Programmable Daisychaining of Microelectrodes for IP Protection in MEDA Biochips

Tung-Che Liang
Duke University
tl221@duke.edu

Krishnendu Chakrabarty
Duke University
krish@duke.edu

Ramesh Karri
New York University
rkarri@nyu.edu

***Abstract*—As digital microfluidic biochips (DMFBs) make the transition to the marketplace for commercial exploitation, security and intellectual property (IP) protection are emerging as important design considerations. Recent studies have shown that DMFBs are vulnerable to reverse engineering aimed at stealing biomolecular protocols (IP theft). The IP piracy of proprietary protocols may lead to significant losses for pharmaceutical and biotech companies. The micro-electrode-dot-array (MEDA) is a next-generation DMFB platform that supports real-time sensing of droplets and has the added advantage of important security protections. However, real-time sensing offers opportunities to an attacker to steal the biochemical IP. We show that the daisychaining of microelectrodes and the use of one-time-programmability in MEDA biochips provides effective bitstream scrambling of biochemical protocols. To examine the strength of this solution, we develop a SAT attack that can unscramble the bitstreams through repeated observations of bioassays executed on the MEDA platform. Based on insights gained from the SAT attack, we propose an advanced defense against IP theft. Simulation results using real-life biomolecular protocols confirm that while the SAT attack is effective for simple instances, our advanced defense can thwart it for realistic MEDA biochips and real-life protocols.**

I. Introduction

Digital microfluidic biochips (DMFBs) have recently been transitioned to the marketplace for automating biomolecular protocols. In 2015, Illumina announced the use of DMFBs for sample preparation [1]. As another example of commercialization, Baebies recently introduced a USDA-approved product to screen newborns for diseases [2]. These milestones highlight the emergence of digital microfluidic technology for commercial exploitation and its potential for point-of-care diagnosis, sample processing, and cell-based assays [3], [4], [5]. It is anticipated that companies will develop intellectual property (IP) in the form of kits for various biochemical applications, and the protocols executed by these kits will incorporate sensitive information, e.g., advanced technology for accelerating deep DNA sequencing or qPCR-based gene-expression analysis.

Recent work has shown that an attacker can reverse engineer a proprietary protocol by analyzing the actuation sequence or the video frames recorded by a CCD camera [6]. Reverse engineering aimed at stealing biomolecular protocols (IP theft) is of particular concern. The IP piracy of proprietary protocols may lead to significant losses for pharmaceutical and biotechnology companies.

*This research was supported in part by the Army Research Office under grant number W911NF-17-1-0320 and the National Science Foundation under grant number CNS-1833622 and grant number CNS-1833624.

Obfuscation is an attractive approach for IP protection. An obfuscation mechanism for DMFBs was proposed recently to secure IP protocols using a finite-state machine [7]. It first proposes a physical unclonable function (PUF) whose response is based on the inherent variation in the fluidic operations of DMFBs. This work then utilizes the combination of PUF response bits and the license issued by the foundry as a key. Trusted users can gain access to the actuation sequences based on the challenge-response pairs associated with the PUF module. However, the PUF-based scheme is not secure against malicious adversaries who pretend to be legitimate users. Because commercialized biochips are sold in open markets, biochip developers cannot ensure that all users are trustworthy. The work in [6] showed that information about the proprietary protocol may be leaked through actuation sequences or data obtained from a CCD camera. Therefore, secret keys should be used to obfuscate actuation sequences and sensor data [6].

The microelectrode-dot-array (MEDA) is a next-generation DMFB platform that supports real-time sensing of droplets and has the added advantage of important security protections [8], [9], [10]. A MEDA biochip is composed of an array of identical microelectrode cells (MCs). It has been fabricated at TSMC using a mainstream μ CMOS process, and an example of a fabricated chip is shown in Fig. 1. Each MC consists of a microelectrode, an electronic control circuit, and a sensing module that enables real-time sensing of droplets. The MCs are connected together to form a daisychain. Therefore, unlike electrodes in traditional DMFBs that are wired to distinct input ports for actuation, a MEDA biochip requires only one input port to actuate all the microelectrodes. The actuation pattern corresponding to all the microelectrodes is mapped to a bitstream, and this bitstream is scanned in to the daisychain through the input port. The translation of the actuation pattern to the bitstream is based on the daisychain structure, and the ordering of the microelectrodes in a daisychain determines the mapping of the actuation sequence to the scanned-in bitstream. By exploiting this mapping in MEDA, the actuation sequences can be scrambled, and information leakage of the proprietary protocols can be avoided.

In this paper, we present a programmable daisychain structure for MEDA biochips and show that the daisychaining and use of one-time-programmability can protect biochemical protocol IP. The daisychain structure is designed to be one-time-programmable after manufacturing and serves as a hidden key so that only authorized protocols are performed on the biochip. In addition, the scan-out data of the MEDA biochip is scrambled by the daisychain structure. To examine the

978-1-7281-4823-6/19/$31.00

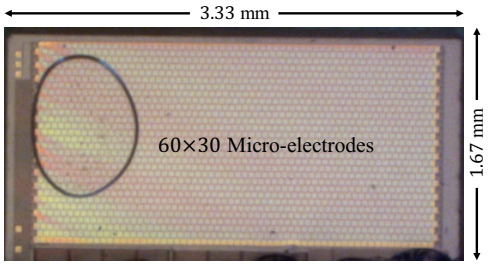


Fig. 1: Photo of a MEDA biochip with a dispensed droplet. (Taken during our lab experiment.) The movement of the droplets can be monitored in real time during biochip operation.

strength of this solution, we develop a Boolean Satisfiability (SAT)-attack that can decipher the daisychain structure by observing the execution of the bioassays on the platform. Based on the insights gained from the SAT attack, we propose an enhanced daisychain structure. Simulation results using real-life biomolecular protocols and a theorem confirm that while the SAT attack is effective on simple instances, it can be thwarted by the advanced defense for realistic MEDA biochips and real-life protocols.

The key contributions of this paper are as follows:

- We describe a programmable daisychain structure that can be scrambled. The programmable structure can be mass-produced in a typical semiconductor foundry. As a result, the cost of adding programmability to the daisychain is negligible.
- We formulate a SAT attack on the programmable daisychain. The objective of this attack is to decipher the microelectrodes-to-actuation sequence mapping by monitoring the inputs and outputs and the movement of the droplets on MEDA.
- We show using real-life bioassay protocols that, even with a programmable daisychain, the SAT attack can decipher the mapping by observing bioassay execution.
- We present an enhanced programmable daisychain structure and prove that this advanced structure can defend successfully against SAT attacks.

The remainder of this paper is organized as follows. Section II describes MEDA and introduces the programmable daisychain. Section III first presents the threat model and possible attacks and then shows that, among these attacks, only the SAT attack can successfully discover the structure of the daisychain. Section IV presents an enhanced defense against the SAT attack. Section V shows experimental simulations that the SAT attack can reveal the structure of the daisychain during bioassay execution, but an advanced daisychain structure can defend against the attack. Section VI discusses the overheads of using the enhanced daisychain structure and compares it with prior work in scan-chain security. Finally, conclusions are drawn in Section VII.

