**FACTA UNIVERSITATIS**
Series: **Electronics and Energetics** Vol. 38, Nº 4, December 2025, pp. 755 - 769
https://doi.org/10.2298/FUEE2504755M

**Original scientific paper**

# DESIGN OF A NOVEL 8-BIT BARREL SHIFTER FOR COMBINED SHIFT AND ROTATE OPERATIONS

**Prachi Mukherji, Seema Rajput, Sagar Vanarase, Apeksha Gundale, Aparna Gumaste, Prerna Patil, Rutuja Dalal**

Department of Electronics & Telecommunication Engineering,
MKSSS's Cummins College of Engineering for Women, Pune, India

| ORCID IDs: | Prachi Mukherji | https://orcid.org/0000-0001-5554-5583 |
|---|---|---|
| | Seema Rajput | https://orcid.org/0000-0002-7096-5145 |
| | Sagar Vanarase | https://orcid.org/0009-0008-0741-3920 |
| | Apeksha Gundale | https://orcid.org/0009-0007-5147-5726 |
| | Aparna Gumaste | https://orcid.org/0009-0005-5156-8127 |
| | Prerna Patil | https://orcid.org/0009-0009-5896-4164 |
| | Rutuja Dalal | https://orcid.org/0009-0000-8354-7252 |

***Abstract.*** *In this paper, a novel 8-bit barrel shifter is designed using both standard logic gates and transmission gates. The novelty of our design supports left shift, right shift, left rotate, and right rotate operations within a single structure, enabling versatile directional control through data reversal techniques that allow streamline control flow and reduce logic complexity. Without limiting functionality to one operation per cycle, this shifter achieves all four operations compactly, optimizing resource use and operational flexibility. The design is implemented and validated at 45 nm and 180 nm CMOS technology nodes, with waveform analysis confirming reliable performance across all conditions. Simulation results indicate that the transmission gate-based configuration significantly improves power efficiency with minimum power dissipation of 98.67 nW achieving up to 99.4% reduction compared to reversible logic-based counterparts, making it ideal for power-sensitive applications. This barrel shifter advances conventional designs, delivering a flexible and power-optimized solution suitable for modern digital environments.*

**Key words**: *barrel shifter, multiplexer, shifter, rotator, data reversal*

Received February 20, 2025; revised April 21, 2025, April 30, 2025 and May 05, 2025; accepted May 08, 2025
**Corresponding author:** Dr. Prachi Mukherji
Department of Electronics and Telecommunication Engineering, MKSSS's Cummins College of Engineering for Women, Pune, Maharashtra, India
E-mail: prachi.mukherji@cumminscollege.in

## 1. INTRODUCTION

The design and implementation of efficient barrel shifters is crucial in modern digital circuit design. Shifters are fundamental components in digital signal processing (DSP) and are essential for optimizing data in critical operations like division and multiplication across units such as Floating-Point Units (FPUs), Arithmetic Logic Units (ALUs), DSP units, and Graphics Processing Units (GPUs). These units rely on efficient data-shifting mechanisms to perform tasks ranging from numerical calculations to complex graphical rendering and signal filtering. Among these mechanisms, the barrel shifter is particularly notable for performing multi-bit shifts in a single operation, a capability inspired by the synchronized movement of a gun barrel. This paper proposes a novel 8-bit architecture of the barrel shifter integrating shift and rotate operations in left and right directions, which is achieved using data reversal, leveraging CMOS technology known for high integration density and low power consumption, ideal for implementing complex digital circuits like barrel shifters.

Barrel shifters are often incorporated as a part of the Arithmetic Logic Unit (ALU) for rapid bit manipulation as well as for fast multiplication and division operations. In DSP algorithms, such as audio equalization (e.g. bass boost and filtering), barrel shifters are essential for rotating and shifting coefficients. By combining shift and rotate operations within a single shifter, the design streamlines these tasks, leading to faster filter updates and smoother real-time audio processing. In graphics applications, data rotation is commonly used for pixel manipulation and bitwise rotation operations, such as shifting color channels or processing textures. A combined shift/rotate unit allows for more efficient handling of pixel data. Barrel shifters are also used in microcontrollers for faster data manipulation in tasks like text processing and data compression algorithms.

The remaining paper is organized as follows: Section 2 presents the literature survey summarizing techniques, technologies, and findings related to barrel shifter design with a focus on low power, compactness, and efficiency across different logic families and technology nodes. Section 3 describes the design and working of the proposed 8-bit barrel shifter using a 2:1 multiplexer, implemented with standard logic gates and transmission gates. Section 4 discusses the design and operation of the barrel shifter. Section 5 presents the simulation results of the 8-bit barrel shifter designed in Cadence Virtuoso 6.1. Section 6 summarizes the design achievements, power efficiency, and comparative advantages of the implemented 8-bit CMOS barrel shifter and outlines future directions for scalability and optimization in advanced VLSI applications.

## 2. LITERATURE SURVEY

The literature survey aims to explore and analyze the existing body of work related to the subject of this research. It provides an understanding of the foundational concepts, methodologies, and technologies that have been previously proposed, implemented, or studied. This review helps in identifying the current trends, research gaps, and areas where further improvement or innovation is possible.

A low-power 4-bit barrel shifter is proposed in [1], optimized for current steering Digital to Analog Converter (DACs) using 65 nm technology. By leveraging Gate Diffusion Input (GDI), modified GDI, and full-swing GDI logic for designing 2:1 multiplexer, the design achieved significant power savings and layout efficiency, demonstrating its applicability in

low-power digital systems. Similarly, the effectiveness of GDI logic in the design of an 8-bit barrel shifter is demonstrated in [2], with a focus on compactness and power efficiency, making it suitable for resource-constrained environments.

