# Computation by Block Copolymer Self-Assembly

by

Hyung Wan Do

B.S. Electrical Engineering
B.S. Business Economics and Management
California Institute of Technology, 2012

S.M. Electrical Engineering and Computer Science
Massachusetts Institute of Technology, 2014

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2018

Signature of Author: _____
Department of Electrical Engineering and Computer Science
May 22, 2018

Certified by: _____
Karl K. Berggren
Professor of Electrical Engineering
Thesis Supervisor

Accepted by: _____
Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

# Computation by Block Copolymer Self-Assembly

by

Hyung Wan Do

Submitted to the Department of Electrical Engineering and Computer Science
on May 22, 2018 in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

## Abstract

Unconventional computation is a paradigm of computation that uses novel information tokens from natural systems to perform information processing. Using the complexity of physical systems, unconventional computing systems can efficiently solve problems that are difficult to solve classically. In this thesis, we use block copolymer self-assembly, a well-studied phenomenon in polymer science, to develop a new approach to computing by applying directed self-assembly to implement Ising-model-based computing systems in materials.

In the first part of the thesis, we investigate directed self-assembly of block copolymer thin films within templates of different polygonal shapes. We define a two-state system based on the two degenerate alignment orientations of the ladder-shaped block copolymer structures formed inside square confinements, and study properties of the two-state system.

In the second part of the thesis, we demonstrate an Ising lattice setup for directed self-assembly of block copolymers defined on two-dimensional arrays of posts. We develop an Ising-model-based simulation method that can perform block copolymer pattern prediction and template design. Finally, we design simple Boolean logic gates as a proof-of-concept demonstration of computation.

Thesis Supervisor: Karl K. Berggren
Title: Professor of Electrical Engineering

# Acknowledgements

First of all, I would like to express my sincere gratitude to my advisor Professor Karl K. Berggren for his active guidance throughout my research. I have learned a lot from his invaluable advice on research as well as communication and writing skills. His organized approach to research helped me define research directions and solve difficult problems. Karl gave me an opportunity to work on such an interesting interdisciplinary thesis topic. Thank you for being my advisor.

I would like to thank my thesis committee, Professor Caroline A. Ross and Professor Christopher J. Terman for their guidance and support. Caroline gave me many helpful advice on anything related to block copolymer self-assembly. Chris gave me an opportunity to work as a teaching assistant for 6.004, and I had a wonderful experience teaching.

I would like to thank Jae-Byum Chang and Hong Kyoon Choi. They have helped me since the beginning of my PhD research. Jae-Byum taught me many useful nanofabrication techniques and Hong Kyoon taught me all about block copolymer processing. They have provided me with helpful feedback and advice on research, career development, and general graduate student life.

I would like to thank the Quantum Nanostructures and Nanofabrication group members and block copolymer meeting members. Especially, I had valuable discussions about block copolymer self-

# Contents

# List of figures

## Chapter 1

## Chapter 2

## Chapter 3

## Chapter 4

## Chapter 5

## Chapter 6

## Chapter 7

## Appendix

# Chapter 1

# Introduction

## 1.1  Overview

Unconventional computation is a paradigm of computation that uses natural systems, physical or biological, to process information. Using the complexity of physical systems, unconventional computing systems can efficiently solve problems that are difficult to solve with classical von Neumann architecture.[1] One such alternative architecture is based on the Ising spin system where a large number of binary states give rise to long-range correlations through strong local interactions.[2] It has been previously shown that Ising-model-based computing systems can perform universal Boolean operations[3] and solve combinatorial optimization problems.[4] In this thesis, we use block copolymer self-assembly, a well-studied phenomenon in polymer science, to develop a new approach to computing by applying directed self-assembly (DSA) of block copolymers to implement Ising-model-based computing systems in materials.

Implementing Ising-model-based computing systems using block copolymer self-assembly is of interest not only from a computational point of view, but also from a materials point of view since block copolymers naturally act as a lithography material. By understanding the interactions between the Ising states defined using block copolymers, we can apply the Ising model to calculate the equilibrium block copolymer morphology given a specific template. Ising model simulations

can complement self-consistent field theory (SCFT) simulations to assist rule-based and inverse template design.

This thesis investigates block-copolymer-based computation using arrays of square confinements and arrays of square lattice posts to define a two-state system. Chapter 1 introduces the basics of block copolymer DSA and the concept of unconventional computation. Chapter 2 and Chapter 3 focus on DSA of block copolymer thin films within templates of different polygonal shapes. Chapter 4, Chapter 5, and Chapter 6 focus on Ising lattice design in block copolymer DSA, Ising-model-based simulation, and Ising-model-based computation. Chapter 7 suggests ideas for future work.

## 1.2   Block copolymer directed self-assembly

Block copolymers are self-assembling macromolecules composed of two distinct repeating monomer units that are connected to each other by a covalent bond. Since the two blocks are chemically immiscible, but are covalently bonded together at the same time, block copolymers microphase separate into periodic arrays of microdomains upon annealing. For thin film applications, one of the blocks is used for pattern transfer while the other block is removed after the self-assembly process is completed. For example, the block copolymer used in this thesis is poly(styrene-*block*-dimethylsiloxane) (PS-*b*-PDMS). Here, the PDMS block has high etch resistance due to its silica backbone whereas the PS block has low etch resistance so that it can be easily removed by performing an $O_2$ plasma treatment.

The phase behavior of block copolymers is governed by the segregation strength $\chi N$ and volume fraction of one block $f$, where $\chi$ is the Flory-Huggins interaction parameter describing the chemical incompatibility between the two blocks and $N$ is the degree of polymerization.[5] Depending on these parameters, bulk block copolymers can microphase separate into diverse

morphologies including hexagonal arrays of spheres, parallel in-plane or out-of-plane cylinders, gyroid, and parallel in-plane or out-of-plane lamellae, where the equilibrium periodicity $L_0$ (~ $\chi^{1/6}N^{2/3}$) and feature size are on the order of the size of the macromolecules.[6] **Figure 1.1** shows the bulk morphologies and phase diagrams for block copolymers. Due to their ability to form periodic and complex nanostructures, block copolymers have been used in high-resolution lithography to fabricate field-effect transistors,[7-9] photovoltaic devices,[10-12] and plasmonic nanostructures.[13-15] However, achieving block copolymer patterns with low defect density and good long-range order has been a challenge.
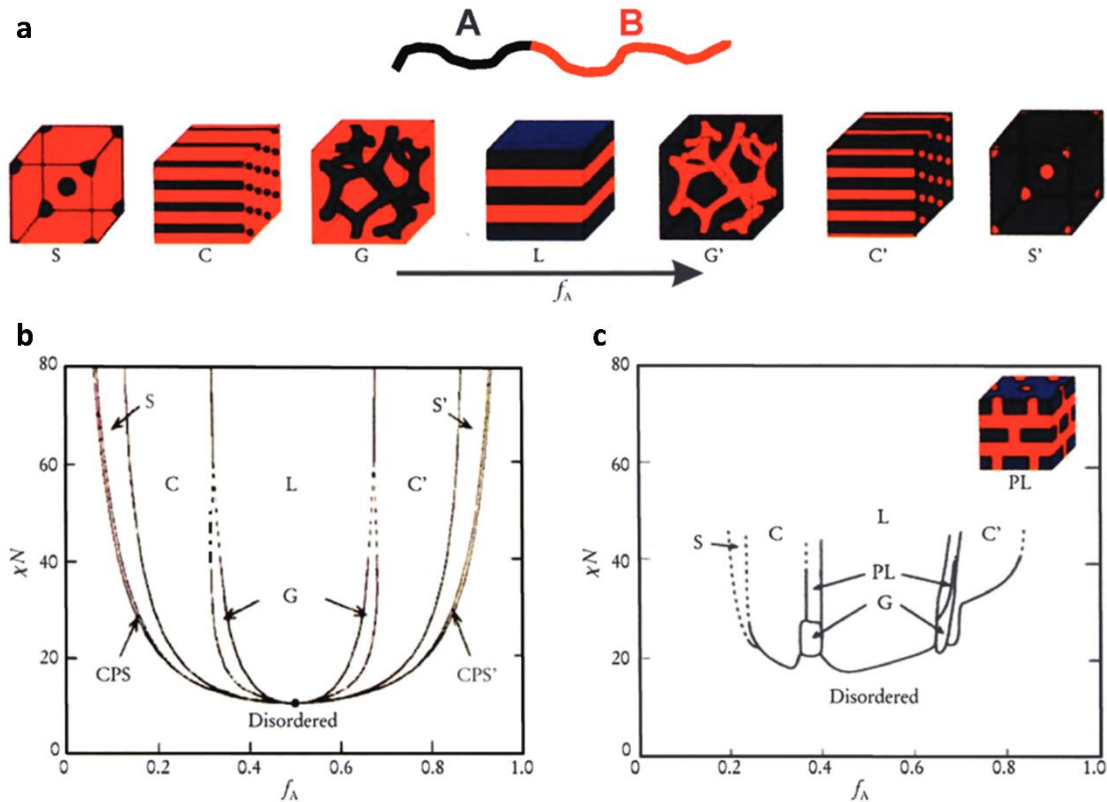


**Figure 1.1** Schematic diagram showing diverse bulk morphologies and phase diagrams of block copolymers.[5] $\chi N$ is the segregation strength and $f_A$ is the volume fraction of block $A$. (a) Equilibrium block copolymer morphologies: spheres (S), cylinders (C), gyroid (G), and lamellae (L). (b) A theoretical phase diagram calculated by SCFT. Close-packed sphere (CPS) morphology is also predicted. (c) An experimental phase diagram of poly(isoprene-*block*-styrene) (PIP-*b*-PS). Perforated lamellae (PL) morphology is also observed. Redrawn from reference 5.

To provide pattern registration, reduce defectivity, and improve long-range order, topographic[16-19] or chemical[20-24] templates have been commonly used to guide the self-assembly of block copolymers. In graphoepitaxy, topographic features are patterned by conventional lithography methods and chemically functionalized to be attractive to one block. In chemoepitaxy, regions of the substrate are chemically patterned to be preferential to one block. Structural frustration and interfacial interactions induced by the template guide the self-assembly of block copolymers. Pattern registration and density multiplication have been demonstrated using both templating strategies. **Figure 1.2** shows examples of complex patterns self-assembled by graphoepitaxy of PS-*b*-PDMS. In this thesis, we focus on the graphoepitaxy of a cylindrical morphology PS-*b*-PDMS using polygonal confinement and post lattice templates.
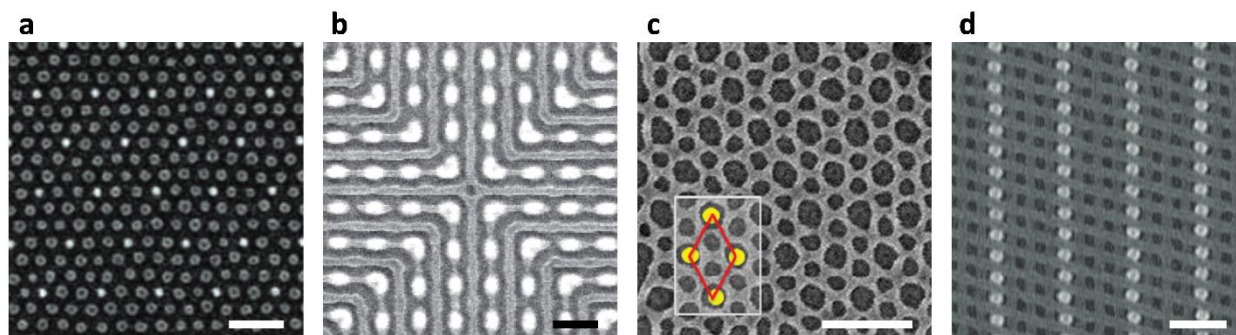


**Figure 1.2** Examples of ordered block copolymer patterns using graphoepitaxy of PS-*b*-PDMS. (a) Hexagonal array of self-assembled spheres.[25] (b) Nested-elbow structures with a center cross-point junction.[18] (c) Hexagonal symmetry nanohole array using a sacrificial template.[26] (d) Three-dimensional bilayer mesh structure.[27] Scale bars, 100 nm. Figures were redrawn from references.

## 1.3  Block copolymer as a material for computation

Unconventional computation is a model of computation that uses novel information tokens to perform computation. Unlike traditional silicon-based computers which electronically implement logic gates, unconventional computers exploit various physical phenomena to perform

computation. A key advantage of unconventional computers is their ability to efficiently solve problems that are difficult to solve classically. For a number of applications, the material properties of the physical system may be utilized for efficient computation. For example, Reyes et al. have demonstrated an analog computing approach for solving a wide class of shortest path problems by using glow discharge inside microfluidic channel systems to explore potential pathway choices in parallel.[28] In another example, deoxyribonucleic acid (DNA) has been used to solve the directed Hamiltonian path problem by trying all possible paths via polymerase chain reactions and agarose gel electrophoresis separations.[29]

Over recent decades, the scientific and engineering community has been pioneering new and unconventional approaches to computing based on novel architectures such as quantum computing,[30-32] and novel information tokens such as photons,[33-35] materials,[36-39] DNA molecules,[40-45] and even biological organisms.[46-49] **Figure 1.3** shows examples of such unconventional computing systems. In these examples, Boolean logic gates have been built and combinatorial optimization problems such as the travelling salesman problem have been solved using physical systems.

One approach to designing a novel unconventional computing systems is to create Ising machines based on the well-known spin-spin system. Ising machines are physical systems capable of finding the ground states of Ising lattices. It has been previously shown that ground states of Ising lattices can encode universal computation,[3] and solving for the ground states of Ising lattices can map to various useful combinatorial optimization problems.[4] As such, Ising machines may be enlisted to serve as an efficient coprocessor for conventional computers faced with particularly challenging computational tasks.
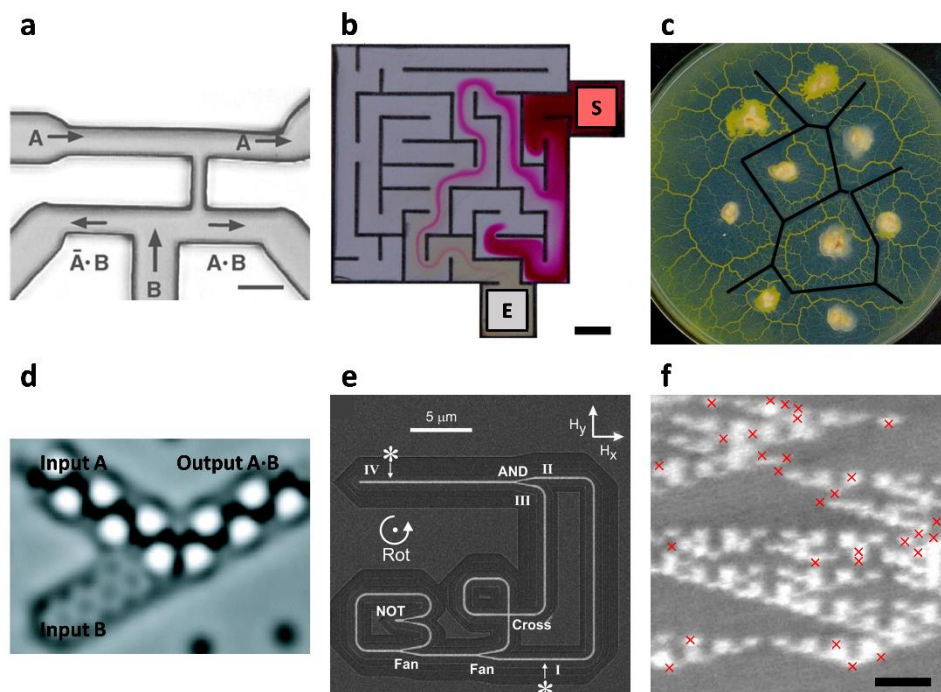
**Figure 1.3** Examples of unconventional computing systems. (a) Microfluidic bubble AND/OR gate.[50] Scale bar, 100 μm. (b) Maze solving using the Marangoni flow generated by the pH gradient in a channel network.[51] Scale bar, 2 mm. (c) Voronoi diagram approximated by a slime mould *physarum polycephalum*.[48] (d) AND gate implemented using carbon monoxide molecules arranged on a copper (111) surface.[39] (e) Magnetic nanowire loop containing a NOT gate, fan-out junction, and cross-over junction.[37] (f) Algorithmic self-assembly of DNA Sierpinski triangles.[44] Scale bar, 50 nm. Figures were redrawn from references.

Recent research has focused intensively on demonstrating these Ising machines, and some have been successful in building, for example, classical Ising machines using optical parametric oscillators[52-54] and CMOS circuits,[55,56] and quantum Ising machines using qubits.[57,58] In this thesis, we develop a classical Ising-model-based computing system by using a totally new high-density approach—self-assembling block copolymers. During the self-assembly process, block copolymers arrange themselves into a minimum free energy configuration given the boundary conditions defined by the guiding templates. In essence, block copolymers are finding a solution

to an energy optimization problem through self-assembly. Ising-model-based computing can thus be implemented with block copolymers.

In addition to solving interesting and useful computation problems, block-copolymer-based computing systems may enable energy-efficient computing. Block copolymers find their ground-state configuration with minimal required energy input. With sufficient annealing time available, arbitrarily low energy consumption is possible. Using the Ising lattice presented in this thesis, an Ising state occupies at most a volume of $L_0^3/2$ with a monolayer of PS-*b*-PDMS (as-spun film thickness $\leq L_0$). Assuming PS-*b*-PDMS has similar specific heat capacity and density as the majority PS block and the sample is thermally annealed at 150°C, we can estimate the upper bound for the energy cost of annealing an Ising state to be in the order of $10^{-15}$ J. The energy cost of a few fJ per operation is already comparable to gate dissipation in digital logic.[59-61] This energy can be provided in the form of waste heat, harvested during cooling of conventional industrial processes (e.g. as cooling components in power plants), and further optimized by controlling annealing times and materials. Indeed, annealing-based architectures can trade-off computing speed and energy costs so that with appropriate materials science and engineering, the energy costs of a self-assembly-based approach to computing could be made arbitrarily small.

Moreover, from a material science perspective, we may be able to study properties of block copolymers by implementing a block-copolymer-based computing system. Many physical systems consist of a large number of simple components. For instance, a block copolymer film spin coated onto a 12-inch silicon wafer contains approximately $10^{14}$ Ising states with nearest-neighbor interactions. By investigating the local interactions between the simple components and studying the resulting complex emergent behavior, we can better understand the self-assembly process. Results from the thesis will have applications in block copolymer pattern prediction as well as

template design. Block-copolymer-based computing systems may even be able to simulate material science, such as phase transition behaviors of block copolymers.

Finally, we expect that the techniques used in this thesis can be extended to other materials. In designing the Ising lattice setup for block copolymers, we did not rely on the specific properties of the self-assembling material other than the antiferromagnetic nearest-neighbor interactions. Therefore, the same designing principles could be applied to different self-assembling materials such as charged colloidal particles or DNA. Ultimately, results of our work will impact fields beyond unconventional computation. For example, a sensing platform could be built by positioning nanoparticles inside the post lattice template which can dynamically change the input bits in the presence of specific chemical or biomolecules. Furthermore, a computing system could perhaps be designed using charged colloidal particles that can dynamically reset their charges subject to the pH of the solution.[62]

# References

1. J. Jones, A. Adamatzky, *Nat. Comput.* **2014**, 13, 1.
2. E. Ising, *Z. Phys.* **1925**, 31.
3. M. Gu, A. Perales, *Phys. Rev. E* **2012**, 86, 1.
4. F. Barahona, *J. Phys. A: Math. Gen.* **1982**, 15, 10.
5. F. S. Bates, G. H. Fredrickson, *Phys. Today* **1999**, 52, 2.
6. A. N. Semenov, *Zh. Eksp. Teor. Fiz.* **1985**, 88, 4.
7. J. G. Son, M. Son, K.-J. Moon, B. H. Lee, J.-M. Myoung, M. S. Strano, M.-H. Ham, C. A. Ross, *Adv. Mater.* **2013**, 25, 34.
8. H. Tsai, J. W. Pitera, H. Miyazoe, S. Bangsaruntip, S. U. Engelmann, C.-C. Liu, J. Y. Cheng, J. J. Bucchignano, D. P. Klaus, E. A. Joseph, D. P. Sanders, M. E. Colburn, M. A. Guillorn, *ACS Nano* **2014**, 8, 5.
9. A. Nourbakhsh, A. Zubair, R. N. Sajjad, A. Tavakkoli K. G., W. Chen, S. Fang, X. Ling, J. Kong, M. S. Dresselhaus, E. Kaxiras, K. K. Berggren, D. Antoniadis, T. Palacios, *Nano Lett.* **2016**, 16, 12.
10. S. M. Lindner, S. Hüttner, A. Chiche, M. Thelakkat, G. Krausch, *Angew. Chem. Int. Ed.* **2006**, 45, 20.
11. S. B. Darling, *Energy Environ. Sci.* **2009**, 2.
12. C. Guo, Y.-H. Lin, M. D. Witman, K. A. Smith, C. Wang, A. Hexemer, J. Strzalka, E. D. Gomez, R. Verduzco, *Nano Lett.* **2013**, 13, 6.
13. C. Hägglund, G. Zeltzer, R. Ruiz, I. Thomann, H.-B.-R. Lee, M. L. Brongersma, S. F. Bent, *Nano Lett.* **2013**, 13, 7.
14. S. K. Cha, J. H. Mun, T. Chang, S. Y. Kim, J. Y. Kim, H. M. Jin, J. Y. Lee, J. Shin, K. H. Kim, S. O. Kim, *ACS Nano* **2015**, 9, 5.
15. P. A. Mistark, S. Park, S. E. Yalcin, D. H. Lee, O. Yavuzcetin, M. T. Tuominen, T. P. Russell, M. Achermann, *ACS Nano* **2009**, 3, 12.
16. C. T. Black, O. Bezencenet, *IEEE Trans. Nanotechnol.* **2004**, 3, 3.
17. T. Yamaguchi, H. Yamaguchi, *Adv. Mater.* **2008**, 20, 9.
18. J. K. W. Yang, Y. S. Jung, J.-B. Chang, R. A. Mickiewicz, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Nanotechnol.* **2010**, 5.
19. J.-B. Chang, H. K. Choi, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Commun.* **2014**, 5, 3305.
20. M. P. Stoykovich, M. Müller, S. O. Kim, H. H. Solak, E. W. Edwards, J. J. de Pablo, P. F. Nealey, *Science* **2005**, 308, 5727.
21. G. M. Wilmes, D. A. Durkee, N. P. Balsara, J. A. Liddle, *Macromolecules* **2006**, 39, 7.
22. M. P. Stoykovich, H. Kang, K. C. Daoulas, G. Liu, C.-C. Liu, J. J. de Pablo, M. Müller, P. F. Nealey, *ACS Nano* **2007**, 1, 3.
23. J. Y. Cheng, C. T. Rettner, D. P. Sanders, H.-C. Kim, W. D. Hinsberg, *Adv. Mater.* **2008**, 20, 16.
24. G. Liu, C. S. Thomas, G. S. W. Craig, P. F. Nealey, *Adv. Funct. Mater.* **2010**, 20, 8.
25. I. Bita, J. K. W. Yang, Y. Jung, C. A. Ross, E. L. Thomas, K. K. Berggren, *Science* **2008** 321, 5891.
26. A. Tavakkoli K. G., S. M. Nicaise, A. F. Hannon, K. W. Gotrik, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Small* **2014**, 10, 3.

27. A. Tavakkoli K. G., K. W. Gotrik, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Science* **2012**, 336, 6086.

28. D. R. Reyes, M. M. Ghanem, G. M. Whitesides, A. Manz, *Lab Chip* **2002**, 2, 2.

29. L. M. Adelman, *Science* **1994**, 266, 5187.

30. D. G. Cory, A. F. Fahmy, T. F. Havel, *Proc. Natl. Acad. Sci. U.S.A.* **1997**, 94, 5.

31. R. Vrijen, E. Yablonovitch, K. Wang, H. W. Jiang, A. Balandin, V. Roychowdhury, T. Mor, D. DiVincenzo, *Phys. Rev. A* **2000**, 62, 1.

32. M. N. Leuenberger, D. Loss, *Nature* **2001**, 410.

33. H. J. Caulfield, S. Dolev, *Nat. Photonics* **2010**, 4.

34. M. Asghari, A. V. Krishnamoorthy, *Nat. Photonics* **2011**, 5.

35. L. Larger, M. C. Soriano, D. Brunner, L. Appeltant, J. M. Gutierrez, L. Pesquera, C. R. Mirasso, I. Fischer, *Opt. Express* **2012**, 20, 3.

36. D. A. Allwood, G. Xiong, M. D. Cooke, C. C. Faulkner, D. Atkinson, N. Vernier, R. P. Cowburn, *Science* **2002**, 296, 5575.

37. D. A. Allwood, G. Xiong, C. C. Faulkner, D. Atkinson, D. Petit, R. P. Cowburn, *Science* **2005**, 309, 5741.

38. A. Imre, G. Csaba, L. Ji, A. Orlov, G. H. Bernstein, W. Porod, *Science* **2006**, 311, 5758.

39. A. J. Heinrich, C. P. Lutz, J. A. Gupta, D. M. Eigler, *Science* **2002**, 298, 5597.

40. R. J. Lipton, *Science* **1995**, 268, 5210.

41. Q. Ouyang, P. D. Kaplan, S. Liu, A. Libchaber, *Science* **1997**, 278, 5337.

42. Q. Liu, L. Wang, A. G. Frutos, A. E. Condon, R. M. Corn, L. M. Smith, *Nature* **2000**, 403.

43. T. Head, G. Rozenberg, R. S. Bladergroen, C. K. D. Breek, P. H. M. Lommerse, H. P. Spaink, *BioSystems* **2000**, 57, 2.

44. P. W. K. Rothemund, N. Papadakis, E. Winfree, *PLoS Biol.* **2004**, 2, 12.

45. J. Elbaz, O. Lioubashevski, F. Wang, F. Remacle, R. D. Levine, I. Willner, *Nat. Nanotechnol.* **2010**, 5.

46. M. Dorigo, L. M. Gambardella, *IEEE Trans. Evol. Comput.* **1997**, 1, 1.

47. S. Tsuda, M. Aono, Y.-P. Gunji, *BioSystems* **2004**, 73, 1.

48. A. Adamatzky, *Philos. Trans. Royal Soc. A* **2015**, 373, 2046.

49. R. Mayne, A. Adamatzky, *PLoS One* **2015**, 10, 10.

50. M. Prakash, N. Gershenfeld, *Science* **2007**, 315, 5813.

51. K. Suzuno, D. Ueyama, M. Branicki, R. Tóth, A. Braun, I. Lagzi, *Langmuir* **2014**, 30, 31.

52. Z. Wang, A. Marandi, K. Wen, R. L. Byer, Y. Yamamoto, *Phys. Rev. A* **2013**, 88, 6.

53. A. Marandi, Z. Wang, K. Takata, R. L. Byer, Y. Yamamoto, *Nat. Photonics* **2014**, 8.

54. T. Inagaki, K. Inaba, R. Hamerly, K. Inoue, Y. Yamamoto, H. Takesue, *Nat. Photonics* **2016**, 10.

55. M. Yamaoka, C. Yoshimura, M. Hayashi, T. Okuyama, H. Aoki, H. Mizuno, *Hitachi Review* **2015**, 64, 8.

56. M. Yamaoka, C. Yoshimura, M. Hayashi, T. Okuyama, H. Aoki, H. Mizuno, *IEEE J. Solid-State Circuits* **2016**, 51, 1.

57. T. F. Rønnow, Z. Wang, J. Job, S. Boixo, S. V. Isakov, D. Wecker, J. M. Martinis, D. A. Lidar, M. Troyer, *Science* **2014**, 345, 6195.

58. B. Heim, T. F. Rønnow, S. V. Isakov, M. Troyer, *Science* **2015**, 348, 6231.

59. B.-D. Yang, L.-S. Kim, *IEEE J. Solid-State Circuits* **2005**, 40, 8.

60. B. Murmann, *IEEE Micro* **2006**, 26, 2.

61. P. Corsonello, M. Lanuzza, S. Perri, *Int. J. Circuit Theory Appl.* **2014**, 42, 1.

62. F. Reincke, W. K. Kegel, H. Zhang, M. Nolte, D. Wang, D. Vanmaekelbergh, H. Möhwald, *Phys. Chem. Chem. Phys.* **2006**, 8, 33.

# Chapter 2

# Directed self-assembly of a two-state block copolymer system

In this chapter, we study directed self-assembly of block copolymer thin films within templates of different polygonal shapes. Ladder-shaped block copolymer structures consisting of parallel bars, bends, and T-junctions are formed inside square confinement. We define binary states by the two degenerate alignment orientations, and study properties of the two-state system such as distribution of the states, nearest-neighbor correlation, and defect tolerance. We control the binary states by changing the confinement geometry or placing lithographic guiding patterns inside the confinement. The resulting block copolymer patterns could potentially act as a physical read-only memory in addition to functioning as a lithography mask.

## 2.1   Introduction

Block copolymer self-assembly in thin films can spontaneously generate periodic nanoscale patterns such as hexagonal arrays of dots or parallel lines, which have been proposed for applications such as nanoporous filtration membranes,[1,2] plasmonic structures,[3,4] integrated circuit fabrication,[5-7] and magnetic storage media.[8-10] Many of these applications require the nanoscale features to have long-range order or to form specific non-periodic structures with low

defect density. Directed self-assembly (DSA) addresses these issues by using graphoepitaxial[11-14] and/or chemoepitaxial[15-19] templates, fabricated by conventional lithography techniques, to guide the self-assembly of thin films of block copolymers. Various microelectronic device-oriented features such as concentric rings, bends, jogs, terminations, and T-junctions have been made, and these patterns have subsequently been transferred into functional materials to fabricate structures such as metal nanowire ring arrays[20-22] or parallel fins for field-effect transistors.[7,23,24]

Common templates used for DSA include one-dimensional features (trenches or chemical stripe patterns), or two-dimensional features (pits or chemically patterned regions). Although the templating effect from trench confinement has been well studied,[25-27] two-dimensional templates provide a wider set of geometries to guide block copolymer self-assembly, and can lead to formation of multiple degenerate structures. For example, concentric ring structures have been self-assembled inside symmetric confinements,[20,21] and we recently demonstrated nanoscale Archimedean spirals with specific chirality formed inside circular pits.[28] By studying such block copolymer systems that have energetic degeneracy, graphoepitaxial pattern control inside two-dimensional templates can be better understood. Moreover, by assigning different states or bits to the two degenerate morphologies, the resulting block copolymer patterns could act as a physical read-only memory.

