IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 11, Issue 4, Ver. IV (Jul.-Aug .2016), PP 13-22
www.iosrjournals.org

Optimization, FPGA Implementation & CMOS Realization of Reversible Shift Registers

K.N.B.R.Gopal Tejaswi[1], Rupali Singh[2]
[1.]M.Tech , VLSI Design, SRM University, Ghaziabad, India
[2.]AssistantProfessor,SRM University, Ghaziabad, India

Abstract: *Recently reversible logic has emerged as one of the most important designs as it is known to provide zero power dissipation under ideal conditions. Reversible logic is used in the field of low power VLSI, nano technology, communications, high speed VLSI, digital signal processing etc. The optimization is carried out in reversible circuits by reducing gate count, constant inputs, garbage outputs as well as quantum cost. In this paper we are presenting the optimized designs of sequential circuits such as flip-flop and four types of shift registers in both FPGA implementation and CMOS implementation. In the cmos implementation we have used 180nm cmos technology in the design of shift registers. At final we obtained the results of power, delay and power delay product of shift registers using mosfet of 180nm technology.*

Keywords: *Reversible Logic, shift registers, Garbage Outputs, Constant Inputs.*

I. Introduction

Now-a-days as the technology improves, the complexity of the design also increases. In order to provide more number of functions the designers are integrating more number of components in the design. This leads to high power dissipation, which is the great problem in the present electronic industry. The electronic systems designed using conventional logic gates, are irreversible in nature. The information (bits) lost or erased in each computation in these systems. According to Landauer's research, the amount of energy dissipated for each irreversible bit lost is KTln2 joules, where K=1.38*10-^{23}m^{2} kg^{-2}K^{-1}(J/K^{-1}) is Boltzmann's constant and T is the absolute temperature [1]. The heat dissipation for one bit of information is small but if we consider high speed computation large number of bits are lost per second and thus significant power is dissipated and it affects the performance of the device.

In 1973, Bennett showed that KTln2 energy would not dissipate from the system as long the system reproduces the inputs form observed outputs [2] i.e. with reversible computing. Reversible logic supports this feature so the power dissipation of the circuits is less.

II. Concept Of Reversible Logic

The logic gate is called reversible if it has one to one mapping that is every distinct input has distinct output. The number of inputs and outputs of reversible logic gates are equal. For example K*K reversible gate has K inputs as well as K outputs. The important feature of reversible circuits is thatfan-out and feedback is not allowed in these circuits [9].

Parameters:

- Reversible Gates (N): The number of reversible logic gates used in the circuit.
- Garbage Outputs (GO): The numbers of outputs which are not used in the circuit are termed as garbage outputs.
- Quantum Cost (QC): The cost of the circuit in terms of primitive gates used in the circuit. Primitive gates are the basic building gates (1*1 or 2*2) used in the circuit.
- Constants Inputs (CI): The number of 1's and 0's used in the circuit in order to synthesize the given logical function.

In order to build an optimum reversible logic circuit, the parameters such as number of gates, garbage outputs, quantum cost and constant inputs should be minimum[7].

III. Basic Reversible Gates

3.1 NOT Gate:

The NOT gate is a type of 1*1 reversible gate and its quantum cost is zero. The gate with quantum implementation is shown in fig 1.

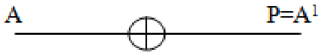

Fig.1: NOT Gate

3.2 Feynman Gate:

The Reversible 2*2 Feynman Gate has the quantum cost of 1. The inputs (A,B) are mapped to the outputs ($P=A$, $Q=A^1B+AB^1$). The Feynman Gate and its quantum implementation is shown in fig.2 and fig.3 respectively.

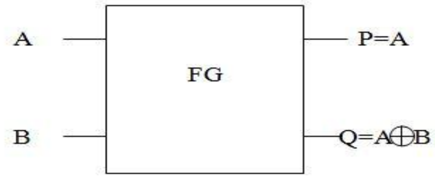

Fig.2: Feynman Gate

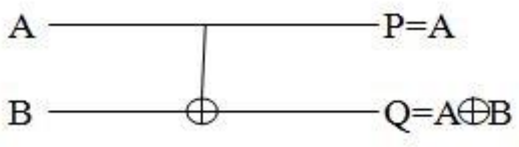

Fig.3: Quantum implementation of Feynman Gate

Transistor Implementation:

Figure 4 shows the transistor implementation of Feynman gate. The total number of transistors required here to implement Feynman gate is nine.

