FPGA Based Implementation of UART bus for AES application

Mayank M. Bhanani[1], Prof. Deepak R. Parashar[2]

[1]PG Scholar, Electronic & Communication Department, G.H. Patel Engineering College, Anand, India
[2]Professor, Electronic& Communication Department, G.H. Patel Engineering College, Anand, India

ABSTRACT

UART (Universal Asynchronous Receiver Transmitter) is a serial communication bus which mostly used for short-distance, low speed, low-cost data exchange between computer and peripherals. Here in this paper we are going to implement UART on the FPGA board (Altera DE1 – EP2C20F484C7). The synthesis and simulation is done on the licensed Altera Quartus II 6.0 web addition. Whole coding is done in the VHDL language and implemented on FPGA to achieve compact, stable and reliable data transmission.

Keyword *UART; asynchronous serial communication; VHDL; Quartus II; simulation;DE1 board*

1. INTRODUCTION

UART is a serial communication bus which converts the parallel data into serial data and transmit it in a serial manner at transmitter side and at the receiver side it converts the serial data input to the parallel data output. UART allows full-duplex communication in serial link, thus has been widely used in the data communications and control system. In actual applications, usually only a few key features of UART are needed. Specific interface chip will cause waste of resources and increased cost [1][2]. Basic UART communication needs only two signal lines (RXD, TXD) to complete full-duplex data communication. TXD is at the transmit side, the output of UART; RXD is at the receiver side [3]. Using UART we can connect two devices in full duplex mode up to maximum 30 meter range and data transfer can take place at the maximum of 115.2kbps speed. It is advantageous as simple resources, reliable performance, strong anti-jamming capability, easy to operate and realize. The UART transmission line is at high state (NRZ state) normally then first the start bit – 0 (low) then 8 bit data then parity bit (optional) and at last there will be a stop bit – 1 (high) as shown in figure 1.

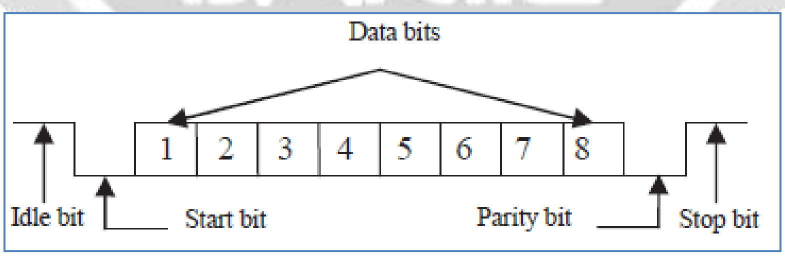

Figure 1 UART bus frame format

2. HARDWARE AND SOFTWARE

2.1 Hardware: Altera DE1 Board EP2C20F484C7

Here, we are going to use this FPGA board to implement this system. It comes with the hardware like Altera Cyclone II 2C20 FPGA device, Altera Serial Configuration device – EPCS4, USB Blaster (on board) for programming and user API control; both JTAG and Active Serial (AS) programming modes are supported, 512-Kbyte SRAM, 8-Mbyte SDRAM, 4-Mbyte Flash memory, SD Card socket, 4 pushbutton switches, 10 toggle switches, 10 red user LEDs, 8 green user LEDs, 50-MHz oscillator, 27-MHz oscillator and 24-MHz oscillator for clock sources, RS-232 transceiver and 9-pin connector, Two 40-pin Expansion Headers with resistor protection, Powered by either a 7.5V DC adapter or a USB cable.

2.2 Software: Altera Quartus II 6.0 web addition

The Altera Quartus II design software provides a complete, multiplatform design environment that easily adapts to our specific design needs. It is a comprehensive environment for system-on-a-programmable-chip (SOPC) design. The Quartus II software includes solutions for all phases of FPGA and CPLD design.

In addition, the Quartus II software allows us to use the Quartus II graphical user interface and command-line interface for each phase of the design flow. We can use one of these interfaces for the entire flow, or we can use different options at different phases.

3. IMPLEMENTATION OF UART

Here, we will describe the FPGA implementation of the UART bus. There will be total three modules of the VHDL coeds 1st is the UART control module, 2nd is the TRANSMITTER module and 3rd is the RECEIVER module.

3.1 UART control module

This unit will control the transmission and reception by giving the component mapping to the TRANSMITTER and RECEIVER unit. This will be the main unit having the inputs as clock of 50MHz, switches as input data, key to trigger, RD for receive data serial line. Here, key and switches used are available on the Altera DE1 board. It will have the output as LEDR (red LED) for received data output, LEDG (green LED) to show input transmission data, TD for transmission serial line.

