

ReTransformer

ReRAM-based Processing-in-Memory Architecture for Transformer Acceleration

Reza Adinepour

Amirkabir University of Technology (Tehran Polytechnic)

`adinepour@aut.ac.ir`

Computer Engineering Department

June 27, 2024



Presentation Overview

- ① Memory Architecture Overview
 - Uniform Memory Access (UMA)
 - Non-uniform Memory Access (NUMA)
 - Cache-only Memory Access (COMA)
- ② Code Snippets
- ③ Differences between UMA and NUMA

Memory Architecture Overview

What is this structure?

— Defines how computer memory is organized and accessed.

- ① Uniform Memory Access (UMA)
- ② Non-Uniform Memory Access (NUMA)
- ③ Cache-Only Memory Access (COMA)

— In this presentation we talk about UMA and NUMA Architecture

Uniform Memory Access (UMA)

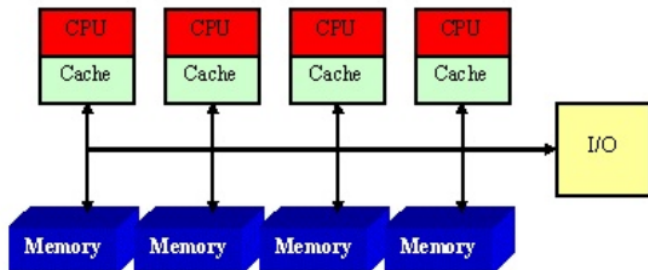


Figure: Processors with equal access to memory

- ① Same latency for all processors to access memory.
- ② Hardware cache typically present with each processor.

Uniform Memory Access (UMA) (Cont.)

- ① Equal memory access for all processors.
- ② Shared memory, any processor can access any part at any time.
- ③ Simple, cost-effective, highly scalable.

Advantages:

- ① **Ease of Implementation:** Minimal hardware modifications, cost-effective.
- ② **Scalability:** Easily scales with more processors without impacting access times.

Disadvantages:

- ① **Memory Contention:** Increased processors may lead to slower access times.
- ② **Limited Bandwidth:** Shared memory bus can become a bottleneck.

Uniform Memory Access (UMA) (Cont.)

Example System:

① Symmetric Multiprocessing (SMP) System:

- ① Multiple processors share common memory.
- ② Controlled by a single operating system.
- ③ Common in servers and high-performance computing.

Summary:

- ① **Strengths:** Simplicity and scalability.
- ② **Weaknesses:** Potential for memory contention and limited bandwidth in larger systems.

Non-uniform Memory Access (NUMA)

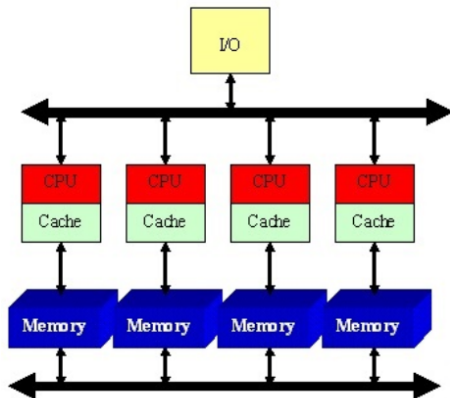


Figure: Processors with equal access to memory

- 1 Each processor has its local memory.

Non-uniform Memory Access (NUMA) (Cont.)

- ① Memory divided into multiple banks, each processor has its local bank.
- ② Processors can access other banks, but at higher latency than local access.
- ③ Efficient memory resource use, potential better performance than UMA for certain workloads.

Advantages:

- ① **Reduced Memory Contention:** Each processor has its local memory, minimizing contention.
- ② **Increased Memory Bandwidth:** Local memory banks lead to higher bandwidth than UMA.
- ③ **Efficient Memory Use:** Allocation based on processor needs enhances resource utilization.

Non-uniform Memory Access (NUMA) (Cont.)

Advantages:

- ① **Higher Implementation Complexity:** Additional hardware and software complexity.
- ② **Higher Latency for Remote Access:** Accessing remote memory incurs higher latency.

Example System:

① **Multi-Socket Server:**

- ① Each socket has its processors and memory banks.
- ② Sockets connected via a high-speed interconnect.
- ③ Commonly used in data centers, offers better performance for specific workloads.

Code Snippets

Run the [code!](#)

— This code is generated by ChatGP

Differences between UMA and NUMA

① Memory access time:

① NUMA:

Memory access time varies depending on the location of the data in memory. Accessing data in the local memory of a processor is faster than accessing data in the memory of a remote processor.

② UMA:

Memory access time is uniform across all processors since they share the same memory pool.

② Scalability:

① NUMA architecture is highly scalable and can support a large number of processors.

② UMA architecture is not as scalable as NUMA and may face performance issues when used with a large number of processors.

The End

Questions? Comments?

You can find this slides here:

[github.com/M-Sc-AUT/M.Sc-Computer-Architecture/Memory
Technologies](https://github.com/M-Sc-AUT/M.Sc-Computer-Architecture/MemoryTechnologies)