International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-4 Issue-7, December 2014

Design of UART Protocol with Interrupt Logic and Status Register

Salman Khan B. R, Arun Patro, Siva S. Yellampalli

Abstract: ***Universal Asynchronous Receiver Transmitter (UART) is used in data communication process especially for its advantages of high reliability, long distance and low cost. This paper targets the interrupt logic and Status register to UART. The 8-bit UART with status register and Interrupt module is coded in Verilog HDL and synthesized and simulated using Xilinx ISE version 12.2 and Modelsim. 9600bps Baud Rate is used for Proposed Architecture. 207.220MHZ maximum frequency is obtained from Spartan 3e Xc3s400.In Proposed Architecture 25MHZ is used as system clock.***

Keywords: Universal Asynchronous Receiver Transmitter, Status Register , Asynchronous Serial Communication.

I. INTRODUCTION

Asynchronous serial Communication has advantages of less transmission lines, high reliability and long transmission distance. UART allows full-duplex communication in serial link, thus has been widely used in the data communications and control system. It is widely used in data exchange between Processor and peripherals. UART converts data from parallel to serial at transmitter with some extra overhead bits using shift register and vice versa at receiver. To the processor the UART appears as an 8-bit read/write parallel port [1], [2]. Basic UART communication needs only two signal lines (Receive, Transmit) to complete full-duplex data communication [2]. UART includes three modules namely, the baud rate generator, receiver and transmitter. The baud rate generator is used to produce a local clock signal which is much higher than the baud rate to control the UART receive and transmit; The UART receiver module is used to receive the serial signals at RXD, and convert them into parallel data; The UART transmit module converts the bytes into serial bits according to the basic frame format and transmits those bits through TXD [3],[5]. When the transmitter is idle, the data line is in the high logic state. Otherwise when a word is given to the UART for asynchronous transmissions, "Start Bit" (logic low) is added to the beginning of each word that is to be transmitted. The Start Bit is used to alert the peripheral receiver that a word of data is about to be sent, and to force the clock in the receiver into synchronization with the clock in the transmitter [6],[4]. After the Start Bit, the individual data bits of the word are sent, with the Least Significant Bit (LSB) being sent first.

Manuscript published on 30 December 2014.
***Correspondence Author(s)**
Mr. Salman Khan B. R, is pursuing his Final year M.Tech. in VLSI and Embeded System at VTU Extension Center, UTL Technologies Ltd, Bangalore, India.
Mr. Arun Patro, is working as a Lecturer in Department of Electronics and Communication Engineering at VTU Extension Center, UTL Technologies Ltd, Bangalore, India.
Dr. Siva S. Yellampalli, is working as a Professor in Department of Electronics and Communication Engineering at VTU Extension Center, UTL Technologies Ltd, Bangalore, India.

Each bit is transmitted for exactly the same amount of time as all of the other bits, and the receiver samples at the wire at approximately halfway through the period assigned to each bit to determine if the bit is a 1 or a 0. When the entire data word has been sent, the transmitter adds a Parity Bit that the transmitter generates [3]. The Parity Bit may be used by the receiver to perform simple error checking. Then at least one Stop Bit is sent by the transmitter. When the receiver has received all of the bits in the frame, it automatically discards the Start, Parity and Stop bits. If another word is ready for transmission, the Start bit for the new word can be sent as soon as the Stop bit for the previous word has been sent. Asynchronous data are "self-synchronizing" if there are no data to transmit, the transmission line is held idle [7],[9]. In this paper, we present UART which includes three modules which are Transmitter, Receiver and Baud Rate Generator. The proposed design of UART satisfies the system requirements of high integration, stabilization, low bit error rate and low cost. It also supports configurable baud rate generator. In Proposed Architecture of UART compare to conventional UART we are adding line status register and Interrupt Logic in Proposed Architecture. The rest of paper is as follows: describe the proposed architecture of UART with Interrupt Logic: Flow chart of Transmitter, Receiver and Interrupt Logic. Simulation Results Transmitter, receiver, Interrupt Logic and Line Status Register using Xilinx ISE simulator and Modelsim.

