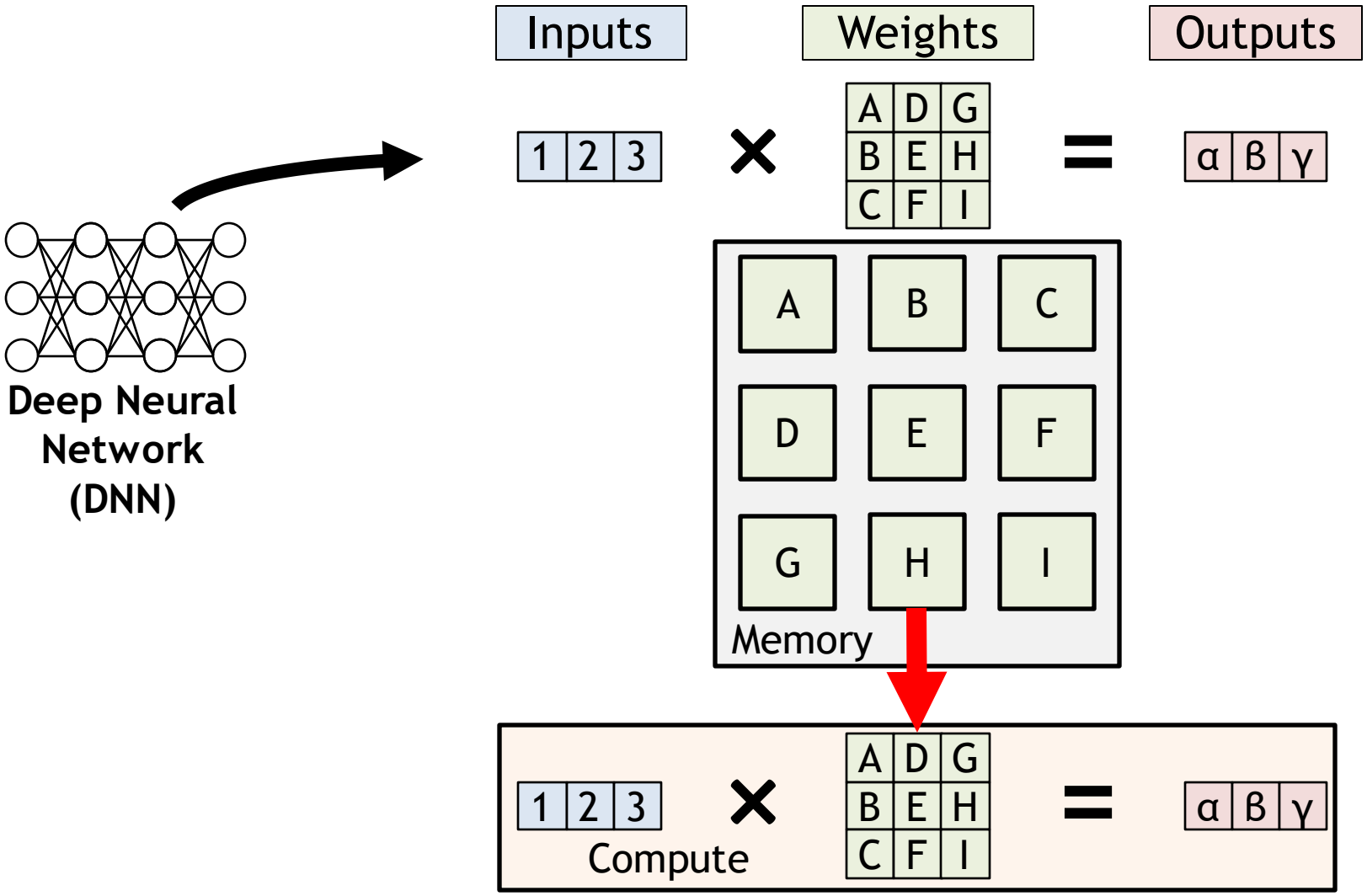
CiMLoop

A Flexible, Accurate, and Fast Compute-In-Memory Modeling Tool

Tanner Andrulis, Joel S. Emer, Vivienne Sze

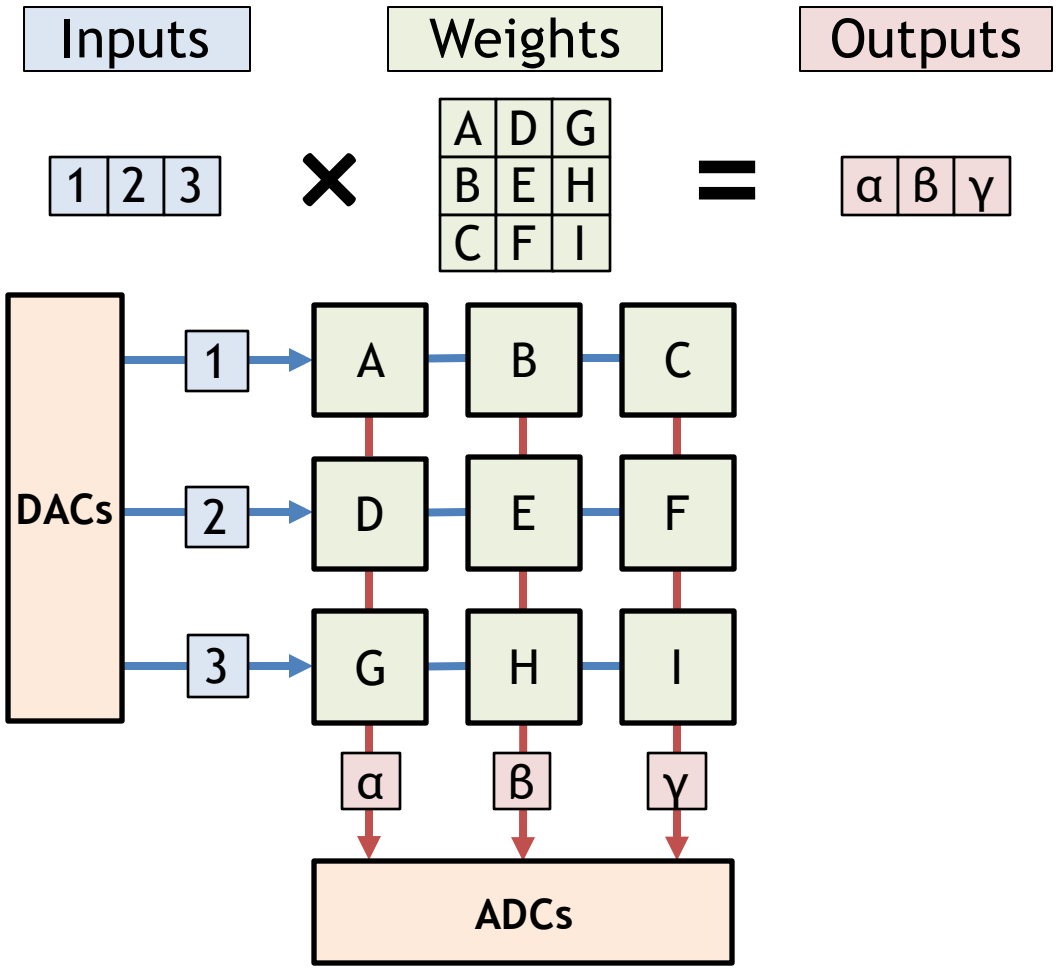
International Symposium on Performance Analysis of Systems and Software (ISPASS) 2024

Accelerating Matrix-Vector Operations



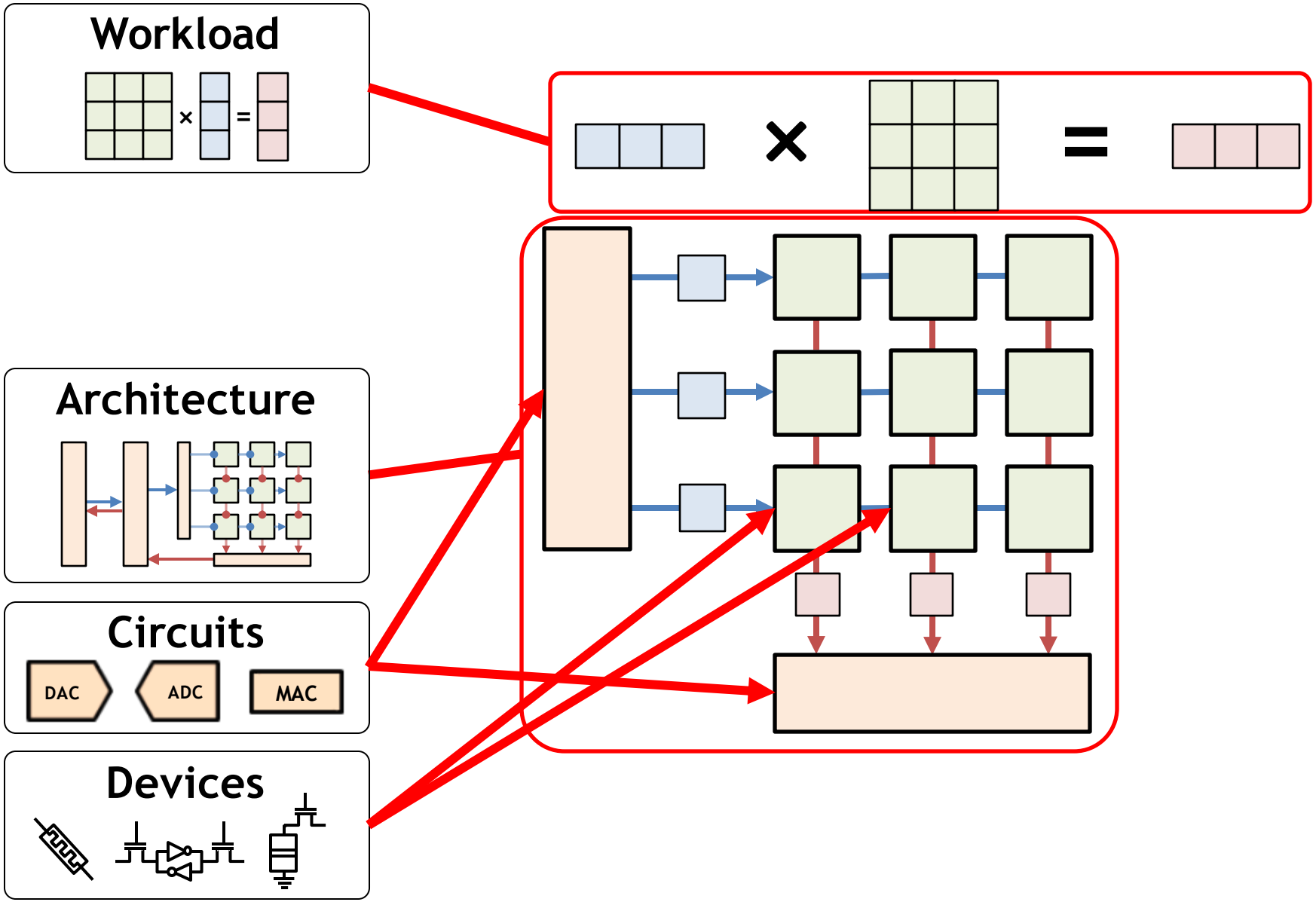
Significant energy spent moving DNN weights from memory

