Implementation of UART Design for RF Modules Using Different FPGA Technologies

Vivek Kumar¹, Aksh Rastogi² and V.K.Tomar³

Department of Electronics & Communication Engineering
GLA University Mathura 281406

Email: vivek.kumar_ee17@glau.ac.in, aksh.rastogi_ec17@glau.ac.in, vinay.tomar@glau.ac.in.

Abstract. This paper is focused on the designing of UART (Universal Asynchronous Receiver Transmitter) for RF based modules, which is a type of serial communication protocol, mainly used for short-distance, low speed and low-cost data transmission between two embedded systems or computer peripherals. These days RF-Microcontroller modules widely uses UART protocol for the interface with other hardware. A reduced FSM (Finite State Machine) UART design has been implemented on different FPGAs. The maximum achievable speed and their corresponding power consumption are calculated using these FPGAs. The proposed design is implemented using Verilog HDL and for simulation purpose, Xilinx ISE 12.1, 14.7, Xilinx Vivado 2018 and Intel Quartus Prime 19.1 are used. The smallest value of maximum power consumption i.e. 20.67mW at 105.47MHz frequency with 180nm technology node is achieved by Intel Max V among all the Intel and Xilinx devices. Furthermore, the maximum achievable frequency i.e. 769.48 Hz has been found in Virtex 4 FPGA device. Xilinx Virtex UltraScale+ devices occupy the minimum area which provides 16-nm FinFET Technology node.

Keywords—UART, FPGA, FSM, Process Technology, Speed, Power consumption.

1. Introduction
UART (Universal Asynchronous Receiver Transmitter) is a serial communication protocol that uses fewer transmission lines and provides high reliability with long-distance transmission. Also, it is broadly used by computer peripherals and RF-microcontroller-based embedded systems these days [1]. UART offers full-duplex mode serial communication. A single UART chip for data transmission only requires few features of the UART protocol. This paper presents the core functionalities of a proposed UART chip and its implementation on FPGA. FSM (Finite State Machine) is used to make this design more stable and reliable. This design consists of three major parts receiver, transmitter, and clock divider circuitry for baud rate generation [2][3]. Field Programmable Gate Array (FPGA) is a type of semiconductor device that can be programmed using HDL (Hardware Description Language) to behave like a specified HDL design. FPGAs have programmable logic elements called ‘logic blocks’, managed in a hierarchy, having reconfigurable interconnects which assist wiring of these blocks together [3]. UART data frame has been shown in Figure 1. The basic UART design uses only two wires (TX, RX) for full-duplex data transmission. RX is the input for the receiver and TX is used to transmit data. These lines are having two states either HIGH or LOW. In an idle state, transmitter i.e.
TX line remains at HIGH state. When a word is transmitted, an extra bit is added called ‘Start Bit’ to
the beginning of a word. This ‘Start Bit’ is an alert to the receiver that the transmitter has started the
transmission of data. After receiving all data bits, a ‘Stop Bit’ is transmitted at the HIGH logic level to
notify the receiver that data transmission has done [4].

![UART Data Frame](image)

**Figure 1. UART Data Frame**

After receiving all bits by the receiver, it rejects the ‘start bit’ and ‘stop bit’ and convert the serial data
into the parallel data format. To provide synchronization between receiver and transmitter ‘Baud rate
generator’ is used. It’s a simple frequency divider circuitry which generates the clock frequency for
receiver and transmitter at a specific baud rate [5]. In our design standard 9600bps baud rate is used.

### 2. IMPLEMENTATION OF UART CHIP

A UART integrated circuit is designed which consists of a receiving and transmission module. The
purpose of receiving module to convert serial received 8-bit data into parallel data. Transmission
module converts 8-bit parallel data and transmits it serially at a specific baud rate. Baud rate
generation is done by a simple clock divider circuit. This clock divider circuit is utilized to provide the
precise sampling of data bits at both transmitter and receiver side. Where the transmitter transmits data
at that baud rate. To generate the baud rate, the system clock is divided by a clock diving frequency
coefficient (M) which is calculated by the following formula [4].

\[
M = \frac{System\ Frequency}{Factor\ of\ baud\ rate \times Band\ rate} \quad (1)
\]

In our design, the system frequency is \(F_{max} = 100MHz\) and taken baud rate is 9600bps and
Factor of baud rate = 1, SoM = 10,416.

