**UART Loopback Project Documentation**

### ****1. Project Overview****

A computer circuit board with many wires

Description generated with high confidenceThis project implements a **UART loopback** system on the **VSDSquadron FPGA Mini**. The received UART data is directly transmitted back, enabling testing of serial communication. Additionally, the onboard **RGB LED** is used as an

indicator.

### ****2. Block Diagram****

The block diagram of the UART loopback system is as

follows:

* **UART\_RX (Pin 15)** → FPGA → **UART\_TX (Pin 14)**
* **FPGA Internal Oscillator** → Clock signal
* **RGB LEDs** indicate received data

**+-------------------+**

**| FPGA |**

**| +--------------+ |**

**clk --->| | UART TX | |**

**| | | |**

**| | TX ------->|-----> TX\_OUT**

**| | | |**

**| | RX <-------|<----- RX\_IN**

**| | UART RX | |**

**| +--------------+ |**

**+-------------------+**

**TX\_OUT is looped back to RX\_IN to create the loopback**

**effect.**

**clk is the clock signal driving the FPGA.**

### A close up of a circuit board Description generated with high confidence****3. Circuit Diagram****

A detailed schematic showing the FPGA pin connections:

| **FPGA Pin** | **Function** |
| --- | --- |
| 14 | UART TX |
| 15 | UART RX |
| 20 | Hardware Clock |
| 39 | Red LED |
| 40 | Green LED |
| 41 | Blue LED |

**+--------------------+**

**| USB-UART |**

**| (FTDI/CP2102) |**

**| |**

**| TX ------------>|------> RX (FPGA)**

**| RX <------------|<------ TX (FPGA)**

**| GND ------------>|------> GND (FPGA)**

**| VCC ------------>|------> VCC (FPGA)**

**+--------------------+**

**+--------------------+**

**| FPGA |**

**| |**

**| TX ------------>|------> RX (USB-UART)**

**| RX <------------|<------ TX (USB-UART)**

**| GND ------------>|------> GND (USB-UART)**

**| VCC ------------>|------> VCC (USB-UART)**

**+--------------------+**

**The USB-UART module (like an FTDI chip) is connected to the FPGA for serial communication.**

**The TX of one device is connected to the RX of the other, and vice versa.**

**GND and VCC connections ensure proper power and signal reference.**

### 

### ****4. UART Transmission Module (8N1 Format)****

The uart\_tx\_8n1 module is responsible for sending UART data. It follows the **8N1 UART format** (8 data bits, No parity, 1 stop bit).

#### **Key Features:**

* **Inputs:**
  + clk → Clock signal.
  + txbyte [7:0] → The byte to be transmitted.
  + senddata → A trigger signal to start transmission.
* **Outputs:**
  + txdone → Indicates that transmission is complete.
  + tx → The UART TX line.

#### **State Machine Implementation:**

* **IDLE (STATE\_IDLE)** → Waits for senddata signal.
* **START TX (STATE\_STARTTX)** → Sends a **start bit (0)**.
* **TRANSMITTING (STATE\_TXING)** → Sends **8 data bits** (LSB first).
* **STOP BIT (STATE\_TXDONE)** → Sends **stop bit (1)**, then returns to IDLE.

### ****5. Implementation Steps****

1. **Synthesize Verilog Code:**
2. make build
3. **Upload to FPGA:**
4. sudo make flash
5. **Open Serial Terminal:**
6. sudo make terminal
7. **Send Data and Verify:**
   * Type in the serial terminal.
   * Verify that the same data is received back.
   * Observe LED behavior.

### ****6. Testing & Verification****

* Use a serial terminal (e.g., picocom or minicom).
* Send test messages and ensure correct loopback.
* Check LED status changes.
* Record a demonstration video.

### ****7. Conclusion****

The **UART loopback** works successfully, verifying FPGA serial communication. The LEDs visually confirm data activity.