International Journal on Recent and Innovation Trends in Computing and Communication
Volume: 4 Issue: 4

ISSN: 2321-8169
573 - 576

Implementation of AMBA AHB protocol using verilog HDL

Abhinandan Aggarwal
Dept. of Electronics and communication Engineering
Maharaja Agrasen Institute of Technology, IPU
New Delhi, India
abhi2094@gmail.com

Prof. (Dr.) Neelam Sharma
Dept. of Electronics and communication Engineering
Maharaja Agrasen Institute of Technology, IPU
New Delhi, India
neelam_sr@yahoo.co.in

***Abstract*—** Advanced Microcontroller Bus Architecture (AMBA) is a series of bus protocols provided by ARM. These include AHB, APB and ASB. AMBA provides an on-chip communication standard for these buses to be used in high performance microcontrollers. In this paper we have implemented the AMBA AHB (Advanced High-Performance Bus) which is used for high performance transfers among different modules of the microcontroller. AHB supports efficient connection of processor, on chip memory, DMA and off chip external memories. AMBA AHB system bus supports multiple bus masters and slaves. The design is implemented using Verilog HDL and simulated using ModelSim 6.4a. Synthesis for the design is done using Xilinx ISE 14.4.

Keywords-*AMBA;AHB;Arbitration;Burst; HDL ; Simulation;Synthsis;SOC; Verilog;*

*****

I. INTRODUCTION

In an SOC multiple blocks of IPs have to communicate and this is achieved by well-defined communication protocols. They are of great importance as they can have a great impact on the systems performance. Some of the popular protocols used in the industry include PCI, I2C, SPI, AMBA etc.

In this paper we have implemented a widely used industrial protocol AMBA AHB. ARM's AMBA specification defines an on chip communication standard for high performance systems. AMBA is today the de facto standard for processors as it is well documented and can be used without any royalties. It is widely used in RAM controllers, DMA controllers and SOCs.

AHB sits above APB and implements the features required for high performance which are:

- Burst Transactions
- Split Transactions
- Single clock edge operation
- Non – tristate bus implementation.
- Pipelined operation

AHB consists of multiple masters from which one is selected at a time using the arbiter. Decoder is used to select one among the multiple slaves that can be addressed.
Figure 1. Shows how a typical AMBA based system looks like.

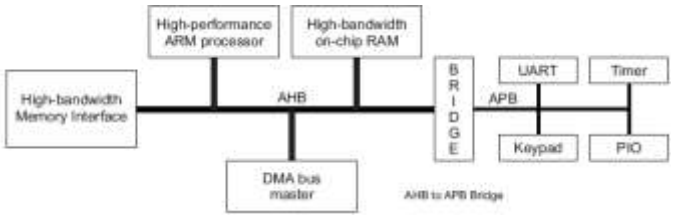

Figure 1 AMBA based system

II. BUS INTERCONNECTIONS

An AHB system consists of multiple masters from which address and data signals are selected. The AMBA AHB bus protocol is designed to be used with a central multiplexer interconnection scheme. In this all bus masters drive out the address and control signals indicating the transfer they wish to perform and the arbiter determines which master has its address and control signals routed to all of the slaves. A central decoder is also required to control the read data and response signal multiplexor, which selects the appropriate signals from the slave that is involved in the transfer.
Figure 2 shows the interconnections between 3 masters and 4 slaves.

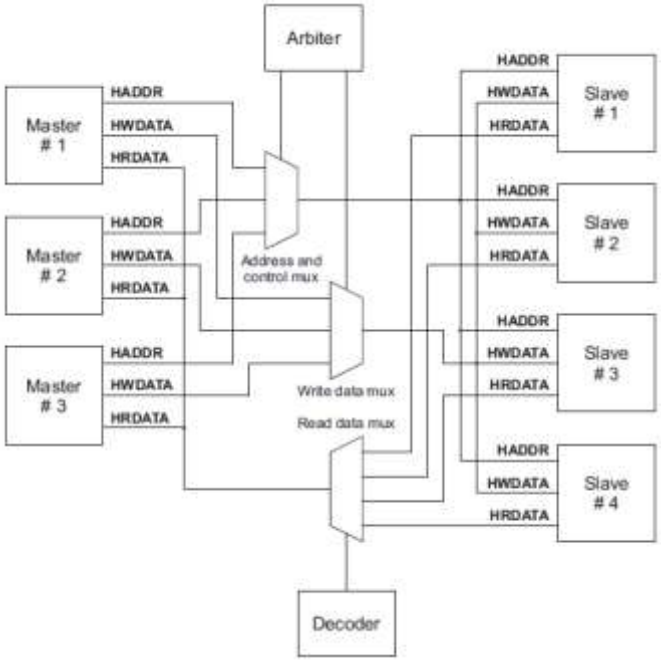

Figure 2 Interconnections

.