Compute-In-Memory (CiM)



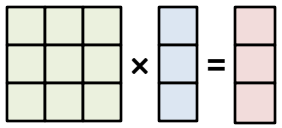
✅ No energy spent moving DNN weights
✅ Memory arrays can run many operations in parallel

The CiM Stack

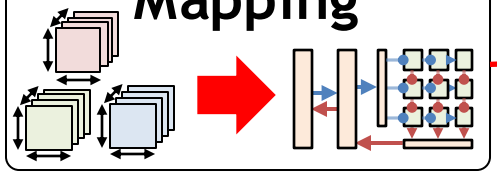


The CiM Stack

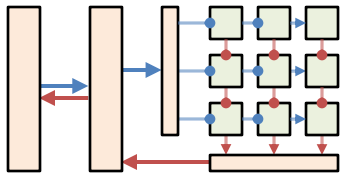
Workload



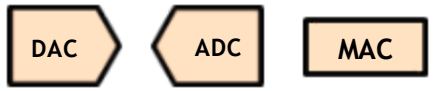
Mapping



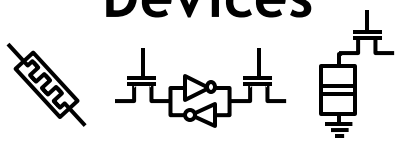
Architecture

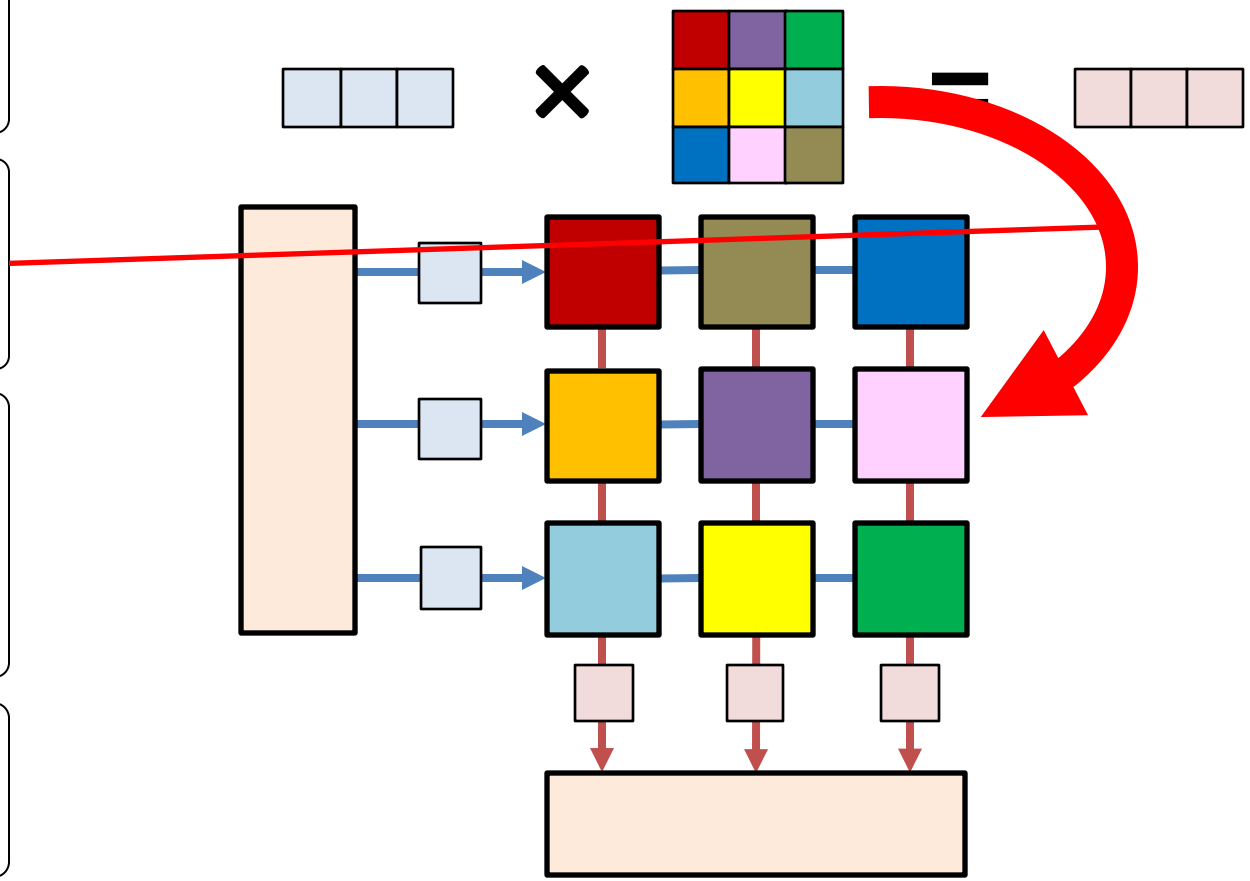


Circuits



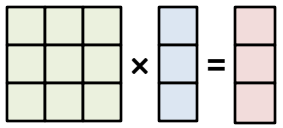
Devices



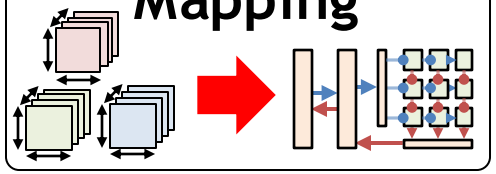


The CiM Stack

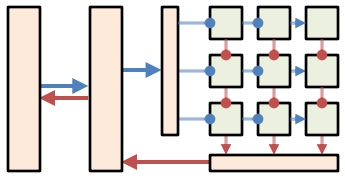
Workload



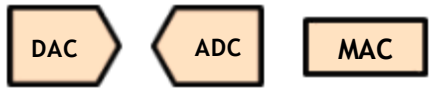
Mapping



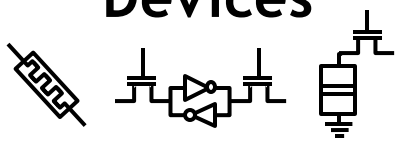
Architecture

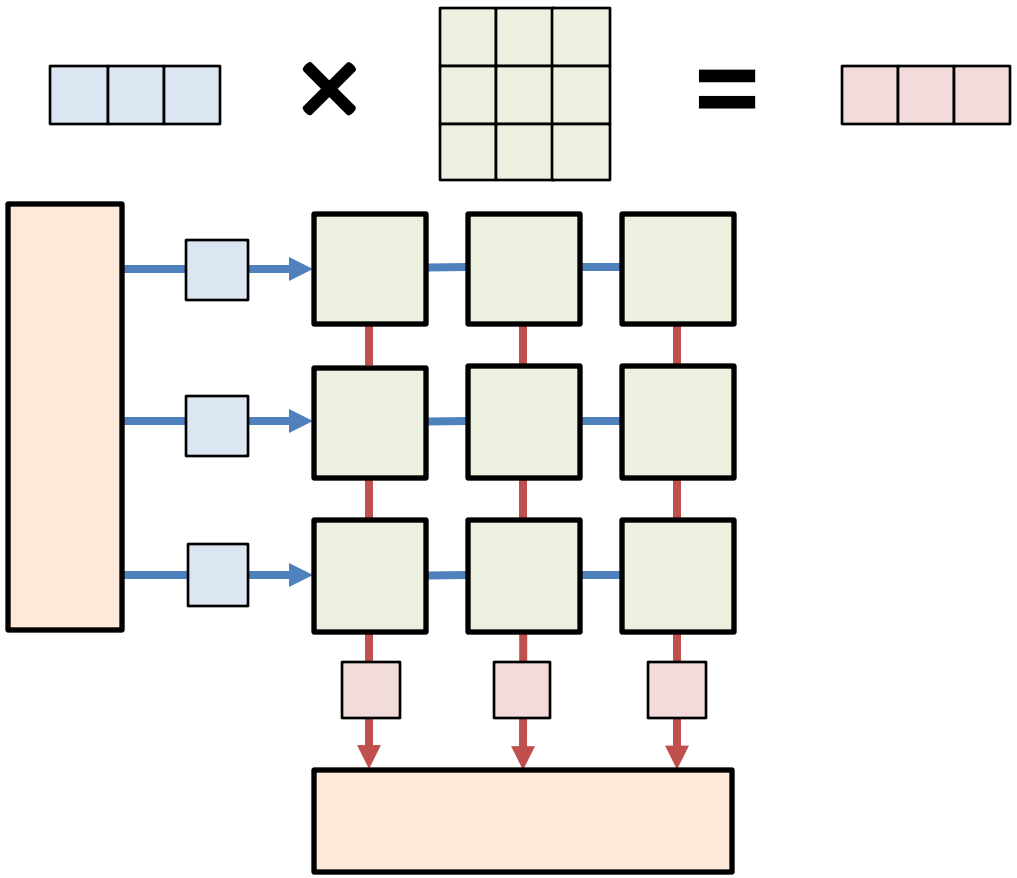


Circuits



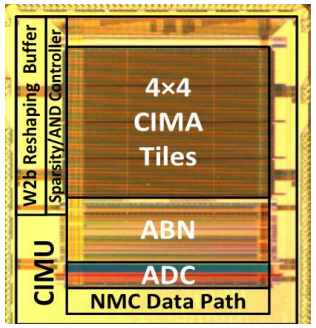
Devices



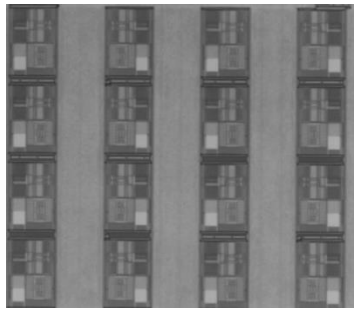


Opportunities in exploring all levels of the stack

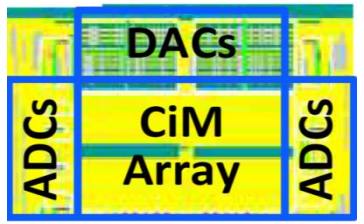
The CiM Stack



[Jia, JSSC 2020]



[Sinangil, JSSC 2021]



[Wang, VLSI 2022]

