



TI USB Hub Test Tool: User Guide

Document Revision: 1.4

Date: October 1, 2012.

Table of Contents

1	INTRODUCTION	3
1.1	REQUIRED EQUIPMENT	3
2	USING THE TI USB HUB TEST TOOL	4
2.1	RUNNING THE SETUP PROGRAM	4
2.2	OPENING THE APPLICATION	6
2.3	GRAPHICAL USER INTERFACE (GUI)	7
3	AUTOMATIC TESTING.....	9
3.1	TEST ENVIRONMENT CONFIGURATION	9
3.2	TESTING PROCESS	9
4	IMPORTANT NOTES	13

1 Introduction

This document provides detailed information on how to use the TI USB Hub Test Tool. This software is a Windows-based manufacturing tool that will enable the user to test port functionality on a TI USB 3.0 Hub.

1.1 Required Equipment

The following is required in order to use this software:

- PC running Windows XP™, Windows Vista™, Windows 7™ or Windows 8™ (32-bit or 64-bit OS).
- Microsoft .NET Framework 3.5 installed on the test system.
- xHCI Compliant USB 3.0 Host Controller (See list of supported Host Controllers below)
- A Texas Instruments USB SS Hub.
- USB 3.0 devices to connect on each USB Hub port.

1.1.1 Supported xHCI Host Controllers

- If using Windows XP, Vista or Windows 7:
 - Texas Instruments TUSB73x0 Host Controller (VID 0x104C)
- If using Windows 8:
 - Texas Instruments TUSB73x0 Host Controller (VID 0x104C)
 - Fresco Logic xHCI Host Controller (VID 0x1B73)

* Support for more Host Controller Vendors coming up...

2 Using the TI USB Hub Test Tool

2.1 Running the Setup Program

Extract the setup program on a directory of your choice. Run the installer's executable file and let the installation wizard guide you through the installation process.

The TI USB Hub Test Tool requires Microsoft's .NET Framework 3.5 as a pre-requisite. The installer will guide you through this process in case there's no such .NET framework version installed on your system. Ensure that you have access to an internet connection since the installation will be performed through a web download directly from Microsoft's website (see Figure 2-1).

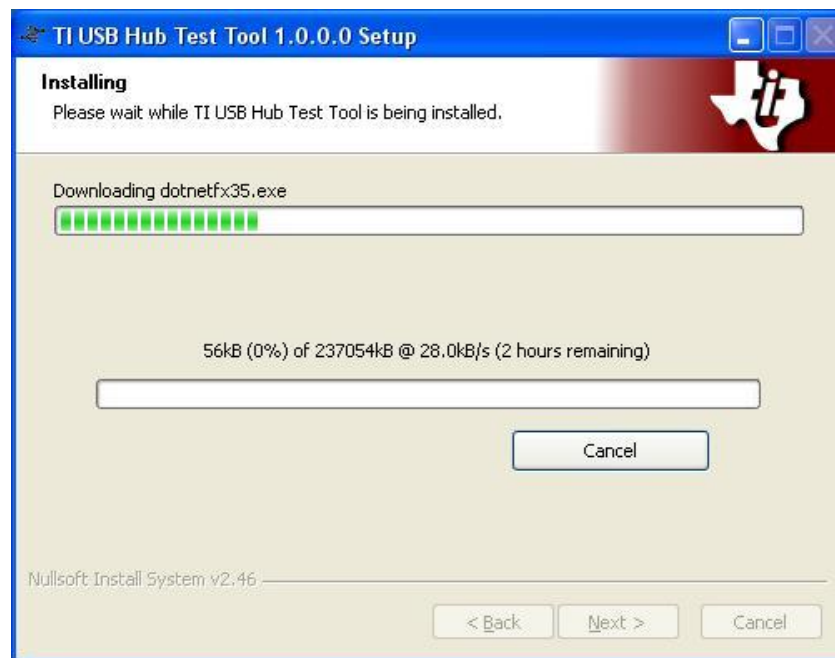


Figure 2-1 .NET Framework 3.5 Installation

After the .NET validation has finished, the installer will copy the necessary files to your local disk and will execute a driver co-installer that will install the "TiHwAccess" driver in your system. Depending on your system settings, you may get a warning message during the driver installation process. When prompted, click on "Continue Anyway" for XP or "Install this driver software anyway" for Vista/Win-7/Win-8 (see Figure 2-2 and Figure 2-3).

