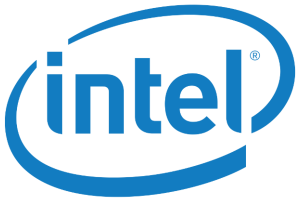


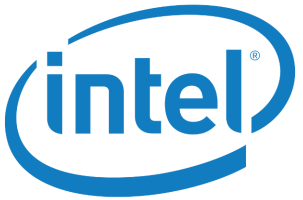
AMBA and AXI4

Peripheral Devices for Embedded Systems

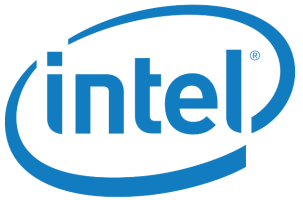
Rafal Kapela

June 26, 2016

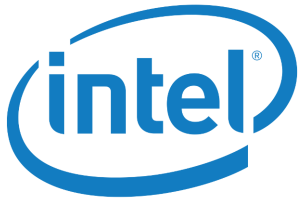
Outline



Outline



What is AXI?

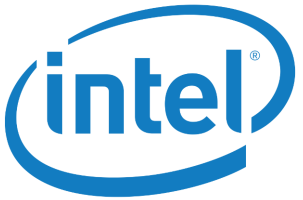


The ARM Advanced Microcontroller Bus Architecture (AMBA) is an open-standard, on-chip interconnect specification for the connection and management of functional blocks in system-on-a-chip (SoC) designs.

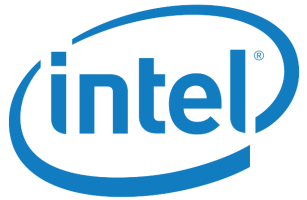
AXI, is targeted at high performance, high clock frequency system designs and includes features that make it suitable for high speed sub-micrometer interconnect:

- separate address/control and data phases,
- support for unaligned data transfers using byte strobes,
- burst based transactions with only start address issued,
- issuing of multiple outstanding addresses with out of order responses,
- easy addition of register stages to provide timing closure.

Outline



Handshaking concept



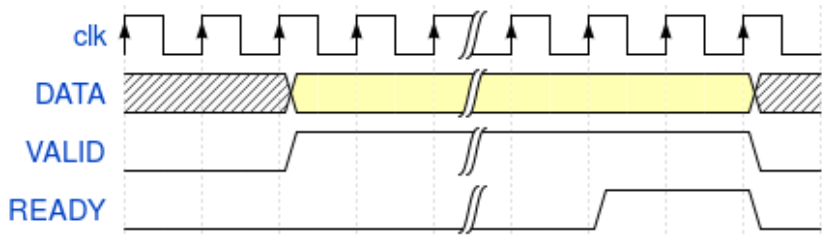
All AXI Channels Use A Basic "VALID/READY" Handshake.

- SOURCE asserts and holds VALID when DATA is available
- DESTINATION asserts READY if able to accept DATA
- DATA transferred when VALID and READY $= 1$
- SOURCE sends next DATA (if an actual data channel) or deasserts VALID
- DESTINATION deasserts READY if no longer able to accept DATA

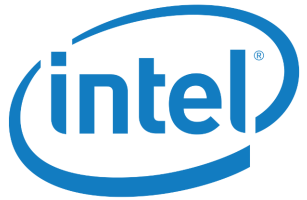
Handshaking concept

AXI uses a valid/ready handshake acknowledge

- Each channel has its own valid/ready:
  - Address (read/write)
  - Data (read/write)
  - Response (write only)
- Flexible signaling functionality
  - Inserting wait states
  - Always ready
  - Same cycle acknowledge



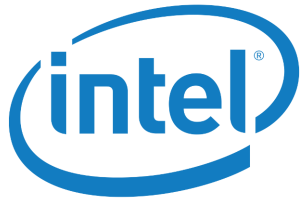
AXI Interconnect



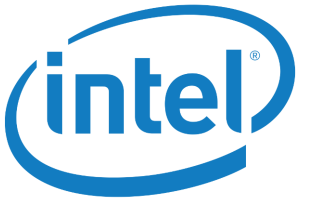
axi_interconnect component:

- Highly configurable:
  - Pass Through
  - Conversion Only
  - N-to-1 Interconnect
  - 1-to-N Interconnect
  - N-to-M Interconnect – full crossbar
  - N-to-M Interconnect – shared bus structure
- Decoupled master and slave interfaces
- Xilinx AXI Reference Guide(UG761)

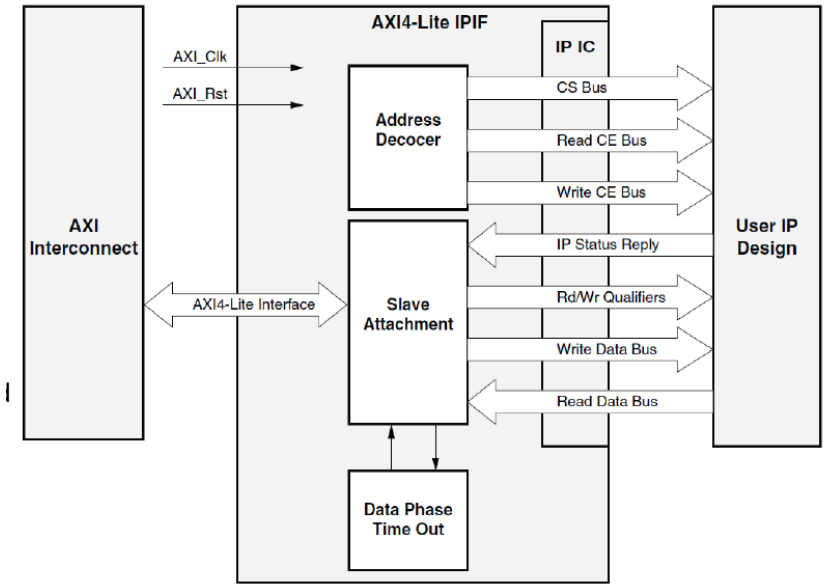
Outline



AXI Lite IPIF – Block Diagram



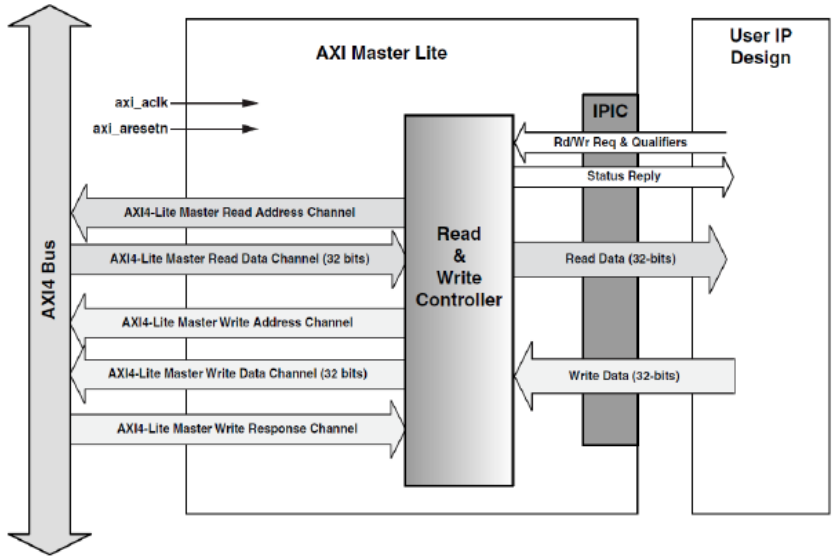
- Basic services
  - Slave attachment
  - Address decoding
  - Timeout generation
  - Byte strobe forwarding
- Optional services
  - Master user logic
  - Soft reset core
  - User logic software registers, and Timeout logic inclusion



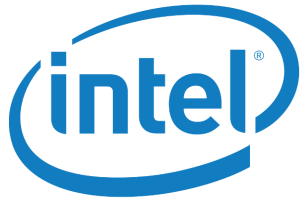
AXI4 Lite Master – Block Diagram

AXI4 Lite mastering capability

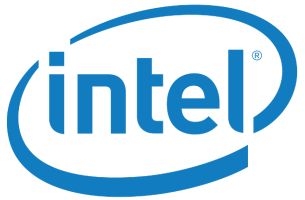
- Single data phase only (one to four bytes)
- Only 32-bit data width



