On Design of a Novel Nano Metric Parity Preserving Reversible Random Access Memory

Ghahreman Pourvali
Sama technical and vocational training college,
Islamic Azad University, Parsabad Branch, Parsabad, Iran

ABSTRACT
Attempts are kept going to decrease energy consumption and reversible circuits are seen to be of high importance to do so. Reversible logic is used in some area such as Nanotechnology, quantum computing, optical computing and low-power CMOS design. In the present study a novel parity preserving reversible random access memory is designed. General designs for components of PPRRAM are introduced. In addition a new reversible gate, PH3, is introduced which is Parity preserve and capable of being utilized in various reversible circuits. We have used it to design parity preserving reversible master slave D flip-flop and parity preserving reversible memory cell. The proposed master slave D flip-flop and write enable master slave D flip- flop is compared with existing works and its efficiency is shown in terms of gate counts and garbage outputs. All the scales are in the Nano metric area.

General Terms

Nanotechnology, VLSI Design, Fault Tolerance System, Quantum Computing, Low Power Design

Keywords

Reversible Logic, Parity Preserving, Random Access Memory, Flip-flop, Garbage Output.

1. INTRODUCTION
In the early 1960s, landauer showed that in irreversible computation for every bit of information that is erased, KTln2 joules of energy dissipates as heat where K is Boltzmann's constant and T is the absolute temperature at which the computation is performed. Bennett showed that in reversible computation do not lose information which can avoid ktln2 energy dissipation [2]. A circuit is reversible if there is a one-to-one correspondence between input and output, put it simply, in reversible gates input vector can be achieved by output vector. One of the key parts of the electronic devices is memory. Efficiency designing of memory would develop the efficiency of an electronic machine. On the other hand, reversible circuits, having many advantages, especially in energy consumption, could help design effective memory. Among the used memories, random access memory is an important part in electronic devices, computer in particular. Few studies have been done Realm of reversible technology for designing of reversible random Access memory (RRAM). The present study wants to establish Parity preserving reversible random access memory (PPRRAM) and therefore, its components, i.e., multiplexer, decoder, and memory cells have been designed. The general designs of memory components is introduced which can be used for designing of circuits with arbitrary input and output counts. Designs compared with existing works and their efficiency is shown. There are several metric for evaluation of reversible circuit. In this study, circuits are compared in terms of gate counts and garbage outputs. In this work, a novel design of master slave D flip-flop is presented which use optimized gate counts and garbage outputs. A new reversible conservative, also parity preserving gate, namely ph3 gate, is introduced for designing master slave D flip-flop. Ph3 is a Useful gate which can be used for designing of various flip-flops. Parity checking is one of the most popular methods for error detection in digital system. Since in reversible circuits fan-out not allowed, if we use irreversible methods for parity checking in reversible circuits, the number of used gates will be increased. Parity preserving reversible gates can be used for parity checking in reversible circuits. A reversible gate is parity preserve if the parity of input and output is the same. In this work, parity preserving random access memory is implemented, thus, all gates is used in each components of PPRRAM is parity preserving.

2. BASIC CONCEPTS
In this section we explain some basic concepts in quantum circuits and reversible gates which are as follow: Reversible gate, garbage output, quantum gate and quantum cost.

2.1 Reversible gate
A gate with equal number of input and output in which input status vectors can be revealed through output vectors. If the input vector of a reversible gate is defined by $I_V=(I_1,I_2,I_3,\ldots,I_n)$, the output vector can be represented as $Ov=(O_1,O_2,O_3, \ldots,O_n)$. A reversible gate can be represented as N×N which the number of its inputs and outputs is N.

2.2 Garbage output
A garbage output is an output that is added to change an irreversible gate to a reversible one and not used for the next calculations. It could be said that in a reversible circuit too, some outputs are produced with some other particular outputs that are not used. Then, a myriad number of garbage outputs are not appropriate. The least number of garbage output needed to change a reversible gate to a reversible one is $\lceil \mathrm{Log}_2^N \rceil$ in which N is the repetition of particular pattern in an output. For instance, to refry OR gate, since '1' repeats three times in output, therefore, two bits is required ($\lceil \mathrm{Log}_2^3 \rceil = 2$).

2.3 Quantum gate
Quantum gates are based on quantum computing and are reversible too. We can use quantum technique for realizing 1×1 and 2×2 quantum gates. Since bigger quantum gates such as 3×3 can not be realized by quantum technique directly, 1×1 and 2×2 quantum gates are used for realizing bigger quantum gates such as 3×3 and 4×4 and etc.

