

DESIGNE OF ALU USING REVERSIBLE LOGIC GATES

K. Kirankumar¹, D. Ajay Kumar²

¹Student of VLSI Design, ²Assistant Professor, E.C.E Department, Sir C.R. Reddy college of Engg, Andhra University, Andhrapradesh, India.

Abstract: Reversible logic is one of the most vital issue at present time and it has different areas for its application, those are low power CMOS, quantum computing, nanotechnology, cryptography, optical computing, DNA computing, digital signal processing (DSP), quantum dot cellular automata, communication, computer graphics. It is not possible to realize quantum computing without implementation of reversible logic. The main purposes of designing reversible logic are to decrease quantum cost, depth of the circuits and the number of garbage outputs. This paper provides the basic reversible logic gates, which in designing of more complex system having reversible circuits as a primitive component and which can execute more complicated operations using quantum computers. The reversible circuits form the basic building block of quantum computers as all quantum operations are reversible. This paper presents the data relating to the primitive reversible gates which are available in literature and helps researches in designing higher complex computing circuits using reversible gates.

Keywords: Reversible logic, Reversible gate, Power dissipation, Garbage, Quantum cost, Quantum-dot Cellular Automata, Reversible Computing.

I. INTRODUCTION

Rofl Landauer proposed in 1961 that a logical computing device with information-bearing degrees of freedom will act as a heat sink for the energy required for computation, resulting in computing errors.. Therefore, the entropy gain in such a device without a bijection between input and output states can be found using Boltzmann's entropy equation $S=KB \ln W$, where the number of states is represented by $W=2^N$, giving an energy dissipation of $kB N \ln 2$ joules per computing cycle [2]. C.H. Bennett showed that the dissipated energy directly correlated to the number of lost bits, and that computers can be logically reversible, maintain their simplicity and provide accurate calculations at practical speeds . Resultantly, a new paradigm in computer design arose with the goal of reducing the entropy increase and subsequent energy dissipation. Further, such a logical structure must possess the same number of inputs and outputs and a one-to-one mapping between the input and output states. Any device designed to these constraints is known as a reversible logic device. Reversible logic is useful in mechanical applications of nanotechnology, given that the friction generated by contacting corpuscles within a confined volume can be significantly reduced by eliminating sliding contact using mechanical reversible logic [4]. In addition, reversible logic is also applicable to fields such as quantum computing, since the laws of quantum physics are time

reversible [5], and the bijection between the input and output states enables probabilistic computations. Any arithmetic logic unit must be able to produce a variety of logic outputs based on inputs determined by the programmer for implementation in an instruction set architecture. Therefore, reversible logic devices used in an environment must have both fixed select input lines that receive opcode signals manipulated by the programmer and permanent output lines where the result of the logical output is produced. In this paper, we first describe some basics of reversible logic, present some preliminaries including some definitions and theorems related to programmable reversible logic. Second, two novel 4*4 reversible logic gates are proposed with minimal delay, and may be configured to produce a variety of logical calculations on fixed output lines based on programmable select input lines. Third, the design of a reversible ALU is presented which is implemented with the proposed gates and analyzed.

II. BASIC DEFINITIONS PERTAINING TO REVERSIBLE LOGIC

A. Reversible Function: The multiple output Boolean function $F(x_1; x_2; \dots; x_n)$ of n Boolean variables is called reversible if: a. The number of outputs is equal to the number of inputs; b. Any output pattern has a unique pre-image. In other words, reversible functions are those that perform permutations of the set of input vectors [7].

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 correspondence between the vector of inputs and outputs. It not only helps us to determine the outputs from the inputs but also helps us to uniquely recover the inputs from the outputs.

C. Ancilla inputs/ constant input: This refers to the number of inputs that are to be maintain constant at either 0 or 1 in order to synthesize the given logical function [11].

D. 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. 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. We use the words —constant inputs| to denote the present value inputs that were added to an $(n; k)$ function to make it reversible. The following simple formula shows the relation between the number of garbage outputs and constant

inputs. Input + constant input = output + garbage.

E. 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.

F. Flexibility: Flexibility refers to the universality of a reversible logic gate in realizing more functions .

G. Gate Level: This refers to the number of levels in the circuit which are required to realize the given logic functions.

