International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 05 | May - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Design and Development of UART Protocol Using Verilog with UVM Testbench

Krushna K. Chavan[1], Shreya S. Kirdak[1] , Aboli S. Bhagwat[1] , Snehal R. Sabale[2], Ajit Gangad[2]

[1]B.E-Electronics & Telecommunication Engineering/Student, Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering and Technology, Baramati, India

[2]M.E-Electronics & Telecommunication Engineering/Assistant Professor Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering and Technology, Baramati, India

Email address: krushnachavan2310@gmail.com (Krushna K. Chavan) snehal.sabale@vpkbiet.org (Snehal R. Sabale)

---***---

Abstract –

This paper's primary goal is to use Verilog (V) to design and validate a full duplex UART module. The Universal Asynchronous Receiver/Transmitter, or UART, is a two-way transmission channel that not only significantly increases the efficiency of information transfer between computers and external devices but also guarantees information accuracy and consistency through the use of baud rate settings, the elimination of metastable state, and other techniques. In this work, the definition and operation of UART are well understood, and the UART is constructed using the Verilog HDL language, EDA simulation, picture, and data. The findings of the experiment demonstrate that this module's sending and receiving module functions well and satisfies the specifications of full-duplex serial communication equipment. Without a doubt, the layout of this paper has made the fundamental working concept of UART more clearly explained, which will aid in its future.

Key Words:- UART, EDA , BRG, IC, DUT, UVM

1. INTRODUCTION

Universal Asynchronous Receiver Transmitter – UART is a universal serial communication protocol that transmits data serially between systems. To operate a computer interface, an integrated circuit (IC) in a microcontroller is used as computer hardware. UARTs are used for both sending and receiving data. Since data transmission is asynchronous, a clock is not necessary. The name Universal Asynchronous Receiver Transmitter comes from the fact that both the data format and the transmission speed are configurable. For communication, the majority of peripherals employ parallel data formats. In the transmitter part, UART transforms data received in parallel format into serial data before sending it to the recipient. Prior to transferring the serial data to the peripheral devices, the receiver needs convert it to parallel format. The IC verification approach enhances the rapid development process.

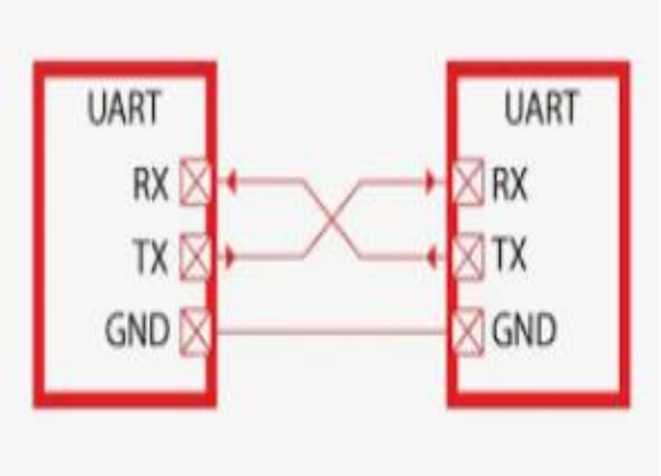


Fig 1. Implementation of UART

2. DESIGN

Designing a UART include designing the transmitter side, receiver side, and Baud Rate generator. In order to provide complete duplex communication across systems, two UART modules are crafted as illustrated in Fig 1.

A. Data Format

It generally consists of one start bit, which is always at logic 0, one data bit (usually between five and eight), and one or two programmable stop bits. There is never a stop bit other than logic 1. Therefore, 10 bits of data can be transmitted using a normal UART. The transmitter and the receiver in the UART systems have distinct local clock signals and do not share a clock signal.

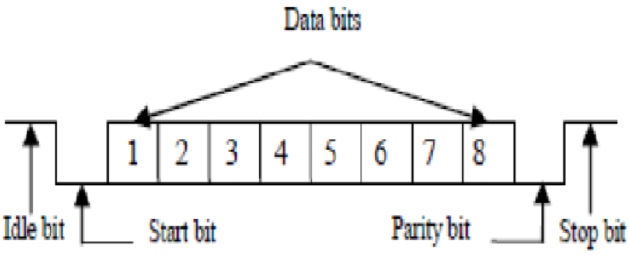


Fig 2.Data Format

B. Wishbone Interface

Is a common computer bus interface that facilitates communication between integrated circuit components .Used to speed the design in order to solve SOC integration issues.

