SAMSUNG

Aquabolt-XL: Samsung HBM2-PIM with in-memory processing for ML accelerators and beyond

Jin Hyun Kim, Shin-haeng Kang, Sukhan Lee, Hyeonsu Kim, Woongjae Song, Yuhwan Ro, Seungwon Lee, David Wang, Hyunsung Shin, Bengseng Phuah, Jihyun Choi, Jinin So, YeonGon Cho, JoonHo Song, Jangseok Choi, Jeonghyeon Cho, Kyomin Sohn, Youngsoo Sohn, Kwangil Park, and Nam Sung Kim

Samsung Electronics

ACKNOWLEDGMENT: We have partners and customers that we discussed and shared many ideas with. Nevertheless, unlike companies that cannot open due to NDA issue, we must mention who helped with HW, SW, and system enabling. Hsien-Hsin Sean Lee, Liu Ke, from Facebook for AXDIMM and Yun Wu, Arun Vr Varadarajan Rajagopal, Matt Klein, Kumar Subramany, Henley Liu, from Xilinx for U280-PIM

Contents

1. Processing-in-Memory Concept

2. Aquabolt-XL HBM2-PIM

- Hardware Architecture
- Software Stack and Programming Model

3. Aquabolt-XL system evaluation

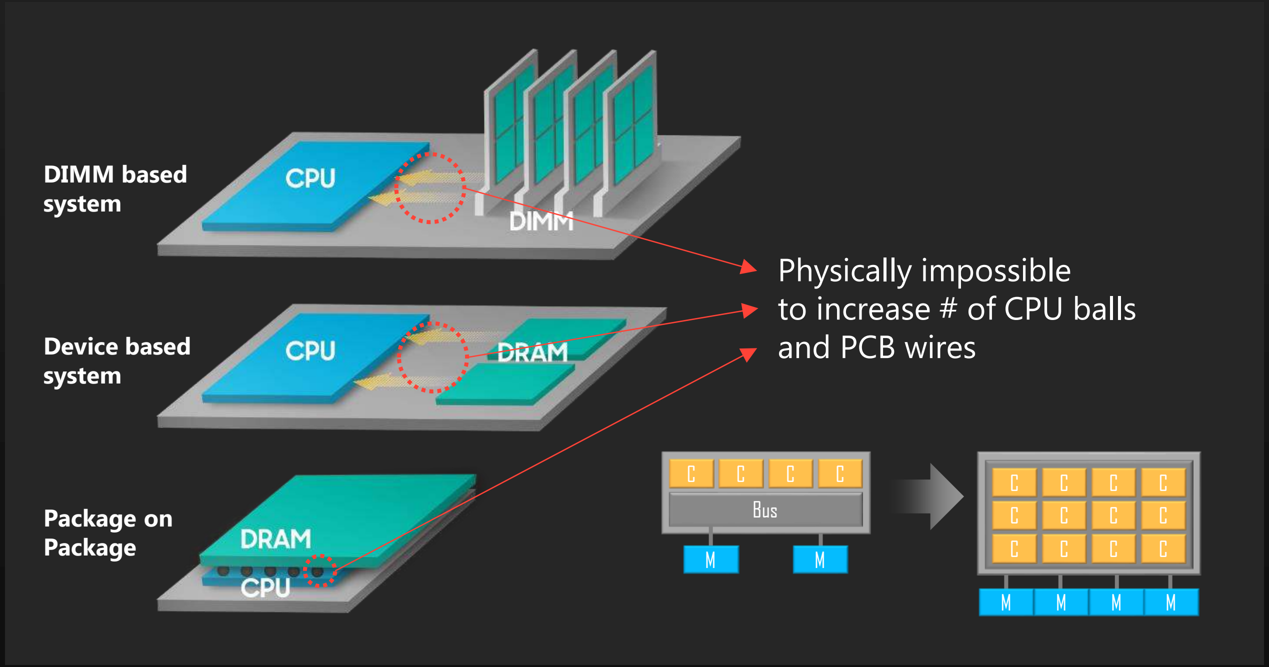
- Aquabolt-XL with GPU
- Aquabolt-XL with Xilinx FPGA (U280)

4. Next PIM solution

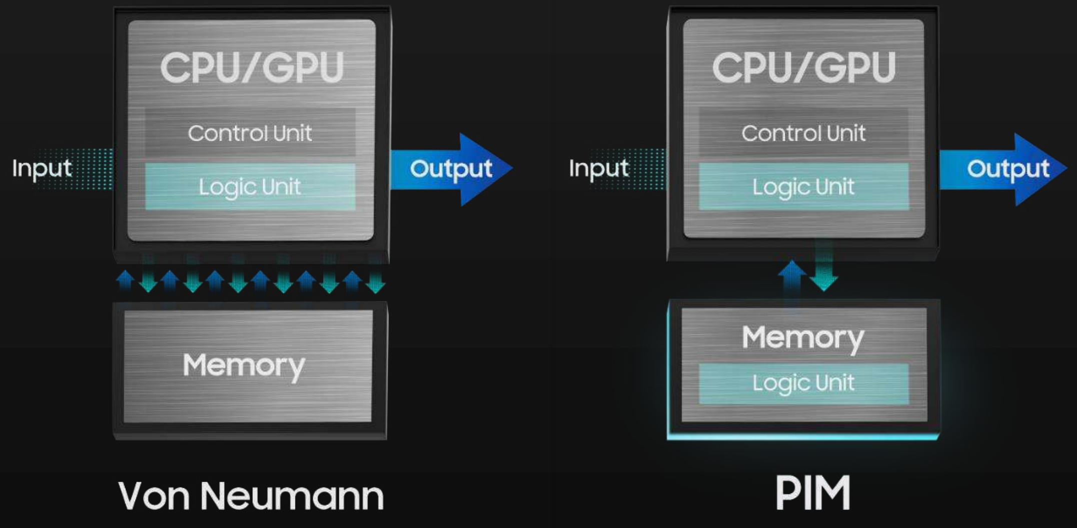
- LPDDR5-PIM
- AXDIMM, DIMM-PIM

Using PIM to overcome memory bottleneck

- Although various bandwidth increase methods have been proposed, it is physically impossible to achieve a breakthrough increase.
  - Limited by # of PCB wires, # of CPU ball, and thermal constraints
- PIM has been proposed to improve performance of bandwidth-intensive workloads and improve energy efficiency by reducing computing-memory data movement.



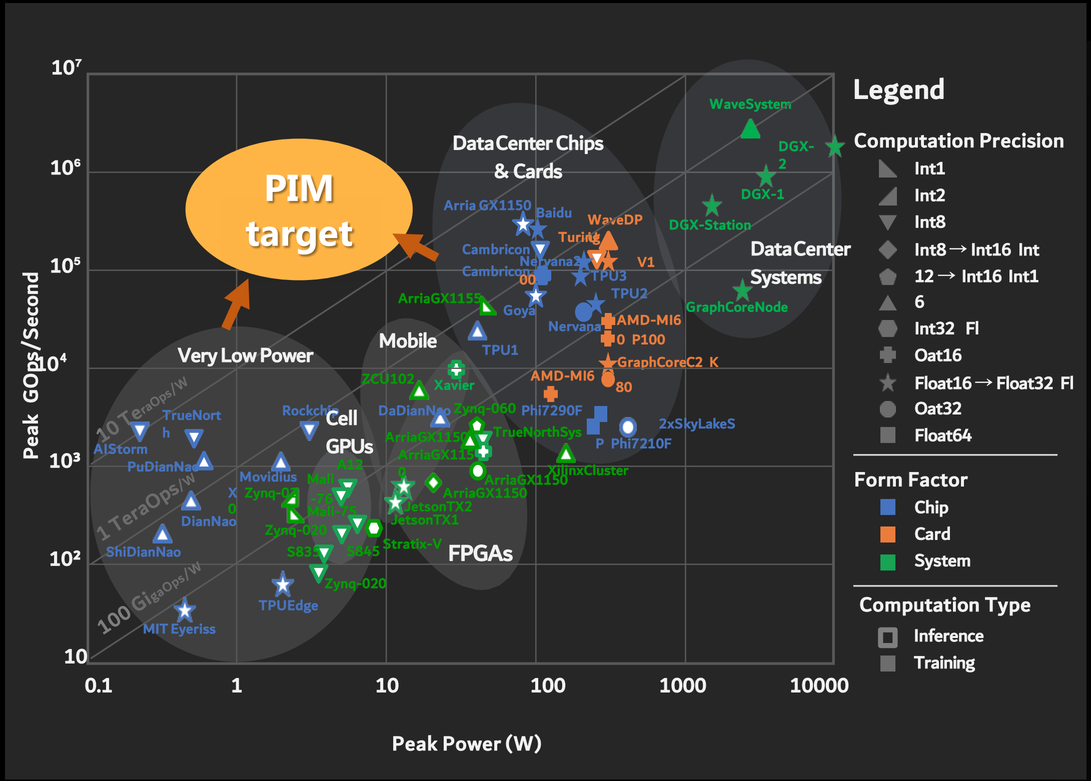
Physically impossible to increase # of CPU balls and PCB wires

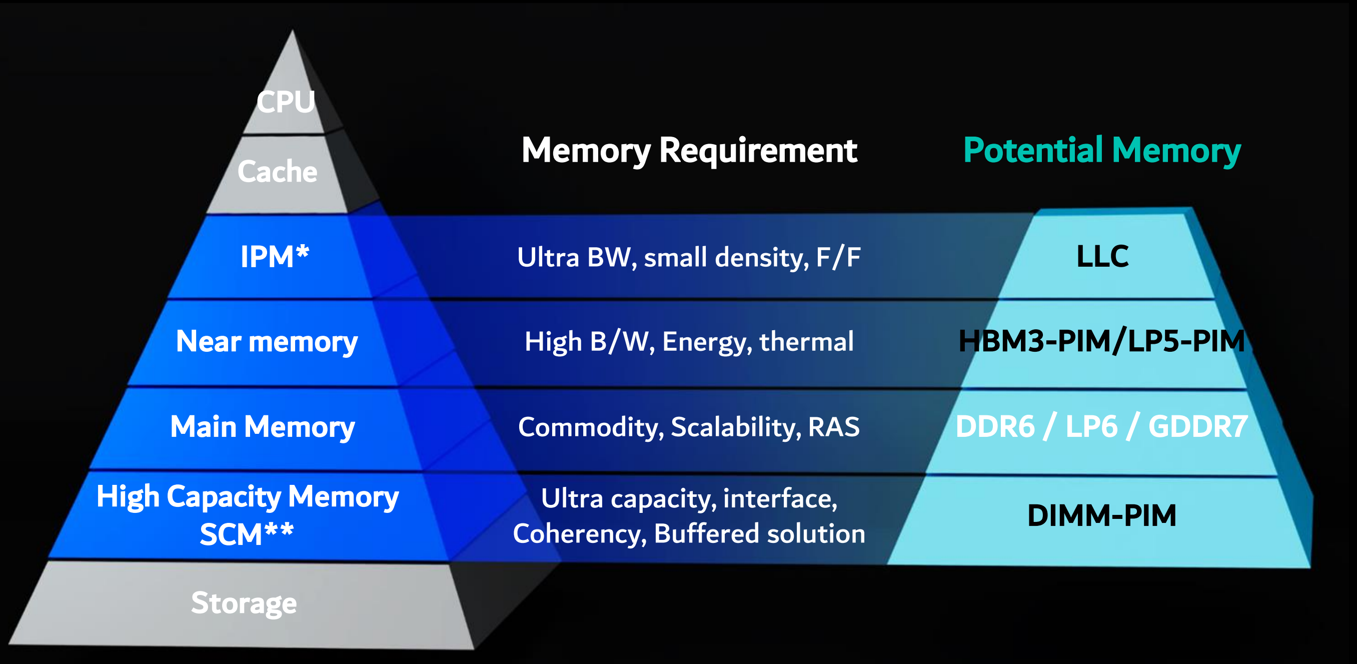


SAMSUNG

Re-thinking PIM and Re-architecting memory hierarchy

- PIM provides high ops/second and low power [Survey and Benchmarking of Machine Learning Accelerators]
  not only extension of commodity but also various hierarchical optimized solutions have been proposed.
- Closes performance gap and delivers energy-efficient solutions [Hotchips '16, Jin Hyun Kim, Samsung]
  - PIM in LLC/Near memory/DIMM





