

VHDL Implementation of 4-Bit Full Adder Using Reversible Logic Gates

Patel Fenil S.¹ Tarunkumar C. Lad²

¹M.Tech (Purs.) ²M.Tech

¹CGPIT, Maliba Campus, UTU ²SVNIT, SURAT

Abstract--- A reversible gate has the equal number of inputs and outputs and one-to-one mappings between input vectors and output vector, thus the vector of input states can be always reconstructed from the vector of output states. A gate with k inputs and k outputs is called a k*k gate. Reversible logic Gate has one of the most important approaches for the power factor with its application in low power VLSI design. The Quantum computing having with applications in the domains like Nano-technology, Cryptography and Communications. It's also called as lossless circuits. In this paper addition operation is realized using reversible logic gates like DKG gate. Here we implement 4-Bit Full adder using DKG reversible logic gate.

Keywords: Reversible logic, Reversible gates, Quantum Computer, Power dissipation, Garbage, Full Adder.

I. INTRODUCTION

Every day and every time new technology is faster, smaller and more complex. Millions of gates used to perform logical operations are irreversible. Every time a logical operation is performed some information about the input is erased or lost and is dissipated as heat. With the help of Irreversible and Reversible logic gate Power consumption is the most important factors in VLSI circuit design.

Irreversible logic circuits/gates loss $kT \cdot \log_2$ Joule ($K = 1.3806505 \cdot 10^{-23} m^{-2} kg s^{-2} K^{-1}$ and T is the room temperature) heat for every bit of information [1]. Here we use Reversible logic gates to reduce heat dissipation. In 1973, Bennett showed that $KT \ln 2$ energy would not dissipate from a system as long as the system allows the reproduction of the inputs from observed outputs [2].

With the help of Reversible logic circuit input vector can be uniquely recovered from the output vector. And there is a one-to-one mapping between its input and output assignments, i.e. not only the outputs can be uniquely determined from the inputs, but also the inputs can be recovered from the outputs. Energy dissipation can be reduced or even eliminated if computation becomes Information-lossless.

Reversible logic design different types of combinational logic design approaches [3]. In reversible logic circuit the number of input lines must be equal the number of output lines, each

Output will be used only once and the resulting circuit must be acyclic [3]. A reversible circuit should be designed using minimum number of reversible logic gates Reversible circuit design; there are many parameters for determining the complexity and performance of circuits.

- 1) How many numbers of Reversible gates (N): The Number of reversible gates used in circuit [4].
- 2) The number of constant inputs (CI): Refers to the

number of inputs that are to be maintained constant at either 0 or 1 in order to given logical function [4].

- 3) The number of garbage outputs (GO): This refers to the number of unused outputs present in a reversible logic circuit [4].
- 4) Quantum cost (QC): This refers to the cost of the circuit in terms of the cost of a primitive gate. It is calculated knowing the number of primitive reversible logic gates (1*1 or 2*2) required to realize the circuit [4].

II. BASIC DEFINITIONS OF REVERSIBLE LOGIC GATES:

A. Reversible function [4]:

The multiple output Boolean function $F(x_1; x_2; \dots; x_n)$ of n Boolean variables is called reversible if:

The number of outputs is equal to the number of inputs;

- 1) Any output pattern has a unique pre-image.

B. Reversible logic gate:

Reversible Gates are circuits in which number of outputs is equal to the number of inputs and there is a one to one mapping between the vector of inputs and outputs [5]-[7]. It not only helps us to find the outputs from the inputs but also helps us to find the inputs from the outputs.

C. Garbage outputs:

Additional inputs or outputs can be added so as to make the number of inputs and outputs equal whenever necessary. This also refers to the number of outputs which are not used in the synthesis of a given function [6]. In certain cases these become mandatory to achieve reversibility. Garbage is the number of outputs added to make an n-input k-output function ((n; k) function) reversible [7].

Input + constant input = output + garbage.

D. Quantum cost:

Quantum cost refers to the cost of the circuit in terms of the cost of a primitive gate. It is calculated knowing the number of primitive reversible logic gates (1*1 or 2*2) required to realize the circuit. The quantum cost of a circuit is the minimum number of 2*2 unitary gates to represent the circuit keeping the output unchanged. The quantum cost of a 1*1 gate is 0 and that of any 2*2 gate is the same, which is 1 [9].

E. Design Constraints for Reversible Logic Circuits:

The following are the important design constraints for reversible logic circuits [4].

