# AMBA®3 AHB Lite Bus Architecture

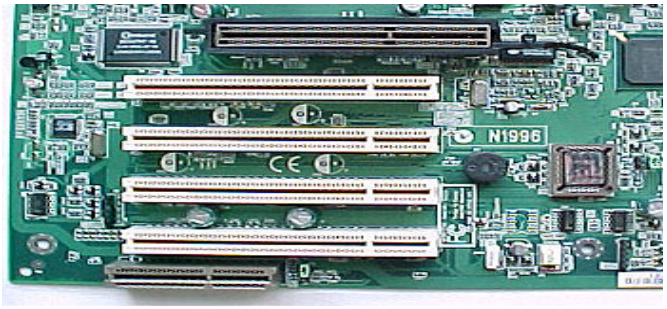
# Module Syllabus

- What is a Bus
- Bus Types
- ARM AMBA System Buses
- AMBA3 AHB-Lite Bus
- Bus Operation in General
- AHB Bus Components
- AHB Bus Signals
- AHB Bus Basic Timing
- AHB Bus Implementation
- Resources

# Bus in General

# What is Bus

- Traditionally, a bus is a communication system that allows data to be transferred between different components in a computer.
- The infrastructures is defined in both hardware and software :
  - Hardware infrastructure includes the physical implementation, such as cables or wires. For example, the PCI uses PCI cable to connect components inside a desktop.
  - Software infrastructure includes the bus protocol, e.g. PCI bus protocol.




**PCI socket on a mother board**

**PCI bus cable**

# Bus Types

- Buses can be categorized into two types:
  - External bus
    - Used to connect external devices, such as a computer to a printer;
  - Internal bus
    - Used to connect internal components inside a computer, such as a CPU to a memory;
    - Also known as system bus;
    - Less overhead, e.g. not need for electrical characteristics handling and configuration detection etc..
    - Thus typically runs faster than the external bus;
    - In a SoC design, the internal bus is integrated onto a single chip, thus can also be referred on-chip system bus.

# ARM AMBA Buses

# ARM AMBA System Bus

- AMBA: Advanced Microcontroller Bus Architecture
  - AMBA protocol is an open standard (except AMBA-5), on-chip interconnect specification;
  - Used as the on-chip bus in ARM-based SoC designs;
  - Provides the interface standard that enables IP re-use;
  - Facilitates right-first-time development of multi-processor designs with large numbers of controllers and peripherals;
  - Widely used in modern portable mobile devices, such as tablets and smartphones.

# ARM AMBA Bus Families

| AMBA family | Bus protocol | Processor |
|---|---|---|
| AMBA 5 | CHI | Cortex-A57, A53 |
| AMBA 4 | ACE | Cortex-A7, A15 |
| | AXI4 | |
| AMBA 3 | AXI | Cortex-A9, A8, R4, R5 |
| | AHB (**AHB-Lite**) | **Cortex-M0**, M3, M4 |
| | APB | Cortex-M0, M3, M4 |
| | ATB | |
| AMBA 2 | AHB, APB | ARM7, ARM9 |
| AMBA 1 | ASB, APB | |

As of Sept 2013

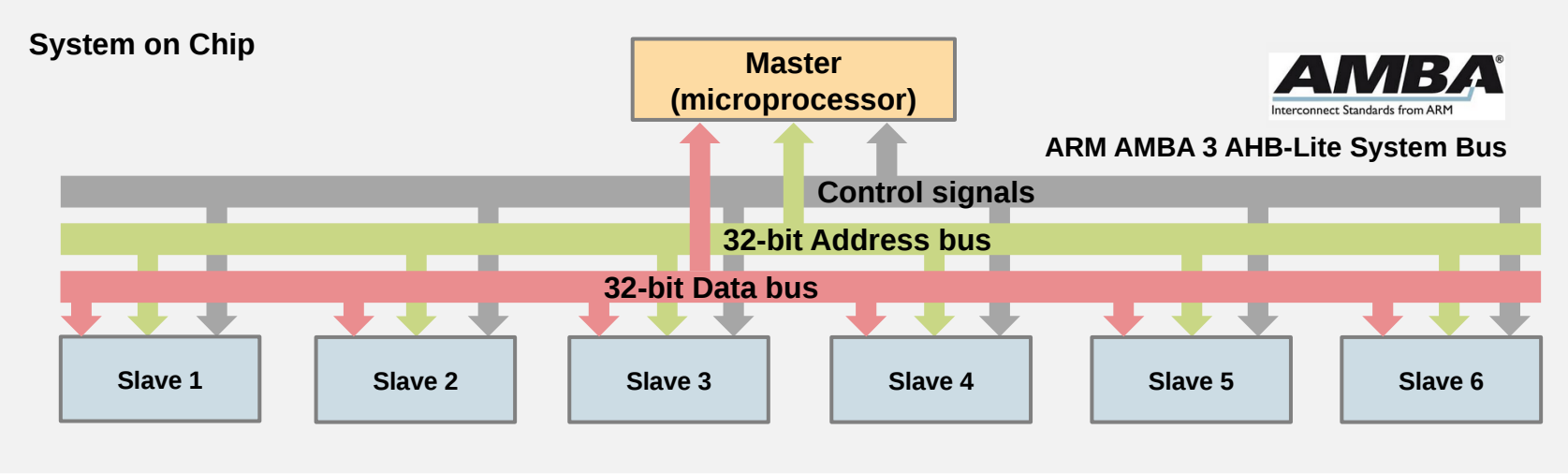
# AMBA3 AHB-Lite Bus

- AHB: Advanced High Performance Bus
  - High-performance synthesizable designs;
  - Supports multiple bus masters;
  - Provides high-bandwidth operation;

- AHB-Lite:
  - A subset of AHB;
  - Simplifies the design of AHB bus, e.g. typically with a single master.

