Picasso: An Area/Energy-Efficient End-to-End Diffusion Accelerator with Hyper-Precision Data Type

CastLab

Sungyeob Yoo, Geonwoo Ko, Seri Ham, Seeyeon Kim, Yi Chen, and Joo-Young Kim

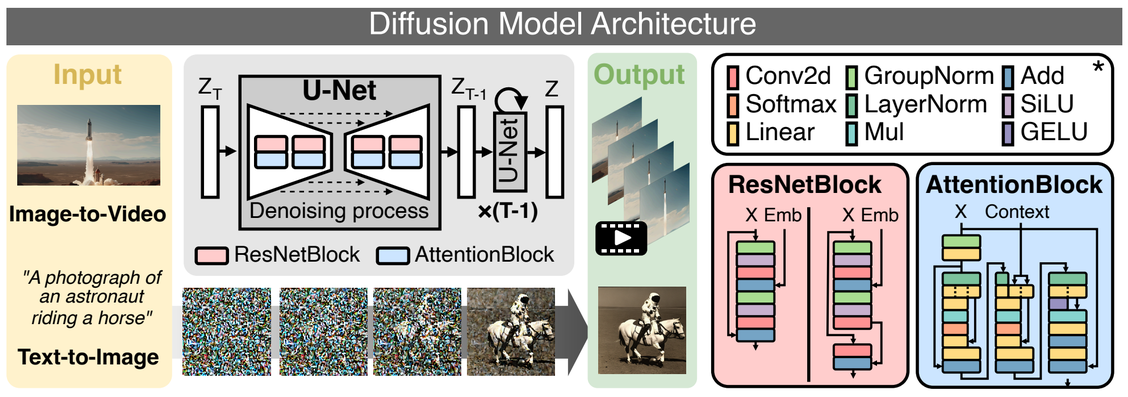
KAIST

Abstract

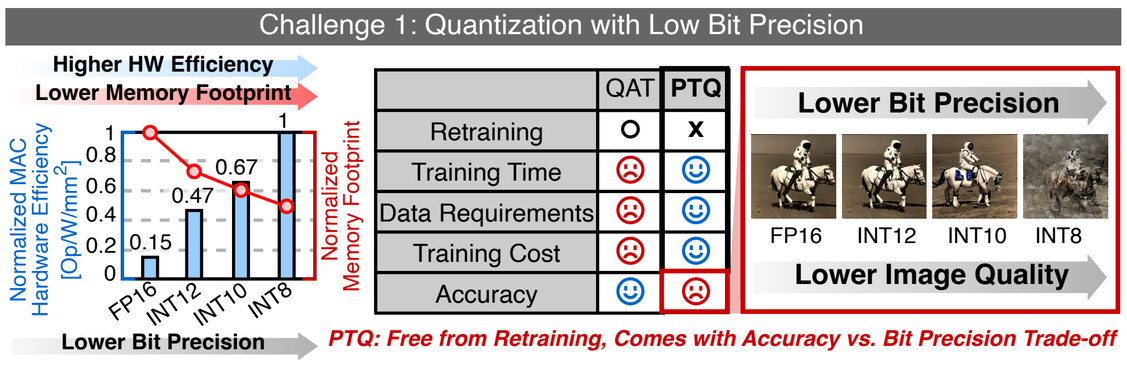
This work presents **Picasso, an end-to-end diffusion accelerator** designed for enhancing the efficiency of diffusion-based machine learning models used in applications such as image and video generation, and inpainting. Picasso introduces a novel hyper-precision 8 (HYP8) data type and a reconfigurable architecture designed to significantly enhance hardware efficiency, providing an extended dynamic range without sacrificing accuracy. It also features a unified engine that streamlines the processing of all non-matrix operations and employs sub-block pipeline scheduling to reduce overall latency. Fabricated in 28nm CMOS technology, this accelerator achieves an energy efficiency of **4.96 TOPS/W** and a peak performance of **9.83 TOPS**. Compared to previous works, Picasso demonstrates speedups ranging **from 8.4× to 26.8×** while also improving energy and area efficiency by **1.1×-2.8×** and **3.6×-30.5×**, respectively.

Motivation & Contributions

Diffusion Model Architecture



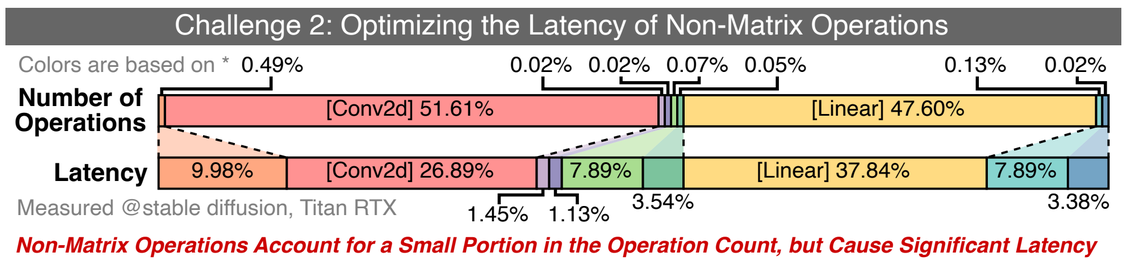
Challenge 1: Quantization with Low Bit Precision



	QAT	PTQ
Retraining	O	X
Training Time	☹	☺
Data Requirements	☹	☺
Training Cost	☹	☺
Accuracy	☺	☹

PTQ: Free from Retraining, Comes with Accuracy vs. Bit Precision Trade-off

Challenge 2: Optimizing the Latency of Non-Matrix Operations

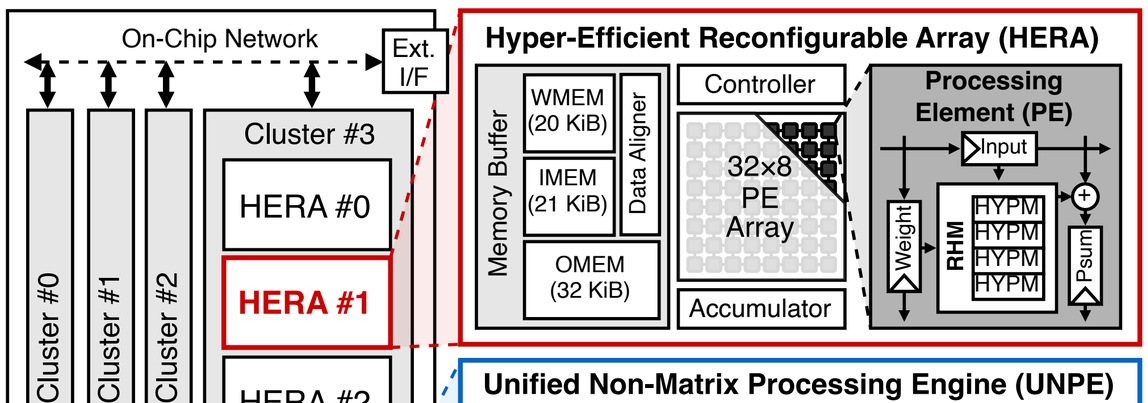


Non-Matrix Operations Account for a Small Portion in the Operation Count, but Cause Significant Latency

※ Contributions

1. **Hyper-Precision Data Type (HYP8)**
   - **Provides high resolution and a broad data range without losing accuracy**
2. **Hyper-Efficient Reconfigurable Array (HERA)**
   - **Supports HYP8 enabling dynamic range extension with no accuracy drop**
3. **Unified Non-Matrix Processing Engine (UNPE)**
   - **Streamlines all non-matrix operations and reduces end-to-end latency**

