Optik 125 (2014) 1030–1033

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.de/ijleo

All-optical method of developing some fundamental and functional quaternary logic gates

Sisir Kumar Garai

Department of Physics, M.U.C. Women's College, Burdwan, West Bengal 713104, India

ARTICLE INFO

Article history:
Received 20 March 2013
Accepted 21 July 2013

Keywords:
Quaternary logic gates
Nonlinear polarization rotation
Frequency conversion
Semiconductor optical amplifier
Add and drop multiplexer

ABSTRACT

All optical multivalued logic processors are of paramount importance in optical computing and signal processing. In this article the author proposes a method of developing all-optical quaternary Inversion, NOT and Bitswap logic gates, which are essential parts of quaternary arithmetic and logical processors. Nonlinear switching and add/drop multiplexing (ADM) properties of semiconductor optical amplifier (SOA) are exploited here to develop the frequency encoded quaternary logic gates.

1. Introduction

Boolean logic based optical processors suffer from speed and information handling capacity limitations due to its data representation system, only by two distinct states '0' and '1' respectively. On the other hand, multivalued logic system, such as tristate, quaternary systems etc., has more than two distinct bits and obviously expected to have larger information handling capacities. Moreover, using multivalued logic carry- and borrow-free different arithmetic operations can be performed more quickly and therefore expected to fulfill the demand of faster speed of processing. So many research efforts on multivalued logic processors have been focused using different data encoding techniques such as intensity encoding [1,2], polarization encoding [3,4], spatial encoding [5], symbolic substitution [6] etc. Most of these suffer from bit-error problem. In this scenario optical processors working on frequency encoded data believe to perform impressive role for error free secure data processing [7,8].

'Quaternary algebra' deals with the numerous operations among the quaternary digits i.e., qudit {0, 1, 2, 3} [4,6,9,10]. Qudits 0, 1, 2 and 3 can be considered as 2-bit binary equivalent 00(absolute low), 01(intermediate low), 10(intermediate high), and 11(absolute high) respectively. A quaternary bit '*A*' can be expressed by two binary bits '0' and '1', packed together using the following notion '$A = \langle a_1, a_0 \rangle$' where a_1 and a_0 are the constituent binary bits such that binary equivalent value of '$2a_1 + a_0$' represents the corresponding qudit '*A*'. Numerous operations of quaternary algebra are classified into fundamental and functional operations [9]. AND, OR, NOT, Bitswap operations etc., are belonging to fundamental operations. On the other hand, functional operations are the combination of two or more fundamental operations. Inward inverter, outward inverter, equality checker, MIN, MAX, XOR etc., are belonging to functional operations.

NOT operation of the quaternary bit '*A*' is denoted here by '$\bar{A}$' and it is obtained by taking the complement of binary bits 'a_1' and 'a_0' respectively, i.e., $\bar{A} = \langle \overline{a_1}, \overline{a_0} \rangle$. Therefore, NOT operation of the quaternary bit $A = 2 = \langle 1{,}0 \rangle$ will give $\bar{A} = \langle \bar{1}, \bar{0} \rangle = \langle 0, 1 \rangle = 1$, and so on.

Inward inversion operation of the quaternary bit '*A*' is denoted here by 'A^{I}' and it is obtained replacing 'a_1' by its complement and 'a_0' by 'a_1'. Therefore, $A^{I} = \langle \overline{a_1}, a_1 \rangle$. For example, inward inversion operation of quaternary bit $A = 3$ is $A^{I} = \langle \bar{1}, 1 \rangle = \langle 0, 1 \rangle = 1$.

Outward inversion operation of the quaternary bit '*A*' is denoted here by '!*A*' and its binary equivalent value is obtained by taking the complement of its equivalent binary bit 'a_1' in place of both the 'a_1' and 'a_0' leaving 'a_0' bit i.e., $!A = \langle \overline{a_1}, \overline{a_1} \rangle$. Therefore, outward inversion operation of a quaternary bit $A = 3$ will give '$!A$' $= \langle \bar{1}, \bar{1} \rangle = \langle 0, 0 \rangle = 0$, and so on.

