Thwarting Bio-IP Theft Through Dummy-Valve-Based Obfuscation

Mohammed Shayan, *Member, IEEE*, Sukanta Bhattacharjee, *Member, IEEE*, Ajymurat Orozaliev, Yong-Ak Song, Krishnendu Chakrabarty, *Fellow, IEEE*, and Ramesh Karri, *Fellow, IEEE*

***Abstract*— Researchers develop bioassays following rigorous experimentation in the lab that involves considerable fiscal and highly-skilled-person-hour investment. Previous work shows that a bioassay implementation can be reverse-engineered by using images or video and control signals of the biochip. Hence, techniques must be devised to protect the intellectual property (IP) rights of the bioassay developer. This study is the first step in this direction and it makes the following contributions: (1) it introduces the use of a dummy valve as a security primitive to obfuscate bioassay implementations; (2) it shows how dummy valves can be used to obscure biochip building blocks such as multiplexers and mixers; (3) it presents design rules and security metrics to design and measure obfuscation. In our preliminary work, we presented the concept through the use of sieve-valve as a dummy-valve. However, sieve-valves are difficult to fabricate. To overcome fabrication complexities, we propose a novel multi-height-valve as an obfuscation primitive. Moreover, we showcase the suitability of multi-height-valve for obfuscation through COMSOL simulations. We demonstrate the practicality of the proposal by fabricating an obfuscated biochip using multi-height valves. We assess the cost-security trade-offs associated with this solution and study the practical implications of dummy-valve based obfuscation on real-life biochips.**

***Index Terms*— Microfluidics, security, intellectual property (IP), reverse engineering.**

I. Introduction

A BIOCHIP platform integrates complex laboratory operations into a small chip of few square centimeters in size. It has revolutionized biochemical applications such as point-of-care diagnostics [2], DNA purification [3], and biomedical research [4]. The microfluidics market was valued at $8.28 billion in 2017, and it is expected to grow at a compound annual growth rate of 22.6% to reach $27.91 billion by 2023 [5]. Due to rapid commercialization and deployment, intellectual property (IP) piracy has become financially rewarding [6]. Therefore, protecting bioassay IPs is of paramount importance to its developers.

Pharmaceutical companies invest large sums of money and person-hours in a slow and expensive drug development process laced with tough regulations. This process is prone to stealing of sensitive research data [7]. In 2016, two scientists at a leading pharmaceutical company were indicted for colluding with a competitor to steal promising drug research secrets [8]. For rapid and low-cost drug development, pharmaceutical companies are using various types of microfluidic biochips that minimize the assay time and reagent requirement [9].

Continuous flow-based microfluidic biochips (CFMBs) have evolved rapidly in the last decades [4], [10]. The CFMBs allow fluid flow in a network of micro-channels made of PDMS material [10]. This fluid flow can be automatically controlled by suitable (de)pressuring of micro-valves. Such controlled fluid flow is used to mimic fluidic operations like mixing, incubating, filtering and, washing. A bioassay is implemented as a sequence of such fluidic operations [11]. Previous work has shown that a bioassay implementation on a CFMB can be reverse engineered using biochip images and actuation sequence [12].

Biochips can also advance point-of-care diagnosis [13]. The response to the COVID-19 pandemic was limited in many places by inadequate testing resources and trained personnel [14]. An automated, low-cost platform, such as a microfluidic biochip, can help in overcome this challenge. In fact, Baebies has recently received National Institute of Health Rapid Acceleration of Diagnostics (RADx) funding for COVID-19 research using microfluidic biochips [15]. However, the wide-scale adaptation of biochips for clinical diagnosis will only be possible if the stakeholders find it trustworthy.

A. Motivation

We demonstrate IP piracy through a bioassay implementation on a CFMB (Fig. 1). The platform consists of a multiplexer that selects from two input reagents R_1 and R_2 and uses a rotary mixer to mix them in the desired ratio [16]. Fluidic operations corresponding to a bioassay are mapped to

Manuscript received May 17, 2020; revised August 23, 2020, October 12, 2020, and December 11, 2020; accepted December 17, 2020. Date of publication December 28, 2020; date of current version February 1, 2021. This research is supported in part by the Army Research Office under grant number W911NF-17-1-0320, NSF Award numbers CNS-1833622 and CNS-1833624, NYU Center for Cyber Security (CCS), and NYU Abu Dhabi Center for Cyber Security (CCS-AD). This article was presented in the Proceedings of DATE 2019. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ulrich Rührmair. *(Corresponding author: Mohammed Shayan.)*

Mohammed Shayan and Ramesh Karri are with the Department of Electrical and Computer Engineering, New York University, Brooklyn, NY 11201 USA (e-mail: mos283@nyu.edu; rkarri@nyu.edu).

Sukanta Bhattacharjee is with the Computer Science and Engineering Department, Indian Institute of Technology, Guwahati 781039, India (e-mail: sukantab@iitg.ac.in).

Ajymurat Orozaliev and Yong-Ak Song are with the Biomedical Engineering Department, New York University Abu Dhabi, Abu Dhabi, United Arab Emirates (e-mail: ajymurat.orozaliev@nyu.edu; rafael.song@nyu.edu).

Krishnendu Chakrabarty is with the Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708 USA (e-mail: krish@ee.duke.edu).

Digital Object Identifier 10.1109/TIFS.2020.3047755

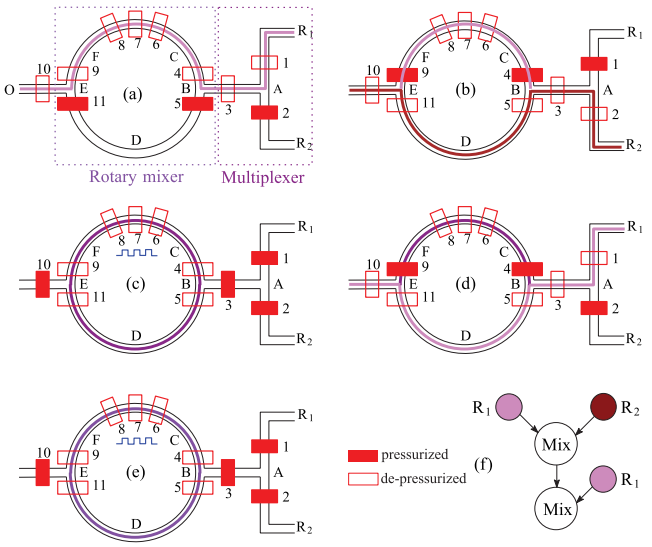


Fig. 1. A bioassay implementation: (a) Push reagent R_1 into the upper half of the mixer. (b) Push reagent R_2 into the lower half of the mixer. (c) Mix. (d) Push R_1 into the lower half of the mixer. (e) Mix. (f) The sequencing graph inferred from the images.

a sequence of actuation steps for controlling the valve state. Let us illustrate the bioassay execution on the CFMB in which all valves are initially closed. The first set of actuations fills R_1 in the upper half of the mixer (Fig. 1(a)). Next, R_2 fills the lower half of the mixer (Fig. 1(b)). The valves $6, 7, 8$ are activated in a sequence to form a peristaltic pump that circulates the fluid in the rotary mixer, producing a mixture of R_1 and R_2 in 1:1 ratio (Fig. 1(c)). Next, the lower half of the mixer is replaced with R_1 (Fig. 1(d)), and the peristaltic pump is activated (Fig. 1(e)). The resulting fluid contains R_1 and R_2 in a 3:1 ratio (Fig. 1(f)).

Fig. 1 shows the one-to-one mapping between the actuation sequence and fluidic operations. It can be inferred from the actuation sequence and biochip layout that the bioassay mixes two input fluids in a 3:1 ratio [12]. The corresponding sequencing graph (IP) is shown in Fig. 1(f). The mixing time can also be determined from the actuations. This example demonstrates the ease with which the bioassay description and its parameters can be reverse-engineered [12].

B. Preliminary Work

To thwart the reverse-engineering of bioassays, we need to obfuscate the one-to-one mapping between the actuation sequence, biochip layout, and fluidic operations. This can be achieved by careful insertion of sieve-valves in the biochip [1]. In other words, the use of sieve-valves (dummy) along with normal valves obfuscates the biochip layout and the actuation sequence. Without the knowledge of the type of the valve (normal/sieve), fluidic operations cannot be determined [1]. However, there are significant challenges associated with the fabrication of a sieve-valve. In particular, a sieve-valve requires the careful modification of the flow channel [1]. It requires the use of a negative photoresist to create a rectangular cross-section channel, whereas normal valves require positive photoresist to create a rounded cross-section channel. This requirement leads to a corresponding increase in fabrication complexity, which might not be acceptable in practice.

C. Contributions

There is clearly a need to devise a practical method that protects the bioassay IP implementation on biochips. To overcome the fabrication complexity, we propose the use of a new multi-height-valve. Our contributions are summarized as follows:

- We designed multi-height-valve as an obfuscation primitive, which can be easily fabricated by controlling the photoresist spin speed or the number of spin iterations [17]. This ease of fabrication is essential for scaling of the number of obfuscation primitives in the biochip.
- We simulate the suitability of multi-height-valve for obfuscation using COMSOL tool.
- We showcase different types of obfuscation using these primitives:
  - Structural: information on what building blocks a biochip is composed of.
  - Behavioral: information on how each block is operating, e.g., which fluid is selected by a multiplexer.
  - Parametric: information of the reagent volume and mix-time.
- We develop design rules, security metrics, and cost trade-offs for an obfuscated biochip. We also discuss special cases that reduce the attacker's reverse engineering effort.
- We demonstrate the practicality of the obfuscation method by fabricating an obfuscated biochip using multi-height valves and applying the proposed method to a real-life biochip benchmark (see appendix). To the best of our knowledge, our work is a first that expands simulation-level IP security work to prototype fabrication for microfluidic biochips.

The rest of the paper is organized as follows. Section II provides the relevant background and dummy-valve primitives are discussed in Section III. The threat model for bio-IP security is presented in Section IV. Section V describes the use of sieve-valves to achieve bioassay obfuscation. Section VI develops the metrics and design rules associated with the sieve-valve based obfuscation. Section VIII provides experimental results of obfuscation applied to real-life biochips, and Section IX concludes the paper.

II. Background

In this section, we present the background on CMFBs, its fabrication, and bioassay implementation.

