Research Journal of Applied Sciences, Engineering and Technology 13(1): 42-47, 2016
DOI:10.19026/rjaset.13.2888
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: December 5, 2015 Accepted: April 22, 2016 Published: July 05, 2016

Research Article

Validating Advanced Extensible Interface Protocol Using Randomized Verification Environment

Avinash Yadlapati and Hari Kishore Kakarla
Department of ECE, Koneru Lakshmaiah Education Foundation, KL University, Vaddeswaram, Guntur (Dist.,), A.P., India

Abstract: The aim of this exploration paper is to approve the Advanced Extensible Interface Bus Protocol utilizing Randomized Verilog Environment. System on-Chip (SOC) design and verification has turned out to be more perplexing. Step by step instructions to verify a configuration has adequately has turned into a challenge. In this study, how to develop the verification Environment of AXI using Verilog HDL and Randomization is presented. The Design under Test (DUT) AXI bus is elaborated, followed by a comprehensive analysis of the verification plan has been made according to the protocol. Integrated verification environment with Functional coverage and constrained arbitrary vectors generation is executed. With this environment, more coverage and minimized time spending verification has been accomplished. AXI or the Advance Extensible Interface is a development of AMBA interface characterized in the AMBA 3 specification. It is focused for high performance and high clock frequency system designs and incorporates qualities to make it suitable for fast sub-micron interconnects. AXI underpins a whole lot of features such as separate address/control and data phases, keeping up unaligned data transfers utilizing byte strobes, burst based transactions through scarcely start address issued, issuing of various addresses without of order response and simple adding of register stages to present timing closure.

Keywords: AMBA, AXI protocol, channel, SOC

INTRODUCTION

As semiconductor development improves, System-On-Chip (SOC) plans are getting the opportunity to be well known. A SOC stage as a general rule includes distinctive outline parts resolved to show application spaces (ARM Limited, Year). Remembering the final objective to ensure the functional precision of a SOC, finding and changing the outline botches at in front of timetable setup stages is crucial in today's equipment improvement streams. The strategy of finding outline bumbles is called "Verification" (Bergeron, 2003). In any case, as setup unconventionality extends, experience shows that various bugs stay undetected regardless of the way that huge resources and time have been given to plan check, which can realize troublesome issues (Brown and Vranesic, 2000). In light of the centrality of ensuring an outlines useful rightness, a considerable measure of effort has been committed to plan confirmation in order to abatement check exertion and advance configuration quality (Spear, 2008). In any case, most existing direct recreation based check frameworks can't guarantee sufficient scope of the outline, achieving undetected bugs. Additionally, they can't accurately handle non-deterministic issues that are ending up being more basic nowadays. Remembering the deciding objective to crush the imprisonment of test check and arbitrary confirmation techniques, Limitation irregular check (CRV) to confirmation plan rightness has been proposed as a promising distinctive alternative for direct test recreation. To improve plan quality and decrease confirmation exertion, we propose Scope Driven Limitation Arbitrary Check in this piece (ARM Limited, 2007). With advances in semiconductor development, useful confirmation continues staying one of the vital challenges in SOC outlines today. As the estimations in industry reviews show up, regardless of the path that up to 70% of endeavor resources have been given to practical check, only 33% of SOC outlines are spot on the key pass and 75% of all arrangement imperfections are inferable from legitimate or useful bugs on account of inadequacies in utilitarian confirmation (ARM Limited, 2004). The Scope Driven Confirmation (CDV) joins programmed test era, self-checking, test seats and scope estimations to in a

Corresponding Author: Avinash Yadlapati, Department of ECE, Koneru Lakshmaiah Education Foundation, KL University, Vaddeswaram, Guntur (Dist.,), A.P., India, Tel.: +91 9989835092

general sense diminish the time spent confirming a configuration and accomplish the scope objective. It tracks progress with utilitarian scope to ensure test arrangement criteria are met moreover ensures corner cases are hit. This strategy adds to the general confirmation methodology by ensuring all around described properties that can guarantee the precision in and between squares. This improves measured nature of confirmation and engages finding blunders in the outline process.

