ASIC Implementation of Universal Asynchronous Receiver and Transmitter using 45nm Technology

B.Venkataramana[1], P. Srikanth Reddy[2], P. Kishore Kumar[3], K Sivasankaran[4]
[1,2,3]Student, M.Tech (VLSI Design),SENSE, VIT University, Vellore,Tamilnadu, India
[4]Assistant Professor(Senior), SENSE, VIT University, Vellore,Tamilnadu, India
[4]ksivasankaran@vit.ac.in

***Abstract*— This paper presents design of UART module for serial communication used for short-distance, low speed and exchange of data between computer and peripherals. UART mainly contains Transmitter, Receiver and Baud Rate Generator. Baud Rate Generator generates the clock for the UART. We can achieve the desired Baud Rate by using divide factor from system clock. If we increase the baud rate, speed of serial data transmission increases. As the dividing factor decrease baud rate increases. in this paper we set the system clock frequency as 50MHz and time to transfer each data bit is 23.75ns with baud rate of 42.1 Mbps (dividing factor is 32). Due to increase in the baud rate the time taken to transfer the data decreases, so it is very useful for faster communication devices. Transmitter and Receiver blocks designed by algorithm state machine method simulated in Cadence IUS, synthesized in Cadence RTL Compiler in TSMC 45 nm CMOS cell library.**

Keyword- Baud rate generator, Frequency divider, Receiver, Transmitter, UART.

I. INTRODUCTION

Universal asynchronous receiver transmitter (UART) is an integrated circuit which is used for the serial data communication in the field of telecommunication. UART allows full duplex communication in a serial transmission and asynchronous serial communication is established with this so that it is widely used in the data communication. UART design mainly contains three blocks those are Transmitter, Receiver and Baud Rate Generator. Receiver performs the serial to parallel conversion on the asynchronous data frame received from the serial data input. Transmitter performs parallel to serial conversion on the data received from CPU [1-4]. Each UART contains shift register which is fundamental method of conversion between serial and parallel forms. The UART requires buffer for storing high speed data transmissions. UART data frame format is as shown in the figure 1.

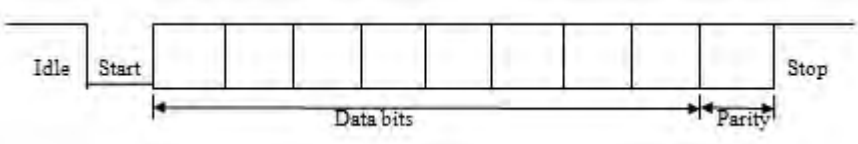

Fig.1 UART Data Frame Format

Data Frame contains 8-bit data, 1-start bit, 1-stop bit and 1-parity bit (optional bit). When start bit is low that indicates to the receiver to receive the serial input data from the transmitter. After 8-bit data is transmitted completely one parity bit will be generated by the transmitter and added to the data. Parity bit is either even or odd to indicate there is any error in the data received which is obtained from the transmitter, which means the parity used should be same for transmitter data and receiver data. The sum of all bits is odd number means odd parity. Even parity means getting sum of all bits in even. After data and parity bits are received the receiver checks for the stop bit if it does not appear when it is supposed to occur, the UART considers it as an error which is usually due to the difference in speeds of transmitter and receiver clocks or due to any signal interruption. The receiver of UART knows the baud rate of incoming data so that baud rate of data transmission must be known. There are different methods are used to detect the baud rate; one of the method is Phase Locked Loop (PLL). The PLL adjusts the clock frequency to synchronize the incoming data.

II. IMPLEMENTATION OF UART

In UART separate clocks are generated to synchronize the transmitter and receiver [1-4].The three main modules in UART are Transmitter, Receiver and Baud Rate Generator. Baud Rate Generator (BRG) is a block which generates the clock which is used by the Transmitter and Receiver also it is the main factor that decides the total frequency of operation for all the modules used in the circuit. Shift registers are used in Transmitter and Receiver designs to transfer the data from parallel to serial and vice-versa. The block diagram of the UART module is as shown in figure 2.

