Unit 9: Design for Manufacturability

• Course contents:
  – Lithography basics
  – Double/multiple patterning
  – Extreme ultraviolet (EUV)
  – Electron beam (e-beam)
  – Directed self-assembly (DSA)

• Readings
  – W&C&C: Chapter 12
Collaboration for Nanometer Design

- Crucial to have 3-way close collaboration among process, design, and EDA to achieve high-quality nanometer circuit design
Outline

- Lithography Basics
- Multiple Patterning
- Extreme Ultraviolet (EUV)
- Electron Beam (e-beam)
- Directed Self-Assembly (DSA)
- Future Research Directions
Optical Lithography System

- Patterns on a mask are transferred onto a wafer

\[ R = \frac{k_1 \lambda}{NA} \]

\( R: \) resolution; \( k_1: \) resolution constant \( (\geq 0.25); \) \( \lambda: \) wavelength

\( NA: \) numerical aperture \( = f(\text{lens, refractive index}) \)

\[ 0.25 \times 193 \text{ nm} / 1.13 = 42 \text{ nm} \]
Sub-Wavelength Lithography Gap

- **Sub-wavelength lithography**: use light of larger wavelength (193nm) to print features of smaller sizes

[S. Borkar, MICRO’04]  
EUV (13.5nm)?  
e-beam?
Sub-wavelength Lithography Problem

Desired pattern

227nm @ 0.85NA

114nm

136nm

91nm

68nm

[Numerical Technologies]
Most Expected Patterning Technologies

- Electron beam lithography (EBL)
- Extreme ultraviolet lithography (EUVL)
- Multiple patterning lithography (MPL)
- Directed Self-Assembly (DSA)

Each technology has different difficulties and requires solutions for a breakthrough.
Technology Roadmap

Source: S. Segars @ 2014 Kaufman Award presentation (with revision by Y.-W. Chang)
Outline

Lithography Basics

Multiple Patterning

Extreme Ultraviolet (EUV)

Electron Beam (e-beam)

Directed Self-Assembly (DSA)

Future Research Directions
Litho-Etch-Litho-Etch (LELE) Double Patterning

- **Pro:** Simpler layout decomposition into masks
- **Con:** overlay error with misalignment between masks

![Diagram showing the LELE process with two masks and two exposure-etching processes.](image-url)
Major LELE Patterning Challenges

- Layout decomposition (LD): graph 2-coloring problem
  - Not always feasible to decompose a 2D layout
  - Some conflicts can be resolved with stitch insertion

- Stitch minimization
  - Stitches may cause yield loss, due to overlay error
Self-Aligned Double Patterning (SADP)

- Becomes more popular due to its better overlay and critical dimension (CD) controllability

- Processes in SADP
  - Positive tone: spacers define lines
  - Negative tone: spacers define trenches
SADP Challenges: Layout Decomposition

- SADP Layout decomposition (LD) is not intuitive
  - Positive tone: spacers define patterns
  - Negative tone: spacers define spacing among patterns
2D SADP LD Is Harder

- Negative tone 2D LD
  - Patterns are formed by either core patterns or assist core patterns
  - Patterns with arbitrary spacing values may be distorted after LD
LD in Triple Patterning Lithography (TPL)

- TPL is required for more complex layouts
- TPL LD is modeled as a 3-coloring problem
  - Determine if a graph is 3-colorable is NP-complete
  - TPL LD is NP-complete even for a 3-colorable, planar graph
Stitch Finding in TPL

- Yu et al. [ICCAD’11] directly applied the projection method used in DPL to find stitch candidates.
- We observed that the projection method may miss legal stitches for TPL [DAC’12, TCAD’14]
  - Some generated conflicts can be resolved with stitch insertion.
Stitch Finding in TPL

- Yu et al. [ICCAD’11] directly applied the projection method used in DPL to find stitch candidates.
- We observed that the projection method may miss legal stitches for TPL [DAC’12, TCAD’14]
  - Some conflicts can be resolved with stitch insertion.

