Deep Learning Hardware: Past, Present, and Future

Orange County ACM

March 16, 2022

Bill Dally

Chief Scientist and SVP of Research, NVIDIA Corporation

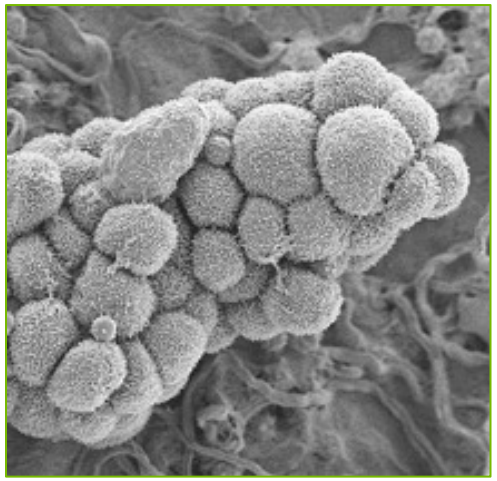
Adjunct Professor, Stanford University

DL Applications

Deep Learning Everywhere

INTERNET & CLOUD

Image Classification
Speech Recognition
Language Translation
Language Processing
Sentiment Analysis
Recommendation



MEDICINE & BIOLOGY

Cancer Cell Detection
Diabetic Grading
Drug Discovery

MEDIA & ENTERTAINMENT

Video Captioning
Video Search
Real Time Translation

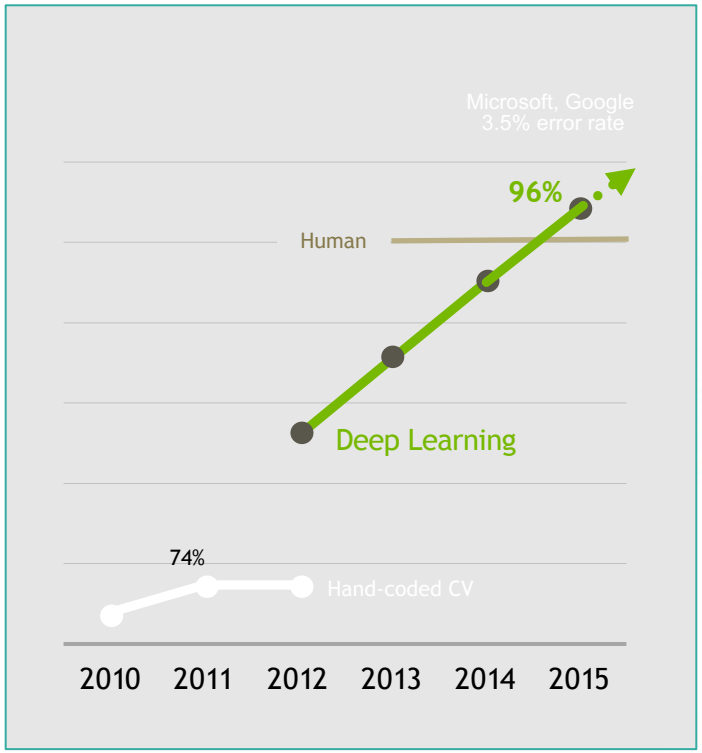
SECURITY & DEFENSE

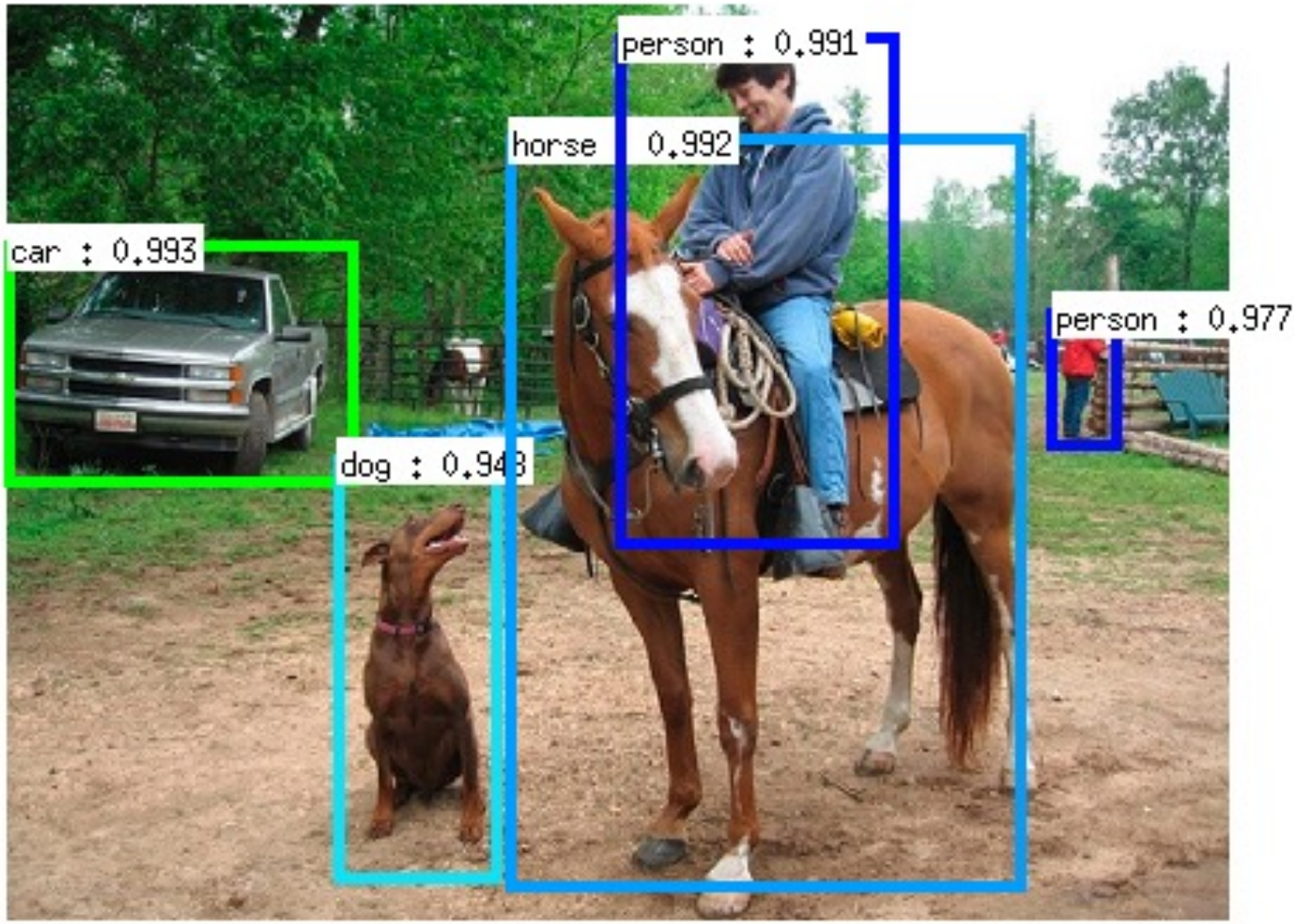
Face Detection
Video Surveillance
Satellite Imagery

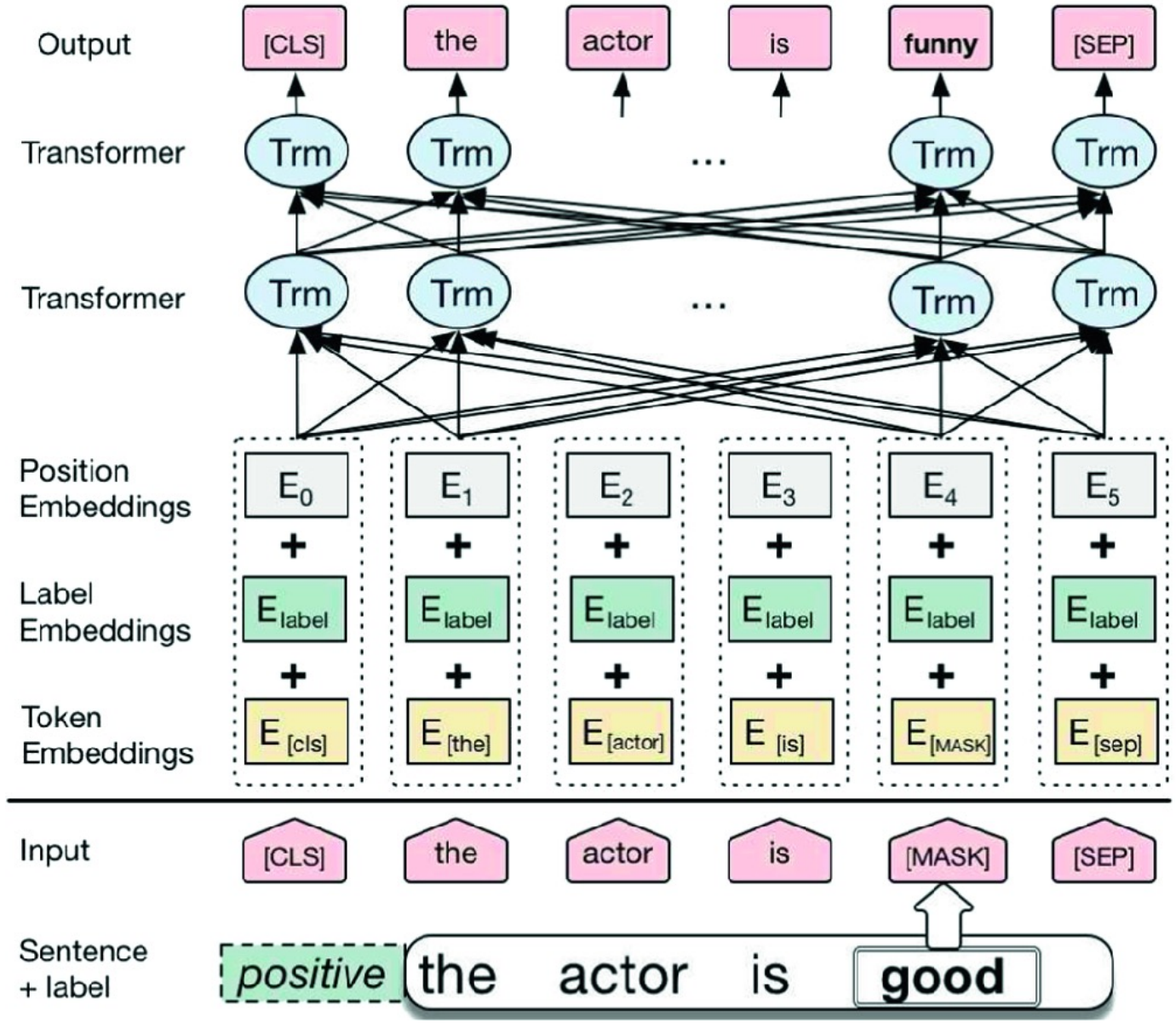
AUTONOMOUS MACHINES

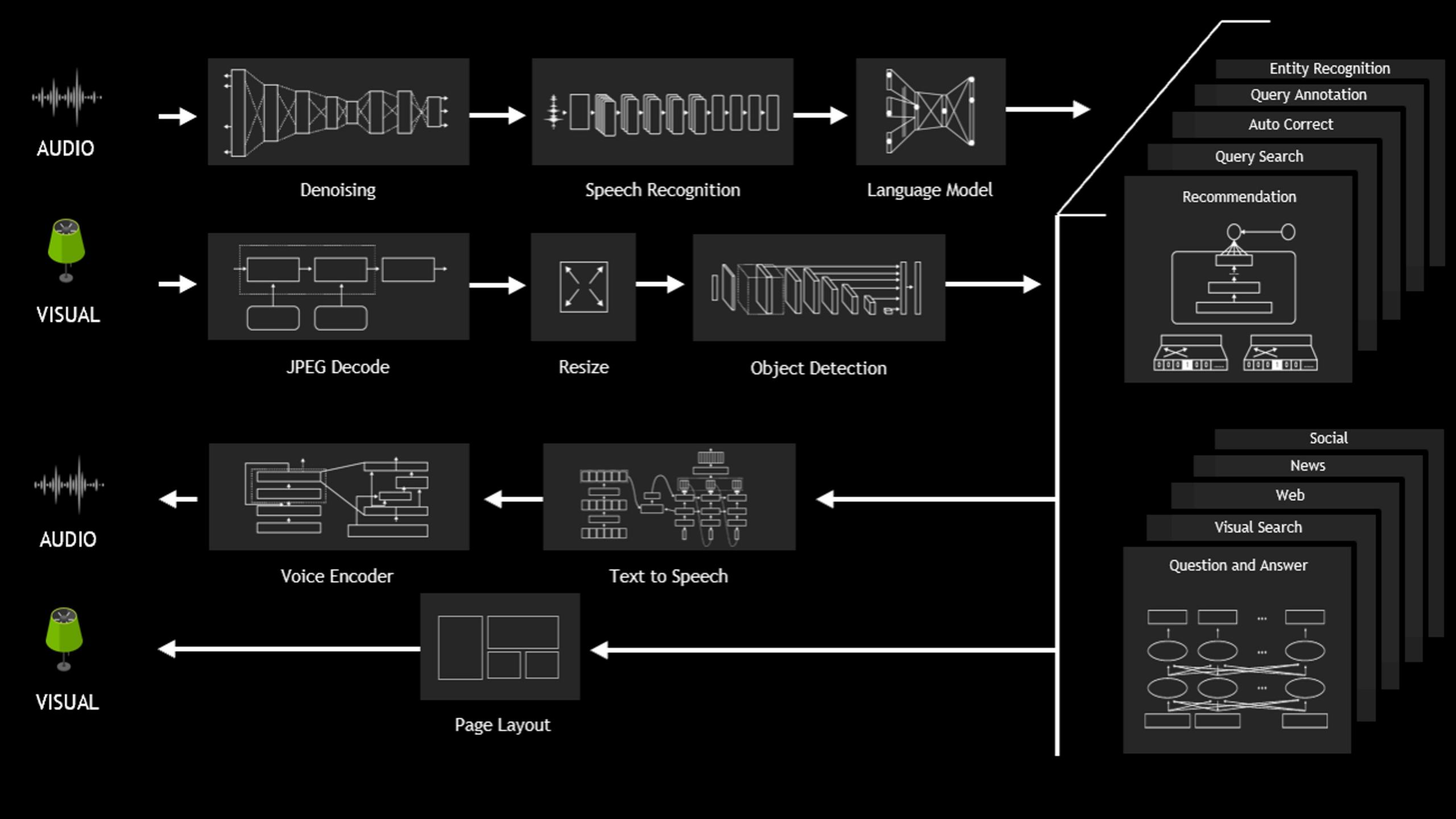
Pedestrian Detection
Lane Tracking
Recognize Traffic Sign

Computer Vision






Output
[CLS]
the
actor
is
funny
[SEP]
Transformer
Trm
Trm
...
Trm
Trm
Transformer
Trm
Trm
...
Trm
Trm
Position Embeddings
E0
E1
E2
E3
E4
E5
Label Embeddings
Elabel
Token Embeddings
E [cls]
E [the]
E [actor]
E [is]
E [MASK]
E [sep]
Input
[CLS]
the
actor
is
[MASK]
[SEP]
Sentence + label
positive
the actor is good


AUDIO
Denoising
Speech Recognition
Language Model
VISUAL
JPEG Decode
Resize
Object Detection
Entity Recognition
Query Annotation
Auto Correct
Query Search
Recommendation
AUDIO
Voice Encoder
Text to Speech
Social
News
Web
Visual Search
Question and Answer
VISUAL
Page Layout

RECOMMENDERS — THE ENGINE OF THE INTERNET

Billions of Users - Trillions of Items

100’s of millions of etail items -> Amazon & Alibaba recommenders

1000’s of movies -> Netflix recommender

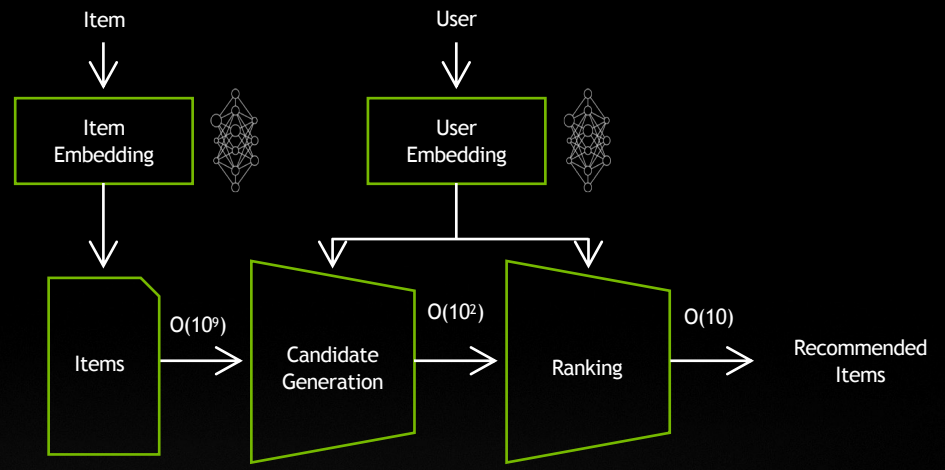
10’s of millions of songs -> Spotify & iTunes recommenders

