International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Conference Proceedings: Volume: 05 ACME – 2026 DOI: 10.55041/ISJEM.ACME153
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Synthesis of AMBA APB BUS Protocol

Mr .N .Dillip Kumar
Department of ECE
Annamacharya Institue of Technology and Sciences
Tirupati, India
Dilipkumar.aits@gmail.com

D . Anitha
Department of ECE
Annamacharya Institue of Technology and Sciences
Tirupati, India
anithadanduboina4@gmail.com

P.Daata Chakra Ganesh
Department of ECE
Annamacharya Institue of Technology and Sciences
Tirupati, India
dattuposina286@gmail.com

V.Bhavana
Department of ECE
Annamacharya Institue of Technology and Sciences
Tirupati, India
vemulabhavana497@gmail.com

Afrooz shaik
Department of ECE
Annamacharya Institue of Technology and Sciences
Tirupati, India
skafrooz1426@gmail.com

***Abstract*——**The ARM Holdings AMBA (Advanced Microcontroller Bus Architecture) standard is widely used in System-on-Chip (SoC) design to enable efficient communication between IP cores. It provides a modular and scalable framework that simplifies system integration, reduces redesign complexity, and shortens development time. Different versions of AMBA support various communication interfaces such as APB, ASB, AHB, AXI, ACE, and CHI, each designed to meet specific performance and application requirements. The architecture is developed using detailed Register Transfer Level (RTL) schematics and implemented with Verilog HDL following VLSI design principles. In this work, a new codebase was created from scratch to synthesize the proposed architecture. The optimized design demonstrates improved overall performance, reduced total power consumption, minimized switching power, and efficient cell area utilization under high-effort conditions, highlighting effective architectural and design optimization.

Index Terms—AMBA, APB, RTL, Verilog, SoC, Master, Slave, Signals, Bus Protocol, Power Optimization.

I. INTRODUCTION

The VLSI design flow is a systematic process used to develop integrated circuits, beginning with defining the overall architecture, system requirements, and detailed specifications of individual sub-blocks. This initial stage ensures that the design goals and functionality are clearly established.

Once the specifications are defined, the RTL (Register Transfer Level) design phase begins. In this stage, designers implement the functionality of each sub-block using hardware description languages such as Verilog or SystemVerilog, enabling simulation and verification of the design logic.

To ensure efficient communication between different components of the chip, standardized bus protocols are used. AMBA-based buses like APB, AHB, and AXI provide structured data transfer mechanisms, improving integration and performance within System-on-Chip (SoC) designs.

Physical design stages such as floor planning and power planning follow RTL design. Floor planning determines the placement of different blocks on the chip, while power planning ensures stable and efficient distribution of power across all components.

After physical design, the final chip layout is generated in the form of a GDSII file. This layout undergoes thorough verification and testing to confirm that it meets all design rules, performance targets, and functional requirements before fabrication.

AMBA, introduced by Arm Holdings in 1996, is an open-standard protocol that simplifies on-chip communication. Over time, it has evolved with versions like AMBA 2 (featuring AHB) and later AMBA 4 and 5, which introduced advanced features such as ACE and CHI, making it a fundamental standard in modern SoC design.

II. LITERATURE SURVEY

The ARM Holdings AMBA protocol is a standard on

chip bus architecture widely used in semiconductor systems.APB, as part of AMBA, supports low-bandwidth peripheral communication. Various studies implemented APB designs, bridge architectures, verification methods, and physical implementations using industry tools and FPGA platforms.

Ravikumar et al. [1] presented the design and verification of
the amba apb protocol with emphasis on RTL modelling and simulation basedvalidation. Their work focused on ensuring protocol compliance, proper signal sequencing, and reliable data transfer between master and slave modules.

Yadav et al. [2] implemented the AMBA APB protocol for SoC applications. The study demonstrated structured a development, hardware architecture synthesis, and integration capability with peripheral devices. Performance and area utilization were analyzed to validate design efficiency.

Shanthi et al. [3] proposed the design and FPGA implementation of an AHB to APB bridge. The bridge architecture enabled communication between high performance and low power bus domains. The implementation addressed protocol conversion, synchronization, and hardware optimization.

Dwivedi et al. [4] focused on assertion based and functional coverage driven verification of the APB protocol using System Verilog. Their methodology improved verification
quality by identifying corner cases and enhancing functional coverage metrics.

Jain and Rao [5] developed and verified the AMBA APB protocol using simulation driven approaches. The work validated timing characteristics and ensured stable communication between master and slave components.

Roopa et al. [6] implemented a UART controller as an APB slave module. The study demonstrated practical peripheral in
tegration within an APB based embedded system, emphasizing simplified communication and low power operation.