2.4 Quantum cost

There are several methods for computing of quantum cost of quantum circuits or reversible circuits. In simple terms, quantum cost is the number of 1×1 and 2×2 quantum gates is used in quantum circuits or reversible circuits.

3. REVERSIBLE GATES

Several reversible gates have been introduced so far, For instance, NOT gate, Feynman gate (FG), Toffoli Gate (TG), peres Gate (PG). There are several basic gates which can be used for realizing reversible gates or circuits, such as, NOT, Controlled-V, Controlled V^+ and Controlled NOT (Feynman Gate). Since the quantum cost of basic gates is taken unity, thus, they are used for computing of quantum cost of reversible circuits.

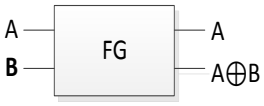
3.1 NOT Gate

The quantum implementation of NOT gate is shown in Fig.1. NOT gate is a 1×1 reversible gate and its quantum cost is unity. It should be noted that Controlled-V is a square-root-of NOT gate and V^+ is its hermitian.

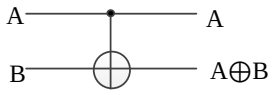
Fig. 1: NOT Gate

3.2 Controlled-NOT Gate (CNOT)

CNOT gate is a 2×2 reversible gate and Also known as Faynman gate. The block diagram and quantum implementation of Feynman gate is represented in Fig. 2(a) and 2(b). One of the main limitations of designing reversible circuits is lack of fan-out use. Therefore, FG can be used to copy a signal in reversible circuits. In way that if input vector of FG is quantified I_v= (A, 0), the output vector of circuit will be O_v= (A, A), represented in Fig. 2(c).



(a)



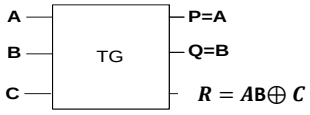
(b)

(c)

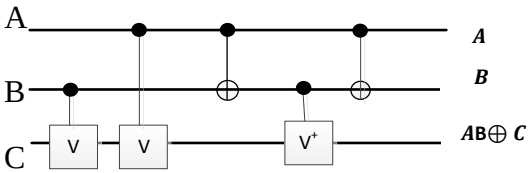
Fig. 2: (a) CNOT gate (Feynman gate) (b) Quantum implementation of CNOT gate (c) Copying an input signal by Feynman gate

3.3 Toffoli Gate (TG)

Toffoli gate is a 3×3 reversible logic gate and also known as Controlled Controlled-NOT (CCNOT) gate. The block diagram and quantum representation of Toffoli gate is shown in Fig. 3(a) and 3(b). Toffoli gate can be used to make AND operation. If input vector of Toffoli gate is quantified I_v= (A, B, 0), the output vector of TG will be O_v= (A, B, AB). Quantum cost of TG is 5.



(a)



(b)

Fig. 3: (a) Toffoli gate (b) Quantum representation of Toffoli gate

3.4 Parity preserving Reversible gates

In these kinds of reversible gates, the parity of input and output is the same. Put it simply, XOR of input and output vector are equal. Parity preserving reversible gate is a fault tolerant reversible gate. Some of the parity preserving reversible gates is as follow: New fault tolerant Gate (NFT) and Feynman Double Gate (F2G) which has been used in this work, represented in Fig. 3, 4 respectively. Parity preserving reversible gates are used for designing of PPRRAM, therefore, F2G and NFG is used to make copy a signal and perform of AND operation respectively. Quantum cost of F2G is 2 and NFG is 5.

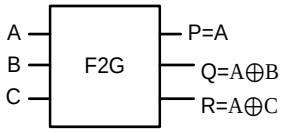


Fig. 3: Feynman Double-Gate (F2G)

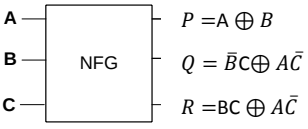
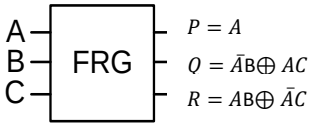


Fig. 4: New Fault Tolerant Gate (NFT)

3.5 Conservative reversible Gates

The fault tolerant reversible gates, not only in compass parity preserving gates, the equality of input and output parity, but also have an equal hemming weight. Put it other way, in conservative reversible gates the number of ones in input and output are similar. It's clear that the conservative gates are a parity preserving, while not Vic versa. FRG is one of the few conservative gates have been introduced in the literature which have been used in this work and shown in Fig. 5. Quantum cost of FRG is 5.



