ARM®

PrimeCell® Infrastructure AMBA™ 3 AXI Upwards-synchronizing Bridge (BP134) Revision: r0p0 **Technical Overview**

This Technical Overview describes the functionality of the AXI upwards-synchronizing bridge in the following sections:

1 Preliminary material

1.1 Release information

Table 1 lists the changes to this document.

Table 1 Change History

Date	Issue	Confidentiality	Change
28 June 2005	A	Non-Confidential	First issue for r0p0

1.2 Proprietary notice

1.3 Confidentiality status

This document is Non-Confidential. This document can only be distributed in accordance with the terms set out in your license agreement.

1.4 Product status

The information in this document is final, that is for a developed product.

1.5 Web address

`http://www.arm.com`

2 About the AXI upwards-synchronizing bridge

The AXI upwards-synchronizing bridge, SyncUpAxi, enables a slow AXI clock domain to communicate with a faster one using a common clock select. Figure 1 shows SyncUpAxi used in a system where an AXI master is running at 100MHz and an AXI slave is running at 200MHz.

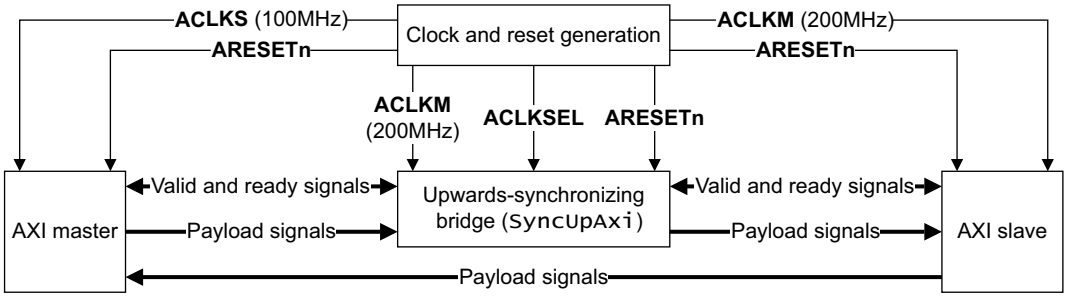

Figure 1 upwards-synchronizing bridge block diagram

The SyncUpAxi 1:n bridge has the following features:

- 1:n clock synchronizing using a common clock select input where n >= 1
- **CSYSREQ**, **CSYSACK**, and **CACTIVE** are synchronized on the low-power interface
- synchronizes the channel flow control signals on the AXI interface:
  - **AW** Write address channel.
  - **W** Write data channel.
  - **B** Write response channel.
  - **AR** Read address channel.
  - **R** Read data channel.
- only the handshake signals are synchronized on the write response, and read data channels
- buffered synchronization of the write address, write data, and read address channels
- maximum latency of one clock period in the slow clock domain
- the HDL code is supplied as Verilog.

3 Functional description

Figure 2 shows a block diagram of the major internal component blocks.

