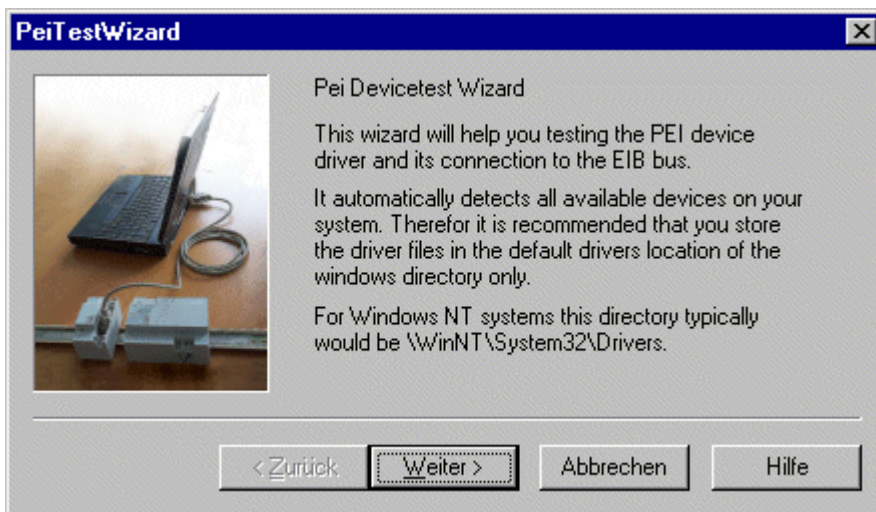


Introduction

When the connection of a Windows PC to the EIB via a serial cable does not work, various reasons are possible. The PeiTestWizard narrows these down by testing the serial adapter chipset and the communication with the local BCU.

Start



This page contains some general information about what the wizard does.

Usually, the driver files are stored at the correct position by a setup routine, but the hint may be considered if the wizard cannot find the driver.

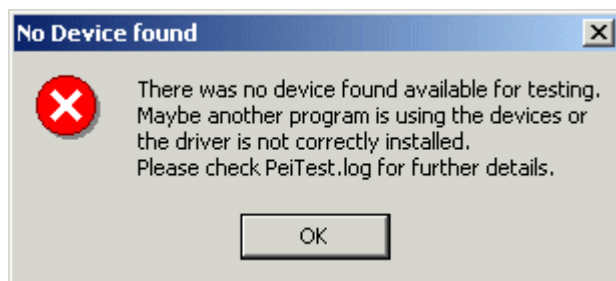
Device selection

PeiTestWizard



The window in the center of the screen shows all PEI devices found by the Wizard.

In **Windows NT** there typically is only one (maybe two) device(s) in the list, depending on the number of COM ports reserved for use by the PEI driver. This is because the PEI NT driver hogs "its" COM ports, which are otherwise reserved by the standard serial driver. Whenever a port is already used by another application, it will not show up as available in the list! The result (if no other port is free to be used) is the following message box:

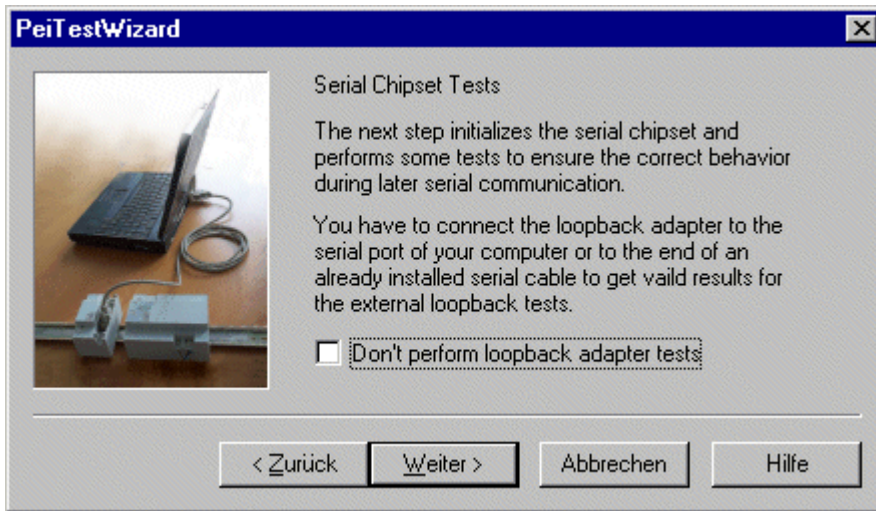


While in Windows NT the drivers are loaded at boot time, in **Windows 9x** the PEI driver is designed as a VxD (Virtual x Device). The VxD communicates with the serial port via an interface (VCOMM), which enables the driver to dynamically reserve the port. Therefore, in Windows 9x VxDs for all present serial ports which are not already used are shown, but only those connected to the EIB will pass the EIB tests.

At the moment, the PEI driver does not work together with **Windows 2000**. This is because Win2K implements a driver model different to WinNT/Win9x, called the **Windows Driver Model (WDM)**. In Win2K the PEI driver will be layered above the standard serial driver or any serial driver fully compatible with it.

Immediately start endurance test: [see Endurance Test](#).

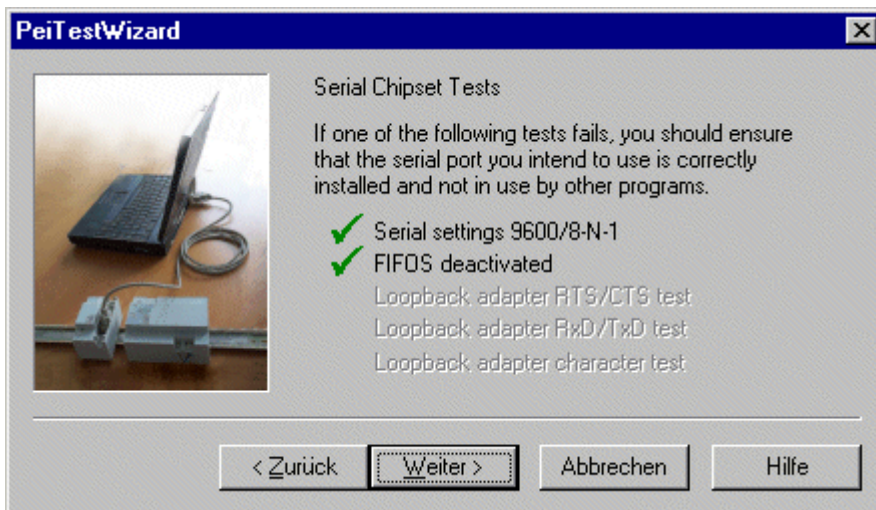
PeiTestWizard Serial chipset tests



The "serial chipset tests" tests – as its name implies – only test the settings of the serial port. If a serial loopback adapter is connected to the serial port, a loopback test can be performed.

In a **loopback adapter test**, the output lines RTS/TxD are connected to the input lines CTS/RxD, respectively, by an externally connected loopback adapter. The connection is tested, which means that faults in any of the input/output lines should be detected.

The following screenshot shows the results of a serial chipset test performed without the loopback test.



There are two things the Wizard has tested:

PeiTestWizard

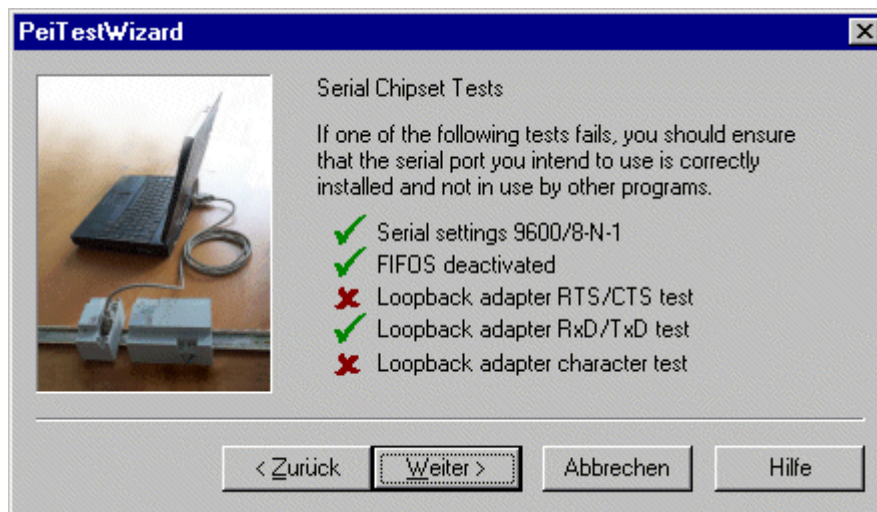
1. Serial settings 9600/8-N-1

This means that the serial port is configured for 9600 baud, 8 data bits, No parity and 1 stop bit (PEI16 protocol specification).

2. FIFOs deactivated

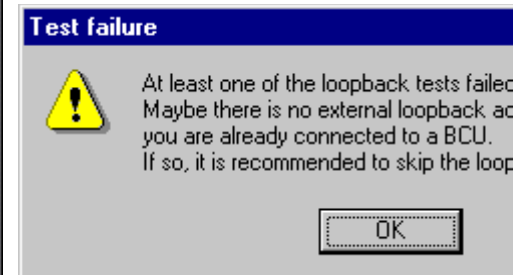
In Windows 9x the FIFOs (First-In-First-Out buffers. These buffers enable the port hardware to send/receive more than one character without being serviced by the driver; this feature can *not* be used for the PEI16 protocol) have to be switched off manually using the device manager. In Windows NT the driver can switch off the FIFOs by itself.

If the loopback test is performed *without using a loopback adapter*, but with connection to the local BCU, the result looks like this:



RTS/CTS: handshake lines.

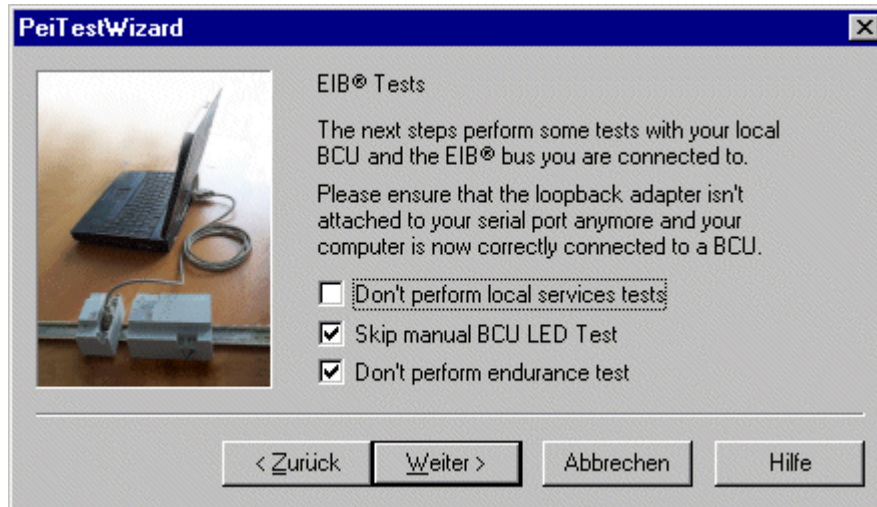
RxD/TxD: data transmission lines.



RTS is not connected to CTS, but the BCU responds to a character sent by the serial port, so the RxD/TxD test seems to be OK. But the returned character differs from the character sent, so the character test fails.

