A Low-Cost Field-Programmable Pin-Constrained Digital Microfluidic Biochip

Daniel T. Grissom, Jeffrey McDaniel, and Philip Brisk, *Member, IEEE*

***Abstract*—This paper introduces a field-programmable pin-constrained digital microfluidic biochip (FPPC-DMFB), which offers general-purpose assay execution at a lower cost than general-purpose direct addressing DMFBs and highly optimized application-specific pin-constrained DMFBs. One of the key cost drivers for DMFBs is the number of printed circuit board (PCB) layers, onto which the device is mounted. We demonstrate a scalable single-layer PCB wiring scheme for several FPPC-DMFB variations, for PCB technology with orthogonal routing capacity of at least three; for PCB technology with orthogonal capacity of two, more PCB layers are required, but the FPPC-DMFB retains its cost advantage. These results offer new insights on the relationship between PCB layer count, pin count, and cost. Additionally, to reduce the execution time of assays on the FPPC-DMFB, we present efficient algorithms for droplet routing, with and without contamination removal via wash droplets.**

***Index Terms*—Digital microfluidic biochip (DMFB), PCB escape routing, pin-constrained DMFB.**

I. Introduction

A FIELD-PROGRAMMABLE, pin-constrained digital microfluidic biochip (FPPC-DMFB), can execute any assay (biochemical protocol) after the device has been manufactured [11]. Prior general-purpose DMFBs were based on direct addressing, which provides independent control over each electrode in the device, at the cost of a high number of control pins, which requires expensive multilayer printed circuit boards (PCBs). Prior pin-constrained DMFBs, allow each control pin to drive multiple electrodes [35], reducing the number of control pins and PCB layers; however, they have all been application-specific [8], [17]–[20], [22]–[24], [27], [35], [37], [39]–[41], which limits their usability. In contrast, the FPPC-DMFB is general purpose and can be implemented in one PCB layer. We demonstrate that the FPPC-DMFB is cheaper than direct addressing and prior application-specific pin-constrained DMFBs. Our results and analysis provide new understanding into the relationship between pin count, layer count, and the actual cost of the chip. We also demonstrate efficient droplet routing algorithms for the FPPC-DMFB, and establish the overhead of contamination removal via wash droplet routing.

Manuscript received April 3, 2014; revised June 9, 2014; accepted July 15, 2014. Date of current version October 16, 2014. This work was supported by the National Science Foundation (NSF) under Grant CNS-1035603. The work of D. T. Grissom was supported by an NSF Graduate Research Fellowship. A preliminary version of this paper was presented at the IEEE/ACM 2013 Design Automation Conference, Austin, TX, USA, Jun. 2013. This paper was recommended by Associate Editor T.-Y. Ho.

D. T. Grissom is with the Department of Computer Science, Azusa Pacific University, Azusa, CA 91702 USA (e-mail: dgrissom@apu.edu).

J. McDaniel and P. Brisk are with the Department of Computer Science and Engineering, University of California Riverside, Riverside, CA 92521 USA (e-mail: jmcda001@ucr.edu; philip@cs.ucr.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.2014.2347931

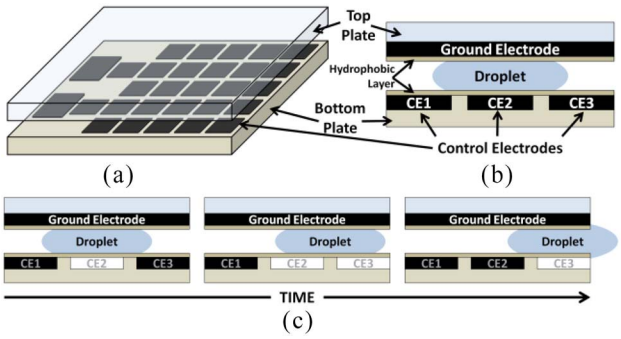

Fig. 1. (a) DMFB is a planar array of electrodes. (b) Cross-sectional view. (c) Droplet is transported from CE2 to CE2 by activating CE3, and then deactivating CE2 (white: activated; black: deactivated).

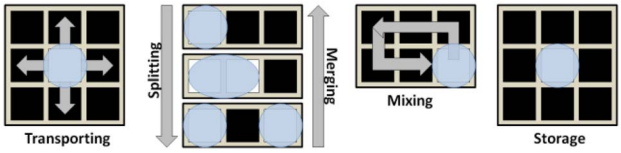

Fig. 2. Fundamental microfluidic operations form instruction set of a DMFB; these operations can be combined to form larger assays.

A. DMFB Technology Overview

1) Background: Physical Droplet Manipulation: DMFBs execute assays by manipulating nanoliter-sized droplets of fluid. DMFBs are based on a phenomenon known as electrowetting [30]. A DMFB, shown in Fig. 1, consists of top and bottom plates coated with a hydrophobic layer. The bottom plate contains an array of droplet-sized control electrodes, while the top plate has one conducting electrode that spans the entire array [Fig. 1(a)]. Each droplet is sandwiched between the two plates and remains in place when its underlying electrode is activated. If a droplet is not centered on an activated electrode, it will unpredictably drift across the DMFB in; thus, an electrode underneath a droplet must be activated to store it in place. In Fig. 1(b), a droplet centered on electrode CE2 overlaps neighboring electrodes CE1 and CE3. In Fig. 1(c), activating CE3 pulls the droplet to the right, and deactivating CE2 centers the droplet over CE3.

Fig. 2 depicts the instruction set of a DMFB: droplet transport, splitting, merging, mixing, and storage.

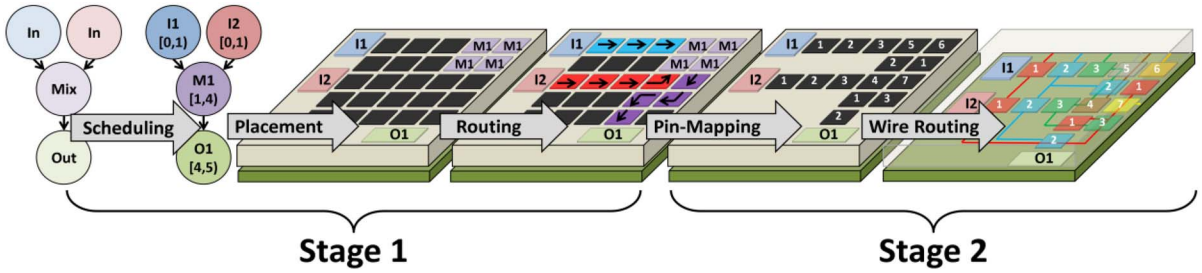

Fig. 3. Typical microfluidic synthesis flow dictates that a microfluidic assay is represented in the form of a DAG. In Stage 1, its operations are scheduled and placed onto the DMFB array and droplets are routed between operation locations. In Stage 2, pin-mapping and wire routing are performed to eliminate unused electrodes and connect the electrodes to an external edge of the device to be driven by a microcontroller.

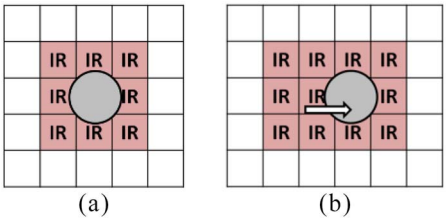

Fig. 4. Interference region (IR) for a droplet at (a) beginning and (b) end of a droplet-actuation cycle.

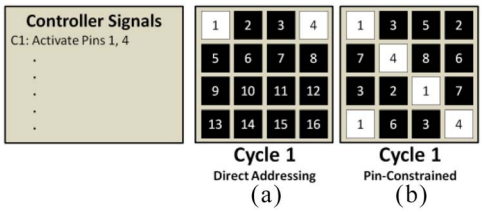

Fig. 5. Activating a pin on (a) direct-addressing DMFB activates (white) exactly one electrode per pin and (b) pin-constrained DMFB activates one + electrodes per pin, depending on the pin layout.

Sensor-based detection operations execute by moving a droplet to a detector (placed above an electrode) and storing the droplet in-place. Dispense and output operations are performed by I/O reservoirs on the perimeter of the DMFB.

2) Background: High-Level Assay Synthesis: Synthesis is the process that maps a biochemical reaction (an assay), onto a DMFB, as shown in Fig. 3. An assay is specified as a directed acyclic graph (DAG), where vertices represent fluidic operations (e.g., mix, split, etc.) and edges represent precedence and droplet transfer between operations. The DAG in Fig. 3 depicts a simple assay that inputs two droplets, mixes them, and outputs the resulting droplet.

The first step of synthesis is scheduling, which assigns start and stop time to each operation (e.g., the mix operation, *M1*, executes from time Steps 1–4 in Fig. 3). Next, the placement step decides which DMFB locations perform each operation. For dispense (input) operations, a reservoir containing the appropriate fluid is chosen. "reconfigurable" operations such as mixing and splitting can be performed anywhere on the chip; for example, in Fig. 3, the mix operation *M1* is placed in the 2×2 array of cells in the top-right corner of the DMFB; however, *M1* could be placed in any unoccupied 2×2 array of cells on the DMFB. The array of cells that denotes the location of an operation is called a module. Lastly, the routing step computes paths for droplet transport between operations that have been scheduled and placed. The router produces a list of electrodes to activate during each droplet actuation cycle, i.e., the time it takes to move a droplet from one electrode to the next. While computing droplet routes, the router must ensure that droplets do not interfere with one another while traversing the DMFB array [12], [13], [34]; to prevent droplets from accidental collisions, the router creates an interference region around a stationary droplet at the beginning of a droplet actuation cycle, as seen in Fig. 4(a). As the droplet moves to an adjacent electrode, the interference region stretches to include all the electrodes surrounding the droplet's initial and final electrode [Fig. 4(b)]. Collisions are prevented by ensuring that no droplet enters the interference region of any other droplet (unless they are about to merge).

3) Background: Pin Mapping and PCB Wire Routing: Synthesis typically targets an existing DMFB that has been designed and fabricated; alternatively, it can be coupled with pin mapping and wire routing phases to produce an optimized application-specific chip. The default pin mapper instantiates a direct addressing DMFB, in which each electrode is wired to an external electrical control pin, as shown in [Fig. 5(a)]. Direct addressable DMFBs offer the most flexibility in terms of droplet coordination; however, the large number of control pins can increase the 2-D area, and thus cost, of the PCB on which the DMFB is mounted [Fig. 6(a)].

Pin mapping converts a direct-addressing DMFB into a pin-constrained DMFB by tethering multiple electrodes together so that a single control pin can activate them [Fig. 5(b)]. In Fig. 3, the pin-mapper removes the 10 nonused electrodes and connects the remaining ones in to reduce the number of control pins from 15 to 7. Reducing the number of control pins reduces the overall cost of the device [23], [35], [37], [39], [41].

PCB wire routing establishes physical connections within the PCB from each external control input to the electrode(s) that it drives, as determined by the pin mapper. In Fig. 1(b), electrodes reside on the lower substrate of the DMFB, while wire routing is performed within the PCB [shown in green in Figs. 3 and 6(b)]. PCB wire routing for pin-constrained DMFBs is a multiterminal variant of the PCB escape routing problem [6], [25], [36]. Effective escape routers can reduce the number of PCB layers, and thus the overall cost of the device.

Past work on this topic has focused on the number of control pins and/or the number of PCB layers as a proxy for PCB cost. In this paper, we estimate the actual cost of the PCB layout in $ and discuss the various tradeoffs involved.

