

Design and Verification of AMBA AHB-Lite protocol using Verilog HDL

Sravya Kante^{#1}, Hari KishoreKakarla^{*2}, Avinash Yadlapati^{#3}

1, 2 Department of ECE, KL University
Green Fields, Vaddeswaram-522502, A.P, India

^{#1}kante.sravya@gmail.com

^{*2}kakarla.harikishore@kluniversity.in

^{#3}avinash.amd@gmail.com

Abstract—The SOC plan confronts a crevice between generation limit and time to market weights. The outline space develops with changes underway limits as far as measure of time to plan a framework utilizing these abilities. On one hand, shorter product life cycles are forcing an aggressive reduction of the time-to-market, fast simulation capabilities are required for coping with the immense design space that is to be explored; these are specially needed during early stages of the design. This need has driven the improvement of exchange level models, which are theoretical models that have been designed to run much quicker than synthesizable models. The pressure for faster executing models extends especially to the frequently reused communication libraries. AMBA AHB-Lite addresses the requirements of high-performance synthesizable designs. It is a transport interface that provides support to a solitary transport ace and gives elite data transfer capacity. This paper describes the system level modelling of the Advanced HighperformanceBus Lite (AHB-Lite) subset of AHB which is a part of the Advanced Microprocessor Bus Architecture (AMBA). It also includes the design and verification of AHB-Lite protocol for sequential and non-sequential (increment and wrap of different burst sizes) transfers.

Keyword- AMBA (Advanced Microcontroller Bus Architecture), AHB-Lite (Advanced High Performance Bus-Lite), SoC (System on chip).

I. INTRODUCTION

The bus protocol utilized by the CPU is a vital part of co-verification since this is the primary correspondence between the CPU, memory, and other custom equipment. Installed plan frameworks by and large and particularly SoC will be held under the practical and ecological requirements. Following the planned framework will keep running on a very much determined working environment, practical strict prerequisites might be particularly characterized. Natural confinements, then again are more diverse: for instance, to minimize the cost, foot shaped impression, and force utilization. Because of the adaptability of a SoC plan, ARM processors use distinctive transport conventions relying upon when the core was designed for achieving the set goals, it includes analyzing a multidimensional space plan. The degrees of freedom stem from the process element types and characteristics, their allocation, the mapping of functional elements to the process elements, their interconnection with busses and their scheduling. The enormous complexity of these protocol results from tackling high-performance requirements. Protocol control can be distributed, and there may be non-atomicity or speculation.

AHB-Lite systems based around the Cortex-M™ processors ARM delivers the DMA-230 "micro" DMA controller [13]. ARM delivers DMA controllers for both high-end, high-performance AXI systems based on the Cortex-A™ and Cortex-R™ families and cost-efficient AHB systems built around Cortex-M™ and ARM9 processors.

The CoreLink Interconnect family includes the following products for AMBA protocols:

- Network Interconnect (NIC-301) for AMBA 3 systems including support for AXI, AHB and APB
- Advanced Quality of Service (QoS-301) option for NIC-301

The third generation of AMBA characterizes at high performance, high clock frequency system designs and includes features which make it very suitable for high speed sub-micro meter interconnect. In the present papersome discussion is made on the family of AMBA and also briefly described the AHB-Lite Protocol. Further the design and the verification of AHB-Lite protocol with different test cases are shown.

II. AMBA PROTOCOLS

Figure 1 shows the different protocols performances from the time of initialization [9].

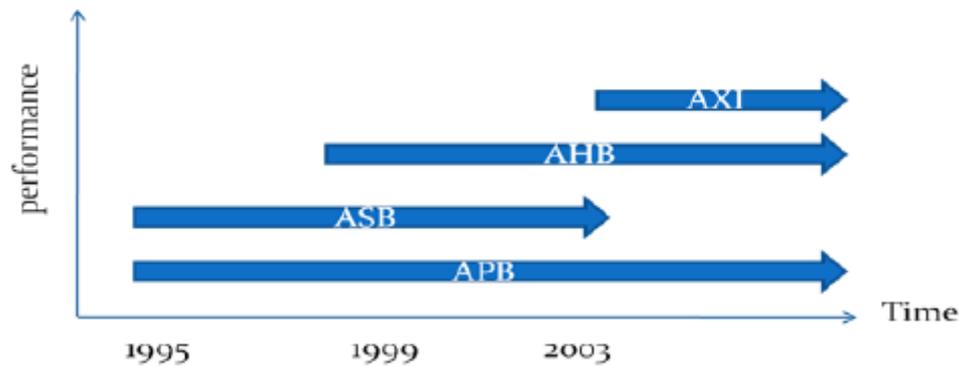


Fig 1. Protocols of AMBA[9]

