As the complexity of electronics for airborne applications continues to rise, an increasing number of applications need to comply with the RTCA DO-254/Eurocae ED-80 standard for certification of complex electronic hardware, which includes FPGAs and ASICs.

The DO-254 standard requires that for the most stringent levels of compliance (levels A and B), the verification process for FPGAs and ASICs must measure and record the verification coverage by running tests on the device in its operational environment. What this essentially means is that verification engineers and designers need to compare the behavior of the physical outputs of the device on the hardware device pins with their corresponding RTL model simulation results. In addition, the standard requires running robustness tests on the interface pins. The robustness testing is accomplished by forcing abnormal or slightly out-of-spec behavior on the device and ensuring that it is able to deal with this behavior without catastrophic results.

These requirements become especially challenging for high-speed interfaces, such as DDR3 or PCIe, because it is not possible to create and observe the abnormal behavior when an FPGA is connected to the regular operational interfaces. For example, when a real memory is connected to a DDR model, there is no way to control the DDR behavior and use different kinds of DDR memories. As for robustness testing, there is no way to test incorrect behavior of a memory while connected to a real memory, because the real memory does not allow for the type of error injection desired. It is also impossible to test different timing behavior of the high speed interface signals.

To overcome these challenges, Verisense developed a new approach, the Advance Validation Environment (AVE), which makes it possible to comply more completely with the DO-254 requirements for in-hardware testing of high speed interfaces at the pin level and enables users to easily run an array of robustness tests. Based on the Universal Verification Methodology (UVM), this novel approach involves the migration of a UVM testbench and verification environment into an FPGA operational environment. Applying UVM concepts and components to target-hardware simulation enables reuse of all simulation runs for the in-hardware testing. It also provides the ability to monitor all FPGA interfaces, including high-speed interfaces, at the FPGA pin level.

MEETING DO-254 STANDARDS

The goal of DO-254 certification is safety. No shortcuts can be taken. It is mandatory to prove the design correctness of an FPGA by verifying its entire feature set. High-speed interfaces are complicated interfaces which are usually linked to the main functionality of a specific FPGA. Thus the comprehensive testing of these interfaces must be a basic requirement in the process of verifying the design correctness of any device.

Therefore, an FPGA using a high speed DDR interface, for example, cannot be properly validated solely by connecting it to a standard DDR device and testing read/write operations, because it does not allow testing and verifying the FPGA behavior in abnormal conditions.

The best way to qualify Level A and Level B DO-254 devices is to compare the physical device outputs to the simulation model results. Doing this comparison for complex FPGAs is a complicated technological challenge. Adding high-speed interfaces makes it monumental. While in simple devices one can use scopes and logic analyzers for monitoring and comparing hardware signals, doing so for complex devices is not viable.

In the avionic industry, the need for completely bug-free designs is similar to other industries; however, such need arises from ensuring there are absolutely no critical safety issues. Since FPGAs in the avionic industry can rise to the level of ASIC complexity, there becomes a necessity to innovate an advanced hardware validation approach to cover all the safety requirements.

The new AVE approach suggested by Verisense for providing a comprehensive solution that can qualify Level A and Level B devices is based on applying the concepts of constrained random verification in the simulation world, to the real hardware environment. Based on advanced pre-silicon verification methodologies that will be introduced, the article will present a new approach for hardware testing that makes it not only possible but also straightforward to
comply more completely with the DO-254 requirements for real in-hardware testing including easily running an array of robustness tests.

ADVANCED HARDWARE VALIDATION ENVIRONMENT

At the most basic level, the AVE methodology implements the UVM verification environment in a unique testing platform. The main idea behind the UVM is that the verification environment is a sophisticated software machine that emulates the FPGA or ASIC real-life system environment.

**Figure 1: The UVM Pre-Silicon Verification Environment**

Figure 1 provides an overview of a UVM environment. In its simplest form each of the DUT interfaces is connected to an agent. Each agent is made up of a number of blocks, the most important of which is the sequencer, which coordinates the feeding of transactions (stimulus) into the driver. The driver converts these transactions, which are written at a given level of abstraction, to a lower level of abstraction according to the protocol being verified.

