Highly Efficient Neuromorphic Computing Systems with Emerging Nonvolatile Memories

Bonan Yan

Dept. Electrical & Computer Engineering

Duke University

Slides available at: https://bonanyan.github.io/bn/

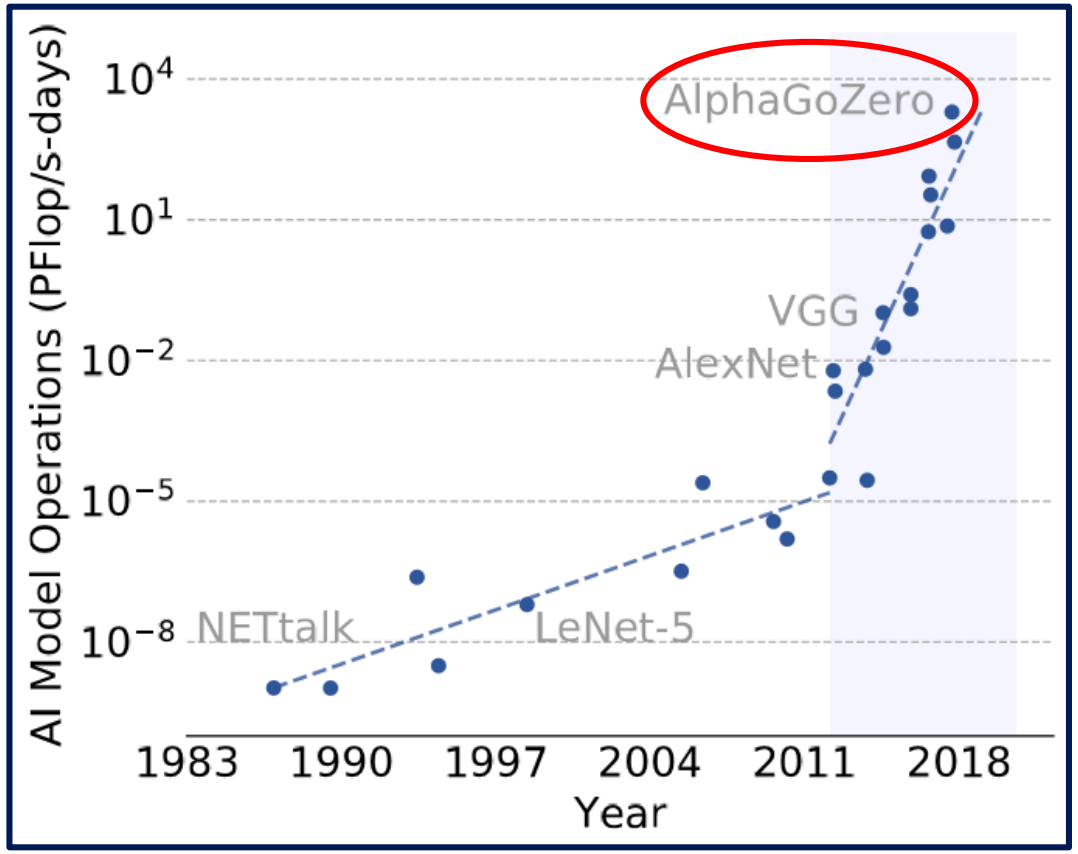
AI Needs Lots of Computation

Run on 1920 CPUs & 280 GPUs

~250,000Watts

human brain ~20Watts

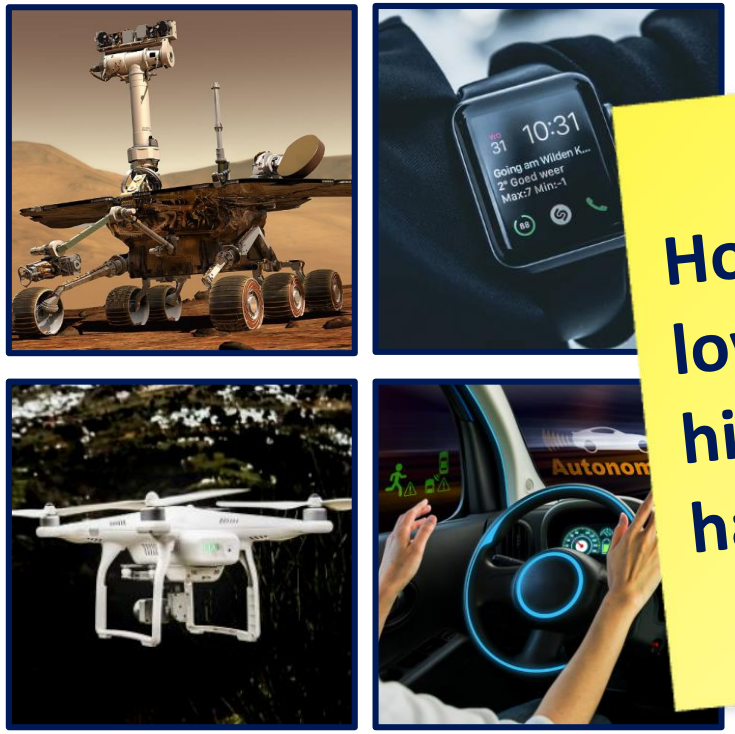
Compute Demand is Exponentially Increasing



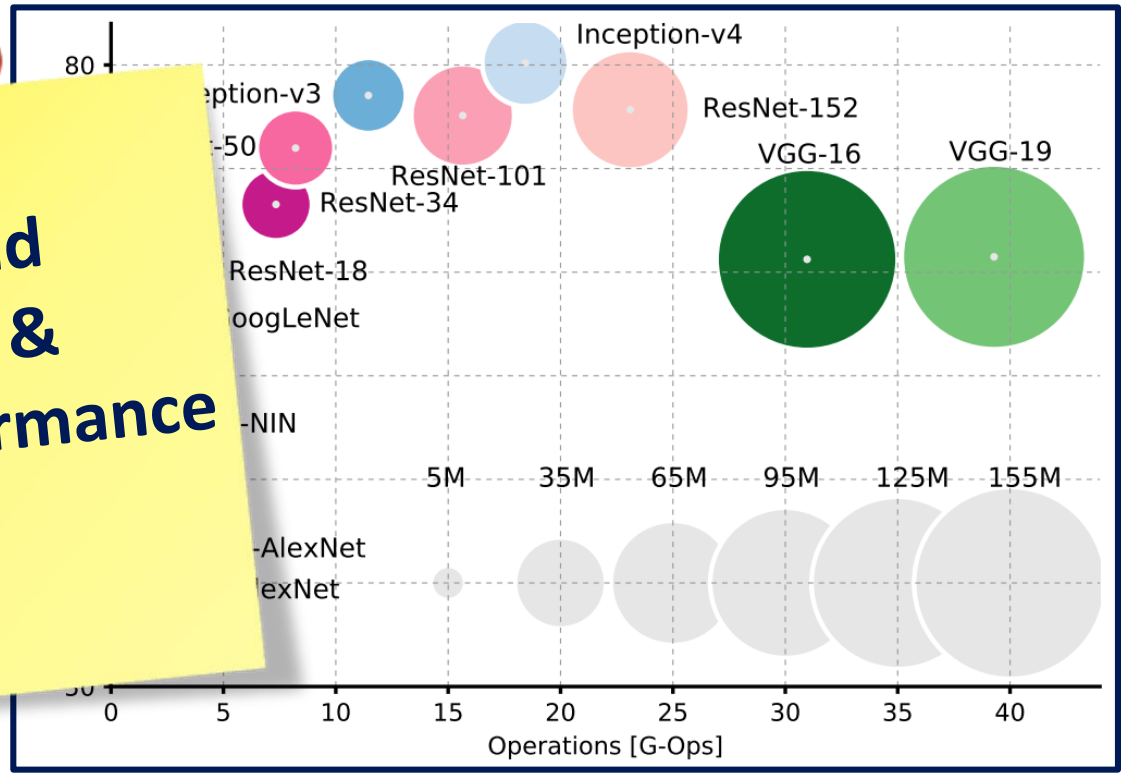
Source: [Silver, Nature 2016], DeepMind, OpenAI

Efficiency Is The Key to Ubiquitous AI

Limited Power/Energy



Better Accuracy Comes From Larger Models

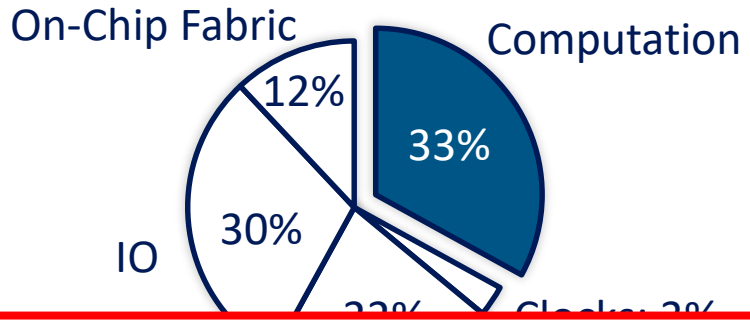


How to build low-power & high-performance hardware?

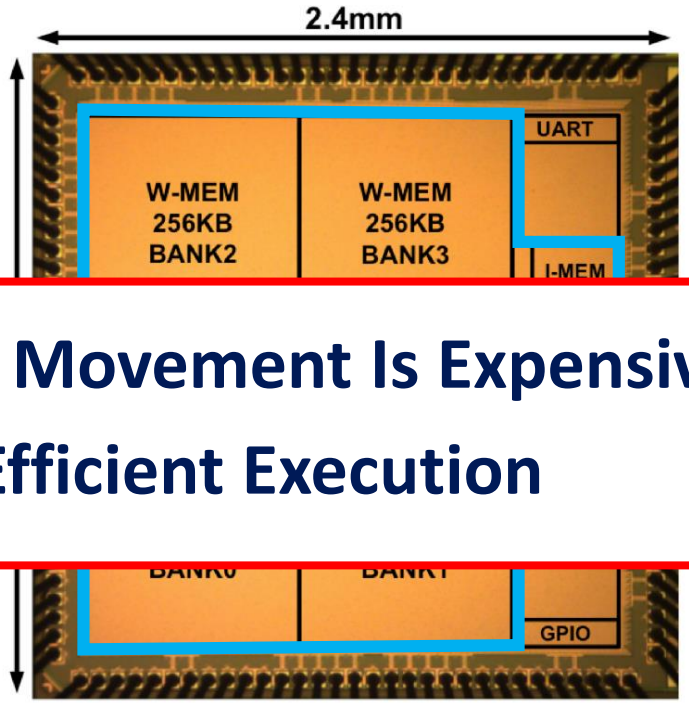
Source: IEEE Spectrum, arXiv:1605.07678

Overhead Dominated by Memories

Power



Area



- **Memory is the Bottleneck; Data Movement Is Expensive**
- **Build Specialized Hardware for Efficient Execution**

En

Float ADD	0.9
Register File	1
Int Multiply	3.1
Float Multiply	3.7
SRAM Cache	5
DRAM Memory	640

: Function Unit

: On-chip Memory

: Control Module

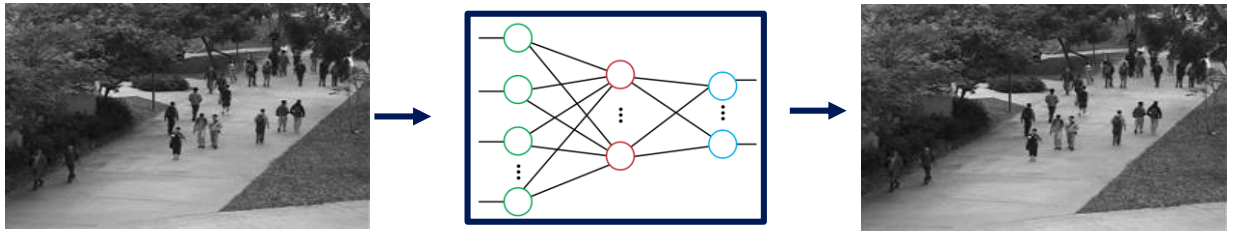
Source: AMD, Intel, [Whatmough, ISSCC 2017]

