WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

ENHANCING 64-BIT BARREL SHIFTER PERFORMANCE WITH LOW POWER OPTIMIZATION

Dr .K.VENUGOPAL RAO [1], P. AKHILA [2], G. MADHU [3], P. NAVYA[4], M. MANOJ [5]

[1]ASSOCIATE PROFESSOR IN DEPARTMENT OF ECE IN JYOTHISHMATHI INSTITUTE OF TECHNOLOGY &SCIENCE IN JITS RD, RAMAKRISHNA COLONY, VACHUNUR, TELANGANA, 505481.
[2,3,4,5]FINAL YEAR STUDENTS IN DEPARTMENT OF ECE IN JYOTHISHMATHI
INSTITUTE OF TECHNOLOGY & SCIENCE IN JITS RD, RAMAKRISHNA COLONY, VACHUNUR, TELANGANA, 505481.

ABSTRACT:

A shifter of barrels may be progressed circuit that can exchange a information by a certain number of bits in a single clock cycle. This might be conceivable to execute no of multiplexers, in such execution one mux's yield is related with taking after independently, in such a way that is predicated on the move. Along with number juggling and method of reasoning operations barrel shifter is utilized to move a needed number of bits in a needed course. Barrel shifter is a Computerized circuit that can move a information word by a indicated number of bits in one clock cycle. It can be executed as a grouping of multipliers (Mux), and in such an execution the yield of one Mux is associated to the input of the another Mux in a way that depends on the move separate. In math and rationale operations barrel shifters is utilized to move a wanted number of bits in a wanted direction. In conclusion, the extend speaks to a effective combination of progressed innovation and imaginative strategies. Which create calculable comes about. It illustrates that the approach yields, tall speed barrel shifter & delay have to reduce.

INTRODUCTION

A barrel shifter is a advanced circuit that can move a information word by a indicated number of bits. It can be executed as a grouping of multiplexers. In this usage, the yield of one MUX is associated to the input of the following MUX in a way that depends on the move remove. The number of multiplexers required is n*log2(n), for an n bit word. Four common word sizes and the number of multiplexers required are recorded below: • 64-bit --- 64*log2(64) = 64*6=384 • 32-bit --- 32*log2(32) = 32*5=160 • 16-bit --- 16*log2(16) = 16*4=64 • 8-bit --- 8*log2(8) = 8*3=24 A barrel shifter is a pivotal component in computer frameworks, especially in errands requiring information control such as Computerized Flag Preparing (DSP). Its essential work is to productively move or turn information by foreordained bit positions inside a single clock cycle. By utilizing incremental stages, barrel shifters streamline the moving prepare, disposing of the require for different clock cycles and diminishing the time required for information control. This productivity makes barrel shifters vital in computationally seriously applications where fast information preparing is essential. In advanced frameworks, moving includes moving twofold digits (bits) of a parallel number cleared out or right by a indicated number of positions. Revolution, a variation of moving, offers likenesses with moving in that it moves bits to the cleared out. Be that as it may, in revolution, the bits that "drop off" from the cleared out side are added back onto the right side as lower- arrange bits, successfully shaping a circular course of action. In differentiate, moving takes off purge spaces in the lower-order bits after the move, regularly filled with zeros. Both moving and turning operations are crucial in different computer operations, extending from address interpreting to number-crunching computations. The significance of barrel shifters is underscored by their arrangement on the basic way in numerous computing assignments. Thus, critical investigate endeavors have been committed to optimizing barrel shifter plans for speed, pointing to minimize the time required for information control. Be that as it may, with the rise of portable computing, vitality proficiency has risen as a basic thought nearby speed in circuit design. In a venture pointed at creating 64-bit barrel shifters, a extend of plan varieties is investigated, enveloping contrasts in gate-level executions, engineering arrangements, and contemplations for the working environment. By tending to different angles of barrel shifter plan, such as speed, control proficiency, and natural versatility, this extend looks for to offer a comprehensive understanding of barrel shifter innovation and its suggestions for cutting edge computing systems.