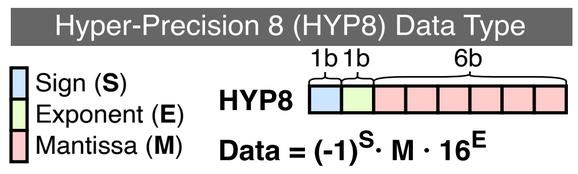
Overall Architecture



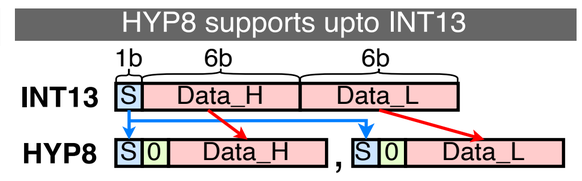
- Picasso includes 4 clusters, top controller, PLL, and network-on-chip
- Each cluster contains 3 HERAs, and a UNPE

Hyper-Precision Data Type

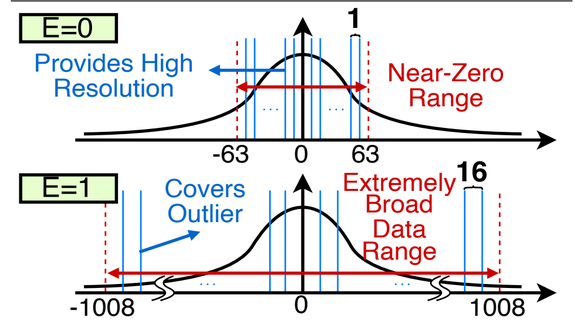
Hyper-Precision 8 (HYP8) Data Type



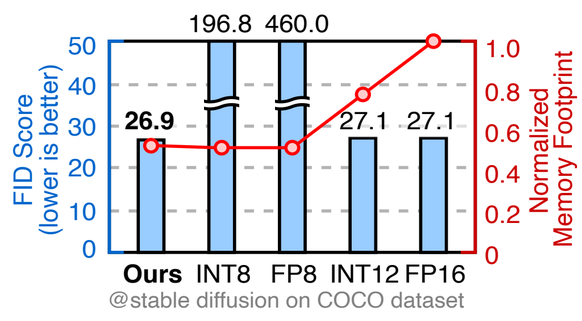
HYP8 supports upto INT13



Data Coverage of HYP8



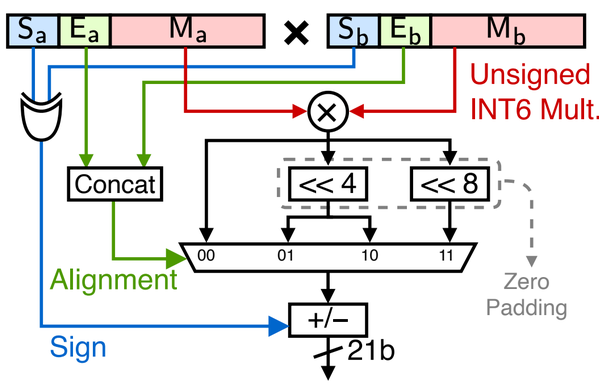
FID Score & Memory Footprint Graph



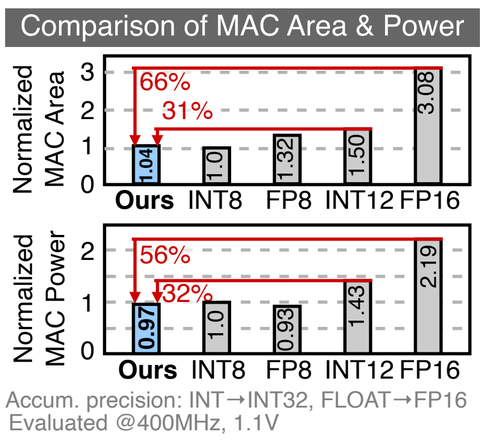
- Supports dual mode (near-zero range / extremely broad data range)
- Low memory footprint w/ no accuracy drop

Hyper-Efficient Reconfigurable Array

HYP8 Multiplier (HYPM)

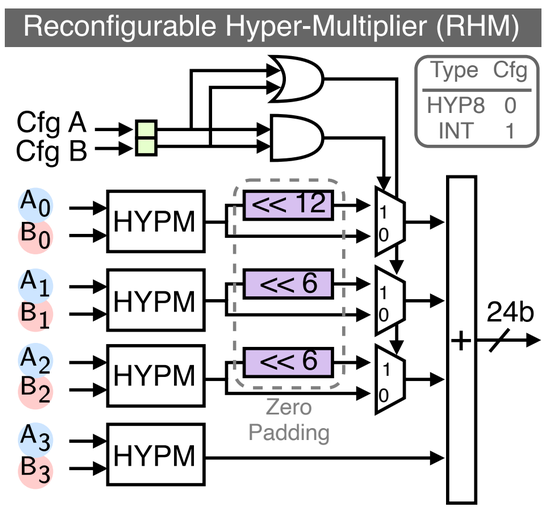


Comparison of MAC Area & Power

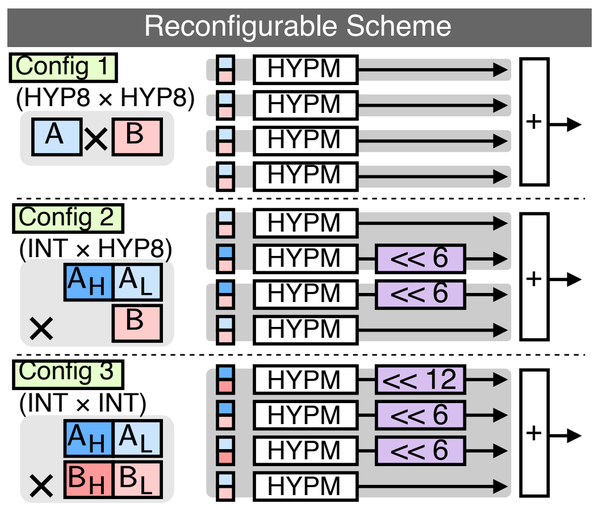


Accum. precision: INT→INT32, FLOAT→FP16
Evaluated @400MHz, 1.1V

Reconfigurable Hyper-Multiplier (RHM)



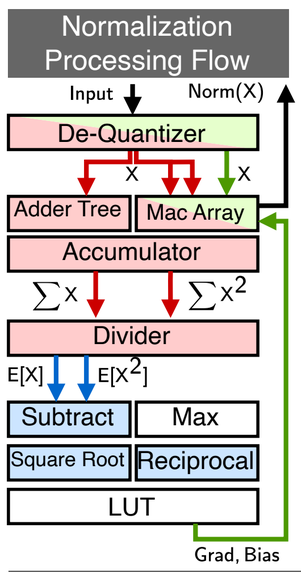
Reconfigurable Scheme



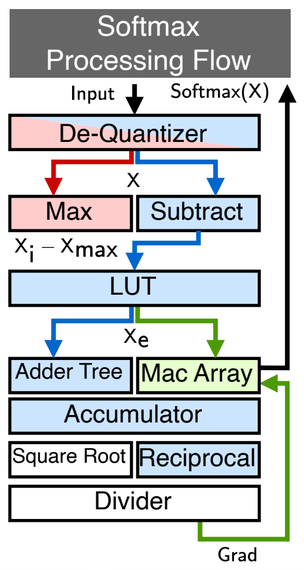
- HYPM supports HYP8 multiplication hardware-efficiently
- RHM supports INT8-INT13 w/ reconfigurable hardware for high accuracy

