International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 3 Issue 11 ǁ November. 2015 ǁ PP.59-67

Direct Design of Reversible Combinational and Sequential Circuits Using PSDRM Expressions

G.Janani[1], S.ChandraSekhar[2]
[1]PG scholar, Dhanekula Institute of Engineering and Technology, Vijayawada, India.
[2]Assistant Professor, Department of Electronics and Communication Engineering, Dhanekula Institute of Engineering and Technology, Vijayawada, India.

Abstract- Reversible logic will be a favourable logic by dissipating less heat than the thermo dynamic limit for the emerging computing technologies. Also it has become very promising for low power designs. Reversible designs of Combinational and Sequential circuits are built by replacing the latches, flip-flops and associated combinational gates of the traditional irreversible designs by their reversible counter parts. But this replacement technique is not very promising because it leads to high quantum cost and garbage outputs. So, in this paper we presented both the direct design and replacement designs of 5-bit up down counter and universal shift register which are practically important using reversible logic and PSDRM expressions. Replacement design is done by replacing the RTL design using reversible designs. Direct design is done by representing the state transitions and the output functions of the circuits using PSDRM expressions which are obtained from truth table or state transition table. Thus my direct design of a 5-bit updown counter and universal shift register save 42.66%, 9.79% quantum cost and 93.75%, 40% garbage outputs respectively than the replacement design.

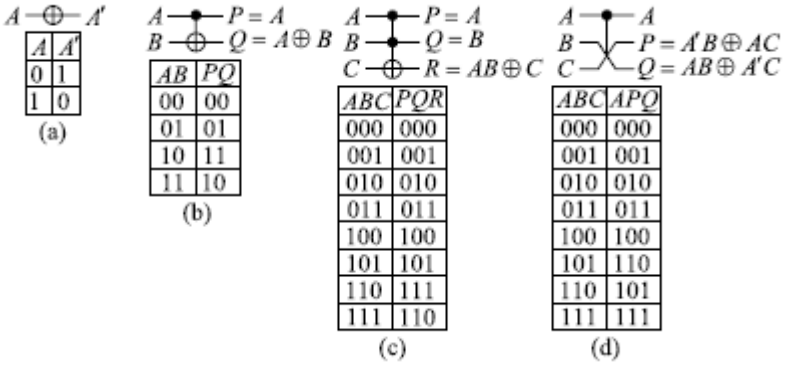
I. INTRODUCTION

Reversible logic has already found wide application in many emerging computing technologies. Therefore, developing efficient methods for reversible logic synthesis and also designing practically important reversible circuits have become very important. Most of the reversible logic synthesis attempts are concentrated on reversible combinational logic synthesis. As feedback is considered as a restriction in reversible logic, some researchers argue that reversible sequential logic is not possible. However, in 1980, Toffoli argued that if the feedback is provided through a delay element, then the feedback information will be available as the input to the reversible combinational circuit in the next clock cycle and sequential logic is possible. However, very recently, only limited attempts have been made in the field of reversible sequential logic synthesis. Reversible designs of building blocks of sequential circuits such as latches and flip-flops on the top of reversible gates and suggest that sequential circuits be constructed by replacing the latches, flip-flops, and other combinational gates of traditional irreversible designs by their reversible counter parts.

The first attempt of direct design of synchronous sequential circuit using reversible gates , where there design level-triggered up counters using positive polarity Reed–Muller (PPRM) expression for representing the state transition of the counter. The designed up counter is more efficient than the replacement design in terms of both quantum cost and garbage outputs. In that they express next state of the counter as a function of clock and present state. Then, we express the next state using PPRM expression and realize the PPRM expression using reversible gates. Similar PPRM-based reversible circuit synthesis is done. Fixed-polarity Reed– Muller (FPRM) expression requires less or at most same number of product terms than PPRM expression for a given function. Thus, FPRM-based reversible circuit synthesis is more efficient than PPRM-based reversible circuit synthesis. FPRM-based reversible circuit synthesis is done. Pseudo Reed–Muller (PSDRM) expression is a more generalized class of Reed–Muller expression and requires less or at most equal number of product terms than FPRM. Thus, PSDRM-based reversible circuit synthesis is more efficient than PPRM- and FPRM based reversible circuit synthesis.

II. REVERSIBLE LOGIC

A reversible gate (or a circuit) maps every input combination to a unique output combination. This unique mapping implies that a reversible circuit has the same number of inputs and outputs. A reversible circuit with n inputs/outputs is called an $n\times n$ reversible circuit. A reversible circuit is constructed as a network of reversible gates. Fig. 1 shows the commonly used reversible gates such as 1 × 1 NOT gate, 2 × 2 Feynman gate, 3 × 3 Toffoli gate, and 3 × 3 Fredkin gate. Toffoli gate may have more than three inputs/outputs and they are called multiple-controlled Toffoli gates.



A	A'
0	1
1	0

AB	PQ
00	00
01	01
10	11
11	10

ABC	PQR
000	000
001	001
010	010
011	011
100	100
101	101
110	111
111	110

ABC	APQ
000	000
001	001
010	010
011	011
100	100
101	110
110	101
111	111

Fig.1. Commonly used reversible gates. (a) NOT gate. (b) Feynman gate. (c) Toffoli gate. (d) Fredkin gate.

