Online Design Internship Report

Design of AHB2APB Bridge

Name: Varun Ram S

Reg no: 20BAC10038

Email Id: varun.ram2020@vitbhopal.ac.in

College: Vellore Institute of Technology, Bhopal

Project Guide: Mr. Manjunath NL

Report Submitted on:

29 June 2023

Acknowledgement

In the first place, I would like to record my gratitude to my Project Guide and Instructor, **Mr. Manjunath NL** sir**,** for his supervision, advice and guidance from the very start of the project as well as giving an extraordinary experience throughout the work. Above all and most needed, his guidance was beyond amazing for completing this project work on “Designing AHB2APB Bridge”.

I would also like to extend my gratitude to **Mr. Sivakumar PR** sir, CEO and Founder of Maven Silicon for his guidance and teaching in the Verilog Programming lectures.

I’d also not like to forget to thank the Support team, instructors and other staff members of Maven Silicon who were crucial in solving queries and offered great support by timely responses in the email and WhatsApp group.

Lastly, I would like to thank the management of my college Vellore Institute of Technology Bhopal, as they played a crucial role in encouraging me to enroll in this amazing Online Design Internship Course.

Table Of Contents

Abstract

The design and implementation of an Advanced High-performance Bus (AHB) to Advanced Peripheral Bus (APB) bridge will be thoroughly discussed in this study. The main goal of this project is to provide an effective interface between the AHB and APB buses that will allow high-performance devices linked through the APB bus to exchange data and control signals. This study will start by providing a thorough block diagram of the top module, which symbolizes the main structure of the AHB to APB bridge. The block diagram will provide a visual representation of the overall design by highlighting the important elements and how they link to one another. The detailed explanation of each sub-block inside the bridge will outline the precise function it performs in facilitating communication between the AHB and APB buses. The output waveform, which shows the signals and data flowing through the AHB to APB bridge during operation, will also be examined in this report. In order to assess the bridge's efficacy in allowing communication between the high-performance devices linked through the APB bus, a greater knowledge of the timing, synchronization, and overall performance of the bridge is made possible by the waveform analysis. This report will serve as a complete reference for comprehending the project's goals and results by meticulously outlining the design, architecture, and functionality of the AHB to APB bridge. It will offer insightful information about the complex workings of the bridge, assisting in the assessment of its efficiency, dependability, and general performance.

Aim of the Project:

This project aims to design a bridge between the Advanced High-performance bridge (AHB) and the Advanced Peripheral bus (APB) which are two different distinct buses defined within the Advanced Bus Microcontroller Architecture (AMBA) Specification. The bridge actually allows Communication between High Performance devices and Low Peripheral Devices

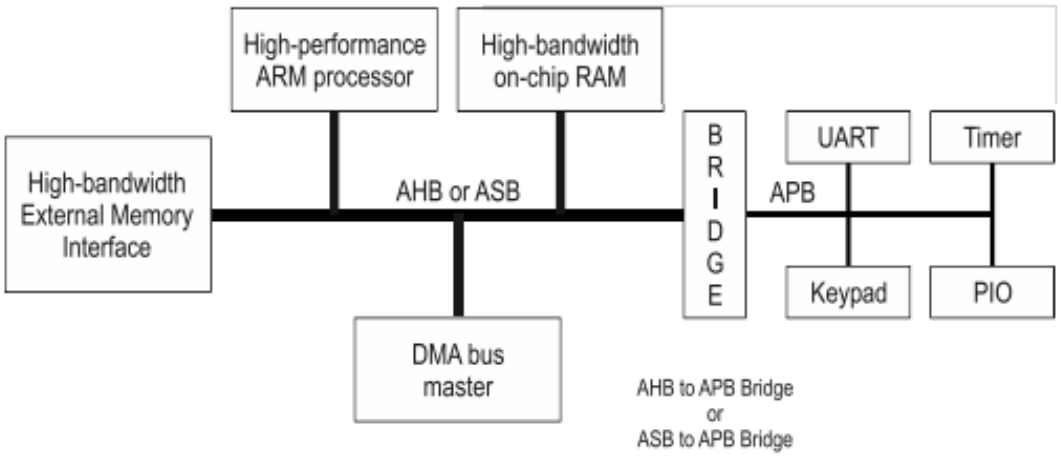
Objectives of the Project:

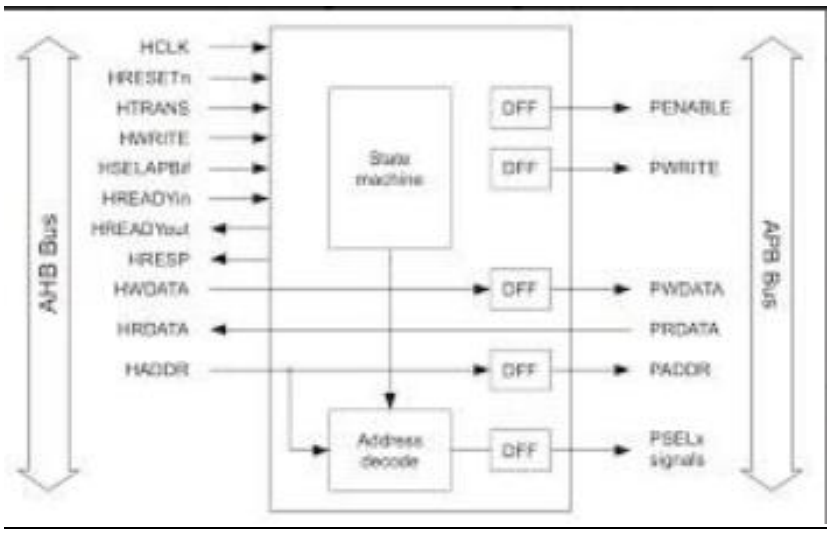
1) Study the Top Module Block Diagram
2) Write Verilog code for different modules of the APB2AHB Bridge
3) Verify the same using a Test Bench.
4) Generate relevant output waveforms for better visualization and understanding.

Basic Software and Synthesis tools:

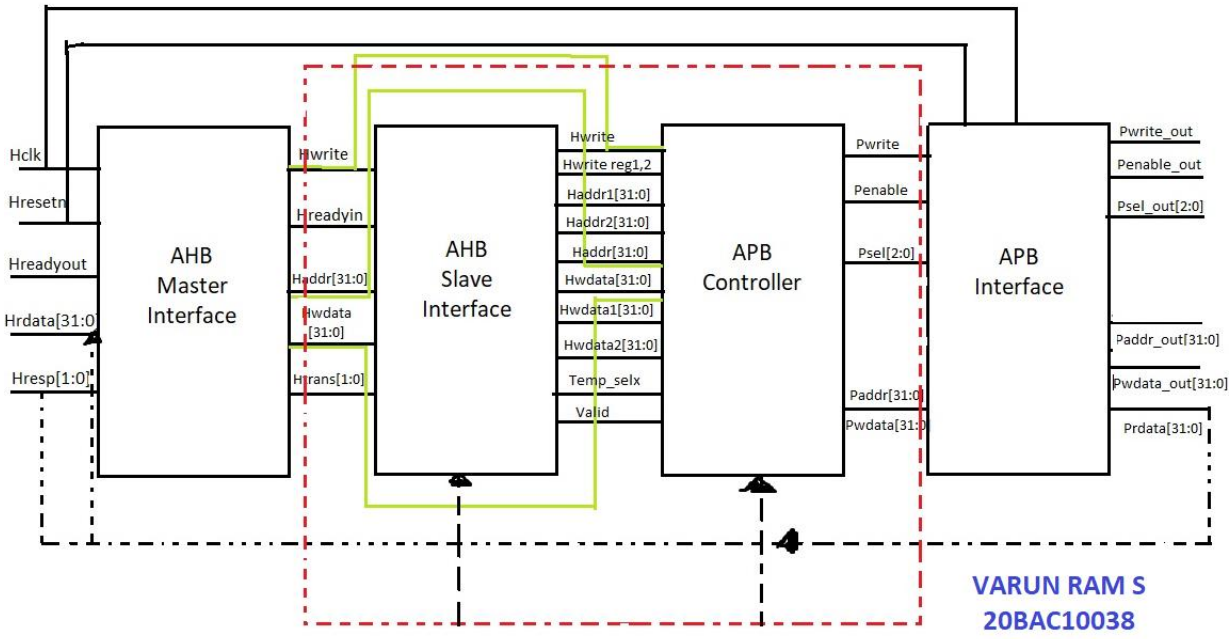
1) HDL Used: Verilog
2) Simulator Tool Used: ModelSIM
3) Synthesis Tool Used: Quartus Prime