LITERATURE REVIEW

Many People worked on barrel shifters for improving the enhancement on some factors such as improving the area efficiency, power optimization and reducing LUTs and many more factors. Muchukota Suresh Babu et.al, [1] emphasizes that a system's speed hinges on frequency and parallel processing, crucial within embedded real-time systems. Current systems typically operate at 2 to 3 GHz, and high-speed processing is provided by the 64-bit ALC Control Unit (CU), supporting arithmetic, logic, and code conversion tasks. Structurally implemented in Verilog HDL, it's optimized for speed, low power consumption (under 1W), and can execute 32 operations. Utilizing a xc7a100tcsg324-1 Artix 7 FPGA with 100K gates, clock rates from 10 MHz to 20 GHz have been achieved.. Sandeep Chintala et.al, [3] describes the creation, confirmation, and evaluation of a 32-bit CPU. With target verification, the Xilinx Vivado System Design Suite software tools process the whole front-end design flow performed on the Artix 7 FPGA [6]. The verification procedure uses the concept of virtual I/O. It will do 32 separate procedures, such as the creation of parity and the conversion of binary and grey code. Power analysis was done on this low-power Verilog HDL architecture at clock frequencies ranging from 10MHz to 100GHz [6]. Power analysis has been demonstrated for multipleI/O standards, such as LVCMOS12, LVCMOS25, and LVCMOS33 (Low Voltage Complementary Metal Oxide Semiconductor), at all frequencies [8]. Anjali et.al, [4] provides a comparison between autogenerated and semicustom 4-bit barrel shifter designs. The semi-custom design takes up less space than the autogenerated design. Semicustom design reduces area by 35 percent. Regarding size and energy usage, the semicustom design is ideal.32 operations. Utilizing S. Kiran Babu et.al,[5] expound that a barrel shifter is an electronic circuit capable of shifting a data term created by a precisely how many bits are used in one clock cycle. applied in a sequential manner of multipliers (Mux), each output is connected to the next based on the shift distance. These are used in arithmetic and logic operations to shift bits in a predetermined direction. The paper proposes and implements 16-bit and 32-bit barrel shifter architectures in Verilog code. Priyanka Agrawal et.al,[6] and [7] explicate the two distinct MUXes are two distinct designs were created, one utilising pass transistor logic and the other universal gates. When designing NAND gates, 14 transistors are employed, while pass transistor logic uses 6 transistors. Using a traditional barrel shifter circuit with a universal gate requires 56 transistors, resulting in higher power consumption and time delay. The barrel shifter that is recommended employs 24 transistors, which reduces power consumption and time delay. Table 2 displays the reduced percentage. Thus, the proposed barrel shifter outperforms the standard circuit. . Prasad D. Khandekar et.al, [9] and [10], explains RISC processors are prevalent in embedded systems, leveraging barrel shifters for logical and arithmetic tasks like shift, rotation, division, and multiplication, offering significant speed advantages. While standard shifters require 'n' clock cycles for n-shift or rotate operations, barrel shifters can accomplish them in a single cycle. 64-bit barrel shifters are particularly valued for 64-bit CPU development. This work showcases the RTL design, implementation, and verification of a 64-bit barrel shifter utilizing a front-end computer programme and an Artix 7 FPGA.

PROBLEM DEFINATION

The involvement allow the reproduction of the inputs from the observed [2]. Data shifting and rotating is often used operation, in this regard, barrel shifters which are capable of performing n-bit shifting and rotating of data in a single cycle, are normally used in embedded processors such as: digital signal processors [4] and high- of the design and implementation of a 32-bit FPGA-based barrel shifter with the primary objective of executing left and right logical and arithmetic shifts on binary data with precision.

The key requirements encompass optimizing performance through efficient utilization of FPGA capabilities, allowing configurability for shift type and amount, and ensuring resource efficiency within FPGA constraints. Challenges include managing a fixed bit width, addressing timing constraints, and balancing complexity with performance considerations. Successful completion involves delivering a hardware description language (Verilog/VHDL) implementation, synthesis, and the functional deployment of the 32-bit barrel shifter on the target FPGA platform. Success will be determined by rigorous testing, demonstrating accurate functionality, and adherence to specified requirements.

LIMITATIONS

There are some of the limitations based on previous methods:

- Resource utilization.
- Scalability.
- Timing Constrains.
- Efficiency.
- Area.

