

VHDL Acceleration, Today and Tomorrow

Charlie C. Cheng
Zycad Corporation
1380 Willow Road
Menlo Park, CA 94025

Abstract

VHDL, like many other simulation languages, requires acceleration to meet users' changing needs. With evolving applications of the language, the requirements for acceleration will also change. This paper will discuss today's technology and offer direction for future technology development.

VHDL Today

Design methodology and design methodology changes are a process of gradual evolution. While many have proposed that VHDL will bring about complete system level top-down design methodology, the reality of implementation is not a simple change. Tools have to mature, and designers must migrate their design methodologies to take advantage of VHDL.

In surveying the VHDL community today, it becomes evident that a large percentage of the designers use VHDL in their ASIC development methodology. VHDL is used in two ways: as a synthesis language for the targeted ASIC cells, and as a method to model the rest of the system around the ASICs including behavioral testbench.

Because of how the language is used, most of the system's detailed description lies in the ASICs. The ASIC designs start at a synthesizable RT level, and later on can be as detailed as the ASIC vendor's macro cell level. At the same time, the rest of the system (including microprocessors, glue logic, and memory) are often described as bus functional behavioral models.

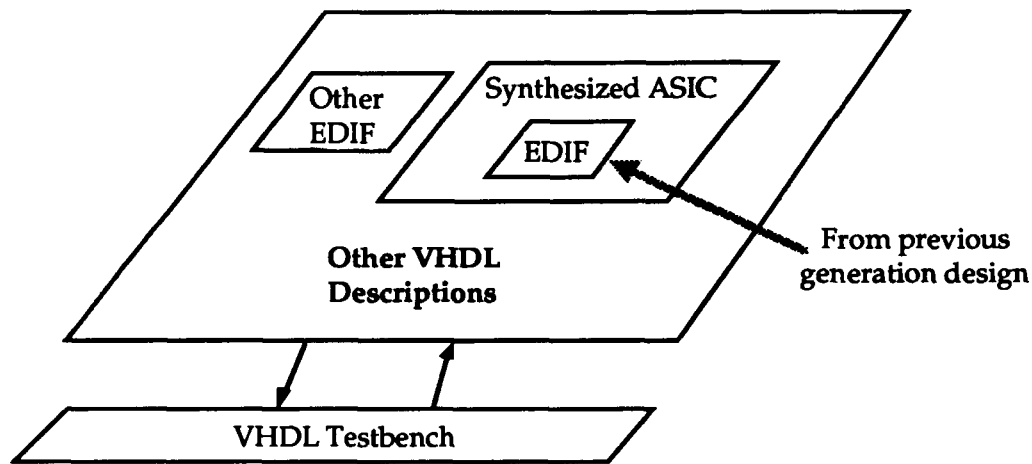


Illustration 1. Today's VHDL Usage

Current Simulation Challenges

In terms of simulation performance, it is commonly accepted that VHDL is often perceived as slower than other gate level simulators for comparable detailed models. This means that while traditional gate level software simulators do not reach their performance limit until the gate count reaches over 50,000 gates, VHDL simulation performance can limit its practical application to ASICs in the 20,000 gate range.

Current Acceleration Technologies

It would be ideal to accelerate all VHDL constructs, thereby letting users describe the ASIC and system freely, with no "acceleratable subset". However, given that such technology is not available today, current acceleration technologies have been designed with two insights in mind:

1. Most of the simulation time is spent inside the ASIC model, and
2. The ASIC, after synthesized to the detailed form, can be simulated by gate level accelerators.

The architecture of this accelerated VHDL simulation environment is usually one of co-simulation. That is:

1. The structural description of the system is mapped to run in a gate level simulation accelerator.
2. The rest of the VHDL description is run in the workstation.
3. Interface software is developed to connect the boundary of gates and behavior during compile time, and to synchronize the events between the two machines during the runtime.

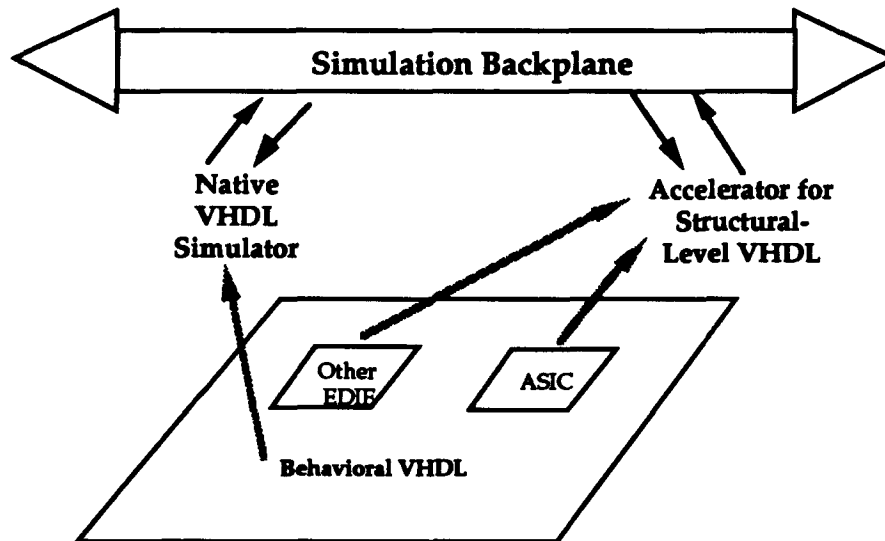


Illustration 2. Today's Acceleration Technology

In terms of performance, the current technology could potentially increase the performance of the structural level VHDL by a factor of 50 or more. This estimate is based upon the fact that:

1. Gate level accelerators today can usually deliver from 10X to 100X gain in speed compared to workstation, and
2. Given a comparable gate level design, traditional gate level simulators can be as fast as 5X that of a VHDL simulator.

