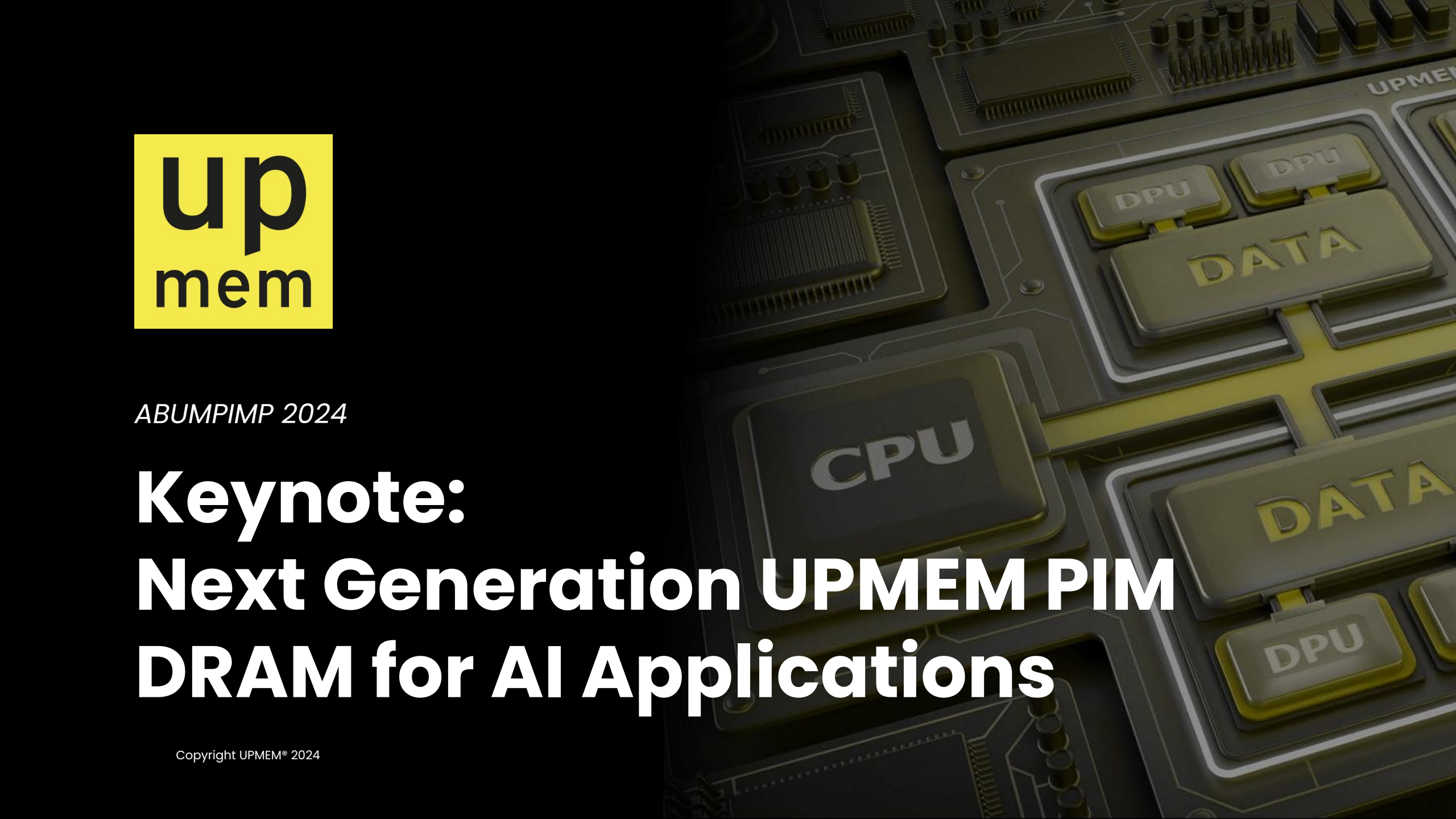


ABUMPIMP 2024

Keynote: Next Generation UPMEM PIM DRAM for AI Applications

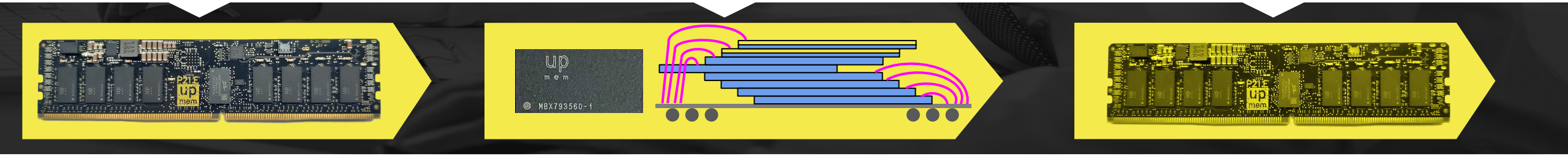
Hardware Architecture