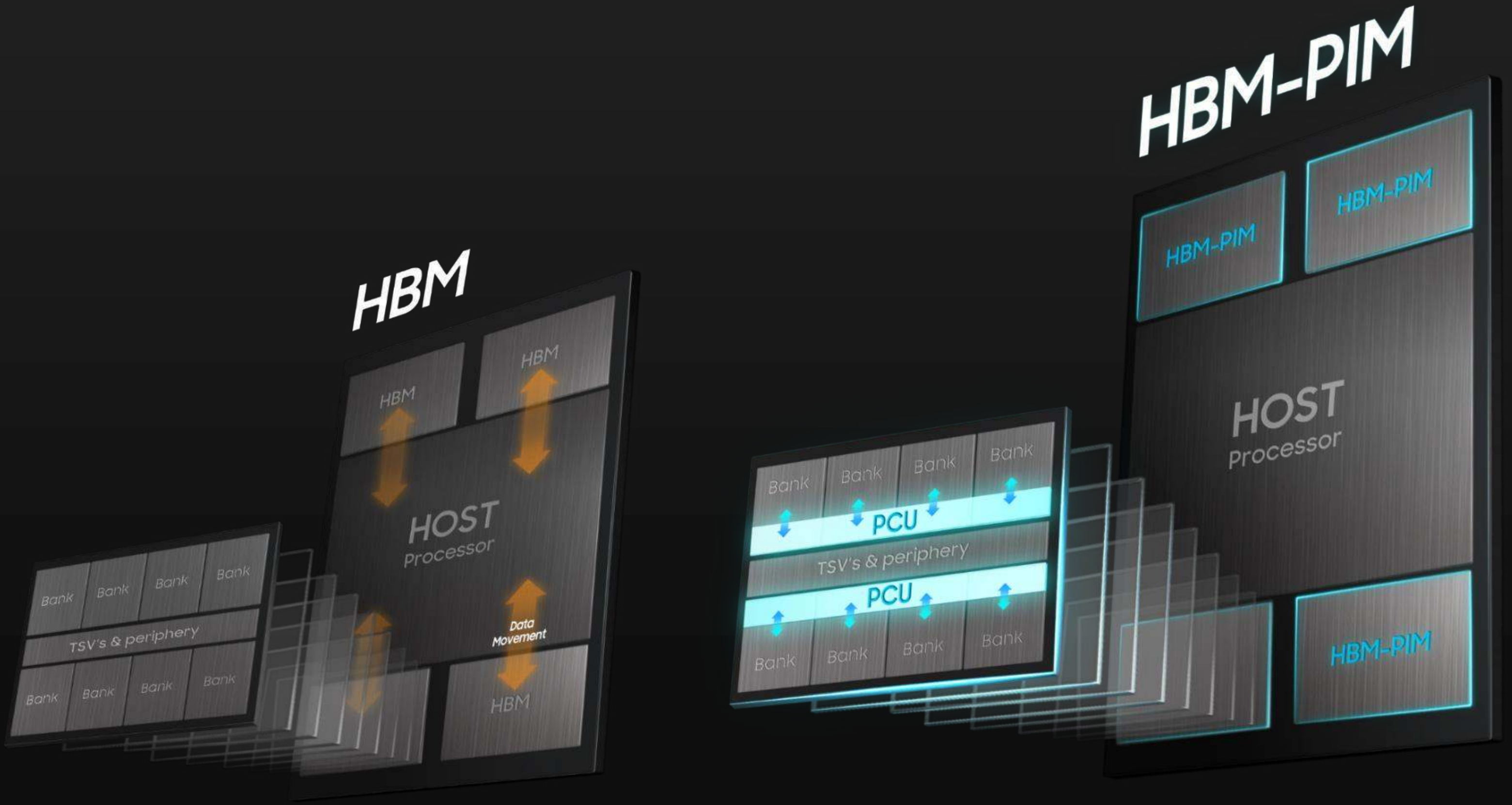
	Memory Requirement	Potential Memory
IPM*	Ultra BW, small density, F/F	LLC
Near memory	High B/W, Energy, thermal	HBM3-PIM/LP5-PIM
Main Memory	Commodity, Scalability, RAS	DDR6 / LP6 / GDDR7
High Capacity Memory SCM**	Ultra capacity, interface, Coherency, Buffered solution	DIMM-PIM

* : In package memory
**: Storage Class Memory

Aquabolt-XL, System-Level 1st PIM memory

- The first demonstrator vehicle of PIM is based on HBM2 Aquabolt, which is used in leading edge AI and HPC systems.
- PCU* is integrated with a memory core on a single chip to enable parallel processing and minimize data movement.
- Improves the performance and energy efficiency of the system with in-DRAM processing
  - Performance : Utilize up to 4× higher in-DRAM bandwidth by multi-bank parallel operation
  - Energy Efficiency : Reduce data movement energy by utilizing in-DRAM data processing unit

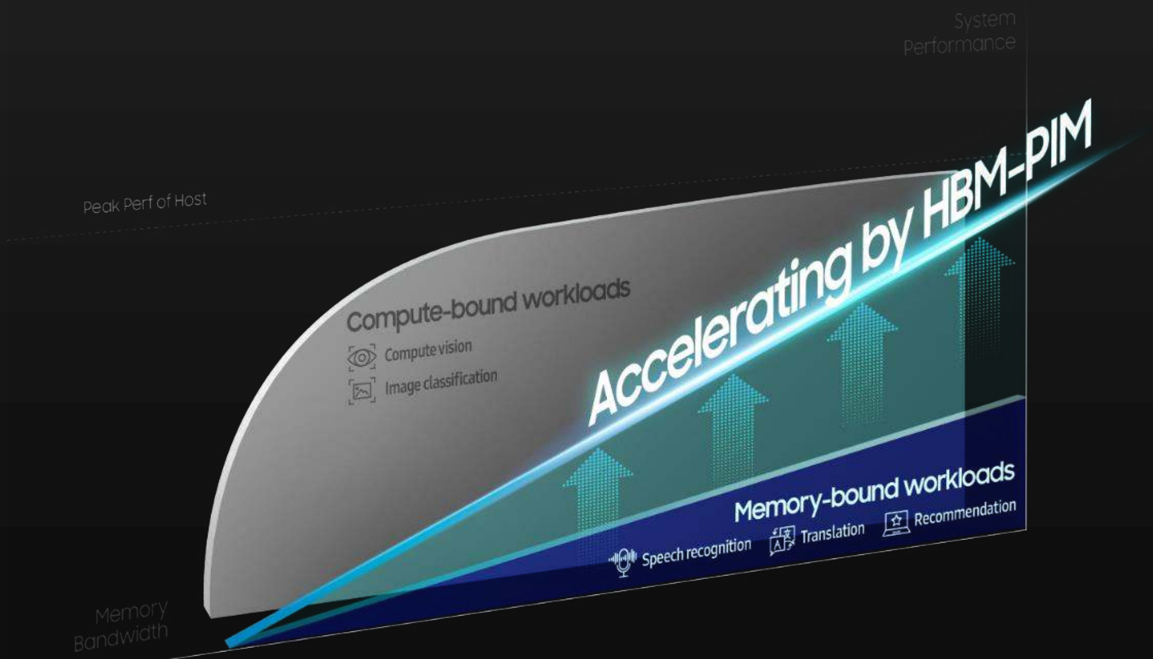
*: Programmable Computing Unit

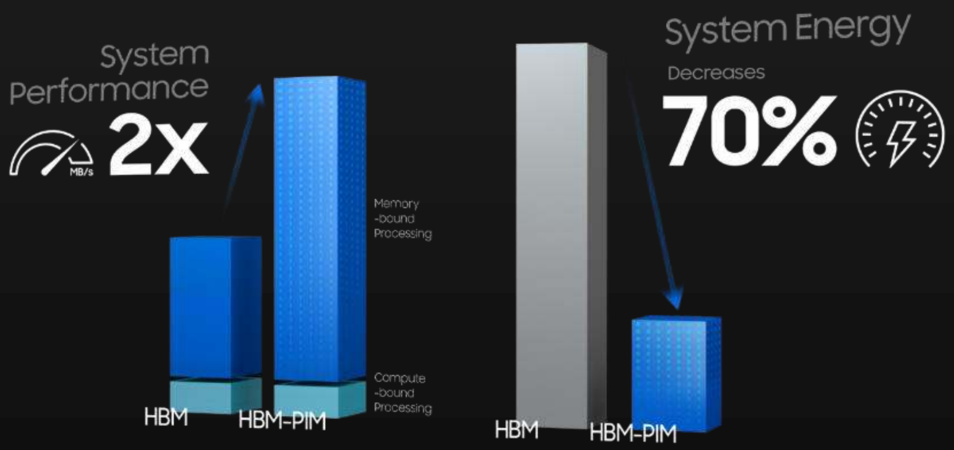


SAMSUNG

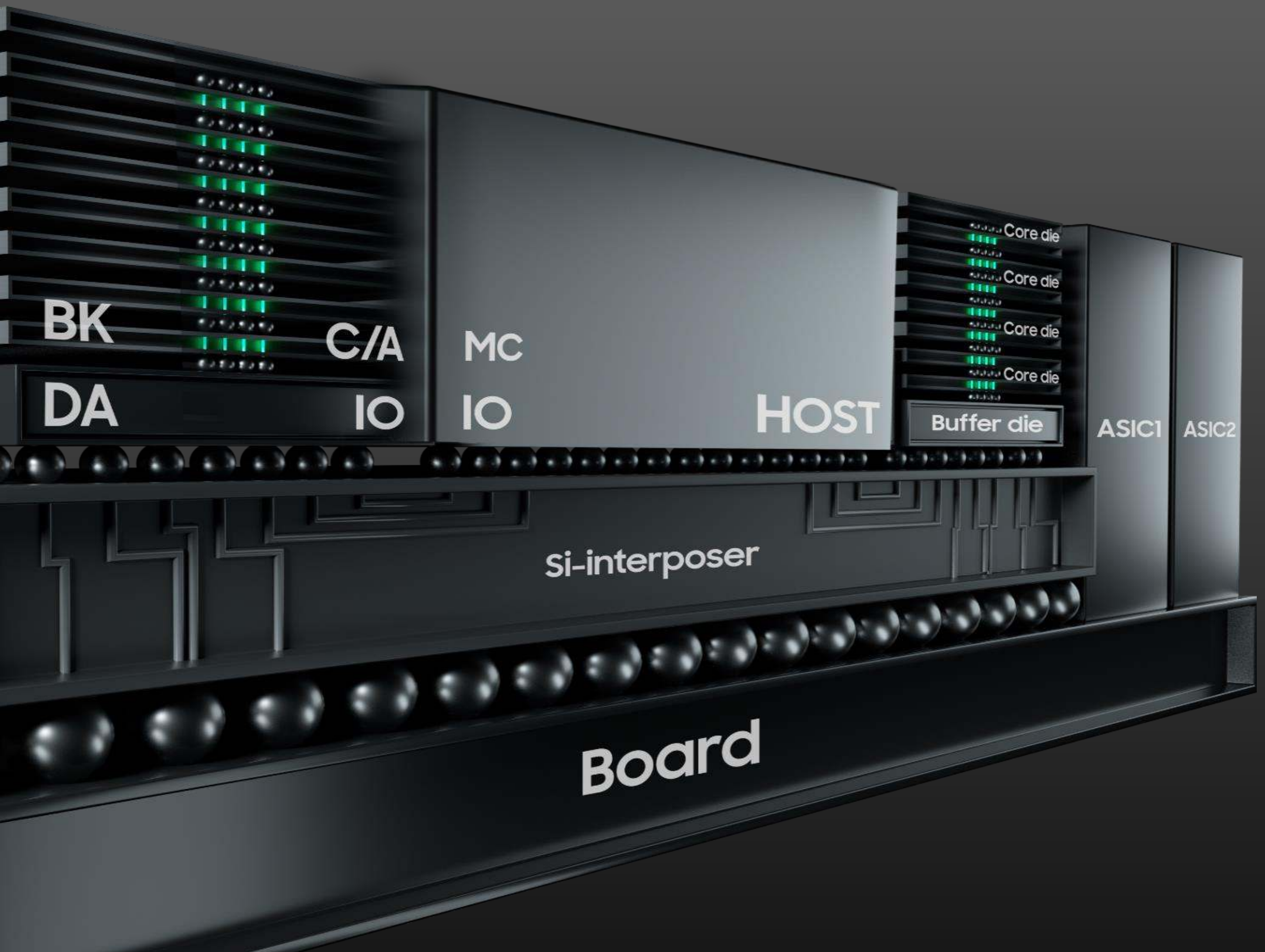
Aquxbolt-XL, System-Level Evaluation

- XPUs are highly performant for compute-bound workloads but expend significant time and energy processing memory-bound workloads with low arithmetic intensity.
- PIM naturally complements xPUs for optimal system balance and performance per watt for memory-bound workloads
  - Speech Recognition, Natural Language Translation, Recommendation
- Aquabolt-XL is able to deliver over 2X system performance while reducing energy consumption by more than 70%.