II. Technology Overview

The schematic of an MC is shown in Fig. 2. Each MC can be operated in three modes: 1) scanning, 2) actuation, and 3) sensing. In the scanning mode, the bit stored in an MC is passed to the next MC when the clock signal of the flip-flop changes. In the actuation mode, the stored bit determines if the microelectrode is actuated. If the stored bit is '1', the microelectrode is actuated; the microelectrode remains at low voltage otherwise. In the sensing mode, the presence of a droplet over the microelectrode is determined by the MC. If there is a droplet over the microelectrode, a '1' is stored in the flip-flop, '0' otherwise.

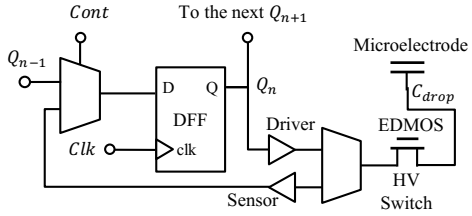


Fig. 2: The microelectrode cell in a MEDA biochip.

All of the MCs in a MEDA biochip operate synchronously. Fig. 3 shows a droplet on a MC array. An actuation sequence is applied to the array in order to transport the droplet to the right. First, all MCs are set to the scanning mode, and the translated bitstream, , is scanned to the array. Next, all MCs are set to the actuation mode, and the scanned-in actuation pattern is applied to the microelectrodes. The activated microelectrodes are marked in a dark-gray color. The droplet is thus transported to the right. Next, all MCs are set to the sensing mode, and the droplet information is stored in all MCs. Finally, all MCs are set to scanning mode again, and the sensor data is scanned out as a bitstream. These four steps compose an operation cycle. To perform a bioassay on a MEDA biochip, the operations of the bioassay are synthesized and translated into a sequence of bitstreams and these bitstreams are scanned in to the MEDA biochip cycle by cycle.

A. Daisychaining in MEDA

We next explain how the actuation bitstreams are translated according to the daisychain structure; see Fig. 4. Two MEDA biochips, Chip 1 and Chip 2, contain the same number of microelectrodes, but their daisychain structures are different.

To actuate the same microelectrodes on these biochips (marked with dark-gray color in Fig. 4(a)), the actuation bitstream for Chip 1 differs from that for Chip 2. Let the actuation bitstream for Chip 1 be . The actuation bitstream for Chip 2 is given by . If we exchange the bitstreams for the two chips, i.e., scan in for Chip 2 and for Chip 1, the desired actuation patterns cannot be applied

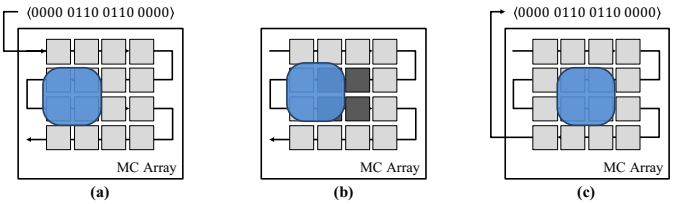
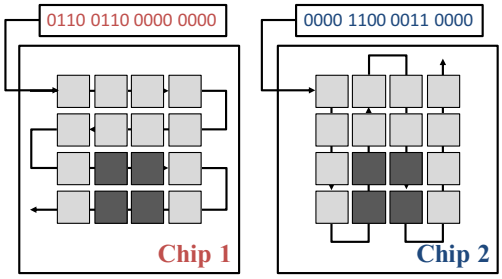
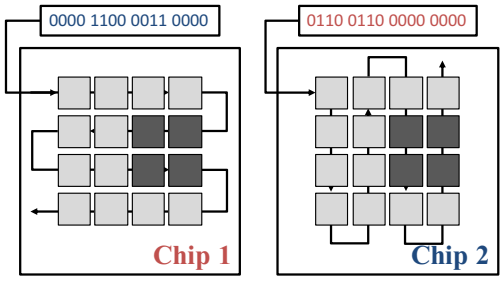


Fig. 3: A droplet on a 4 4 array is transported by the actuation sequence: (a) scan in the actuation bitstream; (b) apply actuation pattern to all microelectrodes; (c) sense and scan out the bitstream.



(a)



(b)

Fig. 4: Two distinct daisychain structures with their actuation sequences. (a) In order to actuate the same microelectrodes on two separate MC arrays, two different actuation bitstreams are required. (b) If we exchange the actuation bitstreams, the desired fluidic operations cannot be performed on the arrays.

to the microelectrodes. These switched actuation patterns are shown in Fig. 4(b). Therefore, without knowing the daisychain structure, a user cannot execute any fluidic operations on the biochip; nor can the user comprehend the scanned-out data for a bioassay. By exploiting this characteristic, MEDA biochips provide an extra translation layer that can be used to scramble actuation bitstreams and the sensor data.

B. Programmable Daisychaining

For a given MC array, each MC is surrounded by two to four MCs (depending on whether the MC is on the edge). The input/output of an MC connects to the output/input of a surrounding MC. To make the daisychain programmable, a multiplexer (MUX) and a demultiplexer (DeMUX) can be added at the input and the output of each MC, respectively; the structure is shown in Fig 5. By carefully assigning control bits to the MUX and DeMUX, the next MC and the previous MC are connected in a specific way. The control bits of all MCs can be wired to a read-only-memory (ROM) array that is one-time-programmable after manufacturing so that the daisychain structure is also one-time-programmable.

The original MC contains 36 CMOS transistors and an extended-drain MOS (EDMOS) [11]. The additional gates, a MUX and a DeMUX, require 36 CMOS transistors. Therefore, the new MC is twice as large as the original MC. However, the increased area overhead is acceptable because it does not affect the fluidic operations on MEDA biochips. The original MC in [12] was designed in a layout area of μ using the μ process. The proposed new MC requires an area of . According to [13], the radius of the smallest droplet that can be dispensed and moved on MEDA biochips is μ . The area of the new MC is approximately the same as the smallest area occupied by a droplet. As a result, all MEDA-enabled operations can be performed on an array made up of the new MCs.

III. Attacks to Discover the Daisychain Structures

In this section, we present a threat model that benefits from discovering the structure of the daisychain. We then propose three attacks to discover the structure of a daisychain. We first investigate if a brute-force attack can discover the daisychain structure, i.e., check if an attacker can enumerate all possible daisychains in an MC array and test each one of them to find out the correct structure. Second, we examine if differential-analysis can reveal the daisychain structure. Because these two attacks cannot effectively reveal the secret structure, we propose a powerful SAT-based attack to discover the daisychain structure.

