BIU_AHB3Lite Specification

Introduction

The `biu_ahb3lite` module is a Bus Interface Unit (BIU) designed to interface with the AHB3 Lite bus, specifically for RISC-V processors. It facilitates communication between a RISC-V core and an AHB3 Lite bus by managing data transfers, burst operations, and handling strobe signals. The module adheres to the AHB3 Lite protocol, ensuring compatibility with systems that require strict adherence to the AHB specification, including the prevention of crossing 1kB address boundaries during burst transfers.

Architecture

The `biu_ahb3lite` module is structured to manage data transfers between a core and an AHB3 Lite bus. It includes the following key components:

- **State Machine**: Controls the data transfer process, managing burst operations and handling strobe acknowledgments.
- **Functions**: Several functions are defined to convert BIU-specific signals to AHB3 Lite compatible signals, such as `biu_size2hsize`, `biu_type2cnt`, `biu_type2hburst`, and `biu_prot2hprot`.
- **Address Calculation**: The `nxt_addr` function calculates the next address in a burst sequence, considering wrapping and boundary conditions.
- **Boundary Check**: The `cross1kB` function checks if a 1kB boundary is crossed during an incremental burst, ensuring compliance with the STRICT_AHB parameter.

The module operates synchronously with the `HCLK` clock signal and is reset asynchronously with the `HRESETn` signal.

Interface

Parameters

Parameter	Type	Default Value	Description
DATA_SIZE	parameter	32	Size of data buses
ADDR_SIZE	parameter	DATA_SIZE	Size of address buses
TAG_SIZE	parameter	DATA_SIZE	Size of user tag buses
STRICT_AHB	parameter	1	Adherence to AHB spec, preventing 1kB boundary crossing

Ports

Signal	Width	In/Out	Description
HRESETn	1	In	Asynchronous reset, active low
HCLK	1	In	Clock signal
HSEL	1	Out	AHB3 Lite bus select signal
HADDR	32	Out	AHB3 Lite address bus
HRDATA	32	In	AHB3 Lite read data bus
HWDATA	32	Out	AHB3 Lite write data bus
HWRITE	1	Out	AHB3 Lite write control signal
HSIZE	HSIZE_SIZE	Out	AHB3 Lite transfer size
HBURST	HBURST_SIZE	Out	AHB3 Lite burst type
HPROT	HPROT_SIZE	Out	AHB3 Lite protection control
HTRANS	HTRANS_SIZE	Out	AHB3 Lite transfer type
HMASTLOCK	1	Out	AHB3 Lite master lock signal
HREADY	1	In	AHB3 Lite ready signal
HRESP	1	In	AHB3 Lite response signal
biu_stb_i	1	In	BIU strobe input
biu_stb_ack_o	1	Out	BIU strobe acknowledge output
biu_d_ack_o	1	Out	BIU data acknowledge output
biu_adri_i	32	In	BIU address input
biu_adro_o	32	Out	BIU address output
biu_size_i	biu_size_t	In	BIU transfer size
biu_type_i	biu_type_t	In	BIU burst type
biu_prot_i	biu_prot_t	In	BIU protection type
biu_lock_i	1	In	BIU lock signal
biu_we_i	1	In	BIU write enable
biu_d_i	32	In	BIU data input
biu_q_o	32	Out	BIU data output
biu_ack_o	1	Out	BIU transfer acknowledge
biu_err_o	1	Out	BIU transfer error
biu_tagi_i	TAG_SIZE	In	BIU tag input
biu_tago_o	TAG_SIZE	Out	BIU tag output

Internal Signals

Signal	Width	Description
burst_cnt	4	Counter for burst length
data_ena	1	Data enable signal
ddata_ena	1	Delayed data enable signal
biu_di_dly	DATA_SIZE	Delayed BIU data input
incr_burst	1	Incremental burst indicator
tag	TAG_SIZE	Tag storage for data phase alignment

Timing

The `biu_ahb3lite` module operates synchronously with the `HCLK` clock signal. The module's state machine ensures that data transfers are aligned with the clock edges, and the `HREADY` signal is used to manage the flow of data and control signals. The latency for output validity is determined by the state machine's transitions and the `HREADY` signal.

Usage

To use the `biu_ahb3lite` module, follow these steps:

1. **Initialization**: Ensure the module is reset by asserting `HRESETn` low. After reset, provide a stable clock signal on `HCLK`.
2. **Data Transfer**:
   - Assert `biu_stb_i` to initiate a data transfer.
   - Provide the address on `biu_adri_i` and data on `biu_d_i` if writing.
   - Set the transfer size, type, and protection using `biu_size_i`, `biu_type_i`, and `biu_prot_i`.
   - Monitor `biu_stb_ack_o` for strobe acknowledgment and `biu_ack_o` for transfer acknowledgment.
   - Check `biu_err_o` for any transfer errors.
3. **Burst Operations**: For burst transfers, configure the burst type and ensure the address does not cross a 1kB boundary if `STRICT_AHB` is set.
4. **Data Reception**: Read data from `biu_q_o` when `biu_ack_o` is asserted.

