Indian Journal of Pure & Applied Physics
Vol. 56, July 2018, pp. 529-532

Reversible circuits with testability using quantum controlled NOT and swap gates

Hari Mohan Gaur[a]*, Ashutosh Kumar Singh[b] & Umesh Ghanekar[a]

[a]Department of Electronics & Communication Engineering, N I T Kurukshetra 136 119, India
[b]Department of Computer Applications, N I T Kurukshetra 136 119, India

Received 2 August 2017; accepted 26 December 2017

A new method of designing reversible circuits with inbuilt testability is presented by exploiting the properties of quantum controlled NOT and Swap gates. The design process is based on the methodology of placement of gates in such a manner that it produces parity preserving circuits. The testability of these circuits can be achieved by comparing the input and output parity under single bit fault detection. Experiments are conducted on a set of benchmark circuits which show an average reduction up to 51% in operating costs, when compared to existing work.

Keywords: Reversible logic, Digital design, Quantum controlled gates: Fault testing, Bit faults

1 Introduction

Reversible logic circuits are theoretically proven for providing nearly energy free computation[1]. It has wide applications in the field of quantum computing, optical computing and nano-technology. A few physical realizations have already been reported[2,3]. The true functioning of these circuits is another issue where testing plays an important role in producing desired results. However, it has achieved a notable attention by the researchers for the detection of single/multiple bit faults, as any fault occurrence results in the change of single/multiple values of bits at the output wires of the circuit[4]. Based on reviews and analysis of the work in this area, parity checking is found most favourable due to the property reversible circuits that, they contain same number of inputs and outputs. Either parity is generated or preserved by means of testable gates[5,6] or any modification principles[7-9]. These processes largely increase the requirements of extra hardware in terms of number of gates, wires, garbage output and logic qubits, and consume lot of time to obtain testable circuits. As a result, the principle cost of the circuit drastically increased which generally accounts 30-60% of the cost of manufacturing in logic circuits[11]. Utilizing quantum controlled NOT and Swap gates, which are known as multiple control Toffoli (MCT) and Fredkin gates (MCF), respectively, we present a new methodology of designing reversible circuits which produces parity preserving circuits, rather modifying standard circuit or using new gates for the same. This property facilitates the ease of testing with these circuits by simply using a parity checker at the outputs, showing their inbuilt testability feature and eliminates the need of extra time and hardware which reduces the principle cost of testing.

*Corresponding author (E-mail: leoharimohan84@gmail.com)

2 Proposed Design for Testability Methodology

When parity preserved architecture are used with an arbitrary design methodology, it ensures detection of faults occurred due to single bit flip in logic circuits without additional hardware[12]. The proposed design methodology is engaged in the creation of parity preserving reversible circuits using MCT and MCF gates followed by the fault detection process. MCT gates are used in the proposed two fold design theory and MCF gates are used without any modification, to make parity preserving circuits. The fault detection process contains the inclusion of a parity checker in the designed circuit. Further sections discuss both the steps in detail.

2.1 Designing parity preserving circuits

An MCT gate has m control inputs (k_1, k_2, ..., k_m) and one target input T, as shown in Fig. 1(a). The control inputs follow same input to output relationship and the output function $f(k_m,T)$ can be calculated by Eq. (1), where i = {*null*, 1, 2}. The gate without any control is called a NOT gate, which inverts its target input.

$$f_i(k_m,T_i) = (k_1 \bullet k_2 \bullet \ldots \bullet k_m) \oplus T_i \quad \ldots (1)$$

Lemma 1: The circuit reproduced by cascading another MCT gate of same size with same control inputs and keeping target input on a wire other than the place assigned for previous gate is parity preserving.

Proof: For a circuit to be parity preserving, the exor of all the inputs and output should result a null value. Consider an MCT gate is cascaded keeping with same control inputs and keeping target input (T_2) on a wire other than the place assigned for T_1, as shown in Fig. 1(b). The new outputs $f_1(k_m,T_1)$ and $f_2(k_m,T_2)$ can be calculated by eq. (1). The exclusive sum of all the inputs (I) and outputs (O) calculated in Eq. (2) is a null value, where $k_{pr}=(k_1 \cdot k_2 \cdot \ldots \cdot k_m)$ and $k_{ex} = (k_1 \oplus k_2 \oplus \ldots \oplus k_n)$. Hence, the circuit produced is parity preserving:

$$I \oplus O = [k_{ex} \oplus T_1 \oplus T_2] \oplus [k_{ex} \oplus f_1(k_m,T_1) \oplus f_2(k_m,T_2)] = [k_{ex} \oplus T_1 \oplus T_2] \oplus [k_{ex} \oplus (k_{pr} \oplus T_1) \oplus (k_{pr} \oplus T_2)] = 0 \quad \ldots (2)$$

