2010

# Fault Testing Quantum Switching Circuits

Marek Perkowski
*Portland State University*, marek.perkowski@pdx.edu

Jacob Biamonte

# Fault testing quantum switching circuits

Jacob D Biamonte and Marek Perkowski*

January 1, 2014

**Abstract**

Test pattern generation is an electronic design automation tool that attempts to find an input (or test) sequence that, when applied to a digital circuit, enables one to distinguish between the correct circuit behavior and the faulty behavior caused by particular faults. The effectiveness of this classical method is measured by the fault coverage achieved for the fault model and the number of generated vectors, which should be directly proportional to test application time. This work address the quantum process validation problem by considering the quantum mechanical adaptation of test pattern generation methods used to test classical circuits. We found that quantum mechanics allows one to execute multiple test vectors concurrently, making each gate realized in the process act on a complete set of characteristic states in space/time complexity that breaks classical testability lower bounds.

## 1 Introduction

Classically, both the fault models considered and the test inputs are localized (not entangled). Form this fact and other reasons, it is not immediately apparent how methods developed in classical test theory could be put to use when testing quantum mechanical switching networks. Here ones views a quantum circuit as being a description of the actions on, and the interactions between qubits. To avoid brute force testing, a method to test these time dependent connections as well as the logical operation of each gate realized in a process is presented. This is done by considering a logical set of failure models designed to drive a switching network to it's bounds of operation and assure main aspects of individual gate functionality.

It is theoretically interesting to combine classical test theory with quantum effects such as entanglement. For instance, quantum states can be designed to execute multiple test vectors concurrently. In addition quantum systems are reversible and any reversible system preserves information. This means that a reversible system preserves the probability that additional information may be present [9]. The additional information present could be used to detect the presence of a fault. In a faulty reversible circuit, the probability of detection for quantum fault $f$ observable with $\hat{A}$ is solely related to the probability of $f$'s presence (see § 2). For instance, one may develop a test set, consisting of input vectors and corresponding observables, that separates a circuit from all considered faults. Based on the information conservative properties of reversible systems [9], this test set can be applied multiple times to detect the probabilistic version of the considered faults.

We consider extending methods developed to test classical circuits, and so present tests designed to separate an oracle from one containing a given set of considered faults [7]. Classically, the testability of the circuit class comprising the oracle has already received much attention after the 1972 paper by Sudhakar M. Reddy [2]. This paper presents a quantum mechanical generalization of this and several other classical methods [1, 2, 10].

**Structure of the paper:** Sec. 1.1 gives an introduction to oracle construction. Sec. 2 discusses the quantum fault models used in this study. The Quantum Test Algorithm is presented in Sec. 3 followed by the conclusion in Sec. 4.

*JDB and MP are with Portland State University, Portland, Oregon 97201, USA. JDB present address: Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, OX1 3QD, UK.

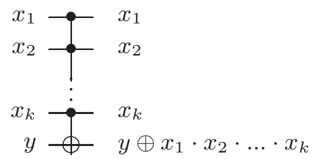


Figure 1: $k-$CN Gate Realizing $y \oplus x_1 \cdot x_2 \cdot ... \cdot x_k$ on the $(k+1)^{th}$ qubit.

### 1.1 Constructing Quantum Oracle Search Spaces

Any Boolean equation may be uniquely expanded to the Positive Polarity Reed-Muller Form (PPRM) [2] as:

$$\begin{aligned} f(x_1, x_2, ..., x_k) = c_0 \oplus c_1 x_1^{\sigma_1} \oplus c_2 x_2^{\sigma_2} \oplus \cdots \oplus c_n x_n^{\sigma_n} \oplus \\ c_{n+1} x_1^{\sigma_1} x_n^{\sigma_n} \oplus \cdots \oplus c_{2k-1} x_1^{\sigma_1} x_2^{\sigma_2}, ..., x_k^{\sigma_k}, \end{aligned} \tag{1}$$

where selection variable $\sigma_i \in \{0, 1\}$, literal $x_i^{\sigma_i}$ represents a variable or its negation and any $c$ term labeled $c_0$ through $c_j$ is a binary constant 0 or 1.

*Example:*

$$\begin{aligned} f(x_1, x_2, x_3, x_4) = 1 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_3 x_4 \oplus x_1 x_3 x_4 \oplus \\ x_1 x_2 x_3 \oplus x_2 x_3 x_4 \end{aligned} \tag{2}$$

Each term in the expansion of Eqn. 2 is called a product term [8], and each variable $x_i$ a literal. For example, $x_3 \cdot x_4$ is a product term, with literals $x_3$ and $x_4$ (constant 1 is not considered to be a product term). Each product term for a given PPRM expansion is realized by an arbitrary quantum controlled-NOT gate ($k-$CN) given in Fig. 1. Repeating this procedure for each product term in Eqn. 2 and sequencing the gates leads to the network realization given in Fig. 2. Above each gate is the label $p_i$, $p$ refers to a product term in the expansion of Eqn. 2, and $i$ the index used to label the seven products. This example will be used again so it is stated explicitly that $p_0$ corresponds to $x_1$, $p_1$ to $x_2$, $p_3$ to $x_3$, $p_3$ to $x_3 x_4$, $p_4$ to $x_1 x_3 x_4$, $p_5$ to $x_1 x_2 x_3$ and finally $p_6$ to $x_2 x_3 x_4$.

In many quantum algorithms, after Boolean function $f$ is constructed by means of a $k-$CN network it is *placed* in a black box oracle ($\mathcal{O}$). The bottom $(k+1)^{th}$ bit contains the realization of $f$ to be read at the box's right. The top $k$ inputs to the box begin in state $|0\rangle$ and the $(k+1)^{th}$ input (target) qubit starts in state $|1\rangle$. The Hadamard operation $H^{\otimes(k+1)}$ is applied placing the input query in a superposition of all $2^{(k+1)}$ classical states. Generally the black box takes as input:

$$H^{\otimes(k+1)} : |0\rangle^{\otimes k} \otimes |1\rangle \longrightarrow (|0\rangle + |1\rangle)^{\otimes k} \otimes (|0\rangle - |1\rangle) \tag{3}$$

Inside the black box all of the targets act on state $|-\rangle$ (an eigenvector of the $k-$CN gate) and the top $k$ qubits remain in a superposition. The true *minterms* are inputs that make a Boolean function evaluate to 1 where false *minterms* evaluate to 0. Each term in the superposition on the top $k$ bits representing a true minterm in the switching function $f$ realized in the oracle will be appended with a negative (relative) phase. The phase of states that do not represent true minterms are left invariant. This is seen by examining the truth table from Fig. 3. The action of an oracle $\mathcal{O}$,[1] realizing a binary function $f(x_1, x_2, ..., x_k)$, is represented by the transform:

$$\mathcal{O} : |k\rangle \otimes |-\rangle \longrightarrow (-1)^{f(k)} |k\rangle \otimes |-\rangle . \tag{4}$$

## 2 Gate Level Quantum Fault Models

Consider the single stage circuit shown in Fig. 4. The numbered locations of possible gate external faults are illustrated by placing an "×" on the line representing a qubits time traversal and here, the gate, initial states ($|i_0\rangle, |i_1\rangle, |i_2\rangle$) and measurements ($m_0, m_1, m_2$) may also contain errors.

