

# UC Irvine

## UC Irvine Electronic Theses and Dissertations

### Title

A Programmable Microfluidic Finite State Machine for the Autonomous Lab on a Chip

### Permalink

<https://escholarship.org/uc/item/3mx5b4wx>

### Author

Ahrar, Siavash

### Publication Date

2015

### Copyright Information

This work is made available under the terms of a Creative Commons Attribution-NonCommercial-ShareAlike License, available at <https://creativecommons.org/licenses/by-nc-sa/4.0/>

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA,  
IRVINE

A Programmable Microfluidic Finite State Machine  
for the Autonomous Lab on a Chip

DISSERTATION

Submitted in partial satisfaction of the requirements  
for the degree of

DOCTOR OF PHILOSOPHY

in Biomedical Engineering

by

Siavash Ahrar

Dissertation Committee :  
Associate Professor Elliot E. Hui, Chair  
Associate Professor James P. Brody  
Assistant Professor Wendy F. Liu

2015



## TABLE OF CONTENTS

	Page
Acknowledgments	iii
Funding Sources	v
List of Figures	vi
Curriculum Vitae	vii
Abstract of Dissertation	ix
CHAPTER 1: Computation, Control, and Microfluidic	
1.1 Microfluidics regulation with embedded controllers	1
1.2 Computing, a brief historical examination	3
1.3 Fluidics, an alternative to electrical computation and control	5
1.4 Microfluidic Control	6
CHAPTER 2: Microfluidic Finite State Machines	
2.1 Finite State Machine	9
2.2 Microfluidic Finite State Machines	13
2.3 Asynchronous Microfluidic Counters	20
CHAPTER 3: Programmable Microfluidic Finite State Machine	
3.1 Programmability and addressability	22
3.2 Programmable logic array	22
3.3 Programmable Finite State Machine	26
CHAPTER 4: Liquid networks	
4.1 Introduction	29
4.2 Serial dilution - microfluidic background	30
4.3 Microfluidic serial dilution ladder	32
4.4 Self contained rotary mixer	36
CHAPTER 5: Conclusions	39
REFERENCES	41

## ACKNOWLEDGMENTS

I owe a great deal of gratitude to many people who have mentored, guided, helped, and traveled with me throughout this past five years on this academic path. These past few years have been the best of years. This work would not have been possible without their kind friendship and support.

First, I have to acknowledge and express sincere gratitude to my mentor and advisor, Prof. Elliot Hui. Elliot has patiently guided me and provided me with numerous opportunities to learn, to grow and to become a stronger person and a better researcher. Grad school is a challenging and difficult process, but I have been fortunate to have had Elliot's kind and generous support throughout all these years.

I have to thank my family with their endless support, patience and kindness. My parents, Afsaneh and Soroush, and my brother Sam. This would not have been possible without you. Afsaneh and Soroush have always supported me throughout all my numerous adventures, from the time that I decided to become an artist (and a poet) to my transition towards science. They have only asked me to be true to myself and live a life unafraid of failure. You both are the best role models. I also have to thank my amazing brother Sam who has been kind and supportive - even answering my electrical engineering questions. Sam, while younger, has been more like an older brother to me. Thank you.

I am thankful and indebted to my committee members, Prof. Wendy Liu and Prof. James Brody, who have been supportive and generous with their time and thoughts throughout these past years. I have to also express gratitude towards Prof. Michael Green who have kindly mentored and supported me since I was an undergraduate student here at UC Irvine.

Finally, I have to thank all the members of the Hui lab. I consider myself fortunate to have been a small member of this fantastic, creative, and eclectic group of people. Particularly I have to thank Transon, and Philip Thomas who mentored me when I was a new undergraduate student. I have to express gratitude to Philip Duncan for all his help. Michelle for all her incredible efforts during the dilution ladder project, Irene for annotating numerous video files and Brian for all his help with the ELISA project. Kamran, David, Katrina, Brandon, Philip Chao, Mohamad, Tiffany, and Hyung have all provided me with their endless support, insight and their friendship. I am a better person thanks to all their friendship and it has been a joy and a privilege to have spent the past few years in their company.

*“We all complete.  
Maybe none of us really understand what  
we've lived through, or feel we've had enough time.”*

*--- Kazuo Ishiguro, Never Let Me Go 2005.*

## **FUNDING SOURCES**

This work was supported in part by the National Science Foundation (NSF) under grant no. ECCS 1102397 and the Defense Advanced Research Projects Agency (DARPA) N/MEMS S&T Fundamentals Program under grant no. N66001-1-4003 issued by the Space and Naval Warfare Systems Center Pacific (SPAWAR) to the Micro/nano Fluidics Fundamentals Focus (MF3) Center. S.A. was supported by a fellowship via the NSF LifeChips IGERT (#0549479).

## LIST OF FIGURES

1. Fig. 1.1	Finite state machine block diagram	3
2. Fig. 1.2	Addressable (serial) and embedded control of liquid networks	7
3. Fig. 2.1	Finite state machine	9
4. Fig 2.2	State transition diagram - traffic light example	10
5. Fig 2.3	Truth table-1 an example state encoding	11
6. Fig 2.4	Truth table-2 state encoding with rest option	12
7. Fig 2.5	Normally closed valve	13
8. Fig 2.6	State transition diagram - microfluidic assay	14
9. Fig 2.7	Microfluidic assay truth table-1	15
10. Fig 2.8	Microfluidic assay truth table-2	16
11. Fig 2.9	FSM-1 with D Latch	17
12. Fig 2.10	FSM-1 time trace	18
13. Fig 2.11	FSM-2 with D flip flop	19
14. Fig 2.12	FSM-3 with input	19
15. Fig 2.13	T flip flop	20
16. Fig 2.14	6-bit counter	21
17. Fig 3.1	Programmable logic array	23
18. Fig 3.2	PLA Next state - 1	24
19. Fig 3.3	PLA Next state - 2	24
20. Fig 3.4	Visual truth tables side by side	25
21. Fig 3.5	Programmable finite state machine	26
22. Fig 3.6	FSM-1 with programmable circuit - branching code	27
23. Fig 3.7	FSM-2 with programmable circuit - count and hold	28
24. Fig 3.8	FSM-2 with programmable circuit - reset code	28
25. Fig 4.1	Dilution ladder topology	32
26. Fig 4.2	Dilution ladder - with valves on all rings	33
27. Fig 4.3	Dilution ladder - pumping pattern	33
28. Fig 4.4	Dilution ladder - with alternating valves	34
29. Fig 4.5	Dilution characterization	36
30. Fig 4.6	Self contained rotary mixer	37
31. Fig 4.7	Self contained sequential wash system	38

## CURRICULUM VITAE

2015	Ph.D.	Biomedical Engineering	University of California, Irvine
2012	M.S.	Biomedical Engineering	University of California, Irvine
2010	B.S.	Electrical Engineering Minor Biomedical	University of California, Irvine

### Honors & Award

Biomedical Engineering Teaching Assistant of the year 2013-2014.  
Award presented by Engineering Student Council (ESC).

Public Impact Fellow - honorable mention - 2013.  
LifeChips Fellow, an IGERT National Science Foundation at University of California,  
Irvine - Class of 2010 (2 years training grant).

UCI Undergraduate Research Opportunities Program Fellow academic year of  
2009-2010.

Smart Grant National Science and Mathematics 2009-2010.

University of California Regent Scholarship 2007-2008, & 2008-2009.

### Peer reviewed publications

[P1] P.N. Duncan, **S. Ahrar**, and E.E. Hui, "Scaling of Pneumatic Digital Logic Circuits."  
*Lab on a Chip*, doi:10.1039/C4LC01048E.

[P2] **S. Ahrar**, M. Hwang, P.N. Duncan, E.E. Hui, "Microfluidic Serial Dilution Ladder."  
*Analyst*, doi:10.1039/C3AN01710A.

[P3] **S. Ahrar** \*, T.V. Nguyen \*, Y. Shi, T. Ikrar, X. Xu, and E.E. Hui, "Optical  
Stimulation and Imaging of Functional Brain Circuitry in a Segmented Laminar Flow  
Chamber.", *Lab on a Chip*, doi:10.1039/C2LC40689F. (\* equal contributions)

[P4] T.V. Nguyen, P.N. Duncan, **S. Ahrar**, and E.E. Hui, "Semi-Autonomous Liquid Handling via On-Chip Pneumatic Digital Logic.", *Lab on a Chip*, doi:10.1039/C2LC40466D.

### Conference proceedings

[C1] **S. Ahrar**, P.N. Duncan, and E.E. Hui, "Programmable Microfluidic Digital Logic for the Autonomous Lab on a Chip" *the 18th International Conference on Miniaturized systems for Chemistry and Life Sciences ( $\mu$ TAS)*, in press.

[C2] T.V. Nguyen \*, **S. Ahrar** \*, P.N. Duncan, and E.E. Hui, "Microfluidic Finite State Machine for Autonomous Control of Integrated Fluid Networks", *the 15th International Conference on Miniaturized systems for Chemistry and Life Sciences ( $\mu$ TAS)*, 2011, pp741-743.

[C3] **S. Ahrar** \*, T.V. Nguyen \*, Y. Shi, T. Ikrar, P.V.Thomas, X. Xu, and E.E. Hui, "Optical Stimulation and Imaging of Functional Brain Circuitry in a Segmented Laminar Flow Chamber", *the 15th International Conference on Miniaturized systems for Chemistry and Life Sciences ( $\mu$ TAS)*, 2011, pp. 873-875. (\* equal contributions).

### Correspondence

Letter to the editors advocating for the use of Microfluidics - Lab on Chip - technologies in comics was published in **Batman Detective Comics** ® issue 879 Sep 2011.

# **ABSTRACT OF THE DISSERTATION**

A Programmable Microfluidic Finite State Machine for the Autonomous Lab on a Chip

By

Siavash Ahrar

Doctor of Philosophy in Biomedical Engineering

University of California, Irvine, 2015

Associate Professor Elliot E. Hui, Chair

Microfluidics is an adaptation of semiconductor technology to the creation of circuits of gas and liquid. They have enabled the establishment of mechanical and liquid circuits with complexity similar to integrated electrical circuits. However, while the microfluidic chips have been miniaturized, their external governing systems have remained unchanged. This lack of embedded control transforms the small chip into a large and often cumbersome system.

Next generation of microfluidic systems will allow reduction or removal of these controllers. Various successful microfluidics for reduction of these external requirements has been demonstrated. However, an autonomous, self-contained, programmable microfluidic finite state machines (FSM) that only requires power to operate has remained absent.

In this work, we present sequential logic circuits implemented in microfluidics rather than electronics for the autonomous control of liquid networks. We demonstrate microcontrollers, simple pneumatic computers, built entirely out of microfluidic parts. We also demonstrate a programmable FSM, first programmable microfluidic computer, built out of pneumatic Boolean logic gates and channels.

We show a 6-bit asynchronous pneumatic counters, useful as an embedded timing reference. Added to the controllers we create liquid systems such as a 7 stage 1:1 serial diluter system, i.e. serial dilution ladder. Finally, we integrate liquid networks with these controllers to create self-contained microfluidic systems.

# Chapter 1

## Computation, Control, and Microfluidic

### 1.1. Microfluidics regulation with embedded controllers

Microfluidics is an adaptation of miniaturization and micro-technologies to circuits of liquid and gas [1-3]. Numerous advances in microfluidics have provided previously impossible opportunities in the fields: of biology [4], chemistry [5], medicine [6], and global health [7,8] by enabling precision manipulation of liquids that were otherwise impossible. Miniaturized manipulations provided by microfluidics include metering [9,10], mixing [11,12], trapping [13], filtering [6], separation[14,15], dilution [16], encapsulation [17] and reaction[18]. Automation of these operations through microfluidic large-scale integration (mLSI) under the control of computers, pressure tubes, and their associated interfaces has proven to be a robust and scalable approach specifically in laboratories with microfluidic expertise [19-22]. The best example for the application of the mLSI is perhaps in the area of single cell profiling, where the enabled science through this technology is of great interest [23].

To operate these systems, traditionally a small mLSI chip, the same size of a credit card, is connected to a mesh of tubes to solenoids and manifolds. Solenoids are in turn controlled by electrical computers [24]. This requirement for the external controllers, with often large size, cost, and tedious interface with the chip itself have precluded the application of mLSI technologies to the clinical point of care application. This is of particular interest since application of microfluidics to the global point of care detection of diseases, i.e. contagious epidemics, can one day revolutionize health care, specifically in the resource-limited regions

[8, 25]. Thus, a reliable, robust, portable, and affordable mLSI applied to the point of care can act as an ideal system. A possible solution to this challenge is via replacing the external controllers with embedded controllers made out of microfluidic parts themselves [24, 26]. With this approach, the complex external regulators are replaced by embedded, monolithic controllers that would render the entire mLSI systems self-contained. An added advantage is the potential cost reduction due to the monolithic manufacture of the entire system. However, ultimately paper microfluidic systems could prove to be the most cost effective approach in point of care detection of diseases [27].

To create a self-contained, self-regulating microfluidic system first Boolean logic gates are required. These gates can then be combined to make combinatorial [26] or synchronized sequential circuits. In other words, to automate microfluidics a simple computer, i.e. a Finite state machine (FSM), that can store different steps of an assay and encodes transitions between them is required. Microvalves much like electrical transistors are integrated and combined with pneumatic channels to create finite state machines alongside the fluid handling systems such as pumps and liquid networks. Figure 1.1.1 describes the block diagram representation of this approach. Through this integration, the complexity of the regulators can seamlessly be removed and transferred from the user and world chip interface to the chip itself. This transfer is much like the transition from large computers, electrical (ENIAC) [28] with their convoluted connections to integrated electronic circuits such as Intel® 4004 microprocessor [29]. To examine the importance of this transition further a brief examination of the history of early computers can be helpful.