In the context of asynchronous systems, completion detection strategies and their integration into barrel shifter designs are explored in [3], enhancing circuit functionality and reliability. Further investigations into low-power designs are conducted in [4], where methodologies for barrel shifters and rotators in 45 nm CMOS technology are compared. The study provides critical insights into achieving optimal trade-offs between power efficiency, performance, and area.

Advancements in nanoelectronics have also played a pivotal role in the evolution of barrel shifter designs. The importance of scalability and low-power operation in nanoelectronics circuits is emphasized in [5], aligning with the work presented in [6], which introduced an 8×4 barrel shifter with a binary-to-gray code converter using FinFET technology at an 18 nm node. This design, while suitable for low-power communication systems, highlighted challenges related to minimizing transistor count.

The use of reversible logic has gained prominence for its energy-efficient properties. Reversible universal barrel shifters designed for low-power applications are presented in [7], while [8] introduces reversible bidirectional barrel shifters using multiple-valued logic. These innovations highlight the potential of reversible logic for next-generation low-power digital systems.

Layout efficiency and design optimization are critical for barrel shifters. An area-efficient CMOS barrel shifter layout is proposed in [9], providing valuable insights into reducing circuit complexity. A comparison between complementary MOS and pseudo-NMOS logic in barrel shifters is conducted in [10], demonstrating the potential of pseudo-dynamic approaches for low power shifting operations. Further investigations into nano-scale CMOS technologies are carried out in [11], analyzing 4-bit barrel shifters at 45 nm and 90 nm nodes to optimize delay and power.

Design alternatives for barrel shifters are explored in [12], introducing functionality for combined shift and rotation operations controlled by select inputs. This work is complemented by [13], which presents a ternary barrel shifter utilizing reversible multiple-valued logic, expanding applications to non-binary digital systems.

Additionally, performance trade-offs in 45 nm CMOS-based barrel shifters are analyzed in [14], with a focus on achieving power and area efficiency. Application-specific designs are addressed in [15] through the development of area-efficient controllers for barrel shifters in reconfigurable Low-Density Parity-Check (LDPC) decoders, further highlighting the versatility of barrel shifter designs. A comprehensive review of barrel shifter designs, emphasizing optimization techniques for improving processing speed and minimizing power consumption, was provided in [16]. These findings are supported by innovations in pseudo-dynamic logic and FinFET technologies, demonstrating their adaptability to modern digital systems.

## 3. METHODOLOGY

In digital design, multiplexers are essential for implementing barrel shifters, data routing, and various control logic. A multiplexer (MUX) is a combinational circuit used to select one of multiple input signals and forward the selected input to a single output line.

Multiplexers are implemented using standard logic gates and transmission gates (TGs).

### 3.1. Multiplexer using Standard Logic Gates:

A 2:1 multiplexer implemented with standard logic gates consists of:
i) Two AND gates to create paths based on the select line input.
ii) One OR gate to combine the outputs of the AND gates.
iii) One Inverter to control the selection between the two input signals.

It consists of two data inputs A0 and A1and select line, INPUT. Depending on the selected line, data inputs are passed at the output.

One advantage of using standard logic gate multiplexers is that they are easy to implement and are robust, providing full voltage swing outputs as shown in Figure 1.

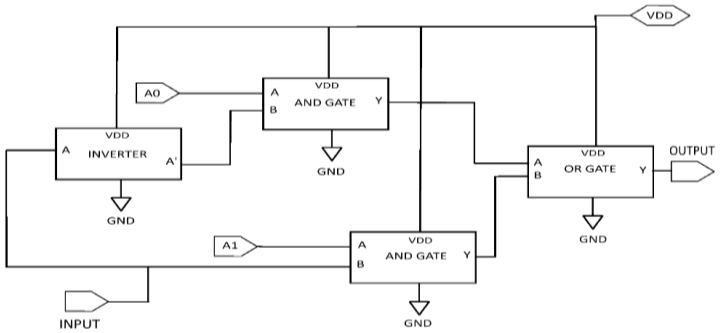


**Fig. 1** Mux using standard gates

### 3.2. Multiplexer Using Transmission Gates (TGs):

A 2:1 multiplexer using transmission gates is typically implemented with:
i) One NMOS and one PMOS transistor paired together as a transmission gate.
ii) Control signals used for selecting the input.
iii) Inverter to control the selection between the two input signals.

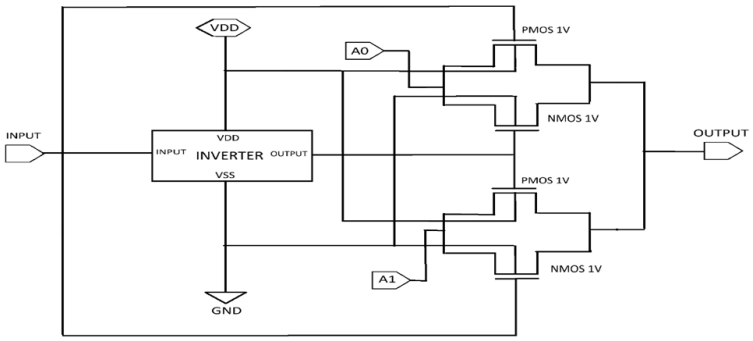


**Fig. 2** Mux using transmission gates

Transmission gates operate by using complementary NMOS and PMOS transistors that allow the selected input signal to pass through without loss of voltage, as illustrated in Figure 2.

TG-based multiplexers provide efficient, high-speed operation with full output voltage swing, reduced power consumption, and fewer transistors.