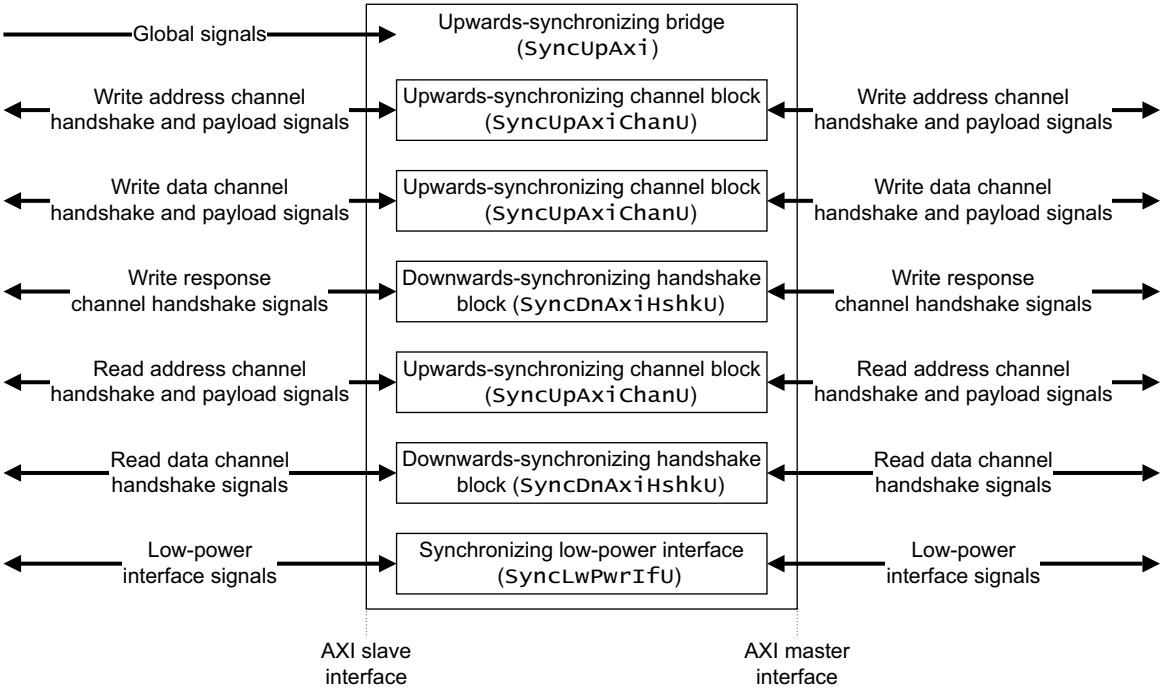

Figure 2 AXI upwards-synchronizing bridge components

The major internal component blocks are:

`SyncDnAxiHshkU`

This is an AXI downwards-synchronizing handshake block. The inputs and outputs are not registered. The block synchronizes the valid and ready signal handshakes in a system where the transfer source component is clocked at a higher frequency than the transfer destination component.

Note

The clock domains must be synchronously related so that the clock select occurs over the coincidental rising edges.

The `SyncDnAxiHshkU` block is used for the AXI channels that Figure 2 shows.

`SyncUpAxiChanU`

This is an AXI upwards-synchronizing channel block. The inputs and outputs are not registered. The block synchronizes the valid and ready signal handshakes in a system where the transfer source component is clocked at a lower frequency than the transfer destination component. A zero latency buffer synchronizes the channel payload signals.

Note

The clock domains must be synchronously related so that the clock select occurs over the coincidental rising edges.

The `SyncUpAxiHshkU` block is used for the AXI channels that Figure 2 on page 4 shows.

`SyncLwPwrIfU`

The AXI low-power signaling mechanism provides a coherent control method for a system-level power-down mode.

For systems that require multiple related clock domains, a low-power mode-aware component can sit in a different clock domain to that of the system low-power controller.

The synchronizing low-power interface, `SyncLwPwrIfU`, ensures that transfers across these boundaries can complete. The interface re-times the **CSYSREQ**, **CSYSACK**, and **CACTIVE** signals between the clock domains.

All low-power channel signals are re-timed to supply a definite timing point for implementation, and to ensure satisfactory synchronization in both domains.

See the *AMBA AXI Protocol Specification* for more information.

3.1 Interface attributes

The following tables list the master and slave interface attributes for the AXI upwards-synchronizing bridge:

- Table 2 on page 6
- Table 3 on page 6.

