EE290C Project Proposal: Performance and Energy Characterization of Gemmini, BOOM, and NVDLA Using State-Sampling from Firesim FPGA Simulation

Vighnesh Iyer, Billy Chau

1 Background and Motivation

The architecture and VLSI literature has produced a litany of dedicated ML accelerators over the last 3 years. Most of these accelerators exploited unique dataflows, weight/activation sparsity, integer arithmetic, novel circuit techniques, or many other techniques to achieve a claimed perf/watt advantage over prior work.

However, architecture papers typically measure the energy consumption of the proposed accelerator against a baseline using a (Python/C++) model of their accelerator, and the energy numbers may suffer from poor accuracy. Some papers use HLS to generate RTL implementations of different architectures but the energy calculation from a synthesized netlist and RTL simulation is limited by the speed of simulation, and thus only small networks can be evaluated at the RTL-power-estimation-level.

On the other hand, chip/VLSI papers measure the energy consumption of their accelerator empirically using the taped out chip, but do not have a reference accelerator implementation taped-out to compare against. Furthermore, once a chip is taped out, the power consumption cannot be measured for individual parts of the accelerator (at module-granularity) and cycle-by-cycle energy numbers cannot be extracted.

2 Prior Work

Strober[6] is a framework which can generate fast and cycle-accurate sample-based energy simulator using Chisel for any arbitrary RTL running on FPGA. It has achieved less than 5 percents error with 99.9 percents confidence against commercial CAD tools with more than two orders of magnitude speedup over existing microarchitectural simulators and four orders of magnitude speedup over commercial Verilog simulators.

Highlights

State Snapshotting using Scan Chain - As detailed information of simulation is required for an accurate power model, a basic scan chain is embedded into the design to capture a replayable RTL snapshot with both register and SRAM values. Using the hardware construction language Chisel, all custom transforms and FAME1 transform can be easily mapped and wrapped around the RTL design.

Sample-based Energy Simulation - Based on the central limit theorem of statistics, a representative estimator of the power model can be generated given random sampling and enough snapshot samples. However, knowing the length of the program's execution is impossible, so the reservoir sampling technique is used to address this problem.

Fine-grained level Simulation and Checking - Independent replayable snapshots are replayed in parallel in commercial gate-level simulators such as VCS for accurate power analysis. To ensure the correctness of the execution, the output values of the design are compared with the output traces.

Simmani[5] is an FPGA-accelerated framework which automatically selects signals most correlated with power dissipation and trains power models in terms of the selected signals for any RTL design. It has achieved accurate power models with acceptable errors on real world machine learning application such as SqueezeNet running with Rocket Core and Hwacha Vector Accelerator.

Highlights

Activity Counter Insertion and FPGA accelerated simulation - Utilizing the Strober framework, any RTL designs are automatically transformed for FPGA-accelerated RTL simulation. The framework is also used to obtain runtime power traces from FPGAs by inserting activity counters.

Toggle Density Matrix with Principal Components Projection - In order to train an accurate power model, toggle pattern matrix is used to represent signals and their toggle frequencies. The intuition behind is that signals showing similar toggle patterns have similar effect on dynamic power dissipation and can be factored to share the same coefficient in the power model, thus minimizing modeling error. However, it suffers from the curse of dimensionality like other machine learning algorithms; therefore, SVD-based dimensionality reduction algorithm is used to extract the top k dimensions.

Automatic Signal Selection - Given N, the number of signals selected from Bayesian Information Criterion, k-means++ algorithm is run to cluster the signals into groups i.e. signals with similar toggle pattern are grouped together. Then, the signals that are the closest to the center of each cluster are selected, which will be the regression variables in power model training.

Power Model Regression - Using Elastic Net for variable regularization and selection with k-fold cross-validated hyperparameters, a regression model is trained using the toggle density matrix and the groud truth power traces obtained from commercial CAD tools calculated from RTL VCD dumps.

DESSERT[4] is an FPGA-accelerated framework for effective simulation-based RTL verification and debugging. It has achieved almost no performance overhead with hardware-based assertion checking and insignificant performance overhead for software-based exhaustive checking in comparison to commercial CAD tools.

Highlights

Error Capturing using Scan Chain - Following the technique in the Strober framework, scan chain based snapshotting is used for error capturing in DESSERT. There are two constructs supporting assertion and logs in FIRRTL: stop and printf, and they are mapped automatically from the source code by DESSERT. In order to detect and replay errors efficiently, two identical and deterministic simulators are run in parallel. The leading master instance is to detect the target RTL bugs while the lagging instance is to checkpoint the target RTL state.

Software-based Golden Model Checking - When the logs are generated from FPGA, they are sent to the buffers in the software simulation driver through DMA. They are then compared with a software-based golden model such as Spike for exhaustive error checking.