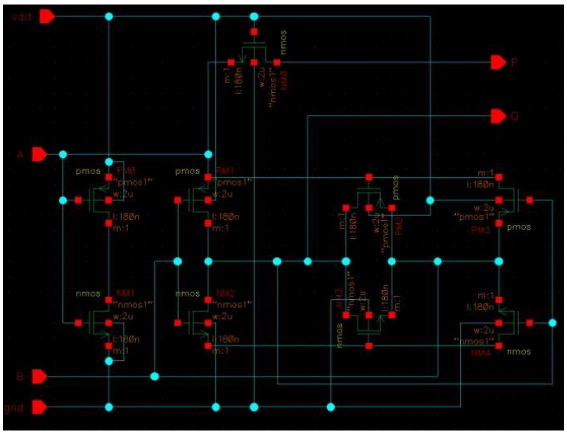

Fig.4: Transistor implementation of Feynman gate

3.3 Fredkin Gate:

Fredkin Gate is 3*3 order and the inputs (A,B,C) are mapped to the outputs ($P=A$, $Q=A^1B+AC$, $R=AB+A^1$ C). Fredkin gate is shown in fig.5

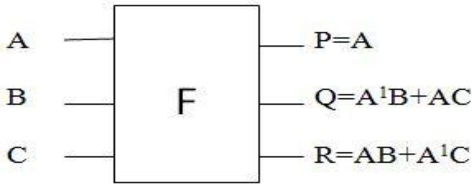

Fig.5: Fredkin Gate

The quantum cost of Fredkin Gate is 5. Quantum implementation is shown in fig.6.

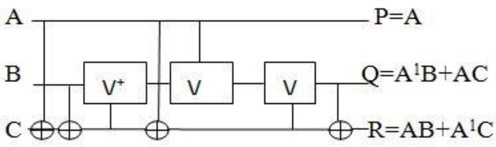

Fig.6: Quantum implementation of Fredkin Gate

Transistor Implementation:

The figure 7 shows the transistor implementation of Fredkin gate which require only four transistors. In this implementation the value of P is directly taken from A by giving gate of nmos transistor to vdd. This design is completely reversible in nature.

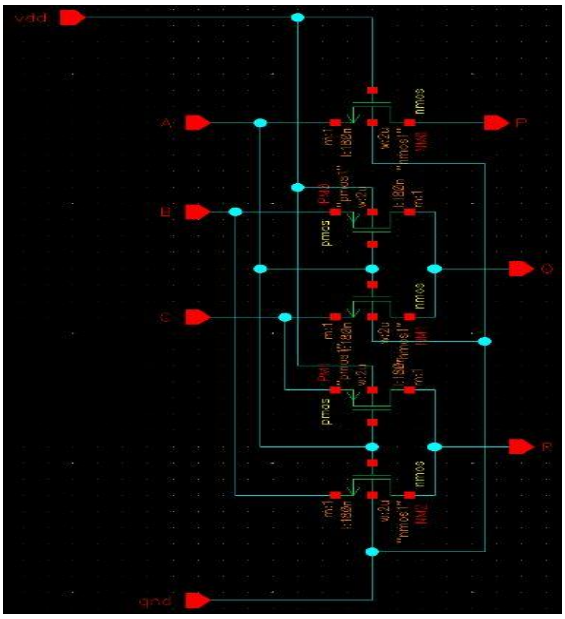

Fig.7: Transistor implementation of Fredkin gate

3.4 Sayem Gate:

The sayem gate has 4 inputs (A,B,C,D) and are mapped to the 4 outputs (P,Q,R,S). The sayem gate is shown in fig 8. The quantum cost of sayem gate is 6 [12].

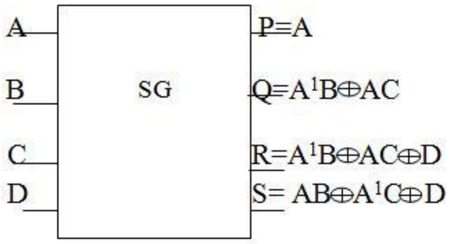

Fig.8: Sayem Gate

Transistor Implementation:

The figure 7 shows the transistor implementation of Sayem gate. By seeing the output expressions of sayem gate of fig.8 we can tell that sayem gate is combination of two Feynman gates. We have introduced two buffers in the design so as to maintain proper voltage level. The sayem gate here is completely reversible in nature i.e it works both forward as well as backward computation.

