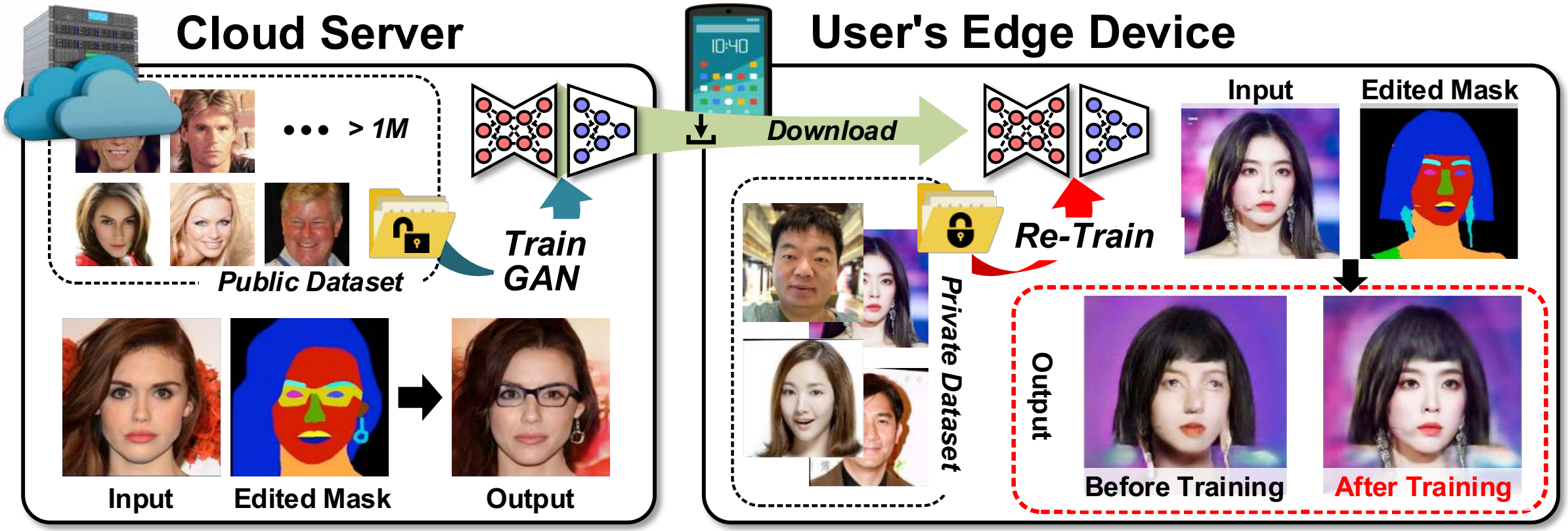
GANPU: A Versatile Many-Core Processor for Training GAN on Mobile Devices with Speculative Dual-Sparsity Exploitation

Sanghoon Kang, Donghyeon Han, Juhyoung Lee,
Dongseok Im, Sangyeob Kim, Soyeon Kim, Junha Ryu, and Hoi-Jun Yoo

Semiconductor System Lab.
School of EE, KAIST

GAN with On-Device Training

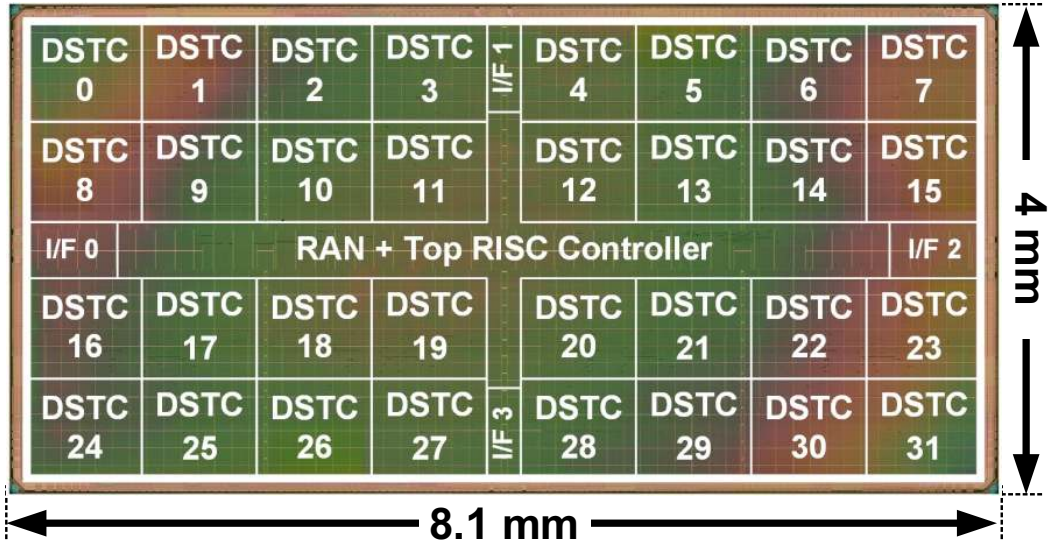
- **GAN Applications on Mobile Need On-Device Training**
  - To restore performance from distortion due to data discrepancy



GANPU: A GAN Training Processor

- **A 135 TFLOPS/W Multi-DNN Accelerator for Mobile Devices**

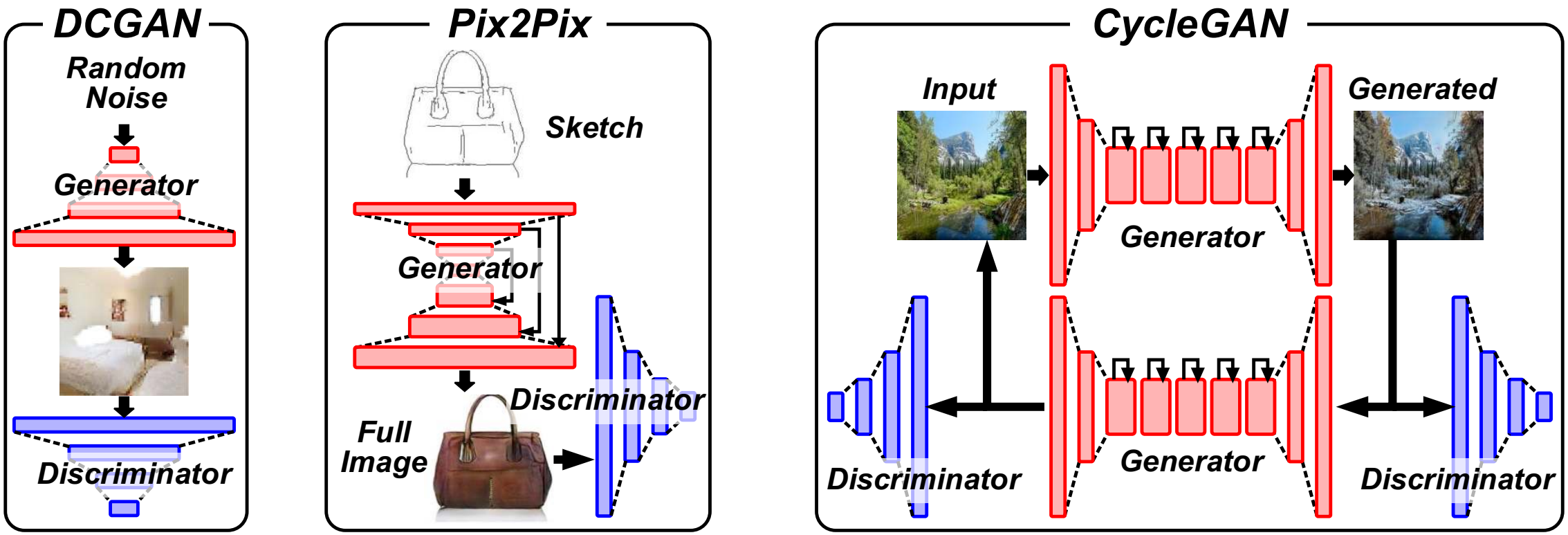
Software Level	Exponent-Only ReLU Speculation
Top-Arch. Level	Adaptive Spatio-Temporal Multiplexing
Micro-Arch. Level	Input-Output Sparse Convolution Core Architecture



- <u>Heterogeneity of multi-DNNs</u> ➔ **Adaptive core & B/W allocation**
- <u>Large # of MAC ops in GANs</u> ➔ **HW architecture & SW algorithm for dual sparsity exploitation**

