Available online at www.ijiere.com

International Journal of Innovative and Emerging Research in Engineering

e-ISSN: 2394 - 3343 p-ISSN: 2394 - 5494

A Slave VIP for Verification of AMBA AXI-3 Master DUT in UVM

Sagar R. Gajjar[a] and Mr.Ankit A. Patel[b]
[a]Vishwakarma Govt. Eng. Collage, Chandkheda, Gujarat, India
[b]PerfectVIPs, Baleshwar Avenue, S-G Highway, Ahmedabad, India

ABSTRACT:
The ARM has developed AMBA is open standard protocol usefull for interconnecting different module of microprocessor with each other. Here in this paper I have developed slave VIP of AMBA AXI3 for verifing the master DUT of AMBA AXI3. Slave VIP has been checked for different scenarios. Different features live slave configuration error and other error scenarios are also added in this slave VIP. In the end of this paper results are shown where slave VIP has been verified under different situations and as it is working gud we can use this VIP for the verification of AMBA AXI3 master DUT.

Keywords: System on Chip(SoC), Intellectual Properties(IP), Design Under Test(DUT), Universal Verification Methodology(UVM), Advanced Microcontroller Bus Architecture(AMBA)

I. INTRODUCTION

The AMBA AXI protocol is targeted at high-performance, high-frequency system designs and includes a number of features that make it suitable for a high-speed submicron interconnect. The objectives of the latest generation AMBA interface are to:

- be suitable for high-bandwidth and low-latency designs
- enable high-frequency operation without using complex bridges
- meet the interface requirements of a wide range of components
- be suitable for memory controllers with high initial access latency
- provide flexibility in the implementation of interconnect architectures
- be backward-compatible with existing AHB and APB interfaces.

The key features of the AXI protocol are:

- separate address/control and data phases
- support for unaligned data transfers using byte strobes
- burst-based transactions with only start address issued
- separate read and write data channels to enable low-cost Direct Memory Access (DMA)
- ability to issue multiple outstanding addresses
- out-of-order transaction completion [6]

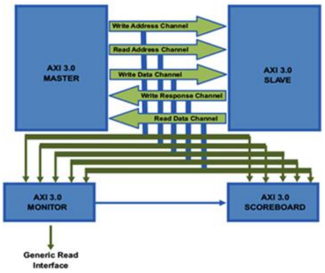


Fig1. AXI Channels
Ref. : www.brinbailey .com

II. ARCHITECTURE

Here as aim is to verify AXI3 MASTER one has to design SLAVE VIP where there is all the components connected together and can automate the verification just by writing different test cases. Space where we connect master to slave is known as verification environment. In this verification environment one slave agent is instantiated which contains driver and receiver class where all logic of slave module is implemented. Configuration class is there to set the global parameter and to get them where ever it is required so one can avoid that hierarchical path of passing variable instead can just get at any class . Monitor is there to check all the data transfer between master and slave and to provide this data to scoreboard for further checking. So except master dut whole environment is known as slave VIP which we have to develop for verification of AXI3 MASTER module but before connecting DUT one has to do vip to vip verification and should be able to calculate functional coverage of that vip so that when its connected to DUT we can know the accuracy. Except scoreboard all the components are designed at this moment. Scoreboard is under construction and after scoreboard just functional coverage is remaining.

Receiver Class :

Basically what ever the data master will send first come to the receiver of the slave. As all know there are three channels from which master can send data to slave. Receiver should always be in free running mode means when ever master want to talk to slave it should always respond. Say in address writes channel master wants to send 20 bursts and then on write data channel it will one by one send data for that 20 burst. Slave must always be ready to work with all the scenarios. AXI is burst based protocol and every burst is having its owned field so when ever any bunch of burst coming at address write channel we have to store them with their respective ids and then when data comes on wdata channel based on wid information there should be check whether we are having any address write information regarding this id or not and accordingly steps are taken. Here flow chart for slave receiver is given bellow which will give the exact idea about the slave receiver is working.

