OrcaProbe

Reconfigurable Electrical Probing System for Thin Film Devices

Verification & Validation Document

ELEC 491 Capstone Project
University of British Columbia

April 8, 2025

Client: Orca Advanced Materials Inc.
Anindya Lal Roy

Authors: Team JY-85
Aaron Loh, Dipak Shrestha, Idil Bil, Kerem Oktay, Peggy Yuan

Table of Contents

1. Project Overview

Our team, JY85, is developing a reconfigurable probing system called OrcaProbe to probe thin film devices as requested by our client, Orca Advanced Materials Inc. This project will create a consolidated testing system capable of carrying out 2, 3, and 4-probe measurements while communicating and being powered by a host PC via USB connection.

2. Document Overview

This document outlines the strategies, tests, and criteria used to confirm our system meets its design specifications and performs in real-world scenarios. The purpose of this document is to list and organize the tests we will be conducting throughout the project to verify and validate our designs. Verification tests focus on ensuring the system's functionality and performance, while validation tests confirm that our integrated system meets our client's needs and use cases. Each test has an overview of the purpose of the test, the procedure required to perform the test, expected outcomes as a result of theory or simulations, and the results of carrying out the test. Each test fulfills requirements, ensuring that our project is on the right track.

3. Product Verification

Verification tests assess whether the system has been built correctly according to its design specifications and requirements. These tests ensure that each component and subsystem performs as intended.

3.1 Verification Plan Description

This section explains the OrcaProbe Verification test plans. Each verification test is outlined below in Section 3.2 using test cards. Each test card confines all the related details about a verification test within its scope. The parts of the test cards are as follows.

- Tests are given an identity label that starts with the prefix VER, followed by 2-letter identifiers for the test field/scope and ends with the test number.
- Each test has a short title.
- ***Test Overview*** explains the scope and aim of the verification test.
- ***Parent Specification*** provides a reference to the specification the test aims to check and see whether it is met or not.
- ***Test Procedure*** explains the steps to execute the test.
- ***Expected Outcome*** lists the behaviour to observe to decide whether the tests pass or not. Any deviation from the expected outcome is considered a “Fail” unless a reason is given. In this case, the expected outcome can be updated.
- ***Date Executed*** and ***Executed By*** provide bookkeeping-related information on who executed the test and when they executed it.
- ***Test Case Results*** highlight the latest state of the verification tests. Additional details can also be provided when necessary.

Verification Tests are divided into functional categories that concern individual parts or the integration of multiple parts. These categories, their scopes and the number of tests under them are as follows:

- **Basic PCB Functionality:** Aims to verify basic and miscellaneous PCB functionalities. These test plans ensure the PCB can be powered up and accessed via USB.
- **Probe Sourcing:** Aims to verify that the probes can be used as inputs by sourcing voltage and current. These test plans ensure PCB can supply the necessary electrical signals to carry out measurements.
- **Probe Monitoring:** Aims to verify that the probes can be used as outputs by measuring voltage and current. These test plans ensure the PCB can sample the necessary electrical signals to carry out measurements.
- **Probe Toggling:** Aims to verify that the probes can be reconfigured to functionally act as input or outputs when necessary. These test plans ensure PCB offers reconfigurability of probe functionalities.
- **Embedded Firmware Program:** Aims to verify the MCU and the firmware program. These test plans ensure that the hardware system overall behaves as intended and executes the measurements.
- **Software GUI:** Aims to verify the software controller GUI, both the backend and the frontend. These test plans ensure that the GUI can control all the necessary components of the hardware and provide a seamless process to the user.
- **Device-to-Host Communication:** Aims to verify the device-to-host communication. These test plans ensure that the GUI and MCU can communicate. All the information exchange happens over this channel. All message structures are tested to verify that their behaviour is as expected.
- **Mechanical Chassis:** Aims to verify that the chassis is durable, securely holds the PCB in place without movement, and provides sufficient room for the probes to extend.
- **End-to-End Measurement:** Aims to verify the complete system datapath from GUI to hardware. There are dedicated tests for each measurement method. These test plans are directly correlated to the main device functionalities.

3.2 Test Plans

3.2.1 Basic PCB Functionality

VER.HW.1 - PCB Bring Up			
Test Overview	This test verifies that the PCB of the hardware system can be brought up.		
Parent Specification	FR-2		
Test Procedure	**1.** Connect the PCB to the host computer using a USB cable. **2.** Check voltages on power rails using a multimeter.		
Expected Outcome	● All voltage rails should be at their dedicated values. (e.g. 5V, 3.3V, 1.8V, etc.)		
Date Executed	2025-03-30		
Executed by	DS		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.HW.2 - PCB Power Status LED			
Test Overview	This test verifies that the PCB can drive the power status LED correctly.		
Parent Specification	FR-4.2		
Test Procedure	**1.** Connect the PCB to the host computer using a USB cable. **2.** Observe the power status LED over multiple power cycles.		
Expected Outcome	● The LED should turn ON whenever the USB power is connected.		
Date Executed	2025-03-30		
Executed by	DS		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.HW.3 - USB Link Status LED			
Test Overview	This test verifies that the PCB can drive the USB link connection status LED correctly.		
Parent Specification	FR-4.2		
Test Procedure	1. Connect the PCB to the host computer using a USB cable. 2. Run the GUI program to open the serial port connection. 3. Observe the USB link connection status LED over multiple power cycles.		
Expected Outcome	• The LED should turn ON whenever the USB link is connected.		
Date Executed	2025-03-30		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.2 Probe Sourcing