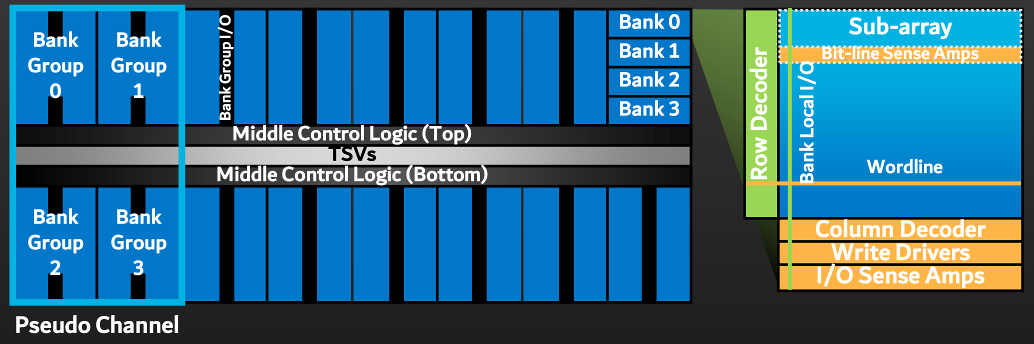
Aquabolt HBM2 (High Bandwidth memory)



System in Package (SiP)

- HBM2 provides high memory bandwidth (more than 256GB/s) by 3D-stacking up to eight DRAM dies
  - DRAM dies communicate with the buffer die using through silicon vias (TSVs)
  - Buffer die is connected to a host xPU with silicon interposer
- An HBM2 stack of eight DRAM dies (8-Hi) has two 64-bit channels per die for a total of 16 pseudo channels and a width of 1024 bits
  - Each pair of two DRAM dies share host memory bus In 8-Hi HBM, which doubles the memory capacity, not bandwidth compared to 4-Hi HBM

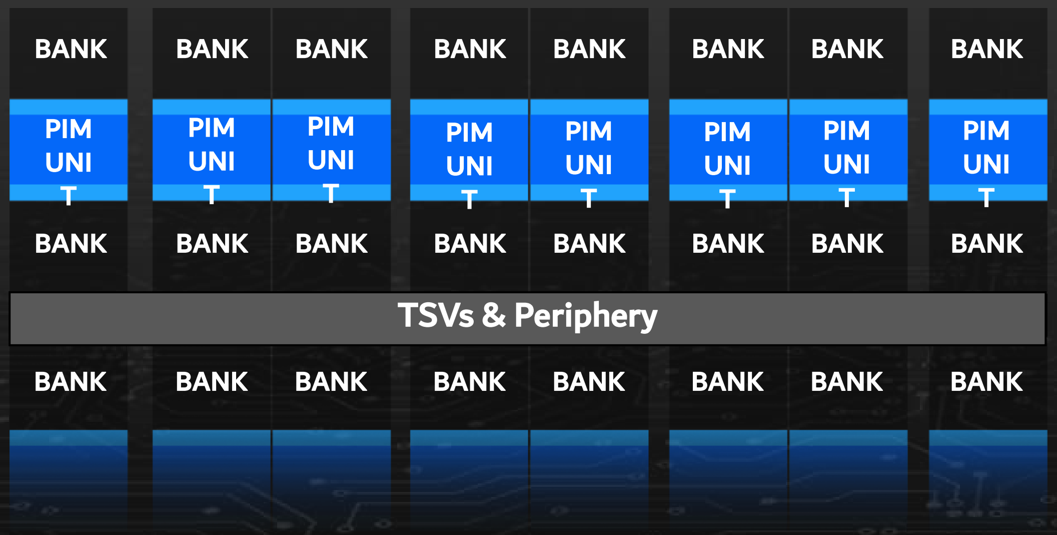
HBM core die



Aquabolt-XL HBM2-PIM Architecture

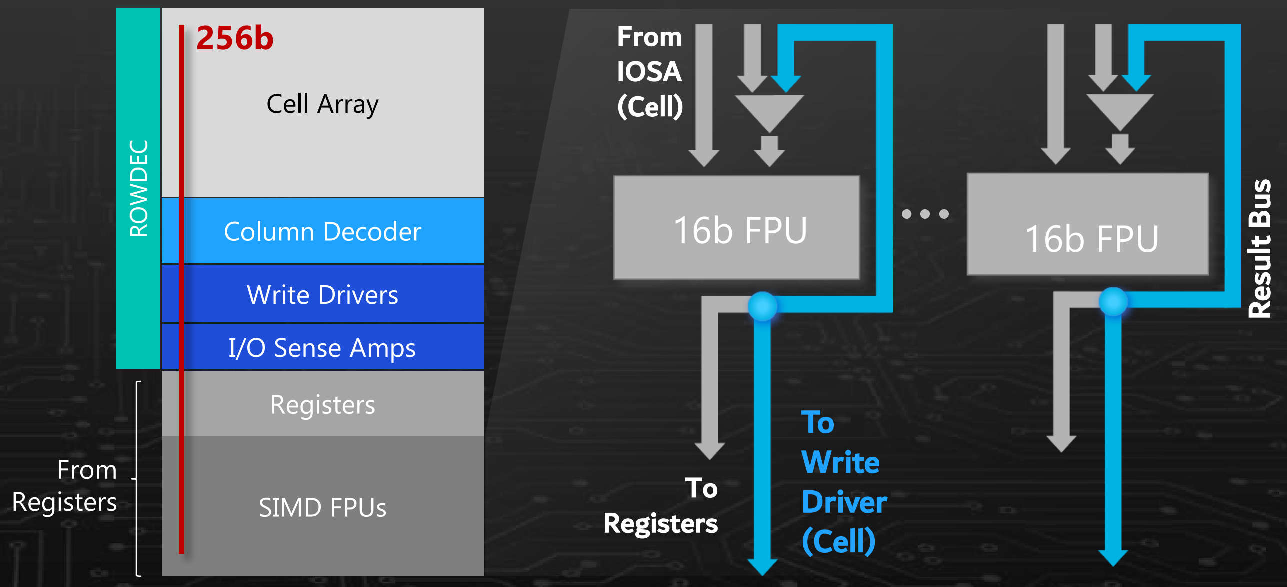
- Place a programmable PIM execution unit at the I/O boundary of a bank based on HBM2
  - Exploit bank-level parallelism: access multi banks/FPUs in a lockstep manner
  - Support both standard HBM and Aquabolt-XL modes for versatility
  - Minimize engineering cost of re-designing DRAM core to support PIM
- Maintain the same form-factor and timing parameters as baseline Aquabolt product
  - Facilitate drop-in replacement of JEDEC-specification compliant Aquabolt HBM2 with Aquabolt-XL HBM-PIM for any system
- DRAM RD/WR command triggers execution of a PIM instruction in PIM mode
  - Preserving deterministic DRAM timing

HBM Core Die



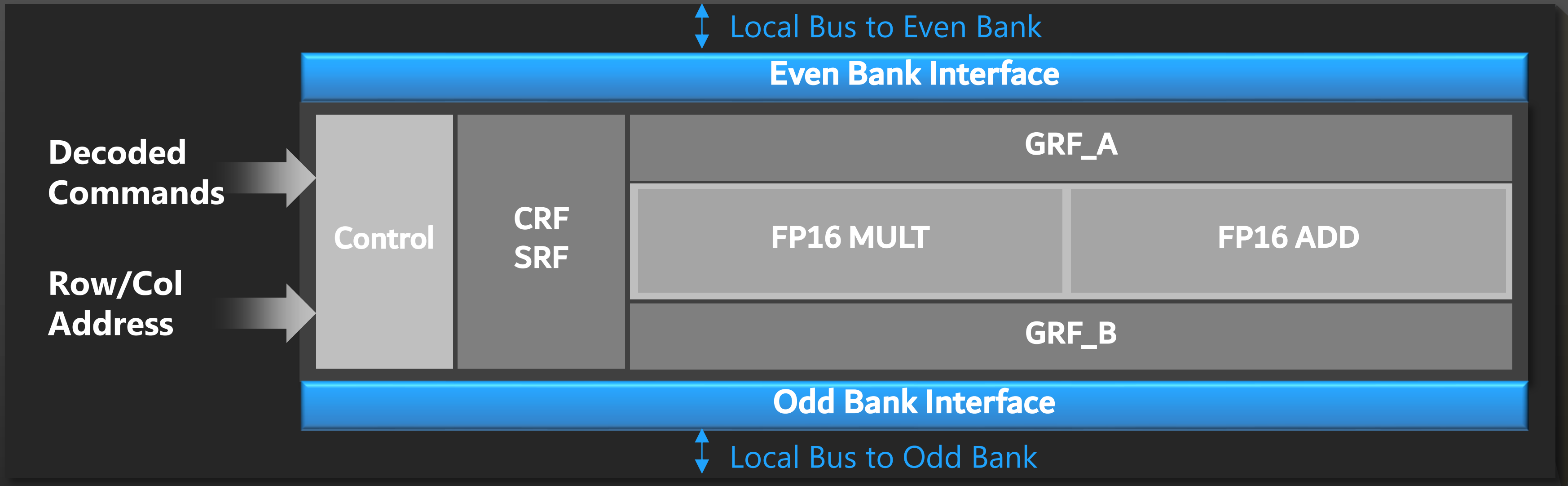
BANK BANK BANK BANK BANK BANK BANK BANK
PIM UNIT PIM UNIT PIM UNIT PIM UNIT PIM UNIT PIM UNIT PIM UNIT PIM UNIT
BANK BANK BANK BANK BANK BANK BANK BANK
TSVs & Periphery
BANK BANK BANK BANK BANK BANK BANK BANK

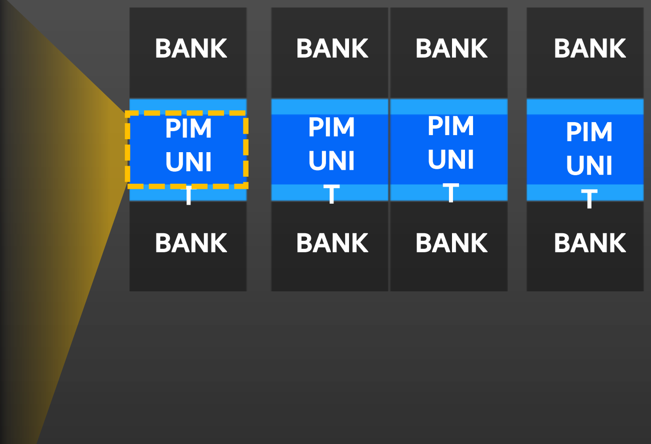
Bank



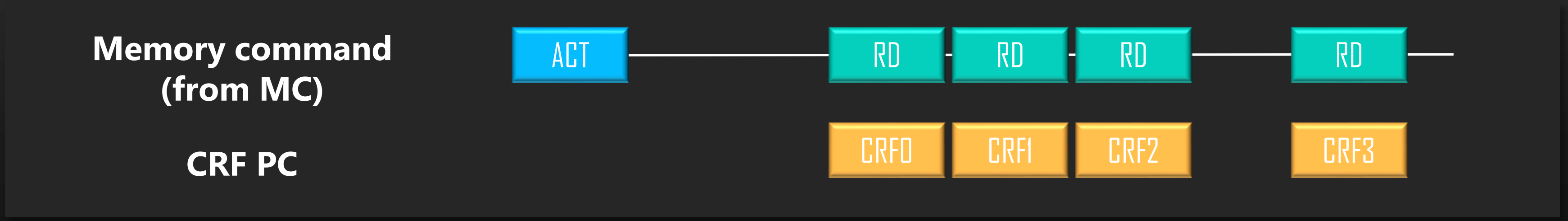
PIM Microarchitecture

- Consist of three major components with DRAM local bus interface:
  - A 16-lane FP16 SIMD FPU array: a pair of 16 FP16 multipliers and adders
  - Register files: Command, General, and Scalar register files (CRF, GRF, and SRF)
  - A PIM unit controller (fetch and decode, controls pipeline signals, forward)





External data (CAS) commands increase CRF PC and read (write) data from to column address at the same time. As a result, PIM units do not impact standard DRAM timing parameters.