![Diagram showing stitch finding in TPL with conflicts and resolution]
Layout Decomposition for TPL

S1: Conflict Graph Construction

S2: Graph Division

S3: Stitch-Aware Mask Assignment

S4: Stitch Insertion
S1: Conflict Graph Construction

- A conflict graph $G=(V,E)$ is constructed
  - A node: a pattern
  - An edge: the distance between the two patterns $< \min_{CS}$

**Huge graph size?**
Divide and Conquer! Graph Division!!
S2: Graph Division Techniques

[ICCAD'11]

1. Connected Component Computation
2. Vertex with Degree Less than Three Removal
3. 2-Edge-Connected Component Computation

Ours

4. 2-Connected Component Computation
5. 3-Edge-Connected Component Computation
S2: Graph Division Techniques

[ICCAD’11]

1. Connected Component Computation
2. Nodes with Degree Less than Three Removal
3. 2-Edge-Connected Component Computation

Ours

4. 2-Connected Component Computation
5. 3-Edge-Connected Component Computation
S2: Graph Division Techniques

[ICCAD'11]

1. Connected Component Computation

2. Vertex with Degree Less than Three Removal

3. 2-Edge-Connected Component Computation

4. 2-Connected Component Computation

5. 3-Edge-Connected Component Computation

Ours

2-edge-connected components

2-connected components

cut edge
S2: Graph Division Techniques

[ICCAD’11]

1. Connected Component Computation
2. Vertex with Degree Less than Three Removal
3. 2-Edge-Connected Component Computation

Ours

4. 2-Connected Component Computation
5. 3-Edge-Connected Component Computation

2-connected components
S2: Graph Division Techniques

1. Connected Component Computation
2. Vertex with Degree Less than Three Removal
3. 2-Edge-Connected Component Computation

[ICCAD’11]

4. 2-Connected Component Computation
5. 3-Edge-Connected Component Computation

Ours

3-edge-connected components
S3: Stitch-Aware Mask Assignment

- Find a mask assignment such that the conflicts are more likely to be resolved with stitch insertion
  - Construct 3 color classes (independent sets) to minimize the total edge weight of the residual graph
    - weight: difficulty of resolving a conflict by stitch insertion
  - Assign each remaining node to a color class such that the corresponding conflict edge has the smallest edge weight

![Diagram of graph with nodes a, b, c, d, e, f, g and colored classes]

- Weight = 1
- Weight = 2
- Color class 1
- Color class 2
- Color class 3
Edge Weight Computation

- Weight of a conflict edge reflects how hard the conflict can be resolved by stitch insertion
  - The more projections overlap on a pattern, the more difficult the conflict can be resolved with stitch insertion
  - Edge weight: maximum density of other overlapped projections

\[ \text{Density}=1 \]
Edge Weight Computation

Density=1

Density=2

No feasible stitch location

Stitch
S4: Stitch Insertion

- Solve conflicts with stitch insertion
  - Partition a target pattern into segments with available colors
  - Apply a plane sweep method to solve a conflict with fewer stitches
Outline

- Lithography Basics
- Multiple Patterning
- Extreme Ultraviolet (EUV)
- Electron Beam (e-beam)
- Directed Self-Assembly (DSA)
- Future Research Directions
Extreme Ultraviolet Lithography (EUVL)

- EUVL is the most invested next-generation lithography technology
  - Its wavelength is only 13.5 nm
  - Reflective optical components and masks are used
Flare in EUVL

- Reflective masks are used
- **Flare**: scattered light due to the surface roughness of the optical system
- **Clear field mask**: layout patterns are formed by absorbers
- Flare will be distributed from vacant regions and reduce image contrast (and thus pattern quality)
Flare Level and Flare Variation

- High flare level in EUVL [Myers et al. 2008]

\[ \text{flare} \propto \frac{\text{surface roughness}}{\text{wavelength}^2} \]

- Large flare variation
  - Non-uniformity of layout patterns
  - **Flare periphery effect**: regions at the periphery of a chip have much less flare