10’s of millions of books -> Amazon recommender

Billions of Tik Tok & YT videos -> TT & YT recommender

Billions of websites -> Google search rank

So much news!!! -> Google & FB news recommenders

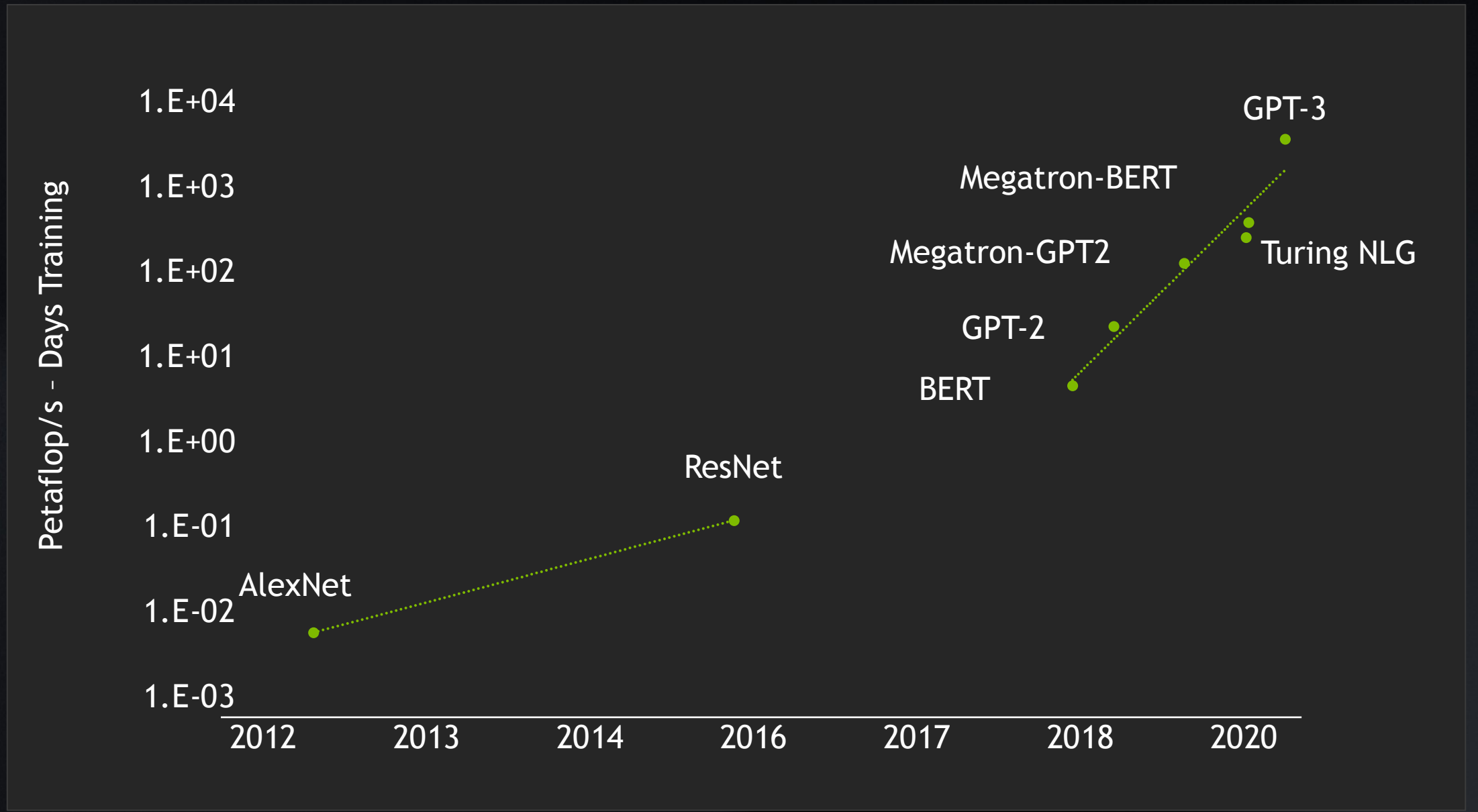


Data → Insight → Value

The AI Revolution has been Enabled by Hardware

Deep Learning was Enabled by Hardware

Deep Learning is Gated by Hardware



Facebook's latest giant language AI hits computing wall at 500 Nvidia GPUs

Facebook AI research's latest breakthrough in natural language understanding, called XLM-R, performs cross-language tasks with 100 different languages including Swahili and Urdu, but it's also running up against the limits of existing computing power.

By Tiernan Ray | November 12, 2019 -- 19:28 GMT (03:28 GMT+08:00) | Topic: Artificial Intelligence

https://www.zdnet.com/article/facebooks-latest-giant-language-ai-hits-computing-wall-at-500-nvidia-gpus/

Nvidia Is Chosen to Power the Fastest Supercomputer in the World -- Facebook's AI Research Data Center

By Nicholas Rossolillo - Jan 29, 2022 at 8:20AM

760 DGX A100 boxes
6,080 A100 GPUs

ARTIFICIAL INTELLIGENCE

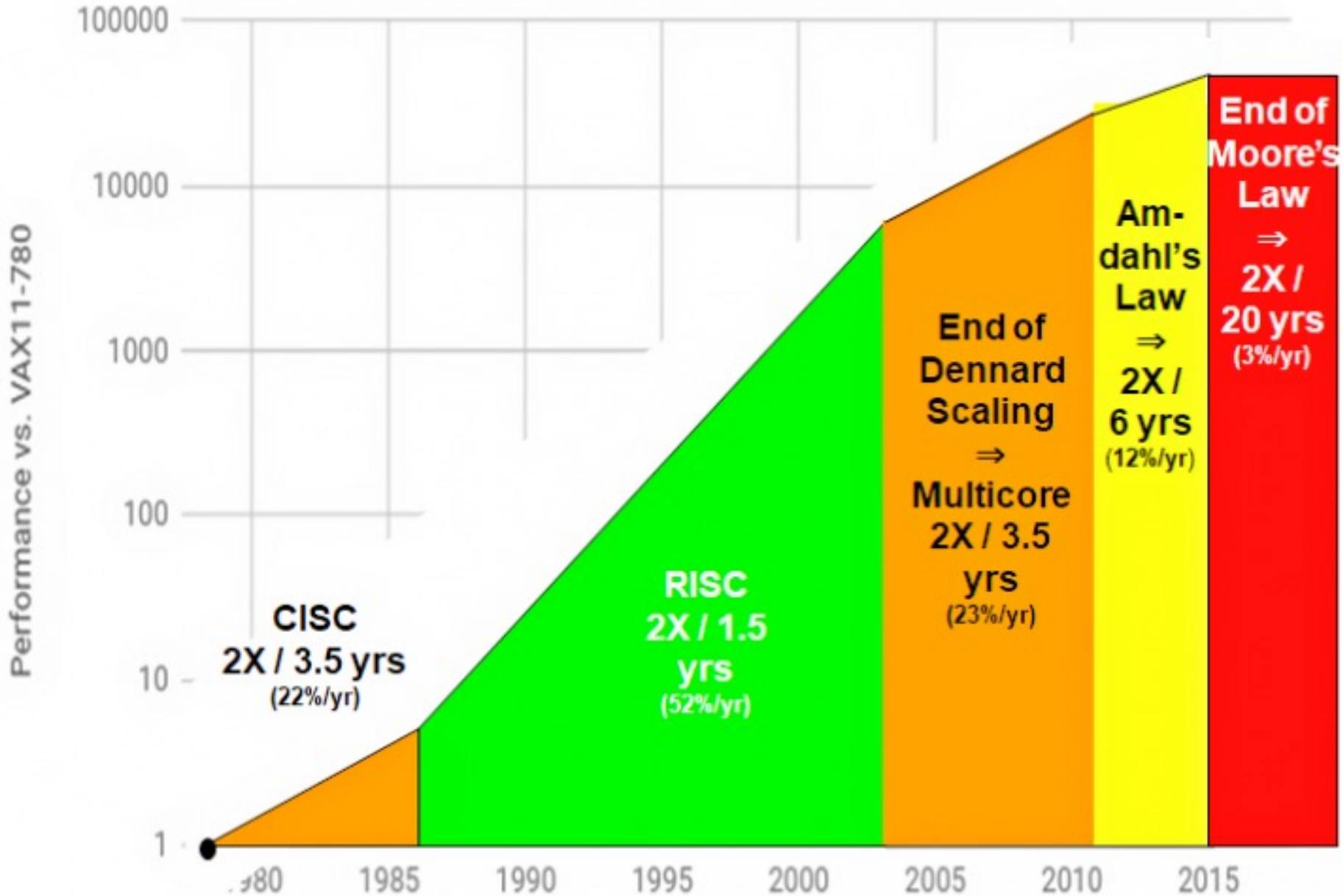
Training a single AI model can emit as much carbon as five cars in their lifetimes

Deep learning has a terrible carbon footprint.

By Karen Hao

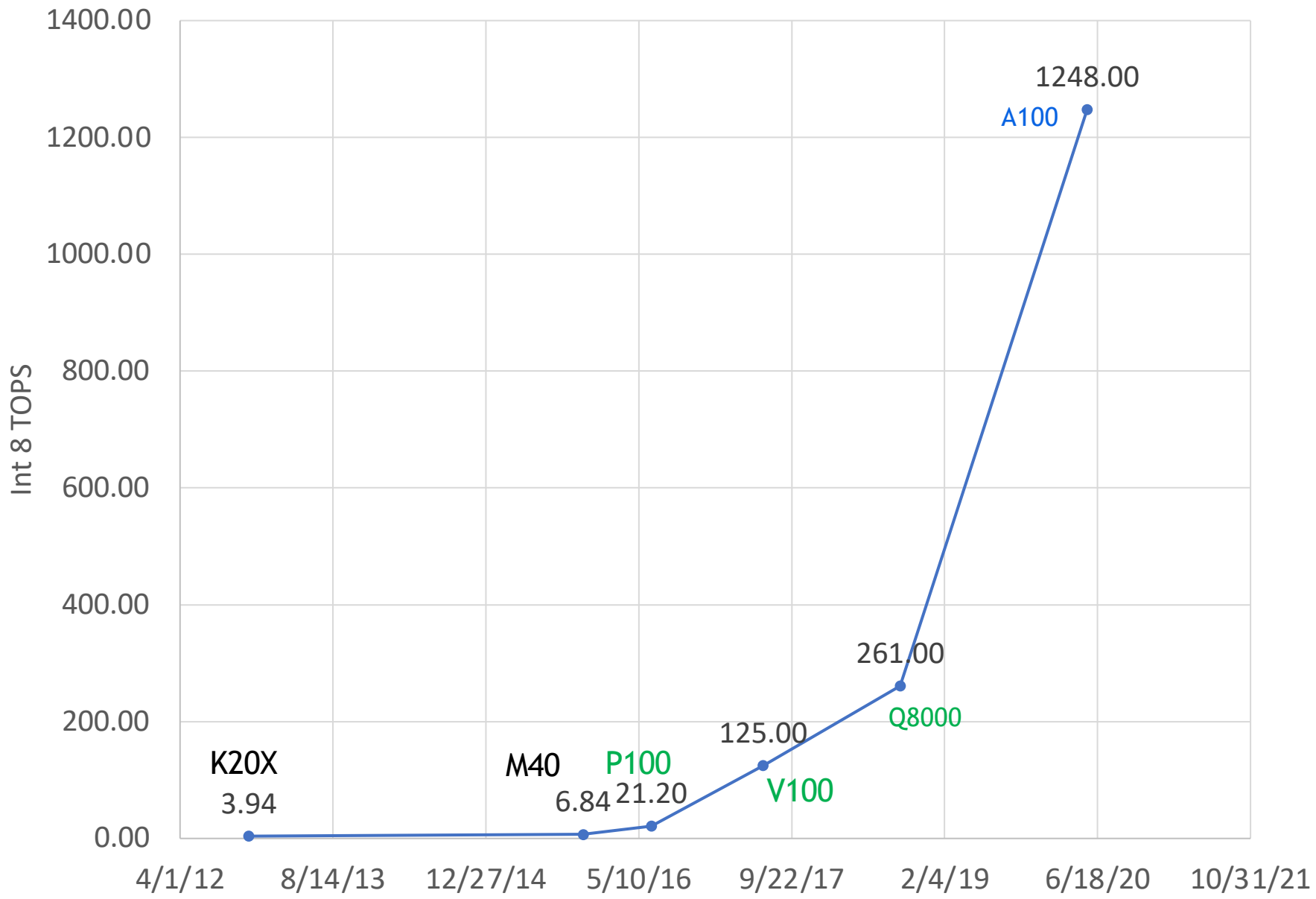
June 6, 2019

But Moore's Law is Dead



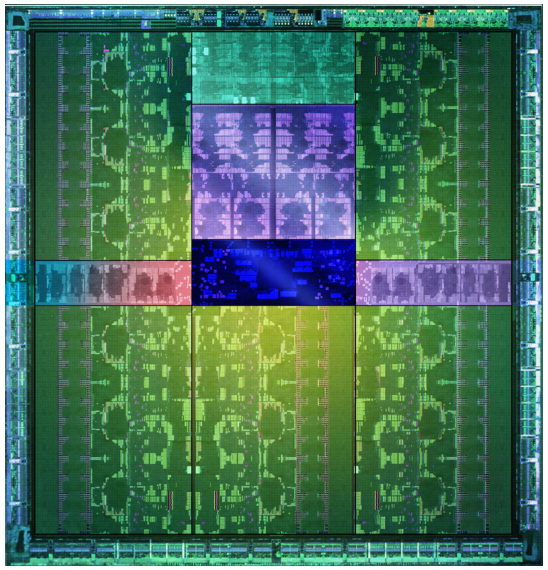
John Hennessy and David Patterson, Computer Architecture: A Quantitative Approach, 6/e. 2018

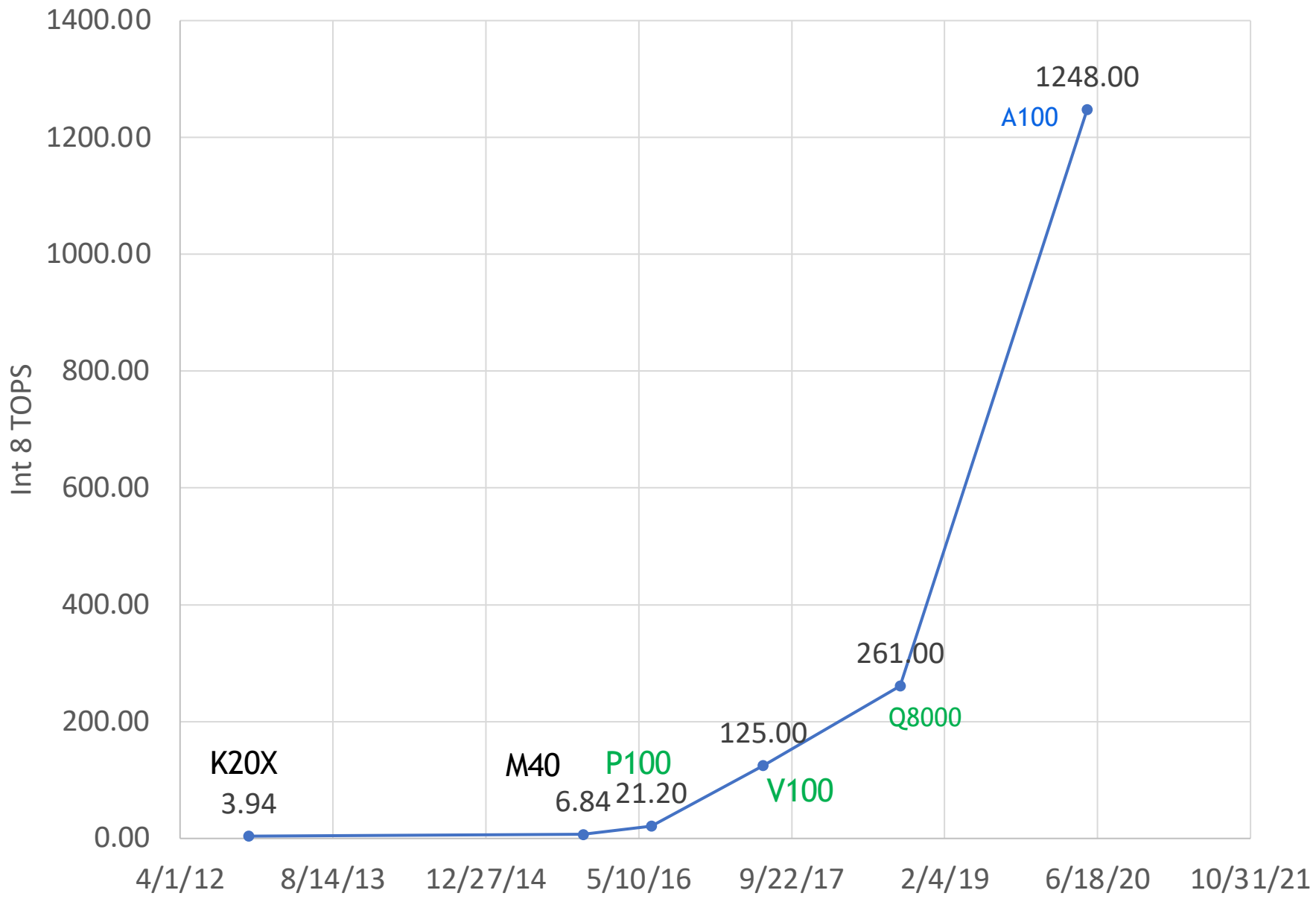
Some History


Single-Chip Inference Performance - 317X in 8 years
Int 8 TOPS
1400.00
1200.00
1000.00
800.00
600.00
400.00
200.00
0.00
4/1/12
8/14/13
12/27/14
5/10/16
9/22/17
2/4/19
6/18/20
10/31/21
K20X
3.94
M40
6.84
P100
21.20
V100
125.00
Q8000
261.00
A100
1248.00

Kepler (2012)

