Platform-Based Design: From Multi-Core Platforms to Biochips and beyond

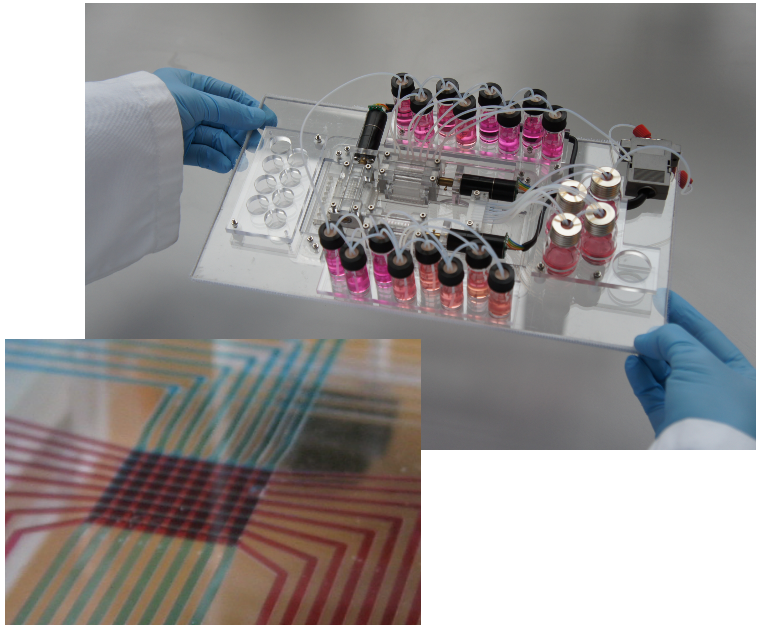
Prof. Jan Madsen

Embedded Systems Engineering

DTU Informatics
Department of Informatics and Mathematical Modeling

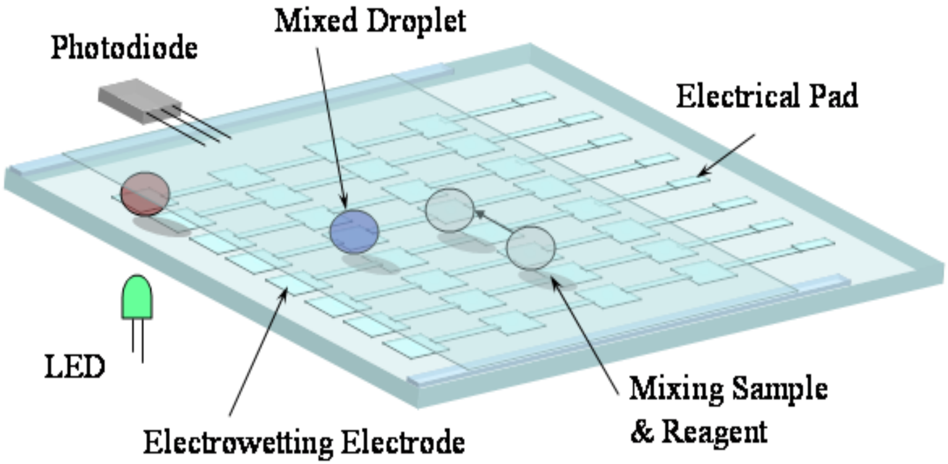
Microfluidic Biochips

Continuous-flow biochips



Technical Univ. of Denmark
2010

Droplet-based biochips



Duke University
2002

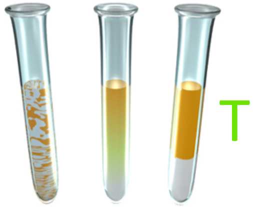
Outline

- Motivation & relation to MPSoC
- Digital Microfluidic Biochips
  - Technology and architectures
  - Module-based synthesis
  - Routing-based synthesis
- Flow-Based Microfluidic Biochips
  - Biochip synthesis
  - Possibilities and challenges

Motivation

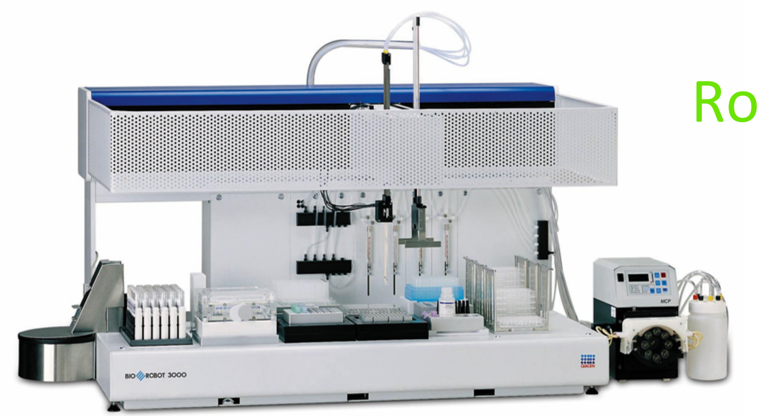
- Biotech
  - DNA analysis
- Medicine
  - Clinical diagnosis
  - Therapeutics
- Ecology
  - Monitoring the quality of air/water/food
- Pharmacy
  - Screening
  - Synthesis of new drugs

Motivation



Test tubes

- □ Automation
- □ Integration
- □ Miniaturization

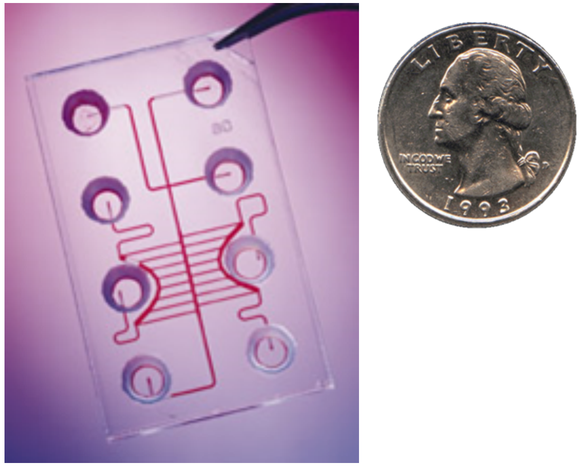


Robotics

- ☑ Automation
- ☑ Integration
- □ Miniaturization

Microfluidics

- ☑ Automation
- ☑ Integration
- ☑ Miniaturization

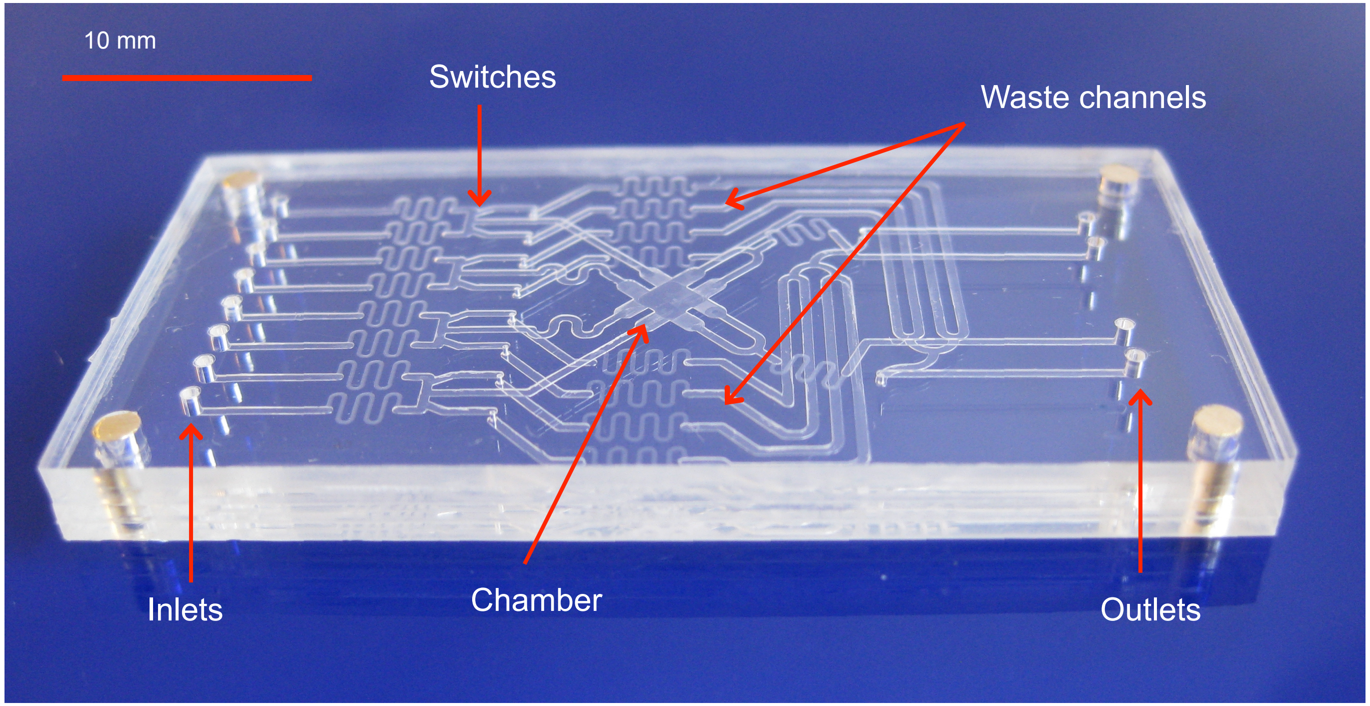


Microfluidic Biochips

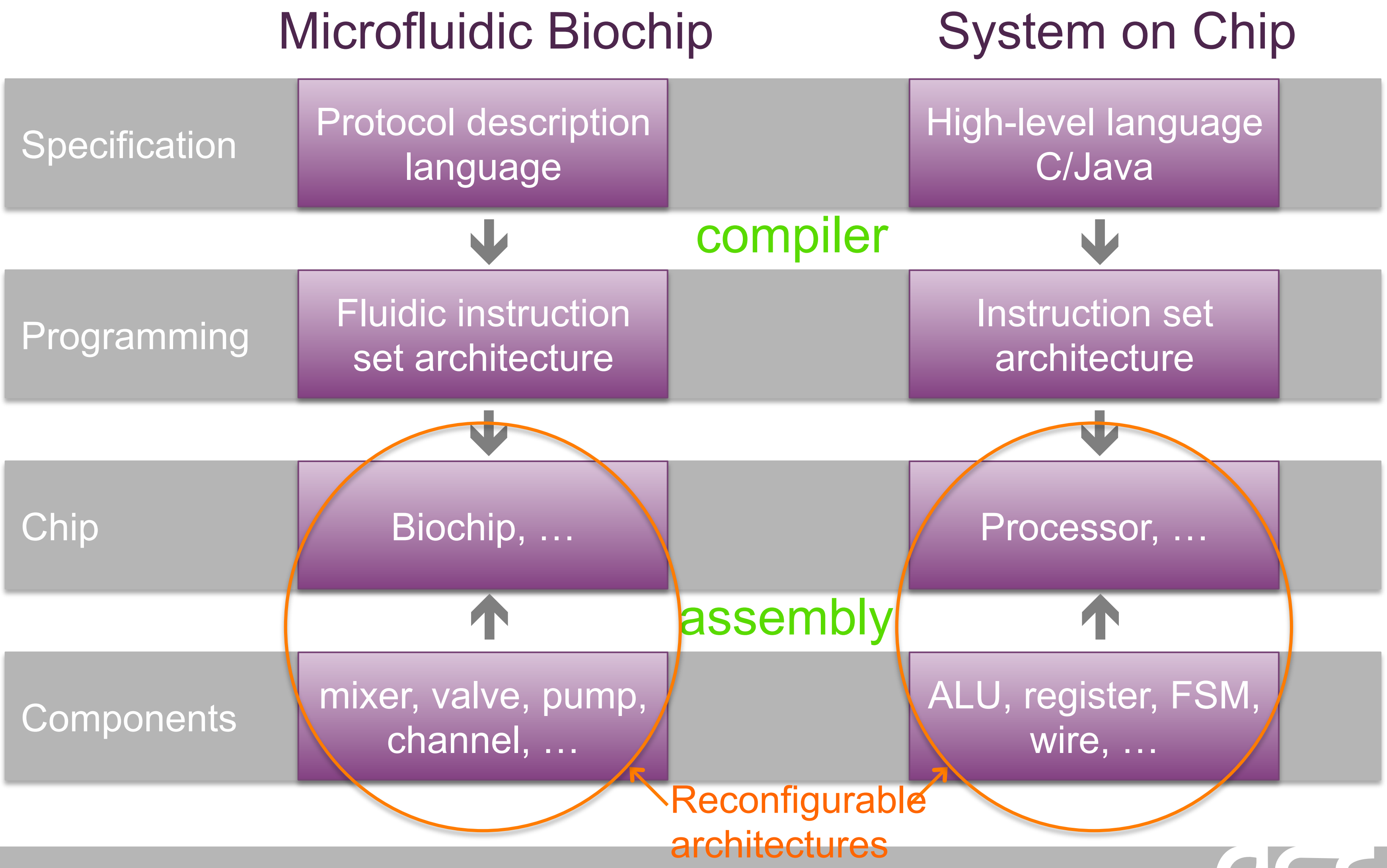
- Advantages:
  - High throughput (reduced sample / reagent consumption)
  - Space (miniaturization)
  - Time (parallelism)
  - Automation (minimal human intervention)

Microfluidic biochip?

- Manipulations of continuous liquid through fabricated micro-channels



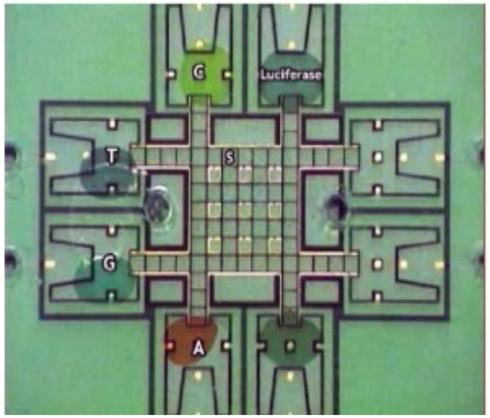
Biochip design



	Microfluidic Biochip	System on Chip
Specification	Protocol description language	High-level language C/Java
Programming	Fluidic instruction set architecture	Instruction set architecture
Chip	Biochip, …	Processor, …
Components	mixer, valve, pump, channel, …	ALU, register, FSM, wire, …

References:

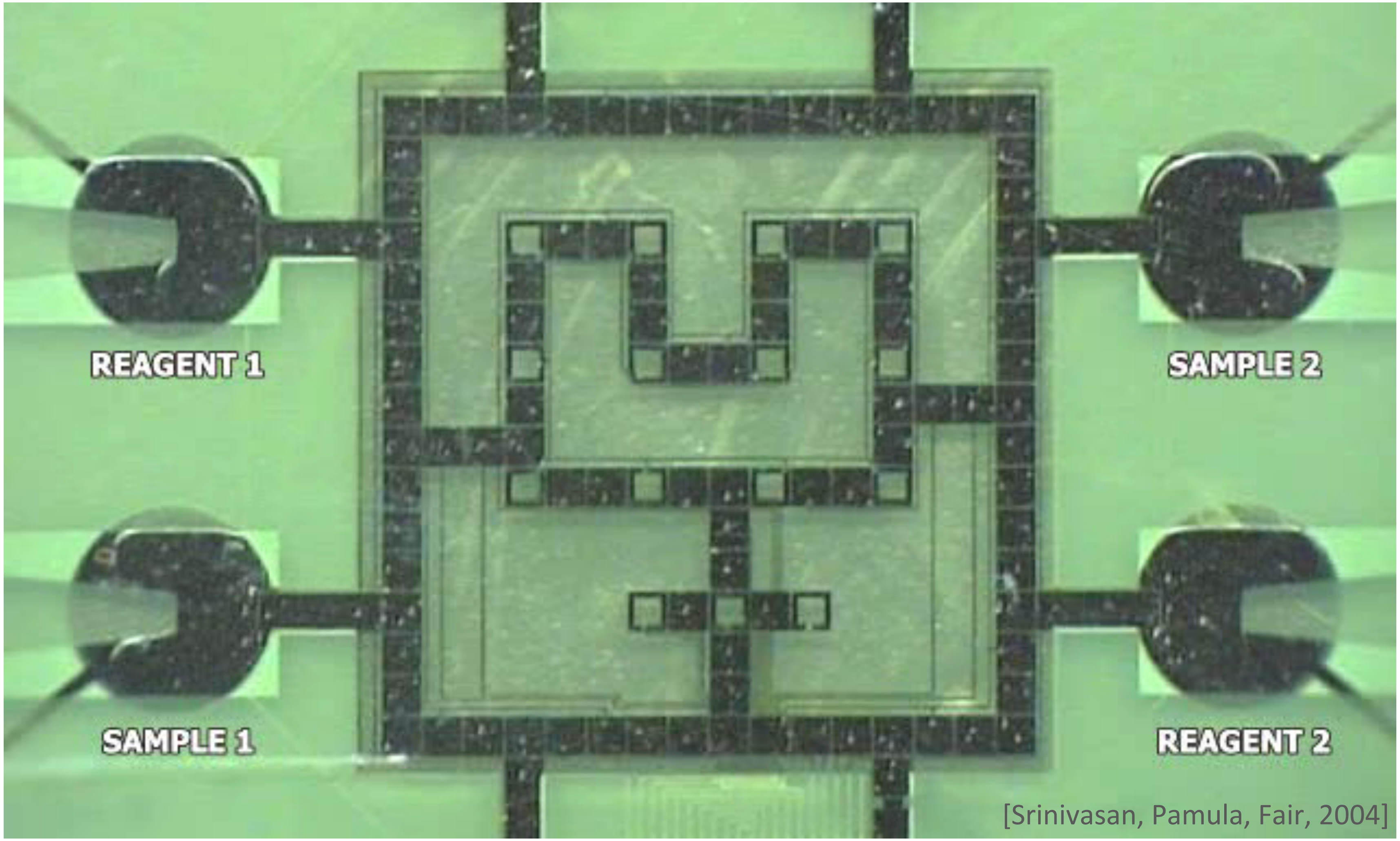
1. Elena Maftei, Paul Pop, Jan Madsen, Resent Research and Emerging Challenges in the System-Level Design of Digital Microfluidic Biochips, Proceedings of the International System on Chip Conference, 2011 (**invited paper**)



PART 2: DIGITAL MICROFLUIDIC BIOCHIPS

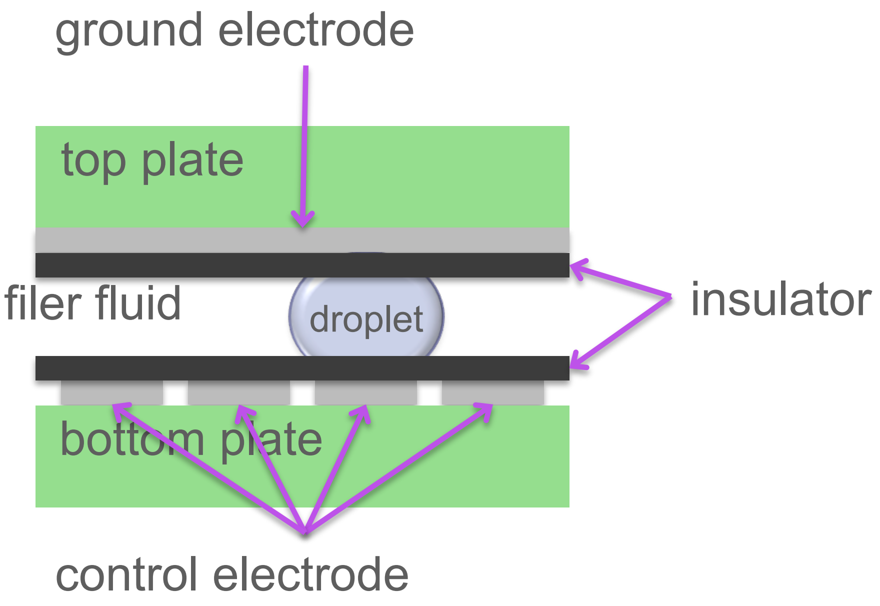
TECHNOLOGY AND ARCHITECTURES

Digital microfluidic biochip



[Srinivasan, Pamula, Fair, 2004]

Digital microfluidic biochip

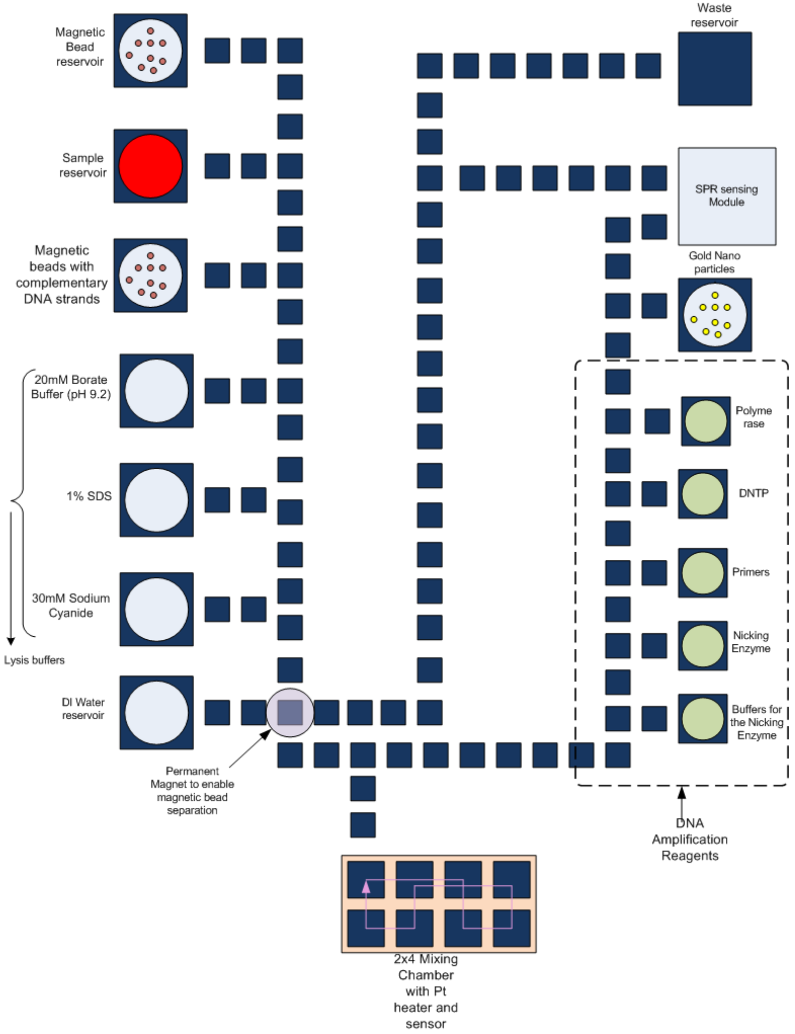


Speed: 12-25 cm/s
Size of electrode: 0.15 cm
Cell-to-cell transport: ~0.01 s

Biochip architecture?

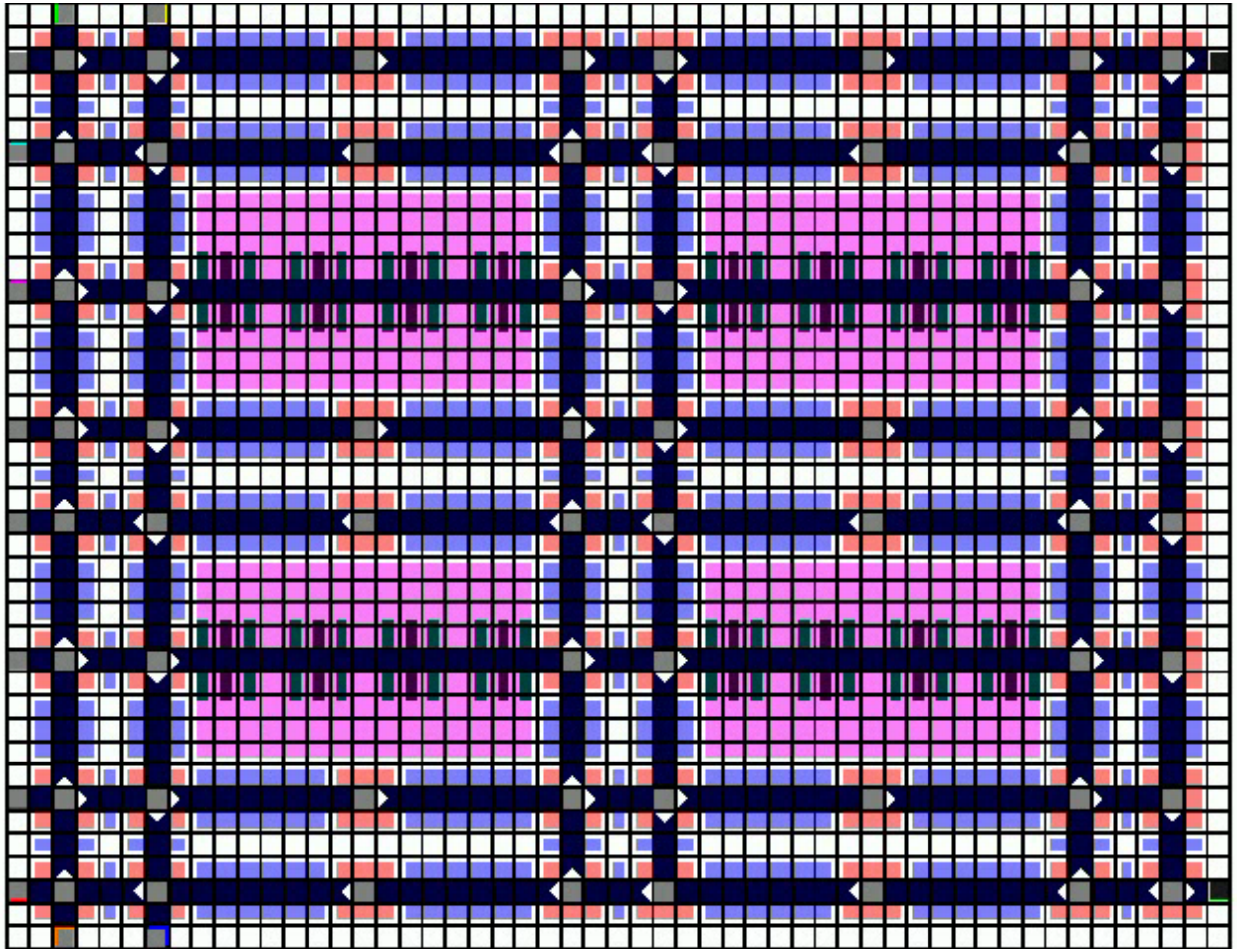
- Application specific architecture
  - Spatial and temporal assignment done at design-time
- General purpose architecture
  - Spatial assignment done at design-time
  - Temporal assignment done at run-time
- Reconfigurable architecture
  - Spatial and temporal assignment done at run-time

Application specific biochip


Magnetic Bead reservoir
Sample reservoir
Magnetic beads with complementary DNA strands
20mM Borate Buffer (pH 9.2)
1% SDS
30mM Sodium Cyanide
Lysis buffers
DI Water reservoir
Permanent Magnet to enable magnetic bead separation
2x4 Mixing Chamber with Pt heater and sensor
Waste reservoir
SPR sensing Module
Gold Nano particles
Polyme rase
DNTP
Primers
Nicking Enzyme
Buffers for the Nicking Enzyme
DNA Amplification Reagents

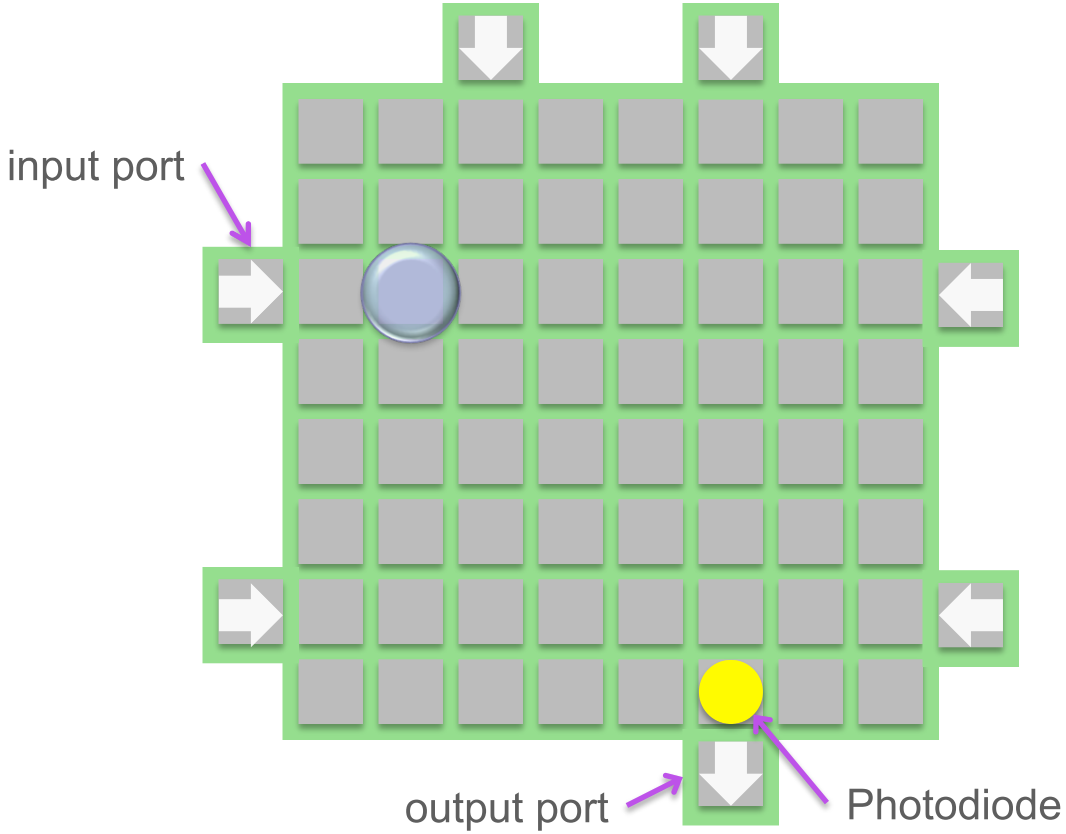
- Biochip for malaria detection
- Operation:
  - Infected cell isolation
  - Cell Lysis
  - DNA extraction
  - DNA amplification using PCR
  - Optical detection using SPR

General purpose biochip



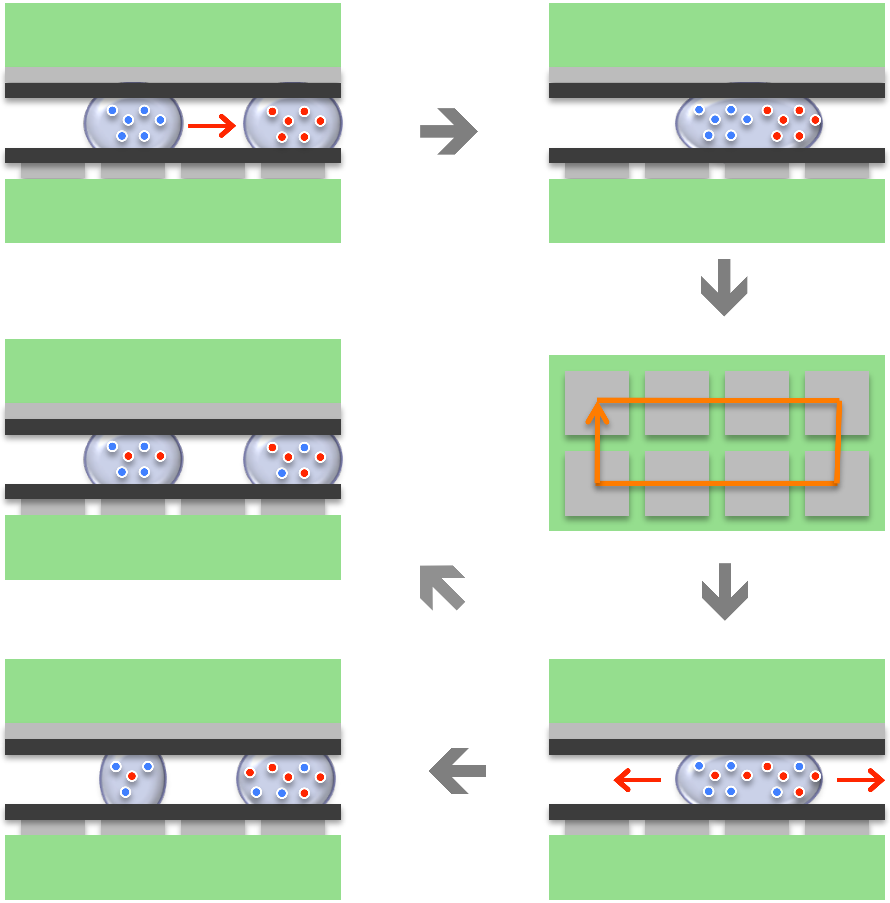
[Griffith, Akella, 2005]

Reconfigurable biochip



Biochemical operations

- Transport
- Merging
- Mixing
- Splitting
- Diluting
- Detection
- …



References:

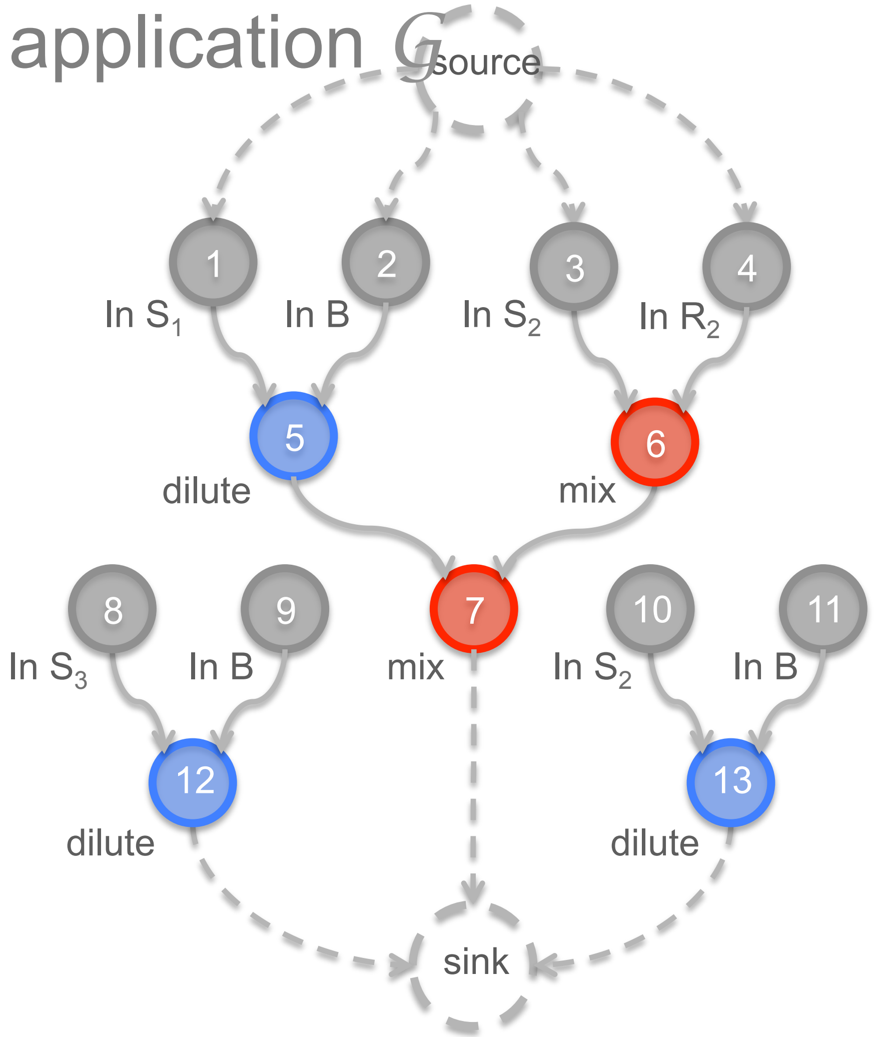
1. Elena Maftei, Paul Pop, Jan Madsen, Tabu Search-Based Synthesis of Dynamically Reconfigurable Digital Microfluidic Biochips. In Proceedings of the International Conference on Compilers, Architectures, and Synthesis for Embedded Systems (CASES), 2009 (**best paper award**).
2. E. Maftei, P. Pop, J. Madsen, Tabu Search-Based Synthesis of Digital Microfluidic Biochips with Dynamically Reconfigurable Non-Rectangular Devices, *Automation for Embedded Systems,* vol: 14, no. 3, September 2010, Pages 287-307.