MATERIALS AND METHODS

AMBA: The Progressed Microcontroller Transport Engineering (AMBA) is an advancement of ARM Constrained and is an open standard, on-chip interconnects outline for the affiliation and supervision of all around arranged squares in a Framework on-Chip (S0C). It supports right-first-time improvement of multi-processor arranges with inconceivable bits of knowledge of controllers and peripherals.

AMBA ASB/AHB/APB: The AMBA AXI tradition is assailed at prevalent, high-recurrence framework outlines and joins a numeral of qualities that make it appropriate for a rapid submicron interconnects. The SOC engineering is appeared in the Fig. 1. It is an AMBA based SOC that consolidate the AXI transport with two experts and three slaves.

As of late, two new details for AMBA bus, Multi-Layer AHB and AMBA AXI, are characterized. Multi-layer AHB gives more adaptable interconnect design (grid which empowers parallel access ways between various masters and slaves) as for AMBA AHB and keeps the AHB protocol unaltered. AMBA AXI depends on the idea point-to-point connection.

AMBA AXI: The AXI convention relies on upon burst based exchange. Every operation has tended to and shapes data on the territory channel to delineate the character of the data to be exchanged. In compose exchanges, in which the information exchange from the expert to the slave, the AXI convention other than have compose reaction channel to permit the slave to send an affirmation to the expert the finishing of the compose exchange as appeared in Fig. 2.

Interface and interconnect: A critical framework contains different pro and slave gadgets related together in the midst of some kind of interconnect, as showed up in Fig. 3.

In numerous frameworks, the location channel transmission capacity limit responsibility is extensively not precisely the data channel information exchange transfer speed prerequisite. Such frameworks can fulfill a decent harmony between framework exhibitions and interconnect many-sided quality by utilizing a common location transport with various information transports to empower parallel information exchanges.

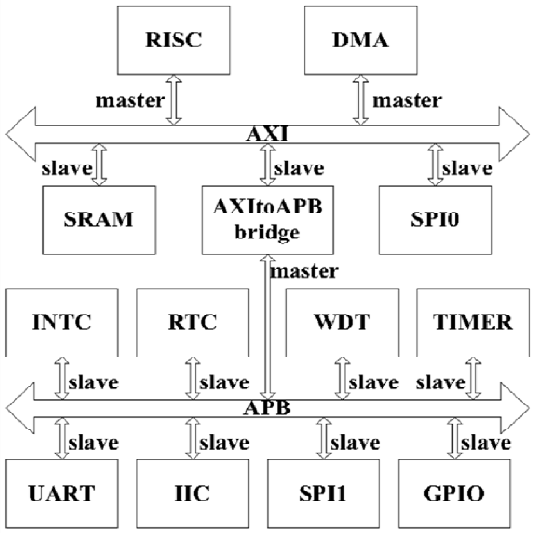

Fig. 1: AMBA AXI based system

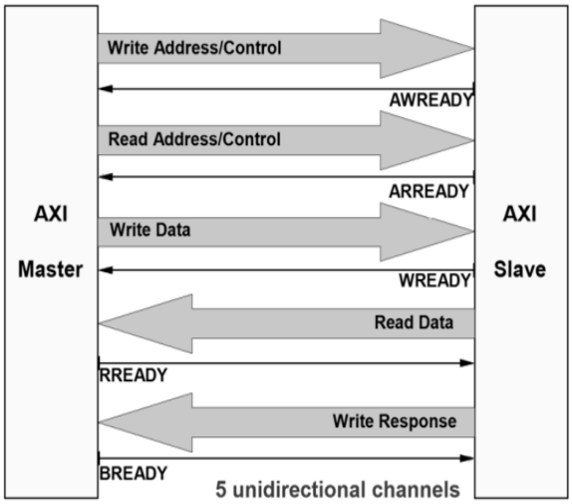

Fig. 2: AMBA AXI channels

AMBA AXI-4:

System components: Figure 4 represents the architecture of AXI Protocol. Here are the requirements of the system components:

- Master
- AMBA AXI-4 Interconnect
- Arbiters
- Decoders
- Slave

AXI expert: AX1 expert starts exchange on the transport. Client configurable handles to keep up various components of expert.