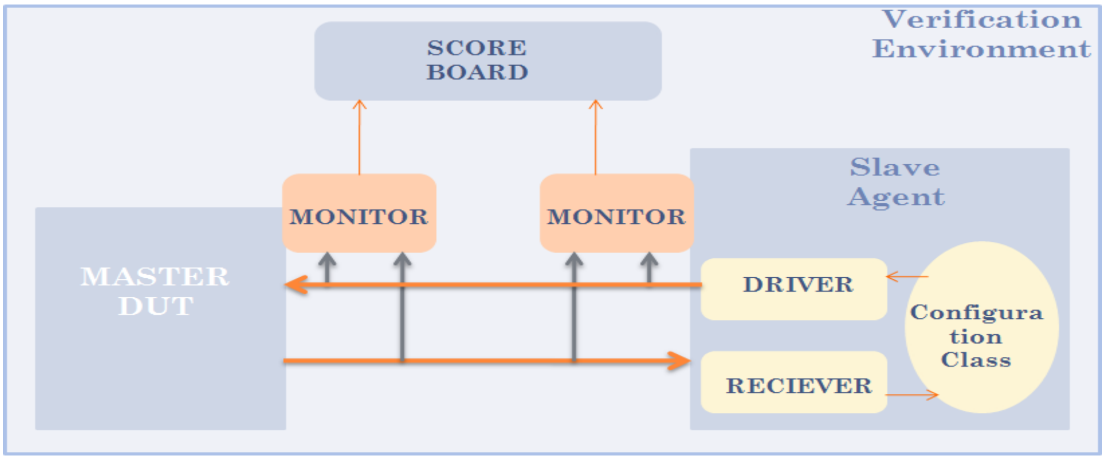


Figure 2. Verification environment

Driver Class :

What ever the information received at receiver based on that driver will take further action whether its for read or write transaction. Detailed flow chart for read and write transaction is shown in following section. For memory write operation driver first checks the address and data and then accordingly writes the memory and sends the write response on the write response channel. While for memory read operation based on information received at ar-channel slave driver sends the data to master from the specified memory location with read response for ever reading of data.

Monitor Class :

Monitor class is there to monitor the activity done at slave receiver interface it tracks the all the activities and further when its required it sends that information to scoreboard or any other component if its required .

Configuration Class :

Configuration class contains parameters which need to change again and again so if configuration class feature is not there then every time one has to go to the code and change it accordingly but by using configuration class just by changing the value of that variable one can easily over write it just by getting this config class where ever change need to be followed. An easy way to comply with the conference paper formatting requirements is to use this document as a template and simply type your text into it.

III. FLOW CHARTS :

Memory Write

When wvalid and wready both are high data will be received at slave receiver. Receiver will decode the strobe and will make check_allow bit equal to one. As check_allow bit is equal to one incoming data will go to salve driver and if it's the first data of the burst awaddr will not increase and it will go to check whether awaddr is in the memory range or not if it in the memory range then memory write operation will take place or data will be discarded and error response will be sent to master. If it is not the first data of the burst then awaddr will be increase by one and again same address check and memory write operation will take place. If it is the last data of the burst then slave driver will send write response to master.

