RESEARCH ARTICLE OPEN ACCESS

Achieving Reduced Area and Power with Multi Bit Flip-Flop When Implemented In UART with Status Register

P.Rajeepriyanka[1], S R Sastry Kalavakolanu[2]
[1](Department of E.C.E, Andhra Loyola Institute of Engineering and Technology, Vijayawada, India)
[2](Assistant Professor, Department of E.C.E, Andhra Loyola Institute of Engineering and Technology, Vijayawada, India)

ABSTRACT
A UART (Universal Asynchronous Receiver and Transmitter) is a device allowing the reception and transmission of information, in a serial and asynchronous way. This project focuses on the implementation of UART with status register using multi bit flip-flop and comparing it with UART with status register using single bit flip-flops. During the reception of data, status register indicates parity error, framing error, overrun error and break error. The multi bit flip-flop is indicated in this status register. In modern very large scale integrated circuits, Power reduction and area reduction has become a vital design goal for sophisticated design applications. So in this project the power consumed and area occupied by both multi-bit flip-flop and single bit flip is compared. The underlying idea behind multi-bit flip-flop method is to eliminate total inverter number by sharing the inverters in the flip-flops. Based on the elimination feature of redundant inverters in merging single bit flip-flops into multi bit flip-flops, gives reduction of wired length and this result in reduction of power consumption and area.

Keywords- Clock buffer, Clock network, Multi bit flip-flop, status Register, Single bit flip-flop.

I. INTRODUCTION

Universal Asynchronous Receiver Transmitter (UART) is a kind of serial communication protocol. UARTs are used for asynchronous serial data communication by converting data from parallel to serial at transmitter with some extra overhead bits using shift register and vice versa at receiver.

Optimizations in VLSI have been done on three factors: Area, Power and Timing (Speed). The implementation of multi-bit flip-flop is an effective method for clock power consumption reduction. By replacing Flip-Flops with multi-bit Flip-Flops power consumption can be reduced. By using single clock pulse the Multi-bit Flip-flop (MBFF) is designed so that the same functionality like two or more single-bit Flip-flop (SBFF) can be achieved. So in this project multi bit flip- flop is implemented in status register of UART. The timing performance of MBFF can be analyzed by simulating in Xilinx. As a result the Clock network such as clock buffer and gate delay can be reduced. So the total area used for designing and power consumption is also reduced.

II. UART CONCEPT

UART is a Universal Asynchronous Receiver-Transmitter, which is used to communicate between two devices. Most computers and microcontrollers include one or more serial data ports utilize to communicate with other serial I/O devices, such as keyboards and serial printers. Serial ports are also used to communicate between two computers using a UART in each computer and a crossover cable, which connects the transmitter of one UART to the receiver of the other, and vice versa. Serial communication uses a transmitter to send data, one bit at a time, over a single communication line to a receiver. You can use this method when data transfer rates are low or you must transfer data over long distances. Serial communication is popular because most computers have one or more serial ports, so no extra hardware is needed other than a cable to connect the instrument to the computer or two computers together.

A UART provides the means to send information using a minimum number of wires. The data is sent bit serially, without a clock signal. The main function of a UART is the conversion of parallel-to-serial when transmitting and serial to- parallel when receiving. The fact that a clock signal is not sent with the data complicates the design of a UART. The two systems (transmitter and receiver) contain separate and unsynchronized local clocks. The proposed design of UART, shown in Fig. 2, has LCR, Transmitter and Receiver as its functional units. All these blocks are explained in brief as course of rest of this section.

2.1 Line Control Register (LCR)

The line control register (LCR) is a byte register. It is used for precise specification of frame format and desired baud rate. The parity bits, stop bits, baud

rate selection and word length can be changed by writing the appropriate bits in LCR.