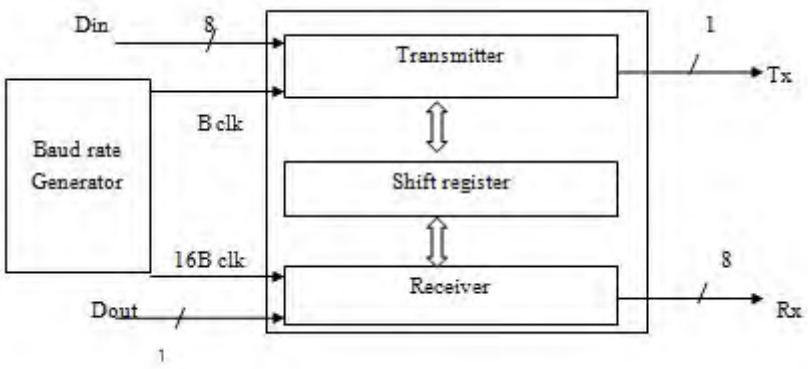

Fig. 2 Block Diagram of UART

In this paper the UART is designed using Finite state machine modelled in Verilog HDL. The Synthesis is done using RTL compiler and physical design is done by using Cadence SoC encounter.

1. Transmitter Module

The main function of the Transmitter module is to convert the 8-bit parallel data to serial bit data using shift registers. When load_shftreg is high, 8-bit parallel data is stored into transmitter hold register (THR) and after data in the THR is shifted to transmitter shift register (TSR) when shift is high [4]. Each data will be transmitted to the receiver starting from the LSB bit until Bit_count condition is false. The serial bit data is transmitted at the rate of 1/16 of clk16x frequency.

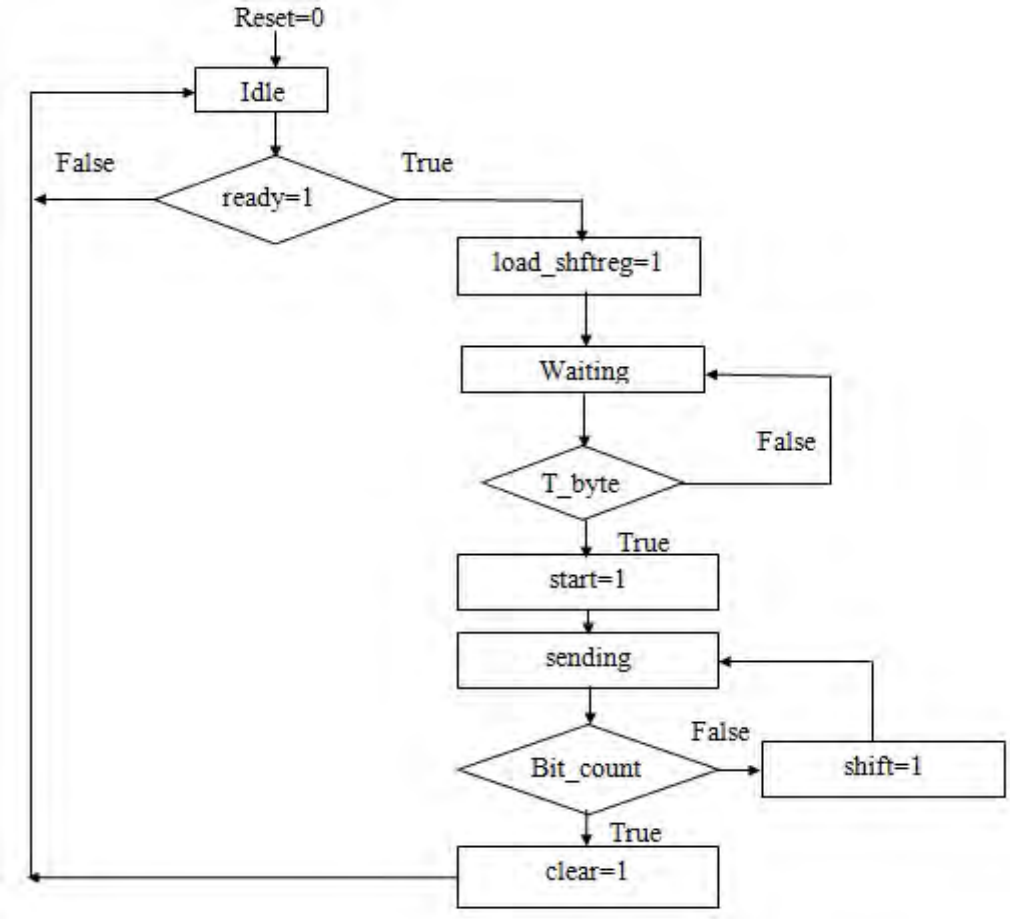

Fig 3: ASM chart of Transmitter

After transmission of data frame another data will be transmitted immediately when Transmitter Hold Register is not empty. Whenever the total data has been sent transmitter provides a parity bit this will serves as a input to the receiver. Algorithmic State Machine (ASM) is used for designing finite state machines. In order to design ASM chart we should follow certain conditions and it offers several advantages. This state charts can be easily converted into other forms. It is easier to understand and represents sequential operations in a given

digital system. ASM chart gives functional description of combinational and sequential circuits.

