Coverage Driven and Assertion Based Verification of Memory Transactions in Axi Protocol using System Verilog Approach

VASUPALLI KALYANI SRI¹, YELITHOTI SRAVANA KUMAR²

¹PG Scholar, Dept of ECE, Pydah College of Engineering and Technology, Visakhapatnam, AP, India.
²Assistant Professor, Dept of ECE, Pydah College of Engineering and Technology, Visakhapatnam, AP, India.

Abstract: In this paper, a coverage driven verification methodology to verify the AXI bus protocol with its verification environment is proposed. The entire verification environment is carried out using the system verilog based modeling approach the AXI verification scenario includes the read and write transaction phases and coverage mode analysis. The functional verification of the AXI is carried mentor graphics, questa-sim with in the code coverage enabled mode.

Keywords: Verification IP Development, Code Coverage Driven Verification, AXI Protocol, Transaction, System Verilog, Questa-Sim.

I. INTRODUCTION

Now a day’s System on chip (SOC) has many intellectual property core built in them and the proper synchronization between the individual core during the communication is a important task. This modern SOC’s majorly use the common bus protocols like advanced high performance bus (AHB), advanced peripheral bus (APB) and advanced extensible interface (AXI) for their synchronized communication. For this reason the improvement of this kind SOC’s the verification of this technologies is very critical and an important task as it covers 70% time as evaluate to design degree which requires most effective 30% of the time. Due to this large time span for verifying an on chip many engineers are involved in verifying the functional properties and synchronization among them using an inbuilt verification environment referred as Verification IP. The bus protocols used in the current SOC’s are categorized based on their performance and power consumption. The various 3 protocols APB bus structure consumes much less power when compared with AHB bus structure but lacks in performance a compared with AHB. The major thing in AHB bus structure primarily based SOC’s will have slightly higher percentage of power consumption as compared with APB bus structure. So the AXI bus consumes moderate power and gives a better performance as compared with AHB and APB bus structures. So the AXI bus structure can be selected as an alternative bus standard for the modern SOC design. In this paper a system verilog based verification IP has been designed for verifying the best of AMBA protocols (i.e., AMBA AXI) using a coverage driven verification methodology.

II. ARCHITECTURE OF AXI

This section consists of bus architecture of the AXI protocol for the data transfer with respect to READ transaction phase and WRITE transaction phase.

A. Read Transaction Phase

Basically the read transaction phase can be divided into two modes of channels namely read address channel (AR) and read data channel, read response (R). The transaction verification commonly occurs between the master and slave interface that is initiated via the signal in read address channel and read data channels. The architecture for the AXI read transaction uses read address and read data between master and slave interface is shown in fig.1(a).

![Fig.1. (a) AXI read transaction with address read and data read.](image)

![Fig.1. (b) AXI read transaction with address ready and address valid signal.](image)

Here each channel sends a valid and ready signal, based on their responses the communication occurs between the master...
and slave interface. The master slave communication with address valid (AW) and address ready (AR) is shown in the fig.1(b). During the read transaction the read phase master will give the read address request with address and control information based on which slave will respond accordingly.

B. Write Transaction Phase

In the write transaction phase is divided into three modes of channels namely write address channel (AR), write data channel (W) and write response channel (B). The transaction verification commonly occurs between master and slave interface that is initiated via the signals in write address channel and write data channel. The architecture for the AXI write transaction uses write address and write data and write response between master and slave interface is shown in fig.2(a).

Fig.2. (a) AXI write transaction with address write and data write.

Fig.2. (b) AXI write transaction with address ready and address valid signal.

In the entire operation of the write transaction phase master initiates the request for write operation with write data request and with write data correspondingly the slave writes the response to the master. The entire master slave write data transaction with the data valid and data ready signal are shown in the fig.2(b).

III. VERIFICATION IP ENVIRONMENT

This methodology to verify the system components in SOC using the intellectual verification IP concepts is more and more beneficial, because it saves the time for verifying the chip and decreases time to the manufacture without any faults. This form of verification environment allows us to reuse it for any type of component verification. Because of these possibilities it is easy to develop so many cases to verify DUV. Now a day’s the verification procedure is initiated through code coverage and functional coverage to verify the functionalities of the whole design under different scenario.

