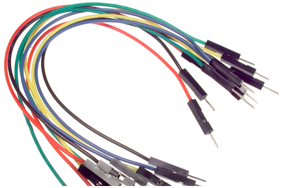
https://www.halvorsen.blog

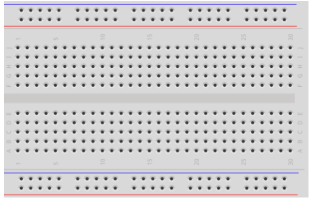
Thermistor

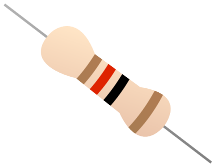
Hans-Petter Halvorsen

Hardware

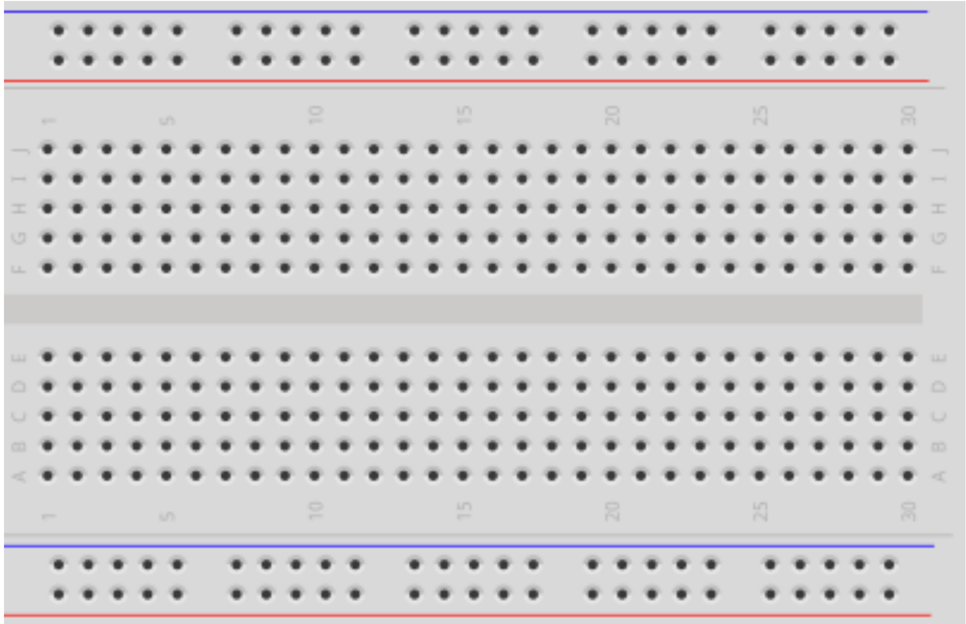
- DAQ Device (e.g., USB-6008)
- Breadboard
- Thermistor 10K (Temperature Sensor)
- Resistor, $R = 10k\Omega$
- Wires (Jumper Wires)



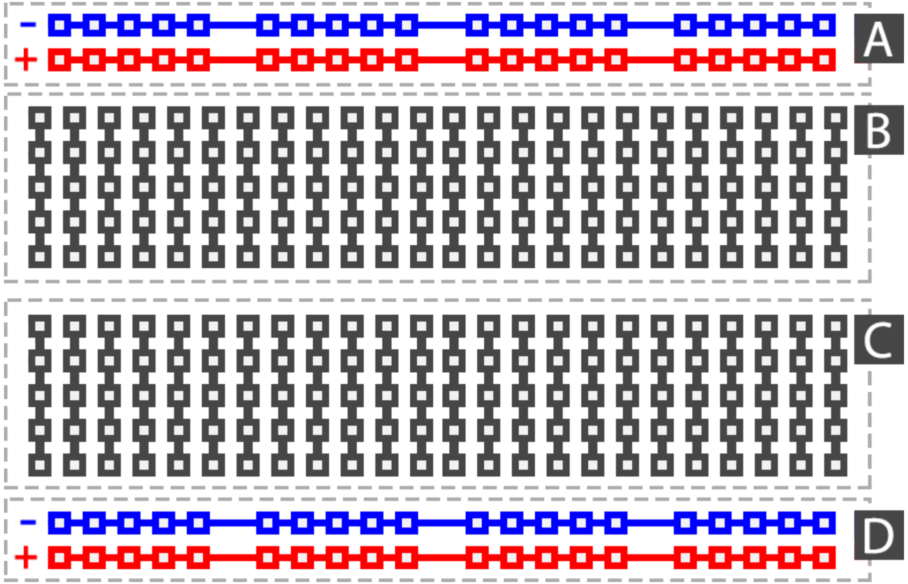




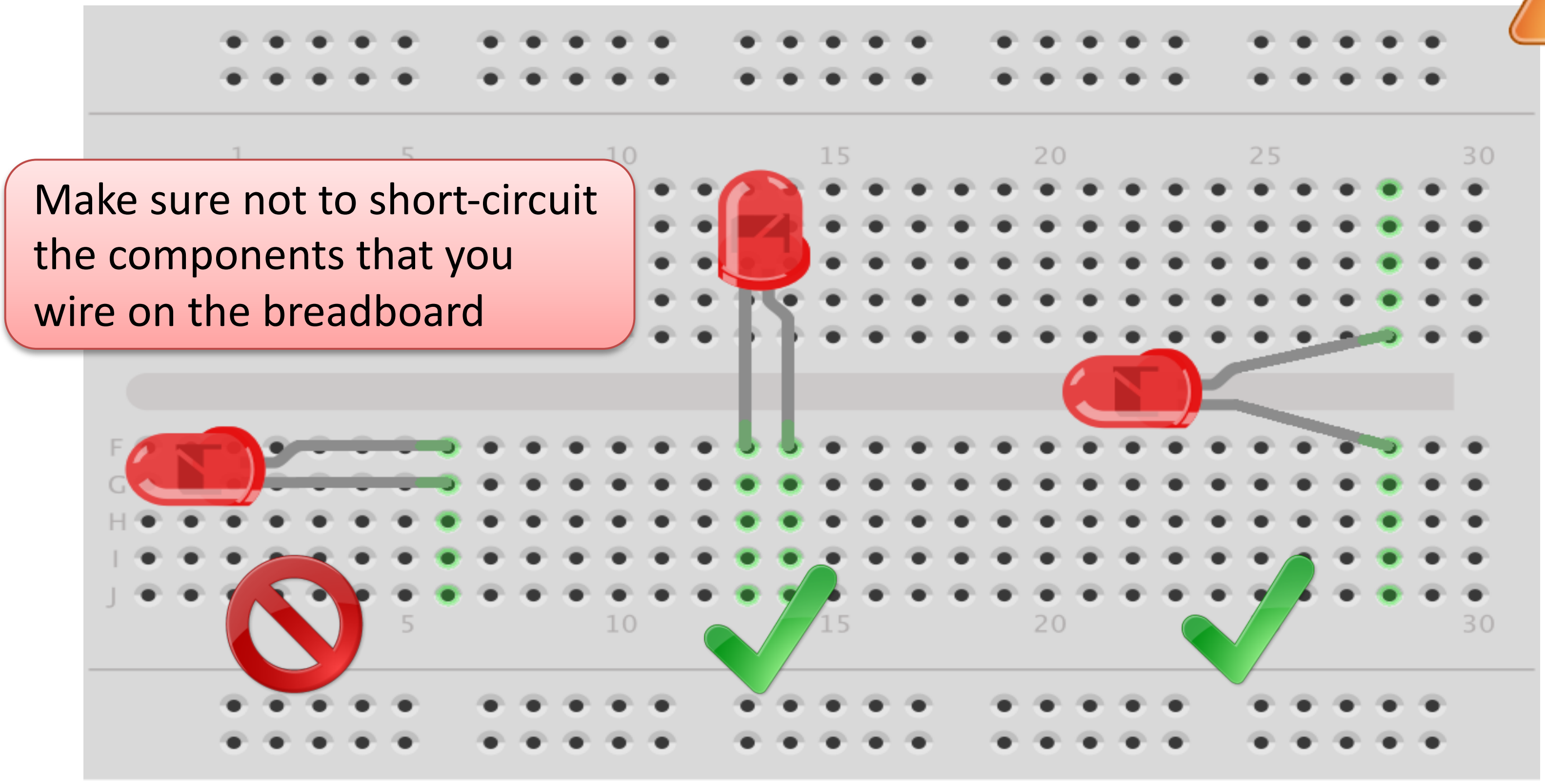
Breadboard



A breadboard is used to wire electric components together



Breadboard Wiring



The Breadboard is used to connect components and electrical circuits

https://www.halvorsen.blog

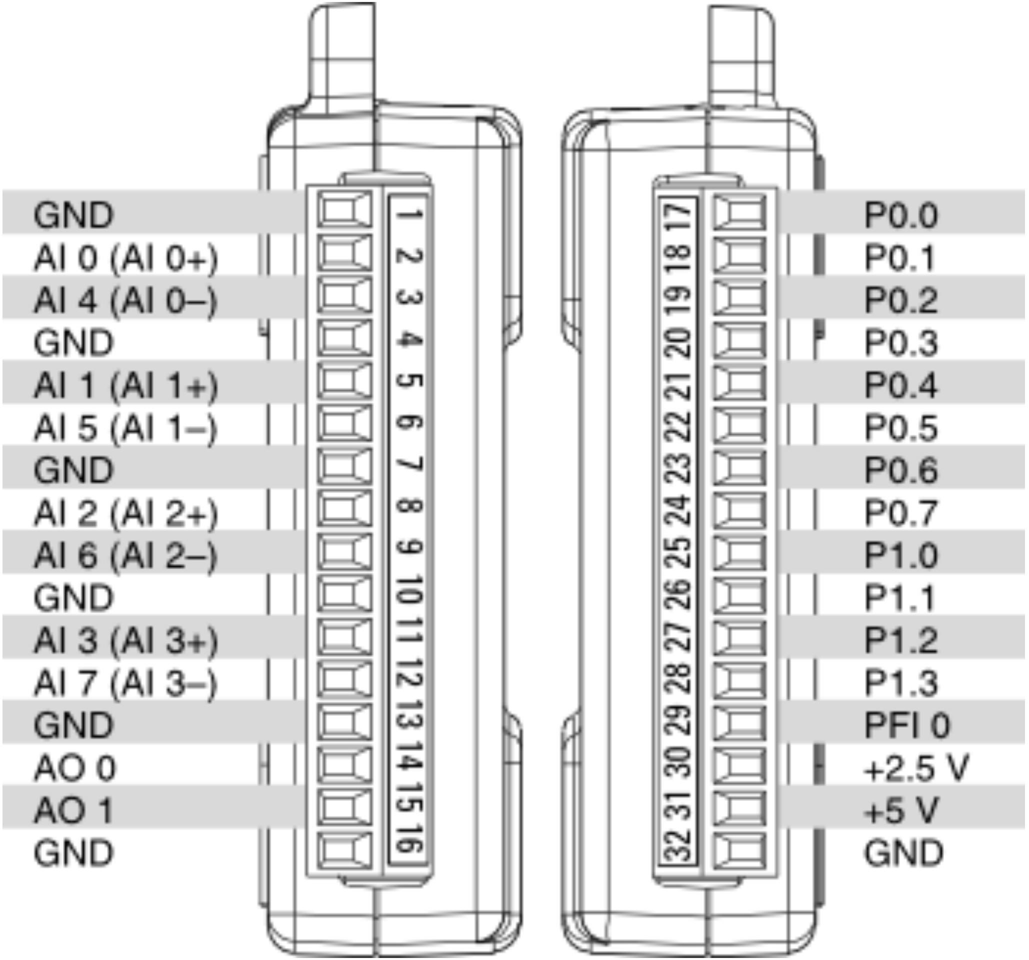
USB-6008

Hans-Petter Halvorsen

USB-6008

1
17
NATIONAL
INSTRUMENTS
Analog
Digital
NI USB-6008
8 Inputs, 12-bit, 10 kS/s Multifunction I/O
16
32

