Design of Some Useful Logic Blocks Using Quaternary Algebra

Mrs.Yashika A.Gaidhani
Lecturer , Dept of Electronics Engg.
Y.C.C.E
Nagpur,India
Shweta29gaidhani@gmail.com

Mrs.Monica N. Kalbande
Lecturer , Dept of Electronics Engg.
Y.C.C.E
Nagpur,India
monicakalbande@gmail.com

Abstract—**The quaternary logic scheme used here is obtained by extending Boolean algebra into quaternary domain. The logic is capable of handling both quaternary and coupled-binary inputs, where binary operands are coupled in pairs to form quaternary entities. A set of operators capable of handling both coupled-binary and ordinary inputs are used. To demonstrate the functionality of this novel logic scheme, some useful logic blocks such as decoder, multiplexer, demultiplexer & encoder are derived mathematically and demonstrated graphically using simulated timing.**

Keywords—quaternary logic; quaternary algebra; priority encoder; decoder; multiplexer; demultiplexer;

I. INTRODUCTION

Multi-valued logic is an extension of binary logic where any *proposition* can have more than 2 values. *Boolean algebra* in binary logic system is established by defining some fundamental operators which are fully analogous to union, intersection and compliment operations in set theory. Various researchers have been working on multi-valued logic for a long time, but the lack of physically implement able and efficient circuit models has limited the development of this field severely. In this paper we have briefly revisited a new scheme for *quaternary logic* system which is closely related to binary system and was first proposed in [1]-[2]. Multi-valued logic circuits are more complex in nature than binary logic circuits; it is often difficult to design a large system from the scratch. Besides, these schemes are not suitable for coupled-binary inputs. That is why a new logic scheme is required to overcome these problems. Although multi-valued logic scheme can be a solution for the demand of increasing data storage capability and faster computing [3],[4], limitations imposed by VLSI technology restrain the usage of too many levels in logic systems. Thus quaternary logic seems to be a perfect choice in this regard. In this paper, we have defined the basic operators of our proposed quaternary logic system. The system has evolved from and closely related to binary logic system. We have derived several theorems and properties to facilitate the development of a fully functional algebra. We proposed a novel quaternary logic close to Boolean algebra. It extends binary functions in quaternary, at the same time incorporates some new functions of its own. Extending Boolean algebra makes this logic system ideal for replacing binary logic as its logic blocks are compatible with their binary counterparts.

In this paper we have shown the design of some important logic blocks such as multiplexer, demultiplexer, encoder and decoder.

II. QUATERNARY ALGEBRA

Quaternary states (0, 1, 2, 3) can be imagined as 2-bit binary equivalents 00, 01, 10, 11. If the bits of the binary equivalent interchange their positions and still the quaternary state remains unchanged, then it is said to have *binary symmetry*; otherwise it is *asymmetrical*. Thus 0, 3 are symmetrical and 1, 2 are asymmetrical. When expressed as a number, a single quaternary digit is called a qudit.

The algebra we have developed so far in [1] has two types of operators – basic operators and special operators. The basic quaternary operators (Fig. 1) are very similar to binary operators and they are obtained from Boolean algebra. They operate as *bitwise binary operators* working on 2-bit operands, as shown in Table I. The basic operators are *OR*, *AND*, *BASIC INVERTER* and *XOR*. Their derivatives are *BASIC NOR*, *BASIC NAND* and *BASIC XNOR*. The inverter is named as basic inverter since there are other special inverters or *compound inverters* in quaternary system to facilitate complex logic design.

The special operators are all unary operators. They are -
(a) *Outward Inverter* or *Full Inverter*
(b) *Inward Inverter* or *Half Inverter*
(c) *Binary Bitswap*

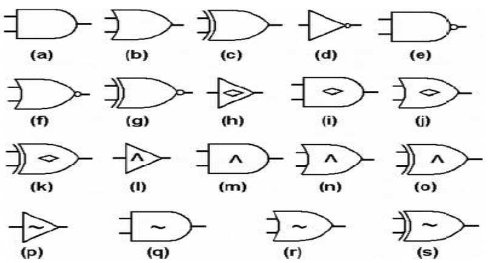


Figure 1. . Circuit symbols for the quaternary operators. (a) and; (b) or; (c) xor; (d) basic NOT; (e) basic nand; (t) basic nor, (g) basic xnor; (h) Inward Inverter; (i) Inward nand; (j) Inward nor; (k) Inward xnor; (1)Outward Inverter; (m)Outward nand; (n) Outward nor; (0) Outward xnor; (p) Binary Bitswap; (q) Bitswap and; (r) Bitswap or; (s) Bitswap xor.

