Nanoscale Integrated Circuits and System Lab

Kaiyuan Guo, Yu Xing, Shulin Zeng, Yu Wang
Electronic Engineering, Tsinghua University, zengsl18@mails.tsinghua.edu.cn

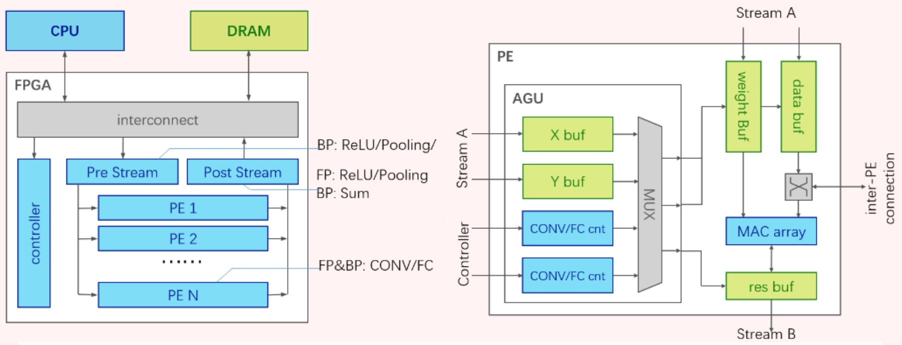
Programs

- (Work 1) Compressed CNN Training with FPGA-based Accelerator.
- **Challenges**: 1. Using quantization and pruning in early stage is risky 2. Result-sparse pattern and sparse matrix transportation in BP phase 3. Variety loop dimensions of FP & BP.
- (Work 2) DNNVM: End-to-End Compiler Leveraging Operation Fusion on FPGA-based CNN Accelerators.
- **Challenges**: 1. Full-stack compiler for ISA-based Single Computation engine (SGE) 2. Operation Fusion introduces huge design space exploration.
- (Contest) **1st place** of 55th DAC system design contest.

Work 1: Training on FPGA

- We propose a **hardware friendly training process** with advanced quantization and pruning. Specially, we explore the effect of when to apply quantization and pruning.
- We design dedicated processing elements (PEs) on FPGA to support both **operator-sparse** and **result-sparse** patterns. The sparse matrix transposition function is supported by specific scheduling method with a novel data organization in external memory.
- We design **configurable loop mapping strategy** for both FP&BP CNN computation.
- Experimental results show that the proposed accelerator achieves **641GOP/s** equivalent performance and **3x** better energy efficiency compared with GPU.

Work 1: Architecture & Result

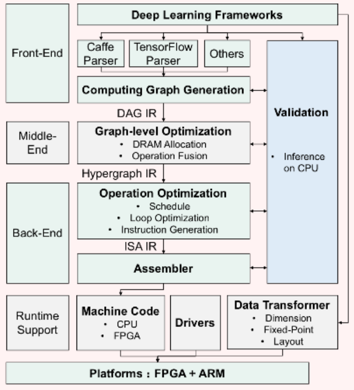


	Data Format data+gradiant	Forward time (us)	Backward time (us)	Total time (us)	Performance (GOP/s)	Power (W)	Power efficiency (GOP/s/W)
GPU	float + float	7795	15700	23495	1252.7	150	8.4
FPGA	8bit + 24bit	10931	34981	45912	641.1	29	22.1

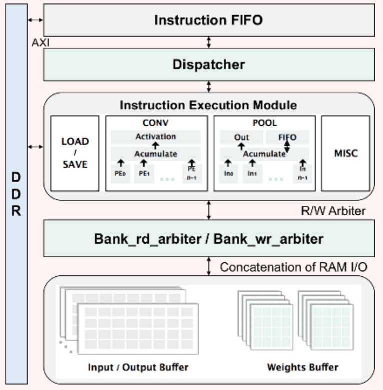
Work 2: DNNVM

- **DNNVM** is a completely end-to-end compilation infrastructure that firstly transforms algorithms from different deep learning frameworks into our domain-specific directed acyclic graphs(DAGs) called **XGraph**.
- Based on **XGraph**, we design some heuristic fusion rules and a set of fusion templates to leverage the operation fusion on the entire computation graph.
- We propose a heuristic subgraph isomorphism algorithm to enumerate fusion candidates instead of greedy algorithms. Then, we evaluate the performance of each fused subgraph and adopt an efficient heuristic shortest-path algorithm to find the optimal execution strategy.
- Our experiments demonstrate that leveraging operation fusion achieves up to **2.3x local speedup** compared with baseline without fusion. We achieve the throughput for VGG to be **334 GOP/s on ZU2@330MHz** and **2.82 TOP/s on ZU9@330MHz**, with the energy efficiency to be **44.5** and **123.7 GOPS/W** respectively. We get **1.11X** better performance on **ZU9** than xfDNN on VU9P@500MHz.

Work2: Workflow & Hardware



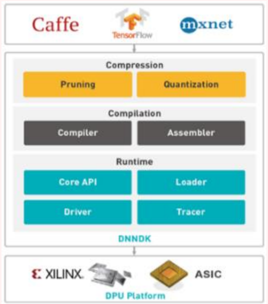
DNNVM Workflow



Hardware Design

System Design Contest

- Algorithm: SSD optimization. Fixed point training(INT8) w/ Compression
- Software: System & Functional level optimization with **DeePhi DNNDK**
- Hardware: **DeePhi DPU IP** on PYNQ
- Result(PYNQ):**0.62 IOU/4.2W/12fps**
  Result(GPU):0.69 IOU/12.6W/24fps
- https://github.com/hirayaku/DAC2018-TGIIF



Acknowledgements

These programs are supported by Beijing Research Center for Information Science and Technology (BNRist) and DeePhi (Part of Xilinx)