Do You Know What Your Assertions Are Up To? A New Approach to Safety Critical Verification

Lee C. Smith[1]

[1]Rockwell Collins (400 Collins Road N.E. m/s 131-102, Cedar Rapids, Iowa 52498 lee.smith@rockwellcollins.com)

Abstract-This paper introduces a new methodology and technology enhancements that ease DO-254 compliance for complex airborne electronics hardware (AEH) designs by modifying the way assertions are tracked and recorded using augmented verification IP (VIP) and by using VIP to accelerate coverage of standard protocols. The approach presented improves requirements tracing and helps assure Designated Engineering Representatives (DER) that these requirements have been met. The paper will explain the motivations behind using VIP and assertion-based verification (ABV), how they are useful, and outline the steps taken to make them more effective (applicable?) in the DO-254 compliance process.

Although developed for FPGA DO-254 compliance, this novel methodology has a widening applicability in electronic design and verification because the number of protocols per design is growing and these protocols are becoming faster and more complex. Further, low fault-tolerance design methodologies are becoming more prevalent in the commercial, financial, and security realms.

I. Introduction

DO-254 is a standard enforced by the FAA that requires certification of avionics suppliers' designs and design processes to ensure reliability of airborne systems. The DO-254 compliance process ensures that all specified design requirements have been verified in a repeatable and demonstrable way. All the requirements of the system must be well specified, and each of those requirements must be demonstrated to have been verified. The key to this is traceability.

The assertion-based verification (ABV) methodology is increasingly used to handle the complexity of present day airborne electronics hardware (AEH) designs in the avionics industry. Requirements tracing using the ABV methodology can be accomplished by associating targeted functionality from requirements to assertion execution results. The entire process includes a simulation log, assertion waveform, and assertion coverage. Ultimately the goal of traceability is to satisfy the Designated Engineering Representatives (DER) that these requirements have been met.

This paper will share a new approach that eases achieving DO-254 compliance for complex AEH designs. This novel methodology and supporting techniques, although targeting FPGAs for DO-254 compliance, has a widening applicability in electronic design and verification because the number of protocols per design is growing and these protocols are becoming faster and more complex. Furthermore, the complex and stringent nature of DO-254 is going to become a lot more prevalent in the commercial realm. Whereas DO-254 is more about safety, low fault-tolerance design methodologies similar to DO-254 are applicable in many areas; such as financial institution and internet security.

II. Problems with Advanced Verification for DO-254

The biggest challenge on the DO-254 front is traceability — from the requirements through the verification cases and procedures to a result. Ultimately the goal of traceability is to satisfy the DO-254 DER that these requirements have been met. This is where things get tricky for the verification engineer. Advanced verification technologies like assertions and coverage-driven testbenches are outside the area of expertise for most DERs. This is why verification engineers need a way to prove that the assertions are doing what they are supposed to do in a way that someone not familiar with their complex syntax can understand. Likewise, they need a way to demonstrate that their design is fully verified using coverage metrics.

Part of the challenge in using a traditional ABV flow is that, typically, when a verification engineer uses assertions, they are looking for only those assertions that fire, which tells them there is a possible error in the design. In the context of DO-254 compliance, assertion logs must record exactly the opposite. Instead of paying attention only to those assertions that flag a bug (and the assertion "fires"), they must be able to show that an assertion was exercised even when it does not fire. In other words, in the context of DO-254 compliance, the DER needs to see evidence that the assertions are actually checking the design against the requirements, even when the design is behaving correctly.

