Introduction To AMBA

ARM®

ARM DVI 0010A

Introduction To AMBA

Release Information

The following changes have been made to this book.

Change History

Date	Issue	Confidentiality	Change
October 1996	A	NonConfidential	First release

Proprietary Notice

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by ARM and the party that ARM delivered this document to.

Product Status

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

Web Address

`http://www.arm.com`

Contents
Introduction To AMBA

Chapter 1
Introduction To AMBA

This document provides an overview of the ARM *Advanced Microcontroller Bus Architecture* (AMBA).

AMBA is a specification for an on-chip bus, to enable macrocells (such as a CPU, DSP, peripherals, and memory controllers) to be connected together to form a microcontroller or complex peripheral chip.

1.1 Introduction to AMBA

The AMBA specification describes an on-chip communications standard for designing high-performance 16 and 32-bit microcontrollers, signal processors and complex peripheral devices.

AMBA has been proven in and is being designed into:

- PDA microcontrollers, with a high number of integrated peripherals but also with very low power consumption
- multi-media microcontrollers with floating-point co-processors, on-chip video controller and high memory bandwidth
- complex peripheral ASICs for consumer products
- digital mobile communication devices integrating control and signal-processing functions

ARM's policy is to encourage the use of AMBA wherever possible. ARM partners have access to HDL models, development boards and other tools that support AMBA. Tutorials on AMBA are held regularly. All prospective users are invited to contact ARM by email to info@armltd.co.uk.

ARM acknowledges the contribution by its partners, and particularly GEC Plessey Semiconductors, in the generation of the AMBA Specification.

1.1.1 Aims

The AMBA specification aims to:

- facilitate 'right-first-time' development of embedded microcontrollers with one or more CPU/Signal Processor and multiple peripherals
- minimise the silicon overhead required for both an on-chip bus and for off-chip manufacturing test access
- be technology-independent:
  - to enhance re-usability of peripheral and system macrocells across a wide range of IC processes
  - in an appropriate manner for Full-Custom, Standard Cell and Gate Array technologies
- facilitate a chip family roadmap with reduced time-to-market. AMBA encourages modular design and processor-independence, aiding development of peripheral libraries, and allowing ready use of advanced cached-CPU cores

In order to achieve the above, the architecture has the key features of:

- highly modularity
- multi-master support
- low-power emphasis
- innovative test methodology

1.1.2 Licensing AMBA

The AMBA specification is open. Anyone can obtain a free copy of the specification from ARM and use the protocol to design chips. There is no associated licence or royalty. ARM partners may license MicroPack, which provides HDL of an example AMBA system, the compliance test suite and various license rights. Licensing queries can be e-mailed to info@armltd.co.uk.

However, there are no rights to modify or distribute the specification document. Users are not allowed to claim to be AMBA-compliant unless they have used the compliance test suite (provided in MicroPack V1.1).

ARM holds patent US/A/5525971 and other patents are pending.

1.2 AMBA Specification

The AMBA specification defines:

- a high-speed, high-bandwidth bus, the Advanced System Bus (ASB)
- a simple, low-power peripheral bus, the Advanced Peripheral Bus (ASP)
- access for an external tester to permit modular testing and fast test of cache RAM
- essential housekeeping operations (reset/power-up, initialisation and power-down)