Table 2 Master interface attributes

Attribute	Description	Value
Combined issuing capability	The maximum number of active transactions that a master can generate	Master-dependent
Read ID capability	The maximum number of different **ARID** values that a master can generate for all active read transactions at any one time	Master-dependent
Read ID width	The number of bits in the **ARID** bus	Master-dependent
Read issuing capability	The maximum number of active read transactions that a master can generate	Master-dependent
Write ID capability	The maximum number of different **AWID** values that a master can generate for all active write transactions at any one time	Master-dependent
Write ID width	The number of bits in the **AWID** and **WID** buses	Master-dependent
Write issuing capability	The maximum number of active write transactions that a master can generate	Master-dependent

Table 3 Slave interface attributes

Attribute	Description	Value
Write acceptance capability	The maximum number of active write transactions that a slave can accept.	Slave-dependent
Read acceptance capability	The maximum number of active read transactions that a slave can accept.	Slave-dependent
Combined acceptance capability	The maximum number of active transactions that a slave can accept. This must be specified if read and write address storage is combined.	Slave-dependent
Write interleave depth	The number of active write transactions for which the slave can receive data. This is counted from the earliest transaction.	Slave-dependent
Read data reorder depth	The number of active read transactions for which a slave can transmit data. This is counted from the earliest transaction.	Slave-dependent

Note

The master-dependent and slave-dependent values in the Value column of Table 2 on page 6, and Table 3 on page 6 indicate that the bridge adopts the attribute value of the master or slave that the interface is connected to.

4 Physical data

This section describes:

- *AC characteristics*
- *Gate count*.

4.1 AC characteristics

The upwards-synchronizing bridge adheres to the following timing guidelines. The figures relate to the percentage of clock cycle permitted for each function:

- inputs to registers must be valid for 40% prior to the rising edge of the fast clock
- outputs from registers must be valid for 20% after the rising edge of the fast clock
- combinatorial paths must not take longer than 10% of the complete fast clock cycle.

Note

There are combinatorial paths from **ACLKSEL** to:

- **AWVALIDM**
- **ARVALIDM**
- **WVALIDM**
- **BREADYM**
- **RREADYM**.

The delay on these paths is a maximum of 10% of the clock cycle.

Note

If the component is used only for clock ratios greater than 1:1, you can relax these constraints accordingly.

Timing characteristics are confirmed by performing synthesis on the block using the slow-slow process point of the Artisan SAGE HS library for the TSMC CL013G process at a target speed of 200MHz.

4.2 Gate count

Total gate count is approximately 2500 NAND2x1 equivalent gates with respect to the library that *AC characteristics* describes.

Note

The gate count estimate does not include scan logic.

5 Signal descriptions

Figure 3 shows the AXI upwards-synchronizing bridge signal connections.