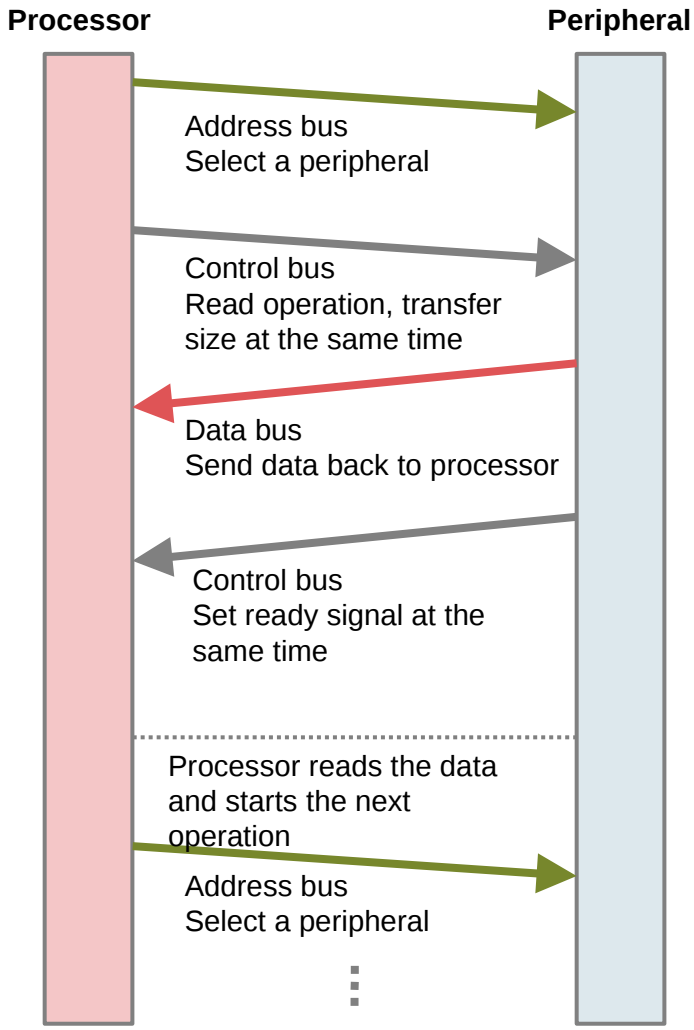
# Bus Operation in General

- Processor controls all peripherals via an AHB-Lite system bus;
- The AHB-Lite bus consists of a data bus and an address bus, with additional control signals;
  - Data bus is used to exchange data information;
  - Address bus is used to select one of the peripherals (or one register of a peripheral);
  - Control signals are used to synchronize and identify transactions, such as ready, write/ read, transfer mode signals.



