# Dynamic data loading from Flash to DRAM for LLM inference

Licheng Xue

2024.08.06

the Future of Memory and Storage

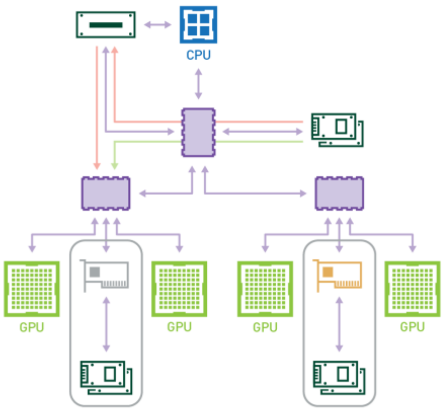
# LLM is driving the next industrial revolution

the Future of Memory and Storage

## GPUs and HBM in the AI Era: Driving Innovation and Performance !

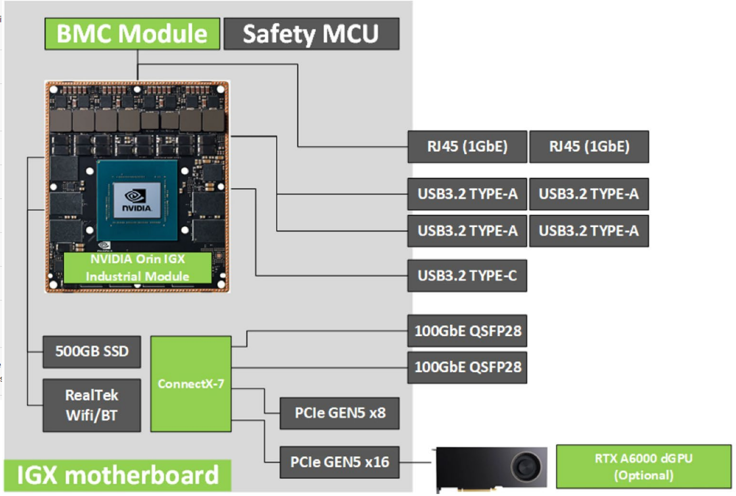
### NVIDIA DGX H200

| Specifications | |
|---|---|
| GPU | 8x NVIDIA H200 Tensor Core GPUs, with 141GB of GPU memory each |
| GPU memory | 1,128GB total |
| Performance | 32 petaFLOPS FP8 |
| NVIDIA NVSwitch™ | 4x |
| System power usage | 10.2kW max |
| CPU | Dual Intel® Xeon® Platinum 8480<br>112 Cores total, 2.00 GHz (Base),<br>3.80 GHz (Max Boost) |
| System memory | 2TB |
| Networking | 4x OSFP ports serving 8x single-<br>ConnectX-7 VPI<br>> Up to 400Gb/s InfiniBand/Et<br>2x dual-port QSFP112 NVIDIA Co<br>> Up to 400Gb/s InfiniBand/Et |
| Management network | 10Gb/s onboard NIC with RJ45<br>100Gb/s Ethernet NIC<br>Host baseboard management cc<br>(BMC) with RJ45 |
| Storage | OS: 2x 1.92TB NVMe M.2 |
| Internal storage: | 8x 3.84TB NVMe U.2 |




### NVIDIA IGX Orin

| | IGX Orin Developer Kit |
|---|---|
| AI Performance | Up to 1705 TOPs with optional RTX 6000 Ada GPU (Sparse) |
| SOM (System on Module) | GPU: 2,048-core NVIDIA Ampere architecture with 64 Tensor Cores<br>CPU: 12-core Arm® Cortex®-A78AE v8.2 |
| NVIDIA ConnectX-7 | NVIDIA ConnectX-7<br>2x 100GbE<br>32-lane Gen 5 PCIe swi<br>Downstream) |
| Safety MCU (sMCU) | Infineon Aurix TC397 |
| NVIDIA BMC (Baseboard Management Controller) Module | Aspeed AST2600<br>Microchip ERoT |
| GPU Max Frequency | 1.185GHz |
| CPU Max Frequency | 1.971GHz |
| DL Accelerator | 2x NVDLA 2.0 Engines |
| DLA Max Frequency | 1.4GHz |
| Vision Accelerator | 1x PVA v2 |
| Memory | 64GB 256-bit LPDDR5<br>204.8GB/s |
| Storage | 500GByte NVMe (PCIe<br>(4x SATA 6Gbps expans |