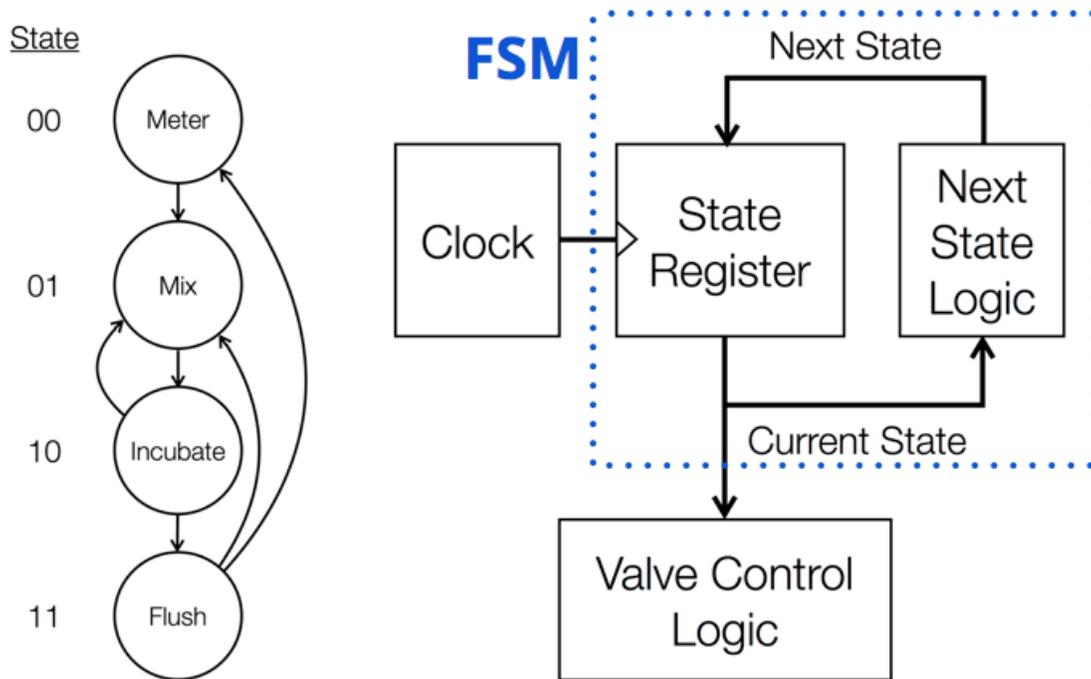


Fig. 1.1. **Finite state machine block diagram.** A state transition sequence can be encoded with an embedded controller such as a finite state machine (FSM). FSM is composed of state registers and a next state logic. Liquid networks can be regulated via valve control logic that is instructed by the FSM.

## 1.2. Computing, a brief historical examination

Charles Babbage's designs for the unfinished mechanical Analytical Engine has shaped the foundation of modern computing. Analytical Engine was the generalization of Babbage's earlier Difference Engine, and it included both memory and a processor [30]. Inspired by Joseph Jacquard's mechanical loom, which used sets of hole-punched cards to weave various patterns, Babbage proposed using read only punch cards to store programs. Punch cards were also to be used to interface the computer for data input and output. In fact, a program to compute Bernoulli numbers using Analytical Engine was proposed to Babbage by Lady Ada Lovelace [30, 31]. However, physical manufacture and realization of the entire Analytical Engine remained elusive

and impractical. These challenges with reliable manufacturability and the physical realization would represent a historic barrier for many computers.

By late 1920s, major breakthroughs in practical computing were realized through analog computers e.g. Vannevar Bush differential analyzer, which could solve sixth order differential equations [31]. This computer was both mechanical and electrical. It contained wheels, disks, gears, relay and vacuum tubes[31, 32]. A breakthrough in modern computing was in fact realized by Claude Shannon while working on the differential analyzer. Claude Shannon developed the foundation of digital design by implementing Boolean operations out of arrangements of relay circuits, i.e. electrical switches [33]. The major insight in Shannon's work was perhaps the realization of the importance of switches for the formulation of all combinatorial circuits. Concepts such as: digital sequential circuits, universal machines, and programmable computers, and the breakdown of various units inside a computer were then soon to follow through the works of Turing, Zues, Atanasoff, Von Neumann, and others[31,34]. However, major problems with the operation of early computers due to their interface and connections precluded their transfer application to everyday life.

The majority of these breakthroughs were accomplished in computers that used electrical signals to transfer and store information, and actuate switches. Miniaturization and integration of electrical switches, from relays to vacuum tubes to ever smaller transistors inside integrated circuits, has paved the way for the success of modern electrical computers [31, 35]. Recently scientific reports regarding the establishment of carbon nanotube-based computers and nanotube finite state machines suggest the continuation of this trend and possibilities of even smaller and more integrated electrical computers [36, 37]. This reliable miniaturized manufacture has been

the key to advancing electrical computers. Today the electrical nature of computers are taken for granted; however, fluidics circuits both analog and digital offered an alternative to control and manipulate fluidic and hydraulic systems.

### **1.3. Fluidics, an alternative to electrical computation and control**

Fluidic circuits were once (1950-1970s) considered as an alternative to electrical computers for the autonomous regulation of physical systems [38, 39]. Many Fluidic elements: oscillators, combinatorial logic gates, flip flops and even integrated sequential systems a card based fluidic controller for precision positioning of a radial drill were successfully demonstrated [38,40].

Both pure fluidic system and fluidic systems with moving parts were developed. Fluidic amplifiers that required turbulent flow were the most successful. For example in one system, turbulence amplifier, a transition between the laminar to turbulent flow constituted the transition from an output signal from ON to OFF. These systems could operate with pressures as low as 0.5 psi and switch states as fast as 2 msec. Other variations such as turbulent reattachment amplifier could switch faster than 1 msec; however they came at the expense of higher pressure requirements, e.g. 1 to near 15 psi. Various logic gates and Counters out of pure fluidics and moving parts such as gears were created. Of particular interest was the work at the University of Birmingham in precision control of a drill with all pneumatic machine tools that could be programmed with punch cards [38, 40]. However, miniaturization of semiconductors resulted in the ultimate success of electronics by addressing their reliable manufacture and tyranny of numbers problem.

#### 1.4. Microfluidic Control

Microfluidics have enabled miniaturization and integration of fluidic circuits [1]. However, miniaturization of embedded logic can provide the much needed autonomous control to microfluidics and the integration of the entire system[24]. Early microfluidic memory and control elements were directly inspired by the miniaturization of Fluidic circuits [41]. However, the lack of mixing and laminar flows associated with microfluidics here manifested as a challenge to the operation of these systems that requires a turbulent flow. Alternatively, some have taken advantage of capillary actions in microfluidic channels to create Boolean logic and timers [42]. Hydrogels have been also used as valving elements [43]. Computation and logic with Droplets are another active and successful area of investigation. Here a droplet can represent a physical bit of information. Early droplet logic where mostly combinatorial, some demultiplexing systems i.e. asynchronous shift registers were also demonstrated [44,45]. Recent droplet flip flops use external coordinating element to create synchronized droplet gates and systems [46].

As compared to other options creation of Boolean logic made out of microvalves controlled by gas or liquid, analogous to transistors, has offered the most intuitive, cascade-able and scalable solution[26]. Often valves or a series of valves are addressed by dedicated external pressure signals, Fig 1.2. The creation of combinatorial embedded logic e.g. 8-bit adder [47], addressable demultiplexing systems and latching memory [48], pneumatic flip flops and integrated system [49] and even a 12-bit pneumatic shift register has simplified the external control requirements [50]. Each of these efforts represents important advances towards fully self-contained microfluidic logic. However, they depend on accepting an external serial signal, o

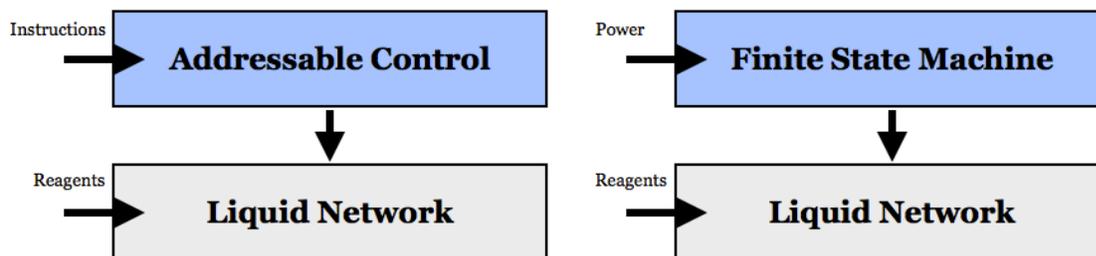


Fig. 1.2. **Addressable, serial, and embedded control of liquid networks.** Addressable systems accept instructions from external controllers and distribute them to the liquid networks. These external controllers suffer from the lack of miniaturization. An embedded controller, finite state machine, can autonomously encode state transitions independently from external controllers.

ften already coded, which is then redistributed in parallel through on-chip demultiplexers [24]. Thus, the electronic computer are required for signal coding and transfer.

A true stand-alone and programmable finite state machine can overcome the challenge of encodings states on the chip itself. Here we summarize major contributions provided by us and others prior to the establishment of finite state machines.

Through the application of their normally closed valves, Mathies laboratory first demonstrated microfluidic platforms that function under the direct instruction of external controllers such as pumps [51-53], and a serial dilution circuit [54]. Soon integrated on-chip systems, i.e. latching multiplexer [48] and combinatorial circuits such as ripple carry adder was demonstrated [47]. The combinatorial logic demonstration was in fact a major first step that hints at the feasibility of the establishment of embedded logic with Mathies and Grover valves.

Burns laboratory used a similar approach to that of Mathies normally closed valves and using a vacuum signal as a digital logic high, to create a great example of multiplexed embedded system. Burn microfluidics technology was entirely made out of PDMS. An on-chip oscillator,

gates, memory elements and finally shift registers were successfully demonstrated[49]. Through the configuration of these parts, Burns demonstrated the ability to accept an external serial trigger pulse, which is translated into a parallel operation of valves using the shift register approach. Unger and his group also expanded on this concept of shift register driven logic, but instead of vacuum they used positive pressure as their signal. They demonstrated a 12-bit shift register that can be addressed by external controllers that can translate trigger signals and patterns into on-chip valve control [50].

Weaver and company have also contributed the implementation of on-chip serial instructions. They have constructed a four layer PDMS system that uses a rigid disk to create gain that is required for valve actuation [55]. Finally, Takayama's group created a creative alternative method that uses hydraulic pressure and constant flow to create oscillators and temporal control of valves[56, 57]. Specifically by combining switching and check valves flow of liquids is restricted and released to create oscillation of pulses of liquids. This system seems to be particularly useful for embedded control of biological assays such as cell culture [58].

Our group have previously demonstrated tunable ring oscillators, timers, self-driving peristaltic pumps [59] and a semi-autonomous rotary mixing pump that required an off-chip controller for state selection[60]. Here we present a programmable finite state machine (FSM) built out of pneumatic boolean logic for the autonomous regulation of microfluidic liquid networks. Finite state machine (FSM) is a microcontroller built around the concept of stepping through a series of program states. The FSM stores the current state and calculates the next state from a set rules that can consider a user input plus the current state to enable branching decision making.

# Chapter 2

## Microfluidic Finite State Machine

Finite state machine (FSM) is a digital microcontroller built around the concept of stepping through a series of program states [61]. This simple synchronized computer encodes sets of states and the correct transitions between them. In the microfluidic context, these states are the steps of an assay, i.e. meter and mix. However, the architecture of an FSM is agnostic to the nature of the states. There is two essential compartment to a State machine these are state registers, or memory block of the circuit, and a next state combinatorial branch Fig. 2.1. The combinatorial branch is akin to a set of rules, a book of rules, that are used in combination with the current state of the machine to determine each following step. While not essential an input, from a user, can enable branching decision-making. Branched decision making is the ability to jump in between steps in a previously defined manner.

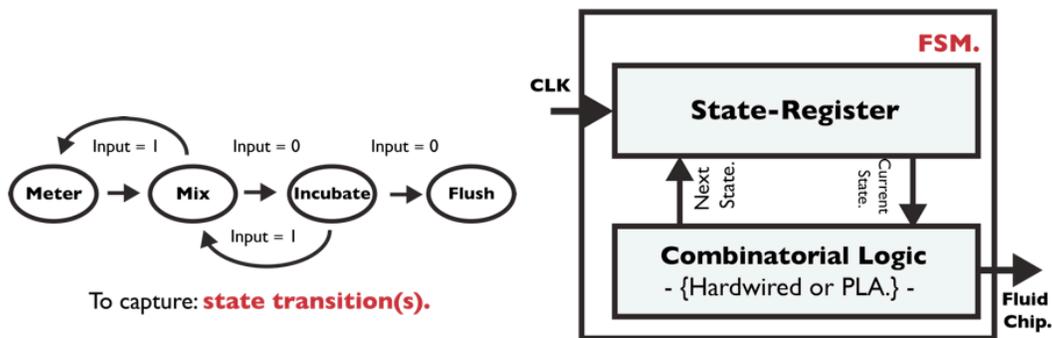


Fig. 2.1. **Finite state machine** is a simple computer that can encode a sequence of operations. An FSM uses its combinatorial branch and its stored states to calculate the transition in between a set of steps.

Additionally, the set of rules that are implemented in the combinatorial branch of the FSM can be either fixed, one machine a single sequence, or reconfigurable. The ability to program a machine was once described by Alan Turing as providing the machine with a new rulebook each time to execute a given task. Thus by enabling the exchange of the rules it is possible to render an FSM programmable. To better communicate the breakdown of an FSM control of a system, and design process for this control is next detailed.

