

Novel Approaches to the Design and Optimization of Universal Shift Register with Reversible Logic

Deepanshu Joshi*, Aman Pal, Dev Vrat Goswami, Kamal Kishor Upadhyay, Shivaji Sinha, Manoj Joshi

Abstract

Reversible logic is gaining importance in new multi-disciplinary fields, such as quantum computing, low-power computing, and even nanotechnology. The study of reversible gates has quickly evolved as a distinctive area of research as a result of the new developments aimed at fully utilizing resources and performing operations without any loss of information. With chip integration, these USRs, or Universal Shift Registers, have now become fundamental sequential components needed in various prototypical digital systems. In this research work, we employed an awesome configuration using an efficient D flip flop structure: Fredkin Gate and Feynman Gates, alongside a multiplexer design made of standard Fredkin gates, and we have designed and realized this design really well for an optimized 4-bit reversible universal shift register, or 4-bit RUSR. Using necessary metrics, including things like garbage output counts (GO), gate counts (GC), and quantum cost (QC), we are showing that we have a really efficient design. We have achieved the minimum quantum cost, and we also went with fewer reversible logic gates, which increases the efficiency of our design, which is optimized for resources. Validation and simulation of the reversible 4-bit universal shift register circuit were carried out using Xilinx Vivado 2024.2 software.

Keywords: Quantum cost (QC), garbage outputs (GO), gate count (GC), reversible logic, universal shift register (USR), fredkin gate, feynman gate, ancilla inputs (AI)

INTRODUCTION

Reversible computing is a paradigm shift from traditional irreversible computation, motivated primarily by its capability to overcome Landauer's principle defined limits of energy dissipation [1] as well as its inherent requirement for quantum algorithm realization [2, 3]. The development of quantum computation and ultra-low-power electronics is advancing quickly, and the possibility for reversible logic to provide efficient implementations of digital logic primitives is becoming more and more crucial. The most important among them is the Universal Shift Register (USR), which is primarily the versatile workhorse for data manipulation, serial-parallel conversion, and the arithmetic logic unit.

*Author for Correspondence

Deepanshu Joshi

E-mail: deepanshujoshi2002@gmail.com

Research Scholar, Department of Electronics and Communication Engineering, Jagadguru Sri Shivarathreshwara Academy of Technical Education, Noida, Uttar Pradesh, India

Received Date: June 05, 2025

Accepted Date: July 31, 2025

Published Date: December 13, 2025

Citation: Deepanshu Joshi, Aman Pal, Dev Vrat Goswami, Kamal Kishor Upadhyay, Shivaji Sinha, Manoj Joshi. Novel Approaches to the Design and Optimization of Universal Shift Register with Reversible Logic. Journal of VLSI Design Tools & Technology. 2025; 15(3): 21–28p.

Reversible logic gates, as they allow that each input state can be uniquely retrieved from the output state, provide the basis for building quantum circuits that are not only energy efficient, but also can manage complicated computational tasks [4-8]. Reversible logic is important not only in quantum computing but also in areas such as nanotechnology and optical computing. In such fields, the priority is to generate the least amount of heat and consume the least amount of energy, as devices keep getting smaller. Reversible gates allow building circuits that produce negligible heat, thereby improving the durability and reliability of nanoscale devices.

In addition, in optical computing, where light is used to transmit and process data [9–13], reversible gates achieve significantly lower energy dissipation, resulting in faster and less energy-consuming optical circuits [14]. Reversible logic gates are primarily used in designing efficient arithmetic circuit adders, and multipliers are core building blocks of digital signal processing (DSP) and several computational systems. Such adder circuits traditionally produce a high amount of garbage outputs, meaning needless intermediate values that are not involved in the outcome [15–18]. The occurrence of these outputs not only consumes computational resources but also adds to the circuit complexity and power consumption.

Although reversible implementation of fundamental arithmetic circuits has been widely explored, achieving an efficient implementation of more complex sequential elements such as Reversible USRs (RUSRs) through the use of the Reversible SoC framework remains a non-trivial task. The efficiency of reversible circuits is widely accepted to be measured in the following ways: Quantum Cost (QC) which accounts for any overhead in terms of primitive quantum gates, the Gate Count (GC) denoting the total number of basic reversible gates and Garbage Output (GO) gates which do not contribute to the primary logic function but are required in order to remain reversible [4]. A survey of previous attempts at RUSR demonstrates that many of them are afflicted by dependency on custom gate structures or generally [5–8].