EIB tests

PeiTestWizard

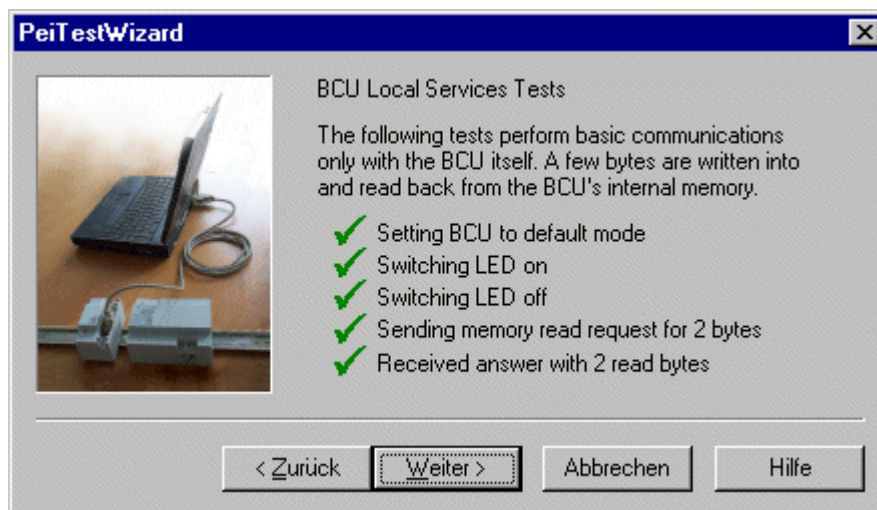


It should be noticed that, as mentioned in the screenshot above, all tests are performed using the local BCU only. The local BCU has to be connected to an EIB installation, but nothing more than a working power supply is expected.

The following tests can be (de-)selected:

- Don't perform local services test
By default, local services are tested; includes generic memory read and toggling the LED status.
- Skip manual BCU LED test
The manual BCU LED test allows the user to switch the BCU LED on and off.
- Don't perform endurance test
[See Endurance Test.](#)

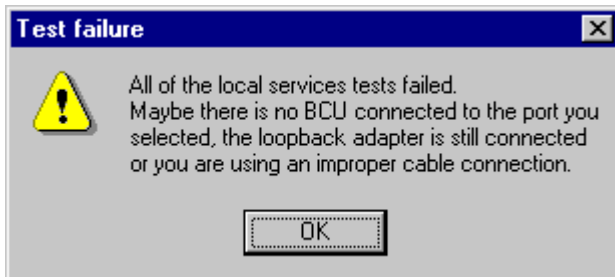
Local services tests



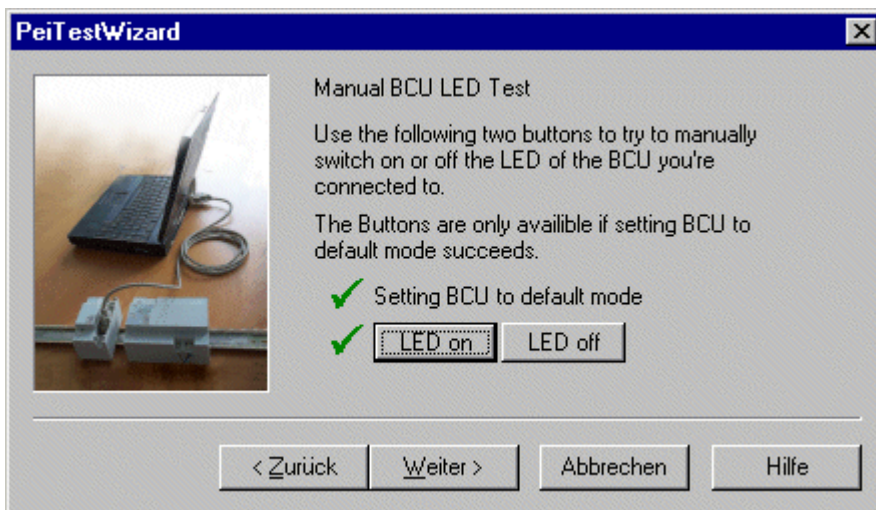
PeiTestWizard

The local services tests show whether or not the communication with the BCU is working.