PROPOSED SYSTEM

The design of a 64-bit barrel shifter for FPGA implementation follows a systematic approach. Firstly, the selection of an appropriate algorithm for left and right logical and arithmetic shifts on 64-bit data is crucial. Algorithms like iterative shift, parallel shift, or bitwise rotation are common choices depending on performance requirements.

In the design specification phase, engineers establish input and output interfaces, determining the shift type (logical or arithmetic) and configuring the shift amount. Additionally, they prioritize features like parallel processing to boost processing speed and optimize resource usage, ensuring efficient utilization of FPGA components. This phase lays the groundwork for subsequent design stages, guiding the development of a shift register that meets performance requirements while effectively utilizing hardware resources. Detailed consideration of these factors ensures alignment with project objectives and facilitates the creation of a robust and efficient shift register design.

Performance analysis follows implementation to evaluate speed, efficiency, and resource utilization.

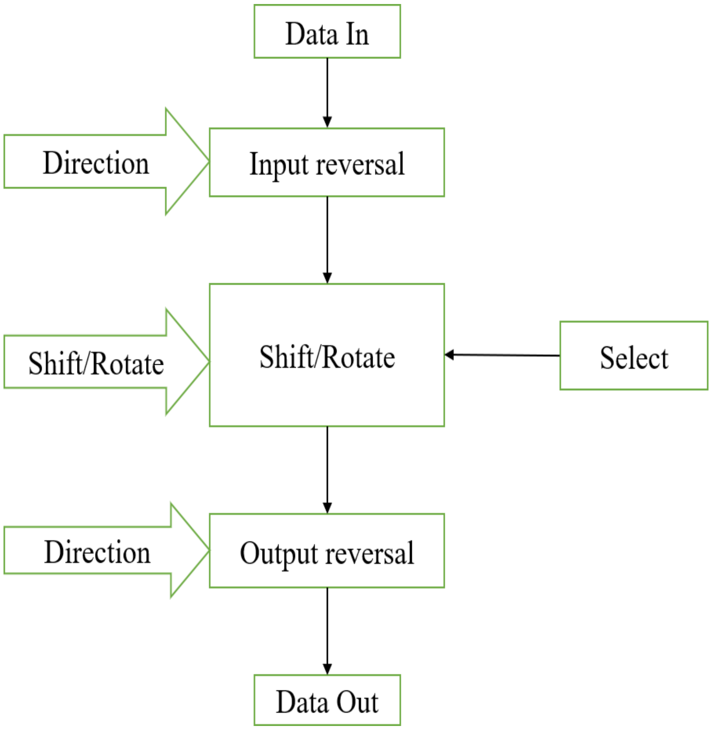


Fig: Block Diagram

Any identified areas for improvement are addressed to optimize the design further. By adhering to this structured methodology, a well-designed and efficient 64-bit barrel shifter is developed for FPGA deployment. meeting the specified requirements and ensuring reliable performance.

Working

Input Data (Data In):- The Barrel Shifter receives a 64-bit data input. This data represents the information that needs to be shifted or rotated.

Input Reversal:- The input reversal operation involves rearranging the order of bits in the input data. This initial step ensures that the data is in a suitable format for subsequent processing stages. By reversing the input data at the outset, subsequent operations can be performed more effectively. This preparatory step is often employed to streamline data processing pipelines and optimize performance in digital systems.

Shift/Rotate Stage:- The heart of the Barrel Shifter lies in this stage. It performs the actual shifting or rotation of bits. Depending on the control signals (often denoted as S0, S1, and S2), the data can be shifted left or right. For example:If S0 is asserted and S1 and S2 are unasserted, the value at D0-7 is passed to the next most significant output (Q1-Q7).

The shifter can cycle the order of the bits ABCD as DABC, CDAB, or BCDA, preserving all bits without loss.It can shift all outputs up to three positions to the right, creating any cyclic combination of A, B, C, and D.

Output Reversal:- The output reversal stage involves reversing the order of bits in the shifted data after the shift or rotate operation. This crucial step ensures that the shifted data is presented in the desired format for the final output. By reversing the shifted data before output, the resulting data is properly formatted for downstream processing or transmission. This preparatory step is often necessary to ensure data integrity and compatibility with external systems or interfaces.

