Heterogeneous Chiplet-based Architecture for In-Memory Acceleration of DNNs

Gokul Krishnan[1]*, Sumit K. Mandal[2], Chaitali Chakrabarti[1]
,Jae-sun Seo[1], Umit Y. Ogras[2], Yu Cao[1]

[1]Arizona State University, USA
[2]University of Wisconsin-Madison, USA
***Email: gkrish19@asu.edu, Yu.cao@asu.edu**

Agenda

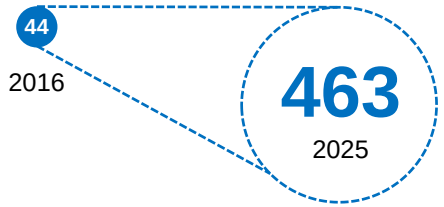
- Introduction
- Motivation
- SIAM Benchmarking Tool
- Big-Little Heterogeneous Architecture
- Experiments and Results
- Conclusion

Everything Goes Up

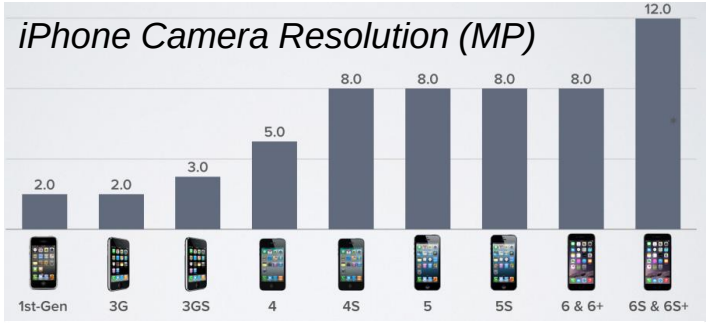
- From data volume to information processing algorithms

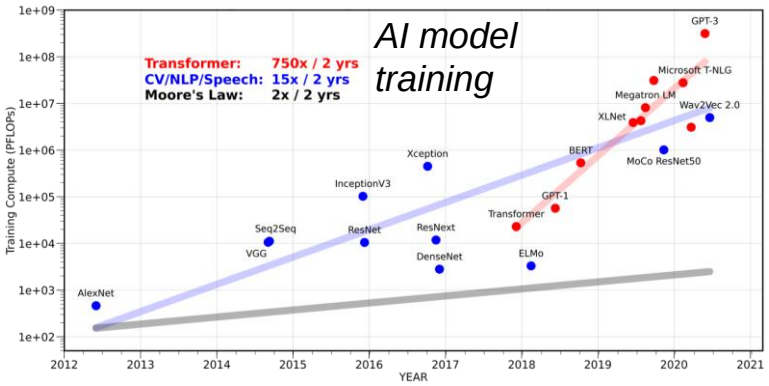
- Further stress on the hardware execution platform

