

# Memory Controller for DDR Memories

Pranav M – PES2UG23EC076

Lalith – PES2UG23EC074

Harshini – PES2UG23EC058

# Introduction and Problem Statement

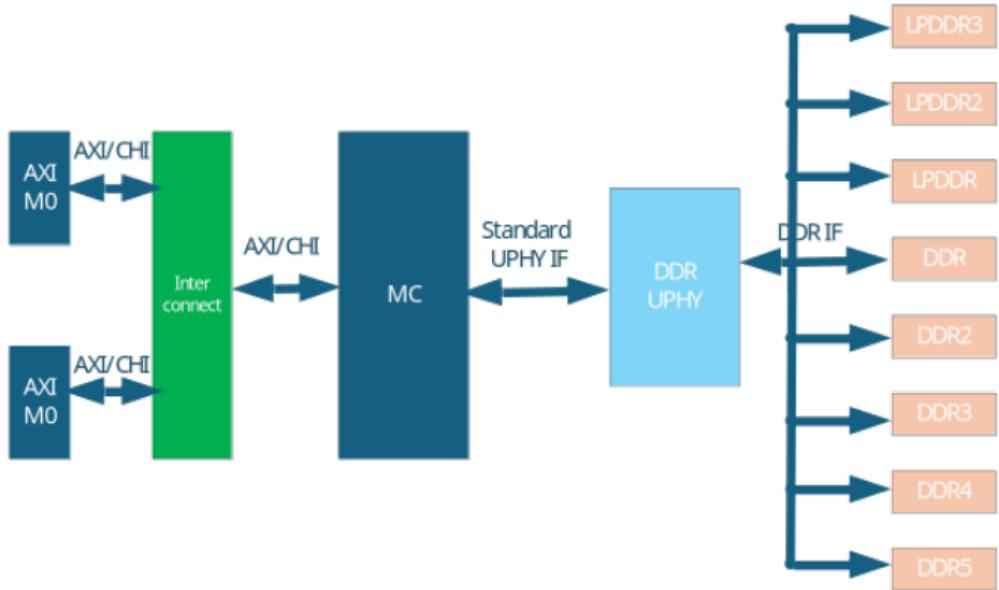
## What is a Memory Controller?

- Bridge between processor/SoC and DRAM
- Manages read/write transactions, timing, refresh
- Critical for system performance and power

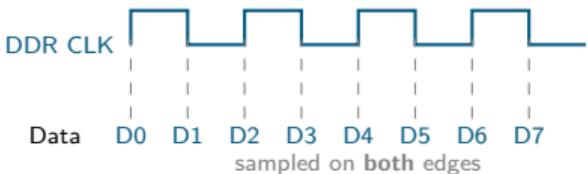
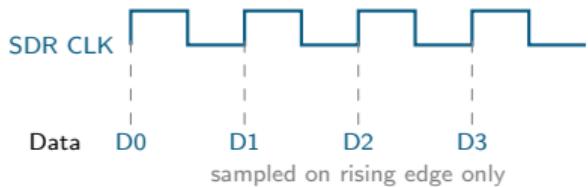
## Our Goal:

Design and implement a reconfigurable memory controller capable of supporting multiple DDR generations.

# System Block Diagram



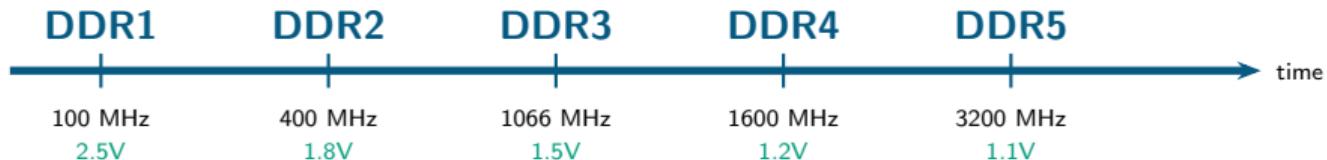
# SDR vs DDR – What Changed?



