

VHDL and it's Application for Department of Defense Contracts

Adam E. Rosenberg
Protocol, A Division of Zycad Corp.
100 Enterprise Drive
Rockaway, NJ 07866
Phone: (201) 366-8100
adamr@protocol.zycad.com

2Lt. Tony Harris
F-22 System Program Office
ASC/YF(FAS)
Wright-Patterson AFB, OH, 45433-6503
Phone: (513) 255-7503

ABSTRACT

In today's Air Force, the development of weapon system electronics is both a complicated and integration intensive endeavor. With the advent of Integrated Weapon System Management (IWSM) (i.e., 'cradle to grave' responsibility), proper representation of electronic hardware is crucial for efficient post-production reprocurement. This paper describes how VHDL and a Top Down System Simulation (TDSS) program can provide solutions to the aforementioned challenges.

Section 1. Introduction

The Air Force is currently facing cultural and technological changes affecting the development and procurement of weapon systems. The complexity of the aircraft being designed today dwarfs that of older platform configurations and such platforms demand state of the art electronics that push the performance/complexity envelope. Considering the shrinking defense budget, the F-22 program was faced with the task of defining a new, more efficient way of developing digital electronic devices. Experience has shown that 50 percent of the newly developed digital electronic devices are reworked due to interface related problems. With little in the way of existing methodology, it was decided that a system modeling/simulation effort, using VHDL, offered the best chance of first pass system success.

The modeling effort was developed and packaged within a process entitled Top Down System Simulation. The dual focus of the TDSS program was design process risk reduction and 'as built' reprocurement documentation. Due to the number of F-22 vendors, a common understanding of the interfaces and functional requirements, across shared resources and at different system levels, was critical. This was to be accomplished by verifying, via simulation, critical functions and interfaces before key program milestones (Preliminary Design Review, Critical Design Review, etc.). The models are the F-22 Team's vehicle for exposing specification misinterpretations and/or ambiguities prior to fabricating hardware with serious design errors.

The F-22 program was also concerned about supportability of the aircraft after production. This includes weapon system upgrades due to new threats, new missions, or advancements in technology. With the complexity of the digital electronics, as well as, the intensive integration of aircraft functions it was determined that the supporting agencies would need something more than the normal paper documentation. The models enhance their capabilities by allowing them to easily modify and test the upgraded designs. Furthermore, the support personnel can verify whether the design

will operate within the whole system. The models permit the support personnel to expose any integration related problems before the design is fabricated.

VHDL was chosen as the common language across vendors because it provides the versatility the Air Force needs in the procurement environment. VHDL also is called out in MIL-STD-454M (Requirement 64, Microelectronic Devices), which is currently a contractual requirement on the F-22 program, as a means of documenting digital ASICs. VHDL was not used solely because it was a requirement. With all that VHDL had to offer, it made sense to use it for the proposed TDSS program. Granted, VHDL was relatively immature for this type of application, but the F-22 program felt that it had the resources to help drive it to maturity.

Section 2 The Process-

To ensure the successful insertion of VHDL into the F-22 program, additional up front planning was required. This planning impacted not only the government and prime contractors but also all the subcontractors responsible for an original digital design.

The first tasks of the F-22 program included developing TDSS guidelines and implementation plans, setting up a Bulletin Board System (BBS) and developing a set of VHDL standards and practices known as the F-22 VHDL Model Specification (see Figure 1).

Numerous feedback paths were inserted to ease the introduction of TDSS into the F-22 program (i.e., problem reporting system, design reviews). The philosophy of addressing issues early in the program and passing problem data back as early as possible was the only way to assure the program's success.

With the rules established, the first models were developed. These were Bus Functional Models (BFMs) that detailed the interface of a given device without providing its internal functionality. The models served as electronic executable interface specifications and were used to iron out specification issues. The BFMs came through as a TDSS win by uncovering many specification ambiguities early on in the design phase.

The BFM models were developed in several different simulation environments making it difficult to exchange them between the various F-22 contractors. To solve this problem, the models were ported to the three most common VHDL simulation environments. Additionally, the models were verified to ensure they were functionally equivalent to the original model (producing the same results) and could be compiled and simulated in the original environment. These ported models were then distributed to all team members.