UACEE
Globalize The Research
Localize The World

The outward inverter (Fig. 1(l)) changes 0, 1 to 3 and 2, 3 to O. The inward inverter (Fig. l(h)) changes 0, 1 to 2 and 2,3 to 1. The binary bitswap (Fig. 1(p)) swaps the two bits of the binary-equivalent of the quaternary operand. It leaves the symmetrical numbers unchanged but inverts (basic inversion) the asymmetrical numbers. That is why it is classified as a special inverter-like operator. The mathematical definitions of the special operators are given below:

$$\text{Inward Inverter,}\quad a' = \begin{cases} \overline{a}.2 & ; a < 2 \\ \overline{a}+1 & ; a > 1 \end{cases}$$

$$\text{Outward Inverter,}\quad \hat{a} = \begin{cases} \overline{a}+3 & ; a < 2 \\ \overline{a}.0 & ; a > 1 \end{cases}$$

$$\text{Binary Bitswap,}\quad \tilde{a} = \begin{cases} \overline{a} & ; a \text{ asymmetric} \\ a & ; a \text{ symmetric} \end{cases}$$

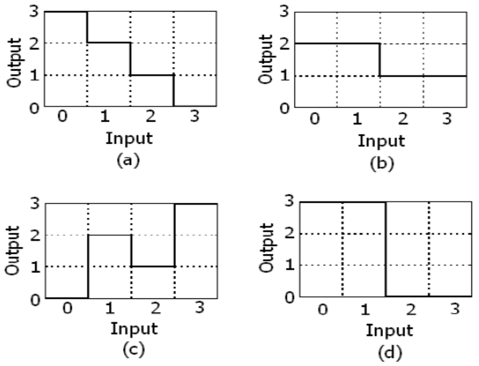


Figure 2. Transfer functions of the quaternary inverters and inverter-like operators: (a) Basic Inverter; (b) Inward or Half Inverter; (c) Binary Bitswap;(d) Outward or Full Inverter.

Table I Quaternary Single-input Operators

A	BASIC NOT $\overline{A}$	INWARD NOT A'	OUTWARD NOT $\hat{A}$	BINARY BITSWAP $\tilde{A}$
0	3	2	3	0
1	2	2	3	2
2	1	1	0	1
3	0	1	0	3

Table II Basic Quaternary Multi-input Operators

A	B	AND	OR	XOR	BASIC NAND	BASIC NOR	BASIC XNOR
0	0	0	0	0	3	3	3
0	1	0	1	1	3	2	2
0	2	0	2	2	3	1	1
0	3	0	3	3	3	0	0
1	1	1	1	0	2	2	3
1	2	0	3	3	3	0	0
1	3	1	3	2	2	0	1
2	2	2	2	0	1	1	3
2	3	2	3	1	1	0	2
3	3	3	3	0	0	0	3

Like binary Boolean algebra, the proposed quaternary logic scheme has its own algebra. Since it has evolved from Boolean algebra, the properties and axioms of Boolean algebra are applicable for basic quaternary operators. These include commutativity, associativity, distributivity, etc.

Apart from the basic properties, the special operators have many other important properties of their own [1]. These properties are indispensable in designing quaternary logic circuits. Some of these properties are as follows:

$$\overline{\overline{a}} = a,\quad \tilde{\tilde{a}} = a,\quad a'' \neq a,\quad \hat{\hat{a}} \neq a$$

$$\widetilde{a'} = \begin{cases} a+1 \,;\, a < 2 \\ a.2 \,;\, a > 1 \end{cases}$$

$$\tilde{a} + a = \begin{cases} 3 \,;\, a \neq 0 \\ 0 \,;\, a = 0 \end{cases}$$

$$\tilde{a}.a = \begin{cases} 3 \,;\, a = 3 \\ 0 \,;\, a \neq 3 \end{cases}$$

$$\tilde{a} \oplus a = \begin{cases} 3 \,;\, a \text{ asymmetric} \\ 0 \,;\, a \text{ symmetric} \end{cases}$$

