

Enhancing Test-Driven Development for Reconfigurable Hardware through High-Level Synthesis and Early-Stage Validation

Roman Diachok, Halyna Klym
Specialized Computer Systems Department
Lviv Polytechnic National University
Lviv, Ukraine
rodyachok@gmail.com, halyna.i.klym@lpnu.ua

SUMMARY

High-level synthesis tools help engineers deal with the challenges of building complex systems that use reconfigurable technologies. Such serves as a precursor to well-established methods in the software industry, such as Test-Driven Development, in the development process of hardware components of an embedded system. However, the assistance offered by the high-level synthesis validation tools could be strengthened and targeted at the early stages of project development. This paper describes a hardware testing framework as a means to quickly evaluate the capabilities of embedded components using a unit testing paradigm, leading to Test-Driven Development implementation on reconfigurable hardware.

High-Level Synthesis (HLS) has recently gained significant popularity, contributing to the acceleration of FPGA-based development and simplifying the verification process. HLS expands the capabilities of the FPGA market by allowing even users with basic programming knowledge to evaluate available architectural options quickly. FPGA technology is accessible to software engineers and hardware developers. However, HLS does not provide immediate verification of project credibility, which creates particular challenges in the development process.

This paper aims to develop and implement an HLS-based verification framework that leverages the TDD approach to overcome existing challenges in verifying FPGA-based systems. Additionally, this study evaluates the effectiveness of HLS in accelerating FPGA development and improving program verification. The proposed framework is expected to provide a standardized testing methodology that simplifies FPGA verification while enhancing reliability and reducing time-to-market.

The Test Manager, Device Under Test (DUT), and communication partner are interconnected via AMBA technology, which ensures efficient data transmission. The Test Manager is directly connected to the bus, receiving messages that control the DUT's testing process. It integrates into the development process by delegating data transfer tasks to DDR memory. In this model, test cases provide stimulus data to reserved DDR regions, which the Test Manager reads to

configure and trigger the DUT. Subsequently, DUT outputs are written back to DDR, where the Test Manager reads and compares them to predefined reference values.

This paper describes a framework for verifying the Ceedling protocol. This framework aims to develop and implement an HLS-based design verification environment that utilizes a TDD approach to address the current challenges of verifying systems implemented with FPGA technology. Ceedling is a complete package that uses HLS-based hardware component testing to facilitate and improve the design of embedded systems targeting the FPGA platform. The main objectives of this framework include evaluating the effectiveness of HLS in accelerating FPGA-based development, developing a test environment for modeling hardware components, and implementing TDD principles to ensure the correctness of designs. To the best of our knowledge, this is the first study to create an on-board verification environment that can be used throughout the design cycle, regardless of the level of abstraction associated with system specifications. This facilitates the development of the necessary hardware and software components and saves time and effort. In addition, Ceedling allows you to use the TDD method in system design, enabling you to take advantage of the popular software development method. Ceedling supports functional testing and timing testing; this is a new complexity related to the nature of real-time projects and hardware. As a result, the proposed verification framework includes three primary levels of abstraction regarding the specifics of the embedded system (functional, RTL, and implementation/physical). The main contribution of this framework is creating a configurable and standardized test environment for heterogeneous devices, which allows engineers to reduce the amount of testing and accelerate the path to FPGA implementation. After all, automatic generation makes this proposition accessible and straightforward for software and hardware developers, who do not need extra effort to use it quickly.