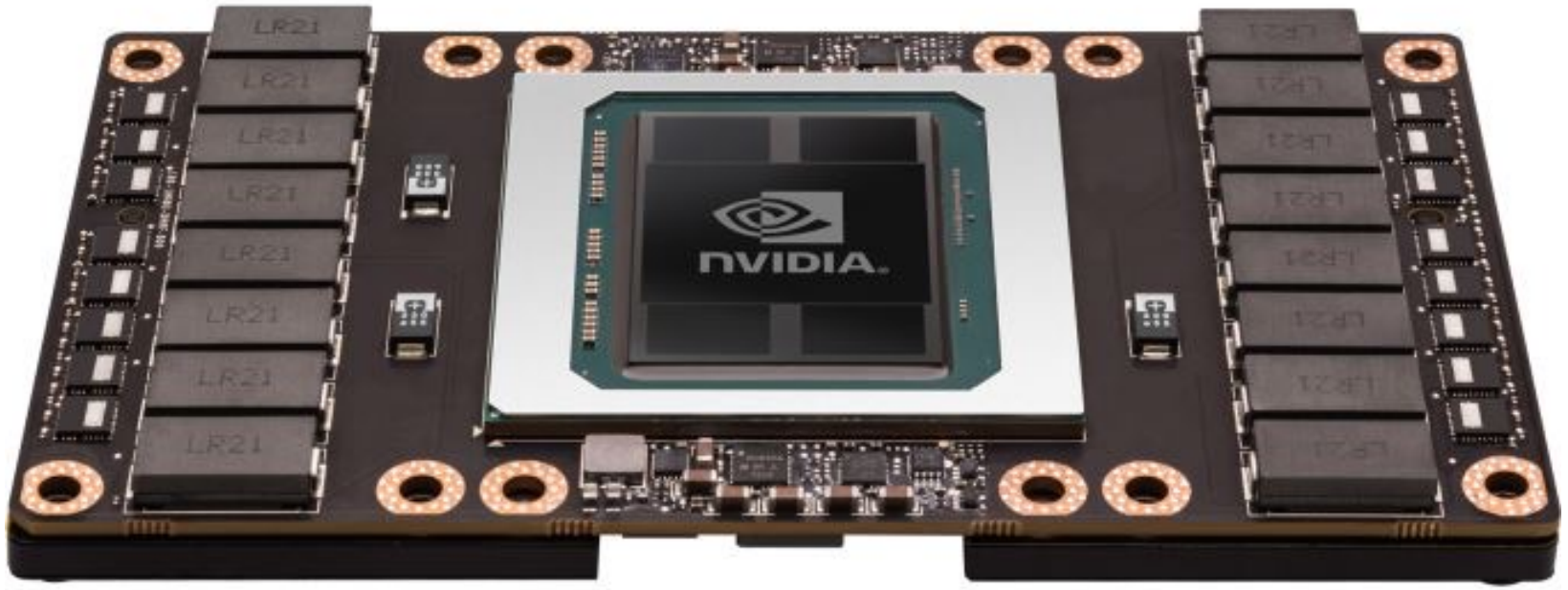
3.95 TFLOPS (FP32)
250 GB/s
300W
28nm

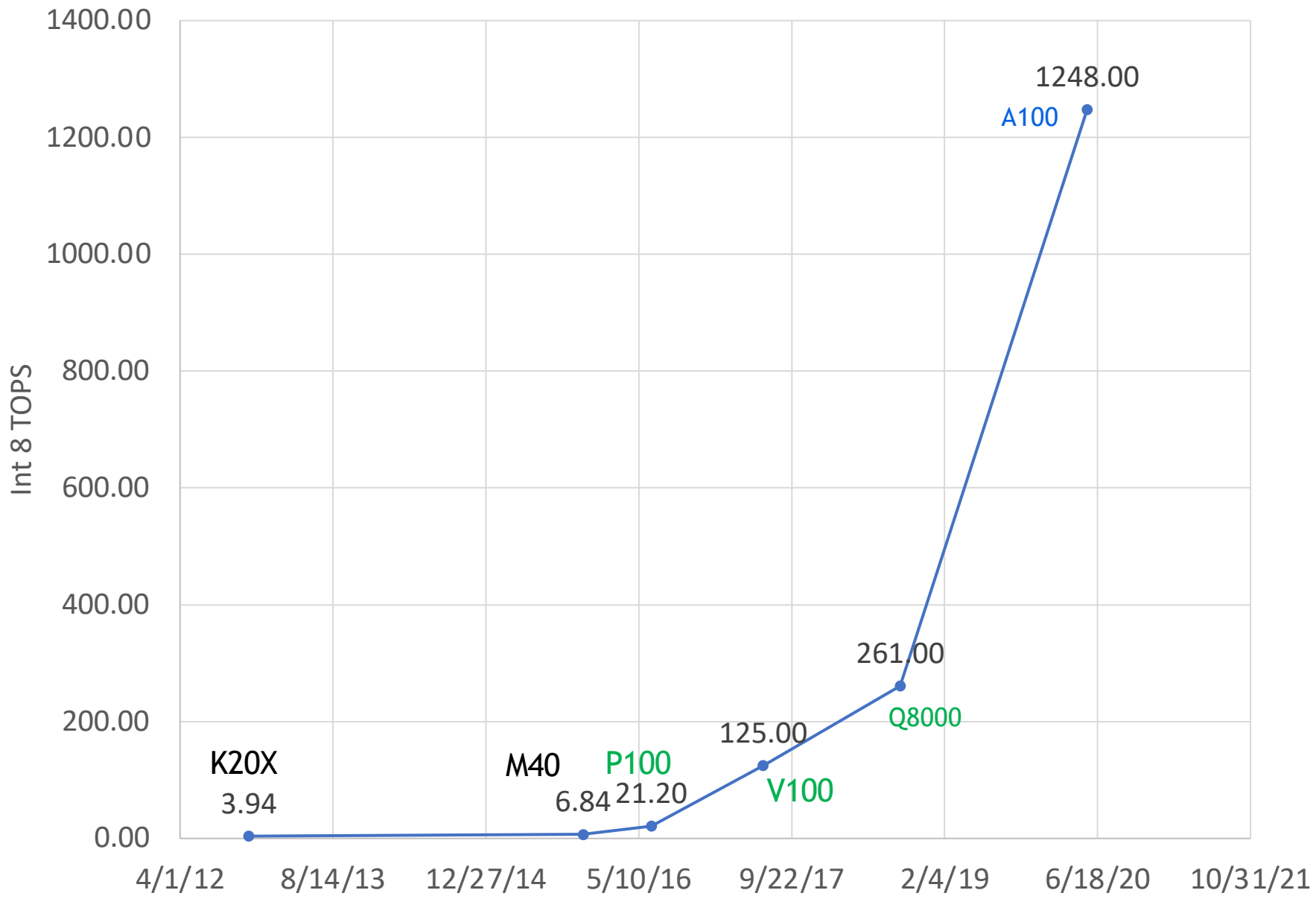



Single-Chip Inference Performance - 317X in 8 years
Int 8 TOPS
1400.00
1200.00
1000.00
800.00
600.00
400.00
200.00
0.00
4/1/12
8/14/13
12/27/14
5/10/16
9/22/17
2/4/19
6/18/20
10/31/21
K20X
3.94
M40
6.84
P100
21.20
V100
125.00
Q8000
261.00
A100
1248.00

Pascal (2016)

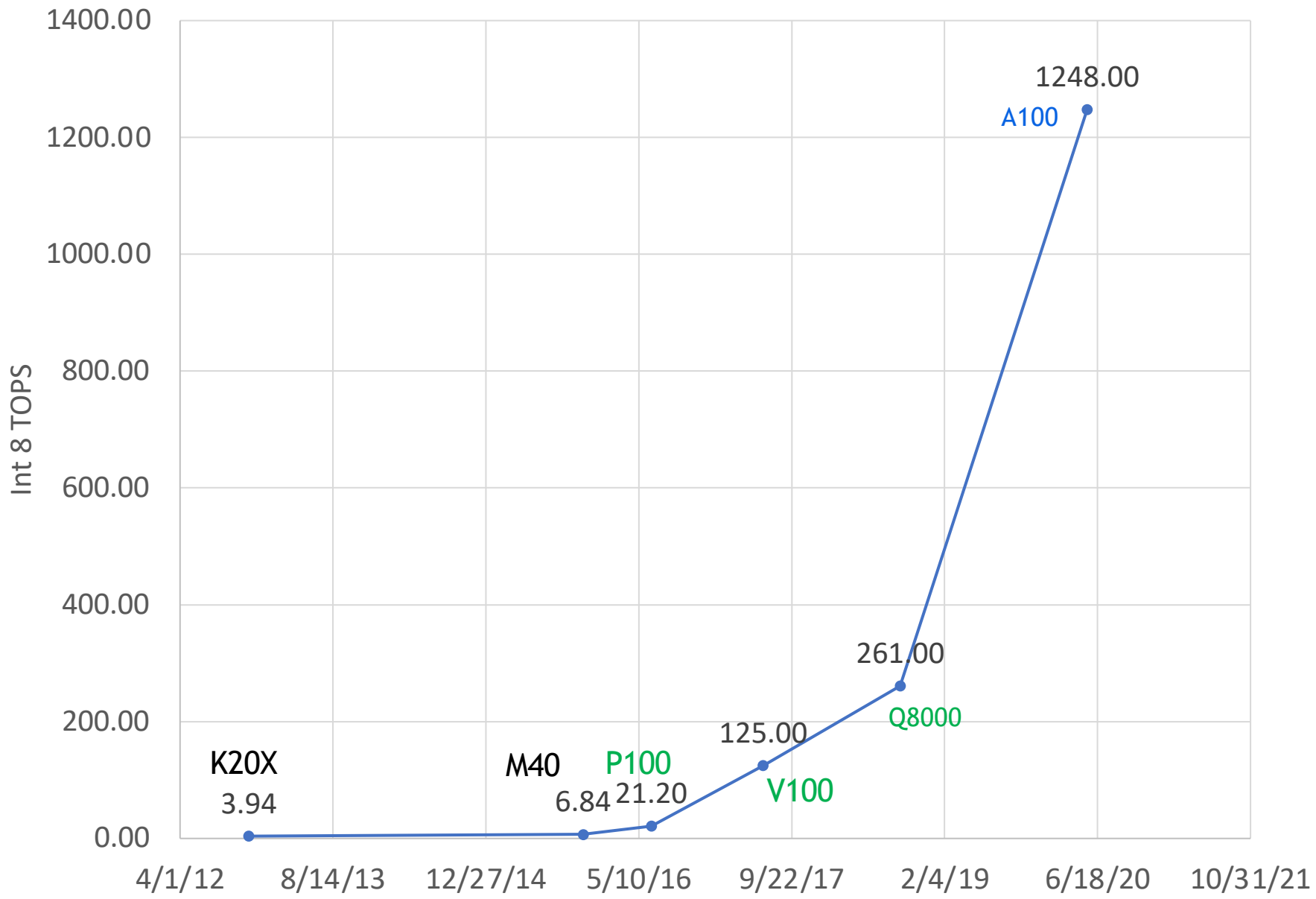
10.6 TFLOPS (FP32)
21.3 TFLOPS (FP16)
FDP4
732 GB/s (HBM)
NVLink
300W




Single-Chip Inference Performance - 317X in 8 years
Int 8 TOPS
1400.00
1200.00
1000.00
800.00
600.00
400.00
200.00
0.00
4/1/12
8/14/13
12/27/14
5/10/16
9/22/17
2/4/19
6/18/20
10/31/21
K20X
3.94
M40
6.84
P100
21.20
V100
125.00
Q8000
261.00
A100
1248.00

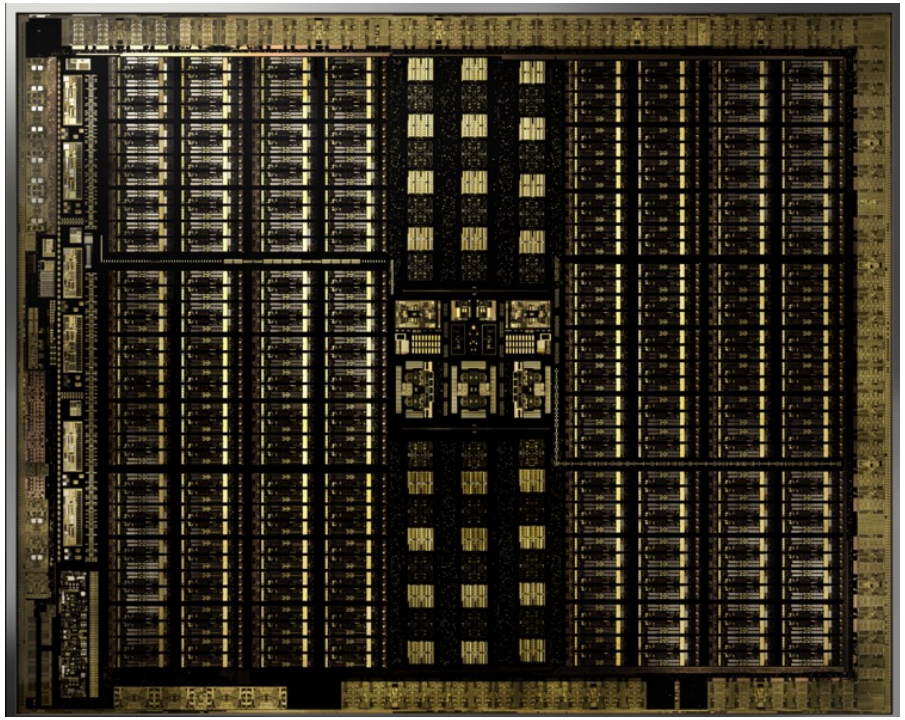
Volta (2017)

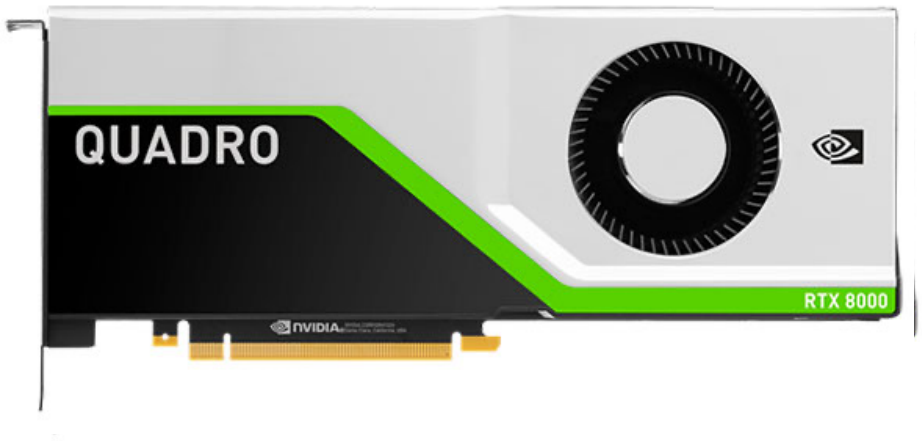
Tensor Cores!
15 TFLOPS (FP32)
125 TFLOPS (FP16)
HMMA
900 GB/s (HBM)
300 GB/s NVLink
300W

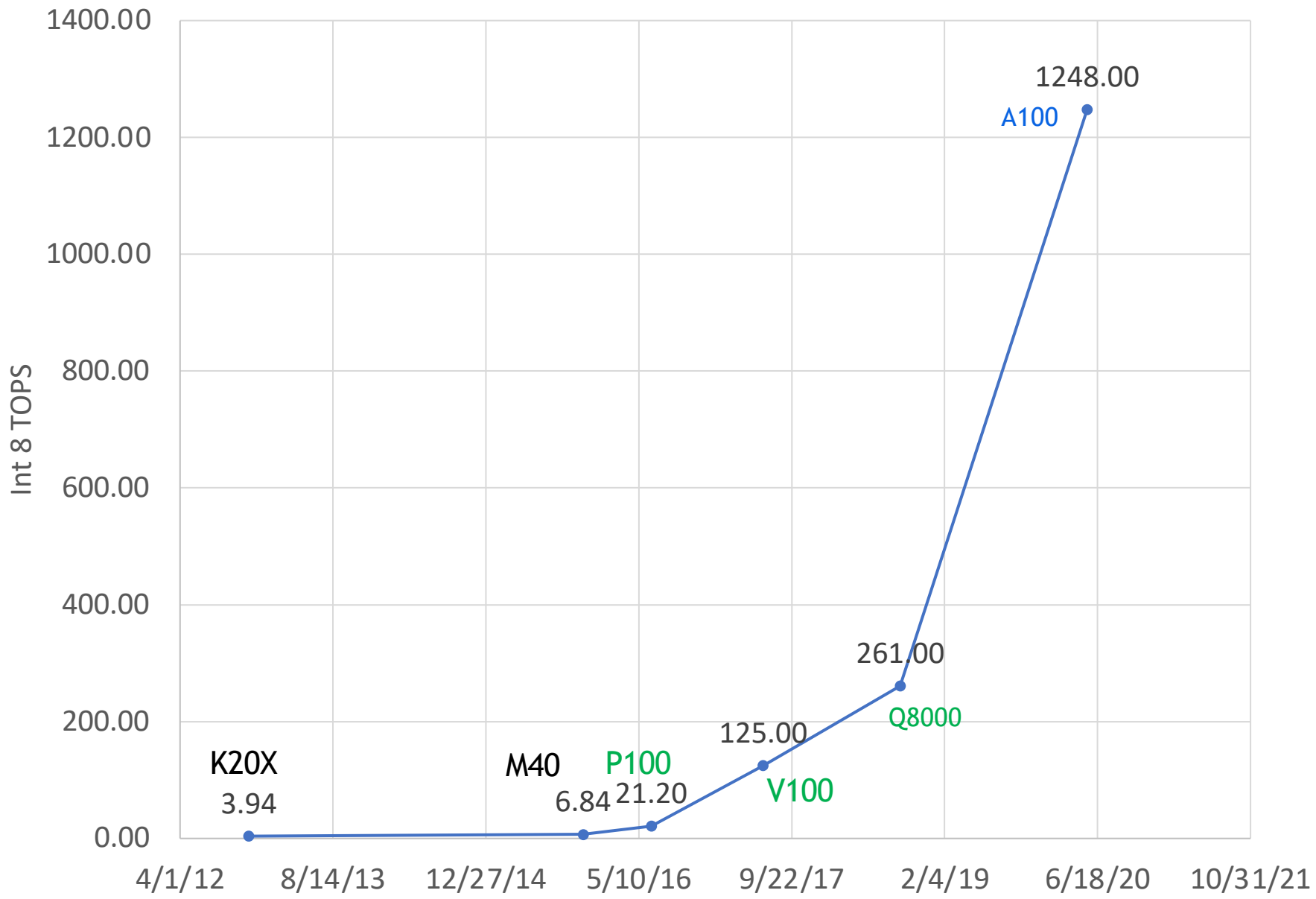

Single-Chip Inference Performance - 317X in 8 years
Int 8 TOPS
1400.00
1200.00
1000.00
800.00
600.00
400.00
200.00
0.00
4/1/12
8/14/13
12/27/14
5/10/16
9/22/17
2/4/19
6/18/20
10/31/21
K20X
3.94
M40
6.84
P100
21.20
V100
125.00
Q8000
261.00
A100
1248.00

Turing (2018)

Integer Tensor Cores!
65 TFLOPS (FP32)
130 TFLOPS (FP16)
261 TOPs (Int8)
IMMA
672 GB/s (G5)
Ray Tracing!