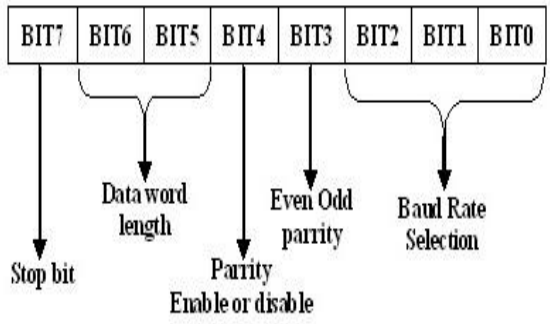

Figure 1: LCR format

2.2 UART Transmitter

The transmitter section accepts parallel data, makes the frame of the data and transmits the data in serial form on the Transmitter Output (TXOUT) terminal. Data is loaded from the inputs TXIN0-TXIN7 into the Transmitter FIFO by applying logic high on the WR (Write) input. If words less than 8 bits are used, only the least significant bits are transmitted. FIFO is 16-byte register. When FIFO contains some data, it will send the signal to Transmitter Hold Register (THR), which is an 8-bit register. At a same time, if THR is empty it will send the signal to FIFO, which indicates that THR is ready to receive data from FIFO. If Transmitter Shift Register (TSR) is empty it will send the signal to THR and it indicates that TSR is ready to receive data from THR. TSR is a 12-bit register in which framing process occurs. In frame start bit, stop bit and parity bit will be added. Now data is transmitted from TSR to TXOUT serially.

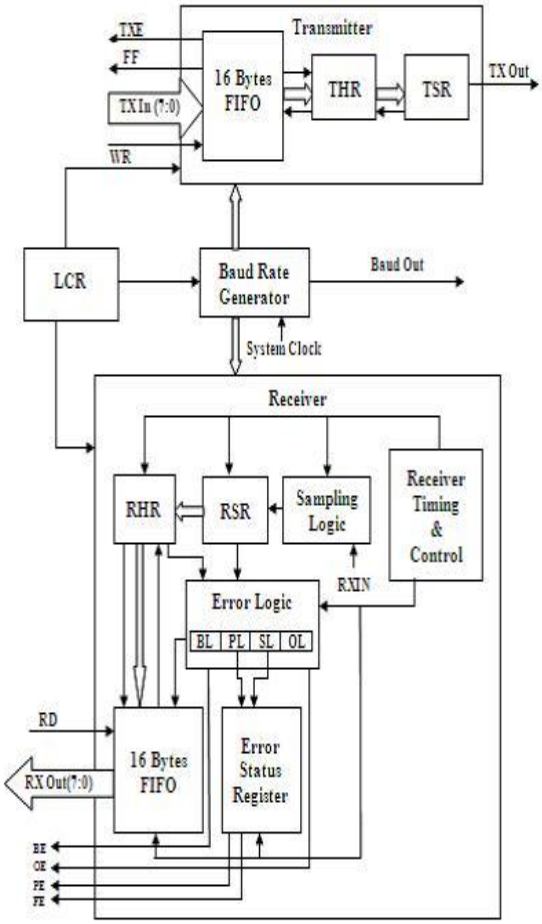

figure 2: UART Architecture

2.3 UART Receiver

The transmitted data from the TXOUT pin is available on the RXIN pin. The received data is applied to the sampling logic block. The receiver timing and control is used for synchronization of clock signal between transmitter and receiver. Initially the logic line is high whenever it goes low sampling and logic block will take 4 samples of that bit and if all four are same it indicates the start of a frame. After that remaining bits are sampled in the same way and all the bits are send to Receiver Shift Register (RSR) one by one where the entire frame is stored. RSR is a 12 bit shift register. Now if the Receiver Hold Register (RHR) is empty it sends signal to RSR so that only the data bits from RSR goes to RHR which is an 8 bit register. The remaining bits in the RSR are used by the error logic block. Now if receiver FIFO is empty it send the signal to RHR so that the data bits goes to FIFO. When RD signal is asserted the data is available in parallel form on the RXOUT0-RXOUT7pins. The error logic block handles 4 types of errors: Parity error, Frame error, Overrun error, break error. If the received parity does not match with the parity

generated from data bits PL bit will be set which indicates that parity error occurred. If receiver fails to detect correct stop bit or when 4 samples do not match frame error occurs and SL bit is set. If the receiver FIFO is full and other data arrives at the RHR overrun error occurs and OL bit is set. If the RXIN pin is held low for long time than the frame time then there is a break in received data and break error occurs and BL bit is set.