- Both the high flare level and the large flare variation could damage the pattern quality
Flare Mitigation with Dummification [DAC’12]

- Flare can be mitigated with dummification (dummy fill)
  - Existing dummy fill algorithms are for CMP optimization to maximize layout uniformity
  - Need to consider global flare distribution for flare mitigation
Flare Minimization with Dummification

• Input
  – A grid-based layout
  – Maximum available dummy density of each grid

• Objective
  – Assign a dummy value to each grid to simultaneously minimize the flare level and the flare variation

• Output
  – Inserted dummies in each grid without conflicting to the original design

1st work on flare minimization with dummification [DAC’12, DAC’14]
Dummification Algorithm

Start

Input Layout

Density map generation

Flare map computation

Global Dummification

Top-down framework

Dummy demand map generation

Dummy value assignment

Flare map computation

Local Refinement

Dummy assignment

Flare map computation

Flare variation decreases?

yes

no

Output layout

Finish
Flare Map Computation

- EUVL Flare can be modeled as a scattering point spread function (PSF)
- A flare map can be computed as

\[ I_F(x, y) = I_D(x, y) \otimes PSF(x, y) \]

- Full-chip flare map computation is time-consuming
Estimated Flare Map Computation

- Apply multiple convolutions with coarsened grids of different sizes to speed up the computation
  - A sub-PSF with larger variation → use smaller grid size
  - A sub-PSF with smaller variation → use larger grid size

\[ I_D(x, y) \otimes PSF(x, y) = I_F(x, y) \]
Dummy Demand Map Generation

- A flare reduction map due to dummy fill
  \[ I_F(x, y) - I'_F(x, y) = I_{dummy}(x, y) \otimes PSF(x, y) \]

- A dummy demand map is generated by
  \[ D(x, y) = I_F(x, y) \otimes Q(x, y) \]

\[ Q(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} PSF(x - f, y - g)PSF(x, y)dfdg \]

Insert dummies guided by the dummy demand map
Local Refinement

- Further minimize the flare variation
  - Identify grids with excessive flare values
  - Propagate flare reduction demand of a grid to the refinement demand of neighboring grids
  - Assign available dummy values for those grids with larger demands
Outline

1. Lithography Basics
2. Multiple Patterning
3. Extreme Ultraviolet (EUV)
4. Electron Beam (e-beam)
5. Directed Self-Assembly (DSA)
6. Future Research Directions
Electron Beam Lithography (EBL)

- EBL is a maskless next-generation lithography technology
  - Maskless: no more diffraction limitation of light
  - Can define very fine patterns
Subfield Scheduling in EBL

- **Contiguously sequential writing**: writing proceeds in the order from one subfield to the next adjacent subfield
  - High voltage beams with contiguously sequential writing cause resist heating effects and critical dimension (CD) distortion

- **Subfield scheduling**: writing proceeds in a non-contiguously sequential way to avoid successive writing
  [Babin et al., SPIE’03; Fang et al., ISPD’12, TCAD’14]

![Diagram of contiguously sequential writing and subfield scheduling](image)
Effect of Subfield Scheduling

- Line-end roughness is improved (smaller) with subfield scheduling (non-contiguously-sequential writing)

Machine: ELS7500
Voltage: 50keV
Current: 600pA
Photoresist: ZEP520A
Subfield Scheduling Problem

- Subfield scheduling: find a writing order such that the distance between two successive subfields is maximized.
- Resort to the maximum scatter traveling salesman problem (MSTSP)!!
  - Find a Hamiltonian path that is most scattered
  - The general MSTSP is NP-complete

Traditional TSP
minimize total distance

MSTSP
maximize the shortest edge
Writing a subfield will raise the temperature of neighboring subfields

**Blocked subfield**: subfield with temperature higher than a threshold value
- Should not be written in the following writing process before the temperature drops below the threshold
Problem Formulation

- Given: A mask layout and the pre-defined size of a subfield
- Objective: An MSTSP solution where the length of the shortest edge in the path is maximized
- Constraint: Blocked box constraint
Our Algorithm