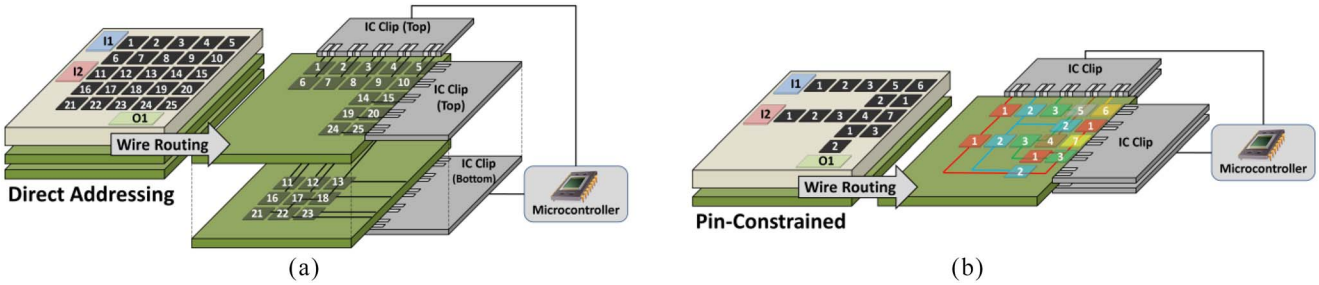

Fig. 6. DMFB has PCBs (green layers) underneath the substrate containing the control electrodes that serve as the medium for wire-routing. A microcontroller sends signals to and interfaces with the DMFB via one or more integrated circuit (IC) clips. (a) Direct addressing DMFB is thought to require multiple layers of PCB. (b) Pin-constrained DMFB performs is thought to require fewer PCB layers.

B. Contribution

The FPPC-DMFB is the first pin-constrained DMFB that is also general-purpose [11]. This paper introduces the FPPC-DMFB pin assignment scheme, describes a synthesis flow that targets it, and presents a PCB wire routing solution that minimizes its overall cost. Our results establish the feasibility of concurrent assay operations and droplet transport on the FPPC-DMFB, quantify the performance overhead of cross-contamination elimination via wash droplet routing, and show that a PCB designed for an optimized FPPC-DMFB can be significantly cheaper than a PCB designed for both direct-addressing and application-specific pin-constrained DMFBs.

II. Related Work

Early pin mappers tried to minimize the pin count, but did not consider the impact of pin count reduction on the number of PCB layers. Under array partitioning [35], different groups of control pins are assigned to each partition, reducing or eliminating interferences among droplets that are transported concurrently. Broadcast electrode addressing [41] examines the electrode activation sequence produced by a synthesis tool and identifies electrodes that can share a control input. Luo and Chakrabarty [24] introduced a pin assignment scheme that facilitates interference-free and deadlock-free concurrent transport of up to two droplets. Several other papers optimize pin assignment in conjunction with other synthesis tasks, especially droplet routing [8], [17], [22], [23], [27], [40].

Escape routing for PCBs routes known pins in a large array to the array perimeter [25], [36]. For pin-constrained DMFBs, the escape routing problem must accommodate multiterminal nets for control inputs that drive multiple electrodes. One paper has been published that focuses explicitly on escape routing for DMFBs [6], while another optimizes the PCB layout for multiple DMFBs that execute the same protocol concurrently in a lock-step [31]. Several papers have also been published that optimize pin assignment in conjunction with escape routing [18]–[20], [37]; they optimize application-specific, not than general-purpose pin-constrained DMFBs.

The FPPC-DMFB is a pin-constrained virtual topology [10], [12], [13], which segregates the DMFB surface area into modules that perform assay operations (mixing, splitting, storage, detection, etc.) and a network of streets that transport droplets between modules and I/O reservoirs.

In a direct-addressing chip, virtual topologies limit the flexibility and reconfigurability of the device in order to facilitate fast online synthesis algorithms, which can respond to sensory feedback provided by the device in real-time [12], [13]. Pin-constrained DMFBs exhibit limited flexibility and reconfigurability; imposing a virtual topology to achieve general-purpose operation is a favorable innovation [11].

Chang *et al.* [7] introduced a pin-constrained DMFB that shares some similarities with the FPPC-DMFB proposed here. Their device does not account for some of the finer details of module/device synchronization and I/O addressed in this paper (e.g., the ability to independently load/unload droplets into modules). It is also unclear if the layout and wiring solution is scalable to larger devices. In contrast, this paper presents a design variation of the FPPC-DMFB which can be routed in one PCB layer, and can scale to arbitrary numbers of operational and storage modules.

III. Pin Assignment

The FPPC-DMFB employs a pin assignment scheme that enables all of the basic assay operations (Fig. 2) to execute in a conflict-free manner. Fig. 7 shows two similar, but different, 10×16 FPPC-DMFB layouts. For simplicity we first focus on the pin-optimized version in Fig. 7(a) to show the general characteristics of our FPPC-DMFB design.

Similar to virtual topologies [10], [12], [13], the FPPC-DMFB reserves specific regions for assay operations and others for routing. The FPPC-DMFB contains a vertical column of mixing modules on the left (blue/orange electrodes, *Pins 10–20*) and a vertical column of modules on the right (orange electrodes, *Pins 31–36*) that perform splitting, storage, and detection (which requires an external detector affixed above the module); we call these modules SSD modules.

White electrodes define droplet routing regions, which ensure full connectivity between all modules. I/O reservoirs can be placed anywhere along the top or bottom of the chip, as seen in Fig. 7. The green electrodes, *Pins 21–30*, allow droplets to enter/exit each module. An interference region (gray) surrounds each module to isolate droplets within it from droplets in the routing region or adjacent modules; these regions are not functional and do not contain electrodes.

The layout is designed for operation concurrency; mix and SSD modules can execute different operations that may start and stop at any time-step (i.e., droplets may enter/exit modules while other modules continue operating). The architecture is scalable and can be vertically lengthened or shortened to produce a DMFB with any desired number of modules.

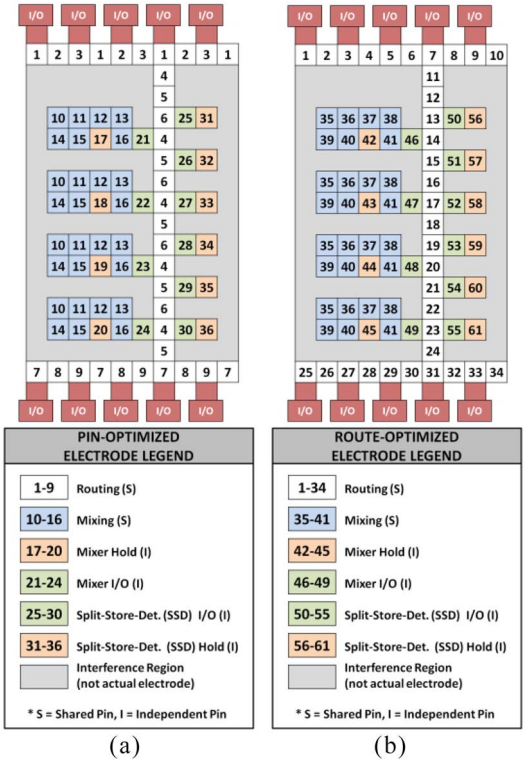

Fig. 7. Pin assignment scheme for a 10 × 16 FPPC-DMFB which can accommodate four mixing modules and six split/store/detect (SSD) modules. Mixing pins are shared; the interference region is empty space and does not contain any electrodes. Holding and I/O electrodes are independently wired to single control pins for flexibility and programmability. Fluidic I/O reservoirs are connected to the top/bottom horizontal buses. (a) Pin-optimized FPPC-DMFB version with shared routing pins. (b) Route-optimized FPPC-DMFB version with independent routing pins.

A. DMFB Operations and Synchronization

1) Droplet Transport: The FPPC-DMFB facilitates droplet transfer between horizontal and vertical transport buses. Routable paths exist between all modules and I/O reservoirs on the chip's perimeter. Chips of arbitrary height can be instantiated without altering the wire-routing pattern (see Section IV-B). The mix and SSD module-hold electrodes [Fig. 7(a)], *Pins 17–20* and *31–36*, respectively) remain active during routing to ensure that droplets within the modules do not drift.

Fig. 7 presents pin-mappings for pin-optimized and route-optimized designs of the FPPC-DMFB, which share the same topology (white electrodes). The route-optimized design has a higher cost (more control pins), but offers more opportunities to transport multiple droplets concurrently (Section VII).

a) Pin-optimized FPPC-DMFB: Fig. 8 shows that at least three pins are required to successfully transport a droplet along a straight path; this is called a 3-phase transport bus [32]. In Fig. 7(a), *Pins 1–3* and *7–9* control two horizontal buses; *Pins 4–6* drive a vertical transport bus at the center of the array.

3-phase transport buses cannot hold droplets in the routing area while other droplets enter/exit a module. For example, Fig. 9 shows two droplets in the vertical routing bus: for the lower droplet to enter the lower mixing module, the DMFB must activate *Pin 20*, while simultaneously deactivating *Pin 4*, as the electrode underneath a droplet must be activated to hold it in place. Activating *Pins 4*, *5*, or *6* to transport or hold

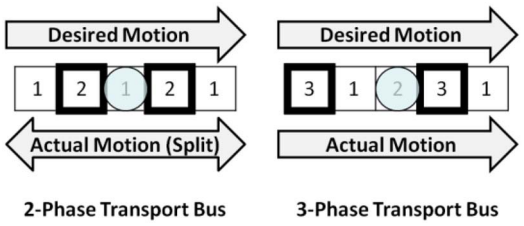

Fig. 8. At least three repeatable pins are needed to move a droplet along a straight path without causing the droplet to split. Electrodes with bold borders indicate electrodes being activated next cycle.

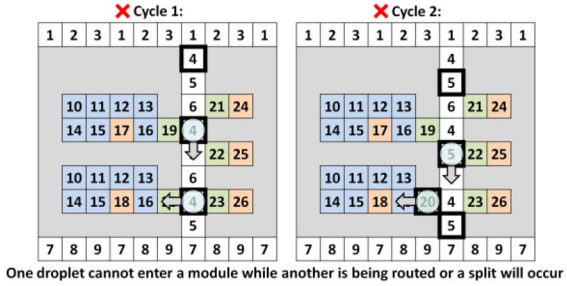

Fig. 9. Multiple droplets moving through the vertical bus will result in an unintentional split when one tries to enter a module.

the upper droplet, will inadvertently split the lower droplet. The supplemental section of [11] elaborates on the futility of concurrent droplet routing with the pin-optimized design.

b) Route-optimized FPPC-DMFB: Replacing 3-phase transport buses with direct addressing buses rectifies the situation in Fig. 9. The route-optimized design increases the control pin count by 25 to facilitate concurrent droplet routing; the pin assignment for mixing and SSD modules is unchanged.

2) Droplet Dispensing and Output: I/O reservoirs are attached to the top and bottom horizontal transport buses, and have individually addressable electrodes, (red in Fig. 7) to allow droplets to enter/exit the chip.

3) Merging/Mixing: Fig. 10(a) shows a droplet (*D2*) entering and exiting a mixing module (*M2*) without conflicting with droplets in other modules (*D1*, *D3*). On top, *D2* reaches the electrode adjacent to the mixing module (*M2*); *D1* is in mixing module *M1* and *D3* is stored in SSD module *SSD1*. All SSD module electrodes are activated (*Pins 24–26*) to hold the stored droplets in place during mixing module I/O. Activating *Pin 20* (*M2s* I/O cell) moves *D2* adjacent to *M2*. Activating *Pin 16* draws *D2* into *M2*, while transporting *D1* to an adjacent cell within *M1*. Next, all mixer hold cells (*Pins 17* and *18*) move *D1* and *D2* to identical positions within *M1* and *M2*. The electrode sequence is reversed when a droplet exits a mixing module.

