AXI_SP32B1024 Specification

Introduction

The `AXI_SP32B1024` module is a memory interface controller designed to work with an AXI-4 Slave interface. It facilitates read and write operations to a memory block using the AXI protocol, which is widely used in high-performance systems for its efficient data handling capabilities. The module supports 32-bit data transactions and addresses a memory space of 1024 locations. It adheres to the AXI-4 protocol standards, ensuring compatibility with other AXI-compliant devices.

Architecture

The `AXI_SP32B1024` module is structured to handle AXI-4 transactions and interface with a memory block. The architecture includes:

- **Top-Level Structure**: The module interfaces with an AXI-4 Slave and a memory block. It manages address, data, and control signals to facilitate read and write operations.
- **Key Operations**:
  - **Address Handling**: Captures write and read addresses from AXI transactions.
  - **Data Handling**: Manages data input and output between the AXI interface and memory.
  - **Control Logic**: Implements state machines for read and write operations, ensuring proper sequencing and signal assertion.
- **Implementation Details**:
  - The module operates on a single clock domain (`CLK`) and uses a negative edge-triggered reset (`RST`).
  - It uses internal registers to manage data paths and control signals, ensuring proper timing and synchronization.

Interface

Signal	Width	In/Out	Description
CLK	1	In	Clock signal for synchronous operations.
RST	1	In	Active-low reset signal, negative edge-triggered.
axi_awvalid	1	In	Write address valid signal from AXI master.
axi_awready	1	Out	Write address ready signal to AXI master.
axi_awaddr	32	In	Write address from AXI master.
axi_awprot	3	In	Write protection type from AXI master.
axi_wvalid	1	In	Write data valid signal from AXI master.
axi_wready	1	Out	Write data ready signal to AXI master.
axi_wdata	32	In	Write data from AXI master.

Signal	Width	In/Out	Description
axi_wstrb	4	In	Write strobe signal indicating valid bytes in `axi_wdata`.
axi_bvalid	1	Out	Write response valid signal to AXI master.
axi_bready	1	In	Write response ready signal from AXI master.
axi_arvalid	1	In	Read address valid signal from AXI master.
axi_arready	1	Out	Read address ready signal to AXI master.
axi_araddr	32	In	Read address from AXI master.
axi_arprot	3	In	Read protection type from AXI master.
axi_rvalid	1	Out	Read data valid signal to AXI master.
axi_rready	1	In	Read data ready signal from AXI master.
axi_rdata	32	Out	Read data to AXI master.
Q	32	In	Data output from memory block.
CEN	1	Out	Chip enable signal for memory block.
WEN	1	Out	Write enable signal for memory block.
A	10	Out	Address signal to memory block.
D	32	Out	Data input to memory block.

Timing

- **Latency**: The module introduces a latency of a few clock cycles for read and write operations due to the state machine transitions and signal propagation.
- **Signal Behavior**:
  - The module samples inputs on the rising edge of the clock (`CLK`).
  - Outputs are updated based on the state machine transitions, with read data (`axi_rdata`) and write responses (`axi_bvalid`) being synchronized to the clock.

Usage

To use the `AXI_SP32B1024` module:

1. **Initialization**: Ensure the `RST` signal is asserted low to reset the module. Release `RST` to begin normal operation.
2. **Write Operation**:
   - Assert `axi_awvalid` with a valid address on `axi_awaddr`.
   - Assert `axi_wvalid` with valid data on `axi_wdata` and appropriate byte enables on `axi_wstrb`.
   - Monitor `axi_awready` and `axi_wready` to ensure the module is ready to accept the address and data.
   - After the write operation, check `axi_bvalid` to confirm the write response.
3. **Read Operation**:
   - Assert `axi_arvalid` with a valid address on `axi_araddr`.

- Monitor `axi_arready` to ensure the module is ready to accept the address.
- After the read operation, check `axi_rvalid` and read the data from `axi_rdata`.

The module handles all necessary control signals internally, ensuring proper sequencing and data integrity during AXI transactions.

Functional Description (Generated by funcgen)

Functional Description of Verilog Modules

Module: AXI_SP32B1024

Source File: AXI_SP32B1024.v

Purpose

The `AXI_SP32B1024` module serves as an interface between an AXI-4 bus and an internal memory. It translates AXI signals to memory control signals for read and write operations, facilitating data exchange between the AXI interface and a 1024-entry, 32-bit wide memory.