PIM Instruction-Set Architecture

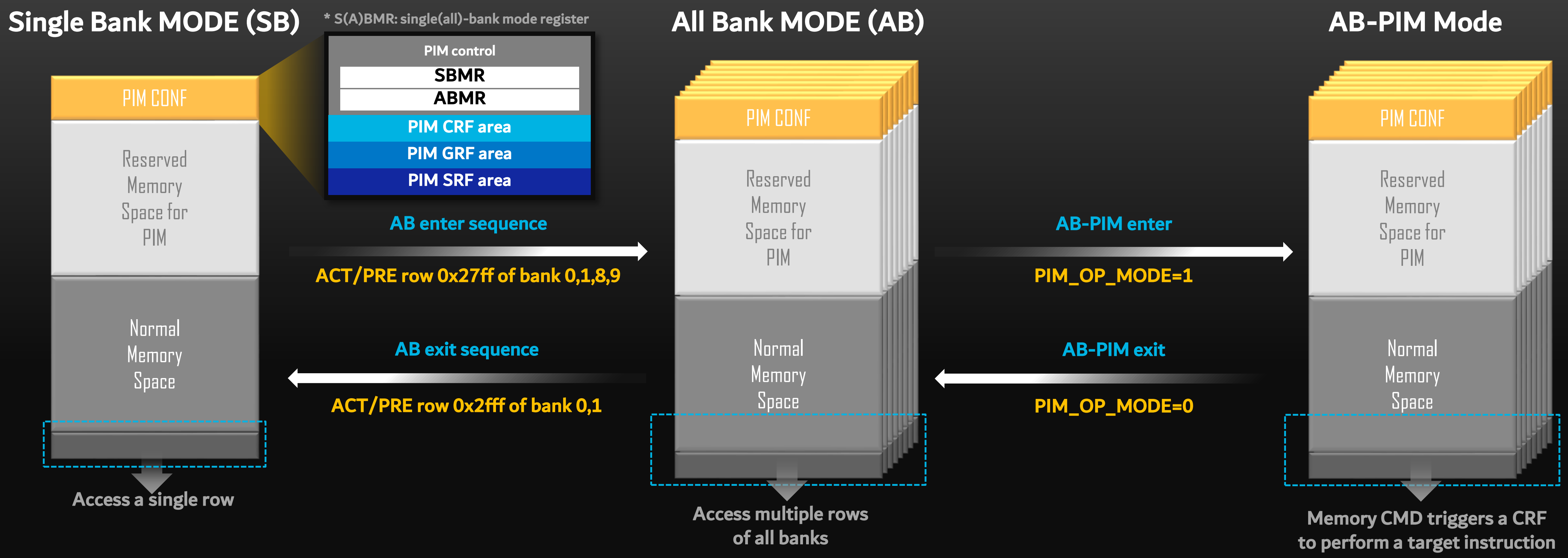
- Supports RISC-style 32-bit instructions.
- Three Instruction types (Total 9 instructions)
  - 3 Control flow: NOP, JUMP, EXIT
  - 2 Data transfer: MOV, FILL
  - 4 Arithmetic: ADD, MUL, MAC, MAD

- JUMP instruction:
  - Zero-cycle static branch: supports only a preprogrammed numbers of iterations
- Operand type: Vector Register(GRF_A, GRF_B), Scalar Register(SRF), and Bank Row Buffer
  - DRAM commands decide where to retrieve data from DRAM for PIM arithmetic operations

Type	Operation	Operand (SRC0)	Operand (SRC1)	Result (DST)	# of combinations
Arithmetic	ADD	GRF, BANK, SRF	GRF, BANK, SRF	GRF	40
	MUL	GRF, BANK	GRF, BANK, SRF	GRF	32
	MAC	GRF, BANK	GRF, BANK, SRF	GRF_B	14
	MAD	GRF, BANK	GRF, BANK, SRF	GRF	28
Data Transfer	MOVE	GRF, BANK		GRF, SRF	24
	FILL	GRF, BANK		GRF, BANK	12
Control Flow	NOP				1
	JUMP				1
	EXIT				1

PIM Operation mode

- Three execution mode: Single Bank (SB), All Bank (AB), and All Bank PIM (AB-PIM)
- Use sequence of standard JEDEC-standards compatible DRAM commands for mode transition
  - Using PIM configuration area

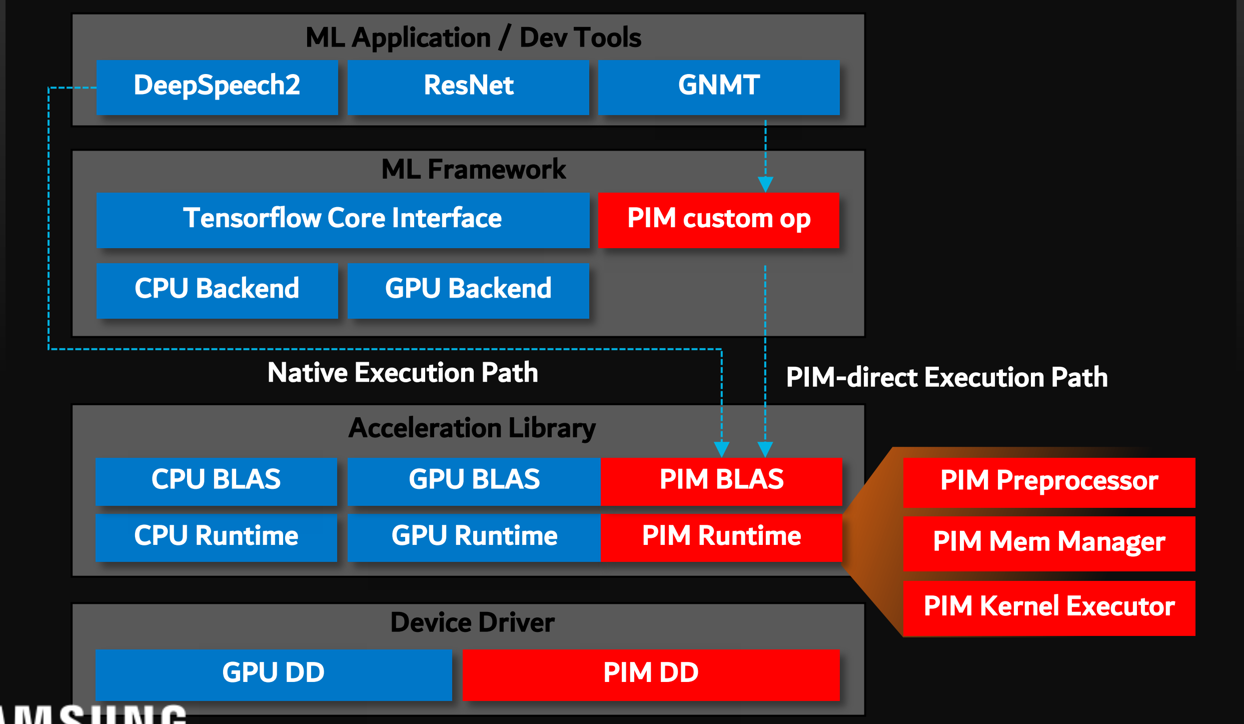


SAMSUNG

PIM Software Stack

- Developed PIM SW stack to allow users to run unmodified source code based on ML frameworks (TensorFlow and Pytorch)
- Supports two execution path
  - Native execution path: Automatically offload PIM-friendly TF operations, does not require any modification of application source code.
  - PIM-direct execution path: Explicitly call custom PIM-specific TF operations

PIM Software Stack for GPU

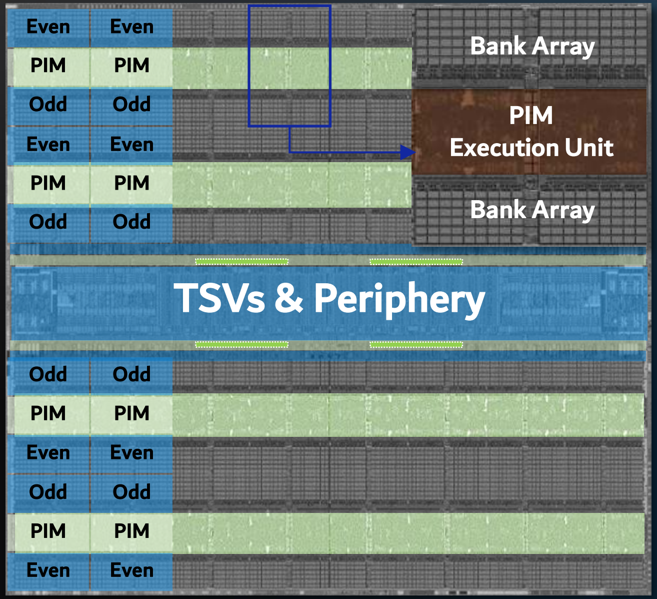


PIM-direct execution path

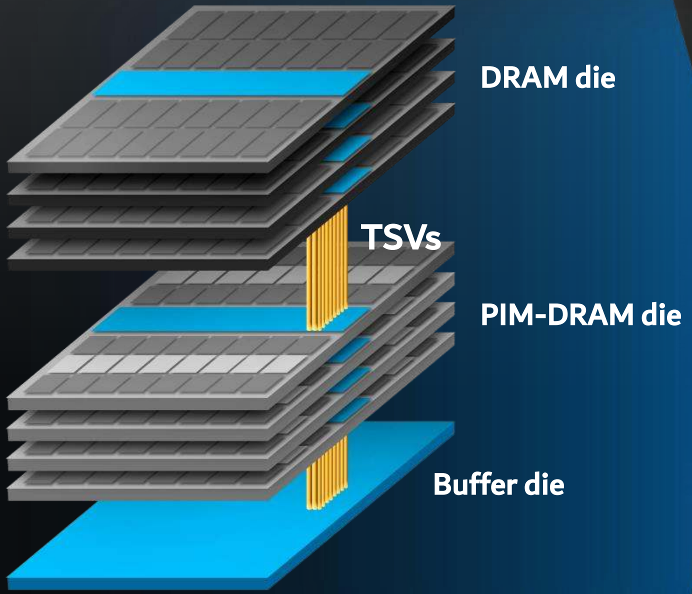
PIM TF custom op	tf_PIM_ops.python.ops.PIM_ gemv_ops (a, b)
PIM BLAS Lib.	PIMblasSgemv (int m, int n, const fp16 *A, const fp16 *B, ...)
GPU Kernel for PIM	PIM_gemv_kernel <numBlocks, threadsPerBlock>(...);
PIM micro-kernel	MAC(AAM) GRF_A, EVENBANK, GRF_A JUMP -1, 7 EXIT

Chip Implementation and Integration with systems

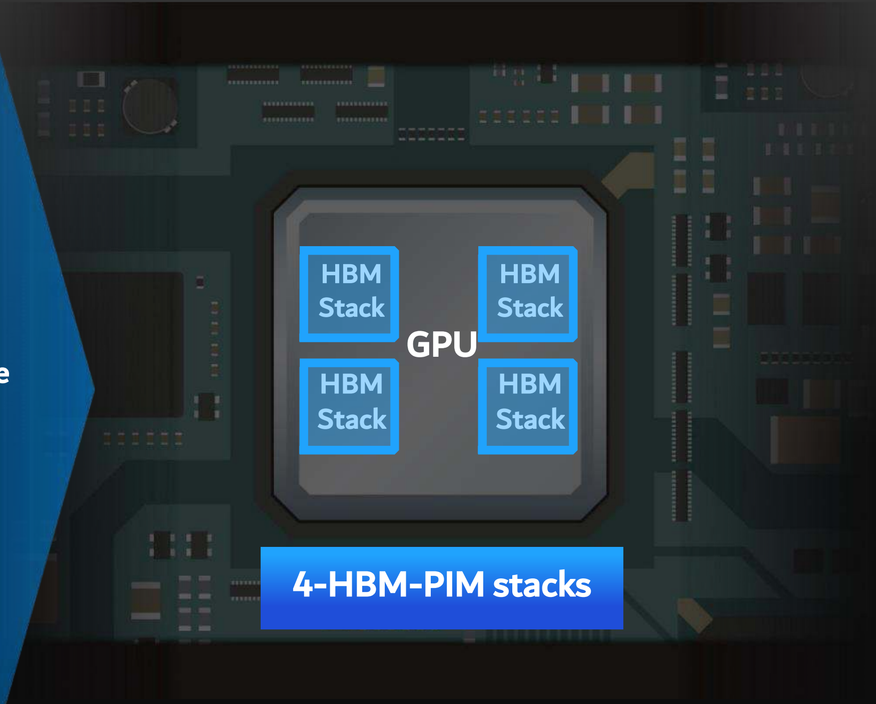
- Implemented PIM by modifying a commercial HBM2 design (Aquabolt). Resulting HBM-PIM device codenamed Aquabolt-XL
- Integrated the fabricated Aquabolt-XL with an unmodified GPU and Xilinx FPGA
  - Validated fabricated HBM-PIM in system with unmodified HBM controller
  - Off-chip and on-chip PIM compute bandwidth is 1.23 TB/s and 4.92 TB/s, respectively.