A. Threat Model

We assume that the attacker has access to fabricated MEDA biochips and the actuation bitstreams of bioassays. The goal of the attacker is to pirate as many biochemical protocols as possible for financial gain. The attacker is interested in discovering the mapping between the actuation bitstreams and the microelectrodes. Once the attacker comprehends the mapping, s/he can recover the fluidic operations in the bioassays without executing them on a MEDA biochip and observing their execution. Therefore, we make these assumptions: 1) an attacker applies an available bioassay with the complete kit provided by the vendor on a MEDA biochip to collect information, and the kit includes reagent solutions as well as actuation bitstreams; 2) s/he can observe the droplet movements on the MEDA biochip during the bioassay execution; 3) s/he has access to the scanned-in/scanned-out bitstreams by monitoring the signals at the input/output port. These observations are not intrusive, and thus s/he can acquire as much information as possible for a given bioassay. Once enough information is collected, s/he analyzes these observations in order to discover the mapping between the bitstream and the microelectrodes and steal biochemical IPs.

B. Brute-force Attack

The number of daisychains on a 2D-array of MCs can be formulated as the number of Hamiltonian paths in a 2D grid graph. The daisychaining of MCs must satisfy two constraints.

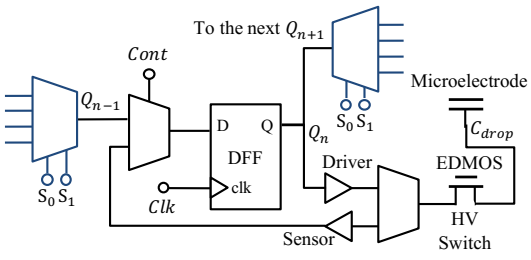


Fig. 5: The programmable microelectrode cell.

First, all MCs need to be connected as a chain for scanning in/out actuation/sensing bitstreams. Second, the MCs need to be connected using shortest wiring to obtain a small biochip. This can be accomplished by wiring physically adjacent MCs on the array as a daisychain. A MEDA biochip with MC array can be modeled as an undirected grid graph , where nodes in are the MCs. Let the node corresponding to the th row and th column be denoted as . For any given if and are physically adjacent on the MC array, i.e., (and) or (and). A *Hamiltonian path* is a path in the graph that visits each vertex exactly once [14]. As a result, a daisychain in an MC array can be modeled as a Hamiltonian path in the corresponding graph . Fig. 6 shows a MC array and the corresponding graph . Four Hamiltonian paths can be found in .

It is well-known that determining whether a graph contains a Hamiltonian path is an NP-complete problem [15]. As a result, it is even more difficult to determine how many distinct Hamiltonian paths exist in a given graph. To the best of our knowledge, a closed-form solution for this problem is not available, and the problem remains unsolved. However, algorithms have been proposed to exhaustively enumerate all paths for a given grid graph [15], [16]. Because of computation and memory limits, these algorithms can only enumerate paths for small-sized grid graphs. The enumeration of all paths for square grid graphs that are or smaller is shown in Table I. For a grid graph that contains vertices, there are distinct Hamiltonian paths [16], [15]. Because algorithms today can only enumerate Hamiltonian paths for less than vertices in a grid graph, for a normal-sized MEDA biochip (with MCs), it is not feasible for an attacker to exhaustively enumerate all possible daisychains.

C. Differential Analysis

We propose a differential analysis to discover the mapping between the bitstreams and the MCs. Consider the example in Fig. 7 where three consecutive observations are made during a bioassay execution on a MC array. Let the set of all microelectrodes be and the set of microelectrodes that are under the droplets at time be . In this example, , , and . The overlaps between these sets can be shown by a Venn diagram as in Fig. 8. Because and , the mapping of to the corresponding bit in the bitstream can be found. Similarly, because , the mapping of to the corresponding bit in the bitstream can also

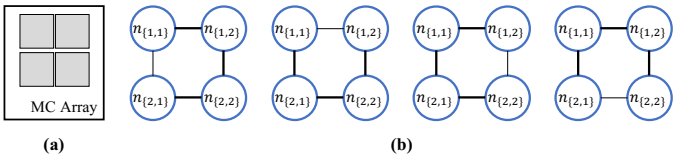


Fig. 6: An example of the daisychain enumeration. (a) The MC array contains 2 2 microelectrodes. (b) Four Hamiltonian paths can be found in the corresponding grid graph , .

TABLE I: Number of Hamiltonian paths on an square lattice, where 1 17 [15].

	Number of paths
2	4
3	20
4	276
5	4324
6	299348
7	13535280
8	3023313284
9	745416341496
10	730044829512632
11	786671485270308848
12	3452664855804347354220
13	16652005717670534681315580
14	331809088406733654427925292528
15	7263611367960266490262600117251524
16	66263471738497979323881410137798878688 4
17	664289947391594699694401195797368076126655 40

be found. The five sets of microelectrodes of size are colored green in Fig. 8, and the microelectrode mappings that correspond to these sets can be found. However, the remaining microelectrode mappings cannot be found (even though all mappings are analyzed using the bitstreams). Therefore, a total of possible structures can be obtained using the differential analysis, and the probability of guessing the right structure is —.

From the above example, we can examine the best and the worst cases from the attacker's perspective for differential analysis if we obtain three observations on a MC array. For the worst-case scenario, , and the sets in the corresponding Venn diagram contain either more than one or zero elements. Therefore, no microelectrode mappings can be found. The number of possible structures is , and the probability of guessing the right structure is — . On the other hand, the ideal scenario for the attacker is that seven-out-of-eight sets in the corresponding Venn diagram contain only one element, respectively, and the other set contains two elements. In this case, the probability of guessing the right structure is —. However, for this ideal case, the microelectrodes in any set (, , or) form a non-rectangular shape on the MC array, which is impossible to obtain from the sensor result of a droplet. For example, the droplet cannot be an 'L' shape on the MC array. As a result, it is not possible for an attacker to obtain the observations of the ideal case from bioassay execution.

Based on the analyses of the best-case and worst-case scenarios, we learn that observations obtained from a bioassay execution impact the effectiveness of differential analysis.

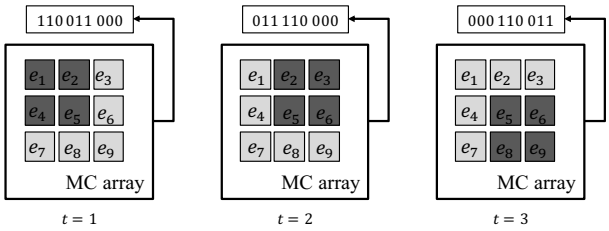


Fig. 7: An example of three observations.

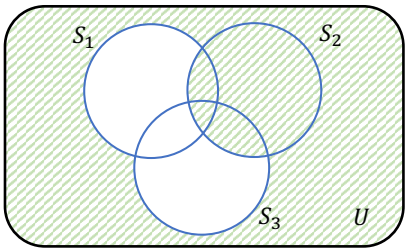


Fig. 8: Relationships between the microelectrode sets in Fig. 7.