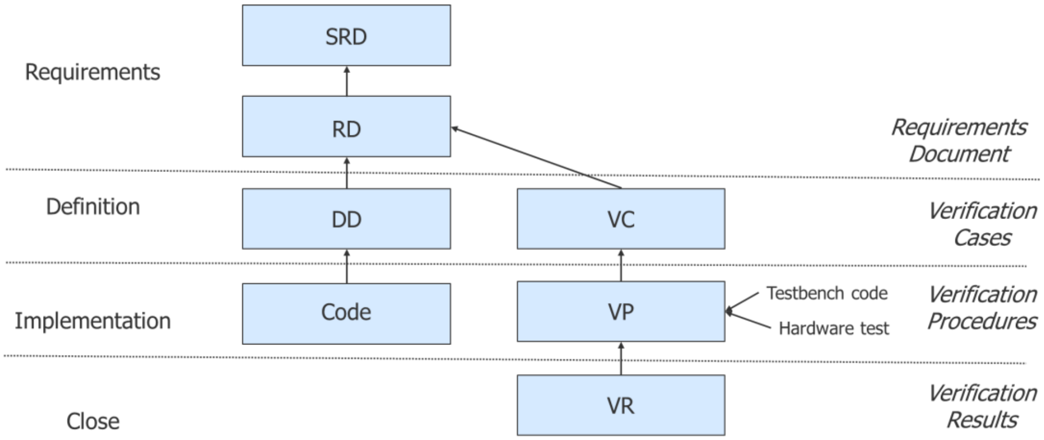


Figure 1. DO-254 for DAL-A

Before we look at the new approach, let's briefly look at the DO-254 design flow, which shows the relationship between requirements, verification cases, verification procedures, and verification results. DO-254 requires a waterfall approach as illustrated in figure 1. The requirements must be developed before the design and verification begins. The process can be divided into two separate tracks, one for design and one for verification, with each phase of the process driven by the output from the previous phase. The relationship between these documents must be tracked to ensure correct validation and verification. DO-254 requires the ability to trace from requirements to both implementation and verification results to show the design was implemented correctly and tested.

The device **shall** assert LED[0] (red LED) and de-assert LED[1] (green LED) when the C0_RST_F pin is asserted.

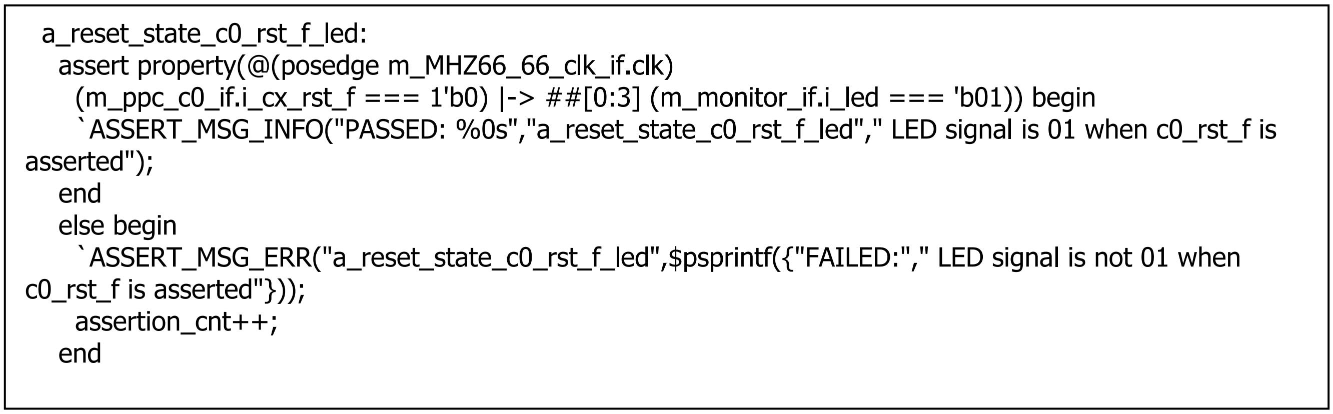
Figure 2. Sample Requirement

Figure 2 shows a sample requirement that is a good candidate for verification through an assertion.

When C0_RST_F is asserted, LED[0] must be asserted and and LED[1] must be deasserted.