The complexity of reversible circuit design is compared in terms of **quantum cost** (the number of primitive quantum gates required to realize the circuit) and the number of **garbage outputs** (the final outputs that are not used as the primary outputs). The 1 × 1 and 2 × 2 gates are technology realizable primitive gates and their quantum costs are assumed to be one. Thus, the quantum cost of NOT gate and Feynman gate is one each. Toffoli and Fredkin gates are macro level gates and need to be realized on the top of 2 × 2 gates. The 3 × 3 Toffoli gate and the Fredkin gate can be realized using five 2 × 2 primitive gates and thus their quantum cost is five each. Realization of multiple-controlled Toffoli gates from primitive quantum gates are presented where quantum costs for up to 16 × 16 Toffoli gates are reported. The quantum costs for 4 × 4, 5 × 5, and 6 × 6 Toffoli gates are 14, 20, and 32, respectively. Classical AND, OR gates can be realized using Toffoli gates. Reversible realization of two- and three-input AND gates are shown in Fig. 2(a) and (b), respectively.

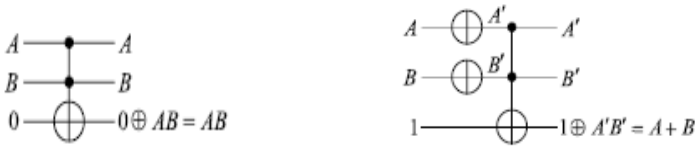


Fig.2. Reversible realizations of (a) 2-input AND gate. (b) 2-input OR gate.

Reversible realization of two-input AND gate requires five quantum cost and two garbage outputs and that of three-input AND gate requires 14 quantum cost and three garbage outputs. Reversible realization of two-input OR gate is shown in Fig. 2(c), which requires seven quantum cost and two garbage outputs. Reversible realizations of level-triggered and falling-edge triggered D flip-flops are shown in Fig. 3(a) and (b), respectively.

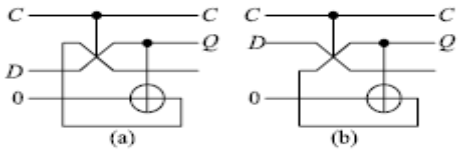


Fig.3. Reversible realization of (a) +ve level triggered. (b) -ve edge triggered D flip-flops

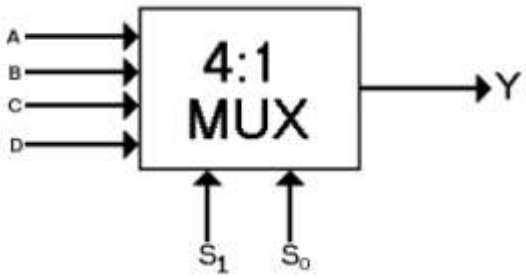
In Fig. 3(a), the state output is copied using a Feynman gate and fed back to the second input of the Fredkin gate. When the clock *C* is zero, then the feedback is connected to the state output maintaining the state output unchanged. When *C* becomes one, then the *D* input is connected to the state output performing the level-triggered load operation. This realization requires six quantum costs and two garbage output. In Fig. 3(b), the feedback is connected to the third input of the Fredkin gate. When *C* is one, then the feedback is connected to the state output maintaining the state output unchanged. When *C* becomes zero, then the *D* input is connected to the state output performing the falling-edge triggered load operation. This realization requires six quantum costs and two garbage output.

III. REVERSIBLE REALIZATION OF 4:1 MULTIPLEXER USING PSDRM

An *n*-variable Boolean function $f(x_1, x_2, \ldots, x_i, \ldots, x_n)$ can be expanded on the variable x_i using any of the following expansions $f(x1, x2, \ldots, xi, \ldots, xn) = f0 \oplus xi\, f2$ (positive Davio, pD)
$f(x1, x2, \ldots, xi, \ldots, xn) = f1 \oplus xi\, f2$ (negative Davio, nD) where $f0 = f(x1, \ldots, xi-1, 0, xi+1, \ldots, xn)$, $f1 = f(x1, \ldots, xi-1, 1, xi+1, \ldots, xn)$ and $f2 = f0 \oplus f1$.If we apply pD expansion on all variables of an *n*-variable

Boolean function *f (x*1, *x*2, . . . , *xn)*, then the resulting expression can be represented as $f(x1,x2,...,x_n) = f_{00..00} \oplus f_{00..01} x_n \oplus f_{00..10} x_{n-1} \oplus f_{00..11} x_{n-1} x_n \oplus .... \oplus f_{11..11} x_1x_2...x_{n1}x_n$ where the coefficients are *(∀ i ∈ {0, 1}n) fi ∈ {0,* 1}.If a subscript of a coefficient is one, only then the corresponding variable appears in the uncomplemented form in the associated product term. If a coefficient is one, only then the associated product term appears in the expression.

Consider a 4:1 MUX as shown below with 4 inputs A, B, C, D, 2 selection lines inputs S_1, S_0 and 1 output Y as shown in the fig. 4. Now considering its truth table, PSDRM expression can be computed directly from the PSDRM tree, as shown in Fig. 5.



(a)

Selection		Output
S_1	**S_0**	**Y**
0	0	A
0	1	B
1	0	C
1	1	D

(b)

Fig. 4. (a) 4:1 Multiplexer. (b) Truth table.