# Bus Operation in General

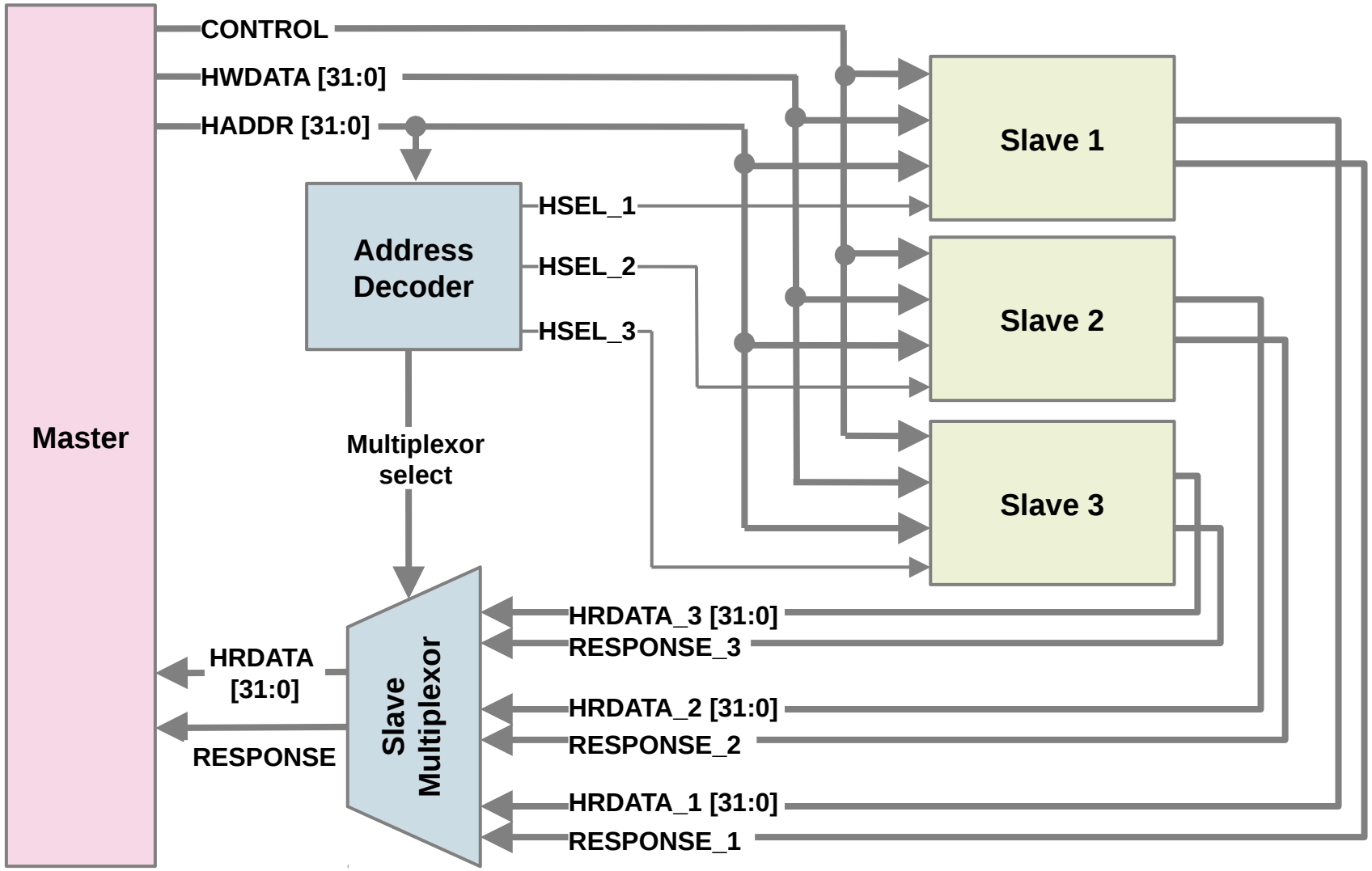
- A typical operation to access a peripheral mainly consists of:
  - Select one peripheral (or one register) by giving the address to the address bus;
  - At the same time, set control signals, such as read or write, transfer size and so forth;
  - Wait for the peripheral to be ready, and then read the data from data bus.




- Apart from above, AHB-Lite bus (or any other commercialized bus) has more functionalities, such as transfer size, burst mode, etc…
- The following slides explain the components and signals used in AHB-Lite bus. However, to just perform a basic data transfer, not all the signals are needed.

# AMBA3 AHB-Lite Bus Components

# AHB-Lite Bus Block Diagram

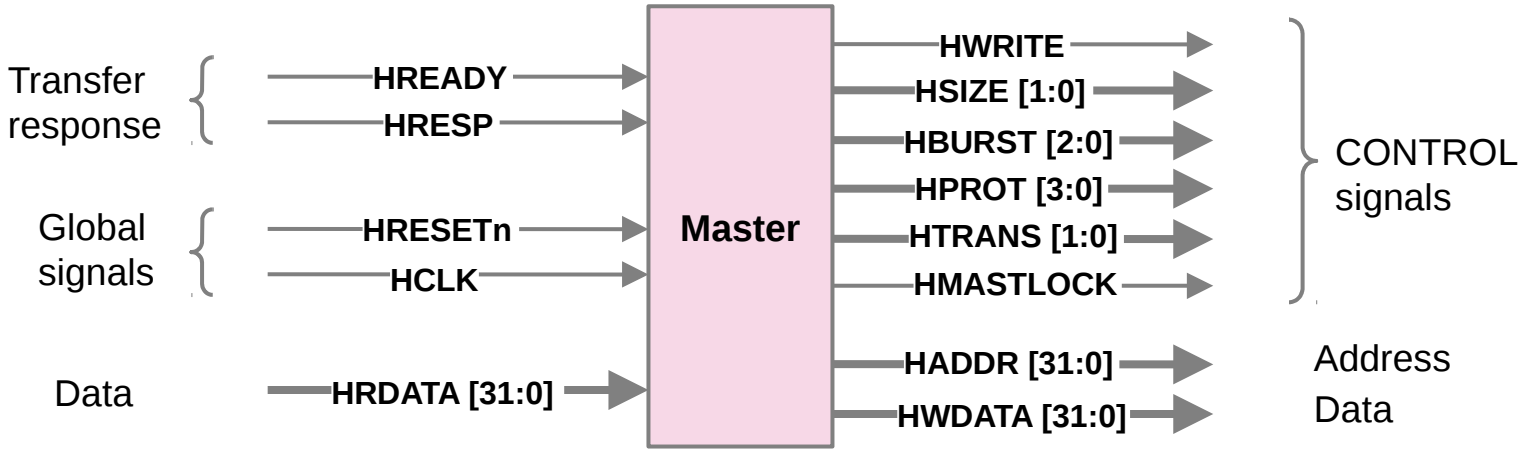


# Global Signals

| Signal | Direction | Description |
|---|---|---|
| HCLK | clock source → all components | The bus clock times all bus transfers. All signal timings are related to the rising edge of HCLK |
| HRESETn | Reset controller → all components | The bus reset signal is active low and resets the system and the bus |

# AHB-Lite Master Interface

- The AHB-Lite master provides address and control information to initiate read and write operations.
- The master also receives the response from the slave, including data, ready and response signal.

