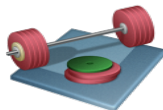
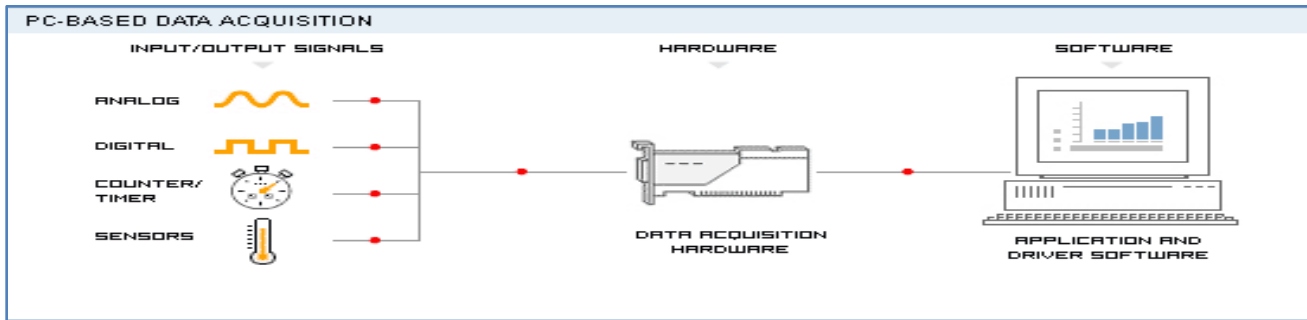




<http://home.hit.no/~hansha/?page=labview>

Introduction to DAQ with LabVIEW

TC-01
Thermocouple



with Self-paced Step-by-Step Exercises

Contents

- LabVIEW
- What is DAQ?
- Using TC-01 Thermocouple Device in LabVIEW
- Plotting
- Datalogging
- Measurement Filter



LabVIEW Installation

Note! You get the Serial Number from your Teacher, but the software can be used for 30 days before you need to enter a valid Serial Number.

Download the software here:

<http://home.hit.no/~hansha/?page=labview>

These are the main modules we use in the different courses at Telemark University College:

- **LabVIEW** (LabVIEW Professional Development System 32-Bit: English)
- **NI-DAQmx** (Hardware Driver for NI USB-6008, NI TC-01, etc.)
- **LabVIEW Control Design and Simulation Module**
- **LabVIEW MathScript RT Module**

Note! These packages are separate downloads!

All LabVIEW Software can be downloaded here: www.ni.com/download

Additional LabVIEW Resources

Here you will find lots of Videos, Tutorials and Exercises



- LabVIEW Training for Students (National Instruments):

<http://ni.com/students/learnlabview>

- LabVIEW Course:

<http://home.hit.no/~hansha/?training=labview>

Learning by Doing!

It is recommended that you watch some of the videos before you read further

High-Level Design Tools

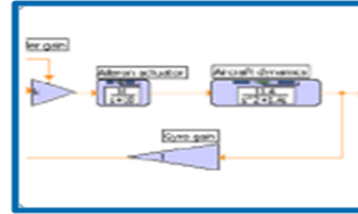
Configuration



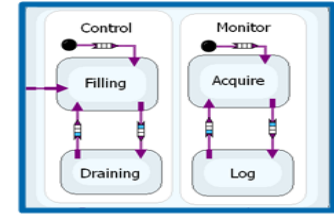
Textual Math

```
1 c = 0.285 + 0.013i;  
2 [X Y] = meshgrid(x, y);  
3 z = X + i*Y;  
4 for k=1:30  
5   z = z.^2 + c;  
6 end
```

Simulation



Statechart



LabVIEW

Graphical Programming

Linux®



Macintosh

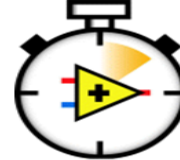


Windows

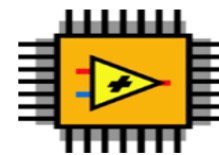


Desktop Platform

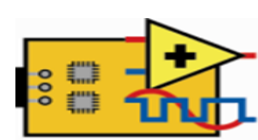
Real-Time



FPGA

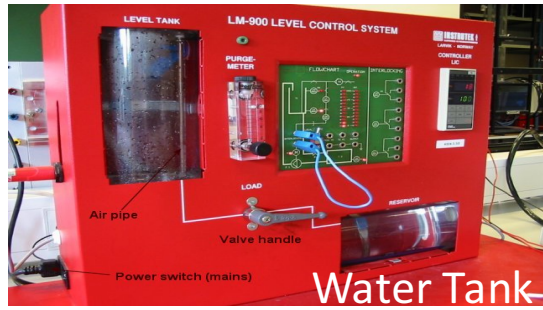


MPU



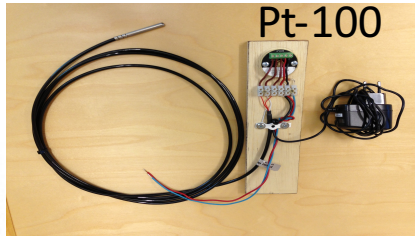
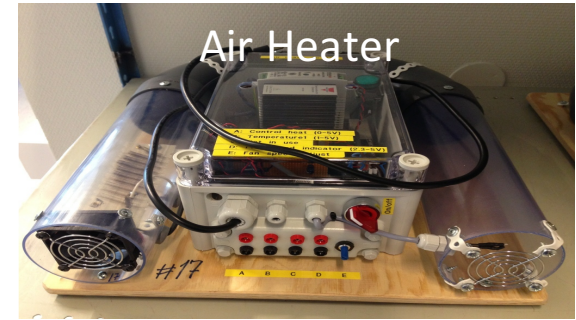
Embedded Platform

Hardware



USB-6008

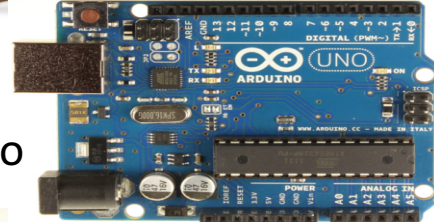
Wi-Fi DAQ



TC-01



Arduino



Vision System

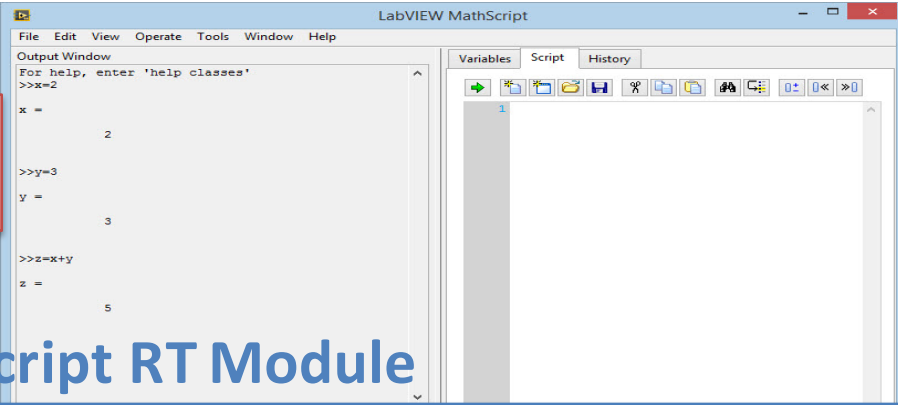
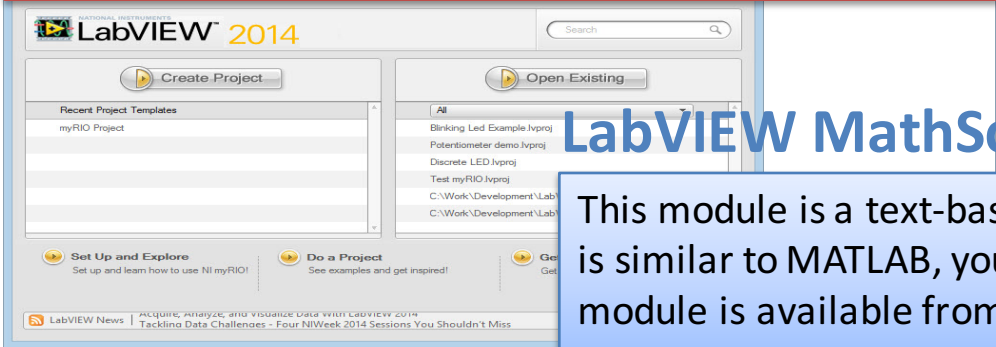


NOx Sensor



LabVIEW

This is the core LabVIEW installation that installs the LabVIEW Programming Environment.

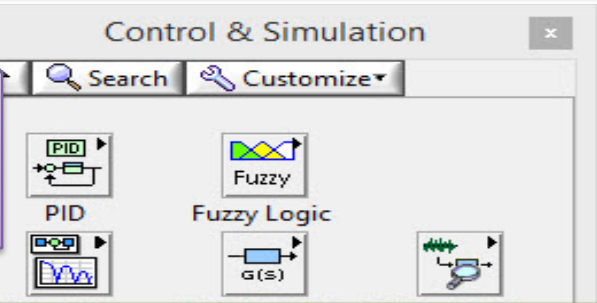


LabVIEW MathScript RT Module

This module is a text-based tool that is very similar to MATLAB. The syntax is similar to MATLAB, you can create and run so-called m files, etc. The module is available from the Tools menu inside LabVIEW.

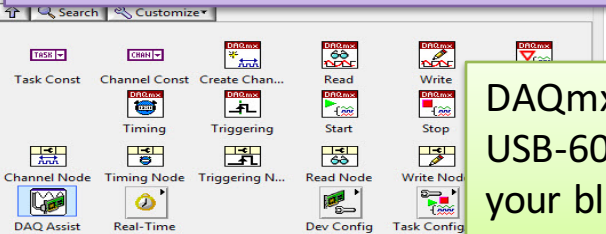
LabVIEW Control Design and Simulation Module

This module is used for creating Control and Simulation applications with LabVIEW. Here you will find PID controllers, etc. The module is available as a palette on your block diagram.



NI-DAQmx

DAQmx is the Hardware Driver needed in order to use hardware devices like NI USB-6008, NI TC-01, etc. inside LabVIEW. The module is available as a palette on your block diagram.





Keyboard Shortcuts

File		Operate		Right-Click	
Ctrl-N	Create new VI	Ctrl-Z	Undo last action	Right-Click	Display controls/ functions palette
Ctrl-S	Save VI	Ctrl-Shift-Z	Redo last action	Shift-Right-Click	Display tools palette
Ctrl-P	Print			Ctrl-T	Tile block diagram and front panel windows
Edit		Window		Help	
Ctrl-V	Paste object	Ctrl-R	Run VI	Ctrl-H	Display context help
Ctrl-U	Clean up diagram	Ctrl-.	Abort VI		
Ctrl-Space	Activate quick drop				
Ctrl-B	Remove broken wires				
Ctrl-C	Copy an object				
Ctrl-X	Cut object				

Editing Tools

Tool	Icon	Description
Show Context Help		Display the context help window
Text Settings		Change the font setting for the VI, including size, style, and color
15pt Application Font		
Align Objects		Align selected objects
Distribute Objects		Space objects evenly
Resize Objects		Resize multiple front panel objects to the same size
Reorder		Reorder the layers of the objects
Clean Up Diagram		Rearrange wires and objects on the block diagram
Enter		Appears when a new value is available to replace an old value

Debugging Tools

Tool	Icon	Description
Run		Execute the VI
List Errors		List errors that prevent the VI from running
Run Continuously		Execute the VI continuously until abort or pause is pressed
Abort Execution		Stop VI execution immediately
Execution Highlighting		Animate data movement on the block diagram wires
Pause		Temporarily stop execution to debug a portion of the VI
Step Into		Single-step into a subVI or structure to debug it
Step Over		Execute a subVI or structure and pause at the next one
Step Out		Execute a subVI or structure and resume single-stepping

Tools Palette

Tool	Icon	Description
Automatic Tool Selection		Automatically choose the appropriate tool
Operating Tool		Change the value of a control or select the text within a control
Positioning Tool		Position, resize, and select objects
Labeling Tool		Edit text and create free labels
Wiring Tool		Wire objects together on a block diagram
Scrolling Tool		Scroll the window without using the scroll bars
Breakpoint Tool (Used for debugging)		Set breakpoints on VIs, functions, wires, loops, sequences, and cases
Probe Tool (Used for debugging)		Create probes on wires and display intermediate values on a wire in a running VI
Get Color Tool		Copy colors for pasting with the Color Tool
Coloring Tool		Set the foreground and background colors



NATIONAL INSTRUMENTS

LabVIEW



DAQ

Hans-Petter Halvorsen, M.Sc.