Figure 3 shows the circuit of the proposed integrated and efficient 8-bit barrel shifter. A total of forty-eight 2:1 multiplexers are used.

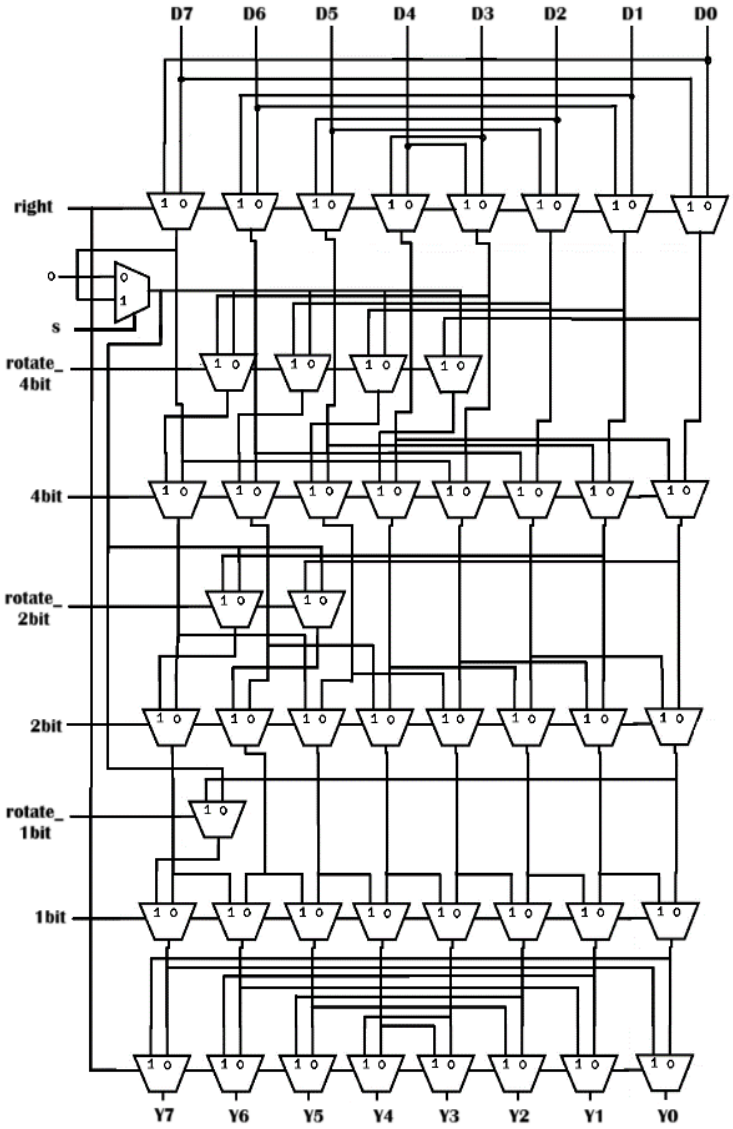


**Fig. 3** Schematic Diagram of Barrel Shifter

A barrel shifter can perform both logical shifts and circular shifts (rotations). In a logical shift, data bits are moved left or right, and the vacated bit positions are filled with zeros. In

a circular shift, also known as rotation, the bits that are shifted out on one end are wrapped around and reintroduced at the opposite end, maintaining the overall bit count.

The barrel shifter is controlled using several signals: right, s, rotate_1bit, rotate_2bit, rotate_4bit, shift_1bit, shift_2bit, and shift_4bit. The right signal determines the direction of the operation. When right is 0, all shift and rotation operations are performed to the left. When right is 1, they are performed to the right. In this case, the data is first

reversed, then the operation is applied, and finally the data is reversed again to achieve the correct rightward behavior.

The s signal selects between shift and rotate operations. When s is 0, a logical shift is performed. When s is 1, a rotation is performed. While performing a rotation, the same select pins used for shifting (such as 1 bit, 2 bit, or 4 bit) must be set high, and the shift select pins must be set low. Conversely, for a shift operation, the rotate select pins should be kept low.

Table 1 provides a summary of all operations. The user configures the appropriate select inputs (e.g., rotate_1bit, shift_2bit) based on the desired number of bits to be shifted or rotated.

**Table 1** Input Selection for Shift and Rotate Operations

| Select Inputs | | Operations |
|---|---|---|
| s | right | |
| 0 | 0 | Left shift |
| 0 | 1 | Right shift |
| 1 | 0 | Left rotate |
| 1 | 1 | Right rotate |

## 4. Design and Operation of Barrel Shifter

This section discusses the design and operation of the barrel shifter by performing a 2-bit right rotate operation on an 8-bit binary input. The circuit consists of eight rows of multiplexers, which allows for flexible bit-shifting, and rotation based on specified control signals.

### 4.1. Shift and Rotate Operation:

**For left shift:** Set s=0, right=0, and 1 bit/ 2 bit/ 4 bit which are the control signals that can be set according to the desired number of bits the user wants to shift.

**For right shift:** Set s=0, right=1, and 1 bit/ 2 bit/ 4 bit which are the control signals and are kept as per the desired number of bits the user wants to shift.

**For left rotation:** Set s=1, right=0, and keep rotating bits (rotate_1bit/ rotate_2bit/ rotate_4bit) and shifting bits (1 bit/ 2 bit/ 4 bit) the same as per the desired number of bits the user wants to rotate.

**For right rotation:** Set s=1, right=1, and keep rotating bits (rotate_1bit/ rotate_2bit / rotate_4bit) and shifting bits (1 bit/ 2 bit/ 4 bit) the same as per the desired number of bits the user wants to rotate.

