https://www.halvorsen.blog

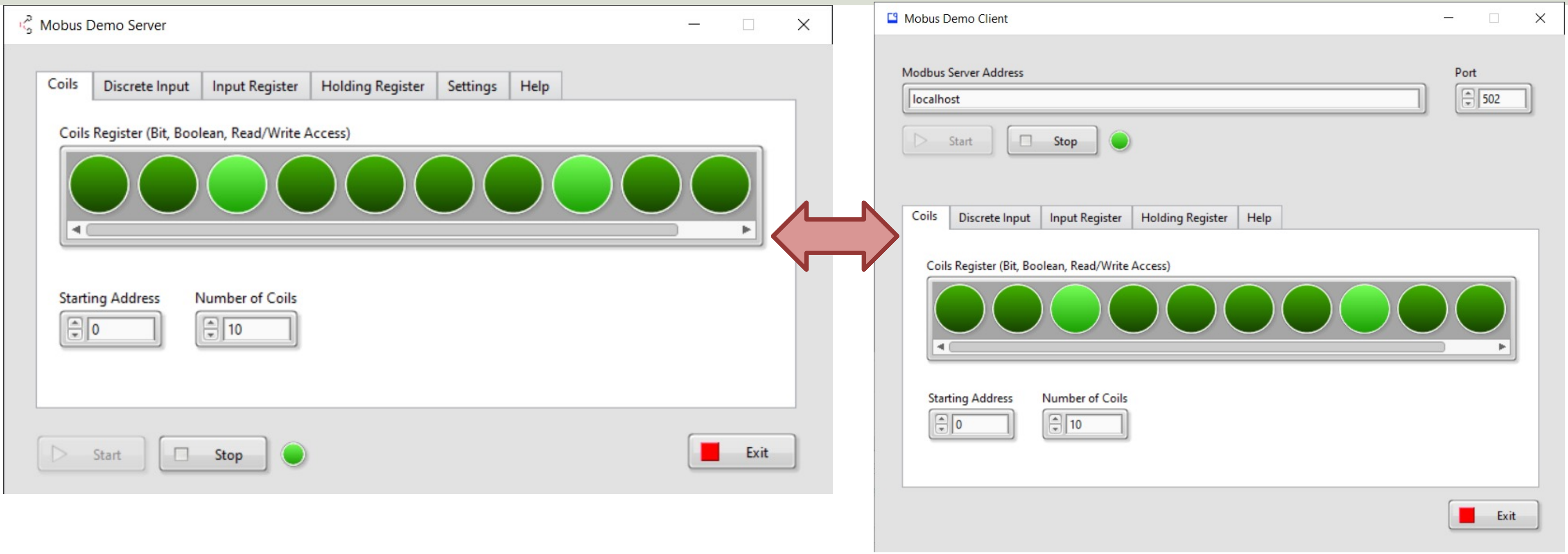
Modbus Demo Software

Hans-Petter Halvorsen

Contents

- Modbus
- Modbus Demo Software
  - Modbus Demo Server
  - Modbus Demo Client
  - Developed using LabVIEW
- Modbus Examples and Demonstrations
  - Communicating with the Modbus Demo Software using MATLAB
- LabVIEW Modbus Demo Project (Source Code)
  - You need LabVIEW

Modbus Demo Software



Mobus Demo Server

Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

Mobus Demo Client

Modbus Server Address: localhost

Port: 502

Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

The Modbus Demo Software consists of executable Applications for a Modbus Demo Server and a Modbus Demo Client, which you can use to test your own Modbus system or your own Modbus applications.