Therefore, if a bioassay is executed on an MC array in a particular way such that the observations obtained from the execution are not informative (i.e., deliberately differential analysis resistant), differential analysis does not prune away possible structures, and the probability of guessing the correct structure remains low. We simulate the executions of three real-life bioassays, namely the multiplexed in-vitro bioassay [17], the chromatin immunoprecipitation protocol (ChIP) [18], and the gene-expression analysis [19], on a MEDA biochip with MCs. The bioassays are executed in a way such that some microelectrodes are not used during its execution. Because these microelectrodes are not used during the execution, the corresponding bits in the bitstream are all '0's. We then apply differential analysis to the observations obtained from these executions and show the results in Fig. 9. Let the number of observations that are used by differential analysis be . The simulation results show that the number of discovered microelectrode-to-bitstream mappings increases when . However, the number of discovered mappings saturates when because beyond this point, it only offers information on the already-discovered mappings.

Let the number of mappings found from differential analysis be , where is the number of observations. Therefore, the number of possible structures obtained from differential analysis is , and the probability that an attacker can decipher the scrambled microelectrode-to-bitstream structure is ———————. Because for all according to the simulation, the probability of guessing the right structure is less than ——————— ——— ———. Thus, it is unlikely that an attacker can guess the correct structure using differential analysis on a MEDA biochip.

D. SAT-Based Attack to Discover the Daisychain Structure

In order to further evaluate the security strength of the programmable daisychaining, we propose a SAT-based attack to discover the daisychain structure. In this section, we first present the notation and the SAT model. We next show that even though the attacks described thus far in this section cannot unscramble the scanned-out bitstreams, the SAT attack can.

1) Notation and Formal Model

We adopt the attack model used by the brute-force and differential-analysis attacks. Let the th microelectrode on the MEDA biochip be denoted as , and the naming order is as shown in Fig. 7. Assume the attacker obtains vectors and for each observation from a MEDA biochip with microelectrodes, where $\mathbb{B}$, $\mathbb{B}$, and $\mathbb{B}$. The vector

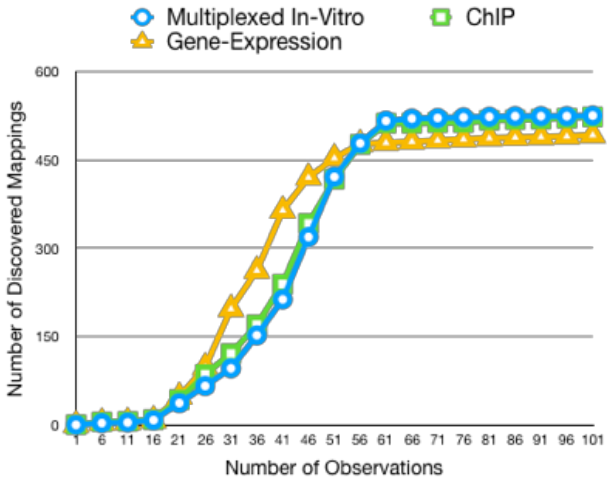


Fig. 9: The number of discovered microelectrode mappings using the differential analysis on three real-life bioassays. The entire 60 30 MEDA biochip is monitored by the attacker.

represents the droplet locations; if a component , the th microelectrode is under a droplet; if a component , the th microelectrode is not covered by a droplet. The vector represents the scanned out bitstream. An example is shown in Fig. 7(a), where and .

We define a Boolean matrix to be a key matrix, where . For a scanned-out bitstream, each microelectrode should map to a unique bit in it. The vector $\mathbb{B}$ represents bits in that may map to the microelectrode . Let be the th component of . If a component , the microelectrode can be map to the th bit in the bitstream. Without any observations, the for all because each microelectrode can be mapped to any bit in the bitstream.

When is set to the correct matrix, . For consecutive observations, . If an attacker observes a small subset of and , i.e., is small, the attacker might obtain a key matrix that satisfies all observations; however, may not apply to the new ( )th observation [20]. Therefore, this key is not correct.

2) SAT Attack: Problem Formulation

Let the correct key be . The attacker's goal is to find . This is equivalent to solving the quantified Boolean formula:

3) Algorithm Overview

We can use a SAT solver to generate a to satisfy all observations. However, the solution may not satisfy a new observation. Consider the example in Fig. 7 from Section III-C. Three observations are made during protocol execution.

Let us consider the scenario when the attacker obtains one observation at time in Fig. 7, i.e., and . When these vectors are fed to the SAT solver with the proposed

model, the solver returns possible solutions. However, some of these solutions are not legal for the daisychain structure. For example, in this case, the solver returns two illegal solutions $K'_{9,9}$ and $K''_{9,9}$, where

$$K'_{9,9}=\begin{bmatrix}1&1&0&0&1&1&0&0&0\\1&1&0&0&1&1&0&0&0\\0&0&1&1&0&0&1&1&1\\1&1&0&0&1&1&0&0&0\\1&1&0&0&1&1&0&0&0\\0&0&1&1&0&0&1&1&1\\0&0&1&1&0&0&1&1&1\\0&0&1&1&0&0&1&1&1\\0&0&1&1&0&0&1&1&1\end{bmatrix} \quad K''_{9,9}=\begin{bmatrix}1&0&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0&0\\0&0&1&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0\\0&1&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0&1\end{bmatrix}$$

The matrix $K'_{9,9}$ is not the key matrix $K^c_{9,9}$ because $\exists L(e_i)$ such that $\sum_j l_i^j \neq 1$, where l_i^j is the jth component in $L(e_i)$. This implies that the ith microelectrode can be mapped to more than one bit in the bitstream. As for the matrix $K''_{9,9}$, even though $\sum_j l_i^j = 1\ \forall L(e_i)$, $K''_{9,9}$ is not a legal matrix because it represents an infeasible daisychain structure. Let the daisychain structure of $K''_{9,9}$ be denoted as a vector DC. Note that $DC = \langle e_1, e_5, e_3, e_6, e_4, e_2, e_7, e_8, e_9\rangle$ because e_1 is related to the first bit of the bitstream, and e_5 is related to the second bit. For a legal daisychain structure, two adjacent components e_x and e_y in DC should be physically adjacent on the MC array because e_x should connect to e_y in the array to form a daisychain. While e_1 and e_5 are adjacent bits in DC, they are not physically adjacent on the MC array. Illegal key matrices can be automatically eliminated by: 1) checking if the sum of each row equals 1; 2) checking if the components in the translated DC are adjacent.

By pruning away the illegal solutions obtained by the SAT solver, we can retain legal key matrices. Two legal matrices are listed below:

$$K^3_{9,9}=\begin{bmatrix}1&0&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0&0\\0&0&0&0&0&0&1&0&0\\0&1&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&0&1&0\\0&0&1&0&0&0&0&0&0\\0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0&1\end{bmatrix} \quad K^4_{9,9}=\begin{bmatrix}1&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0\\0&0&0&0&0&1&0&0&0\\0&0&0&0&1&0&0&0&0\\0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0&1\end{bmatrix}$$

Both are legal for the observation $\overrightarrow{X_1}$ and $\overrightarrow{Y_1}$. We cannot decipher the correct structure (i.e., the key matrix) because we have a limited number of observations.

