Synthesis of Cyberphysical Digital-Microfluidic Biochips for Real-Time Quantitative Analysis

Mohamed Ibrahim, *Student Member, IEEE*, Krishnendu Chakrabarty, *Fellow, IEEE*, and Kristin Scott, *Member, IEEE*

***Abstract*—Considerable effort has recently been directed toward the implementation of molecular bioassays on digital-microfluidic biochips (DMFBs). However, today's solutions suffer from the drawback that multiple sample pathways are not supported and on-chip reconfigurable devices are not efficiently exploited. As a result, impractical manual intervention is needed to process protocols for gene-expression analysis. To overcome this problem, we first describe our benchtop experimental studies to understand gene-expression analysis and its relationship to the biochip design specification. We then introduce an integrated framework for quantitative gene-expression analysis using DMFBs. The proposed framework includes: 1) a spatial-reconfiguration technique that incorporates resource-sharing specifications into the synthesis flow; 2) an interactive firmware that collects and analyzes sensor data based on quantitative polymerase chain reaction; and 3) a real-time resource-allocation scheme that responds promptly to decisions about the protocol flow received from the firmware layer. This framework is combined with cyberphysical integration to develop the first design-automation framework for quantitative gene expression. Simulation results show that our adaptive framework efficiently utilizes on-chip resources to reduce time-to-result without sacrificing the chip's lifetime.**

***Index Terms*—Biochip, cyberphysical systems, microfluidics, quantitative analysis, synthesis.**

I. Introduction

ACCORDING to market research studies from Yole Développement, the world market for microfluidic devices is forecast to grow from $1.1 billion in 2011 to $5.7 billion by 2018 [1]. As an indicator of commercial success, Illumina, a market leader in DNA sequencing, has recently transitioned microfluidic biochips to the marketplace for sample preparation [2]. This significant milestone highlights the emergence of digital-microfluidic biochips (DMFBs) for commercial exploitation and their potential for immunoassays for point-of-care diagnosis [3], proteomic sample processing, and cell-based assays [4], [5]. Using DMFBs, bioassay protocols are scaled down in terms of liquid volumes and assay times and executed by enabling precise control of discrete droplets using a patterned array of electrodes. Therefore, picoliter droplets of samples and reagents can be dispensed, incubated, transported, mixed, split, heated, or electroporated under software control in a cost-effective manner [6]. The flexibility provided by this technology, along with advances in the integration of sensors [6], [7] and droplet monitoring using CCD cameras [8], were exploited in [9] to develop a physical-aware system reconfiguration technique.

Due to the fundamental importance of genomic analysis, major advances have also been reported on miniaturized technology platforms [10]–[15]. With the advent of these platforms, genomic bioassays such as nucleic-acid (NA) isolation, DNA purification, and DNA amplification have been successfully realized and integrated on DMFBs. Temperature-cycling of samples for DNA amplification, also referred to as polymerase chain reaction (PCR), has also been demonstrated on a DMFB.

However, since these platforms were intrinsically designed for experimentation on a sample-limited setting, on-chip devices were allocated *a priori* with respect to the bioassays constituting the protocol; thus the flexibility and reconfigurability of these devices have not been efficiently exploited. For example, today's DMFBs can be flexibly equipped with a mechanism for temperature-cycling through which a target sample is kept immobile while the temperature of an associated chip region is precisely tuned [13], [14]. This mechanism enables *resource sharing*, i.e., the option of using such regions for various purposes, such as PCR, thermal cell lysis or even traditional sample processing. Since quantitative analysis protocols include multiple sample pathways that are independently manipulated, there is inherent uncertainty about the order of basic fluidic steps; thus resource sharing is a significant challenge. Inefficient exploitation of on-chip devices for real-life protocols entails significant cost if devices such as heaters and sensors need to be replicated.

On the other hand, the electrodes located within the heating regions are unique in terms of their role since they control the droplets that are subjected to thermal manipulation. Therefore, excessive usage of these electrodes leads to degradation [14], which significantly impacts biochip lifetime. The same concern applies to other on-chip devices such as sensors and electrodes that are in proximity to magnets used

Manuscript received January 28, 2016; revised May 27, 2016; accepted July 24, 2016. Date of publication August 16, 2016; date of current version April 19, 2017. This work was supported by the National Science Foundation under Grant CNS-1135853. A preliminary version of this paper was accepted for publication at the IEEE/ACM Design Automation and Test in Europe in March 2016. This paper was recommended by Associate Editor A. Srivastava.

M. Ibrahim and K. Chakrabarty are with the Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708 USA (e-mail: mohamed.s.ibrahim@duke.edu; krish@ee.duke.edu).

K. Scott is with the Department of Molecular Genetics and Microbiology, Duke University, Durham, NC 27710 USA (e-mail: kristin.scott@duke.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCAD.2016.2600626

for bead-based assays. Hence, adaptive resource allocation is needed to harness the biochip capabilities for large-scale quantitative analysis without sacrificing the chip's lifetime.

Furthermore, microfluidics-based platforms used for quantifying gene expression rely on an off-chip, nonadaptive analysis framework to determine the performance of the protocol and the expression level for a target gene [13]. As a result, these platforms cannot be utilized in down-stream molecular analysis such as epigenetic regulation studies, which are based on decisions of gene-expression level [16]. The need for providing an adaptive framework for gene-expression analysis can be addressed through cyberphysical integration in DMFBs.

Cyberphysical integration has been recently exploited in DMFBs for the purpose of dynamic reconfiguration due to error occurrences [9], [17]. This reconfiguration technique, however, does not provide the versatility needed to support quantitative analysis due to the following reasons.

1) Quantitative gene-expression analysis relies on the detection outcome for multiple samples. Current reconfiguration approach for error recovery can only address one sample at a time; thus it is unable to support global decisions for the entire multisample protocol.
2) PCR-based quantitative analysis requires the recording and analysis of detection data over the course of thermal cycling. It does not depend on an instantaneous detection operation (e.g., capacitive detection [18]).

Accordingly, there is a need to empower DMFBs with an interactive firmware layer that collects data from on-chip sensors, performs real-time data analysis, and communicates the protocol-flow decisions to the resource-allocation software.

In this paper, we advance cyberphysical integration of DMFBs by introducing the first integrated platform and design-automation solution for quantitative analysis, e.g., the study of gene expression in molecular biology. Based on benchtop experimental work, we present a laboratory-based approach for executing quantitative analysis bioassays. We then present a method for physical-aware resource allocation for multiple sample pathways. The main contributions of this paper are as follows.

1) We first discuss our benchtop experimental study for gene-expression analysis along with the resulting outcomes. This experimental approach is used to guide biochip synthesis for the underlying protocol.
2) We present a spatial-reconfiguration technique that incorporates resource-sharing constraints into the synthesis flow.
3) We describe and evaluate a physical-aware resource-allocation framework that enables tunable resource allocation among bioassays.
4) To bridge the gap between cyber and physical spaces in DMFBs for quantitative analysis, we introduce a software library for an interactive firmware layer. This layer collects and analyzes data from on-chip sensors. Thus, it communicates flow decisions to the resource-allocation scheme. We also describe a software utility that enables the firmware to provide quantitative results in existence of imprecise amplification data.

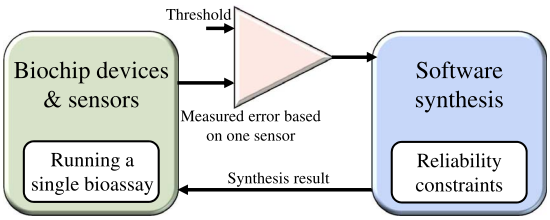

Fig. 1. Cyberphysical integration of DMFBs for error recovery [9].

Note that even though we consider the gene-expression analysis protocol in this paper due to its widespread use, the proposed framework is applicable to alternative methods for quantitative analysis that seek the quantification of other sample properties such as concentration (in glucose testing) [19]. In other words, it is not limited to the measurement of relative abundance of a gene expression as considered in this paper.

The rest of this paper is organized as follows. Section II describes related prior work. An introduction to gene-expression analysis and the implemented benchtop experiment are presented in Section III. Protocol miniaturization using DMFBs and the associated layered-software support are described in Section IV. Section V explains the requirement of spatial reconfiguration for the efficient realization of protocols. Next, the shared resource-allocation algorithm is presented in Section VI. Section VII describes the software architecture of the interactive firmware layer. Finally, results of our experimental evaluation are presented in Section VIII and the conclusions are drawn in Section IX.

II. Related Prior Work

Early research on design automation for digital microfluidics focused on scheduling of fluidic operations, resource binding, droplet routing, and mapping of control pins to electrodes [20]–[25]. However, these methods are limited to single sample pathways and they cannot handle uncertainties in the order of fluidic steps in bioassay execution. Moreover, the interplay between hardware and software in the biochip platform (cyberphysical system design) was not considered in these early methods.

Design and optimization techniques for cyberphysical DMFBs have thus far considered only error recovery [9], [17], [26], [27] and termination control of a biochemical procedure such as PCR [28]. These designs consist of two main functional components.

1) The feedback sensor connectivity that ensures real-time detection of the sample droplets.
2) Intelligent control software (online synthesis) that captures the detection outcome and provides an appropriate action (see Fig. 1).

However, these designs do not support real-time decision making for protocols designated for multiple samples, and they do not exploit the potential of DMFBs for quantitative analysis-based biochemical reactions. Gao *et al.* [29] proposed a modular design with built-in electronic control to enhance the reliability and robustness of DMFBs. However, this design can only be used for droplet volume measurement and position control. With this computationally expensive method, the

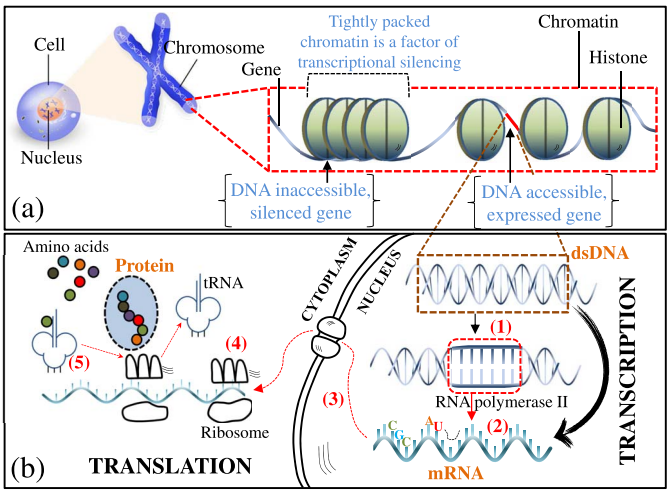

Fig. 2. Sketch of biological pathways inside a cell. (a) Illustration of gene expression and silencing. (b) Steps in protein synthesis from an expressed DNA; i.e., the transcription and translation processes: (1) the DNA double helix is unwound, (2) the enzyme RNA polymerase II then moves along the exposed DNA strand to generate RNA, (3) the resulting mRNA leaves the nucleus through a nuclear pore to go to a ribosome, and (4) and (5) the ribosome links amino acids together in the order specified by mRNA molecules to create protein.

control of more than a few droplets transported concurrently on an array becomes impractical. In [30], an experimental demonstration of hardware-based real-time error recovery in integrated cyberphysical DMFBs has been shown. In this paper, a capacitive sensor, signal conditioning circuit, and a hardware control machine have been used to address reliable bioassay execution in a practical setting. Yet this framework lacks the support for real-time quantitative measurement.

