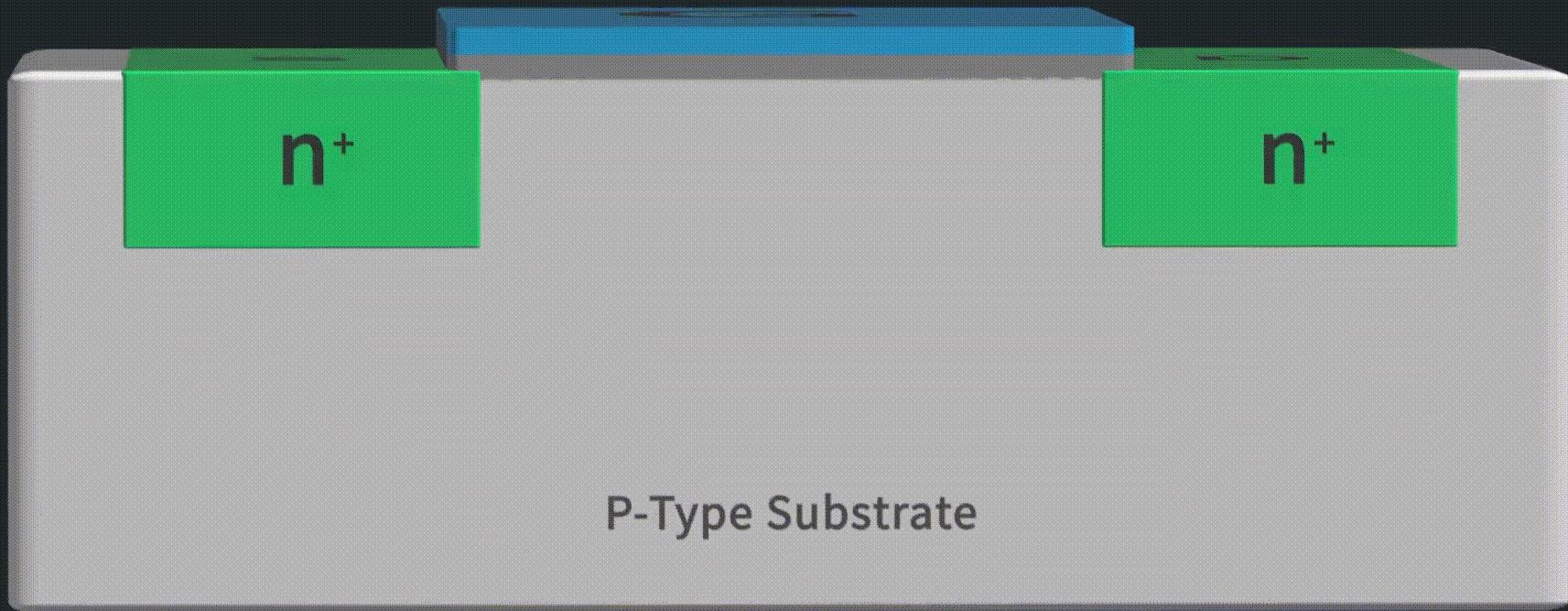


# Graph Algorithms for VLSI Power and Clock Networks

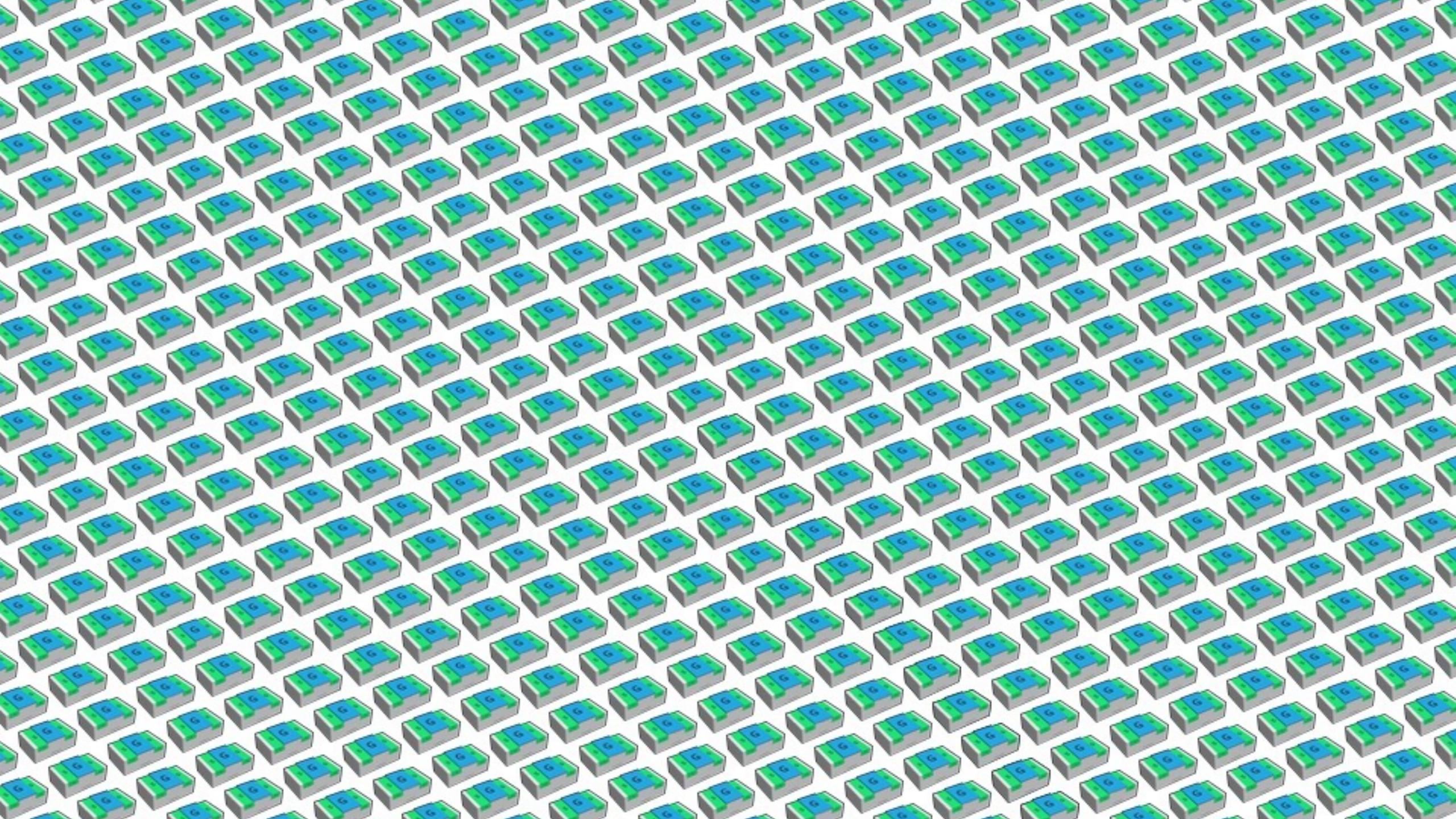
Rassul Bairamkulov

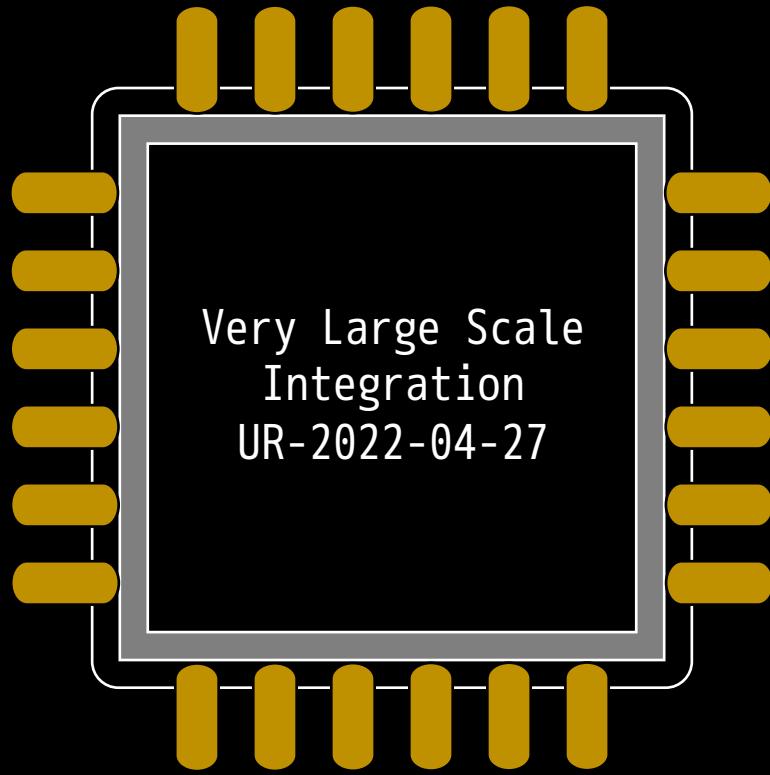
Department of Electrical and Computer Engineering