Ma et al. [7] designed an APB bridge based on AMBA specifications to support protocol translation and efficient data transfer. The architecture reduced latency and ensured compatibility between different bus structures.

Manu and Prabhavathi [8] implemented an ASB to APB bridge to facilitate communication between Advanced System Bus and Advanced Peripheral Bus. The work highlighted reli able signal conversion and interoperability across bus domains.

Rawat et al. [9] presented RTL implementation of AMBA ASB APB protocol at system level. The research emphasized
modular design, protocol validation, and subsystem integration within SoC architecture.

Gupta et al. [10] performed physical design implementation of a 32 bit AMBA ASB APB module. The study included synthesis, placement, routing, and timing analysis to achieve improved performance and optimized area utilization.

III. DESIGN DESCRIPTION

A. Proposed Architecture

Outlined herein is the hardware structure proposal, boasting two slave units and a master unit synchronized to the rising clock edge of the APB BUS Protocol, operating in parallel. The design meticulously manages signal priorities, namely PSEL(P-Select), PRESET, PWRITE, PENABLE, and PREADY. Data transmission is commenced with the low-to-high shift of PENABLE, whereas the high-to-low shift of PREADY marks its conclusion. The APB Bridge acts as the interface between the master system and the APB slave devices. It receives input signals such as pclk presetn paddr pwdata pwrite and pselin. Based on the transfer request, the bridge generates APB control signals including penable, pwrite paddr pwdata and psel. These signals control the communication with the selected APB slave. The bridge also monitors response signals such as pready prdata and pslverr from the slaves. It ensures proper synchronization of the read and write operations and produces status outputs like apb bridge ready apb data valid, error flag, and paddr err to indicate the state of the transaction. The architecture includes multiple slave modules such as APB Slave 0 and APB Slave 1. A multiplexer selects the appropriate slave based on the address decoding and psel signal. Each slave receives control signals such as paddr, pwdata, pwrite, penable, clk, and resetn to perform the required read or write operation

B. Block Diagram

From the below given Fig. 1 , each of the two APB slave circuits can hold up to 256 bits and are supported in the proposed setup. The central point of connection between them is the APB slave circuit, which translates APB protocol into simple read and write actions. Verification

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Conference Proceedings: Volume: 05 ACME – 2026 DOI: 10.55041/ISJEM.ACME153
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

setup can be streamlined with the use of an attached APB Wrapper, potentially improving efficiency.

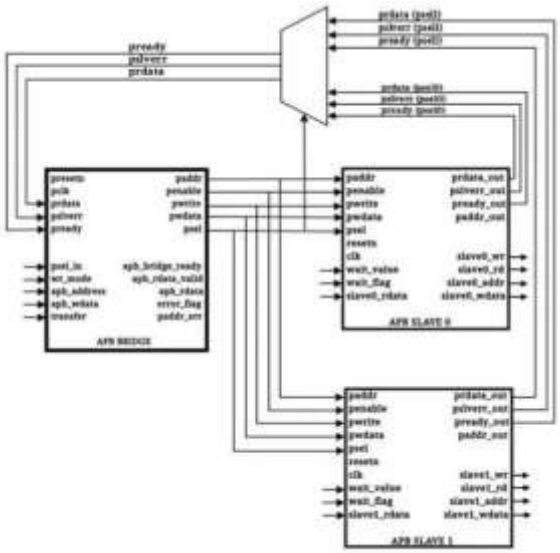

Fig. 1: Block Diagram of APB Bus Protocol Module

C. Flow Chart

The APB BUS protocol flowchart for the project is illustrated in Fig. 2. The process initiates with a block reset, transitioning to the IDLE state. Subsequently, PSelx is employed to select either Psel1 or Psel2, designating one of the two slaves. PSelx and PEnable are critical in determining the system's state transitions. There are three states: IDLE, SETUP, and ACCESS. The default state is IDLE, where

PSelx=0. The system moves to the SETUP state when PSelx=1 and PEnable=0, and to the ACCESS state when both PSelx=1 and PEnable=1.