II. PROPOSED ARCHITECTURE OF UART

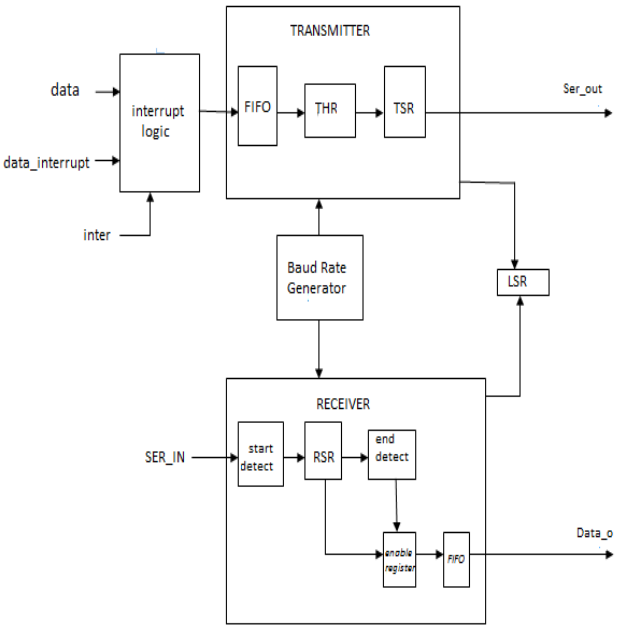

Figure 1: Proposed architecture of UART

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

A. *UART Transmitter:*

The Transmitter accepts the parallel data from the processor or peripherals and transmits the data in serial form on the output terminal of UART transmitter *ser_out*. Data is loaded from the interrupt logic block depending on the select line *inter* which will load the *data* or interrupt data into the transmitter. When fifo contains some data, if *tsr* is ready which is 9-bit register to receive data from the fifo then data is received. Now data is transmitted from tsr to *ser_out*. As Shown in figure1 the Transmitter block consists of FIFO, Transmitter hold register, Transmitter shift register .The flow of data in Transmitter is as shown in figure 2.

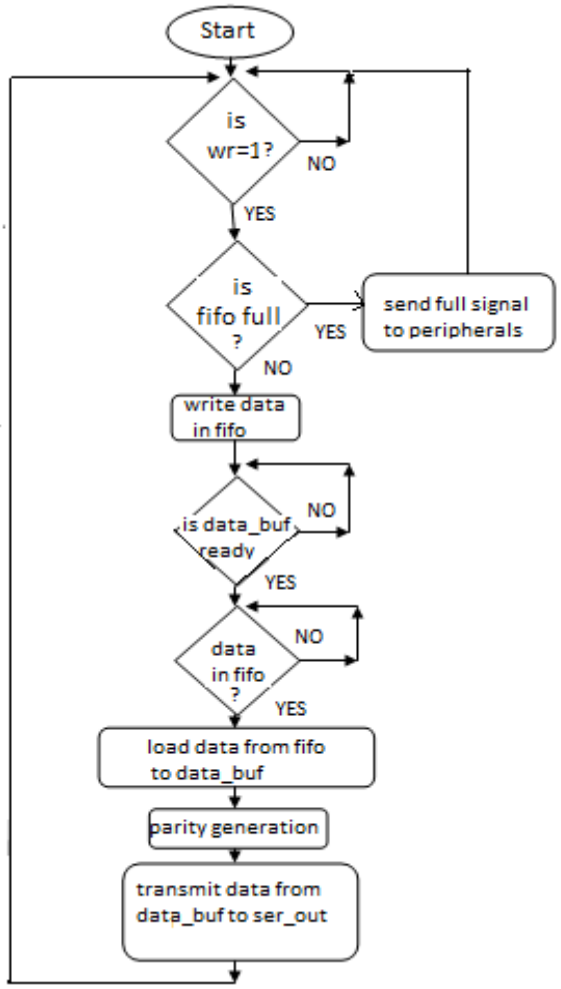

Figure 2: Flowchart of Transmitter

B. *UART RECEIVER:*

The Receiver takes the serial data which is available on *ser_in* pin and converts it into parallel data which is available on output of receiver. The received serial data is available on the *ser_in*. The data is transmitted serially in receiver through ser_in, when start bit detected the data is loaded in *rsr* which is 8-bit register. The data is then loaded in enable register which is 1-bit register and simultaneously end detector will detector will detect the end bit or stop bit, when stop bit detected it will stop the enable the register and the data is loaded in FIFO and FIFO output to *data_o*. The main function of receiver is to convert the serial data into parallel data. As shown in figure 1 receiver block consists of start bit detector, receiver shift register, end detector, Enable register and FIFO. The flow of data in receiver is as shown in figure 3.

