

AMBA DEDICATED DMA CONTROLLER WITH MULTIPLE MASTERS USING VHDL

Archana Tiwari¹ & D.J.Dahigaonkar²

This paper describes implementation of DMA controller of AMBA Bus with two masters using VHDL. DMA controller is a synthesizable soft IP core connected to the AMBA AHB Bus for easy integration into SOC implementation. The Direct Memory Access (DMA) Controller is a hardware feature that enables movement of blocks of data from peripheral to memory, memory to peripheral or memory to memory. This movement of data by a separate entity significantly reduces the load on the processor. A DMA controller can be used to save power in a system by putting the CPU in a low power state and using the DMA controller (fewer gates/transactions) to move the data. The simulation results obtained indicate bidirectional memory read and write operation of AMBA AHB DMA controller. The proposed architecture provides bus access to any one master at a time for improved speed and performance.

Keywords: DMA, AMBA, SOC, IP core, VHDL

1. INTRODUCTION

The heart of the system is a microprocessor. This microprocessor is able to access peripherals (also on chip) via a special bus for embedded applications. This special bus is called AMBA (Advanced Microprocessor Bus Architecture) [3].

AMBA is specified by the custom microprocessor company ARM [10]. ARM limited's use of a generic bus in the Advanced Microcontroller Bus Architecture (AMBA).

AMBA bus which has the largest market-share is simpler in architecture than any other buses. The AMBA bus is applied easily to small scale SOCs [1] [2]. Most IP vendors have produced large quantities of AMBA compatible IPS. Therefore, the AMBA bus has been the representative of the SOC market though the bus efficiency shows the limitation in performance with the growing SOC scale.

Three distinct buses are defined within the AMBA specification [4]. The Advanced High-performance Bus (AHB), The Advanced System Bus (ASB), The Advanced Peripheral Bus (APB).

A typical AMBA-based microcontroller will incorporate either an AHB or an ASB together with an APB as illustrated in Fig.1. The ASB is the older form of system bus, with AHB being introduced later to improve support for higher performance, synthesis and timing verification. The APB is generally used as a local secondary bus which appears as a single slave module on the AHB or ASB.

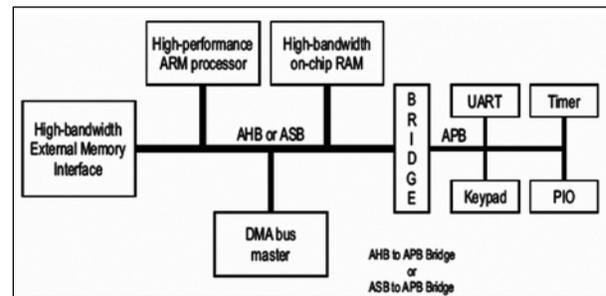


Fig.1: Typical AMBA BUS Structure

To access memory, processor must use a special memory controller also on the AMBA bus. In order to function efficiently (for data capture, telemetry etc) it is necessary for the AMBA bus peripherals to directly write blocks of data to memory. This function is performed by a DMA controller [6] [7], which is created by this project.

The Direct Memory Access (DMA) Controller [5] is a hardware feature that enables movement of blocks of data from peripheral to memory, memory to peripheral or memory to memory. In common with all the other modules of this project the controller is written in VHDL.

2. METHODOLOGY

The paper describes implementation of DMA controller of AMBA Bus using VHDL. DMA controller is a synthesizable soft IP core connected to the AMBA AHB Bus. The IP blocks built using hardware description languages VHDL can give great advantages, because they provide a high-level description of components (behavioral description) leaving to logic synthesis tools the burden of using the best available physical resources available to efficiently implement the described hardware [9].

^{1,2} Department of Electronics Engineering, Shri. Ramdeobaba Kamlu Nehru Engineering College, Nagpur, RTM Nagpur University

Email: t.arch01@yahoo.com¹, dipak_dahigaonkar@rediffmail.com²

FPGA vendors such as Altera [11] or Xilinx [12], provide high performance and accurate IP blocks, also at a very coarse grain, such as the Nios and Microblaze processors [13] [14].

3. PROPOSED ARCHITECTURE

The key idea of the project is to design DMA controller for AHB of AMBA Bus with multiple masters. In my design, I am implementing DMA controller of AMBA Bus with two masters.

A bus master is able to initiate read and write operation by providing address and control information. Only one bus master is allowed to actively use the bus at any one time. In my design, I am including Direct memory Access (DMA) as one of the master. Another master is Host which is included just for demo purpose.

Fig. 2 shows the proposed architecture of DMA controller for AMBA bus. It consists of DMA system, Host and Arbiter. The DMA system consists of DMA, Peripheral 1, Peripheral 2 and arbiter. DMA consists of memory and control unit.