An MCF gate has m control inputs (k_1, k_2, ..., k_m) and two target inputs t_1 and t_2 to form a $(m+2)\times(m+2)$ reversible gate. The gate passes all the control inputs directly to respective outputs and the target outputs f_j are given by Eq. (3), where $k_{pr}=(k_1 \cdot k_2 \cdot \ldots \cdot k_m)$. The illustration is also shown in Fig. 1(c). The gate swap its target input only if $k_1 = k_2 = \ldots = k_m = 1$, else they directly passes to their corresponding outputs. A gate with $m = 0$ is called a swap gate, where $f_1 = t_2$ and $f_2 = t_1$.

$$f_j = k_{pr1} \bullet t_1 + k_{pr2} \bullet t_2 \begin{cases} for \ldots j=1; k_{pr1} = \overline{k_{pr}}, k_{pr2} = k_{pr} \\ for \ldots j=2; k_{pr1} = k_{pr}, k_{pr2} = \overline{k_{pr}} \end{cases} \quad \ldots (3)$$

Lemma 2: MCF gates are parity preserving.

Proof: Consider an MCF gate shown in Fig. 1(c), the exor of inputs (I) and output (O) calculated in Eq. (4) results null value. Here, $k_{pr} = (k_1 \cdot k_2 \cdot \ldots \cdot k_m)$ and $k_{ex} = (k_1 \oplus k_2 \oplus \ldots \oplus k_n)$. Hence MCF gates themselves are parity preserving without necessity of any modification.

$$I \oplus O = \\ = [k_{ex} \oplus t_1 \oplus t_2] \oplus [k_{ex} \oplus f_1(k_m,T_1) \oplus f_2(k_m,T_2)] \\ = [k_{ex} \oplus t_1 \oplus t_2] \oplus [k_{ex} \oplus (\overline{k_{pr}} \bullet t_1 + k_{pr} \bullet t_2) \\ \oplus (k_{pr} \bullet t_1 + \overline{k_{pr}} \bullet t_2)] = [t_1 \oplus t_2] \oplus [t_1 \oplus t_2] = 0 \quad \ldots (4)$$

The method of designing reversible circuit is based on the selection of gates and the output function to be generated. The output function is first separated into small logical expressions, the selection of gates and their placement will be done in accordance with them. The proposed model is depicted in Fig. 2. MCT gates are placed in two folds following the methodology considered in Fig. 1(b). If a NOT gate is placed on a wire, another NOT gate will be situated at any place on the wires except the position assigned for previous gate, as shown in block B_1. For an MCT gate, the second gate of the same size is placed whose control inputs will be fixed as that of the previous gate and the target input will be situated at any place on the wires of the circuit except the position assigned to the previous gate, as pictured in block B_2. If the target input of second gate does not find any expression to build, it can take a constant input wire to create the same. MCF gates are placed in singles and can take any position on the wires, as shown in block B_l.

Preposition 1: The proposed design methodology produces testable circuits.

Proof: Consider a circuit containing B_l blocks as shown in Fig. 2. Since, the intermediate stages are parity preserving, the exor of previous stage (P_s) and next stage (M_s), $P_s \oplus M_s = 0$; $\forall B = (B_1 \text{ to } B_l)$. Hence, the circuit produced will be parity preserving which provide the ease of testing without excess costs.

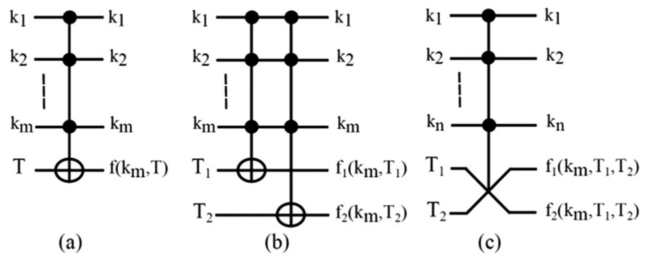

Fig. 1 — Quantum representation (a) MCT gate, (b) equivalent DFT and (c) MCF gate.

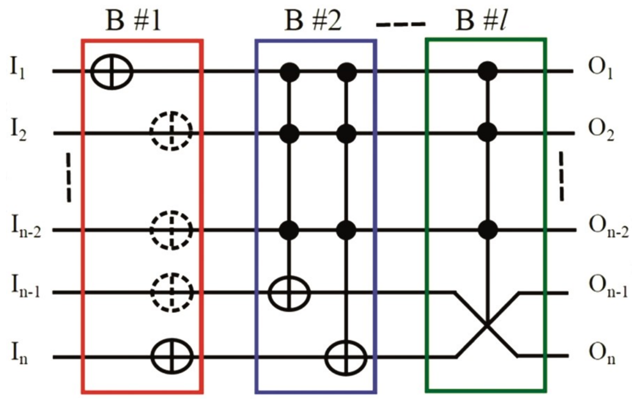

Fig. 2 — Proposed DFT model.