The low level block is the Bus Functional Model (BFM). This block is responsible for toggling the relevant interface according to its protocol with the relevant transactions which were received from the driver. The monitor is the component responsible for monitoring the interface buses and reporting all the transactions which occurred on the interface to the reference model block.

The reference model generates the expected transaction data on the output interfaces as a result of the input stimulus. Both the expected transactions and the monitored output transactions are sent to a scoreboard for comparison.

The UVM provides the ability to randomly generate all the possible scenarios according to constraints which limit the generation into real and possible scenarios; this is why it is known as constrained random verification. Running all random scenarios results in 100% functional coverage.

Although AVE was built from the ground up, as can be seen by comparing Figures 1 and 2, the AVE post-silicon environment is a direct descendent of the UVM environment. This is made possible by the coupling of Questa® with the AVE platform. In this scenario, the UVM agents are swapped out for Verisense emulators. The emulators are memory models implemented

**Figure 2: AVE Advanced Post-Silicon Validation Environment**
as Verisense IP, allowing, for example, a real DDR memory to be emulated. Like the UVM agent blocks, the emulator includes sequencers, drivers, and monitors, so AVE can emulate all the interfaces exactly as in a simulation. The same reference model is used and a scoreboard test-report generator summarizes results. In addition, voltage, clock, and signal control blocks have been added to qualify the DUT against tolerance specifications; this could not be done with even the most advanced UVM verification environment. A further benefit, is that the AVE hardware architecture is modular, allowing a high degree of reuse between projects.

The AVE testing platform architecture is divided into two main components: software and hardware.

The main function of the hardware is to implement the low-level components of the verification environment — basically the BFM and the monitor. In addition, through the use of high-speed interface emulators that connect to the DUT, the user is given full control of testing all interfaces with maximal flexibility.

All of the DUT interfaces, including high-speed interfaces, are connected to Verisense emulators, which are able to function as BFMs as well as monitors and emulate the full functionality. The emulator effectively has complete visibility of the DUT high-speed input and output pins, and it provides full testing capabilities, ranging from simple protocol checkers to complicated robustness testing. The hardware emulator uses its own memory to store both input data information (BFM) and the recorded output data (monitors).

The requirement for robustness testing on all interfaces includes controlling timing of signals, voltage levels, and in the case of a memory, also changing the actual data and data sequences. The AVE solution is able to provide extensive support for robustness testing on all interfaces, including high-speed interfaces, because the DUT pins connect directly to the hardware emulators; thus, the environment not only monitors activity on the pins but also can inject data and create abnormal situations. The emulators can control the timing on all interfaces or any subset thereof. It also has the ability to modify the latency of the protocols, the frequency, and even the voltage level of a given interface. All the interface electrical and logical values can be modified within the interface specification or beyond the specification limits for verifying real abnormal situations.

The software implements the high-level components of the verification environment, including the input generation, reference models, and off-line comparisons between the behavior of the DUT in the verification environment and its behavior on the final target hardware.

The software is preprogrammed via automatically generated configuration files with the relevant setting of the testing environment and the DUT hardware. Based on this information, the software configures the validation environment hardware and sets the clocks and voltages to the DUT as required for each test.

The steps in the software test flow are identical for both regular and high-speed interfaces and are as follows:

1. The software simulation environment generates Value Change Dump (VCD) files from the waveforms of the verification tests. The VCD file is an industry standard file format containing waveform information that is generated from the simulation environment, regardless of the verification methodology used in the software simulation. The tester software parses the VCD files and generates two sets of vectors. The first is used as input vectors to inject at the DUT pin level on the hardware platform. The second is used as the expected results vectors for comparing with the hardware tester results at the end of the process.
2. The hardware tester injects the input vectors onto the pins of the DUT, running the test that was previously run in the software simulation environment.

3. While the test is running, the tester monitors and records all the DUT output pins.

4. Once the test is completed, the recorded pin behavior on the tester is processed by the software.

5. The actual recorded pins waveform is then automatically compared with the expected results from the software simulation. Any mismatches are flagged and reported. If there are no mismatches, the test has passed successfully.

6. All appropriate log and test result files are then generated automatically for documentation and traceability purposes.

