

Droplet-Trace-Based Array Partitioning and a Pin Assignment Algorithm for the Automated Design of Digital Microfluidic Biochips*

Tao Xu and Krishnendu Chakrabarty
Department of Electrical and Computer Engineering
Duke University, Durham, NC 27708, USA
{tx, krish}@ee.duke.edu

Abstract

Microfluidics-based biochips combine electronics with biology to open new application areas such as point-of-care medical diagnostics, on-chip DNA analysis, and automated drug discovery. Bioassays are mapped to microfluidic arrays using synthesis tools, and they are executed through the manipulation of sample and reagent droplets by electrical means. Most prior work on CAD for biochips has assumed independent control of electrodes using a large number of (electrical) input pins. Such solutions are not feasible for low-cost disposable biochips that are envisaged for many field applications. A more promising design strategy is to divide the microfluidic array into smaller partitions and use a small number of electrodes to control the electrodes in each partition. We propose a partitioning algorithm based on the concept of “droplet trace”, which is extracted from the scheduling and droplet routing results produced by a synthesis tool. An efficient pin assignment method, referred to as the “Connect-5 algorithm”, is combined with the array partitioning technique based on droplet traces. The array partitioning and pin assignment methods are evaluated using a set of multiplexed bioassays.

Categories & Subject Descriptors

B.7.2 [Integrated Circuits]: Design Aids; J.3 [Computer Applications]: Life and Medical Sciences—*Biology and genetics; health*

General Terms:

Algorithms, Design, Performance

Keywords:

Connect-5 algorithm, droplet-based microfluidics, droplet trace, pin-constrained biochip design, synthesis

1. Introduction

Microfluidics-based biochips constitute an emerging technology area that can potentially open up several exciting applications. These devices enable precise control of microliter and nanoliter volumes of biological samples. They combine electronics with biology, and they integrate various bioassay operations, such as sample preparation, analysis, separation, and detection [1, 2, 3], in a single miniaturized platform. It has been predicted that, by providing miniaturization, automation and integration, microfluidic biochips will revolutionize

laboratory procedures in molecular biology with applications to point-of-care diagnostics, DNA analysis, and automated drug discovery [2, 3].

Currently, most commercially-available biochips are either based on microarrays [4] or they rely on continuous fluidic flow in etched microchannels [1]. An alternative design approach utilizes droplets with microliter and nanoliter volumes, thereby obviating the need for cumbersome micropumps and microvalves. Droplets are actuated using on-chip electrodes and moved under the control of a system clock; this microfluidic system is similar in operation to a digital microprocessor. Thus, this novel technology is referred to as “digital microfluidics”. The “digital” structure also offers reconfigurability and a scalable system architecture based on a two-dimensional array [5, 6].

A typical biochip consists of a two-dimensional patterned metal electrode array (e.g., chrome or indium tin oxide), on which droplets containing biological samples are dispensed, transported, mixed, incubated, separated or detected. As bioassays increase in complexity, e.g., for high-throughput DNA sequencing [7] and large-scale protein assays for drug discovery [8], design tools are needed to map and execute them on the digital microfluidic platform. In the next few years, biochip integration and design complexity level are expected to increase significantly. Automated design therefore becomes necessary for this emerging marketplace. An appropriate addressing scheme must be used to activate individual electrodes (unit cells) in the array.

Design and CAD research for digital microfluidic biochips has mostly been focused on directly-addressable arrays [9, 10, 11, 12, 13, 14]. In such schemes, each cell of the patterned electrodes can be accessed directly and independently via a dedicated control pin. This method is adequate for small/medium-scale microfluidic electrode arrays (with fewer than 10×10 electrodes). However, the number of pins for a design based on direct addressing can be prohibitively high for a large array. For example, a total of 10^4 pins are needed to independently control the electrodes in a 100×100 array. Multi-layer electrical connection structures and wire routing solutions are complicated by the large number of independent control pins in such arrays. Product cost, however, is a major marketability driver due to the one-time-use (disposable) nature of most emerging devices. Hence, simpler routing solutions are necessary so that the electrical wiring can be easily incorporated in a low-cost implementation.

In this paper, we propose an automated digital microfluidic biochip design method based on the partitioning of the microfluidic array, and the assignment of a small number of control pins to a large number of electrodes. The partitioning algorithm is based on the concept of “droplet trace”, which is extracted from the scheduling and droplet routing results produced by a synthesis tool.