Digital data created: Billion GB/Day



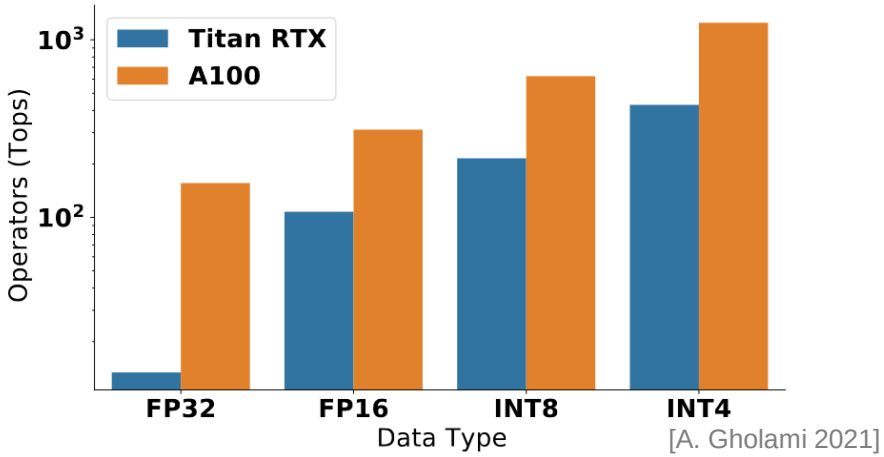
Global E-mail usage: Billion/Day



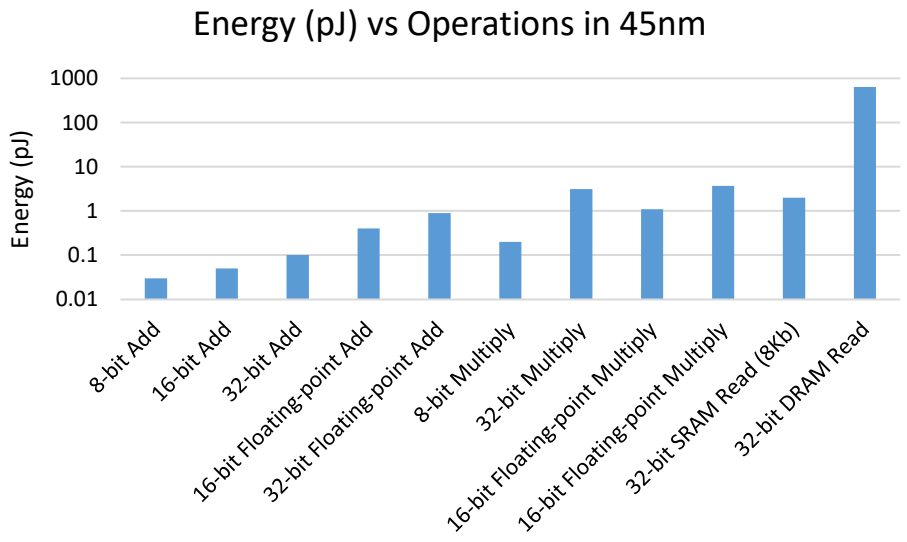


[Micro Focus; A. Gholami, 2020]

Hardware Cost Analysis



[A. Gholami 2021]



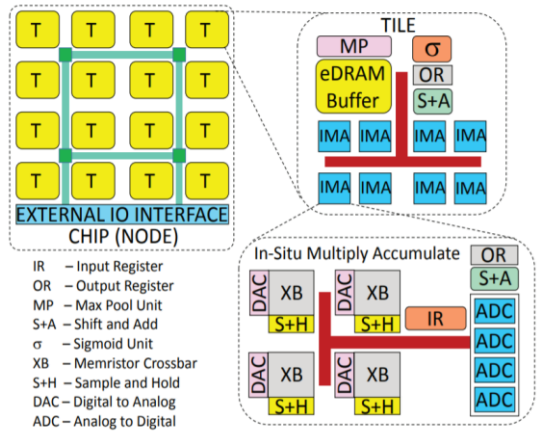
Operation	Area(um^2)*
8-bit Add	36
16-bit Add	67
32-bit Add	137
16-bit Floating-point Add	1360
32-bit Floating-point Add	4184
8-bit Multiply	282
32-bit Multiply	3495
16-bit Floating-point Multiply	1640
16-bit Floating-point Multiply	7700
32-bit SRAM Read (8Kb)	-
32-bit DRAM Read	-

*45nm Technology Node

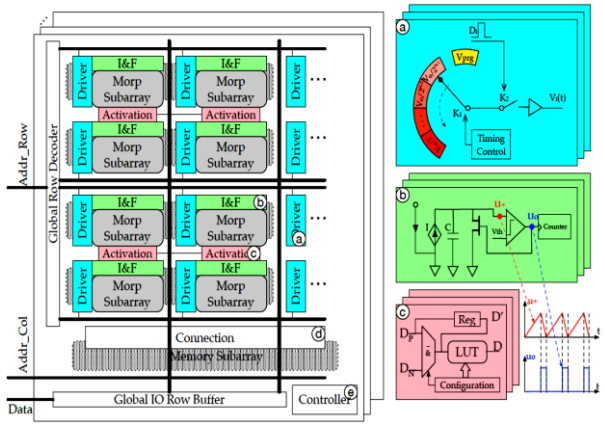
Note: All the above data is adopted from A. Gholami 2021

In-Memory Computing

- In-memory computing (IMC) provides a realistic solution to mitigate the von-Neumann bottleneck
- Combines both memory access and the computation into a single unit through analog domain computation
- Crossbar-based architecture provides a good platform for MVM computations in DNNs



[A. Shafiee 2016]

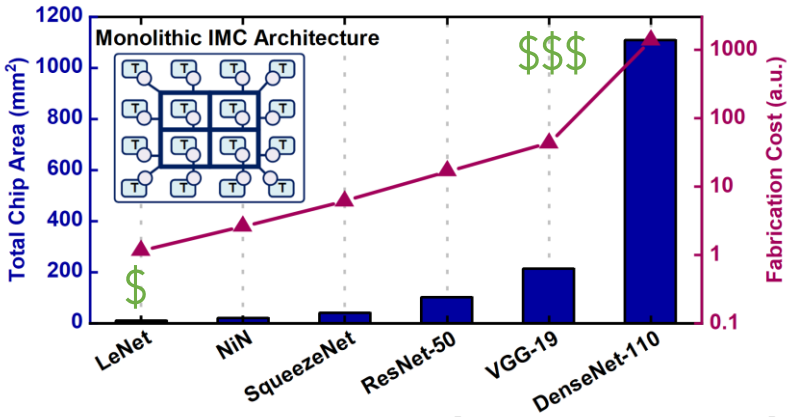


[L. Song 2017]