- 1) Reversible logic gates do not allow fan-outs [4].
- 2) Reversible logic circuits should have minimum

- quantum cost [4].
- 3) The design can be optimized so as to produce minimum number of garbage outputs [4].
- 4) The reversible logic circuits must use minimum number of constant inputs [4].
- 5) The reversible logic circuits must use a minimum logic depth or gate levels [4].

III. REVERSIBLE LOGIC GATES

There are many number of reversible logic gates that exist at present. The quantum cost of each reversible logic gate is an important parameter. The quantum cost of a 1x1 reversible gate is assumed to be zero while the quantum cost of a 2x2 reversible logic gate is taken as unity. Some of the important reversible logic gates are,

A. NOT Gate

The simplest Reversible gate is NOT gate and is a 1*1 gate [9]. The Reversible 1*1 gate is NOT Gate with zero Quantum Cost is as shown in the Fig.1 [9].

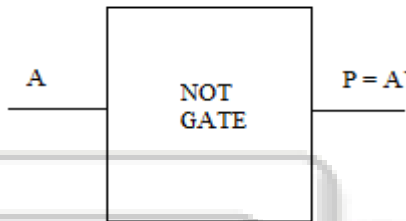


Fig. 1: NOT Gate

A	P
0	1
1	0

Table 1 :- 1 x 1 NOT Gate

B. CNOT Gate

CNOT gate is also known as controlled-not gate. It is a 2*2 reversible gate. The CNOT gate can be described as:

$$I/p = (A, B) ; o/p = (P= A, Q= A B)$$

Quantum cost of CNOT gate is 1[9]. Fig. 2 shows a 2*2 CNOT gate and its symbol [9].

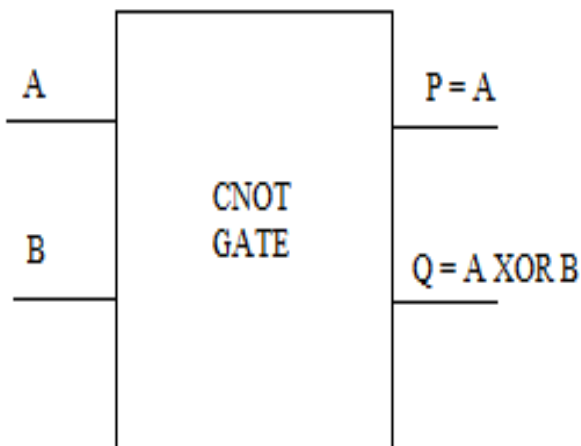


Fig. 2.: CNOT Gate

C. TOFFOLI Gate

TOFFOLI gate shown in fig. 3 which is a 3*3 gate with inputs (A, B, C) and outputs P=A, Q=B, R=AB XOR C. It has Quantum cost five [9].

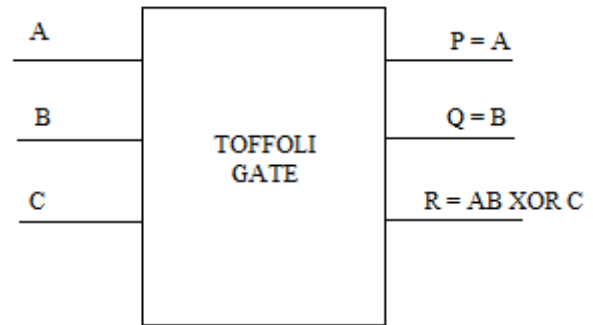


Fig. 3: Toffoli Gate

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

Table 2: - 3x3 Toffoli Gate

D. FEYNMAN Gate

The Feynman gate shown in fig. 4 which is a 2*2 gate and is also called as Controlled NOT and it is widely used for fan-out purposes. The inputs (A, B) and outputs P=A, Q= A XOR B. It has Quantum cost one [9].

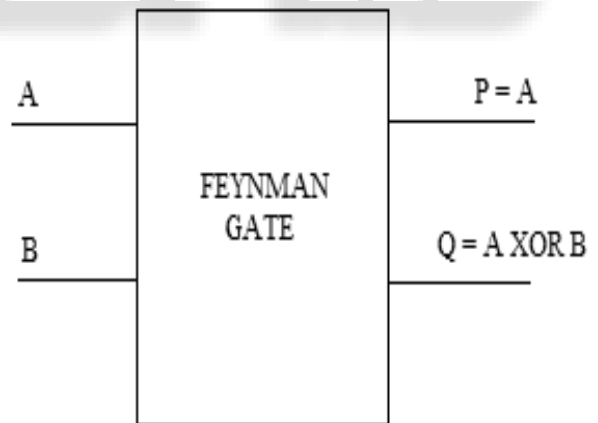


Fig. 4: Feynman Gate

A	B	P	Q
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Table 3: - 2 x 2 Feynman Gate

E. FREDKIN Gate

Fredkin gate shown in fig. 5 which is a 3*3 gate with inputs (A, B, C) and outputs P=A, Q=A'B+AC, R=AB+A'C. It has Quantum cost five [9].