III. AHB OPERATION

As seen in figure 2 AHB consists of multiple masters and slaves, which are selected using arbiter and decoder respectively. An AHB transfer starts when a master sends a request to the arbiter for the bus. The arbiter uses a predefined priority algorithm to arbitrate among the requests from different masters. After selecting a particular master it then grants the bus to that master.

The master which is granted the bus can now initiate the transfer. It starts the transfer by driving the address and control signals. These provide various information about the address and the kind of operation (these will be discussed in a following section). The master can also perform burst transfers. There are two kinds of burst transfers that it can perform incremental or wrapping burst transfers.

Each transfer consists of the following two cycles:

- One address and control cycle
- One or more data cycles

Slaves can insert wait states for sampling data but address should be sampled at the same cycle. This can be done using the HREADY signal by the slave. When low this signal indicates the insertion of wait states by the slave. Slave can also reflect the status of the transfer using the HRESP[1:0] signal. Using this slave can provide OKAY, ERROR, RETRY AND SPLIT responses.

IV. AHB COMPONENTS

AMBA AHB consists of the following two global signals which are same for all the components:

- HCLK: this is the clock signal. All operations are performed at the positive edge of this clock.
- HRESETn: this is an active low reset. It is asserted asynchronously and de-asserted synchronously.

The components of an AHB system are:

A. AHB Master

Master is the component that initiates the transfer. It sends the request to the arbiter using the HBUSREQx singal and then starts the transfer after receiving the grant from the arbiter this is done by using the HGRANTx signal.

The transfer is performed in two phases the address phase and the data phase. The address phase is a one cycle long phase in which the address and the control signals are sent this is performed at the start of the transaction after this the data phase is started. In each cycle of the data phase the address of the next data to be transferred is sent, this provides the pipelined operation in AHB which makes it a high performance protocol. The control information sent with the address includes:

- HTRANS[1:0]: Provides the transfer type which can be IDLE , BUSY , SEQ , NON-SEQ . The first transfer in all burst transfers is NON-SEQ and the remaining are SEQ.
- HWRITE: High for write transfer and low for read transfer.
- HSIZE[2:0]: Indicates the size of transfer.
- HBURST[2:0]: Indicates the kind of burst transfer.
- HPROT[2:0] : It is a protection control signal. In this paper it is considered to have a constant value.

In burst transfers the subsequent addresses are calculated by the master according to the size of transfer, also the addresses are required to be size aligned. Bursts should be such that they do not cross 1kB address boundaries.

Apart from these signals a few auxiliary signals required by the master to get information about address, control and data to be sent are given to it.

B. AHB Arbiter

Arbiter is used to select a particular master among the different masters that are currently requesting for the bus. A fixed priority algorithm is used here to arbitrate among the different requesting masters. In this algorithm the master are given a fixed priority and the master having the highest priority among the currently requesting masters is given the grant to the bus. It also asserts the corresponding HMASTER signal for the granted master in the next cycle, which is used by the multiplexer to select the signals of the selected master as shown in figure 2. HSPLITx signal is used by the arbiter give back grant to a master which had been previously removed due to a split response by a split capable slave.

During a burst transfer the master need not hold the HBUSREQx signal as the arbiter will arbitrate again only after the current burst transfer has completed or if the slave has issued a split response.

A master can also assert the HLOCKx signal along with the HBUSREQx signal which would indicate the arbiter to perform a locked transfer and prevent it from granting the bus to any other master. The arbiter indicates this by asserting the HMASTLOCK signal.

