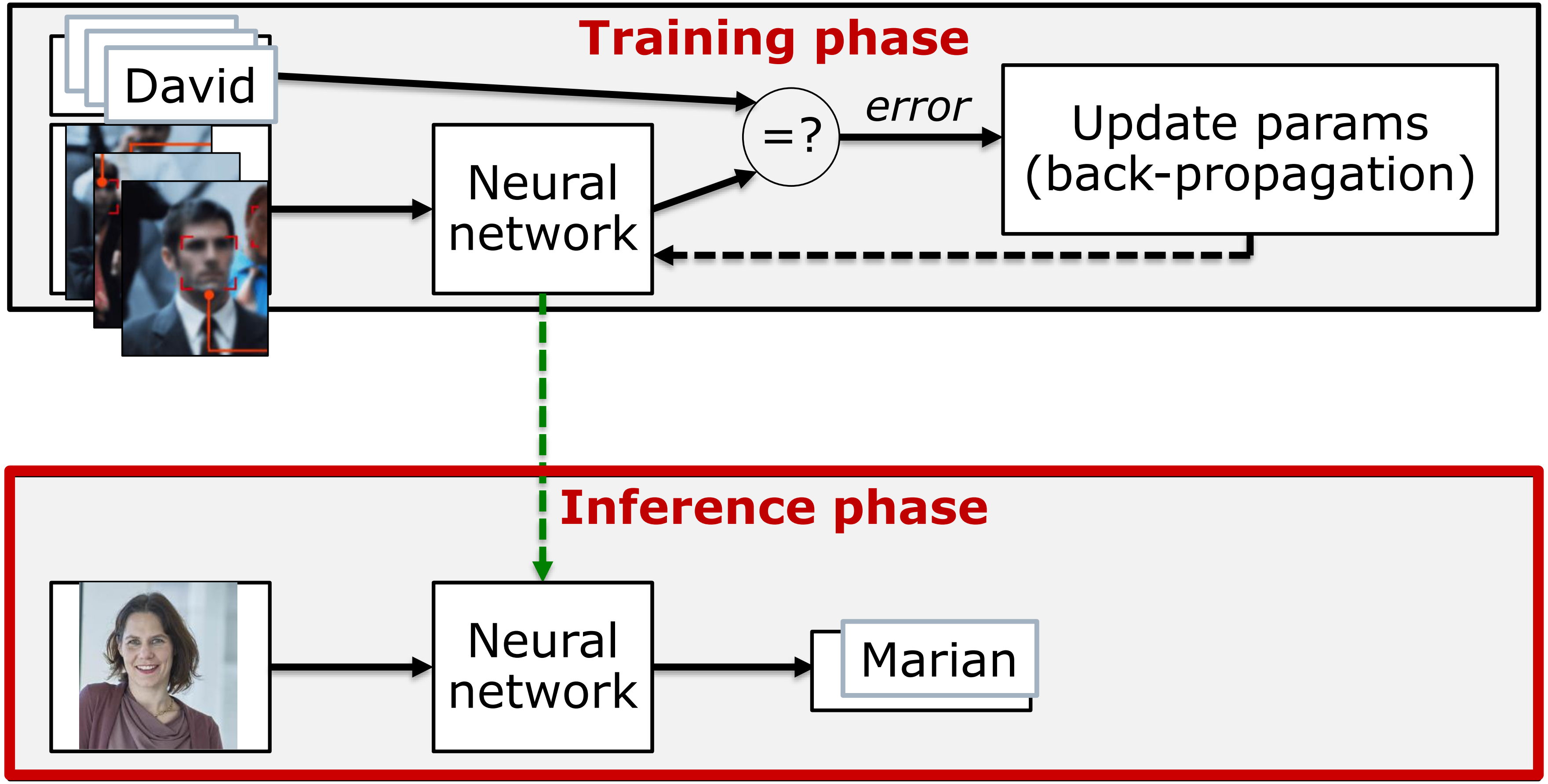
ISSCCedu 2018:

Efficient hardware implementation of deep neural network processing

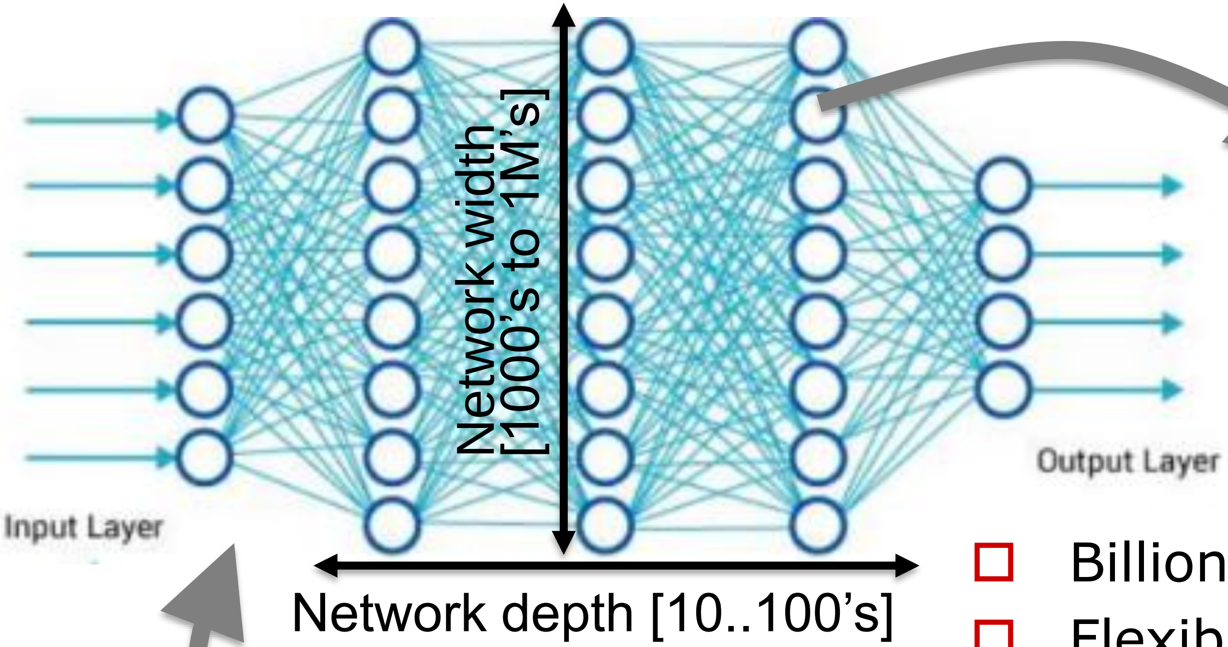
Marian Verhelst
MICAS, KU Leuven, Belgium
Marian.Verhelst@kuleuven.be