3.2.2.1 Voltage Source Tests

VER.PS.1 - No-Load Voltage Test			
Test Overview	This test verifies that the output voltage wave matches the input setting under no-load conditions.		
Parent Specification	FR-1.1.1		
Test Procedure	**1.** Extend 2 probes. **2.** Make sure the probes are not in contact with any test samples (load). **3.** Connect the USB power cable to the PCB. **4.** Configure the AD9833 to generate a sine wave at a specific frequency and amplitude. **5.** Measure the waveform directly at the probes with an oscilloscope.		
Expected Outcome	● The measured amplitude, frequency, and waveform should match the input setting within acceptable tolerances.		
Date Executed	2025-03-30		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.PS.2 - Load Regulation			
Test Overview	This test verifies that the voltage amplitude of the waveform remains consistent under different test samples.		
Parent Specification	FR-1.1.1		
Test Procedure	**1.** Prepare a selection of different test samples. **2.** Configure the probe position. **3.** Connect the USB power cable to the PCB. **4.** Configure the AD9833 to generate a sine wave at a specific frequency and amplitude. **5.** Probe one of the test samples and measure, with the lab equipment, the voltage being supplied to the sample. **6.** Repeat step 5 with different samples.		
Expected Outcome	● The waveform generator should source the same voltage amplitude across all the test samples.		
Date Executed	2025-03-30		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.PS.3 - DC Source (0.1Hz)	
Test Overview	This test verifies that the waveform generator outputs a DC source when the frequency is set to 0.1 Hz.
Parent Specification	FR-1.1.1
Test Procedure	**1.** Extend 2 probes. **2.** Make sure the probes are not in contact with any test samples. **3.** Connect the USB power to the PCB. **4.** Configure the AD9833 to generate an output at 0.1 Hz and various voltage amplitudes: 0.5V, 1V, 2.5V. 5V. **5.** Measure the output voltage across the probes with the lab equipment. **6.** Repeat steps 4-5 with different voltage amplitudes.

Expected Outcome	• The output waveform should be a flat line at the voltage set by the microcontroller, with no curves representing a DC output.		
Date Executed	2025-03-30		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.PS.4 - High-Frequency Stability (12.5MHz)			
Test Overview	This test verifies that the waveform is stable at high frequencies.		
Parent Specification	FR-2.1.2		
Test Procedure	**1.** Extend 2 probes. **2.** Make sure the probes are not in contact with any test samples. **3.** Connect the USB power to the PCB. **4.** Configure the AD9833 to generate a sine wave at its maximum frequency output of 12.5 MHz and various voltage amplitudes: 0.5V, 1V, 2.5V, and 5V. **5.** Measure the output voltage across the probes with the lab equipment. **6.** Repeat steps 4-5 with different voltage settings.		
Expected Outcome	• The output waveform should be stable with no stepping effect on the sine wave.		
Date Executed	2025-03-30		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*
	As agreed by the client, FR-2.1.2 was changed to have a maximum operating frequency of 1MHz instead of 10MHz. This was to trade off accuracy performance against sampling speed. A high-frequency supply was still tested to ensure the hardware system is able to support it if needed in the future.		

3.2.2.2 Current Source Tests

VER.PS.5 - No-Load Test			
Test Overview	This test verifies that the current source's output equals the calculated output current configured by the digital potentiometers without any load.		
Parent Specification	FR-1.1.1		
Test Procedure	**1.** Extend 2 probes. **2.** Make sure the probes are not in contact with any test samples. **3.** Connect the USB power to the PCB. **4.** Sweep the digital potentiometer resistance from ~70Ω to 100kΩ. **5.** Measure the output current at the probes using a multimeter.		
Expected Outcome	● Output current should range from ~50μA(100kΩ) to ~15mA(70Ω).		
Date Executed	2025-03-30		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.PS.6 - Load Regulation Test			
Test Overview	This test verifies that the output current remains stable under varying loads.		
Parent Specification	FR-1.1.1		
Test Procedure	**1.** Prepare a selection of different test samples. **2.** Configure the probe position. **3.** Connect the USB power cable to the PCB. **4.** Connect the USB cable. **5.** Set the digital potentiometer to ~70Ω **6.** Extend 2 probes to contact the test sample. **7.** Measure the current through the load using a multimeter. **8.** Sweep the potentiometer to 100kΩ. **9.** Compare the currents across different resistive loads. **10.** Repeat steps 5-9 with different samples.		
Expected Outcome	• Output current across the samples should be the same as the no-load condition and should not vary with load unless saturated.		
Date Executed	2025-03-30		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.3 Probe Monitoring

<table>
<tr><th colspan="4">VER.PM.1 - DC Voltage Measurement Test</th></tr>
<tr><td>Test Overview</td><td colspan="3">This test verifies that the monitoring system can accurately measure a constant DC voltage value.</td></tr>
<tr><td>Parent Specification</td><td colspan="3">FR-2.2</td></tr>
<tr><td>Test Procedure</td><td colspan="3">Extend 2 test probes.
Connect the DC power supply outputs to the probes.
Connect the USB power to the device.
Enable output.
Capture and record the data from the monitoring system.
Visualize the data using either the software GUI or an external software.
<table><tr><th>Voltage Amplitudes (V)</th><th>Frequency Values</th></tr><tr><td>0.5</td><td>0.5 Hz</td></tr><tr><td>1</td><td>1 Hz</td></tr><tr><td>2.5</td><td>1 kHz</td></tr><tr><td>5</td><td>10 MHz</td></tr></table></td></tr>
<tr><td>Expected Outcome</td><td colspan="3">For each voltage and frequency level, the output should be a series of data points at or near the set voltage level.</td></tr>
<tr><td>Date Executed</td><td colspan="3">2025-03-30</td></tr>
<tr><td>Executed by</td><td colspan="3">DS - KO</td></tr>
<tr><td>Test Case Results</td><td>Pass</td><td>Fail</td><td>Incomplete</td></tr>
</table>

<table>
<tr><th colspan="4">VER.PM.2 - Sine Wave Measurement Test</th></tr>
<tr><td>Test Overview</td><td colspan="3">This test verifies that the monitoring system can accurately measure a sine wave up to 10 MHz.</td></tr>
<tr><td>Parent Specification</td><td colspan="3">FR-2.2</td></tr>
<tr><td>Test Procedure</td><td colspan="3">

Extend 2 test probes.
Connect the function generator outputs to the probes.
Connect USB power to the device.
Set the lab function generator to a sine wave on a specified voltage level and frequency.
Enable output.
Capture and record data from the monitoring system.
Visualize the data using either the software GUI or an external software.