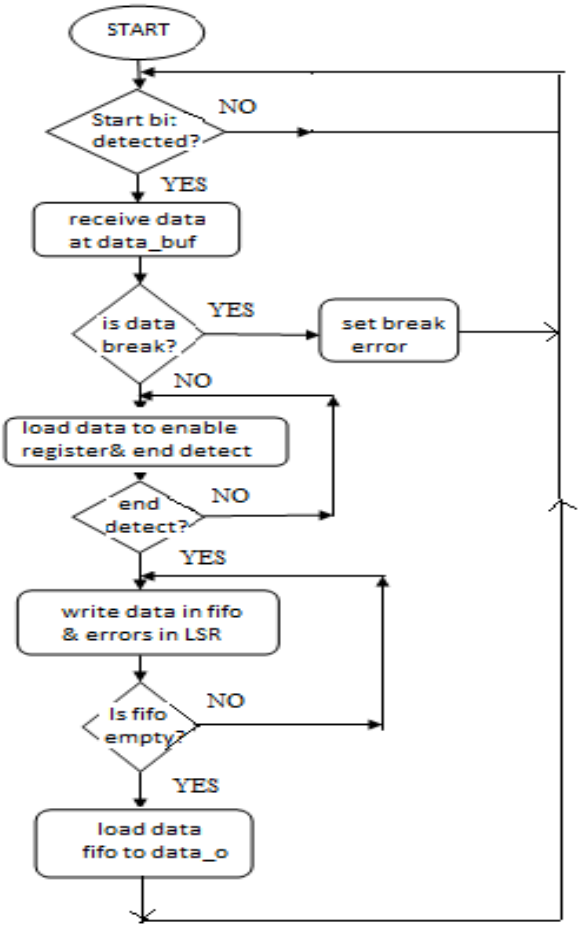

Figure 3: Flowchart of Receiver

C. *INTERRUPT LOGIC IN UART:*

Interrupt logic is used in UART as shown in figure 1. Interrupt logic block shown in figure, which consists of 2:1 mux in which 2-inputs, 1-output and one select line. *Data* and *data_interrupt* are the two inputs, inter is the select line and y as the output. Depending on the select line inter the *data_i* or data_interrupt is selected. On which the data is loaded in UART transmitter as shown in Fig1. When the *inter 0* is selected the data is loaded into the UART Transmitter, when inter 1 is selected the data_interrupt is loaded into the UART Transmitter. When inter 1 is selected the UART will stop the operation and UART is reset and *data-interrupt* is loaded to the UART. I*nter* is connected to the reset of UART Receiver, UART Transmitter and Baud rate generator. The data flow in Interrupt is shown in figure 5 flowchart of Interrupt Logic. Interrupt Logic is used in Proposed Architecture of UART is as shown in figure 1. When Interrupt occurs the UART will stop and reset .Then the Interrupt data flows in the Design.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

