A Low Power Fault Tolerant Reversible Decoder Using Verilog HDL

Chintakindi Mahipal
M Tech Student
JNIT College, Hyderabad.

P Anjaiah, M.Tech
Assistant Professor
JNIT College, Hyderabad.

ABSTRACT:

This paper demonstrates the reversible logic synthesis for the n-to-2n decoder; where n is the fault tolerant Fredkin and Feynman double gate. Thus, the entire scheme inherently becomes fault tolerant. Algorithm for designing the generalized decoder has been presented. In addition, several lower bounds on the number of constant inputs, garbage outputs and quantum cost of the reversible fault tolerant decoder have been proposed. Transistor simulations of the proposed design are shown in micro wind 3.0 version where power area and delay are calculated.

Keywords: Decoder, Delay, Garbage Output, Low Power Design, Quantum Cost, Reversible & Fault Tolerant Computing.

INTRODUCTION

Reversible Logic has gained importance in the recent past. The rapid decrease in the size of the chips has lead to the exponential increase in the transistor count per unit area. As a result, the energy dissipation is becoming a major barrier in the evolving nano-computing era. Reversible logic ensures low energy dissipation R.W. Keyes et.al (1970), C.H. Bennet (1988). An operation is said to be physically reversible if there is no energy to heat conversion and no change in entropy. In reversible logic, the state of the computational device just prior to an operation is uniquely determined by its state just after the operation. In other words, no information about the computational state can ever be lost. Hence the reversible logic can be viewed as a deterministic state machine. R.Landauer (1961) has shown that for every bit of information that is erased during an irreversible logic computation $kTln2$ joules of heat energy is generated, where k is the Boltzmann constant and T is the temperature in Kelvin at which the system is operating. C.H.Bennett (1973) showed that the $kTln2$ amount of energy dissipation would not occur if a computation is carried out in a reversible way. Computations performed by the current computers are commonly irreversible, even though the physical devices that execute them are fundamentally reversible. At the basic level, however, matter is governed by classical mechanics and quantum mechanics, which are reversible. With computational device technology rapidly approaching the elementary particle level, it has been argued many times that this effect gains in significance to the extent that efficient operation of future computers requires them to be reversible C.H.Bennett (1988), R.Landauer (1961). Hence, reversible logic is gaining grounds. A reversible gate is a logical cell that has the same number of inputs and outputs. Also, the input and output vectors have a one-to-one mapping. Direct fan-outs from the reversible gate are not permitted. Feedbacks from gate outputs to inputs are not allowed. A reversible gate with n inputs and n-outputs is called a n x n reversible gate. Several reversible gates have been designed till date that is discussed below. The Feynman gate R.Feynman (1985) shown in Fig.1 is one of the popular example of a 2x2 reversible gate. In this gate the first input is passed to the output without any change and the second output is the XOR of the first and second inputs. Toffoli Gate T.Toffoli (1980) is a n x n universal reversible gate. The first n – 1 input are directly passed to the corresponding outputs and the nth output is the logical XOR of the nth input with the logical AND of all first n – 1 inputs. The 3x3 Toffoli gate is shown in Fig.2. E.Fredkin (1982) proposed a 3x3 universal reversible gate. The Fredkin gate functions as a multiplexer with the select signal as the first input. Functionality of the Fredkin gate is

shown in Fig.3. Many other gates have been proposed that include Kerntopf gate by P.Kerntopf (2002) and Margolus gate by N.Margolus (1988). An elaborate list of reversible gates studied in the literature is presented in P.Kerntopf (2002).
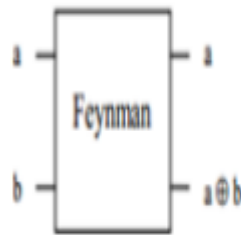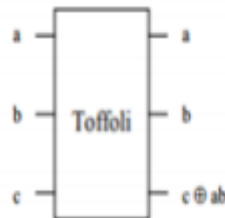

Fig.1. Feynman Gate.
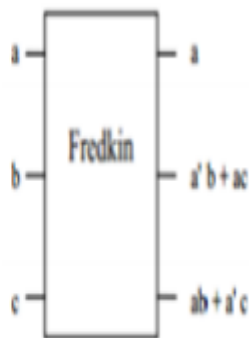

Fig.2. 3×3 Toffoli Gate.

Fig.3. Fredkin Gate.