<table>
<tr><th>Voltage Amplitudes (V)</th><th>Frequency Values</th></tr>
<tr><td>0.5</td><td>0.5 Hz</td></tr>
<tr><td>1</td><td>1 Hz</td></tr>
<tr><td>2.5</td><td>1 kHz</td></tr>
<tr><td>5</td><td>10 MHz</td></tr>
</table>
</td></tr>
<tr><td>Expected Outcome</td><td colspan="3">For each voltage and frequency pair, the output should be a series of data points that resemble the specified sine wave.</td></tr>
<tr><td>Date Executed</td><td colspan="3">2025-03-30</td></tr>
<tr><td>Executed by</td><td colspan="3">DS - KO</td></tr>
<tr><td>Test Case Results</td><td>Pass</td><td>Fail</td><td>Incomplete</td></tr>
</table>

VER.PM.3 - DC Current Measurement Test			
Test Overview	This test verifies that the monitoring system can accurately measure a constant DC current level.		
Parent Specification	FR-2.2		
Test Procedure	1. Extend 2 test probes. 2. Connect the positive side of the function generator output to the 1kΩ resistor. 3. Connect an 11kΩ resistor to one test probe. 4. Connect the negative side of the function generator output to the second test probe. 5. Connect the USB power to the device. 6. Configure the DC power supply to output at the specified voltage level. 7. Enable output. 8. Capture and record data from the monitoring system.		
Expected Outcome	• For each voltage level, the output should be a series of data points at or near the expected current level.		
Date Executed	2025-04-01		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

Voltage Amplitudes (V)	**Expected Current (mA)**
1	1
2.5	2.5
5	5
10	10

<table>
<tr><th colspan="4">VER.PM.4 - Current Sine Wave Measurement Test</th></tr>
<tr><td>Test Overview</td><td colspan="3">This test verifies that the monitoring system can accurately measure a current sine wave up to 10 MHz.</td></tr>
<tr><td>Parent Specification</td><td colspan="3">FR-2.2</td></tr>
<tr><td>Test Procedure</td><td colspan="3">
1. Extend 2 test probes.
2. Connect the positive side of the function generator output to the 1kΩ resistor.
3. Connect an 11kΩ resistor to one test probe.
4. Connect the negative side of the function generator output to the second test probe.
5. Connect USB power to the device.
6. Set the lab function generator to output voltage sine wave at the specified voltage level and frequency.
7. Enable output.
8. Capture and record data from the monitoring system.
<table>
<tr><th>Voltage Amplitudes (V)</th><th>Expected Current (mA)</th><th>Frequency Values</th></tr>
<tr><td>1</td><td>1</td><td>0.5 Hz</td></tr>
<tr><td>2.5</td><td>2.5</td><td>1 Hz</td></tr>
<tr><td>5</td><td>5</td><td>1 kHz</td></tr>
<tr><td>10</td><td>10</td><td>10 MHz</td></tr>
</table>
</td></tr>
<tr><td>Expected Outcome</td><td colspan="3">• For each voltage and frequency pair, the output should be a series of data points that resemble the specified sine wave.</td></tr>
<tr><td>Date Executed</td><td colspan="3">2025-04-01</td></tr>
<tr><td>Executed by</td><td colspan="3">DS - KO</td></tr>
<tr><td>Test Case Results</td><td>Pass</td><td>Fail</td><td>Incomplete</td></tr>
</table>

3.2.4 Probe Toggling

VER.PT.1 - Single Pin Toggle Between Different Inputs			
Test Overview	This test verifies that the signal toggling mechanism works for a single pin.		
Parent Specification	FR-2.3		
Test Procedure	1. Set up the microcontroller program to output DC and AC voltage and current. 2. Apply DC voltage on the first pin, AC voltage on the second, current on the third, and GND on the fourth. 3. Use a load resistor to measure current on the third probe. 4. Measure the voltage/current for each signal at the output.		
Expected Outcome	• The resulting voltage and current at the output should be the supplied signal at the input, depending on which switch is toggled.		
Date Executed	2025-03-30		
Executed by	DS - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.5 Embedded Firmware Program

3.2.5.1 General Functionality

VER.FW.1 - Boot State Reset			
Test Overview	This test ensures the proper booting and shutdown of the MCU, verifying that the firmware resets to its initial state with the default values upon power-up.		
Parent Specification	FR-2.6.1		
Test Procedure	**1.** Connect to the PCB using USB and ST-Link stick. **2.** Check default values for firmware variables using the STM32CubeIDE debugger. **3.** Using the GUI, send configuration data. **4.** Ensure variables are updated as intended. **5.** Disconnect and reconnect the USB to power-cycle the PCB and MCU. **6.** Check default values for firmware variables using the STM32CubeIDE debugger.		
Expected Outcome	• Variables seen in steps 2 and 6 should match.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.FW.2 - User Asserted Reset	
Test Overview	This test ensures the proper resetting of the MCU, verifying that the firmware resets to its initial state with the default values upon a user-asserted reset.
Parent Specification	FR-2.6.2
Test Procedure	**1.** Connect to the PCB using USB and ST-Link stick. **2.** Check default values for firmware variables using the STM32CubeIDE debugger. **3.** Using the GUI, send configuration data. **4.** Ensure variables are updated as intended. **5.** Assert the hardware reset by pressing the mechanical reset button.

