

Multi-Output Galois Field Sum of Products Synthesis with New Quantum Cascades

Mozammel H. A. Khan[§], Marek A. Perkowski^{*}, Pawel Kerntopf[†]

[§]*Department of Computer Science and Engineering, East West University, 45 Mohakhali, Dhaka 1212, BANGLADESH. mhakistan@ewubd.edu*

^{*}*Department of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology, 373-1 Gusong-dong, Yusong-gu, Daejeon 305-701, KOREA. mperkows@ee.kaist.ac.kr*

[†]*Institute of Computer Science, Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warsaw, POLAND. pke@ii.pw.edu.pl*

Abstract

Galois Field Sum of Products (GFSOP) leads to efficient multi-valued reversible circuit synthesis using quantum gates. In this paper, we propose a new generalization of ternary Toffoli gate and another new generalized reversible ternary gate with discussion of their quantum realizations. Algorithms for synthesizing ternary GFSOP using quantum cascades of these gates are proposed. In both the synthesis methods, 5 ternary shift operators and ternary swap gate are used. We also propose quantum realizations of 5 ternary shift operators and ternary swap gate. In the cascades of the new ternary gates, local mirrors, variable ordering, and product ordering techniques are used to reduce the circuit cost. Experimental results show that the cascade of the new ternary gates is more efficient than the cascade of ternary Toffoli gates.

1. Introduction

Multi-valued quantum logic synthesis methods are still very immature, though a number of works have been done [3-6, 34]. From these works, however, it is more or less evident that Galois Field Sum of Products (GFSOP) is a good choice for multi-valued reversible logic synthesis. In this paper, we focus only on ternary GFSOP synthesis with cascades of quantum gates.

The unit of memory (information) for binary quantum computation is a *qubit*, the simplest quantum system that exists in a linear superposition of two basic states labeled $|0\rangle$ and $|1\rangle$. In 1996, Mattle et al [29] used the term *trit* for a ternary equivalent of qubit (however, *qutrit* is appropriate). In 1997, Chau [11] introduced the concept of a *qudit*, a d -dimensional quantum system that generalizes a qubit and has basis states $|0\rangle, |1\rangle, |2\rangle, \dots, |d-1\rangle$. Subsequently, limited work was done in multi-valued quantum logic. The work of Chau [11], Rains [37] and Ashikhmin and Knill [7], extended quantum error-correcting codes to multi-valued

logic for correcting codes in single and multiple qudits. Gottesman [19] and Aharonov and Ben-Or [2] developed fault-tolerant procedures for implementing two-qudit and three-qudit analogs of universal binary gates. Burlakov [10] proposed to use correlated photon pair to represent qutrit. Since 2000 the works have got momentum. Muthukrishnan and Stroud [32] developed multi-valued logic for multi-level quantum computing systems and showed their realizability in linear ion trap devices. However, this approach produces circuits of too large dimensions. Universality of n -qudit gates was discussed in [9, 32] but no design algorithms were given. Picton [35] presented an approach called Universal Architecture for multi-valued reversible logic but this approach produces circuits that are far from minimum and have no relation to quantum realization. Since 2001 Al-Rabadi et al proposed Galois Field approach to quantum logic synthesis (see [3-6, 34]). In this work Galois quantum matrices were proposed for swap and Toffoli gates, but without the proof that they can be built from only $1*1$ and $2*2$ gates. Several regular structures for multi-valued quantum logic were also proposed, including cascades, but these cascades do not allow realization of powers of GFSOP and are thus non-universal. This work was based on previous works on GFSOPs and similar forms of Galois and similar logic [1, 8, 12, 14-18, 20-25, 27, 31, 33, 36, 38-41], in which canonical expansions of Post literals and arbitrary functions were shown. However, no constructive methods for GFSOP and cascade minimization were given, nor programs were written for them. Factorized reversible cascades and complex gates (which usually yield better result) were not proposed. De Vos proposed two ternary $1*1$ gates and two ternary $2*2$ gates [13], but no synthesis method was proposed. New efficient reversible multi-valued gates were proposed in [26] and quantum realizations of multi-valued Toffoli gate in [33]. However, very little has been published on synthesis algorithms for multi-output multi-valued quantum circuits. Therefore, it is very important to look for efficient methods to synthesize *multi-output GFSOP* functions using quantum cascades. In this paper, we concentrate on quantum cascaded realization of only ternary GFSOP functions.

The rest of the paper is organized as follows. Section 2 provides brief introduction to ternary Galois Field logic. Multi-valued quantum logic is briefly discussed in Section 3. In Section 4, a new generalization of ternary Toffoli gate is proposed. A new generalized reversible ternary gate is proposed in Section 5. In Section 6, synthesis of multi-output ternary GFSOP with quantum cascade of new Toffoli gates is proposed. Synthesis of multi-output ternary GFSOP with quantum cascade of new ternary gates is proposed in Section 7. In Section 8, experimental results are presented. Conclusions about the paper and future research ideas are presented in Section 9. Sections 10 and 11 provide acknowledgement and references, respectively.

2. Ternary Galois field logic

In Galois Field Sum of Products (GFSOP) the product terms are GF products and the sums are GF sum operations. In this paper we concentrate only on ternary GFSOPs. Ternary Galois Field (GF3) consists of the set of elements $T = \{0, 1, 2\}$ and two basic binary operations – **addition** (denoted by $+$) and **multiplication** (denoted by \cdot or absence of any operator) as defined in Table 1. GF3 addition and multiplication are closed, i. e., for $A, B \in T$, $A+B \in T$ and $AB \in T$. GF3 addition and multiplication are also commutative and associative, i. e., $A+B = B+A$ and $AB = BA$ (commutative), and $A+(B+C) = (A+B)+C = A+B+C$ and $A(BC) = (AB)C = ABC$ (associative). GF3 multiplication is distributive over addition, i. e., $A(B+C) = AB+AC$.

