AXI4-Lite GPIO Peripheral and DDR Memory Controller

Learning Outcomes:

At the end of this module, you will be able to:

- Outline AXI4-Lite features and usage of low-power interface.
- Describe functionality and properties of General-Purpose Input and Output (GPIO).
- Explain the structure and operation of memory access and addressing.
- Identify the properties of various types of memory used in a typical SoC including volatile and non-volatile memory.
- Describe the structure and read/write operations for SRAM and DRAM.
- Describe the role and function of a Memory Controller.

AMBA AXI4-Lite

- AXI4-Lite:
  - Suitable for simpler control interfaces, register-style
  - Light version of full AXI4
- Key features:
  - All transactions are of burst length one
  - All data accesses are based on full-width data bus (AXI4-Lite supports a data bus of 32-bit width or 64-bit width)
  - All accesses are non-modifiable, Non-bufferable
  - No support of exclusive accesses

AXI4-Lite Signals

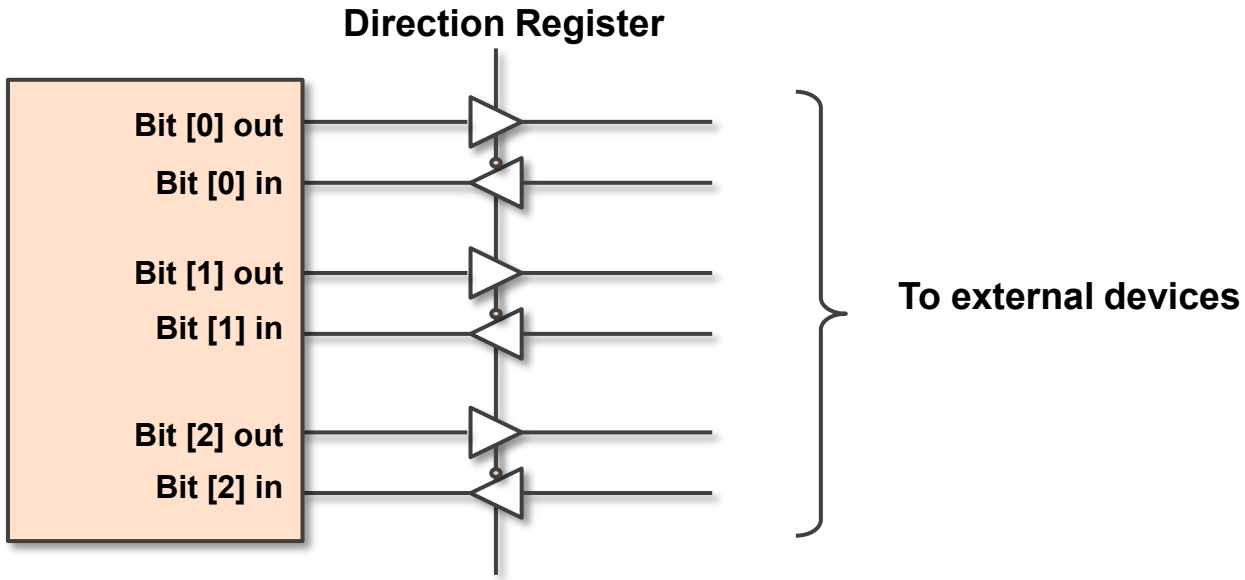
Global	Read Address	Read Data	Write Address	Write Data	Write Response
ACLK	ARVALID	RVALID	AWVALID	WVALID	BVALID
ARESETn	ARREADY	RREADY	AWREADY	WREADY	BREADY
	ARADDR	RDATA	AWADDR	WDATA	BRESP
	ARPORT	RRESP	AWPORT	WSTRB	