H. Hardware Complexity: This refers to the total number of logic operation in a circuit. Means the total number of AND, OR and EXOR operation in a circuit 14 and 15.

I. Design Constraints for Reversible Logic Circuits:

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

- Reversible logic gates do not allow fan-outs.
- The design can be optimized so as to produce
- minimum number of garbage outputs.
- The reversible logic circuits must use minimum
- number of constant inputs.
- The reversible logic circuits must use a minimum
- logic depth or gate levels

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 optimization 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. The quantum cost of other reversible gates is calculated by counting the number of V, V+ and CNOT gates present in their circuit. V is the square root of NOT gate and V+ is its Hermitian. The V and V+ quantum gates have the following properties:

- V * V = NOT (1)
- V * V+ = V+ * V = 1 (2)
- V+ * V+ = NOT (3)

Some of the important reversible logic gates are,

1) NOT Gate The simplest Reversible gate is NOT gate and is a 1*1 gate. The Reversible 1*1 gate is NOT Gate with zero Quantum Cost is as shown in the Figure 1.

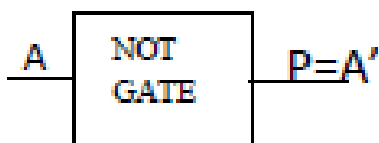


Figure 1. NOT gate

2) FEYNMAN Gate

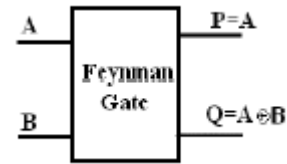


Figure 2: Feynman Gate

The Feynman gate 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.

3) FREDKIN Gate:

Fredkin gate 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.

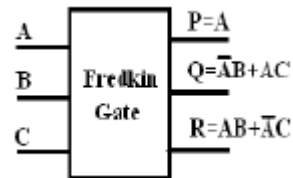


Figure.3: Fredkin Gate

4) PERES Gate:

Peres gate which is a 3*3 gate having inputs (A, B, C) and outputs P = A; Q = A XOR B; R = AB XOR C. It has Quantum cost four

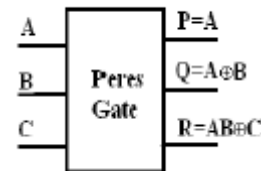


Figure 4: Peres Gate

5) DKFG PERES Gate:

Peres gate which is a 3*3 gate having inputs (A, B, C) and outputs P = A; Q = A XOR B; R = A(B XOR BC)XORBC=Carry,S=A XOR(B XOR C)=CAarry It has Quantum cost four

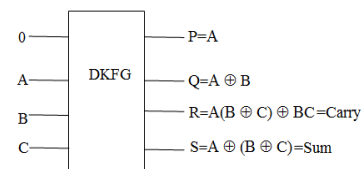


Figure 5: Peres Gate

A. ARITHMETIC UNIT:

The basic component of the arithmetic section of the ALU is a parallel adder. A parallel adder is constructed with a number of full adders and as here we have to design a 4bit ALU, we are using four full adders. By controlling the data inputs to the parallel adder, it is possible to get different type of micro operations. The input carry Cin goes to the full

adder circuit in the least significant bit position and comes out as Cout from the last adder. And the sums are obtained from the sum outputs of the full adders. The arithmetic addition is achieved when one set of input goes through the A inputs and the other set goes through the B inputs and the input carry is maintained as 0. By making Cin=1 it is possible to add 1 to the sum in F. Now if we complement all the bits of b then we can get $F=A+B'$. And by giving 1 in the Cin we can get $F=A-B$ as we know that $A-B=A+B'+1$. Similarly if we put all 0's in the input B, then we can have the transfer A operation.