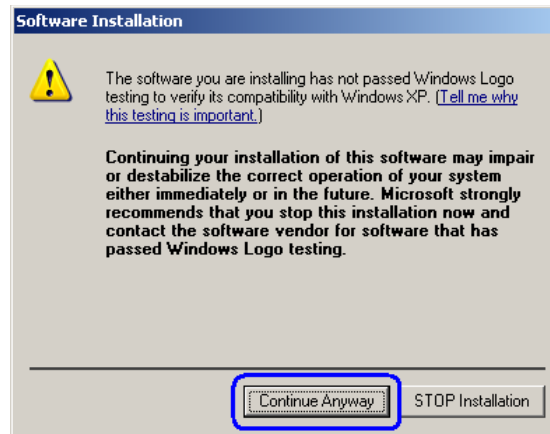


Figure 2-2 Driver Installation warning for XP

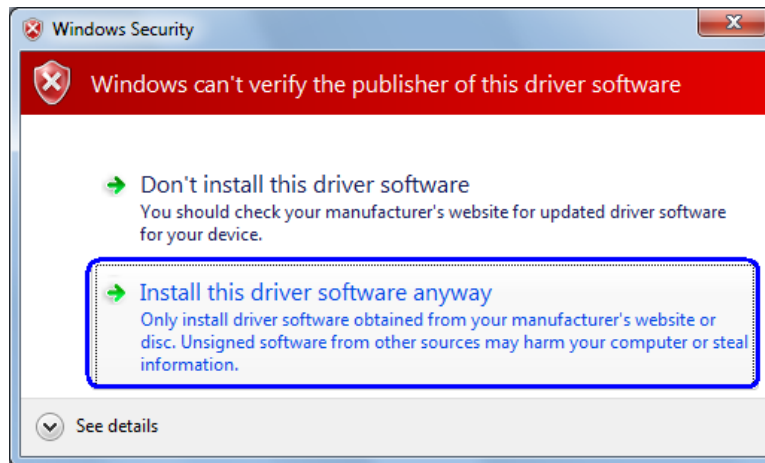


Figure 2-3 Driver Installation Warning for Vista/Win-7/Win-8

Simply follow the installation instructions to complete the installation process (Figure 2-4).



Figure 2-4 Installation Completed.

2.2 Opening the application

You can access the TI USB Hub Test Tool by clicking on the "TI USB Hub Test Tool" shortcut located on your desktop or through the Windows start menu under "Texas Instruments Inc → TI USB Hub Test Tool → TI USB Hub Test Tool"

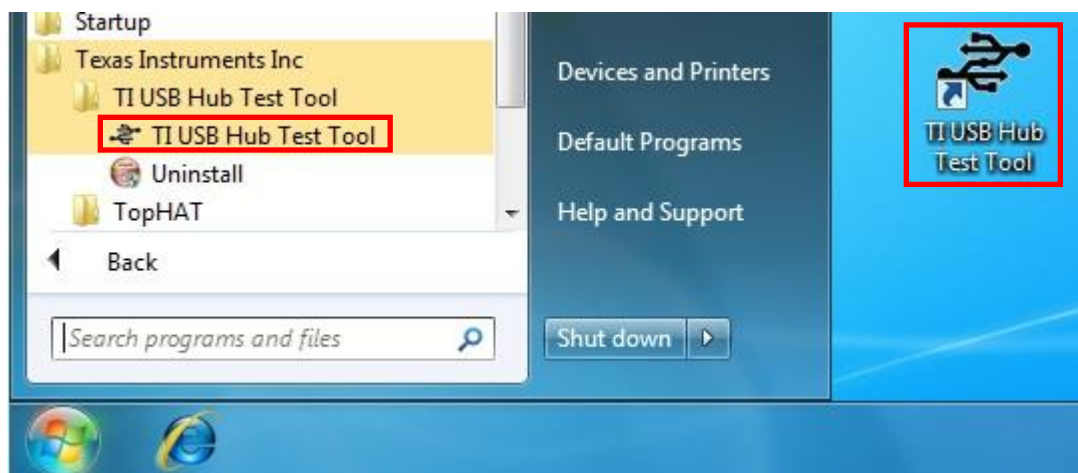


Figure 2-5 TI USB Hub Test Tool Access.