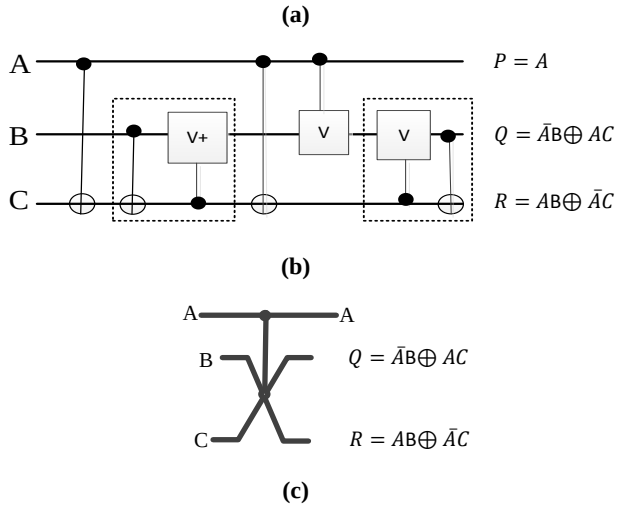


Fig. 5: (a) Fredkin Gate (b) Quantum implementation (c) Symbol of Fredkin Gate

4. DESIGN OF PARITY PRESERVING REVERSIBLE RANDOM ACCESS MEMORY

One of the key elements of electronic circuits is different array of memory. Random access memory (RAM) is of high significance, among other varieties and efficiency designing of which play an integral role in the effectiveness of electronic devices. With regards to the advantages of reversible gates in terms of energy wasting, this study wants to design a Parity preserving reversible random access memory (PPRRAM).

4.1 Parity Preserving Reversible Multiplexer

2×1 multiplexer is presented in [19] can be used to establish a general design of multiplexer. The 4×1 multiplexer is shown in Fig. 6, presented in [17].

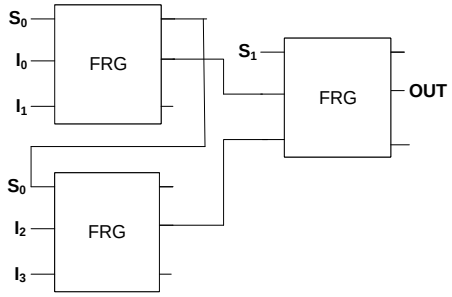


Fig. 6: parity preserving reversible 4×1 multiplexer [17].

A general design of multiplexer with what even number of input in literature has not been presented. In contrary, in this section a general design of multiplexer with favorite inputs by means of 2×1 multiplexer is proposed, similar design of Fig. 6 is shown in Fig. 7. Since only FRG is used to design a 2^n×1 MUX, therefore, the represented design is characterized with parity preserving feature too. The MUX has 2^m inputs and m selectors. Inputs are defined by I_0... I_{n-1} and selectors are defined by $S_0 \ldots S_{m-1}$.

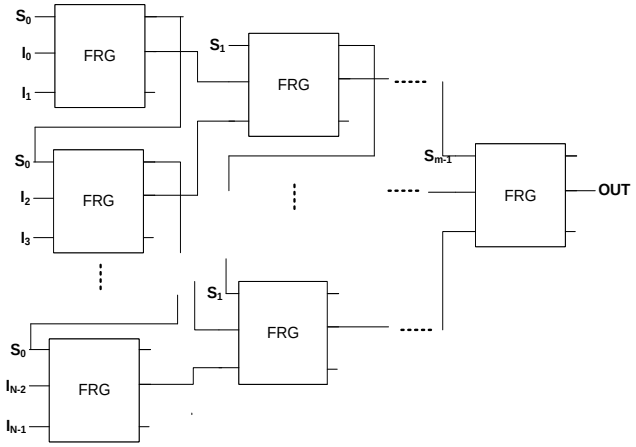


Fig. 7: Proposed general design of parity preserving reversible multiplexer

According to the above Figure, should we think of an N×1 multiplexer with 2^m input and "m" would be the number of its selectors, the design needs N-1 or 2^m-1 reversible gates and N+m-1 or 2^m+m-1 garbage outputs to create the multiplexer. For example a 4×1 multiplexer, 3 reversible gates and 5 garbage outputs in which N is 4 and m is 2, similar a 8×1 multiplexer would need 7 reversible gate and 10 garbage outputs. Quantum cost of FRG is 5, therefore, quantum cost of proposed design is $5*(2^m-1)$. The evaluation of proposed general design of parity preserving multiplexer is shown in Table 1.