A. Continuous-Flow-Microfluidic Biochips

CFMB consists of two layers of permanently etched microchannels called the flow and the control layer, as shown in Fig. 2(a). At the intersection of the two layers, a "valve" is formed. An external pressure source can control this valve. When the valve is pressurized, the flexible membrane of the control layer deflects deep into the flow layer blocking the fluid flow (Fig. 2(b)). By opening/closing of the valves, complex

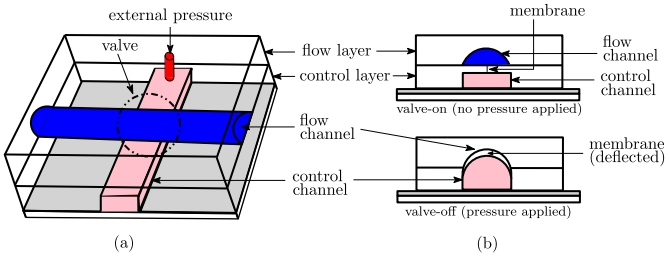


Fig. 2. Schematic of a two-layer microfluidic device: ordinary valve (a) top view and (b) cross-section view.

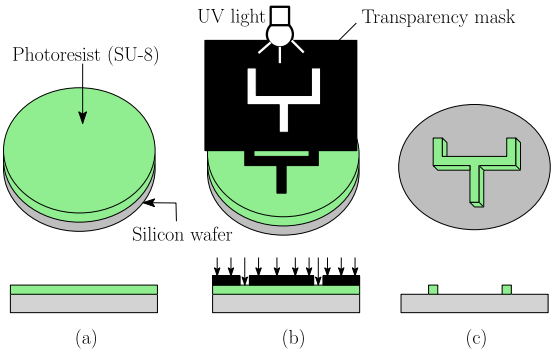


Fig. 3. CFMB fabrication steps (cross-section view): (a) Spin coating negative photoresist on a silicon wafer. (b) Exposing to UV light through a transparency mask. (c) Removal of the non-cross-linked photoresist from the wafer.

fluid handling operations such as mixing, incubation, transportation, and storage can be performed [10]. Advancement in multi-layer soft lithography techniques enables thousands of valves to be integrated into a tiny chip [10]. We showcase a sample CFMB manufactured at our lab in Fig. 4. This biochip can be used to realize the bioassay described in Section I-A. We demonstrate the biochip working in the video link.[1] Next, we describe its fabrication procedure.

B. CFMB Fabrication

The microchannel patterns for individual layers are transferred to the silicon wafer using a photolithography process. In this process, a negative photo-resist (e.g., SU-8) is spin-coated to the desired thickness on the silicon wafer (Fig. 3(a)). Then, the wafer is exposed to the UV light through a transparency mask (Fig. 3(b)). The exposed part of the photo-resist to the UV becomes cross-linked. The unexposed photo-resist film remains soluble and is washed to transfer the desired pattern on the silicon wafer (Fig. 3(c)).

Soft lithography is used to transfer flow and control channel patterns from the silicon wafers to the polydimethylsiloxane (PDMS). After the two PDMS layers are cast separately, they are aligned and bonded irreversibly. At the intersection of the two layers, a 'valve' is formed (see Fig. 2(a)). A pressurized valve closes the flow layer by deflecting the flexible membrane of the control layer deep into the flow layer (see Fig. 2(b)).

C. Bioassay Implementation

Benchtop bioassays are realized on the biochip platforms described above using the following steps [18]:

[1] https://youtu.be/dE8M4xWERbY

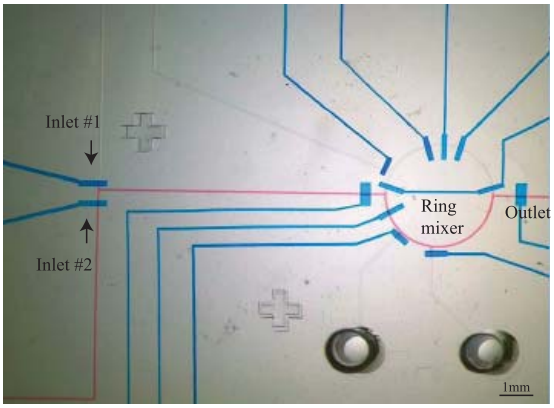


Fig. 4. A continuous-flow biochip with two inlets, one ring-mixer and one outlet.

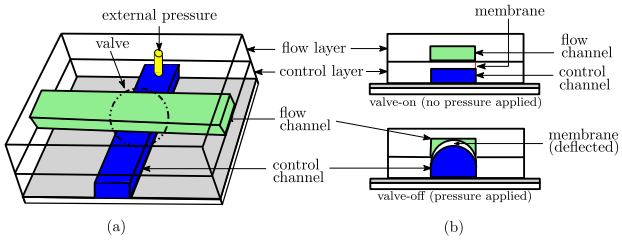


Fig. 5. Schematic of a sieve valve: (a) top view and (b) cross-section view.

1) Setting of the outcome objectives of the bioassay, such as execution time, output quality, and reagent wastage.
2) Re-imagining the benchtop operations such as washing, filtering, culturing, and mixing.
3) Determining appropriate values for parameters to attain the desired objectives. The developer, through many trials, determines the parameter value's range, which includes mixing time, incubation time, mixing ratio, reagent volume, and concentration.

Thus, bio-protocol development requires a systematic understanding of the interplay between numerous parameters, and it is unraveled through experimental iterations. Further, a bioassay implementation needs to overcome manufacturing defects that contribute to operation-time failures. To expose such failures and to facilitate error recovery, the biochip cyber-physical system incorporates one or more sensors [19]. The sensor feedback control entails sensing the quality of the assay outcome at various stages. Based on the sensor data, control-flow decisions are made. The intermediate bioassay outputs are verified against quality criteria; based on this verification, the relevant bioassay steps are repeated.

III. Obfuscation Primitive

Here, we describe the structure of obfuscation primitives - sieve-valve and multi-height-valve. Next, we explain why we choose the latter primitive for obfuscated biochip fabrication. We provide simulation and experimental results to establish the efficacy of the multi-height-valve as an obfuscation primitive.

A. Sieve Valve

In a normal valve, the flow channel is semi-circular shaped. When the valve is pressurized, it seals the flow channel

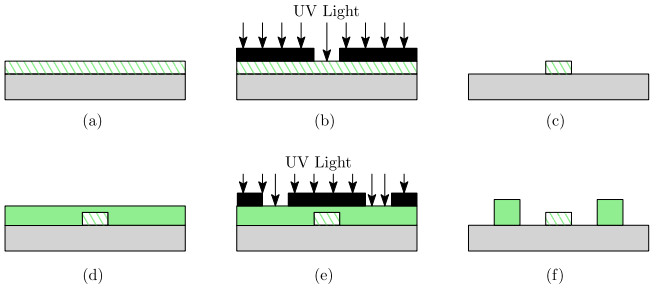


Fig. 6. Fabrication of multi-height structure: a cross-section view of (a) spin coating with photoresist (b) UV exposure (c) removal of uncured photoresist (d) spin coating with photoresist (e) UV exposure (f) removal.

(Fig. 2(b)). However, if the flow channel is rectangular, the pressurized valve membrane partially closes the flow channel, as shown in Fig. 2(d). This is the sieve valve (Fig. 2(c)) [1]. These are used in CFMBs to trap cells. Closing the sieve valve blocks the cells but allows the fluid to pass through [20]. Fabrication of rectangular flow channel requires a negative photoresist instead of a positive photoresist, which is used for normal valve fabrication.

B. Multi-Height Valve

The amount of pressure needed to open or close a valve is determined by the membrane thickness in the valve region [21]. If the height of the control channel is lowered, then it results in a thicker membrane, as shown in Fig. 6. This requires a higher pressure to operate (close/open) compared to the normal membrane. When it is operated at a lower pressure, the valve does not close/open completely [17], [22]. For example, the work in [17] shows that a 34 μm membrane valve requires a minimum pressure of 12 psi to operate, whereas a 28 μm membrane requires a minimum pressure of 8 psi to operate.

C. Ideal Obfuscation Primitive

An ideal obfuscation primitive would be easy to integrate with the biochip and hard to differentiate from a normal valve. A sieve-valve results in a discontinuity in the flow-layer shape from semi-circular (normal) to rectangular. While biochips have only a few sieve valves [20], obfuscation requires the insertion of one extra sieve-valve for every normal-valve. This leads to a corresponding increase in fabrication complexity. On the other hand, the fabrication of a multi-height-valve requires modification of the height of the control valve. This can be easily achieved by simply controlling the photoresist viscosity or spin speed or number of spin iterations. In other words, multi-height-valves are easier to integrate with a biochip and easier to scale up in terms of their numbers than sieve-valves.

Next, we establish the efficacy of the multi-height-valve in acting as an obfuscation primitive. We performed COMSOL simulation of the valve behaviour for different heights. We describe this in Appendix. We also present experimental evidence through a prototype multi-height-valve fabrication and subsequent demonstration, as shown in Fig. 7. Using this,

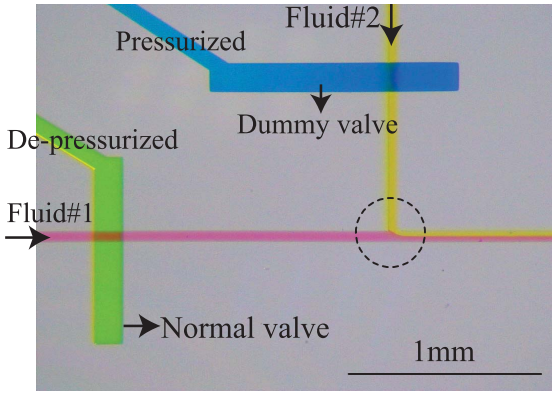


Fig. 7. A multi-height valve as an obfuscation primitive. The normal and dummy valve are both open even though the normal valve is de-pressurized, and the dummy valve is pressurized. As a result both the fluids #1 and #2 are flowing, as shown in the dotted circle.

we were able to obfuscate the sample biochip shown in Fig. 4. The similar images of the valves provide validation for the efficacy of multi-height valves as an obfuscation primitive. We also provide a link to a video, that demonstrates that there is no obvious visible difference between the operation of a dummy and a normal valve.[2] We describe this in more detail in Section VIII.