This chapter describes DSA of block copolymer films within templates of different polygonal shapes. In square templates, two degenerate morphologies can form, and the presence of asymmetry inside the templates breaks the degeneracy. We describe the properties of the binary states including distribution, correlation, and defect tolerance, and present methods for controlling the binary state orientations.

## 2.2 Experimental methods

To control the self-assembly in block copolymer thin films, we used electron-beam lithography to fabricate templates of different polygonal shapes, and solvent annealing to facilitate the block copolymer self-assembly process. In this section, we describe the experimental methods for template fabrication, block copolymer self-assembly, reactive-ion etching, and metrology.

### 2.2.1 Template fabrication

The topographic templates were fabricated using electron-beam lithography with a hydrogen silsesquioxane (HSQ) resist. A silicon substrate was spin coated with 42-nm-thick HSQ film (XR-1541 2% solids, Dow Corning). The thickness was determined by ellipsometry. A Raith 150 electron-beam lithography system operated at 30 kV acceleration voltage was used to expose topographic features with various geometries. After exposure, the samples were developed in a 24°C high contrast salty developer (1% NaOH and 4% NaCl in de-ionized water) for 4 min, rinsed in de-ionized water for 3 min, and blow dried with $N_2$ gas.[29] Template dimensions were inspected by scanning electron microscope (SEM) imaging. Templates for 16 kg/mol poly(styrene-*block*-dimethylsiloxane) (PS-*b*-PDMS) were fabricated using 30-nm-thick HSQ film.

### 2.2.2 Block copolymer self-assembly

To make the templates attractive to the majority PS block, the templates were chemically functionalized with a PS brush (11 kg/mol, Polymer Source Inc.) by spin coating 1% brush solution in propylene glycol monomethyl ether acetate (PGMEA) and annealing the samples in a vacuum oven at 170°C for 14 h. The samples were rinsed with toluene for 1 min after annealing to remove excess PS brush. The resulting thickness of the PS brush bonded to the substrate was 5 nm. Next,

2% PS-*b*-PDMS ($M_w$ = 45.5 kg/mol, $f_{PDMS}$ = 32%, PDI = 1.08, Polymer Source Inc.) solution in PGMEA was spin coated onto the templated substrate. The resulting film thickness was 27 nm. The samples were solvent annealed using a 5:1 mixture of toluene and heptane at room temperature for 5 h. We placed the samples on a glass slide stack (0.8 cm in height) inside a crystallization dish (1.5 cm in height, 5 cm in diameter) and added 1.5 ml of the 5:1 toluene and heptane mixture. The chamber was covered with a petri dish (10 cm in diameter). During the 5 h annealing, leakage of solvent vapor occurred at a rate of 585 µg/min. 16 kg/mol PS-*b*-PDMS ($f_{PDMS}$ = 31%, PDI = 1.08, Polymer Source Inc.) was spin coated to a thickness of 25 nm and thermally annealed in a vacuum oven at 150°C for 14 h.

### 2.2.3  Reactive-ion etching

Reactive-ion etching of the annealed block copolymer film was performed in two steps. First, the top PDMS wetting layer was removed using a 5 s $CF_4$ plasma treatment with a power of 50 W and pressure of 15 mTorr. Next, the PS matrix was removed using a 22 s $O_2$ plasma treatment with a power of 90 W and pressure of 6 mTorr. This step also oxidized the PDMS cylinders. For 16 kg/mol PS-*b*-PDMS, the $CF_4$ plasma was applied for 3 s and $O_2$ plasma was applied for 12 s.

### 2.2.4  Metrology

Metrology was performed by examining the HSQ templates and the reactive-ion etched block copolymer films using a SEM. Top down SEM images were obtained using a Raith 150 SEM operated at 10 kV acceleration voltage and 6 mm working distance, and Zeiss Sigma SEM operated at 3 kV acceleration voltage and 4 mm working distance.

## 2.3   Results and discussion

In this section, we describe our approach for defining the basis states inside lithographic confinement, and discuss different methods for controlling the binary states.

### 2.3.1  Defining the basis states inside lithographic confinement

We first demonstrate the morphologies of a block copolymer film within polygonal templates. The block copolymer is a 45.5 kg/mol cylindrical morphology PS-*b*-PDMS (SD45). Thin films of SD45 microphase separate into a layer of PDMS cylinders with in-plane orientation surrounded by a PS matrix, and a wetting layer of PDMS at the air interface. Electron-beam lithography was performed using an HSQ resist on silicon substrates to fabricate topographic features of various geometries. The topographic templates were chemically functionalized with a hydroxyl-terminated PS brush. SD45 block copolymer was spin coated onto the substrate to a thickness of 27 nm, solvent annealed in a vapor of toluene and heptane, and reactive-ion etched to reveal a pattern consisting of oxidized PDMS cylinders.

Figure 2.1 shows an example of oxidized PDMS patterns without any template (**Figure 2.1a**) and within polygonal confinement (**Figures 2.1b-2.1f**). On an untemplated substrate, the periodicity of the PDMS cylinders ($L_0$) was ~36 nm. As shown in **Figures 2.1b-2.1d**, the PDMS cylinders formed a one-state system of concentric rings inside circular, hexagonal, and pentagonal confinement. However, the PDMS cylinders formed a two-state system inside square confinement and a three-state system inside triangular confinement (**Figure 2.1e and Figure 2.1f**). For both confinements, bars parallel to one of the sides were formed inside an outer ring, creating 90° T-junctions for square confinement and 60° Y-junctions for triangular confinement. As the interior angle is decreased, high deformation is imposed on the microdomains at the corners,[15,17] which

is relieved by transitioning to a pattern of parallel bars instead of concentric rings. For square confinement, orientation of the parallel bars was restricted to either the horizontal or vertical direction, and these degenerate states are defined as the basis states of the system.
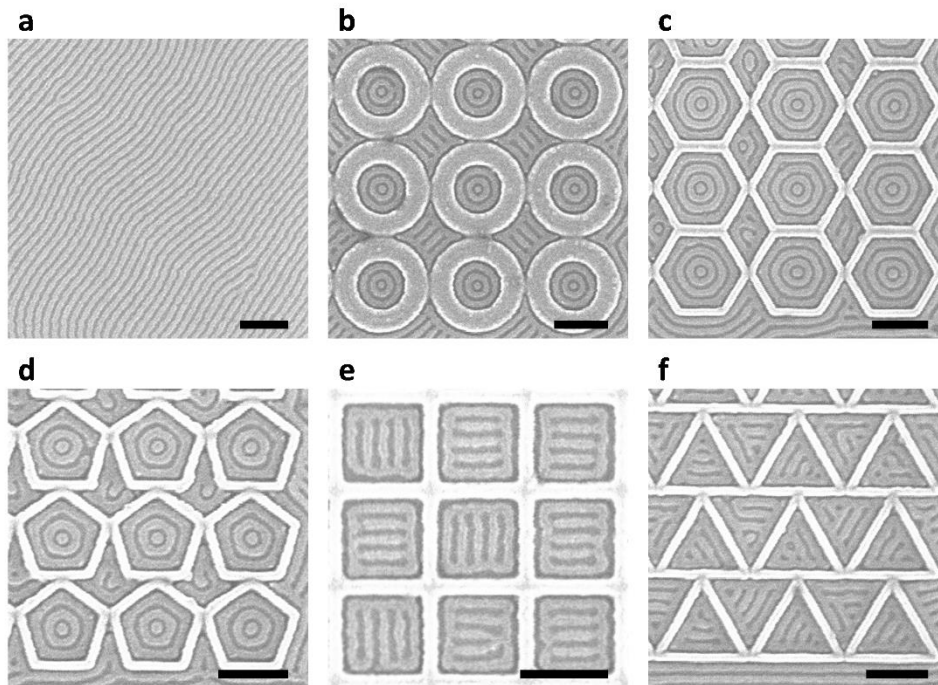


**Figure 2.1** SEM images of untemplated and templated block copolymer patterns. The HSQ templates were functionalized with the majority PS block. (a) Untemplated PDMS cylinders with $L_0$ = ~36 nm. (b) One-state system with concentric rings inside circular confinement. Radius was $2.4L_0$. (c) One-state system inside hexagonal confinement. Apothem was $3.5L_0$. (d) One-state system inside pentagonal confinement. Apothem was $2.5L_0$. (e) Two-state system with degenerate ladder-shaped structures inside square confinement. Apothem was $2.4L_0$. (f) Three-state system inside triangular confinement. Apothem was $1.8L_0$. The radius and apothem were measured by subtracting brush thickness from confinement dimensions. Scale bars, 200 nm.

Inside circular, hexagonal, and pentagonal confinement, we observed random formation of defective structures consisting of Archimedean spiral patterns instead of concentric rings **(Figure 2.2)**. Choi et al. have reported experimental and simulation work on formation of the spiral structures within a circular template, and demonstrated chirality control using notched features.[28]

Investigation of a two-state system which uses chirality of the spiral structures as the basis states may be of interest for future research.
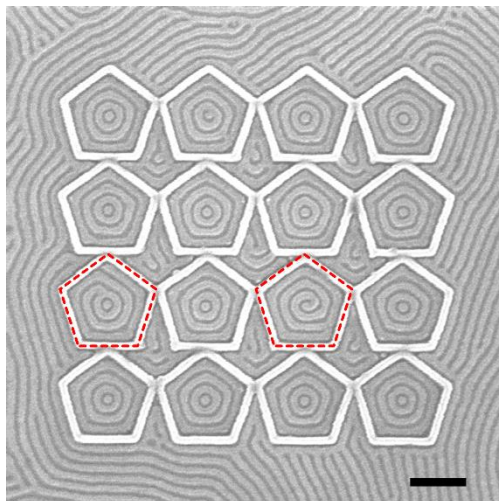


**Figure 2.2** SEM image of block copolymer patterns formed inside pentagonal confinement. As indicated by the red dashed lines, partial (left) and complete (right) Archimedean spiral patterns were observed at random locations. Similar spiral patterns were also observed in circular and hexagonal confinement. Scale bar, 200 nm.

We focus our study on the square confinement since it resulted in a well-defined two-state system with 90° bends and T-junctions. Commensurability is achieved when the width of the confinement minus the brush thickness is equivalent to an integer multiple of $L_0$. **Figure 2.3** shows the ladder-shaped block copolymer patterns formed inside square templates, with the number of parallel bars increasing with confinement dimensions. The smallest templates produced a one-state system consisting of a single ring **(Figure 2.3a)**, then a PDMS sphere was formed inside the outer ring as the confinement dimension was increased between $2L_0$ and $3L_0$ **(Figure 2.3b)**. In this regime, both ladder-shaped structures (two-state system) and concentric ring structures (one-state system) were observed. With increasing dimensions the interior spheres were either horizontally or vertically connected to the outer ring resulting in a two-state system, then an additional bar was

formed inside the outer ring **(Figures 2.3c-2.3h)**. For larger incommensurate templates, ladder-shaped structures were still produced, but the number of parallel bars varied by one from structure to structure.
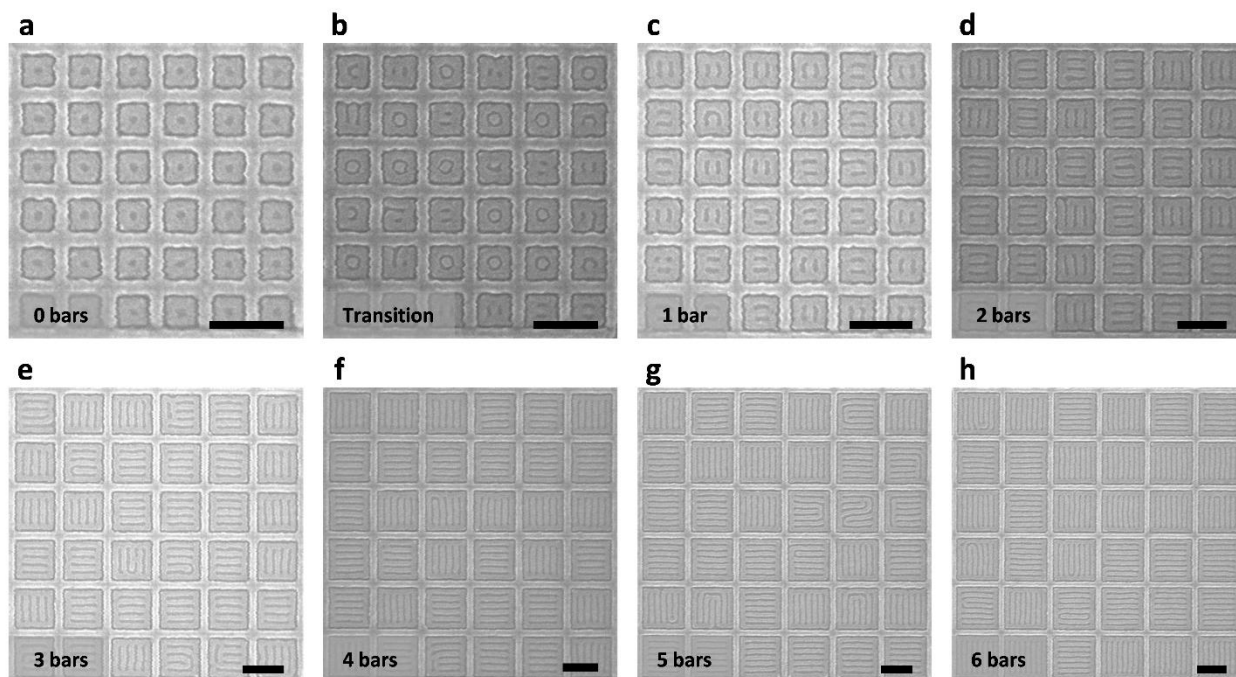


**Figure 2.3** SEM images of ladder-shaped block copolymer patterns inside square confinement. Figure 2.3a and Figures 2.3c-2.3h show commensurate conditions while Figure 2.3b shows transition between a one-state system (Figure 2.3a) and a two-state system (Figure 2.3c) inside incommensurate confinement dimensions. Width of the square confinement was (a) $2.2L_0$, (b) $2.8L_0$, (c) $3.0L_0$, (d) $4.1L_0$, (e) $5.1L_0$, (f) $6.1L_0$, (g) $7.1L_0$, and (h) $8.1L_0$ ($L_0 = 36$ nm). Depending on the confinement width, 0 to 6 parallel bars were formed inside an outer ring. Scale bars, 200 nm.

## 2.3.2 Properties of the two-state system

To show that the two basis states are degenerate, we created arrays of 10 by 10 square templates and measured the distribution of horizontally aligned and vertically aligned ladder-shaped structures. For simplicity, we defined horizontal alignment as the '0' state and vertical alignment as the '1' state. Among 600 examined structures, 51.5% were in 0 state and 48.5% were in 1 state,

forming with essentially equal probability. For a null hypothesis $H_0$: $p = 0.5$ and alternative

hypothesis $H_1$: $p \neq 0.5$ where $p$ denotes ratio of 1 state, the Z-test statistic was 0.735. The

corresponding $p$-value was 0.462, and we failed to reject the null hypothesis at 5% level of

significance. In addition, the binary states had tolerance to defects in the sense that even with

defects present the patterns could be assigned as 0 or 1 **(Figure 2.4a)**.
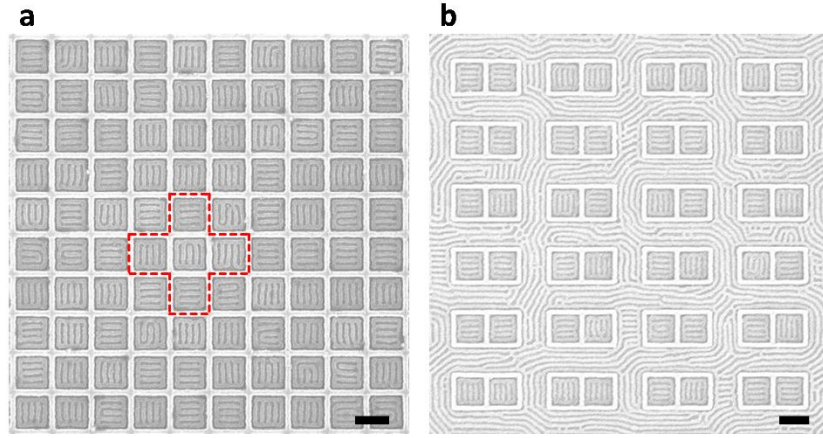


**Figure 2.4** SEM images of ladder-shaped block copolymer patterns inside square confinement for measuring distribution and correlation. (a) The two binary states were equally probable in a large array of square confinements. (b) The two states were uncorrelated in pairs of adjacent square confinements. For both figures, width of the square confinement was $5.1L_0$, resulting in PDMS patterns with 3 parallel bars inside an outer ring. Scale bars, 200 nm.

Next, we investigated whether the binary state of the four neighbors was correlated with

the binary state of the surrounded square. For each square not positioned on the boundary of the

square array, there were four adjacent squares as indicated by the red dashed line **(Figure 2.4a)**.

The normalized mean state-state correlation is

$$\rho = \frac{\sum_{i \sim j} s_i s_j}{\sum_{i \sim j} |s_i s_j|} = 0.00078$$

where $s_k = +1$ for 1 state, $s_k = -1$ for 0 state, and the sum was taken over every pair of adjacent states. The negligible value of correlation suggests that there is no nearest-neighbor influence.

To investigate the correlation between isolated pairs of adjacent square confinements in samples shown in **Figure 2.4b**, we define $n_{XY}$ as the number of cases where the left binary state is $X$ (0 or 1) and the right binary state is $Y$ (0 or 1). For 576 structures, the resulting counts were $n_{00}$ = 144 (25.0%), $n_{01} = 137$ (23.8%), $n_{10} = 143$ (24.8%), and $n_{11} = 149$ (25.9%) with 3 (0.5%) defects. The $\phi$ coefficient calculated as

$$\phi = \frac{n_{00}n_{11} - n_{01}n_{10}}{\sqrt{(n_{00} + n_{01})(n_{00} + n_{10})(n_{01} + n_{11})(n_{10} + n_{11})}} = 0.023$$

was close to zero, indicating negligible association between two adjacent states in isolated pairs of square confinements. A similar set of samples made with a template wall height of 30 nm instead of 42 nm yielded $\phi = -0.03$, again indicating negligible association. For wall height below 30 nm, the PDMS cylinders crossed the walls leading to poorly defined block copolymer structures within the templates.

### 2.3.3 Methods for controlling the binary states

Having established the non-interacting binary-state system described above, we now discuss methods to control the alignment of the states. A simple method for controlling the orientation of the binary states is by changing the confinement to a rectangular shape. **Figure 2.5** shows ladder-shaped block copolymer patterns formed inside rectangular confinements with an aspect ratio of 2:1. Similar to the square confinement, we were able to accurately control the number of parallel bars using confinement dimensions. When the vertical confinement width was commensurate with $L_0$, the horizontal confinement width was also commensurate with $L_0$ since the aspect ratio was an

integer. However, we observed only 0 state at these conditions.
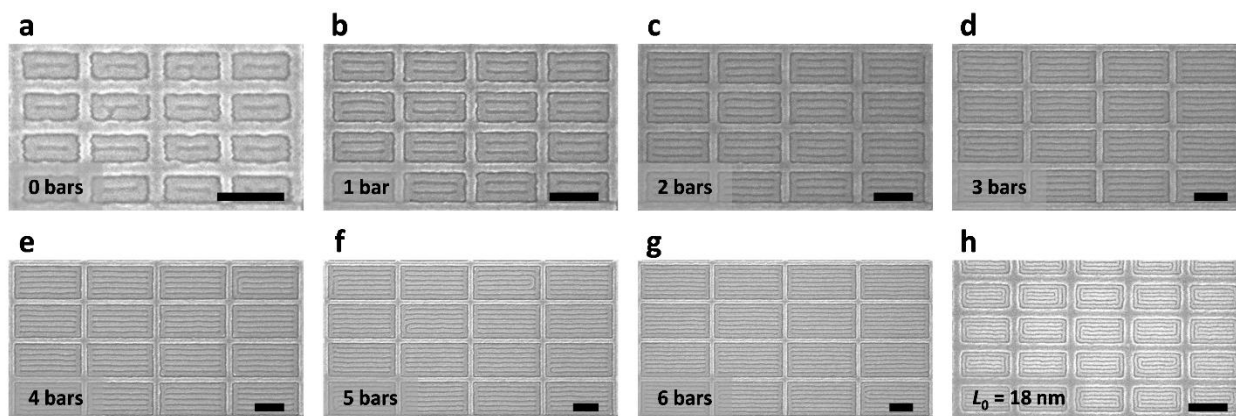


**Figure 2.5** SEM images of aligned ladder-shaped block copolymer patterns inside rectangular confinement. Aspect ratio was 2:1. Vertical width of the rectangular confinement was (a) $2.0L_0$, (b) $3.1L_0$, (c) $4.1L_0$, (d) $5.1L_0$, (e) $6.1L_0$, (f) $7.0L_0$, and (g) $7.9L_0$ for SD45 ($L_0 = 36$ nm). For Figure 2.5h, 16 kg/mol PS-$b$-PDMS with $L_0 = 18$ nm was used. Parallel bars were formed in the horizontal direction (0 state) to minimize the number of T-junctions. Scale bars, 200 nm.
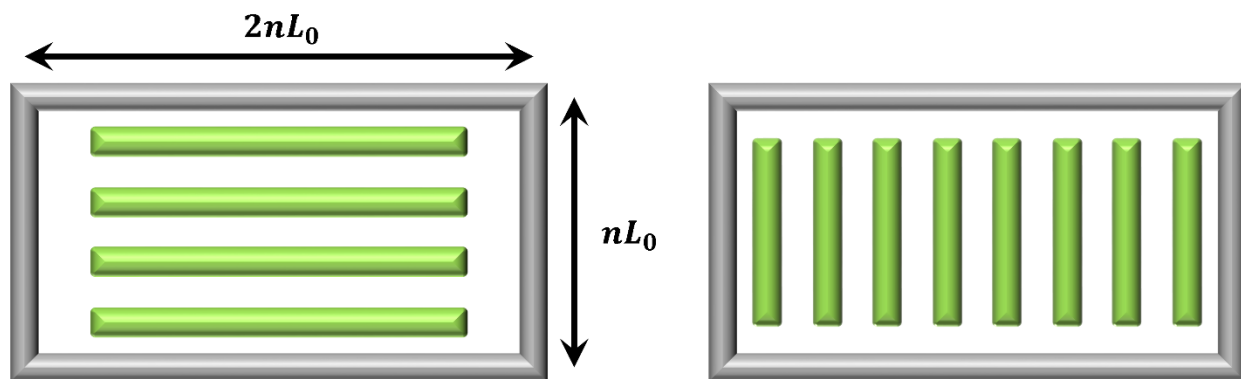


**Figure 2.6** Illustration of horizontally (left) and vertically (right) aligned ladder-shaped structure inside rectangular confinement with confinement dimensions equal to $2nL_0$ and $nL_0$ ($n$ is an integer). Only parts of the ladder-shaped structure that are parallel to the alignment orientation are shown. When $n = 4$, four T-junctions are formed inside a horizontally aligned ladder-shaped structure, whereas twelve T-junctions are formed inside a vertically aligned ladder-shaped structure. To minimize the number of T-junctions, alignment orientation parallel to the longer sidewall is favored.

**Figure 2.6** illustrates the number of T-junctions formed inside a ladder-shaped structure in

horizontal and vertical alignment. When the horizontal and vertical confinement widths are equal to $2nL_0$ and $nL_0$, respectively, a ladder-shaped structure in the 0 state results in $2n-4$ T-junctions ($n$ is an integer). On the other hand, a ladder-shaped structure in the 1 state results in $4n-4$ T-junctions. Since T-junctions are energetically unfavorable,[30-32] the 0 state was favored over the 1 state to minimize the number of T-junctions. The preferential alignment can also be understood as the longer sidewall having a stronger templating effect compared to the shorter sidewall, analogous to the perpendicular orientation of lamellar morphology PS-$b$-PDMS observed within deep trenches functionalized with a preferential sidewall brush and a neutral bottom surface.[33]

In **Figure 2.7**, the horizontal and vertical dimensions of the confinement were approximately commensurate, $4.1L_0$ and $2.9L_0$, respectively, and the structure formed two T-junctions instead of four T-junctions. Non-integer aspect ratios in which only the shorter dimension satisfies the commensurability condition can also be used to further promote alignment **(Figure 2.7b)**. As the aspect ratio increases, the confinement approximates a trench leading to well-ordered microdomains parallel to the sidewalls.
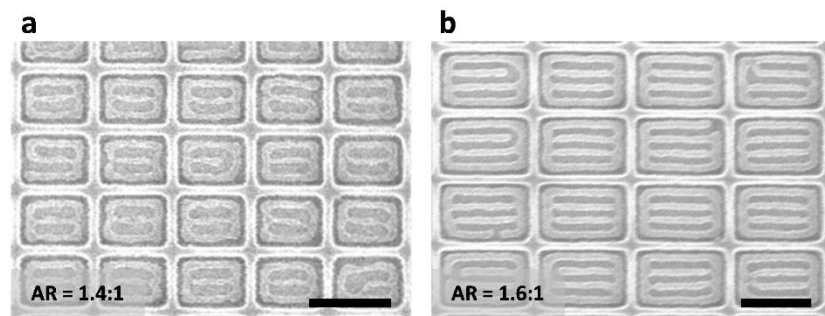


**Figure 2.7** SEM images of aligned ladder-shaped block copolymer patterns inside rectangular confinement with a non-integer aspect ratio (AR). Vertical width of the rectangular confinement was (a) $2.9L_0$ and (b) $4.1L_0$, approximately commensurate with the equilibrium periodicity. Horizontal width of the confinement was (a) $4.1L_0$ and (b) $6.6L_0$. Aspect ratio was (a) 1.4:1 and (b) 1.6:1. Scale bars, 200 nm.

This approach was extended to fabricate aligned T-junctions with sub-10-nm spacing using a 16 kg/mol cylindrical morphology PS-*b*-PDMS with $L_0$ = 18 nm, thermally annealed on a functionalized patterned substrate. As shown in **Figure 2.5h**, sub-10-nm half-pitch ladder-shaped structures were formed inside rectangular confinement with the microdomains primarily parallel to the longer side. Unlike SD45, the ladder-shaped structures were influenced by the curvature of the corners of the template due to the smaller $L_0$, and a complete ring was formed between the confinement and the ladder-shaped structure.

Preferential alignment is observed in non-rectangular geometries such as trapezoids and isosceles triangles, as shown in **Figure 2.8**. For trapezoidal confinement, T-junctions formed with desired bending angles because the microdomains aligned parallel to the longer side. Isosceles triangles typically showed preferential alignment parallel to either of the two longer sides. This produced two T-junctions whereas alignment parallel to the shorter side resulted in four T-junctions. Thus the confinement geometry determines the number of states, i.e. a three-state system in equilateral triangles **(Figure 2.1f)**, a two-state system in acute isosceles triangles or possibly a one-state system in obtuse isosceles triangles.
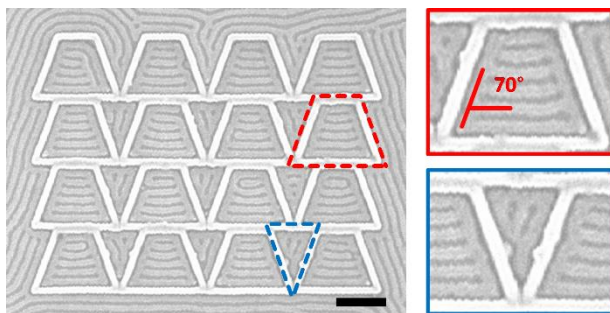


**Figure 2.8** SEM images of aligned ladder-shaped block copolymer patterns inside trapezoidal (red) and isosceles triangular (blue) confinement. The ladder-shaped structures were typically aligned parallel to the longer side to minimize T-junction formations. Vertical width of the confinement was 5.8$L_0$. Scale bar, 200 nm.

An alternative method for controlling the orientation of the binary states is by placing lithographically defined guiding patterns inside the confinement. The effect of posts, dashes, or walls has been previously studied in detail.[13,14,34] **Figure 2.9** shows square confinements, each with two horizontal HSQ walls where the walls were positioned a distance $L_0$ away from the edges. Because two PDMS bars in the ladder-shaped block copolymer structures were replaced with the horizontal HSQ walls functionalized with the majority PS block, all block copolymer patterns were also horizontally aligned and set to the 0 state.
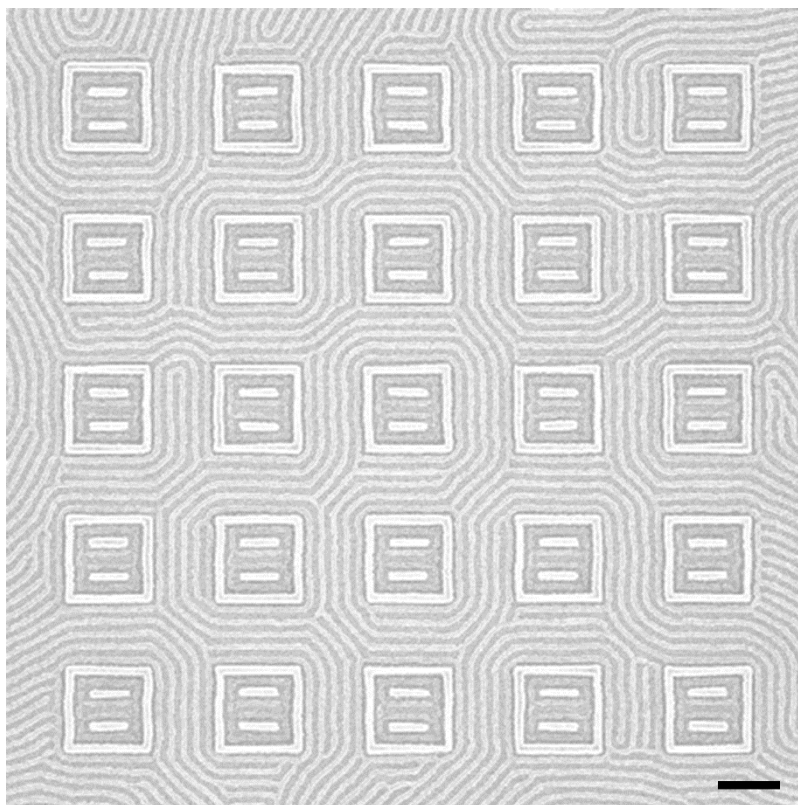


**Figure 2.9** SEM image of aligned ladder-shaped block copolymer patterns inside square confinement with horizontal guiding patterns. Without the guiding patterns, ladder-shaped structures with four horizontally or vertically aligned PDMS bars were formed with equal probability. However, by placing functionalized HSQ walls where the PDMS bars should form, alignment orientation was controlled. Scale bar, 200 nm.