	6. Check default values for firmware variables using the STM32CubeIDE debugger. 7. Using the GUI, send configuration data. 8. Ensure variables are updated as intended. 9. Assert software reset by pressing the reset button on the GUI. 10. Check default values for firmware variables using the STM32CubeIDE debugger.		
Expected Outcome	• Variables seen in step 2, step 6 and step 10 should match.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.FW.3 - Switch Network Control			
Test Overview	This test verifies that the MCU can control all the individual relays of the switch network and connect all 4 probes to each functionality circuit.		
Parent Specification	FR-2.3		
Test Procedure	1. Connect to the PCB using USB and ST-Link stick. 2. Run a dedicated program in DEBUG mode that goes over each GPIO pin for each relay and toggles them. 3. Monitor values for firmware variables using the STM32CubeIDE debugger. 4. Check for connectivity between the probes and the functionality circuit using a multimeter.		
Expected Outcome	• Multimeter should pass continuity checks by showing a continuous path.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.5.2 MCU to IC Communication

VER.FW.4 - Voltage Source Configuring via SPI			
Test Overview	This test verifies the SPI bus functionality between the MCU and voltage source IC, AD9833, ensuring desired data communication between the two ends.		
Parent Specification	FR-3.3.2, FR-2.1		
Test Procedure	1. Connect the oscilloscope to the SPI bus test lines. 2. Connect to the PCB using USB and ST-Link stick. 3. Set up the firmware to pause at SPI routines. 4. Using the GUI, send configuration data that involves sweeping. 5. When the breakpoint hits for the SPI routine, trigger the oscilloscope. 6. Continue SPI routine. 7. Capture the waveform on the oscilloscope, including the serial clock, chip select and serial data lines.		
Expected Outcome	• The captured waveform on the oscilloscope should match the serial data pattern specified on the AD9833 datasheet.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.FW.5 - Current Source Configuring via I2C	
Test Overview	This test verifies the I2C bus functionality between the MCU and potentiometers of the current mirror, ensuring desired data communication between the two ends.
Parent Specification	FR-3.3.2, FR-2.1
Test Procedure	1. Connect the oscilloscope to the I2C bus test lines. 2. Connect to the PCB using USB and ST-Link stick. 3. Set up the firmware to pause at I2C routines. 4. Using the GUI, send configuration data that involves sweeping. 5. When the breakpoint hits for the I2C routine, trigger the

	oscilloscope. 6. Continue the I2C routine. 7. Capture the waveform on the oscilloscope, including the serial clock and serial data lines.		
Expected Outcome	• The captured waveform on the oscilloscope should match the serial data pattern specified on the MCP453X datasheet.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.FW.6 - ADC Clock Quality			
Test Overview	This test verifies that the clock signal generated by the MCU is fed to the ADCs (AD9235) in the required quality for sampling to function.		
Parent Specification	FR-2.2		
Test Procedure	1. Connect the oscilloscope to the clock test lines. 2. Connect to the PCB using USB and ST-Link stick. 3. Set up the firmware to pause at DMA triggering routines. 4. Using the GUI, send configuration data that involves static sampling. 5. When the breakpoint hits for the DMA routine, trigger the oscilloscope. 6. Continue the DMA routine. 7. Capture the waveform on the oscilloscope.		
Expected Outcome	• The captured waveform on the oscilloscope should show minimal ringing, overshoot and jitter on the clock signal. • It needs to have a 50% duty cycle.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.6 Software GUI

3.2.6.1 Front-End

VER.SW.1 - Boot State Reset			
Test Overview	This test ensures the proper booting and shutdown of the GUI, verifying that the software resets to its initial state with the default values.		
Parent Specification	FR-3.7		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Start the GUI by running the Python file. **3.** Change the probe functionalities, toggle the measurement menu and enter input values for the measurements. **4.** Terminate the GUI. **5.** Start the GUI and observe the changes made.		
Expected Outcome	• All the changes made should go back to their initial, default state.		
Date Executed	2025-02-02		
Executed by	IB		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.SW.2 - Measurement Type Menu Toggling and Page Selection	
Test Overview	This test verifies that the measurement type selection menu on the left is toggleable, allowing only one measurement to be selected at a time, and ensures the corresponding page is displayed upon selection.
Parent Specification	FR-3.2
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Start the GUI by running the Python file. **3.** Toggle the measurement selection menu on the left to see what is displayed under each measurement category (2-probe, 3-probe, 4-probe). **4.** Select a measurement type and observe the corresponding page.

	5. Repeat steps 3-4 for all possible measurement types.		
Expected Outcome	• Selecting a measurement type should display its corresponding page in the main area. • Previously selected measurement types should automatically be deselected when a new one is chosen.		
Date Executed	2025-02-02		
Executed by	IB		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.SW.3 - Providing Measurement Input			
Test Overview	This test verifies that selecting a measurement type displays its corresponding input fields, allowing the device operator to enter values into the provided inputs.		
Parent Specification	FR-3.3		
Test Procedure	1. Connect the device to the operator's pc. 2. Start the GUI by running the Python file. 3. Choose a measurement type from the measurement selection menu on the left. 4. Observe the input fields on the measurement page. 5. Enter values in the input fields. 6. Monitor corresponding firmware variables using STM32CubeIDE. 7. Repeat steps 3-5 for all possible measurement types.		
Expected Outcome	• Input fields specific to the selected measurement type should be displayed. • Device operator should be able to successfully enter values into the displayed input fields without errors.		
Date Executed	2025-03-30		
Executed by	IB - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.SW.4 - Individual Probe Configurability Setting			
Test Overview	This test verifies that the probe configuration bar in the GUI allows configuring individual probe functionalities and selecting which pins are in use.		
Parent Specification	FR-3.1, FR-1.1.X		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Start the GUI by running the Python file. **3.** Choose a measurement type. **4.** Change the probe configuration based on how many probes are needed. **5.** Set the probe's functionalities based on the measurement type. **6.** Monitor corresponding firmware variables using STM32CubeIDE. **7.** Check for continuity between the selected probe and its functionality circuit. **8.** Repeat steps 3-7 for all possible measurement types and probe-functionality combinations.		
Expected Outcome	● Only the selected probes should be active, while the others remain "Off" as configured in the GUI. ● Active probes should pass a continuity check for the functionality circuit specified in the probe configuration bar dropdowns.		
Date Executed	2025-03-30		
Executed by	IB - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.SW.5 - Measurement Control			
Test Overview	This test verifies the functionality of the Start and Stop buttons for measurements.		
Parent Specification	FR-3.3.1, FR-3.3.2		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Start the GUI by running the Python file. **3.** Choose a measurement type from the measurement selection menu on the left. **4.** Enter relevant values in the input fields. **5.** Start measurement using the button. **6.** Observe when it stops. **7.** Repeat steps 3-6 for all possible measurement types.		
Expected Outcome	● For static measurements: The Start button should initiate the measurement. ● For dynamic measurements: The Start button should initiate the measurement, and it should stop automatically when the process completes based on the input values entered.		
Date Executed	2025-03-30		
Executed by	IB		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.6.2 Back-End