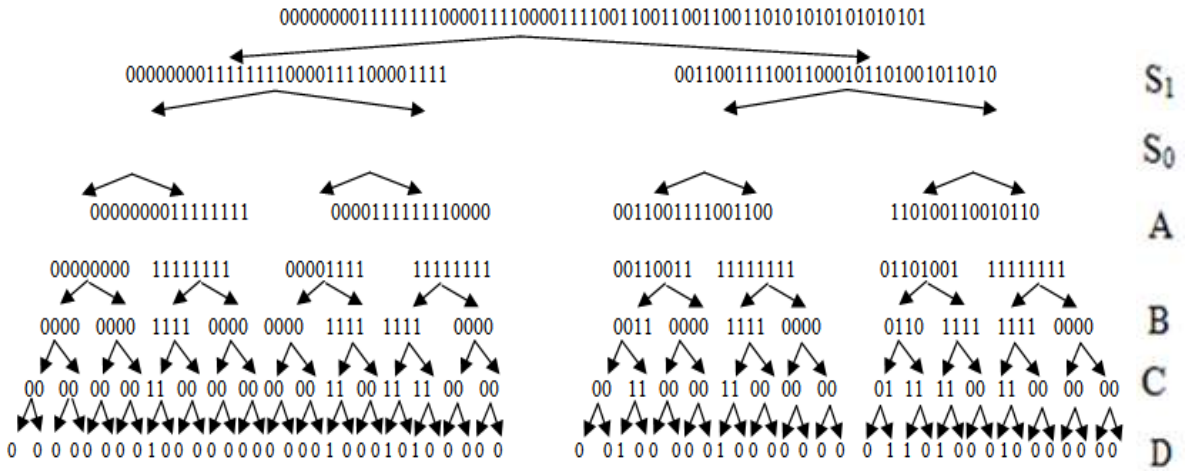


Fig.5. PSDRM tree of a 4:1 multiplexer

If we apply pD expansion on the variable S_1, then *f*0 = 00000000111111110000111100001111, *f*1 = 00110011001100110101010101010101, and *f*2 = 00110011110011000101101001011010. Now, *f*0 goes to the left child of the root and *f*2 goes to the right child of the root of the tree of Fig. 5. Similarly, the pD expansion is applied on the other internal nodes traversing the path from the root to a leaf with coefficient one. The leaves represent the final PPRM expression. The resulting expression is determined from the ones of the coefficient vector and their corresponding input combinations.

If we apply nD expansion on the variable S_1, then *f*0 = 00110011001100110101010101010101, *f*1 =00000000111111110000111100001111, and *f*2 = 00110011110011000101101001011010. Now, *f*0 goes to the left child of the root and *f*2 goes to the right child of the root of the tree of Fig. 5. Similarly, the nD expansion is applied on the other internal nodes traversing the path from the root to a leaf with coefficient one. The leaves represent the final NPRM expression. The resulting expression is determined from the ones of the coefficient vector but writing the variable in complemented form according to the corresponding input combinations.

But if we want PSDRM expression we have to choose randomly pD or nD expansion. For the PSDRM tree designed above, even though we considered getting minimum number of ones at the descendents, we get all pD expansions only hence here there is no need of complimented variables. Thus we get the 1's for 8,20,24,26,34,40,49,50,52,56 which can be written in the form of binary expansion as 001000, 010100, 011000, 011010, 100010, 101000, 110001, 110010, 110100, 111000 .above binary codes can be converted into PSDRM expression as below. Thus the resulting PSDRM expression of the tree of Fig. 4 is

$f(S_1,S_0,A,B,C,D) = A \oplus S_0\, B \oplus S_0\; A \oplus S_0\, A\, C \oplus S_1\, C \oplus S_1\, A \oplus S_0\, S_1\, D \oplus S_1\, S_0\, C \oplus S_0\, S_1\, B \oplus S_1\, S_0\, A$

The above PSDRM expression can be realized using reversible gates, as shown in Fig. 6, which is self-explanatory. The circuit of Fig. 5 requires one Feynman gate, four 3 × 3 Toffoli and five 4 × 4 Toffoli gates. Therefore, its quantum cost is 1 × 1 +4 × 5 + 5 × 14 = 91. It has one primary output and six unused outputs. Therefore, it has six garbage outputs.

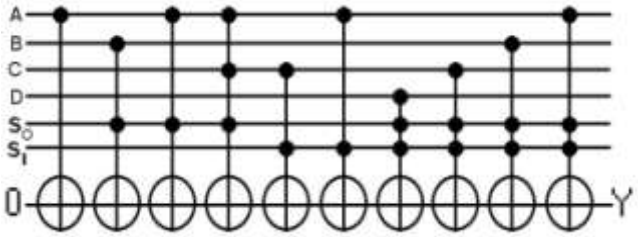


Fig.6. Reversible realization of above PSDRM expression (4:1 MULTIPLEXER)

IV. DESIGN OF REVERSIBLE 5-BIT UP DOWN COUNTER

In this section, we will first consider the Direct Design obtained from the reference [22] and then make the reversible Replacement Design and then calculate the quantum cost and garbage output for a Reversible 5-Bit Updown Counter circuit.

Firstly the Direct Design involves design of next state logic expressions. For designing the next state logic of a 5-bit up down counter, consider the classical design of a 5-bit updown counter as shown in the fig. 7. Now we can construct transition table considering the clock (designated *C)*, present states (designated Q_0 , Q_1 , Q_3 , Q_3 , Q_4*)*,mode selection input (designated as M) as inputs and the next states (designated Q_0+ , Q_1+ , Q_2 +, Q_3 + ,Q_4+*)* as the outputs. So, When M=0 it acts as an up counter and when M=1 it acts as a down counter as shown in the state transition table below.

