Comparative Analysis of AMBA 2.0 and AMBA 3 AXI Protocol-Based Subsystems

Meni Jayaswal
Corporate Applications Engineer
Synopsys
Agenda

- Introduction
- AMBA® 2.0 and AMBA 3 AXI™ features
- Two subsystem example designs
- Compare and contrast QoR results
- Synopsys DesignWare® IP solutions for AMBA
- Summary
Introduction

Criteria for making system level choices

• Choosing the right bus for your system

• Peripherals selection

• Performance requirements based on your application
AMBA 2.0: Key Features

- **Enables high performance operation**
  - Burst transfers
  - Split transactions
  - Single-cycle bus master handover
  - Single-clock edge operation
  - Wider data bus configurations (64/128 bits).

- **Ensures system efficiency**
  - Provides high bandwidth operation
  - High data throughput

- **Modular system design**
  - Single channel implementation
  - Supports multiple bus masters

- **Ease of extension of subsystem**
  - Simpler design
  - Efficient bridging between AHB and APB
  - Legacy IP
AMBA 3 AXI: Key Features

- **Channels Enable High Performance**
  - 5 unidirectional channels; Separate Address/Control and data phases
  - Out of order transaction completion

- **Efficient use of bus bandwidth**
  - All transactions are burst based (only start address)
  - Variable burst lengths, data widths supported
  - Multiple Outstanding transactions
  - Data Interleaving and unaligned transfers

- **Timing Closure**
  - Register slices can be inserted across any channel
  - Enables max frequency by tweaking channel latency

- **Flexibility of Implementation**
  - Variety of choices for Interconnect Implementation (SASD, SAMD, MAMD)

- **Support for Complex Transactions**
  - Exclusive and Locked accesses, secure transactions
# AMBA 2.0 AHB vs AMBA 3 AXI

<table>
<thead>
<tr>
<th>AMBA 2.0 AHB Protocol</th>
<th>AMBA 3 AXI Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed pipeline for address and data transfers</td>
<td>Five independent channels for addr/data and response; out of order data</td>
</tr>
<tr>
<td>Separate address for every data item</td>
<td>Burst based—one address per burst</td>
</tr>
<tr>
<td>Only one transaction at a time</td>
<td>Multiple outstanding transactions; simultaneous reads and writes</td>
</tr>
<tr>
<td>Simpler interconnect (SASD), less area and routing congestion</td>
<td>Complex interconnect (SASD, MASD, MAMD)</td>
</tr>
<tr>
<td>Commodity microprocessor; lot of legacy IP already available; ease of extension; reuse</td>
<td>Newer architecture, still in adoption phases</td>
</tr>
<tr>
<td>Bidirectional link with complex timing relationships</td>
<td>Each channel is unidirectional except for single handshake for return path</td>
</tr>
<tr>
<td>Hard to isolate timing; Limits frequency of operation</td>
<td>Register slices isolate timing; Frequency scales with pipelining</td>
</tr>
<tr>
<td>No support for unaligned and exclusive accesses and security</td>
<td>Native support for unaligned and exclusive accesses and security support</td>
</tr>
</tbody>
</table>
Example Design 1: PDA

High Bandwidth Bus
- System Processor
- USB Host
- DMA Controller
- NAND Controller
- Power Management
- Interrupt Controller
- Memory Controller
- USB Device
- Camera Interface
- LCD Controller

Low Bandwidth Bus
- I2C Interface
- SPI
- UART
- Real Time Clock
- GPIO
- Timer
PDA: System Requirements

Requirements for a mid-range, slim form factor PDA

- Capable of running a light weight mobile OS
- Low power consumption
- NAND flash based storage
- USB interface
- 2 mega pixel camera
- LCD screen, 240 * 320 @ 16-bit color depth.
  - Requires a bandwidth of 0.4 Mb/s

What does this mean for our subsystem?

- High performance bus required for running a light weight mobile OS
- There are no latency critical, or ultra high bandwidth applications
- Space in the device is at a premium => die area is a critical factor
- Using small battery and running a light weight mobile OS places power consumption at a premium => power optimization is a critical factor
Architecture Choice : Why AMBA 2.0?

