

SRTM Board Monitoring GUI

WinCC OA Setup and Configuration Guide

Prepared by: Nathan Herling
M.S. Candidate, Data Science
University of Arizona
nth@arizona.edu

Assisted by: Claude (Anthropic)
February 2026

Table of Contents

| Section | Page |
|------------------------------------|------|
| 1. OPC UA Node Discovery Tools | 1 |
| 2. Creating a New WinCC OA Project | 2 |
| 3. OPC UA Driver Configuration | 4 |
| 4. Datapoint Configuration | 5 |
| 5. GUI Panel Creation | 6 |
| 6. Key File Locations | 10 |
| 7. Quick Reference Commands | 10 |

Overview

This guide documents the process of creating a monitoring GUI for the SRTM (Slow Rate Timing Module) board using WinCC OA. The SRTM board exposes sensor data via an OPC UA server, which WinCC OA connects to for real-time visualization.

| Component | Details |
|------------------|---------------------------------|
| SRTM Board | IP: 192.168.0.117 |
| OPC UA Server | opc.tcp://192.168.0.117:4841 |
| WinCC OA Server | epp-bigmem3.physics.arizona.edu |
| WinCC OA Version | 3.19 |
| Project Name | N_SRTM_test02 |

1. OPC UA Node Discovery Tools

Before configuring WinCC OA, it's useful to browse the available OPC UA nodes on the SRTM board. Two Python scripts are available for this purpose:

| Script | Output | Description |
|------------------------------|-----------------|--------------------------|
| browse_opcua.py | opcua_nodes.csv | CSV format for analysis |
| browse_opcua_to_file_text.py | opcua_nodes.txt | Formatted text tree view |

Location: /home/naherlin/ on epp-bigmem3

Usage:

```
/usr/bin/python3 browse_opcua.py
```

Exploratory Script: Reading a Single Node

Before configuring WinCC OA, you can verify OPC UA connectivity using this simple script that reads the FPGA temperature directly:

File: read_fpga_temp.py

Location: /home/naherlin/

```
#!/usr/bin/python3
from opcua import Client
import time

OPCUA_URL = "opc.tcp://192.168.0.117:4841"
NODE_ID = "ns=2;i=SRTM.FPGA_temp"
PING_TIME = 10 # seconds

client = Client(OPCUA_URL)
```

```
try:  
    client.connect()  
    node = client.get_node(NODE_ID)  
    for i in range(PING_TIME):  
        value = node.get_value()  
        ts = time.strftime('%Y-%m-%d %H:%M:%S')  
        print(f'{ts} | FPGA_temp = {value}')  
        time.sleep(1)  
    except Exception as e:  
        print("ERROR:", e)  
    finally:  
        client.disconnect()
```

Usage:

```
/usr/bin/python3 read_fpga_temp.py
```

Tip: This script confirms that the OPC UA server is accessible and the node path (ns=2;s=SRTM.FPGA_temp) is correct before configuring WinCC OA.

2. Creating a New WinCC OA Project

Starting the Project Administrator

Launch the Project Administrator from terminal:

```
startPA
```

Method A: Using the Project Wizard

1. Click the **New Project** button (blank document icon in toolbar)
2. Select **Distributed project**
3. Check 'I have read the SIMATIC WinCC OA Security Guideline'
4. Click **Next** and follow the wizard

Important: Use 'Distributed project' type, NOT 'Standard project'. Standard projects cause errors in later configuration steps. Datapoints will have a `dist_1:` prefix.

Note: The wizard may fail with 'Error creating project!' In that case, use Method B below.

Method B: Copying an Existing Project (Recommended)

If the wizard fails, copy an existing working project and update the configuration:

Step 1: Copy the project directory

```
cd ~/WinCC_Projects  
cp -r N_SRTM_test02 SRTM_Monitor_v3  
cd SRTM_Monitor_v3
```

Step 2: Update config/config file

```
sed -i 's/N_SRTM_test02/SRTM_Monitor_v3/g' config/config
```

Step 3: Change pmonPort to avoid conflict

```
sed -i 's/pmonPort = 4999/pmonPort = 5001/g' config/config
```

Step 4: Clear old log files

```
rm -rf log/*
```

Step 5: Verify config/config

```
cat config/config
```

| Setting | Value |
|-------------|---|
| proj_path | /home/naherlin/WinCC_Projects/SRTM_Monitor_v3 |
| pmonPort | 5001 (unique port) |
| distributed | 1 |

Step 6: Stop the original project before starting

The copied project shares default port configurations with the original. You must stop the original project first:

```
pkill -u $USER -f 'OriginalProjectName'  
# Verify processes stopped:  
ps aux | grep -i wincc | grep -v grep
```

Step 7: Refresh startPA — the new project should appear in the list. Start it.

