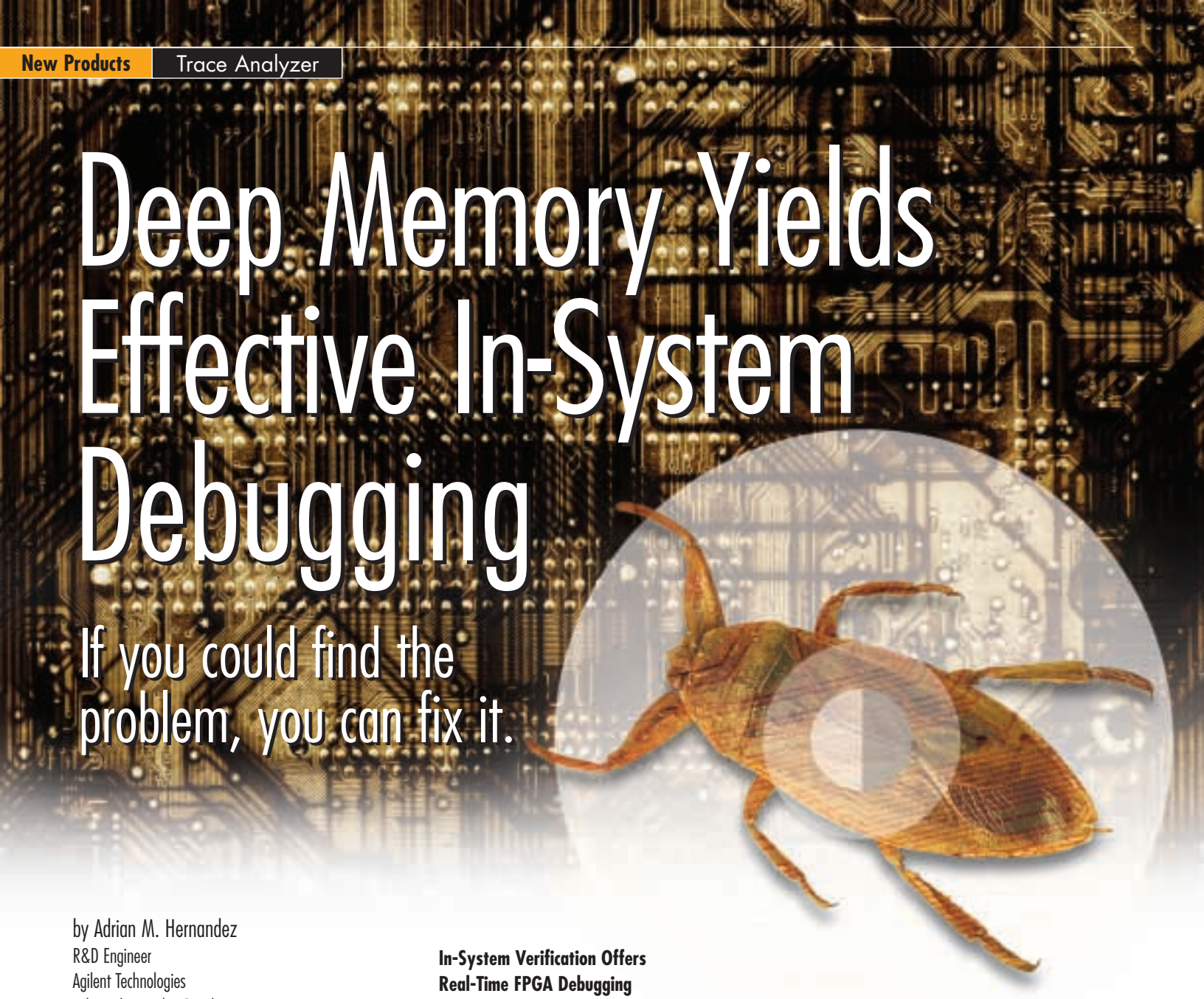
Deep Memory Yields Effective In-System Debugging