Table 1. Ternary Galois Field (GF3) operations.

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

There are six reversible ternary unary operations corresponding to six possible permutations of 0, 1, and 2. These unary operations are called **reversible ternary shift operations**. We propose names of these six shift operations, their operator symbols and equations, and gate symbols in Table 2. Among these six shift operations only single-shift, dual-shift (both also called *Post cycles* [28] and *cyclic negations* in [30]), and self-dual-shift (also called *inverse* [16]) were previously used in the context of quantum computation. All these six shift operators can be built as reversible ternary gates. We propose quantum realization of the ternary shift gates (except the buffer, which is quantum wire) in Figure 1. These realizations require two to three quantum wires. However, we hope that physicists will be able to construct these gates as 1*1 gates. A ternary signal can be converted to one of the six forms using one of the

reversible ternary shift gates as shown in Table 3. We have used this technique in our cascades to change signal form.

Table 2. Reversible ternary shift operations.

Input A	Operator Names, Symbols, and Equations					
	B A	SS $A' = A+1$	DS $A'' = A+2$	SIS $A''' = 2A$	SISS $A^\# = 2A+1$	SIDS $A^\wedge = 2A+2$
0	0	1	2	0	1	2
1	1	2	0	2	0	1
2	2	0	1	1	2	0

B: Buffer, **SS:** Single-Shift, **DS:** Dual-Shift, **SIS:** Self-Shift, **SISS:** Self-Single-Shift, **SIDS:** Self-Dual-Shift

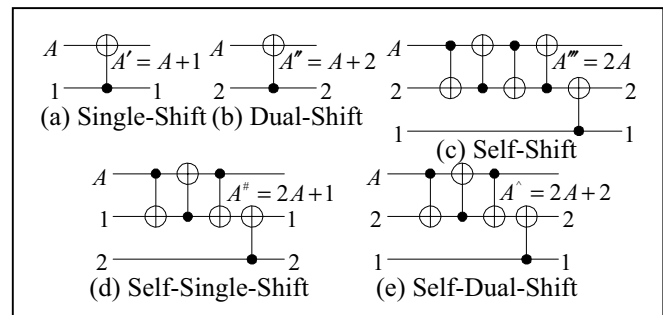
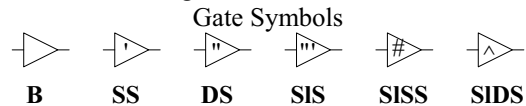


Figure 1. Quantum realization of ternary shift gates.

Table 3. Conversion of one shift form to another shift form using ternary shift gates.

Input	Output					
	A	A'	A''	A'''	$A^\#$	A^\wedge
A		SS	DS	SIS	SISS	SIDS
A'	DS		SS	SISS	SIDS	SIS
A''	SS	DS		SIDS	SIS	SISS
A'''	SIS	SISS	SIDS		SS	DS
$A^\#$	SISS	SIDS	SIS	DS		SS
A^\wedge	SIDS	SIS	SISS	SS	DS	

The GF3 basic literal of a variable A is an element of the set $\{1, 2, A, A', A'', A''', A^\#, A^\wedge, A^2\}$. It should be noted that all ternary literals, except A^2 , are reversible. A reversible ternary literal multiplied by 2 yields another reversible ternary literal as follows: $2 \cdot 1 = 2$, $2 \cdot 2 = 1$, $2A = A'''$, $2A' = A^\wedge$, $2A'' = A^\#$, $2A''' = A$, $2A^\# = A''$, and $2A^\wedge = A'$. Again, a ternary literal may have a power of only 2, since $A^3 = A$ (can be verified from Table 1), $A^4 = A^3 A = A^2$, and so on. A product term is a GF3 product of some literals. For example, AB'' is a product term. Ternary GFSOP is GF3 sum of some product terms. For example, $2 + AB'' + B^2 C' + A' C''$ is a ternary GFSOP.

3. Multi-valued quantum logic

In multi-valued (MV) Quantum Computing (QC), the unit of memory (information) is *qudit*. MV quantum logic operations manipulate qudits, which are microscopic entities such as a photon or atomic spin. Ternary logic values of 0, 1, and 2 are represented by a set of distinguishable different states of a *qutrit*. These states can be a photon's polarizations or an elementary particle's spins. After encoding these distinguishable quantities into multiple-valued constants, qutrit states are represented by the notations $|0\rangle$, $|1\rangle$, and $|2\rangle$.

Qudits exist in a linear superposition of states, and are characterized by a wavefunction ψ . As an example ($d=2$), it is possible to have light polarizations other than purely horizontal or vertical, such as slant 45° corresponding to the linear superposition of $\psi = \frac{1}{2}[\sqrt{2}|0\rangle + \sqrt{2}|1\rangle]$. In ternary logic, the notation for the superposition is $\alpha|0\rangle + \beta|1\rangle + \gamma|2\rangle$. These intermediate states cannot be distinguished, rather a measurement will yield that the qutrit is in one of the basis states, $|0\rangle$, $|1\rangle$, or $|2\rangle$. The probability that a measurement of a qutrit yields state $|0\rangle$ is $|\alpha|^2$, state $|1\rangle$ is $|\beta|^2$, and state $|2\rangle$ is $|\gamma|^2$. The sum of these probabilities is one. The absolute values are required since, in general, α , β and γ are complex quantities.