- Constrained MSTSP is NP-complete
- To tackle its high complexity and the additional blocked box constraint:
  
  ◆ Develop a linear-time optimal algorithm for a special case of the MSTSP
  
  ◆ Extract sub-problems:
    1. Conform to the special case
    2. Satisfy the blocked box constraint
  
  ◆ Solve each sub-problem independently
  ◆ Merge sub-solutions into the final one

Can be a 2-approximation algorithm
A Special MSTSP

• Special case:
  - Points are on two parallel lines
  - Each line has an odd number of points
  - Points on different lines are aligned

Line a

\[ \text{Line } a \hspace{2cm} a_1 \text{ } a_2 \text{ } a_3 \text{ } a_4 \text{ } a_5 \]

Line b

\[ \text{Line } b \hspace{2cm} b_1 \text{ } b_2 \text{ } b_3 \text{ } b_4 \text{ } b_5 \]

• This special MSTSP can be optimally solved in linear time
Optimal Algorithm for the Special MSTSP

- **V**: a set of input points
  - $|V| = 2n$, $n = 2k + 1$

**Optimal algorithm for the special case**

- **Step 1**: For each point $a_h$, connect $a_h$ to $b_i$ and $b_j$, where

  $$i = \begin{cases} 
  h + (1 + k), & \text{if } h + (1 + k) \leq n \\
  h + (1 + k) - n, & \text{if } h + (1 + k) > n 
  \end{cases}$$

  and

  $$j = \begin{cases} 
  h + k, & \text{if } h + k \leq n \\
  h + k - n, & \text{if } h + k > n 
  \end{cases}$$

- **Step 2**: Delete the shortest edge in the generated cycle
Subfield Scheduling Flow

A Layout, Subfield Size

Blocked Box Generation

S1: Maximum Subfield Subsequence Construction

S2: Hamiltonian Subpath Construction

S3: Hamiltonian Subpaths Merging

Subfield Scheduling Result
S1: Maximum Subfield Subsequence Construction

- **Maximum subfield subsequence construction**: for a pair of rows, find a set of subfields that
  - Conform to the special case
  - Satisfy the blocked box constraint
  - Maximize the total area of blocked boxes
- Apply dynamic programming to find a max subfield sequence
- Apply the optimal algorithm for the special MSTSP to find the extracted sequence in linear time
S2: Hamiltonian Subpath Construction

- Find a pairing of all subfield rows to maximize the length of the shortest edge among all sub-paths

**Step 1:** Construct a compatible graph
- A node: a subfield row
- An edge: two rows without vertical blocked box overlap
- Edge weight: the length of the shortest edge in the extracted optimal Hamiltonian path of the two rows

**Step 2:** Find a maximum matching such that the smallest edge weight is maximized

Two Hamiltonian subpaths found: \( r_1 + r_3 \) and \( r_2 + r_4 \)
S3: Hamiltonian Subpaths Merging (1/2)

- Merge Hamiltonian subpaths and maximize the lengths of edges connecting Hamiltonian subpaths

- **Step 1:** Construct a compatible graph
  - A node: an endpoint of a Hamiltonian subpath
  - An edge: two endpoints belong to different paths
  - Edge weight: the distance between two endpoints
**S3: Hamiltonian Subpaths Merging (2/2)**

- **Step 2:** Merge paths by finding a maximum matching where the smallest edge weight is maximized.
- **Step 3:** Delete the smallest edge in each resulting cycle and update the compatible graph.
- Repeat Step 1 to Step 3 until all Hamiltonian subpaths are merged into one Hamiltonian path.

