IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Processing-In-Memory As An Impressive Computational Solution: A Review

Challenges of PIM, and a new direction for further research and innovation

[1]**Mrinal Bhaskar**

[1] Lead Patent Engineer & Research Scholar
[1] Operations, Licensing Team,
[1]Intellect-Partners, Delhi, India

Abstract: Workloads involving higher computational operations require impressive computational units. Computational operations may involve very complex operations of very high precision levels. These complex operations may involve artificial neural networks and require a very fast execution speed. Based on this particular requirement of high computational need for an efficient and fast execution operation, computational operations need to be performed in the memory system itself. The memory system with computational capabilities is called a Processor-In-Memory (PIM) or Computational Memory. In this paper, different PIM memories and their challenges have been reviewed, and their solutions have been proposed. The proposed solution tends to provide energy-efficient and dynamic configurable PIM units.

Index Terms **- Processor-In-Memory, PIM, computational memory, Near memory computing, near-data processing, computation-in-memory, high-performance computing**

I. INTRODUCTION

Artificial intelligence is permeating every aspect of life and all electronic devices. It employs various advanced operations such as image recognition, audio recognition, and pattern recognition, all of which rely on artificial neural networks with multiple layers and complex arithmetic operations. To execute these advanced arithmetic operations, an advanced memory system is required. Memory technologies encompass HBM, HMC, DDR5, GDDR6, and LPDDR5 memory technologies. The HBM and HMC memory systems stack multiple DRAMs together to create a stack. These memories can significantly enhance machine learning performance by utilizing their high bandwidth and computational capabilities.

The paper will provide an overview of the Processor-In-Memory (PIM) system with various memory technologies manufactured by different manufacturers. We will examine and explore different PIM techniques and discuss the challenges encountered in existing PIM systems. Lastly, **we will address new research directions that have the potential to enhance the computational capability of Processor-In-Memory systems by addressing these challenges.** This new direction will serve as a valuable reference for researchers investigating various PIM techniques.

This PIM system can be utilized for implementing various machine learning and artificial intelligence algorithms, including Deep Neural Networks (DNNs) and Convolutional Neural Networks (CNNs), by leveraging multiple artificial neural networks. The PIM system integrates a Neural Network Accelerator (NNA) within a memory system to expedite diverse computational operations. The Neural Network Accelerator executes computational tasks within memory cells independently of a host system or an external processing unit. The computed result is subsequently stored in memory cells.

The PIM system offers various computational capabilities that have captured researchers' attention for conducting diverse computational operations. One of the most critical aspects of PIM is its reduction in data movement. PIM facilitates minimized data movement operations, given that the computational processes occur within the memory cell itself.

Various manufacturers employ distinct memory technologies to implement the PIM system. The memory technologies utilized in a PIM system may include:

- DRAM
- HBM
- GDDR6
- LPDDR5X
- HMC

2.1 DRAM

JEDEC JESD79-4D and **JEDEC JESD79-5B** standards describe DDR5 DRAM and DDR4 DRAM memory systems, which are types of dynamic random-access memory, and are extensively employed in various computational operations. DRAM stores data in cells, each consisting of one transistor and one capacitor. Binary data, represented as 0 or 1, is stored in memory cells by using fully charged and discharged capacitors.

Table 1 — DDR4 - 2 Gb/ 4Gb/ 8Gb/ 16 Gb Addressing Table

Configuration		512 Mb/ 1GB/ 2GB/ 4GB x4	256 Mb/ 512 Mb/ 1GB/ 2 GB x8	128 Mb/ 256 Mb/ 512 Mb/ 1GB x16
Bank Address	# of Bank Groups	4	4	2
	BG Address	BG0~BG1	BG0~BG1	BG0
	Bank Address in a BG	BA0~BA1	BA0~BA1	BA0~BA1
Row Address		A0~A14	A0~A13	A0~A13
Column Address		A0~A9	A0~A9	A0~A9
Page size		512B	1KB	2KB