AXI low-power interface

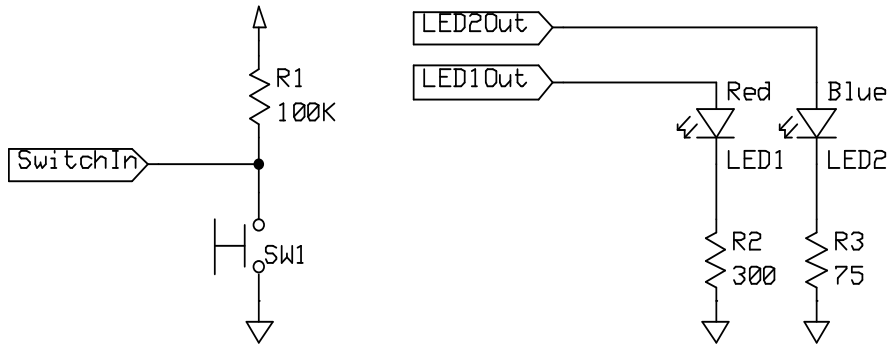
- AXI low-power interface provides control during entry into and exit from a low-power state.
- It is an optional extension to the AXI protocol that targets
  - Any peripheral without powerdown sequence that can indicate when its clocks can be turned off
  - Any peripheral that requires a powerdown sequence and that can have its clocks turned off only after it enters a low-power state
- Low-power clock control
  - Low-power clock control interface signals:
    - One signal from the peripheral indicating when its clocks can be enabled or disabled
    - Two handshake signals for the system clock controller to request exit from or entry into a low-power state
  - A number of different peripherals can be combined into the same low-power clock domain, to treat that clock domain like a single peripheral.

GPIO Overview

- General-purpose input/output (GPIO)
  - Used for general purpose, no special usage defined
  - Widely used for most of the applications
  - The direction of input/output is controlled by the direction register.
  - A mask register is often used to mask out certain bits.



