2016-03

MINT - Microfluidic Netlist

Sanka, Radhakrishna

IWBDA

MINT

A MIcrofluidic NetlisT Format

Radhakrishna Sanka[1], Haiyao Huang[1], Ryan Silva[1], and Douglas Densmore[1]

[1]Department of Electrical & Computer Engineering, Boston University, Boston, MA

{sanka,rjsilva,dougd}@bu.edu

1. INTRODUCTION

Fluigi [6] is a microfluidic design framework that allows researchers to realize abstract descriptions of liquid flow relationships automatically as physical devices and corresponding control software. Its goal is to provide synthetic biology researchers with the tools to use microfluidics for novel computation, discovery, and test applications. A critical component of this work-flow is MINT, a format for describing the microfluidic components and the connectivity of the control and flow layers in the microfluidic device. This work describes MINT and where it falls in the larger Fluigi software flow.

Figure 1 illustrates the two work-flows that are a part of Fluigi. In particular, it demonstrates how Fluigi generates designs from both MINT design files and "Functional Descriptions" using tools like Cello [8]. The output generated by Fluigi can be either a photomask drawing for traditional soft layer photolithography [4] or a vector drawing output for the Makerfluidics [10] toolchain to fabricate the microfluidic devices. Using MINT, designers can share their designs with the rest of the community and make use of Fluigi to fabricate their designs.

2. MINT

MINT is based on the microfluidic netlist abstraction introduced in [7]. The microfluidic component descriptions have been expanded to include additional microfluidic components from the scientific and microfluidic literature [6] and has syntax and semantics extended to create a self contained Microfluidic Hardware Description Language with the ability to describe constructs required to perform common microfluidic operations.

The MINT microfluidic constructs are primarily described as either Primitives [7] or Modules [6] where the Modules are a collection of Primitives which are treated as a single entity. All of the MINT constructs can be customized to fit specific design needs by modifying their parameters. The primitives consist of 'PORT', 'MIXER'[11], 'CHANNEL', 'VALVE'[12], 'NET' and 'CELL TRAP'[9] and are the fundamental building blocks. The MINT modules currently supported by Fluigi are 'BANK', 'MUX'[12], 'TREE', 'LOGIC ARRAY', 'GRADIENT GENERATOR'[3], 'DROPLET GENERATOR' [11] and 'ROTARY PUMP'[13].

In a MINT device description, the microfluidic constructs are declared in either the "FLOW" or the "CONTROL" layer by declaring them within the respective 'LAYER' blocks. In addition to the primitives and the standard modules, MINT also allows the designer to import devices created by the user and instantiate them. MINT lets designers tag any of the 'PORT' components present in their design as a 'TERMINAL'. Once tagged the device can be instantiated and be treated any like modules. By instantiating them before the layer blocks, the designer can then make connections to the instance 'TERMINAL'. An example of this importing capability has been demonstrated in Figure 2. A detailed documentation and a tutorial of the MINT example shown in Figure 2 is available in the MINT Wiki [1].

3. INTEGRATION WITH FLUIGI

MINT allows the designer to use Fluigi as and end-to-end microfluidic design tool. The Fluigi software tool consists of a parser for the MINT, a place and route layout system based on [2] and [5] algorithms, a control sequence generator, and an output generator for drawing photomask and vector design files of the device layout. It can use the MINT design files to place and route the device layouts. It is also capable of importing and instantiating designs to generate complex hierarchical designs as shown in Figure 2. In addition, it is also capable of generating microfluidic devices for genetic circuits and their corresponding control sequences [7].

4. CONCLUSION

Using MINT designers can now share their microfluidic designs and create complex Lab on Chip (LoC) or Microfluidic Large Scale Integration (MLSI) systems utilizing the "sandbox" created by the combination of MINT, Fluigi and Makerfluidics.

The integration of synthetic biology, novel microfluidic fabrication techniques, and embedded electronics has the potential to transform the landscape of engineering biological systems. Since applications in cell-based therapeutics and biosensors expand on the idea of distributed biological computation, they could be greatly benefited with advances in CAD tools that not only automate the design process for Flow Microfluidics but also take advantage of novel fabrication techniques.

5. REFERENCES

[1] MINT Github Wiki. https://github.com/CIDARLAB/mint/wiki, 2016.

[2] V. Betz and J. Rose. VPR: A new packing, placement, and routing tool for FPGA research. In *International Workshop on Field Programmable Logic and Appliation*, 1997.