The ASM chart of a state machine controller for the Transmitter is shown in the figure 3. Initially Transmitter is reset to idle state until reset is not asserted, if ready bit and T_byte are asserted it will load the data in to transmitter hold register [4]. After start bit is asserted sends the data to Transmitter Shift Register and transmitted each data bit till shift is high. Transmitter simulation waveforms are shown in figure 4. In the waveform AAH (1010_1010) is the 8-bit data frame which will be transmitted as single bit serial data serial_out.

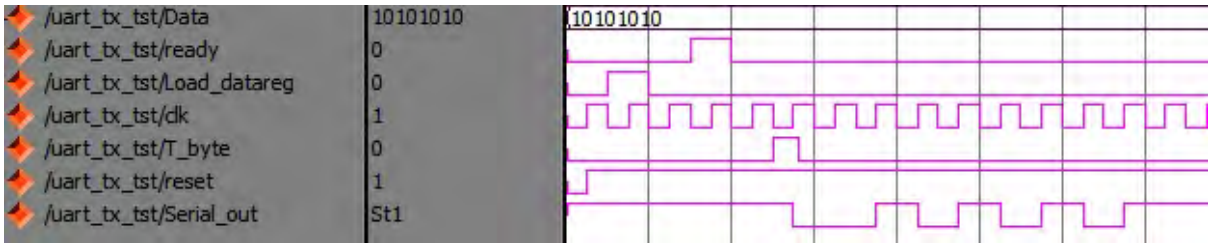

Fig 4: Simulation wave form of Transmitter

2. Receiver Module

The main function of the Receiver module in UART is to convert the serial data into parallel data using shift registers. A high to low transition in the serial input may be treated as a start bit in the receiver module. The start bit is detected if it is at least 50% of the receiving clock because internal clock clk16x is 16 times the receiving clock frequency so start bit is valid if it is low up to 8 clk16x clock cycles. After start bit is detected each serial bit data coming from transmitter will be stored in receiver shift register. When load is high the data in RSR will be loaded into received data register[4-7]. The ASM chart of a state machine controller for the receiver is shown in figure 5.