A traffic light regulated by an FSM is a common example [61] that is used to illustrate the application of control elements. We first consider a traffic light that is independent of a user, e.g. pedestrian, input. Here there are 3 states that FSM controls:

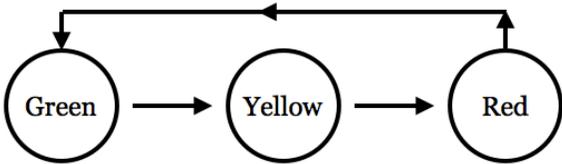


Fig. 2.2. **State Transition Diagram**, captures both the state and the proper transition between a set of state.

{Green Light, Yellow Light, Red Light}

The proper transition between these states is best described the flow diagram, Fig 2.2, known as state transition diagram. At each clock transition FSM will recall its current state, e.g. Green, and calculates where it should go next, e.g. Yellow. To use digital logic then we assign each state with a unique binary identifier. For example, Green light will be state 001, Yellow light state 010, and Red light state 100. While this allocation is not the most optimal state implementation, it serves well as an example. The first constraint arising from this allocation is the requirement for 3 individual memory elements for storing each bit of the states. Then the state are rearranged in a

table format commonly known as a truth table or transition table Fig. 2.2. Where the states S2, S1 and S0 current state of the FSM and states N2, N1 and N0 are the next states of the system.

Current States			Next States		
S2	S1	S0	N2	N1	N0
0	0	1	0	1	0
0	1	0	1	0	0
1	0	0	0	0	1

Fig. 2.3. **Truth table-1** is a re-arrangement of the state transition which is used to calculate the excitation equations. Excitation equations implement the next state logic in the FSM.

Then using Boolean arithmetic the table can be translated into a set of equations, where:

$$N0 = S2 \text{ not}(S1) \text{ not}(S0),$$

$$N1 = S0 \text{ not}(S2) \text{ not}(S1),$$

$$N2 = S1 \text{ not}(S2) \text{ not}(S0).$$

These equations are referred to as excitation equations, and they will shape the combinatorial branch of an FSM. Assuming that the inverse of the states, e.g. **not(S0)**, are readily available 3 x 3-input AND gates are required to implement the next state block of this circuit. We designate the number of switches - valves in microfluidics transistors in electronics- required in the next state block as the logical effort. Assuming 3 switches required for AND gates here logical effort is 9 switches. Minimizing logical effort can be important when implementation of a circuit is “expensive” due to constraints such as limited space.

Now we consider a case where a user input, e.g. a pedestrian attempting to cross the road, is added as a constraint. The state transition diagram is re-arranged such that when input from the sensor is activated the traffic light should switch to red, independent of the previous state. We have arranged this transition inside another truth table Fig 2.4.

Where the states **S2**, **S1** and **S0** current state of the FSM and states **N2**, **N1** and **N0** are the next states of the system.

<b>S2</b>	<b>S1</b>	<b>S0</b>	<b>Input</b>	<b>N2</b>	<b>N1</b>	<b>N0</b>
0	0	1	0	0	1	0
0	1	0	0	1	0	0
1	0	0	0	0	0	1
0	0	1	1	1	0	0
0	1	0	1	1	0	0
1	0	0	1	1	0	0

Fig 2.4. **Truth table-2** An input from the user can reset the FSM.

Then using Boolean arithmetic the table can be translated into a set of equations, where:

$$\mathbf{A = Input,}$$

$$\mathbf{N0 = S2 \text{ not}(S1) \text{ not}(S0),}$$

$$\mathbf{N1 = S0 \text{ not}(S2) \text{ not}(S1),}$$

$$\mathbf{N2 = S1 \text{ not}(S2) \text{ not}(S0)\text{not}(A) + A \text{ S0 \text{ not}(S1) \text{ not}(S2) + A \text{ S1 \text{ not}(S0) \text{ not}(S2) + A \text{ S2 \text{ not}(S0) \text{ not}(S1).}}$$

Assuming the inverse of the states are readily available, the circuit requires:

4 x 4-input **AND** gates,

1 x 4-Input **OR** gate.

Thus the logical effort for the new circuit is 20 switches, and the circuit complexity has increased. Later in this chapters, we examine a case where space efficient circuit designs can be achieved by minimizing logical efforts.

## 2.2 Microfluidic finite state machine

The next generation of microfluidic systems are expected to include self-regulating controllers. To this aim, we demonstrate the implementation of microfluidic FSM as a robust, scalable, approach Fig 2.5. We first present an FSM with no user input. Then to enable branched decision making we demonstrate an FSM with this option. Finally, we present an asynchronous counter, which can be used as a timing reference or a microfluidic controller.

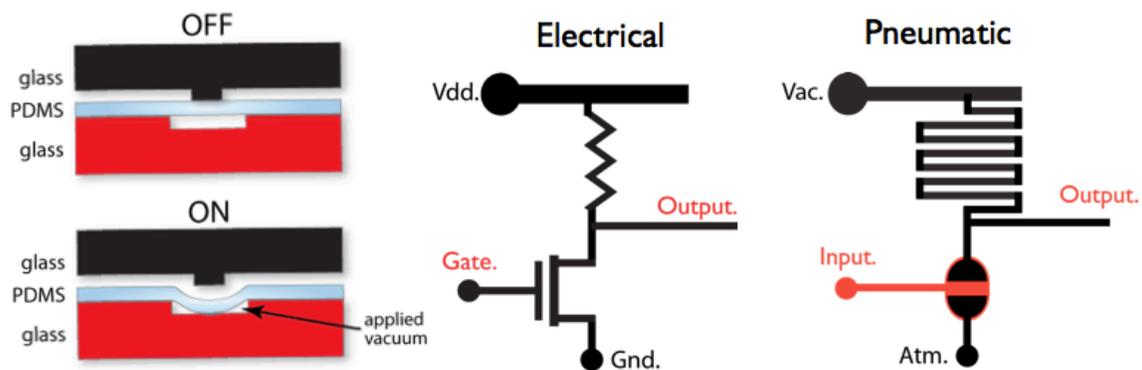


Fig. 2.5. **Normally closed valves** are the basic building block of pneumatic Boolean logic. Valves are normally closed. Vacuum applied to their gate input will deflect the membrane sandwiched between two glass layers and will open the valve. Valves can be arranged to create a pneumatic gates such as an inverter. A side by side NMOS electrical inverter with a pneumatic inverter is presented.

The basic building block of our microfluidic technology are normally closed valves pioneered by Grover, and Mathies lab [47,52]. These normally closed valves can be combined to create various gates and systems. These pneumatically actuated valves are closed at rest and can be opened by application of vacuum to their gate input. There is a great one to one analogy

between these valves and NMOS only transistors and their logic family Fig 2.5. We can replace transistors with valves, and wire and resistors with channels to establish Boolean logic.

Replacing electric potential, with a pneumatic potential we can establish pneumatic inverters Fig 2.5. We define vacuum as digital logic 1 or high and atmospheric pressure as digital logic 0 or low. Application of input 1 opens the valve and connects the atmospheric ground to the output while the long resistor channel shields the output from the vacuum supply. Application of logic 0 will keep the valve closed, which connects the power to the output of the inverter. We have demonstrated that valve transfer function exhibits good non-linear gain as oppose to hydraulic valves. Moreover, the valve provides good noise margins for safe transition between the open and closed states. By cascading an odd number of these inverters, i.e. typically 3, we have previously demonstrated tunable ring oscillators with oscillation frequencies spanning from 5-50 Hz [60]. Through the geometric arrangement of these valves, other gates such as NAND, NOR, AND, OR, and XOR can be built to create a logic complete family of logic[47,59,60]. These individual gates now can be integrated to create governing systems.

To create a self- regulating microfluidic system, we first identified the following steps as desirable steps in a microfluidic assay, these are:

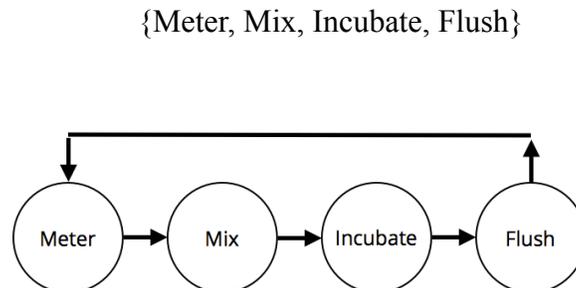


Fig. 2.6. **State Transition Diagram**, for a simple microfluidic assay with 4 states.

We first considered a case where the regulation of the system is independent of the user input and branching decision making is not possible. The state are then arranged inside the following state transition diagram Fig 2.6, where transitions between steps occur after each clock transition.

Since there are four required steps in the state transition diagram a 2-bits of memory is sufficient for the FSM implementation. Next we have to assign numeric (binary) state allocators to each one of the assay steps. Since there are 4 states, there are  $4! = 24$  unique arrangements possible.

Perhaps the most intuitive state assignment is:

{1. Meter (00) , 2. Mix (01), 3. Incubate (10), 4. Flush (11)}

S1	S0	N1	N0
0	0	0	1
0	1	1	0
1	0	1	1
1	1	0	0

Fig. 2.7 **Microfluidic assay truth table-1** demonstrates an inefficient state encoding

We calculate the excitation equations:

$$N0 = \text{not}(S0),$$

$$N1 = S0 \text{ not}(S1) + S1 \text{ not}(S0).$$

Assuming the inverse of the states are not readily available, the circuit requires:

3 **NOT** gate,

2 x 2-input **AND** gates,

1 x 2-Input **OR** gate.

Resulting in the logical effort of 9 valves (switches) totals in the combinatorial branch.

Now we consider an alternative state encoding where only one bit of information changes during each transition. With this approach the assignment can be:

{1. Meter (00) , 2. Mix (10), 3. Incubate (11), 4. Flush (01)}

S1	S0	N1	N0
0	0	1	0
1	0	1	1
1	1	0	1
0	1	0	0

Fig. 2.8 **Microfluidic assay truth table-2** demonstrates an efficient state encoding.

We calculate the excitation equations:

$$N0 = S1,$$

$$N1 = \text{not}(S0).$$

Assuming the inverse of the states are not readily available, the circuit requires only one additional valve (switch) in its combinatorial branch. Thus, the second sequence due to its less expensive logical effort is a more attractive option for state encoding.

Our second consideration was in the architecture of the memory elements that store the elements. We first considered level sensitive latches, where writing and changing memory is accessible throughout the period that clock input is enabled. While the level sensitive systems are not robust and are sensitive to clock requirements the small circuit footprint offered an attractive first step.

Device fabrication in the glass is previously described in detail elsewhere [60]. Briefly, microfluidic channels and valves are patterned via photolithography and photomasks on the glass

substrates (Telic Co., Valencia, CA, USA). Then the wafers are developed and etched with concentrated 48% HF acid wet etching. Holes, i.e. access ports for pneumatic power or atmospheric grounds, are manually drilled in glass using diamond grinding point bits. Vias in the PDMS membrane are bored using blunt needles prior to the assembly while the membrane rest on one of the two glass plates. Then the two complementary parts are aligned and brought together by hand under a dissection scope. Device behavior and characterization are achieved by video recording and video analysis. Specifically light reflection off of individual valves in an operating circuit is recorded and analyzed using a script that plots pixel intensities of selected regions of intensities over the course of the video.

While successful as a rudimentary system as predicted the choice of level sensitive D Latches for

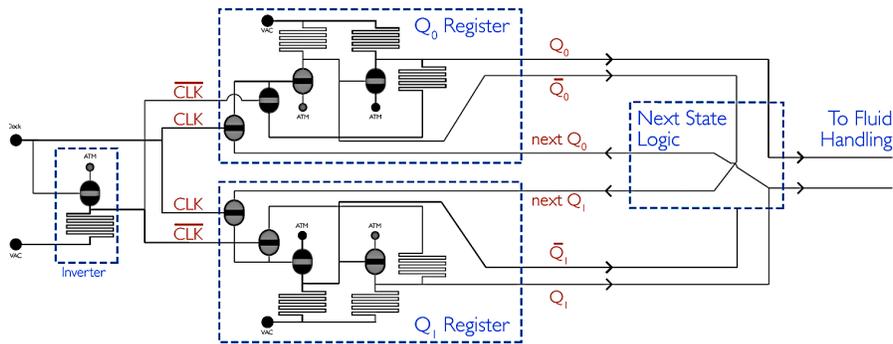


Fig. 2.9. **First microfluidic FSM with D latches.** A rudimentary FSM was established using level sensitive registers known as D latches.

the memory block rendered the FSM sensitive to the shape and duration of operating clock. Only open clock cycles as long as 0.1-0.15 seconds would enable correct operations Fig.2.10 This is however independent of the duration that the clock signal is left off.

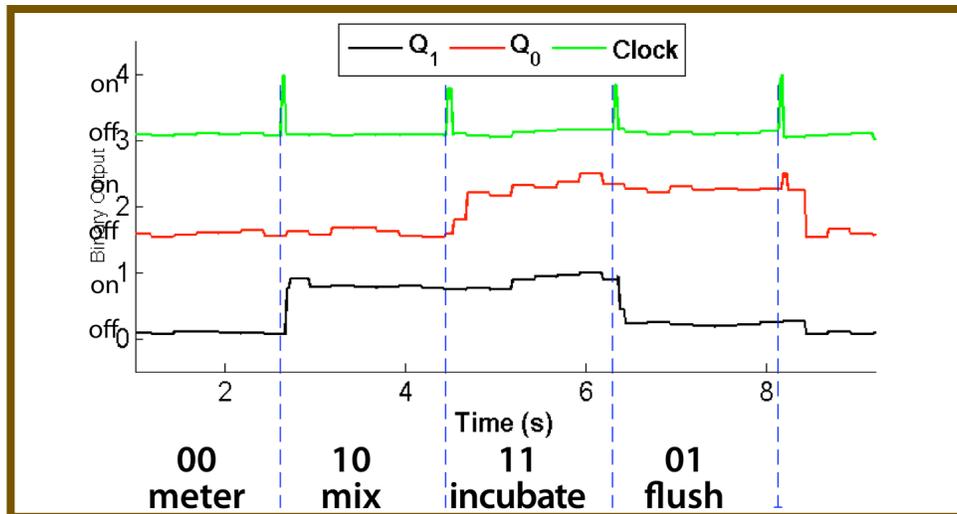


Fig. 2.10. **FSM time trace**, was measured using video and intensity analysis the recorded videos from the circuit operation. We measured and quantified light reflection from the state indicator valves opening and closing. Note that noise in the measurement is due to optical artifacts such as background light.