II. EXISTING SYSTEM AND PROPOSED SYSTEM

Low Power digital CMOS becomes more and more interesting, due to the general advances in process technology and new low power applications. The advancements in the field of CMOS technology have promoted a continuous increase in the density of integration as well as in the frequency of operation of the VLSI ICs. As technology advances push for smaller devices and faster operations, power consumption and noise become severe problems when designing high speed ICs. Over the past decade, power consumption of VLSI chips has been continuously increasing. The need for low-power design is becoming an essential parameter in high-performance digital systems. There are numerous techniques being encountered for the design of low power VLSI circuits. Low power has made an important note that power dissipation has a consideration on performance and area. Static power and Dynamic power being the main components determining the power consumption in CMOS circuits.

A. Existing System

Reversible Fault Tolerant Gates: A Reversible gate is a logical cell that has the same number of inputs and outputs. Also, the input and output vectors have a one-to-one mapping. Direct fan-outs from the reversible gate are not permitted. Feedbacks from gate outputs to inputs are not allowed. A reversible gate with n-inputs and n-outputs is called a n x n reversible gate. Mainly Reversible Fault Tolerant Gates are discussed two gates. They are:

• Feynman Double Gate

•Fredkin Gate

For these reversible fault tolerant gates, we can calculate quantum computing and q-bit value.

1.Feynman Double Gate:
The Feynman double gate is a reversible gate. It is a 3*3 reversible logic gate and it defines as Iv = (A, B, C) and its output as Ov = (A, A xor B, A xor C). It is also a Universal Reversible gate. It takes three inputs and it produces three outputs. There is no feedback between input and output vectors. In Feynman Double gate we can use only two Xoroperations. Feynman Double gate operation follows as, if input Iv = (A, B, C) and its output performs as:

• The First output is equal to First input.

• The Second output is equal to Xor of First and Second input variables.

• The Third output is equal to Xor of First and Third input variables.

Fig.4 Block diagram of Feynman Double gate.

From the above fig.4, we can use two Xor operations i.e.., for the second and three outputs. Based upon the fig.5, we can design the quantum equalization circuit

Fig.5. Quantum Equivalent Realization Circuit.

From the above circuit, we can calculate the Quantum cost and Hardware Complexity i.e. Quantum cost is defined as no of operations performing with in the circuit. Here we can use two Xor operations, so the Quantum cost is equal to 2. For the design procedure we can require twelve transistors to Feynman Double gate (F2G) respectively. Its design is bases upon P_{mos} and $\bar{N}_{mos}$ transistors as shown in Fig.6.
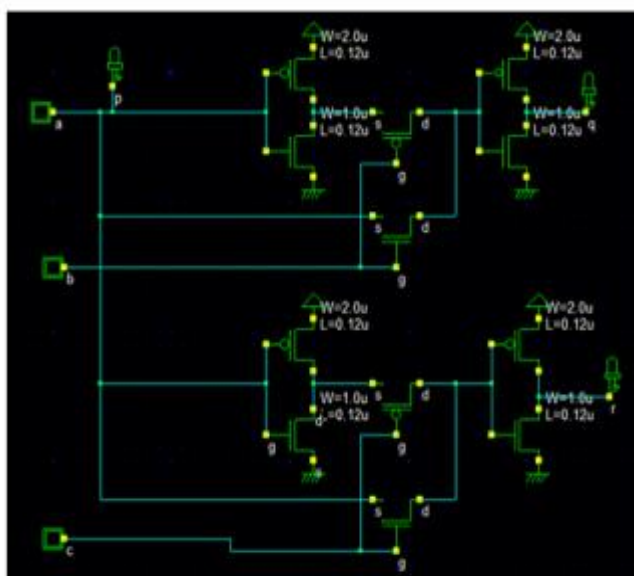

Fig.6. Transistor Realization of F2G.

It can be designed by using the micro wind tool. Here input vectors take the binary values from 0 to 7 and its output is based upon the operations. This is a parity preserving property allows to detect a faulty signal from the circuit's primary output and it takes Even or Odd priority. Below Table shows the Truth Table of Feynman Double gate.