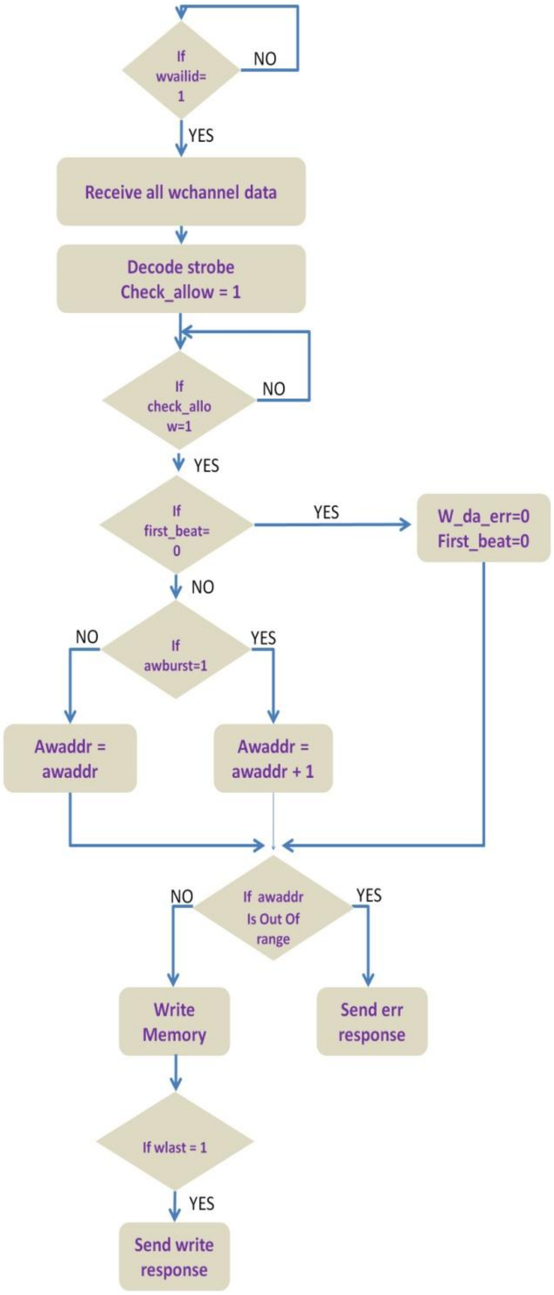


Figure 3. Memory write

Memory Read :
Similarly for memory read write operation if arvalid and arread are high slave receiver will receiver all the data from ar-channel and will increase read_allow now at slave driver side if read_allow is greater than zero read from the memory operation will take place. Arlen indicates the number of data each burst require from the memory so by decreasing arlen check whether it is zero or we can calculate each burst length. If it is the last data then slave driver will make rlast high so that master will come to know that it is the last of the current read burst. Unlike write transaction here for every data slave has to send the read response. After ever last read data read_allow will be decrease by one.

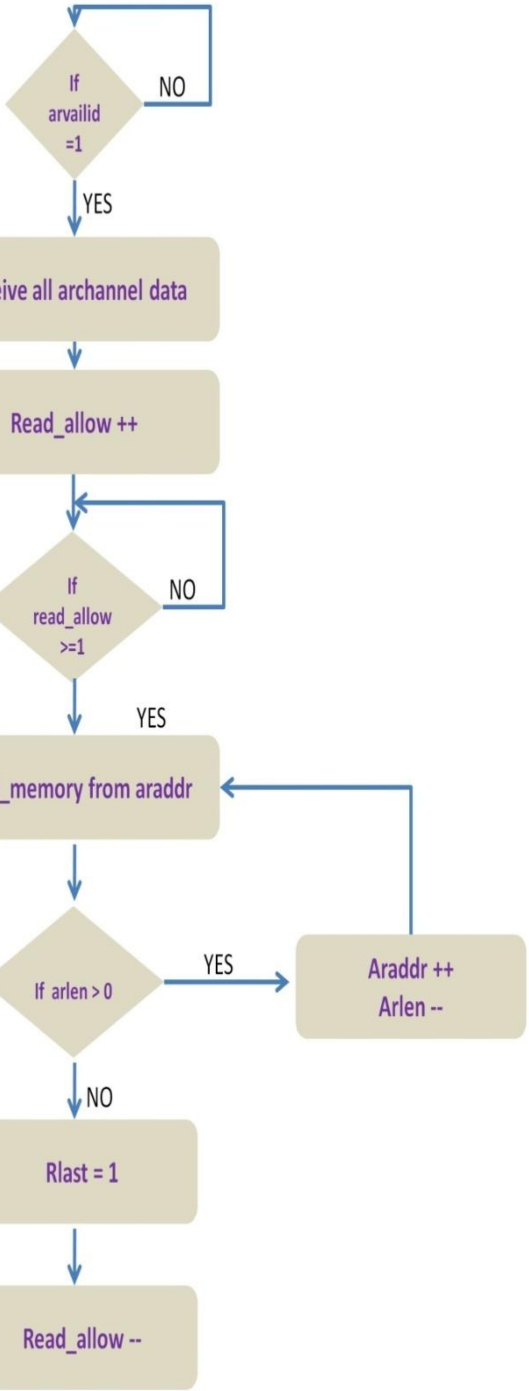


Figure 4. Memory read

IV. TEST CASES :

Basic Read Write :

Here in this section basic read write operation of AXI3 is shown.Fig.19 is of aw-channel which collects address write information for memory write operation so when wdata comes on w-channel as show in fig.20 slave driver will use the previous aw-channel information to store the wdata based on its id. Similar approach is there for read data. Master will provide from which memory address how much data it wants to read through ar-channel accordingly slave driver will drive the data to master through r-channel.