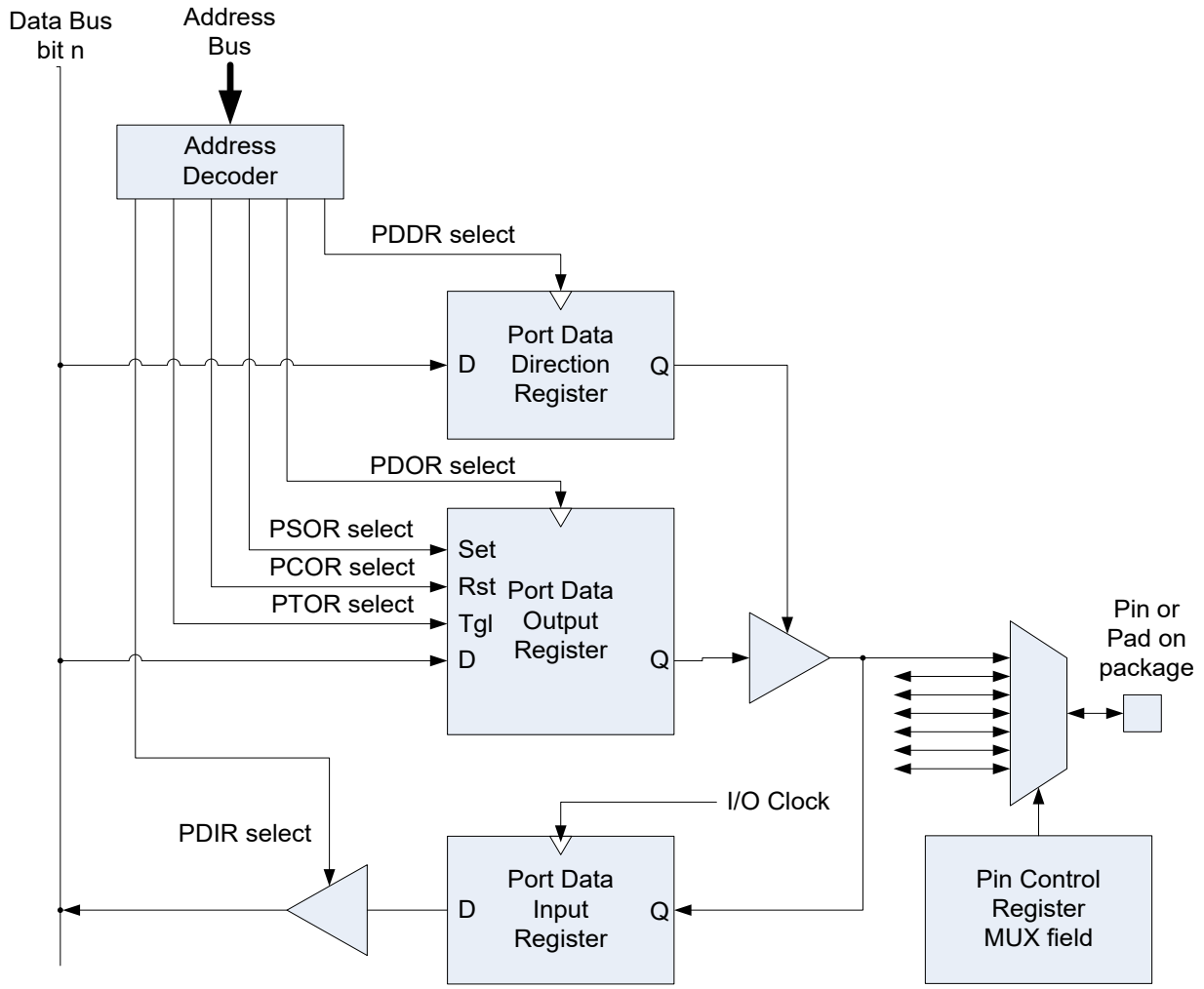
Basic Concepts



- Goal: light either LED1 or LED2 based on switch SW1 position
- GPIO
  - Input: program can determine if input signal is a 1 or a 0
  - Output: program can set output to 1 or 0
- Can use this to interface with external devices
  - Input: switch
  - Output: LEDs

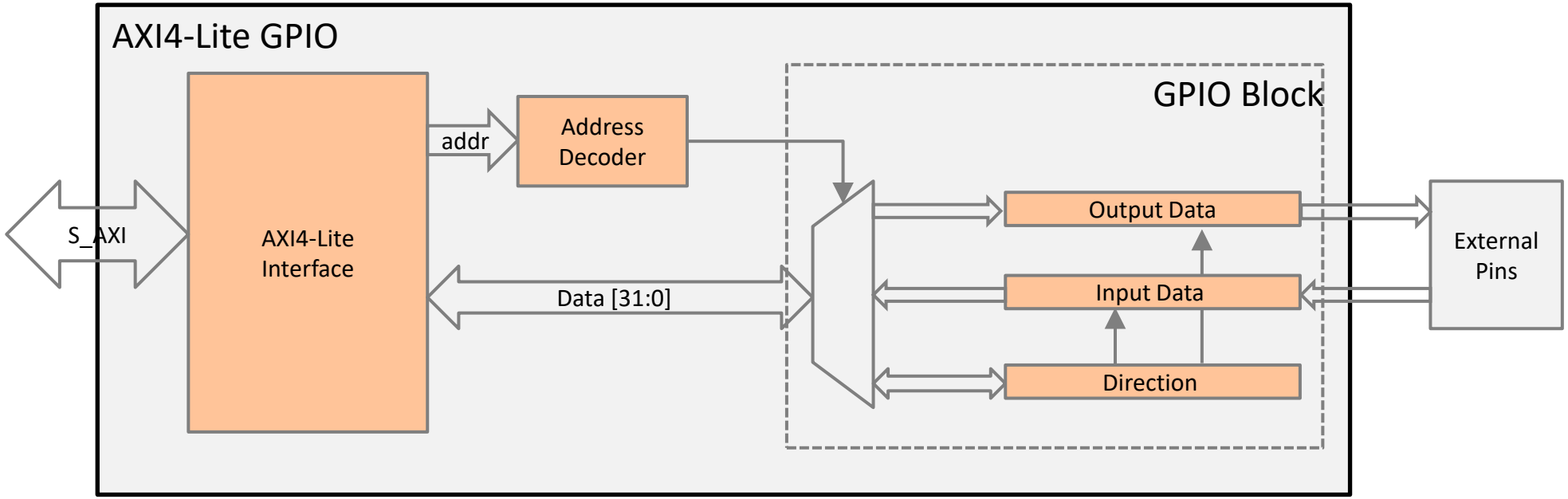
Example GPIO Port Bit Circuitry in MCU

- Control
  - Direction
  - MUX
- Data
  - Output (different ways to access it)
  - Input