Data Out:- The resulting data, after all the operations, emerges as the output.

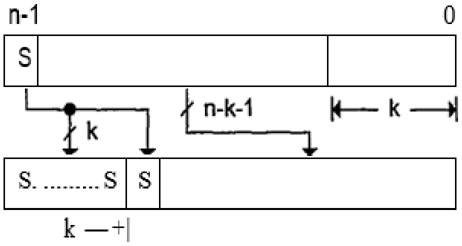


Fig : Shift Right Arithmetic

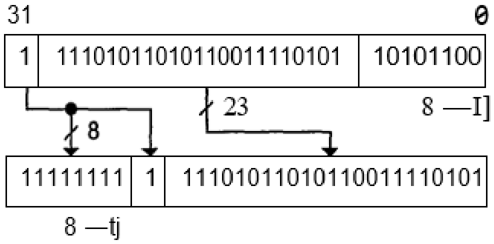


Fig : Shift Right Arithmetic by 8

This shifted or rotated data is now ready for further processing or storage. The Barrel Shifter is a crucial component in modern microprocessors, often used alongside the Arithmetic Logic Unit (ALU). It enables efficient data manipulation, such as address generation, variable-length coding, and bit ordering.

Program or Make Device

Finally, the gate-level netlist is used to program an FPGA (Field-Programmable Gate Array) or an ASIC (Application-Specific Integrated Circuit). The programmed device becomes the physical implementation of the digital system The Block diagram of the project is Shown in figure

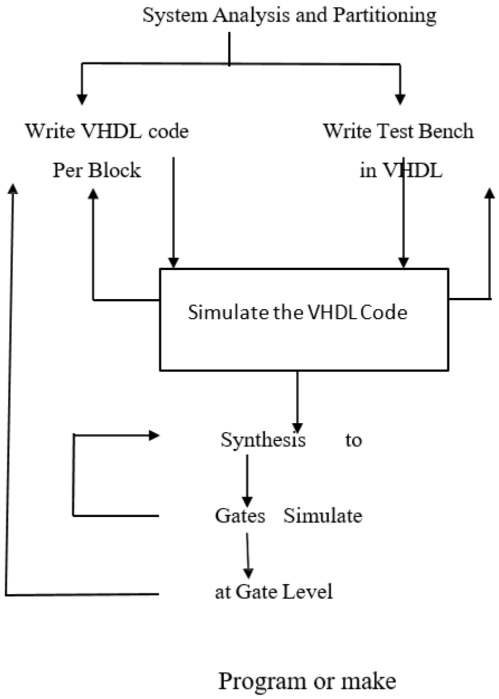
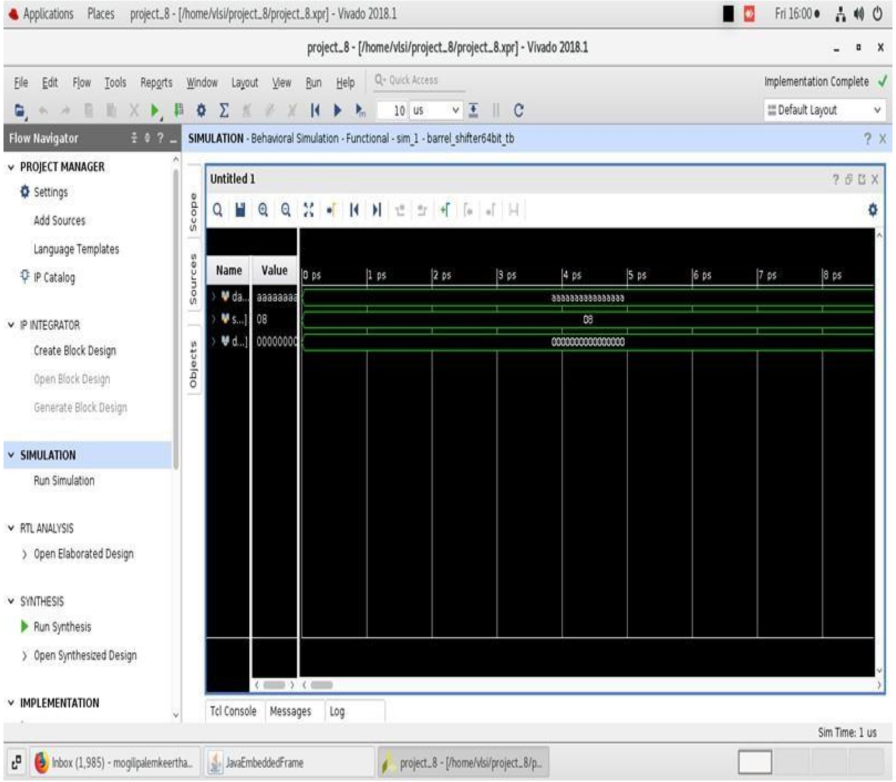


