CHAPTER 1 – INTRODUCTION & THEORY

1.1 Overview of Memory in VLSI

In any digital system, **memory** is used to store data and instructions for processing.
In VLSI design, memory is implemented either **on-chip** (inside the processor or SoC) or **off-chip** (external).
There are two main categories:

- **Volatile memory** – loses data when power is turned off (e.g., RAM)
- **Non-volatile memory** – retains data even without power (e.g., ROM)

1.2 What is RAM & ROM?

- **RAM (Random Access Memory)**:
  Stores data temporarily and allows both **read** and **write** operations.
  Examples: SRAM, DRAM.
- **ROM (Read-Only Memory)**:
  Stores data permanently and generally allows **read-only** access.
  Examples: Mask ROM, PROM, EEPROM.

1.3 Single Port vs. Dual Port Memory

- **Single Port Memory**:
  Only one set of address/data lines, so either a read or write can occur at one time.
- **Dual Port Memory**:
  Has two independent sets of ports (Port A and Port B), allowing:
  - Simultaneous read & write
  - Read from both ports at the same time
  - Write from one port and read from the other

Advantages of Dual Port Memory:

- Higher speed due to parallel access
- Useful in multi-processor or multi-core systems
- Reduces data access bottlenecks

1.4 Synchronous vs. Asynchronous Operation

- **Synchronous Memory**:
  - All operations are triggered by a **clock signal**.
  - Data read/write happens at the **rising or falling edge** of the clock.
  - Example: Synchronous Dual Port RAM in FPGAs.
  - Pros: Predictable timing, easy to integrate with synchronous systems.
  - Cons: Slightly slower for random access compared to asynchronous.
- **Asynchronous Memory**:
  - Operations happen immediately based on control signals, without waiting for a clock.
  - Faster for certain applications (like direct data lookup).

    - Pros: Lower latency in some designs.
    - Cons: Harder to synchronize in multi-clock systems.

1.5 Applications in VLSI Systems

- **Dual Port RAM**:
    - Buffering in communication systems
    - Video/image processing
    - Shared memory between processors
- **Dual Port ROM**:
    - Lookup tables for DSP functions
    - Microcode storage in CPUs
    - Predefined data for graphics/fonts

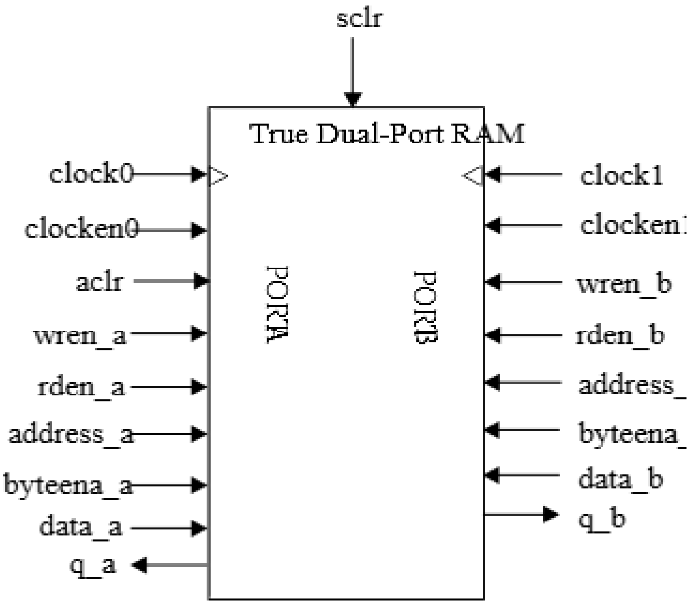
1.6 Project Objective

The objective of this project is to:

- **Design** and **simulate** a Dual Port RAM and Dual Port ROM in both **synchronous** and **asynchronous** modes.
- Demonstrate:
    - Independent access from two ports
    - Read/write capability in RAM
    - Read-only capability in ROM
- Implement using **Verilog HDL**
- Verify the design with simulation waveforms
- Compare performance between synchronous and asynchronous operations.

