Grenze International Journal of Engineering and Technology, June Issue

Realization of Verilog-based Low Power Universal Asynchronous Receiver / Transmitter (UART) Communication Protocol

Mansi Tripathi[1] and Gaurav Verma[2]
[1]National Institute of Technology Kurukshetra, School of VLSI Design and Embedded Systems, Kurukshetra, India
Email: 323408209@nitkkr.ac.in
[2]National Institute of Technology Kurukshetra, Department of Electronics and Communication Engineering, Kurukshetra, India
Email: gaurav@nitkkr.ac.in

***Abstract*—This article is a thorough research paper on the low power techniques using Universal Asynchronous Receiver Transmitter (UART) communication protocol with the goal of educating pertinent personnel about its applications and guiding principles while also advancing UART research's advancement. In terms of digital communication, Serial connections between distant embedded systems can be established via UART. For low-cost and low-speed applications, this protocol is utilized to attain resilience amid permissible frequency variations between systems. This study examines whether clock gating and other low power strategies may be effectively utilized in a UART receiver to minimize clock power. The suggested clock gating technique minimizes needless clock transition of operative blocks in the inactive state makes certain that the dynamic power in the operational state does not adversely impact the design.**

***Index Terms*— UART communication protocol, Xilinx Vivado, Clock Gating, Baud Rate.**

I. INTRODUCTION

In digital communication, one of the most rudimentary protocols is the Universal Asynchronous Receiver Transmitter protocol (UART) [1]. This technology has experienced decades of evolution and is extensively utilized across several domains, including the Internet of Things, industrial control, and medical treatment [2]. It is a serial data transfer protocol utilized for serial communication between devices. In simplex, communication occurs in a single direction; in half-duplex, either the transmitter or receiver communicates exclusively at any given moment, with the other remaining idle throughout transmission. When in full duplex mode, the transmitter and receiver operate simultaneously. The UART communication protocol stipulates that wireless signal transmission happens at predetermined frequency. From bytes of data, the UART successively transmits the individual bits. At the destination, the pieces are put back together into complete bytes using a second UART. The CPU connects to the UART by reading or writing one of the eight bytes, or ports. Considering a computer system frequently has many UARTs, the port addresses vary depending on which UART is being used. It is beneficial to be familiar with this communication protocol while creating reliable, high-quality goods. Error-free

Grenze ID: 01.GIJET.11.2.505

data transmission and reception can be achieved by understanding how to deliver data with just two wires and how to carry a payload, or entire pack of data [3]. The clock-gating technique, which decreases both the switching actions at the operative block level and the switching load with capacitance on the clock the distribution channel, can be utilized to cut down on clock power. Clock gating is regarded as one of the best techniques for reducing architectural and RTL power. Due to designers' knowledge of the number of flip flops or registers allocated through the data channel as well as manager of mentioned architecture, this approach is widely supported at the RTL level. It also prevents power waste from undesirable charging and discharging of the idle circuits through combination (utilizing an AND gate) of the clock which is using a signal for regulating the gate to disable the clock to a module when it is not in use. Transmitter, receiver, and baud rate generator are the three primary components of UART, which are adopted in this article using the VHDL description language. Our suggested UART is more stable, dependable, small, and simpler for serial data transmission since we use state machines for both the transmitter and the receiver. This results in reduced area consumption of the chip, or the use of LUTs and slice flip flops [4,5].

II. Framework

The sample UART is a complete, synthesizable core for a global asynchronous receiver-transmitter. The core is highly small in size and adaptable. In this paper, the baud generator, receiving module, and transmitting module are the three submodules that make up UART serial communication. Consequently, the UART communication module implementation is essentially the realization of these three smaller modules. The UART receive and transmit module is controlled by a baud generator-generated nearby clock pulse, which is substantially higher than the baud rate. Serial signals are received by the receiver module, which then transforms them into parallel data. In accordance with the fundamental frame format, the UART transmits module, interprets bytes into serial bits and sends those bits [6].

A. Baud Rate Generator

In asynchronous communication, the transmission speed is determined by the baud rate generator. It is the count of transferred symbols per second. Each bit is 1/ (baud rate) wide. Baud rate=clock freq./ (16 X divisor). A much faster clock allows the transmitter to react promptly and allows the receiving end to position the sample wave exactly intended. Knowing where to take data samples is essential. The ideal sampling time is at the centre of each serially incoming data bit [7,8].

B. Receiver Module

The data stream from the Rx pin is read by the receiver block using an oversampling technique. This technique estimates and retrieves the centre points of transmitted bits at these places. In essence, the oversampling technique serves as a clock. Here, the central point of each serial bit is estimated using sampling ticks rather than the rising edge, which is used to signify when the input signal is legitimate. The estimation can be incorrect by no more than 1/16, even though the receiver is unaware of the precise moment the start bit-the first bit-arrives [9,10].