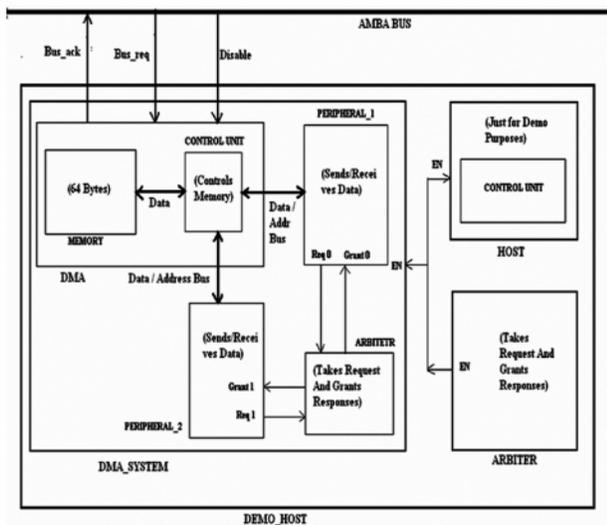


Fig.2: Proposed Architecture of DMA Controller for AMBA Bus

The DMA controller, which consists of memory and control unit is the basic module of our project. The memory is of 64 bytes. The address is of 6 bits and data is of 8 bits. The control unit will control the reading and writing operation. When peripheral 1 want to read data from memory or write data into memory, peripheral 1 will send enable signal high. When peripheral 1 wants to read the data from memory, it will make read write bar signal high. Similarly it will make read write bar signal low to write the data into memory. Same process is done with peripheral 2. When peripheral 2 want to read the data from memory, it

will make read write bar signal high and it will make read write bar signal low to write the data into memory.

But at a time only one peripheral can work, that is only one peripheral can take the access of BUS for reading or writing the data. So that we are also designing arbiter to give priority between peripheral 1 and peripheral 2. The arbiter will take the request from peripheral 1 and peripheral 2 and will grant the response to peripheral 1 and peripheral 2 respectively. But when both the peripherals will give the request at a time then arbiter will give response to peripheral 1 and peripheral 2 alternatively.

One more arbiter is used to give priorities between DMA system and the host.

4. SIMULATION

For simulation of VHDL modules of DMA controller of AMBA BUS, I am using Xilinx 9.1 FPGA vendor. Xilinx provide high performance and accurate IP blocks of DMA controller [8]. The simulation results for proposed architecture i.e. DMA controller of AMBA BUS is shown below.

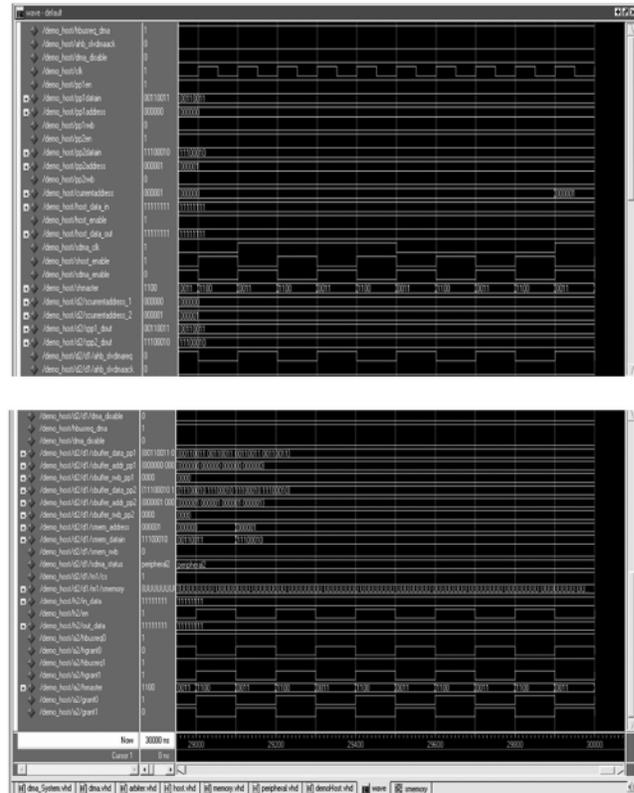


Fig. 3: Simulation Results which Shows Writing Operation

The simulation for writing operation is shown in fig 3. After application of clock signal, peripheral 1 and peripheral 2, both are requesting bus for writing operation to the control unit. The control unit will grant bus access alternatively to both the peripherals.

After getting bus access, peripheral sends 6 bit address, 8 bit data, and read /write signal to control unit. The data given by the peripheral is then written in the memory location whose address is also provided by the peripheral.

The data which is given by peripheral 1 and peripheral 2 is written in the memory location whose address is also given by the peripheral 1 and peripheral 2 respectively. Fig 4. shows the Data transfer from peripheral to the selected memory locations.



Fig. 4: Memory Status after Writing the Data from Peripherals

After writing operation, reading operation starts where data is read from memory. This is shown in fig. 5. The data from selected memory location is read by the peripheral which appear at data output line.