Table 1: Evaluation of proposed general design of parity preserving reversible 2^m×1 multiplexer

	Gate counts	garbage outputs	Quantum cost
2^m×1 multiplexer	2^m-1	2^m+m-1	$5*(2^m-1)$

4.2 Parity preserving Reversible Decoders

To design a 1×2 decoder FG can be used, is represented in [18], and for parity preserving circuits F2G is used. The reversible 1×2 decoder design is shown in Fig. 8 and a parity preserving decoder is shown in Fig. 9.

Fig. 8: reversible 1×2 decoder

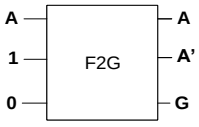


Fig. 9: proposed parity preserving reversible 1×2 decoder

For designing a 2×4 decoder, the design of 1×2 decoder is used in way that 1×2 decoder has been utilized as primary input for the two reversible FRG gates. Fig. 10 shows 2×4 decoder design. In F2G two outputs of A and $\bar{A}$ is available and these two outputs merge with B input and FRG gate and

give forth $\overline{A}\overline{B}$, $\overline{A}B$, $A\overline{B}$ and AB. On the basis of 2×4 decoder, a similar design can be used to establish reversible decoder of high level. The achieved design, based on a F2G and more FRG gates, possess parity preserving feature too.

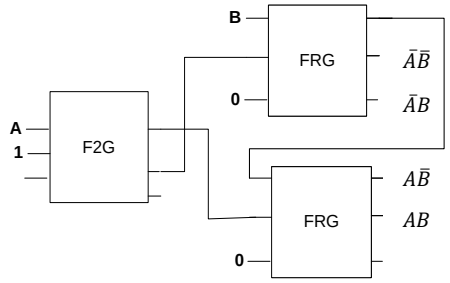


Fig. 10: proposed parity preserving reversible 2×4 decoder

Now, to designing of a general design for every individual decoder, Fig. 11 can be considered. As it can be seen in a general design of a decoder more level FRG gate is used. In this figure the general design of decoder shown sketches as $n\times2^n$. The inputs "I" ranges from I_1-I_n. With a critical inspection of the design, it is observed that to design an $n\times2^n$ decoder, 2^n-1 reversible gates and 2^n-1 garbage outputs is required. For example, to design a 1×2 decoder one reversible gate and one garbage output is required. This would be 7 reversible gates and 7 garbage outputs the case of a 3×8 decoder. Evaluation of the proposed general design of parity preserving reversible $n\times2^n$ decoder is shown in table 2.

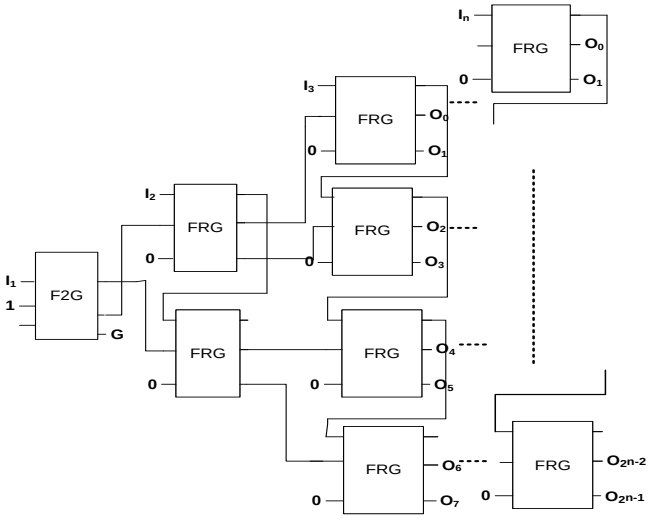


Fig. 11: Proposed general design of parity preserving reversible decoder

Table 2: Evaluation of the proposed general design of parity preserving reversible $n\times2^n$ decoder

	Gate counts	garbage outputs	Quantum cost
$n\times2^n$ decoder	2^n-1	2^n-1	$5*(2^n-1)-3$

4.3 Proposed Parity preserving memory cells

To create memory cells, which is the integral part of a memory, the new reversible PH3 gate is proposed. PH3 is a 4×4 reversible gate. The whole illustration and output of PH3 is represented in Fig. 12. PH3 is a conservative reversible gate. Therefore, it can be used in designing parity preserving reversible circuits. The true table of PH3 has been shown in Table 3. Whit a look to the true table of PH3, it can be seen that it is a conservative reversible gate, since the number of ones in inputs and outputs are equal.