**Definition 1** *Error/Fault Location: The wire locations between stages as well as any node, gate initial state or measurement in a given network (see Fig. 4).*

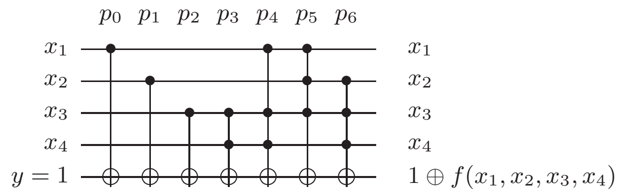


Figure 2: Quantum Network Realization of Eqn. 2 built from arbitrary $k$-CN gates as shown in Fig. 1. The truth table of this oracle is given in Fig. 3.

| *phase* | *state* | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $f$ |
|---|---|---|---|---|---|---|
| + | $\|0000\rangle$ | 0 | 0 | 0 | 0 | 0 |
| + | $\|0001\rangle$ | 0 | 0 | 0 | 1 | 0 |
| − | $\|0010\rangle$ | 0 | 0 | 1 | 0 | 1 |
| + | $\|0011\rangle$ | 0 | 0 | 1 | 1 | 0 |
| − | $\|0100\rangle$ | 0 | 1 | 0 | 0 | 1 |
| − | $\|0101\rangle$ | 0 | 1 | 0 | 1 | 1 |
| + | $\|0110\rangle$ | 0 | 1 | 1 | 0 | 0 |
| + | $\|0111\rangle$ | 0 | 1 | 1 | 1 | 0 |
| − | $\|1000\rangle$ | 1 | 0 | 0 | 0 | 1 |
| − | $\|1001\rangle$ | 1 | 0 | 0 | 1 | 1 |
| + | $\|1010\rangle$ | 1 | 0 | 1 | 0 | 0 |
| + | $\|1011\rangle$ | 1 | 0 | 1 | 1 | 0 |
| + | $\|1100\rangle$ | 1 | 1 | 0 | 0 | 0 |
| + | $\|1101\rangle$ | 1 | 1 | 0 | 1 | 0 |
| + | $\|1110\rangle$ | 1 | 1 | 1 | 0 | 0 |
| − | $\|1111\rangle$ | 1 | 1 | 1 | 1 | 1 |

Figure 3: Oracle Truth Table for Eqn. 2 implemented by the network in Fig. 2: Boolean function $f$ is implemented quantum mechanically. Each of the $2^k$ terms in a superposition input that evaluate to *logic-one* will be marked with a negative phase (also shown in Eqn. 12, in Sec. 3).

**Definition 2** *Quantum Single Fault Model: For simplification the "quantum single fault model" is assumed in this work. In the single fault model, test plans are optimized for all considered faults assuming that only a single failure perturbs the quantum circuit exclusively. Multiple faults will accumulate and be detected, but the single fault model makes it much easier to develop test plans.*

**Conjecture 1** *A test set designed to detect all considered single errors will detect and sample the accumulated impact of multiple errors at multiple locations.*

The following definitions are used to define some of the fault types considered in this work. Complete fault coverage occurs after a test set has determined that the considered fault(s) are not present in a given circuit.

**Definition 3** *Pauli Fault Model: The addition of an unwanted Pauli matrix in a quantum network, at any error location and with placement probability $p$. The Pauli matrices are given in Eqn. 5, 6 and 7.*

$$\sigma_x = |1\rangle\langle 0| + |0\rangle\langle 1| \tag{5}$$

$$\sigma_y = i\,|0\rangle\langle 1| - i\,|1\rangle\langle 0| \tag{6}$$

$$\sigma_z = |0\rangle\langle 0| - |1\rangle\langle 1| \tag{7}$$

**Definition 4** *Initialization Error: A qubit that statistically favors correct preparation in one basis state over the other.*

**Definition 5** *Measurement Fault Model: A single functional measurement gate is replaced with a faulty measurement gate that statistically favors returning* logic-zero *or a* logic-one.

[1] $\mathcal{O}$ is sometimes represented as, $\sum_x (-1)^{f(x)} |x\rangle\langle x|$.

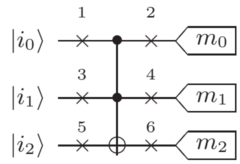


Figure 4: $2-$CN gate with error locations.

**Quantum Test Requirement 1** *A bit flip ($\sigma_x$ or $\sigma_y$) at any error location must be detectable.* ■

**Quantum Test Requirement 2** *A phase flip ($\sigma_z$ or $\sigma_y$) at any error location must be detectable.* ■

**Quantum Test Requirement 3** *Each qubit must be initialized in both basis states $|0\rangle$ and $|1\rangle$.* ■

**Quantum Test Requirement 4** *With the target acting on state $|-\rangle$: Each gate must be shown to attach a relative phase to arbitrary activating state $|a\rangle$ with both positive and negative eigenvalues. Furthermore, each gate must be shown not to attach a relative phase to arbitrary non-activating state $|n\rangle$ with both positive and negative eigenvalues. The target state must remain globally invariant under both $|a\rangle$ and $|n\rangle$.* ■

**Quantum Test Requirement 5** *With the target acting on state $|+\rangle$: relative phase must be shown not to change under arbitrary activating state $|a\rangle$ with both positive and negative eigenvalues. Furthermore, relative phase must not change under arbitrary non-activating state $|n\rangle$ with both positive and negative eigenvalues.* ■

**Quantum Test Requirement 6** *For the target acting separately on basis state $|0\rangle$ and $|1\rangle$: All controls in a gate must be activated concurrently. Furthermore, each control must be addressed with a non-activating state.* ■

**Quantum Test Requirement 7** *Each target must separately act on basis state inputs $|0\rangle$ and $|1\rangle$.* ■

**Quantum Test Requirement 8** *Each qubit must be measured in both* logic-zero *and* logic-one *states.* ■

## 2.1 Conclusions based on the Gate Level Fault Models

In practice, the choice of the fault model will be determined by a particular quantum circuit technology, as well as how the circuit will be used. In this work the functional use of $k-$CN networks are oracle search spaces. In this setting, any $k-$CN gate exhibits twelve, functionally distinct actions.

**Theorem 1** *A quantum $k-$CN gate is capable of four characteristic classical operations. (By characteristic it is meant that all other operations are variants of this basic set.)*

**Proof 1** *The gate is able to act on a $|0\rangle$ and a $|1\rangle$ state when all controls are set. The two remaining functions are simply to act on $|0\rangle$ and $|1\rangle$ when one or more control(s) is addressed with a non-activating state. There are $2^k-1$ input states that do not activate the gate, but these inputs all probe the* off *function.*

*Similarly, each control has two logical functions. The first is to be addressed with a logical $|0\rangle$ and the second is to be addressed with a $|1\rangle$.* ■

$$\begin{array}{lcccccc} v_0 \rightarrow & 0 & 0 & 0 & 0 & \cdots & 1 \\ v_1 \rightarrow & 0 & 0 & 0 & 0 & \cdots & 0 \\ v_2 \rightarrow & 1 & 1 & 1 & 1 & \cdots & 1 \\ v_3 \rightarrow & 1 & 1 & 1 & 1 & \cdots & 0 \end{array}$$

Figure 5: Classical test vectors ($v_0$, $v_1$, $v_2$, $v_3$) acting on binary basis vectors $\{0, 1\}$ with the gate first off $(v_0, v_1)$ and then on $(v_2, v_3)$. The rightmost bit in the figure is applied to the $(k+1)^{th}$ bit.

| *Minterm* | *Target State* | *Minterm* | *Target State* |
|---|---|---|---|
| $e^{+i\phi}\|true\rangle$ | $(\|0\rangle + e^{+i\varphi}\|1\rangle)$ | $e^{+i(\phi+\varphi)}\|true\rangle$ | $(\|0\rangle + e^{+i\varphi}\|1\rangle)$ |
| $e^{-i\phi}\|true\rangle$ | $(\|0\rangle + e^{+i\varphi}\|1\rangle)$ | $e^{-i(\phi-\varphi)}\|true\rangle$ | $(\|0\rangle + e^{+i\varphi}\|1\rangle)$ |
| $e^{+i\phi}\|false\rangle$ | $(\|0\rangle + e^{+i\varphi}\|1\rangle)$ | $e^{+i(\phi)}\|false\rangle$ | $(\|0\rangle + e^{+i\varphi}\|1\rangle)$ |
| $e^{-i\phi}\|false\rangle$ | $(\|0\rangle + e^{+i\varphi}\|1\rangle)$ | $e^{-i(\phi)}\|false\rangle$ | $(\|0\rangle + e^{+i\varphi}\|1\rangle)$ |
| $e^{+i\phi}\|true\rangle$ | $(\|0\rangle + e^{-i\varphi}\|1\rangle)$ | $e^{+i(\phi-\varphi)}\|true\rangle$ | $(\|0\rangle + e^{-i\varphi}\|1\rangle)$ |
| $e^{-i\phi}\|true\rangle$ | $(\|0\rangle + e^{-i\varphi}\|1\rangle)$ | $e^{-i(\phi+\varphi)}\|true\rangle$ | $(\|0\rangle + e^{-i\varphi}\|1\rangle)$ |
| $e^{+i\phi}\|false\rangle$ | $(\|0\rangle + e^{-i\varphi}\|1\rangle)$ | $e^{+i\phi}\|false\rangle$ | $(\|0\rangle + e^{-i\varphi}\|1\rangle)$ |
| $e^{-i\phi}\|false\rangle$ | $(\|0\rangle + e^{-i\varphi}\|1\rangle)$ | $e^{-i\phi}\|false\rangle$ | $(\|0\rangle + e^{-i\varphi}\|1\rangle)$ |