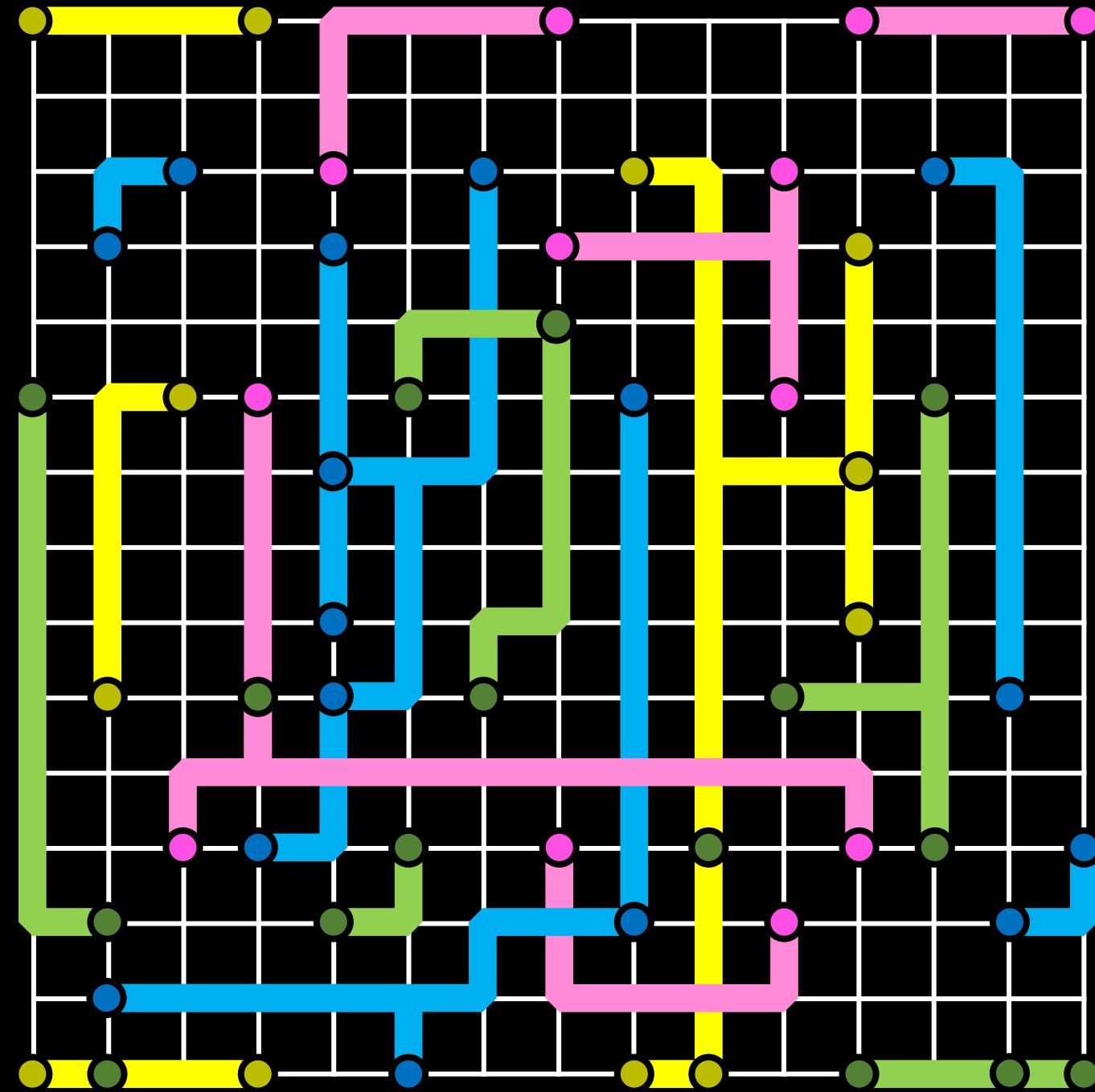
Advisor: Eby G. Friedman

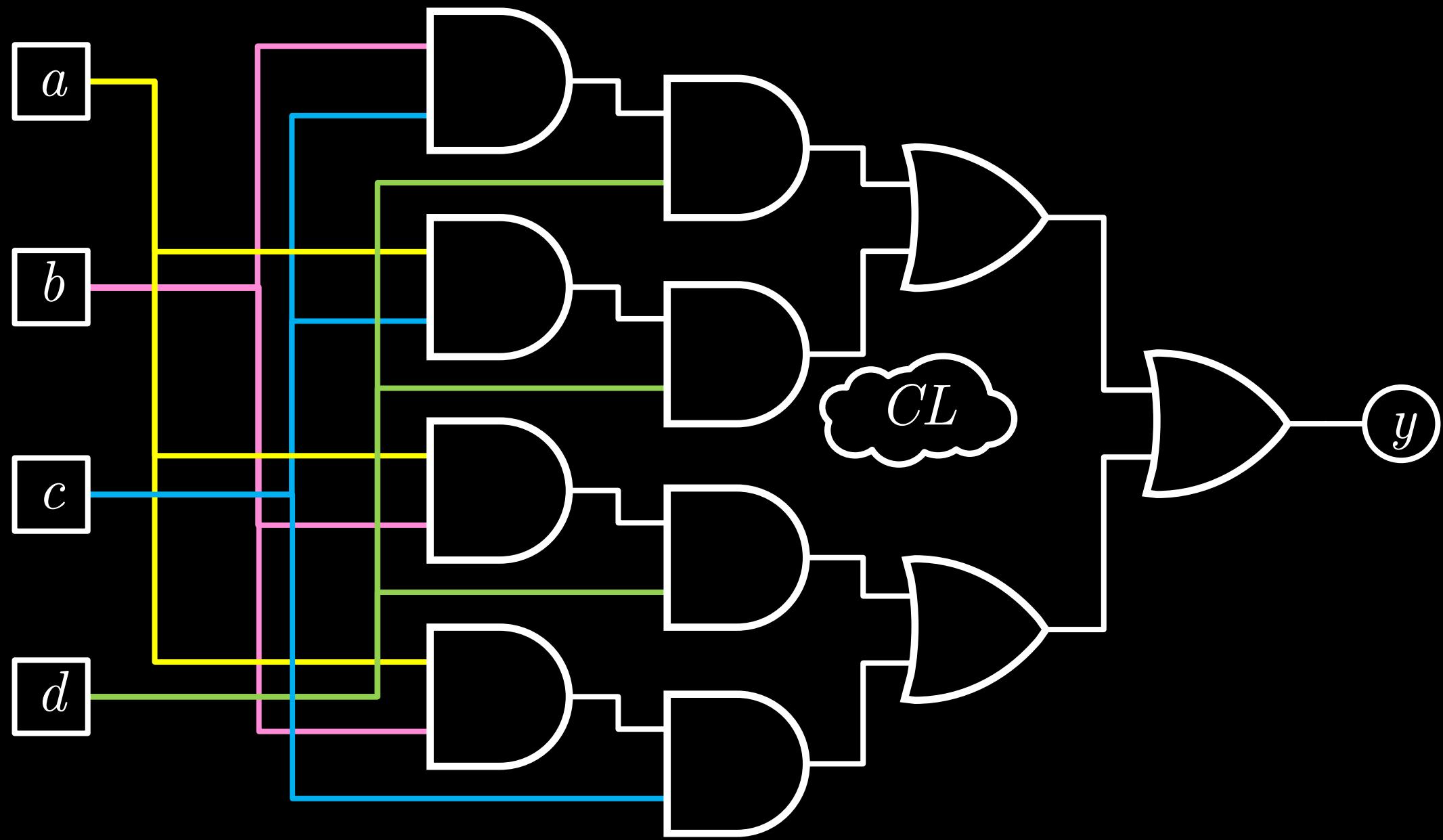


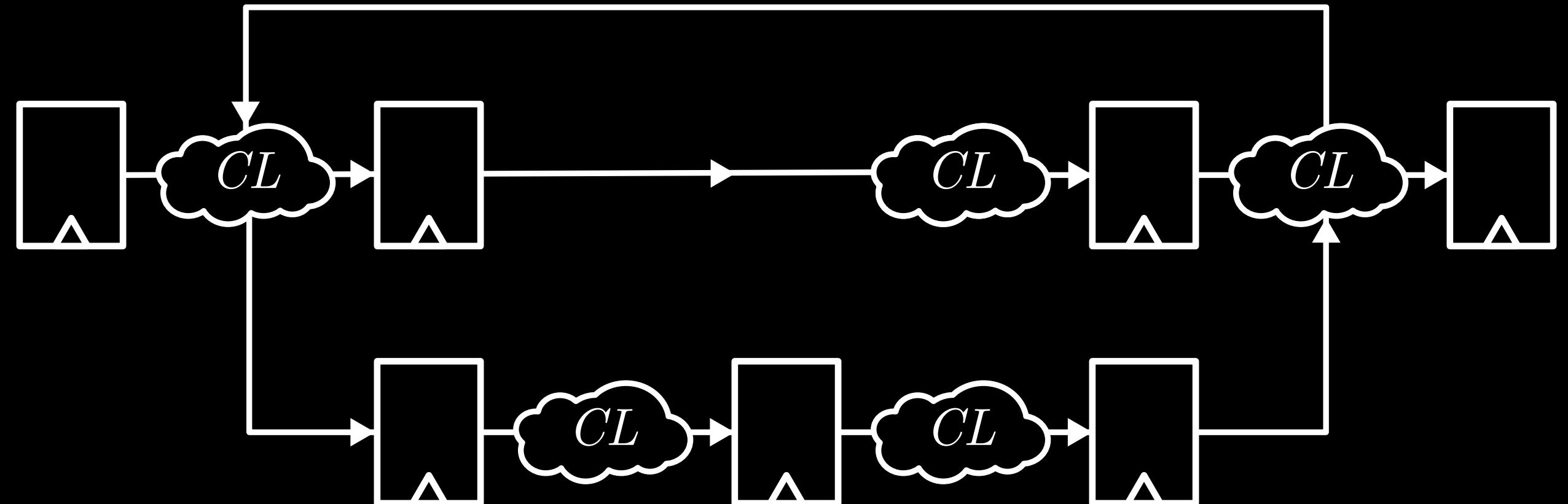


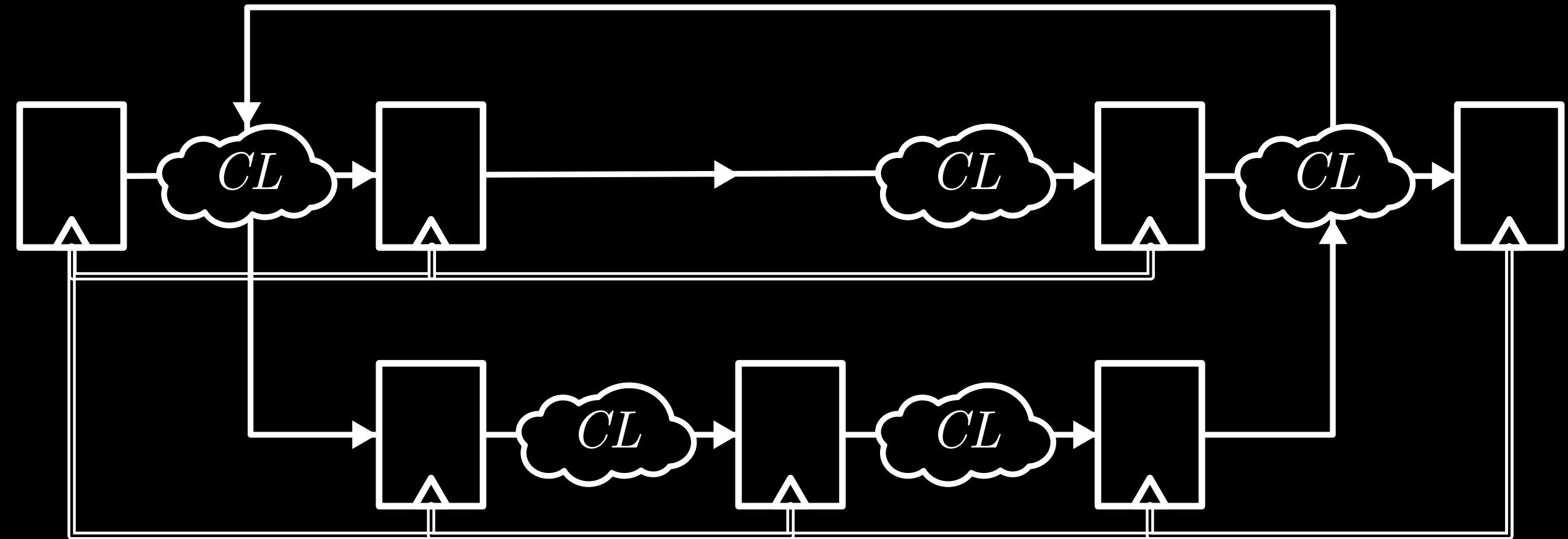


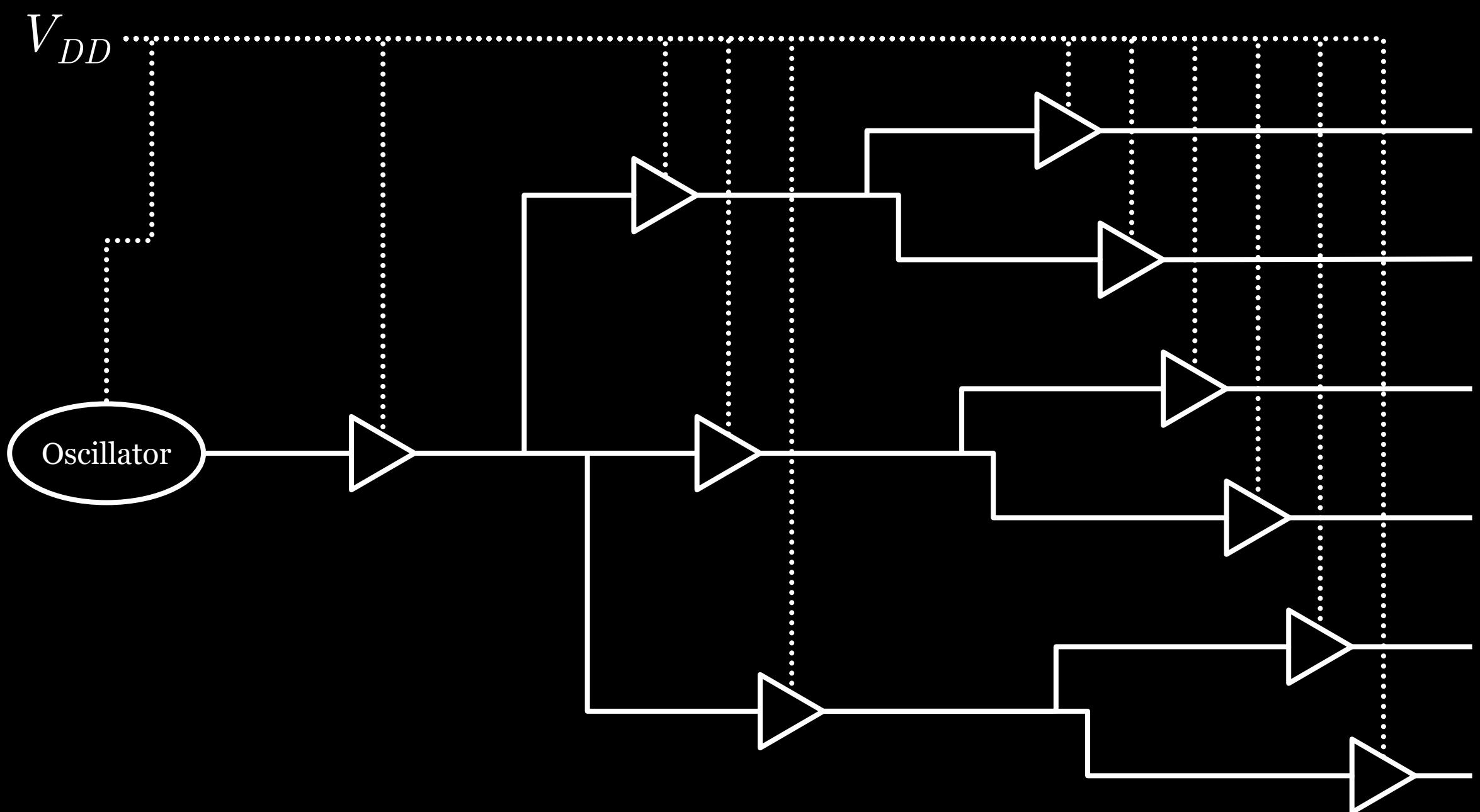


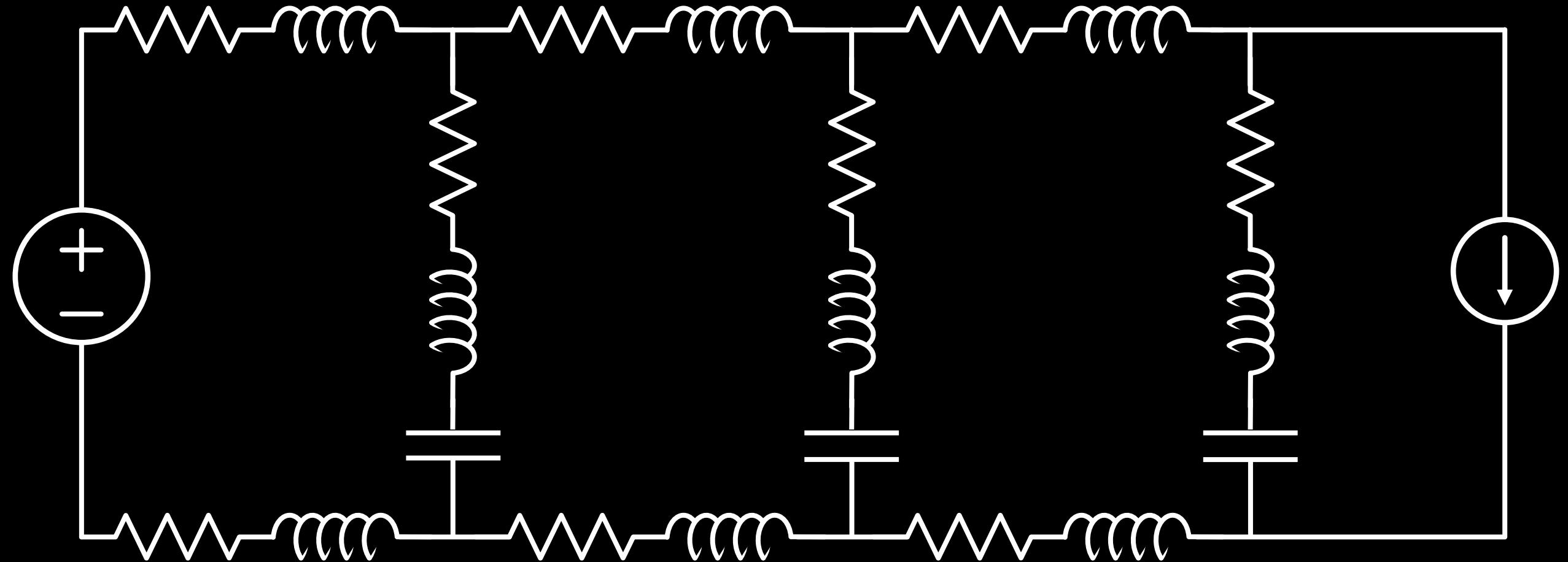


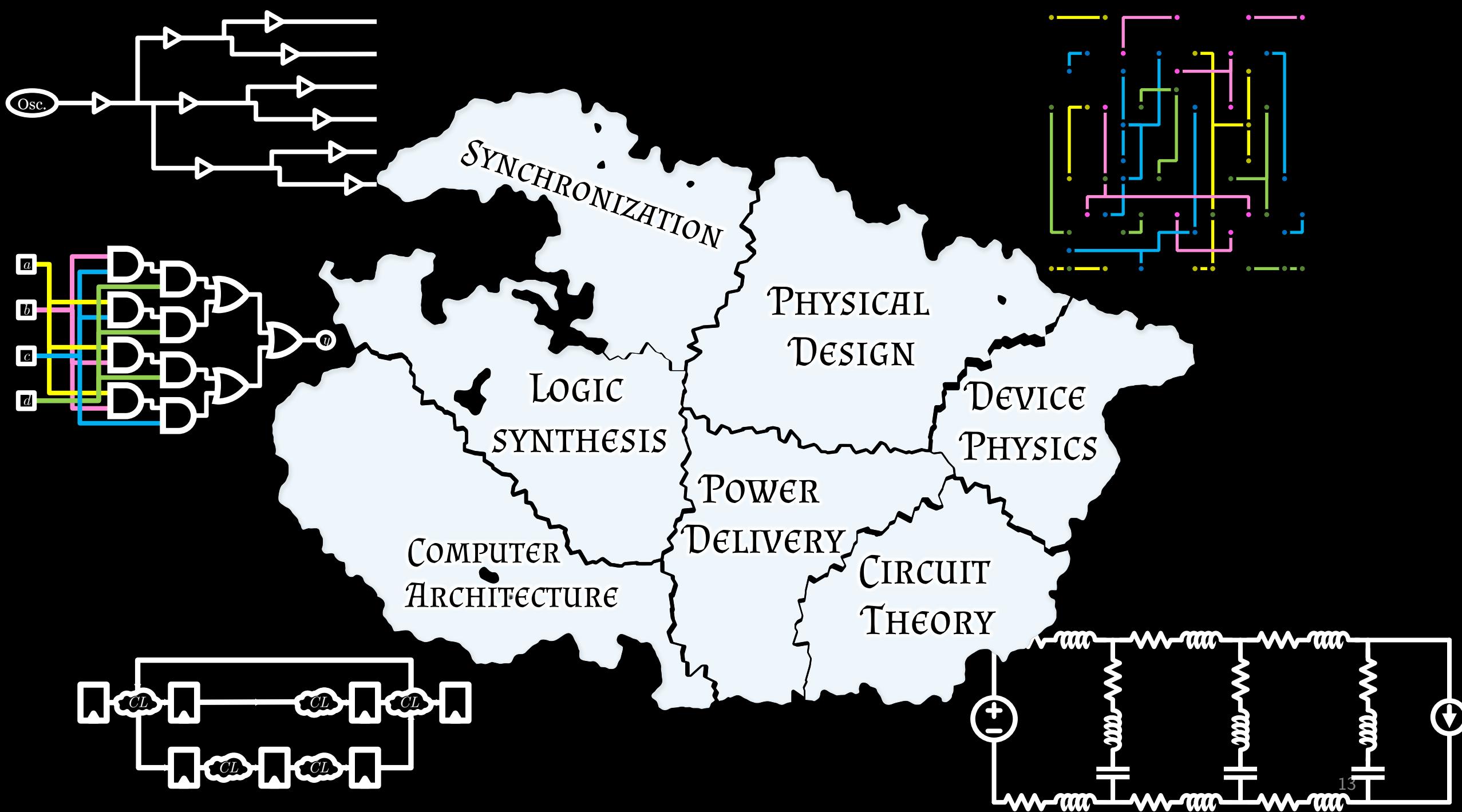




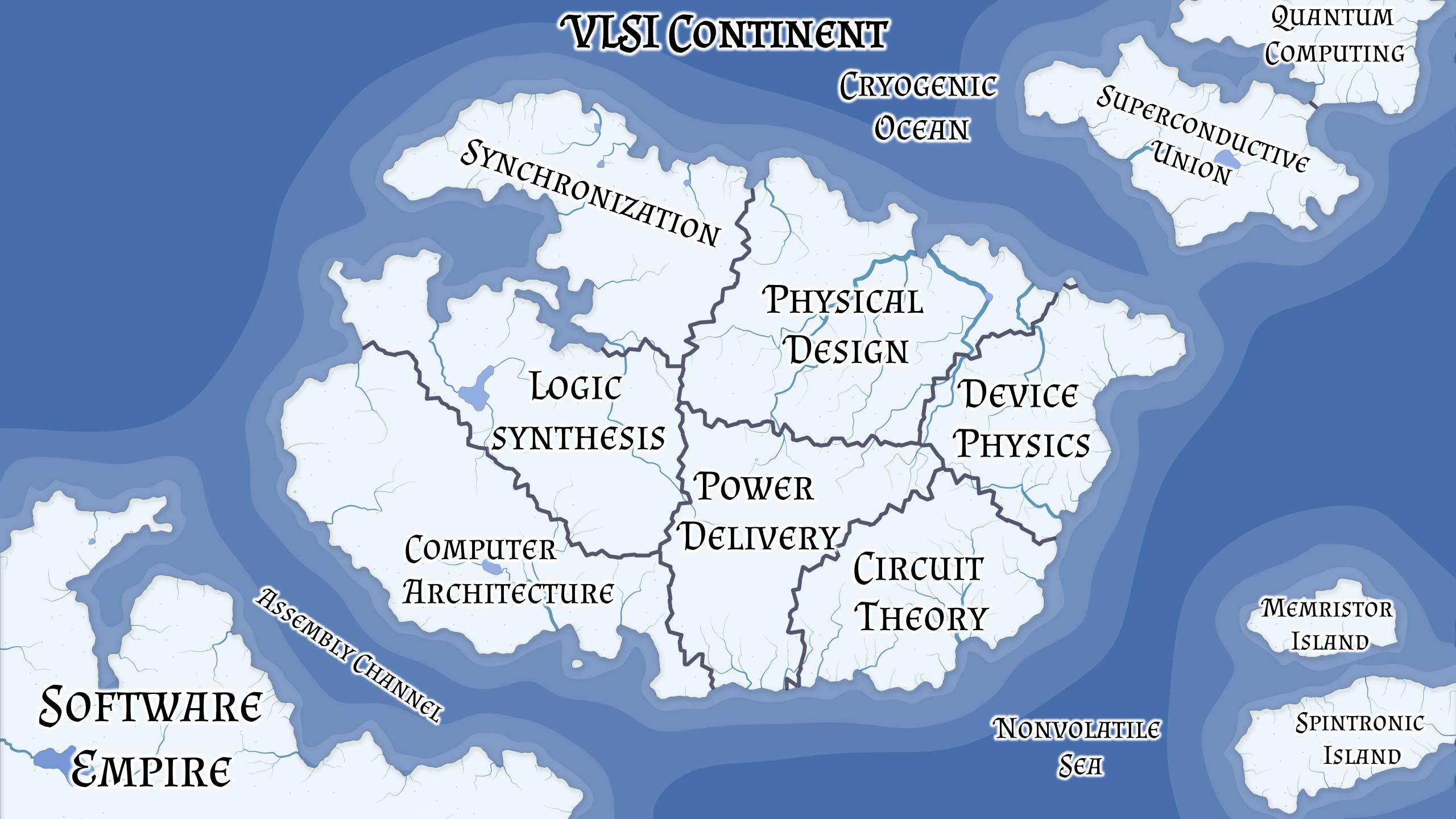








# VLSI CONTINENT



SOFTWARE  
EMPIRE

CRYOGENIC  
OCEAN

SUPERCONDUCTIVE  
UNION

MEMRISTOR  
ISLAND

NONVOLATILE  
SEA

LOGIC  
SYNTHESIS

COMPUTER  
ARCHITECTURE

SYNCHRONIZATION

POWER  
DELIVERY

CIRCUIT  
THEORY

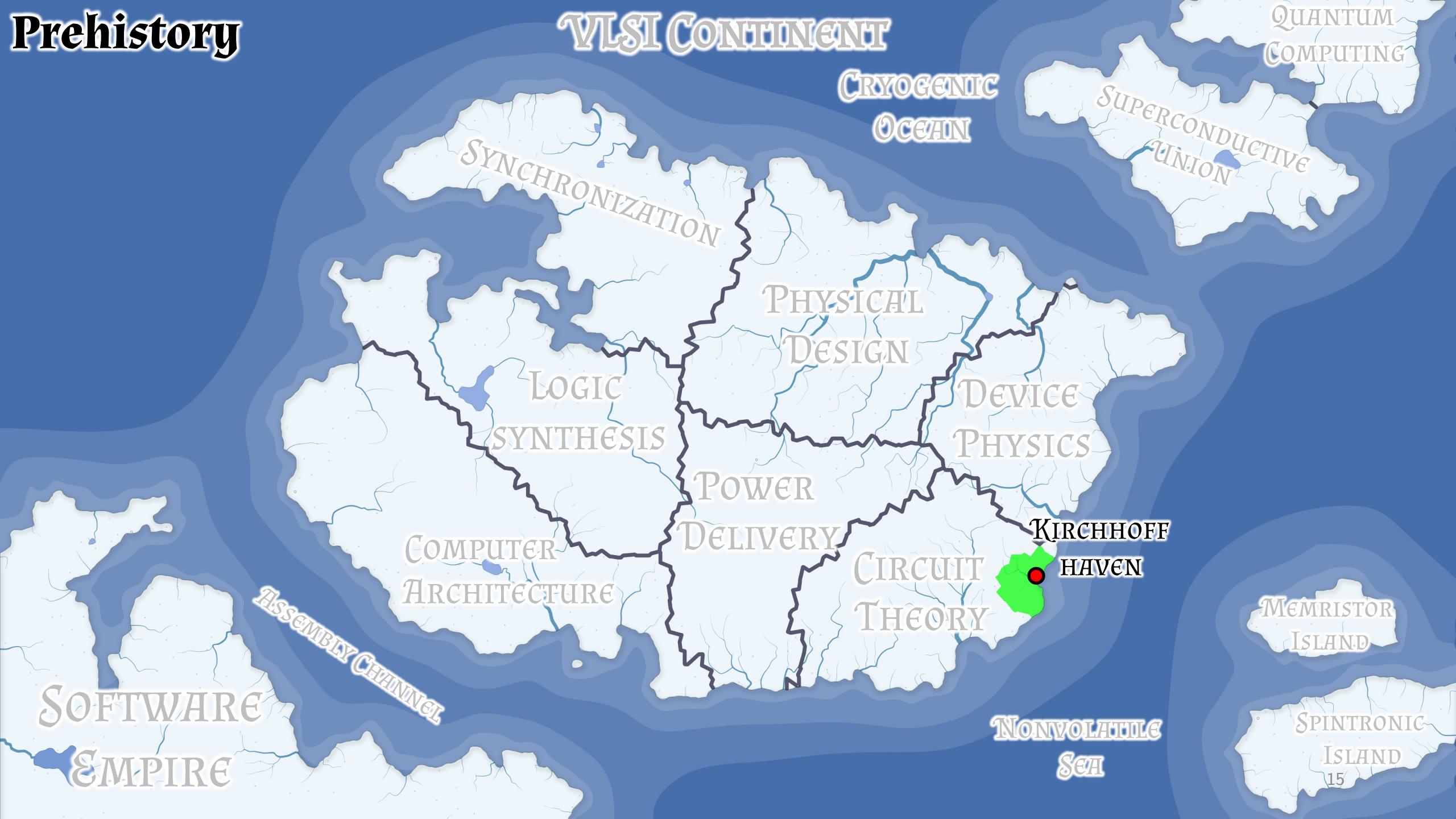
PHYSICAL  
DESIGN

DEVICE  
PHYSICS

ASSEMBLY CHANNEL

# Prehistory

## VLSI CONTINENT



NONVOLATILE  
SEA

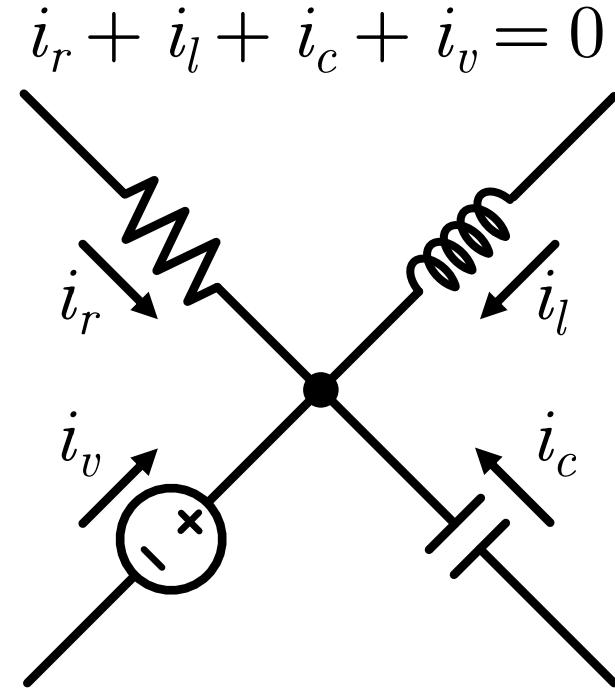
MEMRISTOR  
ISLAND

SPINTRONIC  
ISLAND  
15

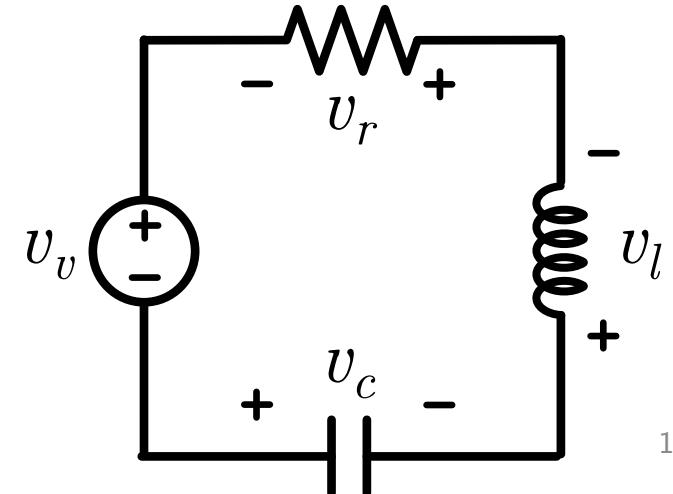
SOFTWARE  
EMPIRE

# Circuit Analysis

- In 1847, G. R. Kirchhoff introduced graph methods into circuit theory
- Postulated two fundamental laws of electrical circuits
  - Kirchhoff Current Law
  - Kirchhoff Voltage Law
- Number of independent cycles within network is
$$\mu = |E| - |V| + 1$$



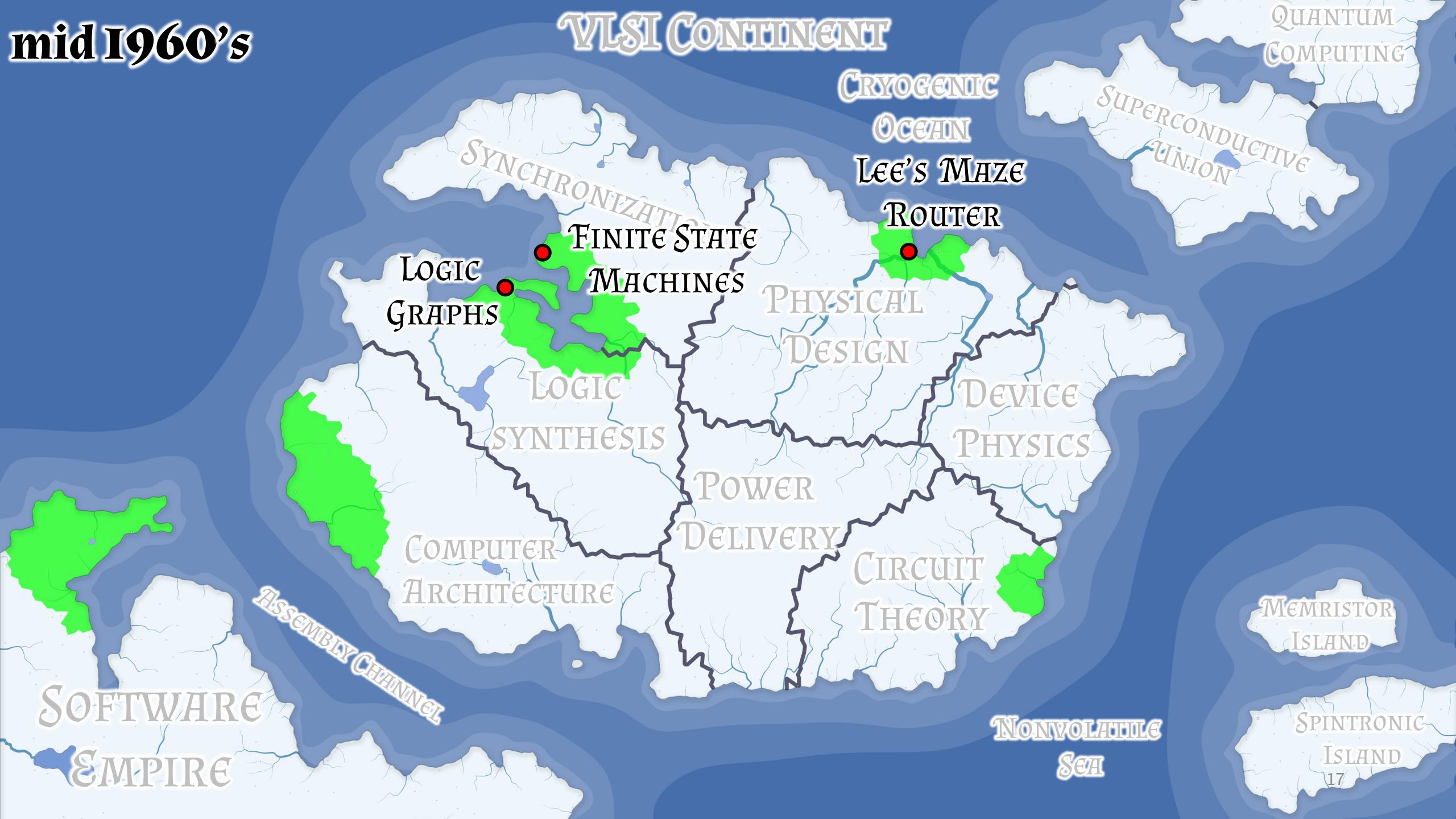
$$v_r + v_l + v_c + v_v = 0$$



G. R. Kirchhoff, "Ueber die Auflösung der Gleichungen, auf welche man bei der Untersuchung der linearen Vertheilung galvanischer Ströme geführt wird," *Annalen der Physik*, Vol. 148, No. 12, pp. 497-508, October 1847.

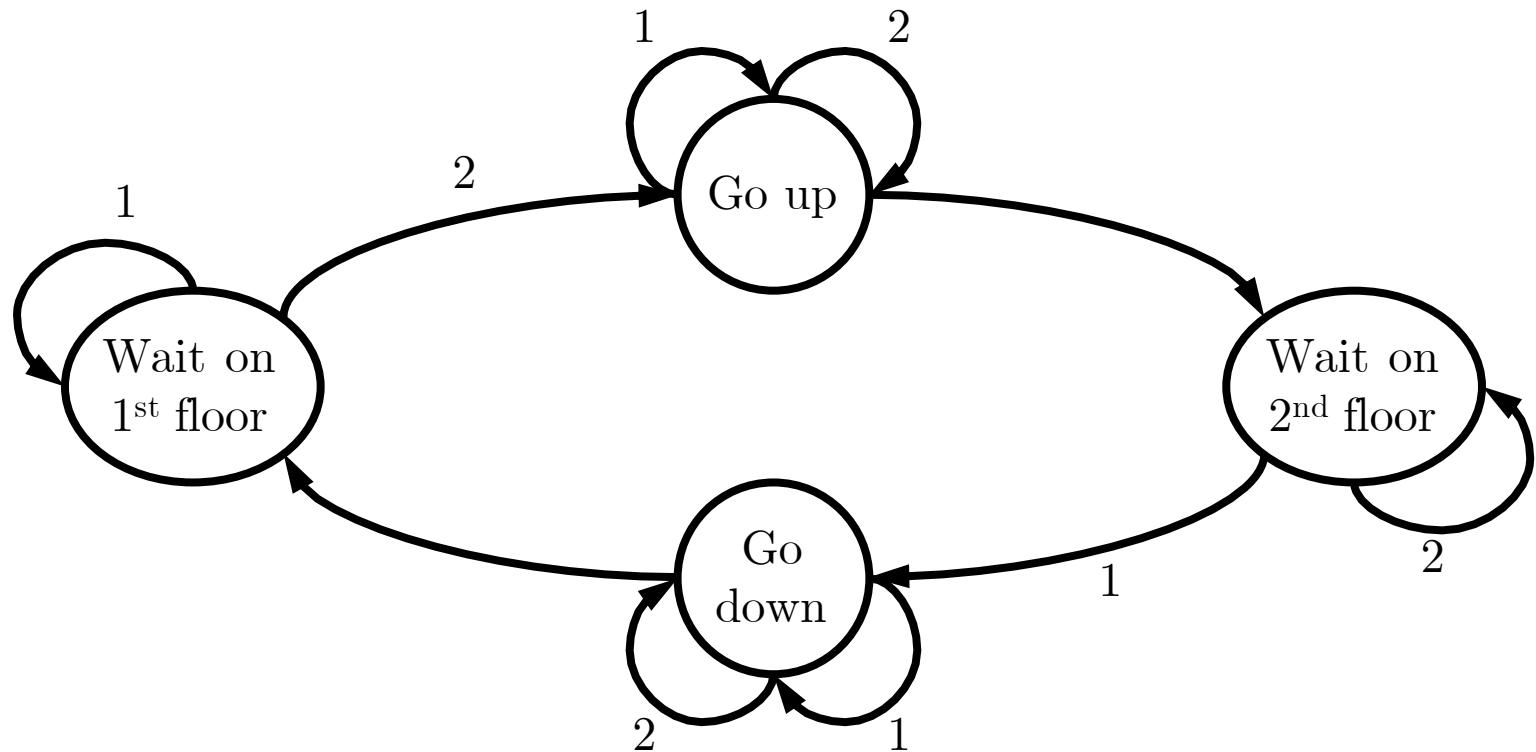
mid 1960's

# VLSI CONTINENT



# Finite State Machine

- Model of computational system
- Nodes represent states
- Edges represent transitions between states

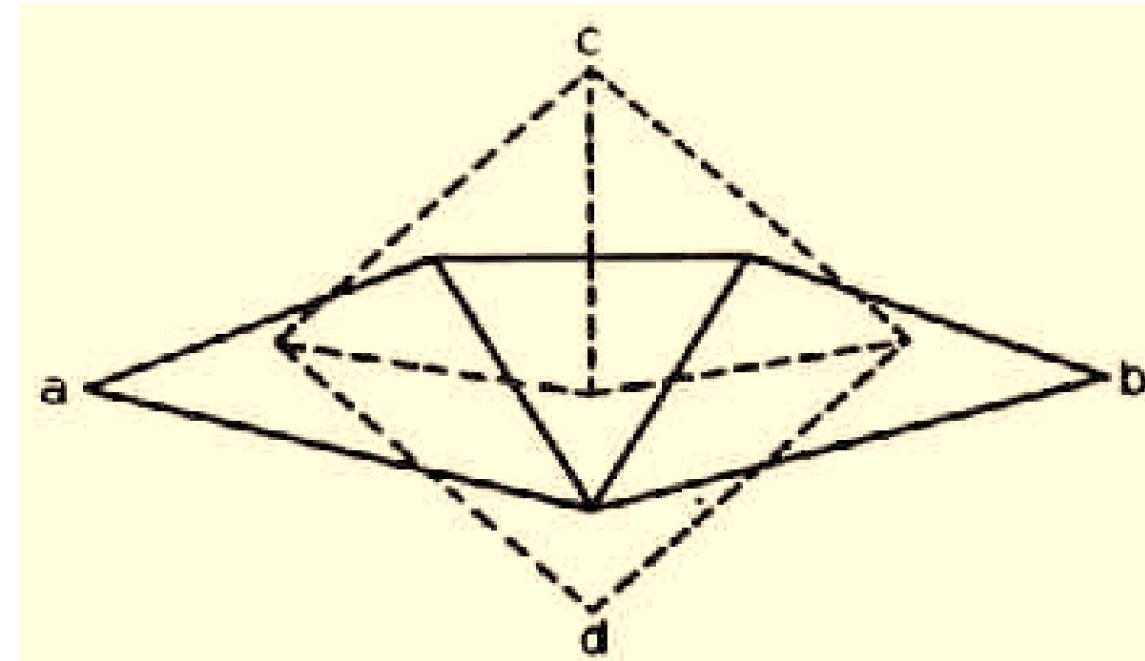
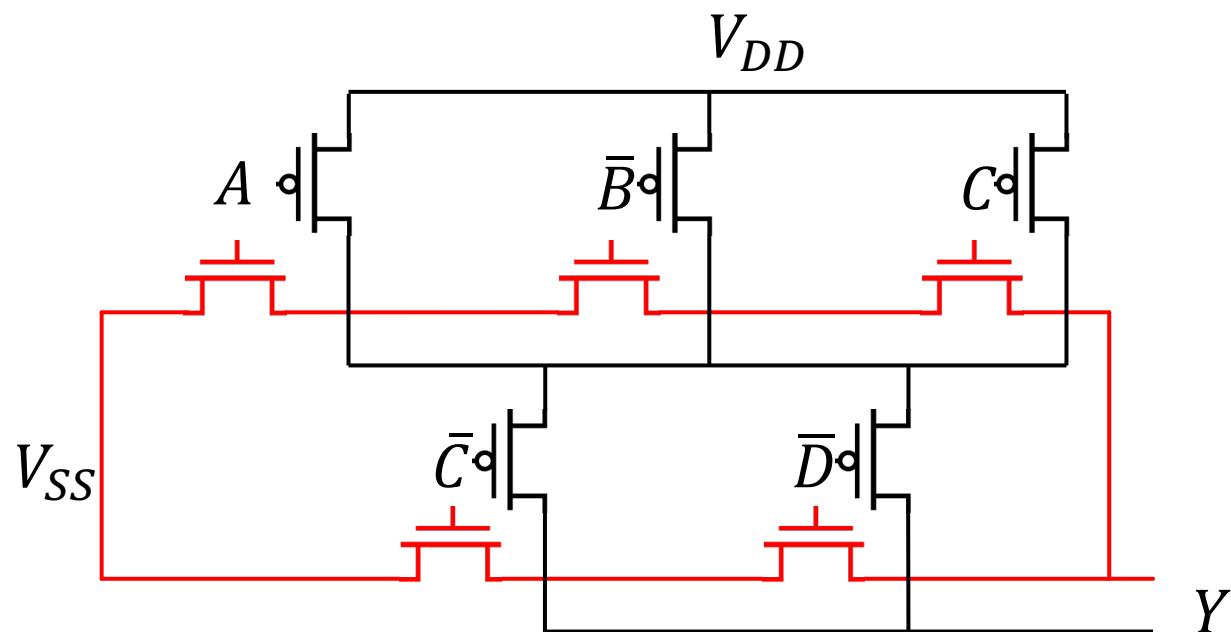


G. H. Mealy, “A Method for Synthesizing Sequential Circuits,” *The Bell System Technical Journal*, Vol. 34, no. 5, pp. 1045–1079, September 1955.

E. F. Moore, “Gedanken-Experiments on Sequential Machines,” *Automata Studies*, Vol. 34, pp. 129–154, April 1956.

# Logic Graph

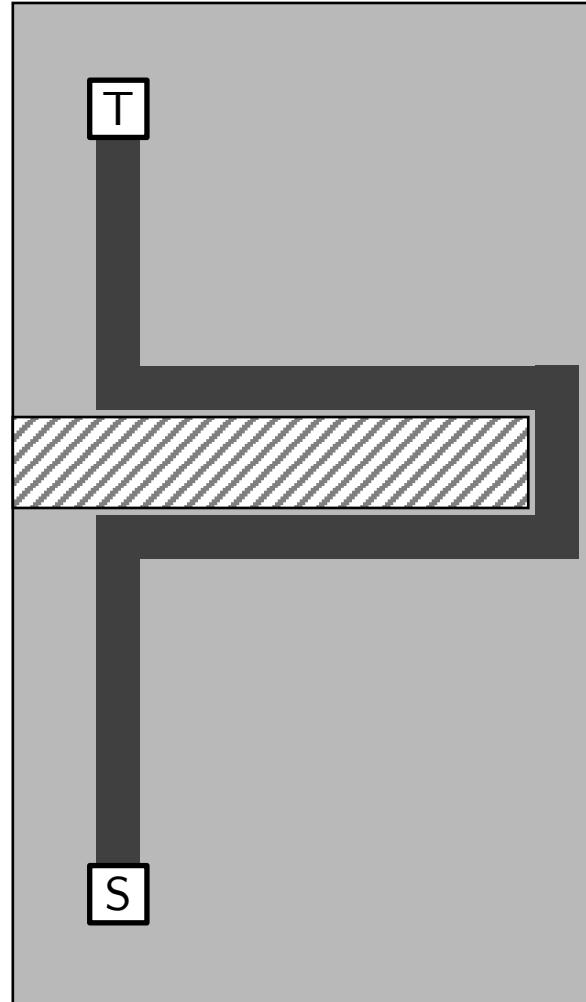
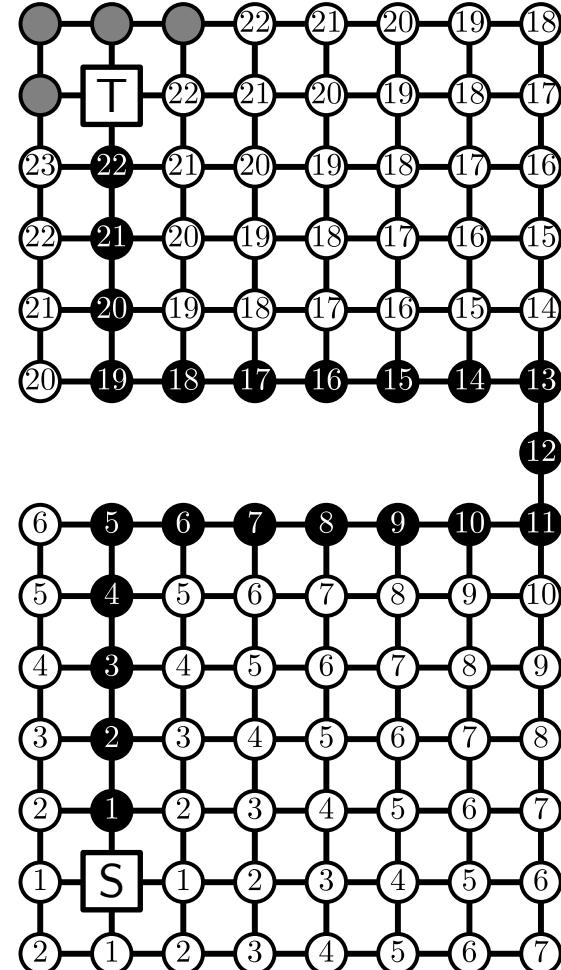
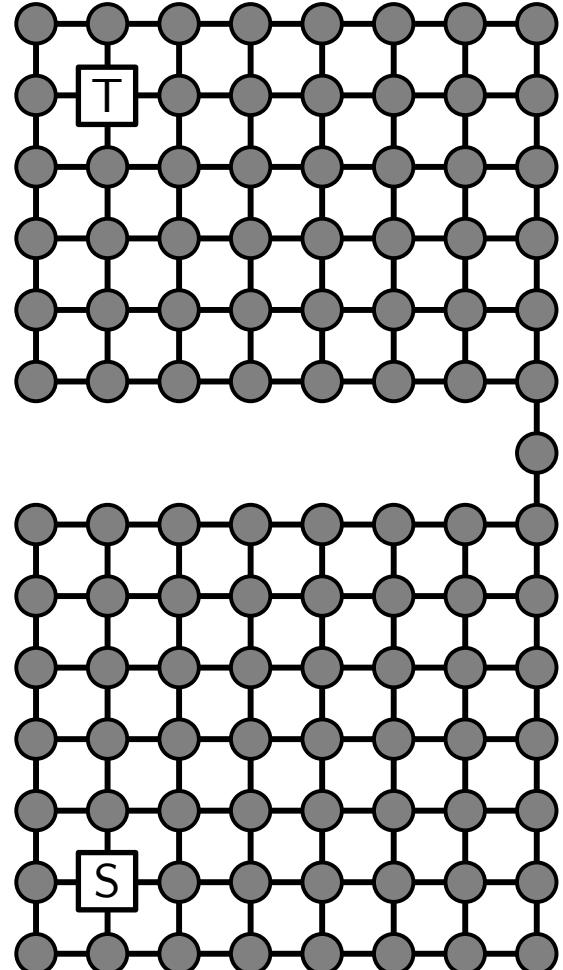
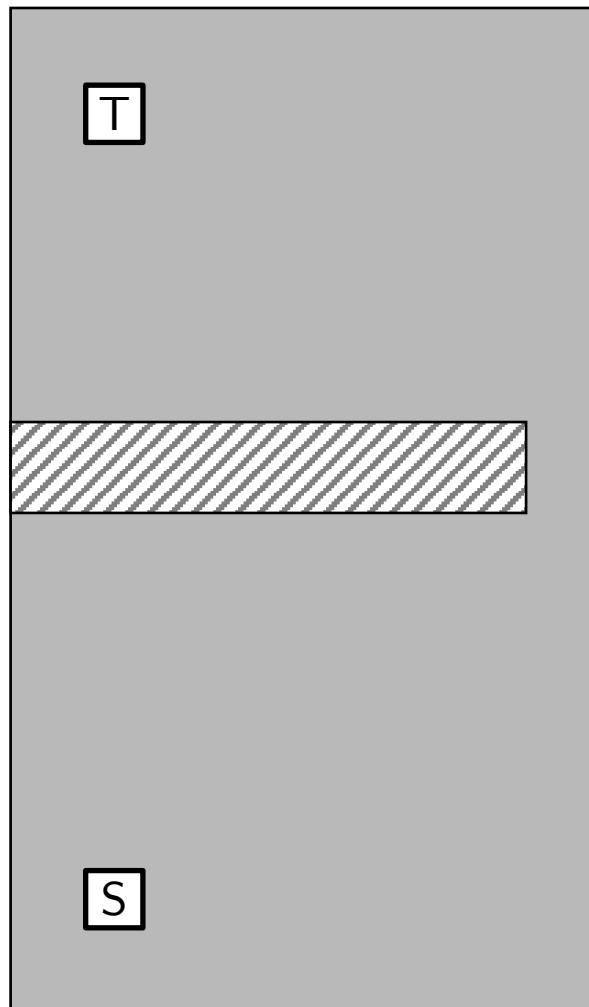
- Graphical representation of logical operations
- Powerful analysis of complex logic



**Figure 17. Superposition of a network and its dual**

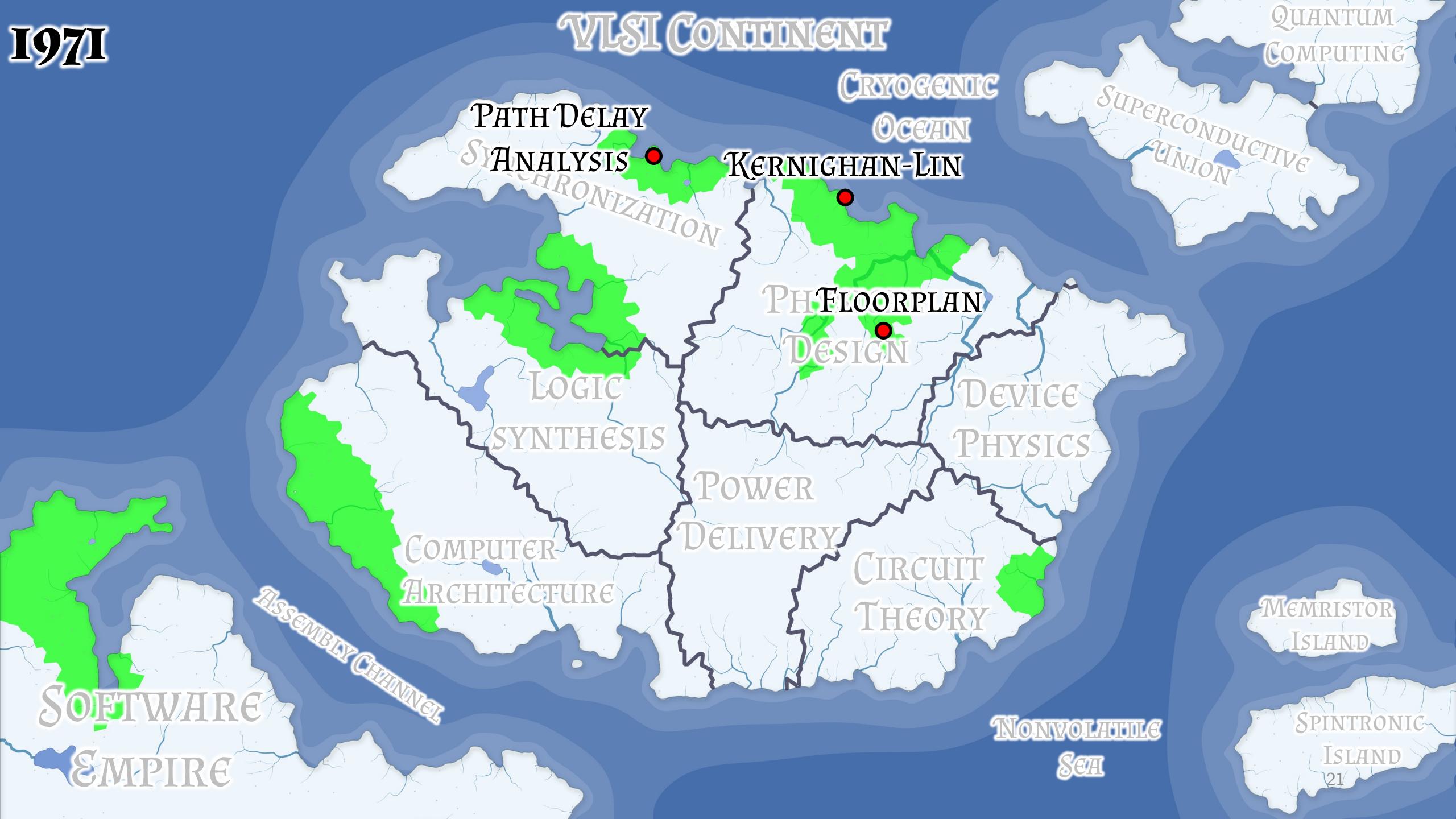
C. E. Shannon, "A Symbolic Analysis of Relay and Switching Circuits," *Electrical Engineering*, Vol. 57, no. 12, pp. 713–723, December 1938.

# Maze Routing



I97I

# VLSI CONTINENT



# Floorplanning

- Floorplan efficiently represented using graphs
- Encodes relations between blocks
  - Adjacency
  - Relative position

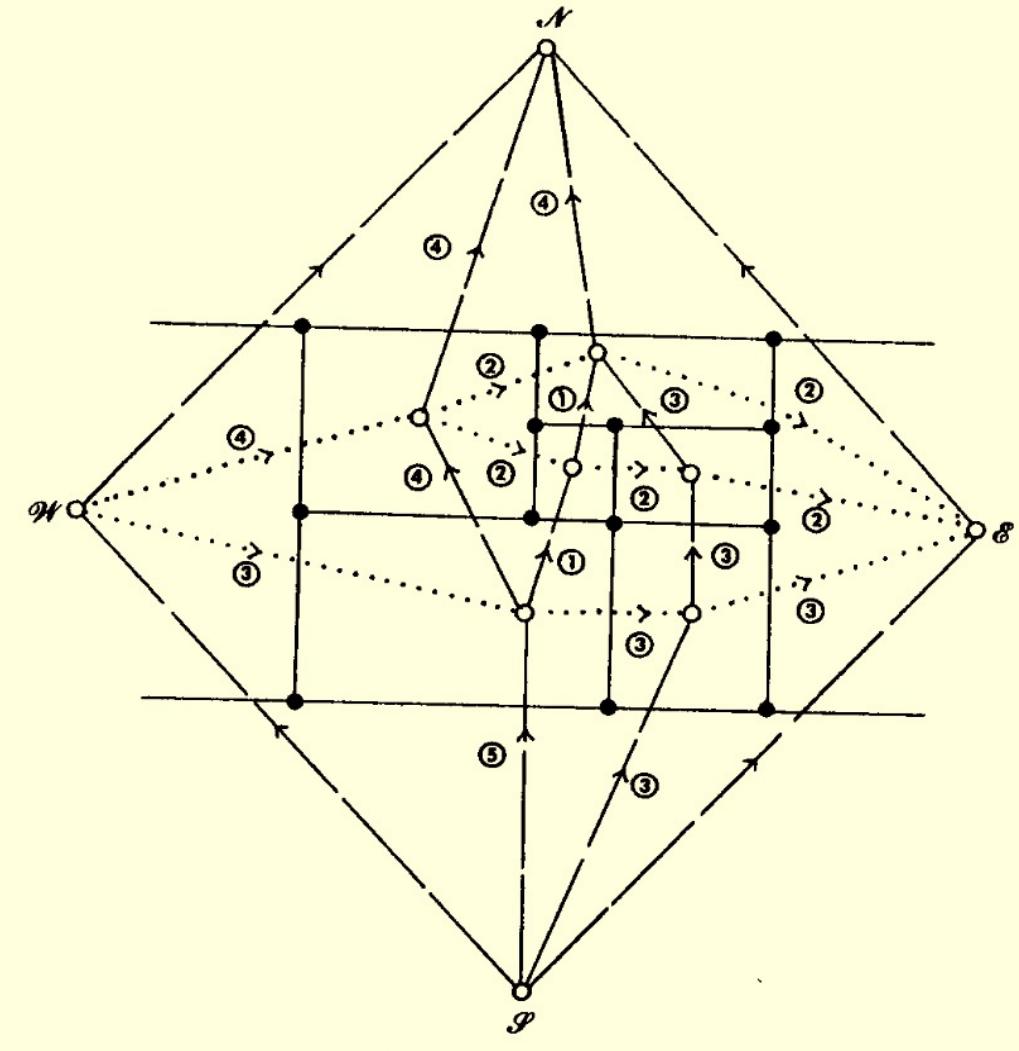
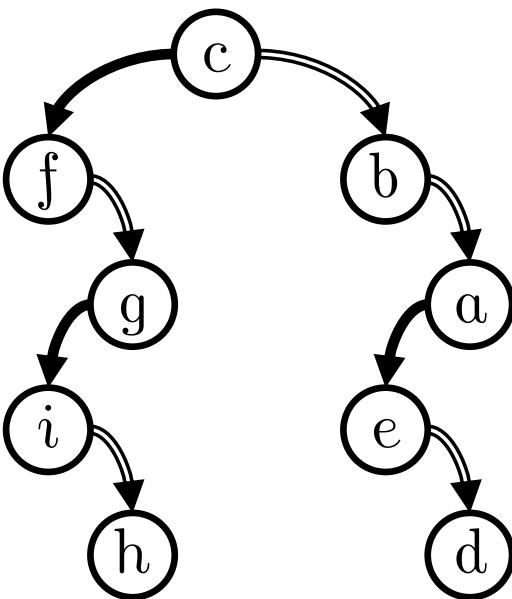
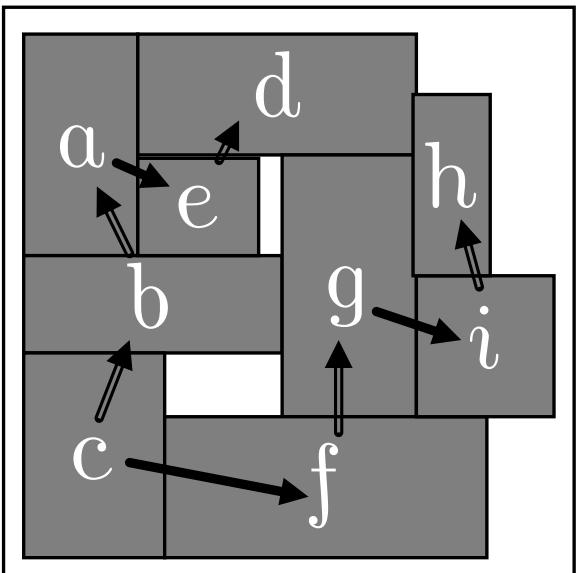