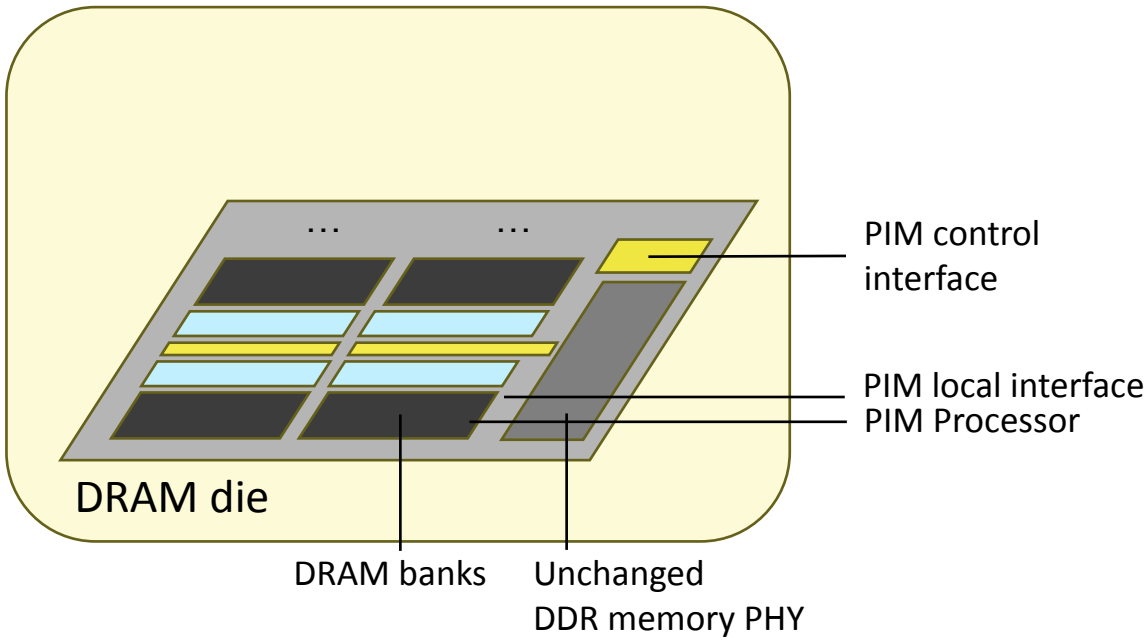
PIM-AI chip architecture overview

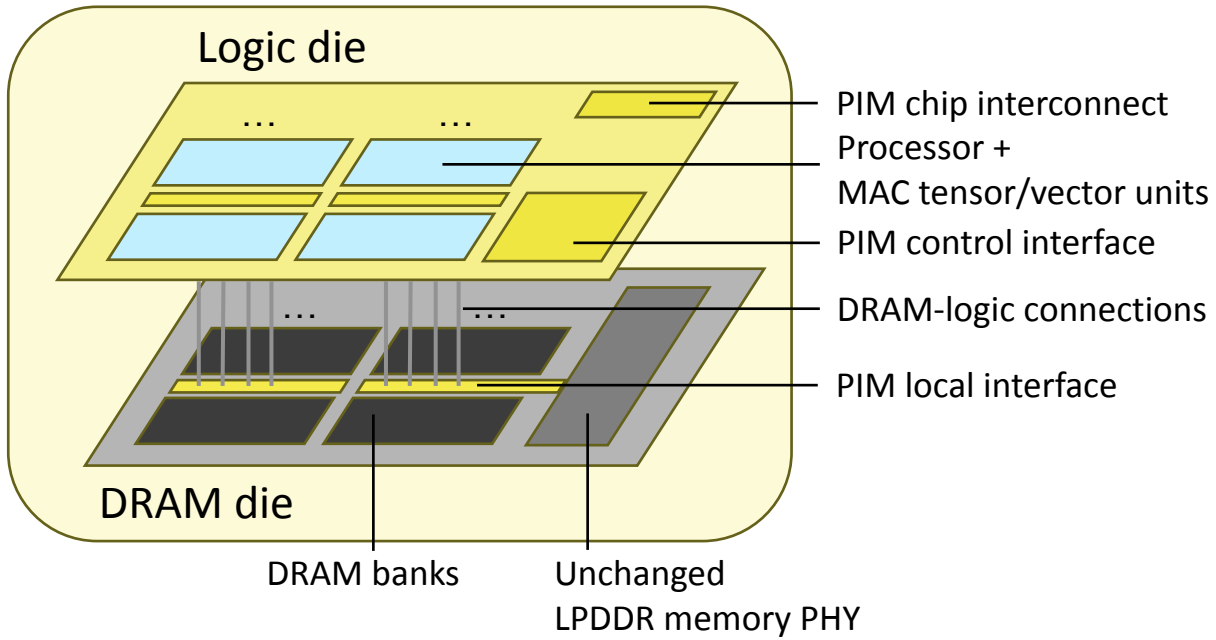
Existing PIM DRR architecture
Based on DDR4 DRAM design