Duke

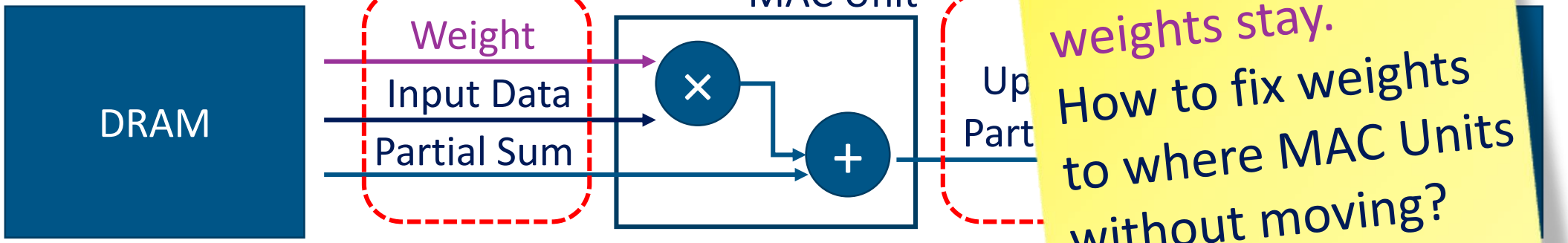
Uniqueness of Neural Network Execution

Multiply-Accumulate (MAC):

$$\overset{\text{streaming}}{[x_1\ x_2\ \dots\ x_m]} \begin{bmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \text{stationary} & \vdots \\ w_{m1} & \cdots & w_{mn} \end{bmatrix} = [y_1\ y_2\ \dots\ y_n] \quad \Rightarrow \quad y_j = \sum_{i=1}^{m} x_i \cdot w_{ji}$$

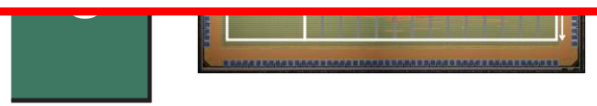
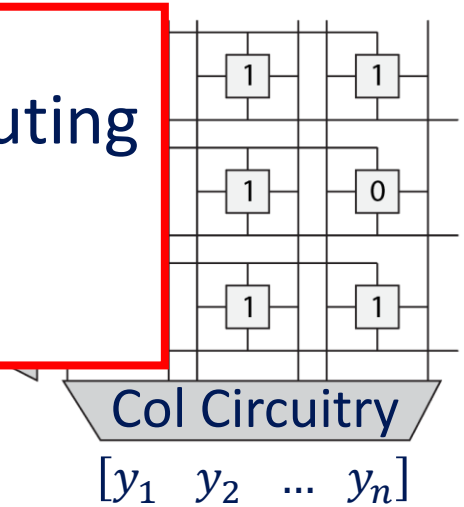


Execution:



Inference:
Inputs change;
weights stay.
How to fix weights to where MAC Units without moving?

Make Memory Access Less Expensive

	CMOS Accelerator (Planar Integration)	Near-Memory Computing	In-Memory Computing
Design Approach		High Bandwidth Memory	
Energy to Access Weights	200×~6× of MAC Energy	55×~6× of MAC Energy	**0**

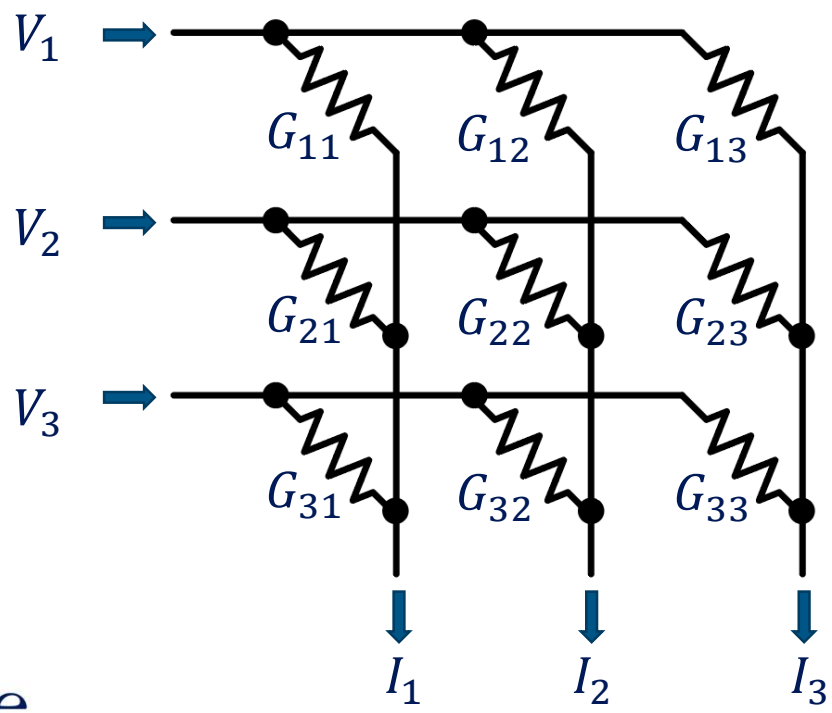
- Need new circuitry to support in-memory computing
- Rethink memory from the ground up!

DRAM On-Chip Buffer

Source: AMD, Y.-H. Chen, JSSCC,2017

Key Idea of In-Memory Computing

$$[V_1 \quad V_2 \quad V_3]\begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{21} & G_{22} & G_{23} \\ G_{31} & G_{32} & G_{33} \end{bmatrix} = [I_1 \quad I_2 \quad I_3]$$



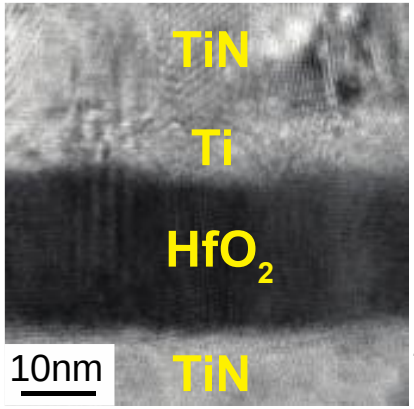
- Weight Matrix Stored as Conductance $\boldsymbol{G}$
- Rely on Analog Computation (Kirchhoff's Current Law) for "almost free"
  - Multiplication: $\mathrm{I} = V \cdot G$
  - Addition: $I^{column} = I_1^{row} + I_2^{row} + I_2^{row}$
- Ideal Nanoscale Devices for $\boldsymbol{G}$:
  - Programmable Conductance
  - Multi-Level Cell
  - Small Footprint/High Density
  - Compatible with Existing CMOS Process

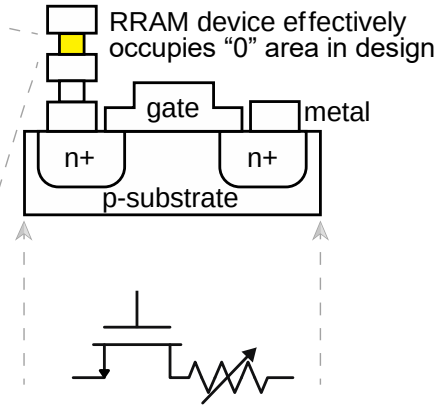
Duke

Memristors for In-Memory Computing

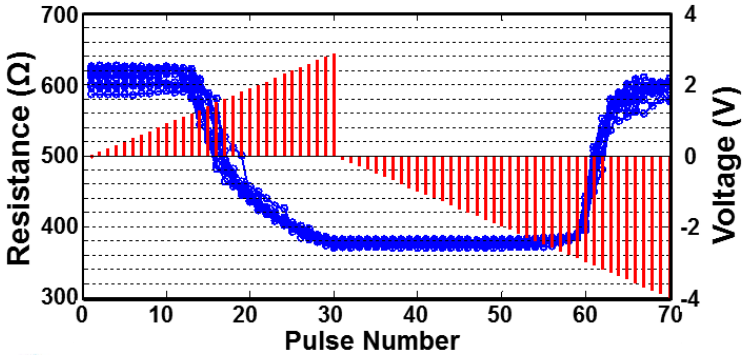
Also Called **R**esistive **R**andom **A**ccess **M**emory, RRAM or ReRAM

Cross-section TEM





1T1R ~50F²



Programmable resistor w/ analog states

ISSCC: Intel adds embedded ReRAM to 22nm portfolio

January 03, 2019

TSMC to start embedded RRAM production in 2019

According to reports, Taiwan Semiconductor Manufacturing Company (TSMC) is aiming to start producing embedded RRAM chips in 2019 using a 22 nm process. This will be initial "risk production" to gauge market reception.

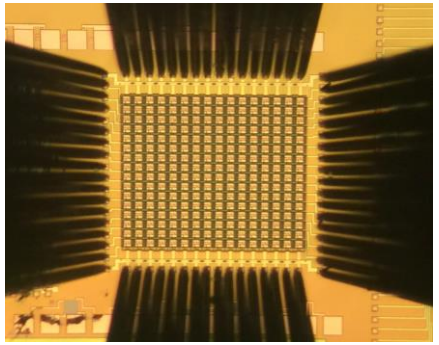
	Multi-Level Cell	Cell Area	R/W Speed
SRAM	×	large	Fast
DRAM	×	medium	Medium
1T1R	√	medium	Medium Fast
Flash	√	small	Slow

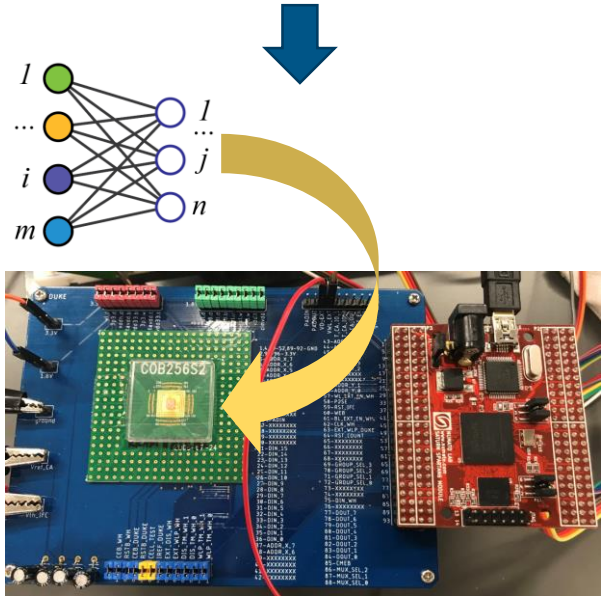
Duke

My work: Emerging Memory-Centric Design

- **Circuits & Systems Implementation**
  - Spike-based Interface [DAC'15, DAC'18, DAC'20]
  - Implementation of Neural Networks [VLSI'19, DAC'20]

- **Tolerate/Exploit Non-ideal Behavior of Memristors**
  - Device Nonlinearity [ISCAS'16, IEDM'17, IEDM'19]
  - Read Disturbance [ICCAD'17]



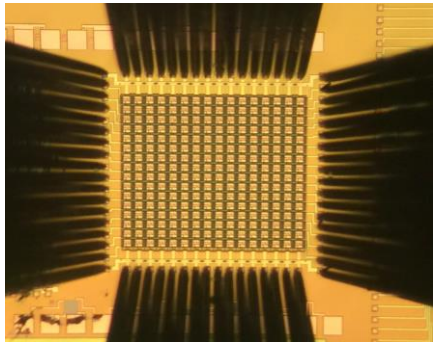


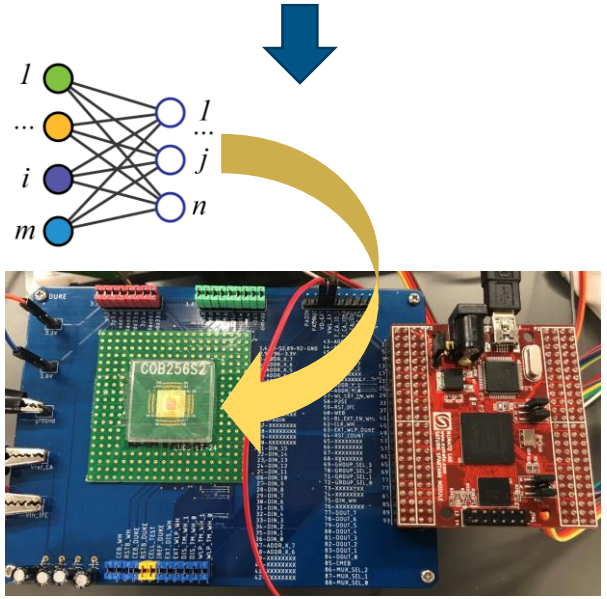
Duke

My work: Emerging Memory-Centric Design

- **Circuits & Systems Implementation**
  - Spike-based Interface [DAC'15, DAC'18, DAC'20]
  - Implementation of Neural Networks [VLSI'19, DAC'20]

- **Tolerate/Exploit Non-ideal Behavior of Memristors**
  - Device Nonlinearity [ISCAS'16, IEDM'17, IEDM'19]
  - Read Disturbance [ICCAD'17]