Therefore, a major limitation of all prior work on design automation and cyberphysical integration is that they are limited to simple droplet manipulation on a chip; however, in order to make DMFBs useful to a biologist, we need a new design paradigm to demonstrate that these chips can be used for actual biomolecular protocols from microbiology.

III. Benchtop Protocol for Gene-Expression Analysis

Contemporary molecular biology research relies extensively on gene-expression analysis [31], which is required in many areas such as disease diagnostics [32], pathogen detection [33], and forensic identification [34]. Gene-expression analysis is used for elucidating the *transcriptional* profile of biological systems. In the gene-expression process (Fig. 2), a particular segment of DNA is enzymatically processed, or transcribed, into an RNA molecule [35]. Then, a specific product of RNA, namely messenger RNA (mRNA), contributes to the *translation* process. This step leads to proteins that form the functions of our life [36]. Notably, the DNA sequences involved in the establishment of proteins are said to be "expressed genes." On the other hand, the DNA sequences that do not elucidate a high level of expression (i.e., sequences that are not regularly transcribed or are under the influence of "epigenetic transcriptional control") are said to be "silenced/suppressed genes." Cancer and neurodegenerative disorders are generally associated with genes that have a reduced level of expression, therefore methods used to silence genes are being increasingly used to produce therapeutics to combat such diseases [37].

Accurate quantification of the expression level for a target gene requires a multistep protocol on sample cells [16]. In this section, we describe our benchtop experimental study to analyze the expression level of *Schizosaccharomyces pombe (S. pombe)* strains.

A. Benchtop Experiment and Decision Making

The objective of the experiment was to study the transcriptional profile of a green fluorescent protein (GFP) reporter gene under epigenetic control. In this experiment, three types of *S. pombe* strains were analyzed: 1) control (GFP not under epigenetic control) strains ($\mathcal{S}_c$); 2) experimental (GFP under epigenetic control) strains ($\mathcal{S}_e$); and 3) wild-type strains used as a reference to improve the outcome efficiency ($\mathcal{S}_r$). Note that these strains were run manually and independently in a set of reactions in which the investigated gene is GFP. Ultimately, the need to improve the efficiency of the experiment and target combinations of multiple genes impose practical limitations on the benchtop approach. To reduce the experiment time and to minimize the process cost, it is necessary to design a digital-microfluidic platform that can run tens of droplets through independent sample pathways under automated control.

Fig. 3 depicts a flowchart of the protocol for quantitative gene-expression analysis that we have studied using a benchtop setup. The expression level of *S. pombe* strains were analyzed by quantitative PCR (qPCR) following cell lysis, mRNA isolation and purification, and complementary DNA (cDNA) synthesis. The details of the experiment are described below.

1) The samples were first placed in culture medium and cells were grown overnight under controlled condition. Cell culture is a critical step that is needed to ensure to that cells remain alive after they are isolated from their natural environment; i.e., a tissue [38].
2) In order to evaluate cell concentration and assess whether a cell environment is uncontaminated, the cells of each sample were observed under a phase contrast microscope. Contaminated cells or samples with improper cell concentration must be discarded and the cell-culture process is repeated.
3) Cell lysis was performed for each sample to release intracellular contents, e.g., protein, NA (DNA and RNA), and cell debris. Glass beads were used for cell lysis, since glass beads can mechanically disrupt cell walls.
4) To get rid of protein and cell debris, NAs were isolated and precipitated with the aid of 100% ethanol. The NAs were then resuspended in ultrapurified Diethylpyrocarbonate water.
5) The purity of the isolated NA for each sample was assessed by spectrophotometry. A spectrophotometer was used to evaluate the light absorbance for all samples—that is an indication to the isolation quality [39]. Note that a sample with poorly isolated NA

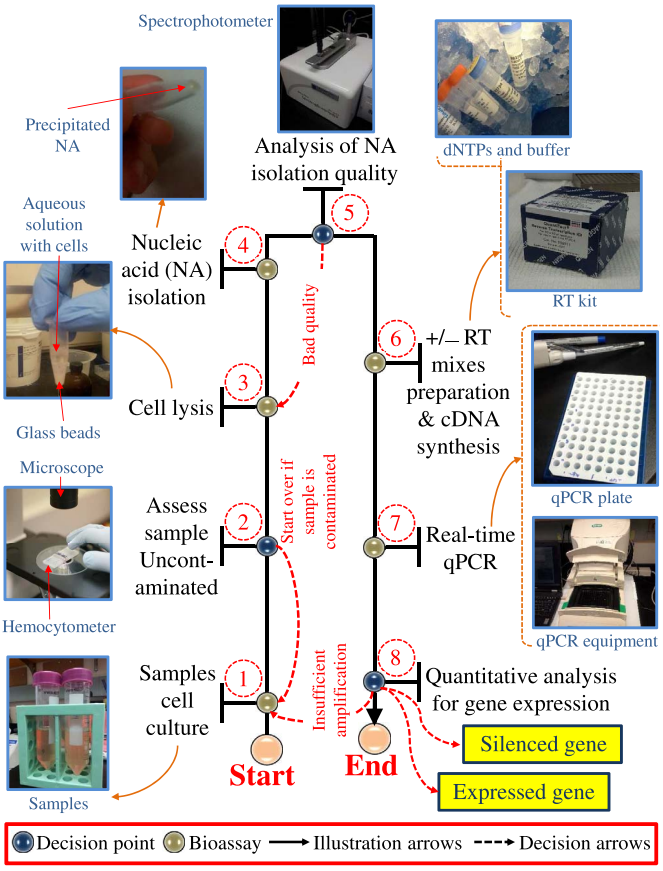

Fig. 3. Protocol flowchart for quantitative analysis of gene expression using a benchtop setup.

(i.e., NA co-exists with protein) has to be discarded and we have to start over with a new sample of the same type.

6) We proceeded to this step after successful isolation of NA. Using "QuantiTect Reverse Transcription (RT)" kit (from Qiagen, Inc.), a DNA-wipe-out (DNAse) was added to each sample to eliminate all the DNA, leaving the mRNA in the solution. Recall that NAs consist of DNA and RNA molecules. Next, positive and negative RT mixes (constructed from buffers and Deoxynucleotides) were prepared to assist in generating the cDNA for these particular strains. Negative mixes (also known as negative or no-template controls) serve as a general control for extraneous NA contamination [40].
7) Finally, DNA amplification through qPCR was carried out using the following steps.
   a) Each sample was iteratively tenfold diluted to create multiple copies; each being with a unique DNA concentration (serial dilution).
   b) qPCR reactions were performed in duplicate with GFP-specific primers (positive and negative RT mixes) in the presence of SYBR green using a BioRad iCycler (BioRad, Inc).

Note that the expression level of GFP in the *S. pombe* samples needs to be quantified with respect to a reference level. This was achieved by qPCR analysis using gene-specific

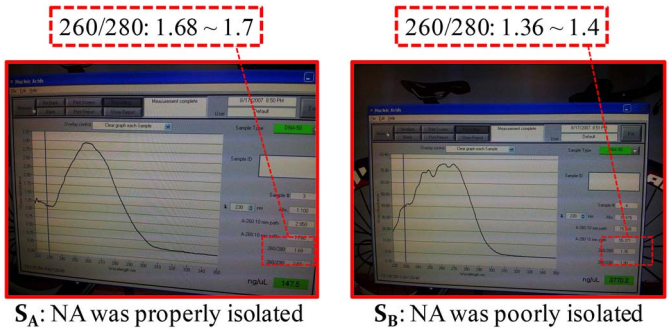

Fig. 4. Assessment of NA isolation quality for two samples $\mathcal{S}_A$ and $\mathcal{S}_B$.

primers to the constitutively expressed β-actin gene, which is constitutively expressed constantly under any experimental conditions [41]. Therefore, steps 1–6 were concurrently conducted twice using the same types of samples; we used a GFP-complementary primer in the first run and a β-actin-complementary primer in the second run. The dilutes of both runs were included in the qPCR analysis.

Furthermore, note that in the described protocol, intermediate decision points have been used to control the protocol flow for every sample. To illustrate the fundamental role of these decision points, we present the outcome of NA-isolation assessment (step 5 in Fig. 3) for two different samples $\mathcal{S}_A$ and $\mathcal{S}_B$. According to [39], the use of spectrophotometer enables us to distinguish NA from protein, which ideally have absorbance maxima at 260 and 280 nm, respectively. Typically, the ratio of absorbances at these wavelengths (referred to as 260/280 ratio) has been used as a measure of purity in both NA and protein extractions. A resulting ratio in the range 1.7–2.0 indicates a purely isolated NA.

As shown in Fig. 4, the spectrum of sample $\mathcal{S}_A$ as well as the 260/280 ratio indicate that NA was properly isolated and extracted in this sample; thus we proceeded to the following step. On the other hand, the spectrum of $\mathcal{S}_B$ and its associated ratio show that NA is contaminated with protein; i.e., it was poorly isolated. In this case, $\mathcal{S}_B$ cannot be used since it leads to inaccurate quantification results. Therefore, sample $\mathcal{S}_B$ was discarded and a new sample belonging to the same type was resuspended for the reaction.

It is therefore evident that the incorporation of sample-dependent decision points is crucial for accurate quantitative analysis.

B. Protocol Efficiency and Quantitative Analysis

Determination of the amplification efficiency is a critical step in a quantitative-analysis protocol. It gives a measure of whether the DNA amplification is approximately the same for all samples—that is a necessary condition for successful gene-expression analysis. qPCR provides the following terminologies for the assessment of DNA amplification [41].