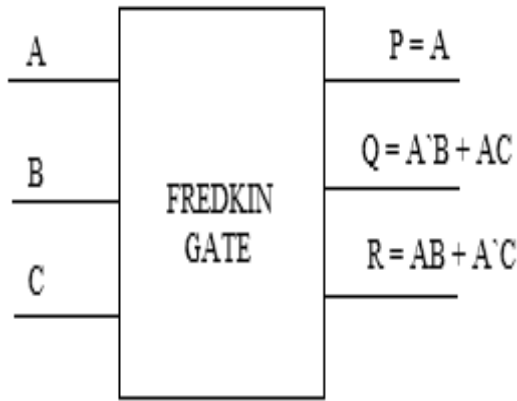


Fig. 5. Fredkin Gate

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

Table 4: - 3 x 3 Fredkin Gate

F. PERES Gate

Peres gate shown in fig. 6 which is a 3*3 gate having inputs (A, B, C) and outputs $P = A$; $Q = A \text{ XOR } B$; $R = AB \text{ XOR } C$. It has Quantum cost four [9].

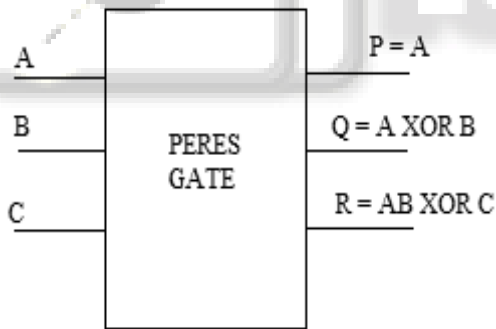


Fig. 6: Peres Gate

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

Table 5: - 3 x 3 Peres Gate

G. DKG gate

DKG gate shown in fig. 7 which is a 3*3 gate having inputs (A,B,C) and outputs $P = B$; $Q = A'C + AD'$; $R = (A \text{ XOR } B) (C \text{ XOR } D) \text{ XOR } CD$; $S = B \text{ XOR } C \text{ XOR } D$.

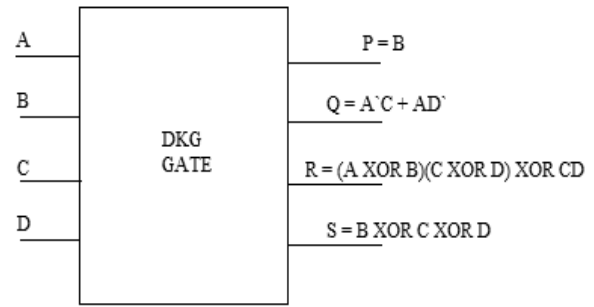


Fig. 7: Reversible DKG gate

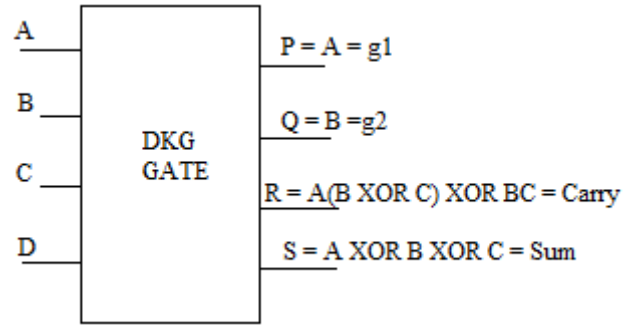


Fig. 8: DKG gate implemented as Full adder

IV. DESIGN AND IMPLEMENTATION:

A. Inputs

Here we implement 4-bit full adder. Thus the inputs are 4 bits. Three input vectors and a single bit Cin. Two of the three input vectors are added 4-bits values. The remaining vector could be called the ancilla vector which is filled with zeros.

B. Outputs

The outputs of the system are one garbage vector of 8 bits, one sum vector of 4 bits and a Cout (Carry out) bit.

