i=1358359080	10:53:03
OPCUARRead	Anonymous	ns=3;i=1358359136	10:53:02

Subscriptions

Integration Objects' OPC UA Client

“OPC UA Client” Tool

Message Type: TimeStamp Message

[Control] 2022-02-08 13:03:09 Read operation of the variable [ns=2;ns=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.

OPC UA .NET SDK 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();
}
```

<https://www.halvorsen.blog>

Datalogging System



Hans-Petter Halvorsen

[Table of Contents](#)

<https://www.halvorsen.blog>

Read Data from OPC UA Server



Hans-Petter Halvorsen

[Table of Contents](#)

OPC UA .NET SDK Example

OPC UA Server Simulator

File Settings Help

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

Sessions

SessionId	Name	User	Last Contact
OPCUAWrite	Anonymous	ns=3;i=1358359080	10:53:03
OPCUARead	Anonymous	ns=3;i=1358359136	10:53:02

Subscriptions

SubscriptionId	Publishing Interval	Item Count	Seq No
----------------	---------------------	------------	--------

Status: Running Current Time: 10:53:03 Sessions: 2 Subscriptions: 0 Items: 0

Integration Objects OPC UA Client

Home New Open Save Save as Connect Disconnect Settings UA Settings Help About Define Remove Certificate Manage

OPC UA Client Tool

Sessions

Data View History View Event View

Display Name	Node Id	Value	Data Type	Server Timestamp	Source Timestamp	Status Code	Subscription	Session
--------------	---------	-------	-----------	------------------	------------------	-------------	--------------	---------

Address Space

Forward

Real Time Data

- Tag1
- Tag2
- Tag3
- Tag4
- Tag5
- Tag6
- Tag7
- Tag8
- Tag9
- Tag10

References and Attributes

- Tag10 Read
- Tag10 Write

Attribute Value

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

OPC UA Read Client

OPC Value:

29.48

Start

Stop

Connected to OPC Server

OPC UA .NET SDK 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();
}
```

<https://www.halvorsen.blog>

Save Data to SQL Server

Hans-Petter Halvorsen



[Table of Contents](#)

Stored Procedure Example

```
IF EXISTS (SELECT name
FROM sysobjects
WHERE name = 'StudentGrade'
AND type = 'P')
DROP PROCEDURE StudentGrade
GO
```

```
CREATE PROCEDURE StudentGrade
```

```
@Student varchar(50),
@Course varchar(10),
@Grade varchar(1)
```

```
AS
```

```
DECLARE
@StudentId int,
@CourseId int
```

```
select @StudentId=StudentId from STUDENT where StudentName =
@Student
```

```
select @CourseId=CourseId from COURSE where CourseName = @Course
```

```
insert into GRADE (StudentId, CourseId, Grade)
values (@StudentId, @CourseId, @Grade)
GO
```

A Stored Procedure is like a Method in C# - it is a piece of code with SQL commands that do a specific task – and you reuse it

This part is not necessary – but if you make any changes, you need to delete the old version before you can update it

Procedure Name

Input Arguments

Internal/Local Variables

Note! Each variable starts with @

SQL Code (the “body” of the Stored Procedure)

Using the Stored Procedure:

```
execute StudentGrade 'John Wayne', 'SCE2006', 'B'
```

Saving Data to SQL from C#

```
public void CreateBook(string connectionString, Book book)
{
    try
    {
        using (SqlConnection con = new SqlConnection(connectionString))
        {
            SqlCommand cmd = new SqlCommand("CreateBook", con);
            cmd.CommandType = CommandType.StoredProcedure;

            cmd.Parameters.Add(new SqlParameter("@Title", book.Title));
            cmd.Parameters.Add(new SqlParameter("@Isbn", book.Isbn));
            cmd.Parameters.Add(new SqlParameter("@PublisherName", book.PublisherName));
            cmd.Parameters.Add(new SqlParameter("@AuthorName", book.AuthorName));
            cmd.Parameters.Add(new SqlParameter("@CategoryName", book.CategoryName));

            con.Open();
            cmd.ExecuteNonQuery();
            con.Close();
        }
    }
    catch (Exception ex)
    {
        throw ex;
    }
}
```