1) *Sample Amplification Plot:* It is a sigmoid-like plot (Fig. 5) which shows the PCR cycle number on the x-axis, whereas the fluorescence from the amplification (which is proportional to the amount of the amplified

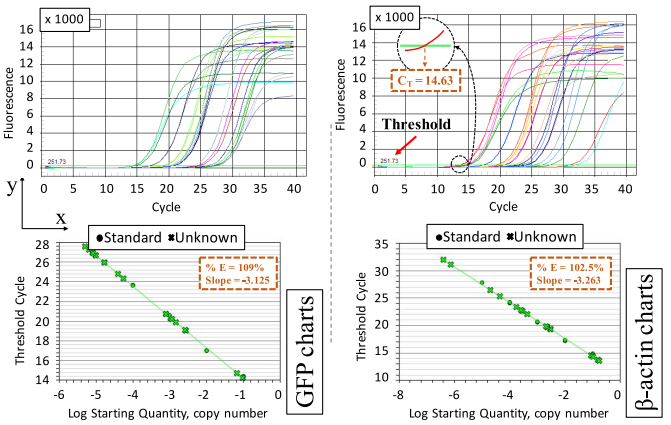

Fig. 5. Generated amplification and standard curves for samples which target GFP and β-actin (housekeeping) genes. Results are obtained using dilutes of $\mathcal{S}_c$, $\mathcal{S}_e$, and $\mathcal{S}_r$ and recorded using a BioRad iCycler.

product in the sample) is shown on the y-axis. An important aspect of this plot is the threshold cycle $\mathcal{C}_T$, which signifies the cycle at which sufficient amplified product starts to accumulate to yield a detectable fluorescence signal. $\mathcal{C}_T$ for a sample is calculated based on a prespecified threshold. Fig. 5 shows amplification plots based on our benchtop experiments.

2) *Standard Curve:* A powerful way to determine whether qPCR implementation is optimized is to run serial dilution on every sample that has purified NA, as described in Section III-A. The diluted samples are then amplified and used to generate a standard curve (Fig. 5). Construction of a standard curve is carried out by plotting the log of the sample's dilution factor (tenfold dilution factor in our experiment) against the $\mathcal{C}_T$ value obtained during the amplification of each dilution.

Next, the amplification efficiency E^* is calculated based on the slope of the standard curve using the following formula [42]:

$$E^* = \left(10^{-\frac{1}{\text{slope}}} - 1\right) \times 100\%$$

where an ideal reaction implies that $E^* = (2-1) \times 100\% = 100\%$; i.e., there is a twofold increase in the number of DNA copies with every cycle. Normally, an amplification efficiency in the range 95%–105% is acceptable.

Once amplification efficiencies are assessed to be nearly 100%, we quantify the gene-expression of GFP gene (target gene) relative to β-actin gene based on $\mathcal{S}_e$ and $\mathcal{S}_c$. The $2^{-\Delta\Delta\mathcal{C}_T}$ (Livak) method is used for quantification as follows [43].

Step 1: Normalize the $\mathcal{C}_T$ of GFP to that of β-actin for both types of samples

$$\Delta\mathcal{C}_{T(\mathcal{S}_e)} = \mathcal{C}_{T(\text{GFP},\mathcal{S}_e)} - \mathcal{C}_{T(\beta\text{-actin},\mathcal{S}_e)}$$
$$\Delta\mathcal{C}_{T(\mathcal{S}_c)} = \mathcal{C}_{T(\text{GFP},\mathcal{S}_c)} - \mathcal{C}_{T(\beta\text{-actin},\mathcal{S}_c)}.$$

Step 2: Normalize the $\Delta\mathcal{C}_T$ of $\mathcal{S}_e$ to that of $\mathcal{S}_c$

$$\Delta\Delta\mathcal{C}_{T=\Delta\mathcal{C}_{T(\mathcal{S}_e)}-\Delta\mathcal{C}_{T(\mathcal{S}_c)}}.$$

Step 3: Finally, calculate the expression ratio as follows:

$$2^{-\Delta\Delta\mathcal{C}_T} = \text{Normalized expression ratio.}$$

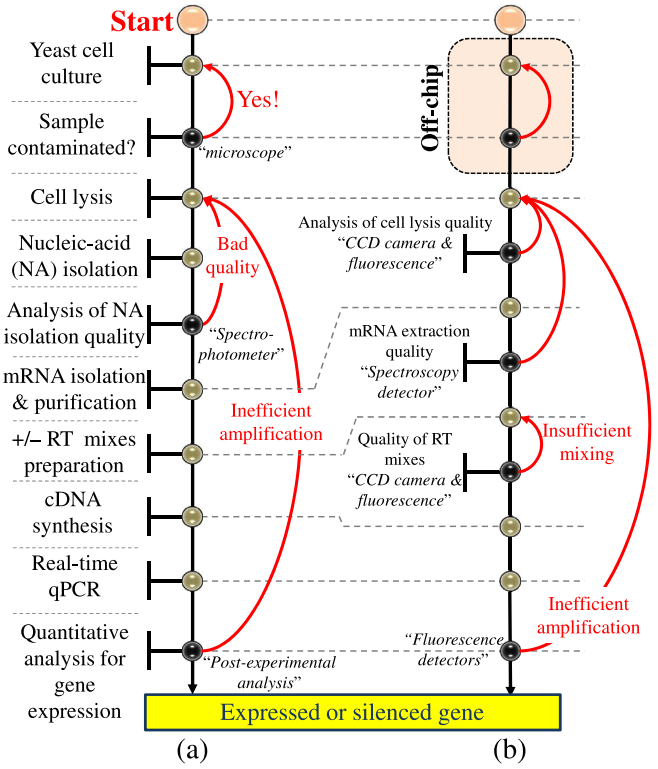

Fig. 6. Mapping of the quantitative protocol for gene-expression analysis from (a) a benchtop approach to (b) DMFBs.

The result obtained in this manner is the fold increase (or decrease) of the expression of the target gene normalized to the expression of a housekeeping gene. Hence, a target gene is said to be expressed if the fold increase is above a certain limit. This global result can then be used to guide the next steps for studying epigenetic regulation.

IV. Digital Microfluidics for Gene-Expression Analysis

In this section, we present our mapping of gene-expression analysis to DMFBs.

A. Protocol Miniaturization

An early nonadaptive implementation of gene-expression analysis was realized using digital microfluidics [13]. Our recent benchtop experience with multiple sample pathways highlights the need for incorporating decision-making and adaptation capability, hence we have developed the enhanced and miniaturized protocol shown in Fig. 6. On-chip operation begins with the dispensing of sample droplets containing cultured cells. The cells are then lysed in order to obtain intracellular materials (DNAs, RNAs, proteins, etc.). Using magnetic beads, enzymes, and a washing step, mRNA can be isolated and then reverse-transcribed into the corresponding cDNA with primers and other reverse-transcription reagents. Next, the resulting cDNA samples are subjected to thermal cycling via qPCR to amplify the target gene.

Similar to the benchtop approach, miniaturization of gene-expression analysis requires the execution of this protocol

on two sample droplets, one of which is used to quantify the amplification of the gene-under-investigation, whereas the other droplet is used to quantify the amplification of a reference gene, also known as a housekeeping gene [41]. Since the primers are gene-specific, the two droplets are chemically treated using different types of primers. However, based on the outcomes of the intermediate decision points, the two droplets can be utilized in an unpredictable manner by different bioassays; this unpredictability makes microfluidic control difficult. Furthermore, with the randomness exhibited in molecular interactions, effective gene-expression analysis requires that the experiment be conducted on at least three replicates. The expression level is first calculated for each replicate, then averaged across the three replicates.

Hence, designing an autonomous digital-microfluidic system for gene-expression analysis requires the concurrent manipulation of independent samples. Utilizing the decision points shown in Fig. 6(b), we incorporate sample-dependent decision-making capability into the cyberphysical system. In addition, the specification of the protocol efficiency and the level of gene expression are included on-chip in the feedback system. Illustration of the firmware design that assists in evaluating the protocol efficiency and gene-expression level on DMFBs is presented in Section VII.

B. Layered-Software Support and Protocol Model

As discussed in Section II, the cyberphysical configuration shown in Fig. 1 is suitable only for error recovery; such a configuration does not avail DMFBs to support quantitative protocols that carry out analysis based on multiple sample pathways. Therefore, there is a need to modify the configuration of cyberphysical integration, as shown in Fig. 7. The proposed configuration relies on a layered-software architecture, which includes the following layers.

1) A firmware layer that is responsible for collecting readings from the sensors, analyzing their data, and providing decisions about protocol flow.
2) Adaptive resource-allocation layer that applies spatial reconfiguration to the chip resources in order to achieve constrained resource allocation among pathways.
3) Software synthesis layer that performs architectural- and physical-level synthesis based on constraints from the resource-allocation layer. Such a configuration facilitates the interaction between the system controller and the multiple sample pathways involved in the protocol.

A number of hardware/software co-synthesis techniques for embedded systems capture control dependencies in the task specifications by using conditional process graphs [44]–[46]. Similarly, in DMFBs, in order to solve the synthesis problem for the new paradigm, we represent the protocol as a control flow graph (CFG) $G_c = (V_c, E_c)$, in which every node $v_c \in V_c$ (referred to here as a *supernode*) signifies a bioassay, e.g., cell lysis, and a directed edge $e_c(v_{c1}, v_{c2}) \in E_c$ represents a potential transition (decision-making) path from v_{c1} to v_{c2}. The bioassay operations corresponding to a supernode v_c, in turn, are represented in a directed sequencing graph $G_s = (V_s, E_s)$, which shows the timing and the

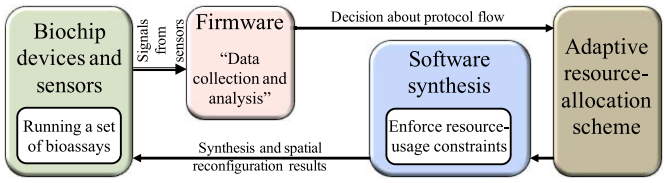

Fig. 7. Proposed cyberphysical integration and layered-software architecture for DMFBs to support gene-expression analysis.

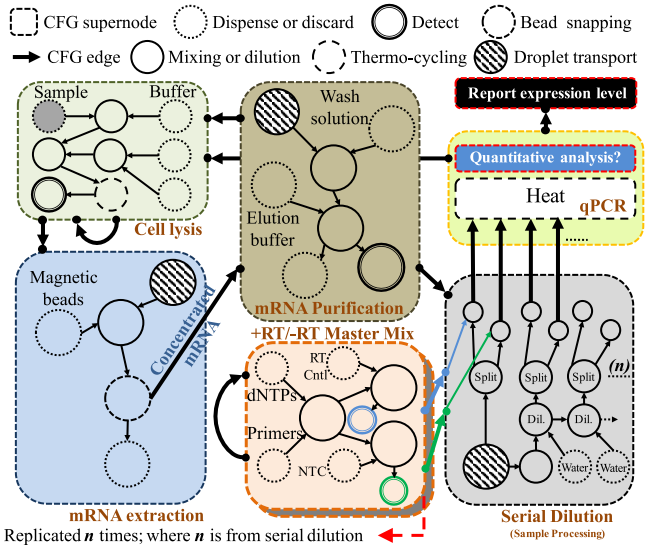

Fig. 8. CFG representation of the gene-expression analysis protocol for a single sample pathway.

interdependencies among these operations. The synthesis tool must be able to synthesize the supernodes of the protocol CFG based on the decisions made for each sample during running time. Fig. 8 illustrates the representation of the protocol used for quantitative analysis of gene expression. Due to the mapping of many bioassays onto this platform, the design of the underlying DMFB must include on-chip devices such as heaters, magnets, and sensors that are required to complete the execution of the protocol. Existing biochip synthesis methods [20], [23], [24] must be extended to handle dynamic adaptation for multiple sample pathways and the inherent uncertainties associated with them. Dynamic resource allocation and spatial reconfiguration in DMFBs are motivated by the following realities about biochemistry protocols.

