

# Reversible Realization of Quaternary Decoder, Multiplexer, and Demultiplexer Circuits

Mozammel H. A. Khan, *Member, IAENG*

**Abstract**—A quaternary reversible circuit is more compact than the corresponding binary reversible circuit in terms of number of input/output lines required. Decoder, multiplexer, and demultiplexer are very important building blocks of digital systems. In this paper, we show reversible realization of these circuits using quaternary shift gates (QSG), quaternary controlled shift gates (QCSG), and quaternary Toffoli gates (QTG). We also show the realization of multi-digit QCSG and QTG using QSGs and QCSG.

**Index Terms**— Decoder, demultiplexer, multiple-valued logic, multiplexer, quaternary logic, reversible logic

## I. INTRODUCTION

Reversible logic [1] is a very prospective approach of logic synthesis for power reduction in future computing technologies. In a reversible circuit, the number of inputs and outputs are same and there is a one-to-one mapping between input values and output values. Reversible circuits are constructed using reversible gates.

Quaternary logic is very suitable for encoded realization of binary logic functions by grouping 2-bits together into quaternary digits. This sort of quaternary encoded reversible realization of binary logic function requires half times input/output lines than the original binary reversible realization. As the number of input/output lines is reduced, this quaternary encoded realization of binary logic function makes the circuit more compact and manageable, especially for the quantum technology, where the cost of qudit (quantum digit) realization and qubit (quantum bit) realization are almost same. However, quaternary encoded realization of binary logic function requires encoding of binary inputs into quaternary values and decoding of quaternary outputs into binary values. Reversible realization of such binary-to-quaternary encoder and quaternary-to-binary decoder are discussed in [2] in the context of quaternary quantum logic. Similar realization can also be done in other quaternary reversible technologies.

Quaternary reversible logic synthesis is a very new research area [2-4]. Paper [2] presented a quaternary reversible logic synthesis method using quaternary Galois field sum of products (QGFSOP) expression and paper [3] presented determination of QGFSOP expression for a given function using quaternary Galois field decision diagram (QGFDD). Papers [2, 4] presented realization of quaternary

Toffoli gates (QTG) needed for synthesis of QGFSOP expression. Experience shows that CAD based general synthesis methods are sometimes not very cost effective for synthesis of some medium-scale building blocks. In this case, problem specific manual synthesis of such medium-scale building blocks is relatively cost effective.

Decoder, multiplexer, and demultiplexer are very important building blocks of digital systems. In this paper, we show a problem specific manual approach of reversible realization of these circuits using quaternary shift gates (QSG), quaternary controlled shift gates (QCSG), and quaternary Toffoli gates (QTG). We also show the realization of multi-digit QCSG and QTG using QSGs and QCSG.

## II. BACKGROUND ON GF(4) OPERATIONS

Galois field (4), abbreviated as GF(4), is a finite field  $F = \{0, 1, 2, 3\}$  with two binary operations – addition (denoted by  $+$ ) and multiplication (denoted by  $\cdot$  or absence of any operator) as defined in Table 1. Addition and multiplication operations are both commutative and associative. Multiplication operation is distributive over addition operation. For more details about GF(4) operations, see [2].

Table 1. GF(4) operations

$+$	0	1	2	3	$\cdot$	0	1	2	3
0	0	1	2	3	0	0	0	0	0
1	1	0	3	2	1	0	1	2	3
2	2	3	0	1	2	0	2	3	1
3	3	2	1	0	3	0	3	1	2

## III. QUATERNARY SHIFT GATES

There are  $4! = 24$  possible permutations of 0, 1, 2, and 3. Therefore, there are 24 possible truth tables for 1-digit reversible gates. Among these 24 1-digit reversible gates, we will use only four gates of the form  $y = x + z$  [GF(4)] for  $z \in \{0, 1, 2, 3\}$ . The GF(4) expressions and corresponding truth tables of these gates are shown in Table 2. We will refer the 1-digit gates of Table 2 as *quaternary shift gates* (QSG). The gate for  $y = x + 0$  can be regarded as a buffer gate or a wire. The QSGs can be realized using liquid ion-trap quantum technology [5] and other reversible technologies. We will graphically represent the QSGs using the symbol of Figure 1.

Two QSGs in cascade behave like another QSG. Equivalent QSG corresponding to all cascade pairs are given in Table 3, which can be easily verified using GF(4) expressions or truth tables of QSGs from Table 2. If  $a$

Manuscript received September 28, 2007.