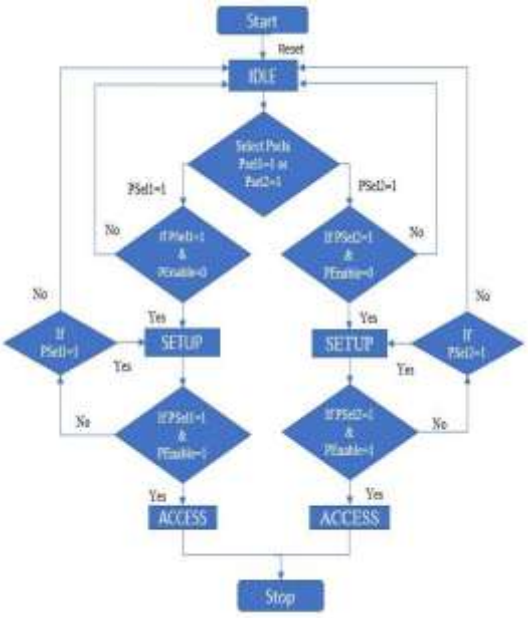

Fig. 2: Flow Chart of APB Bus Protocol Module

D. Finite State Machine

The FSM flow chart as shown in Fig. 3, explains the SETUP and ACCESS states, which are considered for READ and WRITE operations. The default state is IDLE, and when a write transfer request is asserted, the FSM moves into WRITE_SETUP and WRITE_ACCESS. The READ_SETUP state is asserted upon read transfer request, and the READ_ACCESS state is asserted upon PEnable assertion.

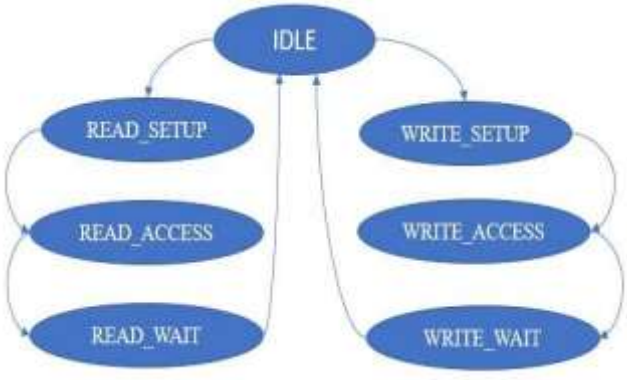

Fig. 3: FSM of APB Bus Protocol

IV. IMPLEMENTATION &RESULTS

This is seen in Fig. 4a. The waveform illustrates the functional simulation of the AMBA APB protocol, showing proper interaction between master and slave components over time. The clock signal (PCLK) drives the entire operation, while PRESETn ensures system initialization. Address (PADDR), write control (PWRITE), and write data (PWDATA) signals demonstrate data transfer during write operations. The read data (PRDATA) reflects successful data retrieval, confirming correct read functionality. The PREADY signal indicates the completion of each transfer, coordinating timing between master and slave. The waveform transitions clearly depict the APB states—IDLE, SETUP, and ACCESS—ensuring protocol compliance. No error signal (PSLVERR) is observed, indicating reliable communication. The pulse_out and interrupt signals show additional system activity and response behavior. Overall, the simulation verifies correct read/write operations, proper synchronization, and stable performance of the APB-based design, confirming that the implemented architecture functions accurately and efficiently under test conditions.

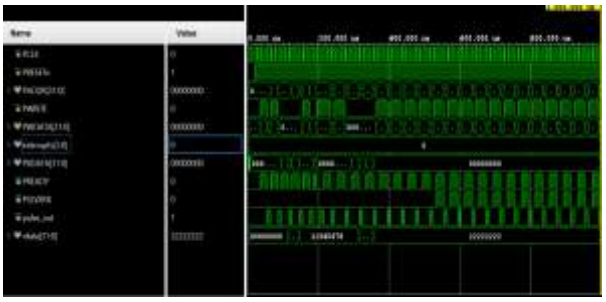

Fig. 4a: Waveforms

This is seen in Fig. 4b.The schematic diagram represents the implementation of an AMBA APB-based system with one master and multiple slave modules. The master controls data transfer by generating address, control, and write signals, while the slaves respond with read data, ready, and error signals.

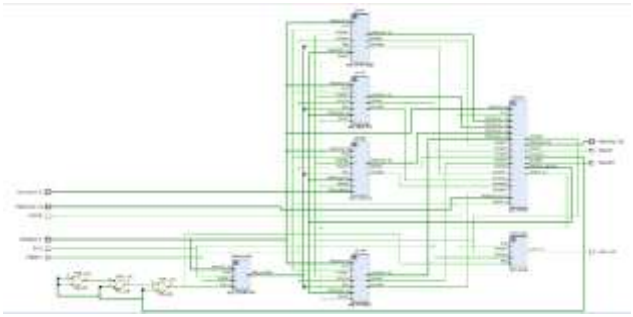

Fig. 4b: Schematic diagrams

A decoder (fast decode block) is used to select the appropriate slave based on the address. Multiple APB slave modules such as register, FIFO, interrupt, and control blocks are integrated to handle different functionalities. The design also includes a pulse generator module for additional signal generation. All components are synchronized using the clock (PCLK) and reset (PRESETn) signals. Overall, the architecture ensures efficient communication, proper data transfer, and modular integration of multiple peripherals using the APB protocol.