C. Transmitter Module

Transformation of the 8-bit serial data into single-bit data is the purpose of the transmitter module. The rate is managed by the baud rate generator's one-clock cycle enabling ticks. In this case, there is no oversampling and the tick frequency is sixteen times lower than that of the UART receiver. The transmitter shares the UART receiver's baud rate generator rather than using a separate counter. It employs an internal counter to accurately record the quantity of enable ticks. A bit will be moved out after each of the 16 enable ticks [11,12].

III. Implementation

Clock-gating is one of the main strategies for power reduction that we examine. We look into how the RTL level can be used to enable this technique. The designer will need to focus on behavioural descriptions of designs at the RTL level. Before synthesis, clock-gating logic would need to be added to the behavioural description in this scenario. The transmitter and receiver do not exchange clock signals while using the UART protocol. Prior to data transmission, it is necessary to use a standard data transfer rate (baud rate) because there is no clock sharing. In other words, the baud of the transmitting UART must be communicated to the receiving UART. The rates at the transmitting and receiving end are nearly equal. LSB is the first piece of data that the transmitter shifts out.

The transmitting UART comprises of a state machine, bit cell counter, transmitted bit counter, and a serializer while the receiving end includes control state-machine, support logic and deserializer [13,14].

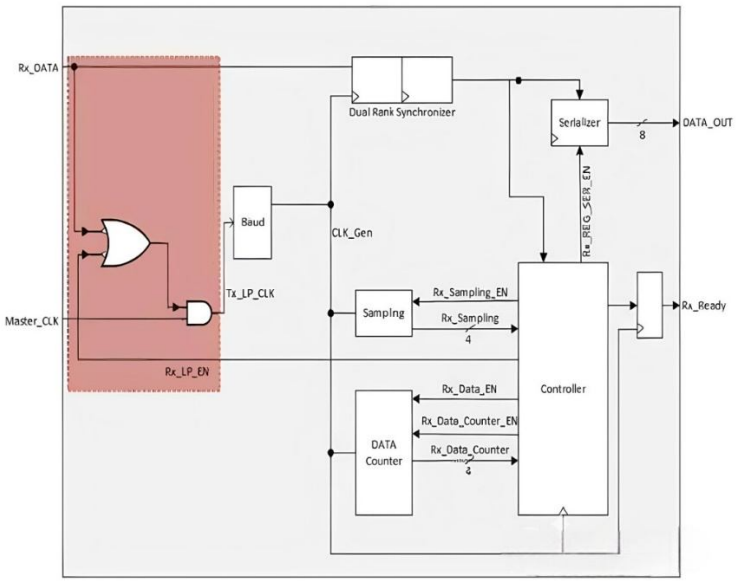

Figure 1. Component Diagram for a Clock Gated UART Receiver

The receiver of UART enters the idle state when it receives the stop bit, during which The UART receiver block is driven by the master clock needlessly. The UART receiver's functioning won't be impacted when we add logic to the receiver block at this point. To create the signal, a NOR operation is carried out between the receiver data and the controller's Rx_LP_EN enable signal. The produced signal and master clock are then ANDed to produce the Tx_LP_clk signal. After that, the newly created clock, clk_gen, will be developed by baud rate generator that will either activate or deactivate the remaining UART receiver modules in the IDLE state [15-17].

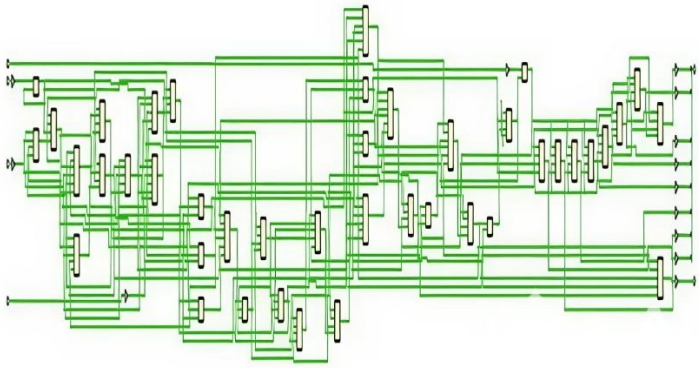

Figure 2. RTL Schematic of UART Receiver

IV. RESULTS

After using clock gating, a decrease clock power consumption is observed at the UART receiving end in the second phase. Obtaining the STOP bit of the data stream has decreased the toggling of the modified clock, lowering the dynamic power consumption of UART receiver when not in use. Following clock gating, the utilization factor, I/O consumed, LUTs, and FFs as a percentage of resources have decreased [18-20].

