IMPLEMENTATION OF AMBA-APB BUS DEVICES ON FPGA

Niraj Kumar Mishra[1], Dr. R.P. Singh[2], Dr. Jaikaran Singh[3],

[1] *PG Student(VLSI) ,Electronics & Communication Department, SSSIST,Sehore,M.P.-India*
[2] *Professors, Electronics & Communication Department, SSSIST,Sehore,M.P.-India*
[3] *HOD, Electronics & Communication Department, SSSIST,Sehore,M.P.-India*

ABSTRACT

In the present era SoC design is done through IP development and implementation technology and implemented using block reuse. In SoC communication high performance, low power consumption and testability are essential. A The verification also follows the same pattern and verification IPs has been used regularly.The AMBA-APB is a very useful protocol in SoC design and various communication protocol base. Here, we present the simpler design of APB controller which handles and controls the transactions between the designed master and periphera(slave)l devices. The final design which integrates the peripheral devices with the APB controller structure is implemented on the FPGA device. We are tested the design at various FPGA based architecture.

Keyword: *SoC , FPGA, AMBA, Advanced Peripheral Bus (APB)*

1. Introduction

This modern era of VLSI technology the computing devices depend more on system on-chip communication (SoC) protocols for data switch [1]. In any SoC the communication standard should be trustworthy. With the aim to deal with many off chip communication issues sophisticated RISC Machine (ARM) introduced Advanced Microcontroller Bus Architecture (AMBA). AMBA define both a bus design and a technology-independent method for scheming, implementing [2], and testing customized controllers. beginning of AMBA buses by ARM finished a notable change in the accessible communication scenarios. It was a resolution for many on hand bottle necks like adhoc design approach, infrastructure portability, centralized state machine etc [3]. ARM has defined various bus stipulation from 1996.Such terms address two types of busses: scheme bus and peripheral bus [4]. The rising complexity of Systems-on-Chip (SOC) has led to the serious “design capability gap” problem. On-chip statement architectures have a noteworthy impact on system presentation, power indulgence and time to-market. System designer, as well as the investigate community have paying attention on the issue of explore, evaluate, and scheming SOC communication architectures to meet the embattled design goals [5].

AMBA 2.0 specification defines three different buses [1]:-
1. Advanced high Performance Bus (AHB)
2. Advanced System Bus (ASB)
3. Advanced Peripheral Bus (APB)

An AMBA specified embedded controller normally consists of a high-performance structure backbone bus (AMBA AHB or ASB), which is able to uphold the outer memory bandwidth. AHB bus provide a high-bandwidth interface stuck between different elements which are implicated in the different transfers. There is also an accessibility of the AHB to APB Bridge, which is mainly used to detailed access low peripheral devices on high recital bus. APB Bridge is only the master for APB bus.

1.1 Electrical Characteristics

No specified information linked to electrical features and characteristics is supplied or provided within the AMBA 2.0 specification as this will be totally dependent on the built-up process technology that is chosen for the design.

1.2 Timing Specification

The AMBA 2.0 procedure define the performance of various signals at the sequence level. The accurate timing requirements will depend on the development technology used and the frequency of operation. Because the exact timing requirements are not defined by the AMBA 2.0 protocol, the system integrator is given maximum litheness in allocating the hint timing budget amongst the various modules on the bus.

2. The AMBA Standard

AMBA uses specific terminology for SoC components. Since this terminology will be used throughout this book, we introduce it here with an illustration of a simplified SoC based on the AMBA BUS shown in Fig. 2.1.

2.1. Master: A master device, usually linked to a high-throughput scheme bus, is an IP which is capable of initiating communications on the bus. Figure 2.1 contains two such masters: Master 1 and Master 2.

2.2. Arbiter SoC may contain numerous masters, an suitable arbitration algorithm is desirable. In the ASB system as shown in Fig. 2.1, a central arbiter is available, which connects to all masters in the system directly, and can provide bus entrée to them using an suitable scheduling scheme. The central arbitration scheme can be round-robin, prioritized, etc.