It is recommended to create and use a **Stored Procedure**.
It is also recommended that the Connection String is stored
in **App.config**

<https://www.halvorsen.blog>

Alarm System

Alarm Generation and Alarm Monitoring



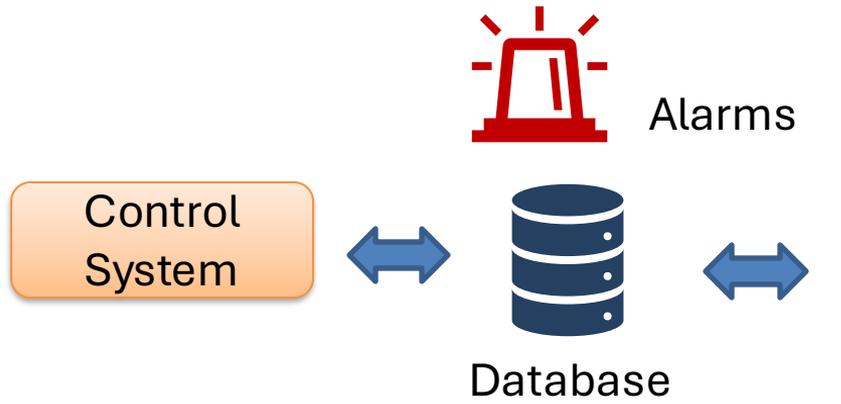
Hans-Petter Halvorsen

[Table of Contents](#)

Alarm System

- The Alarm System typically checks for Alarms and saves the Alarm information in your Database.
- The operator should be able to see the Alarms and make some actions to these alarms, i.e., the operator should have the possibility to Acknowledge Alarms, Show Alarms with different Priorities, etc.
- A **Database Trigger** can be used in order to generate Alarms in the Database
- A Database **View** can be created and used to retrieve Alarm Data from the Database

Alarm System Example



Trigger that checks new Process values against Alarm Levels and generates Alarms

Alarms that need to be Acknowledged by the Operator
Simple Example:

Alarm Application					
Alarm List:					Operator: Nils-Olav
AlarmId	TagName	AlarmType	Priority	ActivationTime	AckTime
5	Level	High	High	12:45	Ack
6	Temp	Low	Low	12:10	Ack
9	Pressure	High	Low	12:20	12:22
12	Level	Low	High	12:30	12:31
14	Pressure	High	Low	12:35	12:36
4	Level	HighHigh	High	12:40	12:42

Alarms that have been Acknowledged by the Operator

Trigger

A Trigger is executed when you insert, update or delete data in a Table specified in the

Trigger.

```
IF EXISTS (SELECT name
           FROM sysobjects
           WHERE name = 'CalcAvgGrade'
           AND type = 'TR')
    DROP TRIGGER CalcAvgGrade
GO

CREATE TRIGGER CalcAvgGrade ON GRADE
FOR UPDATE, INSERT, DELETE
AS
    DECLARE
    @StudentId int,
    @AvgGrade float

    select @StudentId = StudentId from INSERTED
    select @AvgGrade = AVG(Grade) from GRADE where StudentId = @StudentId
    update STUDENT set TotalGrade = @AvgGrade where StudentId = @StudentId
GO
```

This part is not necessary – but if you make any changes, you need to delete the old version before you can update it

Name of the Trigger

Specify which Table the Trigger shall work on

Specify what kind of operations the Trigger shall act on

Internal/Local Variables

Note! “INSERTED” is a temporarily table containing the latest inserted data, and it is very handy to use inside a trigger

Inside the Trigger you can use ordinary SQL statements, create variables, etc.