IV. Bio-IP Security

Having provided the background of CFMB, we now present the context of CFMB application, threat model, and relevant previous work.

A. Application Context

Consider a bioassay developer who invests heavily in bioassay IP development. Such a developer can be either a pharmaceutical company performing drug trials or developing personalized medicine for its customers. The developers use microfluidic platforms, manufactured in-house, to conduct large-scale (high-throughput, parallel, and automated) experiments [4]. The biochip controller is connected to the network for round-the-clock online monitoring and control [23]. The user can focus the CCD camera on the area of interest to verify the biochip's state. We discuss one such context in the below example.

Example 1: The CFMB showcased in Fig. 4 is used for evaluating the response of the particular dose of a drug on the bacterial cells. The drug is loaded through one of the inlets, and a dilution buffer can be loaded through the other inlet. The drug is diluted through a sequence of load and mix operations. The resulting diluted drug is then injected into a chamber of living cells, and its response is recorded. This experiment is repeated for various concentrations to find the minimum drug concentration required to kill the bacterial cells.

B. Threat Model

An attacker can be a competitor who is motivated to steal the IP from the developer without incurring any cost of development. Thus, the attacker gains economically by accessing

[2]https://youtu.be/RS3fRLoQwSQ

such high-value IP, i.e., the bioassay. To reverse-engineer the bioassay, the attacker accesses the actuation sequence and the snapshots of the biochip layout. The attacker can then map the actuation sequence to fluidic operations and rebuild the bioassay, as shown in Section I-A. The attacker accesses these through a network attack. A recent ransomware cyberattack in the pharmaceutical industry underlines the seriousness of this threat [24].

The biochip controller is connected to the network for around-the-clock online monitoring and control [23]. Further, the actuation sequence is stored in the biochip controller to support online error correction [25]. A remote attacker can gain access to the network [26], [27]. Then the attacker can launch one of the two attack levels

1) The attacker uses the network for a short duration to access a single image of the biochip layout and the actuation sequence of the bioassay. It is more likely that the attacker can escape detection for a short duration. This becomes more practical when the bioassay process is not being actively monitored.
2) The attacker logs on to the network through the entire period of bioassay execution to gain access to (high- or low-resolution) images of all stages the execution as well as the actuation sequence. This increases the chances of detection of unauthorized access.

The attacker can differentiate between the pressurized valve and de-pressurized valve due to the visual difference. The attacker does not have access to the CFMB. However, the attacker can build a prototype from the biochip layout snapshots. Such a prototype can be used to remove any ambiguity left in the reverse-engineering process.

The proposed solution comprehensively addresses the first type of attack, wherein the attacker has access to the biochip layout image and the actuation sequence. Note that the type-1 attack is more realistic as it requires only a momentary access to the network. On the other hand, type-2 attack requires network access throughout the duration of the bioassay, which can be as long as couple of hours.

An attacker can launch the second type of attack to deduce the bioassay from secondary effects of fluid flow through channels, such as the widening of channel walls due to fluid flow. However, this requires the camera that is monitoring the bioassay to be of extremely high precision so that it can capture these secondary effects. Further, the sensors are focused on the area of interest (mixer or culture chamber) so attacker cannot have access to images of all valves. In other words, the security problem is reduced to balancing between signal processing capabilities and secondary effects of fluid flow. The defender can reduce the secondary effects by controlling the applied pressure. The simulation of the observable secondary effects requires the modeling of fluid-flow pressure and the signal processing setup. Such a study can shed more light on the type-2 attack and the adaptation of the proposed solution. We will address this issue as part of future work.

C. Related Prior Work

An assessment of IP threats in the supply chain due to the distributed microfluidic design flow is presented in [6]. The threats include overbuilding, reverse-engineering, and counterfeiting of biochips. Watermarking of bioassay was introduced to protect the IP rights of the developer [18]. A watermark serves as proof-of-ownership in a court of law. However, this does not prevent reverse-engineering. A bioassay locking scheme was proposed to obfuscate the sequencing graph description of the bioassay [28].

Bioassay locking defends against an overproduction attack, such that an untrusted foundry cannot overproduce biochip hardware and sell it for profit. However, this technique does not prevent or resist reverse-engineering of a bioassay from the corresponding actuation sequence. In other words, bioassay locking and watermarking are measures that protect the IP rights of a commercial product designer, whereas the proposed method prevents reverse-engineering of a bioassay implemented in a lab. Recall that we are considering a network-based attack model. Network-based solution such as multi-factor authentication can be used to secure the network. However, the recent spate of cyber-attacks on the pharma and medical network infrastructure motivates us to look for non-network security solutions [29], [30]. Our solution does not replace the network security solutions but augments them.

A method for camouflaging the biochip layout by inserting extra valves and channels is reported in [12]. However, this approach fails as an attacker can reverse-engineer the IP by combining the actuation sequence and biochip layout. Another potential defense is to add extra actuations on idle valves to confuse the attacker. However, the attacker will be able to reverse-engineer the bioassay, albeit with the added extra operations. In other words, the actual bioassay (consequential part) is not hidden but only extra actuations (inconsequential) are added. Further, the attacker can easily discard these spurious actuations.

An orthogonal approach to our work can be to use 3D microfluidic design technique to obfuscate by distributing the design over multiple layers [31]. In fact, dummy valve-based obfuscation can be applied to each layer of a 3D microfluidic design to harden the design against reverse-engineering. However, 3D fabrication is more time-consuming and labor-intensive, requiring multiple lithography steps and precision alignment [32].

V. Obfuscation for IP Protection

To deter the reverse-engineering of a bioassay, we propose to obfuscate the actuation sequence by carefully inserting sieve or multi-height valves in the biochip. In the rest of the paper, we refer to a 'sieve or multi-height' valve as a 'dummy' valve. The bioassay developer keeps the bioassay description and the dummy-valve locations a secret. The developer uses a CAD tool on a trusted offline computer to synthesize the obfuscated actuation sequence. The obfuscated sequence is loaded in the biochip controllers that are used to conduct the high-valued-experiments, as shown in Fig. 8. Minor software updates are handled in the biochip controller, and major updates are performed in the trusted offline computer.

Note that the insertion of only dummy valves or only dummy actuations fails to to protect the IP. If only dummy

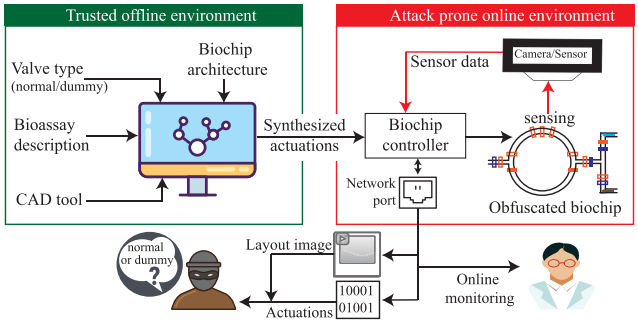


Fig. 8. Proposed dummy-valve-based obfuscation technique.

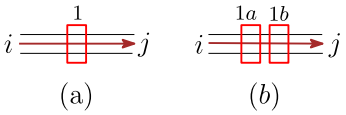


Fig. 9. Channel $i \rightarrow j$ with (a) valve 1 and (b) valves $1a$, $1b$.

TABLE I
BOOLEAN VARIABLES TO CHARACTERIZE A CHANNEL

Parameter	Description	Interpretation	
		1	0
v_a	Status of a valve	pressurized	de-pressurized
g_a	Type of a valve	normal	dummy
$c_{ij}^k, k \in \mathbb{N}$	Status of channel $i \rightarrow j$	open	closed

k: #valves on the fluidic channel connecting ports i and j.

valves are added, then attacker can reverse engineer the IP by using the actuation sequence. If there are no dummy valves but only dummy actuations, then the attacker's problem is to map the real actuations to the biochip valves. Actuations are the sequence of control signals applied for the valves. The biochip controller converts the stored actuations to the electrical signals to the valves. Even with a momentary access to the controller, an attacker can easily prune the real actuations from the dummy actuations by observing the electrical signals provided to the valve. To provide a defense against this attack, we need to insert dummy valves in the biochip.

A. Dummy-Valve-Based Obfuscation

Consider the channel between port i and j, as shown in Fig. 9(a). Let the channel $i \rightarrow j$ be open if the valve 1 is pressurized, else it is closed. Such a valve is a normal valve. On the other hand, if the valve is a dummy, then the channel $i \rightarrow j$ is always open, regardless of the actuation state of valve 1. To capture the differences between a normal and a dummy valve, consider the Boolean variables defined in Table I. Using these variables, we describe the channel in Fig. 9(a) as:

$$c_{ij}^1 = \overline{g_1} \vee (g_1 \wedge v_1) = \overline{g_1} \vee v_1 \tag{1}$$

Here, '$\wedge$', '$\vee$', and '$\overline{g_1}$' represent Boolean operations 'and', 'or', and 'negation', respectively. As per the attack model, g_1 is secret, and v_1 is known from the actuation sequence. Equation (1) captures the obfuscation introduced in the fluid channel characteristics due to the unknown valve type. Without the knowledge of g_1, an attacker does not know the channel status. The dummy valve reduces the flow-rate in the channel.

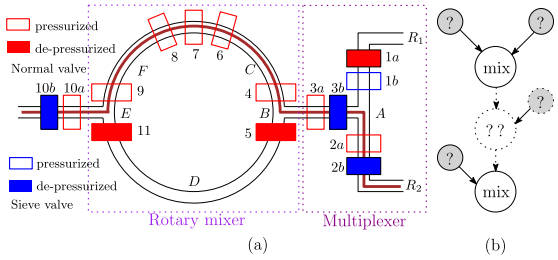


Fig. 10. (a) Dummy-valve-based obfuscated functional modules of mixer and multiplexer. (b) The obfuscated sequencing graph.