DAQ Hardware Examples

NI TC-01 Thermocouple Temperature Measurements

We will use this
device



NI USB-6008 I/O Module

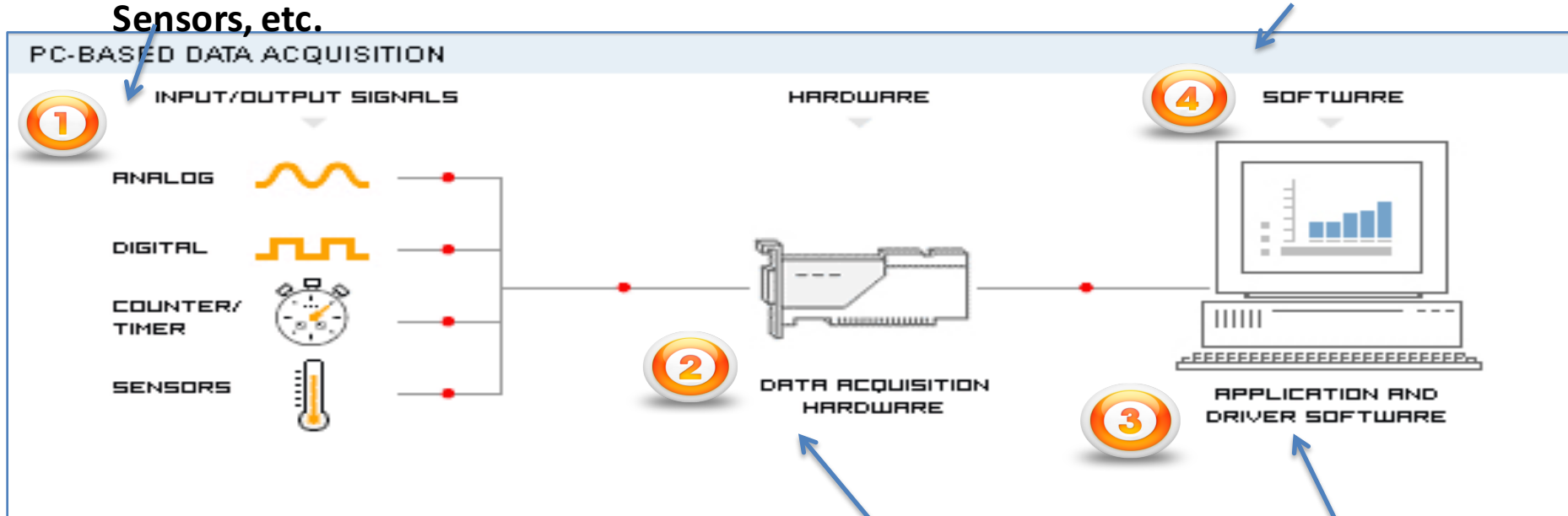
Analog/Digital Inputs/Outputs



Note! The **DAQmx** Driver is needed in order to use them inside LabVIEW!!

DAQ – Data Acquisition

Your App created with LabVIEW



A DAQ System consists of 4 parts:

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

NI TC-01 Thermocouple Device
or
NI DAQmx Driver
NI USB 6008 DAQ Device



Using TC-01 in LabVIEW

Hans-Petter Halvorsen, M.Sc.

TC-01

How-To use TC-01 with LabVIEW



Connect device to
PC using USB

USB



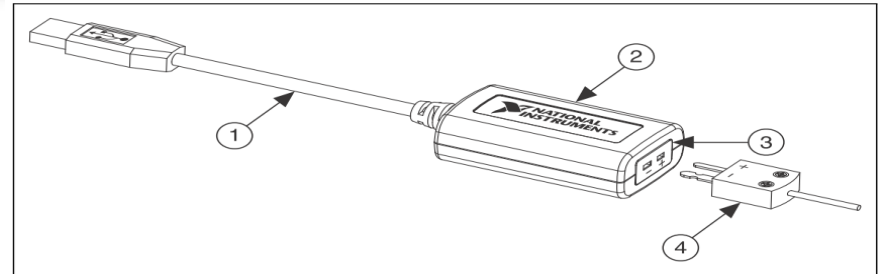
NI TC-01

Temperature (Thermocouple) Device

Used to log Temperature Data using the LabVIEW software



DAQmx Driver needed!

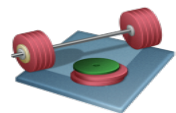


- | | | | |
|---|--------------------|---|----------------------------|
| 1 | USB Cable | 3 | Thermocouple Input |
| 2 | NI USB-TC01 Device | 4 | Minithermocouple Connector |

The NI USB-TC01 provides connections for one thermocouple. Thermocouple types J, K, R, S, T, N, E, and B are supported. At TUC we will use the J type.

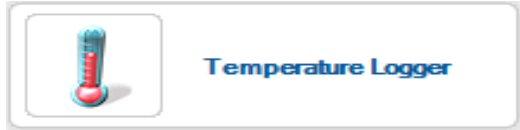
Getting Started with TC-01

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”):

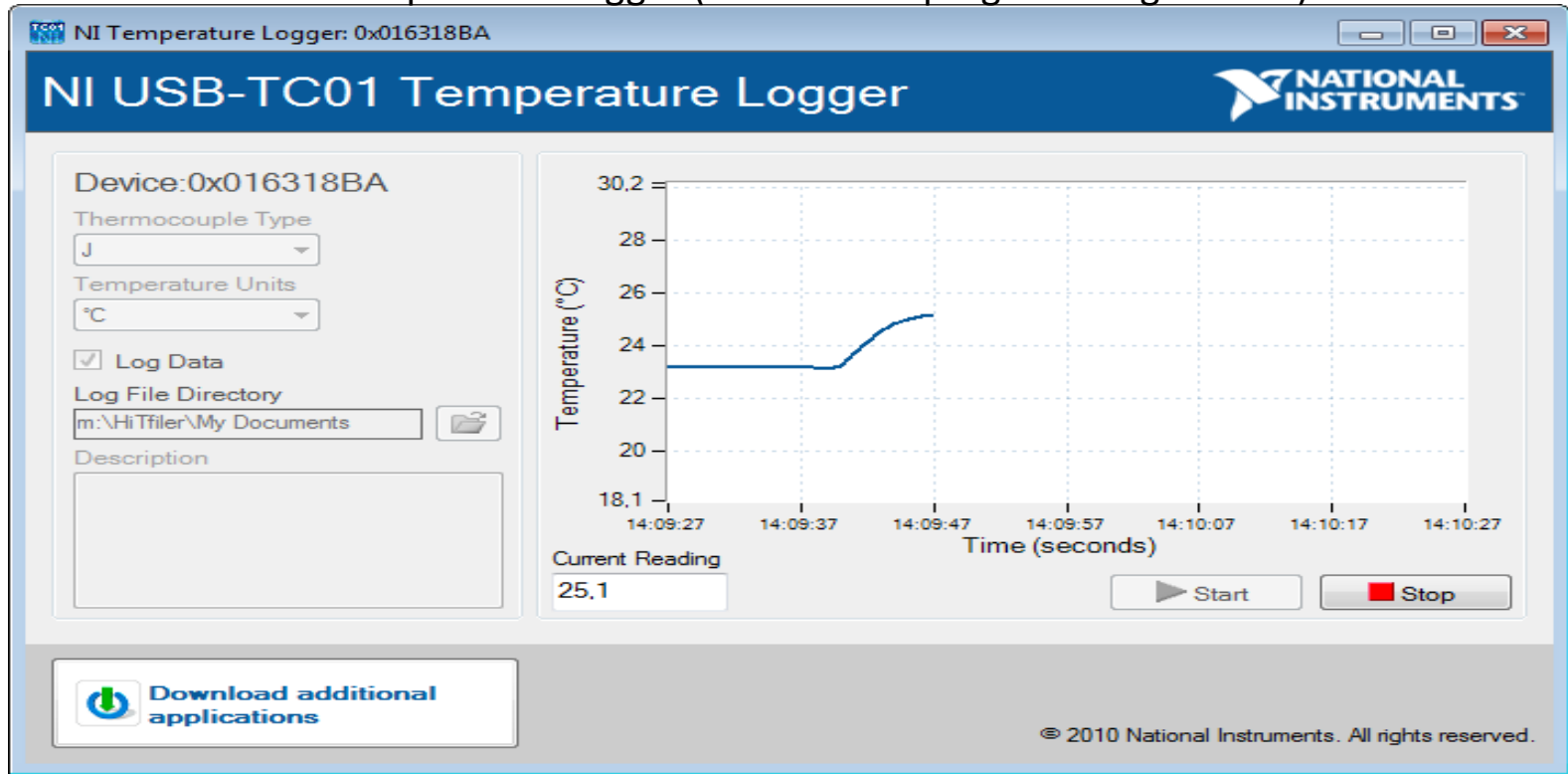


Students: Plug in the device into your USB port and see if this window appears

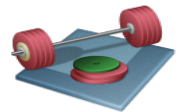
NI TC-01 Built-in Temperature Logger



Built-in Temperature Logger (No Driver or programming needed)



The screenshot shows the NI USB-TC01 Temperature Logger software interface. The window title is "NI Temperature Logger: 0x016318BA". The main title bar reads "NI USB-TC01 Temperature Logger" and features the National Instruments logo. On the left, there are configuration options: "Device: 0x016318BA", "Thermocouple Type" set to "J", "Temperature Units" set to "°C", a checked "Log Data" box, "Log File Directory" set to "m:\HiTfiler\My Documents", and a "Description" text area. The central graph plots "Temperature (°C)" on the y-axis (ranging from 18.1 to 30.2) against "Time (seconds)" on the x-axis (ranging from 14:09:27 to 14:10:27). The graph shows a blue line that is constant at approximately 23.2°C until 14:09:37, then rises to about 25.1°C by 14:09:47. Below the graph, the "Current Reading" is displayed as 25.1. At the bottom right of the graph area are "Start" and "Stop" buttons. At the bottom left, there is a "Download additional applications" button. The footer contains the copyright notice: "© 2010 National Instruments. All rights reserved."



Students: Test the Built-in Temperature Logger, Log Data to File, etc.

MAX – Measurement & Automation Explorer

NI USB-TC01 "Dev1" - Measurement & Automation Explorer

File Edit View Tools Help

My System

- Data Neighborhood
- Devices and Interfaces
 - ASRL1::INSTR "COM1"
 - ASRL2::INSTR "COM2"
 - ASRL10::INSTR "LPT1"
 - NI USB-TC01 "Dev1"**
 - Network Devices
- Scales
- Software
- Remote Systems

Save Refresh Self-Test Test Panels... Create Task... Configure TEDS... Hide Help

Settings

Name	Dev1
Vendor	National Instruments
Model	NI USB-TC01
Serial Number	0163188A
Status	Present

Settings Attributes

Test Panels : NI USB-TC01: "Dev1"

Analog Input

Channel Name	Dev 1/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		

Amplitude vs. Samples Chart

Auto-scale chart

Start Stop

Close Help

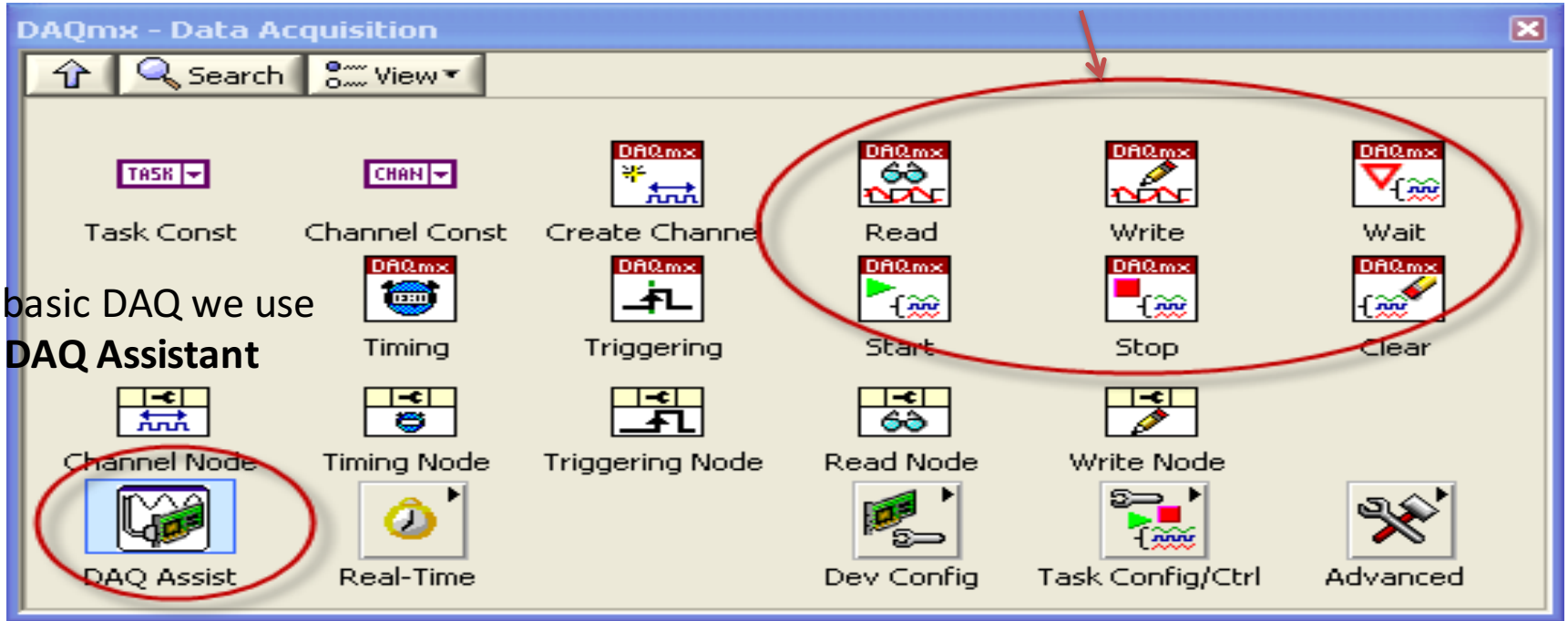


Students: Make sure that your device can be located in MAX. Run a "Self-Test" and use the "Test Panels" to make sure the device works properly.

