ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

MZI Implementation of Reversible Logic Gates, Multiplexers, Standard Functions and CLA Using Verilog HDL

Mr.Rama Krishna A
M-Tech Student,
Department of ECE,
Godavari Institute of Engineering and Technology,
NH – 16, Chaitanya Knowledge City,
Rajahmundry – 533296,india.

Mrs.K.Jyothi
HOD,
Department of ECE,
Godavari Institute of Engineering and Technology,
NH – 16, Chaitanya Knowledge City,
Rajahmundry – 533296,india.

Abstract:

With the advancements in semiconductor technology, there has been an increased emphasis in low-power design techniques over the last few decades. Now-a-days, semiconductor optical amplifier (SOA)- based Mach–Zehnder interferometer (MZI) plays a vital role in the field of ultra-fast all-optical signal processing. Reversible computing has been proposed by several researchers as a possible alternative to address the energy dissipation problem. Several implementation alternatives for reversible logic circuits have also been explored in recent years, like adiabatic logic, nuclear magnetic resonance, optical computing, etc. Recently researchers have proposed implementations of various reversible logic circuits in the all-optical computing domain. Most of these works are based on semiconductor optical amplifier (SOA) based Mach-Zehnder interferometer (MZI), which provides desirable features like low power, fast switching and ease of fabrication.

In this paper we present an all-optical implementation of a digital multiplexer using MZI switches.We exploring this project with MZI based Carry lookahead Adder(CLA). Both non-reversible and reversible versions of multiplexer design are proposed, along with analytical evaluation of the design complexities both in terms of delay and resource requirements. The final optical netlists obtained have been compared against traditional reversible synthesis approaches, by using an available synthesis tool and then mapping the reversible gates to MZI switch based implementations. Some techniques for optimizing the final optical netlists have also been proposed.

Keywords:

Reversible logic, optical computing, MZI switch, low-power design

I.INTRODUCTION:

Reversible logic is becoming a popular emerging paradigm because of its applications in various emerging technologies like quantum computing, DNA computing, optical computing, etc. [3], [11], [15], [16], [19]. It is also considered as an alternate low-power design methodology. A reversible circuit consists of a cascade of reversible gates without any fanout or feedback connections, and the number of inputs and outputs must be equal. There exists various ways by which reversible circuits can be implemented like NMR technology, optical technology, etc. [4], [7], [12], [14]. In the optical domain, a photon can store information in a signal having zero rest mass and provide very high speed. These properties of photon have motivated researchers to study and implement reversible circuits in optical domain. Theoretically from the decade old principles of Landauer [10] and Bennett [2], reversible logic is considered as a potential alternative to low-power computing. Optical implementation of reversible gates can be one possible alternative to overcome the power dissipation problem in conventional computing. In recent times researchers have investigated various reversible logic gates and their all-optical implementations using micro- resonator [18] and semiconductor optical amplifier (SOA) based Mach-Zehnder interferometer (MZI) switch [4], [8], [9], [12], [13], [14], [19]. Also MZI-based implementation of reversible logic gates offer significant advantages like ease of fabrication, high speed, low power, and fast switching time [4], [14]. In this paper, an all-optical implementation methodology of reversible multiplexers has been proposed. Cost and delay analysis in implementing arbitrary functions using such multiplexers have also been discussed. The rest of the paper is organized as follows. Section II provides a brief summary of MZI-based switch, reversible logic circuits, and implementing reversible logic gates using MZI-based switches. Section III describes the proposed all-optical multiplexer design methodology, followed by

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

discussions on implementing arbitrary functions in Section IV. Section V, followed by discussions on implementing carry lookahead adder(CLA), summarizes the results of experimental studies, followed by concluding remarks in Section VI.

II. LITERATURE SURVEY:

In this section we briefly discuss some relevant background about MZI-based optical switch, various reversible gates, and their all-optical implementations.