2.3 Slave: A slave device respond to communication desires from a master. While peripheral devices, memory controllers, and system controllers are categorized as slaves, even processing elements such as DSPs may act as slaves. In Fig. 2.1, the SoC contains a single reminiscence controller slave.

2.4. Decoder: Masters use the address bus of the interconnect to initiate slaves. A decoder operates and decodes the address (or a part of it) into applicable control signals to facilitate slaves. For example, in Fig. 2.1, the decoder reads the address issue by the masters, and translates it into a slave select signal DSel0 for the memory controller slave

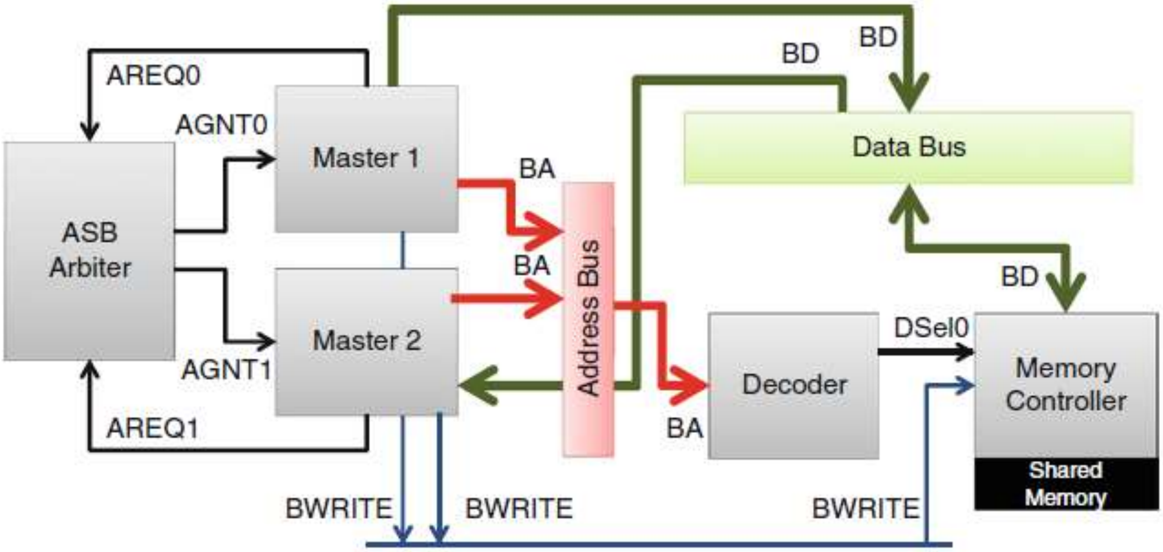


Fig. 2.1 AMBA Architecture section

3. Architecture overview

The architecture design is produced and developed based on the customary specifications [2] of the protocol. The architecture explains the utility modules designed.

3.1 Design Module

The module is a product of amalgamation of low Bandwidth peripheral devices with the APB controller. The maneuver of the slave devices are controlled by APB controller. The Top module is presented first and the sub modules are discussed in further sections.

