The Future of Low-bitwidth Reconfigurable and Parallel AI Computing

2023/12/12

Masato Motomura

AI Computing Research Unit (ArtIC)
Tokyo Institute of Technology (Tokyo Tech)

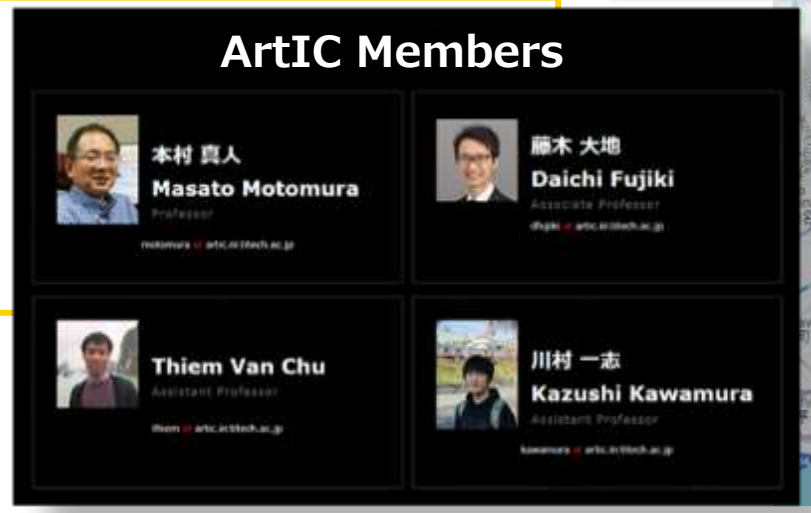
Introduction: Myself and Our Group (ArtIC@Tokyo Tech)

- ～ March 2011 **NEC Research Laboratories**
  - Parallel processors, near memory computing
  - **Dynamically reconfigurable processor (DRP)**

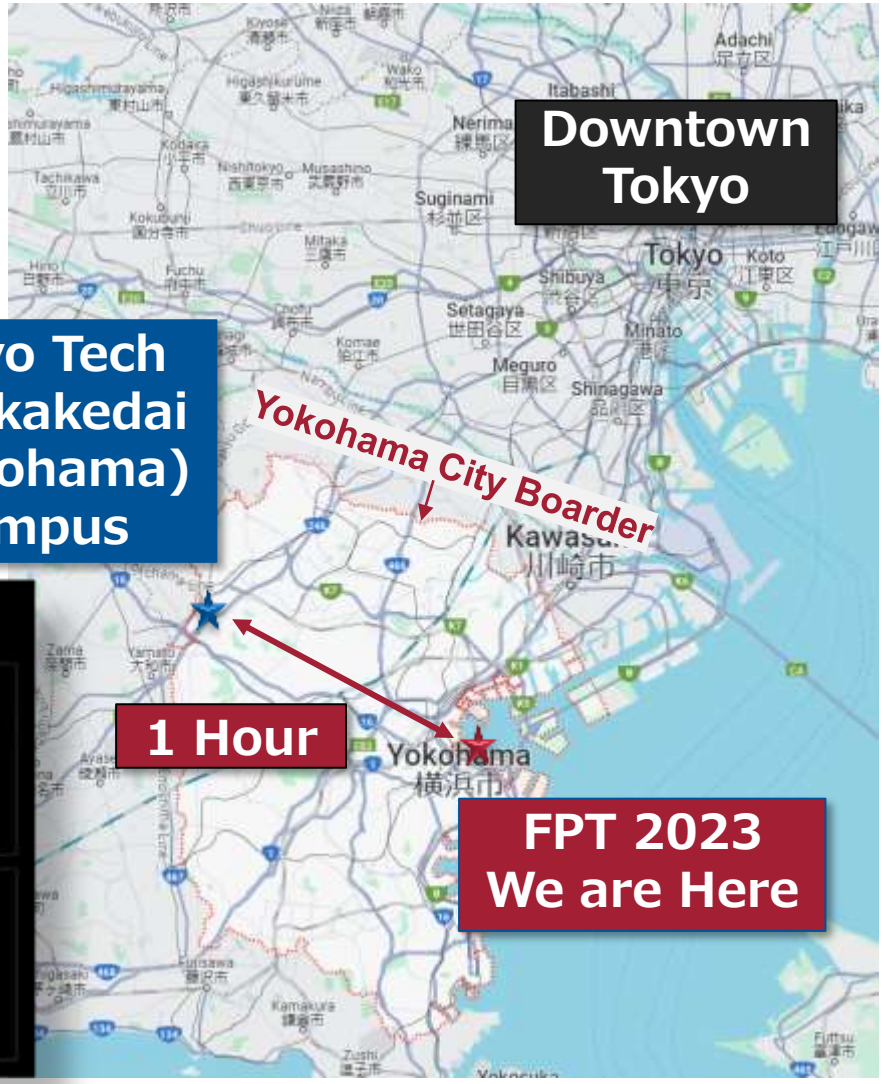
Part 1 (Brief Overlook)

- ～ March 2019 **Hokkaido University**
  - Reconfigurable, near/in-memory accelerators for AI computing
- April 2019 ～ **Tokyo Tech**
  - **Art**ificially **I**ntelligent **C**omputing Research Unit
  - => **ArtIC** (<= Art of IC)
  - In **Yokohama**

Part 2 (A Bit In Depth)



Greater Tokyo Area



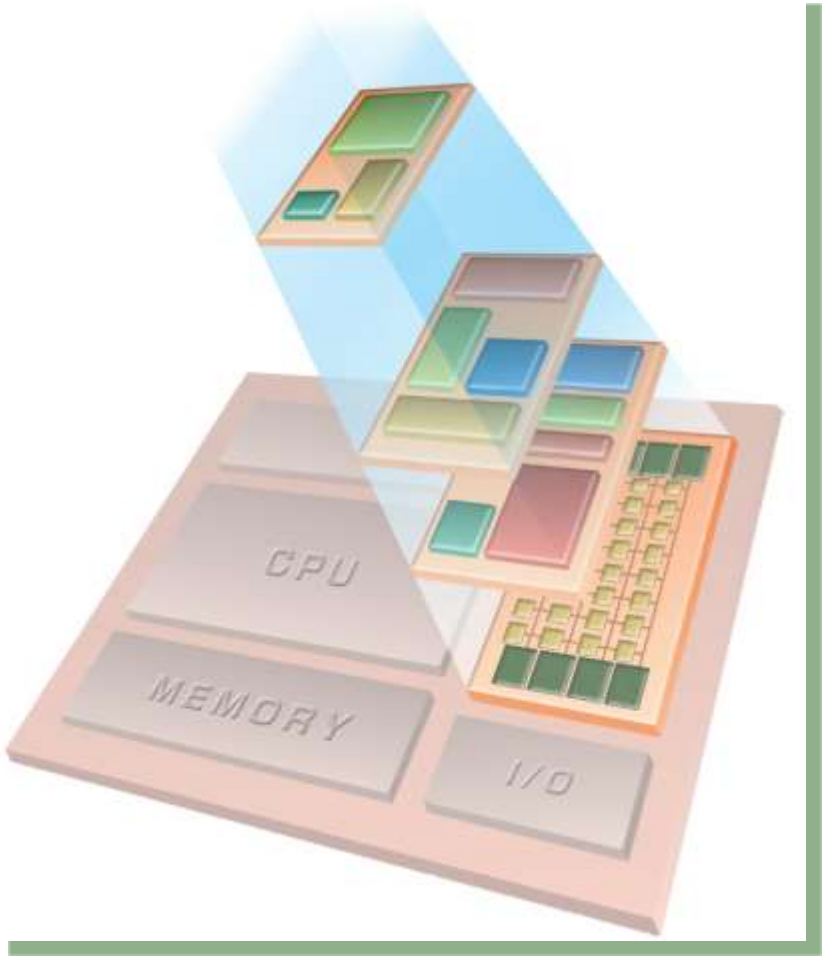
Part 1: Dynamically Reconfigurable Processor (DRP)

Tokyo Tech

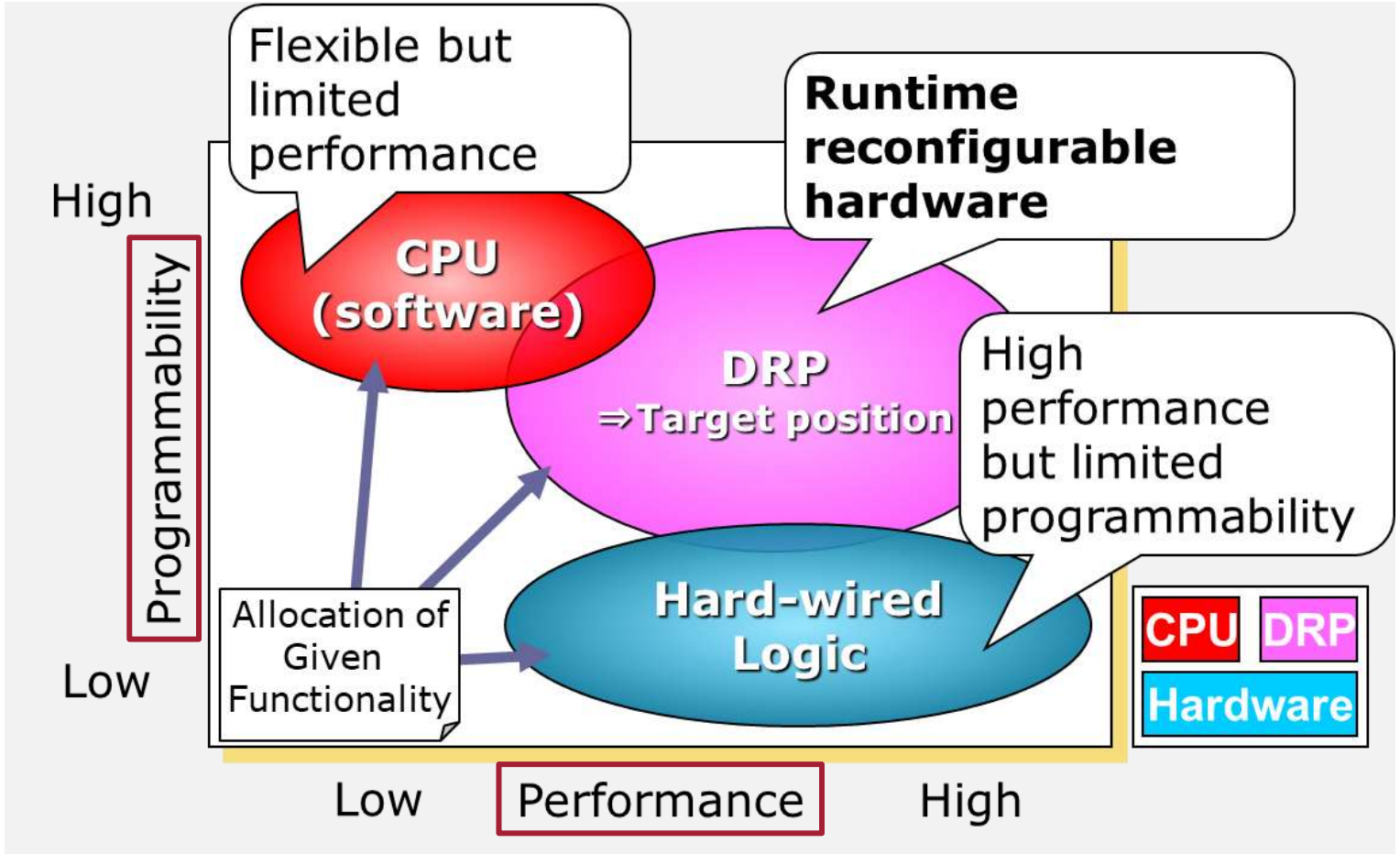
ARTIC

DRP Research Started Around 2000

An Accelerator IP core in an SoC



Filling the gap between CPU and hard-wired logic



DRP: 1st Presented at Microprocessor Forum 21 Years Ago

November 25, 2002

PDF Version

New NEC Array Speeds Data

NEC Introduces Its Dynamically Reconfigurable 512-Processor Array

By Max Baron

Digital media and communications are in their infancy. Most of their development and deployment roadmaps are still in the future, but they promise to become an indispensable part of everyday life. For computer architects, the new applications represent both challenges and rewards. The workloads are data intensive and require performance levels that are often impractical to implement with general-purpose processors. Challenge and opportunity are engendering specialized architectures that are competing for a chance to show their might and enjoy a slice of revenues that may rival those of the PC market. Two years ago, in Japan, NEC's research team started looking at an interesting engine that could be used in the new applications.

On October 16, 2002, at the annual **Microprocessor Forum**, Masa Motomura, an architect at NEC's System ULSI Development Division, unveiled details of the company's new massively parallel architecture, a dynamically reconfigurable processor (DRP). The new architecture can be used as a network processor or as a DSP engine in applications requiring high performance.

DRP Brings Together Three Powerful Concepts

The DRP is not the first-ever massively parallel engine, nor will it be the last, but the innovative features that set it apart really demand a second look. Three notable features stand out from the rest. To begin with, most arrays are designed as network processors or as DSP engines; the DRP can perform both functions. It can also pinch-hit as a semiefficient, but working, general-purpose processor.

Second, NEC's architects have created an architecture that can change its array configuration on a cycle-by-cycle basis, making these changes indistinguishable, timing-wise, from instructions. Most other designs have defined longer-reconfiguration delays that work best if the resulting interconnections are kept fixed for the duration of a thread.

Finally, the DRP applies a different solution to the propagation delays that must be taken into account as data moves across the chip. Where most other architectures are synchronizing units via clocked registers and processing elements (PE), the DRP can define multiple propagation paths to become one pipe stage—a small asynchronous engine walled between clocked registers to make it cooperate with other parts of the array.

PE Architecture Supports Flow-Through Data

Figure 1 shows the DRP's byte-wide processing element, which consists of a data-management unit (DMU) and an ALU designed to operate on 8-bit and 1-bit data. The DMU can execute 25 instructions that include inversion, shifting, masking, and constant generation, using 8-bit and 1-bit operands. A special command named WIRE is used to cause the DMU to pass the operand unchanged to the PE's outputs. The ALU can execute 23 arithmetic/logic instructions on 8-bit data and can use a carry propagation path to process data that is wider than 8 bits. Like the DMU, the ALU has a WIRE command.

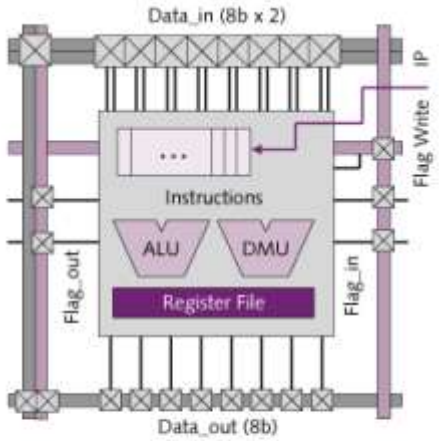


Figure 1. Simplified block diagram of NEC's processing ele
file, which can
used as a pipe
flow-through d
instructions fo
units also hold
PE.

Guess Who He is…

… Long Story, Hmm

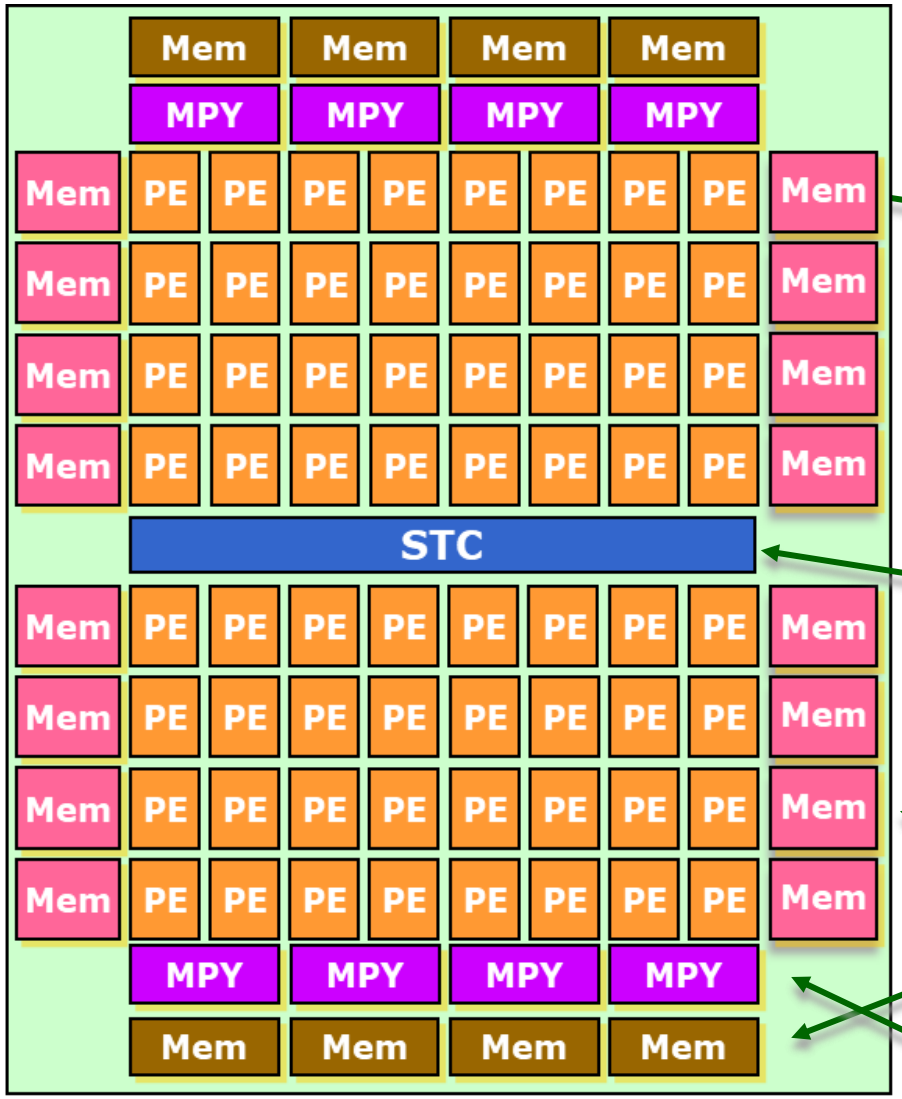
Masa Motomura of NEC unveiled details about DRP at MPF 2002. Photo by Ross Mehan.

Aside from the usual multiwindowed programmer interface, NEC's compiler offers graphic views of the scheduled dataflow graph and the scheduled state-transitions diagram. Place and route-determined connections are also displayed to help in analyzing critical-path delays. The programmer can assign a critical-path delay to be used by the high-level synthesis program. The program will divide the implementation into multiple states to fit within the critical-path-delay budget. It is expected that the visual display of information could help in speeding up place-and-route work but will be of limited use in programming and debugging complex code. The DRP has been provided with internal logic to help debug programs.

The DRP in Action

System designers must be able to take advantage of the chip's most prominent features: applicability beyond digital signal processing, one-cycle datapath change, and dataflow. NEC's architects have endowed the PE with capabilities that can support general data-intensive processing, but they had to add eight 32-bit multipliers to meet DSP needs such as could be encountered in high-end image processing. NEC's compiler provides a seamless environment for writing code aimed at PEs and multipliers.

DRP Features Tiled CGRA Architecture



Single Tile

CGRA: Coarse Grained Reconfigurable Array

Processing Element (PE)

- Byte-oriented ALUs
- Byte-width X/Y buses and registers
- Several tens of configuration sets

State Transition Controller (STC)

- **Controls “dynamic reconfiguration”**

Data Memory (Mem)

- Dual port
- Single port

16b Multiplier (MPY)

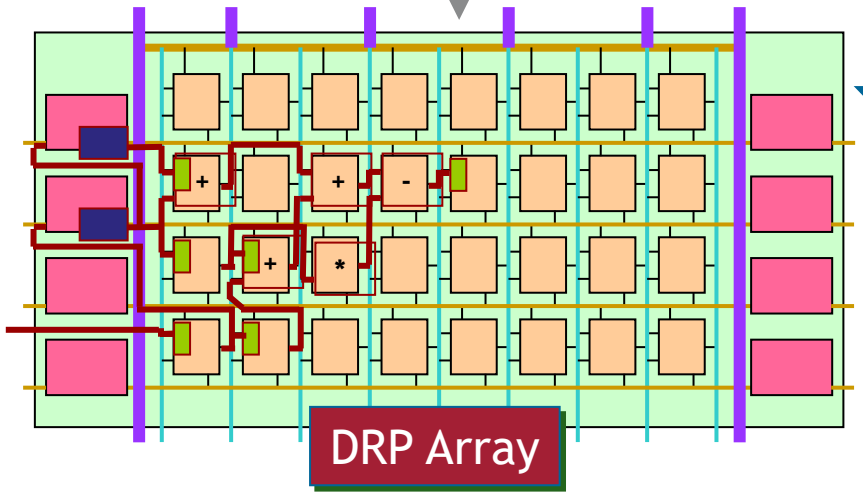
Execution Model (1): Spatial Mapping

Example: 3x3 Filter

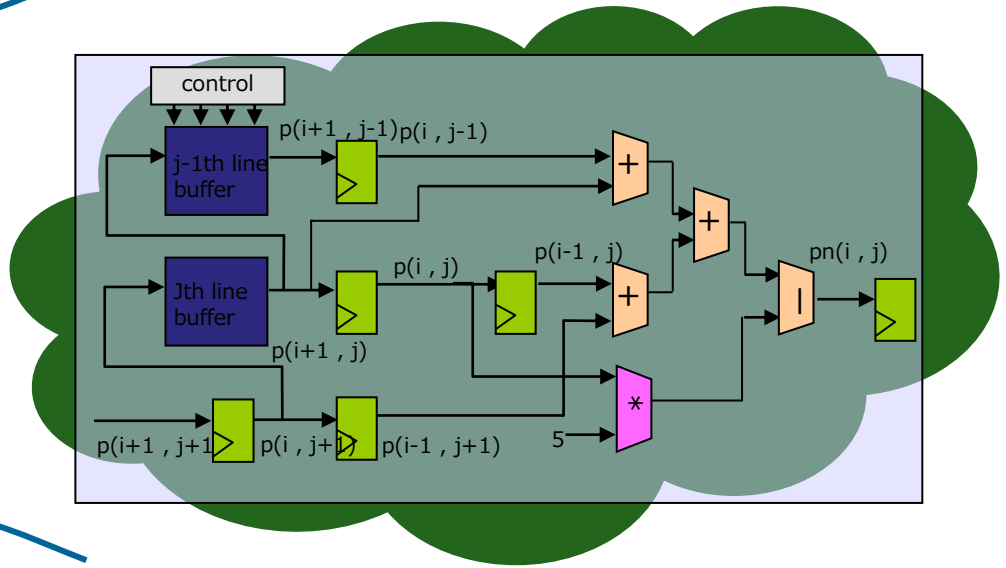
```
for( i = 0; i < N; i++ ){
    for( j = 0; j < N; j++){
        f(i, j) = 5*f(i, j) - f(i, j-1) - f(i-1, j)
                  - f(i+1, j) - f(i, j+1);
    }
}
```