The verification IP environment consists of a generator (G), mail box, bus functional model (BFM), AXI master, AXI slave, AXI interface and the monitor each module in the verification environment is designed according to its functionality is shown in the fig.3. The generator generates different test cases for different user applications. Mail box connects the generator and bus functional model and mail box acts like a synchronizer which synchronizes the transactions between generator and bus functional model BFM places a major role in verification environment as it receives the transactions from generator and drives in to the AXI interface. This interface interconnects the master and slave which is also responsible for the proper hand shaking during communication all the outputs will be taken from the AXI interface it includes all the signals possible for transaction. Here the monitor keeps checking the valid transaction and hand shaking of signals.

B. Verification Plan

The verification plane tells the information about the properties going to be verified in the design under verification (DUV) with respect to the corresponding test strategies. The properties which might be being verified in the (DUV) are listed below as follows

- Verifying the system connectivity during read and write cycles
- Transaction routing
- Data integrity

For the effective verification of those properties, the coverage driven verification methodology is followed. By using this kind of verification plan able to achieve 100% effectiveness in the verification process.

IV. RESULTS AND DISCUSSION

In the verification process of the AXI master/slave bus protocol system verilog is used for modeling the AXI master/slave with their verification environment the verification environment includes generator mailbox BFM and AXI interface all of those are modeled in system verilog and used for the verification process of this bus protocol mentor graphics questa-sim tool is used to verify the functionality of this design in the code coverage enabled.
Coverage Driven and Assertion Based Verification of Memory Transactions in Axi Protocol using System Verilog Approach

mode to do the entire verification of this bus protocol. During the verification of this bus protocol first the read architecture and write architecture with all of the channels are verified and then checked using verification IP environment in code coverage enable report mode

A. Verification Of Write Architecture

In this verification stage all three write signals named as write address, write data and write response are verified for each transaction. The write address includes AWID, AWLEN, AWADDR, AWSIZE, AWREADY and AWVALID signals alerts toggles for every positive high edge of the global clock. And finally writes the address in the channel. AWID is a write address with ID which represents a particular tag for each write address, it should match the write data WID. During the toggling action of the clock at positive edges with the high enable logic value in WVALID and WREADY, the write data channel acknowledgment will takes place. In addition the write response will happen at the high state of BVALID and BREADY signals. Here the signals AWLEN is of four bit size [0:3] which generates different transactions from one to sixteen. If AWLEN is 0100 then it will have 0101 transactions which mean it will increments the transaction by using one. That is actually illustrated in the waveform clearly at fig.4. From the waveform it is observed that AWSIZE suggests the size of each transaction. The whole write architecture is simulated and verified for all of the signal toggle counts which is clearly shown in the waveform in fig.4.

B. Verification Of Read Architecture

In this read cycle verification, all the read architecture signals ARVALID, RVALID, ARREADY, RREADY, RDATA, RLAST and ARSIZE are verified for each transaction. The read architecture includes two channels i.e; read response and read address channels. The read address channel will initialize its address fetching at the high state of ARVALID and ARREADY signals for every positive edge of the global clock. Similarly after a gap period of delay read response will instanitated to high mode for every positive edge of RVALID and RREADY signals. RLAST indicates the final transaction within the RDATA signal. In addition ARSIZE and ARLEN are equal in comparison to that of write architecture. The whole waveform for the verification level of the read architecture is shown below in fig.5.

C. Coverage Mode Analysis

The identical read and write phase is verified using the code coverage mode analysis. And the coverage driven report is given in the figs.6 and 7. The code coverage mode evaluation is covering approximately the 80% of verification IP using the random test bench based verification methodology. The sample coverage report for the AXI master/slave as follows.
V. CONCLUSION

Assertion based verification is employed to achieve 100% protocol verification. The AXI protocol verification and also the signals employed in every channel are verified and analyzed using the code coverage mode analysis. The most advantage of this kind of verification is using the coverage driven verification wherever the time to market is less and applicable for advanced designs using system verilog.

Future Scope: In future we tend to develop a test case to verify both write and read phases simultaneously different locations of read and write.

VI. REFERENCES


Author’s Profile:

Vasupalli Kalyani Sri, M.Tech, VLSI
Pursuing M.Tech in Pydah College of Engineering & Technology, Visakhapatnam, AP, India.

Yelithoti Sravana Kumar, M.Tech,
Assistant Professor in ECE Department
Pydah College of Engineering & Technology,
Visakhapatnam, AP, India.