Let us consider the case when the attacker is able to make two more observations, i.e., for time $t = 2$ and $t = 3$ in Fig. 7. These observations are $\overrightarrow{X_2} = \langle 0,1,1,0,1,1,0,0,0\rangle$, $\overrightarrow{X_3} = \langle 0,0,0,0,1,1,0,1,1\rangle$, $\overrightarrow{Y_2} = \langle 0,1,1,1,1,0,0,0,0\rangle$, and $\overrightarrow{Y_3} = \langle 0,0,0,1,1,0,0,1,1\rangle$. Using the SAT solver, we can obtain the key matrix $K^4_{9,9}$. Since this is the only legal key matrix, the secret key is deciphered with only three observations, i.e., $K^4_{9,9} = K^c_{9,9}$. Recall that differential analysis cannot decipher the structure of the daisychain using the same three observations. Furthermore, this key matrix satisfies all new observations. Algorithm 1 outlines the SAT attack to discover the mapping between bits in the bitstream and the MCs in the MEDA biochip.

Algorithm 1 Unscramble the daisychain

Input: N observations of $\overrightarrow{X_i}$ and $\overrightarrow{Y_i}$
Output: Legal and possible key matrices $K^c_{N,N}$
1: $Solutions = \{\}$; Initialize a SAT solver SAT; $i = 0$;
2: **for** $i < N$ **do**
3: $\quad SAT(\overrightarrow{X_i}^T \cdot K_{N,N} = \overrightarrow{Y_i}^T)$;
4: **end for**
5: **for** a solution $K_{N,N}$ from SAT **do**
6: $\quad$ **if** ($K_{N,N}$ is legal) **then** Add $K_{N,N}$ to $Solutions$; **end if**
7: **end for**
8: **return** $Solutions$;

An attacker requires three observations to decipher the daisychain structure in a 3×3 MEDA biochip. This example shows that the simple daisychain structure cannot defend against the SAT attack. Hence, we propose an enhanced daisychain structure.

IV. Defense Against the SAT Attack

The SAT attack works on the simple daisychain because each microelectrode is associated with one fixed bit in the actuation and in the scanned-out sequence. Therefore, we propose to scramble the actuation and the scan-out bitstreams in each operation cycle, thwarting the SAT attack.

The enhanced daisychain structure divides the original daisychain into Q smaller sub-daisychains, and the sub-daisychains are enabled one after another based on the states of an n-bit linear feedback shift register (LFSR). The enhanced daisychain is shown in Fig. 10. When the biochip is in the actuation mode, only the enabled sub-daisychain receives the scanned in bitstream. When the biochip is in the scan-out mode, only the enabled sub-daisychain scans out the sensor data.

Consider an example where the enhanced daisychain has three sub-daisychains (each with l MCs) and a 4-bit LFSR. The states of the LFSR are shown in Table II. The controller is designed such that each LFSR state enables only one of the sub-daisychains. Here, we consider the scenario when the biochip is in the sensing (i.e., scan-out) mode. The same working principle applies to the actuation (i.e., scan-in) mode.

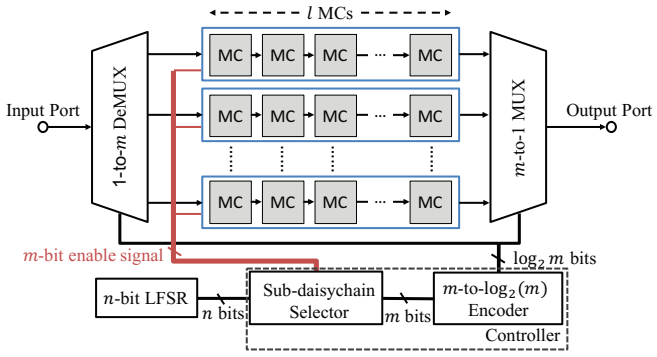


Fig. 10: The enhanced daisychain structure.

TABLE II: An example of the enhanced daisychain.

LFSR states	Enabled sub-daisychain	Scanned-out bitstream
0001, 1000, 0100	1, 2, 3	
0010, 1001, 1100	1, 3, 2	
0110, 1011, 1010	2, 3, 1	
1010, 1101, 1110	2, 1, 3	
1111, 0111, 0011	3, 1, 2	

When the biochip is activated, the LFSR is initialized with a seed of from an on-chip ROM, and the controller enables the first sub-daisychain based on this LFSR state. When the first bits of the sensor data is scanned out from the first sub-daisychain, the LFSR enters the next state , and the second sub-daisychain is enabled by the controller. For the subsequent LFSR state, the third sub-daisychain is enabled. After three states, the full sensor bitstream is scanned out. Let the sub-bitstreams from the first, the second, and the third sub-daisychain be , , and , respectively. The first full scanned-out bitstream can be denoted as . Similarly, the second full bitstream can be scanned out as . The remaining full bitstreams are shown in Table II.

In the above case, the initial seed is programmed in the ROM as . However, for each fabricated chip, the initial seed can be programmed differently. For example, if the initial seed is chosen as , then the first full bitstream is . Therefore, by assigning distinct initial seeds to different MEDA biochips, the order of scrambled patterns for each biochip can be unique.

A. Security Assessment

We consider attacks on the enhanced daisychain structure when: 1) an attacker does not know the architecture of the enhanced daisychain structure and 2) an attacker knows the structure of the LFSR.

1) Attacks Without Knowing the Architecture

Because the attacker does not know the architecture, we assume s/he can only use the SAT attack with all observations. The sub-bitstream order in the scanned-out bitstreams is randomized based on the LFSR states, i.e., the bits in the scanned-out bitstreams are not uniquely mapped to specific microelectrodes. The SAT attack in Section III is effective because each bit in the bitstream is associated with a unique microelectrode. When we randomize the association of each bit with a microelectrode in the scanned-out bitstreams, the SAT constraints from the observations become inconsistent. The SAT solver cannot generate a satisfying solution. To establish this claim, we present a theorem with its proof in the Appendix.

Theorem 1. *When we adopt the enhanced daisychain, the SAT attack cannot unravel the structure of the daisychain based on the continuous observations of bioassay execution.*

2) Attacks with the Knowledge of the Architecture

We next assume that an attacker knows the structure of the LFSR as well as the number of sub-daisychains, i.e., the attacker knows the number of scrambled patterns. For example, s/he can de-layer the MEDA biochip in order to discover the architecture of the MEDA biochip, including the LFSR size. Consequently, the attacker can employ the same SAT-based attack on the scanned-out bitstreams that correspond to the same scrambled pattern. If the attacker acquires a sufficient number of observations, s/he may successfully unscramble the bitstreams. The success of the attack depends on the number of observations for each scrambled pattern.