C. BRG

A frequency divider is a BRG. As seen in Fig. 2, UART features a customizable baud generator. To create a baud clock (BCLK), BGR divides an input clock from the processor clock generator by a divisor that is kept in a divisor latch. Data is stored for 16 baud clock cycles, or sixteen times the baud rate.

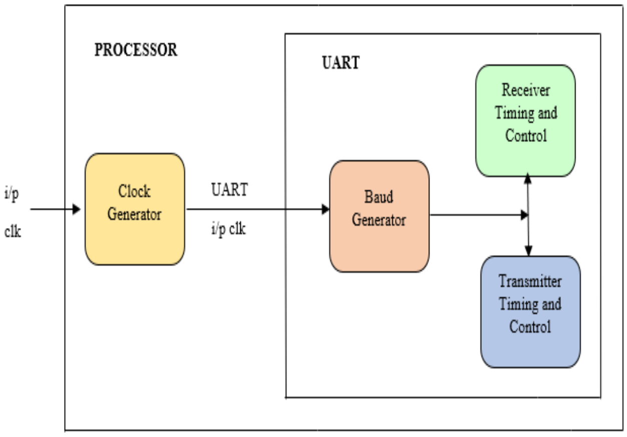


Fig 3: Baud Rate Generator

D. UART Transmitter

A UART transmitter converts parallel data from a microcontroller into serial data for transmission. It synchronizes with the receiver using a shared baud rate and sends data one bit at a time, including start, optional parity, and stop bits. Widely used in microcontrollers and embedded systems, UARTs provide reliable serial communication over short to moderate distances, essential for diverse applications such as sensor networks and industrial automation.

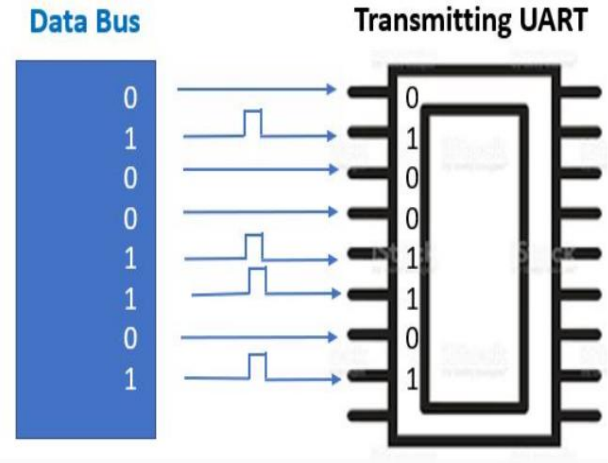


Fig 4. Transmitting UART

E. UART Receiver

An essential part of serial data communication systems, a UART receiver is in the role of taking in and processing asynchronous data streams. It is essential for maintaining the integrity and dependability of incoming data by framing, synchronizing, and error-checking it. To reconstruct the parallel data for additional processing, the receiver usually extracts the start and stop bits, data bits, and optional parity. In many other applications where serial data transmission is necessary, such as computer peripherals and microcontroller-based systems, UART receivers are employed extensively.

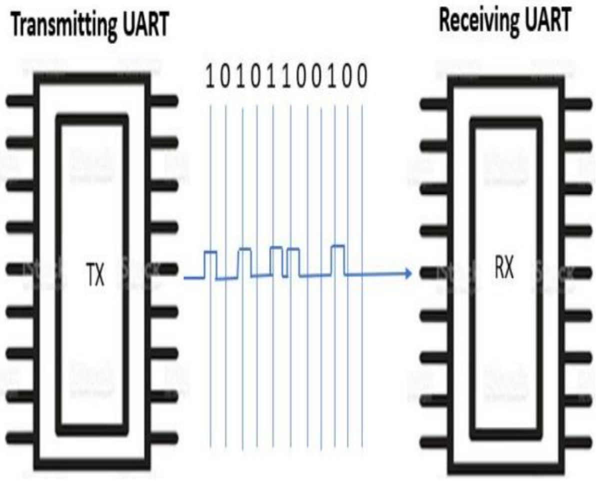


Fig 5.Receiving UART

3. Verification Methodology

The difficulty of verification is rising due to the rapid evolution of IC design. The verification discipline has advanced significantly as a result of the perception that verification is a distinct process from design. The verification sector is shifting toward Verilog , or the extended Verilog version. Verilog is a language for both hardware description and hardware verification. Verilog is a popular language for developing and verifying digital systems. Verification efficiency is increased and complexity is decreased by developing the verification environment using Verilog.