However, this co-simulation technology is not without its shortcomings:

1. Workstation and accelerator interaction:

The workstation and the accelerator usually have two types of communication activities: synchronization controls, and event passing. If the communication is not optimized to keep at a minimum, the workstation can slow down the accelerator.
2. Level of abstraction:

To achieve acceleration the user must describe the system at the RTL subset of the language. There are three implications to this restriction:

 - The user is limited to a subset of VHDL. Depending upon the synthesis tool used, this subset may or may not be acceptable to the user.
 - Part of the design which is described in a higher level of abstraction must run in the workstation, which can add communication overhead.
 - Lastly, by having to synthesize the design to an acceleratable gate level description, artificially detailed information gets generated. Therefore, while the accelerator may be 50 times the performance of the workstation at the structural level, it may only be 5 to 15 times the performance of the workstation from a co-simulation perspective, because the workstation runs the RTL description of the design.

Life of Current Technology

In spite of these weaknesses of the gate level accelerators for VHDL co-simulation, today's technology will be around for several years for two main reasons. First, the VHDL simulation algorithm is generally perceived as being slower than gate-level traditional simulation algorithms. Second, gate level models and ASIC libraries will continue to be used in the next generation designs. And thus, accelerators can provide real value when added to the VHDL simulation environment.

To a large extent, the life of current acceleration technology actually depends upon when ASIC vendors will switch to using pure VHDL to describe their ASIC library cells. Currently, the biggest obstacle is having an industry wide accepted methodology on back annotation. The VHDL 1992 effort is under way to address this issue. Given that gate level ASIC libraries (in EDIF form) exists today, gate level accelerators for VHDL will have a useful life for sometime to come.

VHDL Tomorrow

As the user community matures, and the availability of models becomes more prevalent, VHDL will be used as the tool to design and verify the entire system, not just the ASICs. Therefore, the entire system will be modeled in more detail, both in terms of functionality and timing. The system will be constructed of models for commercial parts, such as processors/controllers, ASSPs (Application Specific Standard Parts), glue logic, as well as user's custom ASIC designs. In terms of complexity, the model will grow from a 70,000 gate ASIC with some bus functional peripherals to an entire system that can span up to 10,000,000 gate equivalents.

Emerging Acceleration Requirements

The simulation challenge will now be how to accelerate the entire system rather than just the ASICs. In addition, the "synthesizable subset" will no longer be acceptable, as commercial models can not be limited to any given subset. Finally, with the entire system modeled at a detailed level, users will want to simulate the system with software to obtain anywhere from 2 to 30 seconds of real time. In summary the following acceleration requirements are emerging:

1. Accelerate most, if not all of the 1076 specification.
2. Provide at least two orders of magnitude performance advantage over general purpose workstations across all levels of abstraction.
3. Provide a smooth migration path for users of current acceleration solutions to upgrade to new technologies.
4. Tight integration into EDA vendor VHDL software simulation offerings.

Emerging Acceleration Architecture

The following is a high-level outline of an architecture which would address the emerging requirements:

1. **Transferring current technology:**

Current accelerator technology has many virtues that should be adopted by a VHDL accelerator. Among them are:

 - **Hardwired algorithms**

Many of the VHDL algorithms could be imbedded in hardware to yield a 5X to 10X performance improvement.
 - **Very wide buses.**

Instead of being restricted to a Von Newman variety bus architecture, accelerators can optimize bus design for maximum bus bandwidth requirement of simulation algorithms.
 - **No operating system overhead.**

Without the overhead of the UNIX operating system, memory sizes, operations, and interruptions can be kept at a minimum.
2. **Adopting RISC technology:**

Part of the VHDL simulation cycle, namely process execution is comparable to running a C program. It is, therefore, desirable to take advantage of the RISC technology processors for this portion of execution .
3. **Exploit parallelism within VHDL:**

VHDL is inherently parallel, and the parallelism can be used for performance enhancement. Signal assignment, waveform editing, process execution, etc. can all be done in parallel.
4. **Backplane integration:**

Today's VHDL simulators are usually tied to a simulation backplane so they can co-simulate with C behavioral models, hardware modelers, etc. Emerging technologies should be built to be integrated into the backplane, so it can take advantage of the richness of the simulation environment.

Summary

While VHDL has many well documented advantages as a design language, it has several limitations in terms of performance. At the same time, the need for complete system level models of hardware and software is dramatically increasing the required simulation performance. Currently, simulation acceleration technology has been successfully applied to help solve the gate level performance problem. However, having simulation acceleration at all levels of abstraction is the solution the industry is waiting for to solve the growing simulation performance requirements.