## 2.4 Conclusion

The self-assembly of block copolymers inside discrete and interacting polygonal templates is investigated. Square and triangular confinement with dimensions of a few $L_0$ produced ladder-shaped structures instead of the concentric rings seen in smaller confinements or in circular pits. In square confinement, the two degenerate orientations of the ladder-shaped structures could be considered as independently controlled binary states with tolerance to defects. The binary states were selected by either changing the confinement aspect ratio or placing additional lithographic features inside the confinement. The resulting line segments, bends, and T-junctions composing the ladder-shaped structures may be useful as circuit-relevant geometries or binary information storage. Although the multi-state composite structures used in our work are less than 1 µm in dimension, we expect larger sizes to yield similar result. If the binary states could be read out optically or electrically in a very large array, the block copolymer patterns could be used to physically store information. In the next chapter, we use the two-state system designed in this chapter and place openings on the square confinement walls to allow neighboring states to be connected. We investigate the effect of wall openings on the two-state system.

# References

1. H. Uehara, M. Kakiage, M. Sekiya, D. Sakuma, T. Yamonobe, N. Takano, A. Barraud, E. Meurville, P. Ryser, *ACS Nano* **2009**, 3, 4.
2. W. A. Phillip, B. O'Neill, M. Rodwogin, M. A. Hillmyer, E. L. Cussler, *ACS Appl. Mater. Interfaces* **2010**, 2, 3.
3. C. Hägglund, G. Zeltzer, R. Ruiz, I. Thomann, H.-B.-R. Lee, M. L. Brongersma, S. F. Bent, *Nano Lett.* **2013**, 13, 7.
4. S. K. Cha, J. H. Mun, T. Chang, S. Y. Kim, J. Y. Kim, H. M. Jin, J. Y. Lee, J. Shin, K. H. Kim, S. O. Kim, *ACS Nano* **2015**, 9, 5.
5. H. Yi, X.-Y. Bao, J. Zhang, C. Bencher, L.-W. Chang, X. Chen, R. Tiberio, J. Conway, H. Dai, Y. Chen, S. Mitra, H.-S. P. Wong, *Adv. Mater.* **2012**, 24, 23.
6. H. Yi, X.-Y. Bao, R. Tiberio, H.-S. P. Wong, *Nano Lett.* **2015**, 15, 2.
7. H. Tsai, J. W. Pitera, H. Miyazoe, S. Bangsaruntip, S. U. Engelmann, C.-C. Liu, J. Y. Cheng, J. J. Bucchignano, D. P. Klaus, E. A. Joseph, D. P. Sanders, M. E. Colburn, M. A. Guillorn, *ACS Nano* **2014**, 8, 5.
8. X. Yang, L. Wan, S. Xiao, Y. Xu, D. K. Weller, *ACS Nano* **2009**, 3, 7.
9. O. Hellwig, J. K. Bosworth, E. Dobisz, D. Kercher, T. Hauet, G. Zeltzer, J. D. Risner-Jamtgaard, D. Yaney, R. Ruiz, *Appl. Phys. Lett.* **2010**, 96, 5.
10. R. Ruiz, E. Dobisz, T. R. Albrecht, *ACS Nano* **2011**, 5, 1.
11. C. T. Black, O. Bezencenet, *IEEE Trans. Nanotechnol.* **2004**, 3, 3.
12. T. Yamaguchi, H. Yamaguchi, *Adv. Mater.* **2008**, 20, 9.
13. J. K. W. Yang, Y. S. Jung, J.-B. Chang, R. A. Mickiewicz, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Nanotechnol.* **2010**, 5.
14. J.-B. Chang, H. K. Choi, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Commun.* **2014**, 5, 3305.
15. G. M. Wilmes, D. A. Durkee, N. P. Balsara, J. A. Liddle, *Macromolecules* **2006**, 39, 7.
16. J. Y. Cheng, C. T. Rettner, D. P. Sanders, H.-C. Kim, W. D. Hinsberg, *Adv. Mater.* **2008**, 20, 16.
17. M. P. Stoykovich, M. Müller, S. O. Kim, H. H. Solak, E. W. Edwards, J. J. de Pablo, P. F. Nealey, *Science* **2005**, 308, 5727.
18. M. P. Stoykovich, H. Kang, K. C. Daoulas, G. Liu, C.-C. Liu, J. J. de Pablo, M. Müller, P. F. Nealey, *ACS Nano* **2007**, 1, 3.
19. G. Liu, C. S. Thomas, G. S. W. Craig, P. F. Nealey, *Adv. Funct. Mater.* **2010**, 20, 8.
20. Y. S. Jung, W. Jung, C. A. Ross, *Nano Lett.* **2008**, 8, 9.
21. S.-J. Jeong, J. E. Kim, H.-S. Moon, B. H. Kim, S. M. Kim, J. B. Kim, S. O. Kim, *Nano Lett.* **2009**, 9, 6.
22. J. Chai, D. Wang, X. Fan, J. M. Buriak, *Nat. Nanotechnol.* **2007**, 2.
23. J. G. Son, M. Son, K.-J. Moon, B. H. Lee, J.-M. Myoung, M. S. Strano, M.-H. Ham, C. A. Ross, *Adv. Mater.* **2013**, 25, 34.
24. A. Nourbakhsh, A. Zubair, R. N. Sajjad, A. Tavakkoli K. G., W. Chen, S. Fang, X. Ling, J. Kong, M. S. Dresselhaus, E. Kaxiras, K. K. Berggren, D. Antoniadis, T. Palacios, *Nano Lett.* **2016**, 16, 12.
25. Y. S. Jung, C. A. Ross, *Nano Lett.* **2007**, 7, 7.
26. R. Ruiz, N. Ruiz, Y. Zhang, R. K. Sandstrom, C. T. Black, *Adv. Mater.* **2007**, 19, 16.
27. E. Han, H. Kang, C.-C. Liu, P. F. Nealey, P. Gopalan, *Adv. Mater.* **2010**, 22, 38.

28. H. K. Choi, J.-B. Chang, A. F. Hannon, J. K. W. Yang, K. K. Berggren, A. Alexander-Katz, C. A. Ross, *Nano Futures* **2017**, 1, 1.
29. J. K. W. Yang, K. K. Berggren, *J. Vac. Sci. Technol. B* **2007**, 25, 6.
30. S. P. Gido, E. L. Thomas, *Macromolecules* **1994**, 27, 21.
31. E. Burgaz, S. P. Gido, *Macromolecules* **2000**, 33, 23.
32. D. Duque, K. Katsov, M. Schick, *J. Chem. Phys.* **2002**, 117, 22.
33. W. Bai, K. Gadelrab, A. Alexander-Katz, C. A. Ross, *Nano Lett.* **2015**, 15, 10.
34. A. F. Hannon, Y. Ding, W, Bai, C. A. Ross, A. Alexander-Katz, *Nano Lett.* **2014**, 14, 1.

# Chapter 3

# Nearest-neighbor interactions via confinement wall opening

In this chapter, we study directed self-assembly of block copolymer thin films within square confinements with different number of openings placed around the confinement walls. We investigate templating effect of the wall openings on binary states defined inside the confinement. Self-consistent field theory simulations show the templating effect from the openings and reproduce the experimental results. We demonstrate scaling of a single binary state into a larger binary state array with individual binary state control. We discuss the effect of opening size variation on the nearest-neighbor interactions.

## 3.1   Introduction

In the previous chapter, we designed a two-state system using directed self-assembly (DSA) of block copolymer thin films within square confinements. Inside each confinement, a binary state was defined based on the two degenerate alignment orientations of the ladder-shaped structures. We measured properties of the two-state system such as distribution of the states and nearest-neighbor correlation, and demonstrated defect tolerance of the system. By studying the effect of changing the confinement geometry and placing lithographic guiding patterns inside the

confinement, we controlled the alignment orientation and formation of parallel bars, bends, and T-junctions. The resulting circuit-relevant geometries may be useful for fabrication of FinFETs.[1-8] In addition, if the binary states could be read out optically[9-13] or electrically[13-17] in a very large array after pattern transfer, the block copolymer patterns may be used to physically store and read out information.

Having established a defect-tolerant and non-interacting two-state system, we now study the effect of placing openings on the confinement walls. An opening placed on the wall separating two adjacent states allows the two states to be physically connected to each other. Then, a block copolymer pattern formed inside one confinement may influence the block copolymer morphology inside its adjacent confinement via interactions through the opening. At the same time, the opening itself is a narrow topographic feature that may introduce a strong templating effect, thereby dominating any effect caused by the nearest-neighbor interactions. Therefore, the templating effect of the openings on the binary states must be investigated to better understand graphoepitaxial pattern control inside templates of various geometries, and to ultimately design an interacting two-state system.

This chapter describes the effect of placing different number of openings around the square confinement walls. Both experimental and self-consistent field theory (SCFT) simulation results show strong templating effect from the wall openings. Using the openings as a new method for controlling the binary state orientations, we demonstrate the propagation of a single binary state into a larger array with orientation control. Finally, we discuss the effect of opening size variation on the nearest-neighbor interactions.

## 3.2 Experimental methods

For template fabrication, block copolymer self-assembly, reactive-ion etching, and metrology, the same experimental methods as described in Chapter 2 were used. A 45.5 kg/mol cylindrical morphology poly(styrene-*block*-dimethylsiloxane) (PS-*b*-PDMS) was used for all experiments in this chapter.

## 3.3 Simulation methods

The SCFT simulations were performed in collaboration with Karim R. Gadelrab and Prof. Alfredo Alexander-Katz at MIT.

We consider a monodispersed melt of $n$ A-B diblock copolymer of volume $V$, with each diblock molecule composed of $N$ segments. The $A$ and $B$ blocks consist of $fN$ and $(1 - f)N$ chain segments, respectively. The interaction between the dissimilar blocks is controlled by a Flory-Huggins parameter $\chi$. Within the mean-field approximation, the free energy of the system $F$ is expressed in terms of field variables

$$\frac{F}{nVk_{\mathrm{B}}T} = \frac{1}{V}\int dr(\chi\phi_A(r)\phi_B(r) - w_A(r)\phi_A(r) - w_B(r)\phi_B(r) - p(r)[1 - \phi_A(r) - \phi_B(r)])$$
$$- \ln Q[w_A, w_B]$$

where $\phi_\alpha(r)$ is the volume fraction of species $\alpha$ at position $r$. $Q[w_A, w_B]$ is the partition function of a non-interacting polymer in external fields $w_\alpha(r)$. The polymer is assumed to be incompressible, so the constraint $\phi_A(r) + \phi_B(r) = 1$ is enforced through a pressure field $p(r)$. The free energy $F$ is compared to the thermal energy $k_{\mathrm{B}}T$.

The single chain partition function can be evaluated as follows

$$Q = \frac{1}{V} \int dr q(r, 1)$$

where $q(r, s)$ is a restricted chain partition function (propagator) that could be calculated by solving a modified diffusion equation

$$\frac{\partial q}{\partial s} = \nabla^2 q(r, s) - w_A(r) q(r, s), \qquad 0 \le s < f$$

$$\frac{\partial q}{\partial s} = \nabla^2 q(r, s) - w_B(r) q(r, s), \qquad f \le s < 1$$

subjected to the initial condition $q(r, 0) = 1$. Since the two ends of the polymer are distinct, a complementary partition function $q^*(r, s)$ is defined similarly and satisfies the same modified diffusion equation with an initial condition $q(r, 1) = 1$. Here, we utilize $s$ as a chain contour variable in units of $N$. All lengths are expressed in units of the unperturbed radius-of-gyration of a polymer, $R_g = (Nb^2/6)^{1/2}$, where $b$ is the statistical segment length. The solution to the modified diffusion equation is conducted following the pseudo-spectral method.[18,19] An iterative relaxation of the fields towards their saddle-point values is implemented following the method by Sides et al.[20,21]

By evaluating $q(r, s)$ and its complementary, the segments' volume fractions can be determined as follows

$$\phi_A(r) = \frac{1}{Q} \int_0^f ds q(r, s) q^*(r, s)$$

$$\phi_B(r) = \frac{1}{Q} \int_f^1 ds q(r, s) q^*(r, s)$$

The numerical implementation of the SCFT for the entire template is performed on a 2D square grid of size $N_x = N_y = 240$ pixels (pixel size is $0.2 R_g$) with periodic boundary conditions in

both directions. The volume fraction $f$ and degree of incompatibility $\chi N$ are chosen such that striped domains (projection of in-plane cylinders in 2D) are generated. Hence, for the purpose of this work, we chose $f = 0.5$ and $\chi N = 12$.

The role of the DSA of the polymer domains is depicted through a masking method. A pressure potential $w_+ = (w_B + w_A)/2$ is imposed as a mask on the location of the walls to create excluded areas for the polymer. A magnitude of $w_+ = 10$ is applied on walls of thickness of six pixels. To incorporate the effect of surface preferentiality towards the majority block in experiments, an exchange potential $w_- = (w_B - w_A)/2 = 3.5$ is applied surrounding the walls with a thickness of four pixels to attract block $B$. The confining template is displaced from the boundaries of the computational domain by 70 pixels to minimize the effect of the mirror image (periodic boundary conditions) on the polymer domains near the walls.

## 3.4   Results and discussion

In this section, we describe experimental and SCFT simulation results showing the templating effect from confinement wall openings, and demonstrate scaling into a larger binary state array. We discuss the effect of opening size variation on the resulting block copolymer patterns.

### 3.4.1  Templating effect from confinement wall openings

We first demonstrate that the wall openings have a strong templating effect, and the orientation of the binary states can be successfully controlled by creating different number of openings in the confinement. **Figure 3.1** shows five possible types of $5L_0$ wide square confinements with one to four $1L_0$ wide openings on the sides of the confinement. We created arrays of square confinements with varying number of openings and measured the fraction of 0 (horizontal cylinders) states for

each type of confinement.
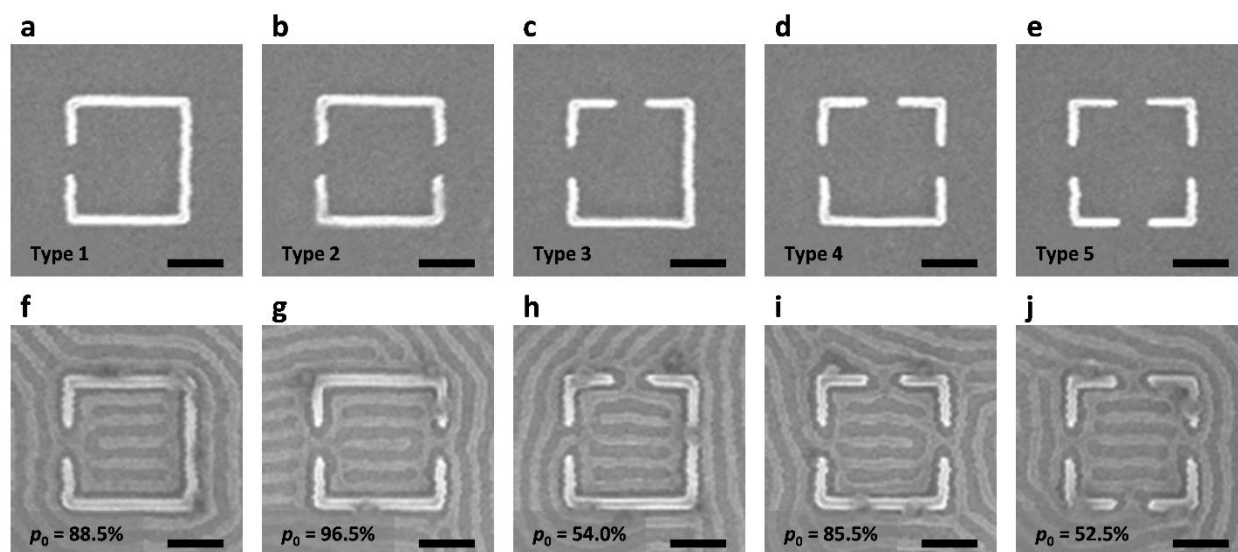


**Figure 3.1** Scanning electron microscope (SEM) images of square confinement with openings and the resulting block copolymer patterns. Ratio of 0 state (denoted by $p_0$) for each type was measured in a large array. Binary states were determined based on the number of horizontal and vertical openings. (a,f) Square confinement with one horizontal opening. Preferential horizontal alignment (0 state) was observed. (b,g) Square confinement with two horizontal openings on non-adjacent sides. Stronger preferential horizontal alignment was observed. (c,h) Square confinement with two openings on adjacent sides. Alignment in both directions was observed with equal probability. (d,i) Square confinement with three openings. Preferential horizontal alignment was observed. (e,j) Square confinement with four openings. Alignment in both directions was observed with equal probability. These results demonstrate the templating effect from openings. Scale bars, 200 nm.

When there were equal numbers of openings on the top and bottom sides and left and right sides as in type 3 and 5 configurations **(Figure 3.1c and Figure 3.1e)**, there were equal numbers of 1 and 0 states. However, when there were more openings on the left/right sides than top/bottom sides as in type 1, 2, and 4 configurations **(Figure 3.1a, Figure 3.1b, and Figure 3.1d)**, the 0 state was favored. The highest yield of a preferential alignment was achieved in the type 2 configuration where both openings were in the same direction. When there was one less opening in the same direction (type 1) or one more opening in the other direction (type 4), the yield was decreased by

~10%. As the number of openings around the square template was increased, there was less templating effect from the confinement and more defects were formed, but the binary state was still evident **(Figure 3.2)**. For arrays of square confinements with four openings, the normalized mean state-state correlation was $\rho = -0.00207$, suggesting no nearest-neighbor influence even with openings present between states.
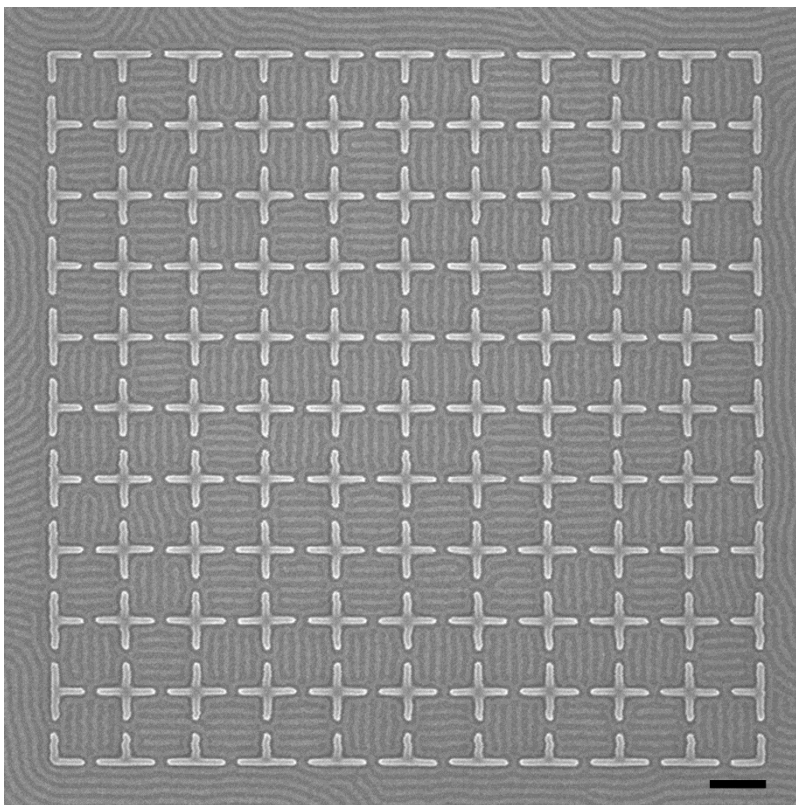


**Figure 3.2** SEM image of a 10 by 10 array of square confinements with four openings. Due to the defect tolerance of the system, the binary state inside each confinement can be uniquely determined even in the presence of defects. Scale bar, 200 nm.

SCFT was used to model the effect of a wall opening as a templating method on the final state of the system. We focused on the type 2 template that had the highest preference for one binary state. Although SCFT does not calculate dynamics directly, after several steps order emerges and the model is expected to resemble the evolution of the physical system. **Figure 3.3**

shows the evolution of the SCFT model starting from a random state. The high attraction at the walls initiated a uniform wetting layer **(Figure 3.3a)** from which an ordering front propagated away from the walls resulting in concentric squares surrounding the template. Two symmetric junctions were formed at the openings in the template walls **(Figure 3.3b)** connecting the inner and outer wetting layers of the template; however, the $1L_0$ wide openings became blocked isolating the inner polymer domains. In addition, the small outward curve of the wetting layer at the openings caused two horizontally aligned polymer domains to form which eventually bridge to create a horizontally disconnected polymer stripe, in remarkable agreement with experiment **(Figure 3.3c)**. The presence of this stripe biases the system into the 0 state **(Figure 3.3d)**.
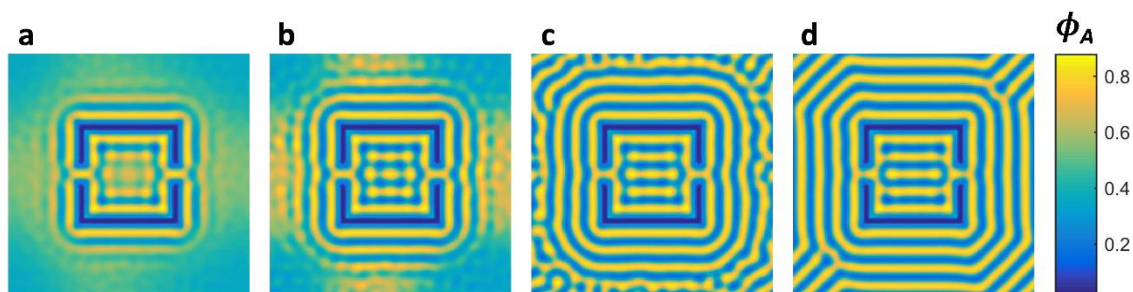


**Figure 3.3** SCFT simulations showing the evolution of the polymer self-assembled pattern. (a) Early stages of the simulation show a wetting layer is formed of the inner and outer surface of the walls while the polymer is still disordered in the region surrounding the template. A junction is formed at every opening in the template wall, connecting the inner and outer wetting layers. (b) An ordering front is propagating away from the walls creating a series of concentric squares. Polymer microdomains are nucleated adjacent to the bends in the inner wetting layer caused by the opening in the walls. (c) Bridging of the polymer domains creates a horizontal stripe between the two openings. (d) Final polymer self-assembled pattern reaches 0 state as shown experimentally.

**Figure 3.4** shows that the templating effect of the junctions was observed for various simulation conditions ($\chi N$ and strength of wall preferentiality) where the alignment of polymer domains was consistently parallel to the wall openings. On the other hand, the junction connecting

the inner and outer wetting layers was only stable for low $\chi N$ and strong wall attraction. This suggests the robustness of this templating approach to direct the system into a particular state.
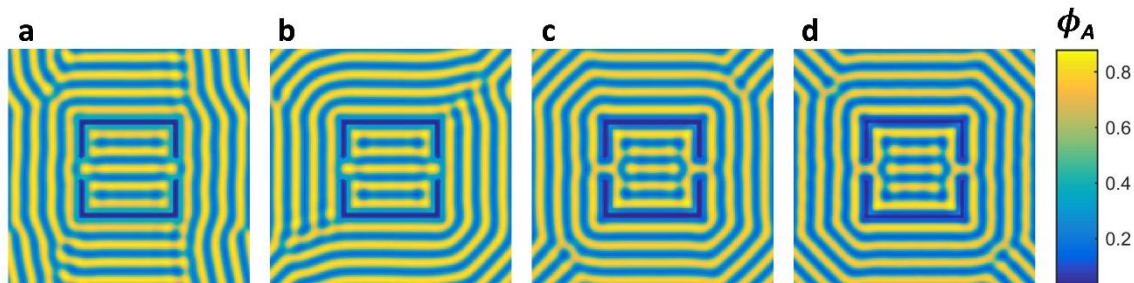


**Figure 3.4** SCFT simulations showing the density distribution of block *A* for different wetting conditions $w_-$. The higher the magnitude of $w_-$, the stronger the attraction to *A* at the walls. (a) $w_- = 1.5$, (b) $w_- = 2.5$, (c) $w_- = 4.5$, and (d) $w_- = 5.5$. The wetting at the walls does not affect the final state of the self-assembled polymer, indicating the effectiveness of having two opposite openings in directing the self-assembly of block copolymer. The experimental results are reproduced for $2.5 < w_- < 5.5$.

## 3.4.2 Scaling into a larger binary state array with individual state control

We have so far demonstrated binary state control by creating openings around the confinement (Chapter 3), changing the confinement geometry, or placing lithographic guiding patterns inside the confinement (Chapter 2). The three approaches for controlling the binary state orientations can be simultaneously used to fabricate desired binary state arrays.

We first show that the templating effect from confinement geometry is stronger than the templating effect from wall openings. We fabricated an array of vertical and horizontal rectangular confinements adjacent to square confinements with an opening placed in between (**Figure 3.5**). Aspect ratio of the rectangular confinement was 2:1. When the opening was placed on a longer side of the rectangular confinement as shown in **Figure 3.5a**, there were two competing templating effects. The rectangular confinement's vertical geometry induces vertical alignment, whereas the horizontal opening induces horizontal alignment. However, only 1 states were observed inside the

rectangular confinements, indicating that the templating effect from confinement geometry was more significant. Even when a local horizontal alignment was induced near the opening (red dashed box), overall vertical alignment was still observed inside the rectangular confinement. When the opening was placed on a shorter side of the rectangular confinement as shown in **Figure 3.5b**, 0 state was achieved since the two templating effects both induced horizontal alignment. Since lithographic guiding patterns in general have the strongest templating effect,[22-24] it was sufficient to just compare the relative templating strength from the confinement geometry and wall openings. Therefore, the three templating approaches can be used simultaneously to form a larger array without the templating effects influencing each other.
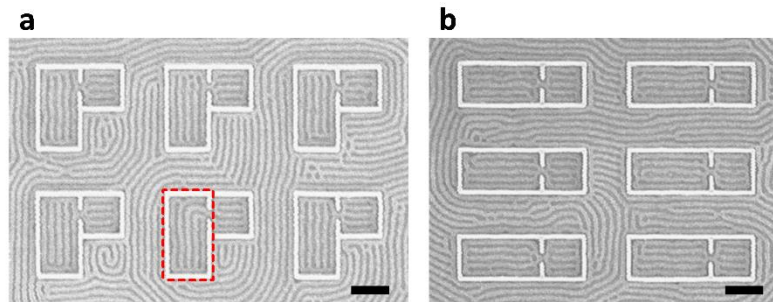


**Figure 3.5** SEM images of rectangular confinements with a horizontal opening. (a) Vertical rectangular confinements adjacent to square confinements with an opening placed in between. There were two completing templating effects from the confinement geometry and opening, but the templating effect from confinement geometry was more significant. (b) Horizontal rectangular confinements adjacent to square confinements with an opening in between. The two templating effects both induced horizontal alignment. Scale bars, 200 nm.

Using the three approaches for controlling the binary state orientation, we show how the orientation can be propagated within a larger template array. We showed in the previous chapter that neighboring binary states were uncorrelated with each other in square confinements separated by walls. However, by creating openings in the walls, specific orientations can be programmed by selecting cells from the five types of square confinement. **Figure 3.6** demonstrates two examples

of $4 \times 4$ binary patterns each composed of 16 independently controlled binary states. The target patterns are shown in **Figure 3.6a** and **Figure 3.6d**, the templates in **Figure 3.6b** and **Figure 3.6e**, and the successfully produced self-assembled pattern in **Figure 3.6c** and **Figure 3.6f**. For a given target pattern of binary states, in general more than one template can be chosen.
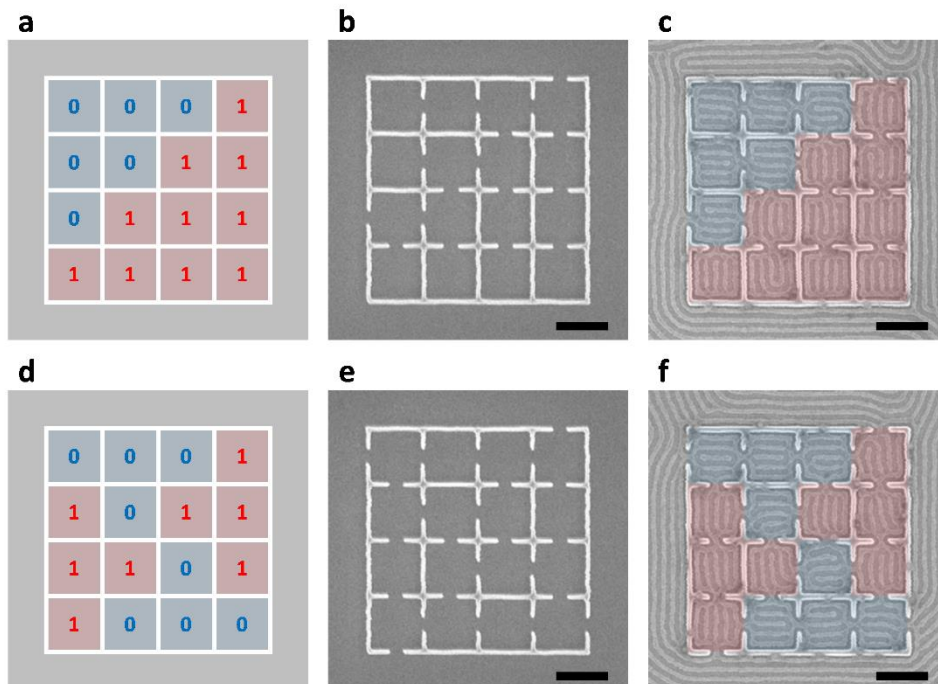


**Figure 3.6** Fabrication of binary state arrays. (a,d) Diagram of desired $4 \times 4$ binary state arrays. (b,e) Templates fabricated by electron-beam lithography. (c,f) Resulting block copolymer patterns matching the desired binary states. Red indicates 1 state and blue indicates 0 state. Scale bars, 200 nm.

