2015 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)
19-20 December 2015, BUET, Dhaka, Bangladesh

An Efficient Approach to Design a Compact Reversible Programmable Logic Array

Nazma Tara, Lafifa Jamal and Hafiz Md. Hasan Babu
Department of Computer Science and Engineering, University of Dhaka, Bangladesh
Email: nazmatara2000@gmail.com, lafifa@yahoo.com and hafizbabu@hotmail.com

***Abstract*—Reversibility of logic module has eminent application in low power CMOS design, quantum computing, nanotechnology and optical computing. On the other hand, configurability of PLDs (Programmable Logic Devices) reduces NRE (Non-recurring engineering) cost and makes faster design process that offers customer a wide range of logic capacity, features, speed and voltage characteristics. In this paper, we propose a design methodology of RPLA (Reversible Programmable Logic Array) which reduces number of reversible gates, garbage outputs, quantum cost and constant inputs. An algorithm has been proposed for the construction of AND Plane and OR Plane of the RPLA. Comparative results show that the proposed design outperforms the existing designs in terms of numbers of gates, garbage outputs, quantum cost and constant inputs.**

***Index Terms*—Reversible logic; PLA; Decoder gate; NMG gate; AND Plane; OR Plane.**

I. Introduction

In the past few decades, reversible logic has become one of the most promising research areas. In modern technologies, power dissipation is an important issue and overheating is a serious concern for both manufacturer and consumer. Reversible computing dissipates zero energy in terms of information loss [1]. Whereas, in irreversible world, for each bit of information, $kT\ln 2$ joules of energy is dissipated, where k is Boltzmann constant and T is absolute temperature [2]. Variety of PLD (Programmable Logic Device), PLA (Programmable Logic Array) and PAL (Programmable Array Logic) was introduced by Fleisher and Maissel [3]. In [4] authors show the design of Reversible PLA (RPLA) with Fredkin and Feyman gates whereas, authors in [5] used Mux gate. Both designs are non-programmable and noncascade-able. In addition, in [6] a cascade-able and programmable design is proposed. These designs are not compact and efficient in terms of the performance comparison parameters.

Five main contributions are addressed in this paper:

1) We propose the AND plane of RPLA which is compact in terms all cost parameters of reversible logic.
2) A new gate, namely New Mux Gate, is proposed to design low cost OR plane of RPLA.
3) An Algorithm is proposed to construct the RPLA.
4) Two theorems prove the exactness of the design.
5) Comparative results demonstrate the supremacy of the design methodology.

The organization of this paper is as follows: In the next Section, basic definitions and properties of Reversible Logic and Programmable Logic Array are given. In Section III, we describe the earlier approaches of RPLA design and their limitations. In Section IV, we propose an algorithm to construct the RPLA. In Section V, we show the performance analysis of the proposed circuit. Finally, the paper is concluded with Section VI.

II. Basic Definition

Definition 1: A Reversible Circuit is a circuit in which the number of inputs and the number of outputs are equal. There is one-to-one mapping between input and output vectors, i.e. $I_V \leftrightarrow O_V$, where input vector
I_V = (I_1, I_2, ... , I_{k-1}, I_k) and output vector O_V= (O_1, O_2, ... , O_{k-1}, O_k).

Definition 2: Every gate's output, that is not used as input to other gates or as a primary output is known as Garbage.

Definition 3: Every quantum circuit is built from 1×1 and 2×2 quantum primitives and its cost is calculated as a total sum of 2×2 gates used since 1×1 gate costs nothing i.e. zero. Basically the quantum primitives are matrix operation which is applied on qubits state. All the gates of the form 2×2 has equal Quantum cost and the cost is unity i.e. 1.

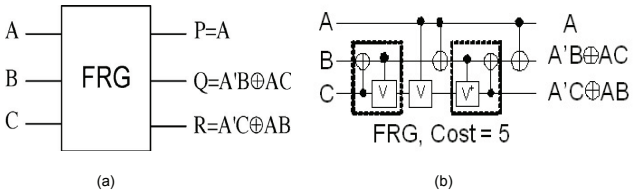

Fig. 1. Fredkin Gate : (a)Block diagram (b) Quantum circuit

The quantum cost of Fredkin gate [7] is 5, whereas, the quantum cost of HL gate [8] is 7. Block diagrams and quantum circuits of Fredkin gate and HL gate are given in Fig.1 and Fig.2, respectively.