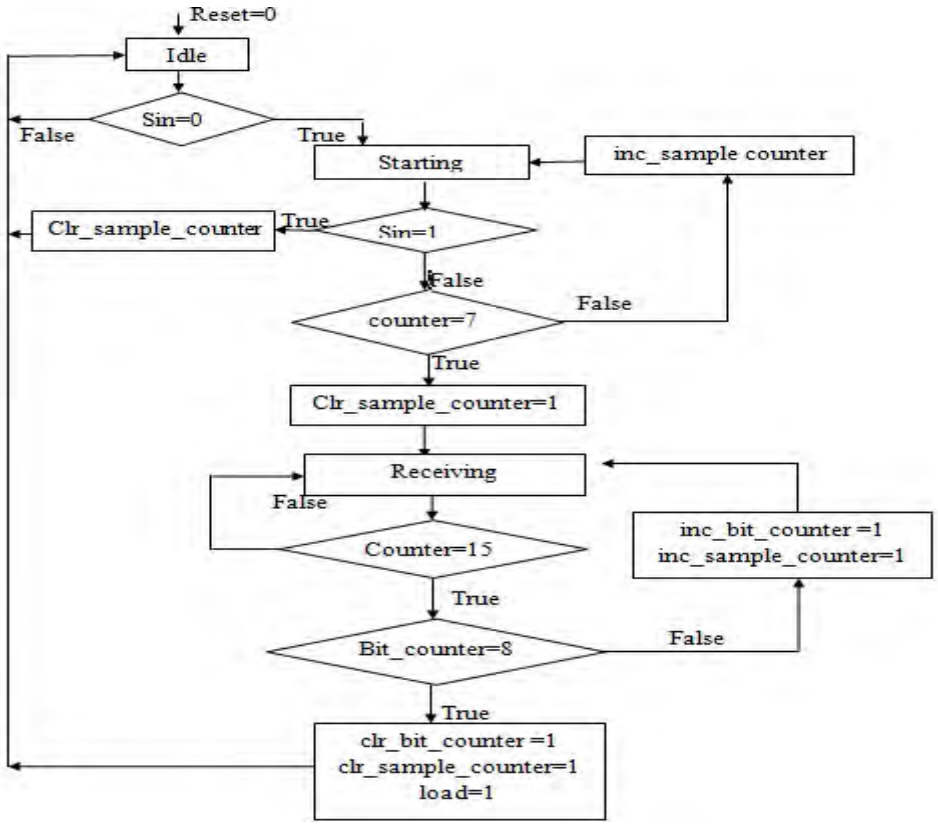

Fig 5: ASM chart for Receiver

The machine has three states: idle, starting, and receiving. The serial input data from transmitter is loaded into receiver shift register in parallel manner is shown in receiver simulation waveform in figure 6. In that when shift is high the serial data will be loaded into receiver shift register and shift by 1-bit. When stop bit is detected the data in the receiver shift register is loaded into receiver data register by asserting load value as high.

In the idle RCV_datareg value should be high. whenever the Sin is low for at least 50%of system clock then the start bit is initiated after that it samples the signal equal clock periods after receiving the data it is sent to the RCV_datareg.for the synchronization of clock baud rate generator generates clocks.

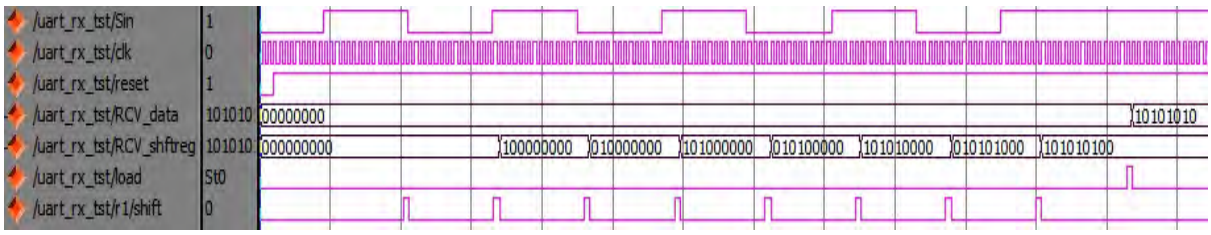

Fig 6.Simulation wave form of receiver

3. Baud Rate Generator:

Baud Rate Generator is generally a frequency divider. Baud rate frequency factor can be obtained with the help of system clock frequency and the required baud rate and that factor is used as a divider. In this design we apply synchronized clock to both receiver and transmitter. The clock applied to receiver is 16 times to that of transmitter clock[8]. Assume system clock frequency is set to 24MHZ and required baud rate is 9600bps then the divide factor is obtained as 156.

Divide Factor=System clock frequency/16*Baud rate

Baud Rate means the rate at which the amount of data has to be transmitted. The purpose of Baud Rate Generator is to precisely sample the serial input data at the receiver because it is difficult to detect where the sample has to be done in the serial data. The simulation waveform of the Baud Rate Generator is as shown in figure 7. In this design we applied the clock frequency as 16 times to the baud rate clock.

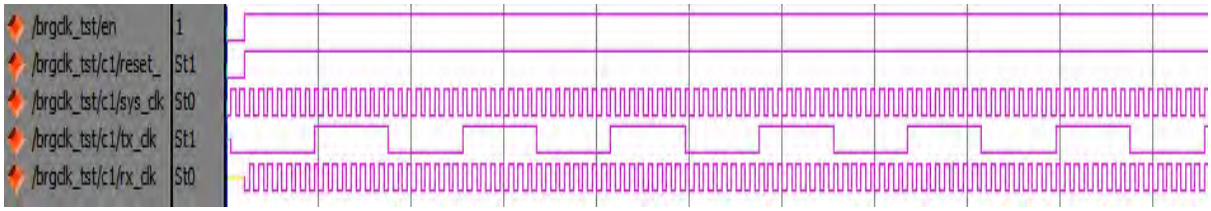

Fig 7: Simulation waveform of BRG

III. RESULTS AND DISSCUSSIONS

In transmitter section the data frame value is AAH which will be converted into serial data and received by the receiver is shown in the simulation waveform of UART in the above figure 8. The three important modules of UART are receiver transmitter and baud rate generator are synchronized to get the functionality [8].