3 Proposal

We propose a framework that enables:

- RTL-fidelity simulation of ML accelerators over full DNN inference passes

  We plan to re-use the Firesim infrastructure that automatically transforms arbitrary RTL to a deterministic, cycle-accurate model that executes on an FPGA.

- Gate-level fidelity energy modeling (with module-level granularity)

  We plan to modify and add passes to Golden Gate that add a stitched scan chain and SRAM hijack ports that allows the host to inject and pull all μArch state to and from the DUT. This technique is distinct from DESSERT and STROBER since those works use a shadow scan chain that cannot inject state into the DUT, and may be more resource intensive.

  This allows us to advance simulation time, pause at random intervals, and dump the DUT state to the host, similar to STROBER. The randomly sampled RTL-level activity traces are formally mapped into gate-level traces and fed into Voltus with a gate-level synthesized netlist for cycle-by-cycle energy numbers.

- Performance instrumentation and bottleneck analysis (PE utilization, memory traffic analysis)

  We plan to use the auto-instrumentation (AutoCounter) feature in FirePerf[3] to capture design-specific performance counters to evaluate bottlenecks in DNN inference. We also plan to instrument the off-chip memory interface to record the DRAM address and transfer size access patterns.

- Energy estimation of DRAM accesses

  The memory access trace can be replayed in DRAMSim2[7] to extract average power numbers across time. DRAM access energy is often neglected in RTL-level power analyses and isn't done in the STROBER flow.

- Exploration of the impact different dataflows and tiling patterns on energy and performance

  Finally, once the framework described has been implemented, it can be used to guide software optimizations.

We plan to use this framework to evaluate Gemmini[2] and BOOM, and if time permits, NVDLA on Firesim[1].

4 Project Infrastructure

For the initial phase of the project, we don't require any special infrastructure apart from our laptops and the BWRC servers for Genus/Jasper/Voltus (we hope VPN works). Once software simulation is successful, we will use some AWS F1 instances.

5 Project Timeline

1. **Checkpoint 1 (4/10)**: RTL simulation of a simple circuit (Risc/GCD) after default Firesim transformation, ASAP7 Genus synthesis of the circuit and SRAM macros, formal mapping of RTL VCD to gate-level VCD, scan-chain stitching and SRAM hijack FIRRTL pass working in RTL simulation

2. **Checkpoint 2 (4/24)**: Sample circuit simulating on F1 FPGA, μArch state dumping, initial energy estimation evaluation

3. **Final Report (5/8)**: Rocket with Gemmini simulating on F1 FPGA, DRAM interface and Gemmini-specific performance instrumentation, Gemmini energy estimation for pre-written ResNet50 and Mobilenet implementations, DRAMSim2 energy estimation

The timeline is quite aggressive and there may be points where we find ourselves stuck. It is likely that the custom FIRRTL passes introduce difficult to find bugs and may delay this timeline. It is not clear how to perform formal mapping of RTL VCDs to gate VCDs, but there's probably a Cadence RAK for it.

References

[1] Farzad Farshchi, Qijing Huang, and Heechul Yun. *Integrating NVIDIA Deep Learning Accelerator (NVDLA) with RISC-V SoC on FireSim.* 2019. arXiv: `1903.06495 [cs.DC]`.

[2] Hasan Genc et al. *Gemmini: An Agile Systolic Array Generator Enabling Systematic Evaluations of Deep-Learning Architectures.* 2019. arXiv: `1911.09925 [cs.DC]`.

[3] Sagar Karandikar et al. "FirePerf: FPGA-Accelerated Full-System Hardware/Software Performance Profiling and Co-Design". In: *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems.* ASPLOS '20. Lausanne, Switzerland: Association for Computing Machinery, 2020, pp. 715–731. ISBN: 9781450371025.

[4] Donggyu Kim et al. "DESSERT: Debugging RTL Effectively with State Snapshotting for Error Replays across Trillions of Cycles". In: *2018 28th International Conference on Field Programmable Logic and Applications (FPL)*. IEEE, Aug. 2018.

[5] Donggyu Kim et al. "Simmani: Runtime Power Modeling for Arbitrary RTL with Automatic Signal Selection". In: *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2019, Columbus, OH, USA, October 12-16, 2019.* ACM, 2019, pp. 1050–1062.

[6] Donggyu Kim et al. "Strober: Fast and Accurate Sample-Based Energy Simulation for Arbitrary RTL". In: *43rd ACM/IEEE Annual International Symposium on Computer Architecture, ISCA 2016, Seoul, South Korea, June 18-22, 2016.* IEEE Computer Society, 2016, pp. 128–139.

[7] P. Rosenfeld, E. Cooper-Balis, and B. Jacob. “DRAMSim2: A Cycle Accurate Memory System Simulator”. In: *IEEE Computer Architecture Letters* 10.1 (Jan. 2011), pp. 16–19. ISSN: 2473-2575.