TABLE I: Truth Table for Feynman Double gate

INPUT			OUTPUT			PARITY
A	B	C	P	Q	R	(EVEN/ODD)
0	0	0	0	0	0	EVEN
0	0	1	0	0	1	ODD
0	1	0	0	1	0	ODD
0	1	1	0	1	1	EVEN
1	0	0	1	1	1	ODD
1	0	1	1	1	0	EVEN
1	1	0	1	0	1	EVEN
1	1	1	1	0	0	ODD

2.Fredkin Gate: The Fredkin 3-bit gate is universal for computational logic, and is reversible. Classically, it is impossible to do universal computation using reversible 2- bit gates only. It is also known as Controlled swap gate (CSWAP) and it is also universal for computation. The Fredkin gate is the reversible three-bit gate that swaps the last two bits if the first bit is 1 .It is a 3*3 reversible logic gate and it defines the input vectors as Iv = (A, B, C) and it produces the output vectors are Ov= (A, A'B Xor AC, A'C Xor AB).There is no feedback between input and output vectors.

Fredkin gate operation follows as Iv = (A, B, C) and its operation follows as:

• The First output is equal to First input.

• The Second output is equal to Inversion of First input with Second input and it is Xor of First and Third inputs.

• The Third output is equal to Inversion of First input with Third input and it is Xor of First and Second inputs.

 From the above figure, we can use two Xor operations and 2 AND gates i.e.., for the second and third outputs. Based upon the above figure, we can design the equivalent circuit.
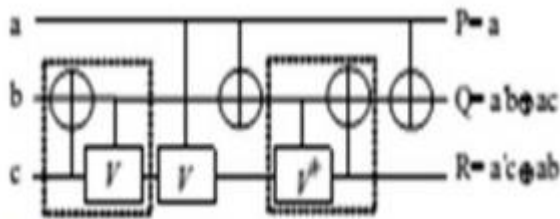
Fig.7. Quantum Equivalent Realization circuit.

From the above circuit, we can calculate the Quantum cost and Hardware complexity i.e.., Quantum cost is based on the no of operations performing with in the circuit. Here, total quantum cost is equal to five. In Fredkin gate, we can use two Xor, four and, two not gate operations are used. From the above fig.7, each rectangle is equivalent to 2*2 quantum primitives, therefore its quantum cost is considered to be one. To realize the Fredkin gate, only four transistors are needed. To design this circuit it is based on PMOS and NMOS transistors.
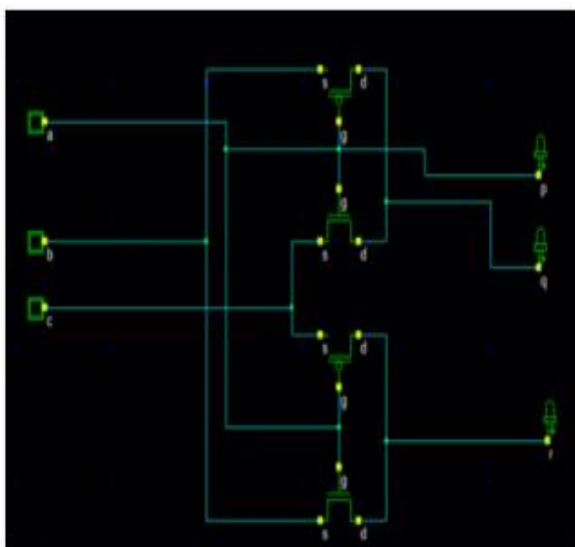

Fig.8. Transistor Realization of Fredkin gate.

The above Fig.8 is designed by using micro wind tool. The Fredkin gate operation is based on its binary values i.e..,(0 and 1) from 0 to 7. This is a parity preserving property allows to detect a faulty signal from the circuit's primary output and it takes Even or Odd priority. Below Table shows the truth table for Reversible Fredkin gate.

TABLE II: Truth Table for Fredkin gate

INPUT			OUTPUT			PARITY
A	B	C	P	Q	R	(EVEN\ODD)
0	0	0	0	0	0	EVEN
0	0	1	0	0	1	ODD
0	1	0	0	1	0	ODD
0	1	1	0	1	1	EVEN
1	0	0	1	0	0	ODD
1	0	1	1	1	0	EVEN
1	1	0	1	0	1	EVEN
1	1	1	1	1	1	ODD

Disadvantages:
• Feed back is strictly restricted.
• Maximum and Minimum fan-out is always one.
• Low area and less power dissipation.
• Low speed and High delay.