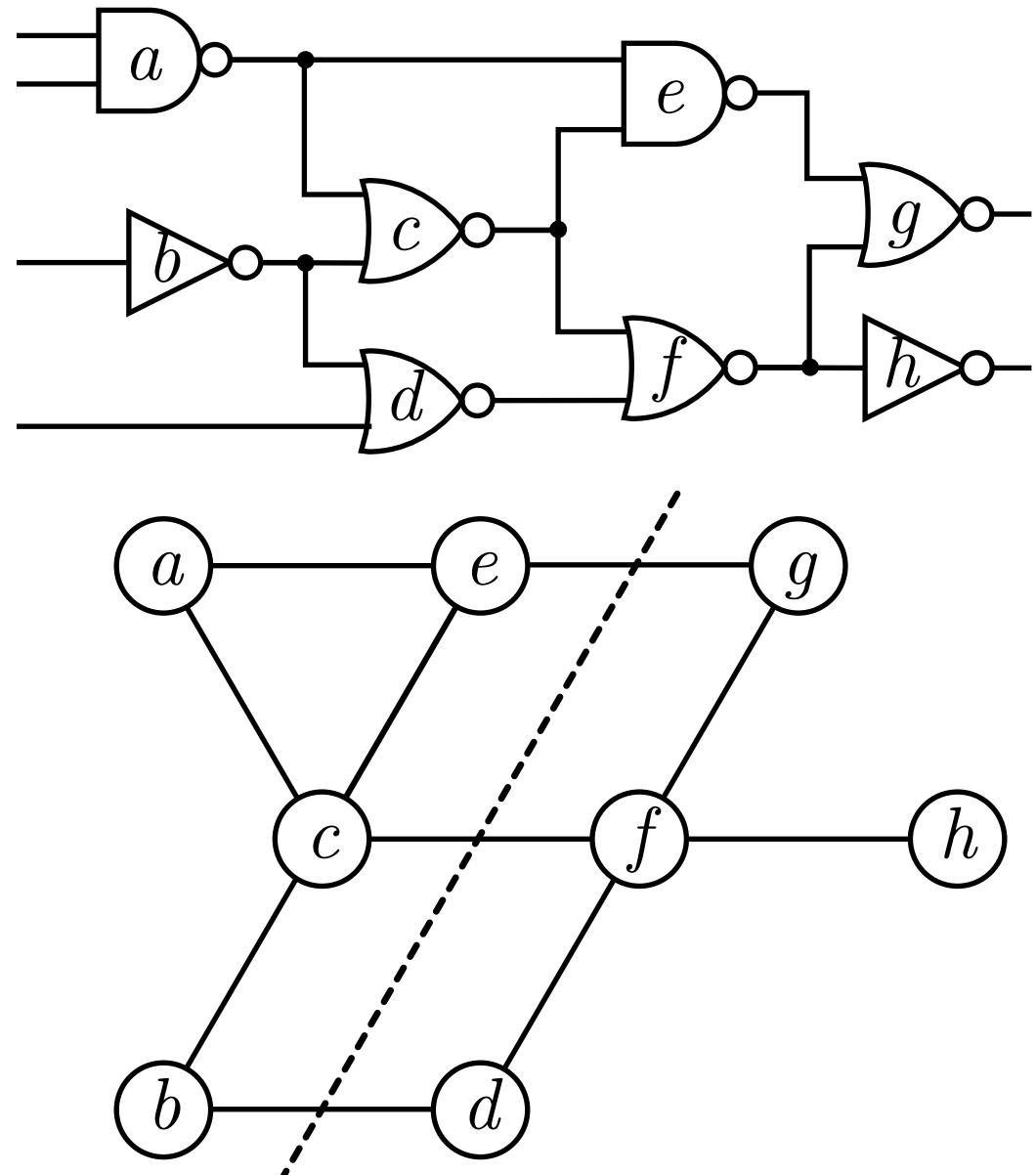
Figure 3  
FLOOR PLAN GRAPH WITH DUAL GRAPH

F. Mao, N. Xu, and Y. Ma, "Hybrid Algorithm for Floorplanning using B\*-Tree Representation," *Proceedings of the IEEE International Symposium on Intelligent Information Technology Application*, Vol. 3, pp. 228–231, November 2009

J. Grason, "An Approach to Computerized Space Planning Using Graph Theory," *Proceedings of the ACM/IEEE Design Automation Workshop*, pp. 170-179, June 1971

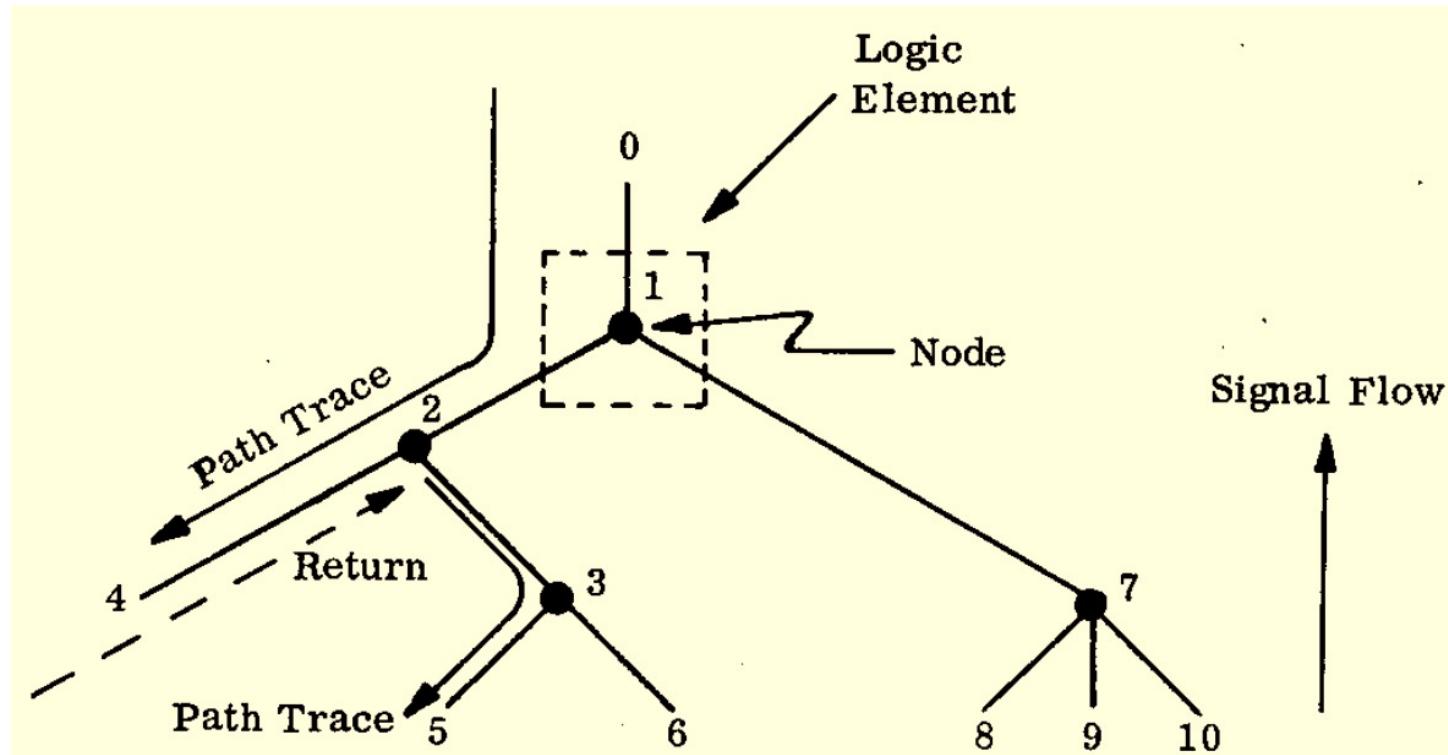
# Circuit Partitioning

- Split circuit into modules
- Minimize number of pins
  - External connections
- Solved using graph minimum cut
  - Split graph into two equal sets
  - Reduce number of external edges



# Race Analysis System (RAS)

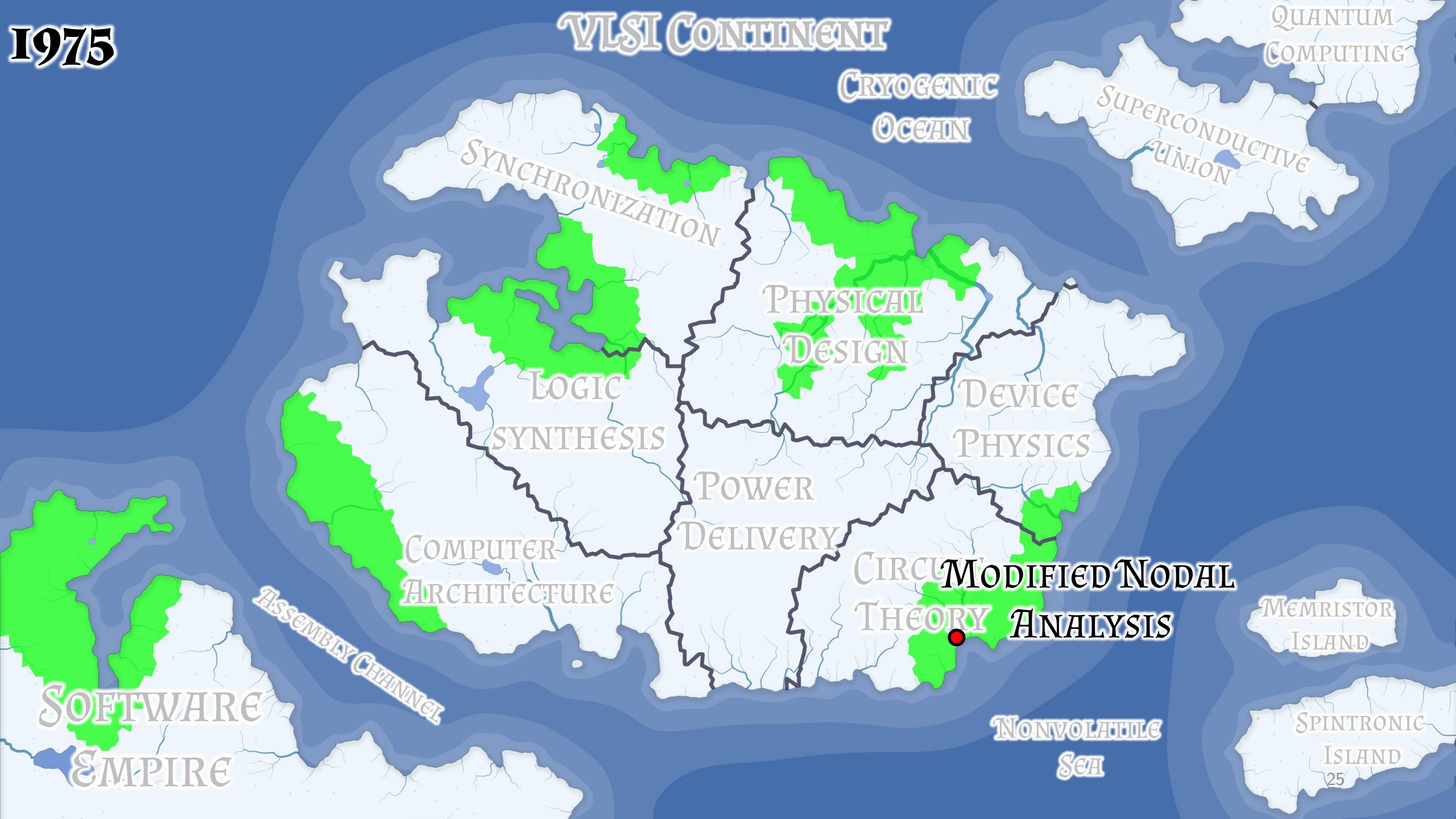
- Sequential circuit converted to data flow graph
- Signal delay accumulated from each flip flop output to clock source
- Timing hazards are efficiently located



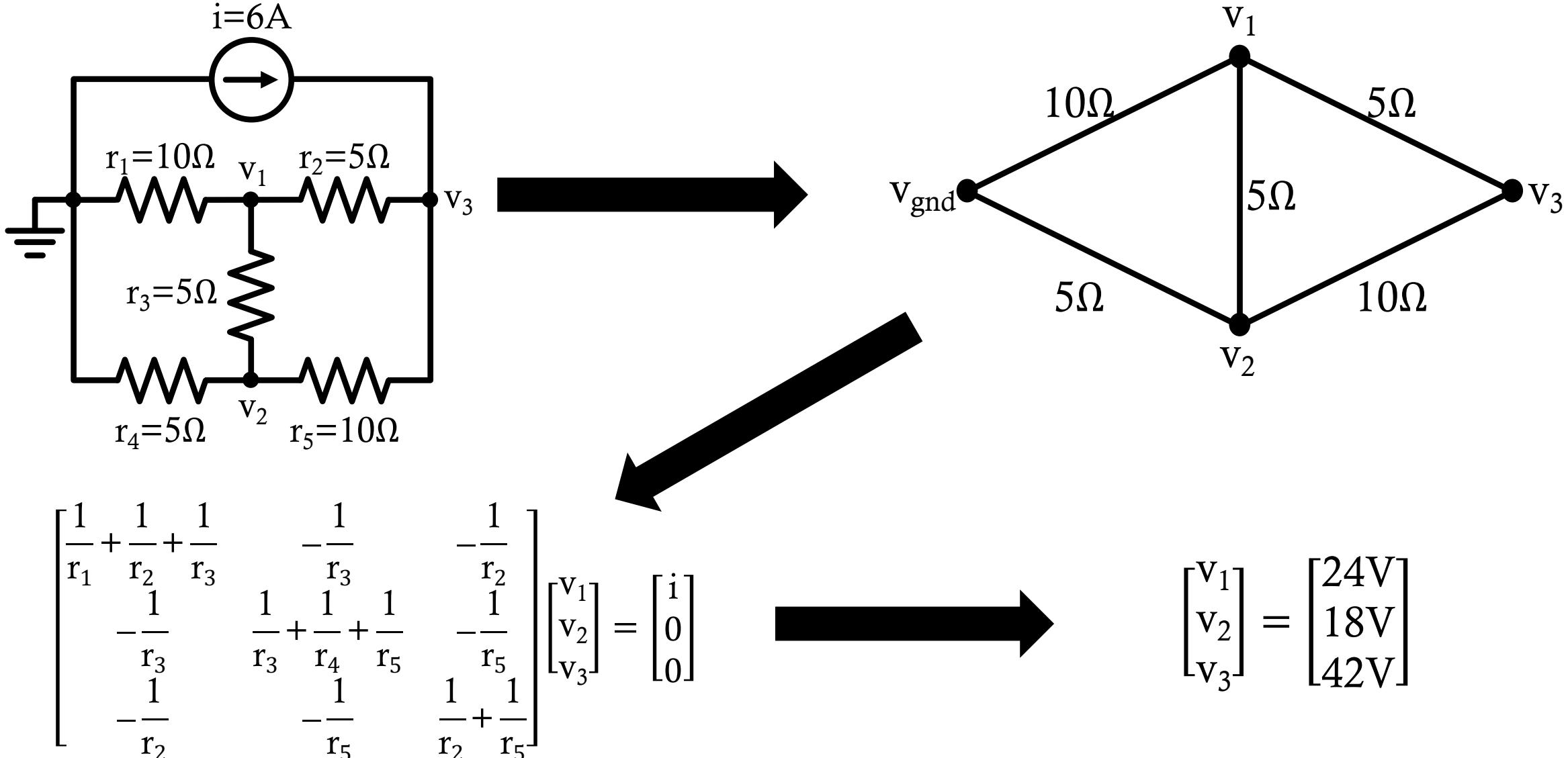
R. A. Harrison and D. J. Olson, “Race Analysis of Digital Systems Without Logic Simulation,” *Proceedings of the IEEE/ACM Design Automation Workshop*, pp. 82–94, June 1971.

I975

# VLSI CONTINENT

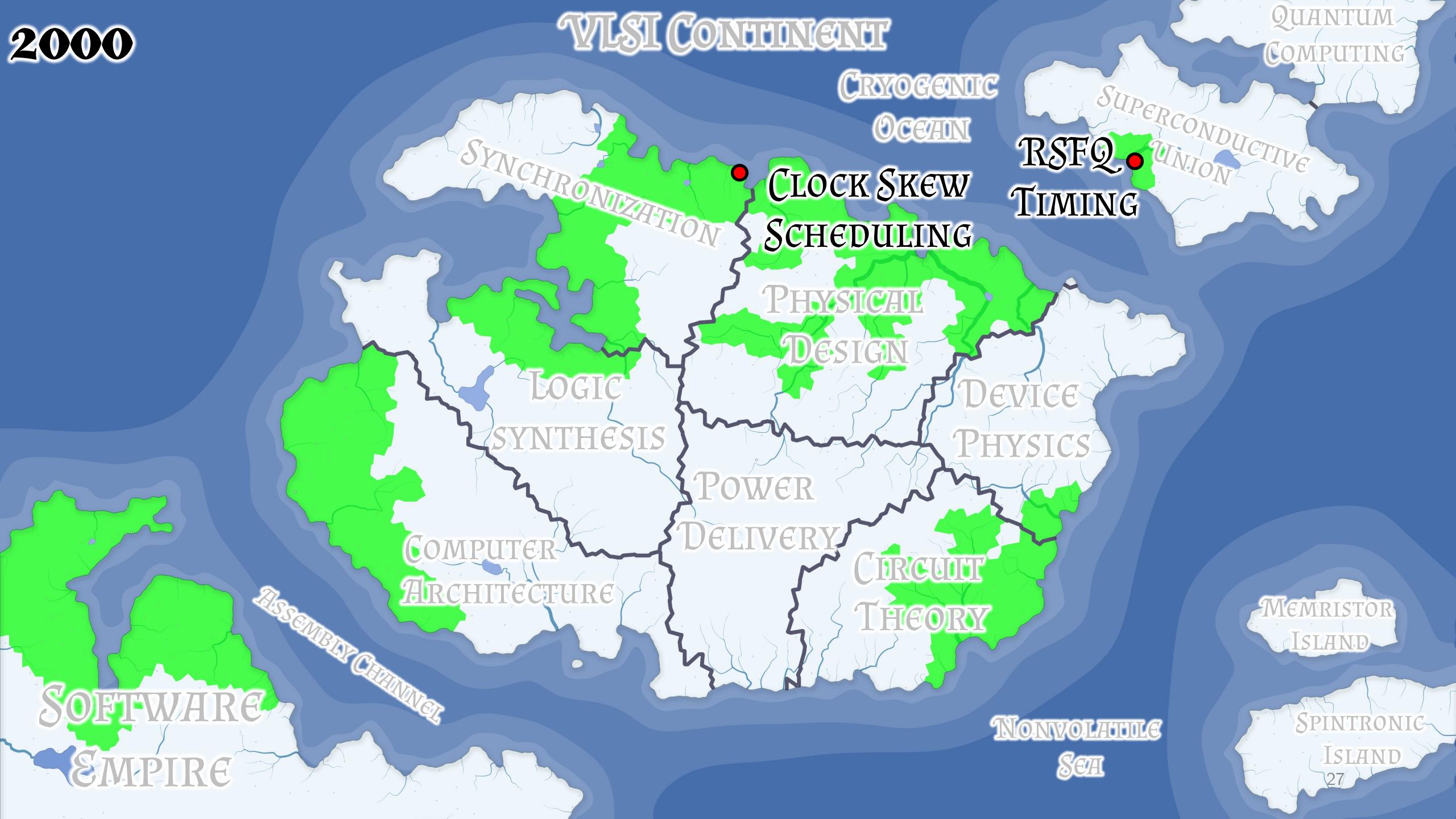


# Modified Nodal Analysis



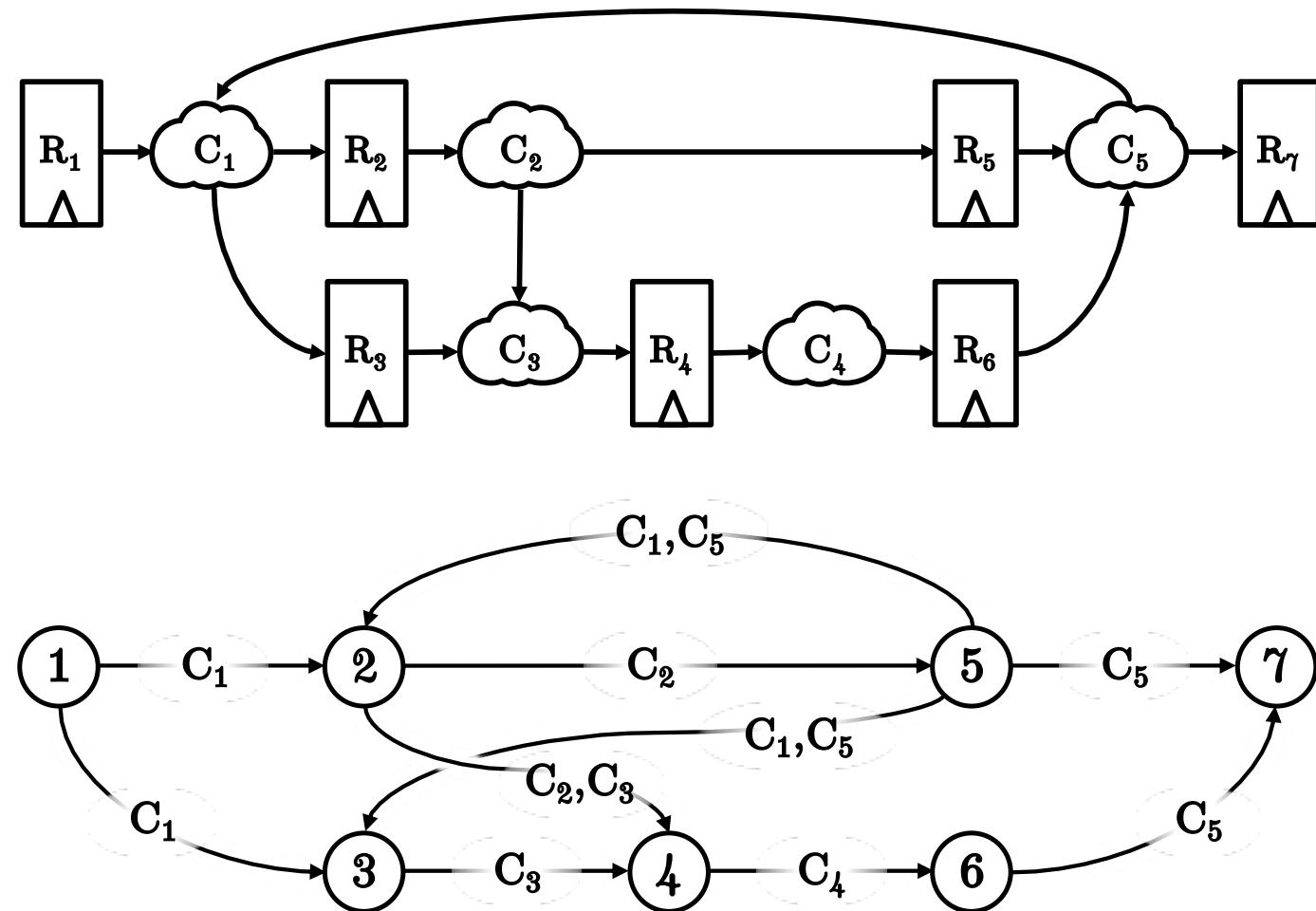
2000

# VLSI CONTINENT



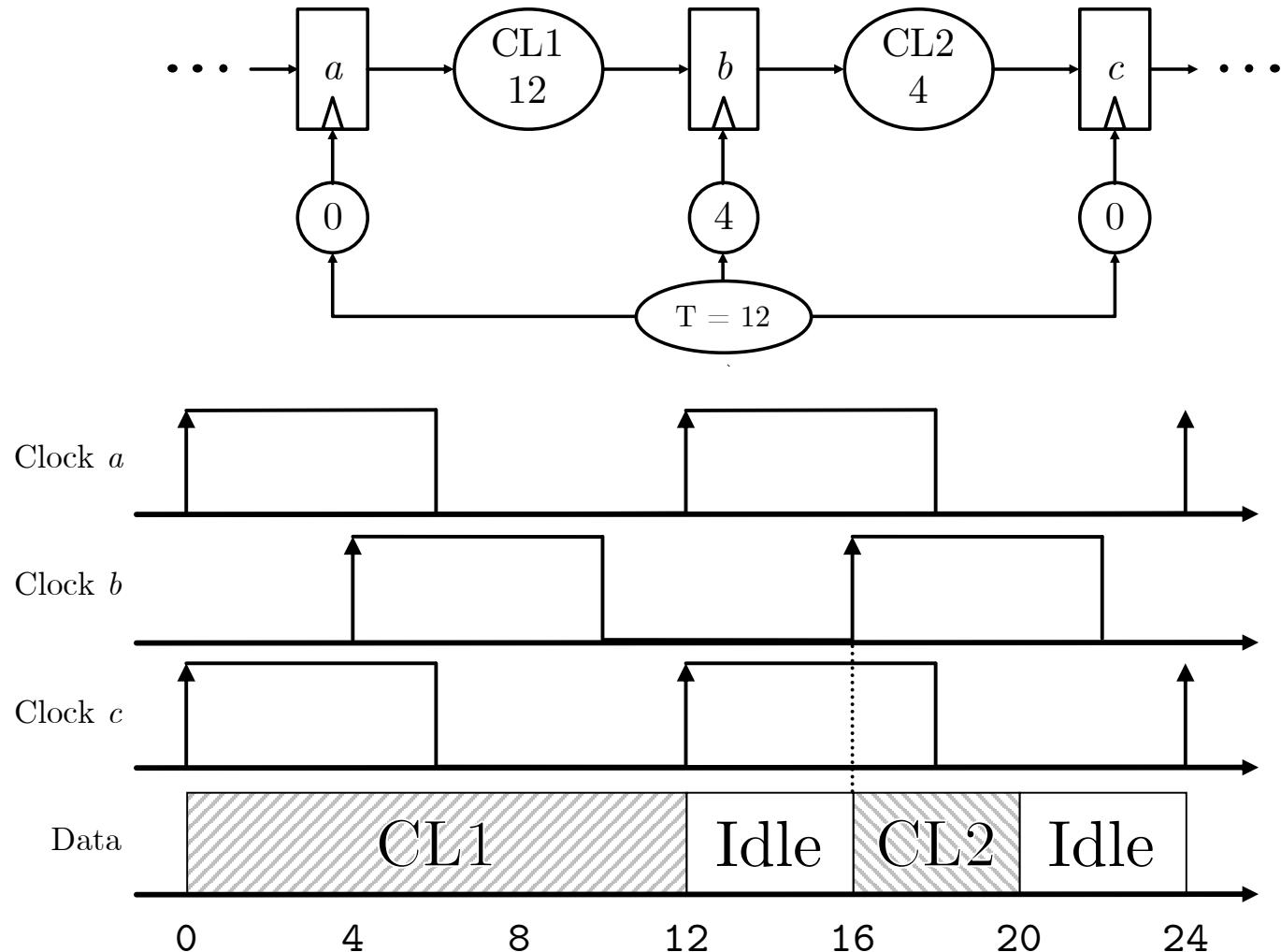
# Synchronization in VLSI Systems

- VLSI systems are mostly synchronous
  - Combinatorial logic executes functions
- Registers synchronize data flow
  - Clock signal required
- Modeled with timing graph
  - Nodes = registers
  - Edges = data paths



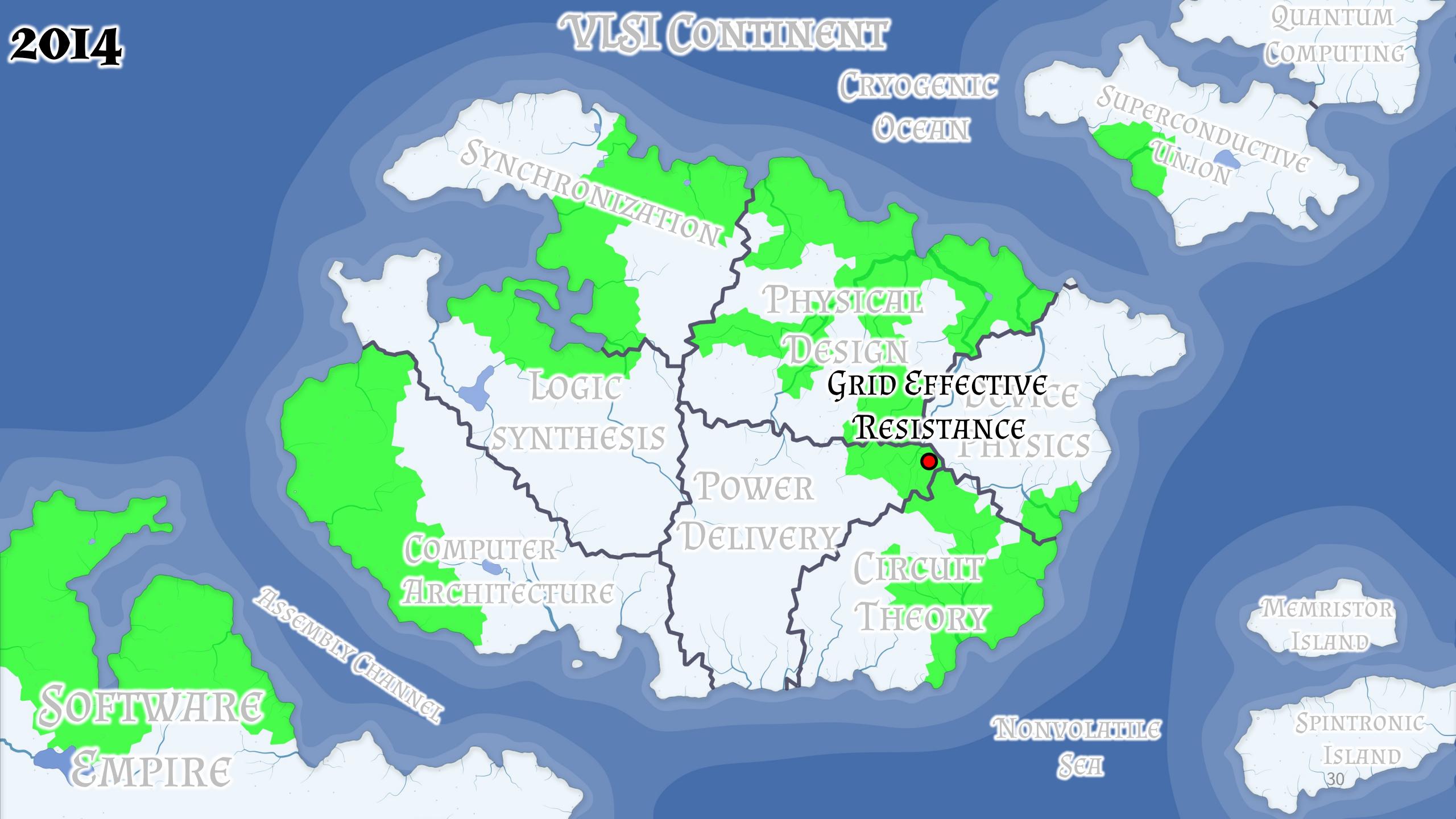
# Clock Skew Scheduling

- Sequential circuit graph
  - Efficiently determine clock arrival time
  - Optimized using quadratic programming
- Clock arrival time adjusted to enhance
  - Robustness
  - Performance



2014

# VLSI CONTINENT

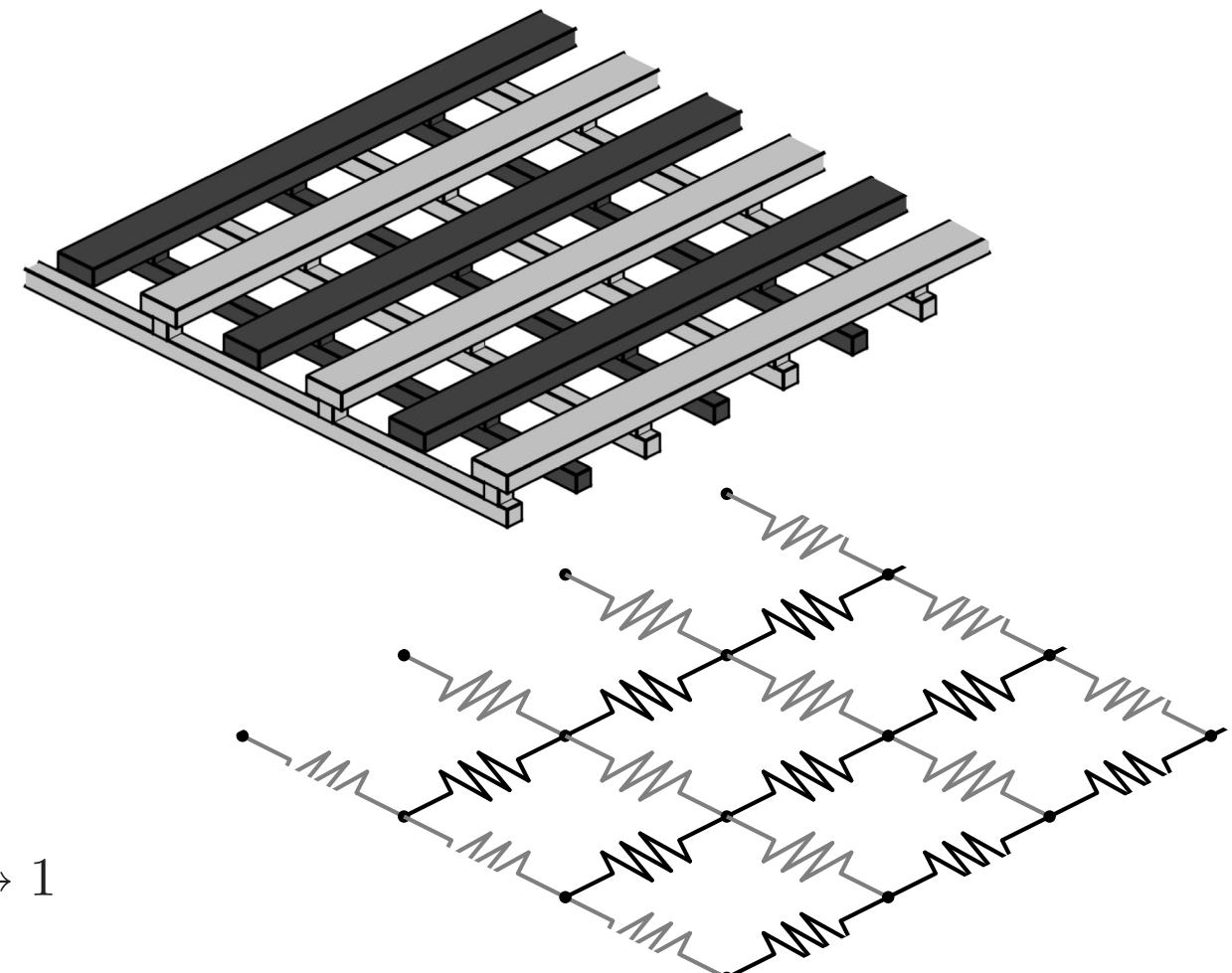


# Effective Resistance in Resistive Mesh

- Effective resistance in mesh estimated in  $O(1)$  time
- Accelerates IR drop analysis

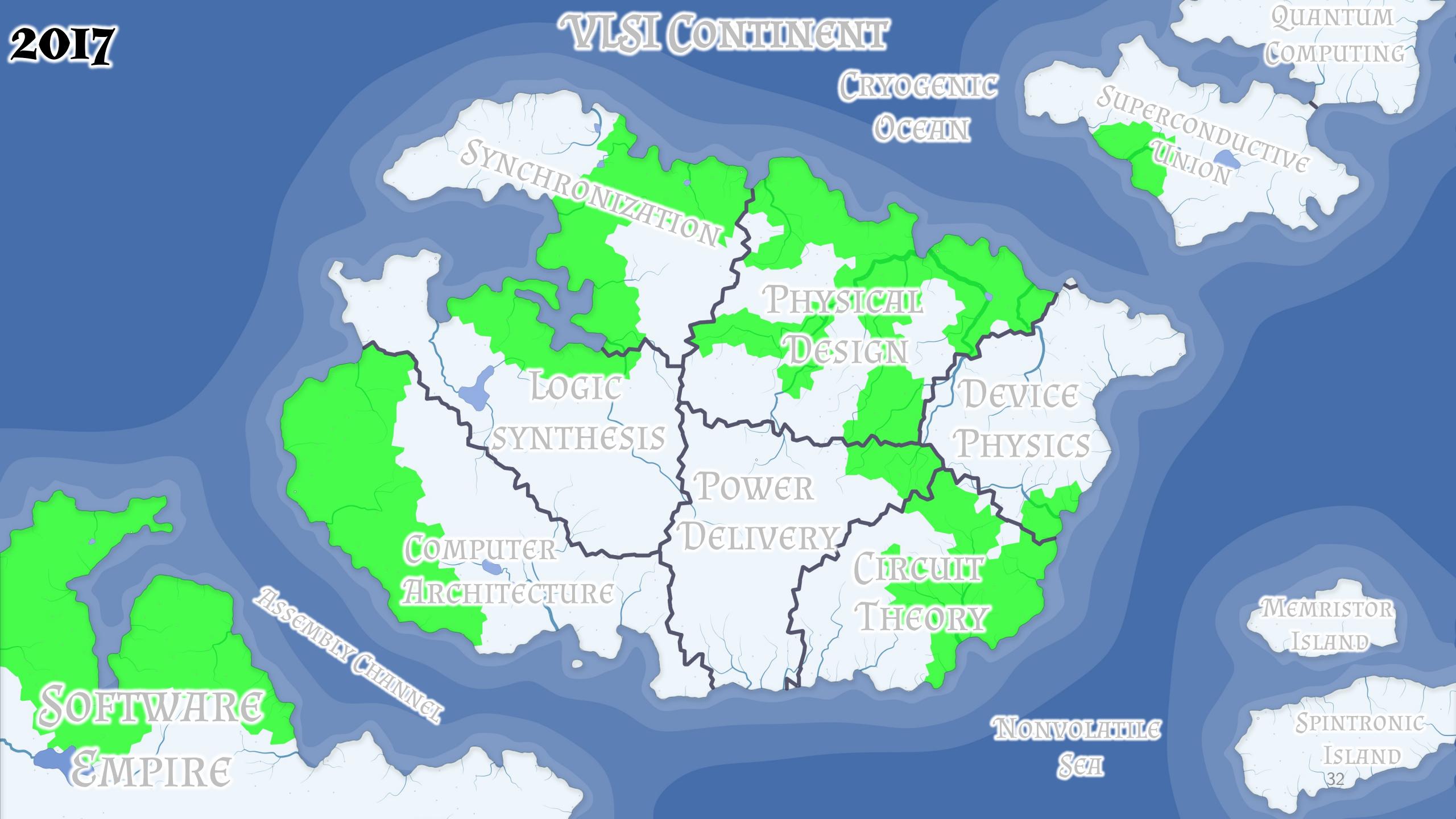
$$R_{x,y} = \frac{kr}{\pi} \int_0^\pi \frac{(2 - e^{-|x|\alpha} \cos y\beta)}{\sinh \alpha} d\beta$$

$$R_{x,y}/r = \frac{\sqrt{k}}{2\pi} [\ln(x^2 + ky^2) + 3.44388] - 0.033425k - 0.0629k(k-1) \quad \text{for } k \rightarrow 1$$



