AU-B2000: AMBA AHB Bus Slave Core

The AU-B2000 AMBA AHB Bus Slave provides the bus slave function for the AMBA AHB Bus. It is used to connect a device in the user's logic to the AMBA Bus. It accepts AMBA Bus transactions that target the device and only passes the transactions that require data transfers, to the device in the user's logic. AMBA Bus transactions that do not transfer data- IDLE and BUSY, are handled entirely within the AMBA Slave. Additionally, the AMBA Slave also performs other AMBA Bus protocol functions so that the device does not have to implement these functions. The AMBA Bus Slave is available as a synthesizable Verilog model from Aurora VLSI, Inc. Contact CustomerService@auroravlsi.com.

The AMBA Slave can interface to either a 32 bit or 64 bit AMBA AHB Bus. A Verilog parameter indicates the AMBA Bus width for the AMBA Slave.

When an AMBA Bus transaction targets a device, if the AMBA Bus transaction does not require a data transfer (IDLE and BUSY AMBA Bus transactions), the AMBA Slave responds by driving the OKAY response onto the AMBA bus, without any assistance from the device in the user's logic. When an AMBA Bus transaction that is targeting a device does require a data transfer, the AMBA Slave repackages the AMBA Bus transaction into a request that it sends to the device. If the device is not ready to accept the request, the AMBA Slave drives a RETRY response onto the AMBA Bus. If the device accepts the request, then when the device can transfer the data, it issues a response to the AMBA Slave and the data is transferred between the AMBA Slave and the device. The AMBA Slave handles all AMBA Bus protocol requirements when driving the device response onto the AMBA Bus and transferring device data to/from the AMBA Bus.

The AMBA Slave supports all required AMBA AHB Bus features including all AMBA burst and wrapping types, AMBA sizes up to the AMBA Bus width, and all AMBA Bus responses. Informational signals with each request from the AMBA Slave to the device, indicate burst size, wrapping, and other burst information so that the device may prefetch read data and/or pack write data. The data transfer size and whether it is a read or write request, are also part of each request sent to the device. The ID number of the AMBA Master that initiated the request is also supplied to the device so that the device can issue a SPLIT response and remove the split at a later time.

The AMBA Slave can accept a response from the device in the same cycle as it issues a request to the device. This ensures a slave response on the AMBA Bus with no wait states, and therefore highest system performance. Alternatively, the device can respond up to fifteen cycles after receiving a request from the AMBA Slave before the AMBA Slave drives a RETRY response onto the AMBA Bus.

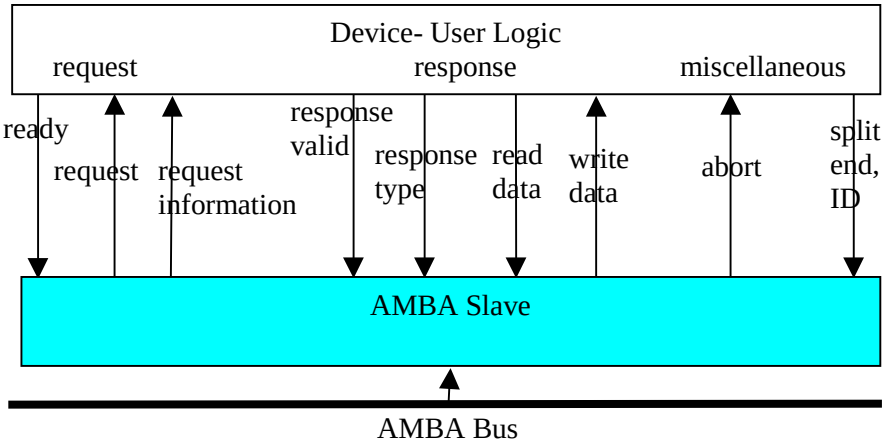
The device transfers data to and from the AMBA Slave on two data buses. One data bus transfers read data and the other data bus is dedicated to write data. When the data transfer size is less than the AMBA Bus width, the AMBA Slave performs all data alignment functions. AMBA Bus data can be either big endian or little endian, as indicated by a Verilog parameter. Because the AMBA Slave handles data alignment, the device does not need to know the AMBA Bus endianness.

When the device response is something other than OKAY- ERROR, RETRY, or SPLIT, data is still transferred. This feature can be used to implement early burst termination by the device, as opposed to by the AMBA Master.

AMBA Bus Slave features are summarized:

- AMBA AHB Bus slave function
- 32 bit or 64 bit AMBA AHB Bus- user configurable
- Handles all IDLE and BUSY transactions without any user logic
- Automatic RETRY when the user's device is not ready
- Implements AMBA Bus timeout and RETRY response
- Supports all required AMBA AHB Bus features
- Informational signals to support read data prefetching, write data packing, and splits
- Same cycle device request/response is supported for highest throughput
- Independent read and write data buses from/to the device
- Handles all data alignment for data transfer sizes less than the AMBA Bus width
- User configurable for big or little endian AMBA Bus- transparent to the device
- Data transferred regardless of response type

The AMBA Slave is used to connect a device that can be an AMBA Bus transaction target in the user's logic, to the AMBA Bus. This is shown below.



The AMBA Slave issues requests to the device along with the request information. The request and request information is valid for one cycle. If the device is ready during this request cycle, it captures the request information. If the device is not ready, the AMBA Slave drives a RETRY response onto the AMBA Bus.

The device indicates that it is ready to transfer the data associated with a request by sending a response to the AMBA Slave. On read requests, the device sends the read data to the AMBA Slave along with the response. On write requests, the device captures write data from the AMBA Slave in the cycle after the response.

If the device does not respond within fifteen cycles after capturing a request, the AMBA Slave drives a RETRY response onto the AMBA Bus, and the request is aborted. If the device responds with a SPLIT response, when it subsequently ends the split it identifies the AMBA split master.

The core is delivered as a synthesizeable RTL Verilog model. Deliverables include:

- RTL Verilog source code model of the core
- Verilog testbench and test cases
- Synthesis scripts examples
- Complete detailed documentation and training class notes