**High-Speed Interface Example - DDR 3**

In 2014, Verisense provided the VS-254 FPGA verification tester tool to multiple avionics customers to assist them in the certification of their DO-254 Level A and B products. The VS-254 is the first, and remains the only, solution that provides the complete required DO-254 functionality for high-speed interfaces, including pin-level verification on all pins and full robustness testing on all high-speed interfaces.

The VS-254 platform contains a complicated DUT FPGA that has two DDR3 interfaces. The DUT FPGA DDR interfaces are connected directly to three other FPGAs and not to DDR memories. The FPGAs also include the DDR emulators.

The VS-254 includes many enhanced features to reduce the verification effort and to make the engineering effort predictable. These include a high degree of reuse between the simulation environment and the hardware verification, built-in support for regression and waivers, and the VS system simulation tool for easier debug of test failures. The VS-254 FPGA implements the AVE methodology and includes a DDR3 high-speed interface.

**DDR3**

DDR3 is a type of double data rate-synchronous dynamic random access memory (DDR-SDRAM). In an SDRAM, the DDR is synchronized with its master (e.g., a processor). Unlike asynchronous memories, which react to changes in the control, synchronous memories can be pipelined, thus achieving high speeds and efficient access to memory. The DDR data is stored in a simple dynamic physical element, which enables higher densities and, thus, a lower cost per bit. To keep the data from getting lost, the information is refreshed from time to time. The refresh mechanism adds complexity to the controller, and although it somewhat reduces the memory throughput, it is considered a good tradeoff. The memory works on both clock edges, which doubles the memory throughput. As memory speeds rise, signal integrity becomes an important issue. Among other things, there is high sensitivity for the signal termination, which makes signal probing and monitoring operations much more complicated and challenging.

The complexity of the transactions and the number of signals in a 40-bit width high-speed DDR3 interface running at 333 MHz can be seen in Figure 5. Meeting DAL A and B robustness and abnormal testing requirements is next to impossible using a manual or semi-automatic testing environment. And if the FPGA DDR is connected directly to a DDR memory, there is absolutely no control of the pins and virtually no ability for robustness testing on these pins.
The DDR emulators perform all the special testing features required for DO-254 hardware testing requirements, including the complete set of robustness testing. The emulator is programmed through multiple configuration registers, which control all the DDR parameters, including:

- DDR latency
- Control and data line timing
- DDR power level
- DDR frequency shifts
- DDR error injection and correction

The AVE runs the following tests on the high-speed interfaces:

- Validate the physical layer connections and test the normal interface behavior
- Compare the transaction level data with the reference model from the software verification environment
- Modify the input data to the DUT, and test for abnormal situations
- Vary the voltage levels and clock frequencies to understand their impact on the DUT
- Change the physical layer signal timings including control and data

Providing this level of functionality and features is challenging, especially considering that high-speed interfaces often include complex protocols and frequently interface to multiple clients that require arbitration governed by master/slave relationships.

CONCLUSION

Taking concepts from advanced pre-silicon verification methodologies (e.g., UVM) and applying them to a hardware tester validation platform (i.e., AVE) delivers unprecedented control over all aspects of hardware testing, especially high-speed interfaces. As technology progresses, we expect to see more and more devices in the market with different types of high-speed interfaces.

This breakthrough approach enables, for the first time, complete testing that includes high-speed interfaces. It gives developers and manufacturers a much higher degree of certainty in the correctness and safety of their complex electronic devices.

DO-254 expects developers to use their best effort to validate their designs. There is no absolute definition of what is a “good enough” effort. In the absence of absolute proof, you need to invest much time and energy to show that what you have done is as good as technically possible. As solutions are provided that allow for increased coverage, these solutions eventually become the de-facto DO-254 requirement. Because AVE is currently the most complete technological solution to validate high-speed interfaces, we expect it to become a de-facto requirement.
VERIFICATION ACADEMY
The Most Comprehensive Resource for Verification Training

21 Video Courses Available Covering
- Intelligent Testbench Automation
- Metrics in SoC Verification
- Verification Planning
- Introductory, Basic, and Advanced UVM
- Assertion-Based Verification
- FPGA Verification
- Testbench Acceleration
- Power Aware Verification
- Analog Mixed-Signal Verification

UVM and Coverage Online Methodology Cookbooks
Discussion Forum with more than 5000 topics
UVM Connect and UVM Express Kits

www.verificationacademy.com