PART 2: DIGITAL MICROFLUIDIC BIOCHIPS

MODULE-BASED SYNTHESIS

Biochemical application

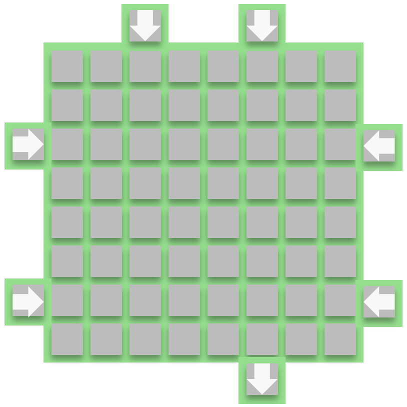
application $\mathcal{G}$



library $\mathcal{L}$

module	Operation	Area (cells)	Time (sec)
M1	Mixing	2x4	3
M2	Mixing	2x2	4
D1	Dilution	2x4	4
D2	Dilution	2x2	5

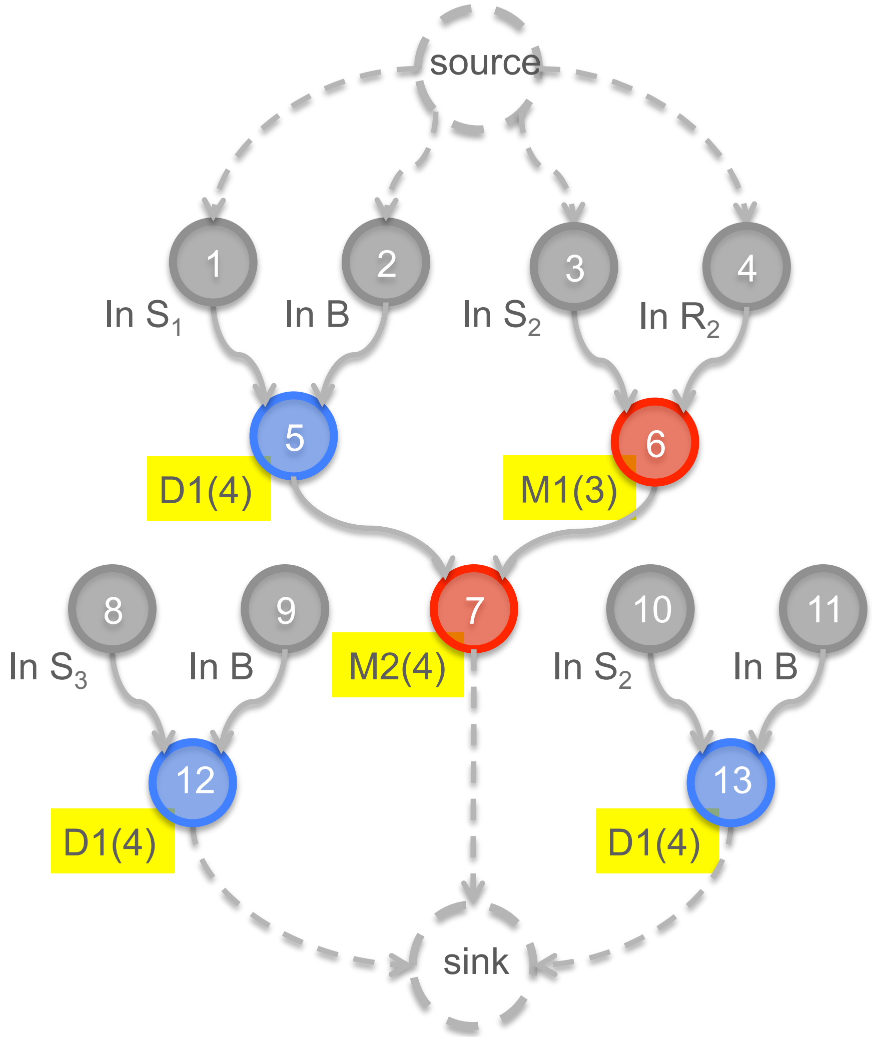
array C

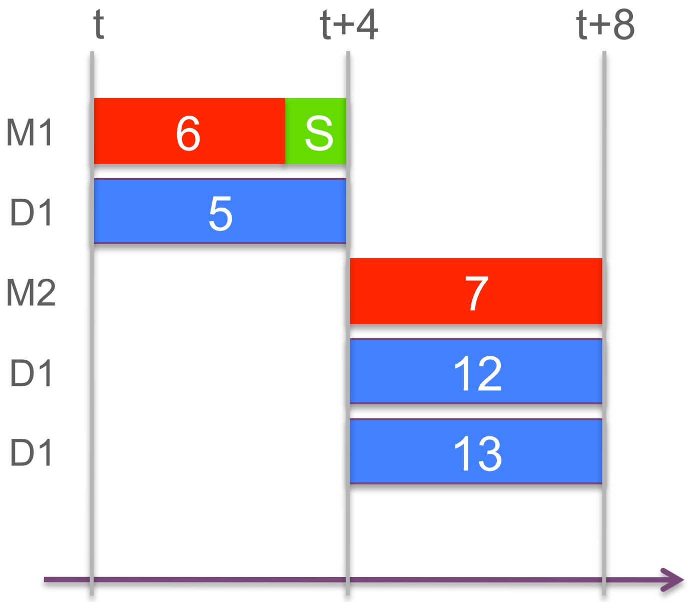


Mapping biochemical applications onto microfluidic biochips

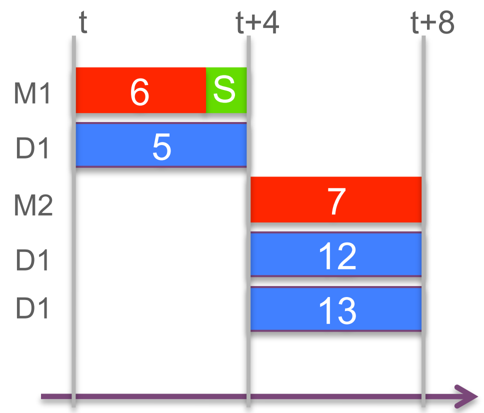
- Allocation $\mathcal{A}$
  - Determine modules $\mathcal{M}_k$ from library $\mathcal{L}$
- Binding $\mathcal{B}$
  - Assign each operation O_i to a module $\mathcal{M}_k$
- Schedule $\mathcal{S}$
  - Determine start time t_i^{start} of each operation O_i
- Placement $\mathcal{P}$
  - Place modules on the $m \times n$ array
- Synthesis Ψ
  - Given $<\mathcal{G}, \mathcal{C}, \mathcal{L}>$, find $\Psi = <\mathcal{A},\mathcal{B},\mathcal{S},\mathcal{P}>$ which minimize the schedule length $\delta_{\mathcal{G}}$

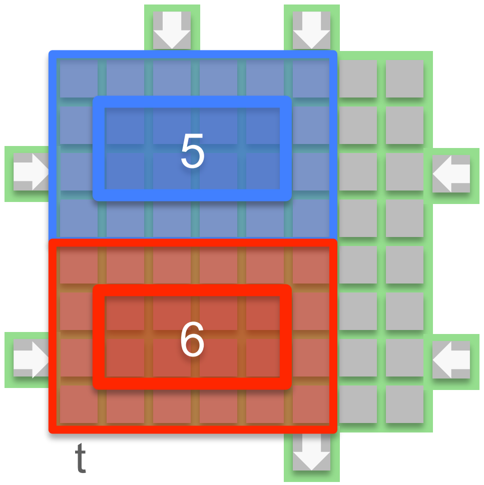
Scheduling

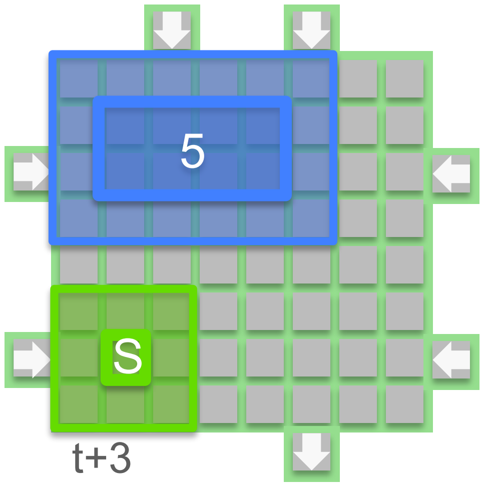

source
1
2
3
4
In S_1
In B
In S_2
In R_2
5
6
D1(4)
M1(3)
8
9
7
10
11
In S_3
In B
M2(4)
In S_2
In B
12
13
D1(4)
D1(4)
sink

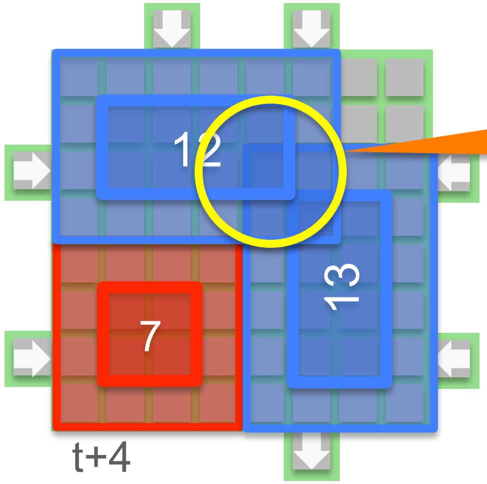

t
t+4
t+8
M1
6
S
D1
5
M2
7
D1
12
D1
13

Scheduling





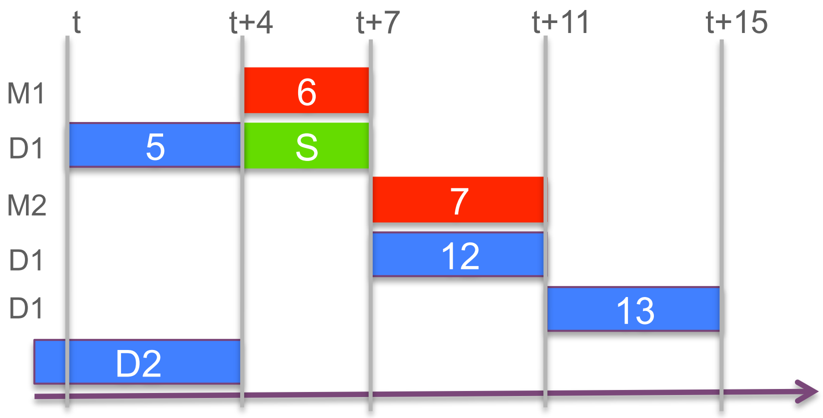


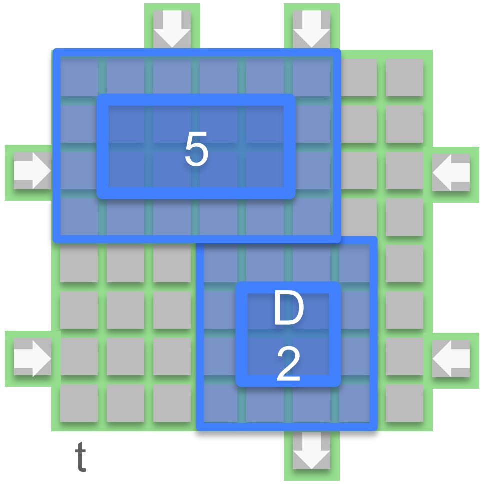


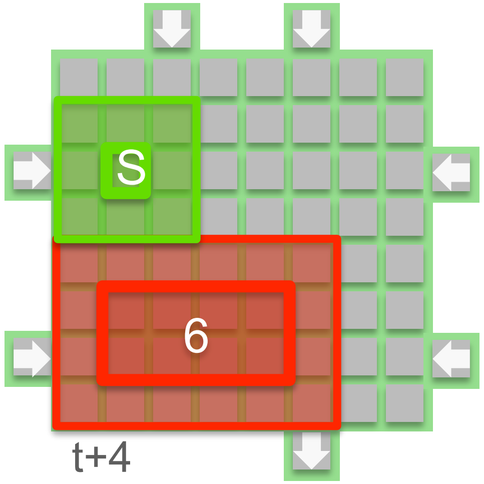
Overlapping modules

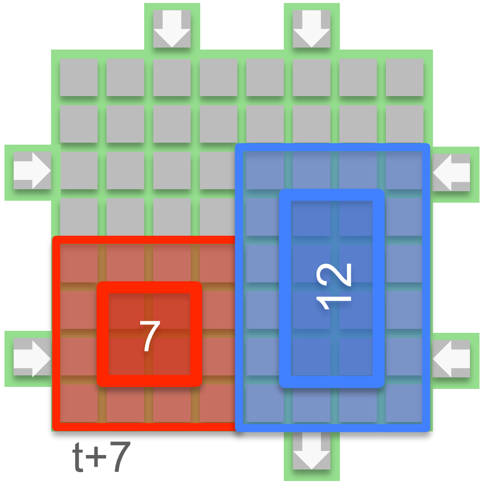
Concurrent biochemical applications

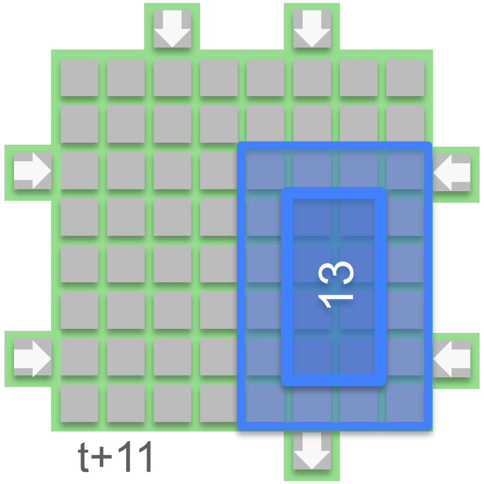
Scheduling with placement



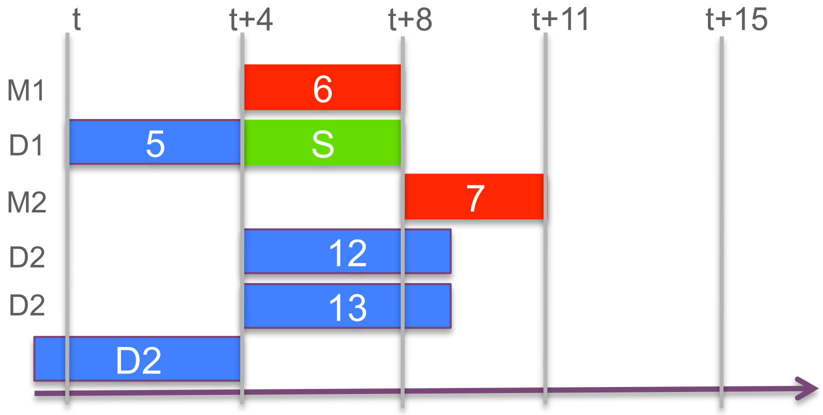


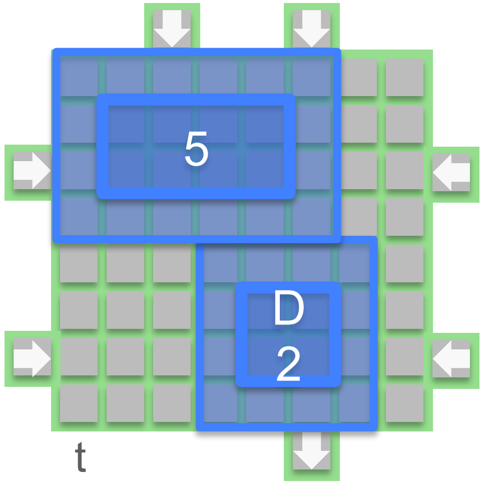


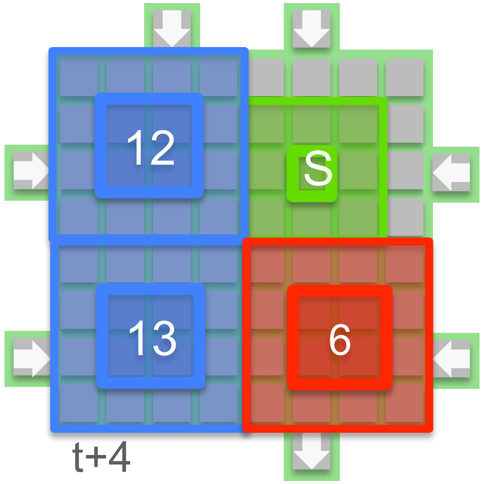


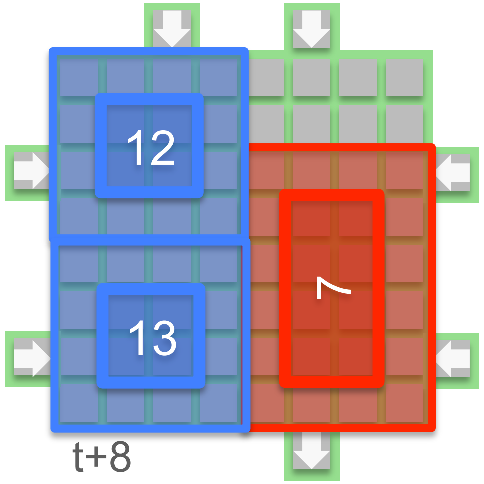


Scheduling with placement

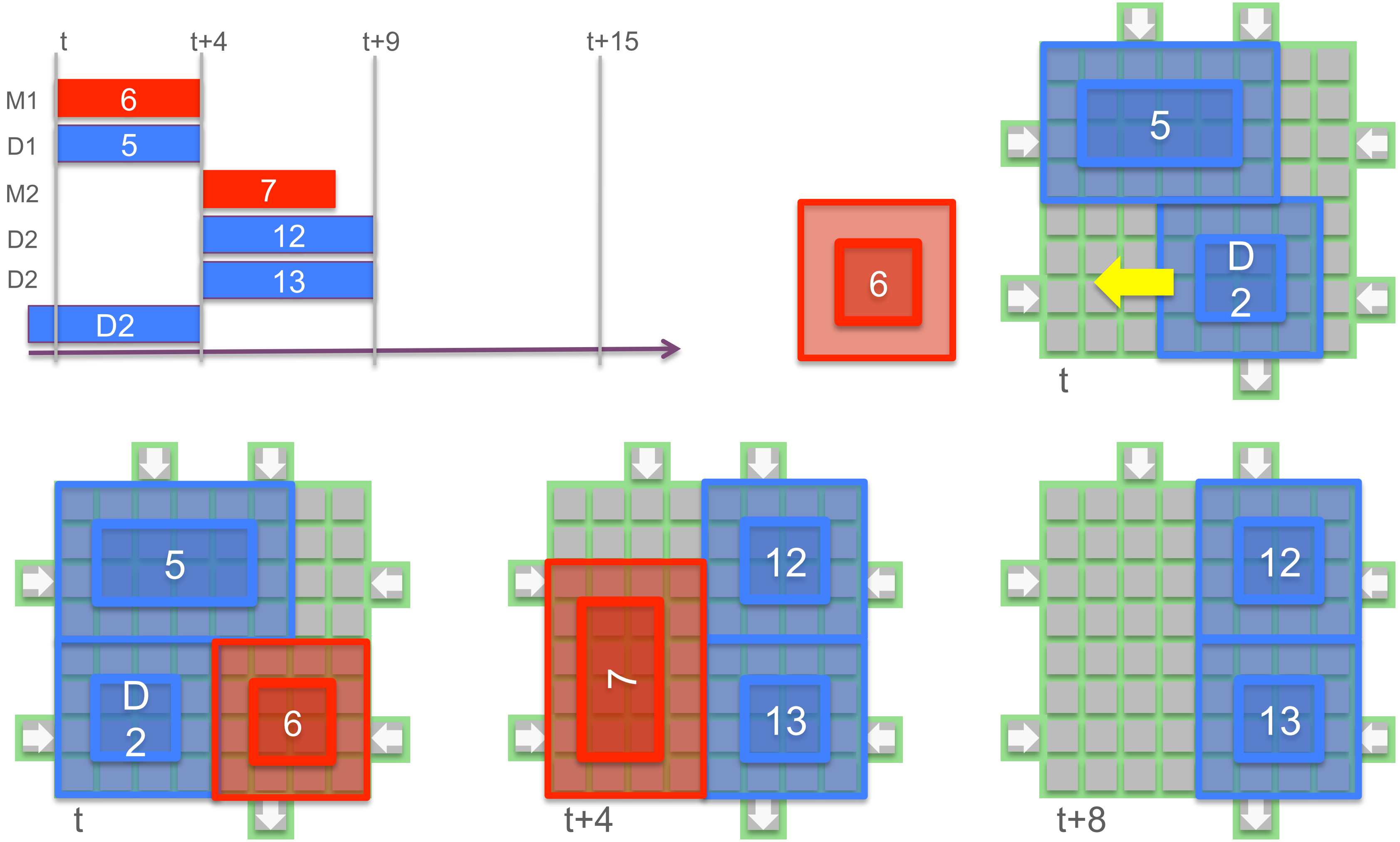








Scheduling with dynamic placement



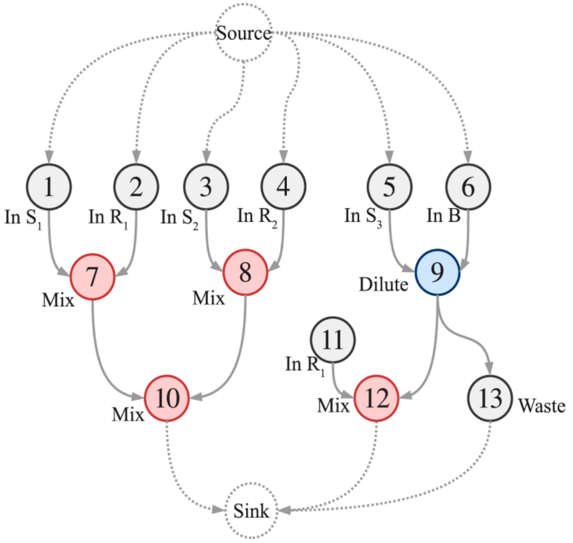
References:

1. Elena Maftei, Paul Pop, Jan Madsen, Routing-Based Synthesis of Digital Microfluidic Biochips. Proceedings of the Compilers, Architecture, and Synthesis for Embedded Systems Conference (CASES'10), pp. 41-49, 2010 (**best paper candidate**)

PART 2: DIGITAL MICROFLUIDIC BIOCHIPS

ROUTING-BASED SYNTHESIS

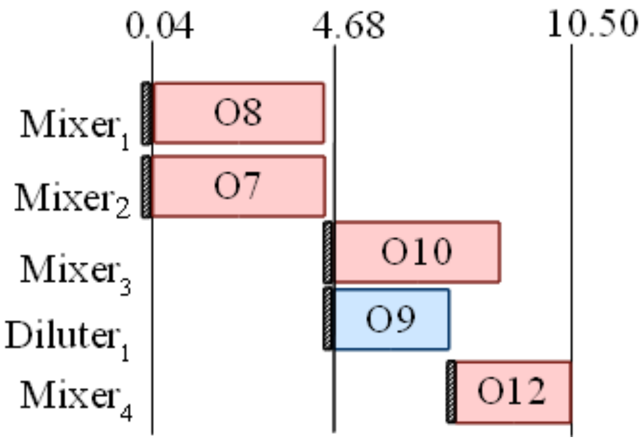
Module-Based Design Tasks



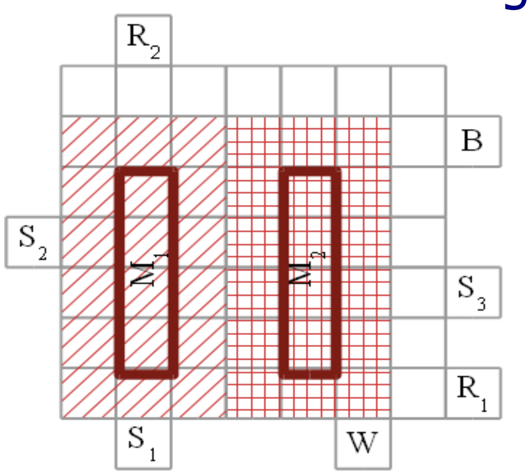
Allocation

Operation	Area(cells)	Time(s)
Mix/Dlt	2x4	2.8
Mix/Dlt	1x4	4.6
Mix/Dlt	2x3	5.6
Mix/Dlt	2x2	9.96