2.4 The UART Standard Data Format

Serial data are contained within frames of 8 data bits, as well as coded information bits. Between successive transmissions, the transmission line is held high. A transmission is initialized by a leading low start bit. Next to the leading low start bit comes 8 bits of data information, beginning with the LSB and afterwards represented at increasing significance order up to the MSB. Next to the 8 data bits comes the parity bit, representing the parity result of the 8 data bits. The parity bit can be encoded true based on even parity or odd parity mode. Next to the parity bit comes a trailing high stop bit indicating the end of a data frame.

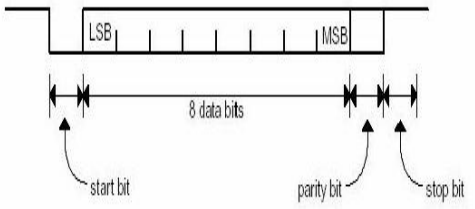

Figure 3: UART data format

III. MULTI BIT FLIP- FLOP CONCEPT

In this section, we will introduce multi-bit flip-flop concept. The proposed method for multi bit flip flop is merging of clock pulse. Before that, we will review single-bit flip-flop. Figure 3 shows an example of single-bit flip-flop. A single-bit flip-flop has two latches (Master latch and slave latch). The latches need "Clk" and "Clk' " signal to perform operations, such as Fig.4 shows.

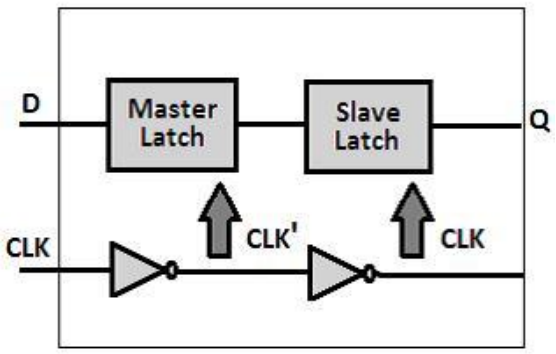

Figure 4: Single-Bit Flip-Flop

In order to have better delay from Clk-> Q, we will regenerate "Clk" from "Clk'". Hence we will have two inverters in the clock path. Fig.5 shows an example of merging two 1-bit flip-flops into one 2-bit flip-flop. Each 1-bit flip-flop contains two inverters, master-latch and slave-latch.

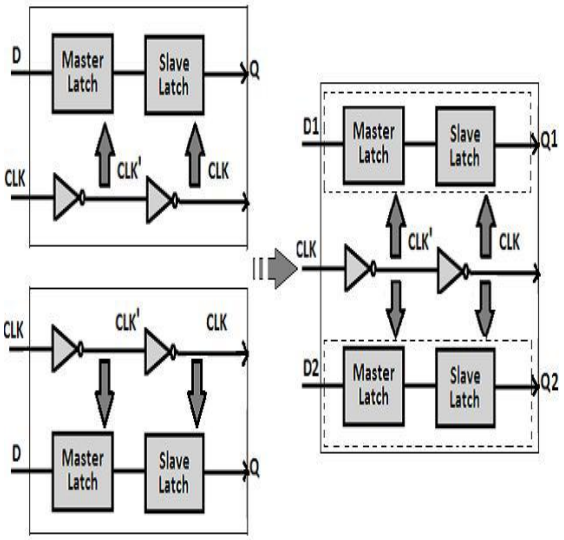

Figure 5: An example of merging two 1-bit flip-flops into one 2-bit flip-flop.