Multi-DNN Architecture of GAN

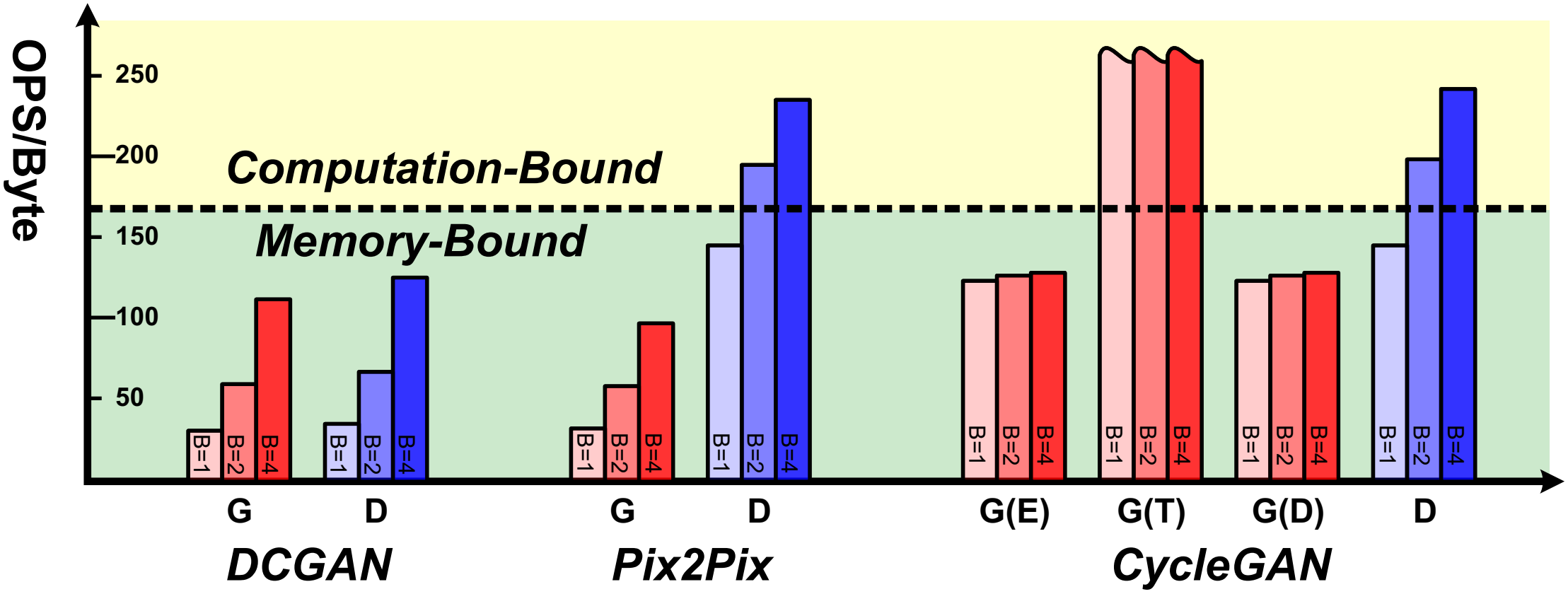
- **Various Multi-DNN Architectures**



Complex Architectures w/ Many DNNs

Multi-DNN Architecture of GAN

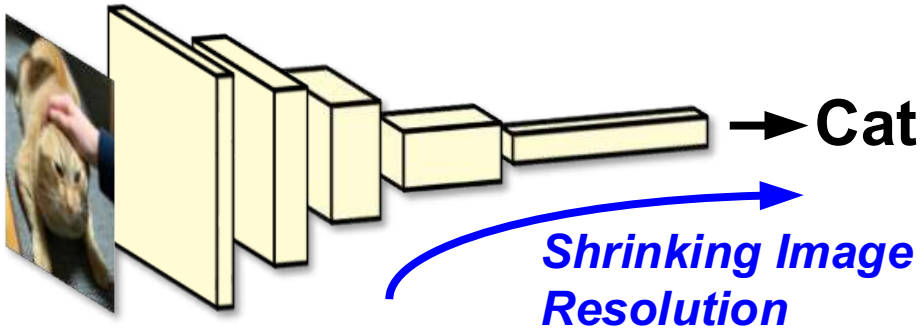
- **Co-Existing Memory Bottleneck & Computation Bottleneck**
  - **Underutilization** of resources (Core & B/W)



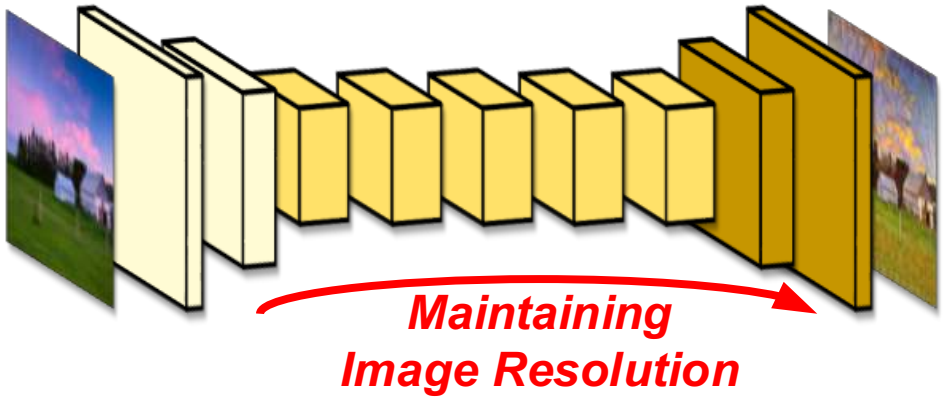
High Computational Requirement

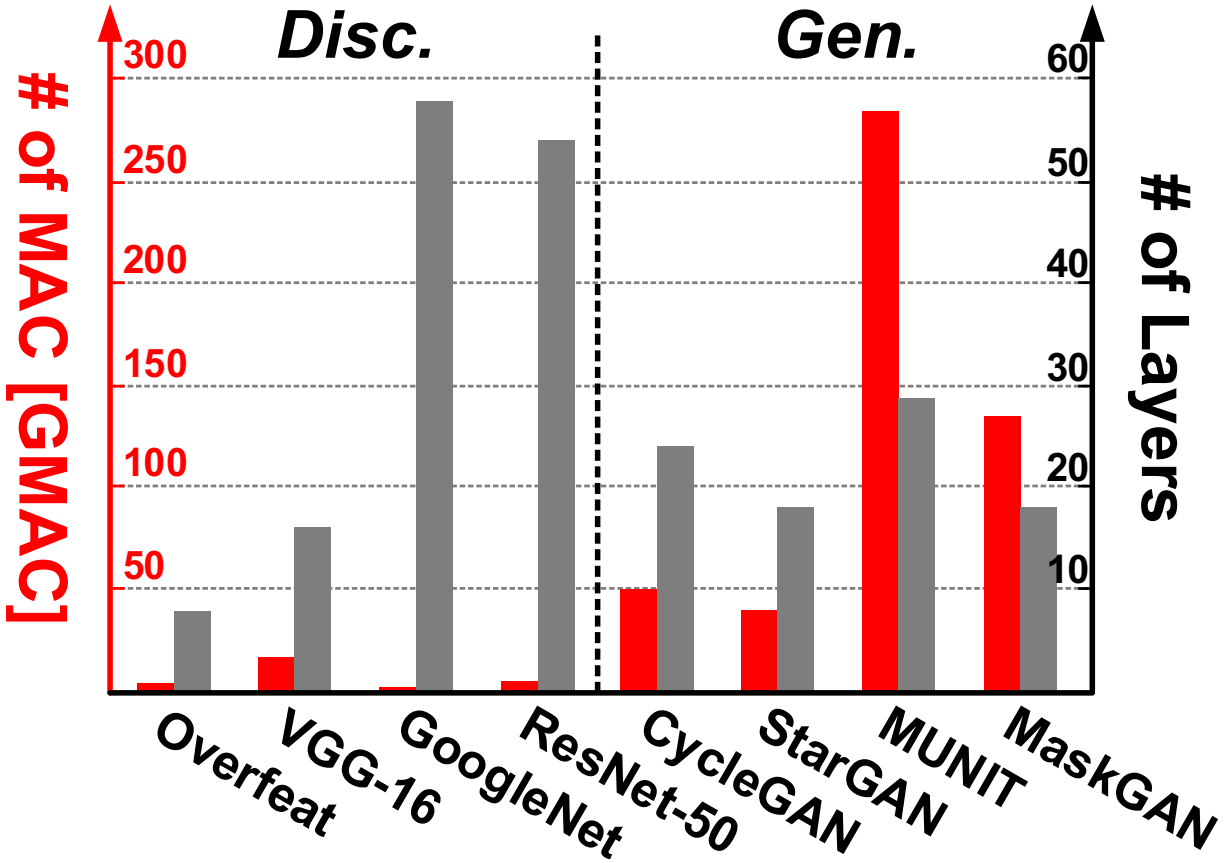
- **More Computation from Larger Intermediate Activations**