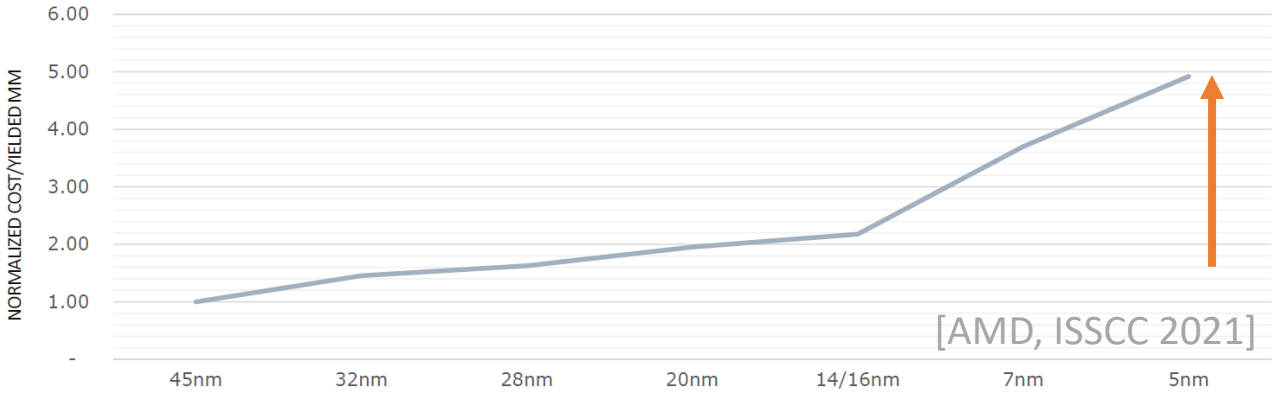
Why not Monolithic IMC?

- IMC accelerators utilize a weight stationary architecture with all weights on-chip

- Monolithic IMC chips result in high fabrication cost with increasing area. Larger area -> more defects, lesser yield, and higher cost



[G. Krishnan, 2021]



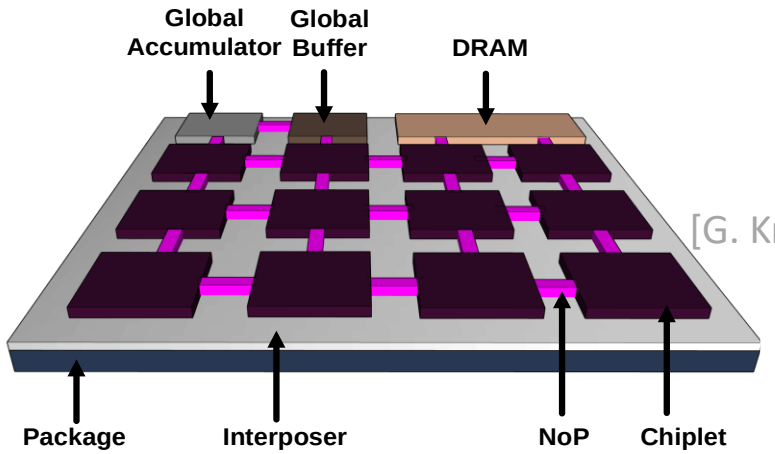
[AMD, ISSCC 2021]

2.5D/Chiplet IMC – An Alternative

- 2.5D or chiplet architecture combine many small chips using an interposer to form a large system

- Each chiplet with smaller size improves the design effort, yield, reduces defect ratio, and reduces fabrication cost

- To achieve similar or better performance as monolithic architectures, careful design of the chiplet architecture and dataflow are necessary



[G. Krishnan, 2021]

We propose **SIAM** !!

What **SIAM** can do and How it Helps !

- Platform for architectural exploration for chiplet-based IMC architectures (RRAM and SRAM)

- Designers get a wide range of parameters to tune and adjust for different architectural choices (e.g. mapping, partition schemes, IMC cell etc.)

Simulator	Architecture	Circuit	Interconnect	NoP Interconnect	DRAM
GenieX	Monolithic	SPICE-based	No	No	No
RxNN	Monolithic	SPICE-based	No	No	No
NeuroSim	Monolithic	SPICE-based	P2P (H-Tree)	No	No
MNSIM	Monolithic	Behavior model	NoC-mesh	No	No
SIAM	**Monolithic & Chiplet**	**SPICE and Behavioral Model**	**NoC-mesh, NoC-tree, and H-Tree**	**Supported (driver and interconnect)**	**Supported**