If you could find the problem, you can fix it.



by Adrian M. Hernandez
R&D Engineer
Agilent Technologies
adrian_hernandez@agilent.com

FPGAs that can incorporate whole systems have definitely made in-system debugging more challenging. On-chip debugging methodologies aren't always adequate to provide trace memory that is deep enough to capture a sufficient event history. Plus, critical internal nodes may not be readily accessible to external logic analyzers. A new solution comprising Xilinx ChipScope™ Pro tools, the Agilent FPGA Trace Port Analyzer, and the Agilent Trace Core combines the key advantages of internal and external logic analysis. Virtex™-II and Virtex-II Pro™ users now have a solution that combines the best of on-chip debugging with high-speed, deep, external trace using a limited number of pins.

In-System Verification Offers Real-Time FPGA Debugging

Although simulation continues to play an important role in verification of complex FPGAs, in-system verification provides strong complementary value. The primary advantages of in-system verification are that it runs at real-world speeds, enjoys the benefits of real-world stimulus, and has real-world modeling accuracy.

In-System Today

Logic analyzers remain the dominant tool for in-system debugging, with internal nodes routed to the pins. This manual process consumes significant routing resources and – most important – precious pins, but it does offer powerful triggering capabilities, deep memory, and time-correlation. Plus, the logic analyzer can be used for other tasks.

Some designers prefer on-chip debugging that uses an internal logic analyzer, such as ILA (Integrated Logic Analyzer), to develop an internal trace. Here, a logic analysis core is inserted into the FPGA design, and block RAM is used to store resulting traces. JTAG (Joint Test Action Group) is used to set up the logic analyzer and to move the trace buffer from the FPGA to a PC for analysis. The popularity of this emerging method derives from two factors:

- It requires no additional pins outside of JTAG.
- The tools are inexpensive.

Shallower trace depths, triggering that is more limited than with external analyzers, and the lack of time correlation are the primary tradeoffs of this methodology.