The organization of the rest of the paper is as follows. In Section 2, we provide an overview of digital microfluidic biochips. Section 3 discusses related prior work on biochip design automation and pin-constrained system design. Section 4 describes the proposed

*This work was supported in part by the National Science Foundation under grants IIS-0312352 and CCF-0541055.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CODES+ISSS'06, October 22–25, 2006, Seoul, Korea.
Copyright 2006 ACM 1-59593-370-0/06/0010...\$5.00.

partitioning and pin assignment algorithm for a large microfluidic array. Section 5 evaluates the proposed method using a set of real-life bioassays. Finally, conclusions are drawn in Section 6.

2. Digital Microfluidic Biochips

A digital microfluidic biochip utilizes the phenomenon of electrowetting to manipulate and move microliter or nanoliter droplets containing biological samples on a two-dimensional electrode array [5]. A unit cell in the array includes a pair of electrodes that acts as two parallel plates. The bottom plate contains a patterned array of individually controlled electrodes, and the top plate is coated with a continuous ground electrode. A droplet rests on a hydrophobic surface over an electrode, as shown in Figure 1. It is moved by applying a control voltage to an electrode adjacent to the droplet and, at the same time, deactivating the electrode just under the droplet. This electronic method of wettability control creates interfacial tension gradients that move the droplets to the charged electrode. Using the electrowetting phenomenon, droplets can be moved to any location on a two-dimensional array.

By varying the patterns of control voltage activation, many fluid-handling operations such as droplet merging, splitting, mixing, and dispensing can be executed in a similar manner. For example, mixing can be performed by routing two droplets to the same location and then turning them about some pivot points. The digital microfluidic platform offers the additional advantage of flexibility, referred to as reconfigurability, since fluidic operations can be performed anywhere on the array. Droplet routes and operation scheduling result are programmed into a microcontroller that drives electrodes in the array. In addition to electrodes, optical detectors such as LEDs and photodiodes are also integrated in digital microfluidic arrays to monitor colorimetric bioassays [3].

To address the need for low-cost, PCB technology has been employed recently to inexpensively mass-fabricate digital microfluidic biochips. Using a copper layer for the electrodes, solder mask as the insulator, and a Teflon AF coating for hydrophobicity, the microfluidic array platform can be fabricated by using an existing PCB manufacturing process [15]. This inexpensive manufacture technique allow us to build disposable PCB-based microfluidic biochips that can be easily plugged into a controller circuit board that can be programmed and powered via a standard USB port; see Figure 2. However, a large number of independent control pins necessitates multiple PCB layers, which adds significantly to the product cost.

3. Related Prior Work

Recently years have seen growing interest in the design of microfluidic biochips and CAD methods for system design [10, 12, 14, 16, 17]. In [10], classical architectural-level synthesis is adapted for automated biochip design based on bioassay protocols. The problem of microfluidic module placement, where array area and fault tolerance serve as the placement criteria, is discussed in [12]. A unified synthesis method, which combines operation scheduling, resource binding, and module placement, is proposed in [16]. A drawback of these CAD techniques is that they assume a direct-addressing scheme, which requires a very large number of independent control pins for large-scale biochips. Thus these methods are unlikely to be useful in practice for low-cost disposable devices.

Pin-constrained design of digital microfluidic biochips was recently proposed and analyzed in [3]. The number of control pins for a fabricated electrowetting-based biochip is minimized by using a multi-phase bus for the fluidic pathways. Every n th electrode in an n -phase bus is electrically connected. Thus, only n control pins are needed for a transport bus, irrespective of the number of electrodes

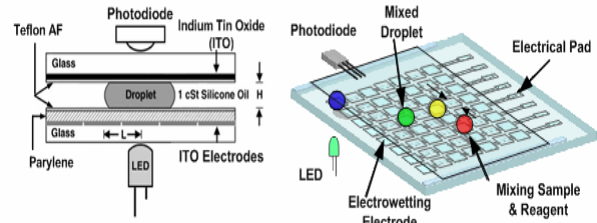


Figure 1: Schematic diagram of a digital microfluidic biochip.

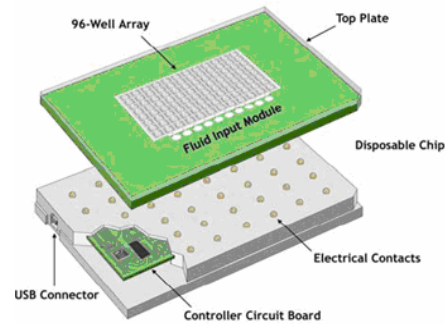


Figure 2: The concept of a commercial disposable microfluidic biochip.

that it contains. Although the multi-phase bus method is useful for reducing the number of control pins, it is only applicable to a one-dimensional (linear) array.