2.3 Graphical User Interface (GUI)

Once the application has been launched, the graphical interface in the figure 2-6 below will be displayed. In this section we will be detailing each one of its components.

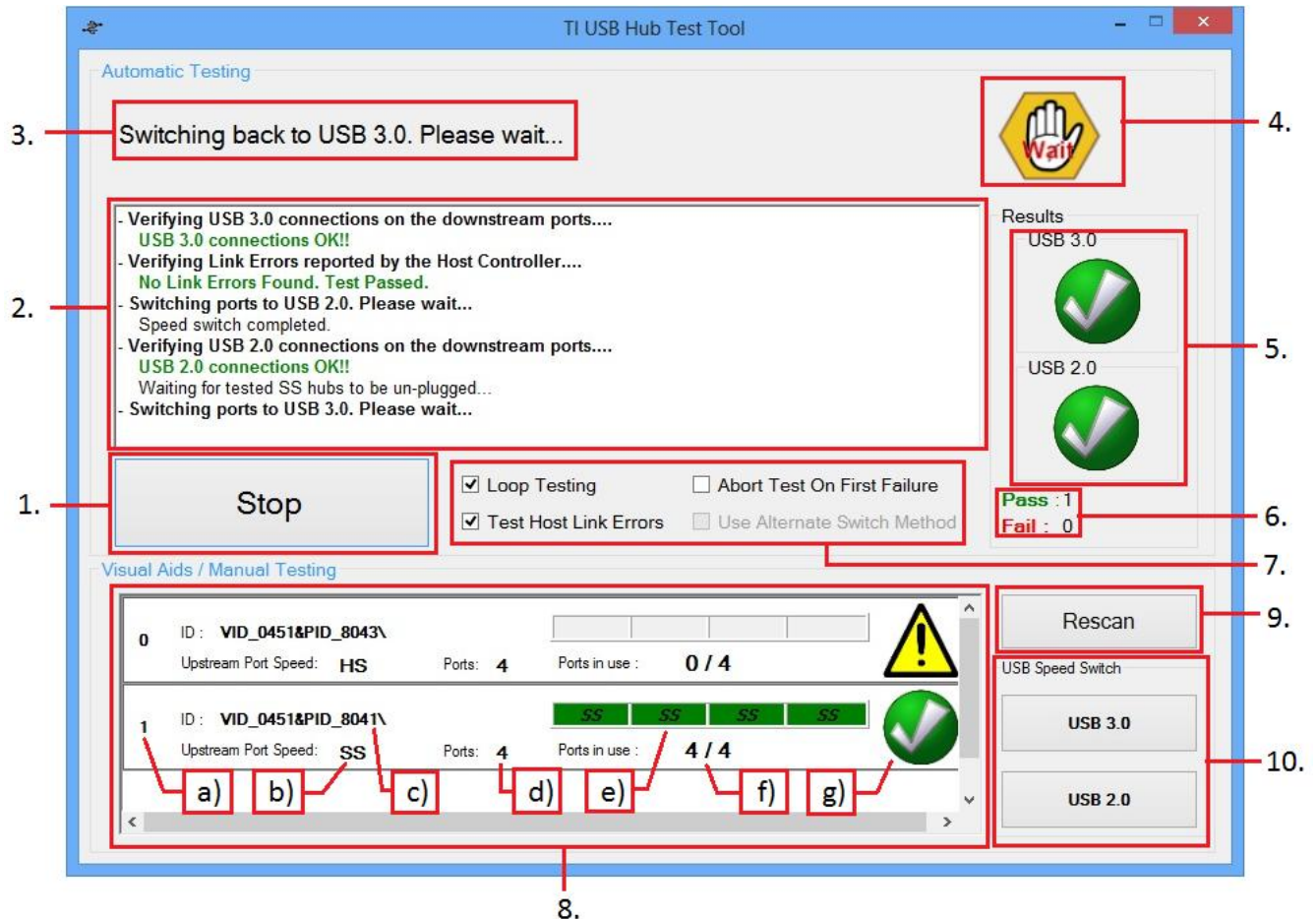


Figure 2-6 Graphical Interface.