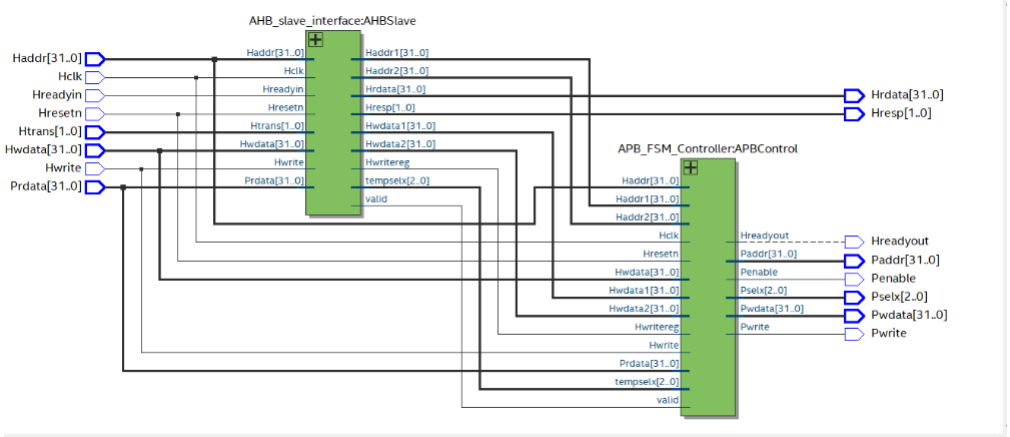
Overview Diagram for reference:





Top Module Block Diagram and functionalities:





The primary functionalities of the Top Module in general include:

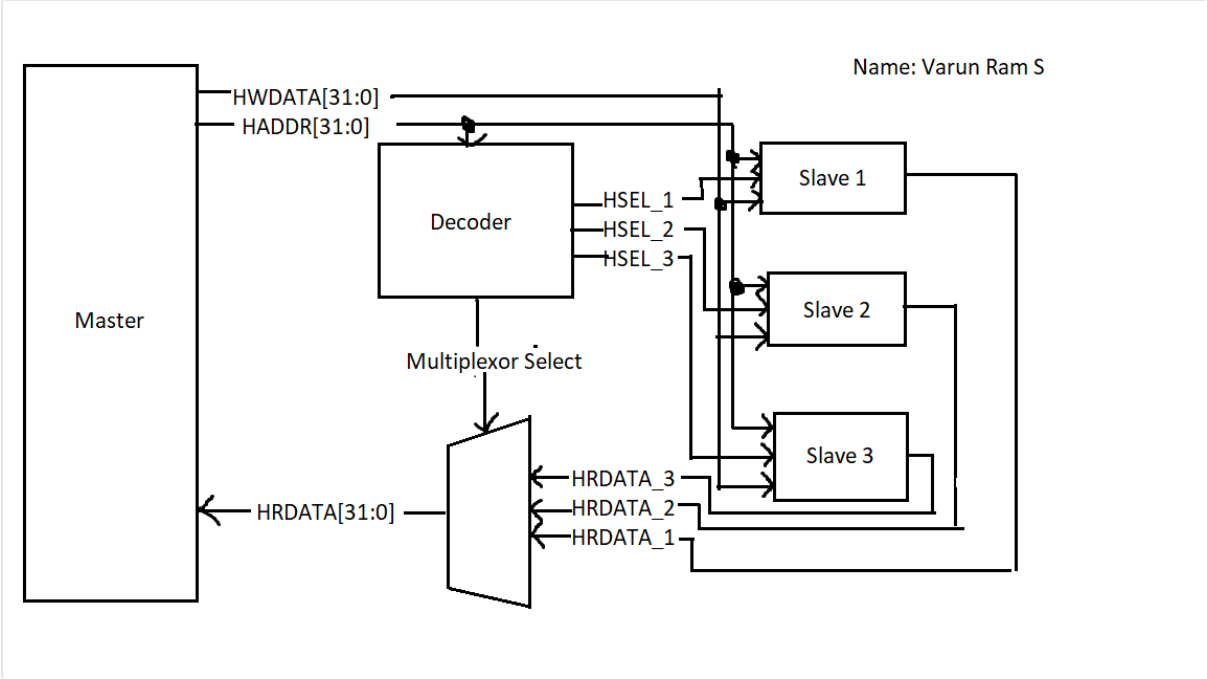
1. Bus Protocol Conversion: To ensure compatibility between the two bus standards, the Top Module converts the AHB bus protocol into the APB bus protocol. It deals with signal conversion, data format conversion, and bus timing conversion.
2. Address Decoding: The Top Module identifies if the matching memory or peripheral is on the APB side by decoding the incoming AHB addresses. The AHB addresses are mapped to the proper APB addresses.
3. Bus Arbitration: The Top Module controls bus arbitration and establishes the precedence of each transaction when more than one AHB master tries to access the AHB2APB Bridge. By coordinating several requests, it guarantees equitable access to the APB bus.
4. Data Transfer Control: Data movement between the AHB and APB buses is managed by the Top Module. Initiating read or write operations, controlling data flow, and managing handshaking signals between the two buses are all things it does.
5. Error Handling: The Top Module could provide systems for reporting and error detection. It keeps track of all bus transactions for issues like bus congestion, protocol infractions, or timeouts and takes the necessary steps to address or report them.
6. Clock Domain Crossing: The AHB and APB buses may run at several clock frequencies or be a part of various clock domains in some systems. Clock domain crossover is managed by the Top Module, maintaining synchronisation and data integrity during the transfer between the two buses.