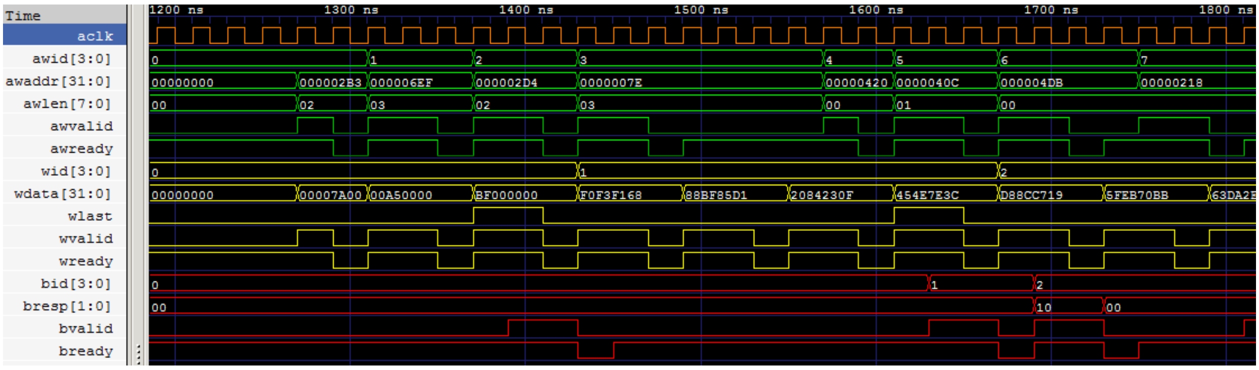


Figure 5. Basic write

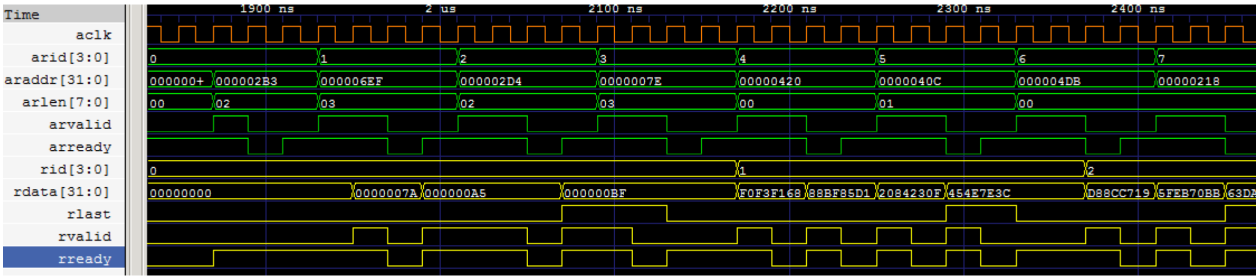


Figure 6. Basic read

Without Multiple Outstanding :

When multiple outstanding is zero means address and data comes one by one as shown in fig.7 means after completion of data for previous address new address is send on aw-channel.

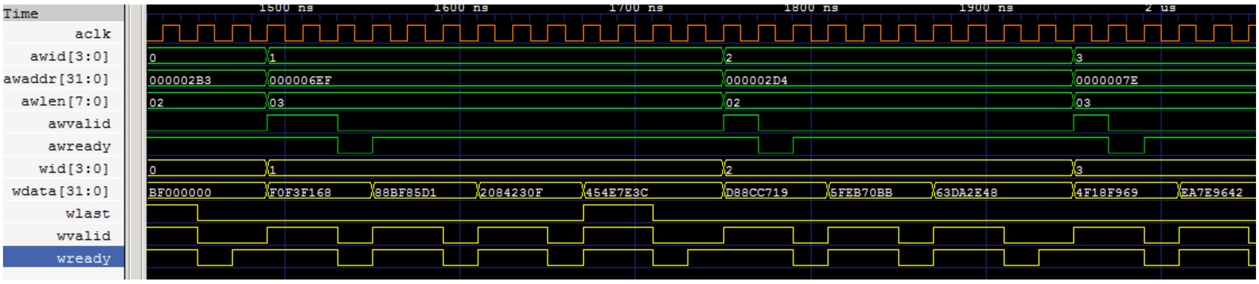


Figure 7. Write without multiple outstanding

With Multiple Outstanding :

When multiple outstanding is one all address information on aw-channel and ar-channel comes together where data comes one by one on w-channel as shown in fig.8 . there is no synchronization between address channel and data channel here.