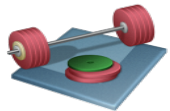
Data Acquisition Palette in LabVIEW

Functions Palette: "Measurement I/O" -> "NI DAQmx"

For more "advanced" DAQ we use these functions



For basic DAQ we use the **DAQ Assistant**

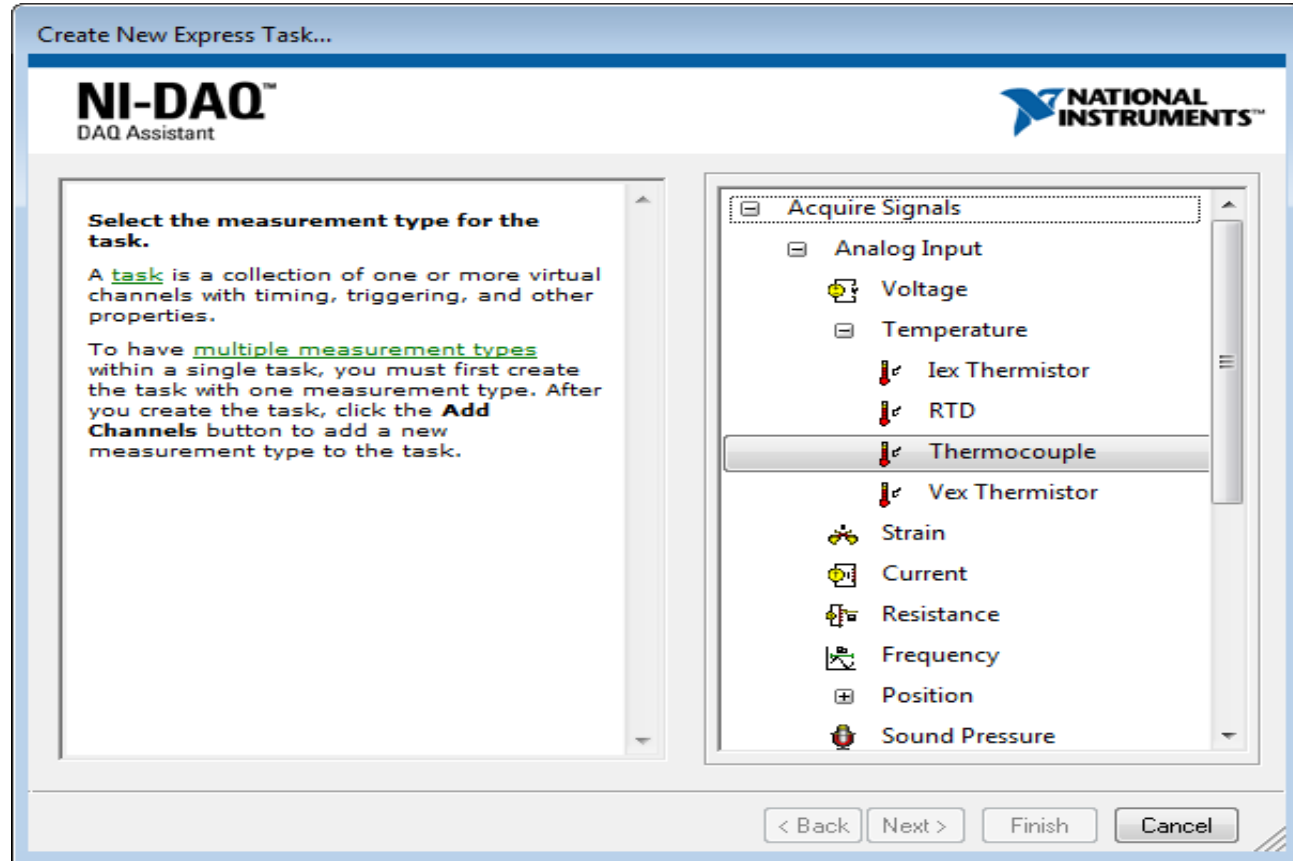


Students: Make sure that you have this palette installed. If not, install the latest DAQmx driver!

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 Input/Output Type

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.

Acquire Signals

- [-] Analog Input
 - Voltage
 - Temperature
 - [-] Iex Thermistor
 - [-] RTD
 - [-] Thermocouple
 - [-] Vex Thermistor
- [-] Strain
- [-] Current
- [-] Resistance
- [-] Frequency
- [-] Position
- [-] Sound Pressure

LabVIEW DAQ Assistant



Set Properties

DAQ Assistant

Undo Redo Run Add Channels

Express Task Connection Diagram

Channel	Value
Temperature	0

Table Display Type

Configuration Triggering Advanced Timing Logging

Channel Settings

Temperature

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

Thermocouple Setup

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

OK Cancel



Select Channel

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.

Physical

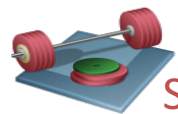
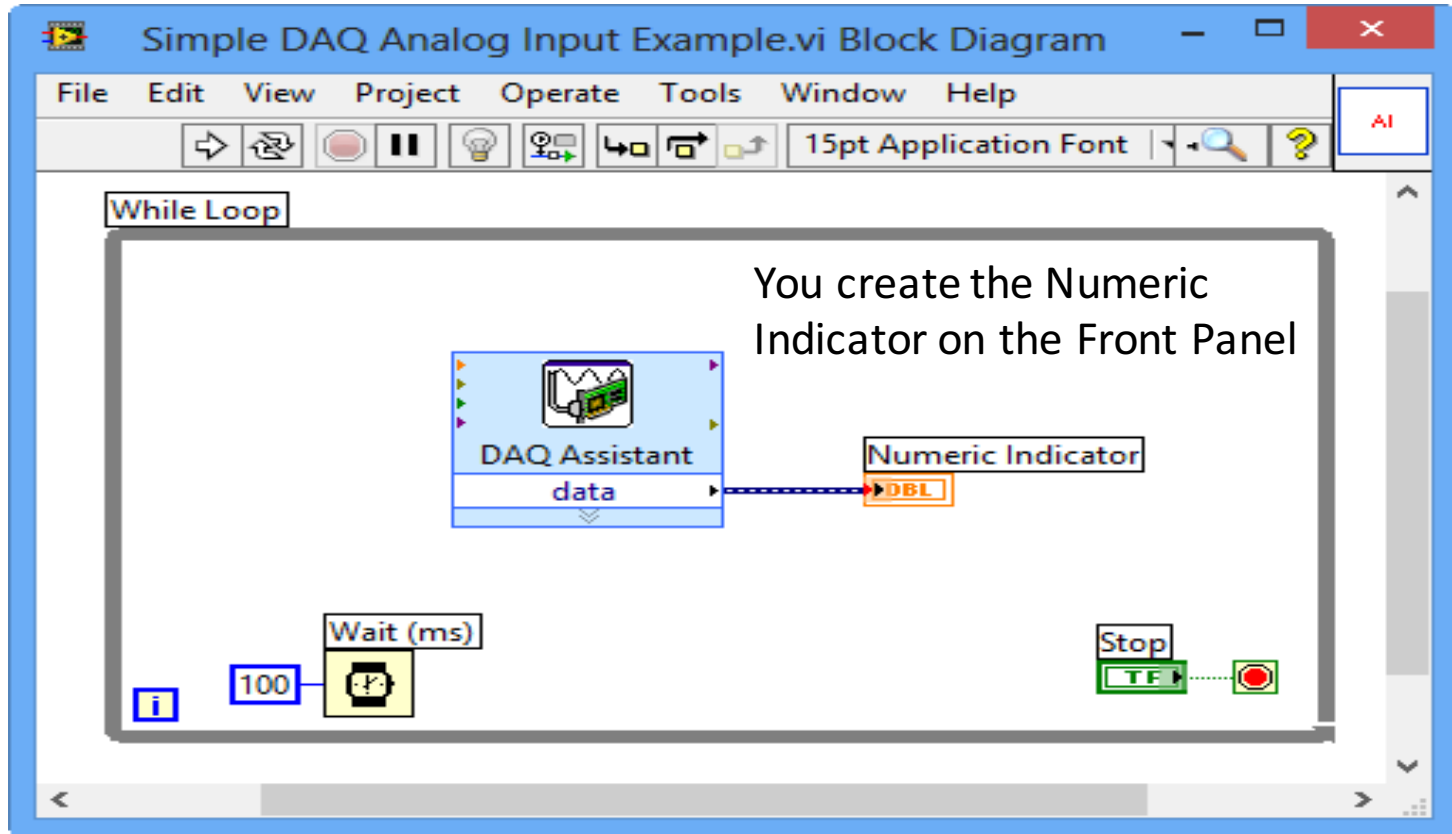
Supported Physical Channels

- [-] Dev1 (USB-TC01)
 - ai0

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

< Back Next > Finish Cancel

Read Data from TC-01 Device



Students: Create this Example and Run the Program



NATIONAL INSTRUMENTS

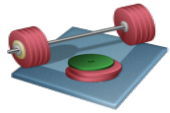
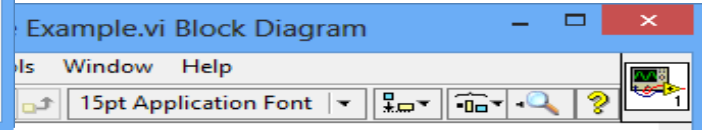
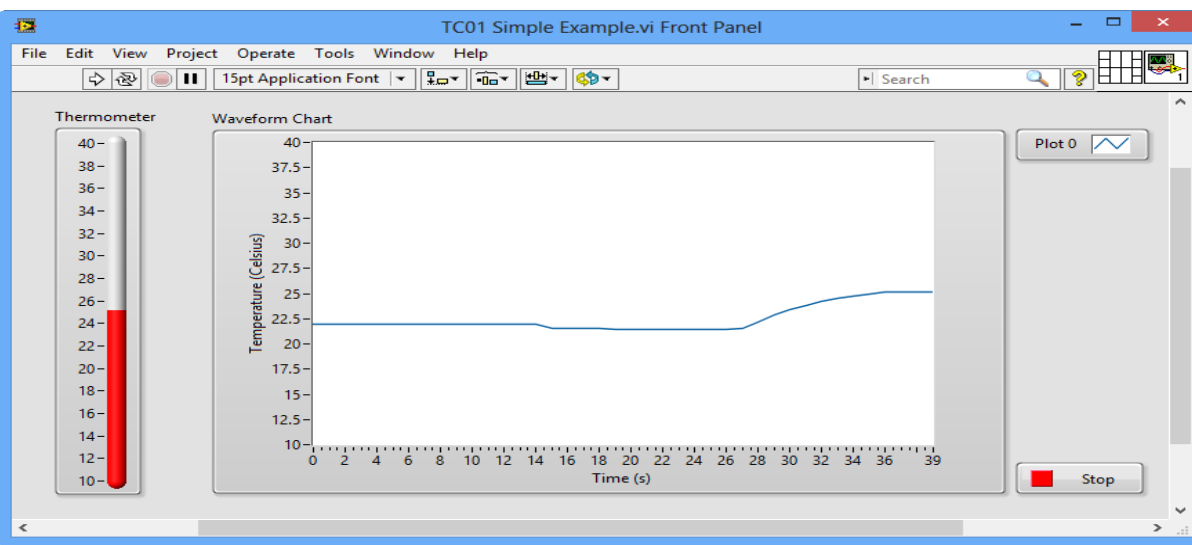
LabVIEW



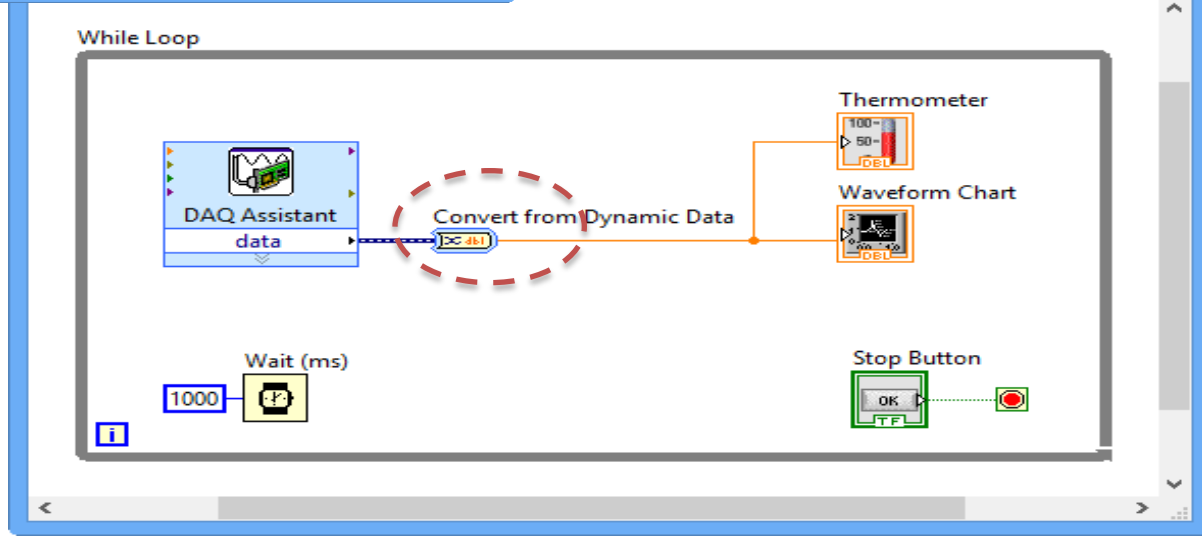
Plotting

Hans-Petter Halvorsen, M.Sc.