TABLE I: COMPARISON

Parameter	Existing Model	Proposed Model
Cell Count	2349	2027
Total Cell Area	8288.370	7672.086
Total Power	100.26	99.80

The Table.1comparison clearly demonstrates that the proposed model provides significant improvements over the existing model in terms of design efficiency and optimization. The cell count is reduced from 2349 to 2027, indicating a more compact and optimized architecture with fewer logic elements. Similarly, the total cell area decreases from 8288.370 to 7672.086, showing better utilization of hardware resources and more efficient layout design. In addition, the total power consumption is slightly reduced from 100.26 to 99.80, which is beneficial for low-power and energy-efficient applications. These improvements suggest that the proposed model not only minimizes hardware complexity but also enhances overall performance. Therefore, it is more suitable for modern System-on-Chip (SoC) designs where area, power, and efficiency are critical factors.

V. CONCLUSION

This project focuses on the AMBA architecture, particularly the APB protocol, to achieve an efficient and area-optimized design using Xilinx Vivado. It uses master and slave components with a multiplexer to enable parallel communication, controlled by signals like PRESET, PSEL, PENABLE, PREADY, and PWRITE. Testbench verification and waveform analysis ensure correct read/write operations and reliable APB functionality.

REFERENCES

[1] E Ravikumar, K B Ganesha, C Y Chethana, and T M Parikshith Roy, “Design and Verification of Advanced Microcontroller Bus
Architecture (AMBA) Advanced Peripheral Bus (APB) Protocol”,
2024 International Conference on Smart Systems for applications in Electrical Sciences
(ICSSES), May 2024, DOI: 10.1109/ICSSES62373.2024.10561369

[2] Sirimalla Karun Kalyan Yadav, P. Satyanarayana, Bavineni Vaibhav
Krishna, Kolli Ranjith Kesav Sai, Arun M, and V. Gokula Krishnan,
“Design and Implementation of AMBA-APB Protocol for System-onChip (SoC) Designs”, 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics
(ICDCECE), April 2024, DOI: 10.1109/ICDCECE60827.2024.10549635.

[3] Shanthi, G., Sravani, K.G., Vali, S.S. et al. Design and FPGA implementation of AHB-to-APB bridge. Microsyst Technol (2024). https://doi.org/10.1007/s00542-024-05703-1

[4] Prashant Dwivedi, Neha Mishra, and Amit Singh-Rajput, “Assertion
& Functional Coverage Driven Verification of AMBA Advance
Peripheral Bus Protocol Using System Verilog”, 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), February 2021, DOI: 10.1109/ICAECT49130.2021.9392518

[5] Padmaprabha Jain, and Satheesh Rao, “Design and Verification of
Advanced Microcontroller Bus Architecture-Advanced Peripheral Bus (AMBA-APB) Protocol”, 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile
Networks (ICICV), February 2021, DOI: 10.1109/ICICV50876.2021.9388549

[6] Sirimalla Karun Kalyan Yadav, P. Satyanarayana, Bavineni Vaibhav
Krishna, Kolli Ranjith Kesav Sai, Arun M, and V. Gokula Krishnan,
“Design and Implementation of AMBA-APB Protocol for System-onChip (SoC) Designs”, 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics
(ICDCECE), April 2024, DOI: 10.1109/ICDCECE60827.2024.10549635.

[7] Shanthi, G., Sravani, K.G., Vali, S.S. et al. Design and FPGA implementation of AHB-to-APB bridge. Microsyst Technol (2024). https://doi.org/10.1007/s00542-024-05703-1

[8] Prashant Dwivedi, Neha Mishra, and Amit Singh-Rajput, “Assertion
& Functional Coverage Driven Verification of AMBA Advance
Peripheral Bus Protocol Using System Verilog”, 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), February 2021, DOI: 10.1109/ICAECT49130.2021.9392518

[9] Padmaprabha Jain, and Satheesh Rao, “Design and Verification of
Advanced Microcontroller Bus Architecture-Advanced Peripheral Bus (AMBA-APB) Protocol”, 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile
Networks (ICICV), February 2021, DOI: 10.1109/ICICV50876.2021.9388549

[10] K. Akhila, N. Karuna and Y. J. M. Shirur, "Design and Implementation of Power Efficient Logic BIST With High Fault Coverage Using Verilog," 2018 International Conference on Networking, Embedded and Wireless Systems (ICNEWS), Bangalore, India, 2018, pp. 1-6, doi: 10.1109/ICNEWS.2018.8903923