Microelectronics Journal 79 (2018) 38–45

Contents lists available at ScienceDirect

Microelectronics Journal

journal homepage: www.elsevier.com/locate/mejo

Efficient spanning-tree-based test pattern generation for Programmable Microfluidic Devices

Alessandro Bernardini *, Chunfeng Liu, Bing Li, Ulf Schlichtmann

Chair of Electronic Design Automation, Technical University of Munich, Germany

ARTICLE INFO

Keywords:
Microfluidic biochips
Programmable Microfluidic Devices (PMDs)
Manufacturing defects
Test

ABSTRACT

Microfluidic biochips are revolutionizing the traditional biochemical experiment flow with their high execution efficiency and miniaturized fluid manipulation. To improve the flexibility of such chips, Programmable Microfluidic Devices (PMDs) have been introduced, with valves and channels built as a regular array. To take advantage of this new architecture, design automation methods have started to appear, but an efficient test generation algorithm is still unavailable. In this paper, we propose to generate test patterns based on spanning trees for this architecture. Compared with previous work, the proposed method is around 100 times faster, while the number of test patterns is still comparable.

1. Introduction

Microfluidic biochips are revolutionizing the traditional biochemical experiment flow with their high execution efficiency and miniaturized fluid manipulation [1]. On such a chip, biochemical assays are executed at the nanoliter level and the execution is controlled by a microcontroller without human intervention. The efficiency and reliability of such miniaturized and automated chips endow them with a great potential in biology laboratories and health-care centers.

A flow-based microfluidic biochip is constructed from basic components such as micro channels and valves. Flow channels are used to transport reaction samples and reagents between different locations. Control channels are built above flow channels to conduct air pressure to valves in order to control the flow of fluid, as illustrated in Fig. 1(a), where three valves are constructed at the shown intersections. Since the channel width has been miniaturized down to 50 μm [2] thanks to the advance of manufacturing technology, a huge number of channels and valves can already be integrated into a single biochip to perform large-scale experiments and diagnoses.

With valves as basic controlling components, complex devices can be constructed. For example, mixers can be built using channels and valves to execute mixing operations, which are very common in biochemical applications. The structure of a mixer is shown in Fig. 1(b), where the three valves at the bottom can drive a circular flow by peristalsis to mix fluid samples and reagents. In a biochip, such devices are connected by channels to form a transportation network.

In a biochip, the transportation of fluid samples and the execution of mixing operations are realized by valves built between channels. In Fig. 1(b), the alternate switching activities of the three valves drive a flow inside the loop-shaped channel for mixing. This channel, however, needs not to be round to implement the mixing function. In general, three valves are sufficient to form a small mixer. Bigger mixers can be formed using more valves. This observation has been demonstrated in Ref. [3], where a regular valve array is built to form Programmable Microfluidic Devices (PMDs) [1,3].

A part of the large valve array in Ref. [3] is shown in Fig. 2(a) to demonstrate the architecture of PMDs. In this architecture, valves (solid blocks) are arranged in a regular structure along horizontal and vertical flow channels (light color). These valves are controlled by air pressure sources through control channels (narrow channels). Transportation paths can be formed by opening and closing specific valves on the array, respectively.

Besides transportation channels, mixers can also be constructed on the valve array directly, taking advantage of the flexibility and reconfigurability of this architecture. For example, a 4 × 2 mixer and a 2 × 4 mixer can be constructed as in Fig. 2(b) and (c), respectively. In such a dynamic mixer, the eight valves along the enclosed channel

* Corresponding author.
E-mail addresses: alessandro.bernardini@tum.de (A. Bernardini), chunfeng.liu@tum.de (C. Liu), b.li@tum.de (B. Li), ulf.schlichtmann@tum.de (U. Schlichtmann).

https://doi.org/10.1016/j.mejo.2018.06.011
Received 14 February 2018; Received in revised form 23 May 2018; Accepted 18 June 2018
Available online XXX

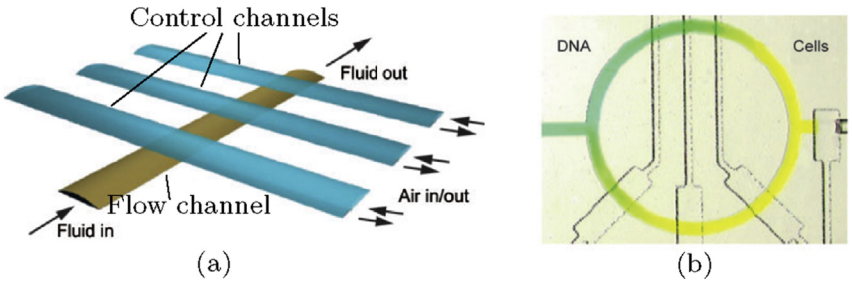

Fig. 1. Components and structure of flow-based biochips. (a) Valves constructed at intersections of flow/control channels [1]. (b) Mixer built with valves as the peristalsis drivers [1].

function as peristalsis valves. Compared with the traditional mixer in Fig. 1(b), these dynamic mixers have a different shape and more peristalsis valves, eight in each case, to form a strong circular mixing flow.