C. Design Theory

First of all, we need to know that in order to build a reversible circuit we must use reversible gates [11]. There are different ways to implement a reversible adder. These different implementations depend on a balance between gates count, garbage outputs, ancilla bits. Being universal, these previously presented gates can implement any logical function and therefore they can also implement the well-known functions for a full-adder:

$$S = X'Y'Zin + X'YZin' + XYZin + XY'Zin'$$

$$S = X'Y'Zin + XYZin + X'YZin' + XY'Zin'$$

$$S = Zin (X'Y'+XY) + Zin' (X'Y+XY')$$

$$\text{Let } X'Y + XY' = X \text{ XOR } Y = A'$$

$$X'Y + XY' = A$$

$$S = Zin A' + Zin' A$$

$$= Zin \text{ XOR } A$$

$$= X \text{ XOR } Y \text{ XOR } Zin$$

$$\text{COUT} = XY + YZin + XZin$$

$$= XY (Zin + Zin') + YZin (X + X') + XZin (Y + Y')$$

$$= XYZin + XYZin' + XYZin + X'YZin + XYZin +$$

$$XY'Zin$$

$$= XYZin' + XY'Zin + XYZin' + XYZin$$

$$= XY \text{ XOR } YZin \text{ XOR } XZin$$

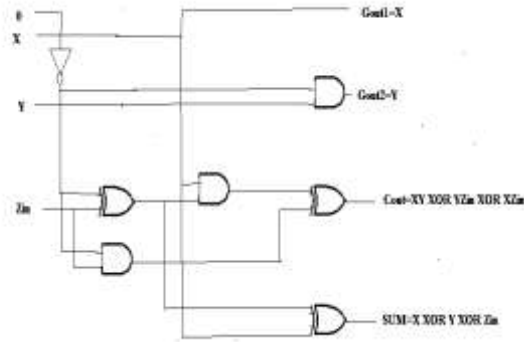


Fig. 9: DKG Traditional Logic implementation

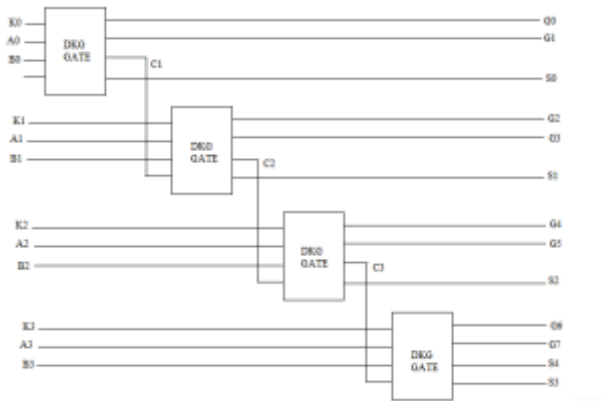


Fig. 10: 4-Bit DKG gate implemented as Full adder

V. RESULT ANALYSIS:

Here we design 4-bit Full-adder using DKG Reversible logic gate. We use 4 DKG logic gates. Here K0-K3, A0-A3, B0-B3 and Cin are as input vector and C1, C2, C3 are signals. G0-G7, S0-S4 is output vectors. Now we can show the result,

A. Device utilization summary:

- 1) Selected Device: 3s100evq100-5
- 2) Number of Slices: 4 out of 960 0%
- 3) Number of 4 input LUTs: 8 out of 1920 0%
- 4) Number of IOs: 26
- 5) Number of bonded IOBs: 22 out of 66 33%

B. RTL Design view:

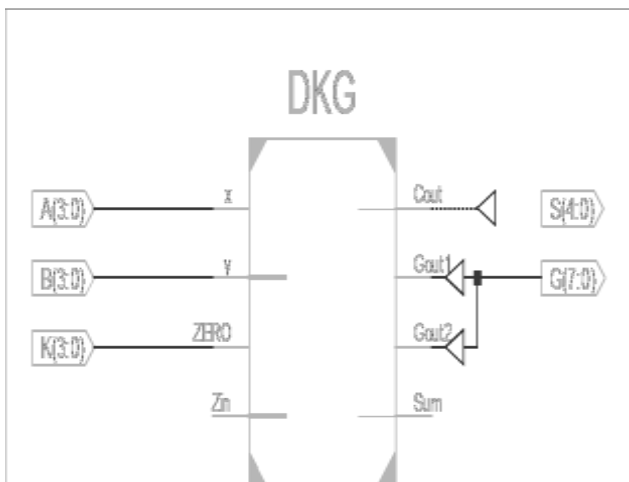


Fig. 11: RTL view

C. Result for Behavior Simulation:

Value	0 ps	10 000 ps	20 000 ps	30 000 ps
a[3:0]	0001	0101	0110	1000
b[3:0]	0100	0111	0010	1010
cin	0			
k[3:0]	0000			
g[7:0]	00100001	00100001	00111011	00011100
s[4:0]	00101	01100	01000	10010