1) The benchtop characteristics of a contemporary microbiology application (i.e., quantitative analysis), where multiple sample pathways are manipulated concurrently and investigated independently.
2) The need for a robust design of a DMFB, since a non-robust designs may lead to system failure and inefficient quantification.

While we take care of the application characteristics by adopting dynamic resource allocation on DMFBs, the design constraints are enforced through a spatial-reconfiguration approach. Utilizing both mechanisms in a combined manner ensures a robust miniaturization of a microbiology application on a resource-limited DMFB.

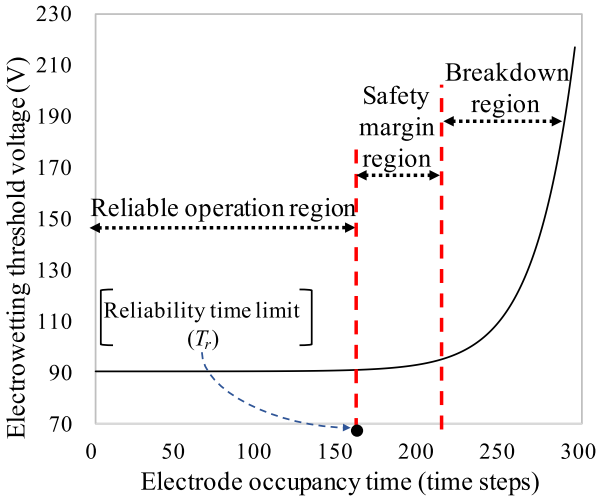

Fig. 9. Electrode degradation model in terms of electrowetting threshold voltage [14].

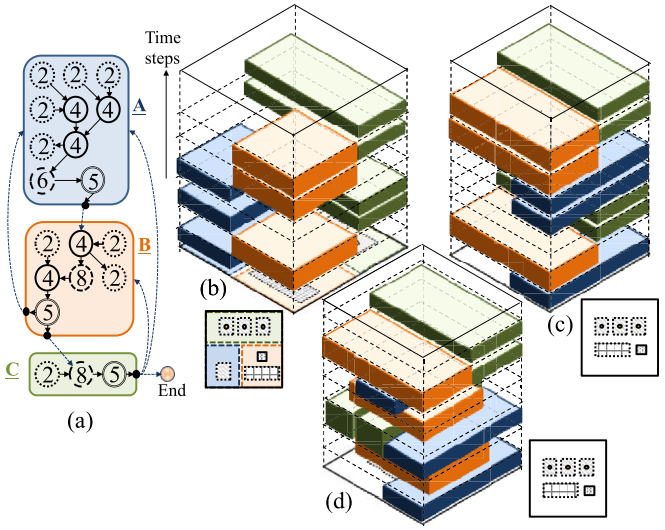

Fig. 10. Illustration of spatial-reconfiguration schemes based on a quantitative protocol described in (a). (b) Nonreconfigurable scheme. (c) Restricted resource-sharing scheme. (d) Unrestricted resource-sharing scheme.

V. Spatial Reconfiguration

A. Resource-Sharing Schemes

Gene-expression analysis using DMFBs relies on the concurrent manipulation of a collection of droplets. Since there is uncertainty about the order of execution of bioassays, and the allocation of on-chip devices among the samples is not known *a priori*, there is a need for a reconfiguration technique that can map specifications of the bioassays to the space of chip resources; we refer to this as *spatial reconfiguration*. We consider three levels of spatial reconfiguration.

1) *Nonreconfigurable Scheme ($\mathcal{NON}$):* This scheme is adopted by current prototypes; on-chip devices are allocated *a priori* [13], [14]. Interbioassay communication is achieved by passing droplets between the dedicated areas.
2) *Restricted Resource Sharing ($\mathcal{RR}$):* The restriction here is in terms of the reconfigurability of the shared devices among bioassays. For instance, a heat-detect device can be shared between qPCR and cell lysis for the purpose of thermal manipulation or even sample processing, but it cannot be used by other bioassays.
3) *Unrestricted Resource Sharing ($\mathcal{NR}$):* In this scheme, no restriction is imposed on resource sharing. Therefore, heat-detect modules cannot only be used for thermal manipulation or sample processing in cell lysis or qPCR, but they can also be utilized by sample processing operations in all bioassays.

B. Motivation for Degradation-Aware Resource Sharing

For a reliable process, mapping the specifications of the bioassays to the space of chip resources must take into account the degradation caused by a bioassay. Norian *et al.* [14] provided a degradation model for the electrodes. An electrode's lifetime can be divided into three regions: 1) *reliable operation*; 2) *safety margin*; and 3) *breakdown* (see Fig. 9). In the reliable operation region, the threshold voltage needed for actuation is constant. The threshold voltage increases linearly in the safety margin region. In the breakdown region, a significant increase in the electrowetting voltage is needed in order to transport a droplet. This increase in voltage, however, quickly leads to dielectric breakdown [47].

The above analysis motivates the need to pinpoint the pros and cons associated with each reconfiguration level. Therefore, we provide an example of mapping a simplified quantitative protocol, which consists of: 1) cell lysis; 2) mRNA isolation and purification; and 3) sample processing and DNA amplification, as shown in Fig. 10(a). Similar to our benchtop experiment, three samples $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$ are concurrently subjected to biochemical reactions in which the three experimental pathways are as follows: 1) $\mathcal{S}_1$ traverses the pathway **A-B-C**; 2) $\mathcal{S}_2$ traverses the pathway **A-B-A-B-C**; and 3) $\mathcal{S}_3$ traverses the pathway **A-B-C-A-B-C**. Fig. 10(b)–(d) shows the mapping of bioassays into the space of resources for $\mathcal{NON}$, $\mathcal{RR}$, and $\mathcal{NR}$, respectively.

As discussed in Section I, the electrodes located underneath the heating region are unique; thus it is imperative to consider the impact of spatial reconfiguration on their lifetime. In Fig. 11, we evaluate the reconfiguration levels based on our example with respect to the following criteria: 1) total completion time; 2) chip size; 3) number of required heaters; and 4) worst-case heater occupancy time, which indicates the worst-case degradation level. Based on this example, it is obvious that the nonreconfigurable scheme is not scalable as it uses larger chip area and more heaters to complete the protocol even though both completion time and occupancy time are low. Also, it is apparent that unrestricted resource sharing achieves minimum completion time, but it reduces chip lifetime. On the other hand, restricted resource sharing decelerates degradation of the chip, but the completion time is higher. Therefore, there is a need for a resource-allocation scheme that combines the best of both worlds; i.e., lower completion time and less degradation of the chip. We refer to this scheme as *degradation-aware resource allocation* and describe it in the next section.

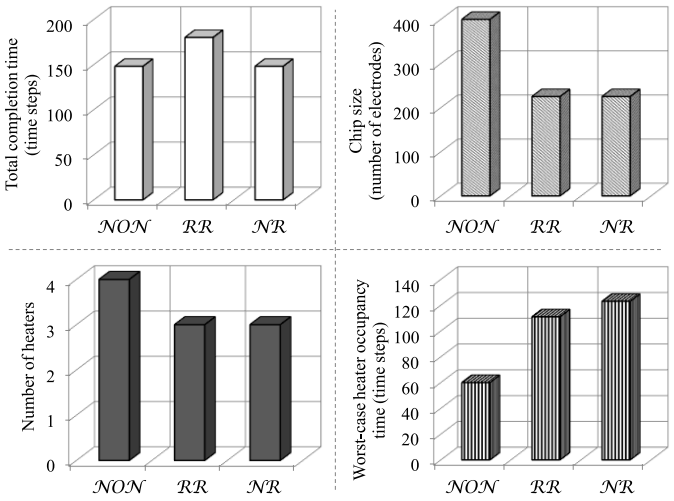

Fig. 11. Comparison of spatial-reconfiguration schemes: 1) $\mathcal{NON}$ refers to the nonreconfigurable scheme; 2) $\mathcal{RR}$ refers to restricted resource sharing; and 3) $\mathcal{NR}$ refers to unrestricted resource sharing.

TABLE I
NOTATIONS USED IN THIS PAPER

B	The complete set of the protocol bioassays
b_i	A bioassay b_i
R	The complete set of the chip modules
R_{sh}	The set of shared, reconfigurable modules
r_i	The set of shared resources that are essential for a bioassay b_i
gr_j	The set of shared resources that are granted to a bioassay b_j
D_i^j	The degradation caused by a bioassay b_i on the shared resources gr_j
T_i^j	The completion time for a bioassay b_i when the resources gr_j are used

VI. SHARED-RESOURCE ALLOCATION

This section formulates the resource-allocation problem and describes the proposed solution.

A. Problem Formulation

The notation used in this paper is listed in Table I. The system described here is composed of three types of modules: 1) *nonreconfigurable* modules (input and output ports); 2) *sample-processing* modules (mixers); and 3) *reconfigurable* modules (heaters, detectors, and magnet regions). Unlike the nonreconfigurable and samples processing modules, the reconfigurable modules are shared among the protocol's bioassays, and access control is managed by the resource allocator. In addition, we make the following important observations.

1) Each bioassay $b_i \in B$ is mapped to a local space of nonreconfigurable and sample processing modules.
2) A shared module $r \in R_{\text{sh}}$ is essential for a bioassay b_i if and only if the absence of this module leads to a failure in execution of this bioassay. An example is the heater resource for a thermal-cycling bioassay.
3) A shared module $r \in R_{\text{sh}}$ that is granted to a bioassay b_i *may not* be essential for the execution of the bioassay, i.e., $r_i \subset gr_i$. Typically, this module can be used for sample processing.

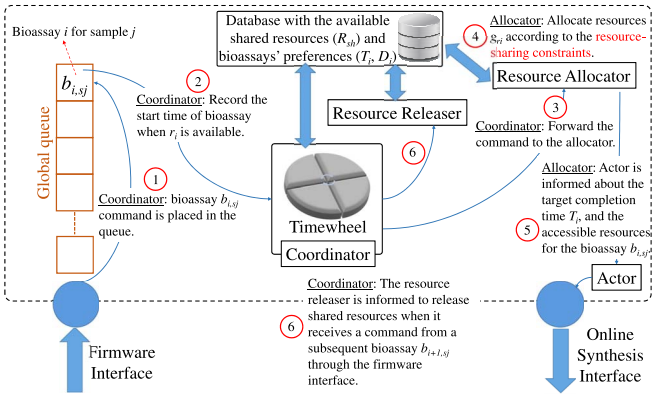

Fig. 12. Components of the shared-resource coordination and allocation.

Our problem formulation is as follows.