AXI slave: AXI slave reacts to the exchange began by the expert. Totally organized hold up states and slave mistake reaction.

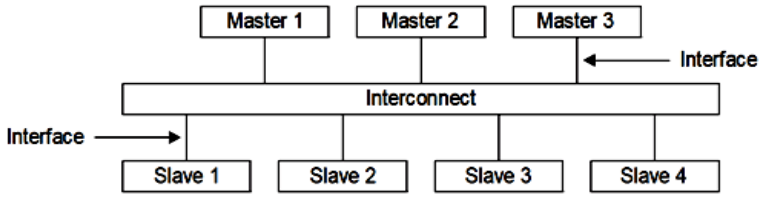

Fig. 3: Interface and interconnect

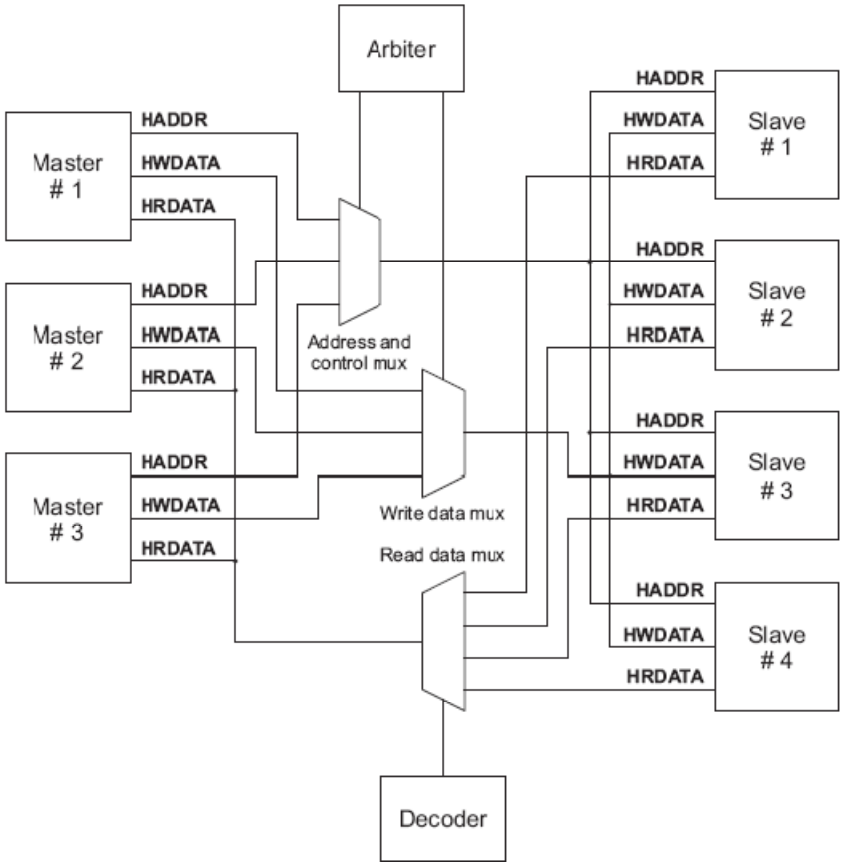

Fig. 4: AXI architecture

AXI interconnect/referee: AXI mediator can take a couple experts without a moment's delay and parleys for the transport using configurable need plan.

AMBA AXI tradition is troubled at high-execution, high-recurrence frame work plans and it consolidates different elements that detail it suitable for rapid sub-micron interconnects.

And also the information exchange convention, the AXI convention consolidates elective enlargements that cover motioning for low-control operation.

Specification: Table 1 shows signal descriptions of AMBA AXI-4 Protocol.