However, this can be neutralized by either The by increasing the fluid pressure at the inlet or allowing more time for the fluid flow. For the purposes of our analysis, we ignore the change in flow-rate. Consider an increase in the number of valves on the channel, as shown in Fig. 9(b). The characteristic of the channel is given by following Boolean equation,

$$\begin{aligned} c_{ij}^2 &= (\overline{g_{1a}} \vee v_{1a}) \wedge (\overline{g_{1b}} \vee v_{1b}) \\ &= (\overline{g_{1a}} \wedge \overline{g_{1b}}) \vee (\overline{g_{1a}} \wedge v_{1b}) \vee (\overline{g_{1b}} \wedge v_{1a}) \vee (v_{1a} \wedge v_{1b}) \end{aligned} \tag{2}$$

If there are n such valves on a channel $i \rightarrow j$, the characteristic of the channel can be captured as

$$c_{ij}^n = \bigwedge_{\gamma=1}^{n} (\overline{g_\gamma} \vee v_\gamma) \tag{3}$$

Comparing Equation (2) and Equation (3), increasing the number of valves increases the channel obfuscation due to the corresponding increase in the number of unknown parameters (g_*). Using this primitive, we describe the obfuscation of the reagent load operation, the biochip structure, and the bioassay parameters such as mix-time and reagent volume.

B. Reagent Load Obfuscation

A biochip consists of functional modules such as a fluid inlet/outlet, mixer, storage, reaction chamber, and multiplexer/demultiplexer. As shown in Section I-A, the actuation signals of a biochip have a one-to-one mapping to the fluidic operations. We insert dummy valves in the biochip functional modules so that the actuation-signal to fluidic-operation mapping is no longer preserved. Since the valve type is kept secret, the channel characteristic can be obfuscated, as shown in Equation (3). Thus, the attacker cannot determine the fluidic operations correctly to reverse-engineer the sequencing graph (IP). This is called *behavioral obfuscation*.

Consider the biochip shown in Fig. 1 with a two-input multiplexer and a rotary mixer. It mixes two input reagents R_1 and R_2 in a 3 : 1 ratio, as explained in Section I-A. Additional valves (normal and dummy) are added to obfuscate the biochip, as shown in Fig. 10(a). In the modified CFMB platform, one or more dummy valves on the input-to-output paths can be de-pressurized to deceive the attacker from identifying the correct fluidic path. From Equation (3), the channel state (open/close) depends on the valve type (g_*), which is unknown to the attacker. The following example illustrates obfuscation on the fluidic path.

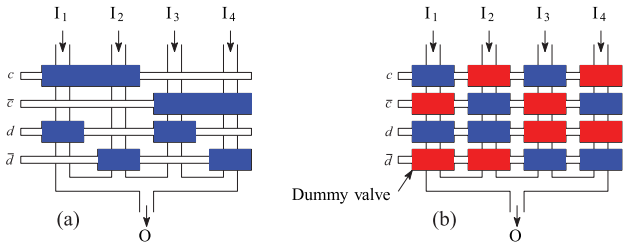


Fig. 11. (a) A 4-inlet binary multiplexer design. For example, inlet I_1 flows to output 'O' when control lines c and d are depressurized. (b) Obfuscated binary multiplexer.

Example 2: In Fig. 10(a), let $\{1b, 2b, 3b, 10b\}$ *be dummy valves and the rest be normal valves, i.e.,* $g_{1b}, g_{2b}, g_{3b}, g_{10b} = 0000$*, then the actuation set* $v_{1a}, v_{1b}, v_{2a}, \cdots, v_{10a}, v_{10b}, v_{11} = 011010101111110$ *pushes* R_2 *into the mixer (ref. Fig. 10(a)). On the other hand, if* $\{1a, 2a, 3b, 10b\}$ *are dummy valves and rests are normal valves, then the same actuation set will push* R_1 *into the mixer. Without knowing the valve type (dummy or normal), an attacker cannot determine the inputs to the mix operation in a sequencing graph, as shown in Fig. 10(b).*

As the size of the multiplexer increases, the number of ports increases [10]. A variety of schemes are used to enable the scaling of the multiplexer. A binary multiplexer of N (vertical) inlets requires $2\log_2(N)$ (horizontal) control channels. Valves are formed only where a wider section of the (horizontal) control channel intersects the vertical flow channel, as shown in Fig. 11(a). Here, the control channels are binary pairs [10]. In such a case, the multiplexer structure can be easily obfuscated by modifying the thin section of the control channel to a dummy valve, as shown in Fig. 11(b). The advantage of obfuscating a binary multiplexer is that it does not increase the number of control ports.

C. Mix-Time Obfuscation

The mixer in Fig. 10(a) has a ring with one inlet channel $A \rightarrow B$ and an outlet channel $E \rightarrow O$. The mixing time can be deduced from a sequence of opening and closing of the inlet/outlet channels followed by the peristaltic pumping operation. The status of the channels $A \rightarrow B$ and $E \rightarrow O$ can be obfuscated by adding extra valves; viz Equation (3). This leads to ambiguity in the mixing time and the number of mixing steps. The following example describes it in detail.

Example 3: In Fig. 10(a), consider the valve types as in Example 2, i.e., all $g_* = 1$ *except* $g_{1b}, g_{2b}, g_{3b}, g_{10b} = 0000$*. Let the valve actuation be* $v_{1a}, v_{1b}, v_{2a}, \cdots, v_{10a}, v_{10b}, v_{11} = 011010101111100$*. This opens the mixer inlet/outlet to push out the mixer content, which denotes the end of the previous mixing step. If the actuation is followed by a peristaltic pumping operation, then it denotes the start of a new mixing. On the other hand, if* $\{3a, 10a\}$ *are dummy valves, and* $\{3b, 10b\}$ *are normal valves, then the given actuation set does not open the inlet/outlet of the mixer. Hence, the previous mixing step has not ended, and a new mixing step has not started. This leads to obfuscation in the deduction of the mixing time and the number of mixing steps, as shown by the dotted nodes in the sequencing graph in Fig. 10(b).*

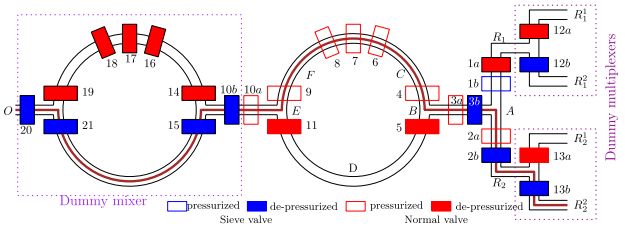


Fig. 12. Structural obfuscation: a dummy multiplexer addition at inlets R_1, R_2 and a dummy mixer addition at outlet O.

D. Structural Obfuscation

The *behavioral obfuscation* does not change the structure of the biochip but inserts extra valves on the existing channels. Furthermore, the structure of the biochip can be obfuscated by inserting dummy channels, multiplexers and/or mixers. This is *structural obfuscation*. A channel can be mimicked by a dummy multiplexer with a dummy valve on the original inlet-so that it is always open and a normal valve on a dummy inlet-which is kept closed. Without the knowledge of the valve type, the attacker cannot know which inlet is selected when both the valves are closed. Alternately, the channel can be mimicked by a dummy mixer with dummy valves forming an always open channel in the ring mixer. The valves of this module are pressurized like a mixing module to mislead the attacker. To resolve this ambiguity, an attacker has to do trial-and-error by replacing each mixing operation in the actuation with a transportation operation.

Example 4: In Fig. 12, ports $12b$*,* $13b$ *are dummy valves and ports* $12a$*,* $13a$ *are normal valves. For actuation set* $v_{12a}, v_{12b}, v_{13a}, v_{13b} = 0000$*, paths* $R_1^2 \rightarrow R_1$ *and* $R_2^2 \rightarrow R_2$ *are open. On the other hand, if ports* $12b, 13b$ *are normal valves and ports* $12a, 13a$ *are dummy valves, then for the same actuation set* $R_1^1 \rightarrow R_1$ *and* $R_2^1 \rightarrow R_2$ *are open. This leads to obfuscation of the fluid selected. Furthermore, a dummy mixer with* $\{15, 20, 21\}$ *as dummy valves is added to path* $E \rightarrow O$*. The valves of this mixer can be pressurized to mimic a normal mixer, whereas in reality, it is a* $E \rightarrow O$ *channel controlled by valve port* $10a$*.*

E. Reagent Volume Obfuscation

A bioassay implementation requires the mixing of reagents in measured quantities. The reagent volume is a key parameter that determines a bioassay's outcome and its precision [33]. The bioassay developer finds the reagent volume parameter through numerous trials on the biochip [34]. A metering block can be used to measure different quantities of reagent before loading them, and we elucidate it through the following example.

Example 5: Consider a metering circuit shown in Fig. 13 (a). The regular 1:1 mixing can be achieved by 1) A fluid can be loaded by opening valves in1, 1-4, 2, and o2. 2) Another fluid can be loaded by opening valves in1, 5-8, and o2. Further, 3:1 can be achieved by 1) A fluid can be

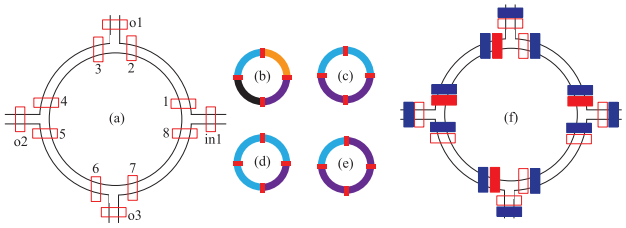


Fig. 13. (a) A mixer with a metering capability. The mixer can be used to mix fluids in different ratios - (b) 1:1:1:1, (c) 1:1, (d) 3:1, and (e) 1:3. (f) A mixer with an obfuscated metering block.

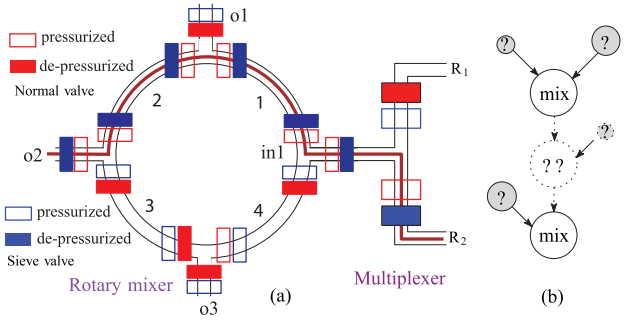


Fig. 14. (a) Obfuscated mixer with a metering circuit. The peristaltic pumps are excluded for simplicity. (b) DAG description of the obfuscated fluid selection and metering volume. The droplet type and volume are unknown.

loaded by opening valves in1, 1-6, and o3. 2) Another fluid can be loaded by opening valves in1, 7-8, and o3. In other words, it can support different mixing ratios such as 1:3, 3:1, 1:1 of two input fluids, as shown in Fig. 13(b-e).