Figure 6: A $k-$CN Gate Truth Table (Case: 2 top, Case: 1 bottom): Illustrating all of the different possible gate actions for orthogonal setting of variables $\phi$ and $\varphi$. A $|true\rangle$ minterm activates the gate, any $|false\rangle$ minterm does not.

Provided the state of the top $k$ bits is some equal superposition and the target of the gate acts on a state with the following form: $|0\rangle + e^{\pm i\varphi}|1\rangle$. Under this condition, the inputs to a $k-$CN gate are expressed as:

$$|\psi_{in}\rangle \longrightarrow \left[\sum_{x\,=\,0}^{2^k-1} w_x\,|x\rangle\right] \otimes (|0\rangle + e^{\pm i\varphi}|1\rangle), \tag{8}$$

where $w_x = e^{\pm i\phi}$. Similarly, as in the case of Theorem 1, certain operations define the gate's function.

The arbitrary quantum superposition state defined in Eqn. 8 allows one to consider each input as a separate state. In the column denoted minterm from Fig. 6, $|true\rangle$ minterms activate the gate while $|false\rangle$ terms do not. Under this consideration the following holds:

**Theorem 2** *A $k-$CN gate is capable of eight characteristic quantum operations.* (*We consider quantum operations as those that manipulate quantum phase and non-classical superposition states; characteristic has the same meaning as in Theorem 1.*)

**Proof 2** *The proof is constructive:*

Case 1: *When activated, quantum gates exhibit phase kickback when the state of the target is $|0\rangle + e^{-i\varphi}|1\rangle$. The activating state can have a phase of $+w_x$ or $-w_x$. Furthermore, a non-activating state can have a phase of $+w_x$ or $-w_x$ and of course, nothing should happen when acted on by the $k-$CN gate.*

Case 2: *(The opposite of Case 1.) The alternative case is that the target acts on state $|0\rangle + e^{+i\varphi}|1\rangle$. As before, the activating and non-activating states can have phases of $+w_x$ or $-w_x$. Nothing should happen under the case of both an activating and a non-activating state. This functionality is probed in four additional tests.*

*We draw the readers attention now to the table in Fig. 6 for the illustration of Case* 1 *and Case* 2*. Variables $\phi$ and $\varphi$ are set to create states that are operated on by the $k-$CN gate, these are the combinations of actions considered. The Proof is concluded by mentioning that, all the quantum functions of the $k-$CN gate represent one variant of these eight cases when used in a phase oracle.* ■

Thus according to Theorems 1 and 2 in total we need $4 + 8 = 12$ non-entangled tests to identify the function of any $k-$CN gate.

# 3 The Fault Detection Algorithm

Tests $T_1$, $T_2$, $T_5$ and $T_6$ verify all classical degrees of freedom. Tests $T_3$ and $T_4$ verify the phase kickback features of the oracle. As a proof of concept the introduced method holds the test set size to constant six, increasing the complexity of added stages for tests $T_3$ and $T_4$. This approach helps better tie classical ideas with quantum test set generation. This is due to the fact that classically, circuits realizing linear functions are easy to test due to their high level of controllability.

**Definition 6** *Quantum Build In Self Test Circuit (*QBIST*): A quantum circuit designed to test a second quantum circuit; the quantum circuit under test (*QCUT*). A* QBIST *circuit may be built at the input and/or output terminals of the* QCUT*, and the* QBIST *stage is always assumed to contain no errors.*

Consider the example circuit presented in Fig. 2. The analysis given in the coming subsections begins by generating an input state that turns all the gates in the network *on* and *off* concurrently. Denote these tests as $T_1$ and $T_2$, and their general form on a $k$ variable function follows:

$T_1$**:** $(|0\rangle^{\otimes k} + |1\rangle^{\otimes k}) \otimes |0\rangle$

$T_2$**:** $(|0\rangle^{\otimes k} - |1\rangle^{\otimes k}) \otimes |1\rangle$

The classical equivalent of tests $T_1$ and $T_2$ was given in Fig. 5 (where $T_1$ corresponded to vectors $v_0$ and $v_2$, and $T_2$ corresponded to both $v_1$ and $v_3$). Together tests $T_1$ and $T_2$ will be shown to satisfy Requirements 1, 3, 6, 7 and 8 in Sec. 3.1 and 3.2.

Sec. 3.3 considers tests $T_5$ and $T_6$. These tests are shown to satisfy Requirement 5 by using the following states as oracle inputs: $|+\rangle^{\otimes k} \otimes |+\rangle$ and $|-\rangle^{\otimes k} \otimes |+\rangle$. In both tests, the state at the controls will not impact the state at the target, leaving all qubits—ideally—unchanged (since no net entanglement is generated).

Sec. 3.4 and 3.5 investigate the ability of the network to both attach a relative phase to each activating term in the superposition and to leave non-activating states unaltered. This in general is a complex procedure, that in the first case can be done in two tests denoted as $T_3$ and $T_4$. Test $T_3$ utilizes state $|+\rangle^{\otimes k} \otimes |-\rangle$ and test $T_4$ utilizes state $|-\rangle^{\otimes k} \otimes |-\rangle$ as input to the oracle. However, additional "design-for-test" stages must be added to the end of the circuit, thereby leading to a deterministic measurement. Tests $T_3$ and $T_4$ are shown to satisfy Requirement 4.

### 3.1 Test $T_1$: $(|0\rangle^{\otimes k} + |1\rangle^{\otimes k}) \otimes |0\rangle$

In test $T_1$, all qubits are initialized as: $|0000\rangle \otimes |0\rangle$. The action of the first $QBIST_{11}$ stage (from Fig. 8) creates the following oracle input state:

$$QBIST_{11} : |0000\rangle \otimes |0\rangle \longrightarrow \left(|0\rangle^{\otimes k} + |1\rangle^{\otimes k}\right) \otimes |0\rangle . \tag{9}$$

The left half of the entangled test sequence is $|0000\rangle \otimes |0\rangle$. It is clear that for a "*gold circuit*" not one gate turns on, and the target qubit will be left untouched. For the right half of the entangled test vector, each gate in the circuit turns on, and this cycles the $(k+1)^{th}$ qubit initially starting in $|0\rangle$ back and forth between basis states. The state of the last qubit after the oracle is $|0\rangle$.[2] The purpose of $QBIST_{12}$ is simply to remove the phase induced entanglement experienced on the top $k$ qubits. The intermittent states at each stage of the circuit under test $T_1$ are shown in Fig. 7. The final step in the $QBIST_{12}$ circuit applies a Hadamard gate to the top qubit, resulting back in the starting state, $|0000\rangle \otimes |0\rangle$, thereby completing test $T_1$. The complexity of the added CN and H gates needed for test $T_1$ is $2(k-1)$CN+2H.

| Stage | Action of Stage |
|---|---|
| $in \longrightarrow$ | $\|0000\rangle \otimes \|0\rangle$ |
| $QBIST_{11} \longrightarrow$ | $(\|0000\rangle + \|1111\rangle) \otimes \|0\rangle$ |
| $p_0 \longrightarrow$ | $\|0000\rangle \|0\rangle + \|1111\rangle \|1\rangle$ |
| $p_1 \longrightarrow$ | $\|0000\rangle \|0\rangle + \|1111\rangle \|0\rangle$ |
| $p_2 \longrightarrow$ | $\|0000\rangle \|0\rangle + \|1111\rangle \|1\rangle$ |
| $p_3 \longrightarrow$ | $\|0000\rangle \|0\rangle + \|1111\rangle \|0\rangle$ |
| $p_4 \longrightarrow$ | $\|0000\rangle \|0\rangle + \|1111\rangle \|1\rangle$ |
| $p_5 \longrightarrow$ | $\|0000\rangle \|0\rangle + \|1111\rangle \|0\rangle$ |
| $p_6 \longrightarrow$ | $\|0000\rangle \|0\rangle + \|1111\rangle \|1\rangle$ |
| $QBIST_{12} \longrightarrow$ | $\|0000\rangle \otimes \|0\rangle$ |

Figure 7: $T_1$ test pattern and impact at each gate in the circuit. Gates as labeled left to right $p_1$ to $p_6$.