Suboptimal architectural choices that increase overall resource overhead, in the form of QC, GC, or GO, thus limit their viability in larger integrated reversible systems. Reversible logic gates help mitigate these problems to a certain extent by reducing the number of garbage outputs and using fewer ancilla inputs, which are extra input bits required in a computation to hold the value needed to maintain reversibility [19–21]. In contrast to digital design, where area, delay, and power serve as fundamental metrics, reversible circuit evaluation is based upon a different set of cost parameters, which are indispensable for determining the realizability and efficiency of reversible circuits, particularly in quantum implementation:

Gate Count (GC): The total number of primitive reversible logic gates (e.g., Feynman, Fredkin, Toffoli, Peres gates) that compose the circuit implementation. Lower GC typically means simpler topology and possibly also lower implementation complexity, although it is not always directly reflective of lower quantum resource usage.

Garbage Outputs (GO): There must be an equal number of inputs and outputs for the reversible gates. GO denotes those output lines not used as primary outputs or input to subsequent gates in the intended logic path. These need to be output just to keep the gate reversible. Matter (GO) has its core in GO because each output waste GO involves a consumption of resources (e.g., a qubit wire on a quantum circuit) that must be properly terminated or handled otherwise, incurring complexity.

Quantum Cost (QC): Perhaps the most important metric for circuits meant for quantum computation. QC measures how complex a reversible gate or a circuit is by the minimum number of primitive 1×1 (NOT) and 2×2 (CNOT/Feynman, V , V^\dagger) quantum gates needed to realize it [4]. The QC of simple gates like the Feynman (CNOT) gate is $QC=1$, whereas more complex gates like the Fredkin gate have $QC=5$. Reducing QC can then directly result in a reduction of the execution time and an increase in fault tolerance of any quantum algorithm applied through the circuit.

Ancillae Inputs (AI): Ancillae (or Ancilla) Inputs are the extra input bits needed in a reversible logic circuit to help in the computation. Reversible design is a term widely used, and these supplementary inputs are required for the development of this reversible design and are used to perform some operations, such that the reversibility of the circuit is not compromised.

REVERSIBLE GATES

Reversible Not Gate

The NOT gate is the simplest reversible gate with 1×1 or 1 input and 1 output, which can be used for performing inversion of inputs. The gate is quantum cost 0, unit delay. Figure 1 shows the working of the NOT gate and its quantum representation.

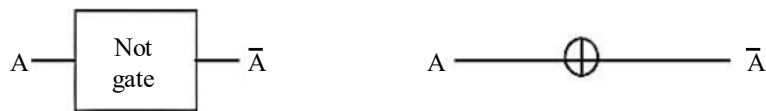


Figure 1. (a) NOT gate, (b) NOT gate quantum implementation.

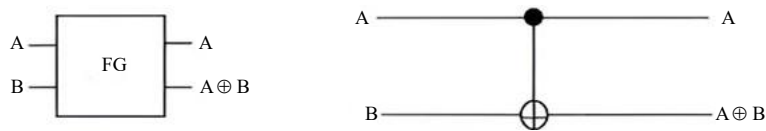


Figure 2. (a) Feynman gate, (b) Quantum implementation of Feynman gate.

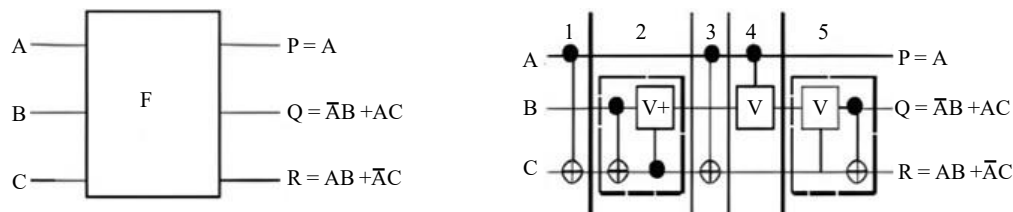


Figure 3. (a) Fredkin gate, (b) Quantum implementation of the Fredkin gate.