Unified Non-Matrix Processing Engine

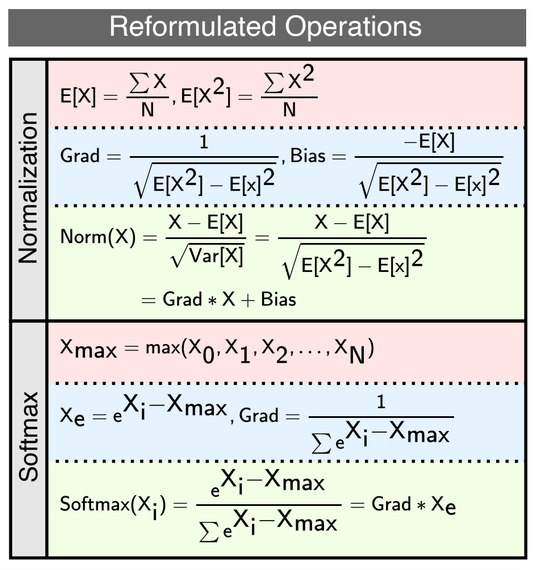
Normalization Processing Flow



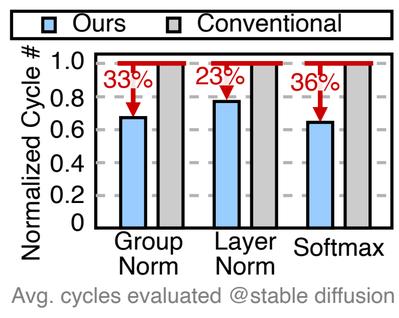
Softmax Processing Flow



Reformulated Operations

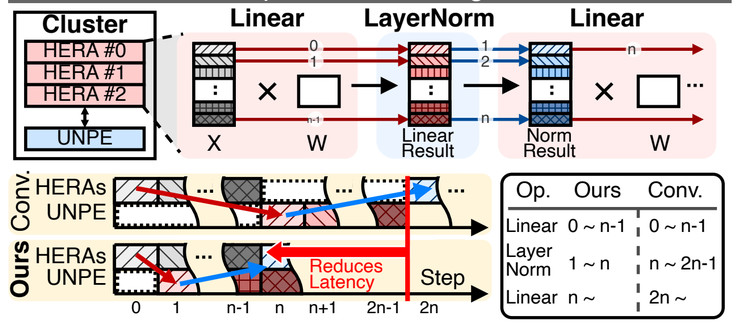


Latency of Non-Matrix Op.



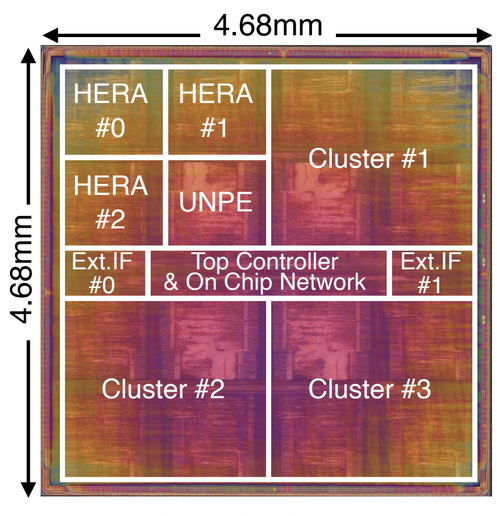
Avg. cycles evaluated @stable diffusion

Sub-Block Pipeline Scheduling in a Cluster



- Supports complex non-matrix operations efficiently by operation reformulation
- The sub-block pipeline scheduling maximizes the utilization of a cluster

Specification



Chip Photograph

	ISSCC'22 [2]	JSSC'22 [3]	ISSCC'23 [4]	ISSCC'23 [5]	**This Work**
Technology [nm]	28	16	12	28	28
Supported Networks	Transformer	Attention, RNN, Linear	Transformer	Transformer	Diffusion Model Transformer
Data Precison	INT12	FP8	FP4/FP8	INT8	HYP8/INT8-13
End-to-End Support[1] Diffusion Model	×	△	△	△	○
Accruacy (FID Score ↓)[2]	-	460.0	196.8	196.8	**26.9**
Die Area [mm²]	6.82	8.84	4.6	3.93	21.9
Core Voltage [V]	0.56 - 1.1	0.55 - 1.0	0.62 - 1.0	0.64 - 1.03	0.62 - 1.2
Core Frequency [MHz]	50 - 510	130 - 573	77 - 717	20 - 320	25 - 400
Peak Performance [TOPS or TFLOPS]	0.52 [5]	1.17	0.367 (FP8)	0.49 [5]	**9.83**
Energy Efficiency [3] [TOPS/W or TFLOPS/W]	4.25 [5]	4.46	1.77 (FP8) [5]	4.31 [5]	**4.96** [4]
Area Efficiency [3] [TOPS/mm² or TFLOPS/mm²]	0.0762 [5]	0.0432	0.0147 (FP8)	0.125 [5]	**0.449**

1) X: Not supporting normalization, △: Notable accuracy loss with 8-bit post-training quantization.
2) Accuracy of stable diffusion with PTQ applied at each precisions. (The FID Score @FP16 is 27.1)
3) Normalized to 28nm technology node: Energy efficiency ∝ (Technology / 28), Area efficiency ∝ (Technology / 28)²
4) Measured @0.62V, 25MHz, HYP8×HYP8 operation. 5) We assume no sparsity due to insufficient sparsity in diffusion model.

Acknowledgements

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No.2022-0-01036, Development of Ultra-Performance PIM Processor Soc with PFLOPS-Performance and GByte-Memory & No.2022-0-01037, Development of High Performance Processing-In-Memory Technology based on DRAM).

If you have any questions, please contact
sungyeob.yoo@kaist.ac.kr

Picasso: An Area/Energy-Efficient End-to-End Diffusion Accelerator with Hyper-Precision Data Type

Sungyeob Yoo, Geonwoo Ko, Seri Ham,
Seeyeon Kim, Yi Chen and Joo-Young Kim

KAIST CastLab

Abstract

This work presents **Picasso, an end-to-end diffusion accelerator** designed for enhancing the efficiency of diffusion-based machine learning models used in applications such as image and video generation, and inpainting. Picasso introduces a novel **hyper-precision 8 (HYP8) data type** and a **reconfigurable architecture** designed to significantly enhance hardware efficiency, providing an extended dynamic range without sacrificing accuracy. It also features a **unified engine** that streamlines the processing of **all non-matrix operations** and employs sub-block pipeline scheduling to reduce overall latency.

Fabricated in 28nm CMOS technology, this accelerator achieves an energy efficiency of **4.96 TOPS/W** and a peak performance of **9.83 TOPS**. Compared to previous works, Picasso demonstrates speedups ranging **from 8.4× to 26.8×** while also improving energy and area efficiency by **1.1× to 2.8×** and **3.6× to 30.5×**, respectively.

Diffusion Model

- The state-of-the-art powerful generative model
  - Diffusion models are transforming the generative AI market with their exceptional performance in emerging applications.

Multi-Modal Generation

"A photograph of an astronaut riding a horse"

Text-to-Image Generation

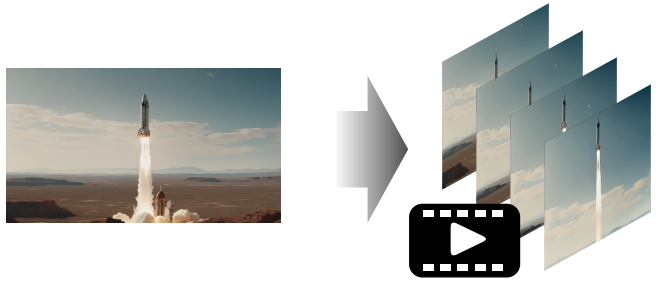
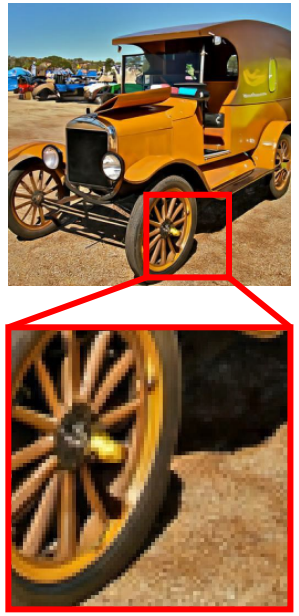