AXI4-Lite GPIO

- General-purpose input/output (GPIO) with AXI4-Lite interface
  - Used to transmit data between the AXI bus and the GPIO block
  - 32-bit data transmission with write strobes

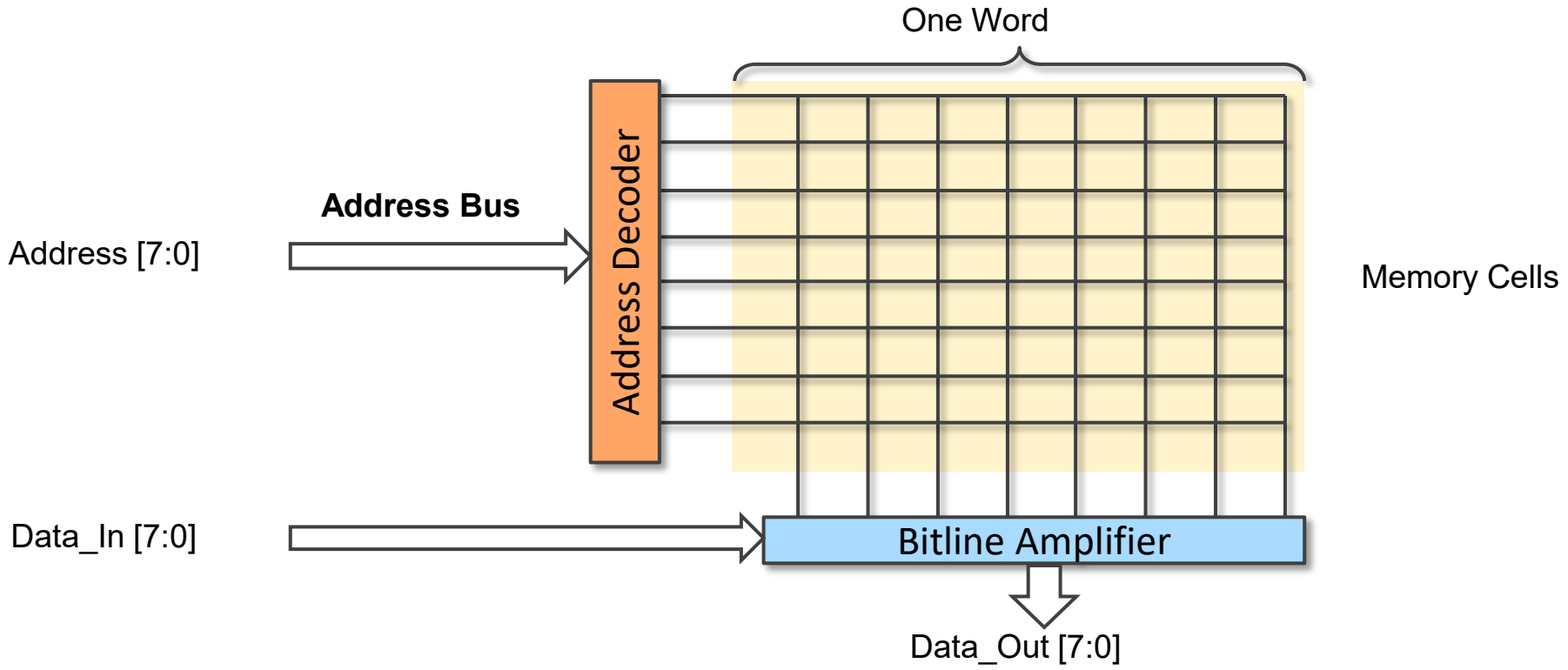


Computer Memory

- Computer memory: a physical device that is used to store program code or data of a processor on either a temporary or permanent basis
- We can distinguish between two types of memory:
  - Volatile memory
    - Requires power to keep the stored data
  - Non-volatile memory
    - Can retain stored data after powerdown

Memory Access

- A memory is accessed by presenting it with an address, then writing or reading the data at that address.
- For example, a memory architecture with 8-bit width and 8-bit data is shown below:



Memory Accessing

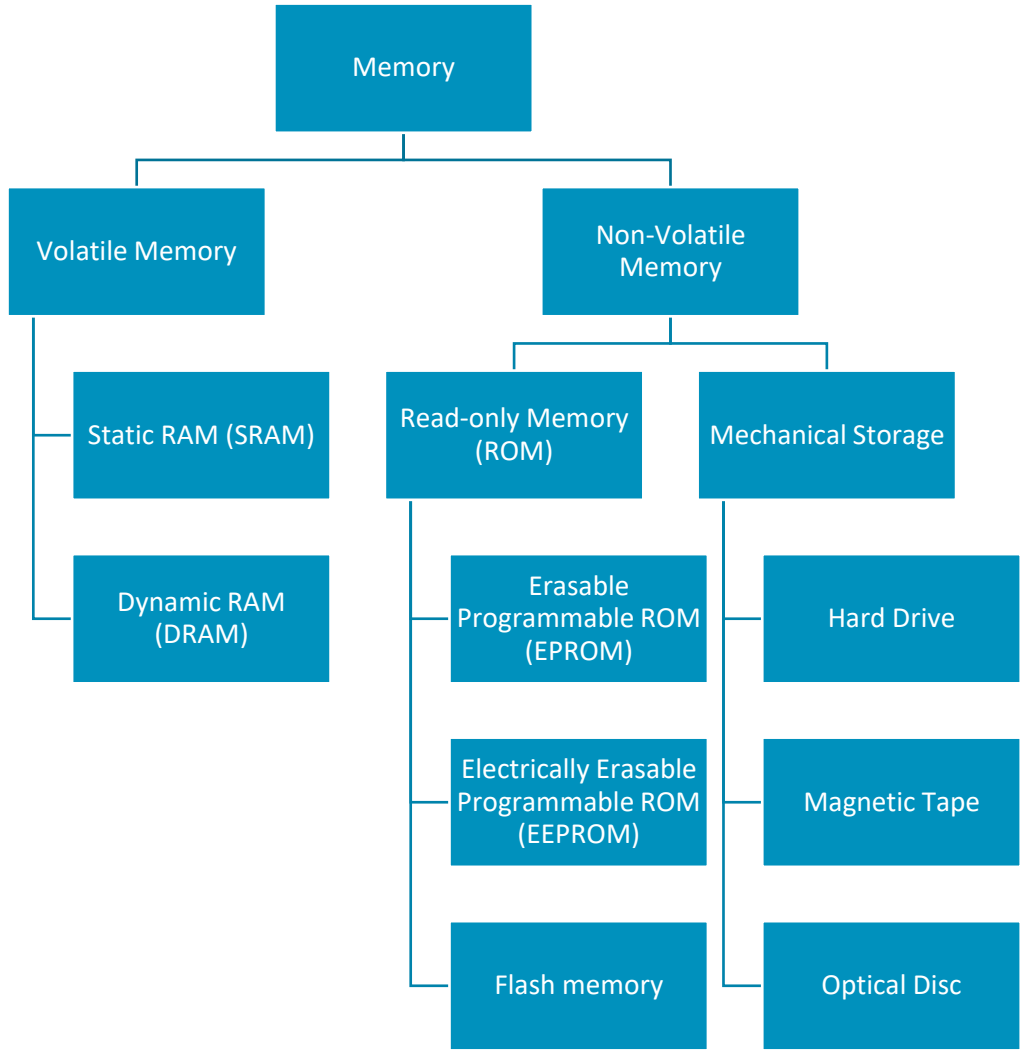
- Present an address:
  - Provide an address to the memory address bus
  - A particular word is then selected by the address decoder and connected to the bitline amplifier.
  - For larger memory, the address may be divided into row and column sections.
- Read operation:
  - The selected data is connected to the bitline amplifier.
  - The amplifier restores the signal to the proper voltage level, then outputs it to the Data_Out port.
- Write operation:
  - Present the data to the Data_In port
  - The amplifier sets the bitlines to the desired values and drives that value from the bitline to the memory cell.