Sub-Blocks Block and State Diagram with functionality:

1. Bridge Top

The AHB slave interface and the APB controller are the two main parts of the bridge's top module. These modules are created within the main bridge top, and the signal flow that is provided in each sub-module's port list is used to create their interconnection.

2. AHB Slave Interface



The AHB (Advanced High-performance Bus) Slave Interface is a component in the AHB bus protocol that allows peripheral devices or memories to connect to the AHB bus as slaves. It provides the necessary signals and protocol for communication with AHB masters, enabling data transfers and control operations.

Some of the important Signals to be described under this interface include:

I. Address and Control Signals:
   a. HADDR: The address bus signal that carries the address of the accessed location or peripheral.
   b. HSEL: The slave select signal that indicates the selection of a specific AHB slave. Multiple slaves can be connected to the AHB

bus, and the HSEL signal selects the desired slave for communication.

c. HWRITE: The write signal that indicates whether a write operation is being performed (HWRITE=1) or a read operation (HWRITE=0).
d. HSIZE: The size signal that specifies the size of the data transfer (e.g., byte, half-word, word).
e. HBURST: The burst signal that specifies the type of burst transfer (e.g., single, incrementing, wrapping).
f. HPROT: The protection signal that defines the privilege level or access permissions of the transaction.

II. Data Signals:
    a. HWDATA: the write data bus signal, which carries the data that the AHB master will write.
    b. HRDATA: the data read from the AHB slave is carried on the read data bus signal.

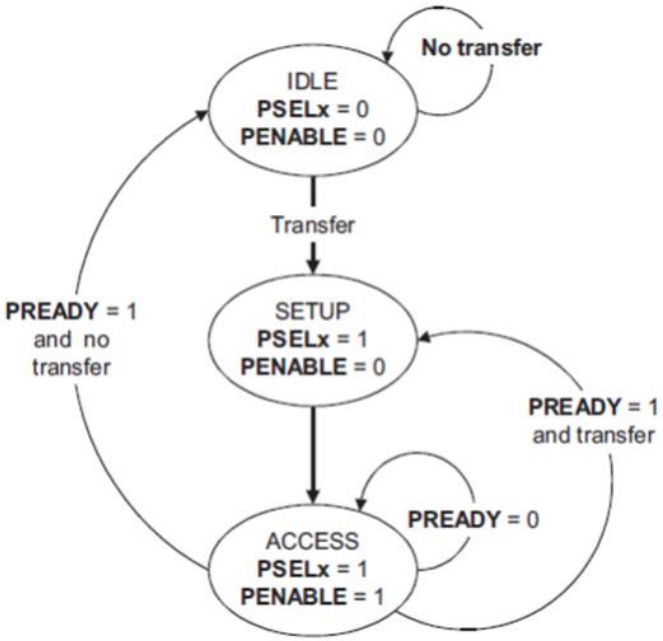
III. Handshaking Signals:
    a. HREADY: Whether the AHB slave is prepared to accept the following transaction is indicated by the ready signal. When the slave asserts HREADY, the AHB master can go on to the next transfer since the slave is ready.
    b. HRESP: the response signal that conveys the prior transaction's status. It may accept the following values: OKAY, ERROR, RETRY, or SPLIT. During bus operation, HRESP can communicate a successful completion (OKAY), an error condition (ERROR), or a number of other situations.

