AHB2WB Specification

Introduction

The `AHB2WB` module serves as a bridge between the Advanced High-performance Bus (AHB) and the Wishbone bus, facilitating communication between an AHB master and a Wishbone slave. This module adheres to the AHB-Lite protocol for the master side and the Wishbone protocol for the slave side. It supports 32-bit data transfers and does not handle burst transfers. The design ensures that data is synchronized between the two bus systems, with a one-cycle delay for the AHB master due to the differing data availability timings between AHB and Wishbone.

Architecture

The `AHB2WB` module is a single-level design that interfaces directly with both the AHB master and the Wishbone slave. Key operations include:

- **Signal Registration**: AHB signals such as address, write control, transaction type, size, and select are registered to ensure stable signal transitions.
- **Control Logic**: Determines whether the master intends to perform a read or write operation and generates corresponding Wishbone control signals.
- **Data Flow**: Data from the AHB master is directly transferred to the Wishbone slave, with appropriate control signals to manage the data flow.
- **Clock and Reset Management**: The module operates synchronously with the AHB clock and uses an active-low reset signal.

Notable Implementation Details

- The module uses a high-priority mechanism to pull down the `HREADYOUT` signal to prevent the AHB master from latching incorrect data.
- The `HREADYOUT` signal is driven by the acknowledgment from the Wishbone slave, ensuring that the AHB master only proceeds when the Wishbone transaction is complete.

Interface

Signal	Width	In/Out	Description
HCLK	1	In	AHB clock signal.
HRESETn	1	In	Active-low reset signal for the AHB interface.
HADDR	32	In	AHB address bus.
HWDATA	32	In	AHB write data bus.
HWRITE	1	In	AHB write control signal.
HSEL	1	In	AHB select signal, indicating the target peripheral.
HTRANS	2	In	AHB transfer type signal.

Signal	Width	In/Out	Description
HSIZE	3	In	AHB transfer size signal.
HREADY	1	In	AHB ready signal, indicating the master is ready for the next transfer.
HRDATA	32	Out	AHB read data bus.
HRESP	1	Out	AHB response signal, always zero indicating OKAY response.
HREADYOUT	1	Out	AHB ready output signal, indicating the slave is ready for the next transfer.
wb_clk_o	1	Out	Wishbone clock output, driven by HCLK.
wb_rst_o	1	Out	Wishbone reset output, active-high, driven by HRESETn.
wb_adr_o	32	Out	Wishbone address output.
wb_dat_o	32	Out	Wishbone data output.
wb_sel_o	4	Out	Wishbone byte select output.
wb_we_o	1	Out	Wishbone write enable output.
wb_stb_o	1	Out	Wishbone strobe output.
wb_cyc_o	1	Out	Wishbone cycle output.
wb_dat_i	32	In	Wishbone data input.
wb_ack_i	1	In	Wishbone acknowledge input.

Timing

- **Latency**: The module introduces a one-cycle delay for the AHB master due to the need to synchronize data availability between AHB and Wishbone.
- **Signal Behavior**: All operations are synchronous to the rising edge of the `HCLK`. The `HREADYOUT` signal is asserted based on the Wishbone acknowledgment (`wb_ack_i`), ensuring that the AHB master only proceeds when the Wishbone transaction is complete.

Usage

To use the `AHB2WB` module:

1. **Initialization**: Ensure that the `HRESETn` signal is asserted low to reset the module. Upon release, the module will be ready for operation.
2. **Data Transfer**:
   - For a write operation, set `HWRITE` high, provide the address on `HADDR`, and data on `HWDATA`.
   - For a read operation, set `HWRITE` low and provide the address on `HADDR`.
3. **Control Signals**: Ensure `HSEL` is asserted to select the module and `HTRANS` is set to a valid transfer type.

4. **Monitor Outputs**: Check `HREADYOUT` to determine when the AHB master can proceed with the next transaction. The `HRDATA` will contain valid data during read operations when `HREADYOUT` is asserted.

This module is designed for seamless integration into systems requiring AHB to Wishbone communication, ensuring efficient data transfer between different bus architectures.

Functional Description (Generated by funcgen)

Module: AHB2WB

Purpose

The `AHB2WB` module provides an interface bridge between the AMBA Advanced High-performance Bus (AHB) and the Wishbone bus, enabling communication between devices on either side. It ensures data and control signals are transferred seamlessly between these two bus protocols.

Parameters

This module does not have configurable parameters.