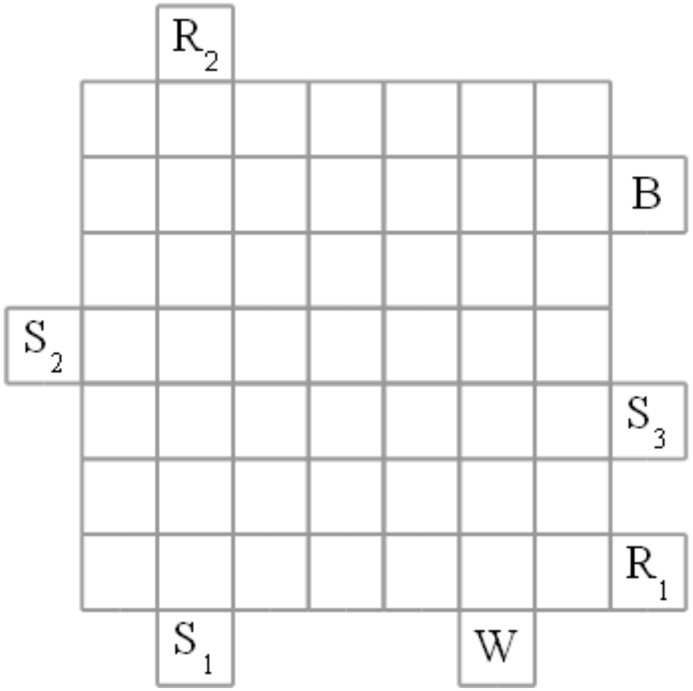
Binding & Scheduling

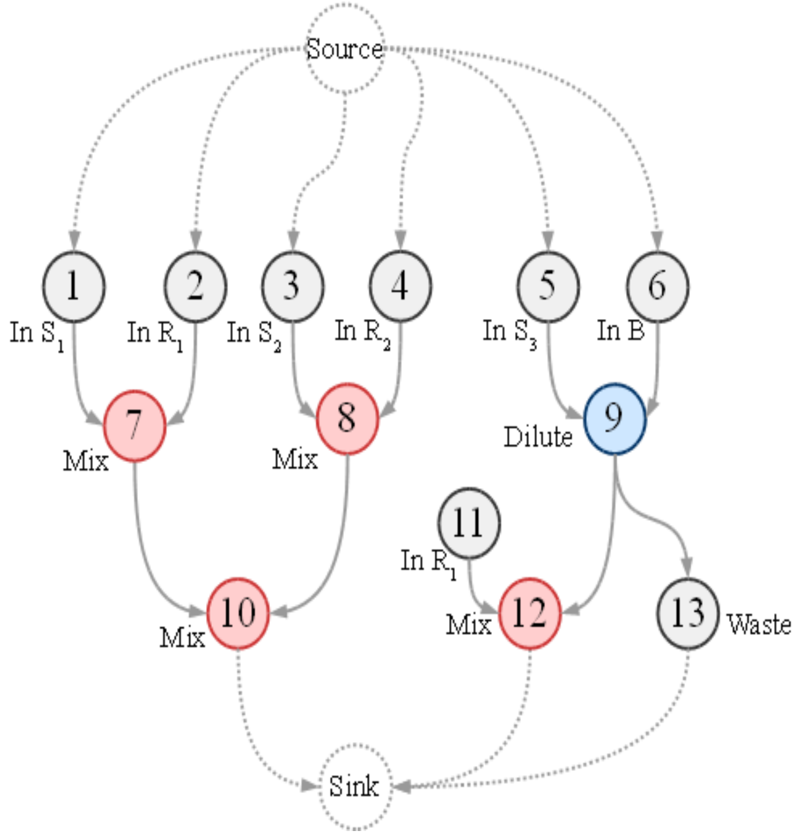


Placement & Routing



Module-Based Synthesis



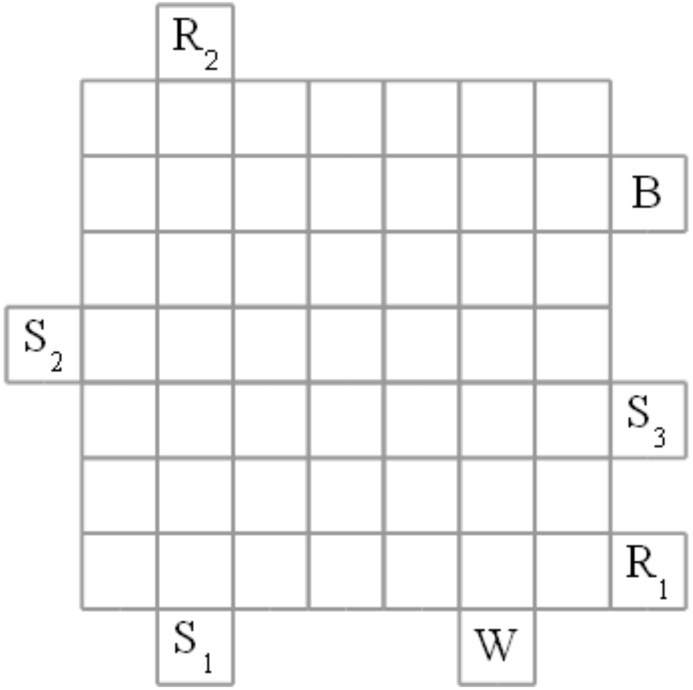


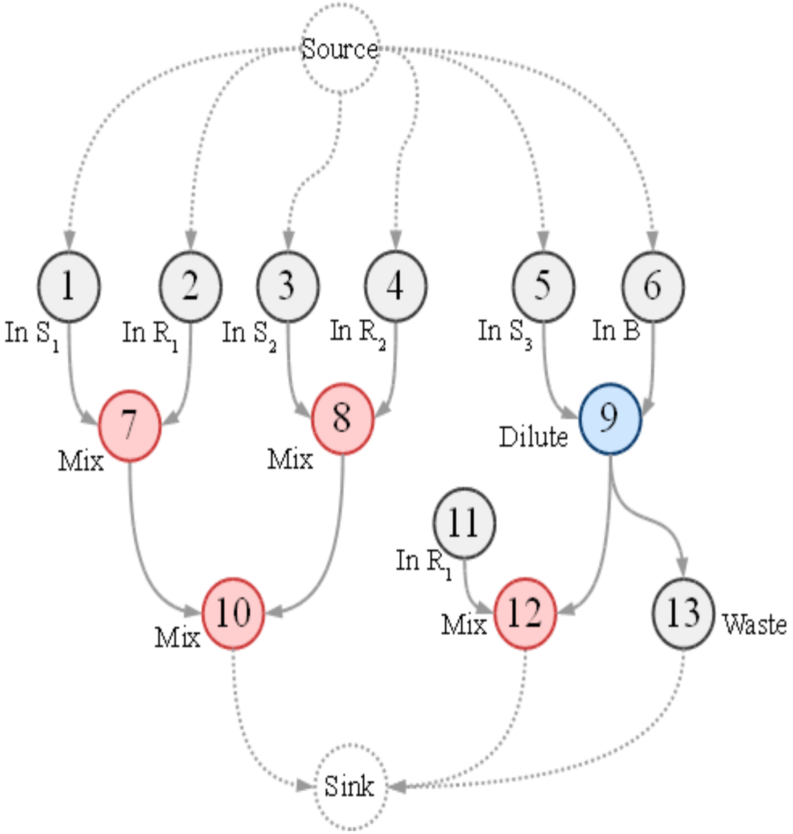
Module-Based Synthesis

O7

O8

O9



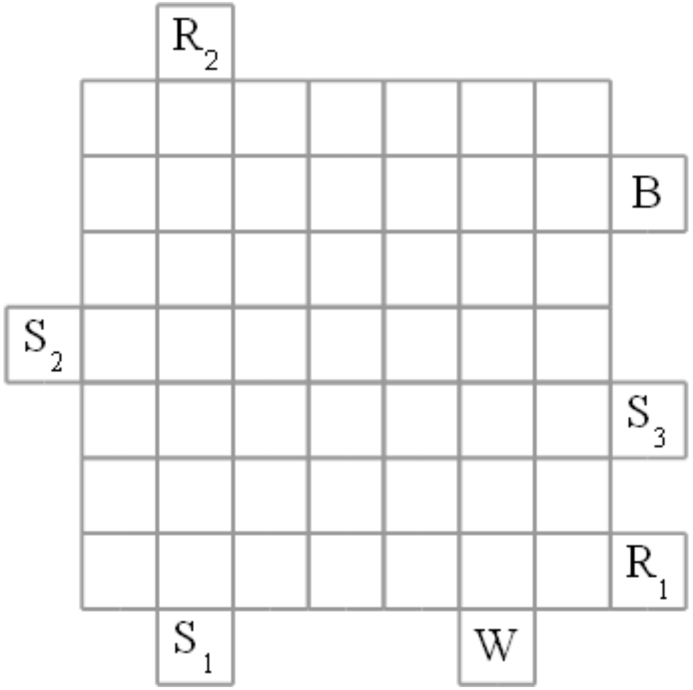


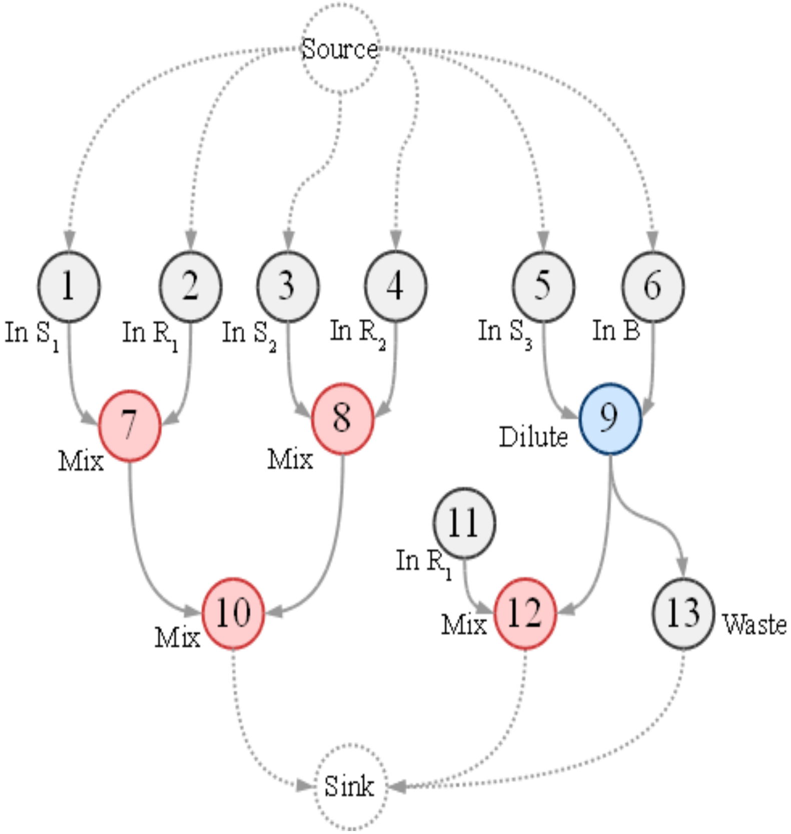
Module-Based Synthesis

O7

O8

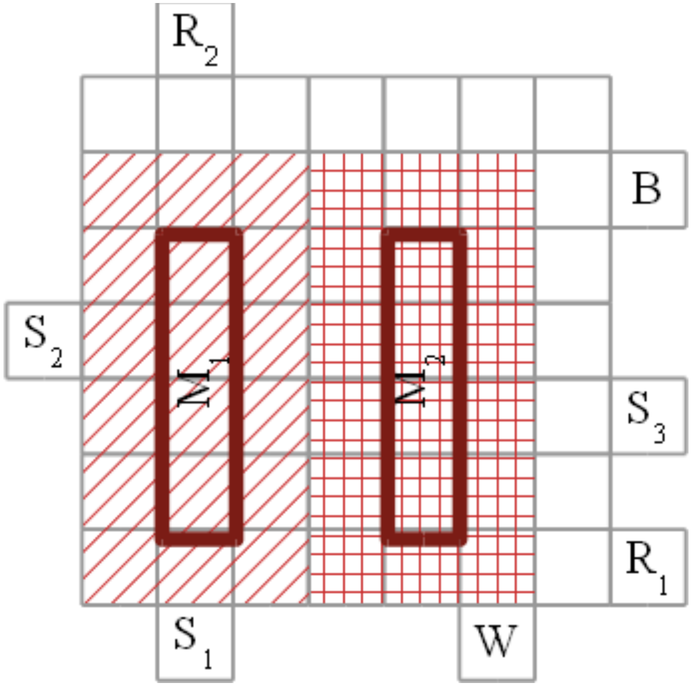
O9

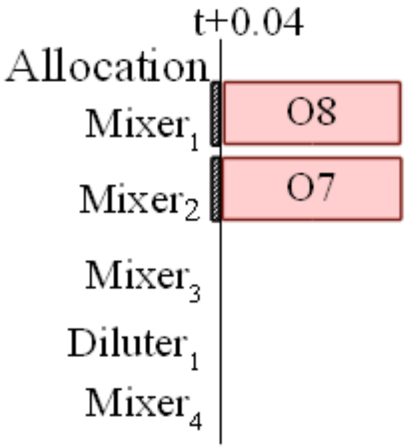




Module-Based Synthesis

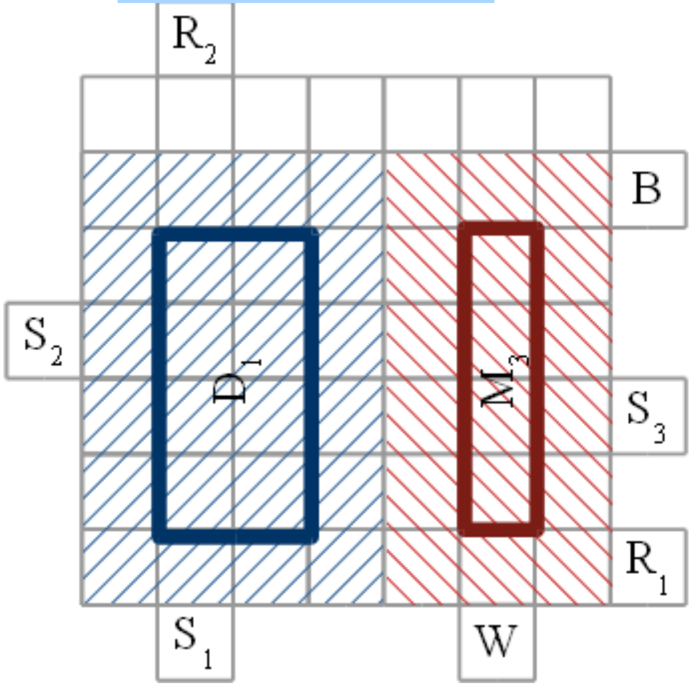
O7	1x4
O8	1x4

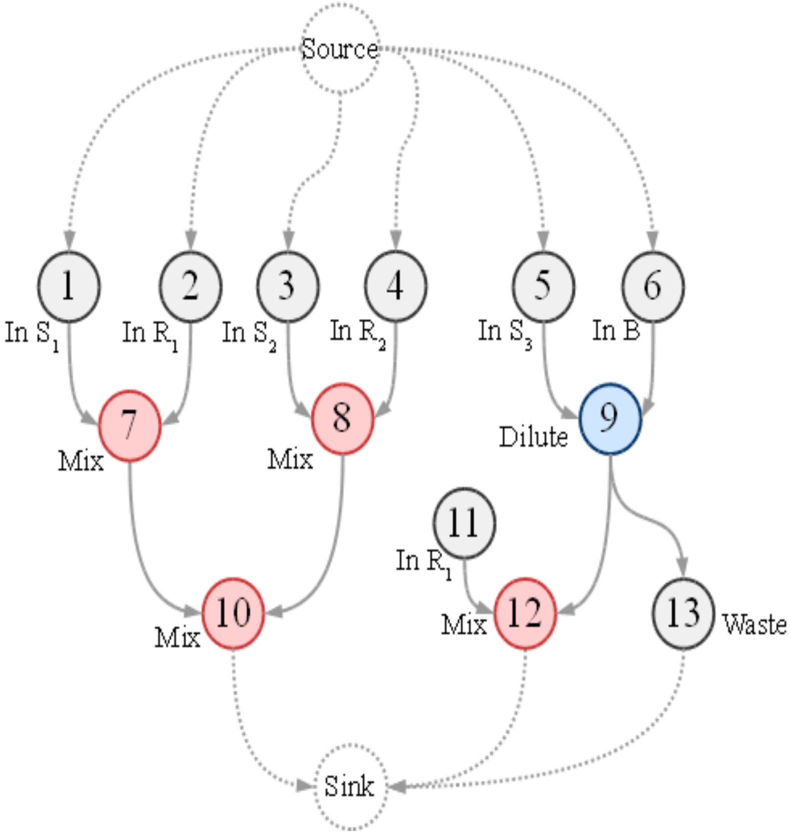




Module-Based Synthesis

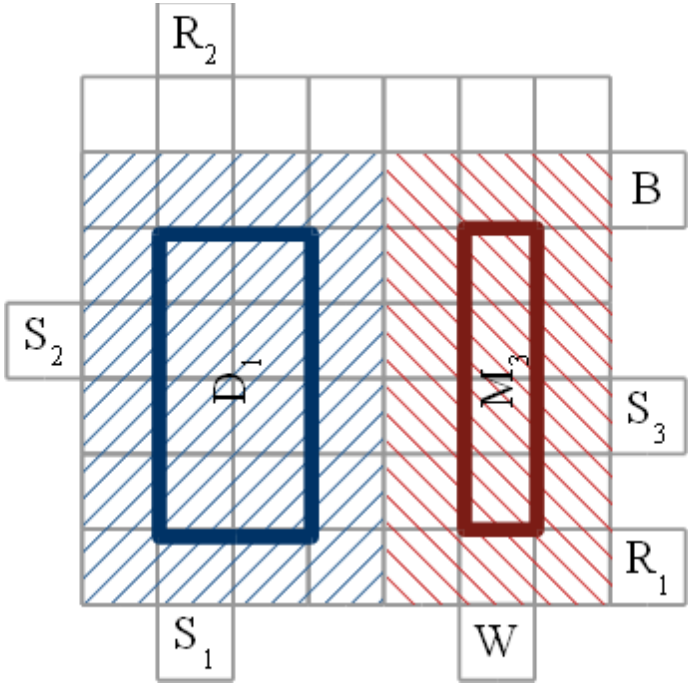
O9	2x4
O10	1x4

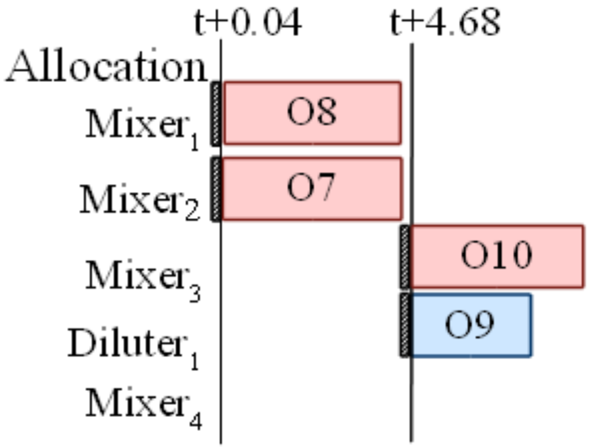




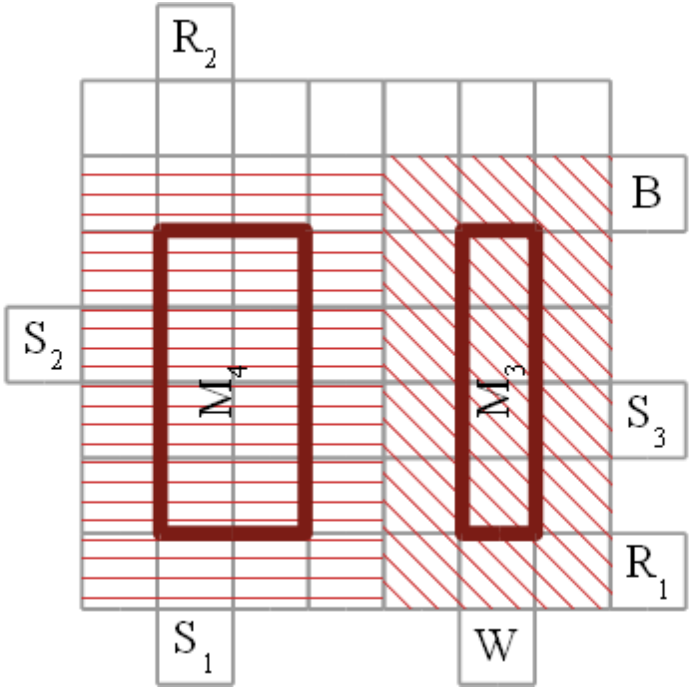
Module-Based Synthesis

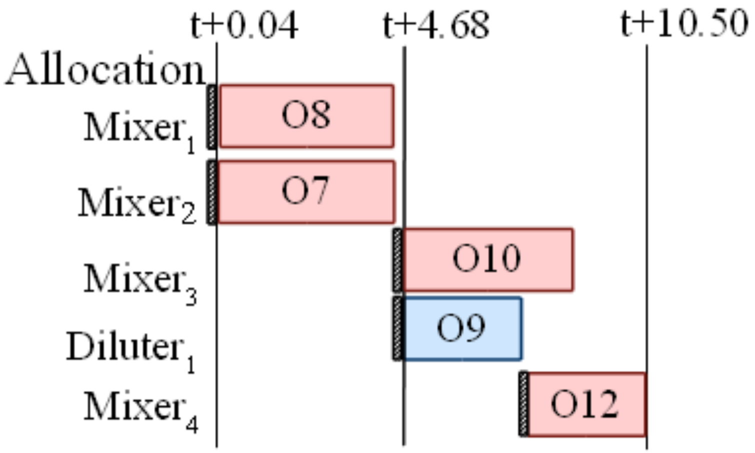
O9	2x4
O10	1x4



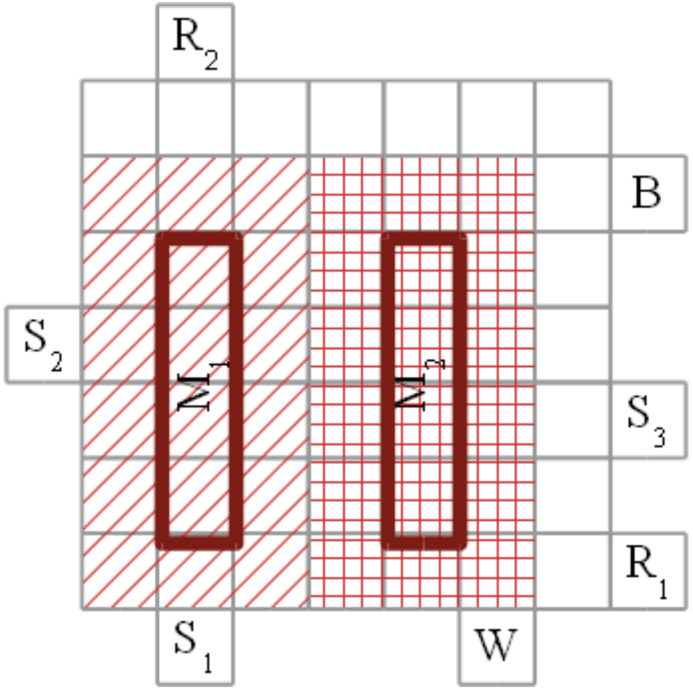


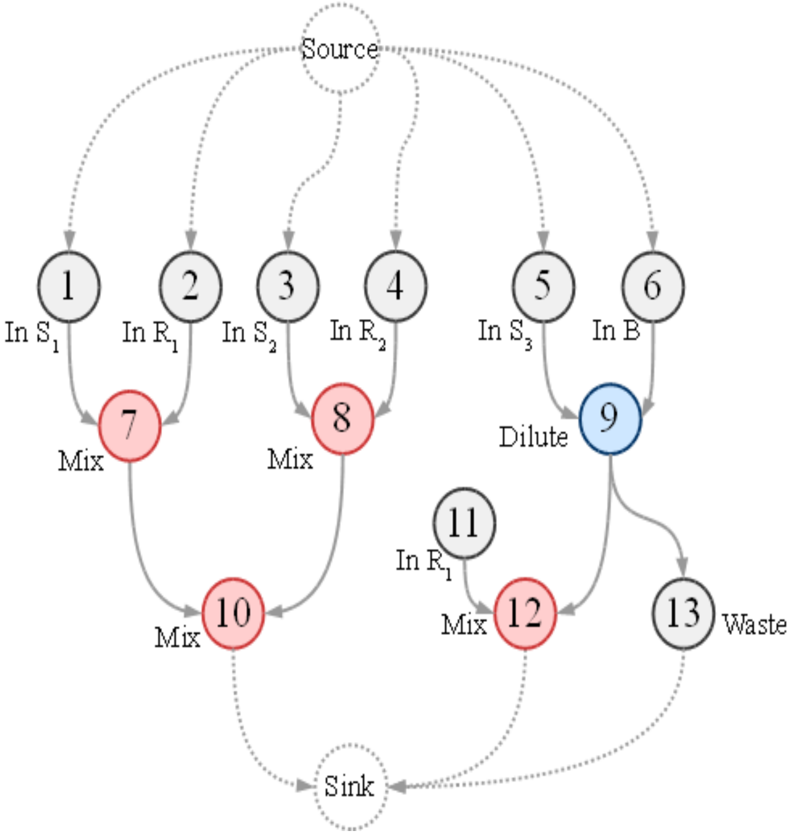
Module-Based Synthesis



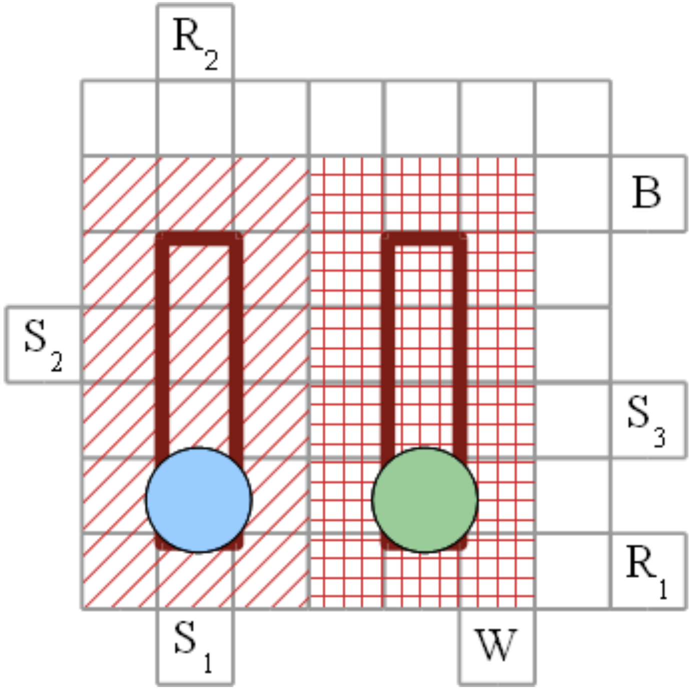


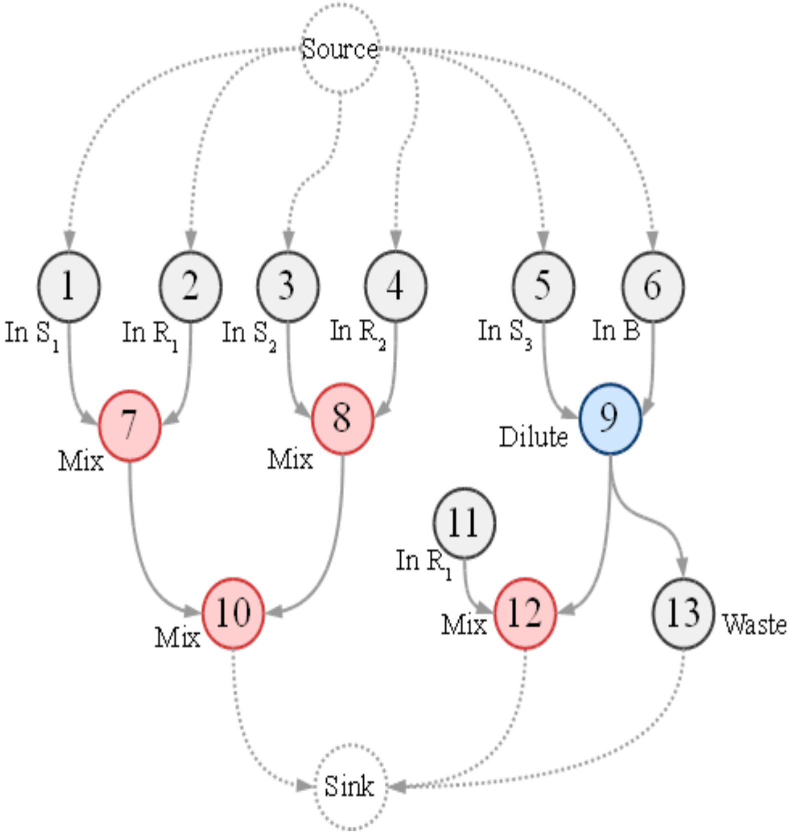
Reconfigurability



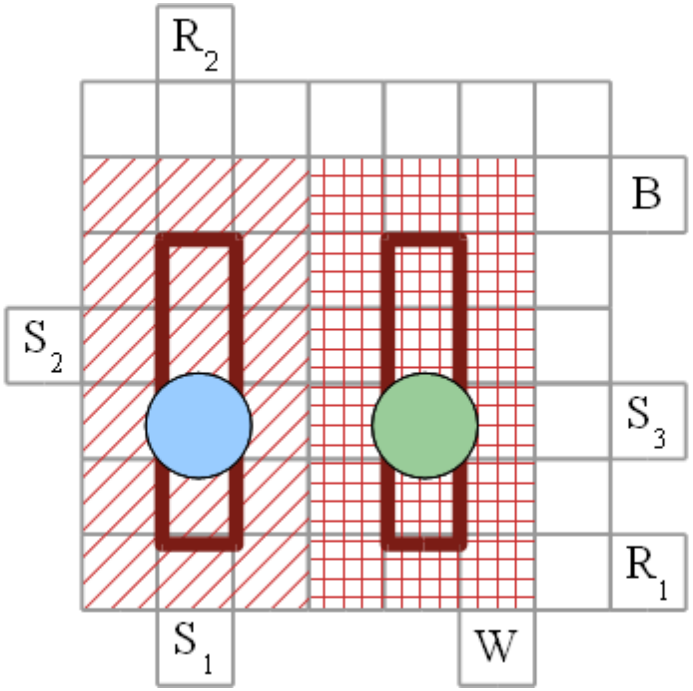


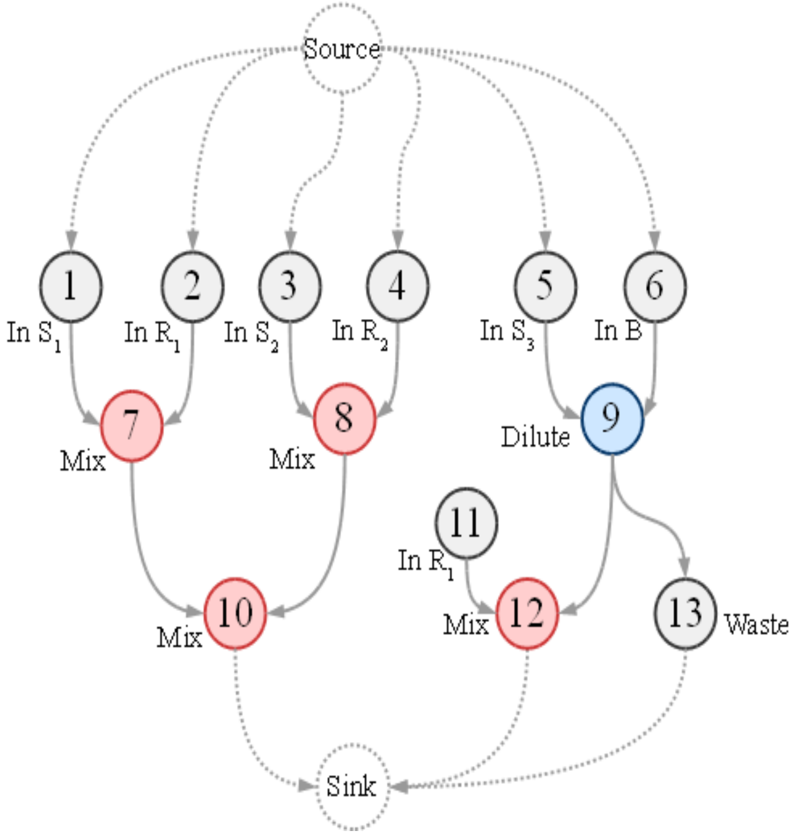
Reconfigurability



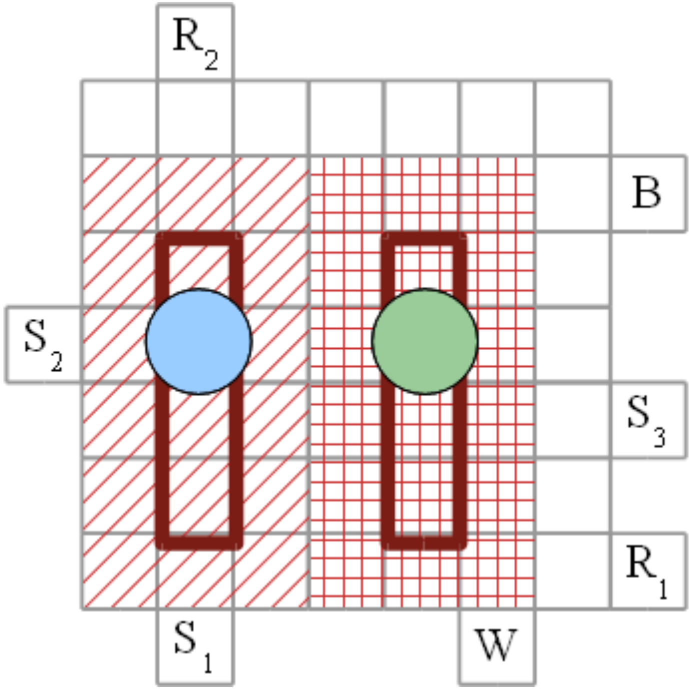


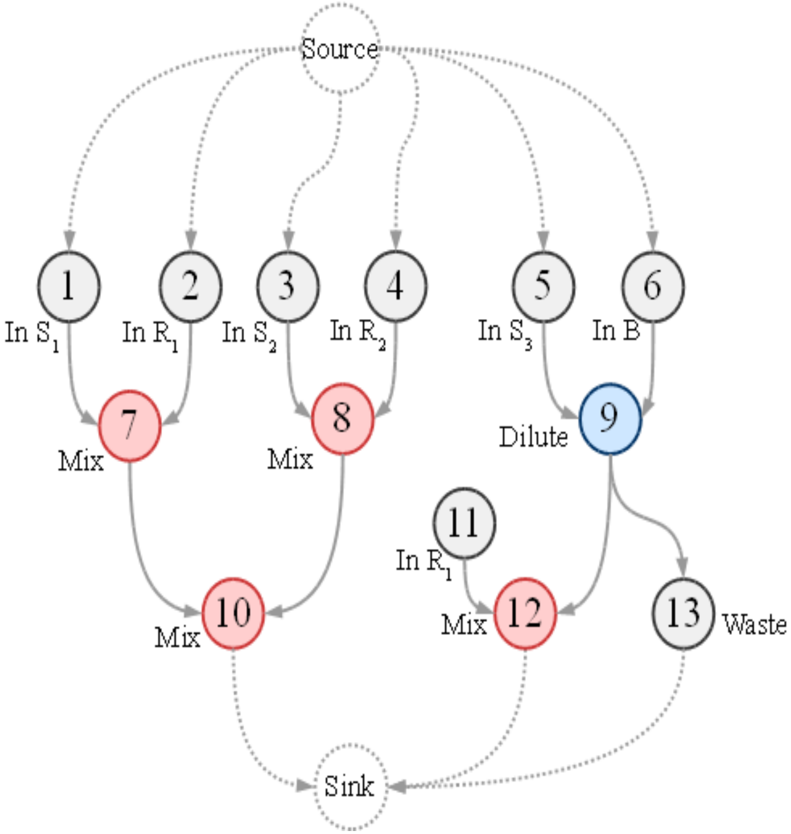
Reconfigurability



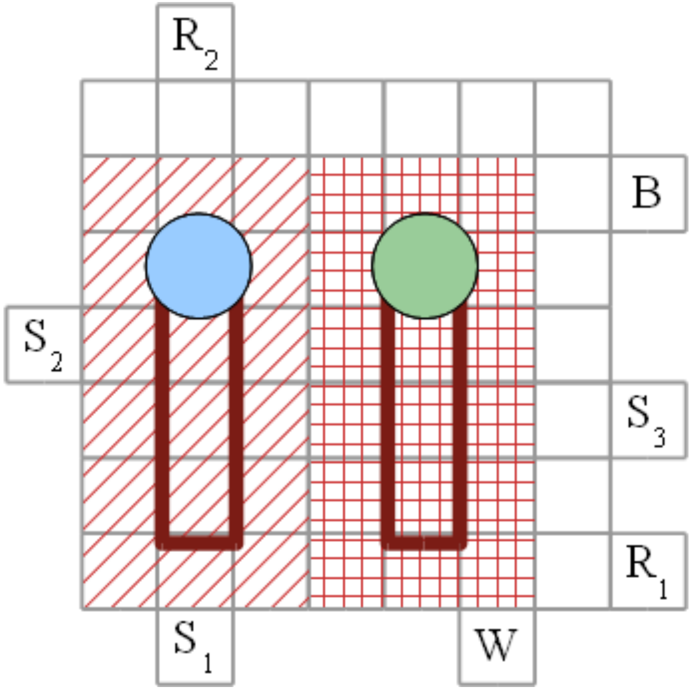


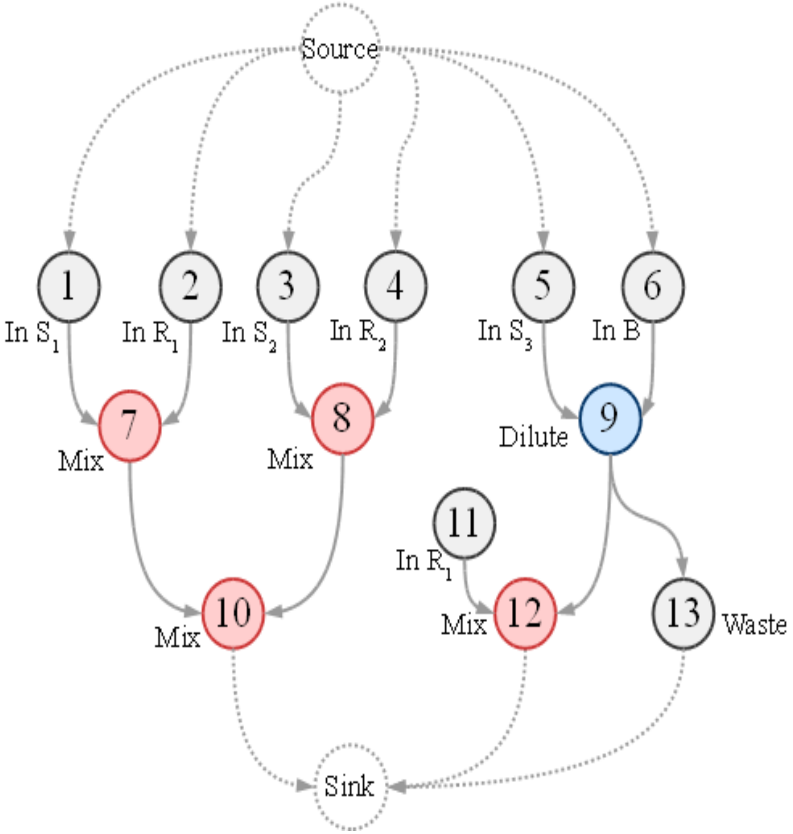
Reconfigurability



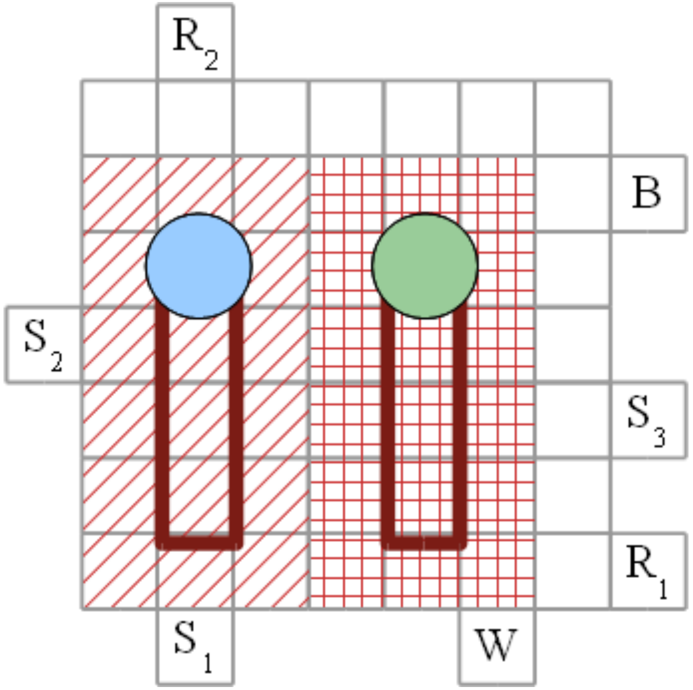


Reconfigurability



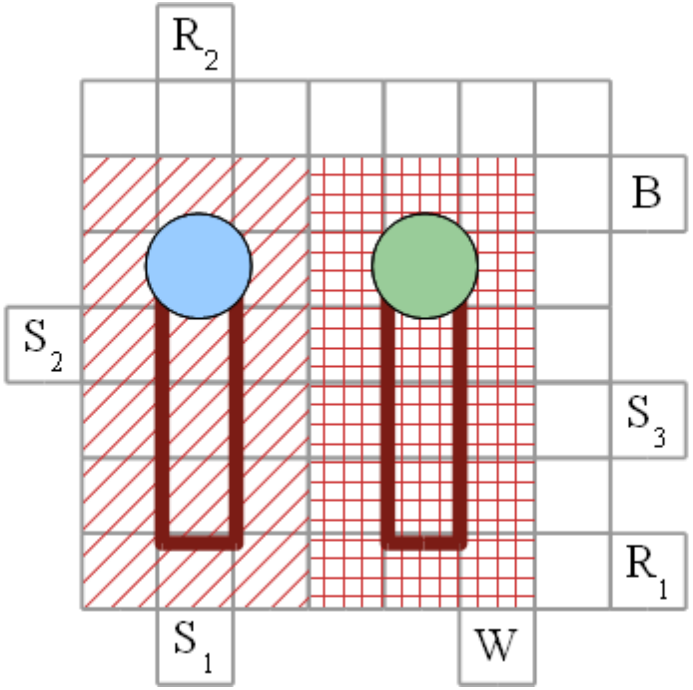


Reconfigurability



- Disadvantages of modules:
  - Pessimistic segregation area
  - Routing performed post-synthesis

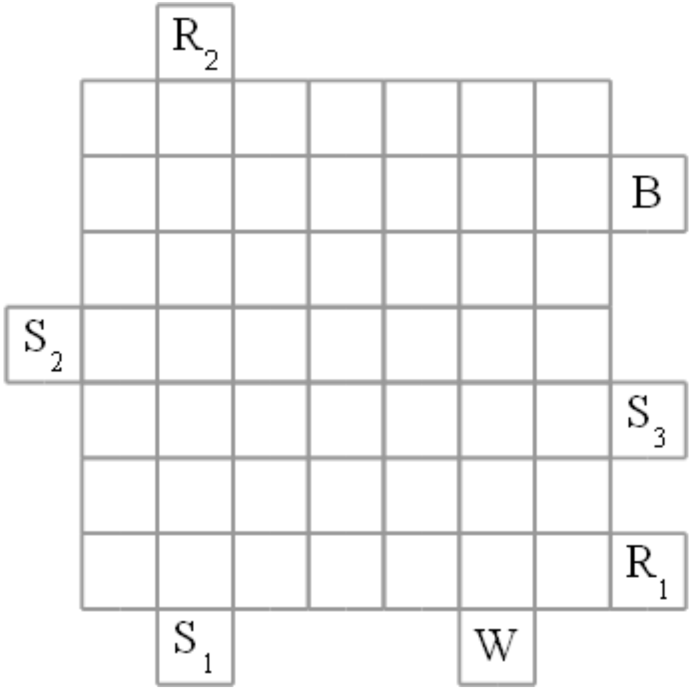
Reconfigurability

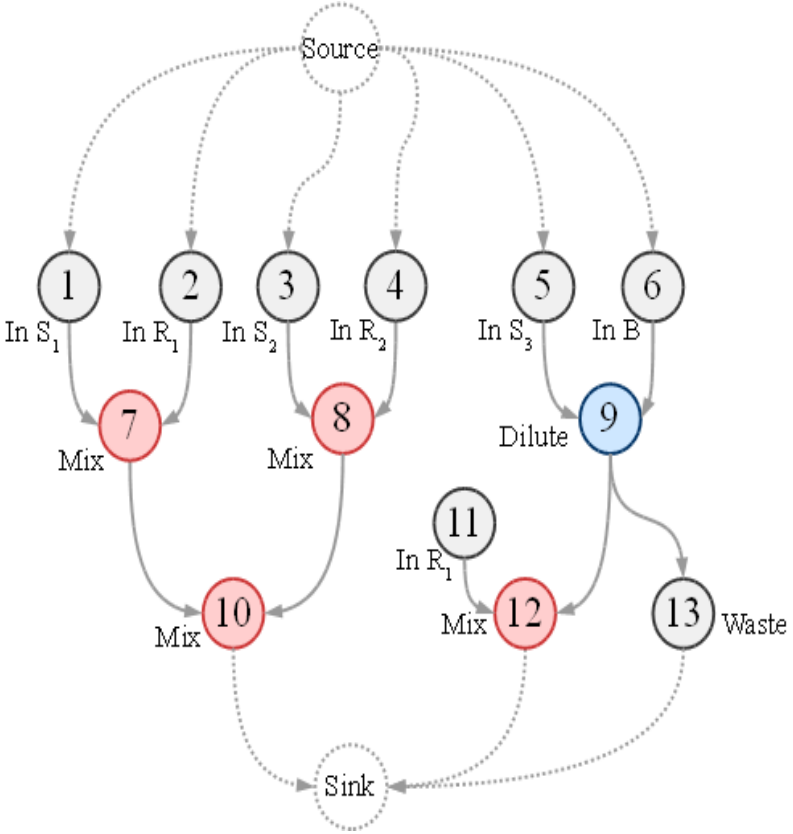


- Disadvantages of modules:
  - Pessimistic segregation area
  - Routing performed post-synthesis

