Vol 05, Article 07380 July 2014
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556

VHDL IMPLEMENTATION OF UART WITH ADAPTIVE BAUD RATE GENERATOR

DEEPTI AGARWAL[1], NIDHI SHARMA[2], CHIRANJEET KUMAR[3]
[1]Department of Electronics of Communication, NIET, Knowledge Park, Greater Noida, U.P., INDIA
[2] Department of Electronics of Communication, NIET, Knowledge Park, Greater Noida, U.P., INDIA
[1]erdeepti.ag@gmail.com, [2]sonikaushi07@gmail.com, [3]chiranjeet9650530563@gmail.com

ABSTRACT

This paper concentrates on adaptation of UART module with the baud rate sent by the transmitter. Receiver module automatically generates a system clock according to the baud rate sent by the transmitter. Each separate module has been constructed by the using VHDL language and is simulated using modelsim 6.5c. The hardware implementation is done on FPGA Spartan 6 XC6SLS9 using Xilinx 14.1i. Baud rate generation is according to an encrypted data as shown in the simulation result. Encrypted information of baud rate is sent by the transmitter at a default baud rate. All design units are properly synthesized and are implemented as complete module on FPGA. Simulated waveform generates the baud rate at system frequency of 50 MHz. Range of baud rate adapted is from 300 to 38462. The reliability of the design can be observed by the simulated result shown in this paper.

Keywords: *UART, adaptive baud rate generator, sampling rate, Xilinx Spartan14.1,Spartan 6 XC6SLS9.*

[1] INTRODUCTION

Most of the MCS unit or microprocessor unit contains Universal Asynchronous receiving and transmitting(UART) module is integrated inside it. Serial communication is very important communication protocol as it makes possible communication for the greatest distance with least distortion in the channel. This communication protocol is mostly adapted all over the world for its own advantages. Serial communication is used in SoC systems for transfer of an 8 bit data with specific baud rate by the transmitter set at the time of transmission. Receiver on the other hand receives the same data with the same baud rate on which transmitter sent it. In recent years research has been made on various UART designs like automatic baud rate synchronizing capability, predictable timing behavior to allow integration of nodes with imprecise clocks in time triggered real time systems, recursive running sum filter to remove noisy samples.

Implementation of basic UART requires only two signals RxD and TxD pins for reception and transmission of data respectively. UART supports full duplex mode of communication because of separate channel for transmission and reception. TxD remains in idle state when there is no data to send. In idle state the TxD remains at logic high. A data frame is constructed including the first bit at logic 0 indicating the start bit for the data frame and then 8 bit for the data and at last stop bit is added to give the command that data frame has finished when control invokes the transmitter to transmit a data asynchronously. The advancement in this paper is whenever there is command to send a data, transmitter will first send an encrypted data at a default baud rate of 9600 containing the information of the baud rate at which the actual data would be transmitted. Transmitter automatically configures clock generator at a 9600 baud rate for the encrypted data and then generates the actual baud rate at which the main data will be transmitted.
At the receiver end RxD remains in idle state at logic 1 and whenever there is start bit at this pin the receiver configures to receive the encrypted information at default baud rate 9600. The encrypted data is decoded and fed to clock generator to generate the baud rate at which the main data is sent and the next all continuous data will be received at this baud rate in which the receiver is configured. The benefit of this technique is that it is not necessary to configure both receiver and transmitter at same baud rate. The receiver is smart enough to be configured at baud rate at which the transmitter has sent the data. Data frame for universal asynchronous receiving and transmitting is given in fig. 1.

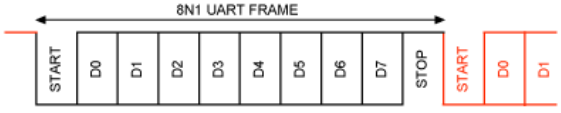


Fig . 1. Data Frame

This paper mainly focus on the three block of the design.
1. Transmitter.
2. Automatic Baud rate generator.
3. Receiver.

Vol 05, Article 07380 July 2014
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556

This UART design fulfils the probability of low bit error rate, high integration and least cost. This design has capability of adaptation of receiver at a baud rate at which transmitter has sent the data. The first byte keeps the information of baud rate at which the receiver is going to be configured and next each frame contains bytes of data in the standard format shown in Fig. 1.
The configuration of data is done only at the transmitter end by providing baud rate information in the form of encrypted input.
This complete design contains three main modules in it and that is Transmitter, Adaptive baud rate generator and Receiver. The transmission and reception will contain 8 bit data format. The baud rate generation is done with the help of three manual inputs to implement it in hardware environment and for manual inputs DIP switches are used. The data transmission is shown using LED's connected with Xilinx Spartan 6. For functional simulation model sim 10.1b is used.