IV. Other Signals:
    a. HTRANS: The transfer type signal, which can be either idle, non-sequential, sequential, or non-sequential last transfer, shows the type of transfer being carried out.

3. APB Controller

An essential part of the AHB2APB bridge architecture is the APB Controller. The connection between the AHB (Advanced High-performance Bus) and APB (Advanced Peripheral Bus) buses is greatly

facilitated by this device. The APB Controller manages the data transmission and control activities as an interface between the AHB masters and the APB slaves.



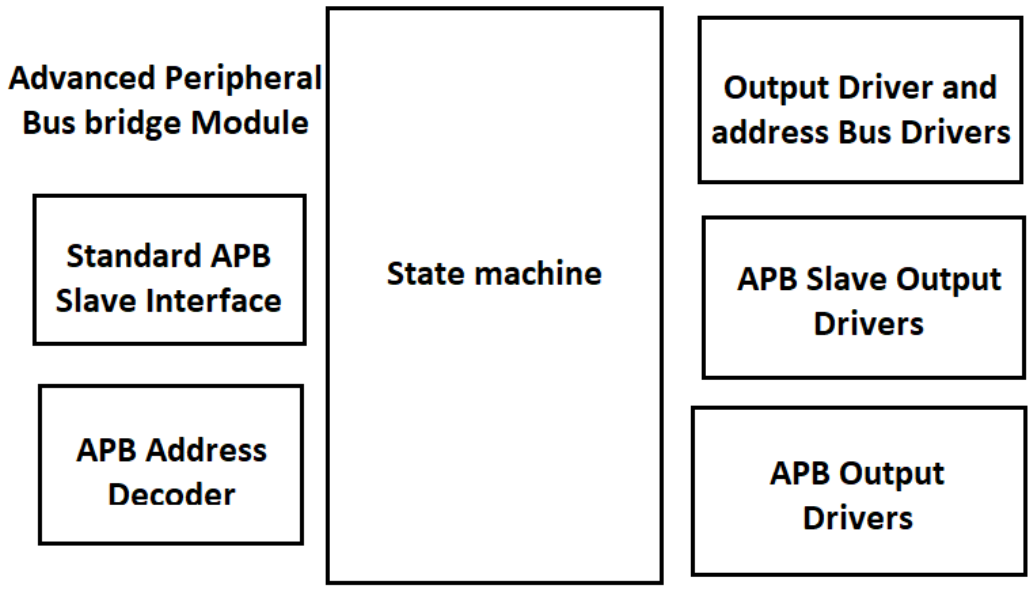
The APB (Advanced Peripheral Bus) interface consists of several signals that facilitate communication between the APB master(s) and the APB slave(s). These signals govern the control, data transfer, and synchronization aspects of the APB bus.

- PCLK (Peripheral Clock): The clock signal that powers the APB bus is called PCLK. It guarantees synchronised communication between the APB master and slave devices as well as serving as the time reference for all APB transactions.
- PRESETn (Peripheral Reset): PRESETn is an active low signal that resets the APB slave devices. When asserted, it initializes the internal registers and states of the APB slaves, preparing them for subsequent operations.
- PADDR (Peripheral Address): PADDR carries the address information for APB transactions. It specifies the memory location or peripheral register being accessed by the APB master.
- PSELn (Peripheral Select): PSELn is an active low signal used for device selection. It indicates which specific APB slave device is being accessed