Pairs of qutrits are capable of representing nine distinct states, $|00\rangle$, $|01\rangle$, $|02\rangle$, $|10\rangle$, $|11\rangle$, $|12\rangle$, $|20\rangle$, $|21\rangle$, and $|22\rangle$, as well as all possible superpositions of the states. This property is known as "entanglement", and may be mathematically described using the Kronecker product (tensor product) operation \otimes . As an example, consider two qutrits with $\psi_1 = \alpha_1|0\rangle + \beta_1|1\rangle + \gamma_1|2\rangle$ and $\psi_2 = \alpha_2|0\rangle + \beta_2|1\rangle + \gamma_2|2\rangle$. When the two qutrits are considered to represent a state, that state ψ_{12} is the superposition of all possible combinations of the original qutrits, where

$$\psi_{12} = \psi_1 \otimes \psi_2 = \alpha_1\alpha_2|00\rangle + \alpha_1\beta_2|01\rangle + \dots + \gamma_1\gamma_2|22\rangle.$$

Superposition property allows qubit states to grow much faster in dimension than classical bits, and qudits faster than qubits [32]. In a classical system, n bits represent 2^n distinct states, whereas n qutrits correspond to a **superposition** of 3^n states. In the above formula some coefficient can be equal to zero, so there exist a constraint bounding the possible states in which the system can exist. As observed in [32] – "Allowing d to be arbitrary enables a tradeoff between the number of qudits making up the quantum computer and the number of levels in each qudit". These all contribute to difficulty in understanding the concepts of quantum computing and creating efficient

analysis, simulation, verification and synthesis algorithms for QC. Generally, however, we believe that much can be learned from the history of Electronic Computer Aided Design as well as from MV logic theory and design, and the lessons learned should be used to design efficient CAD tools for MV quantum computing.

In terms of logic operations, anything that changes a vector of qudit states can be considered as an operator. These phenomena can be modeled using the analogy of a "quantum circuit". In a quantum circuit, wires do not carry ternary constants but correspond to 3-tuples of complex values, α , β , and γ . Quantum logic gates of the circuit map the complex values on their inputs to complex values on their outputs. Operation of quantum gates is described by matrix operations. Any quantum circuit is a composition of parallel and serial connections of blocks, from small to large. Serial connection of blocks corresponds to multiplication of their (unitary) matrices. Parallel connection corresponds to Kronecker multiplication of their matrices. So, theoretically, the analysis, simulation and verification are easy and can be based on matrix methods. Practically they are tough because the dimensions of the matrices grow exponentially.

4. A new generalization of ternary Toffoli gate

We propose a ternary generalization of Toffoli gate in Figure 2(a), where f_k is an arbitrary ternary function of the input variables A_1, A_2, \dots, A_k . By inspection of the truth table it can be shown that the gate is a ternary reversible gate. Depending on f_k and the value of n many possible gates can be constructed.

The method of implementing arbitrary non-reversible functions is shown in Figure 2(b). In this method an arbitrary non-reversible one-output function controls an arbitrary reversible operator. G^{-1} is an inverse gate of gate G . H is a reversible gate that includes arbitrary controlled ternary function. Several garbage signals can be internally created between G and G^{-1} . But, as G^{-1} gate uses these garbage signals to restore the constants, they do not contribute dramatically to the increase of the scratchpad register width. The original inputs are restored at the output terminals, thus allowing the inputs to be used in the next gates in the cascade. Of course, this approach is restricted to some types of cascades like those of sum-of-products type which reuse inputs. Using this method, the ternary Toffoli gate can be realized as shown in Figure 2(c).

5. A new generalized reversible ternary gate

We propose a new generalized reversible ternary gate in Figure 3(a), where f_k is an arbitrary ternary function of the input variables A_1, A_2, \dots, A_k , $f_k^* \in \{f_k', f_k''\}$, and $A_i^* \in \{A_i', A_i''\}$. By inspection of the truth tables it can be shown that all gates of the proposed family are ternary

reversible gates. Depending on f_k and the choice of the shift, many possible gates can be constructed. A special subset of gates of our present interest is shown in Figure 3(b). Realization of the gate of Figure 3(b) using the principle of Figure 2(b) is shown in Figure 3(c). In Figure 3(d), we propose quantum realization of **ternary swap gate**.

Input lines A_{k+1} and A_{k+2} of the new ternary gate are used as controlled inputs. Depending on the values of these input signals, the gate can be used in cascades in 16 different modes of operations as shown in Table 4.

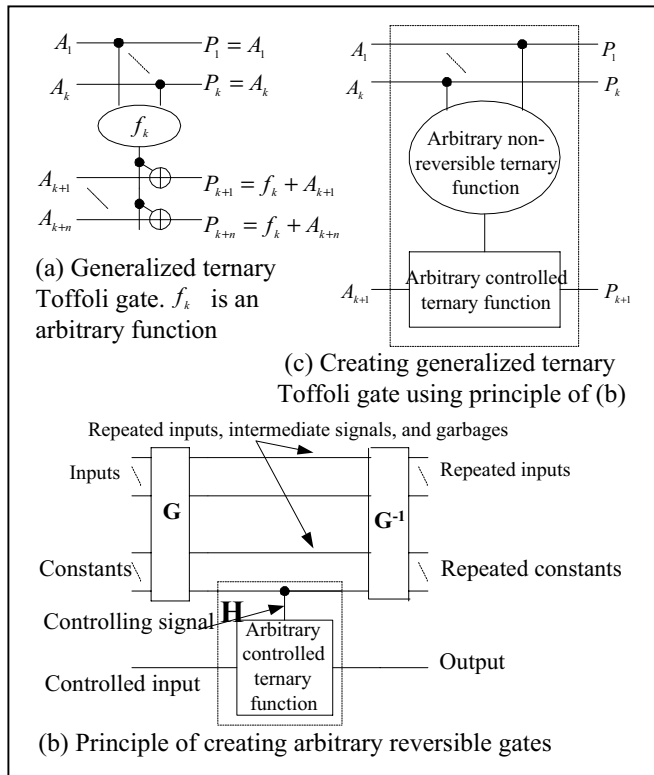


Figure 2. Generalized ternary Toffoli gate.