A. Verification Environment

UVM Testbench

The VLSI industry uses the Universal Verification approach (UVM), a system verilog language-based verification approach. The UVM methodology is designed to construct test benches in order to validate the design. A specified process called UVM is used to create modular test benches for design verification. A collection of classes at UVM facilitates the design and implementation of modular stimulus and testbench components.

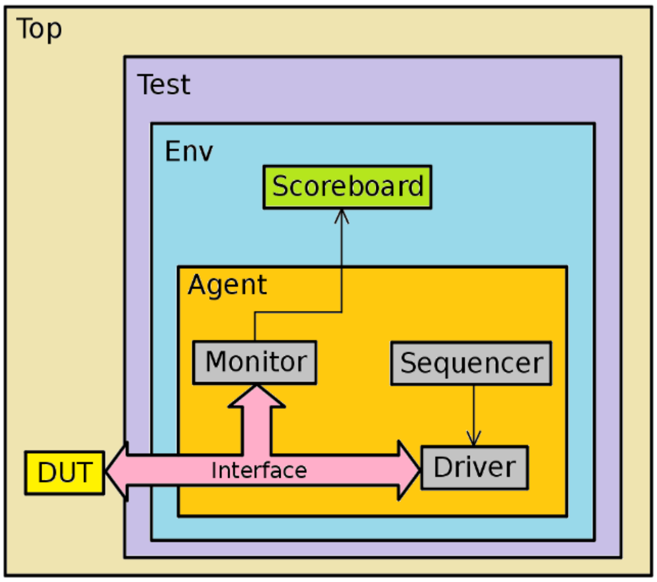


Fig 6. UVM Testbench Architecture

- **Testbench:** Instantiates the both unit under test and test class. And also establishes connection between them.
- **Test:** Configures the testbench, construction of a higher level lower in the hierarchy will serve as the initial testbench parts development measure and instantiates the sequence tobegin the stimulus.
- **Environment:** Verification components like scoreboard and agent are grouped together for reusability.
- **Agent:** Incorporates monitor, driver and sequencer as a signal entity through the TLM interface.
- **Sequencer:** The task of the sequencer is to

guide transactions (also known as sequence items) that are generated in sequence to the driver or vice versa.

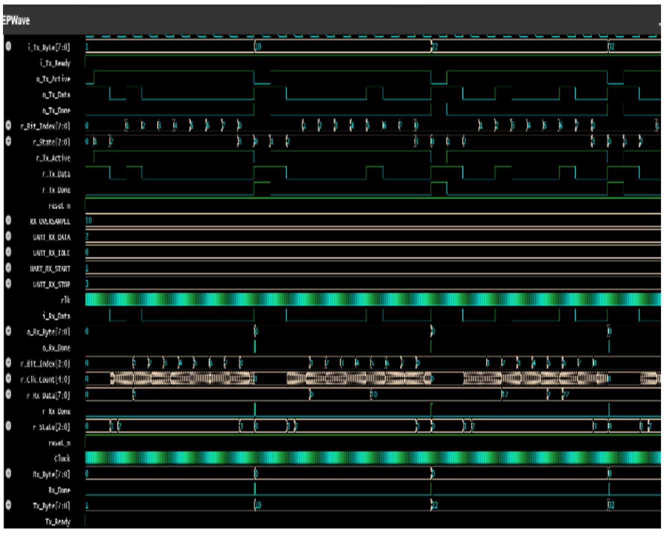
- **Driver:** Retrieves data from sequencer and sends the same to DUT and reference model through interface.
- **Monitor:** Samples the unit being tested and reference model, records the data present in transactions, and then compares it. Scoreboard: Consists of checkers to verify the design functionality.

4. Simulation Results :

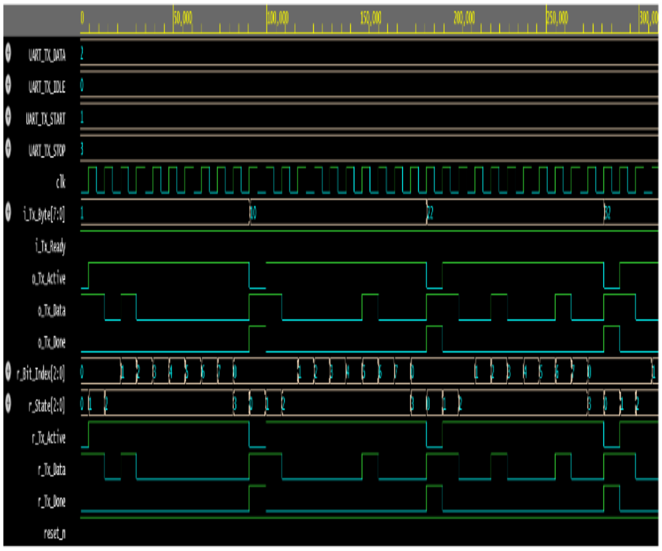
1. Design Results :