The initial interface testing involved F-22 Team members simulating their designs against another member's designs to flush out any specification ambiguities. This exercise involved modifying test benches to allow new models to be connected to existing designs. This greatly increased the Team's confidence that different designers had the same interpretation of the specification.

The next step in the process involved producing higher fidelity behavioral models. These models were also ported and distributed throughout the community. The models include increased functionality, higher specification fidelity, and greater error checking. They also incorporated changes identified by earlier testing and feedback from other team members.

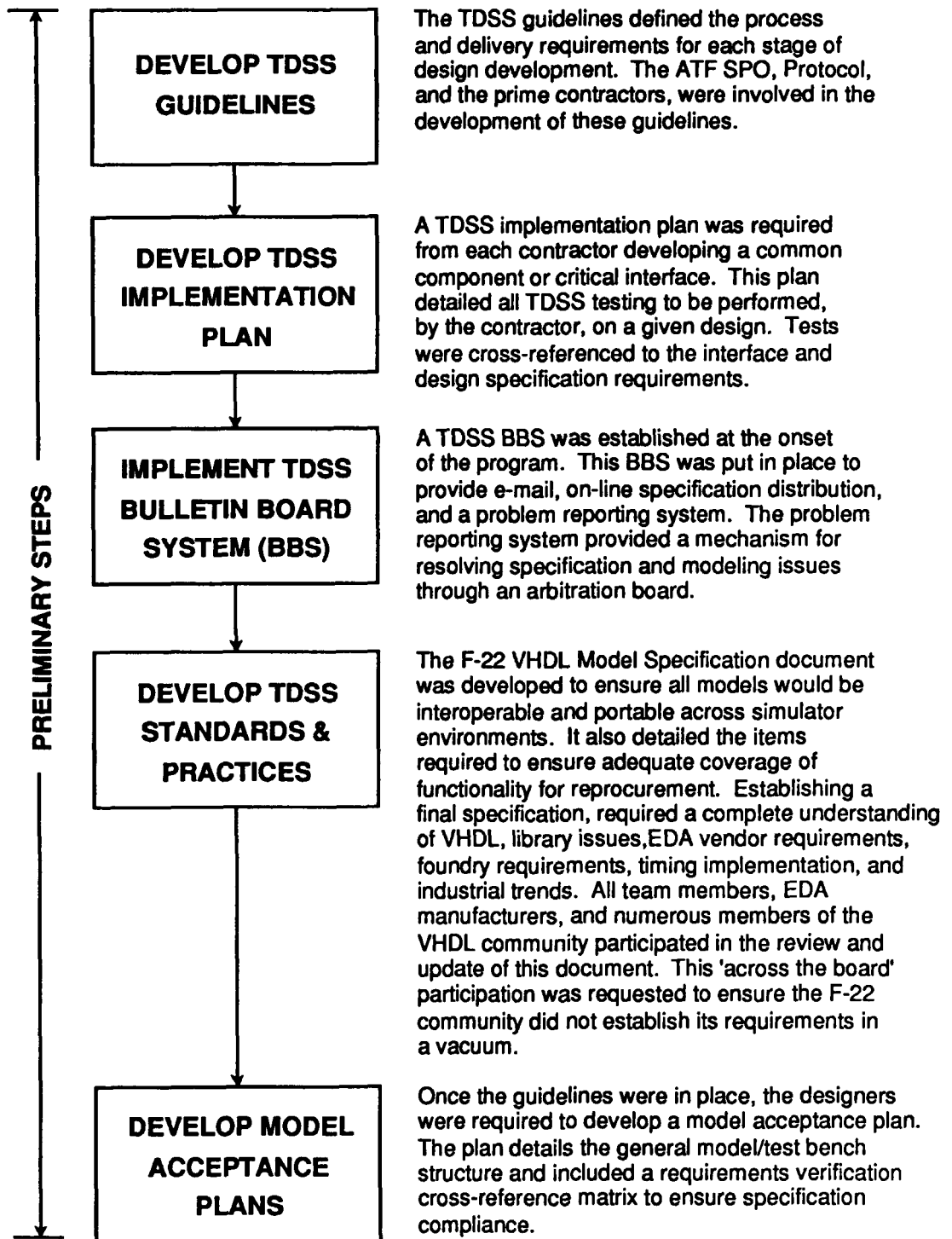


Figure 1. F-22 TDSS Program Preliminary Tasks.