An alternative method based on a cross-reference driving scheme is presented in [18]. This method allows control of an $N \times M$ grid array with only $N+M$ control pins. The electrode rows are patterned on both the top and bottom plates, and placed orthogonally. In order to drive a droplet along the X-direction, electrode rows on the bottom plate serve as driving electrodes, while electrode rows on the top serve as reference ground electrodes. The roles are reversed for movement along the Y-direction. This cross-reference method facilitates the reduction of control pins. However, it requires a special electrode structure (i.e., both top and bottom plates containing electrode rows), which results in increased manufacturing cost for disposable microfluidic chips. Moreover, this design is not suitable for high-throughput assays because droplet movement is inherently slow.

More recently, a promising design method based on array partitioning has been proposed for pin-constrained biochips [17]. The microfluidic array is divided into several partitions and sets of pins are determined, where each set of pins correspond to a partition and all the sets are of the same size. For example, if a biochip of arbitrary size is divided into six partitions and five pins are allocated per set, only $5 \times 6 = 30$ pins are needed to independently address the individual unit cells of the array. By carefully controlling the number of partitions, the total number of pins is reduced significantly compared to the direct-addressing scheme.

However, the design method presented in [17] suffers from several drawbacks. First, the array partitioning in [17] is ad-hoc and no systematic algorithm has thus far been presented. Secondly, microfluidic modules such as mixers, splitters, and detectors are not considered in the ad-hoc partitioning method; an additional design step is needed to handle these modules separately. Moreover, the partitioning method assumes *a priori* that partitions do not overlap; this restriction can be a limitation for many bioassays. Finally, no pin-assignment algorithm is presented in [17].

4. Array Partitioning and Pin Assignment

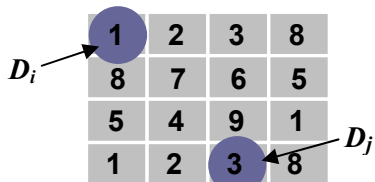


Figure 3: An example to illustrate electrode interference.

We first review the problem of electrode interference in microfluidic arrays. This problem can appear if multiple electrodes are controlled using a single pin. For example, assume that a droplet rests on an electrode (unit cell) and two of its neighbors are connected to the same pin. Recall that to move the droplet to one of two neighbors (i and j) that share the same pin, we must deactivate the electrode where the droplet rests and activate the destination electrode i . However, when electrode i is activated, the other neighbor electrode j is also activated since it shares the same pin with electrode i . In this case, the droplet undergoes a split, instead of being moved to electrode i . This problem can be solved by addressing each electrode and its neighbors with distinct pins. Since one electrode can have at most four neighbors in a two-dimensional array, the minimum number needed is five. Recent experimental studies have shown that five independent pins are adequate to route a droplet to any place on the chip for single droplet manipulation [17].

When multiple droplets are manipulated simultaneously on the chip, a pin-constrained layout may also result in unintentional droplet movement or other unintended consequences. For the example in Figure 3, electrode interference will occur if we attempt to move Droplet D_i and let Droplet D_j stay where it is. To move D_i one cell downwards, we need to activate Pin 8 and deactivate Pin 1. To hold Droplet D_j , we need to activate Pin 3. However, since both Pin 3 and Pin 8 are charged, D_j will be split unintentionally. This type of problem is referred to as electrode interference.

Electrode interference can be solved by “virtually” partitioning the array into regions, with each of them having only one activated cell at any point in time. Mutually-exclusive sets of pins are utilized for manipulating the droplets in different regions. The partitions can be viewed as subarrays that can contain at most one droplet. Recall that regardless of size, a two-dimensional array only needs five independent pins to ensure full control of a single droplet. By using different sets of five pins for electrode control in different partitions, electrode interference among partitions can be avoided. Therefore, for the partitioned array, the number of droplets that can be simultaneously transported without stall cycles is equal to the number of partitions, and the total number of control pins needed is equal to five times the number of partitions. The above partitioning solution was proposed recently in [17].

However, both array partitioning and the assignment of control pins to electrodes in [17] are done in an ad-hoc manner. No systematic algorithms have been proposed thus far to implement the partitioning-based pin-assignment method and incorporate it in automated design tools. Here we propose an algorithm based on the concept of droplet trace, which unifies array partitioning and pin assignment.