Discriminative Models



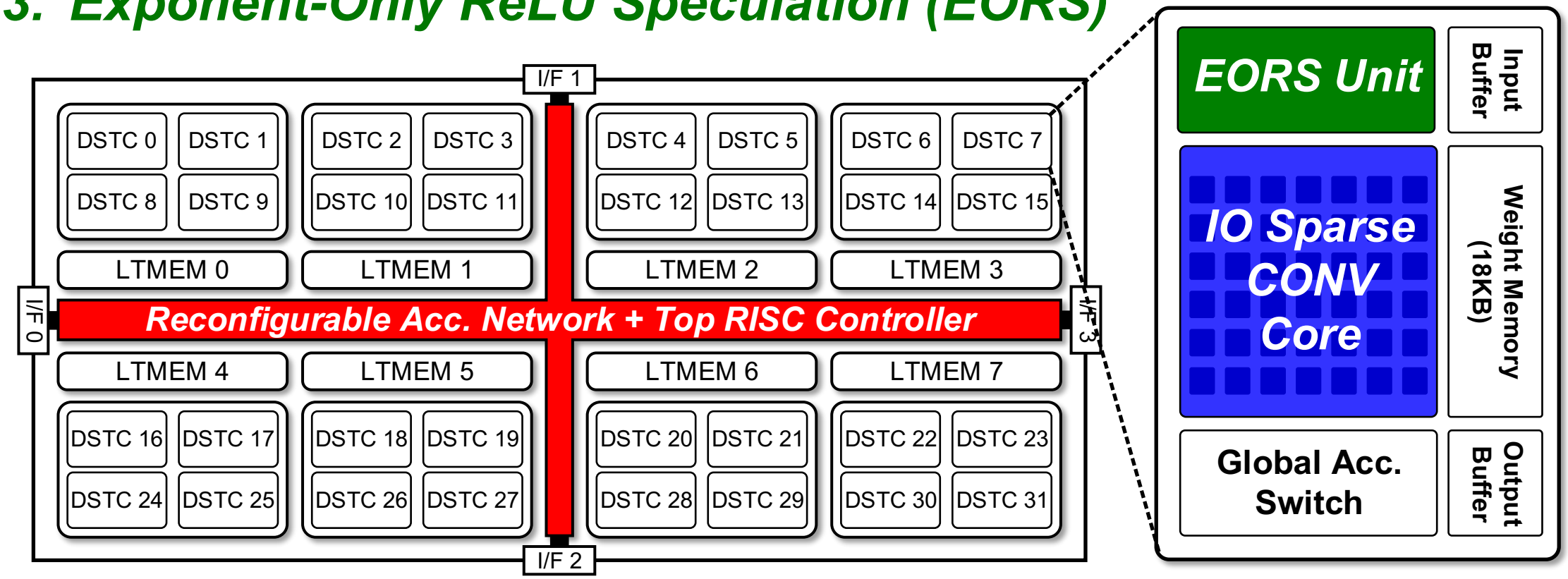
Generative Models





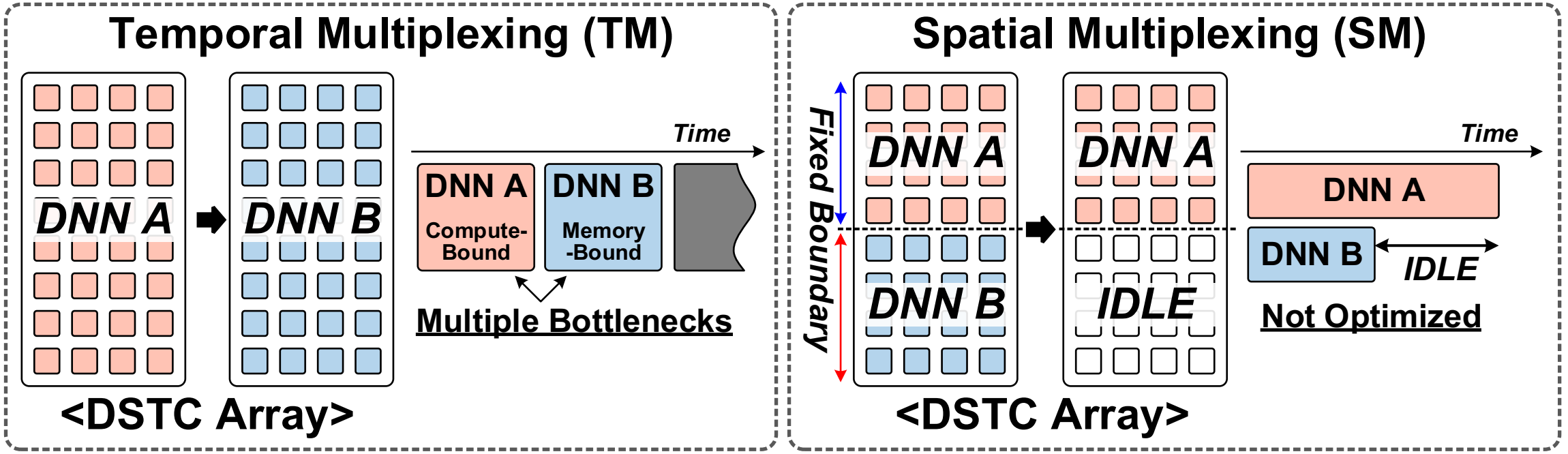
Overall Architecture of GANPU

1. *Adaptive Spatio-Temporal Multiplexing (ASTM)*
2. *Input-Output Sparse Convolution Core Architecture (IOSC)*
3. *Exponent-Only ReLU Speculation (EORS)*



Multi-DNN Workload Allocation

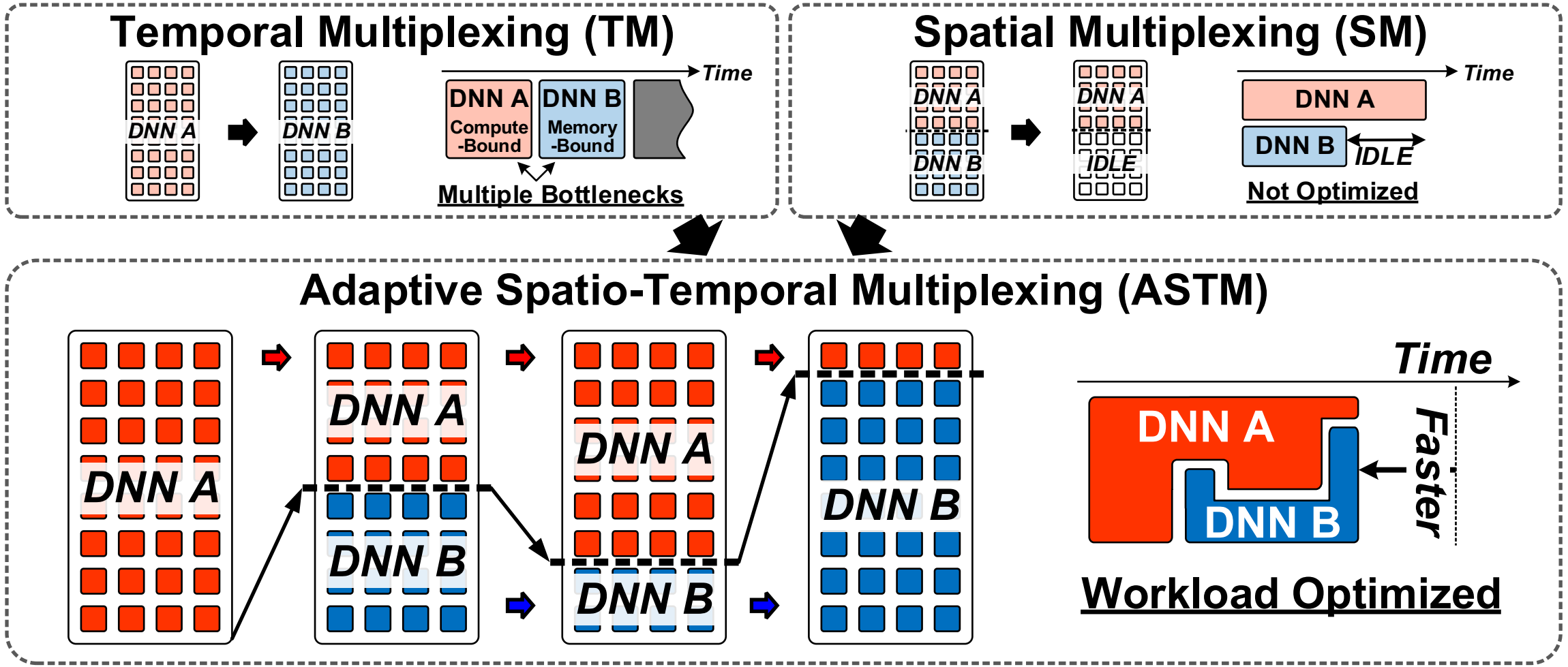
- **Conventional Workload Multiplexing Methods**



  - Varying computation & B/W workloads ➔ ***Multiple bottlenecks***
  - Fixed DSTC allocation ➔ ***No adaptation for optimal processing***