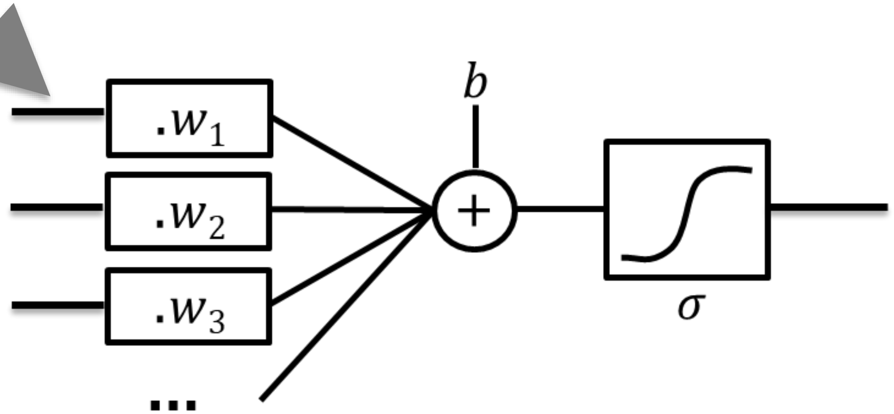
The rise of deep neural networks (NN)

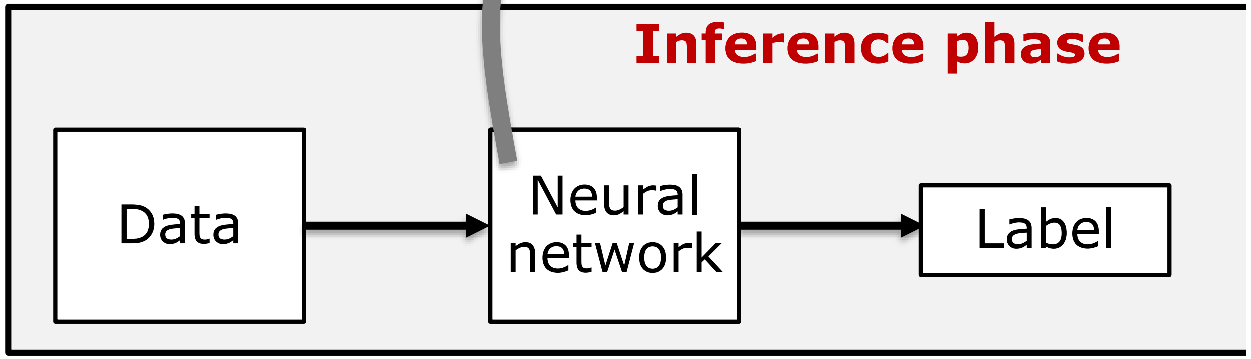


Energy efficiency!