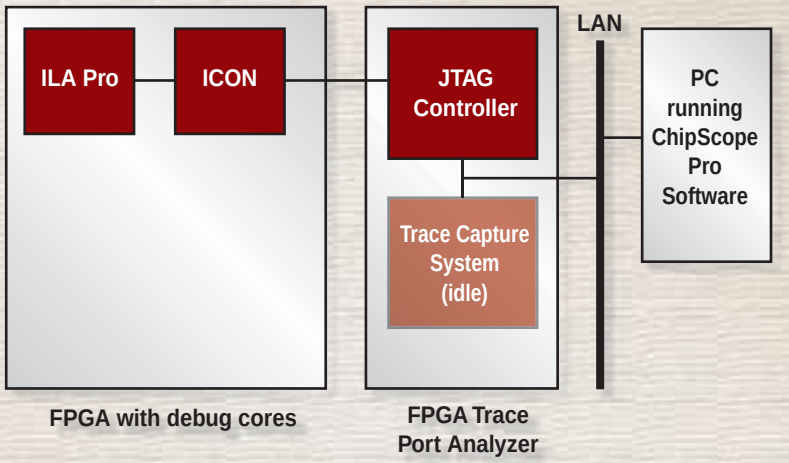


Figure 1 - FPGA Trace Port Analyzer connection to ILA Pro

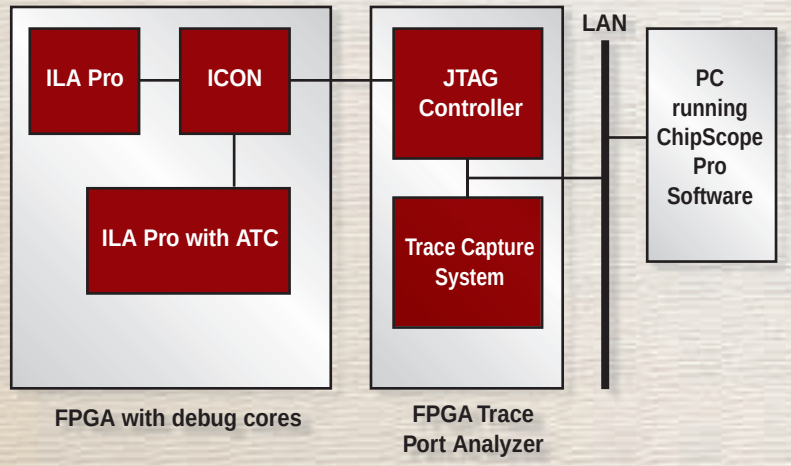


Figure 2 - FPGA Trace Port Analyzer connection using deep external trace memory

The In-System Solution with Deep Memory

The Xilinx/Agilent collaboration has produced a solution that offers the real-time/real-world benefits of in-system debugging with deep trace depth and enhanced triggering. The following describes the different components of this solution.

The Agilent E5904B Option 500 FPGA Trace Port Analyzer provides up to two million states of trace depth for each signal probed, at acquisition speeds up to 200 MHz. This is roughly 60 times deeper than the maximum trace depth offered by ILA Pro (32K) using block RAM. The additional trace depth is especially beneficial in capturing elusive events where symptom and cause may be separated by a long period of time. Another benefit of external trace storage is that it allows you to retain internal FPGA memory for the design instead of dedicating this valuable resource to debugging.

Reduces Dedicated Debugging Pins

The ChipScope Pro 5.1i software ships with a version of ILA Pro connected to an Agilent Trace Core (ATC). The ATC uses time division multiplexing to reduce the number of pins required to pass trace information to the FPGA Trace Port Analyzer for storage. With time division multiplexing, the internal data is accelerated, so a wide bus can be sent out on a few pins. The choices for acceleration, 1x, 2x, and 4x, represent the number of internal nodes sent through a single pin. While single-ended signals can be driven at 200 MHz on the pins, an internal circuit may run at 50 MHz. In this case ATC would produce a 4:1 pin compression ratio. This pin compression translates to being able to view up to 75 signals using just 20 pins.

High-Speed LAN-Based Cable Capabilities

The FPGA Trace Port Analyzer consists of two blocks: a trace acquisition sub-system, and a JTAG control sub-system. The JTAG controller provides a high-speed LAN cable interface between ChipScope Pro and ILA blocks (Figure 1). The controller, which can run up to 30 MHz, is used to configure FPGAs, set ILA triggers, and read back the stored trace data from the ILA block RAM control sub-system. This enables the FPGA Trace Port Analyzer to work with stand-alone ILA Pros, or a combination of ILA Pro with ATC ILA Pro core and one or more ILA Pros (Figure 2).

The Agilent Trace Port Analyzer lets you debug FPGAs remotely via a LAN, which means you can drive ChipScope Pro from your desk and control a design board located in a remote lab. This feature can be quite powerful, especially in conditions where many designers share a single prototype.

Connecting the FPGA Trace Port Analyzer to the Target System

The AMP MICTOR (Matched Impedance Connector) is designed into your target system via a connection with an Agilent Trace Port Analyzer. The MICTOR is a high-speed board connector capable of operating at clock rates above 200 MHz; it has a predefined pinout for the Agilent Trace Port Analyzer (Figure 3). This particular connector and pinout is compatible with both the IBM PowerPC™ 405 CPU trace connector and the Agilent logic ana-

lyzer connector. Thus, three instruments can use the same connector for debugging, one instrument at a time – a powerful advantage, especially when the FPGA under design is a Virtex II Pro device.

Debugging an FPGA in the Context of a Larger System

When debugging the FPGA in-system, it is often necessary to time-correlate FPGA events to other system events. The Agilent Trace Port Analyzer enables you to determine quickly whether your FPGA is operational. Using a design as simple as a counter connected to an ILA Pro with Agilent Trace Core, you can validate the FPGA programming and I/O interface in one step. Issues such as JTAG chain connections, FPGA pin configurations, non-functioning system clocks, and stuck traces, can be identified very simply by using the trace output to monitor the activity of the FPGA inputs.