You'd like to find the most energy-efficient architecture

But published results have different…

Technology nodes

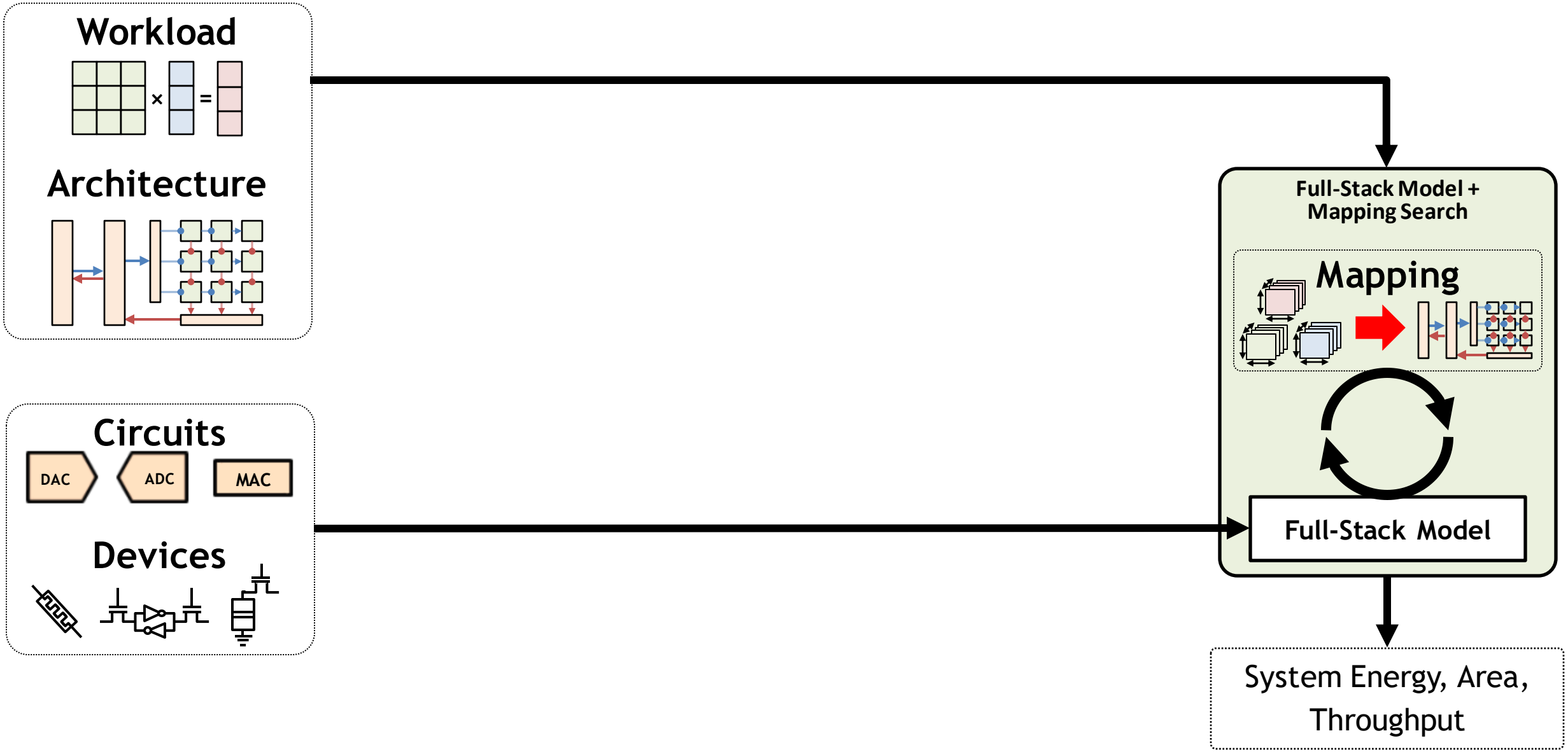
Devices

Workloads

Supported resolutions

Levels interact with one another!
Can't compare architectures based on published results alone.

Building a Modeling Framework



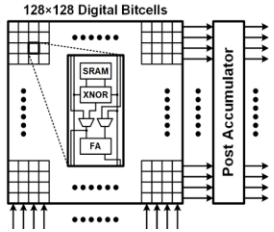
CiMLoop Goals

1. **Represent the co-design space**
   - **Challenge:** There are diverse choices at each level
   - **Solution:** Flexible user-defined specifications

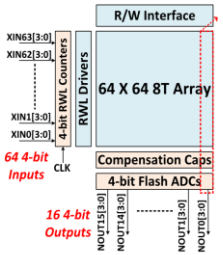
2. **Accurately model energy**
   - **Challenge:** Workload values and architecture representations affect circuit energy
   - **Solution:** Energy models that capture these cross-stack interactions

3. **Quickly explore the large co-design space**
   - **Challenge:** Accurate energy models may simulate many (>10^{12}) values
   - **Solution:** Statistical models that are 1000x faster than prior accurate models

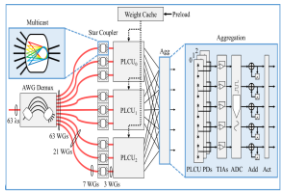
The Co-Design Space: Components



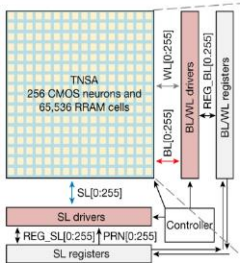
[Kim, JSSC 2021]



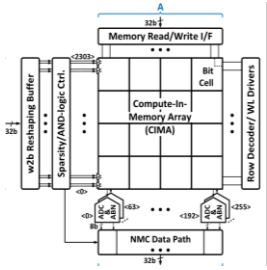
[Sinangil, JSSC 2021]



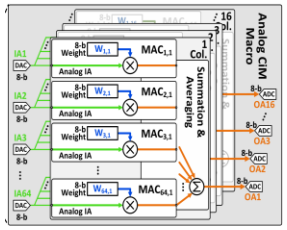
[Shiflett, ISCA 2021]



[Wan, Nature 2022]



[Jia, JSSC 2020]



[Wang, VLSI 2022]

Eight-way multiplexed NMC data path

Weighting Capacitors

Arrayed Waveguide Grating

Pulse Train DAC

9T SRAM

Post Accumulator

SRAM + Digital MAC

C-2C Multiplier

Controller

8T SRAM

Photodiodes

4b Flash ADC

Asynchronous SAR ADC

Column Registers

R-2R DAC

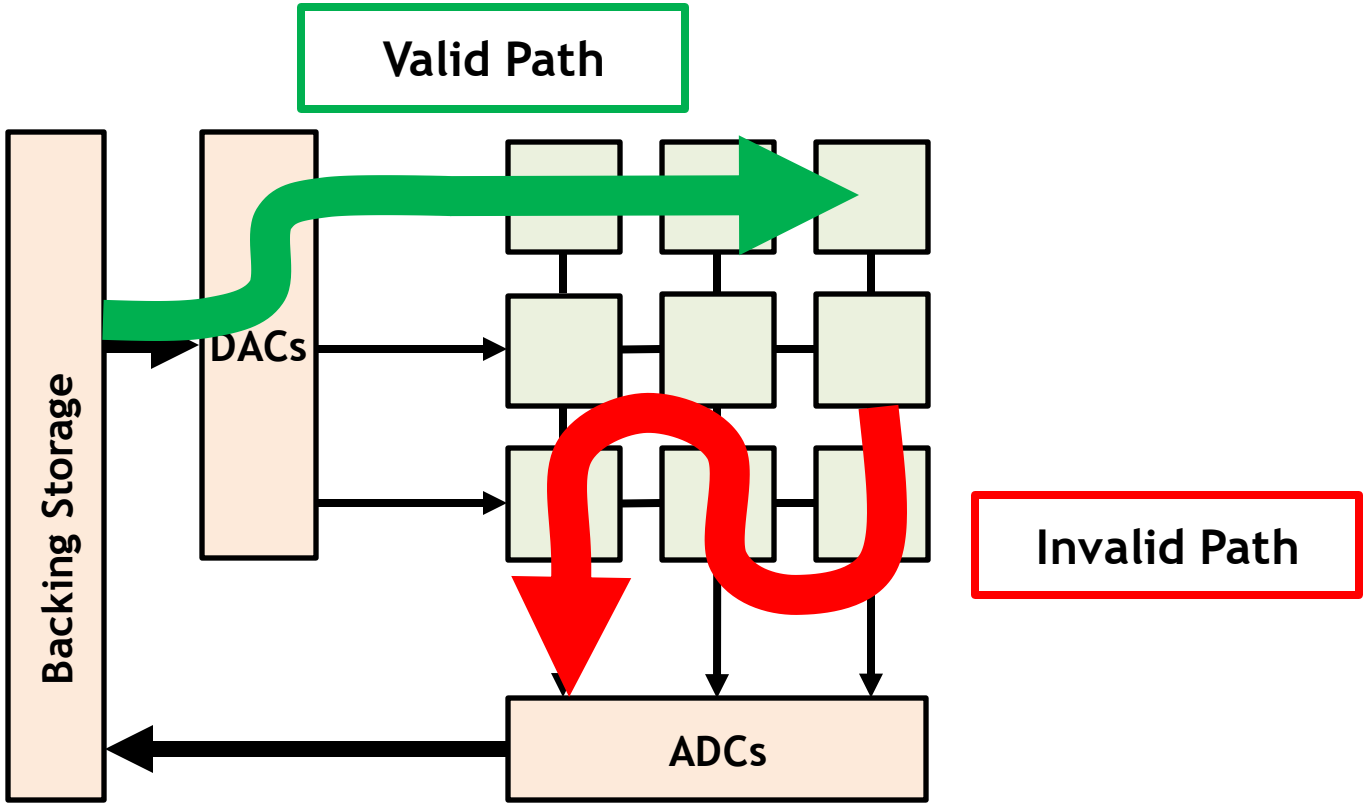
6T SRAM

Library of circuit and device models

+

Plug-in interface for users to create more models

The Co-Design Space: Connections



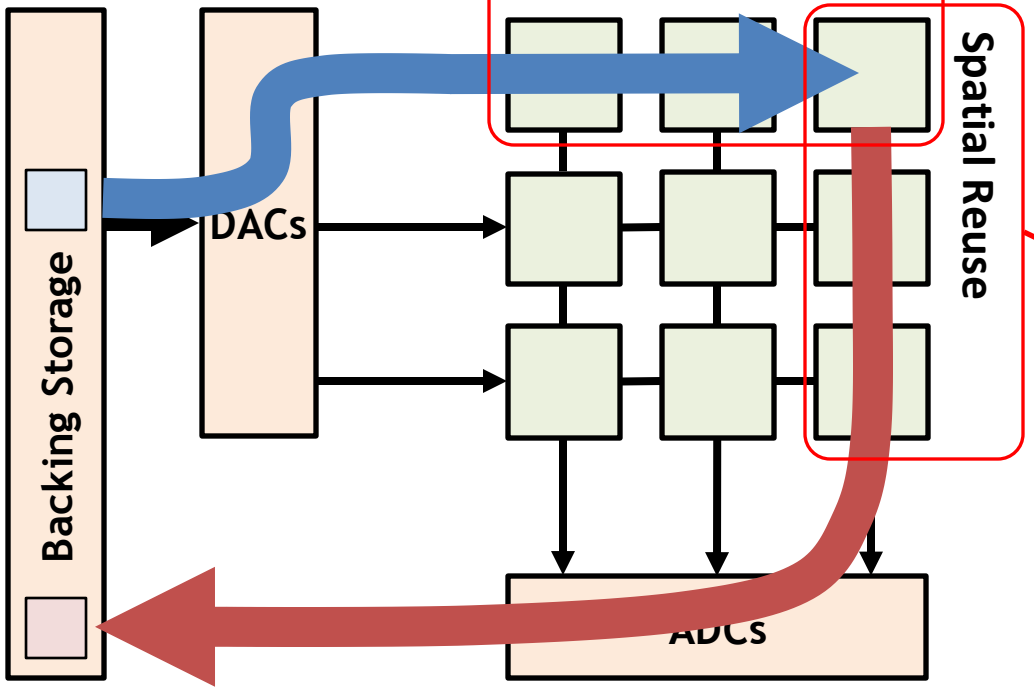
Must define how data may move through the system