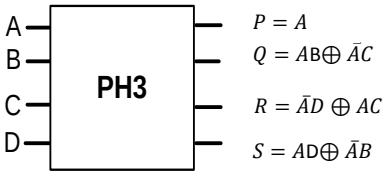


Fig. 12: Conservative reversible PH3 gate

4.3.1 Proposed Parity preserving master-Slave D flip-flop

PH3 can be used for designing various reversible circuits. The design brought up in the coming lines is the D flip-flop designing, by means of PH3 conservative reversible gate. D Flip-Fop is master-Slave kind and is Edge-triggered.

Table 3: True table of reversible PH3 gate

A	B	C	D	P	Q	R	S
0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	0
0	0	1	0	0	1	0	0
0	0	1	1	0	1	1	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	1	1
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	1
1	0	1	0	1	0	1	0
1	0	1	1	1	0	1	1
1	1	0	0	1	1	0	0
1	1	0	1	1	1	0	1
1	1	1	0	1	1	1	0
1	1	1	1	1	1	1	1

As can be seen in Fig. 13, the first latch output or Q_{m+1} is available from forth output of PH3 and second latch output or Q_{s+1} is available from the second output of PH3 which enters to the first F2G input, to expand it, and the final output is achieved through the second output of F2G gate. Regarding the output function of latches, it should be said that the output of master latch or the first latch is as clkD+$\overline{\text{clk}}$Qs in which clk stands for clock and the output of second latch is as $\overline{\text{clk}}$Qs+clkQm. So once clock is high, the master latch is active and slave is inactive, once clock is low, the slave latch

is active and the master latch is inactive and the output of master latch enters slave latch. According to the bellow mental issues, D Flip-flop works with descending of clock and as for as the clock is zero, it will never accept new input and will preserve the previous amount. The researches for passed the designed flip-flop with existing designs to this end, Table 4 is figured, showing the results of comparing the present design with Min-Lun Chuang (2008) [8], NoorMuhammad (2009) [12], Himanshu Thapliyal (2010) [17] and Bhagyalakshmi H R (2012) [19] designs. It shows that the number of constant inputs for [8], [12], [17] and [19] is 3 while 1 for the proposed design. The number of used gates in [8], [12], [17] and [19] is 4, 5, 5, 4 and 3 respectively while 2 for the proposed design. The number of garbage outputs in [12] and [19] is 3, Thapliyal and Bhagyalakshmi is 2 while 1 for the proposed design. Therefore, the proposed design is better than the compared ones. Additionally, the proposed design is more advantages in terms of the parity preserving feature.

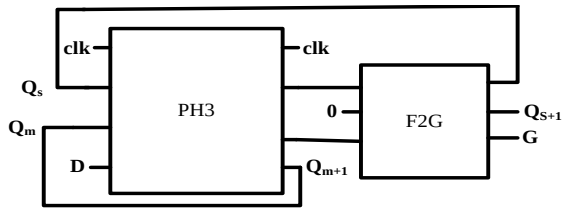


Fig. 13: Proposed parity preserving reversible master-Slave D Flip-flop

Table 4: comparison of the proposed parity preserving reversible master-Slave D Flip-flop with the existing counterparts

master-Slave D Flip-flop	Existing [8]	Existing [12]	Existing [17]	Existing [19]	This work
Constant input	2	2	2	3	**1**
Gate Counts	5	5	4	3	**2**
Garbage Output	3	3	2	2	**1**
Parity preserving	no	no	no	no	**yes**

4.3.2 Proposed write enable reversible master slave D flip-flop

Reversible memory cell with writing control line to design memory, so that the designed reversible flip-flop have the ability to control writing, i.e., with controlling line like "w" the new input data can be controlled, as in Fig. 14. According to the figure, a FRG gate is added to control flip-flop input. The new input would not enter flip flop till the "w" is zero and flip-flop and consequently the output would remain unchanged once "w" is one, the new input enters the flip-flop and the output will change in descending of clk as mentioned above flowingly.