B. Proposed System
Reversible Fault Tolerant Decoders:

Reversible Decoder is used to reduce the power dissipation in the decoder circuit; where as conventional decoder is used as normal decoder. A decoder is a combinational circuit used in many devices for processing. Reversible Decoder is used to overcome the drawbacks of conventional decoder. In Conventional Decoder, the output is not dependent on the number of inputs and it always less than the number of inputvectors, where as in reversible decoder the output is always dependent on the input vectors. Reversible Fault Tolerant Decoders are plays important role on reversible logic gates. These Decoders are also find out the where the fault occurs either at input or output side.

Reversible Fault Tolerant Decoders are mainly classified in to three types:
• 1-to-2 Reversible Fault Tolerant Decoder(RFD)
• 2-to-4 RFD
• 3-to-8 RFD

Proposed 1-To-2 Reversible Faults Tolerant Decoder:
Proposed 1-to-2 RFD can be designed by using Reversible Fault Tolerant gates. This decoder is

similar to normal reversible decoder. In proposed 1-to-2 RFD, we can use single Feynman gate as shown in Fig.9 .We can use 2*2 Reversible Feynman double gate for Reversible Fault Tolerant decoders. In this gate, it requires one constant input and two primary inputs and it produces two constant outputs and one garbage (unused) output. This decoder also follows the same principle of normal decoder i.e.., it is a principle of a collection of logic gates fixed up in a specific way such that for an input combination, all output terms are low expect one and these terms are the Min-terms. In this decoder, it follows the "n" inputs and it produces "2^n" outputs.
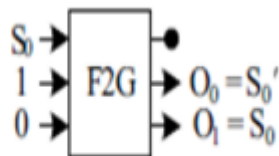
Fig.9. Block Diagram for proposed 1-to-2 RFD.

In this decoder, we can have less quantum cost and less no of operations are performed in the circuit. This is also a simple Reversible Decoder.

Proposed 2-To-4 Reversible Fault Tolerant Decoder:

2-to-4 RFD can be designed by using the Reversible Logic gates. Normally, Reversible Fault Tolerant Decoders are used the Fault Tolerant gates. In 2-to-4 RFD, we can use the both reversible gates. It will be designed by using the output of 1-to-2 RFD as shown in Fig.10. Normally, in all reversible decoders we can use single Feynman double gates. Because, In Feynman Double gate it requires a more no of transistors and low quantum cost, delay will be high. To overcome, these drawbacks we can use the Fredkin gates. By using Fredkin gates in reversible decoder, it produces high quantum cost and less delay. In 2-to-4 RFD design procedure, it requires single Feynman double gate and two Fredkin gates respectively. Mainly, in normal 2-to- 4 decoder has 4 different 2*2 logical AND Operations. A Reversible Fault Tolerant AND operation requires at least "3" quantum cost. But, in 2-to-4 Reversible Fault Tolerant Decoder is realized with at least 12 quantum cost. In this decoder, we can

have two constant inputs (S0, S1) and four primary inputs and it produces the four constant outputs.
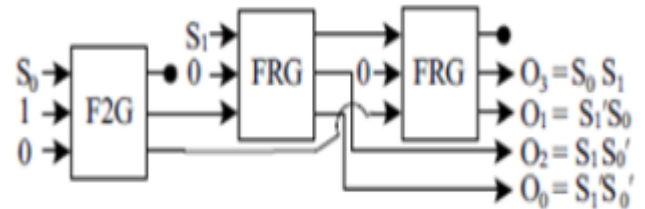

Fig.10. Block diagram of Proposed 2-to-4 RFD.

Proposed 3-To-8 Reversible Fault Tolerant Decoder:

3-to-8 RFD also designed by using the Fault tolerant gates. In 3-to-8 RFD, we can use the both reversible fault tolerant gates. It can be designed by using the output of 2-to-4 RFD. Normally, in all reversible decoders, it requires only single Feynman double gates as shown in Fig.11. In Feynman Double gate it requires a more no of transistors and low quantum cost, delay will be high. To overcome, these drawbacks we can use the Fredkin gates. By using Fredkin gates in reversible decoder, it produces high quantum cost and less delay. For the design procedure of 3-to-8 RFD, it realizes four Fredkin gates respectively. In 3-to-8 RFD, it requires only Fredkin gates and output of 2-to-4 RFD. It follows the same principle of Normal decoder i.e.., it takes "3" data inputs and produces the 2^n i.e..,(2^3) is equal to "8" outputs.
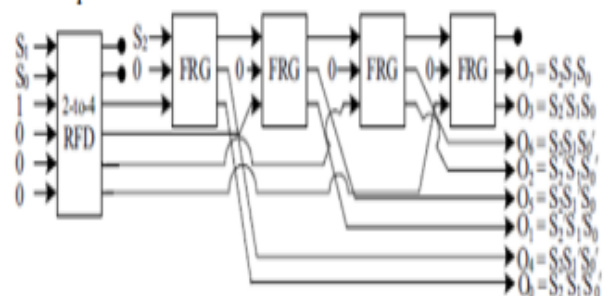

Fig.11. Block Diagram of proposed 3-to-8 Reversible Fault Tolerant Decoder.

