Laconic Deep Learning Computing

The Edward S. Rogers Sr. Department of Electrical & Computer Engineering
UNIVERSITY OF TORONTO

Sayeh Sharify, Mostafa Mahmoud, Alberto Delmas Lascorz, Milos Nikolic, Andreas Moshovos

Abstract

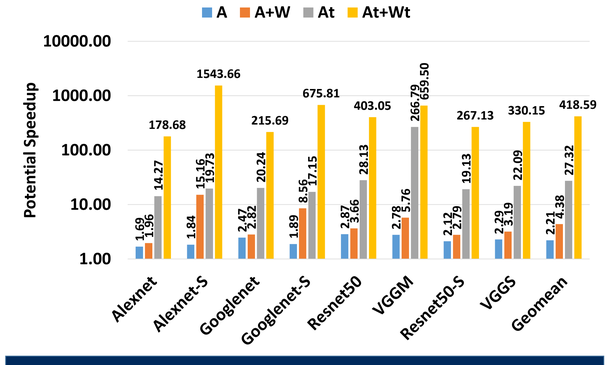
Laconic:
Deep Neural Network inference accelerator

- Targets primarily CNNs, can do any layer type
- Exploits *effectual bit content for activations and weights*
- Term-serial accelerator

20.5x faster than data-parallel accelerators
2.6x more energy efficient
26% area cost

Motivation

- Previous accelerators exploit ineffectual computations including zero skipping, precision variability, and zero bit skipping to improve the performance and energy efficiency of CNNs.
- We show that policy **"At+Wt",** eliminating the ineffectual computations at the bit level for both the activations and the weights, has the highest potential speedup.



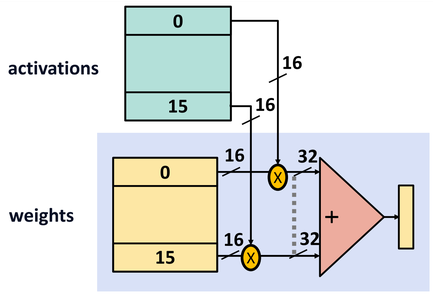
Laconic's Approach

$$\begin{matrix}01010110\\00101010\end{matrix} \quad \frac{8\times8}{4\times3} = 5.3\times$$

- ***Performance gain***
  - Convolutional layer: $\frac{P_{base}\times P_{base}}{\langle t_a\rangle\times\langle t_w\rangle}$
  - Fully-connected layer: $\frac{P_{base}}{\langle t_w\rangle}$
- ***Data access reduction***
  - Activations: $\frac{P_{base}-P_a}{P_a}$
  - Weights: $\frac{P_{base}-P_w}{P_w}$

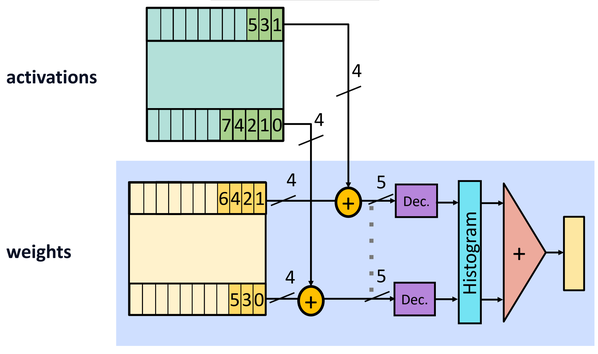
Baseline

Conventional bit-parallel accelerator
Execution time does not vary with data bit content