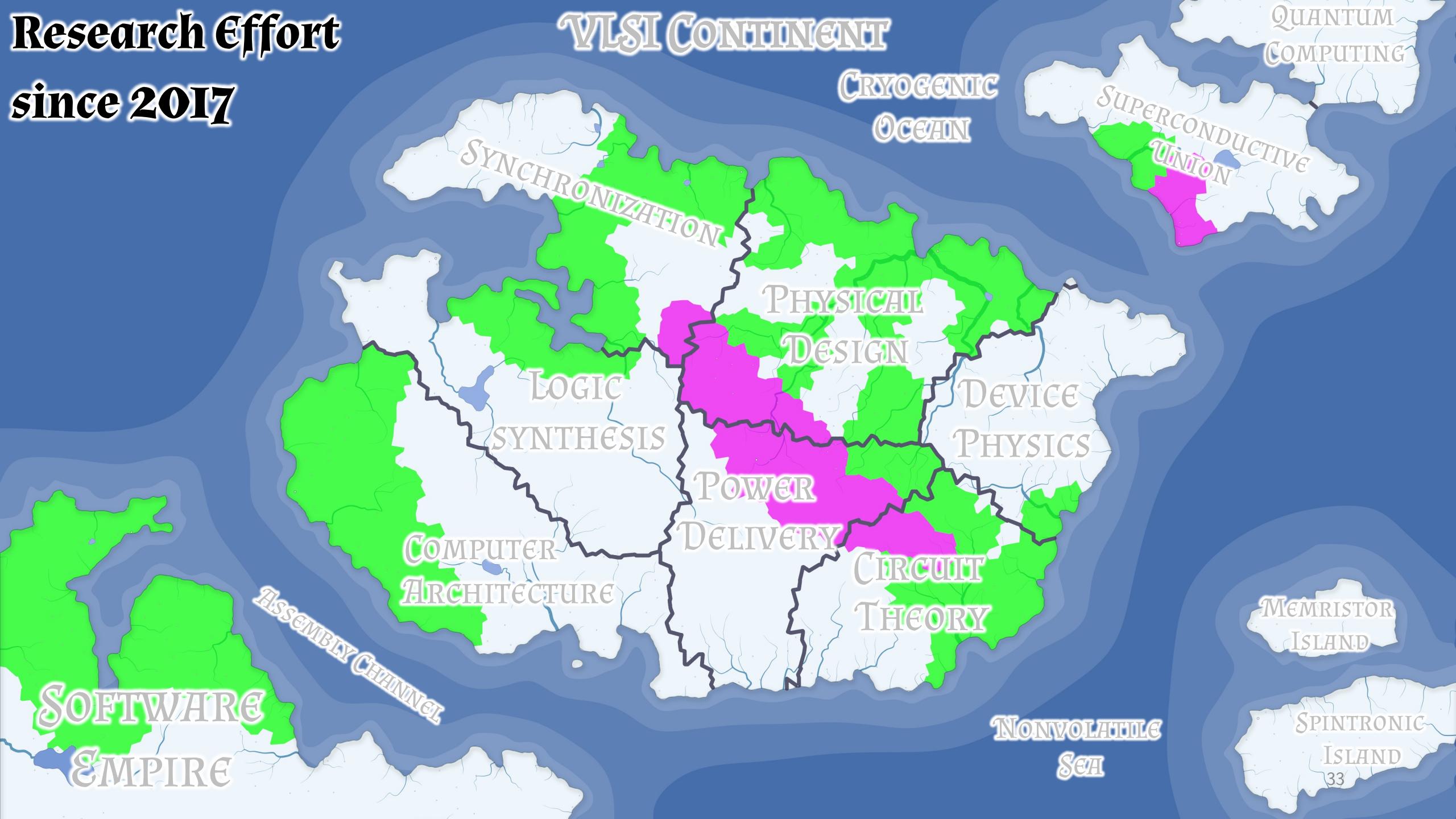
2017

# VLSI CONTINENT



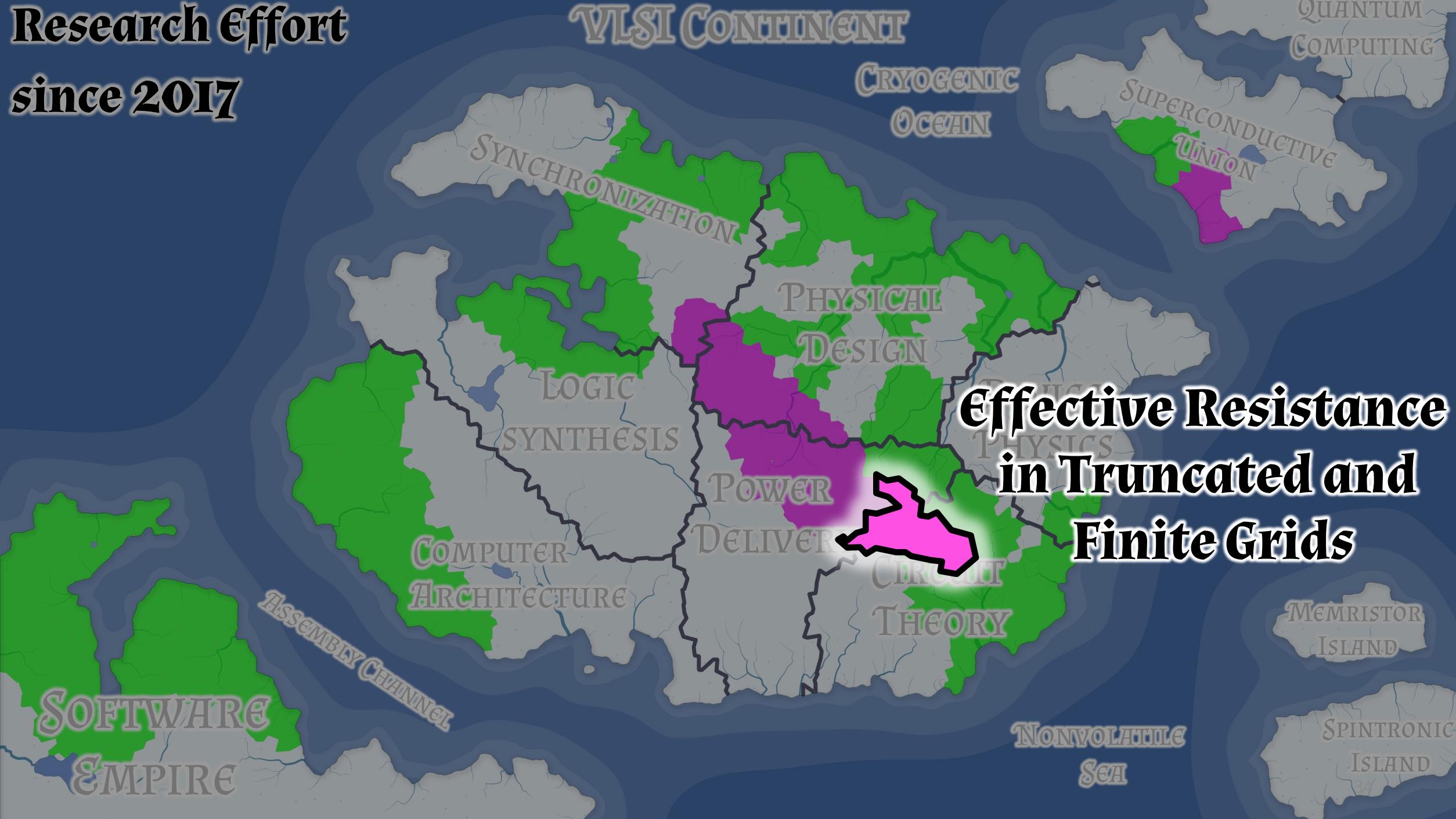
# Research Effort since 2017

## VLSI CONTINENT



**Research Effort  
since 2017**

**VLSI CONTINENT**

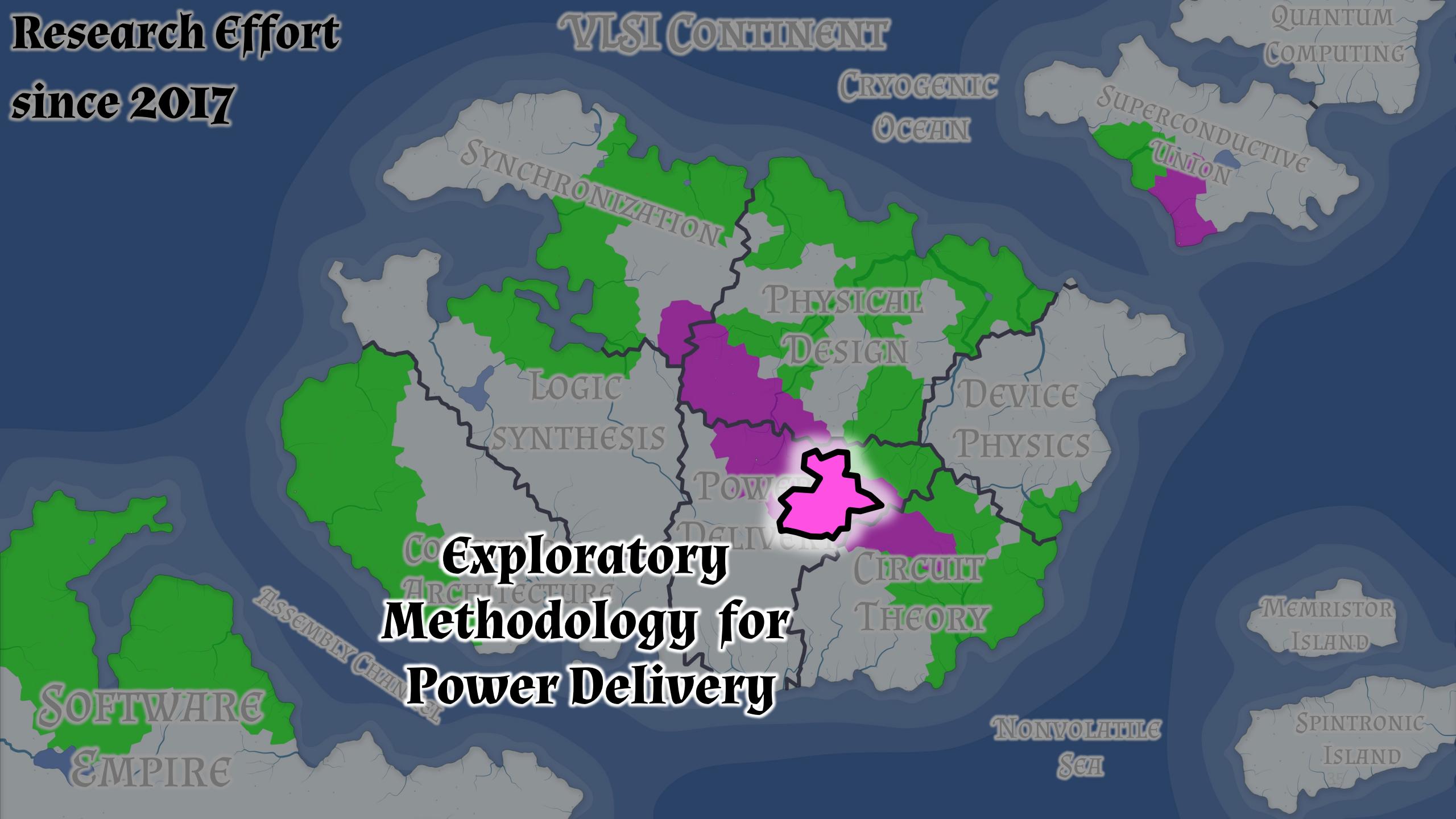


**Effective Resistance  
in Truncated and  
Finite Grids**

**Research Effort  
since 2017**

**VLSI CONTINENT**

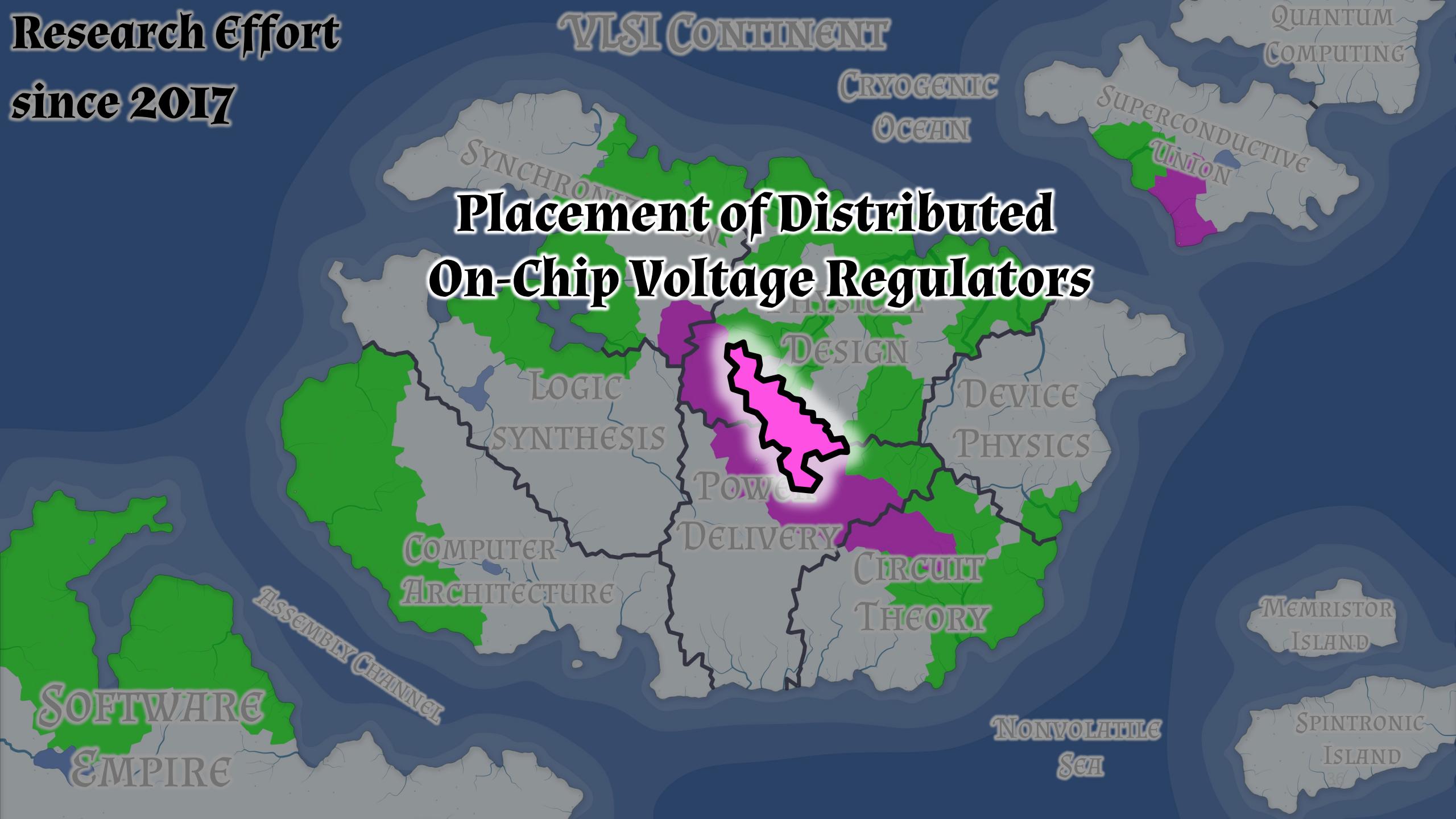
# **Exploratory Methodology for Power Delivery**



**Research Effort**  
since 2017

VLSI CONTINENT

# Placement of Distributed On-Chip Voltage Regulators



**Research Effort  
since 2017**

**VLSI CONTINENT**

**CRYOGENIC  
OCEAN**

**SUPERCONDUCTIVE UNION**

**SYNCHRONIZATION**

**PHYSICAL  
DESIGN**

**LOGIC  
SYNTHESIS**

**DEVICE  
PHYSICS**

**POWER  
DELIVERY**

# **SPROUT – Smart Power ROUTing Tool for Power Network Prototyping**

**ASSEMBLY CHANNEL**

**SOFTWARE  
EMPIRE**

**NONVOLATILE  
SEA**

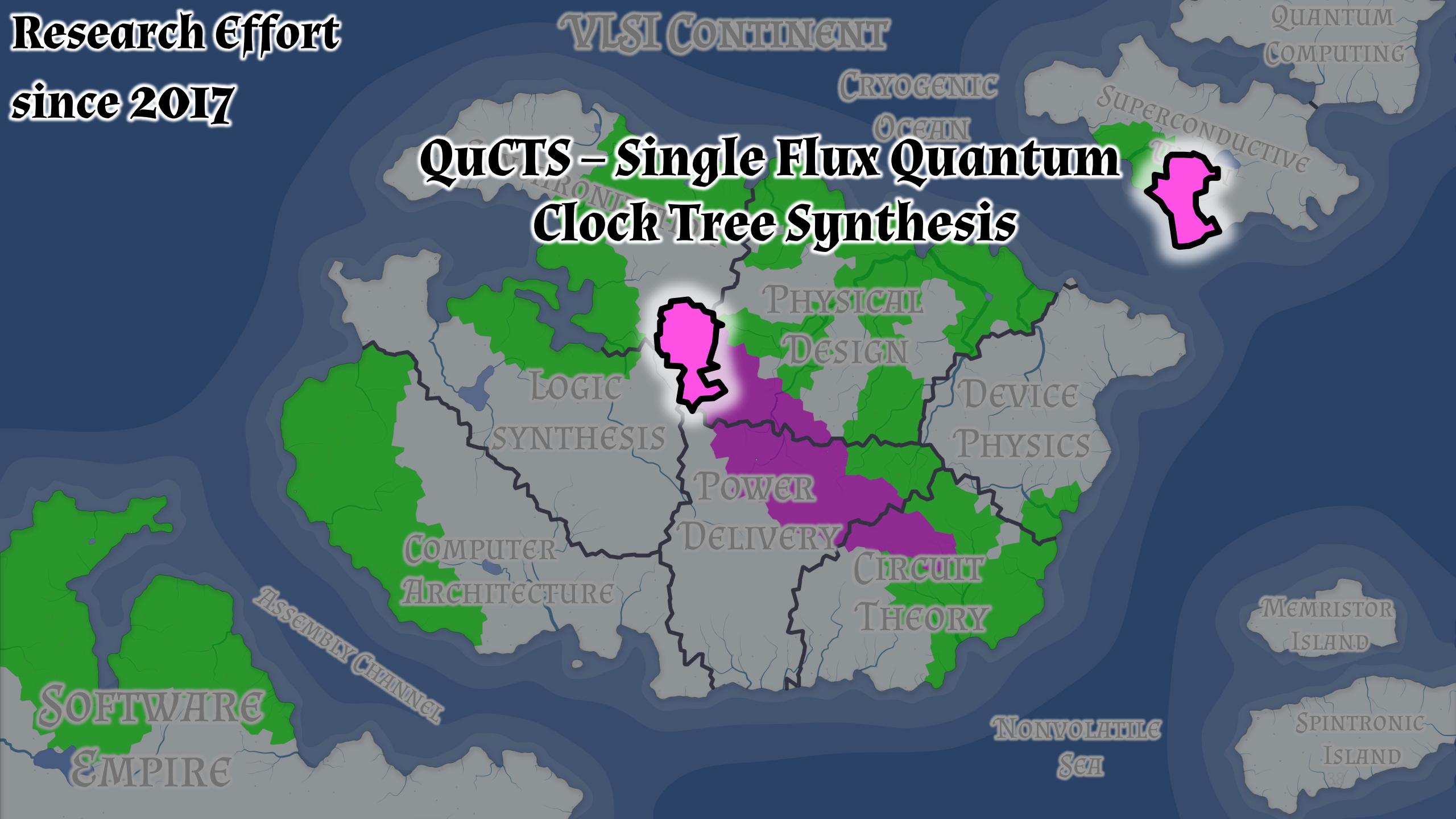
**MEMRISTOR  
ISLAND**

**SPINTRONIC  
ISLAND**

**Research Effort**  
**since 2017**

VLSI CONTINENT

# QuCTS – Single Flux Quantum Clock Tree Synthesis



**Research Effort  
since 2017**

**VLSI CONTINENT**

# **Exploratory Methodology for Power Delivery**



**Qualcomm**

**SOFTWARE  
EMPIRE**

**CRYOGENIC  
OCEAN**

**SUPERCONDUCTIVE  
UNION**

**QUANTUM  
COMPUTING**

**NONVOLATILE  
SEA**

**MEMRISTOR  
ISLAND**

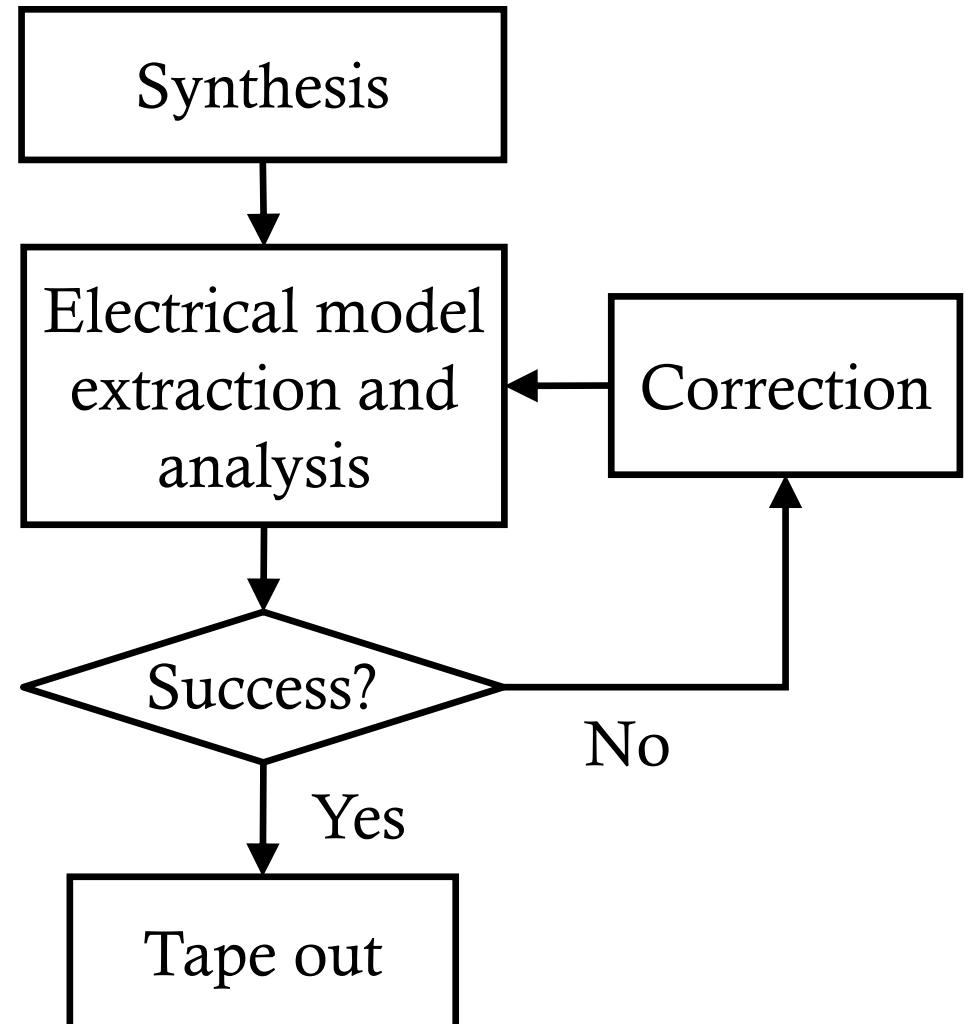
**SPINTRONIC  
ISLAND**

# Iterative Design Process

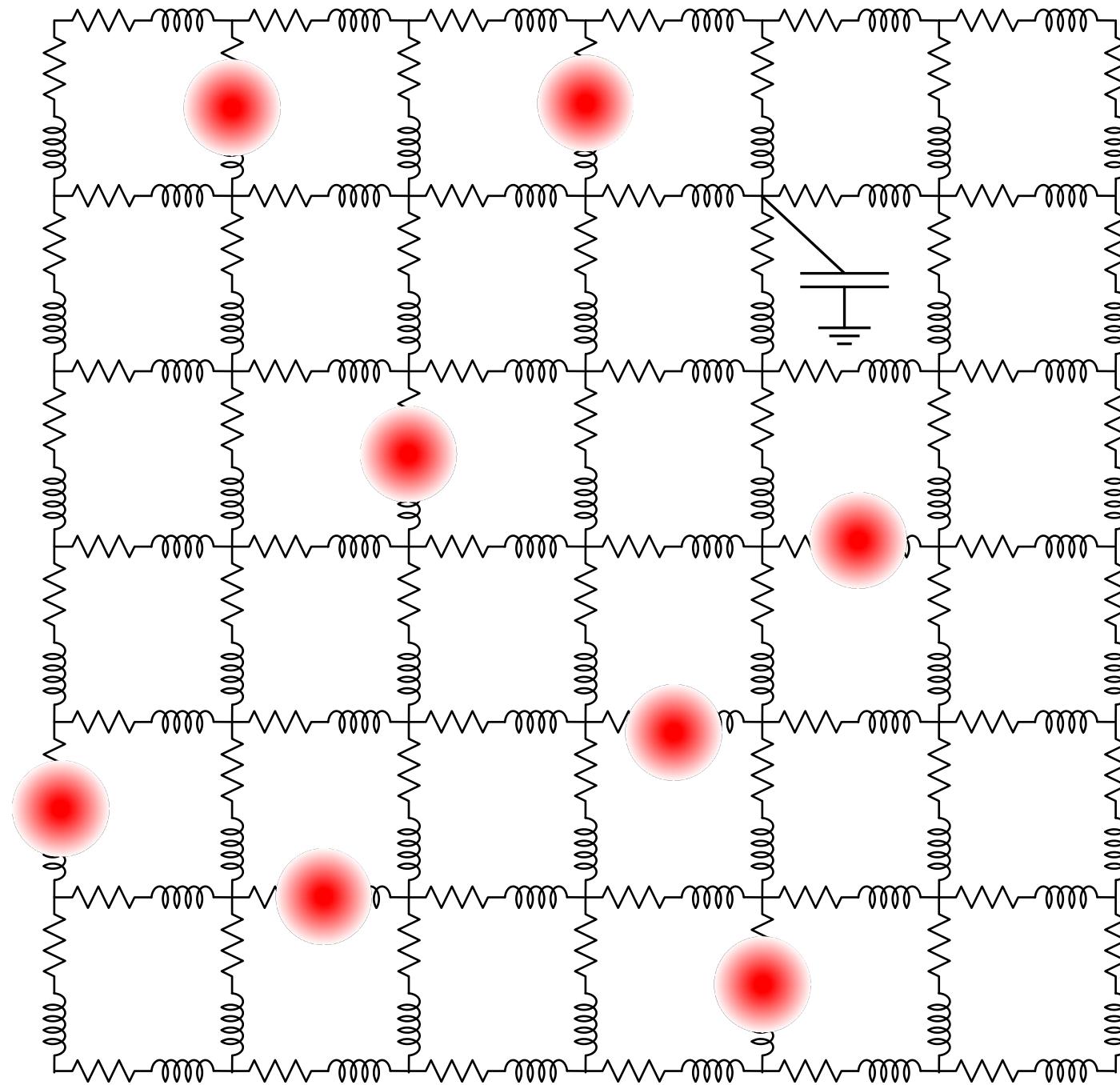
- Conventional analysis iterative in nature

$$T=Nt$$

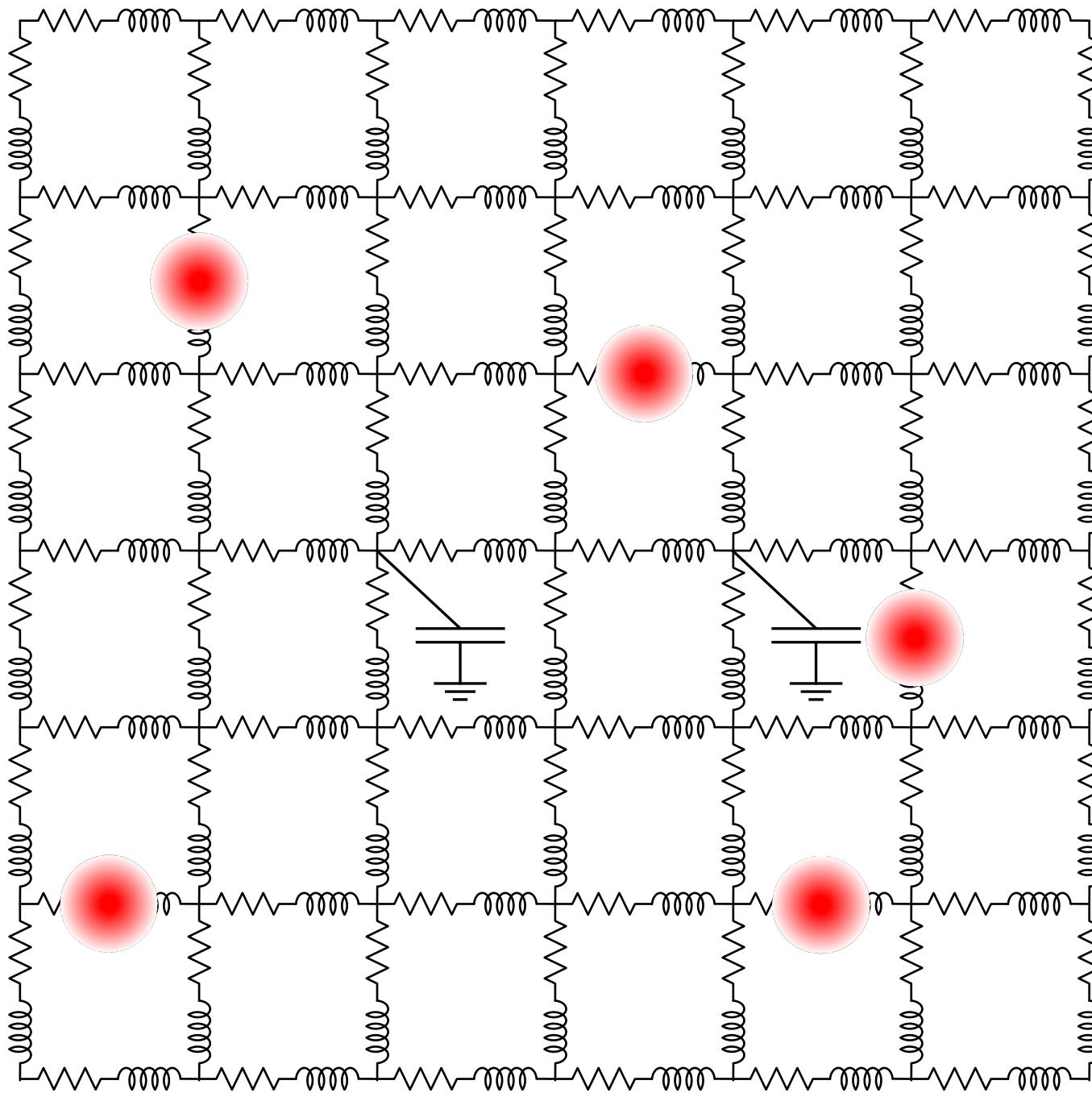
- $t$  – time for correction and analysis
- $N$  iterations to converge
- Significant delay in development process