Bitswap operation of a quaternary bit '*A*' is denoted here by '$\sim A$' and it is obtained by interchanging the position of 'a_0' and 'a_1' i.e., '$\sim A$' $= \langle a_1, a_0 \rangle$. Therefore, bitswap operation of quaternary bit $A = 2$ will be '$\sim A = \langle 0, 1 \rangle = 1$'.

If '*A*' and 'B' represent two quaternary bits such that '*A*' $= \langle a_1, a_0 \rangle$ and 'B' $= \langle b_1, b_0 \rangle$, and if F(*A*,B) represents the quaternary

E-mail address: skgarai@gmail.com

0030-4026/$ – see front matter
http://dx.doi.org/10.1016/j.ijleo.2013.07.103

Table 1
Truth table of frequency different quaternary logic gates.

Input quaternary bit A (equivalent binary value)	Output			
	Inward inversion $A^{/}$	Outward inversion !A	NOT $\overline{A}$	Bit swap $\sim A$
$\nu_0(0)$	$\nu_2(2)$	$\nu_3(3)$	$\nu_3(3)$	$\nu_0(0)$
$\nu_1(1)$	$\nu_2(2)$	$\nu_3(3)$	$\nu_2(2)$	$\nu_2(2)$
$\nu_2(2)$	$\nu_1(1)$	$\nu_0(0)$	$\nu_1(1)$	$\nu_1(1)$
$\nu_3(3)$	$\nu_1(1)$	$\nu_0(0)$	$\nu_0(0)$	$\nu_3(3)$

operation between '*A*' and 'B', and, '*f*(a,*b*)' represents the similar binary operation among the equivalent binary bits, then $F(A,B)=(\langle a_1,a_0\rangle, \langle b_1,b_0\rangle)=(\langle f(a_1,b_1), f(a_0,b_0)\rangle)$.

In this article the author proposes a method of developing two fundamental quaternary logic gates NOT and Bitswap, and, two functional logic gates Inward and Outward inversion respectively using the frequency encoded data. Here quaternary bit '0', '1','2' and '3' are encoded by the optical beams of frequency 'ν_0', 'ν_1', 'ν_2' and 'ν_3' respectively. The truth tables for frequency encoded quaternary NOT, inversion and Bitswap gates are shown in Table 1.

2. Working principles of the proposed scheme

In this proposed scheme, quaternary logic gates work on the principles of frequency conversion [11–14] and frequency routing [15,16] of the optical beams exploiting nonlinear properties of SOA which are briefly mentioned below.

Here SOA based frequency conversion is due to rotation of the state of polarization (SOP) of the probe beam in SOA. The frequency conversion scheme is explained with the help of Fig. 1(a). Here '*P*' is the linearly polarized probe beam and its state of polarization is maintained by polarization controllers (PC). 'AN' is the optical analyzer and its pass axis is kept crossed with respect to electric field of the incoming linearly polarized probe beam '*P*'. When the intense pump beam '*A*' is applied in SOA, the pump beam introduces nonlinear refractive index difference between the 'TE' and 'TM' (transverse electric and magnetic) modes of the probe in the bulk material of SOA, which in turn rotates the state of polarization (SOP) of the probe beam. Consequently, a component of the electric field of the amplified probe beam '*P*' passes through the pass axis of the 'AN'. In the absence of pump beam, the pass axis of the 'AN' being crossed with respect to the direction of vibration of the probe beam, no light passes through the 'AN'. SOA based frequency converter is symbolically presented in Fig. 1(b).