To overcome this limitation, the next generation of the FSM were designed with 2-stage master-servant D flip-flops, also known as master - slave, registers where memory allocation is only accessible during edges of clock transitions.

We then demonstrated the implementation of the proposed FSM using D flip flops, Fig 2.11 . The operation of the FSM were verified against various clock conditions, and as long as valve could be physically actuated the D flip-flops and the FSM were operational (0.05 seconds on, 0.05 seconds off). Since D flip-flops are stable memory elements the stored state are kept as long as the power to the circuit is present. While few version of the circuit was implemented, Fig. 2.11 , represent a space efficient and compact version of the circuit. In the following chapters we demonstrate the application of this circuit topology for the autonomous regulation of a rotary mixing liquid network.

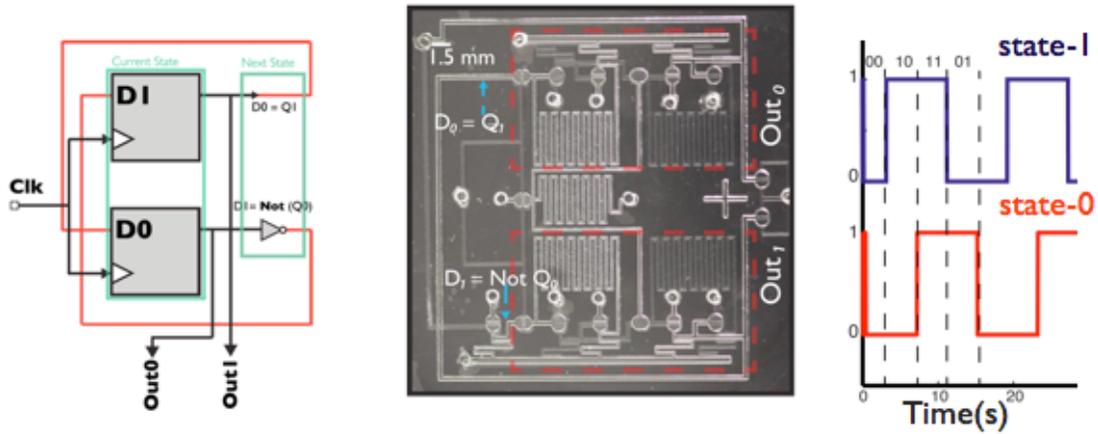


Fig. 2.11. **Finite state machine with D flip flops.** Block diagram and physical representation of an FSM using robust memory elements known as D flip flops. Time trace characterization of the circuit behavior

The ability to accept user input is also a desirable feature of the FSM architecture. This addition can enable branching decision making. To demonstrate the feasibility of addition of input to the system the following sequence was put together. To make detection of the states visually more appealing, we decided to divide the state transitions such that the states are either out of phase or in phase.

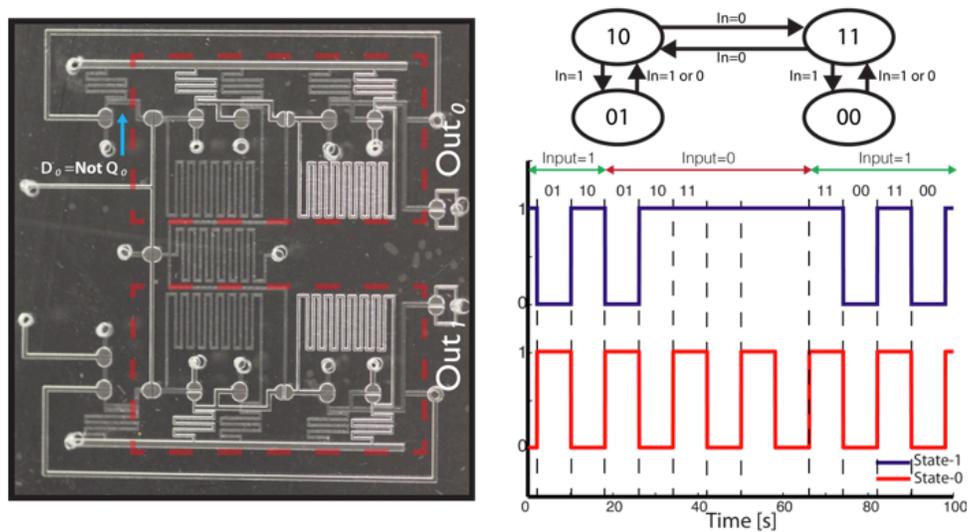


Fig. 2.12. **Finite state machine with input,** branching decision making is enabled by providing an input from a user.

Transitions between these two branches of the state transition could only happen under specific conditions. The device was fabricated, and its operation verified (Fig 2.12). We later demonstrate more sophisticated state transitions with user input in the programmable FSM chapter (chapter 3).

### 2.3. Asynchronous microfluidic counters

Synchronized systems such as FSM are controlled by a governing clock that coordinates their operations. Their design provides them with stability and robustness to noise, and race conditions. However, asynchronous sequential circuits, e.g. up counters constructed out of T flip flops are an attractive alternative particularly when used as timing elements, Fig.2.13. Oscillators can serve as clock signals, but a counter is required if keeping track of passage of time is needed. This for example could arise when a reaction needs to be timed on chip itself. To this aim any number of T flip flops can be cascaded and each time the input of the first bit is toggled one passage of time is added to the tally of oscillations in the system. It is note worthy to add that we created T flip flops from existing D flip flops by providing the out put of the circuit as an input for itself. So each time the clock is activated the stored bit of informations is toggled to its opposite state.

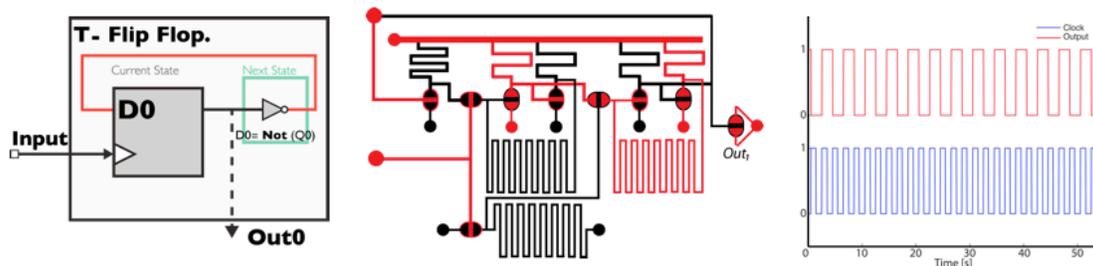


Fig. 2.13. **T. flip flops** are the building blocks of counters. A block diagram and gate level diagram of a T. flip flop. Time trace of a single T. flip flop is measured using video analysis

Here we demonstrate a 6-bit counter that can operate with input frequencies as fast as 1/3 Hz. this means 64 unique individual states can be uniquely identified and the transitions from the first state, i.e. 000000, to the last state, i.e. 111111 takes about 3 mins. Since T flip flops are static memory elements the states are held as long as power is provided to the system. 6-bits is the maximum number of bits that we have been able to place on a 4x4 inch on the glass wafer that holds both complimentary portions of the circuit. Using alternative manufacturing technologies particularly micro-machining of plastic substrate our lab has been able to create a 12-bit counter with an are of 360 mm<sup>2</sup>. The reduction in size also enables the system to operate at the maximum frequency of 6 Hz [62].

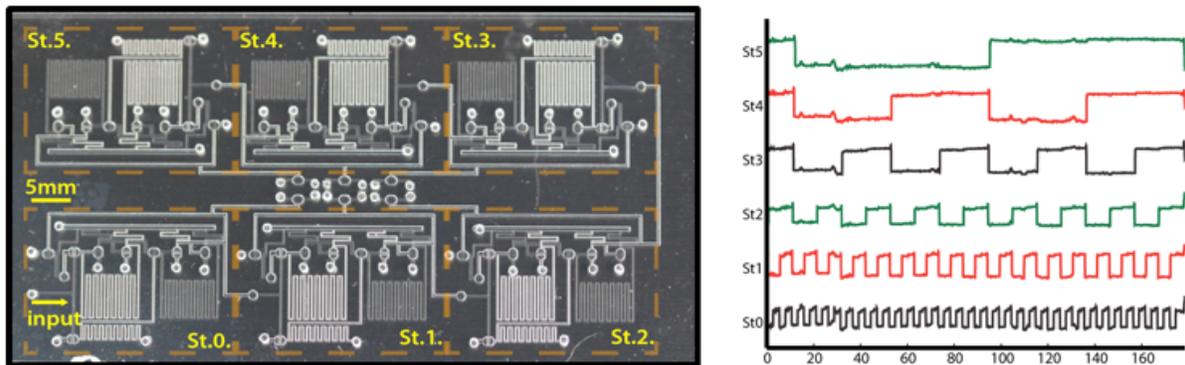


Fig. 2.14. **6-bit counter** constructed out of cascading T flip flops. A counter can enumerate passage of time, by tallying each time an input is presented. Time trace analysis of the circuit behavior, from state [000000] to the state [111111].

# Chapter 3

## Programmable microfluidic finite state machine

### 3.1. Programmability and addressability

Alan Turing once described programmability as the ability to provide a system with a set of rule to accomplish a certain task added with the ability of exchange of those rules [34]. Up to this point all the demonstrated FSMs had a single state transition diagram embedded in their next state block. Previously if a change in the state transitions was required a new circuit has to be designed, and fabricated. This is not desirable. However, it is possible to create a programmable FSM by replacing the next state block with a programmable circuit.

It is important to note that while previously many microfluidic devices have been called programmable, they have been rather addressable [63-65] and the programmability have been provided through electronic computers. These are powerful microfluidic platforms; however, they often require the electronic computers and their connections. Another class of microfluidic devices that are controlled by shift registers are among other addressable microfluidic devices. Here temporal signals are generated, i.e. programed, with a computer and then tallied into an addressable shift register[50]. This approach again depends on encoding of information through conventional electrical computers. Here we describe an embedded microfluidic system that can be programmed via re-configuration of the microfluidic circuit itself.

### 3.2 Programmable logic array

We decided to replace the next state block of the FSM with a Programmable Logic Array (PLA) that can be configured to capture a versatile range of state transitions. Similar to the field

programmable logic arrays (FPGA), a PLA is typically an array of AND gate banks followed by an array OR gate banks. In electronics, two wire networks could represent circuit nodes that can be connected to implement boolean equations, i.e. a sum of products combinations. Fig.3.1 represents both the [AND, OR] gate level implementation of the circuit. A variety of next state transitions can be encoded through the use of PLA. To implement we the microfluidic PLA we constructed a design entirely out of NAND gates.

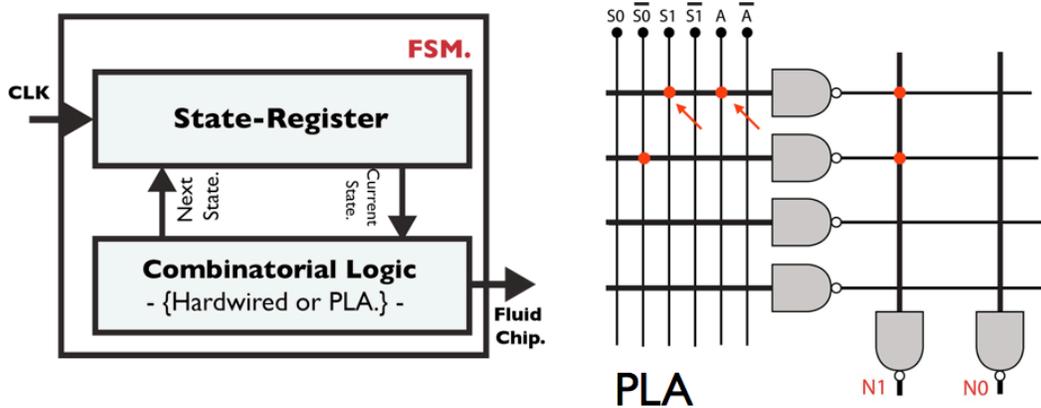


Fig. 3.1. **Programmable logic Array (PLA)** can replace the hardwired combinatorial branch of an FSM to render it programmable. To implement a new FSM.

To allow the programmability of the FSM state transitions, we first implemented a 6-input, 2-output programmable logic array. The circuit employs 6 NAND gates and is designed to be integrated into a 2 bit FSM that can also accept a bit of user input. PLA was programmed by boring via holes in the membrane either by manual punching or laser ablation. Here the channels on complementary Each pattern of holes enable a unique connection of nodes in the circuit, bringing etched glass plates together from opposite glass plates, and that encodes for state transitions. Fig 3.2, and Fig 3.3 represent example state transitions mapped into the PLA and its operation verified by the visual truth table.