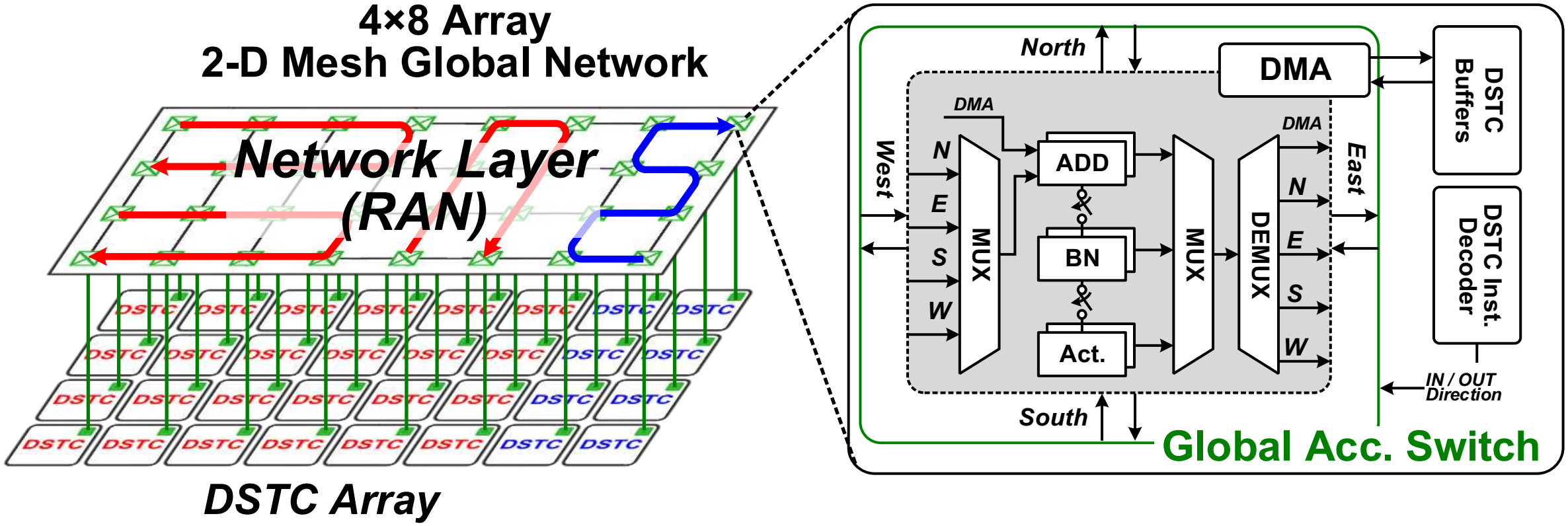
Multi-DNN Workload Allocation

- **Adaptive Spatio-Temporal Multiplexing (ASTM)**



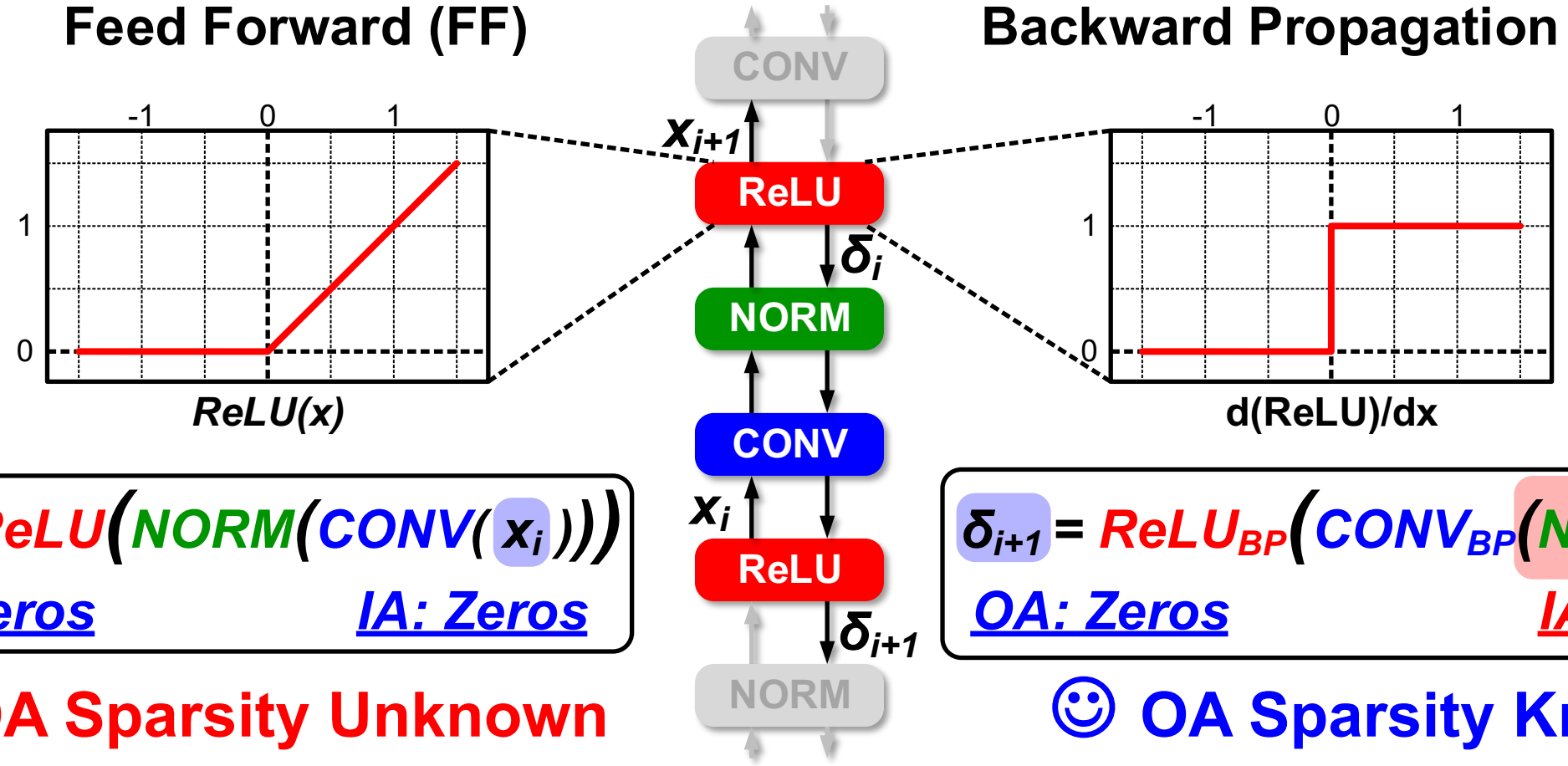
Reconfigurable Accumulation Network

- **Supporting Versatile Accumulation Patterns**
  - Stream partial sums across DSTCs through **global accumulation switch**
  - **14% throughput increase** in Pix2Pix