Prior to the development of higher fidelity Gate Level Models, the contractors were required to develop a simulation test report. This is actually a dual submittal document. The first submittal details how a model will be tested and the second submittal adds the actual test results. Once the first submittal is accepted by the prime contractor, the designer can begin developing the Gate Level Models.

The next level of models developed were gate level models. These models were again ported and distributed throughout the community. Any issues that arose, were investigated and resolved throughout this phase. A majority of the F-22 team members opted to translate their designs, created by non-VHDL tools, to Gate Level VHDL. Some team members synthesized their designs from RTL VHDL models – a true Top Down approach.

Due to the size of the simulations on this program, any design over 50,000 gates required an acceleratable library as part of the delivery. These models were then interconnected to perform stress testing.

Stress testing ensures that the design can withstand the worst-case loading without ill-effect. The testing involved exercising each of the design's interfaces simultaneously, using the maximum throughput rate for each interface.

After stress testing comes the inter-LRM testing phase. This testing phase ensures interoperability across the multiple design paths throughout the system. Several of these simulations are in the 12 million gate range and require multiple libraries.

At this point the companies fabricate their designs. The same test vectors which have been run against the VHDL models are now run against the silicon. All VHDL designs must be verified to hardware before government acceptance. This final VHDL code becomes part of the documentation of the design and is necessary for design reprourement. The contract also requires the delivery of the behavioral models to support technology independent reprourement at a later time.

Section 3. The Lessons Learned -

There were several problems solved and numerous lessons learned on this program. A major lesson learned was the value of VHDL modeling. VHDL proved its worth by uncovering incompatibilities and specification ambiguities before fabricating the silicon. Of the problems encountered, most can be grouped into the following categories:

- Starting TDSS too late
- Simulator inconsistencies
- Library issues

Starting too late was one of the most common problems encountered. The deliverable requirements might not have seemed so burdensome, if the designs were started in VHDL and followed the progression from BFM to Behavioral to RTL to Gate level. Unfortunately, most companies opted to design in a non-VHDL environment and either have a parallel VHDL design effort or translate to VHDL. Of all problems, this probably had the greatest cost impact on the program because the translation was an additional step that could have been avoided by greater up front co-ordination and co-operation. Failure to come to terms with standards and practices also plagued the program. Also, some of the original requirements were harsh and unnecessary. While they were being updated, companies had to continue with their designs, unsure of the final delivery requirements.

Simulator inconsistencies were caused by the tool vendors' different interpretations of the VHDL Language Reference Manual. Any issues that arose during model porting were forwarded to the appropriate simulator manufacturer, but often the different vendors stuck to their own language interpretations, leaving no alternative but to use a 'least common denominator' subset of VHDL. Versioning was also a problem, when, at the beginning of 1993, the three most popular simulation environments changed versions. This required all previously ported models to be re-simulated in the new environments because no single version could be selected due to tool immaturity. Another important lesson involved Non-Disclosure Agreements with the tool vendors. If a simulator problem occurred, such as an internal compilation failure, the simulator manufacturer requested a copy of the code that was running to help replicate the problem. With the proprietary nature of the code involved, Non-Disclosure Agreements had to be signed prior to its release. This often delayed the turn around time required to troubleshoot and affect the necessary corrections.

ASIC and standard part library issues were also an area of concern. Some library suppliers required a fee for circulating models that contained any of their source code libraries. Many of the large simulations required the accelerators to handle multiple foundry look-up tables due to the vast number of libraries used in the inter-LRM testing. Other library related issues involved methods for handling timing, derating, back annotation, and library availability.

Section 4. Future Government Programs

With the direction that military weapon system programs are heading, (i.e., tighter budgets, cutting-edge capabilities, and management of advanced technologies) it is imperative that contractors embrace a true top down design methodology using VHDL. The F-22 program has laid the foundation and provided the experience necessary to keep VHDL moving forward in the DoD procurement environment. However, the F-22 program has only scratched the surface. The next generation of DoD programs, must take the lessons learned on the F-22 program and improve on the process.

