Electronic Design Automation
TUM Department of Electrical and Computer Engineering
Technical University of Munich

Forschungspraxis, Interdisciplinary Project, Bachelor's Thesis, Master's Thesis

Accelerating Fault Simulation at RTL on GPU Compute Clusters

One of the crucial tasks in designing, testing, and verifying a digital system is the early estimation of its fault tolerance. For this, fault injection simulations can be used to evaluate this tolerance at different phases of development. At the Register Transfer Level (RTL), an existing but not yet implemented hardware design can be simulated with higher accuracy than its simulation at the instruction or algorithm level. However, accuracy comes at a cost that grows with the number of simulations done within a fault injection analysis. For example, fault injection simulation of a CPU at RTL instead of Instruction Set Architecture (ISA) level would increase the simulation effort to include micro-architectural registers (pipeline, functional units, etc.),

To allow faster fault space exploration at RTL, one could do the following: (a) accelerate the simulation of an individual Device Under Test (DUT) and fault, or
(b) launch multiple fault simulations concurrently

In the case of (a), state-of-the-art research on fast RTL simulations has aimed to reduce simulation cost by multi-threaded simulation on CPUs [1][2][3] or GPUs [4][5][8].

In case (b), multiple independent simulations may be launched simultaneously on a distributed compute cluster. Parallelization of an individual simulation is not as substantial, since the computational platform is utilized anyway, i.e., task-level parallelism of individual simulations for long simulations: one CPU core per fault experiment.

Existing solutions for (b) mainly aim for CPU-based clusters [6][7], whereas for (a), the maximum speed of a single DUT simulation is required. In this work, we want to explore efficient fault simulation at RTL on GPU-based compute clusters that maximizes utilization with the number of individual experiments launched.

Tasks:

1. Set up GPU-accelerated RTL simulation of [8][9]
2. Implement a new fault injection arbitration framework for the RTL simulator
3. Explore multi-experiment partitioning for fault simulation on the new platform
4. Compare and benchmark against a CPU-based fault exploration scheme [6][7]

References:

- [1] W. Snyder, P. Wasson, D. Galbi, et al. Verilator. https://github.com/verilator/verilator, 2019. [Online].
- [2] S. Beamer and D. Donofrio, "Efficiently Exploiting Low Activity Factors to Accelerate RTL Simulation," 2020 57th ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 2020, pp. 1-6, doi: 10.1109/DAC18072.2020.9218632.
- [3] Kexing Zhou, Yun Liang, Yibo Lin, Runsheng Wang, and Ru Huang. 2023. Khronos: Fusing Memory Access for Improved Hardware RTL Simulation. In Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO '23). Association for Computing Machinery, New York, NY, USA, 180–193. https://doi.org/10.1145/3613424.3614301

- [4] H. Qian and Y. Deng, "Accelerating RTL simulation with GPUs," 2011 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), San Jose, CA, USA, 2011, pp. 687-693, doi: 10.1109/ICCAD.2011.6105404.
- [5] Dian-Lun Lin, Haoxing Ren, Yanqing Zhang, Brucek Khailany, and Tsung-Wei Huang. 2023. From RTL to CUDA: A GPU Acceleration Flow for RTL Simulation with Batch Stimulus. In Proceedings of the 51st International Conference on Parallel Processing (ICPP '22). Association for Computing Machinery, New York, NY, USA, Article 88, 1–12. https://doi.org/10.1145/3545008.3545091
- [6] Johannes Geier and Daniel Mueller-Gritschneder. 2023. VRTLmod: An LLVM based Open-source Tool to Enable Fault Injection in Verilator RTL Simulations. In Proceedings of the 20th ACM International Conference on Computing Frontiers (CF '23). Association for Computing Machinery, New York, NY, USA, 387–388. https://doi.org/10.1145/3587135.3591435
- [7] J. Geier, L. Kontopoulos, D. Mueller-Gritschneder and U. Schlichtmann, "Rapid Fault Injection Simulation by Hash-Based Differential Fault Effect Equivalence Checks," 2025 Design, Automation & Test in Europe Conference (DATE), Lyon, France, 2025, pp. 1-7, doi: 10.23919/DATE64628.2025.10993266.
- [8] Guo, Zizheng, et al. "GEM: GPU-Accelerated Emulator-Inspired RTL Simulation.https://guozz.cn/publication/gemdac-25/gemdac-25.pdf
- [9] Github NVLabs https://github.com/NVlabs/GEM

Prerequisites

- Excellent C++, Python
- Good understanding of GPU programming (CUDA) or interest to learn
- Decent knowledge of hardware design languages (Verilog, VHDL) and EDA tools (Vivado, Yosys)
- Decent knowledge of Statistics and probability

Contact

Apply with CV and Transcript of Records directly to:

johannes.geier@tum.de

Advisors

Johannes Geier