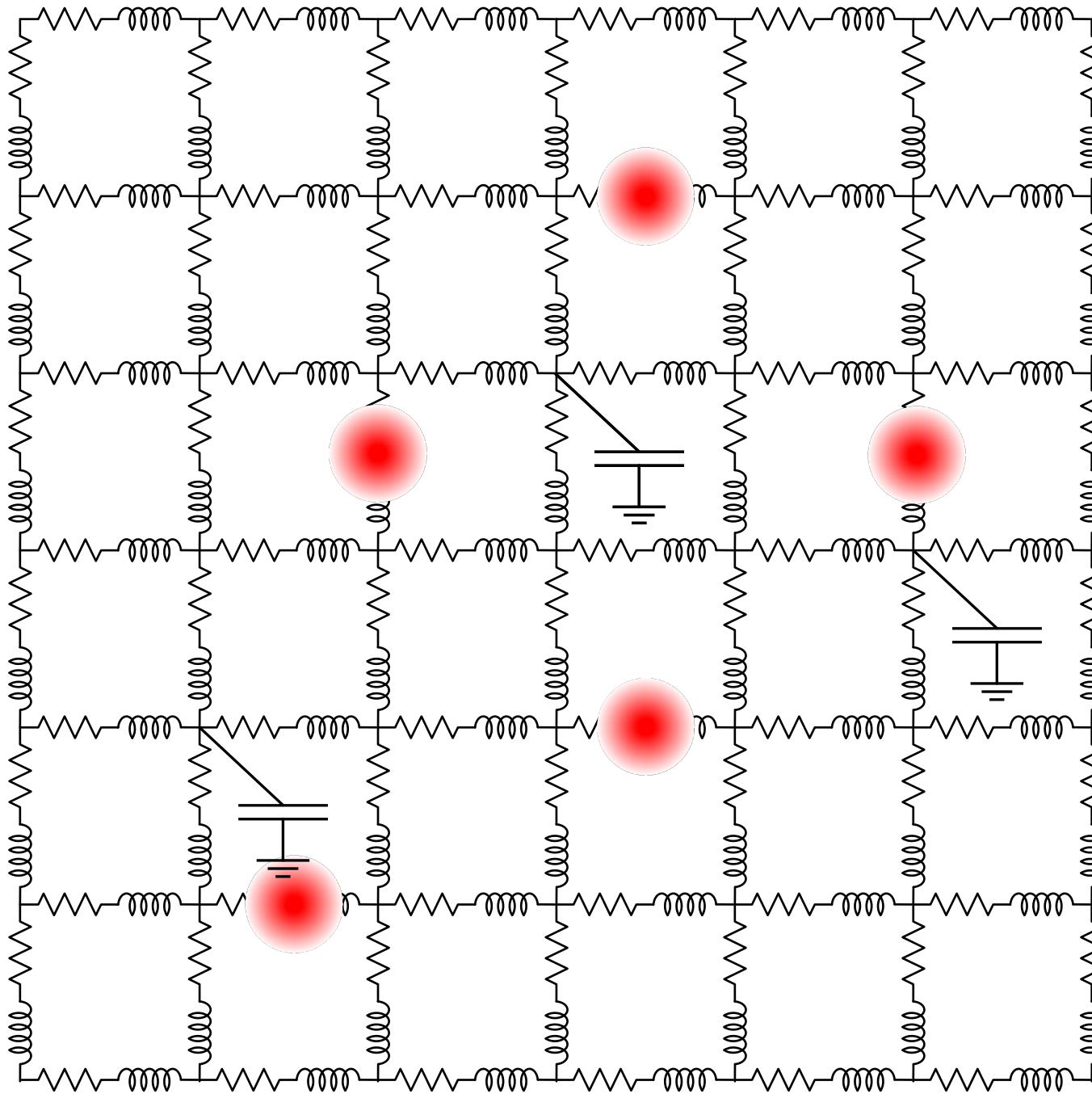
Iteration 1



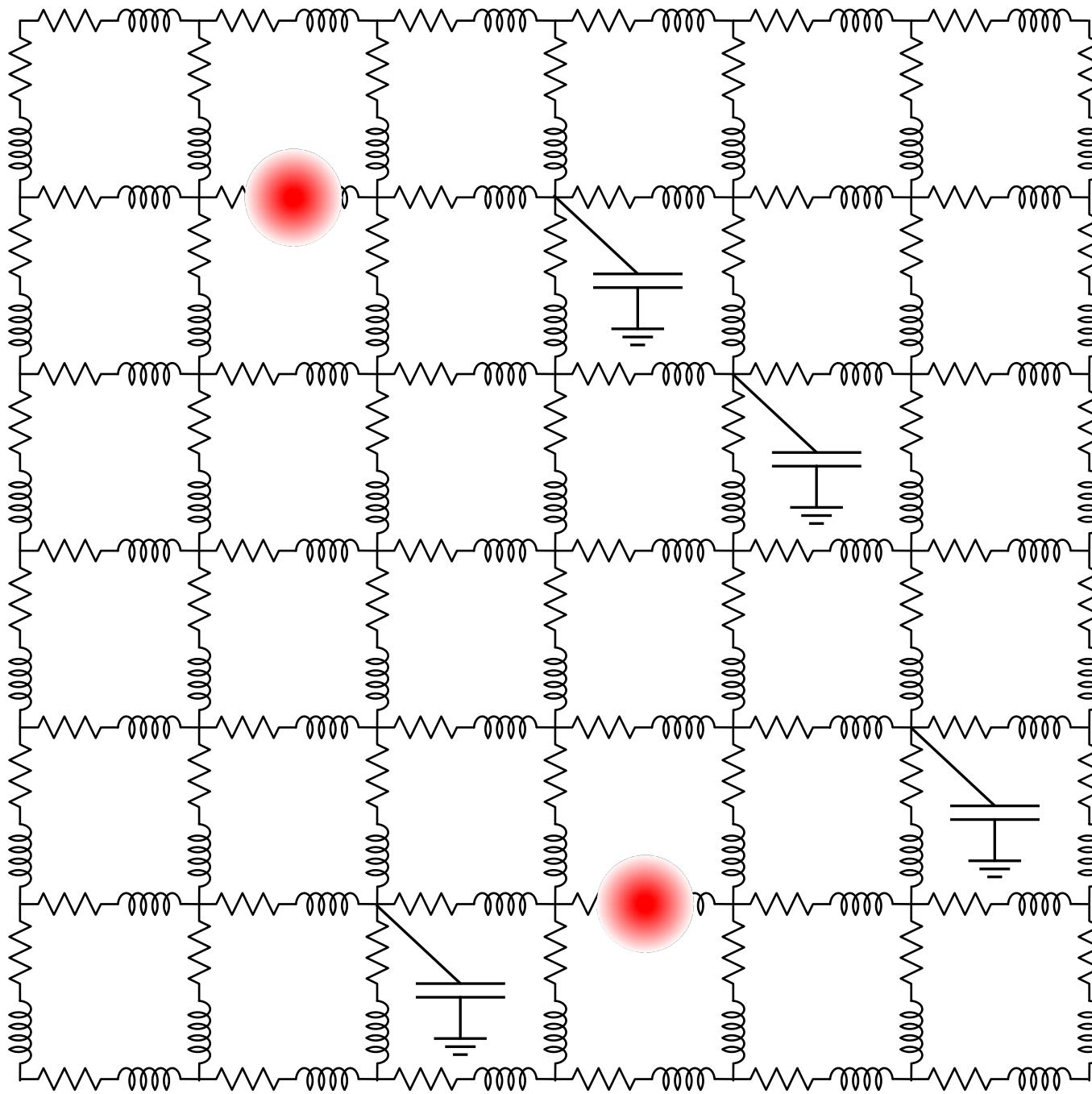
Iteration 2



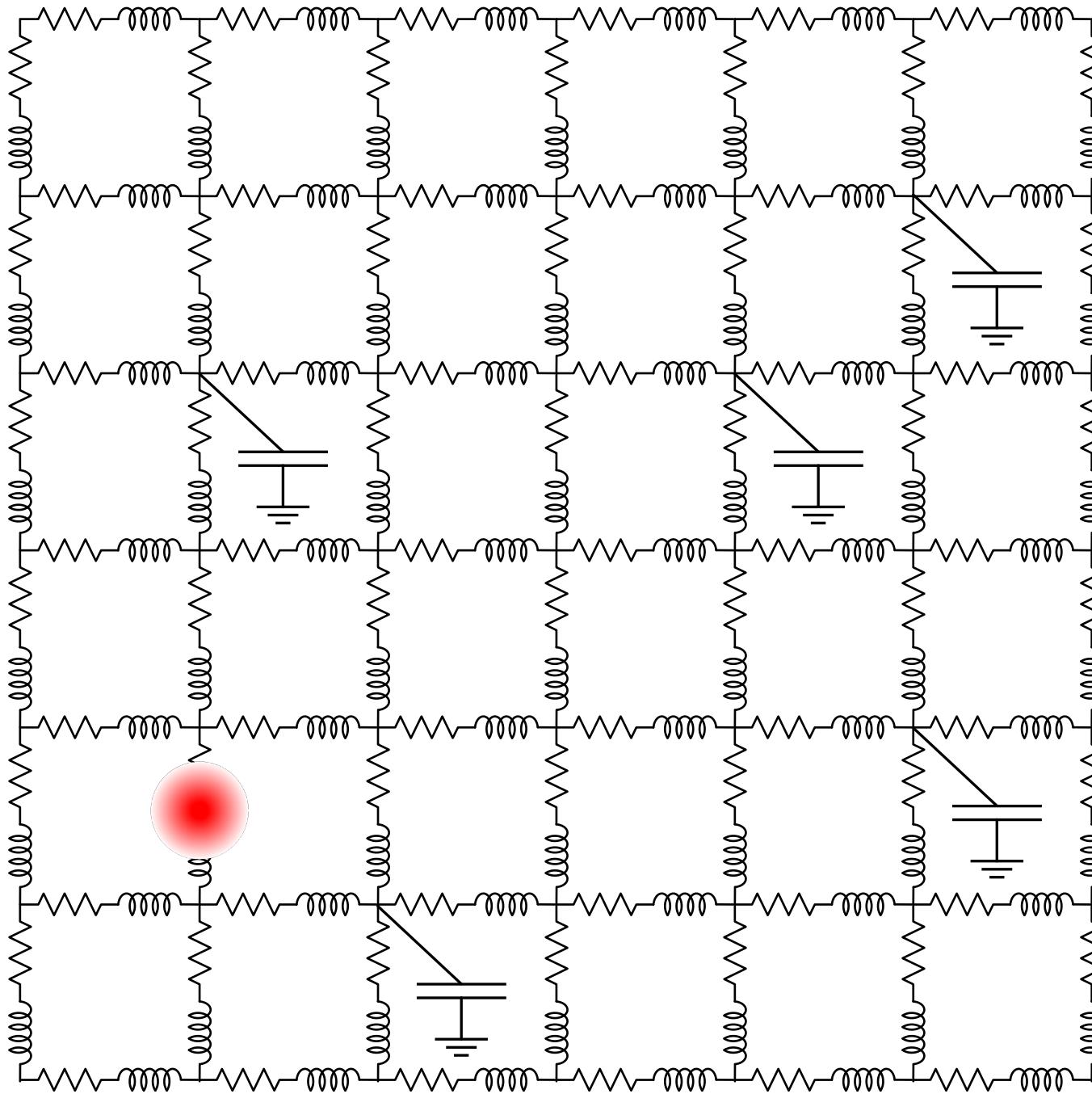
Iteration 3



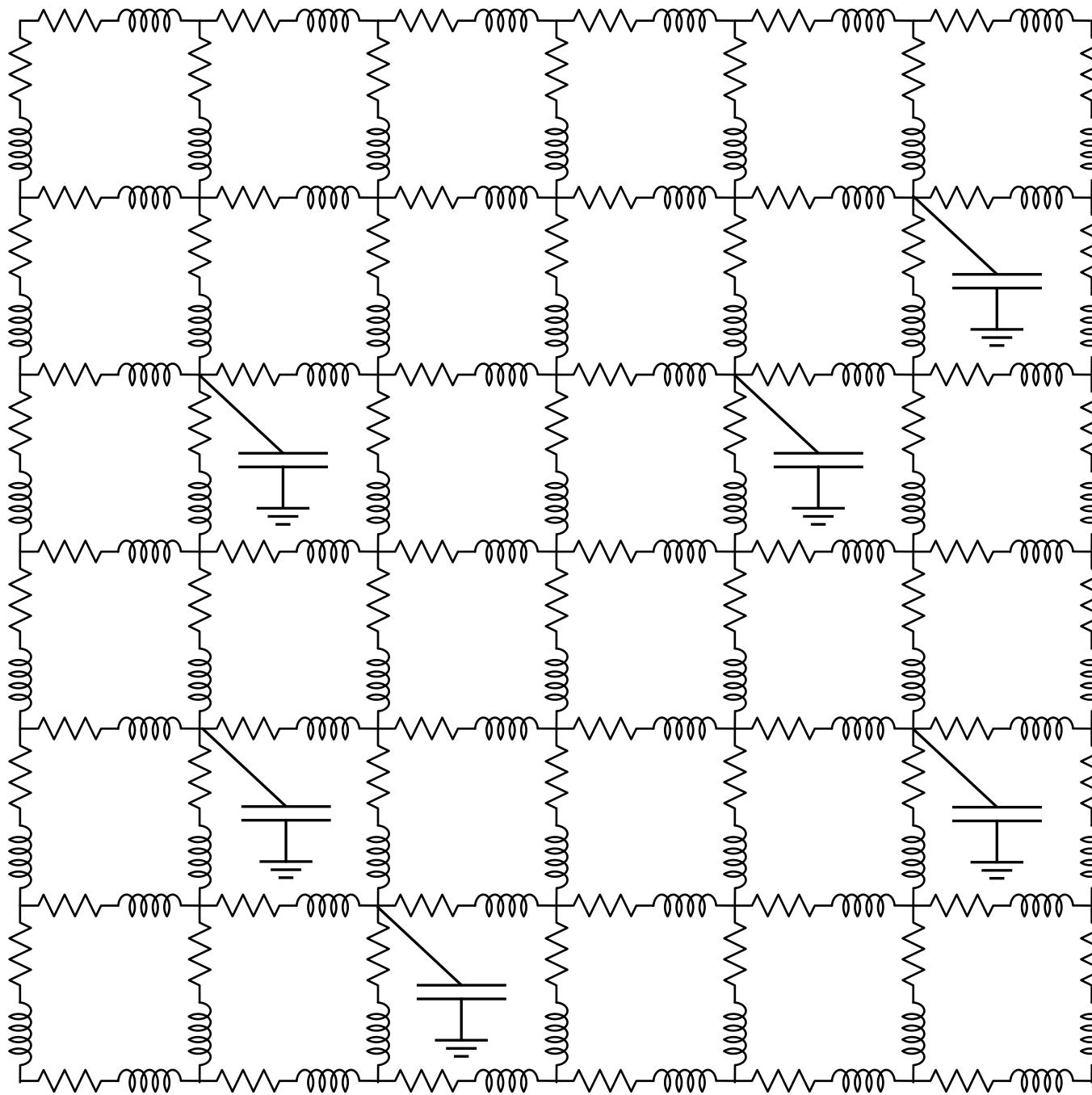
Iteration 4



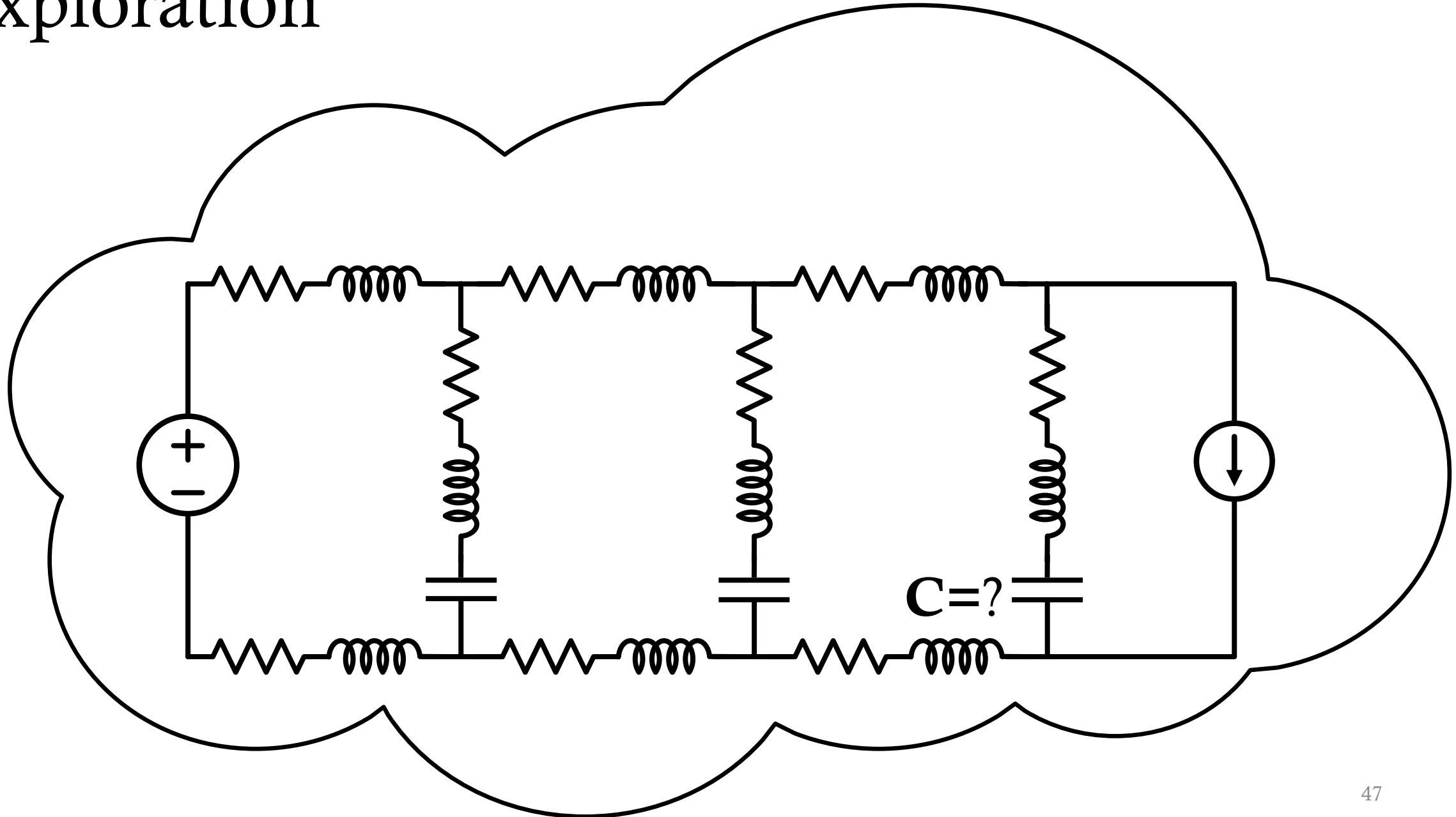
Iteration 5



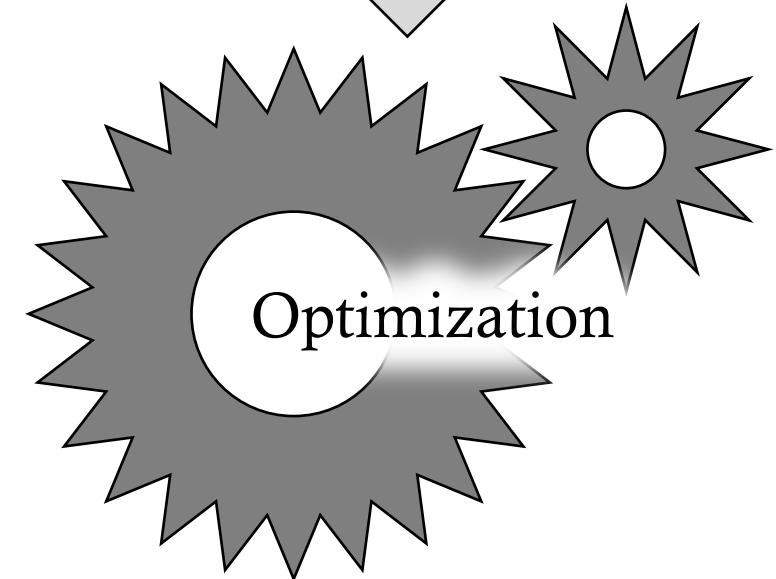
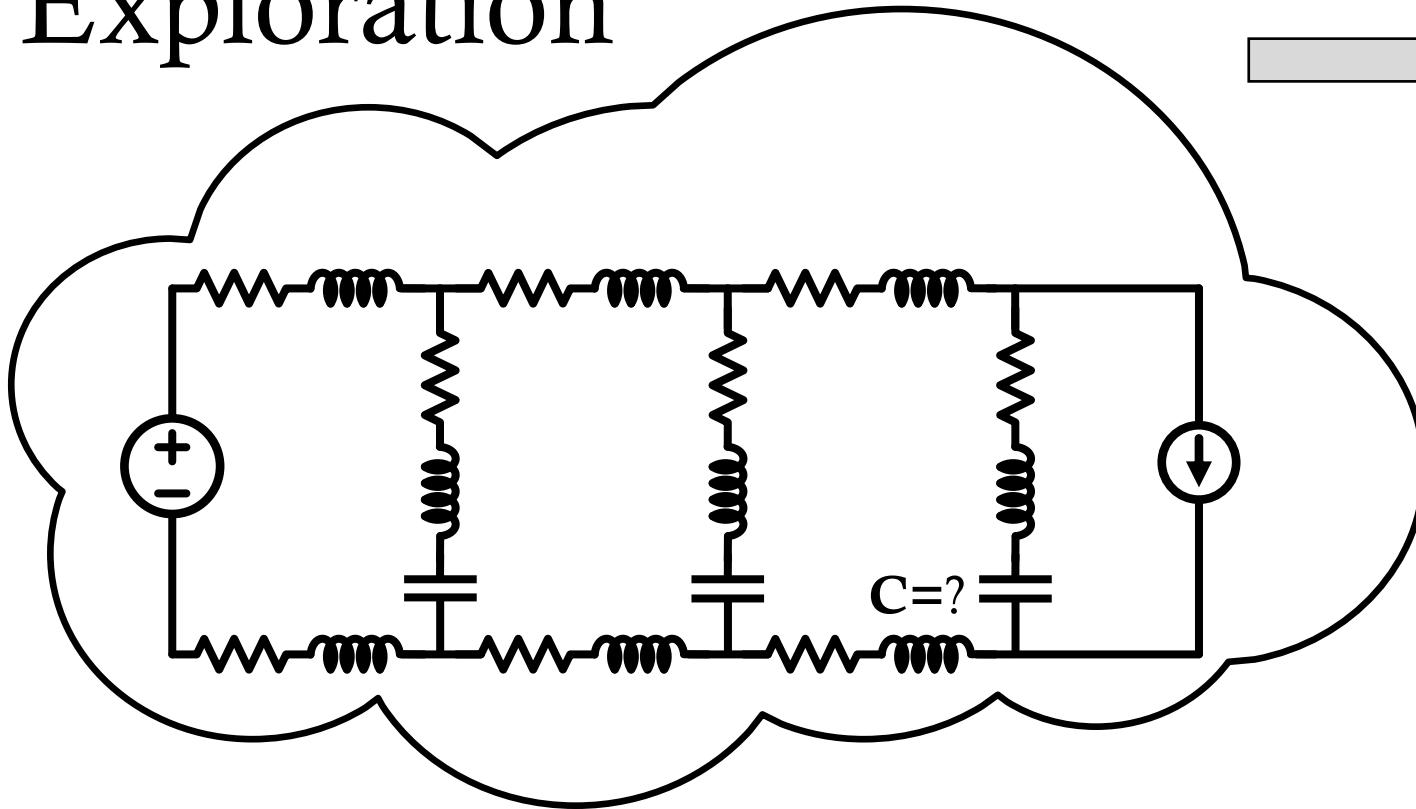
Iteration 6



# Exploration



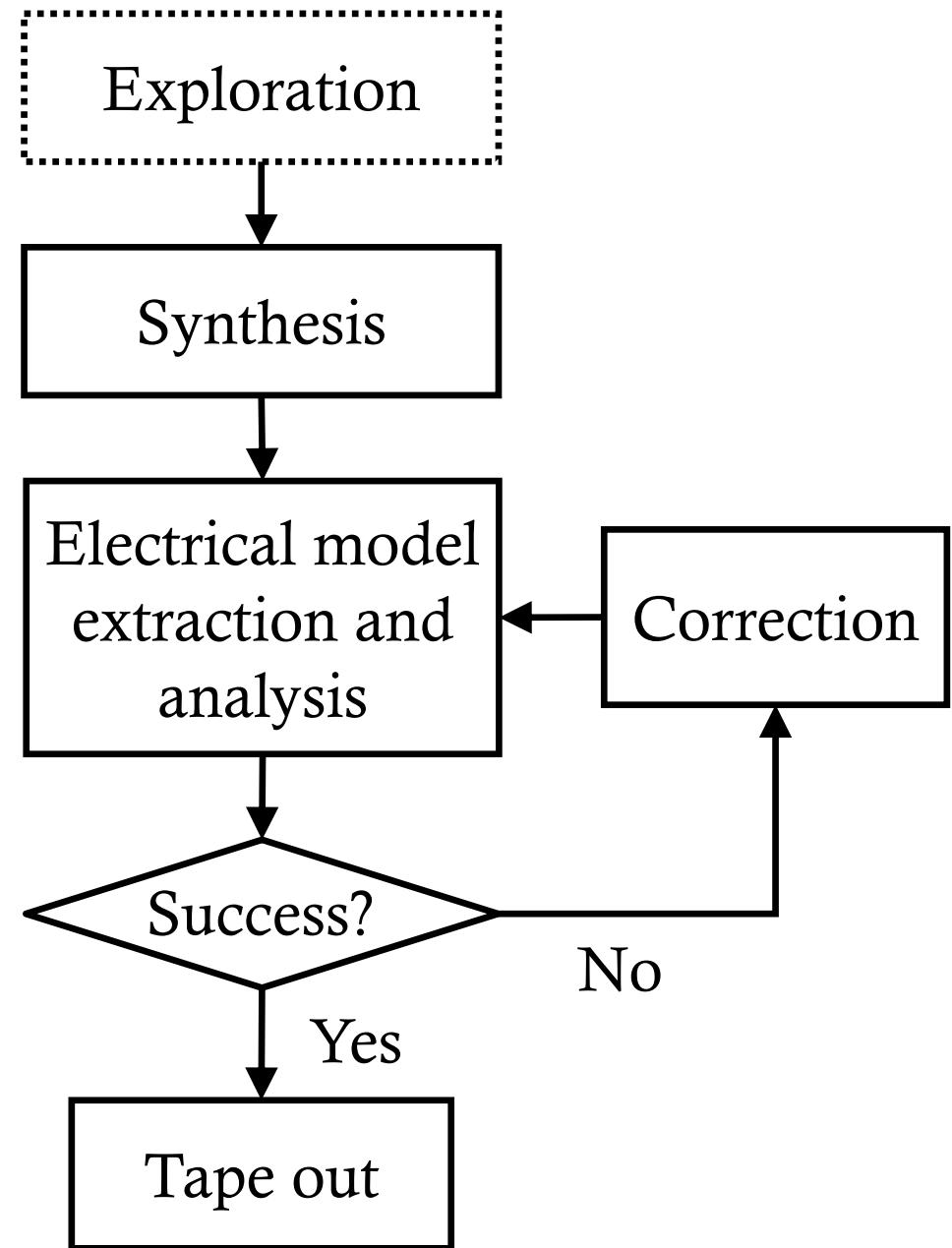
# Exploration



$C=500\text{nF}$

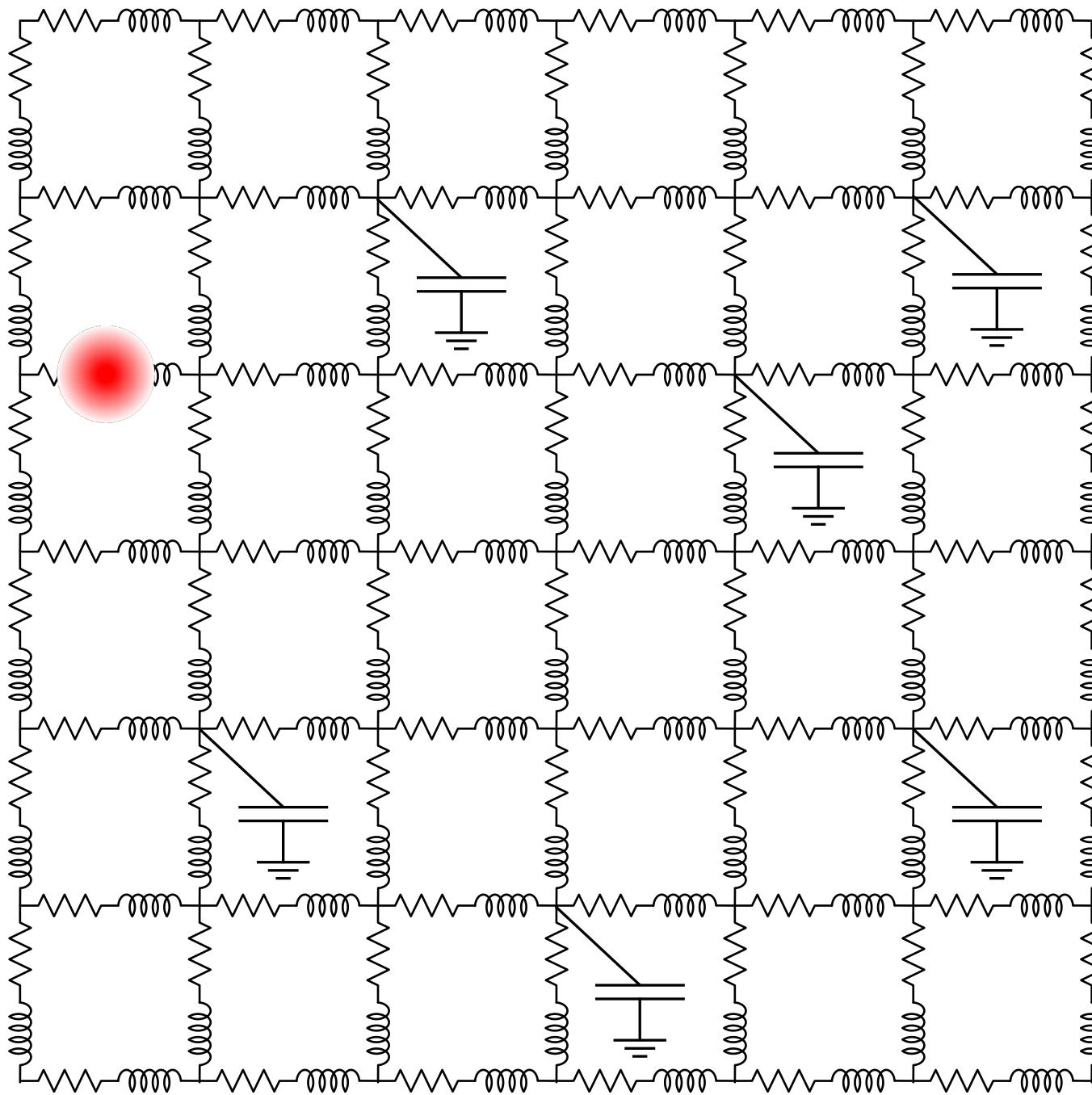
# Fewer Iterations with Exploration

- Bring initial system closer to optimum
- Fewer iterations to converge

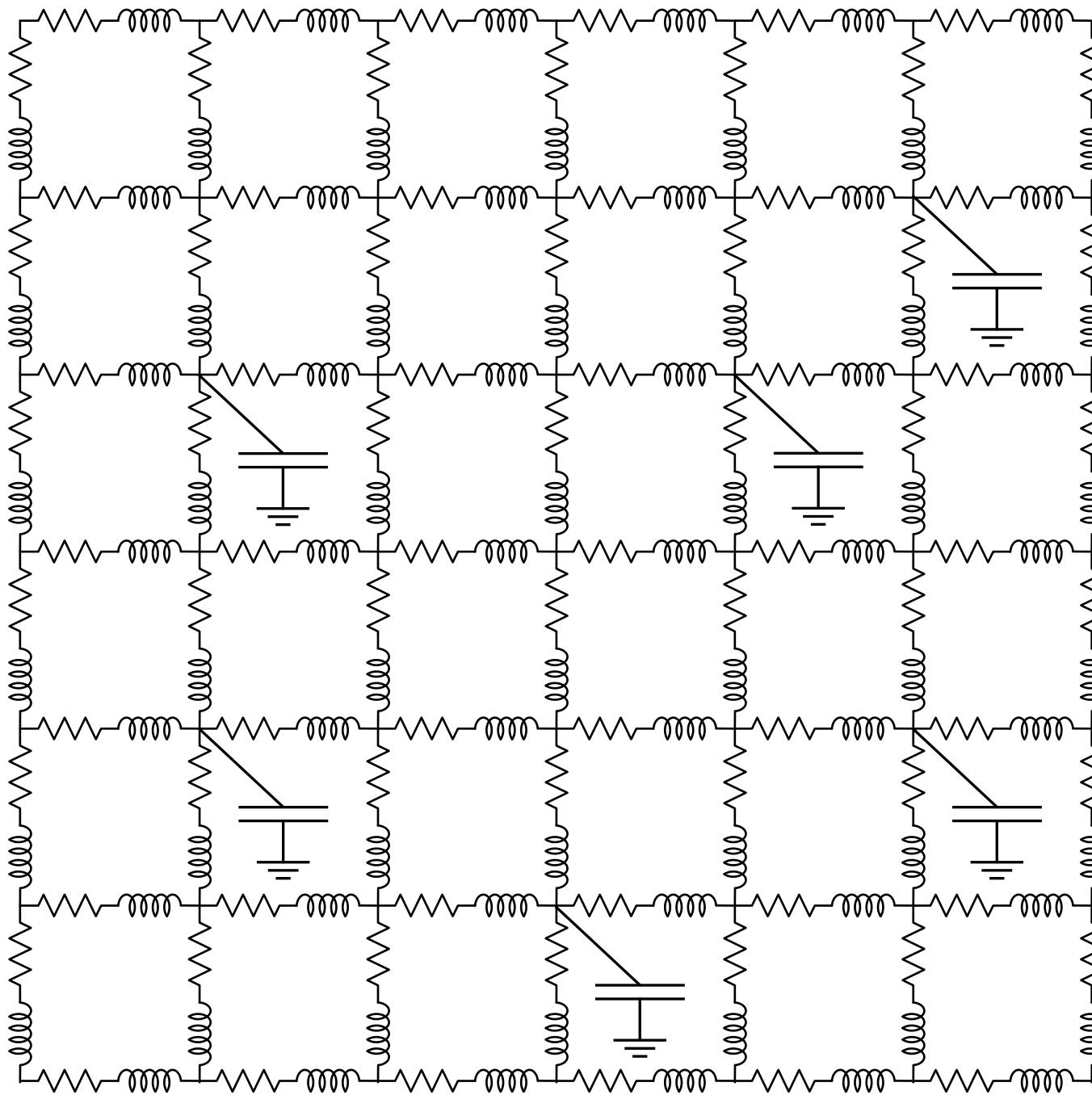


R. Bairamkulov, K. Xu, M. Popovich, J. S. Ochoa, V. Srinivas, and E. G. Friedman, "Power Delivery Exploration Methodology Based on Constrained Optimization," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 39, No. 9, pp. 1916 - 1924, September 2020.