UART communication involves two main components: the transmitter and the receiver. The transmitter sends data serially, one bit at a time, to the receiver. The communication is asynchronous, meaning there is no clock signal to synchronize the data transmission. Instead, both the transmitter and receiver agree on a specific baud rate (the number of bits transmitted per second).

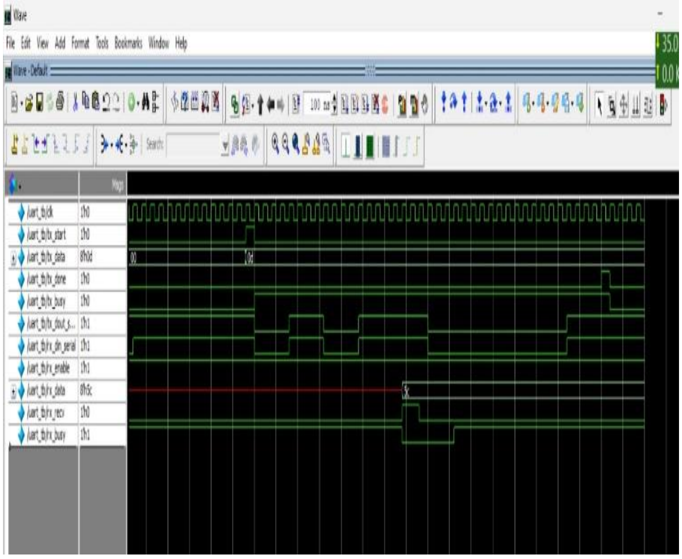
A . UART Transmitter Simulation



B. UART Receiver Simulation

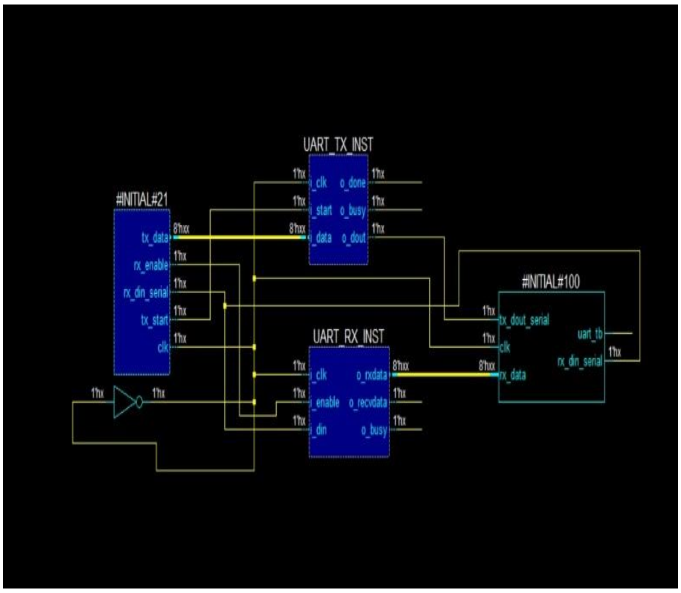


C. UART Protocol Simulation



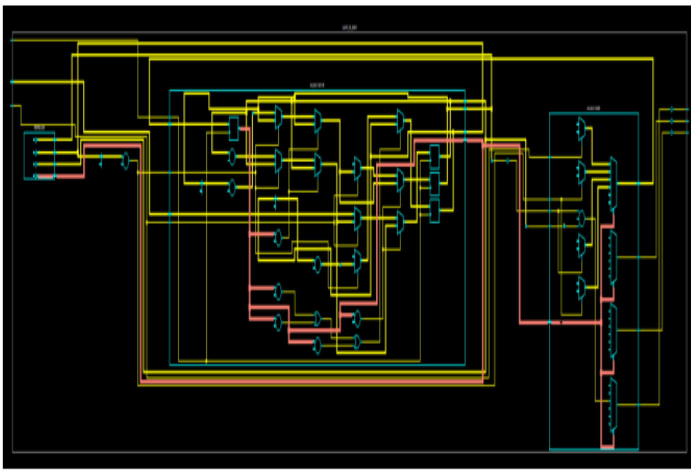
2. UART Schematic Results

The following schematics depict the detailed design of both the UART transmitter and receiver. These diagrams are essential for understanding the structural layout and functional interactions within the UART module. Each component's role and its integration into the overall system are highlighted and discussed to illustrate how the UART achieves its intended functionality.