1. **Start/Stop Button:** The Start button is used to launch the automatic testing process. Unless a critical failure is found, the test will be running on an infinite loop. Once the automatic test process is running, the button can be used to abort the ongoing process and stop the test.
2. **Test Log:** This text box will display the details of the tests being performed to the Device Under Test (DUT, in this case our SS USB Hub). Once the DUT is unplugged, the log will be erased to start logging the information of the next hub to test.
3. **Instructions Label:** Is used to indicate the instructions that must be followed by the user. Instructions are as simple as plug the DUT, remove the DUT and wait.

4. **Graphical Instructions Indicator:** Displays a drawing representing the instructions that must be followed by the user.
5. **Partial Results Graphical Indicators:** Each DUT is tested at two different speeds, Super-Speed (SS) and High-Speed (HS); these graphical indicators will represent the test result for each one on the tested speeds
6. **Test Results Counter:** Shows the number of times a DUT has passed or failed the tests being performed
7. **Test Options:** Enable the user to configure the automatic testing behavior:
 - Loop Testing: When checked the automatic test will run in an infinite loop that can be only interrupted by a fatal error or by pressing the Stop button. If it is not checked, it will only run once and automatically will be stopped after finishing. This option is enabled by default.
 - Abort on first failure: When checked, the automatic test process will stop when it finds the first failure. This option is disabled by default.
 - Test Host Link Errors: When checked, the automatic test process will check for the Link Errors reported by the xHCI Host Controller SS ports. These errors are usually good indicators of a problem on the connection between the host controller and the hub's Upstream port; although it doesn't necessarily mean there's a problem with the DUT. This option is enabled by default.
 - Use Alternate Switch Method: The automatic test process includes a process to automatically switch the DUT from SS to HS without the need to manually interchange devices on the Downstream ports or USB cables. This is achieved by manipulating the xHCI Host Controller registers; although, the behavior of the Host Controller might vary depending on the manufacturer. By checking this option, the application will use a secondary speed switch method we have identified works on different xHCI Host Controllers. This option is unavailable on Windows 8 and disabled by default.
8. **Hub Status Visual Aids:** These graphical components are visually representing a real time status of the DUT and displays relevant information about it. In consequence, it is extremely helpful to identify where and why a failure occurs.
 - a) The Hub index which identifies the number of TI Hub within the Hub's list
 - b) Upstream port connection speed; that is the connection speed between the USB Host controller and the USB Hub under test.
 - c) The USB Hub hardware ID that identifies this hub on the system.
 - d) The total number of ports in the DUT.
 - e) A graphical representation of the USB Hub ports occupancy and downstream ports connection speed.
 - f) The number of ports on the USB SS Hub (DUT) that are currently in use by a USB device
 - g) A graphical aid that helps to quickly identify when a downstream device is missing.
9. **Rescan Button:** Rescans the system looking for compatible SS Hubs to test and refreshes the Hub Status Visual aids.

- 10. USB Speed Switch Buttons:** Tweak the xHCI Host Controller registers to try to force the connected devices to switch to the desired speed. When switching to USB 3.0, the user must un-plug the USB devices and plug them back again to they can be re-enumerated at SS.

3 Automatic Testing

3.1 Test Environment Configuration

The automatic testing process requires a USB 3.0 device to be connected on each one of the SS Hub (DUT) downstream ports.