Iteration 1

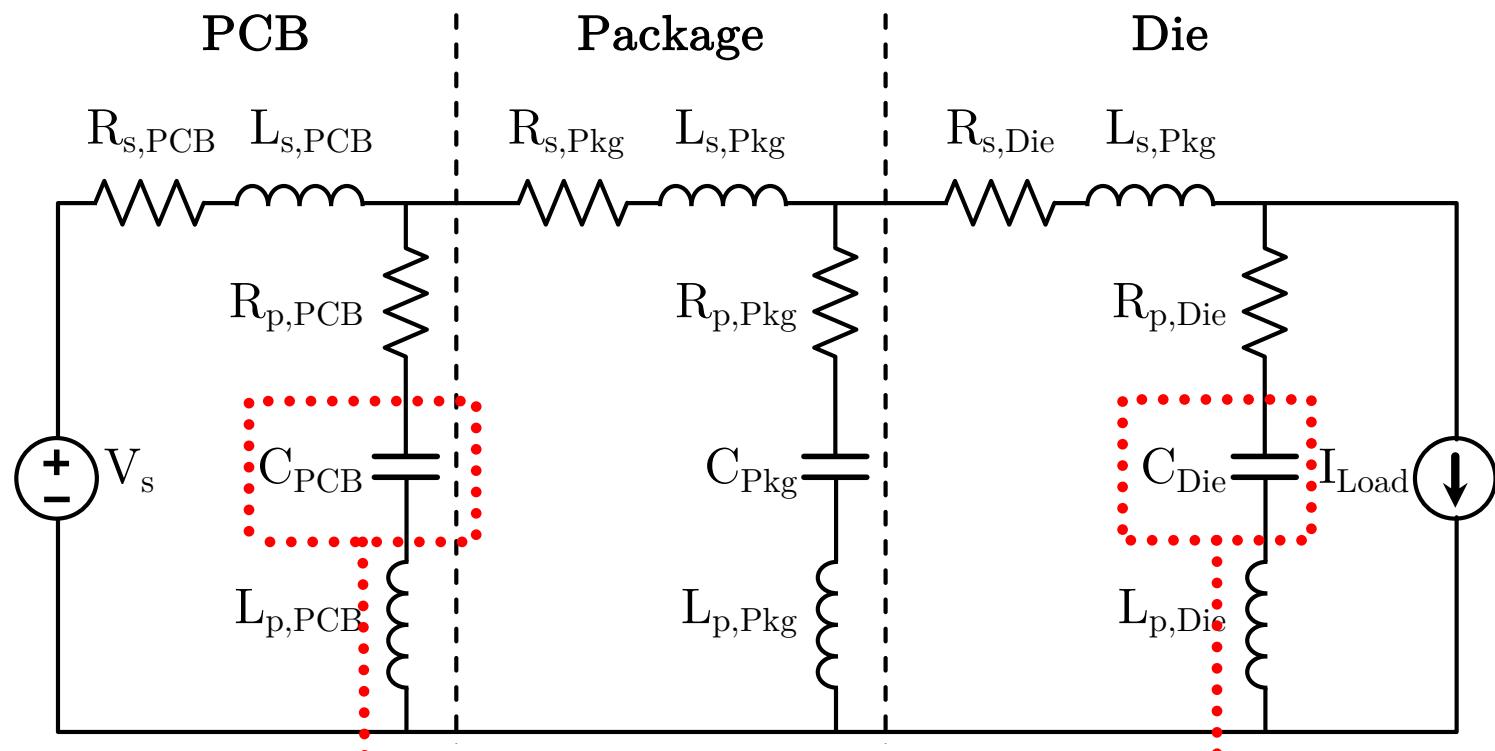


Iteration 2



# Case Study: Decoupling Capacitor Allocation

- Objective
  - Minimize the cost of decoupling capacitors
- Parameters
  - Capacitance at each level
  - Supply voltage
- Constraints
  - Minimum voltage
  - Maximum voltage
  - Power

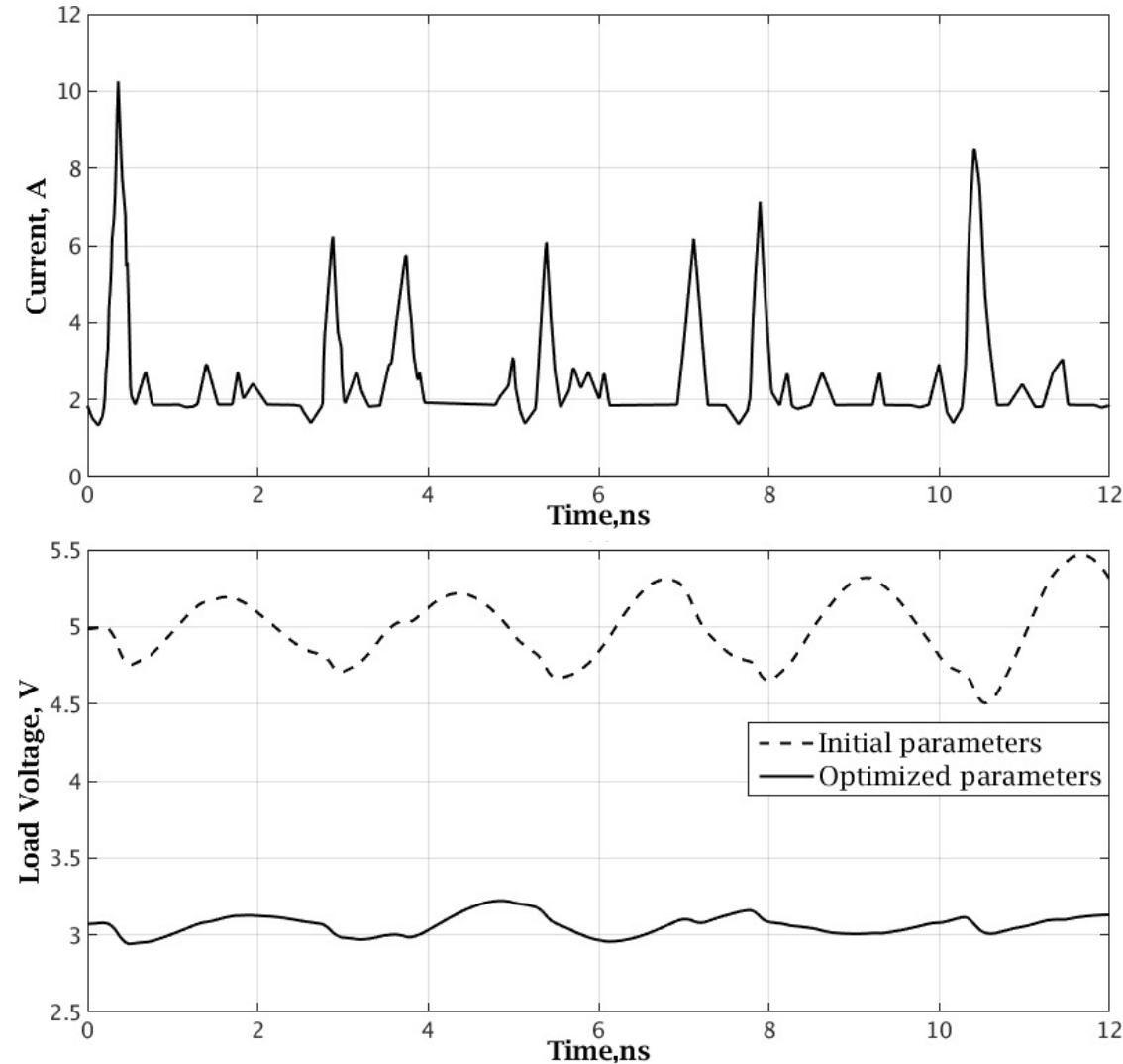


- Less effective
- Inexpensive
- Effective
- Expensive

# Decoupling Capacitor Allocation: Results

- Decoupling capacitor cost reduced by 15%
- Power reduced by 39%

	Lower bound	Upper bound	Initial value	Optimized value
Supply voltage	1.4 volts	10.0 volts	5.0 volts	3.09 volts
PCB decap	25.0 nF	10.0 $\mu$ F	5.00 $\mu$ F	2.71 $\mu$ F
Package decap	50.0 pF	100 nF	50.0 nF	9.77 nF
Die decap	2.00 pF	10.0 nF	5.00 nF	9.32 nF
Minimum load voltage	1.40 volts	—	2.96 volts	2.94 volts
Power dissipation	—	10.0 watts	10.6 watts	6.51 watts
Load voltage	—	10.0%	19.3%	9.07%
Normalized cost	—	—	0.317	0.270



**Research Effort  
since 2017**

**VLSI CONTINENT**

**CRYOGENIC  
OCEAN**

**QUANTUM  
COMPUTING**

**SUPERCONDUCTIVE  
UNION**



UNIVERSITY of  
**ROCHESTER**

**Qualcomm**

**PHYSICAL  
DESIGN**

**DEVICE  
PHYSICS**

**LOGIC  
SYNTHESIS**

**POWER  
DELIVERY**

# **SPROUT – Smart Power ROUTing Tool for Power Network Prototyping**

**SOFTWARE  
EMPIRE**

**ASSEMBLY CHANNEL**

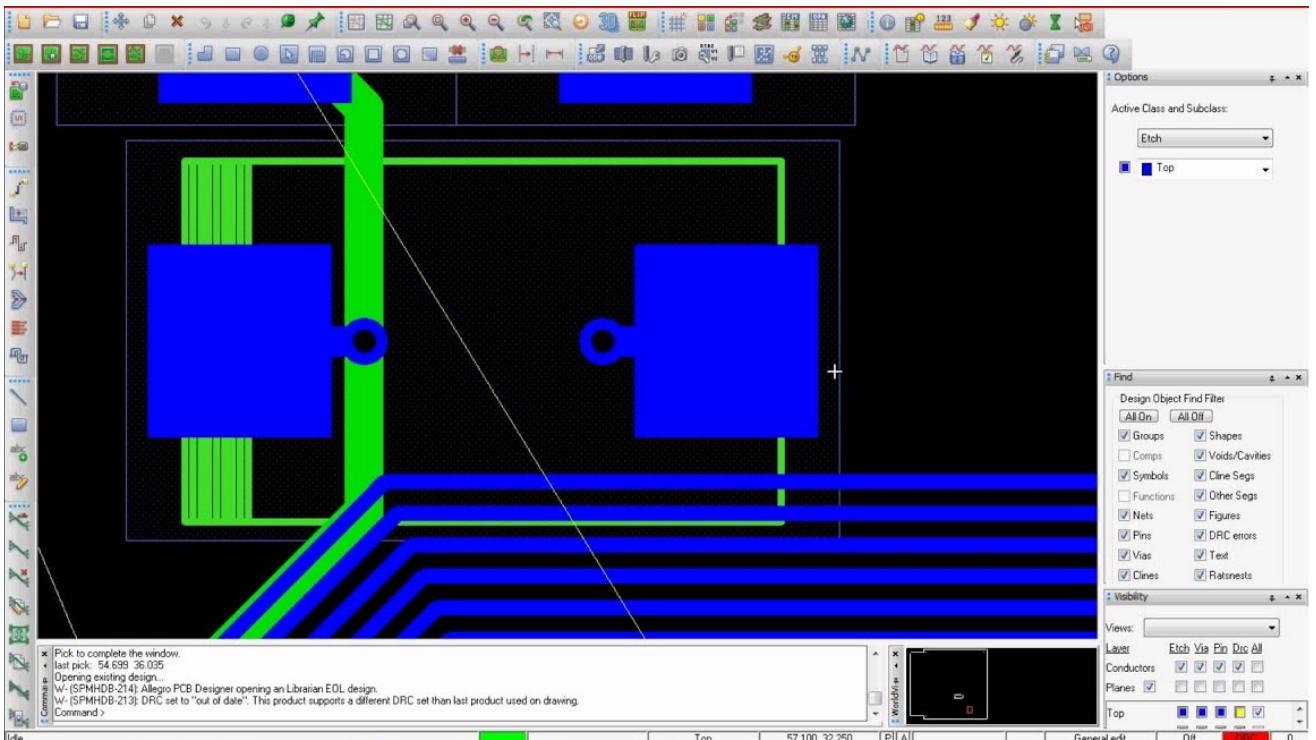
**NONVOLATILE  
SEA**

**MEMRISTOR  
ISLAND**

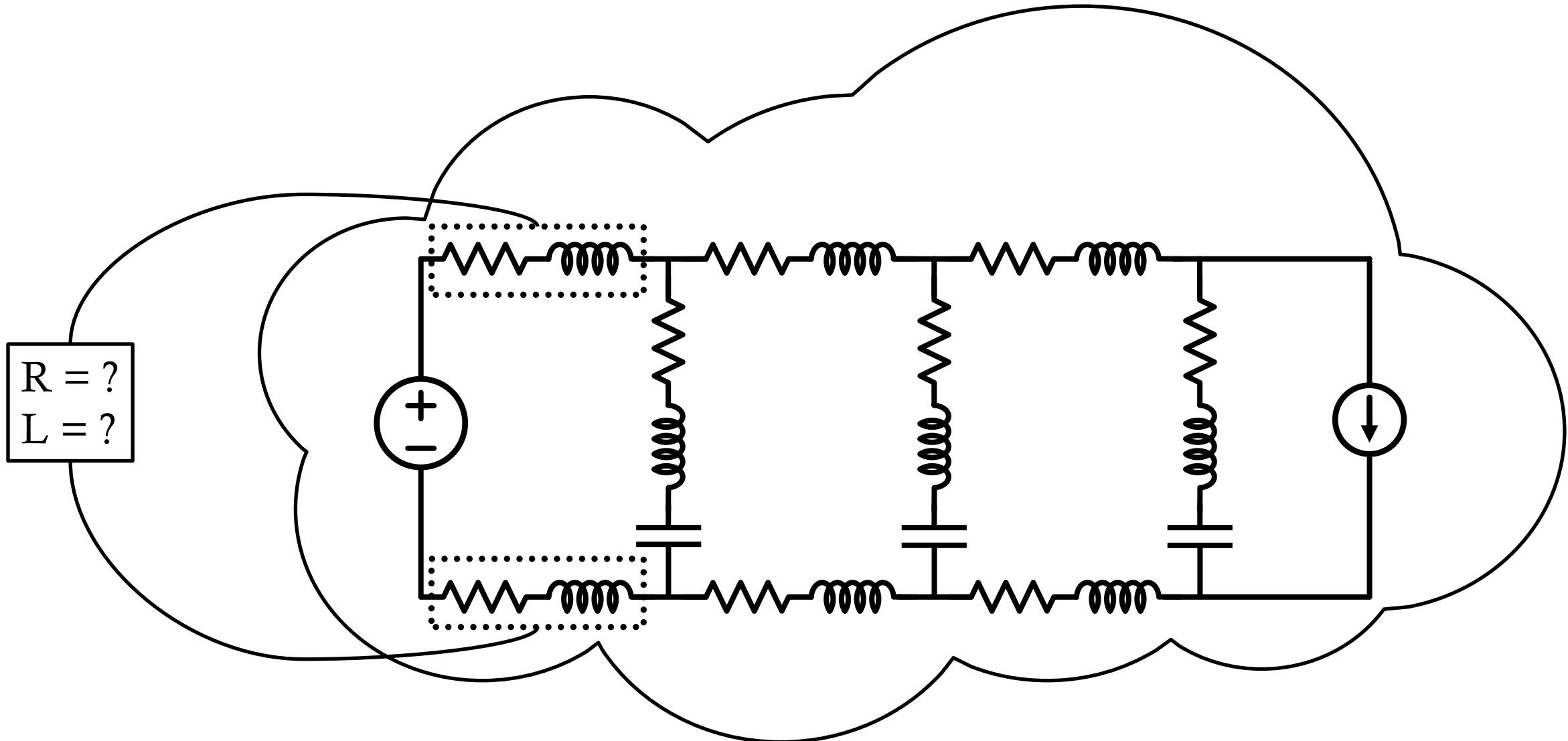
**SPINTRONIC  
ISLAND**

# Board-level Power Network Synthesis

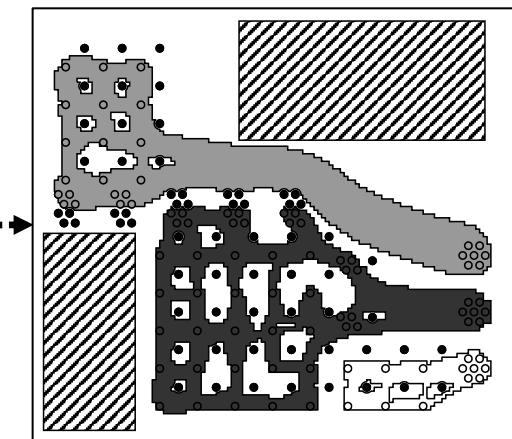
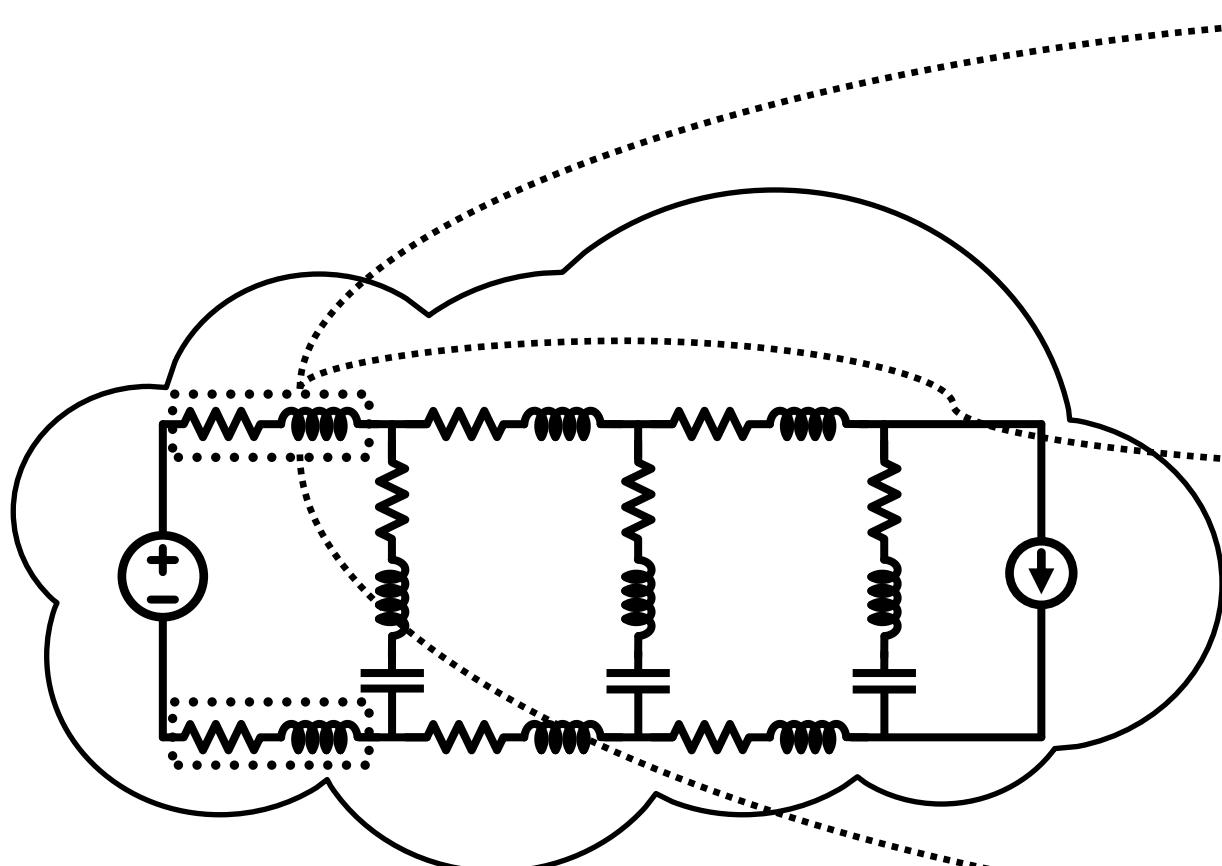
- Typical approach
  - Manual design
- Proposed approach
  - First fully automated board-level power network layout synthesis



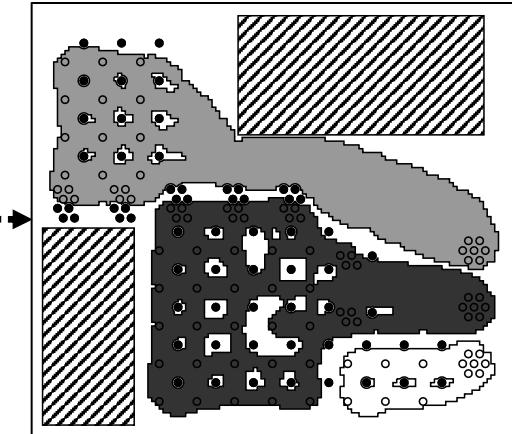
# Lack of Information



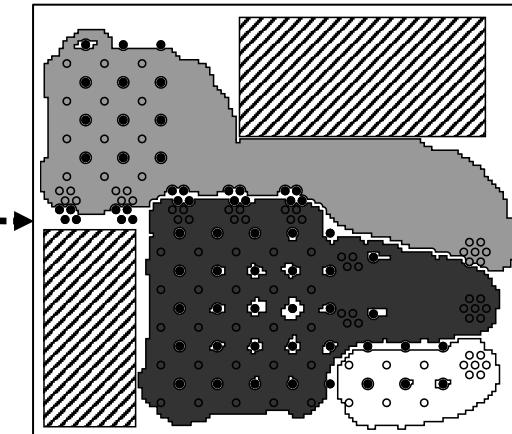
# Lack of Information



$R = 50 \text{ m}\Omega$   
 $L = 250 \text{ pH}$



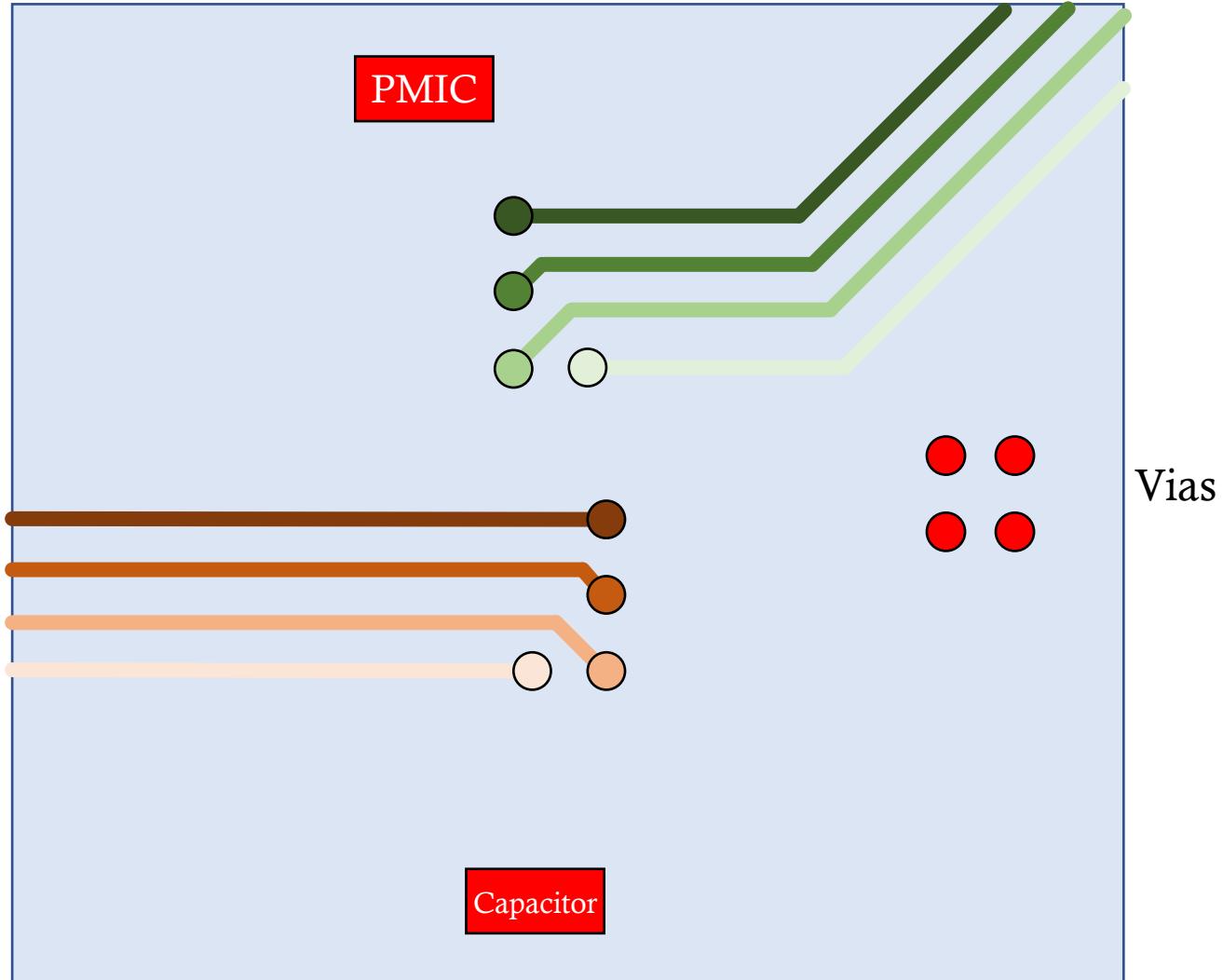
$R = 30 \text{ m}\Omega$   
 $L = 130 \text{ pH}$



$R = 20 \text{ m}\Omega$   
 $L = 110 \text{ pH}$

# Prototyping Process

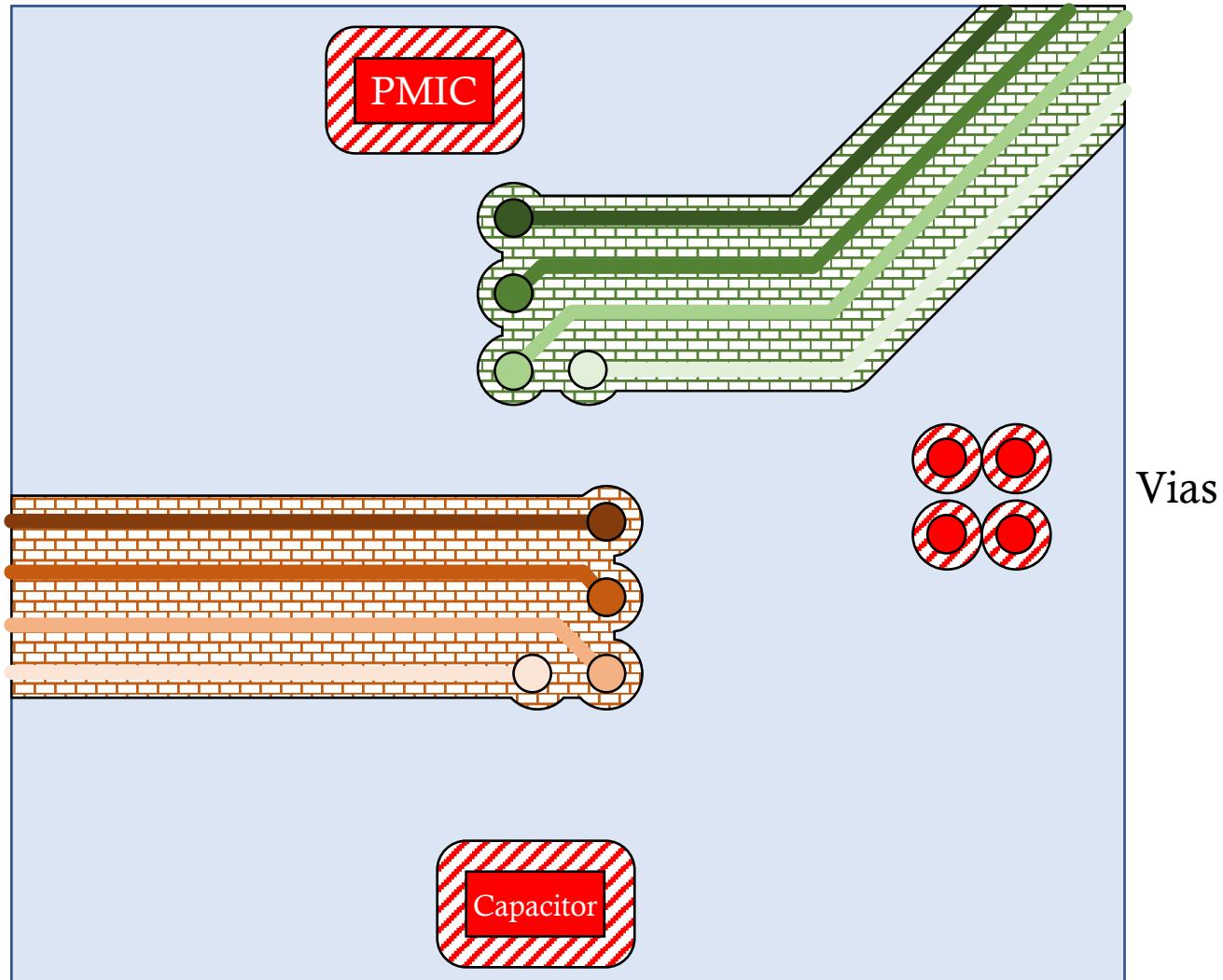
- Power network prototype generated in five steps
  - Available space



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, “SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

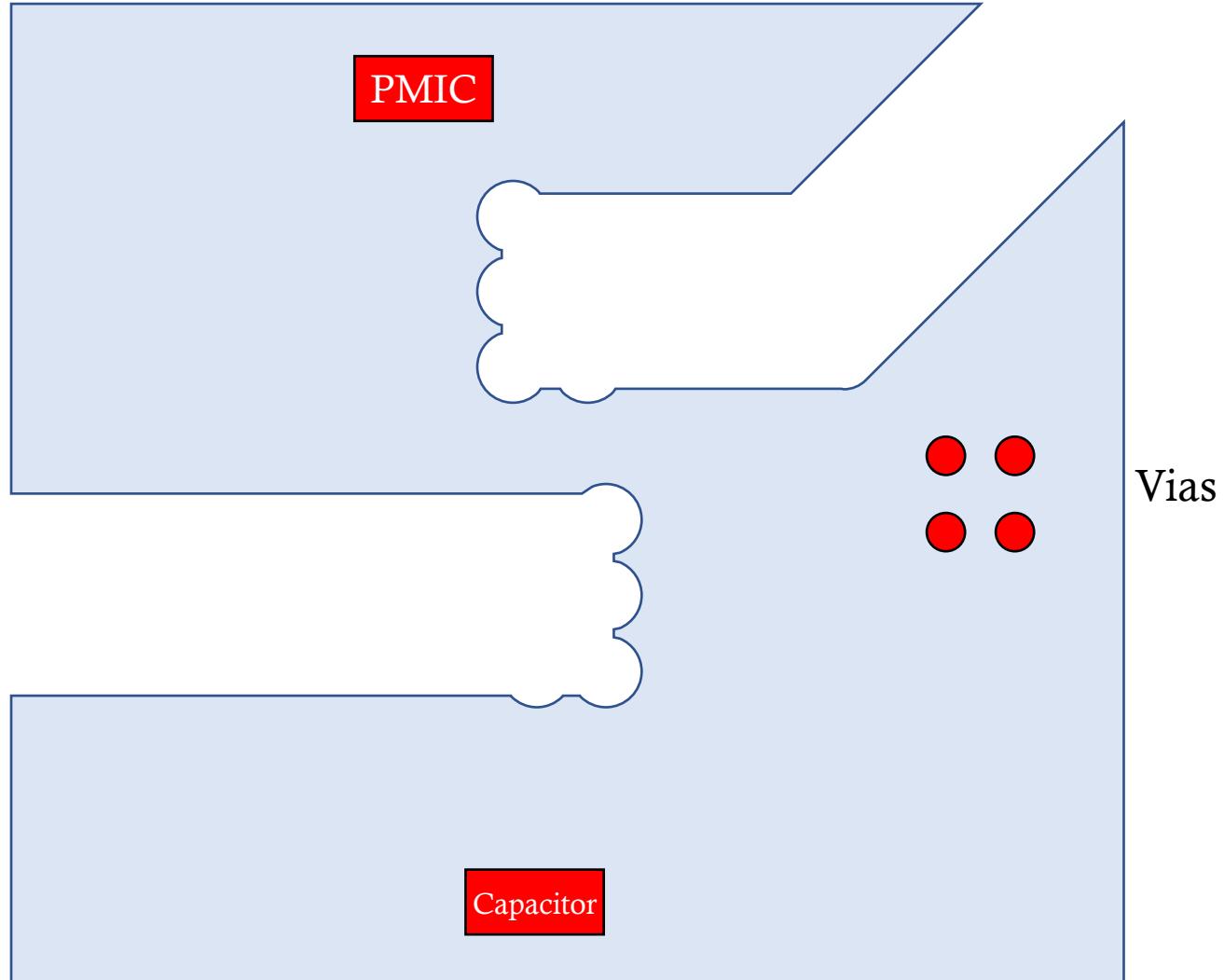
- Power network prototype generated in five steps
  - Available space



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

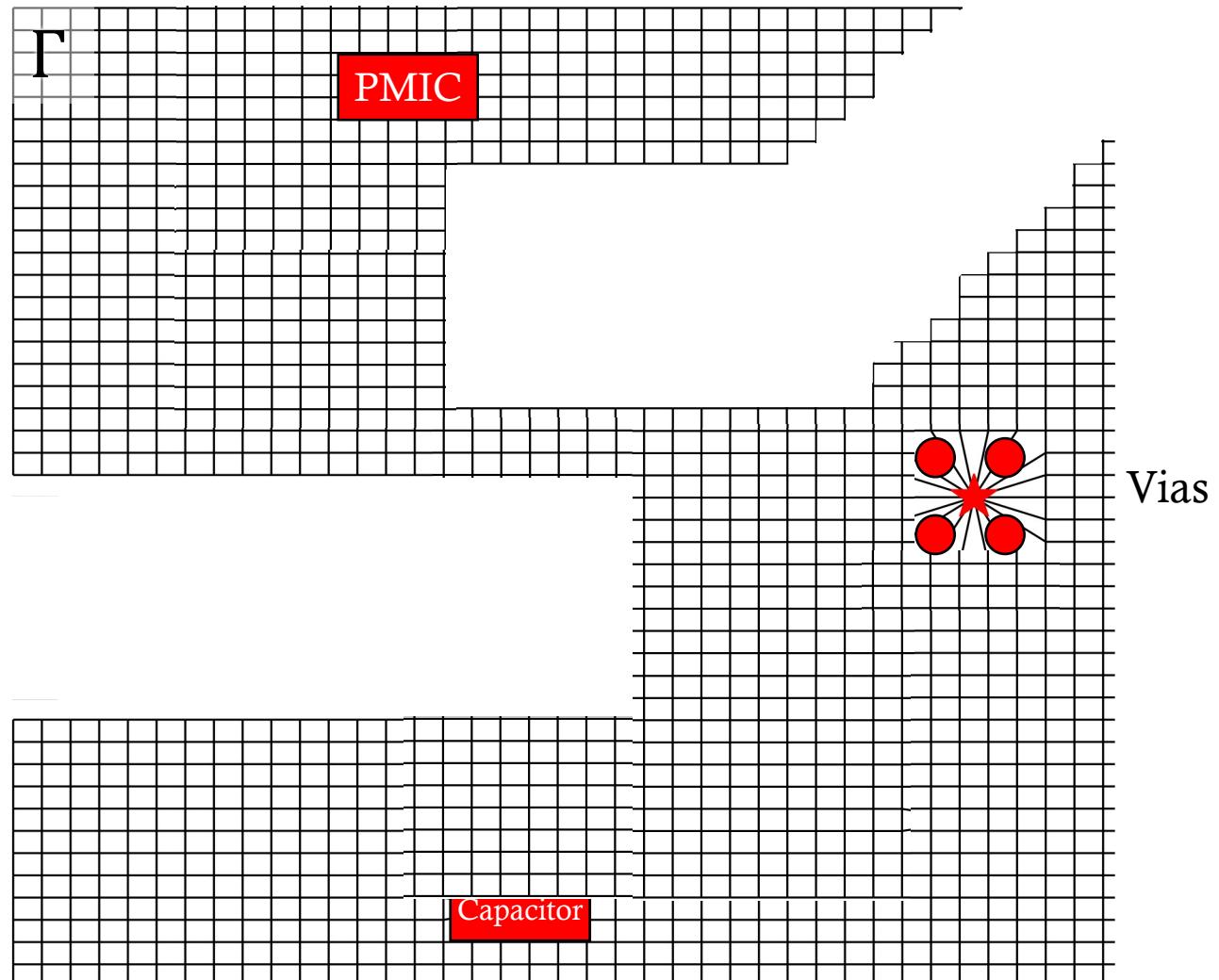
- Power network prototype generated in five steps
  - Available space



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

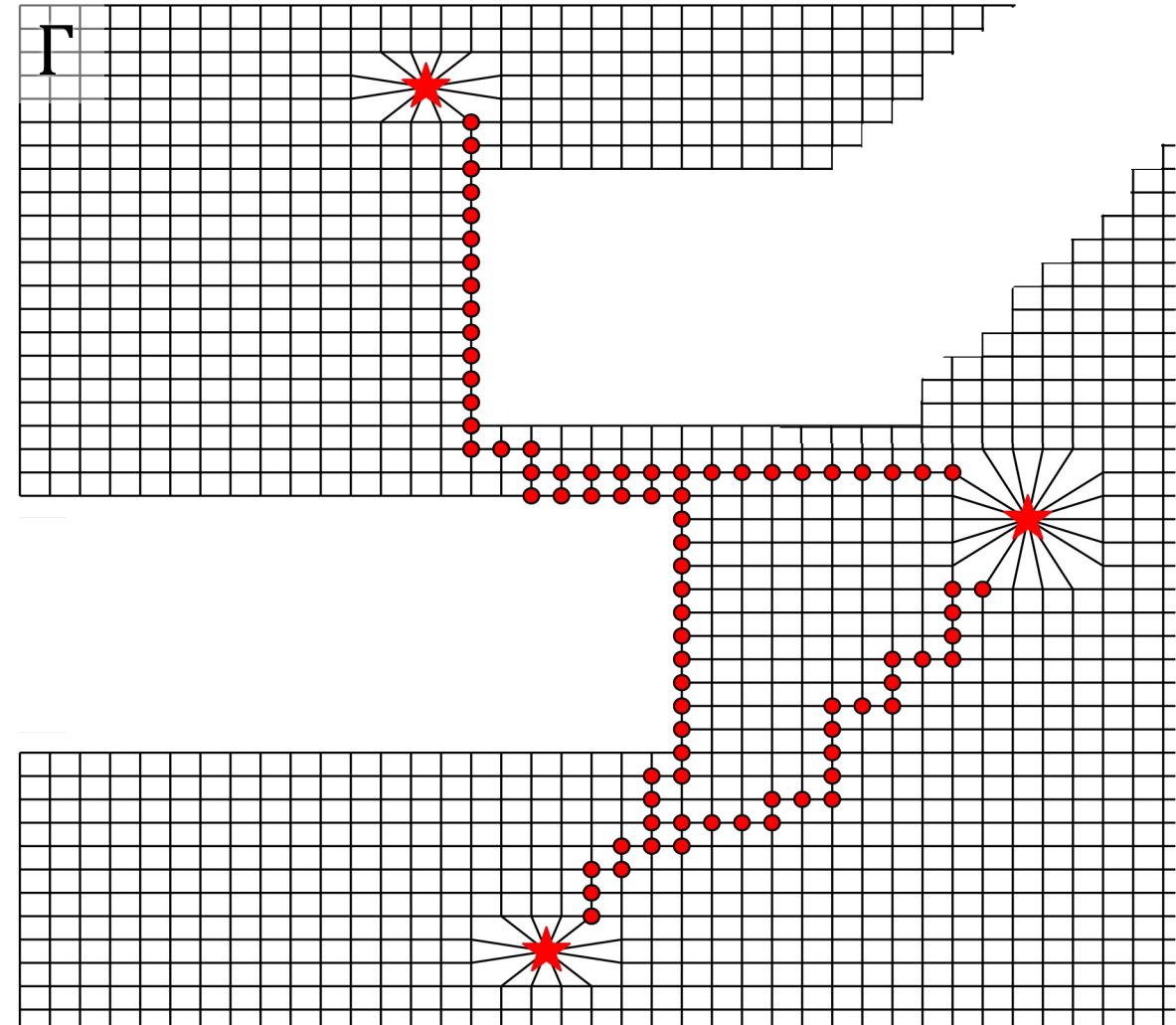
- Power network prototype generated in five steps
  - Available space
  - Seed



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, “SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