5. RTL Design Full Wave of UART

Designing a UART requires precise timing and control logic for accurate data framing and synchronization. The RTL design involves modules for transmitting and receiving, incorporating shift registers, data registers, control logic, and baud rate generators to ensure reliable serial communication



Conclusion :

Using a UVM testbench and Verilog, the design and implementation of a UART protocol has been successfully shown in this research article. We have demonstrated how to effectively send and receive data in serial communication by putting this protocol into practice. Serial communication is a vital component of many electronic systems and gadgets. The UVM testbench does extensive testing to guarantee the UART protocol's functionality and dependability, while the Verilog code offers an organized method for creating it. This work advances digital communication technologies and provides a useful resource for researchers and engineers studying digital design and embedded systems.

References:

1. B. Priyanka, M. Gokul., A. Nigitha. and J. T.Poomica., "Design of UART Using Verilog And Verifying Using UVM," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2021, pp. 1270-1273, doi: 10.1109/ICACCS51430.2021.9441789.

2. S. Harutyunyan, T. Kaplanyan, A. Kirakosyan and H. Khachatryan, "Configurable Verification IP for UART," 2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO), Kyiv, Ukraine, 2020, pp. 234-237, doi:10.1109/ELNANO50318.2020.9088878

3. C. Elakkiya, N. S. Murty, C. Babu and G. Jalan, "Functional Coverage - Driven UVM Based JTAG Verification," 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Coimbatore, India, 2017, pp. 1-7, doi: 10.1109/ICCIC.2017.8524556.

4. W. Ni and X. Wang, "Functional coverage-driven UVM based UART IP verification," 2015 IEEE 11th International Conference on ASIC (ASICON), Chengdu, China, 2015, pp. 1-4, doi: 10.1109/ASICON.2015.7517188

5. L. M. Kappaganthu, M. D. Prakash and A. Yadlapati, "I2C protocol and its clock

stretching verification using system verilog and UVM," 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 2017, pp. 478- 480, doi: 10.1109/ICICCT.2017.7975245.

6. N. B. Harshitha, Y. G. Praveen Kumar and M. Z. Kurian, "An Introduction to Universal Verification Methodology for the digital design of Integrated circuits (IC's): A Review," 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, 2021, pp. 1710-1713, doi:10.1109/ICAIS50930.2021.9396034.

7. Girdhar, D., & Shukla, N. K. (2014). Design and verification of amba apb protocol. International Journal of Computer Applications, 95(21).

8. Yammanuru P., & Reddy, M. A. (2021). Design And Verification of Advanced Peripheral Bus Protocol Using Uvm. Journal of Artificial Intelligence, Machine Learning and Neural Network (JAIMLNN) ISSN: 2799-1172, 1(02), 1-9.

9. P. Jain and S. 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), Tirunelveli.

10. Kumar, K. A., & Krishna, M. S. (2015). Design and Functional Verification of A SPI Master Slave Core using UVM. International Journal of Scientific Engineering and Technology Research (IJSETR), 4(51), 11023-11030.

11. Sharath, S. G. (2022). UVM based Verification of Watchdog Timer with APB. Journal of Algebraic Statistics, 13(3), 5343- 5350

12. Kumar, K. A., & Krishna, M. S. (2015). Design and Functional Verification of A SPI Master Slave Core using UVM. International Journal of Scientific Engineering and Technology Research (IJSETR), 4(51), 11023-11030

13. Kamath, A., Mendez, T., Ramya, S., & Nayak, S. G. Design and Implementation of Power-Efficient FSM based UART. Journal of Physics: Conference Series, 2022, 2161(1), 012052**.**

14. Thai, N. P., Ngoc, B. H., Duy, T. D., Truong, P. Q., & Phan, V. C. A novel multichannel UART design with FPGA-based implementation. International Journal of Computer Applications in Technology, 2021, 67(4), 358.

15. Amrit P. Singh, Khushboo Gupta, & Juhee Mala. Determination of UART Receiver Baud Rate Tolerance. International Conference on Computer and Communication Technology, 2015, 265–270, 2818655.