- APB (Advanced Peripheral Bus) mainly used as an ancillary or general purpose register based peripherals such as timers, interrupt controllers, UARTs, I/O ports, etc. It is connected to the system bus via a bridge, helps reduce system power consumption. It is also easy to interface to, with little logic involved and few corner-cases to validate.
- AHB (Advanced High Performance Bus) is for high performance, high clock frequency system modules with suitable for medium complexity and performance connectivity solutions. It supports multiple masters.
- AHB-Lite is the subset of the full AHB specification which intended for use where only a single bus master is used and provides high-bandwidth operation.

III. AHB-LITE PROTOCOL SYSTEM

AMBA AHB-Lite protocol is designed for high-performance synthesizable designs. It is a transport interface that provides a simplex transport mechanism and ensures high speed data transfer capacity. AHB-Lite implements the features required for high-performance, high clock frequency systems including: [1]

- Burst Transfers
- Single-Clock Edge Operation
- Non-Tristate Implementation
- Wide Data Bus Configurations, 64, 128, 256, 512, And 1024 Bits.

The most common AHB-Lite slaves are memory devices, interfaces and high bandwidth peripherals. Although low-bandwidth peripherals can be incorporated as AHB-Lite slaves, for system performance reasons they typically reside on the AMBA Advanced Peripheral Bus (APB). Bridging between this higher level of bus and APB is done using AHB-Lite slave, also known as an APB bridge.

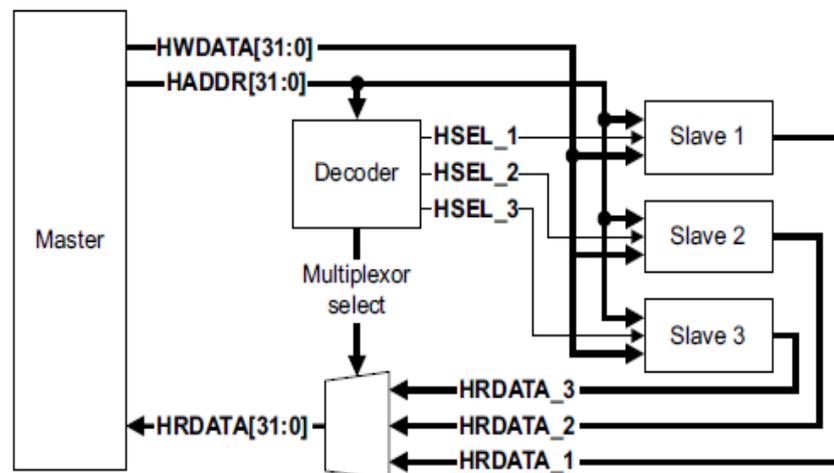


Fig 2. AHB-Lite block diagram

Figure 2 shows a single master AHB-Lite system design consisting of one master and three slaves. The bus interconnect logic consists of one address decoder and a slave-to-master multiplexor. The decoder monitors the address from the master so that the appropriate slave is selected and the multiplexor routes the corresponding slave output data back to the master. The main component types of an AHB-Lite system are:

- Master

- Slave
- Decoder
- Multiplexor
- AHB-Lite Master: It provides address and control information to initiate read and write operations
- AHB-Lite Slave: The slave responds to the transfers initiated by masters in the system. The slave uses the **HSELx** select signal from the decoder to control when it responds to a bus transfer.
- Decoder: It is used to decode the address of each transfer and provides a select signal for the slave that is involved in the transfer. It also provides a control signal to the multiplexor.
- Multiplexor: A single centralized multiplexor is required in all AHB-Lite implementations that use two or more slaves. A slave-to-master multiplexor is required to multiplex the read data bus and response signals from the slaves to the master. The decoder provides control for the multiplexor.