Deep NN inference workload



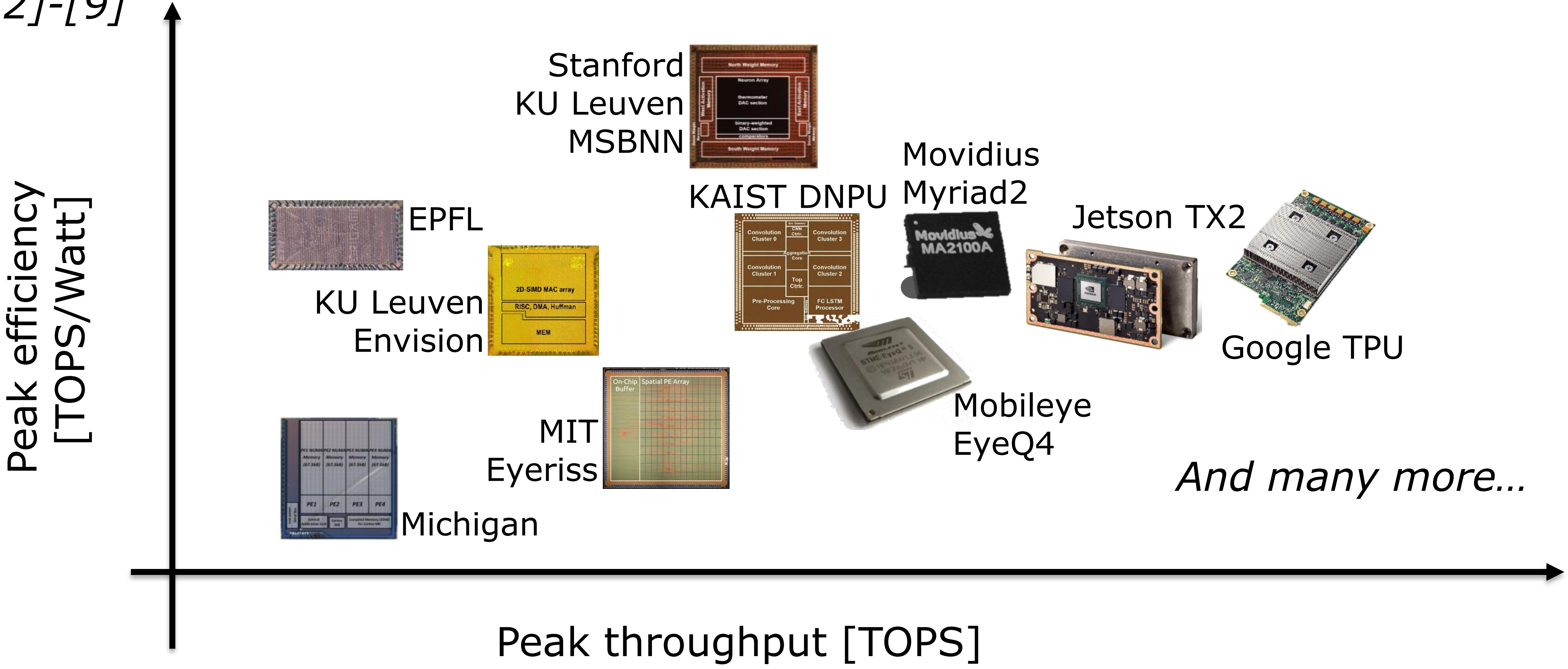




- Billions of multiply-accumulate (MAC)
- Flexibility regarding:
  - Network depth
  - Network width
  - Layer structure (Inception, Resnet, DenseNet, Xception, CapsNet,… [1])
- Need for custom programmable platform

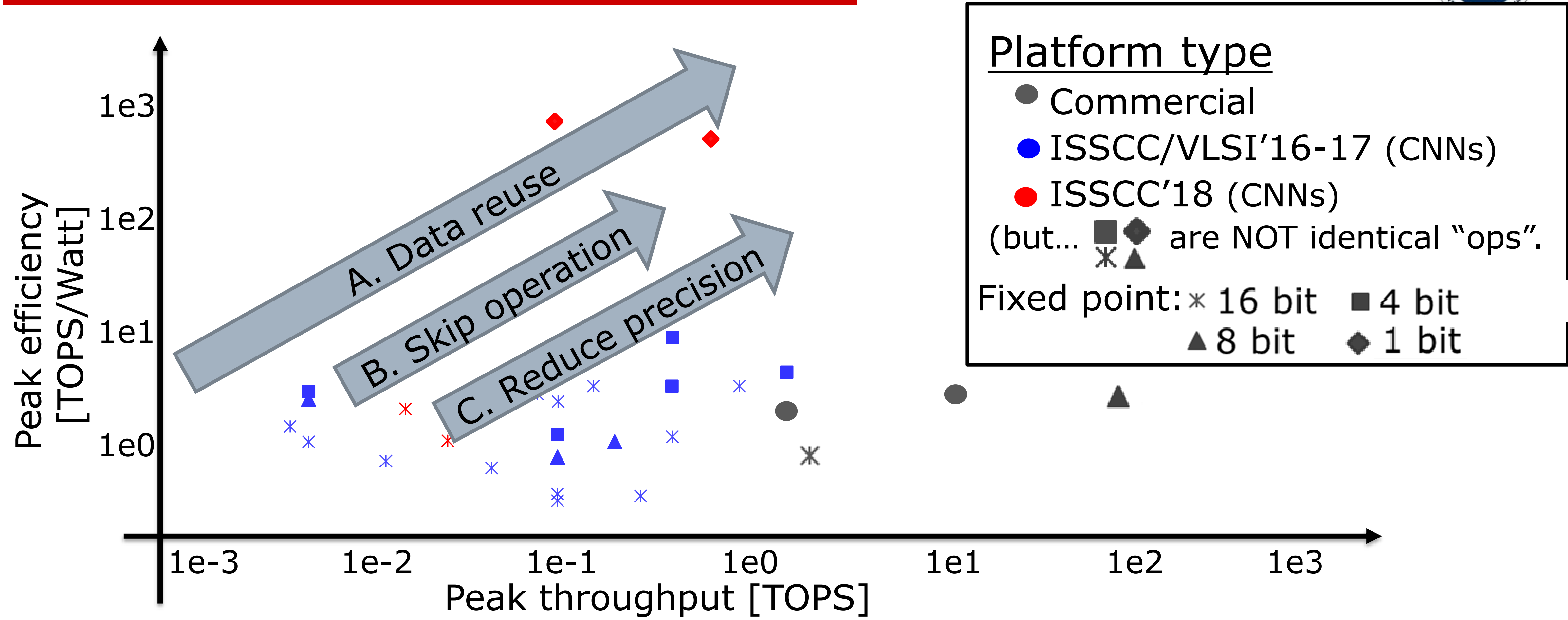
The zoo of deep neural network processors

Refs [2]-[9]