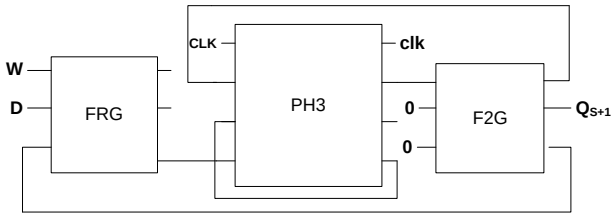


Fig. 14: proposed design of Parity preserving write enable reversible master-Slave D Flip-Fop

Table 7 shows the results of comparing proposed write enable reversible master slave D flip-flop with A.S.M. Sayem et al. (2009) [10] and Md. Selim et al. (2012) [18]. It can be verified that the number of gate counts in [10] and [18] is 7 and 5 respectively and for proposed design is only 3. The number of garbage outputs in [10] and [18] is 7 and 3 respectively and for this work is only 3.

Table 5: comparison of proposed reversible write enable master slave D flip-flop with existing work

Write enable master slave D flip-flop	Gate counts	Garbage outputs	Parity preserving
Existing [10]	**7**	**7**	**No**
Existing [18]	**5**	**3**	**No**
Proposed design	**3**	**3**	**YES**

The general design of reversible memory cell is illustrate in Fig. 15. As is shown in this figure, the memory cell contains three input line and three output lines. Three input lines include D, as input of memory cell, clk as a clock, and W, the writing control line.

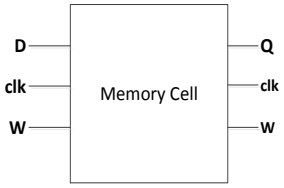


Fig. 15: The general design of proposed parity preserving reversible memory cell

5. PROPOSED DESIGN OF PARITY PRESERVING REVERSIBLE RANDOM ACCESS MEMORY (PPRRAM)

After designing different parts of a PPRRAM, it can be designed by merging those parts. The general design of a PPRRAM is shown in Fig. 16. Stating that since the completive design for the parts of memory, in addition to having the reversibly feature have parity preserving feature, in the general design this point has been considered. So the designed memory is a parity preserving reversible random access memory. This memory is like $2^n \times m$, in which 2^n is the address line and the input data is m bit. According to the figure, the parity preserving reversible decoder used for selection of particular row of memory, so, 2^n line of memory can be addressed. A $2^n \times 1$ Multiplexer is used to situate cell

content in the output which 2^n is the number of address lines. The number of Multiplexers is m which is the number of input data bits. So in whatever time the content of an n bit word from 2^n address line of memory cells in the output can be situated.

There is another point regarding the figure, in which F2G is used instead FG to expand input data. This is due two reasons. Firstly, the present design is parity preserving and secondly since the output of decoder is power of 2, and with each F2G two expansions is created, the exact number of applied F2G, is the half of the number of decoder outputs, that is 2^n and there will be one garbage output. Finally, due to the structure of design of multiplexer, the number of input data bits can be increased when needed to do so, it is necessary to put a $2^n \times 1$ bit multiplexer in the output.