Download the Software from https://www.halvorsen.blog/

https://www.halvorsen.blog/documents/technology/modbus/

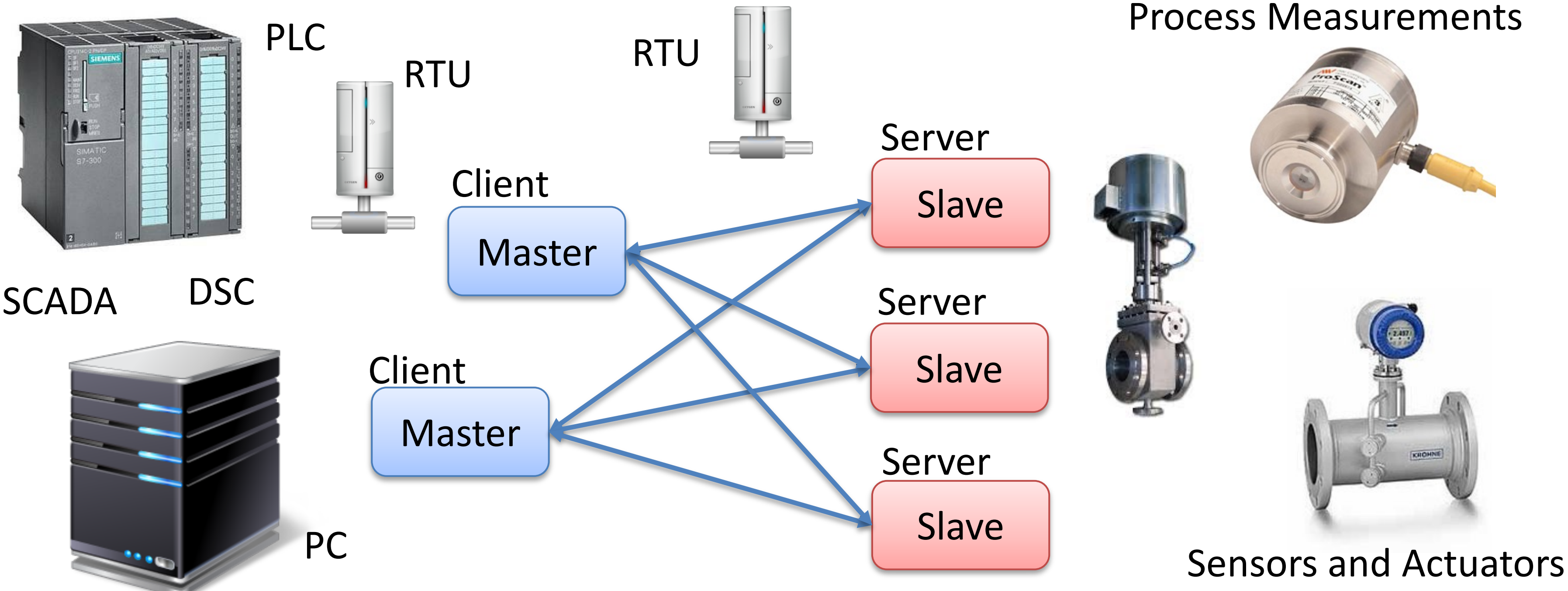
Modbus

What is Modbus?

- **Modbus is a serial communications protocol** originally published by Modicon (now Schneider Electric) in 1979 for use with its programmable logic controllers (PLCs).
- **Simple and robust**, it has since become a de facto standard communication protocol, and it is now a commonly available means of connecting industrial electronic devices
- The development and update of Modbus protocols has been managed by the Modbus Organization since April 2004, when Schneider Electric transferred rights to that organization (https://modbus.org)
- Modbus became the first widely accepted fieldbus standard.

Modbus

The master typically is a PLC (Programmable Logic Controller), PC or DCS (Distributed Control System)



PLC

RTU

RTU

Process Measurements

Client

Master

Client

Master

Server

Slave

Server

Slave

Server

Slave

SCADA

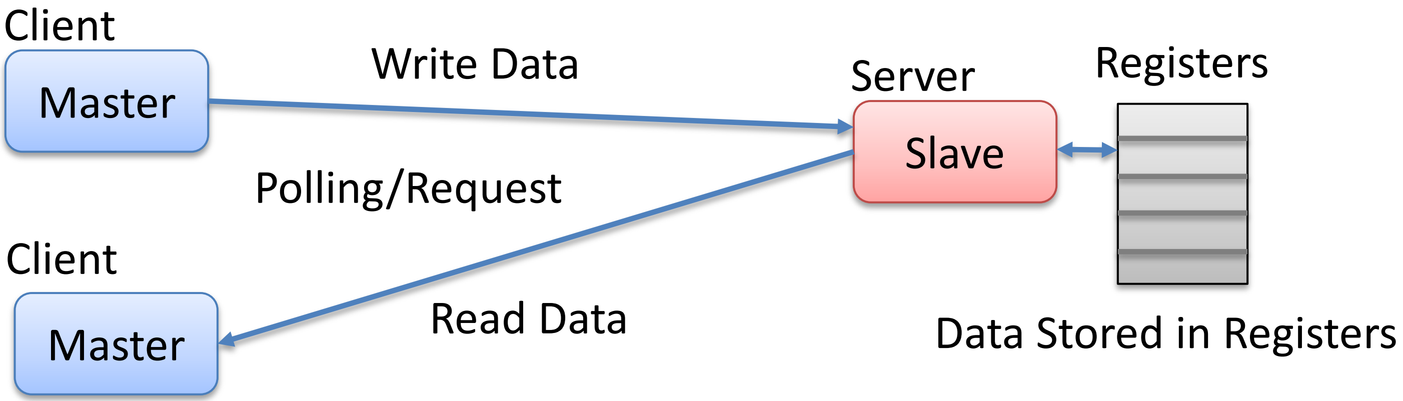
DSC

PC

Sensors and Actuators

A remote terminal unit (RTU) is a microprocessor-controlled electronic device that interfaces objects in the physical world to a DCS or SCADA System

Master/Slave or Client/Server



Note! The terms “Master” and “Slave” used in Modbus has been replaced with the terms “Client” and “Server”. It takes time to establish these new terms, so in different literature and different software you may still see both the old and the new terms.

Modbus Register Types

- **Coil** (Discrete Output)
  - Coils are 1-bit registers, used to control discrete outputs, Read or Write
- **Discrete Input** (Read Only)
  - 1-bit registers
- **Input Register** (Read Only)
- **Holding Register** (Read/Write)

Access Levels

Register Type	Data Type	Client Access	Server Access
Coils	Bit (Boolean)	Read/Write	Read/Write
Discrete Input	Bit (Boolean)	Read-only	Read/Write
Input Register	Unsigned Word	Read-only	Read/Write
Holding Register	Unsigned Word	Read/Write	Read/Write

An **Unsigned Word** is a 16-bit nonnegative Integer Value between 0 – 65535 (2^16)