Mozammel H. A. Khan is with the Department of Computer Science and Engineering, East West University, 43 Mohakhali, Dhaka 1212, Bangladesh, phone: +880-2-9882308; fax: +880-2-8812336; e-mail: mhakhan@ewubd.edu.

*cascade pair of two QSGs is equivalent to +0 gate, then the second QSG is said to be the inverse of the first QSG. Inverse gates are used to restore the input signal for reuse in the circuit.*

Table 2. GF(4) expressions and truth tables for quaternary shift gates

Input $x$	$y = x+0$	$y = x+1$	$y = x+2$	$y = x+3$
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

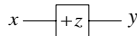


Figure 1. Symbol of quaternary shift gates (QSG)

Table 3. Equivalent QSGs corresponding to cascade pairs of two QSGs

		2 <sup>nd</sup> gate		
		+1	+2	+3
1 <sup>st</sup> gate	+1	+0	+3	+2
	+2	+3	+0	+1
	+3	+2	+1	+0

IV. QUATERNARY CONTROLLED SHIFT GATES

The symbol of the 2-digit *quaternary controlled shift gate* (QCSG) is shown in Figure 2. The gate applies a shift operation (Table 2) on the controlled input  $x_2$  when the controlling input  $x_1$  is 3. The outputs of the gate are  $y_1 = x_1$  and  $y_2 = x_2 + z$  if and only if  $x_1 = 3$ ;  $y_2 = x_2$  otherwise. The QCSG can be realized using liquid ion-trap quantum technology [5] and other reversible technologies.

We propose an  $n$ -digit QCSG using the symbol of Figure 3. In Figure 3,  $x_1 \dots x_{n-1}$  are  $n-1$  controlling inputs and  $x_n$  is the controlled input. The controlled output is  $y_n = x_n + z$  if and only if  $x_1 = 3 \wedge x_2 = 3 \wedge \dots \wedge x_{n-1} = 3$ ;  $y_n = x_n$  otherwise.

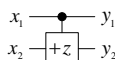


Figure 2. Symbol of 2-digit quaternary controlled shift gate (QCSG)

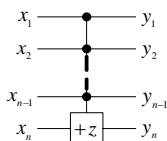


Figure 3. Symbol of  $n$ -digit quaternary controlled shift gate (QCSG)

The realization of 3-digit QCSG using 2-digit QCSGs is shown in Figure 4. From Figure 4, we see that if and only if  $x_1 = 3 \wedge x_2 = 3$ , then  $a = 3$  and  $+z$  shift is applied on  $x_3$ . For all other combinations of  $x_1$  and  $x_2$ ,  $a$  will never be 3 and  $+z$  shift will not be applied on  $x_3$ . The right most two 2-digit QCSGs are inverse gates of the left most two QCSGs and are used to restore the input constant 0 for reuse in the circuit. An auxiliary constant input used in the design of a reversible circuit is called ancilla digit. This realization needs 5 elementary gates and 1 ancilla digit. The realization of 4-digit QCSG using 2-digit QCSGs is shown in Figure 5. In Figure 5, if and only if  $x_1 = 3 \wedge x_2 = 3$ , then  $a = 3$ . Again, if and only if  $a = 3 \wedge x_3 = 3$ , then  $b = 3$ . This implies that,  $b = 3$  if and only if  $x_1 = 3 \wedge x_2 = 3 \wedge x_3 = 3$  and  $+z$  shift is applied on  $x_4$  controlled input. This realization needs 11 elementary gates and 2 ancilla digits. In this way, we can realize an  $n$ -digit QCSG. We can easily show that for an  $n$ -digit QCSG, the number of ancilla digits is  $(n - 2)$  and the number of elementary gates is  $4(n - 2) + 1$ .