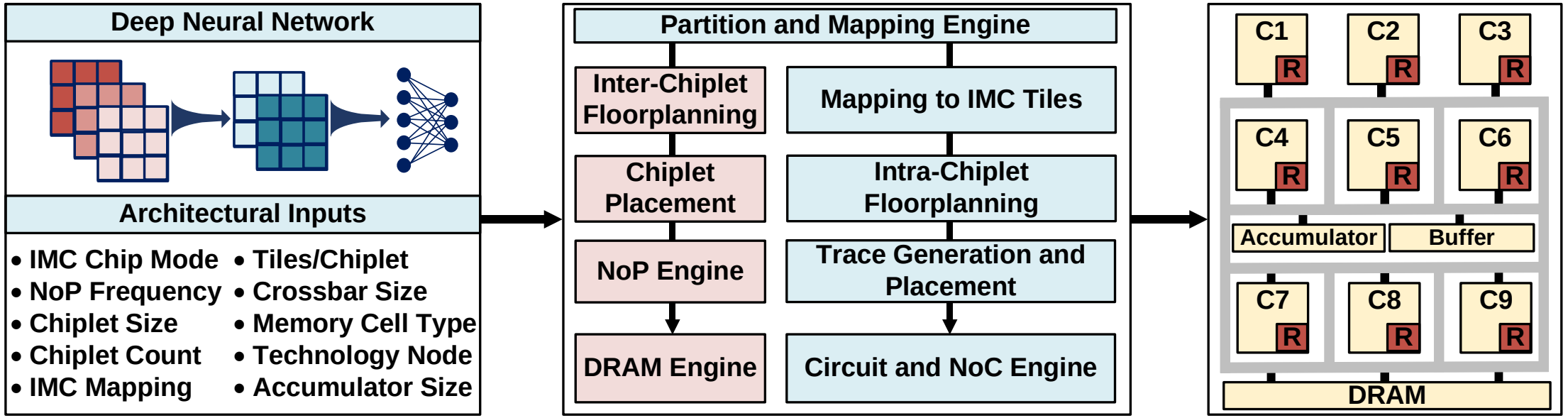
SIAM has been open-sourced and is available at the LINK

SIAM Performance Benchmarking Tool

- SIAM Block Diagram
- SIAM Architecture
- Benchmarking Engines within SIAM
- Dataflow

SIAM Block Diagram

- In-memory computing (IMC) hardware performance benchmarking tool that combines device, circuits, architecture, network-on-chip (NoC), network-on-package (NoP), and DRAM evaluation



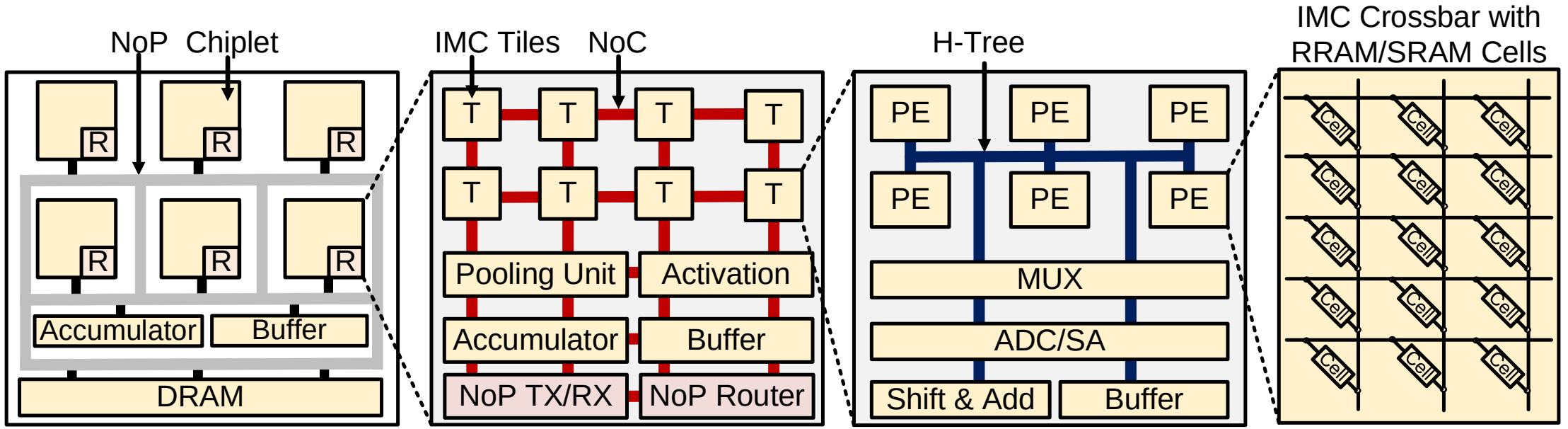
Inputs to SIAM

User Input	Description	User Input	Description
DNN Algorithm		**Device and Technology**	
Network Structure	DNN network structure information	Tech Node	Technology node for fabrication
Data Precision	Weights and activation precision	Memory Cell	RRAM or SRAM
Sparsity	DNN layer-wise sparsity	Bits/Cell	Number of levels in RRAM
Intra-Chiplet Architecture		**Inter-Chiplet Architecture**	
Crossbar Size	IMC crossbar array size	Chip Mode	Monolithic or chiplet-based IMC architecture
Buffer Type	SRAM or Register File	Chiplet Structure	Homogeneous or custom chiplet structure
ADC Resolution	Bit-precision of flash ADC	Chiplet Size	Number of IMC tiles within each chiplet
Read-out Method	Sequential or Parallel	Total Chiplet Count	Fixed count or DNN specific custom count
NoC Topology	Mesh or Tree	Global Accumulator Size	Size of global accumulator
NoC Width	Number of channels in the NoC	NoP Frequency	Frequency of the NoP driver and interconnect
Frequency	Frequency of operation	NoP Channel Width	Number of parallel links for TX and RX