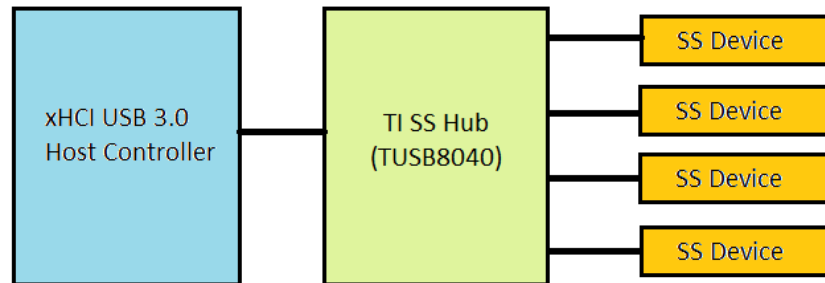


Figure 3-1 Required Hardware Configuration

3.2 Testing Process

1. Press the Start button. Notice that the button text will change to “Stop”, letting you abort the test process at any time.
2. The “Instructions Label” will prompt you to insert the SS Hub to test, and in accordance to this instruction, the “Graphical Instructions Indicator” shows a drawing of a USB cable with a green arrow. Now it is time to insert the DUT (USB SS Hub).

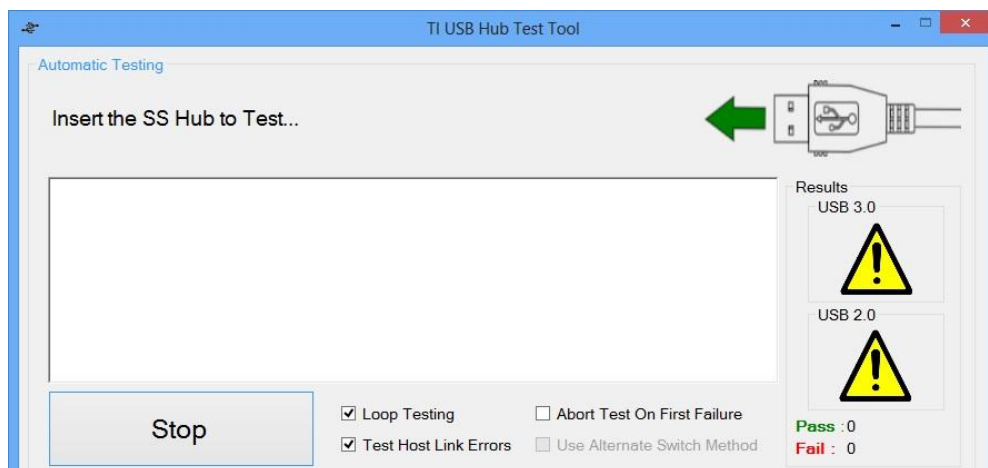


Figure 3-1 Automatic Test: Insert a DUT.

3. After plugging in the DUT, the “Instructions Label” will indicate that the SS connections are being verified, there’s no action required from the user at the moment, so the “Graphical Instructions Indicator” will display a drawing indicating to wait. Once the DUT has been fully enumerated by the system the tool will verify that:
 - a) There’s a device connected at each one of the downstream ports
 - b) Both the upstream port and all downstream ports are properly connected at SS.
 - c) The xHCI Host Controller hasn’t reported Link Errors (Only when the “Test Host Link Errors” option is enabled).

If no link errors are found, the tool will try to disable the SS connections in your xHCI Host Controller, forcing the connected devices to re-enumerate at HS, thus enabling the tool to test the HS connections on your Hub without the hassle of manually switching USB cables or devices.