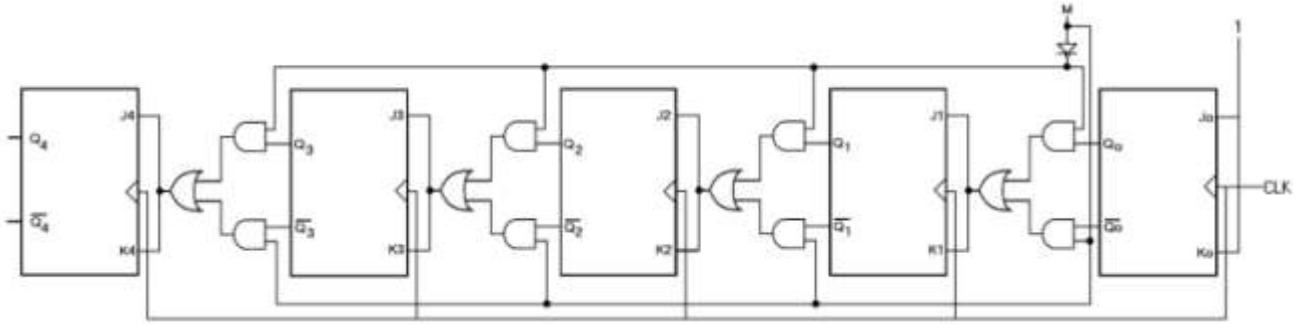


Fig.7. Classical design of a 5-bit up down counter

Table. 1. State Transition table of 5 Bit Up Down Counter

Present state	**Mode (M)**	**Next state**	
		C=0	**C=1**
Q_4 Q_3 Q_2 Q_1 Q0		***Q_4+ Q_3+Q_2+ Q_1+ Q0+***	***Q_4+ Q_3+ Q_2+ Q_1+Q0+***
00000	0	00000	00001
00001	0	00001	00010
00010	0	00010	00011
.....	....	.....	.....
.....	...	.....	.....
11101	0	11101	11110
11110	0	11110	11111
11111	0	11111	00000
11111	1	11111	11110
11110	1	11110	11101
11101	1	11101	11110
.....	.....	.....	.....
.....	.....	.....	.....
00001	1	00001	00000
00000	1	00000	11111

PSDRM expressions for the next states Q_0+ , Q_1+ , Q_2+, Q_3+ ,Q_4+ of a 5 bit updown counter of the Table I is obtained from the expressions (20)-(23) from the Reference [22] as below.

$$Q3+ = Q3 \oplus C\,(M \oplus Q2)\,(M \oplus Q1)\,(M \oplus Q0)$$
$$Q2+ = Q2 \oplus C\,(M \oplus Q1)\,(M \oplus Q0)$$
$$Q1+ = Q1 \oplus C\,(M \oplus Q0)$$
$$Q0+ = Q0 \oplus C$$

These expressions can be realized as shown in the Fig. 8.This realization needs one 6 × 6 Toffoli gate, one 5 × 5 Toffoli gate, one 4 × 4 Toffoli gates, one 3 × 3 Toffoli gates and fifteen Feynman gates. Therefore, its quantum cost is 1 × 32 + 1 × 20 + 1 × 14 + 1 × 5 + 15 × 1 = 86. The circuit has two garbage outputs (M, C) as shown below.

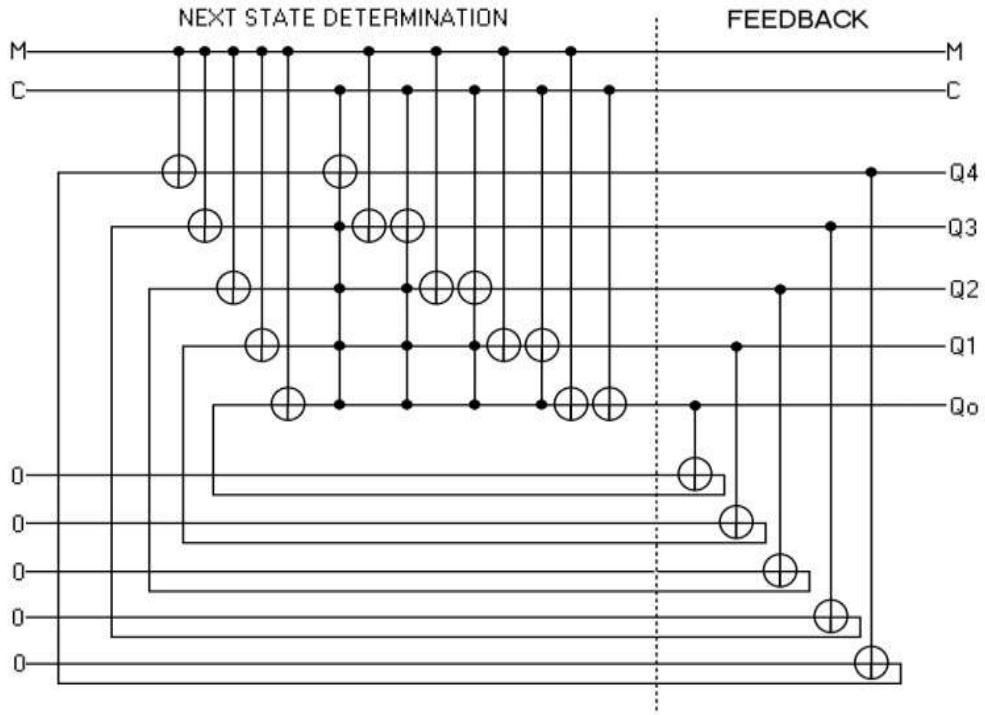


Fig.8. Reversible realization of Direct Design of a 5 bit up down counter

Secondly the Replacement Design involves the process of converting the Fig. 7. into reversible design which requires the reversible designs of JK Flip Flop, And gate, Or gate, and Inverter so that it can be replaced as shown in Fig. 10. As discussed in the chapter II we have all the And gate, Or gate, and Inverter designs .Also JK Flip Flop can be designed using D Flip Flop using the equation D = J Qbar + K' Q as shown in the Fig. 9.