I/O Pins



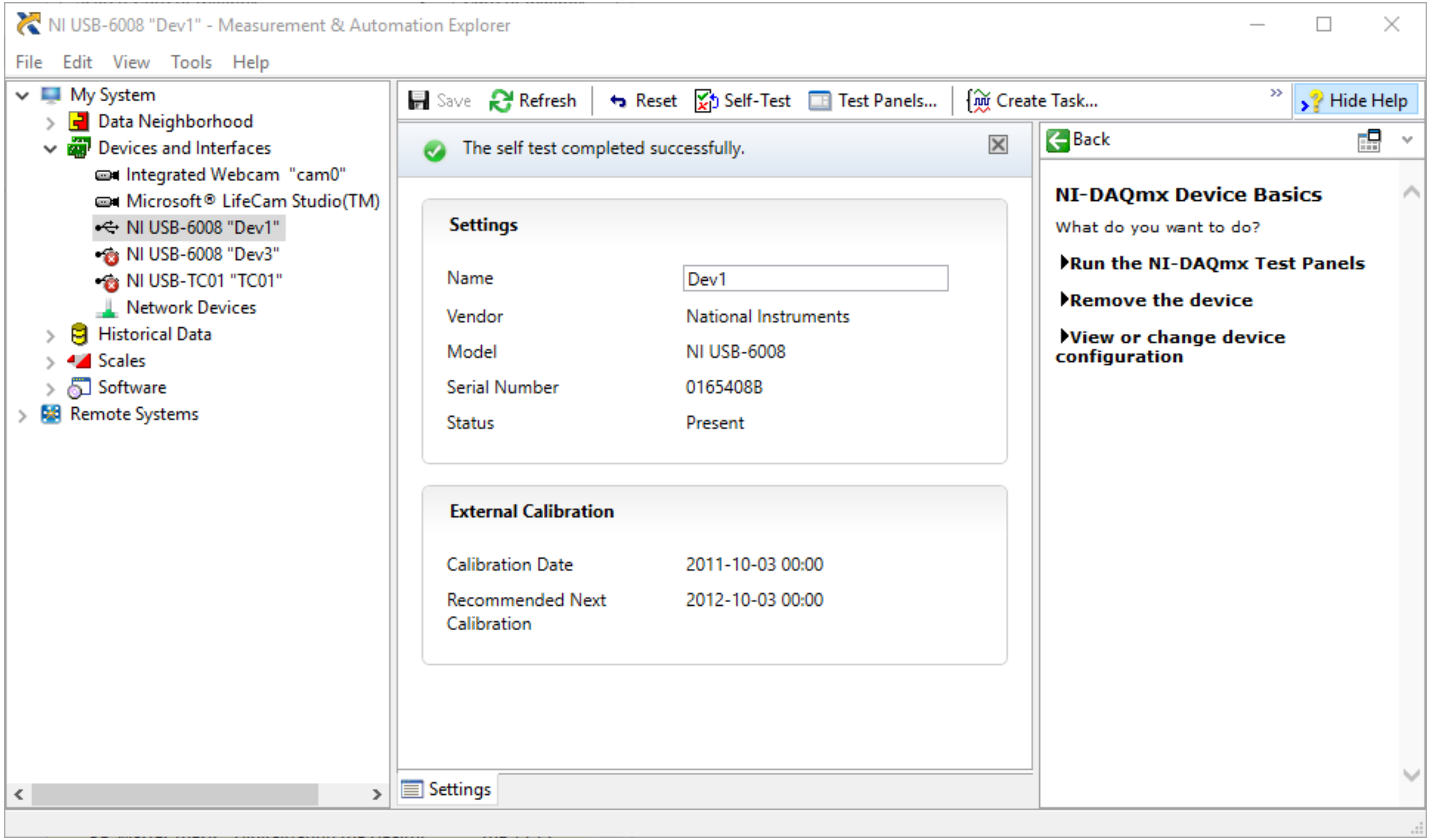
GND
AI 0 (AI 0+)
AI 4 (AI 0–)
GND
AI 1 (AI 1+)
AI 5 (AI 1–)
GND
AI 2 (AI 2+)
AI 6 (AI 2–)
GND
AI 3 (AI 3+)
AI 7 (AI 3–)
GND
AO 0
AO 1
GND
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
P0.0
P0.1
P0.2
P0.3
P0.4
P0.5
P0.6
P0.7
P1.0
P1.1
P1.2
P1.3
PFI 0
+2.5 V
+5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

https://www.halvorsen.blog

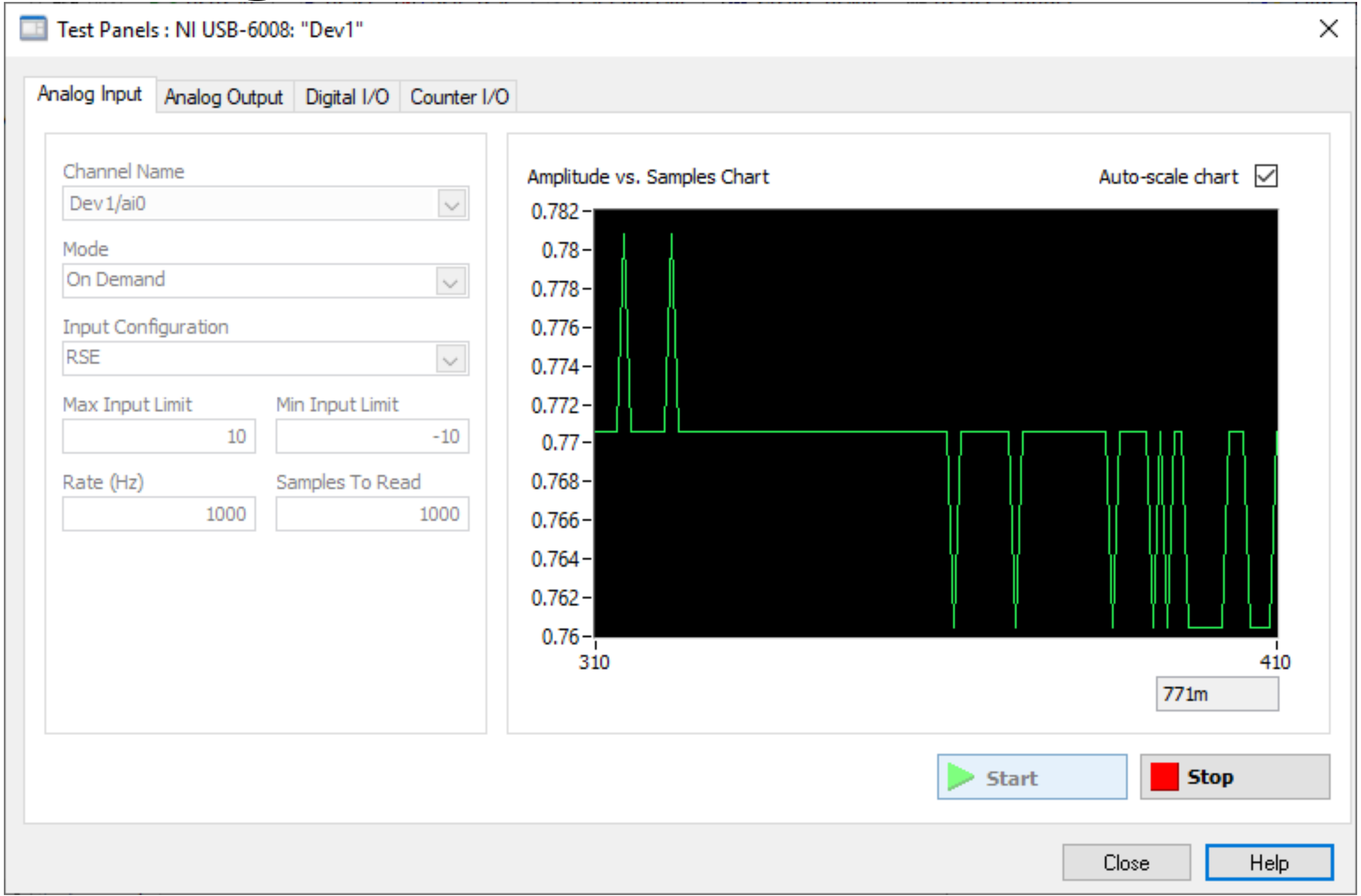
DAQmx

Hans-Petter Halvorsen

Measurement & Automation Explorer (MAX)



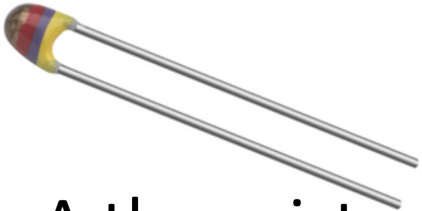
Using the Test Panel in MAX



https://www.halvorsen.blog

Thermistor 10kΩ

Hans-Petter Halvorsen



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 an 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

$A = 0.001129148, B = 0.000234125 \text{ and } C = 8.76741E-08$

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:

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

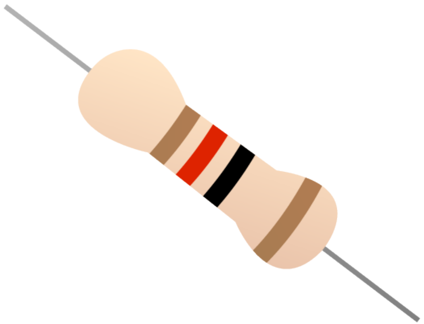
$$A = 0.001129148,$$
$$B = 0.000234125$$
$$C = 0.0000000876741$$

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

The Temperature in degrees Celsius will then be:

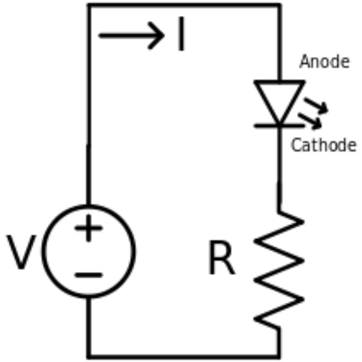
$$T_C = T_K - 273.15$$

Resistors



Resistance is measured in Ohm (Ω)

Resistors comes in many sizes, e.g., 220Ω , 270Ω, 330Ω, 1kΩm 10kΩ, ...



