Implementation of AMBA-ASB Memory Controller with Power Analysis

Vipin Kumar Yadav[1], Nikhil Kumar Sharma[2]

[1]Asst. Professor, Department of Electronics, SRM University, NCR Campus, Ghaziabad [U.P] INDIA
[2]Student, M.Tech, Department of Electronics, SRM University, NCR Campus, Ghaziabad [U.P] INDIA
Vipinyadav_85@yahoo.co.in[1],niksharma52@gmail.com[2]

***Abstract*—** The work presented here is a summary of result obtained when AMBA-ASB memory controller was simulated and synthesized, using ModelSim 6.4a and Xilinx ISE 14.7. Memory controller is designed using master and slave circuit. Memory controller controls the flow of data from master to slave peripheral. Memory controller is a digital circuit. AMBA-ASB has several features i.e it provides parallel communication, high clock frequency, high performance system.

Keywords*—AMBA, ASB, APB, Xillinx, ModelSim.*

*****

I. INTRODUCTION

In our advent into the world circuit simulation and synthesis, we have selected a subject, which is center of many developments in the world of VLSI (Very large scale integration). VLSI has seen its extended boundary. It is our intent to contribute, in our own small way, to this large growing pool knowledge.

Synthesis technique allows us to analyze VLSI circuit on large scale. We also determine the power dissipation. There are many simulation and synthesis tool available. In this case, we use ModelSim 6.4a for simulation and Xilinx ISE 14.7 for synthesis.

This paper describes the AMBA-ASB memory controller using master and slave circuit. Section 2 describes the architecture of ASB-MC and its various features. Section 3 lists the simulation and synthesis result of memory controller and hardware parameter. Section 4 describes conclusion and also about the power analysis. When multiple processors are required AMBA is widely used. The full name of "AMBA" is Advance Microcontroller Bus Architecture. ASB-MC is an Advance system bus memory controller.

The AMBA is designed, tested and licensed by ARM Limited. ARM is embedded processor which is SoC (System on chip). By using AMBA-ASB we can do on chip communication for high performance embedded microcontroller [15].

II. **Architecture of Asb-Mc**

AMBA (Advance Microcontroller Bus Architecture) defines an on chip communication for high performance embedded system. When AHB bus is not required in that case we use ASB bus which means AMBA ASB is alternate to AHB. ASB-MC has been designed using five module i.e APB bridge, Arbiter, Decoder, memory system, external memory interface.[2]

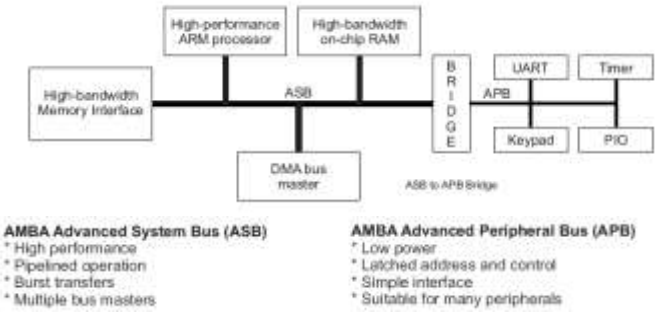

Fig. 1 AMBA based microcontroller system [2].

A. APB Bridge

The APB (Advance peripheral Bus) Bridge is used for connection establishment between the master and slave circuit [7]. Bridge is used to change the ASB signal into APB signal. By using wait state we insert pipelining in ASB. We design APB bridge using state machine. Case 1 when state is ST_IDLE, reset the bridge and no peripheral is selected. Next state is ST_READ if DSELPeri is logic high and BWRITE is low. Case 2 state is ST_RESD, present state is ST_IDLE when non-sequential cycle and ST_ENABLE when sequential cycle. Next state is ST_ENABLE. Case 3 state is ST_WWAIT, present state is ST_IDLE when non-sequential cycle and ST_ENABLE. Next state is ST_WRITE. Case 4 state is ST_WRITE, present state is ST_WWAIT when non-sequential and ST_ENABLE when sequential cycle. Next state is ST_ENABLE. Case 5 state is ST_ENABLE, present state is ST_READ and ST_WRITE. Next state is ST_IDLE[4].