Fig : Design flow

RESULTS

In binary, 64'hAAAAAAAAAAAAAAAA represents the pattern 0b1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010. A shift of 8 bits to the right would cause this pattern to rotate, moving the rightmost 8 bits to the leftmost positions. The result would be 64'hAAAAAAAA (binary 0b1010 1010 1010 1010 1010 1010 1010 1010). The expected output is 64'hAAAAAAAAAAAAAA (binary 0b1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010 1010).



SIMULATION - Behavioral Simulation - Functional - sim_1 - barrel_shifter64bit_tb

Name	Value	
da...	aaaaaaaa	aaaaaaaaaaaaaaaa
s...]	08	08
d...]	0000000	0000000000000000

Fig : Simulation Results for Shift Right Operation

CONCLUSION

Today's technology uses 64-bit and 32-bit processing processors. The bulk of earlier barrel shifter designs are 32-bit in capacity. The barrel shifter provides a versatile and efficient of custom IP design with 28nm. As a result, the concept is implemented with 64-bit size enhance the propagation delay, power usage, number of LUT, area and efficiency, through the application of low power techniques.

The future of 64-bit Barrel Shifters is promising, marked by their scalability to larger word sizes, catering to evolving computational needs. Their adaptability to specialize in specific tasks ensures relevance in diverse application domains. Integration with emerging technologies like neuromorphic computing enhances their utility in advanced computing paradigms. Continuous improvements in performance, power efficiency, and reliability further solidify their position in next-generation processor architectures.

REFERENCES

[1] D. A. Devi and L. S. Sugun, "Design, implementation and verification of 32-Bit ALU with VIO," 2018 2nd International Conference on Inventive Systems and Control(ICISC), Combatore, 2018, pp. 495-499, do: 10.1109/ICISC.2018.8399122.

[2] S. Kiran Babu, K. Niranjan Reddy, Design and Implementation of FPGA Based 32-Bit Barrel Shifter, International Journal of VLSI System, Vol.04, Issue 03, p-p: 0169-0171, ISSN: 2322-0929. March-2016.

[3] ASHA DEVI Dharmavaram, SANDEEP Chintala, SUGUN L, Sai. Design of Power Efficient 32-Bit Processing Unit. International Journal of Engineering & Technology, [S.1.], v. 7, n.2.16, p. 52-56, apr. 2018. ISSN 2227-524X.

[4] Dharmavaram Asha Devi, Muchukota Suresh Babu. Design and Analysis of Power Efficient 64-Bit ALCCU, International Journal of Recent Technology and Engineering (IRTE), ISSN:2277-3878, Volume-8 Issue-2, pp-162-166, July 2019.

[5] Anjali Thakur, Rajesh Mehra, CMOS Performance Analysis of 4-bit Barrel Shifter, Journal of Emerging Technologies and Innovative Research, Vol V, Issue 6, p-p: 123-128, eISSN: 2349-5162. June-2018.

[6] Prasad D Khandekar, Dr. Mrs. Shaila Subbaraman, “Low Power 2:1 MUX for Barrel Shifter,” International Conference On Emerging Trends In Engineering and Technology, IEEE, pp.404-407, July 2008.

[7] Latif khan, Designing of low power high speed 32-bit barrel shifter for ARM7 Processor, International Journal of Advanced Research in Computer Science, “ Vol. IV, No. 3, March 201