4.1. Trace-Based Partitioning Algorithm

As discussed above, partitioning can effectively avoid electrode interference if each partition includes only one droplet. Hence, the partitioning criterion here is to ensure at most one

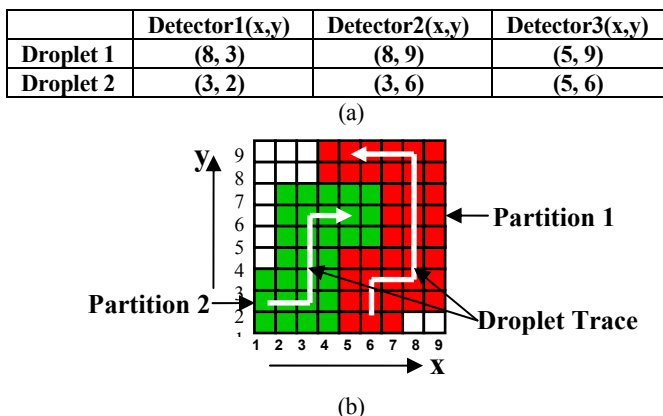


Figure 4: (a) Detectors used in bioassay; (b) Routing result and array partitions.

droplet is included in each partition. However, partitions with no droplets (at any point in time) should be avoided because no droplet manipulation is done in this region with the additional set of pins assigned to it. Hence it is best to ensure that each partition has exactly one droplet in it.

Based on this requirement, we find that the droplet trace, defined as the set of cells traversed by a single droplet, serves as a good tool for generating the array partitions. Since we view pin assignment as the last step in system synthesis, information about module placement and droplet routing is available *a priori*. The droplet trace can be easily extracted from the droplet routing information and the placement of the modules it is routed to. A trace extraction example is shown in Figure 4, where two droplets are to be manipulated on the microfluidic array. Both of these are required to be detected by an optical sensor three times in a specific bioassay. The placement of these detectors is shown in Figure 4(a). The droplet routes, i.e., the path taken by droplets, are shown by the arrows in Figure 4(b). The connected arrows illustrate the traces of the two droplets. For each droplet, we create a partition composed of all the cells on its trace as well as the cells adjacent to the trace. The adjacent cells are included to form a “guard ring” along the trace to avoid inadvertent mixing and movement. The guard rings are a consequence of the fluidic constraint described in [19].

Note that in Figure 4(b), there are two “white” regions that belong to neither partition. They are referred to as “don’t-care” regions because they are similar to the “don’t-care” terms in logic synthesis; they can either be assigned to any partition or they can together form an additional partition if multi-droplet-operation modules, e.g. mixers, can be positioned in them.

In order to reduce the number of partitions, we introduce a time-division pin-sharing method. The basic idea is to merge partitions that have no overlapping time spans, where a *time span* for a partition is defined as the period of time during which it contains a droplet. The time spans for all the partitions can be easily calculated from the operation schedule, module placement and droplet routing results [19]; the overlaps can then be readily determined. Partitions with non-overlapping time spans are merged to form a larger partition. This check-merge procedure continues until all partition pairs overlap in their time spans. By reducing the number of partitions, we can reduce the number of control pins needed for the array. Note that droplet traces may have spatial overlap, i.e., they may intersect at one or more unit cells on the array. In this case, the requirement of one droplet per partition is not met and electrode interference may occur. This problem is handled by simply modifying the partitioning result.

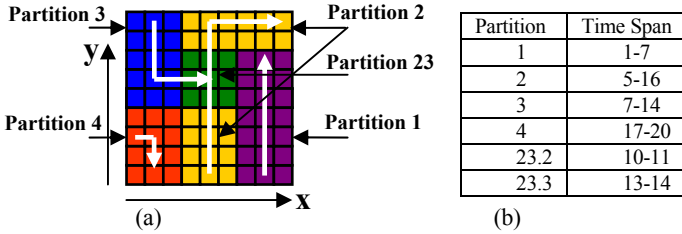


Figure 5: (a) Routing result and partitioning (b) Time-span table for the droplets.

We next study the case where droplets traces intersect on the array. This implies that partitions derived by the proposed method overlap in some regions. Sets of pins from an “overlapping” partition cannot be used in the overlapped region since the reuse of the pins leads to electrode interference. One solution to this problem is to make the overlapping region a new partition, referred to as the overlapping partition, and use direct-addressing for it. Again, time-division pin-sharing (TDPS) can be used to reduce the number of pins since pin sets of the other (non-overlapping) partitions can be candidates for direct-addressing in the overlapping partition.