**Figure 3.7** shows examples of different target patterns and corresponding template designs determined by trial and error. The target patterns shown in **Figure 3.7a** and **Figure 3.7c** are the same as the target patterns shown in **Figure 3.6a** and **Figure 3.6d**. New templates shown in **Figure 3.7b** and **Figure 3.7d** were designed by changing the location of certain openings and verifying the result. However, not all target patterns are obtainable by creating openings in the walls, since the opening locations determined by the four neighbors may conflict with the opening locations

required for the surrounded square. For example, a target pattern consisting of alternating states **(Figure 3.7k)** cannot be obtained from creating openings. By placing guiding patterns inside the confinement where such conflict may occur, we are able to design a template that will produce the target pattern **(Figure 3.7l)**. In general, a template for arbitrary binary state pattern can be designed by combining the three templating approaches.
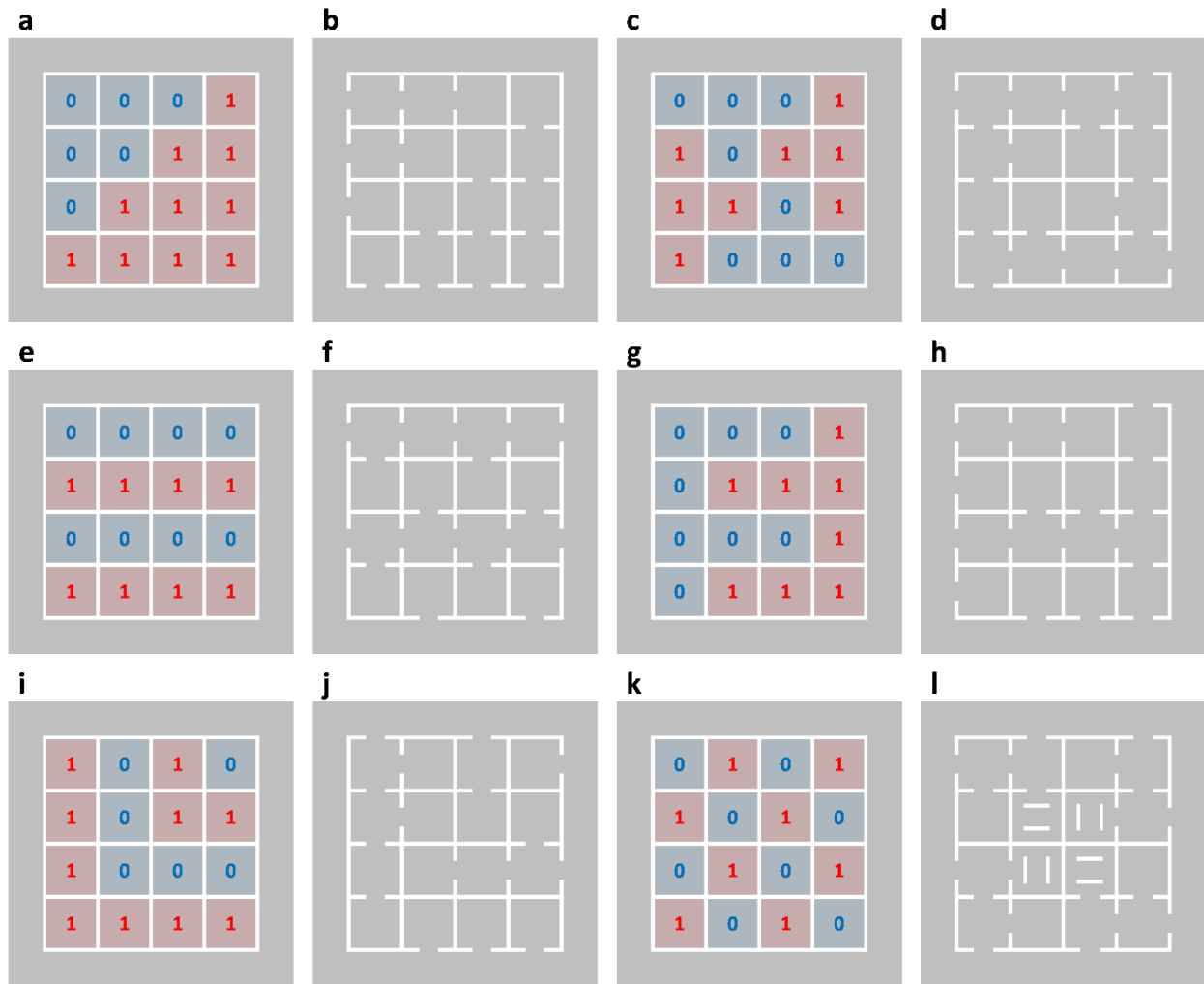


**Figure 3.7** Examples of target patterns and corresponding template designs. (a,c,e,g,i,k) Diagram of desired $4 \times 4$ binary state arrays. (b,d,f,h,j,l) Template designs that will produce the target patterns. Figure 3.7b and Figure 3.7d show alternative template designs for the same target patterns from Figure 3.6. Figure 3.7k shows a target pattern unobtainable from creating openings in the walls. Figure 3.7l shows a template design consisting of both openings guiding patterns.

### 3.4.3  Effect of opening size variation

Finally, we studied the effect of opening size variation on the nearest-neighbor interactions. As discussed in the previous section, no nearest-neighbor influence was observed with $1L_0$ wide openings present between the two states. An outer ring formed inside each confinement blocked any possible interactions between adjacent states. To prevent formation of the outer ring in the ladder-shaped structures, we decreased the templating effect from the confinement walls by increasing the opening size beyond $1L_0$.
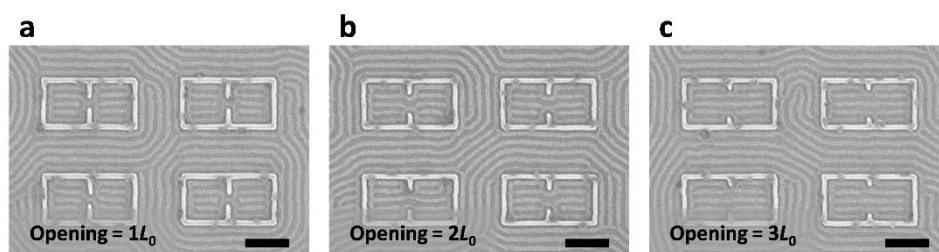


**Figure 3.8** SEM images of two horizontally adjacent square confinements with an opening placed in between. The opening sizes were (a) $1L_0$, (b) $2L_0$, and (c) $3L_0$. Templating effect from the opening was observed for all opening sizes, but the outer ring formation was prevented for larger opening sizes. Scale bars, 200 nm.

**Figure 3.8** shows the block copolymer patterns formed inside two horizontally adjacent square confinements with an opening placed in between. Width of the square confinement was $5L_0$ and the opening size was varied between $1L_0$ and $3L_0$. With an opening size of $1L_0$, both binary states were set to the 0 state **(Figure 3.8a)**. When the opening size was increased beyond $1L_0$, both binary states were still set to the 0 state without any defects **(Figure 3.8b and Figure 3.8c)**. However, the outer ring of the ladder-shaped structures was not fully formed when the opening size was greater than $2L_0$ due to the decreased templating effect from the confinement walls. As the opening size is increased, the confinement approximates a rectangular confinement where the

opening size is equal to the confinement width. By increasing the opening size, we were able to prevent the outer ring formation and allow parallel bars inside adjacent states to be directly connected to each other.

We now show that the block copolymer morphology at the opening can be controlled based on the two adjacent binary states separated by the opening. We fabricated a square confinement with multiple openings where width of the square confinement was $6L_0$ and the opening size was varied between $2L_0$ and $4L_0$. The four adjacent states were set to the 0 state using horizontal rectangular confinements. Wider openings ($\geq 2L_0$) allowed parallel bars inside one state to pass through the opening and influence its adjacent state. As shown in **Figure 3.9**, the two adjacent states can be either separated or connected depending on the morphology at the opening. When two 0 states were vertically adjacent (red dashed box), the two states were separated by a horizontal block copolymer structure at the opening. When two 0 states were horizontally adjacent (blue dashed box), the two states were connected to each other through the opening.
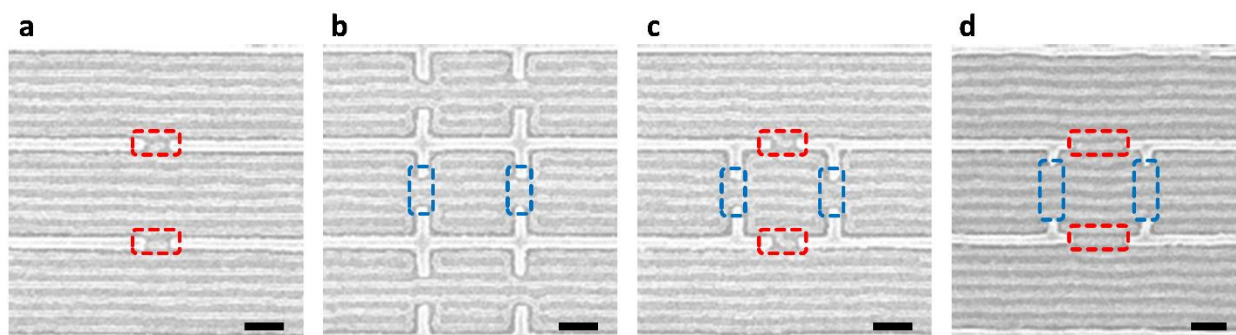


**Figure 3.9** SEM images of block copolymer morphology at the opening. Opening sizes were (a-c) $2L_0$ and (d) $4L_0$. A horizontal line structure blocked the opening when two 0 states were vertically adjacent (red dashed box). Line structures were connected through the opening when two 0 states were horizontally adjacent (blue dashed box). Scale bars, 100 nm.

We wanted to further study the opening morphology control, but placing wider openings around the confinement resulted in decreased overall templating effect from the confinement walls. As a result, defect tolerance of the two-state system was lost and defective structures such as diagonally aligned ladder-shaped structures and disordered patterns were formed inside more than 50% of the confinements. If the nearest-neighbor interactions through the opening can be better understood and defect tolerance of the system can be maintained for larger opening sizes, it may even be possible to build a four-input majority gate as proposed in **Figure 3.10**.
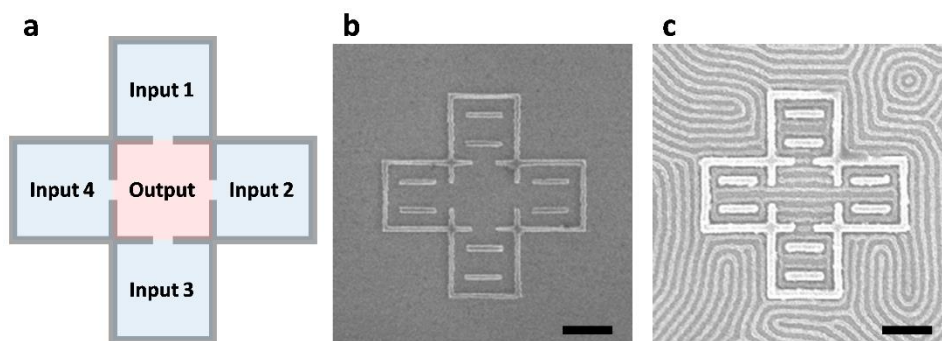


**Figure 3.10** Proposed design for a four-input majority gate. (a) Output state is defined inside a center square confinement with four openings. Four input states are defined inside the adjacent confinements using lithographic guiding patterns. (b) SEM image of a template where the four input states were set to the 0 state. (c) SEM image of the resulting block copolymer pattern. Top and bottom openings were blocked due to input 1 and input 3 being set to the 0 state. Therefore, the two remaining horizontal openings determined the output state. The output state was set to the 0 state even though the center confinement had four openings. Other input combinations were not investigated due to formation of defective structures. Scale bars, 200 nm.

## 3.5   Conclusion

The self-assembly of block copolymers inside square confinements with different number of wall openings is studied. Both experimental results and SCFT simulations showed templating effect due to a junction formed at the opening. We demonstrated larger binary state arrays where individual binary states were independently controlled. Nearest-neighbor interactions for larger

opening sizes were observed, although defect tolerance of the system was lost. With further research on the nearest-neighbor interactions through the opening, self-assembly based computation using the two-state system might be possible. In the following three chapters, we study a simpler two-state system defined on two-dimensional arrays of posts, where the binary states can directly interact with its nearest neighboring states.

# References

1. S.-M. Park, O.-H. Park, J. Y. Cheng, C. T. Rettner, H.-C. Kim, *Nanotechnology* **2008**, 19, 45.
2. Y.-C Tseng, S. B. Darling, *Polymers* **2010**, 2, 4.
3. J. G. Son, M. Son, K.-J. Moon, B. H. Lee, J.-M. Myoung, M. S. Strano, M.-H. Ham, C. A. Ross, *Adv. Mater.* **2013**, 25, 34.
4. H. Tsai, J. W. Pitera, H. Miyazoe, S. Bangsaruntip, S. U. Engelmann, C.-C. Liu, J. Y. Cheng, J. J. Bucchignano, D. P. Klaus, E. A. Joseph, D. P. Sanders, M. E. Colburn, M. A. Guillorn, *ACS Nano* **2014**, 8, 5.
5. M. J. Maher, C. T. Rettner, C. M. Bates, G. Blachut, M. C. Carlson, W. J. Durand, C. J. Ellison, D. P. Sanders, J. Y. Cheng, C. G. Wilson, *ACS Appl. Mater. Interfaces* **2015**, 7, 5.
6. L. Wan, R. Ruiz, H. Gao, K. C. Patel, T. R. Albrecht, *ACS Nano* **2015**, 9, 7.
7. A. Nourbakhsh, A. Zubair, R. N. Sajjad, A. Tavakkoli K. G., W. Chen, S. Fang, X. Ling, J. Kong, M. S. Dresselhaus, E. Kaxiras, K. K. Berggren, D. Antoniadis, T. Palacios, *Nano Lett.* **2016**, 16, 12.
8. H. S. Suh, D. H. Kim, P. Moni, S. Xiong, L. E. Ocola, N. J. Zaluzec, K. K. Gleason, P. F. Nealey, *Nat. Nanotechnol.* **2017**, 12.
9. Y.-R. Hong, K. Asakawa, D. H. Adamson, P. M. Chaikin, R. A. Register, *Opt. Lett.* **2007**, 32, 21.
10. S.-J. Jeong, H.-S. Moon, J. Shin, B. H. Kim, D. O. Shin, J. Y. Kim, Y.-H. Lee, J. U. Kim, S. O. Kim, *Nano. Lett.* **2010**, 10, 9.
11. D. O. Shin, J. H. Mun, G.-T. Hwang, J. M. Yoon, J. Y. Kim, J. M. Yun, Y.-B. Yang, Y. Oh, J. Y. Lee, J. Shin, K. J. Lee, S. Park, J. U. Kim, S. O. Kim, *ACS Nano* **2013**, 7, 10.
12. J. H. Mun, S. K. Cha, H. Kim, H.-S. Moon, J. Y. Kim, H. M. Jin, Y. J. Choi, J. E. Baek, H. Shin, S. O. Kim, *Small* **2014**, 10, 18.
13. P. W. Majewski, A. Rahman, C. T. Black, K. G. Yager, *Nat. Commun.* **2015**, 6, 7448.
14. J. Chai, D. Wang, X. Fan, J. M. Buriak, *Nat. Nanotechnol.* **2007**, 2.
15. S.-J. Jeong, J. E. Kim, H.-S. Moon, B. H. Kim, S. M. Kim, J. B. Kim, S. O. Kim, *Nano Lett.* **2009**, 9, 6.
16. Y. S. Jung, J. H. Lee, J. Y. Lee, C. A. Ross, *Nano Lett.* **2010**, 10, 9.
17. R. A. Farrell, N. T. Kinahan, S. Hansel, K. O. Stuen, N. Petkov, M. T. Shaw, L. E. West, V. Djara, R. J. Dunne, O. G. Varona, P. G. Gleeson, S.-J. Jung, H.-Y. Kim, M. M. Koleśnik, T. Lutz, C. P. Murray, J. D. Holmes, P. F. Nealey, G. S. Duesberg, V. Krstić, M. A. Morris, *Nanoscale* **2012**, 4.
18. G. Tzeremes, K. Ø. Rasmussen, T. Lookman, A. Saxena, *Phys. Rev. E* **2002**, 65, 4.
19. K. Rasmussen, G. Kalosakas, *J. Polym. Sci. B* **2002**, 40, 16.
20. S. W. Sides, G. H. Fredrickson, *Polymer* **2003**, 44, 19.
21. S. W. Sides, G. H. Fredrickson, *J. Chem. Phys.* **2004**, 121, 10.
22. Y. S. Jung, C. A. Ross, *Nano Lett.* **2007**, 7, 7.
23. R. Ruiz, N. Ruiz, Y. Zhang, R. L. Sandstrom, C. T. Black, *Adv. Mater.* **2007**, 19, 16.
24. Y. S. Jung, C. A. Ross, *Adv. Mater.* **2009**, 21, 24.

# Chapter 4

# Ising lattice design in two-dimensional post array

In this chapter, we demonstrate an Ising lattice setup for directed self-assembly of block copolymers defined on two-dimensional arrays of posts. Using a square lattice post template with pitch set to the equilibrium block copolymer periodicity, we map a square lattice Ising model with nearest-neighbor interactions. We study properties of the Ising lattice such as distribution of the states and nearest-neighbor correlation. We compare the two Ising model parameters $J$ and $h$ required for simulated annealing, and discuss the effect of placing incommensurate double posts inside the post template.

## 4.1   Introduction

Block copolymers are self-assembling polymer materials composed of two covalently bonded macromolecular blocks. Due to their chemical functionalities, tunability, and low cost, block copolymers have been used in directed self-assembly (DSA) for various applications in nanofabrication.[1-8] To calculate the equilibrium morphology inside a given DSA template, self-consistent field theory (SCFT) methods have been previously developed. In SCFT simulations, a set of nonlinear and nonlocal equations associated with the free energy functional are numerically

solved to find the equilibrium morphology with minimum free energy.[9-13] Equilibrium block copolymer morphology has been calculated and experimentally reproduced for various template geometries with high accuracy.[14-18] SCFT simulations have also been used to solve the inverse design problem of finding the optimal template design required to achieve a desired block copolymer morphology.[19-20]

Here, we consider the possibility of using block copolymers as a material for computation instead of lithography by applying the Ising model to explain DSA of block copolymers. During the self-assembly process, block copolymers arrange themselves into a minimum free energy configuration given some boundary conditions defined by the guiding templates. Intrinsically, block copolymers are solving an energy optimization problem through self-assembly. In SCFT simulations, the energy optimization problem is numerically solved from a series of modified diffusion equations. By using a simpler Ising Hamiltonian to model the energy optimization problem, simulation time can be greatly reduced. Moreover, it has been previously shown that ground states of Ising lattices can encode various computation problems.[21,22] If an Ising lattice setup for DSA of block copolymers can be developed, block copolymers may be used to perform Ising-model-based computation.

This chapter describes an Ising lattice setup for DSA of block copolymers defined on two-dimensional arrays of posts. We demonstrate that block copolymer self-assembly maps rigorously onto the Ising model. We measure relative magnitude of the two Ising model parameters $J$ and $h$, and discuss how to interpret the effect of placing incommensurate double posts inside the post arrays to define specific boundary conditions.

## 4.2 Experimental methods

In this section, we describe the experimental methods for post lattice template fabrication, block copolymer solvent annealing, reactive-ion etching, and metrology.

### 4.2.1 Template fabrication

The post lattice templates were fabricated using electron-beam lithography with a hydrogen silsesquioxane (HSQ) resist. A silicon substrate was first treated with $O_2$/He plasma (50 W, 10 s) to remove possible organic residues and make the substrate surface hydrophilic. The substrate was spin coated with 35-nm-thick HSQ film (XR-1541 2% solids, Dow Corning). The thickness was determined by ellipsometry. A Raith 150 electron-beam lithography system operated at 30 kV acceleration voltage was used to expose two dimensional arrays of posts with various diameters and pitches. After exposure, the samples were developed in a 24°C high contrast salty developer (1% NaOH and 4% NaCl in de-ionized water) for 4 min and rinsed in de-ionized water for 3 min.[23] A thin layer of water was maintained on the hydrophilic surface of the substrate during development, which prevented capillary force-induced collapse of the post structures. The final structure was rinsed with a low surface tension liquid, isopropanol prior to blow drying with $N_2$ gas to prevent pattern collapse.[24-26] The samples were further treated with $O_2$/He plasma (50 W, 10 s) to convert the HSQ structures into silicon oxide. Template dimensions were inspected by scanning electron microscope (SEM) imaging.

### 4.2.2 Block copolymer self-assembly

To make the templates attractive to the minority polydimethylsiloxane (PDMS) block, the templates were chemically functionalized with a hydroxyl-terminated PDMS brush (0.8 kg/mol,

Polymer Source Inc.) by spin coating 2% brush solution in toluene and annealing the samples in a vacuum oven at 170°C for 14 h. After annealing, the samples were first rinsed with toluene and then submerged in a toluene bath at room temperature for 10 min to remove excess PDMS brush. The resulting thickness of the PDMS brush bonded to the substrate was 1 nm. Next, 2% poly(styrene-*block*-dimethylsiloxane) (PS-*b*-PDMS, $M_w$ = 53 kg/mol, $f_{PDMS}$ = 30%, PDI = 1.05, synthesized in Avgeropoulos group, University of Ioannina)[27-30] solution in propylene glycol monomethyl ether acetate (PGMEA) was spin coated onto the templated substrate. The resulting film thickness was 33 nm. The samples were solvent annealed using a 5:1 mixture of toluene and heptane at room temperature for 3 h. We placed the samples on a glass slide stack (0.8 cm in height) inside a crystallization dish (1.5 cm in height, 5 cm in diameter) and added 1.5 ml of the 5:1 toluene and heptane mixture. The chamber was covered with a petri dish (10 cm in diameter) to allow slow evaporation of the solvent mixture.

### 4.2.3  Reactive-ion etching

Reactive-ion etching of the annealed block copolymer film was performed in two steps. First, the top PDMS wetting layer was removed using a 5 s $CF_4$ plasma treatment with a power of 50 W and pressure of 15 mTorr. Next, the PS matrix was removed using a 22 s $O_2$ plasma treatment with a power of 90 W and pressure of 6 mTorr. This step also oxidized the PDMS cylinders.

### 4.2.4  Metrology

Metrology was performed by examining the HSQ templates and the reactive-ion etched block copolymer films using a SEM. After developing the HSQ, several post structures collapsed depending on the HSQ quality as well as the post height and diameter. To avoid further processing

of defective templates, the HSQ templates were examined before chemically functionalizing the surface. The etched block copolymer films were sputter coated with a thin layer of gold/palladium to reduce charging and enhance image contrast. Top down SEM images were obtained using a Raith 150 SEM operated at 10 kV acceleration voltage and 6 mm working distance, and Zeiss Sigma SEM operated at 3 kV acceleration voltage and 4 mm working distance.

## 4.3  Results and discussion

In this section, we present an Ising lattice setup using two-dimensional arrays of posts for implementing the Ising model in block copolymer systems.

### 4.3.1  Ising model and block copolymer self-assembly

The Ising model is a mathematical model of interacting magnetic spins originally invented to explain ferromagnetism and phase transitions in statistical mechanics.[31] However, universal computation as well as combinatorial optimization problems can be encoded into the ground states of Ising lattices by designing specific Hamiltonians that embed such operations.[21,22] The Ising states can take values +1 for spin up, −1 for spin down, and can interact only with their nearest-neighboring states (**Figure 4.1**). The Hamiltonian of a state configuration $\sigma$ can be expressed as

$$H(\sigma) = -\sum_{\langle i\,j \rangle} J_{ij}\sigma_i\sigma_j - \sum_{j} h_j\sigma_j$$

where the first sum corresponds to energy due to each nearest-neighbor interaction and the second sum corresponds to energy due to individual states. An analytic solution for the general two-

dimensional square lattice Ising model has yet to be found, but one can use the simulated annealing algorithm to find an approximate minimum Hamiltonian configuration.
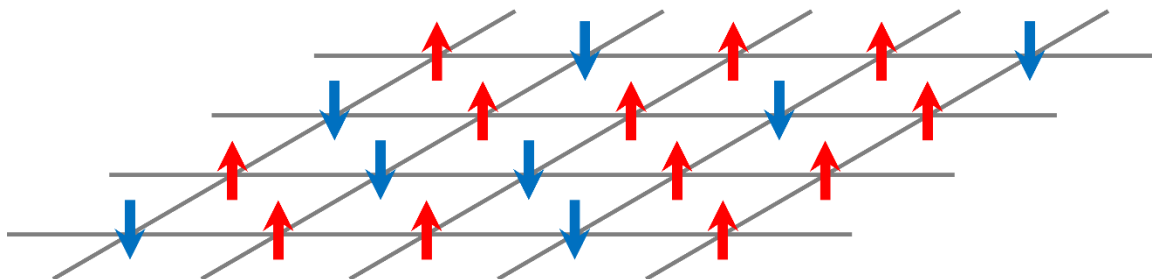


**Figure 4.1** Diagram of a square Ising lattice. The model consists of spins that can be in either up state ($\sigma_j = +1$) or down state ($\sigma_j = -1$). Each state can interact with its four nearest-neighboring states. Behavior of the system is governed by the Ising model parameters $J_{ij}$ and $h_j$.
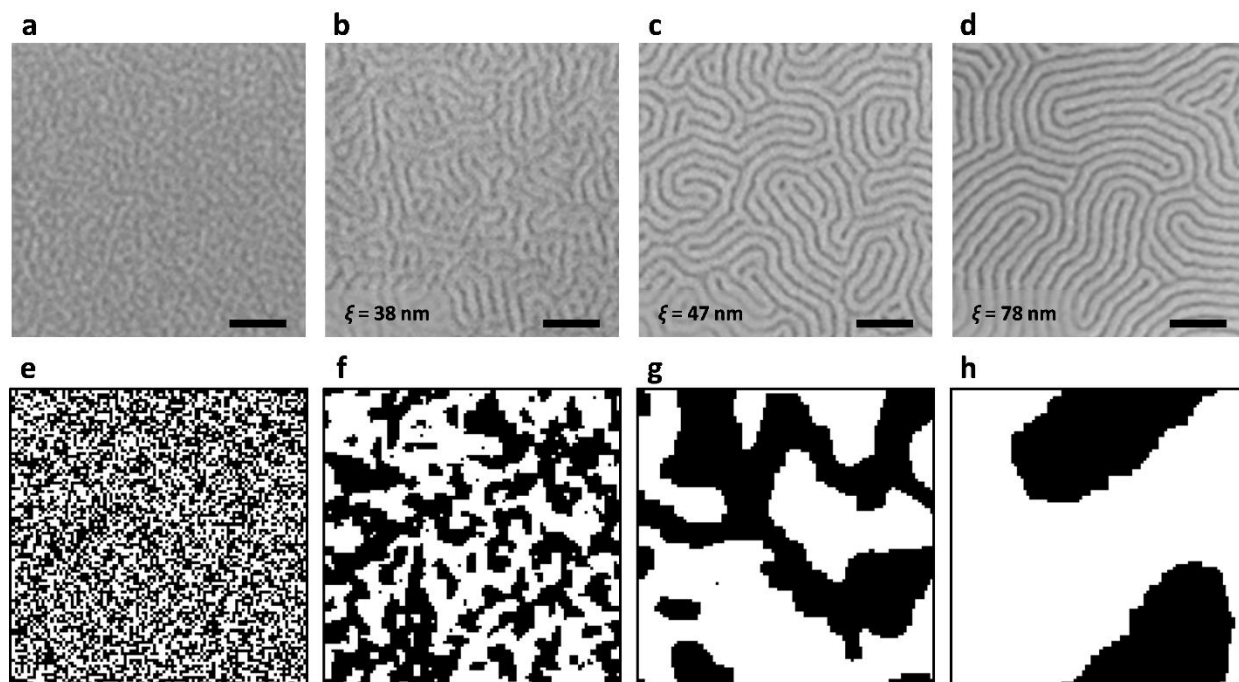


**Figure 4.2** Comparison of block copolymer morphologies and Ising states. (a-d) Morphology and correlation length $\xi$ of poly(styrene-*block*-methylmethacrylate) (PS-*b*-PMMA) annealed at 190°C for (a) 10 s, (b) 1 min, (c) 10 min, and (d) 32 h. [32] Scale bars, 100 nm. Redrawn from reference 32. (e-h) Phase transition behavior of a $100 \times 100$ array of ferromagnetic Ising states after (e) 0 (random initial array) (f) 1,000 (g) 10,000, and (h) 100,000 simulated annealing steps. Simulation details are discussed in the next chapter.

We were particularly interested in applying the square lattice Ising model to DSA of block copolymers. In addition to its simplicity and applicability in encoding computation, the Ising model exhibits a similar phase transition behavior shown in block copolymer self-assembly. **Figure 4.2** demonstrates phase transition behavior in block copolymer systems **(Figure 4.2a-4.2d)** and Ising states with ferromagnetic interactions ($J_{ij} > 0$) and $h_j = 0$ **(Figure 4.2e-4.2h)**. In block copolymer self-assembly, it has been shown that correlation length and grain size are increased as annealing time is increased.[32] In a simulated annealing of ferromagnetic Ising states, we observed similar phase transition behaviors as the number of simulation steps was increased. Therefore, it may be possible use the Ising model to explain self-assembly of block copolymers, at least when templates are used.