3.1 OPERATIONS OF AHB-LITE:

The master starts a transfer by driving the address and control signals. These signals provide information about the address, direction, width of the transfer, and indicate if the transfer forms part of a burst. Transfers can be: [11].

- single
- incrementing bursts that do not wrap at address boundaries
- Wrapping bursts that wrap at particular address boundaries.

TABLE I. Transfer Types

Transfer Type	Description	HTRANS[1:0]
IDLE	No data transfer is required	00
BUSY	Enables masters to insert idle cycles in the middle of a burst	01
NON-SEQUENTIAL	Address and control signals are unrelated to the previous transfer.	10
SEQUENTIAL	Address and control signals are related to the previous transfer.	11

The write data bus moves data from the master to a slave, and the read data bus moves data from a slave to the master. Every transfer consists of: [2]

- **Address phase** one address and control cycle
- **Data phase** one or more cycles for the data.

A slave cannot request that the address phase is extended and therefore all slaves must be capable of sampling the address during this time. However, a slave can request that the master extends the data phase by using **HREADY**. This signal when LOW, causes wait states to be inserted into the transfer and enables the slave to have extra time to provide or sample data.

The slave uses **HRESP** to indicate the success or failure of a transfer

Table II. Transfer Responses

HRESP	HREADY	
	0	1
0	Transfer pending	Successful transfer completed
1	ERROR response, first cycle	Error response, second cycle

IV. IMPROVEMENTS OVER AHB

The AHB-Lite specification differs from AHB specification in the following ways [2]:

- There exists only one master. There is only one source of address, control, and write data, so no Master-to-Slave multiplexor is required.
- There is no arbitration technique used here so the signals associated with the arbiter are not used.
- Master has no **HBUSREQ** output. If such an output exists on a master, it is left unconnected.
- Master has no **HGRANT** input. If such an input exists on a master, it is tied HIGH.
- Slaves must not produce either a Split or Retry response.
- The AHB-Lite lock signal is the same as **HMASTLOCK** and it has the same timing as the address bus and other control signals. If a master has an **HLOCK** output, it can be retimed to generate **HMASTLOCK**.
- The AHB-Lite lock signal must remain stable throughout a burst of transfers, in the same way that other control signals must remain constant throughout a burst.

V. AHB-LITE ADVANTAGES

The advantage of using the AHB-Lite protocol is that the bus master does not have to support the following cases [15]:

- Losing ownership of the bus. The clock enable for the master can be derived from the **HREADY** signal on the bus.
- Early terminated bursts. There is no requirement for the master to rebuild a burst due to early termination, because the master always has access to the bus.
- Split or Retry transfer responses. There is no requirement for the master to retain the address of the last transfer to be able to restart a previous transfer.

VI. SIMULATION RESULTS

Simulation is being carried out on NCSim which is trademark of Cadence, using Verilog as hardware verification language. The test cases are run for multiple operations. The different test case patterns are used to verify the AHB-Lite slave.

To perform various transactions like write and read operations between master and slave, the concatenated input format and their values passed to invoke a function. Simulation is carried out in ModelSim (from Mentor Graphics) tool and Verilog is used as programming language.