- Analysis shows we require 5 masters and 11 slaves.

- Results show a 5 master 11 slave DW_ahb will synthesize @166Mhz (90nm).

- This gives a peak bus bandwidth of $166 \times 32 = 5.31$ Gb/s for a 32-bit system.

- This allows sufficient bandwidth for our mobile OS and bus traffic for our camera, LCD screen and other system elements.
How are we going to build these subsystems?

• Multiple DesignWare AMBA-based components:
  coreAssembler

• Configure, synthesize and verify a subsystem of components

• Auto-connection of AXI, AHB, APB interfaces

• coreAssembler features available when using DW IP are:
  • Automated address map generation
  • Automated subsystem synthesis scripts creation
  • Automated subsystem testbench creation (includes DW Verification IP interface models/monitors)
  • Automated subsystem stimulus creation (AXI or AHB based ping tests)
Example Design 1: PDA

AHB

- System Processor
- USB Host
- DMA Controller
- NAND Controller
- Power Management

ICTL

- Memory Controller
- USB Device
- Camera Interface
- LCD Controller

AHB to APB Bridge

APB

- I2C Interface
- SSI
- UART
- Real Time Clock
- GPIO
- Timer

DesignWare IP

Synopsys

ARM Developers’ Conference & Design Pavilion 2007

Confidential
PDA : Subsystem Implementation in coreAssembler

- Interfaces for third party IP are exported.
- A 20% i/o delay is used at all access points to the AHB interconnect.
# PDA : Speed and Area Results

<table>
<thead>
<tr>
<th>Library</th>
<th>Frequency (MHz)</th>
<th>Area (nand2 – equiv)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSMC 65nm</td>
<td>200 MHz</td>
<td>266 K gates</td>
</tr>
<tr>
<td>TSMC 90nm</td>
<td>166 MHz</td>
<td>291K gates</td>
</tr>
</tbody>
</table>

* Results using Design Compiler with topographical technology
Example Design 2: Digital Camera

High Bandwidth Bus
- System Processor
- DMA Controller
- Image Processor
- USB Device
- Memory Interface
- NAND Controller
- LCD Controller
- CMOS Sensor Image Buffer

Low Bandwidth Bus
- NAND Controller
- WDT
- Timer
- GPIO
- ICTL
- Lens Control
System Requirements: Digital Camera

Requirements for a high-end compact consumer digital camera

• 10 mega pixel image sensor.
• 352 * 288 LCD display.
• Video record capability: 640 * 480 mpeg2 @ 30 frames per second.
• Raw image data output.
• USB interface.

What does this mean for the bandwidth requirements for our system?

• 10 Mp image sensor produces 240 Mb of data per raw image.
• Video recording bandwidth requirements
  • To sample @ 30 fps requires a bandwidth of 30*240Mb/s = 7.2 Gb/s (uncompressed stream from the image buffer)
  • Compressed stream to the storage media requires bandwidth of 15Mb/s
• LCD display requires a bandwidth of 4 Mb/s for video data.
Architecture Choice : Why we cannot use AMBA 2.0

Why not an AHB Implementation?

• Results show a 4 master 8 slave DW_AHB bus will synthesize to 200MHz @ 90nm.

• For a data width of 32, peak system bandwidth is $32 \times 200 = 6.4$ Gb/s.

• Since AHB is a shared address and data bus architecture, other system traffic must be considered.
Architecture Choice: Why AMBA 3 AXI?

Why an AXI Solution?

- Results show a 4 master 8 slave DW_axi will synthesize to 400 Mhz @ 90nm.

- For a data width of 32, peak data bandwidth of $400 \times 32 = 12.8$ Gb/s on a single master/slave link.

- For all master slave links in read and write directions we get a system bandwidth of $2 \times 4 \times 12.8 = 102.4$ Gb/s.

- Separate address busses means there is no control overhead to affect bandwidth.

- Multiple address multiple data architecture reduces arbitration latencies.