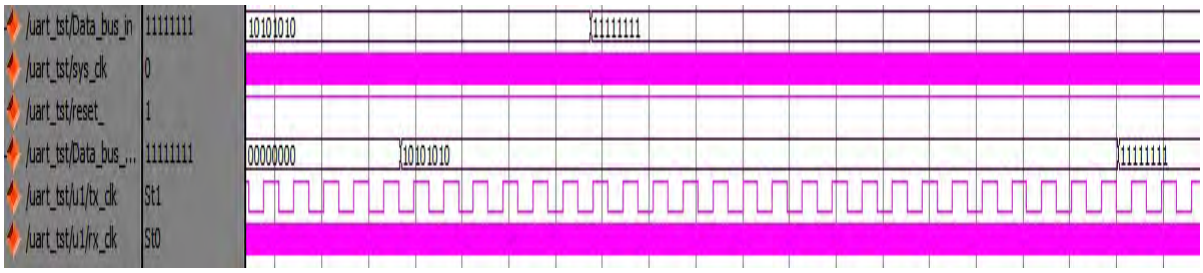

Fig 8.Simulation waveform of UART module

The UART module is synthesized in RTL compiler using TSMC 45nm technology. The total Power and area consumed by the individual modules and the integrated UART module in the 45nm technology is shown in the below table1.

Table 1. Power and area consumed at 45nm technology

Module	Area(sq.nm)	Total power(nw)
Transmitter	239	9518.169
Receiver	324	3143.230
BRG	49	1571.49
UART	603	14410.887

In our design system clock frequency is set to 50 MHZ. the time taken for 8-bit data transfer is 0.19 us. So time to transfer each data bit is 0.19 us/8=23.75 ns. From this baud rate can be calculated as 1/23.75ns=42.1Mbps.The below table.2 indicates that by varying baud rates the taken to transfer the data is reduced [7-12]. From the results it clearly shows decrease in the time interval by increasing the baud rate. The high speed UART transmits the serial data at the rate of 1 bit each 23.75ns (equivalent to the baud rate of 42.1Mbps), 1.4us (equivalent to the baud rate of 714.285Mbps) and 15.2ms (equivalent to the baud rate of 65.78Kbps) using 50 MHZ system clock.

Table.2 Comparison table at different baud rates

Freq Divisor	Time interval[8Data bits]	Baud
1	0.19us	42.1Mbps
2	11.2us	714.285Kbps
32	121.6us	65.78Kbps
256	3.52ms	2.272Kbps

The physical chip design of UART is implemented using Cadence SoC Encounter RTL-to-GDSII. It offers flexible and more accurate results.