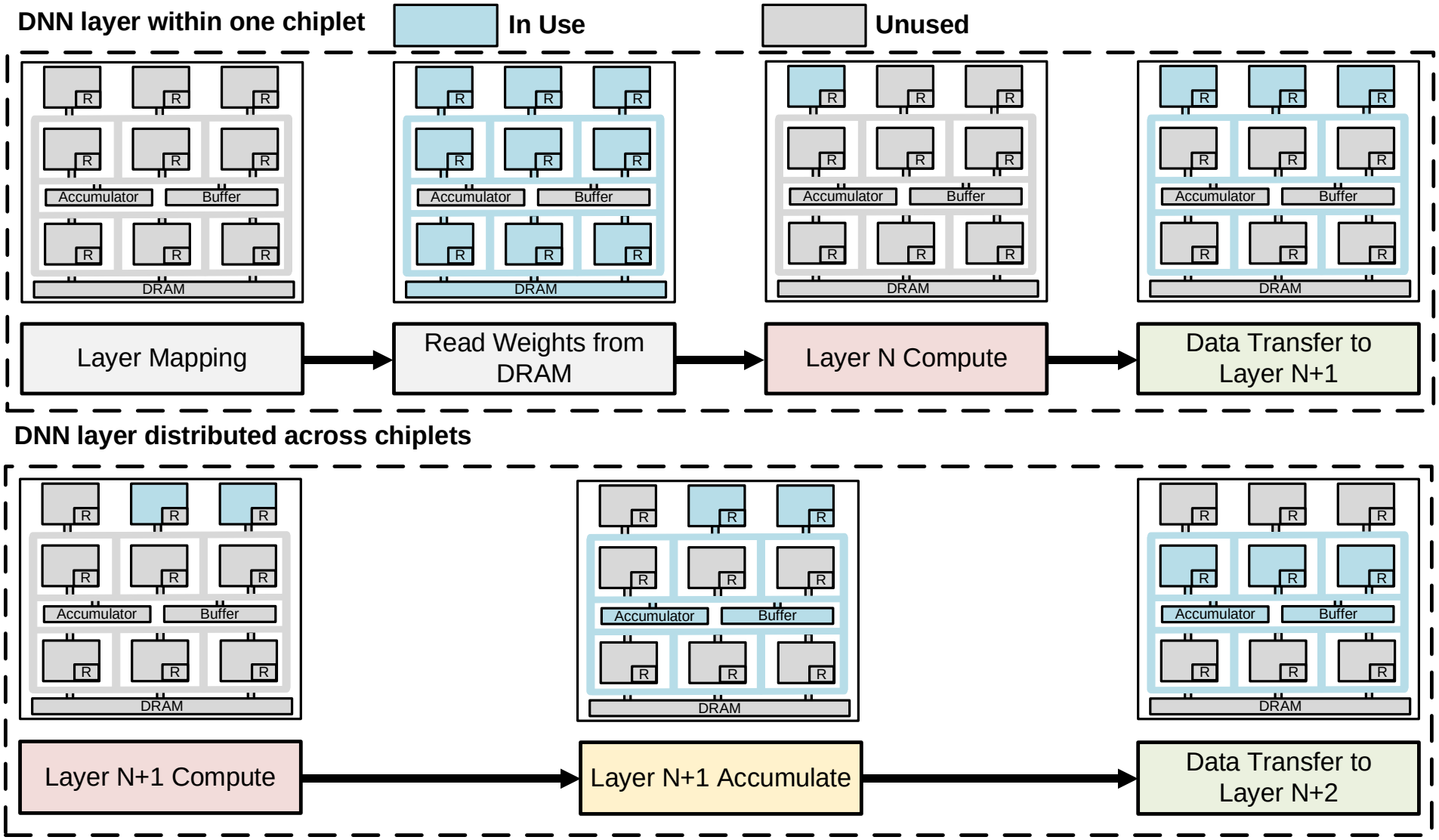
SIAM Architecture

- Array of IMC chiplets, accumulator, buffer, and DRAM connected by an NoP fabric