## 4.3.2 Designing an Ising lattice in block copolymer systems

To apply the Ising model to DSA of block copolymers, an Ising lattice setup must be constructed first such that desired Ising Hamiltonians can be measured. As shown in **Figure 4.3**, block copolymer self-assembly maps onto an Ising model of a square lattice of binary states with nearest-neighbor interactions. On a silicon substrate chemically functionalized with a hydroxyl-terminated PDMS brush, 53 kg/mol cylindrical morphology PS-*b*-PDMS (SD53) naturally formed complex fingerprint patterns **(Figure 4.3a)**. To define the Ising lattice, we fabricated a square lattice post template using electron-beam lithography with an HSQ resist. SD53 block copolymer was spin coated onto the functionalized template to a thickness of 33 nm and solvent annealed using a mixture of toluene and heptane. Previous work have shown that that two-dimensional arrays of posts can facilitate well-ordered block copolymer patterns with low defect rates.[6,14,18,20] For this work, the horizontal and vertical pitch between the posts was set to the equilibrium block copolymer periodicity, $L_0 = 39$ nm **(Figure 4.3b)**. Since both horizontal and vertical directions

were commensurate with $L_0$, the block copolymer grains were degenerately aligned parallel to the x-axis or the y-axis **(Figure 4.3c)**.
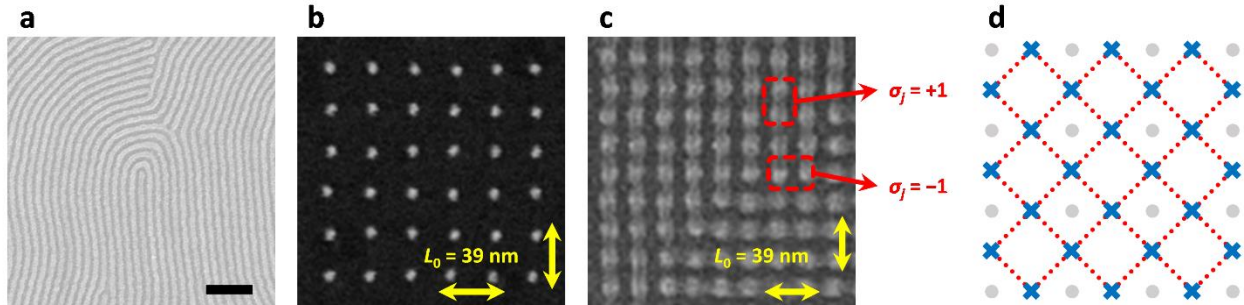


**Figure 4.3** Ising lattice design in DSA of block copolymers. (a) SEM image of untemplated PDMS cylinders with $L_0$ = 39 nm. Scale bar, 200 nm. (b) Square lattice post template fabricated by electron-beam lithography. The horizontal and vertical pitch was set equal to $L_0$. (c) Block copolymer pattern formed on the post lattice template showing two degenerate alignment orientations. When two adjacent posts were connected by a PDMS cylinder, the Ising state was defined as +1. Otherwise, the Ising state was defined as −1. (d) Diagram showing the designed Ising lattice in the post lattice template. The axes are rotated by 45°. The posts (grey dot), Ising states (blue cross), and nearest-neighbor interactions (red dashed line) are indicated.

Here, we defined a binary state, +1 or −1, between each adjacent pair of posts. As indicated by the red dashed boxes in **Figure 4.3c**, we assigned +1 ("spin up") to a state when two adjacent posts were connected by a block copolymer structure, and −1 ("spin down") otherwise. Such assignment ensured a one-to-one mapping between a block copolymer pattern and the corresponding binary state array. Establishing a one-to-one mapping was especially important for defining the Ising lattice. For example, consider an alternative Ising lattice where the states are defined at the posts based on alignment orientation of the block copolymer structure formed at each post. In such system, a one-to-one mapping cannot be achieved due to formation of bends which do not show a particular horizontal or vertical alignment orientation. Even if the four types

of bends are interpreted as new states, it is still very challenging to fully understand the interactions between all possible state pair combinations.

**Figure 4.3d** shows the corresponding Ising lattice setup, where the square lattice posts (grey dot), Ising states defined between each adjacent pair of posts (blue cross), and nearest-neighbor interactions between the Ising states (red dashed line) are indicated. Using this Ising lattice setup, any block copolymer pattern can be uniquely mapped to a binary state array, and vice versa. An example of a block copolymer pattern and its corresponding binary state array is shown in **Figure 4.4**. In this setup, we assumed that only the nearest-neighbor interactions are allowed, since the nearest neighbors in this setup directly interact via block copolymer chains. The Ising Hamiltonian is given by

$$H(\sigma) = -J \sum_{\langle i\, j \rangle} \sigma_i \sigma_j - h \sum_j \sigma_j$$

where $J_{ij}$ and $h_j$ from the original Ising Hamiltonian were assumed to be independent of lattice location for simplicity.
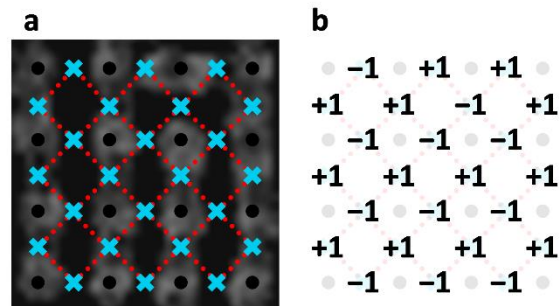


**Figure 4.4** Sample block copolymer pattern and its associated binary state array. The posts (black dot), Ising states (blue cross), and nearest-neighbor interactions (red dashed line) are indicated. (a) SEM image of a block copolymer pattern formed on the post lattice template. (b) Associated binary state array. Antiferromagnetic interactions make intuitive sense since high density of one domain (PDMS or PS) is disfavored.

Next, to show that the two Ising states were truly degenerate, we measured the distribution of +1 and −1 states. As shown in **Figure 4.5**, we created large arrays of periodic posts and measured the distribution of +1 and −1 states in the resulting block copolymer pattern. We expected the two Ising states to form with essentially equal probability since parallel arrays of PDMS cylinders correspond to alternating Ising states. The distribution of +1 state was 47.3% and the distribution of −1 state was 52.7%, agreeing with our prediction.
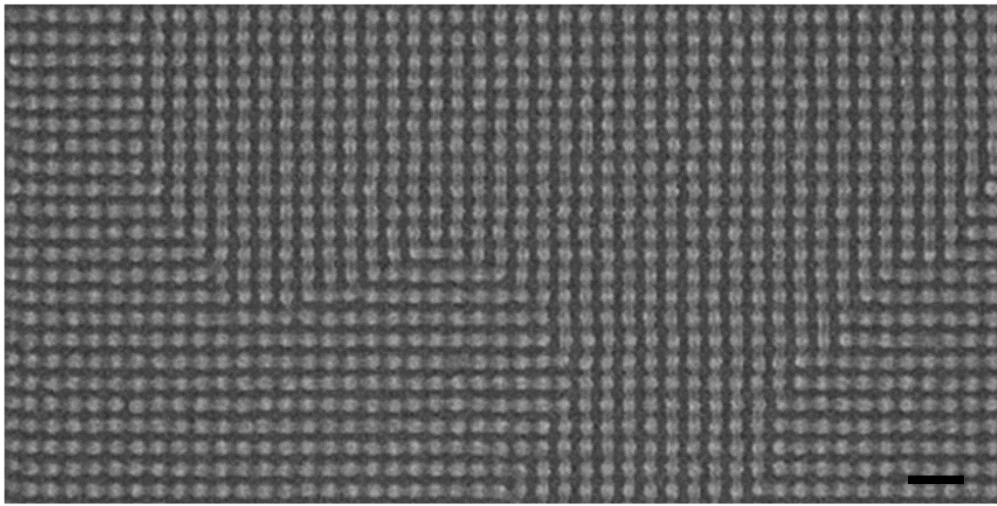


**Figure 4.5** SEM image of a block copolymer pattern formed on the post lattice template. The pattern consisted of several regions of parallel PDMS cylinders which were degenerately aligned either horizontally or vertically. +1 and −1 states were formed with essentially equal probability. Scale bar, 100 nm.

In addition, we calculated the normalized mean state-state correlation to determine the sign of $J$. The measured correlation is

$$\rho = \frac{\sum_{i\sim j} \sigma_i \sigma_j}{\sum_{i\sim j} |\sigma_i \sigma_j|} = -0.822$$

where the sum was taken over every pair of adjacent states. The negative value of $\rho$ close to $-1$ indicates that the neighboring binary states experienced antiferromagnetic interactions ($J < 0$ in the Ising Hamiltonian). For a $+1$ state, it is energetically favorable for its four nearest neighbors to have a $-1$ state, since adjacent $+1$ states indicate formation of bends or T-junctions which are energetically costly.[33,34] For a $-1$ state, it is energetically favorable for its four nearest neighbors to have a $+1$ state, since adjacent $-1$ states indicate formation of a region with only PS domains. On the other hand, alternating $+1$ and $-1$ states result in parallel PDMS cylinders which are energetically favorable. Hence, the antiferromagnetic interaction agrees with our previous understanding of block copolymer self-assembly.

### 4.3.3  Determining relative magnitude of *J* and *h*

To perform simulated annealing and calculate the minimum Hamiltonian configuration, we must know the relative magnitude of *J* and *h* in the Ising Hamiltonian. We compared the two parameters by measuring the distribution of the center state given the four nearest-neighbor configurations. The four nearest neighbors of a given state can have four to zero $+1$ states and zero to four $-1$ states. For example, **Figure 4.6** shows the case where the four nearest neighbors have three $+1$ states and one $-1$ state. The center state is $-1$ in **Figure 4.6a** and $+1$ in **Figure 4.6c**. For both configurations, we can calculate the Ising Hamiltonian by converting the block copolymer pattern into a corresponding binary state array as shown in **Figure 4.6b** and **Figure 4.6d**. When the center state is $-1$, the resulting Hamiltonian is $H_1(\sigma) = 2J - h$. When the center state is $+1$, the resulting Hamiltonian is $H_2(\sigma) = -2J - 3h$. In such nearest-neighbor configuration, 98.7% of the center state was $-1$ as shown in **Figure 4.6e**. From this distribution data, we can conclude that the Hamiltonian of the configuration where the center state is $-1$ is less than the Hamiltonian of the configuration

where the center state is +1. The inequality comparing the two Hamiltonians, $H_1(\sigma) < H_2(\sigma)$, reduces to $h < -2J$.
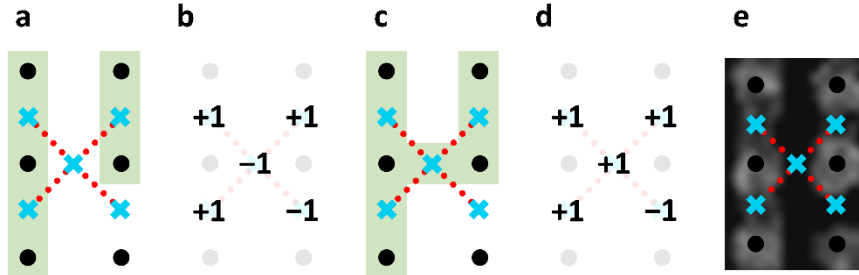


**Figure 4.6** Sample configurations where the four nearest neighbors have three +1 states and one −1 state. The posts (black dot), four nearest-neighboring states and center state (blue cross), and nearest-neighbor interactions (red dashed line) are indicated. (a) Diagram of the configuration where the center state is −1. (b) Associated binary state array where the center state is −1. The Ising Hamiltonian is $H_1(\sigma) = 2J - h$. (c) Diagram of the configuration where the center state is +1. (d) Associated binary state array where the center state is +1. The Ising Hamiltonian is $H_2(\sigma) = -2J - 3h$. (e) SEM image of the center state observed with higher probability. A majority of the center state was −1 given this specific four nearest-neighbor configuration.

We can obtain a similar inequality involving $J$ and $h$ for different configurations. The five possible configurations of the four nearest neighbors are shown in **Figure 4.7**. **Figures 4.7a-4.7e** show diagrams representing the block copolymer connections formed in the center state and four nearest-neighboring states. Since we were interested in calculating the Hamiltonian for these five states, other block copolymer connections (formed between the top two posts or the bottom two posts) were omitted for simplicity. **Figures 4.7f-4.7j** show the associated binary state arrays that were used for calculating the Hamiltonian of a given configuration. In **Figures 4.7k-4.7o**, the center state with higher observed probability is shown for each nearest-neighbor configuration. We can obtain $h < -4J$ from **Figure 4.7k**, $h < -2J$ from **Figure 4.7l**, $h < 0$ from **Figure 4.7m**, $h > 2J$ from **Figure 4.7n**, and $h > 4J$ from **Figure 4.7o**. Combined with the antiferromagnetic

interactions ($J < 0$), these inequalities simplify to $2J < h < 0$. For simplicity, we set $J = h < 0$ for

our simulated annealing algorithm which is discussed in the next chapter.
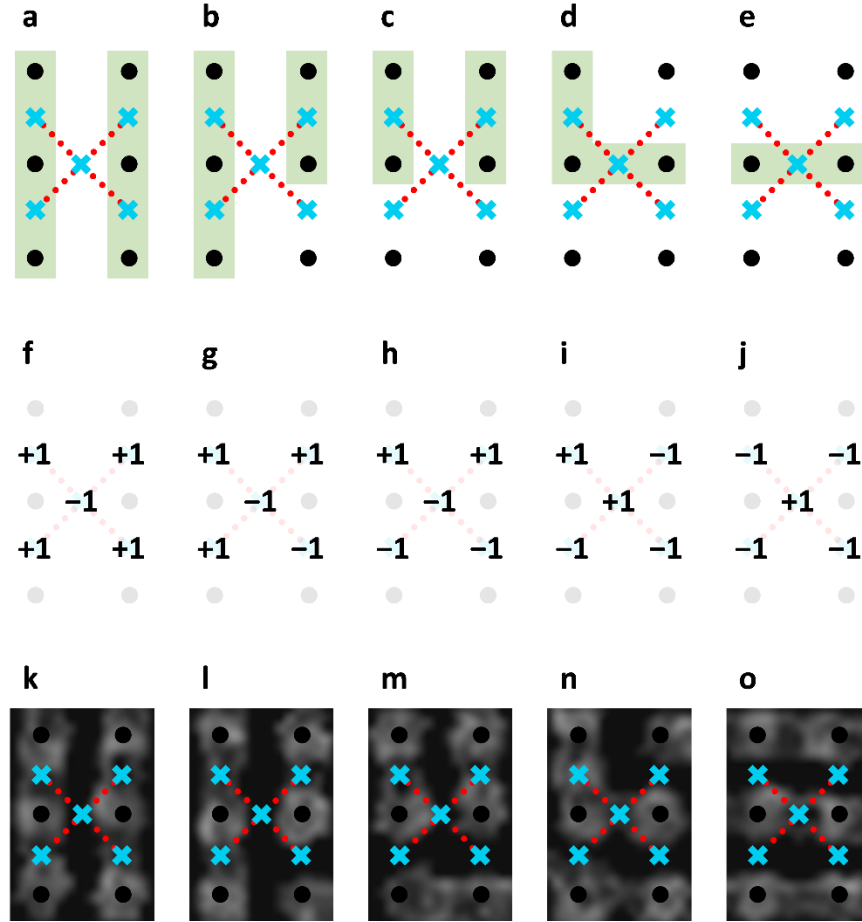


**Figure 4.7** The minimum Hamiltonian configuration when the four nearest neighbors have (a,f,k) four, (b,g,l) three, (c,h,m) two, (d,i,n) one, and (e,j,o) zero +1 states. The posts (black dot), four nearest-neighboring states and center state (blue cross), and nearest-neighbor interactions (red dashed line) are indicated. (a-e) Diagrams of the configurations representing the block copolymer connections. (f-j) Associated binary state arrays. (k-o) SEM images of the minimum Hamiltonian configuration observed with higher probability. The observation probabilities of the center state set as shown in the SEM images were (k) 100%, (l) 98.7%, (m) 98.5% (n) 96.7%, and (o) 99.8%.

Next, we investigated the effect of placing incommensurate double posts, which can be

used to define input states and boundary conditions. A double post was defined as a set of two

posts separated by a distance less than $L_0$, which forces formation of a PDMS cylinder between

the posts. As shown in **Figure 4.8a** and **Figure 4.8b**, placing an additional post inside a square lattice post array alters the Ising lattice as well as the associated *J* and *h*. First, the binary states defined between the double posts (black cross) are fixed to +1 by incommensurability. Furthermore, analysis of previous experimental results[18] showed that in the minimum Hamiltonian configuration, the six adjacent states (red cross) were always fixed as shown in **Figure 4.8c**. The corresponding block copolymer pattern is a straight line around the double posts meaning that bends, terminations, and T-junctions cannot be formed at the double posts **(Figure 4.8d)**. Although we were not able to measure the exact values of *J* and *h* in the altered Ising lattice, we were able to determine the effect of placing double posts on the final block copolymer morphology.
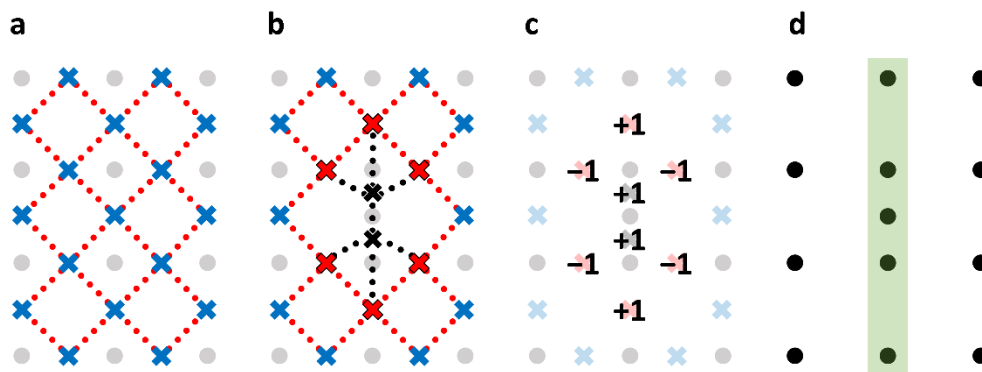


**Figure 4.8** Diagrams showing the Ising lattice in the post lattice template with double posts. The posts (grey/black dot), Ising states (cross), and nearest-neighbor interactions (dashed line) are indicated. (a) The original Ising lattice before placing a double post. (b) Placing a double post at the center directly affects the two states (black cross), the seven nearest-neighbor interactions (black dashed line), and the six adjacent states (red cross). (b) The eight affected states were fixed due to strong templating effect from the double posts. (c) The corresponding block copolymer pattern is a straight line formed around the double posts.

## 4.4　Conclusion

We demonstrated an Ising lattice setup for DSA of block copolymers using a square lattice post template. The Ising states were defined based on block copolymer connections formed between each pair of adjacent posts. By measuring the distribution and nearest-neighbor correlation of the Ising states, we measured the sign of $J$ as well as the relative magnitude of $J$ and $h$. Based on these results, simulated annealing algorithm can be applied to calculate the minimum Hamiltonian configuration. We also discussed the effect of placing incommensurate double posts. By using double posts, input states can be initialized to desired values, and specific boundary conditions can be designed. In the next chapter, we apply the simulated annealing algorithm to calculate the minimum Hamiltonian configuration given specific boundary conditions. We validate our Ising lattice setup by comparing the Ising model simulation results with previously reported experimental and SCFT simulation results.

# References

1.  M. P. Stoykovich, H. Kang, K. C. Daoulas, G. Liu, C.-C. Liu, J. J. de Pablo, M. Müller, P. F. Nealey, *ACS Nano* **2007**, 1, 3.
2.  A. M. Welander, H. Kang, K. O. Stuen, H. H. Solak, M. Müller, J. J. de Pablo, P. F. Nealey, *Macromolecules* **2008**, 41, 8.
3.  J. Y. Cheng, C. T. Rettner, D. P. Sanders, H.-C. Kim, W. D. Hinsberg, *Adv. Mater.* **2008**, 20, 16.
4.  X. Yang, L. Wan, S. Xiao, Y. Xu, D. K. Weller, *ACS Nano* **2009**, 3, 7.
5.  O. Hellwig, J. K. Bosworth, E. Dobisz, D. Kercher, T. Hauet, G. Zeltzer, J. D. Risner-Jamtgaard, D. Yaney, R. Ruiz, *Appl. Phys. Lett.* **2010**, 96, 5.
6.  J. K. W. Yang, Y. S. Jung, J.-B. Chang, R. A. Mickiewicz, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Nanotechnol.* **2010**, 5.
7.  J. Y. Cheng, D. P. Sanders, H. D. Truong, S. Harrer, A. Friz, S. Holmes, M. Colburn, W. D. Hinsberg, *ACS Nano* **2010**, 4, 8.
8.  H. Yi, X.-Y. Bao, J. Zhang, C. Bencher, L.-W. Chang, X. Chen, R. Tiberio, J. Conway, H. Dai, Y. Chen, S. Mitra, H.-S. P. Wong, *Adv. Mater.* **2012**, 24, 23.
9.  F. Drolet, G. H. Fredrickson, *Phys. Rev. Lett.* **1999**, 83, 21.
10. K. Rasmussen, G. Kalosakas, *J. Polym. Sci. B* **2002**, 40, 16.
11. S. W. Sides, G. H. Fredrickson, *Polymer* **2003**, 44, 19.
12. H. D. Ceniceros, G. H. Fredrickson, *Multiscale Model. Simul.* **2004**, 2, 3.
13. S. W. Sides, G. H. Fredrickson, *J. Chem. Phys.* **2004**, 121, 10.
14. J.-B. Chang, J. G. Son, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *ACS Nano* **2012**, 6, 3.
15. A. Tavakkoli K. G., K. W. Gotrik, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Science* **2012**, 336, 6086.
16. A. Tavakkoli K. G., A. F. Hannon, K. W. Gotrik, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Adv. Mater.* **2012**, 24, 31.
17. A. Tavakkoli K. G., S. M. Nicaise, A. F. Hannon, K. W. Gotrik, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Small* **2014**, 10, 3.
18. J.-B. Chang, H. K. Choi, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Commun.* **2014**, 5, 3305.
19. A. F. Hannon, K. W. Gotrik, C. A. Ross, A. Alexander-Katz, *ACS Macro Lett.* **2013**, 2, 3.
20. A. F. Hannon, Y. Ding, W. Bai, C. A. Ross, A. Alexander-Katz, *Nano Lett.* **2014**, 14, 1.
21. M. Gu, A. Perales, *Phys. Rev. E* **2012**, 86, 1.
22. F. Barahona, *J. Phys. A: Math. Gen.* **1982**, 15, 10.
23. J. K. W. Yang, K. K. Berggren, *J. Vac. Sci. Technol. B* **2007**, 25, 6.
24. T. Tanaka, M. Morigami, N. Atoda, *Jpn. J. Appl. Phys.* **1993**, 32, 12B.
25. Y. Yamashita, *Jpn. J. Appl. Phys.* **1996**, 35, 4A.
26. A. Kawai, K. Suzuki, *Jpn. J. Appl. Phys.* **2006**, 45, 6B.
27. N. Politakos, E. Ntoukas, A. Avgeropoulos, V. Krikorian, B. D. Pate, E. L. Thomas, R. M. Hill, *J. Polym. Sci. B* **2009**, 47, 23.
28. B. O'Driscoll, R. A. Kelly, M. Shaw, P. Mokarian-Tabari, G. Liontos, K. Ntetsikas, A. Avgeropoulos, N. Petkov, M. A. Morris, *Eur. Polym. J.* **2013**, 49, 11.
29. C. C. Kathrein, W. Bai, J. A. Currivan-Incorvia, G. Liontos, K. Ntetsikas, A. Avgeropoulos, A. Böker, L. Tsarkova, C. A. Ross, *Chem. Mater.* **2015**, 27, 19.

30. K. Lee, M. Kreider, W. Bai, L.-C. Cheng, S. S. Dinachali, K.-H. Tu, T. Huang, K. Ntetsikas, G. Liontos, A. Avgeropoulos. C. A. Ross, *Nanotechnology* **2016**, 27, 46.

31. E. Ising, *Z. Phys.* **1925**, 31.

32. F. F. Lupi, T. J. Giammaria, M. Ceresoli, G. Seguini, K. Sparnacci, D. Antonioli, V. Gianotti, M. Laus, M Perego, *Nanotechnology* **2013**, 24, 31.

33. M. P. Stoykovich, M. Müller, S. O. Kim, H. H. Solak, E. W. Edwards, J. J. de Pablo, P. F. Nealey, *Science* **2005**, 308, 5727.

34. G. M. Wilmes, D. A. Durkee, N. P. Balsara, J. A. Liddle, *Macromolecules* **2006**, 39, 7.

# Chapter 5

# Ising-model-based simulation in two-dimensional post lattice

In this chapter, we demonstrate an Ising-model-based simulation method for calculating the minimum Hamiltonian configuration using the simulated annealing algorithm. We discuss the effect of changing the two simulated annealing parameters $n$ and $\alpha$. We validate our Ising lattice setup in two-dimensional post array by comparing the Ising model simulation results with previously reported experimental and simulation results. The Ising model simulation method could have potential applications in template design as well as block copolymer pattern prediction.

## 5.1   Introduction

In the previous chapter, we implemented an Ising lattice setup in directed self-assembly (DSA) of block copolymers using a square lattice post template. The Ising states, $+1$ or $-1$, were defined between each pair of adjacent posts based on whether the posts were connected by a polydimethyl siloxane (PDMS) cylinder ($\sigma_j = +1$) or not ($\sigma_j = -1$). We measured properties of the two-state system such as distribution of the states and nearest-neighbor correlation, and determined the two Ising model parameters $J$ and $h$ required for performing simulated annealing. We also discussed how to define input states and boundary conditions by placing incommensurate double posts to fix

specific states. The next step is to validate our Ising lattice setup by building an Ising-model-based simulation for calculating the minimum Hamiltonian configuration given some boundary conditions, and comparing the Ising model simulation results with experimental and self-consistent field theory (SCFT) simulation results. We used the simulated annealing algorithm to perform Ising model simulations.

The Ising model simulation method can act as a valuable tool for block copolymer DSA in addition to Ising-model-based computation since block copolymers naturally act as a lithography material.[1-8] The simplicity of the Ising model allows fast simulation of block copolymer morphology that can complement SCFT simulations and further enhance the potential of block copolymer lithography. The Ising model simulations may be used to design new templates for DSA and predict the resulting block copolymer morphology on a given template, impacting various self-assembly applications such as rule-based template design[9-11] and inverse design of complex circuit-relevant patterns.[12,13] Moreover, verifying that the Ising model simulations can correctly calculate the equilibrium block copolymer morphology predicted by SCFT simulations suggests the possibility of performing Ising-model-based computation.[14,15]

This chapter describes an Ising model simulation method for the Ising lattice setup defined on a two-dimensional post lattice template. We discuss parameter choices for the simulated annealing algorithm, and compare the Ising model simulation results with previously reported experimental and SCFT simulation results on tile-based design rules.[9]

## 5.2   Simulation methods

Simulated annealing is a probabilistic technique used to approximate the global optimum of a given function in a large search space.[16-18] Starting from a random initial solution, a neighbor

solution is selected by inverting one random state in the initial solution. The Ising Hamiltonian of the initial solution and the neighbor solution are calculated. If the Hamiltonian of the neighbor solution is lower, the initial solution is updated to the neighbor solution with probability of 1. Otherwise, the initial solution is still updated to the neighbor solution with an acceptance probability which is a function of the two Hamiltonians and a parameter called annealing temperature. This step is necessary to escape from a local minimum. Then, the process is repeated as the annealing temperature decreases until a threshold value is reached. The acceptance probability is decreased as the annealing temperature is decreased so that the evolution of the solution eventually becomes more sensitive to finer energy variations. After sufficient iteration steps, the global minimum of the Ising Hamiltonian can be approximated.

**Figure 5.1** shows a flowchart of the simulated annealing algorithm we implemented. Given a boundary condition consisting of fixed states at specific locations, a random initial solution $\sigma$ was generated where the boundary states were fixed while the remaining states were randomly chosen between +1 and −1. For each iteration, we generated a neighbor solution $\sigma'$ by randomly selecting a non-boundary state and inverting it. The two Ising Hamiltonians $H(\sigma)$ and $H(\sigma')$ were calculated from the equation

$$H(\sigma) = -J \sum_{\langle i\,j \rangle} \sigma_i \sigma_j - h \sum_j \sigma_j$$

where $J = h < 0$ based on results from the previous chapter. Moreover, we set $J = h = -1$ since only the relative magnitude of $J$ and $h$ was needed to determine the inequality between $H(\sigma)$ and $H(\sigma')$. If $H(\sigma') < H(\sigma)$, $\sigma$ was updated to $\sigma'$ for the next iteration. Otherwise, $\sigma$ was still updated to $\sigma'$ with an acceptance probability given by

$$P(H(\sigma), H(\sigma'), T_k) = exp\left(-\frac{H(\sigma') - H(\sigma)}{T_k}\right)$$

where $T_k$ is the annealing temperature. For the first iteration, $T_k = T_1 = 1$. We performed $n$ iterations for each annealing temperature $T_k$, and then decreased the annealing temperature using an annealing schedule $T_{k+1} = \alpha T_k$ where $\alpha$ is a scaling factor ($0 \leq \alpha < 1$). The simulated annealing algorithm reduces to the greedy algorithm for $\alpha = 0$. Iterations were continued until $T_k$ reached a threshold value, $T_{\text{threshold}} = 0.01T_1 = 0.01$. In our simulations, iterations were performed $n$ times each at $\left\lfloor \log_\alpha \frac{T_{\text{threshold}}}{T_1} \right\rfloor + 1$ different annealing temperatures. The effect of changing iterations per annealing temperature $n$ and scaling factor $\alpha$ is discussed in the next section.
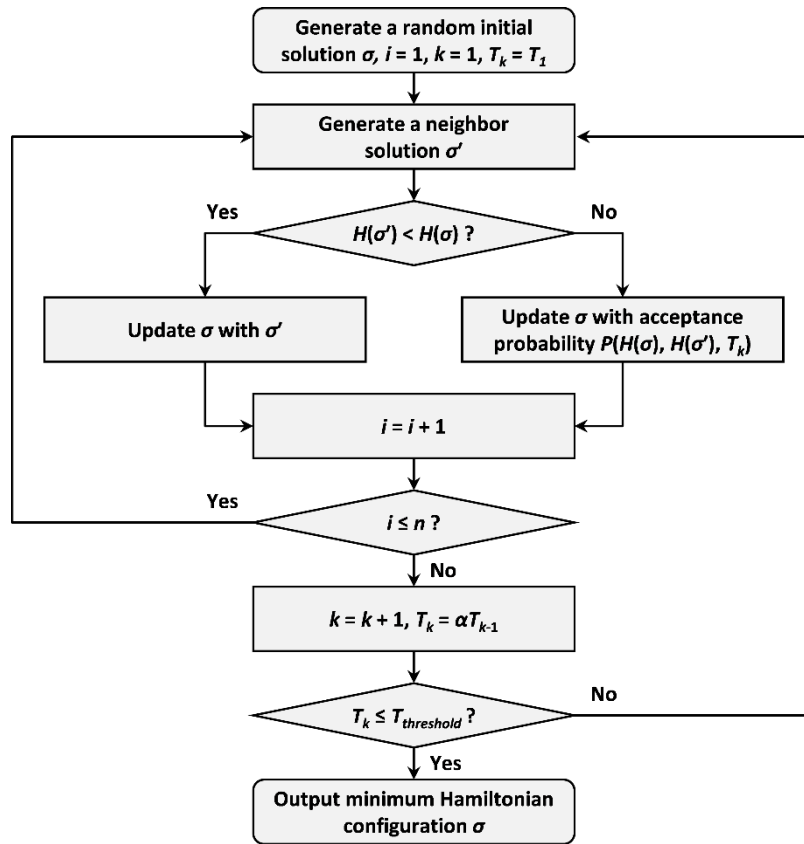


**Figure 5.1** Flowchart of the simulated annealing algorithm implemented in this chapter.

## 5.3    Results and discussion

In this section, we describe Ising-model-based simulation results for different boundary conditions. We compare the Ising model simulation results with previously reported experimental and SCFT simulation results.