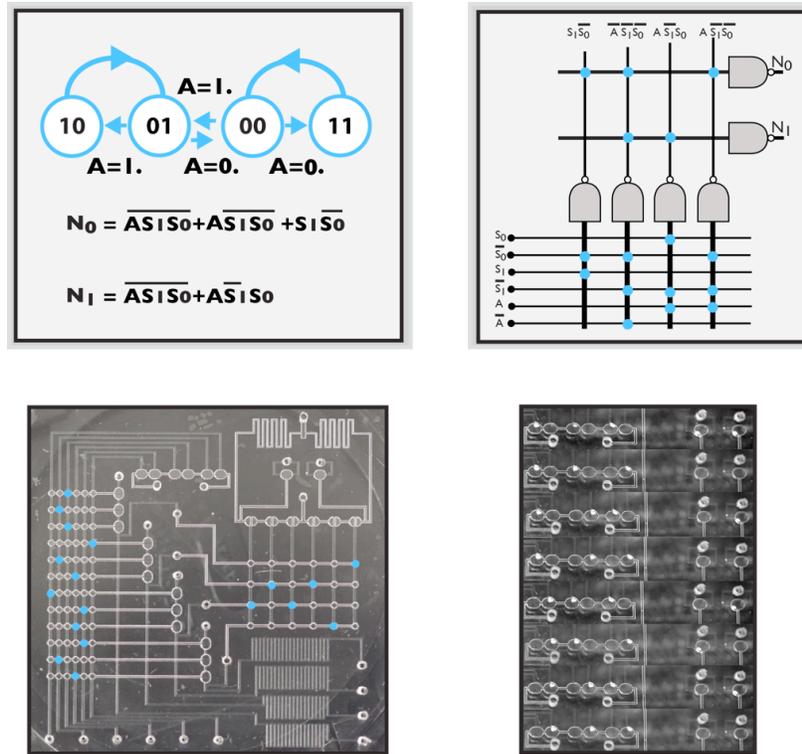


Fig. 3.2. **PLA-Next state-1** To program a PLA circuit first a state transition is designed. Then the excitation equations are calculated. Based on the equations the connections are mapped onto the gate level diagram of the PLA. Blue dots are the connections, bringing nodes of the circuits together. The gate level mapping is then translated into the actual circuit.

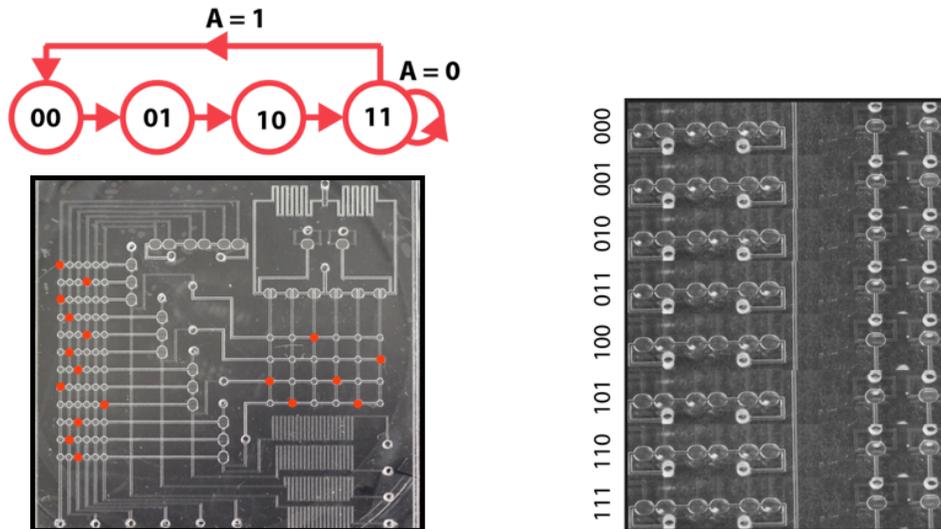


Fig. 3.3. **PLA-Next state-2** An alternative next state for a second FSM is implemented using the same chip only by changing the location of vias.

We first tested two unique state transitions in the stand alone PLA and verified the full 8-row truth table, given that there are 2 states and 1 user input, for each Boolean expression. Thus, we demonstrated that we can program two unique and correct transitions inside the PLA by boring the membrane. The PLA was then added to the state registers to enable sequential transition between these states. The approach of programming the microfluidic state machine with the bored membrane is similar to using punch cards in early computing.

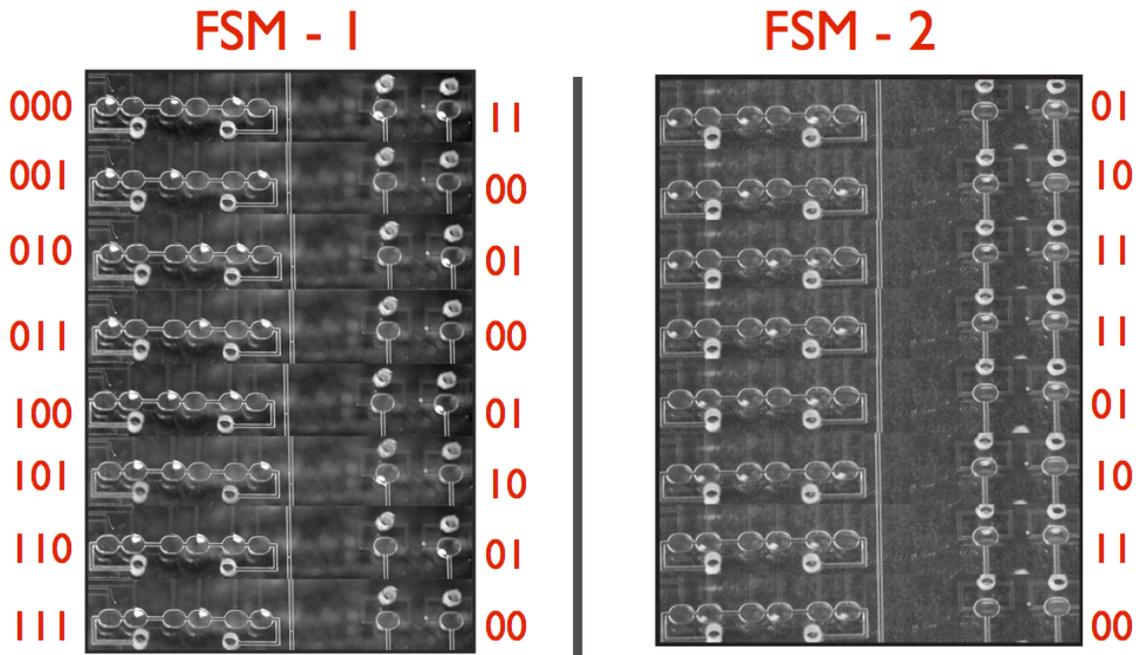


Fig. 3.4. Visual truth tables are presented side by side to demonstrate the programmability of the PLA.

### 3.3 Programmable Finite State Machine

To demonstrate the first programmable microfluidic computer, we then integrated the programmable logic array with two state registers, i.e. D flip flops, and a bit of user input. Here the user input is manually provided. To program the machine again vias were bored in proper locations connecting various channels that are equivalent of circuit nodes Fig 3.5.

First we implemented the simple state transition sequence with a lean logical effort. This sequence doesn't take advantage of user input, i.e. demonstrate branching decision making. However it was important to first establish the possibility of the operation of the device with small logical efforts implemented within the PLA Fig 3.5. The system's operation was verified by video recording. We confirmed that the programmable FSM can operate robustly with clock frequencies as fast as 5Hz.

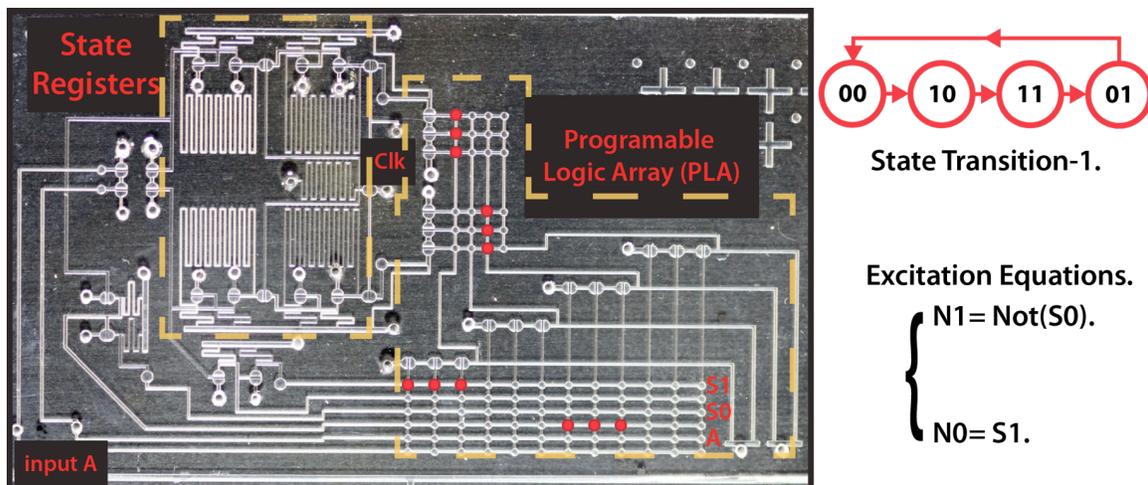


Fig. 3.5. **Programmable FSM** is implemented by integration of the PLA with D flip flop based state registers. State transition-1 is implemented inside the circuit by boring the membrane on red-dot. Using an external clock operation of the circuit was verified.

Then we designed 3 programs to demonstrate the wide range of State programs that can be implemented with the programmable FSM Fig 3.6-3.8 The first program well demonstrates the ability of branching decision making. The second program demonstrates the ability to hold a given state given a certain condition. Finally the third program demonstrates the ability to rest a sequence to its starting state each time the user provides an input.

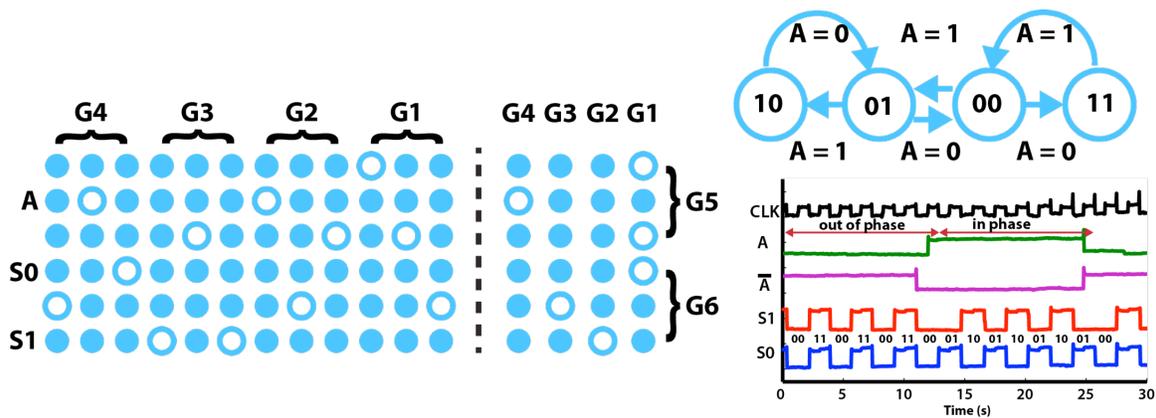


Fig. 3.5. **Programmable FSM - Branching code.** Using the programmable FSM the blue branching state transition diagram is implemented. The patterns of bores is represented in the matrix of circles where empty circles represent boring the membrane. Time trace from the circuit is calculated using video analysis and measuring valves opening and closing . This sequence best demonstrate the branching decision making ability of the PLA FSM.

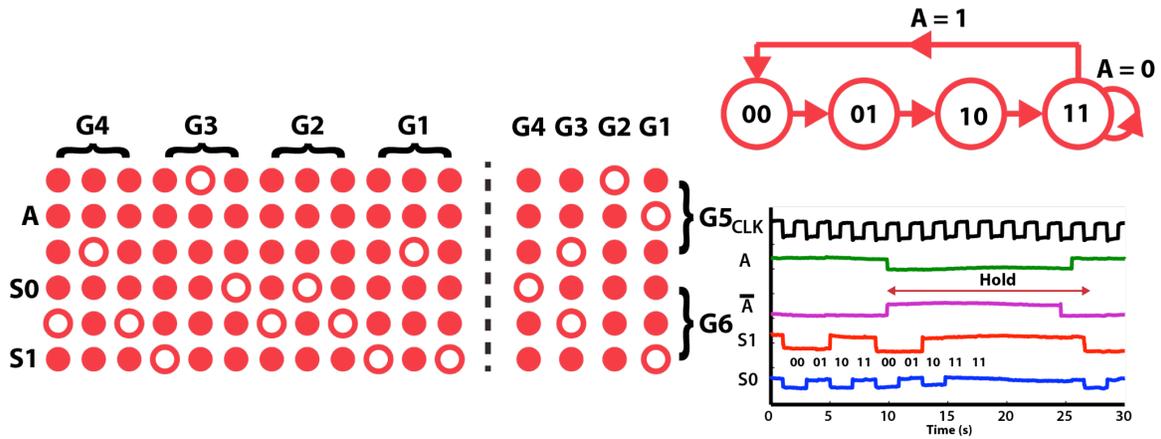


Fig. 3.6. **Programmable FSM - count and hold.** Red state transition diagram is implemented inside the FSM by boring membrane in correct spots on the PLA matrix. Time trace from the circuit is again calculated using video analysis.