The module is designed to handle both single and burst transfers, with support for incremental and wrapping bursts as per the AHB3 Lite specification.

Functional Description (Generated by funcgen)

Module: `biu_ahb3lite` (File: `biu_ahb3lite.sv`)

Purpose

The `biu_ahb3lite` module is a Bus Interface Unit (BIU) designed for integrating RISC-V processors with AHB3 Lite compliant bus systems. It translates core requests into AHB transfers, handles strobe acks, and processes tags returned with the data phase.

Parameters

- `DATA_SIZE`: Default value is 32. Represents the size of the data buses.
- `ADDR_SIZE`: Default is set to `DATA_SIZE`. Specifies the size of the address buses.
- `TAG_SIZE`: Default is set to `DATA_SIZE`. Defines the size of the user tag buses.

- **STRICT_AHB**: Default value is 1. When set, strictly adheres to the AHB spec, disallowing crossing a 1kB address boundary.

Ports

AHB3 Lite Bus Interface

- **HRESETn**: Input, 1-bit. Active low reset.
- **HCLK**: Input, 1-bit. Clock signal.
- **HSEL**: Output, 1-bit. Module selection signal on the AHB bus.
- **HADDR**: Output, 32-bit. Address bus for AHB system.
- **HRDATA**: Input, 32-bit. Read data from AHB slave.
- **HWDATA**: Output, 32-bit. Write data to AHB slave.
- **HWRITE**: Output, 1-bit. Indicates write transfer.
- **HSIZE**: Output, derived size. Width of data transfer.
- **HBURST**: Output, derived burst size. Specifies burst type.
- **HPROT**: Output, derived protection size. AHB protection control.
- **HTRANS**: Output, derived transfer size. Indicates transfer type.
- **HMASTLOCK**: Output, 1-bit. Indicates a locked transaction.
- **HREADY**: Input, 1-bit. Indicates previous transfer is completed.
- **HRESP**: Input, 1-bit. Response signal indicating success/error.

BIU Bus (Core Interface)

- **biu_stb_i**: Input, 1-bit. Core request strobe.
- **biu_stb_ack_o**: Output, 1-bit. Indicates that the core strobe request is acknowledged.
- **biu_d_ack_o**: Output, 1-bit. Acknowledgment for data phase.
- **biu_adri_i**: Input, 32-bit. Address input from the core.
- **biu_adro_o**: Output, 32-bit. Address output to the core.
- **biu_size_i**: Input, derived size. Transfer size indicator.
- **biu_type_i**: Input, derived type. Burst type descriptor.
- **biu_prot_i**: Input, derived prot. Protection descriptor.
- **biu_lock_i**: Input, 1-bit. Lock signal for burst.
- **biu_we_i**: Input, 1-bit. Write-enable signal.
- **biu_d_i**: Input, 32-bit. Data input from the core.
- **biu_q_o**: Output, 32-bit. Data output to the core.
- **biu_ack_o**: Output, 1-bit. Transfer acknowledge to core.
- **biu_err_o**: Output, 1-bit. Transfer error to core.
- **biu_tagi_i**: Input, as per size. TAG input from core.
- **biu_tago_o**: Output, as per size. TAG output to core.

Internal Signals

- **cross1kB**: Detects if a 1kB address boundary is crossed.
- **incr_burst**: Indicates ongoing incremental burst.
- **User-defined Signals**:

  - burst_cnt: 4-bit counter for burst length tracking.
  - data_ena, ddata_ena: Data enable signals indicating valid data phases.
  - biu_di_dly: Delayed data signal.
  - tag: Holds the current tag being processed.

Functionality

Sequential Logic

- Core of the module operates based on a clock-driven state machine.
- On HRESETn, initializes all outputs and internal variables to default states.
- The module uses synchronized process blocks for resetting and maintaining different signals based on clocks and strobes.

Combinational Logic

- Functions transform core signals into AHB-compatible signals (e.g., transfer size to HSIZE, burst type to HBURST).
- Functions also calculate the next address and detect special conditions like 1kB boundary crossing.

State Machines or Control Logic

- State transitions driven by HCLK and synchronized with HREADY signal.
- Handles transaction sequences like starting a burst, handling errors, or completing bursts.
- Maintains different signals depending on the current state (e.g., AHB DEFAULT, TRANSACTION, INCREMENTAL BURST).

Instantiations

- Internal logic and functions act as implicit components.
- There are no explicit sub-module instantiations within biu_ahb3lite.

Inter-Module Connections

- The biu_ahb3lite module interfaces between the core and AHB3 Lite bus, bridging signal transformations and protocol translations.
- Core signals are interpreted, transformed, and issued as AHB bus transactions.

This comprehensive description provides a functional understanding of the biu_ahb3lite module pivotal for further design iterations or verification.