Aquabolt-XL silicon die photo



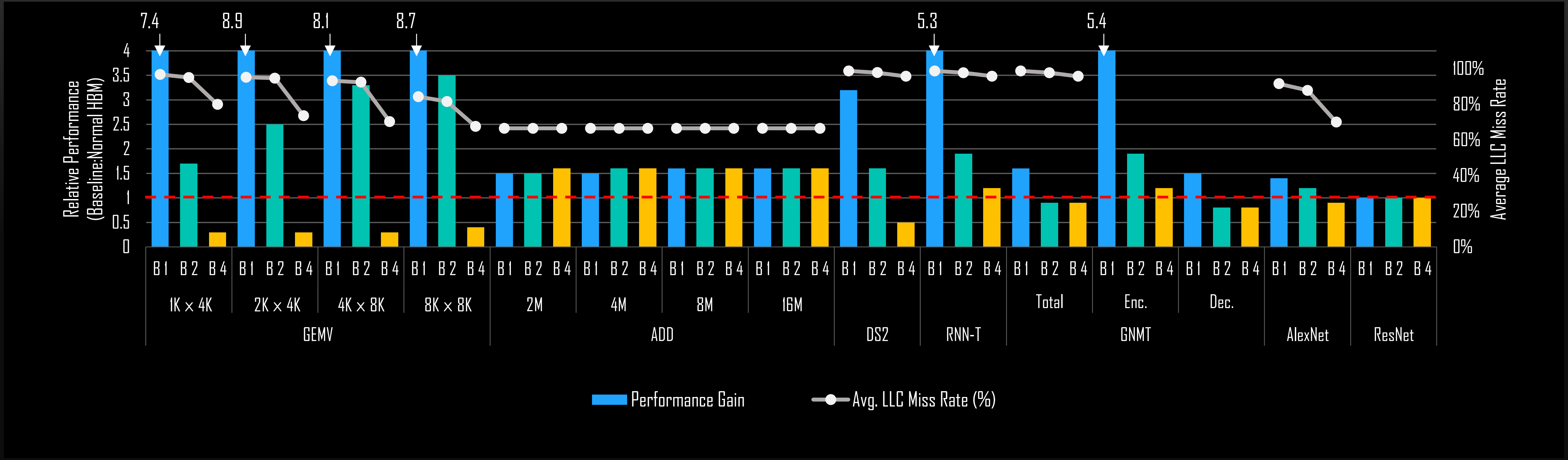
3D-stacked PIM-HBM



SAMSUNG

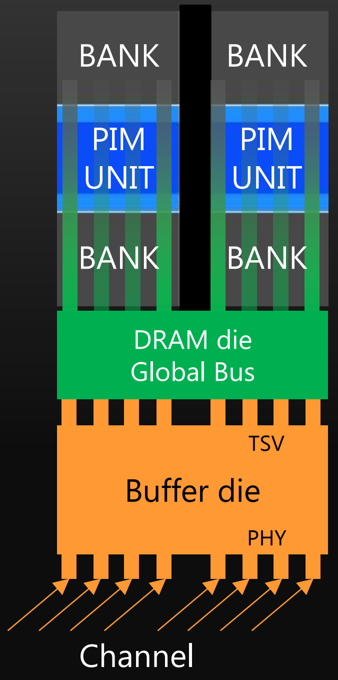
Evaluation – Performance

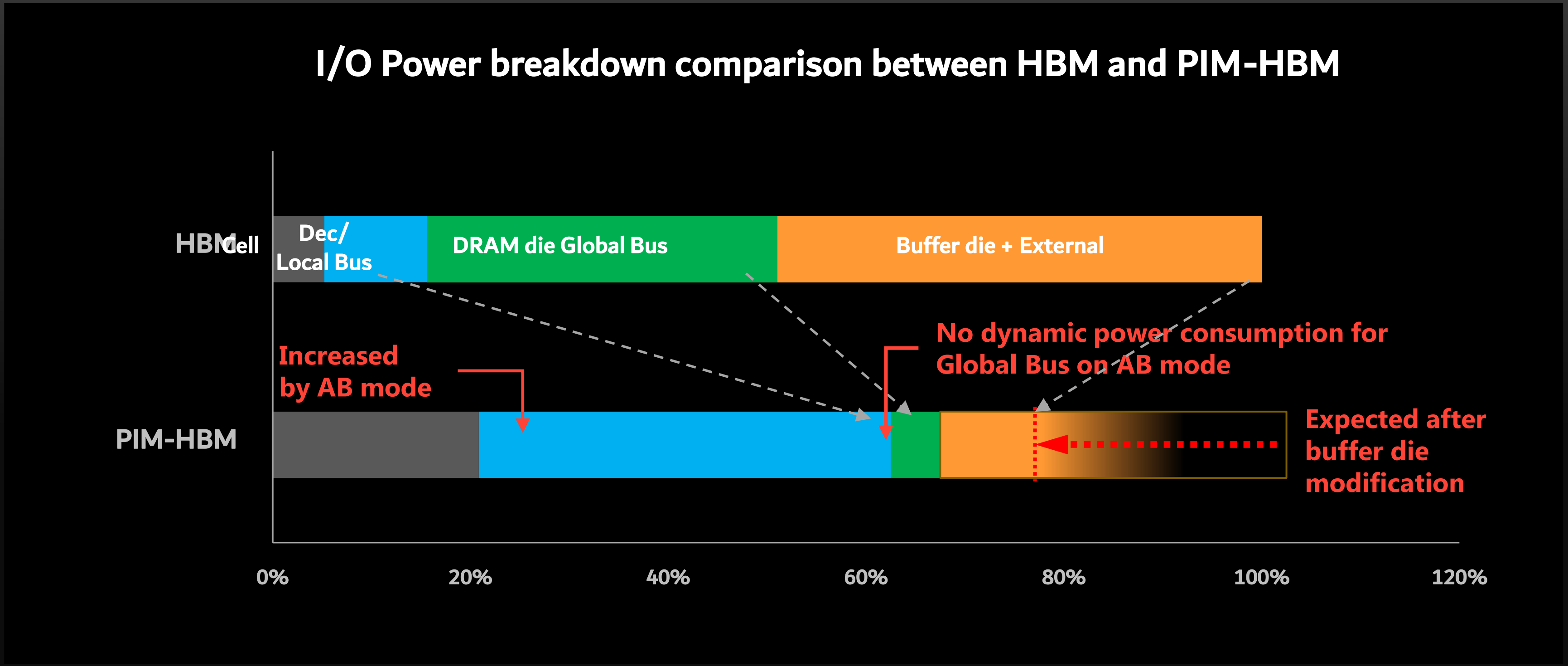
- Evaluated performance and energy efficiency of HBM-PIM-based system
- Performance Gain: 11.2× (Microkernel GEMV) and 3.5× (Speech Recognition Application)
- Energy Efficiency: Reduces the energy per bit transfer by 3.5× (overall energy efficiency of the system running the applications by 3.2×)
  - Micro benchmarks: vector-matrix multiplication (GEMV), element-wise addition (ADD)
  - End-to-end applications: 3 NLP apps (DS2, RNN-T, and GNMT), 2 CV apps (AlexNet and ResNet)



Evaluation – Power/Energy

- HBM-PIM consumes only 5.4% higher power compared to the HBM.
- In HBM-PIM, multiple PIM execution units operate concurrently
  - power consumption of DRAM internal components (gray and blue) increases proportionally, but
  - power consumption of internal global I/O bus (green) and I/O PHYs (orange) considerably decreases.



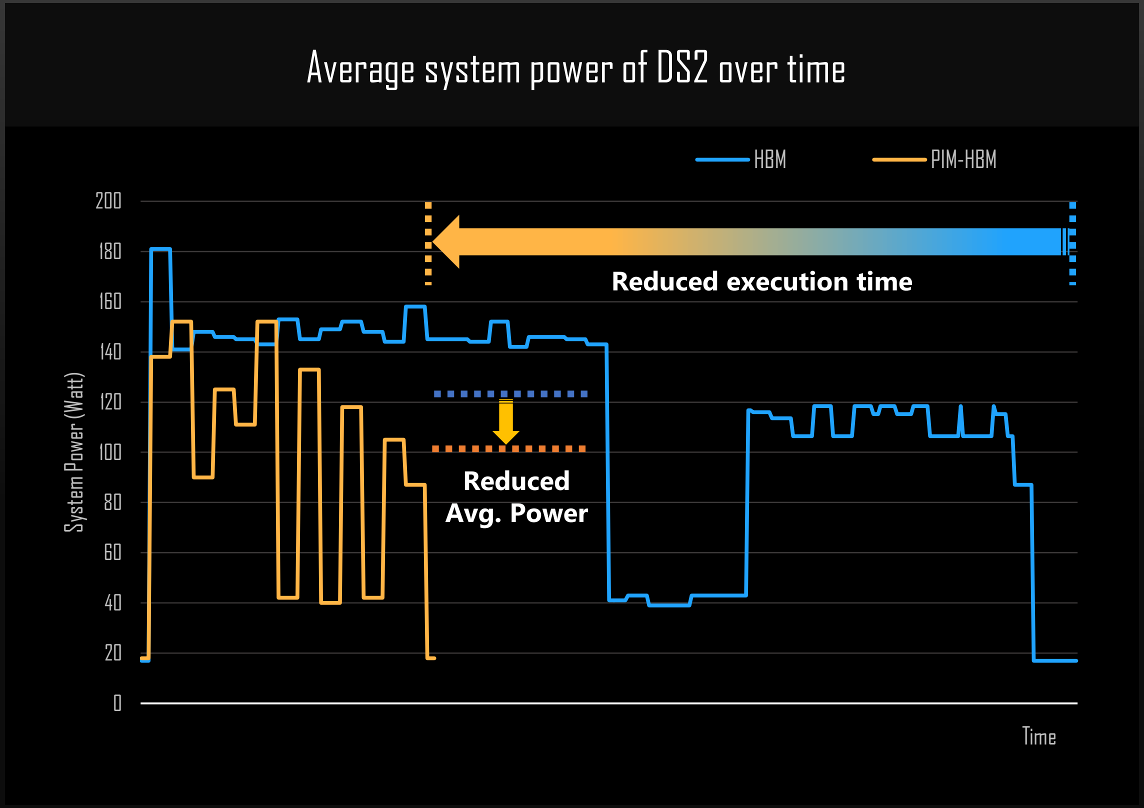


Evaluation – system power analysis, demonstration

- Evaluate the functionality and performance of real-life systems using a tensor flow implementation of Deepspeech2
  - Compare ASR trends, statistics of inference latency and word/character error rate
- HBM-PIM improves energy efficiency by shorter execution time and lower average power consumption