[2]If an an even number of gates were present a slight modification to the final half of the $QBIST_{12}$ circuit must be made. This modification is the removal of the first CN gate at the start of the $QBIST_{12}$ acting on the $(k+1)^{th}$ qubit and controlled by the $k^{th}$ qubit. In general for an odd number of gates in a quantum network prior to the final $QBIST_{12}$ stage the circuit will be in state $|0\rangle^{\otimes k} |0\rangle \pm |1\rangle^{\otimes k} |1\rangle$. The addition of a $\text{CN}_{k,k+1}$ gate removes unwanted entanglement so that the final qubit will be left in a product state.

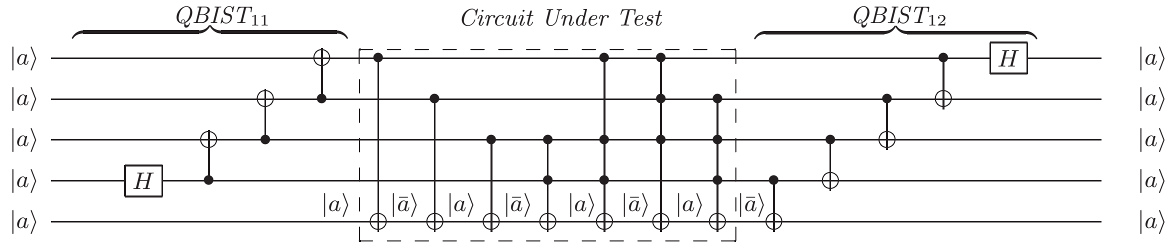


Figure 8: Tests $T_1$ and $T_2$ (GHZ states): In Test $T_1$, $a = 0$ so the circuit starts off in state: $|0000\rangle$. $QBIST_{11}$ maps this state to the oracle's input as: $(|0000\rangle + |1111\rangle) \otimes |0\rangle$. In Test $T_2$, $a = 1$ and the input to the oracle is: $(|0000\rangle - |1111\rangle) \otimes |1\rangle$. $QBIST_{12}$ removes entanglement and returns the system to a product state.

### 3.2 Test $T_2$: $(|0\rangle^{\otimes k} - |1\rangle^{\otimes k}) \otimes |1\rangle$

No physical change is made to the circuit from Fig. 8, however the qubits are now initialized to state $|1111\rangle \otimes |1\rangle$. The outcome is similar to test $T_1$, the bottom qubit is toggled a total of seven times resulting in the final state of $|1\rangle$. (*Each gate that acted on $|0\rangle$ in test $T_1$ now acts on $|1\rangle$ thereby exhaustively probing every classical input combination of each $k-CN$ gate, seen in Fig. 8.*) The $QBIST_{12}$ again disentangles the test responses, resulting back in the initial state of $|1111\rangle \otimes |1\rangle$.

In tests $T_1$ and $T_2$ each node is addressed with both activating and non-activating states. Furthermore, each qubit is initialized and measured in both basis states. Tests $T_1$ and $T_2$ have an added CN and H gate complexity of $4(k-1)$CN+4H. The following Theorems prove which faults have been detected with tests $T_1$ and tests $T_2$ and are general for $n$ bit oracles:

**Theorem 3** *Either test $T_1$ or test $T_2$ will detect $\sigma_x$ and $\sigma_y$ bit flips at any error location, thus satisfying Requirement 1.*

**Proof 3** *Tests $T_1$ and $T_2$ both satisfy Requirement 1. The proof in this section is given for test $T_1$ and is nearly identical to the steps taken for test $T_2$. Consider now test $T_1$:*

Case 1*: The top ($1^{st}$) qubit is flipped:* QBIST$_{12}$ *receives state $(|1\rangle |0\rangle^{\otimes(k-1)} \pm |0\rangle |1\rangle^{\otimes(k-1)})$ as input. After successive applications of $CN_{i-1,i}$ from $i = k$ to $i = 2$ the state will be $(|11\rangle |0\rangle^{\otimes(k-2)} \pm |01\rangle |1\rangle^{\otimes(k-2)}) = (|0\rangle \pm |1\rangle) \otimes |1\rangle \otimes |0\rangle^{\otimes(k-2)}$. Thus, a bit flip impacting the $1^{st}$ bit is detectable on the $2^{nd}$ bit. Given a bit flip impacting any other qubit $q$, $(1 < q \leq k)$* QBIST$_{12}$ *receives $(|0\rangle^{\otimes(q-1)} |a\rangle |0\rangle^{\otimes(k-q)} \pm |1\rangle^{\otimes(q-1)} |\bar{a}\rangle |1\rangle^{\otimes(k-q)})$ as input state. A similar relation holds such that a bit flip on the $(q-1)^{th}$ bit is detectable on the $q^{th}$ and possibly the $1^{st}$ bit if the phase is also inverted. For errors impacting any qubit other than the $1^{st}$, both the $q^{th}$ bit as well as the $(q+1)^{th}$ (impacted bit) will show the error.*

Case 2*: Bottom $(k+1)^{th}$ qubit is flipped: Normally the top $k$ bits and the bottom $(k+1)^{th}$ bits are factorable when entering the final* QBIST$_{12}$ *stage. Assume an even number of gates in the oracle and that instead of state: $(|0\rangle^{\otimes k} + |1\rangle^{\otimes k}) \otimes |0\rangle$ the final* QBIST$_{12}$ *receives the worst case state of $|0\rangle^{\otimes k} |0\rangle + |1\rangle^{\otimes k} \otimes |1\rangle$. The final* QBIST$_{12}$ *will not remove the entanglement associated with the $(k+1)^{th}$ bit. This is detectable based on p, the probability that a bit flip occurred in the computational basis in the first place, satisfying Requirement 1. This is the only fault that, when deterministically present interjects a probabilistic outcome in observability.* ■

**Theorem 4** *Together tests $T_1$ and $T_2$ initialize each qubit in both basis states so that Requirement 3 is satisfied.*

**Proof 4** *In test $T_1$ the initial state of the register is $|0\rangle^{\otimes k} \otimes |0\rangle$ and in test $T_2$ the initial state is $|1\rangle^{\otimes k} \otimes |1\rangle$, therefore Requirement 3 is satisfied.* ■

**Theorem 5** *Taken together tests $T_1$ and $T_2$ activate all controls concurrently and each control is addressed with a non-activating state while the target is separately in basis state $|0\rangle$ and next $|1\rangle$ satisfying Requirement 6.*

**Proof 5** *In tests $T_1$ and $T_2$ the test state prior to application of the oracle is $(|0\rangle^{\otimes k} \pm |1\rangle^{\otimes k}) \otimes |\bar{a}\rangle$. In both tests $T_1$ and $T_2$ the term $|0\rangle^{\otimes k}$ addresses each control with a non-activating state, the term $\pm |1\rangle^{\otimes k}$ activates all gates and in both tests the target is in a basis state. This satisfies Requirement 6.* ■