The mixing ratio can be estimated through the actuation states of the metering block valves. This information can be obfuscated using additional dummy valves, as shown in Fig. 13(f). If the original biochip does not have a metering block, then the biochip can be modified to mimic a metering block to enhance the obfuscation. This is achieved by dividing the mixer into N equal segments using extra N valves - referred to as ratio valves. An input fluid can be filled in 0 to N of these segments. This metering of fluid can be obfuscated by adding dummy valves alongside the ratio valves. Without the knowledge of the dummy and normal valves, an attacker cannot determine the number of segments filled by a fluid.

Example 6: Consider the bioassay described in Section I-A. We replace the simple mixer with a metering circuit, as shown in Fig. 14(a). With this modification, reagents can be loaded in variable volumes (0, 1, 2, 3, or 4 parts). A targeted mixer with $R_1 : R_2 = 3 : 1$ can be achieved by loading R_1 in 3 parts of the meter and R_2 in one part, followed by a mixing operation. On the other hand, using the simple mixer, we require three load and two mix operations, as described in Section I-A. In other words, meter circuit reduces the number of operations. On the other hand, metering circuit enables obfuscation of reagent volumetric information. Without the knowledge of the valve type, the attacker cannot determine which reagent was loaded and the volume of the reagent. This obfuscation is described by the DAG in Fig. 14(b).

VI. Design for Obfuscation

Having outlined the proposed obfuscation technique, we next define the security metrics that capture the security-cost trade-offs and design-for-obfuscation rules.

A. Undoing the Obfuscation

To reverse-engineer the bioassay, the attacker has to interpret the actuation sequences that are ambiguous due to unknown valve type g_*. Such actuations are referred to as *ambiguous actuations*. The attacker can build a biochip prototype from the snapshots without the correct valve types. Note that the result of biochemical reactions is difficult to predict; developers often rely on experimental trials to determine the results, such as drug trials [4]. Moreover, the adaptation of a benchtop protocol to a biochip requires experimental trials on a prototype [34]. The attacker can use such a prototype as an oracle to resolve the ambiguous actuations by:

1) *Crude attack* By trial-and-error, the attacker can replace the ambiguous actuations until the results of the bioassay on the biochip prototype become identical to the known results (such as the targeted bacteria is killed).
2) *Knowledge-based attack* The attacker can use some of the information from the benchtop protocol to deduce the bioassay. We discuss this issue in more detail in Section VII.

The maximum number of experiments required to resolve this ambiguity is referred to as *resolution effort* $\mathcal{E}$. This is used as a metric of the efficacy of an obfuscation method. The design overhead for obfuscation is defined in terms of extra valves, which in turn may lead to extra pins in the biochip and extra memory for storing the corresponding actuation signals. The biochip designer obfuscates the biochip and its actuation sequence to make reverse-engineering hard enough to deter an attacker. To maximize the resolution effort, we propose the following design rules.

B. Design-For-Obfuscation Rules

In a crude attack, the attacker will try all combinations of g_*. However, a smart attacker will leverage functional properties to prune the search space. To achieve a robust obfuscated design, we frame four design rules.

1) Channel: A continuous channel needs to be formed from input port to output to push a fluid in a CFMB. The attacker tries to identify which input-output path is opened in a given cycle. If there exists an input-output path without any de-pressurized valve, then the actuation is unambiguous to the attacker. Else, the attacker has to guess if any of the de-pressurized valves on the input to output paths is a dummy valve. This leads to the first design rule.

> Rule #1: In an ambiguous actuation, every input to output channel path must have at least one closed dummy valve.

Consider a channel that has n_{chl} de-pressurized valves in an ambiguous actuation cycle. Without knowing the valve

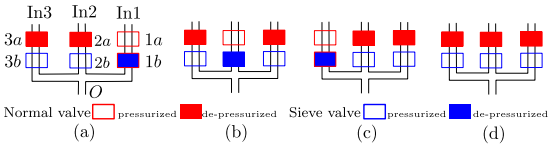


Fig. 15. Multiplexer actuation to push fluid through (a) In1, (b) In2, (c) In3, and (d) no fluid.

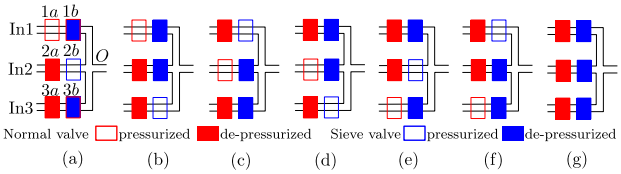


Fig. 16. Multiplexer actuation for (a-b) pushing In1, (c-d) pushing In2, (e-f) pushing In3, and (g) not pushing any fluid.

type g_*, i.e., dummy or normal, the de-obfuscation effort $\mathcal{E}_{chl}$ involves trials that map each closed valve to two possibilities - closed and open. Hence, $\mathcal{E}_{chl} \leq 2^{n_{chl}}$. The effort increases with the number of distinct input-output paths with closed valves in a cycle.

2) Multiplexer: An attacker can use the following properties of a multiplexer to resolve the obfuscation. *P1*: At most, one path of the multiplexer can be open at any time. *P2*: It is likely that each inlet fluid is selected at least once in a bioassay. An attacker can collect all the unique actuations applied to the multiplexer, and along with the properties *P1* and *P2* the attacker can de-obfuscate the multiplexer actuations as discussed in the following example.

Example 7: Consider a 3-inlet multiplexer with two valves $*a$ and $*b$ on each inlet. For each inlet, the set of actuations $v_{*a}, v_{*b} = \{11, 00\}$ is unambiguous, and $v_{*a}, v_{*b} = \{10, 01\}$ is ambiguous. Between any pair of inlets, there are four possible combinations of these ambiguous actuations. In Fig. 15, 3-out-of-4 combinations are used for actuating the valves $v_{1a}, v_{1b}, v_{2a}, v_{2b}$. The unused ambiguous actuation combination on the inlet In1 *and* In2 *is $v_{1a}, v_{1b}, v_{2a}, v_{2b} = 1010$. The attacker can decipher that this actuation opens both inlets* In1 *and* In2 *and hence is not used due to property P1. Alternately, an attacker can guess that the least used actuation on an inlet is used to open the respective inlet. In Fig. 15, actuation $v_{*a}, v_{*b} = 10$ is the least used actuation on each inlet. The attacker can decipher with a high probability that these actuations open their respective inlets due to property P2.*

A naive defense against the above attacks is to increase the number of valves on each inlet. However, this increases cost. To avoid cost escalation, we use two valves per inlet with design rules #2 and #3.

> Rule #2: Apply ambiguous actuations to no more than two inlets at a time. One inlet is the fluid being pushed and one from the other $m-1$ inlets of the multiplexer.

> Rule #3: Apply unambiguous actuations when no fluid is pushed through the multiplexer.

In an m-inlet multiplexer, through these design rules, there are $m-1$ ways of actuating an obfuscated push operation of a fluid. The ambiguous actuation on the same inlet can be used in the obfuscated push operation of other $m-1$ inlets. This defeats the two attacks described in Example 7. The maximum number of unique ambiguous actuations is $s_{mux} = \binom{m}{2}$, as shown in Fig. 16. Each ambiguous actuation can be mapped to two possibilities. If there are s unique ambiguous operations in a bioassay, an attacker needs to perform 2^s experiments. The maximum resolution effort is $\mathcal{E}_{mux} = 2^{s_{mux}} = 2^{\binom{m}{2}}$.

3) Mixer: Reliable mixing requires a minimum mixing time, which depends on the fluid velocity and channel geometries [35]. If ambiguous actuations are inserted in the mixer actuation sequence prior to the minimum mixing interval, then the attacker can map that actuation to an ongoing mix operation and prune the search space. To avoid this, we frame design rule #4.

> Rule #4: The gap between an ambiguous and other mix operations must be more than the minimum mix time, t_{min}.

The number of ambiguous mixer actuations is dependent on the number of valves on the mixer inlet and outlet, provided rule #4 is satisfied. However, to minimize the cost, we use two valves on the mixer inlet (outlet). The number of possible ambiguous actuations on the mixer inlet (outlet) is two. This implies that the maximum number of ambiguous actuations that can be applied to the mixer (inlet and outlet) is $s_{mix} = 4$. The ambiguous actuations can be mapped to one of the two possibilities - a new-mix operation or no new-mix operation. Therefore, the reverse-engineering effort for a mixer is $\mathcal{E}_{mix} = 2^{s_{mix}} = 2^4$.

4) Dummy Structures: The same rules apply to dummy structures such as multiplexers and mixers (Fig. 12). To resolve the ambiguity about n_{dum} dummy structures, ($\mathcal{E}_{dum} = 2^{n_{dum}}$) trial experiments must be performed. However, the cost of introducing dummy structures include not only extra valves but also extra channels and extra input/output ports.

5) Metering Circuits: Consider a metering circuit that loads a fluid in 1 to n_p parts. Such a metering circuit can be obfuscated by adding a dummy valve along with each original valve, i.e., the total number of valves are doubled. In the case of a n_p-part metering mixer, the number of extra valves is $3n_p$. In order to resolve the ambiguity, an attacker needs to determine which of the outlets are open. Since there are n_p outlets, the reverse-engineering effort is $\mathcal{E}_{meter} = 2^{n_p}$. In the case that the original mixer supports only $1:1$ mixing, it has six valves to load upper and lower half, then the extra valves required to build a n_p-part metering mixer is $6n_p - 6$. All the mixer outlets can be merged into a single biochip waste outlet.

VII. Special Cases

The output of a bioassay is the result of the interplay of multiple factors – the biochemistry of the reagents and

parameters such as volume, mix, and incubation time. This complex interplay makes it very difficult to deduce the preceding sequence of operations that lead to a result. Such variability is inherent in a biochemical assay [36]. In this section, we discuss special cases in bioassays that can help the attacker in the reverse-engineering of bio-IP. The following are the cases that aid the attacker.

A. Fluids With Color

The fluids used for bioassays are usually colorless. However, if the fluid is colored, such as blood serum, then the fluid loading path cannot be obfuscated. Note that distinguishing a color fluid depends on color intensity and imaging precision. If a fluid path meets these criteria, then its path needs to isolated from the other obfuscated paths. This is done by avoiding ambiguity in the operations (load, mix) involving the colored fluid. We demonstrate this with an example.