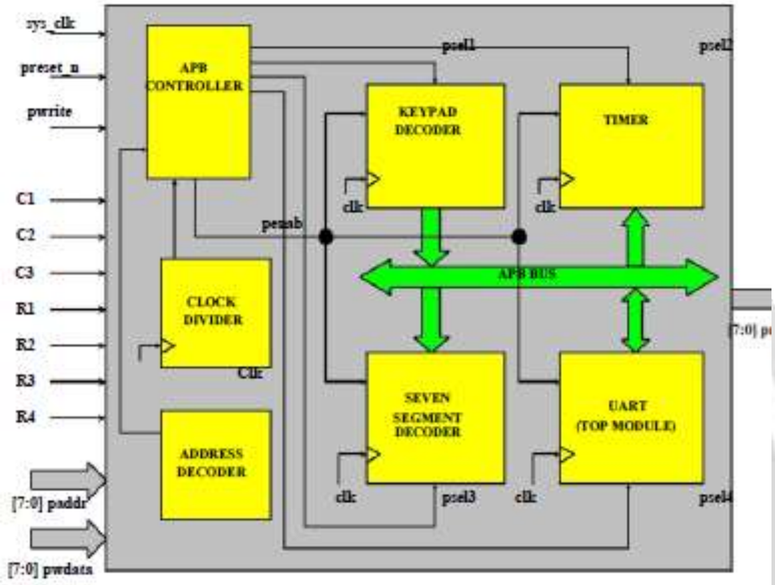


Fig -3.1: APB Controller Block Diagram

3.2 Top Module – APB Integrated with Peripheral Devices

Block diagram presented here does not indicate the internal units of the design. The detailed discussion of the architectures of the different modules is done in future sections

Table 1 Signal list of top module design block

I/O NAMES	I/O DESCRIPTION	I/O DESCRIPTION
sys_clk	input	System clock.
preset_n	input	System reset signal.
pwrite	input	Read or Write signal.
C1	input	Column 1 input.
C2	input	Column 2 input.
C3	input	Column 3 input.
R1	input	Row 1 input.
R2	input	Row 2 input.
R3	input	Row 3 input.
R4	input	Row 4 input.
paddr [7:0]	input	8 bit address bus.
pwdata [7:0]	input	8 bit Write Data Bus.
prdata [7:0]	output	8 bit Read Data Bus.

4. Simulation results

The proposed design and techniques has been coded in simple Gvim editor and then simulate using Modelsim as well as Xilinx ISE tool. The steps has-been explained in the above chapter. In this chapter present the result on the GUI. I show the results of various sub modules on the simulation window. I separately show the read and write operations Graphical user interface approach.

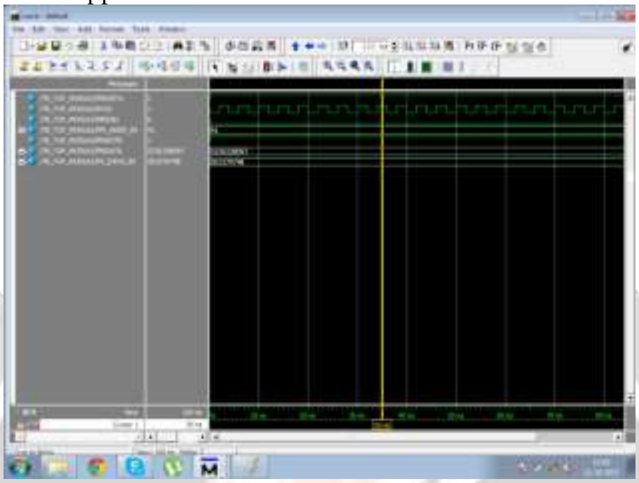


Fig.4.1: Top Module simulation result.