- Supports both RRAM and SRAM-based IMC crossbar architectures

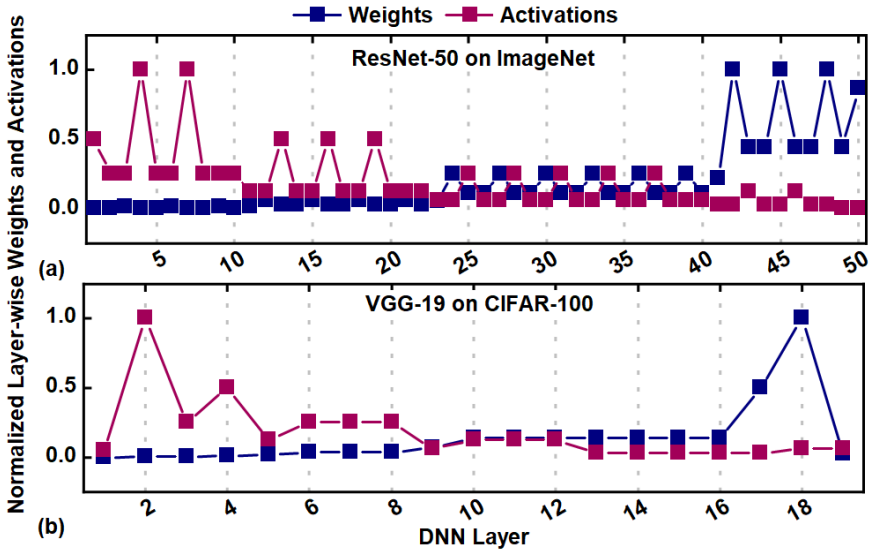


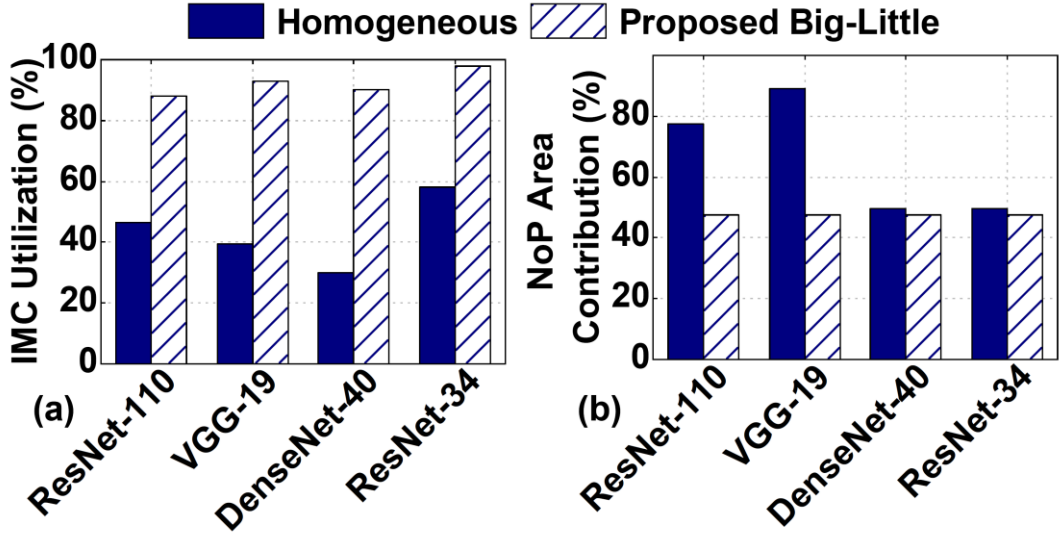
Dataflow


DNN layer within one chiplet
In Use
Unused
R
Accumulator
Buffer
DRAM
Layer Mapping
Read Weights from DRAM
Layer N Compute
Data Transfer to Layer N+1
DNN layer distributed across chiplets
Layer N+1 Compute
Layer N+1 Accumulate
Data Transfer to Layer N+2

Why Heterogeneous Chiplet Architecture

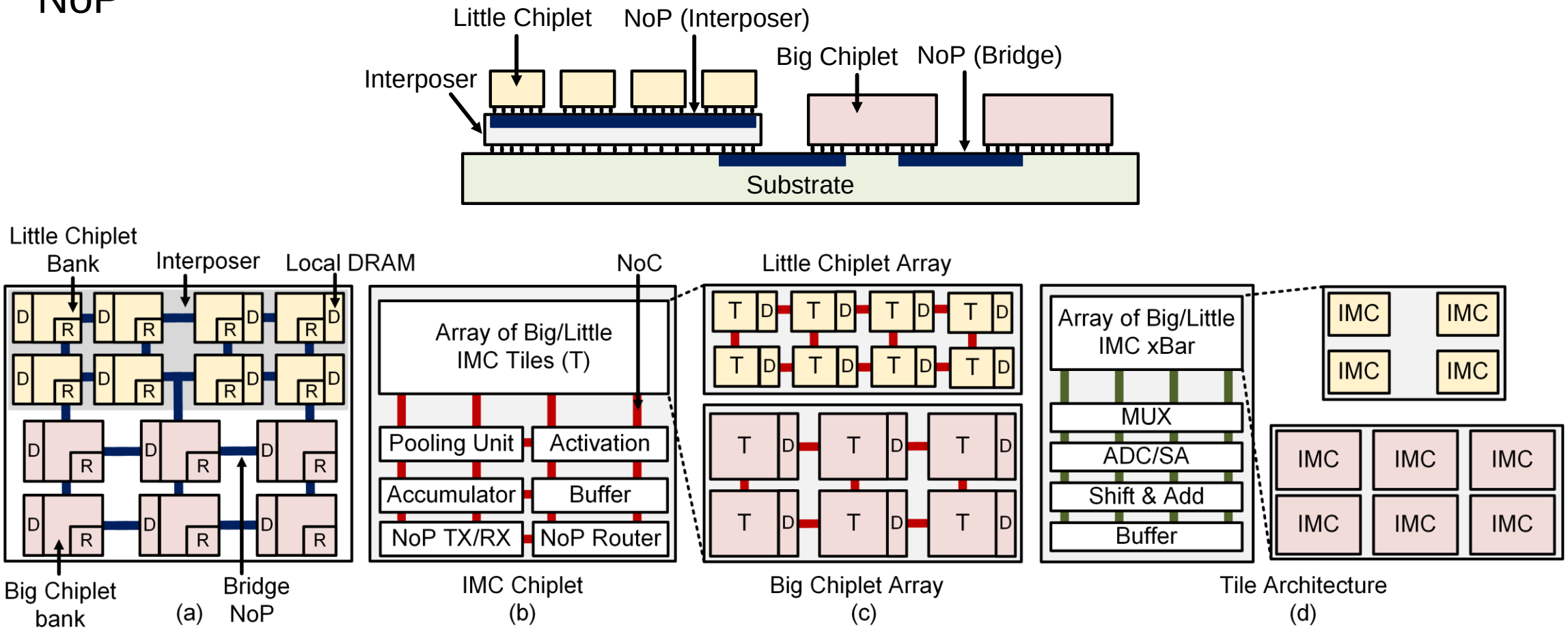
- Inherent non-linear weights and activations distribution in DNNs
- Adverse impact on the IMC utilization resulting in higher area and energy
- Affects hardware cost of the NoP within the architecture