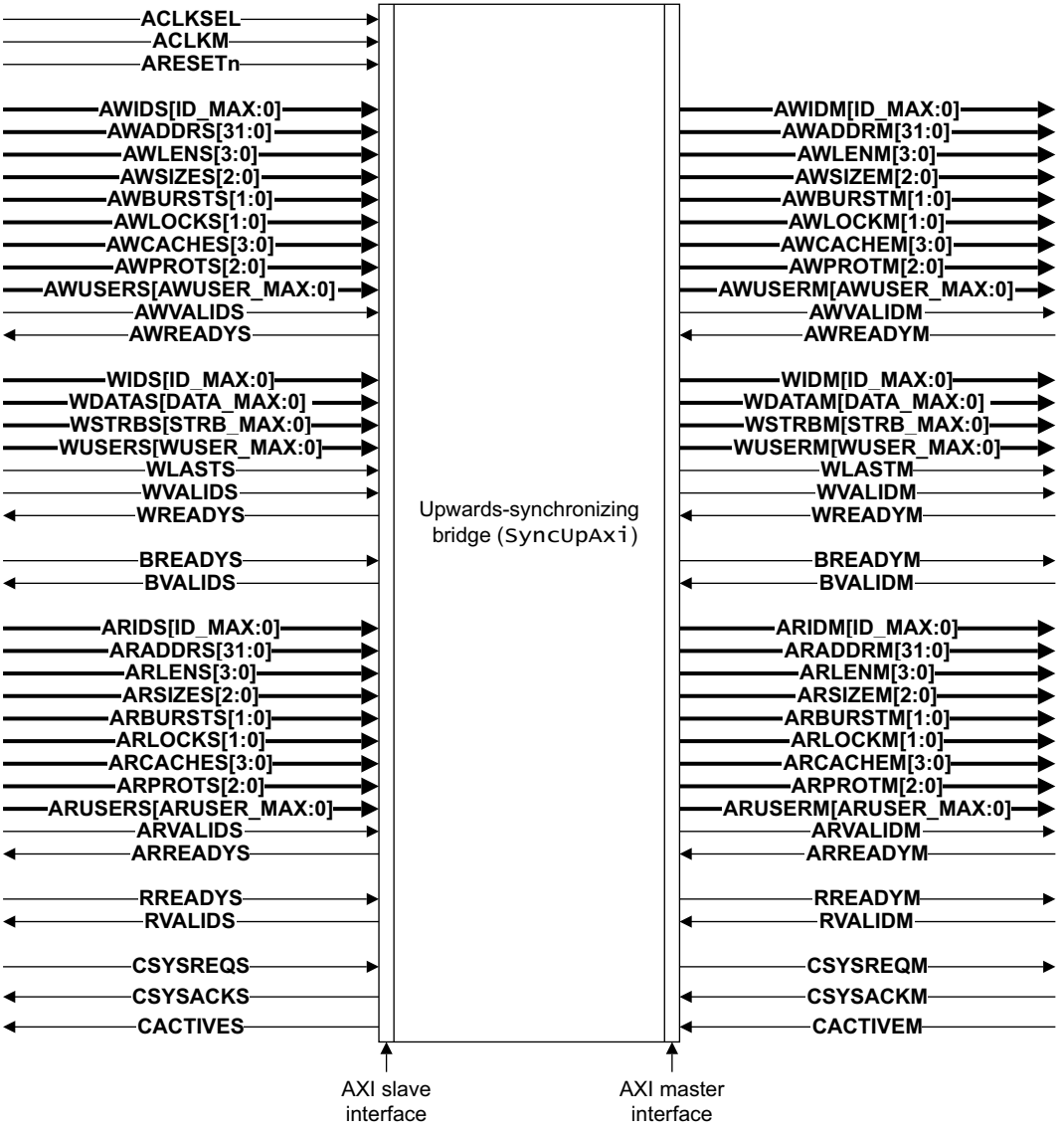

Figure 3 Upwards-synchronizing bridge signal connections

Note

In Figure 3 on page 10:

- The read channel, write channel, and low-power interface signals are standard AMBA AXI signals that the *AMBA AXI Protocol Specification* describes. The signal names are appended with:
  - the letter **M** for signals that connect to the component master interface
  - the letter **S** for signals that connect to the component slave interface.
- The scan signals are not shown.

Table 4 lists the non-standard AXI and scan signals.

Table 4 Non-standard AXI and scan signals

Name	Type	Source/ destination	Description
ACLKM	Input	Clock source	Clock signal from the faster clock domain
ACLKSEL	Input	Clock source	Select signal that indicates relationship between fast and slow clocks
ARUSERM	Output	Scan logic	AR channel user sideband signal to the fast clock domain
ARUSERS	Input	Scan logic	AR channel user sideband signal from the slow clock domain
AWUSERM	Output	Scan logic	AW channel user sideband signal to the fast clock domain
AWUSERS	Input	Scan logic	AW channel user sideband signal from the slow clock domain
SCANENABLE	Input	Scan logic	Scan mode enable
SCANINACLKM	Input	Scan logic	Scan chain clock input
SCANOUTACLKM	Output	Scan logic	Scan chain clock output
WUSERM	Output	Scan logic	AW channel user sideband signal to the fast clock domain
WUSERS	Input	Scan logic	AW channel user sideband signal from the slow clock domain