Eliminate the concept of modules: Routing-based synthesis

Routing-Based Synthesis



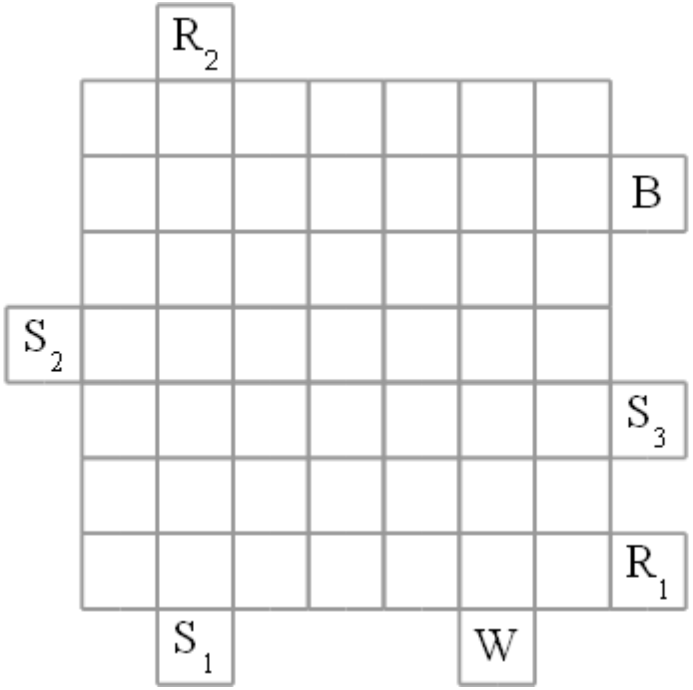


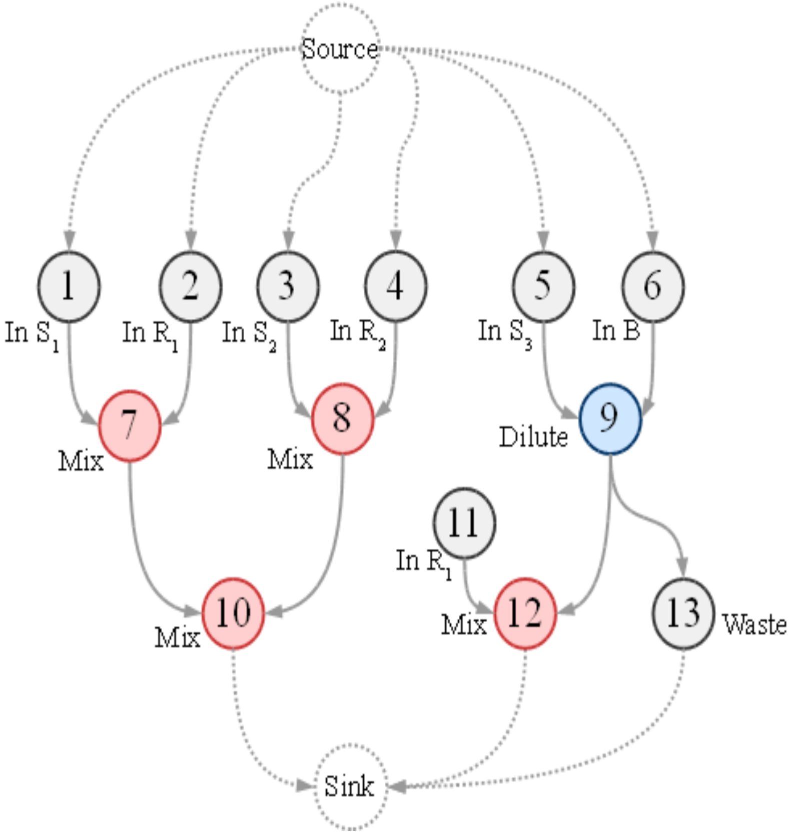
Routing-Based Synthesis

O7

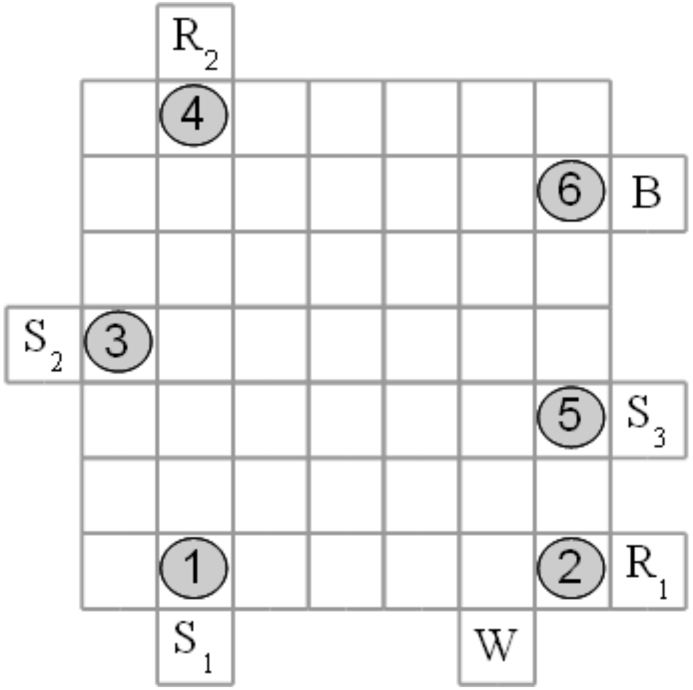
O8

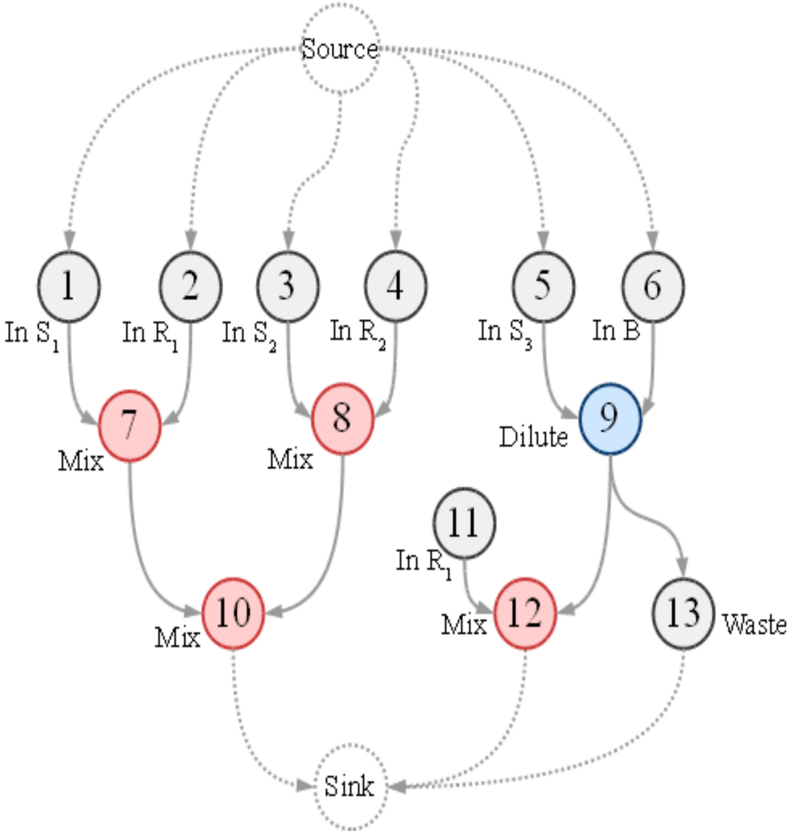
O9



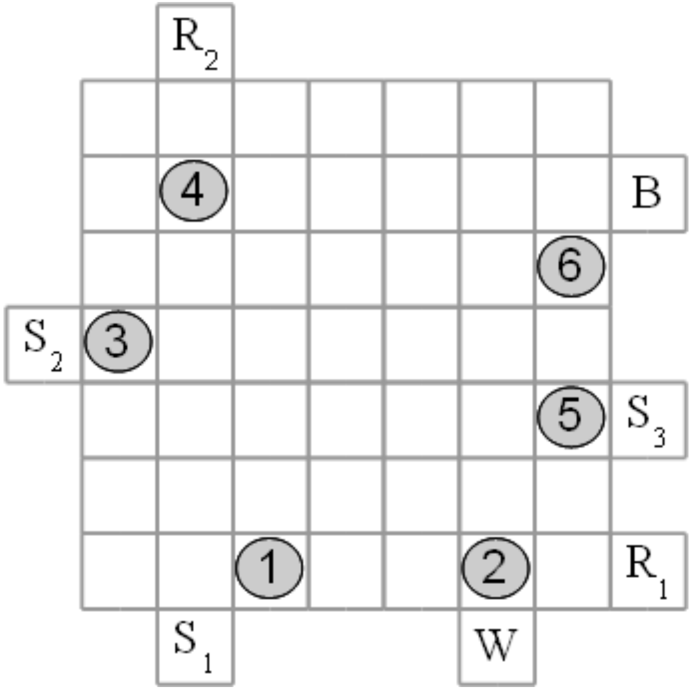


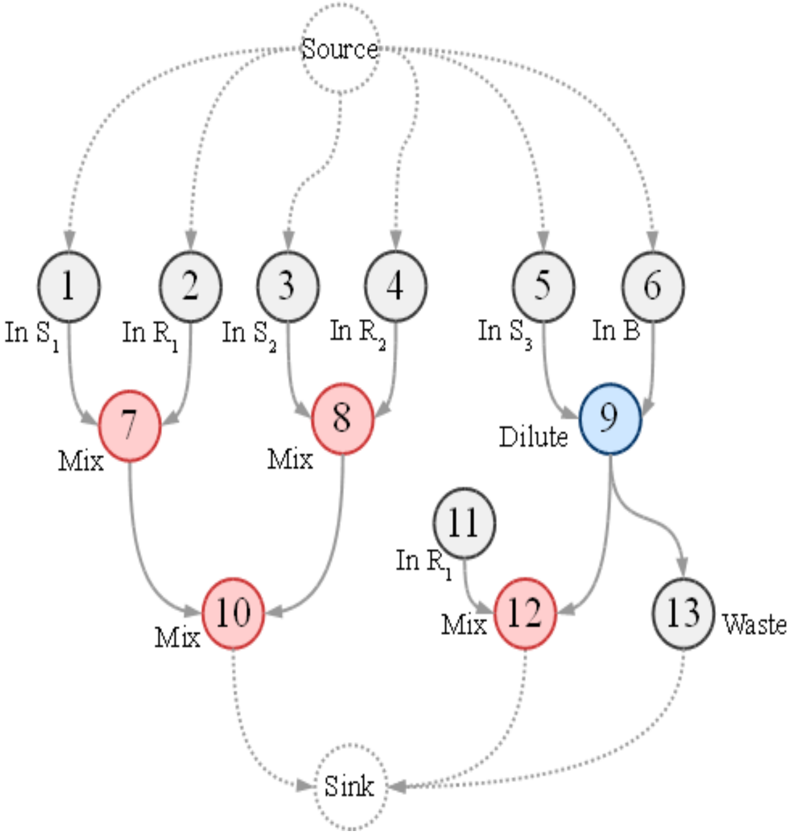
Routing-Based Synthesis



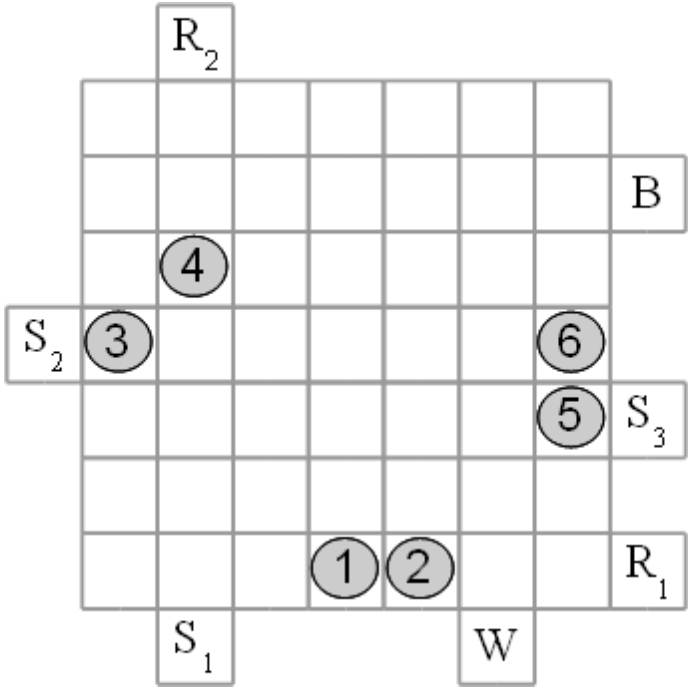


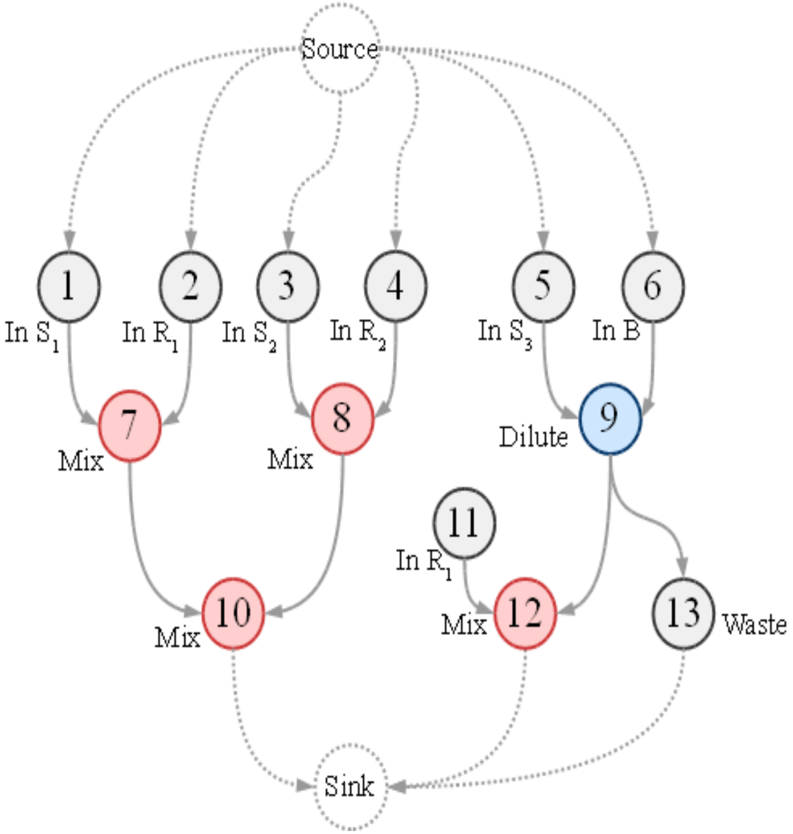
Routing-Based Synthesis



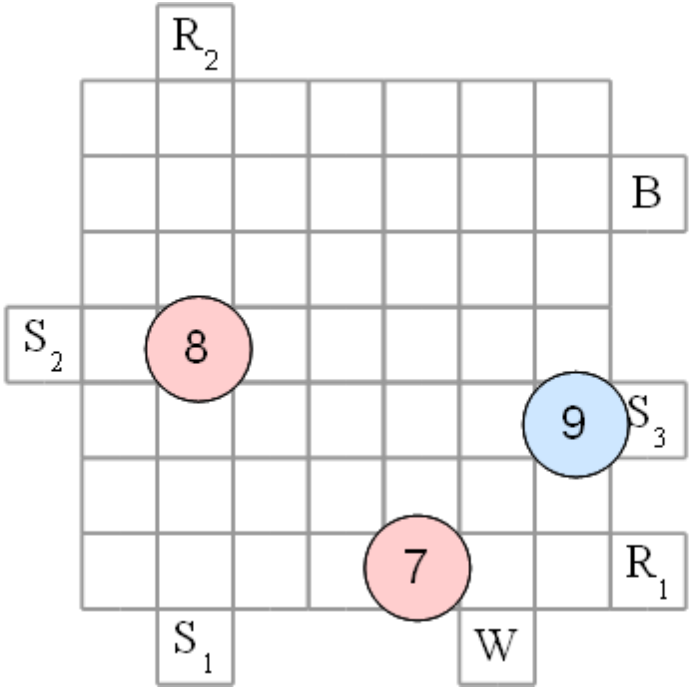


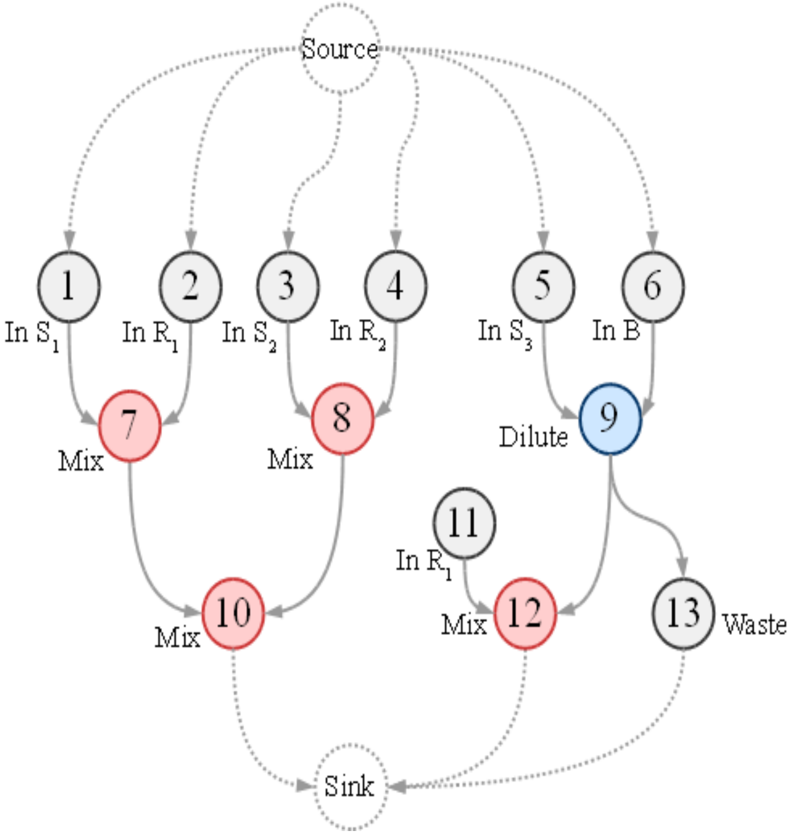
Routing-Based Synthesis



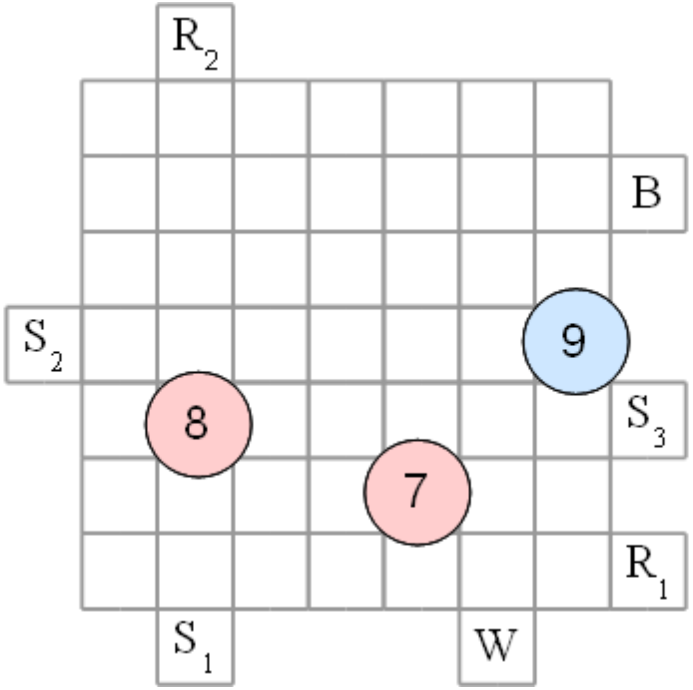


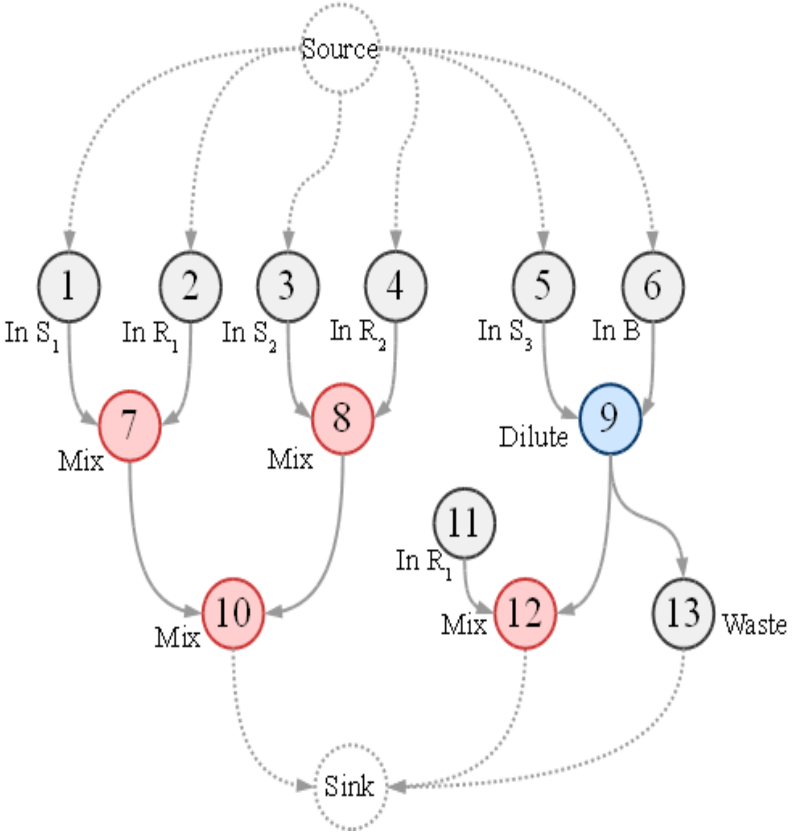
Routing-Based Synthesis



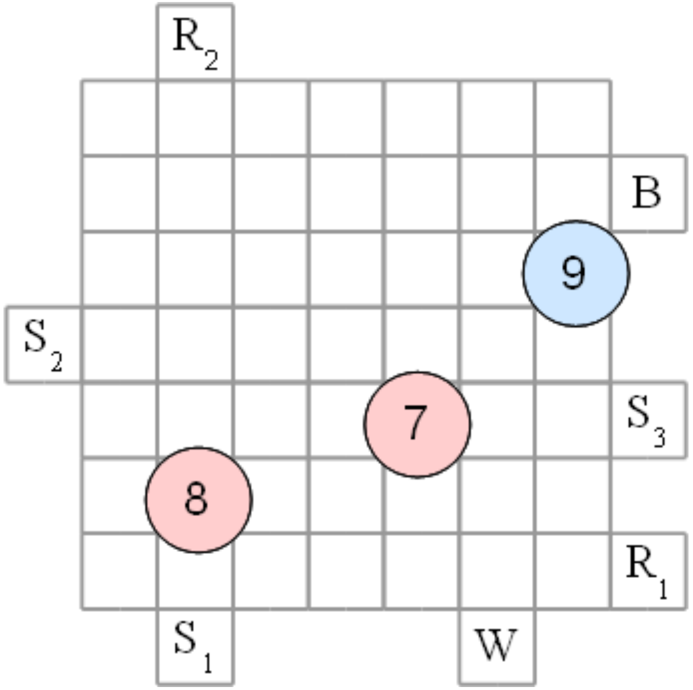


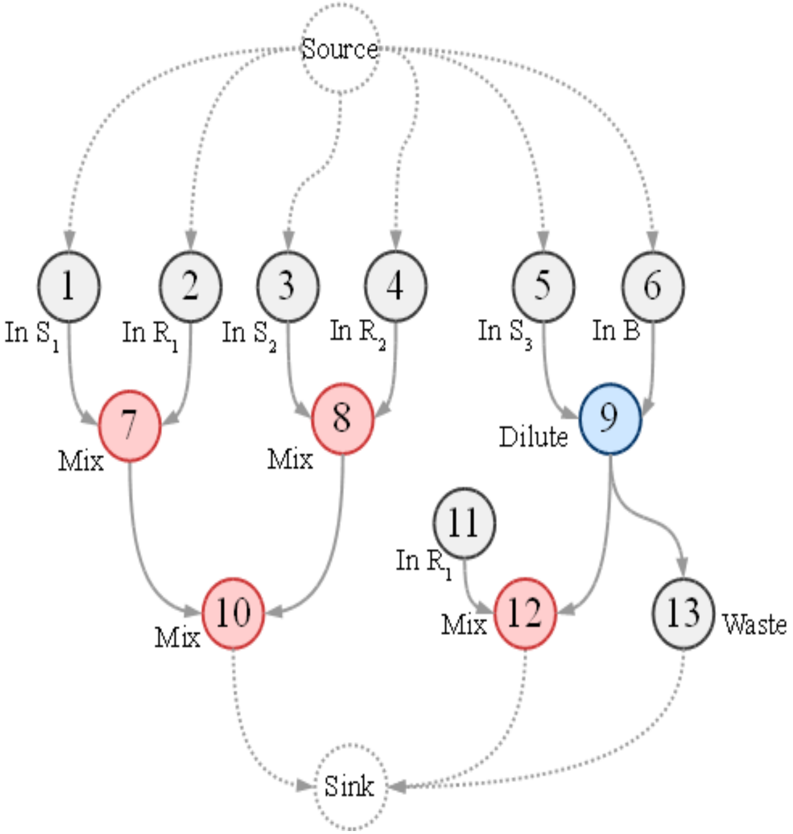
Routing-Based Synthesis



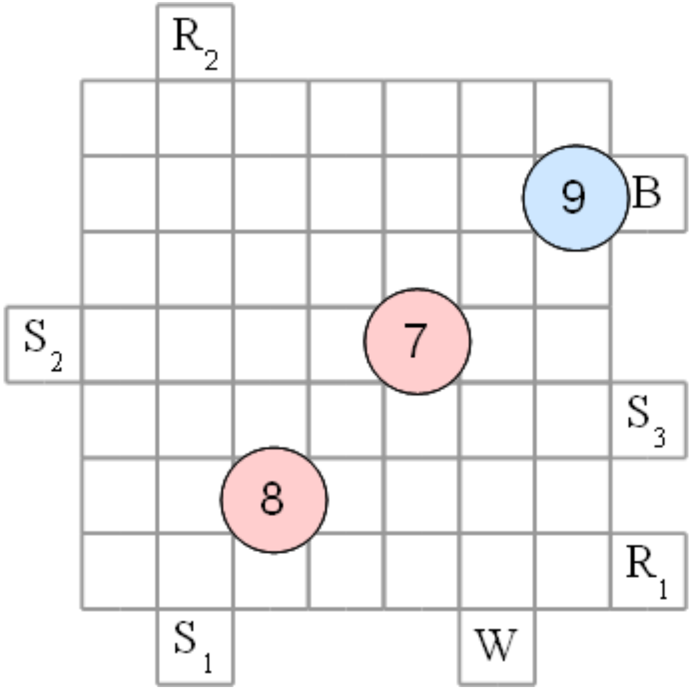


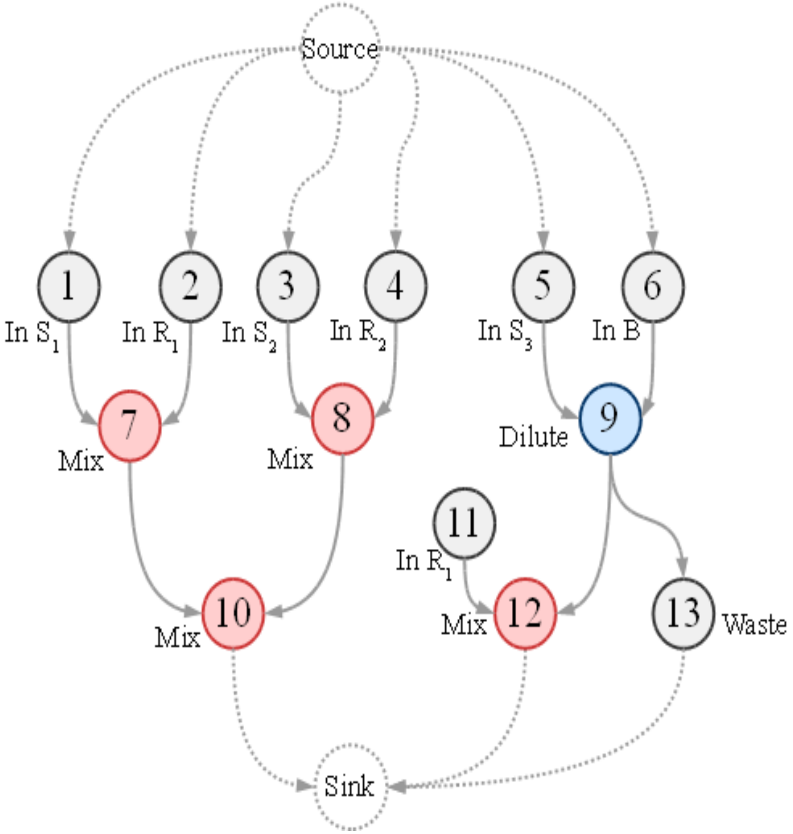
Routing-Based Synthesis



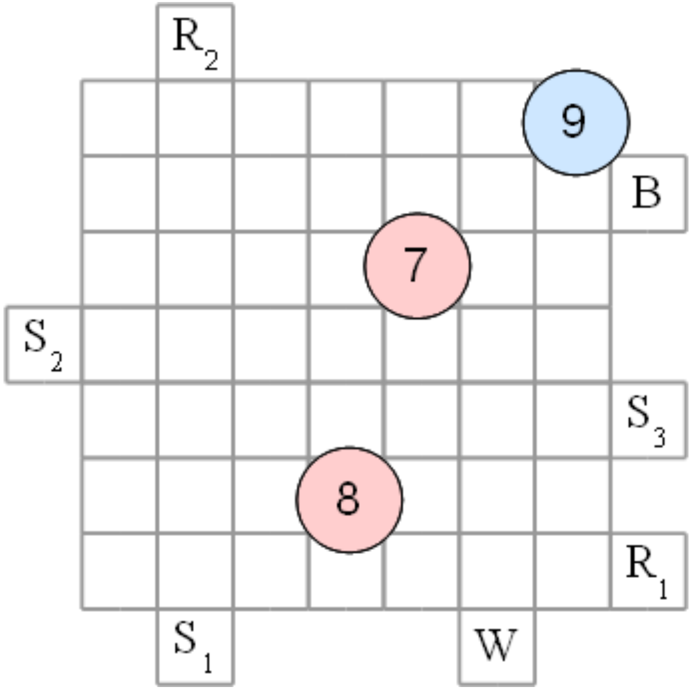


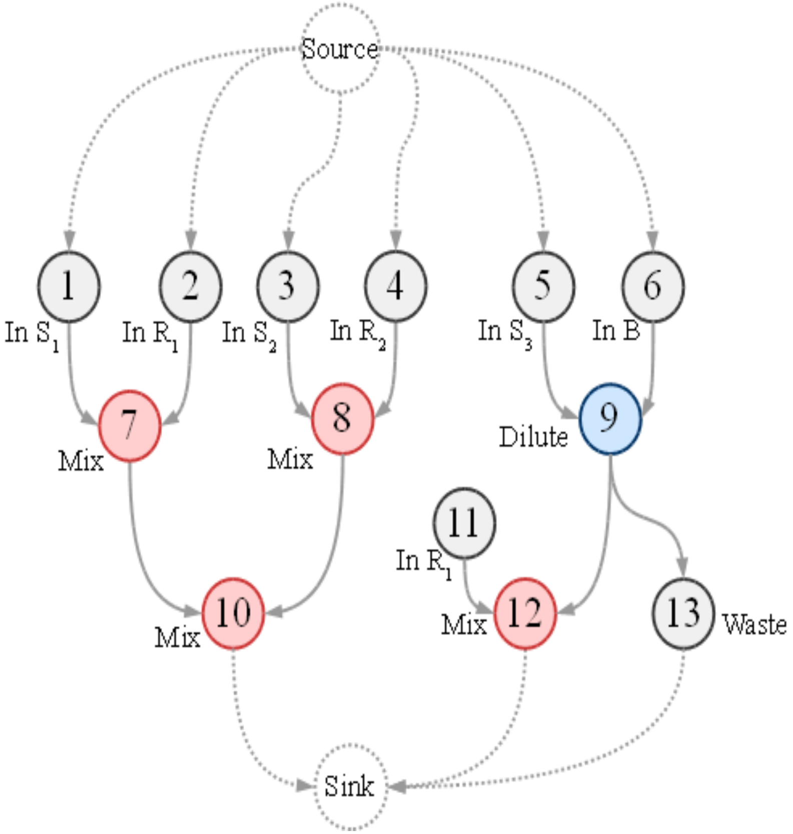
Routing-Based Synthesis



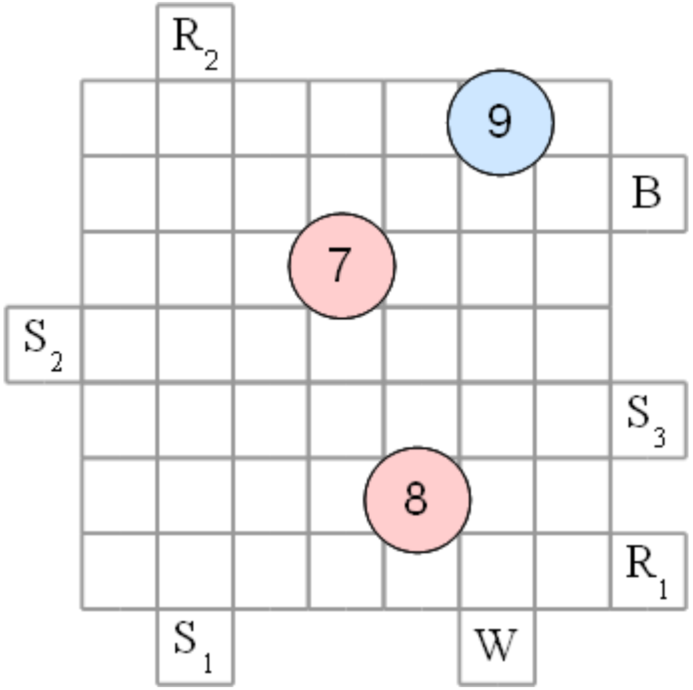


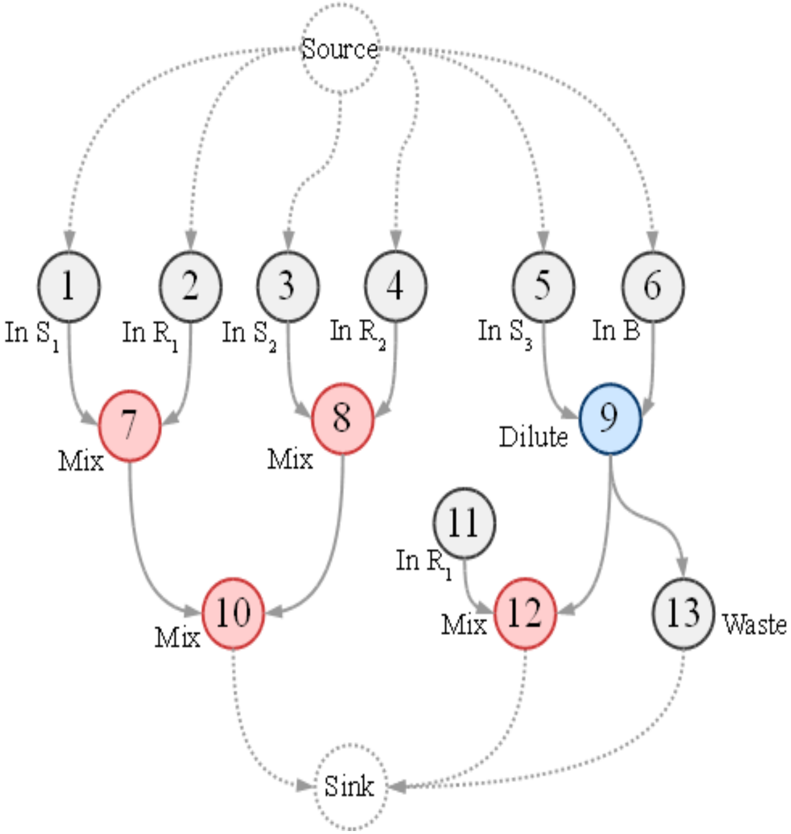
Routing-Based Synthesis



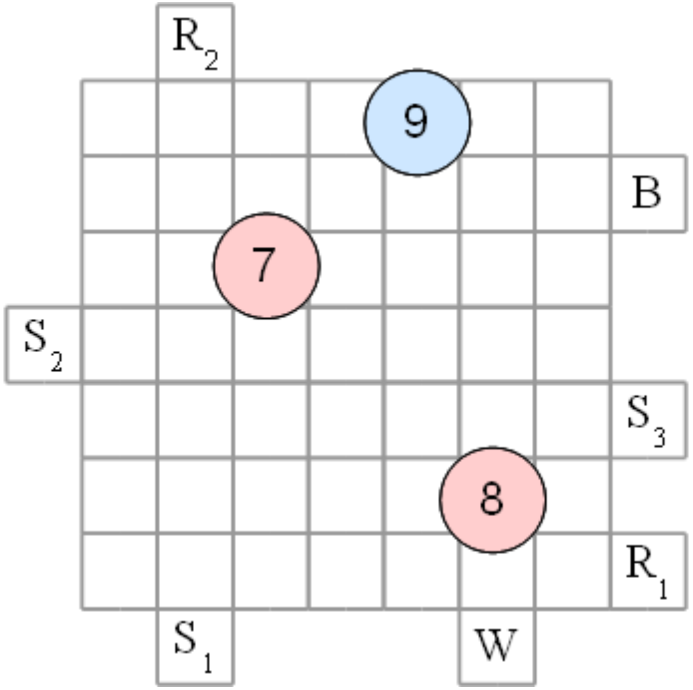


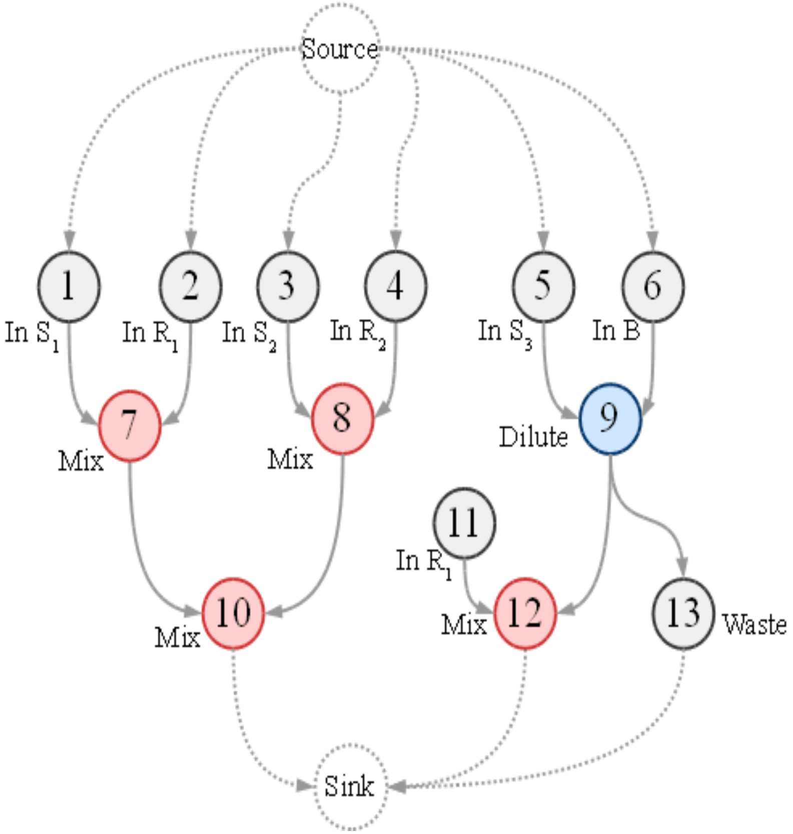
Routing-Based Synthesis