A. Single Burst

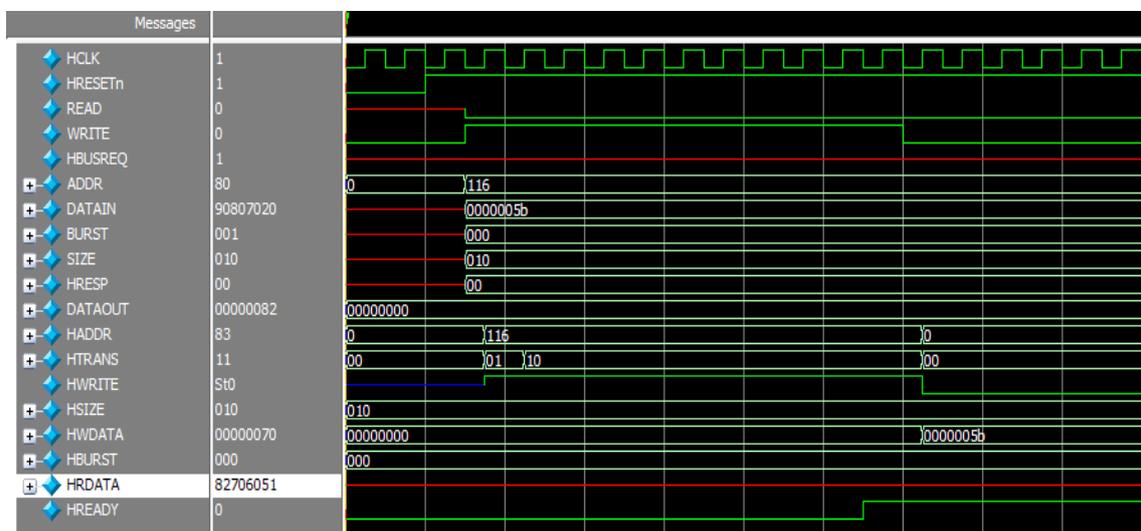


Fig 3. Write Transfer

An AHB-Lite transfer consists of two phases:

Address lasts for a single HCLK cycle unless it's extended by the previous bus transfer. Data that might require several HCLK cycles. Use the HREADY signal to control the number of clock cycles required to complete the transfer. WRITE controls the direction of data transfer to or from the master. Therefore, when:

- HWRITE is HIGH, it indicates a write transfer and the master broadcasts data on the write data bus, HWDATA [31:0]
- HWRITE is LOW, a read transfer is performed and the slave must generate the data on the read data bus, HRDATA [31:0].

The simplest transfer is one with no wait states, so the transfer consists of one address cycle and one data cycle. Fig. 3 shows a simple write transfer and Fig. 4 shows a simple read transfer.