## Key Differences

	SDR	DDR
Transfer	Rising edge	Both edges
Bandwidth	1×	2×
Voltage	3.3V	2.5V / lower
Prefetch	1n	2n

## DDR Generations – Evolution



### Each generation brought:

- Higher clock frequencies and bandwidth
- Lower operating voltage (power savings)
- Improved prefetch depth ( $2n \rightarrow 4n \rightarrow 8n \rightarrow 16n$ )
- Key challenges as speeds increase: signal integrity, timing closure, power delivery, and clock domain crossing

# LPDDR – Low Power DDR

## Requirements

- High bandwidth
- Low latency
- Power efficient

## Advanced Power Management

- Partial Array Self Refresh (PASR)
- Temperature Compensated Self Refresh (TCSR)
- Deep Power-Down (DPD)

## LPDDR vs DDR

	DDR5	LPDDR5
Voltage	1.1V	1.05V
Form factor	DIMM	PoP/BGA
Power modes	Basic	PASR/TCSR/DPD

# Asynchronous FIFO – Functionality

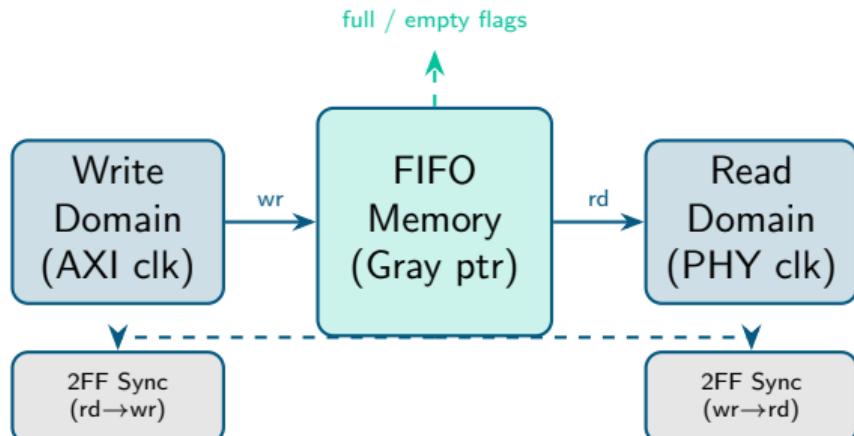
## The Problem: Clock Domain Crossing (CDC)

- AXI clock  $\neq$  PHY/memory clock
- Direct signal passing causes metastability
- Async FIFO safely transfers data between domains

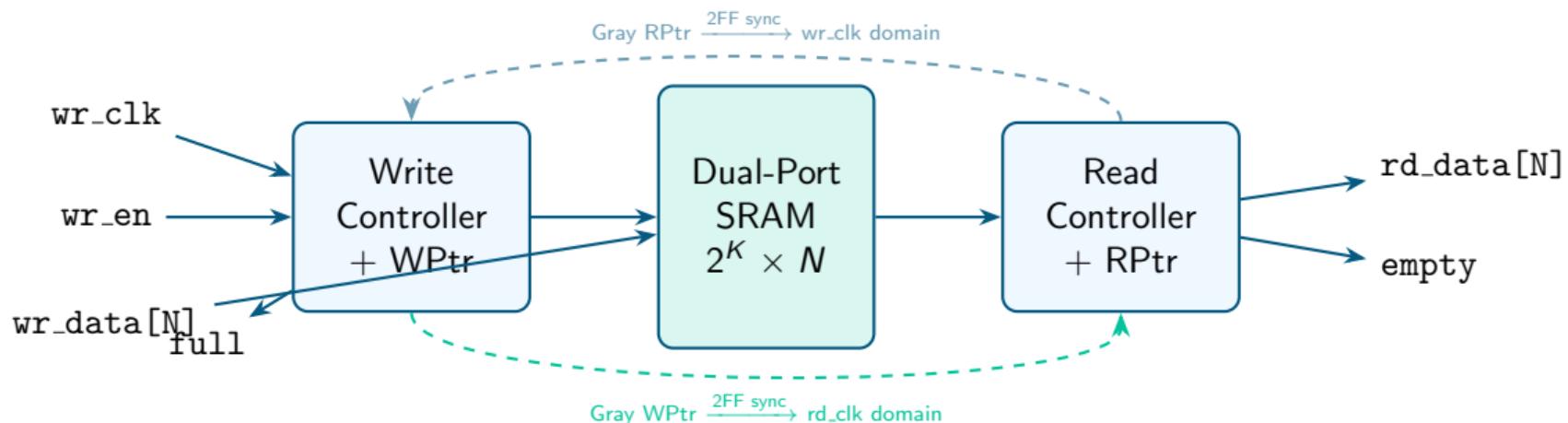
### Key signals:

- wr\_clk, wr\_en, wr\_data
- rd\_clk, rd\_en, rd\_data
- full, empty flags

Gray-coded pointers used for safe CDC of read/write pointers



# Asynchronous FIFO – Block Diagram



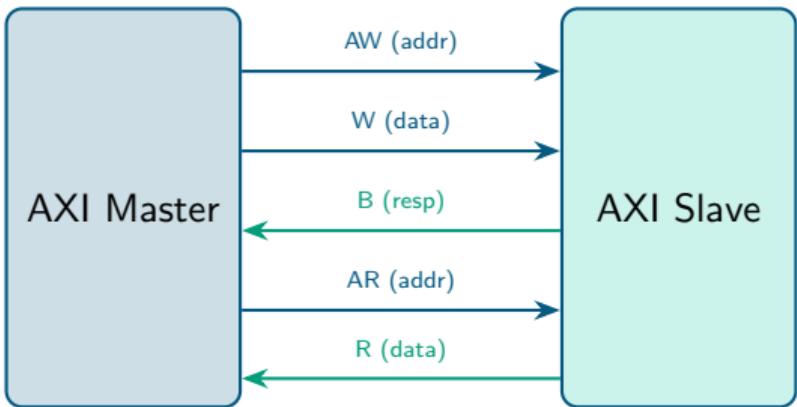
# AXI Interface – Overview

## Why AXI?

- Industry standard (ARM AMBA)
- Separate read and write channels
- Burst transfers, out-of-order support
- Decouples master from slave

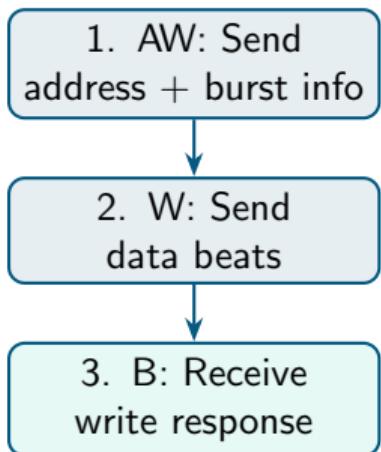
## 5 Channels:

- AW – Write Address
- W – Write Data
- B – Write Response
- AR – Read Address
- R – Read Data

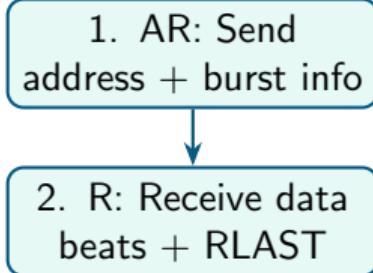


# AXI – Write and Read Transaction Flow

## Write Transaction



## Read Transaction

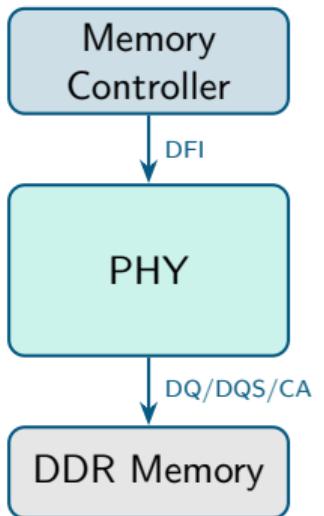


**Handshake:** Every channel uses VALID/READY handshake – transfer occurs only when both are high.

# PHY Layer – Physical Interface

## Role of PHY

- Bridge between the memory controller and the memory
- Temporarily stores instructions and data in buffers
- Performs Write Leveling, DQS Gate Training and Vref Training



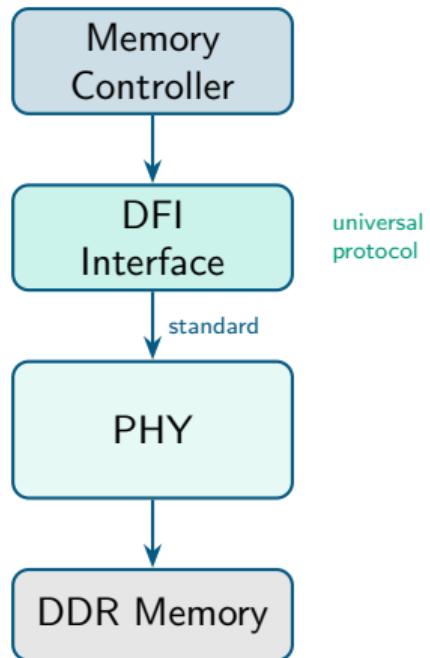
# DFI Interface

## What is DFI?

- Acts as a universal language for PHY interoperability
- Decouples the memory controller from PHY implementation

## Three Pillars of DFI 5.0/5.1

- PHY independent boot sequence
- Expanding frequency range support
- New PHY-to-Controller Interface interaction



## Next Steps

