PIM and Wrap-up

Lecture 20
May 15, 2025

PIM Motivations

- Want to process many items quickly
  - Need lots of parallel compute resources
  - Need to keep parallel compute resources fed with data
- Want to deliver large amounts of data to compute resources quickly
  - Need lots of memory bandwidth

PIM Motivations

- Want to process many items quickly
  - Need lots of parallel compute resources
  - Need to keep parallel compute resources fed with data
- Want to deliver large amounts of data to compute resources quickly
  - Need lots of memory bandwidth

Number of pins used to bring data onto chip limits memory bandwidth

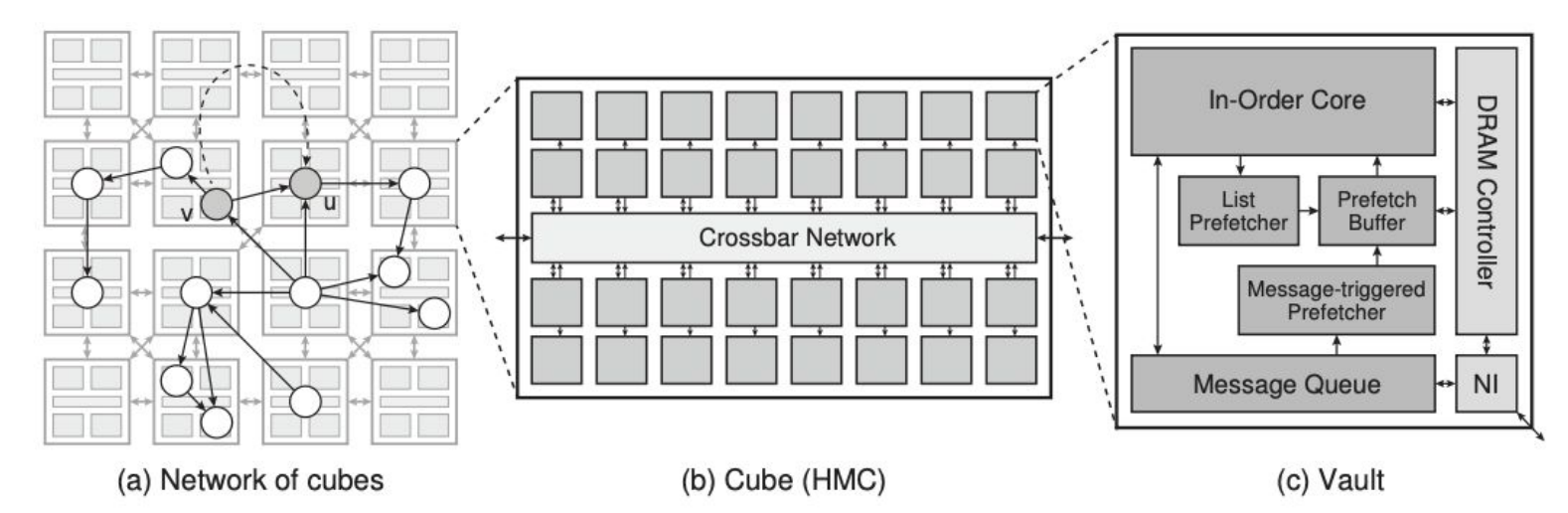
Characteristics of Graph Analysis Workloads

- Large amounts of random memory accesses across large memory regions
  - Poor locality
- High degrees of compute parallelism
- Very small amounts of computation per memory item
  - Hard to hide memory latency except with massive amounts of compute parallelism

Key Insights

- 3D-stacked memory has high **internal** memory bandwidth despite external memory bandwidth still being limited by pin counts
- 3D-stacked memory can be built with a logic layer where computation can be performed
    - Data can be supplied to logic layer at internal memory bandwidth rates

Tesseract



Hybrid Memory Cube

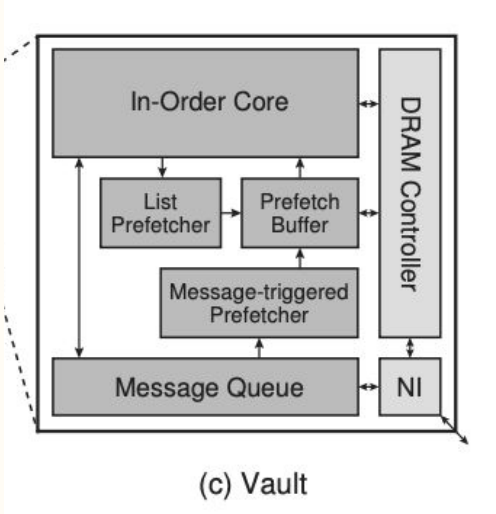
- 8 8Gb DRAM layers
  - Organized into 32 vertical slices called vaults
  - Vaults in a HMC
    - Connected by crossbar
    - Contain 16-bank DRAM partition and memory controller
    - Contain single-issue in-order core
    - 16GB/s internal memory bandwidth to core
- 8 40 GB/s serial links to off-chip