TABLE III: Comparison of Reversible Fault Tolerant Decoders

Circuit Name	No. of gates (GT)	Garbage Output(GO)	Quantum Cost(QC)	Hardware Complexity(HC)	Unit Delay(UD)
2-to-4 Existing Circuit	3	2	15	$6\alpha + 12\beta + 6\gamma$	3
2-to-4 Proposed Circuit	3	2	12	$6\alpha + 8\beta + 4\gamma$	2
3-to-8 Existing Circuit	≥ 7	≥ 3	≥ 35	$\geq 14\alpha + 28\beta + 14\gamma$	≥ 7
3-to-8 Proposed Circuit	7	3	32	$14\alpha + 24\beta + 12\gamma$	4

In this reversible decoder, we can use both AND, NOT and XOR gates. In 3-to-8 RFD, we can have at least "14" quantum cost. In this decoder, we can have the three constant inputs and eight primary inputs and produces the eight constant outputs and one garbage output. According to the proposed designs, it is based on n-to-2^n Reversible Fault Tolerant Decoder. We present the transistor representations of FRG and F2G using MOS transistors. Each of the proposed circuit is simulated with DSCH-2.7. Below Table shows the comparative study of the proposed fault tolerant decoders with existing fault tolerant one.
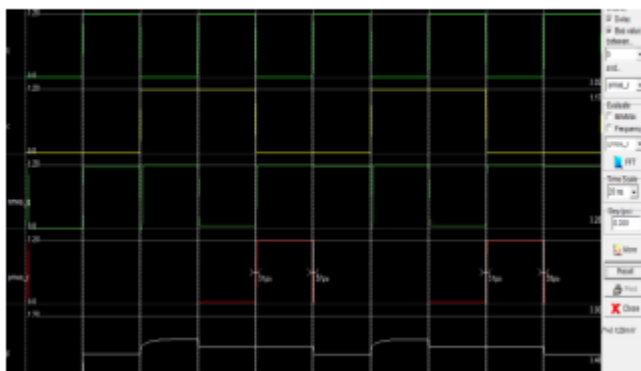
III. SIMULATION RESULTS

Fig.12 Timing Waveform of Feynman Double Gate.

Power = 0.125mW.Here, in this reversible gate it takes more no of transistors and it calculates delay and frequency as shown in Fig.12. Delay will be very high and frequency is calculated at 5 GHz. This reversible gate is compare with Fredkin gate.

Fig.13. Timing Waveform of Fredkin Gate.

Here, the power dissipation is very less compare to Feynman double gate. For the design procedure, the Fredkin gate has less no of transistors compare to Feynman double gate as shown in Fig.13.
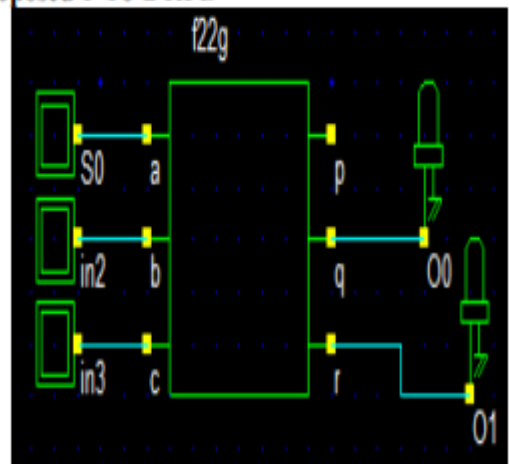
C. Proposed 1-To-2 RFD

Fig.14. Block Diagram for Proposed 1-to-2 RFD.