Example 8: Consider a 3-inlet multiplexer. Let inlet fluid #2 is colored, and the other two fluids #1, #3 are colorless. Obfuscating the path of fluid #2 is not only futile but counterproductive. When fluid #2 is loaded, any ambiguous actuation will provide clues to the attacker. Therefore, the path (of colored fluid #2) is left unambiguous. Similarly, the subsequent mix operation cannot be obfuscated, i.e., must be left unambiguous.

B. Fluids With Dispensed Particles

Some inputs have dispensed particles in the fluids. For example, experimental cells or beads are dispensed in a carrier fluid. The loading paths of such fluids have a sieve valve to filter the particles (cells or beads). An additional dummy sieve valve does not obfuscate the loading as the particles could reveal the loading path. Therefore, such a case needs to be treated in the same way as that of the colored fluid path.

C. Fluids With Sequential Order

An attacker without any knowledge of bio-IP will have to perform trial and error in mapping each fluid input operation. However, an expert attacker with the knowledge of benchtop bioprotocol could predict the order of some of the reagents. For example, MNase reagent is used for DNA digestion in cells, whereas SDS/EDTA lysis buffer is used to arrest the digestion. An attacker with the knowledge of the bioprotocol will be able to predict that the reagent MNase will be used first, followed by SDS/EDTA lysis.

The obfuscation scheme needs to take into account these special cases to optimize the obfuscation of the bio-IP. The extra valve insertion needs to done judiciously to maximize the obfuscation effect. Consider a m-inlet multiplexer, let there be x fluids whose order is known, i.e., it is common knowledge that these x fluids are used one after the other. To optimize the obfuscation, we treat the set of x fluids as one fluid. In other words, the net number of inlets is $m - x + 1$ for puroposes of obfuscation analysis. Using the design rules described for multiplexer, the maximum number of unique ambiguous actuations is $s_{mux} = \binom{m-x+1}{2}$, as shown in Fig. 16.

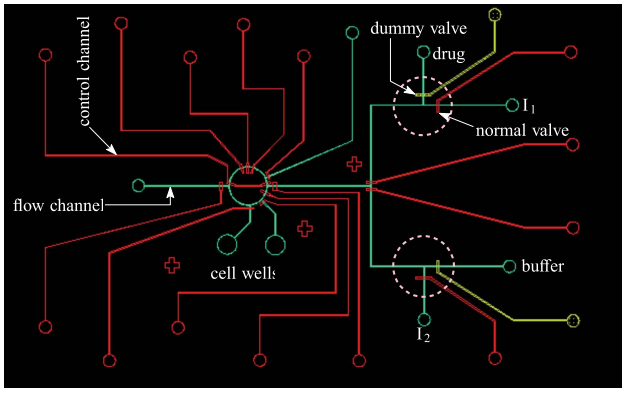


Fig. 17. CAD schematic of a obfuscated biochip: The biochip in Fig. 4 is obfuscated using dummy inlets and dummy valves.

Each ambiguous actuation can be mapped to two possibilities. Therefore, the maximum resolution effort is $\mathcal{E}_{mux} = 2^{s_{mux}} = 2^{\binom{m-x+1}{2}}$.

VIII. Experimental Results

In this section, we describe the fabrication of an obfuscated biochip as a proof-of-concept for the proposal. Next, we analyze the application of the obfuscation techniques on a chromatin immunoprecipitation (ChIP) biochip and other real-life biochips.

A. Fabrication

To demonstrate the practicality of the proposed dummy-valve-based obfuscation, we obfuscated the biochip shown in Fig. 4. A 2-layer microfluidic channel master mold on a 4inch Silicon wafer by using conventional softlithography. We created 1) control master mold with multi-height structure and 2) flow master mold. The detailed fabrication process is described in Appendix. The normal valve has 25 μm thick membrane, whereas the dummy valve was formed with 50 μm thick membrane. For the applied pressure of *25 psi*, normal valve closes completely, whereas dummy valve closes partially, as shown in Fig. 7. The resulting CFMB schematic is shown in Fig. 17. Note that we have inserted two dummy valves and two extra inputs, i.e., we have applied structural obfuscation. Without the knowledge of the valve type, an attacker cannot infer 1) if input operation is performed or not and 2) the correct input fluid loaded. This biochip was used to perform successful drug trials, as explained in Example 1. The lab experimental setup is shown in Fig. 18. Using the obfuscated biochip, sensitive IP (drug concentration) can be protected from theft.

B. Chromatin Immunoprecipitation (ChIP)

The ChIP performs a two-step bioassay: 1) Cell lysis and DNA fragmentation are performed on the sample cells through a series of mixing operations (Fig. 19). This step uses a 5:1 multiplexer that selects cells and reagents being pushed into the mixer Ring-1. 2) The resulting fluid is divided equally into four rings (A-D) to perform ImmunoPrecipitation. These rings are preloaded with anti-body functionalized beads and

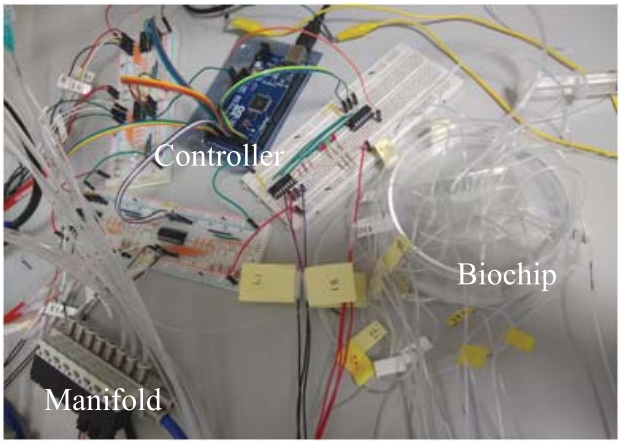


Fig. 18. Experimental setup for drug trial experiment with obfuscated biochip.

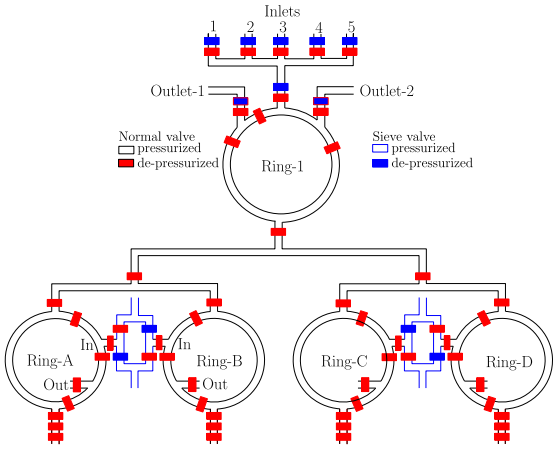


Fig. 19. AutoChIP used for gene enrichment. The extra flow channels are shown in blue color.

mixed with cellular material from step 1. Next, the contents of each of the four rings (A-D) are washed with four different wash buffers (Fig. 20(a)). The washed beads are then moved to micro-centrifuge tubes for qPCR analysis [20]. We apply reagent load, mix-time, and metering obfuscation to step 1 and structural obfuscation to step 2 as follows:

1) Load and Mix-Time Obfuscation: One extra valve is added to each inlet of the 5:1 multiplexer, the ring-1's inlet, and two outlets, i.e., a total of eight extra valves are used, as shown in Fig. 19. This obfuscates the multiplexer selection, the number of mixing operations, and the mixing time. The maximum number of ambiguous actuations applied to the multiplexer and mixer are $s_{mux} = \binom{5}{2}$ and $s_{mix} = 4$, respectively. An attacker's effort in resolving the behavioral obfuscation is $\mathcal{E}_{behav} = \mathcal{E}_{mux} \cdot \mathcal{E}_{mix} = 2^{\binom{5}{2}} \cdot 2^4 = 2^{14}$.

2) Structural Obfuscation: The four ring mixers (A-D) are connected to four fluid inlets that are used to wash the contents of the respective mixer. The inlet channel is replaced by a dummy multiplexer to select between the original wash fluid and a wash fluid corresponding to other mixers, as shown in Fig. 19. This results in eight more valves and four more channels. The effort to resolve the structural obfuscation of $n_{dum} = 4$ multiplexers is $\mathcal{E}_{struct} = 2^{n_{dum}} = 2^4$.

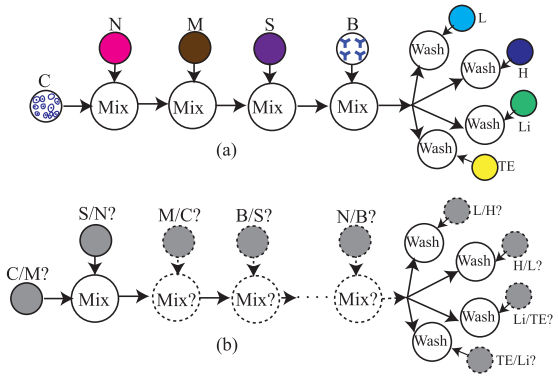


Fig. 20. ChIP bioassay: (a) original and (b) obfuscated. C: Cells under test, N: NP40 buffer, M: Microccal nuclease enzyme, S: SDS/EDTA lysis buffer, B: antibody fictionalized beads, L- Low salt buffer, H- High salt buffer, Li- LiCl buffer, and TE- TE buffer.

3) Metering Obfuscation: We applied metering obfuscation to Ring-1 by transforming the mixer to a 4-segment mixer ($n_p = 4$). Now the reagents can be loaded in variable volumes, i.e., a reagent can be loaded to fill either 0, 1, 2, 3 or 4 segments. Note that the 0-segment loading corresponds to a dummy load operation. This obfuscation requires the addition of 18 valves. The effort to resolve the metering obfuscation is $\mathcal{E}_{meter} = 2^{n_p} = 2^4$. As the cost of this obfuscation is high, we limit it only Ring-1 and do not apply it to Ring A-D.