Big-Little Heterogeneous Architecture

- Bank of big and little chiplets connect by an interposed and bridge-based NoP



Mapping Overview

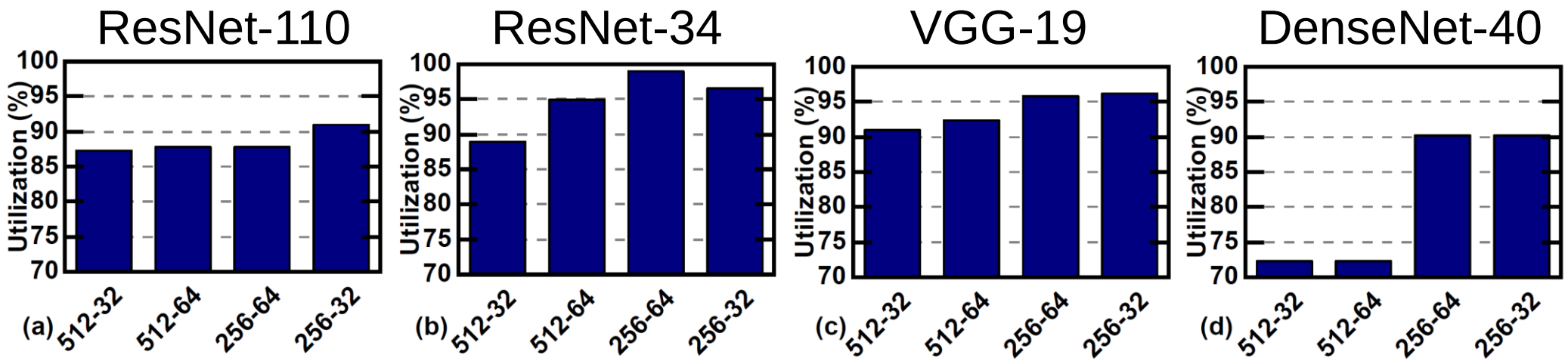
- Mapping algorithm aims to maximize IMC utilization by utilizing the Big-Little IMC chiplet -> determine the config of the architecture
- Little bank with smaller IMC are used for the initial/smaller layers while big bank is used for the larger/deeper layers -> Map the DNN layers
- NoP designed to exploit the volume of data movement in each of the banks -> determine the NoP configurations
- Little bank servicing most of the initial layers has higher data volume movement while the big bank has a lower data movement

Experiments and Results

IMC Utilization with Big-Little

- Utilize the mapping algorithm to determine the best configuration for big and little chiplets

- 256-64 and 256-32 have similar utilization. But 256-64 provides more resources and better energy-efficiency due to reduced peripheral circuits
  - **Little Chiplet:** 25 in number, 25 tiles/Chiplet, and 64x64 IMC size
  - **Big Chiplet:** 11 in number, 36 tiles per chiplet, and 256x256 IMC size



Performance Comparison

- Compare performance with a homogenous all little and all big architecture for VGG-19 on CIFAR-100