6. Synthesis of multi-output GFSOP using quantum cascade of ternary Toffoli gates

For synthesizing GFSOP using cascade of ternary Toffoli gates, we assume that $f_k = A_1 A_2 \cdots A_k$. The general pattern of a cascade to implement ternary GFSOP functions using ternary Toffoli gates is shown in Figure 4. It is obvious that to make it quantum realizable, swap gates have to be added. It should be observed that variables are replicated to allow realization of powers of variables and ternary functions that need them. Moreover, constant 2 is used to realize ternary functions that need it.

Theorem. Any ternary GFSOP function can be realized in a cascade of reversible ternary Toffoli, ternary Swap, and 5 ternary shift gates using at most $2n+2+m$ (optimistically $2n+1+m$) quantum wires, where n is the number of input variables and m is the number of outputs.

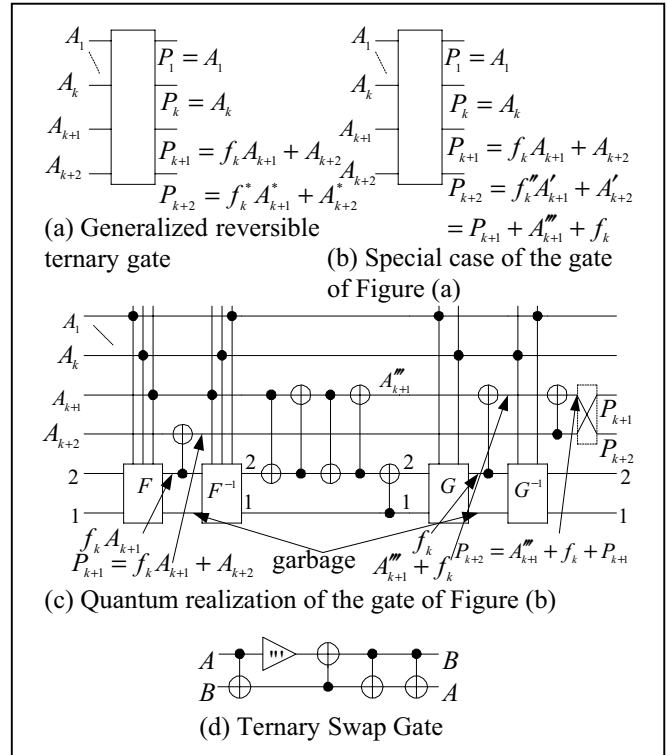


Figure 3. Generalized reversible ternary gate.

Proof. To realize a ternary GFSOP function, product terms can be implemented and added with other product terms using ternary Toffoli gates (see Figure 4). As wire crossing is not permitted in quantum circuits, ternary swap gates must be used. Ternary reversible literals can be implemented using ternary shift gates along the quantum wire. However, to realize literals of the form A^2 , two copies of the variables will be needed. Moreover, a product term of the form $A'A''$ can not be realized without a second copy of the variable. Therefore, $2n$ quantum wires will be required for n input variables. Such a variable copying can be done using a ternary Feynman gate as copying gate. It should be noted that Feynman gate is a special case of Toffoli gate. A reversible ternary literal multiplied by 2 yields another reversible ternary literal and that can be realized by using shift gates along the quantum wire. However, to realize a single variable function, literal of the form $2A^2$ must be realized, since ${}^0A = 2A^2 + 1$, ${}^1A = 2A^2 + 2A$, and ${}^2A = 2A^2 + A$. The product term of the form $2A^2$ can be realized by multiplication of 2, A , and A . For this purpose, a quantum wire for constant 2 will be required. Moreover, realization of shift gates requires quantum wires for constants 1 and 2 (see Figure 1). So, 2 quantum wires will be needed for input constants 1 and 2. Finally, m quantum wires will be needed for outputs. Therefore, a total of $2n+2+m$ quantum wires will be needed for realizing any ternary GFSOP function. However, for a particular function, copies of all input variables may not be required and, therefore, $2n+2+m$ is the maximum number of quantum

wires needed to realize a ternary GFSOP function. Anyway, we hope that physicists will be able to create all 5 ternary shift gates as 1*1 gates. In that situation, 2 quantum wires for input constants 1 and 2 to realize ternary shift gates will not be required; only 1 quantum wire for input constant 2 will be sufficient to realize the product terms like $2A^2$. In this optimistic situation, a maximum of $2n+1+m$ quantum wires will be needed to realize any ternary GFSOP function. **(End of proof)**

Table 4. Different modes of operation of the new reversible ternary gate family in cascades.

Mode	$A_{k+1}A_{k+2}$	P_{k+1}	P_{k+2}
A	00	0	f_k
B	01	1	f'_k
C	02	2	f''_k
D	10	f_k	f_k^\wedge
E	11	f'_k	$f_k^{\#}$
F	12	f''_k	$f_k^\#$
G	20	$f_k^{\#}$	1
H	21	$f_k^\#$	2
I	22	f_k^\wedge	0
J	0G	G	$G+f_k$
K	1G	f_k+G	$f_k^\wedge+G$
L	2G	$f_k^{\#}+G$	G'
M	G0	$f_k G$	$f'_k G'+G''$
N	G1	$f_k G+1$	$f'_k G'+G$
O	G2	$f_k G+2$	$f'_k G'$
P	GF	$f_k G+F$	$f_k G+F+G''' + f_k$

The proposed realization method automatically accomplishes the conversion of non-reversible ternary function to reversible ternary function, which is not a trivial problem by itself and has been not presented in the literature. Realization of multi-output GFSOPs $F_1 = 1+2B'C^2 + A^2 B''$, $F_2 = 1+B'+A^2 B'' C + C''$, and $F_3 = 2+2B'C^2 + 2A^2 B'' + B' + A^2 B'' C$ using quantum cascades of ternary Toffoli gates is shown in Figure 5.