Table 2 — DDR5 - 8 Gb/ 16 Gb/24 Gb/ 32Gb/ 64 Gb Addressing Table

Configuration		2 Gb/ 4Gb/ 6 Gb/ 8Gb/ 16Gb x4	1 Gb/ 2 Gb/ 3 Gb/ 4 Gb/ 8 Gb x8	512 Mb/ 1 Gb/ 1.5 Gb/ 2 Gb/ 4 Gb x16
Bank Address	BG Address	BG0~BG2	BG0~BG2	BG0~BG1
	Bank Address in a BG	BA0	BA0	BA0
	# BG / # Banks per BG / # Banks	8 / 2 / 16	8 / 2 / 16	4 / 2 / 8
Row Address		R0~R15	R0~R15	R0~R15
Column Address		C0~C10	C0~C9	C0~C9
Page size		1KB	1KB	2KB
Chip IDs / Maximum Stack Height		CID0~3 / 16H	CID0~3 / 16H	CID0~3 / 16H

2.2 HBM

JEDEC JESD238A Standard describes the HBM3 memory system, which stacks multiple DRAM memory systems together to perform various memory and computational operations. Multiple memory dies are stacked and connected by TSV material, responsible for transmitting data and command signals.

Table 3 — HBM3 Channel Addressing

Density per Channel	**2 Gb**	**4 Gb**	**6 Gb**	**8 Gb**		
Density per PC	**1 Gb**	**2 Gb**	**3 Gb**	**4 Gb**		
Page Size per PC	1 KB	1 KB	1 KB	1 KB		
Refresh Period	3.9 µs	3.9 µs	3.9 µs	3.9 µs		
Configuration[4]	**8 Gb 8High**	**8 Gb 12High**	**8 Gb 16High**	**16 Gb 8High**	**16 Gb 12High**	**16 Gb 16High**
Density per Channel	**4 Gb**	**6 Gb**	**8 Gb**	**8 Gb**	**12 Gb**	**16 Gb**
Density per PC	**2 Gb**	**3 Gb**	**4 Gb**	**4 Gb**	**6 Gb**	**8 Gb**
Page Size per PC	1 KB	1 KB	1 KB	1 KB	1 KB	1 KB
Refresh Period	3.9 µs	3.9 µs	3.9 µs	3.9 µs	3.9 µs	3.9 µs
Configuration[4]	**24 Gb 8High**	**24 Gb 12High**	**24 Gb 16High**	**32 Gb 8High**	**32 Gb 12High**	**32 Gb 16High**
Density per Channel	**12 Gb**	**18 Gb**	**24 Gb**	**16 Gb**	**24 Gb**	**32 Gb**
Density per PC	**6 Gb**	**9 Gb**	**12 Gb**	**8 Gb**	**12 Gb**	**16 Gb**
Page Size per PC	1 KB	1 KB	1 KB	1 KB	1 KB	1 KB
Refresh Period	3.9 µs	3.9 µs	3.9 µs	3.9 µs	3.9 µs	3.9 µs

2.3 GDDR6

JEDEC JESD250D describes the GDDR6 memory system, known as GRAPHICS DOUBLE DATA RATE. It offers high-speed dynamic random-access memory for use during high computational operations. Additionally, the GDDR6 memory implements two independent 16-bit channels.

Table 4 — GDDR6 Channel Addressing

Memory Density	8 Gb		12 Gb		16 Gb		24 Gb		32 Gb	
Device Organization	**x16 mode**	**x8 mode**	**x16 mode**	**x8 mode**	**x16 mode**	**x8 mode**	**x16 mode**	**x8 mode**	**x16 mode**	**x8 mode**
Number channels	2		2		2		2		2	
Channel Memory Density	4 Gb		6 Gb		8Gb		12 Gb		16 Gb	
Channel Density	4,294,967,296		6,442,450,944		8,589,934,592		12,884,901,888		17,179,869,184	
Array Pre-Fetch (bits, per channel)	256	128	256	128	256	128	256	128	256	128
Page Size (per channel)	2K	2K	4K	2K	4K	2K	4K	2K	4K	2K
Refresh	16K/32 ms		16K/32 ms		16K/32 ms		16K/32 ms		16K/32 ms	

2.4 LPDDR5X

JEDEC JESD209-5C describes the LPDDR5 memory system, known as Low Power Double Data Rate. This memory provides high-speed dynamic random-access memory that can be used during high computational operations.