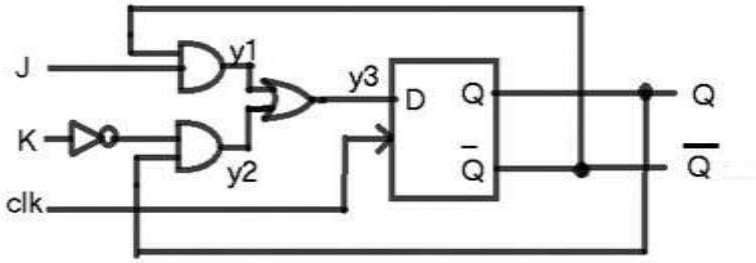


Fig. 9. Classical design of JK Flip Flop design using D Flip Flop

Thus the final reversible replacement design of Fig.7. is as shown in the Fig.10.

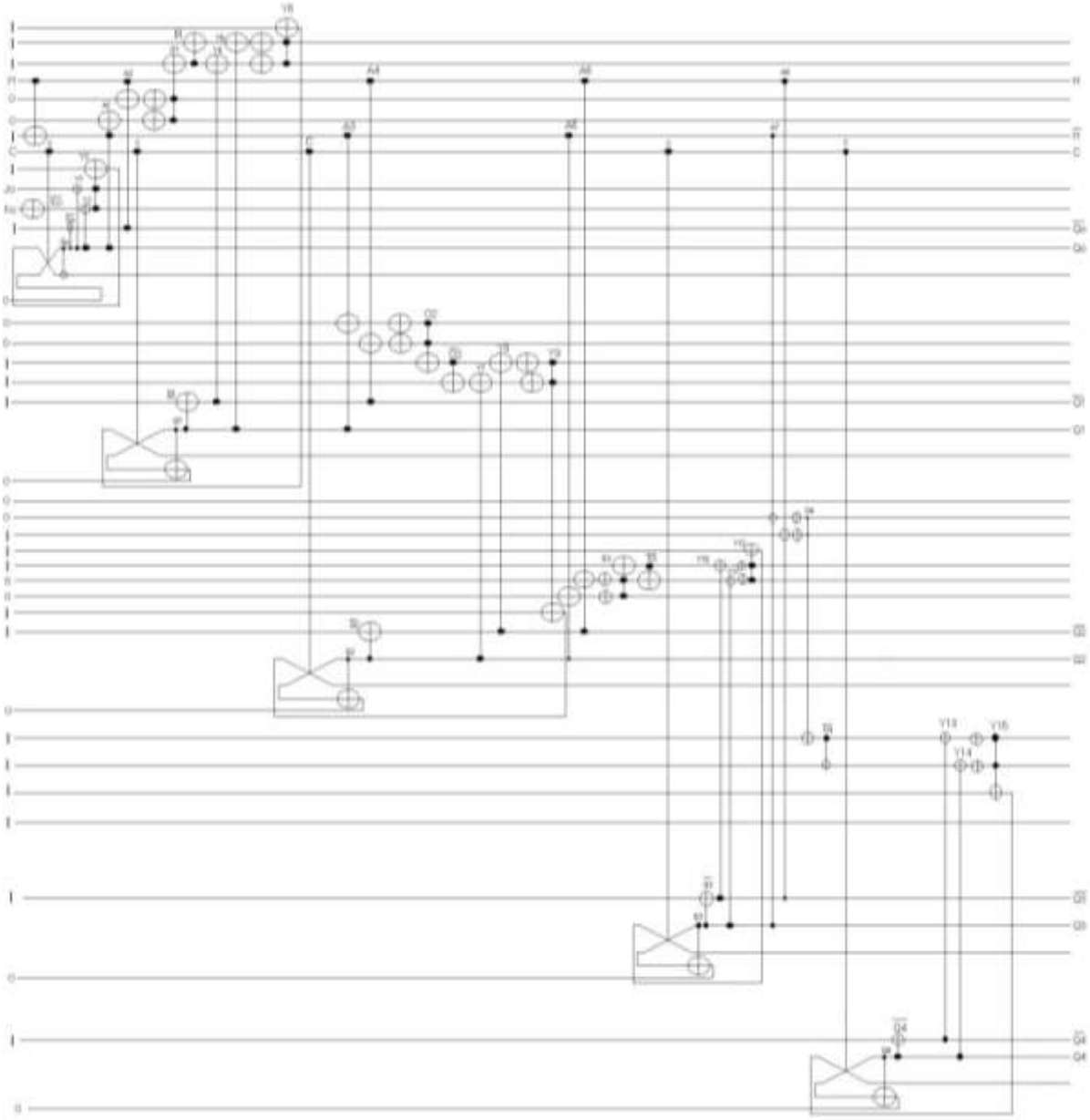


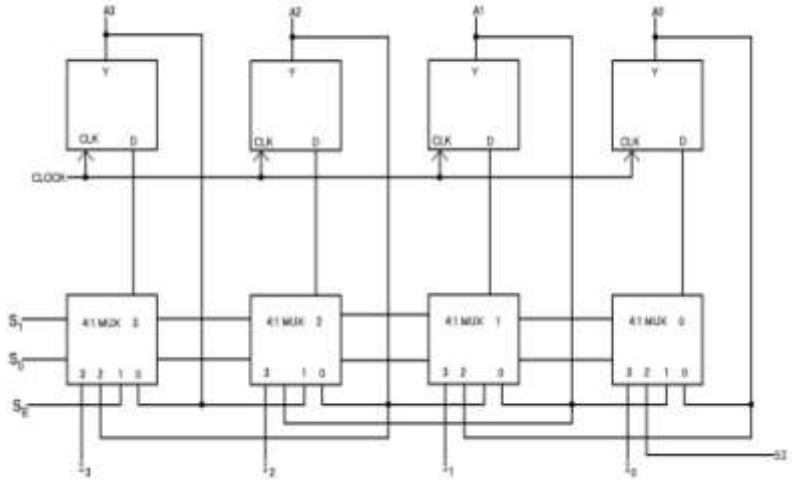
Fig.10. Reversible realization of Replacement Design of a 5 bit up down counter

This reversible Replacement Design realization needs seventeen NOT gates, five D Flip Flops, seventeen 3 × 3 Toffoli gates and twenty Feynman gates. Therefore, its quantum cost is 17 × 1 + 5 × 6 + 17 × 5 + 20 × 1 = 150. The circuit has thirty two garbage outputs.