Plotting Temperature Data



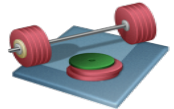
Students: Create this Example and run the program



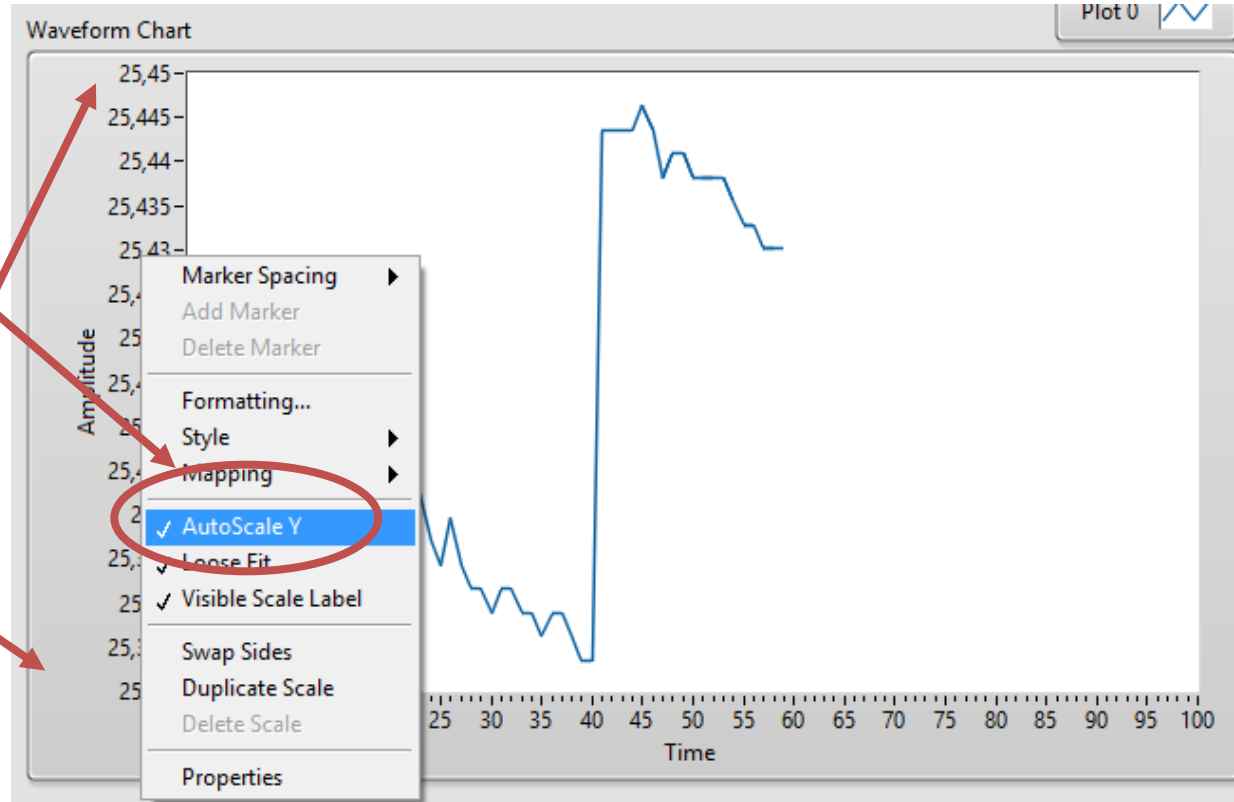
Plot Configuration

In most cases it is recommended that you turn off "Autoscaling"

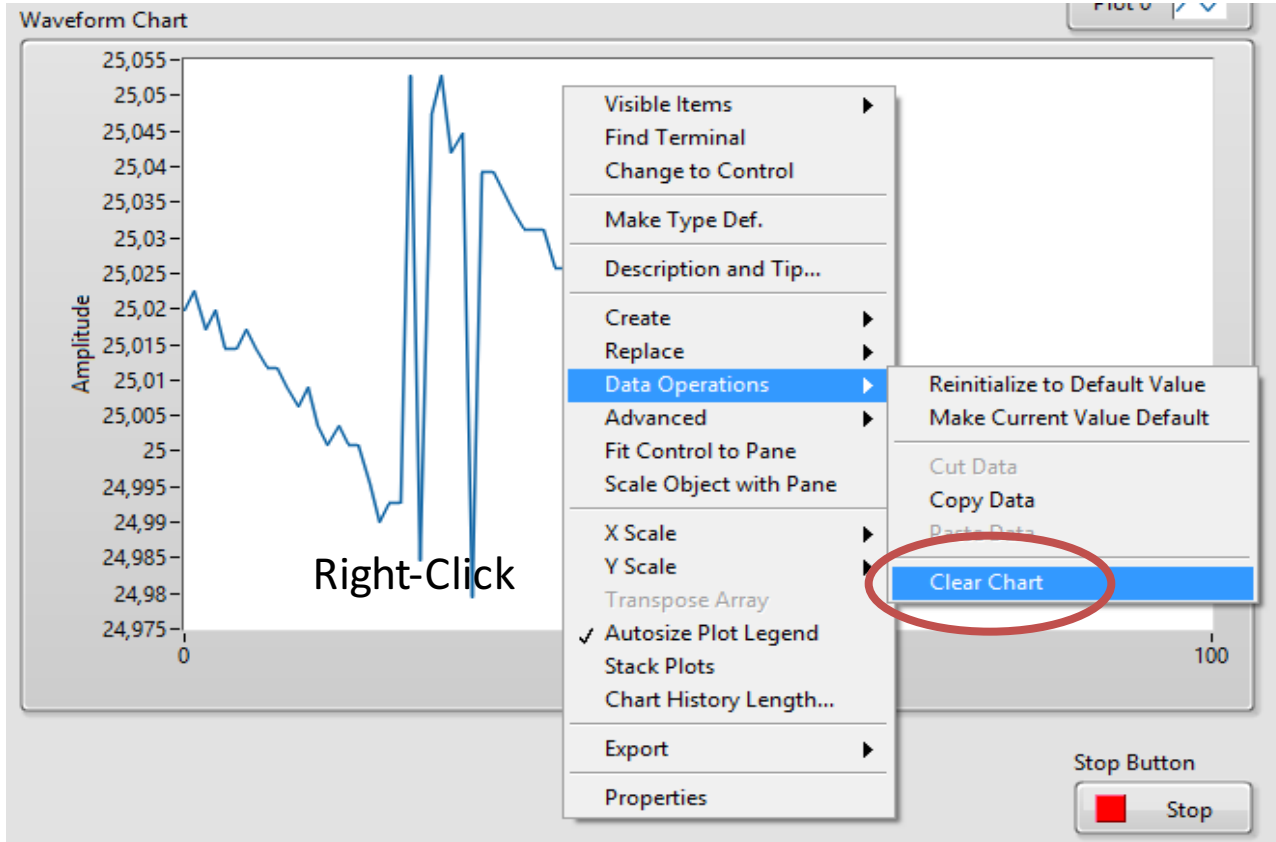
Click on the axis to set Min and Max values



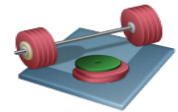
Students: Try this.



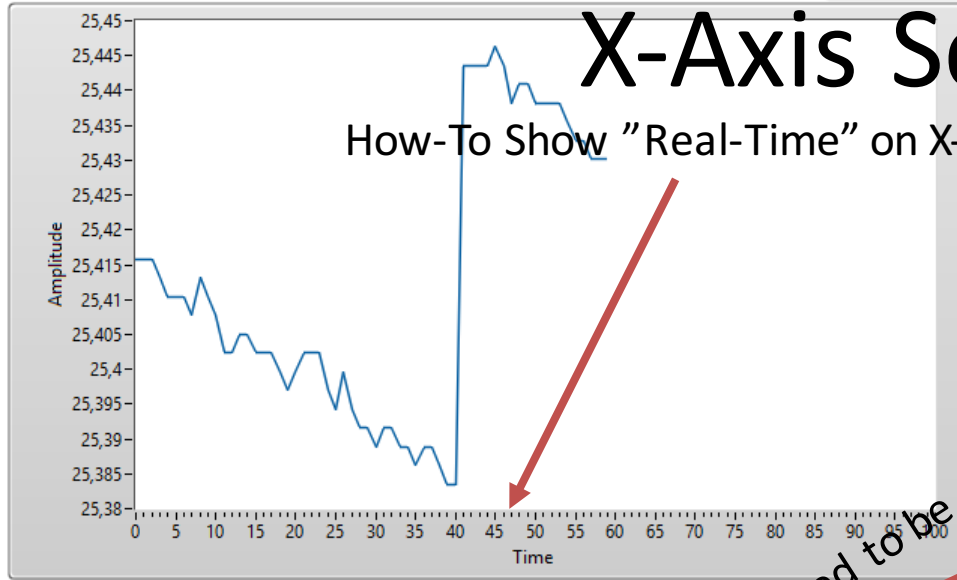
Clear Plot Contents



Students: Try this.



Waveform Chart



X-Axis Scaling

How-To Show "Real-Time" on X-Axis

Chart Properties: Waveform Chart

Appearance | Display Format | Plots | Scales | Documentation | Data Bin

Time (X-Axis)

Name: Time

Show scale label Autoscale

Show scale 0 Minimum

Log 50 Maximum

Inverted

Expand digital buses

Scaling Factors

0 Offset

0,25 Multiplier

Scale Style and Colors

Grid Style and Colors

Ignore waveform time stamp on x-axes

OK Cancel Help

Note! This is in Seconds

These need to be synchronized

Make the x Scale show Real Time

Note! For TC-01:
Max 4 samples per second (see Data Sheet)

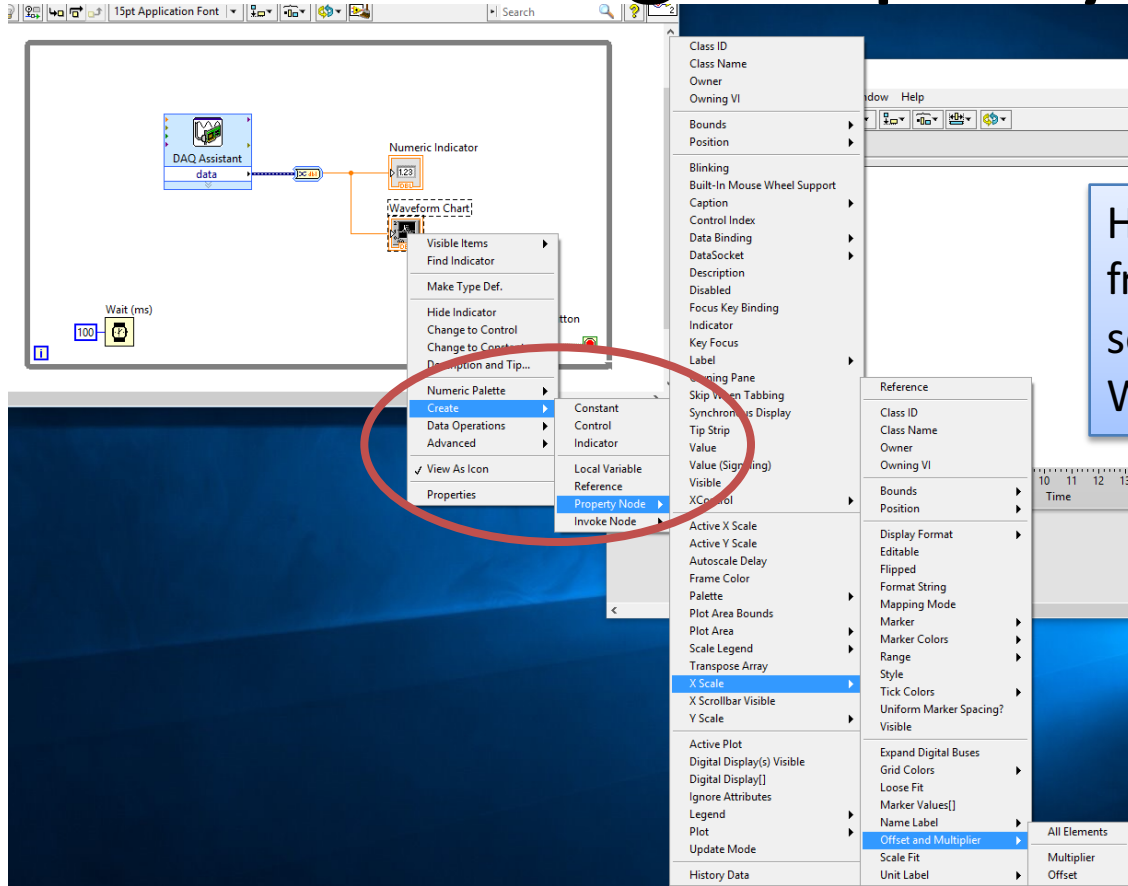
Note! This is in milliseconds

Right-click and select Properties



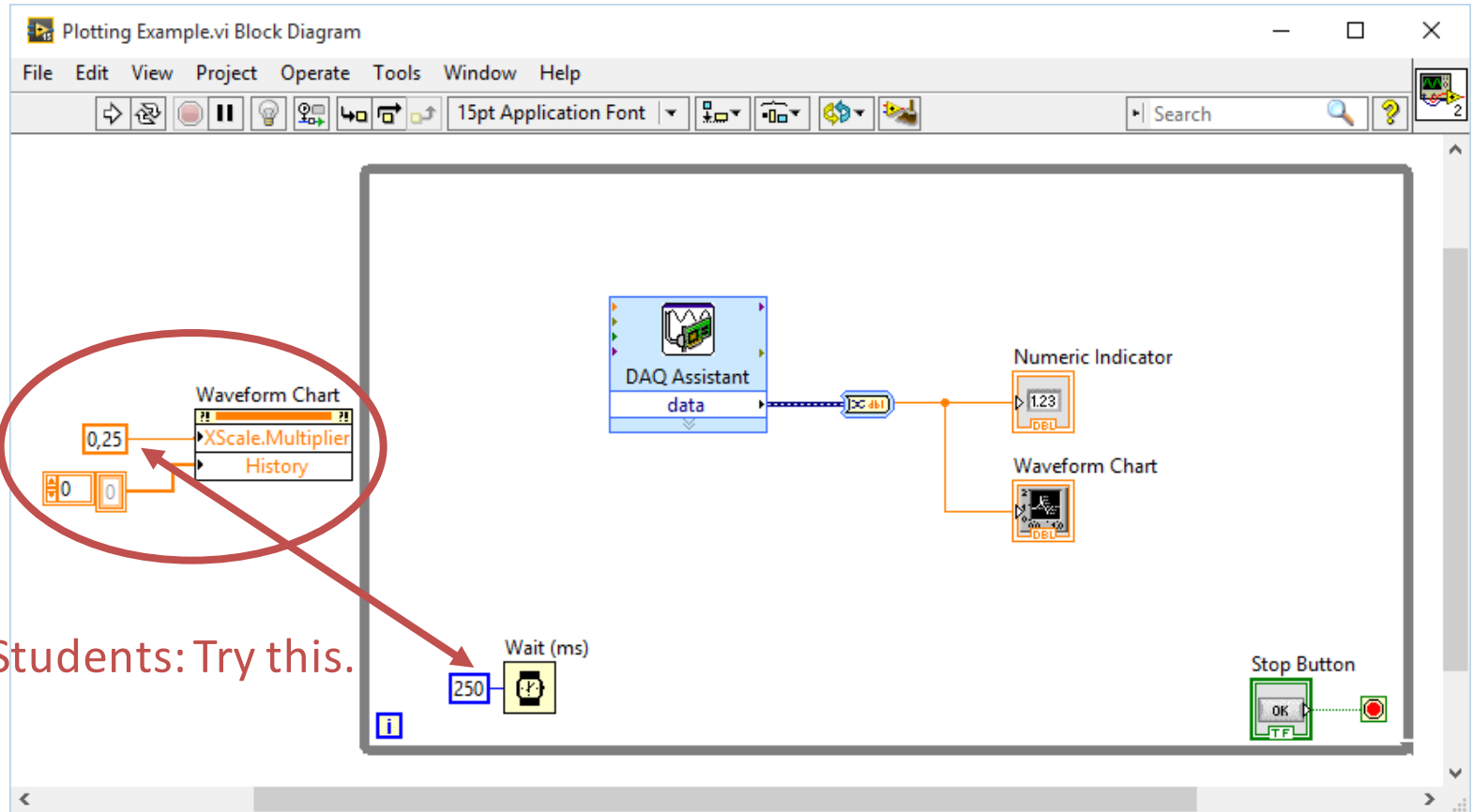
Students: Try this.