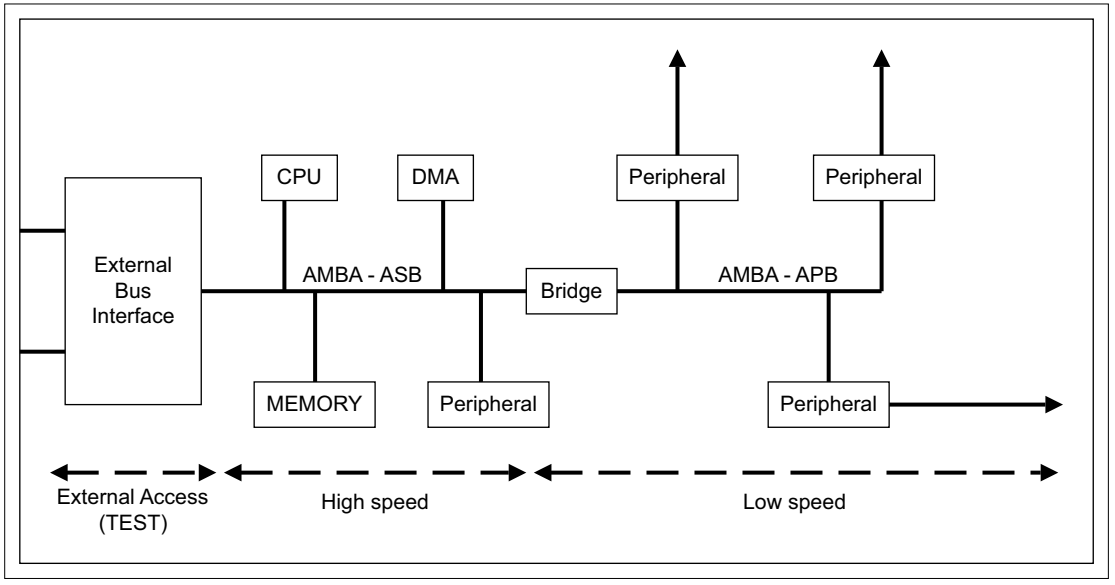

Figure 1-1 A typical AMBA-based microcontroller

Figure 1-1 shows:

Advanced System Bus, (ASB)

the high-performance bus which is the main system 'backbone'. This bus is also able to sustain the data rates required by the external bus interface. The CPU and other bus masters (such as a DMA controller), and high-speed local memory are normally connected to this bus. (The ASB is connected by a bridge to the simpler APB - see below)

Advanced Peripheral Bus, (APB)

the simple, low-speed, low-power peripheral bus.This is often, but not always, a narrower bus and is designed to be simple (i.e.unpipelined) for connecting many common peripherals such as timers, parallel I/O ports, UARTs, etc. (By placing these infrequently accessed peripherals on the APB, and partitioning them away from the ASB, loading on the ASB is reduced and allows maximum performance on the ASB to be more readily achieved.)

External interface

this is also used for test access. (This is normally an external memory interface but any suitable set of pins can be used.) An external tester can take control of the bus and check each component separately in turn. This modular approach promotes re-usability of test vectors. The parallel access method allows fast testing especially of cached CPUs.

The AMBA-based microcontroller also contains reset and power-management information to ensure:

- a single bus-driver on power-up
- consistent power-saving or power-down modes
- a reset mechanism for cold/warm/watchdog resets

Note

The specification describes signals and their protocol.

AMBA is technology-independent and does not detail the manufacturing process, nor the AC or DC characteristics (no layout information, no clock frequency figures, no current drive levels, etc.). Other areas outside the specification include clocking strategy and debug mechanism. ARM has considerable expertise in the above areas, and direct assistance through the ARM consultancy group is available.

1.2.1 The Advanced System Bus (ASB)

The ASB is designed for high-performance, high-bandwidth usage:

- Non-multiplexed (i.e. separate) address and data buses
- support for pipelined operation (including arbitration)
- support for multiple bus masters, with low silicon overhead
- support for multiple slave devices, including a bridge to the peripheral bus (APB)
- centralised decoder and arbiter

Absolute transfer rates depend on many design factors, but, for comparison purposes, if a 32-bit data path and a 100MHz clock are assumed, 200Mbytes/sec rate can be achieved. These figures are not limited by the specification but are simply provided for clarification.

Multiple bus masters are supported through the use of bus request, bus grant and bus lock signals. Use of these signals is optional; if you have a single bus master, you do not have the penalty of implementing these bus control lines.