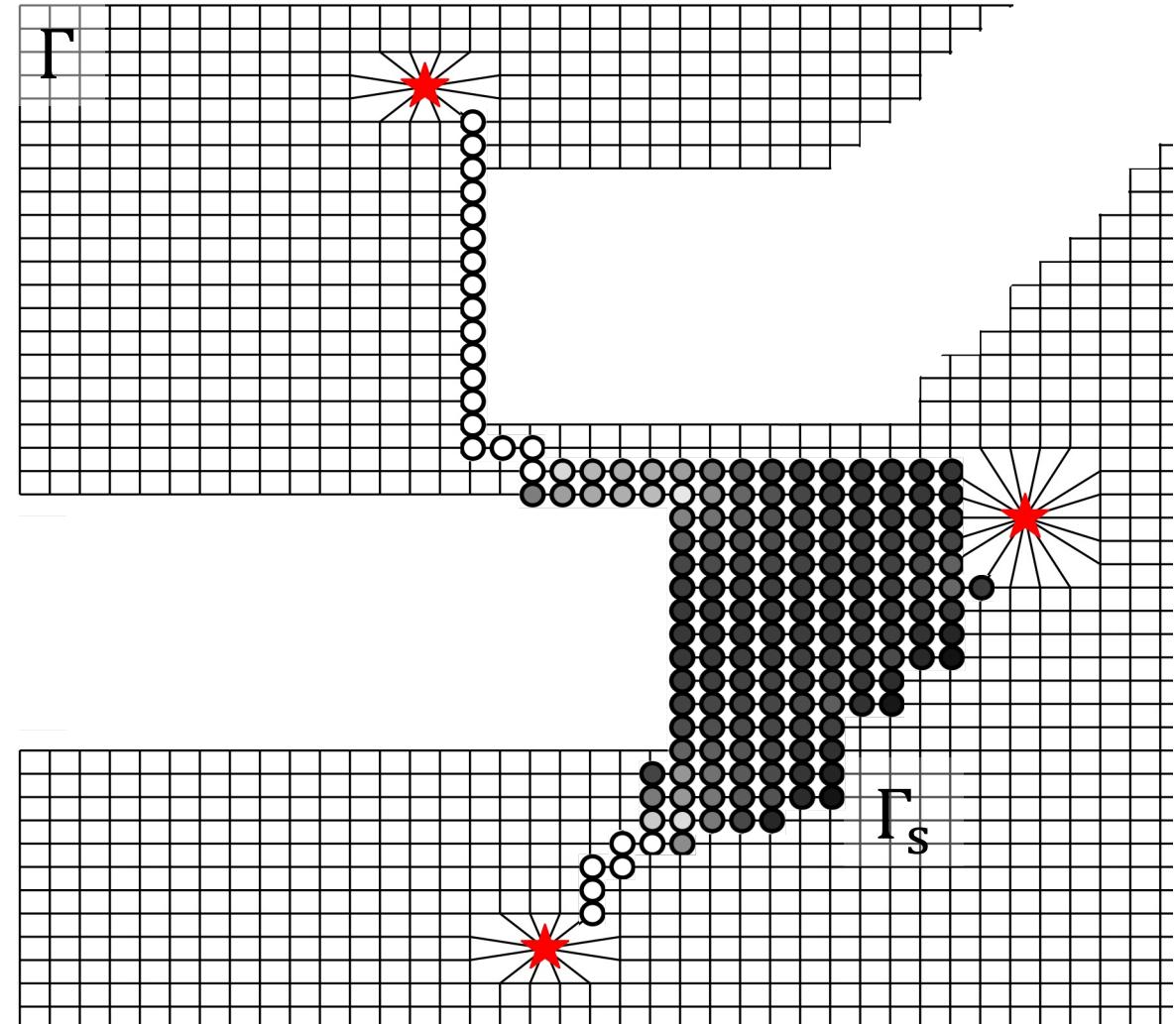
- Power network prototype generated in five steps
  - Available space
  - Seed



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

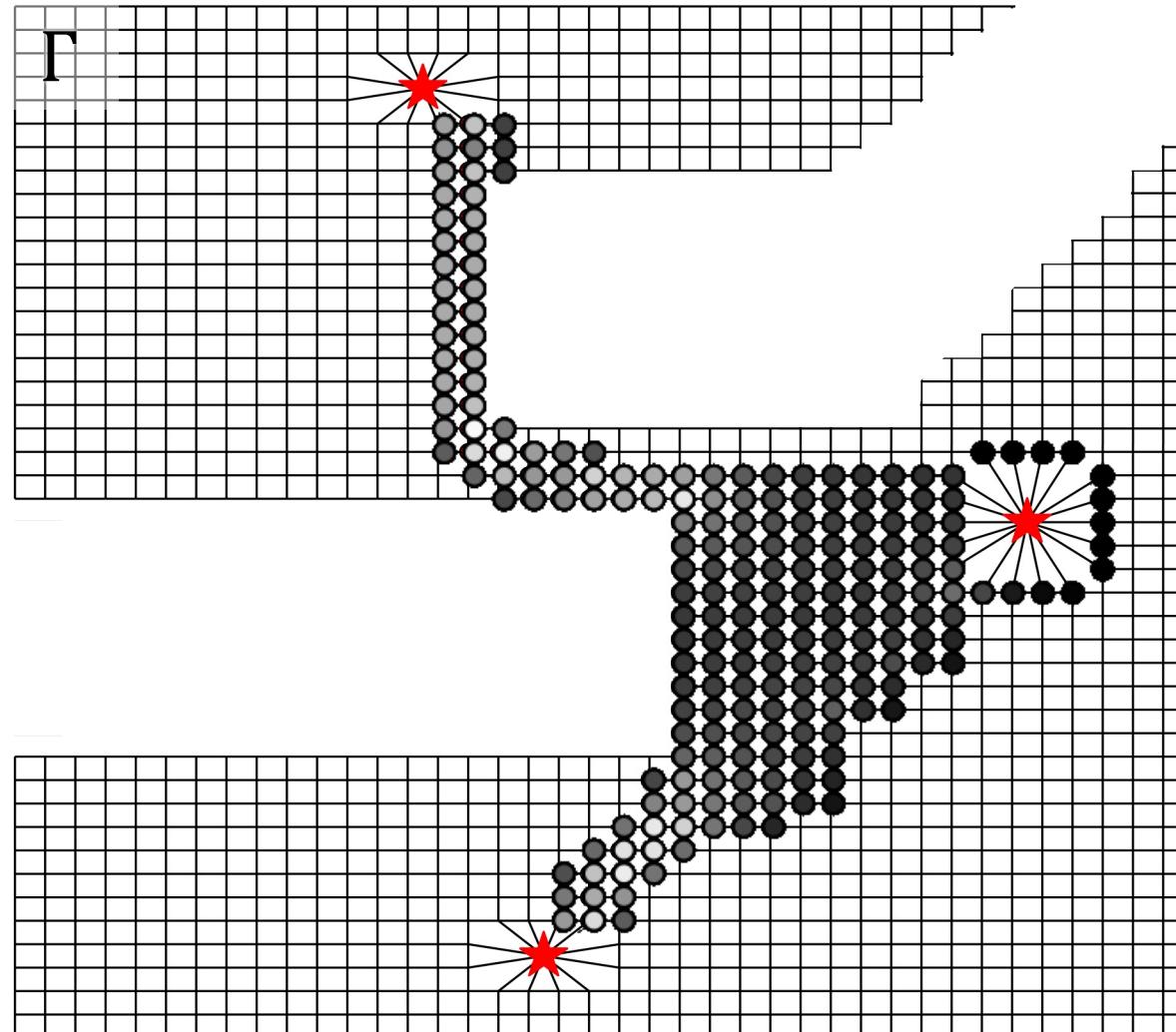
- Power network prototype generated in five steps
  - Available space
  - Seed
  - **Growth**



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

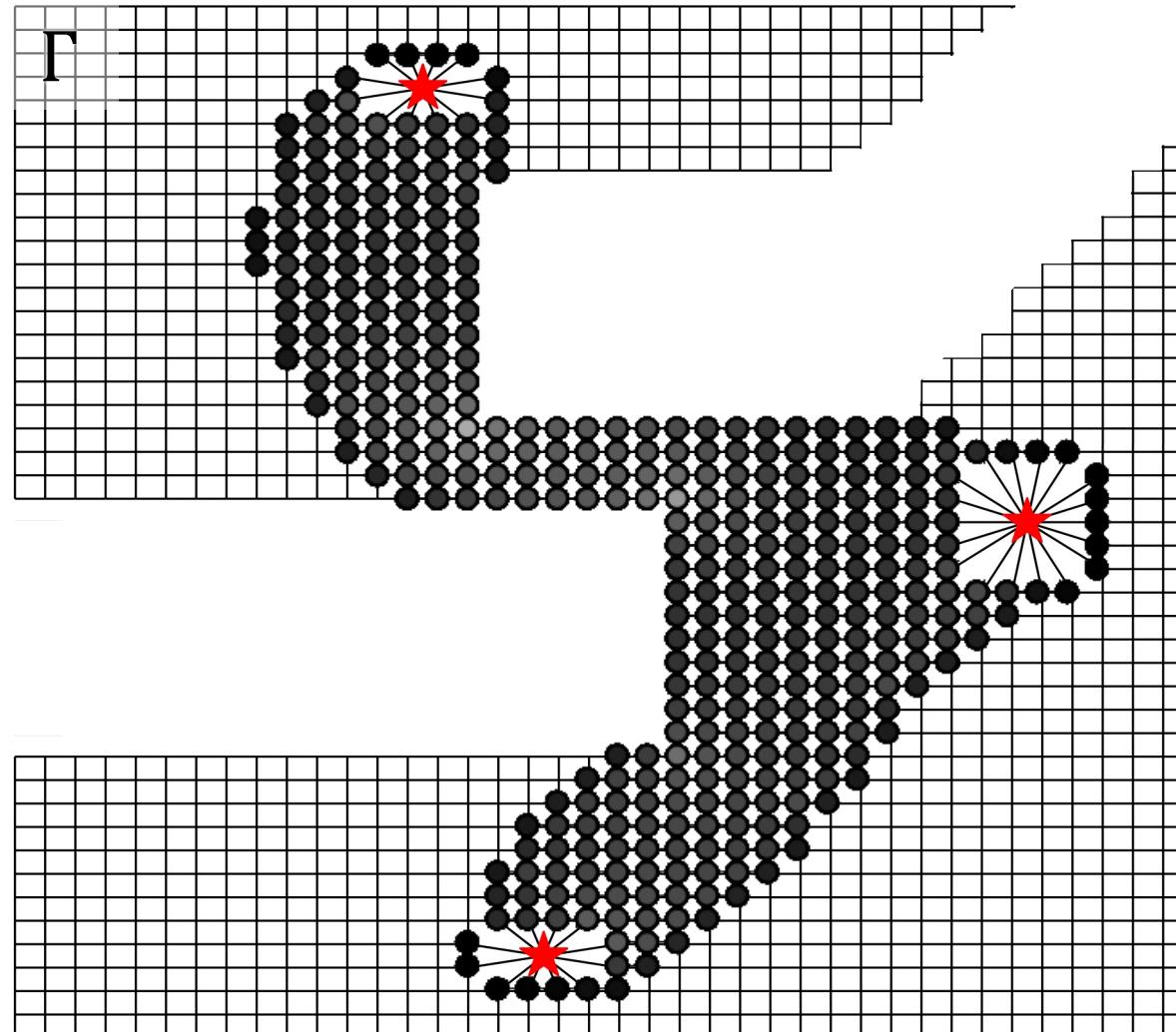
- Power network prototype generated in five steps
  - Available space
  - Seed
  - **Growth**



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

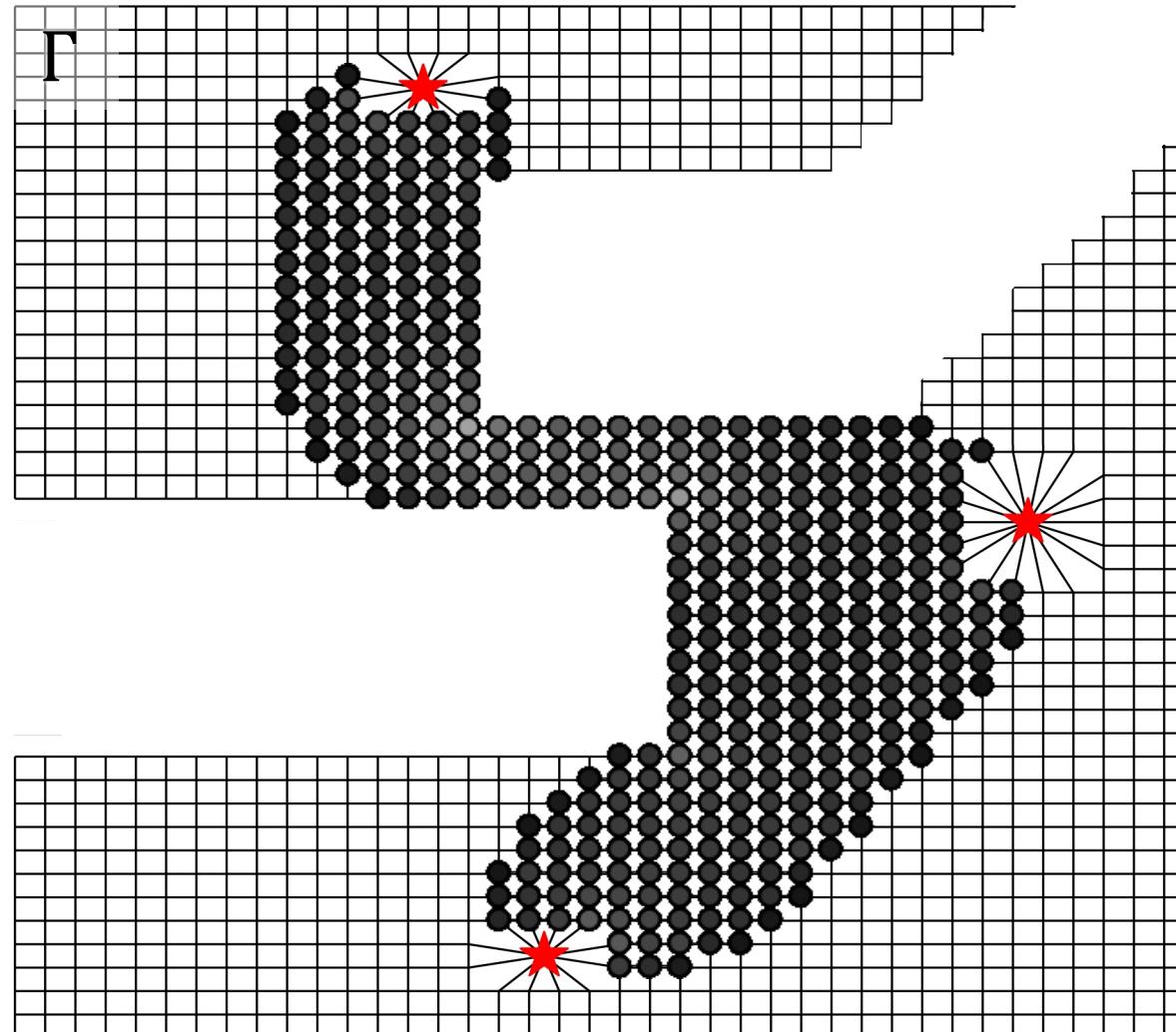
- Power network prototype generated in five steps
  - Available space
  - Seed
  - **Growth**



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

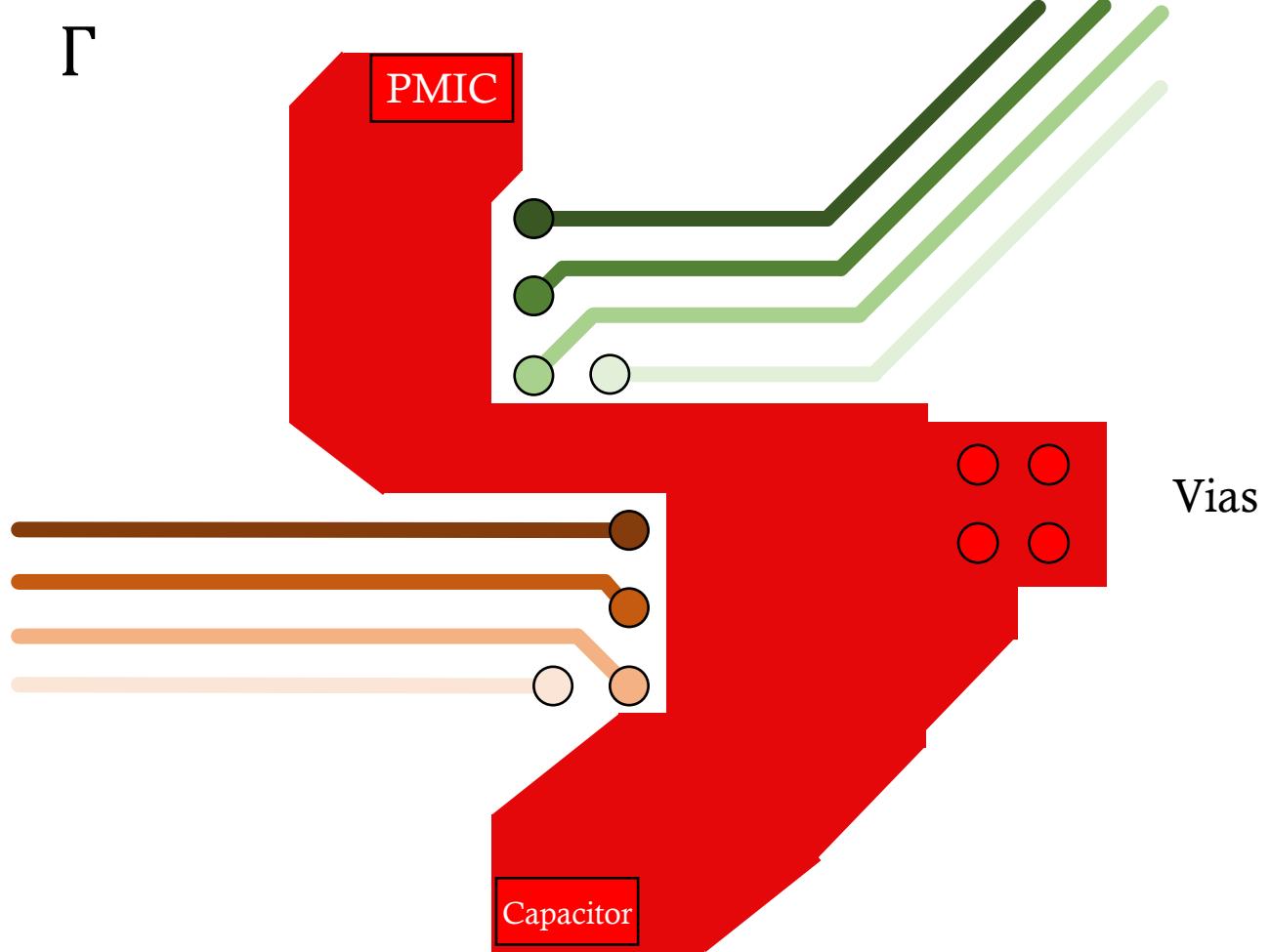
- Power network prototype generated in five steps
  - Available space
  - Seed
  - Growth
  - **Refinement**



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Prototyping Process

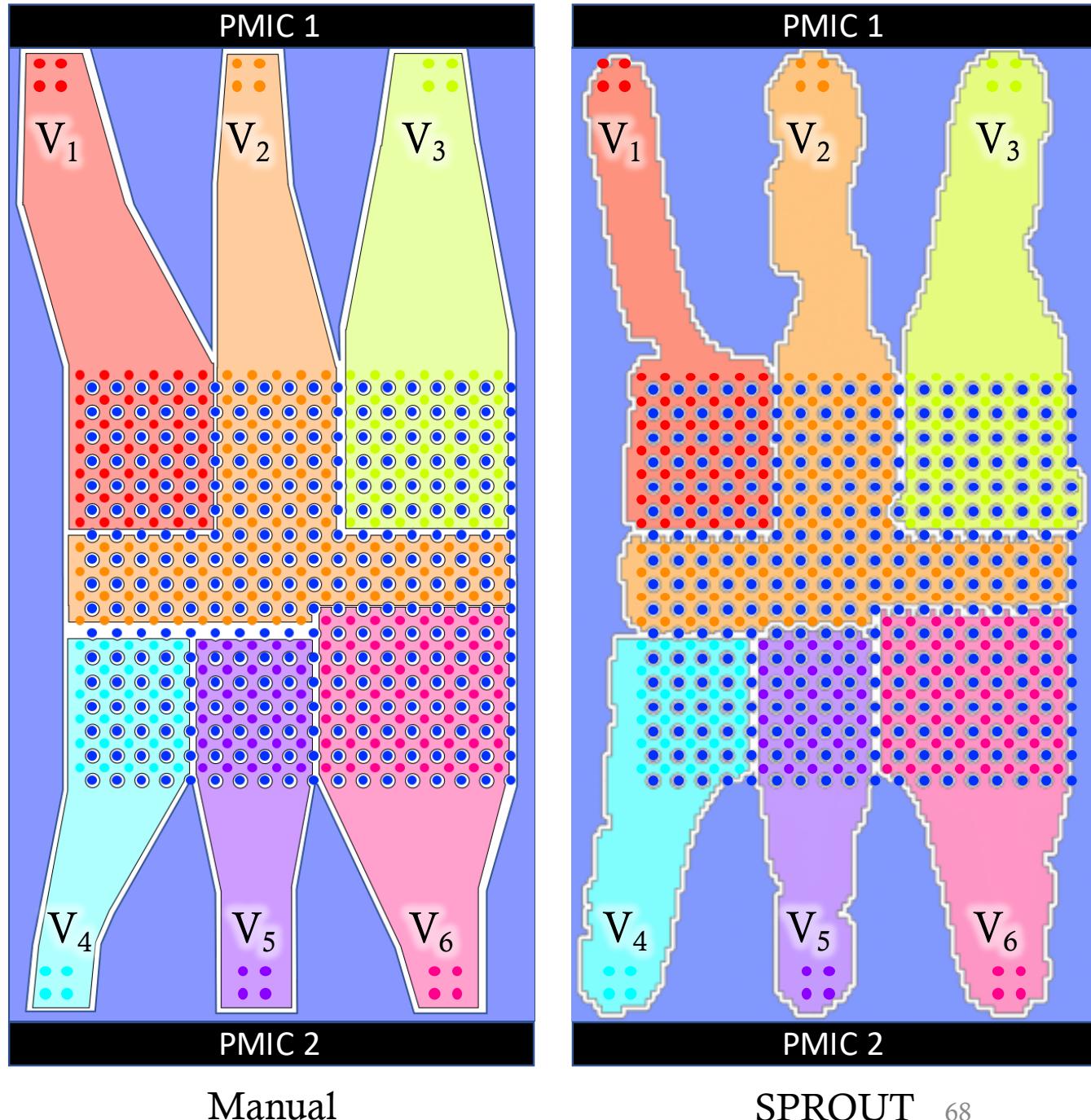
- Power network prototype generated in five steps
  - Available space
  - Seed
  - Growth
  - Refinement
  - Layout



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Case Study: Six-Net System

	Net	Manual	SPROUT
Normalized inductance @ 25 MHz (picohenrys)	$V_1$	133	131
	$V_2$	103	99
	$V_3$	131	127
	$V_4$	161	155
	$V_5$	152	150
	$V_6$	116	114
Normalized DC resistance (millionohms)	$V_1$	15.0	16.8
	$V_2$	8.4	9.1
	$V_3$	13.0	14.2
	$V_4$	18.4	18.2
	$V_5$	18.5	18.9
	$V_6$	9.2	9.2

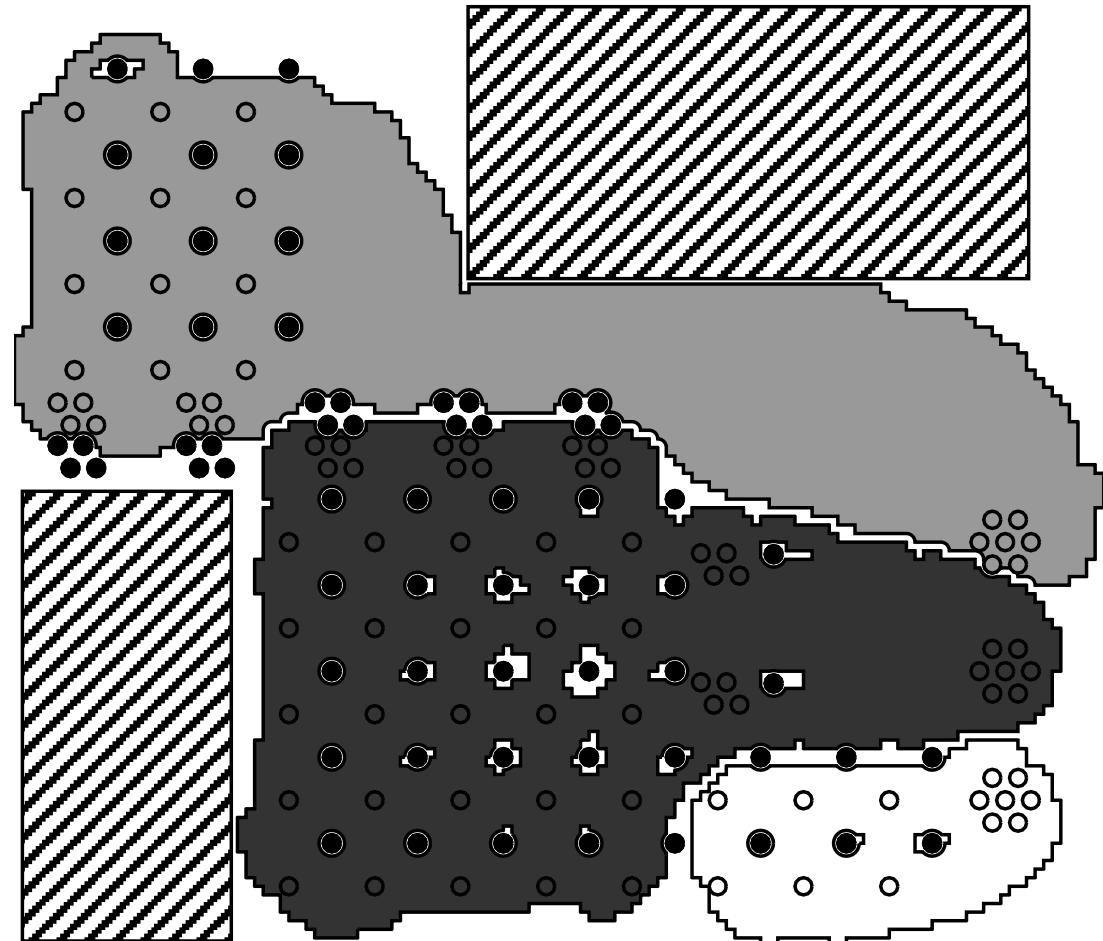


R. Bairamkulov, A. Roy, V. Srinivas, M. Nagarajan, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *Proceedings of the IEEE/ACM Design Automation Conference*, pp. 283-288, December 2021

# Power Delivery Exploration

- Three nets
  1. Modem
  2. CPU
  3. DSP
- Two blockages
- Connect BGA with PMIC
  - L1
  - L2
  - L3
- Explore relationship between
  - Metal area
  - Power network impedance

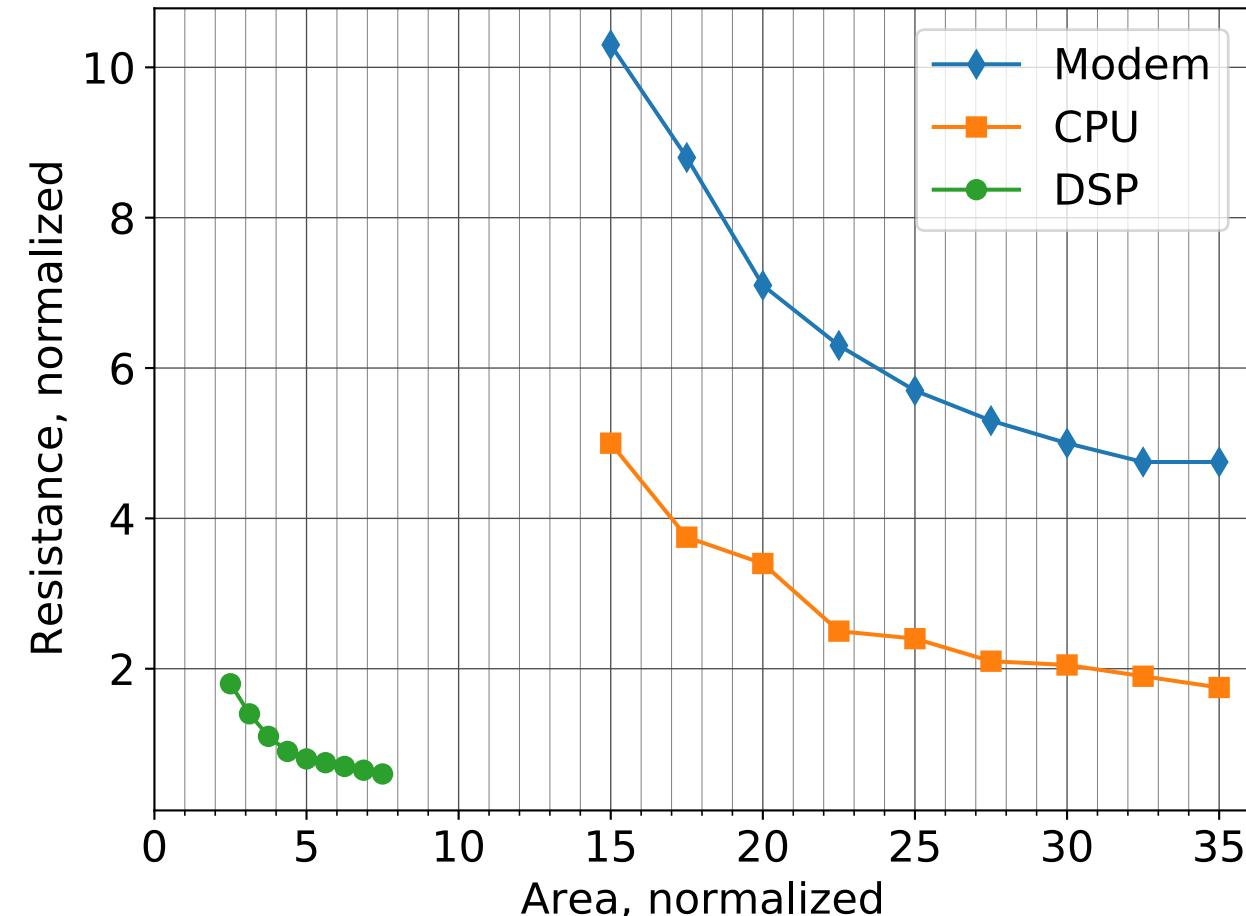
$$A_1 = 35.0, A_2 = 35.0, A_3 = 7.5$$



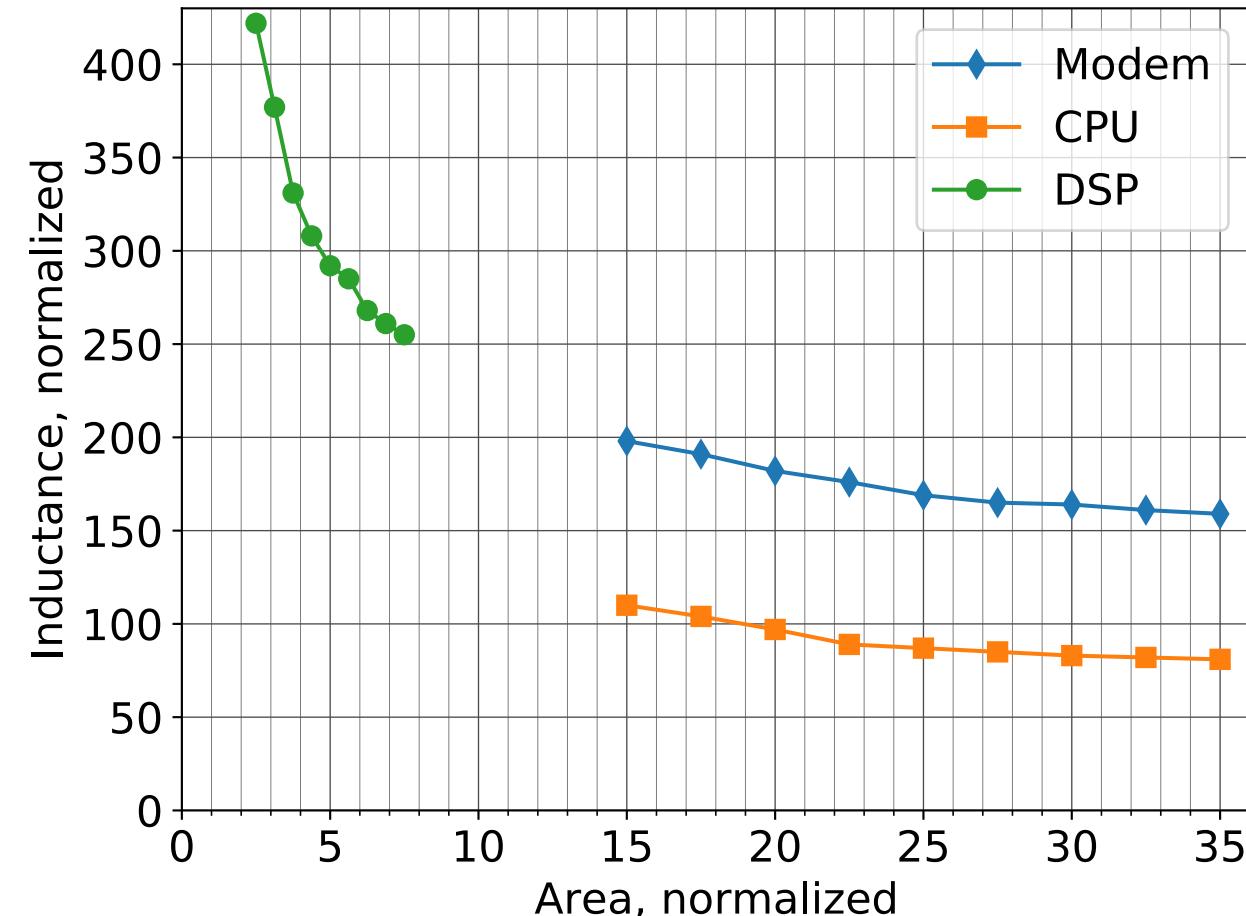
R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, "SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

# Impedance vs. Area

Resistance



Inductance



R. Bairamkulov, A. Roy, M. Nagarajan, V. Srinivas, and E. G. Friedman, “SPROUT - Smart Power ROUTing Tool for Board-Level Exploration and Prototyping,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (in press)