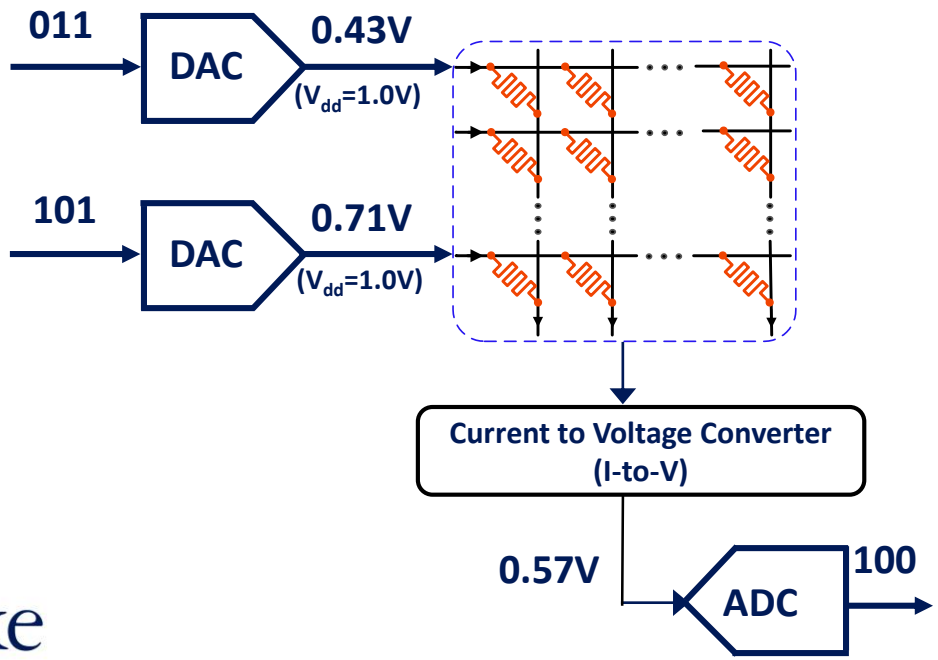


Duke

Conventional ADC is Too Large

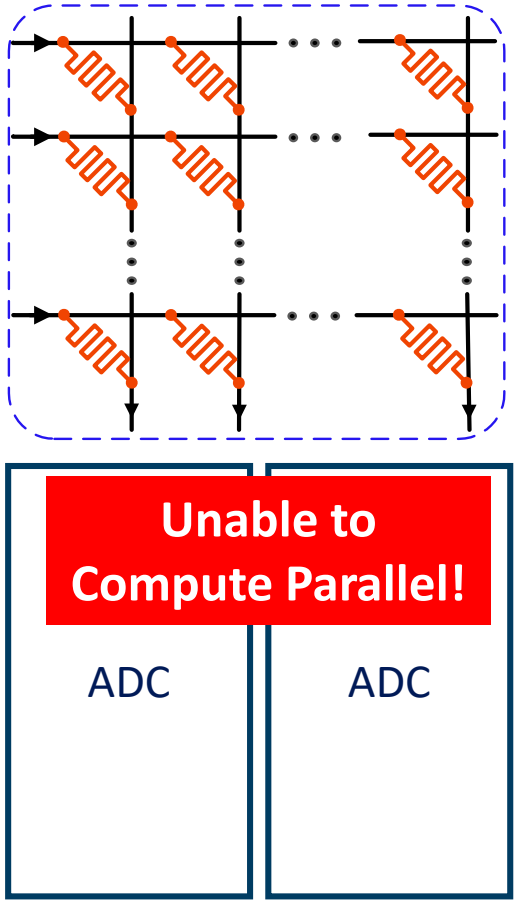
The Level-based Design

- Compatible to existing signal processing
- High speed computation



Actual Layout:

256x256 1T1R Array



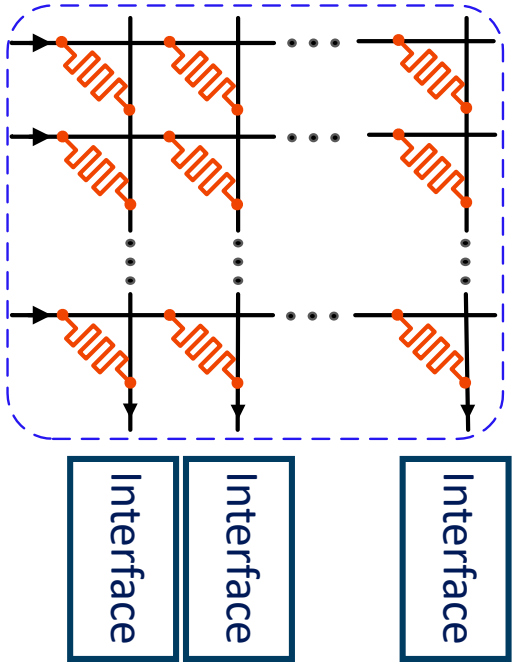
Based on 1.66MF^2 8bit ADC
by K. Ohhata (JSSC 2019)

Duke

My Approach: Spiking Interface Circuit

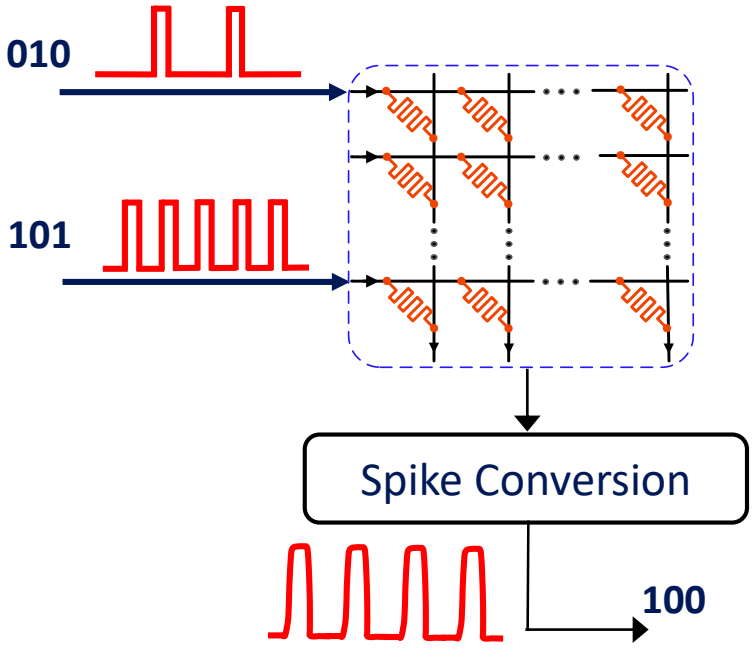
What Better Designs Look Like

- Compute Parallelly (Massive)
- Need Light-Weight Interface Circuitry



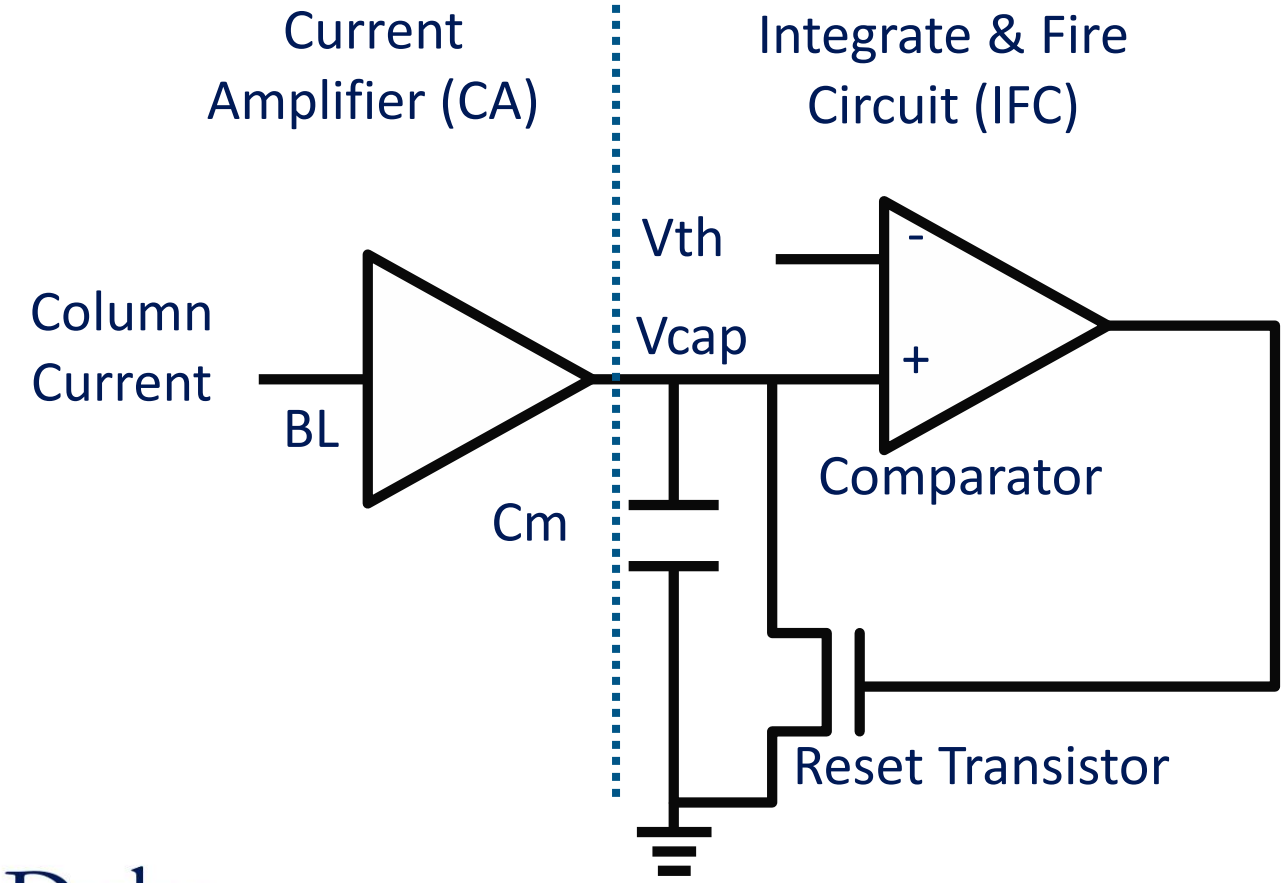
The Spike-based Design

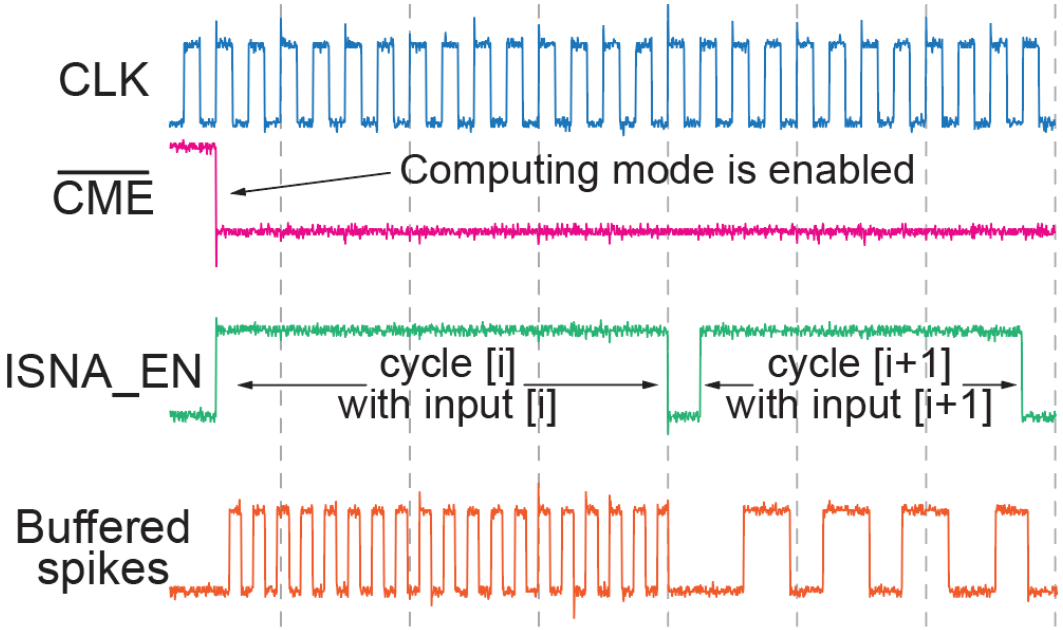
- Closer to biological system
- Extremely high power efficiency