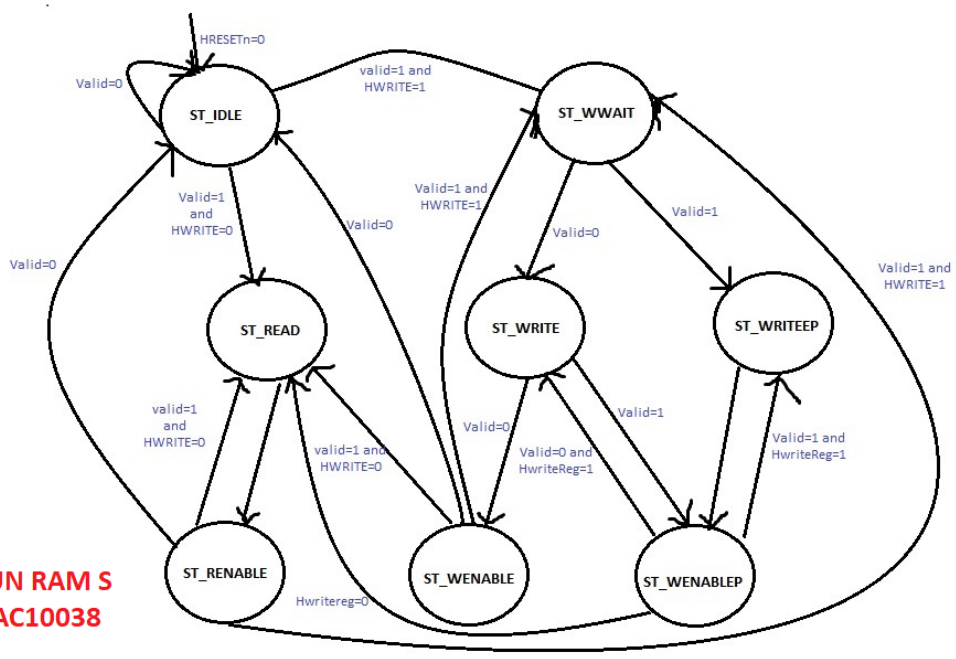
by the APB master. Multiple PSELn signals can be used to select different slaves connected to the APB bus.

- PENABLE (Peripheral Enable): PENABLE is an active high signal that indicates the start and completion of an APB transaction. When asserted, it initiates the data transfer and remains high until the transfer is complete.
- PWRITE (Peripheral Write): PWRITE is a control signal that determines the type of APB transaction being performed. When PWRITE is high, it indicates a write operation, and the APB master is writing data to the APB slave. When PWRITE is low, it indicates a read operation, and the APB master is reading data from the APB slave.
- PWDATA (Peripheral Write Data): PWDATA carries the data to be written to the APB slave during write transactions. The APB master places the write data on this bus to be stored in the specified memory location or peripheral register.
- PRDATA (Peripheral Read Data): PRDATA carries the data read from the APB slave during read transactions. The APB slave places the requested data on this bus for the APB master to read and use.
- PREADY (Peripheral Ready): PREADY is an active high signal indicating the readiness of the APB slave to accept the next transaction. When asserted, it signifies that the APB slave is ready to transfer data or receive commands from the APB master.

VARUN RAM S
20BAC10038



Here is the State Machine Diagram for AHB to APB Interface:



VARUN RAM S
20BAC10038

The Description of Different States are as follows:

- **ST_IDLE:** The PSEL and PENABLE lines are driven LOW, and the APB buses and PWRITE are driven with their most recent values.

- **ST_READ:** In this condition, PADDR receives the address after it has been decoded, the pertinent PSEL line is driven HIGH, and PWRITE is driven LOW.

- **ST_WWAIT:** Due to the pipelined structure of AHB transfers, this state is required to provide the AHB side of the write transfer enough time to finish. As a result, the write data is easily accessible on HWDATA. Consequently, the APB write transfer is started in the following clock cycle. Assuring adequate synchronisation and time alignment between the AHB and APB buses throughout the write transfer procedure is the goal of this state.

- **ST_WRITE:** The address is decoded in this condition before being actively transferred to PADDR. Additionally, the relevant peripheral is chosen as evidenced by the matching PSEL line being set to a HIGH condition. Additionally, a write operation is indicated by the PWRITE signal being set to HIGH. It is noteworthy that no wait state is introduced throughout this operation since just one write transfer is necessary to complete the task without affecting the AHB. With correct address decoding, signal transmission, and synchronisation between the AHB and APB buses, this condition guarantees seamless execution of the write transfer.

- **ST_WRITEP**: In this condition, the necessary PSEL line is driven HIGH, PWRITE is driven HIGH, and the address is decoded and driven onto PADDR. Since there can never be more than one pending transfer between the active APB transfer and the active AHB transfer, a wait state is added.

