Memory & I/O Interface Cheat Sheet (Theory Only)

---

1. Memory Hierarchy

Memory Hierarchy → Organizes different memory types based on **speed, cost, and capacity**.
Memory Levels (Fastest to Slowest):

Memory Type	Speed	Cost per Bit	Capacity
Registers	Fastest	Very High	Very Small
Cache Memory	Very Fast	High	Small
Main Memory (RAM)	Fast	Moderate	Medium
Secondary Storage (HDD/SSD)	Slow	Low	Large
Tertiary Storage (Optical, Tape)	Slowest	Very Low	Very Large

Trade-Off: Faster memory → More expensive → Smaller capacity.

---

2. Main Memory (RAM) & Memory Chip Organization

Types of RAM:

- **SRAM (Static RAM):** Fast, used in Cache.
- **DRAM (Dynamic RAM):** Slower, used in Main Memory.

Memory Chip Organization:

- **Memory Addressing:** Each memory cell has a unique address.
- **Word Size:** Number of bits in a memory location (e.g., 32-bit, 64-bit).
- **Memory Bank:** Groups of memory modules accessed simultaneously.

Example: 2D Memory Organization (4×4 matrix for 16-bit memory):

```
Address | Data
----------------
0000    | 1010
0001    | 1100
0010    | 1111
...     | ...
```

---

3. Auxiliary (Secondary) Memory

Used for **permanent data storage** (non-volatile).
Types:

- **Hard Disk Drive (HDD):** Magnetic storage, high capacity.
- **Solid-State Drive (SSD):** Flash memory, faster than HDD.
- **Optical Disks (CD/DVD/Blu-ray):** Store media and backup data.
- **Magnetic Tapes:** Used for large-scale backups.

---

4. Associative (Content-Addressable) Memory

Memory that searches data by content instead of address.
Used in **cache memory, networking (CAM - Content Addressable Memory), and databases.**
Example:

```
Key      | Data
```

```
----------------
"user1"  | 1010
"user2"  | 1100
"admin"  | 1111
```
Searching "admin" directly returns 1111 without scanning entire memory.

---

5. Virtual Memory

Concept: Uses **secondary storage (HDD/SSD) as an extension of RAM**.
Implemented using Paging & Segmentation.
Page Table maps virtual addresses to physical addresses.
Example:

```
Virtual Address  →  Page Table  →  Physical Memory Address
0001 1010        →  1010        →  0110 0101
```

Advantages:

- Expands memory space.
- Allows multiple programs to run efficiently.

Disadvantages:

- Slower than RAM (due to disk access).

---

6. Cache Memory

Small, fast memory placed between CPU and RAM.
Stores frequently accessed data to improve speed.
Mapping Techniques:

- **Direct Mapping:** Each block maps to a fixed cache location.
- **Fully Associative Mapping:** Any block can go into any cache location.
- **Set Associative Mapping:** Combines Direct & Fully Associative Mapping.

Cache Performance Metrics:

- **Hit Ratio:** % of memory accesses found in cache.
- **Miss Penalty:** Time taken to fetch data from RAM when cache miss occurs.

---

7. Input-Output (I/O) Interface

Allows CPU to communicate with external devices (Keyboard, Monitor, Storage).
Components of I/O Interface:

- **Data Bus:** Transfers data.
- **Address Bus:** Identifies device.
- **Control Signals:** Start/Stop data transfer.

Types of I/O Communication:

- **Memory-Mapped I/O:** Uses memory addresses for I/O devices.
- **Port-Mapped I/O:** Uses separate address space for I/O.

---

8. Asynchronous Data Transfer

Used when CPU & I/O devices operate at different speeds.
Synchronization Methods:

- **Handshake Protocol:** Two-way signaling between sender & receiver.
- **Strobe Signal:** One-way signal to indicate data is ready.

---

9. Modes of Data Transfer

Mode	Description
Programmed I/O	CPU manually handles I/O data transfer (slow).
Interrupt-Driven I/O	CPU is notified only when I/O is ready (better efficiency).
Direct Memory Access (DMA)	Data is transferred directly from I/O to memory without CPU intervention (fastest).

---

10. Priority Interrupt

Interrupt Handling when multiple interrupts occur.

Methods:

- **Polling:** CPU checks each device sequentially.
- **Vectored Interrupt:** Device sends a unique interrupt code to CPU.
- **Daisy Chaining:** Devices connected in series, priority based on position.

---

11. Direct Memory Access (DMA)

Allows I/O devices to directly access main memory without CPU involvement.

Components:

- **DMA Controller (DMAC):** Manages data transfer.
- **Bus Arbitration:** Determines control of system bus.

Steps in DMA Transfer:

1st CPU requests DMA to transfer data.
2nd DMA takes control of memory bus.
3rd DMA transfers data between I/O and memory.
4th DMA notifies CPU when complete.

---

12. Bus Interface Circuits & Standard I/O Interfaces

Buses: Connect different computer components for data transfer.

- **Data Bus (Carries Data).**
- **Address Bus (Identifies Device).**
- **Control Bus (Manages Communication).**

Standard I/O Interfaces:

Interface	Description
PCI (Peripheral Component Interconnect)	High-speed bus for internal hardware (Graphics Card, Sound Card).
SCSI (Small Computer System Interface)	Used for high-speed disk drives and tape storage.
USB (Universal Serial Bus)	Common interface for external devices (Keyboard, Mouse, Flash Drives).

Comparison of Interfaces:

Feature	PCI	SCSI	USB
Speed	High	Very High	Moderate
Devices	Internal	Storage Devices	External
Complexity	Moderate	High	Low

---

13. Case Study – Advanced Processors

Modern Processors Implement:

- **Multi-Core Architectures (Parallel Processing).**
- **Out-of-Order Execution (Improves Efficiency).**
- **Advanced Branch Prediction (Reduces Pipeline Stalls).**
- **High-Speed Cache Memory (L1, L2, L3).**
- **Hyper-Threading (Intel) & Simultaneous Multithreading (AMD).**

Examples:

- **Intel Core i9, AMD Ryzen 9** (Multi-Core, Hyper-Threading).
- **ARM Processors** (Used in smartphones, energy-efficient).
- **RISC-V** (Open-source architecture).

Key Takeaways

Memory Hierarchy: Fastest at the top (Registers) → Slowest at the bottom (HDD).
Virtual Memory: Expands memory using the disk (Paging, Segmentation).
Cache Memory: Stores frequently used data for quick access.
I/O Interfaces: PCI, SCSI, USB handle communication between devices.
DMA: Direct data transfer without CPU overhead.
Modern Processors: Use **Multi-Core, Hyper-Threading, and High-Speed Cache.**

This **Memory & I/O Cheat Sheet** covers **memory hierarchy, cache, I/O interfaces, DMA, and modern processors**. Let me know if you need further explanations!