Assume that observations are obtained from a bioassay execution, the LFSR has states, the enhanced daisychain structure has sub-daisychains, and there are a total of scrambled patterns (i.e., the th bitstream and the th bitstream correspond to the same scrambled pattern). Consider the example provided in Table II. In this example, , , and the number of scrambled patterns is — . If a lengthy bioassay with many operations is executed on a MEDA biochip using the enhanced daisychain, an attacker can acquire a sufficient number of observations to unscramble the bitstreams.

As an example, we consider the execution of the gene-expression analysis bioassay, which has actuation bitstreams, i.e., . For each scrambled pattern, an attacker can obtain — — observations. In our simulation, which will be described in details in Section V, an attacker only needs observations to unscramble the bitstreams using the SAT attack.

From the above example, we learn that the probability that the attacker can unscramble the bitstreams increases with the number of observations. Conversely, the fewer observations an attacker can make, the less likely it is that s/he can unscramble the bitstreams. Therefore, a secure daisychaining should incorporate many scrambled patterns, i.e., should be sufficiently large. Since —, an ideal daisychain structure should contain a large number of the LFSR states ( ) and a small number of sub-daisychains ( ). Consider an enhanced daisychain such that an LFSR has states and sub-daisychains, i.e., and . The number of the scrambled patterns is — . As a result, for a bioassay that allows observations, an attacker can acquire on average — observations for each scrambled pattern.

We consider three representative bioassays (multiplexed in-vitro [17], ChIP [18], and gene-expression analysis [19]) and these offer , , and observations, respectively, based on their corresponding actuation sequences. Assuming that an attacker aims at unscrambling the bitstreams of the gene-expression protocol (with the largest number of observations among three bioassays), s/he can only obtain — observations on average for a scrambled pattern, which is less than for each scrambled pattern. Therefore, the attacker cannot unscramble the bitstreams of these existing bioassays. To unscramble a series of bitstreams that correspond to a scrambled pattern, our experiments (reported in Section V) show that there should be at least observations, i.e., — . As a result, for the attacker to unscramble the bitstreams on this enhanced daisychain structure, s/he needs at least observations from many bioassay executions.

To prevent an attacker from acquiring many observations from a MEDA biochip, we further propose a physical mechanism that periodically disables and resets the daisychain so that only a limited number of observations can be made for

TABLE III: Details of the biomolecular protocols.

	Number of Operations	Number of Operation Cycles
Multiplexed in-vitro bioassay	12	109
ChIP	16	285
Gene-expression analysis	18	317

a single MEDA biochip. A counter is added and integrated with the enhanced daisychain, and the counter records the number of observations that the biochip scans out. Once the number of observations exceeds a threshold, e.g., , which is much larger than that of representative bioassays ( ), the LFSR is initialized with an all-zeros seed. The daisychain is no longer functional, and the biochip cannot execute any bioassay.

Based on the proposed structure with the counter, an attacker can get at most —— observations for each scrambled pattern. Based on the analysis in Section III, the best-case scenario is that possible key matrices are generated. Therefore, the optimistic probability that an attacker can unscramble the bitstreams is ——. To unscramble all series of bitstreams (from all scrambled patterns), the probability is —— ——. As a result, it is extremely unlikely that an attacker can unscramble the bitstreams using a single MEDA biochip with the enhanced daisychain.

Now consider what might happen if an attacker fails in unscrambling bitstreams using one MEDA biochip. S/he may want to reproduce the attack on another MEDA biochip. Recall that each MEDA biochip has unique scrambled patterns based on the different seeds for the LFSR, i.e, the bitstreams of a bioassay is different for each MEDA biochip. Even if the attacker can afford the cost of new biochips, the acquired bitstreams of the same bioassay are different for the new biochips. Therefore, the attack information obtained from a biochip cannot be combined with that from attacking another biochip. Hence, the probability of unscrambling a bioassay does not increase using multiple MEDA biochips.

V. Experimental Results

We simulated the execution of three real-life biochemical protocols on a MEDA biochip: multiplex in-vitro diagnosis [17], gene-expression analysis [19], and chromatin immunoprecipitation (ChIP) [18]. We consider normal sized droplets which occupy microelectrodes. We implemented the simulator using Python on a workstation with 2.5 GHz Xeon processor and 2 GB memory, and we employed the Minisat solver for the SAT attack [21]. During bioassay execution, droplet locations and the scanned-out bitstreams are generated and used as inputs to the SAT solver, i.e., and . Because the numbers of fluidic operations in the above bioassays are different, the numbers of generated observations are also different. Details are shown in Table III.

The simulation results for a simple daisychained MEDA biochip are shown in Table IV. We record the number of legal key matrices and the CPU time required by the SAT solver across various observation points. The results confirm that the simple daisychain structures can be deciphered using the SAT attack.

TABLE IV: Results for simple and advanced daisychain scrambling.

Simple Daisychain scrambling	**Multiplex In-Vitro Analysis**				SAT attack cannot unscramble the advanced daisychain no matter the number of observations; see Theorem 1.
	Number of Observations	25	32	44	45
	Number of Key Matrices	220	105	20	1
	CPU Time (s)	6.72	3.43	2.56	2.11
	Chromatin Immunoprecipitation Protocol				
	Number of Observations	23	46	50	54
	Number of Key Matrices	280	110	35	1
	CPU Time (s)	6.73	3.48	2.84	2.71
	Gene-Expression Analysis				
	Number of Observations	48	54	63	69
	Number of Key Matrices	129	99	26	1
	CPU Time (s)	5.83	3.27	2.58	2.47

We repeated this experiment for a MEDA biochip with an enhanced daisychain. We considered three sub-daisychains and employed a 10-bit LFSR. We used the randomized bitstreams as the input to the SAT model across the observations. As expected from Theorem 1, the SAT attack failed to provide any legal key matrices for all three real-life biomolecular protocols.

VI. Discussion

In this section, we examine the timing and area overheads associated with the enhanced daisychain. We next present a comparison between our scrambling mechanism and existing scan-chain security methods. Finally, we explain how the SAT-based attack in this paper is unrelated to the wealth of research on SAT-based attacks on logic locking of digital designs.

A. Timing and Area Overheads

Timing overhead: The LFSR and the controller in the enhanced daisychain are designed in such a way that the scan mode does not require extra time. Recall that we assume each sub-daisychain contains MCs, and thus the LFSR changes its state when bits of the bitstream are scanned out. When the LFSR state changes, the controller enables one sub-daisychain and disables the previous sub-daisychain simultaneously. Therefore, bitstream scanning operates as in normal daisychaining, i.e., the enhanced daisychain does not incur any time overhead.

As an example, consider the first three scanned-out bitstreams from the enhanced daisychain in Table II. We provide the corresponding timing diagram that corresponds to these bitstreams and the enable signals for the three sub-daisychains in Fig. 11. Each bitstream can be divided into three parts, and each part is scanned out from a sub-daisychain. At any time, only a sub-daisychain is enabled, and the sensed data is scanned out from the enabled sub-daisychain. We can see that, by carefully designing the enable patterns, we can ensure that the enhanced daisychain does not incur any time overhead.