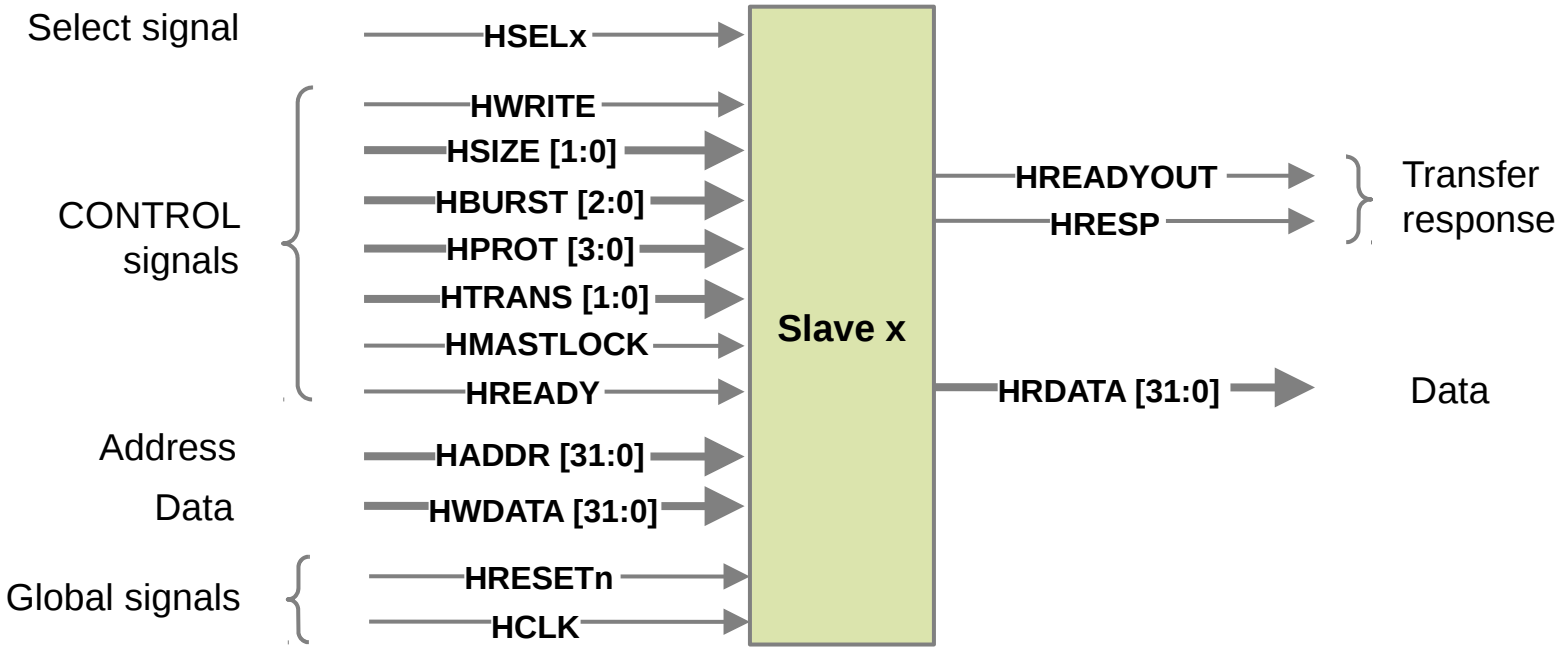


# AHB-Lite Master Signals

| Signal | Direction | Description |
|---|---|---|
| HADDR [31:0] | Master → slaves and decoder | 32-bt system address bus |
| HWDATA [31:0] | Master → slaves | The write data bus transfers data from the master to the slaves during write operations |
| HWRITE | Master → slaves | Indicates the transfer direction. When HIGH this signal indicates a write transfer and when LOW a read transfer. |
| HSIZE [2:0] | Master → slaves | Indicates the size of the transfer, that is typically byte, halfword, or word |
| HBURST [2:0] | Master → slaves | The burst type indicates if the transfer is a single transfer or forms part of a burst |
| HPROT [3:0] | Master → slaves | The protection control signals provide additional information about a bus access and are primarily intended for use by any module that wants to implement some level of protection. |
| HTRANS [1:0] | Master → slaves | Indicates the transfer type of the current transfer. This can be: IDLE, BUSY, NONSEQUENTIAL, or SEQUENTIAL. |
| HMASTLOCK | Master → slaves | When HIGH, this signal indicates that the current transfer is part of a locked sequence. |

# AHB-Lite Slave Interface

- An AHB-Lite slave responds to transfer initiated by the master in the system.
- The signal HSELx is the output from the address decoder, which is used to select one of the slaves at one time.