A.MZI-BASED ALL-OPTICAL SWITCH:

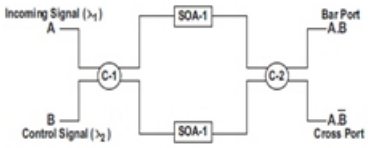
Mach-Zehnder interferometer (MZI) is one of the efficient configurable building blocks in optical computing [19]. Some of the main advantages for using MZI switches in various circuit design methodologies are their compact size, ease of fabrication, thermal stability and fast switching time [3], [14]. Recently optical switches based on MZI have attracted many researchers in the area of all-optical reversible logic implementations [3], [8], [9], [12]. An all-optical MZI switch can be constructed using two SOA and two couplers. SOA amplifies an optical signal with- out converting it to electric signal and uses a semiconductor to administer the gain medium. A coupler is a passive optical component which can either combine or split a signal based on application requirement. Figure 1(a) shows the schematic diagram of a MZI switch. It consists of two input ports and two output ports. At the input port, the optical signal entering at port A is called the incoming signal (λ1) and the optical signal coming from port B is termed as control signal (λ2). The output ports are termed as bar port and cross port. The switch works as follows.

• When both incoming signal (λ1) at A and control signal (λ2) at B are present, there will be a presence of light at bar port and no light at cross port.

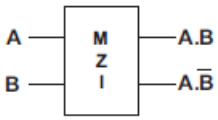
• When there is incoming signal (λ1) at A and no control signal (λ2) at B, then there is absence of light at bar port and presence of light at cross port.

If we represent presence of light as Boolean 1, and absence of light as Boolean 0, the working principle of the MZI switch can be expressed in terms of the following Boolean equations:

$$(A, B) \rightarrow (A.B, A.\overline{B})$$



(a)Semiconductor Optical Amplifier based MZI



(b) Functional behavior of MZI switch
Fig. 1. SOA based MZI Switch

B. REVERSIBLE LOGIC CIRCUITS:

A Boolean function f :Bn → Bn is said to be reversible if it is one-to-one and bijective. In other words, for every input vector, there must be a unique output vector, and vice versa. A reversible logic circuit consists of a cascade of reversible gates, with several constraints. Specifically, the number of input and output lines must be equal, and there cannot be any fanouts or feedback connections [16]. Various reversible gates have been used by researchers for synthesizing reversible gate netlists, like NOT, CNOT [5], Toffoli [20], Fredkin [6], Peres [17], etc. In a reversible gate netlist, extra inputs are often added to make a function reversible, which are called constant input or ancilla input. The outputs that are not used in the circuit but required to maintain reversibility are called garbage outputs. In the proposed reversible multiplexer design, we have used one ancilla input line.

C. IMPLEMENTING REVERSIBLE GATES USING MZI SWITCH:

Several researchers have studied and proposed all-optical implementations of reversible and non-reversible gates like Toffoli, Fredkin, Peres, XOR, NOR, etc. [3], [4], [13], [19], [22], and some function implementations like adder [9] and signed adder [1]. In all the works, the optical cost of implementation has been estimated as the number of MZI switches required, since the costs of beam splitters and beam combiners are relatively small. And the delay has been calculated as the number of stages of MZI switches multiplied by a unit Δ.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Only one work exists in the literature that discusses about all-optical realization of multiplexer [12]; however, the pro- posed implementation using Feynman and Toffoli gates is costly both in terms of optical cost and delay. In the implementation proposed in the present work, both the optical cost and delay are much smaller as compared to that in [12]. Figures 2 and 3 respectively show the all-optical realizations of a CNOT gate and a 3-input Toffoli gate. For the CNOT gate realization, the optical cost is 2, while the delay is 1Δ. Similarly, for the Toffoli gate realization, the optical cost is 3, while the delay is 2Δ. For a multiple-control Toffoligate with n inputs, the optical cost is n, and the delay is $[\log_2(n-1)+1]\Delta$.