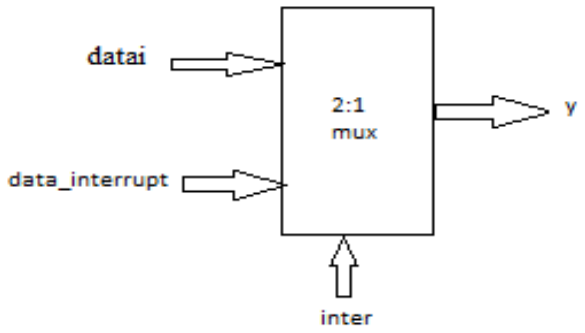

Figure 4: Interrupt logic

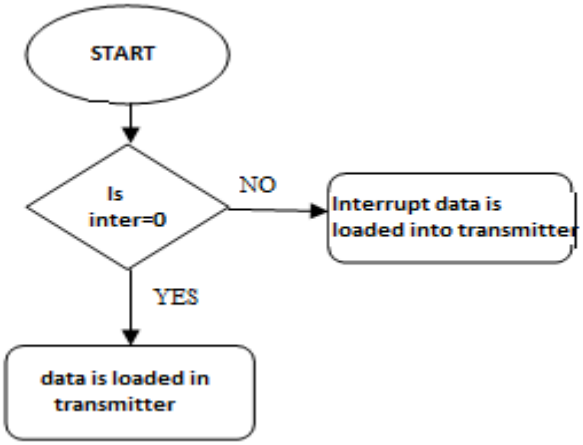

Figure 5: Flowchart of Interrupt logic

D. LINE STATUS REGISTER:

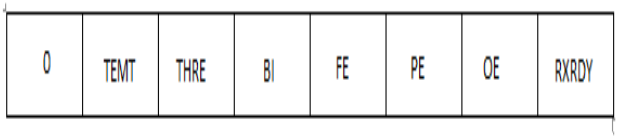

Figure 6: LSR FORMAT

RXRDY: RECEIVE DATA READY.
OE : OVERRUN ERROR.
PE : PARITY ERROR.
FE : FRAMING ERROR.
BI : BREAK INTERRUPT.
THRE : TRANSMITTER HOLDING REGISTER EMPTY.
TEMT : TRANSMITTER EMPTY.

E. BAUD RATE GENERATOR:

The Baud Rate Generator is used to produce a local clock signal which is much higher than the baud rate to control the UART receiver and transmit. The baud rate generator is actually a frequency divider. The frequency factor is calculated according to the given system clock frequency and requested baud rate. Assume that the system clock is 100MHZ, baud rate is 9600bps.Therefore the frequency coefficient (M) of baud rate generator is:

M=100*10^6/16*9600Hz=651- - - - - - - - - - - -[1]

frequency coefficient=M
system clock=100MHZ
Baud Rate= 9600bps

III. SIMULATION RESULTS

The verilog HDL coding and simulation of the design are done in Xilinx tool Modelsim package XC3S250E TQ144-5. The operating clock frequency used for simulation is 100 MHz. The baud rate set is 9600bps. Data word length is 8-bits.

A. Simulation results of Transmitter

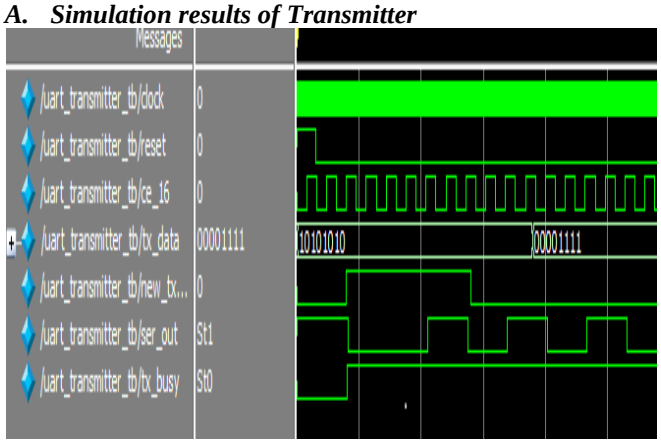

Figure 7: Simulation result of UART Transmitter

As shown in figure 7 simulation result of transmitter in which 10101010 is transmitter input tx_data. Clock, reset, ce_16 which is baud clock, new_tx_data are the inputs of transmitter and ser_out, tx_busy are the outputs of receiver. 10101010 is given to input tx_data when new_tx_data is high then only input data is transmitted on ser_out as shown in Fig 7 and tx_busy will show high because data is transmitting.