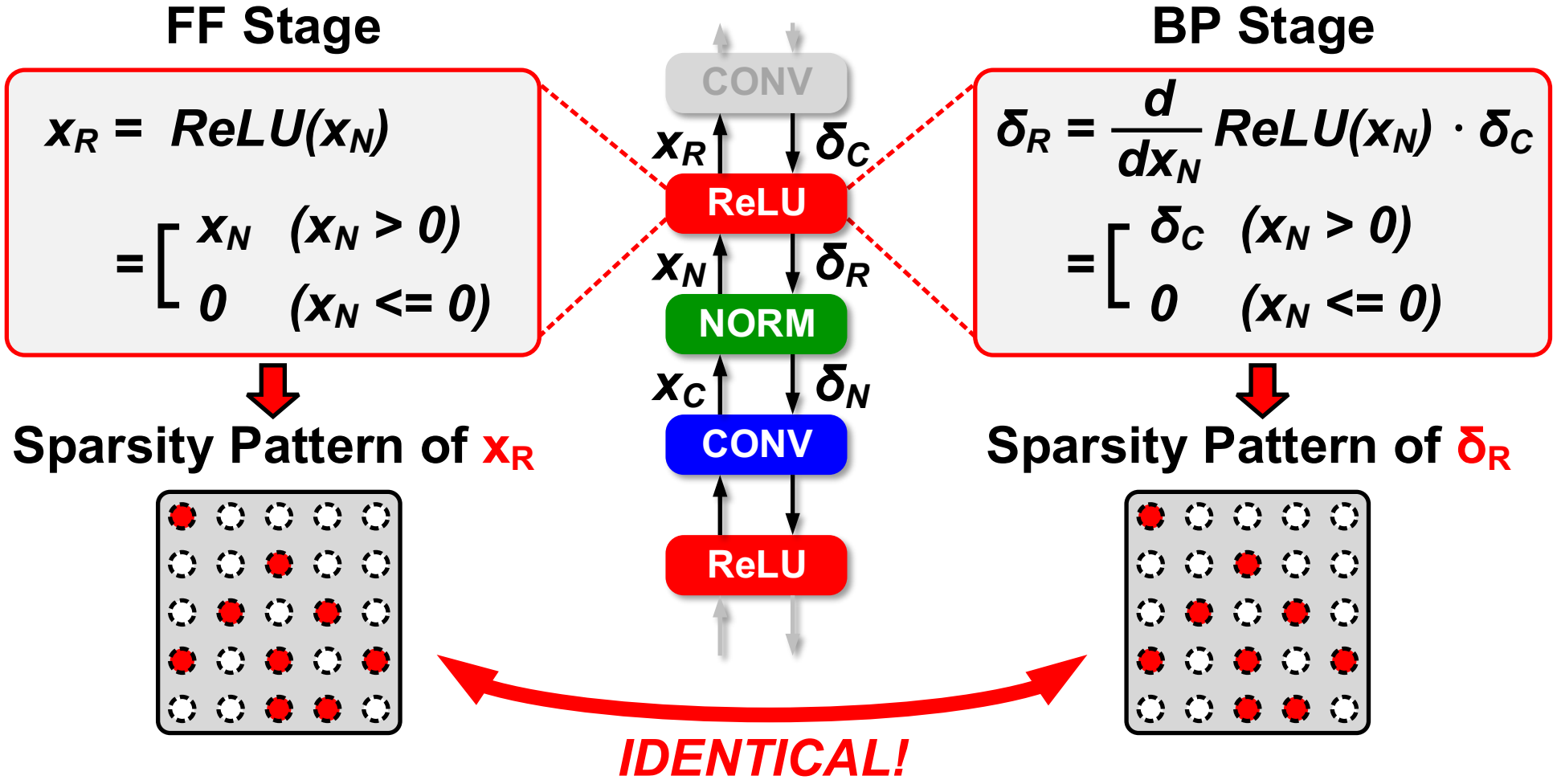
Sparse Activation in DNN Training

- **Activatio Sparsity @ FF & BP Stages Due to ReLU**



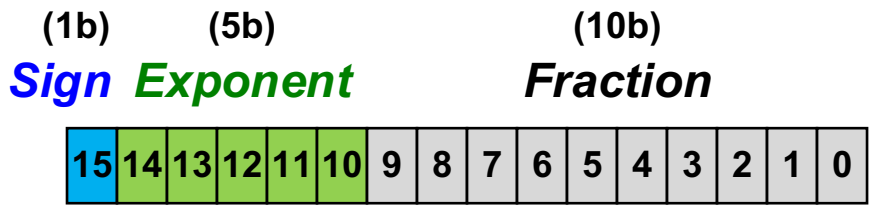
OA Sparsity @ BP Stage

- **Identical Pattern ➔ OA Sparsity Pattern Pre-determined**



OA Sparsity @ FF Stage

- **Exponent-Only ReLU Speculation (EORS)**



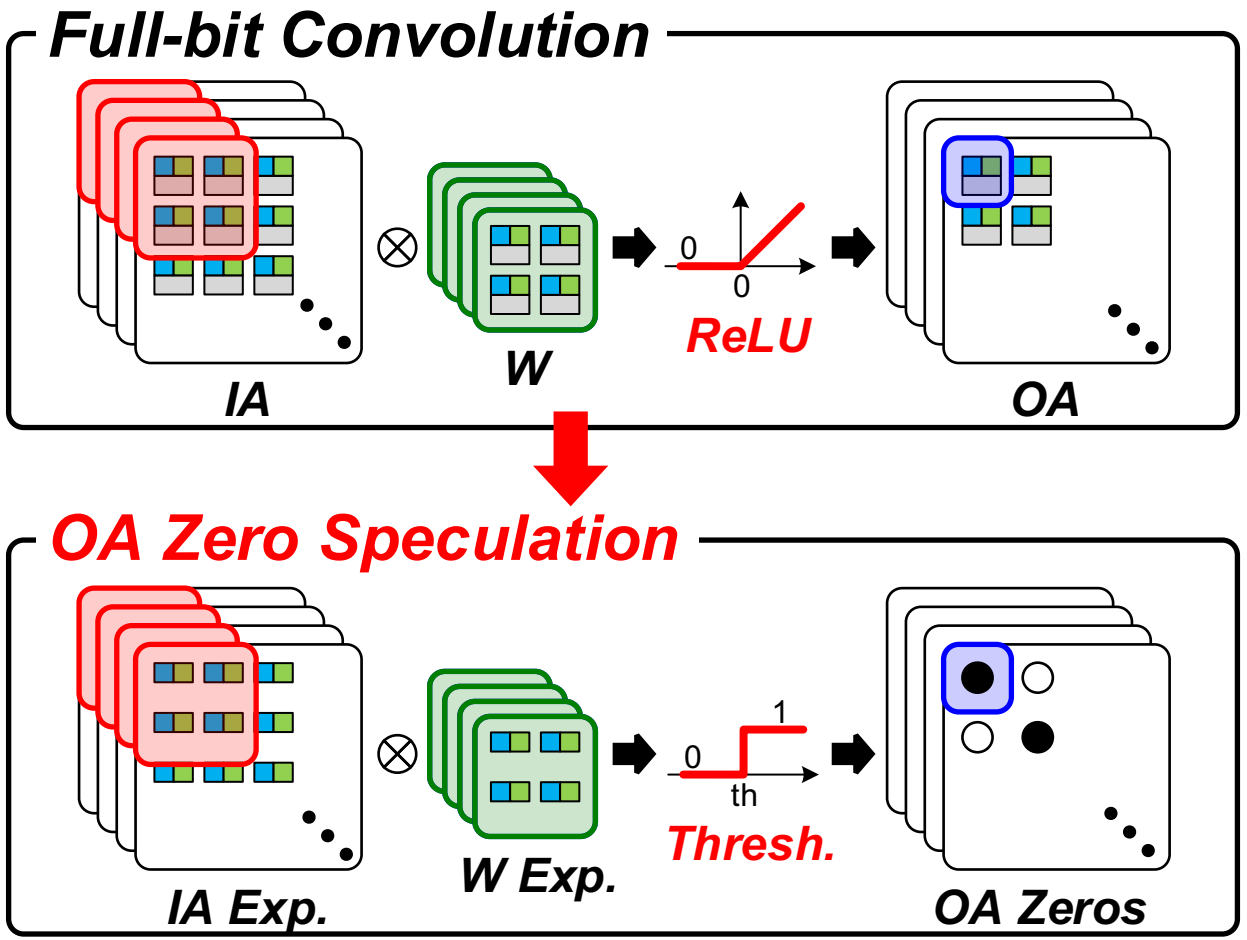
< FP16 (Half Precision) Format >

$$Data = (-1)^{Sign} \times 2^{Exponent-15} \times 1.Fraction_2$$

(less important)