Volatile v Non-volatile Memory

- Volatile memory
  - Requires power to retain the data information
  - Usually faster access speed and less costly
  - Used for temporary data storage such as CPU cache, internal memory
  - Also known as random access memory (RAM)
- Non-volatile memory
  - No power is required to retain the data information
  - Usually slower access speed and more costly
  - Used for secondary storage or long-term persistent storage

Types of Memory

- Volatile memory
  - Static RAM (SRAM)
  - Dynamic RAM (DRAM)
- Non-volatile memory
  - Read-only memory (ROM)
  - Erasable programmable ROM (EPROM)
  - Electrically erasable programmable ROM (EEPROM)
  - Non-volatile random access memory (NVRAM)
  - Flash memory
  - Mechanical storage
  - Hard drive, magnetic tape

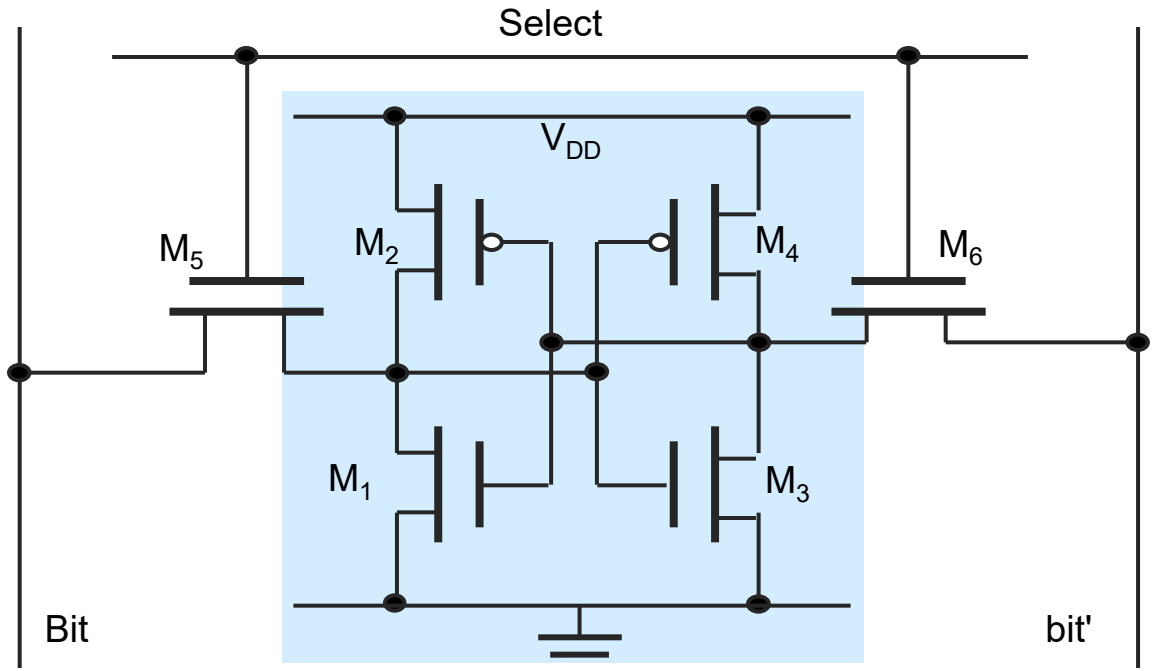


Static RAM

- Static RAM (SRAM)
  - Volatile memory
  - Data is retained as long as power is supplied
  - Usually uses six transistors to store one-bit data
  - Fast data access
  - Larger power
  - Less density, larger block size
  - More expensive

Static RAM Cell

- An SRAM cell is typically made up of six MOSFET transistors.
  - A single bit is stored on four transistors (M1-M4), which form two inverters that are cross-coupled.
  - The access of the bit is controlled by two access transistors (M5 and M6) that are gated by the wordline (select).