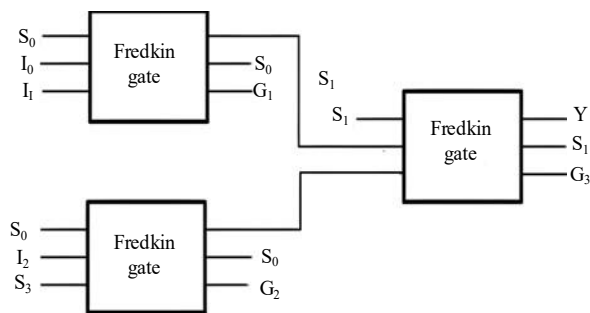


Figure 4. Reversible 4×1 Mux.

CNOT or Feynman Gate (FG)

It is a 2-input, 2-output (2×2) reversible logic gate. The CNOT gate (also referred to as the FEYNMAN gate) is widely used to get around the barrier of fan-out since it can copy the information as well. Unit delay and unit quantum cost for the CNOT gate are shown in Figure 2.

Fredkin Gate

A Fredkin gate is a three-input three-output circuit or device that passes the first bit through unchanged, and swaps the last two bits if and only if the first bit is 1. The quantum cost of a Fredkin gate is 5, as shown in Figure 3.

PROPOSED DESIGN

In this section, the architectural design of the proposed optimized 4-bit RUSR (Reversible Universal Shift Register) is discussed. The design focuses on balancing everything with just the use of standard, efficient reversible logic gates, the Feynman gate, and the Fredkin gate, in all of the components. First, we focus on the reversible implementations of the key building blocks, the 4×1 Multiplexer and the D Flip-Flop, and then demonstrate their interfusing to form the final RUSR configuration.

Reversible 4×1 Multiplexer (RMUX): RMUX, which has been represented conceptually in Figure 4, is introduced based on three standard Fredkin gates (FRG) connected in the same way as in a two-stage tree structure. The control inputs (SEL(1) and SEL(0)) are selected between the four data inputs (I0, I1, I2, I3). It means that the proposed RMUX circuit takes 4 Data inputs, 2 Select inputs, and 2 ancilla inputs, which are just constant '0's, thus a total of 8 inputs. It creates the main useful output

Y, expressed logically as $Y = SEL(1)' SEL(0)' I_0 + SEL(1)' SEL(0) I_1 + SEL(1) SEL(0)' I_2 + SEL(1) SEL(0) I_3$. In addition, it produces four critical garbage outputs (GO) (GO1-GO4 in the model) and other pass-through outputs (O0-O3 in the model) necessary to preserve reversibility, for 9 total output lines.

Efficient Reversible D Flip-flop (RDFF): In this design, an efficient reversible D flip-flop (RDFF) for sequential circuits is used. This setup, seen functionally in Figure 5, is a conceptual block diagram, which consists of one Fredkin gate (FRG) and two Feynman gates (FG), and has a low quantum cost (QC=7), as reported by the conceptual block diagram. This RDFF senses data (D) at the positive edge of the clock signal (CLK) and has a synchronous active-high RESET input. If RESET is high at the clock edge, it forces output Q to 0. On the next positive edge, Q becomes D and remains so until the next positive edge. Thus, the intended RDFF has three inputs (CLK, D, RESET) and three outputs: 1 main data output (Q) and 2 GO (garbage outputs) (GO1, GO2 from the model).

Universal Shift Register: The control logic decides the working mode of their register depending upon control signals SEL(1:0), which specify the instructions of left shift, right shift, parallel load, and hold. Data inputs refer to the ability to load multiple bits at the same time (parallel), while the serial inputs are used for when data is entered one bit at a time during a shift operation. The output lines (Q(3:0)) show the present state of each bit memory available in the register. A clock input (CLK) controls the shifting and loading process, so any data at the flip-flops will only change with a positive clock edge. Also, note that a clear/reset input (RESET) is provided to set the flip-flops to a known state (0000), thus allowing us to easily reset the register.

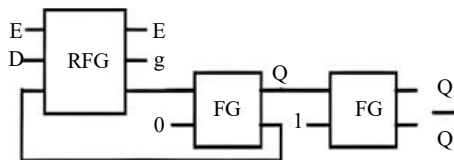


Figure 5. Reversible D flip-flop.