Source Code in C-langage

DRP Compiler



1. Generates a **HW configuration context** from the source code



2. **Spatially** maps onto the array

Execution Model (2): Temporal Sequencing

Source Code in C-language

DRP Compiler

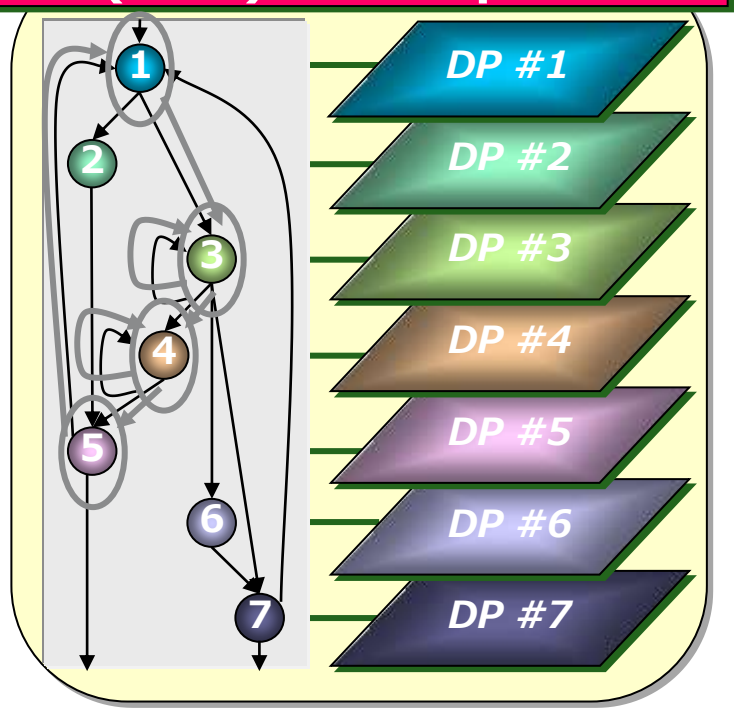
High-Level Synthesis based on CWB* Tool & Technology Mapping

*CWB: CyberWorkBench

Switch among several tens of HW contexts cycle by cycle

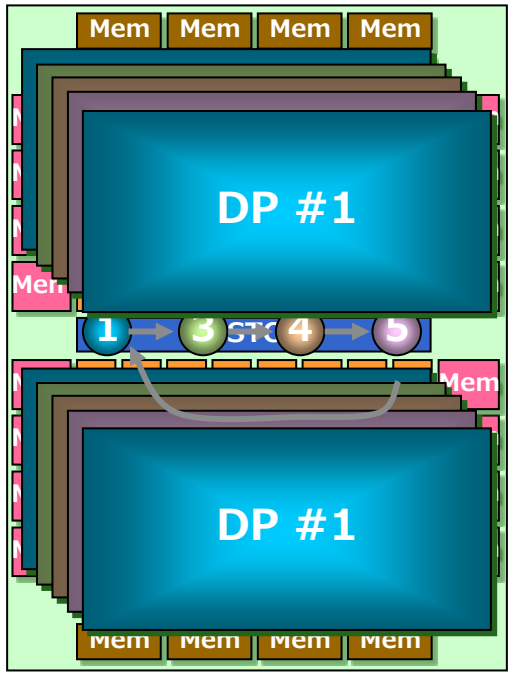
Finite state machine (FSM) + datapath

Reconfiguration time: Hidden behind datapath operation

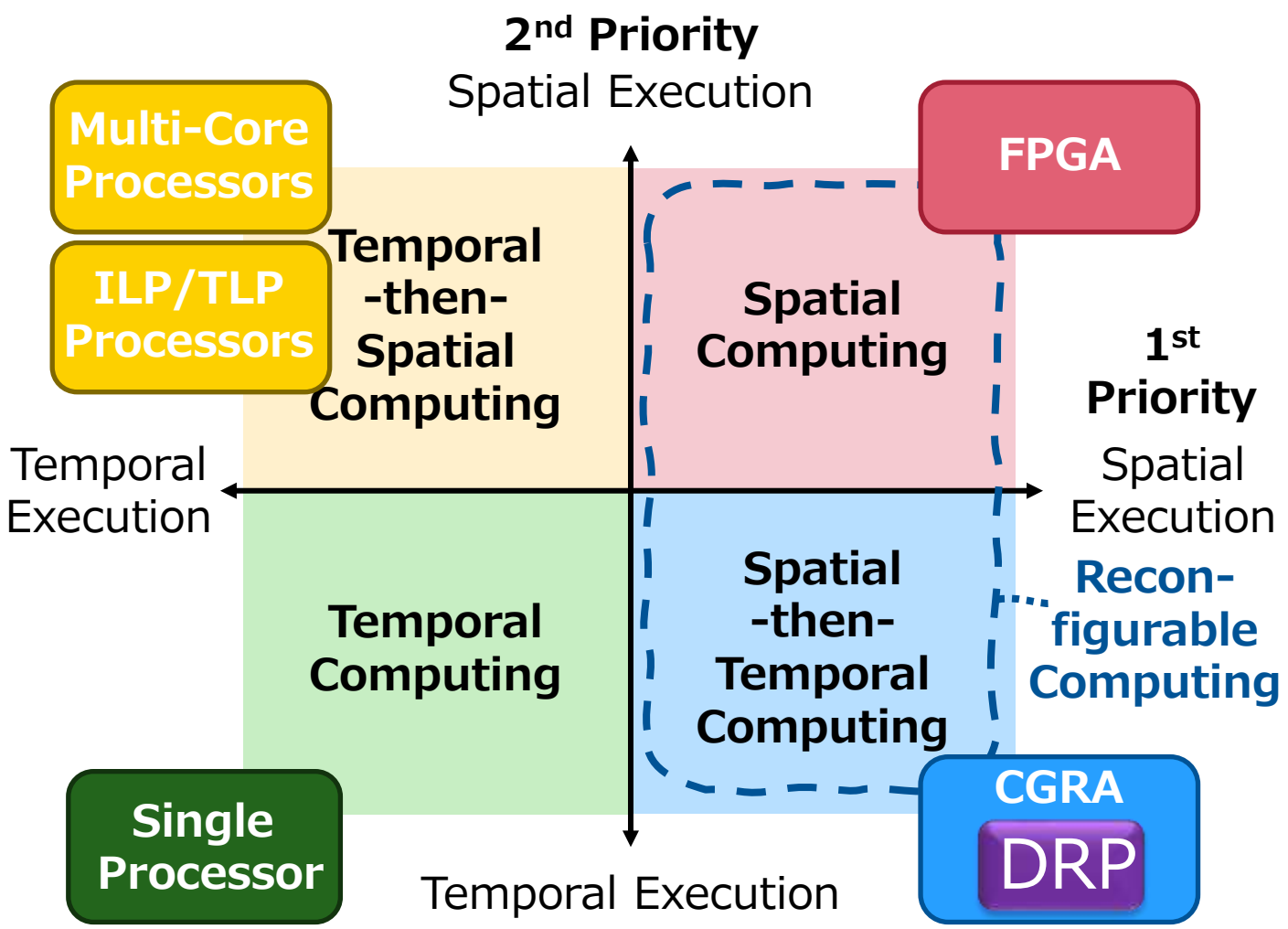


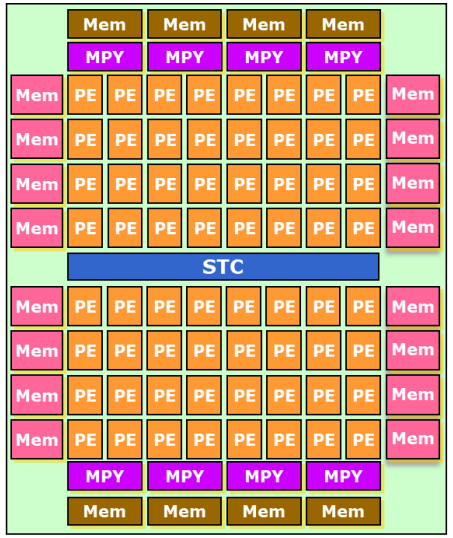
Architecture

State by State Reconfiguration



Putting DRP in Execution Model Landscape





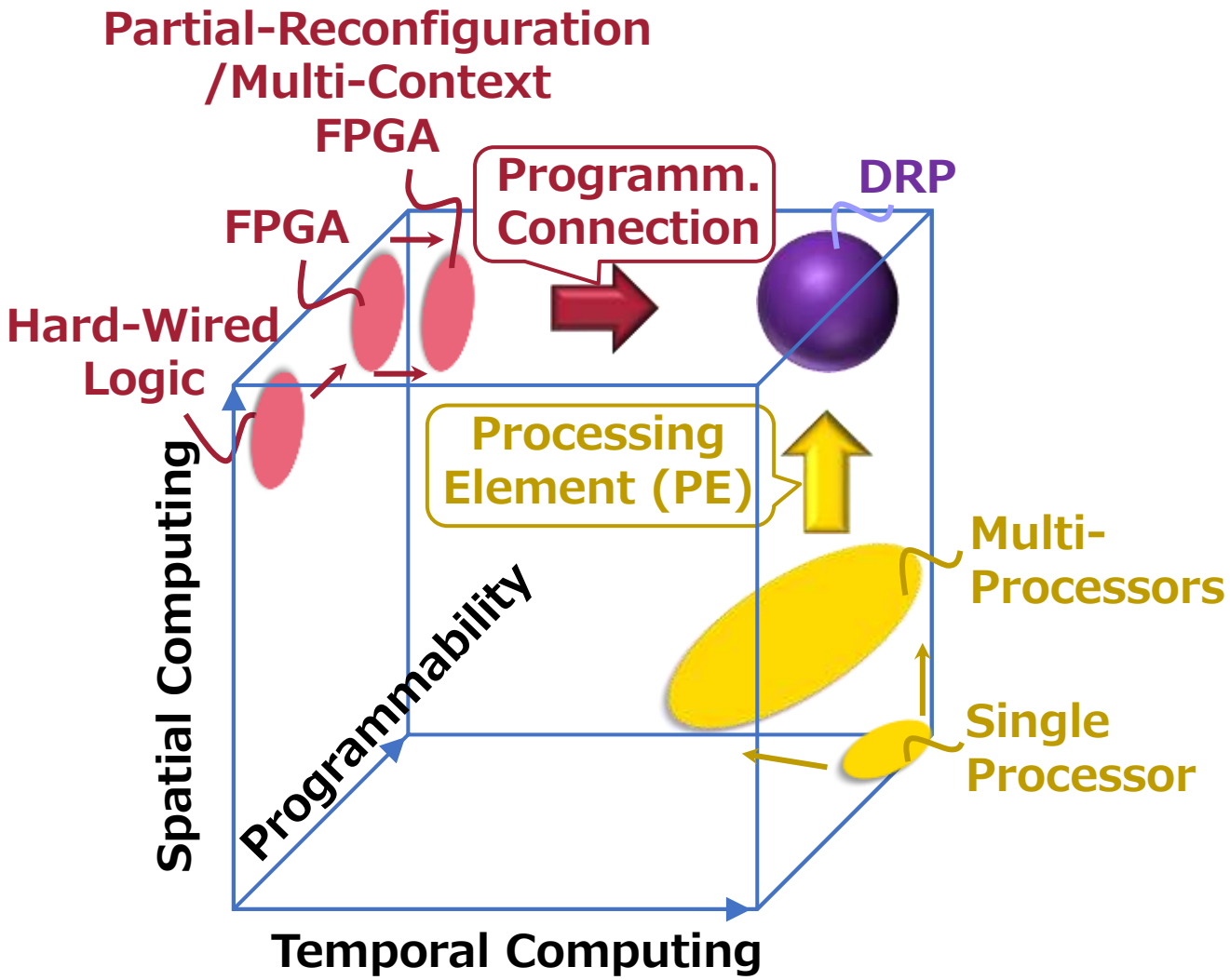
FAQ:
Is a diagram like this a multi-core processor or a CGRA* Core?

The answer lies in its execution model

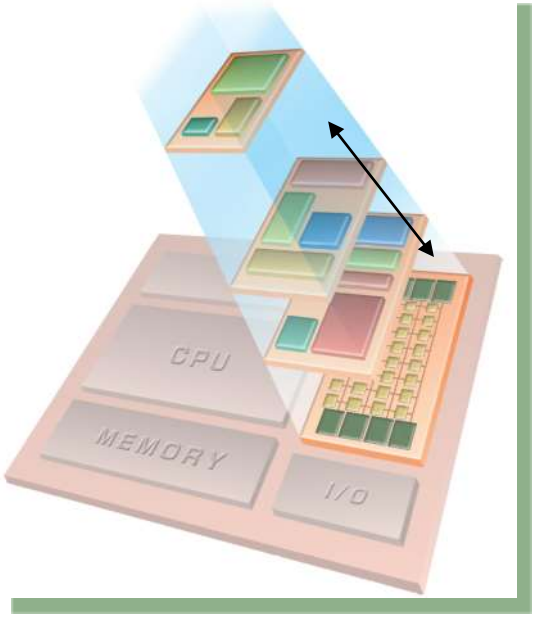
***Coarse-Grained Reconfigurable Array**

DRP represents a **Spatial-then-Temporal** CGRA with FSM-Controlled Dynamic Reconfiguration

Putting DRP in Execution Model Landscape – In 3D

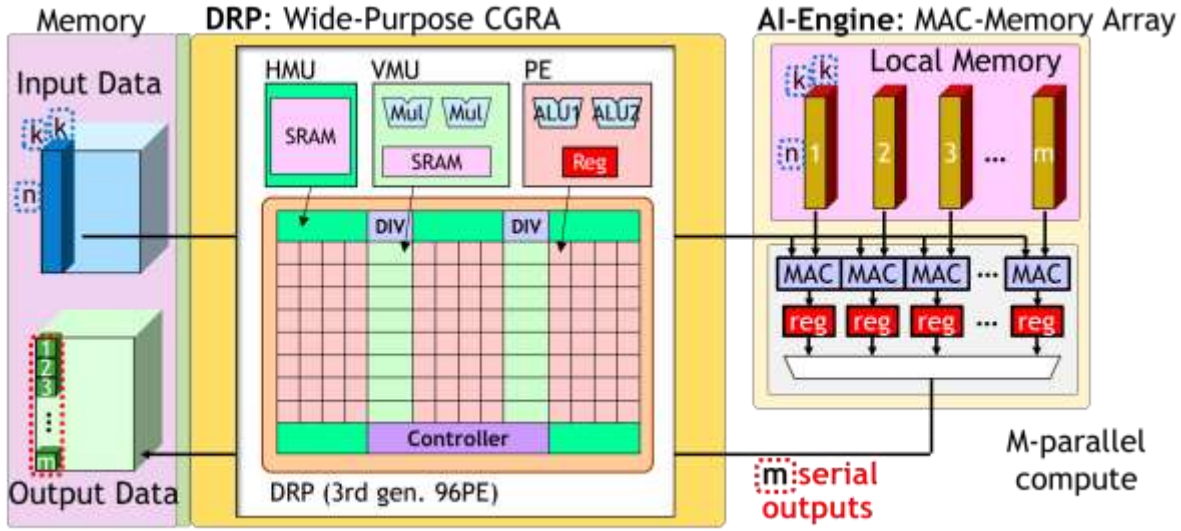


DRP is a **Spatial-then-Temporal** CGRA with FSM-Controlled Dynamic Reconfiguration



Its Cycle by Cycle Datapath Context Switch is for Hardware Virtualization

Recent Evolution: DRP-AI for Neural Networks



Renesas RZ/V2M with DRP-AI Wins 2020 Aspencore's World Electronic Achievement Award (WEAA)

Angus Chan
Senior Manager, MPU Product Department

RZ/V2M, the first product in the RZ/V series of microprocessors (MPUs), which features DRP-AI (Dynamically Reconfigurable Processor), Renesas' exclusive vision-optimized artificial intelligence (AI) accelerator, was selected as a winner of 2020 Aspencore's World Electronic Achievement Awards in the Processor/DSP/FPGA category.

The WEAA honor companies, individuals and excellent products that make outstanding contributions to innovation and development in electronics industry worldwide. The companies, individuals and products nominated for various awards are industry leaders, which fully reflect their leading position and extraordinary in the industry. The winners are jointly selected by a judging committee composed of ASPENCORE global senior industry analysts and website users from Asia, the United States and Europe.

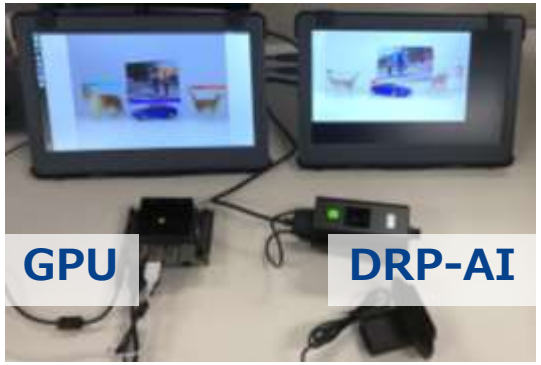
In Industrial 4.0 / IIoT application scenario, those manufacturing systems are designed for advanced production control and predictive maintenance. Therefore, the realization on how to make the terminal equipment become more intelligent is particularly important. Nevertheless, the huge size of AI model and complex computational logic calculation is very big challenge to performance requirement of MCU, which not only takes up a lot of memory, but also generates very large-scale complex mathematical operations. And huge calculations greatly inhibit the real-time operation of on-site equipment.

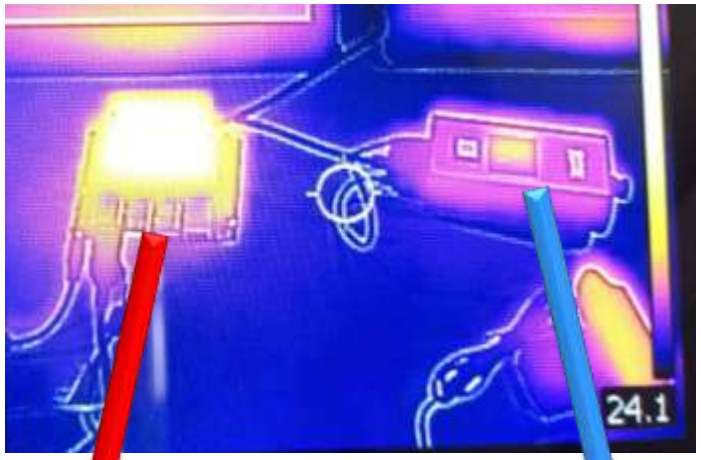
In order to solve these problems, Renesas Electronics introduced RZ/V2M equipped with DRP-AI (Dynamic Reconfigurable Processor) which utilize its hardware resources to accelerate the data calculation of AI model embedded in the terminal equipment by peeling off the entire AI computing process from the CPU. By utilizing powerful computing ability of hardware arithmetic unit to complete the entire AI inference process without taking up CPU resources. Ultimately, efficient intelligence is achieved on terminal equipment and devices.

▪ Solution Features:
Complete AI inferences independently without CPU involvement.

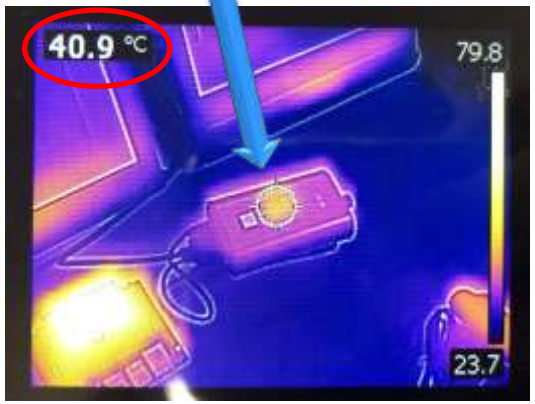
Now used in Renesas's MCU/MPU products.
Total shipment of DRP chips is still rapidly expanding!

DRP-AI Demo & Its New Gen. Exposure at ISSCC 2024

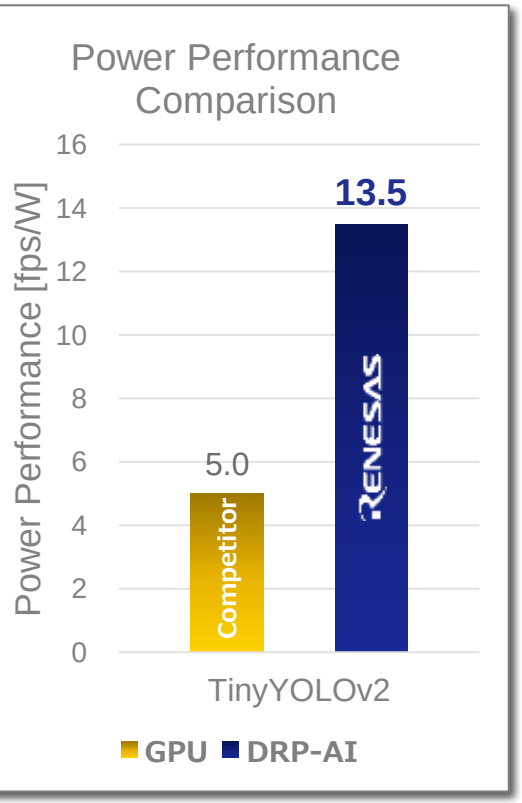




GPU



DRP-AI



Note:
- The benchmark uses the power consumption of the entire board and inference time without pre and post process.
- Measured by Batch size=1 and FP16 Quantization.
- TensorRT7 is applied for Competitor A measurement.