Modbus Demo Software

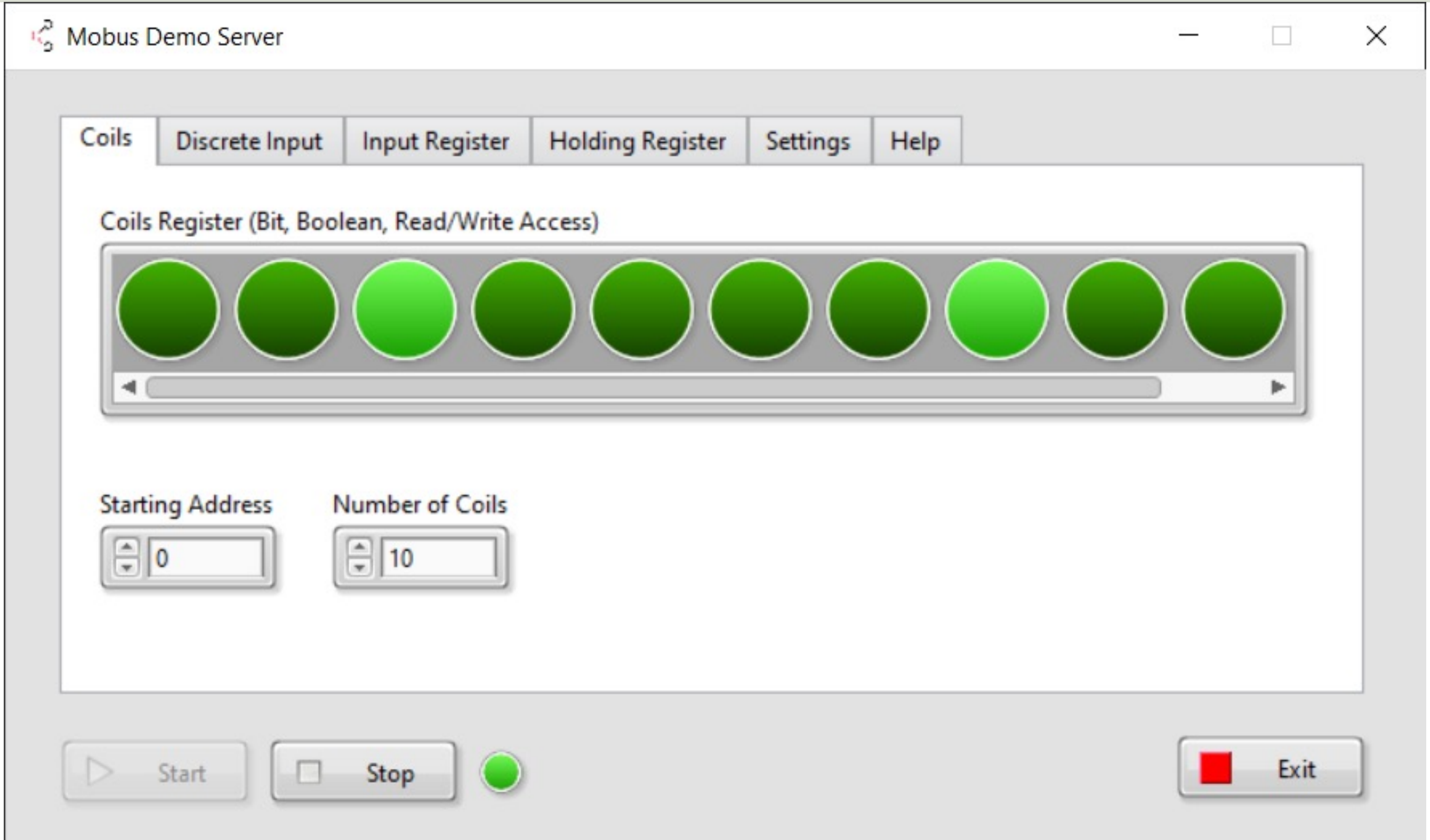
Hans-Petter Halvorsen

Table of Contents

Modbus Demo Software

- The Modbus Demo Software consists of
  - A Modbus Demo Server
  - A Modbus Demo Client
- The Modbus Demo Software is created with LabVIEW
- It supports all 4 Modbus Registers
  - Coils, Discrete Input, Input Register and Holding Register
- In order to use it you need to
  - Download the Software from https://www.halvorsen.blog/
  - https://www.halvorsen.blog/documents/technology/modbus/