1.2.2 The Advanced Peripheral Bus (APB)

The APB is designed to be a secondary bus to ASB, connected by a bridge (which limits the ASB loading). APB is a much simpler bus and has a low power focus:

- data access is controlled by select and strobe only (i.e. no clock, and thereby reducing power)
- almost zero-power consumption when bus is not in use
- simple unpiplined interface, typical of that required by many simple peripheral macrocells.

Data transfer rates are dependent on the speed of the peripherals. A single read or write cycle takes 5 clocks, so assuming a 32-bit data path and 100MHz clock, the data rate is 80Mbytes/sec. These figures are not limited by the specification but are simply provided for clarification.

The data bus of the APB can be more readily optimized to suit the peripherals connected. Many peripherals have narrow data path needs, and one mechanism may be to connect the 32-bit peripherals next to the bridge and 8-bit peripherals furthest away, reducing the die area needed for the bus.

Although the clocking strategy is not specified in AMBA, the partitioning provided by the bridge and APB does suggest a good starting point for minimising power consumption. Many peripherals (timers, baud rate generators, pwm units) require a divided-down system clock, and locating a single programmable divider adjacent to the bridge is convenient and power-efficient.

There is no bus master in APB (except for the bridge). All peripherals act as slaves.

1.2.3 ASB vs APB

In summary:

ASB is used for CPUs, DSP, DMA controllers and other bus masters, or high-performance peripherals (usually with FIFOs)

APB	is used for unpipelined, register-mapped slave peripherals, especially when the number of peripherals is high, and power-consumption needs to be minimised
ASB and APB	share the test methodology incorporated in AMBA

1.2.4 Test Interface Controller (TIC)

The Test Interface Controller (TIC) is an ASB bus master which uses the external bus interface (or other suitable pins) to provide test access for external test equipment. This mechanism allows a low gate count test access port, which offers fast parallel access, which is essential for testing cache RAM.

This test methodology allows re-use of test vectors from chip to chip, saving valuable time and reducing risk. For example, when a peripheral macrocell is re-used, the test vector module (which already exists in a proven, tested form) can be re-used, with only the upper level of the test program being specific from chip to chip.