Due to the manufacturing rules, inverters in flip-flops tend to be oversized. As the process technology advances into smaller geometry nodes like 65nm and beyond, the minimum size of clock drivers can drive more than one flip-flop. Merging single-bit flip-flops into one multi-bit flip-flop can avoid duplicate inverters, and lower the total clock dynamic power consumption.

IV. EXPERIMENTAL RESULTS

By implementing the multi bit flip-flop in UART with status register, it is simulated in Xilinx. The total power consumed by clock in it is 0.00263W whereas the power consumed by UART with status register using single bit flip-flop is 0.00318W. Atleast 20% of power is reduced when it is compared with normal UART with status register. The simulation results for power is shown in Fig.6 and Fig.8

In synthesis report, UART with status register using single bit flip-flop uses four clock buffers whereas in UART with status register using multi bit flip-flop, we can see only two clock buffers are used. This indicates that the area is reduced. The Synthesis report for both single bit and multi bit flip-flop is shown in Fig.7 and Fig 9.

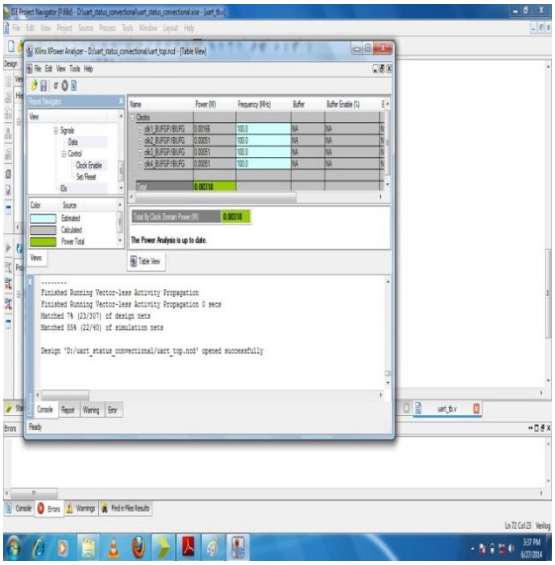

Figure 6: power report using single bit flip-flop

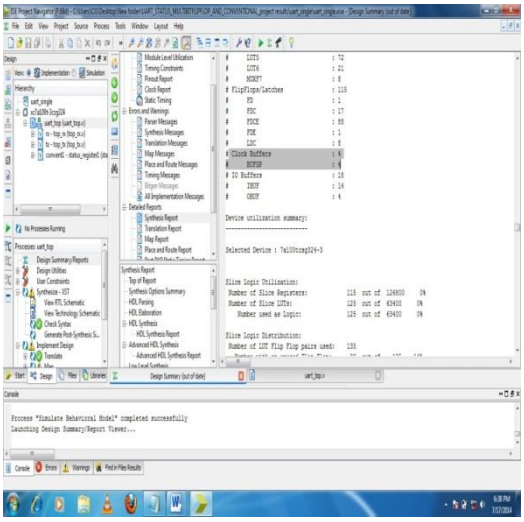

Figure 7: synthesis report for single bit flip-flop

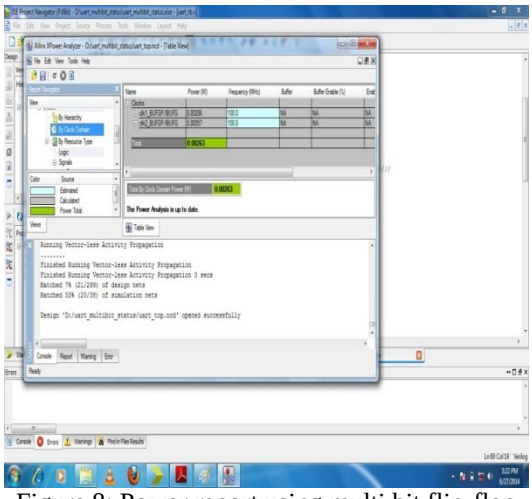

Figure 8: Power report using multi bit flip-flop

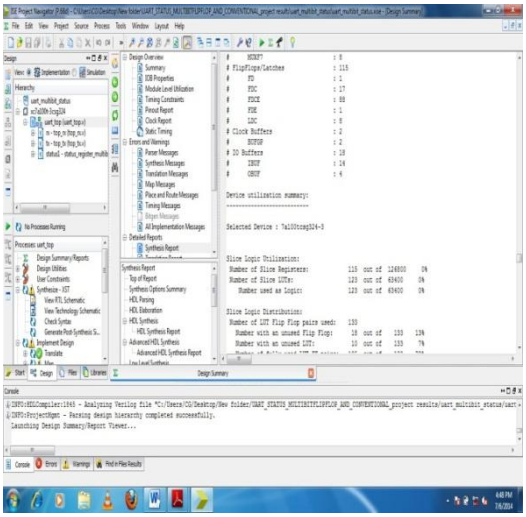

Figure 9: Synthesis report for multi bit flip-flop

The number of clocks used and power consumed by using single bit flip-flop and multi bit flip-flop is indicated in graphical representation. Those graphs are shown in Fig.10 and Fig.11

