Extended Abstracts of the 2016 International Conference on Solid State Devices and Materials, Tsukuba, 2016, pp755-756

PS-5-07

Property-Driven Functional Verification Technique for High-Speed Vision System-on-Chip Processor

Victor Nshunguyimfura, Jie Yang, Liyuan Liu and Nanjian Wu *

State Key Laboratory for Superlattices and Microstructures
Institute of Semiconductors, Chinese Academy of Sciences, Beijing 100083, China
Phone: +86-010-8230-4754 E-mail: nanjian@red.semi.ac.cn

Abstract

The functional verification is a challenging task in vision chip verification process. The stepwise nature of the existing functional verification techniques increases tremendously the verification complexity. This paper proposes a semi-automatic property-driven verification technique. The implementation of all verification components is based on the design properties. We introduce the low dimension property space between the specification space and the implementation space. The aim of this technique is to speeds up the verification process of high parallel processing vision chip. Our experimentation result shows that the proposed technique can effectively improve the verification effort up to 20% for the complex vision chip design while reducing the simulation and debugging overheads.

1. Introduction

Today's high integration and high performance demands make the vision chip design more complex[1-3], the chip functional verification task becomes even more challenging as well. Although different techniques were reported to improve the functional verification [4,5], the number of verification engineers keep increasing at 3.5 times the rate of increase of designer engineers [6]. Our device under verification (DUV) is a high speed vision chip based on multiple levels of parallel processors. The chip mainly consists of a control processor, functional modules and four MIMD computational clusters. Each cluster consists of four processing cores, as shown in Fig.1. This high integration increases the verification complexity. The traditional verification approaches become inefficient to verify such complex system as they consume tremendous efforts during the design development.

In this paper, we propose a property driven approach that builds a practical and efficient verification environment based on the design properties. This approach decreases both the verification complexity and the debugging time, while shortening the verification cycle.

2. Vision processor verification architecture

The functional verification process is the most critical element during the vision chip development. It requires a significant amount of effort and resources. This process goes through many iterations starting from the specification to the debugging stage. The Fig.2 shows the average of time spent at each stage in the common verification process [7]. The intrinsic stepwise nature of the process is the major cause of the high overall time applied to verification task.

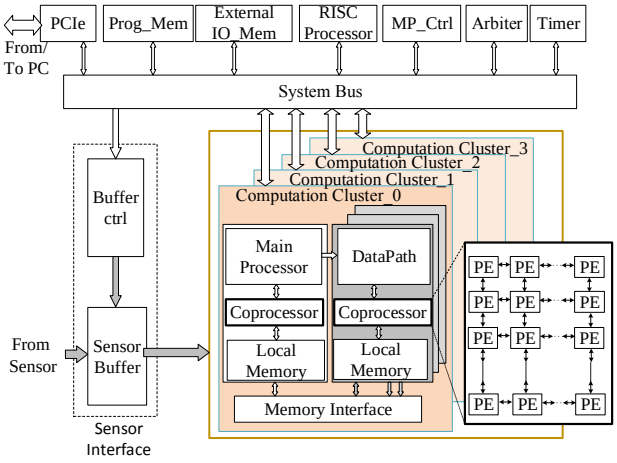

Fig.1 Simplified Architecture of the vision processor

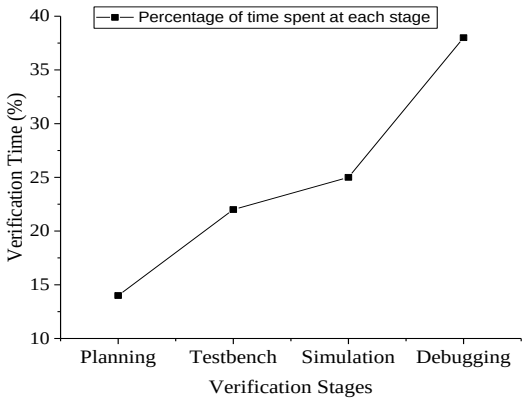

Fig. 2 Percentage of time in common verification cycle.

To reduce the verification burden considerably, we analyzed where we spend most of the time during the functional verification cycle. The following are the six main stages of our vision chip verification:

Property extraction (PE): From the design specification we extract the design intent independently of the verification environment. We adopt SVA [5] to express both the legal and illegal behavior of the design structure, temporal properties and non-determinism properties. We break the complex properties into simple ones to reduce the debug-ging exposure. These properties allow us to define targets with limited ROI and monitor the verification progress.

Formal verification(FV): We adopt mathematical reasoning to exhaustively verify that design intent was preserved during the RTL implementation. This process reduces

systematically the simulation stage overload.

Assertions (AS) : The observability is a challenge to the functional verification quality. We adopt ABV methodology [4] to improve verification quality, while SVA checkers are used to prove or discard some design properties [8]. The assertions used throughout our verification process are grouped into three categories namely block level assertions, interface level assertions and system level assertions.