Moreover, the two mixers in Fig. 2(b) and (c) can share the same area on the biochip as shown in Fig. 2(d), provided that they are not used at the same time.

On the valve array, a given area can execute various functions such as mixing and flow transportation, as well as detection if the corresponding sensors are included in the area. With this flexibility, the design flow of biochips can be simplified significantly, since the dependency between operations of a bioassay and dedicated devices is not required anymore. In addition, this flexible architecture enables fault-tolerance easily. If any valves on the array do not work properly, the operations can be moved to other areas to maintain the function of the biochip.

In recent years, design and optimization methods for such biochips have started to appear quickly. For high-level synthesis, the top-down flow in Ref. [4] generates a biochip architecture and minimizes the execution time of the bioassay, while the method in Ref. [5] minimizes valve switching activities during architectural synthesis. The concept of general modeling of devices is introduced in Ref. [6] to improve the efficiency of the synthesis process, and special devices such as sieve valves are considered in Ref. [7]. On system level, the concept of distributed channel storage in flow-based biochips is also explored in Refs. [8,9].

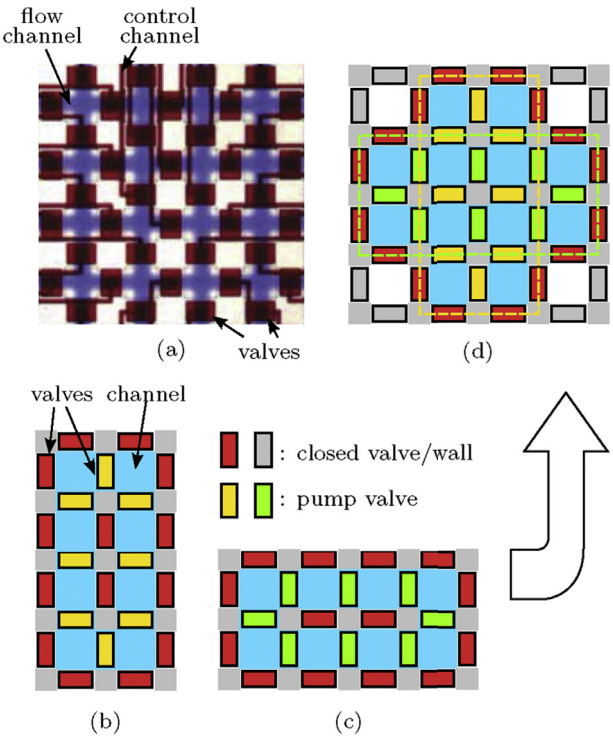

Fig. 2. Dynamical operation execution on PMD. (a) Architecture [3]. (b)/(c) A 4 × 2/2 × 4 dynamic mixer. (d) Dynamic mixers of different orientations sharing the same area.

The placement of devices and routing of channels in flow-based biochips are dealt with simultaneously in Ref. [10] using a sequence-pair representation, and they are formulated as a SAT problem in Ref. [11] to achieve a close-to-optimal result. Physical design considering obstacles is investigated in Ref. [12] and solved using a rectilinear Steiner minimum tree.

Control logic synthesis is investigated in Ref. [13] to reduce the number of control pins. The method in Ref. [14] minimizes pressure propagation delay in the control layer to reduce the response time of valves and synchronize their actuations. Switching patterns of valves are examined in Refs. [15,16] to reduce the largest number of switching activities in the control logic to avoid potential reliability problems. Furthermore, codesign of flow layer and control layer is investigated in Ref. [17] to achieve valid routing results on both layers iteratively, and length-matching is incorporated in routing control channels in Ref. [18] as well. Moreover, flow-layer, control-layer and valve switching are considered together in Refs. [19,20] to simplify overall design complexity.

When flow-based biochips are manufactured, defects may appear at valves and channels, leading to chips not functioning correctly. To deal with such manufacturing defects, fault models and an ATPG-based test strategy for flow-based biochips are proposed in Refs. [21,22]. Design-for-testability and defect diagnosis are further addressed in Refs. [23,24].

Since PMDs are very promising for experiments in future biochemical labs and health-care centers due to their flexibility and fault-tolerance, their specific features have also been explored. The method in Refs. [25,26] takes advantage of the flexibility of dynamic mapping to reduce the switching activity of valves. In addition, the method in Ref. [27] avoids channel crossing efficiently during construction of dynamic flow paths. Furthermore, valve control sequences on such an architecture are investigated in Ref. [28]. For test of manufacturing defects, however, only one method has been proposed in Ref. [29]. This method, unfortunately, relies on an ILP (Integer Linear Programming) formulation. Consequently, it requires much time to generate proper test patterns and for large valve arrays even cannot return valid solutions.

In this paper, we propose an efficient test generation method based on spanning trees. The proposed method is several orders of magnitude faster than the method in Ref. [29], while producing comparable results. In Section 2, we explain two major types of manufacturing faults and formulate the test problem. In Section 3, we describe the proposed test pattern generation using spanning trees in detail. In Section 4, we demonstrate the performance of the proposed method and compare it with [29]. Finally, conclusions are drawn in Section 5.