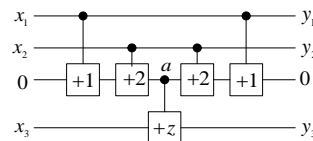


Figure 4. Realization of 3-digit quaternary controlled shift gate (QCSG)

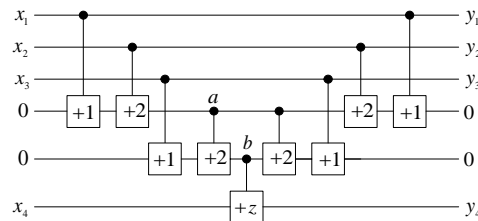


Figure 5. Realization of 4-digit quaternary controlled shift gate (QCSG)

V. QUATERNARY TOFFOLI GATE

The  $n$ -digit *quaternary Toffoli gate* (QTG) is shown in Figure 6, where  $x_1 \dots x_n$  are the inputs,  $y_i = x_i$  (for  $i = 1, 2, \dots, n-1$ ) are the pass through outputs and  $y_n = x_1 x_2 \dots x_{n-1} + x_n$  [GF(4)] is the controlled output. Realization of QTG using QCSGs and 2-digit QCSGs is discussed in [2].

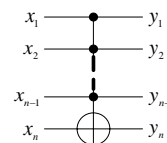


Figure 6. Quaternary Toffoli gate (QTG)

VI. REVERSIBLE REALIZATION OF QUATERNARY DECODER

The truth table of  $2 \times 16$  quaternary decoder with active-1 output is shown in Table 4. For a given select input combination  $A_1A_0$ , only the selected output will be 1 and the remaining outputs will be 0. Reversible realization of  $2 \times 16$  quaternary decoder with active-1 output is shown in Figure 7. The outputs are generated along constant 0 inputs. For a given select input combination  $A_1A_0$ , the input values are made 33 by applying QSGs along  $A_1A_0$  lines and then a QCSG is used to change the corresponding 0 input to output 1, the other outputs remain 0. For example, for select input combination  $A_1A_0 = 00$ , two +3 QSGs are placed along  $A_1A_0$  lines to make the input values 33 and then a QCSG is used to change the corresponding 0 input to output  $O_0 = 1$ . Outputs  $O_1$  through  $O_{15}$  are realized using the same technique. However, the input values corresponding to the QCSGs are made 33 by cascaded QSGs. Therefore, the effective shift at all controlling points are explicitly shown in the figure to make the circuit easily understandable. It can be seen that for a given combination of  $A_1A_0$ , only one QCSG will be active and will apply a +1 shift on the corresponding constant 0 input to produce the 1 output and other outputs will remain 0.

Quaternary decoder of any size can be realized using the

same technique. Moreover, the outputs may be made active-2 or active-3 by simply replacing +1 shifts of the QCSGs by appropriate shifts.

VII. REVERSIBLE REALIZATION OF QUATERNARY MULTIPLEXER

The truth table of  $16 \times 1$  quaternary multiplexer is shown in Table 5, where the output  $O$  is equal to the selected multiplexer input. Reversible realization of the multiplexer is shown in Figure 8. Depending on the select input combination  $A_1A_0$ , one of the decoder outputs becomes 1 and the other decoder outputs remain 0s. The 0 decoder outputs multiplied with their corresponding multiplexer inputs become 0s and these 0s are added to the constant input 0 along the output line. The only 1 decoder output multiplied with the corresponding multiplexer input becomes exactly equal to the multiplexer input and is added with the constant input 0 along the output line to produce the multiplexer output. For example, if  $A_1A_0 = 00$ , then the decoder output  $O_0$  becomes 1 and the other decoder outputs become 0. In this case,  $I_0$  is multiplied by 1 and the other multiplexer inputs are multiplied by 0. Therefore, only  $I_0$  is added with the constant input 0 to produce the output  $O$ , which is exactly equal to  $I_0$ .

Table 4. Truth table of  $2 \times 16$  quaternary decoder with active-1 output

$A_1$	$A_0$	$O_{15}$	$O_{14}$	$O_{13}$	$O_{12}$	$O_{11}$	$O_{10}$	$O_9$	$O_8$	$O_7$	$O_6$	$O_5$	$O_4$	$O_3$	$O_2$	$O_1$	$O_0$
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	3	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	2	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
1	3	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
2	2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
2	3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
3	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
3	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