Target PIM AI LLM chip
Based on LPDDR5 DRAM design

Target PIM AI LLM DIMM
Based on DDR5 / LPDDR5 DRAM design



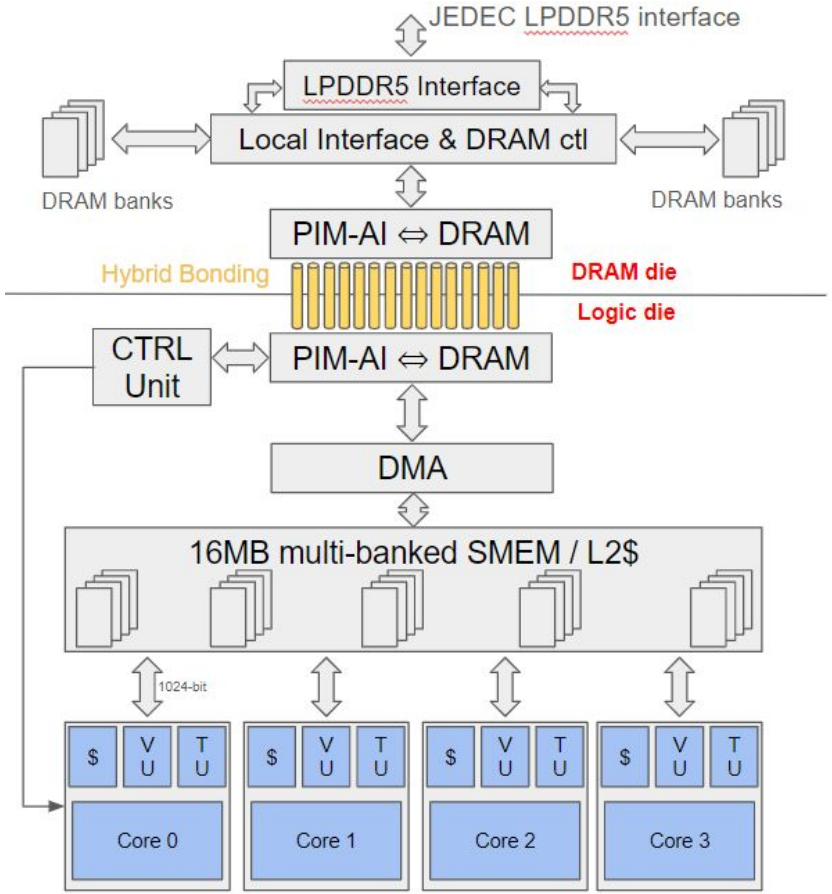




UPMEM PIM LLM structural benefits for beating SOCs with NPUs

UPMEM PIM AI chip specs for smartphones

- PIM orchestration IP
- AI compute fabric with RISCV core, Tensor unit (5/8 TFLOPS), Vector unit....
- 2GB DRAM LPDDR5 / DDR5

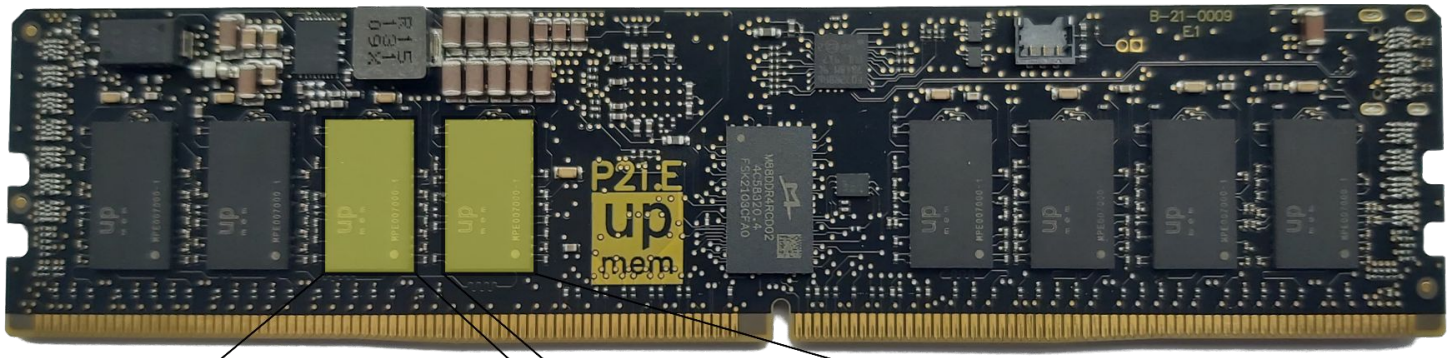


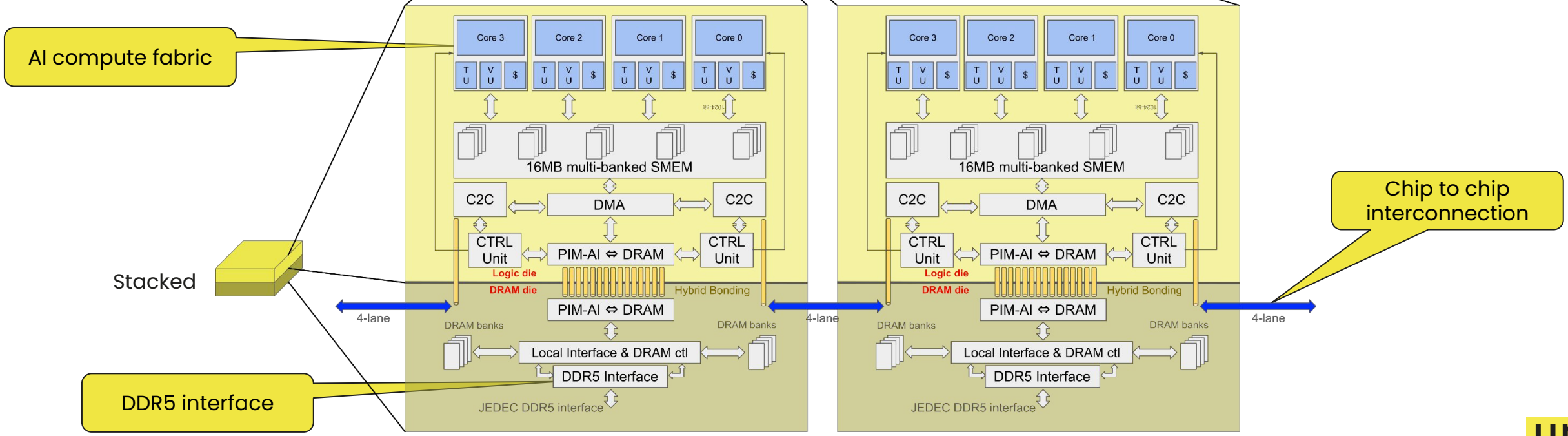
- **PIM AI architecture benefits vs. current SOC**
  - **Much higher DRAM bandwidth** for memory bound LLMs
    - 100GB/s per 2GB DRAM
      - Several x more than when accessing through LPDDR Memory controller
  - **Much lower energy cost per bit** on most of data transfers occurring during generation
    - 1pJ/bit
  - Much better performance, energy efficiency & TCO
  - UPMEM PIM chip: **2GB 5 TFLOPS FP16**
    - Can be associated with several PIM chips
    - Allows standard DRAM mode or PIM-DRAM mode
- While requiring **no change in the SOC**