3.2 TRANSMITTER module

Here, in this module we are giving input data that is to be transmitted by using the switches on the DE1 board. Then trigger the transmission by key and input clock. As the transmitter starts it reads the data on the switches (parallel data) and stores in the data buffer then by using the prescaler logic for the UART system, the data from the buffer is serially transmitted on the TD output serial line. Also we can see the parallel data input on the green LED. Block diagram is shown in figure 2.

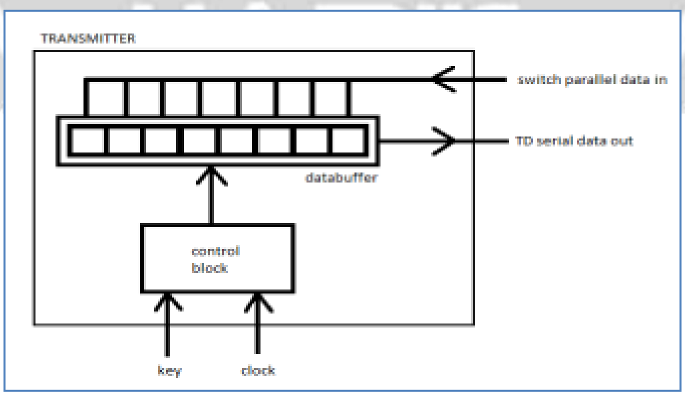

Figure 2 Transmitter

3.3 RECEIVER module

Here, in this module we get the serial data at RD serial line, which is given to the databuffer and the databuffer is filled with the received data according to the prescaler logic of getting data in UART system. The reception will start if no communication is active. After that the parallel data is fetched from the databuffer to the red LED. Block diagram of receiver is shown in figure 3

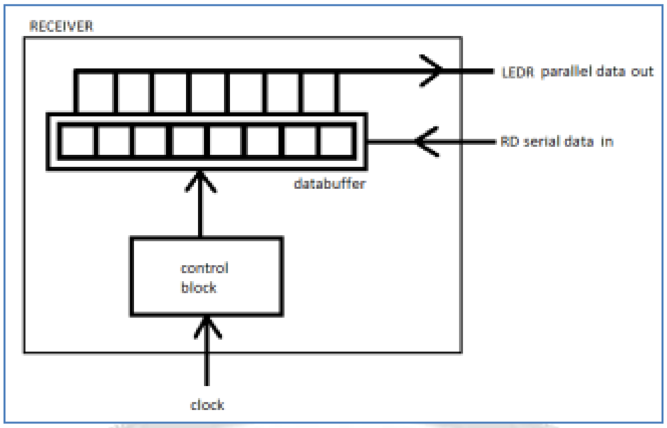

Figure 3 Receiver

4. SIMULATION, SYNTHESIS AND IMPLEMENTAION RESULTS

4.1 Device Utilization

Below figure 4 shows the details about the device utilization by logic cells, logic cell combinational, logic cell registers, total pins, look up tables of LCs, registers of LC & LUT/Register of LCs used to deploy the codes on the DE1 board.

Entity	Logic Cells	LC Combinationals	LC Registers	Pins	LUT-Only LCs	Register-Only LCs	LUT/Register LCs
Cyclone II: EP2C20F484C7							
UART	103 (26)	74 (2)	74 (25)	35	29 (1)	29 (24)	45 (1)
TRANSMITTER:C1	28 (28)	24 (24)	20 (20)	0	8 (8)	4 (4)	16 (16)
RECEIVER:C2	49 (49)	48 (48)	29 (29)	0	20 (20)	1 (1)	28 (28)

Figure 4 Device Utilization

4.2 RTL view

Here, in the below figure 5 the RTL view of the whole system is shown, Where we can see the inputs and outputs ports and the transmitter and receiver blocks.