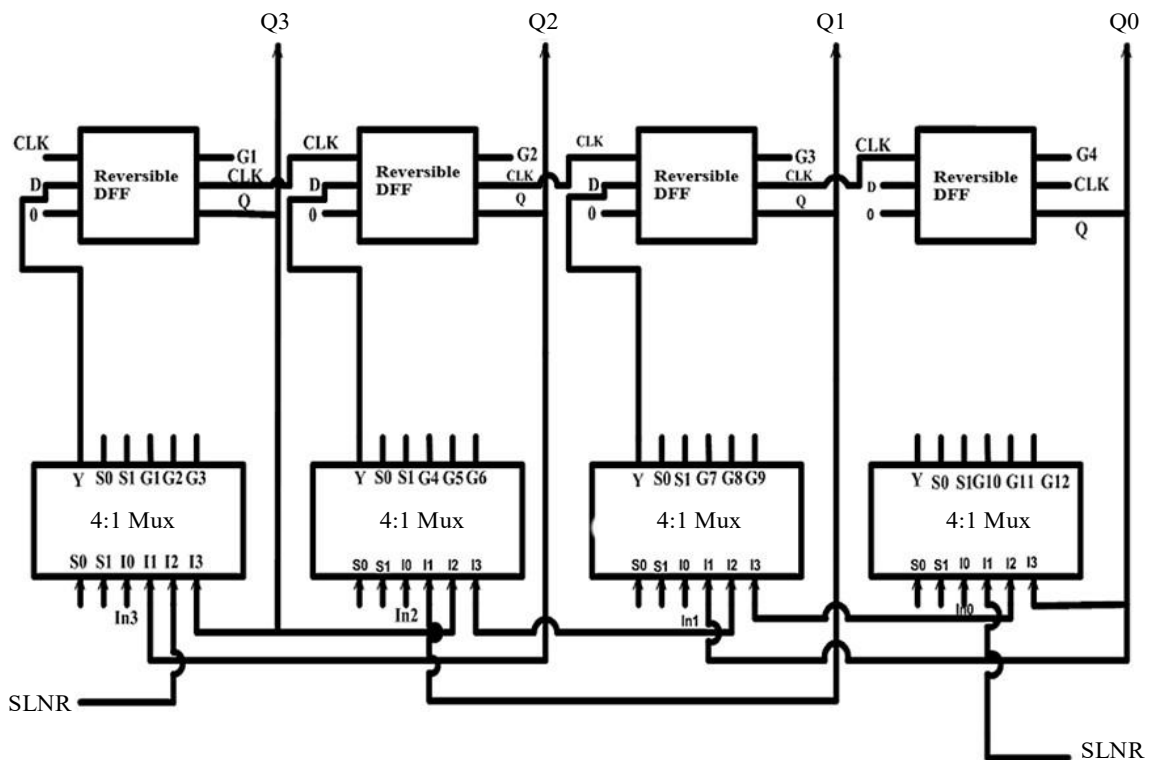


Figure 6. Reversible universal shift register..

Table 1. Functional characteristics of reversible USR.

S1	S0	Output
0	0	No Change
0	1	Right Shift
1	0	Left Shift
1	1	Parallel Load

Functional characteristics of the proposed Reversible 4-Bit Universal Shift Register (RUSR) are given in Table 1. According to Table 1, there exists a condition (SEL=00) that enables D inputs of RDFFs to take the present state of the register when both SEL(1) and SEL(0) are held low. This configuration creates a feedback loop from various RDFF outputs (Q(i)), to their respective MUX input (D(i)). This loop is used so that the next clock pulse uses the same value as the last clock pulse to preserve the state. When SEL=11 (both SEL(1) and SEL(0) high), the information present on the parallel input lines D(3:0) is loaded into the register. It happens when both control lines are in a high state. Following the example shown in Figure 6, depicting the RUSR schematic, the suggested shift register is constructed using four effective RDFFs and four reversible 4×1 multiplexers.

The illustration above represents a 4-bit RUSR architecture that contains D flip-flops and control logic realized by multiplexers. Only reversible gates are employed. Evaluating performance may include, for example, assessing the quantum cost, complexity of the hardware, latency (optional: remove if no longer evaluated), and others. The RUSR's low quantum cost has benefits in quantum computing, low-power digital systems, and other fields where minimal information loss is critical.