Two droplets must merge before mixing, as shown in Fig. 11: droplet *D4* merges with droplet *D2* in *M2* forming a new droplet, *D5* (with twice the volume). Once merged, *D5* is synced with *D1* back to the mixers' hold locations (Cycle 4, Fig. 11). Mixing can then begin, presuming that *D1* is merged.

M1 and *M2* perform concurrent synchronized mixing by activating *Pins 10–16*, in sequence, starting with *Pin 15* and continuing counterclockwise (i.e., *Pins 10–16*), followed by *Pins 17* and *18* together. Mixing can pause if one droplet needs to enter or exit any mixing module.

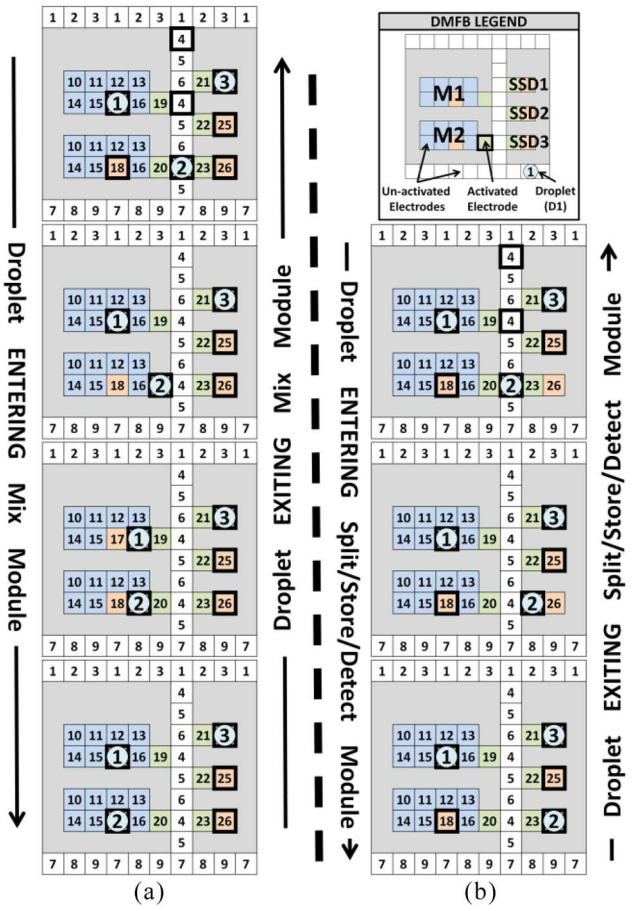

Fig. 10. Pin-activation sequence showing how a single droplet (*D2*) can enter/exit (a) mix modules and (b) split/store/detect modules. Sequences are designed to allow a droplet to enter/exit any module without adversely affecting droplets (*D1*, *D3*) in other modules.

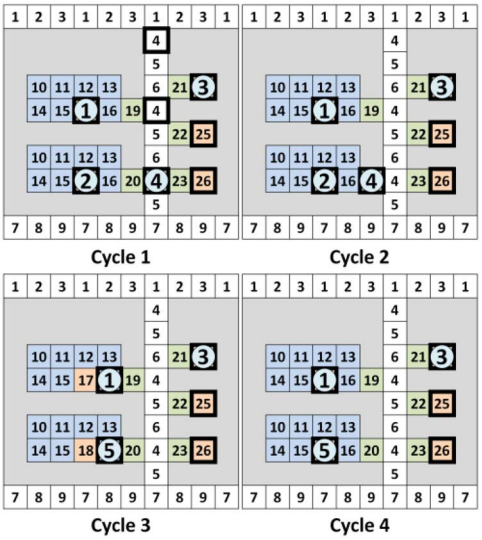

Fig. 11. Electrode/pin activation sequence (from Cycles 1–4) that merges *D4* with *D2* (in *M2*) to become *D5* (twice the volume) and resync with any other droplets in mix modules (i.e., *D1* in *M1*).

4) Storage, Detection, and Splitting: SSD modules perform storage and detection (if equipped with an external detector). Both operations require a droplet to enter an SSD module and remain in place. Fig. 10(b) illustrates a droplet entering/exiting an SSD module (*SSD3*) without affecting droplets in other modules. All SSD hold electrodes are activated, except for *SSD3s*, which allows *D2* to enter. *SSD3s* I/O electrode is then activated, followed by its hold electrode, to complete the entrance. This sequence is reversed to let a droplet exit an SSD module.

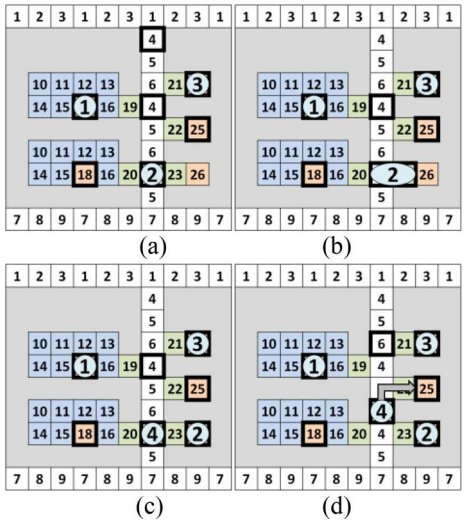

Fig. 12. Pin-activation sequence for droplet splitting and storage using SSD modules: operations sequences are allow droplet *D2* to split and store without adversely affecting droplets *D1*, *D3* in other modules. (a) Cycle 1. (b) Cycle 2. (c) Cycle 3. (d) Cycles 4–7.

Fig. 12(a)–(c) illustrates splitting. Droplet *D2*, which will be split, starts on a vertical transport bus by an SSD module's I/O cell; this cell remains activated throughout the split. Next, the I/O cell is activated, stretching *D2*. Then the SSD module's hold cell is activated and the I/O cell is deactivated, splitting *D2* between the hold cell (*D2*), and the bus (*D4*). *D4* can then be routed to an available SSD module for storage [Fig. 12(d)].

IV. Hardware Layout Optimization

A. Problem Formulation

This section introduces a deterministic and scalable co-optimized pin assignment and single-layer PCB wire routing solution for the FPPC-DMFB. The input is an architecture description, which includes the XY-dimensions of the chip, locations of I/O ports on the periphery of the horizontal buses, PCB feature sizes and a flag that indicates whether the user wants a pin-optimized or route-optimized FPPC-DMFB. The output is a programmable pin-mapping solution, which fits the user-provided dimensions, and a valid single-layer PCB wire routing solution, which adheres to the pin mapping solution.

B. Co-Optimizing Pin Assignment and Wire Routing

As a motivating example, consider a pin assignment for a pin-constrained 15 × 15 assay-specific DMFB designed for the PCR assay [41]. Fig. 13(a) shows a 14-pin layout and highlights the wire routing solution for *Pin 1*; *Pin 1* drives nine electrodes, many of which are on the perimeter of the chip. The wire routing solution for this one pin effectively blocks the ability to route additional wires into the chip on the same PCB layer. Fig. 13(b) shows a complete wire routing solution for all 14 pins; a total of four PCB layers are required.

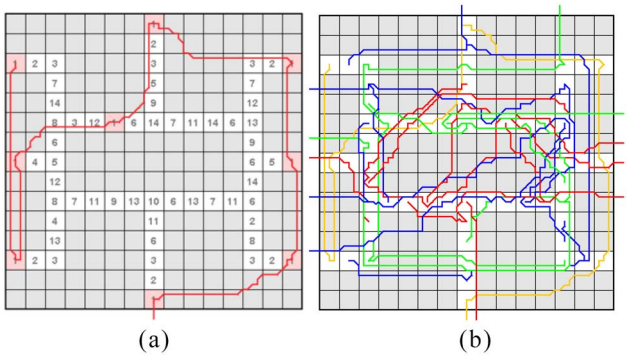

Fig. 13. (a) Pin-mapping for a pin-constrained DMFB for a PCR assay [41] detailing a wire-routing solution for *Pin 1*. (b) Complete four-layer, wire-routing solution (each layer is represented by a different color). Note: Gray cells do not contain electrodes.

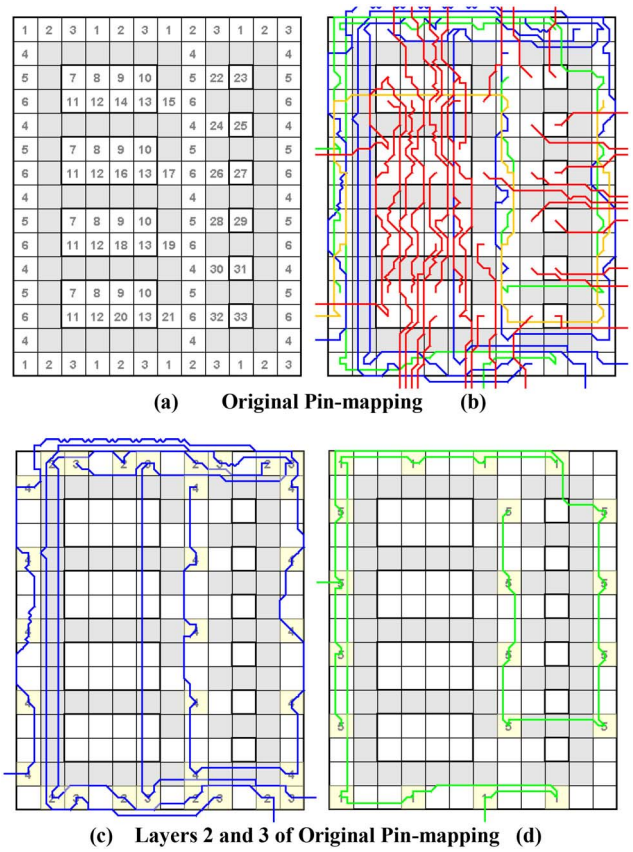

Fig. 14. Original FPPC-DMFB [11] detailing (a) pin-mapping and (b) four-layer wire-routing solution (each color represents a separate layer). (c) Layer 2 from (b) illustrates that *Pins 2* and *3* from the horizontal buses and *Pin 4* from the vertical buses prevent other pins from escaping. (d) Layer 3 from (b) shows that *Pin 1* from the horizontal bus and *Pin 5* from the vertical bus prevent other pins from escaping.

Fig. 14 presents two pin mapping and wire routing solutions for two FPPC-DMFB variants. Fig. 14(a) presents the original pin mapping architecture [11], and Fig. 14(b) shows the wire-routing solution obtained by the negotiated-congestion escape router [25]. This particular variant has three vertical buses (as opposed to the one central vertical bus shown in Fig. 7. Four PCB layers are required for routing, as shown in Fig. 14(b).