### 5.3.1  Effect of changing simulated annealing parameters $n$ and $\alpha$

For the simulated annealing algorithm to find an approximate minimum Hamiltonian configuration that is sufficiently close or equal to the global minimum Hamiltonian configuration, we have to select an appropriate annealing schedule. In our implementation, the number of iterations per annealing temperature $n$ and scaling factor $\alpha$ determine how fast the annealing temperature $T_k$ is decreased. Although the probability of finding a global optimal solution approaches 1 as $n$ is increased and $\alpha$ approaches 1,[19] it is not desirable to excessively extend the annealing schedule since we must also consider the allotted time budget.

Figure 5.2 demonstrates the effect of increasing $n$ on the final minimum Hamiltonian configuration. We created a $100 \times 100$ array of ferromagnetic Ising states with $J = 1$, $h = 0$, and continuous boundaries. $\alpha$ was fixed to 0.9, resulting in 44 different annealing temperatures between $T_1 = 1$ and $T_{44} = 0.0108$. Starting from a random initial solution in Figure 5.2a, the minimum Hamiltonian configuration was calculated for different values of $n$ ranging from 100 to 1,000,000. The global minimum Hamiltonian configuration is an array of only +1's (white) or −1's (black), which was achieved for $n = 1,000,000$. Evident grains were observed starting from $n = 10,000$, and we chose $n$ equal to the total number of states for subsequent simulations where we also increased $\alpha$ to achieve finer temperature steps.
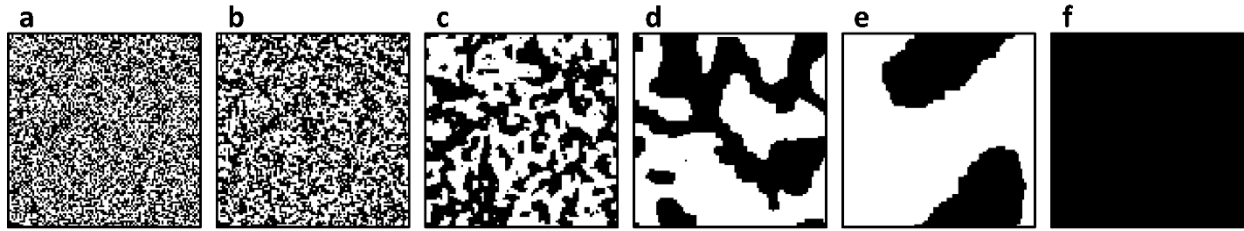
**Figure 5.2** Effect of increasing $n$ on the final minimum Hamiltonian configuration. A $100 \times 100$ array of ferromagnetic Ising states after the simulated annealing was completed. (a) Random initial array. (b-f) $\alpha$ was fixed to 0.9 while $n$ was set to (b) 100, (c) 1,000, (d) 10,000, (e) 100,000, and (f) 1,000,000.

**Figure 5.3** demonstrates the effect of increasing $\alpha$ on the final minimum Hamiltonian configuration using the same $100 \times 100$ array of ferromagnetic Ising states. $n$ was fixed to 10,000, the total number of states, while $\alpha$ was varied between 0.2 and 0.999. The global minimum Hamiltonian configuration was achieved for $\alpha = 0.999$, where 10,000 iterations were performed for each of the 4,603 different annealing temperatures. For subsequent simulations, we chose $\alpha$ equal to 0.99 to reduce simulation time. These parameters were sufficient to achieve the global optimum for most simulations. When the global optimal solution was not found, we increased $n$ beyond the total number of states, and increased $\alpha$ closer to 1.
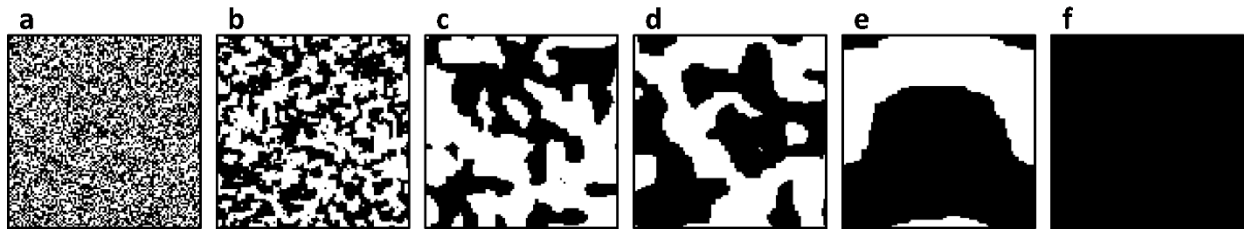


**Figure 5.3** Effect of increasing $\alpha$ on the final minimum Hamiltonian configuration. A $100 \times 100$ array of ferromagnetic Ising states after the simulated annealing was completed. (a) Random initial array. (b-f) $n$ was fixed to 10,000, the total number of states, while $\alpha$ was set to (b) 0.2 (three different $T_k$'s), (c) 0.8 (21 different $T_k$'s), (d) 0.9 (44 different $T_k$'s), (e) 0.99 (459 different $T_k$'s) and (f) 0.999 (4,603 different $T_k$'s).

## 5.3.2 Reproducing simple boundary conditions

We validate our Ising lattice setup by reproducing experimental and SCFT simulation results for various boundary conditions using Ising model simulations. Before introducing any boundary conditions, simulated annealing of the Ising states already produced patterns that are similar to the block copolymer patterns formed on two-dimensional post lattice templates. **Figure 5.4** illustrates how the Ising model simulation results compare with experimental results. **Figure 5.4a** shows a sample pattern resulting from simulated annealing of 800 antiferromagnetic Ising states with $J = h = -1$ and continuous boundaries. The two simulated annealing parameters were chosen to intentionally avoid finding the global optimum solution for better comparison with the block copolymer pattern shown in **Figure 5.4b**. Close similarity between the two patterns suggests that our Ising lattice design is reasonable and the Ising model can be applied to DSA of block copolymers.
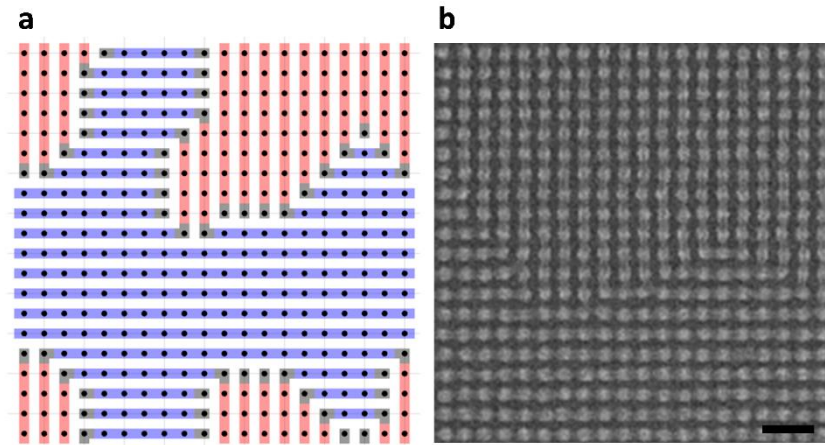


**Figure 5.4** Comparison between a sample pattern resulting from the Ising model simulations and from DSA of block copolymers. (a) Simulated annealing result of 800 antiferromagnetic Ising states defined between a 20 × 20 array of posts. The pattern was colored based on alignment orientation for easier visualization. (b) SEM image of a block copolymer pattern formed on the post lattice template. Scale bar, 100 nm.

Next, we reproduced the four template design rules reported by Chang et al.[9] using the Ising model simulations with specific boundary conditions. In the work by Chang et al., four types of template design rules were developed by placing periodic double posts in the post lattice template. The double post arrangements for the four design rules are shown in **Figures 5.5a-5.5d**. To determine the boundary conditions imposed by a specific template, we interpreted the double posts as having the same effect as fixing their adjacent states to +1's or −1's as described in the previous chapter. **Figures 5.5e-5.5h** show the minimum Hamiltonian configuration calculated for each boundary condition. The Ising model simulation results agreed with the reported experimental results as well as the minimum free energy morphology calculated from SCFT simulations.
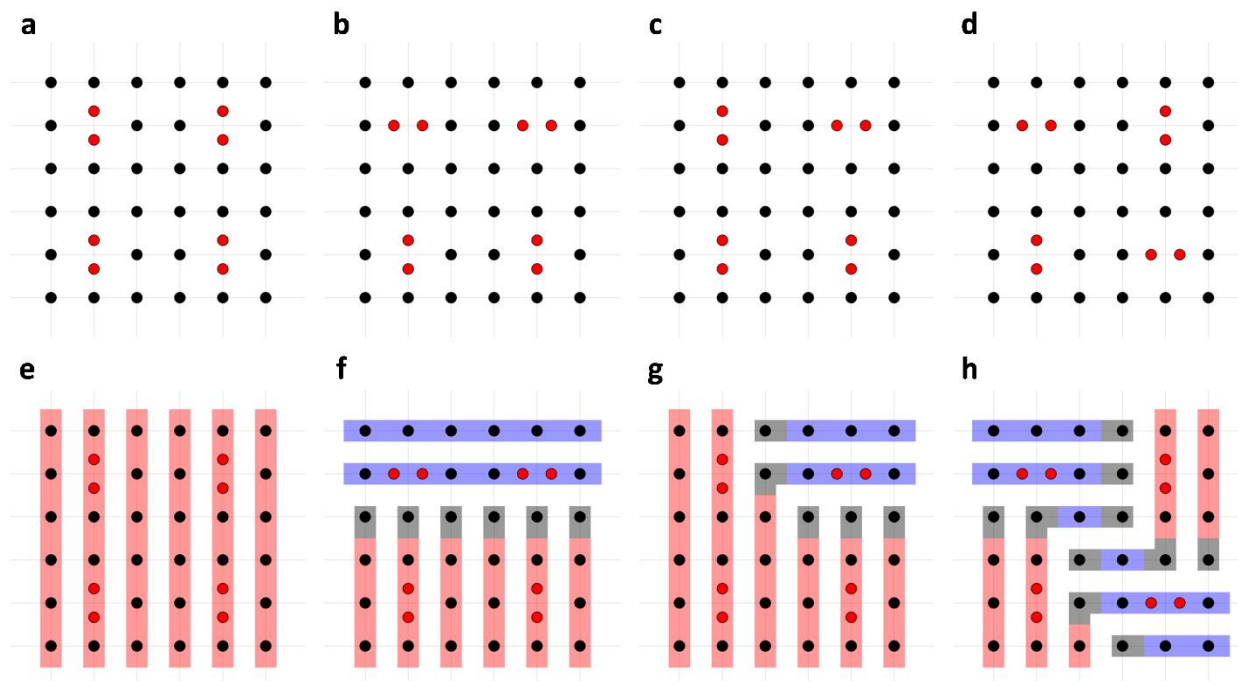


**Figure 5.5** Ising model simulation results for the four template design rules. (a-d) Four different arrangements of four horizontally and vertically oriented double posts (red dot). (e-h) Possible minimum Hamiltonian configurations resulting from the double post templates. The Ising model simulation results agreed with previously reported results.[9] Different minimum Hamiltonian configurations may be achieved depending on how the top, bottom, left, and right boundary states are initially defined.

**Figure 5.6** shows the minimum Hamiltonian configurations calculated for the second template design rule **(Figure 5.5b)** assuming $h < 0$ **(Figure 5.6a)** and $h = 0$ **(Figures 5.6b-5.6d)**. In the previous chapter, we concluded $2J < h < 0$ by measuring the distribution of the center state given its four nearest-neighbor configurations. The Ising model simulation results also suggest $h < 0$ since the minimum Hamiltonian configuration with $h = 0$ result in formation of energetically unfavorable bends[20,21] and T-junctions.[22-24] When the four nearest neighbors have two $+1$ states and two $-1$ states, setting $h < 0$ prevents the center state from remaining in $+1$ state, turning bends and T-junctions into terminations.
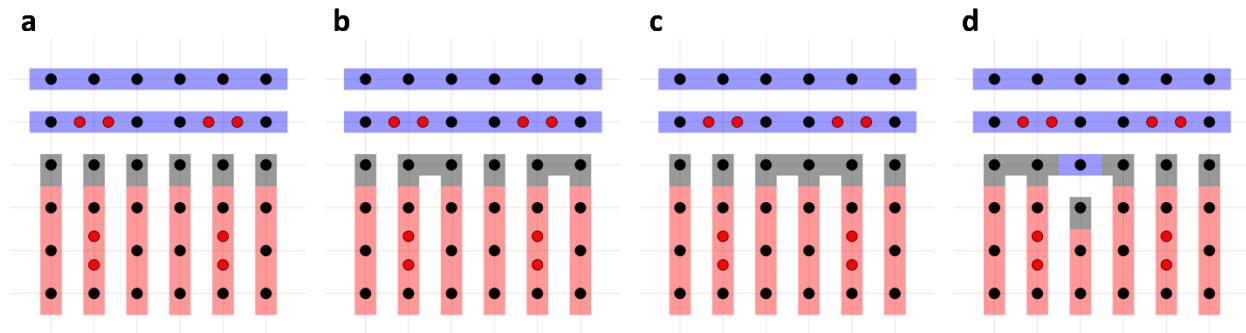


**Figure 5.6** Ising model simulation results for the second template design rule. (a) Minimum Hamiltonian configuration assuming $h < 0$. (b-d) Different minimum Hamiltonian configurations assuming $h = 0$, showing formation of bends and T-junctions.

**Figure 5.7** shows the minimum Hamiltonian configurations calculated for the fourth template design rule **(Figure 5.5d)**. For the fourth double post arrangement, Chang et al. have reported several frequently observed block copolymer patterns, and the most frequent pattern was formed in only 20% of the templates.[9] Similar to these experimental results, the Ising model simulation produced two minimum Hamiltonian configurations with the same top, bottom, left, and right boundary states.
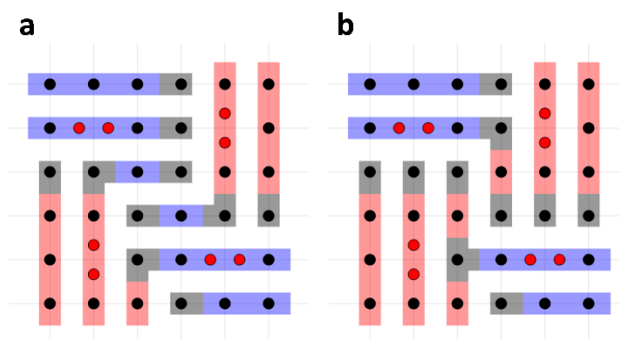
**Figure 5.7** Ising model simulation results for the fourth template design rule. (a) One possible minimum Hamiltonian configuration. (b) Another possible minimum Hamiltonian configuration with the same boundary states.

### 5.3.3 Reproducing complex boundary conditions

Using the Ising model simulations, we were able to reproduce more complex patterns consisting of dense bends and terminations. A target post lattice template with periodic double posts is shown in **Figure 5.8a**. A boundary condition was derived from the positions and orientations of the double posts. As the number of iteration steps in simulated annealing was increased starting from a random initial binary state array **(Figure 5.8b)**, the minimum Hamiltonian configuration **(Figure 5.8g)** eventually reached the reported experimental result except for minor defect formations **(Figure 5.8h)**, and agreed with the minimum free energy morphology calculated from SCFT simulations.[9] A similar set of Ising model simulations was performed to reproduce a different complex pattern as shown in **Figure 5.9**. The Ising model simulations took less than 10 seconds to perform more than 200,000 iteration steps on a computer with a 2.2 GHz processor and 8 GB of random access memory, which is orders of magnitude faster than typical SCFT simulation methods. The global optimum solution was reached only after a few tens of thousands of iteration steps.
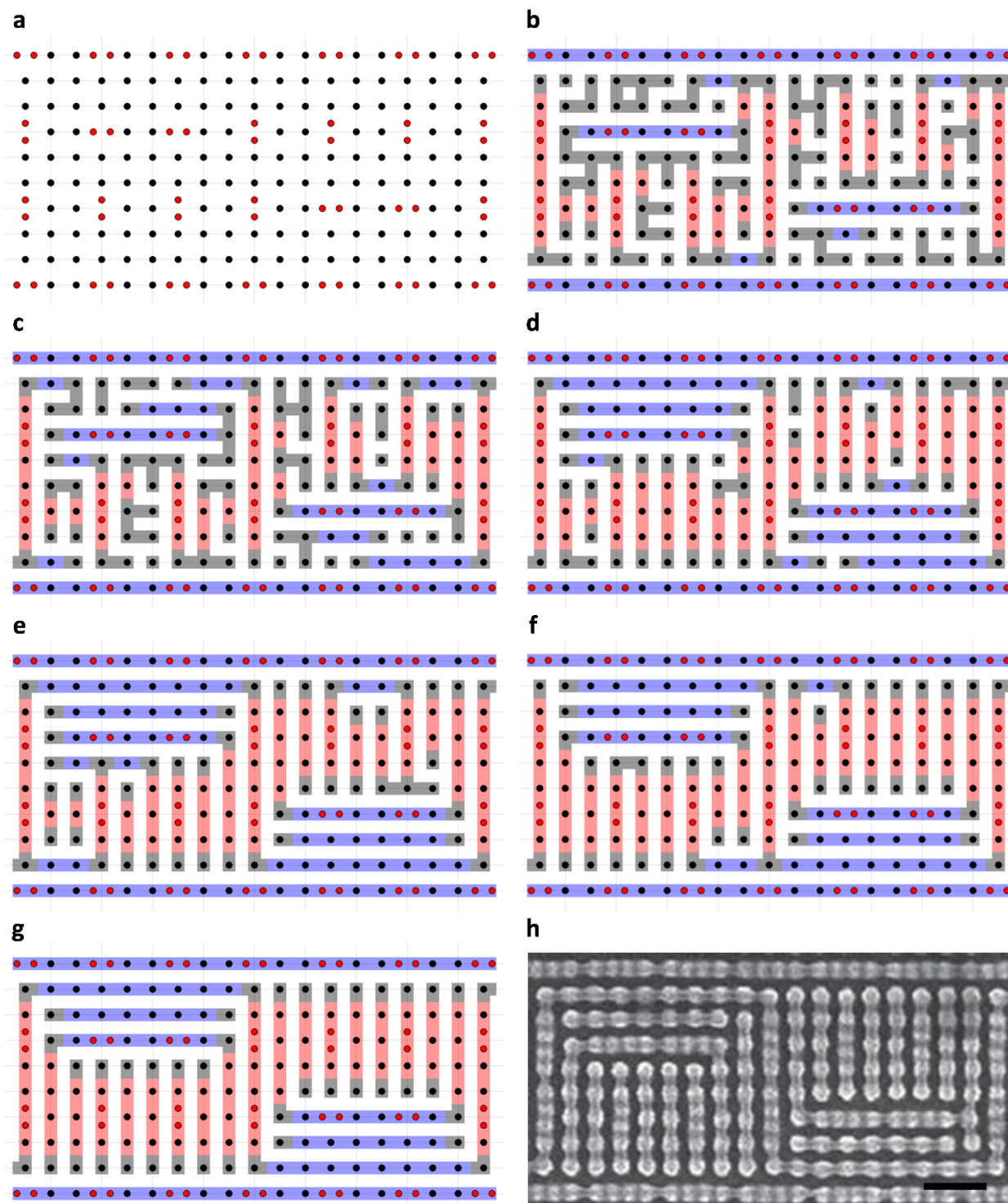
**Figure 5.8** Ising model simulation results for the first complex pattern. (a) Target post lattice template. (b) Random initial array. (c-g) Simulation result after (c) 100, (d) 500, (e) 1,000, (f) 5,000, and (g) 50,000 iteration steps. The minimum Hamiltonian configuration was found after 50,000 iteration steps. (h) Actual block copolymer pattern.[9] Scale bar, 100nm. Redrawn from reference 9.
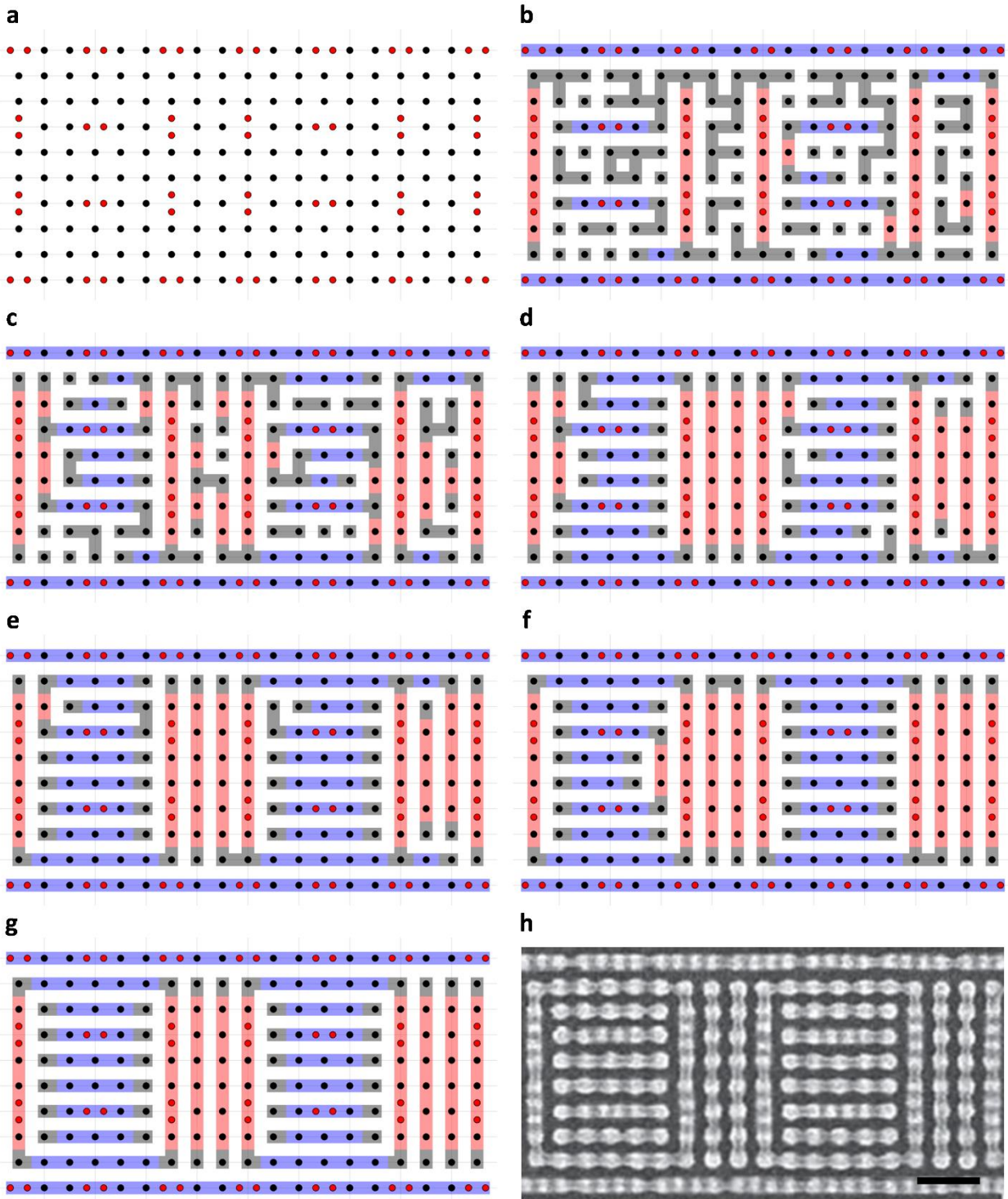
**Figure 5.9** Ising model simulation results for the second complex pattern. (a) Target post lattice template. (b) Random initial array. (c-g) Simulation result after (c) 100, (d) 500, (e) 1,000, (f) 5,000, and (g) 50,000 iteration steps. The minimum Hamiltonian configuration was found after 50,000 iteration steps. (h) Actual block copolymer pattern.[9] Scale bar, 100nm. Redrawn from reference 9.

## 5.4    Conclusion

We developed an Ising-model-based simulation method to calculate the minimum Hamiltonian configuration given specific boundary conditions. We studied the effect of changing the simulated annealing parameters to determine reasonable parameter values for simulation. Ising model simulation results were compared with previously reported results to validate our Ising lattice setup. We expect the Ising model simulations to broadly impact the use of block copolymer DSA as a viable method for lithography. The simulations can complement SCFT simulations to perform fast template design, and assist rule-based and inverse design. In the next chapter, we use the Ising model simulations to design templates with specific boundary conditions that encode desired Boolean operations. We demonstrate several types of Boolean logic gates based on our Ising lattice setup, and discuss theoretical strategies for mapping combinatorial optimization problems into the Ising Hamiltonian.

# References

1. M. Park, C. Harrison, P. M. Chaikin, R. A. Register, D. H. Adamson, *Science* **1997**, 276, 5317.
2. R. R. Li, P. D. Dapkus, M. E. Thompson, W. G. Jeong, C. Harrison, P. M. Chaikin, R. A. Register, D. H. Adamson, *Appl. Phys. Lett.* **2000**, 76, 13.
3. J. Y. Cheng, C. A. Ross, V. Z.-H. Chan, E. L. Thomas, R. G. H. Lammertink, G. J. Vancso, *Adv. Mater.* **2001**, 13, 15.
4. J. Y. Cheng, C. A. Ross, E. L. Thomas, H. I. Smith, G. J. Vancso, *Appl. Phys. Lett.* **2002**, 81, 19.
5. R. Ruiz, H. Kang, F. A. Detcheverry, E. Dobisz, D. S. Kercher, T. R. Albrecht, J. J. de Pablo, P. F. Nealey, *Science* **2008**, 321, 5891.
6. C. Tang, E. M. Lennon, G. H. Fredrickson, E. J. Kramer, C. J. Hawker, *Science* **2008**, 322, 5900.
7. S.-W. Chamg, V. P. Chuang, S. T. Boles, C. A. Ross, C. V. Thompson, *Adv. Funct. Mater.* **2009**, 19, 15.
8. C. M. Bates, M. J. Maher, D. W. Janes, C. J. Ellison, C. G. Wilson, *Macromolecules* **2014**, 47, 1.
9. J.-B. Chang, H. K. Choi, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Commun.* **2014**, 5, 3305.
10. H. Yi, X.-Y. Bao, J. Zhang, C. Bencher, L.-W. Chang, X. Chen, R. Tiberio, J. Conway, H. Dai, Y. Chen, S. Mitra, H.-S. P. Wong, *Adv. Mater.* **2012**, 24, 23.
11. H. Yi, X.-Y. Bao, R. Tiberio, H.-S. P. Wong, *Nano Lett.* **2015**, 15, 2.
12. A. F. Hannon, K. W. Gotrik, C. A. Ross, A. Alexander-Katz, *ACS Macro Lett.* **2013**, 2, 3.
13. A. F. Hannon, Y. Ding, W. Bai, C. A. Ross, A. Alexander-Katz, *Nano Lett.* **2014**, 14, 1.
14. M. Gu, A. Perales, *Phys. Rev. E* **2012**, 86, 1.
15. F. Barahona, *J. Phys. A: Math. Gen.* **1982**, 15, 10.
16. S. Kirkpatrick, C. D. Gelatt Jr., M. P. Vecchi, *Science* **1983**, 220, 4598.
17. S. Kirkpatrick, *J. Stat. Phys.* **1984**, 34, 5.
18. A. Corana, M. Marchesi, C. Martini, S. Ridella, *ACM Trans. Math. Softw.* **1987**, 13, 3.
19. V. Granville, M. Krivanek, J.-P. Rasson, *IEEE Trans. Pattern Anal. Mach. Intell.* **1994**, 16, 6.
20. M. P. Stoykovich, M. Müller, S. O. Kim, H. H. Solak, E. W. Edwards, J. J. de Pablo, P. F. Nealey, *Science* **2005**, 308, 5727.
21. G. M. Wilmes, D. A. Durkee, N. P. Balsara, J. A. Liddle, *Macromolecules* **2006**, 39, 7.
22. S. P. Gido, E. L. Thomas, *Macromolecules* **1994**, 27, 21.
23. E. Burgaz, S. P. Gido, *Macromolecules* **2000**, 33, 23.
24. D. Duque, K. Katsov, M. Schick, *J. Chem. Phys.* **2002**, 117, 22.

# Chapter 6

# Ising-model-based computation by block copolymer self-assembly

In this chapter, we discuss methods for achieving Ising-model-based computation using block copolymer self-assembly. We use the previously developed Ising-model-based simulation method to prototype template designs that can encode specific Boolean operations into the ground states of the Ising lattice. We implement simple Boolean logic gates as a proof-of-concept demonstration of computation. We also discuss potential challenges of mapping complex optimization problems to the Ising lattice, and study theoretical strategies for addressing these issues.

## 6.1   Introduction

Previously, we showed that block copolymer self-assembly maps rigorously onto the Ising model, and the free energy minimization problem in directed self-assembly (DSA) can be cast into the language of Ising-model-based optimization. Our next challenge is to use this new platform for Ising lattices to implement interesting and useful computational functions such as universal Boolean logic operations and combinatorial optimization problems. To map arbitrary Boolean logic to the Ising lattice, specific Hamiltonians that encode the desired Boolean operation must be designed first. To map combinatorial optimization problems to the Ising lattice, we must be able

to control the two Ising model parameters $J$ and $h$ so that desired weights from the optimization problem can be assigned to the parameters of the Ising lattice.