Definition 4: Programmable Array Logic Array (PLA) consists of two planes, the first one is programmable AND plane and the second one is programmable OR plane which is all together known as AND-OR PLA [9]. Fig. 3 shows a PLA.

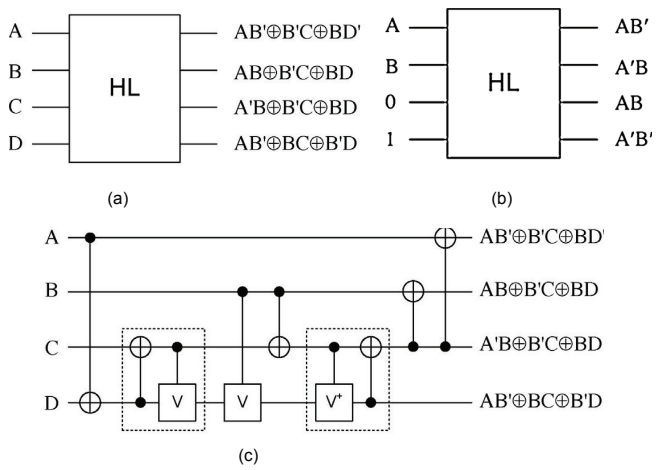

Fig. 2. a) HL Gate (b) HL Gate as Decoder (c) Quantum circuit

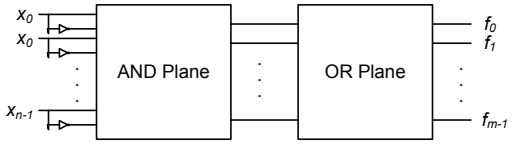

Fig. 3. Block Diagram of Programmable Logic Array

III. Design Analysis of Existing Techniques

In this section, we discuss about different types of existing RPLA [4], [5], [6]. A 3-input RPLA circuit using Feyman and Fredkin (FRG) gate is designed in [4]. Total 16 FRG gates are used to generate minterms of 3 variables, letting first two outputs of all FRG gates as don't care outputs which cause huge garbage outputs. This design is not programmable. That is, the designers cannot program its array to generate any desirable function. Authors in [5] showed a cost effective method to design the RPLA with MUX gate instead of FRG gate. It also has problem in scalability and dynamism. Authors in [6] proposed 4 different architectures of AND plane. All the existing designs require huge constant inputs and gate, produce extra garbage outputs and have high quantum cost. In this paper, we propose a compact and low cost architecture of RPLA circuit.

IV. Proposed Design of Reversible PLA

RPLA consist of two planes: AND plane and OR plane. Product terms are generated by AND plane and feed to OR plane. In OR plane, combination of some product terms are evaluated according to user's required functions. In this section, we describe the design procedure of the RPLA.

A. Proposed Reversible Gate

We propose a 3×3 reversible gate, namely New Mux Gate (NMG), which uniquely maps between input vector I_v = (I_0, I_1, I_2) and output vector O_v= (O_0, O_1, O_2), where, O_0= $A \oplus B$, O_1=$BC' \oplus AC$ and O_2=$B'C \oplus AC'$. Truth table of the NMG gate is given to prove its reversibility in Table I, whereas Fig.4(a), Fig.4(b) and Fig.4(c) show the block diagram, the implementation of the NMG gate as OR and AND functions and quantum circuit of NMG gate, respectively.

TABLE I
Truth Table of NMG Gate

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

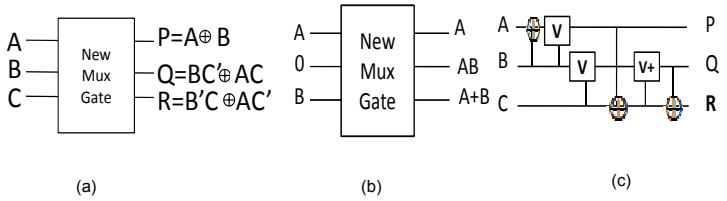

Fig. 4. NMG Gate:(a) Block diagram (b) AND-OR implementation (c) Quantum circuit

B. Design of AND Plane of RPLA

In the AND plane, minterms are generated by the logic gates. To design the general architecture for *n* variables, at first 2-variable minterms are generated. In our proposed design, we use 2-to-4 decoder HL gate proposed in [8]. As, this gate can implement four minterms of two variables and does not require any extra circuit to copy the inputs and to produce outputs, the design of AND plane is compact, cost effective and garbage free for two input variables. Fig. 2(b) shows the implementation of four minterms by HL gate. Fig. 5 shows that a 3-to-8 decoder can be constructed with four FRG gates and one HL gate which will produce eight minterms of three variables for AND plane. This design is scalable and hence reversible AND array of *n* variables can be designed with $(n-1)$-to-$2^{(n-1)}$ decoder and $2^{(n-1)}$ FRG gates.

