TC-01 Thermocouple in LabVIEW
Contents

• Introduction
  – Introduction to TC-01 Thermocouple Device
  – Introduction to DAQ
  – Getting Started with TC-01

• Practical LabVIEW Examples
  – DAQ Assistant
  – Using “Low-level” DAQmx VIs
Introduction
NI DAQ Hardware Examples

TC-01 Thermocouple

USB-6001

myDAQ

NI-DAQmx Hardware Driver

USB-6008

cDAQ
NI USB TC-01 Thermocouple

Connect to PC

Connect Thermocouple Sensor
NI USB TC-01 Thermocouple

J-Type Exposed-Junction Thermocouple

J-Type Grounded Probe Thermocouple
## TC-01 Datasheet

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of channels</td>
<td>1</td>
</tr>
<tr>
<td>A/D resolution</td>
<td>20 bits</td>
</tr>
<tr>
<td>Input range</td>
<td>±731.25 mV</td>
</tr>
<tr>
<td>Common-mode range, Channel-to-USB ground</td>
<td>±30 V</td>
</tr>
<tr>
<td>Common-mode rejection ratio (0 to 60 Hz), Channel-to-USB ground</td>
<td>&gt;145 dB</td>
</tr>
<tr>
<td>Noise rejection (50/60 Hz)</td>
<td>&gt;80 dB</td>
</tr>
<tr>
<td>Temperature measurement ranges</td>
<td>Works over temperature ranges defined by NIST (J, K, R, S, T, N, E, and B thermocouple types; the E type has a maximum limit of 900 °C.)</td>
</tr>
<tr>
<td>Conversion time</td>
<td>250 ms</td>
</tr>
<tr>
<td>Sample rate</td>
<td>4 S/s, maximum, software-timed</td>
</tr>
<tr>
<td>Input bandwidth (-3 dB)</td>
<td>1 Hz</td>
</tr>
<tr>
<td>Differential input impedance</td>
<td>20 MΩ between isolated 3.3 V and ground</td>
</tr>
<tr>
<td>Input noise</td>
<td>2 μVpp</td>
</tr>
<tr>
<td>Open thermocouple bias voltage</td>
<td>3.3 V</td>
</tr>
<tr>
<td>Cold-junction compensation sensor accuracy, 0 °C to 65 °C</td>
<td>1.25 °C maximum, 0.6 °C typical</td>
</tr>
<tr>
<td>Cold-junction compensation sensor resolution</td>
<td>0.0625 °C typical</td>
</tr>
<tr>
<td>Overvoltage protection</td>
<td>30 V max between TO+ and TC-</td>
</tr>
</tbody>
</table>
A DAQ System consists of 4 parts:

• **Physical input/output signals, sensors** – In this case the Thermocouple Sensor/probe

• **DAQ device/hardware** – In this case the TC-01 device

• **Driver** software – In this case the DAQmx software

• **Your software Application (Application Software)** - in this case your LabVIEW application
DAQ System

Input/Output Signals

- Analog Signals
- Digital Signals
- Sensors (Analog/Digital Interface)

Data Acquisition Hardware

- Analog IO
- Digital IO
- USB, etc.

PC

Software

- Application
- Hardware Driver
A computer can only deal with discrete signals.

You typically log data at specific intervals.

The sampling time ($T_s$) is the time between 2 logged values.

$t = \text{Continuous Time}$

$k = \text{Discrete Time}$

$T_s = \text{Sampling Time}$
To use DAQ hardware in LabVIEW we need to use the DAQmx driver. It can be downloaded for free.

You can use MAX to test and configure your DAQ device.

Measurement and Automation Explorer (MAX) is installed as part of the DAQmx software.

MAX – Measurement and Automation Explorer
DAQmx in LabVIEW

This is the DAQmx palette that appears in LabVIEW after installation.

To use DAQ hardware in LabVIEW we need to use the DAQmx driver. It can be downloaded for free.

Getting Started with TC-01
Getting Started with TC-01
Practical LabVIEW Examples
DAQ Assistant

TC-01 Thermocouple
DAQ Assistant
The TC-01 device has only one Analog Input called ai0

Select CJC Source = “Built-in”
Convert from Dynamic Data
While Loop

Typically, you log Data inside a While Loop
Number of Decimals and Units

Make sure to select a proper **Number of Decimals**. Typically, a Temperature Sensor don’t have an Accuracy with 10 Decimals. Make sure to read the Datasheet for the selected Sensor.

Always make sure to add a proper **Unit**. A Value without a Unit makes no Sense!
Using “Low-level” DAQmx VIs

TC-01 Thermocouple

Hans-Petter Halvorsen
Using “Low-level” DAQmx VIs
In the previous example we just used the default setting. If you need to change some of the default setting, just right-click on the select input and create a constant.
Configure Additional Settings
While Loop

TC-01 Low-level DAQ3.vi Block Diagram

While Loop
Increase speed by using “Start Task” and “Stop Task” VIs outside the While Loop
All Applications should have proper Error Handling

Further Improvements: Use the **State Machine** programming principle in 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