Verification of AXI protocol: Figure 5 shows AMBA AXI-4 bus design to which three masters M0, M1, M2 and six slaves S0, S1, S2, S3, S4, S5 are interfaced. The master initiates write or read transaction and with a transaction ID and address it selects the slave. For write operation master puts the data on write data bus and slave puts the data for read operation. Each channel has handshaking signals, write and read data can be terminated by LAST signals.

RESULTS

Reproduction is carried on QuestaSim of Coach representation, with Verilog as confirmation dialect. The experiments keep running for some corners and the reenactment log and scope reports are examined. The principle point of interest of Framework Verilog is re-ease of use of check code for different test situations moreover interconnects to different AXI slave IP pieces. The assorted experiment examples are used to

Table 1: Signal descriptions of AMBA AXI-4 protocol

Signal	Source: Master/slave	Input/Output	Description
Aclk	global	input	global clock signal
Aresetn	global	input	global reset signal
Awid [3:0]	master	input	write address id
Awaddr [31:0]	master	input	write address
Awlen [3:0]	master	input	write burst length
Awsize [2:0]	master	input	write burst size
Awburst [1:0]	master	input	write burst type
Awlock [1:0]	master	input	write lock type
Awcache [3:0]	master	input	write cache type
Awprot [2:0]	master	input	write protection type
Wdata [31:0]	master	input	write data
Arid [3:0]	master	input	read address id
Araddr [31:0]	master	input	read address
Arlen [3:0]	master	input	read burst length
Arsize [2:0]	master	input	read burst size
Arlock [1:0]	master	input	read lock type
Arcache [3:0]	master	input	read cache type
Arprot [2:0]	master	input	read protection type
Rdata [31:0]	master	input	read data
Wlast	master	input	write last
Rlast	slave	output	read last
Awvalid	master	output	write address valid
Awready	slave	output	write address ready
Wvalid	master	output	write valid
Rvalid	slave	output	read valid
Wready	slave	output	write ready
Bid [3:0]	slave	output	write response id
Rid [3:0]	slave	output	Read response id
Bresp [1:0]	slave	output	write response

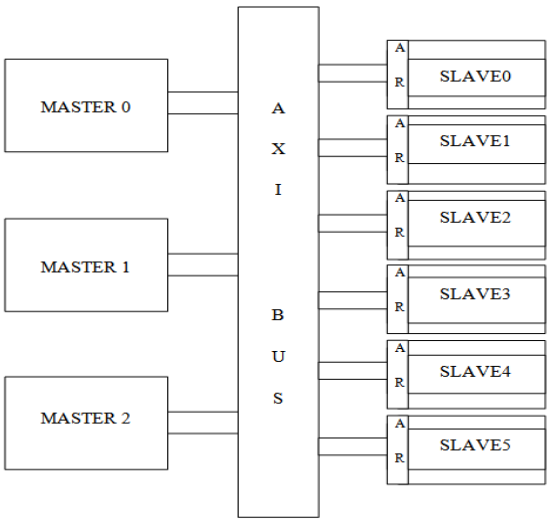

Fig. 5: AXI bus architecture

confirm the AXI slave. To perform distinctive exchanges like compose and read operations in the middle of expert and slave, the linked info position and their qualities went to conjure a capacity. Recreation is completed in Modelsim (from Coach Representation) apparatus and Verilog is utilized as programming dialect.

DISCUSSION

There have been many papers published (Ranga *et al*., 2012; Pradeep and Laxmi, 2014) on this generic Advanced Extensible Peripheral Interface Protocol. Since this specification is generic and used as open standard, the users can implement their own methodologies for Design and Verification. The AMBA (AHB) was the most widely used bus protocol and the Advanced Extensible Interface bus has recently started being used for many SOC Designs due to the following reasons:

- Separate address/control and information stages.
- Support for unaligned information exchanges.
- Ability to issue numerous extraordinary locations.
- Out-of-request exchange fruition.

The work carried out in the research article is to validate the AXI protocol functionality using Verilog HDL and the important point is that we have randomized the complete environment in addition to directed test cases.