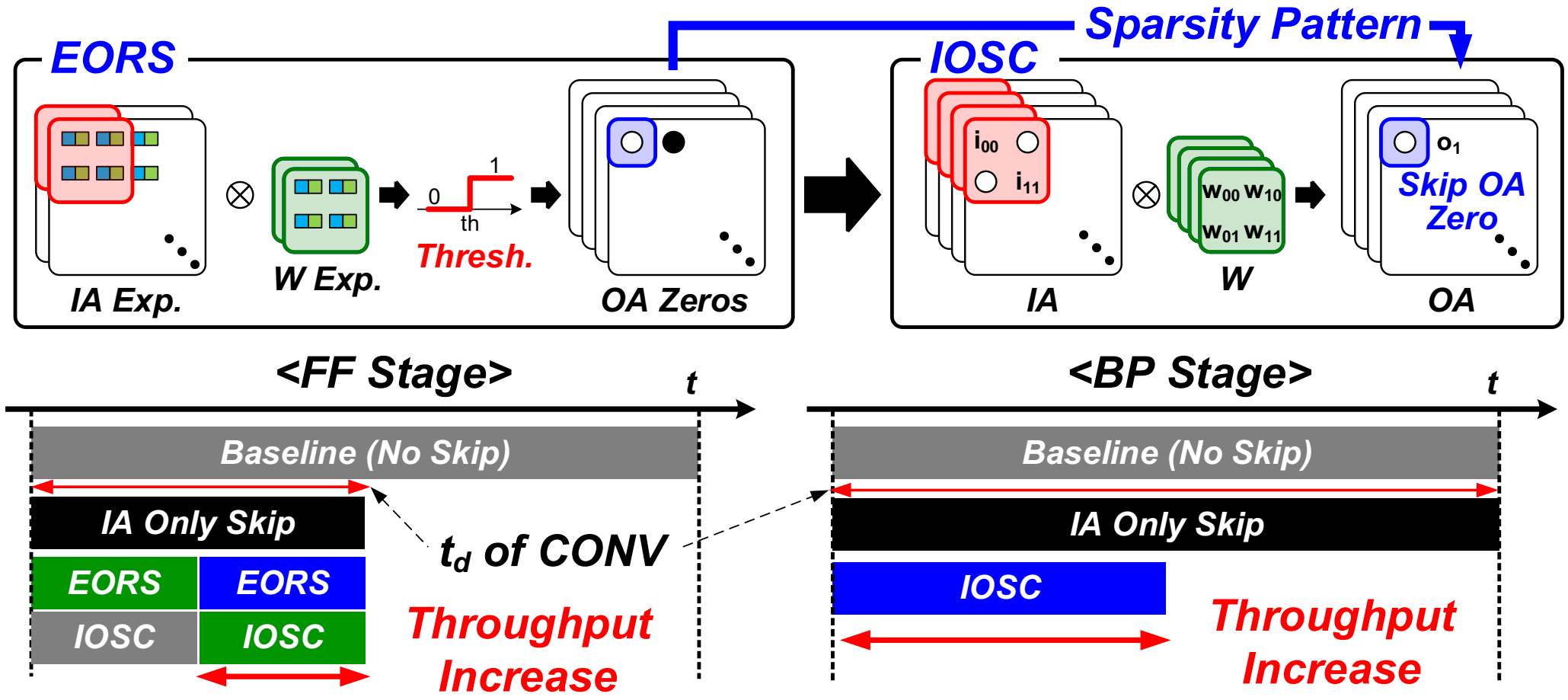
Scale of IA & W

Simple to Estimate OA Sparsity Pattern



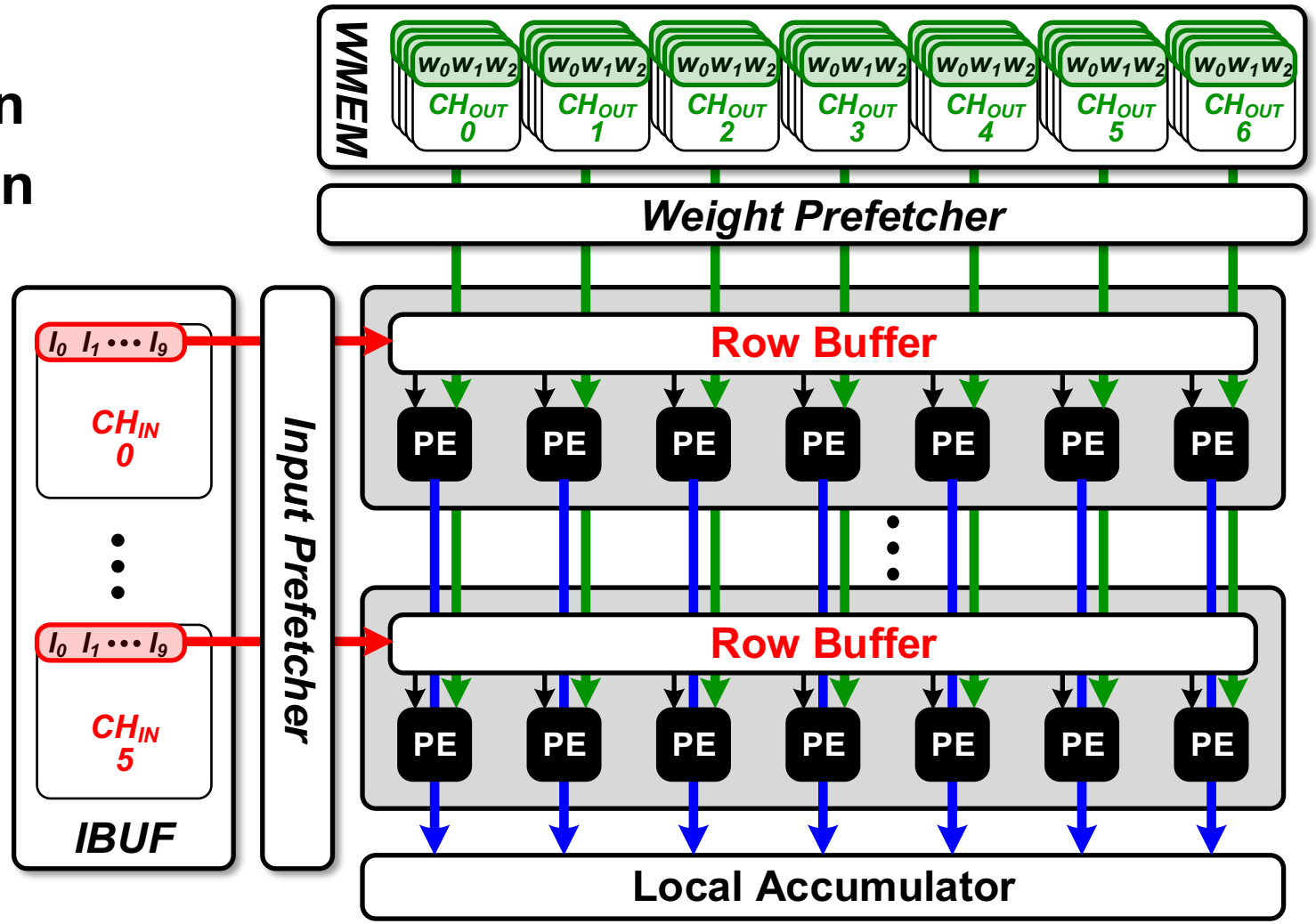
Dual Sparsity Exploitation with EORS

- **Higher Throughput @ Both FF & BP *Through IOSC + EORS***



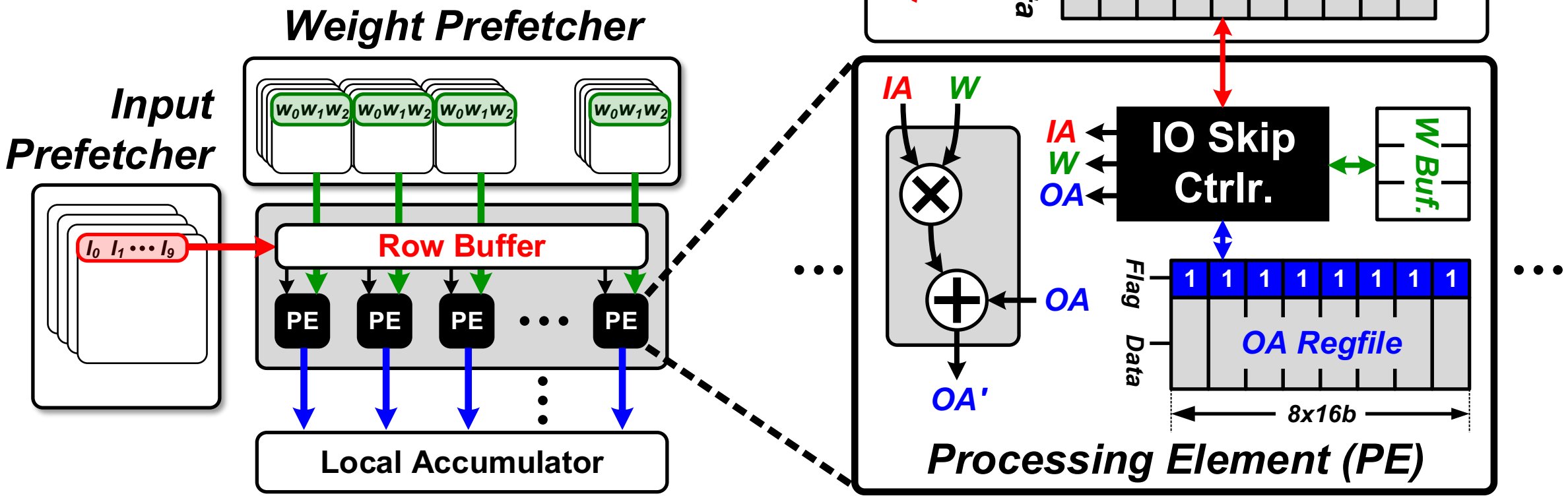
IOSC Architecture

- 6×7 PE Array in IOSC
  - CH_{IN}: Row-wise division
  - CH_{OUT}: Col-wise division
  - ➔ Every PE allocated w/ different CH!
- Hierarchical Dataflow
  - Row stationary (Inside row buffer)
  - Input stationary (Inside PE)