Table 5 — LPDDR5X SDRAM x16 Mode Addressing for BG Mode (4Banks/4Bank Groups)

Memory Density	2 Gb	3 Gb	4 Gb	6 Gb	8 Gb	12 Gb	16 Gb	24 Gb	32 Gb
Configuration	8 Mb x 16DQx 4 BG x 4 banks	12 Mb x 16DQ x 4 BG x 4 banks	16 Mb x 16DQ x 4 BG x 4 banks	24 Mb x 16DQx 4 BG x 4 banks	32 Mb x 16DQ x 4 BG x 4 banks	48 Mb x 16DQ x 4 BG x 4 banks	64 Mb x 16DQ x 4 BG x 4 banks	96 Mb x 16DQ x 4 BG x 4 banks	128 Mb x 16DQ x 4 BG x 4 banks
Number of Banks in BG	4	4	4	4	4	4	4	4	4
Number of Bank Groups	4	4	4	4	4	4	4	4	4
Array Pre-Fetch	256	256	256	256	256	256	256	256	256
Number of Rows	8,192	12,288	16,384	24,576	32,768	49,152	65,536	98,304	131,072
Number of Columns (fetch boundaries)	64	64	64	64	64	64	64	64	64
Page Size (Bytes)	2,048	2,048	2,048	2,048	2,048	2,048	2,048	2,048	2,048

2.5 HMC

Hybrid Memory Cube Specification 2.1 describes the HMC memory system, which stacks various DRAM dies and one logic die. Furthermore, these dies are connected via Through-Silicon Via (TSV) technology. The memory system includes a built-in memory controller for controlled memory operations.

Table 6 — HMC Configurations

	Configurations
Number of links in package	2, 4
Link lane speed (Gb/s)	12.5, 15, 25, 28, 30
Link width[1]	Full, half, quarter
Memory density	4GB, 8GB
Number of vaults	32
Memory banks	4GB: 256 banks 8GB: 512 banks
Maximum aggregate link bandwidth[2]	480 GB/s (3.84 Tb/s)
Maximum DRAM data bandwidth	320 GB/s (2.56 Tb/s)
Maximum vault data bandwidth	10 GB/s (80 Gb/s)

Various memory technologies can be employed to implement Processor-In-Memory systems. These memory systems offer varying memory access latencies and computational capabilities.

III. DIFFERENT PIM SYSTEM

Several existing Processing-In-Memory solutions in the market include UPMEM PIM, Samsung HBM-PIM, Samsung LPDDR5-PIM, SK Hynix GDDR6-AIM, and others.

3.1 SK Hynix GDDR6-AIM

SK Hynix GDDR6-AIM is extensively used in various machine learning and artificial intelligence operations. It can accelerate different machine learning algorithms, such as RNN, LSTM, and MLP, by offloading specific mathematical operations from external processing units.