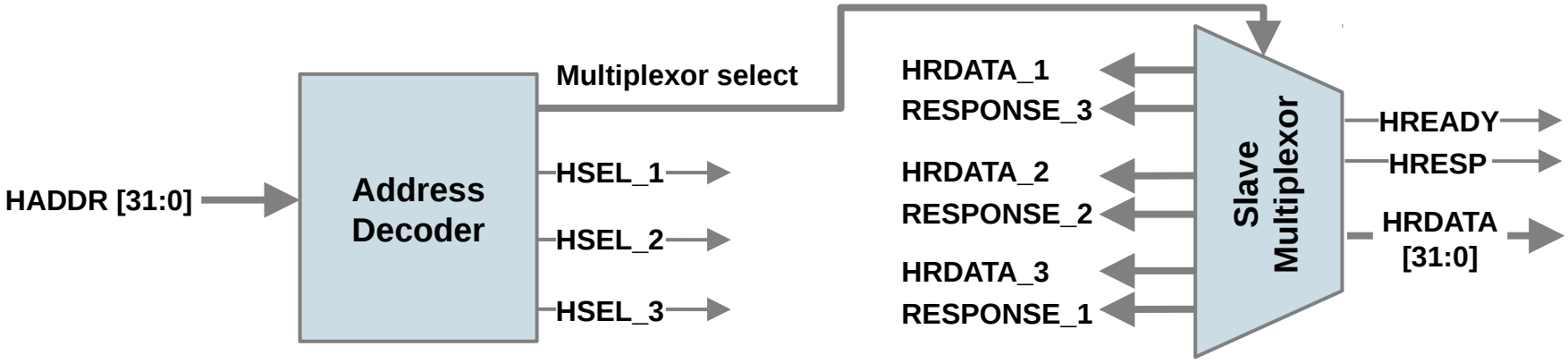


# AHB-Lite Slave Signals

| Signal | Direction | Description |
|---|---|---|
| HRDATA [31:0] | Slave → multiplexor | During read operations, the read data bus transfers data from the selected slave to the slave multiplexor. The multiplexor then transfers the data to the master. |
| HREADYOUT | Slave → multiplexor | When HIGH, the HREADYOUT signal indicates that a transfer has finished on the bus. This signal can be driven LOW to extend a transfer. |
| HRESP | Slave → multiplexor | The transfer response, after passing through the multiplexor, provides the master with additional information on the status of a transfer.<br>When LOW, the HRESP signal indicates that the transfer status is OKAY.<br>When HIGH, the HRESP signal indicates that the transfer status is ERROR. |

# Address Decoder and Slave Multiplexor

- Address decoder
  - Selects one of the slaves depending on the current address bus;
  - Also informs the slave multiplexor.
- Slave multiplexor
  - Inputs the response signals (HRDATA, HREADY and HRESP) from all the slaves, and outputs one of them depending on the selecting signal from the address decoder.



# Decoder and Multiplexor Signals

| Signal | Direction | Description |
|---|---|---|
| HRDATA [31:0] | Multiplexor → master | The read data from the multiplexor to the master |
| HREADY | Multiplexor → master and slaves | The ready signal from the multiplexor to the master<br>When HIGH, the HREADY signal indicates to the master and all slaves, that the<br>previous transfer is complete. |
| HRESP | Multiplexor → master | The transfer response signal from the multiplexor to the master |
| HSELx | Decoder → slaves | Each AHB-Lite slave has its own slave select signal HSELx and this signal indicates that the current transfer is intended for the selected slave. When the slave is initially selected, it must also monitor the status of HREADY to ensure that the previous bus transfer has completed, before it responds to the current transfer. |

