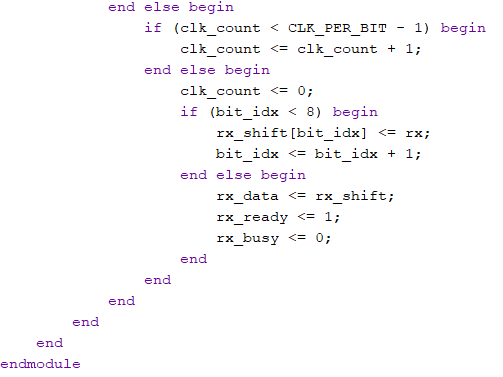
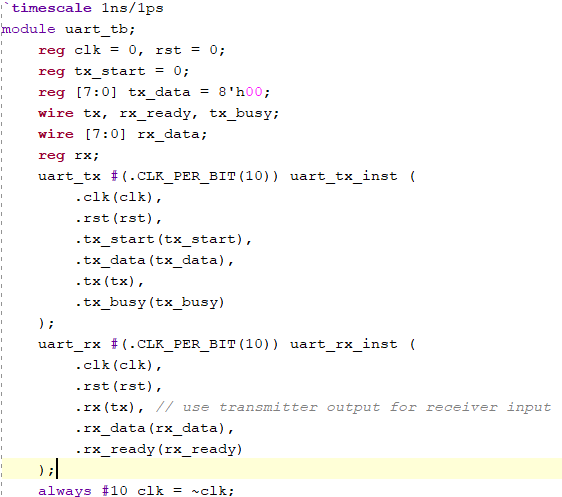
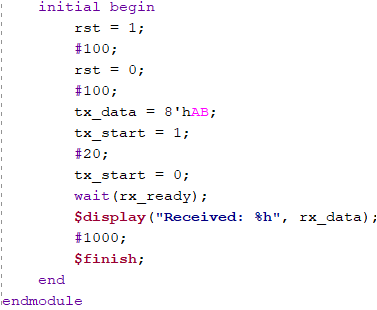
|  |
| --- |
| **VLSI Implementation of a UART Module** |
| **Bani** **Singh** **VLSI July 2025 Batch** **Theory:**  UART (Universal Asynchronous Receiver/Transmitter) is a widely used hardware communication protocol for serial data transmission between devices such as microcontrollers, computers, and peripherals. UART transmits data asynchronously, meaning there is no shared clock signal between the sender and receiver; instead, timing is handled by start and stop bits within each data packet.  **Working Principle**   1. **Serial Communication**: UART sends and receives data one bit at a time over a single line (TX for transmission, RX for reception). 2. **Asynchronous Operation**: UART does not require a clock line. Data integrity and timing are ensured using special bits framing each data byte (start and stop bits).   **Data Packet Structure**:   1. **Start Bit**: Indicates the beginning of a data transfer; typically a low signal (logic 0). 2. **Data Bits**: Typically 5 to 9 bits, representing the actual data (most often 8 bits). 3. **Optional Parity Bit**: Used for error checking. 4. **Stop Bit(s)**: One or two high signals (logic 1) that mark the end of a data packet.   **Transmission Process**: The transmitter UART converts parallel data to serial, frames it, and sends it over the TX line. The receiver UART detects the start bit, samples the data bits according to the agreed baud rate, checks parity, and then delivers the parallel data to the receiving device.  **Baud Rate**: Both UART devices must agree on the baud rate, typically 9600 or 115200 bps, to maintain timing accuracy for reliable communication.  **Applications**: Embedded systems, serial communication with PCs, Bluetooth/Wi-Fi modules, GPS, industrial controllers, and more.  **Advantages**: Simple, cost-effective, robust for point-to-point links, requires only two wires for bidirectional communication.      **code:(software used vivado)** | |