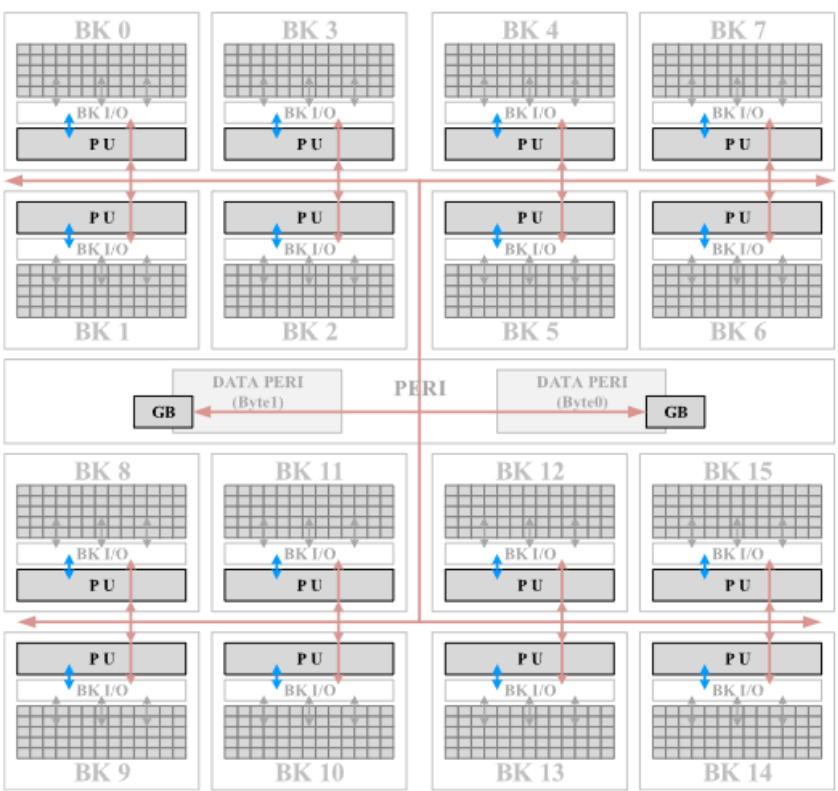

Figure 1. AiM architecture of SK Hynix GDDR6-AIM

The processing units (PU) are located within the memory system, right next to the memory bank, to perform various arithmetic and logical operations in the AiM.

Table 7: Dedicated DL CMD Set implemented in SK Hynix AIM architecture

Bank Activation	
ACT4, ACT16	Activate four/sixteen banks in parallel
ACTAF4, ACTAF16	Activate rows storing Activation Functions LUTs in four/sixteen banks in parallel
Compute Commands	
MACSB, MAC4B, MACAB	Perform MAC in one/four/sixteen banks in parallel
AF	Compute Activation Function in all banks
EWMUL	Perform element-wise multiplication
Data Commands	
RDCP	Copy data from a bank to the Global Buffer
WRCP	Copy data from the Global Buffer to a bank
WRGB	Write to Global Buffer (often Activation vector data)
RDMAC	Read from MAC result register
RDAF	Read from Activation Function result register
WRMAC	Write to MAC result register (or WRBIAS as often BIAS data is written)
WRBK	Write to all activated banks in parallel

Table 7 describes various CMD sets for DL operations. These operations are executed by the PUs located at each memory bank.

3.2 Samsung HBM-PIM and LPDDR5-PIM

The Samsung Aquabolt-XL memory system implements the HBM-PIM system for executing various computational operations.