An example of this approach is shown in Figure 5. The droplet traces are first derived from the droplet routing information. Partitions 1, 2, 3, and 4 are assigned accordingly. Partition 2 and Partition 3 overlap with each other as shown. Thus a new Partition 23 is created. From the scheduling result in Figure 5(b), the time span for Partition 23 is found to be 10-14s. Next the time spans for Partitions 1 and 4 are checked and it is seen that their time spans do not overlap with that for Partition 23. Hence the two set of pins (a total of $2 \times 5 = 10$ pins) in Partitions 1 and 4 can be used to directly address the nine electrodes in Partition 23.

Partitions that share pins with the overlapping partition are empty while droplets are manipulated in the overlapping partition. Therefore, the sharing of pins in these cases does not lead to electrode interference. By introducing the concept of TDPS, we can significantly reduce the number of pins required for independent addressing. The concept of TDPS can also be applied in the spatial dimension to the operations inside the overlapping region to further reduce the number of control pins.

Once a spatially overlapping region is found, we determine if there are temporally overlapping droplets in this region. Depending on the outcome of this procedure, a spatial overlap region can be then divided into two groups—a spatially overlapping but temporally non-overlapping (SOTN) region, and a spatially overlapping as well as temporal overlapping (SOTO) region. For SOTO regions, direct-addressing is used. For SOTN regions, even though droplets traces cross each other, different droplets are sequenced in time (one after the other), i.e., at any point in time, there is at most one droplet inside the region. In this case, a pin set with the minimum size ($k = 5$) for single droplet manipulation is assigned to this SOTN region.

Again, we use the above example of Figure 5 for illustration. Table 1 shows the schedule information needed for carrying out the temporal check for the overlapping region.

Partitions 23.2 and 23.3 represent the manipulation of Droplet 2 and Droplet 3 in Partition 23 respectively. Table 1 shows that the time spans for these partitions do not overlap, thus five pins (in contrast to the nine pins needed for direct-addressing) are adequate for the overlapping partition.

4.2. Extended Partitioning Algorithm

In this subsection, we present an extension of the partitioning algorithm that does not require module placement information. In Section 4.1, we needed the placement information for modules that

Table 1: Time-span table with detailed scheduling results for the overlapping region.

Partition	Time Span
1	1-7
2	5-12
3	7-23
4	17-20
23	10-14

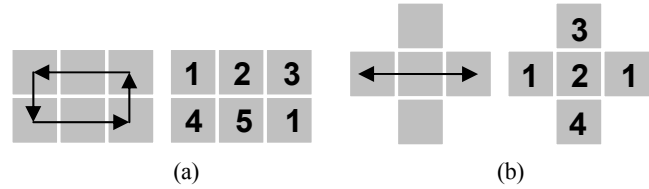


Figure 6: Pin assignment example for (a) a mixer and (b) a splitter.

handle multiple droplets, such as mixers and splitters to determine the droplet traces. Here we rely only on the schedule of operation and droplet routing results to indirectly determine module placement. For example, the mixing operation can be viewed as two droplets being routed together along an identical path simultaneously with the start point in the mixer region. Similarly, droplet splitting can be viewed as two droplets sharing the same start point in both the time and space domains. We can therefore identify mixer regions by checking whether droplet traces exactly overlap instead of just intersecting each other in the same time span; a splitter can be recognized in a similar manner. As a result, overlapping partitions can be assigned to mixers and splitters. Note that splitting and mixing can both be viewed as deliberate (desired) electrode interference. Thus though multiple droplets are manipulated in mixer or splitter regions, five control pins are sufficient, as shown in Figure 6. In this way, the number of pins can be further reduced.

4.3. Pin assignment using the Connect-5 algorithm

In Section 4.1 and Section 4.2, we have described an automated partitioning method for digital microfluidic arrays. Each partition is assigned a pin set. In this section, we address the problem of how to map control pins to the electrodes in a partition. An efficient and easy-to-implement algorithm is presented. The algorithm is based on a strategy of the Connect-5 (Gomoku) board game [20], thus it is referred to as the Connect-5 algorithm.

The sets of pins assigned to the partitions belong to two groups according to their cardinality, i.e., the minimum for single droplet manipulation ($k = 5$) or the number of pins required for direct-addressing. Here we focus on the pin assignment problem for the first case, since pin assignment for direct-addressing is straightforward (there exists a simple one-to-one mapping between pins and electrodes).