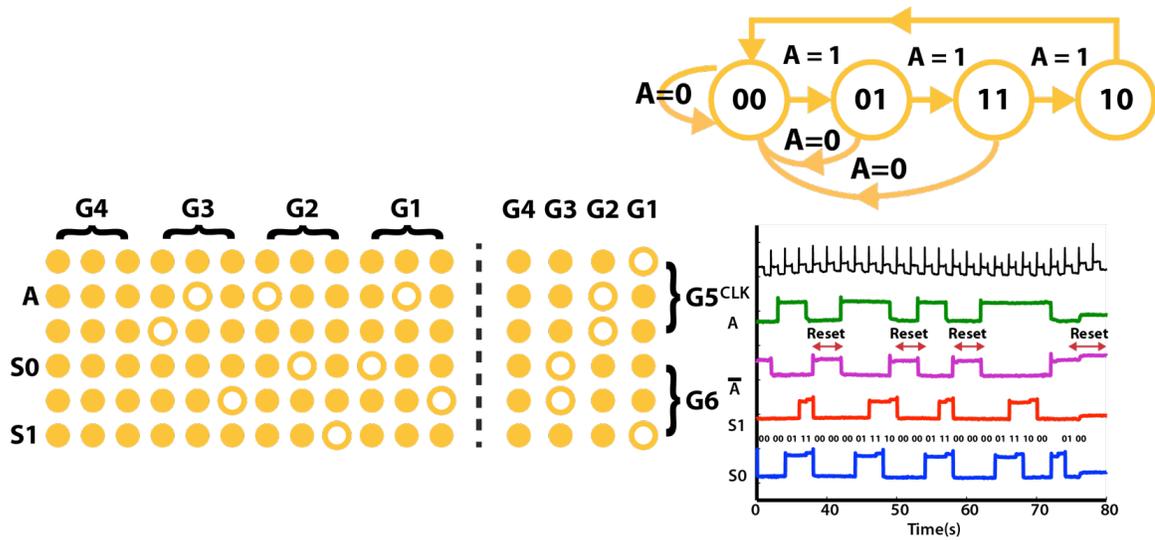


Fig. 3.7. **Programmable FSM - Reset code.** Above state transition diagram allows a user input to reset the next state of the FSM to state 00 (similar to the traffic light example from chapter 2). The correct connections are then identified and pattern of vias are mapped inside the PLA's connection matrix. A segment of the state transition diagrams is represented with the time trace.

# Chapter 4

## Liquid network

### 4.1. Introduction

Microfluidic large-scale integrated systems can enable complex and sophisticated on-chip control and manipulation of liquid networks [26]. Precision metering for assays such as protein detection via ELISA [66], mixing[67], dilutions [54] and reactions can be controlled in a repeatable, high throughput, and scalable fashion. However, the cumbersome external control elements and their interface with the microfluidic chip itself have often precluded their application outside of the laboratories with microfluidic expertise. This interface problem, i.e. the maze of tubing that address the liquid networks, can be compared to the early days of electronic computing and the famous tyranny of number problem [24, 26]. Taking inspiration from the electronic integration of control elements and their interconnects, application of embedded logic with microfluidics can address the proposed challenges.

Using this approach we have previously demonstrated two major contributions in liquid handling systems. These are: tunable self-driving peristaltic pumps governed by ring oscillators [60], and a semi-autonomous liquid handling systems [59]. Self driven peristaltic pumps is a major block of most of the integrated liquid network in our systems. Leveraging the highly frequency tunable ring oscillators efficient self driving pumps were demonstrated. We also have demonstrated that pumping patterns, added to tuning frequency or speed, can be adjusted by embedded pumps. The second contribution from our group is the semi-autonomous fluid handling system. This system combined oscillator-driven peristaltic pumps with simple

combinatorial logic to encode fluid handling operations with a state selector. Here 31 valves are coordinated together to accomplish mixing and dilution by using a self-driven rotary pump system [60]. While the instructions for this system are time-invariant constant signal, i.e. governing patterns are created on the chip itself. This system however still required an external control. In this chapter we first present a stand alone serial dilution system that is compatible with our microfluidic controllers. Then we demonstrate an entirely self-contained rotary mixer system that is governed only by a microfluidic FSM. Additionally this device uses a finger actuated button as a clock to advance in each program state.

#### **4.2. Serial dilution - microfluidic background**

A serial dilution is among the most common and fundamental laboratory processing. It is often used to create a logarithmic range of solution concentrations. Serial dilution can be used to create standards for diagnostics, a range of test solutions for chemical synthesis, or incredibly diluted working solutions for single cell studies or in-vitro directed evolution of molecules. In a typical lab setting the procedure requires a set of pipetting operations that must be performed with accuracy. While this process is tedious, it is straightforward for an experienced technician. However, it is beyond the capability of untrained users, and thus automation of serial dilution, specifically in point-of-care settings is an attractive alternative.

Some prior microfluidic systems for serial dilutions have been reported. These systems can be divided into three major categories: continuous flow approaches [16, 68], valved systems [54], and droplet-based dilution systems [69-71].

Continuous flow approaches often use a tiered network of fluidic resistors to mix 2 fluids at ratios established by the circuit designs. Continuous flow systems are a generalization of the

gradient generating microfluidic chips. These devices output an entire dilution series in parallel and are composed of simple channel networks and require no moving parts. These systems are well suited for continuous cell culture microfluidic systems for examples in organs on a chip. However, the need to sustain the continuous flow requires a relatively large initial sample volume, 5-100  $\mu\text{L}$ , and long channels that are needed to accomplish the fluid mixing [16, 68]. Additionally, the total fluidic resistance of this approach increases nonlinearly as the number of dilutions. Therefore the scaling potential is unclear.

A second approach uses integrated valves to measure and mix defined volumes of sample and buffer. Here the dilution ratio again is predefined, repeating the same pattern combined with partial sample recovery could further dilute a sample. For example, Paegel and colleagues [54] implemented such an approach based on peristaltic pumping around a single ring mixer. A small sample (400 nL) was diluted by a fixed ratio during each step, in which a volume of buffer was metered into the ring and circulated until the liquids were fully mixed. This recirculation eliminates the need for long mixing channels and thus results in a compact device (1  $\text{cm}^2$ ). Due to active sample retrieval there is no upper number of dilutions; however, this comes at the expense of lack of solution storage. Thus, only the final dilution was stored, and the intermediate steps were discarded.

Jensen et. al. also demonstrated a 8 x 8 [64,67] valve grid structure that is extensively capable of sample mixings and dilutions. This platform is extremely versatile and is capable of complex liquid manipulations. The device can, in fact, dilute a small working sample down to multiple stages of dilution and store the intermediate steps. However, it doesn't lend itself well to total automation scheme with microfluidic controllers such as FSM. These platforms also use

both positive and negative pressure for their operations which is not entirely compatible with our approach. Never the less this 8 x 8 addressable automata is a strong platform for sample preparations and dilutions.

Serial dilution through the application of droplets is another strong and alternative platform for serial dilution [70]. As with our method, these approaches are capable of processing small samples and can store each stage of dilution in an individual droplet. Some droplet systems have the advantage of simplicity and no moving parts; however, there are valved microfluidic systems. However, if serial dilution is not the end goal but the beginning of a multi-stage process, the required liquid-handling manipulation may be complex enough that valves become required. Additionally, droplet systems require off-chip pumps and regulators and particular liquids that add to the complexity of the total system. In our approach, the pumps are integrated into the platform itself.

### 4.3 Microfluidic serial dilution ladder

Our dilution ladder represents an evolution of the Paegel [54] dilution ring that is capable of storing every stage of dilution. As demonstrated in Fig 4.1 a series of chambers are arranged like



Fig. 4.1 **Dilution ladder topology** using ladder topology a sample can be 1:1 diluted and stored by connecting and mixing two subsequent steps of the ladder together

rungs in a ladder and primed with a buffer solution. A small sample volume approximately 240 nL is loaded into the first rung.

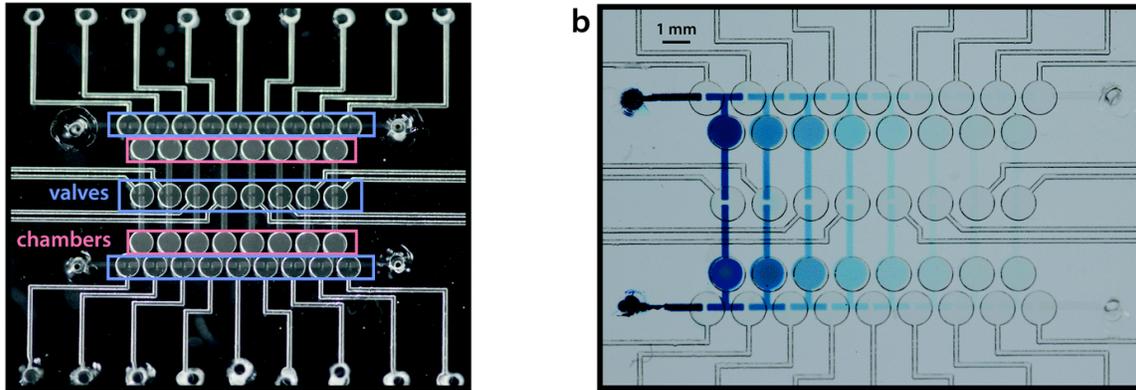


Fig. 4.2. **Dilution ladder** - with valves on all steps. Dilution ladder can dilute and store a sample here up to 7 stages in a 1:1 dilution system. The Chambers inside the ladder help with liquid movement during valves opening and closing. Device behavior is first demonstrated by dilution of food dye inside water.

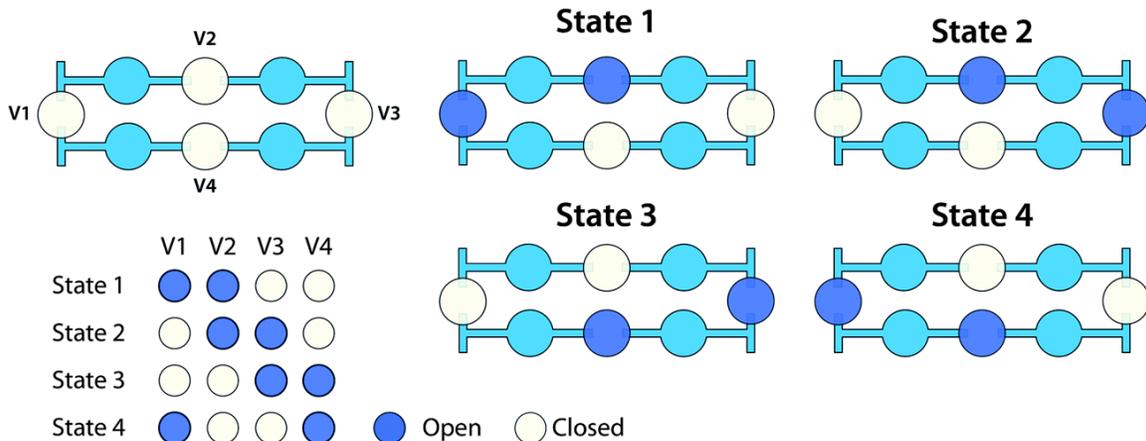


Fig. 4.3. **Dilution ladder pumping pattern** is demonstrated. The four pneumatic valves are actuated at a rate of 3 Hz, meaning the valves cycle through the four state three times every seconds. The liquid is circulated around the two steps, similar to a rotary pump, until they are mixed together.

To begin the dilution process, the top two rungs are connected to form a loop, and the liquid contents are mixed by peristaltic pumping to accomplish a 1:1 dilution. The second and third rung are then connected and mixed to perform a subsequent dilution, and the process is repeated

all the way down the ladder. When accomplished, the full dilution series is stored on the rungs of the ladder. The design is readily scalable, as addition stages can be added with extra rungs. We here demonstrate a 7-stage dilution in an active area of (1 cm<sup>2</sup>). We demonstrated two generations of these systems. First design uses mixing valves, valves at the center of each rung, which are spaced in every step of the ladder Fig.4.2. Second is the design that uses a mixing valve on every rung Fig 4.4. Valves from both systems were individually addressed with external controllers and solenoid valves under the control of LabView software (National Instruments). Fig 4.3 represents the pattern of actuation for the second generation dilution system. The four pneumatic valves are actuated at a rate of 3 Hz, meaning the valves cycle through the four state three times every seconds.

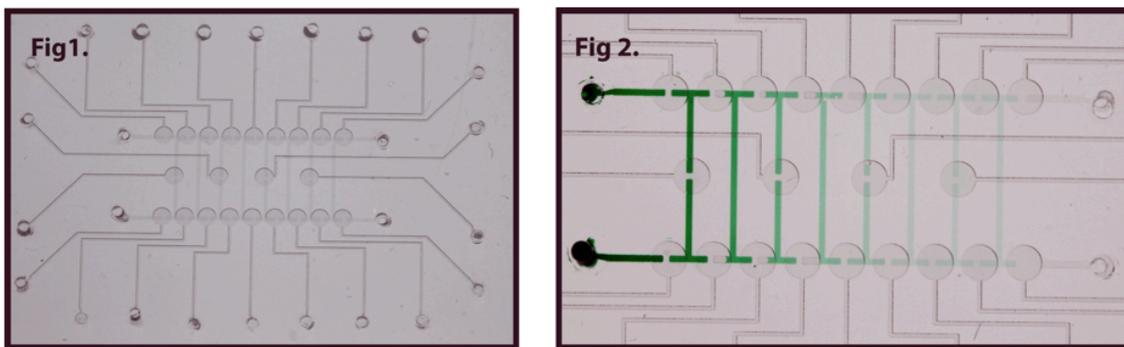


Fig. 4.4. **Dilution ladder with alternative valves.** Liquid movements and mixing in this design is harder than the design reported in figure 4.2. However this ladder requires fewer valves, i.e. states. Therefore, it is a better suited for integration with FSM controllers.