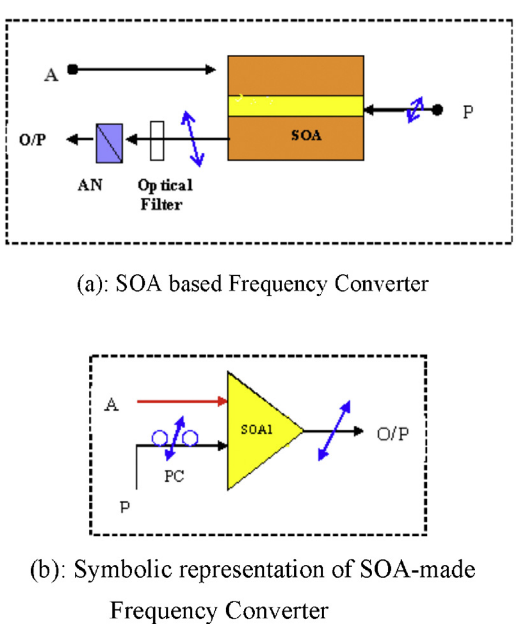

Fig. 1. (a) SOA based frequency converter. (b) Symbolic representation of SOA-made Frequency Converter.

The optical add/drop multiplexers (ADM) are used here to route the optical beams of different encoded frequencies to different channels (optical fibers). This can be accompanied by using SOA with tunable filter in it. The filter can be tuned to reflect the optical beam of a specific frequency by properly selecting the biasing current of SOA. The optical beam of that selected frequency is reflected back by the filter section of the SOA, dropped down from common channel by the optical circulator (Cr) and finally it is guided to the desired path using optical fiber.

3. Optical circuit and operations of quaternary functional logic gates

All-optical circuit for implementing the frequency encoded quaternary functional inversion logic gate is as shown in Fig. 2. Here '*A*' is the frequency encoded input quaternary data beam (signal). ADM0, ADM1 are optical add/drop multiplexers tuned for the reflected beams of frequency ν_0 and ν_1 respectively whereas, both the ADM2 and ADM2$^{/}$are tuned for the reflected beams of frequency 'ν_2'. 'SOA1' and 'SOA2' are the frequency converters having the probe beams 'P_2' of frequency 'ν_2' and 'P_1' of frequency 'ν_1' respectively. 'SOA3' and 'SOA4' are the frequency converters having the probe beams 'P_3' and 'P_0' of frequency 'ν_3' and 'ν_0' respectively. Input beam '*A*' of frequency 'ν_0/ν_1' used as the pump beam of 'SOA1'after reflecting back by ADM0/ADM1 and dropping down by its circulator 'Cr' and the input beam '*A*' frequency 'ν_2/ν_3' is used as the pump beam of 'SOA2'. Output beams of 'SOA1' and 'SOA2' are combined together and then it is split up into two equal parts by the beam splitter (B.S.). One part of the beam is used as the output of the 'Inward inversion' logic gate (Y) and the remaining part the beam of frequency 'ν_2' is used as the pump beam of 'SOA3' and the optical beams other than the beam of frequency 'ν_2' are used as the pump beam of 'SOA4'. Output beams of the 'SOA3' and 'SOA4' are combined together to get the output (X) of the 'outward inversion' gate.

Now the operation of the frequency encoded 'quaternary inverter' is explained with the help the Fig. 2.

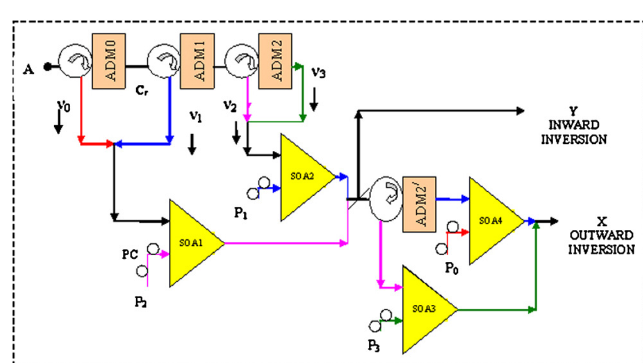

Fig. 2. Frequency encoded quaternary inversion logic gates.