Modbus Demo Server



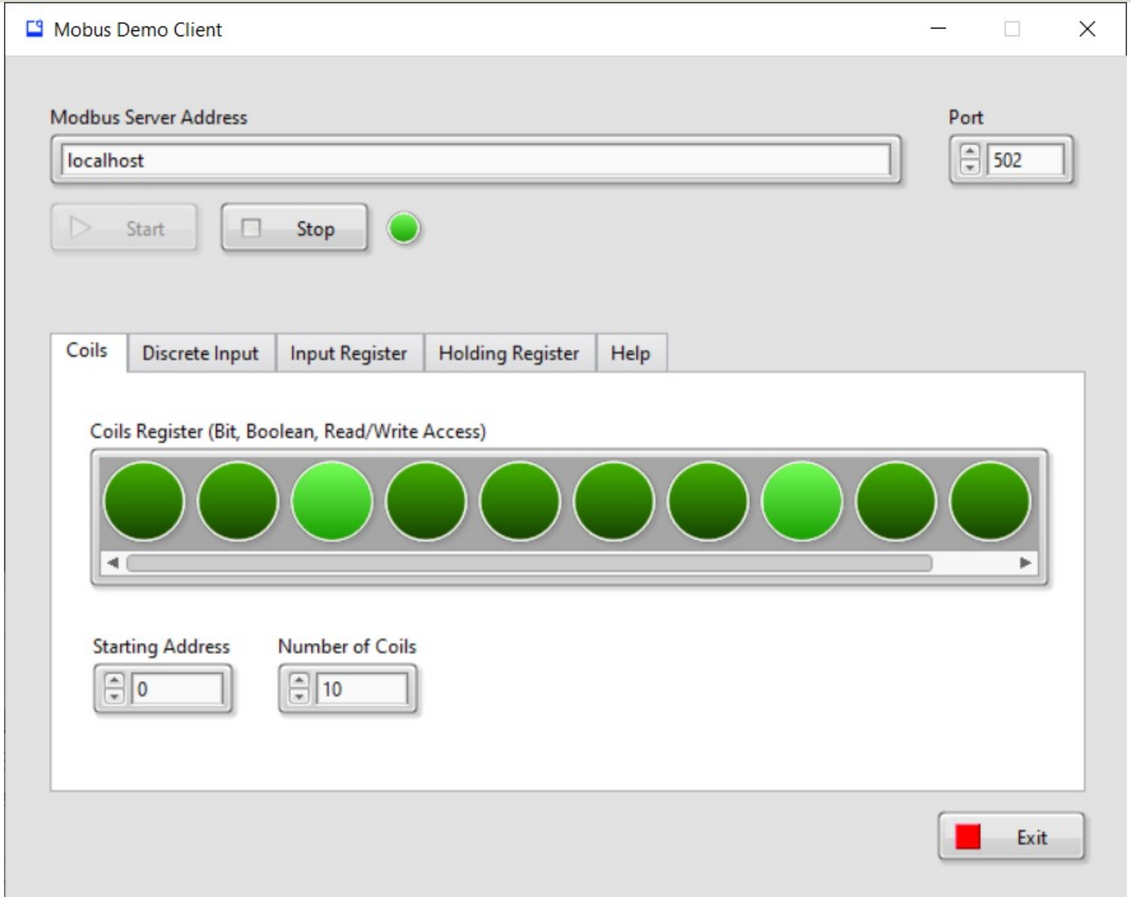
Mobus Demo Server

Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

Modbus Demo Client



Mobus Demo Client

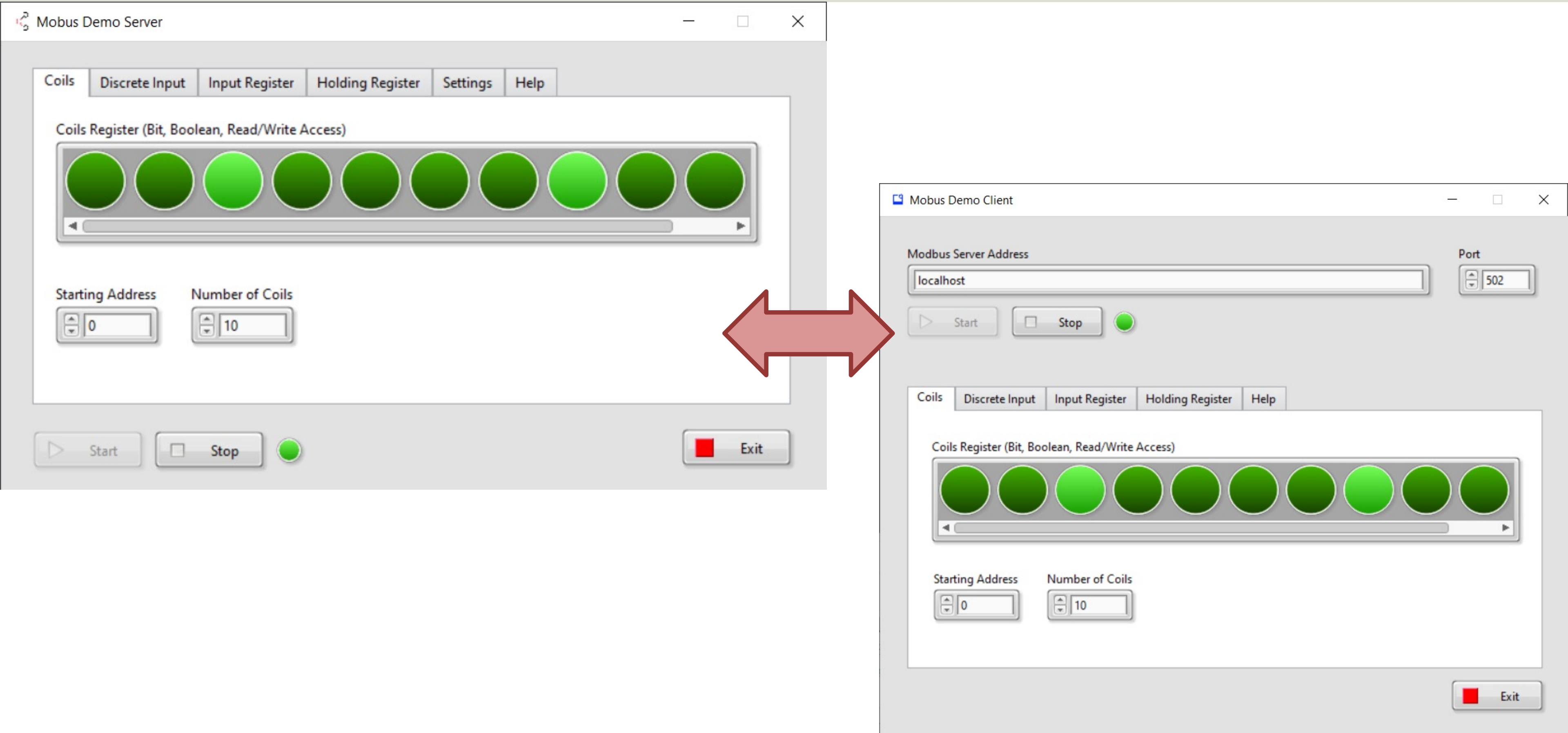
Modbus Server Address: localhost

Port: 502

Starting Address: 0

Number of Coils: 10

Modbus Demo Software



Mobus Demo Server

Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

Mobus Demo Client

Modbus Server Address: localhost

Port: 502

Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

Modbus Examples

Modbus Examples

- Let's show some Modbus examples where we communicate with the Modbus Demo Software
- In theory, you can use any kind of Programming Language for Modbus communication
- You can create it from scratch with in-depth knowledge of the Modbus protocol
- Or you can use an existing Library or Toolkit
- Here, some examples using the Industrial Communication Toolbox in MATLAB will be presented.

Modbus with MATLAB