SQL Code (The “body” of the Trigger)

<https://www.halvorsen.blog>

ASP.NET Core Web Applications



Hans-Petter Halvorsen

[Table of Contents](#)

ASP.NET Core Web Applications

- ASP.NET is a Web Framework for creating Web Applications
- ASP.NET is integrated with Visual Studio and you will use the C# Programming Language
- .NET Core is cross-platform, meaning it will work on Windows, Linux and macOS.
- ASP.NET Core is Microsoft's newest baby, and it is the future of Web Programming

ASP.NET Core Web

Create a new project

Recent project templates

ASP.NET Core Web Application C#

ASP.NET Web Application (.NET Framework) C#

ASP.NET Web Application (Visual Basic (.NET Framework))

Windows Forms App (.NET Core) C#

Python Application Python

Windows Forms App (.NET Framework) C#

Search for templates (Alt+)

Application

C# Windows Web



ASP.NET Core Web Application

Project templates for creating ASP.NET Core web apps and web APIs for Windows, Linux and macOS using .NET Core or .NET Framework. Create web apps with Razor Pages, MVC, or Single Page Apps (SPA) using Angular, React, or React + Redux.

C# Linux macOS Windows Cloud Service Web



Blazor App

Project templates for creating Blazor apps that that run on the server in an ASP.NET Core app or in the browser on WebAssembly. These templates can be used to build web apps with rich dynamic user interfaces (UIs).

C# Linux macOS Windows Cloud Web



gRPC Service

A project template for creating a gRPC ASP.NET Core service using .NET Core.

C# Linux macOS Windows Cloud Service Web



Razor Class Library

A project template for creating a Razor class library.

Back

Next

Select the ASP.NET Core Web Application Project

ASP.NET Core Examples

Recommended Videos:



- ASP.NET Core – Introduction:
<https://youtu.be/zkOtiBcwo8s>
- **ASP.NET Core - Database Communication:**
<https://youtu.be/0Ta3dQ3rxzs>
- ASP.NET Core - Database CRUD Application:
<https://youtu.be/k5TCZDwTYcE>

Download Examples here: <https://www.halvorsen.blog/documents/programming/web/aspn>

ASP.NET Core Resources

Web Programming ASP.NET Core

Hans-Petter Halvorsen



<https://www.halvorsen.blog>

- Textbook
- Videos
- Tutorials
- Example Code

<https://www.halvorsen.blog/documents/programming/web/aspnet>

<https://www.halvorsen.blog>

Cyber Security



Hans-Petter Halvorsen

[Table of Contents](#)

Cyber Security in IACS Systems

- CSMS – Cyber Security Management System
- IACS – Industrial Automation and Control Systems
- Security is critical in IACS systems because a potential hacker can do great damage
- In the Norwegian energy and oil and gas sector alone, more than 50 cyber security incidents are detected the last year.*

Cyber Attacks

- In computers and computer networks an **attack** is any attempt to expose, alter, disable, destroy, steal or gain unauthorized access to or make unauthorized use of the system
- A **cyber attack** is any type of action that targets computer information systems, infrastructures, computer networks, or personal computer devices.
- An **attacker** is a person or process that attempts to access data, functions or other restricted areas of the system without authorization, potentially with malicious intent

Cyber Security Standards

- To protect the cyber environment of a user or organization.
- This environment includes users themselves, networks, devices, all software, processes, information in storage or transit, applications, services, and systems that can be connected directly or indirectly to networks
- Reduce the risks and prevent Cyber Attacks
- IEC62443 – Cyber Security standard for IACS systems

Cyber Security in IACS Systems

Things to consider:

- Data & Cyber Security Issues regarding your SCADA Software
- How can you secure your Software against threats and vulnerabilities?
- What kind of precautions have you done when implementing your system?
- What can/should you/have you done do to protect your Software?

Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

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

Web: <https://www.halvorsen.blog>