Our goal is to ensure that any five adjacent unit cells (a central cell and its four neighbors) that form a “cross” are assigned distinct pins. We refer to the above constraint as the “cross constraint”. The pin assignment problem under cross constraints can be mapped to the famous vertex coloring problem in graph theory [21]. The problem here is to obtain a 5-coloring of the graph derived from a partition, as shown in Figure 7. The unit cells in the partition are mapped to vertices and any two cells that belong to a “cross” are connected by an edge. The graph corresponding to a partition is referred to as the partition graph.

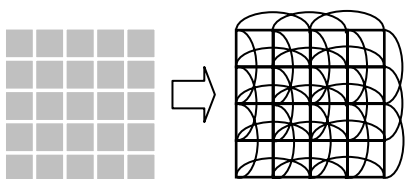


Figure 7: Mapping of an array to an undirected graph.

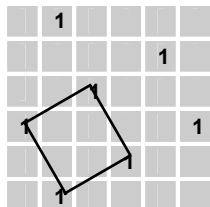


Figure 8: A single Bagua structure (the tilted square) and its repetition in a square partition.

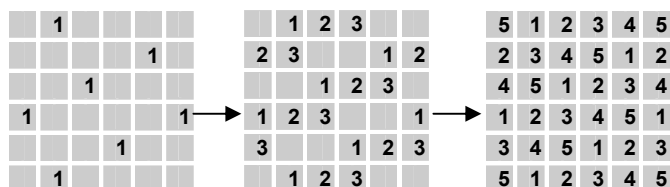


Figure 9: Covering a partition by shifting Bagua repetition along rows.

The graph coloring problem, which involves the determination of the chromatic number $\chi(G)$ for a graph G , is known to be NP-complete [21]. However, if $\chi(G)$ or the number of colors to be used is known, as in the case here, there exists efficient algorithm for graph coloring. However, the regular structure of the partitions can be used to solve the problem more efficiently using tiling. This approach allows us to use a regular distribution of pins, a layout feature that is not directly obtained from graph coloring. The tile (or template) used here is referred to as “Bagua”, a Chinese game strategy for the Connect-5 board game [20]. A Bagua is a tilted square, as shown in Figure 8. By repeating placing Bagua structures next to each other until the partition boundaries are reached, a Bagua repetition is derived as shown in Figure 8. The tiling using Bagua repetitions forms the basis for the Connect-5 algorithm.

Five copies of Bagua repetitions are sufficient to cover a partition of any size. This is because of the following property of a Bagua repetition: vertices connected to the same (shared) pin appear after exactly five cells in the same row or column of the partition. The partition can be covered with Bagua repetitions by simply taking a Bagua repetition and shifting it one cell along an arbitrary direction, e.g., upwards, then assigning it to another control pin and repeating this step four times, as shown in Figure 9. Note that, although the shifting direction is arbitrarily selected at the start of the tiling process, once chosen it must be consistent over the four shifting steps.

As shown in Figure 9, the pin assignment that results from the shifting of Bagua repetition satisfies a cyclic property, i.e., each row is a cyclic repetition of an ordered sequence, and it is also a shifted copy (shift by two cells) of the previous row. This cyclic property provides an easy way to implement the Connect-5 algorithm.

To start, the first row of a partition is selected. Pins are assigned in a fixed cyclic order until the boundary of the partition is reached. Then in the next row, the same order is used for but with a 2-cell-shift to the left/right. The procedure continues until all cells in the partition have been assigned pins. Recall that the

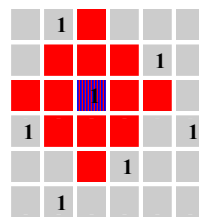


Figure 10: A demonstration that the “cross constraints” are met.

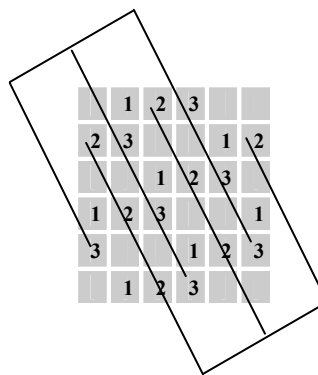


Figure 11: A wiring example for the pin assignment obtained using the Connect-5 algorithm. For each partition, two pins can be wired in one layer.

shifting direction, once chosen, must remain fixed during the assignment procedure for a given partition.

Next we show that control pins assigned to the electrodes this method in a partition allow free movement of a single droplet, i.e., the “cross constraint” is met. To demonstrate this, we consider the cell which is hatched in Figure 10. If the cell is assigned Pin 1, we cannot assign the same pin to the unit cells that are shaded. Otherwise, we will violate the cross constraint in some cases. It can be found that all the unit cells in the Bagua tile and its repetitions stay out of the forbidden area. Thus for each pin assigned to cells in a Bagua repetition, the cross constraint is not violated. Since this is true for any Bagua repetitions and any partition can be tiled by five copies of Bagua repetitions, the “cross constraint” is automatically met for every cell in our pin assignment method.