One area for improvement, from a contractor's point of view, is the hardware development process. With the complexity of current electronic designs, it is very risky to analyze a design and then immediately begin pushing gates around. For past designs, this methodology worked, but not without multiple hardware fabrication iterations affecting both cost and schedule. This is not a viable approach considering the present budget constraints. The contractor must 'raise the level of design abstraction'. This allows for faster development of executable models, permitting the designer to research several implementations and increase the odds of choosing the best solution. Also, the contractor's ability to adapt to changing requirements is greatly enhanced. Successful simulation of these executable models provides an excellent method for verifying preliminary designs while still in the early stages of the program. Of course, the next step is to partition the functionality into manageable portions and begin detailed design. Without synthesis, this step should follow the contractor's normal design process except now the designer has a more direct and better understood path to the lower levels of the design. In this scenario, the design portion of the schedule may or may not be impacted, but the quality and the assurance of design functionality have increased, which will decrease the time needed for hardware integration and test, resulting in a net schedule savings. With synthesis, development of the gate-level is automatic, further decreasing the development time. The only concern is the maturity of the synthesis tools. Overall, this true top down design approach provides the level of modeling the Air Force needs to feel confident the design will perform as expected the first time it is built.

These functional models also allow the contractor to look at the whole system and analyze how their portion interacts and interfaces with the components around them. Modeling provides a means of design communication between two vendors whose components must interface together. This is especially important when a large number of vendors are involved. For instance, there are five vendors involved in designing one of the F-22 Line Replaceable Modules (processing card). In the past, the government had a concern when one company was developing a module. Now, there are five interpretations of the module specification and internal interfaces. This presents an integration nightmare, unless there are models to flush out inconsistencies in the specifications and misinterpretations of the interface requirements. By exchanging models between vendors, a common understanding can be reached.

The process should start with bus functional models to establish a comprehension of the interfaces between components. Then behavioral models are developed to aid in the system (module) functional verification. Finally, gate-level models are created that represent the actual hardware design to be used for fabrication. The vendors are responsible for developing their models and for simulating them with other vendor's models to which they interface. To really make this process work, a system (module) integration team is required. This team oversees the modeling effort, ensuring that vendors meet the system requirements. The team would use the lower level models to test and verify at the system (module) level. This scheme helps to identify performance and functional concerns for correction by the component vendors. The tight feedback loop, along with other checks and balances, ensures that the contractor does not lose sight of the overall system. Complex integration is here to stay, and a narrow view is too risky.

The government would like to see a validated test suite to allow language implementation issues to become negligible if not nonexistent. With the continuance of multiple vendors on procurement programs, model exchangeability is very important. In the past, too much time and money was used to understand what the common subset of constructs are between the different tools to allow portability. There is tri-service support for the development of a VHDL test suite. Both the development of, and the future compliance with, this test suite should be supported by all concerned.

Library companies, with support from the chip foundries, need to take the initiative towards converting their present standard parts libraries to VHDL. With the development of the validated test suite, the government will want contractors to design in VHDL from start to finish, a process that will be extremely difficult without standard VHDL part models. Otherwise, the contractor's will be forced to design in the tool provided language and translate to VHDL later in the program. This method drives up the cost, because of the duplication of design effort, as well as, the need to validate the VHDL model's results against the original model's results. Some library companies offer to code VHDL models on a by request basis, which is acceptable, but not getting those models in a timely fashion could significantly affect the program's schedule. Ultimately, those models need to be provided in a VHDL standard part library.

Commonalty between libraries, necessary to develop a model, needs to be improved. A single ASIC foundry process requires multiple simulator libraries because ASIC design houses have investments in different tools. In other words, multiple models may need to be developed for the foundry to fabricate the design. For instance, a 1 μ m CMOS foundry process requires a model developed against a specific tool design library, a foundry library, an accelerator library, and a VHDL library. The government would like to see commonalty to alleviate extraneous model development.

The use VHDL on a large scale DoD weapon system procurement program is still immature. With the F-22 program, we have started the ball rolling and have learned many lessons. It is up to future programs to continue the refinement of VHDL. It is not going to disappear, especially with the current defense budget situation. Fine tuning

this process must be a joint effort by all involved, with a mature process, the benefits will be reaped by all.

1. VHDL Vis-A-Vis Top Down Design: Understanding the Origins, Applications, and Benefits of VHDL, Tim Davis, Fall VIUF Conference.

2. Gate Level System Simulation, Bond Murray, Lockheed Aeronautical Systems Center, F-22 Program Briefing.