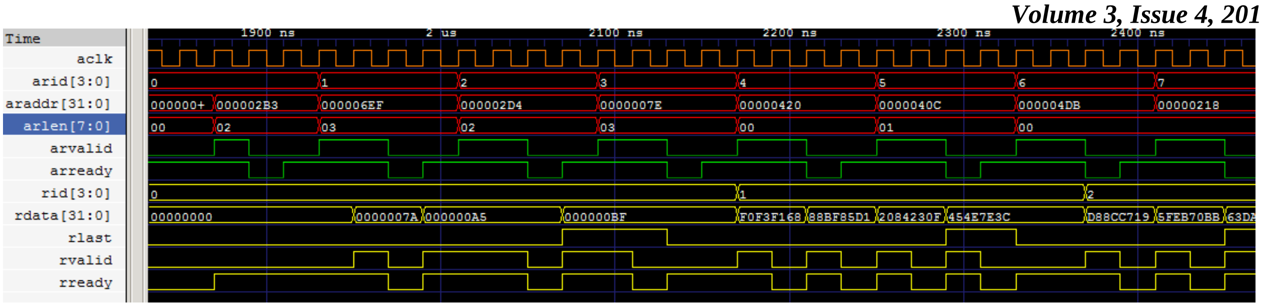


Figure 8. Read with with multiple outstanding

Read first :

Here as shown in fig.9 reading operation is taking place before writing operation so it is obvious that there will be no data into the memory so during reading operation as one can see from the fig.9 on rdata line there is always zero data because when memory is initialized all memory address are initialized with zero.

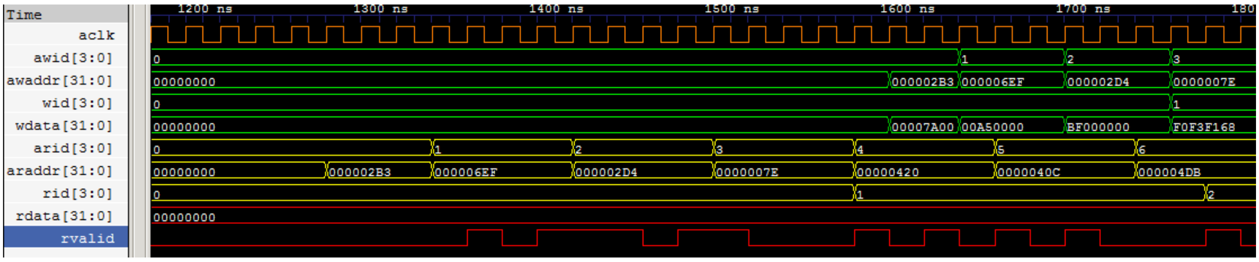


Figure 9. Read before write

Awlen error :

When master is saying that it is going to send awlen no. of data during data cycle and if it send less number of data for that burst then slave will not consider it as burst is completed so it will not send the write response for any of that incomplete bursts. As shown in fig.10 for no burst bvalid line is going high means there is no write response for this kind of burst having awlen error.