- In order to use Modbus with MATLAB you can use the "Industrial Communication Toolbox".
- The "Industrial Communication Toolbox" supports the following Protocols:
  - OPC, both OPC DA and OPC UA (previously OPC Toolbox)
  - MQTT
  - Modbus (which is the focus in this Tutorial)
- **Note!** "Industrial Communication Toolbox" is a new Toolbox that is included in "MATLAB R2022a" and newer versions

https://mathworks.com/help/icomm/modbus.html

Industrial Communication Toolbox

- You can use it to communicate with Modbus Servers, such as a PLC, etc.
- It supports Modbus interface over TCP/IP or Serial RTU

https://mathworks.com/products/industrial-communication.html

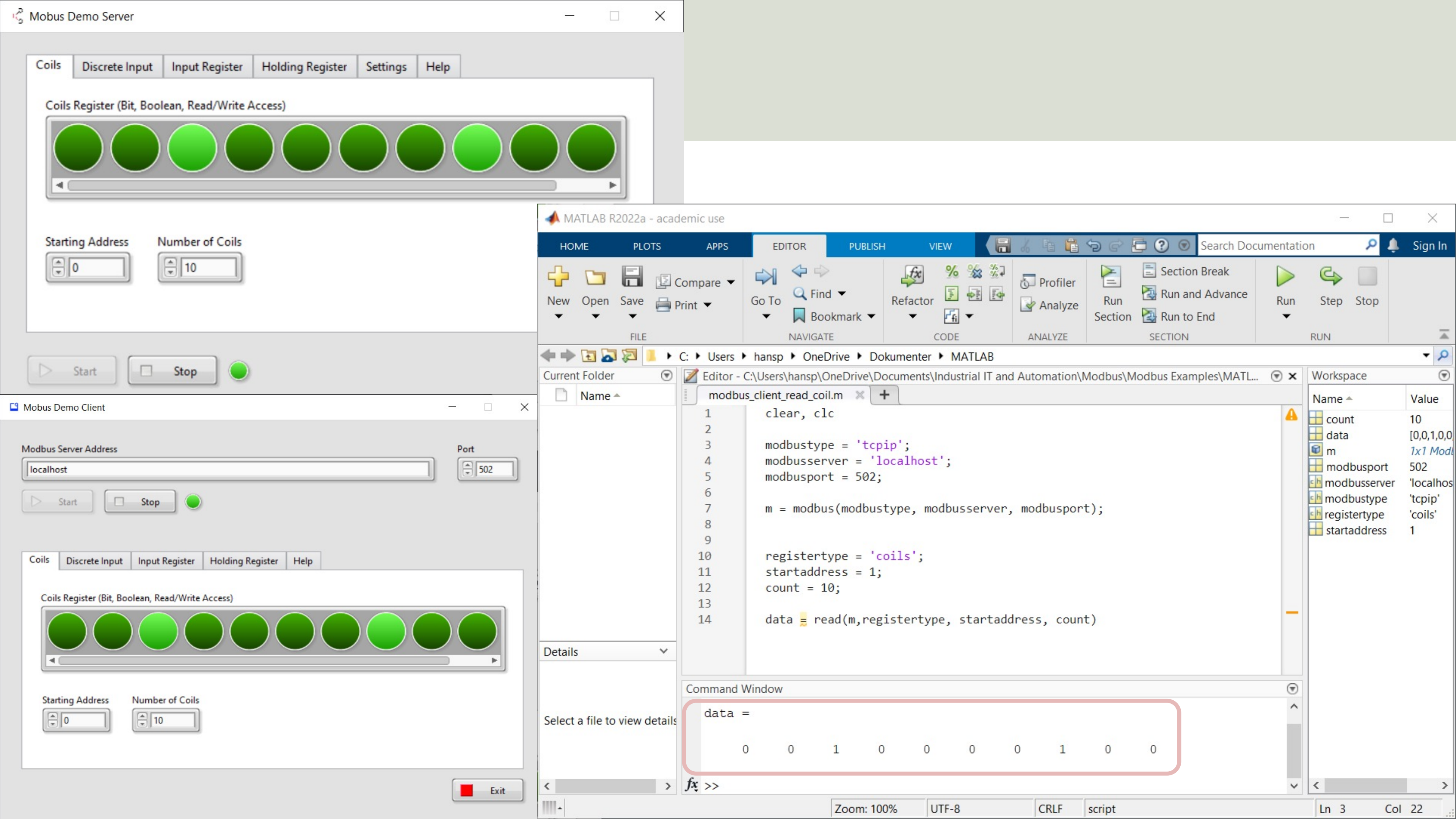
Basic Read Example

```
modbustype = 'tcpip';
modbusserver = 'localhost';
modbusport = 502;

m = modbus(modbustype, modbusserver, modbusport);

registertype = 'coils';
startaddress = 1;
count = 10;

data = read(m,registertype, startaddress, count)
```



Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

Modbus Server Address: localhost

Port: 502

Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

```
clear, clc

modbustype = 'tcpip';
modbusserver = 'localhost';
modbusport = 502;

m = modbus(modbustype, modbusserver, modbusport);


registertype = 'coils';
startaddress = 1;
count = 10;

data = read(m,registertype, startaddress, count)
```

```
data =

     0     0     1     0     0     0     0     1     0     0
```

Name	Value
count	10
data	[0,0,1,0,0
m	1x1 Modl
modbusport	502
modbusserver	'localhos
modbustype	'tcpip'
registertype	'coils'
startaddress	1

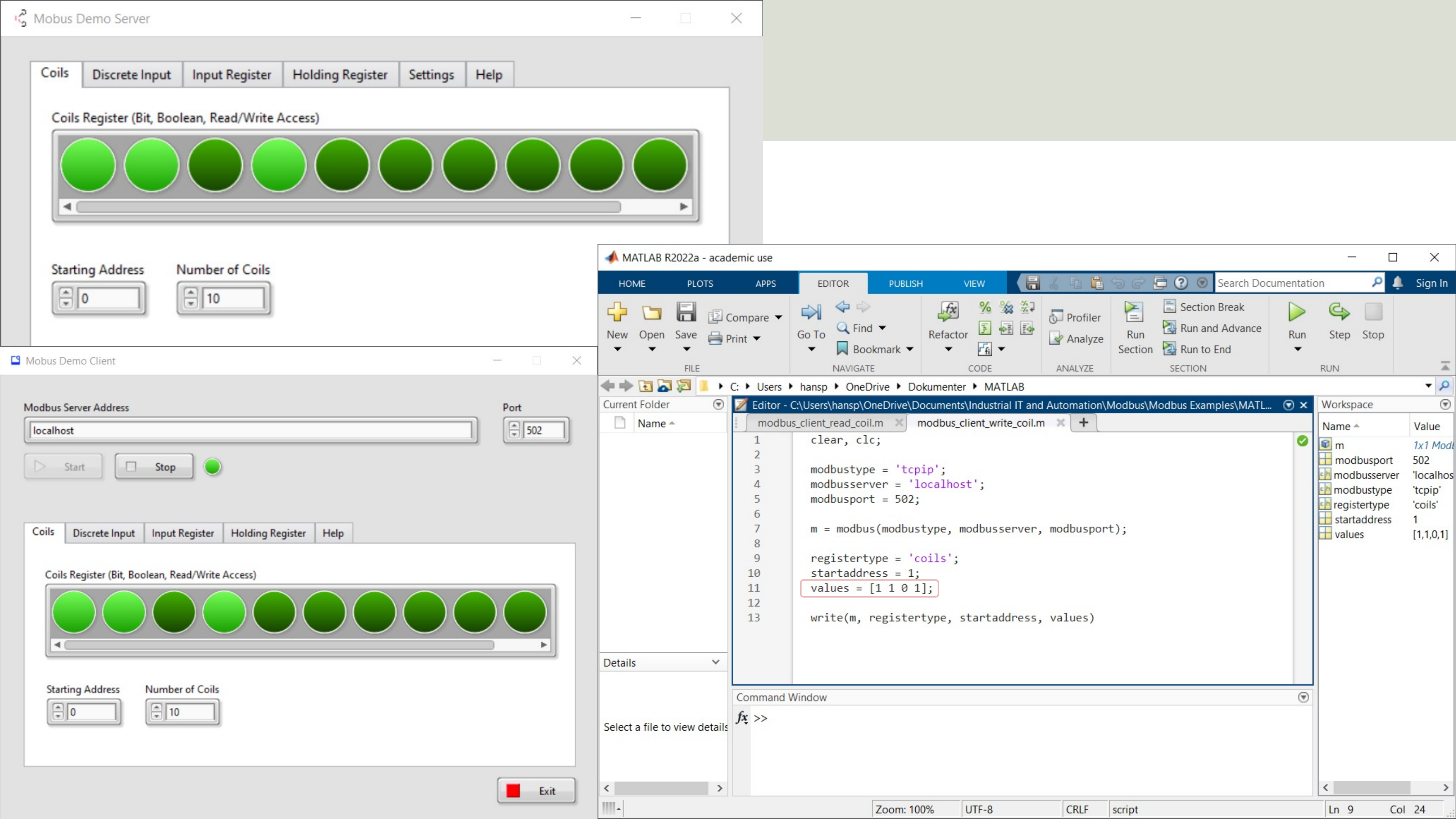
Basic Write Example

```
modbustype = 'tcpip';
modbusserver = 'localhost';
modbusport = 502;

m = modbus(modbustype, modbusserver, modbusport);

registertype = 'coils';
startaddress = 1;
values = [1 1 0 1];

write(m, registertype, startaddress, values)
```



Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

Modbus Server Address: localhost

Port: 502

Coils Register (Bit, Boolean, Read/Write Access)

Starting Address: 0

Number of Coils: 10

Editor - C:\Users\hansp\OneDrive\Documents\Industrial IT and Automation\Modbus\Modbus Examples\MATL...

modbus_client_read_coil.m | modbus_client_write_coil.m

```
clear, clc;

modbustype = 'tcpip';
modbusserver = 'localhost';
modbusport = 502;

m = modbus(modbustype, modbusserver, modbusport);

registertype = 'coils';
startaddress = 1;
values = [1 1 0 1];

write(m, registertype, startaddress, values)
```

Command Window

```
>>
```

Workspace

Name	Value
m	1x1 Modl
modbusport	502
modbusserver	'localhos
modbustype	'tcpip'
registertype	'coils'
startaddress	1
values	[1,1,0,1]

LabVIEW Modbus Demo Software Project

Hans-Petter Halvorsen

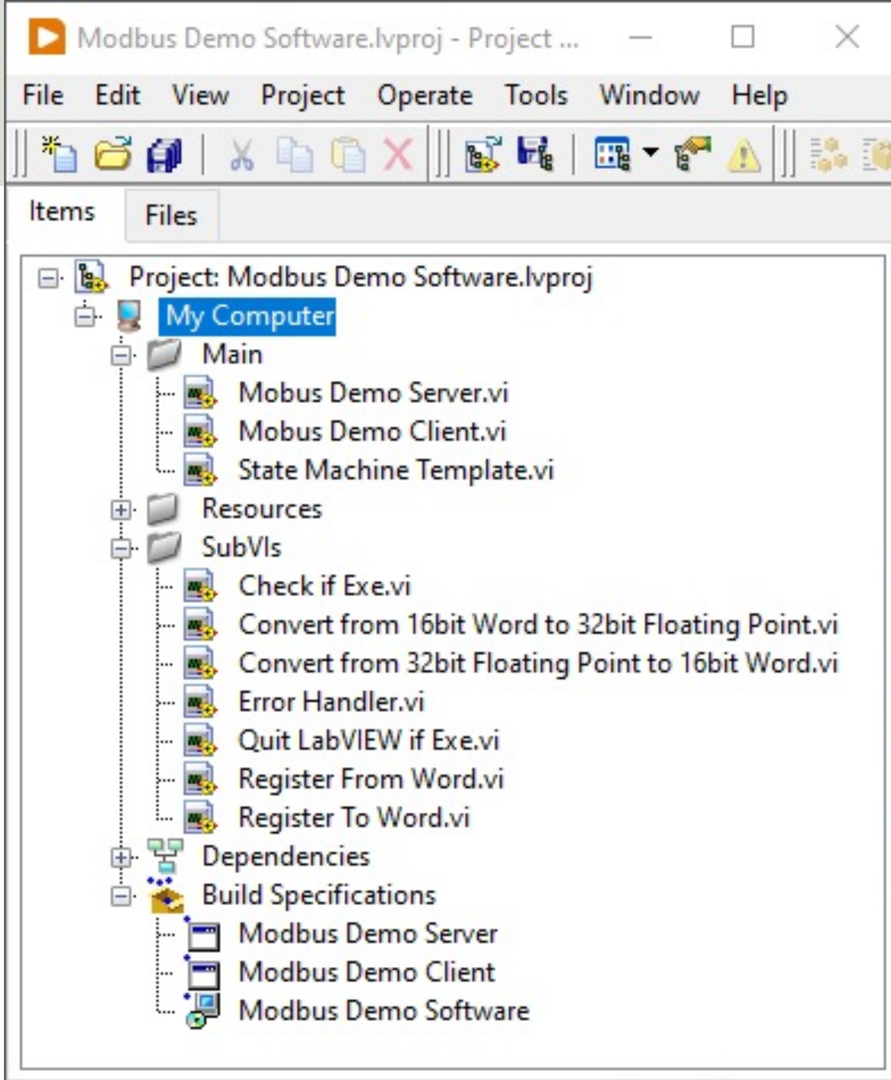
Table of Contents

LabVIEW Project

- A Modbus Demo Server and a Modbus Demo Client has been presented
- The software is available as Executable Applications ready to use for Demo, Testing, etc.
- Those are created using the LabVIEW Programming Environment
- You can freely download the entire LabVIEW Project and the Source Code
- https://www.halvorsen.blog/
- https://www.halvorsen.blog/documents/technology/modbus/