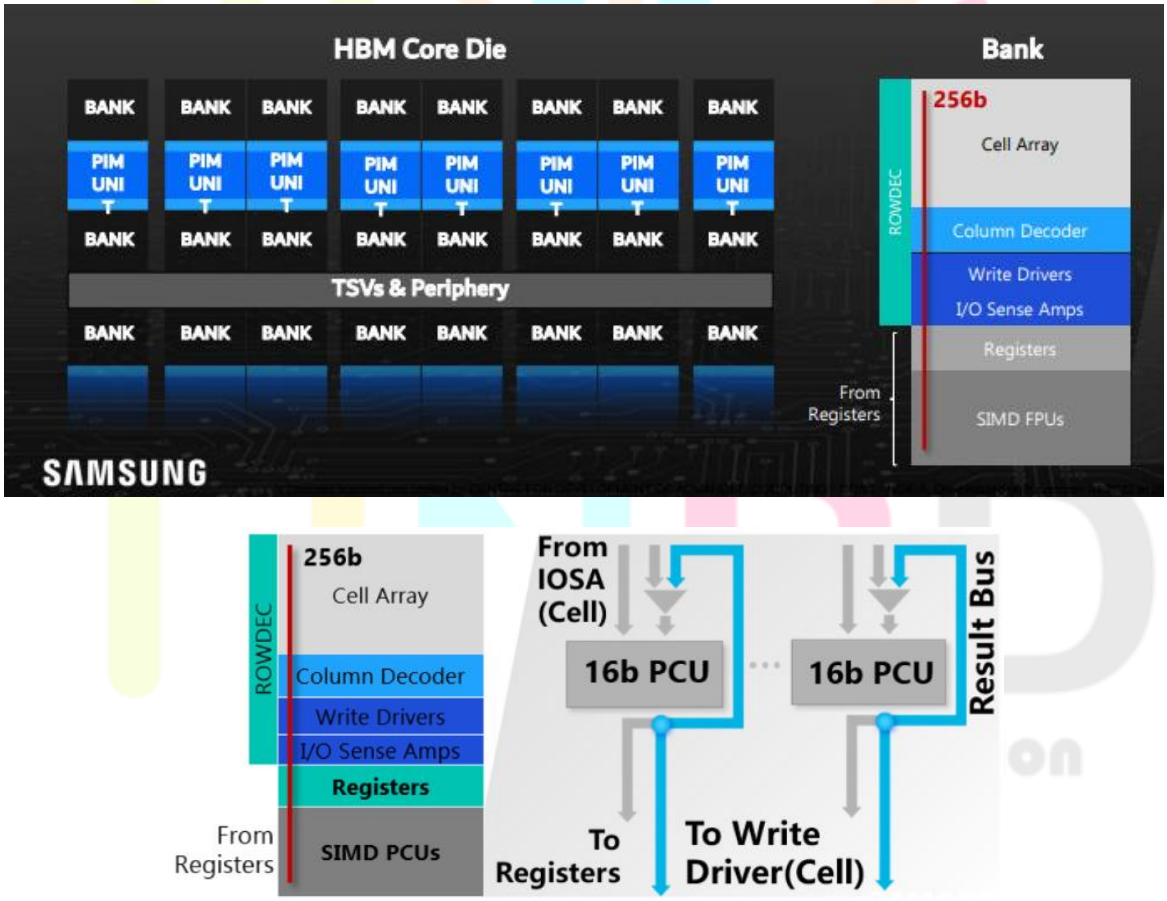

Figure 2. HBM-PIM architecture of Samsung

Each memory bank includes its corresponding PIM units for executing various computational operations. Each DRAM array comprises a bit-line sense amplifier (BLSA) and a word-line driver.

Table 8: Dedicated DL CMD Set implemented in Samsung HBM-PIM architecture

Op. Type	Operand (SRC0)	Operand (SRC1)	Result (DST)	# of Combinations
MUL	GRF, BANK	GRF, BANK, SRF_M	GRF	32
ADD	GRF, BANK, SRF_A	GRF, BANK, SRF_A	GRF	40
MAC	GRF, BANK	GRF, BANK, SRF_M	GRF_B	14
MAD	GRF, BANK	GRF, BANK, SRF_M SRF_A (for SRC2)	GRF	28
MOV (ReLU)	GRF, BANK		GRF	24

Table 8 describes various CMDs for executing computational operations by the PIM unit within a memory system.

3.3 UPMEM PIM

UPMEM PIM comprises multiple DPUs for executing various computational operations. Each DPU implements a specific Instruction Set Architecture (ISA) to carry out multiple operations.

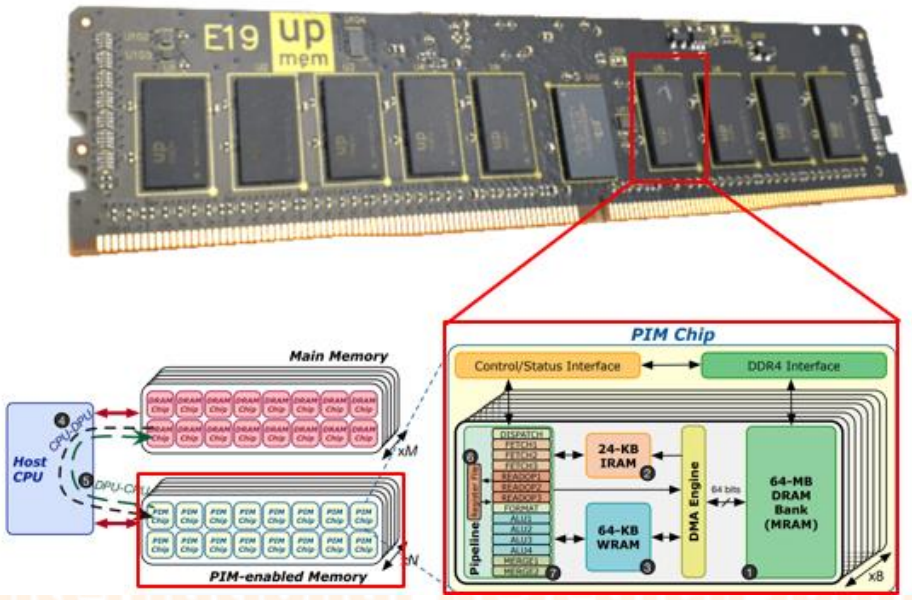

Figure 3. PIM architecture of UPMEM

Various commands in the ISA are implemented to execute diverse machine learning operations.