7. Synthesis of multi-output GFSOP using quantum cascades of new ternary gates

For synthesizing GFSOP using quantum cascades of new ternary gates, we assume that $f_k = A_1 A_2 \dots A_{k-2}$. In this synthesis method we have used a graph based data structure called **implementation graph** as shown in Figure 6(a). In this graph a node represents a new ternary gate. The left incoming edge represents A_{k+1} input line and the right

incoming edge represents A_{k+2} input line. The left outgoing edge represents P_{k+1} output line and the right outgoing edge represents P_{k+2} output line. The product inside the node represents the function f_k . A constant output is converted into another constant, if required, using shift gate (see Table 3) represented in the implementation graph by a box with S inside. A garbage output is converted into 0 by using local mirror represented in the implementation graph by a box with M inside. In the local mirror, a garbage output of the form $2G$ is added with G using Toffoli gate to produce 0. A garbage output of the form G is first multiplied by 2 using self-shift gate to produce $2G$ and then added with G using Toffoli gate to produce 0. The resulting 0 is then converted into another constant, if required, using shift gate.

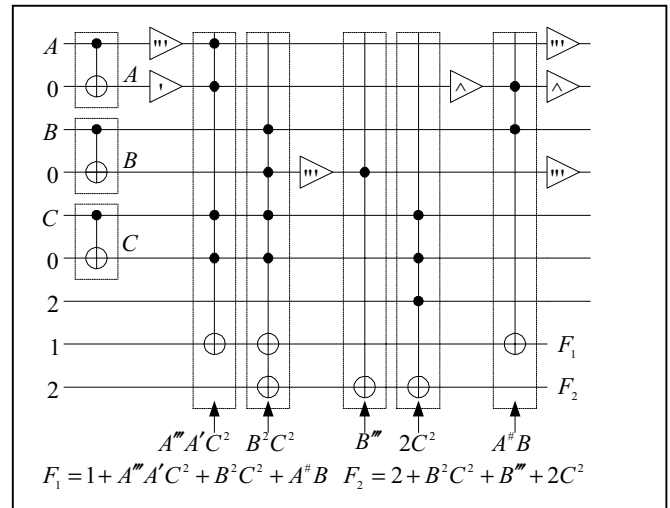


Figure 4. General pattern of a cascade to implement ternary functions using ternary Toffoli gates.

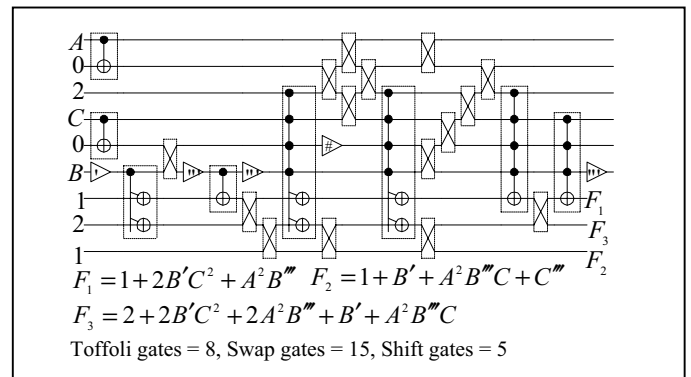


Figure 5. Synthesis of multi-output GFSOP using quantum cascade of ternary Toffoli gates.

The algorithm for synthesizing multi-output GFSOP using quantum cascades of new ternary gates is given below:
 1.1 Factorize the given GFSOPs to satisfy the structure of operating mode P.

- 1.2 If not possible, factorize the given GFSOPs to satisfy the structure of any of the operating modes of J, K, L, M, N, and O.
- 1.3 If not possible, factorize the given GFSOPs to satisfy the structure of any of the operating modes of D, E, and F.
- 1.4 If not possible, factorize the given GFSOPs to satisfy the structure of any of the operating modes of A, B, C, G, H, and I.
2. Create a node of the implementation graph for the selected mode of operation. Determine the input of that node.
3. Repeat steps 1 and 2 recursively for the inputs of the created node until all inputs become constant.
4. If any output of a node is garbage, use local mirror to convert it into constant. Convert output constants into other constants, if needed for one of the next gates, using shift gates.
5. From the implementation graph, realize the quantum cascade. Use variable and product ordering to reduce the number of swap gates.

The method is illustrated using the GFSOPs $F_1 = 1 + 2B'C^2 + A^2B'''$, $F_2 = 1 + B' + A^2B'''C + C'''$, and $F_3 = 2 + 2B'C^2 + 2A^2B''' + B' + A^2B'''C$. F_2 and F_3 can be factorized to satisfy the structure of mode N as $F_2 = 1 + B' + A^2B'''C + C''' = C(B'C^2 + 2A^2B + 2) + 1$ and $F_3 = 2 + 2B'C^2 + 2A^2B''' + B' + A^2B'''C$

$$= (C+1)(B'C^2 + 2A^2B + 2) + (B'C^2 + 2A^2B + 2)'$$

where the function $f_k = C$ and the corresponding inputs are $B'C^2 + 2A^2B + 2$ and 1. This selection is represented by the bottom node of Figure 6(a). F_1 can be factorized as $F_1 = 1 + 2B'C^2 + A^2B''' = (2B'C^2 + 2) + (2A^2B + 2)$. The input $B'C^2 + 2A^2B + 2$ of the bottom node and the function F_1 can be generated using the operating mode K with $f_k = B'C^2$ and inputs 1 and $2A^2B + 2$. This selection is represented by the middle node of Figure 6(a). Finally, the input $2A^2B + 2$ can be realized using operating mode D with $f_k = A^2B$ and inputs 1 and 0. This selection is represented by the top node of Figure 6(a). The garbage output A^2B of the top node is converted into 1 using local mirror. The local mirror is denoted by a box with symbol M in the graph of Figure 6(a). Realization of the implementation graph is shown in Figure 6(b).