Tesseract and Host Processor

- Tesseract
  - is memory-mapped to non-cacheable memory region of host
  - does not support virtual memory
- Host processor distributes input graphs across vaults using customized memory allocator API
- Applications see single physical address space over all HMCs

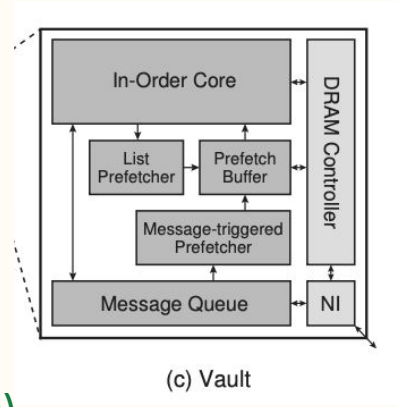
Tesseract Vaults

- Compute cores can only access local DRAM partition (i.e., vault)
- Message passing used to communicate between compute cores
- Remote function call to execute computation on another vault
- Each vault has
  - list prefetcher
  - message-triggered prefetcher



(c) Vault

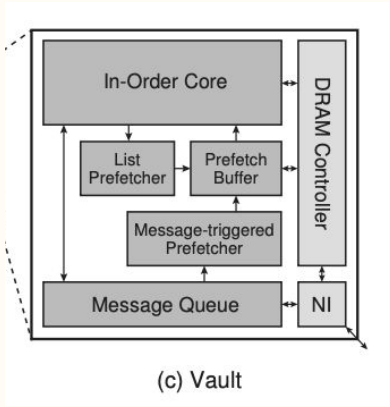
Remote Function Calls



(c) Vault

- Blocking
  - Executes remotely in interrupt mode (non-preemptible)
  - Remote core sends return value
- Non-blocking
  - Remote cores can delay execution of RFCs, enabling batch processing
    - Message queue collects them
    - Do not cross synchronization barriers
  - Can specify memory address to prefetch for this computation

Prefetching



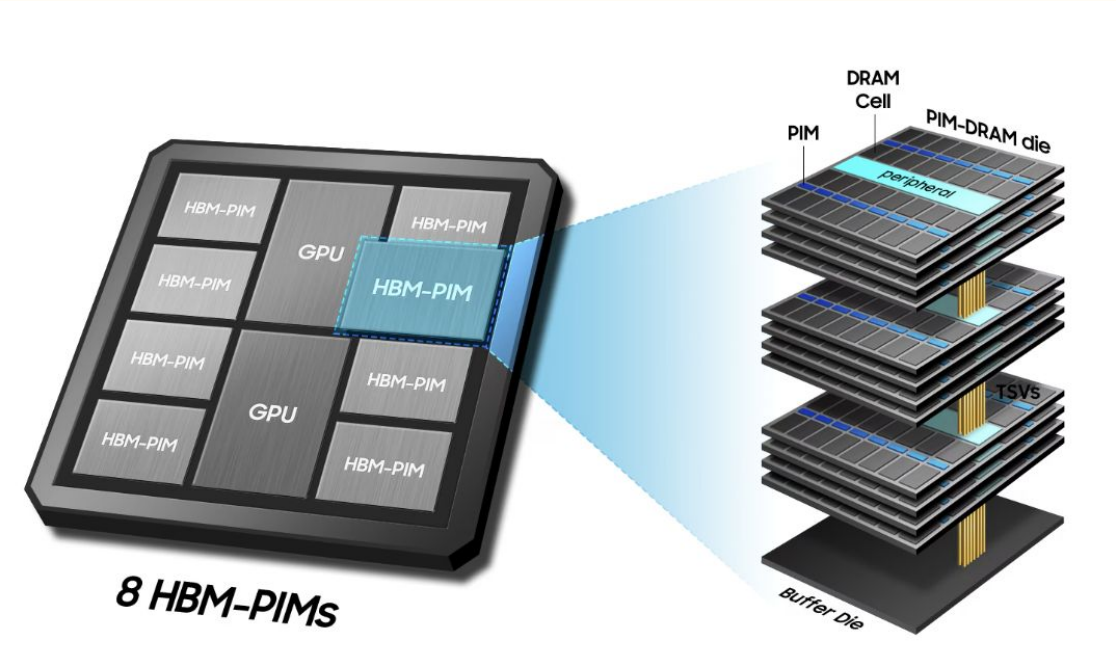
(c) Vault

Prefetch buffer stores prefetched data to not pollute L1 cache

- List prefetcher
  - Strided access patterns predicted
  - Application can provide hints about strided accesses
- Message triggered prefetcher
  - Prefetch data from RFC before execution of RFC
  - Messages marked ready once any prefetched data available

What did they find?