**Theorem 6** *Taken together tests $T_1$ and $T_2$ force each gate in the circuit to act on both basis states, thereby satisfying Requirement 7.*

**Proof 6** *In both tests $T_1$ and $T_2$ the term $\pm|1\rangle^{\otimes k}$ activates all gates. Each gate in test $T_1$ that received target input state $|a\rangle$ received target input state $|\bar{a}\rangle$ in test $T_2$, thus satisfying Requirement 7.* ■

**Theorem 7** *After executing test $T_1$ and $T_2$ each qubit will be measured in both basis states, thus satisfying Requirement 8.*

**Proof 7** *The result of test $T_1$ is $|0\rangle^{\otimes(k+1)}$ and the measured result pending the success of test $T_2$ is $|1\rangle^{\otimes(k+1)}$ thus satisfying Requirement 8.* ■

### 3.3 Tests $T_5$ and $T_6$: $|+\rangle^{\otimes k} \otimes |+\rangle$ and $|-\rangle^{\otimes k} \otimes |+\rangle$

The two following tests are simple to conceptualize, as seen in Fig. 9 they have an added gate complexity of $4kH$. When $a = 0$ test $T_5$ generates input state $|++++\rangle \otimes |+\rangle$ and when $a = 1$ test $T_6$ generates input state $|----\rangle \otimes |+\rangle$. Since the eigenvalue of the target state is $+1$, no change in relative phase should result from propagation through the quantum circuit and the state of the register should not become entangled. Theorem 8 proves that test $T_5$ combined with test $T_6$ satisfy Requirement 5 with an added gate complexity of $4k$H.

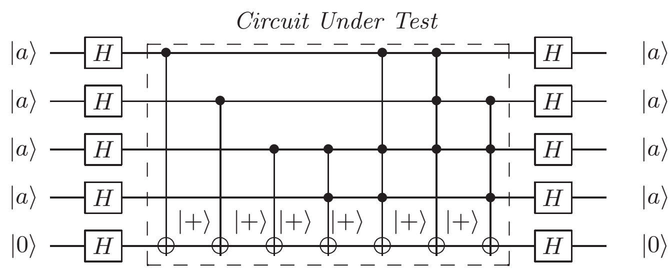


Figure 9: Tests $T_5$ and $T_6$ (Super Tests): Test $|+\rangle^{\otimes k} \otimes |+\rangle$ is first generated ($a = 0$, $T_5$) and next test $|-\rangle^{\otimes k} \otimes |+\rangle$ is applied ($a = 1$, $T_6$). The target of each $k$-CN gate acts on state $|+\rangle$. No entanglement is added in either test, since no relative phase change of individual superposition term(s) will occur.

**Theorem 8** *Together tests $T_5$ and $T_6$ satisfy Requirement 5.*

**Proof 8** *In both tests $T_5$ and $T_6$ the state of the target qubit is $|+\rangle$. Any gate that was activated by a state with an eigenvalue $+1$ in test $T_5$ will be activated by a state with an eigenvalue $-1$ in test $T_6$. Relative phase will not change under arbitrary non-activating and activating states since the target state has an eigenvalue of $+1$, satisfying Requirement 5.* ■

**Theorem 9** *Either one of tests $T_5$ or $T_6$ detects $\sigma_z$ or $\sigma_y$ phase flips and therefore satisfies Requirement 2.*

**Proof 9** *Here the Proof is done considering test $T_5$, however the steps are the same as those needed for test $T_6$. Consider state $|+\rangle^{\otimes k} \otimes |+\rangle$, this is a product state that may be expanded as: $|+\rangle \otimes \cdots \otimes |+\rangle \otimes |+\rangle \otimes |+\rangle \otimes \cdots \otimes |+\rangle$. The state of the target is $|+\rangle$ and therefore phase will not make the state non-local (with an exception of a phase flip on the $(k+1)^{th}$ bit, in that case the bottom bit will deterministically reveal the presence of an error). Given a $\sigma_z$ fault impacting any qubit, the state becomes $|+\rangle \otimes \cdots \otimes |+\rangle \otimes |-\rangle \otimes |+\rangle \otimes \cdots \otimes |+\rangle$. In the final stage of* QBIST$_{52}$ *a Hadamard operation $H^{\otimes(k+1)}$ is applied to the register:*

$$H^{\otimes(k+1)} \cdot |+\rangle \otimes \cdots \otimes |+\rangle \otimes |-\rangle \otimes |+\rangle \otimes \cdots \otimes |+\rangle \longrightarrow |0\rangle \otimes \cdots \otimes |0\rangle \otimes |1\rangle \otimes |0\rangle \otimes \cdots \otimes |0\rangle \,. \tag{10}$$

*Since the $\sigma_z$ bit flip impacts the global state of a qubit, it will be seen as a bit flip in the measured state of $T_5$ satisfying Requirement 2. The proof is concluded mentioning that this result coincides with observations drawn in [7], (Theorem 2, § 4).* ■

| *Requirements* (↓) | *Fault Types Tested* (↓) − *Tests* (→) | $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ | $T_1 \cup T_2$ | $T_3 \cup T_4$ | $T_5 \cup T_6$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Requirement 1 | Any $\sigma_x$ or $\sigma_y$ bit flips occurring? | $\times$ | $\times$ | | | | | $\times$ | | |
| Requirement 2 | Any $\sigma_z$ phase flips occurring? | | $\circ$ | $\circ$ | $\circ$ | $\times$ | $\times$ | $\circ$ | $\circ$ | $\times$ |
| Requirement 3 | Is initialization into $\|0\rangle$ and $\|1\rangle$ O.K.? | $\circ$ | $\circ$ | | | $\circ$ | $\circ$ | $\times$ | $\times$ | |
| Requirement 4 | With $\|-\rangle$ at target is phase kickback O.K.? | | | $\circ$ | $\circ$ | | | | $\times$ | |
| Requirement 5 | Any phase problems with $\|+\rangle$ at the target? | | | | | $\circ$ | $\circ$ | | | $\times$ |
| Requirement 6 | Are the controls activated with $\|0\rangle$ and $\|1\rangle$? | $\circ$ | $\circ$ | | | | | $\times$ | | |
| Requirement 7 | Gate acts on basis $\|0\rangle$ and $\|1\rangle$ O.K.? | $\circ$ | $\circ$ | | | | | $\times$ | | |
| Requirement 8 | Is measurement in $\|0\rangle$ and $\|1\rangle$ O.K.? | $\circ$ | $\circ$ | | | | | $\times$ | | |

Table 1: Tests are depicted in columns $3-11$, fault types in column 2 and Requirements in column 1. A given test (column) with table entry $\times$ below it satisfies the Requirement listed in the row corresponding to that $\times$. Entries with $\circ$ inside correspond to tests that cover some, but not all of the faults depicted in the corresponding row.

The classical degrees of freedom for an oracle have been accounted for in tests $T_1$, $T_2$, $T_5$ and $T_6$ with an added gate complexity of only $4(k+1)$H+$4(k-1)$CN. The phase kickback features of the gates in the oracle are verified next in tests $T_3$ and $T_4$.

The controllability of a circuit represents an ability to propagate a specific input vector through a network, such that it will map a state to a specific fault location. This represents an added challenge in the case of quantum circuits, since inputs will become entangled. However, after a discussion of the upper bounds of tests $T_3$ and $T_4$ in Sec. 3.6 more controllable test input vectors are proposed (Sec. 3.7) replacing the added complexity of these tests with a linear increase in the number of experiments needed.