2.2 Fault Detection

Fault detection can be achieved by generating input and output parity of the circuit. For *n* wire reversible circuit, the parity checking can be done by using *n* CNOT gates from a wire of the circuit to a new wire (T_{in}) before and after the whole circuit[10] as shown in Fig. 3. The output of this new wire is called as test output (T_{out}) which can be given by Eq. (5), where $\sum$ denotes *xor* sum of inputs and outputs.

$$T_{out} = \left(\sum_{i=1}^{n} I_i\right) \oplus \left(\sum_{i=1}^{n} O_i\right) \oplus T_{in} \qquad \dots (5)$$

Preposition 2: For T_{in} = 0, Tout will flips from logic 0 to 1 when any single bit fault occurred in the circuit.

Proof: Consider a circuit containing shown in Fig. 2. For a fault free circuit, T_{out} = 0 when T_{in} = 0. Consider the occurrence of single bit fault at any level of the circuit. Since, each block of the circuit is parity preserving, the same parity information will be transferred to next level. Hence, the values at the output will be inverted in odd numbers. Considering $O_1 \rightarrow \bar{O}_1$, $T_{out} = [(I_1 \oplus I_2 \oplus ... \oplus I_n) \oplus (\bar{O}_1 \oplus O_2 \oplus ... \oplus O_n)] = 1$. Finally, the fault occurrence can be detected.

3 Algorithm

When parity preserved architecture are used with an arbitrary design For given input variables V_i and output function V_o, the Algorithm 1 produces testable circuit C_T. V_o is divided into small expressions f_i to form a function list *F*. Initially, number of gates (*N*) are 0 and wires (*n*) are taken equal to maximum length of V_i and V_o. The gates placement operation (Ψ) is done in accordance with f_i. For ESOP based expression, the operation of placement of an MCT gate (Ψ_m) is done twice. For inversion operation, the operation of placement of a NOT gate (Ψ_i) is done twice. The target input of second gates in these operations can take any position on the wires except, the position assigned for previous gates. If the second gate not found suitable wire to obtain f_i, it may take a constant input wire, which is defined as the termination criteria *T*. For other logical operations, an MCF gate is placed at any positions on the wires (Ψ_f). After each cycle, C_T, *N* and *n* are updated to obtain the properties of resulted circuit. The final circuit is then incorporated with parity checker (Φ) for the detection of faults.

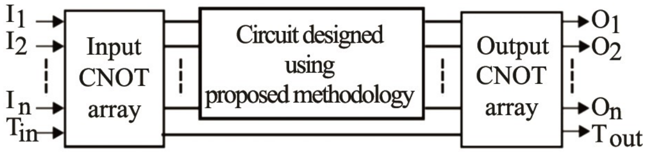

Fig. 3 — Schematic diagram for fault detection.

Algorithm 1 — Operational Summary
```
1:  Input: $V_i$ and V$_o$
2:  Output: Testable circuit ($C_T$)
3:  Define termination criteria $T$
4:  Initialize $n = max(V_i.length, V_o.length)$,
    $C_T = 0, F = V_o, N = 0$
5:  do
6:   Perform operation $\Psi$
7:   Select function $f_i$ from $F$ corresponding to $\Psi$
8:   if $f_i$ is ESOP function then
9:   Perform operation $\Psi_m$
10:  Repeat operation $\Psi_m$ until $T$
11:  if T $\notin F_i$ then
12:  $n = n + 1$
13:  end if
14:  else
15:  if $f_i$ is inversion function then
16:  Perform operation $\Psi_i$ twice
17:  Repeat operation $\Psi_i$ until $T$
18:  if $T \notin F_i$ then
19:  $n = n + 1$
20:  end if
21:  else
22:  Perform $\Psi_f$
23:  end if
24:  end if
25:  Update $C_T$, $N$ and $n$
26: while $|F| \neq 0$
27: $n = n + 1$
28: Perform operation $\Phi$
29: Update $C_T$, $N$ and $n$
```

4 Performance Evaluation

The experiments are performed on Intel Core I7-4790, 3.60 GHz clock with 4 GB memory. A set of benchmark circuit[13] are synthesized in accordance with the proposed methodology using transformation based synthesis on the top of Revkit[14]. The results in terms of wires (*n*), gates (*N*), T-depth (*TD*), T- Count (*TC*), logical qubits (*q*) and garbage output (*G*) of the circuit are demonstrated[15] in Table 1.