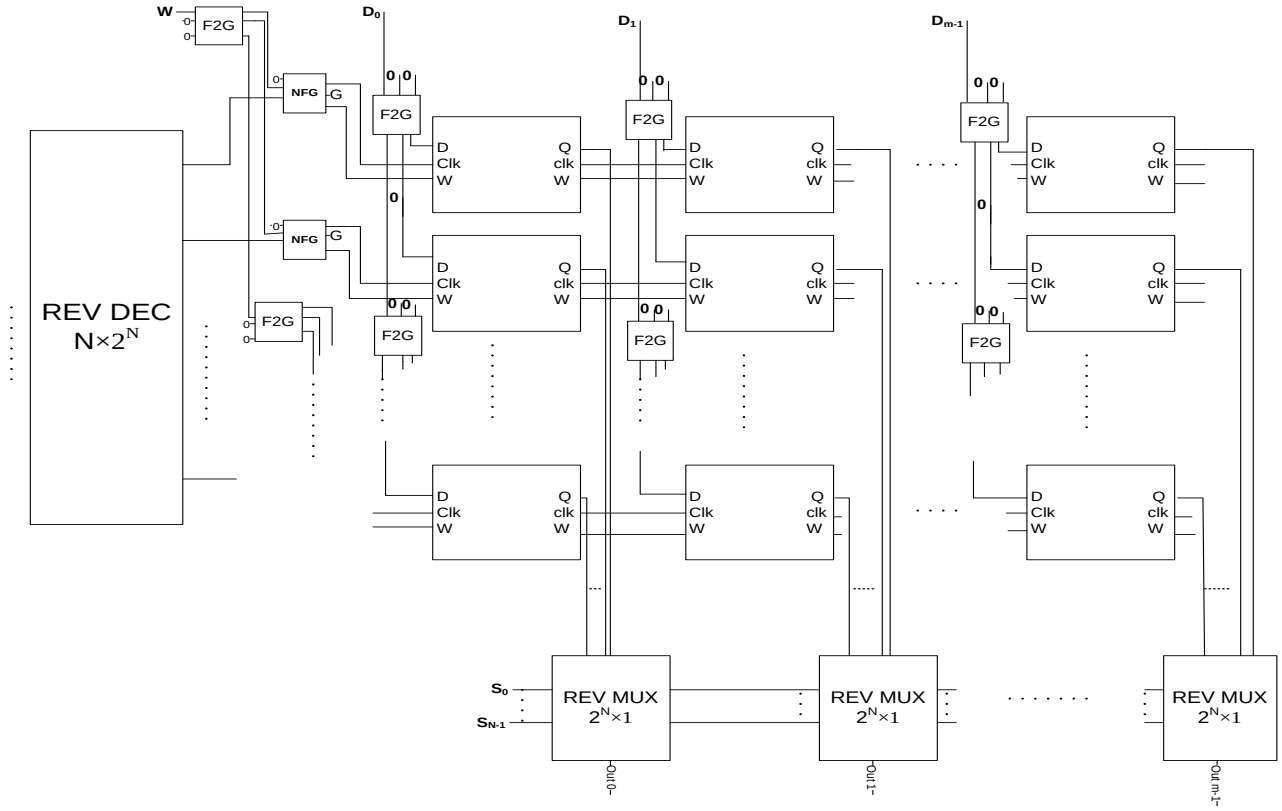


Fig.13: proposed design of $2^n \times m$ parity preserving reversible random access memory (PPRRAM)

Theorem 1: Let gc be the number of gates required to implementation a $2^n \times m$ parity preserving reversible random access memory (PPRRAM), where n be the number of bits and m be the input data bits in the PPRRAM, then $gc \geq 2^{n-1}(9m+5)-m-1$.

Proof: A $2^n \times m$ PPRRAM requires m mux whereas each MUX requires 2^n-1 parity preserving reversible gate. Moreover an $n \times 2^n$ decoder is required and 2^n-1 gates for that decoder. $2^n \times m$ FFs are required for memory cells whereas each FF required 3 parity preserving reversible gates. 2^n NFG and 2^{n-1} F2G are required to perform AND operation and copy operation respectively. $2^{n-1} \times m$ gates are required to perform copy operation of inputs. Let gc be the minimum number of gates to realize the PPRRAM, so $gc \geq m*(2^n-1)+2^n-1+3*m*2^n+2^n+2^{n-1}+m*2^{n-1}$ Hence, $gc \geq 2^{n-1}(9m+5)-m-1$

Theorem2: let gr be the number of garbage outputs which produces in implementation of $2^n \times m$ parity preserving reversible random access memory (PPRRAM), then $gr \geq 2^n*(4m+4)+m*n$.

Proof: A $2^n \times m$ PPRRAM requires $m*2^n$ MUX and according to Table 3 each MUX produces 2^n+n-1 garbage bits. An $n \times 2^n$ decoder, according to Table 4, generates 2^n-1 garbage bits. In the last column, 2^{n+1} garbage bits are generated for clk and W. FFs and NFT gates generate $3*m*2^n$ and 2^n garbage bits respectively (according to Table 5 each FF generates 3 garbage bits and $m*2^n$ FFs are required to realize $m \times 2^n$ PPRRAM). Moreover m+1 garbage outputs is generated for F2Gs. If gr be the minimum number of garbage outputs generated from PPRRAM, then $gr \geq m*(2^n+n-1)+2^n-1+2^{n+1}+3*m*2^n+m+1$ hence, $gr \geq 2^n*(4m+4)+m*n$.