Area overhead: The addition of the LFSR and the controller introduces area overhead. The area overhead depends on the size of the LFSR. Increasing the LFSR size increases the number of states and thus offers security, but it also has a larger area overhead due to the complex decoder in the controller. In Table V, we present the number of CMOS transistors for the added circuits if a 10-bit LFSR is used in the enhanced daisychain. A total of transistors need to be integrated. The

TABLE V: Transistor counts for the added modules.

Module	10-Bit LFSR	Decoder	MUX	deMUX	Counter	Total
Number of Transistors	126	72	18	18	176	410

MC of a state-of-art MEDA biochip consists of 37 transistors and is fabricated in an area of $50\times50\,\mu m^2$ [12]. Since the security circuits are fabricated (along with the MC array) using the same $0.35\,\mu m$ fabrication process as the MCs, we can estimate the required area for the extra circuits to be $0.05\times0.05\times\frac{410}{37}=0.028\,mm^2$. A micro-photo of the state-of-art MEDA biochip is shown in Fig. 12. The area of the MEDA biochip is $3.3\times2.2=7.26\,mm^2$. Hence, the area overhead of the security modules is $\frac{0.028}{7.26}\times100\%=0.4\%$. Several sensing circuits have been fabricated with the MC array in MEDA biochips, such as high-resolution droplet sensing [11], and these circuits are shown in Fig. 12. Likewise, the extra circuits for the enhanced daisychain can be placed around the MC array so that they do not affect the MC array and the fluidic operations.

B. Comparison with Prior Work in Scan-Chain Security

Related to the daisychains in this paper, scan-chains of flip flops in an IC are used for testing. Scan chains enhance controllability and observability. However, they allow a malicious adversary access to confidential data [22]. Therefore, countermeasures have been developed to enable authorized users to operate the test mode. For example, VIm-Scan [23] requires users to scan in the secret keys in several iterations to unlock the chip for testing. In our enhanced daisychaining method, the correct daisychain structure is the secret. Without knowing the correct daisychain, the user cannot exploit the biochip.

Another authorization-based method, named Lock & Key, was proposed in [24]. This method requires that the chip with scan chains operate in two modes: secure and insecure. The chip is initialized (after reset) in the insecure mode. When the chip is in the insecure mode, the scanned-out results are randomly scrambled by the states of an LFSR and the scanned-out data is unpredictable. However, when a correct key is applied one clock cycle after the initialization, the chip switches to the secure mode and allows predictable operation of the scan chains. The chip remains in the secure mode until it is reset. Unauthorized users cannot exploit the scan chains without knowing the secret key. Even though this work and our enhanced daisychain employ LFSRs, there are two main differences: 1) The initial seed is programmed in the biochip so that the order of scrambling is fixed. This seed is provided by the authorized users for Lock & Key. 2) The LFSR in the enhanced daisychain determines the order of scan-in/scan-out bitstreams. On the other hand, the LFSR in [24] enables specific scan chains for testing.

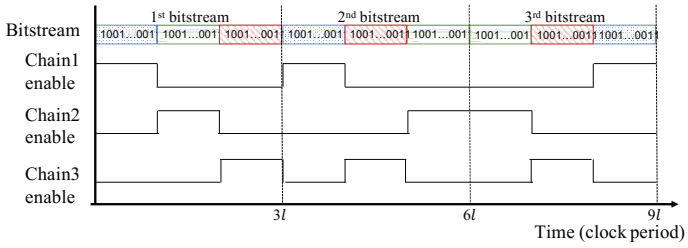


Fig. 11: Timing analysis results for the enhanced daisychain of Table II, where the time unit is the scanning clock period.

TABLE VI: SAT attack on logic locking and daisychain scrambling.

	Logic locking	Daisychain scrambling
Protection	Add key inputs to corrupt digital design outputs.	Scramble daisychain $\leftrightarrow$ actuation/scan-out bitstream map.
SAT inputs	Search for distinguishing inputs (DI) and create SAT constraints.	Observe droplet locations and the scanned out bitstreams during bioassay executions and create SAT constraints.

Scrambling of scan-chain was proposed in [25]. Here extra logic circuits are added in the scan chain. The scanned-out data is locked in a specific way that only authorized users can comprehend. Similar to our enhanced daisychain scrambling, [25] also scrambles the scanned-out result. However, it uses multiplexers in the scan chains to scramble the output. Our solution uses an LFSR to scramble the scan-in and scan-out bitstreams across several operation cycles.

C. Comparison with (Un)-related Work on SAT Attack on Logic Locking

Logic locking has been proposed to combat integrated circuit (IC) piracy and counterfeiting [26]. Logic locking inserts extra gates–the key gates–into the circuit to potentially corrupt the functionality [27], [26], [28]. Logic locking of digital designs is not related to the problem of scrambling of the mapping between the actuation and sensor bitstreams and the microelectrode cells in a biochip. Different SAT formulations are used to attack logic locking and biochip bitstream scrambling. Table VI summarizes the key differences between the SAT attacks on digital designs and on MEDA biochip daisychain structures.

VII. Conclusion

We have presented the use of daisychaining to secure IP protocols that are executed on DMFBs. We have also presented a SAT attack that can discover the simple daisychain scrambling, and simulations confirmed the effectiveness of the attack. An advanced daisychain structure was proposed to scramble the scan-in and scan-out data, and the simulations

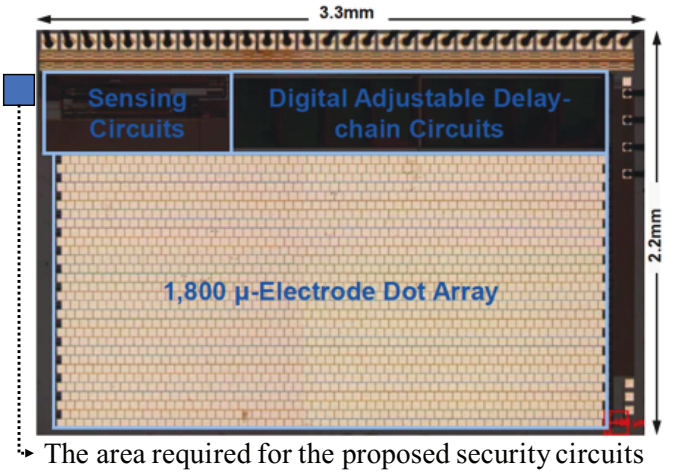


Fig. 12: A micro-photo of a state-of-art MEDA biochip [12]. Sensing modules have already been integrated in the MEDA biochip.

have shown that this defense is very effective against the SAT attack. We have also evaluated attacks on the defense as well as the overheads of using this defense.

ACKNOWLEDGMENT

The authors thank JV Rajendran, Ozgur Sinanoglu, and Benjamin Tan for their inputs on the assessment of the security countermeasures.