8. Experimental results

We have developed a heuristic method of GFSOP synthesis using both types of cascades. There are no known ternary GFSOP benchmark functions. For doing our experiments, we considered some small ternary GFSOP functions and gave them arbitrary names as shown in Table 5. These functions are realized using quantum cascade of Toffoli gates as well as quantum cascade of new ternary

gates. Experimental results for these functions are given in Table 6. From the table, it can be seen that for multi-output GFSOP the quantum cascades of new ternary gates are more efficient than the quantum cascades of ternary Toffoli gates. However, for single-output GFSOP the quantum cascades of ternary Toffoli gates are more efficient than the quantum cascades of new ternary gates.

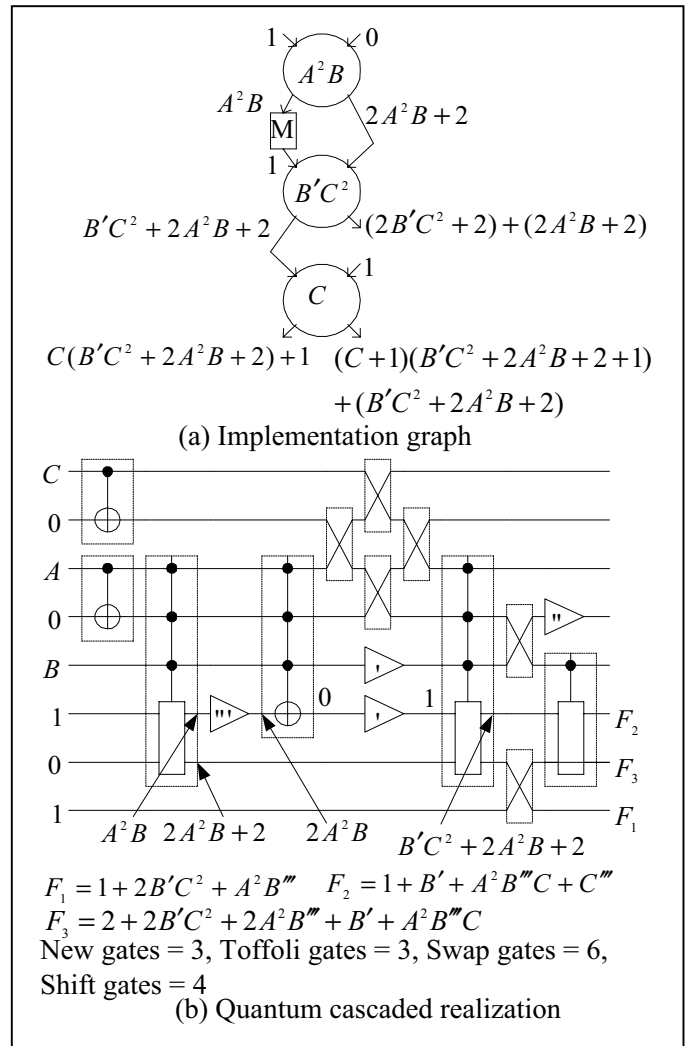


Figure 6. Synthesis of multi-output GFSOP using quantum cascade of new ternary gate.

9. Conclusion

In this paper, we used five (except buffer) reversible ternary unary operators (only three were previously used), which can be built as quantum 1*1 ternary gates. A new ternary generalization of Toffoli gate and a new generalized reversible ternary gate are proposed. Quantum realizations of these gates are also discussed. Two GFSOP-based reversible logic synthesis methods – one using quantum cascade of ternary Toffoli gates and another using quantum cascade of new ternary gates are proposed. We

have chosen cascaded realization, since cascades have the same number of internal signals at every level with small garbage or no garbage at all. Cascades can also be easily represented using standard quantum notation introduced by Feynman. For synthesizing quantum cascade of new ternary gates, **a graph-based data structure** called implementation graph **is introduced**. In this cascade local mirror is used to convert garbage output into constant and then the resulting constant is used as input to the next gate to reduce the scratchpad register width. Wire crossing is not allowed in some quantum technologies. In this case using a ternary swap gate is useful in logic synthesis of ternary reversible circuits. We **propose quantum realization of a ternary swap gate for the first time**. In both the cascades, variable ordering and product ordering are used to reduce the number of ternary swap gates, even then the cascades require a large number of ternary swap gates, which needs to be reduced in the future research. There are no known ternary GFSOP benchmark functions, so we performed experiments with some small GFSOPs and experimental results are reported. From the experimental results, it can be said that for multi-output GFSOP the quantum cascade of new gates is more efficient than the quantum cascade of Toffoli gates. However, quantum cascade of Toffoli gates yield better result for single-output GFSOP. Using smarter factorization techniques would also improve the quality of quantum cascades of new family of ternary gates.

Table 5. Experimental functions.

Name	GFSOP Expression
kpk01	$F_1 = A'B'C'' + B' + A'B$, $F_2 = A'B'C'' + A'B + A'C''$
kpk02	$F_1 = BC'D'E + AB'D'E + D'E$, $F_2 = BC'D'E + AB'D'E + D'E''' + B''C' + A'''B' + 2$
kpk03	$F_1 = AB'$, $F_2 = AB' + BC'$, $F_3 = AB' + AC''$, $F_4 = AB' + AC'' + AD'$
kpk04	$F = AC + AD'' + B'C + B'D'' + A'B'' + CD$
kpk05	$F_1 = 1 + A'BCD'$, $F_2 = 1 + A'BCD' + A'C + B''D'$, $F_3 = B'''C + BD' + AC''$, $F_4 = 1 + BD' + AC''$
kpk06	$F_1 = A''B'C^2 + A^2B^2$, $F_2 = 2 + A''B'C^2 + A^2B^2$
kpk07	$F_1 = 1 + 2B'C^2 + A^2B''$, $F_2 = 1 + B' + A^2B'''C + C''$, $F_3 = 2 + 2B'C^2 + 2A^2B''' + B' + A^2B'''C$

The quantum realizations of binary gates are called pseudo-binary, so by analogy we will call our circuits introduced here – **pseudo-ternary**. They will be both called “permutation circuits” because their unitary matrices are permutation matrices.