Compared to the graph coloring approach, the Connect-5 algorithm offers the important advantage that it allows wiring to be done easily on a 3-layer PCB; see Figure 11. The graph coloring approach does not lend itself to this simple pin layout because of the likelihood of irregular vertex coloring.

5. Evaluation Example: Multiplexed Bioassays

To show how partitioning and pin assignment work for pin-constrained microfluidic biochips, we use a real-life experiment of a multiplexed biochemical assay consisting of a glucose assay and a lactate assay based on colorimetric enzymatic reactions. These assays have been demonstrated recently [3]. The digital microfluidic biochip contains a 15×15 microfluidic array, as shown in Figure 11. The schedule for the set of bioassays, if a microfluidic array with 225 control pins is available, is listed in Table 2; one iteration of the multiplexed assays takes 25.8 seconds [3]. The movement of droplets is controlled using a 50 V actuation voltage with a switching frequency of 16 Hz. A depiction of the droplet paths for multiplexed glucose and lactase assays is shown in Figure 12.

When the partitioning and pin assignment algorithm starts, six partitions are first assigned to the four droplet traces of Reactants 1,

Table 2: Bioassay schedule for a multiplexed bioassay.

Step/Time Elapsed (s)	Operation
Step 1 / 0	Sample 2 and Reagent 2 start to move towards the mixer.
Step 2 / 0.8	Sample 2 and Reagent 2 begin to mix together and turn around in the 2×3-array mixer.
Step 3 / 6.0	Sample 1 and Reagent 1 start to move towards the mixer. Sample 2 and Reagent 2 continue the mixing.
Step 4 / 6.8	Sample 2 and Reagent 2 finish the mixing and product 2 leaves the mixer to optical detection location 2. Sample 1 and Reagent 1 begin to mix in the 2×3-array mixer.
Step 5 / 12.8	Sample 1 and Reagent 1 finish the mixing and product 1 leaves the mixer to the optical detection location 1. Product 2 continues the absorbance detection.
Step 6 / 19.8	Product 2 finishes optical detection and leaves the array to the waste reservoir. Product 1 continues the absorbance detection.
Step 7 / 25.8	Product 1 finishes optical detection and leaves the array to the waste reservoir. One procedure of the multiplexed bioassays ends.

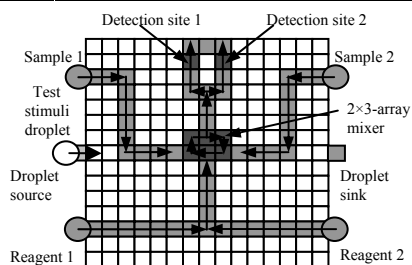


Figure 12: A 15×15 array used for multiplexed bioassays.

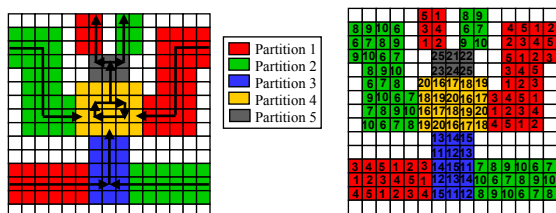


Figure 13: Partition and pin assignment results for the multiplexed bioassay. Blank areas are don't-care regions that can be either left unaddressed or combined with any partition.

2 and Samples 1, 2, and the two traces of the mixed samples going to Detector 1 and Detector 2. Another three partitions are assigned to the three trace-overlapping regions respectively. Next, time-span overlap is checked for the three spatial overlapping partitions (Partitions 3, 4 and 5). Since there is no temporal overlap of droplets being manipulated in both Partition 3 and Partition 5, only five pins are needed for each of them. Partition 4 is recognized as a mixer, thus only five pins are needed for it. In the next step, time span overlap is checked for all partitions pairs. The six partitions corresponding to four droplets traces and two detector paths merge into two partitions (Partition 1 and Partition 2). Finally, the Connect-5 algorithm is applied. The partitions and pin assignment results are shown in Figure 13.