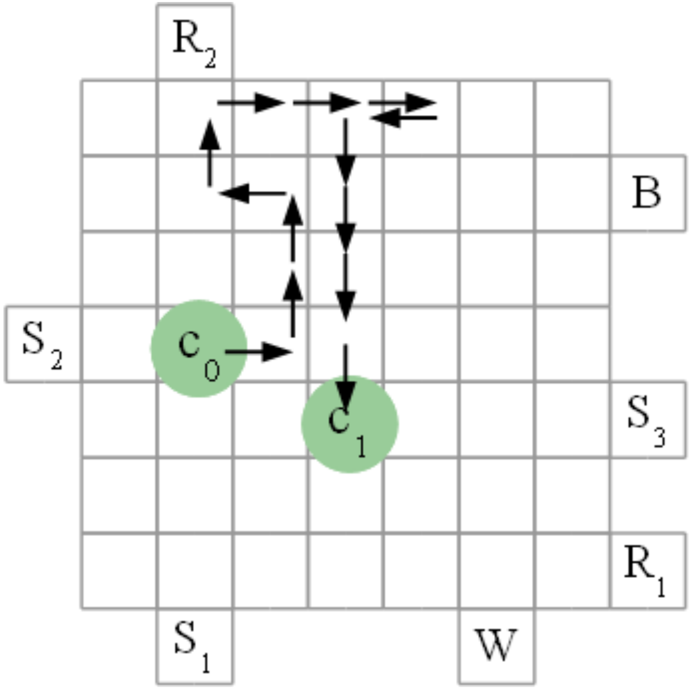


Routing-Based Synthesis





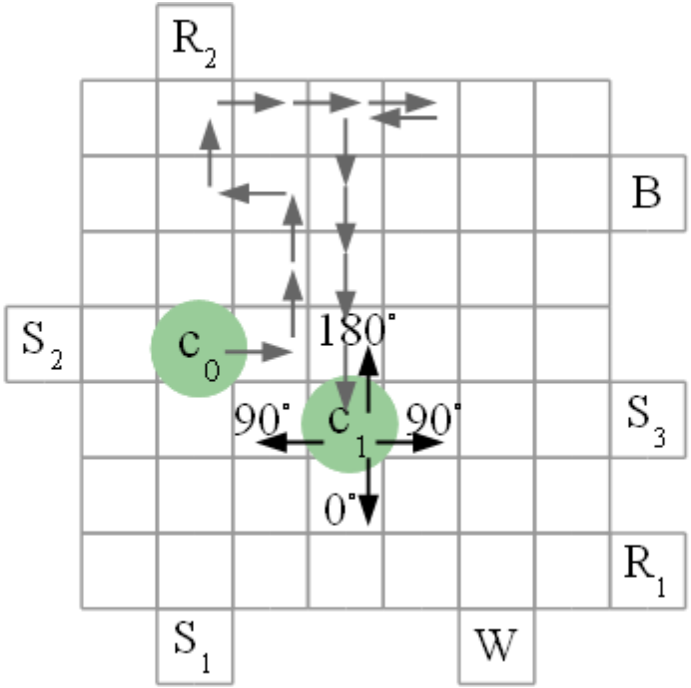
When will the operations complete?



- For module-based synthesis we know the completion time from the module library.

- But now there are no modules, the droplets can move anywhere.
  - How can we find out the operation completion times?

Characterizing operations

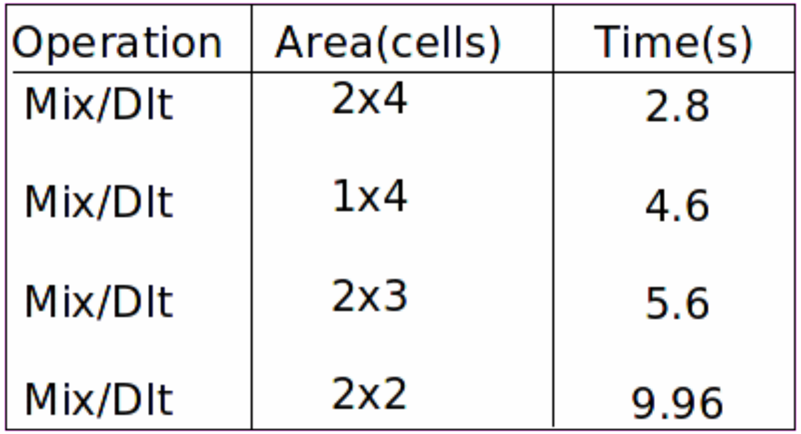


- If the droplet does not move: very slow mixing by diffusion

- If the droplet moves, how long does it take to complete?
- Mixing percentages:

  p^0, p^{90}, p^{180} ?

Characterizing operations



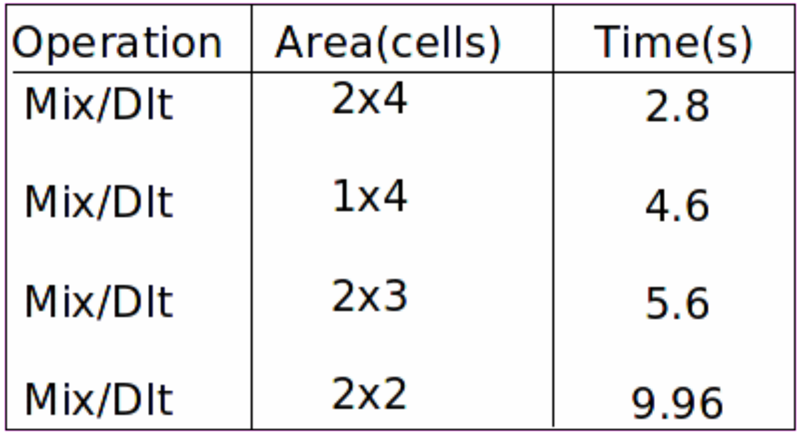
Operation	Area(cells)	Time(s)
Mix/Dlt	2x4	2.8
Mix/Dlt	1x4	4.6
Mix/Dlt	2x3	5.6
Mix/Dlt	2x2	9.96

- We know how long an operation takes on modules
- Starting from this, can determine the percentages?

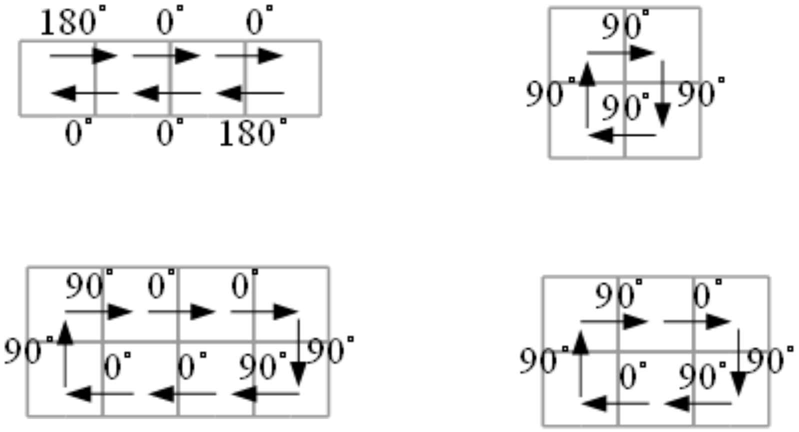
Decomposing modules

Safe, conservative estimates

p^{90} = 0.1%, p^{180} = -0.5%,
p^{0} = 0.29% and 0.58%

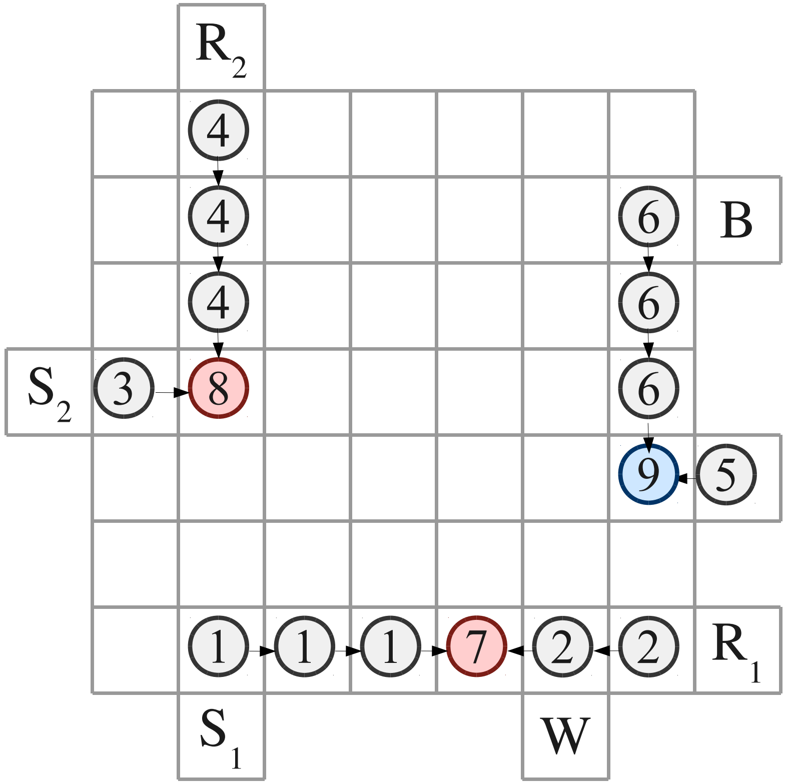


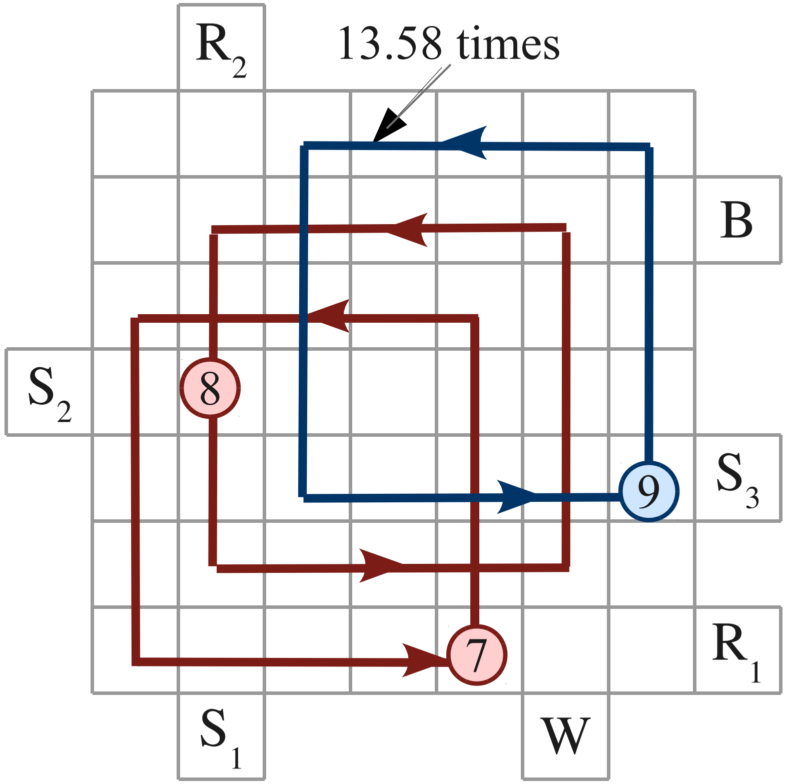
Operation	Area(cells)	Time(s)
Mix/Dlt	2x4	2.8
Mix/Dlt	1x4	4.6
Mix/Dlt	2x3	5.6
Mix/Dlt	2x2	9.96



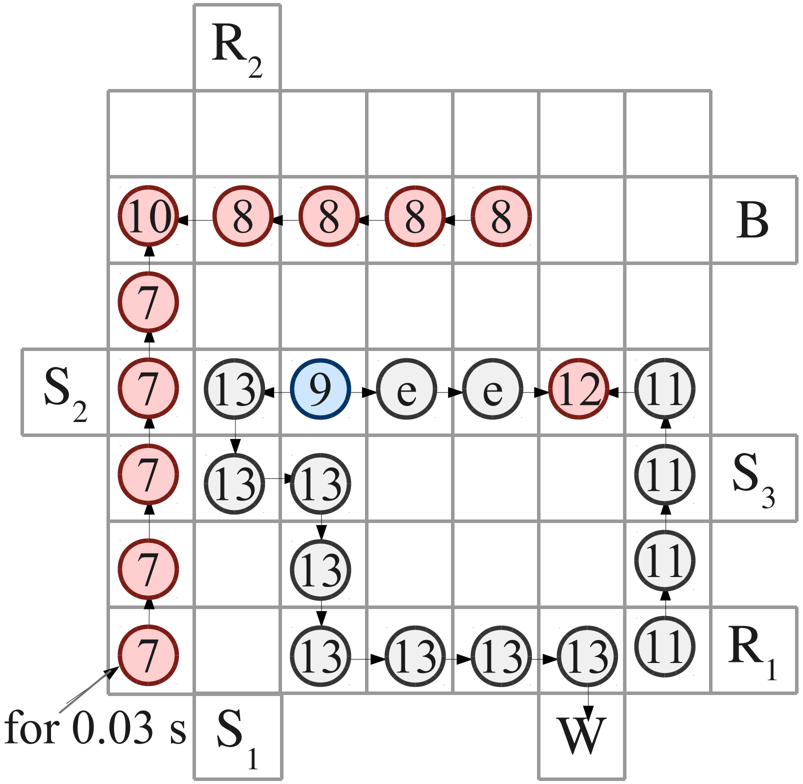
Moving a droplet one cell takes 0.01 s.