VER.SW.6 - Device Serial Port Connection			
Test Overview	This test is to verify that the code can automatically detect the communication port used when the device is connected to the operator's PC.		
Parent Specification	FR-3.8		
Test Procedure	**1.** Connect the device to the operator’s pc. **2.** Start the GUI by running the Python file. **3.** Observe the connection LED on the hardware. **4.** Repeat steps 1-4 for different computers and USB ports.		
Expected Outcome	● Code should automatically detect the COM port used to connect the device. ● Microcontroller should light up the connection LED on the hardware, indicating a successful connection.		
Date Executed	2025-03-30		
Executed by	IB - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.SW.7 - Data Logging	
Test Overview	This test ensures that the results of the measurements are saved correctly in a CSV or JSON file.
Parent Specification	FR-3.5
Test Procedure	**1.** Connect the device to the operator’s pc. **2.** Start the GUI by running the Python file. **3.** Choose a measurement type from the measurement selection menu on the left. **4.** Enter relevant values in the input fields. **5.** Choose the data logging format (.CSV or .JSON). **6.** Start measurement using the button. **7.** Observe the data file saved in the device operator's PC after the measurement output is displayed. **8.** Repeat steps 3-7 for all possible measurement types and data logging combinations.

Expected Outcome	● There should be a data file saved on the Desktop. ● File name should be a timestamp in the YYYYMMDD_HHMMSS format to ensure uniqueness. ● File should contain all the measured values.		
Date Executed	2025-03-30		
Executed by	IB - KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.SW.8 - Measurement Output Display			
Test Overview	This test verifies that each measurement type displays accurate and relevant outputs corresponding to the selected measurement, ensuring alignment with the expected results.		
Parent Specification	FR-3.4		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Start the GUI by running the Python file. **3.** Choose a measurement type from the measurement selection menu on the left. **4.** Enter relevant values in the input fields. **5.** Start measurement using the button. **6.** Observe the output displayed once the measurement is completed. **7.** Repeat steps 3-6 for all possible measurement types.		
Expected Outcome	● For static measurements, the output should display a single value, which is the expected result of the measurement. ● For dynamic measurements, the output should be a graph that is plotted upon the completion of the measurement.		
Date Executed	2025-03-30		
Executed by	IB		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.7 Device-to-Host Communication

VER.CM.1 - Device-to-Host USB communication sanity check			
Test Overview	This test verifies that the device-to-host USB link is established and the message structure works.		
Parent Specification	FR-2.4		
Test Procedure	**1.** Connect the device to the operator’s pc. **2.** Start the debug communication interface by running the dedicated Python file. **3.** Send messages with random addresses and data combinations. **4.** Monitor corresponding firmware registers using STM32CubeIDE. **5.** Request the same addresses randomly and monitor the data the MCU returns.		
Expected Outcome	● The data sent and returned should match for each individual register address.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.CM.2 - Probe Configuration Messages	
Test Overview	This test verifies that the device-to-host USB link can communicate probe configuration messages correctly.
Parent Specification	FR-2.4, FR-3.1
Test Procedure	**1.** Connect the device to the operator’s pc. **2.** Start the debug communication interface by running the dedicated Python file. **3.** Send messages with probe configuration register addresses and data combinations. **4.** Monitor corresponding firmware registers using STM32CubeIDE. **5.** Request the same addresses and monitor the data MCU returns.

Expected Outcome	• The data sent and returned should match for each individual register address.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.CM.3 - Measurement Configuration Messages			
Test Overview	This test verifies that the device-to-host USB link can communicate measurement configuration messages correctly.		
Parent Specification	FR-2.4, FR-3.2		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Start the debug communication interface by running the dedicated Python file. **3.** Send messages with measurement configuration register addresses and data combinations. **4.** Monitor corresponding firmware registers using STM32CubeIDE. **5.** Request the same addresses and monitor the data MCU returns.		
Expected Outcome	• The data sent and returned should match for each individual register address.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.CM.4 - Measurement Sample Flushing Messages			
Test Overview	This test verifies that the device-to-host USB link can communicate sample data at the end of a measurement.		
Parent Specification	FR-2.4, FR-3.4		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Start the debug communication interface by running the dedicated Python file. **3.** Fill the MCU flash with dummy sample data. **4.** Send messages with measurement sample request register addresses and data combinations. **5.** Monitor corresponding firmware registers using STM32CubeIDE. **6.** Monitor the flushed sample data on the host.		
Expected Outcome	● The flushed sample data should match the dummy data.		
Date Executed	2025-03-30		
Executed by	KO		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.8 Mechanical Chassis

VER.MC.1 - Structural Integrity / Force Application Test (Without PCB)			
Test Overview	This test verifies that the chassis is robust and can withstand handling without deformation.		
Parent Specification	FR-4.1		
Test Procedure	**1.** Hold the chassis with one hand. **2.** Apply a grip force to the chassis at different locations (top, bottom, sides, etc). **3.** Visually inspect for cracks, deformation, or structural failures. **4.** Repeat steps 1-3 for all locations.		
Expected Outcome	● Chassis should show no cracks, warping, or functional damage.		
Date Executed	2025-02-08		
Executed by	PY		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.MC.2 - Structural Integrity / Drop Test (Without PCB)	
Test Overview	This test verifies that the chassis is robust and can withstand handling without deformation.
Parent Specification	FR-4.1
Test Procedure	**1.** Hold the chassis at 0.5 m height above a hard surface. **2.** Drop it in 3 orientations: a. Front facing down b. Side facing down c. Rear facing down **3.** Visually inspect for cracks, breakage, or loosening of internal components.
Expected Outcome	● Chassis should show no cracks, warping, or functional damage.