Inputs:

1) The protocol CFG $G_c = (V_c, E_c)$, where $V_c = \{v_{c1}, v_{c2}, \ldots, v_{cm}\}$ represents the supernodes of m bioassays and $e_c(v_{ci}, v_{cj})$; $1 \le i, j \le m$ represents data and biological dependency between all pairs of bioassays b_i and b_j. A supernode v_{ci} comprises a directed sequencing graph $G_s = (V_s, E_s)$, where $V_s = \{v_{s1}, v_{s2}, \ldots, v_{sn}\}$ represents n bioassay operations and $e_s(v_{sk}, v_{sl})$; $1 \le k, l \le n$ represents dependencies between all pairs of operations k and l that belong to the bioassay b_i.
2) The digital microfluidic library, which describes the types and locations of the on-chip modules.
3) The resource preferences of every bioassay b_i, which describes the initial resource requirement. The values of the bioassay's completion time T_i^j and the degradation level D_i^j as a function of the granted resources gr_j are also specified.
4) The resource-allocation constraints (Section VI-B).

Output: Allocation of chip modules to the bioassays such that the constraints on resource-allocation are satisfied.

B. Resource-Allocation Sequence and Constraints

We have developed a shared-resource allocation scheme based on a timewheel that is controlled by the *coordinator*, as shown in Fig. 12. The sequence of actions is indicated by the numbers 1–6. Whenever the coordinator receives a command from the firmware layer about the decision for a certain pathway, it firsts stores the command in a global queue until all preceding requests are fulfilled. When the bioassay's command is ready to be processed (essential resources are available), the coordinator forwards it to the *resource allocator* agent. The resource allocator, in turn, checks the preferences of the bioassay and the scheme-specific constraints on resource allocation. Then, the resource allocation is determined and transferred to the *actor* which, in turn, invokes online synthesis for this particular bioassay.

When the restricted resource-sharing scheme is adopted, the resource allocator must ensure that no nonessential shared resource is allocated to the requesting bioassay; i.e., $r_i = gr_j$. For instance, a bioassay for the preparation of master mixes

cannot get access to the heaters, which are not essential for its execution, but it can get access to additional optical detectors to shorten time-to-completion. Therefore, the worst-case computational complexity for allocating resources to a bioassay b_i is $O(|r_i|)$.

Conversely, the resource allocator in unrestricted resource sharing enables the option of using nonessential shared resources. In this case, these resources will be used for sample processing. For instance, an mRNA extraction bioassay can get access to a heater to perform sample processing in order to shorten its completion time. As a result, the worst-case computational complexity for allocating resources to a bioassay b_i is $O(|R_{\text{sh}}|)$.

Finally, degradation-aware resource-allocation method initially allocates shared resources to requesting bioassays without restrictions. It also keeps track of the actual degradation levels at all shared sources resulting from the synthesized bioassays. This is achieved via direct communication with the synthesis tool. As a result, when the reliable operation time for the electrodes at a certain shared resource is exceeded, the resource allocator imposes restrictions on accessing this resource; i.e., it switches to restricted resource sharing for this particular resource. Since electrode degradation is considered in resource allocation, resources are sorted according to their degradation level whenever a bioassay b_i requires resource allocation. Hence, the worst-case computational complexity of this scheme is $O(|R_{\text{sh}}|^2)$.

While our methodology is the first to consider dynamic resource allocation in DMFBs, the literature of real-time embedded systems is rich with advanced and cost-effective middleware designs that provide dynamic adaptation [48]–[50]. Objectives such as fault tolerance, reduction of power consumption, and reliability of safety-critical applications have been the focus of these designs. Similarly, we foresee many research opportunities for providing optimized dynamic-adaptation techniques in DMFBs.

VII. Firmware for Quantitative Analysis

The firmware is an interactive software layer that needs to be incorporated in the architecture of cyberphysical DMFBs. This layer is constructed based on a set of C++ libraries that can collect data from on-chip sensors via a signal tracer library, use the collected data to perform real-time data analysis, and feed the upper layer with appropriate decisions. Previous studies overlooked the incorporation of such a layer since the demonstrated designs of cyberphysical DMFBs solely focused on error recovery. The feedback system therefore uses simple signal conditioning and checking schemes to provide decisions. However, advancing cyberphysical DMFBs toward automated quantitative analysis requires taking into consideration the challenges imposed by the distributed sensors. In this section, we focus on the design of the firmware component required for evaluating protocol efficiency and the level of gene expression.

A. Firmware Components

Fig. 13 shows an example of a block diagram for the firmware layer that is used for qPCR-based quantitative gene-expression analysis. This layer consists of two libraries: 1) tracing and queuing (TQ) library and 2) analysis and decision-making (AD) library. Since multiple sensors are used in the quantification, the TQ library is responsible for interacting with the hardware. Using the sensor-sample map that is generated by the resource allocator, this library can trace the real-time readout of each sensor at every amplification cycle. The synchronization between the library and the underlying sensors is achieved via a designated protocol stack that is much simpler than its counterpart in wireless sensor networks. The circuitry for each sensor must transmit the readout, packetized along with the sensor identification, to the firmware. Next, the firmware matches the received identification information with the sensor-sample map to recognize the detected sample. The received signal is then logged into the library's queuing system at the appropriate location.

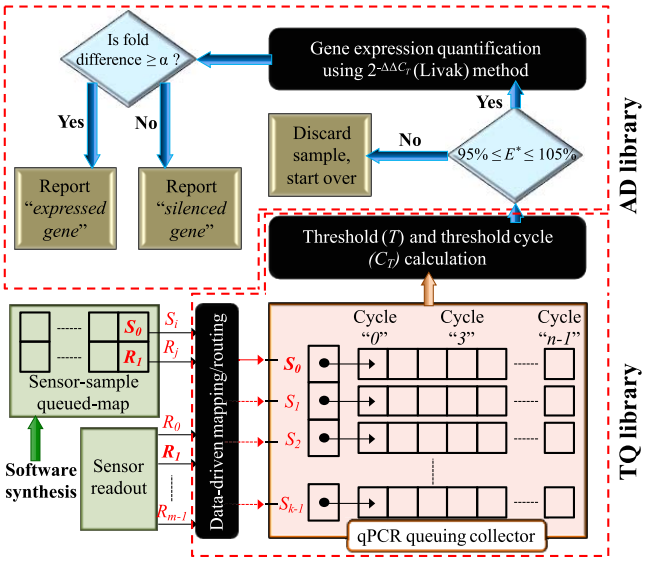

Fig. 13. Structure of the firmware layer for qPCR-based gene-expression analysis.

When the amplification data is collected, there is a need to use this data to provide useful information. Therefore, the TQ library generates the standard curve for the assessment of the amplification quality. Next, TQ library invokes the AD library to trigger the start of AD. The targets of this library is to execute gene-expression quantification methods. As shown in Fig. 13, the captured threshold cycles of the samples, communicated from the TQ library, are used to draw the standard curve and determine the efficiency factor E^* [41]. Based on the resulting value of the efficiency factor, the firmware will command the upper layer (that forms the feedback system) to transition the platform into a new start or to report the final result. If the first path is chosen, the command will trigger online synthesis at the upper layer. On the other hand, if the latter path is chosen, the AD library invokes the Livak procedure [43] to quantify the expression of the gene of interest based on the communicated threshold cycles.

Note that the firmware components, namely TQ and AD libraries, are customized for qPCR-based analysis protocols.

However, these components may be redesigned when applied to different classes of quantitative protocols (e.g., glucose colorimetric analysis [19]). In principle, methods of data acquisition, queuing, and processing rely on the type of target molecule (e.g., DNA, protein, or blood cells), the associated sensor technology (e.g., fluorescence or light absorption), and the property under investigation (e.g., abundance of gene expression or concentration). In other words, there is a practical limitation against defining a generic firmware design due to the increasing level of complexity and specificity in the microbiology protocols.

B. Firmware Operation With Amplification Variations

The firmware is a key layer, not only because of its role in collecting sensor data and conveying decisions, but it is also responsible for ensuring that the reported data is meaningful and precise to a large extent. Impreciseness in reporting the level of gene expression might arise due to the following reasons.

1) Indeterministic component interactions are ubiquitous in biological processes; as a result, DNA amplification response for the same sample may witness a limited random deviation over several trials [51].
2) Similar to the IC design process, the fabrication process of a DMFB and the integrated devices (e.g., heaters and detectors) are subject to process variations and faults [52].

Unreliable design of the firmware will undoubtedly propagate these inefficiencies to the user side through imprecise gene-expression results.

An effective solution to this problem is based on a probabilistic approach. The protocol experiment is iterated several times, and the results with the highest probability of being precise are reported. A simple probabilistic online approach is to use the mean of the collected data to directly report the results [14]. However, a more accurate scheme will consider the fact that, like several other biological processes [53], the trend of deviation in DNA amplification follows a certain distribution function. With several offline trials, the parameters of such a function are identified. Next, the outcomes of an online experiment (e.g., the level of gene expression) can be probabilistically determined. The firmware design incorporates a tool, called RAmp,[1] that carries out such optimization and reports the results based on a probabilistic distribution. Fig. 14 illustrates the working principle of RAmp assuming a normal distribution over C_T. The acceptable range of $C_{T,\text{on}}$[2] can be calculated given a predetermined threshold on the cumulative distribution function $F(C_T)$. For example, the acceptable range can be associated with the following condition: $F(C_{T,2}) - F(C_{T,1}) = 0.5$. Note that the parameters of the constructed model can be fixed along the spectrum of C_T; this approach is referred to as static-variation model ($\mathcal{SM}$), or they can be curve-fitted to a polynomial function over C_T; we refer to this approach as dynamic-variation model ($\mathcal{DM}$). Additional research and experiments are needed to investigate such probabilistic models through decoupling the effect of indeterminism in biological processes and the impact of process variation in biochip fabrication and operation.

[1]RAmp stands for reliable computation with amplification variations.
[2]We use $C_{T,\text{on}}$ to distinguish online C_T values.

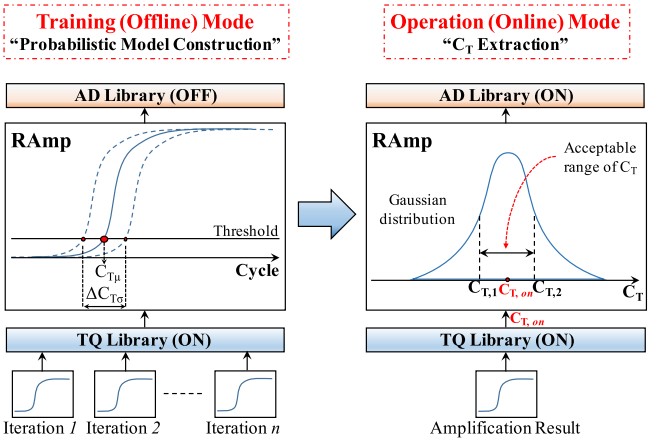

Fig. 14. Using RAmp to: 1) construct a probabilistic model based on amplification variations (offline mode) and 2) specify an acceptable range of experimental C_T based on the constructed model (online mode).