# AHB-Lite Bus Timing

# AHB-Lite Bus Timing

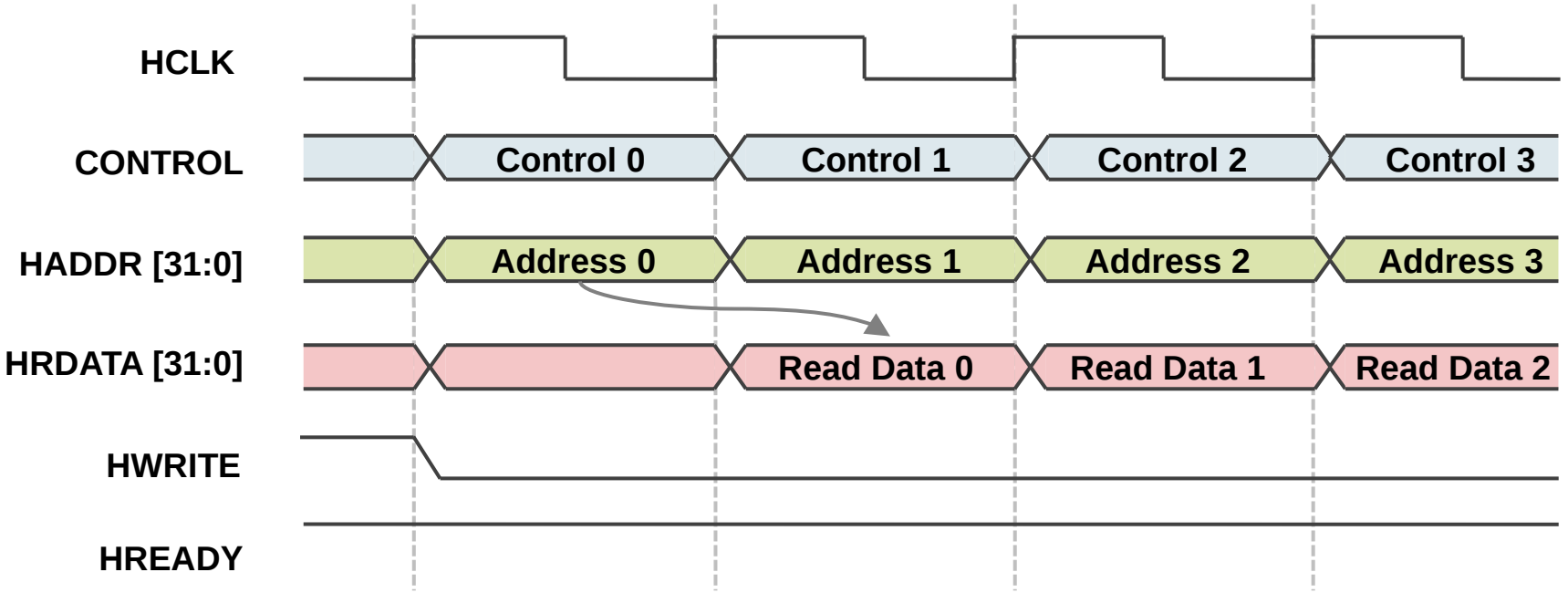
- An AHB-Lite transfer consists of two phases:
  - Address phase
    - Lasts for a single HCLK cycle unless its extended by the previous bus transfer;
  - Data phase
    - That might require several HCLK cycles. Use the HREADY signal to control the number of clock cycles required to complete the transfer.
- Pipelined transfer
  - The data access of the current operation is overlapped with the address access of the next operation;
  - Enables high performance operation while still providing adequate time for a slave to provide the response to a transfer.

# AHB-Lite Bus Timing

- In our EDK teaching material, we only present the basic bus operation, namely:
  - No BURST transaction;
    - HBURST[2:0] is always 3’b000;
  - Never generates locked transactions
    - HMASTLOCK is always 1’b0;
  - All transactions issued are non-sequential transfer
    - HTRANS[1:0] is either 2’b00 (IDLE) or 2’b10 (Non Sequential)
- More advanced bus transaction can be found in the given reference1:
  - AMBA 3 AHB-Lite Protocol Specification

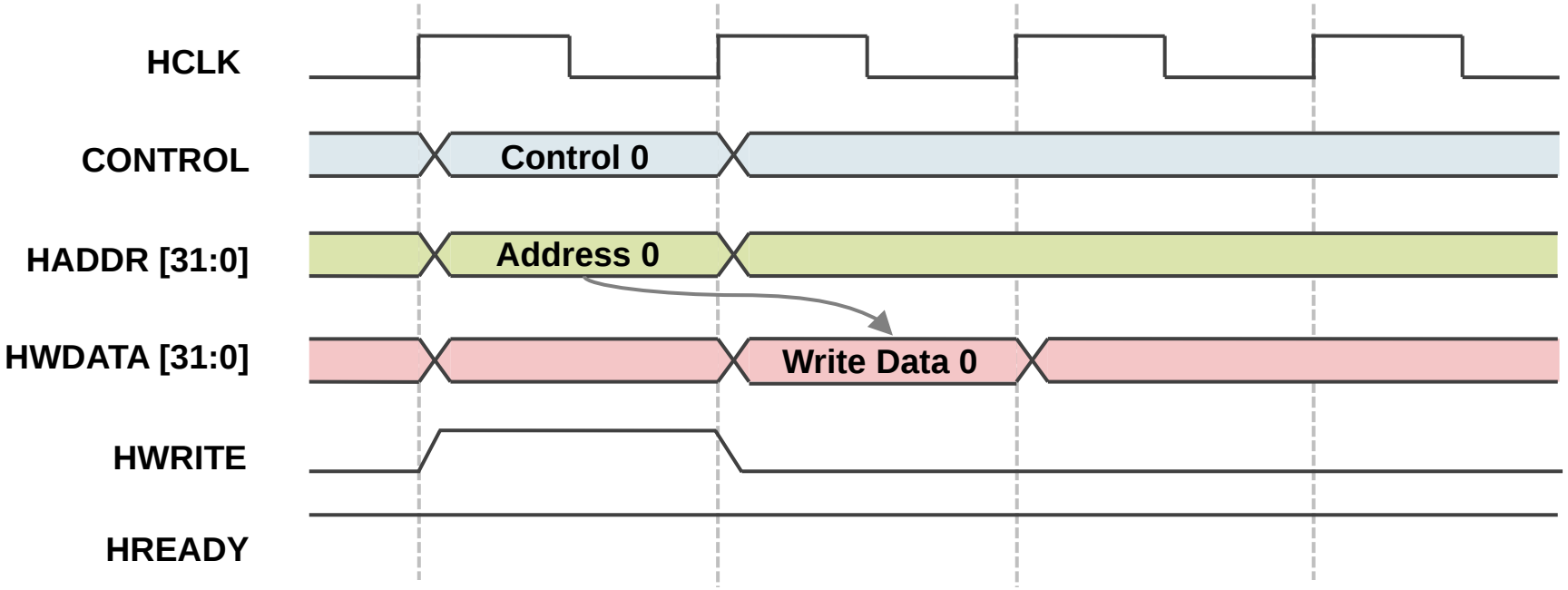
# Basic Read Transfer

- Address phase (first clock cycle)
  - Give Address and control signals, clear HWRITE to zero;
- Data phase (second clock cycle)
  - Data is available at HRDATA.
- No wait states (the slave is always ready to give its data);