Image-to-Video Generation

Image Super-Resolution

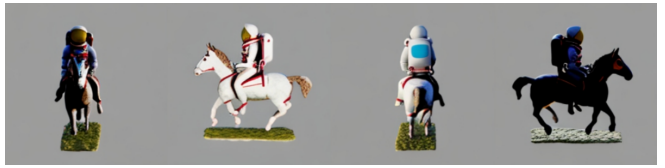
Input

Output



3D-Generation

"A DSLR photo of a ghost eating a hamburger"



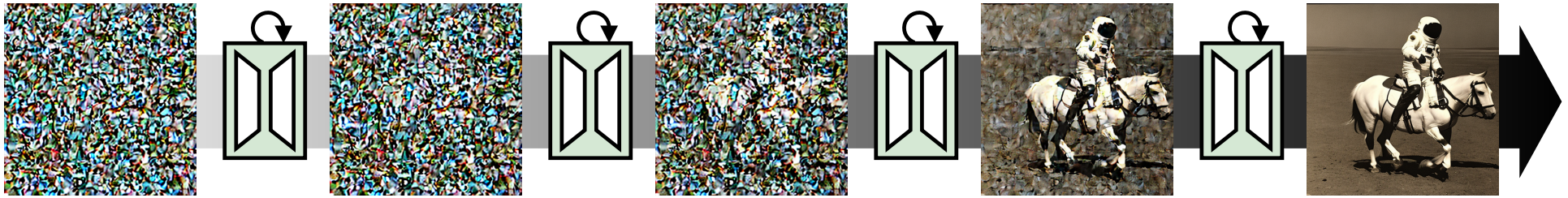
"An astronaut riding a horse"

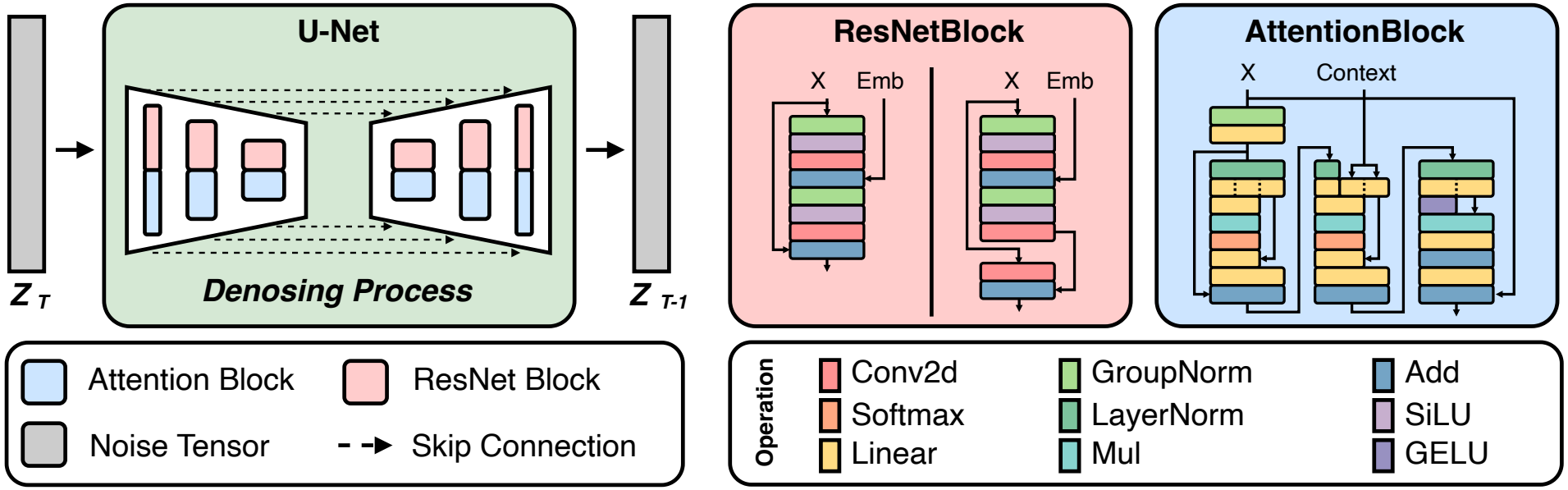
"A bald eagle carved out of wood"

* Rombach et al., "High-resolution image synthesis with latent diffusion models," (2022)
* Blattmann et al., "Stable video diffusion: Scaling latent video diffusion models to large datasets," (2023)
* Saharia et al., "Image super-resolution via iterative refinement," (2022)
* Shi et al., "Mvdream: Multi-view diffusion for 3d generation," (2023)

Diffusion Model

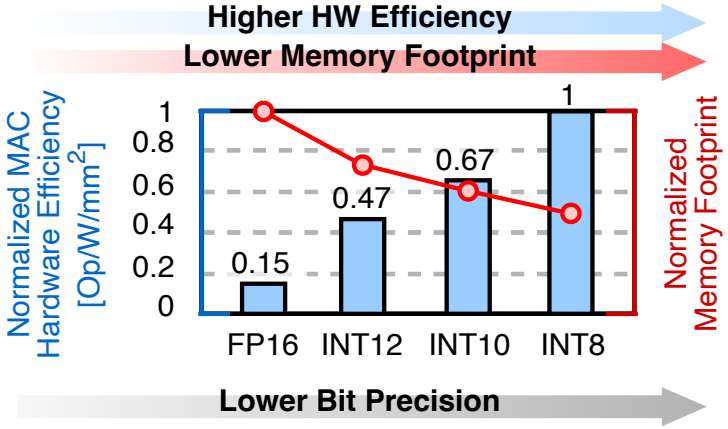
- Diffusion models start processing with initial random noise and perform **iterative denoising process**.
- U-Net architecture consists of ResNet blocks and Attention blocks.



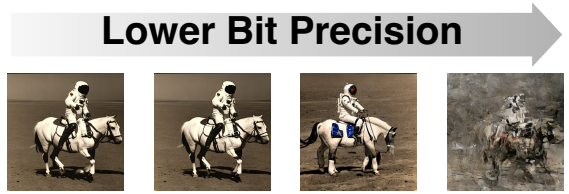


Challenges of Accelerating Diffusion Models

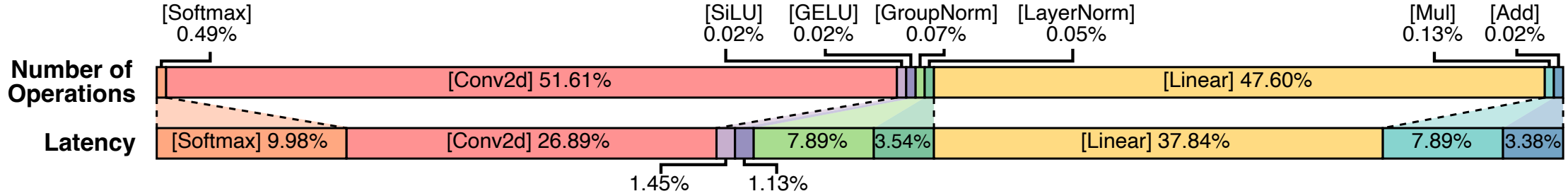
- Challenge 1: quantization with low bit precision
  - PTQ is free from retraining, but comes with an accuracy and bit precision trade-off.