VIII. Simulation Results

We implemented the proposed resource allocation schemes using C++. All evaluations were carried out using a 2.4 GHz Intel Core i5 CPU with 4 GB RAM. In order to obtain the resource preferences for each bioassay before resource allocation begins, synthesis results were obtained from offline simulation using [54]. The same tool is used for online synthesis when the simulation of real-time quantitative analysis starts. Since this is the first work on optimization for multiple sample pathways, a comparison with prior work is not feasible.

A. Evaluation of Resource-Allocation Schemes

Two chip arrays have been utilized to simulate the quantitative protocol; these chips are labeled $\mathcal{C}_{\text{NR}}$ (nonreconfigurable chip) and $\mathcal{C}_R$ (reconfigurable chip). In both arrays, the chip has a dedicated port for each liquid and each bioassay also has its own sample processing (SP) region. However, reconfigurable modules, namely heaters (H), magnet (M), CCD camera region (CCD), and optical detectors (OD), are integrated in the shared area of $\mathcal{C}_R$. This area is accessible to droplets from any bioassay via a ring electrode-bus. The chip sizes are 18×18 and 17×17 for $\mathcal{C}_{\text{NR}}$ and $\mathcal{C}_R$, respectively; these represent the lower bounds on the size required to execute the bioassays. We expect that increasing the array size for $\mathcal{C}_{\text{NR}}$ and $\mathcal{C}_R$ will result in a proportional decrease in completion time. Performance assessment for different chip sizes is left for future work.

We evaluate four resource-allocation schemes: 1) nonreconfigurable scheme ($\mathcal{NON}$); 2) restricted resource sharing ($\mathcal{RR}$); 3) unrestricted resource sharing ($\mathcal{NR}$); and 4) adaptive degradation-aware resource sharing ($\mathcal{DA}$). Using these schemes, we run simulations on the quantitative analysis protocol described in Section I. We consider three samples being concurrently subjected to fluidic operations. The samples are

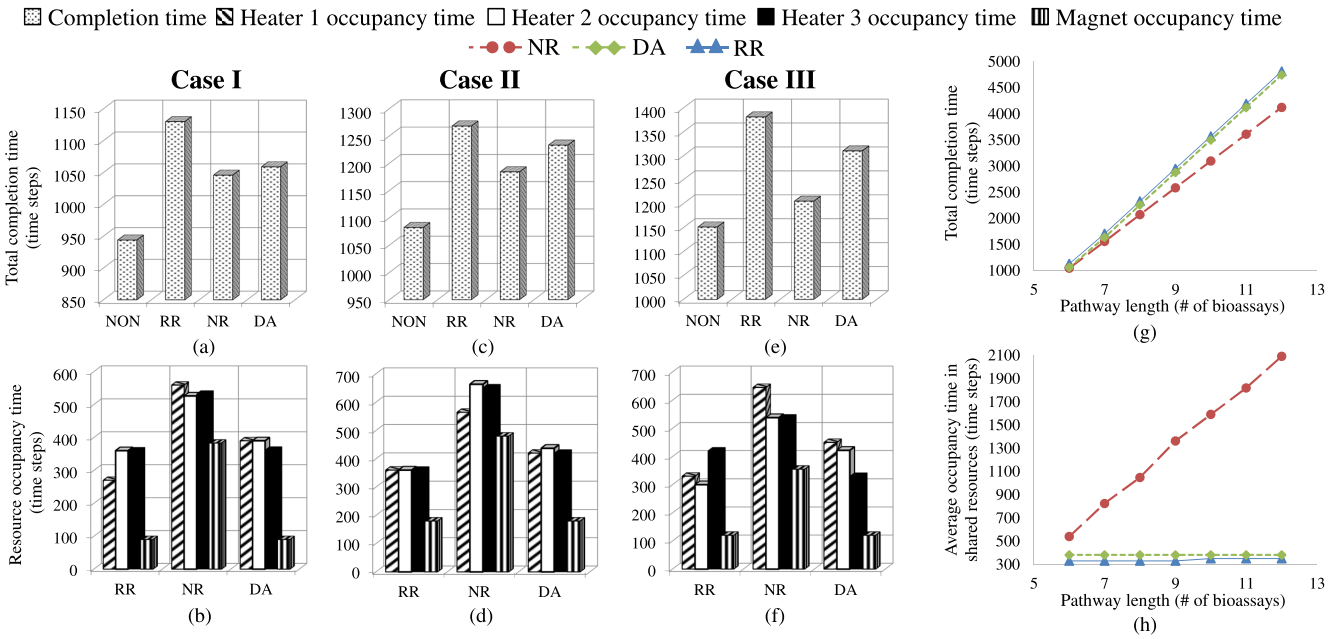

Fig. 15. (a)–(f) Comparison between the four resource-allocation schemes—Nonreconfigurable ($\mathcal{NON}$), restricted resource sharing ($\mathcal{RR}$), unrestricted resource sharing ($\mathcal{NR}$), and degradation-aware resource sharing ($\mathcal{DA}$)—executing three pathways. (g) and (h) Performance of the shared resource-allocation schemes with various lengths of homogeneous pathways.

TABLE II
BIOASSAY NOTATION AND RESOURCE REQUIREMENT

Bioassay	Notation	Minimum resource requirement
Cell Lysis	**CL**	SP, H, and CCD
mRNA Extraction	**mE**	SP and M
mRNA Purification	**mP**	SP and OD
RT Master Mix	**MM**	SP and OD
Serial Dilution	**SD**	SP
Thermal Cycling	**TC**	H

S_1 (GFP gene-targeted sample), S_2 (YFP gene-targeted sample, and S_3 (actin gene-targeted sample). Using the notation in Table II, we consider three different cases in terms of the sample pathways.

1) *Short Homogeneous Pathways (Case I):* An optimistic case in which all the three samples follow the same shortest pathway (CL-mE-mP-MM-SD-TC).
2) *Long Homogeneous Pathways (Case II):* A pessimistic case in which all the three samples follow the same long pathway (CL-mE-mP-CL-mE-mP-MM-SD-TC).
3) *Heterogeneous Pathways (Case III):* A realistic case in which the pathways of these samples are different.

The considered pathways are as follows: CL-mE-mP-MM-SD-TC, CL-mE-mP-CL-mE-mP-MM-SD-TC, and CL-CL-mE-mP-MM-MM-SD-TC, respectively. Table II lists the minimum resource requirements for the bioassays.

The metrics of comparison include: 1) the total completion time of the protocol execution (in time steps) and 2) the occupancy time (degradation level) for the shared resources (heaters and magnet modules). A time-step refers to the clock period, typically in the range of 0.1–1 s [6]. For fair comparison between nonreconfigurable and resource-sharing schemes, the chip array used with the nonreconfigurable scheme (i.e., $\mathcal{C}_{\text{NR}}$) is designed such that the resources are sufficient only for a single pathway; therefore, the resources are not replicated with the number of samples. In addition, due to the absence of resource sharing in the nonreconfigurable scheme, the occupancy time is not considered in our analysis. Table III compares the four scheme based on the chip configuration. As expected, due to the absence of resource sharing in the nonreconfigurable scheme, the chip modules are replicated. For example, three heaters are sufficient for the resource-sharing schemes (in $\mathcal{C}_R$) for thermal manipulation in different bioassays. For $\mathcal{C}_{\text{NR}}$, both cell lysis and thermal cycling bioassays require their own heaters, thus increasing chip fabrication cost.

TABLE III
NUMBER OF ON-CHIP MODULES AND ARRAY ELECTRODES IN $\mathcal{C}_{\text{NR}}$ AND $\mathcal{C}_R$

Resource	# Modules		# Electrodes	
	$\mathcal{C}_{NR}$	$\mathcal{C}_R$	$\mathcal{C}_{NR}$	$\mathcal{C}_R$
SP	1 (CL), 2 (mE), 2 (mP), 2 (MM), 4 (SD)		176	176
H	2 (CL), 3 (TC)	3	80	48
M	1 (mE)	1	18	18
OD	2 (mP), 2 (MM)	2	36	18
CCD	1 (CL)	1	9	9
Chip size	22	18	18×18	17×17

Based on our simulations, the CPU time, which includes the time for resource allocation and online synthesis, averaged over the three cases (I–III), is less than 1 ms for all allocation schemes. Thus, compared to the protocol completion time (in the order of minutes), the CPU time is negligible.

1) Case I (Short Homogeneous Pathways): This case arises when all the samples are perfectly grown in a well-controlled medium. In addition, the reagents are contamination-free. We compare the four methods in terms of completion times [see Fig. 15(a)]. As expected, nonreconfigurable resource allocation leads to the shortest completion time. Restricted resource sharing, on the other hand, shows the worst completion time.

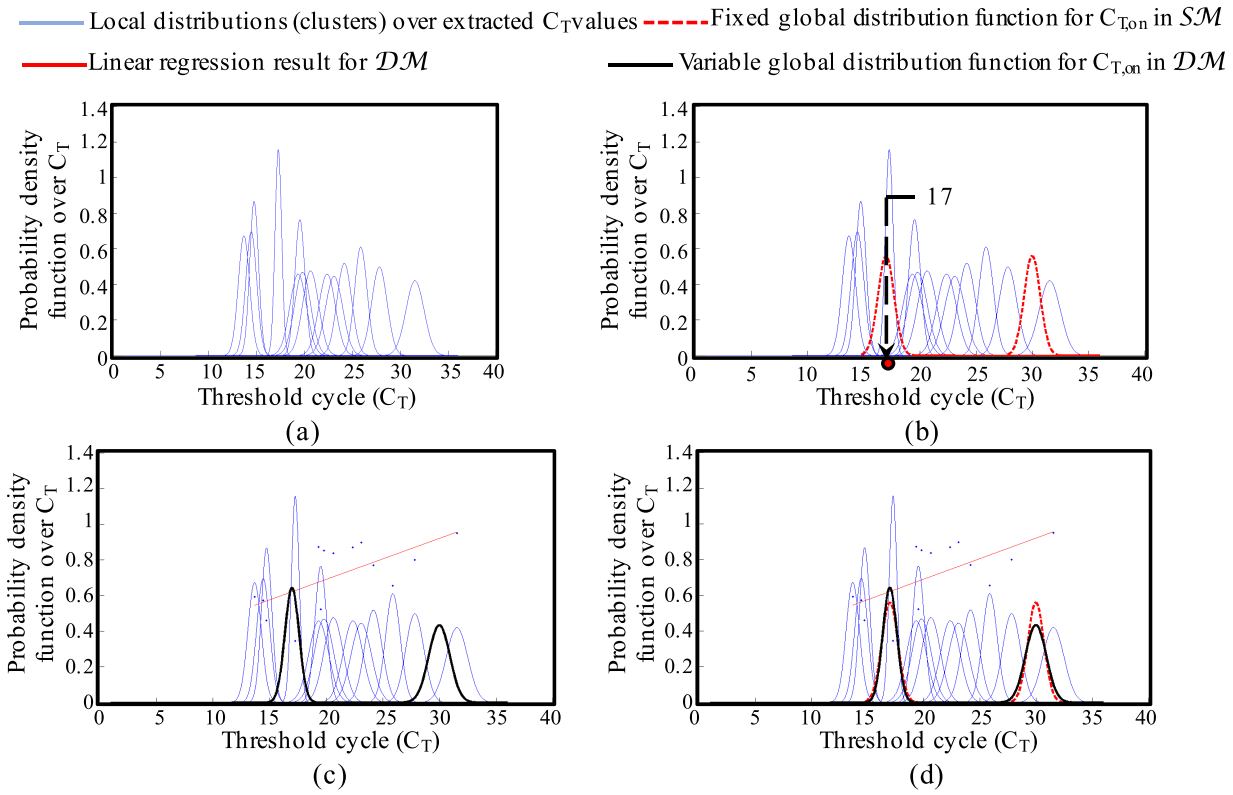

Fig. 16. Illustration of training and operation methodologies in RAmp.