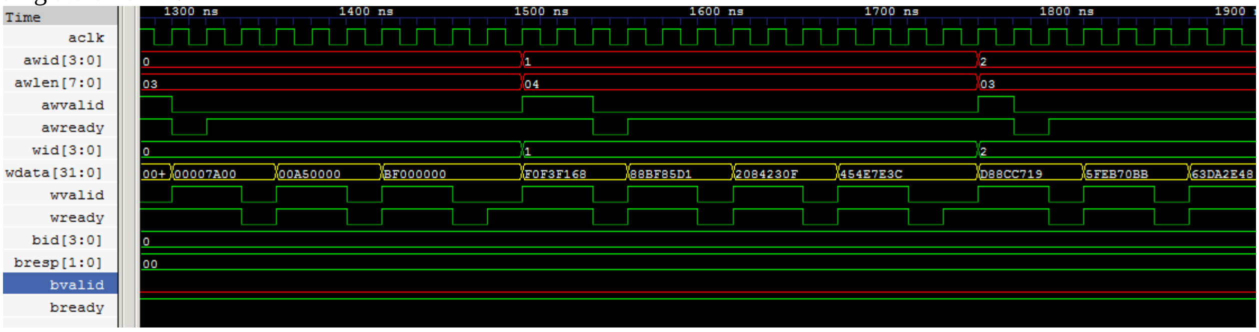
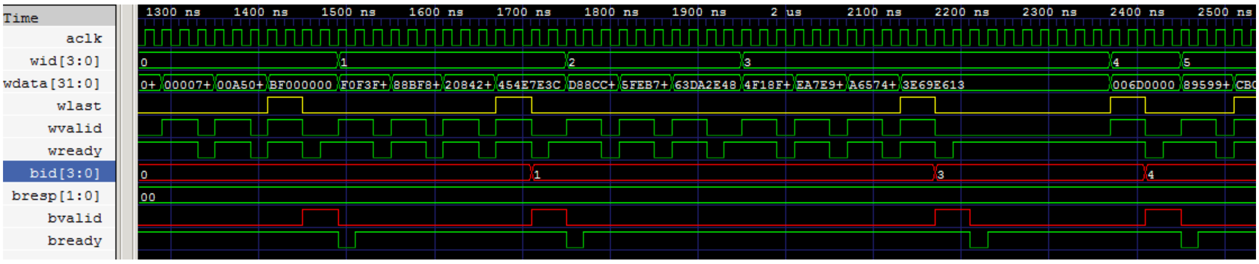


Figure 10. Awlen error

Dependancy on wlast signal of write response :

After every write burst slave has to send the write response to the master at the last transaction data of that write burst so response will always be generated at the negedge of wlast signal sent by master. By chance if master does not generate wlast signal for some particular write burst slave will not send write response for that perticulalr write burst as shown in fig. 11 .



Figue 11. No wlast

Address out of range :

In this test case address sent on aw-channel is out of range of memory address for memory write operation so when its corresponding data will come slave driver will discard the data as there is no such memory address to store the data in memory. Now if you try to read from the memory there will be no data as there is no write operation is done on that memory so in reading there will be zero data always from memory.

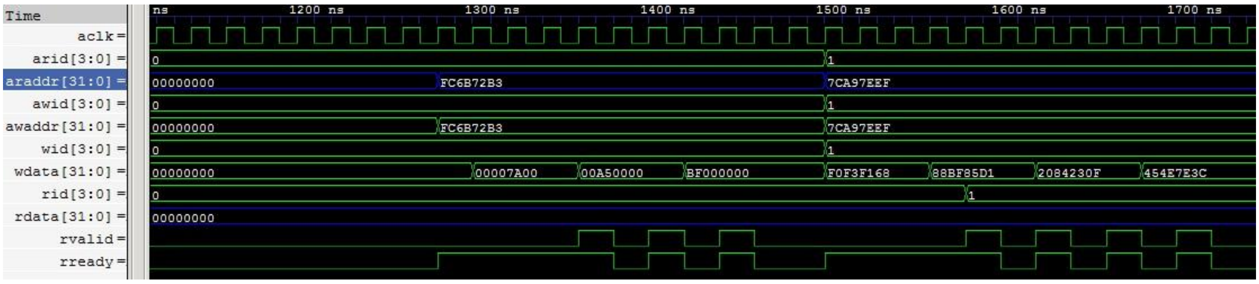


Figure 12. Awaddr is out of the memory range

Slave configuration errors :

Slave configuration error is a specific feature from the slave side. Using this feature one can send the manually generated error response for memory read or write operation. For write transaction by choosing the burst no. one can send the error for that burst while for read transaction on particular transaction of particular burst one can send read error response in terms of either slave error or decoder error or exokay or any two of them or even any three of them. Here in this test case for write and read transaction these errors are generated.