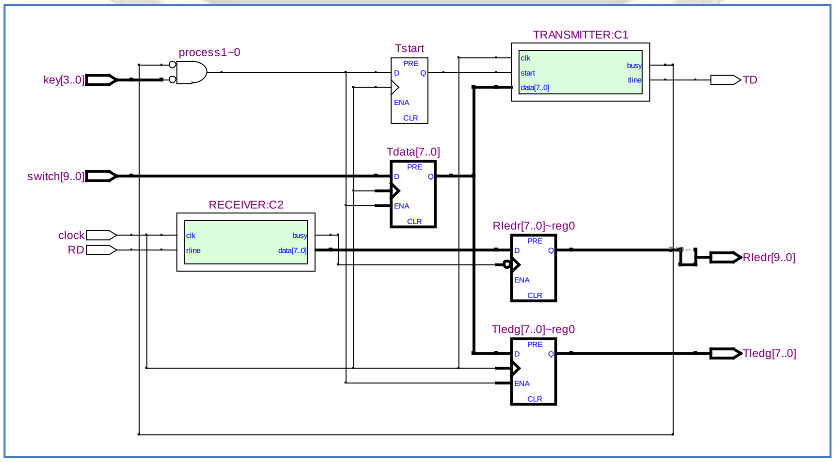

Figure 5 RTL view of the UART

4.3 Simulation waveform

After simulating the codes we get the following output waveforms. Here, the switches are the transmitter parallel data input and the TD is the transmitter output in a serial manner. For receiver, the RD is the serial data input and the Rledr is the parallel output (see figure 6).

Figure 6 simulation waveforms

4.4 Implementation on DE1 board

By implementing this system on the Altera DE1 FPGA board we get the output as expected. First we assign the pins of the board according to our VHDL code requirement and then using the programmer tool in the Quartus II we implemented the UART system on the FPGA. We have assigned the clock to system 50MHz clock, switches to switches (01001001), Rledr to red LED, Tledg to green LED, TD to GPIO_1 pin 0 and RD to GPIO_0 pin 0. Then to receive the data which is transmitted we connected (shorted) the GPIO_0 pin 0 and GPIO_1 pin 0. So by doing this we get the expected result as shown in figure 7.

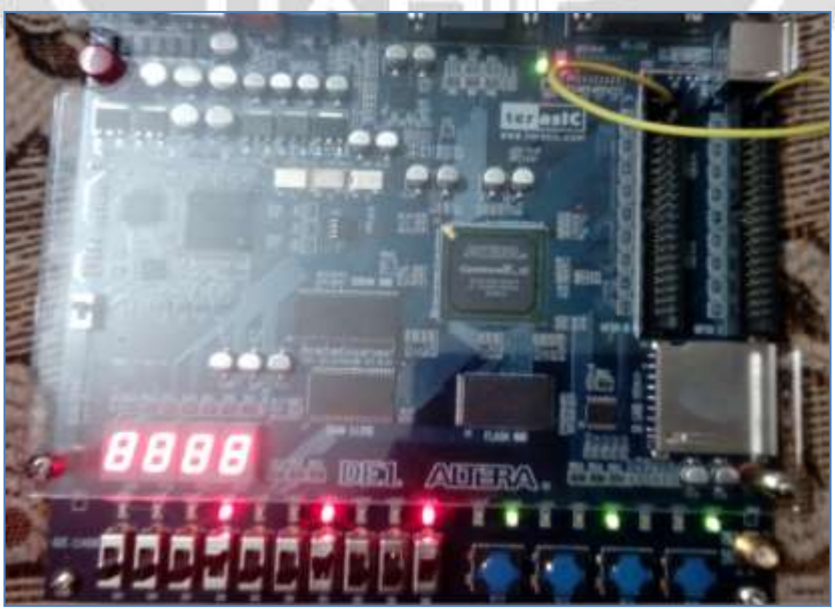

Figure 7 Implementation on board

5. CONCLUSIONS

This design uses VHDL as design language to achieve all the modules of UART. Using Quartus II software, Altera's CycloneII series FPGA chip EP2C20F484C7 to complete simulation and test. The results are stable and reliable. The design has great flexibility, high integration. One can make the use of this design with any other functions or applications.

6. REFERENCES

[1]. Liakot Ali, Roslina Sidek, Ishak Aris, Alauddin Mohd. Ali, Bambang Sunaryo Suparjo. Design of a micro - UART for SoC application [J].In: Computers and Electrical Engineering 30 (2004) 257–268.
[2]. HU Hua, BAI Feng-e. Design and Simulation of UART Serial Communication Module Based on Verilog - HDL[J]. J ISUANJ I YU XIANDA IHUA 2008 Vol. 8
[3]. Frank Durda Serial and UART Tutorial. uhclem@FreeBSD.org
[4]. FANG Yi-yuan CHEN Xue-jun. Design and Simulation of UART Serial Communication Module Based on VHDL IEEE 2011
[5]. He Chun-zhi Xia Yin-shui Wang Lun-yao A Universal Asynchronous Receiver Transmitter Design IEEE 2011
[6]. Dr. Garima Bandhawarkar Wakhle, Iti Aggarwal and Shweta Gaba Synthesis and Implementation of UART using VHDL Codes IEEE 2012.
[7]. DE1_UserManual_v1018.pdf