B. Simulation results of Receiver

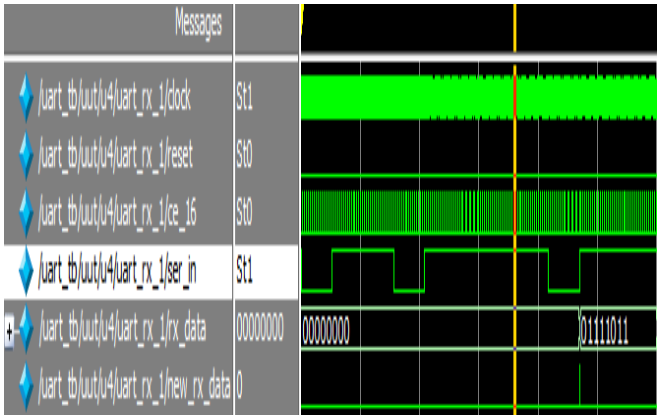

Figure 8: Simulation results of Receiver

As shown in figure 8 simulation results of Receiver in which clock, reset, ce_16, ser_in are the inputs and rx_data, new_tx_data are the outputs of receiver. The input data from the ser_out of transmitter is available on ser_in 01111011 and displayed on output of receiver rx_data and after receiving all bits new_rx_data will become high indicating new data is coming. Receiver block consists of clock, reset, ser_in are the inputs and rx_data, new_rx_data are the outputs of the receiver .When reset is high the output of receiver rx_data which 8-bit output is zero.

C. Simulation results of Interrupt logic

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

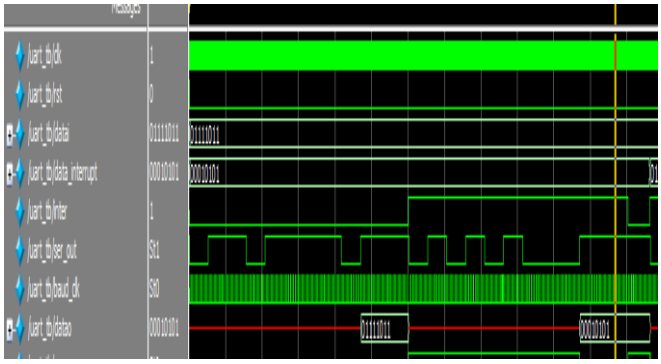

Figure 9: Simulation results of Interrupt logic

As shown in figure 9 simulation results of Interrupt logic in which datai is 8-bit data and data_interrupt is 8-bit interrupt data. The datai which is 01111011data is transmitting. inter which is 1-bit input, the inter value changes from 0 to 1 then the interrupt data which is 00010101 is start to transmit .As shown in figure 9 the datai and data_interrupt value is at the output of data0 which 8-bit output.

D. Simulation results of Line Status Register

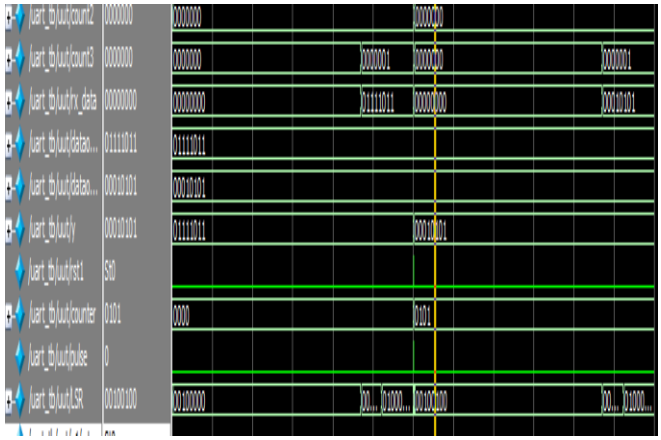

Figure 9: Simulation results of Line Status Register

As shown in figure 9 Simulation results of Line Status Register in which 00100000 the third bit indicates that THRE is high which means UART is ready to accept a new character for transmission. 01000000 the second bit is high indicates TEMT is empty.00010000 the fourth bit is high indicates BI there is an interrupt occur during transmission of data. 00000100 sixth bit is high indicates invalid parity bit is encountered.

E. Synthesis Report

As shown in figure10 show design implementation summary of proposed architecture from Spartan 3e xc3s400.