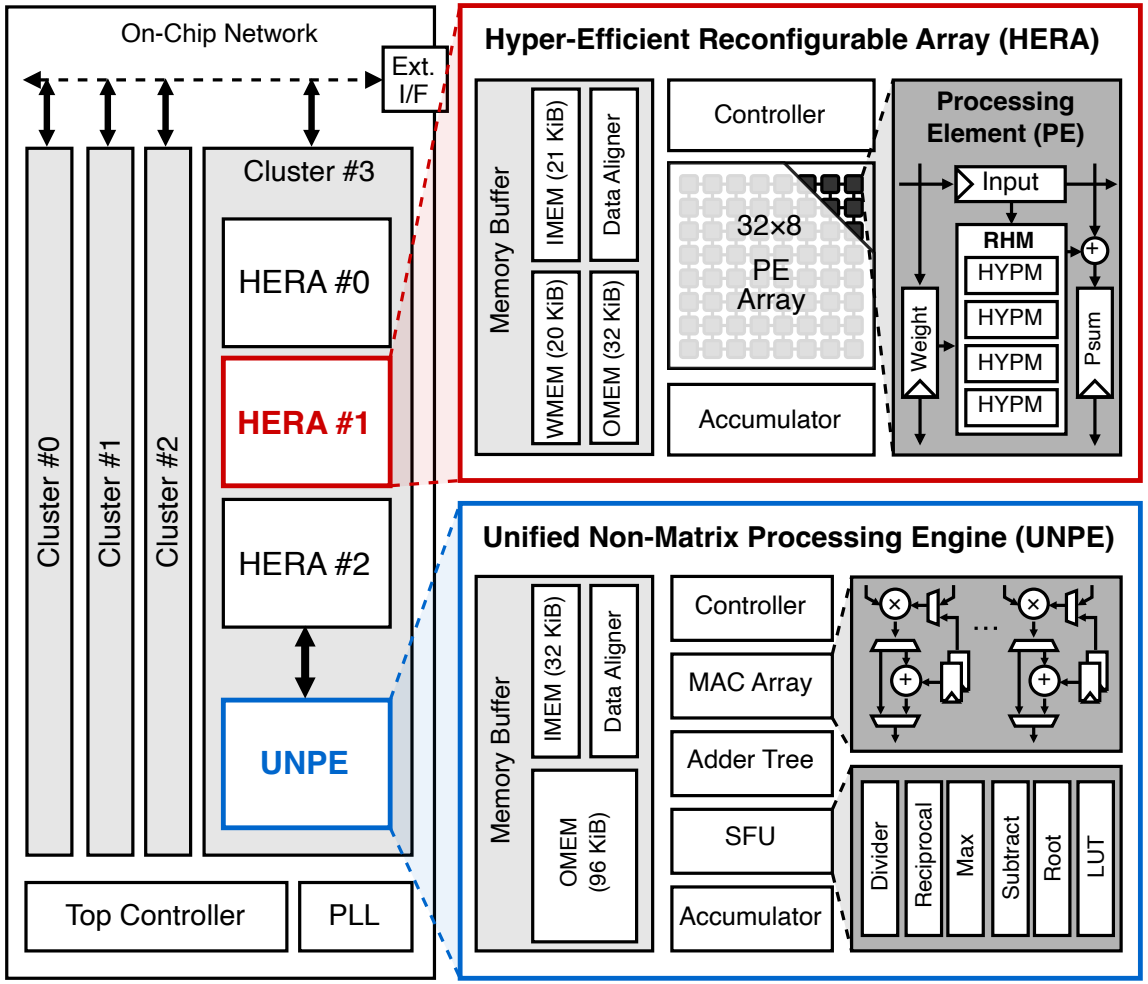
	QAT	PTQ
Retraining	O	X
Training Time	☹	☺
Data Requirements	☹	☺
Training Cost	☹	☺
Accuracy	☺	☹



- Challenge 2: optimizing the latency of non-matrix operations
  - Non-matrix operations accounts for a small portion in the operation count, but cause significant latency.



Overall Architecture

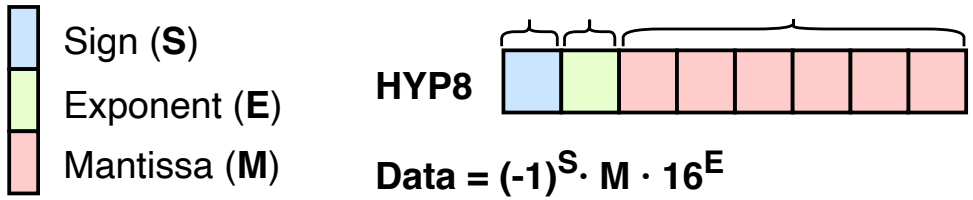


- **Hyper-Efficient Reconfigurable Array (HERA)**
  - Processes matrix operations (e.g., Conv2d and Linear).
  - Composition:
    - 32×8 systolic processing element (PE) array
    - Memory buffer with double buffering (for weight and input data)
    - Reconfigurable Hyper-Multiplier (RHM) with 4 HYP8 multipliers (HYPM)
    - Data aligner
- **Unified Non-Matrix Processing Engine (UNPE)**
  - Performs non-linear, element-wise, quantization, and de-quantization operations.
  - Composition:
    - MAC array
    - Adder tree
    - Special Function Unit (SFU)
    - Look-Up Table (LUT)
    - Accumulator
    - Memory buffer

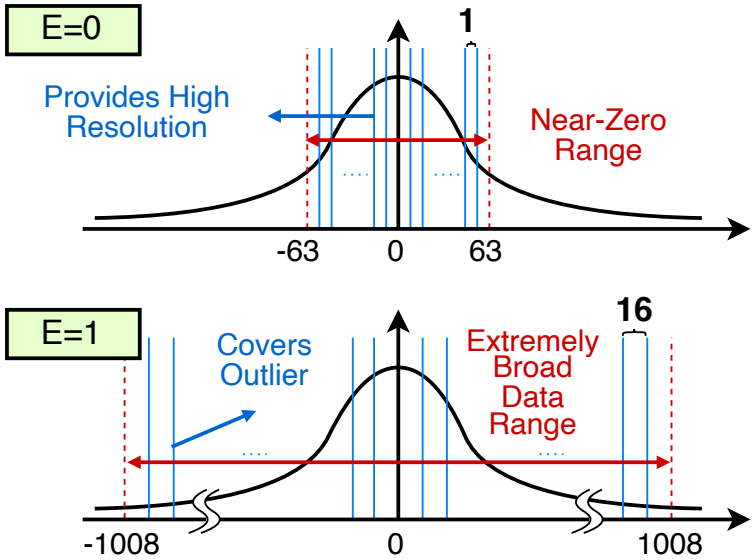
Hyper-Precision Data Type

- Hyper-Precision 8 (HYP8) provides an extended dynamic range and efficient memory footprint without any accuracy loss in diffusion model.

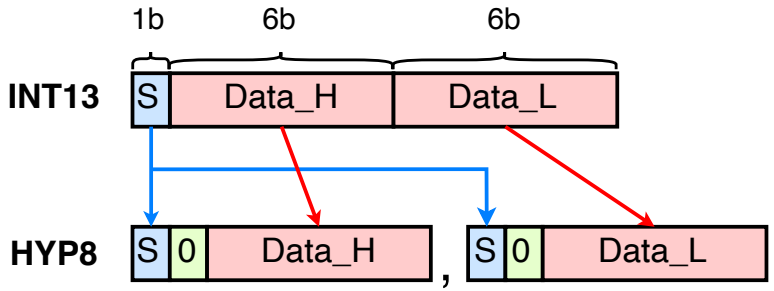
Configuration of HYP8



Data Coverage of HYP8

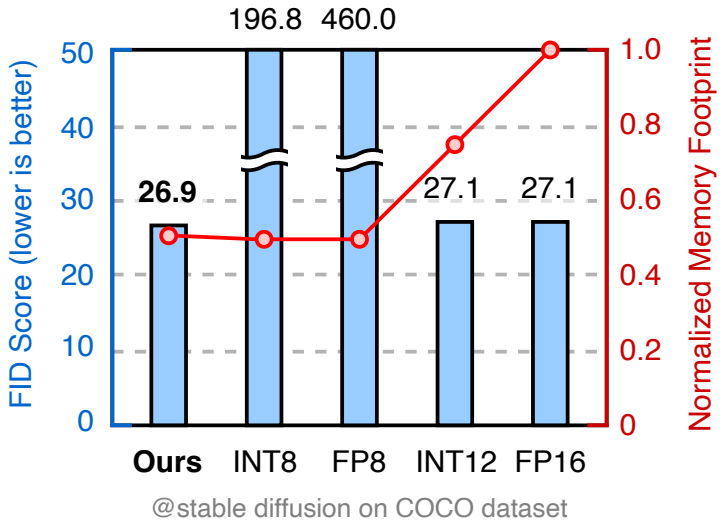


Precision Enhancement of HYP8



- Combining two HYP8 data enables the precision of INT8-13.