SESSION 20 Wednesday, February 21st, 8:00 AM

Machine Learning Accelerators

Session Chair: Chia-Hsiang Yang, *National Taiwan University, Taipei, Taiwan*
Session Co-Chair: Ji-Hoon Kim, *Ewha Womans University, Seoul, Korea*

8:00 AM

20.1 NVE: A 3nm 23.2TOPS/W 12b-Digital-CIM-Based Neural Engine for High-Resolution Visual-Quality Enhancement on Smart Devices
M-E. Shih[1,*], *S-W. Hsieh*[1,*], *P-Y. Tsai*[1,*], *M-H. Lin*[1], *P-K. Tsung*[1], *E-J. Chang*[1], *J. Liang*[1], *S-H. Chang*[1], *C-L. Huang*[2], *Y-Y. Nian*[2], *Z. Wan*[2], *S. Kumar*[2], *C-X. Xue*[3], *G. Jedhe*[2], *H. Fujiwara*[2], *H. Mori*[2], *C-W. Chen*[1], *P-H. Huang*[1], *C-F. Juan*[1], *C-Y. Chen*[1], *T-Y. Lin*[1], *C. Wang*[1], *C-C. Chen*[1], *K. Jou*[1]
[1]MediaTek, Hsinchu, Taiwan
[2]MediaTek, San Jose, CA
[3]TSMC, Hsinchu, Taiwan
*Equally Credited Authors (ECAs)

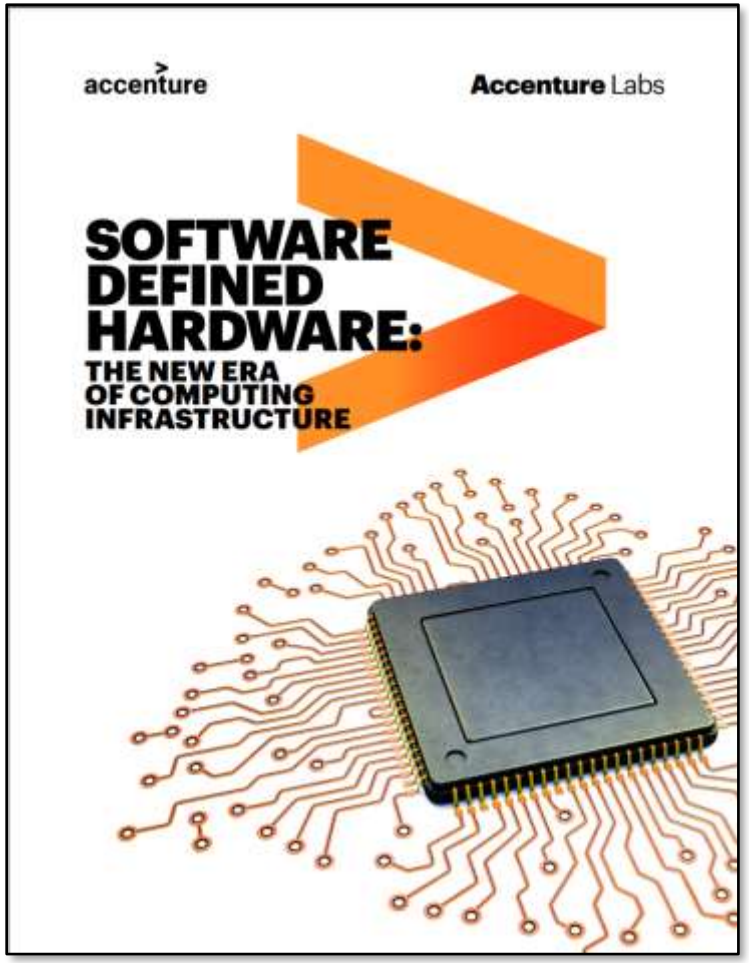
8:25 AM

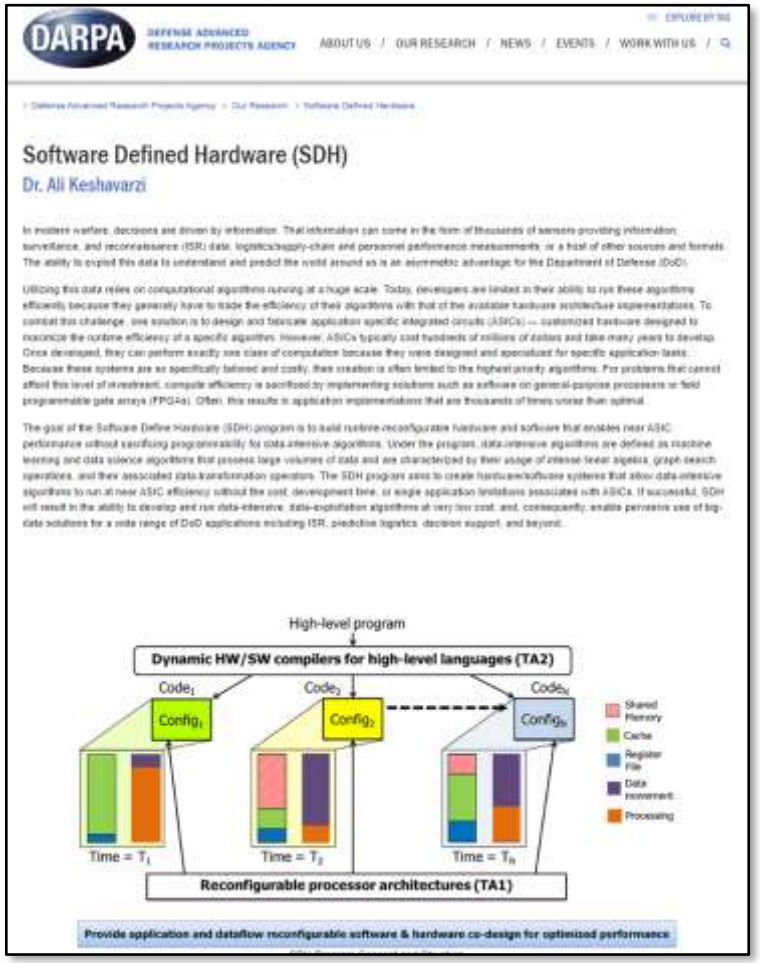
20.2 A 28nm 74.34TFLOPS/W BF16 Heterogenous CIM-Based Accelerator Exploiting Denoising-Similarity for Diffusion Models
R. Guo[1], *L. Wang*[1], *X. Chen*[1], *H. Sun*[1], *Z. Yue*[1], *Y. Qin*[1], *H. Han*[1], *Y. Wang*[1], *F. Tu*[2], *S. Wei*[1], *Y. Hu*[1], *S. Yin*[1]
[1]Tsinghua University, Beijing, China
[2]Hong Kong University of Science and Technology, Hong Kong, China

8:50 AM

20.3 A 23.9TOPS/W @ 0.8V, 130TOPS AI Accelerator with 16× Performance-Accelerable Pruning in 14nm Heterogeneous Embedded MPU for Real-Time Robot Applications
K. Nose, T. Fujii, K. Togawa, S. Okumura, K. Mikami, D. Hayashi, T. Tanaka, T. Toi
Renesas Electronics, Tokyo, Japan

DRP: Early-Coming/Ever-Evolving in SDH/SDC Movement





* Software Defined Hardware/Chip

DRP's Spatial-then-Temporal Processing Style
Lead me to the Structure-Oriented Computing Concept

Part 2: AI Computing
- Algorithm, Architecture, Real Chip

Tokyo Tech

ARTIC

AI's Energy Problem

AI Technology is Now Omnipresent in Our Society

Generative AI for Text/Image

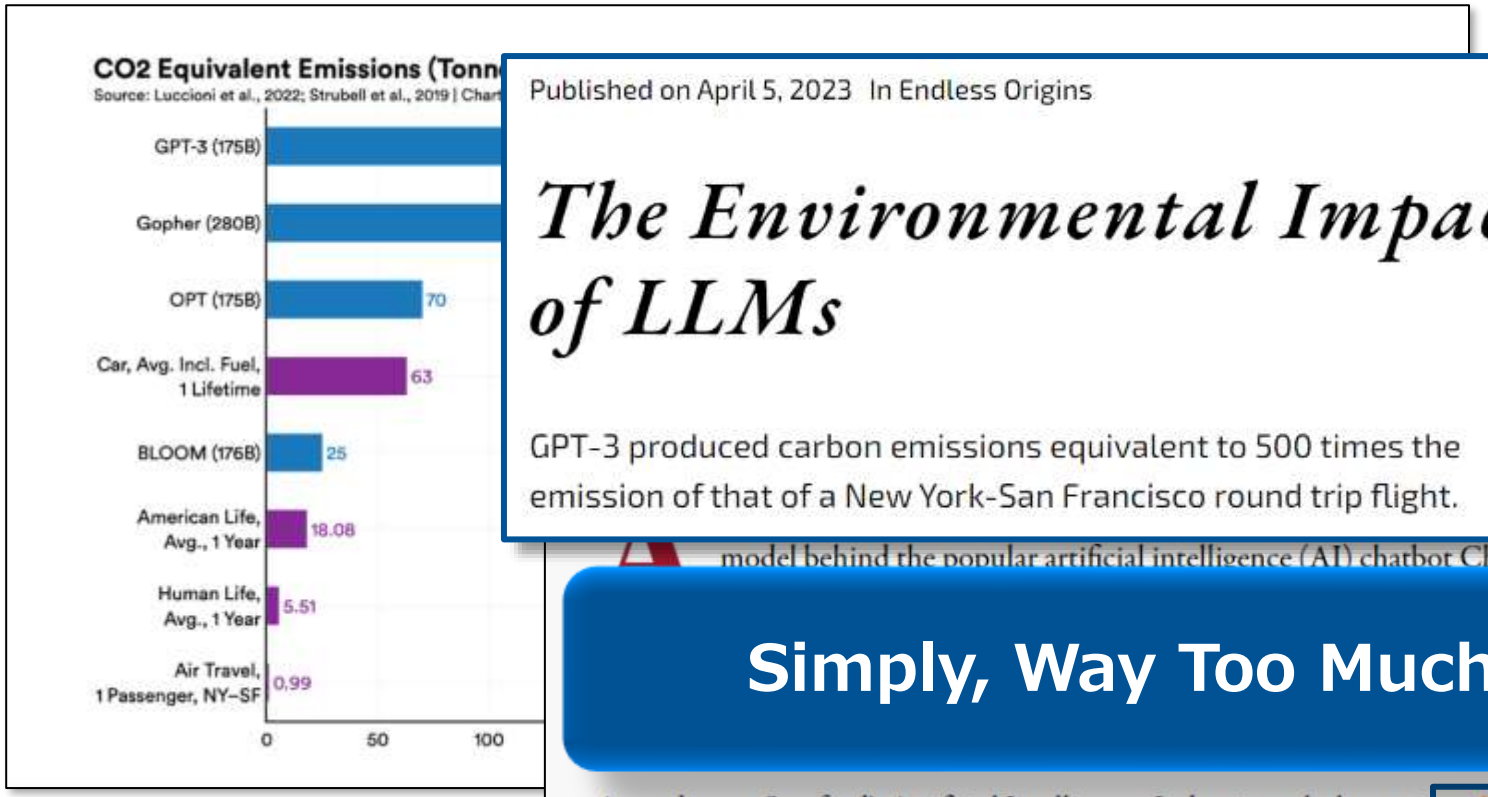
Smart Robotics

Autonomous Drones

Smart Social Infrastructure

- Serious Concern -
Its energy consumption and environmental impact

What Do We Know About It?



(AI Index Report 2023)

Published on April 5, 2023 In Endless Origins

The Environmental Impact of LLMs

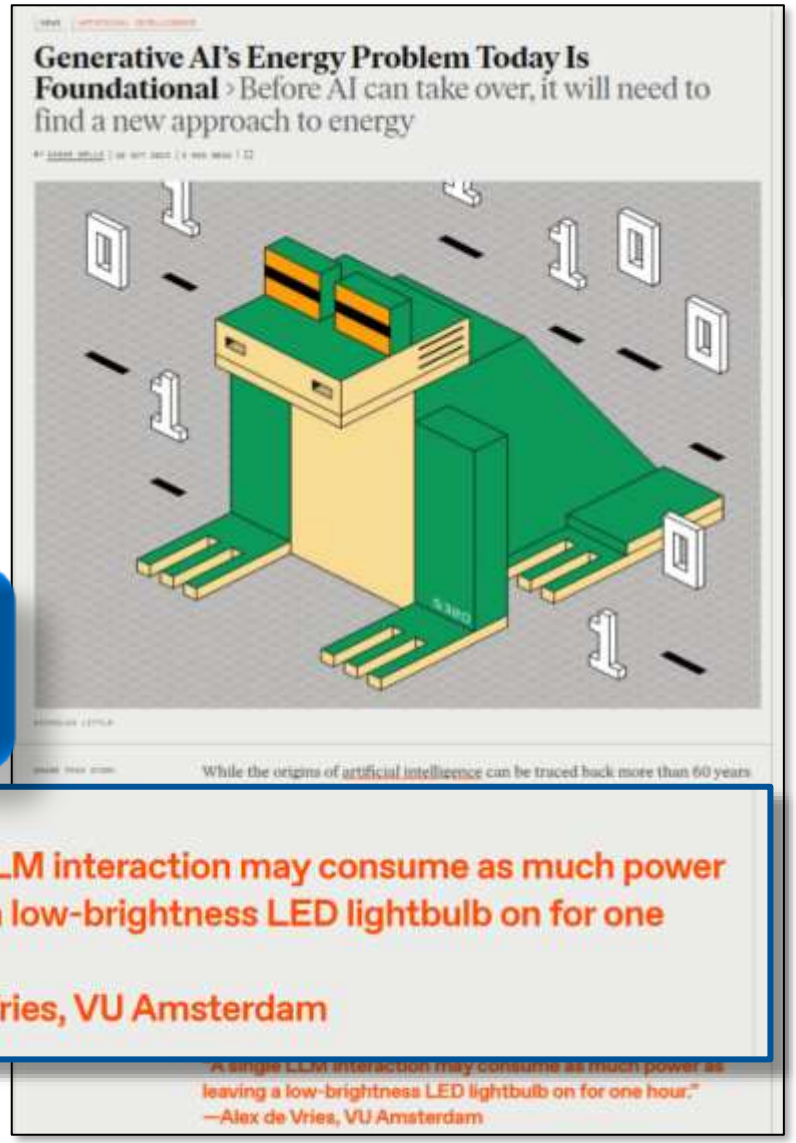
GPT-3 produced carbon emissions equivalent to 500 times the emission of that of a New York-San Francisco round trip flight.

Simply, Way Too Much

According to Stanford's Artificial Intelligence Index, it took the equiv

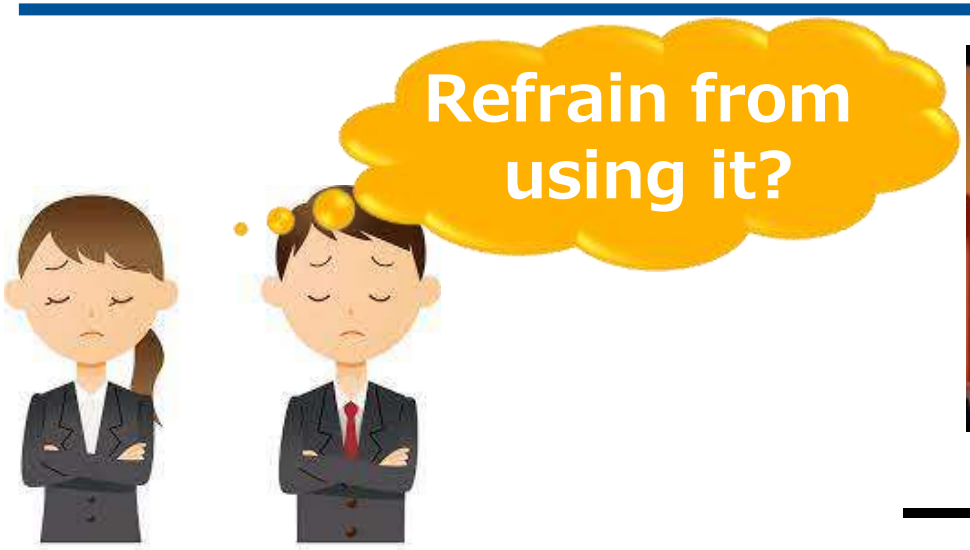
As the researchers calculated, this is the equivalent of the lifetime emissions of 8 cars — or 109 cars' yearly emissions — and enough energy to power an average U.S. home for over 120 years. Of the four models that the report scrutinized, GPT-3 released the most emissions and required the most power consumption.

(Stanford Report 2023)



(IEEE Spectrum 2023)

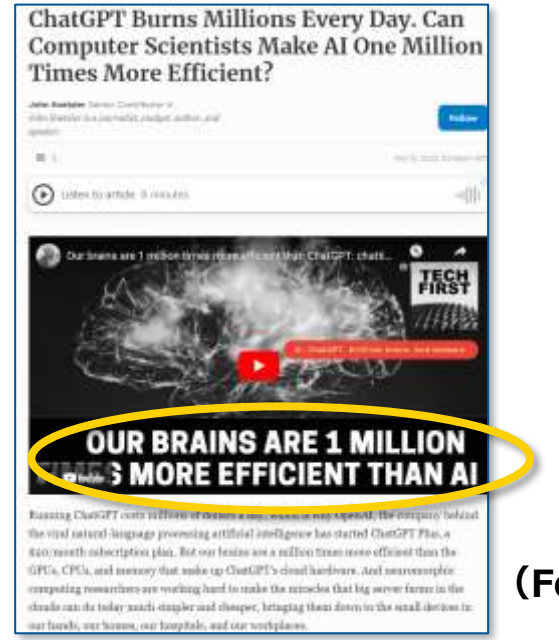
And … What We Can Do About It?



— It is already an unrealistic option —

Hence

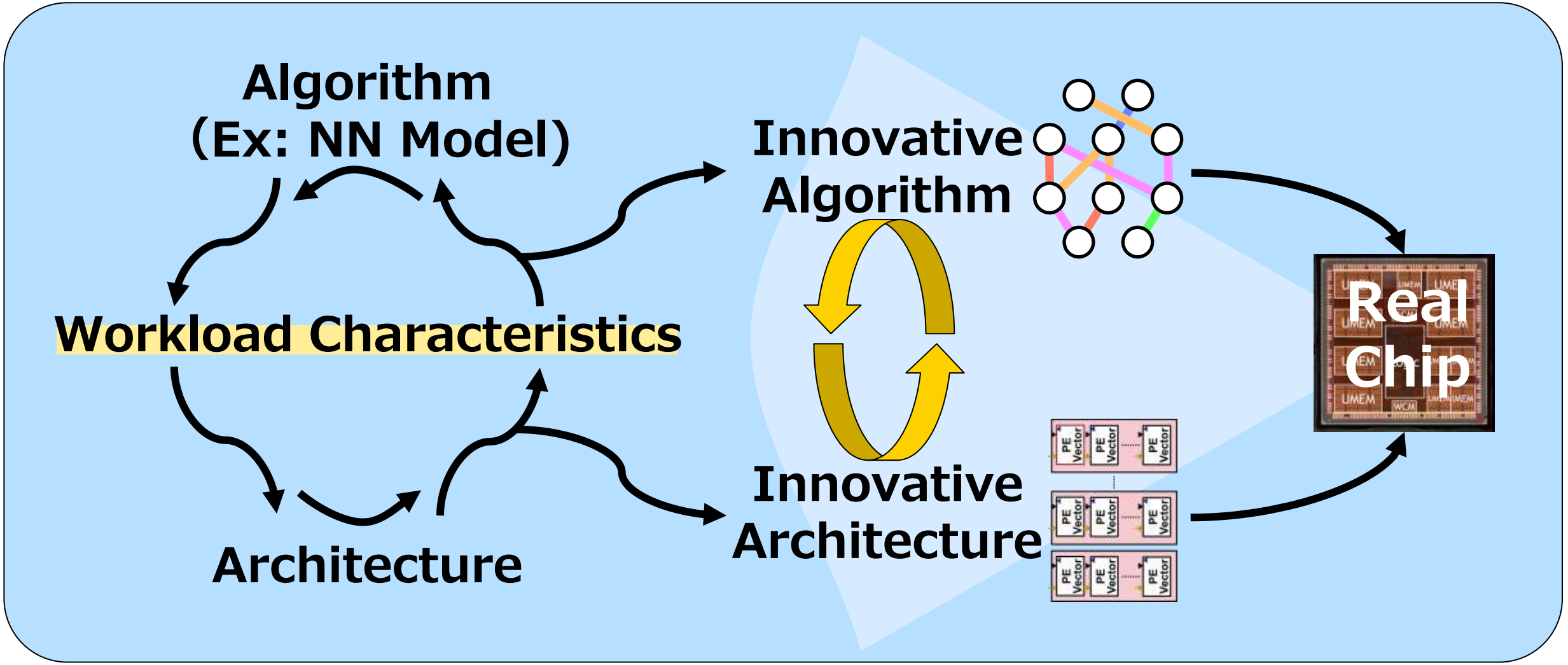
We Should Make AI Computing Several Orders of Magnitude More Energy Conscious



(Forbs 2023)

But, How?

Answer: Interplay Among Algorithm-Architecture-Real Chip



Observation: AI Computing Landscape

It is All About How to Handle Large-Volume Inputs and Outputs

Input

Computers

Output

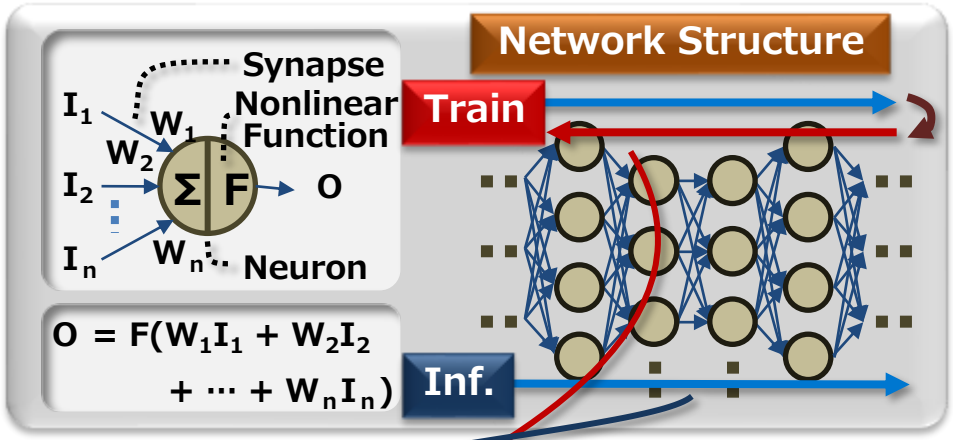
- Traditional ML
- Deep NeuralNets
- Reservoir Computing

- Discrete Optimization
- Annealing Computation
- Graph Processing

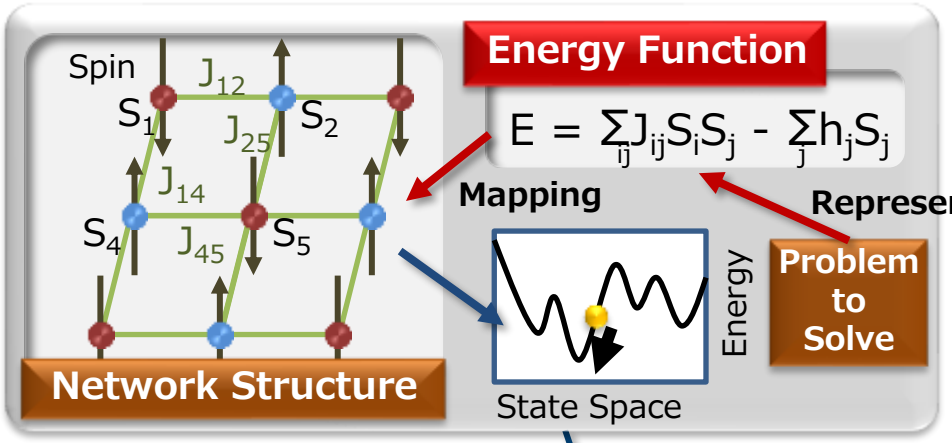
Classify Detect Recognize Predict Generate Recommend Decision Make

AI Computing: Driven by Energy Minimization Principle

Deep NeuralNets: DNNs

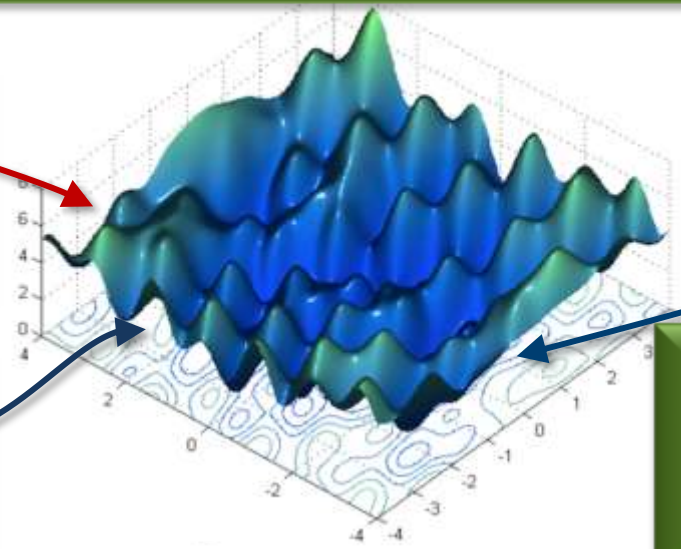


Annealing Computation



Energy Function Having Lots of Local Minimums

Design the Energy Funciton by Back Propagation



Traverse the Energy Surface and Find a Minimum

Traverse the Energy Surface and Find a Minimum

Our Goal: Establishing Common Parallel Architectural Ground for those workloads

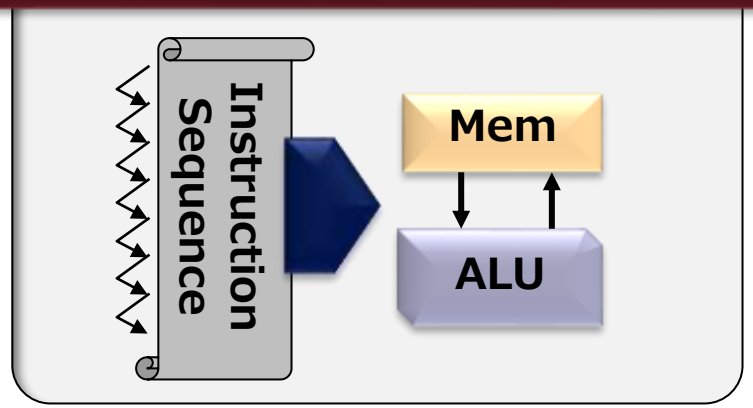
Architectural Shifts from Sequence to Structure