### 3. Receiver Module

During the receiving operation, the receiver receives the serial bit data stream and converts it into
parallel data by rejecting the ‘start bit’ and ‘stop bit’ [4]. Figure 2 shows the states of FSM at the
receiver side. In this design, sampling is done by the state machine using the following states.

1. Idle State: In this state, the receiver module remains in the idle state until it does not receive
   the LOW state. After receiving ‘LOW state’, it moves to the next state i.e. ‘Start State’.
2. Start State: In this state, the receiver’s control checks the status of the start bit. The low level
   of start state implies the ‘start bit’. After sensing the ‘start bit’ it moves to the next state i.e.
   ‘rx_data state’.
3. Rx_data State: In this state, UART starts to receive the 8-bit/16bit-serial data and converts it
   into parallel data. After receiving all the bits, it moves to the next state i.e. ‘Stop State’.
4. Stop State: In this state receiver, gets ‘stop bit’ and halts receiving operation and moves to
   ‘Clear state’.
5. Clear State: In this state, receiver clears all internal registers and acknowledges pin ‘rx_done’, taken to HIGH.

![Figure 2. FSM for Receiver](image)

Only “START”, “RX_DATA” and “STOP” states are proceeding to next state until the value of clock counts become equal to the value of frequency coefficient (M). It changes the state of internal register CLK_COUNT to HIGH. This process divides the system frequency and generates baud rate frequency which helps to achieve the baud rate.

4. Transmitter Module

During the transmission, the transmitter transmits the parallel data by converting it into a serial data stream and includes ‘start bit’ and ‘stop bit’ in a word [4]. Figure 2 shows the states of FSM at the transmitter side.

1. In this design, the transmission is done by the state machine using the following states. Idle State: In this state, the transmitter module remains in the idle state until a HIGH state external input is applied at pin ‘TX_ENABLE’. After the transition from LOW to HIGH at ‘TX_ENABLE’ pin, the control moves to the next state i.e. ‘Hold State’.
2. Hold State: In this state, the transmitter starts to transmit ‘start bit’ by changing the ‘tx’ pin state from HIGH to LOW. After transmitting ‘start bit’ the control moves to the next state.
3. TX_DATA State: In this state, the transmitter starts to transmit the 8-bit or 16-bit data by converting parallel data into serial data. After transmitting all the data bits it moves to the next state i.e. ‘Stop State’.
4. Stop State: In this state, the transmitter transmits the ‘stop bit’ by changing the state LOW to HIGH. After transmitting ‘stop bit’ the control of transmitter moves to next state i.e. ‘Done State’.
5. Done State: In this state, the transmitter clears all internal registers and acknowledge pin ‘tx_done’ is taken to HIGH.

At transmitter side, all the states except “IDEL” and “CLEAR” remains in same state until the state of CLK_COUNT is not at HIGH.
5. Synthesized View of UART chip

This chip embeds both the transmitter and receiver modules and works as a single IC for the transmission as well as receiving purpose. In this IC, an input data is applied at the data[3:0] port which has to be transmitted serially. The rx_serial pin receives the data and output can be seen at the output port i.e. out_data [3:0]. The implementation of this chip is shown in Figure 4.

The pin configurations of the proposed UART circuit are as follows:

1. data[3:0]: The applied data that has to be transmitted through tx_serial(TX).
2. CLK: System clock input i.e. 100MHz.
3. enable: enable pin to enable the transmission.
4. rx_serial: This pin is considered as RX in UART communication.
5. out_data[3:0]: Data output bus when data is received from rx_serial(RX) pin.
6. rx_done: The state of this pin will change to HIGH when receiving of data is completed.
7. tx_act: The state of this pin change to HIGH when the transmission is started.
8. tx_done: The state of this pin change to HIGH when the transmission is completed.
9. **tx_serial**: This pin is considered as TX in UART communication.