Date Executed	2025-02-08		
Executed by	PY		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.MC.3 - PCB Secure Mounting Test			
Test Overview	This test verifies that the PCB remains securely in place without movement.		
Parent Specification	FR-4		
Test Procedure	**1.** Hold the chassis with one hand. **2.** Shake it gently in multiple directions (up/down, left/right, forward/backward). **3.** Listen and feel for any movement or rattling inside. **4.** Repeat steps 2-3 in a different orientation.		
Expected Outcome	● There should be no PCB misalignment, movement or rattling after shaking.		
Date Executed	2025-03-30		
Executed by	PY		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.MC.4 - Probe Slider Mechanism Test	
Test Overview	This test verifies that the sliders function smoothly and extend the selected probes.
Parent Specification	FR-4.1
Test Procedure	**1.** Move each probe slider up and down manually 10 times to check for smoothness. **2.** Observe for any excessive resistance, jamming, or misalignment. **3.** Slide all the probes downward. **4.** Align the probes with the test samples to ensure alignment with the measuring points.

Expected Outcome	• Sliders should operate smoothly with no excessive resistance, and probes should extend properly aligned.		
Date Executed	2025-02-08		
Executed by	PY		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.MC.5 - Visual Indicator Functionality Test			
Test Overview	This test verifies that the LEDs can be viewed from outside the case.		
Parent Specification	FR-4.2		
Test Procedure	**1.** Connect the hardware to a power supply. **2.** Observe from outside the case if the LEDs are clear to see.		
Expected Outcome	• LED indicators should be clearly seen from outside the chassis.		
Date Executed	2025-03-30		
Executed by	PY - DS		
Test Case Results	*Pass*	*Fail*	*Incomplete*

3.2.9 End-to-End Measurements

VER.EE.1 - DC Resistance Measurement (2-probe)			
Test Overview	This test verifies that the device can complete a 2-probe DC Resistance measurement.		
Parent Specification	FR-1.2		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Connect two probes to a material to test. **3.** Start the GUI by running the Python file. **4.** Choose the correct measurement type on the left side menu. **5.** Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar. **6.** Enter the necessary input values. **7.** Start measurement. **8.** Observe the output displayed. **9.** Compare the measured output values to the expected output values. **10.** Repeat steps 5-9 for different probe configurations.		
Expected Outcome	• The measured values should be almost the same as the expected measurement result within acceptable tolerances.		
Date Executed	2025-04-01		
Executed by	TEAM		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.EE.2 - Current-Voltage Measurement (2-probe)			
Test Overview	This test verifies that the device can complete a 2-probe Current-Voltage measurement.		
Parent Specification	FR-1.2		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Connect two probes to a material to test. **3.** Start the GUI by running the Python file. **4.** Choose the correct measurement type on the left side menu. **5.** Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar. **6.** Enter the necessary input values. **7.** Start measurement. **8.** Observe the output displayed. **9.** Compare the measured output values to the expected output values. **10.** Repeat steps 5-9 for different probe configurations.		
Expected Outcome	• The measured values should be almost the same as the expected measurement result within acceptable tolerances.		
Date Executed	2025-04-01		
Executed by	TEAM		
Test Case Results	*Pass*	*Fail*	*Incomplete*

<table>
<tr><th colspan="4">VER.EE.3 - Capacitance-Voltage Measurement (2-probe)</th></tr>
<tr><td>Test Overview</td><td colspan="3">This test verifies that the device can complete a 2-probe Capacitance-Voltage measurement.</td></tr>
<tr><td>Parent Specification</td><td colspan="3">FR-1.2</td></tr>
<tr><td>Test Procedure</td><td colspan="3">1. Connect the device to the operator’s pc.
2. Connect two probes to a material to test.
3. Start the GUI by running the Python file.
4. Choose the correct measurement type on the left side menu.
5. Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar.
6. Enter the necessary input values.
7. Start measurement.
8. Observe the output displayed.
9. Compare the measured output values to the expected output values.
10. Repeat steps 5-9 for different probe configurations.</td></tr>
<tr><td>Expected Outcome</td><td colspan="3">● The measured values should be almost the same as the expected measurement result within acceptable tolerances.</td></tr>
<tr><td>Date Executed</td><td colspan="3">2025-04-07</td></tr>
<tr><td>Executed by</td><td colspan="3">TEAM</td></tr>
<tr><td>Test Case Results</td><td>Pass</td><td>Fail</td><td>Incomplete</td></tr>
</table>

<table>
<tr><th colspan="4">VER.EE.4 - Impedance Spectroscopy Measurement (2-probe)</th></tr>
<tr><td>Test Overview</td><td colspan="3">This test verifies that the device can complete a 2-probe Impedance Spectroscopy measurement.</td></tr>
<tr><td>Parent Specification</td><td colspan="3">FR-1.2</td></tr>
<tr><td>Test Procedure</td><td colspan="3">1. Connect the device to the operator’s pc.
2. Connect two probes to a material to test.
3. Start the GUI by running the Python file.
4. Choose the correct measurement type on the left side menu.
5. Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar.
6. Enter the necessary input values.
7. Start measurement.
8. Observe the output displayed.
9. Compare the measured output values to the expected output values.
10. Repeat steps 5-9 for different probe configurations.</td></tr>
<tr><td>Expected Outcome</td><td colspan="3">● The measured values should be almost the same as the expected measurement result within acceptable tolerances.</td></tr>
<tr><td>Date Executed</td><td colspan="3">2025-04-07</td></tr>
<tr><td>Executed by</td><td colspan="3">TEAM</td></tr>
<tr><td rowspan="2">Test Case Results</td><td>Pass</td><td>Fail (Partially)</td><td>Incomplete</td></tr>
<tr><td colspan="3">Impedance spectroscopy aims to calculate impedance and phase of materials under different frequencies. It is found that the impedance portion is passing but the phase is failing. This was root caused by multiple DMA channels not capturing samples in a synced manner even though ADCs sample in sync (verified with an oscilloscope).</td></tr>
</table>