The proposed 1-to-2 RFD can be designed by using the Digital Schematic (DSCH) with micro-wind 3.7 version tool. In this software, requires all P-mos, N-mos and all basic gates as shown in Fig.14 and 15. Using DSCH3, the Block diagram of 1-to-2 RFD is verified at logic level i.e.., (FILE ->MAKE VERILOG FILE) and then we can generate the code.
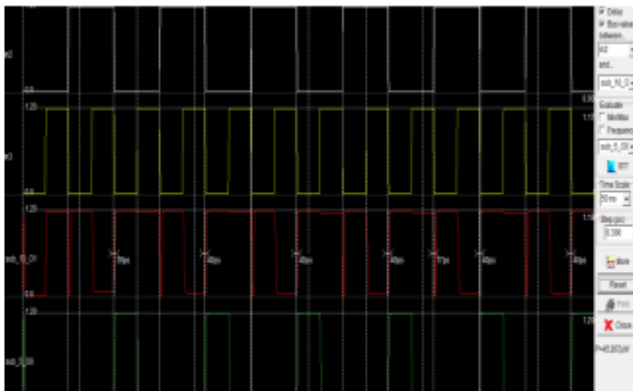
Fig.15. Timing Waveform of Proposed 1-to-2 RFD.

It can be simulated from the Export micro-wind tool and it is 120nm CMOS technology, its waveform is calculated at 50ns time scale and power is 45.207micro watts as shown in Fig.16.
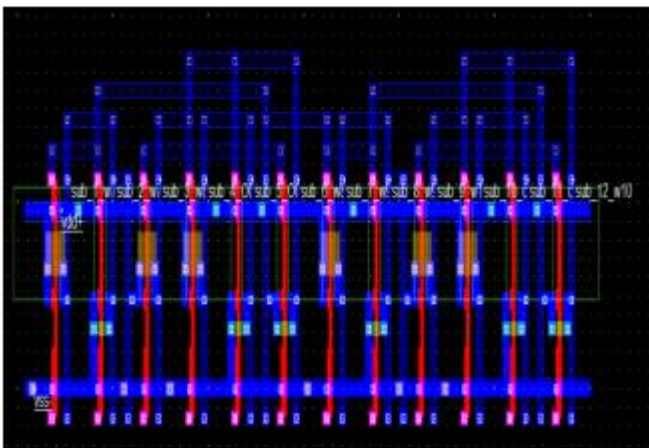

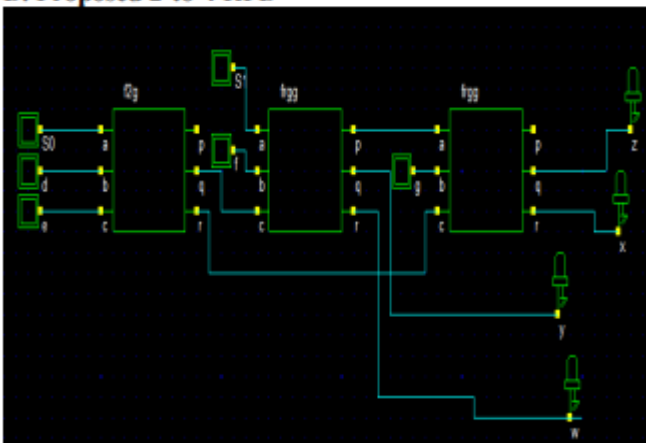Fig.16. Layout for Proposed 1-to-2 RFD.

D. Proposed 2-to-4 RFD

Fig.17. Block Diagram for Proposed 2-to-4 RFD.

The Proposed 2-to-4 RFD also is designed by using DSCH3 with micro-wind version tool. Using DSCH3, the Block diagram of 1-to-2 RFD is verified at logic level i.e..,(FILE ->MAKE VERILOG FILE) and then we can generate the code as shown in Fig.18. In this decoder we can use both F2G and Fredkin gates respectively.
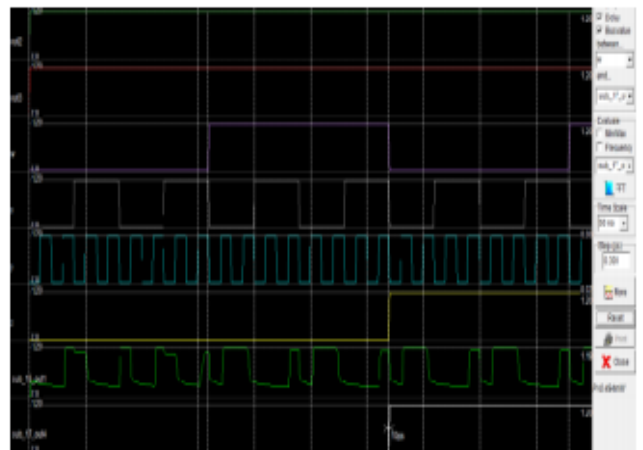

Fig.18. Timing Waveform of Proposed 2-to-4 RFD.