REFERENCES

[1] "Neoprep NFS library prep with digital microfluidics by illumina," 2014. [Online]. Available: https://support.illumina.com/content/dam/illumina-support/documents/documentation/system_documentation/neoprep/neoprep-system-guide-15049720-01.pdf
[2] "FDA advisors approve of baebies seeker analyzer for newborns," 2016. [Online]. Available: https://www.baebies.com/fda-advisors-back-approval-baebies-seeker-analyzer-newborns/
[3] T.-Y. Ho, K. Chakrabarty, and P. Pop, "Digital microfluidic biochips: recent research and emerging challenges," in *ACM CODES*, 2011, pp. 335–344.
[4] J. Fiske, D. Grissom, and P. Brisk, "Exploring speed and energy tradeoffs in droplet transport for digital microfluidic biochips," in *IEEE ASPDAC*, 2014, pp. 231–237.
[5] W.-L. Chou *et al.*, "Recent advances in applications of droplet microfluidics," *Micromachines*, vol. 6, no. 9, pp. 1249–1271, 2015.
[6] H. Chen, S. Potluri, and F. Koushanfar, "Biochipwork: reverse engineering of microfluidic biochips," in *IEEE ICCD*, 2017, pp. 9–16.
[7] C.-W. Hsieh, Z. Li, and T.-Y. Ho, "Piracy prevention of digital microfluidic biochips," in *IEEE ASPDAC*, 2017, pp. 512–517.
[8] G. Wang, D. Teng, and S.-K. Fan, "Digital microfluidic operations on micro-electrode dot array architecture," *IET Nanobiotechnology*, vol. 5, no. 4, pp. 152–160, 2011.
[9] Z. Zhong *et al.*, "Micro-electrode-dot-array digital microfluidic biochips: Technology, design automation, and test techniques," *IEEE TBioCAS*, vol. 13, no. 2, pp. 292–313, 2018.
[10] T.-C. Liang *et al.*, "Execution of provably secure assays on meda biochips to thwart attacks," in *ACM ASPDAC*, 2019, pp. 51–57.
[11] K. Y.-T. Lai *et al.*, "An intelligent digital microfluidic processor for biomedical detection," *Journal of Signal Processing Systems*, vol. 78, no. 1, pp. 85–93, 2015.
[12] Y. Ho *et al.*, "Design of a micro-electrode cell for programmable lab-on-CMOS platform," in *IEEE ISCAS*, 2016, pp. 2871–2874.
[13] Z. Zhong, Z. Li, and K. Chakrabarty, "Adaptive error recovery in micro-electrode-dot-array biochips based on droplet-aliquot operations and predictive analysis," in *IEEE ICCAD*, 2017, pp. 615–622.
[14] J. L. Gross and J. Yellen, *Graph theory and its applications*. Chapman and Hall/CRC, 2005.
[15] J. L. Jacobsen, "Exact enumeration of Hamiltonian circuits, walks and chains in two and three dimensions," *Journal of Physics A: Mathematical and Theoretical*, vol. 40, no. 49, p. 14667, 2007.
[16] O. Bodroza-Pantic *et al.*, "Enumeration of hamiltonian cycles in some grid graphs," *MATCH Commun. Math. Comput. Chem*, vol. 70, no. 1, pp. 181–204, 2013.
[17] F. Su and K. Chakrabarty, "High-level synthesis of digital microfluidic biochips," *ACM JETC*, vol. 3, no. 4, pp. 1–32, 2008.
[18] M. Ibrahim, K. Chakrabarty, and U. Schlichtmann, "Synthesis of a cyberphysical hybrid microfluidic platform for single-cell analysis," *IEEE TCAD*, 2018.
[19] M. Ibrahim, K. Chakrabarty, and K. Scott, "Synthesis of cyberphysical digital-microfluidic biochips for real-time quantitative analysis," *IEEE TCAD*, vol. 36, no. 5, pp. 733–746, 2017.
[20] P. Subramanyan *et al.*, "Evaluating the security of logic encryption algorithms," in *AC IEEE HOST*, 2015, pp. 137–143.
[21] A. Ignatiev, A. Morgado, and J. Marques-Silva, "PySAT: A Python toolkit for prototyping with SAT oracles," in *SAT*, 2018, pp. 428–437. [Online]. Available: https://doi.org/10.1007/978-3-319-94144-8_26
[22] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of data encryption standard," in *IEEE ITC*, 2004, pp. 339–344.
[23] S. Paul, R. S. Chakraborty, and S. Bhunia, "Vim-scan: A low overhead scan design approach for protection of secret key in scan-based secure chips," in *IEEE VTS*, 2007, pp. 455–460.
[24] J. Lee *et al.*, "Securing scan design using lock and key technique," in *IEEE DFTVLSI*, 2005, pp. 51–62.
[25] D. Hely *et al.*, "Scan design and secure chip," in *IEEE IOLTS*, vol. 4, 2004, pp. 219–224.
[26] A. Baumgarten, A.and Tyagi and J. Zambreno, "Preventing ic piracy using reconfigurable logic barriers," *IEEE Design & Test of Computers*, vol. 27, pp. 66–75, 2010.
[27] J. A. Roy, F. Koushanfar, and I. L. Markov, "Ending piracy of integrated circuits," *Computer*, vol. 43, no. 10, pp. 30–38, 2010.
[28] J. Rajendran *et al.*, "Security analysis of logic obfuscation," in *IEEE DAC*, 2012, pp. 83–89.

APPENDIX

Proof of Theorem 1: This proof is based on some lemmas, which we prove first. Let be the droplet location vector in observation , be the scanned-out bitstream in observation , and be a legal key matrix returned by the SAT solver.

Lemma 1. *If the enhanced daisychain is used and , then .*

Proof. The proof is straightforward because the bits associated with a specific microelectrode are randomized in the scanned-out bitstreams using the enhanced daisychain across the th and the th observations. Therefore, if , i.e., the on-chip droplets are not moved for the th and the th observations, the scanned-out bitstreams . □

Lemma 2. *If can be partitioned into two vectors and ( ), then can also be partitioned into two vectors and ( ), where and .*

Proof. Consider the following relationships:

(1)

Also, we know that

(2)

From (1) and (2), we get . □

Lemma 3. *, where is the matrix multiplication of and .*

Proof. Let be the microelectrode set that a droplet occupies at cycle . Assuming the droplet exists across observations and . We know . Therefore, , where the vector is another representation for the droplet locations. □

Theorem 1. *When we adopt the enhanced daisychain, the SAT attack cannot unravel the structure of the daisychain based on the continuous observations of bioassay execution.*

Proof. We employ proof by contradiction. Assume that a legal key is returned by the SAT solver that attacks the enhanced daisychain. Therefore, for any two observations and , and . Based on Lemma 3, , where . We can express and as and , respectively, where $\mathbb{B}$ and $\mathbb{B}$. According to Lemma 2, and . We find out that , i.e., for the same from two observations and , the scanned-out bitstreams are the same. This contradicts Lemma 1. □