The Co-Design Space: Connections

Benefit: 3× Memory cells compute with one input

Cost: 1× Storage read

Cost: 1× DAC convert



Benefit: 3× Memory cells compute with one output

Cost: 1× ADC convert

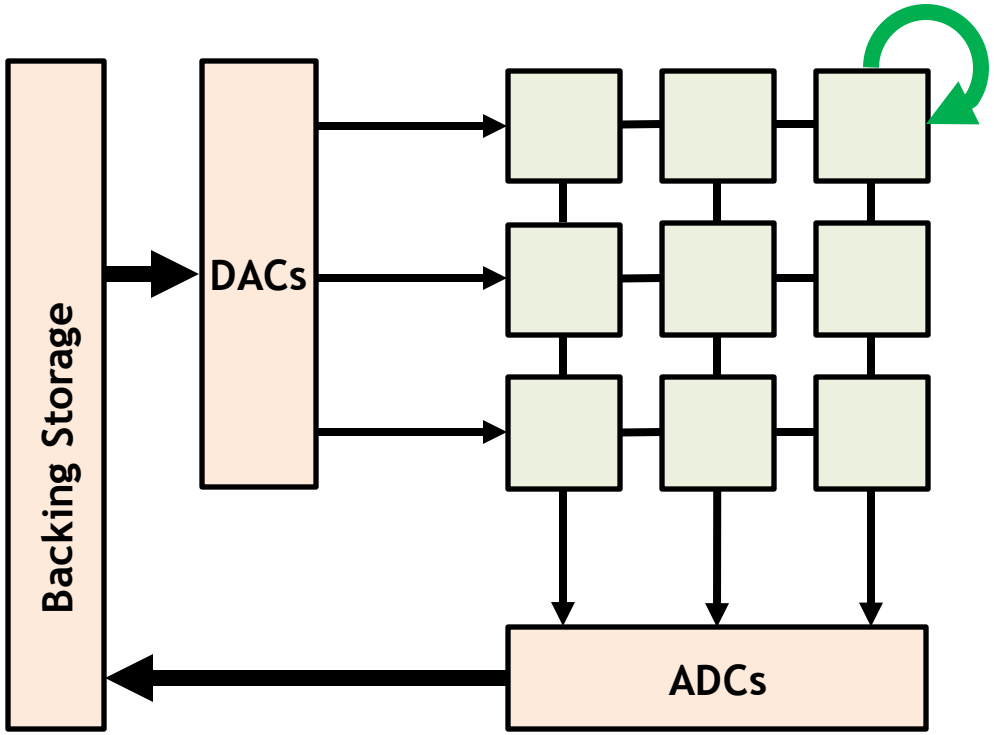
Cost: 1× Storage write

Spatial reuse uses one value across parallel components

More ~~bang~~ for your ~~buck~~!

computations ADC/DAC convert & memory access

The Co-Design Space: Connections



Benefit: ***N*×** timesteps compute with one weight

Cost: **1×** Initial storage write

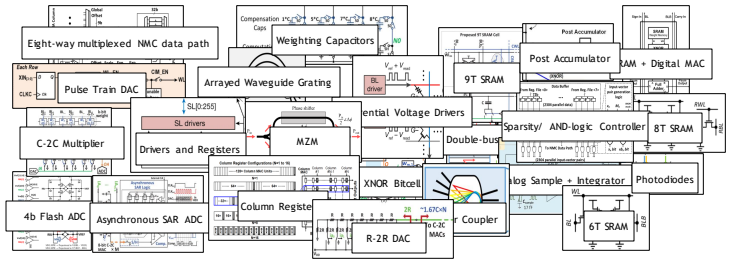
Temporal reuse uses one value across timesteps

More ~~bang~~ for your ~~buck~~!

computations initial weight write

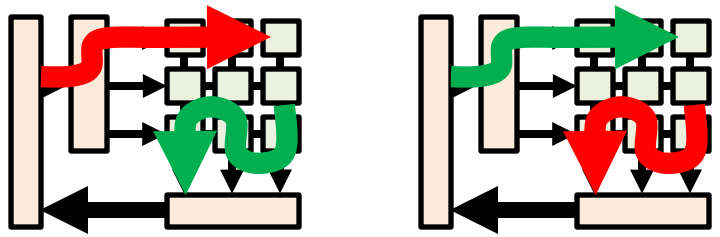
The Co-Design Space: Connections

Define circuits and devices (Components)

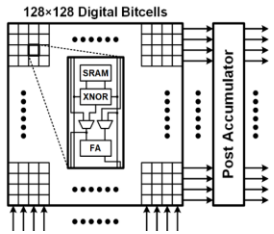


+

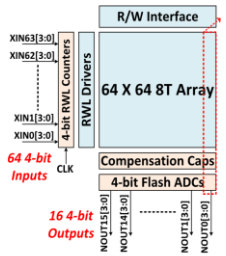
Define permitted reuse patterns (Connections)



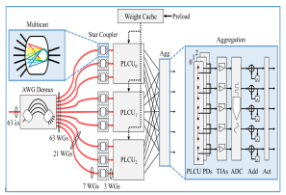
CiMLoop can represent diverse CiM designs
We provide open-source models of six different works



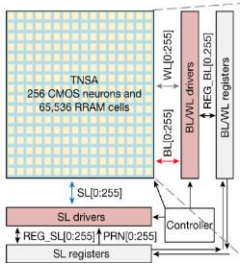
[**Kim**, JSSC 2021]



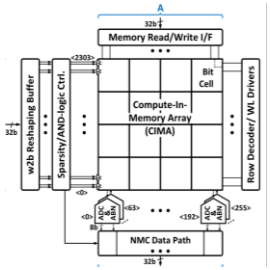
[**Sinangil**, JSSC 2021]



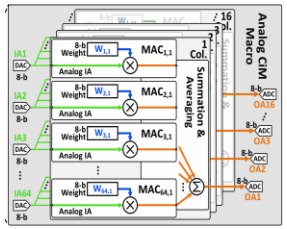
[**Shiflett**, ISCA 2021]



[**Wan**, Nature 2022]

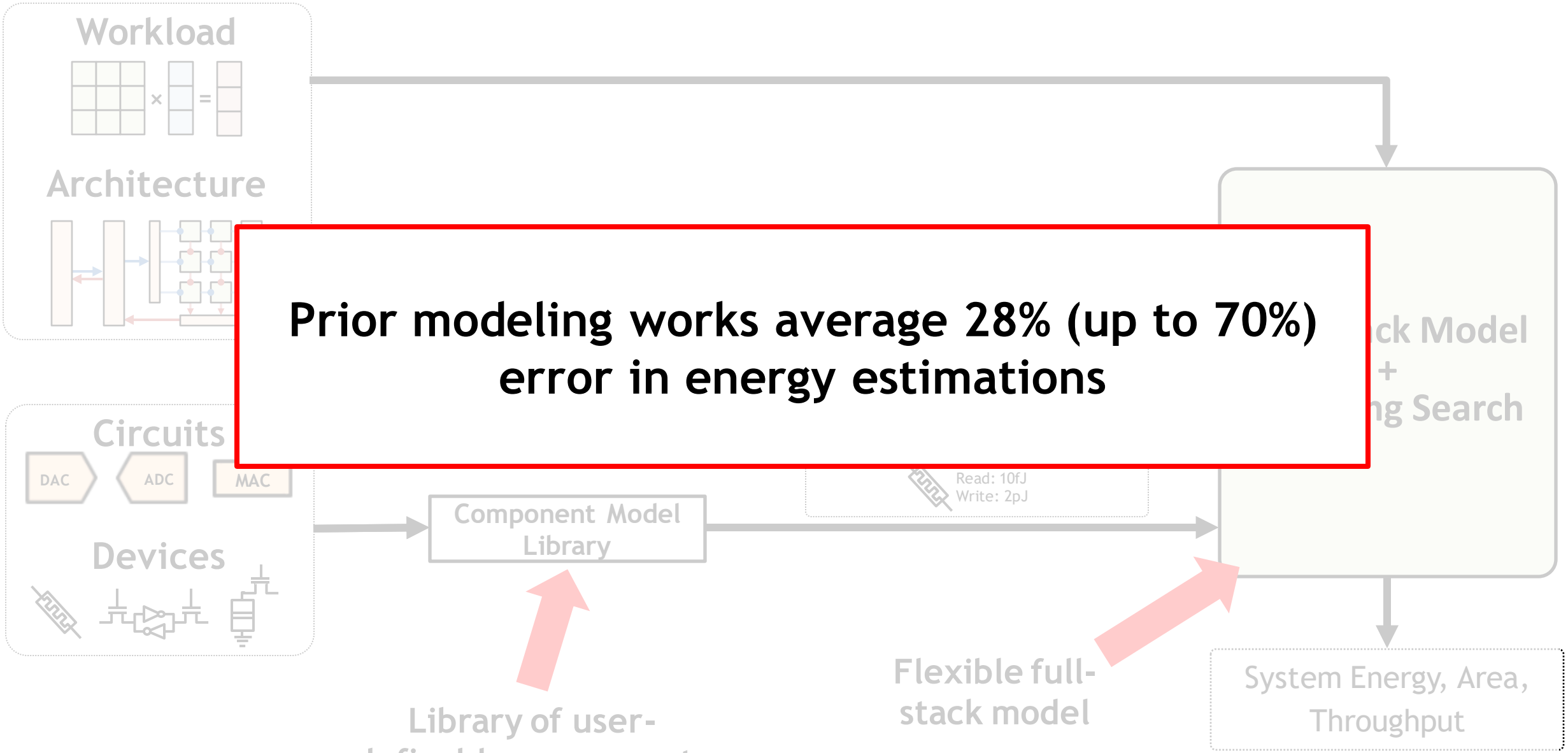


[**Jia**, JSSC 2020]



[**Wang**, VLSI 2022]