- **Making UPMEM PIM the enabler of GenAI (LLMs) on smartphones**

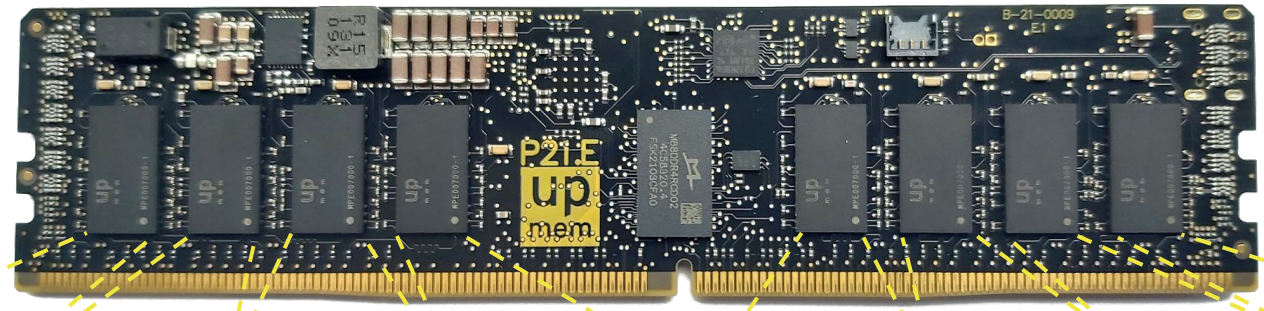
upmem

PIM-AI chip architecture overview (for cloud)



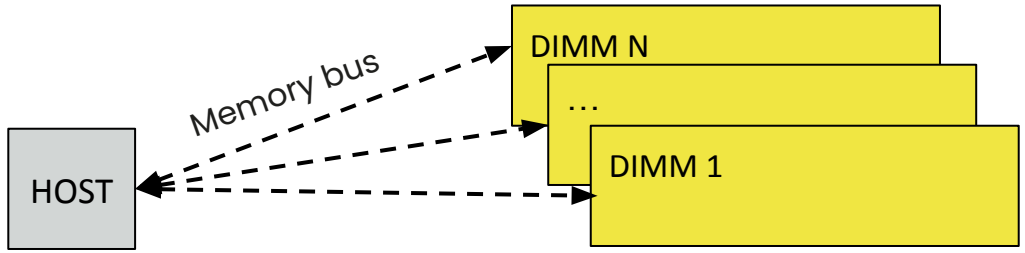


PIM-AI chip architecture overview (for cloud) II



INPUT LLM

- INPUT of LLM is used by all chips when applying tensor parallelism
- Chip interconnect allow faster communication HOST <-> DIMM
- Operations using a single DIMM do not need to synchronize with HOST



Synchronization between PIM-AI DIMMs and multiple PIM-AI chips are required to go through HOST

Benchmark methodology

LLM models decoded

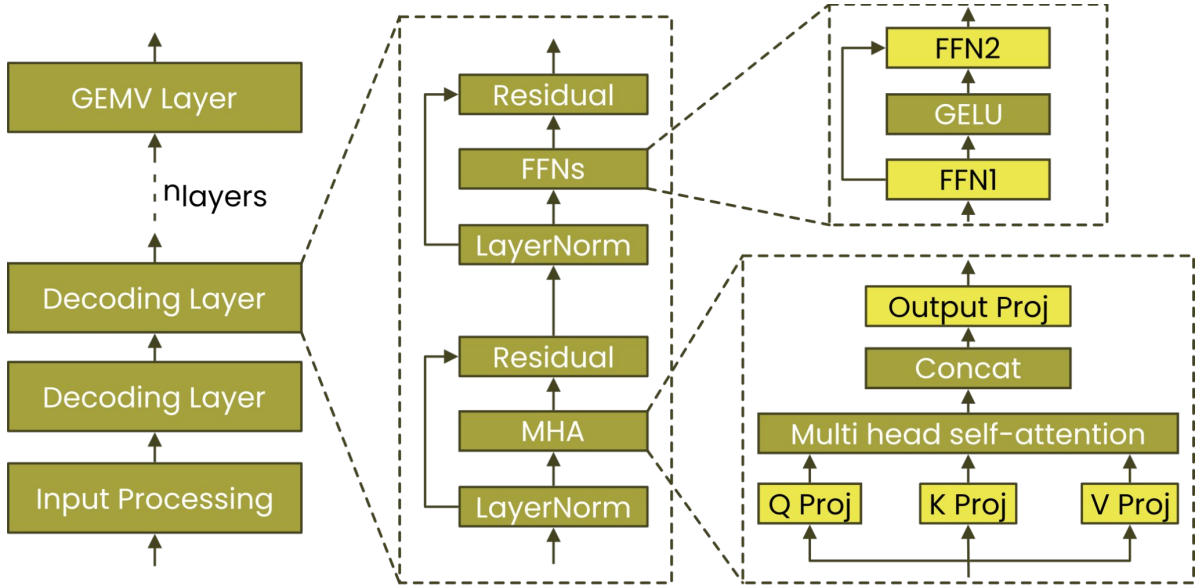
LLM execution mainly consist in two steps
(assuming KV cache)

- **ENCODING**
  - Done a **single time**
  - Input is issued from the prompt
    - Typically a few hundred rows : **matrix**
  - The memory bandwidth is not that much critical
  - Compute performance matters
    - Llama-2-7B model encoding time
      - For 64 tokens

        `8.290s @ 102.4 GFLOPs`
        `0.210s @   5.0 TFLOPs`

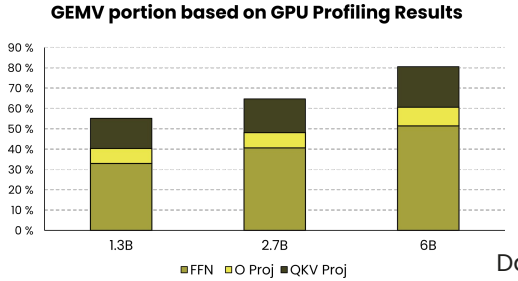
- **DECODING**
  - Done **many times**
    - One time for each new generated token
  - Input is a single row : **vector**
  - The memory bandwidth is critical

    tokens/s ~ memory_bandwidth / model_size
  - Compute performance does not matter



Layers consisting of GEMM during the encoding phase and GEMV during the decoding phase