Outline



AXI Interconnect

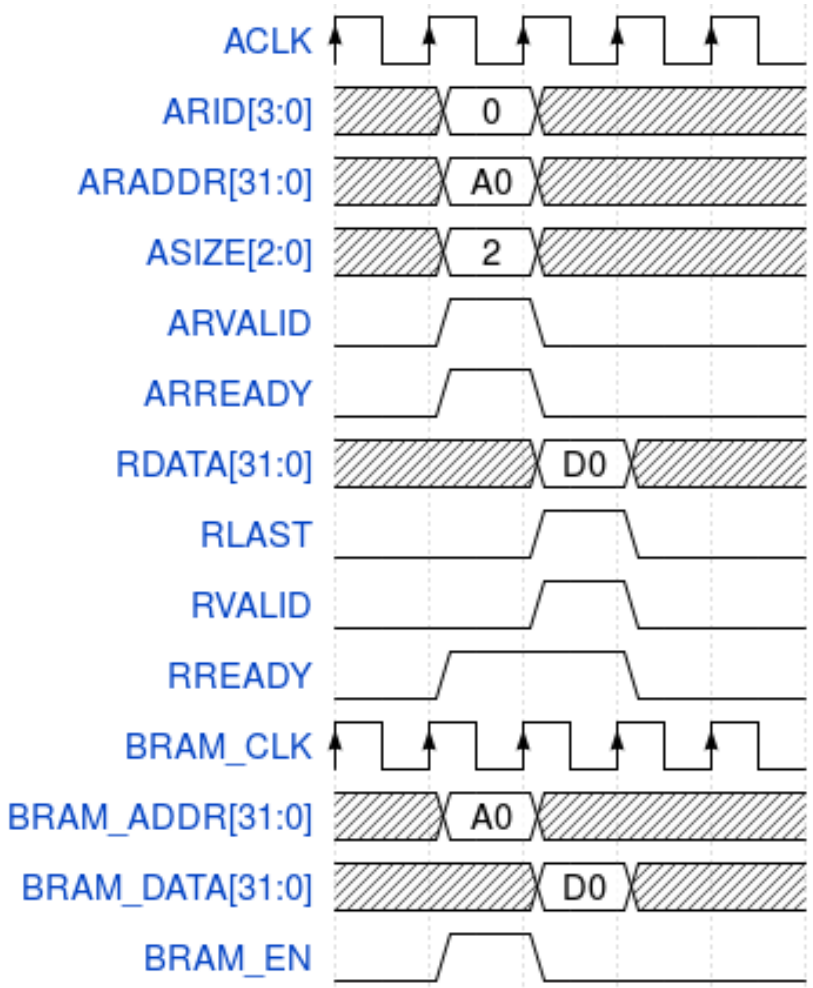


Slave IPIC duties and configurable options:

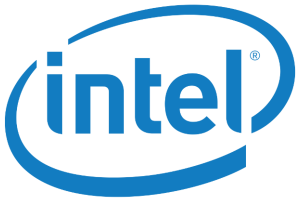
- Address decode and acknowledge
- One or more address spaces
  - Memory interface; that is, chip enable
  - User registers
- Single or burst data phase acknowledgement
- Software reset/MIR register
- Read FIFOs
- Automatic timeout on user slave logic
- Your custom slave attachments