Parameters

No parameters are specified for this module.

Ports

Port Name	Direction	Bit-width	Description
CLK	Input	1-bit	Clock input that drives the operation of the module.
RST	Input	1-bit	Active-low reset signal to initialize internal states.
axi_awvalid	Input	1-bit	Indicates that the address for writing is valid.
axi_awready	Output	1-bit	Asserted to indicate that the module is ready to accept a write address.
axi_awaddr	Input	32-bit	Address for the write transaction.
axi_awprot	Input	3-bit	Protection type for write transactions (unused in this context).
axi_wvalid	Input	1-bit	Indicates that the write data is valid.
axi_wready	Output	1-bit	Asserted to indicate readiness to accept write data.
axi_wdata	Input	32-bit	Write data bus.

Port Name	Direction	Bit-width	Description
axi_wstrb	Input	4-bit	Byte enable signals for write strobes.
axi_bvalid	Output	1-bit	Asserted to indicate that a write response is available.
axi_bready	Input	1-bit	Indicates ready-to-accept the write response.
axi_arvalid	Input	1-bit	Indicates that the read address is valid.
axi_arready	Output	1-bit	Asserted to indicate readiness to accept a read address.
axi_araddr	Input	32-bit	Address for the read transaction.
axi_arprot	Input	3-bit	Protection type for read transactions (unused in this context).
axi_rvalid	Output	1-bit	Asserted to indicate that read data is available.
axi_rready	Input	1-bit	Indicates ready-to-accept the read data.
axi_rdata	Output	32-bit	Read data bus.
Q	Input	32-bit	Data output from memory.
CEN	Output	1-bit	Chip enable for memory, active-low.
WEN	Output	1-bit	Write enable for memory, active-low.
A	Output	10-bit	Address bus for memory access.
D	Output	32-bit	Data input to memory.

Internal Signals

Signal Name	Type	Bit-width	Description
A	Reg	10-bit	Register for capturing the memory address.
DP	Reg	32-bit	Register for holding write data internally.
reading1	Reg	1-bit	Signal to control the first stage of read operation.
reading2	Reg	1-bit	Signal to control the final stage of read operation.
writting1	Reg	1-bit	Signal to control the initial stage of write operation.
writting2	Reg	1-bit	Signal to control the mid-stage of write operation.
writting3	Reg	1-bit	Signal to control the final stage of write operation.

Functionality

The module translates AXI transactions into operations on an internal memory. It handles both read and write transactions using sequential logic to manage

handshakes and data paths:

Sequential Logic

- **Address and Data Capture**: On falling edge of the clock (`CLK`), if reset (`RST`) is not active, the module captures write addresses (`axi_awaddr`) or read addresses (`axi_araddr`) and stores write data (`axi_wdata`) into internal registers if valid signals are asserted.
- **Read Control Logic**: Synchronous logic on the rising edge of `CLK` is used to manage read transaction stages, ensuring data validity (`axi_rvalid`) aligns with handshake readiness (`axi_rready`).
- **Write Control Logic**: Similar logic manages write phases, producing `axi_bvalid` to acknowledge write completions as per AXI specifications.
- **Memory Control**: Control signals for the memory (`CEN`, `WEN`) are updated based on operation flags, only enabling (active-low) when appropriate transactions are moving through valid states.

Combinational Logic

- Immediate setting of `axi_awready`, `axi_arready`, and `axi_wready` to true allows the module to always be ready for the address phase.
- Dynamic data translation using write strobe (`axi_wstrb`) to select bytes from either direct data input or memory data (`Q`), before writing to the memory (`D`).

Instantiations

- **SP32B1024**: An internal memory block (commented out but structured) purportedly modelled for interfacing with this module. The internal signals (`Q`, `CLK`, `CEN`, `WEN`, `A`, `D`) connect appropriately.

Inter-Module Connections

This module works as a stand-alone component facilitating connections with higher-level AXI infrastructure and lower-level memory. It bridges these subsystems, managing signal control for transaction data paths from AXI to memory.

- Hierarchical connectivity could involve higher-level systems generating AI transactions feeding into this block, with further integration into broader memory controller hierarchies.