Fig. 12: Result for 4 Bit Full adder using DKG gate

D. Result for Post Routing:

Value	0 ps	10 000 ps	20 000 ps	30 000 ps	40 000 ps
a[3:0]	1011	0001	0101	0110	1000
b[3:0]	1101	0100	0111	0010	1010
cin	0				
k[3:0]	0000				
g[7:0]	11001000	00100001	00111011	00011100	11001000
s[4:0]	10010	00101	01100	01000	10010

Fig. 13: 4-Bit Full adder using DKG gate with post routing

E. Result for power analysis:

Name	Value	Used	Total Available	Utilization (%)
Logic	0.00013 (W)	8	1920	0.4
Signals	0.00040 (W)	17	-	-
IOs	0.07130 (W)	22	66	33.3
Total Quiescent Power 0.03414 (W)				
Total Dynamic Power 0.07184 (W)				
Total Power 0.10598 (W)				
Junction Temp	30.2 (degrees C)			

Fig. 14: Result for power analysis:

VI. APPLICATION:

Reversible computing may have applications in computer security and transaction processing, but the main long-term benefit will be felt very well in those areas which require high energy efficiency, speed and performance .it include the area like

- 1) Low power CMOS
- 2) Optical computing
- 3) Quantum computer
- 4) Nanotechnology
- 5) Design of low power arithmetic and data path for digital signal processing (DSP)
- 6) Field Programmable Gate Arrays (FPGAs) in CMOS technology for extremely low power, high testability and self-repair.

ACKNOWLEDGMENT

We have presented an approach to the realize the multipurpose binary reversible gates. Such gates can be used in regular circuits realizing Boolean functions. In the same way it is possible to construct multiple-valued reversible gates having similar properties. We present a new efficient 4-Bit Reversible Full Adder using DKG gate which is

implemented in VHDL. We can simulate and synthesis it using Xilinx software and also calculate the power consumption. Here we present Result of Behavior, Post routing Simulation. And last Power analysis of 4-Bit Full adder.

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 J. Res. Develop.*, 17: 525-532.
- [3] Keyes, R.W. and R. Landauer, "Minimal energy dissipation in logic." *IBM J. Res. Develop.*, pp: 152-157, 1970.
- [4] A Review on Reversible Logic Gates and their Implementation
- [5] Abu Sadat Md. Sayem, Masashi Ueda? Optimization of reversible sequential circuits? *Journal of Computing*, Volume 2, Issue 6, June 2010, ISSN 2151-9617.
- [6] Babu HMH, Islam MR, Chowdhury AR, Chowdhury SMA. ? Synthesis of full-adder circuit using reversible logic? 17th International Conference on VLSI Design 2004, 757-60.
- [7] Ashis Kumer Biswas, Md. Mahmudul Hasan, Moshaddek Hasan, Ahsan Raja Chowdhury and Hafiz Md. Hasan Babu.? A Novel Approach to Design BCD Adder and Carry Skip BCD Adder? 21st International Conference on VLSI Design, 1063-9667/08 \$25.00 © 2008 IEEE DOI 10.1109/VLSI.2008.37.
- [8] J.Smoline and David P.DiVincenzo? Five two-qubit gates are sufficient to implement the quantum fredkin gate? *Physics Review A*, vol. 53, no.4, pp. 2855-2856,1996.
- [9] B.Raghu kanth, B.Murali Krishna, M. Sridhar, V.G. Santhi Swaroop? A DISTINGUISH BETWEEN REVERSIBLE AND CONVENTIONAL LOGIC GATES? *International Journal of Engineering Research and Applications (IJERA)* ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 2, Mar-Apr 2012, pp.148-151.
- [10] Raghava Garipelly, P.Madhu Kiran, A.Santhosh Kumar "A Review on Reversible Logic Gates and their Implementation" *International Journal of Emerging Technology and Advanced Engineering*" ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 3, March 2013.
- [11] Hafiz Md. Hasan babu, Md. Rafiqu Islam, Ahsan Raja Chowdhary and Syed Mostahead Ali chowdhary " Reversible logic synthesis for minimization of full adder ckt", *IEEE conference on Digital system design 2003,Enro-micro?03, Belek,Antalya,Tarkey,2003,PP 50-54.*
- [12] Hafiz Md. Hasan babu, Md. Rafiqu Islam, Ahsan Raja Chowdhary and Syed Mostahead Ali chowdhary "Synthesis of full adder ckt using Reversible logic".17th international conference on VLSI Design 2004, Mumbai, India 2004, PP 757-760.