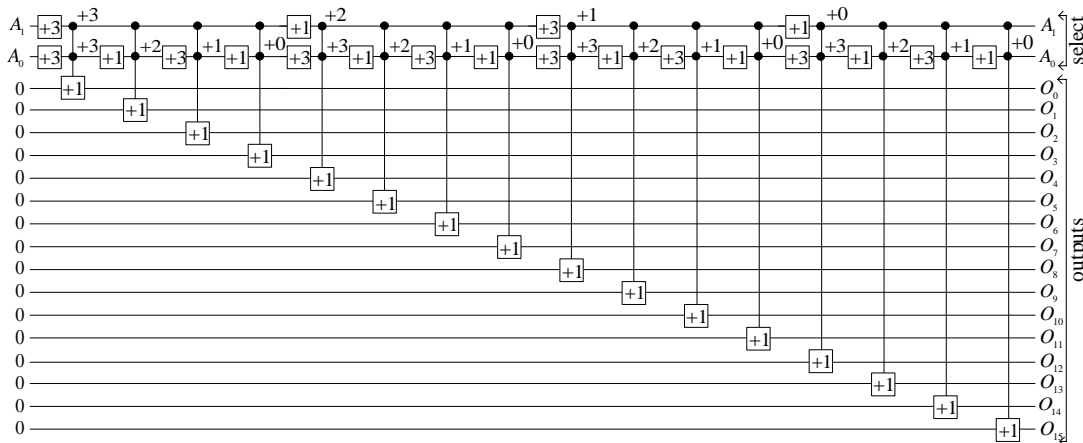


Figure 7. Reversible realization of 2x16 quaternary decoder with active-1 outputs

Table 5. Truth table of 16x1 quaternary multiplexer

$A_1A_0$	$O$	$A_1A_0$	$O$
00	$I_0$	20	$I_8$
01	$I_1$	21	$I_9$
02	$I_2$	22	$I_{10}$
03	$I_3$	23	$I_{11}$
10	$I_4$	30	$I_{12}$
11	$I_5$	31	$I_{13}$
12	$I_6$	32	$I_{14}$
13	$I_7$	33	$I_{15}$

VIII. REVERSIBLE REALIZATION OF QUATERNARY DEMULTIPLEXER

The truth table of 1x16 quaternary demultiplexer is shown in Table 6, where only the selected output is equal to the demultiplexer input  $I$  and the remaining outputs are 0s. Reversible realization of the demultiplexer is shown in Figure 9. Depending on the select input combination  $A_1A_0$ , one of the decoder outputs becomes 1 and the other decoder outputs remain 0s. The 0 decoder outputs are multiplied with the demultiplexer input  $I$  and become 0s. These 0s are then added with the constant input 0s to produce 0 outputs. The only 1 decoder output is multiplied with the demultiplexer input  $I$  to produce  $I$ . This  $I$  is then added with the constant input 0 to produce  $I$  on the selected output line. For example, if  $A_1A_0 = 00$ , then the decoder output  $O_0$  becomes 1 and the remaining decoder outputs become 0s. In this case,  $I$  is added only with the constant input 0 along the demultiplexer output line  $O_0$  and 0s are added with the other constant input 0s.

Therefore, only the demultiplexer output  $O_0$  will be equal to the demultiplexer input  $I$  and the other demultiplexer outputs will be 0s.

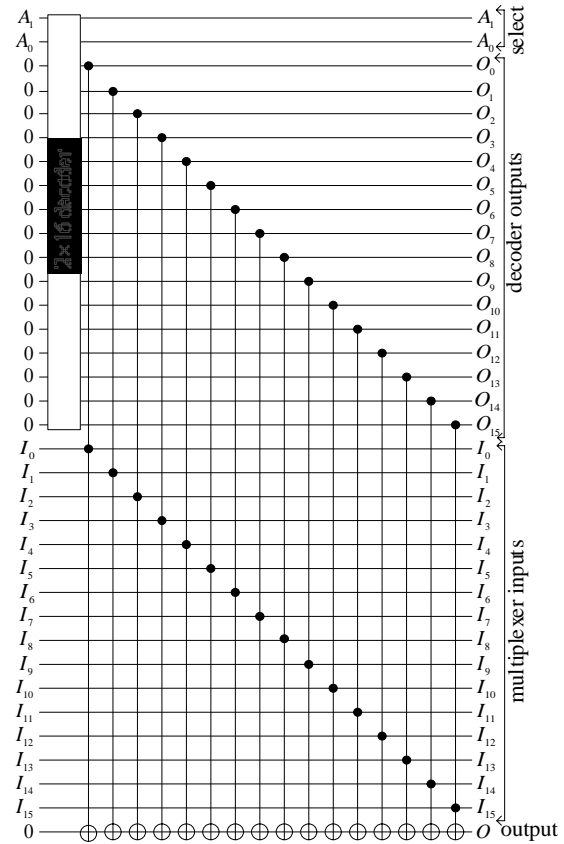


Figure 8. Reversible realization of 16x1 quaternary multiplexer

Table 6. Truth table of 1x16 quaternary demultiplexer

$A_1$	$A_0$	$O_{15}$	$O_{14}$	$O_{13}$	$O_{12}$	$O_{11}$	$O_{10}$	$O_9$	$O_8$	$O_7$	$O_6$	$O_5$	$O_4$	$O_3$	$O_2$	$O_1$	$O_0$	$I$
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$
0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
1	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
2	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0
3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	$I$	0