Aquabolt HBM2

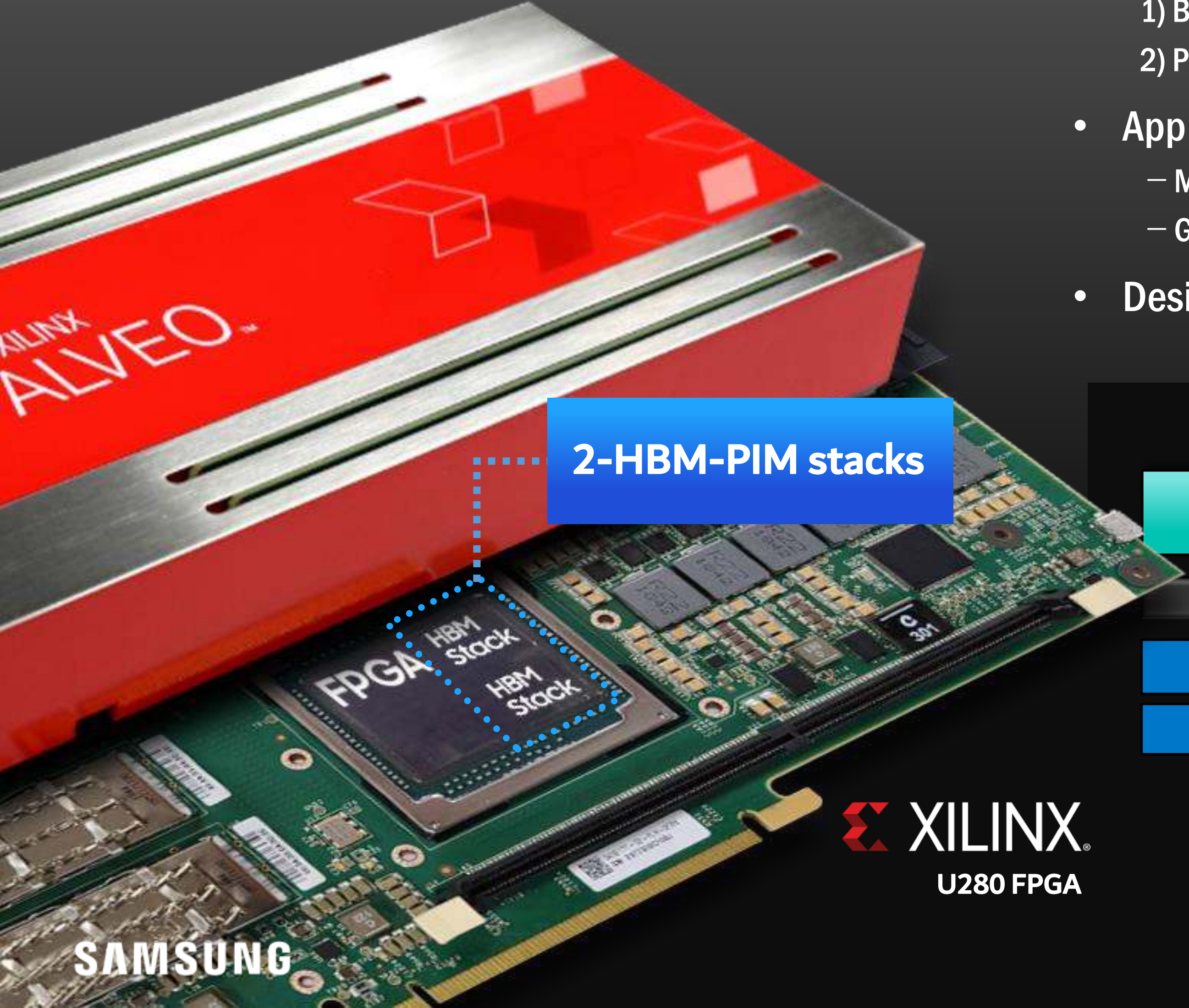
Aquabolt-XL HBM2-PIM



Average system power of DS2 over time

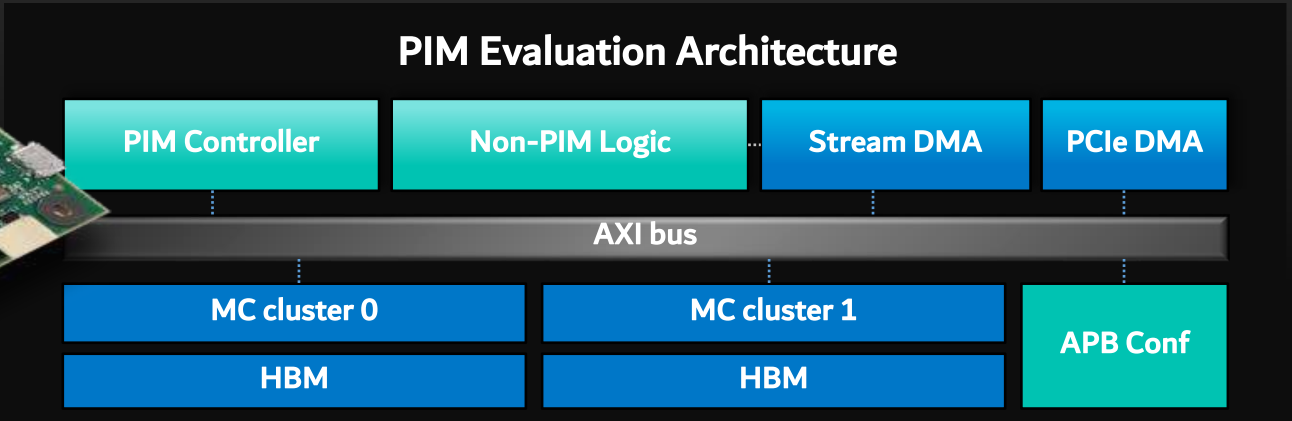
Xilinx Alveo U280: Evaluation setup

- For fair comparison, we created two separate FPGA projects:
  1) Baseline with default memory-controller setting
  2) PIM-accelerated logic with custom memory-controller setting
- Applied different memory controller configurations to each FPGA project
  - Memory controller configuration for PIM doesn't affect baseline performance
  - Guarantee the correct order of PIM-related memory commands and no auto pre-charge
- Designed baseline and PIM logic to utilize maximum HBM bandwidth



2-HBM-PIM stacks

XILINX
U280 FPGA

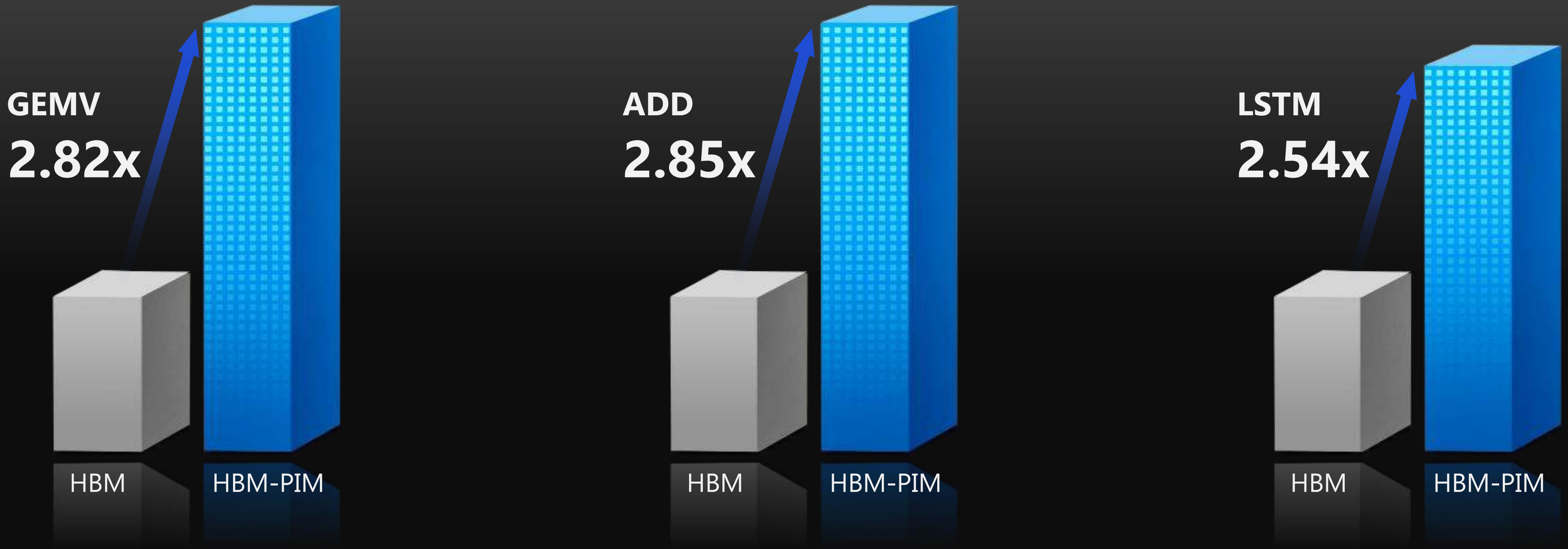


Xilinx Alveo U280: Evaluation Results

- Performance Gain: GEMV 2.82x, Add 2.85x, LSTM* 2.54x
  - GEMV: vector (1280) × matrix (1280 by 640) = vector (640)
  - ADD: vector (64K) + vector (64K) = vector (64K)
  - LSTM: input dimension (1024), hidden dimension (1024)
- Confirmed that our baseline/PIM-accelerated logics use maximum available bandwidth

* Long Short-term Memory Layer

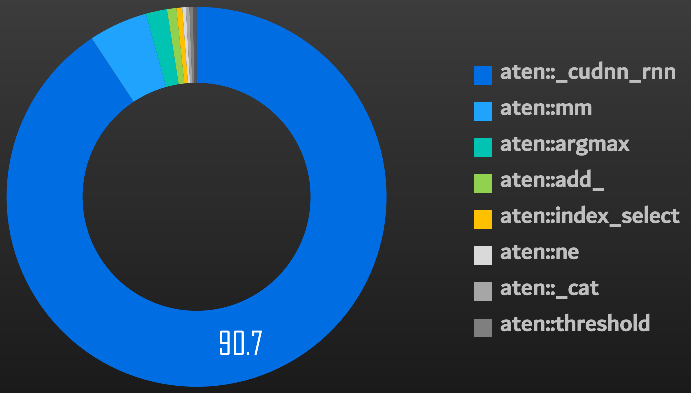
Relative Performance

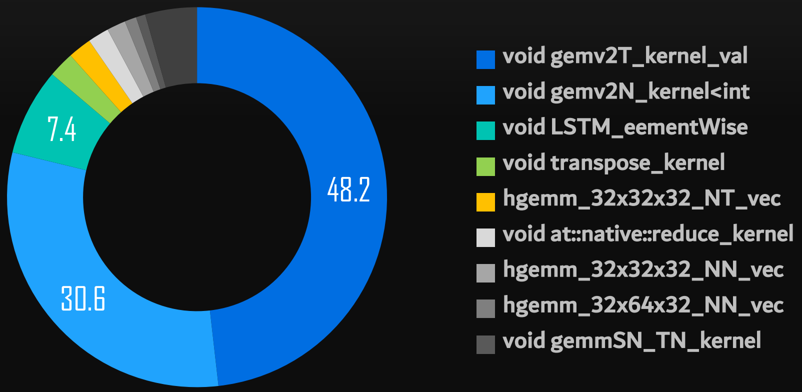


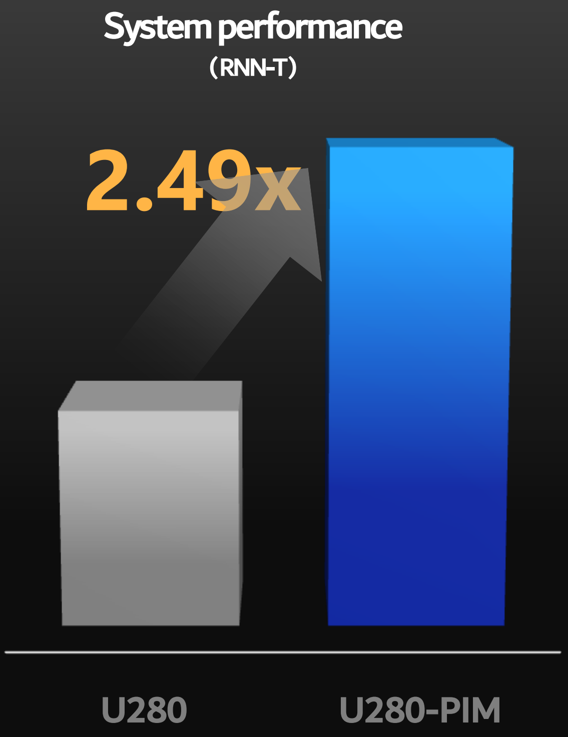
Xilinx Alveo U280: RNN-T Performance and Energy

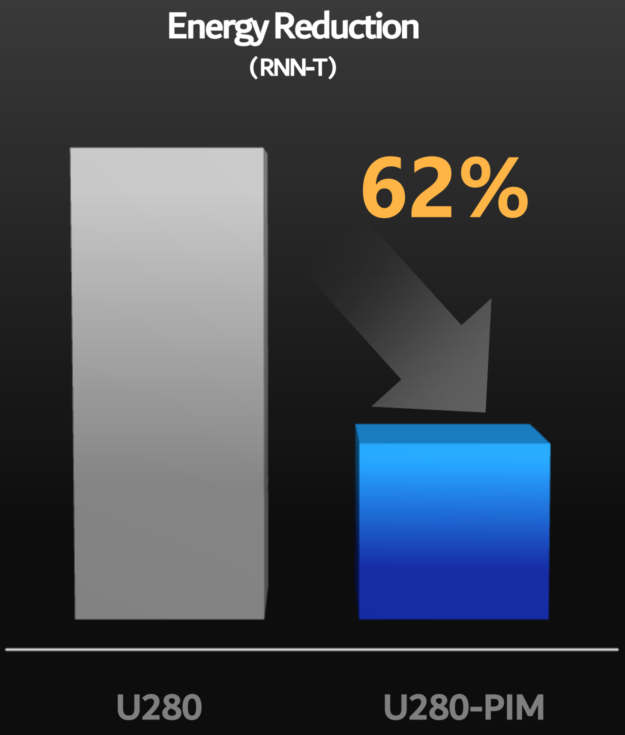
Recurrent Neural Network Transducer

- LSTM layer consumes 90.7% of the running time, while vector-matrix multiplication kernel uses 78.8%.
- PIM-enabled system reduces inference latency by 2.49X and the energy reduction by 62%