Figure 4 shows the circuit of the barrel shifter. Input to the barrel shifter is applied through the Vpulse instance in Cadence Virtuoso. 8-bit input (D0 to D7) is applied to the

barrel shifter, and it shifts/rotates in the left or right direction based on a 3-bit control input (S0 to S2), allowing shift/rotation from 0 to 7 positions. The data inputs are given on the left (D0 to D7), and the shift/rotate control lines are connected at the bottom. The shifted 8-bit output (Y0 to Y7) is displayed on the right.

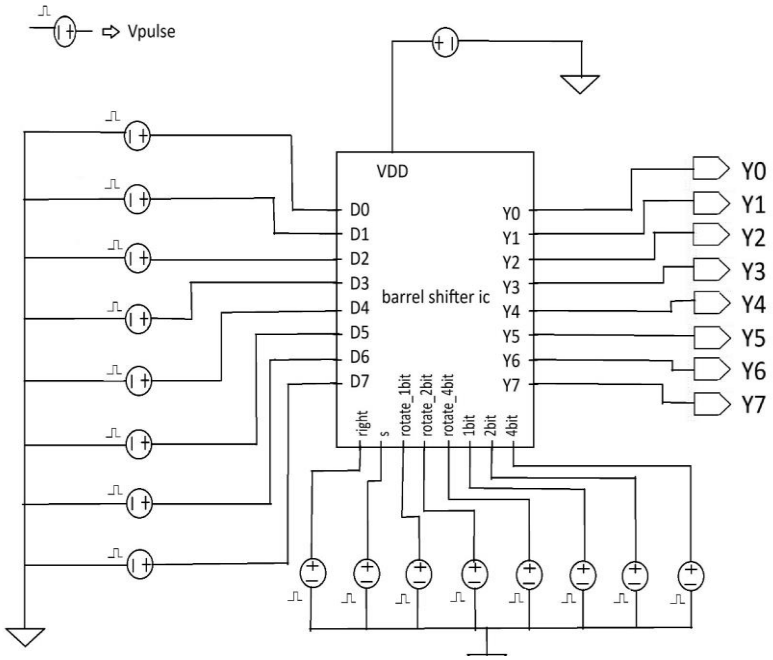


**Fig. 4** IC Circuit of Barrel shifter

Table 2 shows the actual output generated after performing different operations. This table essentially serves as the truth table for the given inputs, detailing the outputs corresponding to all select input pins.

**Table 2** Truth Table of all operations performed by integrated barrel shifter

| Select Inputs | | | Shift | | Rotate | |
|---|---|---|---|---|---|---|
| 4 bit | 2 bit | 1 bit | Left | Right | Left | Right |
| 0 | 0 | 0 | 10100010 | 10100010 | 10100010 | 10100010 |
| 0 | 0 | 1 | 01000100 | 01010001 | 01000101 | 01010001 |
| 0 | 1 | 0 | 10001000 | 00101000 | 10001010 | 10101000 |
| 0 | 1 | 1 | 00010000 | 00010100 | 00010101 | 01010100 |
| 1 | 0 | 0 | 00100000 | 00001010 | 00101010 | 00101010 |
| 1 | 0 | 1 | 01000000 | 00000101 | 01010100 | 00010101 |
| 1 | 1 | 0 | 10000000 | 00000010 | 10101000 | 10001010 |
| 1 | 1 | 1 | 00000000 | 00000001 | 01010001 | 01000101 |

**Example to check the output of 2 bit right rotate operation:**
The initial 8-bit input to the barrel shifter is set as:
**Input: 10100010**
Starting from D7 to D0
Input becomes 01000101
Control Signals:

Rotate 2-bit pin = 1
2-bit shift pin = 1
Right shift pin = 1
s pin = 1

All other control signals are set to 0, as per the user-defined configuration.
**Step 1:**
**Initial Reverse Stage:**
At the first stage, the input data is reversed. This yields the output as 10100010.
**Step 2:**
**4-Bit Rotate Operation:**
The barrel shifter processes a 4-bit rotation, which, in this context, does not introduce any additional shift. Therefore, 4-bit rotate operation output is 10100010.
**Step 3:**
**2-Bit Rotate Operation:**
The next step performs a 2-bit rotate operation on the 4-bit rotated output. After shifting, 2-bit rotate operation output is 10101000.
**Step 4:**
**1-Bit Rotate Operation:**
A 1-bit rotation is applied, which does not further change the output in this configuration. Thus, the 1-bit rotate output is 10101000.
**Step 5:**
**Final Reverse Stage:**
Final Output:00010101

The final output 00010101 is available from Y7 to Y0. Hence the final output becomes 10101000 which matches the expected result, thus verifying the correct functionality of the barrel shifter design in performing a 2-bit right rotate operation. This demonstrates the accuracy of the circuit and the effectiveness of the multiplexer configuration in handling various bit-shift and rotate operations.

## 5. Results And Discussion

The entire 8-bit barrel shifter is simulated using the Cadence Virtuoso 6.1, and its output is as per the expectation.

In the design verification process, hardcoded values are given to simulate user inputs. Through waveform analysis, the functionality of the designed barrel shifter validates its expected operational behavior across all test cases.

The input to the circuit is 8-bit binary data, and the output waveforms of operations are shown from Figure 5 to Figure 8 as follows:

Input for the designed circuit: 10100010

(a) Figure 5 shows 1-bit right shift operation. It gives the output 01010001.