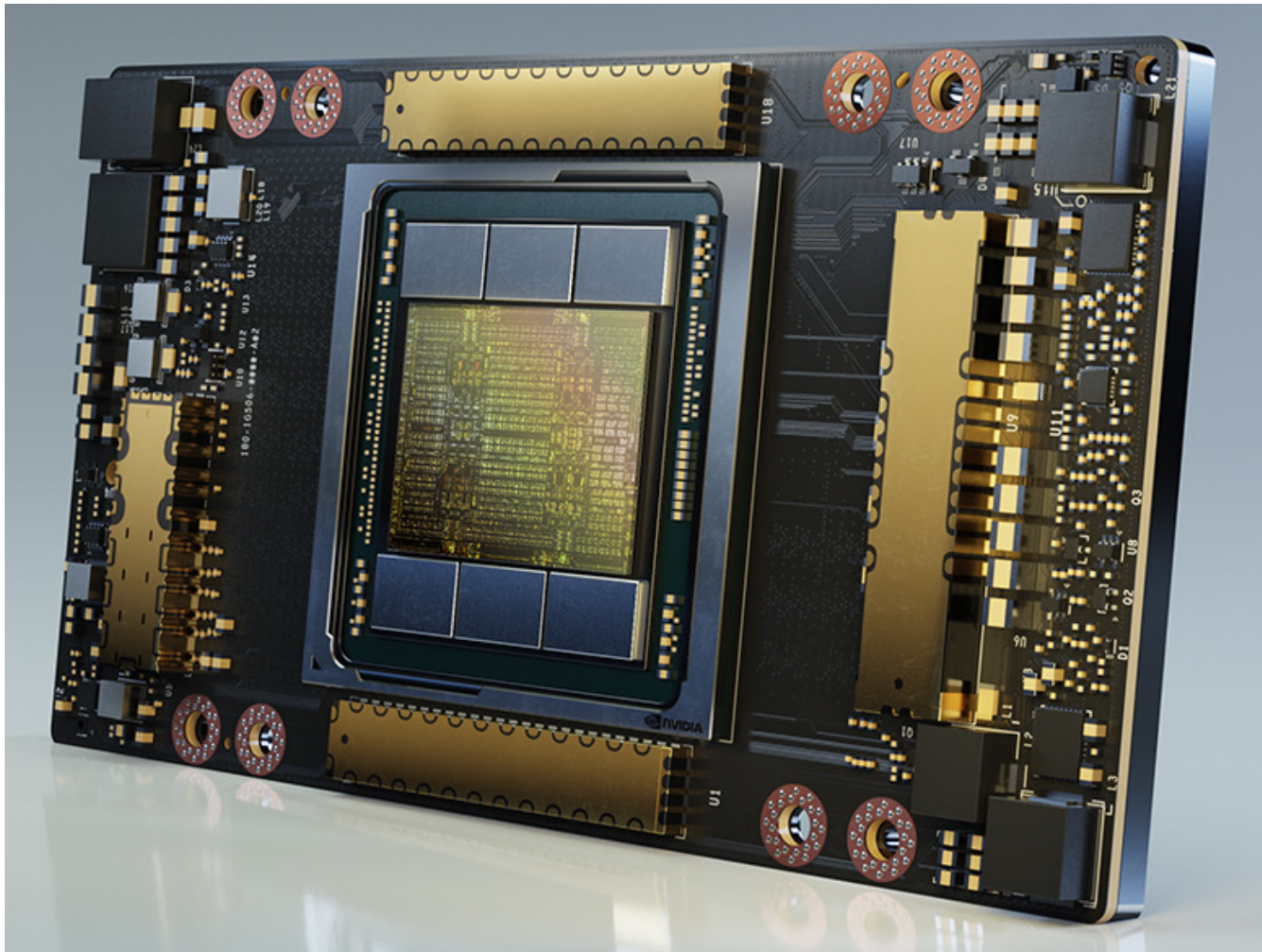



Single-Chip Inference Performance - 317X in 8 years
Int 8 TOPS
1400.00
1200.00
1000.00
800.00
600.00
400.00
200.00
0.00
4/1/12
8/14/13
12/27/14
5/10/16
9/22/17
2/4/19
6/18/20
10/31/21
K20X
3.94
M40
6.84
P100
21.20
V100
125.00
Q8000
261.00
A100
1248.00

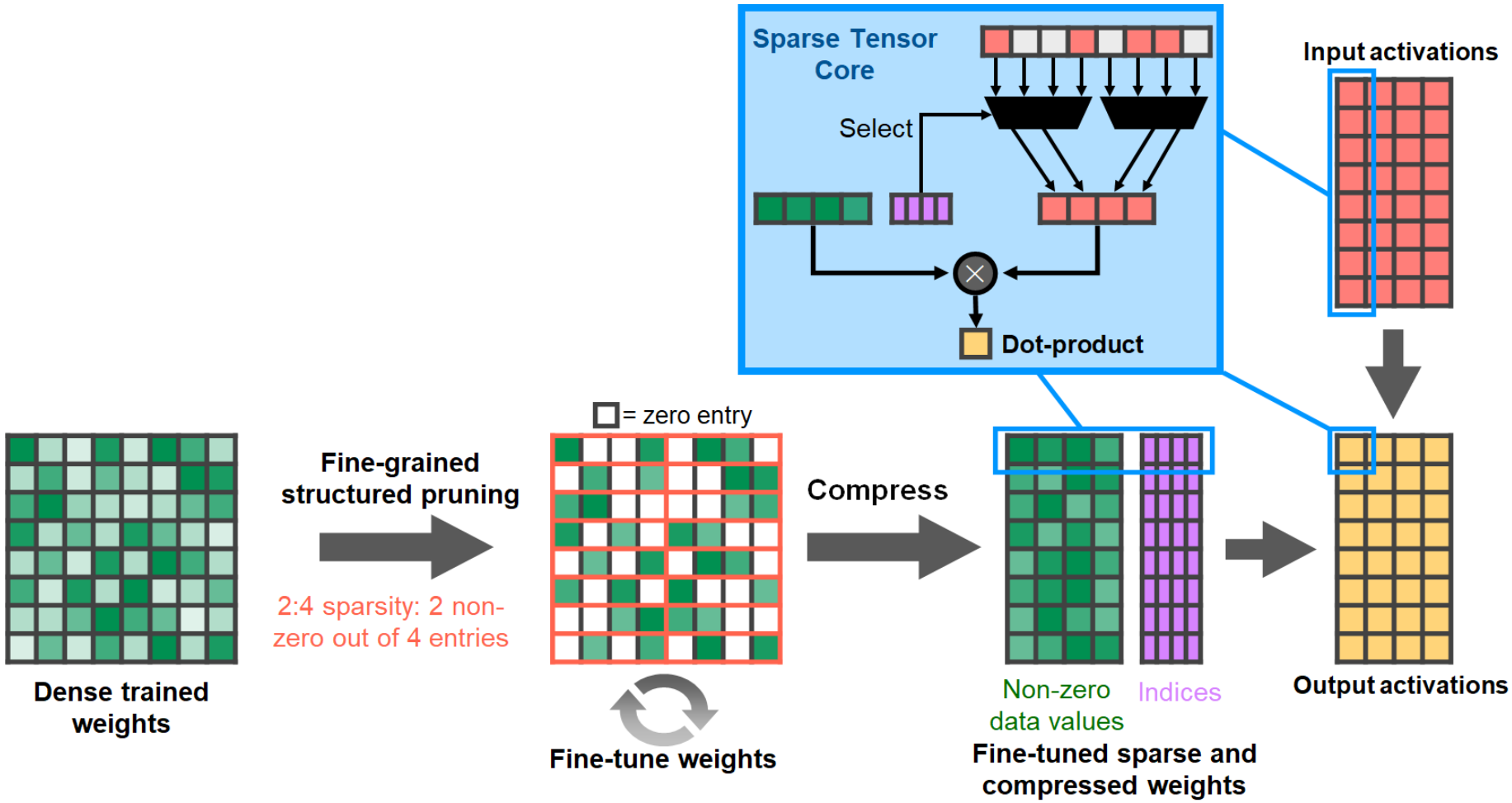
Ampere (2020)

Sparsity!
BF16 & TF32!
156 / 312 TFLOPS (TF32) (dense/sparse)
312 / 624 TFLOPS (FP16 or BF16)
624 / 1,248 TOPS (Int 8)
1,248 / 2,496 TOPS (Int 4)
2TB/s (HBM)
400W

3.12 TOPS/W (Int 8)
6.24 TOPS/W (Int 4)



Structured Sparsity



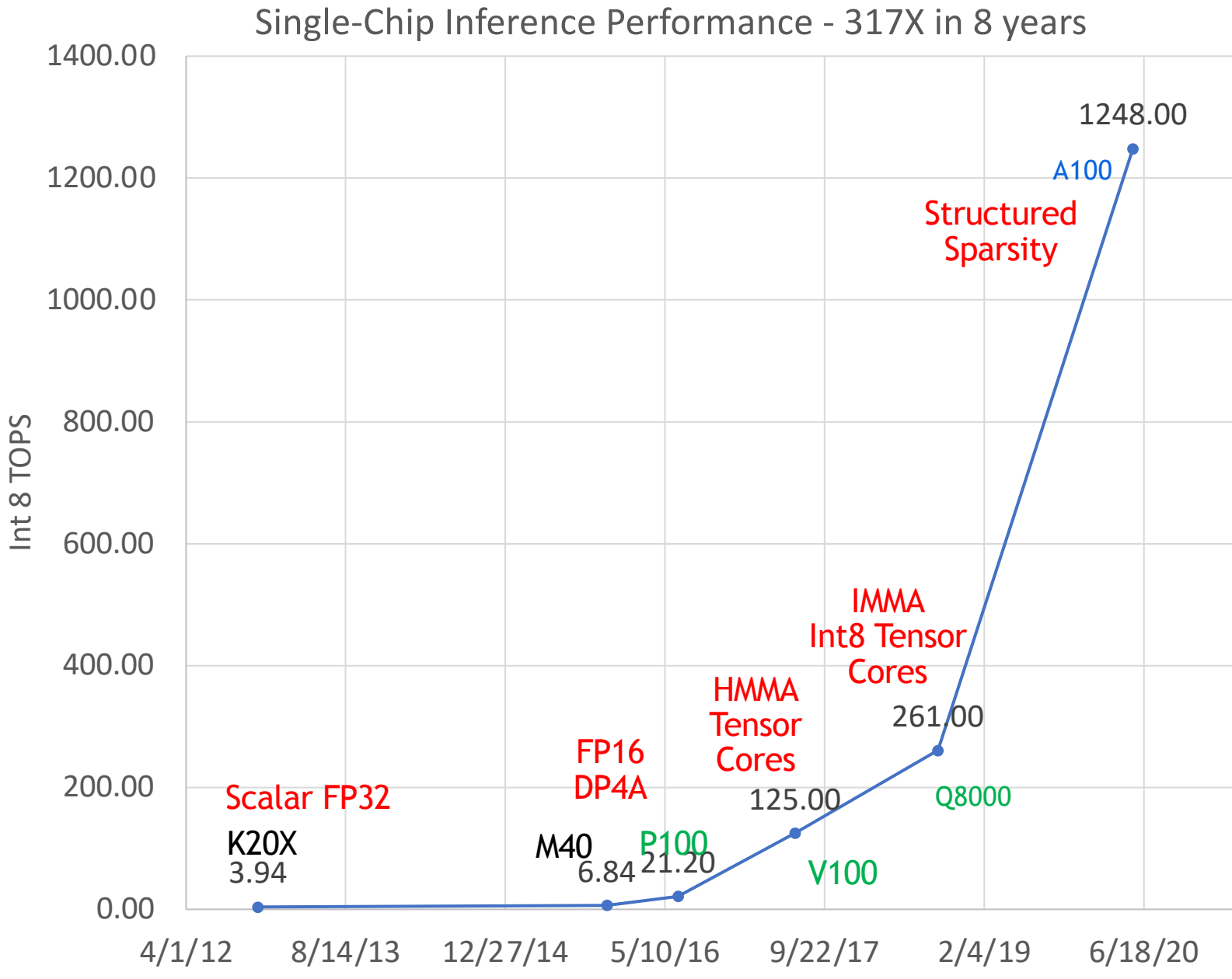
NVIDIA A100 Tensor Core GPU Architecture whitepaper

Gains from

Number representation
FP32, FP16, Int8
(TF32, BF16)

Complex instructions
DP4, HMMA, IMMA

Process
28nm, 16nm, 7nm



Specialized Instructions Amortize Overhead

Operation	Energy**	Overhead*
HFMA	1.5pJ	2000%
HDP4A	6.0pJ	500%
HMMA	110pJ	22%
IMMA	160pJ	16%

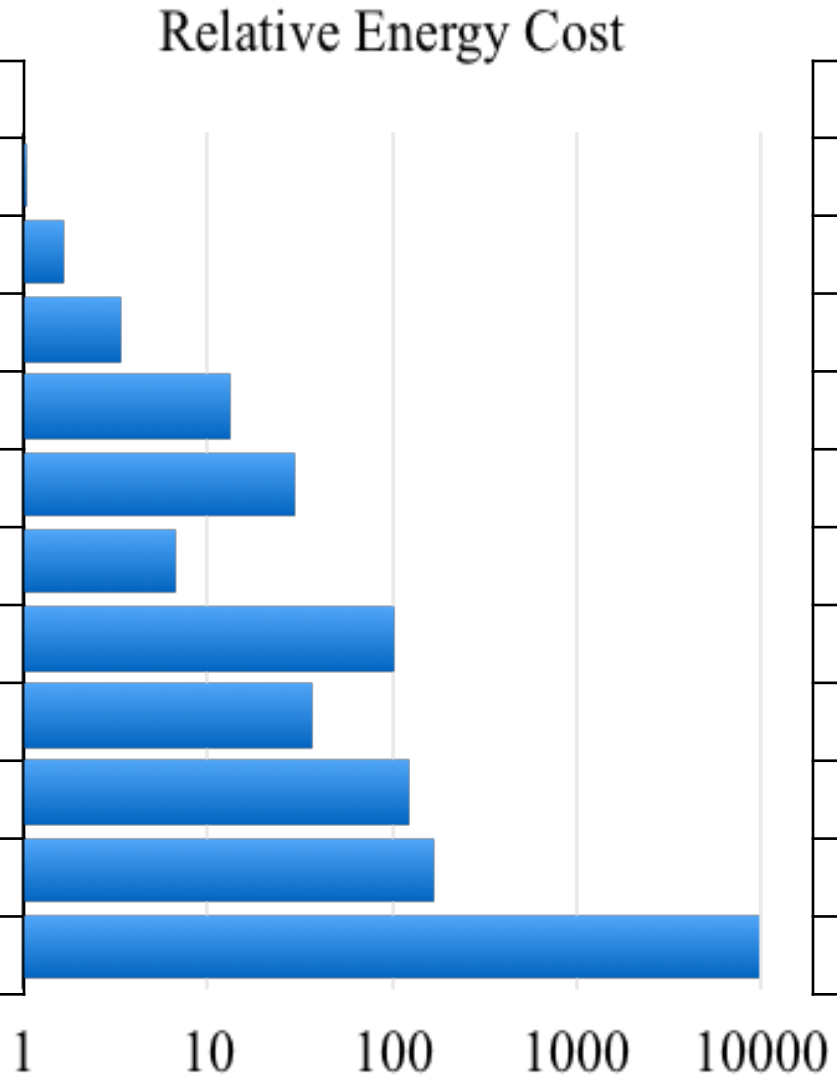
*Overhead is instruction fetch, decode, and operand fetch – 30pJ
**Energy numbers from 45nm process

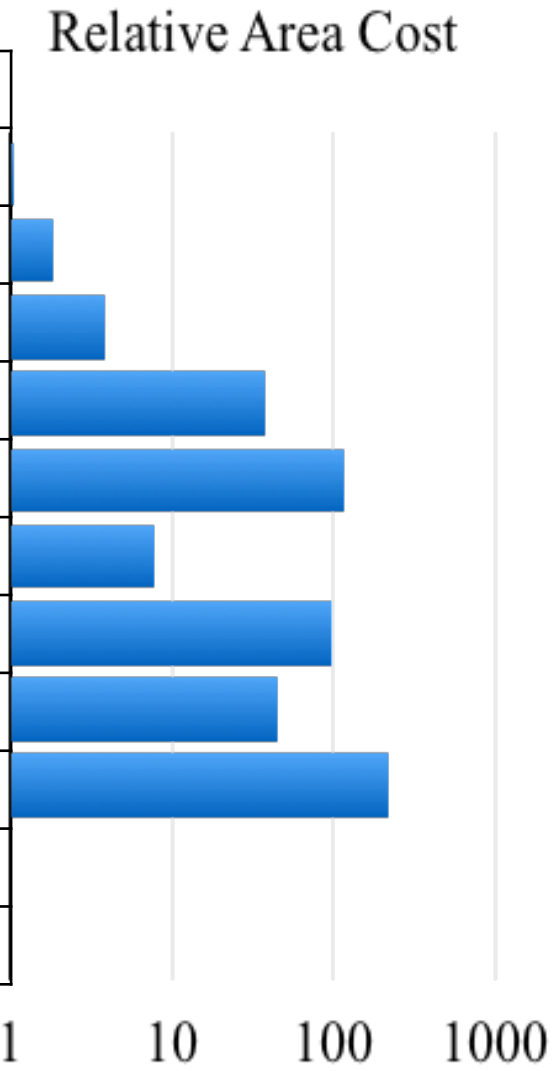
Accelerators

All have a matrix-multiply unit fed by a memory hierarchy.