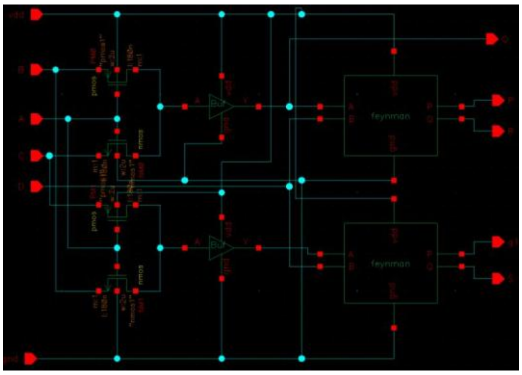

Fig.7: Transistor implementation of Sayem gate

IV. The D-Latch

We know that D latch is level triggered, whenever the enable signal is high the output is exact replica of input. The characteristic equation for D-latch is given as Q(t+1)=DE+E^1Q. This implies that when enable signal high the value of input D is transmitted to the output i.e. Q (t+1)=D. When enable signal is low the flip flop maintains its previous state i.e. Q(t+1)=Q. Here D-latch is realized by using only one sayem gate.

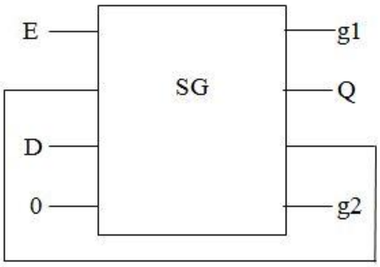

Fig.8: D-latch using sayem gate

Transistor Implementation:

The figure 9 shows D-latch using sayem gate. The symbol is created form the schematic of sayem gate. Now we have used that symbol to form a D-latch by connecting output R to input B.
Here g1 and g2 are the garbage outputs of D-latch.

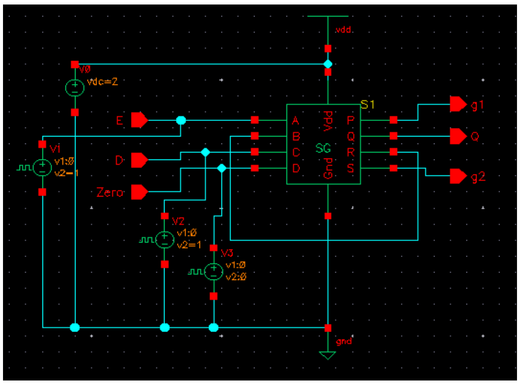

Fig.9: D-latch using sayem gate

V. Proposed Reversible Master-Slave D Flip Flop

D flip flop is defined as the data or delay flip flop because it can't make any necessary transmission unless it is enabled by a clock signal[6]. The reversible master slave D flip flop is combination of two sayem D-latch and Feynman gate. The fig.14 shows the reversible master slave D flip flop which is first presented in Thapliyal et al. in this the authors used first latch as master and the another latch as slave.

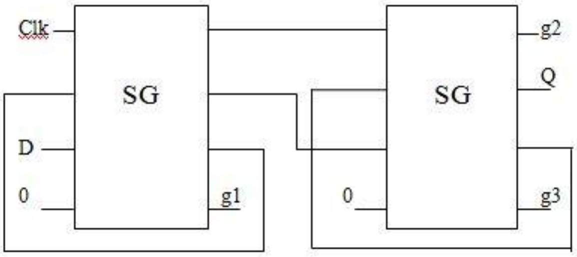

Fig.10: Reversible master slave D flip flop

5.1 Reversible Shift Registers

We have designed the reversible shift registers which are formed by cascading of reversible master-slave D-flip flops and the output of each flip flop is given to the data input to the next flip flop and due to which shifting of bits occurs for each clock pulse. Generally shift register are multidimensional in the sense that the inputs and outputs are given to the shift registers either in series or in parallel [5]. Based on the method how inputs are applied and how outputs are taken shift registers are divided into four types. **5.1 Serial In Serial Out**

Shift Register (SISO): The SISO shift register shown in fig.11. The input is applied seriallyto FF1 and output is producedserially fromthe last flipflop FF4.