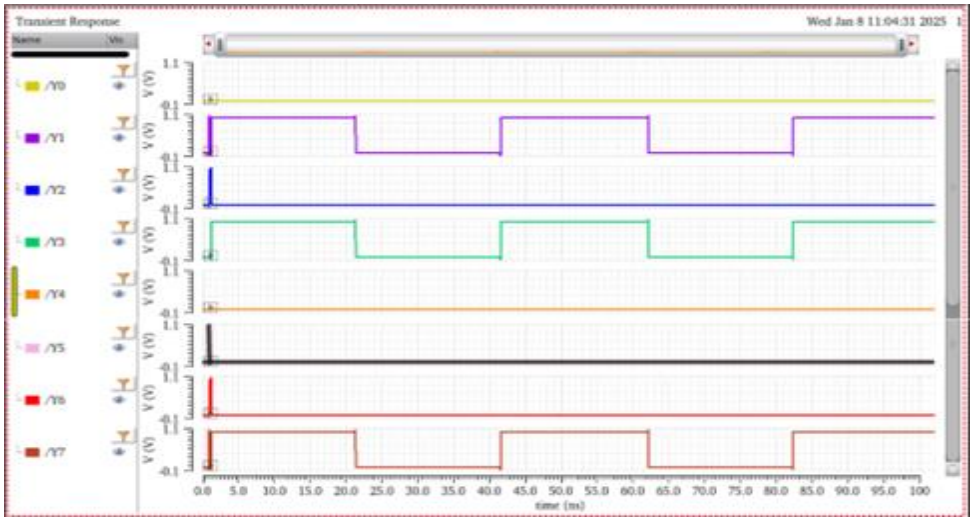


**Fig. 5** Output waveform of 1-bit right shift

(b) Figure 6 shows 2-bit right rotate operation. Output of this is 10101000.

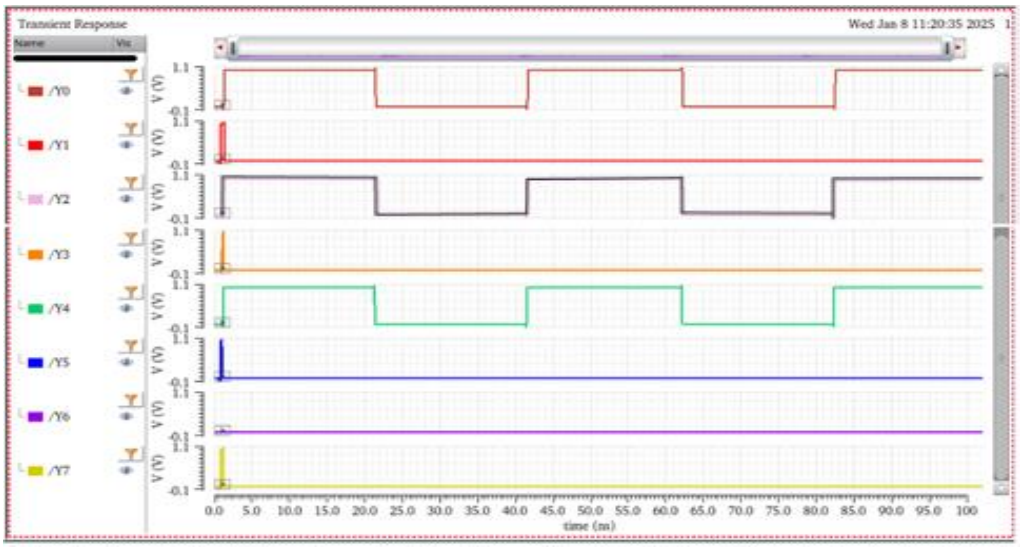


**Fig. 6** Output waveform of 2-bit right rotate

(c) Figure 7 shows 4-bit left rotation operation. Data cyclically shifts to the right, preserving all bits. The output for this operation is 00101010.

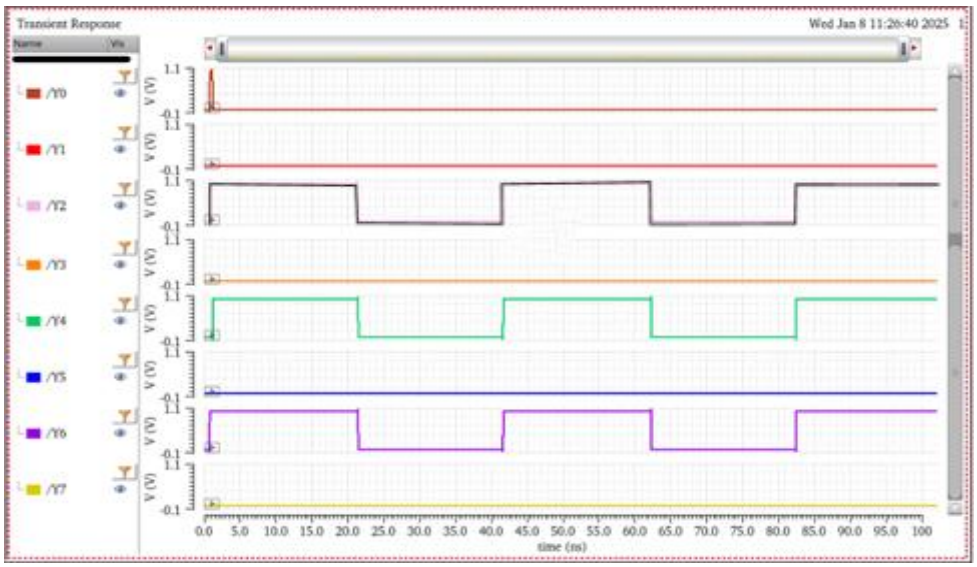


**Fig. 7** Output waveform of 4-bit left rotate

(d) Figure 8 shows 5-bit left shift operation. Data serially shifts to the left. The output for this operation is 01000000.