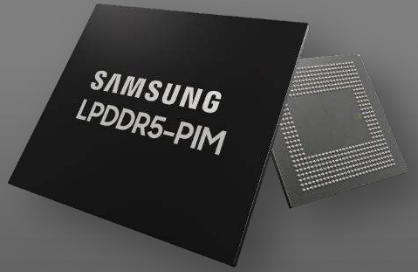




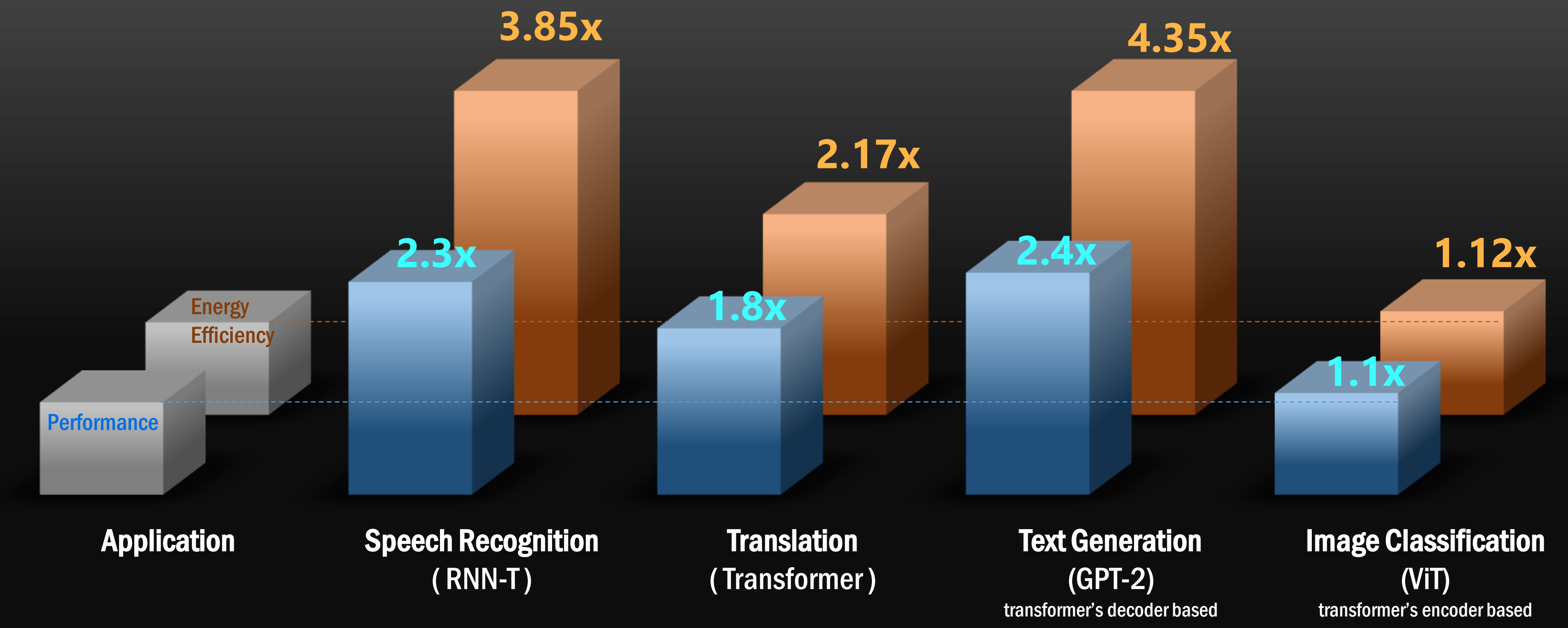


SAMSUNG

Expansion of PIM technology, LPDDR5-PIM



- Evaluated the performance and energy efficiency based on system-level simulation
  - Assumption: 8.2TFLOPS and 2.5TFLOPS/W NPU, 1MB SRAM, FP16, No zero-skipping, 1-channel 12.8GB/s LPDDR5X-6400
- Performance Gain: 2.3x (RNN-T speech recognition), 1.8x (transformer-based translation) and 2.4x (GPT-2 text generation)
- Energy Efficiency: 3.85x (RNN-T), 2.17x (transformer) and 4.35x (GPT-2)



SAMSUNG

Extensive evaluation for LPDDR5-PIM

- Evaluated the performance and energy efficiency based on layer-by-layer analytical model
  - Assumption: 61TOPS and 5TOPS/W NPU, LLC 8MB (4MB is used), INT8, No zero-skipping, Only 20GB/s DRAM BW allocated for NPU
- Performance Gain: 2.1x (transformer-based speech recognition) and 1.2x (video bokeh)
- Energy Efficiency: 2.86x (transformer-based speech recognition) and 1.30x (video bokeh)

Application		Model	Input	PIM Target Operation	Performance	Energy Efficiency
Natural Language Processing	Speech Recognition	Transformer	10.1sec	BLAS1,2	2.1x	2.86x
		Listen-attend-spell	10.1sec	BLAS2	3.5x	8.33x
	Question & Answer	BERT-Large	512 words	BLAS1	1.8x	1.75x
		ALBERT-Large	512 words	BLAS1	1.7x	1.75x
	Machine Translation	DeepSpeech2 + Transformer + Tacotron2	10sec	BLAS2	3.5x	7.69x
Computer Vision	Video Bokeh	DeepLab V3+ (ResNet-101)	512x512	BLAS1	1.2x	1.30x
		DeepLab V3 (ResNet-50)	FHD	BLAS1	1.1x	1.02x
			4K	BLAS1, 1x1 Conv	1.2x	1.08x
			8K	BLAS1, 1x1 Conv	1.3x	1.16x
	Depth Estimate	DORN	4K	BLAS1	1.1x	1.10x
			8K	BLAS1	1.2x	1.11x
	Image2Image translation	StarGAN	8K	BLAS1	1.1x	1.02x

Frequently Asked Questions

Do application programmers need to know how PIM works?

No. Application programmers are able to run existing ML applications on a system using HBM-PIM without any changes to the source code, by using PIM-aware software stack. PIM software stack can detect PIM-beneficial operations and offload them to HBM-PIM without programmer awareness.

How can PIM guarantee JEDEC standard compatibility?

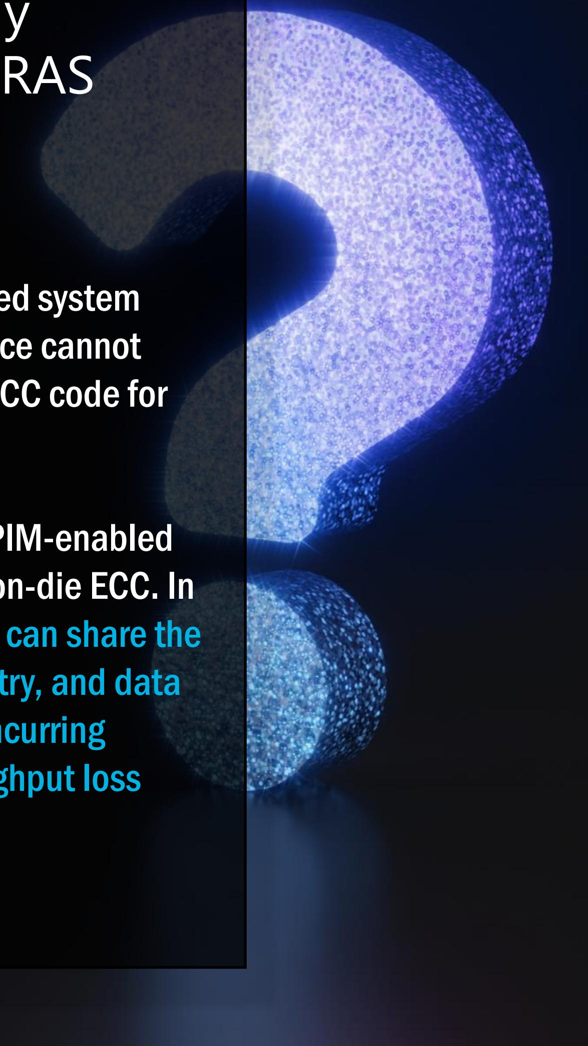
Executions of PIM instructions with standard DRAM commands and deterministic latencies are essential to facilitate HBM-PIM with unmodified JEDEC-standard compliant DRAM controllers

During AB(or AB-PIM) mode, the host processor can control execution of every PIM instruction one by one with its load (LD) and store (ST) instructions which are translated into standard DRAM commands to DRAM

Is PIM technology compatible with RAS features?

For Aquabolt-XL, we disabled system ECC because the HBM device cannot generate system-specific ECC code for PIM-generated data

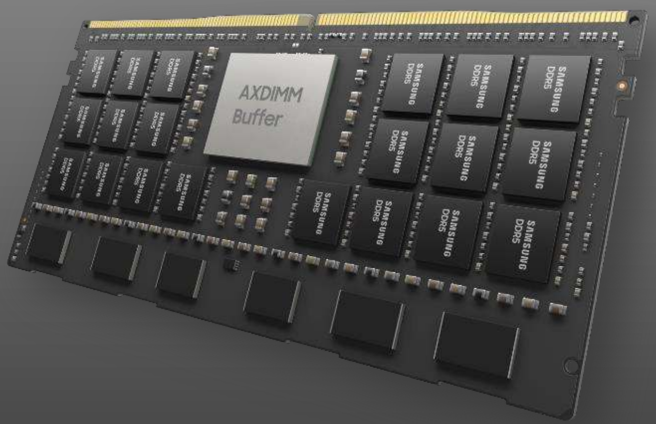
For the next generation of PIM-enabled HBM, we expect to deploy on-die ECC. In this architecture, PIM logic can share the ECC encode/decode circuitry, and data can be protected without incurring additional latency or throughput loss