Conventinal: Sequence-Oriented

Control-Flow Processing
Von Neumann Processor

1949

- Program a Sequence
- Serial in its nature



Traditional Computing

Long Time Royal Road

Paradigm Shift

Augmenting Each Other

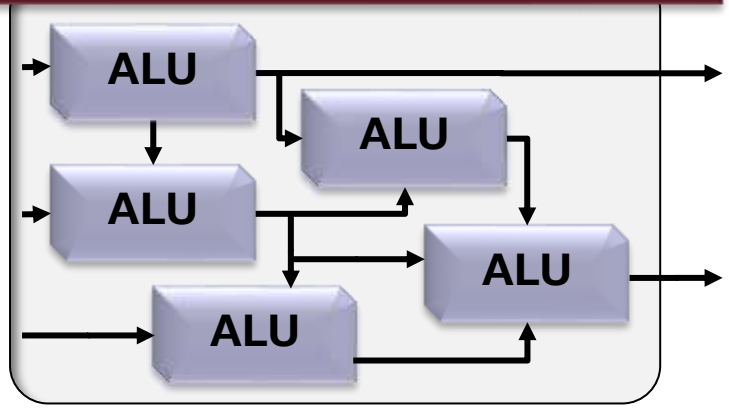
Newly Rising: Structure-Oriented

Data-Flow Processing
Reconfigurable HW

1946

Manual Re-wiring

- Program a Structure
- Parallel in its nature



Recent Evolution

AI Computing

Analogy: bit Dangerous yet Potentially Useful

Logical-Arithmetical Information Processing

Sequence-Oriented Computing is Efficient

Far beyond Energy Efficiency of human brain on such computing tasks

[Left Brain] Sequence-Oriented Engine

(Pinterest)

Intuitive-Spatial Information Processing

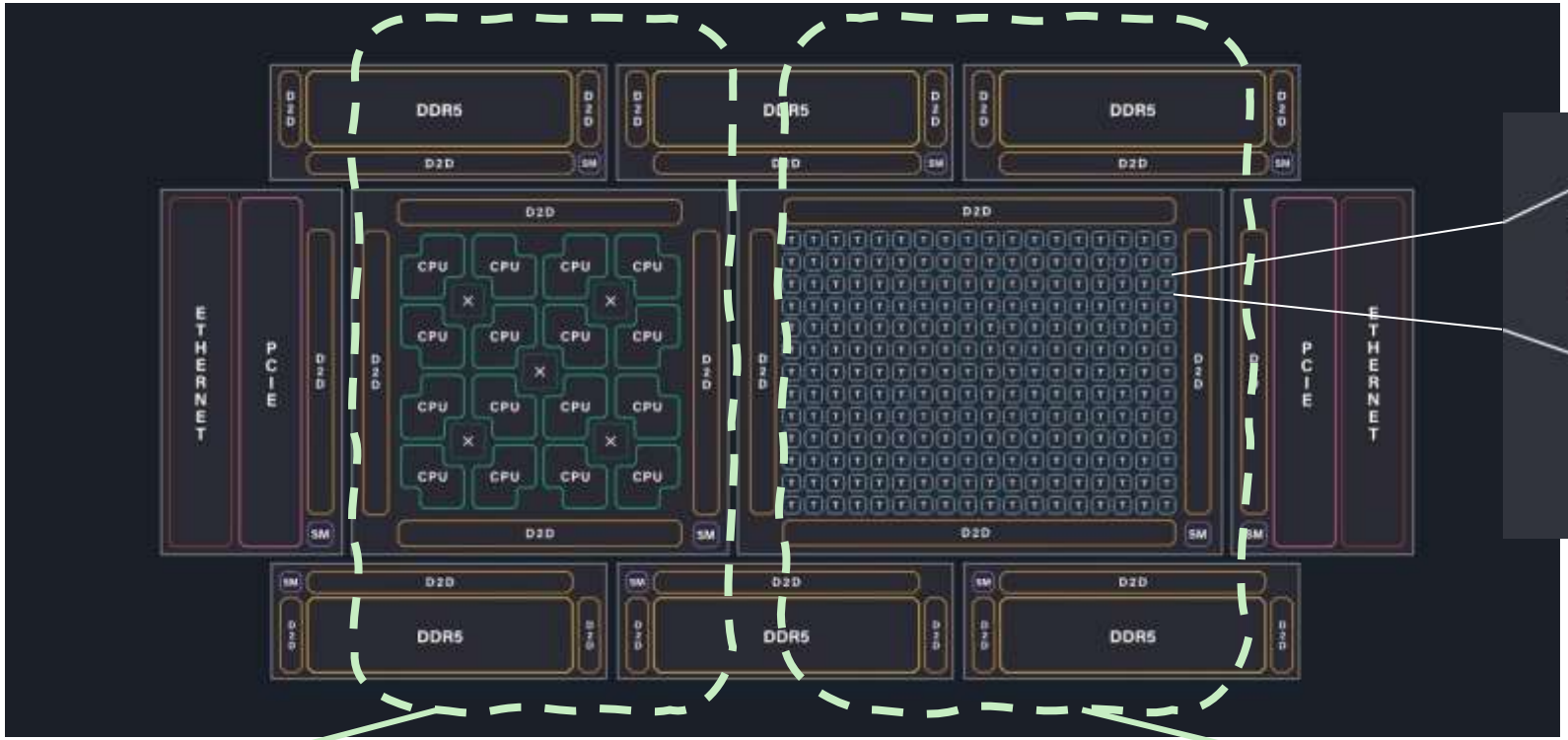
Structure-Oriented Computing is Efficient

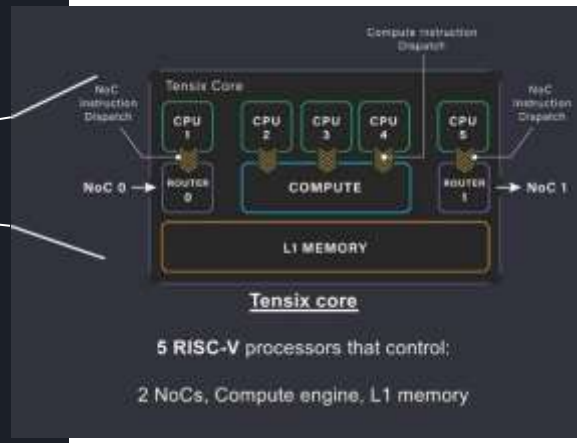
Can Exceed human brain's energy efficiency by architectural innovation

[Right Brain] Structure-Oriented Engine

Toward Robust and Efficient AI Computing Platform

Real World Example: Tenstorrent





[Left Brain]
Sequence-Oriented Engine

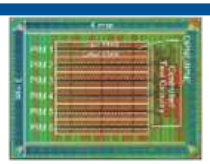
[Right Brain]
Structure-Oriented Engine

Mix of Sequence-Structure Strategy Depends on Each Architecture

Finding the best mix – on each side – is the heart of architecture design

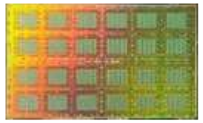
Showcase: AI Computing Chips of Our Own

- Binary/Ternary DNN Accelerator
  - Presented at the **VLSI Symposium 2017**



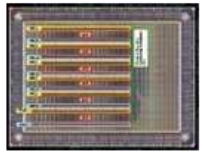
65nm

- Log-Quantized DNN Accelerator with 3D-Integrated SRAM
  - Presented at the **ISSCC 2018**



40nm

- Fully-Connected Fully-Parallel Digital Annealing Engine
  - Presented at the **ISSCC 2020**



65nm

- Shift-Oriented Cartesian-Product Array DNN Inference Accelerator
  - Presented at the **Hot Chips 2021**

40nm

- Fixed-Random-Weight DNN Inference Accelerator
  - Presented at the **ISSCC 2022**

40nm

- Metamorphic Annealing Engine for Fully-Connected Models
  - Presented at the **ISSCC 2023**

40nm

- Progressive-bitwidth DNN Inference Accelerator
  - Presented at the **VLSI Symposium 2023**

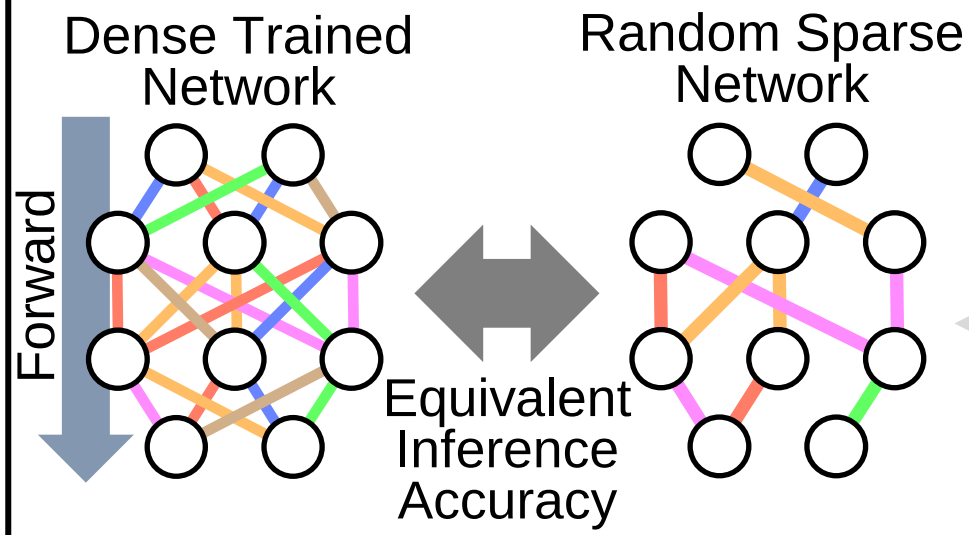
40nm

Lottery Ticket Hypothesis

Lottery Ticket Hypothesis

[J. Frankle+, ICLR 2019]

Existence of subnetworks



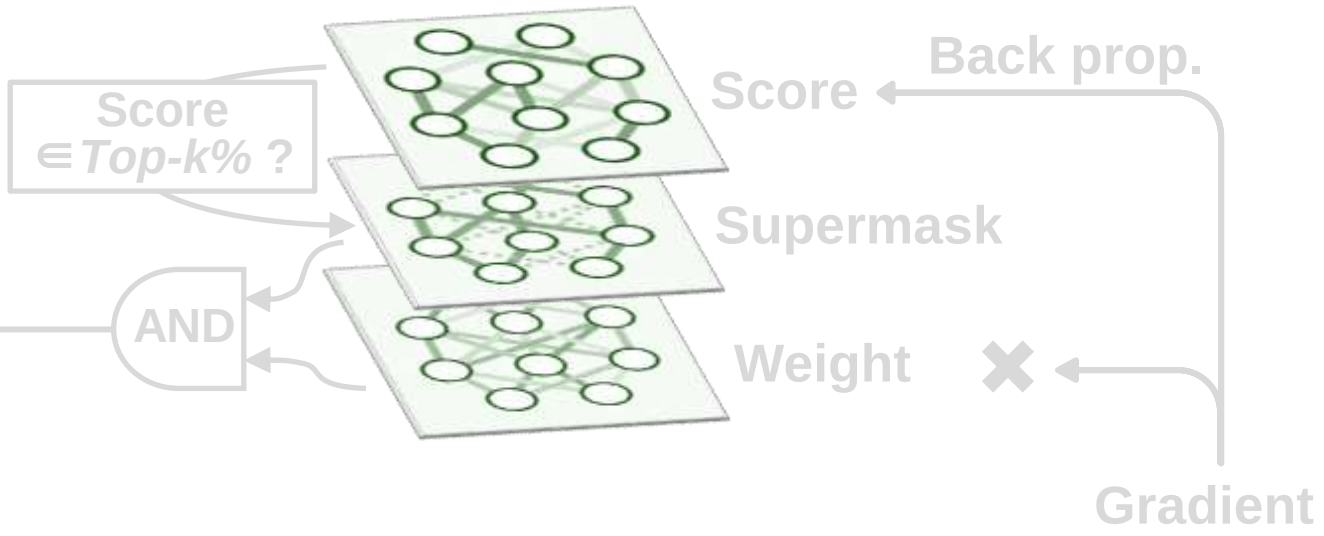
How do we find?

Hidden Network (HNN)

[V. Ramanujan+, CVPR2020]

Algorithm to find a subnetwork!

Edge-popup algorithm



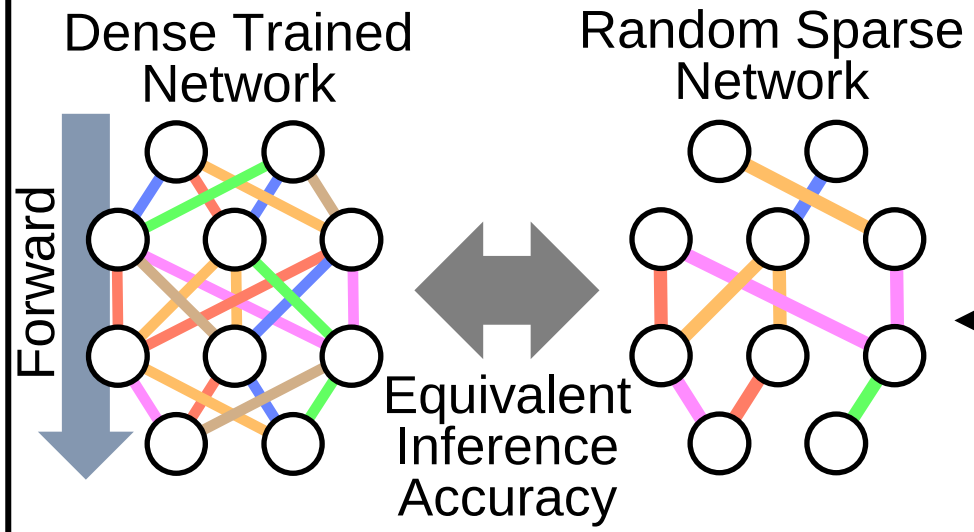
Not update weights but scores

Hidden Networks (HNNs): Strong Lottery Ticket Theory

Lottery Ticket Hypothesis

[J. Frankle+, ICLR 2019]

Existence of subnetworks



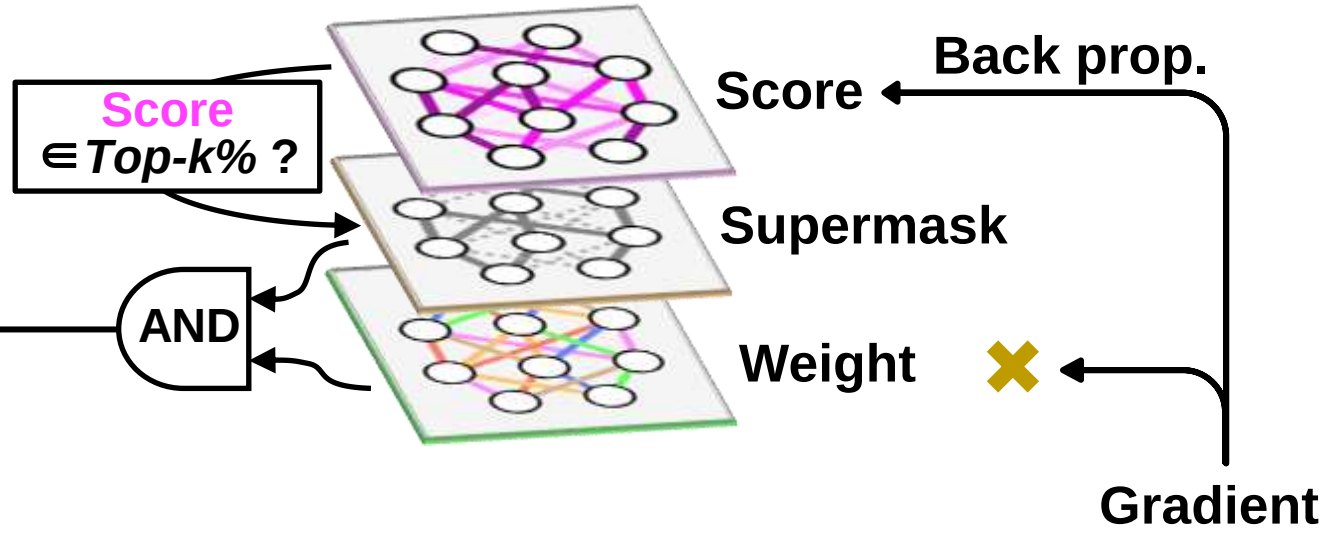
How do we find?

Hidden Network (HNN)

[V. Ramanujan+, CVPR2020]

Algorithm to find a subnetwork!

Edge-popup algorithm

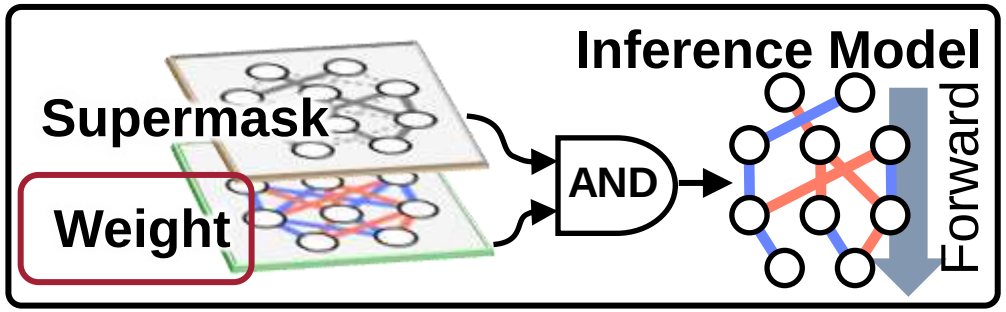


Not update weights but <u>scores</u>

HNN Utilizes Fixed Random Weights

- **Fixed at initial random numbers**

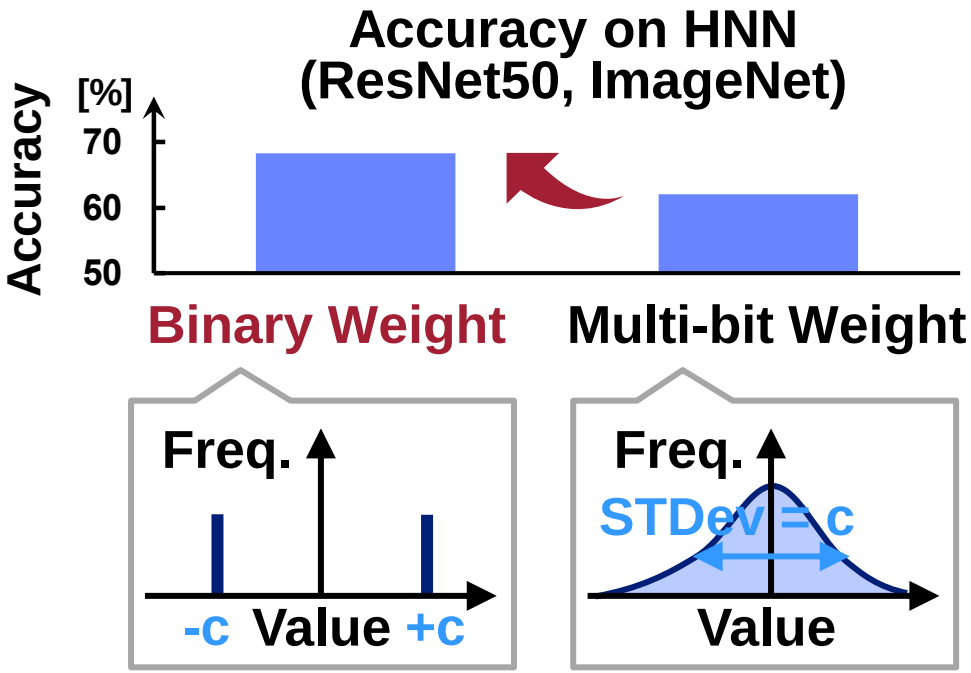
☺ **Weights are no longer variables but are (random) constants**



- **Binary weights {-1, +1} show better accuracy than multi-bit weights**

[V. Ramanujan+, CVPR2020]

☺ **Enhance computation efficiency**

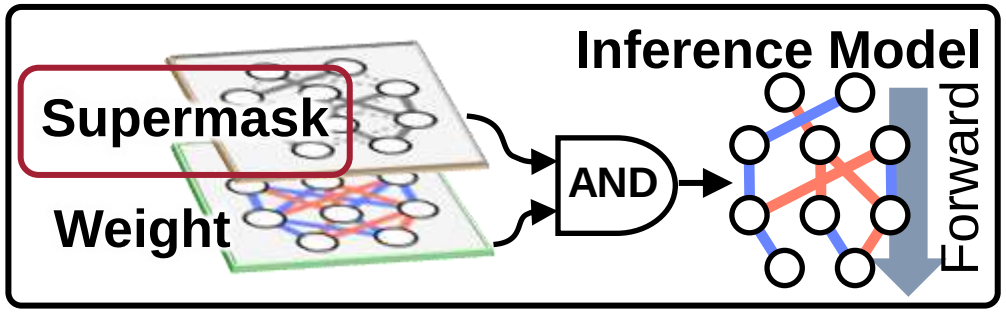


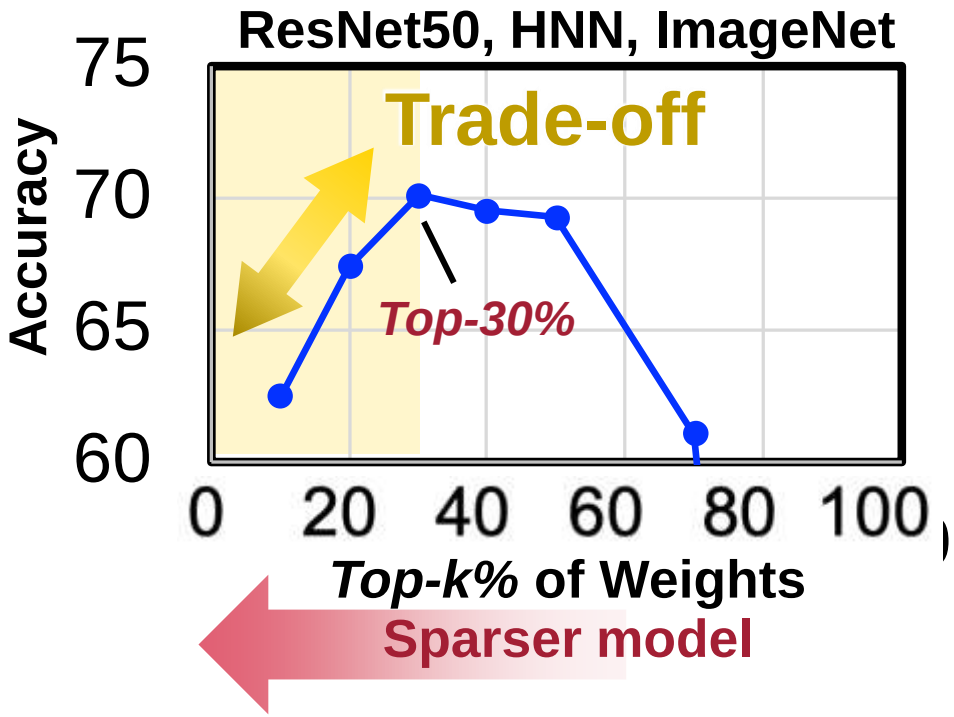
HNN Needs a Supermask

- **A supermask is binary {0, 1} information for selecting connections**

☹ **Conventional NNs do not need supermask**

☺ **A supermask provides the trade-off between accuracy and sparsity**



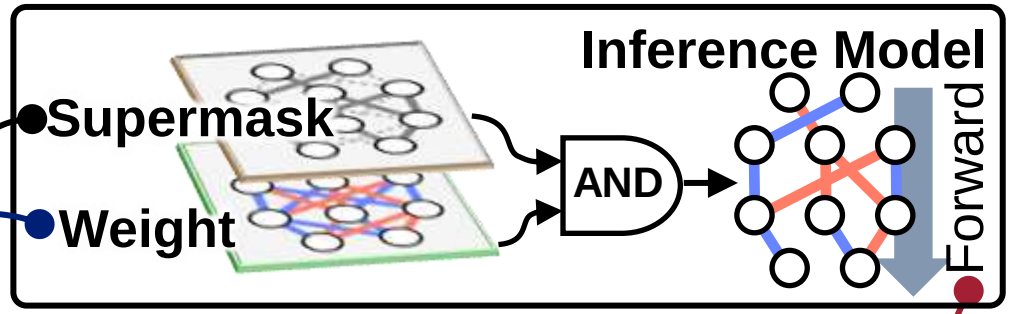


Key Contributions of This Work

The first HNN inference chip, *Hiddenite*:

Hidden Network Inference Tensor Engine

[*]

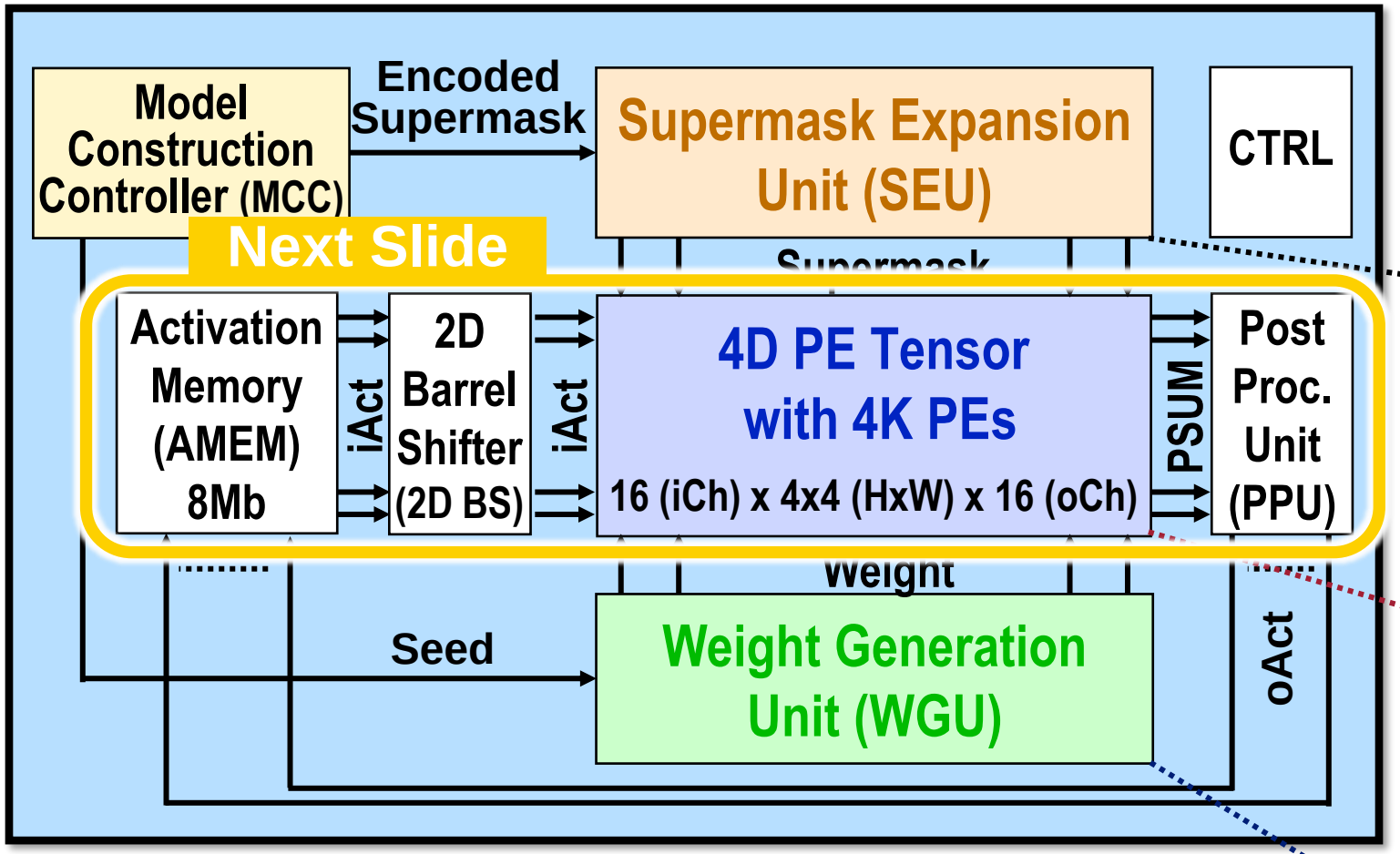


1. **On-chip weight generation**
   - eliminates the need for storing and loading weights
2. **On-chip supermask expansion**
   - reduces the model parameters to load
3. **A high-density 4D parallel processor**
   - improves efficiency by maximizing data re-use

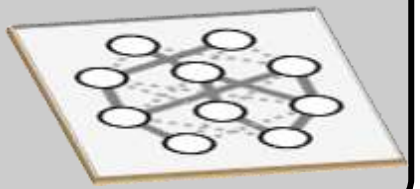
→ **On-chip model construction**

[*] https://en.wikipedia.org/wiki/Hiddenite

Overall Chip Architecture

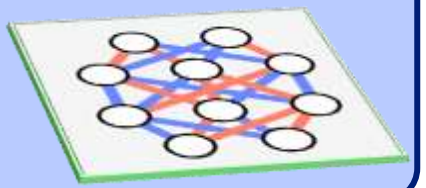


2. ZRL-Encoded Supermask → Decoded Supermask

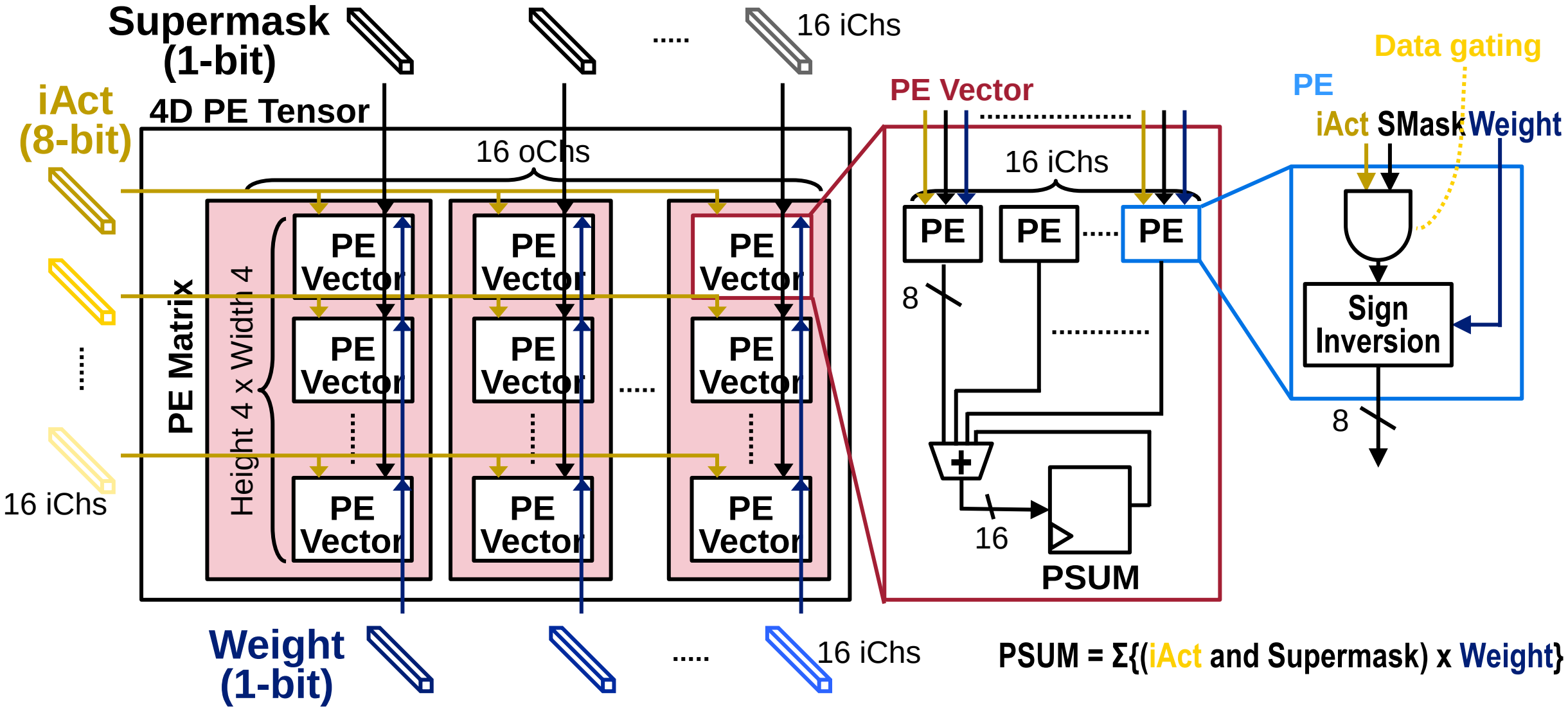


3. PSUM
= Σ{(iAct and Supermask) x Weight}
Binary → PE is so tiny!

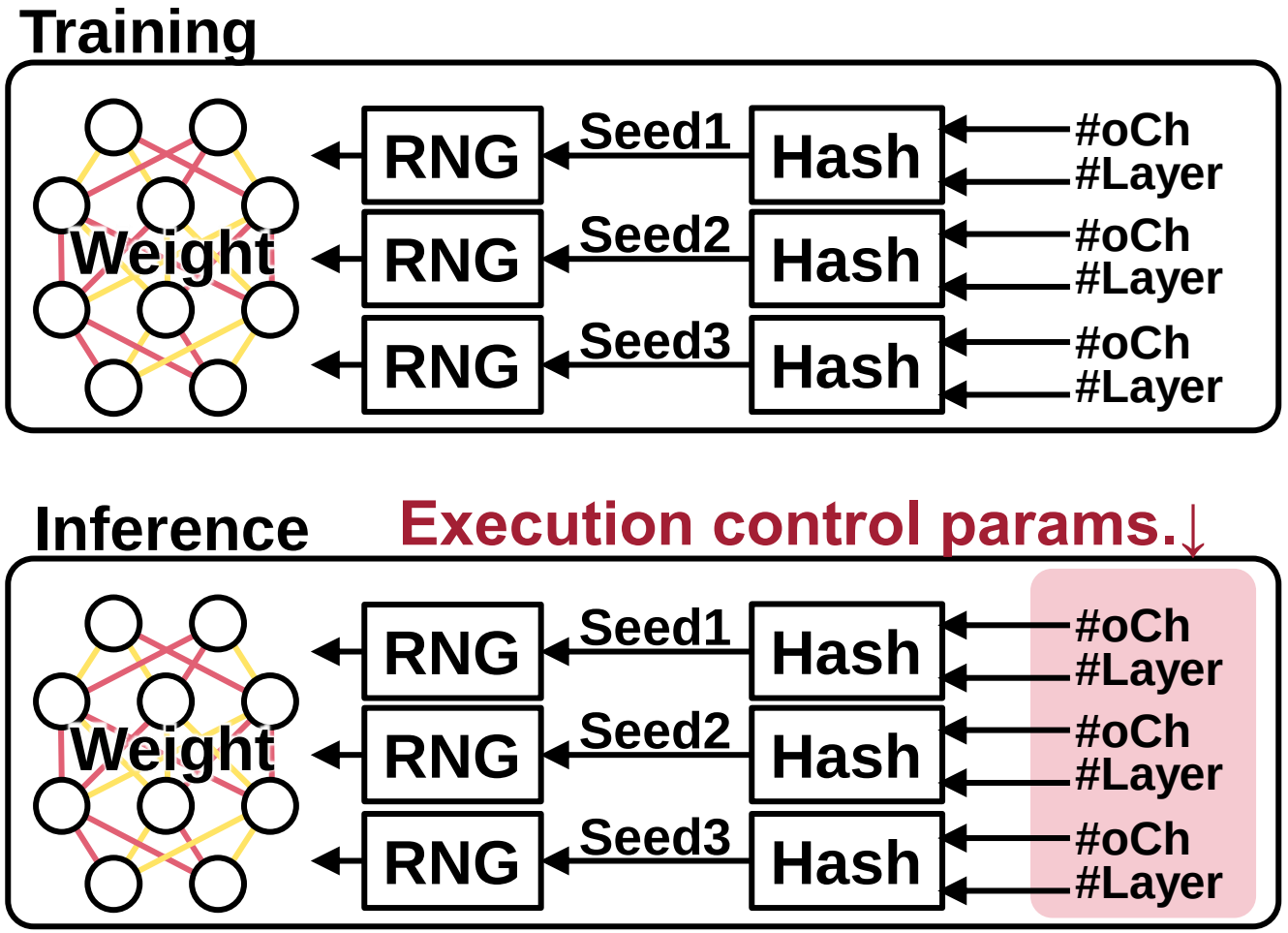
1. Seed → RNG → Weight

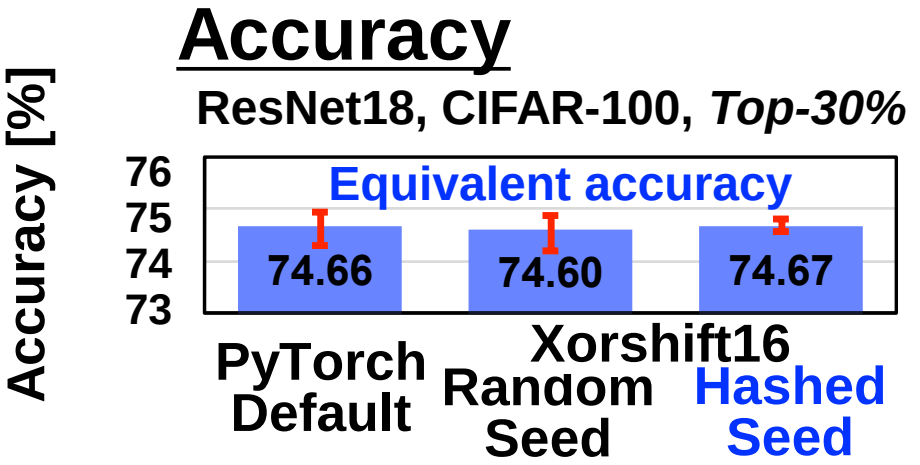


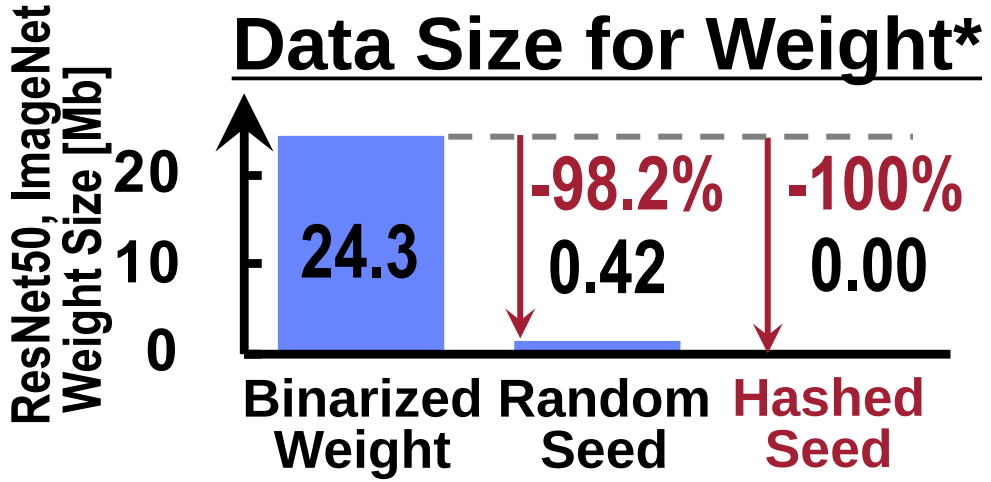
4D PE Tensor: Dataflow



Generating RNG Seeds by Hashing



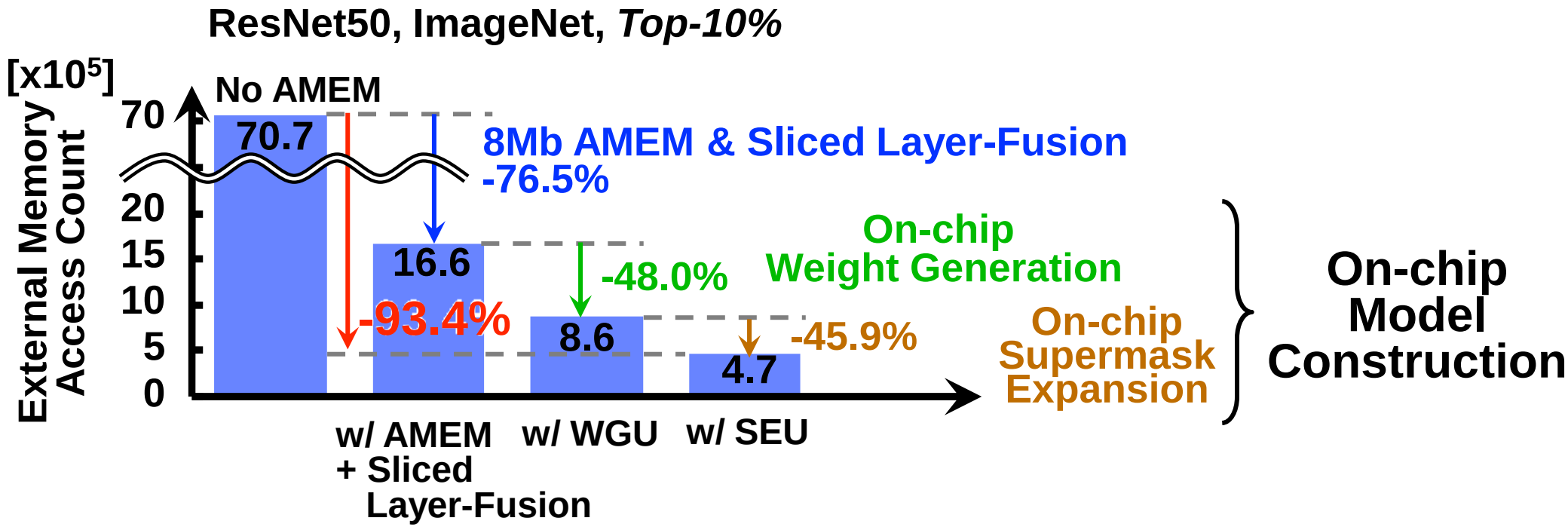




* Supermask will be explained later

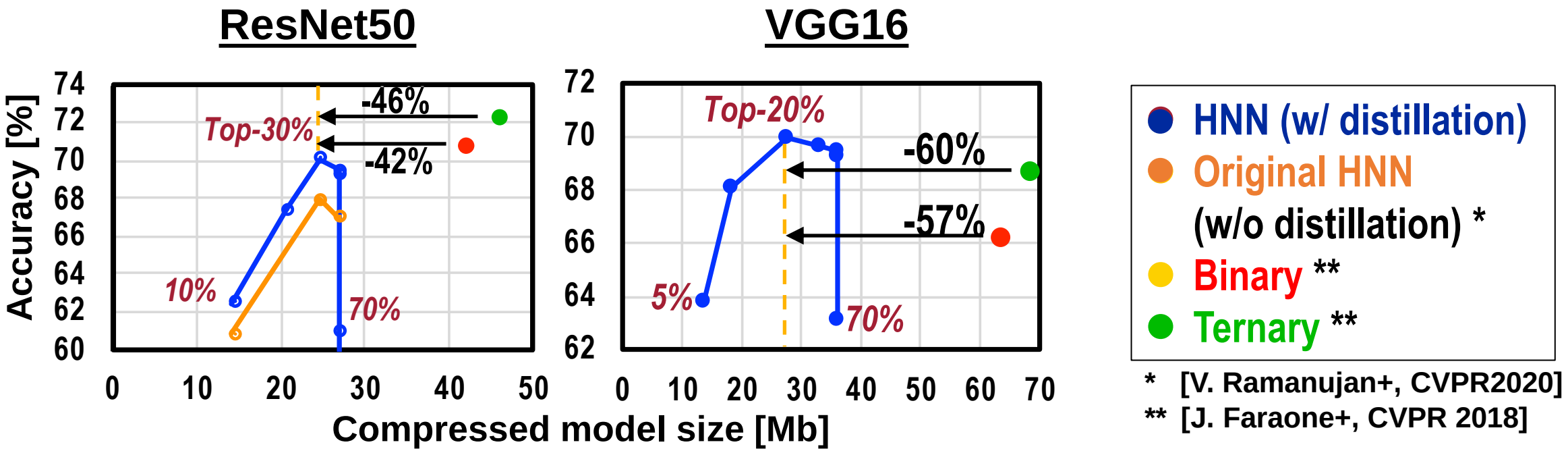
- **Hashed seeds eliminate the need to store weights without accuracy degradation**

