# Test-Driven Development (TDD) Style: QT Display Test Cases

## Overview

The following test cases follow a Test-Driven Development process for validating the Qwt-based QT interface, which includes a thermo widget, curve plot, and table. Each test begins with defining the requirement and expected behavior, followed by implementation and verification.

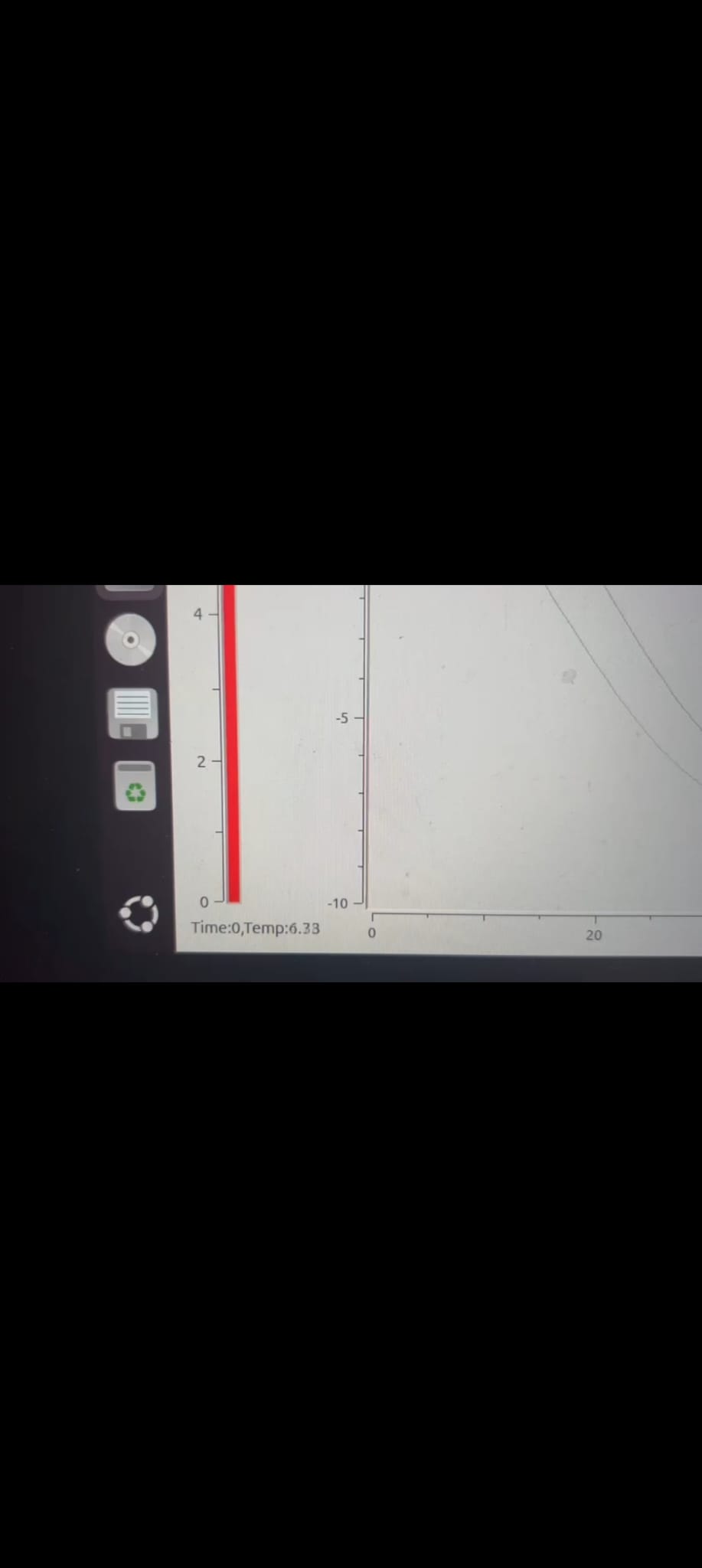
## QT-01: Display Static Thermo

Test Goal: Verify that Qwt Thermo launches and displays a predefined constant value.

Expected Behavior:  
Qwt Thermo appears on screen  
Displays the constant temperature value

Implementation Notes:  
Launch sample Thermo program

Test Result: Passed



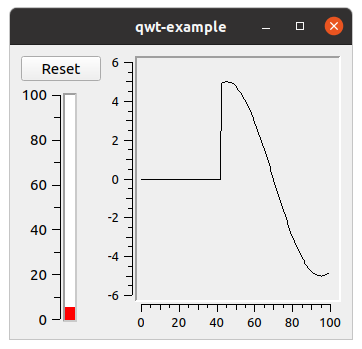
## QT-02: Timer-Based Curve Plot

Test Goal: Verify that a Qwt curve updates periodically using a timer to simulate dynamic plotting.

Expected Behavior:  
Curve updates automatically  
X-axis reflects time, Y-axis reflects temperature

Implementation Notes:  
Use CppTimer class to feed fake temperature data to QwtPlot

Test Result: Passed



## QT-03: Combined Layout – Thermo, Curve, Table

Test Goal: Ensure that Qwt Thermo, Plot, and Table are visible in a unified, horizontal layout.

Expected Behavior:  
All widgets visible without overlap  
Proper spacing and alignment

Implementation Notes:  
Header in Qwt Table was initially cut off

Test Result: Passed with UI improvement noted

## QT-04: Qwt Table — No Cutoff

Test Goal: Ensure the Qwt Table displays DateTime and Temperature values fully.

Expected Behavior:  
No clipped text in headers or cells

Test Result: Passed