If one of the local services tests fails, the test should be rerun. If all of the tests fail, the Wizard provides some hints to the possible reasons:



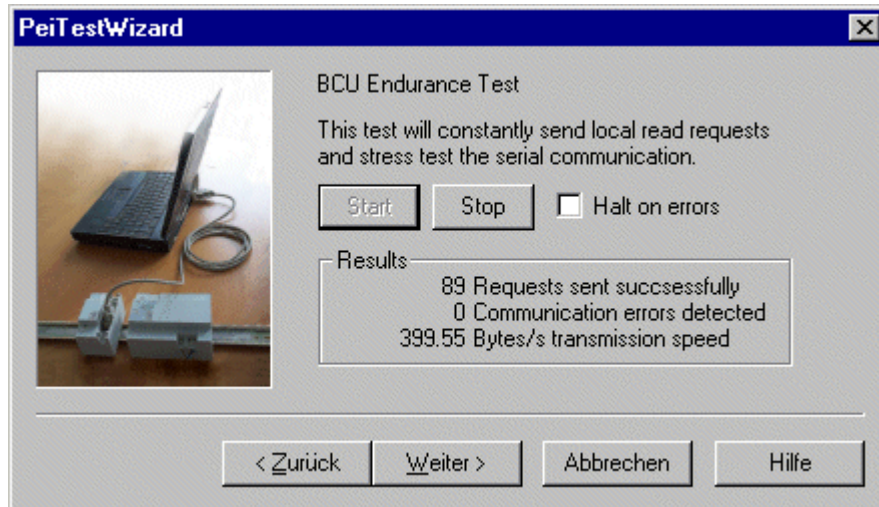
Manual BCU LED test



If the connection is OK, the BCU LED can be switched on and off. If the local services test was passed successfully, it is unlikely that this test will fail.

BCU Endurance Test

PeiTestWizard

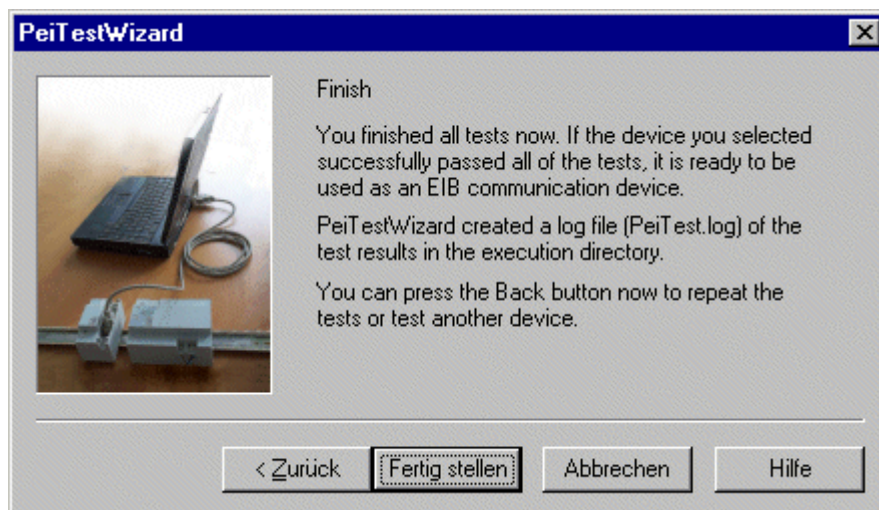


During the endurance test, read requests are constantly sent to the local BCU. The Results window shows some statistics about these requests.

When "Halt on errors" is checked, the endurance test is stopped at each communication error, giving the choice of resuming the test (keeping the statistics) or stopping it.

The transmission speed shown is averaged over time. It is affected by the CPU usage and limited by the transmission rate of 9600 bit/s and the reaction time of the BCU (yielding a maximum transmission speed of ~ 400 byte/s).

Finish



The last screen gives the choice of going back to previous screens or finishing the test.



PeiTestWizard

All of the results are written (appended) to a log file (PeiTest.log) which can be viewed using any standard editor.