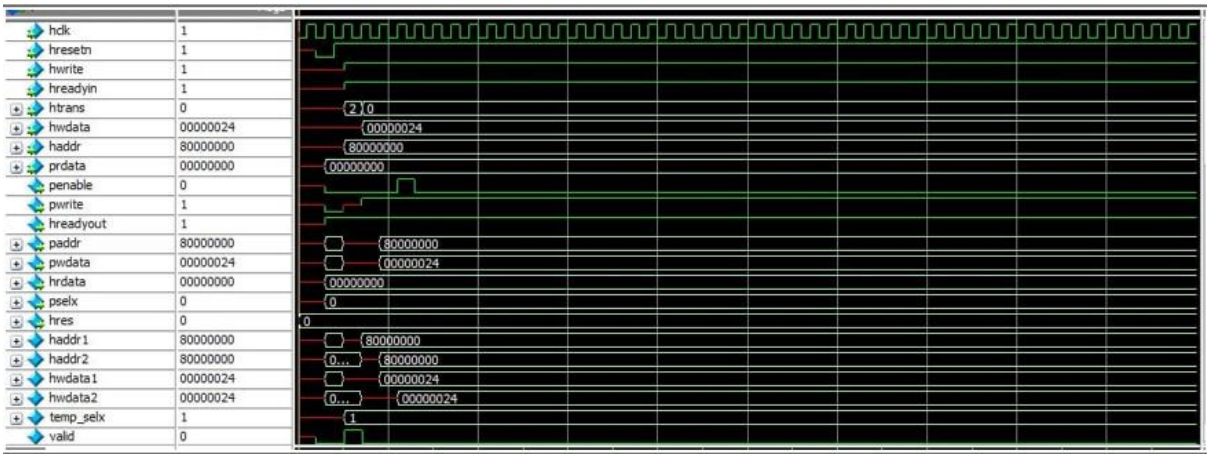
- **ST_RENABLE**: The PENABLE output is pushed HIGH in this condition, permitting the present APB transfer. The other APB outputs are unchanged from the prior cycle in every way. Always starting from ST_READ, the ST_RENABLE state is entered.

- **ST_WENABLE**: The PENABLE output is pushed HIGH in this condition, permitting the present APB transfer. The other APB outputs are unchanged from the prior cycle in every way. Always, ST_WRITE is used to enter the ST_WENABLE state.

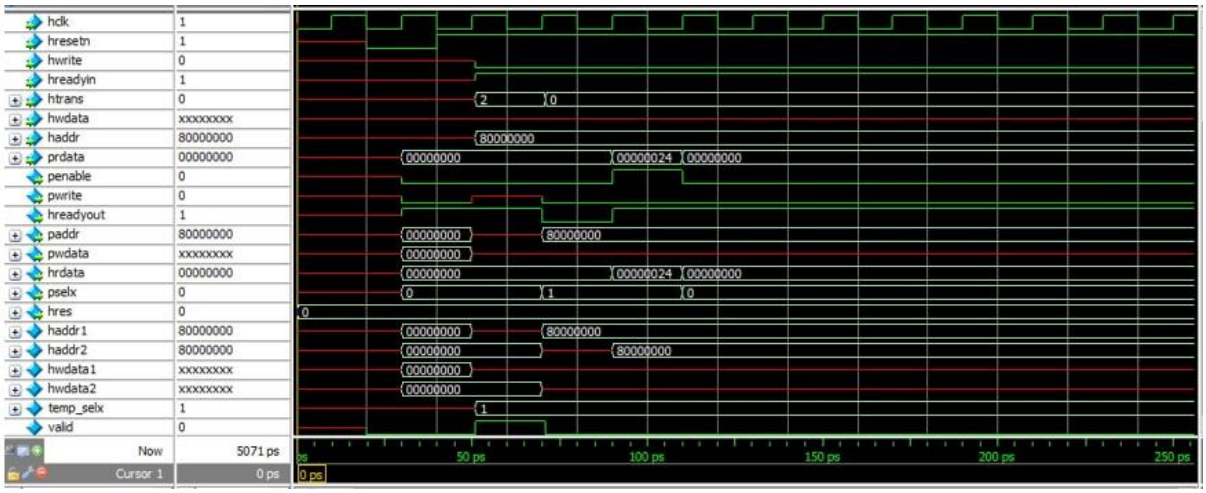
- **ST_WENABLEP**: If the pending transfer is a read, a wait state is added so that the write transfer can finish on the APB before the read is initiated. This is necessary when a read follows a write.