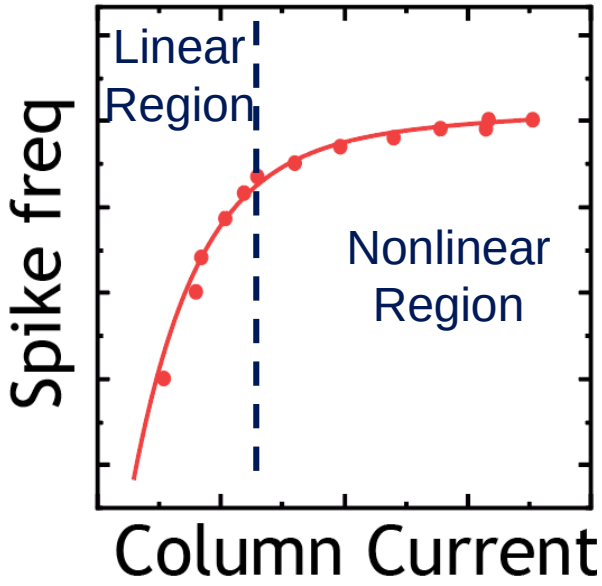


Duke

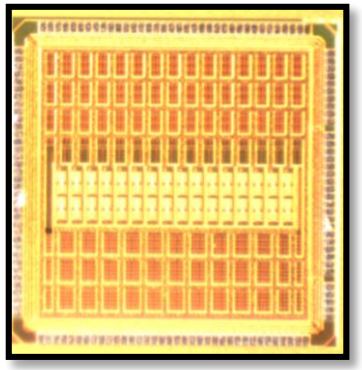
Spike Conversion





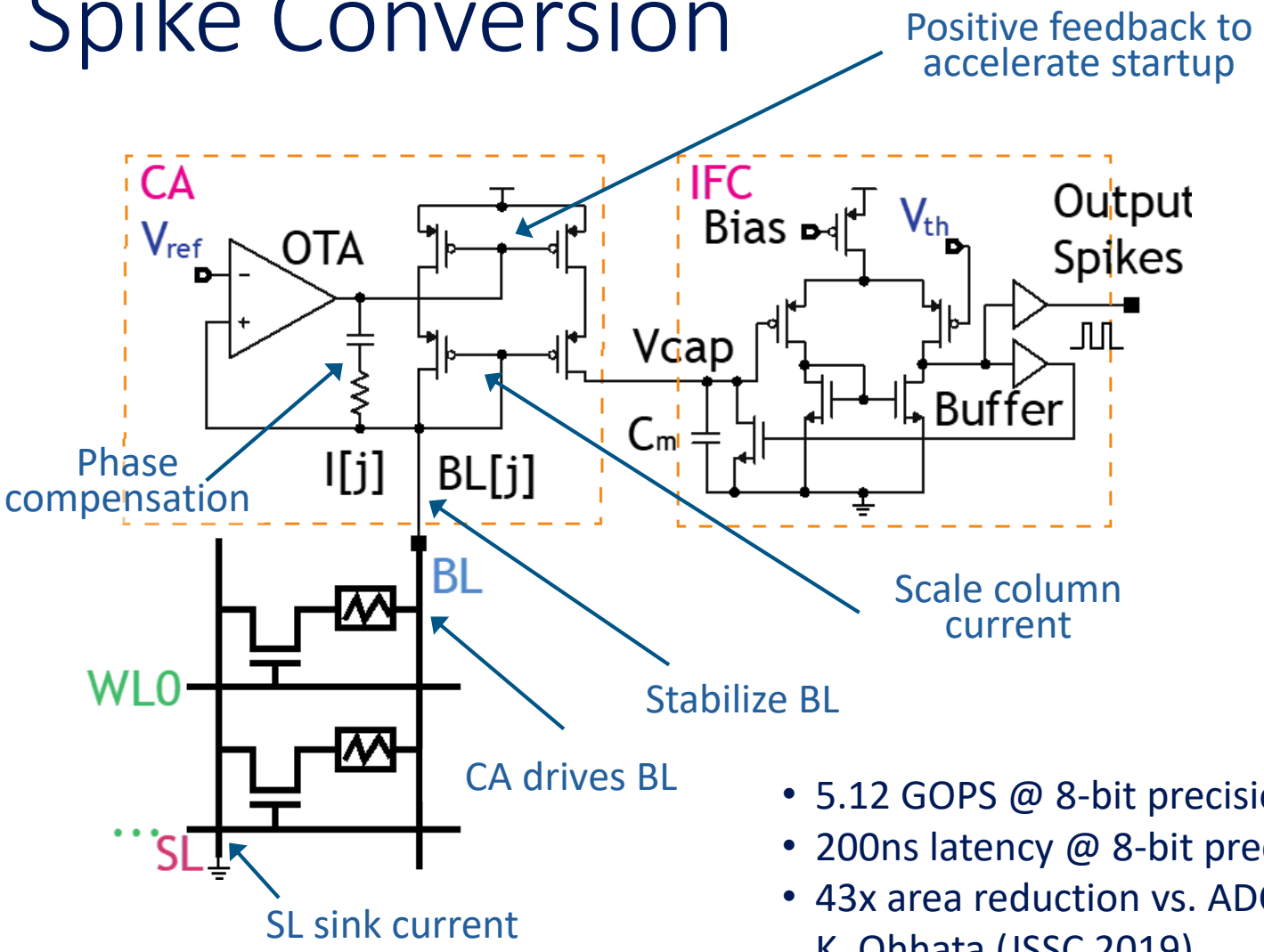


Spike Conversion
Circuit & Controller
3152x3152 μm^2

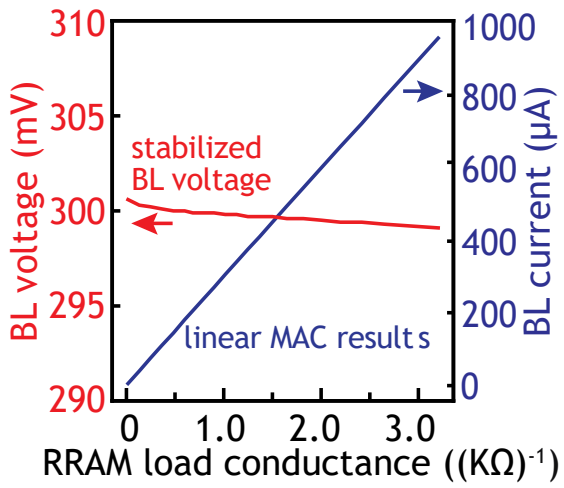


***B. Yan**, et al., ISCAS, 2016*

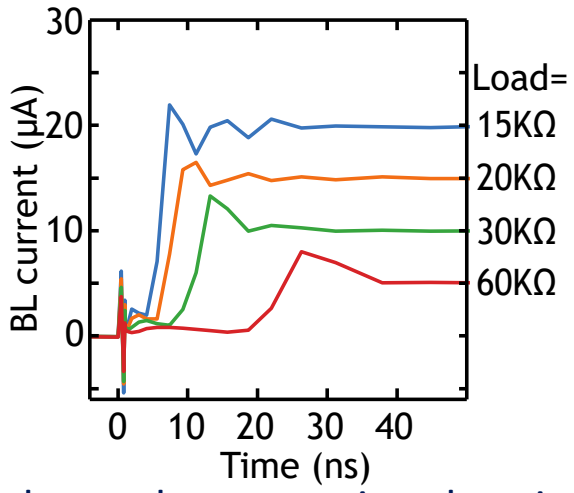
Spike Conversion



- 5.12 GOPS @ 8-bit precision
- 200ns latency @ 8-bit precision
- 43x area reduction vs. ADC by K. Ohhata (JSSC 2019)



Tradeoff between large input current range and response speed



Enlarge phase margin tolerating capacitor positive feedback

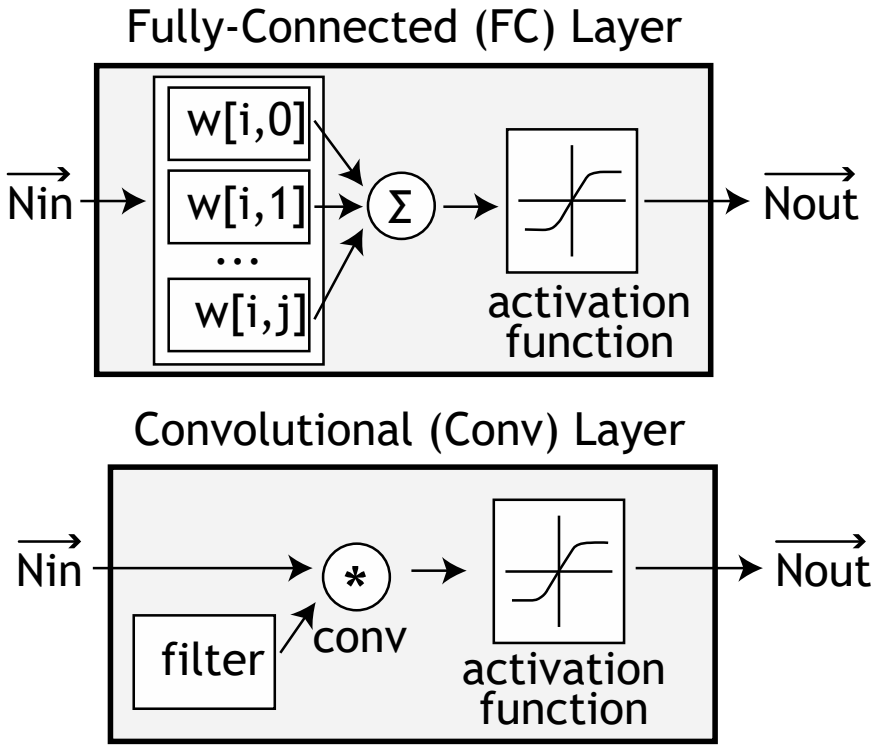
Duke

B. Yan, et al., IEEE VLSI Symp. on Tech. 2019

How to Use Spiking-Based Design to Execute Neural Networks?

Duke

In Situ Nonlinear Activation (ISNA) Function



Single-layer Inference Operation:

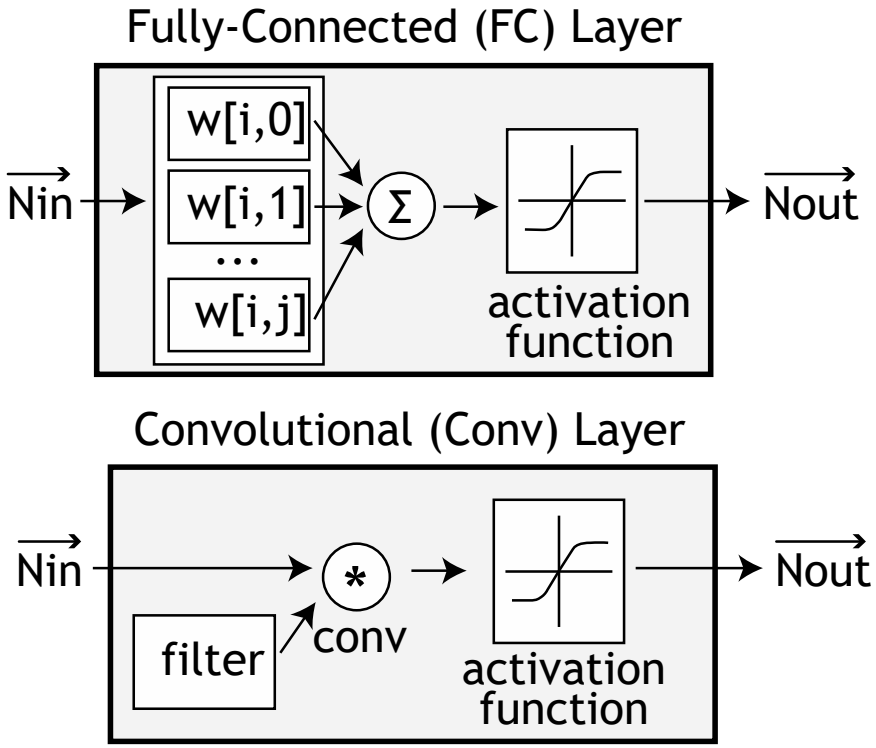
- **D** Step 1: Load data from buffer
- **A** Step 2: Vector-matrix multiplication
- **D** Step 3: Nonlinear activation function
- **D** Step 4: Pooling
- **D** Step 5: Store results to buffer

D : digital domain **A** : analog domain

Duke

B. Yan, et al., IEEE VLSI Symp. on Tech. 2019

In Situ Nonlinear Activation (ISNA) Function



Single-layer Inference Operation:

D • Step 1: Load data from buffer

A • Step 2: Vector-matrix multiplication
• Step 3: Nonlinear activation function

D • Step 4: Pooling

D • Step 5: Store results to buffer

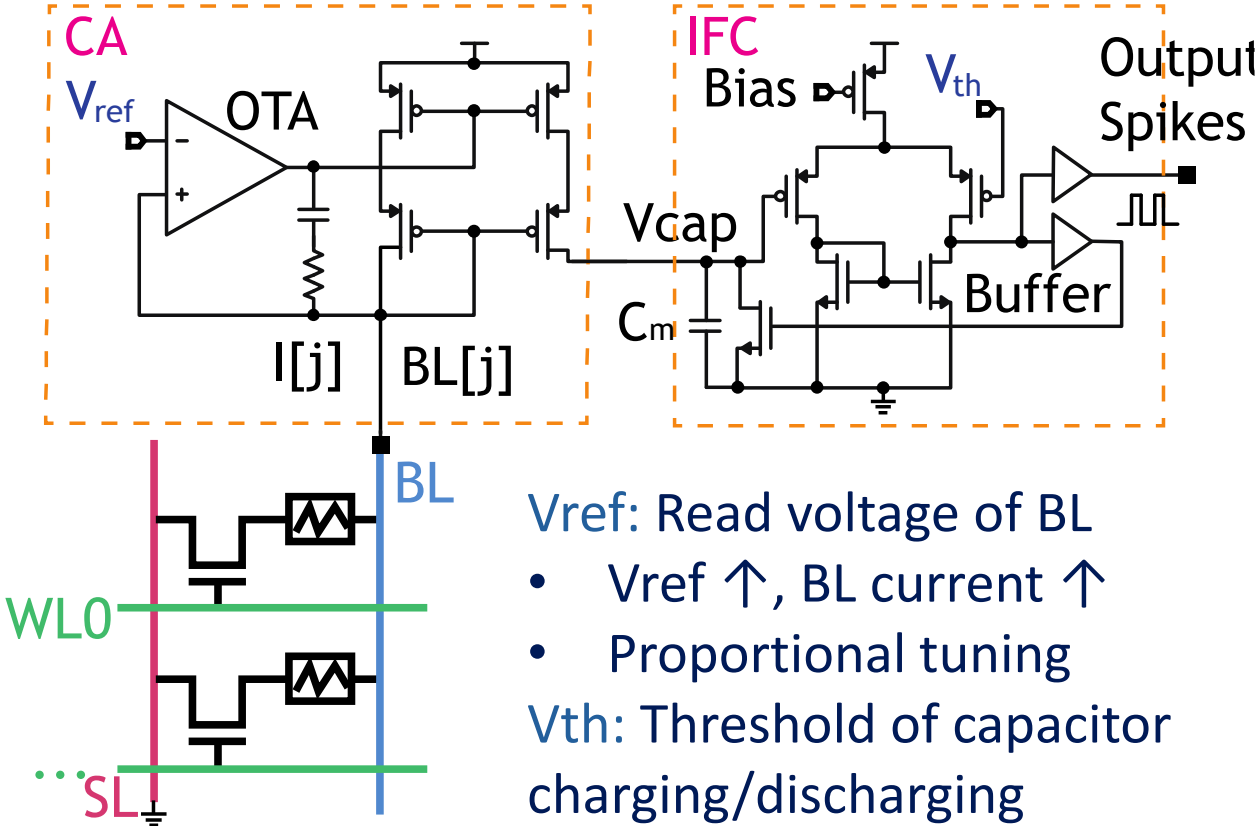
D : digital domain **A** : analog domain

Combine Step 2 & Step 3 to simplify PE operation:
Use linear + nonlinear regions

Duke

B. Yan, et al., IEEE VLSI Symp. on Tech. 2019

Adjust Activation Function

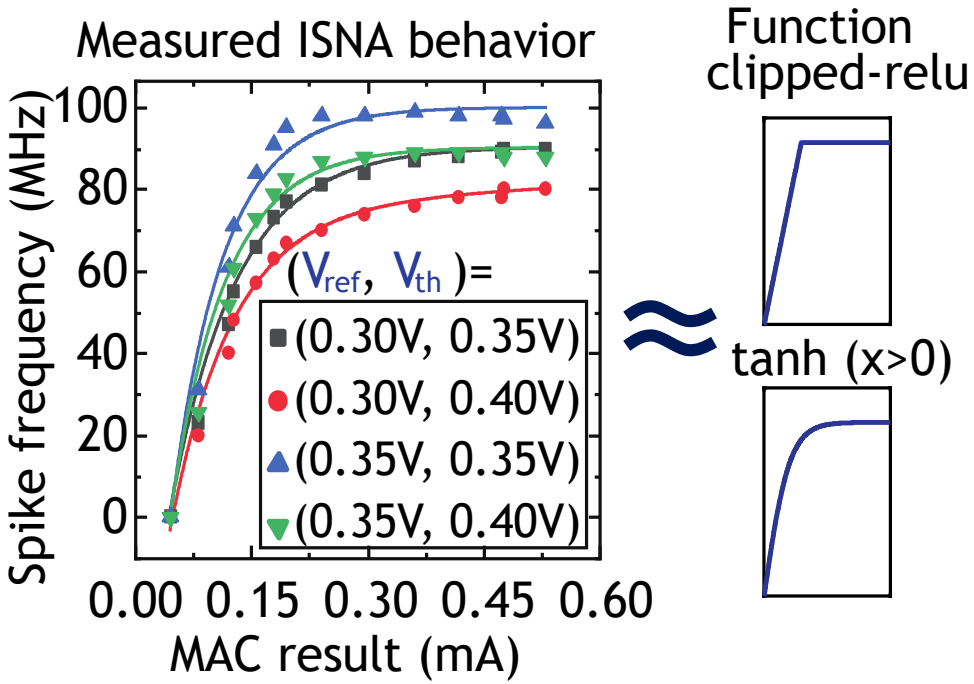


Vref: Read voltage of BL

- Vref ↑, BL current ↑
- Proportional tuning

Vth: Threshold of capacitor charging/discharging

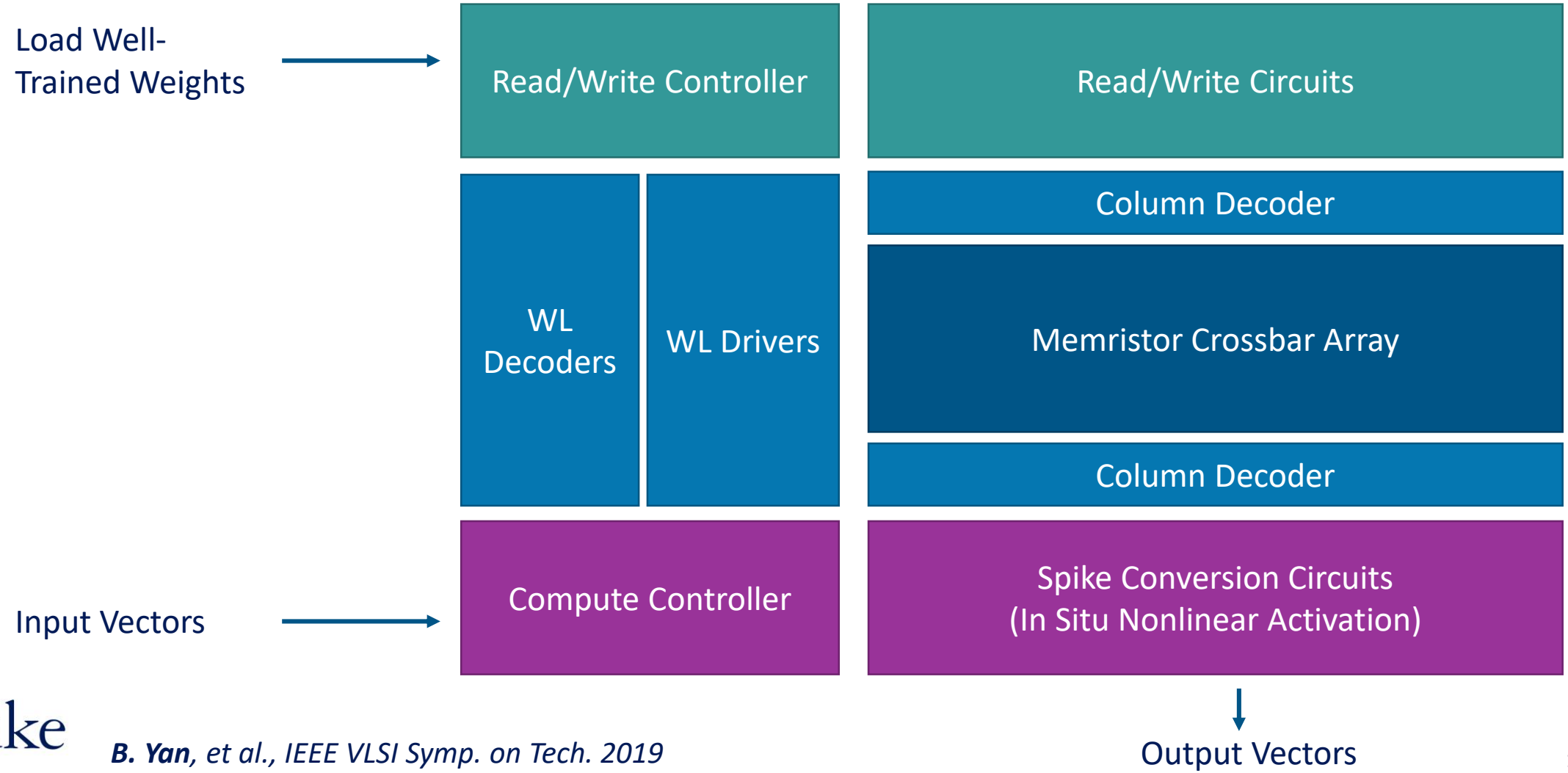
- Vth ↓, Charging/discharging ↑
- Distorted tuning