Total External Memory Access Reduction



- Hiddenite drastically reduces power-consuming external memory accesses

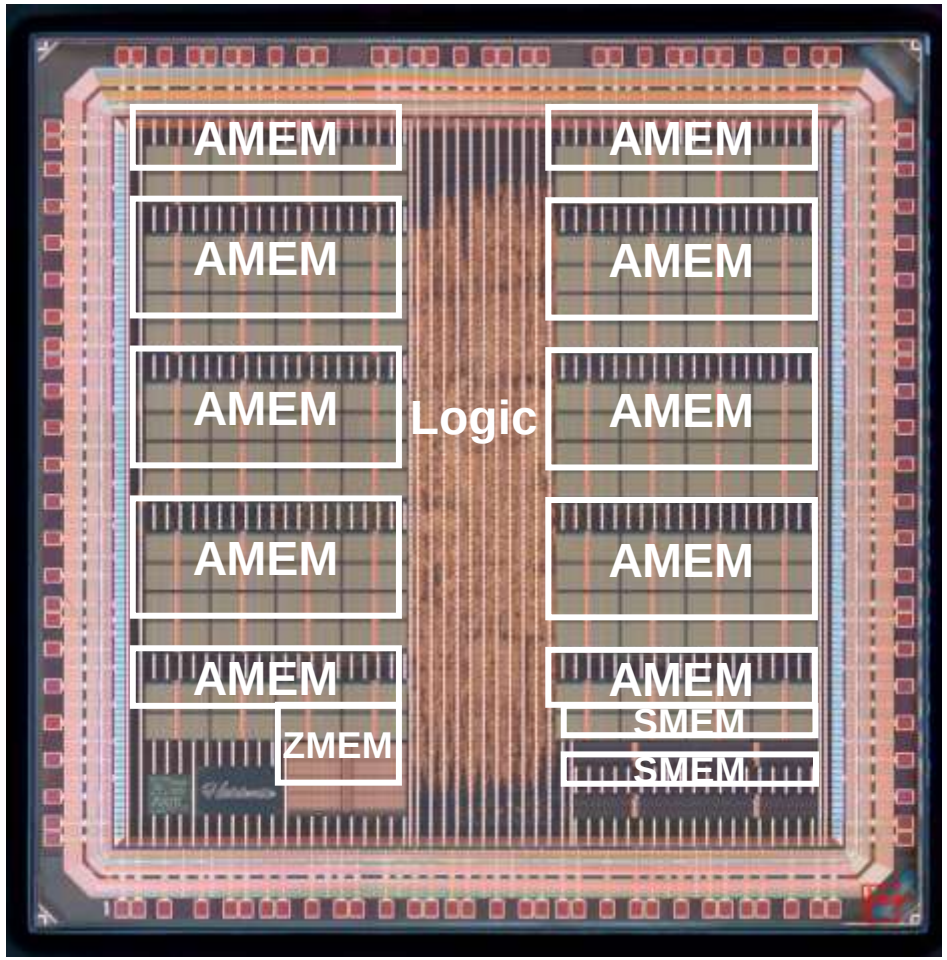
Accuracy vs. Model Size on ImageNet



- Comparable or better accuracies
- Smaller model size than binary model

Hiddenite Chip Summary

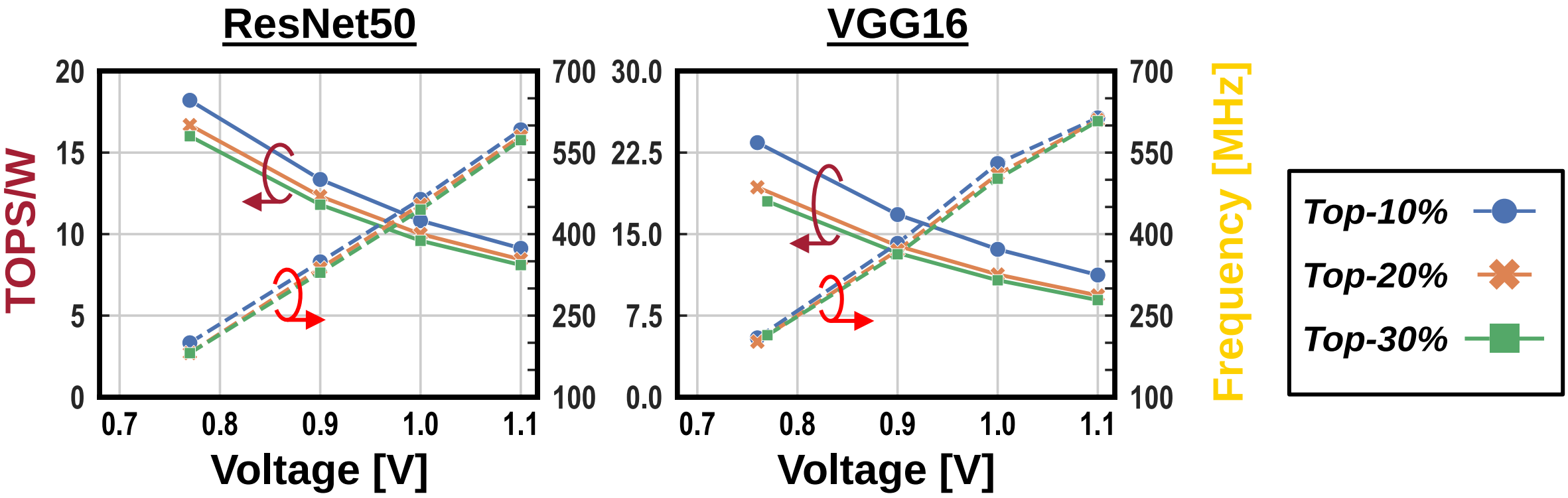
Micrograph



Specification Table

Technology	TSMC 40nm CMOS (LP)
Package	QFN80 (48 Signal Pins)
Chip Size	3mm x 3mm
Core Area	SRAM: 3.78mm^2 Logic: 0.58mm^2 Total: 4.36mm^2
Core V_{DD}	0.8-1.1V
I/O V_{DD}	3.3V
Gate Count	746K Gates
SRAM	AMEM: 8Mb SMEM: 256kb ZMEM: 128kb Total : 8.375Mb

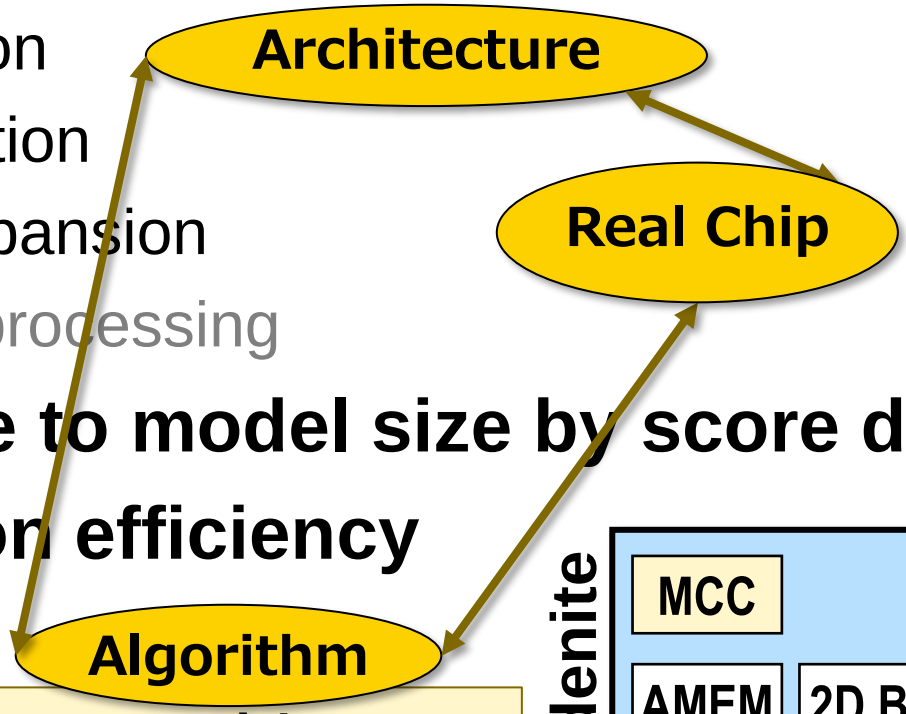
Measured Results on ImageNet

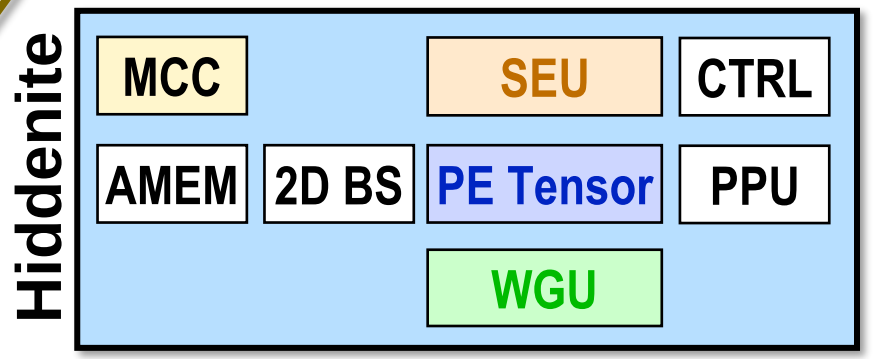


- Efficiency on ResNet50: 18.2-to-16.0TOPS/W at 0.77V
- Maximum frequencies: 614-to-573MHz at 1.1V

What Hiddenite Has Achieved ?

- **Hiddenite is the first HNN inference chip**
- **Drastically reduce external memory access by**
  - On-chip model construction
    - On-chip weight generation
    - On-chip supermask expansion
  - Slice-based layer-fusion processing
- **SOTA accuracy relative to model size by score distillation**
- **SOTA-level computation efficiency**



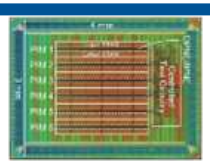


We also presented a new Strong Lottery Ticket training algorithm at ICML 2022.

"Multicoated Supermasks Enhance Hidden Networks"

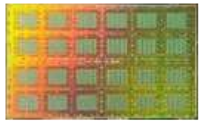
AI Computing Chips from Our Group

- Binary/Ternary DNN Accelerator
  - Presented at the **VLSI Symposium 2017**



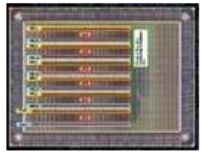
65nm

- Log-Quantized DNN Accelerator with 3D-Integrated SRAM
  - Presented at the **ISSCC 2018**



40nm

- Fully-Connected Fully-Parallel Digital Annealing Engine
  - Presented at the **ISSCC 2020**



65nm

- Shift-Oriented Cartesian-Product Array DNN Inference Accelerator
  - Presented at the **Hot Chips 2021**

40nm

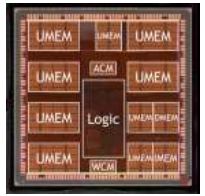
- Fixed-Random-Weight DNN Inference Accelerator
  - Presented at the **ISSCC 2022**

40nm

- Metamorphic Annealing Engine for Fully-Connected Models
  - Presented at the **ISSCC 2023**

40nm

- Progressive-bitwidth DNN Inference Accelerator
  - Will present at the **VLSI Symposium 2023**



40nm

Combinatorial Optimization Appears Everywhere

Finance

Portfolio Optimization

Logistics

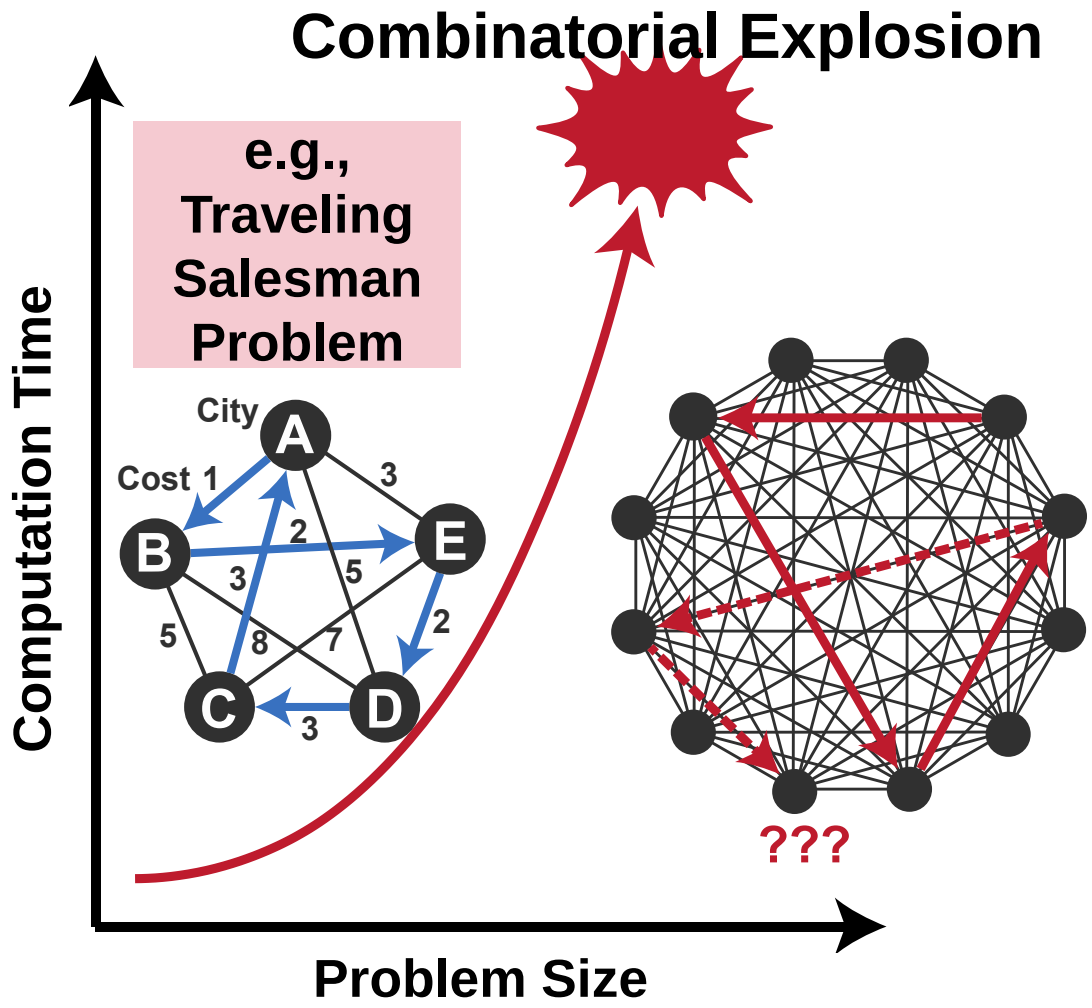
Traveling Salesman

Drug Discovery

Graph Mining

Machine Learning

Hyperparameter Opt.



NP Hard: Notoriously Difficult for Present Computers

Annealing: Optimization based on Ising Models (Inspired by Solid-State Physics)

Combinatorial Optimization Problem

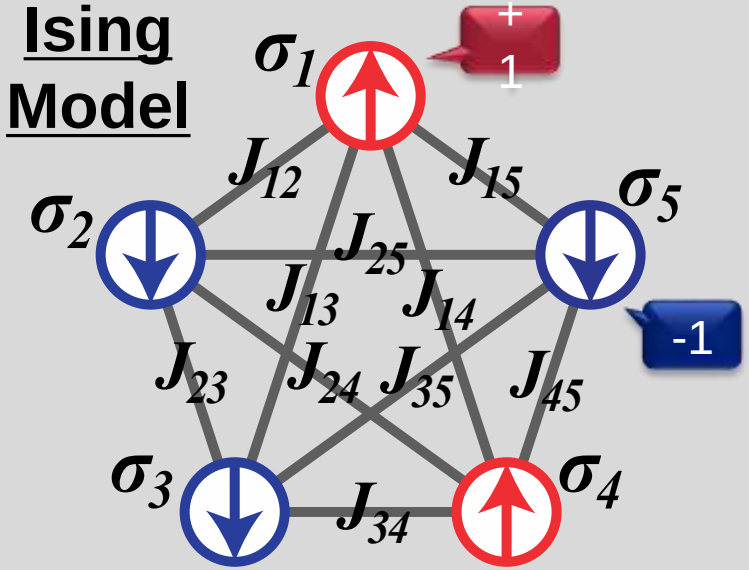
Input : J

Output : σ

Solution

Finding Ground-States (i.e., Minimum Energy) of (Potentially) Fully-Connected Ising Models

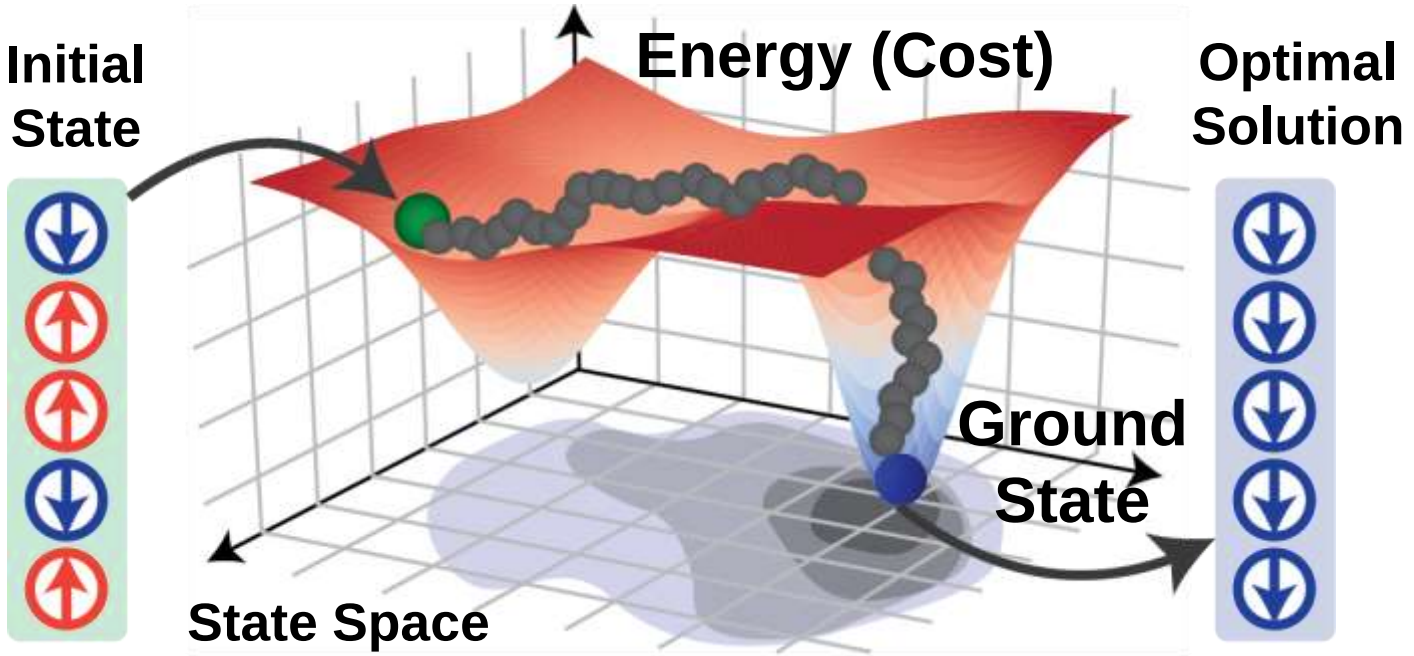
Ising Model



σ_x : Spin (Binary Value)

J_{xy} : Interaction Weight

Annealing Processor

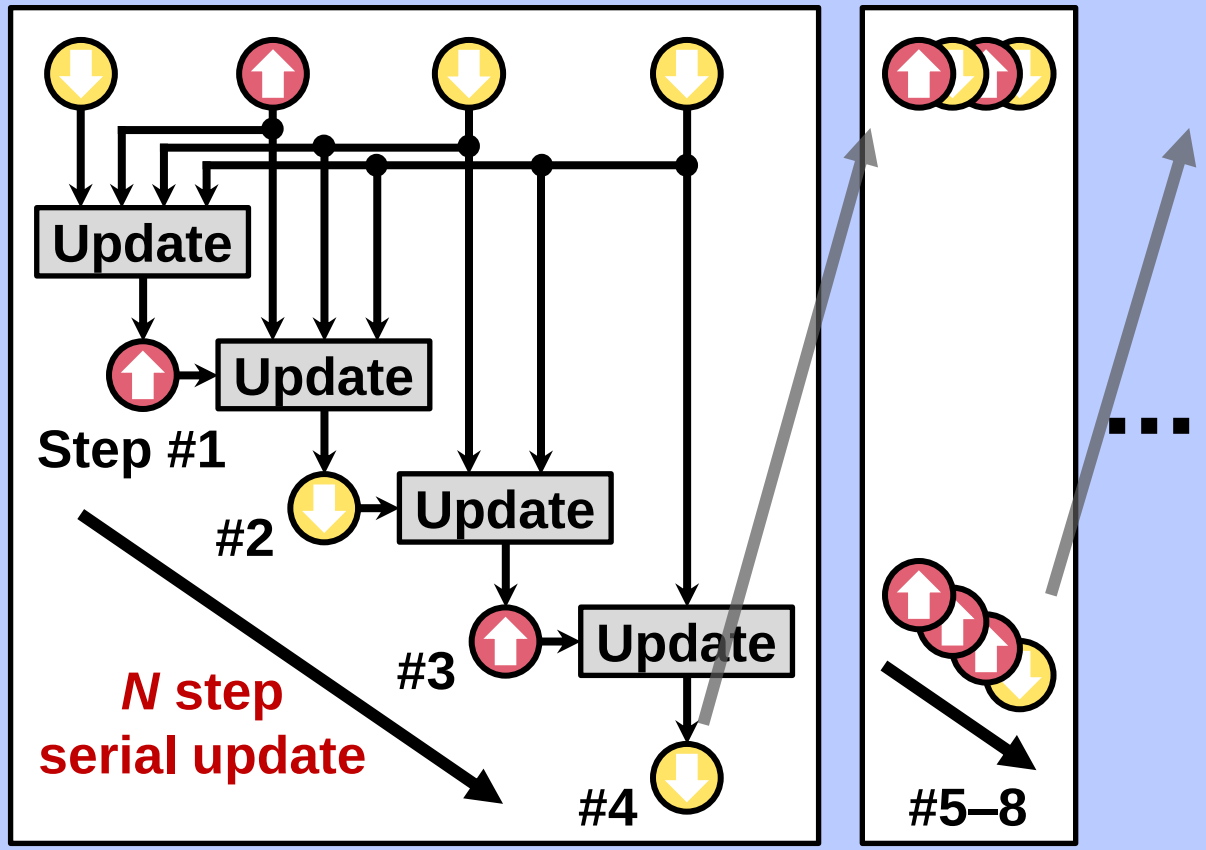


Serial and Parallel Annealing Policies

***N*: #Spins**

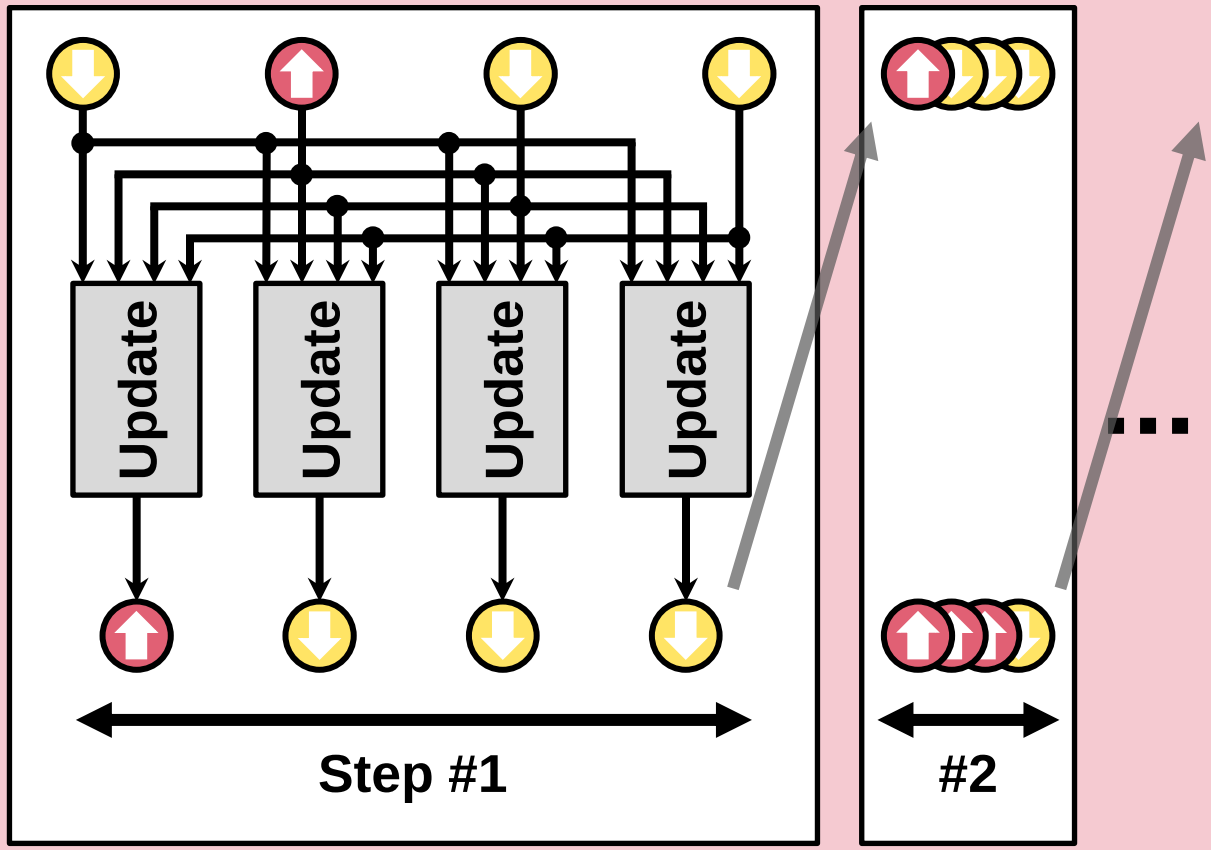
Traditional Method

SA: Simulated Annealing



Our Proposal (ISSCC 2020)