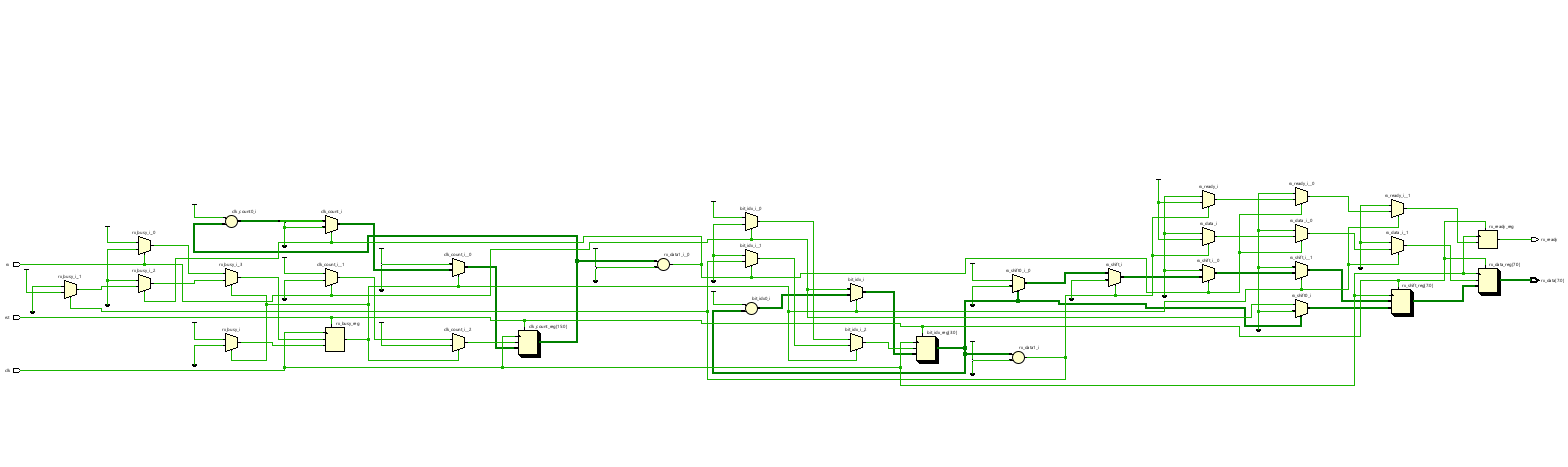


**TEST BENCH CODE:**

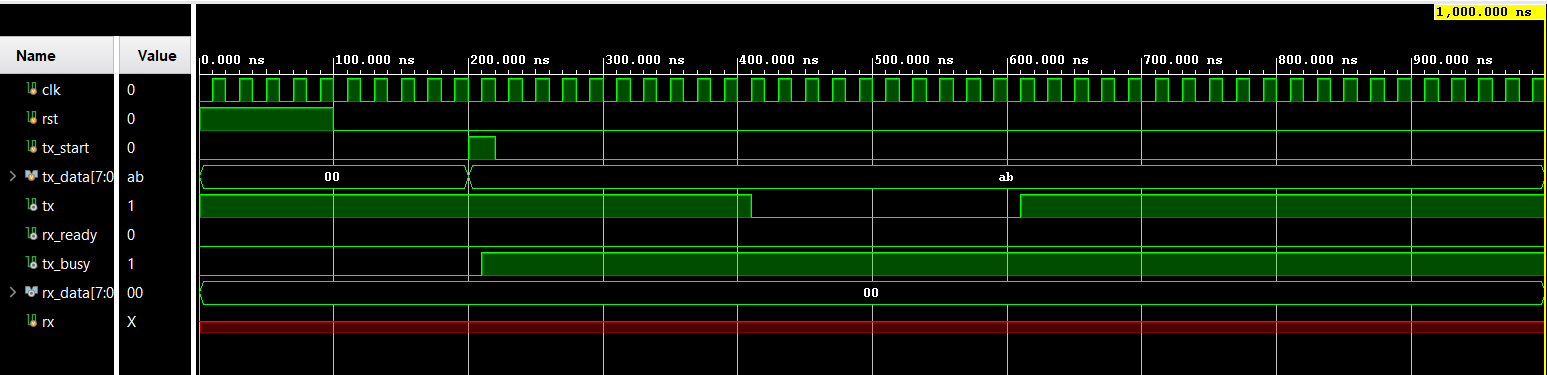
****

****

**SCHEMATIC:**

****

**SIMULATION:**

****