Table 8: Dedicated DL CMD Set implemented in Samsung HBM-PIM architecture

Domain	Benchmark	Short name
Dense linear algebra	Vector Addition	VA
	Matrix-Vector Multiply	GEMV
Sparse linear algebra	Sparse Matrix-Vector Multiply	SpMV
Databases	Select	SEL
	Unique	UNI
Data analytics	Binary Search	BS
	Time Series Analysis	TS
Graph processing	Breadth-First Search	BFS
Neural networks	Multilayer Perceptron	MLP
Bioinformatic	Needleman-Wunsch	NW
Image processing	Image histogram (short)	HST-S
	Image histogram (long)	HST-L
Parallel primitives	Reduction	RED
	Prefix sum (scan-scan-add)	SCAN-SSA
	Prefix sum (reduce-scan-scan)	SCAN-RSS

Table 8 describes various deep learning operations that can be computed by the PIM chip inside the UPMEM chip.

IV. CHALLENGES OF A PIM MEMORY SYSTEM

Multiple PIM systems have been discussed so far, but these PIM systems face certain challenges that require solutions to overcome them. Some of the challenges faced by these memory systems are:

- PIM systems have been limited to a very specific set of instruction sets for performing a particular set of computational operations.
- Fixed precision level processing units in the PIM system are constrained to perform computational operations at a fixed precision level.

The main challenges encountered by the PIM system have been identified, and our solution proposes remedies for these challenges.

V. OUR PROPOSED SOLUTION

Our proposed solution involves:

- Implementing a dynamically configurable processing unit inside the PIM system.
- Utilizing an adaptable precision-sized hardware unit for computational operations.

The dynamically reconfigurable processing unit can be adjusted based on the detected or initiated "recong_PU" condition during PIM system's computational operations. **A dynamically configurable processing unit, achieved using configurable devices, can replace a processing unit with a fixed instruction set.** Generating configuration values for configuring the processing unit can be accomplished using the dynamic microcode of the processing unit.

The reconfiguration of the processing unit can be performed by writing corresponding configuration values to a "RECon_Fn" flag within configuration registers located near the processing unit of the memory bank

Furthermore, **the PIM system can be equipped with adaptable hardware resources with flexible precision levels, resulting in reduced energy consumption during memory operations**. Lower precision operations can be carried out using hardware resources with low precision capabilities, contributing to energy conservation. On the other hand, machine learning operations requiring higher precision can be performed using hardware resources with enhanced precision capabilities.

VI. CONCLUSION

In the proposed system, the challenges of a limited instruction set architecture and fixed precision basis have been resolved by employing dynamically reconfigurable processing units and variable precision-based hardware units. As a result, these two proposed solutions can effectively address the primary challenges encountered by the PIM system. This paper offers a comprehensive review of various PIMs, their challenges, and the solutions, which can serve as a roadmap for future research aimed at enhancing the PIM system.

REFERENCES

[1]. JEDEC, “DDR4 SDRAM,” JESD79-4D, 2021

[2]. JEDEC, “DDR5 SDRAM,” JESD79-5B_v1.20, 2022

[3]. JEDEC, “HBM3 SDRAM,” JESD238A, 2023

[4]. JEDEC, “LPDDR5 SDRAM,” JESD209-5B, 2021

[5]. Hybrid Memory Cube Specification 2.1, 2014

[6]. T. Brown et al., “Language models are few-shot learners,” Advances in neural information processing systems, vol. 33, pp. 1877–1901, 2020.

[7]. O. Mutlu, et al., “Processing data where it makes sense: Enabling in-memory computation,” Microprocessors and Microsystems, vol. 67, pp. 28–41, 2019.