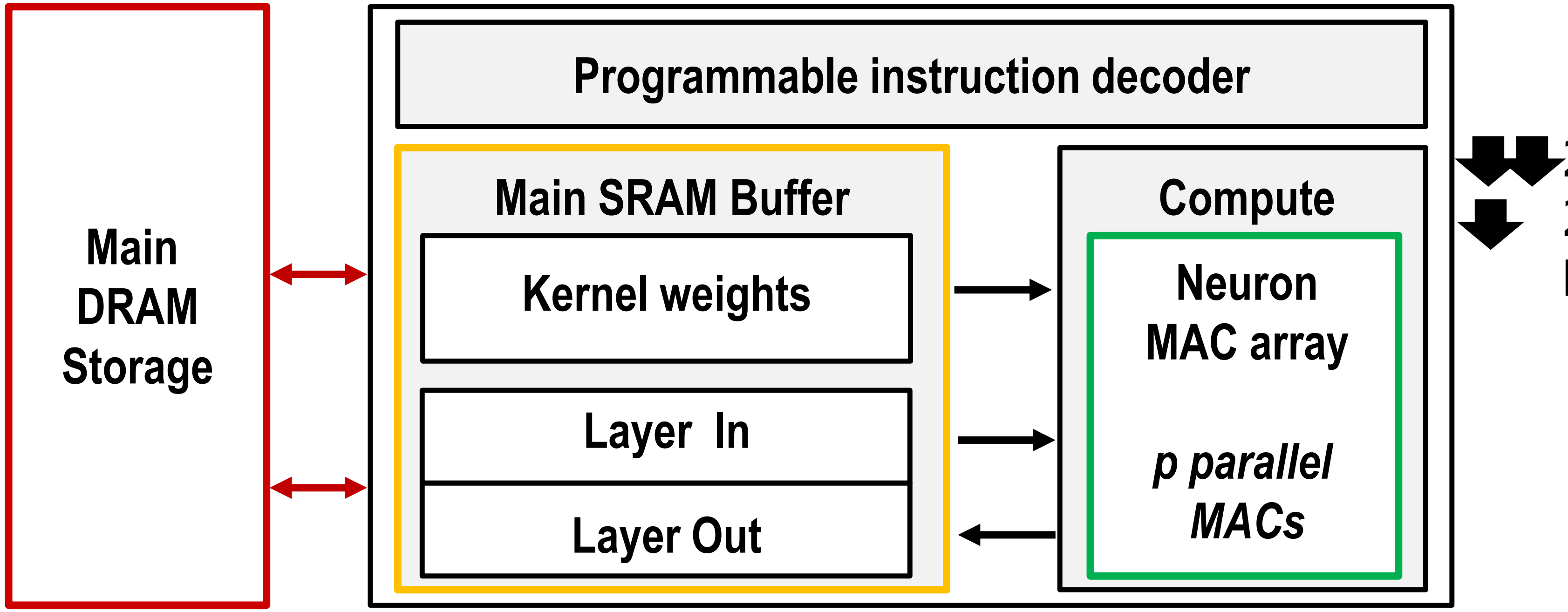


- □ CPU ➔ (embedded) GPU, tensor processing units (TPU) and other accelerators

The zoo of deep neural network processors [1,12]



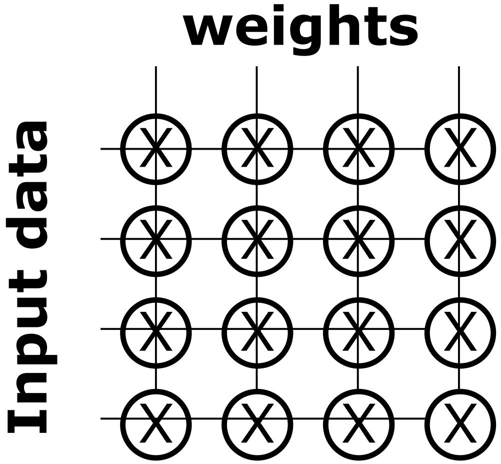
Deep NN processor architectures: A) data reuse