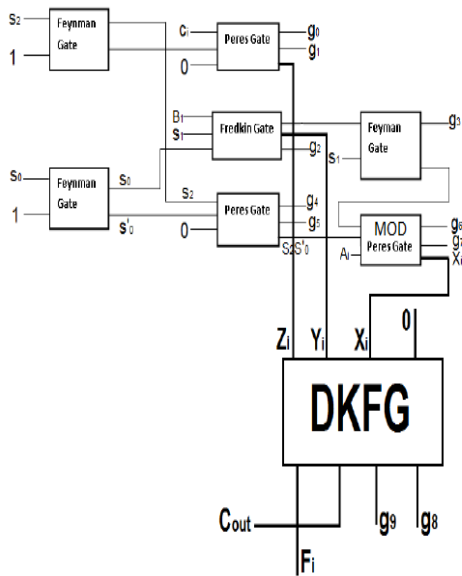


Figure 6: Arithmetic unit

IV. SIMULATION RESULTS

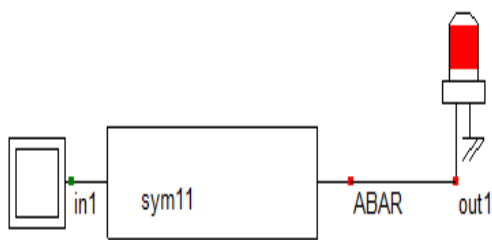


Figure 7: Simulation results in CNOT Gate

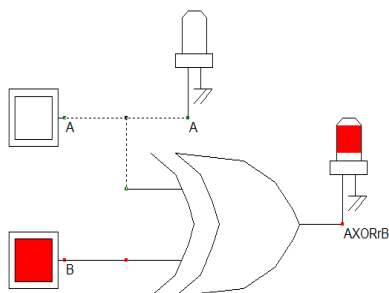


Figure 8: Simulation results in Feynman Gate in DSCH Tool

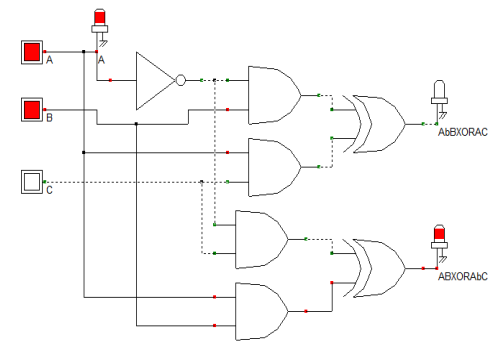


Figure 9: Simulation results in Fredkin Gate in DSCH Tool

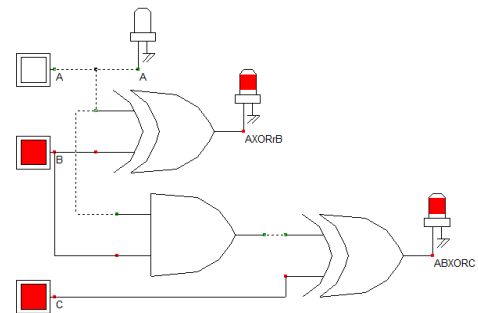


Figure 10: Simulation results in Peres Gate in DSCH Tool

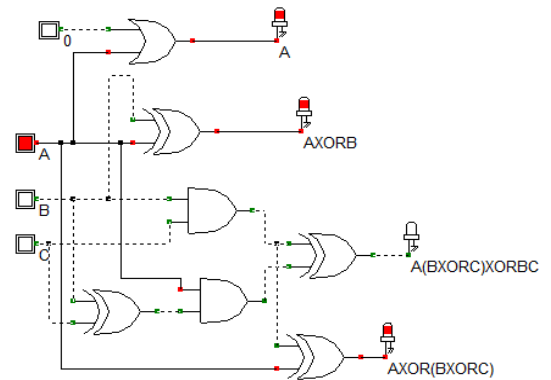


Figure 11: Simulation results in DKFG in DSCH Tool

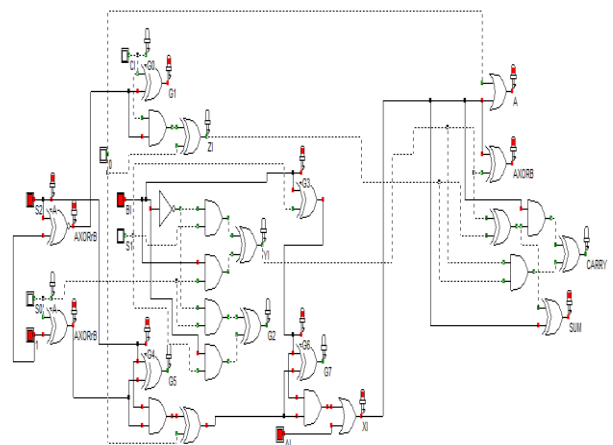


Figure 12: Simulation results in ALU in DSCH Tool

TABLE 1: Truth Table of the Arithmetic Unit

S2	S1	S0	Cin	OUTPUT equals	FUNCTIONS
0	0	0	0	F = A	A Transfer
0	0	1	0	F = A+B	Addition
0	1	0	0	F= A+B'	Subtract with Barrow
0	1	0	x	F= A+1 (or)B-1	Increment A (or) Decrement B
0	1	1	x	F= A-1	Decrement A
1	0	0	0	F= OR Gate	OR gate operation
1	0	0	1	F= NOR Gate	NOR gate operation
1	0	1	0	F= XOR Gate	XOR gate operation
1	0	1	1	F=X NOR Gate	XNOR gate operation
1	1	0	0	F= AND Gate	AND gate operation
1	1	0	1	F=NAND Gate	NAND gate operation
1	1	1	0	F= A'	NOT gate operation
1	1	1	1	F=A	A Transfer

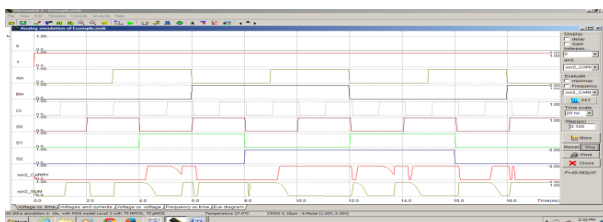


Figure 13: Simulation results in ALU in DSCHE Tool

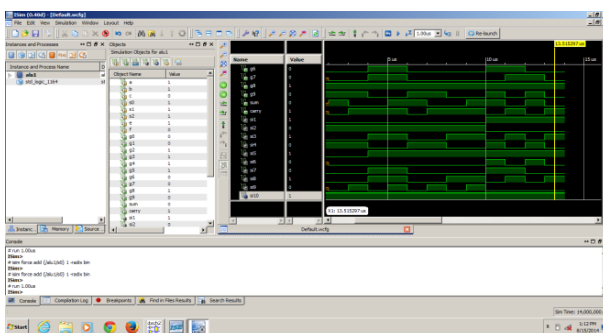


Figure 14: Simulation results in ALU in Xilinx Tool

V. CONCLUSION

The reversible circuits form the basic building block of quantum computers. This paper presents the primitive reversible gates which are gathered from literature and this paper helps researches/designers in designing higher complex computing circuits using reversible gates. The paper can further implemented the design of ALU using Reversible logic gates.