Cost of Operations

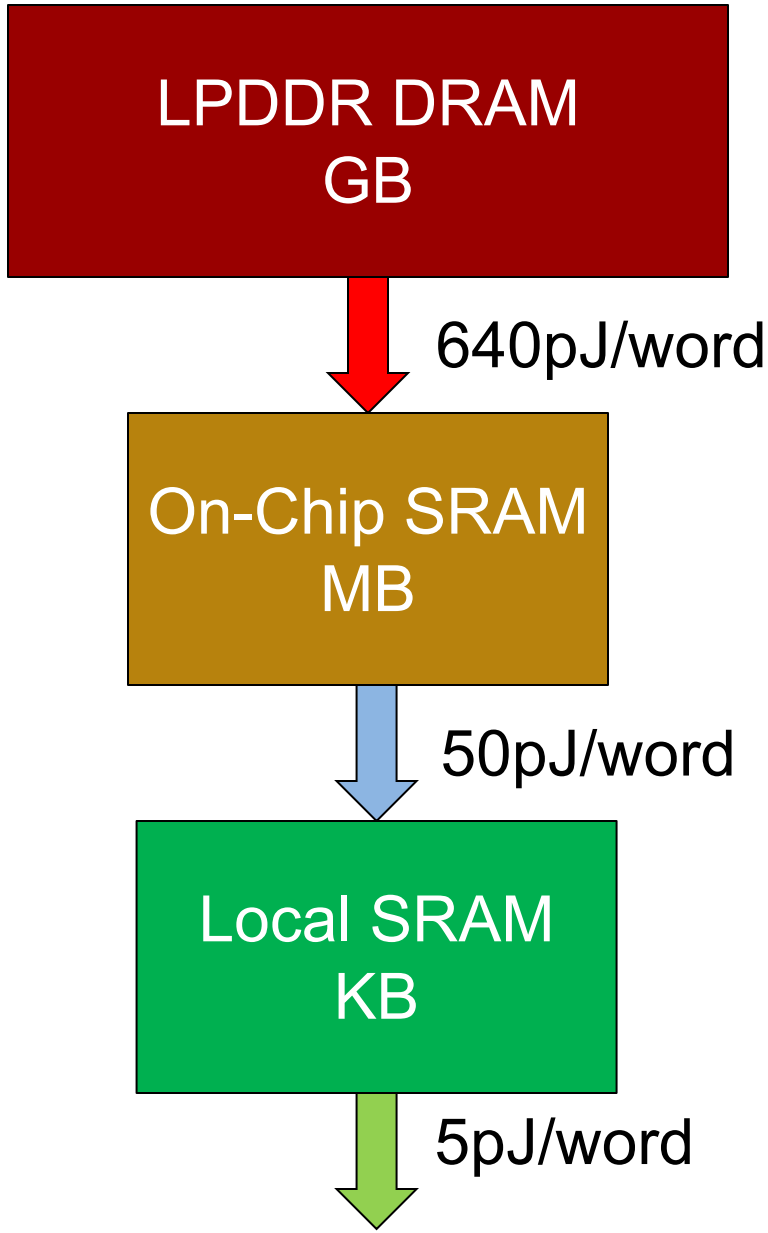
Operation:	Energy (pJ)	Area (μm²)
8b Add	0.03	36
16b Add	0.05	67
32b Add	0.1	137
16b FP Add	0.4	1360
32b FP Add	0.9	4184
8b Mult	0.2	282
32b Mult	3.1	3495
16b FP Mult	1.1	1640
32b FP Mult	3.7	7700
32b SRAM Read (8KB)	5	N/A
32b DRAM Read	640	N/A



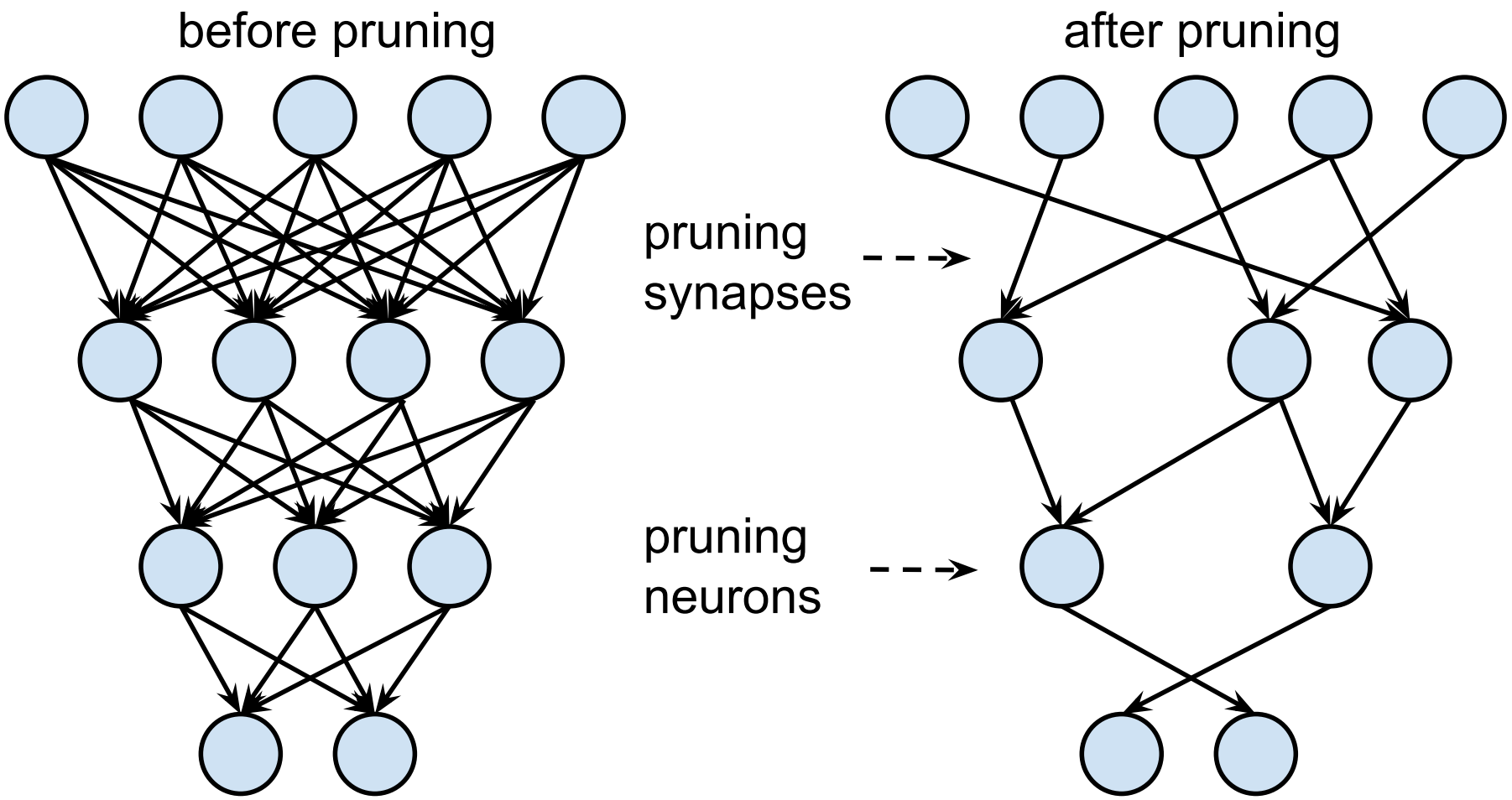


Energy numbers are from Mark Horowitz “Computing’s Energy Problem (and what we can do about it)”, ISSCC 2014
Area numbers are from synthesized result using Design Compiler under TSMC 45nm tech node. FP units used DesignWare Library.

The Importance of Staying Local



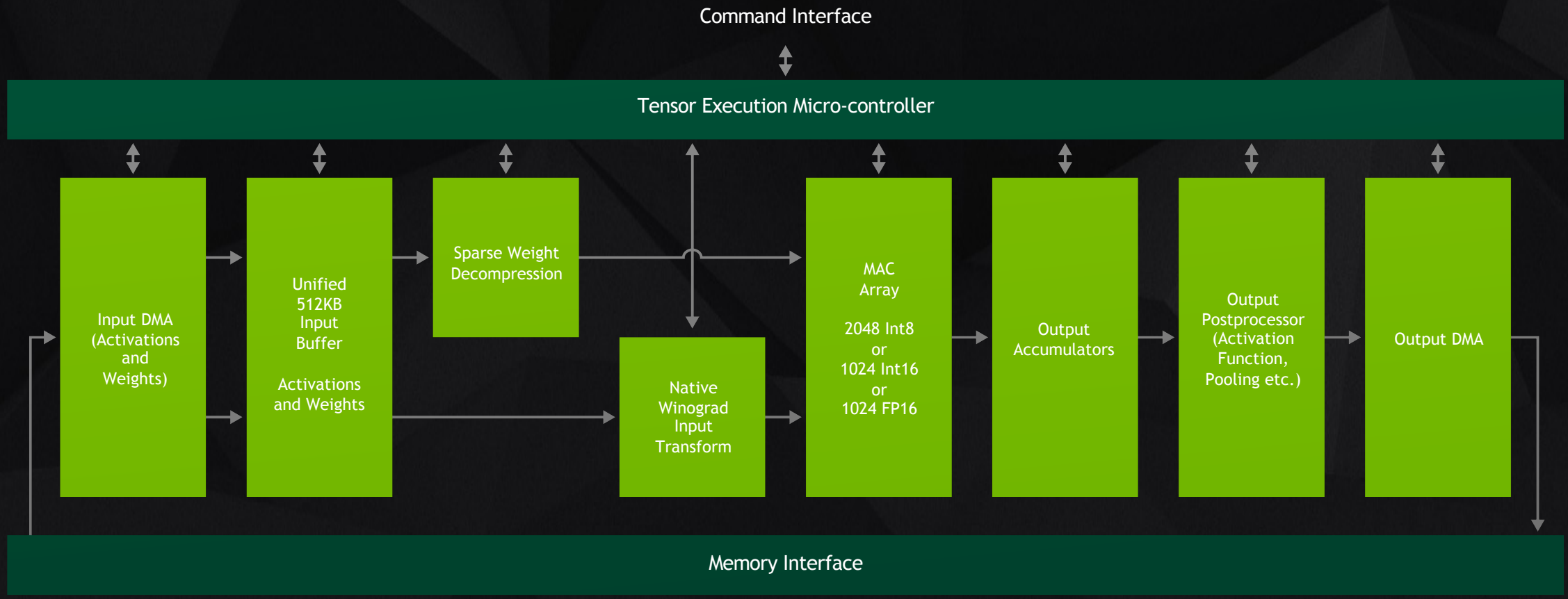
Pruning



Han et al. Learning both Weights and Connections for Efficient Neural Networks, NIPS 2015

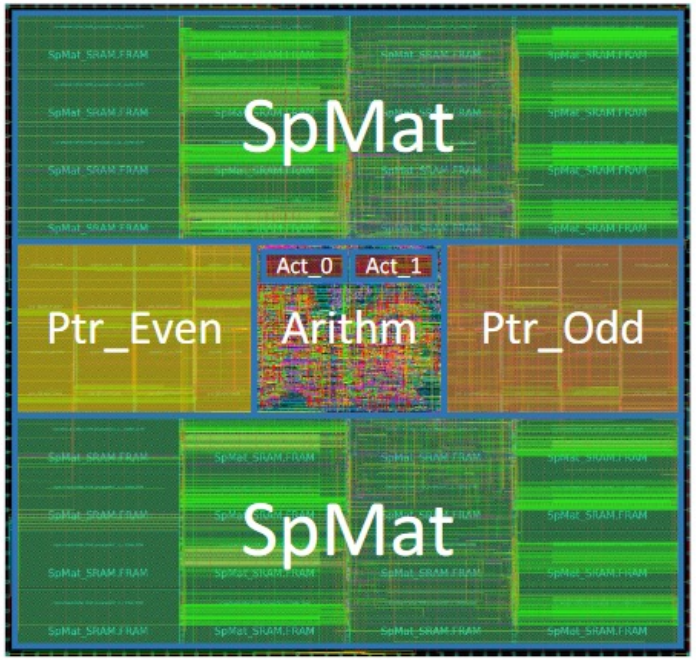
NVIDIA DLA

Sparsity
Compression
Data gating
Winograd



Open-sourced at nvdla.org

EIE (2016)

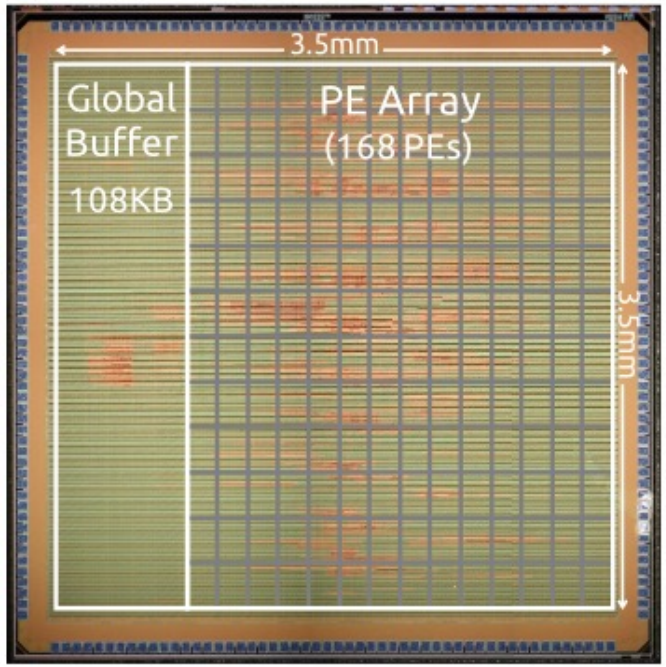


Efficient Inference Engine
for compressed
fully connected layers

Sparsity
- Hardware CSR

Coding
- Scalar Quantization

Eyeriss (2016)

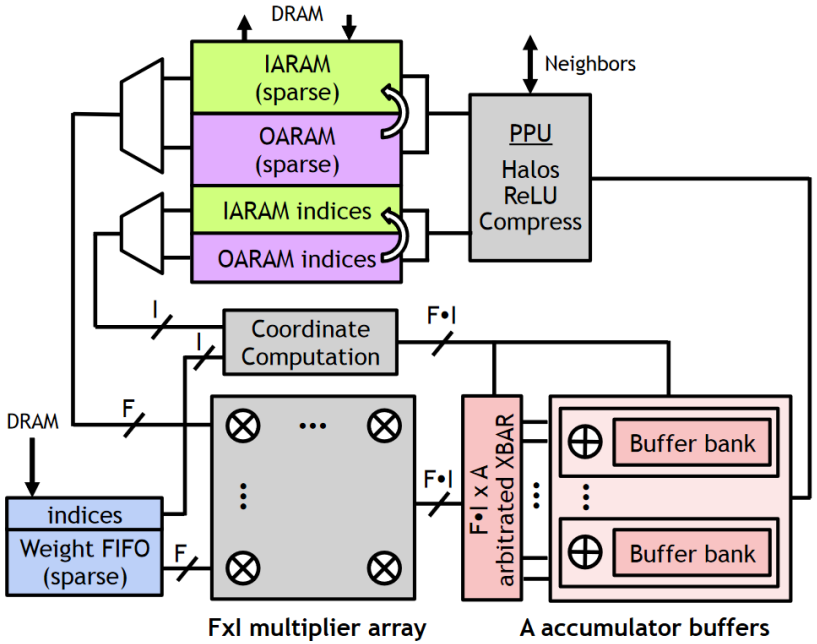


Spatial tiling with optimized dataflows for CNNs

Tiling (dataflows)

- Weight stationary
- Row stationary

SCNN (2017)