C. AHB Slave

Slave responds to the transfers that are initiated by the master. As multiple slaves are possible they are selected by using the HSELx signal from the decoder. They are selected on the basis of HADDR by the decoder. A slave has to be allocated for a minimum of 1kB addresses so that address boundaries are not crossed during a burst transfer.

Slave also provides the response for a transfer, it uses HREADY signal to indicate wait states. A transfer can also be aborted by slave but a master can never cancel a transfer once it has started.

HRESP [1:0] signal is used by the slave to provide OKAY, ERROR, RETRY or SPLIT responses. OKAY can be provided in a single cycle but the others are two cycle responses where in the first cycle HREADY is kept low with the corresponding response on HRESP and in the second cycle HREADY is made high keeping the HRESP same. This helps the master to flush the pipeline.

D. AHB Decoder

Decoder is used to select the slave which is to be used in the transfer. It uses HADDR to select the slave using HSELx signal. This is also used to control the read data mux.

E. AHB Multiplexers

In AHB address and control, write data and read data multiplexers are used. Address and control mux and write data mux are controlled by HMASTER signal from the arbiter and read data mux is controlled by HSELx from the decoder.

V. SIMULATIONS

A. Single master operation

Figure 3 shows the operation of AHB for a single master and slave. The first transaction is an incremental burst 4 write followed by an incremental burst 8 write operation, which is then followed by an incremental burst 8 read operation. The response to this is given as an error response at the 7th cycle.

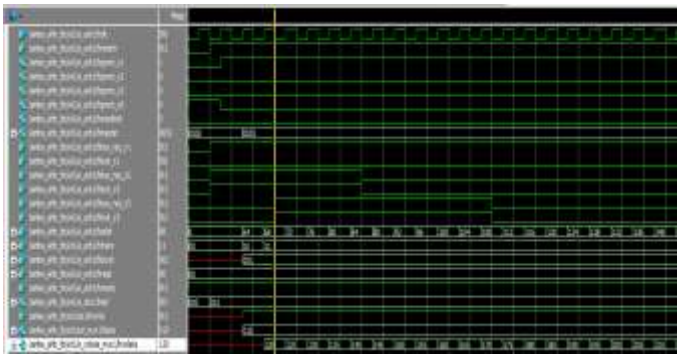

Figure 3 Single master operation

B. Multi master simple write operation

Figure 4 shows the simulation for a single master simple write operation. As shown in the figure master 1 being at the highest priority gets the grant through the arbiter. An undefined length incremental burst is used as the operation in this simulation.

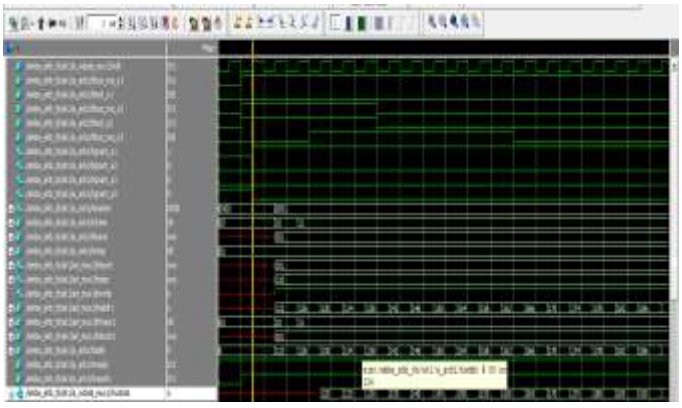

Figure 4 Multi master incremental write operation

C. Burst operations

AHB has the capability of performing different kinds of burst operations. The burst operations available are:

- Single burst
- Undefined length incremental burst (as shown in figure 4)
- Incremental burst of lengths 4, 8, and 16.
- Wrapping burst of lengths 4, 8 and 16.

These are selected using the HBURST[2:0] signals. In any burst transfer the subsequent addresses are calculated using the HSIZE by the master. For a word transfer address is incremented by 4, for a halfword incremented by 2 and by 1 for a byte transfer.

Figure 5 shows the operation for a burst 4 operation for a word transfer followed by a burst 8 operation for a halfword transfer.