V. DESIGN OF REVERSIBLE UNIVERSAL SHIFT REGISTER

In this section, we will Design the practically important Universal Shift Register using the Direct Design and then make the reversible Replacement Design. Finally we will calculate the quantum cost and garbage output and comparing them.

Firstly the Direct Design involves design of next state logic expressions. For designing the next state logic of a Universal Shift Register, consider the classical design of a Universal Shift Register as shown in the fig.11. Now we can construct transition table considering the clock (designated *C),*four present states (designated Q_0 , Q_1 , Q_2 , Q_3*)*, two mode selection inputs (designated as S_1 , S_0) ,shift right and shift left inputs(designated as S_r, S_l), four parallel load inputs (designated as A,B,C,D) and the next states (designated Q_0+ , Q_1+ , Q_2 +, Q_3 +) . Its classical design and state transition table is as shown in the fig.11.



(a)

Selection		USR operation
S_1	**S_0**	
0	0	No Change
0	1	Shift Right
1	0	Shift Left
1	1	Parallel Load

(b)

Fig.11. (a) Classical Design (b) State Transition Table of a Universal Shift Register

PSDRM expressions for the next states Q_0+ , Q_1+ , Q_2+, Q_3+ of a Universal Shift Register of the Table I is obtained from the PSDRM tree as below.

$Q_3+ = Q_3 \oplus C\,Q_3 \oplus C\,S'_1\,S'_0\,Q_3 \oplus C\,S'_1\,S_0\,S_r \oplus C\,S_1\,S'_0\,Q_2 \oplus C\,S_1\,S_0\,A$

$Q_2+ = Q_2 \oplus C\,Q_2 \oplus C\,S'_1\,S'_0\,Q_2 \oplus C\,S'_1\,S_0\,Q_3 \oplus C\,S_1\,S'_0\,Q_1 \oplus C\,S_1\,S_0\,B$

$Q_1+ = Q_1 \oplus C\,Q_1 \oplus C\,S'_1\,S'_0\,Q_1 \oplus C\,S'_1\,S_0\,Q_2 \oplus C\,S_1\,S'_0\,Q_0 \oplus C\,S_1\,S_0\,C$

$Q_0+ = Q_0 \oplus C\,Q_0 \oplus C\,S'_1\,S'_0\,Q_0 \oplus C\,S'_1\,S_0\,Q_1 \oplus C\,S_1\,S'_0\,S_l \oplus C\,S_1\,S_0\,D$

These expressions can be realized as shown in the Fig.12. This realization needs sixteen 5 × 5 Toffoli gate, four 3 × 3 Toffoli gates and ten Feynman gates. Therefore, its quantum cost is 16 × 20 + 4 × 5 + 10 × 1 = 350. The circuit has fifteen garbage outputs as shown below.

