AMBA Bus Integration in SoC

Contents

Introduction (1)

- In an SOC, different bus protocols like AXI, AHB, and APB are used to serve different purposes based on speed, complexity, and functionality.
  - To enable communication between components using different protocols, bus bridges are used to perform protocol conversion.
  - These buses are not directly compatible, so protocol conversion is necessary when:
    - ✓ An AXI master must access a peripheral on APB.
    - ✓ A processor on AHB must read data from a device on AXI.
    - ✓ A bridge connects hierarchical bus structures in a complex SoC.

Introduction (2)

- A bridge module converts signals.
  - AXI-to-APB bridge
    - ✓ Maps AXI ARADDR, ARVALID, RREADY to APB PADDR, PSEL, PENABLE, and vice versa.
    - ✓ Handles handshake differences (AXI uses VALID/READY, APB uses PSEL/PENABLE).
    - ✓ Translates burstless AXI4-Lite transactions to the two-phase APB protocol.
  - AXI-to-AHB or AHB-to-APB bridges
    - ✓ Convert transaction formats, control signals, and timing.
    - ✓ May involve buffering and finite state machines to match timing requirements.
- Due to the increased complexity of other bridges such as AXI to AHB, this lecture focuses solely on the simpler AXI4-Lite to APB bridge.

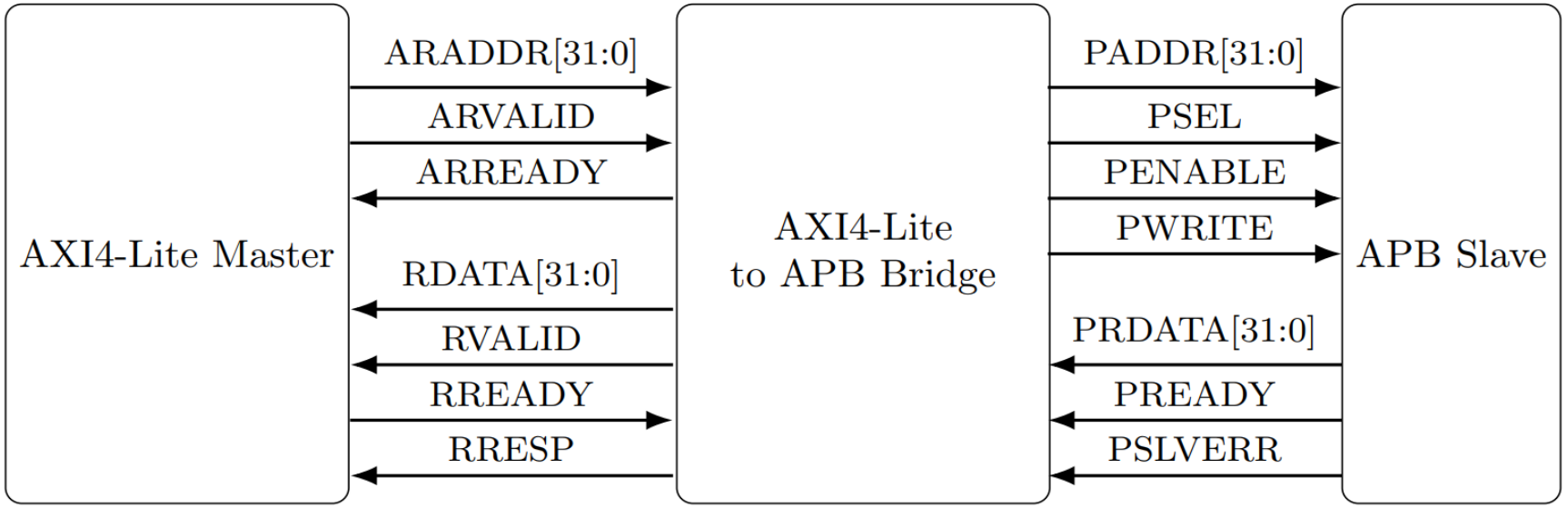
Introduction (3)

- Data width converter
  - The AXI protocol typically operates with a 64-bit or even wider data bus,while the APB protocol commonly uses a 32-bit data bus.
  - An upsizer converts data from a narrower source (e.g., 32-bit) to a wider destination (e.g., 64-bit).
    - ✓ It may group multiple 32-bit transactions or fill unused bits to align with the wider bus.
  - A downsizer performs the opposite operation.
    - ✓ It splits a wider transaction (e.g., 64-bit AXI data) into multiple narrower APB transfers.
  - Although this lecture assumes a 32-bit AXI bus for simplicity, real-world AXI interfaces are usually 64-bit or more, and width conversion logic such as upsizers and downsizers is often required when interfacing with APB or other narrower protocols.