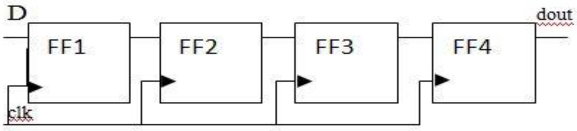

Fig.11: serial in serial out

Schematic of Siso:

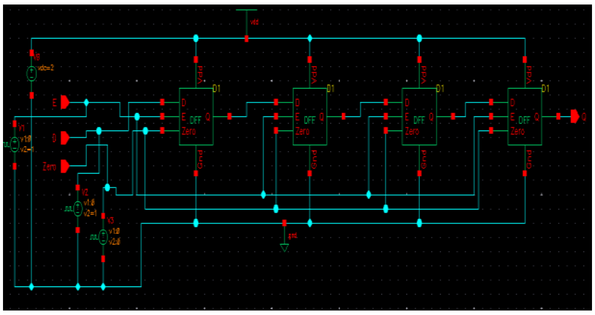

Fig.12: schematic of SISO

5.2 Serial In Parallel Out Shift Register (SIPO): The SIPO shift register shown in fig.13. The input is given serially at FF1and output is taken in parallel from each flipflop.
Here feynman gate is used to copy the output as in reversible logic the fanout is not allowed.

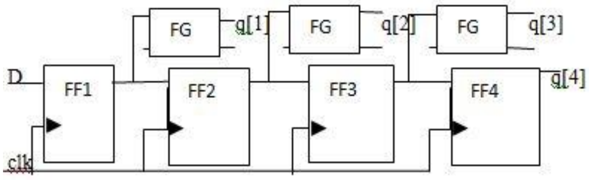

Fig.13: serial in parallel out

Schematic of Siso:

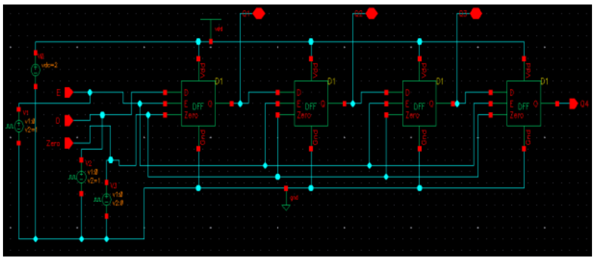

Fig.14: schematic of SIPO

5.3 Parallel In Serial Out Shift Register (PISO):

The PISO shift register is shown in fig.15. In this register the input data is applied in parallel where as the output isobtained serially from FF4.In this design the fredkin gate is used as multiplexer. The output of multiplexer is given as input to the flip flop[8].

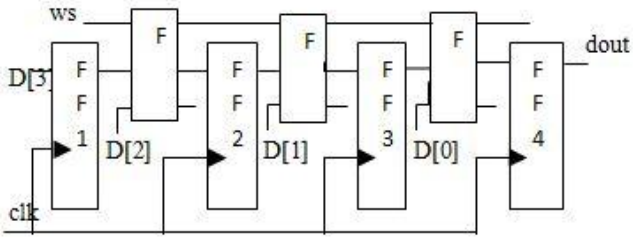

Fig.15: parallel in serial out

Schematic of Piso:

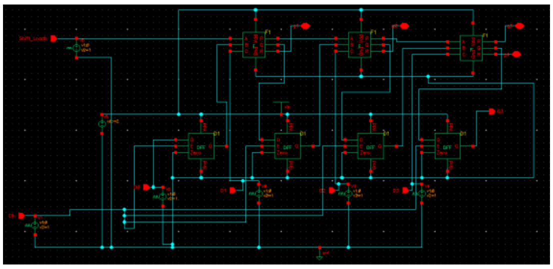

Fig.16: schematic of PISO

5.4 Parallel in Parallel out Shift Register (PIPO):

The PIPO shift register is shown in fig.18. In this circuit the input is given in parallel and the output we get is also parallel from all the flip-flops.