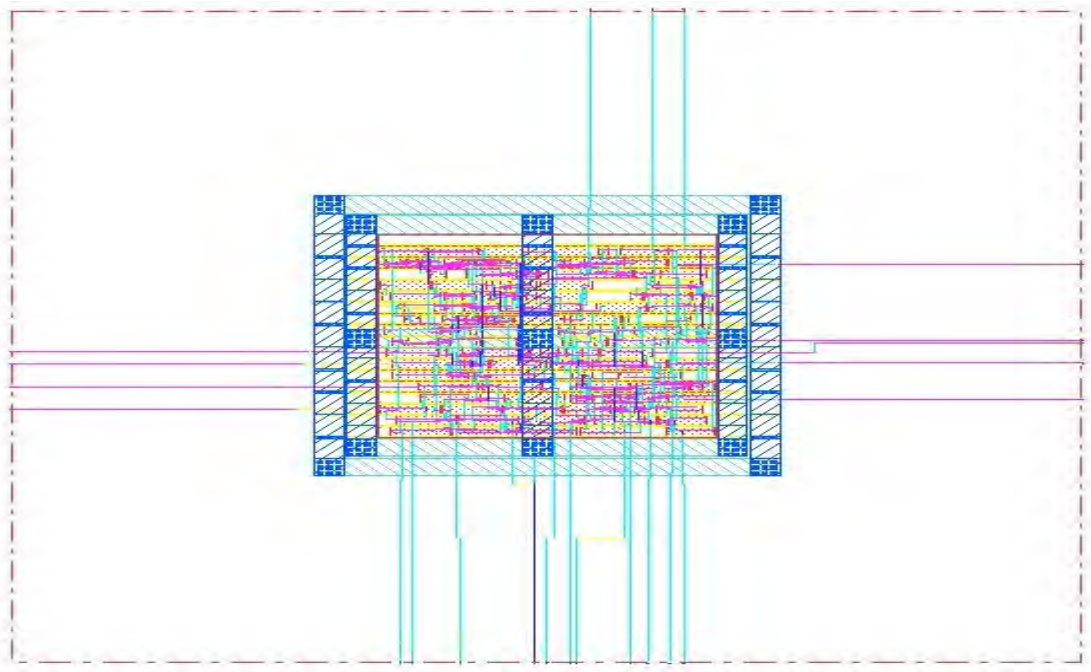

Fig 10. Physical chip design of UART

The time summary of the design after physical implementation is listed in Table 3, which shows the worst negative slack as the positive value indicating there are no violations in the design.

Table 3. Time design summary of UART

Setup Mode	**all**	**in2Out**
WNS(ns)	0.123	0.123
Violating Paths	0	0
All Paths	1	1

IV. CONCLUSIONS

In this work we designed and implemented UART using TSMC 45nm Technology. By changing the division factor we achieve the desired baud rate, the baud rate increases the speed of serial data transmission. The high speed UART transmits the serial data at the rate of 1 bit each 23.75ns (equivalent to the baud rate of 42.1Mbps), 1.4us (equivalent to the baud rate of 714.285Mbps) and 15.2ms (equivalent to the baud rate of 65.78Kbps) using 50 MHZ system clock. The UART were designed using Verilog HDL and it is simulated to check the functionality of each module using Cadence IUS followed by synthesis using RTL compiler. Finally, full chip implementation done by using Cadence SoC Encounter.

REFERENCES

[1] Nennie Farina Mahat, " Design of 9-bit UART Module Based on Verilog HDL," IEEE-ICSE2012 proc. ,2012,Kuala Lumpur, Malaysia, pp570-573.

[2] He Chun-zhi, Xia Yin-shui and Wang Lum-yao, "A Universal Asynchronous Receiver Transmitter Design," Natural Science Foundation of china(No.61041001)2011, pp.691-694.

[3] Y. Fang Yi-yuan and Chen Xue-jun, "Design and Simulation of UART Serial Communication Module Based on VHDL," International Workshop on Intel Sys and App.(ISA 2011),May2011.

[4] Garima Bandhawarkar and Iti Agarwal and Shweta Gaba, "Synthesis and Implementation of UART using VHDL codes, " IEEE International Symposium on Computer, Consumer and Control,2012, pp.1-3.

[5] Michael D. Ciletti , “Advanced digital Design with the Verilog HDL”, Prentice –hall of India private Limited,2005, pp. 378-402.

[6] Nurul Fathish Jusoh, Muhammad Adib Haron, “ An FPGA implementation of shift converter block technique on FIFO for UART ", 2011 IEEE International RF and Microwave conferences (RFM 2011) , pp.320-324.

[7] Shouqian, Y., Y. Lili, " Implementation of a Multi channel UART Controller Based on FIFO Technique and FPGA", 2nd IEEE Conference on Industrial Electronics and Applications, 2007. ICIEA 2007.

[8] J. Norhuzaimin and H.H Maimun, ”The design of high speed UART”, Asia pasific Conference on Applied Electromagnetic Proceedings, 2005 , pp. 306-310.

[9] Debjani Basu and Dipak K Kole, “implementation of AES algorithm in UART module for secured data transfer”, 2012 international conference on advances computing computations, pp.142-145.

[10] Vivek ramakrishnan and Rahul ashok, ”FPGA based DC Servo motor control for Remote Replication of Movements of a Surgical arm”, tencon 2011,pp.671-675.

[11] Xin Wang, Tapani Ahonen and Jari Nurmi, " A Synthesizable RTL Design of Asynchronous FIFO", Tampere, Finland, nov.2004.

[12] Yongcheng Wang and Kefei Song "A New Approach to Realize UART", International Conference on Electronic & Mechanical Engineering and Information Technology, Aug 2011.pp.2749-2752.