The most complex FPGA malfunction occurs at the PCB (printed circuit board) level. The malfunction can come from a variety of devices or conditions external to the FPGA. For example, when an external processor is used, the FPGA must be able to work appropriately with the processor bus and handle all the additional devices on the bus. But because there can be many real-time situations that occur on a processor bus – interrupts, long burst cycles, and so on – it is difficult to simulate such a situation in software. In fact, sometimes the errors on buses or system boards are not logical, but physical. For this reason signal integrity issues, such as cross-talk, are usually difficult to simulate in software but very apparent in hardware.

Check for Signal Integrity

To facilitate PCB and system measurements, the Agilent FPGA Trace Port Analyzer has two ports, "Trig Out" and "Break In." The Trig Out port is an output port that signals other instruments, such as Agilent oscilloscopes and logic analyzers (performance has been validated with Agilent instruments only), to complete their measurement. The Break In port is an input port other instruments use to signal the FPGA Trace Port Analyzer to complete its measurement. The combination of the Trig Out and Break In ports enables you to make various complex measurements, such as checking for signal integrity issues on data lines.

Target Header Pin-Out for the MICTOR Connector

No Connect*	1	2	No Connect*
No Connect	3	4	No Connect*
No Connect*	5	6	ATCLK
No Connect*	7	8	No Connect*
No Connect*	9	10	No Connect*
TD0	11	12	Vref
No Connect*	13	14	No Connect
TCK	15	16	ATD19
TMS	17	18	ATD18
TDI	19	20	ATD17
No Connect*	21	22	ATD16
ATD15	23	24	ATD7
ATD14	25	26	ATD6
ATD13	27	28	ATD5
ATD12	29	30	ATD4
ATD11	31	32	ATD3
ATD10	33	34	ATD2
ATD9	35	36	ATD1
ATD8	37	38	ATD0

*Pins 1, 2, 3, 4, 7-10, 13, and 21 ***must*** be true no-connects. Pins 1-4 are driven when a logic analyzer is connected to the target system through the header connector. Pins 7-10, 13, and 21 are driven by the Trace Port Analyzer.

For designs with less than 20 trace data pins, any unused ATD pins ***must*** be connected to ground.

Figure 3 - FPGA debug connector

To check signal integrity, set up an oscilloscope on the suspect data lines. Next, add an ILA with ATC into your design on the data lines being probed by the oscilloscope. The FPGA Trace Port Analyzer's external port, Trig Out, is connected to the oscilloscope via a cable. The oscilloscope is then configured to stop its measurement when the port out signal from the FPGA is asserted. Now that you have completed this setup, you make your measurement. To do this, first start the oscilloscope. With the oscilloscope running, and the trigger on the ILA with ATC set to the bad data, you then begin the measurement on the FPGA Trace Port Analyzer. When the FPGA Trace Port Analyzer triggers, it will assert the Trig Out signal, which, in turn, signals the

oscilloscope to stop its measurement. Once stopped, you can inspect the oscilloscope waveform, going back in time to where the suspect data can be found. This measurement enables you to determine the root of a signal integrity issue.

Conclusion

The Agilent FPGA Trace Port Analyzer, combined with ChipScope Pro tools, is an affordable solution that enables effective in-system debugging. The combination of these two powerful tools gives FPGA designers internal node visibility during in-system debugging. It gives FPGA designers flexibility to take wide, shallow 32K state deep traces or narrow 2M state deep traces. It also provides remote debugging through the network capabilities of the Agilent FPGA Trace Port Analyzer. These features, along with the FPGA Trace Port Analyzer's ability to work with other Agilent instruments, make a powerful solution for debugging Xilinx FPGAs in-system. Σ