2. HARDWARE IMPLEMENTATION

All the individual units are constructed using VHDL (very high speed integrated circuit hardware description language) and are mutually connected using structural modeling style of VHDL with port mapping technique. The complete design is synthesized using Xilinx 14.1i ISE design suite. After complete synthesis of design it is simulated on model sim 10.1b. hardware environment is created using FPGA(field programmable gate array) of Xilinx Spartan 6. This FPGA device is embedded into a kit named Spartan 6 EDK. Spartan 6 kit contains many interfacing modules such as 16 switches to enter large no. inputs for a design, 16 LED's to capture large no. of outputs of a design, PS2 interfacing, RTC DS1307 working on I2C protocol, Ethernet module etc. FPGA is chosen for hardware implementation as FPGA are suitable for high working frequency of the design.

3. UART COMPLETE DESIGN

The UART module contain three basic units Transmitter ,Adaptive baud rate generator and Receiver as shown in Fig 2. As there is three different modules so complete design is implemented by implementation of individual modules. Transmitter is responsible for converting parallel data into serial data and send it serially through TxD pin of the UART. Baud rate generator is working on much higher frequency and is responsible for generating the desired baud rate clock frequency. The available baud rate with this system is listed in a table. And the receiver module is attached to convert the serial data into parallel data.

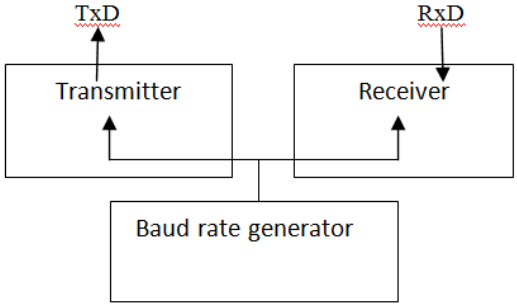


Fig 2. UART components

3.1 Adaptive Baud Rate Generator

Eight different baud rate are available for transmitter and receiver. On the system clock frequency of 50 MHz, clock is first divided by 81 and then it is passed to configurable radix 2 divider circuit. This divided clock frequency is further divided by 2,4,8,16,32,64,128 and 256 to attain the eight different baud rate of the UART system. Table 1**.** describes the available baud rate for the system.

Table 1. Baud Rate Table

Control bit values	Baud rates
000	38400
001	19200
010	9600
011	4800
100	2400
101	1200
110	600
111	300

The values of control bits selects one the available baud rate available in the system. The following circuit given in Fig 4. Explains the attainment of the available baud rate for transmission and reception.

Vol 05, Article 07380 July 2014
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556

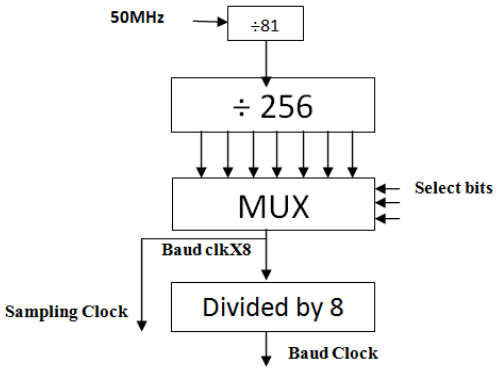


Fig 4. Baud Clock Generator Circuit

3.2 TRANSMITTER

The function of transmitter module is to transmit the encrypted information first, about the baud rate at which main 8 bit data is going to be transmitted and then the actual 8 bit data is transmitted. The important point is the encrypted information is transmitted using the same frame construction as of the main data but at a default baud rate of 9600. In the transmitter module there is a three bit input to select one of the available eight baud rate mentioned in the table given in Fig 3. Transmitter will be sending each bit at each positive clock of baud rate generator which is configured according to the input at three bits for selection of baud rate after the transmission of encrypted information. Transmitter is constructed using Finite State Machine. The ASM diagram is given in Fig 4. Transmitter is implemented using 4 states.

3.2.1 IDLE state: The system waits for data frame to be loaded in a register for transmission and switch to next state BAUDSEND.

3.2.2 BAUDSEND state: When the system is loaded with the data for transmission transmitter first send a code in the form of information of baud rate at which data is going to be transmitted and this code is sent by the transmitter at a default baud rate of 9600 and switched to the next state SYNCH state.

3.2.3 SYNCH state: After the information has been sent, the loaded data is the loaded to shift register for transmitting it serially and simultaneously the 3 bit input for baud rate selection is passed to baud rate generator for generating new transmitting baud rate and transmitter switched to TDATA state.

3.2.4 TDATA: In this state finally all the bits in shift register is transmitted according to the baud rate generated and maintaining the data frame. A sample of state diagram is given in Fig 5.