2N x DRAM access: 500pJ

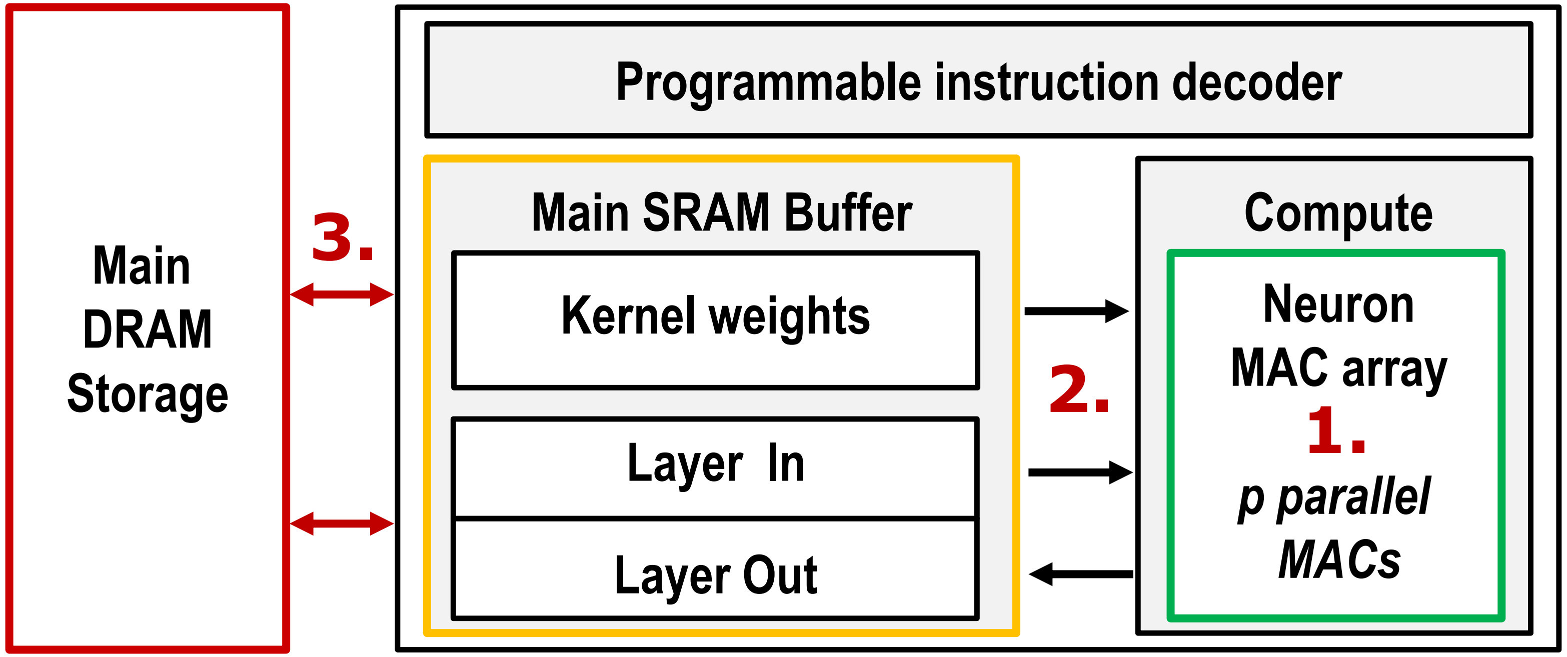
2N x SRAM access: 10pJ

N x Multiply-accum: 5pJ



- □ Avoid extensive off-chip & memory communication
  - ■ Memory hierarchy [7]
- □ Humongous MAC arrays & systolic arrays [6] (eg. TPU: 65,536 [8])
- □ Beware: hardwired data flows are efficient, but not versatile

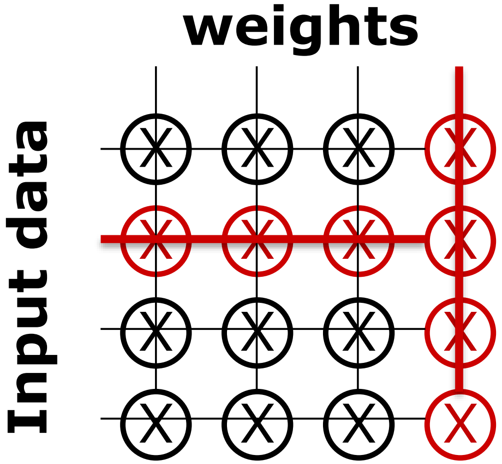
Deep NN processor architectures: B) skip operations



⬇2N x DRAM access: 500pJ

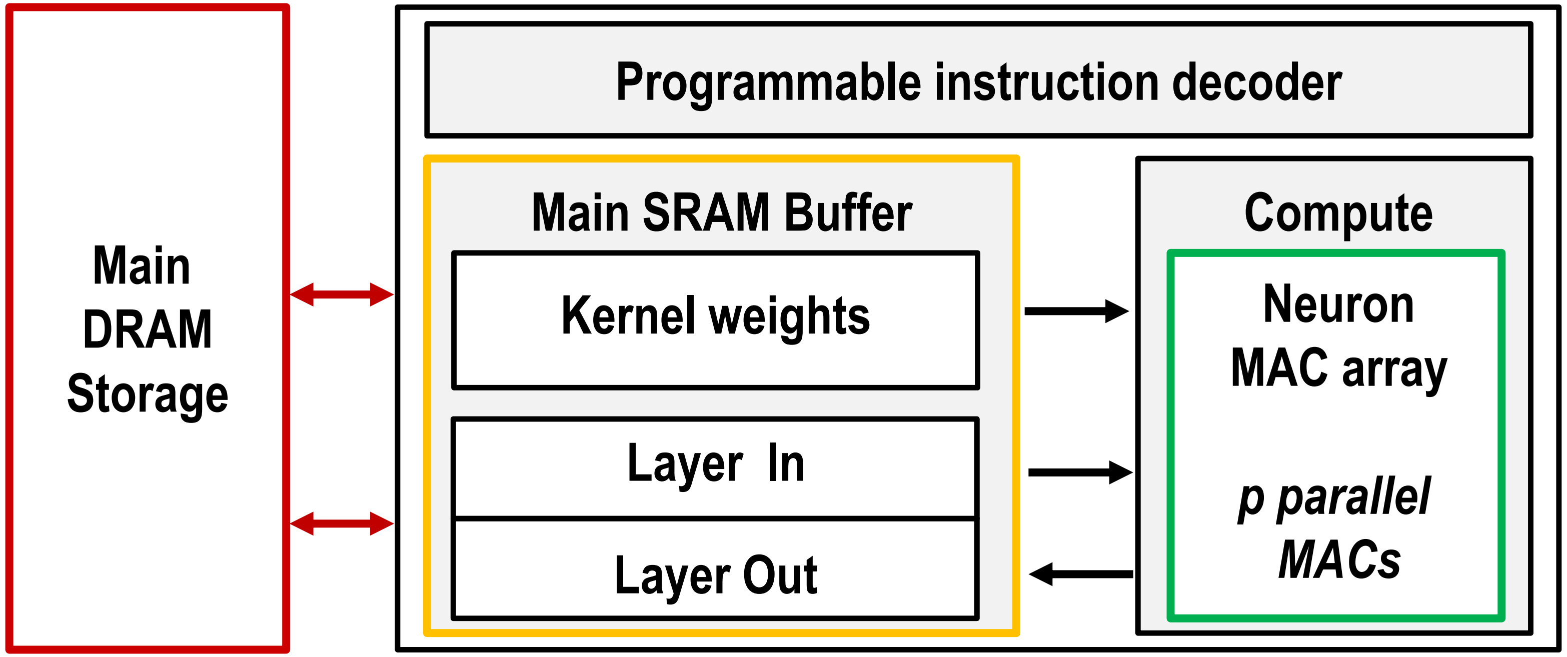
⬇2N x SRAM access: 10pJ

⬇N x Multiply-accum: 5pJ



- □ Many weights and data value are zero [12,13]
  1. Skip multiply accumulate with "0"
  2. Skip reading of zero's from memory
  3. Highly compressive DRAM read/write ➔ encode I/O data