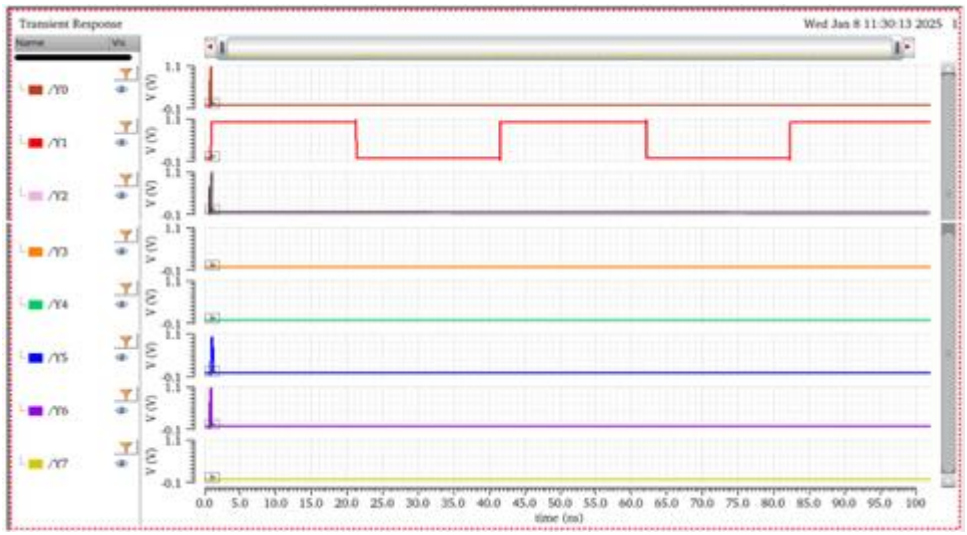


**Fig. 8** Output waveform of 5-bit left shift

### 5.2. Performing Multiple Operations Simultaneously:

Figure 9 illustrates the inputs for multiple operations that can be performed simultaneously.

Consider the following operations in the given sequence on the input 10100010.

**Operation 1**: 4-bit Left Rotate
**Operation 2**: 2-bit Right Rotate
**Operation 3**: 5-bit Left Shift
**Operation 4**: 1-bit Right Shift

Each control signal corresponds to a specific net, and the states of these signals dictate which operations will be performed on the input data.

The control signals are as follows:

s (net8): A signal that likely controls whether the operation is a shift or rotation. It is configured to be 1100.
Right (net12): A signal to initiate a right shift. It is configured as 0101.
Rotate 1-bit (net13): A signal to perform a 1-bit rotation. It is configured as 0000.
Rotate 2-bit (net15): A signal to perform a 2-bit rotation. It is configured as 0100.
Rotate 4-bit (net16): A signal to perform a 4-bit rotation. It is configured as 1000.
1 Bit (net17): A signal that controls a 1-bit shift. It is configured as 0011.
2 Bit (net1): A signal that controls a 2-bit shift. It is configured as 0101.
4 Bit (net2): A signal that controls a 4-bit shift. It is configured as 1010.

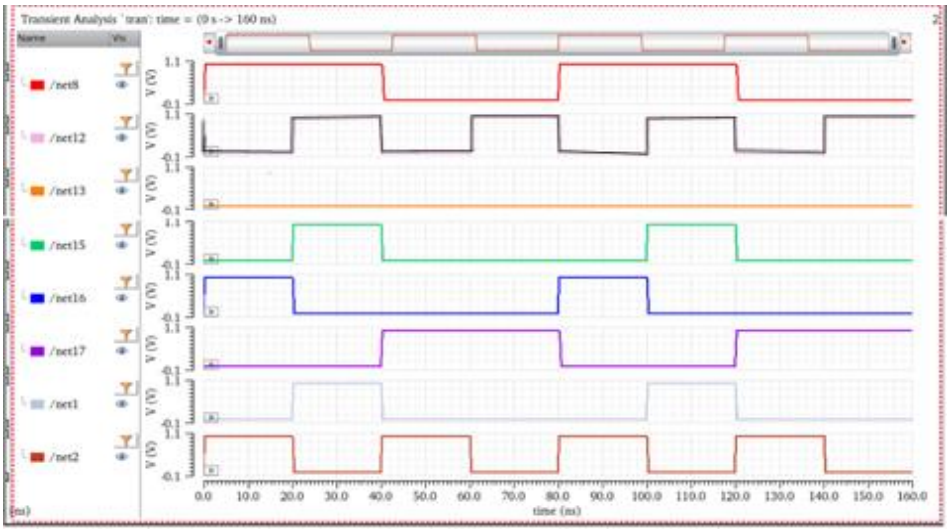


**Fig. 9** Input waveform for all operations

The resulting waveform, shown in Figure 10, illustrates the activation timing of each signal and how these control the behavior of the shift and rotate operations over time. Each pulse corresponds to when a specific operation is applied to the input. The timing parameters are set to perform these operations simultaneously.

All control signals are associated with parameters such as pulse width, period and delay, which define the timing characteristics of the operations. The pulse width specifies

how long a signal remains active, the period indicates the interval between successive activations, and the delay determines the time between signal activation and the start of the operation. The system clock is used for simulation. The timing parameters are uniformly applied to all signals to ensure synchronized execution, allowing multiple shifts or rotate operations to be performed simultaneously on the input data.

This can be illustrated with an example as follows:
net8 (s) has a pulse width of 40 ns, a period of 80 ns, and no delay.
net12 (Right) has a pulse width of 20 ns, a period of 40 ns, and 20 ns delay.
net13 (Rotate_1_bit) has a pulse width of 40 ns, a period of 80 ns, and no delay.
net15 (Rotate_2_bit) has a pulse width of 20 ns, a period of 80 ns, and no delay.
net16 (Rotate_4_bit) has a pulse width of 20 ns, a period of 80 ns, and no delay.
net17 (1 bit) has a pulse width of 40 ns, a period of 80 ns, and 40 ns delay.
net1 (2 bit) has a pulse width of 20 ns, a period of 80 ns, and 20 ns delay.
net2 (4 bit) has a pulse width of 20 ns, a period of 40 ns, and no delay.

