How do I probe a net in EPIC?

There currently is no "probe" command in EPIC, but there is a simple PERL script contained in **$XILINX/userware/utilities** named **add_probe**. You may run this file to generate a script that can be executed in EPIC. You may also add the probe directly in EPIC.

Reference solution #2979: http://www.xilinx.com/techdocs/2979.htm

JTAGProgrammer issues a "Boundary Scan Integrity" error while executing Boundary Scan instructions on a chain of Boundary-Scan-capable devices. What does this error mean?

Chain integrity must be assured before the results of the Boundary Scan testing can be relied upon. A chain may have many faults that can interfere with the integrity of the chain, such as:

✔ A component in the chain may be missing, dead or incorrectly loaded.

✔ A component in the chain may have a broken connection on one of its TAP pins.

✔ A TDO to TDI connection between two components could be shorted to another node.

✔ A component in the chain may not have sufficient power supply.

✔ System or clock noise could cause a component in the chain to jump TAP states.

✔ A combination of all the problems described above may exist in a chain of components making it very difficult to diagnose.

JTAGProgrammer performs the integrity test by relying on the device's ability to set a fixed data pattern in its Instruction Register in the **Capture-IR** state of the TAP Controller. This concatenated data can then be shifted out on the TDO pin of the last device. The value of **attribute INSTRUCTION_CAPTURE** in the BSDL (Boundary Scan Description Language) file for each device in the chain shows what this data will be.

Additionally, the IEEE standard 1149.1 mandates that the two LSBs of this data should be "01".

For a simple test case, assume that there are three devices in the chain and the Instruction Registers are only two bits wide. In this case, for a good chain, we will see "010101" coming out on the TDO pin of the last device.

As an example of a bad chain, assume there is something wrong with the connection of the first device in the chain. This device then will not output "01" on its TDO pin. Instead, it may push a value such as "11". We will see "110101" coming out on the TDO pin of the last device. By examining the data stream closely, we can see that the last two devices in the chain are connected properly and there is a problem with the first device in the chain.

JTAGProgrammer will issue a **Boundary Scan Integrity** error if receives unexpected data at the TDO pin of the last device. The integrity failure can occur for a single device as well as for multiple devices in the chain. ◆

37