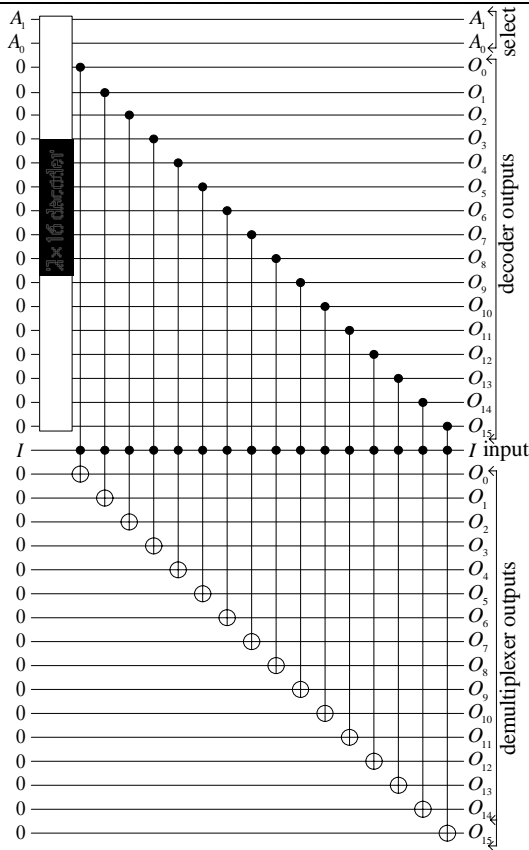


Figure 9. Reversible realization of 1x16 quaternary demultiplexer

IX. CONCLUSION

Multiple-valued reversible circuits are a promising choice for future computing technology. Quaternary logic has the advantage that classical binary logic functions can be expressed as quaternary logic functions by grouping 2-bits together into quaternary digits. Quaternary encoded reversible realization of binary logic function will be half-times compact than the reversible realization of original binary logic function in terms of the number of input/output lines required.

There is only a few works on CAD based general synthesis method of quaternary logic functions [2, 3]. Experience shows that CAD based general synthesis methods are sometimes not very cost effective for synthesis of some medium-scale building blocks. In this case, problem specific manual synthesis of such medium-scale building blocks is relatively cost effective. Here, we present problem specific manual synthesis of quaternary decoder, multiplexer, and demultiplexer circuits using reversible gates. These circuits are practically important medium-scale building blocks for synthesis of large digital systems.

These problem specific manual synthesis results will also help us to compare the complexity of CAD based general synthesis methods.

To aid the synthesis, we have used two macro-level quaternary gates – multi-digit quaternary controlled shift gate (QCSG) and quaternary Toffoli gates and shown their realization using basic quaternary gates.

REFERENCES

- [1] T. Toffoli., “Reversible Computing,” in *Automata, Languages and Programming*, J. W. Bakker and J. van Leeuwen, ED. Springer Verlag, 1980, pp. 632-644.
- [2] M. H. A. Khan and M. A. Perkowski, “GF(4) based synthesis of quaternary reversible/quantum logic circuits,” *Journal of Multiple-Valued Logic and Soft Computing*, vol. 13, 2007.
- [3] M. H. A. Khan, N. K. Siddika, and M. A. Perkowski, “Minimization of quaternary Galois field sum of products expression for multi-output quaternary logic function using quaternary Galois field decision diagram,” *38th IEEE Int. Symp. On Multiple-Valued Logic (ISMVL 2008)*, Dallas, Texas, USA, 22-24 May 2008 (submitted).
- [4] M. H. A. Khan, “Quantum realization of quaternary Feynman and Toffoli gates,” *International Conference on Electrical and Computer Engineering (ICECE 2006)*, Dhaka, Bangladesh, 19-21 December, 2006, pp. 157-160.
- [5] A. Muthukrishnan and C. R. Stroud Jr., “Multivalued logic gates for quantum computation,” *Physical Review A*. vol. 62, 2000, 052309/1-8.