Although it has been previously proven that universal computation can be encoded into the ground states of Ising lattices,[1-8] it is not a trivial task to design the specific Hamiltonians that can embed desired Boolean operations. Furthermore, material properties of the block copolymer impose additional restrictions on the Ising lattice. For example, the number of nearest-neighbor states must be exactly four due to the square lattice geometry of the post lattice template. The two Ising model parameters $J$ and $h$ are fixed to a negative value independent of lattice locations. The input states are inherently static since they are already patterned on the substrate by electron-beam lithography. Since the block copolymer structures are continuously formed across the whole substrate, a boundary must be placed to isolate the templated regions from untemplated regions. This complicates the design of desired Hamiltonians because the outer boundaries will also interact with the inner Ising lattice. These issues must be addressed before we can build a practical computing system using block copolymers.

In this chapter, we use the Ising lattice setup from the previous chapters and demonstrate Ising-model-based computation by designing Boolean logic gates such as a buffer, inverter, wire (cascaded buffer), fan-out gate, and three-input majority gate. The logic gates are defined on a post lattice template with double posts acting as boundaries. Input and output states are chosen inside the boundary, and double posts are used to define static inputs. Yield of the designed logic gates are measured. We also discuss theoretical strategies to map combinatorial optimization problems to the Ising lattice.

## 6.2 Experimental and simulation methods

For template fabrication, block copolymer self-assembly, reactive-ion etching, and metrology, the same experimental methods as described in Chapter 4 were used. For Ising-model-based simulations, the same simulated annealing method described in Chapter 5 was used.

## 6.3 Results and discussion

In this section, we describe both experimental and Ising-model-based simulation results for various Boolean logic gate designs to demonstrate Ising-model-based computation. We discuss theoretical approaches for mapping complex optimization problems to the Ising lattice.

### 6.3.1 Designing a buffer/inverter

Ising Hamiltonian designs for various Boolean operations have been previously developed for other types of Ising machines.[9-14] With antiferromagnetic interactions in magnetic logic for example, a buffer can be built using three states and a three-input majority gate can be built using five states.[13,14] However, these existing designs for Ising Hamiltonians that encode specific Boolean operations are not applicable to our Ising lattice due to the constraints discussed in the previous section. Here, we show that computation can be encoded into the ground states of our Ising lattices by demonstrating designs for simple Boolean logic gates.

The first pair of logic gates we design is a buffer and inverter. The same boundary conditions result in two different Boolean functions depending on our definition of input and output state. We fabricated a post array consisting of a box-shaped boundary defined by incommensurate double posts as shown in **Figure 6.1**. Before placing additional double posts inside the boundary to define an input state, two minimum Hamiltonian configurations exist given

the boundary conditions. In the ground state, the block copolymer structures can either be vertically aligned as shown in **Figure 6.1a**, or horizontally aligned by symmetry as shown in **Figure 6.1d**. After block copolymer self-assembly, the experimental results **(Figure 6.1b and Figure 6.1e)** and the minimum Hamiltonian configurations calculated from the Ising model simulation **(Figure 6.1c and Figure 6.1f)** agreed with our prediction.
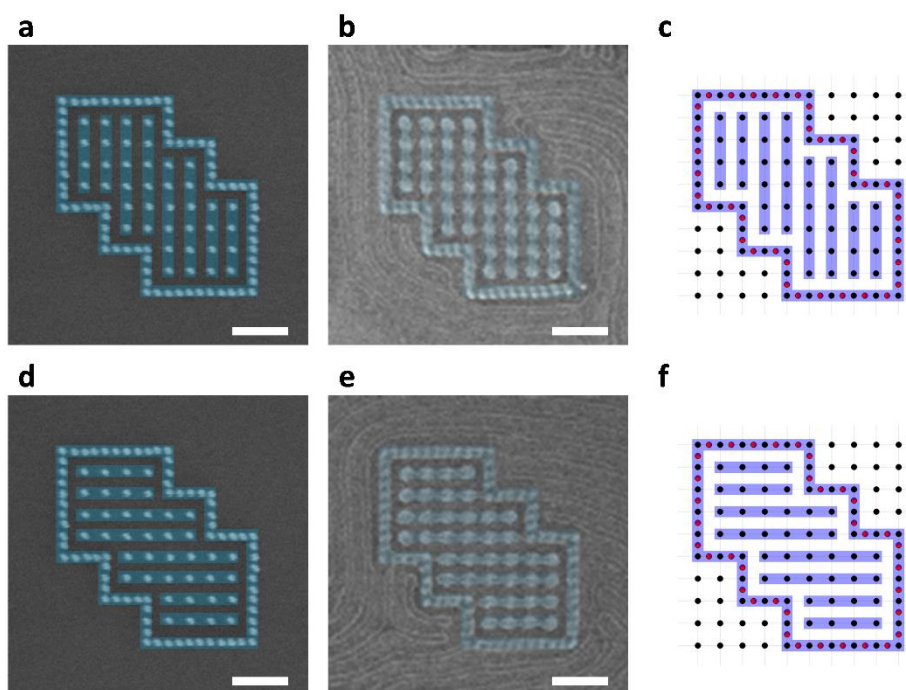


**Figure 6.1** Block copolymer morphology inside a double post boundary for designing a buffer and inverter. Input state has not been defined yet. (a,d) Scanning electron microscope (SEM) image of the post array with a boundary consisting of double posts. The block copolymer structures (blue line) can be aligned either vertically or horizontally in the minimum free energy morphology. (b,e) SEM image of the resulting block copolymer pattern inside the boundary. (c,f) Two minimum Hamiltonian configurations calculated from the Ising model simulation. Scale bars, 100 nm.

Next, we chose an input state and an output state inside the post array. We defined the input state by placing a double post as shown in **Figure 6.2**. The input state can be set to +1 (interpreted as a 1 bit) by placing a double post in the input state **(Figure 6.2a)** or set to −1 (interpreted as a 0

bit) by placing a double post in the state adjacent to the input state **(Figure 6.2d)**. After block

copolymer self-assembly, the output state was set equal to the input state since the overall block

copolymer alignment orientation inside the boundary was determined by the orientation of the

double post at the input state **(Figure 6.2b and Figure 6.2e)**. The Ising model simulation results

agreed with the experimental results **(Figure 6.2c and Figure 6.2f)**. If we define the output state

to be inside the red dashed box, the design acts as a buffer, as the input and output states have the

same parity. We can simply convert the design into an inverter by defining the output state to be

inside the blue dashed box. Since the output state for buffer and inverter are defined adjacent to

each other, they must have different ground state parity due to antiferromagnetic interactions. The

yield for the buffer and inverter was 86.6%.
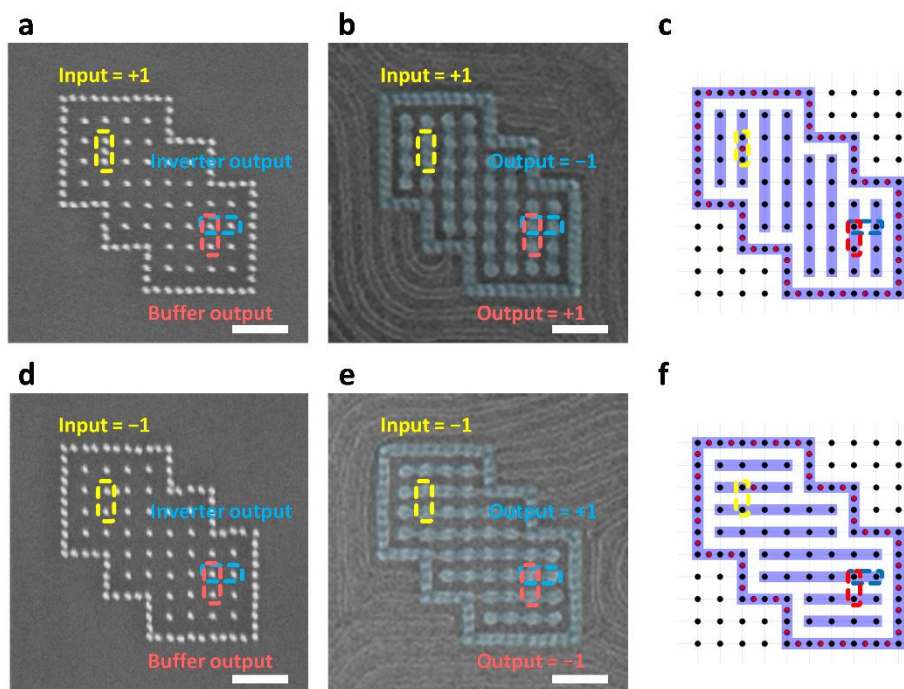


**Figure 6.2** Design for a buffer and inverter. Input state (yellow dashed box), output state for buffer (red dashed box), and output state for inverter (blue dashed box) are indicated. (a,d) SEM image of the post array where the input state was defined using a double post. (b,e) SEM image of the resulting block copolymer pattern. The input state determined the overall block copolymer

alignment orientation, also setting the output state to its desired parity. (c,f) Ising model simulation results. Scale bars, 100 nm.

For non-ground state block copolymer morphologies that resulted in the output state being incorrect, we were able to calculate the difference in the Ising Hamiltonian relative to the minimum Hamiltonian configuration. **Figure 6.3** shows the minimum Hamiltonian configuration with vertically aligned block copolymer structures **(Figure 6.3a)** and a non-ground state configuration with formation of an additional bend and terminations **(Figure 6.3b)**. The input state was set to +1 for both cases. For the non-ground state configuration, formation of an undesired bend changed the alignment orientation, setting the output state to −1 instead of +1. For the 96 states inside the double post boundary, the Ising Hamiltonian was $H_1 = 92J + 32h$ for the left configuration and $H_2 = 80J + 34h$ for the right configuration. The difference in the Ising Hamiltonian, $H_2 − H_1 = −12J + 2h$, was greater than 0 since we concluded $2J < h < 0$ from Chapter 4.
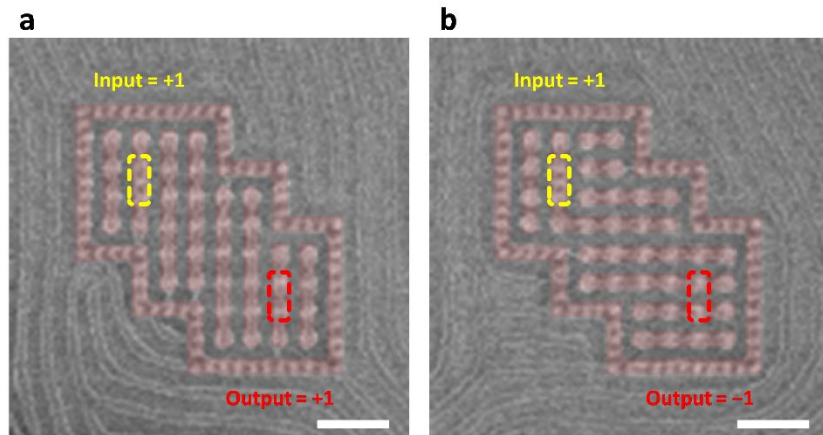


**Figure 6.3** SEM images of the block copolymer pattern formed inside a buffer. Input state (yellow dashed box) and output state (red dashed box) are indicated. (a) In the minimum Hamiltonian configuration, the block copolymer structures were vertically aligned. For the states inside the boundary, the Ising Hamiltonian was $H_1 = 92J + 32h$. (b) With formation of an additional bend and terminations, the non-ground state configuration resulted in an increased Ising Hamiltonian $H_2 = 80J + 34h > H_1$. Scale bars, 100 nm.

## 6.3.2 Designing complex logic gates

We can apply our design approach using boundaries consisting of square boxes to implement other types of Boolean logic gates. Although the logic gate designs must be developed through trial and error by simulating the minimum Hamiltonian configuration inside each design for different input combinations, the Ising model simulation allowed fast pattern prediction. **Figure 6.4** shows a design for a wire composed of two cascaded buffers. We simply extended the box-shaped boundary to implement a larger-sized buffer. The yield was 75.4%. Another example of a modified buffer design is a one-input two-output fan-out gate shown in **Figure 6.5**. The yield was 70.8%.
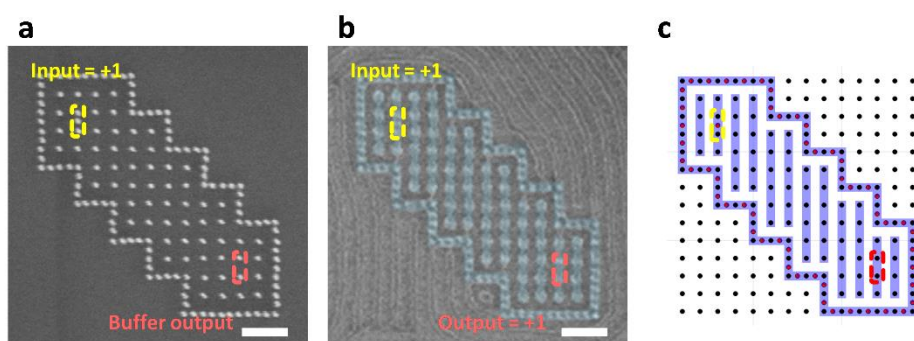


**Figure 6.4** Design for a wire (cascaded buffer). Input state (yellow dashed box) and output state (red dashed box) are indicated. (a) SEM image of the post array. (b) SEM image of the resulting block copolymer pattern. (c) Ising model simulation result. Scale bars, 100 nm.
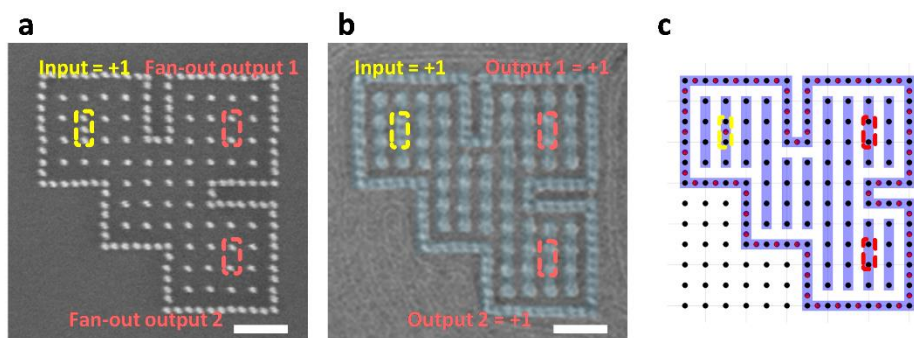


**Figure 6.5** Design for a one-input two-output fan-out gate. Input state (yellow dashed box) and two output states (red dashed box) are indicated. (a) SEM image of the post array. (b) SEM image of the resulting block copolymer pattern. (c) Ising model simulation result. Scale bars, 100 nm.
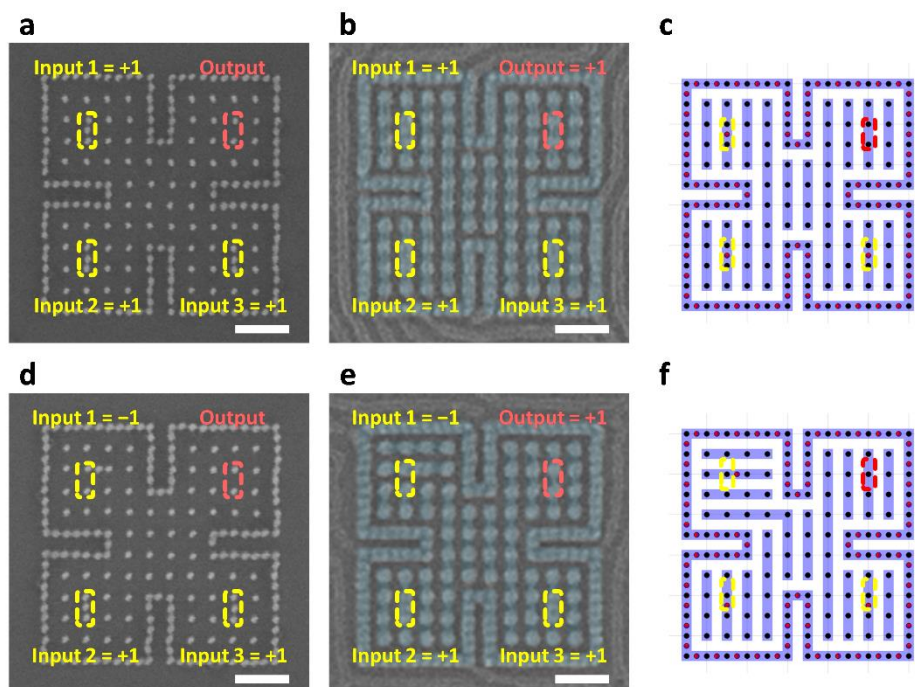
**Figure 6.6** Design for a three-input majority gate. Input states (yellow dashed box) and output state (red dashed box) are indicated. (a,d) SEM image of the post array with (a) three equal inputs and (d) two equal inputs. (b,e) SEM image of the resulting block copolymer pattern. The output state was set equal to the majority input state. (c,f) Ising model simulation results. Scale bars, 100 nm.

By modifying the design for a buffer and inverter, a three-input majority gate can be built as shown in **Figure 6.6**. Similar to the buffer, the block copolymer structures can be horizontally or vertically aligned in the ground state configuration without any input states. When all three input states were set to the same bit as shown in **Figure 6.6a**, the output state also had the same alignment orientation and bit as the three input states. When only two input states had the same bit as shown in **Figure 6.6d**, a bend and/or terminations were formed at the boundary between the two same input states and the differing input state. However, the overall alignment orientation was determined by the two majority input states, and the output state still had the same bit as the two majority input states. In this majority gate design, any of the input or output states can be inverted

by defining an adjacent state as the new input or output. Also, AND gate and OR gate can be implemented by fixing one of the three input states. The yield for the majority gate was 70.0%.

For the logic gates we designed, incorrect outputs were caused by the block copolymer morphology being trapped in the non-ground state configuration, as well as defective structures induced by the double post boundary. The non-ground state configurations were associated with formation of undesired bends and terminations. Such structures can be healed and eventually removed by further optimization of the annealing time and temperature, block copolymer film thickness, and solvent vapor pressure during the annealing process. Other block copolymer materials may be used to improve long-range order. Another source of incorrect outputs was the double post boundary itself. Due to the box-shaped boundary design, several concave corners were formed. At these concave corners, block copolymer structures formed at the boundary were randomly connected to the inner Ising lattice forming a straight line instead of a 90° bend, and resulted in incorrect outputs. We may be able to avoid concave corners by choosing a different boundary shape.

We used the Ising model simulation to design and verify other types of logic gates. **Figure 6.7** shows an alternative design for an AND/OR gate. As with the buffer/inverter and the majority gate, the two input states and the output state can be inverted as desired, converting an AND gate into an OR, NAND, or NOR gate. In the ground state configuration without any input states, the block copolymer structures are horizontally aligned due to the rectangular-shaped center region. If the block copolymer structures are vertically aligned, two additional terminations are formed because of the horizontally long center region, resulting in higher free energy. Therefore, when the two input states are both set to −1 as shown in **Figure 6.7a**, the ground state configuration remains unchanged. Even if one of the input states are set to +1 as shown in **Figure 6.7b**, the templating

effect from the single vertically oriented double post is insufficient to overcome the overall horizontal alignment generated by the rectangular center region. However, if both input states are set to +1 as shown in **Figure 6.7c**, the templating effect from both vertically oriented double posts can shift the overall alignment orientation and change the output state from −1 to +1. Hence, this design acts as an AND gate. **Figure 6.8** shows a design for a more complex logic gate, a three-input OR gate.
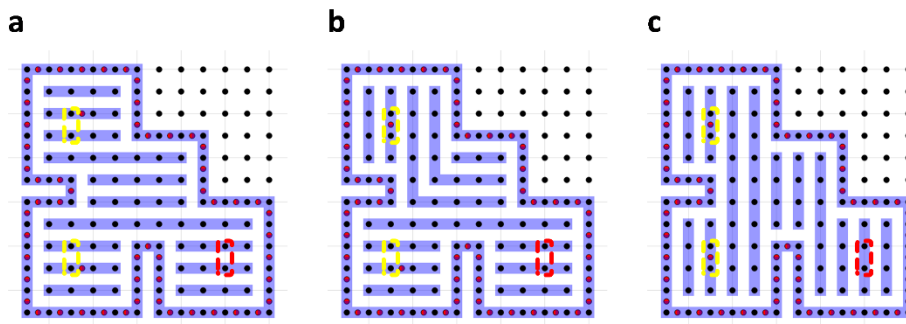


**Figure 6.7** Design for an AND gate. Two input states (yellow dashed box) and an output state (red dashed box) are indicated. (a-c) Ising model simulation results showing the minimum Hamiltonian configuration when the two input states are (a) both −1, (b) +1 and −1, and (c) both +1. The output state is set to +1 only when both input states are set to +1.
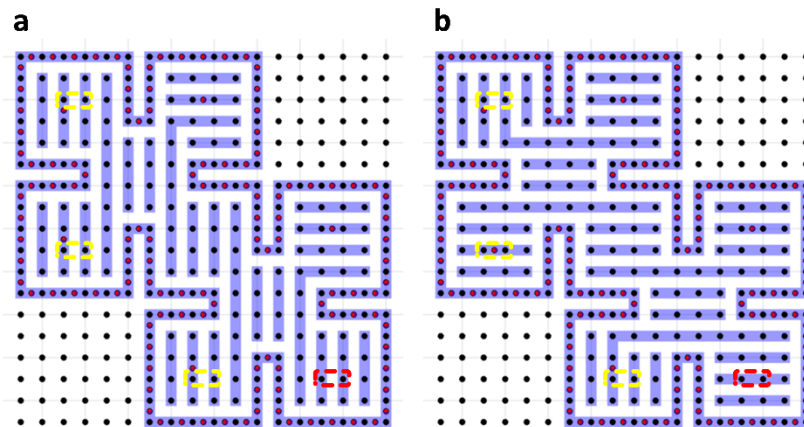


**Figure 6.8** Design for a three-input OR gate. Three input states (yellow dashed box) and an output state (red dashed box) are indicated. (a) The output state is set to −1 only when all inputs states are set to −1. (b) The output state is set to +1 when one or more input states are set to +1.

### 6.3.3  Mapping optimization problems

In the previous sections, we introduced designs for simple Boolean logic gates as a proof-of-concept demonstration of computation. The next step is to think about the possibility of mapping useful optimization problems to this new platform for Ising lattices. A class of problems that naturally maps to the Ising model is combinatorial optimization problems, as solving for the equilibrium states is equivalent to calculating the minimizer of the Ising Hamiltonian function.[1,15-18] Ising formulation has been demonstrated for various combinatorial optimization problems including the Hamiltonian path/cycle problem and the travelling salesman problem.[19] To map such optimization problems, we must investigate how the Ising model parameters $J$ and $h$ can be controlled so that correct weights can be assigned to the Ising lattice. In this section, we discuss the key issues that must be addressed before we can map specific optimization problems to the Ising lattice and ultimately build a practical computing system.

To map specific optimization problems to the Ising lattice, we must first be able to control the two Ising model parameters $J$ and $h$ in some finite range by carefully adjusting the template design. We can slightly alter the post positions without breaking the commensurability since block copolymers can alter their periodicity by ~10% from $L_0$ depending on the periodicity of the template.[20] We have qualitatively demonstrated in previous chapters that $J$'s and $h$'s can be altered by placing a double post, although exact values of $J$'s and $h$'s were not calculated. In **Figure 6.9**, we can see that altering position of a post (yellow dot) or removing a post affects several $h$'s (red cross) associated with the moved post and $J$'s (black dashed line) associated with the affected states. The effect on $J$'s and $h$'s as a function of horizontal and vertical post displacement may be studied by measuring the distribution of the affect states. It may even be possible to directly approximate $J$'s and $h$'s using self-consistent field theory simulations.
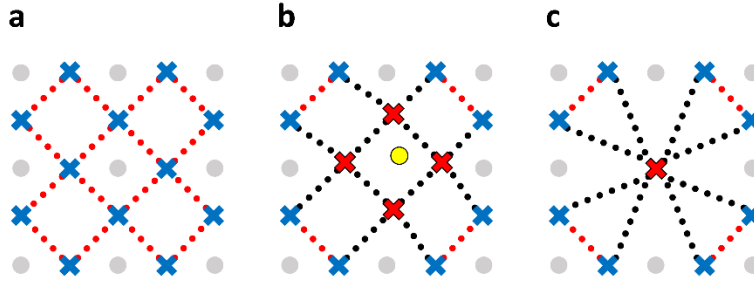
**Figure 6.9** Diagrams of different Ising lattice setups. The posts (grey dot), Ising states (blue cross), and nearest-neighbor interactions (red dashed line) are indicated. (a) Original Ising lattice setup in the square lattice post array. (b) Ising lattice setup after the center post (yellow dot) is slightly moved. (c) Ising lattice setup after the center post is deleted. $J$'s (black dashed line) and $h$'s (red cross) are affected by the change in template design.

Once $J$'s and $h$'s can be controlled by changing the template design, we must develop a neutral boundary, i.e. a boundary with near zero interaction strength ($J \approx 0$). When a computing system is defined on the post lattice template, a boundary must be defined using double posts to isolate the inner Ising states from untemplated block copolymer structures outside the boundary. However, double posts have a strong templating effect which affects the adjacent binary states. By using different geometries such as a cross or T-shape instead of double posts to define the boundary conditions, we may be able to create a boundary that has minimum interaction with its adjacent states. Once neutral boundaries can be achieved, Ising Hamiltonians can be designed with fewer restrictions.

If we can control the $J$'s and $h$'s, we can also resolve the nearest-neighbor restriction caused by the square lattice geometry. The square lattice geometry of the post lattice template restricts the number of nearest-neighbor states to be exactly four. If a specific state has less than four nearest neighbors, we can increase the number of nearest neighbors to four by adding neutral boundaries with $J \approx 0$ adjacent to the state. If the state has more than four nearest neighbors, we can decrease the number of nearest neighbors by dividing the state $\sigma_j$ into two separate states. We can alter the

106

Ising lattice such that the state $\sigma_j$ is split into two states $\sigma_{j'}$ and $\sigma_{j''}$ with $J_{j'j''} \gg 0$ and $h_j = h_{j'} + h_{j''}$. The first condition $J_{j'j''} \gg 0$ forces the two split states $\sigma_{j'}$ and $\sigma_{j''}$ to have the same state value, and the second condition $h_j = h_{j'} + h_{j''}$ ensures that the new state energies sum up to the original state energy.

Furthermore, we can also implement $N$-body interactions ($N > 2$) using arbitrary $J$'s and $h$'s. For example, consider the following Ising Hamiltonian encoding a two-input XOR logic

$$H_1(\sigma) = \sigma_1 \sigma_2 \sigma_3$$

where $\sigma_1$ and $\sigma_2$ are the two input states and $\sigma_3$ is the output state. In a square lattice geometry which only allows nearest-neighbor interactions, the above Hamiltonian cannot be directly implemented using the post lattice template. However, we can introduce an ancillary state $\sigma_*$ to design a new Hamiltonian

$$H_2(\sigma) = (\sigma_1 \sigma_2 + \sigma_1 \sigma_3 + \sigma_2 \sigma_3) + 2\sigma_*(\sigma_1 + \sigma_2 + \sigma_3) + (\sigma_1 + \sigma_2 + \sigma_3 + 2\sigma_*)$$

where $\sigma_1$ and $\sigma_2$ are the two input states and $\sigma_3$ is the output state. We can verify that the minimum Hamiltonian configuration of $\sigma_1$, $\sigma_2$, and $\sigma_3$ for $H_2(\sigma)$ also corresponds to a two-input XOR logic. In general, an $N$-body interaction can be reduced to $k$-body interactions using $N - k$ ancillary states.[21,22]

However, not all Hamiltonians can be implemented due to the square lattice geometry restrictions. From Wagner's theorem in graph theory, a finite graph is planar if and only if its minors include neither a complete graph $K_5$ nor a bipartite graph $K_{3,3}$.[23] Equivalently, from Kuratowski's theorem, a finite graph is planar if and only if it does not contain a subgraph that is a subdivision of $K_5$ or of $K_{3,3}$.[24] Since all Ising Hamiltonians defined on a square lattice can be

expressed as a planar graph with states as vertices and interactions as edges, arbitrary Hamiltonians that contain $K_5$ or $K_{3,3}$ cannot be implemented on a square lattice post template.

Now, we introduce an example of a combinatorial optimization problem that can be mapped to the Ising lattice, assuming the aforementioned restrictions are resolved. In the number partitioning problem, we are given a set of $N$ positive numbers $S = \{n_1, \ldots, n_N\}$. The goal of this problem is to find a partition of this set of numbers into two disjoint subsets $S_1$ and $S_2 = S - S_1$ such that the sum of elements in both sets is the same. The corresponding Ising Hamiltonian is

$$H(\sigma) = \left( \sum_{j=1}^{N} n_j \sigma_j \right)^2 = \sum_{i \neq j} 2 n_i n_j \sigma_i \sigma_j + \sum_j n_j^2$$

and we partition the $n_j$'s into $S_1$ and $S_2$ based on the sign of $\sigma_j$'s in the minimum Hamiltonian configuration.[19] It is clear that if $H(\sigma) = 0$, the sum of elements in $S_1$ and $S_2$ are the same. Therefore, the ground state configuration is a solution to the number partitioning problem. Moreover, in the ground state configuration, the difference between the sum of elements in $S_1$ and $S_2$ is minimized. Since $\sum_j n_j^2$ is a constant term that does not influence the minimum Hamiltonian configuration, we can implement the above Hamiltonian using $N$ states with $J_{ij} = -2 n_i n_j$ and $h_j = 0$ where the interactions between states form a complete graph $K_N$.

## 6.4 Conclusion

We implemented simple Boolean logic gates using block copolymers to demonstrate Ising-model-based computation. The logic gates were designed through trial and error using the Ising-model simulation method. We also discussed challenges and theoretical solutions for achieving systems capable of solving combinatorial optimization problems such as the travelling salesman problem.

However, even if complex optimization problems can be mapped to the Ising lattice, there are still several remaining limitations to this computation approach. For example, time-sensitive inputs, dynamic inputs, and reusable templates are difficult to achieve due to the process flow of block copolymer DSA. In the next chapter, we discuss ideas for future work on solving these issues.