Accessing Static RAM

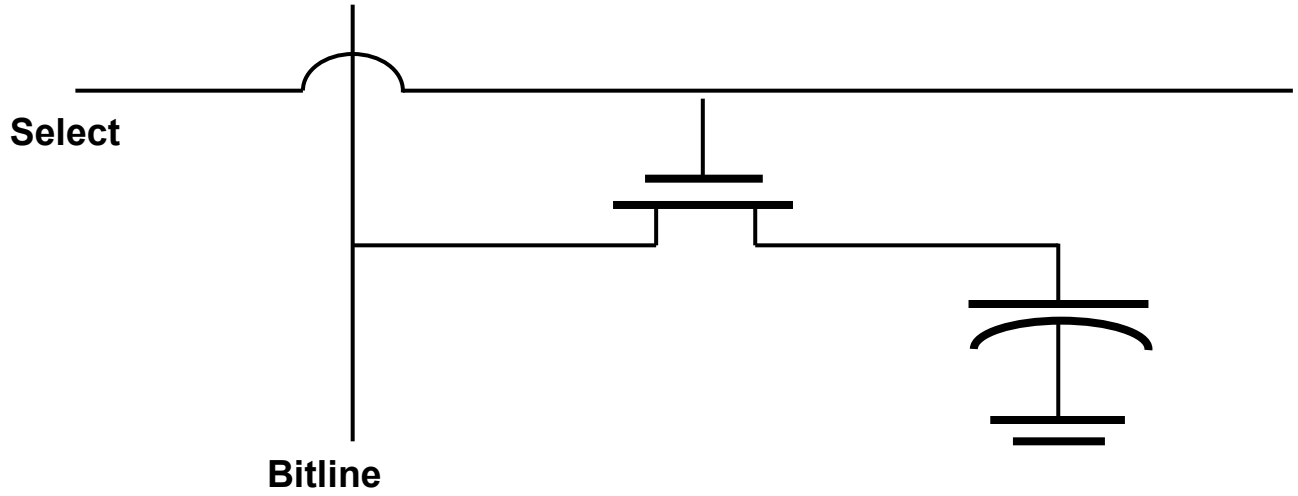
- Read operation
  - The address is decoded and the desired cell is then selected, in which case the select line is set to one.
  - Depending on the value of the four transistors (M1-M4), one of the bitlines will be charged to 1 and the other will be drained to 0.
  - The states of the two bitlines (bit and bit') are then read out as 1-bit data.
- Write operation
  - The two bitlines (bit and bit') are pre-charged to the desired value (e.g., bit = VDD, bit' = VSS).
  - The address is decoded and the desired cell is then selected, in which case, the select line is set to one.
  - The four transistors (M1-M4) are then forced to flip their states (either charged or discharged), since the bitlines normally have much higher capacitance than the four transistors.

Dynamic RAM

- Dynamic RAM (DRAM)
  - One bit of data can be stored in one transistor and capacitor pair; the status of the capacitor (charged or uncharged) indicates the bit state (1 or 0).
  - Needs to be refreshed (or recharged) periodically, since the capacitor leaks its charge, e.g., every 10 ms.
  - Higher density, smaller block size
  - Less expensive
  - DRAM can be categorized according to its data rate and synchronization mode, for example:
    - Single data rate (SDR) and double data rate (DDR)
    - Synchronous DRAM (SDRAM) and non-synchronous DRAM

Dynamic RAM

- In DRAM, each memory cell requires fewer transistors, e.g., a three-transistor cell or even one-transistor cell.
- For example, the one-transistor cell is composed by one transistor and one capacitor.
  - One-transistor: a gate transistor used to select a single cell
  - One-capacitor: stores the value of a single bit