---

Maximum cardinality bottleneck matching → Resulting graph with two cycles → Updated compatible graph after breaking cycles
Outline

Lithography Basics

Multiple Patterning

Extreme Ultraviolet (EUV)

Electron Beam (e-beam)

Directed Self-Assembly (DSA)

Future Research Directions
Directed Self-Assembly (DSA)

- Block copolymer DSA for contact/via patterning
  - Groups of contacts/vias are patterned by guiding templates with traditional lithography
- A large template can be used to pattern multiple close contacts even for sub-7nm nodes

[Xiao, et al., DAC’14]
Different template shapes have different control on the overlay accuracy

- Feasible shapes? Costs?
DSA Cut Redistribution

Original cut distribution

4 DSA templates

Cut distribution A
Conflicts: 2, wire cost: 11

Cut distribution B
Conflicts: 0, wire cost: 4

Extended wire

Real wire

Dummy wire
Cut Redistribution for 1D Layouts

- Construct a conflict graph $G = (V,E)$
  - A node denotes a gap
  - An edge is between two gaps if their cuts conflict each other
- Partition $G$ into vertex-disjoint paths considering conflicts
- Each vertex-disjoint subproblem can be solved in linear time (dynamic programming)
Outline

Lithography Basics

Multiple Patterning

Extreme Ultraviolet (EUV)

Electron Beam (e-beam)

Directed Self-Assembly (DSA)

Future Research Directions
Future Research Directions

1. Non-stitch layout decomposition/routing for TPL, QPL, etc.
2. Positive Tone 2D layout decomposition
3. EUV/e-beam + CMP co-optimization
4. Layout perturbation for multiple e-beam writing
5. Hybrid 193nm SADP & e-beam lithography
6. DSA for close cuts and layouts
7. E-beam/EUV layout-dependent effects
8. Physical Design for MPL/EUV/e-beam/DSA
9. Many more…. 
R1: Non-stitch TPL & QPL

- Layout decomposition for TPL or QPL (3- or 4-coloring problems)

- TPL- or QPL-aware detailed routing

![Diagram showing routing with and without conflicts and stitches.](image)
R2: Positive Tone SADP 2D LD

- Positive tone 2D layout decomposition
  - Patterns are formed by spacers
  - Only for designs with uniform pattern width
  - May have better decomposability since non-uniform spacing can be controlled by core patterns

- Positive-tone SADP routing

![Target layout diagram](image-url)
R3: CMP/e-beam and EUV Co-Optimization

- Conflicting pattern density requirements between CMP planarization and EUV flare optimization [DAC’14]

**Pattern Density Distribution**
- CMP Variation = 0

**EUV Level** = 0.62
**EUV Variation** = 0.30

**Pattern Density Distribution**
- CMP Variation = 0.45

**EUV Level** = 0.22
**EUV Variation** = 0.23

**Flare Distribution**
- CMP Variation = 0
- CMP Variation = 0.45

**Convolution**

**CMP and e-beam?**
R4: Layouts for Multiple e-beam Writing

- Massively parallel e-beam writes a stripe of certain wide (i.e., 2 μm), so a die is split into stripes for parallel writing.

- Patterns cut by stitching lines (boundaries) may incur overlay error, dose variation, etc.
R5: Hybrid 193nm SADP & e-beam Lithography

- How to maximize the cuts that can be printed by 193nm and minimize the cuts for e-beam to increase the throughput? [Du, et al., ASP-DAC’12; Ding, et al., DAC’14]

- Efficient, non-ILP-based solutions for 1D layouts?
- 2D layouts?
- 193nm SADP + EUV?

[Du, et al., ASP-DAC’12]
R6: DSA for Close Cuts & Layouts

- Use DSA to solve conflicts between close cuts
- Minimize extended wire length

![Diagram showing close cuts and conflicts]

- Minimize DSA template costs

![Diagram showing different templates]

Different templates incur different costs
R7: e-beam/EUV Layout-Dependent Effects

- **E-beam**
  - Fogging effect
  - Proximity effect
  - Charge effect
  - Stencil design

- **EUV**
  - Shadowing effect
R8: Physical Design for MPL/EUV/e-beam/DSA

Placement, routing, post-layout optimization (wire sizing & perturbation), dummification for the problems listed earlier

Invited feature article: Chen & Chang, IEEE CSM, Sept. 2009
Collaboration Model Revisited

Tremendous opportunities for process/design co-optimization
Keys to Research Solutions: CAR

Criticality

Abstraction

Restriction
Thank You!!