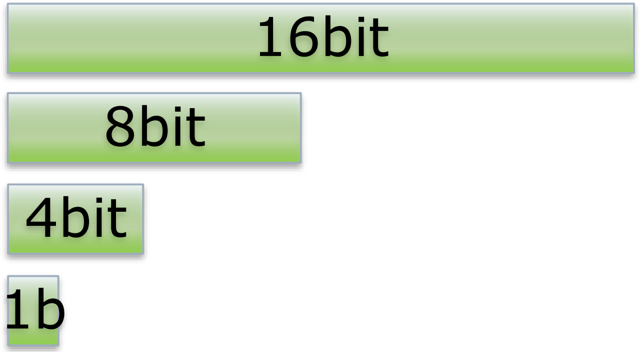
Deep NN processor architectures: C) reduce precision



2N x DRAM access: 500pJ ⬇

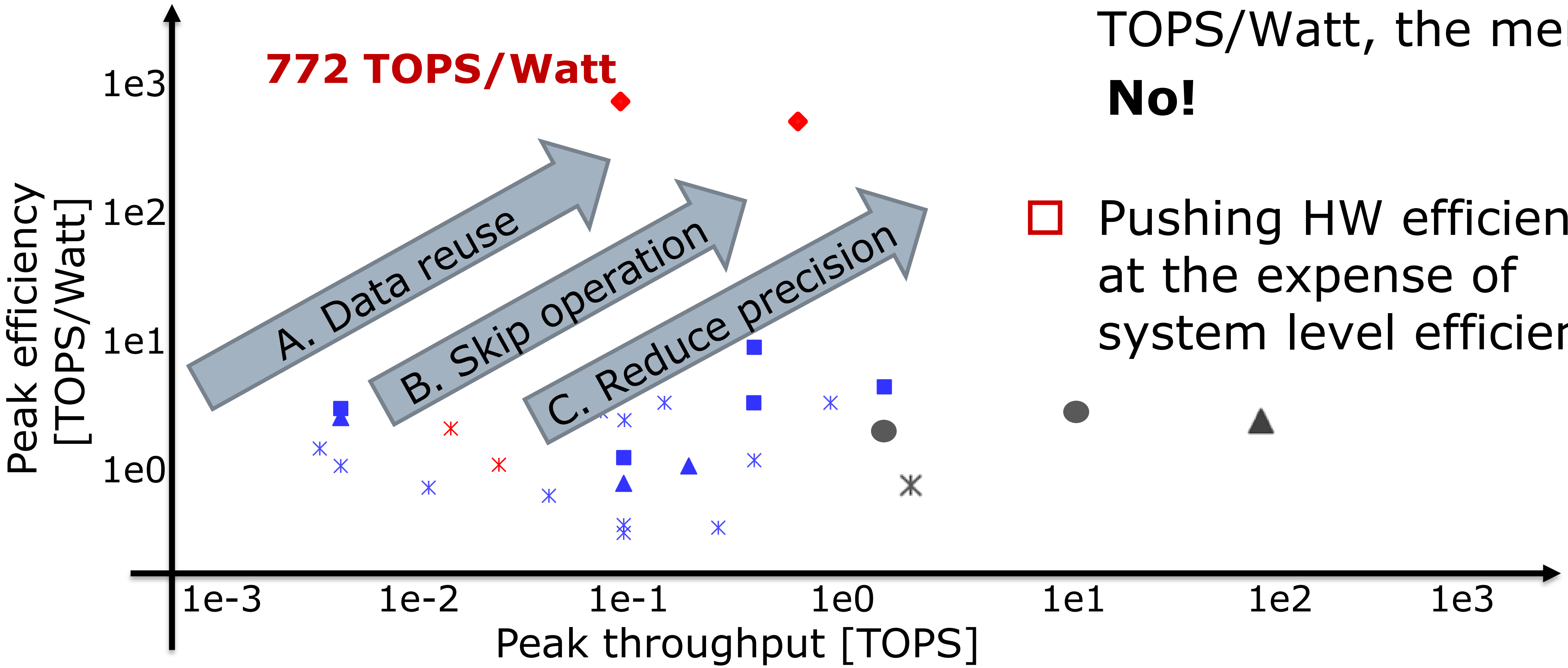
2N x SRAM access: 10pJ ⬇

N x Multiply-accum: 5pJ ⬇⬇



- □ Quantize data to K bits fixed point (K=8b, 4b, even 1b)
  - ■ Extreme = BinaryNets ➔ multiply = XNOR
- □ Reduces both memory access cost & compute cost [2,11,12]

The holy grail of TOPS & TOPS/Watt?!