FID Score & Memory Footprint

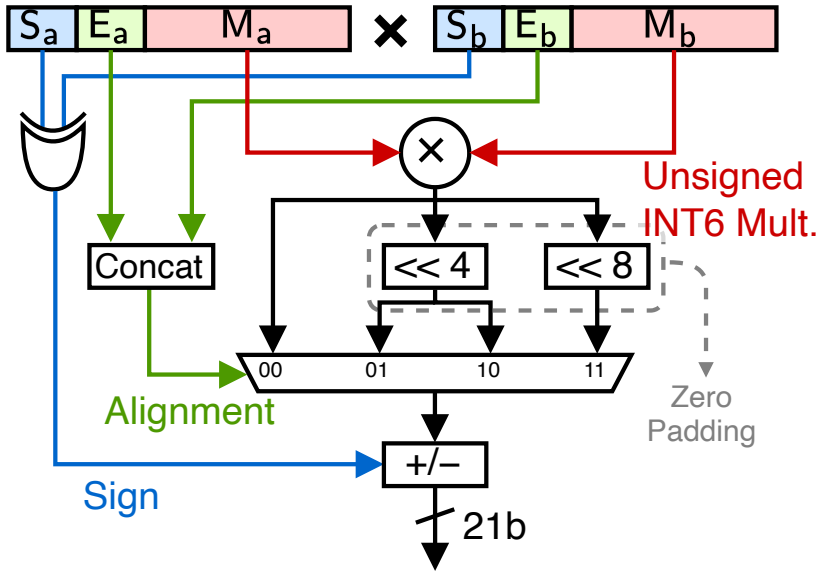


@stable diffusion on COCO dataset

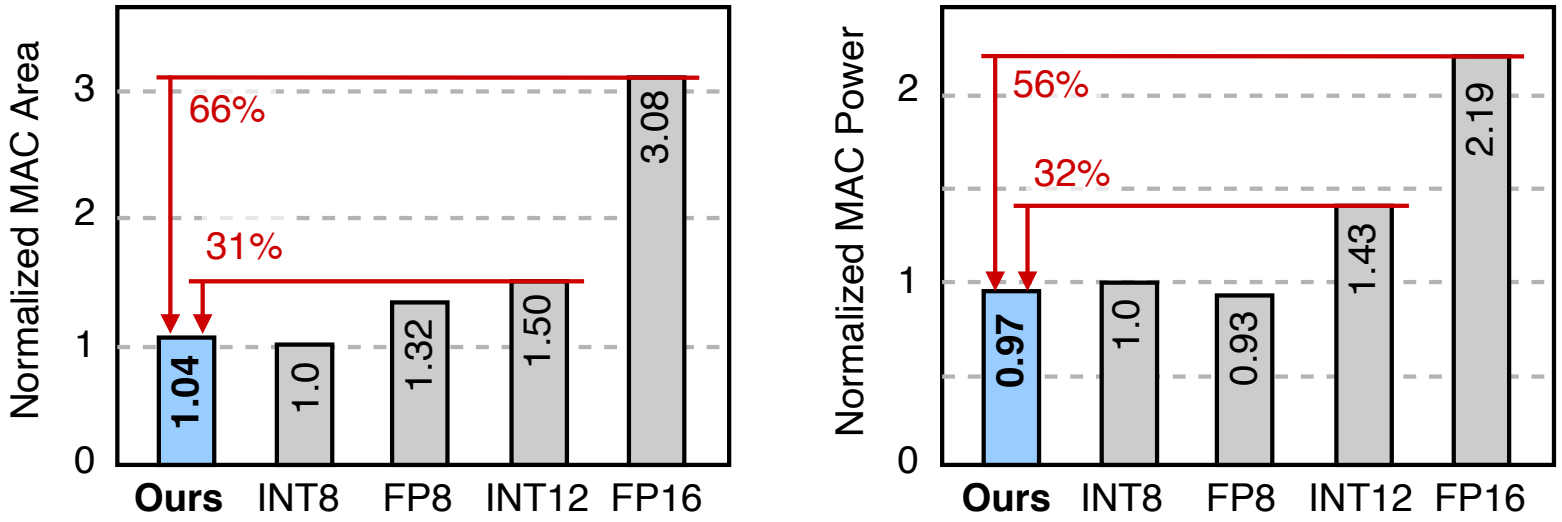
HYP8 Multiplier

- HYP8 Multiplier (HYPM) supports HYP8 multiplication while maximizing hardware efficiency.
- HYPM is area- and power-efficient compared to other MACs.

HYPM Architecture



Comparison of MAC Area and Power

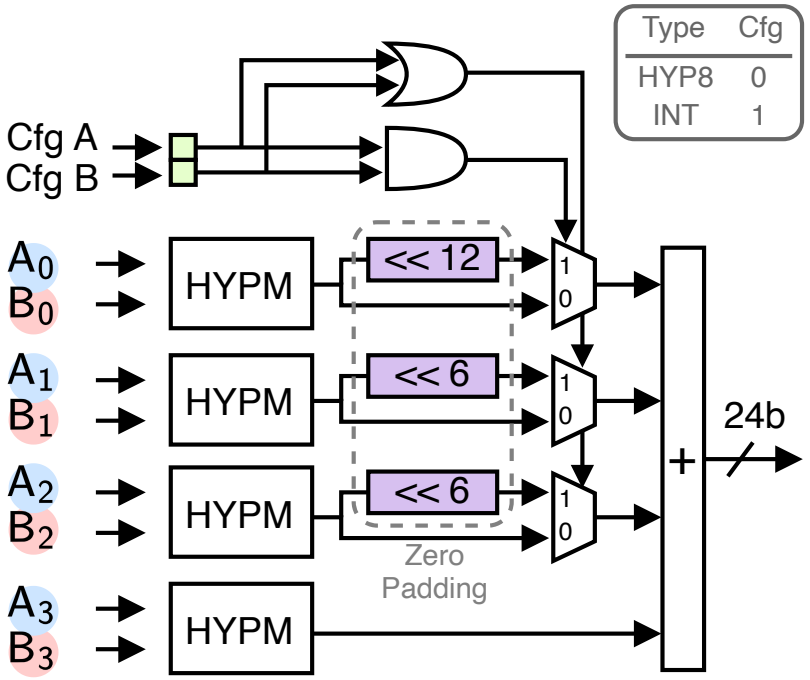


Evaluated @ 400MHz, 1.1V | Accumulation precision: INT → INT32, FLOAT → FP16

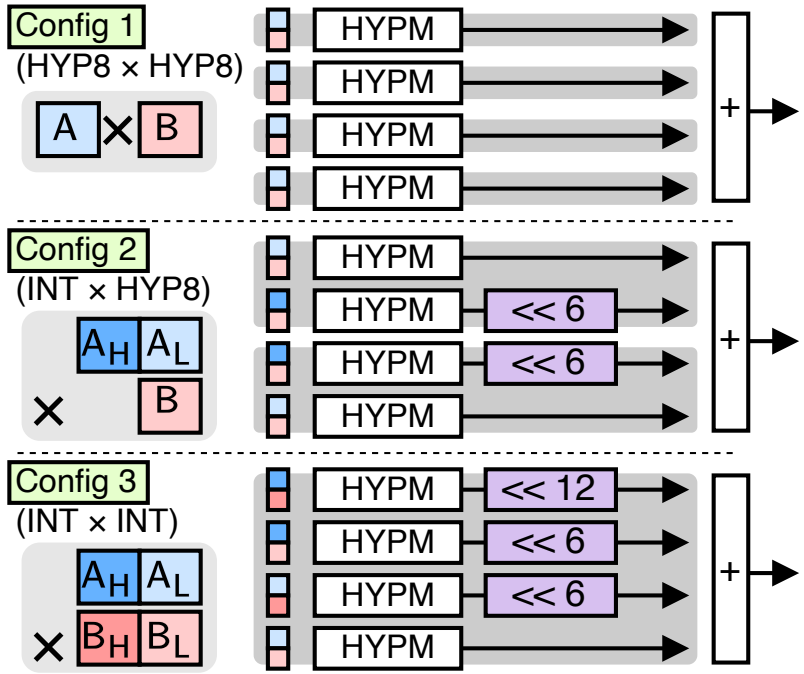
Reconfigurable Hyper-Multiplier

- Reconfigurable Hyper-Multiplier (RHM) supports both HYP8 and INT operations in a reconfigurable manner using 4 HYPMs.
- RHM provides 3 different configuration schemes depending on the input data type.
  - Using RHM, we can balance between better throughput and higher accuracy.