Introducing Real-world HBM-PIM Powered System for Memory-bound Applications - Samsung Electronics DRAM Design Team



As the model size increases, the linear layer O(H2) overwhelms attention layer O(HL)
*H is hidden dimension, L is sequence length

	Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}
	GPT-3 Small	125M	12	768	12	64
	GPT-3 Medium	350M	24	1024	16	64
	GPT-3 Large	760M	24	1536	16	96
Mobile	GPT-3 XL	1.3B	24	2048	24	128
Mobile	GPT-3 2.7B	2.7B	32	2560	32	80
Mobile	GPT-3 6.7B	6.7B	32	4096	32	128
Datacenter	GPT-3 13B	13.0B	40	5140	40	128
Datacenter	GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128

Sizes, architectures and parameters of the GPT-3 models

UPMEM LLM exploration tools and methodology

- **UPMEM LLM simulator description**
  - Standard Pytorch framework running on x86
  - Standard LLM models (sourced from Hugging Face)
  - Supporting any accelerator profile with key parameters description (bandwidths, energy, …)
  - Providing key performance and profiling metrics

- **Llama and Mistral models simulation and profiling information**
  - Multiple targets (UPMEM, Apple, Mediatek, Qualcomm, NVIDIA)
  - ENCODING and DECODING information split
  - Variable input length and number of generated tokens
  - FP16, FP8, INT8, or INT4 operands

- **Hardware metrics confirmation**
  - Cycle accurate simulations on High End RISC-V multicore IP with vector unit
  - Several functions exercised (GEMM, GEMV, softmax, …)

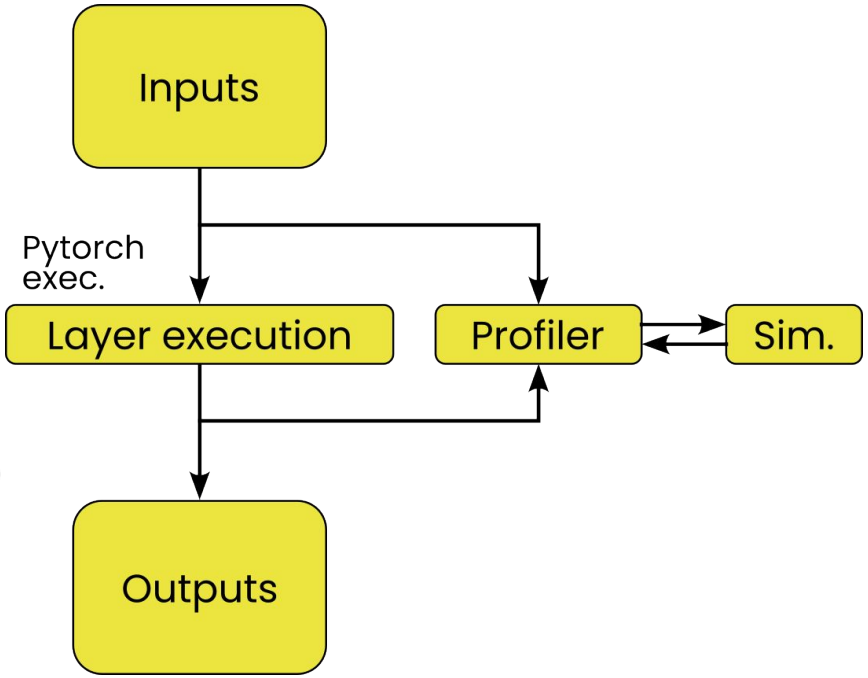
UPMEM LLM hardware simulation tools

UPMEM LLM x86 simulator allows to profile execution on different accelerators targets

- The simulator is fed with the **accelerator description**
  - Bandwidths
    - Host to device (H2D)
    - Device to host (D2H)
    - Main memory to AI logic
  - Compute performance
  - Energy for each of these metrics

- The simulator provides simulated metrics to the profiler
  - Sizes of the data for each layer
  - Dataflows (host to device, device to host, internal main memory)

- At the end of the execution, the profiler collects profiling data
  - **Time and energy** for each layer
  - **ENCODING performance**
  - **DECODING performance**



Evaluation

Mobile accelerator descriptions

Accelerators	Host ⇔ Device			Main Memory		Compute		Notes
	H2D GB/S	D2H GB/s	pJ/bit	BW GB/s	pJ/bit	TFLOPS	pJ/flop	
Apple A17 pro	51.2	51.2	20	51.2	**20**	4.3	0.4	DECODING energy 10X higher than on-chip DRAM Pushing adoption of low precision datatypes (int4) **Qualcomm** : *"Low-bit integer precision is essential for power-efficient inference."* LPDDR interface will limit AI effective bandwidth Shared by all the AP processes Figures per system
Qualcomm Snapdragon 8 GEN3	77	77	10	77	**10**	4.73	0.4	
Mediatek Dimensity 9300	77	77	10	77	**10**	6	0.4	
Samsung LPDDR5 PIM	12.8	12.8	20	102.4	0.95	**0.1024**	**0.8**	On chip standalone processing is **IMPOSSIBLE** Not general purpose processing Poor algorithms flexibility & datatypes support Very heavy scheduling from host, and energy waste **DECODING only** Multiple chips can be grouped together to increase BW/performance
upmem PIM-AI (1 chip)	12.8	12.8	20	**102.4**	**0.95**	5	0.4	Multiple chips can be grouped together to increase BW/performance