Design Utilization Summary	
Selected device	3s400pq208-5
Number of Slice	20 out of 3584 43%
Number of Slice Flip Flops	28 out of 7168 23%
Number of 4-input LUTs	20 out of 7168 39%
Number of IOs	13
Number of bonded IOBs	13 out of 141 24%
Number of Gclks	1 out of 8 12%
Timing summary	
Speed Grade	-5
Minimum period	4.826ns (Max frequency: 207.2 MHZ)
Minimum input arrival time befor clock	5.093ns
Maximum output required time after clock	6.216ns
Maximum Combinational path delay	No path found

Figure 10: Synthesis Report

IV. CONCLUSIONS

Architecture of UART that support 8-bit data word length at 9600 bps baud rate for serial transmission of data with the addition of interrupt logic and line status register for detecting errors in data transfer is introduced. Working of UART has been verified using Xilinx ISE simulator. With error checking status register, we can detect the different types of errors occurred during on-chip communication.

REFERENCES

1. Fang Yi-yuan and Chen Xue-jun, "Design and Simu lation of UART Serial Communication Module Based on VHDL", *in the proceedings of 3rd International Workshop on Intelligent Systems and Applications (ISA), IEEE, May 2011,* DOI: 10.1109/ISA.2011.5873448, pp.1-4.
2. Naresh, Vatsalkumar and Vikaskumar Patel, "VHDL Implementation of UART with Status Register", *in the proceedings of International Conference on Communication Systems and Network Technologies, IEEE Computer Society, 11-13th May 2012,* DOI: 10.1109/CSNT.2012.164, pp.750-754.
3. Dr. Garima Bandhawarkar Wakhle, Iti Aggarwal and Shweta Gaba, "Synthesis and Implementation of UART using VHDL Co des", *in the proceedings of International Symposium on Computer, Consumer and Control, IEEE June 2012,* DOI: 10.1109/IS3C.2012.10.
4. Mohd Yamani Idna Idris, Mashkuri Yaacob and Zaidi Razak, "A VHDL Implementation of UART Design with BIST Capability", *in the proceedings of Malaysian Journal of Computer Science, June 2006,* Vol. 19(1), pp. 73-86.
5. Norhuzaimin J and Maimun H.H, "The design of hi gh speed UART", *in the proceedings of Asia-Pacific Conference on Applied Electromagnetics, APACE 05, IEEE, 20-21st Dec. 2005,* DOI: 10.1109/APACE.2005.1607831, pp.5-8.
6. *Chun-zhi, He; Yin-shui, Xia; Lun-yao, Wang; , "A universal asynchronous receiver transmitter design,"Electronics, Communications and Control (ICECC), International Conference on , vol., no., pp.691-694, 9-11 Sept. 2011.*
7. Mahat N.F, "Design of a 9-bit UART module based on Verilog HDL", *in the proceedings of 10th IEEE International Conference on Semiconductor Electronics (ICSE), 19-21st Sept. 2012,* DOI: 10.1109/SMElec.2012.6417210, pp. 570-573.
8. *Gallo, R.; Delvai, M.; Elmenreich, W.; Steininger, A.; , "Revision and verification of an enhanced UART," Factory Communication Systems, 2004. Proceedings. 2008 IEEE International Workshop on ,vol., no., pp. 315- 318, 22 Sept.2008.*
9. *Idris, M.Y.I.; Yaacob, M.; , "A VHDL implementation of BIST technique in UART design," TENCON 2003. Conference on Convergent Technologies for Pacific Region , vol.4, no., pp. 1450-1454 Vol.4, 15-17 Oct. 2003.*

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Mr. Salman Khan B. R, is pursuing his final year M.Tech. in VLSI and Embeded System at VTU Extension center, UTL Technologies Ltd, Bangalore.

Mr. Arun Patro, is working as a lecturer in Dept. of Eelectronics and communication engineering at VTU Extension center, UTL Technologies Ltd, Bangalore.

Dr. Siva Yellampalli, is working as a professor in Dept. of Electronics and communication engineering at VTU Extension center, UTL Technologies Ltd, Bangalore. His research interest includes Design of RF communication system.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)