# References

1. F. Barahona, *J. Phys. A: Math. Gen.* **1982**, 15, 10.
2. C. Moore, *J. Stat. Phys.* **1997**, 88, 3-4.
3. M. Van den Nest, W. Dür, H. J. Briegel, *Phys. Rev. Lett.* **2008**, 100, 11.
4. G. De las Cuevas, W. Dür, M. Van den Nest, H. J. Briegel, *J. Stat. Mech.* **2009**, 2009, P07001.
5. M. Gu, A. Perales, *Phys. Rev. E* **2012**, 86, 1.
6. J. D. Whitfield, M. Faccin, J. D. Biamonte, *Europhys. Lett.* **2012**, 99, 5.
7. S. Lloyd, B. M. Terhal, *New. J. Phys.* **2016**, 18, 023042.
8. G. De las Cuevas, T. S. Cubitt, *Science* **2016**, 351, 6278.
9. S. Bandyopadhyay, B. Das, A. E. Miller, *Nanotechnology* **1994**, 5, 2.
10. Y. Wang, M. Lieberman, *IEEE Trans. Nanotechnol.* **2004**, 3, 3.
11. P. Delaney, J. C. Greer, *Proc. Royal Soc. A* **2006**, 462, 2065.
12. M. W. Johnson, M. H. S. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk, E. M. Chapple, C. Enderud, J. P. Hilton, K. Karimi, E. Ladizinsky, N. Ladizinsky, T. Oh, I. Perminov, C. Rich, M. C. Thom, E. Tolkacheva, C. J. S. Truncik, S. Uchaikin, J. Wang, B. Wilson, G. Rose, *Nature* **2011**, 473.
13. S. A. Haque, M. Yamamoto, R. Nakatani, Y. Endo, *Sci. Technol. Adv. Mater.* **2004**, 5, 1-2.
14. A. Imre, G. Csaba, L. Ji, A. Orlov, G. H. Bernstein, W. Porod, *Science* **2006**, 311, 5758.
15. S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi, *Science* **1983**, 220, 4598.
16. J. C. Angles d'Auriac, M. Preissmann, R. Rammal, *J. Physique Lett.* **1985**, 46, 5.
17. Y. Fu, P. W. Anderson, *J. Phys. A: Math. Gen.* **1986**, 19, 9.
18. F. Barahona, M. Grötschel, M Jünger, G. Reinelt, *Oper. Res.* **1988**, 36, 3.
19. A. Lucas, *Front. Phys.* **2014**, 2, 5.
20. J.-B. Chang, H. K. Choi, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Commun.* **2014**, 5, 3305.
21. J. D. Biamonte, *Phys. Rev. A* **2008**, 77, 5.
22. R. Babbush, B. O'Gorman, A. Aspuru-Guzi, *Ann. Phys.* **2013**, 525, 10-11.
23. K. Wagner, *Math. Ann.* **1937**, 114, 1.
24. K. Kuratowski, *Fund. Math.* **1930**, 15, 1.

# Chapter 7

# Summary and future work

## 7.1 Summary

In this thesis, we investigated two major approaches to achieve computation using directed self-assembly (DSA) of block copolymers. In the first approach, we used arrays of square confinements to define a two-state system, and studied interactions between neighboring states. In the second approach, we used arrays of square lattice posts to design an Ising lattice for DSA of block copolymers, and demonstrated Ising-model-based computation. In addition to performing computation in self-assembling materials, the results from this work can be applied to fabricate circuit-relevant nanoscale patterns for lithography applications since block copolymers naturally act as a lithography material.

In Chapter 2 and Chapter 3, we studied DSA of block copolymer thin films within templates of different polygonal shapes. We observed formation of ladder-shaped block copolymer structures with two degenerate alignment orientations formed inside square confinement. We studied properties of the two-state system such as distribution, nearest-neighbor correlation, and defect tolerance. We controlled the binary states by changing the confinement geometry, placing lithographic guiding patterns inside the confinement, and placing openings around the confinement, and reproduced the experimental results using self-consistent fiend theory (SCFT)

simulations. Using the binary state control methods, we fabricated a larger binary state array with individual state control. The resulting line segments, bends, and T-junctions formed inside the ladder-shaped structures may be useful as circuit-relevant geometries. Also, arrays of binary states could potentially act as a physical read-only memory. Although we tried to induce nearest-neighbor interactions by increasing the opening size, we were not able to maintain the defect tolerance of the two-state system for larger opening sizes.

In Chapter 4 and Chapter 5, we designed a different two-state system using a square lattice post template. We defined the two binary states based on whether two adjacent posts were connected by a block copolymer structure or not, and mapped a square lattice Ising model with nearest-neighbor interactions. We measured properties of the two-state system such as distribution and nearest-neighbor correlation. Relative magnitude between the two Ising model parameters $J$ and $h$ were determined based on the distribution of the states given their nearest-neighbor configuration. We developed a simulated annealing algorithm to calculate the minimum Hamiltonian configuration given a boundary condition, and validated our Ising lattice setup by comparing the Ising model simulation results with previously reported experimental and SCFT simulation results. The Ising model simulation method may have potential applications in template design as well as block copolymer pattern prediction. Moreover, useful computational functions can be mapped to the Ising lattice, allowing Ising-model-based computation in block copolymer systems.

In Chapter 6, we implemented simple Boolean logic gates as a proof-of-concept demonstration of computation. Ising model simulation method was used to prototype and test various logic gate designs since it is not a trivial task to design the specific Hamiltonians that can embed desired Boolean operations. The ultimate goal of Ising-model-based computation is to map

and solve complex optimization problems through the process of self-assembly. We discussed potential challenges and remedies for achieving this goal such as controlling the Ising model parameters by changing template design, implementing neutral boundaries, and introducing ancillary states.

## 7.2 Future work

In this section, we describe ideas for future work to address limitations of our computation approach and ultimately build a practical computing system using block copolymer self-assembly.

### 7.2.1 Improving error rate

One key issue that must be addressed before implementing practical computing systems with block copolymers is the error rate, since naturally occurring thermodynamic defects will always form in the final pattern. Based on the previous graphoepitaxy results using functionalized templates,[1-3] we expect a sufficiently high yield after rigorous optimization of the annealing conditions. If the measured yield of poly(styrene-*block*-dimethylsiloxane) (PS-*b*-PDMS) is still too low, we can investigate other block copolymers with good long-range order such as poly(styrene-*block*-4-vinylpyridine) (PS-*b*-P4VP).[4-7] Ultimately, error-correcting strategies[8] will have to be devised for block copolymer computing systems. But if we are simulating physical systems, the errors may perhaps become a feature rather than a bug.

### 7.2.2 Reducing fabrication cost and increasing throughput

To reduce fabrication cost and increase throughput, roll-to-roll nanoimprint lithography may be employed in the fabrication process. An example of roll-to-roll-based in-line information

processing might include a pre-patterned tape of nanostructures, customized to provide computational inputs at Gbit/sec data rates by using multiple parallel electron- or optical "input" beams. **Figure 7.1** illustrates a schematic for the proposed fabrication process. By creating periodic square lattice posts by nanoimprint while only writing the aperiodic double posts representing the boundary conditions and inputs by electron-beam lithography, fabrication steps can be greatly reduced. The continuous tape could move through a roll-to-roll materials processing system as new templates are fabricated. Such roll-to-roll system could be integrated with a block copolymer coating/annealing unit and an output readout unit to further enhance throughput.
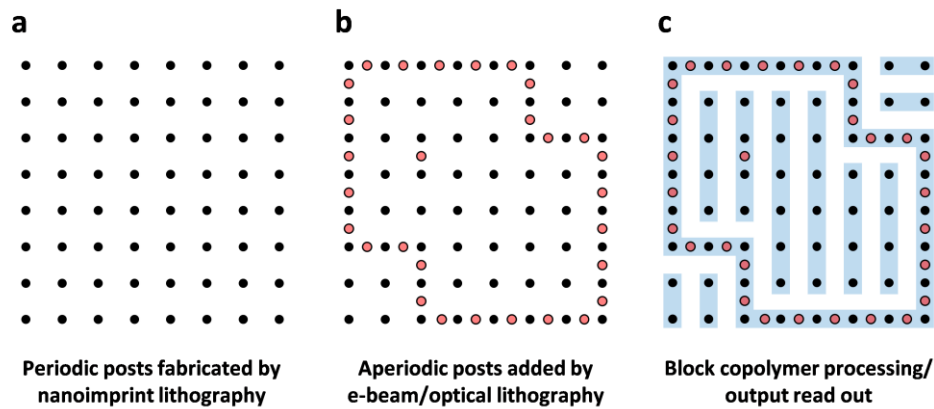


**Figure 7.1** Schematic illustration of the proposed fabrication process to reduce cost and increase throughput. (a) Periodic square lattice posts are fabricated by roll-to-roll nanoimprint lithography. (b) Aperiodic double posts which represent the boundary conditions and input states are fabricated by electron-beam or optical lithography. (c) After block copolymer processing, output states are read out. Once computation is performed, block copolymer film and aperiodic double posts can be removed by solvent rinsing and selective etching, respectively.

## 7.2.3  Implementing programmable inputs and input resetting

Our Ising-model-based computing systems are inherently static since the boundary conditions and inputs are lithographically patterned on a substrate. However, we believe that programmable inputs and input resetting will not be a significant issue as there are strategies for rapid fabrication of new

templates. As discussed in the previous section, both the boundary conditions and inputs may be lithographically programmed into a blank post lattice template. After computation is performed, block copolymer film can be easily rinsed off. We can then use etch selectivity between the single posts (fabricated by nanoimprint) and double posts (fabricated by electron-beam or optical lithography) to selectively remove the double posts and reset the template back to the initial periodic square lattice post array.
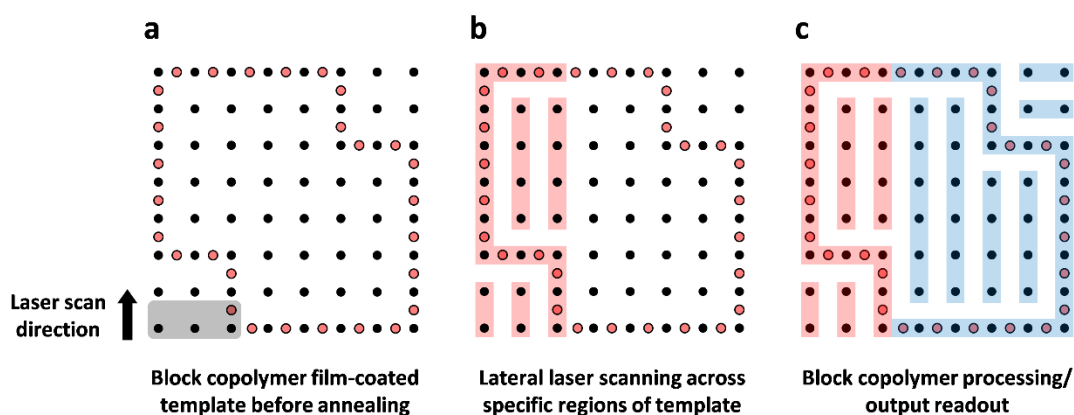


**Figure 7.2** Schematic illustration of the laser zone annealing process for achieving programmable inputs. (a) Template consisting of periodic single posts and aperiodic double posts is coated with block copolymer film. Here, only the boundary conditions are defined by the double posts. (b) When laser is scanned vertically, a zone heat moves across the scanned region and highly-ordered block copolymer structures are formed (red line). (c) Vertically aligned block copolymer structures act as a guiding pattern or "input" for the remaining region (blue line). The input can be changed simply by scanning the laser in horizontal direction.

Alternatively, programmable inputs might be implemented by using directional annealing methods such as laser zone annealing[9-12] or electric-field assisted annealing.[13-16] As shown in **Figure 7.2**, lateral laser scanning across specific regions of the template will create a moving zone heat that will anneal the block copolymer within the area to be highly ordered. The resulting aligned block copolymer structures will induce local preferential alignment, having similar templating effect as aperiodic double post structures. After programming the "inputs" by lateral

laser scanning, the whole substrate can be thermally annealed to compute the output states. By changing the laser scanning direction, input states can be effectively controlled. Directional or sequential annealing will have to be implemented for large scale Boolean logic gates.

### 7.2.4  Strategies for output readout

For readout of the output states, we can initially visually inspect the final block copolymer pattern with an integrated scanning electron microscope. Atomic force microscope may be used for minimum sample damage. Future electronic or optical readout modalities can be envisaged for integration with other devices. For materials or manufacturing applications, we can transfer the block copolymer pattern to metal by various methods such as etching, vapor deposition, ion loading, or particle decoration.[17-20] Since different bits in the output states correspond to horizontal or vertical alignment orientations, electrical or optical properties of ordered metal nanowire patterns can be measured to read out the output states.[21-24] To make the output readout easier, we can enlarge the region containing the output state using cascaded buffers as shown in **Figure 7.3**.

For example, I-V characteristics of aligned metal nanowire arrays will display large resistance anisotropy depending on measurement direction. Contact pads could be patterned along one axis of the template to determine whether the nanowire alignment orientation is along (low resistance) or orthogonal (high resistance) to the measurement direction. If a transparent substrate is used instead for optical measurement, light transmission will show optical anisotropy depending on incident direction of light. Alternatively, pattern transfer could be performed onto metal nanoparticles using selective particle decoration, and plasmonic properties of ordered nanoparticles could be measured to read out the output states.
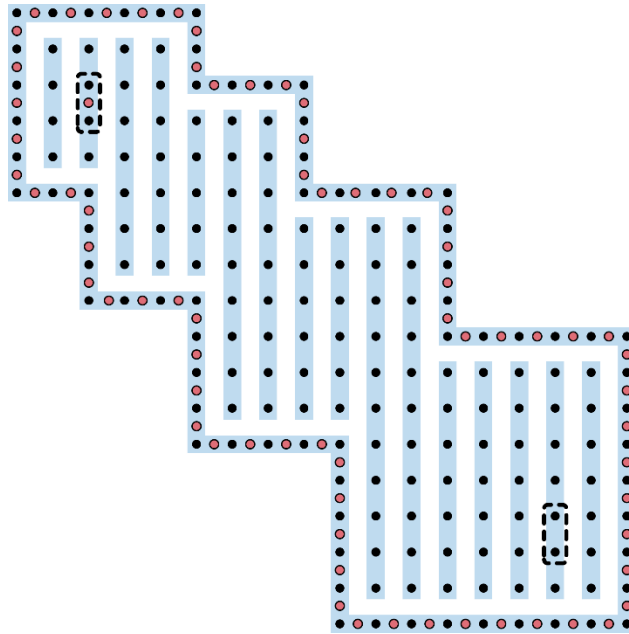
**Figure 7.3** Design for a buffer that can be used to enlarge the region containing the output state. Input and output states (black dashed box) are indicated. To make the output readout process easier, we can cascade multiple buffers with increasing boundary dimensions without affecting the output state.

## 7.3 Future vision of self-assembly-based computation

The Ising-model-based architecture for computing differs radically from the conventional electronics-based model, but we envisage this new paradigm to be scalable and to be compatible with the server- and cloud-based computing systems that are increasingly relevant today. Examples of a scalable self-assembling paradigm may use directed self-assembly of alternative nanoscale materials, and/or roll-to-roll in-line processing. Readout could be accomplished by multi-beam electron and optical microscopes.

The self-assembly architecture may also enable novel materials-based computing applications, where "smart" materials must sense and respond (by modifying their structure and function) to various environmental inputs. Computation in material systems such as block copolymers thus has promise not only for improving, but also expanding on existing computing

117

architectures. As such, we are not discouraged by the apparent differences between self-assembly-based computing and the current dominant computational paradigm. Rather, we view investing our effort in novel forms of computation as key to maintaining a vibrant range of future possibilities for computing as well as for material patterning.

# References

1. I. Bita, J. K. W. Yang, Y. Jung, C. A. Ross, E. L. Thomas, K. K. Berggren, *Science* **2008**, 321, 5891.
2. J. K. W. Yang, Y. S. Jung, J.-B. Chang, R. A. Mickiewicz, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Nanotechnol.* **2010**, 5.
3. J.-B. Chang, H. K. Choi, A. F. Hannon, A. Alexander-Katz, C. A. Ross, K. K. Berggren, *Nat. Commun.* **2014**, 5, 3305.
4. S. Park, B. Kim, J.-Y. Wang, T. P. Russell, *Adv. Mater.* **2008**, 20, 4.
5. S. Park, J. Y. Wang, B. Kim, J. Xu, T. P. Russell, *ACS Nano* **2008**, 2, 4.
6. B. Nandan, E. B. Gowd, N. C. Bigall, A. Eychmüller, P. Formanek, P. Simon, M. Stamm, *Adv. Funct. Mater.* **2009**, 19, 17.
7. X. Zhang, K. D. Harris, N. L. Y. Wu, J. N. Murphy, J. M. Buriak, *ACS Nano* **2010**, 4, 11.
8. I. Kanter, D. Saad, *Phys. Rev. Lett.* **1999**, 83, 13.
9. P. W. Majewski, K. G. Yager, *ACS Nano* **2015**, 9, 4.
10. P. W. Majewski, K. G. Yager, *Macromolecules* **2015**, 48, 13.
11. P. W. Majewski, K. G. Yager, *Nano Lett.* **2015**, 15, 8.
12. P. W. Majewski, K. G. Yager, *Soft Matter* **2016**, 12, 1.
13. A. Böker, H. Elbs, H. Hänsel, A. Knoll, S. Ludwigs, H. Zettl, A. V. Zvelindovsky, G. J. A. Sevink, V. Urban, V. Abetz, A. H. E. Müller, G. Krausch, *Macromolecules* **2003**, 36, 21.
14. V. Olszowka, M. Hund, V. Kuntermann, S. Scherdel, L. Tsarkova, A. Böker, *ACS Nano* **2009**, 3, 5.
15. M.-L. Wu, D. Wang, L.-J. Wan, *Nat. Commun.* **2016**, 7, 10752.
16. C. C. Kathrein, W. Bai, A. Nunns, J. Gwyther, I. Manners, A. Böker, L. Tsarkova, C. A. Ross, *Soft Matter* **2016**, 12.
17. Q. Peng, Y.-C. Tseng, S. B. Darling, J. W. Elam, *Adv. Mater.* **2010**, 22, 45.
18. Q. Peng, Y.-C. Tseng, S. B. Darling, J. W. Elam, *ACS Nano* **2011**, 5, 6.
19. Y.-C. Tseng, Q. Peng, L. E. Ocola, J. W. Elam, S. B. Darling, *J. Phys. Chem. C* **2011**, 115, 36.
20. M. Ramanathan, Y.-C. Tseng, K. Ariga, S. B. Darling, *J. Mater. Chem. C* **2013**, 1, 11.
21. J. Chai, D. Wang, X. Fan, J. M. Buriak, *Nat. Nanotechnol.* **2007**, 2.
22. S.-J. Jeong, H.-S. Moon, J. Shin, B. H. Kim, D. O. Shin, J. Y. Kim, Y.-H. Lee, J. U. Kim, S. O. Kim, *Nano. Lett.* **2010**, 10, 9.
23. D. O. Shin, J. H. Mun, G.-T. Hwang, J. M. Yoon, J. Y. Kim, J. M. Yun, Y.-B. Yang, Y. Oh, J. Y. Lee, J. Shin, K. J. Lee, S. Park, J. U. Kim, S. O. Kim, *ACS Nano* **2013**, 7, 10.
24. P. W. Majewski, A. Rahman, C. T. Black, K. G. Yager, *Nat. Commun.* **2015**, 6, 7448.

# Appendix

Here, we present our MATLAB program for performing simulated annealing. To encode boundary conditions for an $m \times n$ array of posts, we defined a $(2m + 1) \times (2n + 1)$ array called template. Odd row index and even column index, or even row index and odd column index, correspond to location of the Ising states. The template array was first initialized to zero. When a double post was placed between two posts, the template element located between the two posts was set to 2. The six template elements located adjacent to the double post were set to 1's or −1's as described in Chapter 4. All elements of the template array located outside the double post boundary were set to −9, and were ignored after simulated annealing. **Figure A.1** shows an example of a sample post array and its corresponding template array encoding.
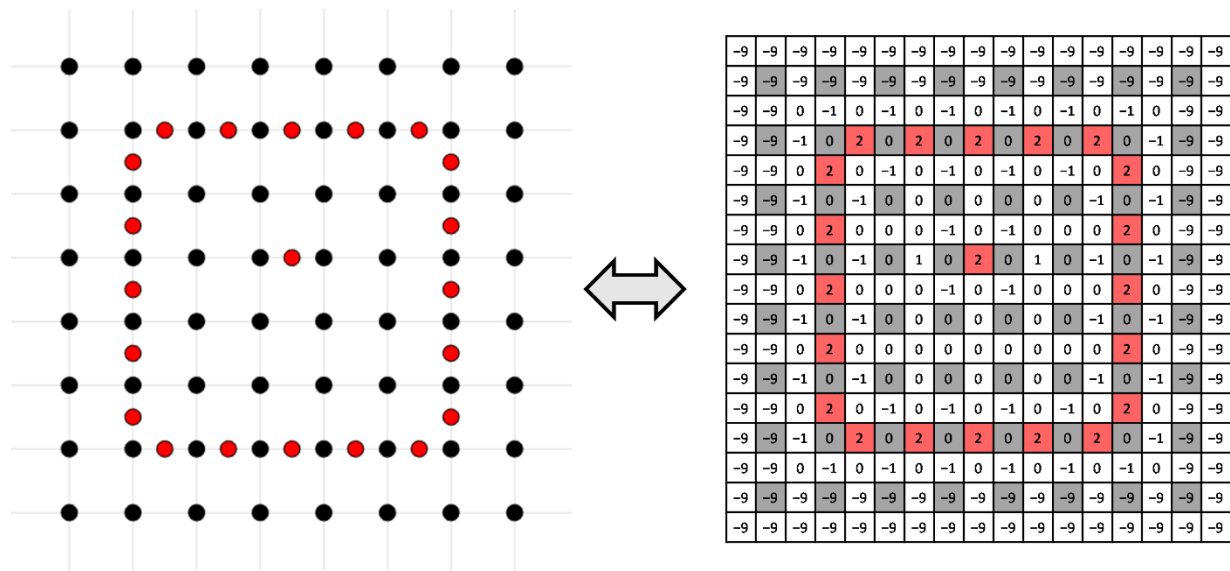


**Figure A.1** Sample post array and its corresponding template array encoding.

Next, a random initial solution was generated. Boundary states specified as 1 or 2 in the template array were fixed to +1, and boundary states specified as −1 were fixed to −1 in the random initial solution. The remaining states were randomly chosen between +1 and −1. After the minimum Hamiltonian configuration is calculated, the resulting binary state array is converted into a corresponding block copolymer pattern by determining the block copolymer morphology at each post based on the four states located adjacent to the post. Single posts were drawn, and double posts were drawn at locations where the template array element is specified as 2. The MATLAB code is presented below.

```matlab
%% Simulated annealing algorithm
clear; clc; close all; tic;

% Simulated annealing parameters
J = -1;              % Ising model J
h = -1;              % Ising model h
T =  1;              % Initial temperature
T_min = 0.01;        % Temperature threshold
alpha = 0.99;        % Scaling factor (Ti+1 = alpha*Ti)

% Template definition
template = [];       % Enter template design here
[row,col] = size(template);
n = (row*col-1)/2;   % Iterations per annealing temperature

% Generate random initial solution
state = 2*randi([0 1],row,col)-1;
for i = 1:row
    for j = 1:col
        if template(i,j) > 0
            state(i,j) = 1;
        elseif template(i,j) < 0
            state(i,j) = -1;
        end
    end
end

% Perform simulated annealing
step = 0;
while T > T_min
    step = step + 1;

    for i = 1:n
        chosen = 0;                                    % Choose a random state
```

```matlab
        while chosen == 0
            row_rnd = randi([1 row]);
            if mod(row_rnd,2) == 1
                col_rnd = 2*randi([1 floor(col/2)]);
            else
                col_rnd = 2*randi([1 ceil(col/2)])-1;
            end

            if template(row_rnd,col_rnd) == 0
                chosen = 1; end
        end

        state_ct = state(row_rnd,col_rnd);       % Center state
        state_tl = state(row_rnd-1,col_rnd-1);  % Top left neighbor
        state_bl = state(row_rnd+1,col_rnd-1);  % Bottom left neighbor
        state_tr = state(row_rnd-1,col_rnd+1);  % Top right neighbor
        state_br = state(row_rnd+1,col_rnd+1);  % Bottom right neighbor

        H1 = -state_ct*((state_tl+state_bl+state_tr+state_br)*J + h);
        H2 = -H1;                                % Calculate Hamiltonians

        if exp(-(H2-H1)/T) > rand()              % Update chosen state
            state(row_rnd,col_rnd) = -state_ct; end
    end

    T = alpha*T;                                 % Update temperature
end

% Determine block copolymer morphology at each post
morphology = zeros(floor(row/2),floor(col/2));
for i = 2:2:row
    for j= 2:2:col
        neighbor = [state(i-1,j) state(i,j+1) state(i+1,j) state(i,j-1)];
        if isequal(neighbor,[1 -1 1 -1]) == 1   % Vertical line
            morphology(i/2,j/2) = 1; end
        if isequal(neighbor,[-1 1 -1 1]) == 1   % Horizontal line
            morphology(i/2,j/2) = 2; end
        if isequal(neighbor,[1 -1 -1 -1]) == 1 % Top termination
            morphology(i/2,j/2) = 3; end
        if isequal(neighbor,[-1 1 -1 -1]) == 1 % Right termination
            morphology(i/2,j/2) = 4; end
        if isequal(neighbor,[-1 -1 1 -1]) == 1 % Bottom termination
            morphology(i/2,j/2) = 5; end
        if isequal(neighbor,[-1 -1 -1 1]) == 1 % Left termination
            morphology(i/2,j/2) = 6; end
        if isequal(neighbor,[1 1 -1 1]) == 1    % Top T-junction
            morphology(i/2,j/2) = 7; end
        if isequal(neighbor,[1 1 1 -1]) == 1    % Right T-junction
            morphology(i/2,j/2) = 8; end
        if isequal(neighbor,[-1 1 1 1]) == 1    % Bottom T-junction
            morphology(i/2,j/2) = 9; end
        if isequal(neighbor,[1 -1 1 1]) == 1    % Left T-junction
            morphology(i/2,j/2) = 10; end
        if isequal(neighbor,[1 1 -1 -1]) == 1   % Top Right bend
            morphology(i/2,j/2) = 11; end
        if isequal(neighbor,[-1 1 1 -1]) == 1   % Bottom Right bend
```

```matlab
                morphology(i/2,j/2) = 12; end
            if isequal(neighbor,[-1 -1 1 1]) == 1    % Bottom Left bend
                morphology(i/2,j/2) = 13; end
            if isequal(neighbor,[1 -1 -1 1]) == 1    % Top Left bend
                morphology(i/2,j/2) = 14; end
            if isequal(neighbor,[1 1 1 1]) == 1      % Cross
                morphology(i/2,j/2) = 15; end
            if isequal(neighbor,[-1 -1 -1 -1]) == 1 % No connection
                morphology(i/2,j/2) = 16; end
            if template(i,j) == -9                   % Do not care
                morphology(i/2,j/2) = 0; end
        end
    end
end
[row,col] = size(morphology);

% Draw block copolymer morphology
grid on; axis equal; axis([0 col+1 0 row+1]);
w = 0.25; l = 0.5; opacity = 0.4;

for i = 1:row
    for j = 1:col
        if morphology(i,j) == 1                  % Draw vertical line
            p = patch((j)+[-w w w -w],(row+1-i)+[l l -l -l],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 2                  % Draw horizontal line
            p = patch((j)+[-l l l -l],(row+1-i)+[w w -w -w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 3                  % Draw top termination
            p = patch((j)+[-w w w -w],(row+1-i)+[l l -w -w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 4                  % Draw right termination
            p = patch((j)+[-w l l -w],(row+1-i)+[w w -w -w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 5                  % Draw bottom termination
            p = patch((j)+[-w w w -w],(row+1-i)+[w w -l -l],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 6                  % Draw left termination
            p = patch((j)+[-l w w -l],(row+1-i)+[w w -w -w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 7                  % Draw top T-junction
            p = patch((j)+[-w w w l l -l -l -w],...
                (row+1-i)+[l l w w -w -w w w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 8                  % Draw right T-junction
            p = patch((j)+[-w w w l l w w -w],...
                (row+1-i)+[l l w w -w -w -l -l],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 9                  % Draw bottom T-junction
            p = patch((j)+[l l w w -w -w -l -l],...
                (row+1-i)+[w -w -w -l -l -w -w w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 10                 % Draw left T-junction
            p = patch((j)+[-w w w -w -w -l -l -w],...
                (row+1-i)+[l l -l -l -w -w w w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 11                 % Draw top right bend
```

```matlab
            p = patch((j)+[-w w w l l -w],(row+1-i)+[l l w w -w -w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 12                   % Draw bottom right bend
            p = patch((j)+[-w l l w w -w],(row+1-i)+[w w -w -w -l -l],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 13                   % Draw bottom left bend
            p = patch((j)+[-l w w -w -w -l],(row+1-i)+[w w -l -l -w -w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 14                   % Draw top left bend
            p = patch((j)+[-w w w -l -l -w],(row+1-i)+[l l -w -w w w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 15                   % Draw cross
            p = patch((j)+[-w w w l l w w -w -w -l -l -w],...
                (row+1-i)+[l l w w -w -w -l -l -w -w w w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end
        if morphology(i,j) == 16                   % Draw no connection
            p = patch((j)+[-w w w -w],(row+1-i)+[w w -w -w],'b');
            set(p,'EdgeColor','none'); set(p,'FaceAlpha',opacity); end

        h = rectangle('Position',[j-w/2 (row+1-i)-w/2 w w],...
            'Curvature',[1,1],'FaceColor','k'); % Draw single post
        if template(2*i+1,2*j) == 2                % Draw vertical double post
            h = rectangle('Position',[j-w/2 (row+1-(i+1/2))-w/2 w w],...
                'Curvature',[1,1],'FaceColor','r'); end
        if template(2*i,2*j+1) == 2                % Draw horizontal double post
            h = rectangle('Position',[(j+1/2)-w/2 (row+1-i)-w/2 w w],...
                'Curvature',[1,1],'FaceColor','r'); end
    end
end

toc;
```