 ISSN: 2321-8169
1428 - 1431

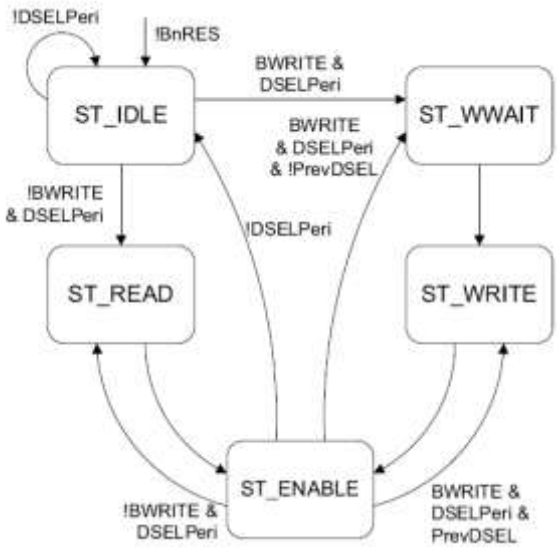

Fig 2 State machine of bridge.

B. Arbiter

AMBA ASB supports multiple bus master system, to complete this we require arbiter. Arbiter acts as an arbitration between bus master completing for access to the ASB. In AMBA-ASB, Arbiter is priority based system. The process of arbiter is to select one bus master at any particular point of time.

According to highest priority we grant the signal. Block diagram of arbiter is as.[11]

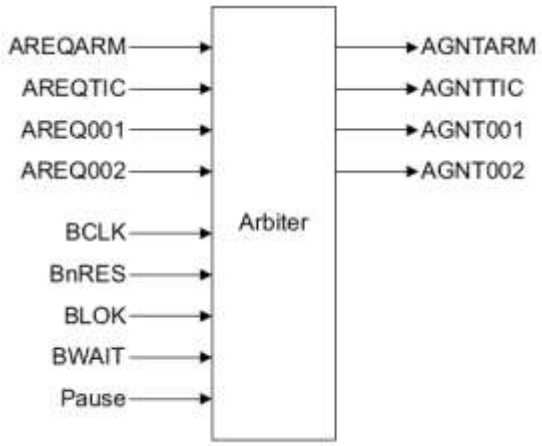

Fig. 3 Arbiter block diagram.

C. Decoder

Decoder has three functions in ASB bus i.e i) address decoder ii) Default transfer response iii) protection unit. Decoder selects the slave signal for every bus slave. Even when there is no transfer required decoder keeps on operating. Decoder simplifies the circuit by using the centralizing decoding. Decoder increases the portability of peripheral by making independent memory map system.

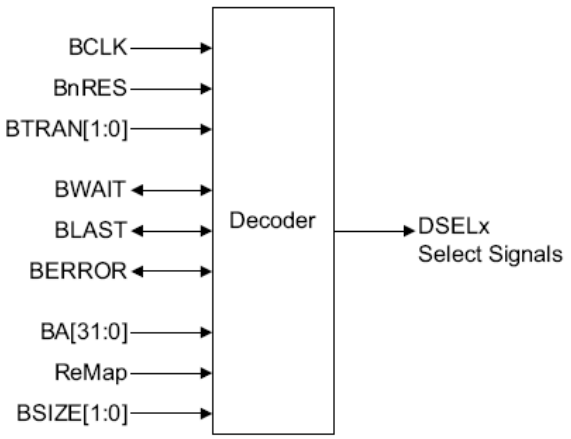

Fig. 4 Decoder block diagram.

D. External memory interface

External memory interface is divided into two section i.e 1) Memory bank select 2) Memory write control signal.[4]

1) Memory Bank Select

ASB-MC design uses Remap signal for the memory mapping. There are four memory banks which are selected by XCSN signal. XCSN chip selects lines which are controlled by values of BA, Remap, and DESELExtMem. When BCLK signal is low, falling edge of clock XCSN signal is generated. When circuit is reset XCSN[7:0] remains at "11111111" state asynchronously.[4]

Table 1 input output relation.

Input			Output
DSELExMem	**Remap**	**BA[30:28]**	**XCSN[7:0]**
0	X	XXX	11111111
1	0	XXX	01111111
1	1	000	11111110
1	1	001	11111101
1	1	010	11111011
1	1	011	11110111
1	1	100	11101111
1	1	101	11011111
1	1	110	10111111
1	1	111	01111111

2) *Memory writes control*

For memory writing XWEN signal is used in ASB-MC. Memory writes in word (32 bits), half word (16 bits) and byte (8 bits).

 ISSN: 2321-8169
1428 - 1431

Table 2 Relationship between XCSN and the inputs from ASB bus.