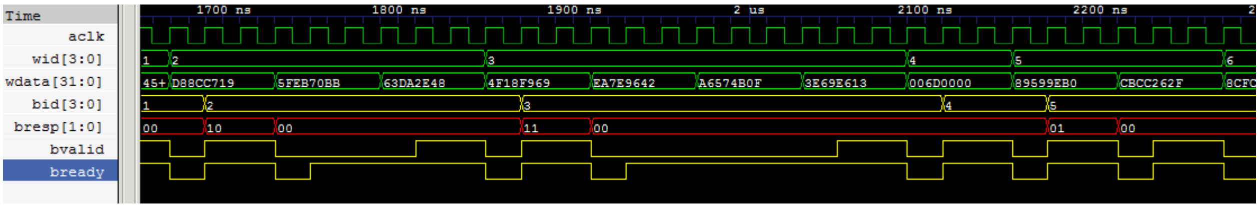


Figure 13. Slave write config error

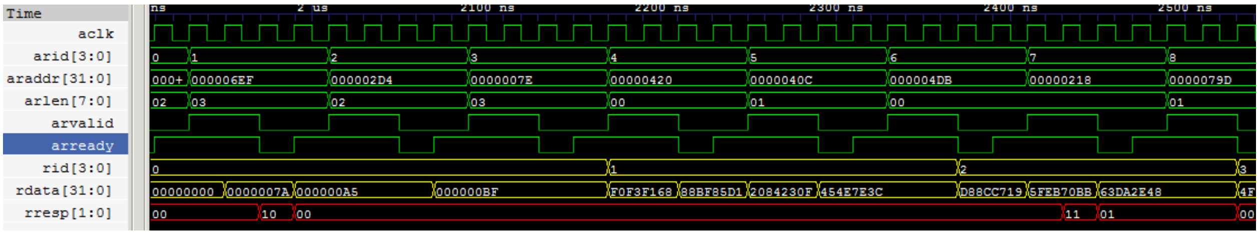


Figure 14. Slave read config error

V. Conclusions

In this paper complete slave VIP has been developed. It has been checked with different test cases and as shown in simulation results its giving the desired outputs. So in a manner we can say that we can use this slave VIP for verifying the master DUT of AXI3..

Acknowledgment

I would like to express my profound gratitude and deep regards to my guides of Perfect VIPs. Mr. Ankit Patel, Asic designer and Mr. Krunal Patel, Asic designer for their constant support, guidance and motivation. I would like to express my deep sense of gratitude to my internal guide Dr. R. A. Thakker, Head of Department, EC Department, VGEC, Chandkheda. I am grateful to him, as he provided me guidance and support. He has been extremely supportive at every stage and showed faith in me.

REFERENCES

[1] Bartley, M.G. ; Elixent Ltd., Bristol, UK ; Galpin, D.;Blackmore, T., " A comparison of three verification techniques: directed testing, pseudo-random testing and property checking " , Design Automation Conference, 2002. 39th Proceedings.

[2] Marat Teplitsky, Amit Metodi, and Raz Azaria ,"Coverage Driven Distribution of Constrained Random Stimuli" Cadence Design Systems, Israel in 2013.

[3] Han Ke, Deng Zhongliang and Shu Qiong , "Verification f AMBA Bus Model Using SystemVerilog" ,Electronic Measurement and Instruments, 2007. ICEMI '07. 8th International Conference on Aug. 16 2007-July 18 2007.

[4] Xu Chen, Zheng Xie, and Xin-An Wang, Development of Verification Environment for AXI Bus Using System Verilog in International Journal of Electronics and Electrical Engineering, Vol. 1, No. 2, pp. 112-114, June 2013. doi:10.12720/ijeee.1.2.112-114

[5] Shrivastav, A. ; Dept. of Electron. & Comm., Priyatam Inst. of Technol. & Manage., Indore, India ; Tomar, G.S. ; Singh, A.K."Performance Comparison of AMBA Bus- Based System-On-Chip Communication Protocol" Communication Systems and Network Technologies (CSNT), 2011 International Conference on 3-5 June 2011.

[6] AMBA AXI3 LRM by ARM.