VER.EE.5 - Transfer Characteristics Measurement (3-probe)			
Test Overview	This test verifies that the device can complete a 3-probe Transfer Characteristics measurement.		
Parent Specification	FR-1.3		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Connect three probes to a material to test. **3.** Start the GUI by running the Python file. **4.** Choose the correct measurement type on the left side menu. **5.** Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar. **6.** Enter the necessary input values. **7.** Start measurement. **8.** Observe the output displayed. **9.** Compare the measured output values to the expected output values. **10.** Repeat steps 5-9 for different probe configurations.		
Expected Outcome	● The measured values should be almost the same as the expected measurement result within acceptable tolerances.		
Date Executed	2025-04-01		
Executed by	TEAM		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.EE.6 - Output Characteristics Measurement (3-probe)			
Test Overview	This test verifies that the device can complete a 3-probe Output Characteristics measurement.		
Parent Specification	FR-1.3		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Connect three probes to a material to test. **3.** Start the GUI by running the Python file. **4.** Choose the correct measurement type on the left side menu. **5.** Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar. **6.** Enter the necessary input values. **7.** Start measurement. **8.** Observe the output displayed. **9.** Compare the measured output values to the expected output values. **10.** Repeat steps 5-9 for different probe configurations.		
Expected Outcome	● The measured values should be almost the same as the expected measurement result within acceptable tolerances.		
Date Executed	2025-04-01		
Executed by	TEAM		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.EE.7- Capacitance-Voltage Measurement (3-probe)			
Test Overview	This test verifies that the device can complete a 3-probe Capacitance-Voltage measurement.		
Parent Specification	FR-1.3		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Connect three probes to a material to test. **3.** Start the GUI by running the Python file. **4.** Choose the correct measurement type on the left side menu. **5.** Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar. **6.** Enter the necessary input values. **7.** Start measurement. **8.** Observe the output displayed. **9.** Compare the measured output values to the expected output values. **10.** Repeat steps 5-9 for different probe configurations.		
Expected Outcome	● The measured values should be almost the same as the expected measurement result within acceptable tolerances.		
Date Executed	2025-04-08		
Executed by	TEAM		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.EE.8 - Electrochemical Measurement (3-probe)			
Test Overview	This test verifies that the device can complete a 3-probe Electrochemical measurement.		
Parent Specification	FR-1.3		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Connect three probes to a material to test. **3.** Start the GUI by running the Python file. **4.** Choose the correct measurement type on the left side menu. **5.** Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar. **6.** Enter the necessary input values. **7.** Start measurement. **8.** Observe the output displayed. **9.** Compare the measured output values to the expected output values. **10.** Repeat the steps 5-9 for different probe configurations.		
Expected Outcome	• The measured values should be almost the same as the expected measurement result within acceptable tolerances.		
Date Executed			
Executed by			
Test Case Results	*Pass*	*Fail*	*Incomplete*
	Due to lack of samples or discrete components that fit into the category of "electrochemical", this measurement was not tested. The code flow is the same as impedance spectroscopy which is verified in other end-to-end system tests. The risk taken by not testing this measurement is justifiable as the code/hardware portion is verified in general and only lacking test is actually seeing performance with electrochemical materials.		

VER.EE.9 - Probe Resistance Measurement (4-probe)			
Test Overview	This test verifies that the device can complete a 4-probe Probe Resistance measurement.		
Parent Specification	FR-1.4		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Connect four probes to a material to test. **3.** Start the GUI by running the Python file. **4.** Choose the correct measurement type on the left side menu. **5.** Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar. **6.** Enter the necessary input values. **7.** Start measurement. **8.** Observe the output displayed. **9.** Compare the measured output values to the expected output values. **10.** Repeat steps 5-9 for different probe configurations.		
Expected Outcome	• The measured values should be almost the same as the expected measurement result within acceptable tolerances.		
Date Executed	2025-04-08		
Executed by	TEAM		
Test Case Results	*Pass*	*Fail*	*Incomplete*

VER.EE.10 - Low Resistance Measurement (4-probe)			
Test Overview	This test verifies that the device can complete a 4-probe Low Resistance measurement.		
Parent Specification	FR-1.4		
Test Procedure	**1.** Connect the device to the operator's pc. **2.** Connect four probes to a material to test. **3.** Start the GUI by running the Python file. **4.** Choose the correct measurement type on the left side menu. **5.** Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar. **6.** Enter the necessary input values. **7.** Start measurement. **8.** Observe the output displayed. **9.** Compare the measured output values to the expected output values. **10.** Repeat steps 5-9 for different probe configurations.		
Expected Outcome	● The measured values should be almost the same as the expected measurement result within acceptable tolerances.		
Date Executed	2025-04-08		
Executed by	TEAM		
Test Case Results	*Pass*	*Fail*	*Incomplete*

<table>
<tr><th colspan="4">VER.EE.11 - Impedance Spectroscopy Measurement (4-probe)</th></tr>
<tr><td>Test Overview</td><td colspan="3">This test verifies that the device can complete a 4-probe Impedance Spectroscopy measurement.</td></tr>
<tr><td>Parent Specification</td><td colspan="3">FR-1.4</td></tr>
<tr><td>Test Procedure</td><td colspan="3">1. Connect the device to the operator’s pc.
2. Connect four probes to a material to test.
3. Start the GUI by running the Python file.
4. Choose the correct measurement type on the left side menu.
5. Set the probe functionalities based on the measurement type and probe amount using the probe configuration bar.
6. Enter the necessary input values.
7. Start measurement.
8. Observe the output displayed.
9. Compare the measured output values to the expected output values.
10. Repeat steps 5-9 for different probe configurations.</td></tr>
<tr><td>Expected Outcome</td><td colspan="3">● The measured values should be almost the same as the expected measurement result within acceptable tolerances.</td></tr>
<tr><td rowspan="2">Test Case Results</td><td>Pass</td><td>Fail (Partially)</td><td>Incomplete</td></tr>
<tr><td colspan="3">Impedance spectroscopy aims to calculate impedance and phase of materials under different frequencies. It is found that the impedance portion is passing but the phase is failing. This was root caused by multiple DMA channels not capturing samples in a synced manner even though ADCs sample in sync (verified with an oscilloscope).</td></tr>
</table>

4. Product Validation

Validation tests determine whether the system as a whole meets the user needs and intended application in its real-world context. These tests confirm that the final system fulfills its purpose effectively, typically involving performance evaluation, user testing, and comparisons against expected outcomes.