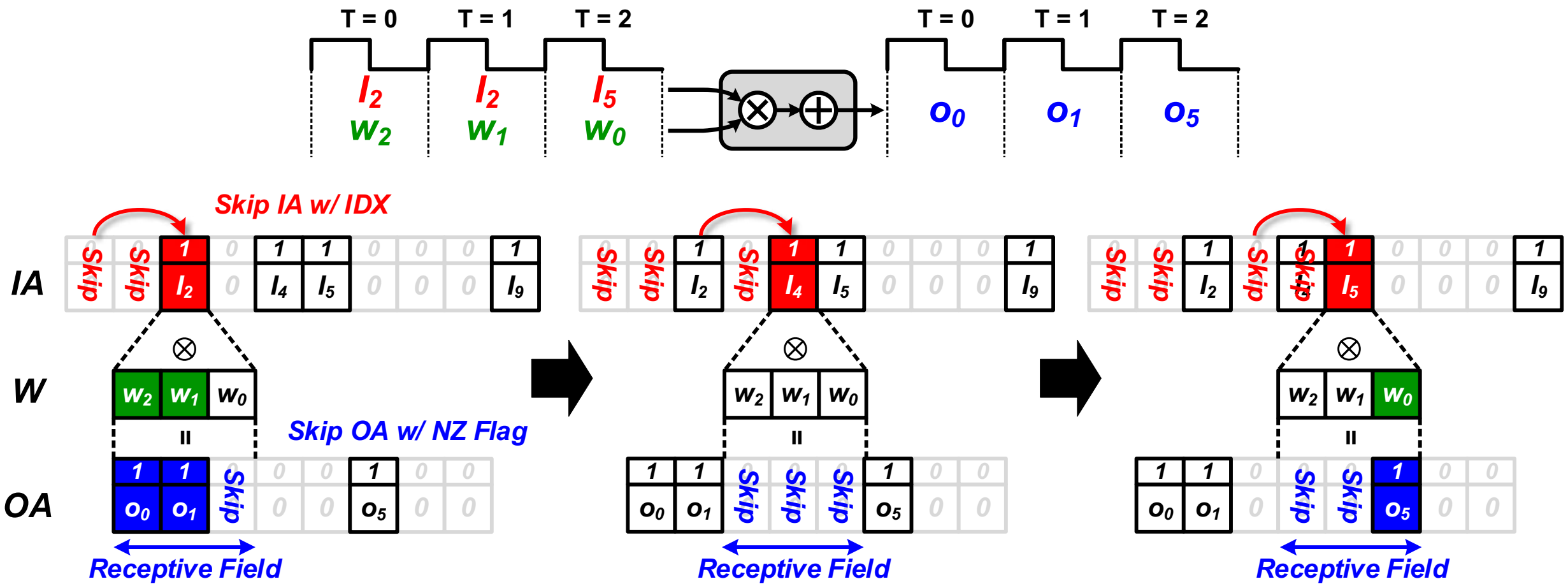
IOSC PE Architecture

- **IO Skip Ctrlr. Inside Each PE**
  - 1) Skip IA w/ IDX
  - 2) Skip OA w/ flag in receptive field