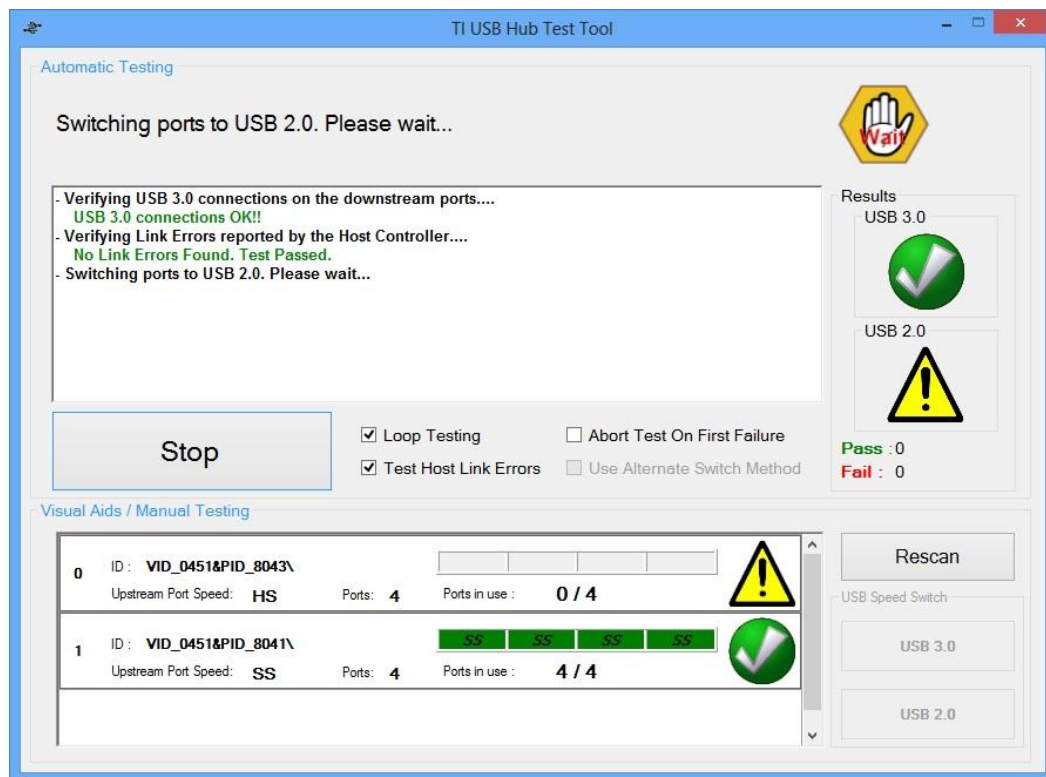


Figure 3-2 Testing USB 3.0 connections

4. Once the DUT and your downstream devices have been re-enumerated at HS, the tool will now proceed to verify the HS connections checking that there’s a device connected at each one of the downstream ports and that both the upstream ports and downstream devices are all enumerated at HS.

To complete the test cycle, you will be prompted to unplug your SS hub, in accordance to this instruction, the “Graphical Instructions Indicator” will display a drawing of a USB cable with a red arrow. Now it is time to remove the DUT (USB SS Hub).

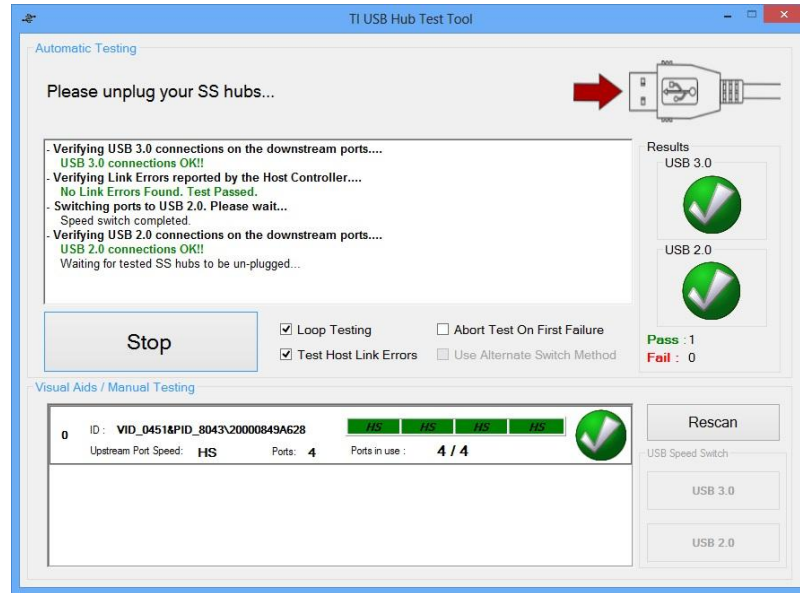


Figure 3.3 – Testing USB 2.0 Connections

Finally, once the DUT has been unplugged, the tool will re-enable the SS ports on your xHCI Host Controller, letting you go back to step 1 to test the next SS Hub.