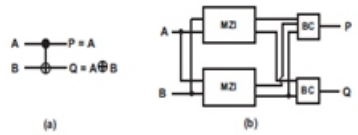


Fig. 2. All-optical CNOT gate realization

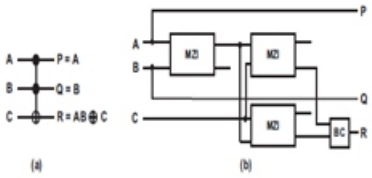


Fig. 3. All-optical Toffoli gate realization

III. PROPOSED ALL-OPTICAL MULTIPLEXER DESIGN:

In this section, we present the all-optical implementation of a digital multiplexer using MZI switches, beam splitters and beam couplers. In the following subsections, we discuss the design of a (2×1) non-reversible all-optical multiplexer, followed by its generalization to (2n×1) multiplexer. Then a design extension to make the multiplexer design reversible is suggested, that requires one additional ancilla line.

A.DESIGN OF (2×1) MULTIPLEXER :

The schematic diagram of a (2×1) multiplexer is shown in Figure 4(a), where I0 and I1 are the two inputs, and S is the select line. The function implemented at the output is also shown. The all-optical implementation of the multiplexer is shown in Figure 4(b), which consists of a beam splitter (BS) for splitting the select input S, two MZI switches which generates the subfunctions I0.S, I0.S, S.I1 and S.I 1 respectively, and finally a beam coupler (BC) that combines two of the MZI outputs to realize the desired functionality at the final output F. The BC essentially performs the logical OR function in the digital domain. In the earlier reported works on implementing logic functions using MZI switches, BS and BC [8], [9], the cost of implementation (referred to as optical cost) has been estimated as the number of MZI switches required, as the relative costs of BS and BC are small. Similarly, the delay is measured as the length of the longest cascade of MZI switches. Denoting the units of cost and delay by MZI and Δ, for the implementation as shown in Figure 4(b),

$$M(1) = 2\ MZI$$
$$D(1) = 1\ \Delta$$

where $M(x)$ and $D(x)$ respectively denote the optical costand delay for a $(2x \times 1)$ multiplexer.

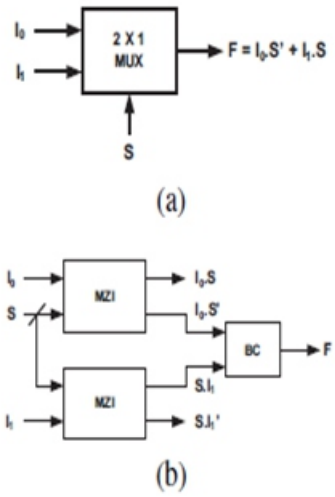


Fig. 4. A 2-to-1 MUX (a) schematic diagram, (b) all-optical implementation

B. DESIGN OF (2N ×1) MULTIPLEXER :

We now show how a multiplexer of any larger size can be constructed using smaller multiplexers as basic building blocks. This is a standard approach followed in conventional logic design; however, in the context of the present work, we shall be analyzing the costs and delays with respect to the all-optical implementations. A (4×1) multiplexer can be constructed using three (2×1) multiplexers, as shown in Figure 5(a),

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

where I0, I1, I2, I3 are the inputs, S0, S1 are the select lines, and F is the output. Each of the three multiplexers can be replaced by their corresponding all-optical netlists, to get the final netlist as shown in Figure 5(b). For this implementation,
M(2) = 6 MZI
D(2) = 2 Δ