Given an input data of **10100010**, the control signals perform a series of shift and rotate operations. The final output is obtained by combining the results of the following four operations:
**Operation 1:** 4-bit left rotate → Output: 00101000
**Operation 2:** 2-bit right rotate → Output: 01000101
**Operation 3:** 5-bit left shift → Output: 01000000
**Operation 4:** 1-bit right shift → Output: 01010001

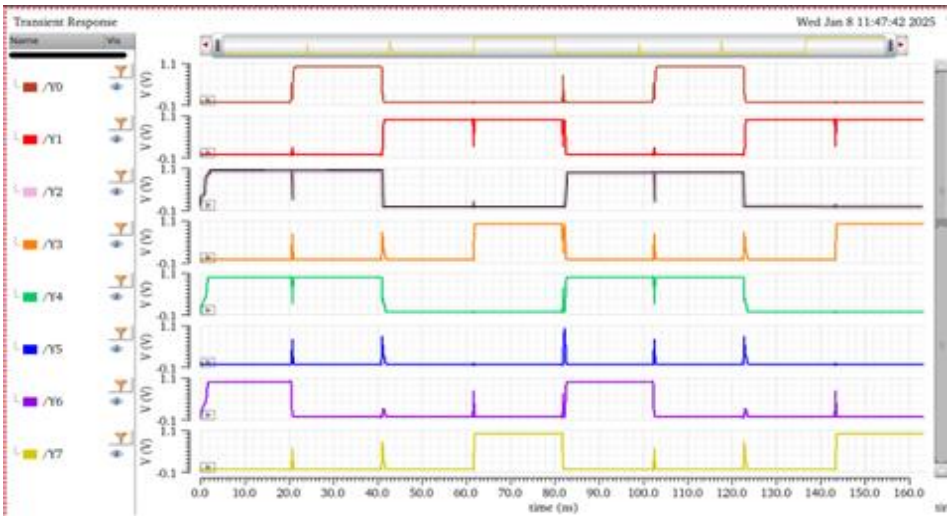


**Fig. 10** Output waveform of all operations

### 5.3. Power dissipation using 45 nm and 180 nm technology:

The power dissipation comparison for multiplexer implementation is given in Table 3. Transmission gate (TG) multiplexers generally consume less power than logic gate-based multiplexers due to lower switching activity and reduced leakage, especially in smaller nodes like 45 nm. TG Multiplexers are ideal for energy-sensitive applications, while logic

gate-based Multiplexers, though higher in power, offer robustness and simplicity, making them suited for high-reliability environments like 180 nm technology.

**Table 3** Power dissipation comparison for Multiplexer implementation

| | 45 nm Technology | 180 nm Technology |
|---|---|---|
| Mux using logic gates | 125.5 nW | 227.3 nW |
| Mux using Transmission Gate | 10.1 nW | 17.5 nW |

The power dissipation of the integrated circuit using two techniques with left/right shifter and Left/Right rotator is given in Table 4. It reveals that 45 nm transmission gates consistently achieve the lowest power consumption across all tested operations, whereas 180 nm standard logic gates exhibit the highest power usage. Consequently, 45 nm technology is recommended for applications with stringent power efficiency demands, while the 180 nm node is advantageous for scenarios requiring enhanced robustness and cost-effectiveness.

**Table 4** Power dissipation comparison for various operations of Barrel Shifter

| | Power dissipation using standard logic gates (45 nm) | Power dissipation using standard logic gates (180 nm) | Power dissipation using transmission gates (45 nm) | Power dissipation using transmission gates (180 nm) |
|---|---|---|---|---|
| 1-bit right shift | 720.1 nW | 1212 nW | 185.1 nW | 285.7 nW |
| 2-bit right rotate | 925.7 nW | 1714 nW | 237.7 nW | 358.7 nW |
| 4-bit left rotate | 497.1 nW | 969.7 nW | 124.2 nW | 195.6 nW |
| 5-bit left shift | 390.7 nW | 579.9 nW | 98.67 nW | 148.5 nW |

Comparison of existing and proposed barrel shifter is given in Table 5; the present work proposes a CMOS-based 8-bit barrel shifter using transmission gates and standard logic to achieve similar functional versatility with improved performance in conventional digital platforms.

**Table 5** Comparison of Existing and Proposed Design

| Feature | Existing Design (Reversible Logic-Based Design) | Proposed CMOS Design |
|---|---|---|
| Technology | 45 nm CMOS | 45 nm & 180 nm CMOS |
| Architecture Type | Reversible Logic | CMOS with Transmission & Logic Gates |
| Gates Used | Feynman, Fredkin, RLM | 2:1 Multiplexers (TGs + Std. Logic) |
| Power Consumption | 16.39 µW | 0.185–0.925 µW (TG Mux of 45nm, by operation) |

Barrel shifters have evolved through various architectures aimed at optimizing power, area, and versatility. One significant contribution is the Universal Reversible Barrel Shifter proposed in [7], which uses reversible logic gates such as Feynman, Fredkin, and the novel Reversible Logic Multiplexer (RLM) gate to enable bi-directional shifting and rotation in a low-power configuration.

A trade-off exists between shift and rotate operations in terms of power consumption. The rotate operations demand more power than shift operations due to the higher number of multiplexers required for bit rotation. Right rotations, in particular, incur additional power consumption since data reversal necessitates more complex multiplexer arrangements.