Using Property Nodes

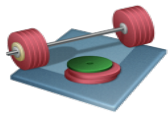
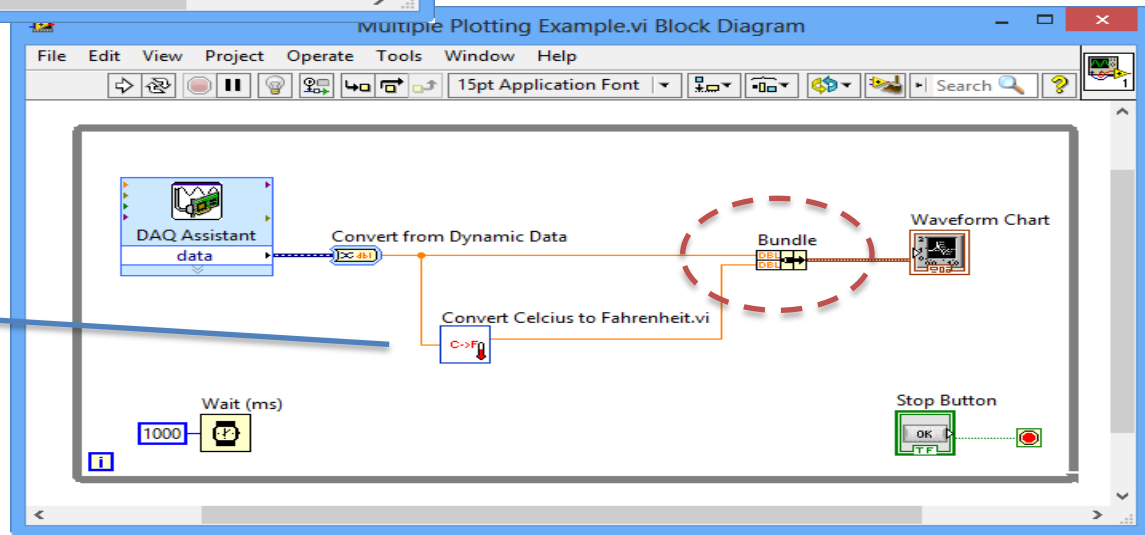
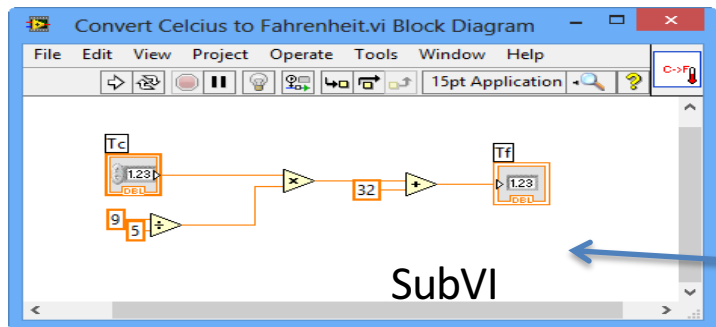
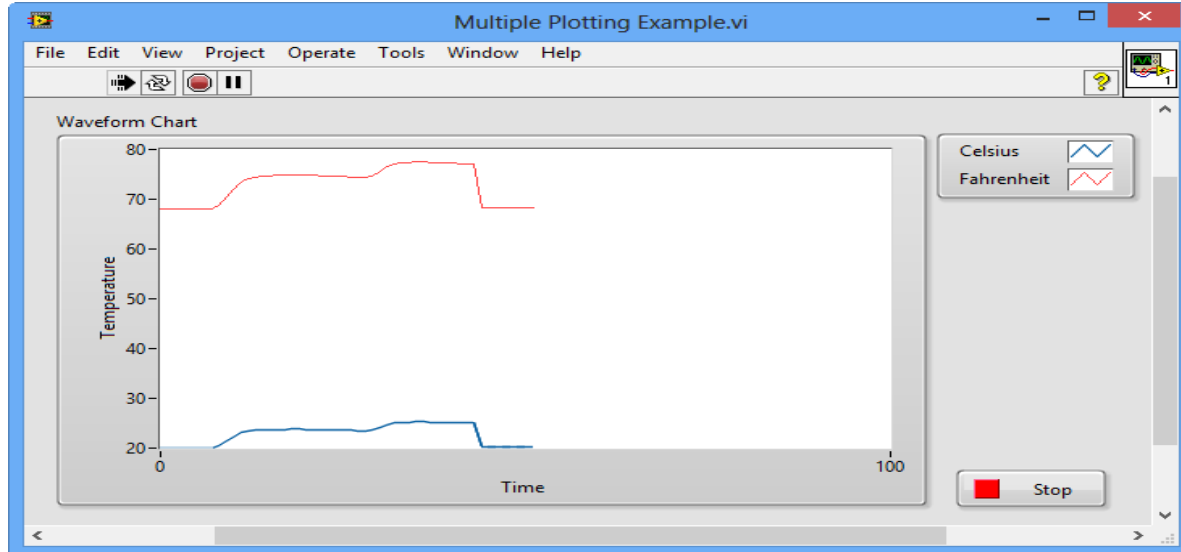


Here we will learn to set Properties from LabVIEW Code instead of setting them in the Property Window, etc. This is very useful!

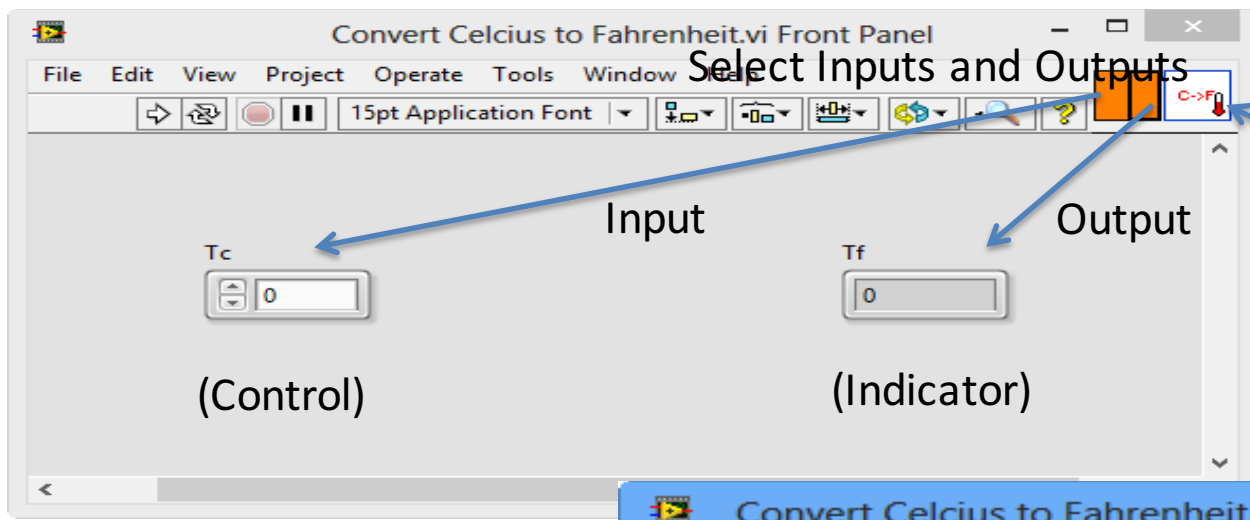
Here we have used “Property Nodes” in order to set proper scaling on X-Axis and clear the Chart



Multiple Plotting and using SubVIs



Students: Create this Example and run it

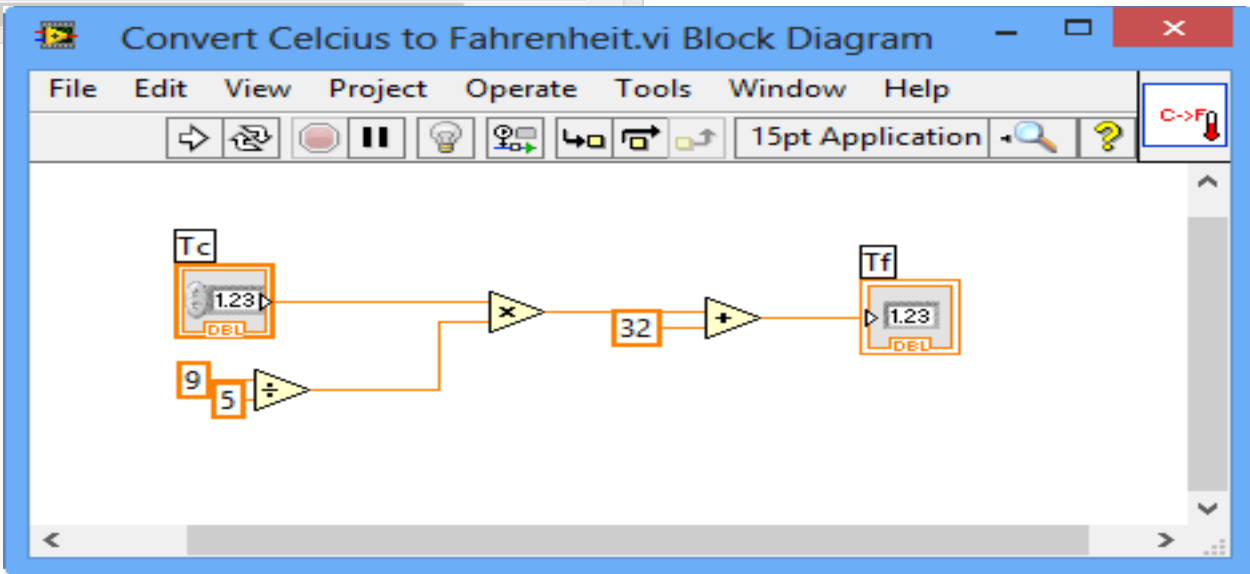


SubVI

Icon that makes it easier to understand what the SubVI is doing

A SubVI is the same as a function or a method used in other languages

$$T_F = \frac{9}{5} T_C + 32$$





NATIONAL INSTRUMENTS

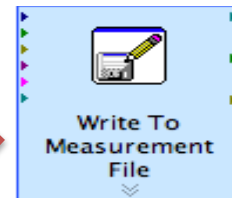
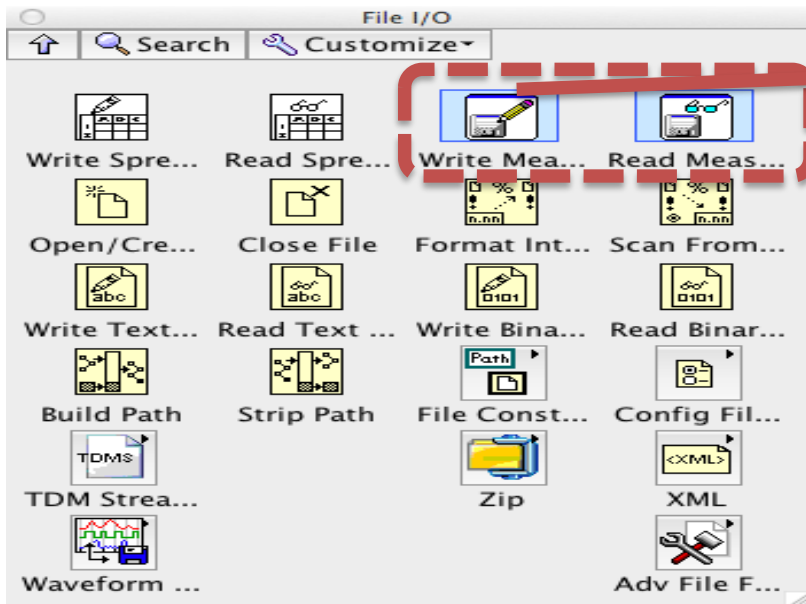
LabVIEW



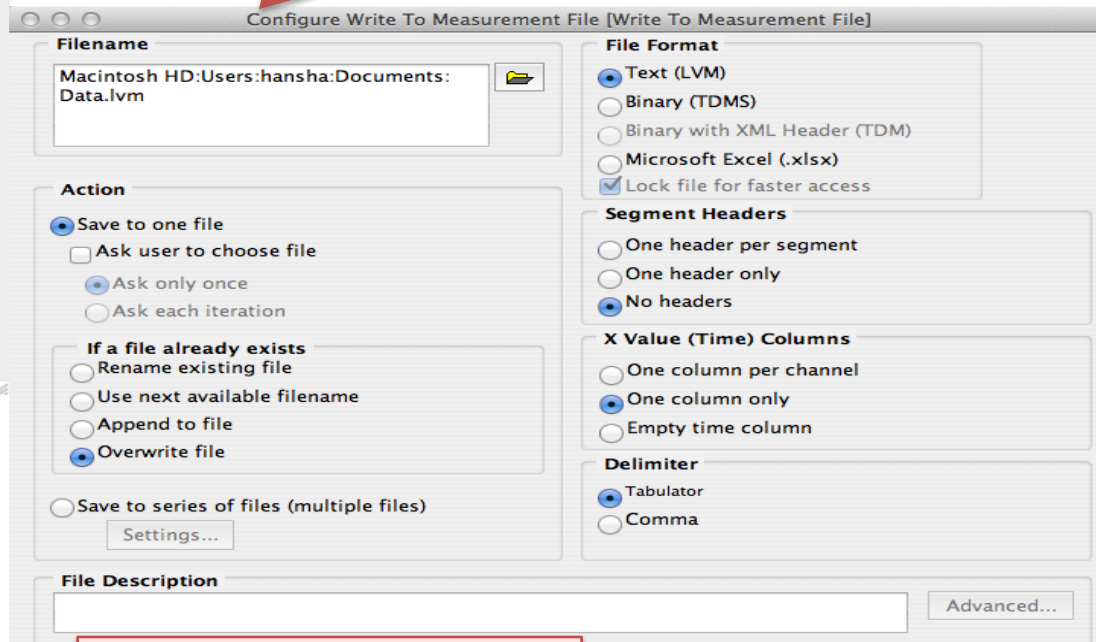
Datalogging

Hans-Petter Halvorsen, M.Sc.