RESULT ANALYSIS

To evaluate the proposed optimized 4-bit reversible universal shift register (RUSR), which was designed from regular reversible logic gates (Fredkin and Feynman), Xilinx Vivado 2024.2 software was used for simulation and functional verification. To perform this evaluation, the design was implemented in Verilog HDL for the constituent components and the top-level RUSR. The main goal of this project has been to implement and design a 4-bit reversible RUSR, focusing on the utilization of commonly used low-cost modules in reversible logic. The proposed circuit uses a 4:1 Reversible Multiplexer (RMUX) circuit using three Fredkin gates and an improved Reversible D Flip-Flop (RDFF) architecture based on one Fredkin gate and two Feynman gates. These elements were chosen to structure the design only to improve performance measures relevant in reversible logic, specifically in terms of Gate Count (GC), Garbage Outputs (GO), and Quantum Cost (QC). For simulation, functional verification was carried out. Figure 7 shows the waveform of the standalone Reversible 4×1 Multiplexer, indicating that it has selected the correct data according to SEL(1:0). Figure 8 shows the induced waveform of the output D Flip-Flop for efficient implementation, confirming the positive-edge-triggered data capture and synchronous reset operation. Finally, Figure 9 shows the simulation waveform of the whole 4-bit Universal Shift Register, which operates correctly in all modes: Hold (SEL=00), Shift Right (SEL=01), Shift Left (SEL=10), and Parallel Load (SEL=11), as defined in the functional characteristics in Table 1.

Table 2. Comparison result of proposed RUSR with existing works.

Ref No.	GC	GO	QC
Ref [22]	42	31	220
Ref [23]	24	40	114
Ref [24]	48	36	110
Ref [25]	41	33	108
Ref [26]	28	40	108
Ref [27]	28	20	98
Proposed RUSR	24	24	88

We show in Table 2 a comparison of our proposed architecture against relevant existing works, as a function of standard reversible circuit cost (gate count, quantum cost, garbage outputs).



Figure 7. Simulation result of 4:1 Mux.

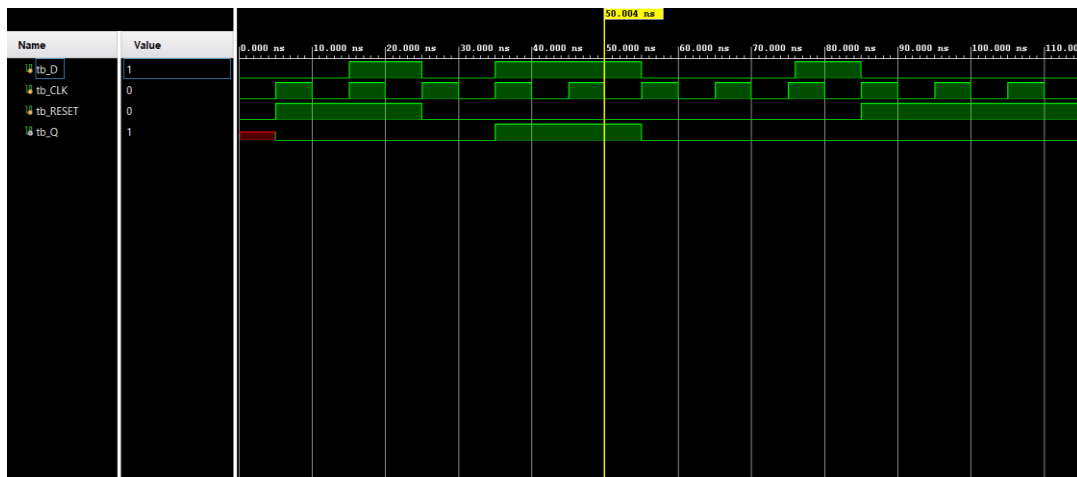


Figure 8. Simulation result of DFF.