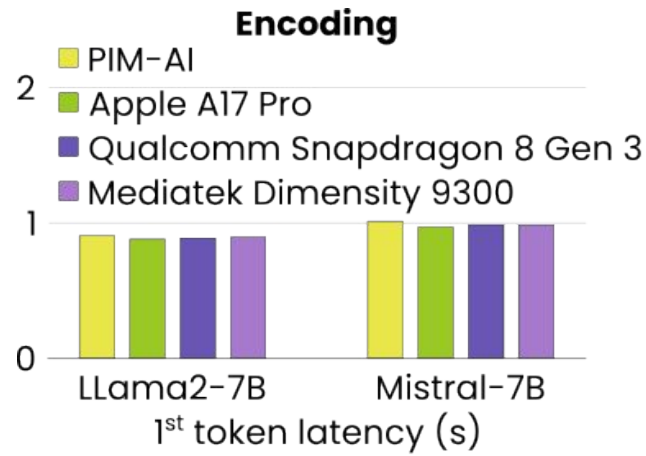
Mobile simulations

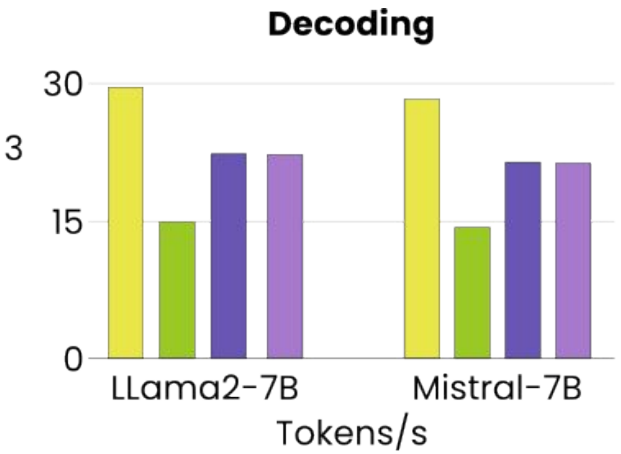
Inference of 1000 tokens in / 100 tokens out

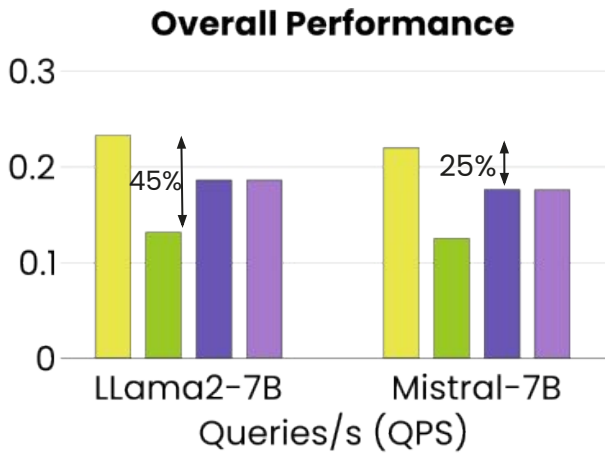
Llama-2-7B	
vocab_size	32000
embedding_size	4096
n_parameters	6.7B
n_context	2048
num. heads	32

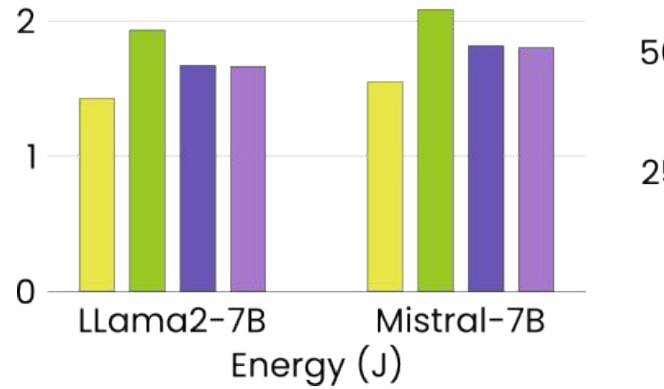
Mistral-7B	
vocab_size	32000
embedding_size	4096
n_parameters	7.2B
num. heads	32
num. kv heads	8

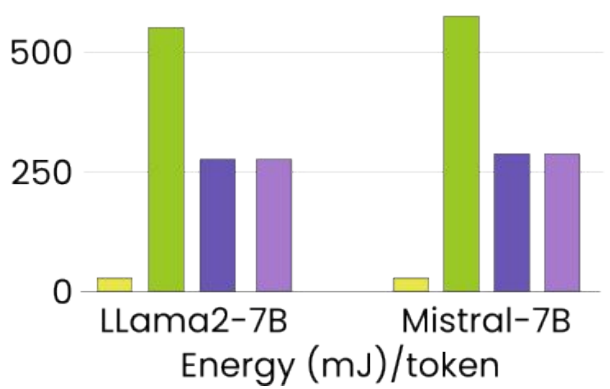
batch size	1
Weights data_type	4-bit
GEMMs data_type	8-bit
Activations data_type	16-bit

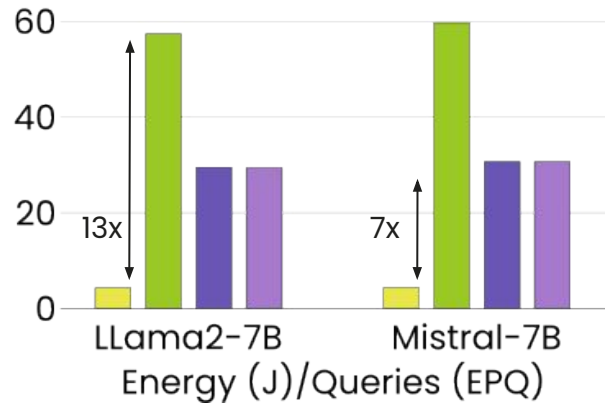












PIM-AI: 10x more requests on a smartphone battery on average (only 1000 requests for SoC)

1 joule = 0.28mWh - smartphone battery ~13Wh / 46 800J

Cloud accelerator descriptions

- DGX-H100 server is 8U:
  - 8xH100 GPUs
  - **640 GB of HBM**
- PIM-AI server is 2U with:
  - 24 PIM-AI DIMMs, each DIMM with:
    - 16 PIM-AI chips with 8 TFLOPS
    - **768 GB of PIM-AI**
  - 8 legacy DIMMs
- Next comparisons are between 1 DGX-H100 server and 4x PIM-AI server (same rack occupancy)