CHAPTER 2 – DESIGN & IMPLEMENTATION

2.1 Dual Port RAM – Architecture & Features



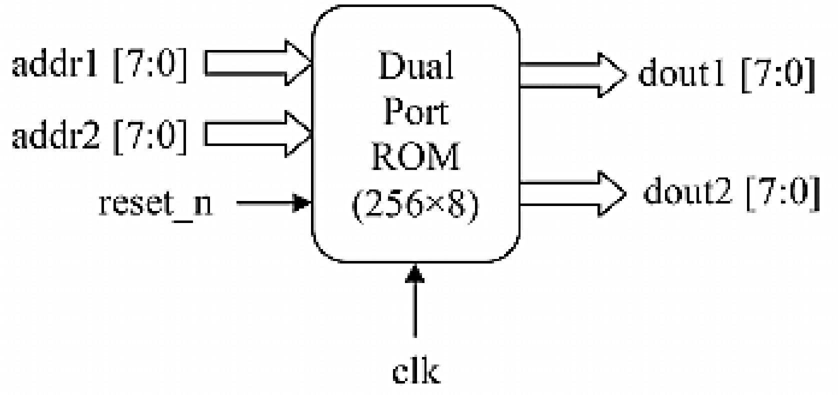
A **Dual Port RAM** allows two independent accesses through **Port A** and **Port B**.
Each port has its own:

- Address lines
- Data lines
- Control signals (Read/Write enable, Chip enable)
- Clock (for synchronous version)

Key Features:

- **Simultaneous Read/Write**: One port can write while the other reads.
- **Independent Addressing**: Each port can access different memory locations.
- **Conflict Handling**: If both ports access the same address for writing, priority logic is required.

2.2 Dual Port ROM – Architecture & Features



A **Dual Port ROM** is similar in structure but:

- Only supports **read** operation from both ports.
- The memory content is **pre-loaded** during design (hard-coded in Verilog or initialized from a file).
- Ideal for lookup tables, coefficients, and constant data storage.

2.3 Read/Write Operations in Synchronous Mode

In **synchronous dual port memory**:

- All operations are triggered by the **clock edge**.
- Data becomes available **after the clock event**.

Table: Synchronous Dual Port RAM Operation

Clock Edge	Port	WE (Write Enable)	RE (Read Enable)	Operation
Rising	A	1	0	Write to Port A
Rising	A	0	1	Read from Port A
Rising	B	1	0	Write to Port B
Rising	B	0	1	Read from Port B

2.4 Read/Write Operations in Asynchronous Mode

In **asynchronous dual port memory**:

- Operation starts immediately after control signals change.
- No clock required.

Table: Asynchronous Dual Port RAM Operation

Control Signals	Operation
WE=1, RE=0	Write immediately
WE=0, RE=1	Read immediately
WE=0, RE=0	Idle
WE=1, RE=1	Invalid (Conflict)

2.5 Block Diagrams

(a) Dual Port RAM – Synchronous Mode

- Two sets of address, data, and control lines
- Shared memory array
- Two clock inputs
- Write/read controlled by clock edges

(b) Dual Port ROM – Asynchronous Mode

- Two address inputs
- Shared memory array with fixed content
- Read data available as soon as address changes

2.6 Simulation Setup & Testbench Explanation

- **Testbench** generates clock (for synchronous designs).
- Applies:
  - Write operations to one port while reading from another.
  - Same address conflict checks.
  - Asynchronous reads for ROM.
- Outputs are verified using **ModelSim/Questa/GTKWave**.

2.7 Synthesis Considerations

- For **FPGA**, synthesis tools map RAM/ROM to **Block RAMs (BRAM)**.
- Dual clock support requires FPGA BRAM features.
- For **ASIC**, memory is mapped using standard cell memory generators.

CHAPTER 3 – RESULTS & CONCLUSION

3.1 Simulation Waveforms

- **Synchronous Dual Port RAM**:
    - Waveforms show **write** on Port A and **read** on Port B happening at the same clock edge.
    - Data appears **one clock cycle later** after read command (due to synchronous nature).
- **Asynchronous Dual Port ROM**:
    - Waveforms show data output changes **immediately** after address change.
    - No clock signal is present.