Contents

AXI4-Lite to APB Bridge (1)

- The AXI4-Lite to APB bridge is a hardware module that connects an AXI4-Lite master interface to an APB slave.
  - It translates AXI4-Lite read or write transactions into equivalent APB operations, enabling AXI-based systems to communicate with simpler APB peripherals.



< AXI4-Lite to APB Bridge (Read channel only) >

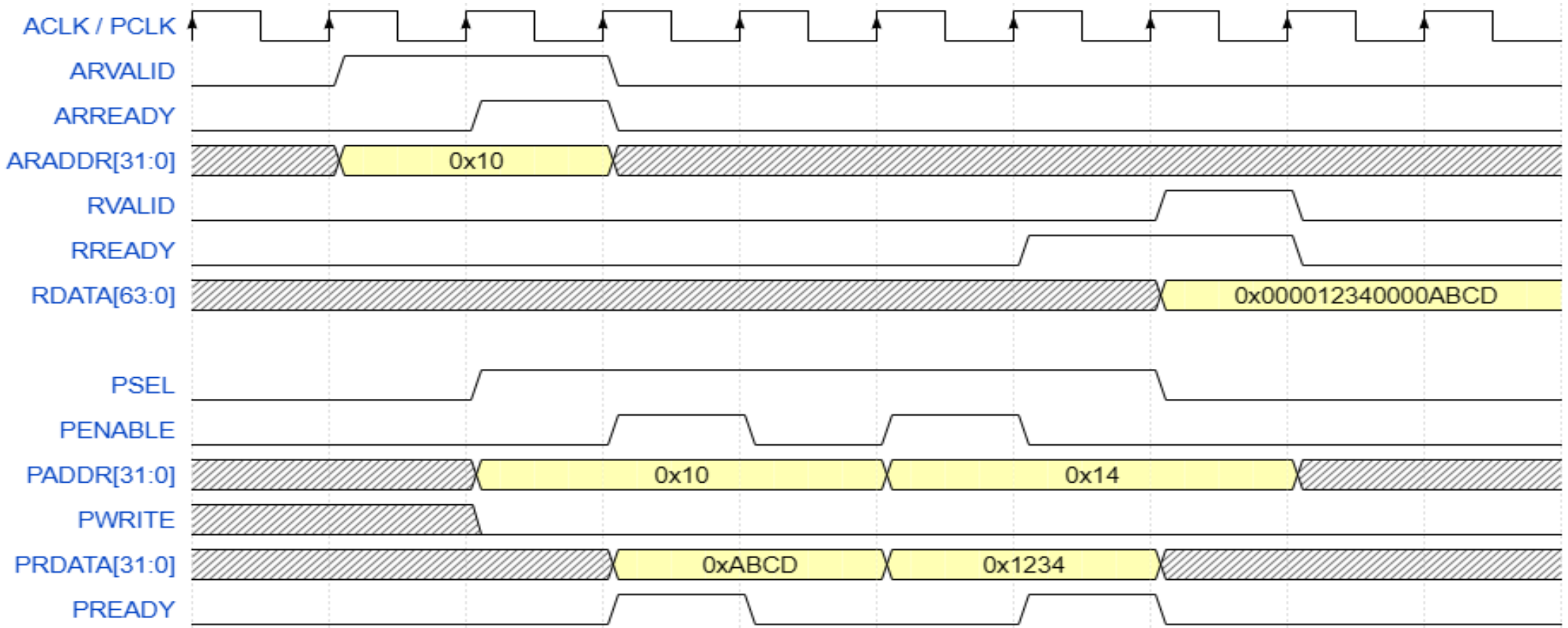
AXI4-Lite to APB Bridge (2)

- The bridge typically operates using a finite state machine (FSM) to control the timing of APB transfers, which require a two-phase access (setup and enable), while maintaining AXI's handshaking mechanism.
  - In a read transaction, the bridge receives the ARADDR and ARVALID signals from the AXI master.
  - It then generates the corresponding APB signals: PADDR, PSEL, and PENABLE.
  - Once the APB slave returns valid data (PRDATA) and asserts PREADY, the bridge responds to the AXI master with RDATA and RVALID.

- Using a bridge often introduces cycle bubbles between address and data phases due to protocol differences and internal control logic.