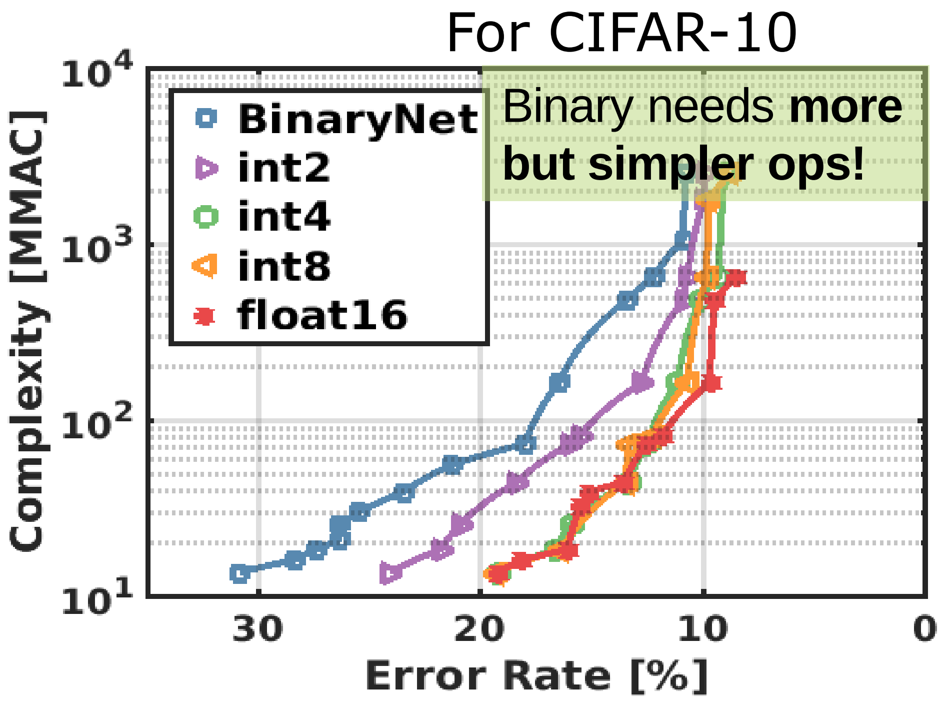


- □ The more TOPS and TOPS/Watt, the merrier?

  No!

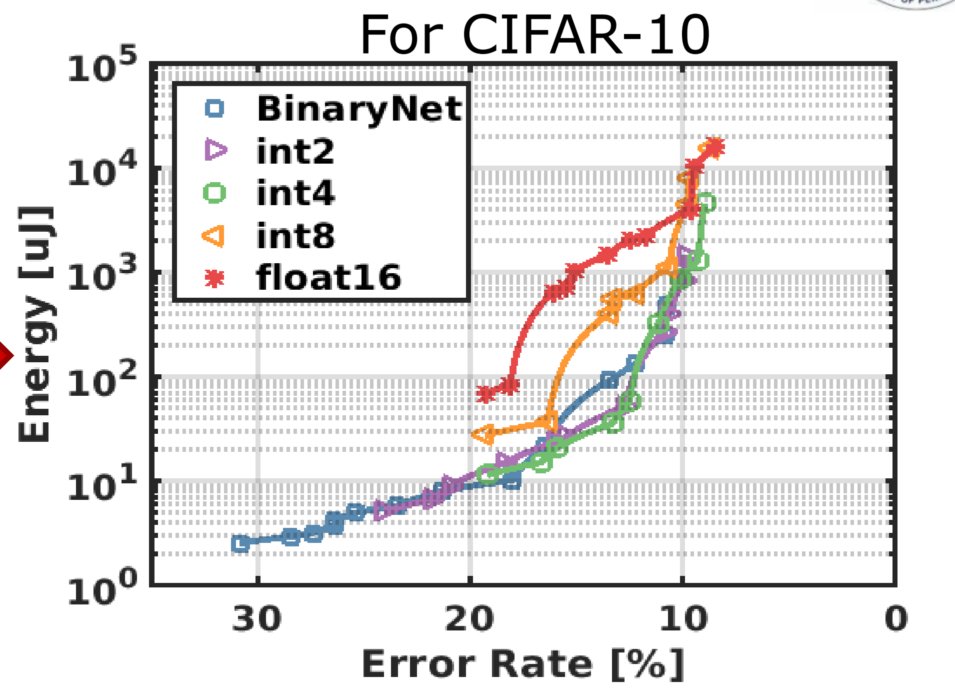
- □ Pushing HW efficiency at the expense of system level efficiency!

Trading off network complexity and precision [11]