Save Data to File (Data logging)



Right-click-Properties



Recommended Settings

OK

Cancel

Help

Configure Write To Measurement File [Write To Measurement File]

Filename
Macintosh HD:Users:hansha:Documents:
Data.lvm

Action

- Save to one file
 - Ask user to choose file
 - Ask only once
 - Ask each iteration
- Save to series of files (multiple files)

If a file already exists

- Rename existing file
- Use next available filename
- Append to file
- Overwrite file

Settings...

File Description

Advanced...

File Format

- Text (LVM)
- Binary (TDMS)
- Binary with XML Header (TDM)
- Microsoft Excel (.xlsx)
- 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

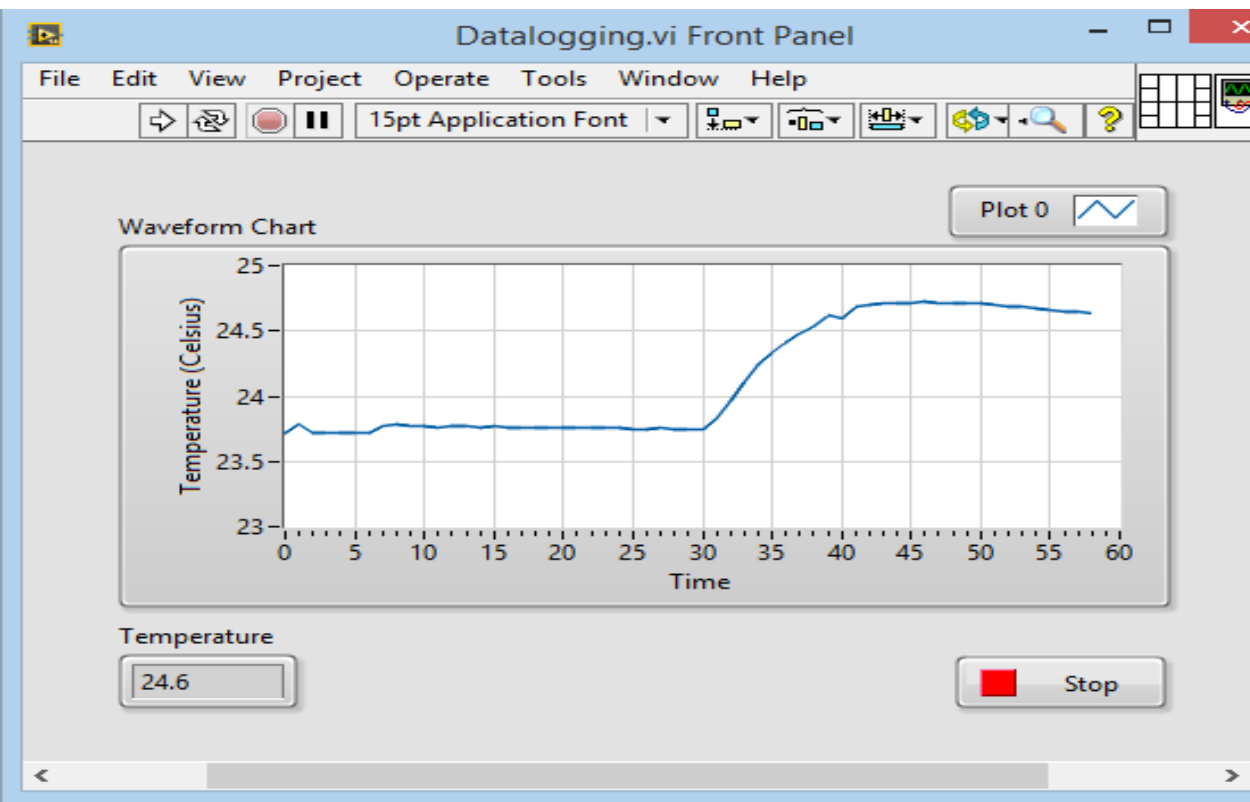
Delimiter

- Tabulator
- Comma

OK Cancel Help

Recommended Settings

Datalogging Example

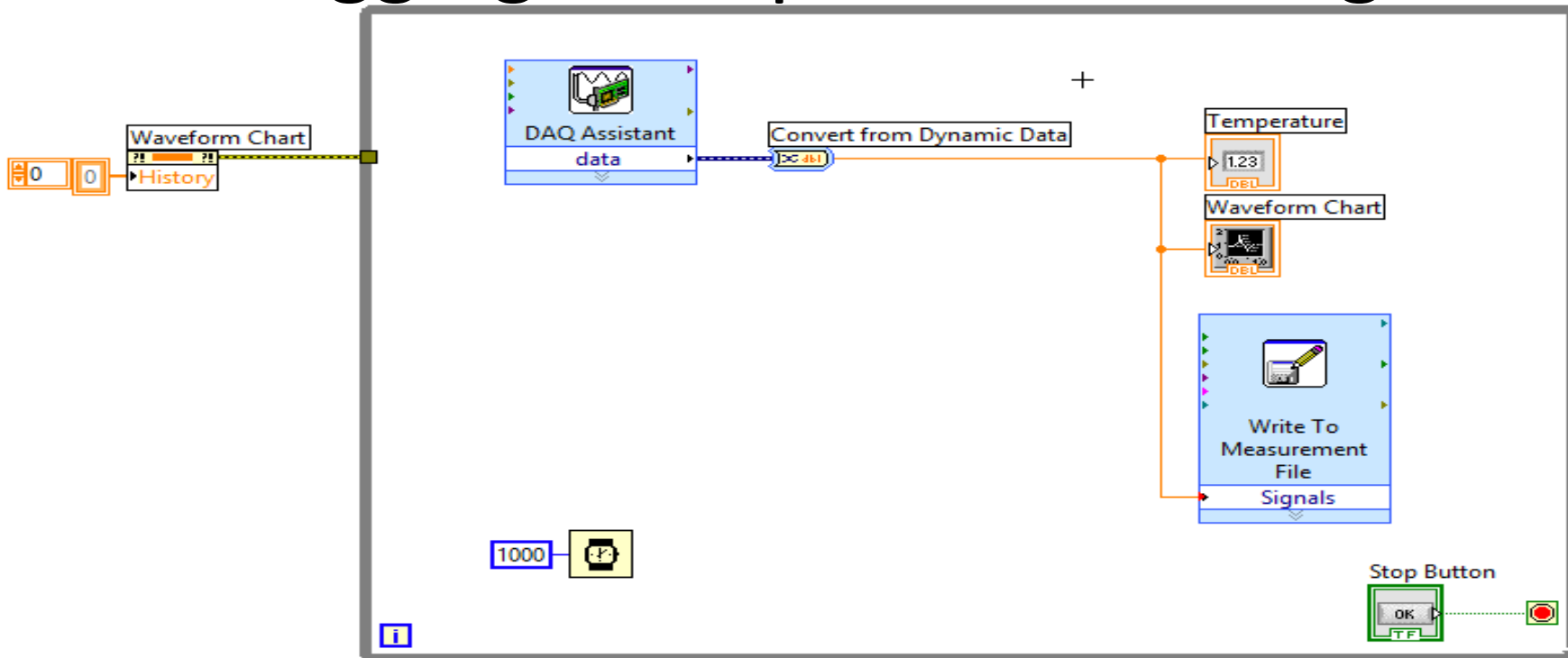


Data.lvm - Notepad

File Edit Format View Help

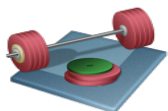
0.000000	23.722386
0.975883	23.782507
1.973000	23.714294
2.977028	23.719689
3.975200	23.719689
4.976168	23.716991
5.974145	23.714294
6.977184	23.774415
7.977247	23.779810
8.976395	23.777113
9.976493	23.771718
10.980489	23.763626
11.976687	23.771718
12.980719	23.766323
13.982748	23.763626
14.983700	23.766323
15.979765	23.763626
16.977789	23.760928
17.979809	23.760928
18.977904	23.760928
19.976963	23.758231
20.977973	23.755534
21.979071	23.755534
22.980054	23.752836
23.979137	23.752836
24.978214	23.750139
25.978157	23.747441
26.978513	23.752836

Datalogging Example – Block Diagram



Students: **(1)** Log Temperature Data, both Celsius and Fahrenheit (use SubVI) to a “Measurement File”. **(2)** Then Plot the Data in the File in Excel.

(3) You should also create a new VI in LabVIEW where you plot the data from the File in a Graph and in an Array.



Read from Measurement File - Example

Configure Read From Measurement File [Read From Measurement File]

Filename
C:\Work\Development\LabVIEW\DAQ\Data.lvm

File Format
 Text (LVM)
 Read generic text files
 Binary (TDMS)
 Binary with XML Header (TDM)
 Lock file for faster access

Time Stamps
 Relative to start of measurement
 Absolute (date and time)

Segment Size
 Retrieve segments of original size
 Retrieve segments of specified size
Samples: 100

Action
 Ask user to choose file

Generic Text File
Delimiter
 Tabulator
 Comma

Start row of numeric data
1

First row is channel names
 First column is time channel

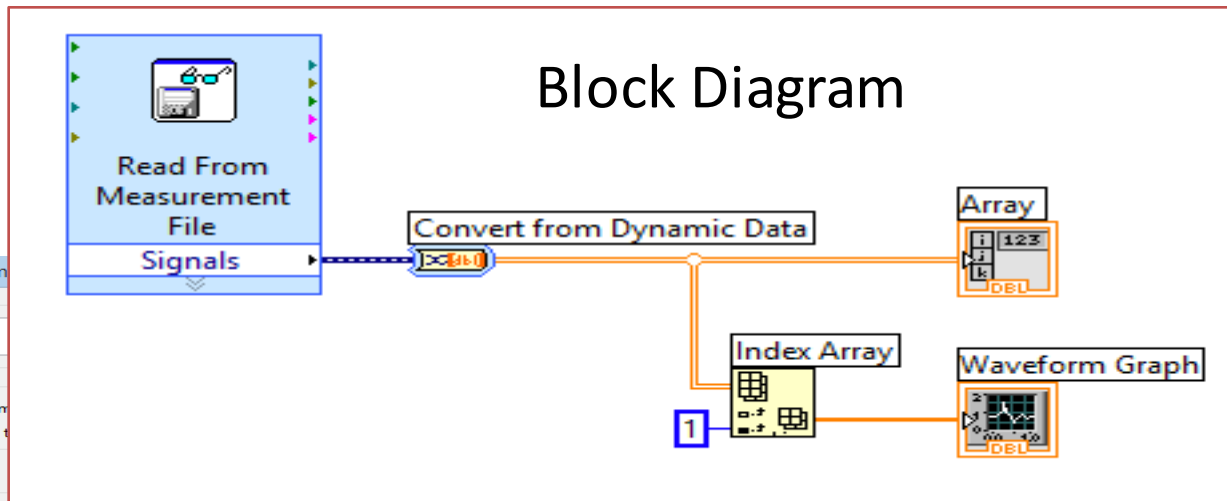
Decimal Point
 . (dot)
 , (comma)

Sample data

Sample	Time	Value
0	23.722386	
0.975883	23.782507	
1.973	23.714294	
2.977028	23.719689	
3.9752	23.719689	
4.976168	23.716991	
5.974145	23.714294	
6.977184	23.774415	
7.977247	23.77981	
8.976395	23.777113	
9.976493	23.771718	
10.980489	23.763626	
11.976687	23.771718	

Read File Now

OK Cancel Help



Right-click-Properties

Read From Measurement File
Signals

Recommended Settings

Configure Read From Measurement File [Read From Measurement File]

Filename

C:\Work\Development\LabVIEW\DAQ\Data.lvm

File Format

- Text (LVM)
 - Read generic text files
- Binary (TDMS)
- Binary with XML Header (TDM)
 - Lock file for faster access

Action

- Ask user to choose file

Time Stamps

- Relative to start of measurement
- Absolute (date and time)

Segment Size

- Retrieve segments of original size
- Retrieve segments of specified size

Samples

100

Generic Text File

Delimiter

- Tabulator
- Comma

Start row of numeric data

1

- First row is channel names
- First column is time channel

Decimal Point

- . (dot)
- , (comma)

Sample data

0	23.722386
0.975883	23.782507
1.973	23.714294
2.977028	23.719689
3.9752	23.719689
4.976168	23.716991
5.974145	23.714294
6.977184	23.774415
7.977247	23.77981
8.976395	23.777113
9.976493	23.771718
10.980489	23.763626
11.976687	23.771718