CONCLUSION

Throughout the years AMBA has supported to offer cutting edge answers for SOC interconnects. With the moderately late expansion of the AXI 4 convention family ARM keeps up a vivacious advantage in the field of superior SOC AMBA AXI4 is an attachment and play IP convention discharged by ARM, characterizes both transport determination and a hardware self-sufficient strategy for outlining,

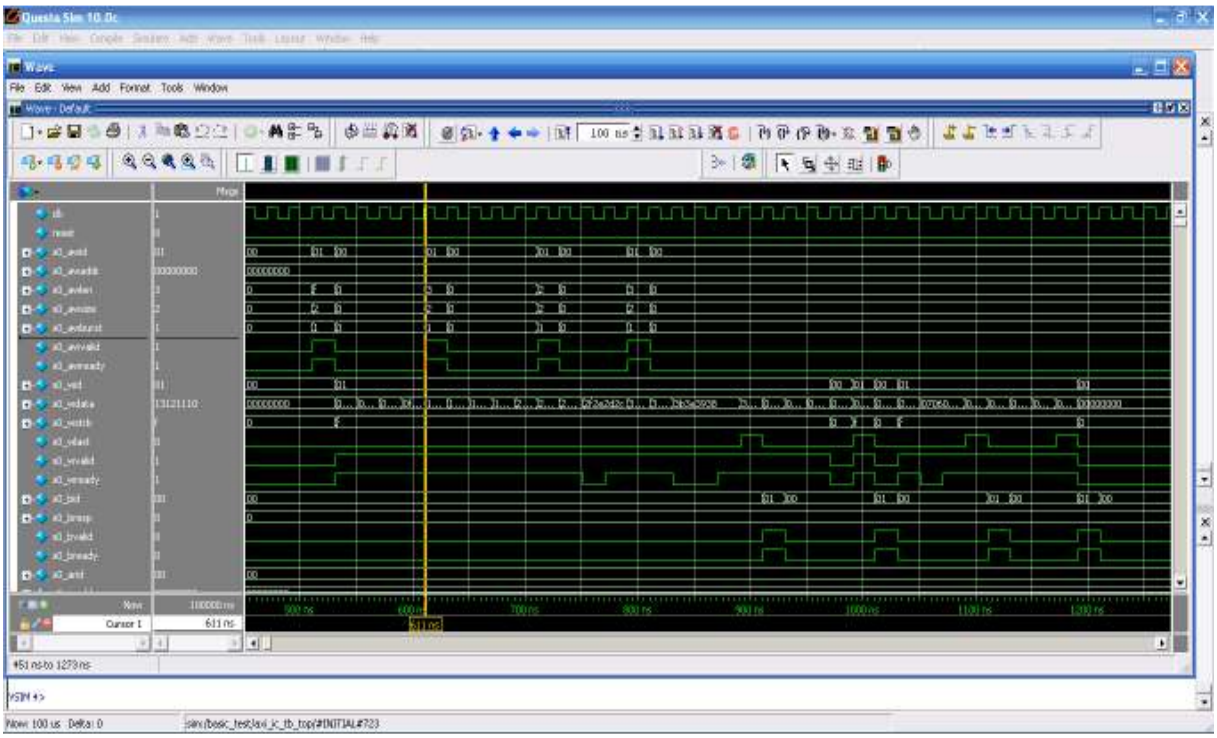

Fig. 6: Basic test write transaction