REFERENCES

- [1] R. Landauer, —Irreversibility and Heat Generation in the Computational Process, IBM Journal of Research and Development, 5, pp. 183-191, 1961.
- [2] C.H. Bennett, —Logical Reversibility of Computation, IBM J. Research and Development, pp. 525-532, November 1973.
- [3] Vlatko Vedral, Adriano Barenco and Artur Ekert, —QUANTUM Networks for Elementary Arithmetic Operations, arXiv:quantph/ 9511018 v1, nov 1995.
- [4] Perkowski, M., A. Al-Rabadi, P. Kerntopf, A. Buller, M. Chrzanowska-Jeske, A. Mishchenko, M. Azad Khan, A. Coppola, S. Yanushkevich, V. Shmerko and L. Jozwiak, —A general decomposition for reversible logic, Proc. RM'2001, Starkville, pp: 119-138, 2001..
- [5] Perkowski, M. and P. Kerntopf, —Reversible Logic. Invited tutorial, Proc. EURO-MICRO, Sept 2001, Warsaw, Poland.

Authors Profile:



D. Ajay Kumar received the Master Degree (M.E) in VLSI Design from the Bannari Amman Institute Of Technology, Sathyamagalam, Tamilnadu affiliated to ANNA University, Coimbatore. He is currently an Assistant Professor with the Department of Electronics & Communication Engineering, SIR.C.R.Reddy College of Engineering, Eluru. His research interests include the relationship of Digital System Testing and Testable Design and very-large-scale integration testing.



K. KIRANKUMAR B. Tech studied in Tenali engg. College. M. Tech in Sir C R Reddy College of Engineering, Eluru. His areas of interest in the Low Power VLSI Design.