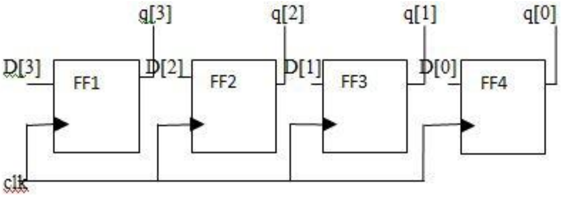

Fig.17: parallel in parallel out

Schematic of Pipo:

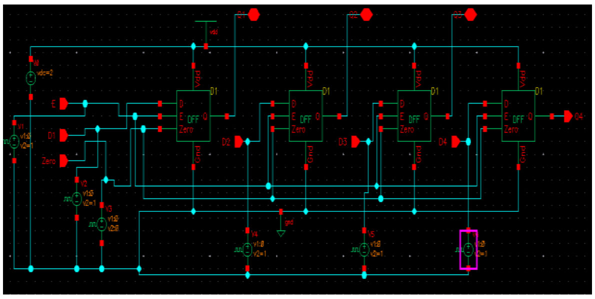

Fig.18: schematic of PIPO

Hardware complexity: It can be calculated by using
a= two input EX-OR gate calculation
b= two input AND gate calculation
c= two input OR gate calculation
d= NOT gate calculation
T= Total logical calculation i.e T=a+b+c+d.

VI. Results and discussion

Hardware complexity based onexisting [5] for the shift registers SISO, SIPO, PISO, PIPO using the Fredkin gate is T=32a+64b+24c+32d,T=32a+64b+24c+32d, T=32a+64b+24c+32d, T=32a+76b+30c+38d respectively.

Hardware complexity based on Proposed design for the SISO, SIPO, PISO, PIPO are T=32a+32b+16c+16d, T=32a+32b+16c+16d, T=32a+32b+16c+32d, T=32a+44b+22c+38d respectively.

Thus proposed design of shift registers using reversible sayem gate requires less logical operations.

Tool used:

1) Xilinx ISE Design Suite 13.2
   FPGA Kit:
   Family : Spartan 3E
   Device : XC3S100E
   Package : CP132
2) Cadence Virtuoso 6.4

Analysis of power, delay and power delay product for shift registers using mosfet of 180nm technology.

Registers	**Number of gates used**	**Delay (ns)**	**Power (□w)**	**Power delay product(J) (*e^-15)**
SISO	8	0.338	2.221	0.7506

SIPO	12	0.38(Q1)
		20.38(Q2)
		40.38(Q3)
		60.34(Q4)
			2.221	0.8439
				45.2339
				89.6839
				134.015

PISO	12	0.340	3.1	1.054

PIPO	8	0.338(Q1)
		0.338(Q2)
		0.338(Q3)
		0.338(Q4)
			2.91	0.9835
				0.9835
				0.9835
				0.9835

Comparison Table:

Designs	**Power (□ w)**			
Registers	SISO	SIPO	PISO	PIPO
Proposed	2.221	2.221	3.1	2.91
Existing[6]	339.8	339.8	367.2	441.5
Existing[5]	325.4	325.4	200.6	274.7

The comparison of power dissipation of various reversible registers is as shown in the table. It proves that power dissipation of proposed registers is much less as compared to the existing register designs. Thus the proposed work can significantly improve the performance of digital sequential circuits. Further there is a scope to improve delay and power delay product by minimizing the number of transistors.