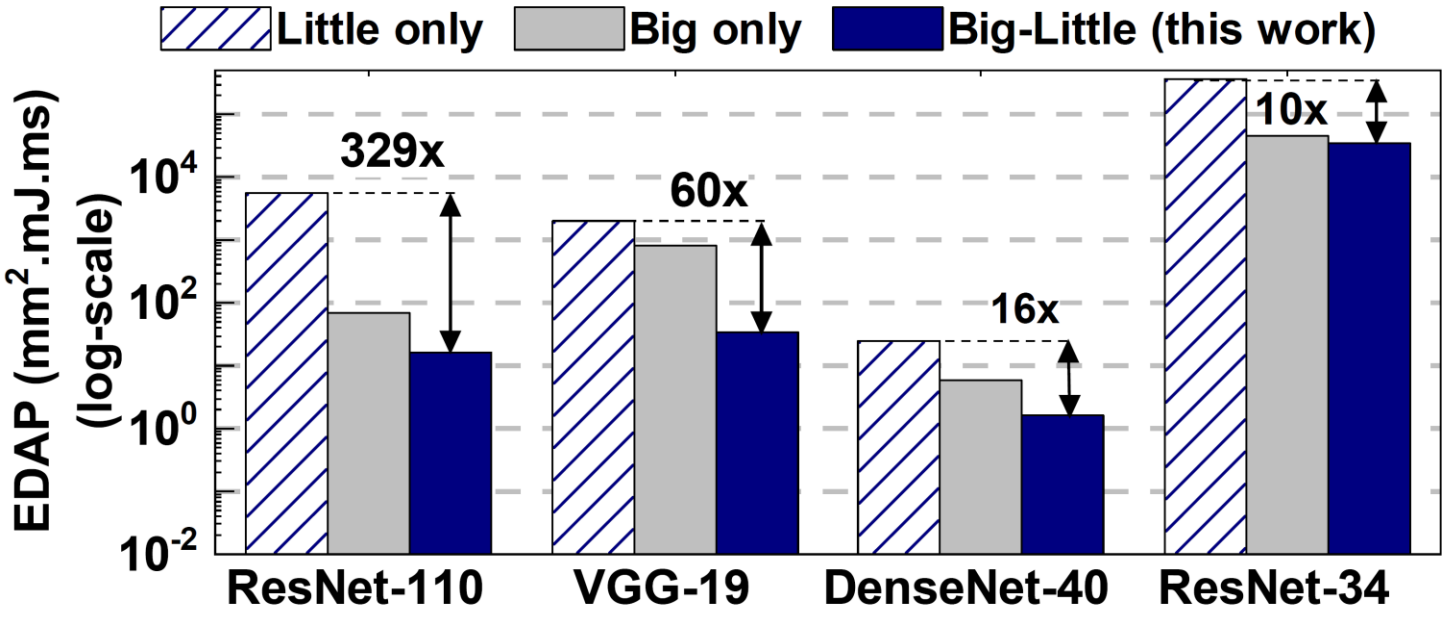
- Proposed big-little architecture achieves reduced area, lower energy, and reduced latency

Configuration	Area					Energy					Latency				
	IMC (%)	NoP (%)	NoC (%)	Total (mm^2)	Normalized to big-little (×)	IMC (%)	NoP (%)	NoC (%)	Total (mJ)	Normalized to big-little (×)	IMC (%)	NoP (%)	NoC (%)	Total (ms)	Normalized to big-little (×)
Little only	11.9	88.0	0.1	952.1	10.9	99.7	0.2	0.1	1.3	4.1	99.7	0.1	0.2	1.6	1.3
Big only	44.0	55.5	0.5	597.2	6.8	78.6	11.0	10.4	0.43	1.3	99.6	0.1	0.3	3.2	2.7
Big-Little (this work)	**52.4**	**47.4**	**0.2**	**87.4**	**1.0**	**99.8**	**0.1**	**0.1**	**0.32**	**1.0**	**99.2**	**0.3**	**0.5**	**1.2**	**1.0**

EDAP Comparison

- We compare the energy-delay-area product of the overall big-little architecture with all little and all big configurations

- Proposed big-little architecture achieves up to 329x improvement in EDA, while consistently outperforming the all big and all little configurations



Support for Unseen Workloads

- We design our architecture to support different workloads by utilizing a local DRAM for each chiplet
- For an unseen workload, weights are written into the IMC arrays multiple times to complete one inference operation
- Ratio of DRAM energy and compute energy for different chiplet configurations

# Chiplets	VGG-16		VGG-19	
	#partitions	Ratio	#partitions	Ratio
36	2	1.1	1	0.08**
25	2	2.1	2	131
16	3	3.6	2	161

** All weights of VGG-19 fit on-chip with this config

Comparison with Other Platforms

- Compared to Nvidia V100 and T4 GPUs, the big-little architecture achieves up to 9.6x improvement in area and 99.6x improvement in energy efficiency

- Compared to state-of-the-art accelerator from Nvidia (SIMBA), the big-little architecture achieves 2.4x area improvement and 18.4x improvement in energy efficiency

Platform	Area (mm^2)	Energy Efficiency (Images/s/W)
Nvidia V100 GPU*	815	8.3
Nvidia T4 GPU*	525	15.5
SIMBA [23]	215	45
Big-Little (this work)	**85**	**827**

Key Take Away

- We motivate the need for chiplet architectures for scalable acceleration of DNNs
- We introduce a novel benchmarking simulator SIAM that can support a wide range of configurations for architectural exploration
- We propose a Big-Little IMC architecture that utilizes a heterogeneous compute and interconnect structure for DNN acceleration
- Experimental evaluation of the proposed big-little architecture shows up to 9.6x improvement in area and 99.6x improvement in energy efficiency over state-of-the art GPUs and accelerators (SIMBA)