TABLE I. BEFORE CLOCK GATING

Resource	Estimation	Available	Utilization %
FF	17	82000	0.02
LUT	14	41000	0.03
I/O	13	300	4.33
BUFG	1	32	3.12

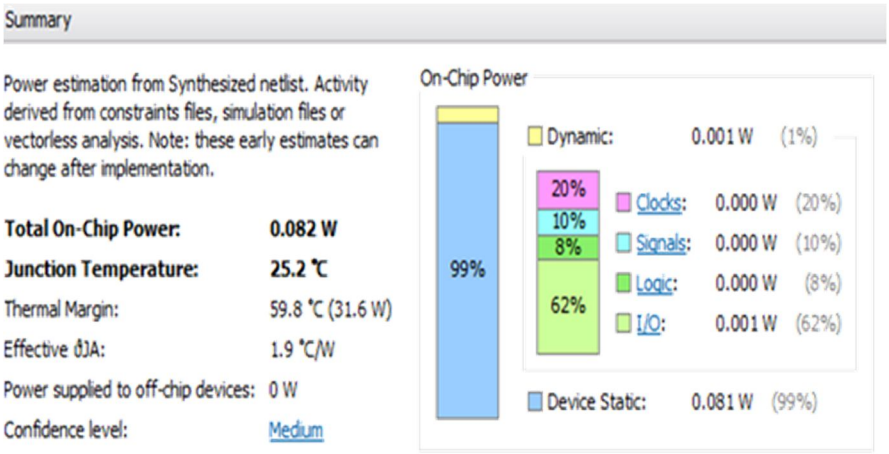

Figure 3. Power Summary of Clock Gated UART Receiver

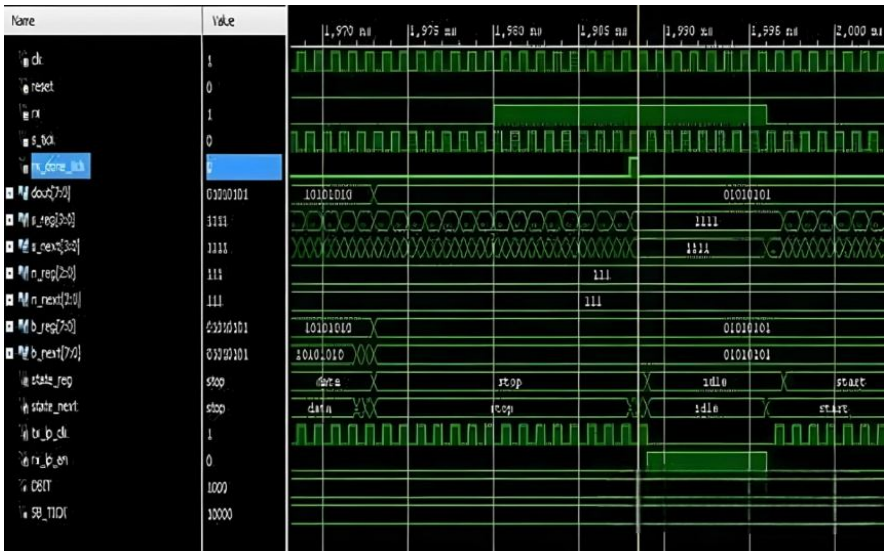

Figure 4. Low power transform of low power receiver transform

TABLE II. AFTER CLOCK GATING

Resource	Estimation	Available	Utilization %
FF	18	82000	0.02
LUT	17	41000	0.04
I/O	13	300	4.33

V. SUMMARY

Low Power Technique is used to construct a unique UART receiver architecture. The clock power consumption, utilization, and timing analysis of two UART design iterations—one with and one without clock gating—are contrasted. In order to lower the clock power consumption during data receiving, the UART receiver's control block is modified. To do this, appropriate logic is added to the UART receiver design at the RTL level. The Xilinx Vivado tool is used to simulate and synthesize the UART design.

ACKNOWLEDGMENT

With all due respect, I would like to thank Dr. Gaurav Verma who contributed his helpful suggestions towards successful accomplishment of this project. I also want to thank all the professors at the School of VLSI Design and Embedded Systems.

REFERENCES

[1] Weilun Huang and Guolun Sheng, "Analysis and Research of UART Communication Protocol" 2024 4th Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS) DOI: 10.1109/ACCTCS61748.2024.00140

[2] Ashok Kumar Gupta, Ashish Raman, Naveen Kumar, and Ravi Ranjan, "Design and implementation of universal asynchronous transmission protocol (UART)" *In 7th International Conference on Signal Processing and Integrated Networks (SPIN)* 2020 Feb 27 (pp. 295- 300).