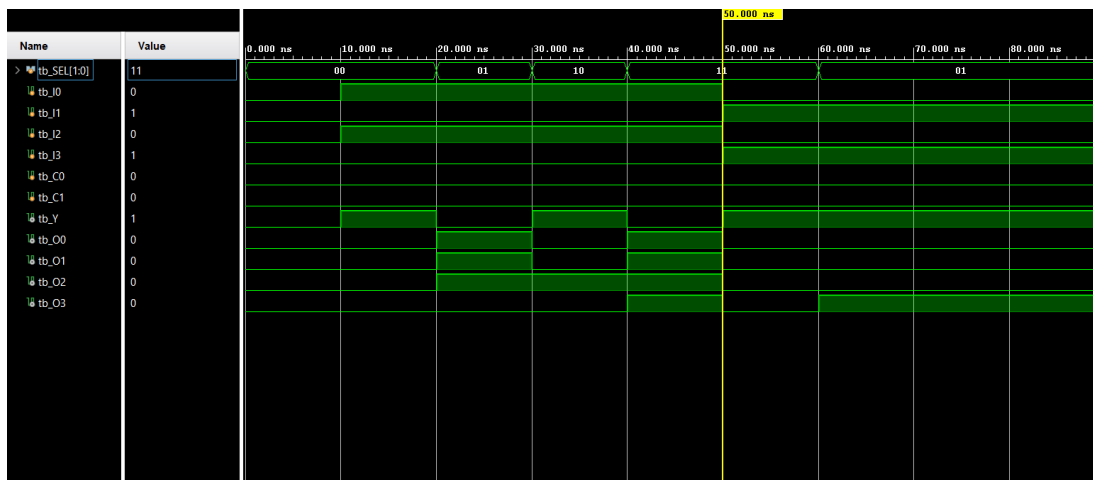


Figure 9. Simulation output of USR.

Based on Table 2, the RUSR design proposed in this material results in a Gate Count of 24, Quantum Cost of 88, and 24 Garbage Outputs. As we can see, there is a critical improvement, through Quantum Cost and Gate Count, compared to many reference designs such as (QC=98, GC=28, GO=20). Although the garbage count is somewhat larger than that reference, the dramatic reduction in QC would be an important optimization for the quantum realizations.

CONCLUSION AND FUTURE WORK

The contribution of this research work was to provide an optimized method of designing reversible universal shift registers, which were essential components of sequential logic designed for reversible and quantum computing systems.

This design is based on the intensive utilization of standard efficient reversible gates: namely, a Reversible 4×1 Multiplexer (composed of Fredkin gates), along with an efficient D Flip-Flop realized from one Fredkin and two Feynman gates, showing smaller cost metrics. We present a 4-bit Reversible-Universal Shift Register (RUSR) showing better results on key features. The architecture uses 24 reversible gates, which are followed by 24 garbage outputs, and it has an optimized quantum cost equal to 88. The quantum cost and gate count improvements are at least in part an outcome of finding the best selection of previously used low-cost primitives, which are reduced via typical architecture. With these in mind, the presented structures can be integrated into larger reversible systems for low-power computation and quantum information processing, possibly leading to more efficient sequential reversible circuit designs in the future. This proposed reversible universal shift register design has a very good future scope. Its provably lower quantum cost makes it attractive for implementation in subroutines of quantum algorithms that require data shifting or buffering. Use case applications range from reversible processor design and fault-tolerant reversible systems to task-specific applications in quantum communication protocols. Other possible avenues could be studying the N-bit scalability of this proposal, investigating state-of-the-art ancilla input/garbage output optimization methods to attempt to minimize the garbage count at no or low cost in terms of QC or GC, and estimating physical resource utilization for specific quantum computing systems.

REFERENCES

1. Roohi A, Khademolhosseini H, Dehghan M, et al. A parity-preserving reversible QCA gate with self-checking cascadable resiliency. *IEEE Trans Emerg Top Comput.* 2018;6(4):450–9.
2. Fredkin E, Toffoli T. Conservative logic. In: *Collision-Based Computing*. London: Springer; 2002. p. 47–81.
3. Toffoli T. Reversible computing. In: *Automata, Languages and Programming. Lect Notes Comput Sci. Vol 85*. Berlin: Springer; 1980. p. 632–44.
4. Bennett CH. Logical reversibility of computation. *IBM J Res Dev.* 1973;17(6):525–32.
5. Bennett CH. Notes on the history of reversible computation. *IBM J Res Dev.* 1988;32(1):16–23.
6. Sultana S, Radecka K. Reversible architecture of computer arithmetic. *Int J Comput Appl.* 2014;93(14):6–14.
7. Alioto M, Palumbo G. Modeling of power consumption of adiabatic gates versus fan-in and comparison with conventional gates. In: Rajendran J, ed. *Proc Int Workshop Power Timing Model Optim Simul*. Berlin: Springer; 2000. p. 265–75.
8. An efficient real-time embedded application mapping for NoC-based multiprocessor system-on-chip. *Wirel Pers Commun.* 2023;128:2937–52.
9. Reddy BNK, Kumar A, Rao TVKH, et al. Evaluating the effectiveness of bat optimization in an adaptive and energy-efficient network-on-chip routing framework. *J Parallel Distrib Comput.* 2024;188:104853.