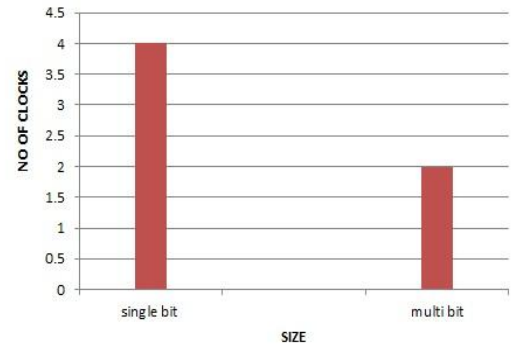

Figure 10: clock buffers

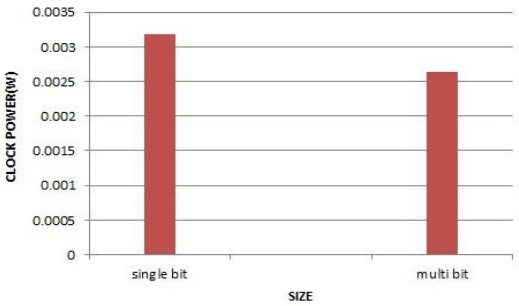

Figure 11: power consumed

V. CONCLUSION

In present VLSI design area is one of the important issues to be addressed. The proposed method is implemented in Xilinx. Experimental results are targeted to number of clock buffer usage and power used by clock buffer. Thus this proposed method is more suitable for reduction of hardware.

REFERENCES

[1] Zhi-Wei Chen and Jin-Tai Yan, *Routability-Driven Flip-Flop Merging Process for Clock Power Reduction*, Computer Design (ICCD) IEEE International Conference, 2010

[2] Ya-Ting Shyuet. Al., *Effective and Efficient Approach for Power Reduction by Using Muti-Bit Flip-Flops*, IEEE transactions on very large scale integration systems, 2012

[3] Jin-Tai Yan and Zhi-Wei Chen, *Construction of Constrained Multi-Bit Flip-Flops for Clock Power Reduction, Green Circuits and Systems* (ICGCS) International Conference, 2010

[4] Chih-Cheng Hsu, Yao-Tsung Chang and Mark Po-Hung Lin, *Crosstalk-Aware Power Optimization with Multi-Bit Flip-Flops*, 17th Asia and South Pacific Design Automation Conference, 2012

[5] 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.

[6] 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

[7] Idris, M.Y.I.; Yaacob, M.;, "*A VHDL implementation of BIST technique in UART design*," TENCON 2003. Conference on Convergent Technologies for Asia-Pacific Region , vol.4, no., pp. 1450- 1454 Vol.4, 15-17 Oct. 2003.

[8] Chun-zhi, He; Yin-shui, Xia; Lun-yao, Wang; , "*A universal asynchronous receiver transmitter design*," Electronics, Communications and Control (ICECC), 2011 International Conference on , vol., no., pp.691-694, 9-11 Sept. 2011