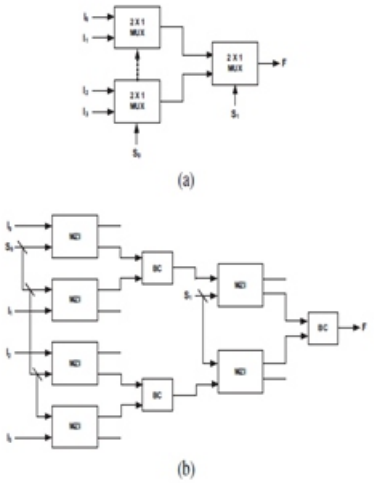


Fig. 5. A 4-to-1 MUX (a) schematic diagram, (b) all-optical implementation

Similarly, an (8 × 1) multiplexer can be built using two(4 ×1) and one (2×1) multiplexers, for which
M(3) = 6*2 + 2 = 14MZI
D(3) = 3 Δ

Generalizing, an (2n × 1) multiplexer can be built using two (2n−1 × 1) and one (2 × 1) multiplexers, as shown in Figure 6. We have seen earlier that M(1) = 2, M(2) = 6, and M(3) = 14. We can express the optical cost M(n) as a recurrence relation and solve it as follows

M(n) = 2.M(n − 1) +M(1)
= 2.M(n − 1) + 2
= 2[2.M(n − 2) + 2] + 2
= 22.M(n − 2) + 22 + 2
= 22[2.M(n − 3) + 2] + 22 + 2
= 23.M(n − 3) + 23 + 22 + 2
...
= 2n−1.M(1) + 2n−1 + 2n−2 + • • • + 2
= 2n + 2n−1 + • • • + 2
= (2n+1 − 2) MZI (1)
Since the number of MZI stages is equal to the number ofmultiplexer select lines in the final netlist, we can write thedelay for an (2n × 1) multiplexer as

D(n) = n Δ (2)

C. REVERSIBLE IMPLEMENTATION OF THE MULTIPLEXER

The all-optical implementations of multiplexer as discussed in the previous subsection are not reversible. One possibleapproach to have a reversible implementation of a multiplexer is to define a suitable reversible embedding for a (2 × 1) multiplexer, and use it to build larger multiplexers. As an alternative, we can add an extra ancilla line and have a reversible implementation, as shown in Figure 7.

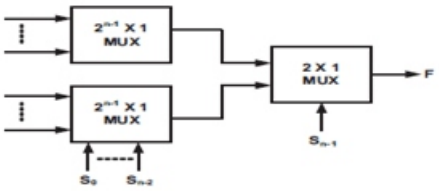


Fig. 6. A 2n-to-1 MUX

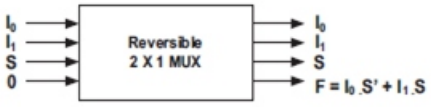


Fig. 7. Reversible multiplexer with one ancilla line

IV. IMPLEMENTING ARBITRARY FUNCTIONS USING MZI-BASED MULTIPLEXERS:

It is a well-known design practice to implement arbitrary logic functions using multiplexers. Any function of n variables can be implemented using an (2n−1 × 1) multiplexer, with an additional inverter if required. Figure 8(a) shows the truth table of an example 3-variable function, and Figure 8(b) the corresponding implementation using a (4×1) multiplexer. An all-optical implementation of the function is shown in Figure 8(c), which requires 6 MZI switches, 5 beam splitters, and 3 beam couplers.

A. Optimization rules:

It may be observed that in the all-optical multiplexer-based realization of functions, some of the MZI switches have constant inputs (0 or 1). Some of these MZI switches may be eliminated from the final netlist. These may be summarized in terms of the following four optimization rules.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

a) If the upper input of a MZI switch is 0, both the outputs will be 0's and the switch can be deleted from the netlist. This is illustrated in Figure 9(a).

b) If the lower input of a MZI switch is 0, the upper output will also be 0, and the upper input will get copied to the lower output. In this case also, the MZI switch can be deleted (see Figure 9(b)).

c) If the upper input of a MZI switch is 1, the two outputs will be respectively the lower input and its complement. A MZI switch in this configuration can be used as an inverter (Figure 9(c)).

d) If the lower input of a MZI switch is 1, the upper input gets copied to the upper output, and the lower output becomes 0. Here again the switch can be deleted (see Figure 9(d)).