We also note that we can use adaptive, degradation-aware resource allocation ($\mathcal{DA}$) to achieve a short completion time, while the chip resources are not severely degraded (degradation is measured in terms of the occupancy time) [see Fig. 15(b)].

2) Case II (Long Homogeneous Pathways): We next study the case where all the samples are resuspended for additional bioassays. As shown in Fig. 15(c) and (d), the proposed schemes show the same profiles of completion times and degradation levels as in case I, but with higher values. However, the completion time for $\mathcal{DA}$ is closer to that obtained with restricted resource sharing.

3) Case III (Heterogeneous Pathways): This is a realistic case that emerges due to the inherent uncertainty about the biological contents of each sample. It introduces the challenge of deciding the allocation of resources among the heterogeneous pathways at runtime. Again, we evaluate the allocation schemes based on the completion time [Fig. 15(e)] and the degradation level [Fig. 15(f)].

4) Convergence of Degradation-Aware Resource Allocation: Finally, we study the convergence of the $\mathcal{DA}$ method. We analyze the completion time and the average degradation level of the shared resource-allocation schemes with various lengths of homogeneous pathways, as shown in Fig. 15(g) and (h). We observe that the completion time for $\mathcal{DA}$ begins to converge to its counterpart in $\mathcal{RR}$ when the length of the pathway is increased. However, due to the restrictions imposed by both $\mathcal{RR}$ and $\mathcal{DA}$ on resource allocation, the degradation levels remain below a certain limit even when we increase the pathway length substantially, as shown in Fig. 15(h). In real-life scenarios, chip users tend to make optimizations in order to make sure that the protocol is finished as early as possible to avoid droplet evaporation [55]. These scenarios make the $\mathcal{DA}$ method especially attractive in practice.

B. Variants of Probabilistic Models in RAmp

We demonstrate two variants for probabilistic-model construction and the use of the firmware: 1) $\mathcal{SM}$ and 2) $\mathcal{DM}$. The following details describe the methodology for training and operation for RAmp.

1) Input Data Set (For Training): We used the C_T values of β-actin that were extracted from the benchtop experiment (Fig. 5) as the input data set. Additional values were randomly generated, but clustered to be near the measured data, and included to the set of extracted points in order to enlarge the training set. Therefore, the amplification of each sample droplet is characterized by a "cluster" of possible readings.

2) Probabilistic Models: The input data set for each cluster was fit to a local normal function, as shown in Fig. 16(a). Next, the standard deviation of the constructed local functions were averaged over C_T spectrum to create a global normal distribution function that is used during operation mode ($\mathcal{SM}$ approach) [see Fig. 16(b)]. In other words, this global function ($\sigma = 0.7139$) is always used to represent the acceptable range for any input amplification result $C_{T,\text{on}}$. On the other hand, to generate $\mathcal{DM}$, the obtained values of standard deviation were curve-fitted furthermore to a function (in our case, it is a first-order polynomial function), as shown in Fig. 16(c).

Based on the training, the standard deviation of the global distribution function will change as $C_{T,\text{on}}$ varies, depending on the deviation trend of the original data clusters. Fig. 16(d) compares the operation outcomes of $\mathcal{SM}$ and $\mathcal{DM}$ for two amplification results ($C_{T,\text{on}} = 17$ and 30). At $C_{T,\text{on}} = 17$, there is a high chance that this result matches the original input data at $C_T = 17.29$. Accordingly, using $\mathcal{DM}$ leads to a steepened global curve, reducing the standard deviation at this point ($\sigma = 0.6211$). However, at $C_{T,\text{on}} = 30$, the $\mathcal{DM}$ flattens the curve ($\sigma = 0.9199$) in an attempt to include the original input data points at $C_T = 27.84$ and 31.56 in the reported probability distribution. Note that the above points represent corner cases, typical cases such as the range $C_{T,\text{on}} : 20 \rightarrow 25$ will exhibit the same behavior for both $\mathcal{SM}$ and $\mathcal{DM}$. Additional statistical research is needed to improve the characteristics of the curve-fitted function. There is a need to reflect the disturbances in qPCR parameters and kinetics of the reaction on this function via stochastic modeling [56].

IX. Conclusion

We have introduced the first automated design method for a cyberphysical DMFB that performs quantitative gene-expression analysis. This design is based on a spatial-reconfiguration technique that incorporates resource-sharing specifications into the software synthesis flow. A firmware has been developed to collect data from sensors during runtime, perform analysis, and make decisions about the multiple pathways for samples. We have also presented an adaptive scheme for shared-resource allocation that efficiently utilizes on-chip modules. This method manages resource access between the bioassays. The proposed scheme has been evaluated based on the completion time and the electrode degradation level for realistic multisample test cases. We have reported results on the performance of the allocation scheme for various pathway lengths.

References

[1] Y. Développment. (2013). *Microfluidic Applications in the Pharmaceutical, Life Sciences, in Vitro Diagnostic and Medical Device Markets*. [Online]. Available: http://www.yole.fr/iso_upload/News/2013/PR_MicrofluidicsApplications_YOLE%20DEVELOPPEMENT_June2013.pdf

[2] Illumina. (2015). *Illumina NeoPrep Library Prep System*. [Online]. Available: http://www.illumina.com/systems/neoprep-library-system.html/

[3] R. Sista *et al.*, "Development of a digital microfluidic platform for point of care testing," *Lab Chip*, vol. 8, no. 12, pp. 2091–2104, 2008.

[4] D. Bogojevic, M. D. Chamberlain, I. Barbulovic-Nad, and A. R. Wheeler, "A digital microfluidic method for multiplexed cell-based apoptosis assays," *Lab Chip*, vol. 12, no. 3, pp. 627–634, 2012.

[5] H.-H. Shen, S.-K. Fan, C.-J. Kim, and D.-J. Yao, "EWOD microfluidic systems for biomedical applications," *Microfluid. Nanofluid.*, vol. 16, no. 5, pp. 965–987, 2014.

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

[7] N. M. Jokerst *et al.*, "Progress in chip-scale photonic sensing," *IEEE Trans. Biomed. Circuits Syst.*, vol. 3, no. 4, pp. 202–211, Aug. 2009.

[8] Y.-J. Shin and J.-B. Lee, "Machine vision for digital microfluidics," *Rev. Sci. Instrum.*, vol. 81, no. 1, 2010, Art. no. 014302.

[9] Y. Luo, K. Chakrabarty, and T.-Y. Ho, "Error recovery in cyberphysical digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 1, pp. 59–72, Jan. 2013.

[10] N. Bontoux *et al.*, "Integrating whole transcriptome assays on a lab-on-a-chip for single cell gene profiling," *Lab Chip*, vol. 8, no. 3, pp. 443–450, 2008.

[11] Y. Gong, A. O. Ogunniyi, and J. C. Love, "Massively parallel detection of gene expression in single cells using subnanolitre wells," *Lab Chip*, vol. 10, no. 18, pp. 2334–2337, 2010.

[12] Z. Hua *et al.*, "Multiplexed real-time polymerase chain reaction on a digital microfluidic platform," *Anal. Chem.*, vol. 82, no. 6, pp. 2310–2316, 2010.

[13] A. Rival *et al.*, "An EWOD-based microfluidic chip for single-cell isolation, mRNA purification and subsequent multiplex qPCR," *Lab Chip*, vol. 14, no. 19, pp. 3739–3749, 2014.

[14] H. Norian, R. M. Field, I. Kymissis, and K. L. Shepard, "An integrated CMOS quantitative-polymerase-chain-reaction lab-on-chip for point-of-care diagnostics," *Lab Chip*, vol. 14, no. 20, pp. 4076–4084, 2014.

[15] P.-Y. Hung, P.-S. Jiang, E.-F. Lee, S.-K. Fan, and Y.-W. Lu, "Genomic DNA extraction from whole blood using a digital microfluidic (DMF) platform with magnetic beads," *Microsyst. Technol.*, pp. 1–8, Apr. 2015.

[16] B. S. Wheeler, B. T. Ruderman, H. F. Willard, and K. C. Scott, "Uncoupling of genomic and epigenetic signals in the maintenance and inheritance of heterochromatin domains in fission yeast," *Genetics*, vol. 190, no. 2, pp. 549–557, 2012.

[17] K. Hu, B.-N. Hsu, A. Madison, K. Chakrabarty, and R. Fair, "Fault detection, real-time error recovery, and experimental demonstration for digital microfluidic biochips," in *Proc. IEEE/ACM Design Autom. Test Europe (DATE)*, Grenoble, France, 2013, pp. 559–564.

[18] H. Ren, R. B. Fair, and M. G. Pollack, "Automated on-chip droplet dispensing with volume control by electro-wetting actuation and capacitance metering," *Sens. Actuators B Chem.*, vol. 98, nos. 2–3, pp. 319–327, 2004.

[19] V. Srinivasan, V. K. Pamula, and R. B. Fair, "Droplet-based microfluidic lab-on-a-chip for glucose detection," *Anal. Chimica Acta*, vol. 507, no. 1, pp. 145–150, 2004.

[20] F. Su and K. Chakrabarty, "High-level synthesis of digital microfluidic biochips," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 3, no. 4, 2008, Art. no. 1.

[21] F. Su, W. Hwang, and K. Chakrabarty, "Droplet routing in the synthesis of digital microfluidic biochips," in *Proc. IEEE/ACM Design Autom. Test Europe (DATE)*, vol. 1. Munich, Germany, 2006, pp. 1–6.

[22] P.-H. Yuh, C.-L. Yang, and Y.-W. Chang, "Placement of digital microfluidic biochips using the T-tree formulation," in *Proc. IEEE/ACM Design Autom. Conf. (DAC)*, San Francisco, CA, USA, 2006, pp. 931–934.

[23] D. T. Grissom and P. Brisk, "Fast online synthesis of digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 33, no. 3, pp. 356–369, Mar. 2014.