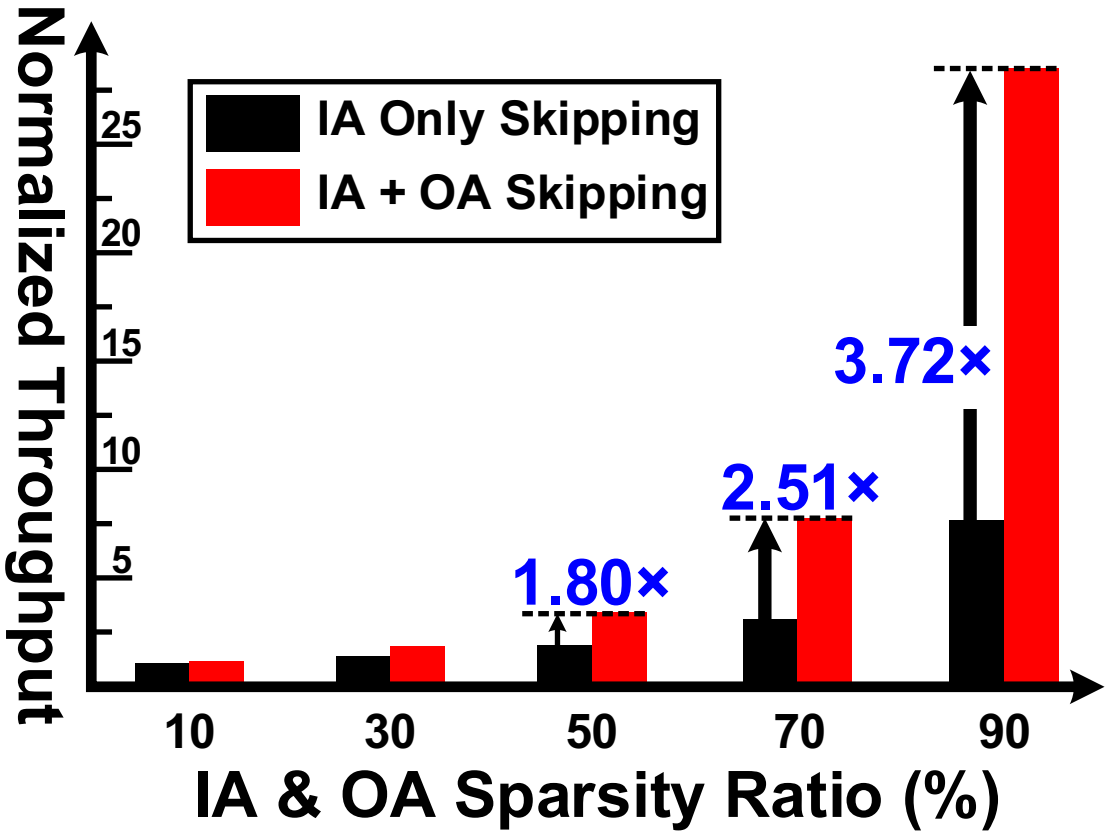
IOSC PE Architecture

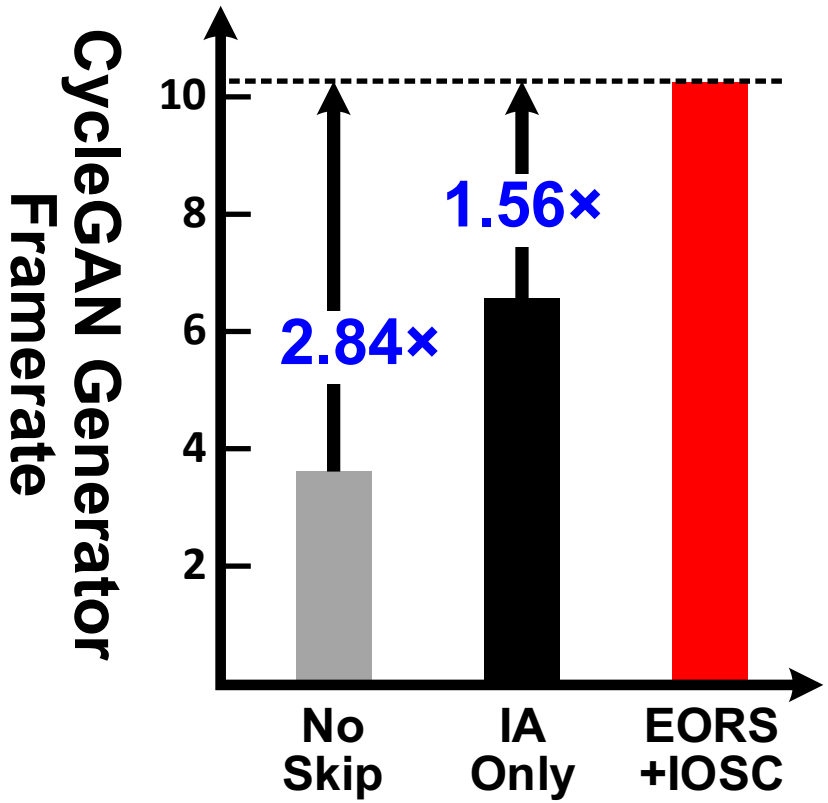
- **Zero-Skipping Inside PE**



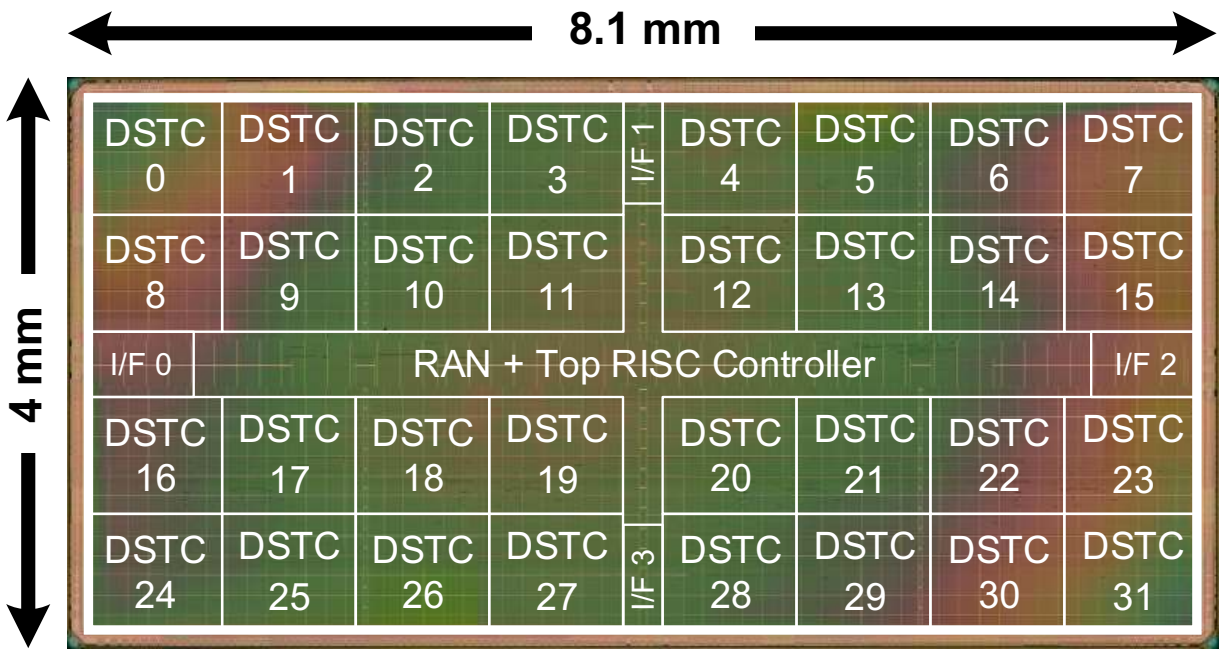
IOSC Performance

- **Throughput Increase with Dual Sparsity Exploitation**
  - **2.84× increase in CycleGAN inference framerate**





Chip Photography and Summary



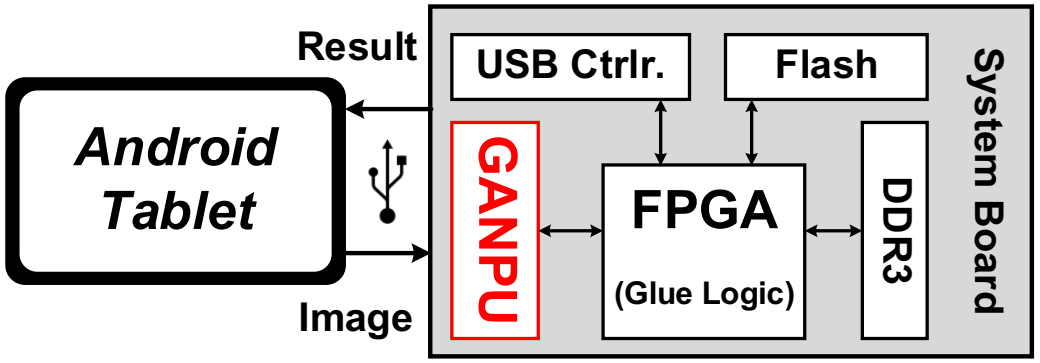
- **CycleGAN Training**
  - **2.04× throughput increase**
  - **76.25% energy-efficiency increase**

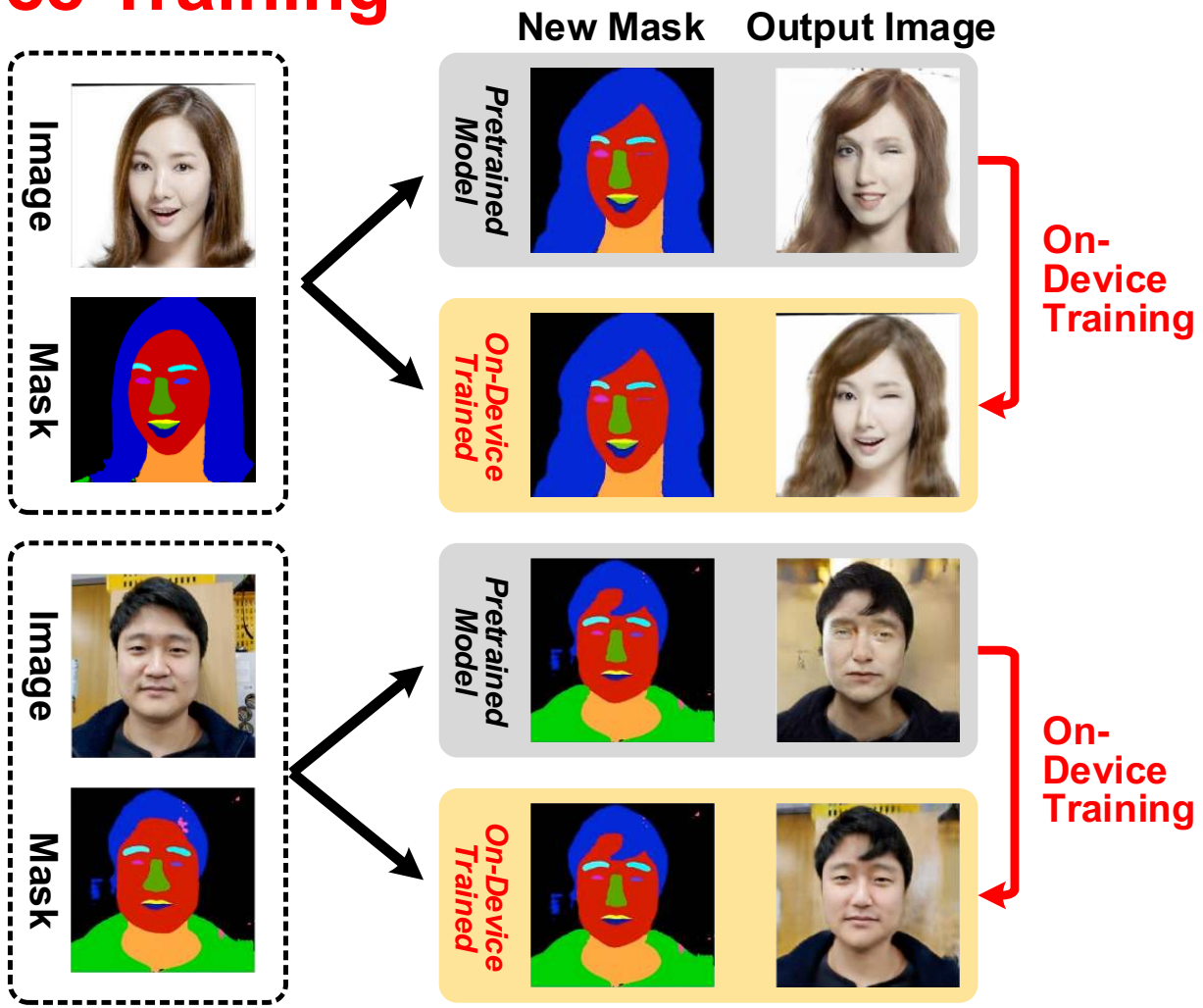
	Specifications	
Technology	Samsung 65nm 1P8M CMOS	
Die Area	4mm × 8.1mm (32.4mm^2)	
SRAM	676 KB	
Supply	0.70V ~ 1.1V	
Frequency	~ 200MHz	
Power Consumption	58mW @(25MHz, 0.70V)	
	647mW @(200MHz, 1.1V)	
	FP16	FP8
Peak Performance	0.54 - 14.02* TFLOPS	1.08 - 24.13* TFLOPS
Power Efficiency [TFLOPS/W]	0.83 - 37.80* @(200MHz, 1.1V)	1.66 - 68.12* @(200MHz, 1.1V)
	1.81 - 75.68* @(50MHz, 0.75V)	3.62 - 135.10* @(50MHz, 0.75V)

** Logical Performance @ 90% IA&OA Sparsity*

Demonstration System

- **Face Manipulation w/ On-Device Training**





Conclusion

- **GANPU: An Energy-Efficient Processor for Training GANs on Mobile Devices**

- **For Energy-Efficient GAN Training**
  - ***Adaptive Spatio-Temporal Workload Multiplexing***
  - ***IA-OA Sparsity Exploitation Convolution Core Architecture***
  - ***Exponent-Only ReLU Speculation***

A 135 TFLOPS/W Multi-DNN Training Processor for Mobile GAN Applications

Thank You!

- **Questions? Feel Free to Contact Me!**
  - E-mail: sanghoon_kang@kaist.ac.kr
  - LinkedIn: https://www.linkedin.com/in/sanghoonkang94/
  - Zoom Meeting: https://zoom.us/j/97717782673?pwd=Z1BSMGQ2bXVZcmZNVlFzS3ZBL3B1QT09 (Password: HC_GANPU)

- **Acknowledgement**
  - This work was supported by the Samsung Research Funding & Incubation Center for Future Technology under Grant SRFC-TB1703-09.