BSIZE[1:0]	BA[1:0]	XWEN[3:0]
10(word)	**XX**	0000
01(half word)	0X	1100
01(half word)	1X	0011
00(byte)	00	1110
00(byte)	01	1101
00(byte)	10	1011
00(byte)	11	0111

III. **SYNTHESIS AND SIMULATION RESULTS**

TABLE 3
THE HARDWARE/SOFTWARE SETUP

Operating System	Microsoft Windows 8 Pro
Processor	Intel Pentium CPU B950 – 2.10 GHz
Memory	2GB
Simulated using	1. ModelSim 6.4a 2. Xilinx ISE 14.7

The architecture of ASB-MC shown in Fig. 1, can be represented in form of a block diagram as shown in Fig. 3. The overall inputs and outputs of the system are also indicated.[13]

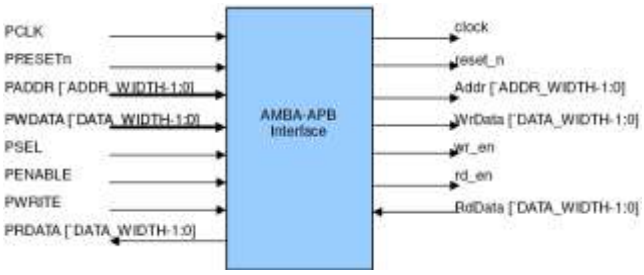

Fig. 3 Block diagram of APB interface.

The function of each module of ASB-MC is simulated by ModelSim 6.4a and each module is tested by Xilinx ISE tool in Verilog.
To implement the AMBA-ASB bus the step we have chosen are as follow: code generated, simulated, synthesis, power analysis, RTL Schematic.

Simulation result of each module is as.

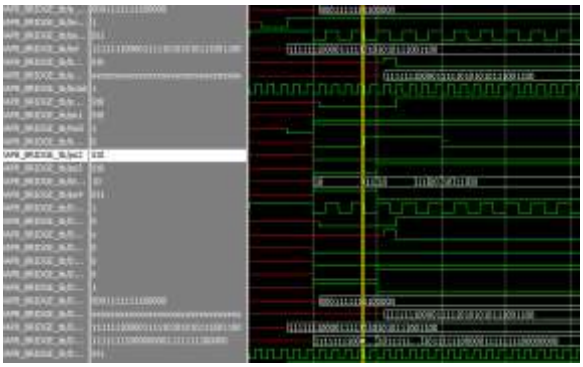

Fig. 5 Simulation of Bridge.

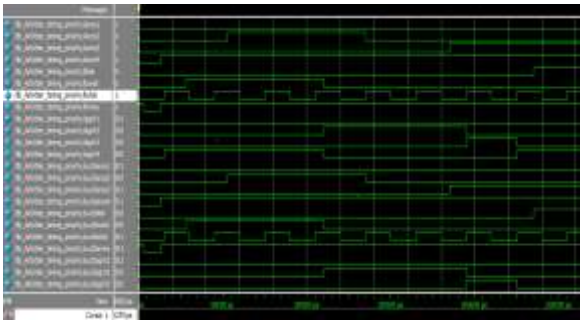

Fig. 6 simulation of Arbiter.

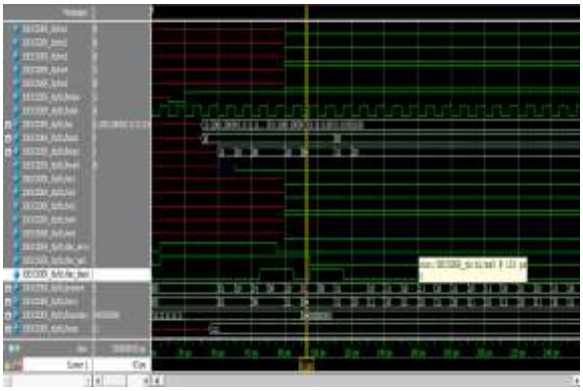

Fig. 7 Simulation of Decoder.

After simulation we synthesis the overall ASB-MC with Xilinx tool fig 8 shows RTL view of circuit. Synthesis the ASB-MC for target device xc7a100t-3csg324.

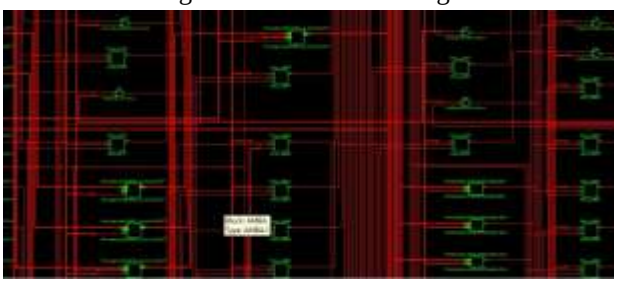

Fig 5. RTL schematic view.1

Device utilization summery is shown in table 4.