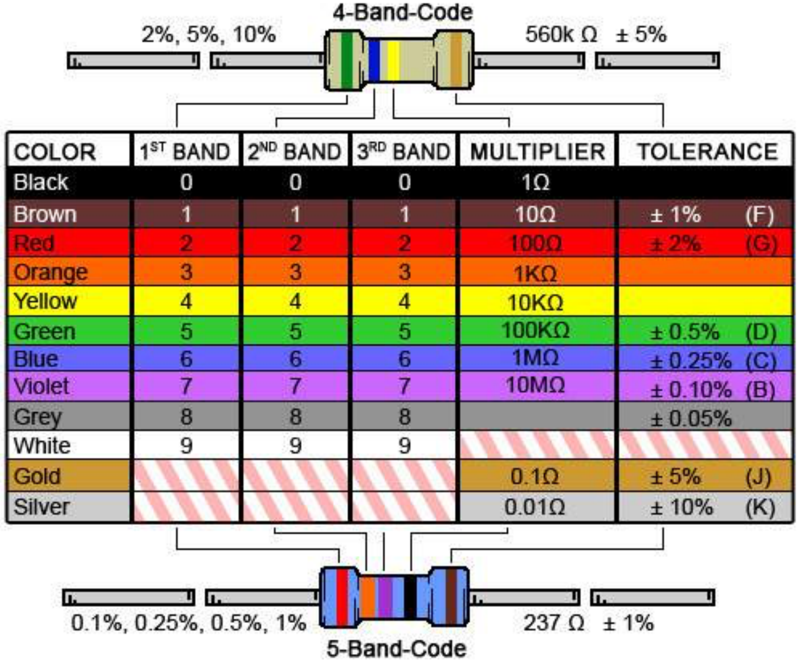
The resistance can be found using Ohms Law

$U = RI$

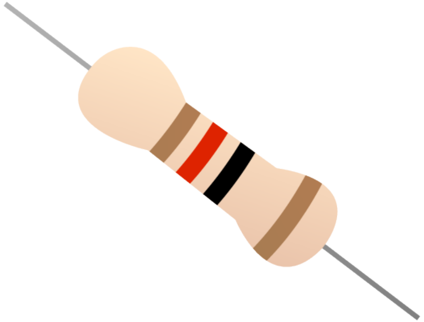
https://en.wikipedia.org/wiki/Resistor

Electrical symbol:

Resistor Color Codes

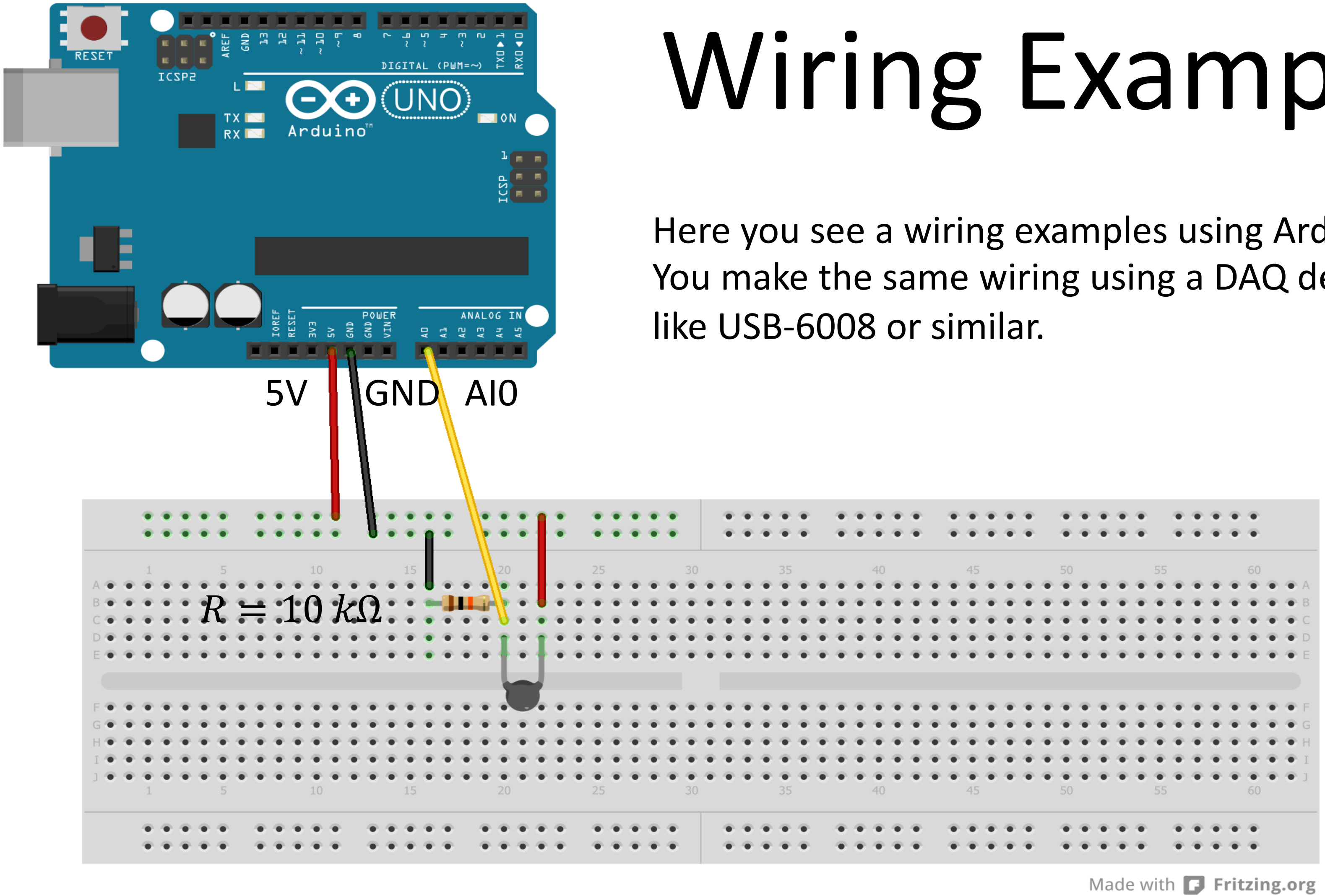


COLOR	1ST BAND	2ND BAND	3RD BAND	MULTIPLIER	TOLERANCE
Black	0	0	0	1Ω	
Brown	1	1	1	10Ω	± 1% (F)
Red	2	2	2	100Ω	± 2% (G)
Orange	3	3	3	1KΩ	
Yellow	4	4	4	10KΩ	
Green	5	5	5	100KΩ	± 0.5% (D)
Blue	6	6	6	1MΩ	± 0.25% (C)
Violet	7	7	7	10MΩ	± 0.10% (B)
Grey	8	8	8		± 0.05%
White	9	9	9		
Gold				0.1Ω	± 5% (J)
Silver				0.01Ω	± 10% (K)



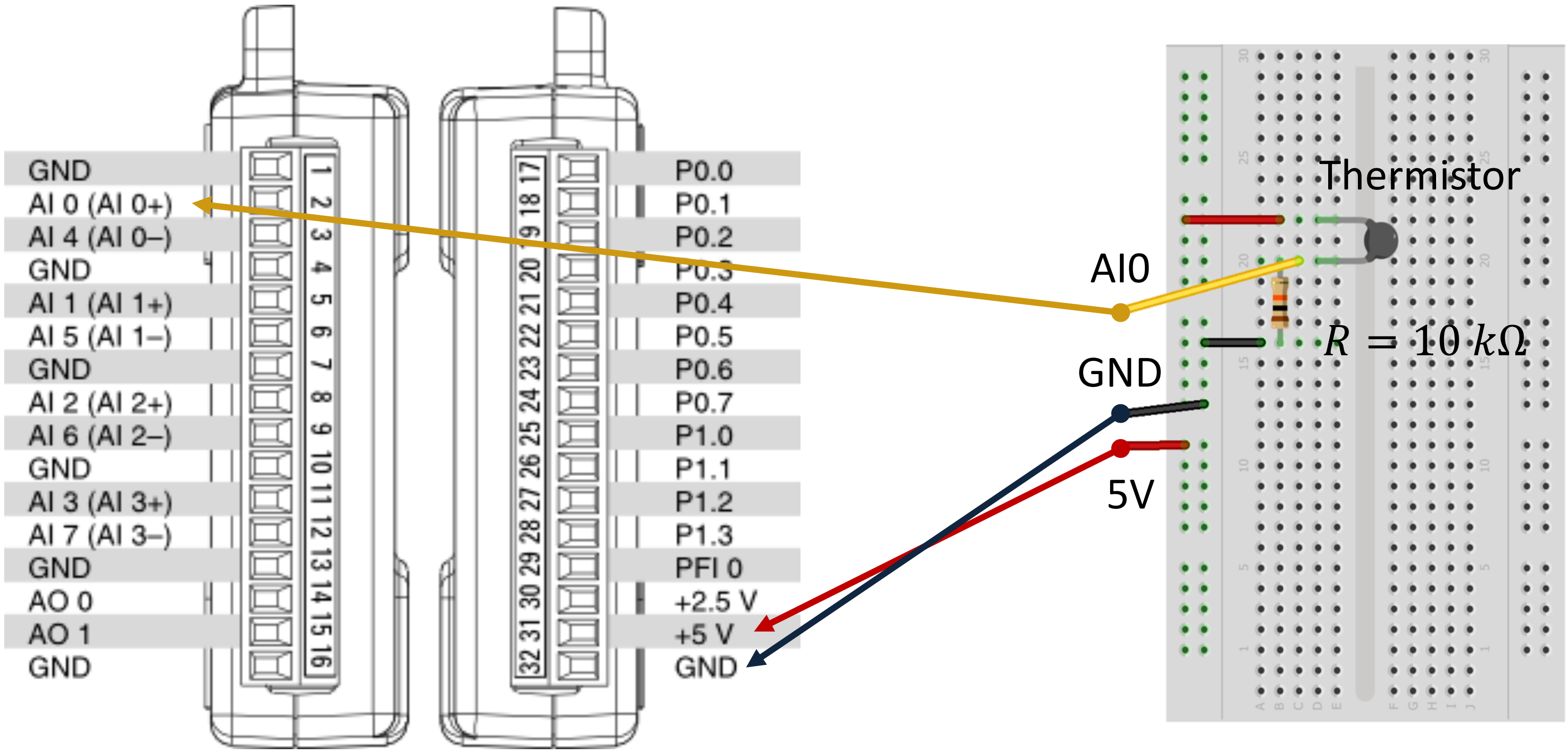
Resistor Calculator: http://www.allaboutcircuits.com/tools/resistor-color-code-calculator/