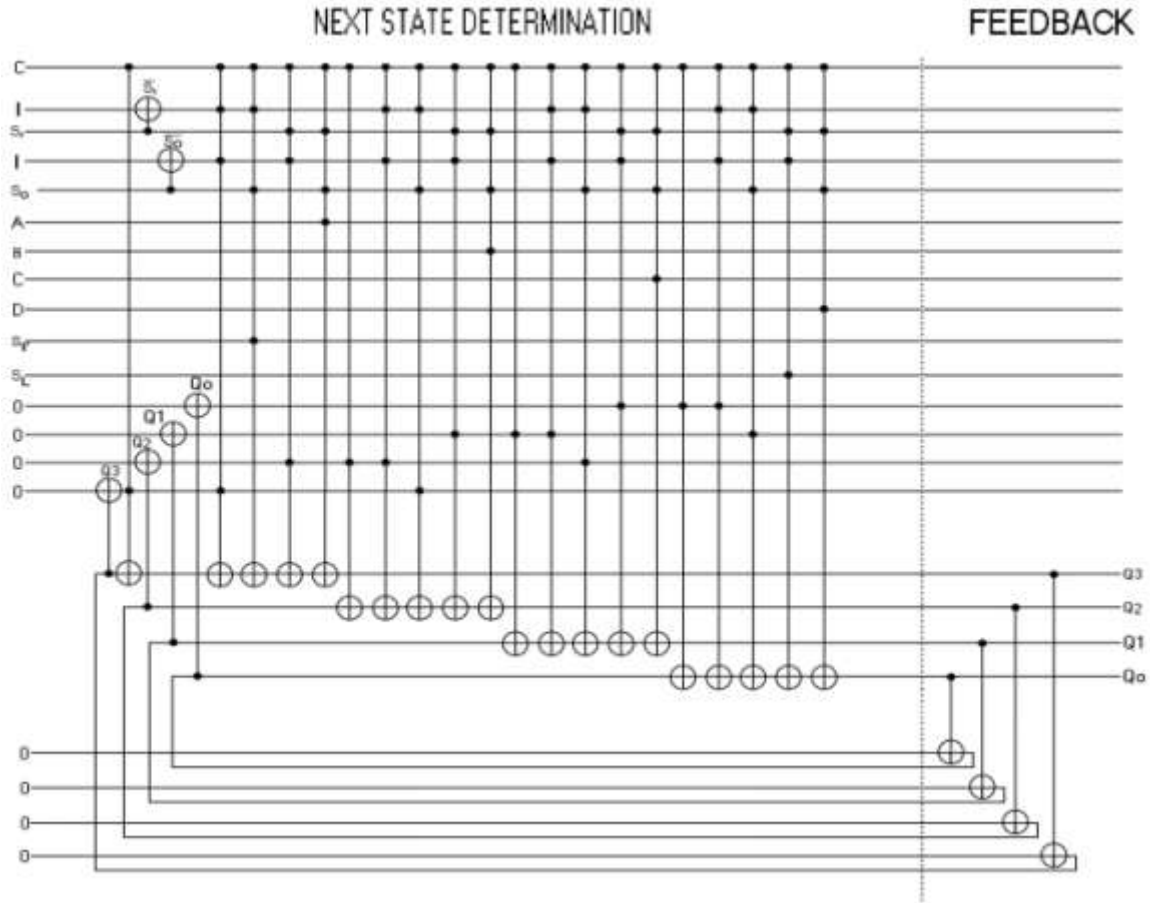


Fig.12. Reversible realization of Direct Design of a Universal Shift Register

Secondly the Replacement Design involves the process of converting the Fig.11. into reversible design which requires the reversible designs of 4:1 Mux's, and D Flip Flop's so that it can be replaced as shown in Fig. 13. As discussed earlier, in the chapter II we have the D Flip Flop and in chapter III we presented the reversible multiplexer design. Thus the final replacement design of Fig.11. is as shown in the Fig.13.

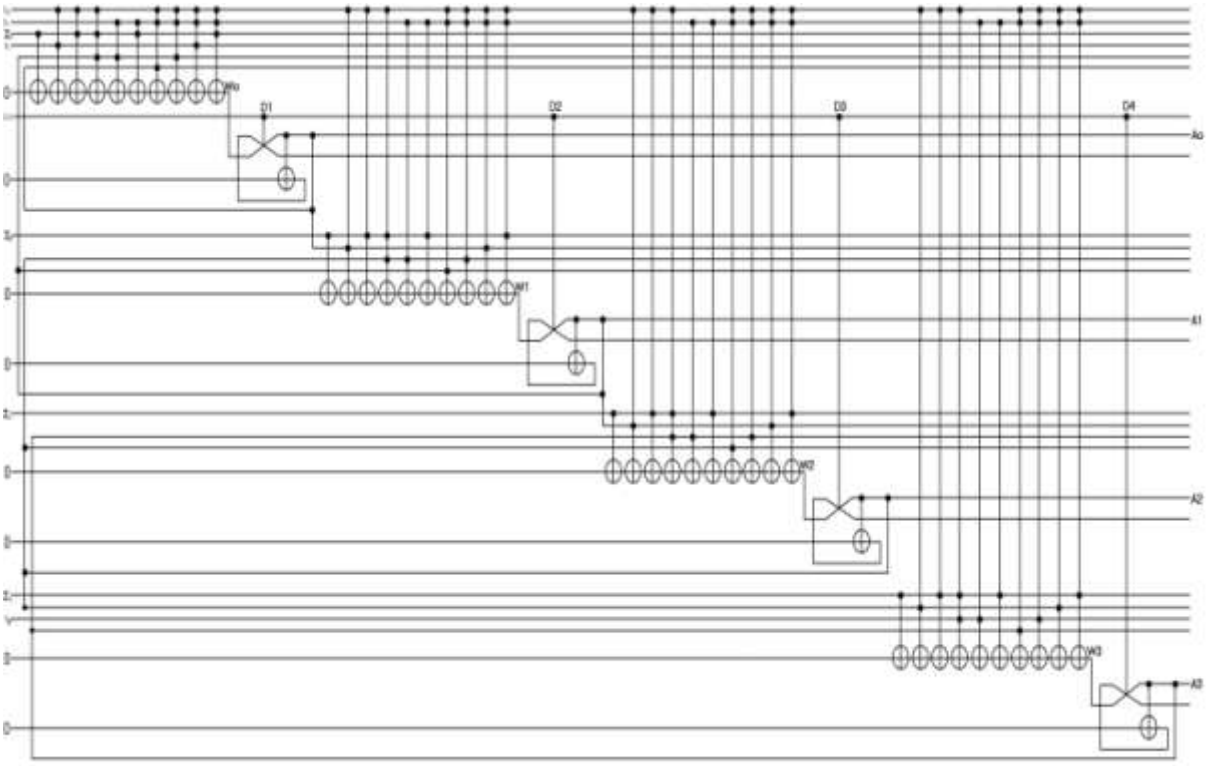


Fig.13. Reversible realization of Replacement Design of a Universal Shift Register

This reversible Replacement Design realization needs four D Flip Flops, sixteen 3 × 3 Toffoli gates, twenty 4 × 4 Toffoli gates and four Feynman gates. Therefore, its quantum cost is 4 × 6 + 16 × 5 + 20 × 14 + 4 × 1 = 388. The circuit has twenty five garbage outputs.

VI. COMPARISIONS

Comparison of the direct design with the replacement design of a 5-bit updown counter is given in the Table II. From table II we see that our direct design requires 42.66% less quantum cost and 93.75% less garbage outputs than the replacement design obtained from the [22].

Table II. Comparison of the direct design with the replacement design of a 5-bit updown counter.

Complexity comparison	Direct Design	Replacement Design	% improvement
Quantum Cost	86	150	42.66
Garbage output	2	32	93.75

Similarly, Comparison of the direct design with the replacement design of a Universal Shift Register is given in the Table III. From table II we see that our direct design requires 9.79% less quantum cost and 40% less garbage outputs than the replacement design.