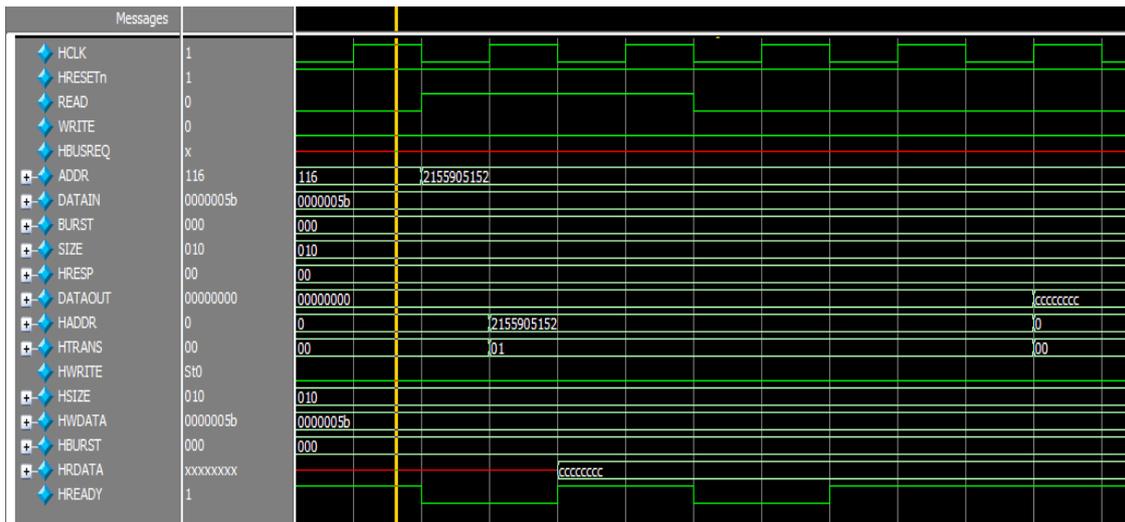


Fig 4. Read Transfer

B. Four-Beat Incrementing Burst (INCR 4)

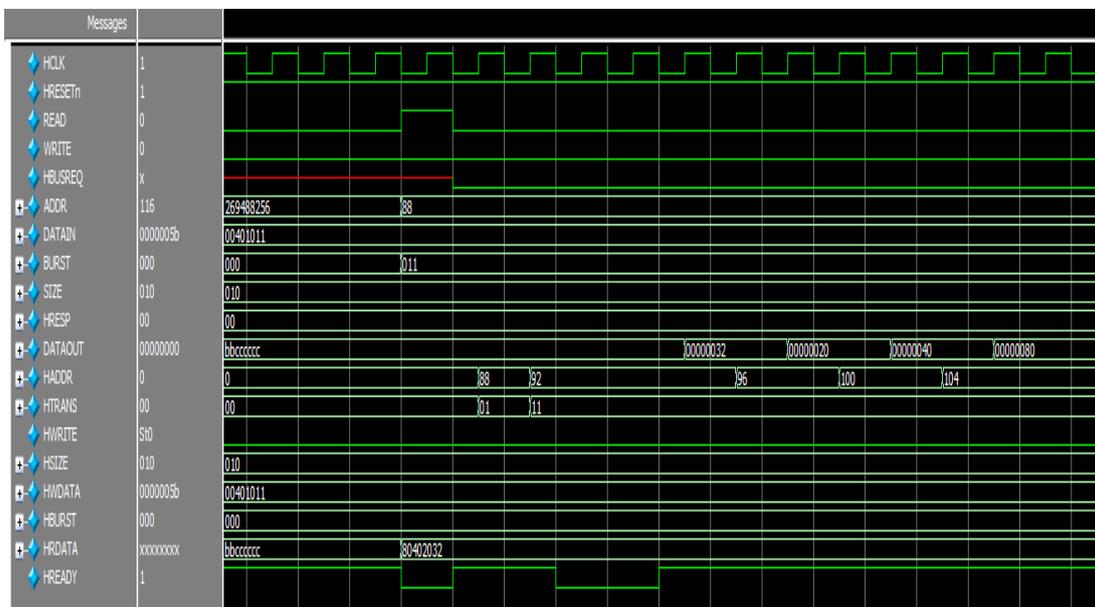


Fig 5.INCR4 Write Transfer

Fig.5 shows a write transfer using a four-beat incrementing burst, with a wait state added for the first transfer. In this case, the address does not wrap at a 16-byte boundary and the address 100 is followed by a transfer to address 104.