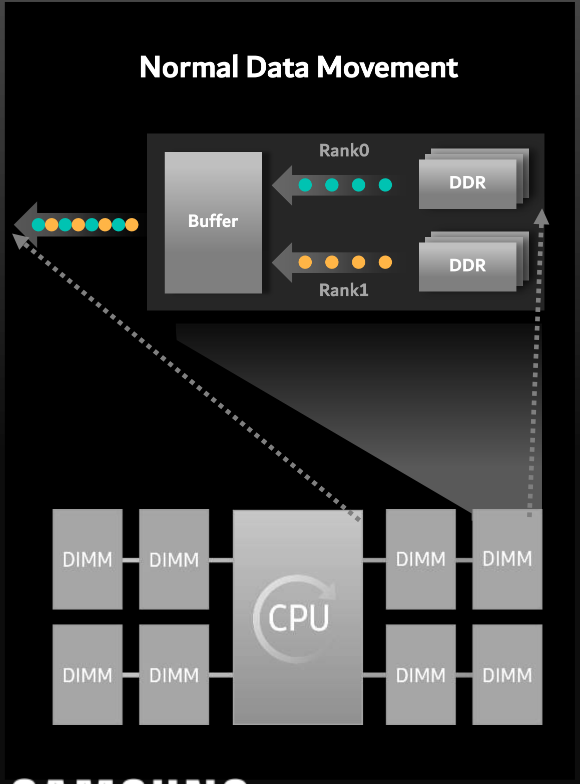


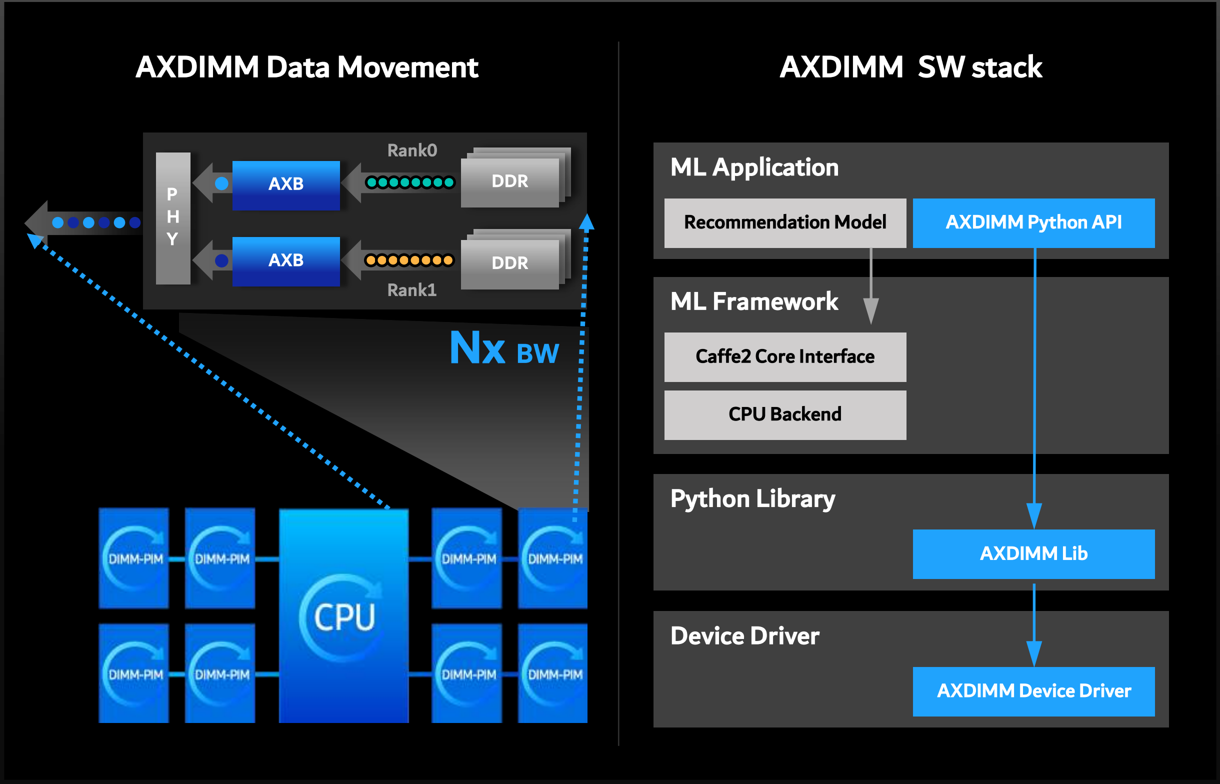
AXDIMM, DIMM-PIM Concept

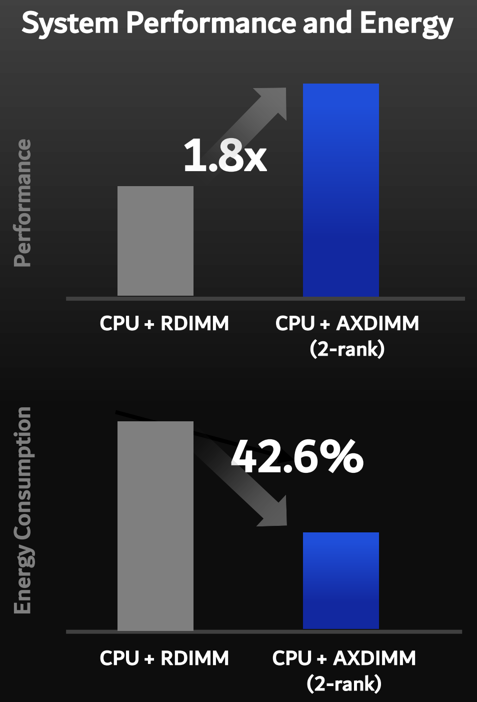
AXDIMM: Acceleration DIMM

- CPU-memory data movement bottlenecks system performance → rank-level parallelism is needed
  - Samsung to provide AXDIMM SW stack to offload the acceleration functions in AXB(AXDIMM Buffer)
- Improve the performance and energy efficiency of the system with in-DIMM processing
  - Utilize up to higher in-DIMM bandwidth by multi-Rank parallel operation, 1.8x by 2-rank
  - Reduce data movement energy by utilizing in-DIMM data processing unit, -42.6% by 2-rank









SAMSUNG

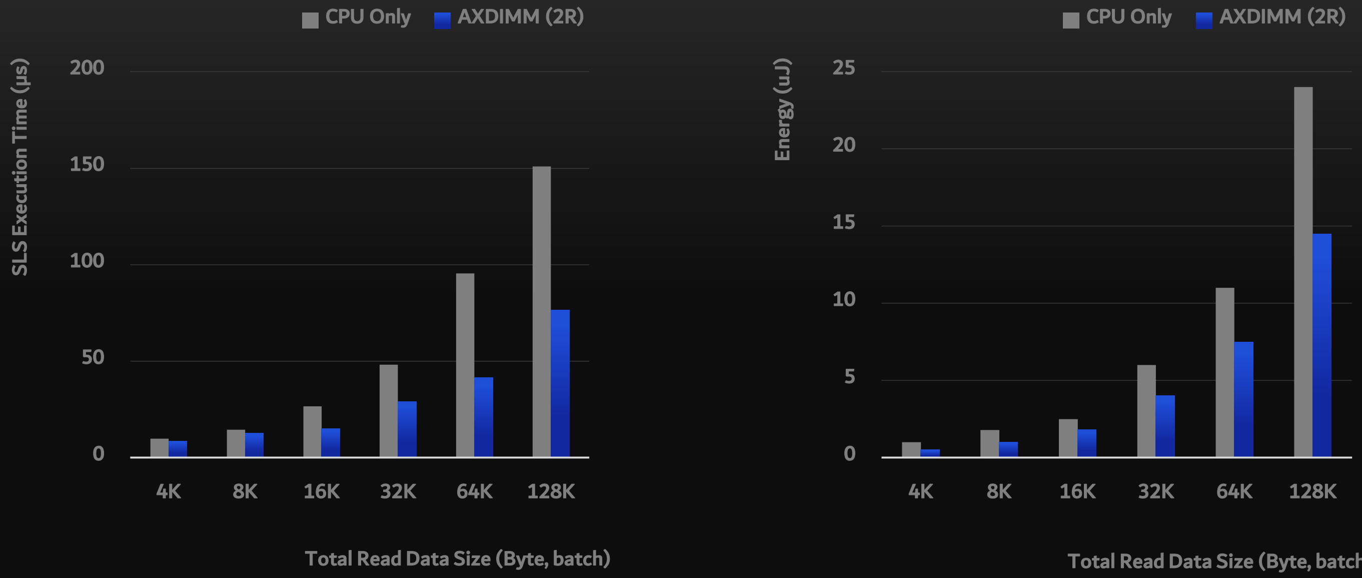
AXDIMM evaluation system and results

- Setup x86 based AXDIMM platform with Xilinx Zynq Ultrascale+ FPGA Chip
- Enabled RecNMP* logic
  - Achieved 1.8x SLS execution speed-up from HW
- Modified DLRM** application utilizing AXDIMM
  - Achieved 1.8x/3.5x/6.9x QPS

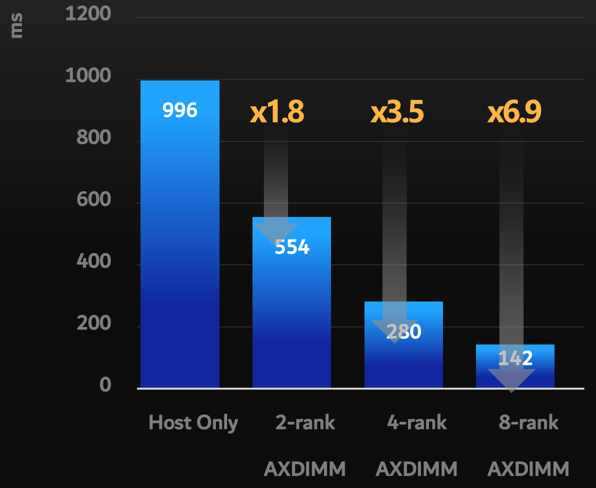
4-AXDIMM per CPU

Intel Broadwell Server

Performance and Energy gain by data size



Total Execution time of DLRM for 700 Requests



*: RecNMP - L. Ke *et al.*, "RecNMP: Accelerating Personalized Recommendation with Near-Memory Processing," *2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA)*
**: DLRM - M. Naumov et al, “Deep learning recommendation model for personalization and recommendation systems,” arXiv preprint, 2019

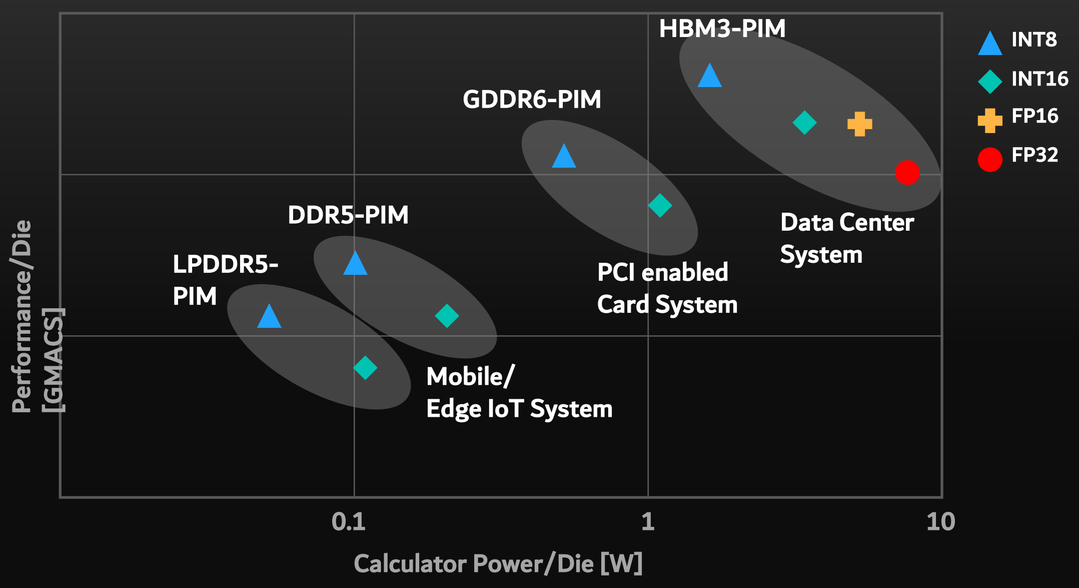
SAMSUNG

Future proposal

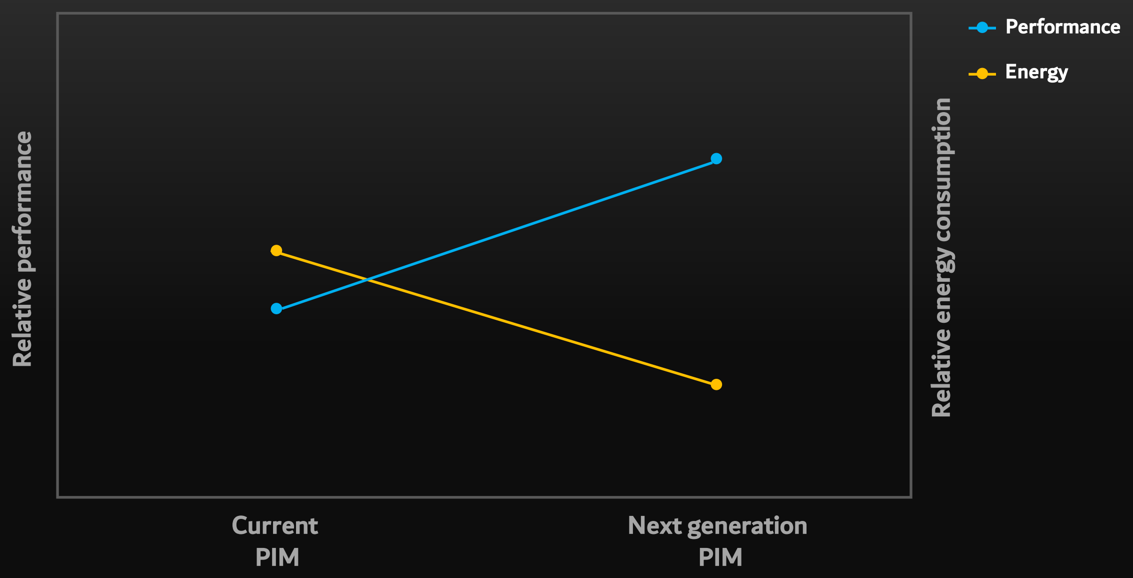
- Wider target applications
  - PIM unit supporting multiple functions
- Various DRAM types
  - LPDDR5, DIMM-DDR5, GDDR6, HBM3
- New standards for PIM
  - Command truth table/timing for PIM

- Addendum or addition to current product specs, not new generations
  - Enhanced performance
  - Reduced energy
- Collaborate with industry
  - Supporting custom functions

Prospective PIM-supported data format



Performance/Energy of next generation PIM



SAMSUNG

Thank you