4.1 Validation Plan Description

This section explains the OrcaProbe validation test plans. Each validation criterion is outlined below in Section 4.2 using the same test cards as in Section 3.2. Please refer to Section 3.1 for the explanation of the test cards.

For the validation criteria, there are certain conceptual changes in the test card. Certain parts are taken out, such as ***Parent Specification***, ***Execution Date***, ***Executed By*** and ***Test Case Results***, as they do not apply to validation tests. Validation criteria are directly or indirectly correlated to the whole set of product specifications. ***Test Results*** are shown in Table 1 after the validation test cards.

For validation tests, actual printed thin film characterization scenarios are mimicked. Table 1 lists all the scenarios that are under the scope of the validation tests. Validation criteria are evaluated on a scenario basis.

4.2 Validation Criteria

VAL.1 - Measurement Accuracy	
Criterion Overview	This criterion validates that OrcaProbe can provide accurate and precise data that is usable from the operator's point of view.
Procedure	**1.** Using conventional methods, measure the desired values of a material and perform the necessary measurements. **2.** Carry out the same measurement(s) using OrcaProbe. **3.** Compare the values and note the difference.
Expected Outcome	● OrcaProbe should produce a value within ±5% of the expected result of the measurement.

VAL.2 - Reconfigurability	
Criterion Overview	This criterion validates that OrcaProbe can provide reconfigurability of the probes and measurement types.
Procedure	**1.** Using conventional methods, measure the desired values. In the process, record the number of times a new tool is used, some setup is changed and/or material is repositioned. **2.** Carry out the measurement using OrcaProbe and record the number of times a new tool is used, some setup is changed and/or material is repositioned. **3.** Compare the number of times OrcaProbe was reused to carry out different measurements.
Expected Outcome	● OrcaProbe should require significantly less number of configuring actions and/or need help from other tools.

VAL.3 - Process Speedup	
Criterion Overview	This criterion validates the process speed-up users achieve by using OrcaProbe compared to conventional methods.
Procedure	1. Using conventional methods, measure the desired values for a material and the measurement(s) that will be executed. Record the time it takes to complete the steps. 2. Carry out the measurement using OrcaProbe and record the time in the same manner. 3. Compare the difference in time spent and the time saved with OrcaProbe.
Expected Outcome	• OrcaProbe should speed up the process by at least 5 times.

Table 1: OrcaProbe Validation Test Results

(Pass / Fail / Incomplete [Incp.]) - (Prl)*: Partially

Test Scenario	Measurement Accuracy			Reconfigurability			Process Speedup		
PDMS PEDOT:PSS - Capacitors									
Sheet Resistance w/2-4 probe resistance	Pass	Fail	Incp.	Pass	Fail	Incp.	Pass	Fail	Incp.
Capacitance character. w/2 probe	Pass	Fail	Incp.	Pass	Fail	Incp.	Pass	Fail	Incp.
Impedance Spectros. w/2-4 probe	Pass	1-Fail (Prl)*	Incp.	Pass	Fail	Incp.	Pass	Fail	Incp.
Carbon Nanotube (MWCNT) - Resistors									
Sheet Resistance w/2-4 probe resistance	Pass	Fail	Incp.	Pass	Fail	Incp.	Pass	Fail	Incp.
Capacitance character. w/2 probe	Pass	Fail	Incp.	Pass	Fail	Incp.	Pass	Fail	Incp.
Impedance Spectros. w/2-4 probe	Pass	1-Fail (Prl)*	Incp.	Pass	Fail	Incp.	Pass	Fail	Incp.
Carbon Nanotube (SWCNT) - Transistors									
Gate Capacitance Character. w/ 3 probe	Pass	2-Fail	Incp.	Pass	Fail	Incp.	Pass	Fail	Incp.
Drain source resistance w/ 2 probe	Pass	Fail	Incp.	Pass	3-Fail	Incp.	Pass	Fail	Incp.
Transistor character. w/ output and transfer	Pass	Fail	Incp.	Pass	Fail	Incp.	Pass	Fail	Incp.

The failed cases in the validation are explained below according to their number labels:

1 - Impedance spectroscopy with 2 and 4-probes for PEDOT:PSS capacitors and Carbon Nanotube resistors partially failed accuracy category due to phase reconstruction not working as desired. The impedance amplitudes were still in the expected range, however, phase differences were off by more than 5%.

2 - 3- probe gate capacitance measurement accuracy in some cases is over 5%. This is due to the difficulty in measuring signals for the small gate capacitances of typical transistors. The test still functions properly, but the accuracy of measurement is lower due to hardware limitations.

3 - Drain source resistance measurement with 2-probes for Carbon Nanotube transistors have failed in the Reconfigurability category due to the dependence on an external power supply to bias the gate electrode. This was mainly due to no 3-probe measurement method covering this case and the only suitable method being a 2-probe one. It can still offer speedup and accurate readings, however does not provide the desired reconfigurability.

Appendix A. Team Roles

Name	Initials	Tech Lead	Management Lead
Aaron Loh	AL	Monitoring System	Team Liaison
Dipak Shrestha	DS	Switching Network	Inventory Manager
Idil Bil	IB	Software / GUI	Project Manager
Kerem Oktay	KO	Firmware / MCU	Document Manager
Peggy Yuan	PY	Driving System & Mechanical Design	Treasurer

Appendix B. Contributions

Section	Major Contribution	Minor Contribution	Author	Reviewer
1. Project Overview	AL	PY	AL	PY
2. Document Overview	DS	KO	DS	KO
3.1 Verification Plan Description	KO	IB	KO	IB
3.2 Test Plans	KO IB	AL DS PY	AL DS IB KO PY	AL DS IB KO PY
4.1 Validation Plan Description	KO	IB	KO	IB
4.2 Validation Criteria	KO IB	-	KO IB	DS