6. CONCLUSION

In this work a new parity preserving reversible random access memory is designed. Its component such as multiplexer, decoder and memory cell also are proposed in parity preserving and reversible form. The general design of those components is introduced. Designs were compared with exiting works and efficiency of designs in gate counts and garbage outputs is shown. PH3 is also introduced which can be used in various reversible circuits. In this work PH3 is used to designing of parity preserving reversible master slave D flip-flop and memory cell. Proposed master slave D flip-flop is compared with existing work and its efficiency in gate counts, garbage outputs and constant inputs is showed.

7. ACKNOWLEDGMENTS

The Author would like to express especial thanks to the anonymous reviewers for their critical suggestions which helped in improving the manuscript.

8. REFERENCES

[1] Landauer, R., 1961, "Irreversibility and heat generation in the computing process".IBM J. Res. Develop., 5: 183-191.

[2] Bennett, C.H., 1973. Logical reversibility of computation, IBM Journal of Research Development, 17: 525-532.

[3] Fredkin, E.And T. Toffoli, 2006. "Conservative logic," Int'l J. Theoretical PHysics, 21: 219-253.

[4] Wein furter,1995, Elementary gates for quantum computation,PHys. Rev., A 52(5): 3457-3467.

[5] M. Haghparast and K. Navi, A new parity preserving reversible gate for nanotechnology based systems, Am. J.Applied Sci., 5 (2008) 519

[6] Majid Haghparast and KeivanNavi, 2010. Novel Reversible Parity preserving Error Coding and Detection

[7] Parhami, B., 2006. "Parity preserving Reversible Circuits" Proc. 40th Asilomar Conf. Signals, Systems, and Computers, Pacific C.A. Grove,

[8] Chuang,M.-L. And Wang, C.-Y. 2008. Synthesis of reversible sequential elements. ACM J. Emerg. Technol. Comput. Syst. 3, 4, Article 19 (January 2008)

[9] H. Thapliyal, M. B. Srinivas, and M. Zwolinski," A beginning in the reversible logic synthesis of sequential circuits",In Proc. The Military and Aerospace Programmable Logic Devices Intl. Conf., Washington, Sept. 2005.

[10] A.S.M. Sayem,M.M.A. Polash, H.M.H. Babu,"Design of a reversible logic block of FPGA", proceedings of silver Jubilee Conference on Communication Technologies and VLSI design (commv'09), VIT University, Vellore India.Oct. 8- 10, 2009, pp: 501- 502.

[11] H. Thapliyal and A. P. Vinod, "Design of reversible sequential elements with feasibility of transistor implementation" In Proc. The 2007 IEEE Intl. Symp. On Cir.and Sys., pages 625–628, New Orleans, USA, May 2007.

[12] Nayeem NM, Hossain MA, Haque MM, Jamal L, Babu HMH (2009). Novel Reversible Division Hardware. IEEE Int. Midwest Symp. Circuits Syst., 1134-1138.

[13] Abu Sadat Md. Sayem, Masashi Ueda Journal Of Computing, Volume 2, ISSUE 6, JUNE 2010, ISSN 2151-9617.

[14] J. E. Rice, A New Look at Reversible Memory Elements. Proc. Int. Symp. Circuits and Systems (ISCAS), Kos, Greece, IEEE, Piscataway, NJ. (2006) 1243.

[15] J. E. Rice, An Introduction to reversible latches, The Computer Journal, 51 (2008) 700.

[16] Siva Kumar SastryHari, ShyamShroff, Sk. Noor Mahammad, V. Kamakoti, "Efficient building blocks for reversible sequential circuit design", Circuits and Systems, 2006, MWSCAS '06, 49th IEEE International Midwest Symposium on August 2006, ISSN: 1548-3746.

[17] R. Thapliyal, N. Ranganathan. Design of Reversible Latches Optimized for Quantum Cost, Delay and Garbage Outputs, vlsid, pp.235- 240, 2010 23rd International Conference on VLSI Design, 2010

[18] Md. Selim Al Mamun, Syed MonowarHossain, "Design of Reversible Random Access Memory" International Journal of Computer Applications (0975 – 8887) Volume 56– No.15, October 2012

[19] Bhagyalakshmi H R, Venkatesha M K, "Design of Sequential Circuit Elements Using Reversible Logic Gates", World Applied Programming, Vol (2), Issue (5), May 2012. 263-271.