6. **Synthesized view of UART connections**

![ UART Connections Diagram ]

**Figure 5.** Assembly between two UARTs

Fig. 5 shows the connection between both UART assemblies. The first chip is acting as a transmitter and transmitting the data which is applied at the input of the transmitter. The second chip is working as a receiver which is receiving the data serially and output can be seen at the output bus of the receiver [6][7]. Both chips are connected in a cross-couple manner. Transmitter’s TX terminal is connected to the receiver’s RX terminal and vice-versa.

7. **Simulation Results**

The simulated waveforms of the proposed UART assembly are shown in Fig.6. In this simulation, a 4-bit data “1100” is applied on the input port “t_data[3:0]” of “d1” instance and transmitted serially. The received data appeared at the output port “r_odata[3:0]” of the “d2” instance. The same process has been performed at another side. The data is sent from “d2” instance with port “r_data[3:0]” and received at the instance “d1” at port “t_odata[3:0]”. After 300µs data is changed to “0011” and the same process has been repeated.

![ Simulation Waveforms ]

**Figure 6.** Simulation of Receiver and Transmitter
7.1. COMPARISON OF IMPLEMENTED UART DESIGN ON DIFFERENT FPGAS

For the performance improvement and power reduction on a complete RF-SoC module, a detailed comparative analysis in terms of power, speed and area has been performed of UART design. The maximum achievable speed and its corresponding maximum power consumption are calculated at different FPGAs technology nodes for a single UART chip. The obtained results are summarized in the form of table 1. The considered FPGAs are shown table 1 are plotted in Fig. 7. It can be seen from Fig. 7 that Cyclone 10 LP FPGA has lowest value of maximum power consumption as compared to “Cyclone V” and “Arria II GX” FPGAs.

| S. No | Family       | Part No.           |
|-------|--------------|--------------------|
| 1     | Cyclone V    | 5CGXFC3B6F23C6     |
| 2     | Arria II GX  | EP2AGX45CU1713     |
| 3     | Cyclone 10 LP| 10CL006YE144C6G    |

Table 1. Intel Devices for Comparison

Figure 7. Chart for Intel Devices

Table 2 and Figure 8 shows the obtained values for “Intel Cyclone IV E” made on 60nm technology node, “MAX II” made on 180nm technology node and “MAX V” made on 180nm technology. The. It is observed that Cyclone IV E FPGA has the lowest value of maximum power consumption and best speed performance as compared to “MAX II” and “MAX V” CPLDs.

| S. No | Family     | Part No.     |
|-------|------------|--------------|
| 1     | Cyclone IV E| EP4CE6E22C6  |
| 2     | MAX II     | EPM240T100C3 |
| 3     | MAX V      | 5M160ZM68C4  |
In table 3, the Xilinx devices of different families are compared. In this comparison “Spartan 3” made on 90nm technology node, “Spartan 6” made on 45nm technology node and “Spartan 7” made on 45nm are taken into consideration and their comparison chart is shown in Figure 9. It can be depicted from Figure 9 that Spartan 6 FPGA has the lowest value of maximum power consumption among Spartan 3 and Spartan 7 FPGAs.

In table 4, the Xilinx devices of different families are compared. In this comparison “Xilinx Virtex 4” made on 90nm technology, “Virtex 5” made on 65 nm technology node and “Virtex 6” made on 40nm technology node summarized in table 4 and their values are plotted in Figure 10.
Table 4. Xilinx Devices for Comparison-II

| S. No | Family    | Part No.  |
|-------|-----------|-----------|
| 1     | Virtex 4  | XC4VLX15  |
| 2     | Virtex 5  | XC5VLX30  |
| 3     | Virtex 6  | XC6VLX75T |

Figure 10. Chart-II for Xilinx Devices

Figure 10 shows that Virtex 4 has the lowest value of maximum power consumption with the highest frequency among Virtex 5 and Virtex 6 FPGAs. In the third comparison, the “Virtex 7” made on 28nm technology node, “Virtex Ultra Scale” made on 20nm technology node and “Virtex Ultra Scale+” made on 16nm technology node are included and presented in the table. 5. The values presented in Figure 5 are plotted in Figure 11. It can be seen from Figure 11 that Virtex 7 FPGA has the lowest value of maximum power consumption and Virtex Ultra Scale devices has highest value of $F_{max}$ as compared to Virtex7 and VirtexUltraScale+ FPGAs.