Fig. 14(c) and (d) depicts two of these four wire-routing layers. Wires that connect to electrodes on the 3-phase buses must

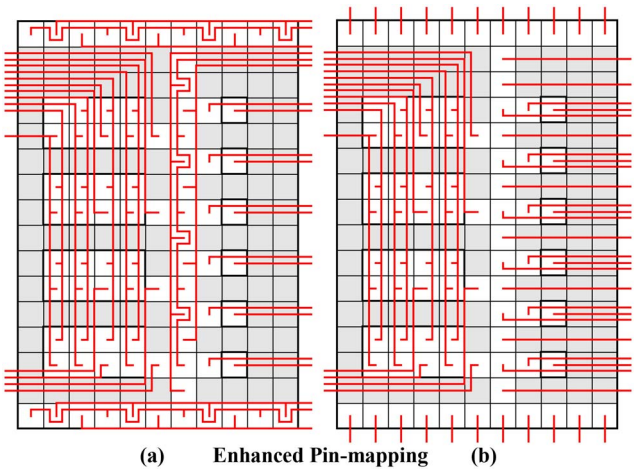

Fig. 15. Single-layer wire-routing solutions for the enhanced FPPC-DMFB designs in this paper for (a) pin-optimized version [Fig. 7(a)] and (b) route-optimized pin-mapping [Fig. 7(b)].

span the entire array, essentially blocking the ability of other wires to escape from the perimeter on the same PCB layer. To eliminate this problem for the pin-optimized version, we removed the two side buses and use separate three-phase buses (*Pins 1–3*, *4–6*, *7–9*) to control the three remaining buses, as shown in Fig. 7(a). This yielded a single-PCB layer wire routing solution, shown in Fig. 15(a). The route-optimized version also removes the periphery vertical buses, but uses individually addressable electrodes, as seen in Fig. 7(b). Fig. 15(b) reveals the single-layer wire-routing solution, showing that the individually addressable bus electrodes are easily escapable and do not create large obstacles for other pins. We have implemented an algorithm to generate these escape routes for FPPC-DMFBs of varying vertical length.

Removing the left and right vertical buses may reduce the number of potential I/O locations; however, if extra I/O is required, the horizontal buses at the top and/or bottom can be extended; alternatively, mixing or SSD modules in the center of the chip could be replaced with an I/O reservoir attached to the central vertical bus. Another subtle detail is that an extra horizontal row is added between the top vertical bus and the topmost mixing and SSD modules; this extra space is needed to provide access for control wires that drive electrodes in the center of the chip to escape, as shown in Fig. 15.

The original design assumed that *Pins 7–13* [see Fig. 14(a)] could be shared by an arbitrary number of mixing modules, regardless of the height of the chip; however, because of the independently controlled module hold and I/O pins [*Pins 14–21* in Fig. 14(a)], there is not enough room to extend the shared pins indefinitely without introducing additional PCB layers to facilitate wire routing to these shared electrodes.

The solution is to limit the number of shared electrodes to groups of four continuous mixing modules (Fig. 15). For chips with more than four mixing modules (see Fig. 16) the same layout and wiring pattern as in Fig. 15 is repeated. Fig. 16(a) shows two groups of four mixing modules, while Fig. 16(b) generalizes the scheme to an arbitrary number. This approach generalizes to the route-optimized design as well.

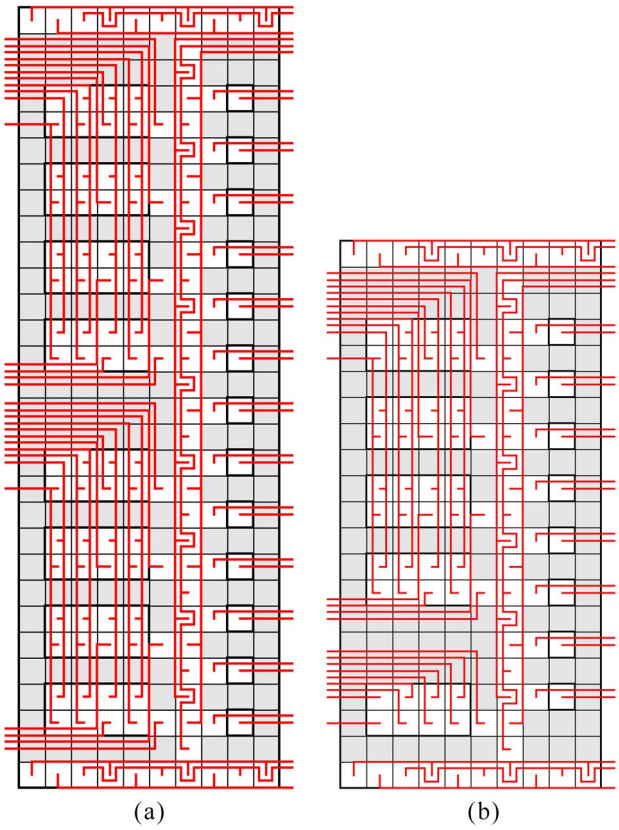

Fig. 16. Wire-routing model for the FPPC-DMFB (pin-optimized version shown) generalizes to an unlimited number of modules; each group of up to four mixing modules shares seven common pins as seen in FPPC-DMFBs with (a) eight mixing modules and (b) five mixing modules.

C. Escape Routing Details

The orthogonal capacity of a PCB is the number of wires that can route between two orthogonally adjacent electrodes. We have assumed an orthogonal capacity of three throughout this section [20], [37]; this allows for a diagonal capacity of six (i.e., at most six wires can be routed between diagonally adjacent electrodes); reference [36] provides more details on modeling these capacities. All escape routing results for architectures other than the FPPC-DMFB presented in this paper were obtained using an internally implemented multiterminal variant of an escape routing algorithm based on negotiated congestion [25].

V. High-Level Synthesis

A. Problem Formulation

The input to the high-level synthesis stage is an architecture description (array dimensions, I/O location, pin assignment), and a DAG representing an assay. After scheduling, placing, and routing the DAG, the output is a valid electrode activation sequence that executes all steps of the assay on the device.

B. Scheduling

Schedulers targeting direct addressing DMFBs treat the device as being reconfigurable, where any operation other than I/O or detection can be performed anywhere. When targeting the FPPC-DMFB, the number of mixing and SSD modules impose a resource limit. We modified list scheduling [14], [33] and path scheduling [15] to target the FPPC-DMFB.

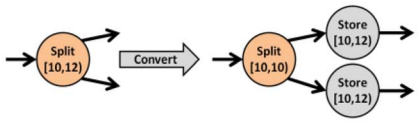

Fig. 17. Split operations are converted to a split and two stores for synthesis.

Modules in direct-addressing DMFBs can perform mixing and storage, and may store multiple droplets. Schedulers targeting direct-addressing DMFBs may route stored droplets from one module to another in order to free up modules to perform other operations [14], [15], [29]. Since SSD modules store at most one droplet, a scheduler targeting the FPPC-DMFB can be eliminate these unnecessary routing transfers.

The scheduler reserves one SSD module to address routing deadlocks, as explained later in Section V-D. Thus, in Fig. 7, only five of the six SSD modules are available for general use.

C. Placement/Binding

Similar to placement algorithms targeting direct-addressing virtual topologies [12], [14], we reduce placement to a binding problem, which is solved using the left-edge algorithm [21]. Synthesis software targeting the FPPC-DMFB does not bind a split operation to a module, as the split yields two immediate storage operations (Fig. 17). Instead, the software binds the children to the SSD modules directly.

D. Droplet Routing

A routing sub-problem refers to the set of droplets that must be routed just before each time-step begins [34], [38]. We refer to the routes that transport droplets between operations in the original assay specification as functional routes. Routes are computed one-at-a-time, ignoring, for the moment, other droplets that also need to be routed during the same time-step: 1) to route a droplet from an input reservoir to module, the router computes a deterministic path over the horizontal and vertical buses, and applies the appropriate module input sequence when the droplet arrives (Section III-A); 2) a similar approach is taken to route droplets from modules to output reservoirs; and 3) module-to-module routing uses the vertical column in the center of the chip, applying appropriate input/output sequences at the start/end of the route.

1) Dependencies and Deadlock: Routing deadlock occurs when one or more droplets wait for resources to become available that will never become free; for example, Fig. 18(a) shows a cyclic dependency involving two droplets. To break the cycle, one droplet (*D3*) is routed to an empty SSD module (*SSD2*), as shown in Fig. 18(b). In Fig. 18(c), the dependency is broken, but droplet *D3* must wait for *D1* to complete its route. The scheduler reserves one SSD module as a routing buffer to break any cyclic dependencies that may result from binding.

2) Wash Droplet Routing: A droplet traveling on a DMFB may leave residue behind that contaminates other droplets. Wash droplets may be dispatched to clean electrodes that have been contaminated by a previous droplet; a future droplet can then travel over the cleaned electrode without being contaminated.

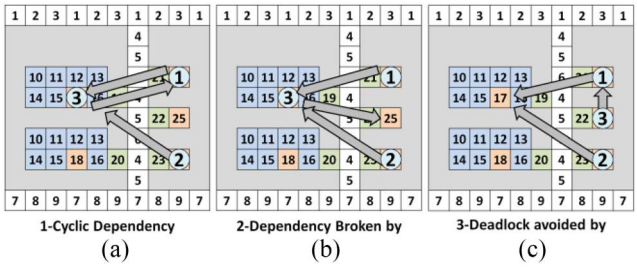

Fig. 18. Cyclic routing dependencies can be broken by routing a droplet in the cycle to an SSD module, dedicated as a buffer. Arrows indicate that the droplet at the tail end is about to travel to the module at the head end.

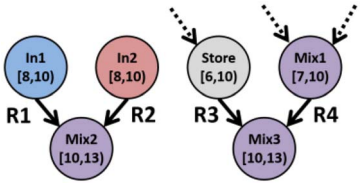

Fig. 19. Partial DAG in which four droplets must be routed (R1–R4) to begin time-step 10 (dotted arrows convey routes in previous time-steps).

In Fig. 19, operations *Mix2* and *Mix3* each demand two droplets to be routed to their assigned modules at time-step 10; Fig. 19 depicts these routes as *R1-R4*. Since *R1* and *R2* travel to the same destination to be merged, they cannot contaminate one another and no wash droplet is needed; the same is true for *R3* and *R4*. Assume that *R1* and *R2* are routed first; if either *R3* or *R4* crosses over any of the same electrodes as *R1* or *R2*, contamination could occur and must be prevented.

First, we designate dedicated wash fluid input and output ports on the top and bottom horizontal buses, respectively. We then identify all operations starting at the given time-step (e.g., for time-step 10 in Fig. 19, *Mix2* and *Mix3*). The first operation is examined and routes are generated to its assigned module (or output port) from all of its parents; without loss of generality, assume *Mix2* is being examined, and thus, *R1* and *R2* are generated to transport droplets from the input reservoirs to *Mix2s* mix module. Next, a wash droplet is generated and a route is computed to cover all of the electrodes covered by *R1* and *R2* on any of the transportation buses. Since the wash I/O ports are on opposite sides of the array, the entire vertical bus is cleaned each during each washing step. When considering *Mix3*, where routes are being cleaned that begin in a module (*R3* begins in a SSD module, *R4* begins in a mix module), the wash droplet enters and cleans each source module on its way down the vertical bus, cleaning all contaminated electrodes in both transportation buses and modules.

3) Concurrent Routing: As discussed in Section III the pin-optimized FPPC-DMFB must employ sequential routing to prevent unintended droplet splitting and drifting. Thus, wash droplet routes are inserted and executed after each operation executes its parent routes. In contrast, the route-optimized FPPC-DMFB employs individually addressable buses, which facilitates concurrent functional and wash droplet routing.

Similar to prior work by Grissom and Brisk [13], the concurrent router compacts the sequential routing solution (in time) so that multiple droplets can route in parallel without inadvertently colliding. Their algorithm compacts droplet routes one-at-a-time, delaying the starting time of each route routes until a legal solution is obtained. In the worse case, the compacted solution ends up being sequential.