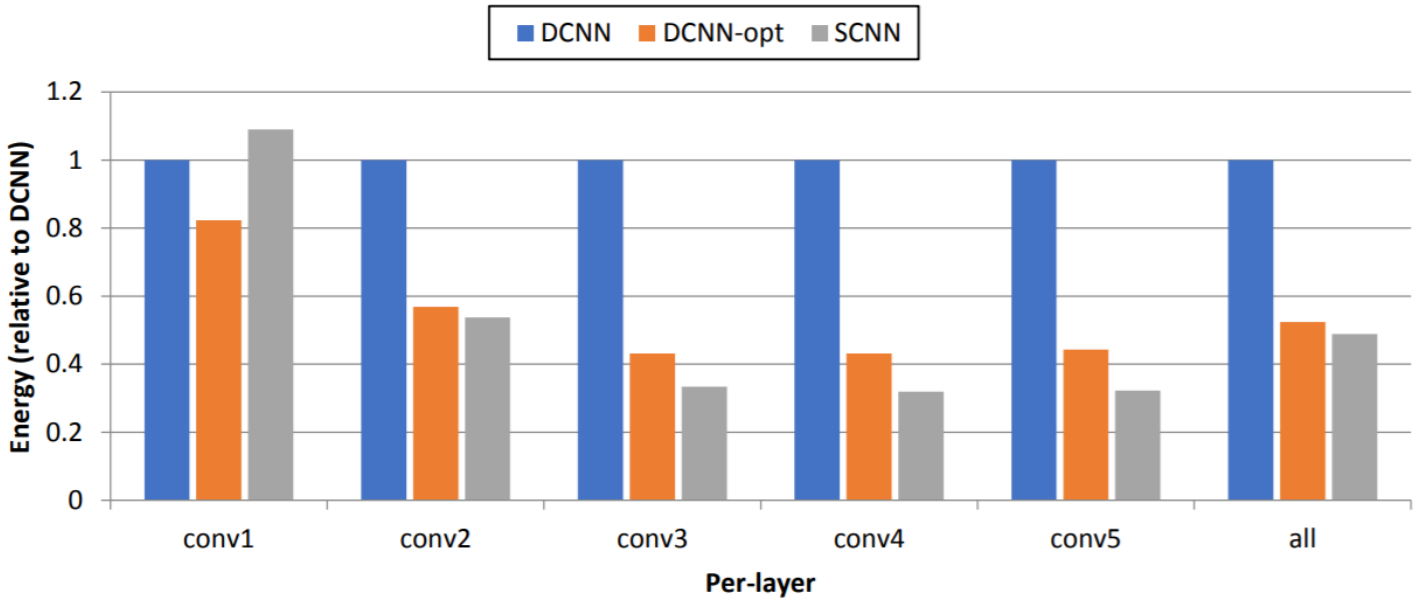


Optimized PE for accelerating compressed Sparse CNNs

Sparsity

Outer product

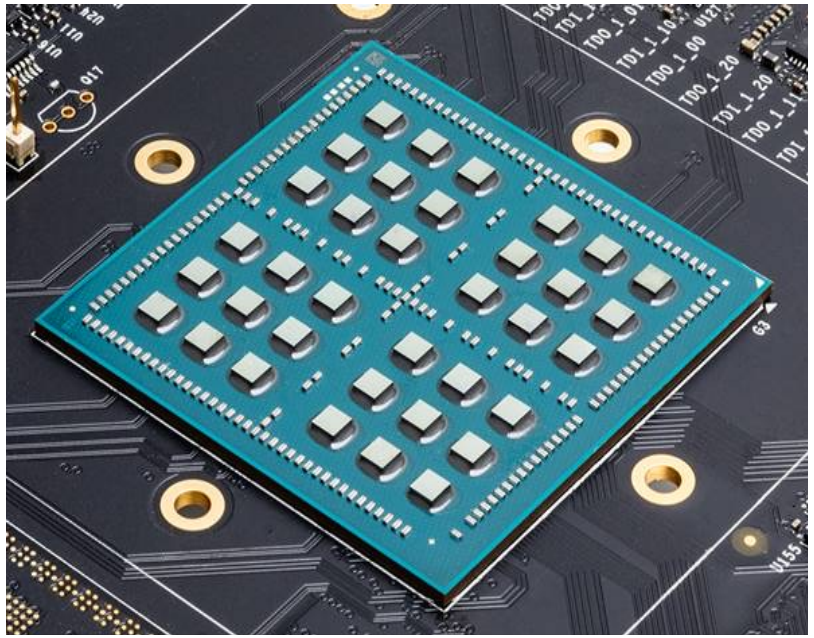
Scatter-Add



(a) AlexNet

NVIDIA

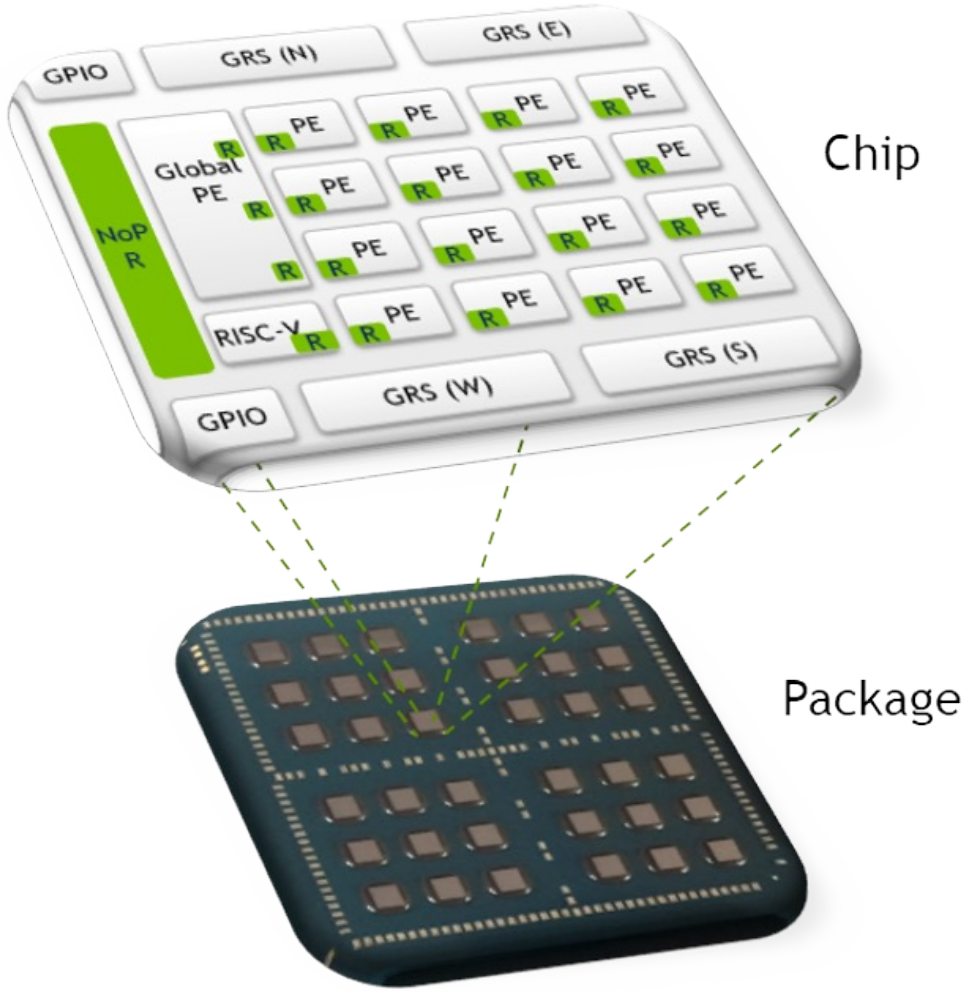
SIMBA (RC18) (2019)



Tiled PEs in a scalable MCM
128 TOPS
0.11 pJ/Op

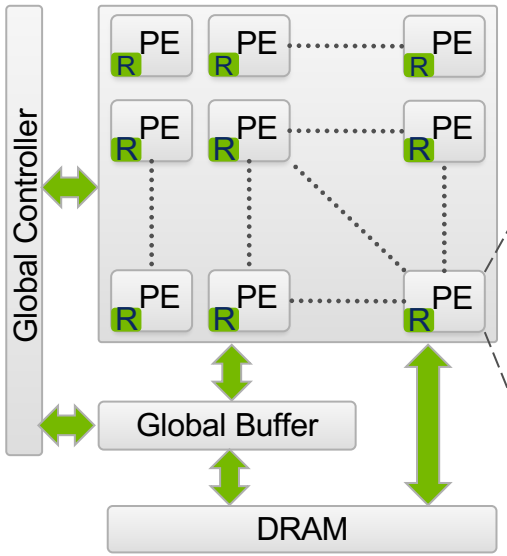
NVIDIA

Scalable
MCM
Hierarchical Mesh

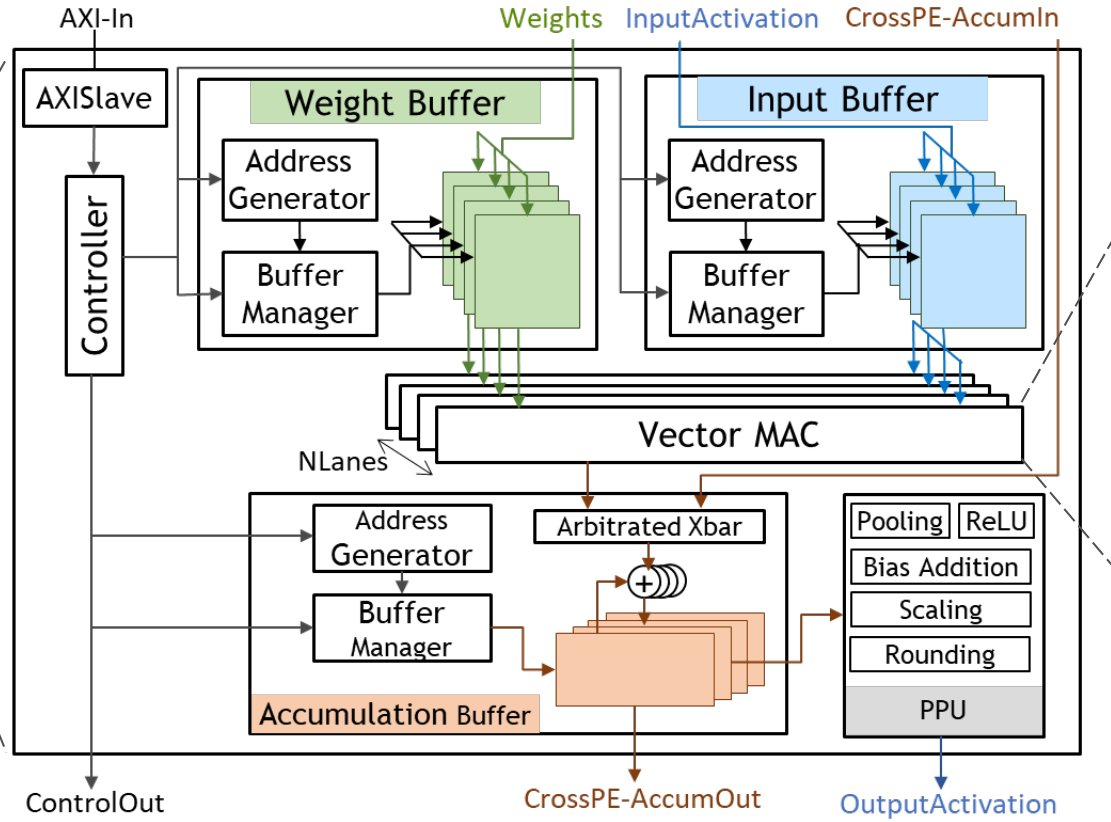


MAGNET

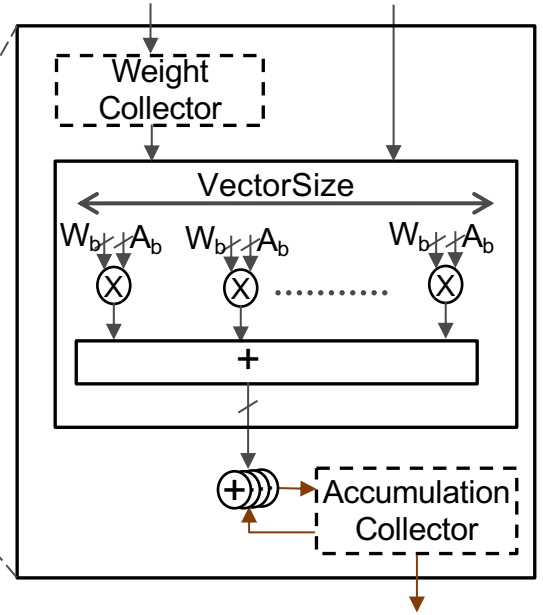
Configurable using synthesizable SystemC, HW generated using HLS tools



MAGNet System



Processing Element (PE)

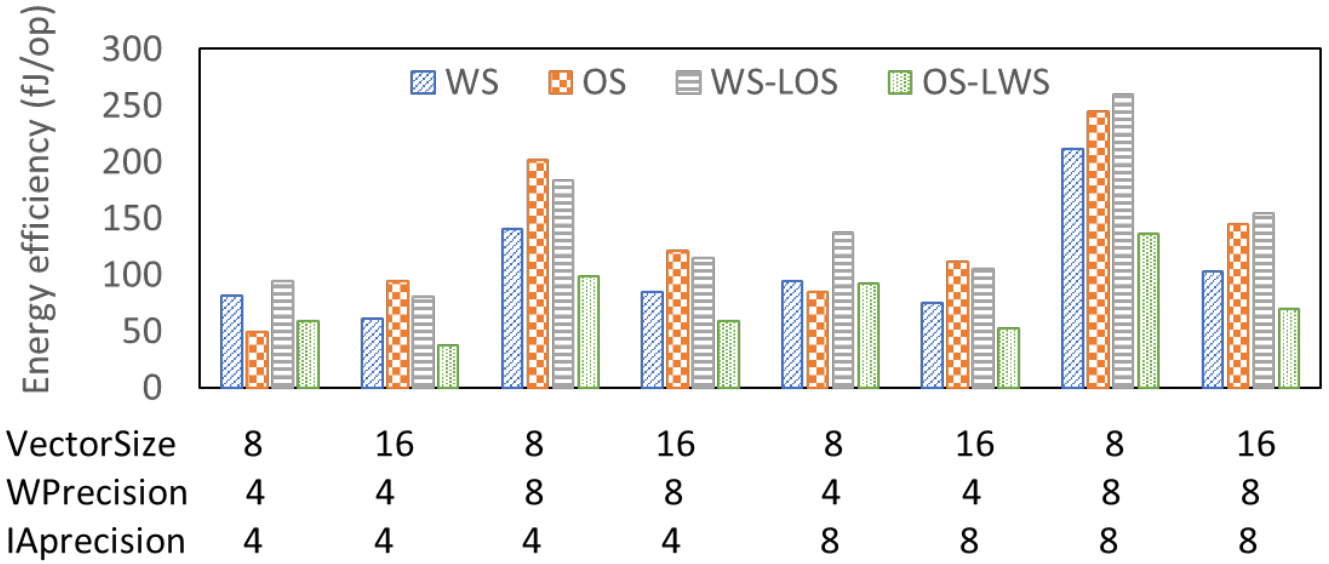


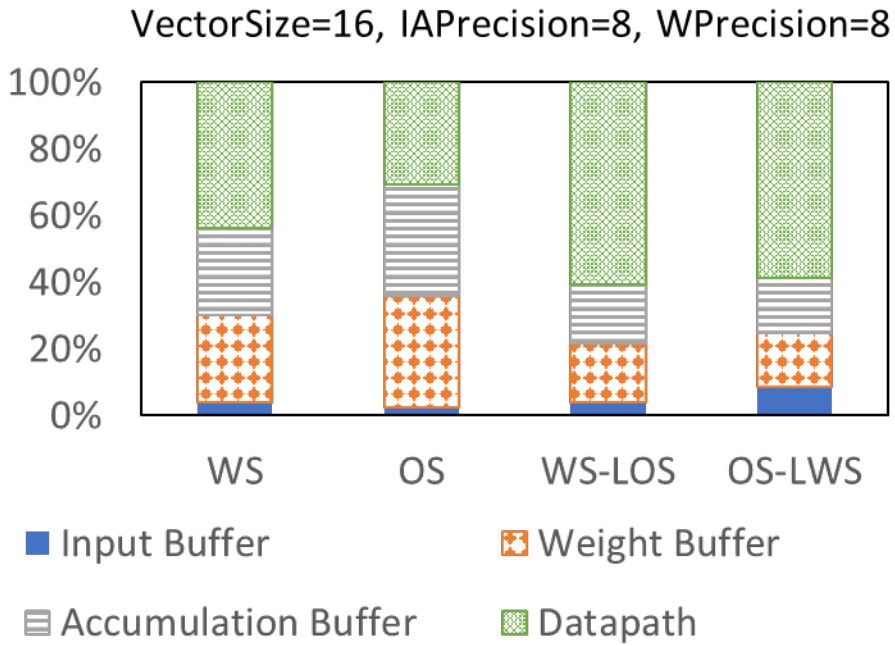
Vector MAC unit

[Venkatesan et al., ICCAD 2019]

MAGNET RESULTS

Design Space Exploration for ResNet-50





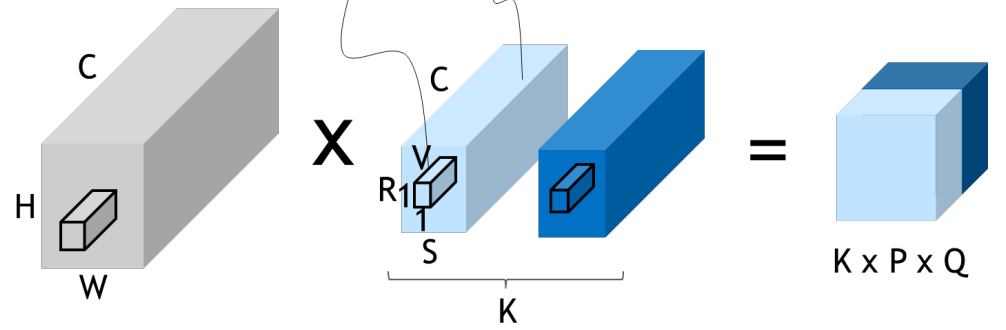
43% Energy Efficiency Improvement from Multi-Level Dataflows

VS-Quant

Per-Vector Scaled Quantization for Low-Precision Inference

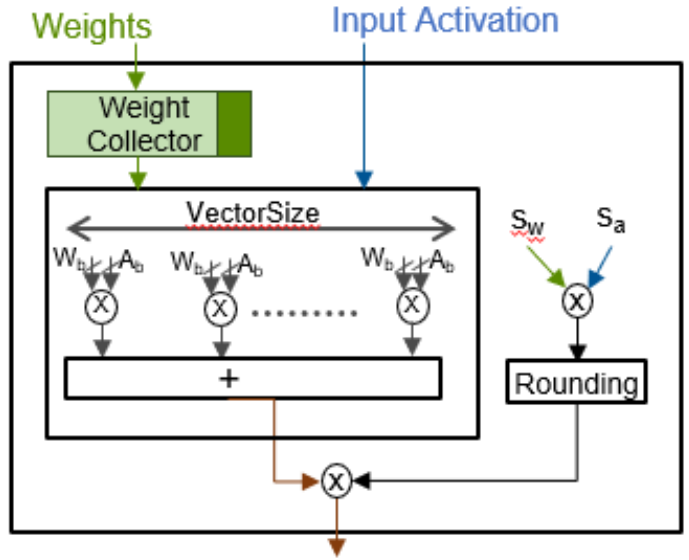
$$v \approx v_{int} \cdot \underbrace{s'_{int} \cdot s''}_{s'}$$

s'_{quant} s''