Wiring Example

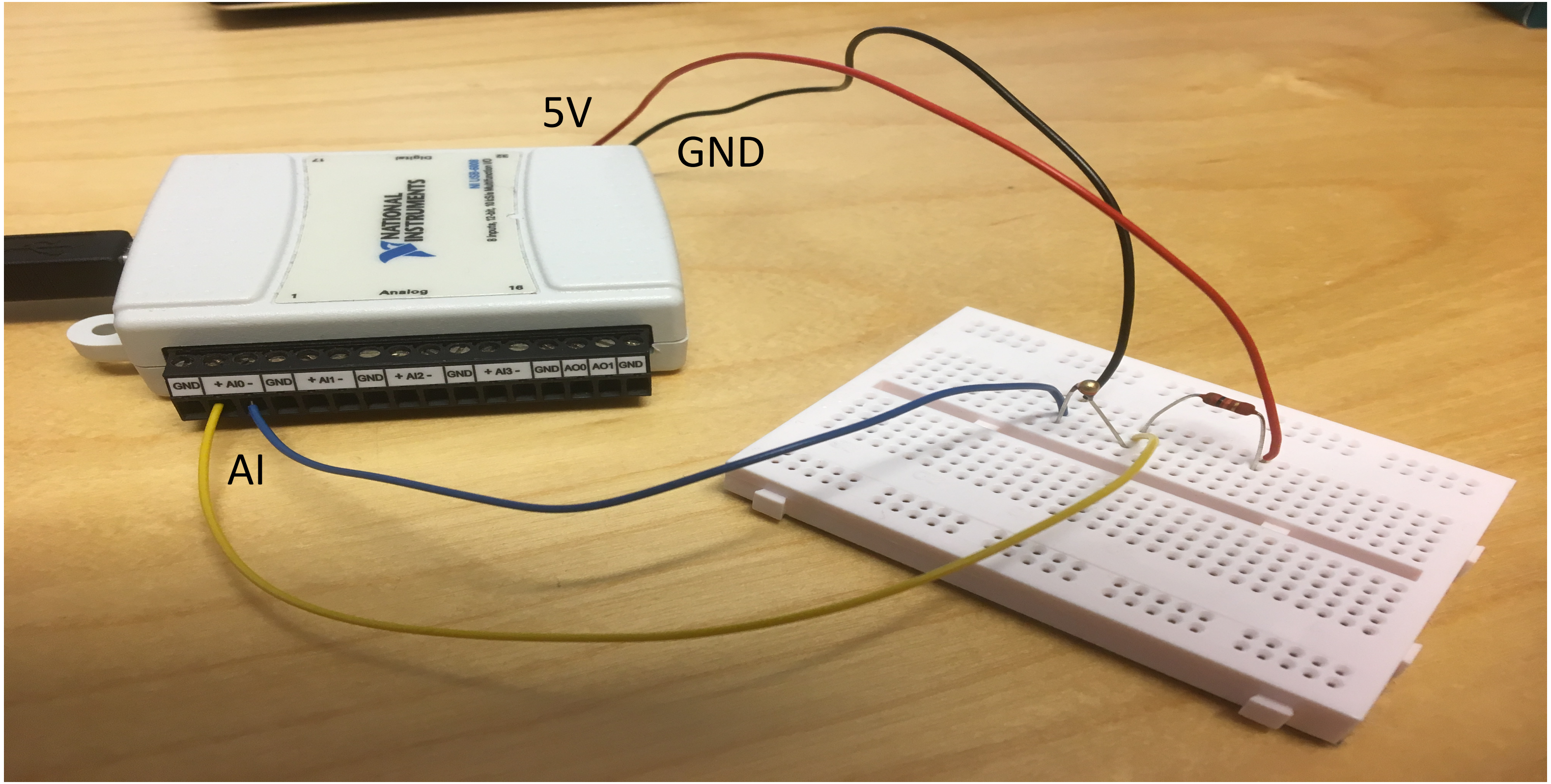


Here you see a wiring examples using Arduino. You make the same wiring using a DAQ device like USB-6008 or similar.

Wiring Example



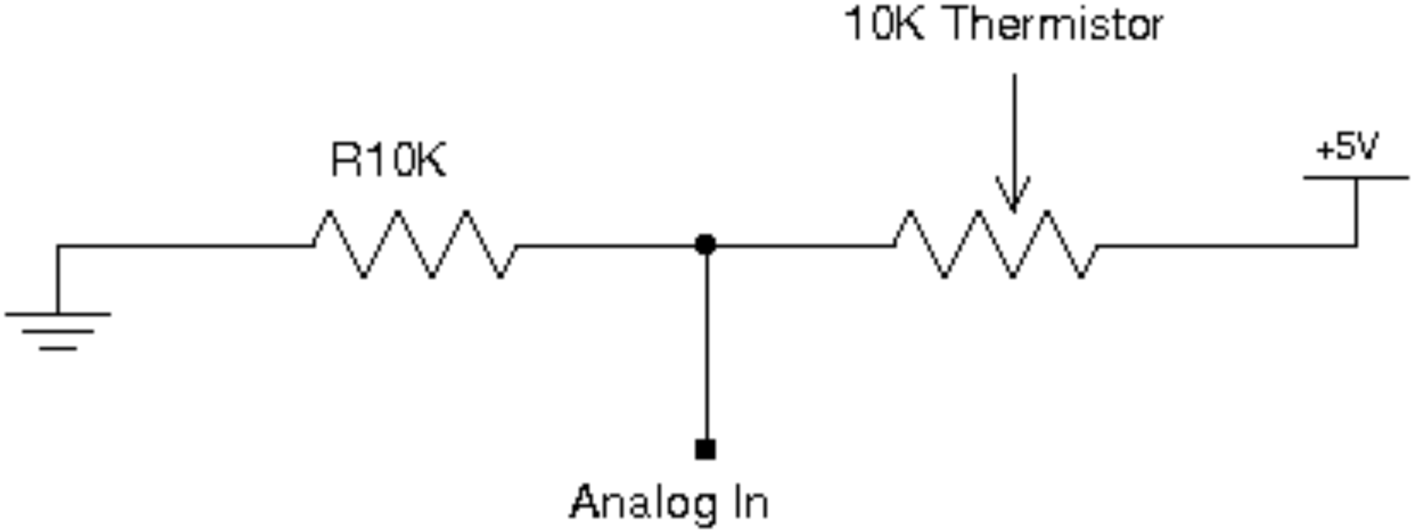
USB-6008 Wiring Example



We connect the sensor to LabVIEW using a USB DAQ Device from National Instruments, e.g., USB-6001, USB-6008 or similar. A breadboard has been used for the wiring.

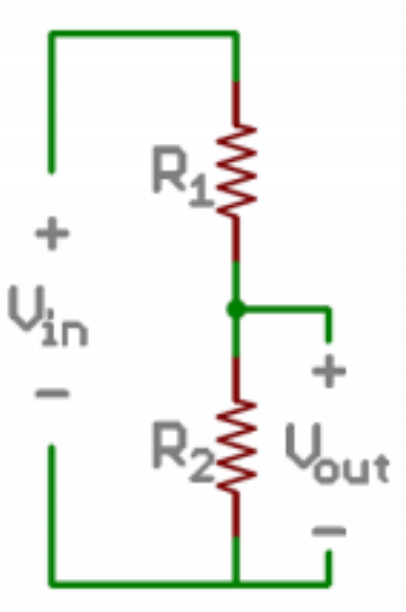
Wiring

The wiring is called a “Voltage divider”:



[https://en.wikipedia.org/wiki/Voltage_divider]

General Voltage Divider



$$V_{out} = V_{in} \cdot \frac{R_2}{R_1 + R_2}$$

https://learn.sparkfun.com/tutorials/voltage-dividers/all

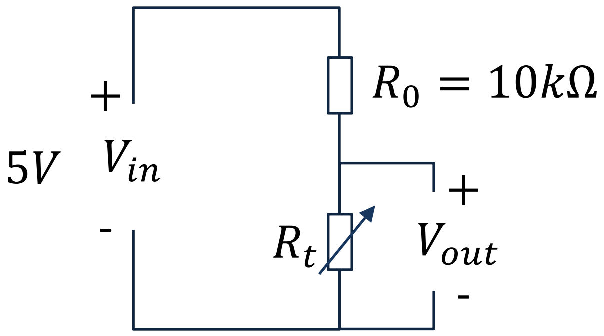
Voltage Divider for our system

Voltage Divider Equation:

$$V_{out} = V_{in} \frac{R_t}{R_0 + R_t}$$

We want to find R_t:

$$R_t = \frac{V_{out} R_0}{V_{in} - V_{out}}$$



R_t - 10k Thermistor. This varies with temperature. From Datasheet we know that $R_t = 10k\Omega$ @25℃

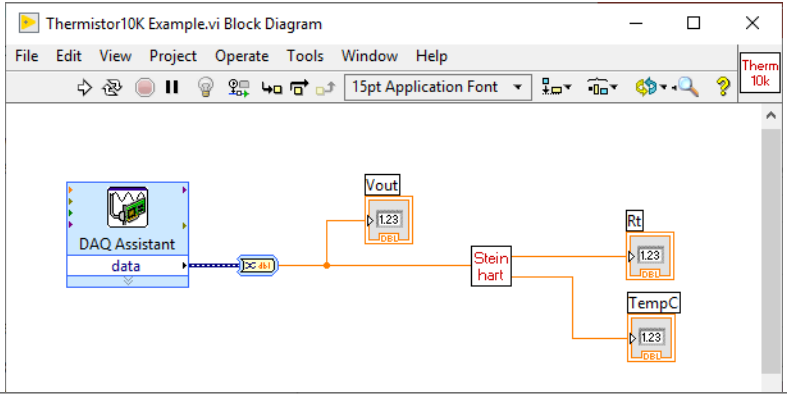
<u>Steps:</u>

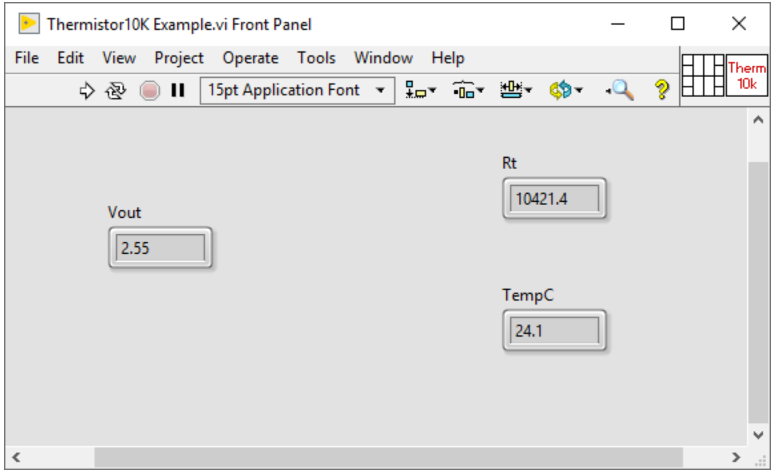
1. We wire the circuit on the Breadboard and connect it to the DAQ device
2. We measure V_{out} using the DAQ device
3. We calculate R_t using the Voltage Divider equation
4. Finally, we use Steinhart-Hart equation for finding the Temperature