Read File Now

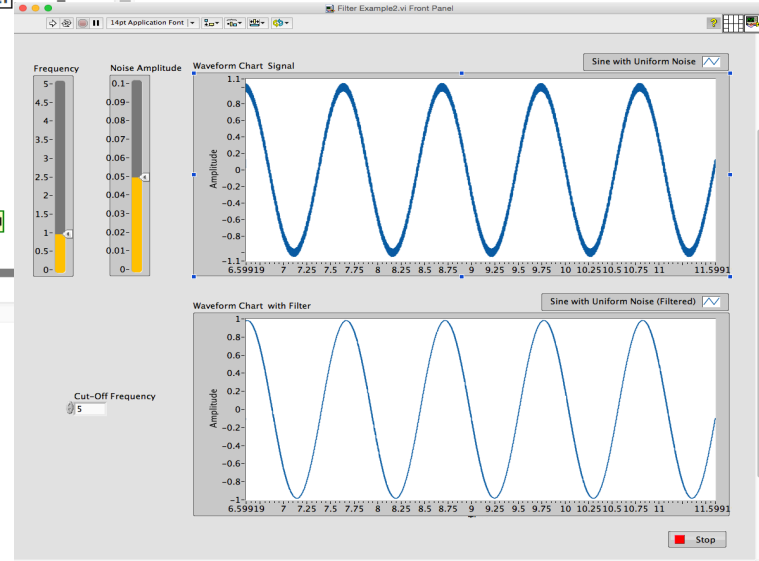
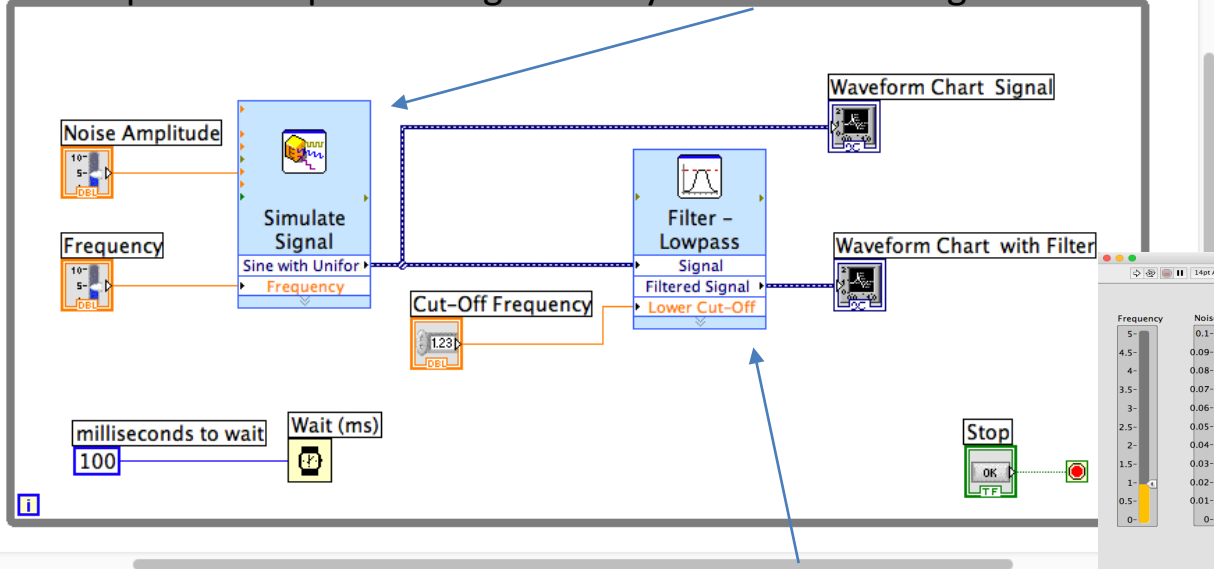
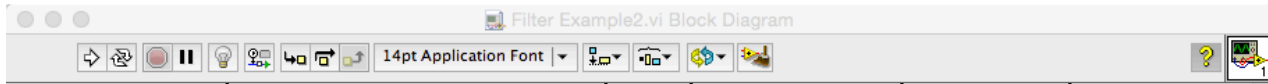
Recommended Settings

OK

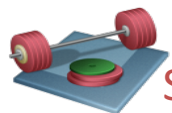
Cancel

Help

Using a Lowpass Filter to reduce Noise



Functions palette: Express -> Signal Analysis -> Filter



Students: Create this Example.
Then Include the Filter in your Logging Application

Signal

Signal type
Sine

Frequency (Hz) 10.3 Phase (deg) 0

Amplitude 1 Offset 0 Duty cycle (%) 50

Add noise

Noise type
Uniform White Noise

Noise amplitude 0.6 Seed number -1 Trials 1

Timing

Samples per second (Hz) 20000 Simulate acquisition timing

Number of samples 2000 Automatic Run as fast as possible

Integer number of cycles

Actual number of samples 2000

Actual frequency 10.3

Time Stamps

Relative to start of measurement
 Absolute (date and time)

Reset Signal

Reset phase, seed, and time stamps
 Use continuous generation

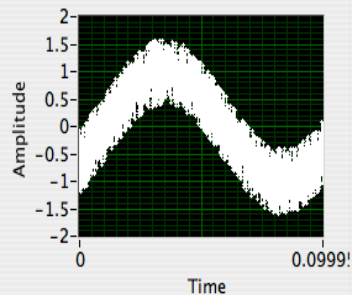
Signal Name

Use signal type name

Signal name
Sine with Uniform Noise

OK Cancel Help

Result Preview



Properties

Configure Filter [Filter - Lowpass]

Filtering Type

Lowpass

Filter Specifications

Cutoff Frequency (Hz) 1500

High cutoff frequency (Hz) 400

Finite impulse response (FIR) filter

Taps 29

Infinite impulse response (IIR) filter

Topology

Butterworth

Order

1

Input Signal

Result Preview

View Mode

Signals Show as spectrum

Transfer function

Scale Mode

Magnitude in dB
 Frequency in log

OK Cancel Help



NATIONAL INSTRUMENTS

LabVIEW



Measurement Filter

from Scratch

(You may skip this part)

Hans-Petter Halvorsen, M.Sc.

Lowpass Filter/Measurement Filter

The measured signals contains noise, so we should remove the noise using a Filter

The differential equation for a Measurement filter may be given as:

$$T_f \dot{y}_{mf}(t) = y_m(t) - y_{mf}(t)$$

Since we shall implement the Measurement Filter in a computer, we need to make a discrete version of the filter. We use the Euler Backward discretization method:

$$\dot{x} \approx \frac{x(t_k) - x(t_{k-1})}{T_s} \quad \text{Where } T_s \text{ is the Sampling Time}$$

Then we get:

$$y_{mf}(t_k) = \frac{T_f}{T_f + T_s} y_{mf}(t_{k-1}) + \frac{T_s}{T_f + T_s} y_m(t_k)$$

Or:
$$y_{mf}(t_k) = (1 - a)y_{mf}(t_{k-1}) + ay_m(t_k)$$

where
$$a = \frac{T_s}{T_f + T_s}$$

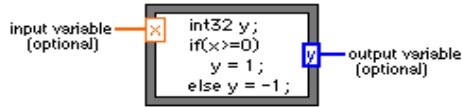
This discrete Measurement Filter can be easily implemented in a computer using e.g., LabVIEW, C#, etc.

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

LabVIEW Formula Node

Context Help

Formula Node

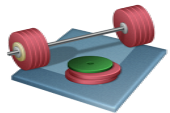
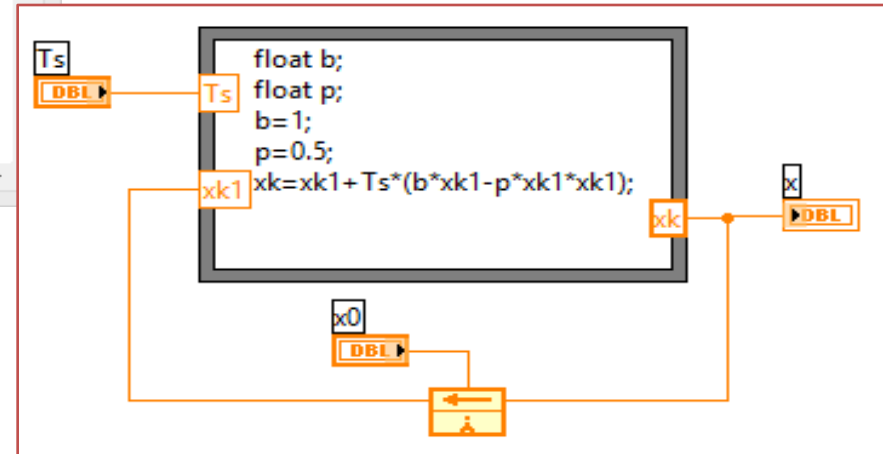


Evaluates mathematical formulas and expressions similar to C on the block diagram. The following built-in functions are allowed in formulas: abs, acos, acosh, asin, asinh, atan, atan2, atanh, ceil, cos, cosh, cot, csc, exp, expm1, floor, getexp, getman, int, intrz, ln, lnp1, log, log2, max, min, mod, pow, rand, rem, sec, sign, sin, sinc, sinh, sizeofDim, sqrt, tan, tanh. There are some differences between the parser in the Mathematics VIs and the Formula Node.

[Detailed help](#)

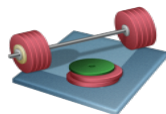
Formula Node: Create and use C code within LabVIEW

Example:



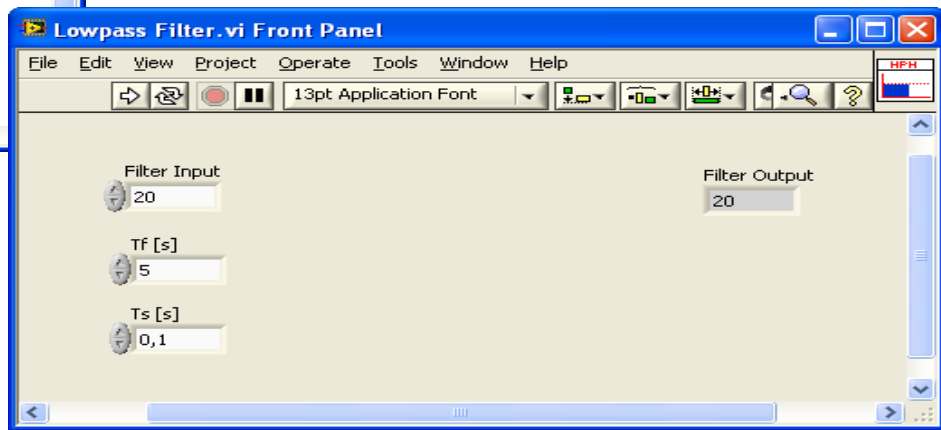
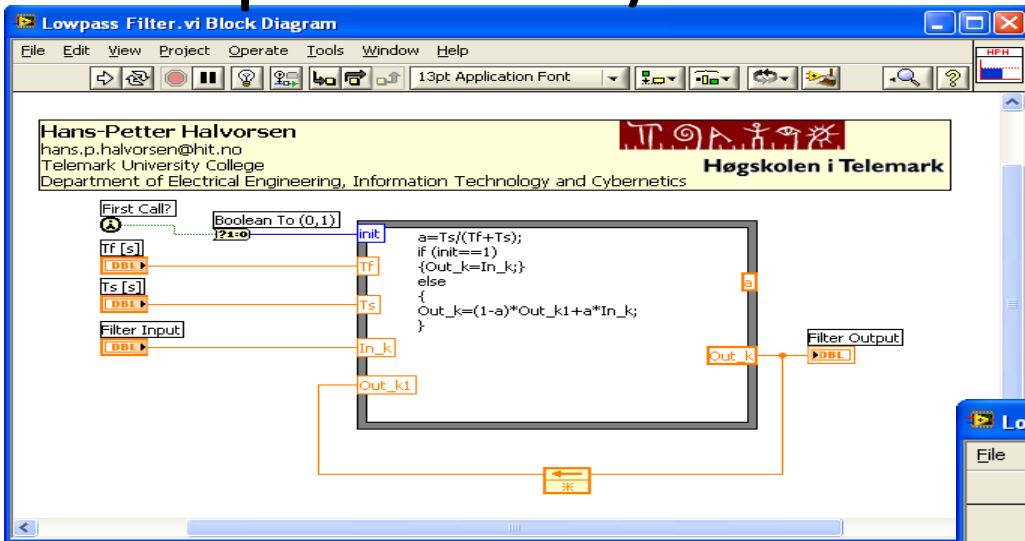
Students: Use the LabVIEW Formula Node in order to implement the Measurement Filter.

Lowpass Filter/Measurement Filter - Example



Students: Implement this Example using a LabVIEW Formula Node.

When finished, try to log data from your sensor with and without the Measurement Filter. Compare the results.



