DRILLS: Debugging RTL Intelligently with Localization from Long Simulation

Donggyu Kim, Vighnesh Iyer

The goal of this project is to develop a productive debugging tool for hardware designs. Specifically, we will develop a methodology to localize the origin of bugs in hardware designs given a runtime assertion failure and an error trace leading up to the failure.

As hardware complexity increases to meet performance targets and implement required functionality, verification becomes much more challenging. A recent study shows that verification dominates time-to-market and it is getting worse over time[1]. Therefore, it is critical to invent effective hardware verification tools to alleviate time and manual effort to trace the root cause of a design bug.

Simulation-based verification has been the most effective technique for system-level verification. To check whether or not the whole system works for real-world workloads, the hardware design is emulated using an FPGA for trillions of cycles. FPGAs deliver much faster simulation than software approaches but suffer from a lack of DUT visibility. DESSERT[2] demonstrates a technique to catch errors and obtain error traces from FPGA-based simulation with assertion synthesis and commit log comparisons. However, bugs are only caught as violations of high-level properties, which need to be manually traced back to the detailed bug location in the source RTL.

There are lots of studies on SAT-based bug localization[345]. The idea is instrument muxes for suspect lines of RTL, and transform the instrumented hardware design into the CNF to let the SAT solver to find bug locations. However, this approach is not scalable for complex hardware designs and long error traces.

In this project, we propose a novel methodology to effectively localize bugs from error traces. Our proposal is localize bugs using fine-grained specs mined from correct traces while catching errors from high-level properties. We collect correct traces from small tests as well as realistic workloads from DESSERT. We also employ template-based spec-mining suggested by Li et. al[6]. Since this approach is computationally efficient, we can train as many simple but fine-grained assertions as possible from long traces. Note that we do not have to merge these simple assertions for more complex properties since these assertions are only used for bug localization, which makes this approach more computationally efficient and justifies adding various templates for effective bug localization.

The timeline for this project is follow:

1. **March**: Vighnesh will develop a tool to convert VCD dumps into module-level delta traces. Donggyu will figure out what templates should be introduced for effective bug localization. We believe assertions in terms of cycles will be very helpful.

2. **April**: We will implement a spec-mining tool using module-level delta traces, and apply this tool for very simple designs. We will also collect correct and error traces for complex designs using DESSERT.

3. **May**: We will present preliminary results with complex designs in the class. We will continue this work even after then and plan to publish this work.

References

[1] H. D. Foster, "Trends in Functional Verification: A 2014 Industry Study," DOI: 10.1145/2745404.2751548Technical. [Online]. Available: http://dx.doi.org/10.1145/2745404.2751548Technical.

[2] D. Kim, C. Celio, S. Karandikar, D. Biancolin, J. Bachrach, and K. Asanovic, "DESSERT: Debugging RTL effectively with state snapshotting for error replays across trillions of cycles," in *Proceedings - 2018 International Conference on Field-Programmable Logic and Applications, FPL 2018*, 2018, ISBN: 9781538685174. DOI: 10.1109/FPL.2018.00021.

[3] A. Veneris, "Fault diagnosis and logic debugging using Boolean satisfiability," in *Proceedings - International Workshop on Microprocessor Test and Verification*, 2003, ISBN: 0769520456. DOI: 10.1109/MTV.2003.1250264.

[4] K. H. Chang, I. Wagner, V. Bertacco, and I. L. Markov, "Automatic error diagnosis and correction for RTL designs," in *Proceedings - IEEE International High-Level Design Validation and Test Workshop, HLDVT*, 2007, ISBN: 1424414806. DOI: 10.1109/HLDVT.2007.4392789.

[5] S. Mirzaeian, F. Zheng, and K.-t. Tim Cheng, "RTL Error Diagnosis Using a Word-Level SAT-Solver," Tech. Rep.

[6] W. Li, A. Forin, and S. A. Seshia, "Scalable specification mining for verification and diagnosis," 2010. DOI: 10.1145/1837274.1837466.