5. SYNTHESIS

In this paper, for synthesis of VHDL modules of DMA controller of AMBA BUS, Xilinx 9.1 is used. The synthesis report is given below in Fig. 6.

# RAMs	4
4x6-bit dual-port distributed RAM	2
4x8-bit dual-port distributed RAM	2
# Adders/Subtractors	5
32-bit adder	5
# Counters	5
32-bit up counter	5
# Registers	66
Flip-Flops	66
# Latches	66
8-bit latch	66
# Multiplexers	6
1-bit 4-to-1 multiplexer	4
4-bit 4-to-1 multiplexer	1
8-bit 64-to-1 multiplexer	1

Fig. 6: HDL Synthesis Report of DMA Controller

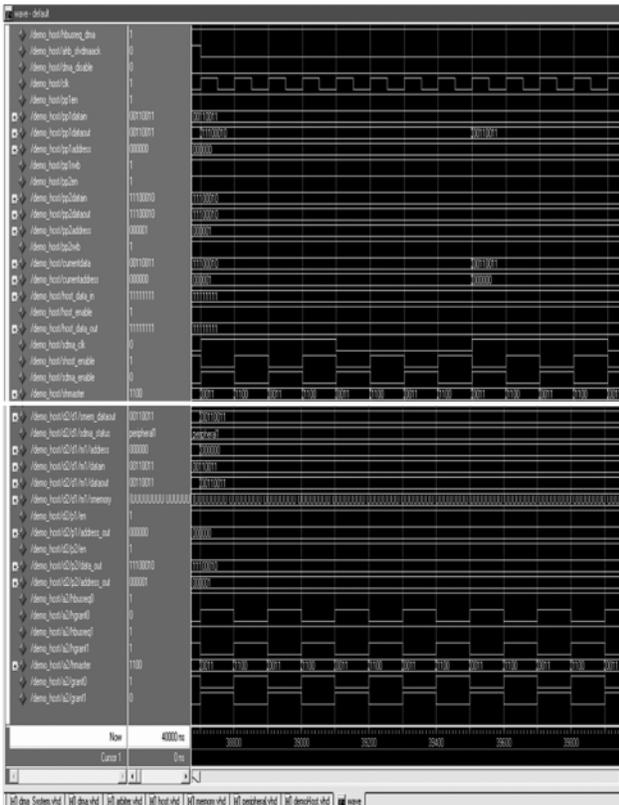


Fig. 5: Simulation Results which Shows Reading Operation

6. CONCLUSION

In this paper, we presented how we approached the VHDL design of DMA controller AMBA Bus with multiple masters i.e. Direct memory Access (DMA) as one of the master and another master is Host. Only one bus master either DMA or Host is allowed to actively use the bus at any one time. The DMA controller modules are implemented using VHDL, simulation and synthesis is done by Xilinx.

REFERENCES

- [1] J. Liang, Swaminathan, S, "ASOC: a Scalable, Singlechip Communications Architecture", Tessier, R. Parallel Architectures and Compilation Techniques, 2000, Proceedings. International Conference, pp.37-46, 2000.
- [2] F. Polmi, D. Bntazzi, L. Bmini, and A. Bogliolo., "Psrfomanncc Aoyalisirr of Arbibration Policics for sd: Communication Architecfurcs", Kluwa Jolnnal on Design Automation for Embedded Systmu, 8, np. 2, pp. 189-210, 2003.
- [3] Flynn, D., "AMBA: Enabling Reuseable On-Chip Designs", IEEE Micro, 17(4), July/August 1997, pp 20-27.
- [4] AMBA Specification (rev2.0) and Multi Layer AHB Specification, Arm: <http://www.arm.com>, 2001.

- [5] AMBA Bus DMA Controller Specification, Draft 1.03, Gareth Morris 2000.
- [6] Primecell DMA Controller (PLO8 I) Technical Reference Manual, <http://www.arm.com>
- [7] C8237 Programmable DMA Controller Core, <http://www.cast-inc.com>
- [8] Xilinx FPGAs: A Technical Overview for the First-Time User. Application Note. XAPP 097 December 12, 1998 (Version 1.3). <http://www.xilinx.com>
- [9] J. Becker, L. Kabulepa, F.M. Renner, M. Glesner, "Simulation and Rapid Prototyping of Flexible Systems-on-a-Chip for Future Mobile Communication Applications", RSP 2000, pp. 160-165.
- [10] <http://www.arm.com/products/solutions/AMBAHomePage.html>
- [11] <http://www.altera.com>
- [12] <http://www.xilinx.com>
- [13] <http://www.altera.com/products/ip/processors/nios/nio-index.html>
- [14] http://www.xilinx.com/xilinx/xebiz/designResources/ip-product_details.jsp?key=micro_blaze