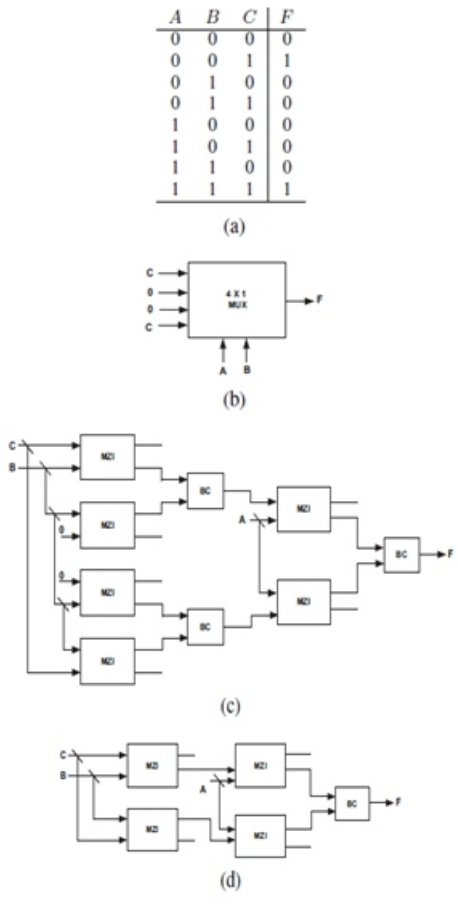


Fig. 8. Example function (a) truth table, (b) multiplexer realization (c)all-optical implementation, (d) after applying optimization rules

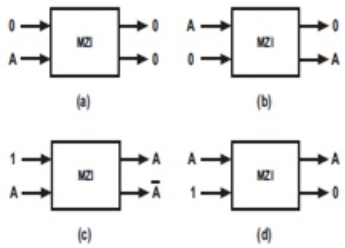


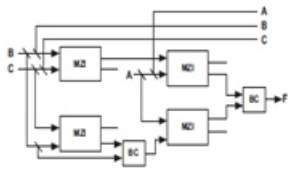
Fig. 9. The optimization rules

As an example, we apply the optimization rules to the all- optical netlist of Figure 8(c). After applying the rules, the netlist shown in Figure 8(d) results, which consists of 4 MZI switches, 3 beam splitters, and 1 beam coupler. To compare the proposed approach with conventional reversible circuit synthesis, experiments have been carried out on some standard reversible logic benchmarks from RevLib [21]. A summary of the results has been tabulated in Table I. The first three columns of the table show the benchmark, number of inputs, and number of outputs, respectively. The next two columns show the gate count (GC) and quantum cost (QC) of the reversible gate netlist available in [21]. The next two columns show the number of MZI switches (Optical Cost) and the corresponding number of stages required (Delay) when each of the Toffoli gates in the netlist is converted into its equivalent all-optical realization. An n-input multiple- control Toffoli gate can be realized using n MZI switches, with [log2(n−1) + 1]number of stages. The last two columns show the corresponding values using the proposed multiplexer based approach.

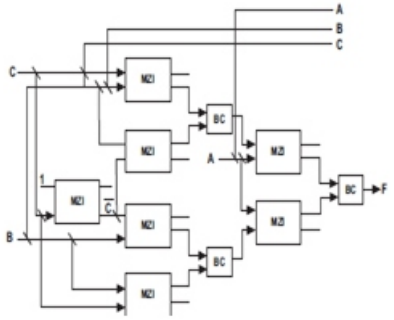
It may be noted that the optical cost and the delay as reported in Table I for the proposed approach uses a straight- forward multiplexer based realization without using the optimization rules as discussed in the previous section. If the rules are used, significant reductions in the number of MZI switches are expected. The table shows that the optical costs for some of the benchmarks are less in the proposed approach, while for some others it is more. However, the delays are significantly less for all the benchmarks in the multiplexer based approach. It is known that all 256 possible Boolean functions of three variables can be expressed in terms of 13 standard functions [23]. Table II compares the optical costs and the delays for these 13 standard functions, for the proposed method against the corresponding costs reported by Kotiyal et al. [8]. The optical cost reported for the proposed method incorporates the optimization rules discussed in the previous section.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