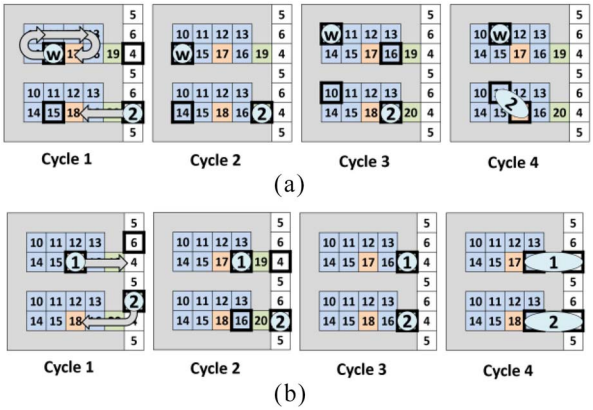

Fig. 20. Small portion of the pin-optimized design showing that shared module pins can cause (a) a droplet (D2) entering a module to be unintentionally split (Cycle 4) if a wash droplet (w) is cleaning another module and (b) droplets exiting (D1) and entering (D2) separate modules result in an unintended split (note: in the route-optimized design, D1 will split while D2 will remain intact).

To incorporate wash droplets and the FPPC-DMFBs virtual topology, several new compaction rules are introduced.

1) No droplet may cross a contaminated electrode.
2) Two mixing modules cannot be cleaned at the same time; a droplet may not enter or exit any mixing module while another is being cleaned.
3) Two (or more) droplets may not simultaneously enter and exit two distinct mixing modules at the same time.

If any of these rules is broken during compaction, stalls are inserted at the beginning of the route until legality is obtained.

To address *Rule 1*, a wash droplet must be routed before the current droplet can be compacted. *Rules 2* and *3* are specific to the FPPC-DMFBs shared mixing module pins (see Fig. 7). To illustrate *Rule 2*, Fig. 20(a) shows how a droplet entering a module during mixing may unintentionally split because of the simultaneous pin activations due to wash droplet traversal. Fig. 20(b) illustrates *Rule 3:* two droplets simultaneously entering and exiting mixing modules may inadvertently split.

Once all routes are compacted, we can reduce the length of the routing sub-problem. Typically, operations such as mixing are paused during the routing portion of an assay to allow all droplets to reach their destination and sync [12], [13]; however, if all droplets still en route to their destinations are traveling to output reservoirs and are not on a mix electrode (shared, hold or I/O), then the next time-step may commence, overlapping a portion of the routing stage with an operational stage. The mix modules can start to activate their shared pins in a lockstep rotational pattern; these activations could interfere with any droplets concurrently exiting a mixing module, but not with droplets undergoing transportation on one of the buses.

VI. Experimental Methodology

A. Wire Routing Cost Analysis

Our experiments estimate the cost (in U.S. dollars) of PCBs for different direct-addressing and pin-constrained DMFB architectures, along with several variants of the FPPC-DMFB.

Prior work has reported pin-count and the number of PCB layers as a proxy for cost, but has not reported the actual costs. This makes it difficult to determine when it is profitable to increase the pin count if doing so reduces the number of PCB layers. Our experimental approach presents a more accurate picture of the tradeoffs involved here.

We use Advanced Circuits' online instant quote feature to estimate the cost of each PCB [1]. We provide the PCB length and width and the number of wire-routing layers. Vias connect multiple layers, and thus it is necessary to specify the via size, along with the wire trace spacing and size, which dictates the thickness of wire traces and the minimum spacing between tracing. All other metrics are left at their default values. Wirelength does not directly affect the cost, as long as the PCB is routable without increasing its area by adding extra space.

We assume that all DMFBs are driven by an Atmega 1284 microcontroller with 32 general purpose I/Os (GPIOs) to address the DMFB array [2]. If a DMFB has 32 or fewer control pins, no additional circuitry is needed; shift registers are required to drive addition pins in excess of 32. Shift registers can be daisy chained to feed an arbitrary number of additional inputs using four microcontroller signals to control the shift register chain: the serial data input (SER), shift register clock input (SCK), storage register clock input (RCK) and the reset input (SCLR). We assume the Fairchild 74VHC595MTC 8-bit shift register [3], which can be purchased from Mouser for \$0.14 per unit in quantities of 2500 [5]. Equation (1) reports the number of shift registers required for a DMFB

$$\text{numShiftRegs} = \begin{cases} \left\lceil \frac{\text{numPins}-28}{8} \right\rceil, & \text{numPins} > 32 \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

The Atmega 1284 operates at 20 MHz [2]; the droplet actuation frequency (i.e., the time it required to transport a droplet between two adjacent electrodes) of a typical DMFB is 100 Hz [38]. With an estimate of five cycles per shift operation, the Atmega 1284 can load 400 pin values into the shift registers in 1% of the droplet actuation cycle, maintaining sufficient signal integrity for proper droplet transportation.

Equation (2) models the PCB wire routing cost as the sum of the PCB price estimate from Advanced Circuits and the shift registers to connect all pins. We do not include the cost of the microcontroller, which is constant, irrespective of the choice of DMFB. We do not consider the cost of circuitry to amplify the voltage produced by the microcontroller to levels appropriate to drive the DMFB. Typical actuation voltages are in the 50–70 V range [28], [30]; low voltage devices that operate at ∼15 V have also been reported [9], [26]

$$\text{Cost}_{WR} = \text{Cost}_{\text{PCB}} + \text{Cost}_{SR}. \quad (2)$$

The wire-routing cost is a function of the number of PCB layers, the PCB area, and wire trace width, as shown in (3). Assuming an orthogonal capacity of three or more, the FPPC-DMFB requires one PCB layer, as shown in Figs. 15 and 16. In all other cases, we use the negotiated congestion escape router [25] to determine the number of PCB layers that achieve a legal route. Using larger feature sizes tends to reduce the PCB cost estimate [1]

$$\text{Cost}_{\text{PCB}} f\left(\text{numLayers}, \text{width}_{\text{PCB}}, \text{height}_{\text{PCB}}, \text{width}_T\right). \quad (3)$$

TABLE I
PCB FABRICATION PARAMETERS

Feature	Symbol
Electrode Pitch	$width_{ELEC}$
Via (Hole) Width	$width_V$
Via (Hole) Contact Width	$width_{VC}$
Wire Trace Width	$width_T$
Min. Space Between Wire Trace	$width_{TS}$
Shift Register Width [4]	$width_{SR}$
Shift Register Height [4]	$height_{SR}$
Spacing Between Shift Registers	$width_{SRS}$

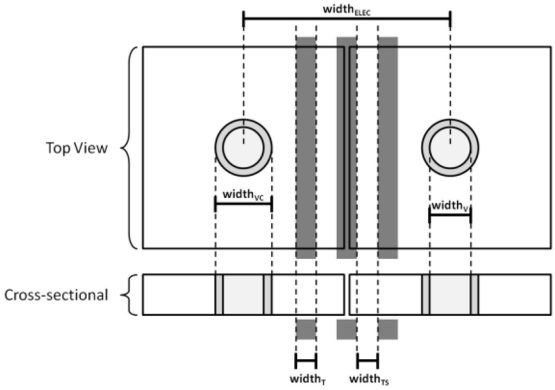

Fig. 21. Top-down and cross-sectional view of a PCB showing dimensions for the electrode pitch (ELEC), via hole (V), via contact (VC), wire trace (T), and minimum wire trace spacing (VS) in a DMFB.

Equation (4) computes the orthogonal capacity according to the metrics in Table I and diagram in Fig. 21. With an electrode pitch of 2 mm, wire trace width and spacing of 0.007 in, via width of 0.014 in and via contact width of 0.024 in, we estimate the orthogonal capacity to be $o_{\text{cap}} = \lfloor 3.12 \rfloor = 3$

$$o_{\text{cap}} = \left\lfloor \frac{\text{width}_{\text{ELEC}} - \text{width}_{VC} - \text{width}_{TS}}{\text{width}_T + \text{width}_{TS}} \right\rfloor. \quad (4)$$

Fig. 22 shows the layout for estimating the PCB dimensions ($\text{width}_{\text{PCB}} \times \text{height}_{\text{PCB}}$). In (5), the PCB height is the array's height plus one inch; (6) shows that the PCBs width may need to account for space allocated for shift registers. The amount of extra space added to the PCB width depends on the number of shift registers, as computed by (7). Shift registers are stacked vertically until there is no more room, at which point additional columns are added

$$\text{height}_{\text{PCB}} = \text{height}_A + 1 \quad (5)$$

$$\text{width}_{\text{PCB}} = \text{width}_A + \text{width}_{\text{PCB\_SR}} + 1 \quad (6)$$

$$\text{width}_{\text{PCB}_{SR}} = \left\lceil \frac{\text{numShiftRegs}}{\left\lfloor \text{height}_{\text{PCB}} / \left(\text{height}_{SR} + \text{width}_{\text{SRS}}\right) \right\rfloor} \right\rceil \times (\text{width}_{SR} + \text{width}_{\text{SRS}}). \quad (7)$$

VII. EXPERIMENTAL RESULTS

We implemented the FPPC-DMFB and associated synthesis algorithms in a publicly available open-source software framework for DMFB compilation [16]. All experiments were performed using a 2.8 GHz Intel Core i7 CPU and 4 GB RAM running a 64-bit version of Windows 7.

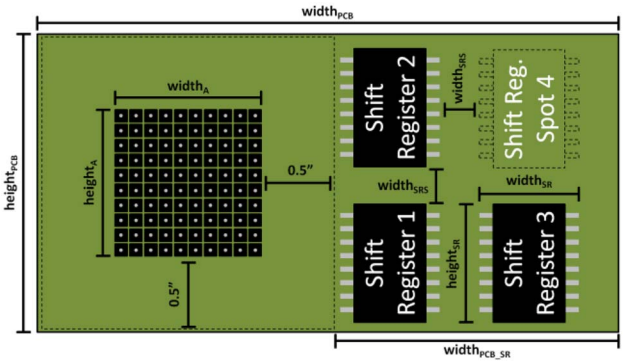

Fig. 22. Component layout for PCB size estimation. The electrode array is surrounded by a 0.5 inch perimeter of empty space. The PCB width is extended to add as many shift registers as necessary.

A. Benchmarks

We extracted eight pin-constrained DMFB layouts from [24] and [41]; these benchmarks are labeled *ZHAO_XXX*, [41] and *LUO_XXX* [24], where "*XXX*" is one of the three assays used in their experiments (*PCR*, *INVITRO*, *PROTEIN*) or a multifunctional chip that is co-designed to perform all three of those assays (*MULTI*). The electrode layouts are identical for both of these works; only the pin-assignment is different. We also created directly addressable versions of the corresponding electrode layouts, entitled *XXX_DA*.

We denote the FPPC as *FPPC_4_MODULE* for the 4-mixer version shown in Fig. 15(a) and *FPPC_8_MODULE* for the 8-mixer version shown in Fig. 16(a). The routing-optimized FPPC-DMFB [Fig. 7(b)] with 4- and 8-mixers are named *FPPC_4_DA_BUS* and *FPPC_8_DA_BUS,* respectively.

We also consider two direct addressing DMFBs that use the 4- and 8-mixer FPPC-DMFB electrode layouts in Figs. 15 and 16(a); these chips are named *FPPC_4_DA* and *FPPC_8_DA*, respectively. Lastly, we include results for three direct-addressing DMFBs having dimensions: of 15×15 (15×15*_DA*), 10×10 (10×16*_DA*), and 10×30 (10×30*_DA*).

PCB escape routing solutions for all FPPC-DMFBs, except for the directly addressable ones, were computed as described in Section IV-B (*FPPC*); all other DMFB wire routes were computed using a multiterminal implementation of an escape router based on negotiated congestion [25]. If a pin's wire net cannot be routed on a top-level layer, it is routed on a lower layer in its entirety using vias to connect the wire net to its corresponding electrodes and to the external driving pin.