TOP MODULE WAVEFORMS OBTAINED:

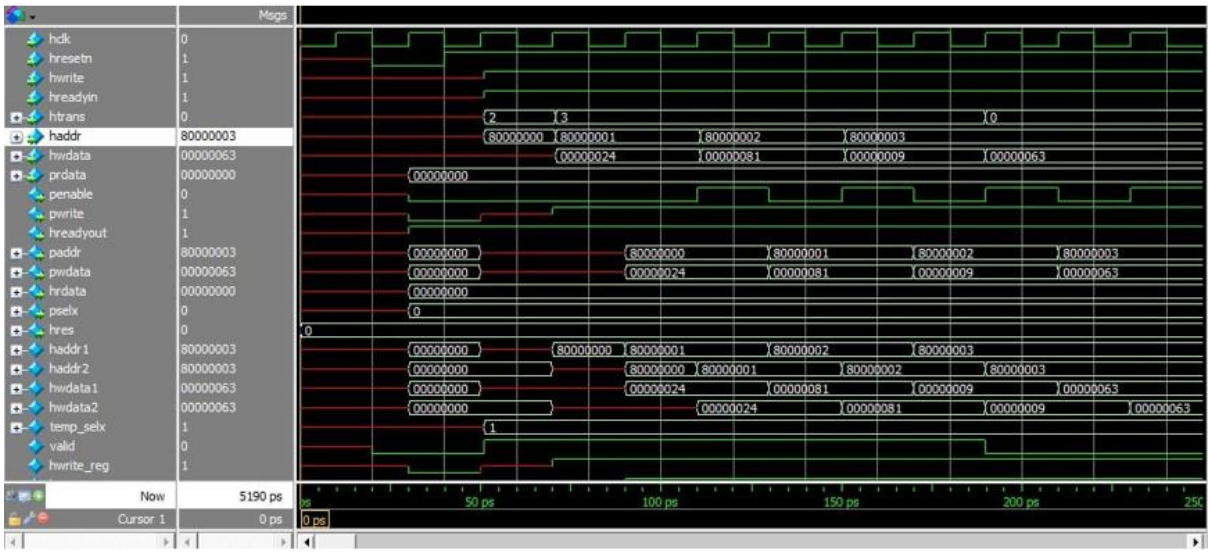
- Waveform for Single Write:



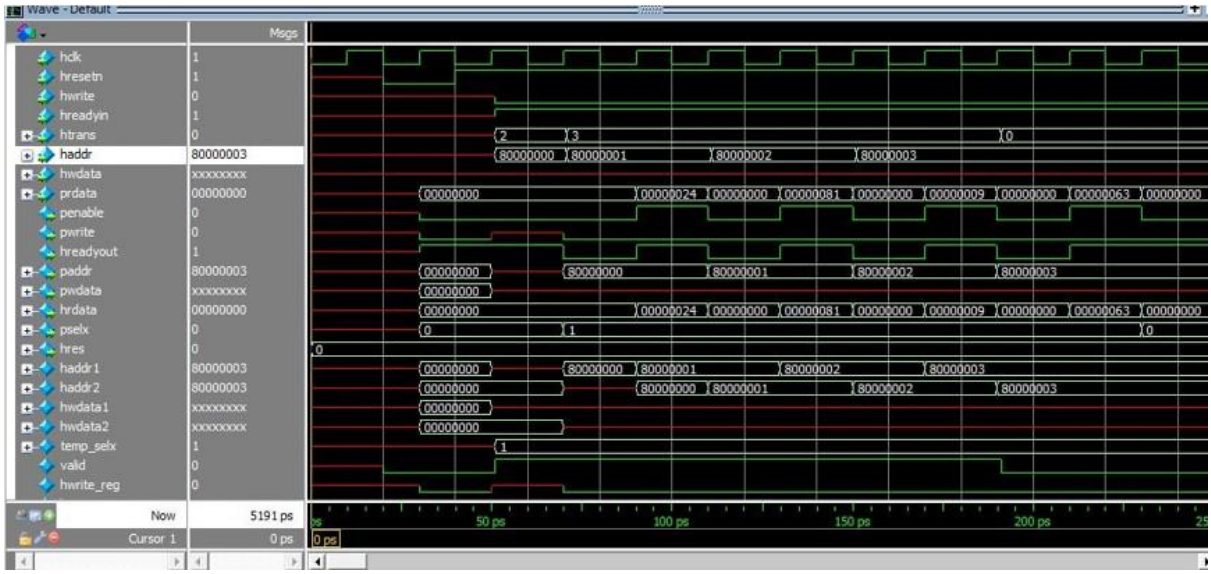
- Waveform for Single read



- Waveform for Burst Write



- Waveform for Burst Read:



----------END OF REPORT----------