$$a'.\hat{a} = \begin{cases} 0 \,;\; a > 1 \\ 2 \,;\; a < 2 \end{cases}$$

$$a' + \hat{a} = \begin{cases} 1 \,;\; a > 1 \\ 3 \,;\; a < 2 \end{cases}$$

$$(a \,\widehat{+}\, b) = \hat{a}.\hat{b} \text{ and } (\widehat{a.b}) = \hat{a} + \hat{b}$$

$$(a \,\widetilde{+}\, b) = \tilde{a} + \tilde{b} \text{ and } (\widetilde{a.b}) = \tilde{a}.\tilde{b}$$

$$\left.\begin{array}{l} (a+b)' \neq a'.b',\ (a'.b') \neq a'+b' \\ (a+b)' \neq a'+b',\ (a'.b') \neq a'.b' \end{array}\right\}$$

$$\left.\begin{array}{l} \hat{\overline{a}} = \overline{\hat{a}},\ (\widehat{a'}) \neq (\hat{a})',\ (\overline{a})' = (\overline{a'}) \\ \overline{\tilde{a}} = \tilde{\overline{a}},\ \tilde{\hat{a}} \neq \hat{\tilde{a}},\ (\widetilde{a'}) \neq (\tilde{a})' \end{array}\right\}$$

$$\tilde{a} \oplus \tilde{b} = (a \,\widetilde{\oplus}\, b)$$

III. THE EQUALITY OPERATOR

The equality operator compares two values and returns 3 if they are identical and 0 if they are different. The circuit can be built in several ways.

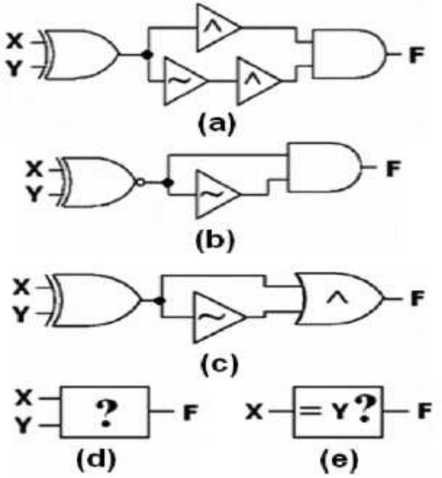


Figure 3. (a) - (c) Quaternary equality operator, (d) circuit symbol for equality operator, (e) compact circuit symbol for equality operator

IV. ELEMENTARY QUATERNARY ALGEBRA

Combinational circuit is the vital element for any digital system. Our proposed quaternary operators can be integrated efficiently for designing some conventional circuits. Many researches have been done for quantum realization of these blocks [6]. We have found a useful way to define these blocks making them analogous to their binary counterparts.

A. Quaternary Decoder

A quaternary 1-to-4 decoder has one input and four outputs. If the outputs are active-high (absolute high, 3) and the input selects any one of the four lines at a time, then the outputs are defined by

$L_i = s^i$; $i = 0, 1, 2, 3$

Here, L_i is output line number i, s is input.

In general, for n-to-4^n decoder

$$L_{i_1 i_2 .....i_n} = s_1^{i_1} . s_2^{i_2} .....s_n^{i_n} \; ; \; i_1,....,i_n = 0, 1, 2, 3$$

Like binary decoders, hierarchical design is also possible in quaternary decoders. The expression for the decoder reveals that

1-to-4 decoder can be used as a building block for constructing larger decoders.

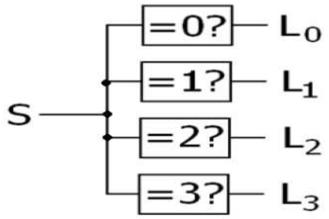


Figure 4. A quaternary 1-to-4 decoder using equality operator. The equality block can be designed as shown in Fig. 5(a)

B. Quaternary Demultiplexer

A 1-to-4 demultiplexer is same as 1-to-4 decoder, but it passes a particular data through a particular line selected by the selector input. So, if the input data is *d*, then using equation for demultiplexer with n selectors is

$$L_{i_1 i_2 .....i_n} = d \,. s_1^{i_1} ..... s_n^{i_n} \; ; \; i_1,....,i_n = 0, 1, 2, 3$$