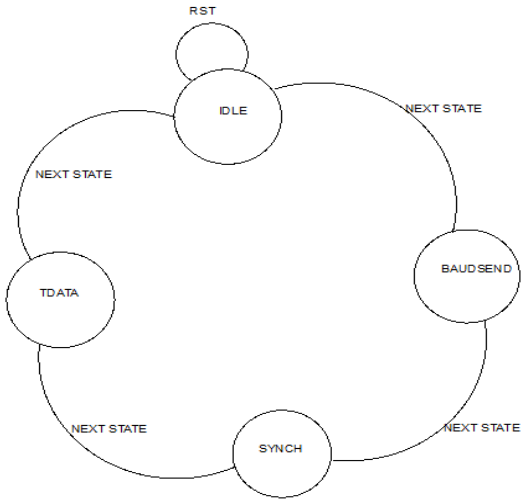


Fig 5. Transmitter FSM, state transition in clockwise direction.

3.3 RECEIVER

Receiver part of UART is implemented using 4 states.

3.3.1 IDLE state: Receiver waits in the idle state until the start bit of data has been detected. Once the start bit is detected the receiver is switched to baud_detect state.

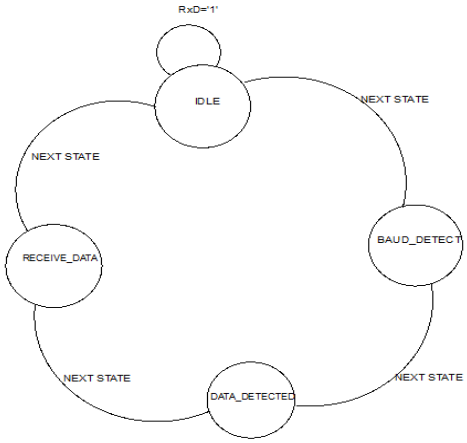
3.3.2BAUD_DETECT state: In the baud_detect state each bit of RxD pin is sampled 8 times and collected at the middle of the sample count, for accurate reception, meanwhile baud rate generator has been set in default to generate baud rate 9600. Two counter is used, one for sampling each bit and other for counting the no. of bits

Vol 05, Article 07380 July 2014
http://ijves.com
International Journal of VLSI and Embedded Systems-IJVES
ISSN: 2249 – 6556

received. Once all the bits are received at default baud rate of 9600, the received code is decoded to find the baud rate at which the next byte will be received and counters are reset. After the first encrypted data is received one flag signal is made high to indicate the baud rate information has been received and the next state is idle state again to check again for the start bit of data. Now once the start bit is detected state is switched to data_detected state.

3.3.3DATA_DETECTED state: In this state the decoded data is passed to baud rate generator unit to generate the baud clock according to the sent information and switched to receive_data state.

3.3.4 RECEIVE_DATA state: In this state each bits of data is sampled and stored in the shift register and passed to the next unit of UART. The reception of data is done in the same manner as it is done for the data received to detect the baud. A sample of state diagram is given in **Fig 6.**



4. SIMULATION RESULTS

Results are implemented using modelsim 6.5c. Simulation results shows that an encrypted data x"41" is sent first at a default baud rate of 9600 and then the next data is transmitted according to the baud rate set by the user by configuring the of the transmitter. Simulation waveform is shown in Fig 7.

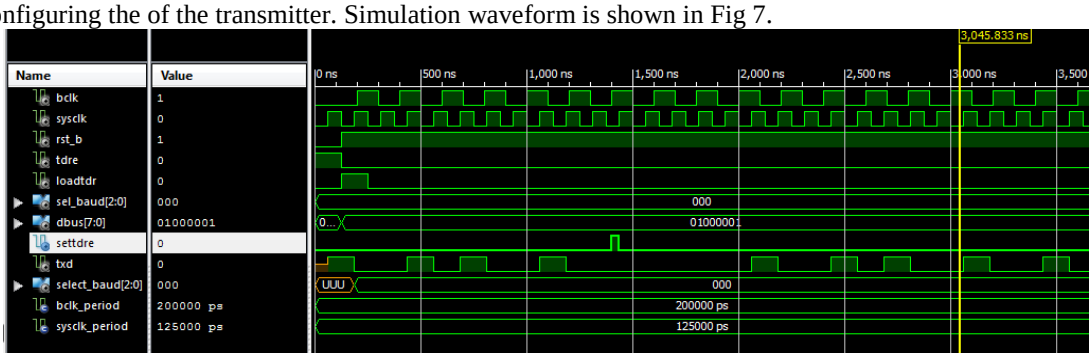


Fig 7.Output waveform on modelsim 6.5c

RTL is generated using Xilinx 14.1i and is shown in Fig 8. and Fig 9.