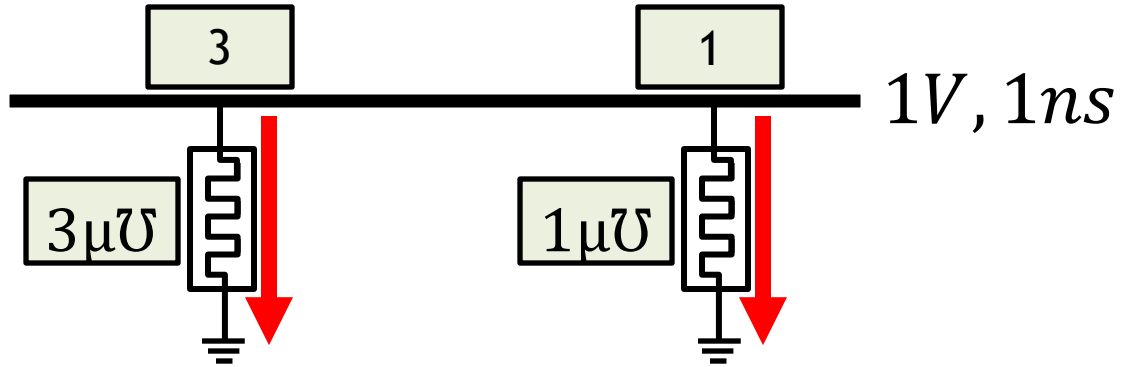
Building a Modeling Framework



Accurately Modeling Energy: Data-Value-Dependence

$$Voltage^2 \times Conductance \times Time$$

$$Conductance\ \mho = \frac{1}{Resistance\ \Omega}$$



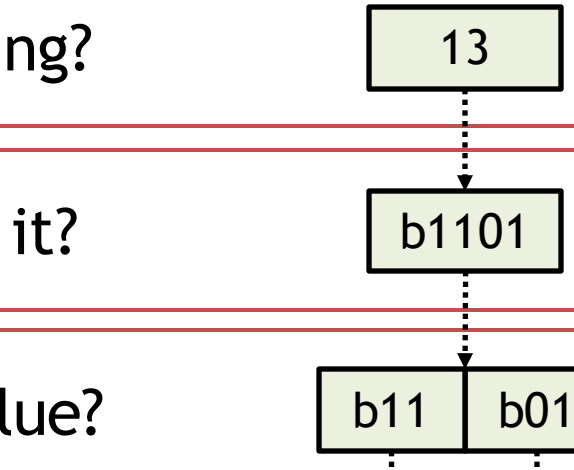
Data-value-dependence significantly impacts device and circuit energy
Prior works assume fixed energy → significant error

Accurately Modeling Energy: Data-Value-Dependence

What DNN value are we processing?

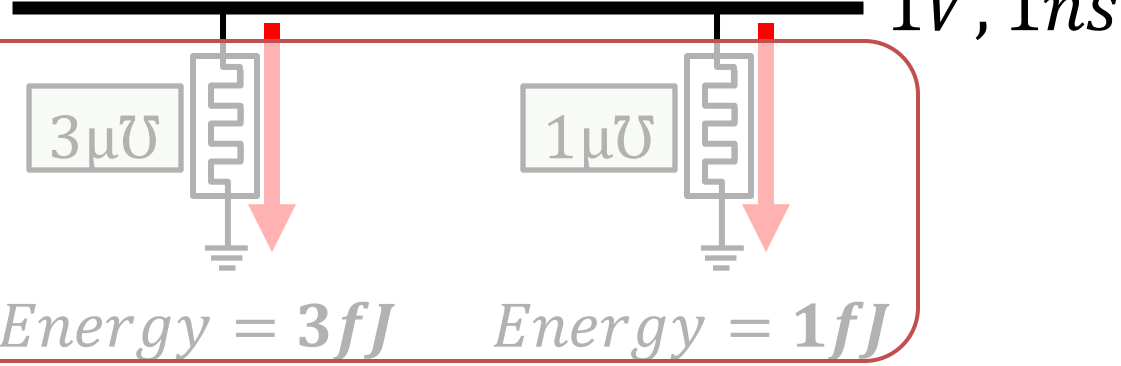
How does the system represent it?

Where do we map bits of this value?



Many encodings possible! Unsigned, differential, XNOR, 2's comp…

Partition bits across components



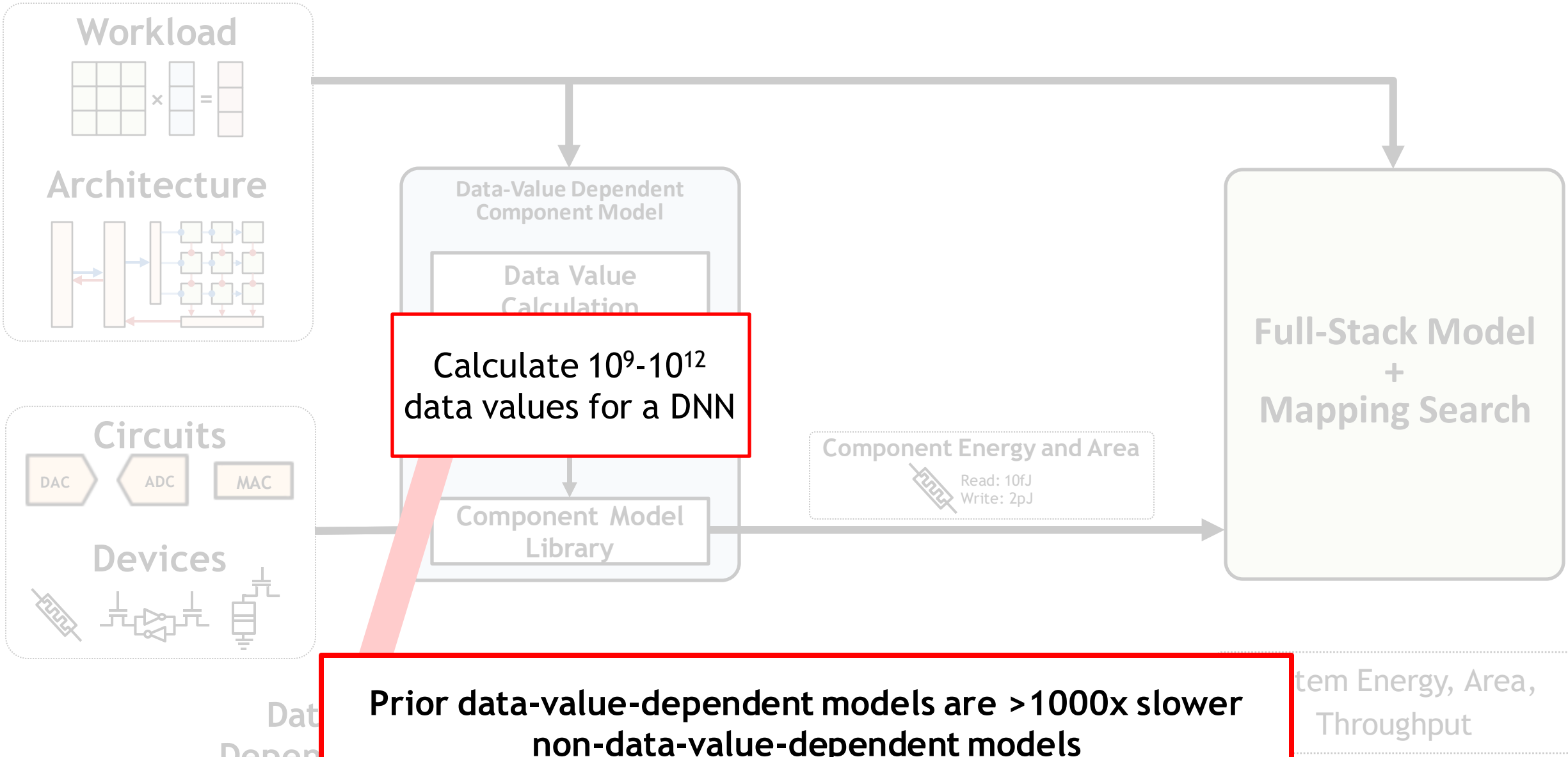
Calculate Energy

$$Voltage^2 \times Conductance \times Time$$

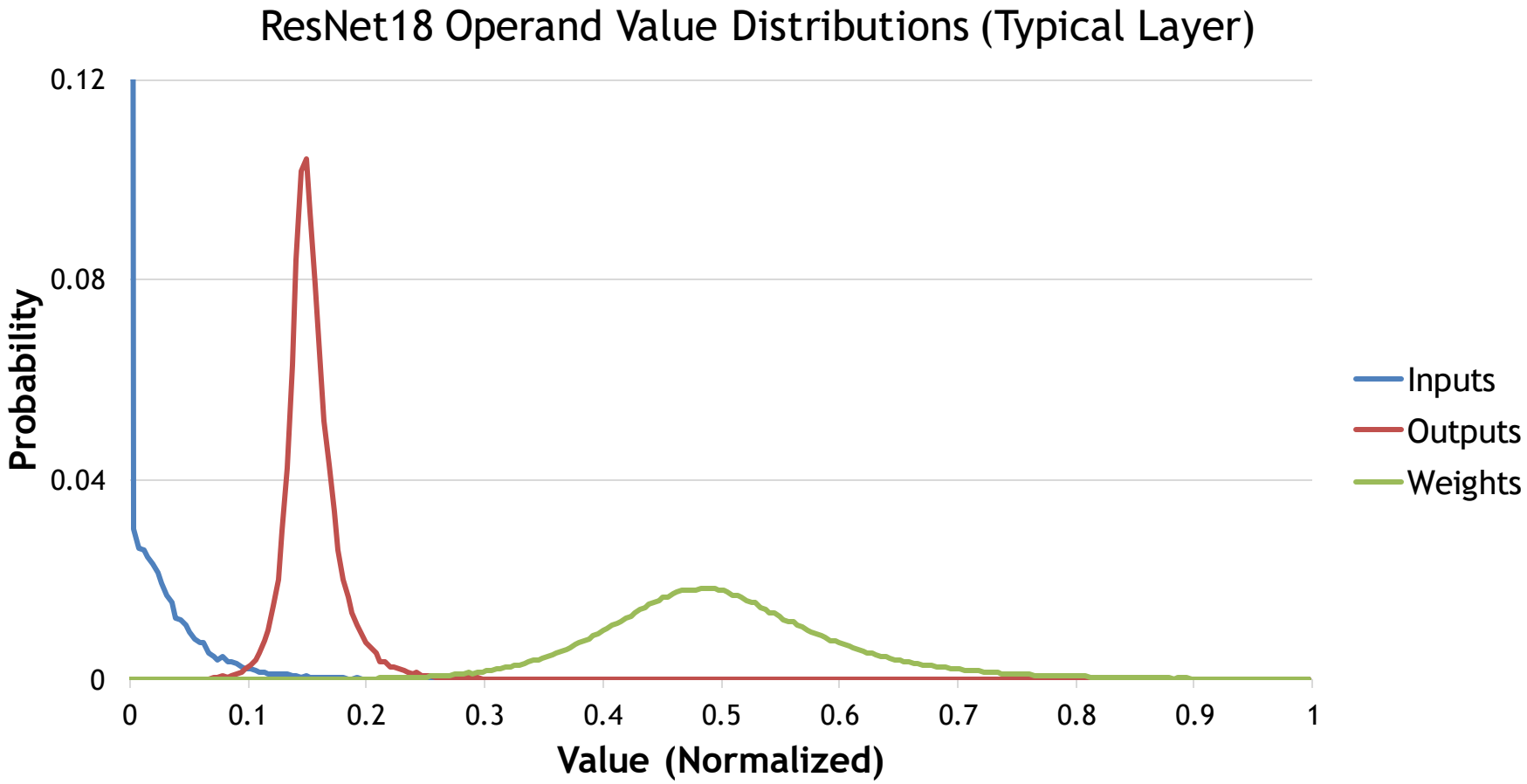
$$Conductance\ \mho = \frac{1}{Resistance\ \Omega}$$

Capture data-value-dependence:
What values are there? ***How*** do we represent them? ***Where*** do we map their bits?