Samsung HBM-PIM (for Neural Network)



https://semiconductor.samsung.com/us/news-events/tech-blog/hbm-pim-cutting-edge-memory-technology-to-accelerate-next-generation-ai/

Samsung HBM-PIM

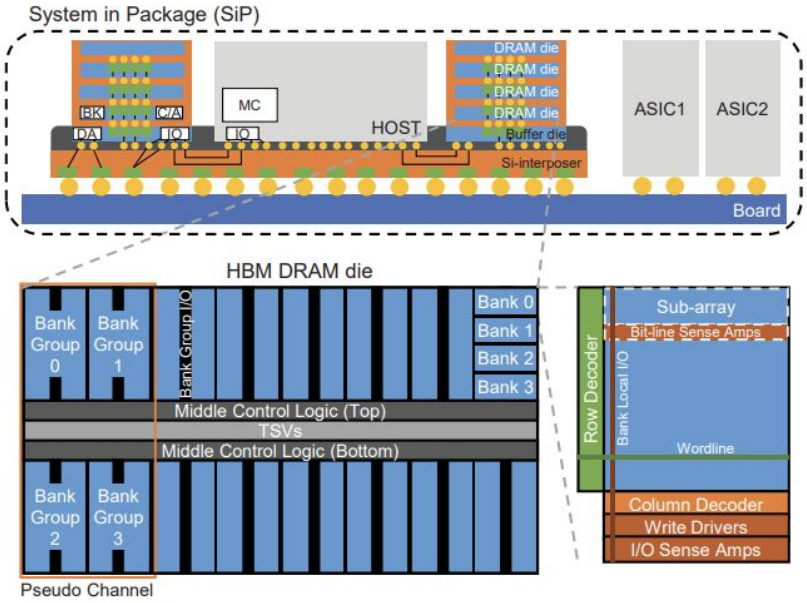


Fig. 2: A cross-section view of a system in package (SiP) comprising an ASIC and HBM devices (top) and an HBM DRAM die organization (bottom).

"Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology" by Lee et al. (ISCA 2021)

Samsung HBM-PIM

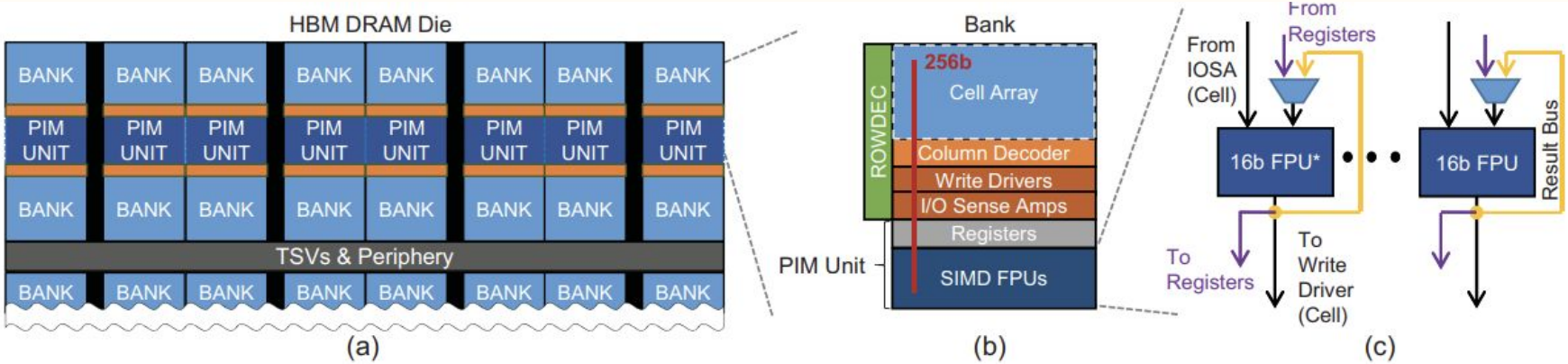


Fig. 1: (a) HBM DRAM die organization, (b) bank coupled with a PIM unit, (c) PIM datapath.

Each PIM execution unit in a bank can grab 16 16bit operands per request
16 operand wide SIMD FPU

"Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology" by Lee et al. (ISCA 2021)

Samsung HBM-PIM

TABLE II: The supporting operations, operand sources, and result destination.

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

"Hardware Architecture and Software Stack for PIM Based on Commercial DRAM Technology" by Lee et al. (ISCA 2021)

What Do the Different Compute Paradigms Do Well?

- Unix processes
- MPI
- Pthreads / OpenMP
- CUDA
- MapReduce
- Spark
- PIM