Figure 3. Verification Case for Sample Requirement

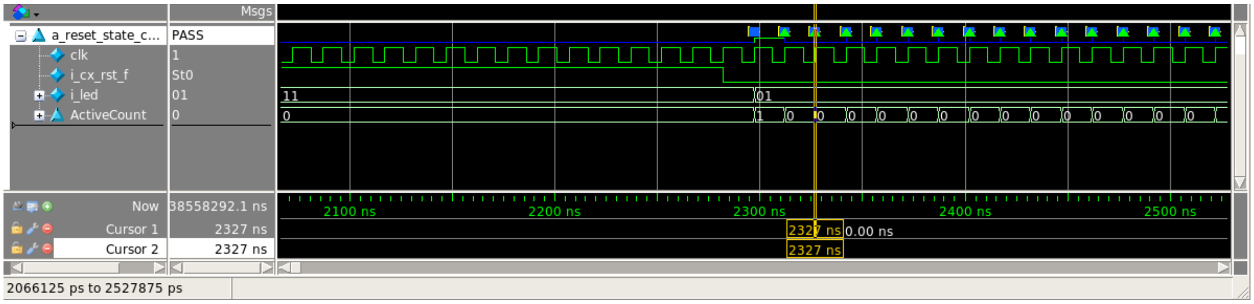
Figure 3 shows a simple verification case that can be used to drive the verification of the requirement in figure 2.



```
  a_reset_state_c0_rst_f_led:
    assert property(@(posedge m_MHZ66_66_clk_if.clk)
      (m_ppc_c0_if.i_cx_rst_f === 1'b0) |-> ##[0:3] (m_monitor_if.i_led === 'b01)) begin
       `ASSERT_MSG_INFO("PASSED: %0s","a_reset_state_c0_rst_f_led"," LED signal is 01 when c0_rst_f is
asserted");
    end
    else begin
       `ASSERT_MSG_ERR("a_reset_state_c0_rst_f_led",$psprintf({"FAILED:"," LED signal is not 01 when
c0_rst_f is asserted"}));
      assertion_cnt++;
    end
```

Figure 4. Assertion to Verify Sample Requirement

Figure 4 shows the assertion code used to verify the sample requirement.



```
                     Msgs
a_reset_state_c...   PASS
  clk                1
  i_cx_rst_f         St0
  i_led              01
  ActiveCount        0

Now        38558292.1 ns
Cursor 1         2327 ns
Cursor 2         2327 ns

2066125 ps to 2527875 ps
```

Figure 5. Wave Capture of the Assertion Signals

Figure 5 shows the wave capture of the assertion signals.

III. Advanced Verification for DO-254

In order to address the two fundamental challenges of traceability and coverage, Rockwell Collins collaborated with the Mentor Graphics® Questa® verification IP (QVIP) team to modify the Mentor Graphics PCIe QVIP so it could produce the artifacts necessary to assure DO-254 DER regulators that assertions had indeed performed what they were intended to do. Mentor and Rockwell Collins also enhanced Mentor QVIP so that it would provide the mechanisms to demonstrate that they achieved complete functional coverage.

A. ABV for DO-254

Rockwell Collins and the Questa VIP team made modifications to the PCIe QVIP to make it compatible with DO-254. These modifications were based on the ABV requirements tracing methodology Rockwell Collins had already created: utilizing a simulation log, assertion waveforms, and assertion coverage to help DERs understand the value of assertions and functional coverage metrics.

The modifications to the PCIe QVIP enabled it to support snapshots of assertions and provide enough information in the log file for functional validation. The log file shows that a particular assertion passed and the snapshot shows that assertion in operation to prove that it worked correctly.

This involves executing assertions automatically aligned to the DO-254 requirements. The simulation results are stored by Mentor's Questa simulator in simulation log files as well as in waveform (.wlf) files. The latter enables debugging the issues in visual form in a waveform window in a swift manner. An assertion snapshot image file is also created, which is a smaller and targeted version of the assertion wave file along with the required result to use as evidence for traceability. Compared to the manual capture of a large number of assertion results, the automated script saves a significant amount of time and even reduces the instances of operator errors.