The proposed multi-output GFSOP synthesis methods are applicable to all kinds of polynomial expansions presented so far in literature as well as new expansions that

have operations of powers, multiplications, sums and reversible one-argument functions.

Table 6. Experimental results.

Func. Name	Cascade of Toffoli Gates			Cascade of New Gates			
	Toff. Gates	Swap Gates	Shift Gates	New Gates	Toff. Gates	Swap Gates	Shift Gates
kpk01	4	3	6	3	0	2	5
kpk02	6	12	12	3	2	10	9
kpk03	4	9	7	4	0	15	6
kpk04	6	14	7	6	10	25	20
kpk05	6	13	8	4	0	12	5
kpk06	6	6	5	2	3	5	5
kpk07	8	15	5	3	3	6	4

The future research includes: (i) investigating more efficient algorithm for reducing the number of swap gates in the cascades, (ii) developing smarter factorization technique for the cascade of new gates, (iii) developing method for multi-output GFSOP minimization, and (iv) creating a good library of ternary GFSOP benchmark functions.

10. Acknowledgements

The authors would like to thank Prof. Soonchil Lee and Dr. Jae-Seung Lee for help in understanding NMR-based quantum computing. However, the authors are the only ones to be blamed for possible mistakes and inaccuracies of this paper. The authors also thank Professors Soonchil Lee, Radomir Stankovic, Tsutomu Sasao and Bogdan Falkowski, and Dr. Anas Al-Rabadi, for help in finding literature and useful feedback on the draft paper.

11. References

- [1] K.J. Adams and J. McGregor, “Comparison of Different Features of Quaternary Reed-Muller Canonical Forms and Some New Statistical Results”, *Proc. 32nd IEEE Int. Symp. on Multiple-Valued Logic*, Boston, Massachusetts, 2002, pp. 83-88.
- [2] D. Aharonov and M. Ben-Or, “Polynomial Simulations of Decohered Quantum Computers”, (Online preprint quant-ph/9611029), *37th Annual Symp. on Foundations of Computer Science*, Burlington, Vermont, October 1996, pp. 4655.
- [3] A. Al-Rabadi, “Synthesis and Canonical Representations of Equally Input-Output Binary and Multiple-Valued Galois Quantum Logic: Decision Trees, Decision Diagrams, Quantum Butterflies, Quantum Chrestenson Gate, Multiple-Valued Bell-Einstein-Podolsky-Rosen Basis States”, *Technical Report #2001/008*, ECE Dept., PSU, August 2001.
- [4] A. Al-Rabadi, “Novel Methods for Reversible Logic Synthesis and Their Application to Quantum Computing”, *Ph. D. Thesis*, PSU, Portland, Oregon, USA, October 24, 2002.
- [5] A. Al-Rabadi, L. W. Casperson, M. Perkowski and X. Song, “Multiple-Valued Quantum Logic”, *Booklet of 11th Workshop on Post-Binary Ultra-Large-Scale Integration Systems (ULSI)*, Boston, Massachusetts, May 15, 2002, pp. 35-45.