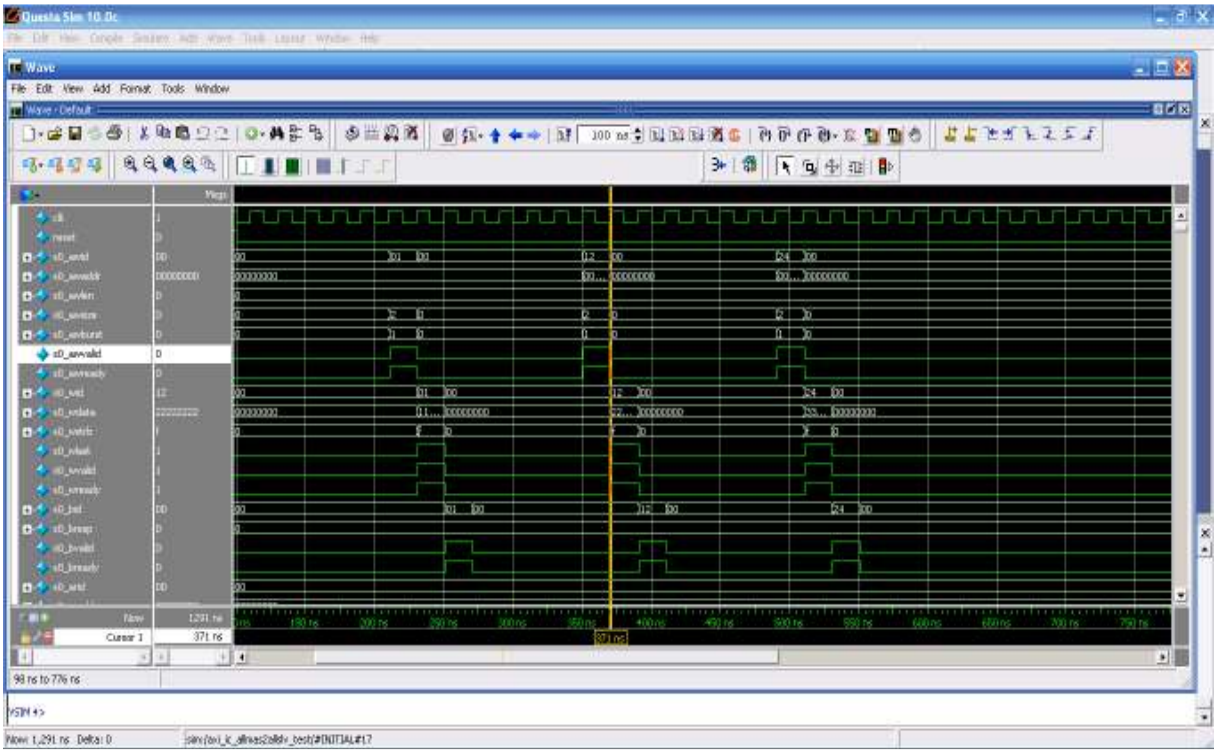

Fig. 7: All masters to all slaves write transaction

executing and testing adjusted high-mix inserted interfaces. The information to be perused or kept in touch with the slave is whispered to be given by the expert and is perused or kept in touch with a conscientious location area of slave amid decoder. In this study, slave was demonstrated in Verilog with working recurrence of 100MHz and recreation results were appeared in Fig. 6 to 8. To perform single read