K x R x S x ceil(C/V) integer scale factors
+
K floating-point scale factors

$$y_q(j) = \left(\sum_{i=0}^{vecsize-1} w_q(i) a_q(i) \right) s_w(j) s_a(j)$$



*Fine-grained scale factors per **vector***

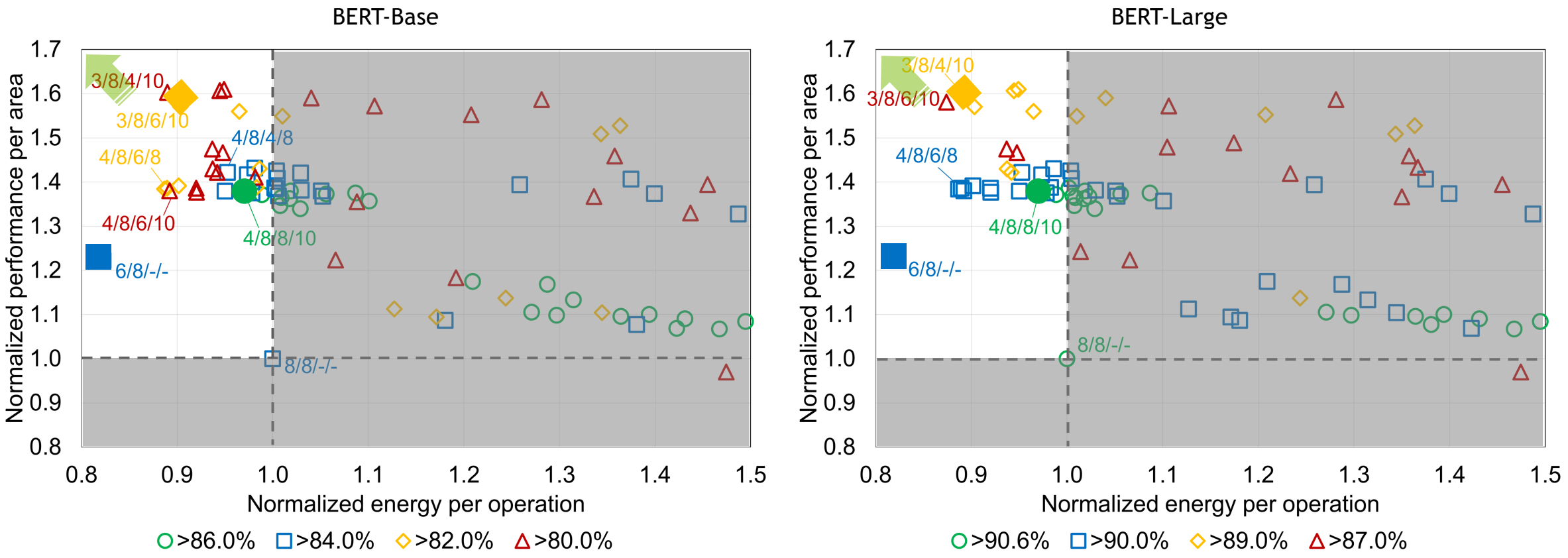
Modified vector MAC unit for VS-Quant

Works with either post-training quantization or quantization-aware retraining!

[Dai et al., MLSYS 2021]

Energy, Area, and Accuracy Tradeoff

BERT-base and BERT-large on SQuAD



* Amount of scale rounding varies among design points

Weight Width / Activation Width / Weight Scale Width / Activation Scale Width
"-" indicates per-channel scaling

Accelerators

- Start with a matrix multiplier
- Tiling (dataflow)
  - Maximize re-use from memory hierarchy
  - Number of levels and size of each level are free variables
- Sparsity
  - Compression (memory and communication)
  - Data gating
  - Sparse computation
- Number representation
  - Coding (makes math expensive)
  - Scaling (put the bits where the do the most good)\
  - Scale by the vector

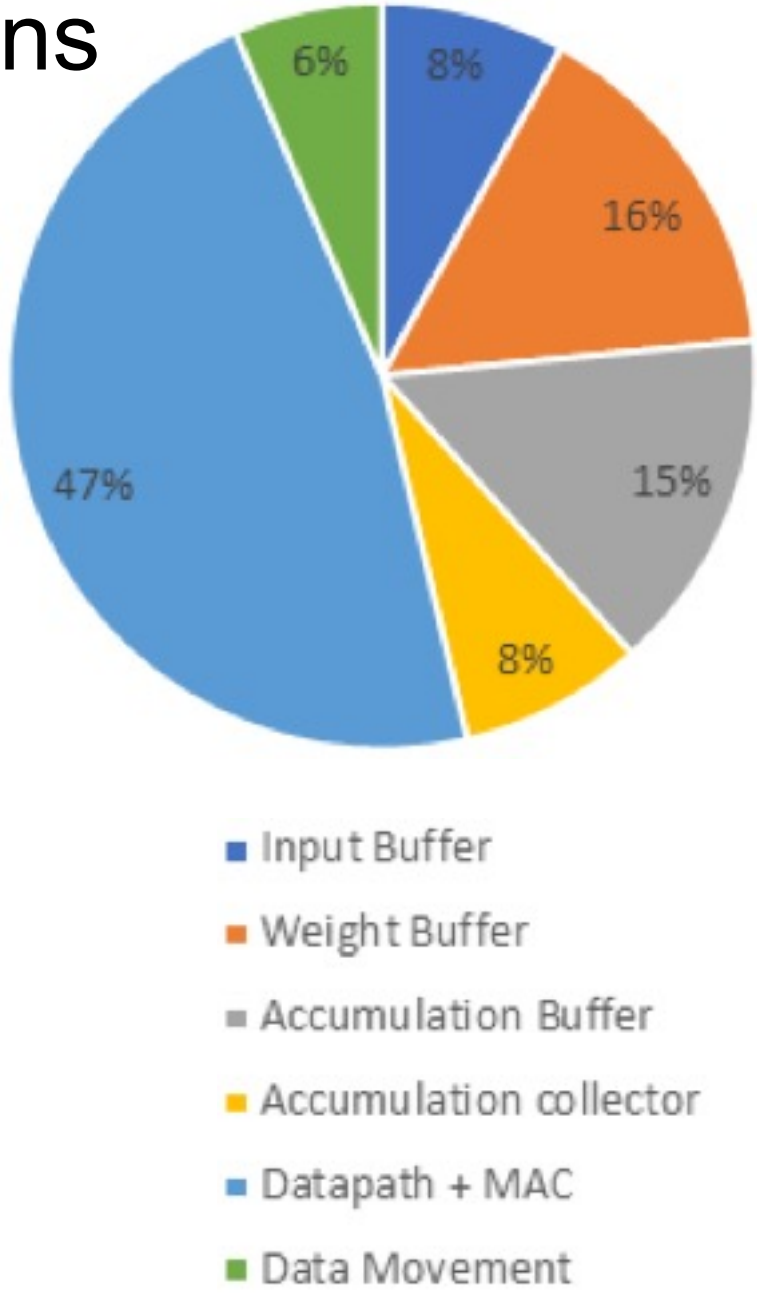
What works in an accelerator will ultimately become part of the GPU

A GPU is a platform for domain-specific hardware

Future Directions

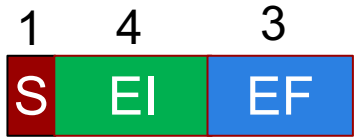
Future Directions

- Log number representation
  - Much cheaper math
  - Smaller numbers
- Better tiling
  - Lower memory energy
- Circuits
  - Memory
  - Communication
- Sparsity
  - Activations
  - Lower density
- Process
  - Capacitance scaling