Table III. Comparison of the direct design with the replacement design of a Universal Shift Register.

Complexity comparison	Direct Design	Replacement Design	% improvement
Quantum Cost	350	388	9.79
Garbage output	15	25	40

VII. CONCLUSION

Reversible logic has shown a good promise for low-power design using emerging computing technologies. A good number of design methods for reversible combinational circuits have been proposed [1]–[2]. However, only a very limited works have been reported on reversible sequential circuit design [14]–[20]. In this paper, we present a novel approach of direct design of 4:1 Multiplexer and important 5 bit updown counter along with a Universal shift register. Design examples show that our direct designs save quantum cost and garbage outputs than the replacement design approach suggested earlier as given in tabular forms in chapter VII. Thus, our proposed direct design method outperforms the previously reported replacement design approach.

REFERENCES

[1] D. Große, R. Wille, G. Dueck, and R. Drechsler, "Exact multiple control Toffoli network synthesis with SAT techniques," *IEEE Trans.Comput.-Aided Design Integr. Circuits Syst.*, vol. 28, no. 5, pp. 703–715, May 2009.

[2] S.Mahammad and K. Veezhinathan, "Constructing online testable circuits using reversible logic," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 1, pp. 101–109, Jan. 2010.

[3] T. Toffoli, "Reversible computing," MIT Lab. Comput. Sci., Cambridge, MA, USA, Tech. Rep. MIT/LCS/TM-151, 1980.

[4] J. E. Rice, "A new look at reversible memory elements," in *Proc. IEEE ISCAS*, May 2006, pp. 243–246.

[5] S. K. S. Hari, S. Shroff, S. N. Mohammad, and V. Kamakoti, "Efficient building blocks for reversible sequential circuit design," in *Proc.49th IEEE MWSCAS*, Aug. 2006, pp.437–441.

[6] H. Thapliyal and A. P. Vinod, "Design of reversible sequential elements with feasibility of transistor implementation," in *Proc. ISCAS*, 2007, pp. 625–628.

[7] M.-L.Chuang and C.-Y.Wang, "Synthesis of reversible sequential elements," *ACM J. Emerg. Technol.*, vol. 3, no. 4, pp. 1–19, 2008.

[8] H. Thapliyal and N. Ranganathan, "Design of reversible sequential circuits optimizing quantum cost, delay and garbage outputs," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 6, no. 4, pp. 14:1–14:35, 2008.

[9] M. Haghparast and M. S. Gharajeh, "Design of a nanometric reversible 4-bit binary counter with parallel load," *Austral. J. Basic Appl. Sci.*, vol. 5, no. 7, pp. 63–71, 2011.

[10] M. H. A. Khan and M. Perkowski, "Synthesis of reversible synchronous counters," in *Proc. 41st IEEE ISMVL*, May 2011, pp. 242–247.

[11] J. Hu, G. Ma, and G. Feng, "Efficient algorithm for positive-polarity Reed-Muller expansions of reversible circuits," in *Proc. ICM*, Dec. 2006, pp. 63–66.

[12] K. Takahashi and T. Hirayama, "Reversible logic synthesis from positive Davio trees of logic functions," in *Proc. IEEE Region Conf. TENCON*, Jan. 2009, pp. 1–4.

[13] Y. Pang, S. Wang, Z. He, J. Lin, S. Sultana, and K. Radecka, "Positive Davio-based synthesis algorithm for reversible logic," in *Proc. 29th ICCD*, Oct. 2011, pp. 212–218.

[14] Z. Jing and J. C. Muzio, "Improved implementation of a Reed-Muller spectra based reversible synthesis algorithm," in *Proc. IEEE Pac Rim Conf. Commun., Comput. Signal Process*, Aug. 2007, pp. 202–205.

[15] M. Soeken, R. Wille, and R. Drechsler, "Hierarchical synthesis of reversible circuits using positive and negative Davio decomposition," in *Proc. 5th IDT*, Dec. 2010, pp. 143–148.

[16] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, *et al.*, "Elementary gates for quantum computation," *Phy. Rev. A*, vol. 52, no. 5, pp. 3457–3467, 1995.

[17] J. A. Smolin and D. P. DiVincenzo, "Five two-bit quantum gates are sufficient to implement the quantum Fredkin gate," *Phys. Rev. A*, vol. 53, no. 4, pp. 2855–2856, 1996.

[18] D. M. Miller, R. Wille, and Z. Sasanian, "Elementary quantum gate realizations for multiple-controlled Toffoli gates," in *Proc. 41st IEEE ISMVL*, May 2011, pp. 288–293.

[19] T. Sasao, "AND-EXOR expressions and their optimization," in *Logic Synthesis and Optimization*, T. Sasao, and Ed. Norwell, MA, USA: Kluwer, 1993, pp. 287–312.

[20] M. M. H. A. Khan and M. S. Alam, "Algorithms for conversion of minterms to positive polarity Reed-Muller coefficients and vice versa," *Inf. Process. Lett*, vol. 62, no. 5, pp. 223–230, 1997.

[21] M. M. H. A. Khan and M. S. Alam, "Mapping of fixed polarity Reed- Muller coefficients from minterms and the minimization of fixed polarity Reed-Muller expressions," *Int. J. Electron.*, vol. 83, no. 2, pp. 235–247,1997.

[22] Mozammel H A Khan, "Design of Reversible Synchronous Sequential Circuits using Pseudo Reed-Muller Expressions" IEEE transaction on VLSI systems, VOL.22, NO. 11, November 2014.