Accelerators	**Host ⇔ Device**			**Main Memory**		**Compute**	
	H2D GB/S	D2H GB/s	pJ/bit	BW TB/s	pJ/bit	TFLOPS	pJ/flop
NVIDIA DGX-H100 (8xH100)	450	450	280/40	**26.8**	**7**	7916	0.5
upmem PIM-AI (1 server)	22	528	1920/50	**39.3**	**0.95**	3072	0.5

Includes interconnect communication between GPUs and DIMMs when broadcasting input (modelling 8 NVIDIA switches)

Cloud simulations

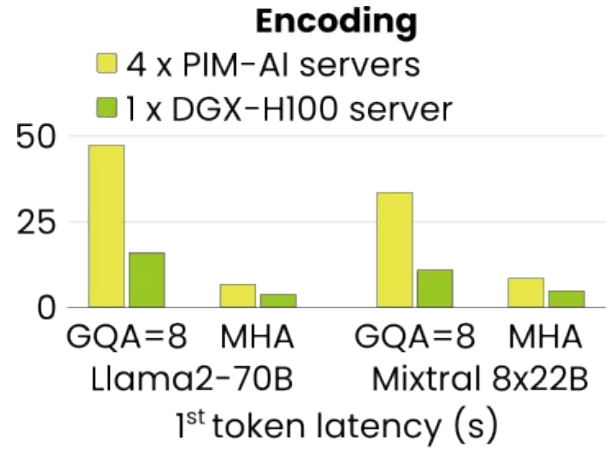
Inference of 1000 tokens in / 100 tokens out

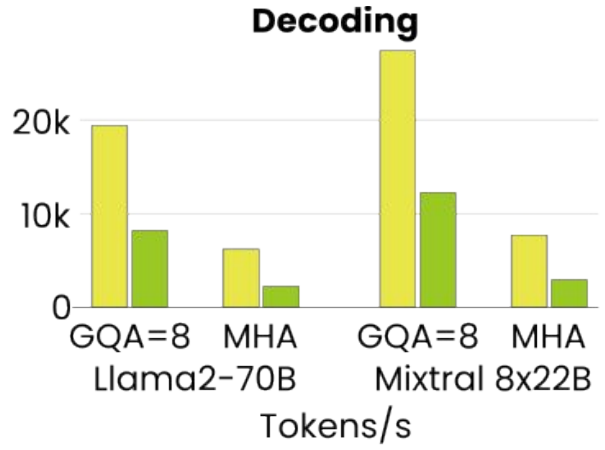
Llama-2-70B

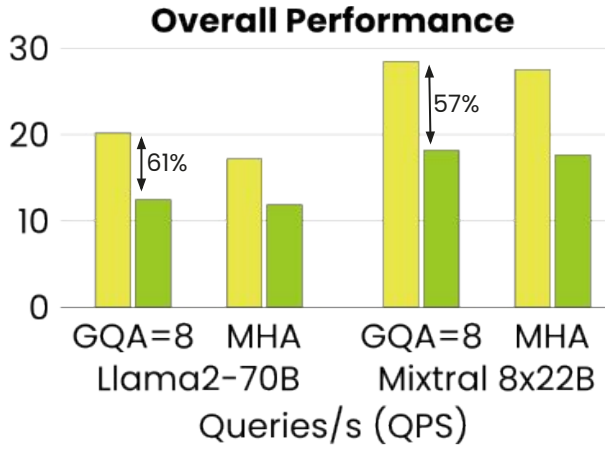
vocab_size	32000
embedding_size	8192
n_parameters	69B
n_context	4096
num. heads	32
num. kv heads	8

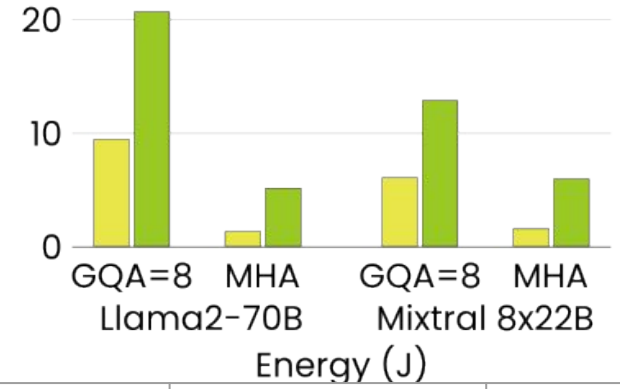
Mistral-8x22B

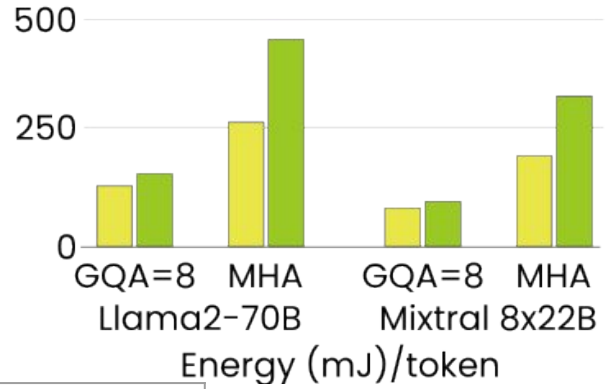
vocab_size	32000
embedding_size	16k
n_parameters	140B/40B
n_context	65k
num. heads	48
num. kv heads	8