The checkers, assertions, drivers, monitors, and other components that come with QVIP to verify that a design is working correctly also contributed to DO-254 qualification, because these are core tasks that must be done.

B. Coverage for DO-254

The second requirement is to be able to use functional coverage metrics and cover groups to show that the device has been adequately exercised and prove that the cover groups were populated correctly. For this purpose, a script was created that scrubs through log files looking for transactions that would be required to populate the cover groups of interest.

This was a significant enhancement over the previous ABV requirements tracing methodology. Compared to the manual capture of a large number of assertion results, the automated script saves a significant amount of time and reduces the instances of operator errors.

For example, Rockwell Collins designs have somewhere in the order of 200 to 500 assertions. The new methodology involves pulling up a particular assertion and its associated signals, capturing a screenshot of the assertion in action, creating a wave file, pulling coverage messages from the UCDB and log file, and finally using all of that to populate the results document. Instead of having to do all of that manually, 200 to 500 times, a script was created to help automate the assertion search and snapshot process.

IV. DO-254 ABV Flow

The assertion based verification flow starts with the running of several hundred constrained random simulations. This presents several problems from a DO-254 traceability perspective. Due to the constrained random envirionment, it's not possible to know which simulation will contain the results for verification of a specific requirement. In order to address this, the assertions have been modified to generate a pass message in the transcript.

When the simulations have completed, a script scrubs through the transcripts looking for the passing message from each assertion. This provides a link between the assertion and evidence that it ran and passed and includes the time when the assertion passed.

The post-processing script will then open the simulation waveform file associated with the transcript in a wavefrom viewing window, it will add the assertion and its sub-signals to the view and format the viewing window to show the signals slightly before the assertion triggers and slightly after the assertion passes. Once this is done, the script will generate a snapshot of the resulting view for inclusion in verification documents.

The assertion snapshot is then used to validate that the assertion is coded correctly and not falsely (or vacuously) passing during review. And as evidence that the design was tested and meets requirements.

V. Importance of Standards

PCIe QVIP was also important in fulfilling its primary purpose; helping the team to quickly and thoroughly verify that a third-party PCIe IP interface worked correctly in a Rockwell Collins memory controller device for use in a commercial aircraft avionics system. The memory controller was based on an existing design that had a PowerPC interface, which they replaced with the more complex PCIe interface.

The Questa VIP library provides engineers with standard Universal Verification Methodology (UVM) SystemVerilog (SV) components using a common architecture across all supported protocols. Test plans, compliance tests, test sequences, and protocol coverage are all included as SV and XML source code, allowing simple reuse and debug. The QVIP components also include a comprehensive set of protocol checks, error injection, and debug capabilities.

The underlying quality that makes all of this possible is the adherence to a standards-based flow. The third-party IP, the QVIP, and the integrated test environment are all based on a single standard. This supports creating complex protocol testbenches that would not otherwise be achievable without slipping project schedules. Design and verification teams do not have to develop industry-standard checks and assertions as they are included in the QVIP package. Because designers come to understand these standards and how they work, they are able to come up to speed quickly and make a difference on a project. It is not necessary to train them on particular requirements or on a particular interface, which often would result in a project being late.

VI. Conclusion

By collaborating with Mentor Graphics to make enhancements to Questa VIP (QVIP) targeting DO-254 certification, Rockwell Collins was able to use the Questa simulator and QVIP for safety critical verification using advanced verification techniques, modify PCIe QVIP to support automated DO-254 qualification of assertions and coverage, and take advantage of the PCIe QVIP infrastructure for standard protocol validation to speed testbench creation and shorten the design schedule.

With protocols getting faster and more complex, having QVIP that delivers this kind of standardized approach and standardized verification components is going to become even more important. Therefore, this methodology will be useful for a range of companies outside of the mil-aero sector.