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


$$y_{mf}(t_k) = (1 - a)y_{mf}(t_{k-1}) + ay_m(t_k)$$

Lowpass Filter.vi Block Diagram

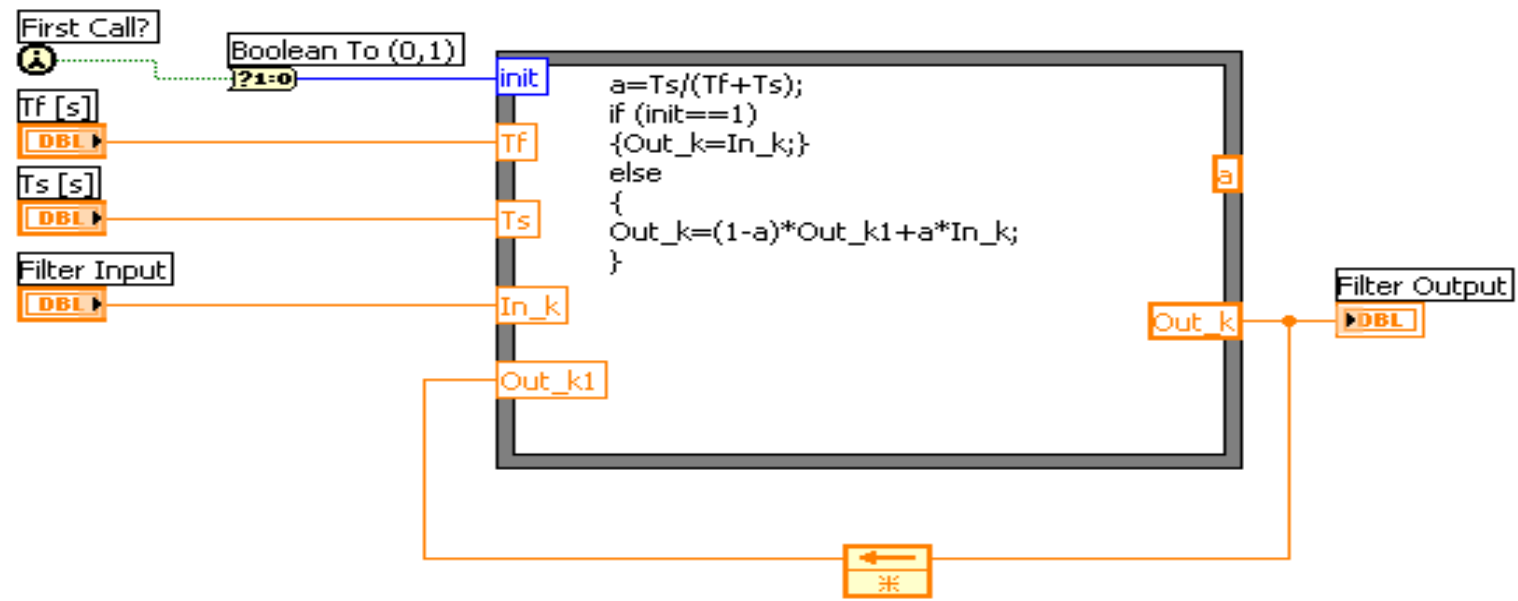
File Edit View Project Operate Tools Window Help



Hans-Petter Halvorsen
hans.p.halvorsen@hit.no
Telemark University College
Department of Electrical Engineering, Information Technology and Cybernetics

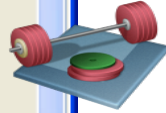


Høgskolen i Telemark

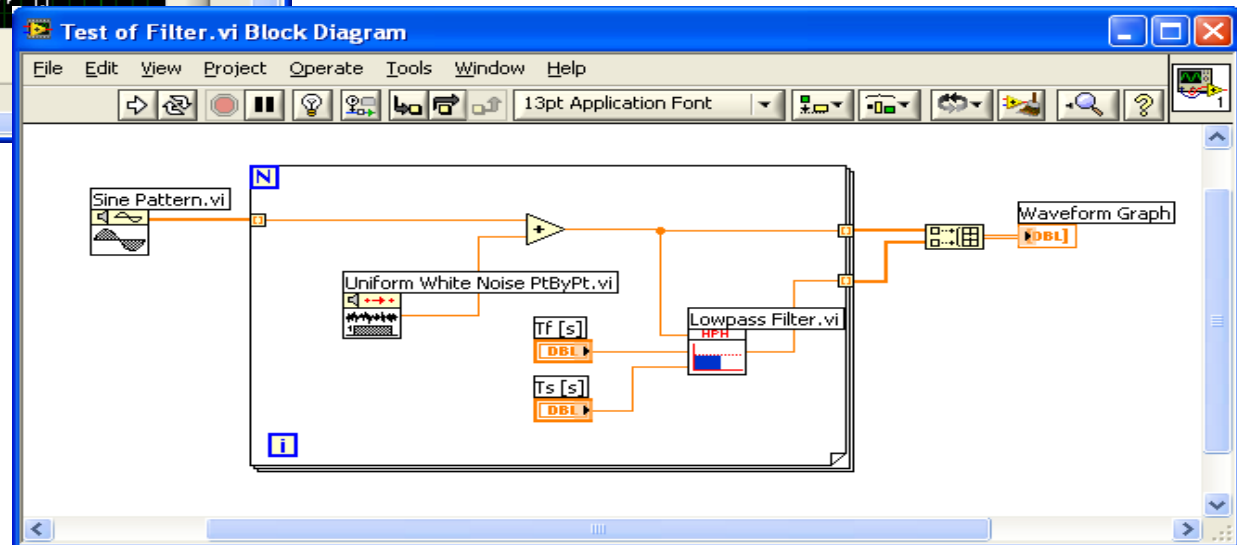
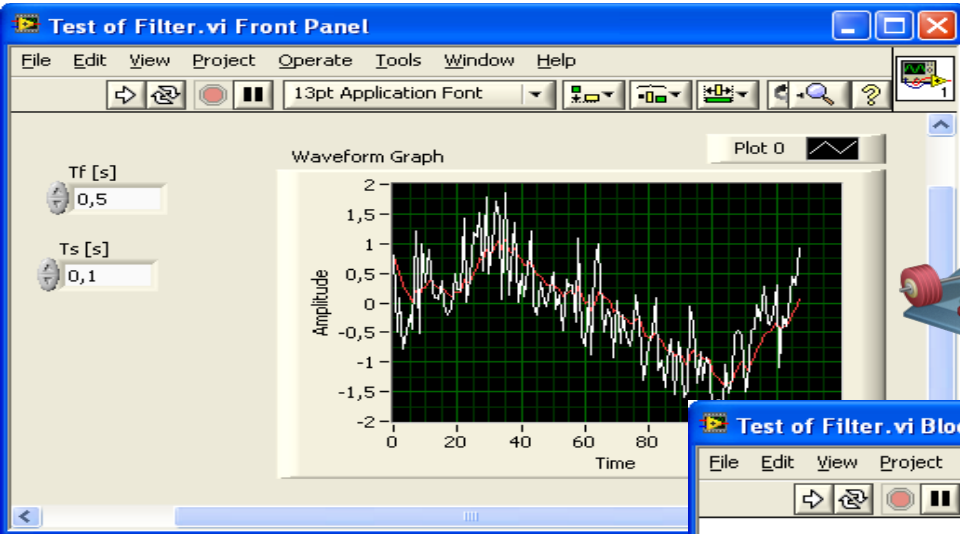


Testing the Filter

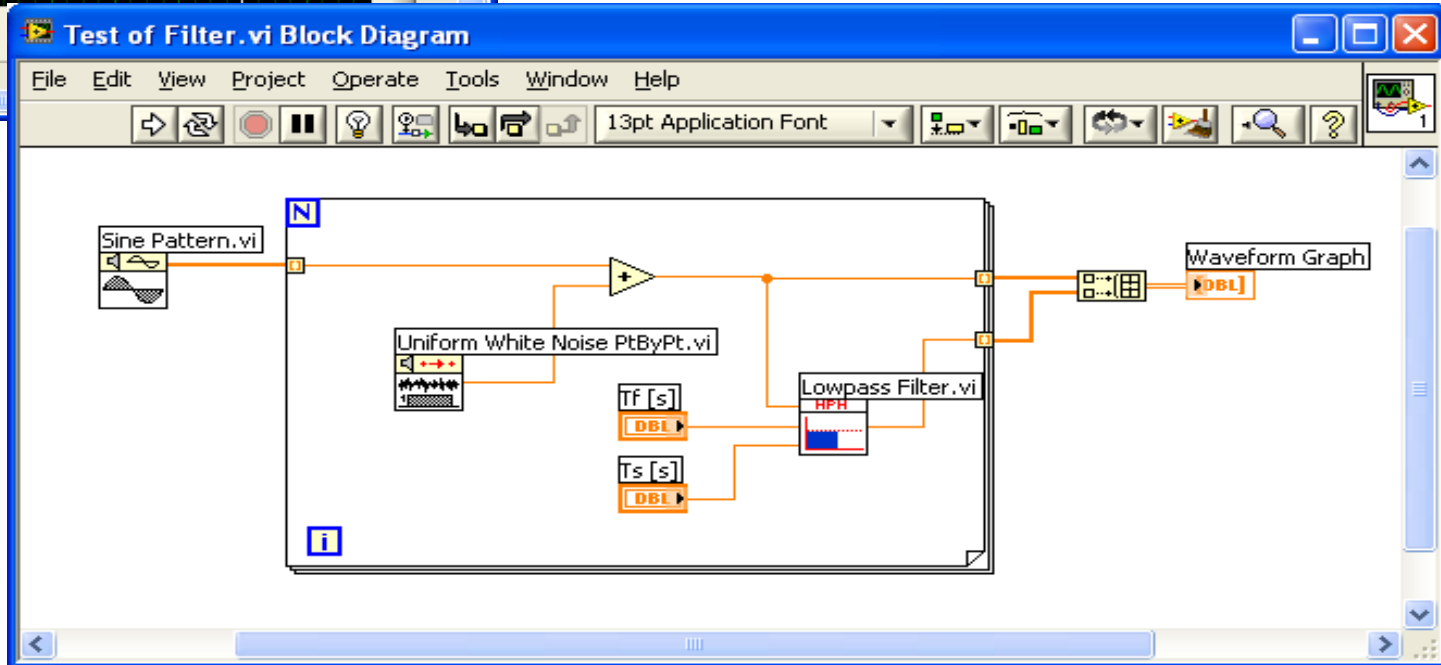
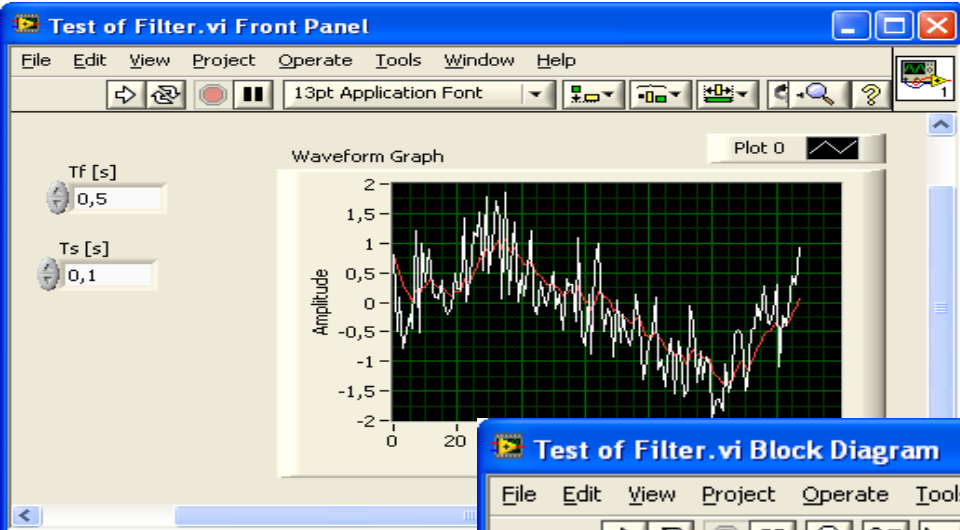
In this example we add noise to a Sine function. We then use the Measurement Filter to see if we can remove the noise afterwards.



Students: Try this Example.
Next: Use the Filter on your Datalogging System



As you can see this gives a good results.
The filter removes the noise from the signal.



Additional LabVIEW Resources

Here you will find lots of Videos, Tutorials and Exercises



- LabVIEW Training for Students (National Instruments):

<http://ni.com/students/learnlabview>

- LabVIEW Course:

<http://home.hit.no/~hansha/?training=labview>

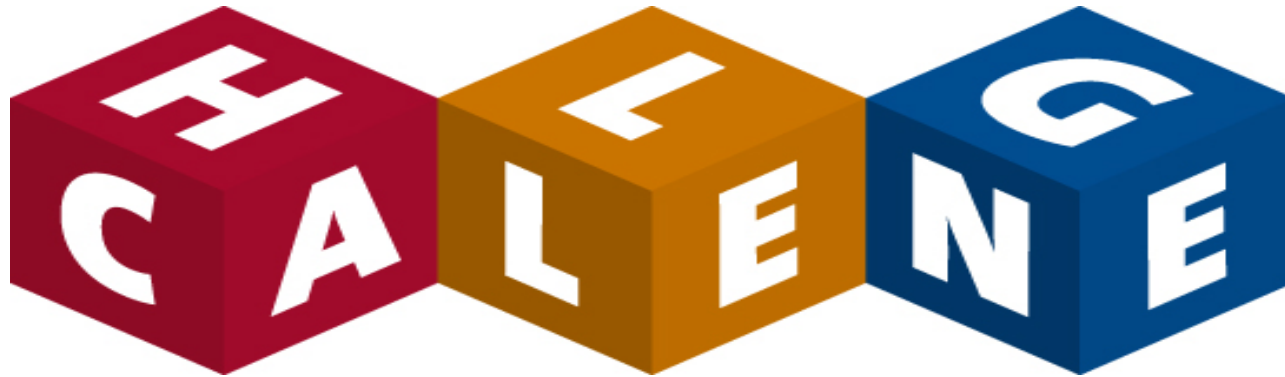
Learning by Doing!

It is recommended that you watch some of the videos before you read further



NATIONAL INSTRUMENTS

LabVIEW



Do you need more Practice? - Select a Challenge

Hans-Petter Halvorsen, M.Sc.



Temperature Logging

1. Create Logging App:

- Log the temperature in your house e.g., during the night using the TC-01 Thermocouple device.
- Plot the temperature in a Chart
- Log the temperature to a File

2. Create Analysis App:

- Read the temperature data from the file into LabVIEW
- Find Max temperature and Min temperature using built-in functions in LabVIEW
- Find also the Average/Mean temperature and the Standard deviation using built-in functions in LabVIEW

3. Do Analysis in Excel:

- Import the data into Excel and create a Plot
- Find Max, Min, Mean/AVG, SD using Excel
- Compare the results

Hans-Petter Halvorsen, M.Sc.



University College of Southeast Norway

www.usn.no

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

Blog: <http://home.hit.no/~hansha/>