[8]. 5. S. Lee et al., “A 1ynm 1.25 V 8Gb, 16Gb/s/pin GDDR6-based Accelerator-in-Memory supporting 1TFLOPS MAC Operation and Various Activation Functions for Deep-Learning Applications,” in 2022 IEEE International Solid-State Circuits Conference (ISSCC), vol. 65, pp. 1–3, 2022.

[9]. S. Lee et al., “Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology: Industrial Product,” ACM/IEEE ISCA, pp. 43-56, 2021.

[10]. H. Park, S. Kim, “Hardware accelerator systems for artificial intelligence and machine learning.” Advances in Computers. Vol. 122. Elsevier, 2021, pp. 51-95.

[11]. S. Han, X. Liu, H. Mao, J. Pu, A. Pedram, M. A. Horowitz, and W. J. Dally, "Eie: efficient inference engine on compressed deep neural network," in Proceedings of the 43rd International Symposium on Computer Architecture , IEEE Press, 2016.

[12]. Y.-C. Kwon et al., "25.4 A 20nm 6GB Function-In-Memory DRAM, Based on HBM2 with a 1.2TFLOPS Programmable Computing Unit Using Bank-Level Parallelism, for Machine Learning Applications," ISSCC, pp. 350-352, 2021.

[13]. M. He et al., "Newton: A DRAM-maker's Accelerator-in-Memory (AiM) Architecture for Machine Learning," IEEE/ACM MICRO, pp. 372-385, 2020

[14]. T. M. Hollis et al., "25.3 An 8Gb GDDR6X DRAM Achieving 22Gb/s/pin with SingleEnded PAM4 Signaling," ISSCC, pp. 348-350, 2021.

[15]. D. Kalamkar, et al., ''A study of BFLOAT16 for deep learning training,'' arXiv (preprint), id. 1905.12322, 2019.

[16]. M. He et al., "Newton: A DRAM-maker's accelerator-in-memory (AiM) 480 architecture for machine learning," in Proc. 53rd Annu. IEEE/ACM 481 Int. Symp. Microarchitecture (MICRO), Oct. 2020, pp. 372–385, doi: 482 10.1109/MICRO50266.2020.00040

[17]. T. M. Hollis et al., "An 8-Gb GDDR6X DRAM achieving 484 22 Gb/s/pin with single-ended PAM-4 signaling," IEEE J. Solid485 State Circuits, vol. 57, no. 1, pp. 224–235, Jan. 2022, doi: 486 10.1109/JSSC.2021.3104093.

[18]. F. Devaux, "The true processing in memory accelerator," in 472 Proc. IEEE Hot Chips Symp. (HCS), Aug. 2019, pp. 1–24, doi: 473 10.1109/HOTCHIPS.2019.8875680.

[19]. K. Sohn et al., "A 1.2V 20nm 307GB/s HBM DRAM with At-Speed Wafer-Level I/O Test Scheme and Adaptive Refresh Considering Temperature Distribution," ISSCC, pp. 316-317, 2016

[20]. J. H. Cho et al., "A 1.2V 64Gb 341GB/s HBM2 Stacked DRAM with Spiral Point-to Point TSV Structure and Improved Bank Group Data Control," ISSCC, pp. 208-209, 2018.

VII. ABOUT AUTHOR

Mrinal Bhaskar holds the position of Lead Patent Engineer at Intellect-Partners and possesses a B.Tech degree. Proficient in patent analysis and searching, he assists clients with various Intellectual Property projects including Infringement searches, claim charting, Invalidity searches, and prior art searches.

He holds substantial expertise across a range of fields, encompassing memory standards (DDR5, DDR4, LPDDR5, LPDDR5, HBM3, GDDR6, NVM, ONFI, eMMC, UFS, NVDIMM-P, NVDIMM-N, and others), Microcontrollers, Configurable logic/Programmable logic (MCU/PLU/CLC), Computer Processors, Computer Graphics, Artificial Intelligence, Autonomous technologies, and Autonomous Driving systems. His proficiency also extends to various miscellaneous technologies such as Printers, Semiconductors, E-commerce, Recommendation systems, Fitness tech, Projectors, Cameras, Virtual Reality, and other technologies.

Email id: mrnlbhaskar@gmail.com
ORC ID: https://orcid.org/0009-0001-5447-7117