For n = 1, $L_i = d \,. s^i$; $i = 0, 1, 2, 3$

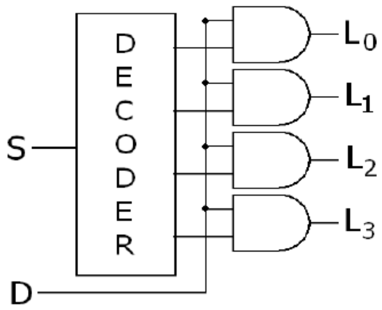
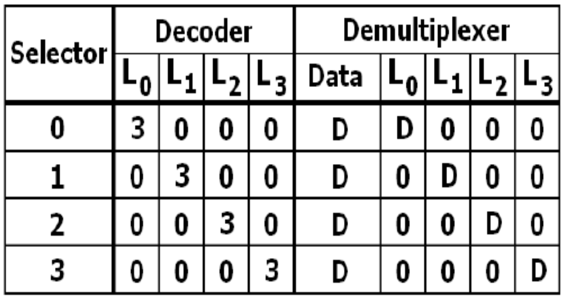


Figure 5. A quaternary 1-to-4 demultiplexer using a 1-to-4 decoder. If the equality operator as shown in Fig. 4(a) is used here, then the AND gates can be used to couple the data D with the equality blocks directly. The external AND gates shown here will be unnecessary in that case.



Selector	Decoder				Demultiplexer				
	L_0	L_1	L_2	L_3	Data	L_0	L_1	L_2	L_3
0	3	0	0	0	D	D	0	0	0
1	0	3	0	0	D	0	D	0	0
2	0	0	3	0	D	0	0	D	0
3	0	0	0	3	D	0	0	0	D

Figure 6. Truth table for quaternary 1-to-4 demultiplexer and decoder. From this table, it is clear that both devices are very similar. The data input D makes the only difference between them

C. Quaternary Multiplexer

A multiplexer has n selectors and 4n inputs; the selectors select a single line and pass its value to the output. So, it is possible to use a decoder to build a multiplexer by combining each output terminal with a data terminal through an AND gate and merging all results using an OR gate. Hence the multiplexer has an expression of SOP form.

$L_{i_1 i_2 \ldots\ldots i_n} = s_1^{i_1} . s_2^{i_2} \ldots\ldots s_n^{i_n}$; $i_1, i_2, \ldots\ldots, i_n = 0, 1, 2, 3$

The output of the multiplexer:

$M_n = \sum L_{i_1 \ldots\ldots i_n} . d_{i_1 \ldots\ldots i_n}$ (

Where $i_1, \ldots\ldots, i_n = 0, 1, 2, 3$

Here, $\sum$ means OR operation.

For n = 1, $M_1 = d_0 . s^0 + d_1 . s^1 + d_2 . s^2 + d_3 . s^3$

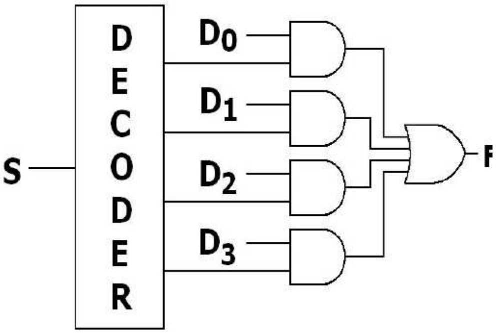


Figure 7. A quaternary 4-to-1 multiplexer using a 1-to-4 decoder. Like the case of demultiplexer, the equality circuit as shown in Fig. 4(a) can be used to couple the data lines without using the external AND gates shown here.

D. Quaternary Encoder

A quaternary 4-to-1 encoder has four inputs and one output. The inputs are active high and the output shows one of the five possible states including a high impedance state if none of the inputs is high. It is also possible to construct priority encoders in quaternary logic system. Priority encoder is more practical than ordinary encoder since it can handle the common problem of having more than one input high at a time. In Fig. 8 and 9, the final stage of the encoder is a tri-state buffer. If at least one of the inputs is absolute high, the enable pin of the tri-state buffer will turn it on due to the operation of OR gate right before it. The buffer will produce a high impedance state if all inputs are absolute low.

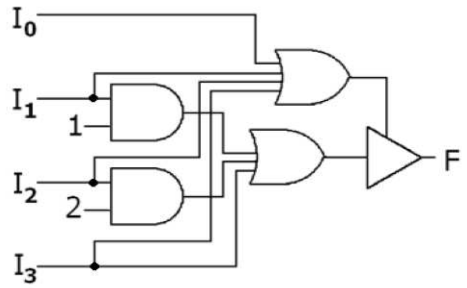


Figure 8. Quaternary 4-to-1 encoder with active high input and output. At a time, one of the input terminals is high (3) and the output F encodes the line number. If no input is high, then the buffer is turned off and the output will be at high impedance state.