### 3.4 Test $T_3$: $|+\rangle^{\otimes k} \otimes |-\rangle$

The goal of test $T_3$ is to verify that phase traverses correctly amongst all gates. For test $T_3$ the Hadamard gates at the left of Fig. 10 are used to prepare the following superposition state input on the top $k$ bits:

$$\begin{aligned}&\Longrightarrow |0000\rangle + |0001\rangle + |0010\rangle + |0011\rangle + |0100\rangle + |0101\rangle \\ &+ |0110\rangle + |0111\rangle + |1000\rangle + |1001\rangle + |1010\rangle + |1011\rangle \\ &+ |1100\rangle + |1101\rangle + |1110\rangle + |1111\rangle \end{aligned} \tag{11}$$

Observe that Eqn. 12 is like a truth table where all the true minterms of the function have phase factors of $-1$, (see Fig. 3). This often results in phase induced entanglement as shown in Eqn. 12.

$$\begin{aligned}&\Longrightarrow |0000\rangle + |0001\rangle - |0010\rangle + |0011\rangle - |0100\rangle - |0101\rangle \\ &+ |0110\rangle + |0111\rangle - |1000\rangle - |1001\rangle + |1010\rangle + |1011\rangle \\ &+ |1100\rangle + |1101\rangle + |1110\rangle - |1111\rangle \end{aligned} \tag{12}$$

In general, a product (*local*) superposition state may be written as:

$$\pm \bigotimes_{i=0}^{k-1} (|0\rangle + a_i |1\rangle) \tag{13}$$

where any $a_i$ term is either $+1$ or $-1$. For the state in Eqn. 12 to be expressible as a product state, Eqn. 14 must be satisfied:

$$(|0\rangle + a_0 |1\rangle)(|0\rangle + a_1 |1\rangle)(|0\rangle + a_2 |1\rangle)(|0\rangle + a_3 |1\rangle). \tag{14}$$

Given Eqn. 14, any one of $2^i$ ($0 \le i < k$) possible choices for $a_i$ results in a local description of the quantum state (the implications of which will be discussed in Sec. 3.6). The general expansion of Eqn. 14 leads directly to the generic state:

$$\begin{aligned}&\Longrightarrow |0000\rangle + a_3 |0001\rangle + a_2 |0010\rangle + a_1 |0100\rangle + a_0 |1000\rangle \\ &+ a_0 \cdot a_1 |1100\rangle + a_0 \cdot a_2 |1010\rangle + a_0 \cdot a_3 |1001\rangle \\ &+ a_1 \cdot a_2 |0110\rangle + a_1 \cdot a_3 |0101\rangle + a_2 \cdot a_3 |0011\rangle \\ &+ a_0 \cdot a_1 \cdot a_2 |1110\rangle + a_0 \cdot a_2 \cdot a_3 |1011\rangle \\ &+ a_0 \cdot a_1 \cdot a_3 |1101\rangle + a_1 \cdot a_2 \cdot a_3 |0111\rangle \\ &+ a_0 \cdot a_1 \cdot a_2 \cdot a_3 |1111\rangle \end{aligned} \tag{15}$$

Comparing Eqns. 12 and 15 for the considered circuit, the system of arithmetic equations given in Eqn. 16 is obtained. This system is clearly not specifying a product state since Eqns. 12 and 15 matched with Eqn. 16 are inconsistent. The interfering terms $a_0 \cdot a_1 \cdot a_2$ and $a_2 \cdot a_3$ could be changed for the system to return to a local, product state description. This may be done by inserting the $QBIST_{32}$ circuit given in Fig. 10. $QBIST_{32}$ inverts the phase on terms $|1110\rangle$ and $|0011\rangle$ to $+1$, making the state factorable as $(|0\rangle + |1\rangle)(|0\rangle + |1\rangle)(|0\rangle + |1\rangle)(|0\rangle - |1\rangle) \otimes |-\rangle$.

$$\begin{pmatrix} a_0 = -1 & a_1 \cdot a_3 = +1 \\ a_1 = -1 & a_2 \cdot a_3 = +1 \\ a_2 = -1 & a_0 \cdot a_1 \cdot a_2 = +1 \\ a_3 = +1 & a_0 \cdot a_1 \cdot a_3 = -1 \\ a_0 \cdot a_1 = +1 & a_0 \cdot a_2 \cdot a_3 = +1 \\ a_0 \cdot a_2 = +1 & a_1 \cdot a_2 \cdot a_3 = +1 \\ a_0 \cdot a_3 = -1 & a_0 \cdot a_1 \cdot a_2 \cdot a_3 = -1 \\ a_1 \cdot a_2 = +1 & \forall i, a_i \in \{-1, +1\} \end{pmatrix} \tag{16}$$

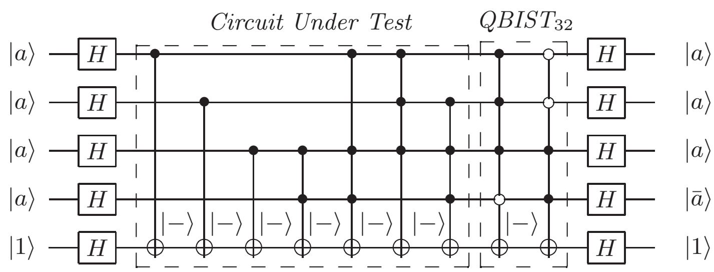


Figure 10: Circuit Under Test $T_3$ and $T_4$: Test $T_3 \longrightarrow |+\rangle^{\otimes k} \otimes |-\rangle$ is first generated ($a = 0$, $T_3$) and next test $T_4 \longrightarrow |-\rangle^{\otimes k} \otimes |-\rangle$ is applied ($a = 1$, $T_4$). Nodes activated with $|0\rangle$ are denoted as ($\circ$). $QBIST_{32}$ removes entanglement returning the system to a product state and has the same form in both tests.

### 3.5 Test $T_4$: $|-\rangle^{\otimes k} \otimes |-\rangle$

Test $T_4$ is an exact dual to test $T_3$ and therefore, the needed $QBIST_{42}$ stage will have the exact same structure as the $QBIST_{32}$ already used. Now the register is initialized into state $|1111\rangle \otimes |1\rangle$ (by setting $a = 1$ in Fig. 10). The Hadamard operators map this initial state as follows:

$$\left(H^{\otimes(k+1)}\right) \cdot (|1111\rangle \otimes |1\rangle) \longrightarrow |- - - -\rangle \otimes |-\rangle \tag{17}$$

and this acts as input to the oracle. The phase of each term is now opposite when compared with $T_3$. $QBIST_{42}$ inverts the phase on term $|1110\rangle$ and $|0011\rangle$ to $-1$, making the state factorable and resulting in this local state description $(|0\rangle - |1\rangle)(|0\rangle - |1\rangle)(|0\rangle - |1\rangle)(|0\rangle + |1\rangle) \otimes |-\rangle$.

Theorem 10 proves that test $T_3$ combined with test $T_4$ satisfy Requirement 4. Tests $T_3$ and $T_4$ have a worst case added gate complexity of at most $\Theta(N - k) + 4kH$, where $\Theta$ is a function of the number of controls needed in the disentanglement stage and the linearity of the oracle.

**Theorem 10** *Together tests $T_3$ and $T_4$ satisfy Requirement 4.*

**Proof 10** *In tests $T_3$ and $T_4$ the state of the target is $|-\rangle$. Any gate that was activated by a state with eigenvalues $\pm 1$ during test $T_3$ is activated by a state with eigenvalues $\mp 1$ in test $T_4$. Furthermore, both tests $T_3$ and $T_4$ contain non-activating terms, each with opposite eigenvalues. Tests $T_3$ and $T_4$ therefore satisfy Requirement 4.* ■

Table 1 provides a concise illustration of the sets of faults entirely covered by given test(s) (denoted by $\times$) as well as the sets of faults partially covered by a given test (denoted by $\circ$). We have developed a quantum test algorithm that probes the logical function of each $k-$CN gate in an oracle. Now upper bounds on the extraction technique ($QBIST_{32}$ circuit stage) will be derived in Sec. 3.6.

### 3.6 Upper Bounds for QBIST$_{32}$:

The concepts of the presented test algorithm are general and therefore work for any circuit. They do however require the successful design of the $QBIST_{32}$. This design varies between oracles and has an upper bound of added depth complexity that depends on the function realized in the oracle.