Routing-Based Synthesis

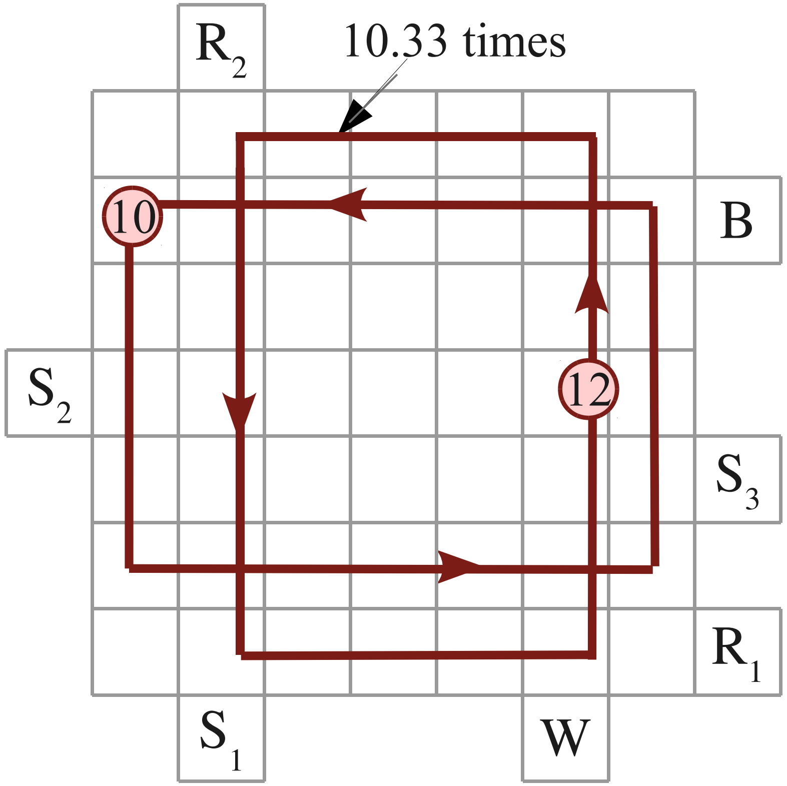




Routing-Based Synthesis



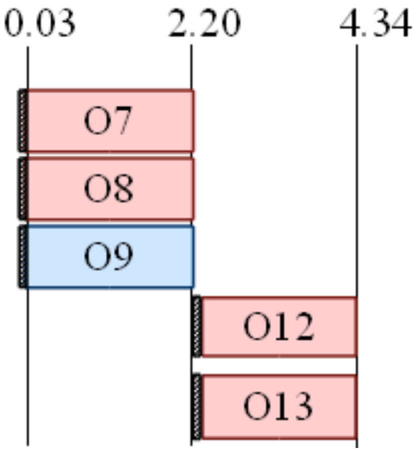
Waits for 0.03 s



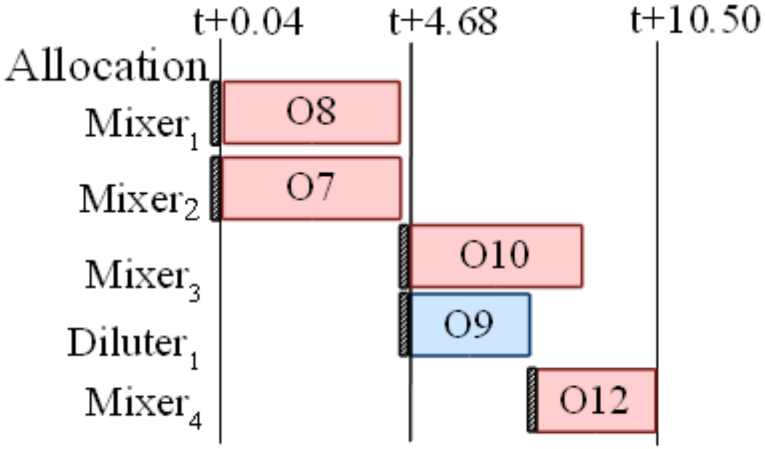
10.33 times

Routing- vs. Module-Based Synthesis

Routing-Based Synthesis



Module-Based Synthesis



References:

1. Elena Maftei, Paul Pop, Jan Madsen, Routing-Based Synthesis of Digital Microfluidic Biochips. Proceedings of the Compilers, Architecture, and Synthesis for Embedded Systems Conference (CASES'10), pp. 41-49, 2010 (**best paper candidate**)

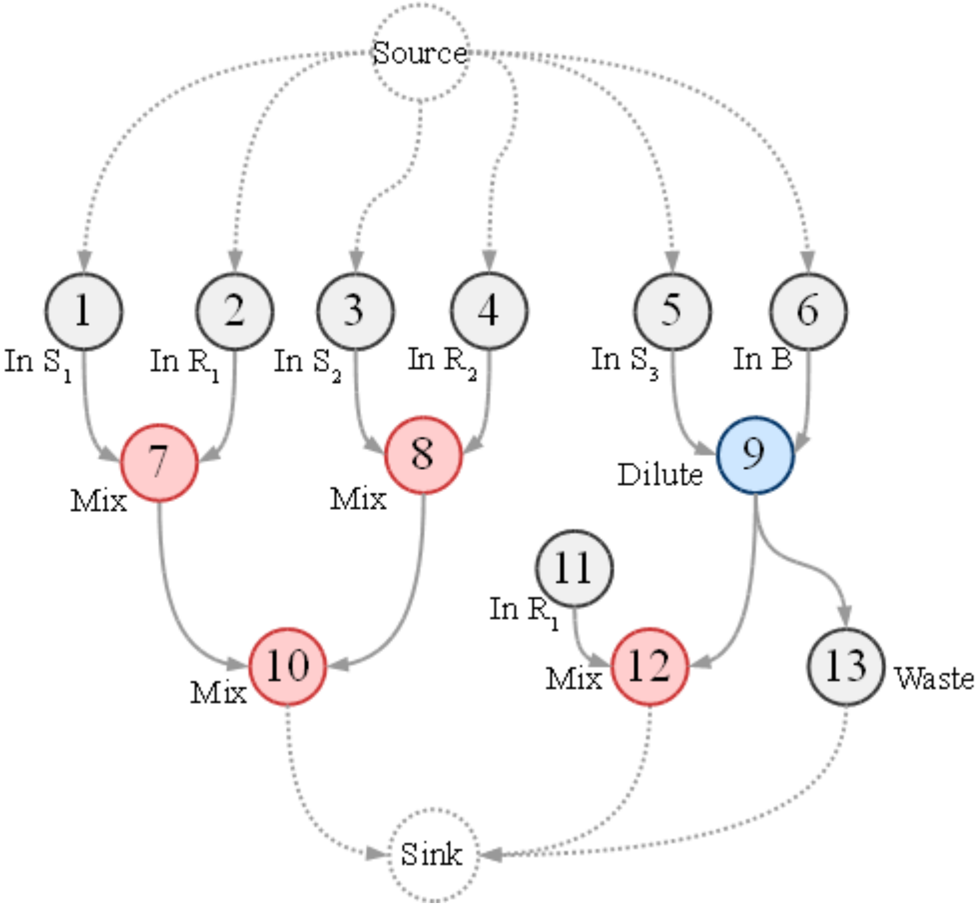
PART 2: DIGITAL MICROFLUIDIC BIOCHIPS

ROUTING-BASED SYNTHESIS ALGORITHM

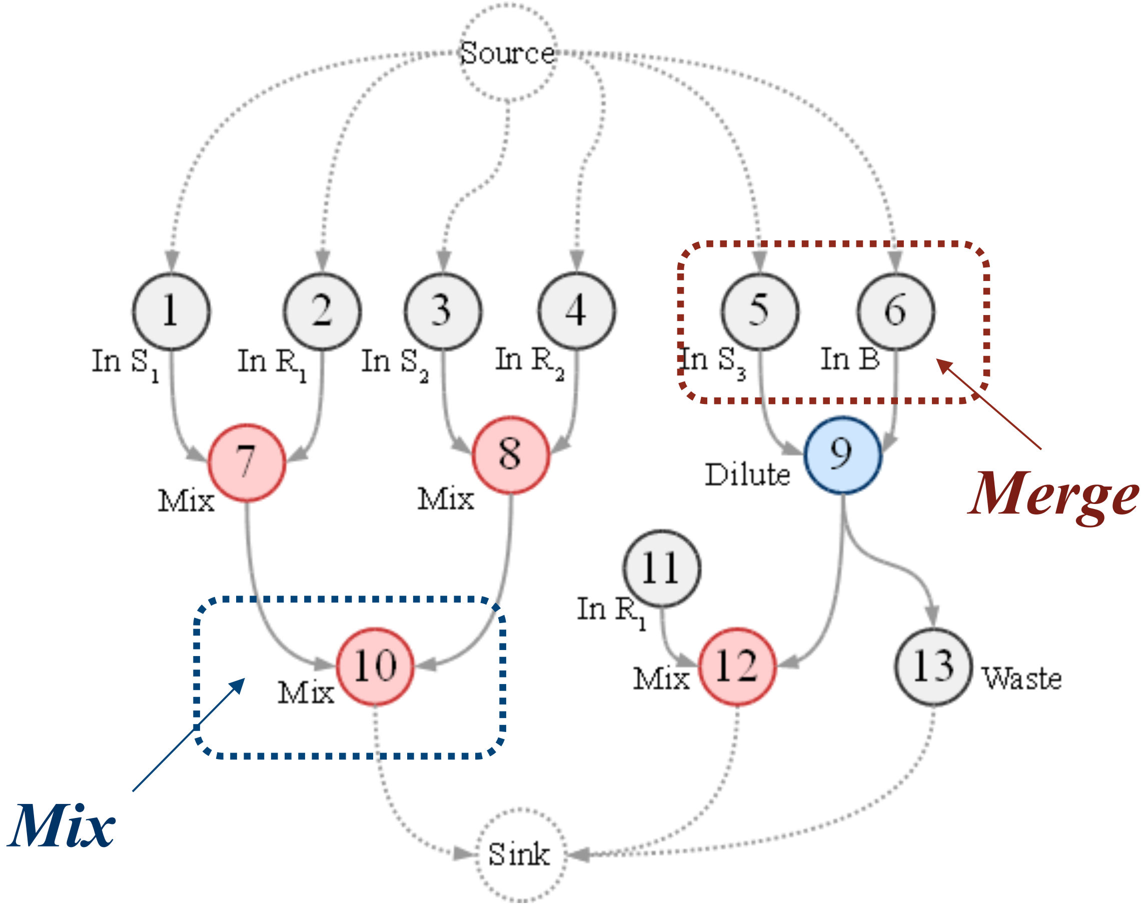
Problem Formulation

- Input
  - Sequencing graph
  - Library of modules
  - Area constraint
- Output
  - Implementation which minimizes application execution time
    - Allocation of modules from modules library
    - Binding of modules to operations in sequencing graph
    - Scheduling of operations
    - Routes of the droplets

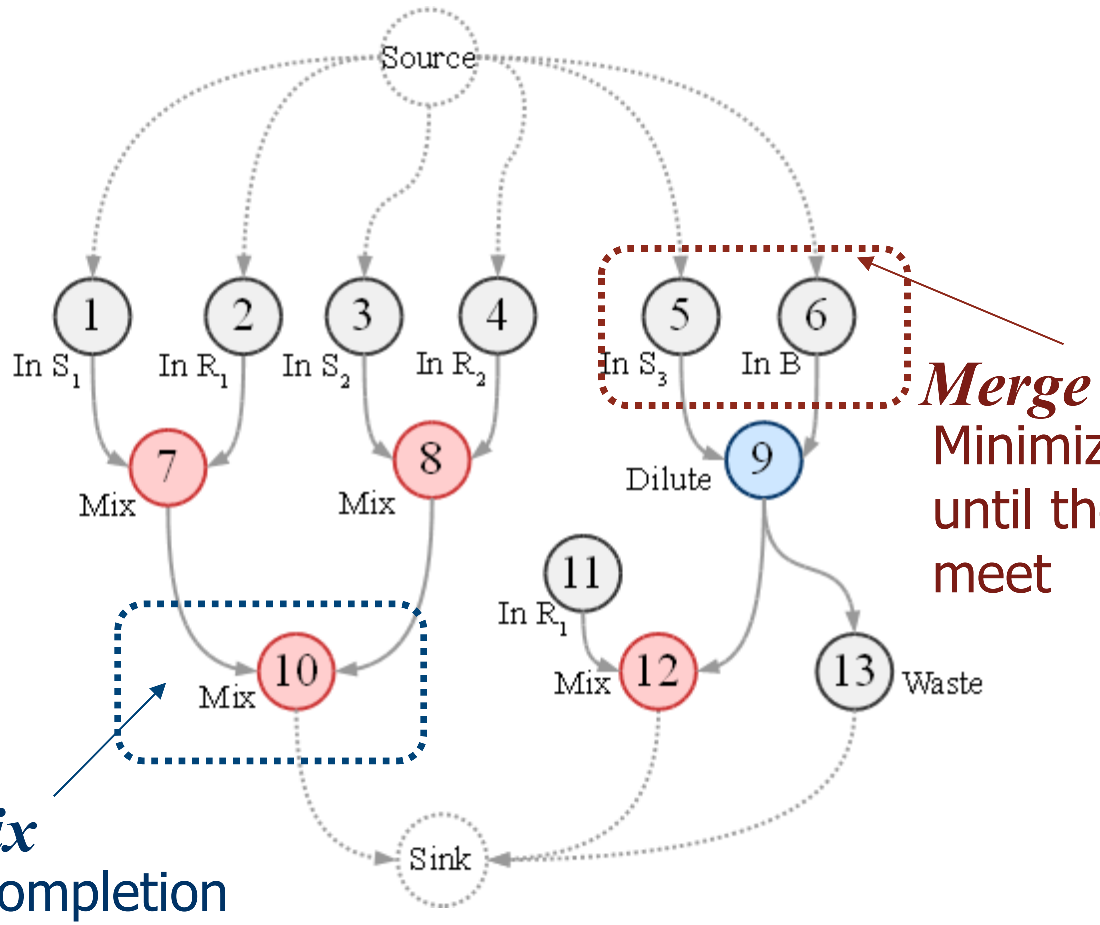
Proposed Solution



Proposed Solution



Proposed Solution



Merge

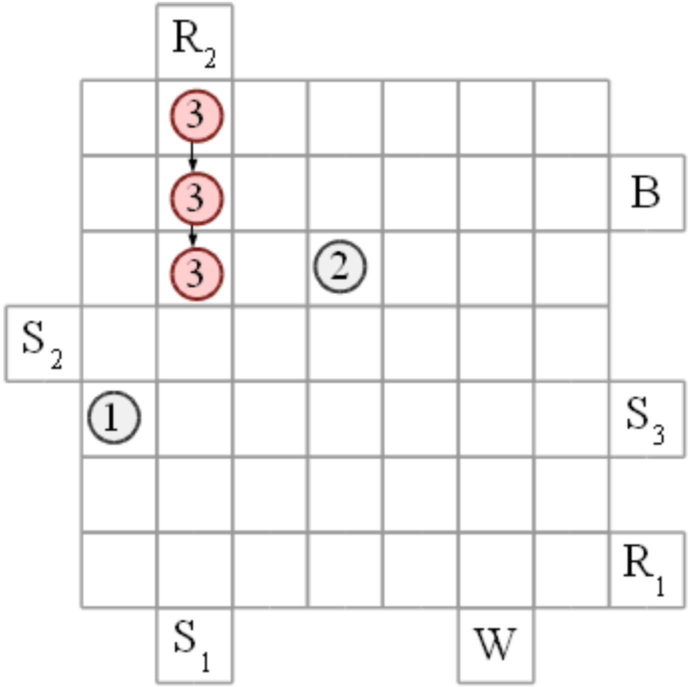
Minimize the time until the droplets meet

Mix

Minimize the completion time for the operation

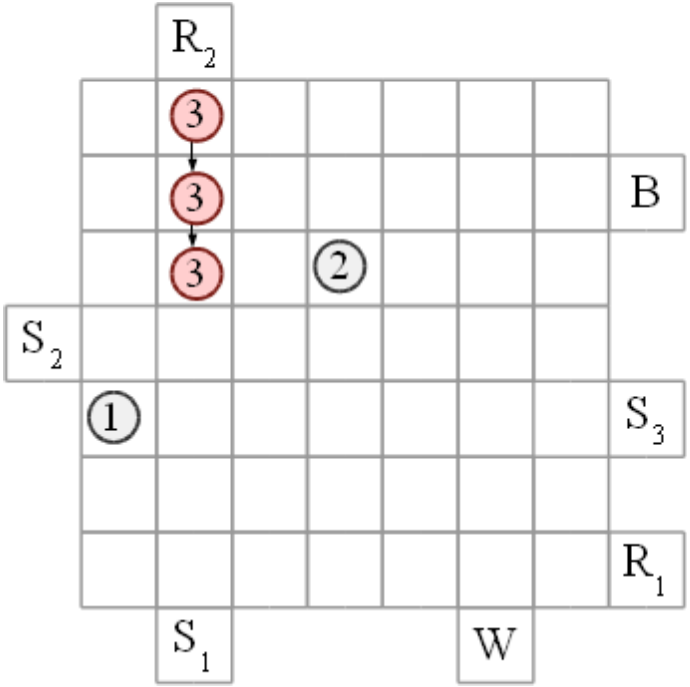
GRASP-Based Synthesis

- Greedy Randomized Adaptive Search Procedure



GRASP-Based Synthesis

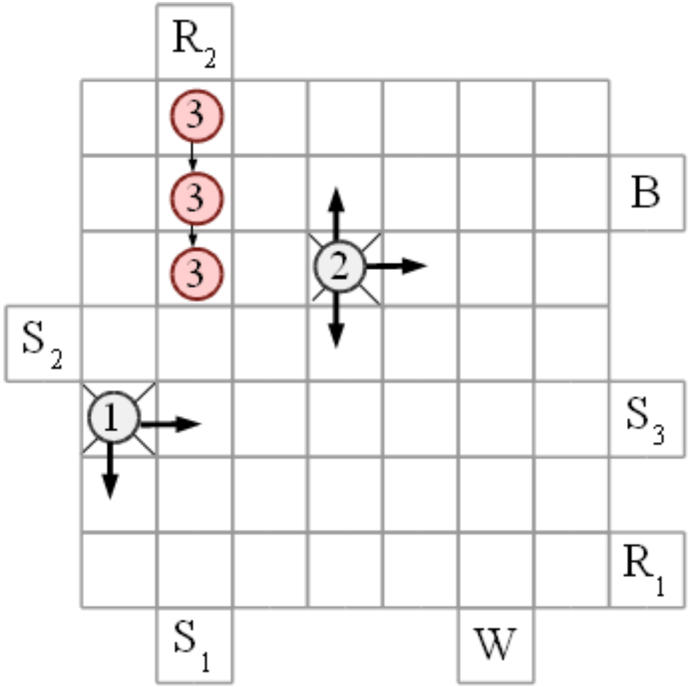
- Greedy Randomized Adaptive Search Procedure



- For each droplet:
  - Determine possible moves
  - Evaluate possible moves
  - Make a list of best N possible moves
  - Perform a randomly chosen possible move from N

GRASP-Based Synthesis

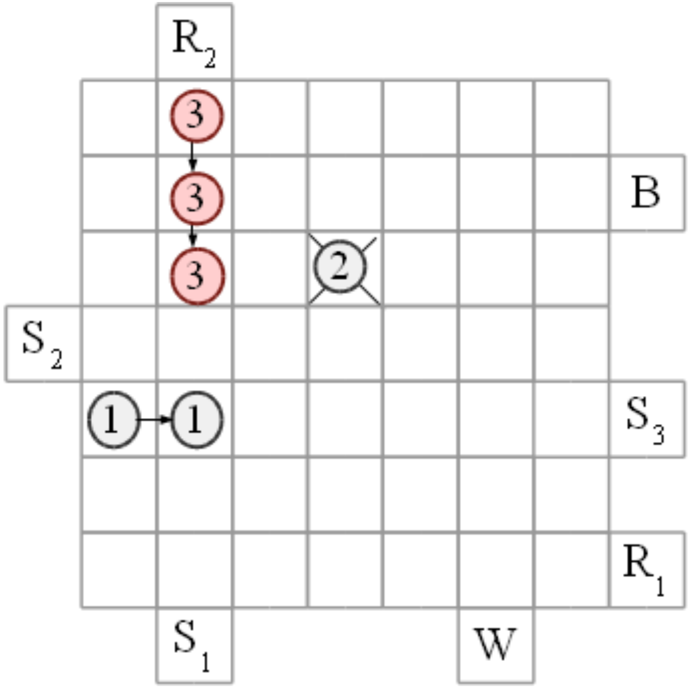
- Greedy Randomized Adaptive Search Procedure



- For each droplet:
  - Determine possible moves
  - Evaluate possible moves
  - Make a list of best N possible moves
  - Perform a randomly chosen possible move from N

GRASP-Based Synthesis

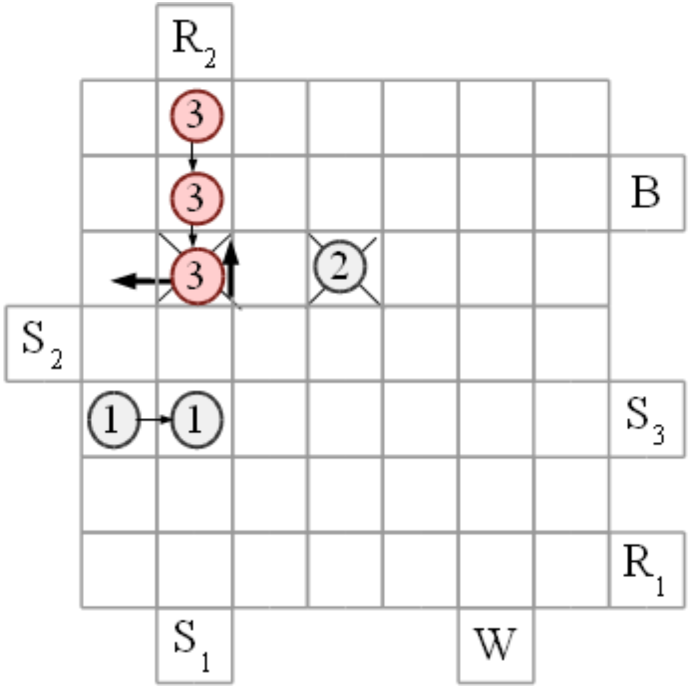
- Greedy Randomized Adaptive Search Procedure



- For each droplet:
  - Determine possible moves
  - Evaluate possible moves
  - Make a list of best N possible moves
  - Perform a randomly chosen possible move from N

GRASP-Based Synthesis

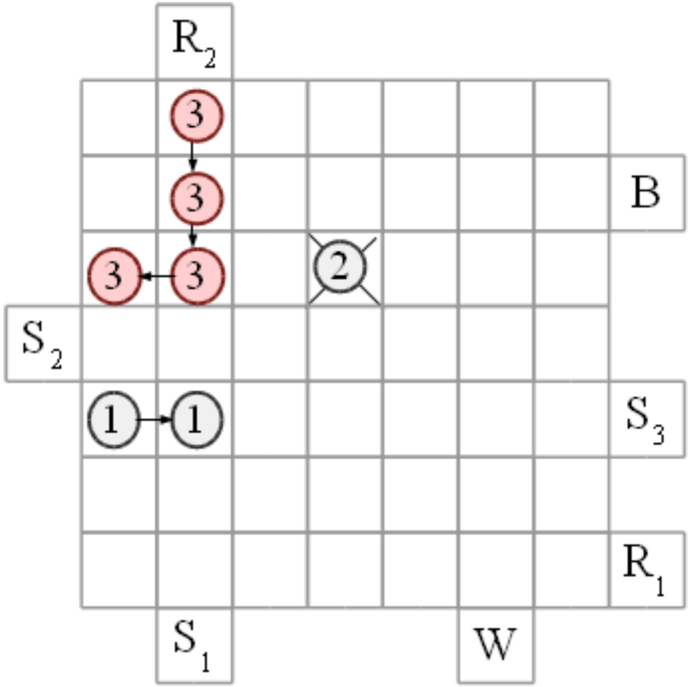
- Greedy Randomized Adaptive Search Procedure



- For each droplet:
  - Determine possible moves
  - Evaluate possible moves
  - Make a list of best N possible moves
  - Perform a randomly chosen possible move from N

GRASP-Based Synthesis

- Greedy Randomized Adaptive Search Procedure



- For each droplet:
  - Determine possible moves
  - Evaluate possible moves
  - Make a list of best N possible moves
  - Perform a randomly chosen possible move from N

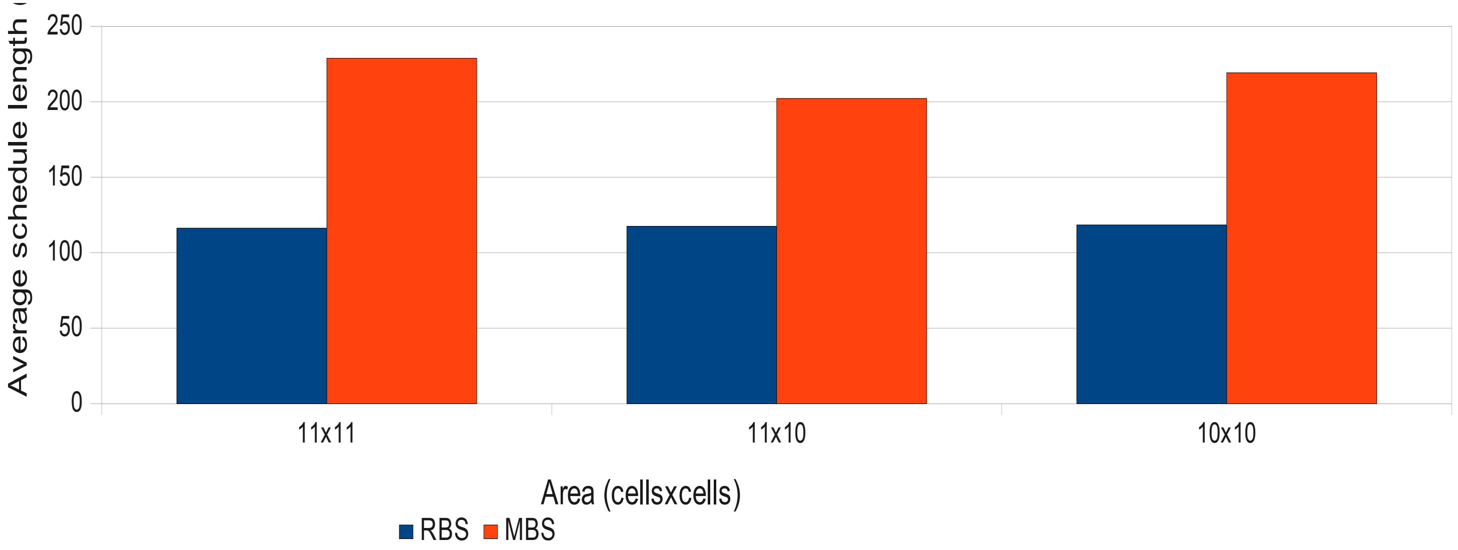
Experimental Evaluation

- GRASP algorithm implemented in Java
- Improvement brought by Routing-Based Synthesis (RBS) compared to Module-Based Synthesis (MBS)
  - Two real-life applications
  - Ten synthetic bechmarks

Experimental Results

Improvement of RBS vs. MBS in schedule length

Colorimetric protein assay



Conclusions

- Characteristics of digital microfluidic biochips resembles those of digital circuits
- It is possible to use techniques and methods from MPSoC to design and analyze biochips, i.e., a module-based approach
- Eliminating the concept of "virtual modules", it is possible to have a routing-based synthesis approach
- The routing-based synthesis leads to significant improvements compared to module-based synthesis

- Can we use similar methods to address the flow-based biochips?