Note: If you need to run both projects simultaneously, add unique ports to config/config:

dataPort = 4898

eventPort = 4899

Otherwise, the default ports will conflict.

FYI: To check settings from an existing project:

```
cat ~/WinCC_Projects/<ProjectName>/config/config
```

Old log files may contain references to the original project name — this is harmless. Use `grep -r 'OldName' . 2>/dev/null` to verify no critical files are affected.

3. OPC UA Driver Configuration

If you copied an existing project (Method B): The OPC UA driver configuration is likely already set up.

Open **SysMgm** → **Driver OPC** → **OPC UA Client** to verify the connection shows 'Connected' and 'Running'. If so, skip to Section 4.

For New Projects

The OPC UA driver must be configured to run as driver number 2 (to avoid conflict with the simulation driver on number 1).

Edit config/progs file:

```
WCCOAopcua | always | 30 | 1 | 1 | -num 2 -host dist_1
```

Configuring the OPC UA Connection (GUI)

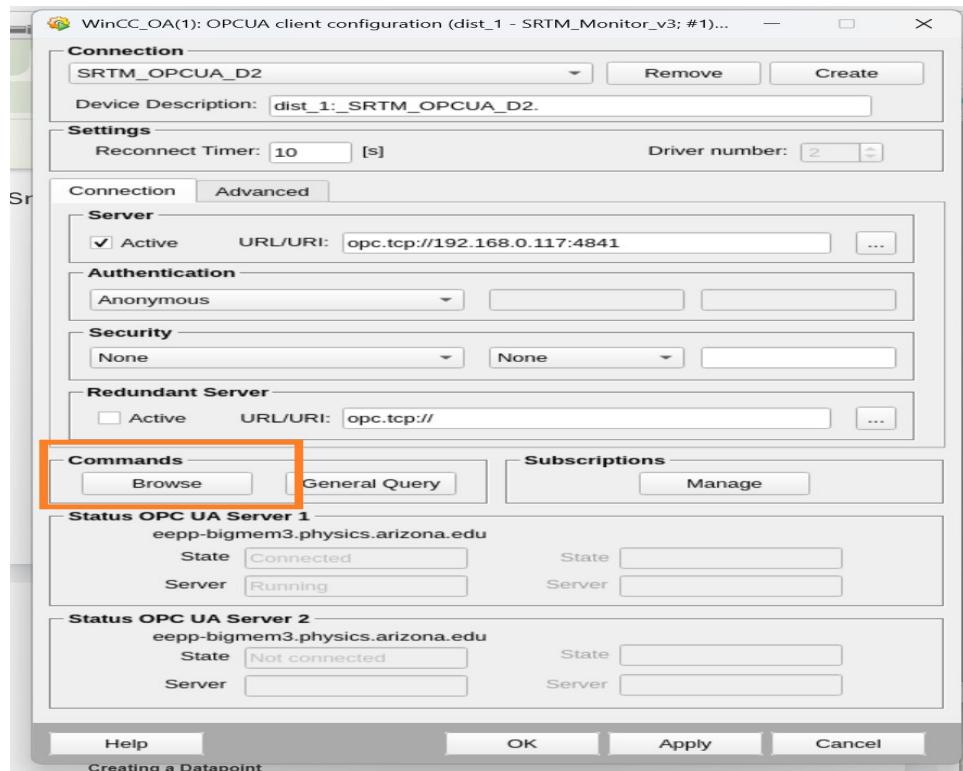
1. Open **SysMgm** → **Driver OPC** → **OPC UA Client**
2. Click **Create** to add a new connection (or select existing)
3. Configure the following settings:

| Setting | Value |
|-----------------|------------------------------|
| Connection Name | SRTM_OPCUA_D2 |
| Driver Number | 2 |
| Server URL | opc.tcp://192.168.0.117:4841 |
| Security Policy | None |