For CIFAR-10

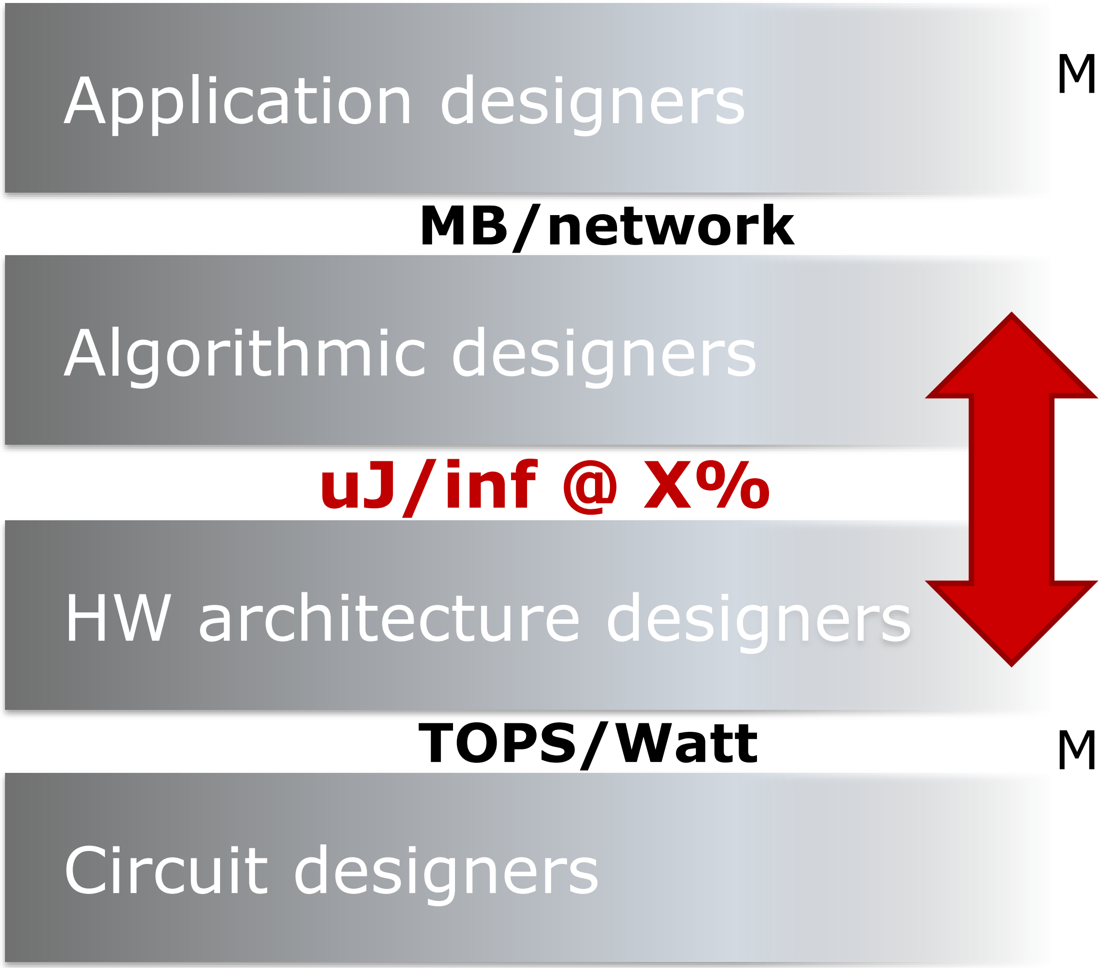
Binary needs **more but simpler ops!**

E_{MAC}, E_{mem}


For CIFAR-10

- □ Network structure allows to trade compute load vs. accuracy (mobilenet, densenet,…)
- □ Low precision networks need deeper and wider networks for same task accuracy
- □ Use **hardware energy model** for **HW-algorithm co-optimization**
  - Simple tasks: 4bits <> More complex tasks optimum at more bits

Conclusion:
How to fairly measure efficiency?



Application designers

MB/network

Algorithmic designers

uJ/inf @ X%

HW architecture designers

TOPS/Watt

Circuit designers

Minimize network size for given accuracy

A. Play with network topology [1]
B. Play with pruning, clustering, …

Minimize energy per inference (task) [11][13]

- *E.g. 10uJ/inf @ 86% CIFAR10*
- On standardized benchmarks
- HW-algorithm co-optimization
- Flexible HW

Maximize operations/Watt [12]

A. Play with computational precision
B. Play with data flow and parallelism
C. Play with guarding data fetches and compute

Key References

[1] B. Moons, CNN architecture comparison: github.com/BertMoons/Comparing-CNN-Architectures

[2] B. Moons, R. Uyterhoeven, W. Dehaene, M. Verhelst ,"ENVISION: A 0.26-10 TOPS/W Subword-Parallel, Computational Accuracy-, Voltage- and Frequency-Scalable Convnet Processor in 28nm FDSOI", ISSCC2017.

[3] D. Bankman, L. Yang, B. Moons, M. Verhelst, B. Murmann, "An Always-On 3.8μJ/86% CIFAR-10 Mixed-Signal Binary CNN Processor with All Memory on Chip in 28nm CMOS", ISSCC 2018

[4] D. Shin, et al, "DNPU: An 8.1 TOPS/W Reconfigurable CNN-RNN Processor for General-Purpose Deep Neural Networks", ISSCC 2018.

[5] Andri, R., Cavigelli, L., Rossi, D., & Benini, L. (2017). YodaNN: An Architecture for Ultra-Low Power Binary-Weight CNN Acceleration. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.

[6] Yu-Hsin Chen, Tushar Krishna, Joel Emer, and Vivienne Sze. "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks." ISSCC 2016.

[7] A 288uW Programmable Deep Learning Processor with 270kB On-chip Weight storage using non-uniform memory hierarchy for Mobile Intelligence", ISSCC 2017

[8] N. P. Jouppi, et al. "In-Datacenter Performance Analysis of a Tensor Processing Unit". In ISCA, 2017.

[9] M. Verhelst, "Deep Learning Processor Survey " [Online]. Available: http://www.esat.kuleuven.be/~mverhels/DLICsurvey.html

[10] K. Ueyoshi , K. Andoet al, "QUEST: A 7.49TOPS Multi-Purpose Log-Quantized DNN Inference Engine Stacked on 96MB 3D SRAM Using Inductive-Coupling Technology in 40nm CMOS", ISSCC 2018.

[11] Moons, B., Goetschalckx, K., Van Berckelaer, N., & Verhelst, M. Minimum Energy Quantized Neural Networks. arXiv preprint arXiv:1711.00215, 2017.

[12] M. Verhelst, B. Moons, "Embedded Deep Neural Network Processing: Algorithmic and processor techniques bring deep learning to IoT and edge devices", SSCS Magazine, Fall Magazine 2017.

[13] Yang, Tien-Ju and Chen, Yu-Hsin and Sze, Vivienne, "Designing Energy-Efficient Convolutional Neural Networks Using Energy-Aware Pruning", The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.