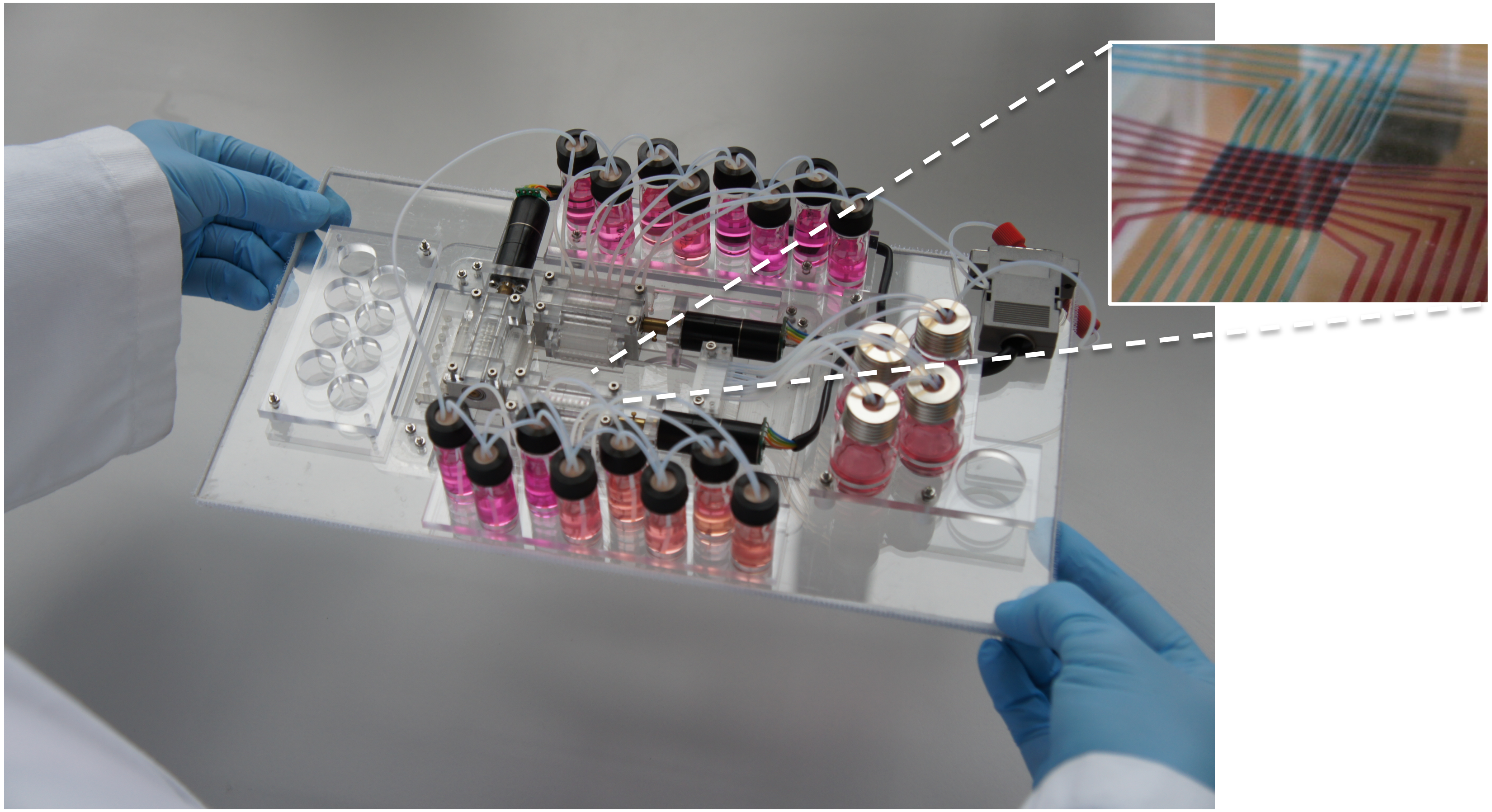
References:

1. Wajid Hassan Minhass, Paul Pop, Jan Madsen, System-Level Modeling and Synthesis of Flow-Based Microfluidic Biochips. Proceedings of the Compilers, Architecture, and Synthesis for Embedded Systems Conference (CASES’11), 2011

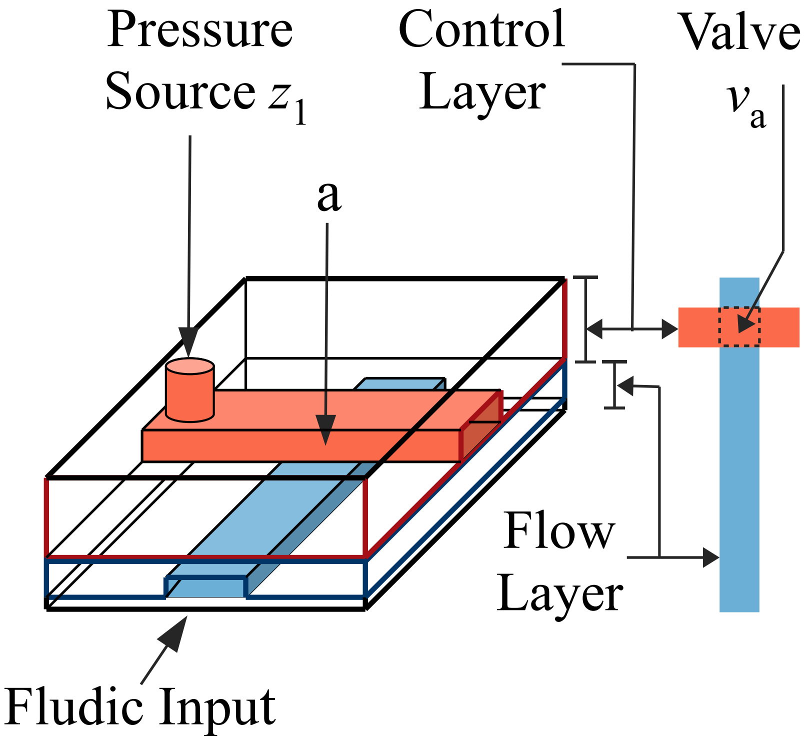
PART 2: FLOW-BASED MICROFLUIDIC BIOCHIPS

BASIC ARCHITECTURE AND COMPONENTS

Flow-Based Microfluidic Biochips

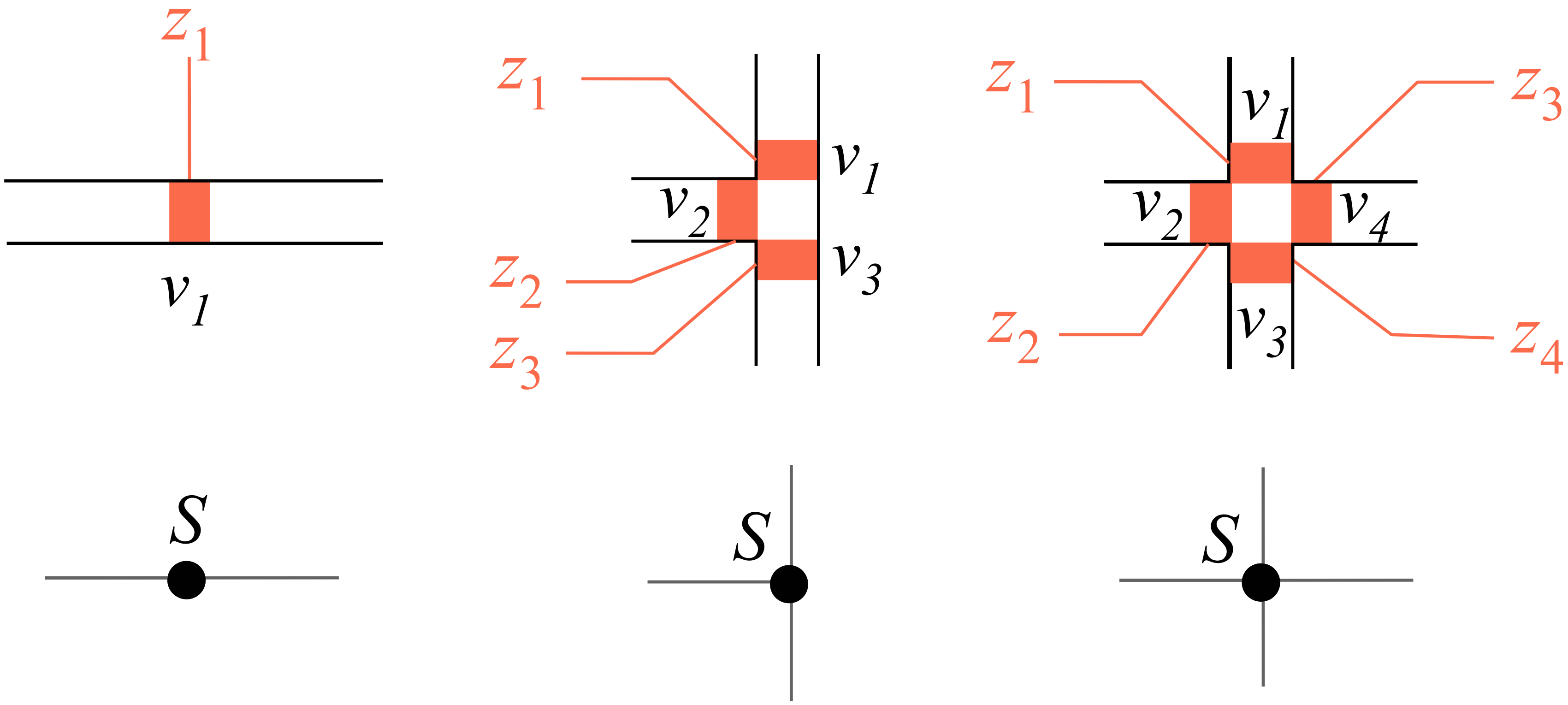


Flow-Based Biochip Components

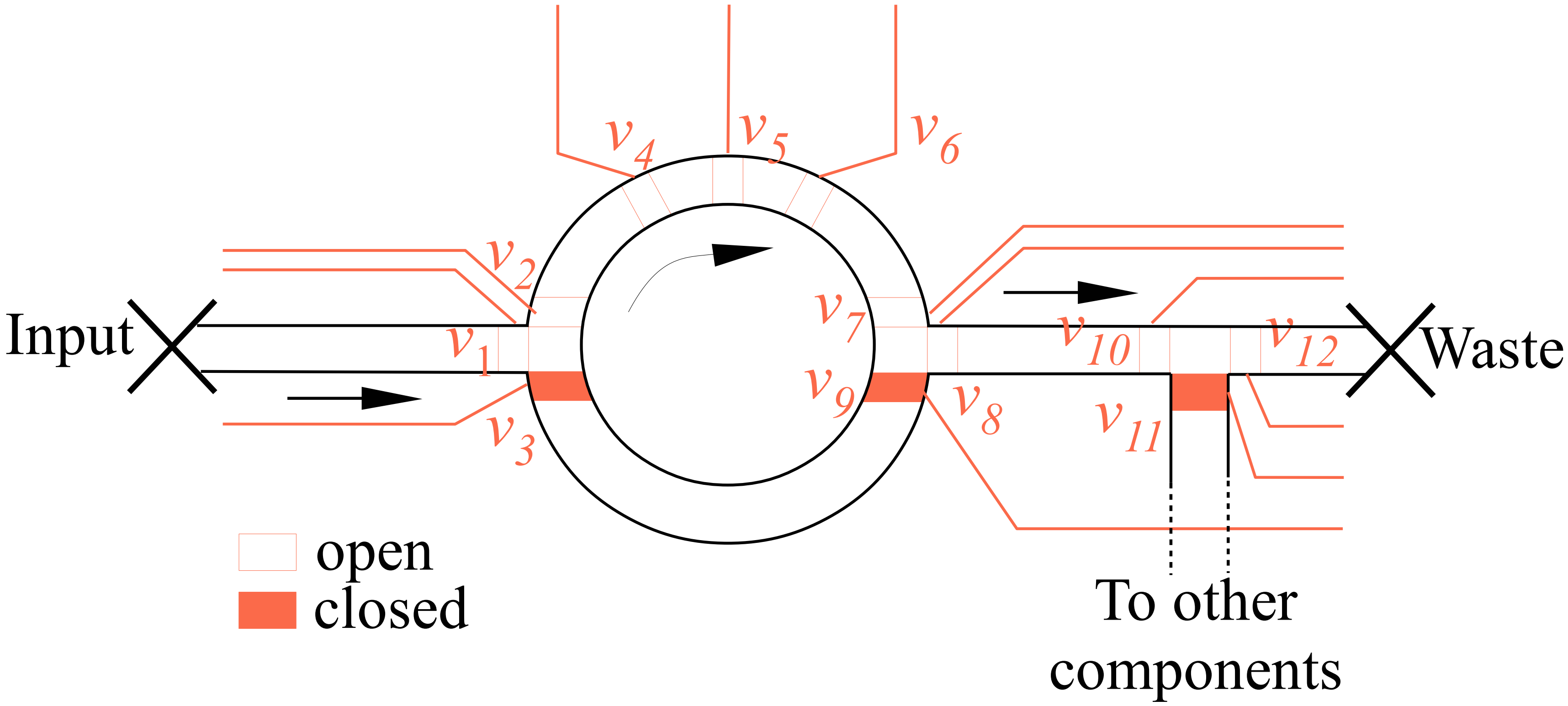


Microfluidic valve

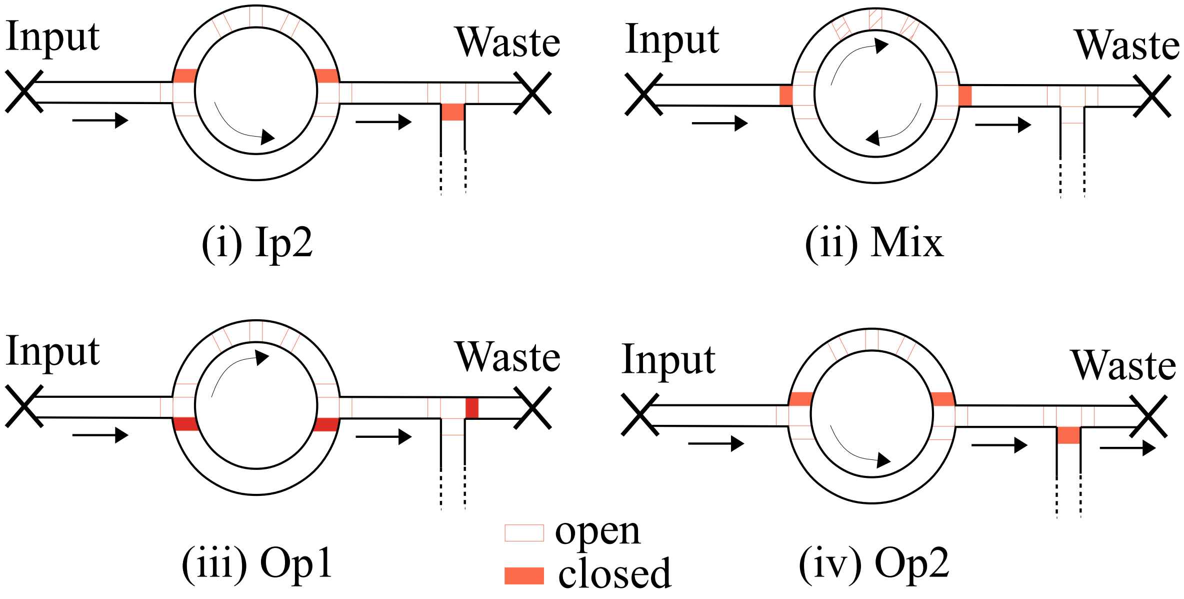
Switch Configurations



Microfluidic Mixer



Microfluidic Mixer: Operational Phases



(i) Ip2

(ii) Mix

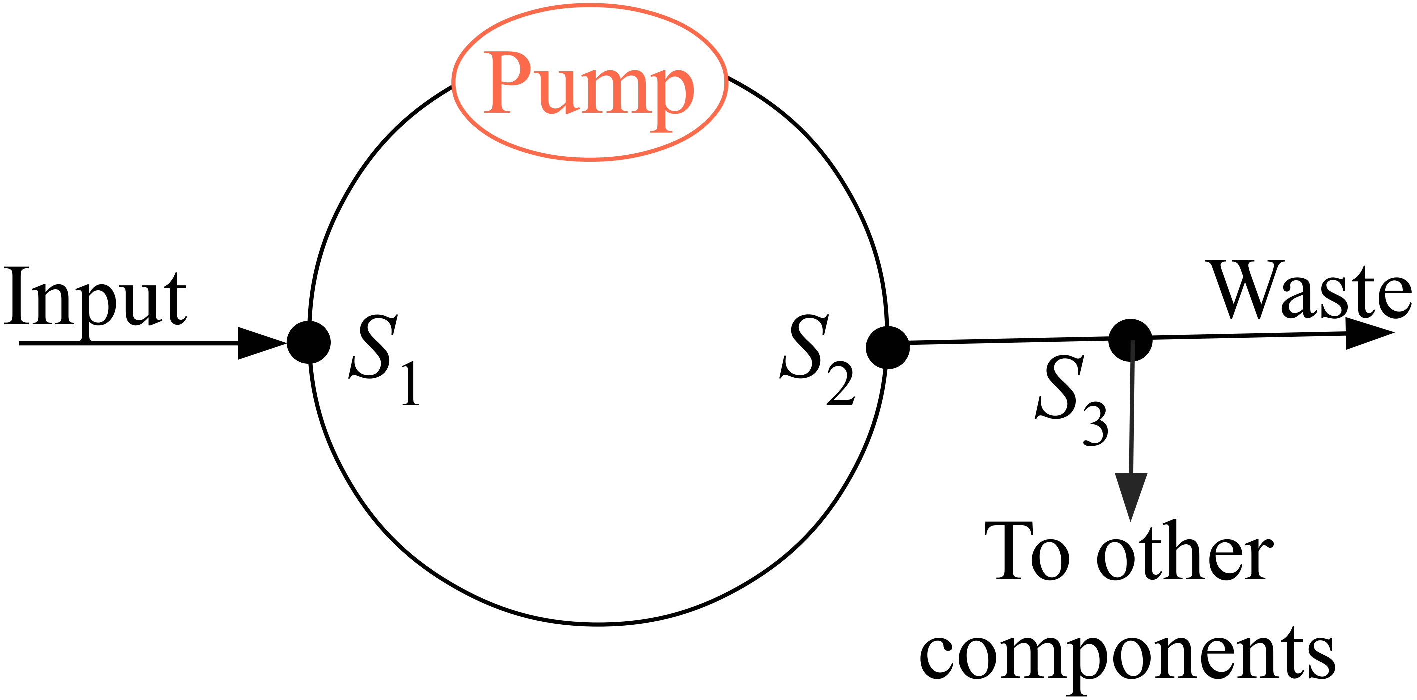
(iii) Op1

(iv) Op2

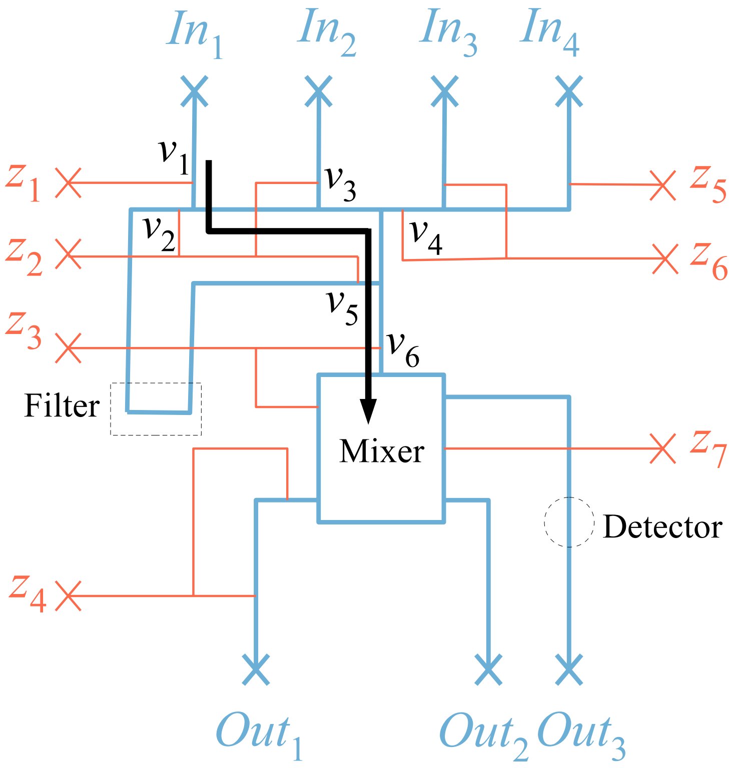
Table 1: Mixer: Control Layer Model

Phase	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9
1. Ip1	0	0	1	0	0	0	0	0	1
2. Ip2	0	1	0	0	0	0	1	0	0
3. Mix	1	0	0	Mix	Mix	Mix	0	1	0
4. Op1	0	0	1	0	0	0	0	0	1
5. Op2	0	1	0	0	0	0	1	0	0

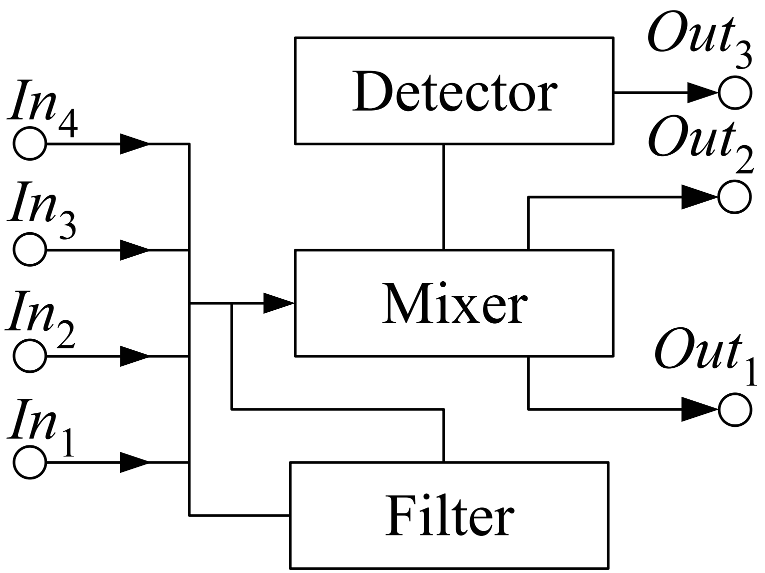
Microfluidic Mixer: Conceptual View



Flow-Based Biochip Architecture



Schematic view



Functional view

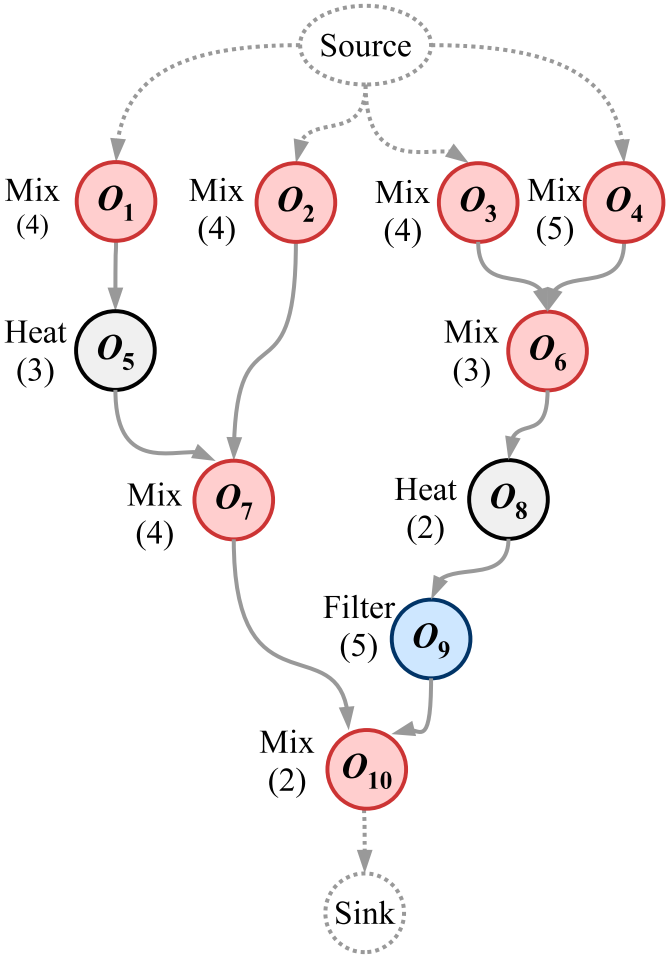
References:

1. Wajid Hassan Minhass, Paul Pop, Jan Madsen, System-Level Modeling and Synthesis of Flow-Based Microfluidic Biochips. Proceedings of the Compilers, Architecture, and Synthesis for Embedded Systems Conference (CASES'11), 2011

PART 2: FLOW-BASED MICROFLUIDIC BIOCHIPS

BIOCHIP SYNTHESIS

Application and Platform Models



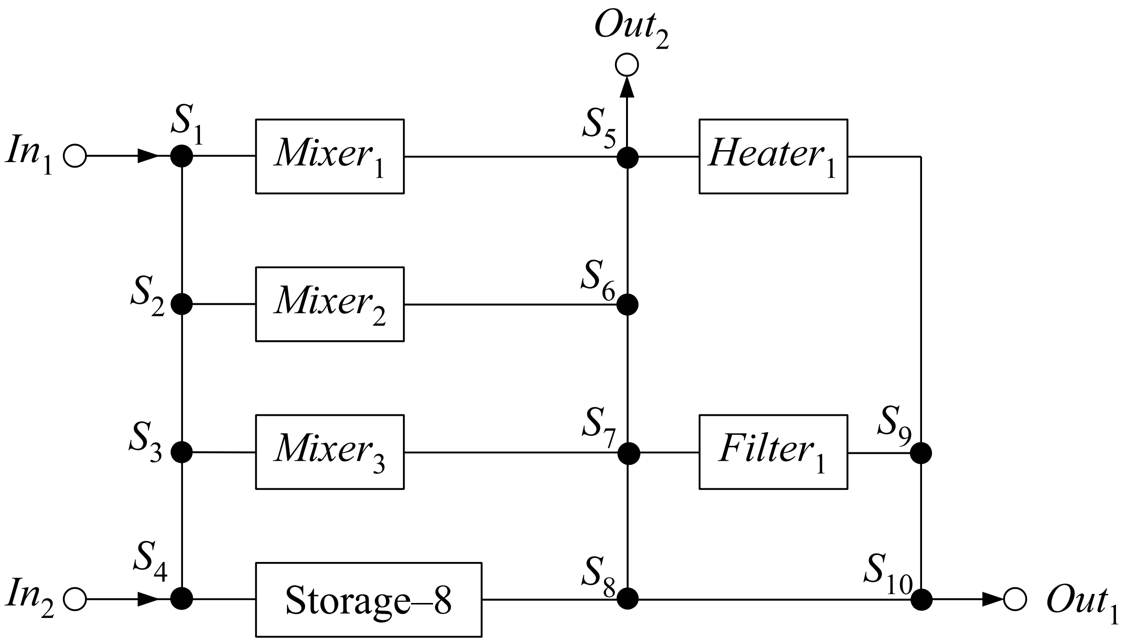
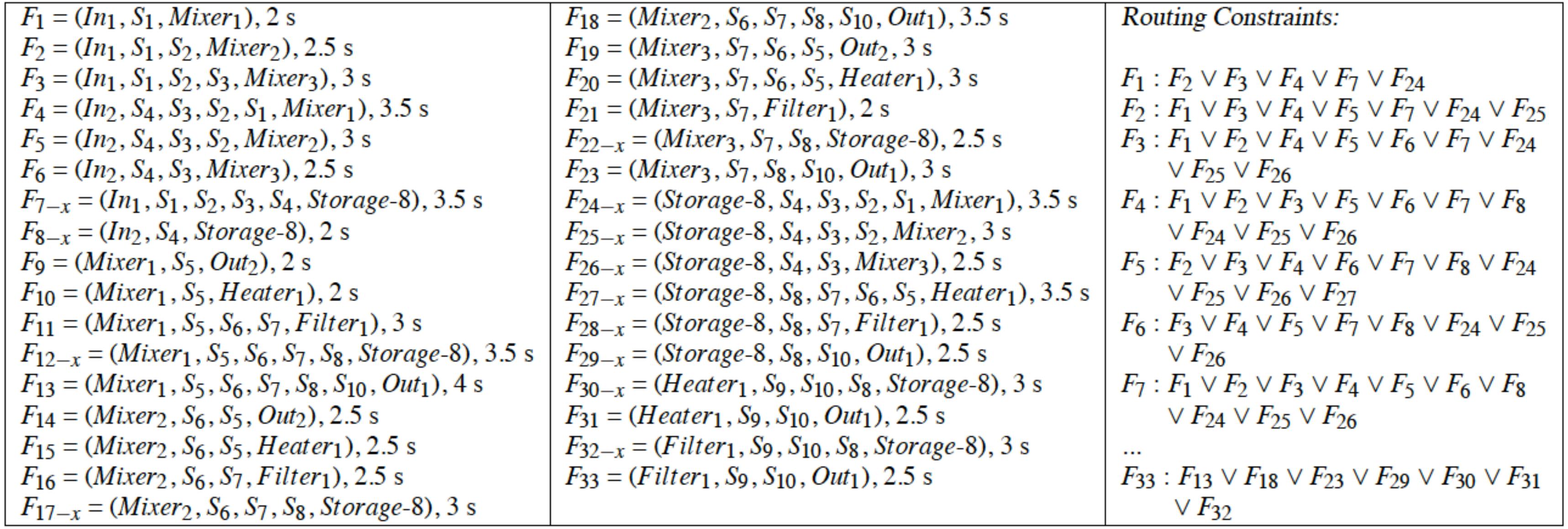


Table 2: Component Library ($\mathcal{L}$): Flow Layer Model

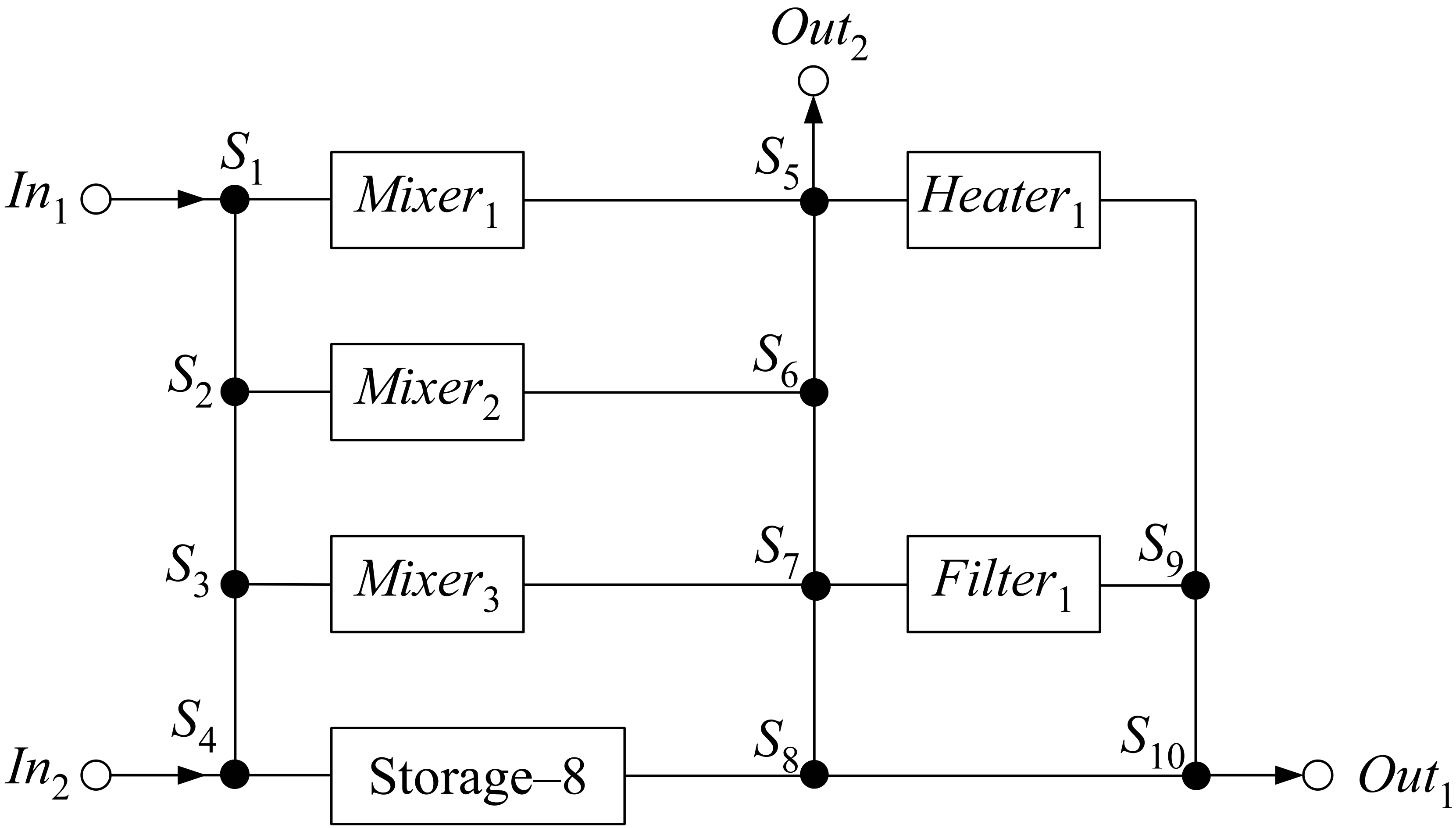
Component	Phases (P)	Execution Time (C)
Mixer	Ip1/ Ip2/ **Mix**/ Op1/ Op2	0.5s
Filter	Ip/ **Filter**/ Op1/ Op2	20s
Detector	Ip/ **Detect**/ op	5s
Separator	Ip1/ Ip2/ **Separate**/ Op1/ Op2	140s
Heater	Ip/ **Heat**/ Op	20°C/s