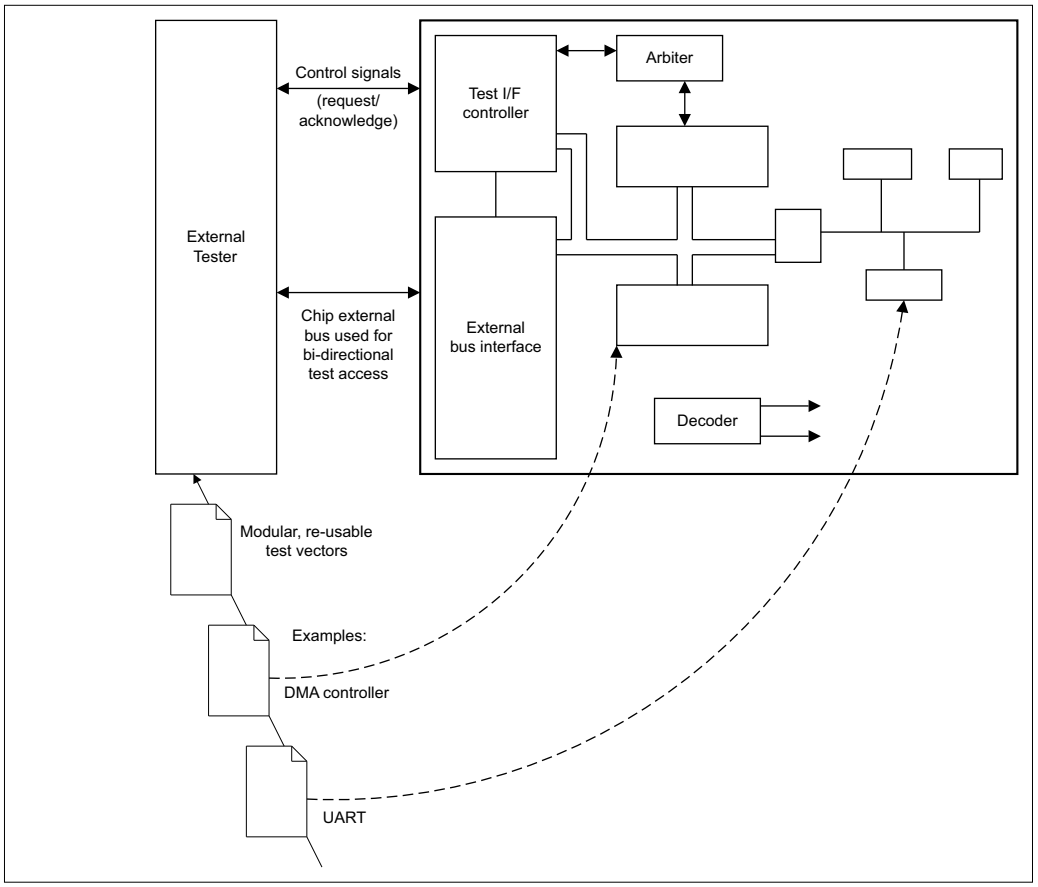

Figure 1-2 The Test Interface Controller using modular, re-usable test vectors

1.3 Summary

The AMBA specification has been stable from September 1995. This is now a proven bus architecture, and has been used in the ARM7100 and many other high-integration microcontrollers.

AMBA successfully addresses the problems of:

the chip architect

by enabling the device to be consistently partitioned into high-speed, low-power and production-test sections.

the project manager

by encouraging macrocell re-use, hence reducing development time and risk

the product roadmap

by being technology- and process-independent and enabling re-use of newer, more competitive processes

production test engineers

by providing a mechanism for fast test of cache RAMs and re-use of test vectors in a modular manner

ARM is continuing to develop support tools and environments around AMBA, including third-party EDA tool vendors (see *Appendices* on page 1-10).

1.4 Appendices

1.4.1 Document references

Table 1-1 Document references

Document	Part or Reference Number
AMBA Specification	ARM IHI 0001
ARM7100 Data sheet	ARM DDI 0035
ARM AMBA-peripheral data sheets	ARM DDI 0017
ARM MicroPack data sheets	ARM DDI 0014
ARM Reference Peripherals Specification	ARM DDI 0062

1.4.2 AMBA-related products and support

ARM is developing extensive support around and upon AMBA, as follows:

MicroPack

MicroPack is a set of HDL models which implement a sample AMBA system and a compliance test suite. Also included are various wrappers to interface CPUs to the AMBA high-speed bus

Many of ARM's partners have licensed MicroPack and can support OEMs directly. For example,

- supply of HDL models of the sample AMBA system
- supply of development card
- integration of peripherals, from ARM and partner.

Development card

This implements AMBA on a PCB. This card allows various CPUs (via a header card) to be used, and peripherals to be prototyped in an FPGA.

Peripheral Library

ARM is developing an increasing number of AMBA-compatible peripherals to speed up the design of high-integration microcontrollers. These peripherals are particularly suitable for PDA and Network Computer applications.

Consultancy

The ARM consultancy group has considerable AMBA expertise. This group has adapted several CPUs to have an AMBA native bus. Other expertise ranges from implementing AMBA-compatible peripherals through to designing high-integration microcontrollers with innovative architectures based on AMBA.

Real-time ICE

ARM has developed and patented an innovative concept for real-time in-circuit emulators, using AMBA. The real-time ICEs are currently under development and are likely to be available Q1’97 onwards.

3rd-Party

EDA tool vendors will be announcing AMBA support from Q4’96 onwards.