RHM Architecture



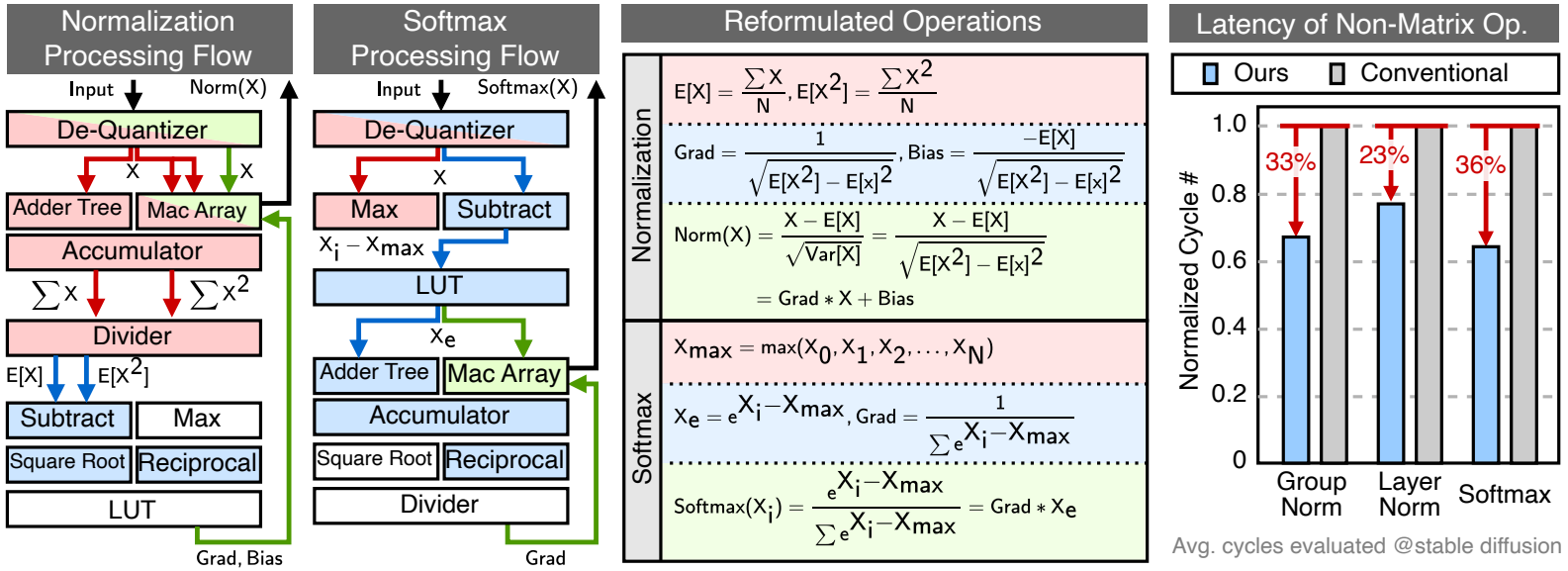
Reconfigurable Scheme



Unified Non-Matrix Processing Engine & Sub-Block Pipeline Scheduling

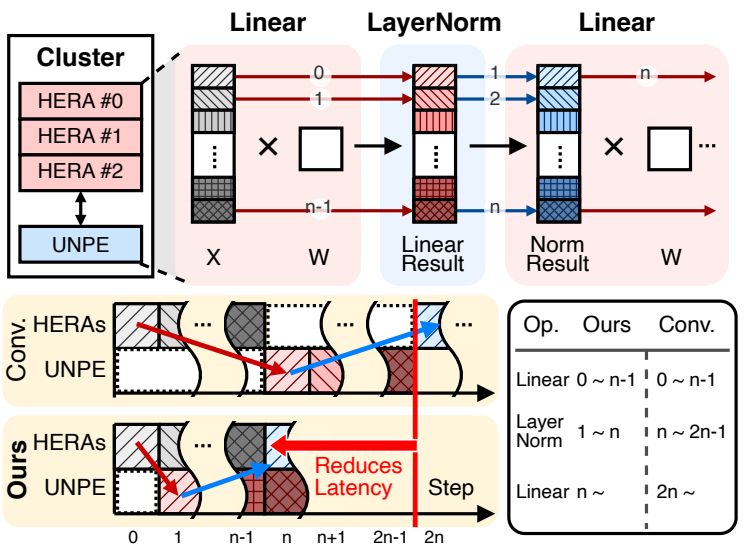
- Unified Non-Matrix Processing Engine (UNPE) efficiently supports various complex non-matrix operations in a single unified engine.
  - Streamlines normalization and softmax operations into 2 steps:
    ① Computing coefficients (gradient & bias) ② Performing linear operation
  - Minimizes non-matrix operation latency.
- Sub-block pipeline scheduling optimizes end-to-end latency and maximizes cluster utilization.
  - Tasks are divided into sub-blocks, allowing HERAs and UNPE to operate concurrently within the pipeline.