$$v = -1^{s}2^{ei.ef}$$

Log4.3

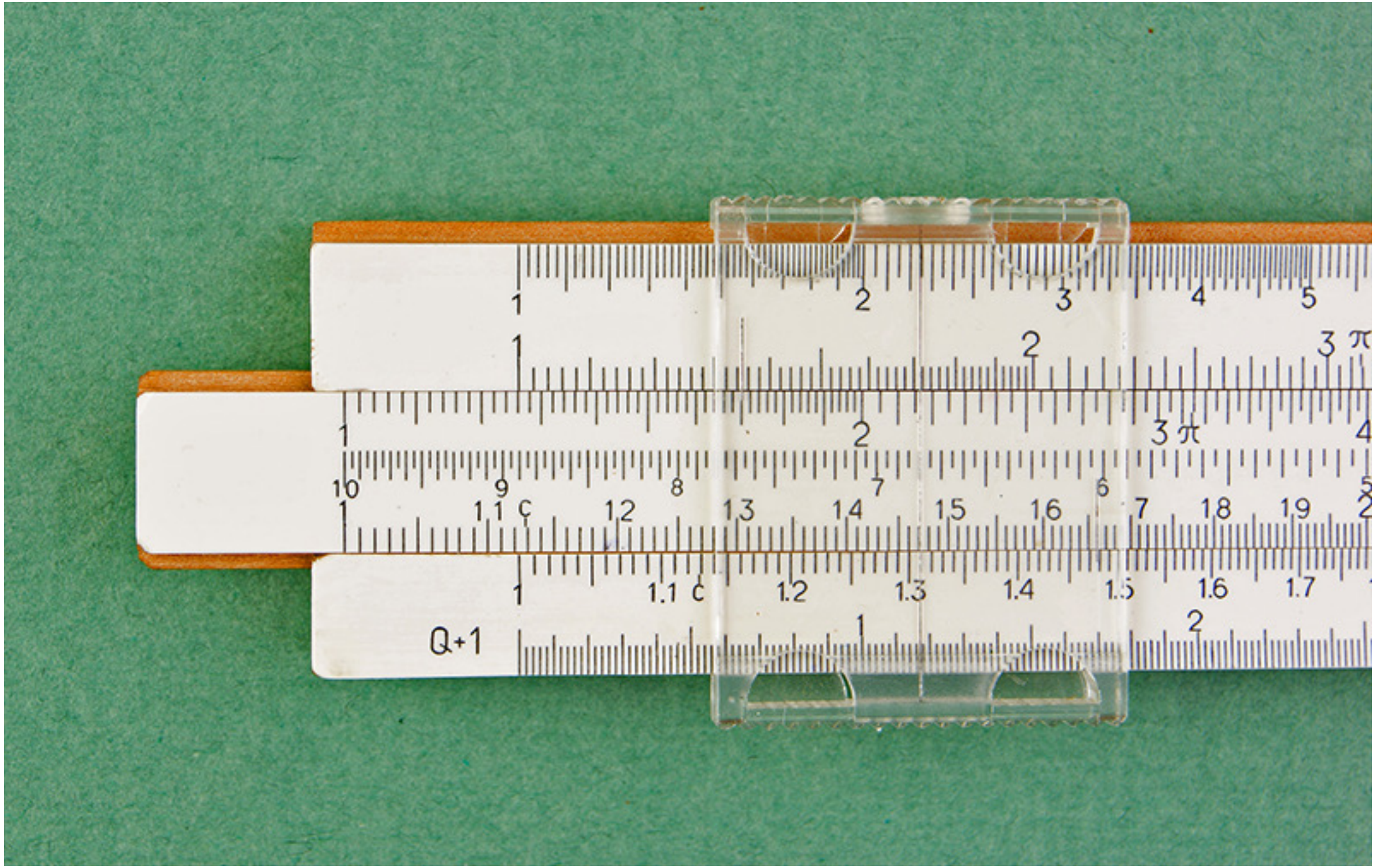


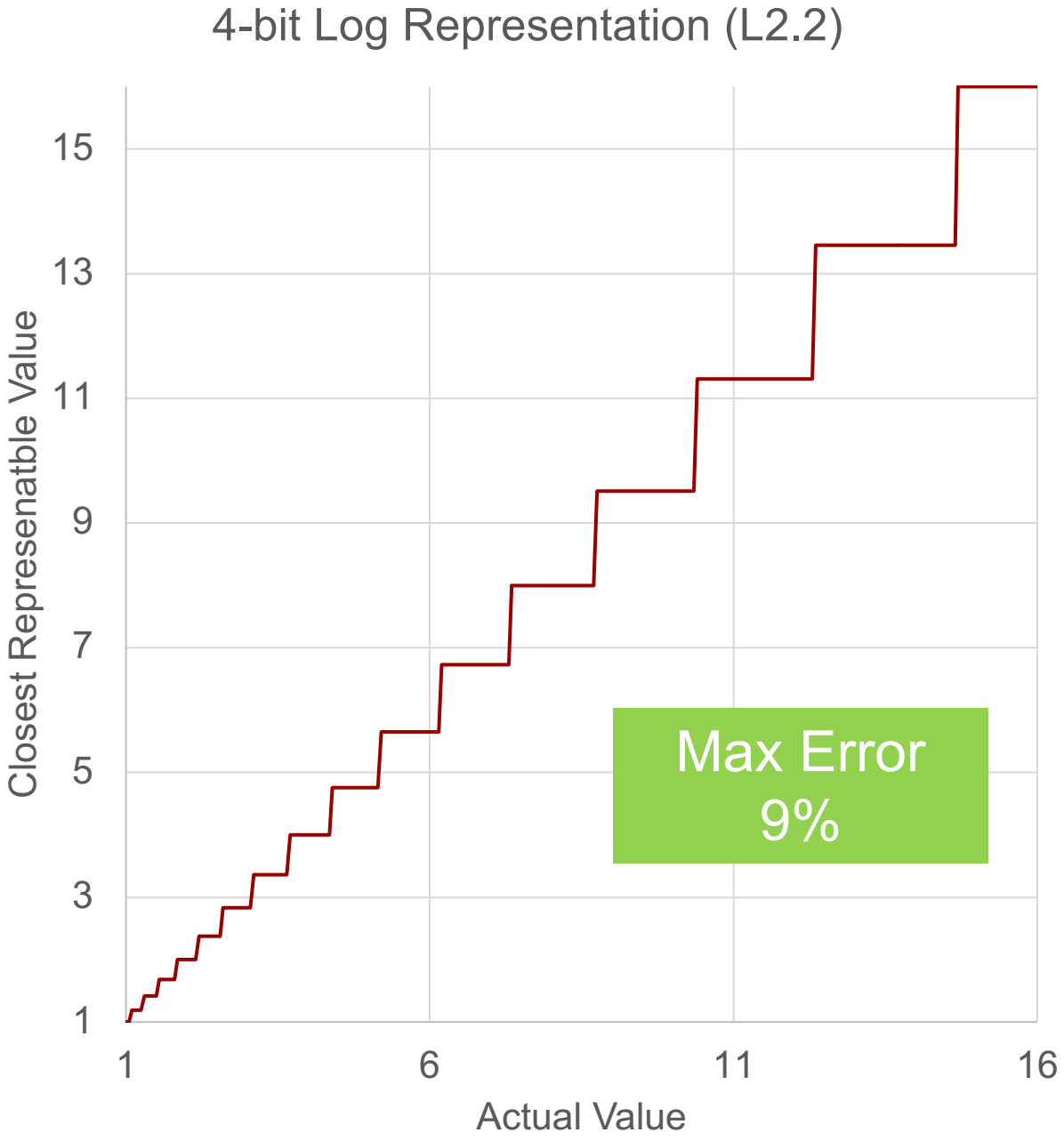
Dynamic Range 10^5
WC Accuracy 4%

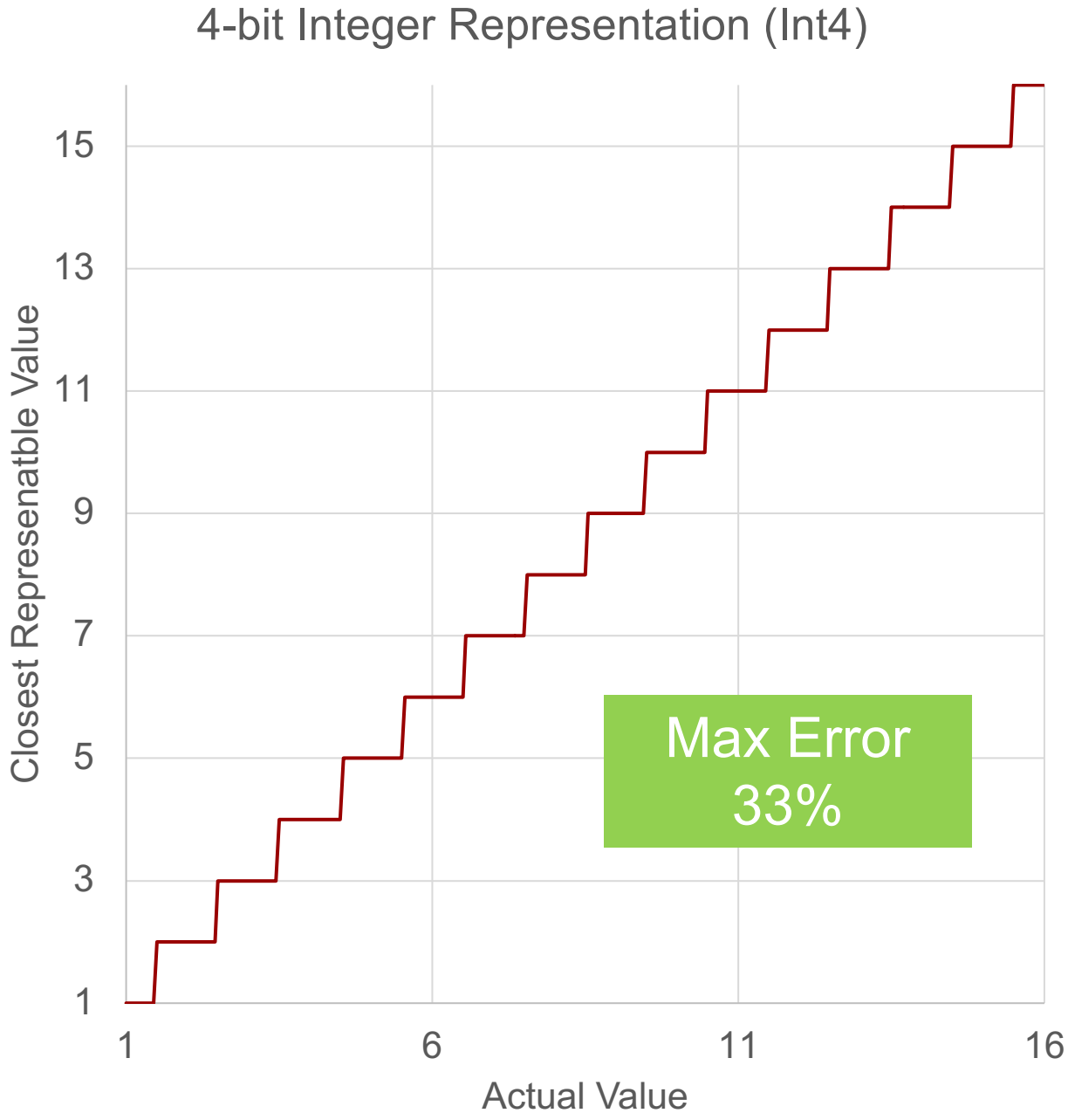
Vs Int8 – DR 10^2
WC Accuracy 33%

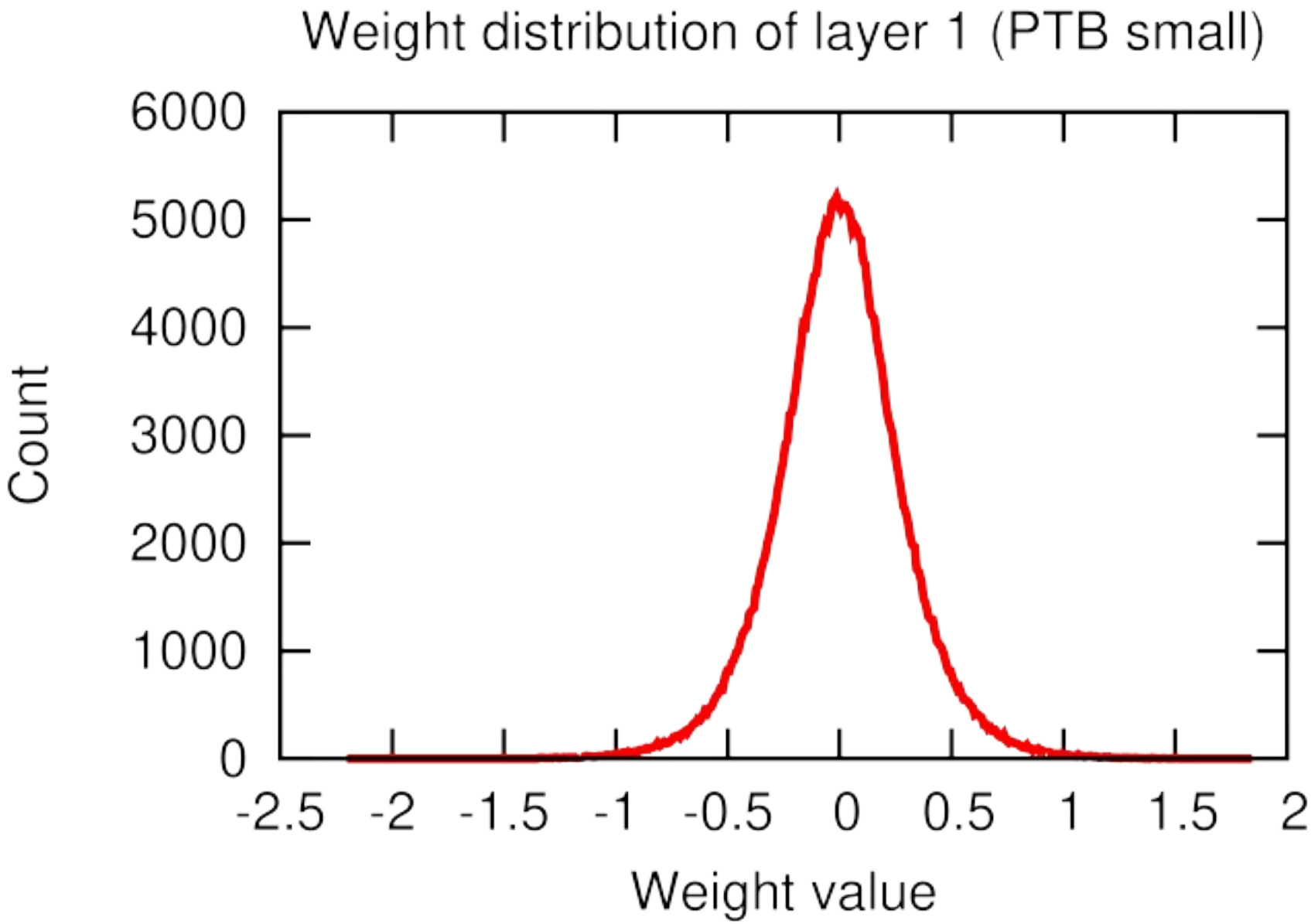
Can apply offset to EI to represent any range of 16 integers, e.g., -8 to 7 (scaling)

Numbers near zero need special treatment


1 2 3 4 5
1 2 3 π
1 2 3 π 4
10 9 8 7 6 5
1 1.1 C 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2
1 1.1 C 1.2 1.3 1.4 1.5 1.6 1.7
Q+1
1 2


4-bit Log Representation (L2.2)
Closest Represenatble Value
15
13
11
9
7
5
3
1
1
6
11
16
Actual Value
Max Error
9%


4-bit Integer Representation (Int4)
Closest Represenatble Value
15
13
11
9
7
5
3
1
1
6
11
16
Actual Value
Max Error
33%


Weight distribution of layer 1 (PTB small)
Count
6000
5000
4000
3000
2000
1000
0
-2.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
Weight value

(19) **United States**

(12) **Patent Application Publication**

Dally et al.

(10) **Pub. No.: US 2021/0056446 A1**

(43) **Pub. Date: Feb. 25, 2021**

(54) **INFERENCE ACCELERATOR USING LOGARITHMIC-BASED ARITHMETIC**

(71) Applicant: **NVIDIA Corporation**, Santa Clara, CA (US)

(72) Inventors: **William James Dally**, Incline Village, NV (US); **Rangharajan Venkatesan**, San Jose, CA (US); **Brucek Kurdo Khailany**, Austin, TX (US)

(21) Appl. No.: **16/750,823**

(22) Filed: **Jan. 23, 2020**

Related U.S. Application Data

(63) Continuation-in-part of application No. 16/549,683, filed on Aug. 23, 2019.

Publication Classification

(51) **Int. Cl.**
G06N 5/04 (2006.01)

(52) **U.S. Cl.**
CPC ***G06N 5/04*** (2013.01); *G06N 20/00* (2019.01)

(57) **ABSTRACT**

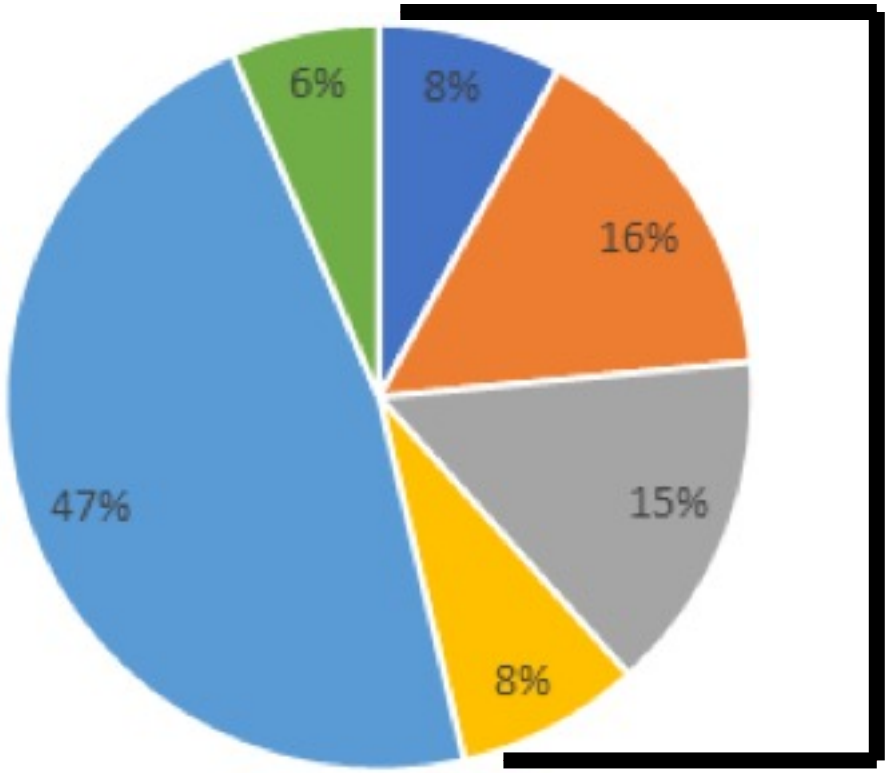
Neural networks, in many cases, include convolution layers that are configured to perform many convolution operations that require multiplication and addition operations. Compared with performing multiplication on integer, fixed-point, or floating-point format values, performing multiplication on logarithmic format values is straightforward and energy efficient as the exponents are simply added. However, performing addition on logarithmic format values is more complex. Conventionally, addition is performed by converting the logarithmic format values to integers, computing the sum, and then converting the sum back into the logarithmic format. Instead, logarithmic format values may be added by decomposing the exponents into separate quotient and remainder components, sorting the quotient components based on the remainder components, summing the sorted quotient components using an asynchronous accumulator to produce partial sums, and multiplying the partial sums by the remainder components to produce a sum. The sum may then be converted back into the logarithmic format.

Sparsity

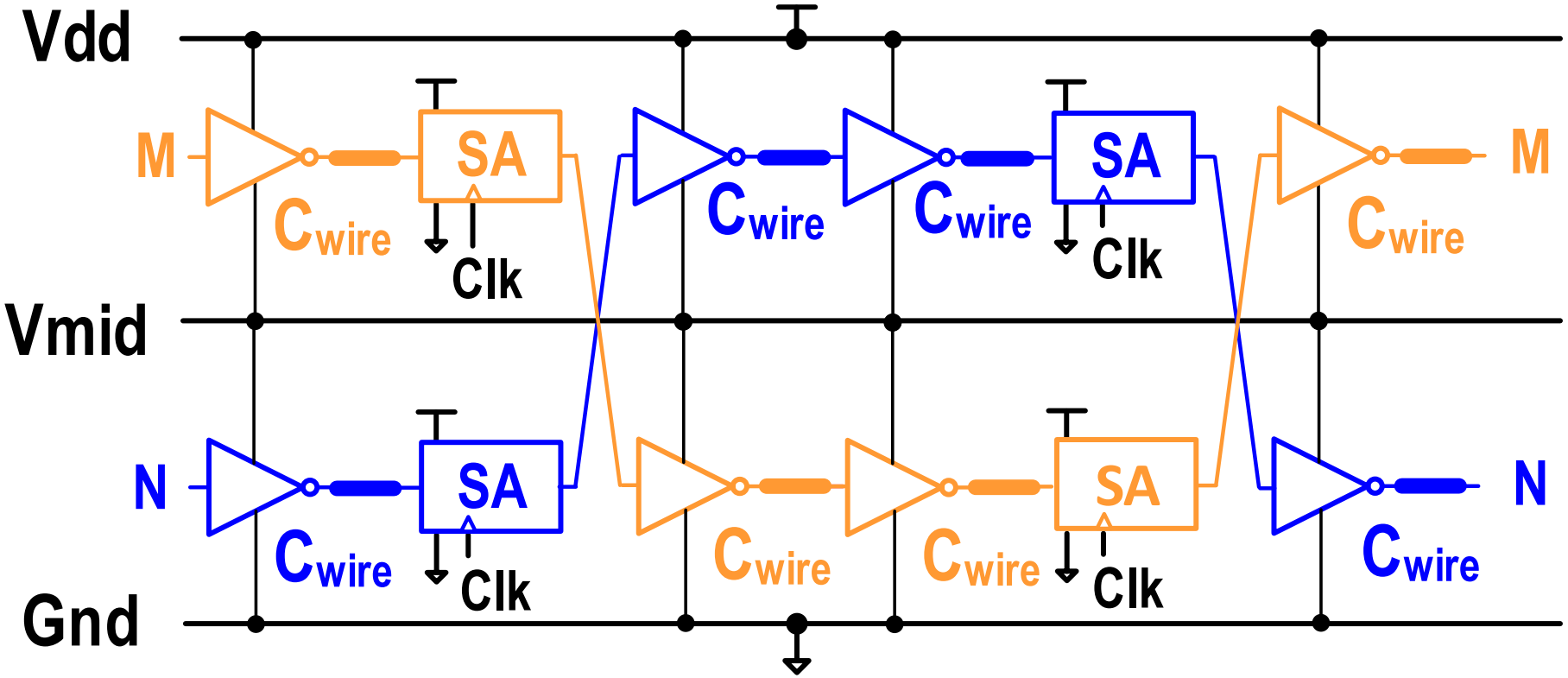
- Getting a real energy win from sparsity surprisingly difficult
  - Reduction in arithmetic ops easily swamped by cost of irregularity
  - Glitching is a particular problem
  - EIE and SCNN give marginal gains at system level
- Ampere structured sparsity is a great start
- Much more possible
  - Sparse activations as well as weights
  - Lower density (25%, 10% vs 50%)

Memory Circuits

- Optimize energy of memory access
- Small read-mostly memories
  - Latch arrays
  - Move output mux to consumer
- Most memory energy is comm energy
  - Lower swing



Communication Circuits
4x Energy Saving – 5-10fJ/bit-mm

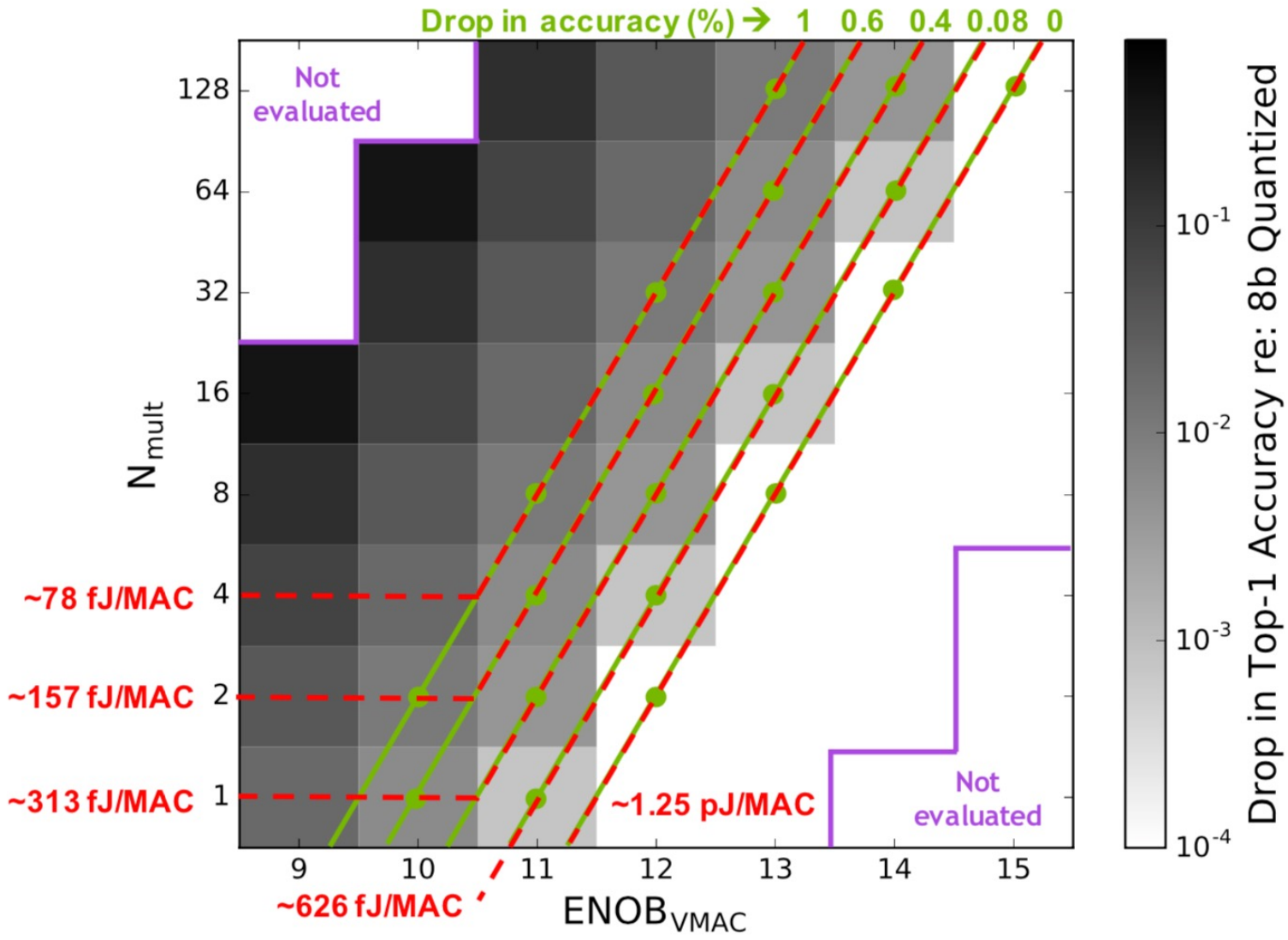


Wilson, J., et al., "A 6.5-to-23.3fJ/b/mm Balanced Charge-Recycling Bus in 16nm FinFET CMOS at 1.7-to-2.6Gb/s/wire with Clock Forwarding and Low-Crosstalk Contraflow Wiring," *ISSCC*, 2016.

Process

- Numbers above are for 16nm (unless otherwise stated)
- Denard scaling is dead, but capacitance does scale
  - Slower than linear
- Expect a factor of 2-2.5x

Analog/Optical Not Competitive



Rekhi et al. “Analog/Mixed-Signal Hardware Error Modeling for Deep Learning Inference, DAC 2019

Conclusion

Conclusion

- GPU inference performance doubling every year
  - Better number representation, FP16, Int8, Int4, …
  - Complex instructions, DP4A, HMMA, IMMA
  - Sparsity
  - Plumbing
- Accelerators experiment with new techniques
  - Sparsity, Tiling (data flows), Number Representation
- How do we keep doubling?
  - Log numbers
  - Sparsity
  - Memory and communication circuits
  - Process