The obfuscated sequencing graph is shown in Fig. 20(b). The effort to resolve the behavioral + structural obfuscation is $\mathcal{E} = \mathcal{E}_{behav} \cdot \mathcal{E}_{struct} \cdot \mathcal{E}_{meter} = 2^{14} \cdot 2^4 \cdot 2^4 = 2^{22}$. Each ChIP trial takes 3.5 h. The time for all trials is over a thousand years. Also, each trial consumes reagents, samples, and biochips. Note that this effort assumes a crude attack; we next consider a special cases that aid the attacker.

4) Special Cases: The two fluid inlets of beads and cells have particulate suspension. Loading of these can be obfuscated as a choice between the two. In other words, the 5:1 multiplexer needs to be treated as a 3:1 and a 2:1 multiplexer. In such a case, the effort in resolving the behavioral obfuscation is $\mathcal{E}_{behav} = \mathcal{E}_{mux} \cdot \mathcal{E}_{mix} = 2^{\binom{3}{2}} \cdot 2 \cdot 2^4 = 2^7$. Thereby, the total reverse-engineering effort is given by $\mathcal{E} = \mathcal{E}_{behav} \cdot \mathcal{E}_{struct} \cdot \mathcal{E}_{meter} = 2^7 \cdot 2^4 \cdot 2^4 = 2^{15}$.

Now, let us consider the case where the attacker is aware of the sequence of the reagent. Then, the multiplexer obfuscation is of little use. Then, the effort in resolving the behavioral obfuscation is $\mathcal{E}_{behav} = \mathcal{E}_{mix} = 2^4$. Further, the overall obfuscation is the result of mixer behavior, structural, and metering obfuscations. Thereby, the total reverse-engineering effort is given by $\mathcal{E} = \mathcal{E}_{behav} \cdot \mathcal{E}_{struct} \cdot \mathcal{E}_{meter} = 2^4 \cdot 2^4 \cdot 2^4 = 2^{12}$.

C. Other Benchmarks

We applied the proposed obfuscation to three more real-life biochips and tabulated the results in Table II. The mRNA iso. and Kinase act. are 4-plex and 2-plex biochips, respectively, where identical assays (attacker trials) are run in parallel. In mRNA iso. ($4 \times 14 = 56$ valves) and Kinase act. ($2 \times 22 = 44$ valves) biochips, $\mathcal{E}$ is smaller due to the replication of a smaller structure. On the other hand, in the larger biochips like ChIP (50 valves) and Nucleic-Acid proc. (54 valves), $\mathcal{E}$

TABLE II
CURDE-ATTACK - REVERSE-ENGINEERING OF REAL-LIFE BIOCHIPS

Biochip	#valves	# mux	Load		#mixers	Mix-time		Structural			Meter			Total effort
			#extra valves	effort		#extra valves	effort	#extra valves	#extra inlets	effort	#extra valves	#extra outlets	effort	
ChIP	50	1 (5:1)	5	2^{10}	4	3	2^4	8	0	2^4	18	2	2^4	2^{22}
Kinase act.	44	2 (3:1)	3	2^3	2	2	2^4	6	3	2^3	36	4	2^4	2^{14}
mRNA iso.	56	4 (2:1)	8	2^2	4	8	2^4	8	4	2^4	72	8	2^4	2^{14}
Nucleic-acid proc.	54	3 (5:1)	6	2^9	3	6	2^4	6	0	2^3	18	2	2^4	2^{20}

is larger for a comparable design cost in terms of the number of extra valves. The results imply that the dummy-valve-based obfuscation scales well with the complexity of the biochip.

D. Analysis: Return on Investment

To assess the impact of different types of obfuscation techniques, we define a metric called *return-on-investment* (RoI). For additional d dummy valves and ports, if the resulting reverse-engineering effort is $\mathcal{E}$, then the return-on-investment is

$$RoI = \frac{\mathcal{E}}{d} \tag{4}$$

An increase in the number of valves leads to an increase in the external solenoid valve and pressure control hardware. It also leads to an increase in the biochip area. However, the cost arising from an increase in the number of valves can be minimized using multiplexer-based addressing of the valves [37].

Type of Obfuscation — We tabulate the RoI for different obfuscation techniques applied to the benchmarks in Table III. The results show that with respect to the extra hardware (valves and ports), the best obfuscation type is load, mix-time, structural, and metering, in that order. However, as discussed in the special cases (Section. VII), mix-time and metering obfuscation are more resilient against an attacker who is aware of the sequence of operations. In other words, when the attacker knows the sequence of operations from benchtop bioprotocol, then load and structural obfuscation are ineffective. We show this in Table IV, where only bioprotocol parameters mix-time and volume can be obfuscated. Note that the valid parameter determination is a critical step during the bioprotocol development process, which requires numerous experimental iterations. Hence, a developer is motivated to protect this critical aspect of the IP in spite of the cost.

E. Comparison Against Other Techniques

The strength of our proposal can be demonstrated in comparison with two IP protection techniques. First, firmware encryption has been used to protect firmware IPs. However, this doesn't apply to the biochips because the biochip actuations are electrical signals applied to either the valves or to the pneumatic actuators. Even if the actuation sequence is encrypted, it has to be decrypted before it is applied to the biochip control ports. Further, the actuations can be extracted by image and video-based reverse-engineering. The proposed obfuscation complements encryption to thwart reverse-engineering of the electrical signals. Second, logic locking is used to prevent IP piracy in VLSI designs. The number of trials needed to de-obfuscate a logic-locked VLSI design is of the order of 2^{128} [38]. These trials can be done on high-speed computers. On the other hand, the bioassay trials take several hours to complete. Also, unlike VLSI, the bioassay recovery trials require perishable reagents and biochips. The cost and time spent on these trials go against an attacker's economic objective of stealing a bioassay IP.

TABLE III
ROI OF DIFFERENT OBFUSCATION TYPES

Biochip	ChIP	Kinase act.	mRNA	Nucleic-acid
Behavioral	2048	25.6	4	682.7
Structural	2	0.9	1.3	1.3
Metering	0.8	0.4	0.2	0.8

TABLE IV
KNOWN-SEQUENCE ATTACK REVERSE-ENGINEERING EFFORT

Biochip	Mix-time			Meter				Total effort
	#extra valves	effort	RoI	tiny#extra valves	#extra outlets	effort	RoI	
ChIP	3	2^4	5.3	18	2	2^4	0.8	2^8
Kinase act	2	2^4	8	36	4	22^4	0.4	2^8
mRNA iso.	8	2^4	2	72	8	2^4	0.2	2^8
NA proc.	6	2^4	2.6	18	2	2^4	0.8	2^8

NA: Nucleic-Acid

IX. CONCLUSION

Microfluidic platforms have immense potential in paving the way for rapid and low-cost biochemical analysis. However, the cyberphysical system that enables biochip automation is susceptible to IP theft. This is a major hurdle in the large-scale adaptation of microfluidic technologies in industries that are prone to the stealing of sensitive research data. Our work addresses this pressing problem with a practical obfuscation methodology that can be easily integrated with the current biochip design flow. We fabricated the first obfuscated biochip using the proposed method. We developed dummy-valve-based obfuscation design rules and showcased their application to the real-life biochips. The results show that the de-obfuscation effort is daunting enough to act as a deterrent to an attacker.

APPENDIX

We used the COMSOL tool to simulate the behavior of valves of multiheight against different pressure [39]. As shown

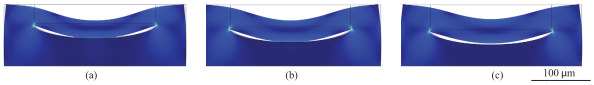


Fig. 21. COMSOL simulation results of a multi-height valve at 1600 psi pressure. The valve membrane height is (a) 30μm (b) 36μm and (c) 40μm. The similar images of the valves provide validation for the proposed obfuscation methodology.

in Fig. 21, the height of the valve can be perturbed to mimic a dummy valve. A defender can decide on the difference in membrane height in normal and dummy valves based on the preferred operating pressure.

Fabrication Procedure of Multi-Height Valve Biochip

Soft lithography is the process of casting soft polymer material (PDMS) onto a mold that contains micro patterns. Microfluidic biochips are produced using this process. A master mold captures the micro architecture of a microfluidic biochip. One mold is created for the flow layer and one for control layer. The microfluidic pattern for each layer is generated using a CAD program and printed onto a transparency film. This pattern is etched onto a silicon wafer using conventional photolithography.

Negative photoresists (SU8-2010 and SU8-2050 Microchem Inc.) were patterned to create a 2-layer microfluidic channel master on 4″ Silicon wafer. The wafer was cleaned and dried before spin coating the SU8-2010 layer on it. Next, mask aligner was used to UV cure. After that it was hard baked and uncured photoresist were removed. The same wafer was then spin coated and UV cured for the second layer of mask. Visual and optical microscopy inspection showed all the structures present and the height of the second layer was found to be $40\mu m$. Similarly, the standard UV lithography is also used to pattern flow master with round shaped profile to allow complete valve closure.

ACKNOWLEDGMENT

The authors like to thank Navajit Singh Baban of New York University Abu Dhabi and Urbi Chattarjee of IIT Kharagpur for their inputs.

REFERENCES

[1] S. R. Quake, J. S. Marcus, and C. L. Hansen, "Microfluidic sieve valves," U.S. Patent 8 932 461 B2, Jan. 13, 2015. [Online]. Available: https://patents.google.com/patent/US8932461B2/en

[2] A. H. C. Ng, U. Uddayasankar, and A. R. Wheeler, "Immunoassays in microfluidic systems," *Anal. Bioanal. Chem.*, vol. 397, no. 3, pp. 991–1007, Jun. 2010.

[3] J. W. Hong, V. Studer, G. Hang, W. F. Anderson, and S. R. Quake, "A nanoliter-scale nucleic acid processor with parallel architecture," *Nature Biotechnol.*, vol. 22, no. 4, pp. 435–439, Apr. 2004.

[4] Y.-H.-V. Ma, K. Middleton, L. You, and Y. Sun, "A review of microfluidic approaches for investigating cancer extravasation during metastasis," *Microsyst. Nanoeng.*, vol. 4, no. 1, pp. 1–13, Apr. 2018.

[5] (2018). *MARKETandMARKET*. [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/microfluidics-market-1%305.html

[6] S. S. Ali, M. Ibrahim, O. Sinanoglu, K. Chakrabarty, and R. Karri, "Microfluidic encryption of on-chip biochemical assays," in *Proc. IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2016, pp. 152–155.