B. PCB Layers and Orthogonal Capacity

The left side gives a description of the 21 DMFB wire-routing benchmarks showing the wire-routing algorithm used (WR Algorithm), array dimensions (Array Dims.), number of electrodes (# Elecs.) and number of control pins (# Pins). The right side shows the number of PCB layers yielded for each DMFB as a function of orthogonal capacity, which varies from 2 to 10; dark squares highlight the lowest orthogonal capacity which achieves the minimum number of PCB layers for each chip.

We examine the relationship between orthogonal capacity, the number of control pins and the number of PCB layers.

TABLE II
DMFB BENCHMARK DESCRIPTION AND NUMBER OF LAYERS PER ORTHOGONAL CAPACITY

DMFB Characteristics						Number of Layers Per Orthogonal Capacity								
Name	WR Alg.	Array Dims. X	Array Dims. Y	# Elecs.	# Pins	2	3	4	5	6	7	8	9	10
ZHAO_PCR	[26]	15	15	62	14	**4**	4	4	4	4	5	4	4	4
ZHAO_INVITRO	[26]	15	15	59	25	4	4	**3**	4	4	4	4	4	4
ZHAO_PROTEIN	[26]	15	15	54	27	4	**3**	4	4	4	4	4	4	4
ZHAO_MULTI	[26]	15	15	81	32	5	5	6	6	6	5	**4**	5	5
LUO_PCR	[26]	15	15	62	22	**5**	5	5	5	5	5	5	5	5
LUO_INVITRO	[26]	15	15	59	21	**5**	5	5	5	5	5	5	5	5
LUO_PROTEIN	[26]	15	15	54	21	**4**	4	4	4	4	5	4	4	4
LUO_MULTI	[26]	15	15	81	27	**6**	6	6	6	6	6	6	6	7
PCR_DA	[26]	15	15	62	62	**1**	1	1	1	1	1	1	1	1
INVITRO_DA	[26]	15	15	59	59	**1**	1	1	1	1	1	1	1	1
PROTEIN_DA	[26]	15	15	54	54	**1**	1	1	1	1	1	1	1	1
MULTI_DA	[26]	15	15	81	81	**1**	1	1	1	1	1	1	1	1
FPPC_4_MODULE	FPPC	10	16	82	36	2	**1**	1	1	1	1	1	1	1
FPPC_8_MODULE	FPPC	10	30	146	65	2	**1**	1	1	1	1	1	1	1
FPPC_4_DA_BUS	FPPC	10	16	82	61	2	**1**	1	1	1	1	1	1	1
FPPC_8_DA_BUS	FPPC	10	30	146	104	2	**1**	1	1	1	1	1	1	1
FPPC_4_DA	[26]	10	16	82	82	**1**	1	1	1	1	1	1	1	1
FPPC_8_DA	[26]	10	30	146	146	**1**	1	1	1	1	1	1	1	1
15x15_DA	[26]	15	15	225	225	3	2	2	2	**1**	1	1	1	1
10x16_DA	[26]	10	16	160	160	2	2	**1**	1	1	1	1	1	1
10x30_DA	[26]	10	30	300	300	2	2	**1**	1	1	1	1	1	1

The left side gives a description of the 21 DMFB wire-routing benchmarks showing the wire-routing algorithm used (WR Alg.), array dimensions (Array Dims.), number of electrodes (# Elecs.) and number of control pins (# Pins). The right side shows the number of PCB layers yielded for each DMFB as a function of orthogonal capacity, which varies from 2 to 10; dark squares highlight the lowest orthogonal capacity which achieves the minimum number of PCB layers for each chip.

The right side of Table II shows the number of PCB layers required to route each DMFB as the orthogonal capacity varies from 2 to 10. The dark squares highlight the lowest orthogonal capacity that achieves the smallest number of layers for each benchmark: $16/21 \approx 76\%$ and $19/21 \approx 90\%$ of the benchmarks minimize the number of layers at orthogonal capacities of 3 or 4. For *ZHAO_XXX* and *LUO_XXX*, the minimum number of PCB layers ranges from 3 to 6.

As an example, consider *ZHAO_PCR*, which is shown in Fig. 13. This particular pin mapping requires long wires along the perimeter of the chip, which prevent other wires from escaping on the same PCB layer; thus, additional layers are necessary. Reducing the wire size to increase orthogonal capacity does not reduce the number of PCB layers; however, converting this chip to a direct-addressing design (*XXX_DA*) yields single-layer escape routing solutions for any orthogonal capacity, but increases the control pin count by $2\times$ to $4.4\times$.

The FPPC-DMFBs shown in Table II can be routed in 1 or 2 PCB layers, even at the smallest orthogonal capacities. *FPPC_4_MODULE*s pin count is comparable to *ZHAO_XXX* and *LUO_XXX*, indicating that the FPPC-DMFB offers the advantages of being both general-purpose and low cost.

C. Wire Routing Cost Analysis

We vary the electrode pitch (1, 2, 2.54 mm), and determine the parameters for each that minimize cost. We translate these results into PCB cost estimates for the DMFBs in Table II.

1) Metric Selection: The predominant factors that impact PCB costs are the wire-trace width/spacing ($\text{width}_T/\text{width}_{TS}$),

TABLE III
PRICE ESTIMATIONS FOR VARYING NUMBERS OF LAYERS AND PARAMETERS OF A 2" × 2" PCB

Electrode Pitch		Advanced Circuit Metrics			oCap	2" × 2" Price (@ 2,500 QTY) with Varying Number of Layers				
mm	in	Trace Size/ Space	Via Size	Via Contact Size		1	2	3	4	5
1	0.0394	**0.005**	**0.010**	**0.013**	**2**	**$1.20**	**$1.20**	**$1.73**	**$1.81**	**$2.09**
		0.004	0.008	0.011	3	$1.88	$1.88	$2.43	$2.52	$2.81
		0.003	0.009	0.012	4	N/A	N/A	N/A	N/A	N/A
		0.002	0.008	0.011	5	N/A	N/A	N/A	N/A	N/A
2	0.0787	0.008	0.028	0.038	2	$0.99	$0.99	$1.41	$1.46	$1.79
		0.007	0.014	0.028	3	$0.99	$0.99	$1.41	$1.47	$1.79
		0.006	**0.012**	**0.024**	**4**	**$0.99**	**$0.99**	**$1.46**	**$1.53**	**$1.80**
		0.005	0.013	0.023	5	$1.10	$1.10	$1.63	$1.71	$1.99
		0.004	0.008	0.026	6	$1.88	$1.88	$2.43	$2.52	$2.81
2.54	0.1000	0.012	0.030	0.040	2	$0.99	$0.99	$1.41	$1.41	$1.79
		0.010	0.020	0.030	3	$0.99	$0.99	$1.41	$1.41	$1.79
		0.008	0.018	0.028	4	$0.99	$0.99	$1.41	$1.46	$1.79
		0.006	0.024	0.034	5	$0.99	$0.99	$1.46	$1.53	$1.80
		0.006	**0.012**	**0.022**	**6**	**$0.99**	**$0.99**	**$1.46**	**$1.53**	**$1.80**
		0.005	0.010	0.020	7	$1.20	$1.20	$1.73	$1.81	$2.09
		0.004	0.022	0.032	8	$1.38	$1.38	$1.93	$2.02	$2.31

The left side shows various metrics used for the Advanced Circuit PCB cost estimator [2] and resultant orthogonal capacity (oCap). The right side provides corresponding price estimates from the Advanced Circuit PCB cost estimator for a 2"×2" PCB with varying numbers of layers; dark rows represent the selected metrics for each electrode pitch.

smallest via size (width_V), PCB dimensions (width_{PCB}, height_{PCB}) and number of layers [1]. Table II shows that the number of layers depends on the orthogonal capacity, which depends on the trace and via metrics, as per (4).

Table III presents parameter combinations that yield a range of orthogonal capacities for 1, 2, and 2.54 mm electrodes. Advanced Circuits' PCB cost estimator provides the following wire trace sizes (all in In): 0.0025, 0.003, 0.004, 0.005, 0.006, 0.007, 0.008, 0.010, and 0.012. To achieve each orthogonal capacity, we chose the lowest trace size and highest minimum via size and via contact size such that $\text{width}_V \geq 2 \times \text{width}_T$ and $\text{width}_{VC} \geq \text{width}_V + 0.01$; these requirements were relaxed for the smaller 1 mm electrodes (such that $\text{width}_{VC} \geq \text{width}_V + 0.003$) because there was not enough space to utilize such conservative size estimates.

For each electrode pitch, we select one set of parameters and report the orthogonal capacity. For 1 mm electrodes, estimations were not available for orthogonal capacities of 4 and 5, and we could not find parameters that could achieve an orthogonal capacity of six; we conservatively opted for an orthogonal capacity of two: even as the number of PCB layers increases to 4, the cost is less than a single-layer PCB with orthogonal capacity of three. Fig. 23 shows a two-layer solution for *FPPC_4_MODULE* with orthogonal capacity of two.

For 2 mm and 2.54 mm electrode pitches, we select the metric set corresponding to orthogonal capacities of 4 and 6, respectively. These are the highest capacities before the price increases significantly for each electrode size and reductions in layer count do not typically occur at higher orthogonal capacities, as shown in Table II.

Table III reveals that, in general, as the feature sizes decrease, particularly the wire trace size, the fabrication costs increase, and that 1- and 2-layer solutions are identical for all cases; this is because PCBs can be printed on two sides.

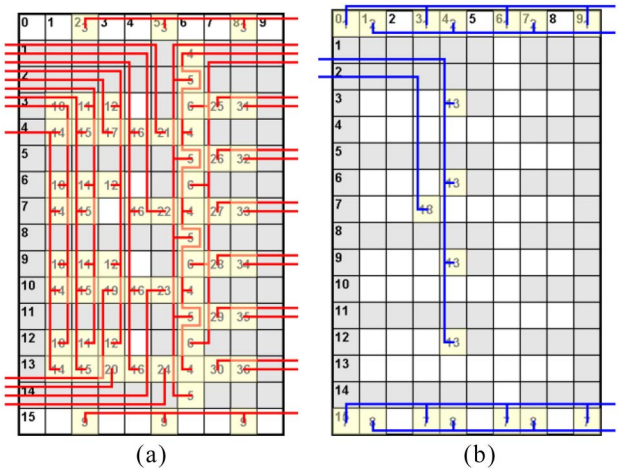

Fig. 23. Two-layer wire-routing solution for *FPPC_4_MODULE* with an orthogonal capacity of two. (a) Layer 1. (b) Layer 2.