Duke

B. Yan, et al., IEEE VLSI Symp. on Tech. 2019

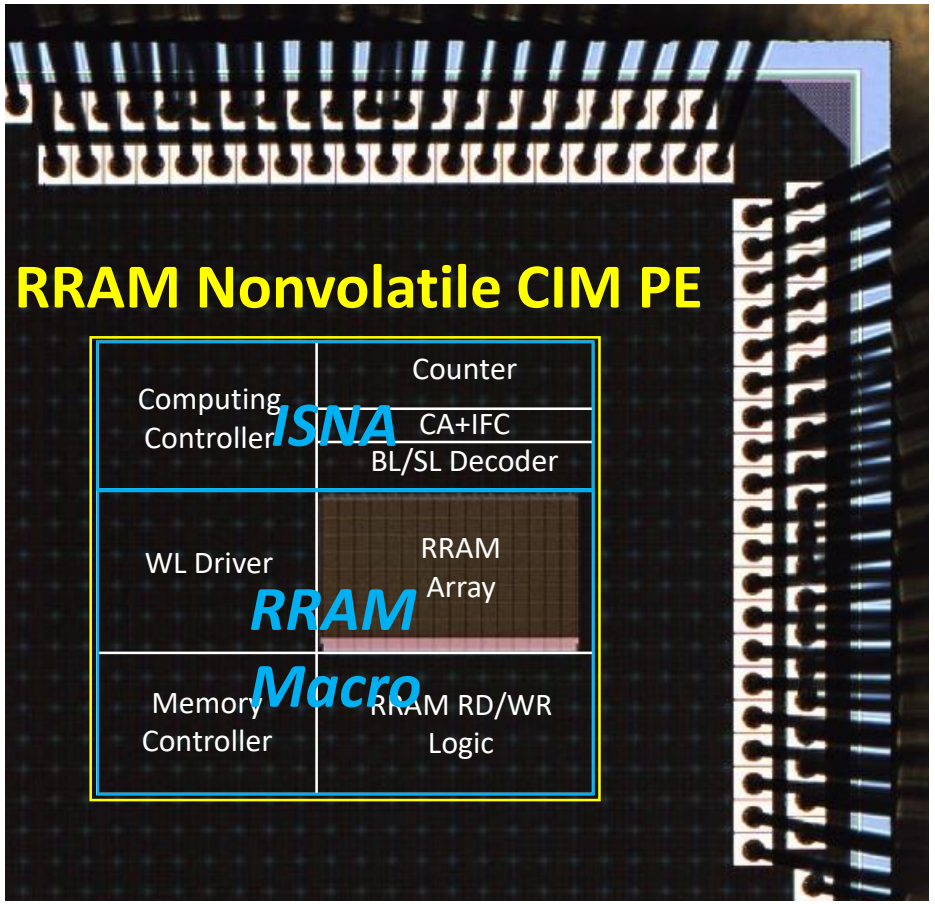
Chip Architecture



Duke

***B. Yan**, et al., IEEE VLSI Symp. on Tech. 2019*

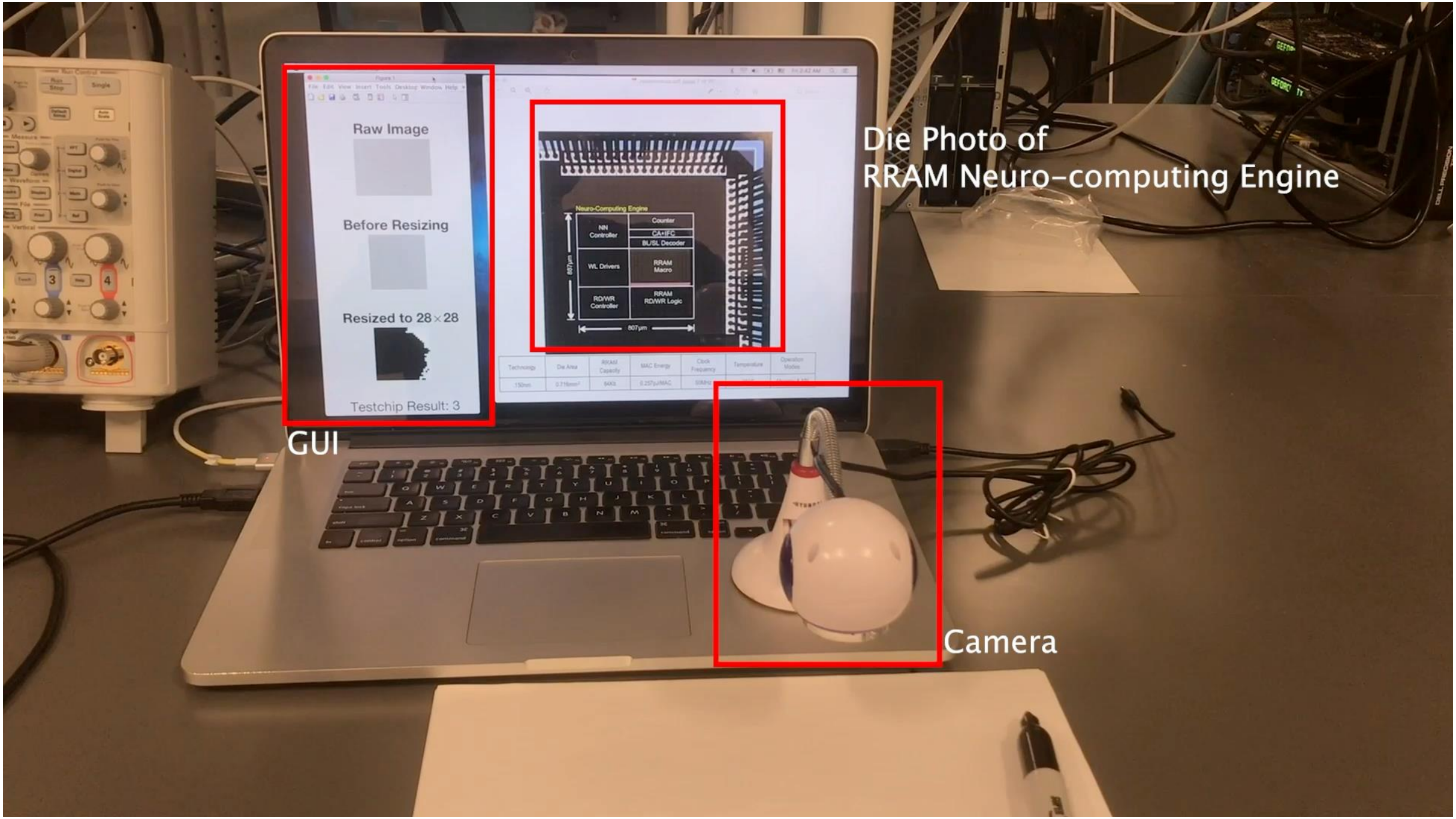
Chip Summary



Technology	150nm CMOS +HfO_x RRAM
Macro Capacity	64K (256×256)
Clock Frequency	50MHz
Energy Efficiency	0.257pJ/Mac
Average Power	1.52 mW
Layer-wise Latency	200ns
Real-time Benchmarks	3-layer perceptrons, LeNet-4, LetNet-5

Demo video online: https://bit.ly/AICHIP

Duke


Raw Image
Before Resizing
Resized to 28×28
Testchip Result: 3
GUI
Neuro-Computing Engine
NN Controller
Counter
CA+IFC
BL/SL Decoder
WL Drivers
RRAM Macro
RD/WR Controller
RRAM RD/WR Logic
807μm
807μm
Die Photo of
RRAM Neuro-computing Engine
Camera

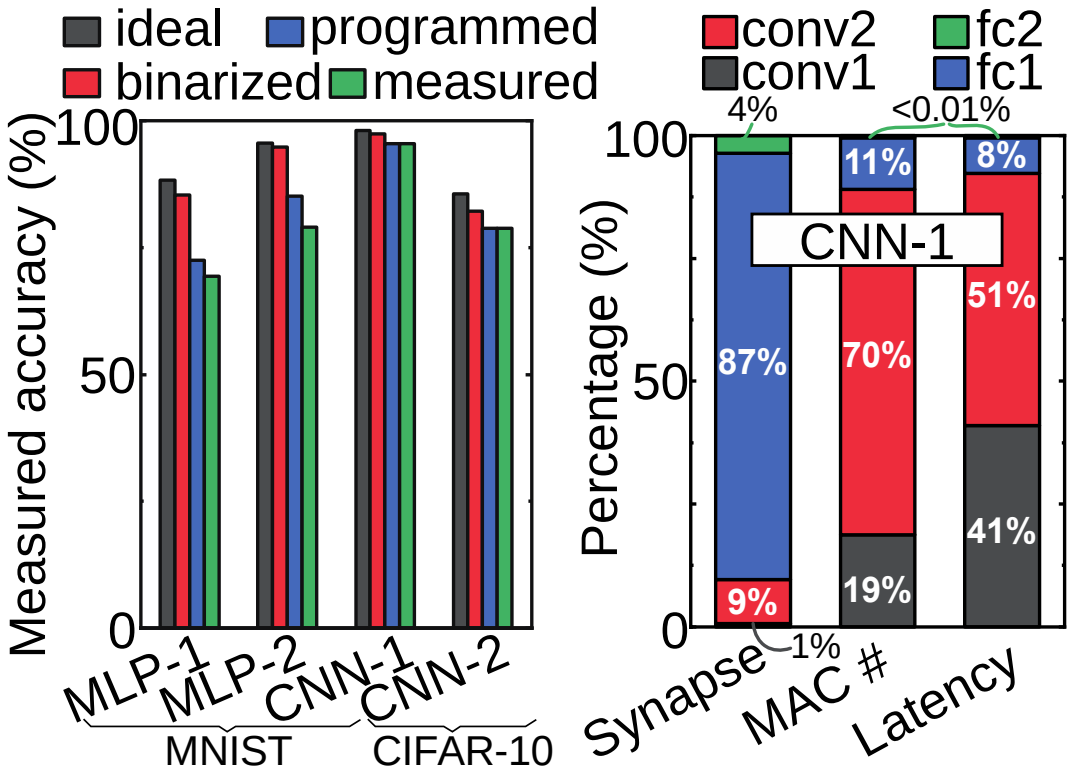
Evaluation: Measured Neural Network Results

MNIST:

- MLP-1: Single-layer perceptron
- MLP-2: 2-layer perceptron
- CNN-1: 4-Layer LeNet

CIFAR-10:

- CNN-2: 5-Layer LeNet



B. Yan, et al., IEEE VLSI Symp. on Tech. 2019

Comparison

Type		CIM Macro				CIM PE		Digital Processor
Work		VLSI'18 Panasonic	ISSCC'18 NTHU	ISSCC'18 NTHU	ISSCC'18 MIT	This work†	ISSCC'18 UIUC	TrueNorth [10]
Technology		180nm	65nm	65nm	65nm	150nm	65nm	28nm
Synapse		1T1R RRAM	1T1R RRAM	6T-SRAM	10T-SRAM	1T1R RRAM	6T-SRAM	SRAM
Nonvolatility		Yes	Yes	No	No	Yes	No	No
Standby current		~zero	~zero	high	high	~zero	high	high
Spiking NN		No	No	No	No	Yes	No	Yes
Capacity		2M	1M	4K	16K	64K	128K	256M
Cell area [F^2]		—	59	124	968	74	~256	—
Normalized die area		12×	—	—	30×	1×	11×	~17240×
Chip power [mW]		15.8	—	—	—	1.52	—	204.4
Activation precision		1 bit	3 bit	1 bit	7 bit	1~8 bit	8 bit	1 bit
Power efficiency [TOPS/W]	MAC only	20.7	16.95	55.8	28.1	—	—	—
	MAC +Activation	—	—	—	—	16.9	3.125	0.4
On-chip Activation Function Integration		No	No	No	No	Yes (tanh)	Yes (relu)	Yes (relu)
FoM*		—	0.86	0.45	0.20	1.83	0.098	—

FoM = energy efficiency maximum activation precision/cell area.

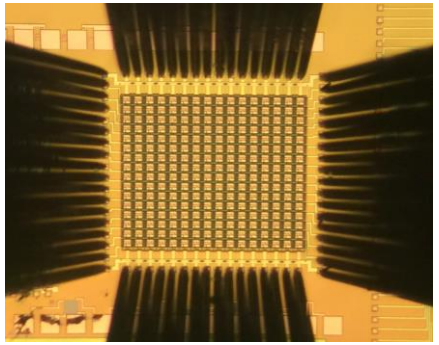
***B. Yan**, et al., VLSI Symp. 2019*

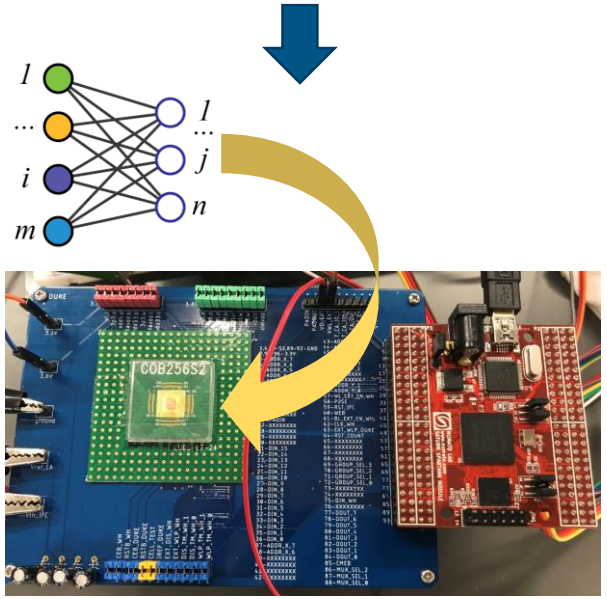
Duke

My work: Emerging Memory-Centric Design

- **Circuits & Systems Implementation**
  - Spike-based Interface [DAC'15, DAC'18, DAC'20]
  - Implementation of Neural Networks [VLSI'19, DAC'20]

- **Tolerate/Exploit Non-ideal Behavior of Memristors**
  - Device Nonlinearity [ISCAS'16, IEDM'17, IEDM'19]
  - Read Disturbance [ICCAD'17]