It can be seen that the proposed implementations require significantly less number of MZI switches and also less delay as compared to [8]. The all-optical implementations of two of the functions (F09 and F13) after optimization using the rules are shown in Figure 10.



(a)Function F09



(b) Function F13

Fig. 10. All-optical implementation of two standard functions

Table I:Synthesis Results On Benchmarks

Benchmark	PI	PO	REVLIB Netlist [21]				Proposed Method	
			GC	QC	Optical Cost	Delay	Optical Cost	Delay
4gt5-v1	4	1	4	30	12	10Δ	14	3Δ
4gt12-v0_88	4	1	5	41	15	11Δ	14	3Δ
4mod5-v0_18	4	1	8	25	19	15Δ	14	3Δ
ex2	5	1	13	144	35	25Δ	30	4Δ
alu-v2_31	5	1	13	107	41	31Δ	30	4Δ
alu-v4_36	5	1	7	32	18	14Δ	30	4Δ
majority_239	5	1	8	137	25	18Δ	30	4Δ
sym6	6	1	20	72	53	40Δ	62	5Δ

Table II:Optical Cost And Delay Analysis Comparison For 13 Standard 3-Variable Boolean Functions

Function No.	Standard Function	Kotiyal et al. [8]		Proposed Implementation	
		Optical Cost	Delay	Optical Cost	Delay
F01	$F = ABC$	6	4Δ	2	2Δ
F02	$F = AB$	3	2Δ	1	1Δ
F03	$F = ABC + AB'C'$	12	6Δ	4	3Δ
F04	$F = ABC + A'B'C'$	12	6Δ	5	3Δ
F05	$F = AB + BC$	5	3Δ	3	2Δ
F06	$F = AB + A'B'C$	10	6Δ	3	2Δ
F07	$F = ABC + A'BC' + AB'C'$	20	12Δ	6	3Δ
F08	$F = A$	2	1Δ	0	0Δ
F09	$F = AB + BC + AC$	10	6Δ	4	2Δ
F10	$F = AB + BC$	9	5Δ	4	2Δ
F11	$F = AB + BC + A'B'C'$	11	5Δ	4	2Δ
F12	$F = AB + A'B'$	7	4Δ	3	2Δ
F13	$F = ABC + A'B'C + AB'C' + A'BC'$	24	11Δ	7	3Δ

V.MZI BASED CLA:

In this section, we present all optical reversible implementation of Carry Lookahead Adder (CLA) with the property of functional reversibility using Mach-Zehnder Interferometer (MZI) switches. We have proposed two design techniques of CLA: - one is hierarchical implementation and another is nonmodular staircase structured implementation. In hierarchical implementation, a generalized design of all optical reversible CLA using SOA-based MZI switches is presented. The design of reversible CLA is improved by introducingstaircase structure. These techniques are discussed in details.

A.HIERARCHICAL DESIGN OF 2N -BIT CLA:

This design is divided in two phases. In initial phase, we design an optimized 4-bit CLA circuit and consider this 4-bit design as basic building block. Integrating several small 4-bit CLA blocks in a hierarchical way, a 2n-bit CLA circuit is constructed.