[3] Dipanjan Bhadra, Vikas S. Vij, Kenneth S. Stevens., "A low power UART design based on asynchronous techniques". *In 2013 IEEE 56th International Midwest Symposium on Circuits and Systems (MWSCAS)* (pp. 21-24).

[4] Christine Qureshi1 .Amita Agnihotri2, "Low Power UART Design Using Low Power Techniques" IJSRD - International Journal for Scientific Research & Development| Vol. 4, Issue 02, 2016 | ISSN (online): 2321-0613.

[5] Tapas Tewary and Atanu. Sen, "A novel approach to realize built-in-self-test (BIST) enabled UART using VHDL," *First International Conference on Automation, Control, Energy and Systems (ACES)*, A disaptagram, India, 2014, pp. 1-6, DOI: 10.1109/ACES.2014.6808031.

[6] Dawoud, "Serial Communication Protocols and Standards RS232/485, UART/USART, SPI, USB, INSTEON, Wi-Fi and WiMAX," *Copyright Year***:** 2020**,** *Book Type***:** River Publishers.

[7] Yuelun Zhang,Zongguang Yu and Xiaogang Yang, "Design and implementation of Configurable UART With Self-Test Function" 2023 5th International Conference on Electronic Engineering and Informatics (EEI) | 979-8-3503-2707-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/EEI59236.2023.10212446.

[8] Garima Bandhawarkar Wakhle, Iti Aggarwal and Shweta Gaba, "Synthesis and Implementation of UART Using VHDL Codes," *International Symposium on Computer, Consumer and Control*, Taichung, 2012, pp. 1-3. DOI: 10.1109/IS3C.2012.10

[9] R Baby Chithra*, Shreyas A R, Noor e Mujassim, Monika Gupta and Prem Kumar N* "Design and Implementation of UART With Effective Serial Communication" 2024 Asia Pacific Conference on Innovation in Technology (APCIT) | 979-8-3503-6153-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/APCIT62007.2024.10673484

[10] Yongcheng Wang, Kefei Song., "A new approach to realize UART" *International Conference on Electronic and Mechanical Engineering and Information Technology*, EMEIT 2011, Harbin, Heilongjiang, China 12-14 August, 2011.

[11] J. Norhuzaimin and H. H. Maimun, "The design of high-speed UART," *Asia-Pacific Conference on Applied Electromagnetics*, Johor, 2005, pp. 5 pp.-DOI: 10.1109/APACE.2005.1607831

[12] Rahul Sonwane, Atul Deshmukh and Swapna Choudhary "UART (Universal Asynchronous Receiver Transmitter) for Serial Data Communication: Design and Implementation on FPGA Platform" 2023 2nd International Conference on Futuristic Technologies (INCOFT) | 979-8-3503-0884-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/INCOFT60753.2023.10425295

[13] Akshatha, "Design and Implementation of Power Efficient FSM based UART", *Journal of Physics: Conference Series, Volume 2161, 1st International Conference on Artificial Intelligence, Computational Electronics and Communication System (AICECS 2021)*, 28-30 October 2021, Manipal, India**.**

[14] Mohammed Azeemuddin and Prof. S.K Ahmeduddin Zakir "Architectural Low Power of UART using Verilog" International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 3 Issue 2, February – 2014

[15] Jayesh More, Rushank Suryavanshi, Gaurav Dasarwar, Sivanantham S* and Sivasankaran K, "FPGA Implementation of Universal Asynchronous Transmitter and Receiver" 2015 Online International Conference on Green Engineering and Technologies (IC-GET 2015) 978-1-4673-9781-0/15/$31.00 © 2015 IEEE

[16] Pranjal Sharma, Anup Kumar and Naresh Kumar "Analysis of UART Communication." *In International Conference on Edge Computing and Applications (ICECAA)*, IEEE 2022.

[17] Lehan Cao, Jianfeng Chen and Jincheng Li, "Working principle and application analysis of UART". 2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA) DOI: 10.1109/EEDA56825.2023.100905

[18] Samir Palnitkar, *"Verilog – A Guide to Digital Design and Synthesis"* by Sunsoft Press 1996.

[19] A. S. Keerthi Nayani,G. Nikhilesh and S. Shiva Tej Kumar, "Design and Implementation of UART Protocol for Avionics Applications," *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*,2019, DOI:10.1109/ICCS45141.2019.9065366.

[20] Naresh Patel, Vatsalkumar Patel, and Vikaskumar Patel, "VHDL Implementation of UART with Status Register," *International Conference on Communication Systems and Network Technologies*, Rajkot, 2012, pp. 750-754. DOI: 10.1109/CSNT.2012.164