These circuits are also formulated in accordance with the existing work related to the present work on

Table 1 — Implementation results.

aCircuit	Proposed work					
	n	N	TD	TC	q	G
xor5	6	4	0	0	6	0
4mod5	6	8	12	28	6	4
3_17	4	8	12	28	4	0
rd32	5	9	18	42	5	2
c17	8	14	129	301	8	0
majority	7	14	216	504	7	5
ex2	7	18	234	546	7	5
con1	10	21	255	595	10	7
5mod5	7	22	171	399	7	5
hwb4	5	26	108	252	5	0
nthprime4	5	27	132	308	5	0
cm82	9	28	186	434	9	5
4_49	5	30	159	371	5	0
2of5	8	33	330	770	8	6
rd53	9	33	255	595	9	5
mod5adder	7	51	813	1897	7	0
hwb5	6	75	711	1659	6	0
hwb6	7	165	2166	5054	7	0
ham7	8	269	7752	18088	8	0
hwb7	8	393	8880	20720	8	0

Table 2 — Comparison of results with existing method.

Circuit	ETG approach [8]			MCT approach [10]		
	N	TD	TC	N	TD	TC
xor5	48	546	1274	4	0	0
4mod5	82	885	2065	8	12	28
3_17	9	12	28	12	12	28
rd32	26	117	273	9	18	42
c17	193	3366	7854	22	180	420
majority	184	3402	7938	26	432	1008
ex2	176	3312	7728	34	360	840
con1	121	2910	6790	43	396	924
5mod5	226	3885	9065	38	261	609
hwb4	28	114	266	29	111	259
nthprime4	55	507	1183	27	132	308
cm82	173	3162	7378	34	165	385
4_49	44	267	623	35	159	371
2of5	154	1975	12275	46	222	518
rd53	212	3255	7595	48	228	532
mod5adder	51	813	1897	56	837	1953
hwb5	71	651	1519	79	723	1687
hwb6	163	2352	5488	171	2229	5201
ham7	166	4230	9870	276	7932	18508
hwb7	402	8565	19985	401	9018	21042

the same platform. The work presented in Nayeem *et al.*[8] exploited the principle of modification using extended Toffoli gates (ETG) and the other utilizes only MCT gates for designing testable reversible circuits[10]. The implementation results for these methods are given in Table 2. All the operating costs, except *q* and *G*, are added for the existing and the proposed work separately and compared. It is calculated that the proposed work achieved a reduction in operating cost by 51%, as compared to previous work under design for testability for single bit fault detection.

5 Conclusions

This paper presents a new method of designing reversible logic circuits incorporated with inbuilt testability feature. The methodology produces parity preserving circuits which can be tested by checking the input and output parity by means of CNOT gates on an extra wire. The requirements of extra hardware and time to modify an original circuit into its testable form can be eliminated. The results correspond to a large reduction in the principle cost and design complexity. Hence, the method provides solutions to both the problem of designing and testability of reversible circuits. MCF gates based synthesis algorithm will be developed for the expansion of the work in near future.

References

1 Bennett C H, *IBM J Res Develop*, 17 (1973) 525.
2 Takeuchi N, Yamanashi Y & Yoshikawa N, *Nature*, 4 (2014) 6354.
3 Vandersypen L M K, Steffen M & Breyta G, *Nature*, 414 (2001) 883.
4 Gaur H M, Singh A K & Ghanekar U, *Proc Comput Sci*, 70 (2015) 384.
5 Vasudevan D P, Lala P K, Di J & Parkerson J P, *IEEE Trans Instrum Meas*, 55 (2006) 406.
6 Thapliyal H & Vinod A P, *Proc IEEE Int Symp Circuits Syst*, (2007) 1085.
7 Mahammad S K N & Veezhinathan K, *IEEE Trans Instr Meas*, 59 (2010) 101.
8 Nayeem N M & Rice J E, *J Elec Test*, 29 (2013) 763.
9 Gaur H M, Singh A K & Ghanekar U, *Int J Light Electron Opt*, 127 (2016) 10593.
10 Gaur H M & Singh A K, *IET Electron Lett*, 52 (2016) 1102.
11 iNEMI-Roadmap executive summary highlights: International electronics manufacturing initiative, 2015.
12 Parhami B, *Fortieth Asilomar Conference on Signals, Systems and Computers,* (2006), 1726.
13 Maslov D, Dueck G & Scott N, Reversible Logic Benchmark Page, 2004, [http://www.cs.uvic. ca/~dmaslov/]
14 Soeken M, Frehse S, Wille R & Drechsler R, Rev Kit: A Toolkit for Reversible Circuit Design, 2017. [http://www.informatik. uni-bremen.de/revkit/].
15 Abdessaied N, *Proc IEEE Int Conf*, VLSI, (2015) 286.