# Storage for LLM systems

## High-Speed Data Storage and Access

- Fast Reading of Training Data
- Storing Intermediate Results and Checkpoints

## Accelerating the Inference Process

- Model Loading
- Real-Time Data Processing

## GPUDirect Storage

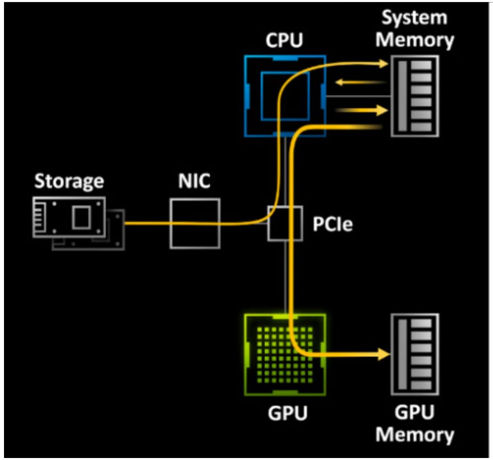
- Avoid extra copies through a bounce buffer in the CPU's memory
- Move data on a direct path into or out of GPU memory

Time To First Token

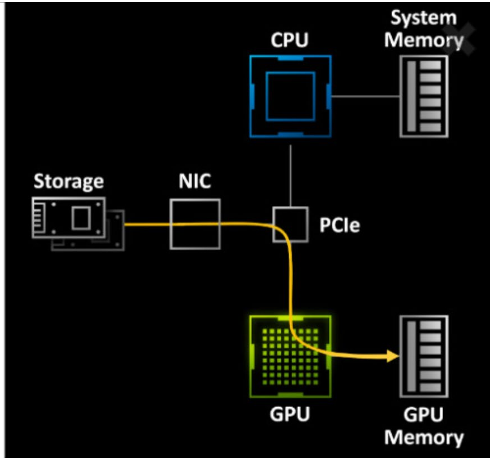
Latency

Time Per Output Token

Throughput




Traditional I/O




GPU direct I/O

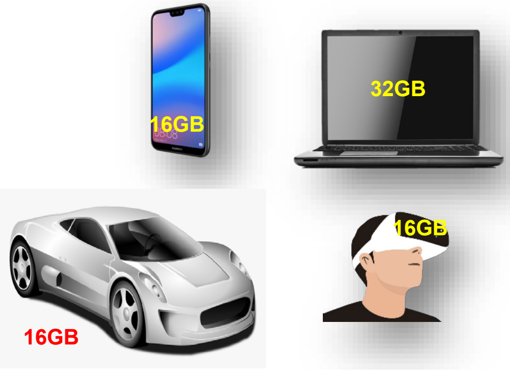
# Inference from Cloud to Edge

## Inference at the Edge

- Improved privacy and security
- Eliminating network latency
- Improved scalability and reduced cost
- Reduced connectivity dependency
- Personalization

## Challenge

- Transformer-based LLMs are significantly larger in size， need too much memory




### Memory vs Storage

| Device | Memory | | | Storage | | |
|---|---|---|---|---|---|---|
| | Size | Bandwidth | Price | Size | Bandwidth | Price |
| Cloud | 1TB | 1.2TB/s | $15,000 | 4*3.84TB | 56GB/s | $1,600 |
| Edge | 64GB | 200GB/s | $1,200 | 4TB | 14GB/s | $400 |
| End | 16GB | 10GB/s | $50 | 1TB | 7GB/s | $100 |

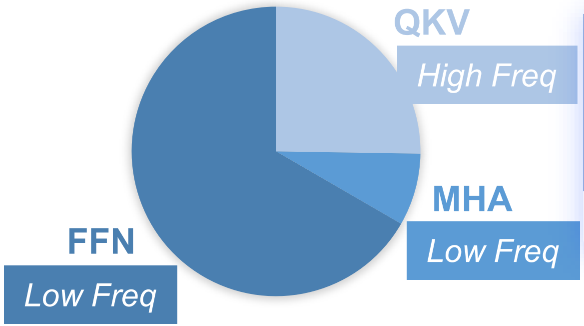
### Minimum memory usage of LLMs inference

| Model | Parameter | Full precision | 16-bit |
|---|---|---|---|
| LLaMA 3 | 8B | 32GB | 16GB |
| ChatGLM | 6B | 24GB | 12GB |
| Qwen | 7B | 28GB | 14GB |
| Mixtral | 8x7B | 188GB | 87GB |

# Inference with limited memory

LLM inference's memory mainly comes from weight storage and KV cache.