Building a Modeling Framework



Quickly Modeling Energy: Distributions

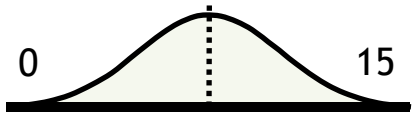


DNN operand values follow predictable distributions

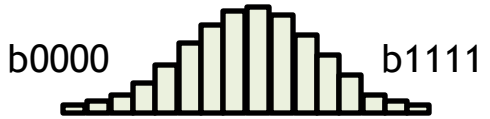
Use distributions to quickly and accurately model energy

Quickly Modeling Energy: Data-Distribution-Dependence

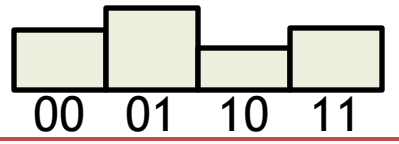
What distribution are we processing?

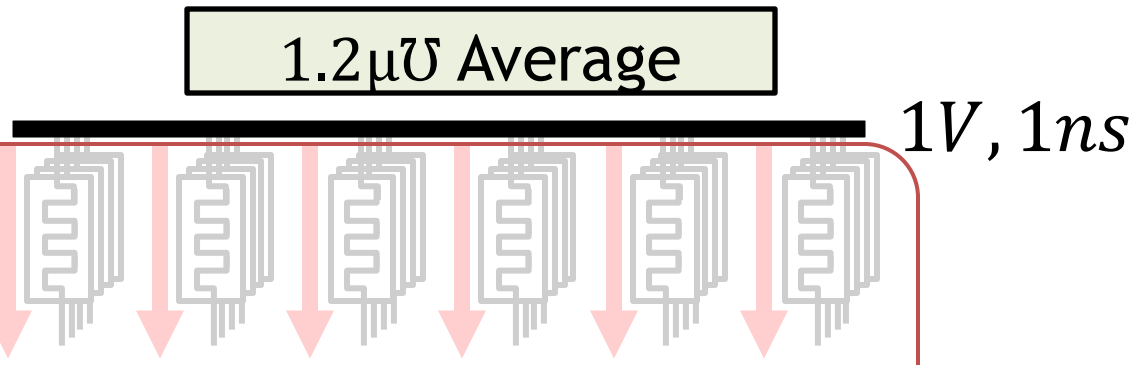


How does the system represent it?



Where do we map distribution bits?





Calculate Average Energy

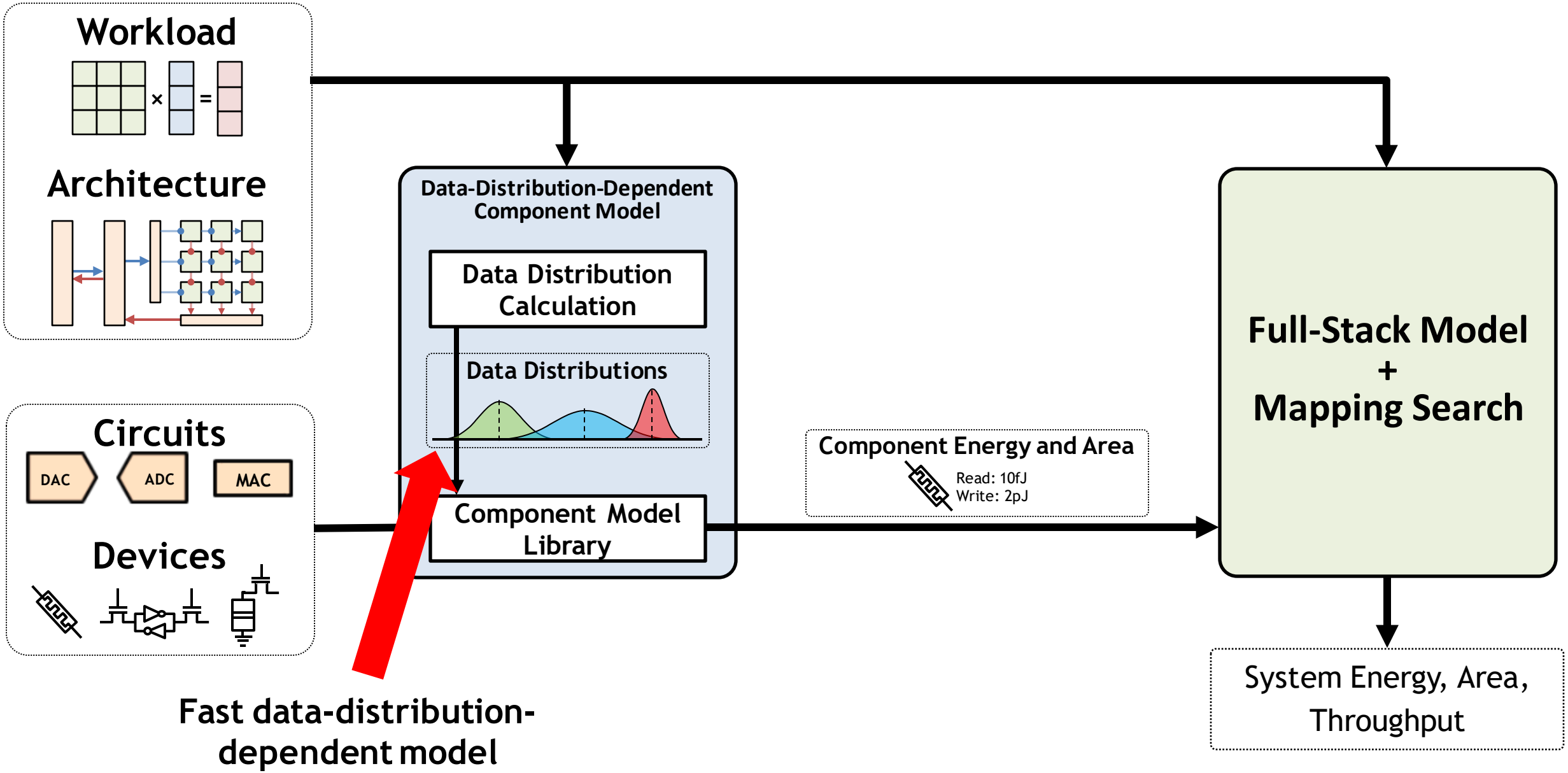
$$Voltage^2 \times Conductance \times Time$$

$$Conductance\ \mho = \frac{1}{Resistance\ \Omega}$$

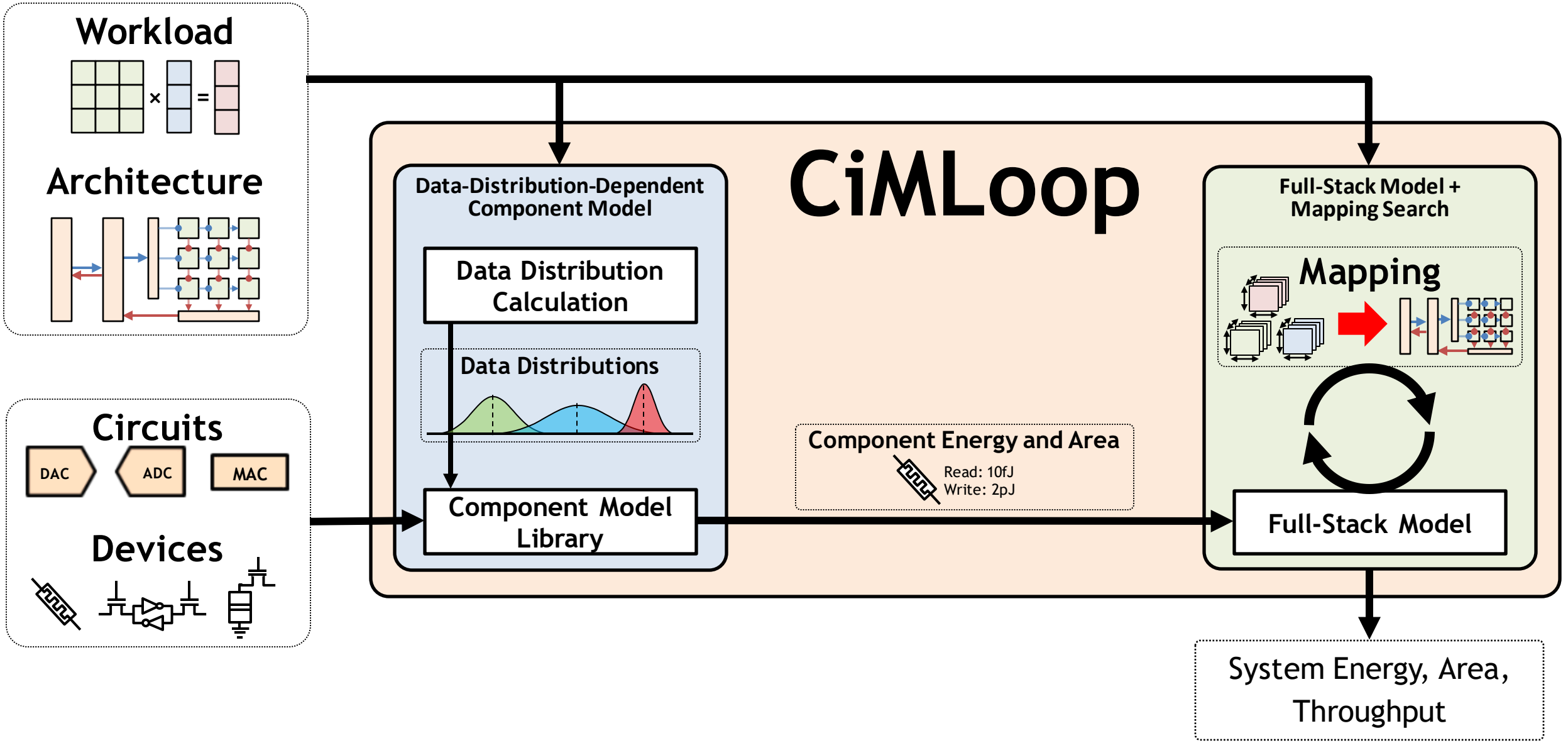
$$Avg.Energy = 1.2\frac{fJ}{Read},\ 10^6 \text{ reads} \rightarrow 1.2\ \boldsymbol{nJ}$$

