CSE/ELE 664
Intro. To SoC Design

Lecture 25

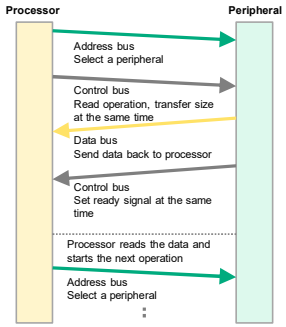
ENGINEERING@SYRACUSE

1

A Typical Bus Operation Example, Part IV

A typical operation to access a peripheral mainly consists of:

1. The master (e.g., a processor) selects one peripheral (or one register) by giving the address to the address bus. At the same time, it sets control signals, such as read or write, transfer size, and so forth.
2. The master waits for the slave (e.g., peripheral) to respond.
3. Once the slave is ready, it sends back the requested data to the processor; at the same time, it sets the ready signal on the control bus.
4. Finally, the master reads the transmitted data, and starts another communication cycle.



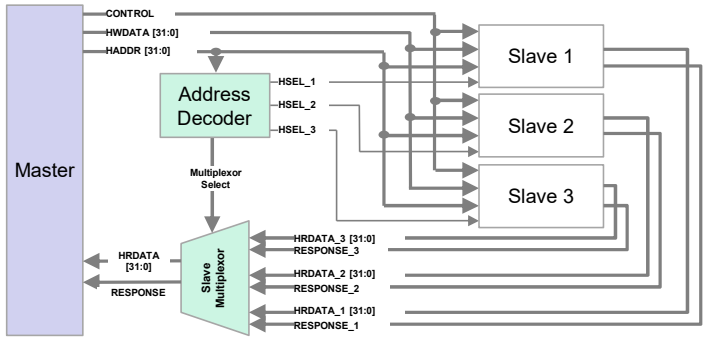
2

AMBA® 3 AHB-Lite Bus Architecture

ENGINEERING@SYRACUSE

3

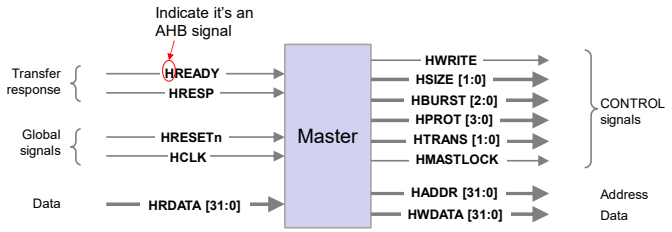
AHB-Lite Bus Block Diagram



4

AHB-Lite Master Interface

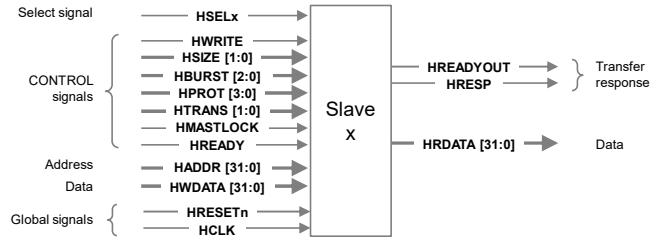
- The AHB-Lite master provides address and control information to initiate read and write operations.
- The master also receives the response from the slave, including data, ready and response signal.



5

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.



6

Read and Write Using AHB-Lite Bus

ENGINEERING@SYRACUSE

7

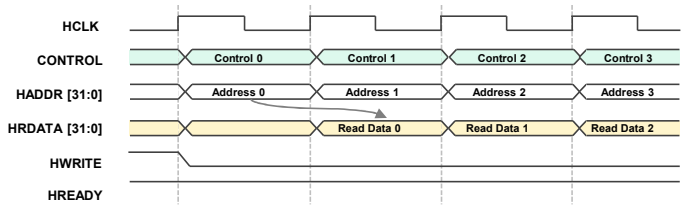
AHB-Lite Operation Principles

- **AHB-Lite supports three types of transfers:**
  1. Single
  2. Incrementing bursts that do not wrap at address boundaries
  3. Wrapping bursts that wrap at particular address boundaries
- **An AHB-Lite transfer consists of two phases:**
  1. **Address phase:** lasts for a single HCLK cycle, unless its extended by the previous bus transfer
  2. **Data phase** might require several HCLK cycles; the HREADY signal is used to control the number of clock cycles required to complete the transfer

8

Basic Read Transfer, Part I

Consider a simple read transfer with no wait states:

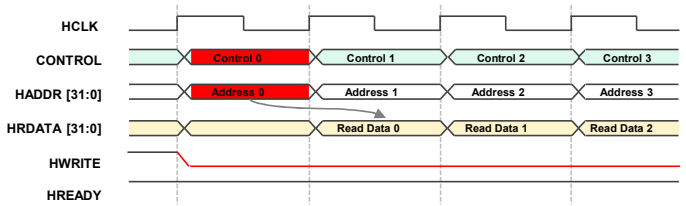


9

Basic Read Transfer, Part II

Consider a simple read transfer with no wait states:

1. Address phase: The master drives the address and control signals onto the bus after the rising edge of HCLK.

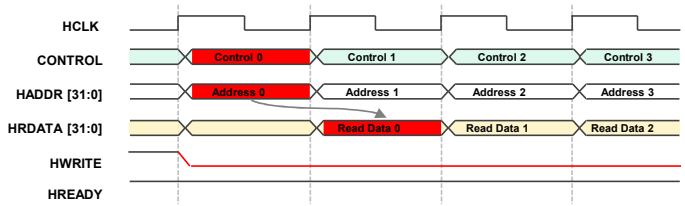


10

Basic Read Transfer, Part III

Consider a simple read transfer with no wait states:

1. Address phase: The master drives the address and control signals onto the bus after the rising edge of HCLK.
2. Data phase: The slave samples the address and control information, and makes data is available at HRDATA, and can start to drive the appropriate HREADY response.

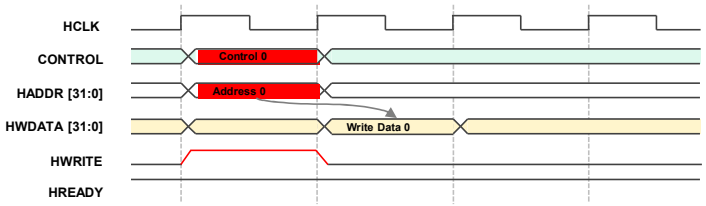


11

Basic Write Transfer

Consider a simple write transfer with no wait states:

1. Address phase: The master drives the address and control signals onto the bus after the rising edge of HCLK, set HWRITE to one.
2. Data phase: The slave samples the address and control information, and makes data is available at HRDATA, and can start to drive the appropriate HREADY response.

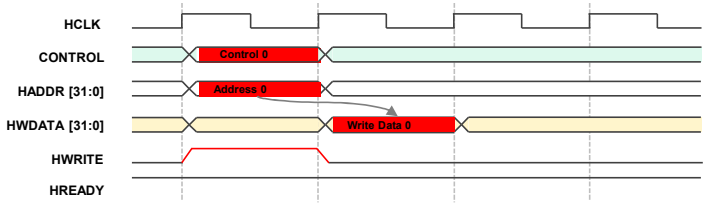


12

Basic Write Transfer (cont.)

Consider a simple write transfer with no wait states:

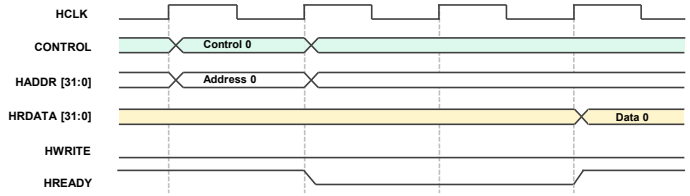
1. Address phase: The master drives the address and control signals onto the bus after the rising edge of HCLK, set HWRITE to one.
2. Data phase: The slave samples the address and control information, and makes data is available at HRDATA, and can start to drive the appropriate HREADY response.



13

Read Transfer with Wait State

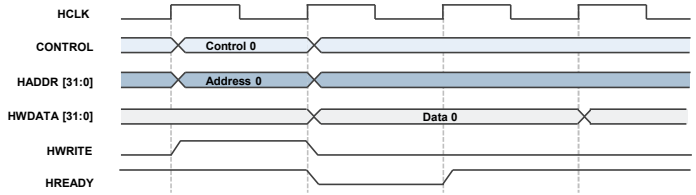
- Address phase (first clock cycle)
  - Give address and control signals; set HWRITE to zero
- Data phase (multiple clock cycles)
  - The slave holds HREADY to zero if it is not ready to provide its data; the master delays its next transaction
  - When the slave is ready, the data will be given at HRDATA;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



14

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 it 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



15