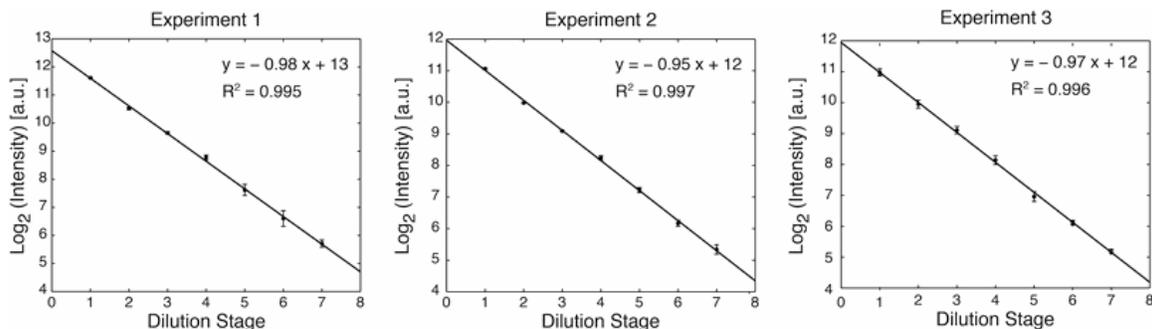
Briefly, the dilution ladder devices use a normally closed valve technology as previously reported. Valves and channels were patterned in a glass wafer (Telic Co., Valencia, CA, USA) with photolithography and 48% HF wet etching. The wafers were diced and drilled for ports fluid access through the glass with diamond-tipped grinding bits (McMaster-Carr). The typical 250  $\mu\text{m}$  thick silicone membrane (HT-6240, Rogers Corp.) was used to enable valve operation.

Adhesion between the glass and PDMS was sufficient to hold the membrane together. Two patterned glass layers were then sandwiched around a 250  $\mu\text{m}$  thick silicone membrane (HT-6240, Rogers Corp.) and held together reversibly by adhesion. Via ports in the membrane were formed with a biopsy punch prior to assembly. Valve actuation was accomplished by deflecting this layer pneumatically. For handling reagents that are not compatible with silicone exchange of alternative membranes: Teflon [72] or polyurethane[73] is a possibility.

We have added a set of fluid expansion chambers that help with better liquid movement in the ladder. The liquid volume is not conserved as the dilution ladder switch between the resting and pumping state. At rest, all four valves in a loop are closed, but during pumping, two valves are open at all times. To accommodate this change in volume, two chambers were added to each rung. Each chamber is similar in structure to a valve but has no pneumatic connection. In each chamber, a flexible silicone membrane separates a volume of liquid and a volume of air. As the liquid volume increases or decreases, the membrane can deflect to accommodate the change.

Food dye was first diluted into water to facilitate visualization of device operation. Next, to quantify dilution efficiency, fluorescein was diluted into TE buffer and imaged by fluorescence microscopy. The log<sub>2</sub> plot of fluorescence intensity for each dilution stage showed excellent linearity ( $R^2 = 0.995$  to  $0.997$  over 3 trials), demonstrating that the device can perform a 7-stage 1:1 serial dilution with high accuracy. Briefly 400  $\mu\text{M}$  fluorescein sodium salt (Sigma Aldrich) was diluted into TE buffer with 0.2% Triton-X at pH 8.3. Fluorescein concentrations were visualized with a 4 $\times$  objective on a Nikon Eclipse TE200 microscope with a QIClick CCD camera (QImaging). Image quantification was performed with ImageJ software (NIH). For each rung of the dilution ladder, the average and standard deviation of fluorescence intensity were

calculated from measurements of six regions distributed across the rung. Dilutions were plotted as the base-2 logarithm of fluorescence intensity for each stage, and linear regression quantified the accuracy of dilution.



**Fig. 4.5. Dilution characterization** Quantitative evaluation of serial dilution efficiency. Fluorescein was serially diluted into TE buffer and measured by fluorescence microscopy. The  $\log_2$  plot of intensity demonstrates good linearity over three independent experiments.

Currently the ladder is under integration with a FSM controller. While 1:1 dilutions is not always the desirable ratio outcome, the topology of the circuit is such that other ratios can be easily enabled. For example, to accomplish 1:10 ratio a pattern of 1:10:1 can be repeated to enable dilutions.

#### 4.4 Self contained rotary mixer

A major goal of our work has been autonomous regulation of microfluidic systems via FSM. In order to create a self-contained platform that only accepts power as an external supply. The idea here is that the governing patterns are generated locally with the embedded controller itself. Here we demonstrate two autonomous systems, first is a self-contained rotary pump mixer that is controlled by a 2-bit FSM with 4 acceptable states (Fig 4.7). The 2-bit FSM is the same circuit topology that was demonstrated in both chapter 2 and 3. The other is a 3-bit FSM with only 3 acceptable states for their operation.

Fundamentally both systems are similar in their governing principle were set of two valves enable access to a rotary pumps. The first system provides a mixing state, while the second design provides 3 sequential washes.

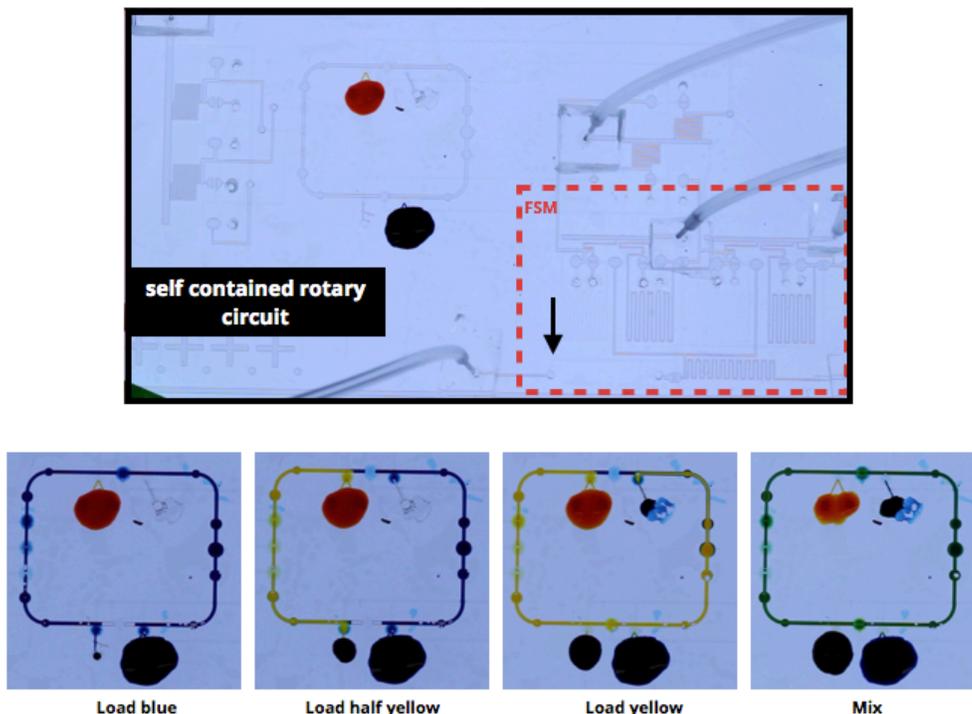


Fig. 4.7. **Self contained rotary mixer** is controlled by a 2-bit, 4 states, FSM. Program states advance each time when a user manually “clicks” (cover and release) a pneumatic clicker. 4 unique program states including a mix state is possible with this system.

A major component of both systems is a pneumatic clicker that connects a source of vacuum signal (cover and release) to the clock input of the machine and enable advancing to the next program state. The self-contained rotary circuit can meter 2 independent liquids. In one state it is possible to meter half or the entire chip with the desired reagent. The two metered liquids can be then mixed together via rotary pump circulation.

The sequential wash system is not much different from the rotary mixing circuit. Here however, only 3 states of a 3-bit FSM is used to regulate allocation of 3 different liquids inside a rotary mixing circuit.

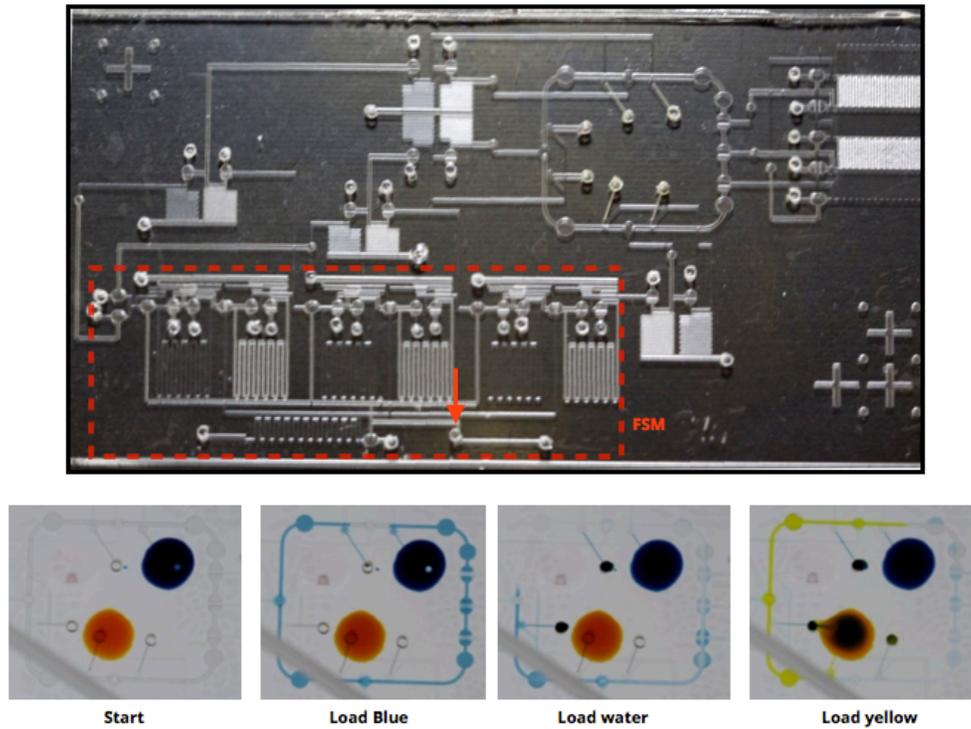


Fig. 4.7. **Self contained sequential wash system** is controlled by a 3-bit FSM with only 3 acceptable states. system sequential routes 3 independent fluids through the same liquid loop.

# Chapter 5

## Conclusions and future directions

Application of digital pneumatic circuits and mLSI for automation of biology and chemistry have gained popularity over the past decade. However, autonomous regulation of these systems have been absent. Here we demonstrated pneumatic microfluidic finite state machines, simple computers, to autonomously regulate these systems to reduce or remove the external controllers and tubing that is associated with these controllers.

We first demonstrated a simple and robust finite state machine (FSM) that steps through the series of program states. These states correspond to the steps of a microfluidic assay. We demonstrated that by adding a user input to the architecture of the FSM, we can enable branching decision making. An asynchronous counter, a timing reference or a controller, was also demonstrated.

Next we demonstrated programmability with re-wiring of the next state block of the FSM, which was replaced by a programmable logic array(PLA). First the stand alone NAND-NAND based PLA was established. Programmability was achieved by boring vias within the membrane inside the system. Then through the integration of the circuit with state-registers the programmable circuits was demonstrated. 4 unique and different FSMs were created with the single programmable chip. We demonstrated circuit operation with clock frequencies as fast as 5 Hz.

Finally we established liquid networks and new topologies that are compatible with our controllers. A 7 stage 1:1 dilution ladder was demonstrated. Automation and integration of this

system is currently ongoing. We then demonstrated an entirely self-contained rotary mixer. Using embedded pump and a 2-bit FSM we demonstrated autonomous regulation of this system. Currently, we work towards autonomous control and regulation of an ELISA assay with these controllers.

These embedded microfluidic controller can simplify and ease the need for external controllers associated with mLSI. They can enable the application of the mLSI outside of sophisticated laboratories, for example to point-of-care detection of diseases. While the glass micro fabrication of these technologies have enabled all the current advances in the microfluidic finite state machine and microfluidic computing, the technology does not scale well. Fortunately, we have demonstrated the alternative manufacture of this system [62] by micro-machining plastic substrates. This alternative manufacture enables both higher circuit densities and faster operation frequencies. For example, 6-bit glass counter could operate as fast 1/3 Hz, while the plastic system can operate as fast 6 Hz.

Advances in circuit architecture and circuit topology, such as the programmable FSM, coupled with the advances in manufacturing of these circuits strongly suggest the possibility translation of mLSI from scientific laboratories to for example global health diagnostic applications.

## References

- [1] Squires, T. M., Quake S.R., Microfluidics: Fluid physics at the nanoliter scale. *Reviews of Modern Physics*, 77(3), 977–1026, 2005.
- [2] Whitesides, G. M. The origins and the future of microfluidics. *Nature*, 442(7101), 368–373, 2006.
- [3] Sackmann, E. K., Fulton, A. L., & Beebe, D. J. The present and future role of microfluidics in biomedical research. *Nature*, 507(7491), 181-189, 2014.
- [4] Bhatia S.N. , Ingber D.E. Microfluidic organs-on-chips. *Nature Biotechnology*, 32:760–772, 2014.
- [5] S. Marre, K. F. Jensen, Synthesis of micro and nanostructures in microfluidic systems *Chemical Society Review*, 39, 1183, 2010.
- [6] Kang, Joo H., et al. An extracorporeal blood-cleansing device for sepsis therapy. *Nature medicine*, 20, 1211-1216, 2014.
- [7] Chin, C. D., Laksanasopin, T., Cheung, Y. K., Steinmiller, D., Linder, V., Parsa, H., Wang, J., et. al. Microfluidics-based diagnostics of infectious diseases in the developing world. *Nature Medicine*, 17(8),1015–1019, 2011.
- [8] Yager, P., Edwards, T., Fu, E., Helton, K., Nelson, K., Tam, M. R., & Weigl, B. H. Microfluidicdiagnostic technologies for global public health. *Nature*, 442(7101), 412–418, 2006.
- [9] Shen, F., Du, W., Kreutz, J. E., Fok, A., & Ismagilov, R. F. Digital PCR on a SlipChip. *Lab on a Chip*, 10(20), 2666-2672, 2010.
- [10] Pompano, R. R., Liu, W., Du, W., & Ismagilov, R. F. Microfluidics using spatially defined arrays of droplets in one, two, and three dimensions. *Annual Review of Analytical Chemistry*, 4, 59-81, 2011.
- [11] Lee, C. Y., Chang, C. L., Wang, Y. N., & Fu, L. M. Microfluidic mixing: a review. *International Journal of Molecular Sciences*, 12(5), 3263-3287 2011.
- [12] Chou, H. P., Unger, M. A., & Quake, S. R. (2001). A microfabricated rotary pump. *Biomedical Microdevices*, 3(4), 323-330, 2001.