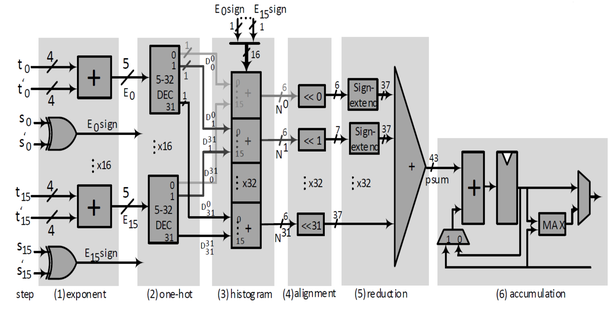
Laconic: Term-Serial Accelerator

Execution time varies with the number of effectual terms



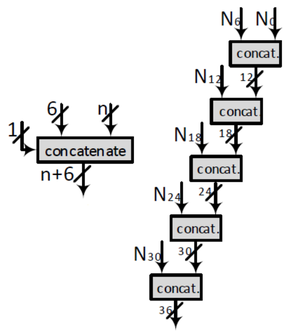
Laconic Processing Element

- Inputs:
  - 16 Activations, 1 term/cycle
  - 16 Weights, 1 term/cycle
- Multiplication is done term-serially
- Reduces the products to a single output
- Supports cascading for smaller layers



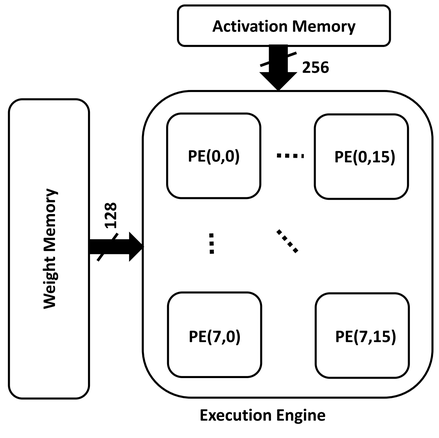
Enhanced Adder Tree

- ***More area and energy efficient adder tree***
- Divides the output of histogram stage to 6 groups, G0, G1, …, G5.
- Outputs within the same group have no overlapping bits that are "1".
- Concatenates the outputs within a group to calculate their sum.

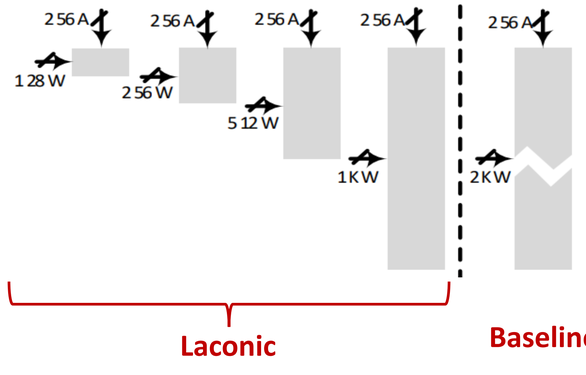


$$G_0 = \{N^{30}, N^{24}, N^{18}, N^{12}, N^{6}, N^{0}\}$$
$$G_1 = \{N^{31}, N^{25}, N^{19}, N^{13}, N^{7}, N^{1}\}$$
$$G_2 = \{N^{26}, N^{20}, N^{14}, N^{8}, N^{2}\}$$
$$G_3 = \{N^{27}, N^{21}, N^{15}, N^{9}, N^{3}\}$$
$$G_4 = \{N^{28}, N^{22}, N^{16}, N^{10}, N^{4}\}$$
$$G_5 = \{N^{29}, N^{23}, N^{17}, N^{11}, N^{5}\}$$

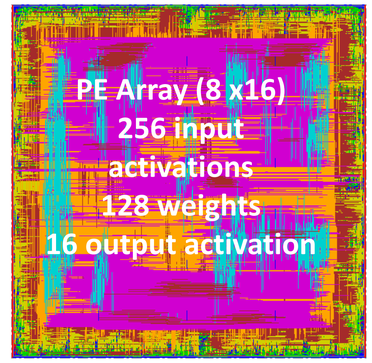
Laconic Architecture



Design Configurations



Datapath Layout



- 65nm TSMC
- Clock frequency: 980 MHz
- 128W Tile:
  - Area: 943422.48 μm²
  - Power: 224.1 mW
- Operations Per Second: **752 GOPS**
- Operations Per Second, Per Watt: **763.4 GOPS/W**

Results Relative to Baseline

Throughput:

S: Sparse

Energy efficiency:

S: Sparse