It can be simulated from the Export micro-wind tool and it is 120nm cmos technology and it is compiled from verilog file and its power is 49.4micro watts as shown in Fig.19.
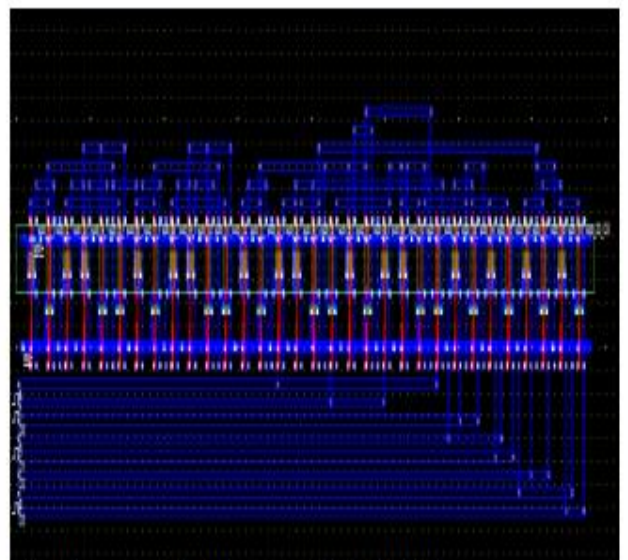

Fig.19. Layout for Proposed 2-to-4 RFD.
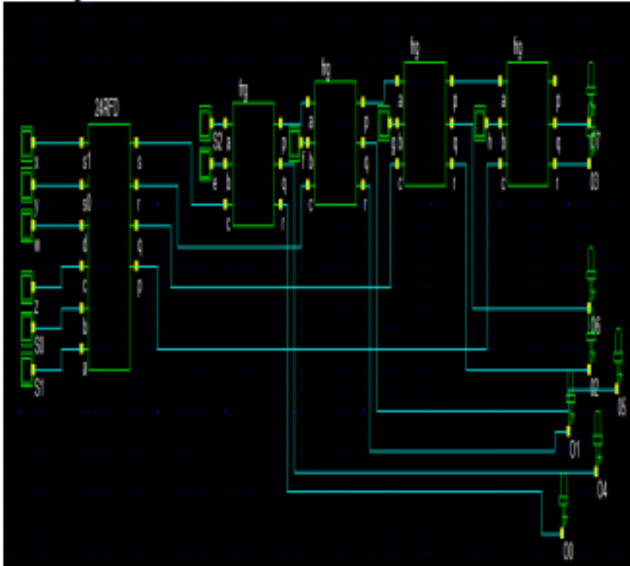
E. Proposed 3-to-8 RFD
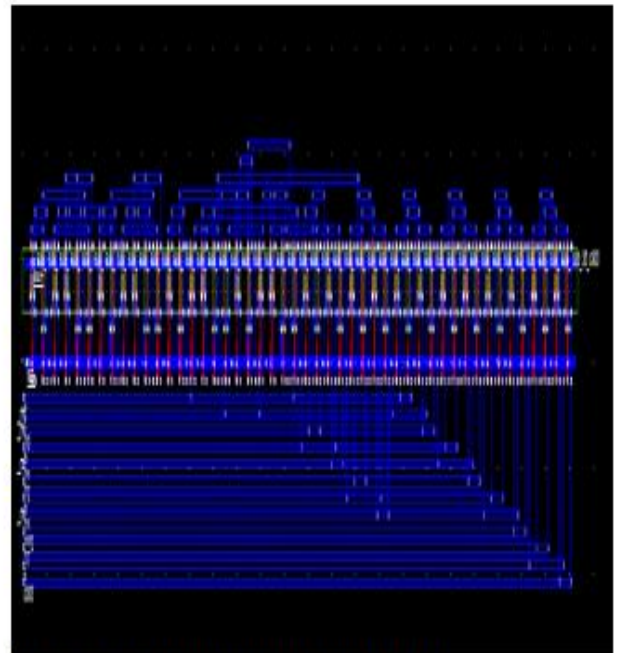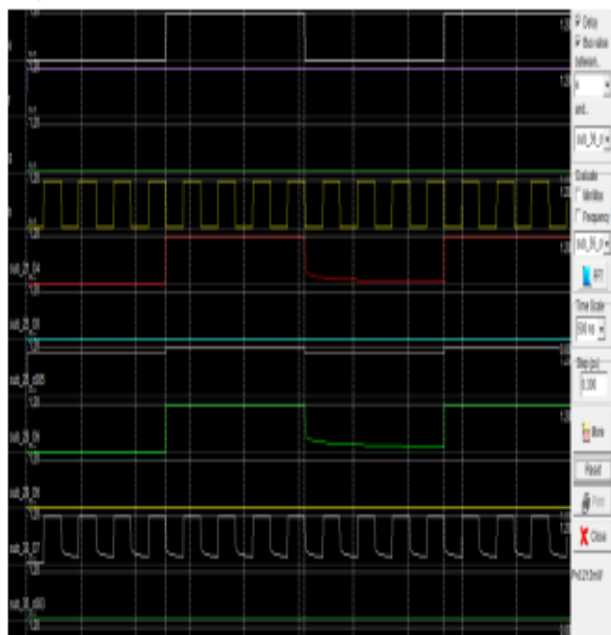