Table 4 Device utilization summary

Logic utilization	Used	Available	Utilization
Number of slices	2784	128600	2%
Number of LUTs	1527	63400	2%
Number of FFs	886	3670	24%
Number of IOBs	76	210	36%
Number of BUFG	16	32	50%

Power analysis of ASB-MC was calculated by using Xilinx XPower Analyzer. Table 5 shows the power report of ASB-MC. The total power dissipated by ASB-MC is 0.130W when maximum frequency is 200MHz.

Table 5 Power analysis report of ASB-MC.

Supply Summary		Total	Dynamic	Quiescent
Source	Voltage	Current (A)	Current (A)	Current (A)
Vccint	1.000	0.056	0.039	0.017
Vccaux	1.800	0.014	0.000	0.013
Vcco18	1.800	0.008	0.004	0.004
Vccbram	1.000	0.000	0.000	0.000
Vccadc	1.710	0.020	0.000	0.020
		Total	Dynamic	Quiescent
Supply Power (W)		0.130	0.047	0.082

IV. CONCLUSIONS

The two operations read and write can be seen as completed as follows when there is no wait state to the external ROM read and write operation is completed. Write operation is marked as completed when there is zero state external RAM.

In this paper, it can be seen that in comparison to serial communication, parallel communication results into improved faster data communication in AMBA-ASB. It also provides parallel read and write operation to be performed simultaneously on master and slave. Their number can be up to a limit of 16.

Wait state are used to overcome the system inability to perform address decoding and memory access operation in one clock cycle due to increase in system frequency but it results in decrease in system performance.

To avoid such performance issue burst method is used. In this method, first data beat compares information about the remaining beats. For example when AMBA-ASB system accepts one beat to perform read operation i.e data required can be read from memory and further stored in FIFO designated to read data. Some delay can be observed in first transfer before data returns. The further beats apart from one experience least delay because of data already present in FIFO.

Power dissipation plays crucial role in high performance of a device. When device counts n clock rates are high it becomes a concert reason. The proposed device consumes an optimized power of 0.130W and operates with maximum frequency of 200MHz.

REFERENCES

[1] Hu Yueli,Yang Ben, "Building an AMBA AHB compliant Memory Controller", IEEE, 2011.
[2] ARM. (1999) AMBA Specification (Rev 2.0). [Online]. Available http://www.arm.com
[3] "PrimeCell AHB SRAM/NOR Memory Controller", Technical Reference Manual, ARM Inc, 2006.
[4] "ASB Example AMBA System", Technical Reference Manual, ARM Inc, 1999.
[5] Clifford E. Cummings, Sunburst Design Inc., "Synthesis and scripting Techniques for Designing Multi-Asynchronous Clock Designs", 2001.
[6] Sudeep Pasricha, "On-Chip Communication Architecture Synthesis for Multi-Processor Systems-on-Chip", University of California, 2008.
[7] Vani.R.M and M.Roopa "Design of AMBA Based AHB2APB Bridge" IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.11, November 2010
[8] S. Osborne, A.T. Erdogan, T. Arslan and D. Robinson "Bus encoding architecture for low-power implementation of an AMBA-based SoC platform" IEE Proc.-Comput. Digit. Tech., Vol. 149, No. 4, July 2002.
[9] Sangik Choi and Shinwook Kang "Implementation of an On-Chip Bus Bridge between Heterogeneous Buses with Different Clock Frequencies" (IDEAS'05) 2005 IEEE
[10] Salita Sombatsiri, Kazuhiro Kobashi, Keishi Sakanushi, Yoshinori Takeuchi and Masaharu Imai "An AMBA Hierarchical Shared Bus Architecture Design Space Exploration Method Considering Pipeline, Burst and Split Transaction" 2013 IEEE.
[11] Sandra Mohan, Anish Joseph, "A Dynamic Priority Based Arbitration Algorithm" International Journal of Innovative Technology and Exploring Engineering (IJITEE) June 2013.
[12] D.Shanthi, Dr.R.Amutha Professor/ECE, "Design Approach to Implementation of Arbitration Algorithm in Shared Bus Architectures (MPSoC)".2011 iiste.org.
[13] Data sheet of AMBA-APB interface.
[14] G.Madhuri, P. Sharmila Rani, Jeevan Reddy K, Vasuja Devi Midasala "Design and Implementation Based On Amba4.0 Of Apb Bridge" IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) (Nov. - Dec. 2012).
[15] S.Ramakrishna , K.Venugopal, ,B.Vijay Bhasker , R.Surya Prakash Rao Hdl Implementation of Amba-Ahb Compatible Memory Controller International Journal Of Computational Engineering Research (ijceronline.com) Vol. 2 Issue. 7.