TABLE IV
COST ESTIMATES FOR 21 DMFBS WITH 1 MM ELECTRODE PITCH AND ORTHOGONAL CAPACITY OF TWO

DMFB Details						Cost ($)		
DMFB Name	# Pins	# SR	Adjusted PCB Dim X (in)	Adjusted PCB Dim Y (in)	# Layers	Board	SR	Total
FPPC_4_MODULE	36	1	1.7638	1.6299	2	$1.01	$0.14	$1.15
ZHAO_PCR	14	0	1.5906	1.5906	4	$1.33	$0.00	$1.33
ZHAO_INVITRO	25	0	1.5906	1.5906	4	$1.33	$0.00	$1.33
ZHAO_PROTEIN	27	0	1.5906	1.5906	4	$1.33	$0.00	$1.33
LUO_PROTEIN	21	0	1.5906	1.5906	4	$1.33	$0.00	$1.33
ZHAO_MULTI	32	0	1.5906	1.5906	5	$1.55	$0.00	$1.55
LUO_PCR	22	0	1.5906	1.5906	5	$1.55	$0.00	$1.55
LUO_INVITRO	21	0	1.5906	1.5906	5	$1.55	$0.00	$1.55
LUO_MULTI	27	0	1.5906	1.5906	6	$1.58	$0.00	$1.58
PROTEIN_DA	54	4	1.9606	1.5906	1	$1.08	$0.56	$1.64
INVITRO_DA	59	4	1.9606	1.5906	1	$1.08	$0.56	$1.64
FPPC_4_DA_BUS	61	5	1.7638	1.6299	2	$1.01	$0.70	$1.71
PCR_DA	62	5	1.9606	1.5906	1	$1.08	$0.70	$1.78
FPPC_8_MODULE	65	5	1.7638	2.1811	2	$1.16	$0.70	$1.86
MULTI_DA	81	7	2.3307	1.5906	1	$1.12	$0.97	$2.09
FPPC_4_DA	82	7	2.1339	1.6299	1	$1.19	$0.97	$2.16
FPPC_8_DA_BUS	104	10	2.1339	2.1811	2	$1.37	$1.39	$2.76
FPPC_8_DA	146	15	2.5039	2.1811	1	$1.57	$2.09	$3.66
10x16_DA	160	17	2.8740	1.6299	2	$1.37	$2.36	$3.73
15x15_DA	225	25	3.4409	1.5906	3	$2.52	$3.48	$6.00
10x30_DA	300	34	3.6142	2.1811	2	$2.94	$4.73	$7.67

Wiring costs for the 21 benchmarks, sorted in increasing order, showing the adjusted PCB dimensions (after adding room for shift registers (SR)), number of layers and resultant breakdown of costs (PCB and shift registers).

Table III reflects this observation, as prices tend to jump more significantly as each odd numbered layer is added, meaning that a new, physical dual-sided PCB layer must be added.

D. PCB Cost Results

Table IV presents cost results for each DMFB architecture in Table II assuming 1 mm electrode sizes and an orthogonal capacity of two, as discussed in the preceding subsection. Table IV reports the number of pins and the subsequent number of required shift registers (SR), along with the PCB dimensions, which include extra space allocated for shift registers. The PCB dimensions are fed directly into the online cost estimator. The reported cost of the PCB is estimated under

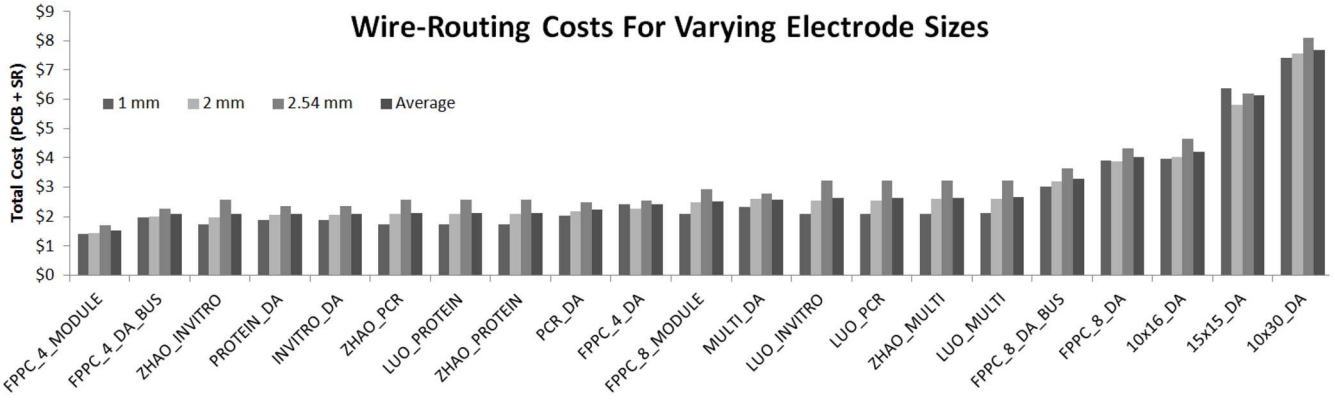

Fig. 24. Total wire-routing fabrication costs per DMFB board, including PCB manufacturing and shift register costs, for 21 DMFB designs utilizing 1, 2, and 2.54 mm electrodes. The benchmarks are sorted, from left to right, in order of increasing average cost.

the assumption of a 4-week delivery time (slowest) at a quantity of 2500 [1]; the shift registers cost \$0.139 apiece at a quantity of 2500 [5]. As described in (2), the total cost is that of the PCB plus shift registers.

In Table IV, the DMFB chips are sorted in increasing order of cost, and the results show that *FPPC_4_MODULE* is the cheapest design by at least \$0.18 per board, compared to assay-specific chips *ZHAO_XXX* and *LUO_XXX*. The *FPPC_4_DA_BUS* design, which can transport multiple droplets concurrently because the 3-phase buses are replaced by direct-addressing buses, is \$0.56 more expensive than its pin-optimized counterpart. Given that droplet transport times are orders of magnitudes faster than operation times [33], [38] and that the FPPC-DMFB with serial routing has proven competitive to pin-constrained assay-specific chips in terms of performance [11], we believe that *FPPC_4_MODULE* is the best overall solution when considering price and performance.

Table IV also shows that the PCB cost for many directly addressable DMFBs are less expensive than pin-constrained counterparts. For example, even though *PCR_DA* is 0.4 in longer than *LUO_PCR*, the PCB is \$0.47 cheaper because it requires fewer layers; however, the directly addressable version requires five shift registers, increasing the cost of the design to be \$0.23 more expensive than *LUO_PCR*.

This overall trend of shift-register costs is shown in the bottom half of Table IV; starting with *PROTEIN_DA* and looking downward, the number of pins, shift registers, and total cost are nondecreasing. Thus, it is important to minimize the pin-count; however, it must be balanced with a reduction in the number of PCB layers, as seen with our FPPC-DMFB, to achieve a comparably low-cost wire-routing solution.

Fig. 24 presents the final cost estimates for each of the 21 DMFBs with 1, 2, and 2.54 mm electrodes, using the parameters reported in Table III, and the shift register costs described previously. All estimates assumed for a 4-week delivery time and shipment quantity of 1000 because price estimates of 2500 could not be obtained online for all benchmarks. The benchmarks are sorted, from left to right, in order of increasing average cost. Although there is some variation when comparing the cost between different electrode sizes (e.g., from *MULTI_DA* to *LUO_INVITRO*, the 1 mm and 2 mm costs decrease, while the 2.54 mm cost increases), the overall trend of increasing price follows the average for each electrode pitch. This indicates that electrode pitch minimally impacts PCB cost since the manufacturing parameters can be adjusted to compensate the orthogonal capacity to reduce the number of layers.

For several of the benchmarks, the 1 mm boards are more expensive than their 2 mm counterparts. Recall that shift registers are stacked vertically and that additional width is added in the X-dimension to accommodate new columns, as necessary. Since the 2 mm instances provide a greater PCB height ($\text{height}_{\text{PCB}}$) than the 1 mm instances, the 2 mm boards sometimes require less additional space ($\text{width}_{\text{PCB\_SR}}$) to accommodate the extra shift registers, which indirectly offsets the initial area cost of using larger electrodes. This occurs, for example, for *FPPC_4_DA* and *FPPC_8_DA*. The extra cost for *15×15_DA* is due to an additional layer (3 versus 2 layers) for the smaller 1 mm case, due to its lower orthogonal capacity.

E. Routing Performance

The top portion of Table V reports the time spent performing operations and routing on nine common benchmark assays [11], [13], [15], [33] for sequential (pin-optimized) and parallel (route-optimized) FPPC-DMFBs, with and without wash droplets; identical schedules were used for both chips. The width of the chip is increased to 11 electrodes, to make room for the two wash I/O reservoirs. All but one benchmark was performed with the standard 16-electrode height, as shown in Fig. 7, allocating four mix modules, five SSD modules, and one routing buffer; the *ProteinSplit3* benchmark requires a larger 11 × 18 array with an extra SSD module. The results show that the parallel routing version (without wash droplets) yields a modest 1–10% savings (up to 4.29 s on *ProteinSplit3*); however, when wash droplets are introduced, parallel routing on the route-optimized FPPC-DMFB reduces the overall completion time by 6–28% (up to nearly 18 s on *ProteinSplit3*).

Using the preferred metrics for 2 mm electrodes in Table III, the price for an 11 × 16 pin-optimized (sequential) and route-optimized (parallel) FPPC-DMFB are \$1.44 and \$2.00, respectively. Thus, as expected, the modest gains in performance for the route-optimized FPPC-DMFB come at the expense of cost. Ultimately, the end user should decide which

TABLE V
ASSAY SCHEDULING/ROUTING RESULTS (SECONDS) FOR DIRECT ADDRESSING DMFB AND FPPC SEQUENTIAL/PARALLEL ROUTING (FPPC)

			Assay Benchmarks								
DMFB	Metric		PCR	InVitro1	InVitro2	InVitro3	InVitro4	InVitro5	ProteinSplit1	ProteinSplit2	ProteinSplit3
FPPC	DMFB Size		11×16	11×16	11×16	11×16	11×16	11×16	11×16	11×16	11×18
	Schedule Length (s)		11	14	18	19	23	29	52	68	123
	No Wash	Sequential Routing (s)	1.52	2.32	3.31	4.89	6.44	8.66	2.17	5.05	10.82
		Parallel Routing (s)	0.89	1.53	2.12	2.81	3.73	5.04	1.48	3.24	6.53
		Overall Assay Savings	**5%**	**6%**	**9%**	**9%**	**10%**	**1%**	**2%**	**3%**	**5%**
	Wash	Sequential Routing (s)	3.88	5.88	8.53	12.66	16.68	22.42	5.99	13.53	29.74
		Parallel Routing (s)	2.17	2.07	3.08	4.85	5.88	8.07	2.28	5.63	11.83
		Overall Assay Savings	**19%**	**21%**	**25%**	**27%**	**28%**	**6%**	**10%**	**12%**	**11%**
Direct Addressing [14]	DMFB Size		15×11	15×11	15×11	15×11	15×11	15×11	15×11	15×11	15×11
	Schedule Length (s)		11	14	20	24	30	44	54	102	164
	No Wash	Parallel Routing (s)	0.44	0.64	1.12	1.4	1.95	2.43	1.22	2.73	5.28
		FPPC Route Savings	**-102%**	**-139%**	**-89%**	**-101%**	**-91%**	**-107%**	**-21%**	**-19%**	**-24%**
		FPPC Overall Savings	**-4%**	**-6%**	**5%**	**14%**	**16%**	**27%**	**3%**	**32%**	**23%**

Performance results showing schedule/route length in seconds. The top half shows parallel/sequential routing times with/without wash droplets and the savings gained (overall assay completion time) by using parallel routing in the FPPC-DMFB. The bottom gives results for a state-of-the art direct addressing synthesis flow (no wash droplets) and details the savings obtained (overall assay completion time) when using the parallel FPPC-DMFB (no wash droplets).

tradeoff to take when considering their desired application and whether cost or performance is more important.

The bottom portion of Table V gives results for a state-of-the-art direct addressing synthesis flow [13]. We use a similarly sized array (11 × 16 versus 15 × 11), although the direct addressing device is wider since it is a better fit for its virtual topology. The virtual topology configuration of the direct addressing model dictates that a 15 × 11 DMFB can allocate four general-purpose modules, which can process mixes, splits, or two simultaneous storage operations [13]. The results show that the direct addressing design is able to obtain better routing results because of its flexibility and extra electrodes; however, although the route saving percentages are high, the actual route savings do not grow larger than 2.5 s. In contrast, as seen from the schedule lengths, the FPPC-DMFB is able to produce significantly better schedules (savings almost 40 s with *ProteinSplit3*). Thus, the overall savings show that the FPPC-DMFB produces superior performance results as the assay size increases. Finally, using the preferred metrics for 2 mm electrodes in Table III, the price for a 15 × 11 direct addressing DMFB is $4.22, which is $2.22 to $2.78 more expensive than the 11×16 FPPC-DMFBs that perform the same assays.

