An Unobtrusive Debugging Methodology for

Actel AX and RTAX-S FPGAs

Jonathan Alexander
Manager, Applications Consulting
Actel Corporation
April 26, 2004

Contact info:
(650) 318-4535
jonathan.alexander@actel.com

1. Abstract

Design debugging is a particularly important process for high-reliability IC design engineers to go through. One failure in a production system could have devastating consequences. Everything that can be done to mitigate the risk of field failure within reasonable means must be done. There are several commonly used techniques that significantly reduce the impact that a failure in an IC could have on a system. Such techniques include: triple redundancy (TMR) and error detection and correction (EDAC). These techniques can prevent logic errors due to glitches or single event upsets (SEU) within a single IC or among multiple IC's. Although these techniques significantly improve reliability of IC's, there may be incorrect logic in the circuit due to timing errors, bugs in the source code, system noise, or any combination of the above. This is why extensive system-level testing is critical. In order for the system level testing to be effective, all observed failures must be fully understood so that preventative measures can be taken to eliminate the possibility of a reoccurrence in production. Custom IC's such as ASICs may have built-in test circuitry such as BIST and manufacturers provide test vectors that provide high test coverage for the device. BIST and test vectors are created specific to the design in such a way that any errors can be pinpointed within the device so root cause can be determined. FPGAs do not have the benefit of test circuitry that is built for the specific design that is implemented since they leave the manufacturer as a generic device with no functionality. On the same note, the manufacturer does not provide test vectors for the device since there is no knowledge of what the device will do. Therefore, FPGA manufacturers need to implement generic test methods that will work for any design.

Today's reprogrammable FPGAs rely on soft test circuitry to be implemented in the design. A fixed number of probe points are pre-determined and routed into the design. This has several key disadvantages. First, the test circuitry consumes device resources. The more points that are selected for probing, the more device gates are needed. Second, the selected probe points are routed to the selected signals. The probe adds loading to the signals that are being observed. Third, if different nodes are selected to observe, the design has to be re-routed to reposition the probe assignments. By doing so, the original failure mechanism may disappear completely because different internal resources are used, thus making it practically impossible to locate the root cause with certainty for many possible failure mechanisms

Actel's AX and RTAX-S antifuse FPGAs contain an unobtrusive built in probe circuit that provides 100% real-time observation inside the FPGA. This unique probe circuit will allow up to four internal nodes to be simultaneously driven to external pins of the device. Virtually any node can be observed at any time while the FPGA is operating on the application board. Reflecting on the disadvantages of reprogrammable FPGAs available today, first, no device internal resources are consumed by this probe circuitry. It is driven by JTAG commands through the JTAG Test Access Port (TAP), and the output is delivered through flexible I/Os, therefore the impact on I/Os is also minimal. Second, the internal probe nodes are separately buffered from the actual routes within the device, so they do not add any loading to the selected signals. Third, since the probe circuit is built-in and has access to virtually any internal node, re-routing is not necessary, so the failure that is being chased will not be affected by changing observation points.

This paper will expand on the advantages of the built-in probe circuit on every AX and RTAX-S FPGA, and will give the reader knowledge of the probe circuit architecture. Specific implementation suggestions, requirements, and limitations for the AX and RTAX-S products will be covered as well as a general debugging methodology for probing these devices. The paper will also provide some of the differences in AX and RTAX-S for users of earlier Actel antifuse FPGAs such as the SX-A and RTSX-S families.