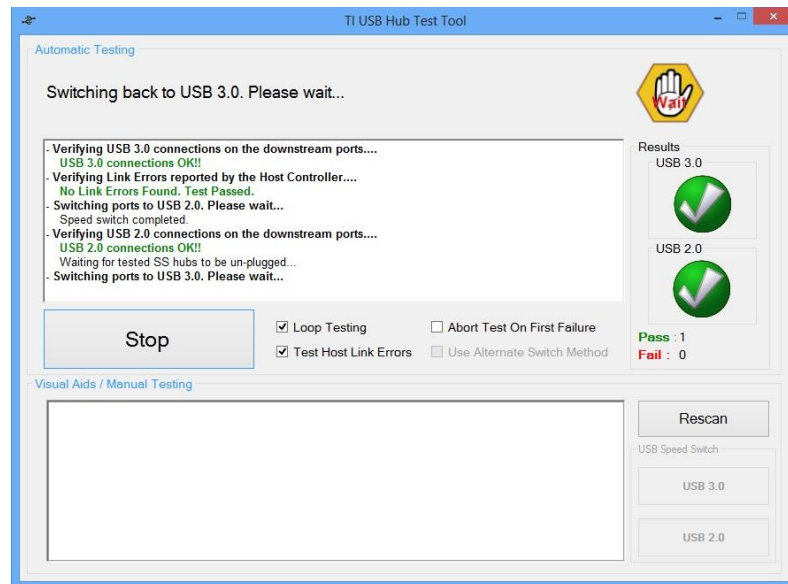


Figure 3.4 – Switching back to SS.

Both HS and SS tests must pass; otherwise it's considered a test failure. Once both tests are completed the test counter will be updated accordingly.

If a failure occurs during the automatic test process, the Test Log will provide you a detailed description of the failure and the possible reasons behind it.

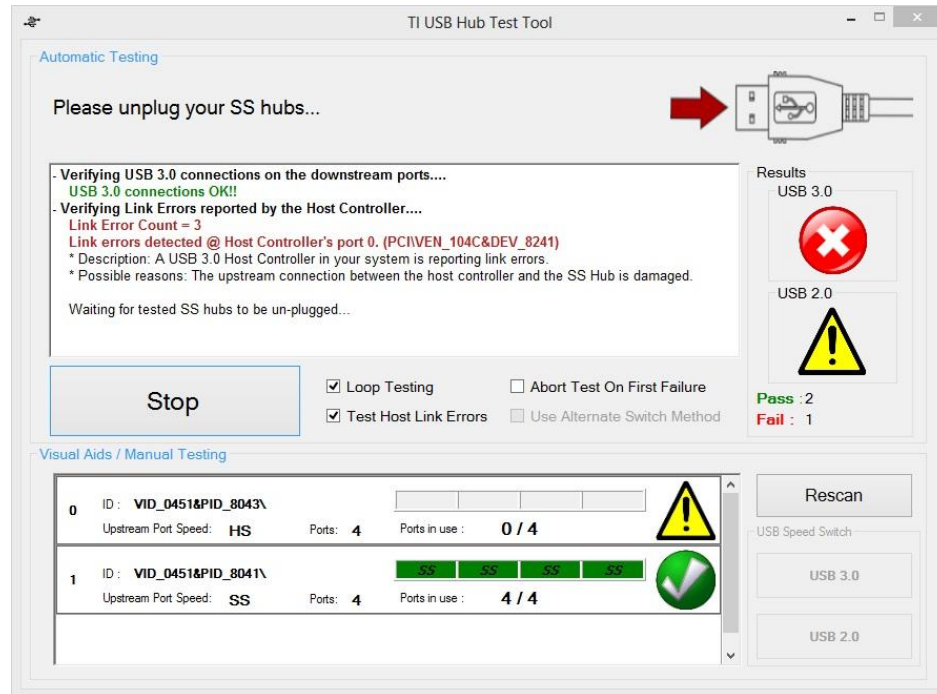


Figure 3.5 – Test Failure.

4 Important Notes

- Only TI manufactured USB hubs will be recognized by this utility.
- USB Hub and device enumeration might take considerable time depending on the device's features and especially if it's the first time that device is connected into the test system. Device enumeration time is reliant on Windows device enumeration, so device changes won't be reflected until the device is fully enumerated (i.e. no Device Manager yellow bangs on the device) by Windows.