C. Design of OR Plane of RPLA

In this section, we propose the design of programmable OR array circuit. The main inputs to this circuit are eight minterms, m_i for i = 0 to 7), and eight selector lines, s_i for i = 0 to 7. The detailed design of this programmable OR array is shown in Fig. 5. NMG gates are used to perform the AND and OR operations.

NMG gates of the first column of the design produce the copy of the minterms plus the AND terms of $m_i.s_i$'s, for i = 0 to 7. NMG gates of the second column of the design

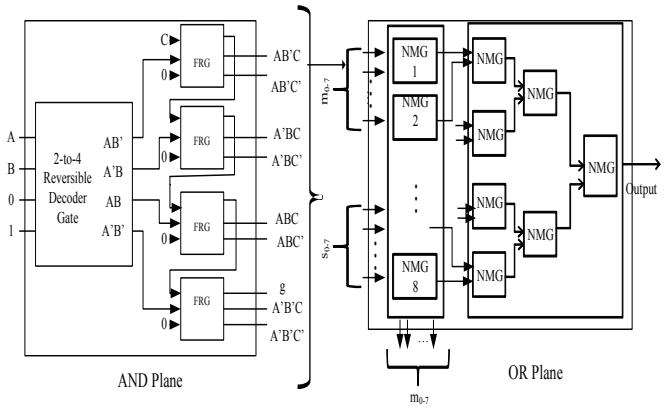

Fig. 5. Proposed AND array and OR array for 3-variable single output function

produce the OR of two terms of $m_i.s_i$ + $m_{i+1}.s_{i+1}$'s, for $i = 0, 2, 4, 6$. NMG gates of the third column produce the OR of four terms of $m_i.s_i$'s. And finally, the NMG gate of the last column produces the main output O of the OR array which is the OR of all $m_i.s_i$'s, for i = 0 to 7. The main output of this OR array, O, can be programmed through the selector lines, s_i, to produce any desired boolean function. The selector lines can be changed to set or reset to select the appropriate minterms of a function. The proposed design of the OR array also produces a copy of the minterms at its outputs. These minterms can be used in another OR array to produce another boolean function, without need to use another AND array. That means the proposed reversible and programmable OR array is also cascade-able circuit.

Algorithm 1 describes the construction of the RPLA circuit.

Algorithm 1: Algorithm to construct the reversible PLA

Input : Any Benchmark Functions
Output: Realization of reversible PLA circuit

1 **begin**
2 Construct n bit decoder with HL gate, that will act like the AND Plane of RPLA and will generate all the minterms of *n* variables.
3 Use 2^{n-1} NMG gate of which the inputs are 2^k minterms and 2^k selection bits, the outputs are copies of 2^k minterms along with selected terms those will be fed to NMG OR gates.
4 Use 2^{n-1} NMG gates to get the resulting OR plane.
5 **end**

Theorem 4.1: *Let, n be the number of input variables of AND plane of RPLA. The number of gates to realize AND plane of RPLA is at least* 2^n-3 *for* $n \geq 2$.

Proof: We prove the above statement by induction.

For $n = 2$, required number of gate is 2^2-3 =1. So, the statements holds for the base case $n = 2$.

Assume that, the statement holds for $n = k$. So, the k-input AND plane requires 2^k-3 gates.

$k + 1$-input AND plane is constructed using k-input AND plane (k-to-2^k decoder) and 2^k Fredkin gates. So, total number of gate required for $n = k + 1$ AND plane is 2^k-3 + 2^k = 2^{k+1}-3.

So, the statement holds for $n = k + 1$.

Therefore, n-input AND array requires at least 2^n-3 gates, where n$\geq$2.■

Theorem 4.2: *Let, n be the number of input variables of AND plane of RPLA. Number of garbage output to realize AND plane of RPLA is at least* $n - 2$, *where* $n \geq 2$.