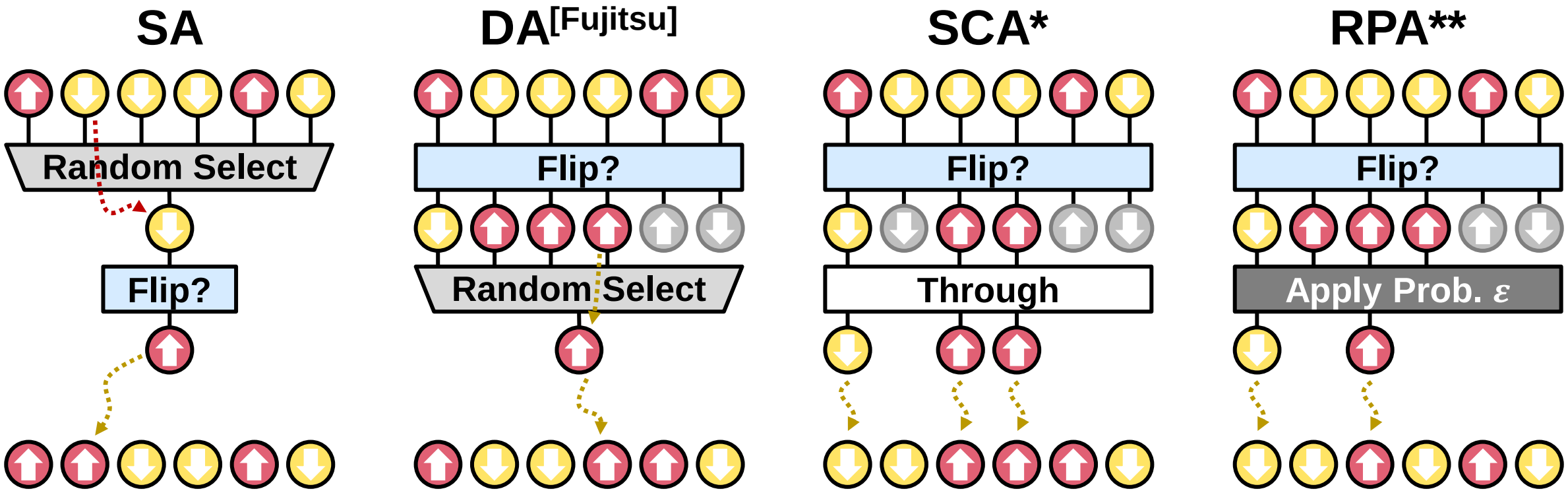
SCA: Stochastic Cellular Automata Annealing



- SCA[6] fromcan realize *O(N)* times faster spin update than SA

Comparison of Annealing Algorithms

Single Flip Trial	Parallel Flip Trial
Serial Spin Update	Parallel Spin Update

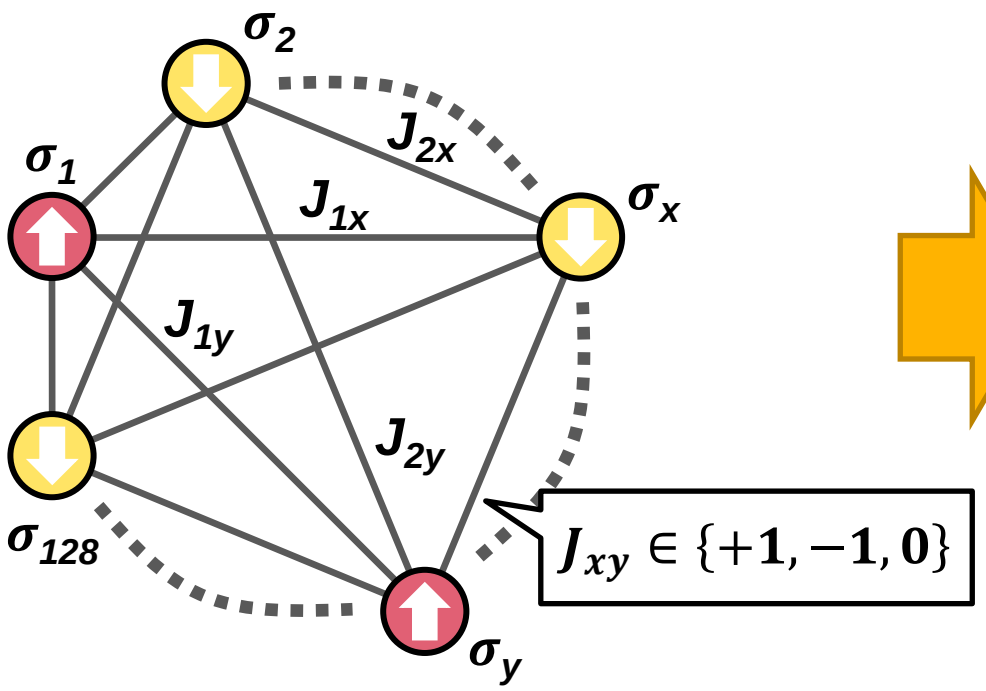


* Stochastic Cellular Automata Annealing
** Ratio-controlled Parallel Annealing

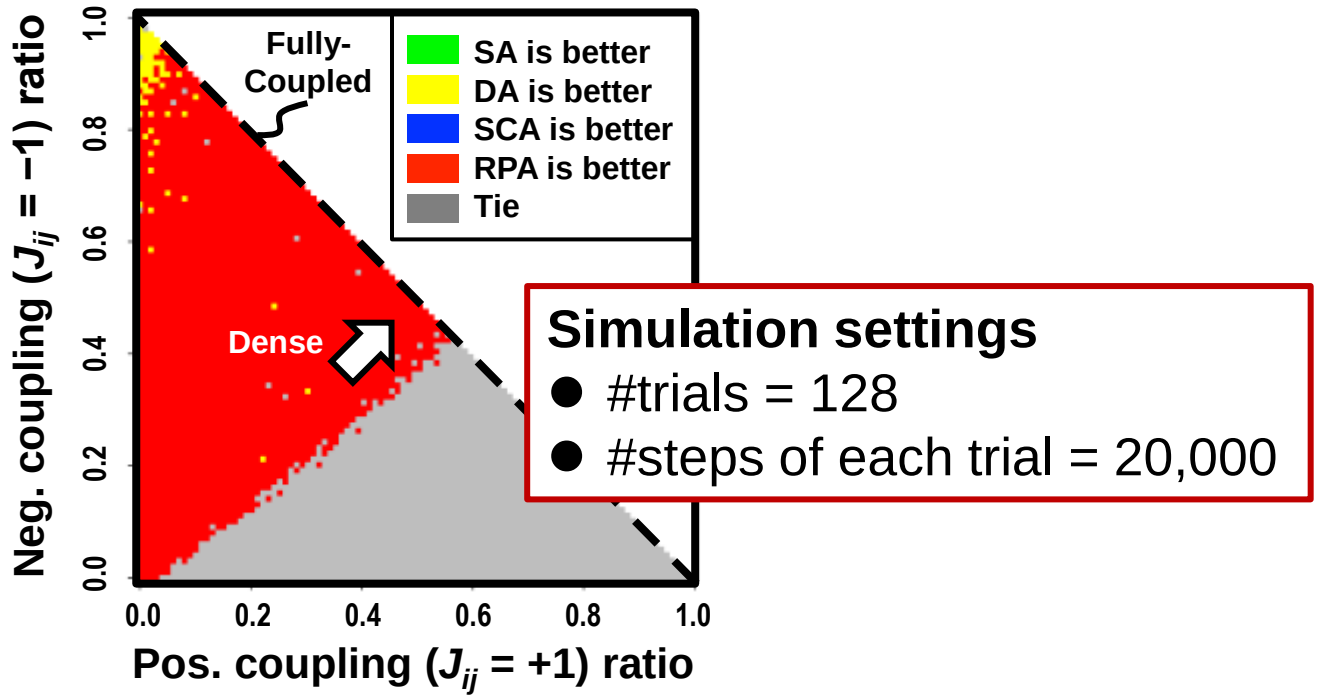
2020 — 2023
Our Proposal

Motivation for Applying Multi-Annealing Algorithms

Example: 128-spin Ising models



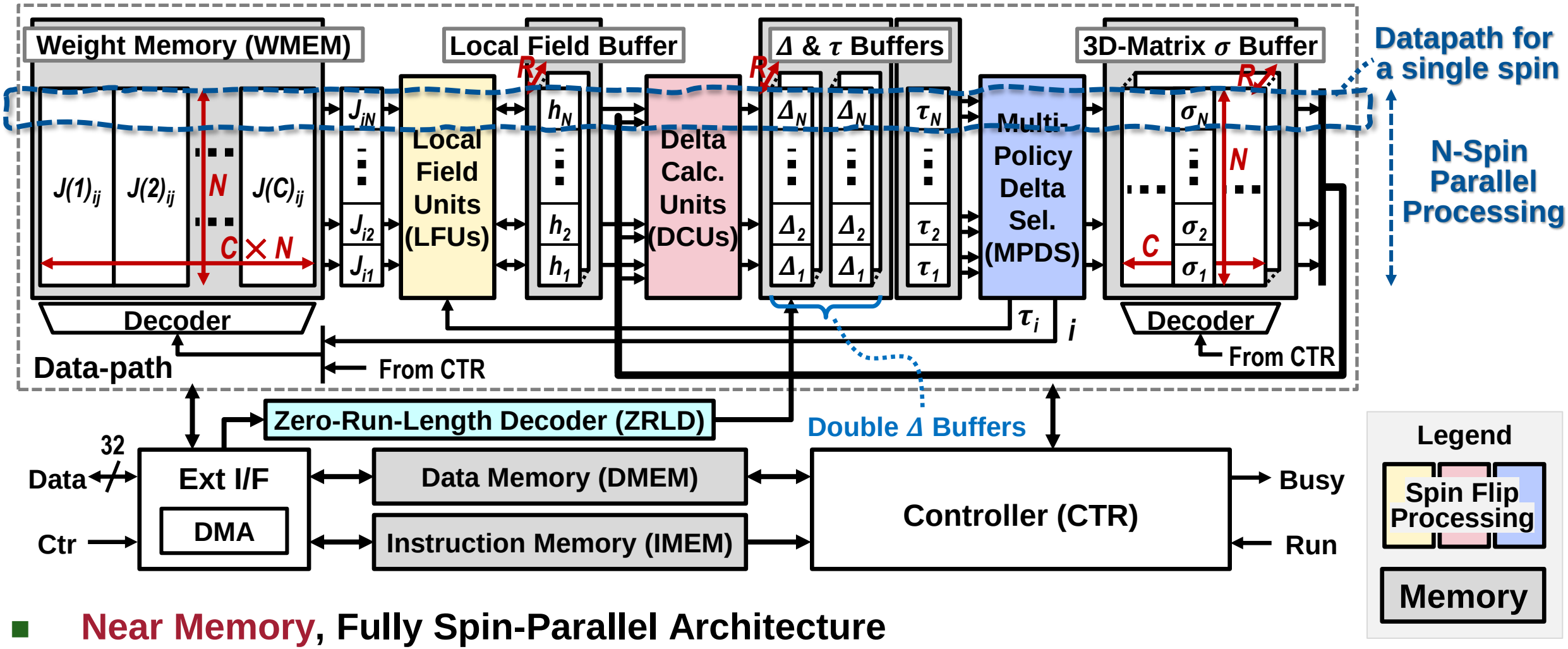
Compare avg. Ising energies of SA, DA, SCA, and RPA



Simulation settings
- #trials = 128
- #steps of each trial = 20,000

- **Optimal policy depends on the Ising model (i.e., Problem to solve)**
  - **RPA works better for the most cases**
  - **DA is better for Ising models having many negative couplings**

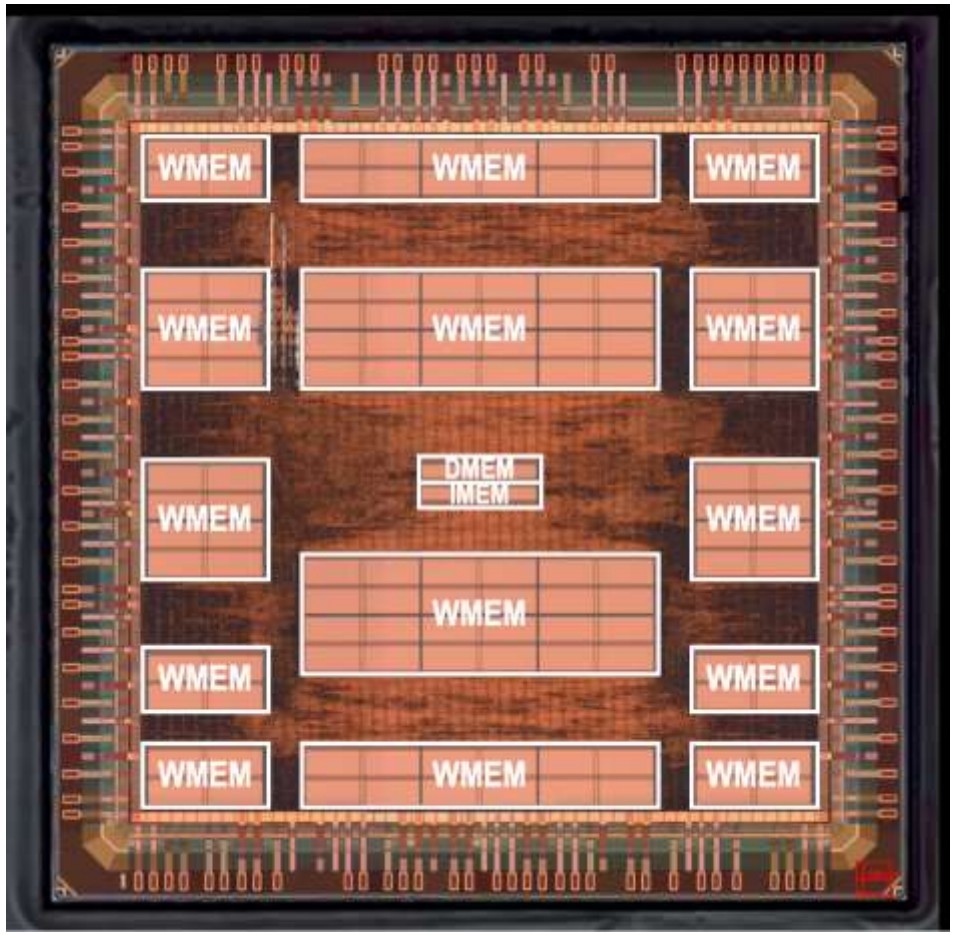
Amorphica: Metamorphic Annealing Architecture



- Near Memory, Fully Spin-Parallel Architecture
- SA/DA/SCA/RPA algorithms are applied with dynamic reconfigurability
- Very close to what Binary Neural Network (BNN) Inference Chip looks

Amorphica Chip Summary

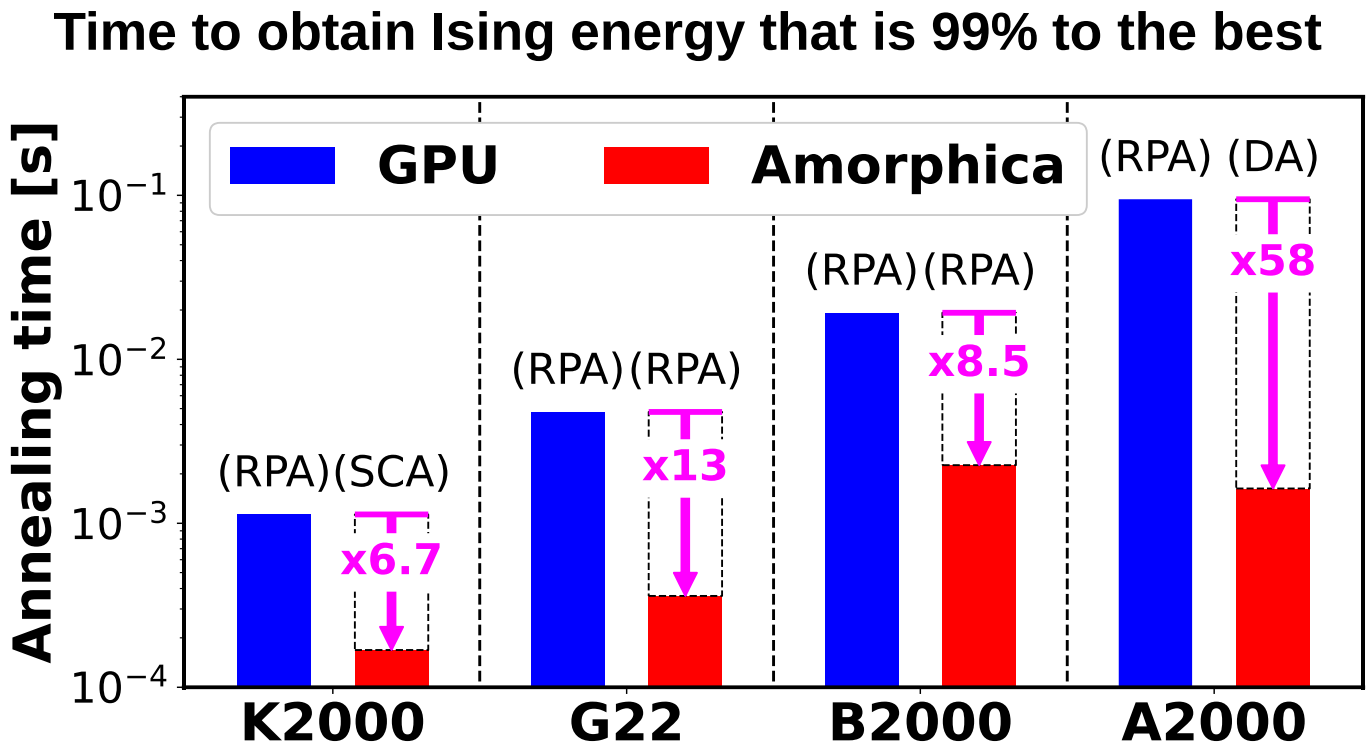
Micrograph



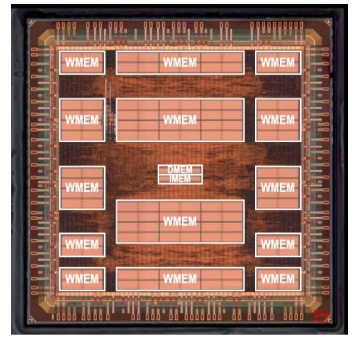
Specification Table

Technology	TSMC 40nm CMOS (LP)
Package	QFN80
Chip Size	3mm x 3mm
Core Area	SRAM: 3.55mm^2 Logic: 1.48mm^2
Core V_{DD}	0.8-1.1V
I/O V_{DD}	3.3V
Max Freqency	336MHz@1.1V 134MHz@0.8V
Gate Count	1.2M Gates
SRAM	WMEM: 8Mb DMEM: 64Kb IMEM: 64Kb Total: 8.125Mb

Comparison to GPU (Nvidia RTX2080-Ti)



≈ 250W



< 500mW

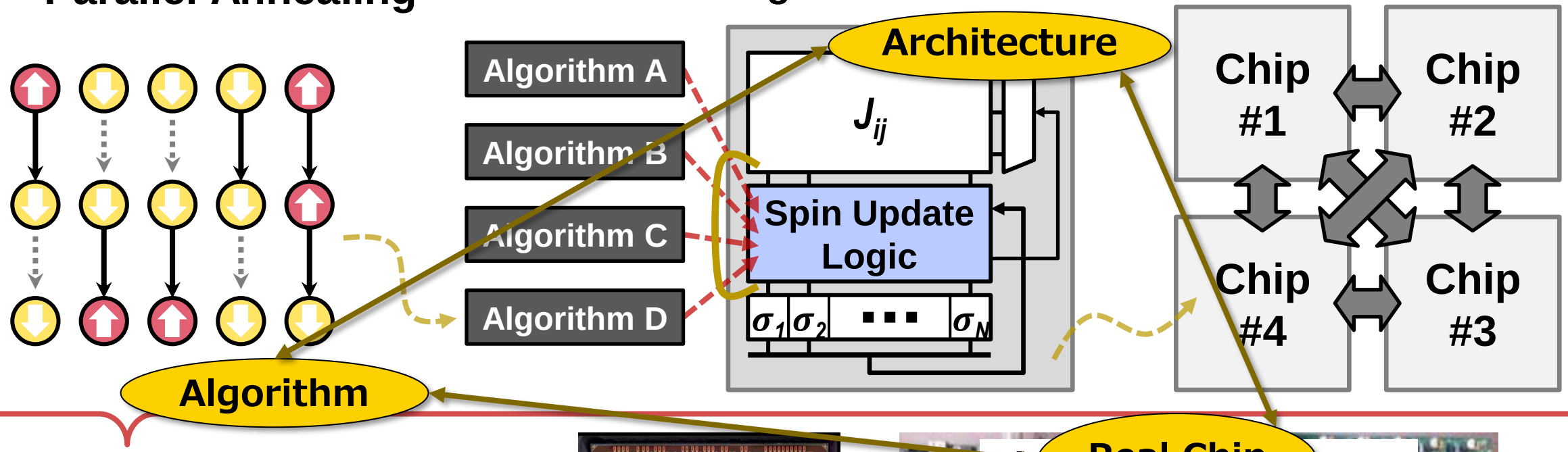
- Up to 58x speed up can be achieved, with around 1/500 power consumption. That is, 30k times more energy efficient.