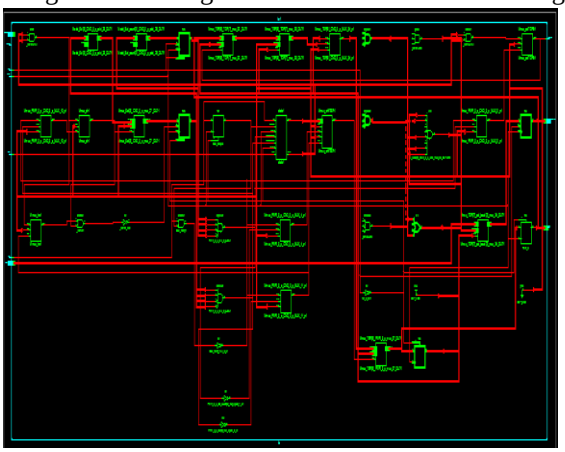


Fig 8. RTL Schematic

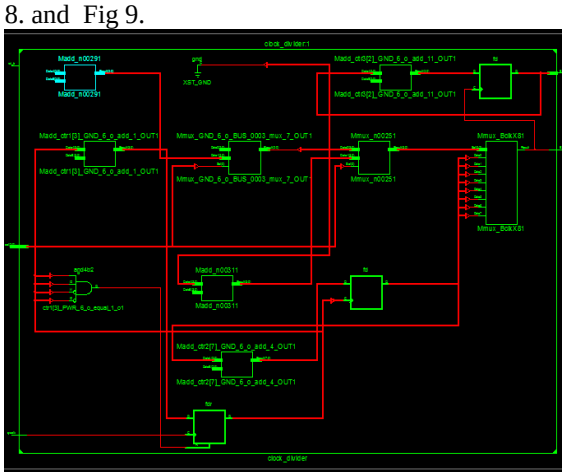


Fig 9. The adaptive clock generator block

Vol 05, Article 07380 July 2014 International Journal of VLSI and Embedded Systems-IJVES
http://ijves.com ISSN: 2249 – 6556

5. RESULTS

Working of design is verified by comport communication using the DB9 port male and female at the FPGA device end and PC end respectively. Hercules software is used to send and receive a byte of information in the form of ASCII character. UART module is completely able to configure itself according to the baud rate sent by the transmitter.

6. CONCLUSIONS

The FPGA implementation of UART with adaptive baud rate generator on Xilinx Spartan 6 XC6SLS9 with Xilinx design suite 14.1i is presented in this paper. The simulation results through waveform and FPGA implementation shows that the design is working very fine on the adaptive nature of UART. The adaptive nature of design leads to put this concept of UART in every SoC unit of microprocessor and microcontroller for making the respective unit to be more reliable on communication protocol.

ACKNOWLEDGEMENTS

We would like to acknowledge the Faculties of Electronics and Communication department, Noida institute of technology, Greater Noida for their valuable support. I Deepti Agarwal specially thank to my guide Professor Mrs. Nidhi Sharma mam for her precious guidance and encouragement towards my paper work.

REFERENCES

[1] Fang Yi-yuan; Chen Xue-jun; , "Design and Simulation of UART Serial Communication Module Based on VHDL," Intelligent Systems and Applications (ISA), 2011 3rd International Workshop on , vol., no., pp.1-4, 28-29 May 2011.
[2] Himanshu Patel; Sanjay Trivedi; R. Neelkanthan; V. R. Gujraty; , "A Robust UART Architecture Based on Recursive Running Sum Filter for Better Noise Performance," VLSI Design, 2007. Held jointly with 6th International Conference on Embedded Systems., 20th International Conference on , vol., no., pp.819-823, Jan. 2007.
[3] Gallo, R.; Delvai, M.; Elmenreich, W.; Steininger, A.; , "Revision and verification of an enhanced UART," Factory Communication Systems, 2004. Proceedings. 2004 IEEE International Workshop on , vol., no., pp. 315-318, 22-24 Sept. 2004.
[4] Elmenreich, W.; Delvai, M.; , "Time-triggered communication with UARTs," Factory Communication Systems, 2002. 4th IEEE International Workshop on , vol., no., pp. 97- 104, 2002J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2.Oxford: Clarendon, 1892, pp.68–73.
[5] Zhichao Zhang, Wuchen Wu (2010) 2*nd International Confer-ence on Computer Engineering and Technology*. 2(7), 331 – 334.
[6] Shouqian Yu Lili Yi Weihai Chen Zhaojin Wen (2007) *IEEE Conference on Industrial Electronics and Applications*.
[7] Raffaele Gallo, Martin Delvai, Wilfried Elmenreich, Andreas Steininger (2004) *Revision and Verification of an Enhanced UART IEEE*.
[8] Donald E. Thomas & Philip R. Moorby (2002) *The Verilog Hardware Description Language, Fifth Edition*.
[9] Charles. H. Roth, Jr. Digital System Design using VHDL(1998).
[10] Pong. P. Chu FPGA prototyping with VHDL(2007). John Wiley and sons. Inc. Publication.