A.1.OPTIMIZED 4-BIT CLA DESIGN WITH MZI:

A reversible full adder circuit implemented with 4 MZI switches, 4 beam splitters (BS) and 3 beam combiners (BC) as shown in Fig. 11(a). Here, apart from sum (Si) and carry (Ci), we define two extra variables: - carry generator (Gi) and carry propagator (Pi), where Gi = ai.bi and Pi =aibi. The carry generate (Gi) generates output carry when the bit values of both the input namely,aiandbi are set to one, where the carry propagate (Pi) helps to propagate the carry from CitoCi+1. The output sum (Si) and carry (Ci) are expressed as Si = Pi CiandCi+1 = Gi + PiCi.

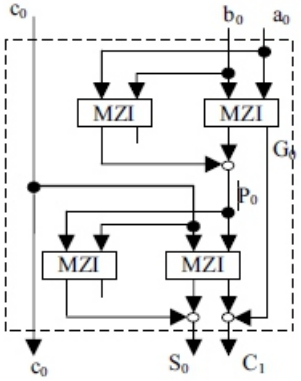


Fig. 11(a): Full adder circuit

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The optimized design of 4-bit CLA circuit using MZI switches is shown in Fig. 11(b). This circuit consists of four full adder circuits as shown by the dotted box [Fig. 3(b)]. In CLA, there are two blocks: - one is carry generate block, and another is carry propagate block.

The inputs to the 4 bit CLA are A(a3a2a1a0), B (b3b2b1b0) and c0. The c0 acts as input carry. This 4-bit CLA performs addition of two four bit binary numbers. The outputs from the CLA are sum (S) and carry (C). We deduce the following relations from Fig. 11(b).

$$S_0 = P_0 \oplus C_0, \text{where } P_0 = a_0 \oplus b_0 \text{ and } C_0 = c_0$$
$$S_1 = P_1 \oplus C_1, \text{where } P_1 = a_1 \oplus b_1 \text{ and } C_1 = G_0 + P_0 C_0$$
$$S_2 = P_2 \oplus C_2, \text{where } P_2 = a_2 \oplus b_2 \text{ and } C_2 = G_1 + P_1 C_1$$
$$S_3 = P_3 \oplus C_3, \text{where } P_3 = a_3 \oplus b_3 \text{ and } C_3 = G_2 + P_2 C_2.$$

A.2. 2N-BIT CLA DESIGN:

We integrate several 4-bit CLA in a hierarchical fashion to design a 2n-bit CLA circuit. In this hierarchical structure, we use two parameters namely, group generate and group 4-bit CLA modules and one additional 4-bit look-ahead block. The additional look-ahead block computes carry bits based on received group generate and group propagate values.

Among the five 4-bit CLA modules, four CLA blocks are identical in the sense that they compute group generate and group propagate values but the fifth one does not. For a 4-bit CLA block, the group generate and group propagate functions are labelled as G0-3 and P0-3, respectively. The group generate and group propagate functions are defined as follows.

$$G_{0\text{-}3} = G_3 + G_2P_3 + G_1P_2P_3 + G_0P_1P_2P_3$$
$$P_{0\text{-}3} = P_0P_1P_2P_3.$$

The group generate and group propagate functions are defined as follows.In this way, we can construct higher order 2n-bit CLA circuit using multiple 4-bit look-ahead blocks. For example, a 64-bit CLA is constructed by four 16-bit CLA and a 4-bit look-ahead block i.e. twenty-one 4-bit look-ahead block is required.

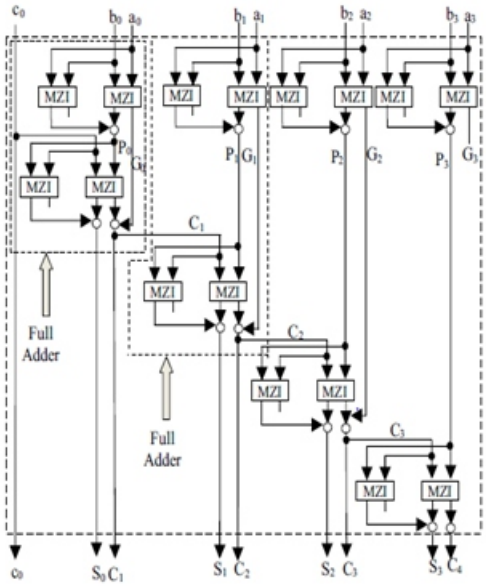


Fig. 3(b): Hierarchical design of 4-bit reversible CLA

VI.CONCLUSION:

In this paper all-optical implementation of multiplexers us- ing Mach-Zehnder Interferometer (MZI) based switches have been presented, along with analysis of the corresponding costs and delays. Using one ancilla line, a reversible implementation of multiplexer is also proposed. A method for reversible implementation of functions using MZI switches and some optimization rules have also been presented. Experimental results for some of the benchmarks reveal that the proposed all-optical implementation results in significantly less delay as compared to the one based on conventional reversible gate implementations. Comparison with a recent work for the 13 standard 3-variable functions has also been reported, which demonstrates significant improvements both in terms of optical cost and delay.

REFERENCES:

[1] A. Al-Zayed and A. Cherri. Improved all-optical modified signed- digit adders using semiconductor optical amplifier and Mach-Zehnder interferometer. Optics and Laser Technology, 42(5):810–818, 2010.

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

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[3] T. Chattopadhyay. All-optical symmetric ternary logic gate. Optics and Laser Technology, 42(6):1014–1021, 2010.

[4] T. Chattopadhyay. All-optical modified Fredkin gate. IEEE Journal of Selected Topics in Quantum Electronics, 18(2):585–592, 2012.

[5] R. Feynman. Quantum mechanical computers. Optic News, 11:11–20, 1985.

[6] E. Fredkin and T. Toffoli.Conservativelogic.Inernational Journal of Theoretical Physics, 21:219–253, 1982.

[7] J. Kim, J.-S. Lee, S. Lee, and C. Cheong.Implementation of the refined Deutsch-Jozsa algorithm on a three-bit NMR quantum computer. Physical Review A (Atomic, Molecular, and Optical Physics), 62(5):27– 37, 2000.

[8] S. Kotiyal, H. Thapliyal, and N. Ranganathan. Mach-Zehnderinterfer- ometer based all optical reversible NOR gate. In Proc. of IEEE Computer Society Annual Symposium on VLSI, pages 207–212, 2012.

[9] S. Kotiyal, H. Thapliyal, and N. Ranganathan. Mach-Zehnderinterfer- ometer based design of all optical reversible binary adder. In Proc. of Design, Automation and Test in Europe (DATE), pages 721–726, 2012.

[10] R. Landauer. Irreversibility and heat generation in computing process. Journal of IBM Research and Development, 5:183–191, 1961.

[11] X. Ma, J. Huang, C. Metra, and F. Lombardi.Detecting multiple faults in one-dimentional arrays of reversible qca gates. Journal of Electronic Testing, 25(1):39–54, 2009.

[12] G. Maity, T. Chattopadhyay, J. Roy, and S. Maity. All-optical reversible multiplexer. In Proc. of Computers and Devices for Communication (CODEC), pages 1–3, 2009.

[13] G. Maity, S. Maity, T. Chattopadhyay, and J. Roy. Mach-Zehnder interferometer based all-optical Fredkin gate. In Proc. of Intl. Conf. on Trends in Optics and Photonics, pages 138–145, 2009.

[14] G. Maity, J. Roy, and S. Maity. Mach-Zehnder interferometer based all-optical Peres gate. In Proc. of Intl. Conf. on Advances in Computing and Communications, pages 249–258, 2011.

[15] R. Merkle. Reversible electronic logic using switches. Journal of Nanotechnology, 4:21–40, 1993.

[16] M. Nielsen and I. Chuang.Quantum Computation and Quantum Information.Cambridge University Press, 2000.

[17] A. Peres. Reversible logic and quantum computers. Physical Review A (Atomic, Molecular, and Optical Physics), 32(6):3266–3276, 1985.