VII. Simulation Results

6.1 Xilinx ISE results:

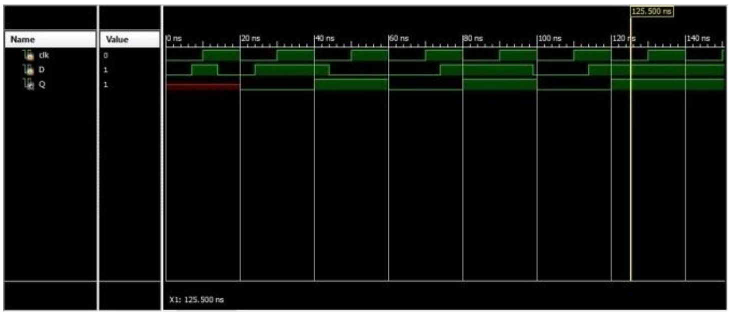

Fig.19: D flips flop

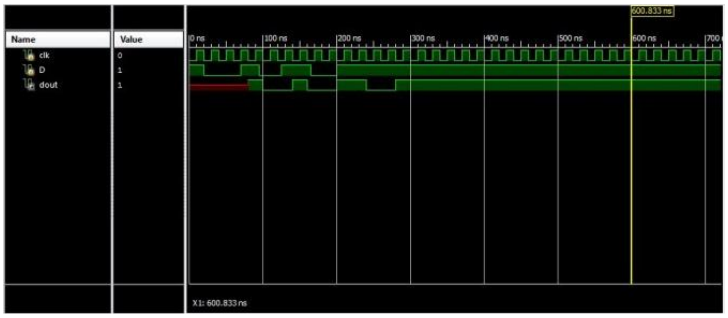

Fig.20: serial in serial out

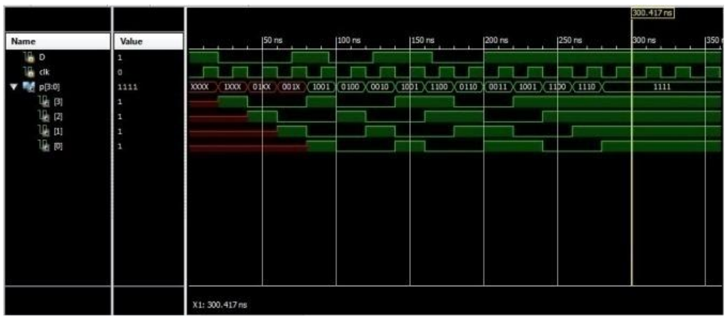

Fig.21: serial in parallel out

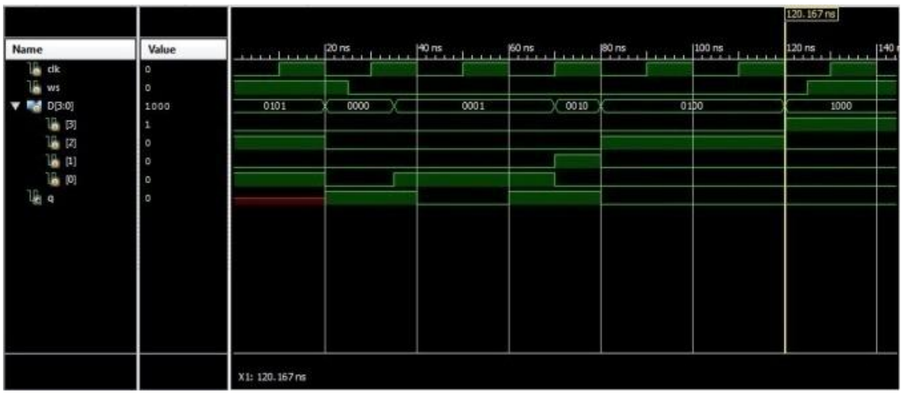

Fig.22: Parallel in serial out

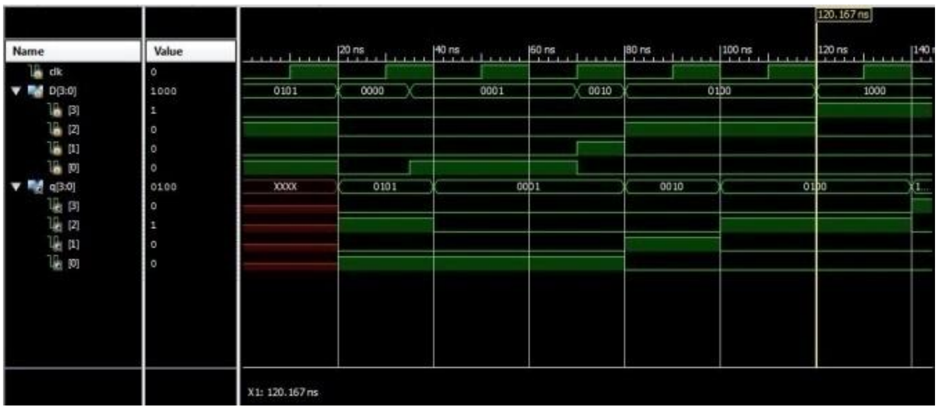

Fig.23: Parallel in parallel out

6.2 Cadence virtuoso results:

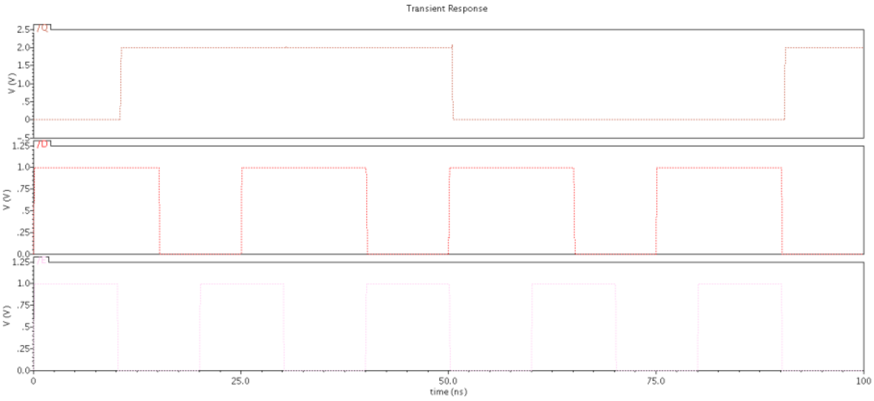

Fig.24: D flip flop

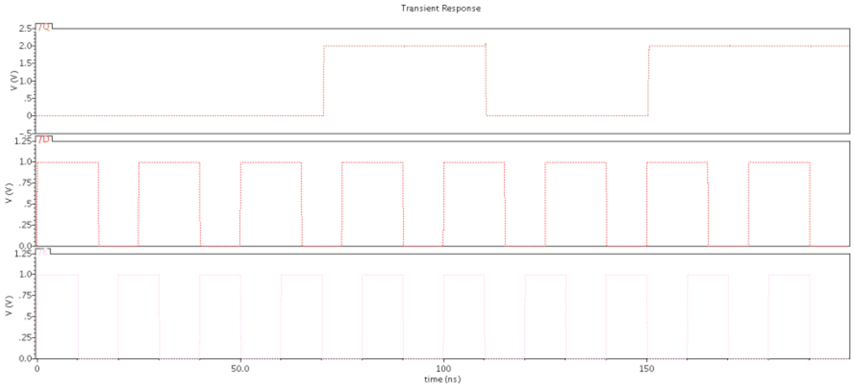

Fig.25: serial in serial out

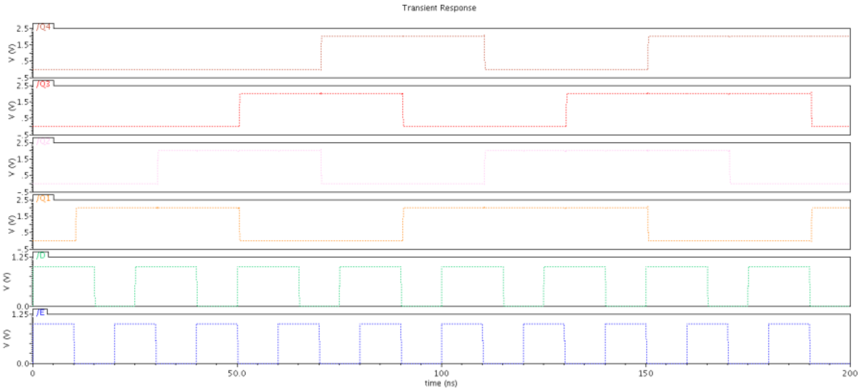

Fig.26: serial in parallel out

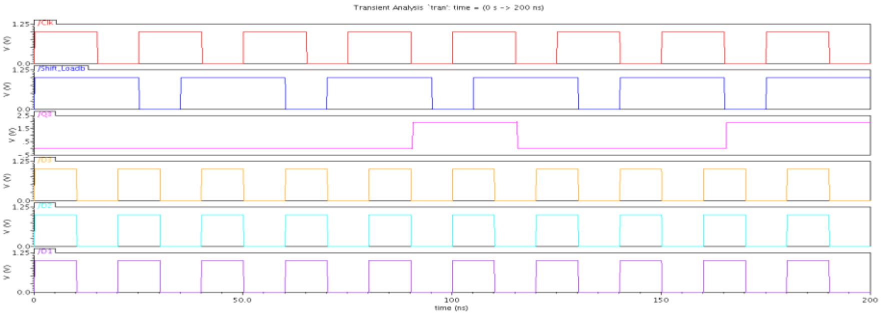

Fig.24: Parallel in serial out

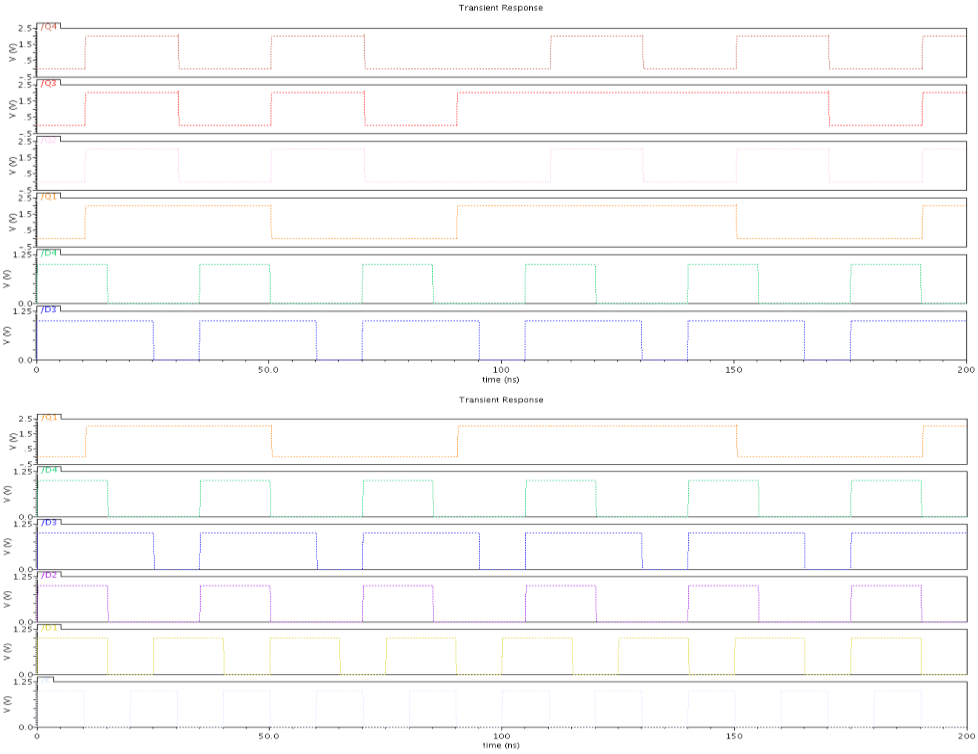

Fig.28: Parallel in parallel out

VIII. Conclusion

Reversible logic design is important for low power and quantum circuit design. In this paper we have presented designs of the optimizedreversible shift