**Definition 7** *An affine Boolean function $A_f(x_1, ..., x_k)$ , on variables $x_1, ..., x_k$ is any function the takes the form*

$$A_f(x_1, x_2, ..., x_k) = c_0 \oplus c_1 \cdot x_1 \oplus c_2 \cdot x_2 \oplus \cdots \oplus c_k \cdot x_k, \tag{18}$$

*where $\cdot$ is Boolean AND, $\oplus$ is EXOR (modulo 2 addition), $c_i \in \{0, 1\}$ and $i = 0, 1, ..., n$ are indices of coefficients. It is easy to see that there exist $2^{k+1}$ affine functions all of which have checkered cube patterns. A linear function is any one of the $2^k$ affine functions generated when coefficient $c_0 = 0$.*

We present the following theorem (11) relating state separability to the function being realized by a given oracle.

**Theorem 11** *Consider oracle $\mathcal{O}$ for which test $T_3$ obtains only separable (local) measurements (requires no disentanglement). $\mathcal{O}$ necessarily realizes only affine functions over $k$ variables.*

**Proof 11** *The proof is based on the straightforward generalization of the following example:*
*Assume input variables $(x_1, x_2, x_3)$. The expression*

$$\begin{aligned}
&|000\rangle\,(+1) + |001\rangle\,a_2 + |010\rangle\,a_1 + |011\rangle\,a_1 \cdot a_2 + \\
&|100\rangle\,a_0 + |101\rangle\,a_0 \cdot a_2 + |110\rangle\,a_0 \cdot a_1 + \\
&|111\rangle\,a_0 \cdot a_1 \cdot a_2
\end{aligned} \tag{19}$$

*corresponds to a classical truth table with $\prod a_i$ expressions corresponding to sum-of-product canonical coefficients. Assuming the encoding*

$$en(+1) = 0, \quad en(-1) = 1, \tag{20}$$

*arithmetic expressions like $a_1 \cdot a_2$ are changed to Boolean values like $en(a_1) \oplus en(a_2)$. Normally one would consider the case that $b_0 = 0$ for linear functions. Because of global phase $b_0$ may take either binary value corresponding to all affine functions on $k$ variables. It is well known from the canonical SOP to PPRM conversion method that $PPRM = b_0 \cdot 1 \oplus (b_0 \oplus b_1) \cdot x_3 \oplus (b_0 \oplus b_2) \cdot x_2 \oplus (b_0 \oplus b_1 \oplus b_2 \oplus b_3) \cdot x_2 \cdot x_3 \oplus (b_0 \oplus b_2 \oplus b_4 \oplus b_6) \cdot x_1 \cdot x_2 \oplus (b_0 \oplus b_4) \cdot x_1 \oplus (b_0 \oplus b_1 \oplus b_4 \oplus b_5) \cdot x_1 \cdot x_3 \oplus (b_0 \oplus b_1 \oplus b_2 \oplus b_3 \oplus b_4 \oplus b_5 \oplus b_6 \oplus b_7) \cdot x_1 \cdot x_2 \cdot x_3$, where $b_i$ are coefficients of minterms, i.e. $b_0$ is a coefficient of $|000\rangle$, $b_1$ is a coefficient of $|001\rangle$, etc. The minterms of canonical SOP obtain thus the following encoding (symbol $\cdot$ is arithmetic multiplication)[3] $b_1 = en(a_2)$, $b_2 = en(a_1)$, $b_3 = en(a_1 \cdot a_2) = en(a_1) \oplus en(a_2) = b_2 \oplus b_1$, $b_4 = en(a_0)$, $b_5 = en(a_0 \cdot a_2) = en(a_0) \oplus en(a_2) = b_4 \oplus b_1$, $b_7 = en(a_0 \cdot a_1 \cdot a_2) = en(a_0) \oplus en(a_1) \oplus en(a_2) = b_4 \oplus b_2 \oplus b_1$.*

*Applying now the encoding from Eqn. 20 and substituting into the above PPRM one obtains $PPRM = b_0 \cdot 1 \oplus (b_0 \oplus b_1) \cdot x_3 \oplus (b_0 \oplus b_2) \cdot x_2 \oplus [(b_0 \oplus b_1 \oplus b_2) \oplus (b_2 \oplus b_1)] x_2 \cdot x_3 \oplus [(b_0 \oplus b_2 \oplus b_4) \oplus (b_4 \oplus b_2)] x_1 \cdot x_2 \oplus (b_0 \oplus b_4) x_1 \oplus [(b_0 \oplus b_1 \oplus b_4) \oplus (b_4 \oplus b_1)] x_1 \cdot x_2 \oplus [(b_0 \oplus b_1 \oplus b_2) \oplus (b_2 \oplus b_1) \oplus (b_4) \oplus (b_4 \oplus b_1) \oplus (b_4 \oplus b_2) \oplus (b_4 \oplus b_2 \oplus b_1)] \cdot x_1 \cdot x_2 \cdot x_3 = b_0 \cdot 1 \oplus (b_0 \oplus b_1) x_3 \oplus (b_0 \oplus b_2) x_2 \oplus (b_0 \oplus b_4) \cdot x_1$. Thus, $PPRM = b_0 \oplus (b_0 \oplus b_1) x_3 \oplus (b_0 \oplus b_2) x_2 \oplus (b_0 \oplus b_4) x_1$ which corresponds to all affine functions on variables $x_1, x_2, x_3$.*

If oracle $\mathcal{O}$ contains function $f(x_1, ..., x_k)$ that is not affine, a modification to any one of the affine functions $A_i(x_1, ..., x_k)$ must be made. This can be done by adding a circuit (such as $QBIST_{32}(x_1, ..., x_k)$) and can be thought of as EXORing it with some function, like this:

$$f(x_1, ..., x_k) \oplus BIST_i(x_1, ..., x_k) = A_i(x_1, ..., x_k). \tag{21}$$

Thus, $f(x_1, ..., x_k) = BIST_i(x_1, ..., x_k) \oplus A_i(x_1, ..., x_k)$. The general disentanglement procedure is as follows:

1. Each function $A_i(x_1, ..., x_k) \oplus BIST_i(x_1, ..., x_k)$ is realized as an ESOP.
2. $BIST_i(x_1, ..., x_k)$ with the minimum cost is selected.

[3] This is also called the polarity table in which one considers a Boolean function over variables $\{-1, 1\}$ instead of $\{0, 1\}$. In this case, XOR ($\oplus$) over $\{0, 1\}$ is equivalent to real multiplication over $\{-1, 1\}$.

3. Function $BIST_i(x_1, ..., x_k)$ is added (XORed) after $f$ as $QBIST_{32}$.

**Theorem 12** *The minimum number of product terms in the ESOP realization of the BIST circuit* $ESOP[\mathrm{BIST}(x_1, ..., x_k) \oplus A_i(x_1, ..., x_k)]$ *where* $A_i$ *is an arbitrary affine function on variables* $x_1, ..., x_k$ *is equal to* $p - k$ *where* $p$ *is the minimal number of product terms in* $ESOP(\mathrm{BIST}(x_1, ..., x_k))$.