10. Thapliyal H, Ranganathan N. Design of reversible latches optimized for quantum cost, delay and garbage outputs. In: Proc 23rd Int Conf VLSI Des. Bangalore, India: IEEE; 2010. p. 235–40.
11. Maity H, Saha D, Mukhopadhyay S, et al. Quantum cost optimized design of 4-bit reversible universal shift register using reduced number of logic gate. *Int J Quantum Inf.* 2018;16(2):1850016.
12. Sai Kumar A, Hanumantha Rao TVK. Scalable benchmark synthesis for performance evaluation of NoC core mapping. *Microprocess Microsyst.* 2020;79:103272.
13. Sai Kumar A, Hanumantha Rao TVK. An adaptive core mapping algorithm on NoC for future heterogeneous system-on-chip. *Comput Electr Eng.* 2021;95:107441.
14. Nayeem NM, Hossain MA, Jamal L, et al. Efficient design of shift registers using reversible logic. In: Proc Int Conf Signal Process Syst. Singapore: IEEE; 2009. p. 474–8.
15. Ali MB, Hossin MM, Ullah ME. Design of reversible sequential circuit using reversible logic synthesis. *Int J VLSI Des Commun Syst.* 2011;2(4):37–45.
16. Kumar AS, Rao KN, Sujith A, Dhanuja T, Vinay MVS. Design and implementation of 1 KB SRAM array in 45 nm technology for low-power applications. In: Proc 3rd Int Conf Adv Comput Commun Embed Secure Syst (ACCESS). India: IEEE; 2023. p. 245–50.
17. Rudra B, Mummadi S. Fundamentals of quantum computation and basic quantum gates. In: *Handb Res Quantum Comput Smart Environ.* Pennsylvania, USA: IGI Global; 2023. p. 1–24.
18. Remon P, Schilling M, Ploetz E, et al. Stochastic reversible logic: a photophysical example of a Feynman gate. *ChemPhysChem.* 2009;10(12):2004–7.
19. Rajeshwari M, Hongal RS, Shettar RB. Reversible logic-based design and implementation of 8-bit shift register. In: Proc Int Conf Curr Trends Converg Technol (ICCTCT). 2018. p. 1–4.
20. Wille R, Soeken M, Drechsler R. Reducing the number of lines in reversible circuits. In: Proc IEEE 47th Des Autom Conf (DAC). 2010. p. 647–52.
21. Mummadi S, Bhawna R. Implementation of reversible logic gates with quantum gates. In: Proc IEEE 11th Annu Comput Commun Workshop Conf (CCWC). 2021. p. 1557–63.
22. Morrison M, Ranganathan N. Design of a reversible ALU based on novel programmable reversible logic gate structures. In: Proc IEEE Comput Soc Annu Symp VLSI. 2011. p. 126–31.
23. Mamun MSA, Mandal I, Hasanuzzaman M. Design of reversible universal shift register using reversible logic. *Int J Eng Technol.* 2012;2(9):1620–5.
24. Krishnaveni D, Priya MG. A new design of reversible universal shift register with reduced delay and quantum cost. *J Comput.* 2012;4(2):164–73.
25. Anwar R, Ben JA. A novel design of reversible universal shift register. *Int J Comput Sci Mob Comput.* 2014;3(3):530–5.
26. Srinivasa Rao N, Satyanarayana P. Design and synthesis of multiplexer-based universal shift register using reversible logic. *IOSR J VLSI Signal Process.* 2017;7(3):8–13.
27. Sarangam SK, Sai Kumar A, Mosherani V, Siddhesh U, Kalyani N, Mounika D. Design and optimization of universal shift registers using novel reversible logic gates. In: Proc 5th Int Conf Electron Sustain Commun Syst (ICESC). 2024. p. 165–70.