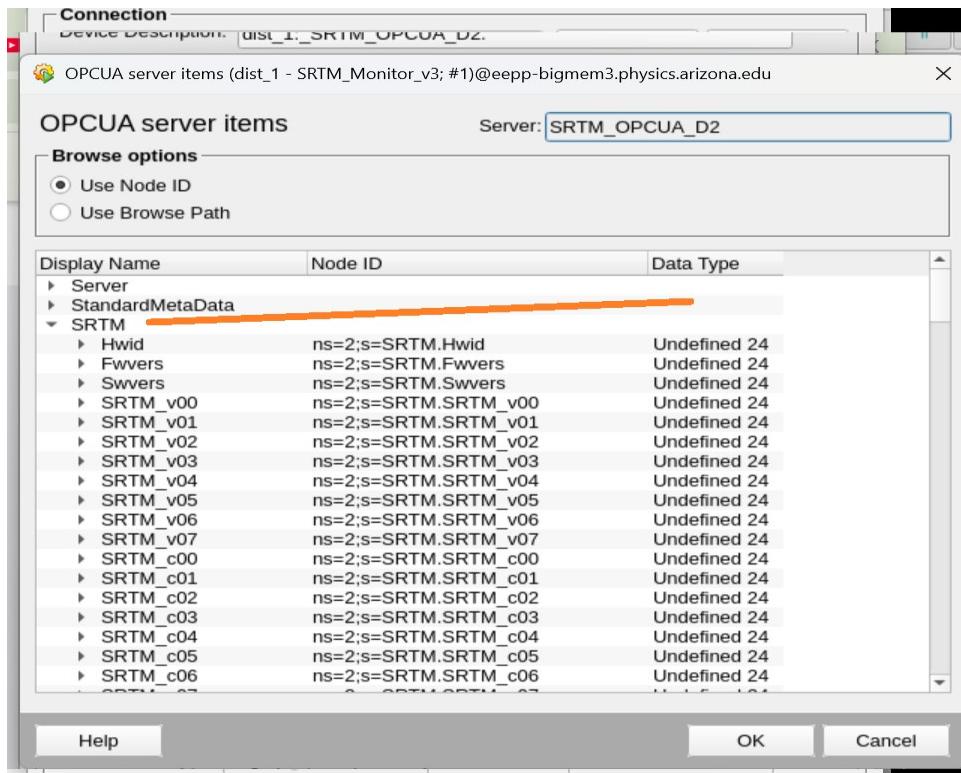
| | |
|----------------|-----------|
| Authentication | Anonymous |
|----------------|-----------|

4. Check **Active** checkbox
5. Click **Apply**
6. Verify **Status OPC UA Server 1** shows 'Connected' and 'Running'
7. Click **Browse** to verify you can see the SRTM nodes
8. Click **OK**

OPC UA Client Configuration Window:



Browse OPC UA Server Items:



4. Datapoint Configuration

Opening PARA (Datapoint Parameterization)

Open PARA via: **SysMgm** → **Database** → **Database Configuration**

If you copied an existing project (Method B): The datapoint **SRTM_FPGA_temp** likely already exists. In PARA, expand **dist_1** → **ExampleDP_Float** → **SRTM_FPGA_temp** and verify the Value column shows the current temperature (e.g., 35.967). If working, skip to Section 5.

Creating a New Datapoint

1. In PARA, right-click on **ExampleDP_Float** → Create new datapoint
2. Configure:

| Property | Value |
|----------|-----------------|
| Name | SRTM_FPGA_temp |
| Type | ExampleDP_Float |

Address Configuration (Periphery Address)

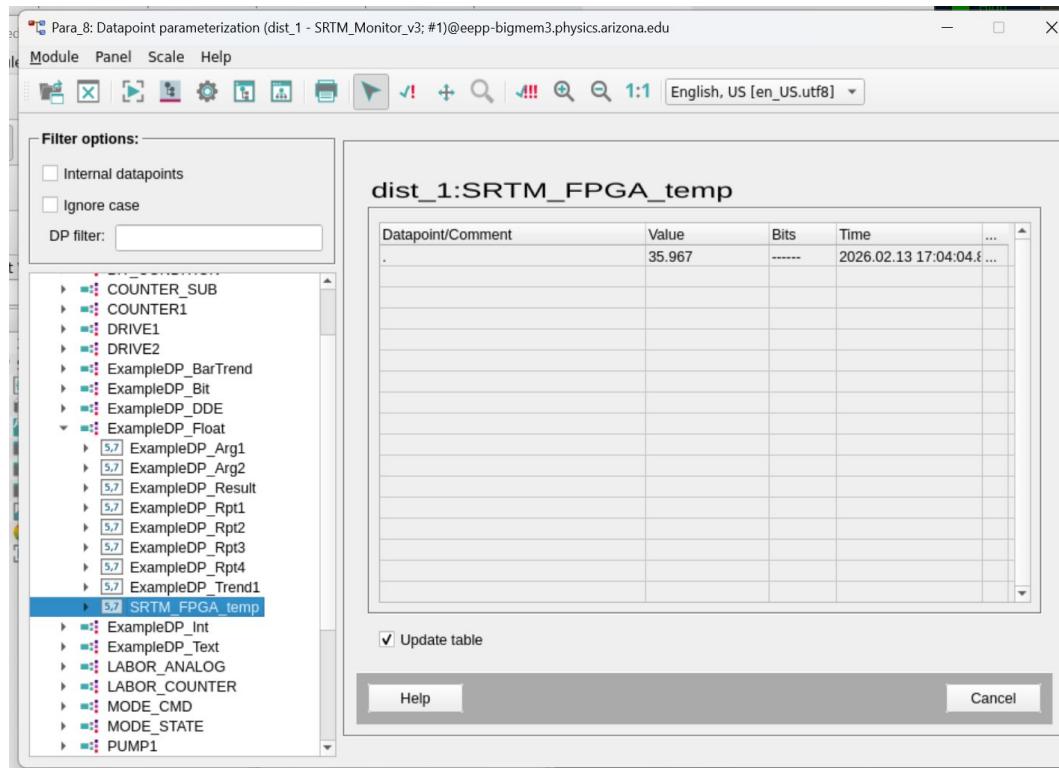
The address maps the datapoint to the OPC UA node on the SRTM board:

| Property | Value |
|----------|-------|
|----------|-------|

| | |
|---------------|--------------------------------------|
| Driver Number | 2 |
| Address Type | OPCUA |
| Reference | SRTM_OPCUA_D2\$ns=2;s=SRTM.FPGA_temp |
| Direction | Input (1) |

Tip: The OPC UA node address format is: ns=2;s=SRTM.<NodeName>

PARA Window showing SRTM_FPGA_temp datapoint with live value:



5. GUI Panel Creation

Opening the Graphics Editor (GEDI)

GEDI is typically already open when you start the project.

Or from terminal:

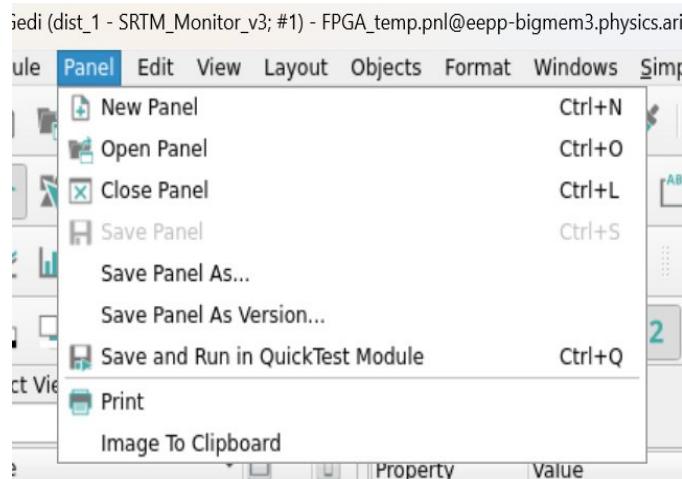
```
/opt/WinCC_OA/3.19/bin/WCCOagedi -PROJ SRTM_Monitor_v3 &
```

Creating a New Panel

1. **Panel → New Panel (Ctrl+N)**
2. Set panel size (e.g., 500 x 400) in the Property Editor
3. **Panel → Save Panel As...**
4. Save as: `FPGA_Monitor.pnl`

Location: `~/WinCC_Projects/SRTM_Monitor_v3/panels/`

Panel Menu:



Adding a Value Display

1. Select Text tool (T) from toolbar
2. Draw text box on panel
3. Select the text element
4. In Property Editor → Event → Initialize → click script button
5. Check 'Display value' and click OK
6. Select datapoint: `SRTM_FPGA_temp`

Adding an LCD Number Display

1. **Objects → More Objects → LCD Number**
2. Draw the LCD area on the panel
3. Select the LCD widget, then in Property Editor find **Event → Initialize**
4. Click the script button to open the Script Editor
5. Enter the following Initialize script:

LCD Initialize Script:

```

main()
{
    dpConnect( "updateLCD" , "dist_1:SRTM_FPGA_temp._online.._value" );
}

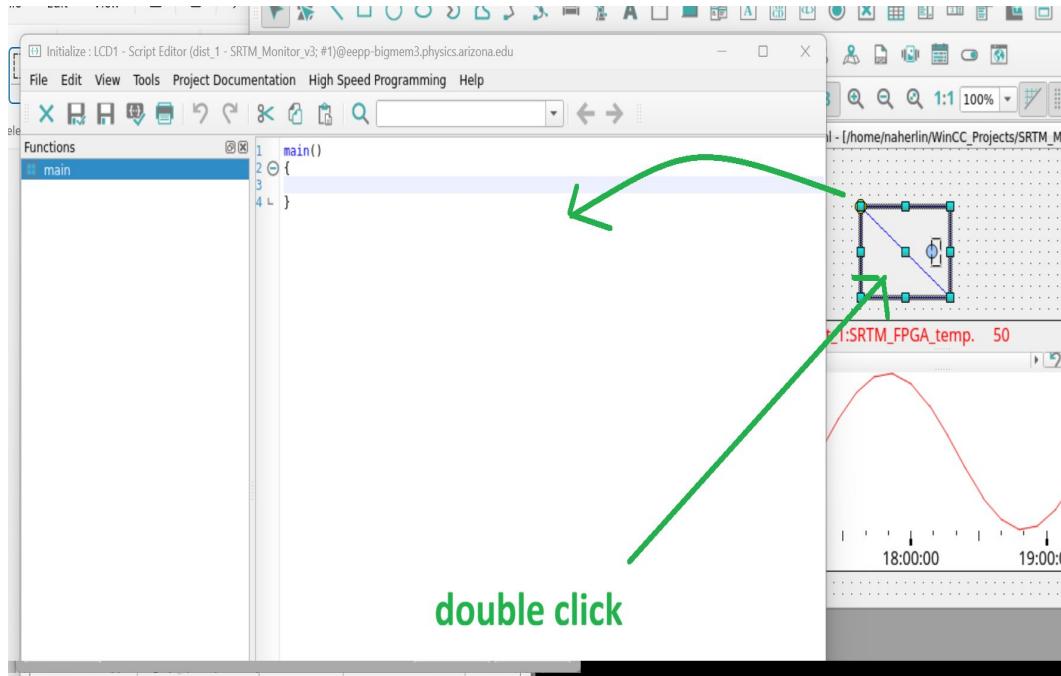
void updateLCD(string dp, float value)
{
    this.value = value;
}

```

6. Save the script (File → Save) and close the Script Editor

Note: Each widget requires a datapoint binding or script to display live data. Double-clicking a widget opens its configuration. The Script Editor (shown when clicking script icons) allows custom control logic using the `dpConnect` function.

Widget Script Editor and Live Panel:

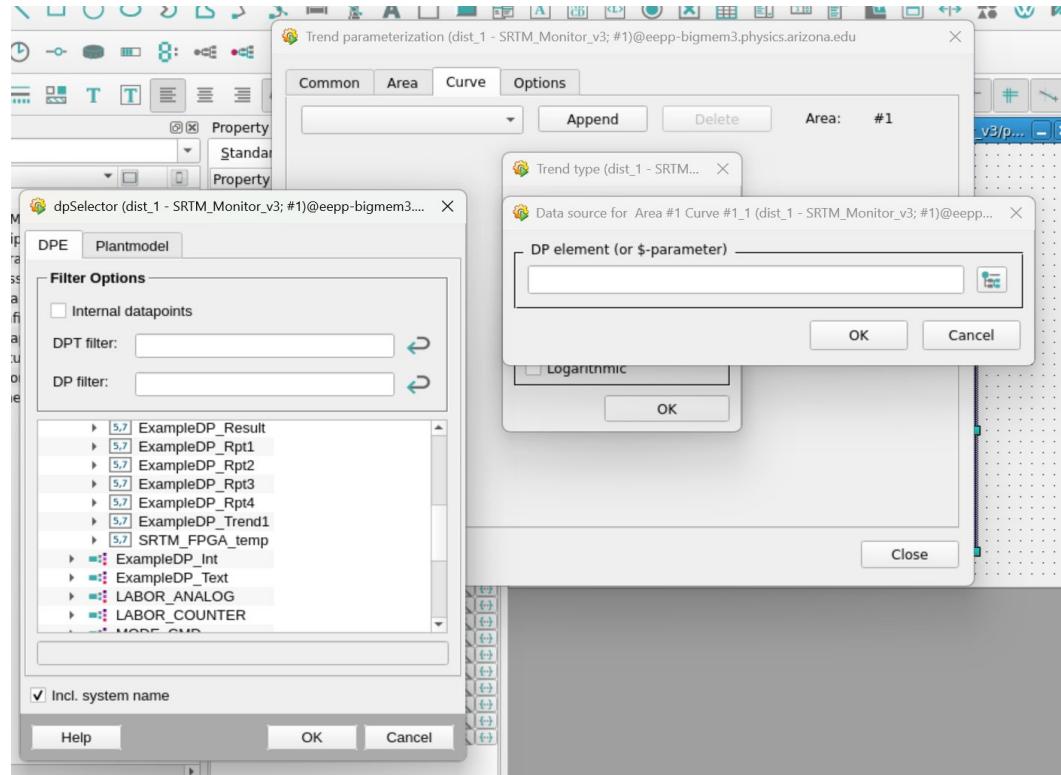


Adding a Trend (Time Series Chart)