Ports

AHB Master-Side Ports

- **HCLK (input, 1-bit):** AHB bus clock signal.
- **HRESETn (input, 1-bit):** Active low reset signal for the AHB bus.
- **HADDR (input, 32-bit):** Address bus from the AHB master.
- **HWDATA (input, 32-bit):** Data bus used to send data from the AHB master to the AHB slave.
- **HWRITE (input, 1-bit):** Control signal that indicates the operation type (read/write).
- **HSEL (input, 1-bit):** Slave select signal.
- **HTRANS (input, 2-bit):** Transaction type signal on AHB.
- **HSIZE (input, 3-bit):** Indicates the size of the transfer.
- **HREADY (input, 1-bit):** Indicates whether the bus is ready for further transactions.
- **HRDATA (output, 32-bit):** Data bus used to send data from the AHB slave back to the AHB master.
- **HRESP (output, 1-bit):** Response signal to indicate error or successful transaction.
- **HREADYOUT (output, 1-bit):** Indicates the readiness of the slave to complete the transaction.

Wishbone Slave-Side Ports

- **wb_clk_o (output, 1-bit):** Wishbone bus clock signal, driven by AHB clock.
- **wb_rst_o (output, 1-bit):** Wishbone reset signal, active high.
- **wb_adr_o (output, 32-bit):** Address bus for the Wishbone master.
- **wb_dat_o (output, 32-bit):** Data output from the Wishbone master.
- **wb_sel_o (output, 4-bit):** Wishbone byte select signal.
- **wb_we_o (output, 1-bit):** Wishbone write enable signal.
- **wb_stb_o (output, 1-bit):** Wishbone strobe signal to indicate valid data/address.
- **wb_cyc_o (output, 1-bit):** Wishbone cycle signal indicating a valid bus cycle.
- **wb_dat_i (input, 32-bit):** Data input to the Wishbone slave.
- **wb_ack_i (input, 1-bit):** Acknowledge signal from the Wishbone slave indicating the completion of a transaction.

Internal Signals

- **rHADDR (reg, 32-bit):** Registered version of the AHB address.
- **rHWRITE (reg, 1-bit):** Registered version of the AHB write control.
- **rHTRANS (reg, 2-bit):** Registered version of AHB transaction type.
- **rHSIZE (reg, 3-bit):** Registered version of AHB transfer size.
- **rHSEL (reg, 1-bit):** Registered version of AHB select signal.
- **_pull_down_HREADYOUT (reg, 1-bit):** Internal control used for HREADYOUT management.
- **master_wants_read (wire, 1-bit):** Indicates when the master wants to read.
- **master_wants_write (wire, 1-bit):** Indicates when the master wants to write.
- **r_wb_cyc_o (reg, 1-bit):** Registered Wishbone cycle indicator to manage cycle state.

Functionality

Sequential Logic

- The module uses both positive-edge and negative-edge triggered logic driven by `HCLK` and `HRESETn`.
- Registers AHB control and address lines (`rHADDR`, `rHWRITE`, `rHTRANS`, `rHSIZE`, `rHSEL`) when `HREADY` is asserted, ensuring synchronization of inputs.
- `r_wb_cyc_o` is used to maintain cycle state across clock cycles and is reset when `HRESETn` is de-asserted or a transaction completes.

Combinational Logic

- Determines whether the master wants a read or write using the `HTRANS` and `HWRITE` signals:
  - `master_wants_read`: When `HTRANS` indicates a non-idle state and write signal is low.
  - `master_wants_write`: When `HTRANS` indicates a non-idle state and write signal is high.
- Generates appropriate Wishbone signals (`wb_stb_o`, `wb_cyc_o`, `wb_we_o`) based on the read/write determination.
- The `HREADYOUT` signal is driven by the Wishbone acknowledge (`wb_ack_i`), indicating Wishbone transaction readiness.

Control Logic

- The Wishbone select line `wb_sel_o` is driven by the availability of the cycle and strobe signals.
- The `HREADYOUT` signal ensures no false data is latched by the master.

State Machines or Control Logic

- Implicitly performs state control using combinational and sequential elements, ensuring that data phases align with control signal assertions.

Instantiations

This module does not instantiate any sub-modules or additional components. It solely connects and manipulates signals based on its internal control logic.

Inter-Module Connections

- This description only covers a single module, having no hierarchical interplay with other modules. The AHB2WB module acts as a bridge interfacing between the AHB and Wishbone protocols, ensuring signals are properly forwarded across these. The key inter-module connection responsibility involves translating AHB transactions into Wishbone transactions independently.