## LLM INFERENCE WEIGHT SIZE




## Inference optimization

**Model Quantization**
Converting the model's weights from higher precision to lower precision.

**Model Optimization**
Retrain the model using MQA or GQA, or use flash attention and page attention, which do not require training, to speed up reasoning.

**Model Pruning**
Pruning removes less important weights or neurons in the model.

**Distillation**
Training a smaller model to replicate the behavior of a larger model.

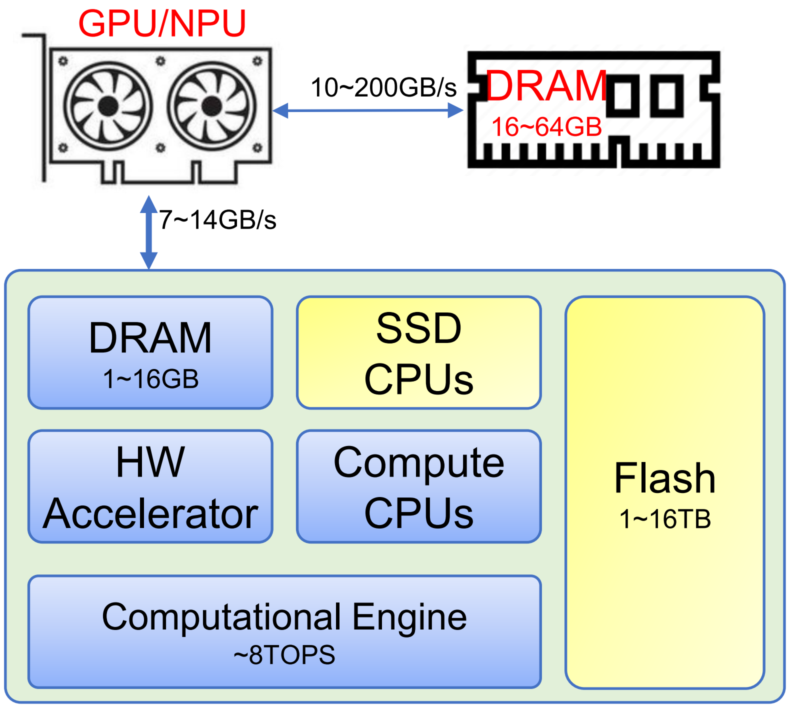
**Memory Optimization Techniques**
Process the model layer-by-layer to keep only the active layer in memory.
Gradient Checkpointing: Save memory by not storing activations for all layers.

**Batch Size Reduction**
Reducing the batch size can lower memory usage
This trade-off can be beneficial if memory is a constraint.

**Sparse Representations**
Leveraging sparsity in model weights can reduce the memory footprint and speed up inference

**Speculative Decoding**
Hard language-modeling tasks often include easier subtasks that can be solved well by more efficient lightweight models

# Inference with Computational Storage




Computational Storage Drive

*High-frequency weights are saved in DRAM, and low-frequency weights are saved in SSD. The weights in SSD are dynamically calculated by CSD to predict the weights needed for inference and loaded into the DRAM of SSD.*

## QKV weight

- [déjà vu] More than 80% of the attention heads become inactive. Choose the head that stands out. Where is the stands out head?
- [FlashAttention] Attention matrix is divided into small blocks and processed block by block.

## FFN weight

- [déjà vu] The sparsity rate of the activation state output by the MLP layer is 95%, that is, more than 95% of the MLP parameters can be excluded from reasoning.
- [LLM in Flash] Use a two-layer MLP to predict which neurons will activate. During inference, dynamically load the parameters corresponding to the predicted activated neurons.