Stimulus generation (SG): Generating high-quality verification patterns is a critical task for the vision chip verification. The general practice requires a very large amount of test data, which is a labor-intensive and a very time-consuming task. To efficiently reduce the wasted time in generating and simulating useless patterns, we generated deterministic and random stimulus, both constrained by the design properties.

Functional Coverage(FC): We use functional coverage to enhance the advantage of the assertions in checking the run-time behavioral properties. We use FC as our metric for measuring how far we have covered the property space and reached all the defined functions.

The last step involves the simulation and analysis tasks(SA). This is the most overloaded stage in most of existing verification methodologies. Fig.3(a) illustrates the stepwise form of the vision chip functional verification. By combining the AS with FV in one stage (T2) and SG with FC in the following stage (T3) we reached to decrease the number of stages down to four stages, as shown in Fig.3 (b).

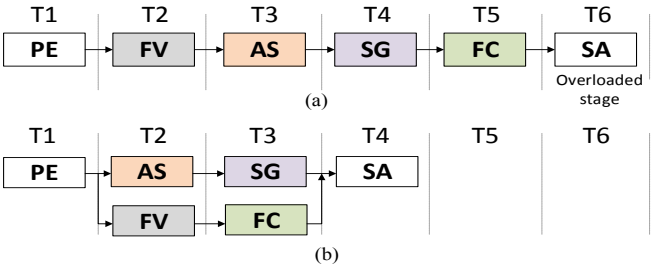

Fig.3 The stage involved in vision chip verification cycle: (a) using stepwise approach, (b) Parallel approach, some tasks are combined in stage T2 and T3.

3. Verification implementation

In general techniques the functional verification effort is not well distributed to prove the vision chip correctness. Contrary, it is mostly oriented to the implementation of the verification environment. By mapping the design specification space to the low dimension property space, we can reduce the dimensionality problem encountered in verification space. We extract the potential properties from the design specification space to create an intermediate property space, as shown in Fig.4. The property space includes the regions of interest (ROI),the block-level properties, interface properties and the system-level properties. All the tasks in each verification stage depend on the properties defined in the intermediate space. This technique improves the verification environment flexibility and scalability. We can add more properties as long as the design functionality increases without changing the existing environment.

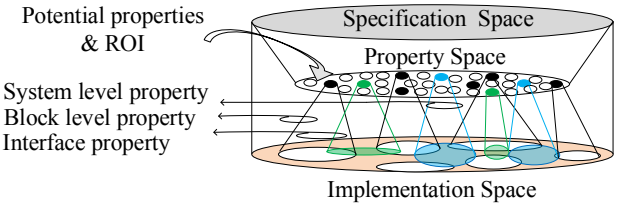

Fig.4 Mapping the design specification space into the intermediate property space.

4. Experimental results

Introducing the low dimension space and the parallel technique improves considerably the verification effort. The deliverable time from one task to start another task is minimized. As shown in Fig.5, using our proposed approach we can effectively overcome the pressure accumulated at the simulation and analysis stages during the common verification cycle. A 20% reduction of the total verification cycle is achieved.

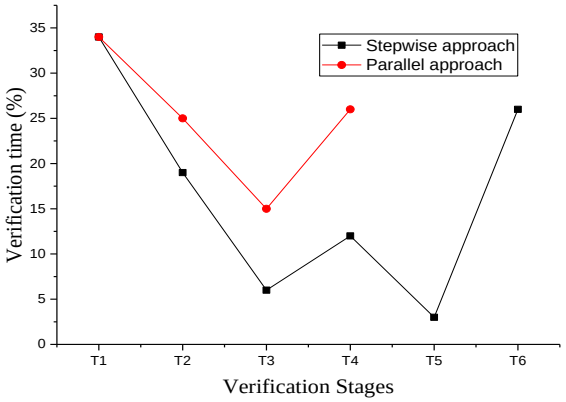

Fig.5 Percentage of time spent at each stage during the vision chip verification process.

5. Conclusions

In this paper we introduced an effective pre-silicon verification solution to the challenges encountered during the Vision SoC verification. The property driven functional verification proposed in this paper breaks the stepwise nature of the existing functional verification methodologies into a new parallel implementation technique. With this technique, we have implemented a high performance verification environment that decreases significantly the vision chip verification task overheads.

References

[1] Shi Cong et al., IEEE J. Solid-State Circuits. 49 (2014) 2067.
[2] W. Zhang , Q. Fu and N. Wu . IEEE J. Solid-State Circuits, vol.46(9),pp.2132-2147, 2011.
[3] Q. Lin et al., Sensors, vol. 9, p.5933-5951, 2009.
[4] N. Victor et al., International Conference on ASIC(2015)B7-5.
[5] D. Bustan et al., Design Test of Computers, IEEE, vol. 29(2), pp. 23-31, 2012.
[6] Wally Rhines, Design and Verification Conference (2016)1.
[7] H. Foster, IEEE Design Automation Conference (2015)1.
[8] M. Mefenza et al., International Microprocessor Test and Verification Workshop(2014)16.