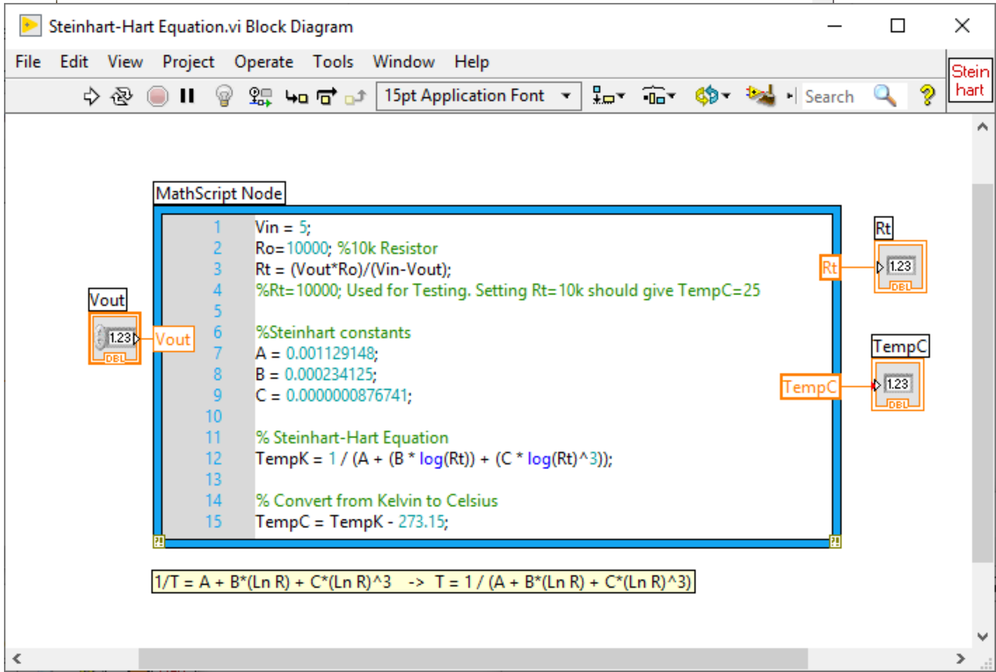
Code

1. Get V_{out} from the DAQ device
2. Calculate $R_t = \frac{V_{out} R_0}{V_{in} - V_{out}}$
3. Calculate $T_K = \frac{1}{A + B\, ln(R_t) + C(ln(R_t))^3}$
4. Calculate $T_C = T_K - 273.15$
5. Present T_C in the User Interface

LabVIEW Example







Arduino Example

```
const int temperaturePin = 0;

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int temperature = getTemp();
  Serial.print("Temperature Value: ");
  Serial.print(temperature);
  Serial.println("*C");
  delay(1000);
}

double getTemp()
{
  // Inputs ADC Value from Thermistor and outputs Temperature in Celsius

  int RawADC = analogRead(temperaturePin);
  long Resistance;
  double Temp;

  // Assuming a 10k Thermistor.  Calculation is actually: Resistance = (1024/ADC)
  Resistance=((10240000/RawADC) - 10000);

  // Utilizes the Steinhart-Hart Thermistor Equation:
  // Temperature in Kelvin = 1 / {A + B[ln(R)] + C[ln(R)]^3}
  // where A = 0.001129148, B = 0.000234125 and C = 8.76741E-08

  Temp = log(Resistance);
  Temp = 1 / (0.001129148 + (0.000234125 * Temp) + (0.0000000876741 * Temp * Temp * Temp));
  Temp = Temp - 273.15;  // Convert Kelvin to Celsius
  return Temp;  // Return the Temperature
}
```

Celsius to Fahrenheit Conversion

In Norway we typically use Celsius as temperature unit, while in US they use Fahrenheit.

Conversion between these are as follows:

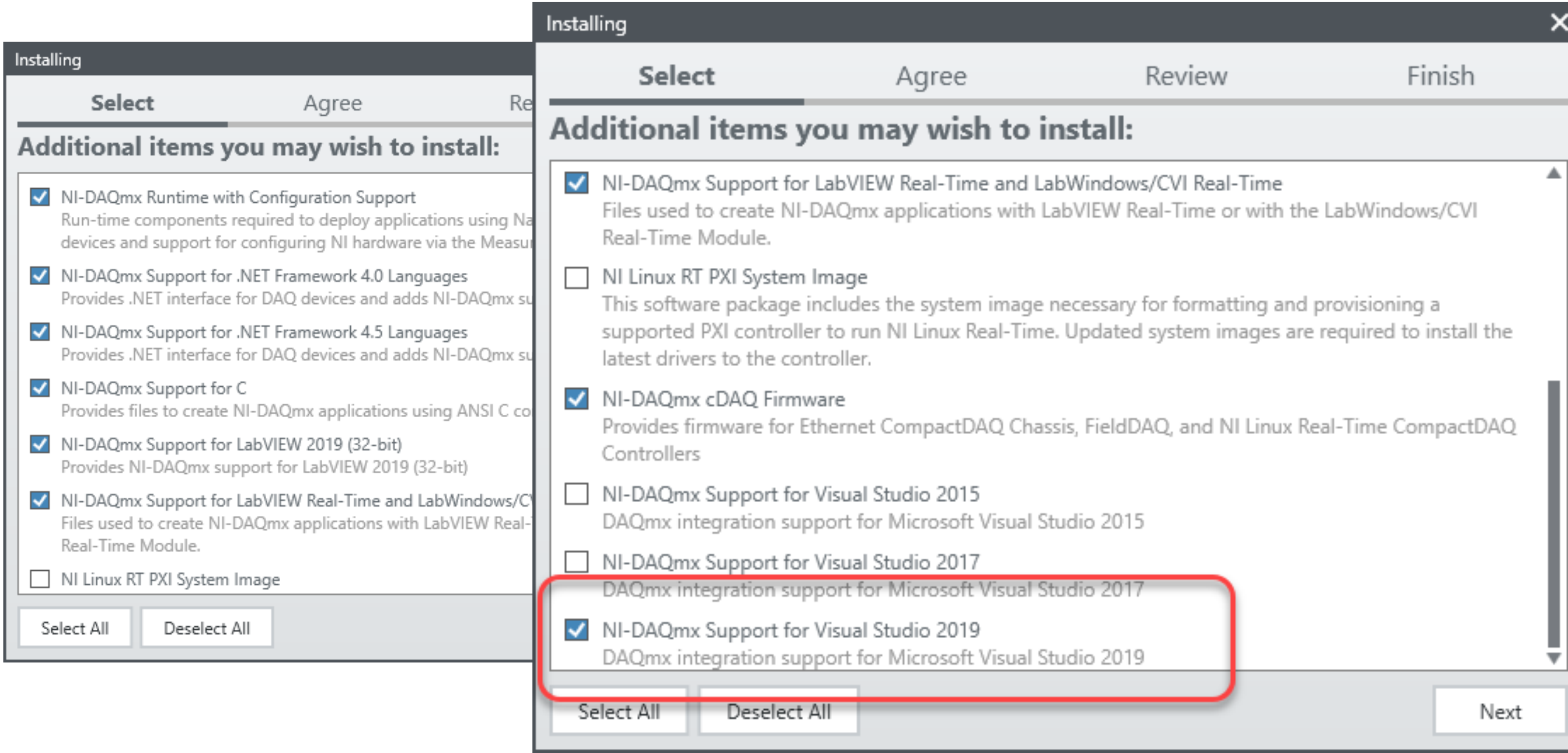
$$T_F = \frac{9}{5} T_C + 32$$

https://www.halvorsen.blog

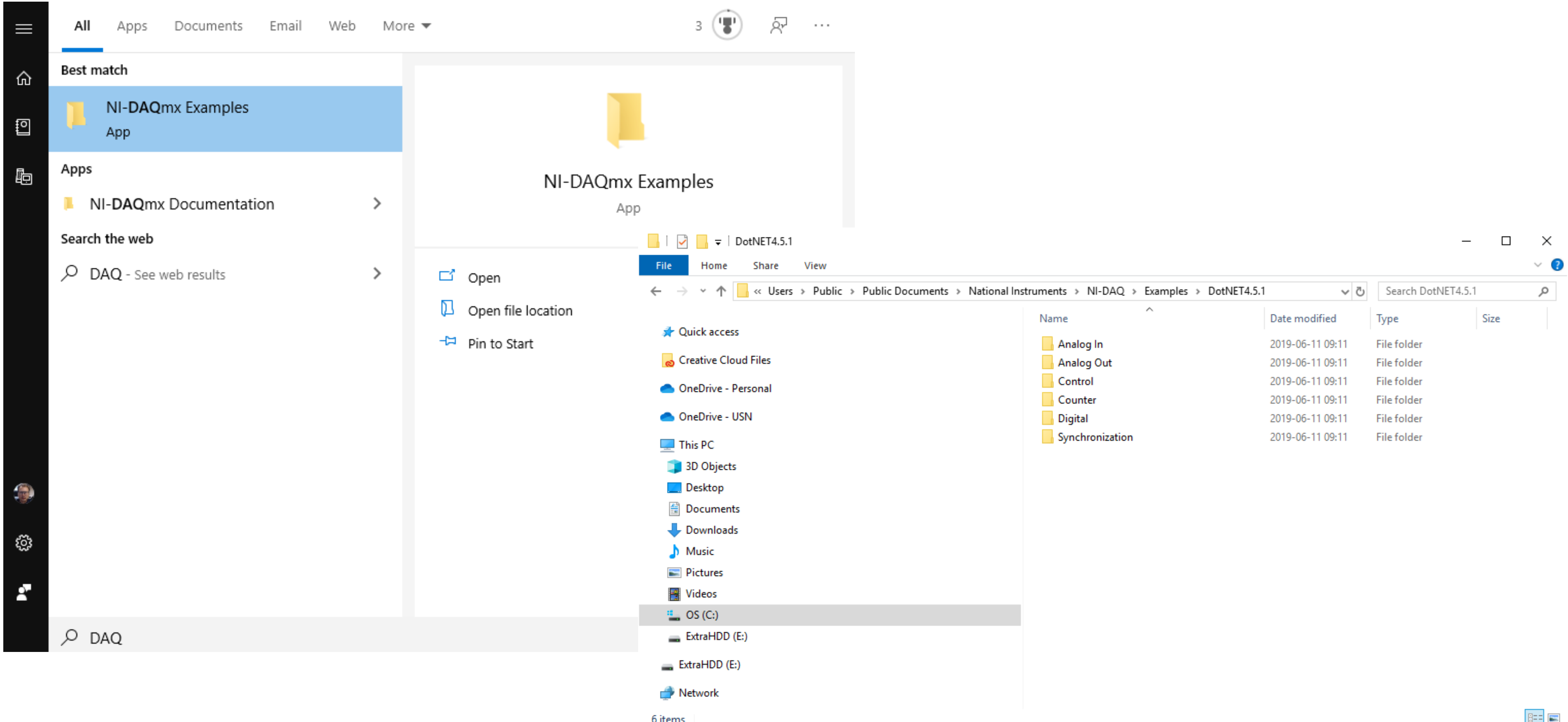
Visual Studio

Hans-Petter Halvorsen

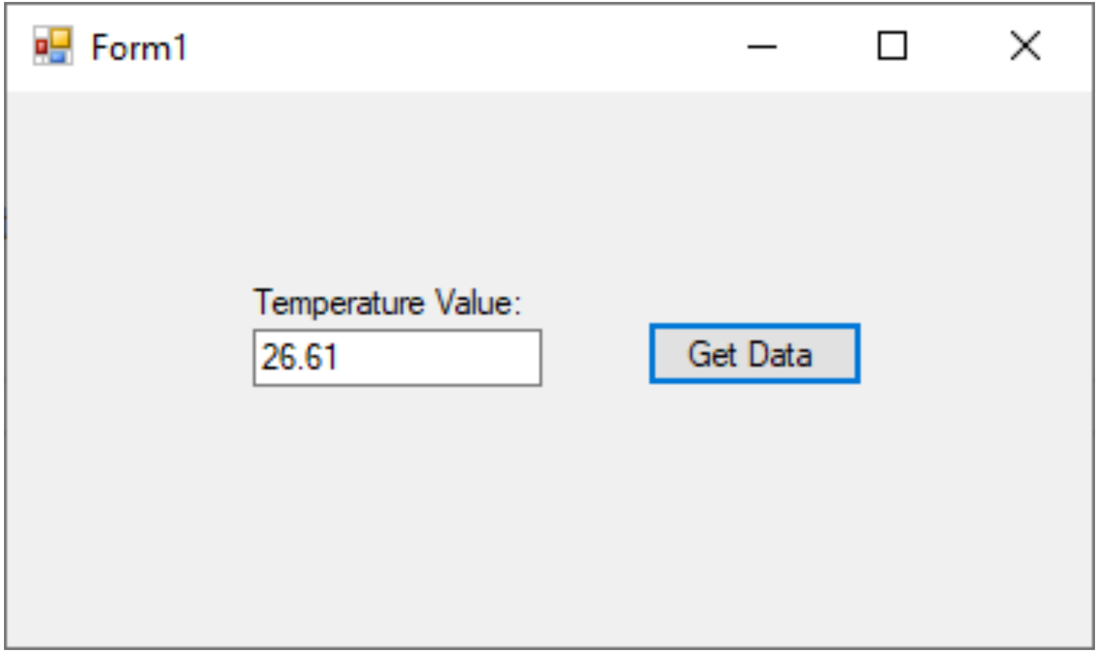
NI-DAQmx Driver

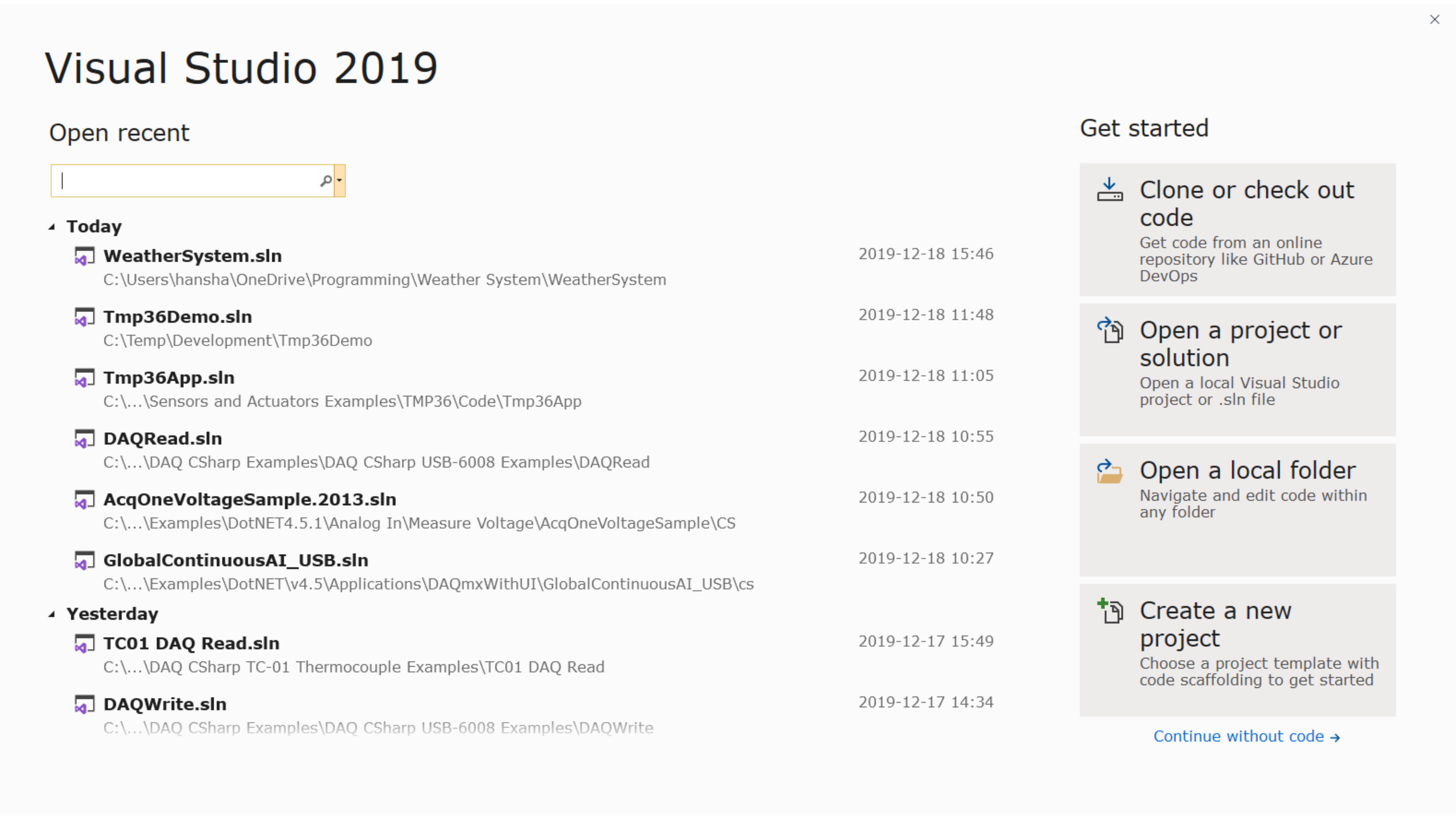


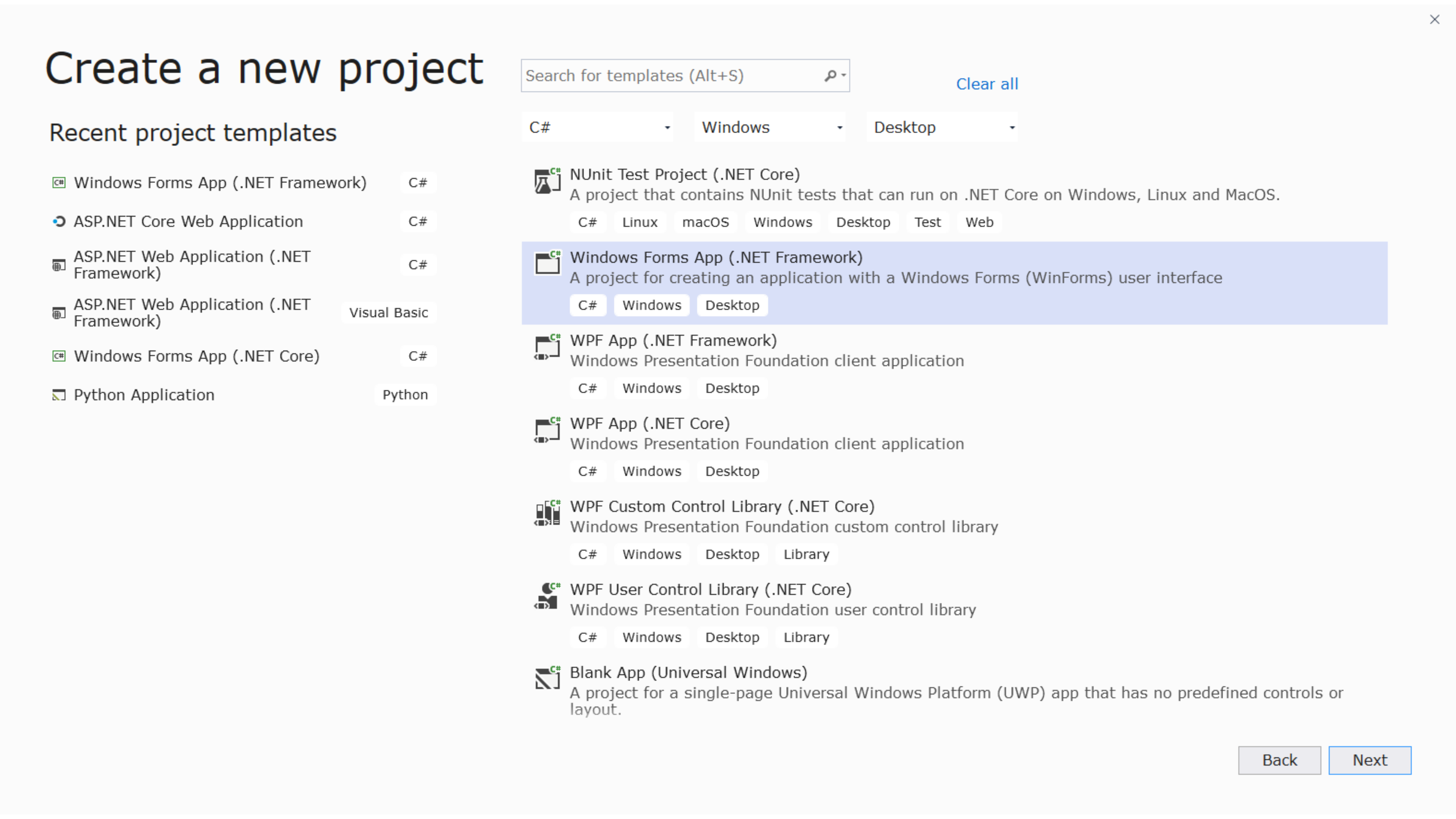
NI-DAQmx Examples

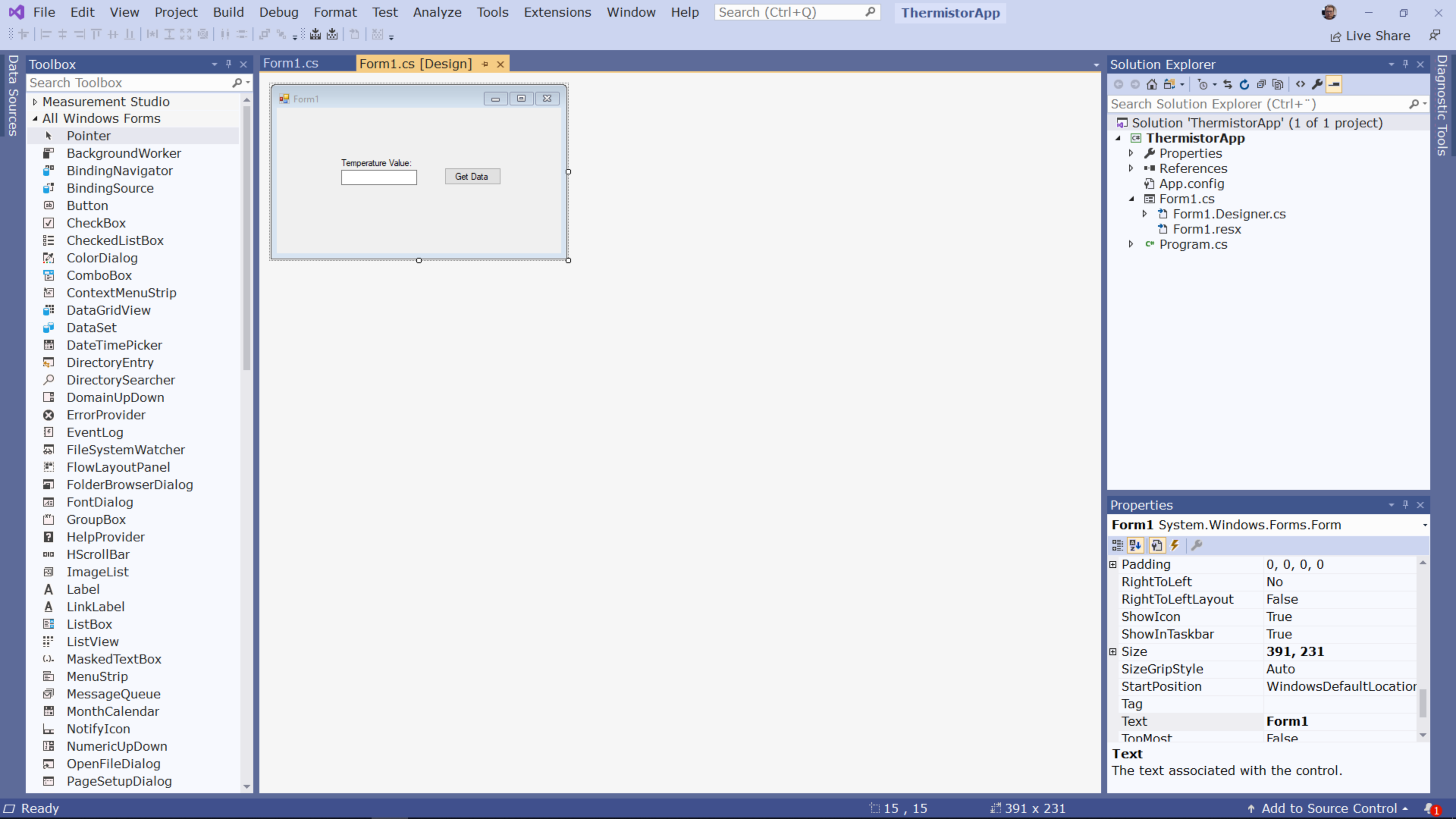


We will make the following Application:

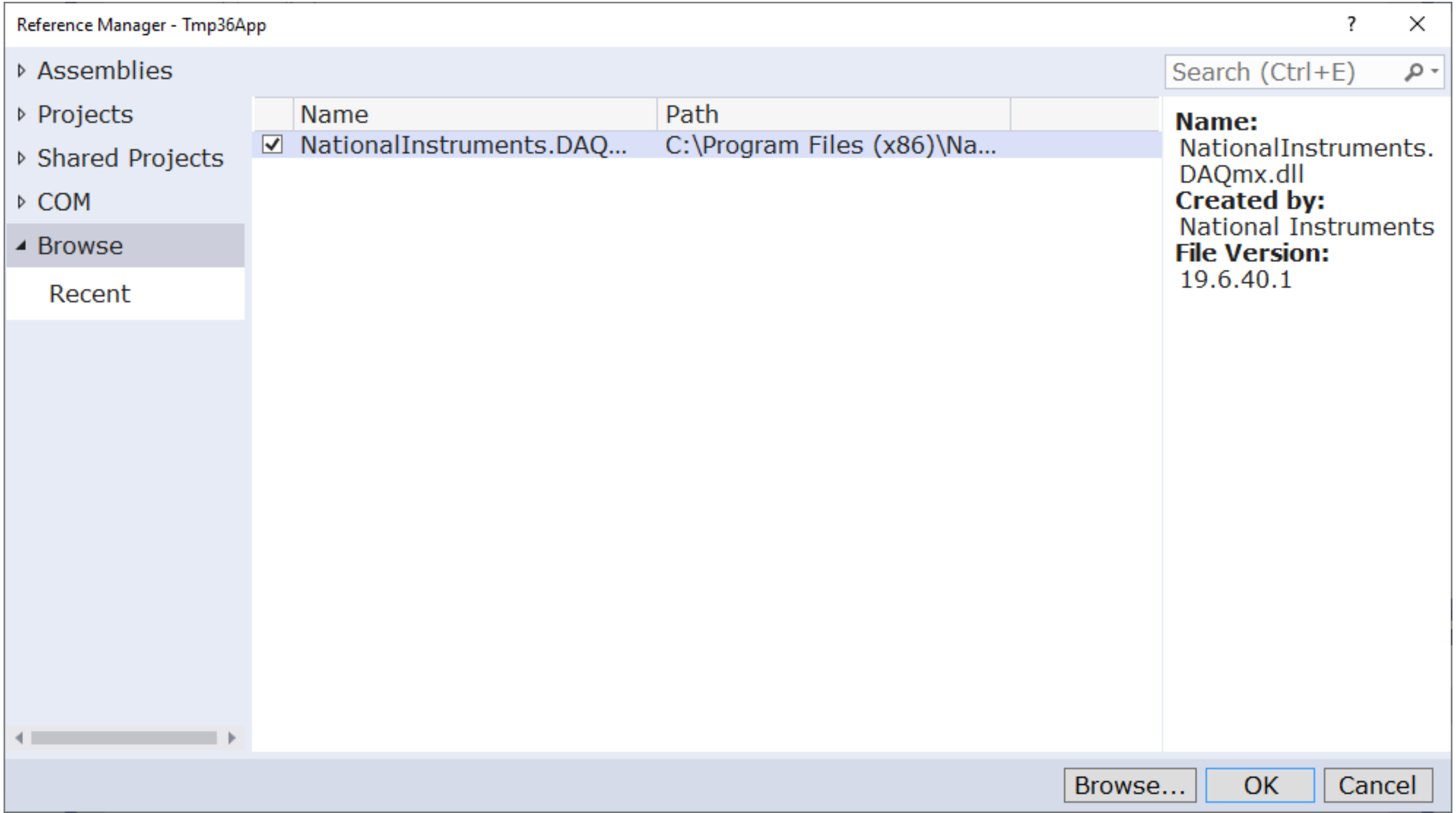


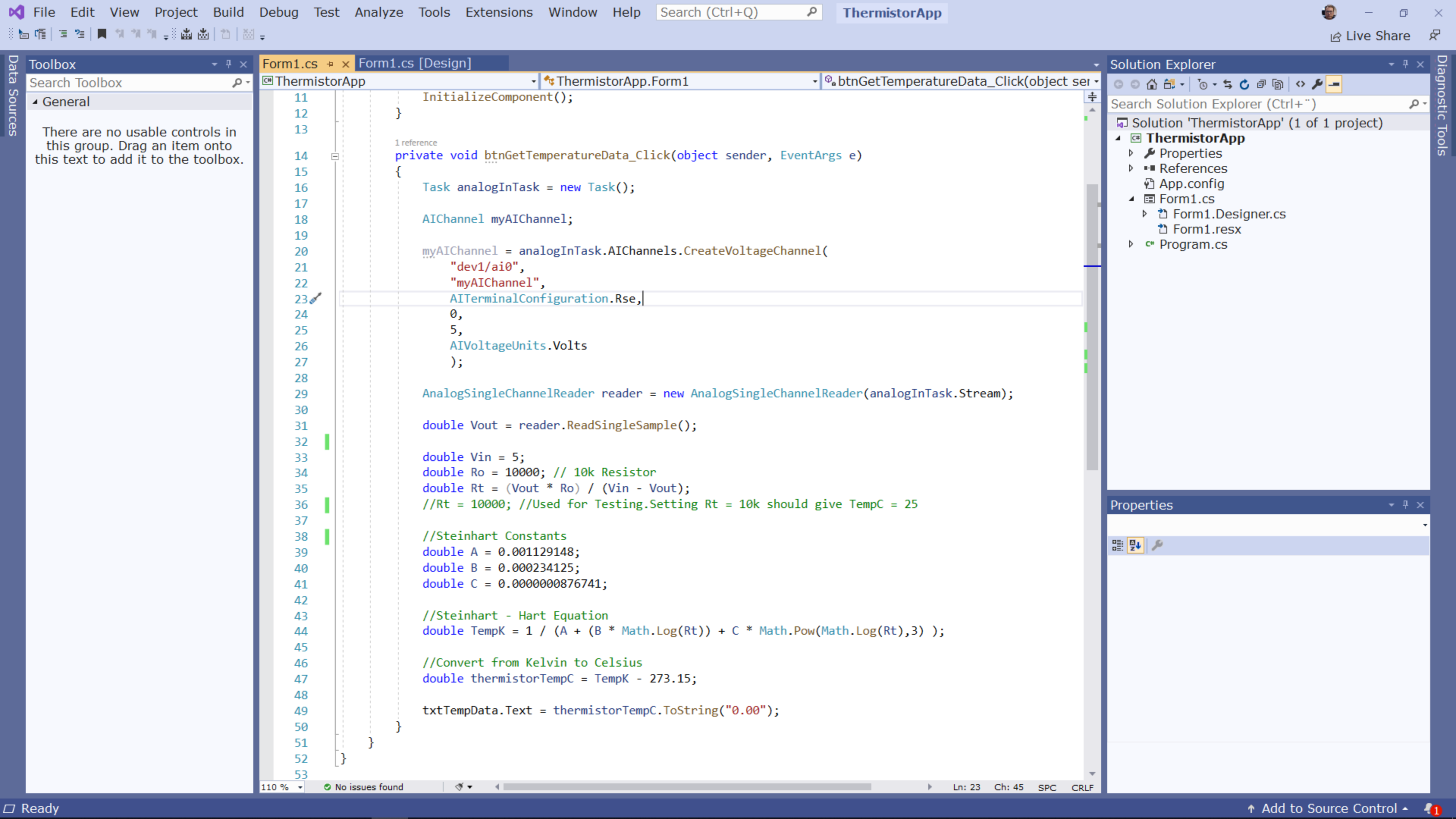

Visual Studio 2019
Open recent
Today
WeatherSystem.sln
C:\Users\hansha\OneDrive\Programming\Weather System\WeatherSystem
2019-12-18 15:46
Tmp36Demo.sln
C:\Temp\Development\Tmp36Demo
2019-12-18 11:48
Tmp36App.sln
C:\...\Sensors and Actuators Examples\TMP36\Code\Tmp36App
2019-12-18 11:05
DAQRead.sln
C:\...\DAQ CSharp Examples\DAQ CSharp USB-6008 Examples\DAQRead
2019-12-18 10:55
AcqOneVoltageSample.2013.sln
C:\...\Examples\DotNET4.5.1\Analog In\Measure Voltage\AcqOneVoltageSample\CS
2019-12-18 10:50
GlobalContinuousAI_USB.sln
C:\...\Examples\DotNET\v4.5\Applications\DAQmxWithUI\GlobalContinuousAI_USB\cs
2019-12-18 10:27
Yesterday
TC01 DAQ Read.sln
C:\...\DAQ CSharp TC-01 Thermocouple Examples\TC01 DAQ Read
2019-12-17 15:49
DAQWrite.sln
C:\...\DAQ CSharp Examples\DAQ CSharp USB-6008 Examples\DAQWrite
2019-12-17 14:34
Get started
Clone or check out code
Get code from an online repository like GitHub or Azure DevOps
Open a project or solution
Open a local Visual Studio project or .sln file
Open a local folder
Navigate and edit code within any folder
Create a new project
Choose a project template with code scaffolding to get started
Continue without code →