We therefore see that array partitioning and pin assignment is effective in reducing the input bandwidth, while maintaining the same throughput that is obtained for a direct-addressable array. Five partitions are satisfactory for preventing interference between multiple droplets on the array, as shown in Figure 13. Since only five control pins are necessary for full control of a single droplet

within each partition, only 25 out of the possible 225 control pins are necessary, i.e., only 11.11% of the total number of electrodes. This represents a significant reduction in input bandwidth without sacrificing throughput.

6. Conclusions

We have presented an efficient algorithm for array partitioning and pin assignment in pin-constrained digital microfluidic biochips. The proposed partitioning algorithm is based on the concept of droplet trace, which is extracted from the scheduling and droplet routing results produced by a synthesis tool. The array partitioning and pin assignment methods have been evaluated using a set of multiplexed bioassays. By drastically reducing the number of control pins with minimal impact on assay throughput, the proposed design technique is expected to reduce cost and lead to further miniaturization of disposable biomedical devices for the emerging healthcare market.

References

- [1] E. Verpoorte and N. F. De Rooij, "Microfluidics meets MEMS", *Proc. IEEE*, vol. 91, pp. 930-953, 2003.
- [2] T. H. Schulte et al., "Microfluidic technologies in clinical diagnostics", *Clinica Chimica Acta*, vol. 321, pp. 1-10, 2002.
- [3] V. Srinivasan et al., "An integrated digital microfluidic lab-on-a-chip for clinical diagnostics on human physiological fluids" *Lab on a Chip*, pp. 310-315, 2004.
- [4] M. Schena, *Microarray Biochip Technology*, Eaton Publishing, Natick, MA, 2000.
- [5] M. G. Pollack et al., "Electrowetting-based actuation of liquid droplets for microfluidic applications", *Applied Physics Letters*, vol. 77, pp. 1725-1726, 2000.
- [6] S. K. Cho et al., "Toward digital microfluidic circuits: creating, transporting, cutting and merging liquid droplets by electrowetting-based actuation", *Proc. IEEE MEMS Conf.*, pp. 32-52, 2002.
- [7] B. M. Paegel, et al., "Microfluidic devices for DNA sequencing: sample preparation and electrophoretic analysis", *Current Opinion Biotechnol.*, vol.14 42-50, 2003.
- [8] V. Srinivasan et al., "Protein stamping for MALDI mass spectrometry using an electrowetting-based microfluidic platform", *Proc. SPIE*, vol. 5591, pp. 26-32, 2004.
- [9] K. Chakrabarty and J. Zeng, "Design automation for microfluidics-based biochips", *ACM J. Emerging Tech. Comput. Syst.* vol. 1, pp. 186-223, 2005.
- [10] F. Su and K. Chakrabarty, "Architectural-level synthesis of digital microfluidics-based biochips", *Proc. ICCAD*, pp. 223-228, 2004.
- [11] F. Su and K. Chakrabarty, "Design of fault-tolerant and dynamically-reconfigurable microfluidic biochips", *Proc. DATE Conf.* pp. 1202-1207, 2005.
- [12] P-H. Yuh et al., "Placement of digital microfluidic biochips using the T-tree formulation", *Proc DAC*, pp. 931-934, 2006.
- [13] F. Su, K. Chakrabarty and R. B. Fair, "Microfluidics-based biochips: technology issues, implementation platforms, and design automation challenges", *IEEE Trans. CAD*, vol. 25, pp. 211-223, February 2006.
- [14] K. Chakrabarty and F. Su, "System-level design automation tools for digital microfluidic biochips", *Proc. CODES+ISSS*, pp. 201-206, 2005.
- [15] J. Gong and C. J. Kim, "Two-dimensional digital microfluidic system by multi-layer printed circuit board," *Proc. IEEE MEMS 2005*
- [16] F. Su and K. Chakrabarty, "Unified high-level synthesis and module placement for defect-tolerant microfluidic biochips", *Proc. DAC*, pp. 825-830, 2005.
- [17] W. Hwang et al., "Automated design of pin-constrained digital microfluidic arrays for lab-on-a-chip applications," *Proc. DAC*, pp. 925-930, 2006
- [18] S. K. Fan et al., "Manipulation of multiple droplets on N×M grid by cross-reference EWOD driving scheme and pressure-contact packaging", *Proc. IEEE MEMS Conf.*, pp. 694-697, 2003.
- [19] F. Su, W. Hwang and K. Chakrabarty, "Droplet routing in the synthesis of digital microfluidic biochips" *Proc. DATE Conf.* pp. 323-328, 2006.
- [20] Connect5 strategies, www.springfrog.com/games/gomoku/
- [21] R. Diestel, *Graph Theory*, Springer, Berlin, 2005.