- Register slicing can be used to ease I/O delay requirements while adding only 1 cycle of latency.
Example Design 2: Digital Camera

System Processor

AXI

AXI to AHB bridge

CMOS Sensor Image Buffer

NAND Controller

Image Buffer

AXI to APB Bridge

USB Device

Memory Controller

Lens Control

ICTL

WDT

Timer

GPIO

AHB

DMA Controller

AHB master to AXI interface

Image Processor

AXI master to AXI interface

Synopsys DesignWare IP

ARM Developers’ Conference & Design Pavilion 2007
Digital Camera : Subsystem Implementation in coreAssembler

- Interfaces for third party IP are exported.
- A 20% i/o delay is used at all access points to the AXI interconnect.
## Digital Camera: Speed and Area Results

<table>
<thead>
<tr>
<th>Library</th>
<th>Frequency (MHz)</th>
<th>Area (nand2 – equiv)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TSMC 65nm</td>
<td>456 MHz</td>
<td>160.3K gates</td>
</tr>
<tr>
<td>TSMC 90nm</td>
<td>404 MHz</td>
<td>216.5K gates</td>
</tr>
</tbody>
</table>

- Results using Design Compiler with topographical technology

---

**ARM Developers’ Conference & Design Pavilion 2007**
#1 Synthesizable IP
- Highly configurable, technically differentiated subsystem building blocks
- Interoperable with other DW Cores

#2 Verification IP
- Highest Performance in VCS
- SystemVerilog & Constrained Random Verification Methodology support
- AMBA 3 Assured (AXI)

#3 Automated Assembly with coreAssembler
- Assemble both IP subsystem and VIP testbench

Comprehensive infrastructure Solution

ARM Developers’ Conference & Design Pavilion 2007
Synopsys DesignWare Library Solution - AMBA 2.0 and AMBA 3 AXI Synthesizable IP

• **Bus Fabric**
  • AMBA 2.0 Bus (DW_ahb)
  • AMBA 3 AXI Bus (DW_axi)

• **Bridges**
  • AHB to AHB Bridge (DW_ahb_eh2h,DW_ahb_h2h)
  • AHB Master to AXI Interface gasket (DW_axi_hmx)
  • AXI to AHB Bridge (DW_axi_x2h)
  • AXI to APB 3.0 Bridge (DW_axi_x2p)
  • AXI to AXI Bridge (DW_axi_x2x)
  • AHB to APB Bridge (DW_apb)

• AHB Central Direct Memory Access Controller (DW_ahb_dmac)
• Memory Controller (DW_memctl)
• AXI Register Slice (DW_axi_rs)
• AHB/APB Interrupt Controllers (DW_ahb_ictl, DW_apb_ictl)
Synopsys DesignWare Library Solution - AMBA 2.0 and AMBA 3 AXI Synthesizable IP

- **AHB Peripherals**
  - AHB Multi-layer Interconnect Matrix (DW_ahb_icm)

- **Generic interface modules**
  - Generic Interface to AMBA AXI Master (DW_axi_gm)
  - AMBA AXI Slave to Generic Interface (DW_axi_gs)

- **APB Peripherals**
  - APB General Purpose Programmable I/O (DW_apb_gpio)
  - APB Remap and Pause (DW_apb_rap)
  - APB Real Time Clock (DW_apb RTC)
  - APB Programmable Timers (DW_apb_timers)
  - APB Watchdog Timer (DW_apb_wdt)
  - APB Universal Asynchronous Receiver/Transmitter (DW_apb_uart)
  - APB I2C Interface (DW_apb_i2c)
  - APB I2S Bus (DW_apb_i2s)
  - APB Synchronous Serial Interface (DW_apb_ssi)
Summary

• Subsystem performance: Key to making right architecture choice between AMBA 2.0 and AMBA 3 AXI for meeting design goals

• Synopsys DesignWare IP solution for AMBA eases the implementation of AMBA 2.0 and AMBA 3 AXI protocol based designs
  • Synthesizable IP
  • Verification IP

• Includes automated subsystem assembly of DesignWare IP using coreAssembler
  • Easy assembly
  • Quick performance analysis
  • Exporting interfaces, importing non-DW IP

• Available as part of DesignWare Library at no additional charge

• For more details on DesignWare IP solutions, visit – http://www.synopsys.com/products/designware/designware.html