[7] (2016). *Drug Development and Intellectual Property Theft*. [Online]. Available: https://digitalguardian.com/blog/drug-development-and-intellectual-prop%erty-theft

[8] (2016). *2 GSK Scientists Indicted in Secrets Case Involving China*. [Online]. Available: https://www.justice.gov/usao-edpa/pr/scientists-indicted-allegedly-stea%ling-biopharmaceutical-trade-secrets

[9] P. S. Dittrich and A. Manz, "Lab-on-a-chip: Microfluidics in drug discovery," *Nature Rev. Drug Discovery*, vol. 5, no. 3, pp. 210–218, Mar. 2006.

[10] J. Melin and S. R. Quake, "Microfluidic large-scale integration: The evolution of design rules for biological automation," *Annu. Rev. Biophys. Biomol. Struct.*, vol. 36, no. 1, pp. 213–231, Jun. 2007.

[11] R. B. Fair, "Digital microfluidics: Is a true lab-on-a-chip possible?" *Microfluidics Nanofluidics*, vol. 3, no. 3, pp. 245–281, Apr. 2007.

[12] H. Chen, S. Potluri, and F. Koushanfar, "BioChipWork: Reverse engineering of microfluidic biochips," in *Proc. IEEE Int. Conf. Comput. Design (ICCD)*, Nov. 2017, pp. 9–16.

[13] (2016). *Fda Advisors Back Approval of Baebies' Seeker Analyzer for Newborns*. [Online]. Available: http://baebies.com/fda-advisors-back-approval-baebies-seeker-analyzer-n%ewborns

[14] A. K. Giri and D. R. Rana, "Charting the challenges behind the testing of COVID-19 in developing countries: Nepal as a case study," *Biosafety Health*, vol. 2, no. 2, pp. 53–56, Jun. 2020.

[15] (2020). *Rapid Acceleration of Diagnostics (RADX)*. [Online]. Available: https://www.nih.gov/research-training/medical-research-initiatives/radx%/radx-programs

[16] J. P. Urbanski, W. Thies, C. Rhodes, S. Amarasinghe, and T. Thorsen, "Digital microfluidics using soft lithography," *Lab Chip*, vol. 6, no. 1, pp. 96–104, 2006.

[17] A. Lau, H. Yip, K. Ng, X. Cui, and R. Lam, "Dynamics of microvalve operations in integrated microfluidics," *Micromachines*, vol. 5, no. 1, pp. 50–65, Feb. 2014.

[18] M. Shayan, S. Bhattacharjee, J. Tang, K. Chakrabarty, and R. Karri, "Bio-protocol watermarking on digital microfluidic biochips," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 11, pp. 2901–2915, Nov. 2019.

[19] T.-Y. Ho, W. Grover, S. Hu, and K. Chakrabarty, "Cyber-physical integration in programmable microfluidic biochips," in *Proc. 33rd IEEE Int. Conf. Comput. Design (ICCD)*, Oct. 2015, pp. 224–227.

[20] A. R. Wu *et al.*, "Automated microfluidic chromatin immunoprecipitation from 2,000 cells," *Lab Chip*, vol. 9, no. 10, pp. 1365–1370, 2009.

[21] S. Chung, J. Park, C. Chung, D. C. Han, and J. K. Chang, "Multi-height micro structures in poly(dimethyl siloxane) lab-on-a-chip," *Microsyst. Technol.*, vol. 10, no. 2, pp. 81–88, Jan. 2004.

[22] K. Hu, F. Yu, T.-Y. Ho, and K. Chakrabarty, "Testing of flow-based microfluidic biochips: Fault modeling, test generation, and experimental demonstration," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 33, no. 10, pp. 1463–1475, Oct. 2014.

[23] (2018). *Laboratory Monitoring*. [Online]. Available: http://tetrascience.com/case-studies/laboratory-monitoring-notable-labs

[24] (2017). *Pharmaceutical Giant Rocked by Ransomware Attack*. [Online]. Available: https://www.washingtonpost.com/news/the-switch/wp/2017/06/27/pharmaceut%ical-giant-rocked-by-ransomware-attack/

[25] M. Ibrahim, K. Chakrabarty, and K. Scott, "Synthesis of cyberphysical digital-microfluidic biochips for real-time quantitative analysis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 36, no. 5, pp. 733–746, May 2017.

[26] U. D. of Homeland Security and CERT. (2018). *Russian Government Cyber Activity Targeting Energy and Other Critical Infrastructure Sectors*. [Online]. Available: https://www.us-cert.gov/ncas/alerts/TA18-074A

[27] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Secur. Privacy Mag.*, vol. 9, no. 3, pp. 49–51, May 2011.

[28] S. Bhattacharjee, J. Tang, M. Ibrahim, K. Chakrabarty, and R. Karri, "Locking of biochemical assays for digital microfluidic biochips," in *Proc. IEEE 23rd Eur. Test Symp. (ETS)*, May 2018, pp. 1–6.

[29] (2018). *Lessons for Pharma From the Merck Cyber Attack*. [Online]. Available: https://www.pharmexec.com/view/lessons-pharma-merck-cyber-attack

[30] K. Collier. (Sep. 28, 2020). *Major Hospital System Hit With Cyberattack, Potentially Largest in U.S. History*. [Online]. Available: https://www.nbcnews.com/tech/security/cyberattack-hits-major-u-s-hospital-system-n1241254

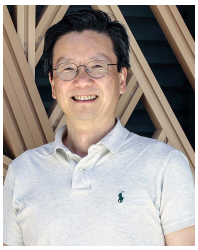
[31] M. Shayan, S. Bhattacharjee, Y.-A. Song, K. Chakrabarty, and R. Karri, "Can multi-layer microfluidic design methods aid bio-intellectual property protection?" in *Proc. IEEE 25th Int. Symp. Line Test. Robust Syst. Design (IOLTS)*, Jul. 2019, pp. 151–154.

[32] C. C. Glick *et al.*, "Rapid assembly of multilayer microfluidic structures via 3D-printed transfer molding and bonding," *Microsyst. Nanoeng.*, vol. 2, no. 1, p. 16063, Dec. 2016.
[33] W. Thies, J. P. Urbanski, T. Thorsen, and S. Amarasinghe, "Abstraction layers for scalable microfluidic biocomputing," *Natural Comput.*, vol. 7, no. 2, pp. 255–275, Jun. 2008.
[34] K. Choi *et al.*, "Automated digital microfluidic platform for Magnetic-Particle-Based immunoassays with optimization by design of experiments," *Anal. Chem.*, vol. 85, no. 20, pp. 9638–9646, Oct. 2013.
[35] P. Paik, V. K. Pamula, and R. B. Fair, "Rapid droplet mixers for digital microfluidic systems," *Lab Chip*, vol. 3, no. 4, pp. 253–259, 2003.
[36] M. Ibrahim and K. Chakrabarty, "Cyber–physical digital-microfluidic biochips: Bridging the gap between microfluidics and microbiology," *Proc. IEEE*, vol. 106, no. 9, pp. 1717–1743, Oct. 2018.
[37] L. M. Fidalgo and S. J. Maerkl, "A software-programmable microfluidic device for automated biology," *Lab Chip*, vol. 11, no. 9, pp. 1612–1619, 2011.
[38] J. Rajendran *et al.*, "Fault analysis-based logic encryption," *IEEE Trans. Comput.*, vol. 64, no. 2, pp. 410–424, Feb. 2015.
[39] (2020). *COMSOL Product: Microfluidic Module*. [Online]. Available: https://www.comsol.com/microfluidics-module

Mohammed Shayan (Member, IEEE) received the M.E. degree in microelectronics from the Indian Institute of Science, Bangalore, in 2012. He is currently pursuing the Ph.D. degree in electrical and computer engineering with New York University, Brooklyn, NY, USA. His research interests include security and intellectual property rights protection in the emerging technologies, such as cyber-physical microfluidic systems and interposer-based integration.

Sukanta Bhattacharjee (Member, IEEE) received the B.Tech. degree in computer science and engineering from the University of Calcutta, India, and the M.Tech. and Ph.D. degrees in computer science from the Indian Statistical Institute, Kolkata, India. He is currently working as an Assistant Professor with the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, India. His research interests include design automation algorithms for microfluidic biochip, formal methods, and security.

Ajymurat Orozaliev received the M.S. degree in microsystems engineering from the Masdar Institute of Khalifa University, Abu Dhabi, United Arab Emirates. He is currently a Nanofabrication Research Engineer with the Micro and Nanoscale Bioengineering Laboratory, NYUAD.



Yong-Ak Song received the B.S., M.S., and Ph.D. degrees from RWTH Aachen University, Germany. He currently holds an affiliate appointment with the Department of Chemical and Biomolecular Engineering, NYU-Poly. His research and teaching interests include interdisciplinary in both mechanical engineering disciplines, such as design and manufacturing of MEMS devices, fluid mechanics, and micro/nanofabrication, as well as in biological engineering areas such as BioMEMS devices for separation and detection of biomolecules, neuroprosthetic implants, and transport phenomena in biological systems.

Krishnendu Chakrabarty (Fellow, IEEE) received the B.Tech. degree from the Indian Institute of Technology, Kharagpur, in 1990, and the M.S.E. and Ph.D. degrees from the University of Michigan, Ann Arbor, MI, USA, in 1992 and 1995, respectively. He is currently the John Cocke Distinguished Professor and the Department Chair of Electrical and Computer Engineering (ECE) with Duke University. His current research interests include design-for-testability of integrated circuits and systems, microfluidic biochips, hardware security, machine-learning accelerators, and neuromorphic computing systems. He is a fellow of ACM and AAAS, and a Golden Core Member of the IEEE Computer Society.

Ramesh Karri (Fellow, IEEE) received the B.E. degree in ECE from Andhra University and the Ph.D. degree in computer science and engineering from the University of California at San Diego. He is currently a Professor of Electrical and Computer Engineering with New York University. He also co-directs the NYU Center for Cyber Security. He also leads the Cyber Security thrust of the NY State Center for Advanced Telecommunications Technologies, NYU. He co-founded the Trust-Hub. His research and education interests include hardware cybersecurity include trustworthy ICs, processors and cyber-physical systems; security-aware computer-aided design, test, verification, validation, and reliability, nano meets security, hardware security competitions, benchmarks, and metrics, biochip security, and additive manufacturing security.