LabVIEW Project

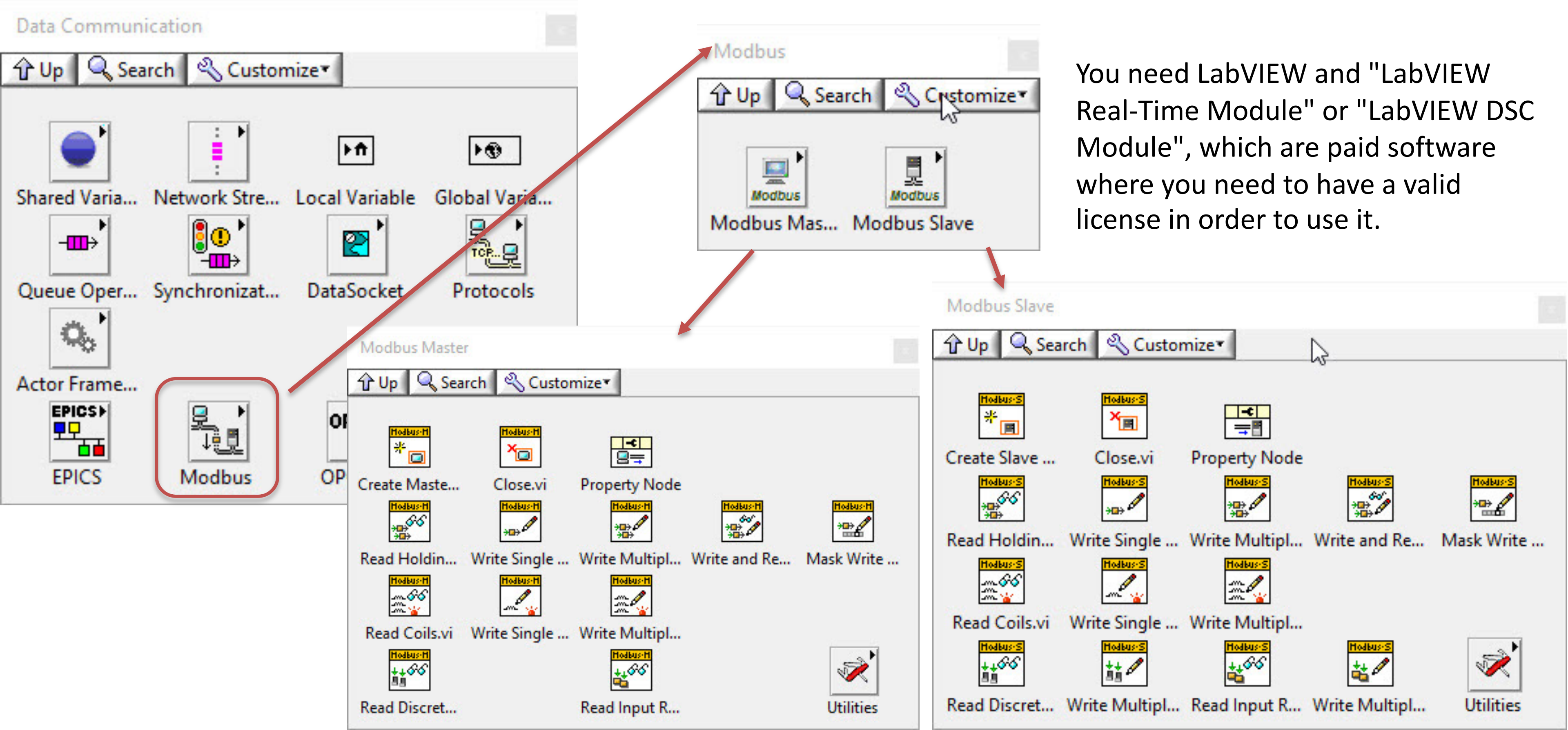
You need LabVIEW and "LabVIEW Real-Time Module" or "LabVIEW DSC Module", which are paid software where you need to have a valid license in order to use it.



Project: Modbus Demo Software.lvproj
- My Computer
  - Main
    - Mobus Demo Server.vi
    - Mobus Demo Client.vi
    - State Machine Template.vi
  - Resources
  - SubVIs
    - Check if Exe.vi
    - Convert from 16bit Word to 32bit Floating Point.vi
    - Convert from 32bit Floating Point to 16bit Word.vi
    - Error Handler.vi
    - Quit LabVIEW if Exe.vi
    - Register From Word.vi
    - Register To Word.vi
  - Dependencies
  - Build Specifications
    - Modbus Demo Server
    - Modbus Demo Client
    - Modbus Demo Software

Modbus in LabVIEW

You need LabVIEW and "LabVIEW Real-Time Module" or "LabVIEW DSC Module", which are paid software where you need to have a valid license in order to use it.



References

- Modbus Organization: http://www.modbus.org
- Modbus (Wikipedia): https://en.wikipedia.org/wiki/Modbus
- Introduction to Modbus (National Instruments): http://www.ni.com/white-paper/7675/en/
- Connect LabVIEW to Any PLC With Modbus (National Instruments): http://www.ni.com/tutorial/13911/en/
- Modbus 101 - Introduction to Modbus: http://www.csimn.com/CSI_pages/Modbus101.html
- Modbus TCP/IP: http://www.rtaautomation.com/technologies/modbus-tcpip/
- Modbus RTU: http://www.rtaautomation.com/technologies/modbus-rtu/
- Using Modbus for Process Control and Automation (PDF): http://www.miinet.com/Portals/0/articles/Using_MODBUS_for_Process_Control_and_Automation.pdf

Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

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

Web: https://www.halvorsen.blog