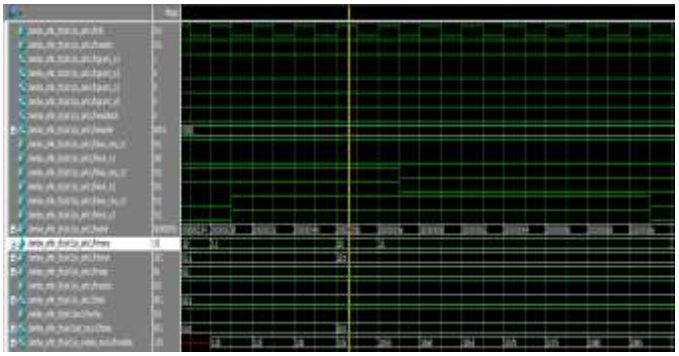

Figure 5 Incremental burst 4 and 8

Wrapping bursts are special kinds of burst in which the address is wrapped around the address boundaries. These are used in cache line fills where the cache line is filled by the master. The address boundary in a wrapping burst can be calculated as:

(No. of Beats) * (size of transfer)

Where beats is the wrapping length (i.e. 4, 8 or 16) and size is given by the no. of bytes in the transfer.

So, for a wrap 8 of word sized transfer the address boundary would be at 8 * 4 = 32 byte boundary.

The expected waveform for such a transfer is shown in figure 6. As seen in the figure the address is wrapped after 0x3C i.e. at the 32 byte boundary to 0x20.

Figure 6 Wrap 8 operation

The waveform for such a transfer is shown in the figure 7 which shows the same result.

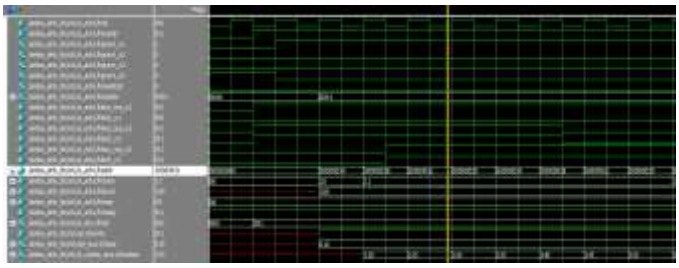

Figure 7 Wrap 8 waveform

VI. SYNTHESIS

The complete design with 3 masters, 4 slaves, arbiter, decoder, address mux, write and read data mux was synthesized using Xilinx ISE 14.4.

Synthesis was done on xc3s500e-4fg320. The synthesis report is shown in figure 8.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	1014	4656	21%
Number of Slice Flip Flops	677	9312	7%
Number of 4 input LUTs	1903	9312	20%
Number of bonded IOBs	647	232	278%
Number of GCLKs	2	24	8%

Figure 8 Synthesis result

VII. CONCLUSION

AMBA defines on chip communication standard for communication among different IPs. AMBA AHB is a popular industry protocol which is used in several applications.

Implementation of AMBA AHB with multi master, slaves, arbiter and decoder was done in this paper using the Verilog HDL. Waveforms were obtained as expected and synthesis was also done using Xilinx ISE.

Further this design can be extended to be used in a complete system where it can be modified to be used with an ARM processor and among the different components connected to it.

REFERENCES

[1] YashdeepGodhal, Krishnendu Chatterjee, Thomas A. Henzinger "Synthesis of AMBA AHB from formal specification", Springer-Verlag, 10 July 2011, Int J Softw Tools Technol Transfer (2013),DOI 10.1007/s10009-011-0207-9.

[2] ARM Ltd. AMBA specification (Rev 2.0)

[3] www.arm.com

[4] Soo Yun Hwang, Dong Soo Kang, Hyeong Jun Park, and Kyoung Son Jhang, Member, IEEE "Implementation of self motivated arbitration scheme for the multilayer AHB busmatrix" " VOL. 18, NO. 5, MAY 2010 IEEE"

[5] D. Flynn's "AMBA: Enabling reusable on-chip designs", IEEE Micro, Vol. 17, no. 4, pages 20–27, July-August 1997.

[6] M. Dubois, Y. Savaria, "A generic AHB bus for implementing high- speed locally synchronous islands", Bois, G. Southeast Conference, Proceedings. IEEE, Page(s). 11 – 16, 2005.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.

[7] www.asic-world.com

[8] Verilog HDL: A Guide to Digital Design and Synthesis, By Samir Palnitkar.