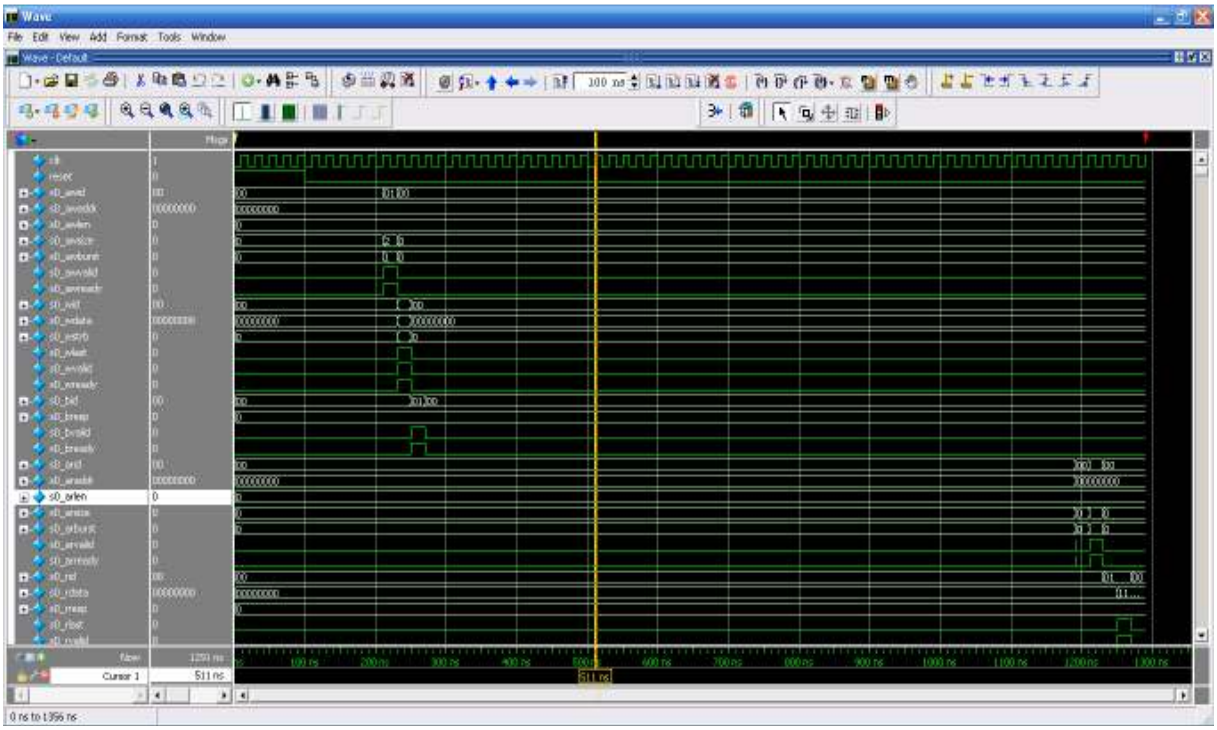

Fig. 8: Write followed by read transaction

operation it devoured 160ns and for the single composes operation 560ns. This study comprehends the complete utilitarian check procedure of complex ASICs a SOC's and it offers chance to attempt the most recent confirmation techniques, programming ideas like Item Situated Programming of Equipment Confirmation Dialects and refined EDA apparatuses, for the astounding check. The usefulness of AXI slave IP is checked by procedure CDV and the code scope and useful scope are watched utilizing spread focuses and cross spread focuses. The CDV system can be utilized to make reusable test seats effectively. The outcome depict 99.96% of usefulness is secured and normal code scope of 91.56% (with proclamation scope of 95%, Branch scope of 100% and switch scope of 90.45%).

REFERENCES

ARM Limited, Year. AMBA AXI Protocol Specification Version 2.0. ARM Limited.

ARM Limited, 2004. PrimeCell AXI Configurable Interconnect (PL300). Revision: r0p1. Technical Reference Manual. ARM Limited, Retrieved from: http://infocenter.arm.com/help/topic/com.arm.doc.ddi0354b/DDI0354.Pdf.

ARM Limited, 2007. AMBA Design Kit Technical Reference Manual. Revision: r3p0, ARM Limited, Retrieved from: http://infocenter.arm.com/help/index.jsp?Topic=/com.arm.doc.ddi0243c/index.html.

Bergeron, J., 2003. Writing Testbenches: Functional Verification of HDL Models. Kluwer Academic Publishers, Springer, US.

Brown, S.D. and Z.G. Vranesic, 2000. Fundamentals of Digital Logic with VHDL Design. McGraw-Hill, Boston.

Pradeep, S.R. and C. Laxmi, 2014. Design and verification environment for Amba Axi protocol for Soc integration. Int. J. Res. Eng. Technol., 3(3): 338-343.

Ranga, A., L. Hari Venkatesh and Venkanna, 2012. Design and implementation of AMBA-AXI protocol using VHDL for Soc integration. Int. J. Eng. Res. Appl., 2(4): 1102-1106.

Spear, C., 2008. SystemVerilog for Verification: A Guide to Learning the Testbench Language Features. Springer, New York.