**Proof 12** *Given is the minimal ESOP, denoted by ESOP(BIST), of function* $BIST(x_1, ..., x_k)$. *Let* $A$ *be an arbitrary affine function on variables* $x_1, x_2, ..., x_k$ *and* $c_0 \oplus c_1 \cdot x_1 \oplus ... c_k \cdot x_k$, *where* $c_i \in \{0, 1\}$. *There are two of these functions that have the maximum number of variables equaling* $k$*;* $x_1 \oplus x_2 \oplus ... x_k$ *and* $1 \oplus x_1 \oplus x_2 \oplus \cdots \oplus x_k = \bar{x_1} \oplus x_2 \oplus ... x_k$. *Assuming that ESOP(BIST) has the minimal number of product terms, the following cube pair types must not be included in it:* $x_i \cdot x_j \oplus x_i$, $x_i \cdot x_j \oplus x_i \cdot \bar{x_j}$, $x_i \cdot \bar{x_j} \oplus \bar{x_i} \cdot x_j$, $x_i \cdot x_j \oplus \bar{x_i} \cdot \bar{x_j}$. *The only product terms possible in ESOP(BIST) are necessarily* $x_i$, $\bar{x_i}$, $x_i \cdot x_j$, $x_i \cdot \bar{x_j}$, $x_i \oplus x_i \cdot x_j \cdot ... \cdot x_k$. *If one writes ESOP(BIST*$\oplus A_i$*) as* $ESOP(BIST) \oplus x_1 \oplus x_2 \oplus \ldots x_k$ *provided all the best merging cases, then all variables (literals) from A are merged, each of them with some literal from ESOP(BIST), like this:* $x_i \oplus x_i = 0$, $\bar{x_i} \oplus \bar{x_i} = 0$ *and* $x_i \oplus \bar{x_i}$. *Each of these cases will decrease the ESOP cost by one. Merging* $x_i$ *with* $x_i \cdot x_j = x_i$*;* $x_i \oplus x_i \cdot x_j = x_i \cdot \bar{x_j}$ *will not change the ESOP cost. All other mergings will increase the cost of the ESOP(*$\mathrm{BIST} \oplus A_i$*) with respect to ESOP(BIST). Thus, the number of terms in the ESOP can be decreased by no more than* $k$. *Observe also that the highest decrease of cost is when BIST is already an affine function.*

### 3.7 Possible Extensions and Applications

An alternative approach based on the theory outlined in tests $T_3$ and $T_4$ utilizes highly controllable test vectors. The growth in additional circuitry is thus replaced with linear growth in the number of experiments needed. The total number of experiments in this second method is $(5 + 4\lceil k/2 \rceil)$. There is little added growth in circuit complexity. Tests $T_3$ and $T_4$ are replaced with first repeating the circuitry needed in test $T_1$. (All replaced tests of course have state $|-\rangle$ at the target.) Next, starting with the top 2 qubits (Fig. 11), an EPR pair is generated to test the oracle and mirrored with a measurement in the Bell basis. This is then moved down all the top $k$ qubits (Fig. 12) a total of $2\lceil k/2 \rceil$ times. The EPR generating circuitry is used to create inputs that are products of state $|01\rangle \pm |10\rangle$ and $|1\rangle$. These must be repeated with both positive and negative versions to satisfy Requirement 4. This results in something in classical test known as *walking-a-zero* [8] (except quantum mechanics allows two zeros to be walked at the same time). This alternative approach however, does not probe the oracle under the types of inputs experienced when used in a Grover search algorithm. It does however illustrate that the algorithm can be modified to reduce the complexity of the stages needed to extract information. Alternative applications of the methods presented in this paper also exist.

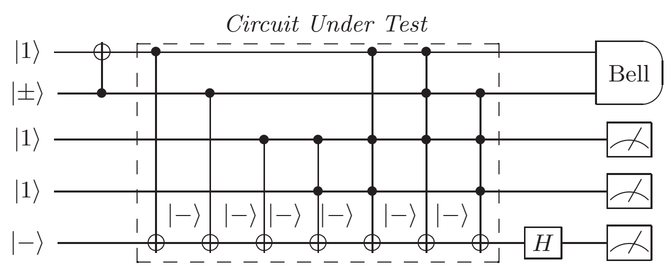


Figure 11: Alternative setup for tests $T_3$ and $T_4$: Test $|0111\rangle \pm |1011\rangle$. The target of each $k$-CN gate acts on state $|-\rangle$. No entanglement is added in either test, since all relative phases will result in a product measurement in the Bell basis.

# 4 Conclusion

This work reduced the classical test problem by utilizing entanglement as a controllability resource. Classically, the lower bound of this circuit class was found to be $(k + 4 + 2n_e)$ by Reddy [2] (where the $2n_e$ term depends on the function being realized). Quantum effects were used to reduce the test problem to a linear growth of $(5 + 4\lceil k/2 \rceil)$ in experiment count. When testing an oracle, states become non-local due to the phase change undergone by all true minterms as seen in tests $T_3$ and $T_4$ in Sec 3.4 and 3.5. It was shown in Sec. 3.6 that all affine oracles generate no net entanglement when used as a search oracle,

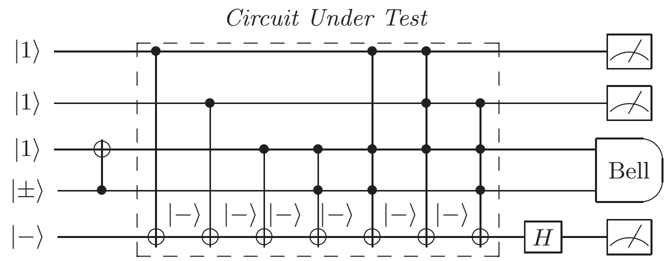


Figure 12: Alternative setup for tests $T_3$ and $T_4$: Test $|1101\rangle \pm |1110\rangle$.

while an oracle realizing a bent function requires the greatest effort to disentangle the state and return the system to a local product state. Since there are $2^{k+1}$ affine functions, Sec. 3.6 addressed the question of how close an arbitrary state is to a factorable state with phase terms that represent the spectrum of an affine function. The distance in many cases is close, but the upper bound is $\sim \Theta(N-k)$. Linear and Affine functions are very easy to test when realized quantum mechanically. Based on the potential limitations highly controllable test vectors were developed in Sec. 3.7 that do not undergo phase induced entanglement when propagation though a phase oracle occurs. In a correspondence from Agrawal in 1981 [12], fault detection probability was shown to be the highest when the information output of a circuit is maximized. An information theoretic approach to quantum fault testing might lead to further useful insight into the quantum test problem.

# References


[1] W. Kautz, *Testing faults in combinational cellular logic arrays*, Proceedings of 8th annu. Symp. Switching and Automata Theory, Oct. 1971, pp. 161-174.

[2] S. Reddy, *"Easily Testable Realizations for Logic Functions,"* IEEE Transactions on Computers, Vol. C-21, No. 11, pages 1183 - 1188, November (1972).

[3] D. Deutsch, *"Quantum computational networks,"* Proc. R. Soc. London A, 425:73, (1989).

[4] A. Barenco, C. Bennett, R. Cleve, D. DiVincenzo, N. Margolus, P. Shor. T. Sleator, J. Smolin, and H. Weinfurter, *"Elementary gates of quantum computation,"* Phys.Rev.A, 52(5):3457-3467, (1995), `quant-ph/9503016`.

[5] A. Gilchrist, N. Langford and M. Nielsen, *"Distance measures to compare real and ideal quantum processes,"* Phys. Rev. A 71, 062310 (2005), `quant-ph/0408063`.

[6] J.B. Altepeter, D. Branning, E. Jeffrey, T.C. Wei, P.G. Kwiat, R.T. Thew, J.L. O'Brien, M.A. Nielsen and A.G. White, *"Ancilla-assisted quantum process tomography,"* Phys. Rev. Lett. 90, 193601 (2003), `quant-ph/0303038`.

[7] J.D. Biamonte, J.S. Allen and M.A. Perkowski, *"Fault Models for Quantum Mechanical Switching Networks,"* 22 pages, (2005), `quant-ph/0508147`.

[8] U. Kalay, M. Perkowski and D. Hall *"A Minimal Universal Test Set for Self-Test of EXOR-Sum-of-Product Circuits,"* IEEE Transactions on Computers, vol. 49, no. 3, pp. 267-276, (2000).

[9] W. Zurek, *"Reversibility and Stability of Information Processing Systems,"* Phys. Rev. Lett. 53, pages 391-394, (1984), `DOI: 10.1103/PhysRevLett.53.391`.

[10] E. McCluskey and C.W. Tseng, *"Stuck-fault tests vs. actual defects,"* in Proc. of Int. Test Conf., pp. 336-343, (2000).

[11] T. Sasao, *"Easily Testable Realizations for Generalized Reed-Muller Expressions,"* IEEE Trans. Computers, 46(6), pp. 709-716, (1997).

[12] V. Agrawal, *"An Information Theoretic Approach to Digital Fault Testing,"* IEEE Transactions on Computers, vol. 30, pages 582 - 587, August (1981).