Accessing Dynamic RAM

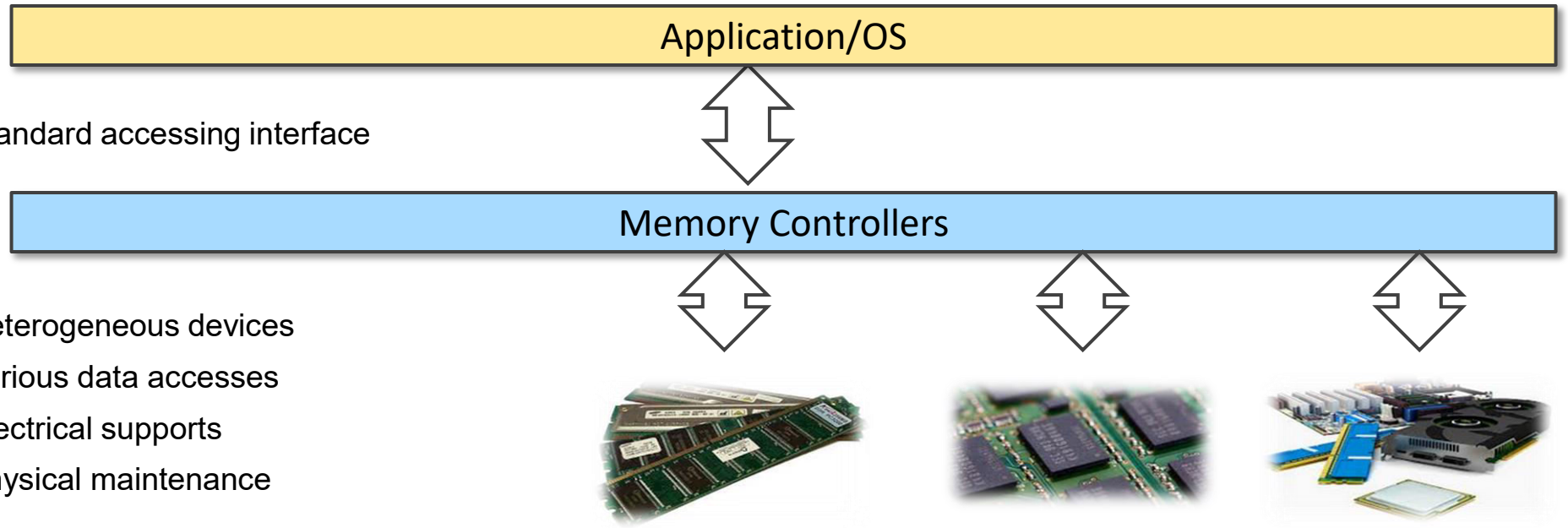
- Read operation
  - The address decoder decodes the address and sets the select line to one
  - The bitline is then changed according to the state of the capacitor
- Write operation
  - The single bitline is pre-charged to a desired value (e.g., VDD or VSS)
  - The address decoder decodes the address and sets the select line to one
  - The capacitor is then either charged or discharged by the bitline

Non-volatile Memory

- Read-only memory (ROM)
  - In early times, ROM was manufactured with the desired data, which could not be changed.
  - Later types allowed data to be reprogramed, but with a degree of effort.
    - Erasable programmable ROM (EPROM)
    - Electrically erasable programmable ROM (EEPROM)
- Non-volatile random access memory (NVRAM)
  - Radom access, data can be both read and written
  - Best-know form is flash memory
- Mechanical storage
  - Non-electrically addressed memories, e.g., hard drives, magnetic tapes, optical drives
  - Less expensive, but slower

Memory Controller

- A piece of hardware that is mainly used for controlling the data flow going to/from the memory block.
- Memory controllers facilitate access to heterogeneous physical devices, e.g., SRAM, DRAM, FLASH, hard disk.



The Roles of a Memory Controller

- The role of a memory controller includes:
  - Interfacing with a particular type of memory block
  - Facilitating memory access by providing a universal interface to the system, e.g., to a standard bus interface
  - Supporting a variety of memory access modes, such as burst mode, memory paging, etc
  - May provide electrical support for the memory, e.g., refreshing a DRAM

Signal Description

Name	I/O	Description
Data [15:0]	I/O	16-bit data input/output bus
Addr [25:0]	I	26-bit address bus
RamCS	I	Chip select; activates the device when LOW
MemOE	I	Output enable: enables the output buffers when LOW
MemWR	I	Write enable: writes to the memory when LOW
RamLB	I	Lower byte enable
RamUB	I	Upper byte enable
RamClk	I	Additional clock input used to synchronize with the system; can be set to LOW as it is not needed in this teaching material
RamAdv	I	Address valid: indicates that a valid address is given on to the address bus; can be set to LOW as it is not needed in this teaching material
RamCre	I	Control register enable; can be set to LOW as it is not needed in this teaching material
RamWait	O	Provides data-valid feedback in the burst mode; not needed in this teaching material

Example: Timing

- The off-chip memory supports a variety of transfer modes (basic read and write timing graph below).
- Copes with different data width, namely, merging two 16-bit data from the memory to form 32-bit data to be transferred through the AXI bus.