VIII. CONCLUSION

This paper has extended the initial development of FPPC-DMFBs [11] with an enhanced design to facilitate more efficient wire routing, and has presented the first cost estimates, in terms of U.S. dollars, for PCB fabrication for DMFBs. A complete synthesis flow, which addresses architectural issues that are specific to the FPPC-DMFB, has been presented, along with a detailed description of its general-purpose, as opposed to assay-specific, capabilities.

Our prior work has shown that the general-purpose FPPC-DMFB is competitive with direct-addressing and prior pin-constrained assay-specific DMFBs in terms of performance [11]; this paper goes one step further by showing that the FPPC-DMFB is less expensive than previous pin-constrained designs, when optimized for cost and PCB design. Thus, the flexibility provided by the FPPC-DMFB is unmatched by prior pin-constrained DMFBs, which were optimized for specific assays, and offers a significant advancement in terms of programmability at a lower overall per-unit cost.

ACKNOWLEDGMENT

Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect those of the NSF.

REFERENCES

[1] (2014, May 1). *Advanced Circuits.* [Online]. Available: http://www.4pcb.com
[2] (2014, May 1). *ATmega1284.* [Online]. Available: http://www.atmel.com/devices/atmega1284.aspx
[3] (2014, May 1). *Fairchild Semiconductor.* [Online]. Available: http://www.fairchildsemi.com
[4] (2014, May 1). *UCR Microfluidics Laboratory.* [Online]. Available: http://microfluidics.cs.ucr.edu
[5] (2014, May 1). *Mouser Electronics.* [Online]. Available: http://www.mouser.com
[6] J.-W. Chang, S.-H. Yeh, T.-W. Huang, and T.-Y. Ho, "An ILP-based routing algorithm for pin-constrained EWOD chips with obstacle avoidance," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 11, pp. 1655–1667, Nov. 2013.
[7] J.-W. Chang, S.-H. Yeh, T.-W. Huang, and T.-Y. Ho, "Integrated fluidic-chip co-design methodology for digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 2, pp. 216–227, Feb. 2013.
[8] S. Chatterjee, H. Rahaman, and T. Samanta, "Multi-objective algorithm for efficient pin-constrained droplet routing technique in digital microfluidic biochip," in *Proc. 14th Int. Symp. Qual. Electron. Design (ISQED)*, Santa Clara, CA, USA, Mar. 2013, pp. 252–256.
[9] M. Dhindsa, S. Kuiper, and J. Heikenfeld, "Reliable and low-voltage electrowetting on thin parylene films," *Thin Solid Films*, vol. 519, no. 10, pp. 3346–3351, Mar. 2011.
[10] E. Griffith and S. Akella, "Performance characterization of a reconfigurable planar-array digital microfluidic system," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 2, pp. 345–357, Feb. 2006.
[11] D. Grissom and P. Brisk, "A field-programmable pin-constrained digital microfluidic biochip," in *Proc. 50th ACM/EDAC/IEEE Design Autom. Conf. (DAC)*, Austin, TX, USA, May/Jun. 2013, pp. 1–9.

[12] D. Grissom and P. Brisk, "A high-performance online assay interpreter for digital microfluidic biochips," in *Proc. Great Lakes Symp. VLSI*, Salt Lake City, UT, USA, May 2012, pp. 103–106.
[13] D. 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.
[14] D. Grissom and P. Brisk, "Fast online synthesis of generally programmable digital microfluidic biochips," in *Proc. 8th IEEE/ACM/IFIP Int. Conf. Hardw./Softw. Codesign Syst. Synth.*, Tampere, Finland, Oct. 2012, pp. 413–422.
[15] D. Grissom and P. Brisk, "Path scheduling on digital microfluidic biochips," in *Proc. 49th Annu. Design Autom. Conf.*, San Francisco, CA, USA, Jun. 2012, pp. 26–35.
[16] D. Grissom *et al.*, "A digital microfluidic biochip synthesis framework," in *Proc. IEEE/IFIP 20th Int. Conf. VLSI Syst. Chip (VLSI-SoC)*, Santa Cruz, CA, USA, Oct. 2012, pp. 177–182.
[17] T.-W. Huang and T.-Y. Ho, "A two-stage integer linear programming-based droplet routing algorithm for pin-constrained digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 2, pp. 215–228, Feb. 2011.
[18] T.-W. Huang, T.-Y. Ho, and K. Chakrabarty, "Reliabliity-oriented broadcast electrode-addressing for pin-constrained digital microfluidic biochips," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, San Jose, CA, USA, Nov. 2012, pp. 448–455.
[19] T.-W. Huang, H.-Y. Su, and T.-Y. Ho, "Progressive network-flow based power-aware broadcast addressing for pin-constrained digital microfluidic biochips," in *Proc. 48th ACM/EDAC/IEEE Design Autom. Conf. (DAC)*, San Diego, CA, USA, Jun. 2011, pp. 741–746.
[20] 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.
[21] F. J. Kurdahi and A. C. Parker, "REAL: A program for register allocation," in *Proc. 24th Conf. Design Autom.*, Miami, FL, USA, 1987, pp. 210–215.
[22] C. C.-Y. Lin and Y.-W. Chang, "Cross-contamination aware design methodology for pin-constrained digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 6, pp. 817–828, Jun. 2006.
[23] C. C.-Y. Lin and Y.-W. Chang, "ILP-based pin-count aware design methodology for microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 9, pp. 1315–1327, Sep. 2010.
[24] Y. Luo and K. Chakrabarty, "Design of pin-constrained general-purpose digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 9, pp. 1307–1320, Sep. 2013.
[25] Q. Ma, T. Yan, and M. D. F. Wong, "A negotiated congestion based router for simultaneous escape routing," in *Proc. 11th Int. Symp. Qual. Electron. Design (ISQED)*, San Jose, CA, USA, Mar. 2010, pp. 606–610.
[26] H. Moon, S. K. Cho, R. L. Garrell, and C-J. Kim, "Low voltage electrowetting on dielectric," *J. Appl. Phys.*, vol. 92, no. 7, pp. 4080–4087, Sep. 2002.
[27] R. Mukherjee *et al.*, "A heuristic method for co-optimization of pin assignment and droplet routing in digital microfluidic biochip," in *Proc. 25th Int. Conf. VLSI Design*, Hyderabad, India, Jan. 2012, pp. 227–232.
[28] J. H. Noh, J. Noh, E. Kreit, J. Heikenfeld, and P. D. Rack, "Toward active-matrix lab-on-a-chip: Programmable electrofluidic control enabled by arrayed oxide thin film transistors," *Lab Chip*, vol. 12, no. 2, pp. 353–360, Dec. 2011.
[29] K. O'Neal, D. Grissom, and P. Brisk, "Force-directed list scheduling for digital microfluidic biochips," in *Proc. IEEE/IFIP 20th Int. Conf. VLSI Syst. Chip (VLSI-SoC)*, Santa Cruz, CA, USA, Oct. 2012, pp. 7–11.
[30] M. G. Pollack, A. D. Shenderov, and R. B. Fair, "Electrowetting-based actuation of droplets for integrated microfluidics," *Lab Chip*, vol. 2, no. 2, pp. 96–101, 2002.
[31] S. Roy, D. Mitra, B. B. Bhattacharya, and K. Chakrabarty, "Congestion-aware layout design for high-throughput digital microfluidic biochips," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 8, no. 3, pp. 17.1–17.23, Aug. 2012.
[32] V. Srinivasan, V. Pamula, and R. Fair, "An integrated digital microfluidic lab-on-a-chip for clinical diagnostics on human physiological fluids," *Lab Chip*, vol. 4, no. 4, pp. 310–315, 2004.
[33] F. Su and K. Chakrabarty, "High-level synthesis of digital microfluidic biochips," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 3, no. 4, pp. 16.1–16.32, Jan. 2008.
[34] F. Su, W. Hwang, and K. Chakrabarty, "Droplet routing in the synthesis of digital microfluidic biochips," in *Proc. Design Autom. Test Eur.*, Munich, Germany, Mar. 2006, pp. 323–328.
[35] 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. Syst.*, vol. 3, no. 3, pp. 14.1–14.23, Nov. 2007.
[36] T. Yan and M. D. F. Wong, "Correctly modeling the diagonal capacity in escape routing," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 2, pp. 285–293, Feb. 2012.
[37] S.-H. Yeh, J.-W. Chang, T.-W. Huang, and T.-Y. Ho, "Voltage-aware chip-level design for reliability-driven pin-constrained EWOD chips," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, San Jose, CA, USA, Nov. 2012, pp. 353–360.
[38] P.-H. Yuh, C.-L. Yang, and Y.-W. Chang, "BioRoute: A network-flow-based routing algorithm for the synthesis of digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 27, no. 11, pp. 1928–1941, Nov. 2008.
[39] Y. Zhao and K. Chakrabarty, "Simultaneous optimization of droplet routing and control-pin mapping to electrodes in digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 2, pp. 242–254, Feb. 2012.
[40] Y. Zhao, K. Chakrabarty, R. Sturmer, and V. K. Pamula, "Optimization techniques for the synchronization of concurrent fluidic operations in pin-constrained digital microfluidic biochips," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 6, pp. 1132–1145, Jun. 2012.
[41] Y. Zhao, T. Xu, and K. Chakrabarty, "Broadcast electrode-addressing and scheduling methods for pin-constrained digital microfluidic biochips," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 7, pp. 986–999, Jul. 2011.

Daniel T. Grissom received the B.S. degree in computer engineering from the University of Cincinnati, Cincinnati, OH, USA, in 2008, the M.S. and the Ph.D. degrees in computer science from the University of California, Riverside, Riverside, CA, USA, in 2011 and 2014, respectively.

He is an Assistant Professor at Azusa Pacific University, Azusa, CA, USA. His current research interests include languages, synthesis and hardware interfacing for digital microfluidic biochips. He has published seven papers, three journals, and filed one provisional patent.

Mr. Grissom was the recipient of the NSF Graduate Research Fellowship Award in 2010.

Jeffrey McDaniel received the B.S. degree in computer science and pure mathematics and the M.S. degree in computer science from the University of California, Riverside (UCR), Riverside, CA, USA, in 2011 and 2014, respectively. He is currently pursuing the Ph.D. degree in computer science from UCR.

His current research interests include languages, synthesis, and hardware interfacing for continuous-flow microfluidic biochips. He has published four conference papers and one journal paper.

Philip Brisk (M'09) received the B.S., M.S., and the Ph.D. degrees, all in computer science, from the University of California, Los Angeles, Los Angeles, CA, USA, in 2002, 2003, and 2006, respectively.

From 2006 to 2009, he was a Post-Doctoral Scholar at the Processor Architecture Laboratory, School of Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. He is an Assistant Professor at the Department of Computer Science and Engineering, the University of California, Riverside, Riverside, CA, USA. His current research interests include FPGAs, compilers, and design automation and architecture for application-specific processors.

Dr. Brisk was the recipient of the Best Paper Award at CASES, 2007 and FPL 2009. He has been a Program Committee Member for several international conferences and workshops, including DAC, ASPDAC, DATE, VLSI-SoC, FPL, FPT, and so on. He has been a General (co-)Chair of the IEEE SIES 2009, IEEE SASP 2010, and the IWLS 2011, and participated in the organizing committee of several other conferences and symposia as well.