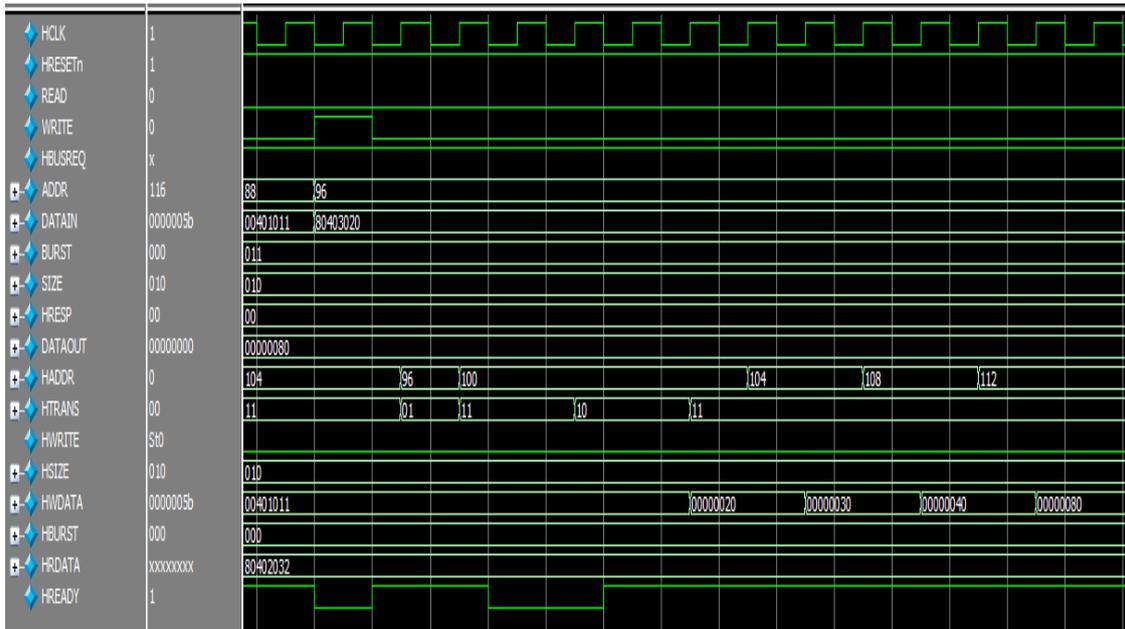


Fig 6.INCR4 Read Transfer

Fig.6 shows a read transfer using a four-beat incrementing burst, with a wait state added for the first transfer. In this case, the address does not wrap at a 16-byte boundary and the address 108 is followed by a transfer to address 112.

C. Four-Beat Wrapping Burst (WRAP 4)

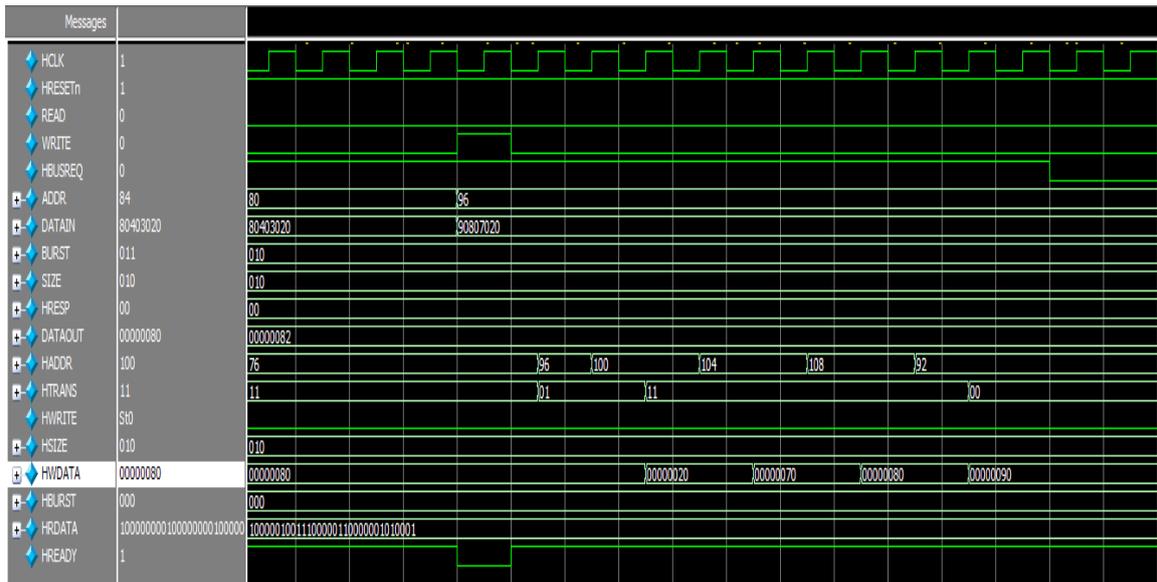


Fig 7.WRAP4 Write Transfer

Fig.7 shows a write transfer using a four-beat wrapping burst, the burst is a four-beat burst of word write transfers, the address wraps at 16-byte boundaries, and the transfer to address 108 is followed by a transfer to address 92.