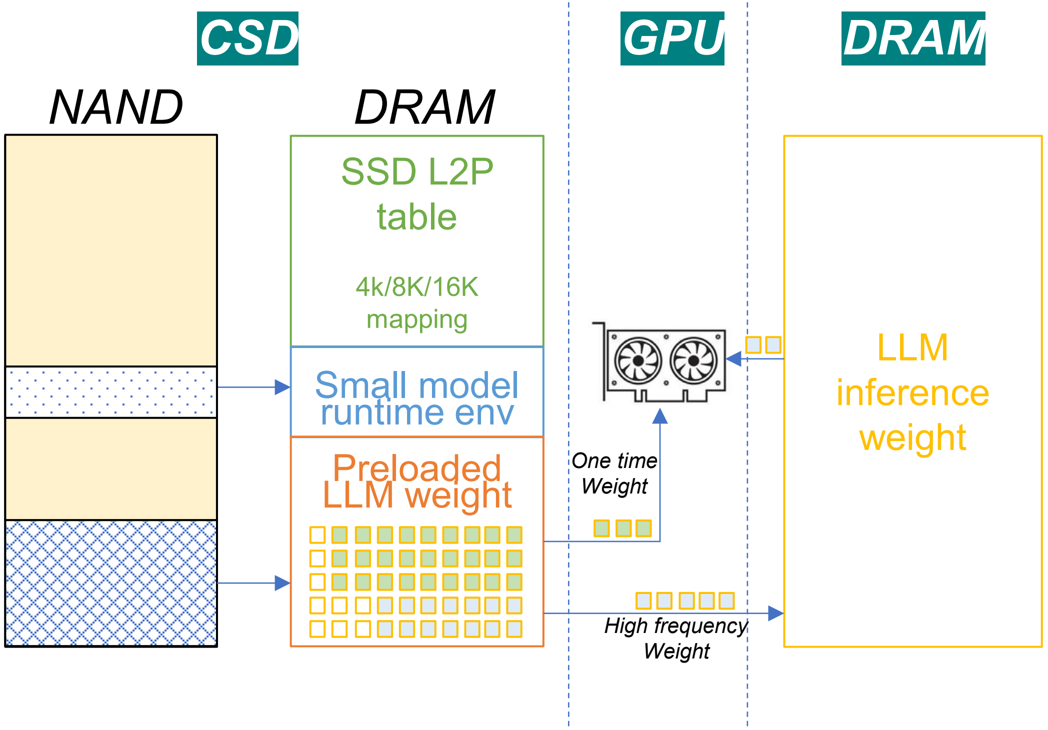
# Inference with Computational Storage

## Train a small model to predict the LLM inference weight access behavior

- Less than ~1/3 high-frequency weight resides in the DRAM of the GPU/NPU
- Other  low-frequency weight exchange between GPU/NPU DRAM and SSD
- 0.1B~0.35B small model predicts the data should be pre loaded from NAND Flash to DRAM in CSD

### Inference DRAM usage @16-bit

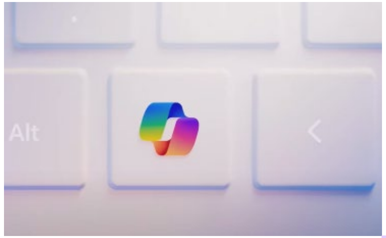
| Model | Parameter | Convention inference | CSD assistant inference |
|---|---|---|---|
| LLaMA 3 | 8B | 16GB | 5GB |
| ChatGLM | 6B | 12GB | 3.5GB |
| Qwen | 7B | 14GB | 4GB |
| Mixtral | 8x7B | 87GB | 24GB |



# Inference in the future

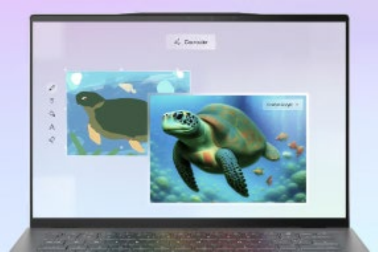
the Future of Memory and Storage

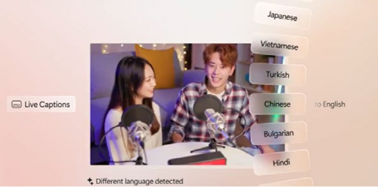
## Application Scenario




*Local Copilot*

*Turn idle to Art*






*Real time subtitles*

## CSD Ecosystem

NVM Express®

**Computational Programs Command Set Specification**

**Revision 1.0**

**December 20th, 2023**

SNIA

Advancing storage & information technology

Computational Storage Architecture and Programming Model

Version 1.0

CXL Compute Express Link

# Thank you.