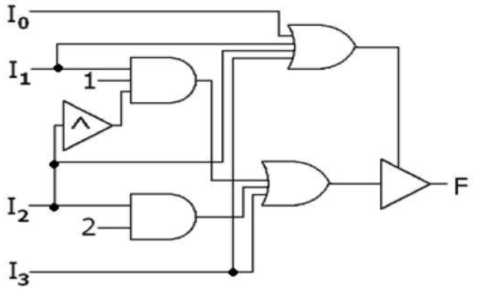
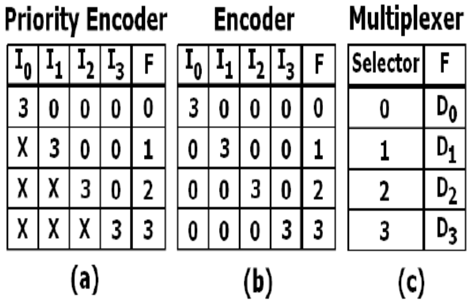


Figure 9. Quaternary 4-to-1 priority encoder. The higher order line has higher priority, so the priority is 3 > 2 > 1 > 0. The input and output lines are active high. If a line of higher priority is high (3), all lines of lower priority are ignored and that line is encoded at output. If none of the inputs is high, then the buffer is turned off and the output is at high impedance state.



Priority Encoder

I_0	I_1	I_2	I_3	F
3	0	0	0	0
X	3	0	0	1
X	X	3	0	2
X	X	X	3	3

(a)

Encoder

I_0	I_1	I_2	I_3	F
3	0	0	0	0
0	3	0	0	1
0	0	3	0	2
0	0	0	3	3

(b)

Multiplexer

Selector	F
0	D_0
1	D_1
2	D_2
3	D_3

(c)

Figure 10. Truth tables: (a) Quaternary 4-to-1 priority encoder with priority 3 > 2 > 1 > 0. (b) Quaternary 4-to-1 ordinary encoder that can handle only one input as high at a time. (c) Quaternary 4-to-1 multiplexer

V. CONCLUSION

In this paper we have proposed a new quaternary logic system. A set of operators are defined; some of which are closely related to Boolean operators, others are chosen for special purposes.

UACEE
Globalize The Research
Localize The World

Many properties of these operators are presented here and a functional quaternary algebra is developed. This new quaternary system is capable of handling ordinary quaternary inputs as well as coupled-binary inputs. we have designed some very useful combinational logic blocks such as decoder, multiplexer, demultiplexer, encoders.

REFERENCES

[1] Jahangir, I.; Hasan, D. M. N.; Islam, S.; Siddique, N. A.; Hasan, M. M.; "Development of a novel quaternary algebra with the design of some useful logic blocks", *ICCIT 2009 - 12th International Conference on Computers and Information Technology*, pp. 197-202, Dhaka, 21-23 Dec. 2009.

[2] Jahangir, I.; Hasan, D. M. N.; Reza, M. S.; "Design of some quaternary combinational logic blocks using a new logic system", *TENCON 2009 - 2009 IEEE Region 10 Conference*, pp. 1 - 6, Singapore, 23-26 Jan.
2009.

[3] http://www.ternarylogic.com

[4] Hurst, S. L; "Multiple-Valued Logic - Its Status and Its Future", Computers, IEEE Transactions onVol. C-33, Issue: 12, pp.1160-1179, Dec. 1984

[5] Chattopadhyay, T.; Roy, J. N.; "Polarization encoded all-optical quaternary multiplexer and demultiplexer - A proposal", Optik – International Journal for Light and Electron Optics, In Press, Corrected Proof, 11 June 2008.

[6] Khan, M.; Biswas, A.K.; Chowdhury, S.; Tanzid, M.; Mohsin, K.M.;Hasan, M.; Khan, A.I.; “Quantum realization of some quaternary circuits”, TENCON 2008, IEEE Region 10 Conference, 19-21 Nov. 2008.

[7] Inaba, M; Tanno, K and Ishizuka,O, “Realization of NMAX and NMIN functions with multi-valued voltage comparators’’, Multiple-Valued Logic, 2001. Proceedings, 31st IEEE International Symposium,pp.27 - 32, 22-24 May, 2001.

UACEE
Globalize The Research
Localize The World

UACEE
Globalize The Research
Localize The World