registers in terms of powerand delay. We have implemented using Xilinx ISE tool with verilog code and for cmos implementation we have used cadence virtuoso tool of 180nm technology.

References

[1]. Landauer.R, "Irreversibility and heat generation in the computing process", IBM J. Research and Development, 5(3):pp.183 -191, 1961.

[2]. Bennett, C.H., "Logical reversibility of Computation",IBM J.Research and Development, 17: pp. 525-532, 1973.

[3]. Santhi Swaroop V.G, "Implementation of Optimized Reversible Sequential and CombinationalCircuitsforVLSI Applications",2014.

[4]. Nagarjun S, Nagendra R, Kiran N, Kiran Kumar K N," Design and Comparison of Reversible and Irreversible Sequential Logic Circuits".

[5]. Ranjith.S, Umarani.P , "Performance Analysis of MOSFET and CNTFET Using Reversible Logic Shift Registers", 2014.

[6]. Ranjith.S, Ravi.T" Fault Tolerant Reversible Logic D-Flip Flop Based Shift Registers In 32nm CMOS Technology" International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 2, February- 2013 ISSN: 2278-0181.

[7]. Rashmi S, Praveen, Design of Optimized Reversible BCD Adder/Subtractor, IACSIT International Journal of Engineering and Technology, Vol.3, June 2011.

[8]. Prashant.R, Yelekar, Prof. Sujata S. Chiwande, "Introduction to Reversible Logic Gates & its Application".

[9]. Nosho.Y, Ohno.Y, Kishimoto.S and Mizutani T,Nanotechnology 18, 415202 (2007).

[10]. Parhami , "Fault tolerant reversible circuits", in Proceedings of40th Asimolar Conf. Signals, Systems, and Computers, PacificGrove, CA, pp. 1726-1729, October 2006.

[11]. A.V.Ananthalakshmi, "Design of 4-Bit Reversible Shift Registers", Volume 12, E-ISSN: 2224-266X, December 2013.

[12]. P. Mohan Krishna, V. G. Santhi Swaroop, S. Harika,"Implementation of Optimized Reversible Sequential andCombinational Circuits for VLSI Applications" ISSN : 2248-9622, Vol. 4, Issue 4(Version 1), April 2014, pp.382-388,www.ijera.com