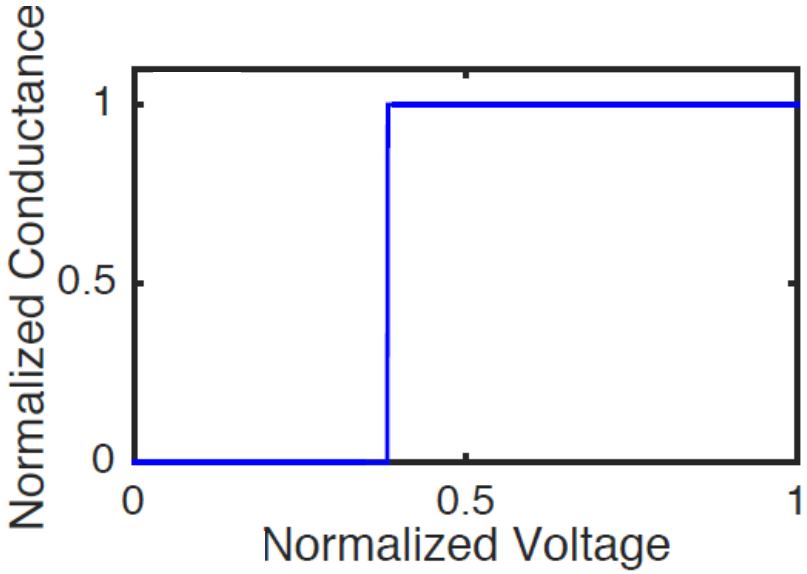


Duke

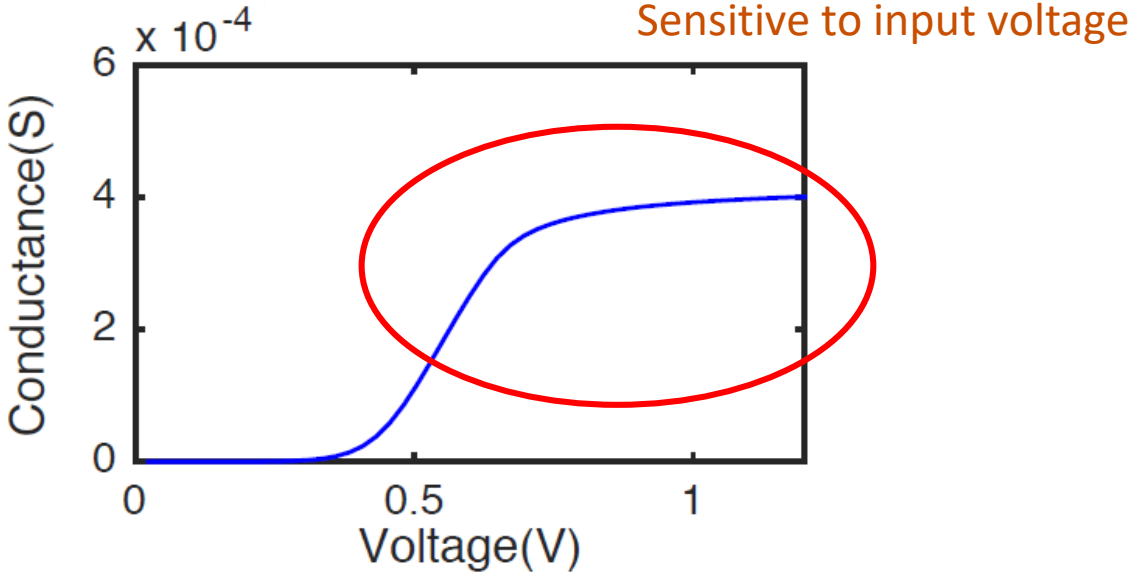
Non-Ideal Memristor - I

- Cell Nonlinearity: conductance varies when applied with different voltages

Ideal Cell



Non-Ideal Cell



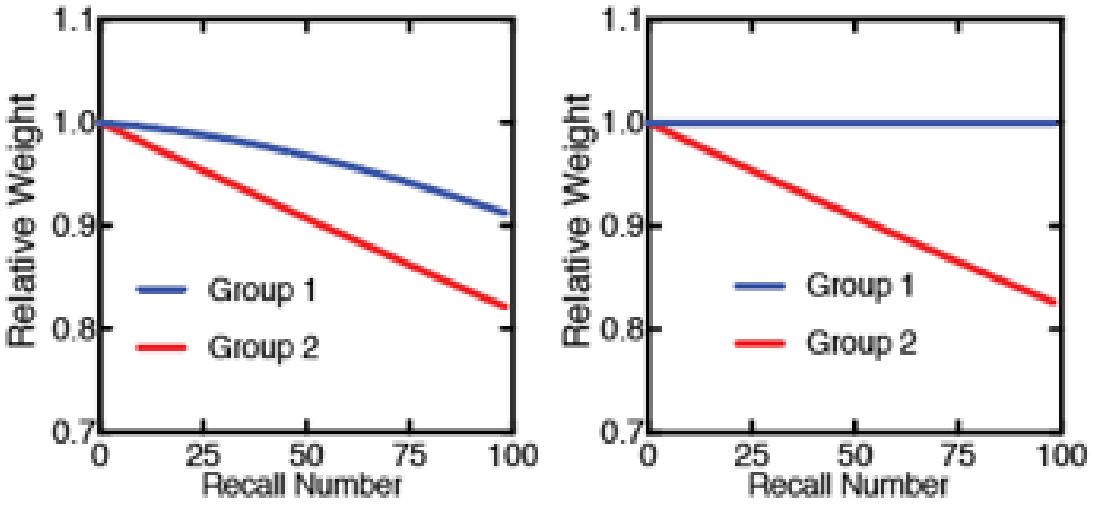
- Solution: Current Amplifier to Clamp Cell Voltage (shown in previous circuit design part)

Duke

***B. Yan**, et al., ISCAS, 2016*

Non-Ideal Memristor - II

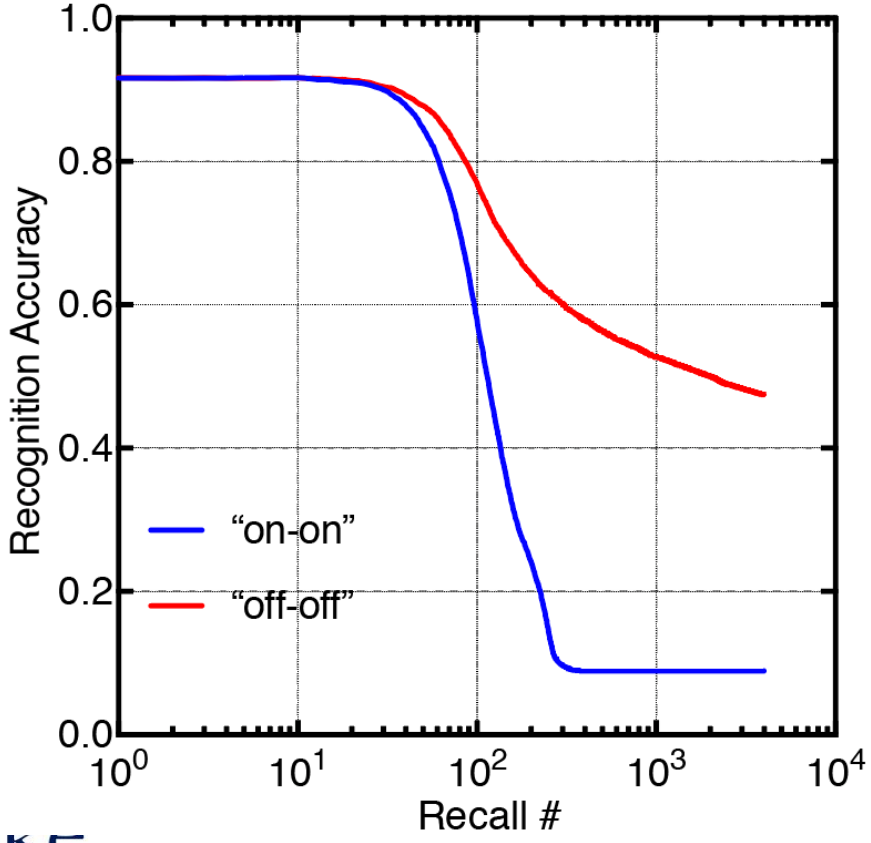
- Memristance Shift: conductance/memristance gradually deviates from original values under read voltage (read disturbance)
- Characteristic:
  - Happened very slow—hard to simulation

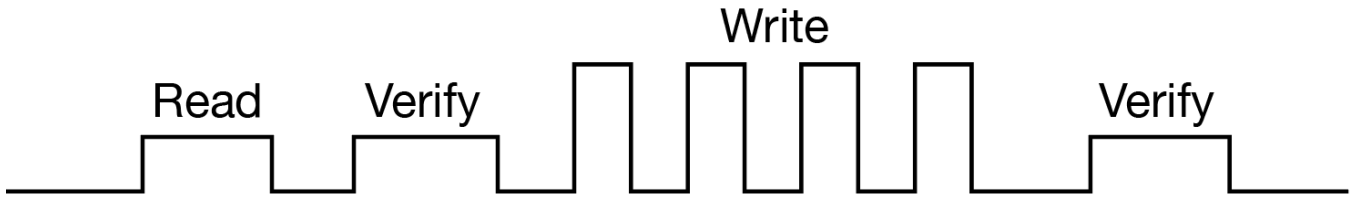


Characterize memristor models with a factor of 1000 times faster drifting speed

Conventional Read-Verify Does Not Work for In-Memory Computing

- Accuracy Evolution





Drawbacks:

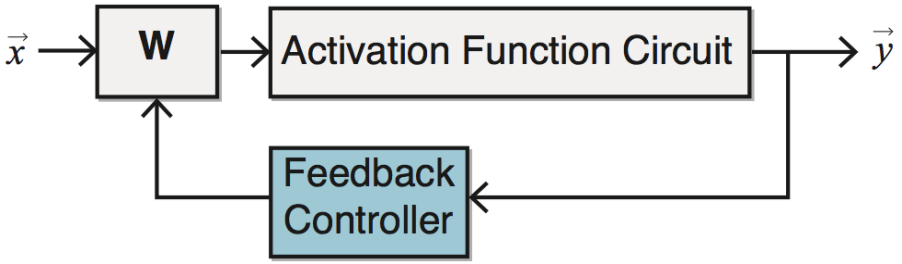
- Multilevel values

  —hard to sense

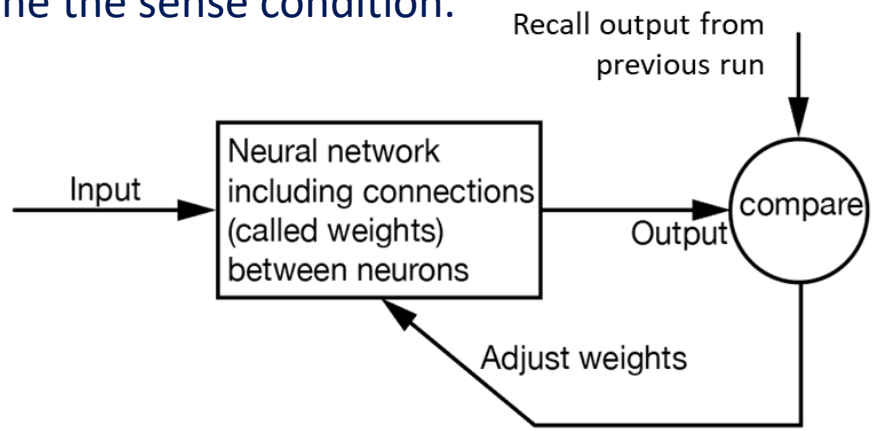
- Time costing
- Design overhead

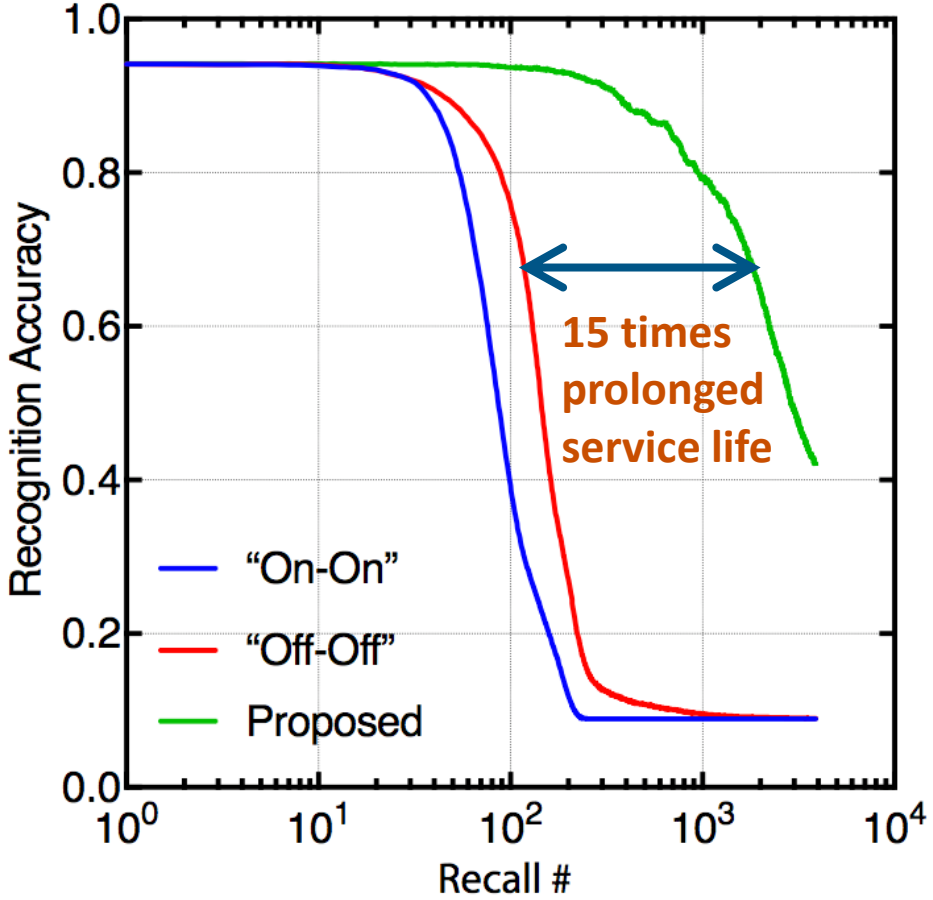
Closed-loop Design to Enhance Weight Stability

Feedback controller: Adjust the voltage condition to compensate the memristance drift.