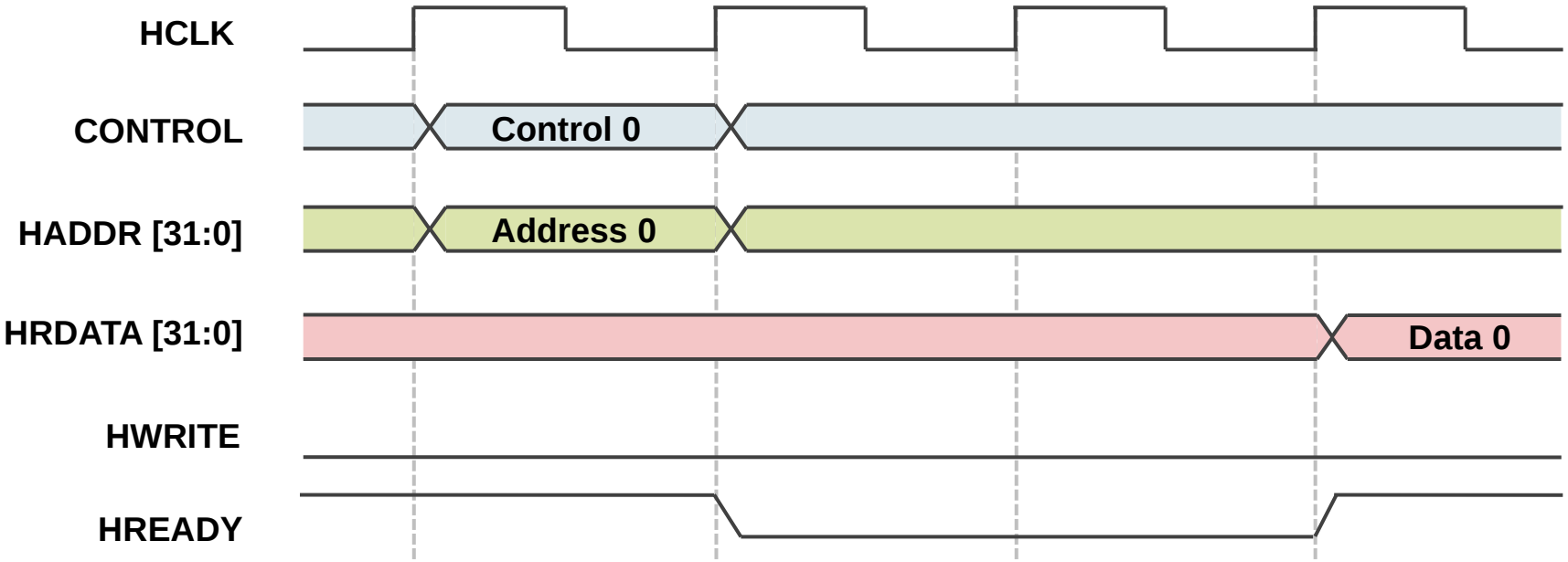
# Basic Write Transfer

- Address phase (first clock cycle)
  - Give Address and control signals, set HWRITE to one;
- Data phase (second clock cycle)
  - Give data to HWDATA.
- No wait states (the slave is always ready to receive the data);