Flow paths in the architecture

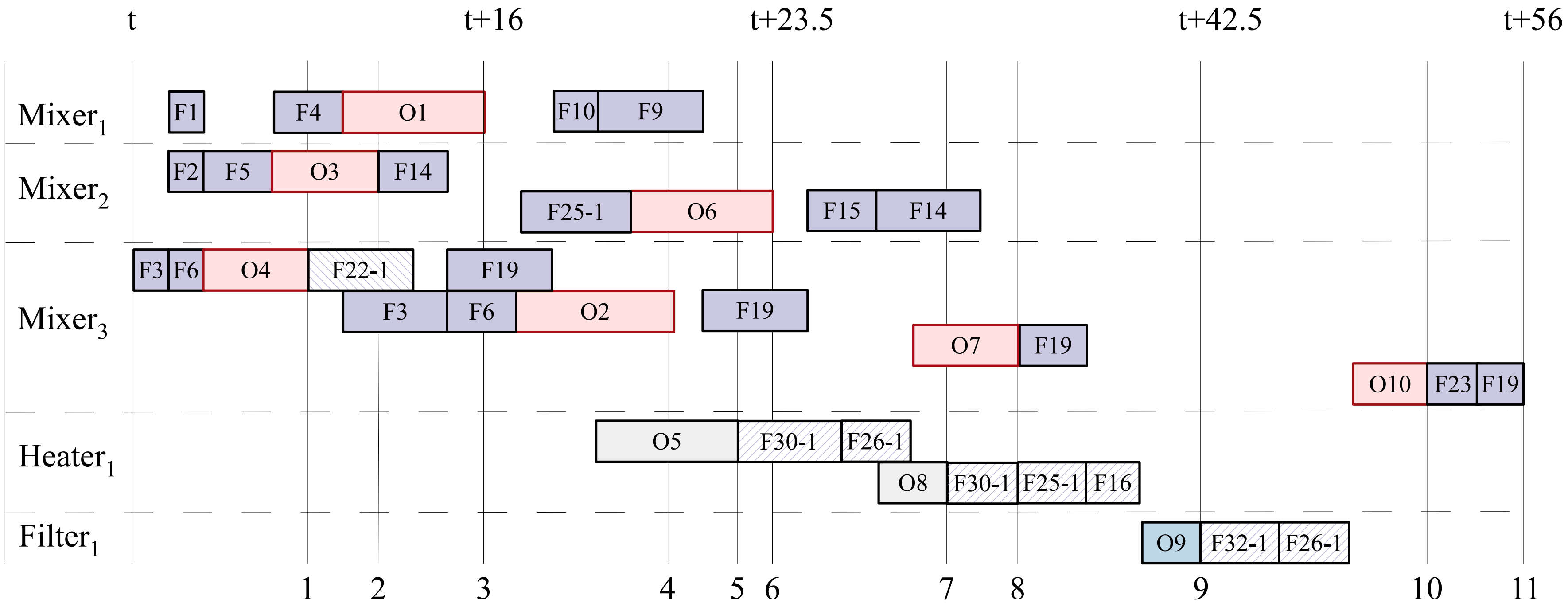


$F_1 = (In_1, S_1, Mixer_1)$, 2 s	$F_{18} = (Mixer_2, S_6, S_7, S_8, S_{10}, Out_1)$, 3.5 s	*Routing Constraints:*
$F_2 = (In_1, S_1, S_2, Mixer_2)$, 2.5 s	$F_{19} = (Mixer_3, S_7, S_6, S_5, Out_2$, 3 s	
$F_3 = (In_1, S_1, S_2, S_3, Mixer_3)$, 3 s	$F_{20} = (Mixer_3, S_7, S_6, S_5, Heater_1)$, 3 s	$F_1 : F_2 \vee F_3 \vee F_4 \vee F_7 \vee F_{24}$
$F_4 = (In_2, S_4, S_3, S_2, S_1, Mixer_1)$, 3.5 s	$F_{21} = (Mixer_3, S_7, Filter_1)$, 2 s	$F_2 : F_1 \vee F_3 \vee F_4 \vee F_5 \vee F_7 \vee F_{24} \vee F_{25}$
$F_5 = (In_2, S_4, S_3, S_2, Mixer_2)$, 3 s	$F_{22-x} = (Mixer_3, S_7, S_8, Storage\text{-}8)$, 2.5 s	$F_3 : F_1 \vee F_2 \vee F_4 \vee F_5 \vee F_6 \vee F_7 \vee F_{24}$
$F_6 = (In_2, S_4, S_3, Mixer_3)$, 2.5 s	$F_{23} = (Mixer_3, S_7, S_8, S_{10}, Out_1)$, 3 s	$\vee F_{25} \vee F_{26}$
$F_{7-x} = (In_1, S_1, S_2, S_3, S_4, Storage\text{-}8)$, 3.5 s	$F_{24-x} = (Storage\text{-}8, S_4, S_3, S_2, S_1, Mixer_1)$, 3.5 s	$F_4 : F_1 \vee F_2 \vee F_3 \vee F_5 \vee F_6 \vee F_7 \vee F_8$
$F_{8-x} = (In_2, S_4, Storage\text{-}8)$, 2 s	$F_{25-x} = (Storage\text{-}8, S_4, S_3, S_2, Mixer_2$, 3 s	$\vee F_{24} \vee F_{25} \vee F_{26}$
$F_9 = (Mixer_1, S_5, Out_2)$, 2 s	$F_{26-x} = (Storage\text{-}8, S_4, S_3, Mixer_3)$, 2.5 s	$F_5 : F_2 \vee F_3 \vee F_4 \vee F_6 \vee F_7 \vee F_8 \vee F_{24}$
$F_{10} = (Mixer_1, S_5, Heater_1)$, 2 s	$F_{27-x} = (Storage\text{-}8, S_8, S_7, S_6, S_5, Heater_1)$, 3.5 s	$\vee F_{25} \vee F_{26} \vee F_{27}$
$F_{11} = (Mixer_1, S_5, S_6, S_7, Filter_1)$, 3 s	$F_{28-x} = (Storage\text{-}8, S_8, S_7, Filter_1)$, 2.5 s	$F_6 : F_3 \vee F_4 \vee F_5 \vee F_7 \vee F_8 \vee F_{24} \vee F_{25}$
$F_{12-x} = (Mixer_1, S_5, S_6, S_7, S_8, Storage\text{-}8)$, 3.5 s	$F_{29-x} = (Storage\text{-}8, S_8, S_{10}, Out_1)$, 2.5 s	$\vee F_{26}$
$F_{13} = (Mixer_1, S_5, S_6, S_7, S_8, S_{10}, Out_1)$, 4 s	$F_{30-x} = (Heater_1, S_9, S_{10}, S_8, Storage\text{-}8)$, 3 s	$F_7 : F_1 \vee F_2 \vee F_3 \vee F_4 \vee F_5 \vee F_6 \vee F_8$
$F_{14} = (Mixer_2, S_6, S_5, Out_2)$, 2.5 s	$F_{31} = (Heater_1, S_9, S_{10}, Out_1)$, 2.5 s	$\vee F_{24} \vee F_{25} \vee F_{26}$
$F_{15} = (Mixer_2, S_6, S_5, Heater_1)$, 2.5 s	$F_{32-x} = (Filter_1, S_9, S_{10}, S_8, Storage\text{-}8)$, 3 s	...
$F_{16} = (Mixer_2, S_6, S_7, Filter_1)$, 2.5 s	$F_{33} = (Filter_1, S_9, S_{10}, Out_1)$, 2.5 s	$F_{33} : F_{13} \vee F_{18} \vee F_{23} \vee F_{29} \vee F_{30} \vee F_{31}$
$F_{17-x} = (Mixer_2, S_6, S_7, S_8, Storage\text{-}8)$, 3 s		$\vee F_{32}$

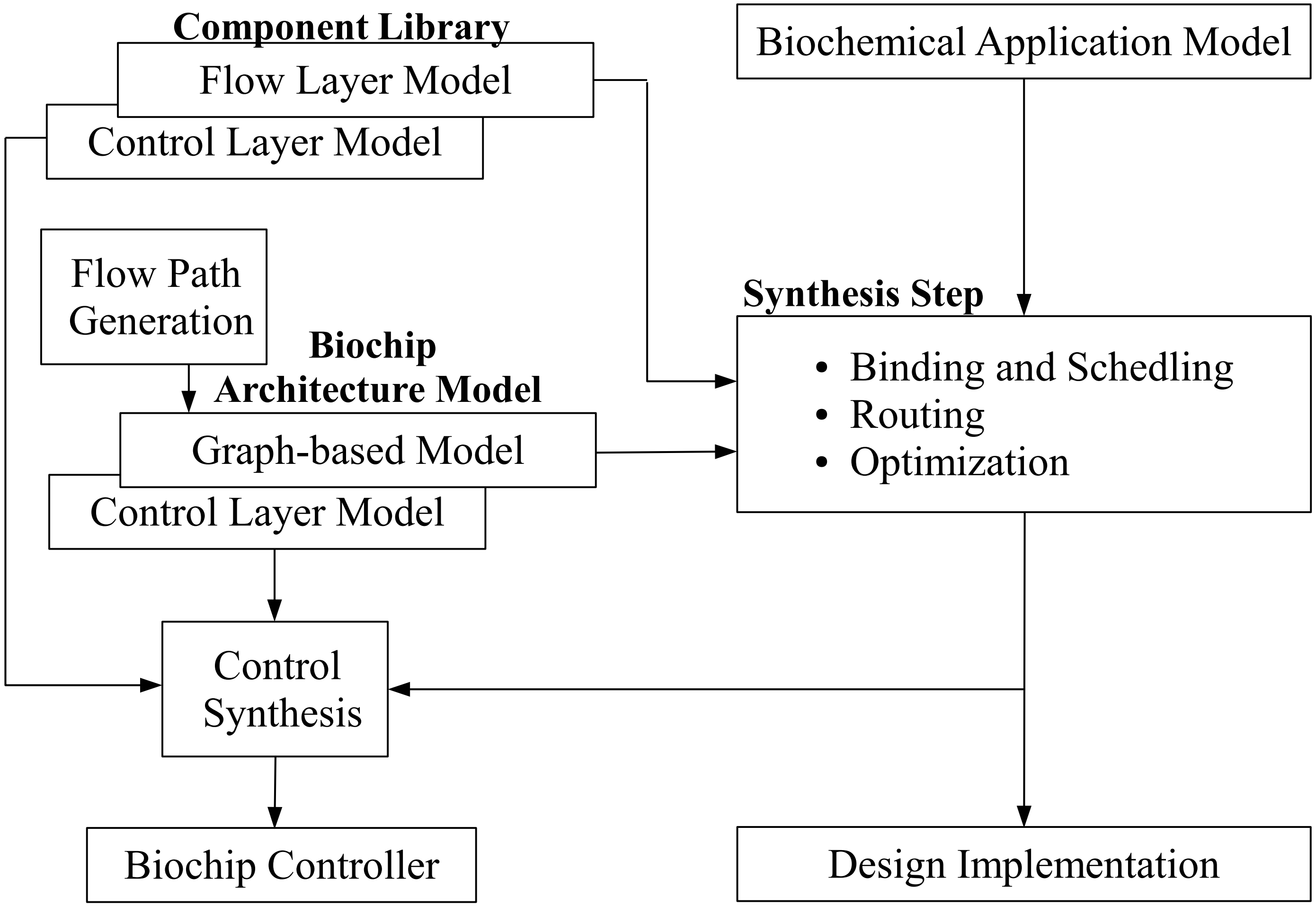
Flow paths in the architecture



Scheduling


t
t+16
t+23.5
t+42.5
t+56
Mixer$_1$
F1
F4
O1
F10
F9
Mixer$_2$
F2
F5
O3
F14
F25-1
O6
F15
F14
Mixer$_3$
F3
F6
O4
F22-1
F19
F3
F6
O2
F19
O7
F19
O10
F23
F19
Heater$_1$
O5
F30-1
F26-1
O8
F30-1
F25-1
F16
Filter$_1$
O9
F32-1
F26-1
1
2
3
4
5
6
7
8
9
10
11

Biochip Design Methodology



Conclusions

- A system-level modeling and synthesis approach for flow-based microfluidic biochips is possible
- The right abstraction allows for using techniques and methods from MPSoC design

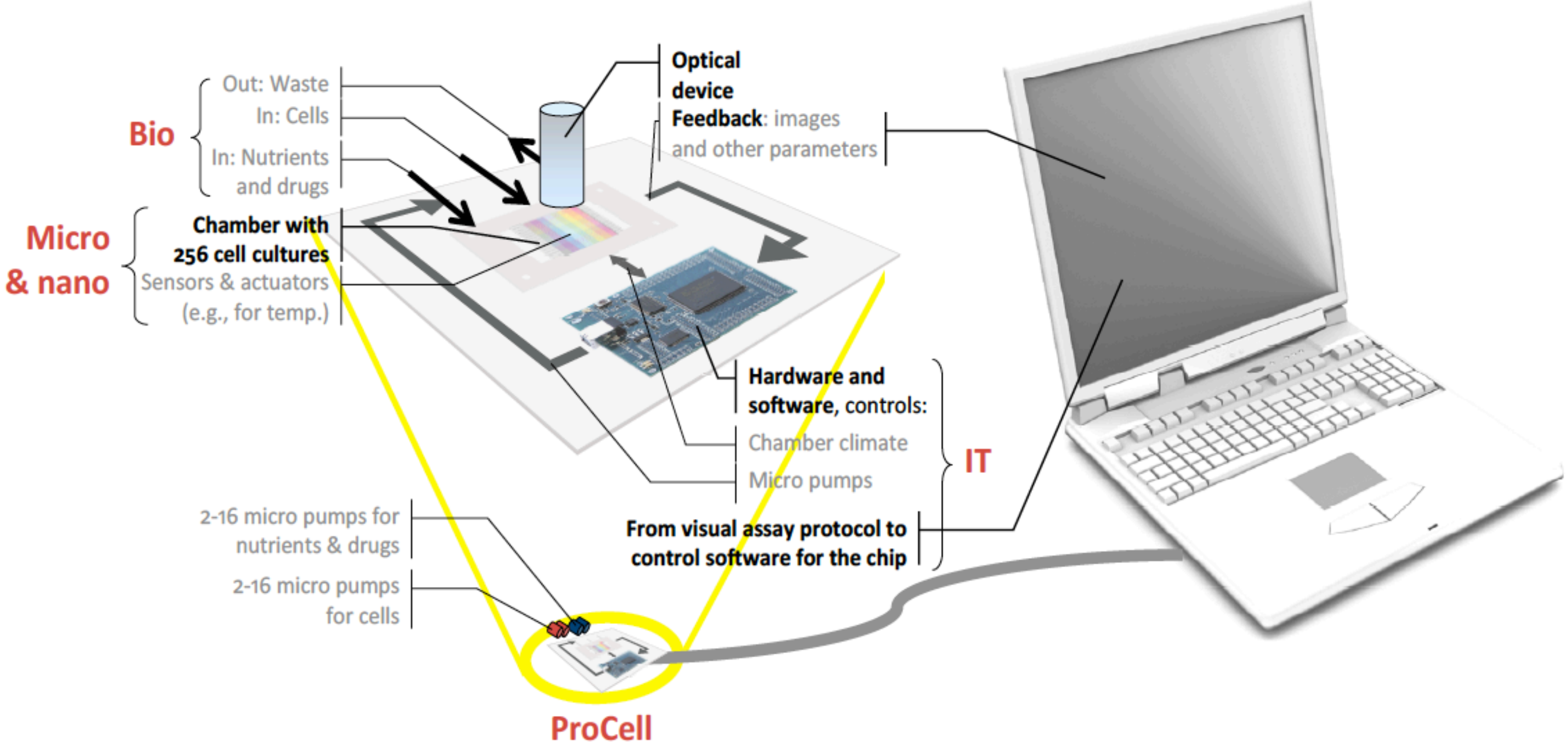
References:

1. Wajid Hassan Minhass, Paul Pop, Jan Madsen, Mette Hemmingsen, Martin Dufva. System-Level Modeling and Simulation of the Cell Culture Microfluidic Biochip ProCell, Symposium on Design, Test, Integration & Packaging of MEMS/MOEMS, 2010
2. Lee *et al.* Stand-alone self-powered integrated microfluidic blood analysis system (SIMBAS). In Lab on a Chip, vol. 11, no. 5, 7 March 2011, pages 845-850

PART 2: FLOW-BASED MICROFLUIDIC BIOCHIPS

POSSIBILITIES AND CHALLENGES

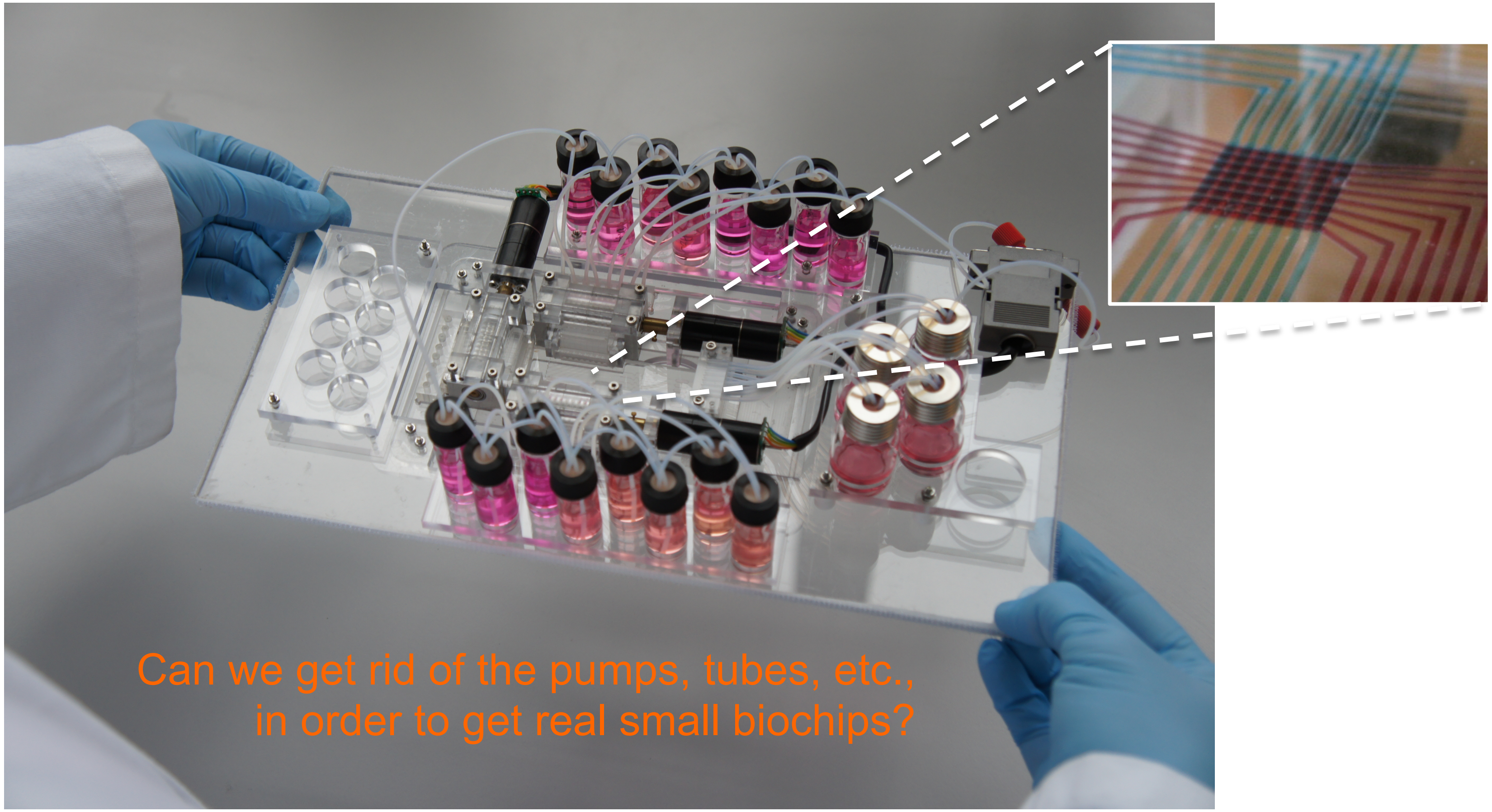
ProCell: Programmable Cell Chip


Bio
Out: Waste
In: Cells
In: Nutrients and drugs
Micro & nano
Chamber with 256 cell cultures
Sensors & actuators (e.g., for temp.)
Optical device
Feedback: images and other parameters
Hardware and software, controls:
Chamber climate
Micro pumps
IT
From visual assay protocol to control software for the chip
2-16 micro pumps for nutrients & drugs
2-16 micro pumps for cells
ProCell

ProCell: Aims

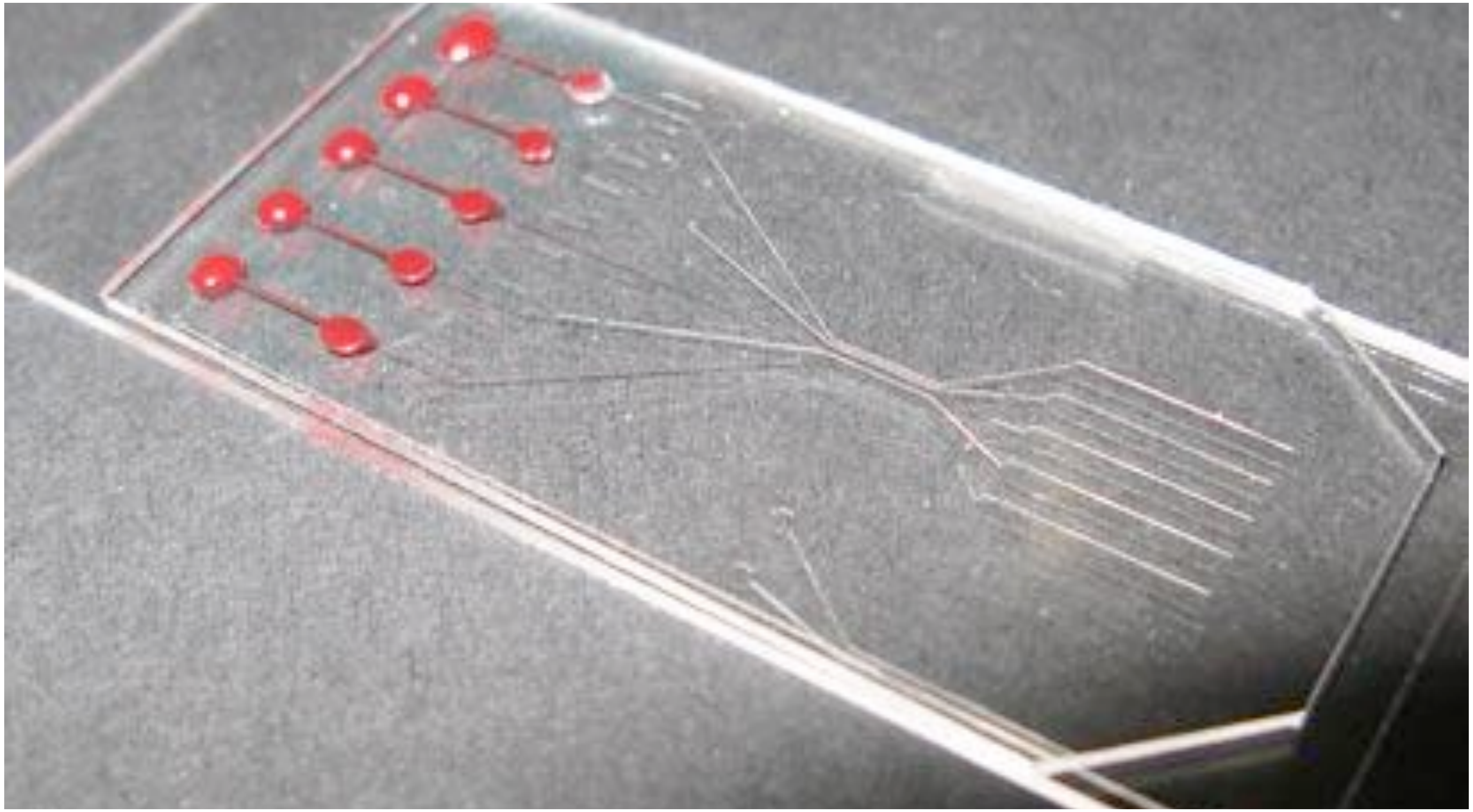
- Culturing and Manipulation of living cells with real-time reaction monitoring
- Automatically manipulate cells based on their observed behavior
- Allows for conditional experiments
- Simulate *in vivo* conditions by *in vitro* experiments

ProCell Prototype



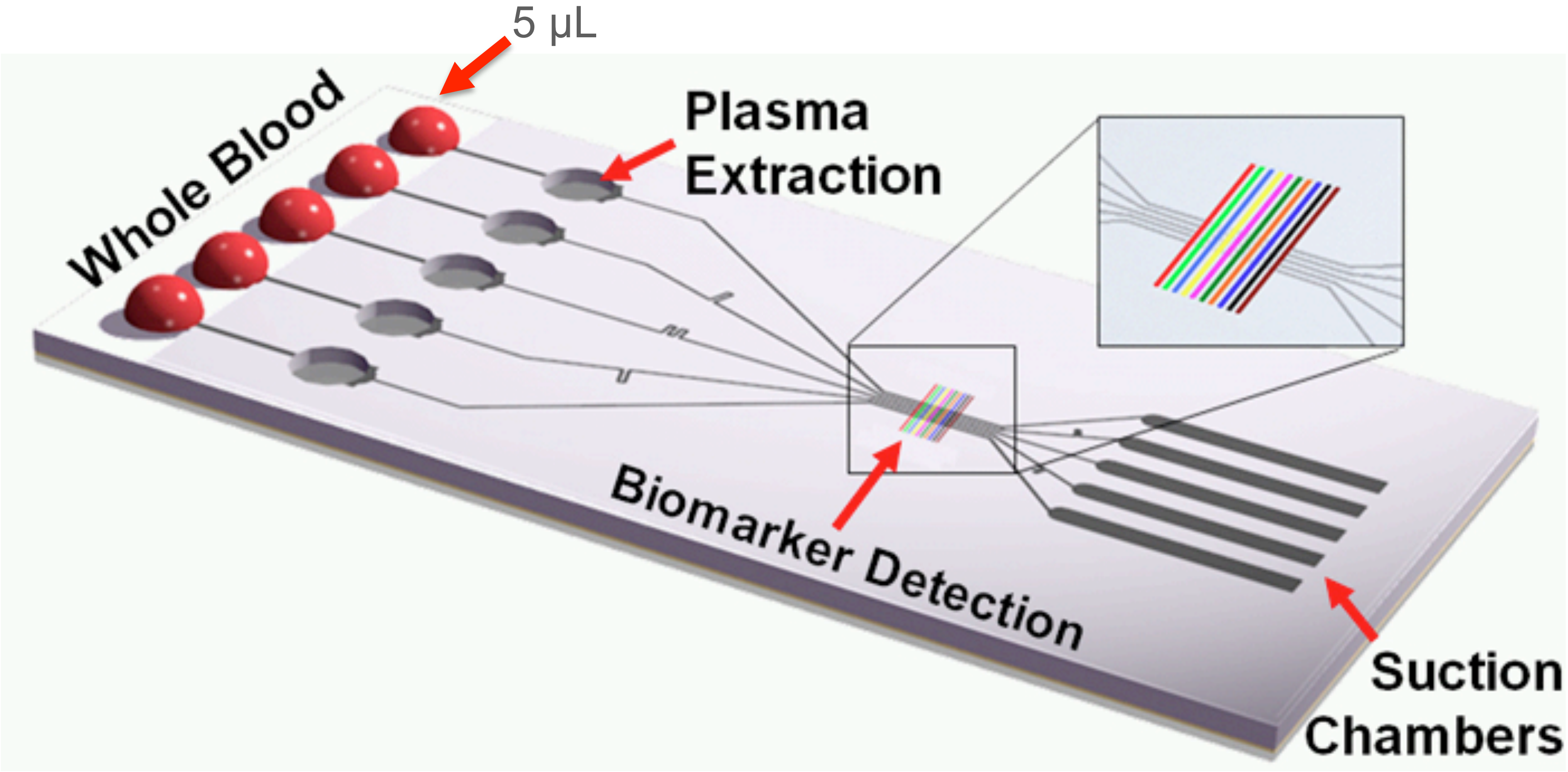
Can we get rid of the pumps, tubes, etc.,
in order to get real small biochips?

SIMBAS:
Stand-alone, Self-powered Biochip!



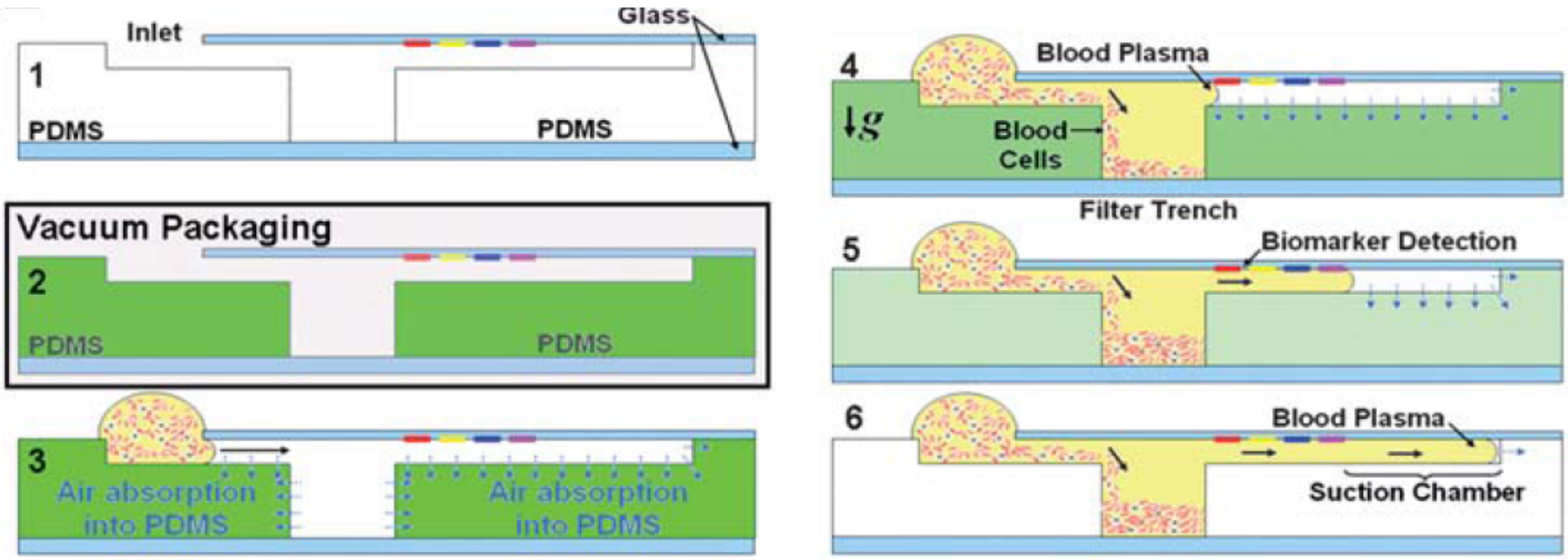
Courtesy: Ivan Dimov http://newscenter.berkeley.edu March 16, 2011

SIMBAS Biochip

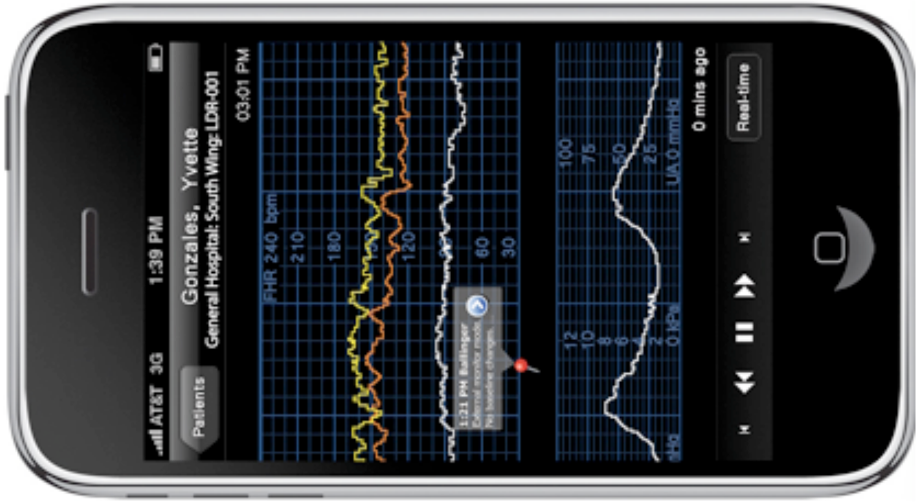


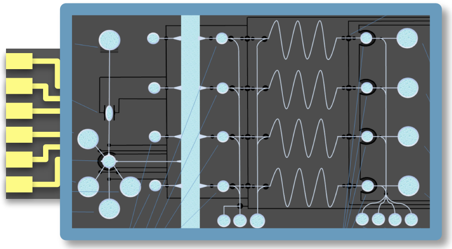
Courtesy: Ivan Dimov http://newscenter.berkeley.edu March 16, 2011

SIMBAS Biochip : Working principles



Future?


AT&T 3G
1:39 PM
Patients
Gonzales, Yvette
General Hospital: South Wing: LDR-001
03:01 PM
FHR 240 bpm
210
180
120
60
30
100
75
50
25
UA 0 mmHg
0 mins ago
Real-time



Thank you for your attention