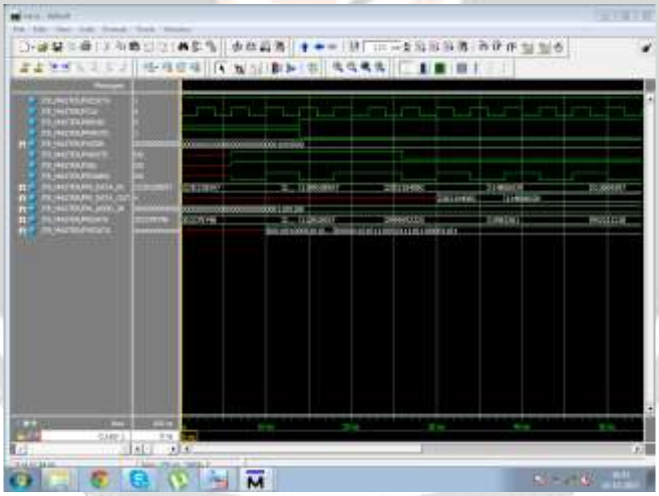


Fig.4.2: Master-Slave Read and write transfer

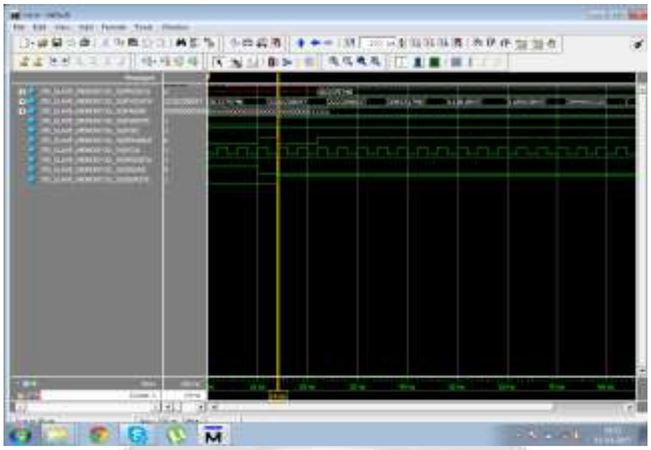
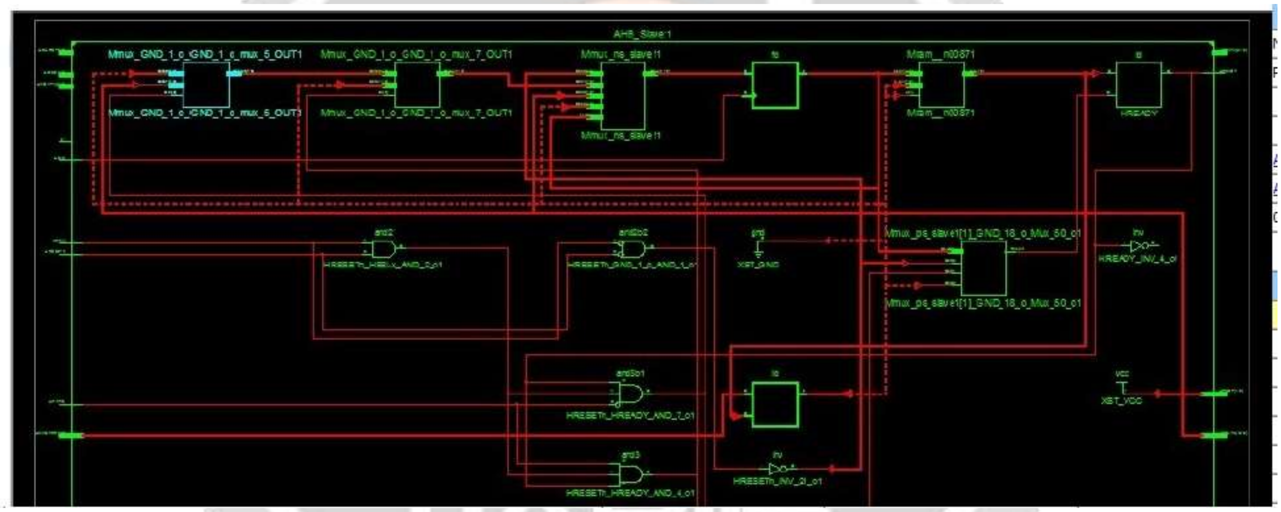


Fig.4.3: Decoder Simulation results

Final synthesised RTL view is given below



6. REFERENCES

[1] Nikil Dutt, Kaustav Banerjee, Luca Benini, Kanishka Lahiri, Sudeep Pasricha, "Tutorial 5: SoC Communication Architectures: Technology, Current Practice, Research, and Trends," vlsid, pp.8, 20th International Conference on VLSI Design held jointly with 6th International Conference on Embedded Systems (VLSID'07), 2007

[2] AMBA Specification (Rev 2.0).

[3] Flynn, D. Adv.RISC Machines Ltd., Cambridge, "AMBA: enabling reusable on-chip designs", IEEE Micro, Publication Date: Jul/Aug 1997.

[4] "Using formal techniques to Debug the AMBA System-on-Chip Bus Protocol" Proceedings of the Design, Automation and Test in Europe Conference and Exhibition, 2003.

[5] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[6] Xilinx ISE Synthesis and Verification Design Guide.