Answer *what*, *how*, *where* for distributions → One calculation for any number of reads

Building a Modeling Framework

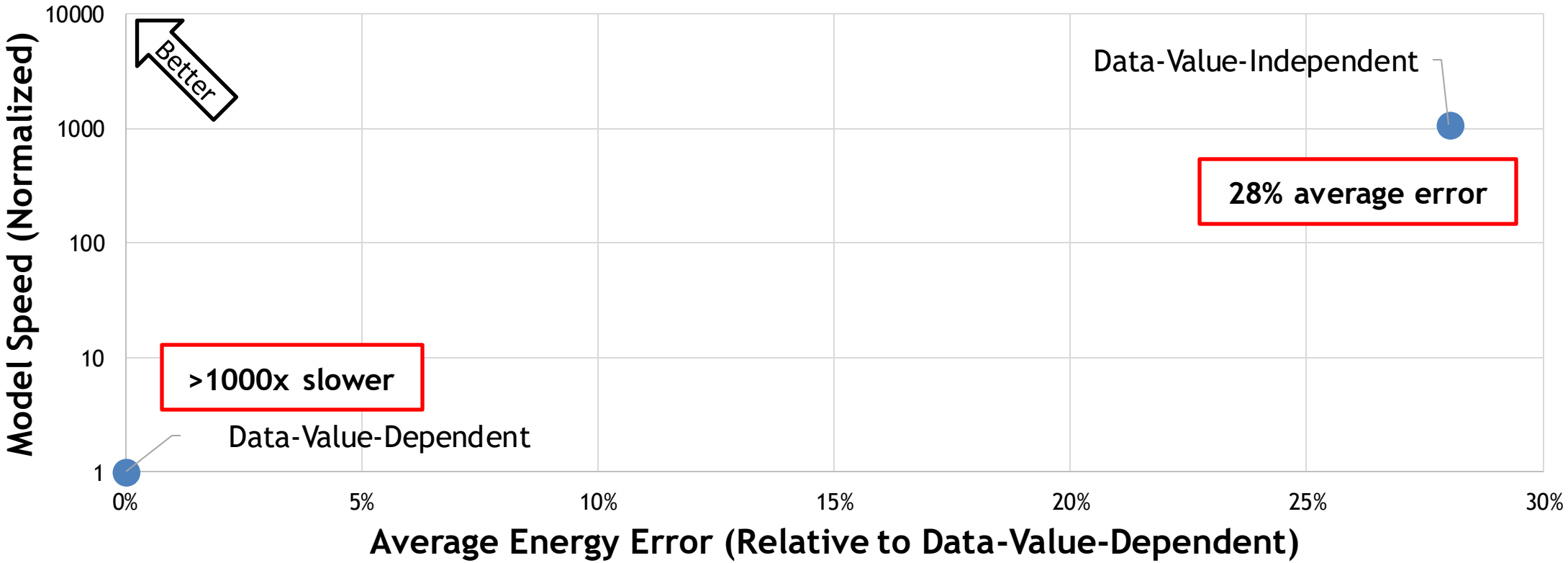


Building a Modeling Framework



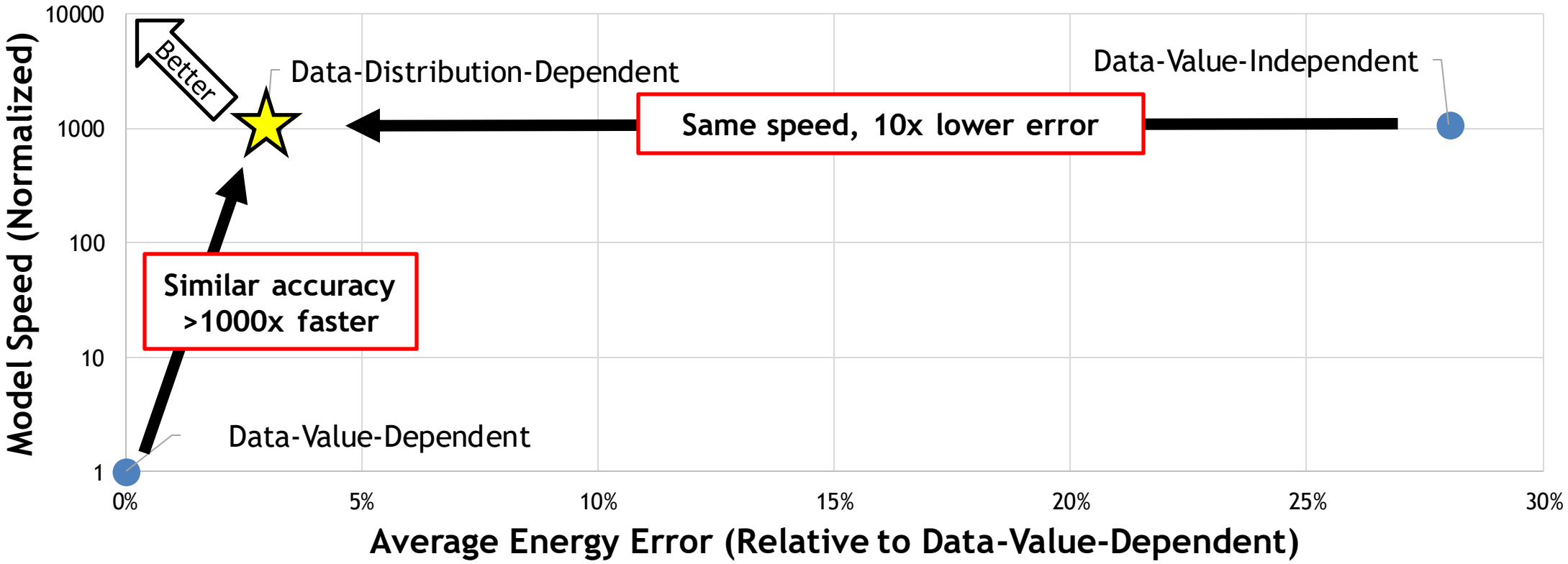
Fast Statistical Energy Modeling

	Data-Value-Dependent NeuroSim [Peng, TCAD 2021]	Data-Value-Independent Timeloop [Parashar, ISPASS 2019]
Model Speed	Low	High
Model Accuracy	High	Low

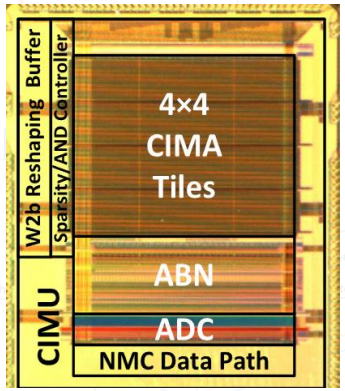


Fast Statistical Energy Modeling

	Data-Value-Dependent NeuroSim [Peng, TCAD 2021]	Data-Value-Independent Timeloop [Parashar, ISPASS 2019]	Data-Distribution-Dependent CiMLoop [This Work]
Model Speed	Low	High	High
Model Accuracy	High	Low	High

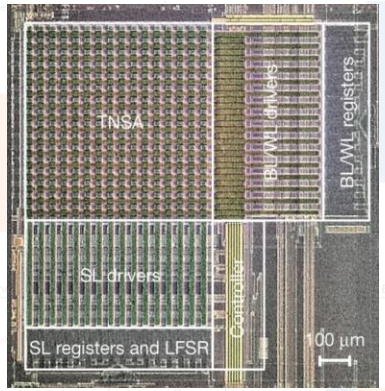


CiMLoop Validation

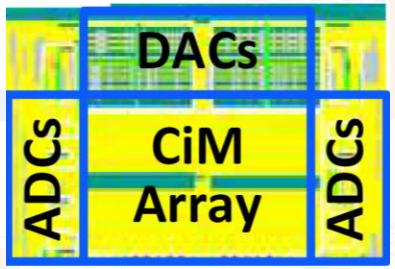


Design A
[Jia, JSSC 2020]

Design B
[Sinangil, JSSC 2021]



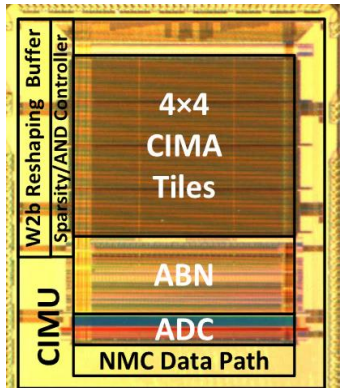
Design C
[Wan, Nature 2022]



Design D
[Wang, VLSI 2022]

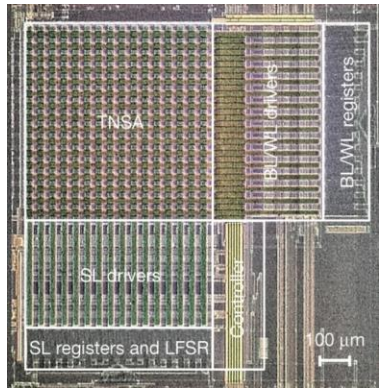
Validated against four fabricated CiM publications
with unique devices, circuits, architecture, workloads, and mapping

CiMLoop Validation

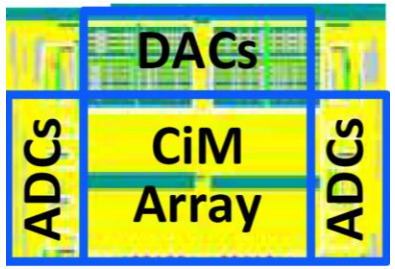


Design A
[**Jia**, JSSC 2020]

Design B
[**Sinangil**, JSSC 2021]



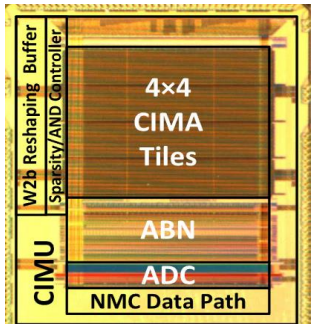
Design C
[**Wan**, Nature 2022]



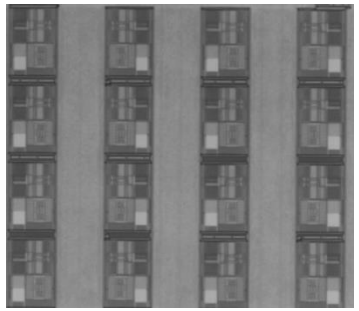
Design D
[**Wang**, VLSI 2022]