"Arrogant principle": recall output is used as the label to determine the sense condition.





B. Yan, et al, ICCAD, 2017

Duke

Summary

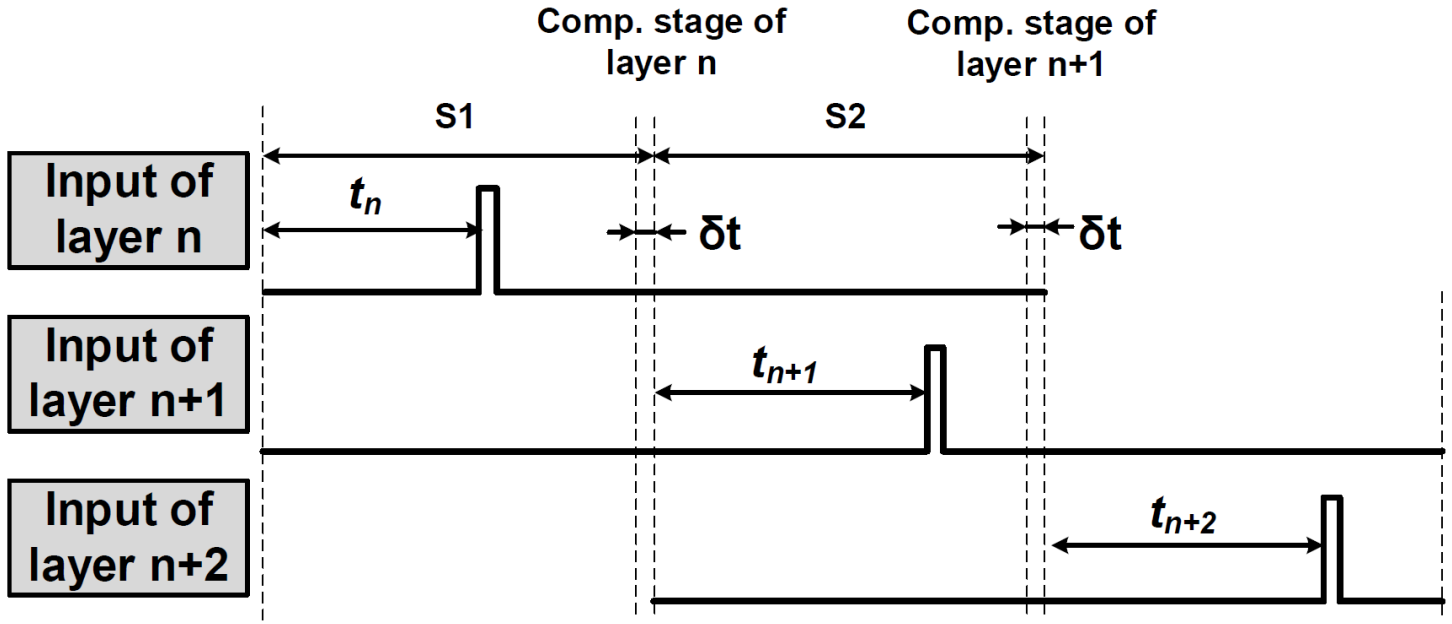
- The Past and Now of AI Hardware
  - In-Memory Computing Eliminates Weights Movement
- My Work:
  - Spiking-based In-Memory Computing Engine Offer Very High Energy Efficiency & Good Performance
  - Clever Design Methodologies (e.g., Closed-Loop Sensing) is Effective to Tolerate Nonideal Features of Memristors

Future Work I: Single Spike Processing Engine

Use Single Spike to Replace Multiple Spikes: ~60x Improvement of Energy Efficiency

MAC: $$y_j = \sum_{i=1}^{m} x_i \cdot w_{ji}$$

Single Spike: $$t_j = \sum_{i=1}^{m} t_i \cdot G_{ji}$$



"ReSiPE: ReRAM-based Single-Spiking Processing-In-Memory Engine"
Simulation Results Coming in July at Design Automation Conference (DAC) 2020

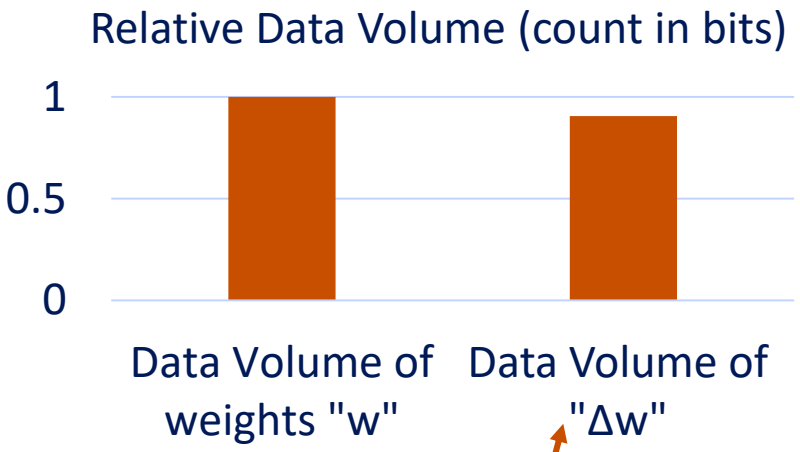
Expect to Tape-Out for Hardware Demonstration

Duke

Future Work II: In-Memory Compute & Cache

Challenge:

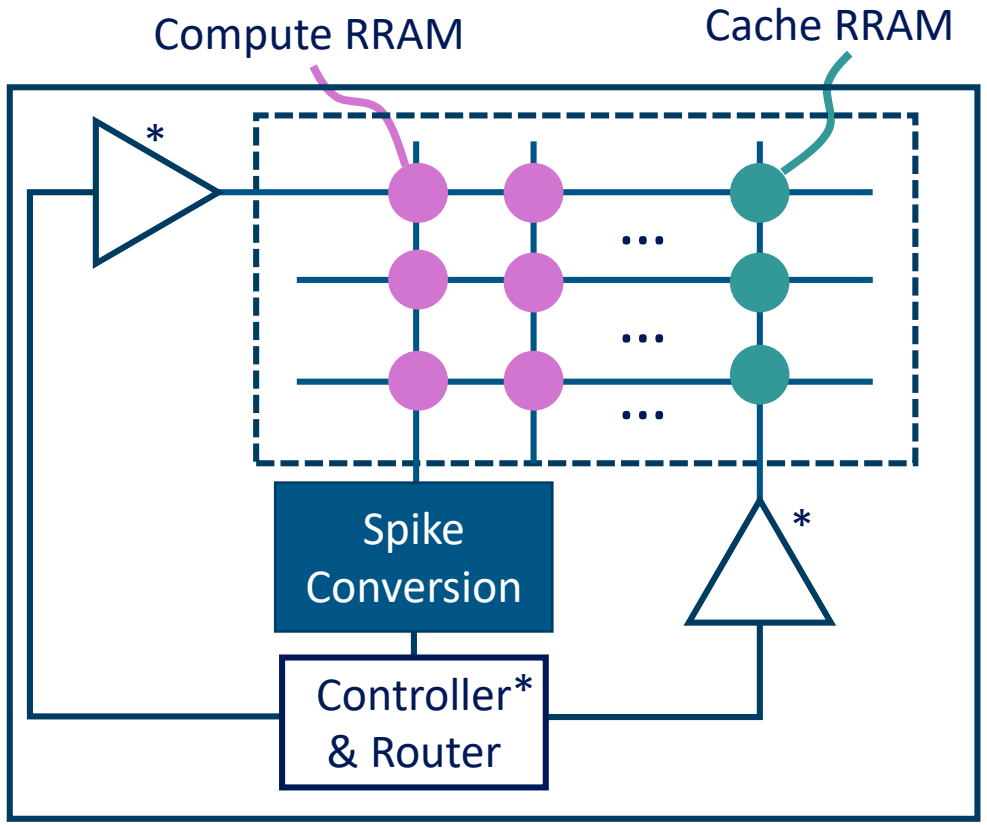
During Training:



No Need to Transport Weight

External DDRAM Access (Expensive!)

Attack:



*: novelty

Prospect Applications:

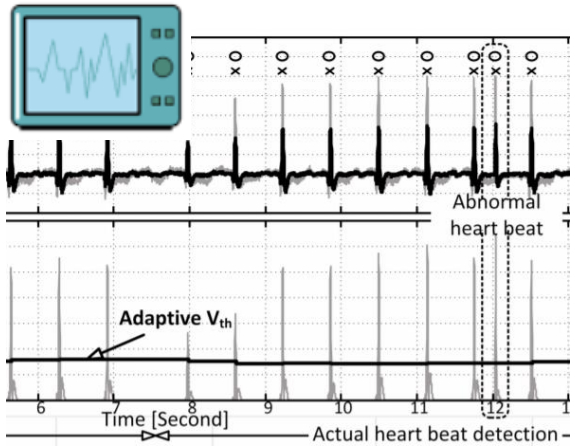
NN Training Acceleration

Self-Updatable In-Memory Computing Engine

Duke

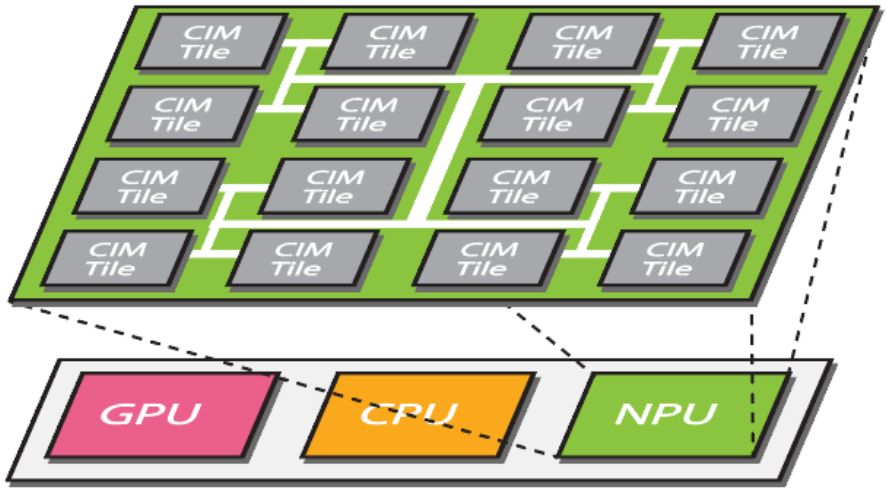
Long-Term Research Prospects

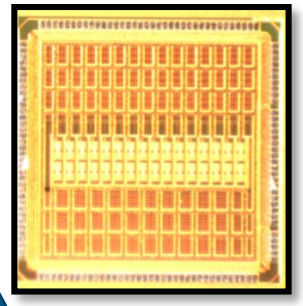
In Situ Data Processing
(Near-Sensor Computing)



Analog Processing

Ultra-Low Power Graphics Processing





Normally-Off Computing
(Memory-Empowered
Processor Evolution)

Nonvolatile

In-Memory
Compute &
Cache

Universal Memory
(One Memory for All)

Duke

Many Thanks for the Support!

Advisors:

Prof. **Hai "Helen" Li**

Prof. **Yiran Chen**

Collaborators:

Prof. **Jianhua (Joshua) Yang**
UMASS Amherst

Prof. **Meng-Fan Chang**
National Tsing-Hua Univ.

Prof. **Qiangfei Xia**
UMASS Amherst

Dr. **Qing Wu**
Air Force Research Laboratory

Prof. **Krishnendu Chakrabarty**
Duke University

Prof. **Weisheng Zhao**
Beihang University

Student Collaborators:

Ziru Li, **Qilin Zheng**, **Brady Taylor**

Duke

Thanks for Listening & Qs!

slides available at:

https://bonanyan.github.io/bn/

Duke