AXI Read transaction

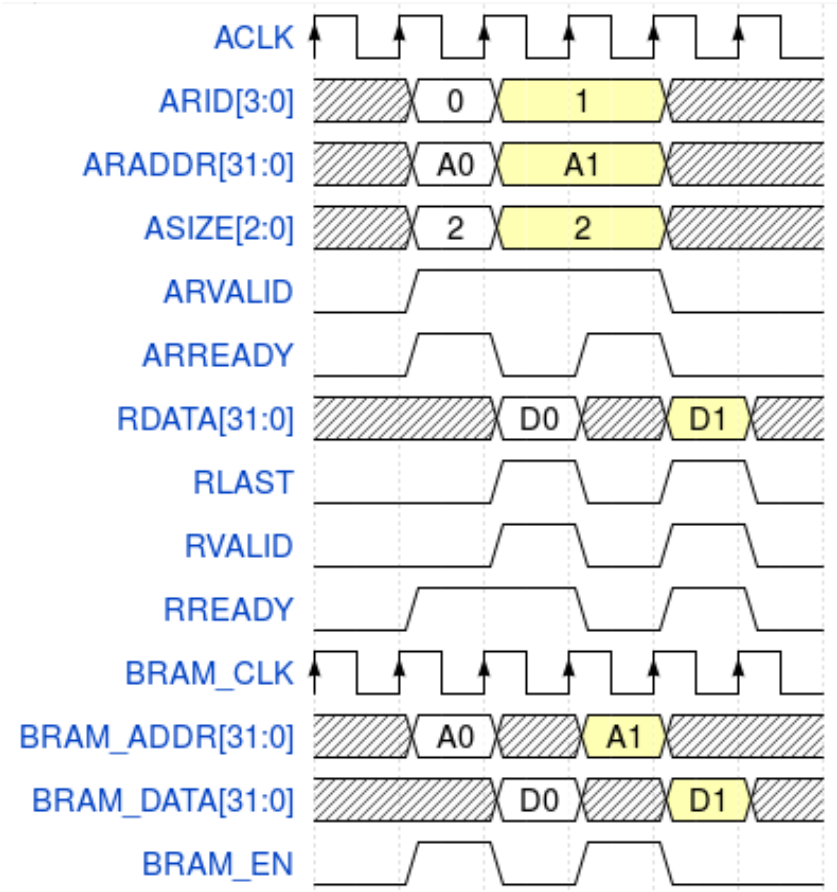
- Single data phase only (one to four bytes)
- Only 32-bit data width



AXI Multiple Read transaction

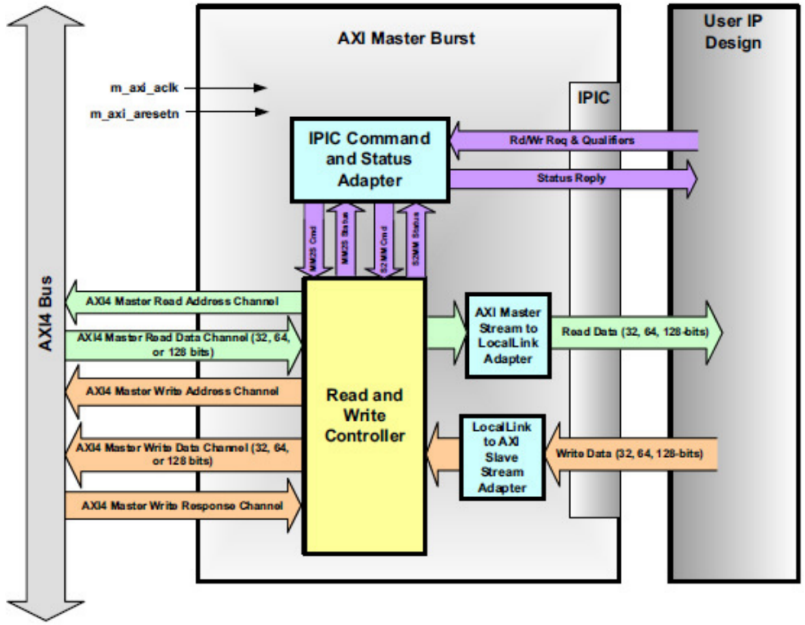


- Multiple read transactions are identified by different read IDs
  - For a burst transaction only one read ID would have been used
- Separate RLAST for corresponding read transactions
- ARSIZE=2 indicates entire word is being read



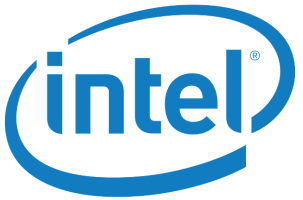
AXI Read transaction

- Parameterizable data width (32, 64, 128)
- Data burst (16, 32, 64, 128, 256 data beats)



Resources

If you want to gain some knowledge by your own...

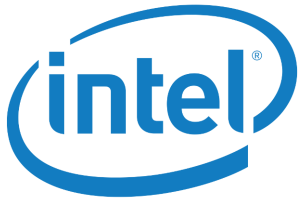


Wikipedia – Embedded system

`http://en.wikipedia.org/wiki/Embedded_system`

Xilinx resources

`http://www.xilinx.com/support.html`



Questions ?

Rafal Kapela
rafal.kapela@put.poznan.pl