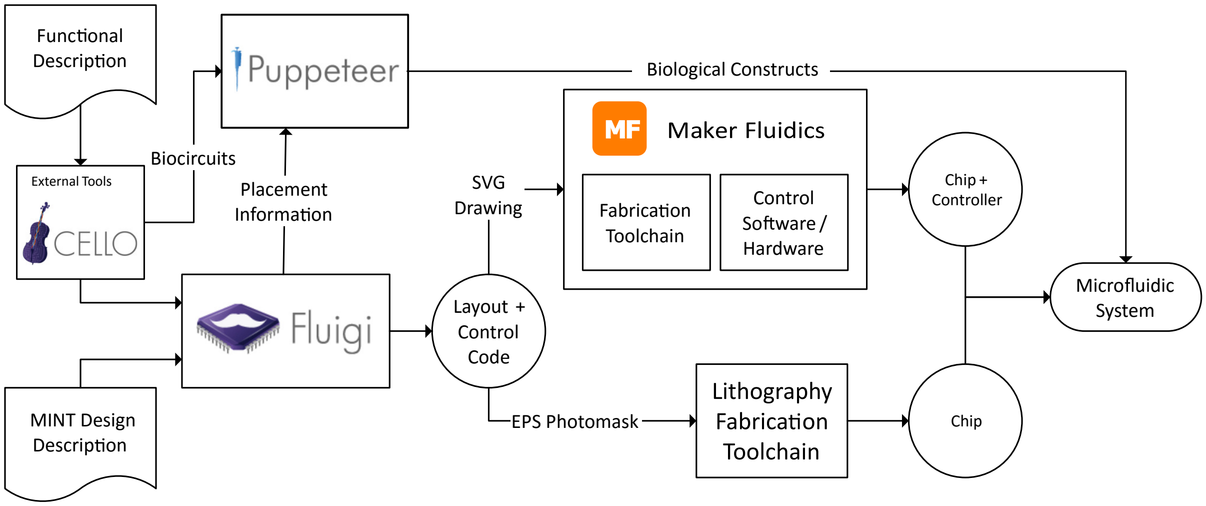
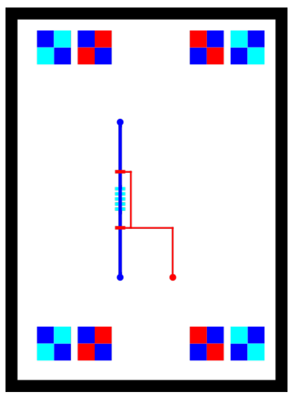


Figure 1: Fluigi Work-flows. Shown is how a MINT Description or a Functional Description of a biological circuit can be synthesized into a microfluidic device. It also shows how the different research projects will interact with Fluigi to automate entire process in the future.

```
DEVICE net_test

LAYER FLOW
PORT p1, p2 r=100;
V LONG CELL TRAP ct1 numChambers=10
    chamberWidth=100 chamberLength=100
    chamberSpacing=50 channelWidth=100;

CHANNEL c1 from p1 3 to ct1 1 w=100;
CHANNEL c2 from ct1 2 to p2 1 w=100;

TERMINAL 1 p1;
TERMINAL 2 p2;

END LAYER

LAYER CONTROL
PORT p3 r=100;
VALVE v1 on c1 w=300 l=100;
VALVE v2 on c2 w=300 l=100;
NET net1 from p3 1 to v1 4, v2 4
    channelWidth=50;
END LAYER
```



```
IMPORT net_test

DEVICE module_test


net_test nt1,nt2,nt3,nt4;

LAYER FLOW

CHANNEL cx1 from nt1 1 to nt2 2 w = 100;
CHANNEL cx2 from nt2 1 to nt3 2 w = 100;
CHANNEL cx3 from nt3 1 to nt4 2 w = 100;



END LAYER
```

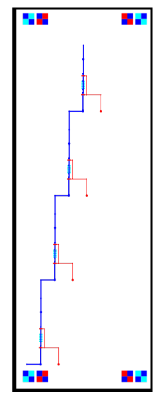


Figure 2: Importing in MINT. The Figure shows MINT code snippets and their corresponding outputs generated by Fluigi on their right. The first code snippet describes the device *net_test* and the second describes *module_test* which imports, instantiates the device *net_test* and makes connections between the instances.

[3] S. K. W. Dertinger, D. T. Chiu, N. L. Jeon, and G. M. Whitesides. Generation of Gradients Having Complex Shapes Using Microfluidic Networks. *Analytical Chemistry*, 73(6):1240–1246, Mar. 2001.

[4] M. S. Ferry, I. A. Razinkov, and J. Hasty. Chapter fourteen - Microfluidics for Synthetic Biology: From Design to Execution. In C. Voigt, editor, *Methods in Enzymology*, volume 497 of *Synthetic Biology, Part A*, pages 295–372. Academic Press, 2011.

[5] F. Hadlock. A shortest path algorithm for grid graphs. *Networks*, 7(4):323–334, 1977.

[6] H. Huang. *Fluigi: An end-to-end Software Workflow for Microfluidic Design*. PhD thesis, Boston University, 2015.

[7] H. Huang and D. Densmore. Fluigi: Microfluidic device synthesis for synthetic biology. *J. Emerg. Technol. Comput. Syst. Special Issue on Synthetic Biology*, 11(3):26:1–26:19, Dec. 2014.

[8] A. A. K. Nielsen, B. S. Der, J. Shin, P. Vaidyanathan, V. Paralanov, E. A. Strychalski, D. Ross, D. Densmore, and C. A. Voigt. Genetic circuit design automation. *Science*, 352(6281), 2016.

[9] A. Prindle, P. Samayoa, I. Razinkov, T. Danino, L. S. Tsimring, and J. Hasty. A sensing array of radically coupled genetic /'biopixels/'. *Nature*, 481(7379):39–44, Jan. 2012.

[10] R. Silva, A. Heuckroth, C. Huang, A. Rolfe, and D. Densmore. Makerfluidics: Microfluidics for all. poster presented at Synberc: Fall 2015, September 2015.

[11] T. M. Squires and S. R. Quake. Microfluidics: Fluid physics at the nanoliter scale. *Reviews of Modern Physics*, 77(3):977–1026, Oct. 2005.

[12] T. Thorsen, S. J. Maerkl, and S. R. Quake. Microfluidic Large-Scale Integration. *Science*, 298(5593):580–584, Oct. 2002.

[13] J. P. Urbanski, W. Thies, C. Rhodes, S. Amarasinghe, and T. Thorsen. Digital microfluidics using soft lithography. *Lab on a Chip*, 6(1):96–104, Dec. 2006.