Fig.20. Block Diagram for Proposed 3-to-8 RFD.

Fig.21. Timing Waveform of Proposed 3-to-8 RFD.

It can be simulated from the Export micro-wind tool and it is 120nm cmos technology and it is compiled from verilog file and its power is 21.3 microwatts. Its power is reduced compare to 2-to-3 RFD as shown in Figs.20 to 22.

Fig.22. Layout for Proposed 3-to-8 RFD.

IV. CONCLUSION AND FUTURE WORK
Conclusion

This Paper presents different reversible decoders by using Digital Schematic. A design and Implementation of Reversible Fault Tolerant Decoder was presented. In this paper, we presented the design methodologies of an n-to-2n reversible fault tolerant decoder, where n is the number of data bits. This Project helps to understand the complete functional verification process of Micro-wind tool for the higher quality verification. We proposed several lower bounds on the numbers of garbage outputs, constant inputs and quantum cost and proved that the proposed circuit has constructed with the optimum garbage outputs, constant inputs and quantum cost. In addition, we presented the designs of the individual gates of the decoder using MOS transistors in order to implement the circuit of the decoder with transistors. Simulations of the transistor implementation of the decoder showed that the proposed fault tolerant decoder works correctly. Proposed reversible fault tolerant decoders can be used in parallel circuits, multiple-symbol differential detection, network components and in digital signal processing.

Future Work

This project used Micro-wind i.e.., the technology is used to dedicate micro electronics and nano technology for ASIC and custom IC design and simulation as well as the latest in electronic designs. In the coming future, the adders, multiplexers and complicated circuits can be done by using DSCH.

V. REFERENCES

[1] C. H. Bennett, "Logical reversibility of computation," IBM J. Res. Dev., vol. 17, no. 6, pp. 525–532, Nov. 1973. [Online]. Available: http://dx.doi.org/10.1147/rd.176.0525.

[2] M. Nielsen and I. Chuang, Quantum computation and quantum information. New York, NY, USA: Cambridge University Press, 2000.

[3] M. Perkowski, "Reversible computation for beginners," 2000, lecture series, 2000, Portland state university.[Online].Available: http://www.ee.pdx.edu/mperkows.

[4] S. N. Mohammed and K. Veezhinathan, "Constructing online testable circuits using reversible logic", IEEE Transactions on Instrumentation and Measurement, vol. 59, pp. 101–109, 2010.

[5] D. Maslow, G. W. Dueck, and N. Scott, "Reversible logic synthesis benchmarks page," 2005. [Online]. Available: http://webhome.cs.uvic. Ca/d Maslow.

[6] R. K. James, T. K. Shahana, K. P. Jacob, and S. Sasi, "Fault tolerant error coding and detection using reversible gates," IEEE TENCON, 2007, 2007.

[7] E. Fredkin and T. Toffoli, "Conservative logic," International Journal of Theoretical Physics, vol. 21, no. 3, pp. 219–253, Apr. 1982. [Online].Available: http://dx. doi.org/10.1007/BF01857727.

[8] R. Feynman, "Quantum mechanical computers," Foundations of Physics, vol. 16, pp. 507–531, 1986, 10.1007/BF01886518. [Online]. Available: http://dx.doi. org/ 10.1007/BF01886518.

[9] DSCH: "Micro-wind and DSCH information page." [Online]. Available: http://www.microwind.org/