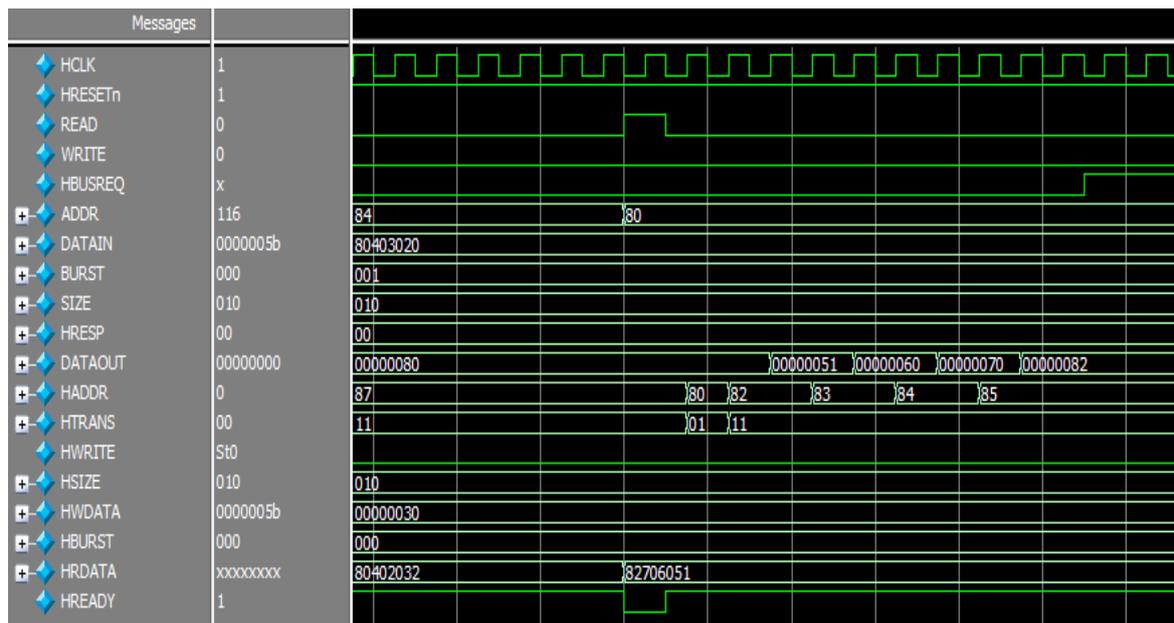


Fig 10.Undefined Length Read Transfer

The second burst is a read consisting of three word transfers starting at address 0x5C. These transfer addresses increment by four. Fig 10 shows incrementing bursts of undefined length of read transfer.

VII.CONCLUSION

In this paper a general definition for AHB-LITE protocol which has high performance represents a significant advance in the capabilities of the ARM AMBA bus on-chip interconnect strategy, by providing a solution that reduces latencies and increases the bus bandwidth. AHB-Lite is fully compatible with the current AHB specification. Each of the major AHB-Lite transfers like Single (non-sequential) transfer and sequential transfers (wrap and increment burst of 4 bit data size) have been verified with individual test cases. All these test cases are run and verified using Mentor Graphics ModelSim simulator.

ACKNOWLEDGEMENT

The authors would like to thank the entire semiconductor team at CYIENT for their immense support and motivation. Without their guidance this paper would not have seen the light of day. A special thanks to Mr. Ram Gollapudi for allowing the authors to use the company's valuable resources to complete the project.

REFERENCES

- [1] ARM, "AMBA_3_AHB-Lite", available at <http://www.arm.com/>.
- [2] ARM, "AMBA Specification (Rev 2.0)", available at <http://www.arm.com/>.
- [3] ARM, "AMBA AXI Protocol Specification", available at <http://www.arm.com/>.
- [4] Samir Palnitkar, "Verilog HDL: A Guide to Digital Design and Synthesis", Second Edition.
- [5] Chris Spear, SystemVerilog for Verification, New York : Springer, 2006.
- [6] <http://www.testbench.co.in>.
- [7] http://www.doulos.com/knowhow/sysverilog/ovm/tutorial_0.
- [8] <http://www.inno-logic.com/resources/VMM.html>.
- [9] Akhilesh Kumar and Richa Sinha, "Design and Verification Analysis of APB3 Protocol with Coverage" IJAET Journal, Vol. 1, Issue 5, pp. 310-317, Nov 2011.
- [10] http://en.wikipedia.org/wiki/Advanced_Microcontroller_Bus_Architecture.
- [11] <http://books.google.co.in/books>.
- [12] <http://www.asicguru.com>.
- [13] <http://www.arm.com>.
- [14] Bergeron, Janick. Writing testbenches: functional verification of HDL models. s.l.: Springer, 2003.
- [15] <http://infocenter.arm.com>.
- [16] Verilog LRM 2005 www.scribd.com/doc/6755724/Verilog-LRM.
- [17] <http://www.testbench.in>.
- [18] Verilog HDL Synthesis by J.Baskar.
- [19] Springer - SystemVerilog for Verification by CHRIS SPEAR Synopsys, Inc.