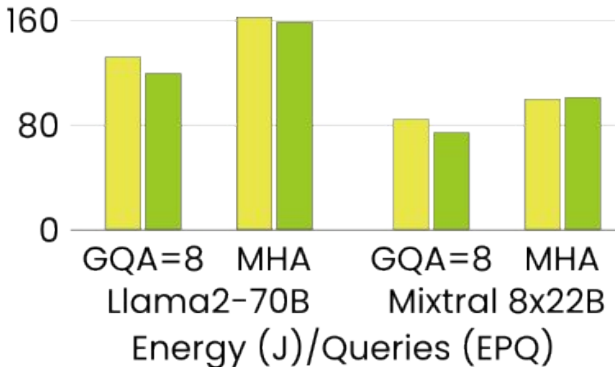












Weights data_type
16-bit
GEMMs data_type
16-bit
Activations data_type
32-bit

Batch size	Llama2-70B		Mixtral 8x22B	
	GQA = 8	MHA	GQA = 8	MHA
DGX-H100	200	46	200	88
PIM-AI server	80	10	80	20

Conclusions

- RISC-V IP with AI capabilities seamlessly integrated in LPDDR5 / DDR5 memory chips
  - No memory controller changes,
  - No memory PHY changes,
  - Up to 8 TFLOPs,
  - less than 1pJ/bit when accessing main memory

- Hardware evaluation shows:
  - Total cost of ownership per QPS can be improved up to 6.94x for cloud scenarios,
  - up to 49.6% better tokens/second in mobile scenarios,
  - energy efficiency per token improved from 10x to 20x in mobile scenarios

- PyTorch LLM simulator to be open sourced

- QEMU / gem5 simulator to be developed

Useful links

- Website
- Resource page
- Github
- SDK

Thank you

Cristobal Ortega, CPU Architect
cortega@upmem.com

Backup

Performance drivers of HW solutions for LLM on mobile

	Main memory (GENERATION)		Compute (SUMMARIZATION)		Notes
	Bandwidth GB / s	Energy pJ / bit	16-bit TFLOPs	MAX TOPS	
Apple A17 pro	51.2 LPDDR5 (8GB)	**> 20**	4.3 (GPU)	35 (ANE)	DECODING energy 10X higher than on-chip DRAM Pushing adoption of low precision datatypes (int4) **Qualcomm** : *"Low-bit integer precision is essential for power-efficient inference."* LPDDR interface will limit AI effective bandwidth Shared by all the AP processes Figures per system
Qualcomm Snapdragon 8 GEN3	77 LPDDR5X	**> 10**	4.73 (GPU A750)	34 (Hexagon)	
Mediatek Dimensity 9300	77 LPDDR5T		6 (GPU G720)	33 (APU 790)	
Samsung LPDDR5 PIM	102.4 internal (2GB)	**< 1**	**0.1024**	**0.2048**	On chip standalone processing is **IMPOSSIBLE** Not general purpose processing Poor algorithms flexibility & datatypes support Very heavy scheduling from host, and energy waste **DECODING only** Per chip = x4 for 4 LPDDR chips
UPMEM PIM-AI (1 chip)	**102.4** internal (2GB)		8 (TPU) 0.256 (VPU)	32 (TPU) 0.512 (VPU)	Per chip => x4 for 4 LPDDR chips

up mem

Example: GPT-3

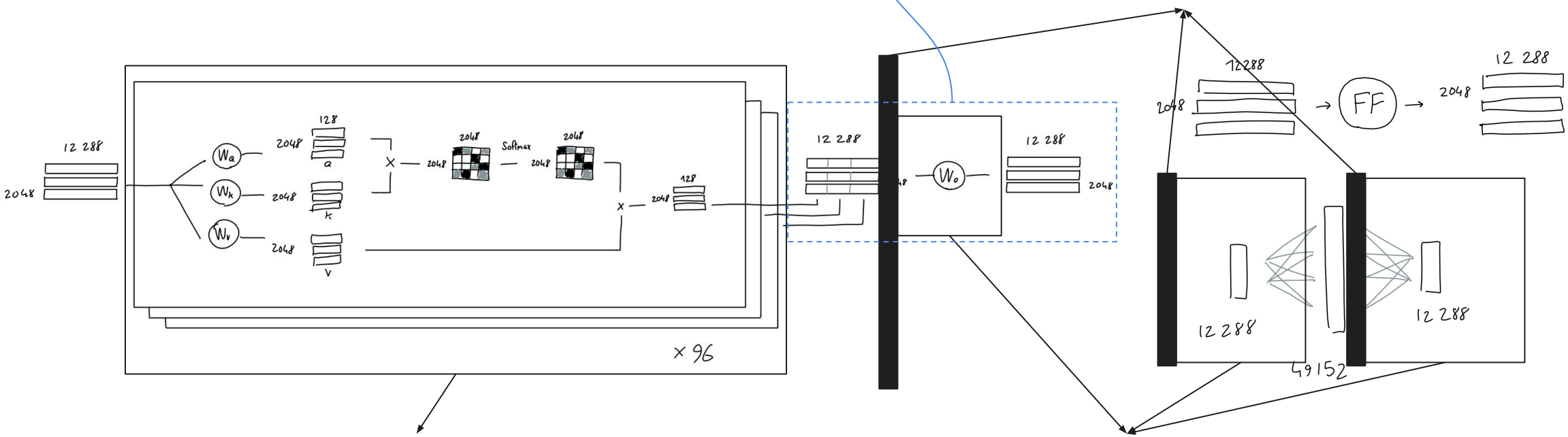
Tensor input representation of GPT-3:
[num_batches, num_tokens, embedding]
num_batches: different requests to the model
num_tokens: tokens within a request, usually padded to the longest request/batch
embedding: num. features representing a token

Operations in WO:
num_batches GEMMs of:
[num_tokens, embedding] x [embedding, embedding]

Parallelizing GPT-3 into 6 DIMMs:

n_context = 2048 rows
n_heads = 96
Hidden_size / dmodel / embedding size = 12288

Sync points with CPU



Each head of the MHA could be computed by 1 chip → 96 chips → 6 DIMMs

Data parallelism across 96 chips