3.2 Output Analysis Table

Case	Port A Operation	Port B Operation	Address Relation	Expected Output Behaviour
1	Write	Read	Same Address	Port B reads newly written data (sync delay if synchronous)
2	Read	Read	Different Addr	Both outputs independent
3	Write	Write	Different Addr	Both writes occur successfully
4	Write	Write	Same Address	Conflict – Priority logic decides winner
5	Read (ROM)	Read (ROM)	Any	Both outputs stable immediately after address change

3.3 Comparison: Synchronous vs. Asynchronous

Feature	Synchronous	Asynchronous
Clock	Required	Not required
Timing Control	Predictable & easy to design	Requires careful control
Access Speed	Slightly delayed (1 cycle)	Immediate
Integration	Easy in synchronous SoCs	Needs synchronization logic

3.4 Key Observations

- Dual Port RAM allows **parallelism** — one port writing while another reads.
- Synchronous operation ensures **predictable timing**, suitable for processor-based systems.
- Asynchronous ROM provides **instantaneous data**, useful for look-up tables and constants.
- Conflict handling is crucial when both ports access the same location for writing.

3.5 Conclusion & Future Scope

In this project, **Dual Port RAM** and **Dual Port ROM** were successfully designed, simulated, and verified in both **synchronous** and **asynchronous** modes using Verilog HDL.
The designs demonstrated:

- Simultaneous dual-port access
- Reliable synchronous operation with clock control

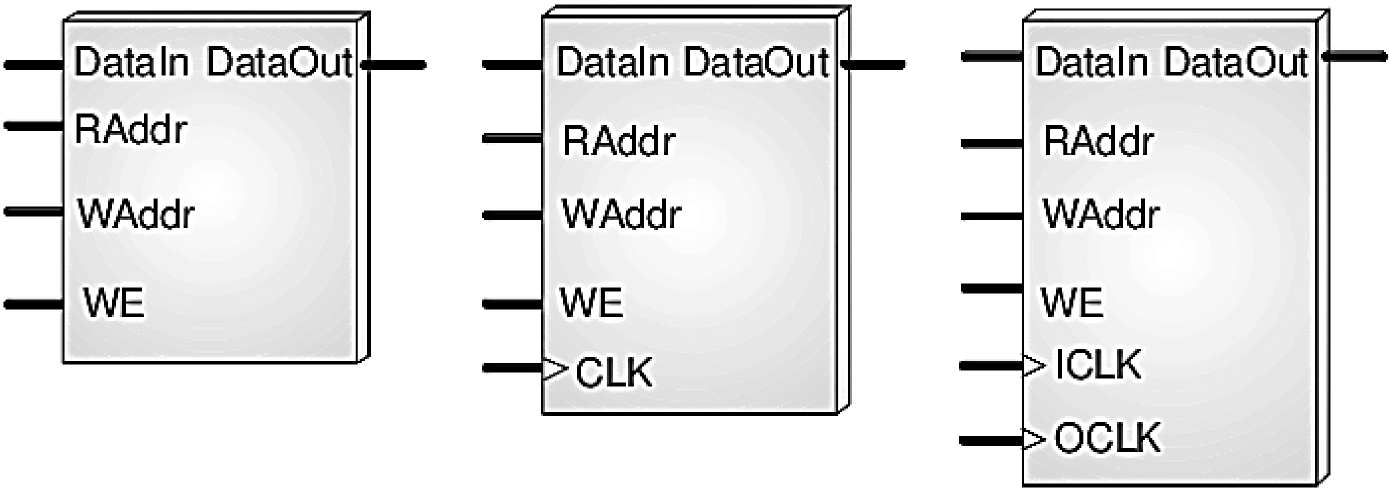
- Fast asynchronous read access for ROM

Future Scope:

- Add **error detection** (ECC – Error Correcting Codes).
- Extend to **true dual clock** RAM for different clock domains.
- Optimize for **low power** in ASIC design.

Block Diagrams (IEEE Style)

Dual Port RAM – Synchronous Mode / Dual Port ROM – Asynchronous Mode



dual-port RAM
asynchronous

dual-port RAM
asynchronous read
synchronous write

dual-port RAM
synchronous

REFERENCES

[1] **Design & Verification of Dual Port RAM Using UVM** – BIST + UVM testbench methodology. Link

[2] **Xilinx Vivado UG901 (2025)** – Verilog coding for dual-port RAM (single & dual clock). Link

[3] **Dual Port ROM-Based Multiplier on FPGA (2021)** – ROM use in DSP multiplier design. Link

[4] **Configurable Multi-Port Memory Architecture (2024)** – Scalable high-speed memory design. Link