- [13] Skelley, A. M., Kirak, O., Suh, H., Jaenisch, R., & Voldman, J. Microfluidic control of cell pairing and fusion. *Nature Methods*, 6(2), 147-152, 2009.
- [14] Gossett, D. R., Weaver, W. M., Mach, A. J., Hur, S. C., Tse, H. T. K., Lee, W., Di Carlo, D. Label-free cell separation and sorting in microfluidic systems. *Analytical and Bioanalytical Chemistry*, 397(8), 3249-3267.
- [15] Mach, A. J., & Di Carlo, D. Continuous scalable blood filtration device using inertial microfluidics. *Biotechnology and Bioengineering*, 107(2), 302-311, 2010.
- [16] Jiang, X., Ng, J. M., Stroock, A. D., Dertinger, S. K., & Whitesides, G. M. A miniaturized, parallel, serially diluted immunoassay for analyzing multiple antigens. *Journal of the American Chemical Society*, 125(18), 5294-5295 2003.
- [17] Klein, A. M., Mazutis, L., Akartuna, I., Tallapragada, N., Veres, A., Li, V., Weitz, D. A., Kirschner, M. W. Droplet Barcoding for Single-Cell Transcriptomics Applied to Embryonic Stem Cells. *Cell*, 161(5), 1187-1201, 2015.
- [18] Song, H., Chen, D. L., Ismagilov, R. F. Reactions in droplets in microfluidic channels. *Angewandte Chemie International Edition*, 45(44), 7336-7356, 2006.
- [19] Unger, M. A., Chou, H. P., Thorsen, T., Scherer, A., Quake, S. R. Monolithic microfabricated valves and pumps by multilayer soft lithography. *Science*, 288(5463), 113-116 2000.
- [20] Thorsen, T., Maerkl, S. J., & Quake, S. R. Microfluidic large-scale integration. *Science*, 298(5593), 580-584, 2002.
- [21] Balagaddé, F. K., You, L., Hansen, C. L., Arnold, F. H., Quake, S. R. Long-term monitoring of bacteria undergoing programmed population control in a microchemostat. *Science*, 309(5731), 137-140, 2005.
- [22] Melin, J., Quake, S. R. Microfluidic large-scale integration: the evolution of design rules for biological automation. *Annu. Rev. Biophys. Biomol. Struct.*, 36, 213-231, 2007.
- [23] Wu, A. R., Neff, N. F., Kalisky, T., Dalerba, P., Treutlein, B., Rothenberg, M. E., Clarke, M.F. Quake, S. R. Quantitative assessment of single-cell RNA-sequencing methods. *Nature Methods*, 11(1), 41-46, 2014.

- [24] Mosadegh, B., Bersano-Begey, T., Park, J. Y., Burns, M. A., & Takayama, S. Next generation integrated microfluidic circuits. *Lab on a Chip*, 11(17), 2813-2818, 2011.
- [25] Chin, C. D., Laksanasopin, T., Cheung, Y. K., Steinmiller, D., Linder, V., Parsa, H., et.al. , Sia, S. K. Microfluidics-based diagnostics of infectious diseases in the developing world. *Nature Medicine*, 17(8), 1015-1019, 2011.
- [26] Araci, I. E., & Brisk, P. Recent developments in microfluidic large scale integration. *Current Opinion in Biotechnology*, 25, 60-68, 2014.
- [27] Hu, J., Wang, S., Wang, L., Li, F., Pingguan-Murphy, B., Lu, T. J., Xu, F. Advances in paper-based point-of-care diagnostics. *Biosensors and Bioelectronics*, 54, 585-597, 2014.
- [28] Burks, A. W., Burks, A. R. First general-purpose electronic computer. *Annals of the History of Computing*, 3(4), 310-389, 1981.
- [29] Aspray, W. The Intel 4004 microprocessor: What constituted invention? *Annals of the History of Computing*, IEEE, 19(3), 4-15, (1997).
- [30] Graham-Cumming, J. John Graham Cumming: the greatest machine that never was (2012 Mar) Ted talk available from:[https://www.ted.com/talks/john\\_graham\\_cumming\\_the\\_greatest\\_machine\\_that\\_never\\_was?language=en](https://www.ted.com/talks/john_graham_cumming_the_greatest_machine_that_never_was?language=en)
- [31] O'Regan, Gerard. A brief history of computing. Springer Science & Business Media, 2008.
- [32] Bush, V. The differential analyzer. A new machine for solving differential equations. *Journal of the Franklin Institute*, 212(4), 447-488, 1931.
- [33] Shannon, C.: A symbolic analysis of relay and switching circuits. Masters Thesis, Massachusetts Institute of Technology, 1937.
- [34] Turing, A.: Computing, machinery and intelligence. *Mind*, 49, 433-460, 1950.
- [35] Kilby, J. S. Invention of the integrated circuit. *Electron Devices*, IEEE, 23(7), 648-654, 1976.
- [36] Shulaker, M. M., Hills, G., Patil, N., Wei, H., Chen, H. Y., Wong, H. S. P., & Mitra, S. Carbon nanotube computer. *Nature*, 501(7468), 526-530, 2013.

- [37] Yao, J., Yan, H., Das, S., Klemic, J. F., Ellenbogen, J. C., & Lieber, C. M. Nanowire nanocomputer as a finite-state machine. *Proceedings of the National Academy of Sciences*, 111 (7), 2431-2435, 2014.
- [38] Foster, K., & Parker, G. A. Fluidics: components and circuits. John Wiley & Sons, 1970.
- [39] Conway, A. (Ed.) Guide to Fluidics. Macdonald, 1971.
- [40] Foster, K., & Parker, G. A. The role of pure fluidic devices in machine tool applications. *Production Engineer*, 48(4), 169-180, 1969.
- [41] Groisman, A., Enzelberger, M., Quake, S. R. Microfluidic memory and control devices. *Science*, 300(5621), 955-958, 2003.
- [42] Toepke, M. W., Abhyankar, V. V., & Beebe, D. J. Microfluidic logic gates and timers. *Lab on a Chip*, 7(11), 1449-1453, 2007.
- [43] Eddington, D. T., & Beebe, D. J. Flow control with hydrogels. *Advanced Drug Delivery reviews*, 56(2), 199-210, 2004.
- [44] Prakash, M., & Gershenfeld, N. Microfluidic bubble logic. *Science*, 315(5813), 832-835, 2007.
- [45] Fuertman, M. J., Gartecki, P. & Whitesides, G. M. Coding/decoding and reversibility of droplet trains in microfluidic networks. *Science* 315, 828–832, 2007.
- [46] Katsikis, G., Cybulski, J. S., & Prakash, M. Synchronous universal droplet logic and control. *Nature Physics*, 2015.
- [47] Jensen, E. C., Grover, W. H., & Mathies, R. A. Micropneumatic digital logic structures for integrated microdevice computation and control. *Microelectromechanical Systems*, 16(6), 1378-1385, 2007.
- [48] Grover, W. H., Ivester, R. H., Jensen, E. C., & Mathies, R. A. Development and multiplexed control of latching pneumatic valves using microfluidic logical structures. *Lab on a Chip*, 6(5), 623-631, 2006.
- [49] Rhee, M., & Burns, M. A. Microfluidic pneumatic logic circuits and digital pneumatic microprocessors for integrated microfluidic systems. *Lab on a Chip*, 9(21), 3131-3143, 2009.

- [50] Devaraju, N. S. G. K., & Unger, M. A. Pressure driven digital logic in PDMS based microfluidic devices fabricated by multilayer soft lithography. *Lab on a Chip*, 12(22), 4809-4815, 2012.
- [51] Grover, W. H., Skelley, A. M., Liu, C. N., Lagally, E. T., & Mathies, R. A. Monolithic membrane valves and diaphragm pumps for practical large-scale integration into glass microfluidic devices. *Sensors and Actuators B: Chemical*, 89(3), 315-323, 2003.
- [52] Grover, W. H., & Mathies, R. A. An integrated microfluidic processor for single nucleotide polymorphism-based DNA computing. *Lab on a Chip*, 5(10), 1033-1040, 2005.
- [53] Kim, J., Kang, M., Jensen, E. C., & Mathies, R. A. Lifting gate polydimethylsiloxane microvalves and pumps for microfluidic control. *Analytical Chemistry*, 84 (4), 2067-2071, 2012.
- [54] Paegel, B. M., Grover, W. H., Skelley, A. M., Mathies, R. A., & Joyce, G. F. Microfluidic serial dilution circuit. *Analytical Chemistry*, 78(21), 7522-7527, 2006.
- [55] Weaver, J. A., Melin, J., Stark, D., Quake, S. R., & Horowitz, M. A. Static control logic for microfluidic devices using pressure-gain valves. *Nature Physics*, 6(3), 218–223, 2010.
- [56] Mosadegh, B., Kuo, C.-H., Tung, Y.-C., Torisawa, Y.-S., Bersano-Begey, T., Tavana, H., Takayama, S. Integrated elastomeric components for autonomous regulation of sequential and oscillatory flowswitching in microfluidic devices. *Nature Physics*, 6(5), 1–5. 2010.
- [57] Kim, S. J., Yokokawa, R., & Takayama, S. Microfluidic oscillators with widely tunable periods. *Lab on a Chip*, 13(8), 1644-1648, 2013.
- [58] Mosadegh, B., Agarwal, M., Tavana, H., Bersano-Begey, T., Torisawa, Y. S., Morell, M., ... & Takayama, S. Uniform cell seeding and generation of overlapping gradient profiles in a multiplexed microchamber device with normally-closed valves. *Lab on a Chip*, 10(21), 2959-2964, 2010.
- [59] Nguyen, T. V., Duncan, P. N., Ahrar, S., & Hui, E. E. Semi-autonomous liquid handling via on-chip pneumatic digital logic. *Lab on a Chip*, 12(20), 3991-3994, 2012.

- [60] Duncan, P. N., Nguyen, T. V., & Hui, E. E. Pneumatic oscillator circuits for timing and control of integrated microfluidics. *Proceedings of the National Academy of Sciences*, 110 (45), 18104-18109, 2013.
- [61] Vahid F., Digital design Publisher Hoboken, N.J.: J Wiley & Sons 2007.
- [62] Duncan, P. N., Ahrar, S., & Hui, E. E. Scaling of pneumatic digital logic circuits. *Lab on a Chip*, 15(5), 1360-1365, 2015.
- [63] Fidalgo, L. M., & Maerkl, S. J. A software-programmable microfluidic device for automated biology. *Lab on a Chip*, 11(9), 1612-1619, 2011.
- [64] Jensen, E. C., Stockton, A. M., Chiesl, T. N., Kim, J., Bera, A., & Mathies, R. A. Digitally programmable microfluidic automaton for multiscale combinatorial mixing and sample processing. *Lab on a Chip*, 13(2), 288-296, 2013.
- [65] Skafte-Pedersen, P., Hemmingsen, M., Sabourin, D., Blaga, F. S., Bruus, H., & Dufva, M. A self-contained, programmable microfluidic cell culture system with real-time microscopy access. *Biomedical Microdevices*, 14(2), 385-399, 2012.
- [66] Wang, T., Zhang, M., Dreher, D. D., & Zeng, Y. Ultrasensitive microfluidic solidphase ELISA using an actuatable microwell-patterned PDMS chip. *Lab on a Chip*, 13(21), 4190-4197. 2013.
- [67] Jensen, E. C., Bhat, B. P., & Mathies, R. A. A digital microfluidic platform for the automation of quantitative biomolecular assays. *Lab on a Chip*, 10(6), 685-691, 2010.
- [68] Kim, C., Lee, K., Kim, J. H., Shin, K. S., Lee, K. J., Kim, T. S., & Kang, J. Y. A serial dilution microfluidic device using a ladder network generating logarithmic or linear concentrations. *Lab on a Chip*, 8(3), 473-479, 2008.
- [69] X. Niu, F. Gielen, J. B. Edel and A. J. deMello, *Nat. Chem.*, 2011, 3, 437–442.
- [70] Korczyk, P. M., Derzsi, L., Jakiela, S., & Garstecki, P. Microfluidic traps for hardwired operations on droplets. *Lab on a Chip*, 13(20), 4096-4102, 2013.
- [71] Thakur, R., Zhang, Y., Amin, A., & Wereley, S. Programmable microfluidic platform for spatiotemporal control over nanoliter droplets. *Microfluidics and Nanofluidics*, 18(5-6), 1425-1431, 2015.

[72] Grover, W. H., von Muhlen, M. G., & Manalis, S. R. Teflon films for chemically inert microfluidic valves and pumps. *Lab on a Chip*, 8(6), 913-918, 2008.

[73] Inman, W., Domansky, K., Serdy, J., Owens, B., Trumper, D., & Griffith, L. G. Design, modeling and fabrication of a constant flow pneumatic micropump. *Journal of Micromechanics and Microengineering*, 17(5), 891, 2007.