## 6. Conclusion and Future Scope

A compact and efficient 8-bit barrel shifter has been developed using CMOS technology, integrating left/right shift and rotate operations within a unified hardware framework. The architecture of barrel shifter employs multiplexers based on standard logic and transmission gates, enabling flexible directional control and efficient data manipulation. Implemented at both 45 nm and 180 nm technology nodes, the design achieves significantly lower power consumption with transmission gate-based configurations. The minimum power consumption of 98.67 nW is observed for a 5-bit left shift operation at 45nm technology node.

In comparison, existing reversible logic-based designs using gates like Feynman and Fredkin exhibit higher design complexity and consume up to 16.39 μW, resulting in a power reduction of up to 99.4% in our proposed design. This demonstrates that the MUX-based CMOS approach is not only more practical for current VLSI systems but also highly energy-efficient for power-constrained applications in DSP and embedded systems.

Future enhancements will focus on expanding support to higher bit widths and adopting FinFET technology for further improvements in power and area optimization. The results of the design proposed in this work are obtained through simulation. Transmission gates used in multiplexers have limitations like increased delay and complexity in designing complex logic. The analysis regarding latency can be the future scope of this work.

**Acknowledgement**: *This work was supported by the Chips to Startup (C2S) grant from the Ministry of Electronics and Information Technology (MeitY), Government of India.*

## References

[1] A. Kumar Adepu, "Design of A Low-Power 4-Bit Barrel Shifter for A Current Steering DAC using Optimized Multiplexers in 65 nm Technology", *J. Electron. Eng.* (*JELE)*, vol. 13, no. 3, pp. 11-22, 2023.

[2] S. Shah, L. Khan and V. Kumar Maurya, "Designing of Low Power GDI Based 8-Bit Barrel Shifter", *J. Emerg. Technol. Innov. Res.* (*JETIR)*, vol. 5, no. 9, pp. 397-400, Sep. 2018.

[3] P. Shrivastav, *Completion Detection in Asynchronous Circuits*, Springer, Sep. 2022, pp.45-58.

[4] J. Sankar Sahoo and N. Kumar Rout, "Comparative Study on Low Power Barrel Shifter/Rotator at 45nm Technology". *Int. J. Adv. Eng. Nano Technol. (IJAENT*), vol. 2, no. 6, pp. 11-18, May 2015.

[5] M. Wajid Khan, V. Vannammal Revathy and S. Kaja Mohideen, *Nanoelectronics, Circuits and Communication Systems*, Springer Nature Publication, Singapore, Jan. 2021, pp. 727-737.

[6] P. Ashok Babu, V. Siva Nagaraju, R. Mariserla and R. Ratna Vallabhuni, "Realization of 8 x 4 Barrel Shifter with 4-bit Binary to Gray Converter using FinFET for Low Power Digital Applications", *J. Phys.: Conf. Ser.,* vol. 1714, pp.1-21, Oct. 2020.

[7] H. V. Ravish Aradhya, J. Lakshmesha J and K. N. Muralidhara, "Design Optimization of Reversible Logic Universal Barrel Shifter for Low Power Applications"*, Int. J. Comput. Appl.*, vol. 40, no. 15, pp. 26-34, Feb. 2012.

[8] S. Kotiyal, H. Thapliyal and N. Ranganathan. "Design of a Reversible Bidirectional Barrel Shifter", In Proceedings of the 11th IEEE International Conference on Nanotechnology (IEEE NANO), Portland, Oregon, August 2011, pp. 463-468.

[9] R. Verma and R. Mehra, "Area Efficient Layout Design Analysis of CMOS Barrel Shifter", *Int. J. Sci. Res. Eng. Technol. (IJSRET)*, vol. 2015, pp. 84-89, Mar. 2015.

[10] S. Gavali and Mr. M. B. Neelagar, "Low Power Barrel Shifter Design Using Complementary and Pseudo-NMOS Logic", *J. Comput. Sci. Mobile Comput. (IJCSMC)*, vol. 7, no. 12, pp.163-166, Dec. 2018.

[11] D. Shakya and S. Agrawal, "Design and Analysis of 4-bit Barrel Shifter using Nano-scale CMOS Technology", *J. Emerg. Technol. Innov. Res.* (*JETIR)*, vol. 5, no. 12, pp. 177-183, Dec. 2018.

[12] M. J. Schulte, E. George Walters III and M. R. Pillmeier, "Design Alternatives for Barrel Shifters", In Proceedings of SPIE Optics + Photonics, Bethlehem, USA, 2002, pp. 11-45.

[13] S. Kotiyal, H. Thapliyal and N. Ranganathan. "Design of a Ternary Barrel Shifter Using Multiple-Valued Reversible Logic". In Proceedings of the 11th IEEE International Conference on Nanotechnology (IEEE NANO), Seoul, Korea, August 2010, pp 1104-1108.

[14] P. Agrawal and Dr. R. Mehra, "Design and Performance Analysis of Barrel Shifter using 45nm Technology", *IOSR J. VLSI Signal Process.* (*IOSR-JVSP)*, vol. 6, no. 3, pp. 38-44, 2016.

[15] O. Daesun and K. K. Parhi, "Area Efficient Controller Design of Barrel shifters for Reconfigurable Low Density Parity Check (LDPC) Decoders", In Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS), Seattle, WA, USA, 2008, pp. 240-243.

[16] D. Bhadouriya and S. Agrawal, "A Review on Barrel Shifter", *Int. J. Eng. Dev. Res. (IJEDR)*, vol. 7, no. 1, pp.241-244, Feb. 2019.