AXI4-Lite to APB Bridge (3)

- A read transaction proceeds as follows:

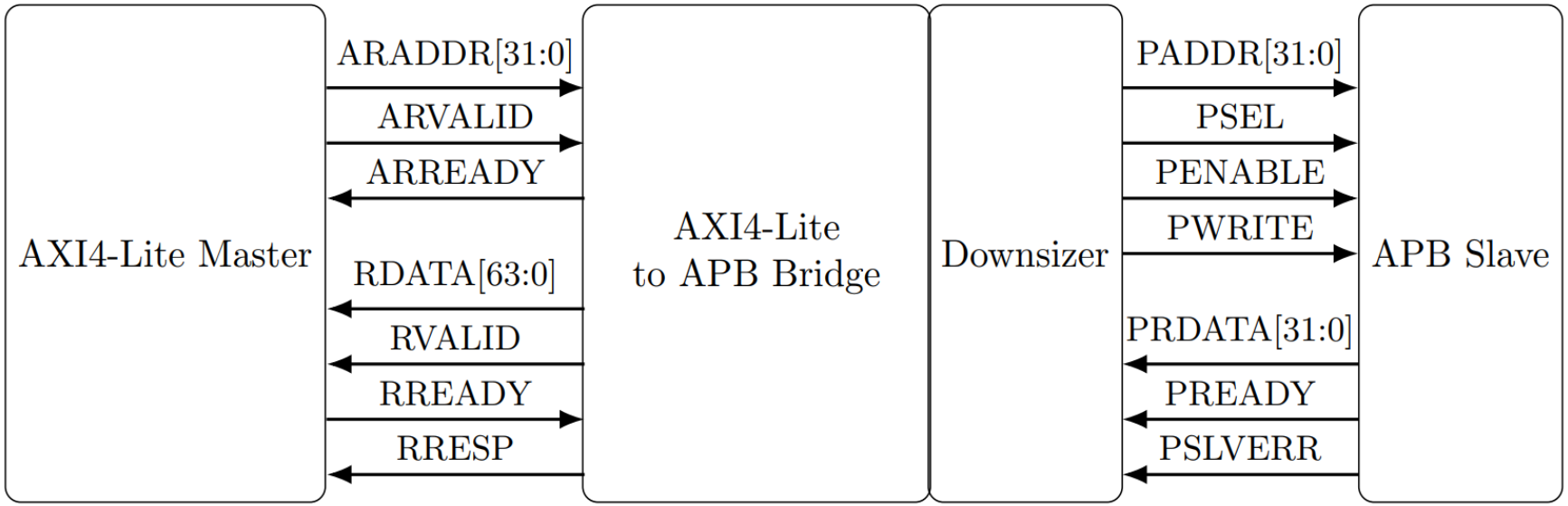


< AXI4-Lite to APB Read Transaction >

Contents

Data Width Converter (1)

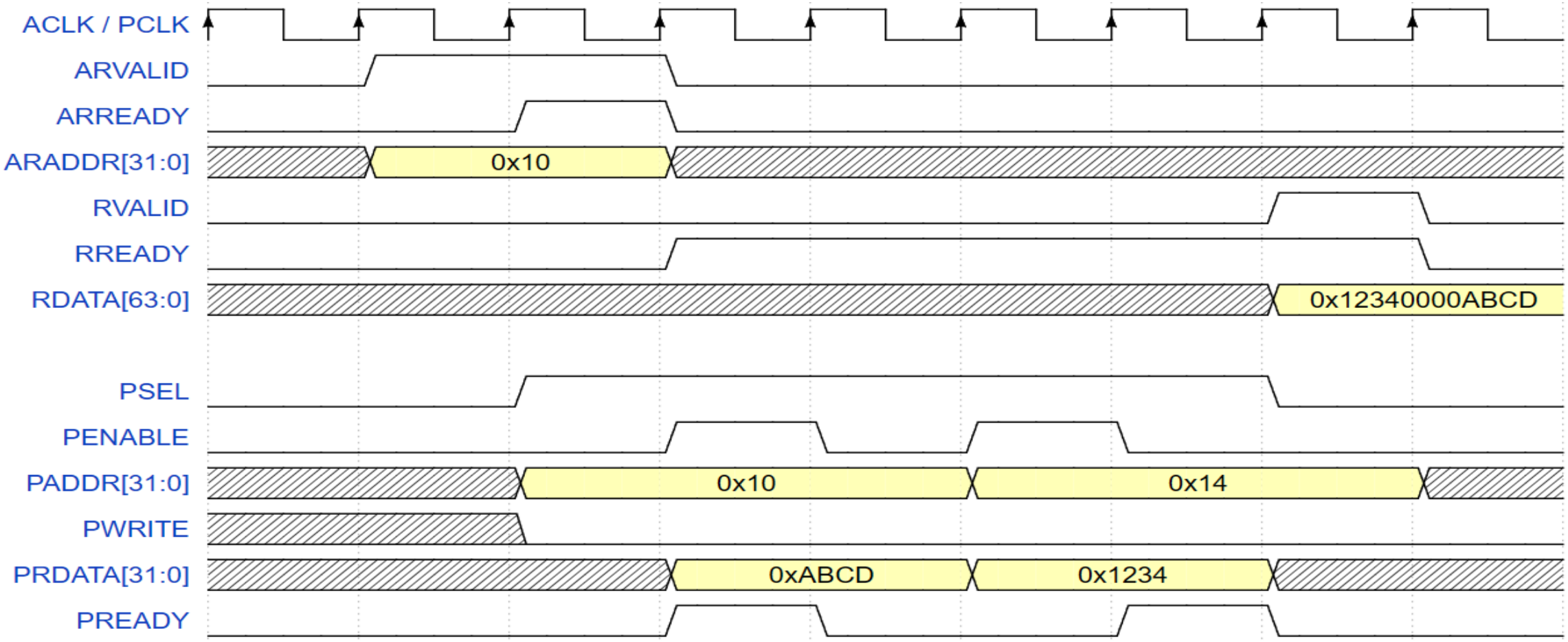
- A 64-bit AXI4-Lite master communicates with a 32-bit APB slave through an AXI4-Lite to APB bridge and a downsizer.
  - The bridge handles the protocol conversion between AXI and APB, while the downsizer adjusts the data width from 64 bits to 32 bits.
  - This allows the AXI master to perform transactions with the narrower APB slave.