[24] T.-W. Huang, S.-Y. Yeh, and T.-Y. Ho, "A network-flow based pin-count aware routing algorithm for broadcast-addressing EWOD chips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 12, pp. 1786–1799, Dec. 2011.

[25] T. Xu, W. L. Hwang, F. Su, and K. Chakrabarty, "Automated design of pin-constrained digital microfluidic biochips under droplet-interference constraints," *ACM J. Emerg. Technol. Comput.*, vol. 3, no. 3, p. 14, 2007.

[26] M. Alistar, P. Pop, and J. Madsen, "Redundancy optimization for error recovery in digital microfluidic biochips," *Design Autom. Embedded Syst.*, vol. 19, no. 1, pp. 129–159, 2015.

[27] C. Jaress, P. Brisk, and D. Grissom, "Rapid online fault recovery for cyber-physical digital microfluidic biochips," in *Proc. IEEE VLSI Test Symp. (VTS)*, Napa, CA, USA, 2015, pp. 1–6.

[28] Y. Luo, B. B. Bhattacharya, T.-Y. Ho, and K. Chakrabarty, "Design and optimization of a cyberphysical digital-microfluidic biochip for the polymerase chain reaction," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 34, no. 1, pp. 29–42, Jan. 2015.

[29] J. Gao *et al.*, "An intelligent digital microfluidic system with fuzzy-enhanced feedback for multi-droplet manipulation," *Lab Chip*, vol. 13, no. 3, pp. 443–451, 2013.

[30] K. Hu, M. Ibrahim, L. Chen, Z. Li, K. Chakrabarty, and R. Fair, "Experimental demonstration of error recovery in an integrated cyber-physical digital-microfluidic platform," in *Proc. IEEE Biomed. Circuits Syst. (BIOCAS)*, Atlanta, GA, USA, 2015, pp. 418–421.

[31] J. Lovén *et al.*, "Revisiting global gene expression analysis," *Cell*, vol. 151, no. 3, pp. 476–482, 2012.

[32] A. Niemz, T. M. Ferguson, and D. S. Boyle, "Point-of-care nucleic acid testing for infectious diseases," *Trends Biotechnol.*, vol. 29, no. 5, pp. 240–250, 2011.

[33] K. Hsieh, A. S. Patterson, B. S. Ferguson, K. W. Plaxco, and H. T. Soh, "Rapid, sensitive, and quantitative detection of pathogenic DNA at the point of care through microfluidic electrochemical quantitative loop-mediated isothermal amplification," *Angewandte Chemie*, vol. 124, no. 20, pp. 4980–4984, 2012.
[34] P. Liu, X. Li, S. A. Greenspoon, J. R. Scherer, and R. A. Mathies, "Integrated DNA purification, PCR, sample cleanup, and capillary electrophoresis microchip for forensic human identification," *Lab Chip*, vol. 11, no. 6, pp. 1041–1048, 2011.
[35] J. D. Watson *et al.*, *Molecular Biology of the Gene*, 7th ed. Boston, MA, USA: Pearson, 2013.
[36] G. H. Reference. (May 2015). *What Are Proteins and What Do They Do?* [Online]. Available: http://ghr.nlm.nih.gov/handbook/howgeneswork/protein
[37] E. Hood, "RNAi: What's all the noise about gene silencing?" *Environ. Health Perspect.*, vol. 112, no. 4, pp. A224–A229, 2004.
[38] V. Marx, "Cell culture: A better brew," *Nature*, vol. 496, no. 7444, pp. 253–258, 2013.
[39] T. F. Scientific. (2011). *NanoDrop Spectrophotometers—Assessment of Nucleic Acid Purity*. [Online]. Available: http://www.nanodrop.com/Library/T042-NanoDrop-Spectrophotometers-Nucleic-Acid-Purity-Ratios.pdf
[40] S. A. Bustin, *A-Z of Quantitative PCR*, 1st ed. San Diego, CA, USA: Int. Univ. Line, 2004.
[41] M. W. Pfaffl, "Quantification strategies in real-time PCR," in *A-Z of Quantitative PCR*, S. A. Bustin, Ed. San Diego, CA, USA: Int. Univ. Line, 2004.
[42] Bio-Rad. (2006). *Real-Time PCR Applications Guide*. [Online]. Available: http://www.bio-rad.com/webroot/web/pdf/lsr/literature/Bulletin_5279.pdf
[43] K. J. Livak and T. D. Schmittgen, "Analysis of relative gene expression data using real-time quantitative PCR and the $2^{-\Delta\Delta C_T}$ method," *Methods*, vol. 25, no. 4, pp. 402–408, 2001.
[44] Y. Xie and W. Wolf, "Allocation and scheduling of conditional task graph in hardware/software co-synthesis," in *Proc. IEEE/ACM Design Autom. Test Europe*, Munich, Germany, 2001, pp. 620–625.
[45] P. Pop, P. Eles, and Z. Peng, *Analysis and Synthesis of Distributed Real-Time Embedded Systems*. New York, NY, USA: Springer, 2013.
[46] D. Wu, B. M. Al-Hashimi, and P. Eles, "Scheduling and mapping of conditional task graph for the synthesis of low power embedded systems," *IEE Proc. Comput. Digit. Tech.*, vol. 150, no. 5, pp. 262–273, Sep. 2003.
[47] A. G. Papathanasiou and A. G. Boudouvis, "Manifestation of the connection between dielectric breakdown strength and contact angle saturation in electrowetting," *Appl. Phys. Lett.*, vol. 86, no. 16, 2005, Art. no. 164102.
[48] T. E. Bihari and K. Schwan, "Dynamic adaptation of real-time software," *ACM Trans. Comput. Syst.*, vol. 9, no. 2, pp. 143–174, 1991.
[49] R. Adler, I. Schaefer, M. Trapp, and A. Poetzsch-Heffter, "Component-based modeling and verification of dynamic adaptation in safety-critical embedded systems," *ACM Trans. Embedded Comput. Syst.*, vol. 10, no. 2, p. 20, 2010.
[50] L. A. Rozo Duque, J. M. Monsalve Diaz, and C. Yang, "Improving MPSoC reliability through adapting runtime task schedule based on time-correlated fault behavior," in *Proc. IEEE/ACM Design Autom. Test Europe (DATE)*, Grenoble, France, 2015, pp. 818–823.
[51] M. Sivaganesan *et al.*, "A Bayesian method for calculating real-time quantitative PCR calibration curves using absolute plasmid DNA standards," *BMC Bioinformat.*, vol. 9, no. 1, p. 120, 2008.
[52] F. Su, S. Ozev, and K. Chakrabarty, "Ensuring the operational health of droplet-based microelectrofluidic biosensor systems," *IEEE Sensors J.*, vol. 5, no. 4, pp. 763–773, Aug. 2005.
[53] J. W. Haefner, *Modeling Biological Systems: Principles and Applications*. New York, NY, USA: Springer, 2005.
[54] D. Grissom and P. Brisk, "A field-programmable pin-constrained digital microfluidic biochip," in *Proc. IEEE/ACM Design Autom. Conf. (DAC)*, Austin, TX, USA, 2013, pp. 1–9.
[55] M. J. Jebrail *et al.*, "A solvent replenishment solution for managing evaporation of biochemical reactions in air-matrix digital microfluidics devices," *Lab Chip*, vol. 15, no. 1, pp. 151–158, 2015.
[56] R. G. Rutledge and D. Stewart, "A kinetic-based sigmoidal model for the polymerase chain reaction and its application to high-capacity absolute quantitative real-time PCR," *BMC Biotechnol.*, vol. 8, no. 1, p. 47, 2008.

Mohamed Ibrahim (S'13) received the B.Sc. (Hons.) degree in electrical engineering and the M.Sc. degree from Ain Shams University, Cairo, Egypt, in 2010 and 2013, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, USA.

He was appointed as a Research and Teaching Assistant with the Faculty of Engineering, Ain Shams University. In 2014, he joined Prof. Chakrabarty's Laboratory to work on the design automation and test of next-generation cyberphysical digital-microfluidic biochips. His current research interests include security aspects of microfluidic devices and big-data analytics for microbiology applications.

Krishnendu Chakrabarty (F'08) received the B. Tech. degree from the Indian Institute of Technology Kharagpur, Kharagpur, India, 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 William H. Younger Distinguished Professor of Engineering with the Department of Electrical and Computer Engineering and a Professor of Computer Science with Duke University, Durham, NC, USA, where he also serves as the Director of Graduate Studies for Electrical and Computer Engineering. He is a Hans Fischer Senior Fellow with the Institute for Advanced Studies, Technical University of Munich, Munich, Germany. His current research interests include testing and design-for-testability of integrated circuits, digital microfluidics, biochips, and cyberphysical systems, optimization of enterprise systems, and smart manufacturing.

Prof. Chakrabarty was a recipient of the National Science Foundation Early Faculty (CAREER) Award, the Office of Naval Research Young Investigator Award, the Humboldt Research Award from the Alexander von Humboldt Foundation, Germany, the 2008 Duke University Graduate School Dean's Award for Excellence in Mentoring, the 2010 Capers and Marion McDonald Award for Excellence in Mentoring and Advising, Pratt School of Engineering, Duke University, the Distinguished Alumnus Award from the Indian Institute of Technology Kharagpur, in 2014, the IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems Donald O. Pederson Best Paper Award in 2015, the IEEE Computer Society Technical Achievement Award in 2015, and 11 best paper awards at major IEEE conferences. He served as the Editor-in-Chief for the IEEE DESIGN AND TEST OF COMPUTERS from 2010 to 2012 and *ACM Journal on Emerging Technologies in Computing Systems* from 2010 to 2015. He currently serves as the Editor-in-Chief for the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS. He is also an Associate Editor of the IEEE TRANSACTIONS ON COMPUTERS, the IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS, the IEEE TRANSACTIONS ON MULTISCALE COMPUTING SYSTEMS, and *ACM Transactions on Design Automation of Electronic Systems*. He served as a Distinguished Visitor of the IEEE Computer Society from 2005 to 2007 and 2010 to 2012, and a Distinguished Lecturer of the IEEE Circuits and Systems Society from 2006 to 2007 and 2012 to 2013. He currently serves as an ACM Distinguished Speaker. He is a fellow of ACM and a Golden Core Member of the IEEE Computer Society. He was a 2009 Invitational Fellow of the Japan Society for the Promotion of Science.

Kristin Scott (M'98) received the B.S. degree in molecular biology from Grove City College, Grove City, PA, USA, in 1992, and the Ph.D. degree in biochemistry from the University of Iowa, Iowa City, IA, USA, in 1998.

She has been with the Department of Molecular Genetics and Microbiology, Duke University, Durham, NC, USA, since 2014, where she is currently an Assistant Professor. From 2008 to 2013, she was an Assistant Professor with the Duke Institute for Genome Sciences and Policy, Durham. Her current research interests include effects of higher order chromatin packaging on gene expression and genome stability.