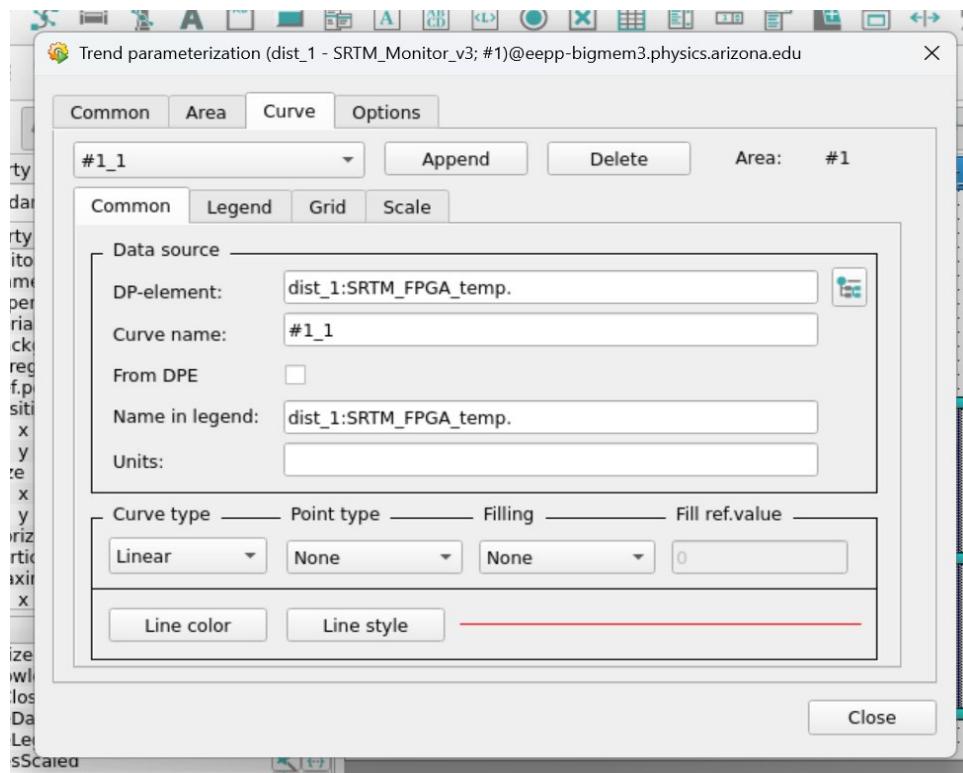
1. Objects → More Objects → Trend
2. Draw trend area on panel
3. Double-click to configure:

| Tab | Setting | Value |
|-----------|--------------------|----------------------------|
| Common | Display time range | 0 Days, 0 Hours, 5 Minutes |
| Curve | Datapoint | SRTM_FPGA_temp. |
| Scale (Y) | Auto scale | Checked (or set Min/Max) |

Selecting the Datapoint for the Trend:



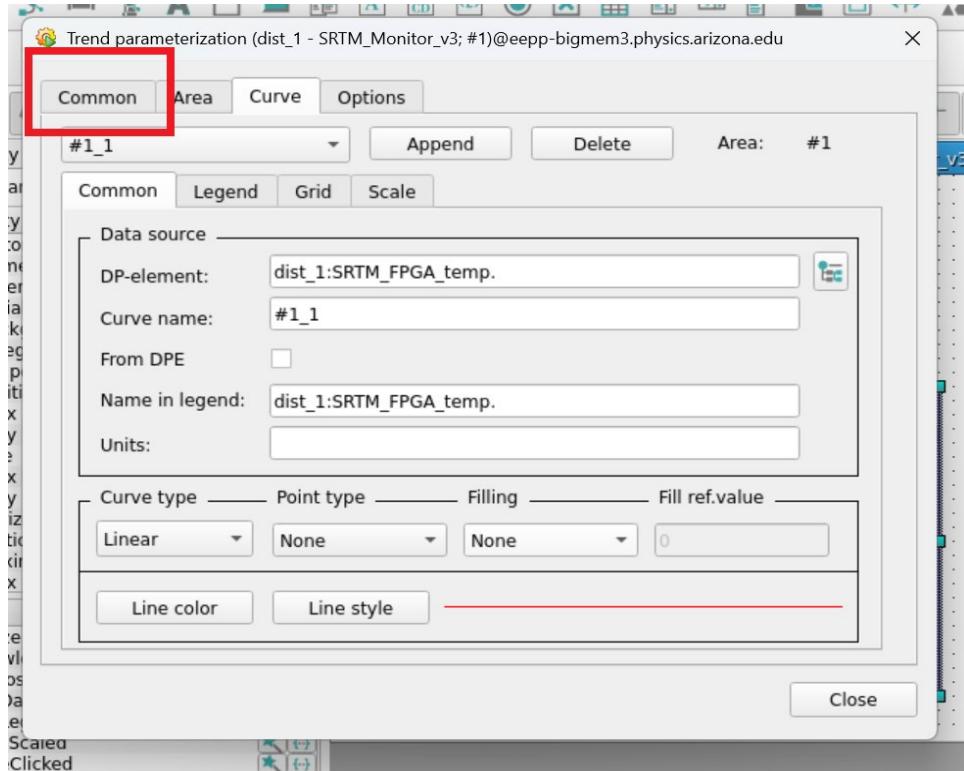
Curve Configuration with Datapoint Bound:



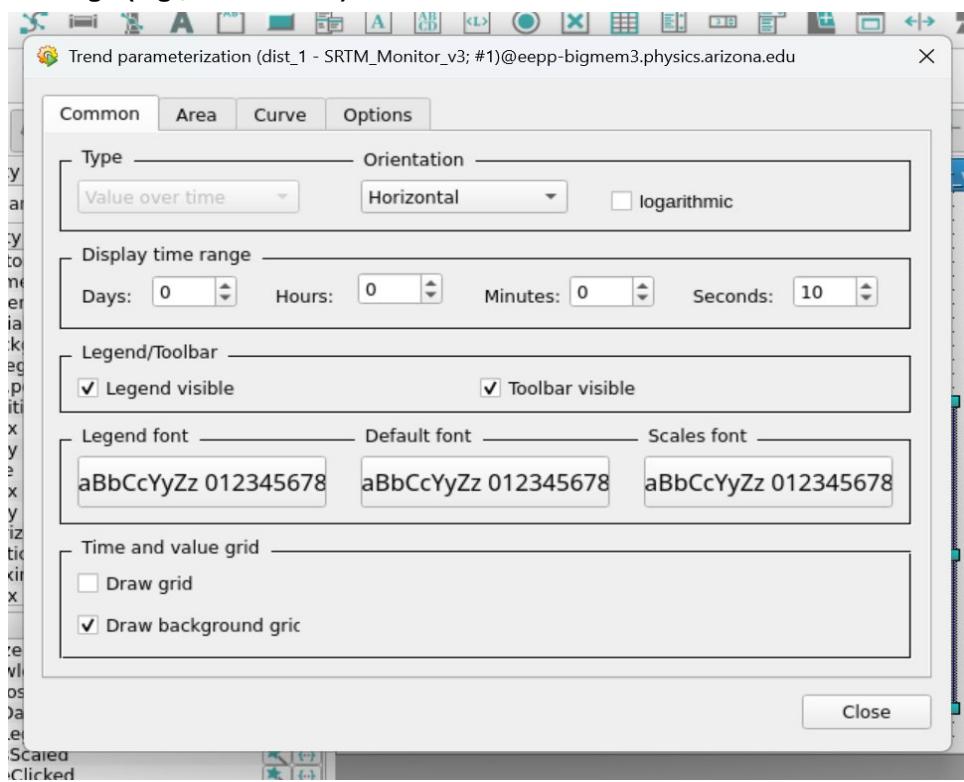
Configuring the Time Scale (X-Axis)

The Trend parameterization has two 'Common' tabs — one at the top level and one under the Curve tab. Use the **top-level Common tab** to set the display time range.

Click the top-level Common tab (highlighted):



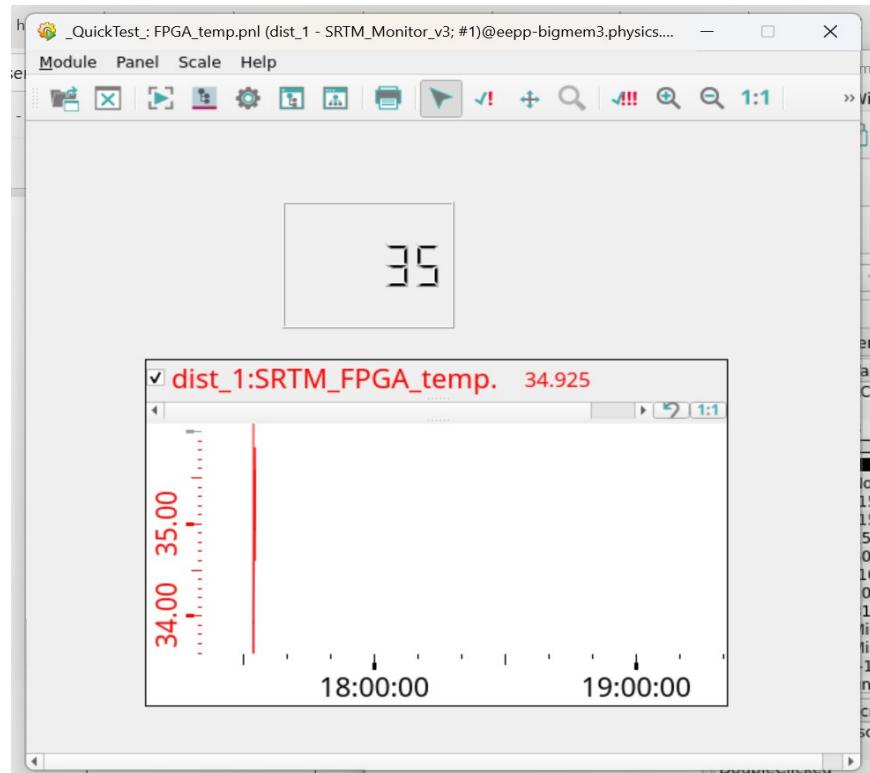
Set Display time range (e.g., 10 seconds):