Key Contributions of This Work

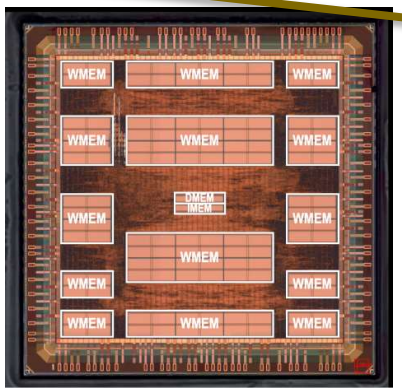
1. **RPA: Ratio-controlled Parallel Annealing**

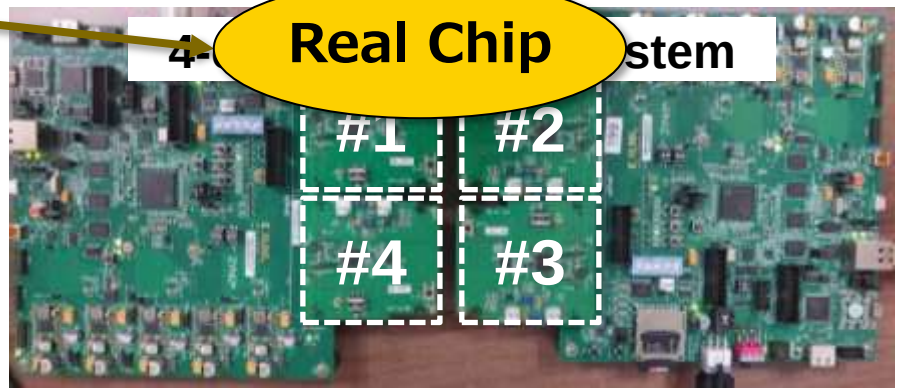
2. **Metamorphic, Near-Mem Annealing Architecture**

3. **Multi-chip Extension**

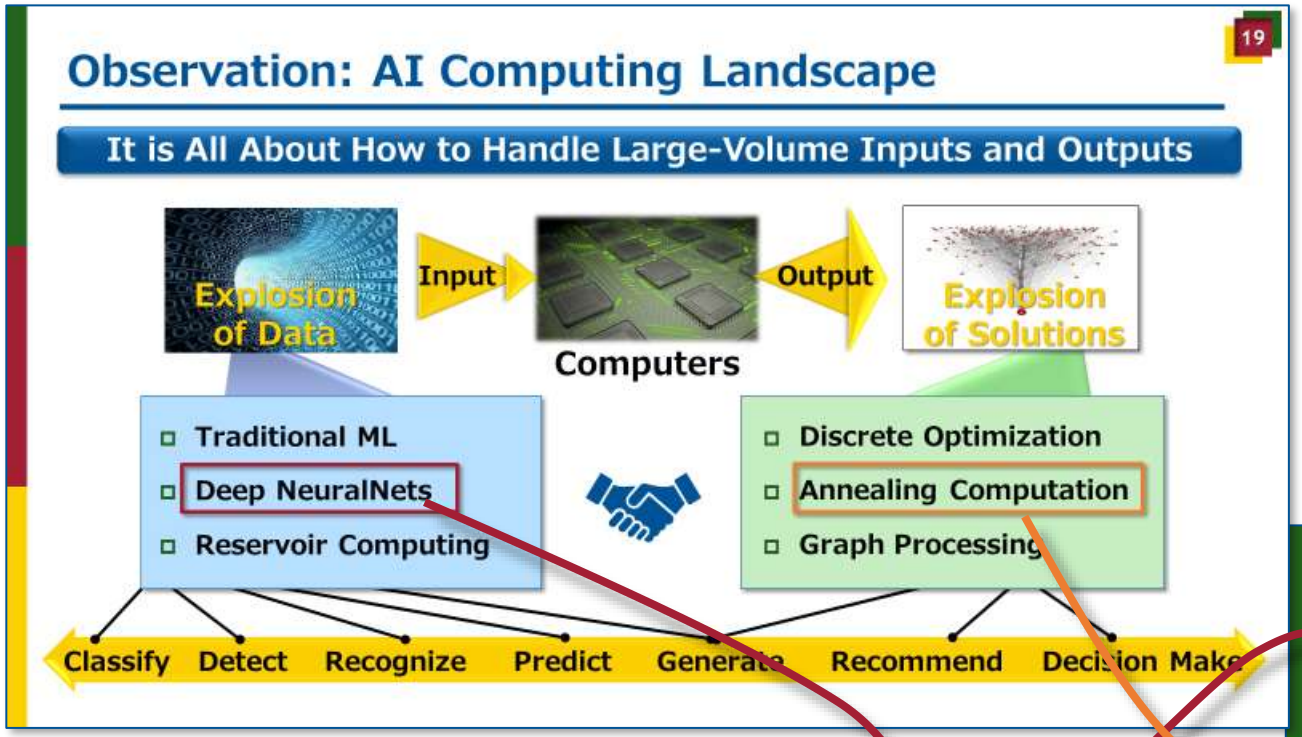


4. **Chip Implementation and Evaluation**





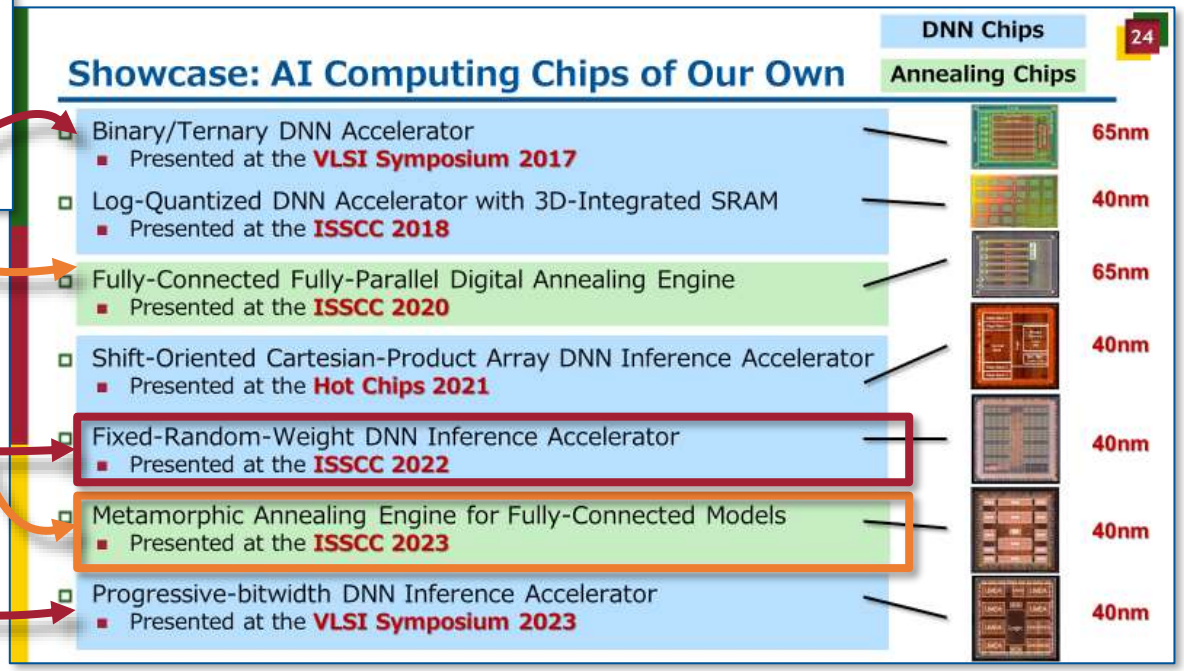
Wrap Up The Two Showcases



They all feature

- **Reduced-Bitwidth**
- **Near-Memory**
- **Element-wise Parallel**

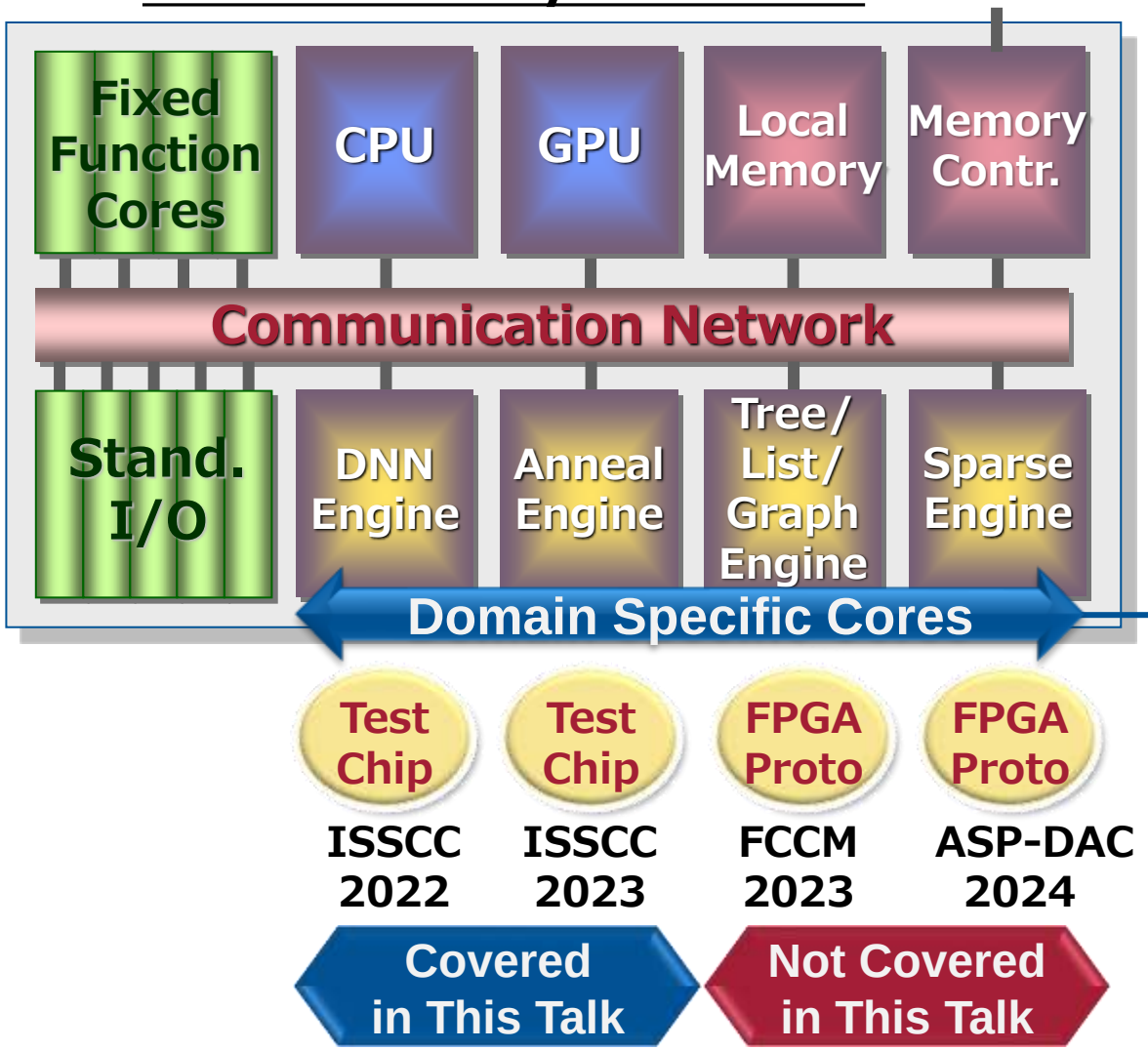
Reconfigurable Structure-Oriented Computing



This type of architectures will play pivotal rolls in near future AI Computing systems

Vision: SoCs/SiPs for the Smart-X Society

General SoC/SiP View



SoC (System on Chip), **SiP** (System in Package) for **Smart-X** Systems, e.g.,

- **Mobile Devices**
- **Mobilities**
- **Wearable Devices**

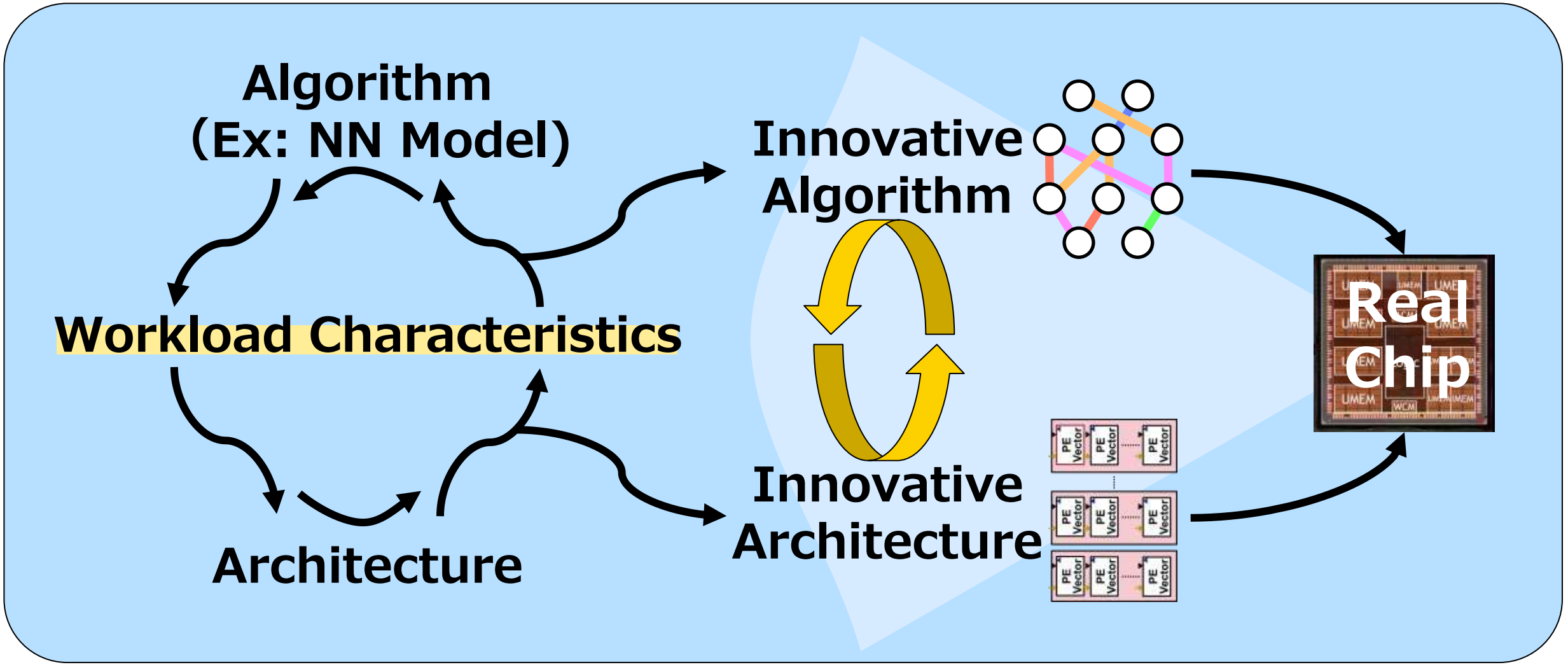
=> **Ensemble of Domain-Specific Engines**

… on some common **low-bitwidth reconfigurable and parallel architecture** foundation.

This vision explains why we value real chip implementation (as opposed to using FPGAs)

Key Takeaways

Importance of the Interplay Among Algorithm-Architecture-Real Chip



Many Thanks to Collaborators!
Questions ?

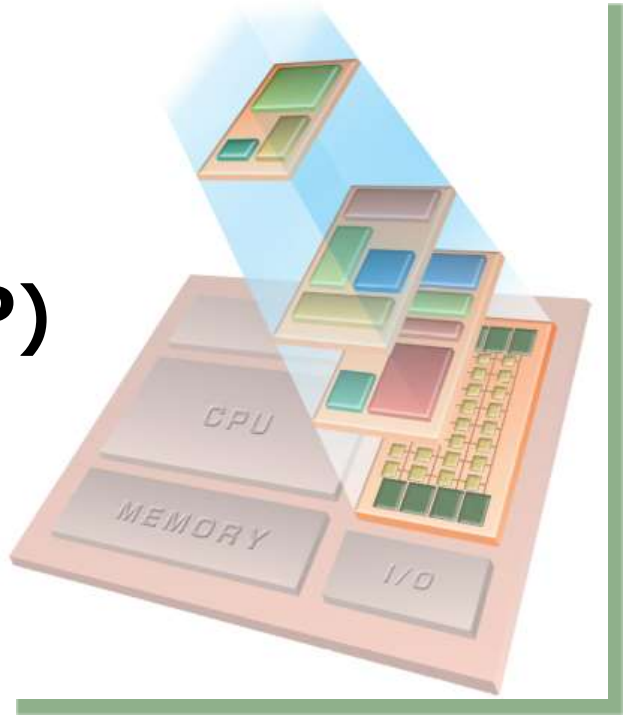
Tokyo Tech

ARTIC

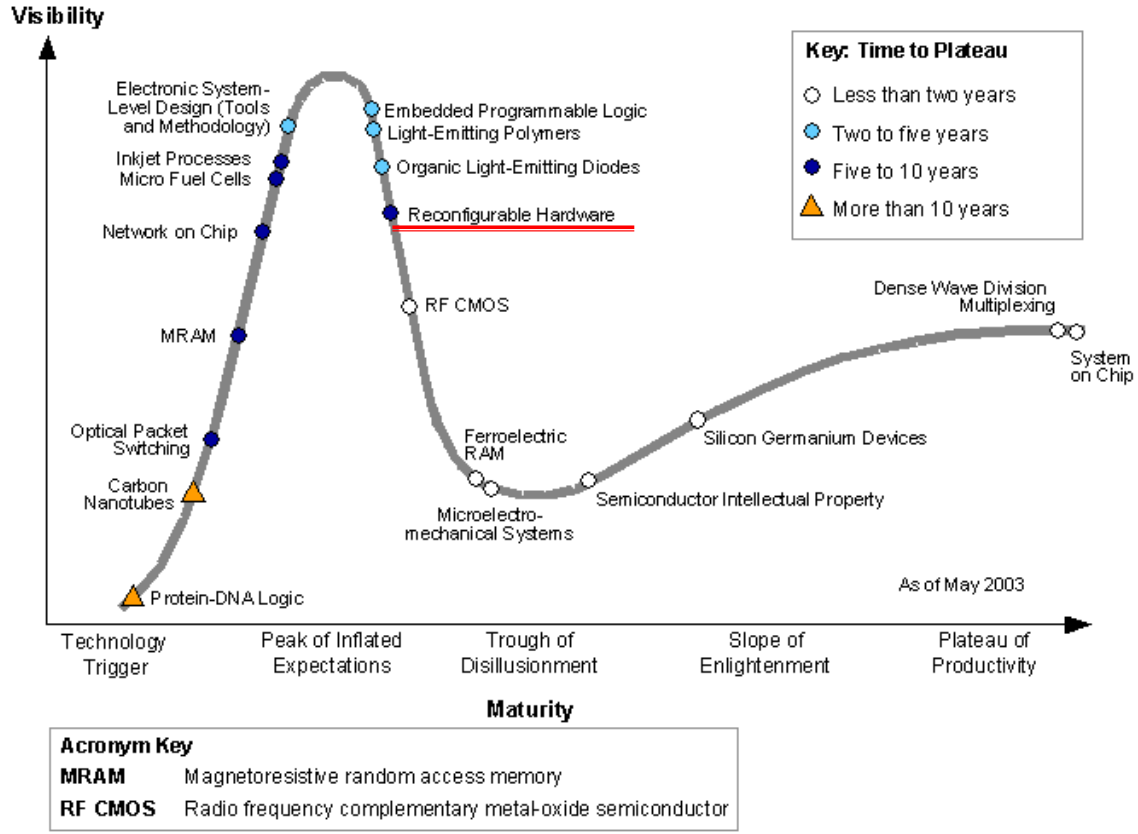
CGRA: Past and Present

- CGRA boom in late 90's to 00's
  - Lots of academic projects and startups
    - Pipe-Rench, Chameleon, IP-flex, etc.
  - Most of them "Hyped-out"

- **Dynamically Reconfigurable Processor (DRP)** started by **NEC**, succeeded by **Renesas**



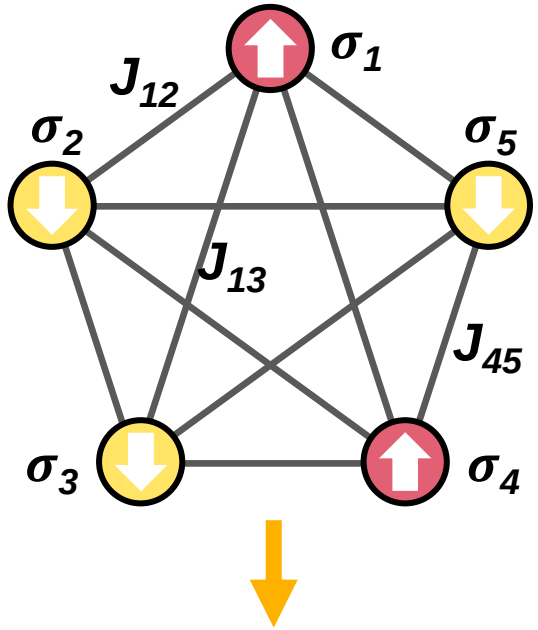
Gartner Hype Cycle 2003



DRP alone survived and continued its growth, and is now glowing beyond the age of 20th!

Challenges of Full-connection Annealing Processors

***N*-spin Ising model**

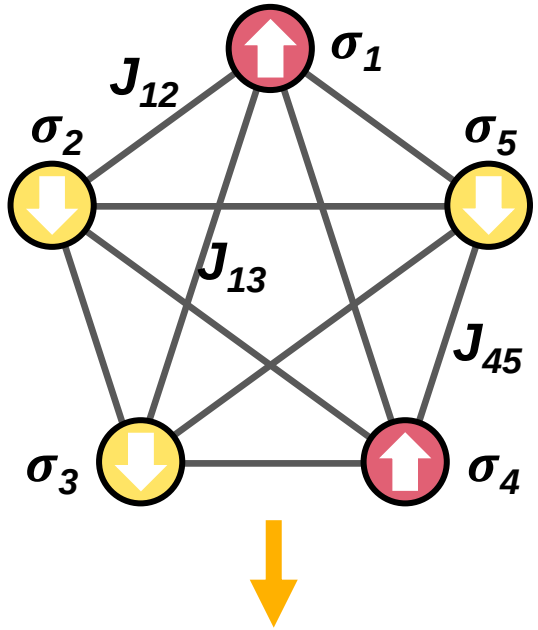


 # coupling weights = $N(N-1)$

1. **Limited Flexibility**
2. **Limited Scalability**

Challenges of Full-connection Annealing Processors

***N*-spin Ising model**

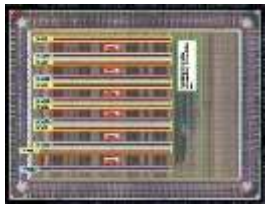


 # coupling weights = $N(N-1)$

1. Limited Flexibility

Fujitsu: Digital Annealer vs. **Our Group: ISSCC 2020**

Serial Spin-Update Policy



Parallel Spin-Update Policy

Slow — **Need Extra Ctrl**

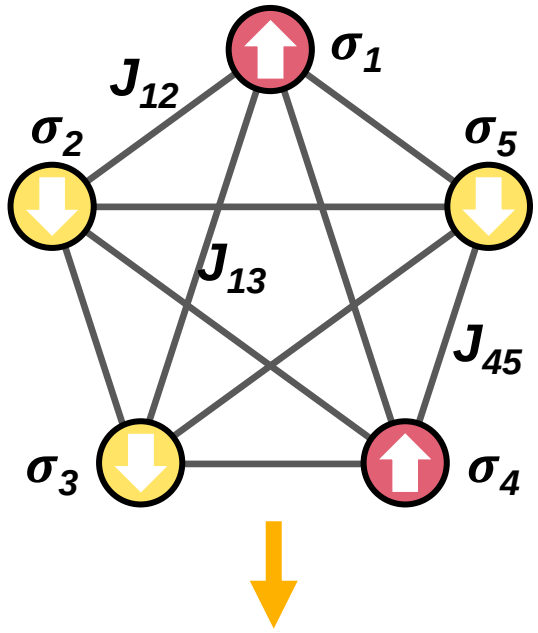
Trade-off between speed & stability

→ Which policy is Superior?

2. Limited Scalability

Challenges of Full-connection Annealing Processors

***N*-spin Ising model**



 # coupling weights = $N(N-1)$

1. Limited Flexibility

2. **Limited Scalability**

Multi-chip Distributed Annealing