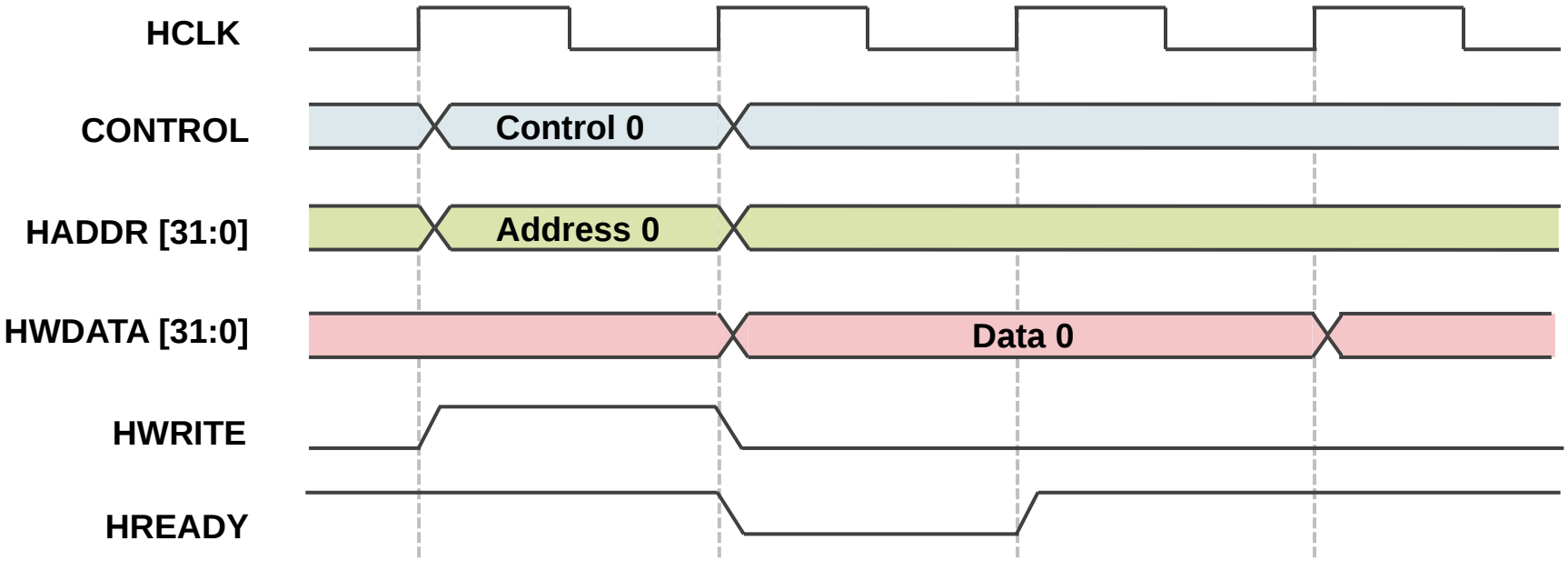
# Read Transfer with Wait State

- Address phase (first clock cycle)
  - Give Address and control signals, set HWRITE to one;
- Data phase (multiple clock cycles)
  - The slave holds HREADY to zero if its is not ready to provide its data; the master delays its next transaction;
  - When the slave is ready, the data will be given at HRDATA; at the same time, HREADY is set to one. The master will then continue its next transaction.



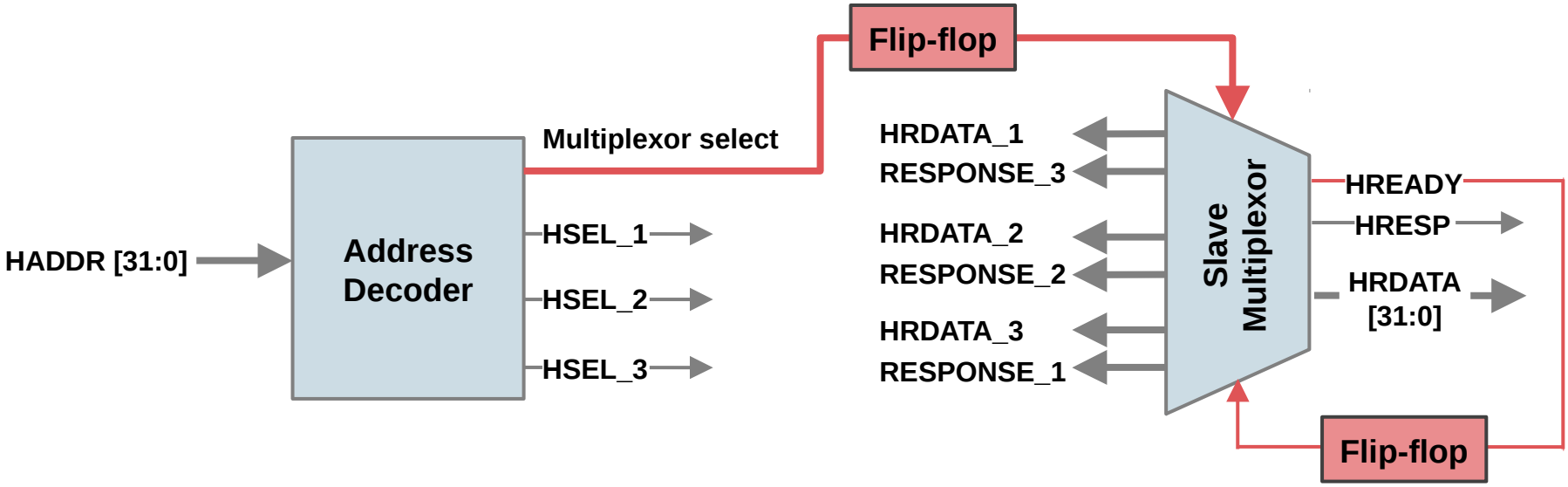
# Write Transfer with Wait State

- Address phase (first clock cycle)
  - Give Address and control signals, clear HWRITE to zero;
- Data phase (multiple clock cycles)
  - The master gives its data at HWDATA; the slave holds HREADY to zero if its is not ready to receive the data; the master delays its next transaction;
  - When the slave is ready, it will receive the data and set HREADY to one. The master will then continue its next transaction.



# Hardware Implementation

- Due to the pipelined operation, some signals have to be deliberately delayed including:
  - The selecting signals from the decoder to the multiplexor is delayed for one clock cycle;
  - The HREADY signal is delayed for one clock cycle before it is feedback to the multiplexor;
- The detailed implementation can be referred in the code that is provided in the EDK.



# Useful Resources

- Reference1
  - AMBA 3 AHB-Lite Protocol Specification

http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ihi0033a/index.html