< Bridge and Downsizer (Read channel only) >

Data Width Converter (2)

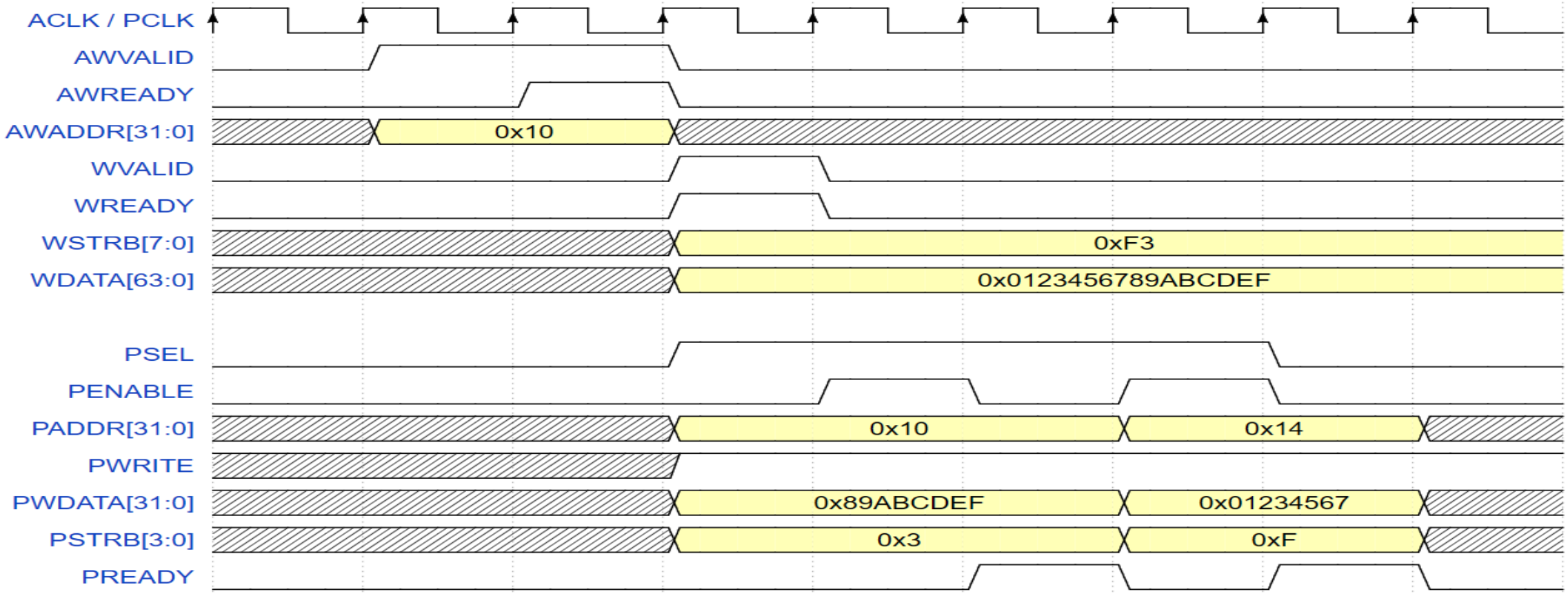
- A read transaction proceeds as follows:



< AXI4-Lite to APB Read Transaction with Data Width Converter >

Exercise 1

- A write transaction proceeds as follows:
  - B channel is not shown.



< AXI4-Lite to APB Write Transaction with Data Width Converter >