- [6] A. Al-Rabadi and M. Perkowski, "Multiple-Valued Galois Field S/D Trees for GFSOP Minimization and their Complexity", *Proc. 31st IEEE Int. Symp. on Multiple-Valued Logic*, Warsaw, Poland, May 22-24, 2001, pp. 159-166.
- [7] A. Ashikhmin and E. Knill, "Non-binary quantum stabilizer codes", <http://citeseer.nj.nec.com/ashikhmin00nonbinary.html>
- [8] B. Benjauthrit and I.S. Reed, "Galois switching functions and their applications", *IEEE Trans. on Computers*, Vol. C-25, No. 1, 1976, pp. 79-86.
- [9] J. L. Brylinski and R. Brylinski, "Universal Quantum Gates", (to appear in *Mathematics of Quantum Computation*, CRC Press, 2002) LANL e-print quant-ph/010862.
- [10] A. V. Burlakov, M. V. Chekhova, O.V. Karabutova, D.N. Klyshko, and S. P. Kulik, "Polarization state of a biphoton: quantum ternary", *Physical Review A*, Vol. 60, R4209, 1999.
- [11] H. F. Chau, "Correcting quantum errors in higher spin systems", *Physical Review A*, Vol. 55, R839-R841, 1997.
- [12] M. Cohn, "Switching function canonical form over integer fields", *Ph.D. Thesis*, Harvard University, 1960.
- [13] A. De Vos, B. Raa, and L. Storme, "Generating the group of reversible logic gates", *Journal of Physics A: Mathematical and General*, Vol. 35, 2002, pp. 7063-7078.
- [14] K. M. Dill, K. Ganguly, R. J. Safranek, and M. A. Perkowski, "A New Zhegalkin Galois Logic", *Proc. 3rd Int. Workshop on Applications of the Reed-Muller Expansion in Circuit Design*, Oxford Univ., U.K., September 1997, pp. 247-257.
- [15] E. Dubrova, "Evaluation of m-valued Fixed Polarity Generalizations of Reed-Muller Canonical Form", *Proc. 29th Int. Symp. on Multiple-Valued Logic*, Freiburg, Germany, 1999, pp. 92-98.
- [16] E. Dubrova and J.C. Muzio, "Generalized Reed-Muller canonical form of a multiple-valued algebra", *Multiple-Valued Logic - An International Journal*, 1996, pp. 65-84.
- [17] B.J. Falkowski and S. Rahardja, "Efficient computation of quaternary fixed polarity Reed-Muller expansions", *IEE Proc., Part E*, Vol. 142, No. 5, 1995, pp. 345-352.
- [18] M.U. Garaev and R.G. Faradzhev, "On an analog of Fourier expansion over Galois Fields and its applications to problem of generalized sequential machines", *Izv. Akad. Nauk Azerb. SSR, Ser. Fiz.- Techn. Math. Nauk*, No. 6, 1965, pp. 1-68.
- [19] D. Gottesman, "Fault-tolerant quantum computation with higher-dimensional systems", *Chaos, Solitons, Fractals*, Vol. 10, No. 10, 1999, pp. 1749-1758.
- [20] D.H. Green, "Ternary Reed-Muller switching functions with fixed and mixed polarities", *Int. J. Electronics*, Vol. 67, No. 5, 1989, pp. 761-775.
- [21] D.H. Green, "Reed-Muller expansions with fixed and mixed polarities over GF(4)", *IEE Proc. Part E* Vol. 137, No. 5, 1990, pp. 380-388.
- [22] D.H. Green and I.S. Taylor, "Modular representations of multiple-valued logic systems", *IEE Proc., Part E*, Vol. 121, No.2, 1974, pp. 166-172.
- [23] B. Harking, and C. Moraga, "Efficient derivation of Reed-Muller expansions in multiple-valued logic systems", *Proc. 22nd Int. Symp. on Multiple-Valued Logic*, Sendai, Japan, May 27-29 1992, pp. 436-441.
- [24] D. Jankovic, R.S. Stankovic, and R. Drechsler, "Efficient Calculation of Fixed-Polarity Polynomial Expressions for Multiple-Valued Logic Functions", *Proc. 32nd IEEE Int. Symp. on Multiple-Valued Logic*, Boston, Massachusetts, May 2002, pp. 76-82.
- [25] U. Kalay, D.V. Hall, and M. Perkowski, "Easily Testable Multiple-Valued Galois Field Sum-of-Products Circuits", *Multiple Valued Logic - An International Journal*, Vol. 5, 2000, pp. 507-528.
- [26] P. Kerntopf, "Maximally efficient binary and multi-valued reversible gates", *Booklet of 10th Intl Workshop on Post-Binary Ultra-Large-Scale Integration Systems (ULSI)*, Warsaw, Poland, May 2001, pp. 55-58.
- [27] K. L. Kodandapani and R.V. Setur, "Multi-valued algebraic generalization of Reed-Muller canonical forms", *Proc. 4th Symp. on Multiple-Valued Logic*, 1974, pp. 505-544.
- [28] L. Macchiarulo and P. Civera, "Ternary Decision Diagrams with Inverted Edges and Cofactors - an Application to Discrete Neural Networks Synthesis", *Proc. 28th IEEE Int. Symp. on Multiple-Valued Logic*, 1998, pp. 58-63.
- [29] K. Mattle, H. Weinfurter, P.G. Kwiat, and A. Zeilinger, "Dense Coding in Experimental Quantum Communication", *Physical Review Letters*, Vol. 76, 1996, pp. 4656-4659.
- [30] D. M. Miller and R. Drechsler, "On the Construction of Multi-Valued Decision Diagrams", *Proc. 32nd IEEE Int. Symp. on Multiple-Valued Logic*, Boston, Massachusetts, 2002, pp. 245-253.
- [31] C. Moraga, R.S. Stankovic, and D. Jankovic, "Complexity analysis of Reed-Muller-Fourier and Galois Field Representations of Four-Valued Functions", *Research Report 668*, Dept. of Comp. Science, Univ. of Dortmund, 1998.
- [32] A. Muthukrishnan, and C. R. Stroud, Jr., "Multivalued logic gates for quantum computation", *Physical Review A*, Vol. 62, No. 5, Nov. 2000, 052309/1-8.
- [33] M. Perkowski, A. Al-Rabadi, and P. Kerntopf, "Multiple-Valued Quantum Logic Synthesis", *Proc. of 2002 International Symposium on New Paradigm VLSI Computing*, Sendai, Japan, December 12-14, 2002, pp. 41-47.
- [34] M. Perkowski, A. Al-Rabadi, P. Kerntopf, A. Mishchenko, and M. Chrzanoska-Jeske, "Three-Dimensional Realization of Multivalued Functions Using Reversible Logic", *Booklet of 10th Int. Workshop on Post-Binary Ultra-Large-Scale Integration Systems (ULSI)*, Warsaw, Poland, May 2001, pp. 47- 53.
- [35] P. Picton, "A Universal Architecture for Multiple-Valued Reversible Logic", *Multiple-Valued Logic - An International Journal*, Vol. 5, 2000, pp. 27-37.
- [36] D.K. Pradhan, "A multi-valued algebra based on finite fields", *Proc. 4th Symp. on Multiple-Valued Logic*, 1974, pp. 95-112.
- [37] E. M. Rains, "Nonbinary Quantum Codes", *IEEE Trans. on Information Theory*, Vol. 45, 1999, pp. 1827 -1832.
- [38] R.S. Stankovic, D. Jankovic, and C. Moraga, "Reed-Muller-Fourier versus Galois Field Representations of Four-Valued Logic Functions", *Proc. 3rd Int. Workshop on Applications of the Reed-Muller Expansion in Circuit Design*, Sept. 19-20, 1997, Oxford, UK, pp. 269-278.
- [39] R. S. Stankovic, and C. Moraga, "Reed-Muller-Fourier representations of multiple-valued functions over Galois fields of prime cardinality", *Proc. IFIP WG 10.5 Workshop on Applications of the Reed-Muller Expansion in Circuit Design*, Hamburg, Germany, September 16-17, 1993, pp. 115-124.
- [40] R. S. Stankovic, C. Moraga, "Reed-Muller-Fourier representations of multiple-valued functions", In *Recent Developments in Abstract Harmonic Analysis with Applications in Signal Processing*, Nauka, Belgrade, 1996, pp. 205-216.
- [41] T.C. Wesslekammer, "Divided difference method for Galois switching functions", *IEEE Trans. on Computers*, Vol. C-27, No. 3, 1978, pp. 232-238.