Proof: We prove the above statement by induction.

The AND plane produces minterms by using decoders.

A 2-to-4 $(n = 2)$ decoder requires only one HL gate. All of the outputs of the HL gates are used to design a 2-to-4 decoder. So, no garbage output is produced by the HL gate (shown in Fig.2(b)).

So, a 2 input AND array generates at least $0(= 2-2)$ garbage output.

The statement holds for the base case $n = 2$.

Assume that, the statement holds for $n = k$. So, a k-to-2^k decoder generates at least $k - 2$ garbage outputs.

A (k + 1)-to-2^{k-1} decoder is constructed using k-to-2^k decoder and 2^k Fredkin gates. A k-to-2^k decoder generates at least $k - 2$ garbage outputs and only the last FRG gate produces one garbage output. So, total number of garbage outputs generated by a (k + 1)-to-2^{k-1} decoder is at least $k-2+1 = (k+1)-2$ So, the statement holds for $n = k+1$.

Therefore, n-input AND array produce at least $n-2$ garbage outputs, where n$\geq$2.■

D. Programming of the proposed RPLA to design 3-input multi-output function

Proposed RPLA can implement any Boolean function. For example, one of the benchmark circuits for three inputs *inc3.pla* is selected to construct the RPLA. The circuit diagram of implementing the *inc3.pla* is given in Fig. 6.

V. Performance Analysis

In this Section, we analyze the performance of our proposed design with existing designs [4], [5], [6]. Table II and Table III show the comparative results of our proposed method and existing methods presented in [4], [5], [6] in terms of number of gates, garbage outputs, quantum cost and constant input for 3-input RPLA and *n*-input RPLA, respectively. These tables indicate the compactness and cost efficiency of the proposed design. From Fig. 6, it is clear that the circuit of any benchmark function can be constructed by following the proposed methodology.

VI. Conclusion

We proposed a compact structure for Reversible Programmable Logic Array (RPLA) which is programmable and

TABLE II
COMPARISON AMONG DIFFERENT DESIGNS FOR GENERATING MINTERMS OF 3-INPUT VARIABLE

	Gate	Garbage	Quantum cost	Constant input
Existing [4]	37	32	101	37
Existing [5]	37	32	85	37
Existing [6]	10	6	38	10
Proposed	5	1	27	6

TABLE III
COMPARISON AMONG DIFFERENT DESIGNS FOR GENERATING MINTERMS OF n-INPUT VARIABLE

	Gate	Garbage	Quantum cost	Constant input
Existing [4]	$2^n(n+2)$-n	2^{n+2}	$n(2^n\text{-}1)+5.2^{n+1}$	$2^n(n+2)$-n
Existing [5]	$2^n(n+2)$-n	2^{n+2}	$n(2^n\text{-}1)+4.2^{n+1}$	$2^n(n+2)$-n
Existing [6]	2^n	n	$5(2^n\text{-}2)+2$	2^n
Proposed	2^n -3	$n-2$	$5(2^n\text{-}4)+7$	2^n -2

scalable for n-variables. The proposed decoder circuit serves as the AND Array whereas, tree structure of NMG gates constructs the OR Array. The Proposed algorithm focuses on the construction of the RPLA, which can realize any multi-output Sum of Product function. Besides, a circuit is constructed for a multi-output benchmark PLA as an example. Furthermore, we proved the efficiency and explained the characteristics of the proposed architecture with several theorems and explanations. Comparative analysis shows that the proposed design methodology requires less number of gates, produces less garbage outputs and reduces quantum cost. This design methodology requires 86.4% less number of gates than [4], [5] and 50% than [6]. The reduced average number of garbage outputs is 96.8% than [4], [5] and 83.3% than [6] respectively, whereas, the quantum cost is improved by 73.2% than [4], 68.23% than [5] and 28.94 % than [6]. At the same time, the constant input is improved by 86.5% than [4], [5] and 40% than [6]. In embedded circuits and other technologies for making low power consumption, the RPLAs are useful [10].