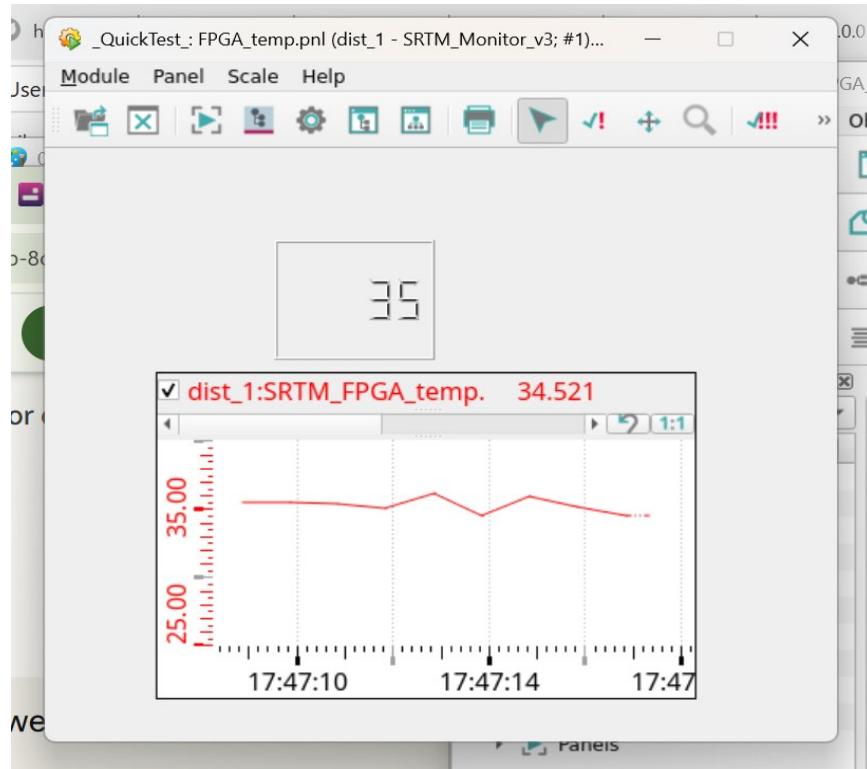
Testing the Panel

1. Save panel: Ctrl+S
2. Quick Test: F5 or Panel → Save and Run in QuickTest Module (Ctrl+Q)

Working GUI (before time-scale adjustment):



After Y-bounds fixed and time-scale set to 10 seconds:



6. Key File Locations

| File/Directory | Path |
|------------------------|---|
| WinCC OA Installation | /opt/WinCC_OA/3.19/ |
| Project Directory | ~/WinCC_Projects/SRTM_Monitor_v3/ |
| Config Files | ~/WinCC_Projects/SRTM_Monitor_v3/config/ |
| Panels | ~/WinCC_Projects/SRTM_Monitor_v3/panels/ |
| Scripts | ~/WinCC_Projects/SRTM_Monitor_v3/scripts/ |
| OPC UA Browser Scripts | /home/naherlin/ |
| Node List (CSV) | /home/naherlin/opcua_nodes.csv |
| Node List (TXT) | /home/naherlin/opcua_nodes.txt |

7. Quick Reference Commands

| Task | Command |
|---------------------|--|
| Start Project | /opt/WinCC_OA/3.19/bin/WCCOApmon -PROJ N_SRTM_test02 & |
| Open GEDI | /opt/WinCC_OA/3.19/bin/WCCOagedi -PROJ N_SRTM_test02 & |
| Browse OPC UA nodes | /usr/bin/python3 ~/browse_opcua.py |
| Check driver status | ps aux grep -i opcua |