Create a new project
Search for templates (Alt+S)
Clear all
Recent project templates
Windows Forms App (.NET Framework) C#
ASP.NET Core Web Application C#
ASP.NET Web Application (.NET Framework) C#
ASP.NET Web Application (.NET Framework) Visual Basic
Windows Forms App (.NET Core) C#
Python Application Python
C#
Windows
Desktop
NUnit Test Project (.NET Core)
A project that contains NUnit tests that can run on .NET Core on Windows, Linux and MacOS.
C# Linux macOS Windows Desktop Test Web
Windows Forms App (.NET Framework)
A project for creating an application with a Windows Forms (WinForms) user interface
C# Windows Desktop
WPF App (.NET Framework)
Windows Presentation Foundation client application
C# Windows Desktop
WPF App (.NET Core)
Windows Presentation Foundation client application
C# Windows Desktop
WPF Custom Control Library (.NET Core)
Windows Presentation Foundation custom control library
C# Windows Desktop Library
WPF User Control Library (.NET Core)
Windows Presentation Foundation user control library
C# Windows Desktop Library
Blank App (Universal Windows)
A project for a single-page Universal Windows Platform (UWP) app that has no predefined controls or layout.
Back
Next


File Edit View Project Build Debug Format Test Analyze Tools Extensions Window Help
Search (Ctrl+Q)
ThermistorApp
Live Share
Toolbox
Search Toolbox
Measurement Studio
All Windows Forms
Pointer
BackgroundWorker
BindingNavigator
BindingSource
Button
CheckBox
CheckedListBox
ColorDialog
ComboBox
ContextMenuStrip
DataGridView
DataSet
DateTimePicker
DirectoryEntry
DirectorySearcher
DomainUpDown
ErrorProvider
EventLog
FileSystemWatcher
FlowLayoutPanel
FolderBrowserDialog
FontDialog
GroupBox
HelpProvider
HScrollBar
ImageList
Label
LinkLabel
ListBox
ListView
MaskedTextBox
MenuStrip
MessageQueue
MonthCalendar
NotifyIcon
NumericUpDown
OpenFileDialog
PageSetupDialog
Data Sources
Form1.cs
Form1.cs [Design]
Form1
Temperature Value:
Get Data
Solution Explorer
Search Solution Explorer (Ctrl+¨)
Solution 'ThermistorApp' (1 of 1 project)
ThermistorApp
Properties
References
App.config
Form1.cs
Form1.Designer.cs
Form1.resx
Program.cs
Diagnostic Tools
Properties
Form1 System.Windows.Forms.Form
Padding 0, 0, 0, 0
RightToLeft No
RightToLeftLayout False
ShowIcon True
ShowInTaskbar True
Size 391, 231
SizeGripStyle Auto
StartPosition WindowsDefaultLocation
Tag
Text Form1
TopMost False
Text
The text associated with the control.
Ready
15 , 15
391 x 231
Add to Source Control

NationalInstruments.DAQmx.dll




File Edit View Project Build Debug Test Analyze Tools Extensions Window Help
Search (Ctrl+Q)
ThermistorApp
Live Share
Toolbox
Search Toolbox
General
There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.
Data Sources
Form1.cs
Form1.cs [Design]
ThermistorApp
ThermistorApp.Form1
btnGetTemperatureData_Click(object ser
InitializeComponent();
}
1 reference
private void btnGetTemperatureData_Click(object sender, EventArgs e)
{
Task analogInTask = new Task();
AIChannel myAIChannel;
myAIChannel = analogInTask.AIChannels.CreateVoltageChannel(
"dev1/ai0",
"myAIChannel",
AITerminalConfiguration.Rse,
0,
5,
AIVoltageUnits.Volts
);
AnalogSingleChannelReader reader = new AnalogSingleChannelReader(analogInTask.Stream);
double Vout = reader.ReadSingleSample();
double Vin = 5;
double Ro = 10000; // 10k Resistor
double Rt = (Vout * Ro) / (Vin - Vout);
//Rt = 10000; //Used for Testing.Setting Rt = 10k should give TempC = 25
//Steinhart Constants
double A = 0.001129148;
double B = 0.000234125;
double C = 0.0000000876741;
//Steinhart - Hart Equation
double TempK = 1 / (A + (B * Math.Log(Rt)) + C * Math.Pow(Math.Log(Rt),3));
//Convert from Kelvin to Celsius
double thermistorTempC = TempK - 273.15;
txtTempData.Text = thermistorTempC.ToString("0.00");
}
}
}
110 %
No issues found
Ln: 23 Ch: 45 SPC CRLF
Solution Explorer
Search Solution Explorer (Ctrl+")
Solution 'ThermistorApp' (1 of 1 project)
ThermistorApp
Properties
References
App.config
Form1.cs
Form1.Designer.cs
Form1.resx
Program.cs
Properties
Diagnostic Tools
Ready
Add to Source Control

```
using System;
using System.Windows.Forms;
using NationalInstruments.DAQmx;

namespace ThermistorApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnGetTemperatureData_Click(object sender, EventArgs e)
        {
            Task analogInTask = new Task();

            AIChannel myAIChannel;

            myAIChannel = analogInTask.AIChannels.CreateVoltageChannel(
                "dev1/ai0",
                "myAIChannel",
                AITerminalConfiguration.Rse,
                0,
                5,
                AIVoltageUnits.Volts
                );

            AnalogSingleChannelReader reader = new AnalogSingleChannelReader(analogInTask.Stream);

            double Vout = reader.ReadSingleSample();

            double Vin = 5;
            double Ro = 10000; // 10k Resistor
            double Rt = (Vout * Ro) / (Vin - Vout);
            //Rt = 10000; //Used for Testing. Setting Rt = 10k should give TempC = 25

            //Steinhart Constants
            double A = 0.001129148;
            double B = 0.000234125;
            double C = 0.0000000876741;

            //Steinhart - Hart Equation
            double TempK = 1 / (A + (B * Math.Log(Rt)) + C * Math.Pow(Math.Log(Rt),3) );

            //Convert from Kelvin to Celsius
            double thermistorTempC = TempK - 273.15;

            txtTempData.Text = thermistorTempC.ToString("0.00");
        }
    }
}
```

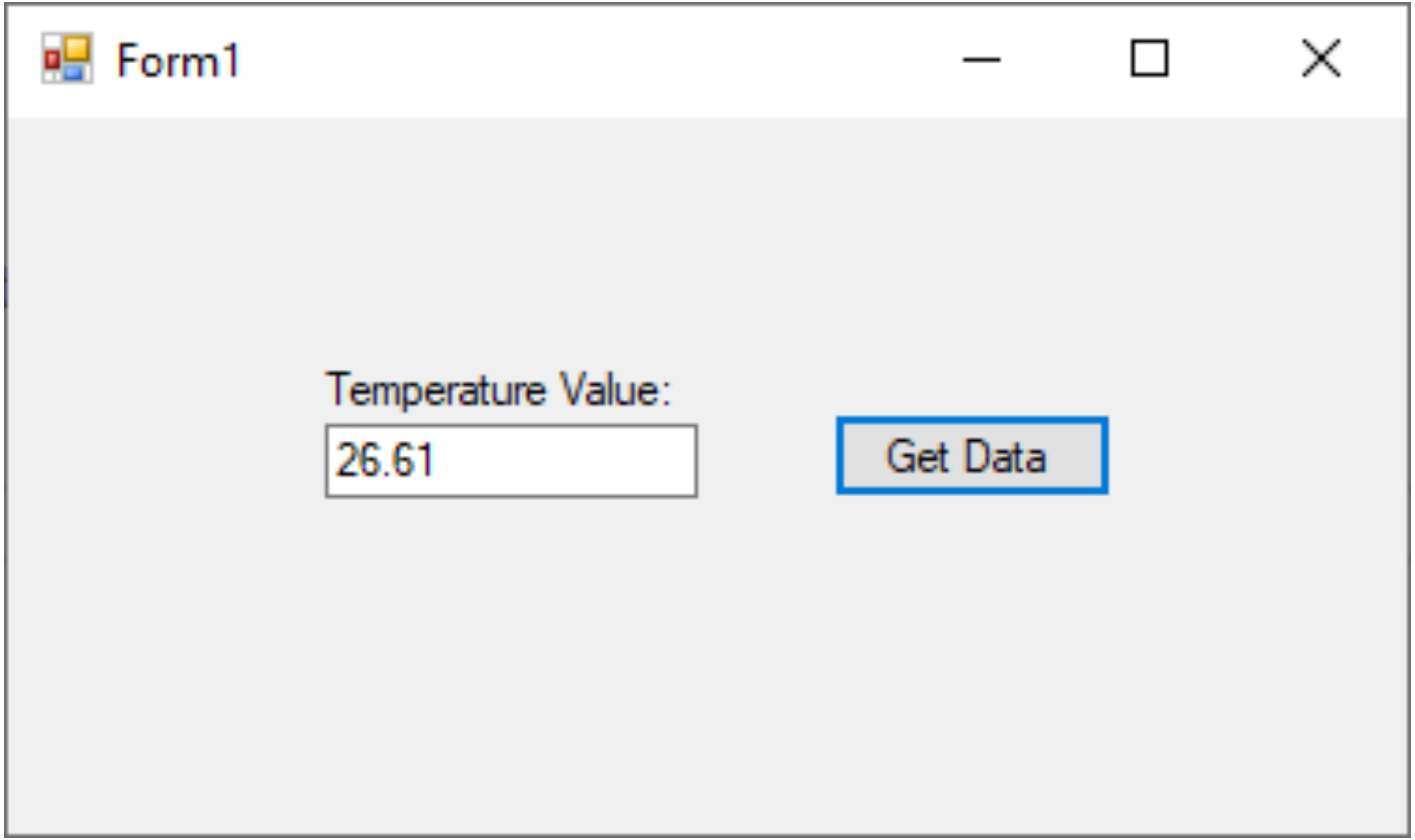
Read from DAQ

```
Task analogInTask = new Task();

AIChannel myAIChannel;

myAIChannel = analogInTask.AIChannels.CreateVoltageChannel(
    "dev1/ai0",
    "myAIChannel",
    AITerminalConfiguration.Rse,
    0,
    5,
    AIVoltageUnits.Volts
    );

AnalogSingleChannelReader reader = new
     AnalogSingleChannelReader(analogInTask.Stream);

double Vout = reader.ReadSingleSample();
```

Calculate Temperature Value in degrees Celsius

```
…
    double Vout = reader.ReadSingleSample();

    double Vin = 5;
    double Ro = 10000; // 10k Resistor
    double Rt = (Vout * Ro) / (Vin - Vout);
    //Rt = 10000; //Used for Testing. Setting Rt = 10k should give TempC = 25

    //Steinhart Constants
    double A = 0.001129148;
    double B = 0.000234125;
    double C = 0.0000000876741;

    //Steinhart - Hart Equation
    double TempK = 1 / (A + (B * Math.Log(Rt)) + C * Math.Pow(Math.Log(Rt),3) );

    //Convert from Kelvin to Celsius
    double thermistorTempC = TempK - 273.15;

    txtTempData.Text = thermistorTempC.ToString("0.00");
…
```

Final Application



Improvements

- Create and use separate **Classes** and in general improve the C# code
- Use a **Timer** in order to read values at specific intervals
- Plot values in a **Chart**
- Save Data to a **Database**
- Save Data to a **Text File**
- etc.

Good luck with your Application

Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

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

Web: https://www.halvorsen.blog