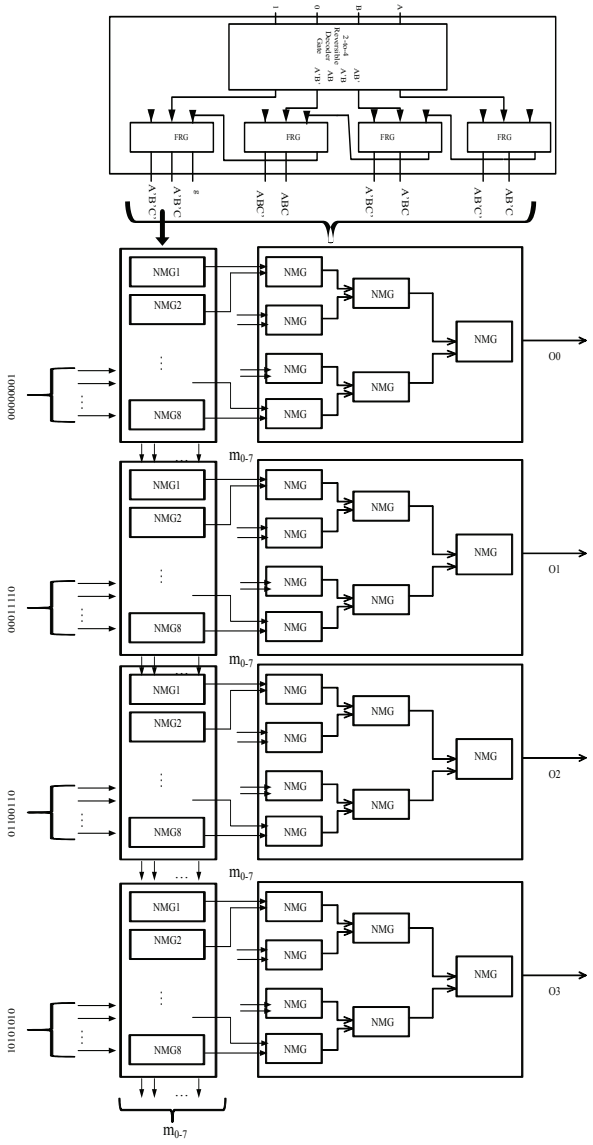

Fig. 6. Implementing inc3.pla by using the proposed reversible PLA architecture

ACKNOWLEDGMENT

We would like to thank ministry of ICT, Government of the People's Republic of Bangladesh for the ICT Innovation Fund of Research.

REFERENCES

[1] C.H. Bennett,"Logical reversibility of computation". *IBM J. Res. Dev.*, vol.17, no. 6, pp. 525-532, 1973.
[2] R. Landauer,"Irreversibility and heat generation in the computing process." *IBM J. Res. Dev.*, vol.5, no. 3, pp. 183-191, 1961.
[3] H. Fleisher and L. I. Maissel, "An itroduction to array logic", *IBM Journal of Research and Development*, vol. 19, 1975.
[4] H. Thapliyal and H. R. Arabnia, "Reversible Programmable Logic Array (RPLA) using Fredkin and Feynman gates for industrial electronics and applications", *Proc. Int. Conf. Embedded Systems and Applications (ESA'06)*, Las Vegas, USA, June 2006.
[5] Pradeep Singla and Naveen Kr. Malik, "A cost effective design of Reversible Programmable Logic Arrray", *International Journal of Computer Applications*, vol. 41,2012.
[6] Mashid Tayari and Mohammad Eshghi, "Design of 3-input reversible programmable logic array", *Journal of Circuits, Systems and Computers*, vol. 20, No.2, pp 283-297,2011.
[7] E. Fredkin, E. Toffoli, "Conservative logic", International Journal of Theoretical Physics 21 (1983) 219253.
[8] Lafifa Jamal, Md. Masbaul Alam and Hafiz Md. Hasan Babu, "An efficient approach to design a reversible control unit of a processor", *Journal of Sustainable Computing:Informatics and Systems,*), vol. 3, pp 286-294, 2013.
[9] Rubaiya Rahman, Lafifa Jamal and Hafiz Md. Hasan Babu, "Design of reversible fault tolerant programmable logic arrays with vector orientation", *International Journal of Information and Communication Technology Research*, vol. 1, no. 8, pp 337-343, 2011.
[10] G. Schrom and S. Selberherr. Ultra-low-power cmos technology. In Semiconductor Conf., Romania, 2002. E. Knill, R. Laamme, and G. J. Milburn. A scheme for eficient quantum computation with linear optics. Nature, 409:46-52, January 2001.