Table 5. Xilinx Devices for Comparison-III

| S. No | Family             | Part No. |
|-------|--------------------|----------|
| 1     | Virtex 7           | XC4VLX15 |
| 2     | Virtex UltraScale  | XC5VLX30 |
| 3     | Virtex UltraScale+ | XCVU3P   |

Figure 11. Chart-III for Xilinx Devices
In this comparison the Virtex UltraScale and Virtex UltraScale+ FPGA have shown a good performance response in terms of speed but they are having more power consumption because, in every VLSI design there is always one parameter trades off among the power, speed and area. The Virtex UltraScale and UltraScale+ devices are having shortest gate lengths. The summarized values in table 5 are plotted in Figure 11. All the above simulations and calculations are done at the ambient temperature of 25°C and the maximum ambient temperature is 84°C with airflow of 0(LFM) and the part specification and technology node (Gate length) information is taken from part’s datasheet for all simulations.

8. CONCLUSION

In proposed work, we have compared the performance of proposed UART design with FPGAs of different process technologies. It has been found that Xilinx Virtex 4 FPGA has a maximum value of frequency i.e. \( F_{\text{max}} = 769.48 \text{MHz} \) among all considered FPGAs. Furthermore, Intel MAX V has the smallest value of maximum power consumption i.e. \( P_{\text{max}} = 20 \text{mW} \) among all considered FPGAs. Based on the performed analysis, Xilinx Virtex 4 FPGA device has the best speed and power performance with 90nm technology node for proposed UART design.

References

[1] Zheng Qiao, Hu Wei-Dong and Liu Li, "An embedded short-range wireless data communication protocol design," 2009 International Conference on Apperceiving Computing and Intelligence Analysis, Chengdu, 2009, pp. 399-402, doi: 10.1109/ICACIA.2009.5361072.

[2] D.V.R. K Raju, B. Narsingarao, K. Vikash, D. Akshay Kumar and S. Ramakrishna, “UART Serial Communication Module Design and Simulation Based on VHDL,” International Journal of Engineering Research and Applications (IJERA), 2015.

[3] FANG Yi-yuan and CHEN Xue-jun, “Design and Simulation of UART Serial Communication Module Based on VHDL,” International Workshop on Intelligent Systems and Applications, 2011.

[4] A. Kumar, B. Pandey, D. M. Akbar Hussain, M. Atiqur Rahman, V. Jain and A. Bahanasse, "Low Voltage Complementary Metal Oxide Semiconductor Based Energy Efficient UART Design on Spartan-6 FPGA," 2019 11th International Conference on Computational Intelligence and Communication Networks (CICN), Honolulu, HI, USA, 2019, pp. 84-87.

[5] Biswajit Roy Dakua, Md. Ismail Hossain and Foisal Ahmed, “Design and Implementation of UART Serial Communication Module Based on FPGA,” International Conference on Materials, Electronics & Information Engineering, ICMEIE-2015.

[6] Ambika, Prof. Anuradha S, “High Speed UART Design Using Verilog,” International Journal of Advanced Research in Computer and Communication Engineering, 2016.

[7] L. V. Bogdanov, "A Wi-Fi to UART Bridge for Firmware Updates of Microcontrollers," 2019 IEEE XXVIII International Scientific Conference Electronics (ET), Sozopol, Bulgaria, 2019, pp. 1-4.

[8] Sung-Mo Kang, Yusuf Leblebici and Chulwoo Kim, “CMOS Digital Integrated Circuits, Analysis and Design,” McGraw Hill Education; Fourth edition (10 January 2019).

[9] K. Kumar, A. Kaur, S. N. Panda and B. Pandey, "Effect of Different Nano Meter Technology Based FPGA on Energy Efficient UART Design," 2018 8th International Conference on Communication Systems and Network Technologies (CSNT), Bhopal, India, 2018, pp. 1-4.

[10] Intel_FPGA_Products, “www.intel.in/content/www/in/en/products/programmable/fpga.html”.

9
[11] Xilinx_FPGA Products,”https://www.xilinx.com/products/silicon-devices/fpga.html”.