Optimizing Non-Matrix Operations in UNPE



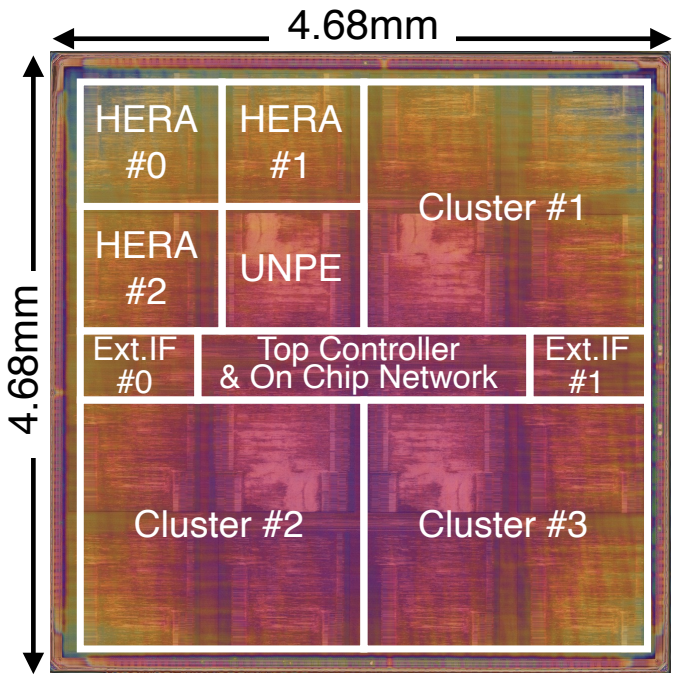
Avg. cycles evaluated @stable diffusion

Sub-Block Pipeline Scheduling



Chip Summary

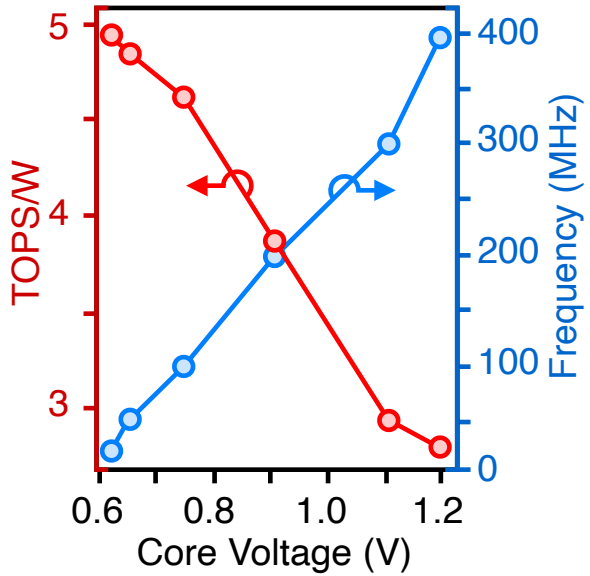
Chip Photograph



Chip Specification

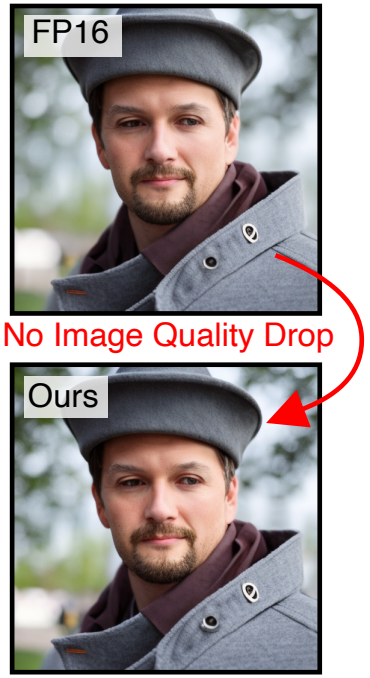
Technology	28 nm
On-Chip Memory	1.36 MiB
Data Precision	HYP8 / INT8 - INT13
Die Area	21.9 mm^2
Core Voltage	0.62 - 1.2 V
Core Frequency	25 - 400 MHz
Peak Performace	9.83 TOPS (HYP8)
Energy Efficiency	4.96 TOPS/W
Area Efficiency	0.449 TOPS/mm^2

Voltage-Frequency Scaling



Evaluated @HYP8×HYP8 operation, stable diffusion on COCO dataset

Generated Image



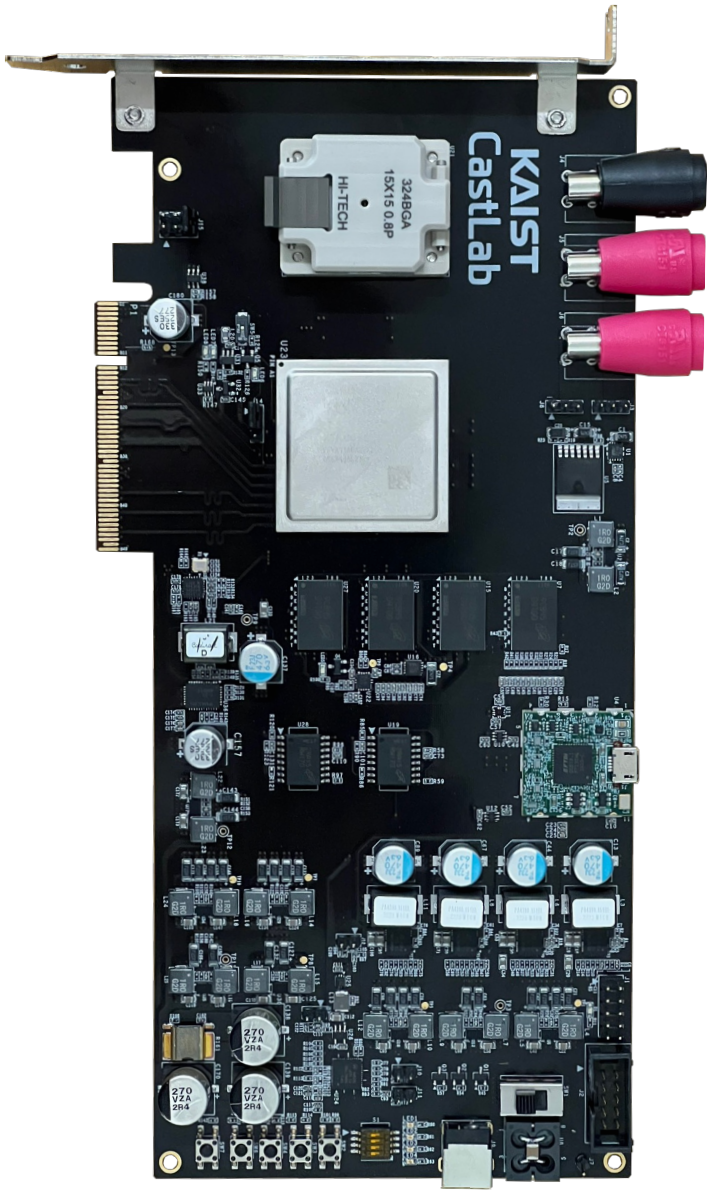
Performance Comparison

- Picasso shows an **8.4× to 26.8× speed-up**, **1.1× to 2.8× higher energy efficiency**, and **3.6× to 30.5× higher area efficiency**.

	ISSCC'22	JSSC'22	ISSCC'23	ISSCC'23	**This Work**
Technology [nm]	28	16	12	28	28
Supported Networks	Transformer	Attention, RNN, Linear	Transformer	Transformer	Diffusion Model Transformer
Data Precison	INT12	FP8	FP4/FP8	INT8	HYP8/INT8-13
End-to-End Support [1] Diffusion Model	×	△	△	△	○
Accruacy (FID Score ↓) [2]	-	460.0	196.8	196.8	**26.9**
Die Area [mm^2]	6.82	8.84	4.6	3.93	21.9
Core Voltage [V]	0.56 - 1.1	0.55 - 1.0	0.62 - 1.0	0.64 - 1.03	0.62 - 1.2
Core Frequency [MHz]	50 - 510	130 - 573	77 - 717	20 - 320	25 - 400
Peak Performance [TOPS or TFLOPS]	0.52 [5]	1.17	0.367 (FP8)	0.49 [5]	**9.83**
Energy Efficiency [3] [TOPS/W or TFLOPS/W]	4.25 [5]	4.46	1.77 (FP8) [5]	4.31 [5]	**4.96** [4]
Area Efficiency [3] [$TOPS/mm^2$ or $TFLOPS/mm^2$]	0.0762 [5]	0.0432	0.0147 (FP8)	0.125 [5]	**0.449**

1) X: Not supporting normalization, △: Notable accuracy loss with 8-bit post-training quantization.
2) Accuracy of stable diffusion with PTQ applied at each precisions. (The FID Score @FP16 is 27.1)
3) Normalized to 28nm technology node: Energy efficiency ∝ (Technology / 28), Area efficiency ∝ (Technology / 28)^2
4) Measured @0.62V, 25MHz, HYP8×HYP8 operation. 5) We assume no sparsity due to insufficient sparsity in diffusion model.

Summary



- **Picasso is a 28nm end-to-end diffusion accelerator** designed to maximize hardware efficiency without sacrificing accuracy, featuring:
  - Hyper-Precision Data Type (HYP8)
  - Hyper-Efficient Reconfigurable Array (HERA)
  - Unified Non-Matrix Processing Engine (UNPE)
- Picasso achieves **up to 26.8×** better performance, **2.8×** better energy efficiency, and **30.5×** better area efficiency than prior accelerators.
- Picasso maintains high image quality comparable to FP16 without accuracy degradation.

Thank you

- Any questions? Feel free to contact us!
  - E-mail: sungyeob.yoo@kaist.ac.kr
  - Slack: #p-kaist-picasso
  - Lab website: https://castlab.kaist.ac.kr

Sungyeob Yoo

- Acknowledgement

  This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No.2022-0-01036, Development of Ultra-Performance PIM Processor Soc with PFLOPS-Performance and GByte-Memory & No.2022-0-01037, Development of High Performance Processing-In-Memory Technology based on DRAM).