CiMLoop Average Error			
	Supply Voltage Sweeps	Bit Precision Sweeps	Per-Component Breakdowns
Energy	**7%**	**6%**	**4%**
Throughput	**2%**	**5%**	N/A
Area	N/A	N/A	**8%**

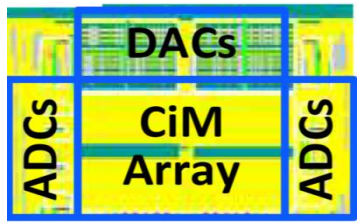
Using CiMLoop: Compare Designs



[Jia, JSSC 2020]



[Sinangil, JSSC 2021]



[Wang, VLSI 2022]

You'd like to find the most energy-efficient architecture

But published results have different…

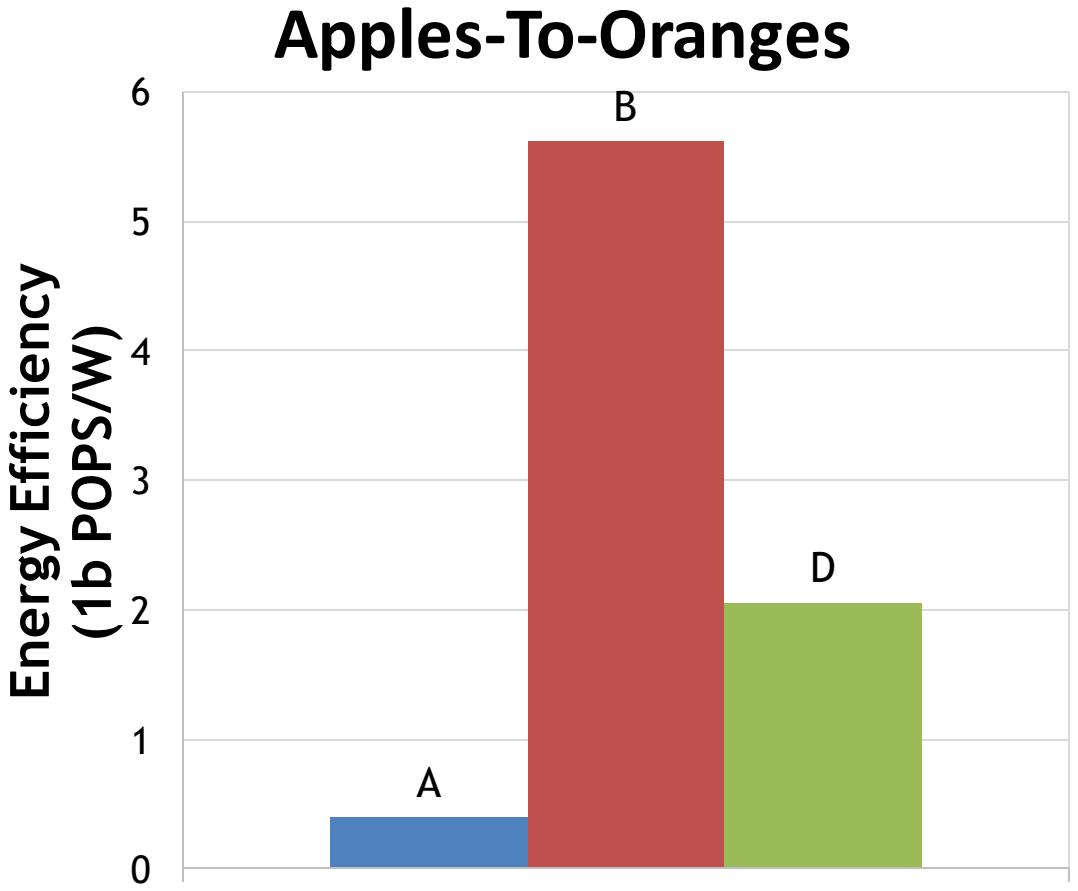
Technology nodes

Devices

Workloads

Supported resolutions

Using CiMLoop: Compare Designs



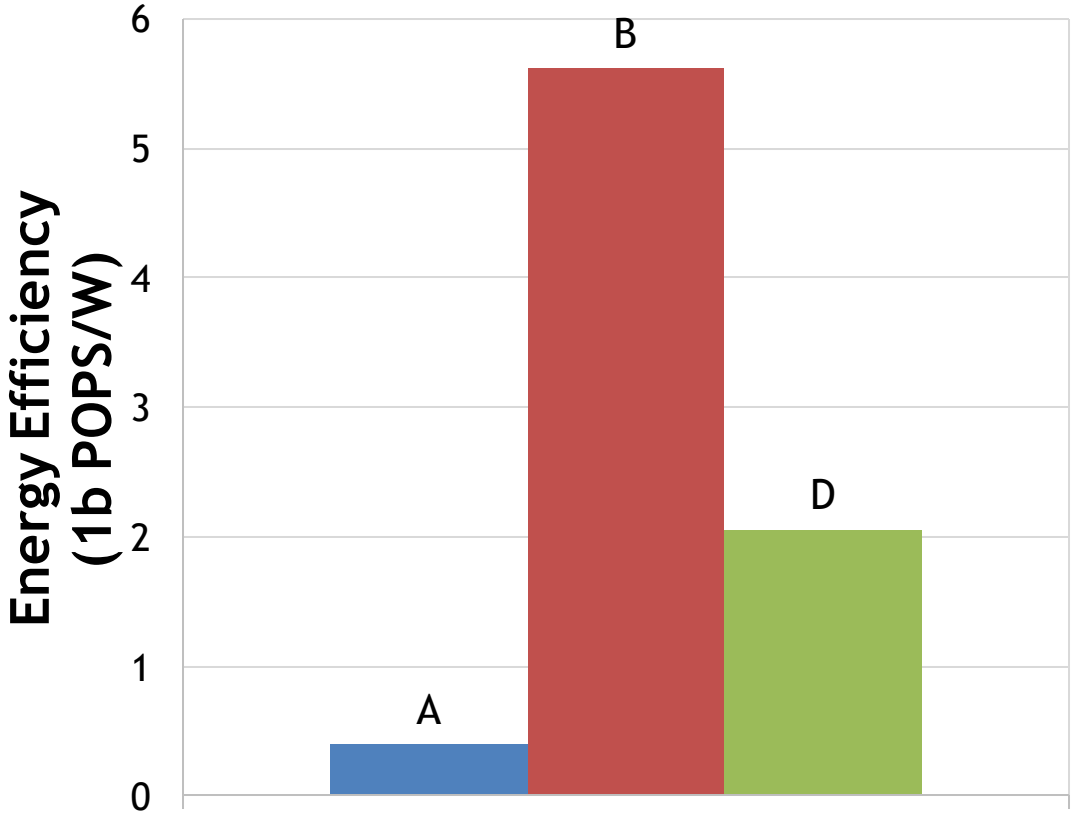
MISLEADING takeaway:
B architecture has best energy efficiency

For all architectures, we:

- **(Devices)** Use the same devices
- **(Circuits)** Scale to 7nm technology node
- (**Circuits/Arch.**) Use the same 8-bit ADC
- **(Workload)** Run the same workload
- **(Arch./Mapping)** Set up the design to support 8-bit computations

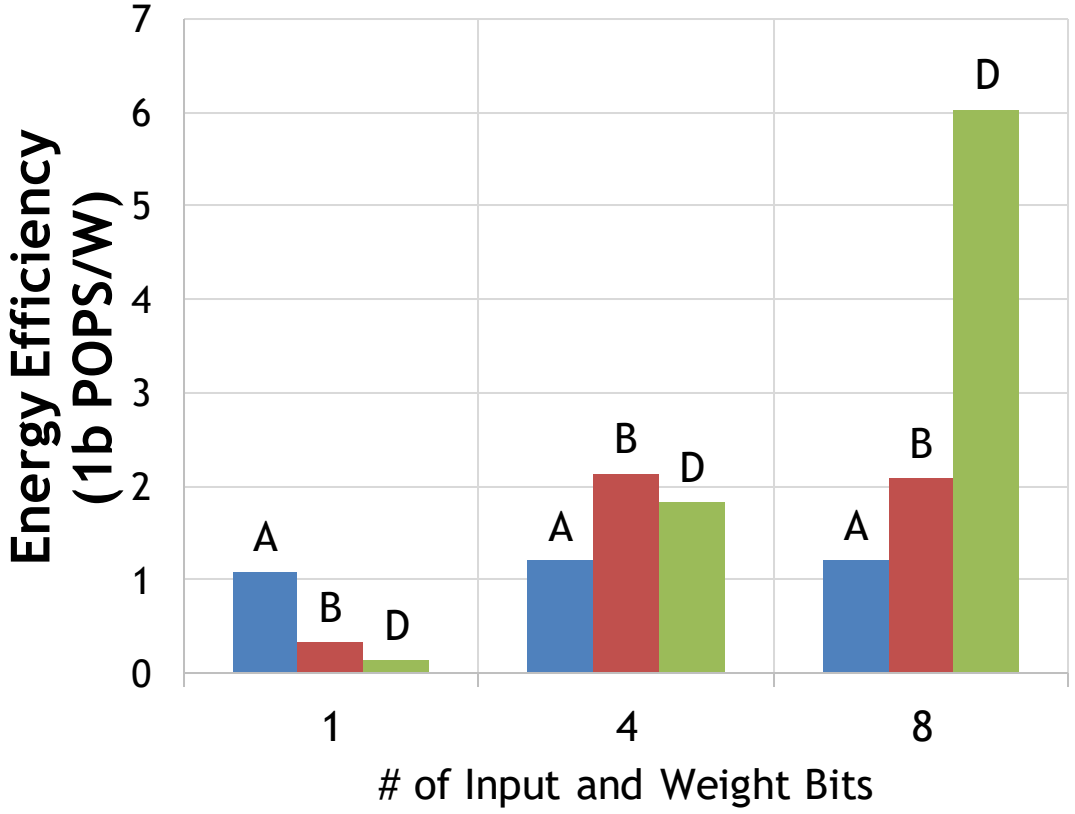
Using CiMLoop: Compare Designs

Apples-To-Oranges



MISLEADING takeaway:
B architecture has best energy efficiency

Apples-To-Apples



ACCURATE takeaway:
Lowest-energy choice depends on workload

Summary and Open-Source Models

- **CiMLoop introduces:**
  - Flexible models to explore devices, circuits, architecture, workload, and mapping
  - Accurate data-distribution-dependent energy modeling (10x lower error)
  - Fast statistical energy models (1000x faster)

- **CiMLoop is open-source and out now! Includes:**
  - Models of published works: 5 CiM designs
  - Full architectures
  - Devices (ReRAM and SRAM) and circuits (component library)
  - DNNs (CNNs and Transformers)
  - Bonus: 1 photonic computing design

CiMLoop tutorials and examples

https://github.com/mit-emze/cimloop

This work was funded in part by Ericsson, TSMC, the MIT AI Hardware Program, and MIT Quest.