**Research Effort  
since 2017**

**VLSI CONTINENT**

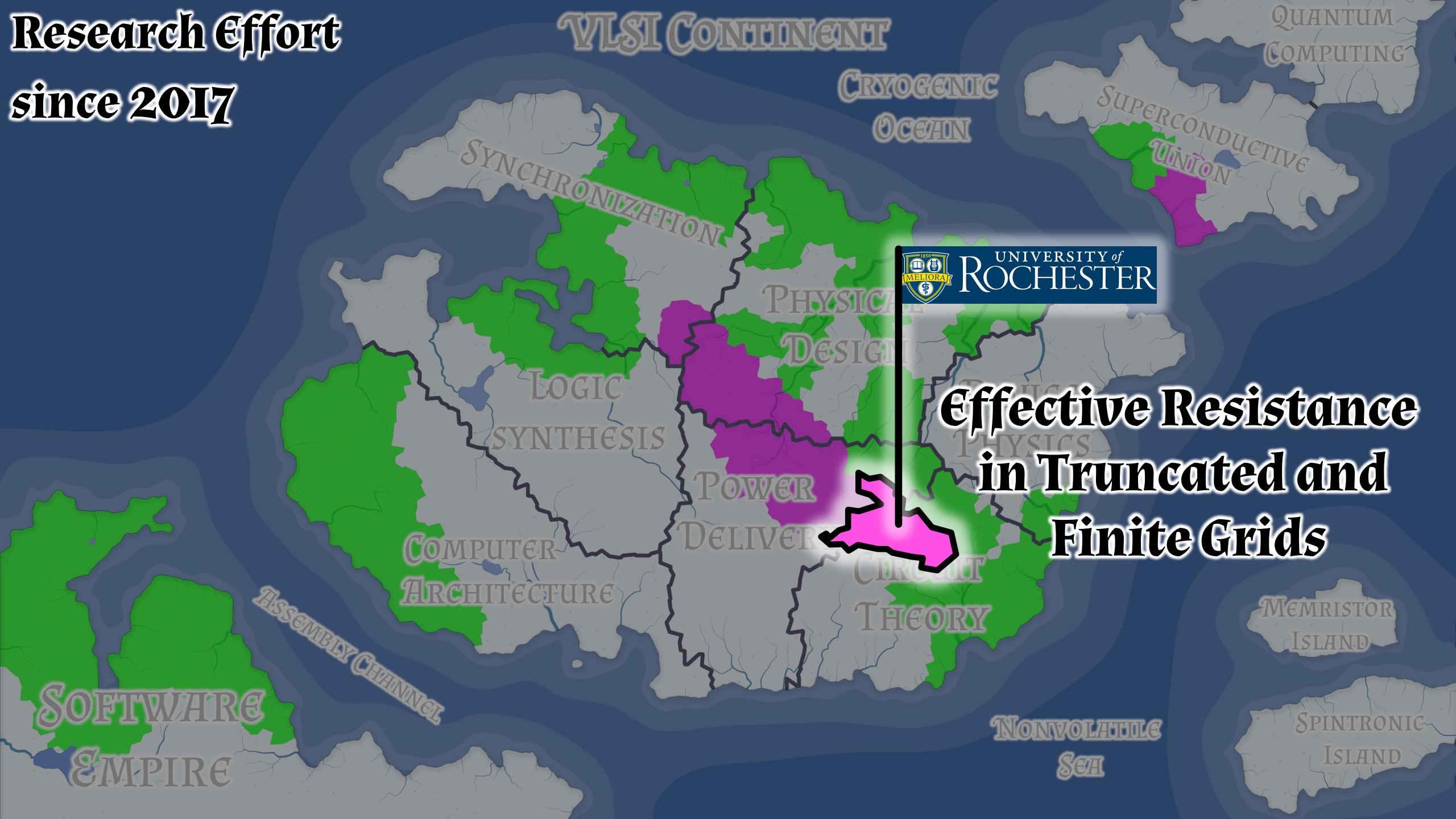
**CRYOGENIC  
OCEAN**

**QUANTUM  
COMPUTING**

**SUPERCONDUCTIVE  
UNION**



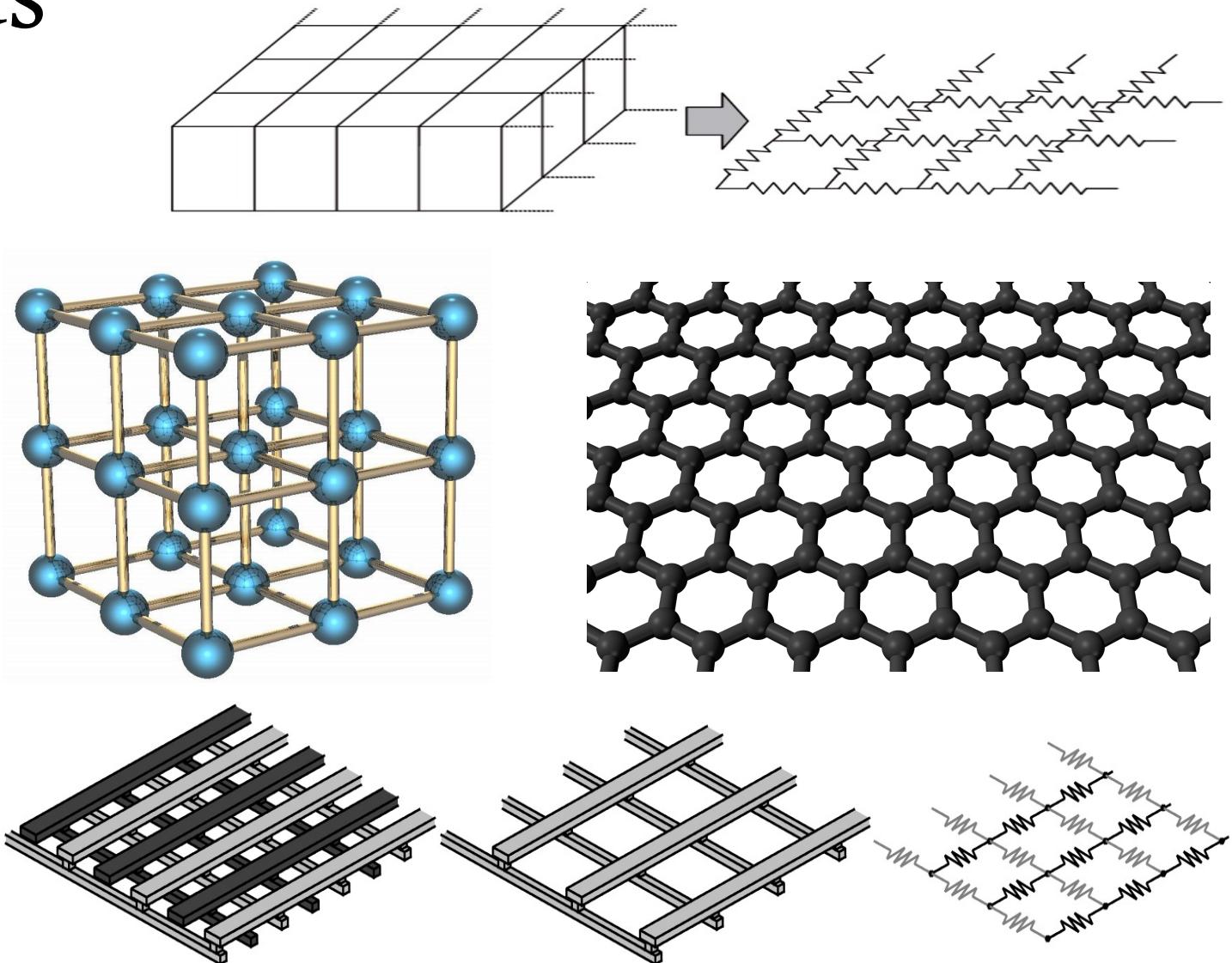
# **Effective Resistance in Truncated and Finite Grids**



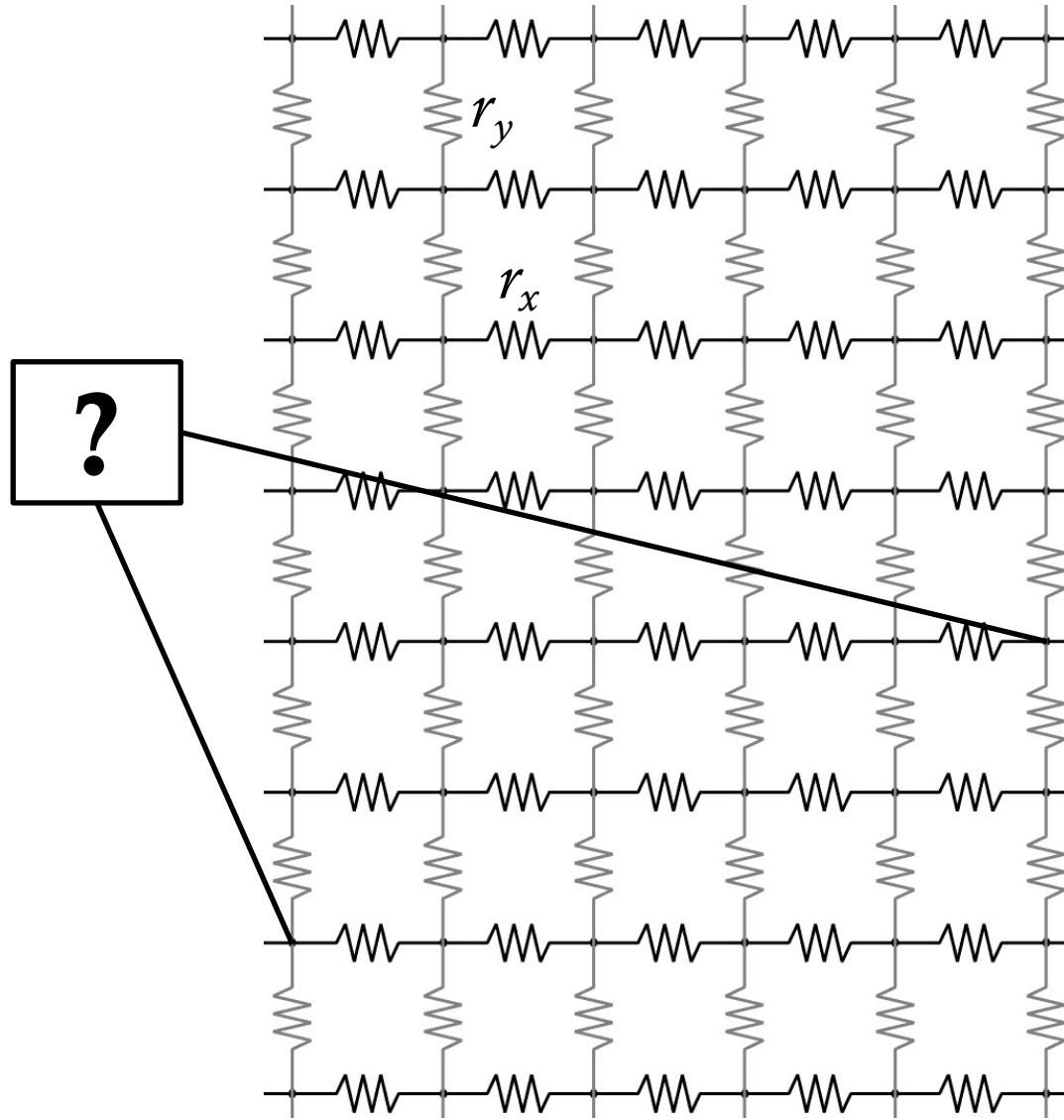
# Importance of Grids

Appear frequently in science and engineering

- Chemical bonds
- Substrate models
- Discretized electrical and thermal models
- **On-chip power and ground networks**

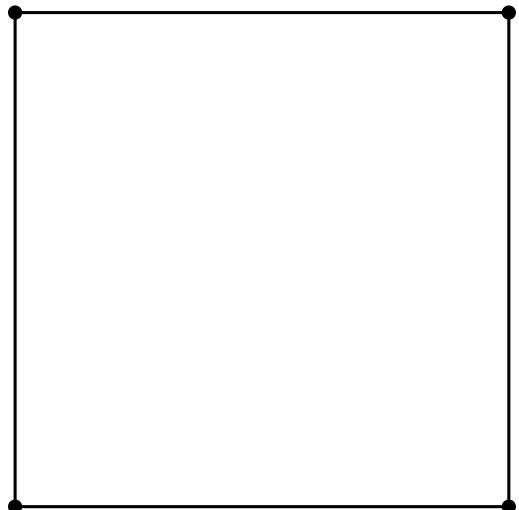


# Analysis of Grids



# Conventional MNA Analysis

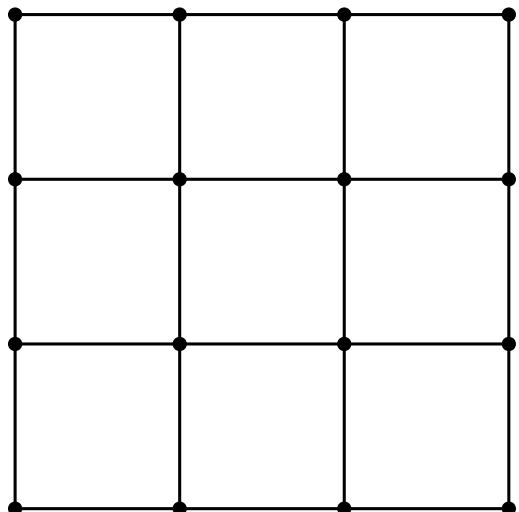
- Superlinear complexity



$$\left[ \begin{array}{cccc} 2 & -1 & -1 & 0 \\ -1 & 2 & 0 & -1 \\ -1 & 0 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{array} \right]$$

# Conventional MNA Analysis

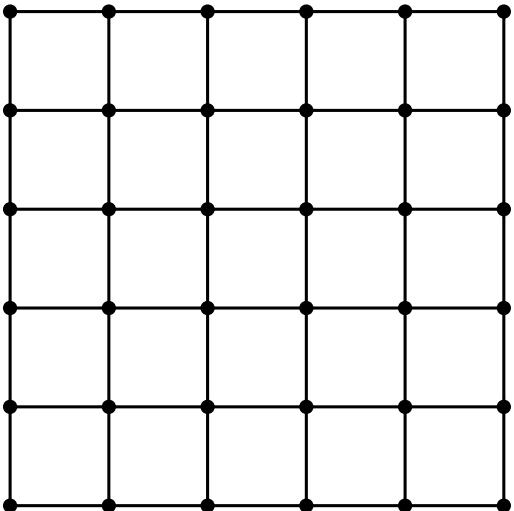
- Superlinear complexity



$$\left( \begin{array}{cccccccccccccc} 2 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 3 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 3 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 3 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 3 & -1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 3 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 2 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 3 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 3 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 3 & -1 \end{array} \right)$$

# Conventional MNA Analysis

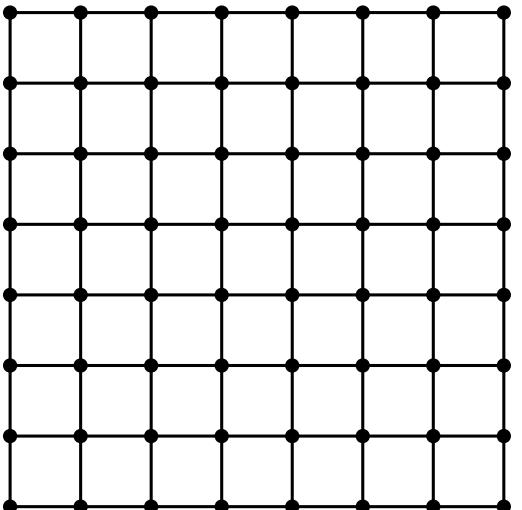
- Superlinear complexity



2 -1 0 0 0 0 -1 0  
-1 3 -1 0 0 0 0 -1 0  
0 -1 3 -1 0 0 0 0 -1 0  
0 0 -1 3 -1 0 0 0 0 -1 0  
0 0 0 -1 3 -1 0 0 0 0 -1 0  
0 0 0 0 -1 2 0 0 0 0 0 -1 0  
-1 0 0 0 0 0 3 -1 0 0 0 -1 0  
0 -1 0 0 0 0 -1 4 -1 0 0 0 0 -1 0  
0 0 -1 0 0 0 0 -1 4 -1 0 0 0 -1 0  
0 0 0 -1 0 0 0 0 -1 4 -1 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 -1 0 0 0 0 -1 4 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 -1 0 0 0 0 0 -1 3 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 -1 0 0 0 0 0 3 -1 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 -1 0 0 0 0 0 -1 4 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 -1 0 0 0 0 0 -1 4 -1 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 -1 3 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 -1 0 0 0 0 0 3 -1 0 0 0 0 -1 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 -1 4 -1 0 0 0 0 -1 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 -1 4 -1 0 0 0 0 -1 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 0 0 -1 4 -1 0 0 0 0 -1 0 0 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 0 0 -1 4 -1 0 0 0 0 0 -1 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 0 0 -1 3 0 0 0 0 0 0 -1 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0 0  
0 -1 0 0 0 0 0 0  
0 -1 0 0 0 0 0  
0 -1 0 0 0 0  
0 -1 0 0 0  
0 -1 0 0  
0 -1 0  
0 -1

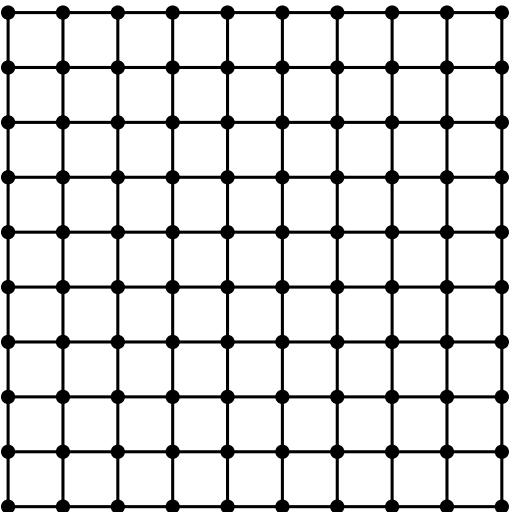
# Conventional MNA Analysis

- Superlinear complexity

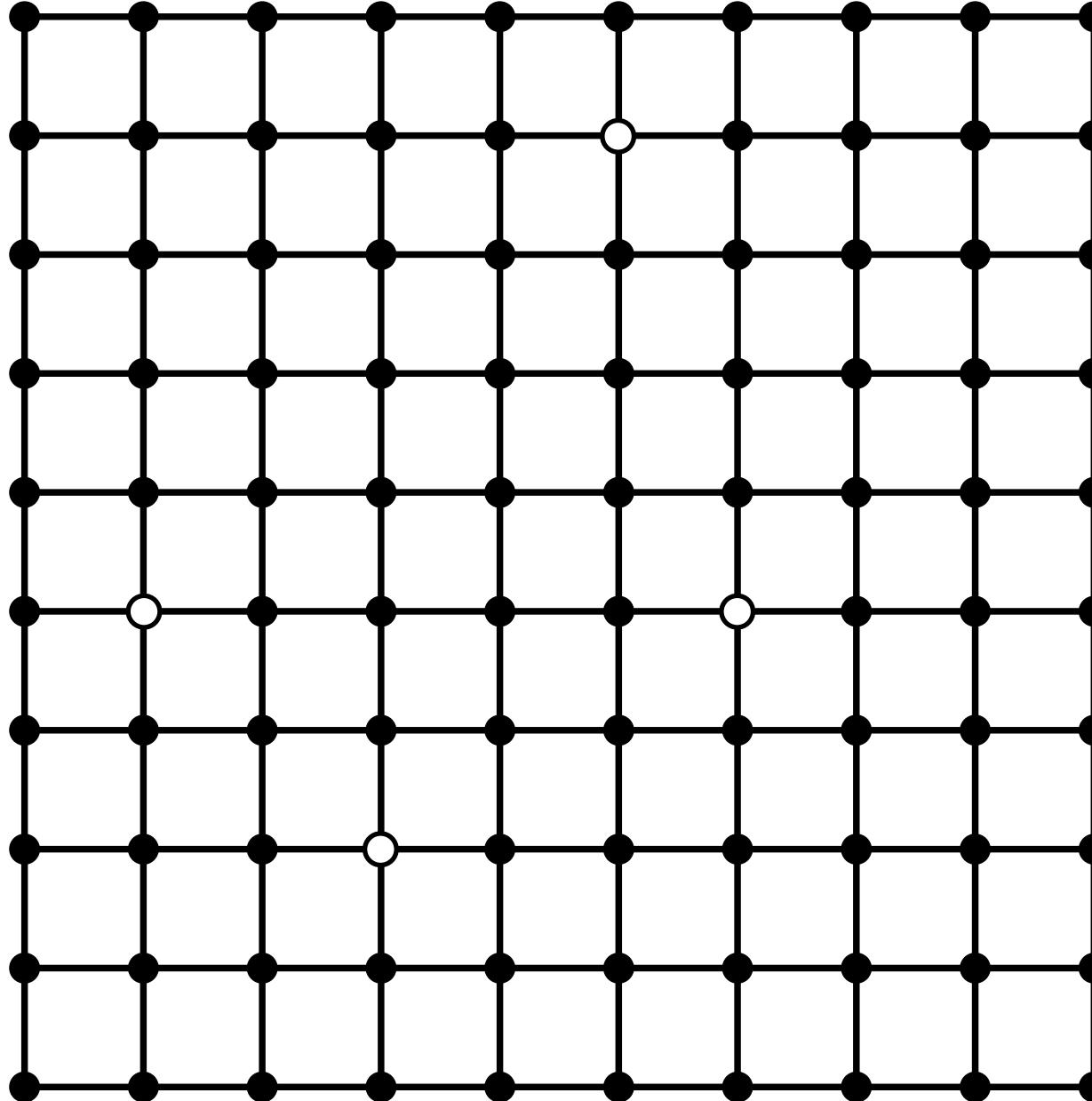


# Conventional MNA Analysis

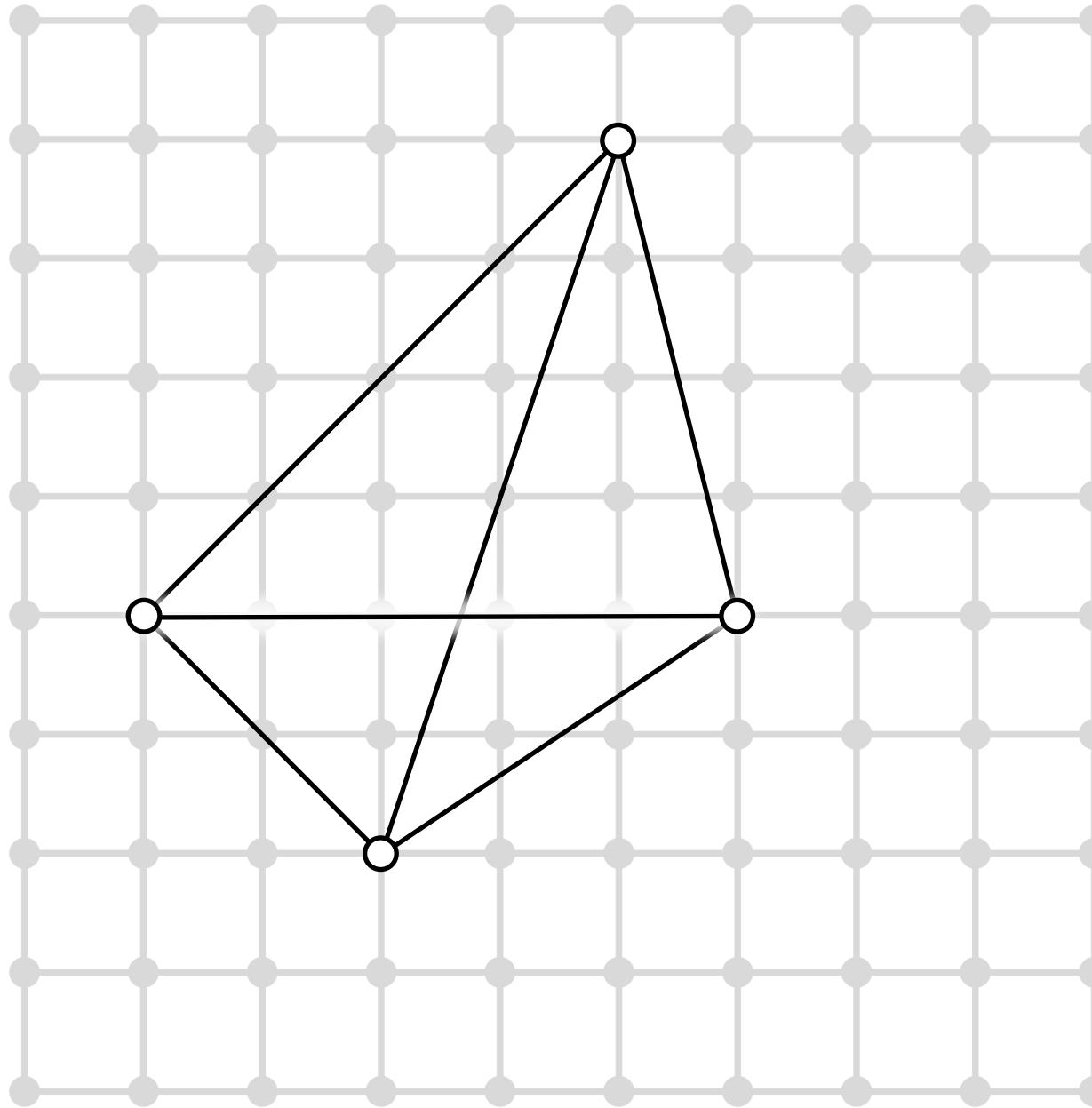
- Superlinear complexity



# Effective Resistance



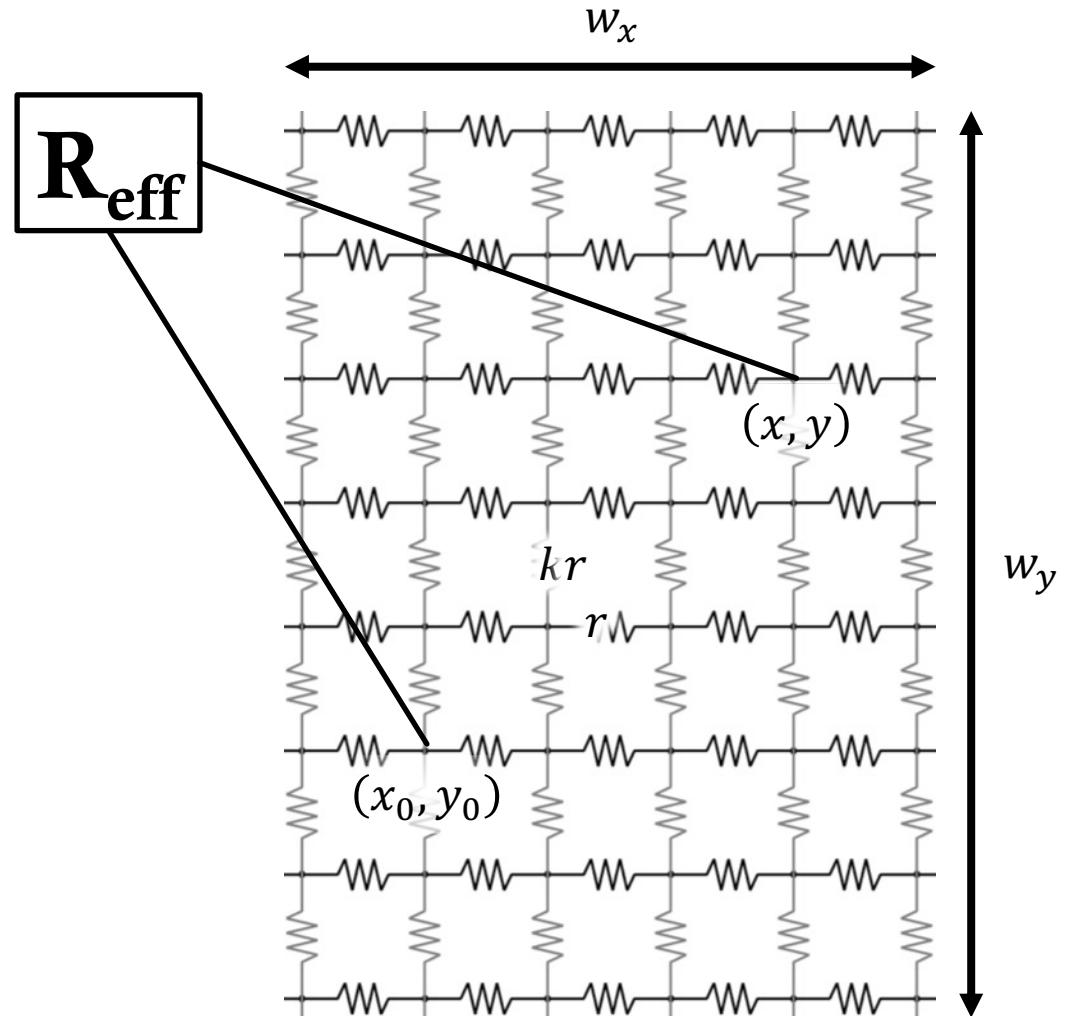
# Effective Resistance



# Problem Statement

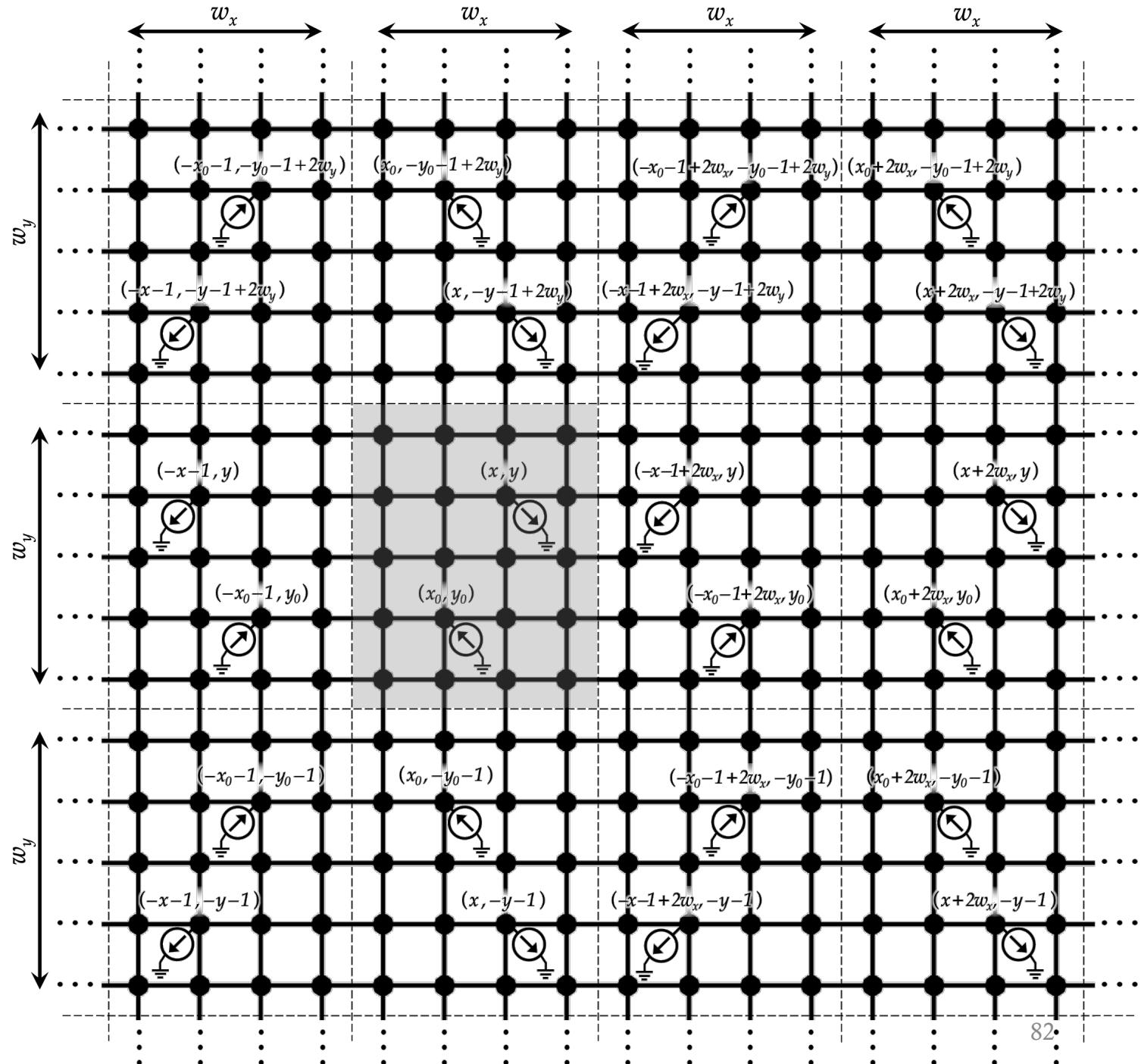
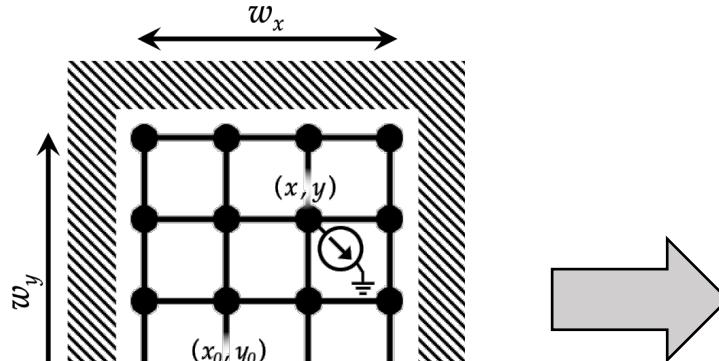
Find effective resistance between two arbitrary nodes in rectangular resistive grid with given

- Size  $w_x \times w_y$
- Node coordinates  $(x_0, y_0), (x, y)$
- Vertical resistance  $r$
- Horizontal resistance  $kr$



# Infinity Mirror Technique

- Effective resistance is found in  $O(1)$  time

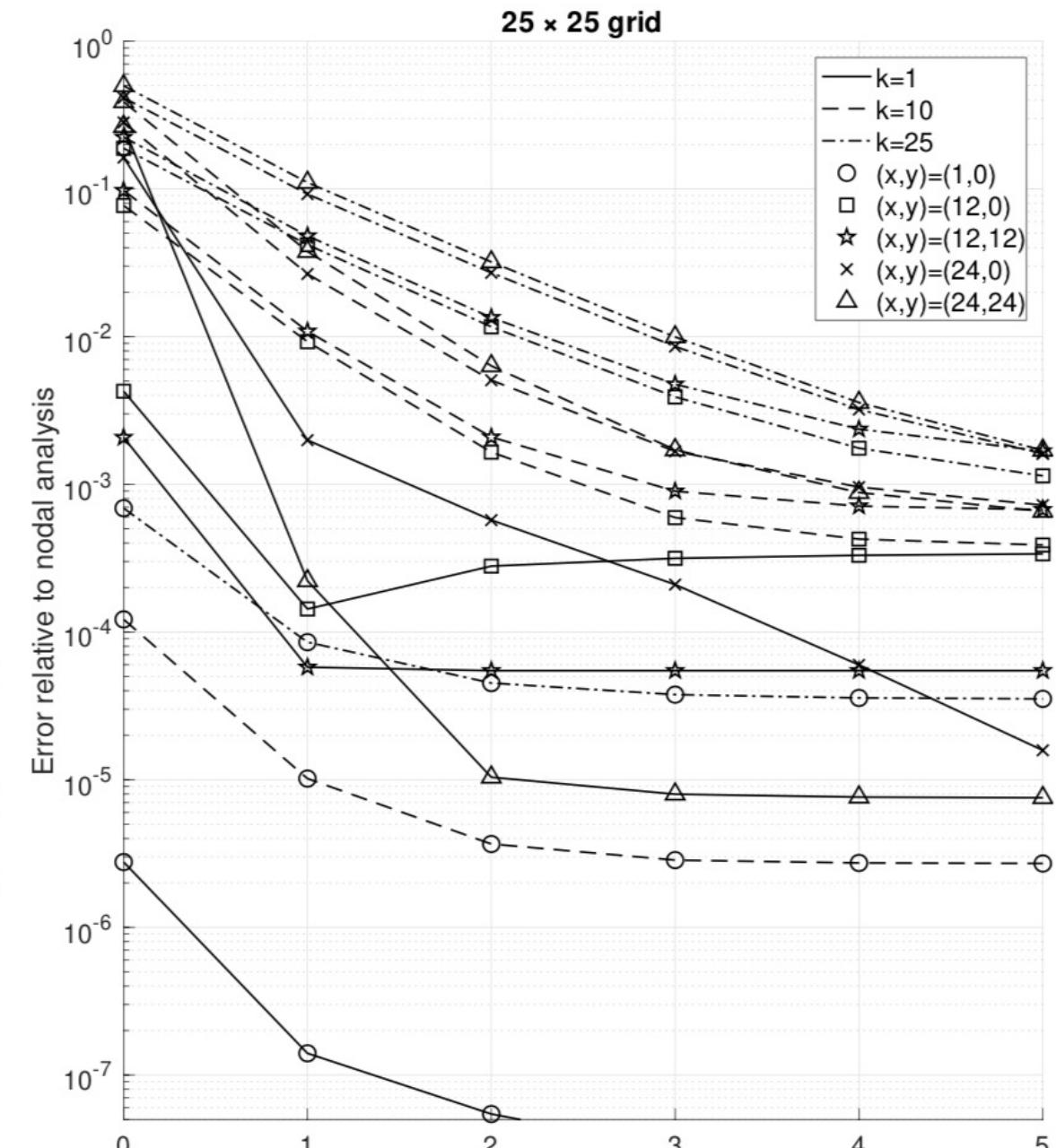


R. Bairamkulov and E. G. Friedman, "Effective Resistance of Finite Two-Dimensional Grids Based on Infinity Mirror Technique," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 9, pp. 3224-3233, September 2020

# Model Accuracy

- Infinitely many images
- Only closest images are sufficient

$$\begin{aligned}
 R_{w_x, w_y} = & \sum_{i \in \mathbb{Z}} \sum_{j \in \mathbb{Z}} (2\Omega_{ij}^k(x-x_0, y-y_0) + 2\Omega_{ij}^k(x-x_0, y+y_0+1) \\
 & + 2\Omega_{ij}^k(x+x_0+1, y+y_0+1) + 2\Omega_{ij}^k(x+x_0+1, y-y_0) \\
 & - \Omega_{ij}^k(2x_0+1, 0) - \Omega_{ij}^k(2x_0+1, 2y_0+1) - \Omega_{ij}^k(0, 2y_0+1)) \\
 & - \Omega_{ij}^k(2x+1, 0) - \Omega_{ij}^k(2x+1, 2y+1) - \Omega_{ij}^k(0, 2y+1) - 2\Omega_{ij}^k(0, 0)
 \end{aligned}$$



# Summary

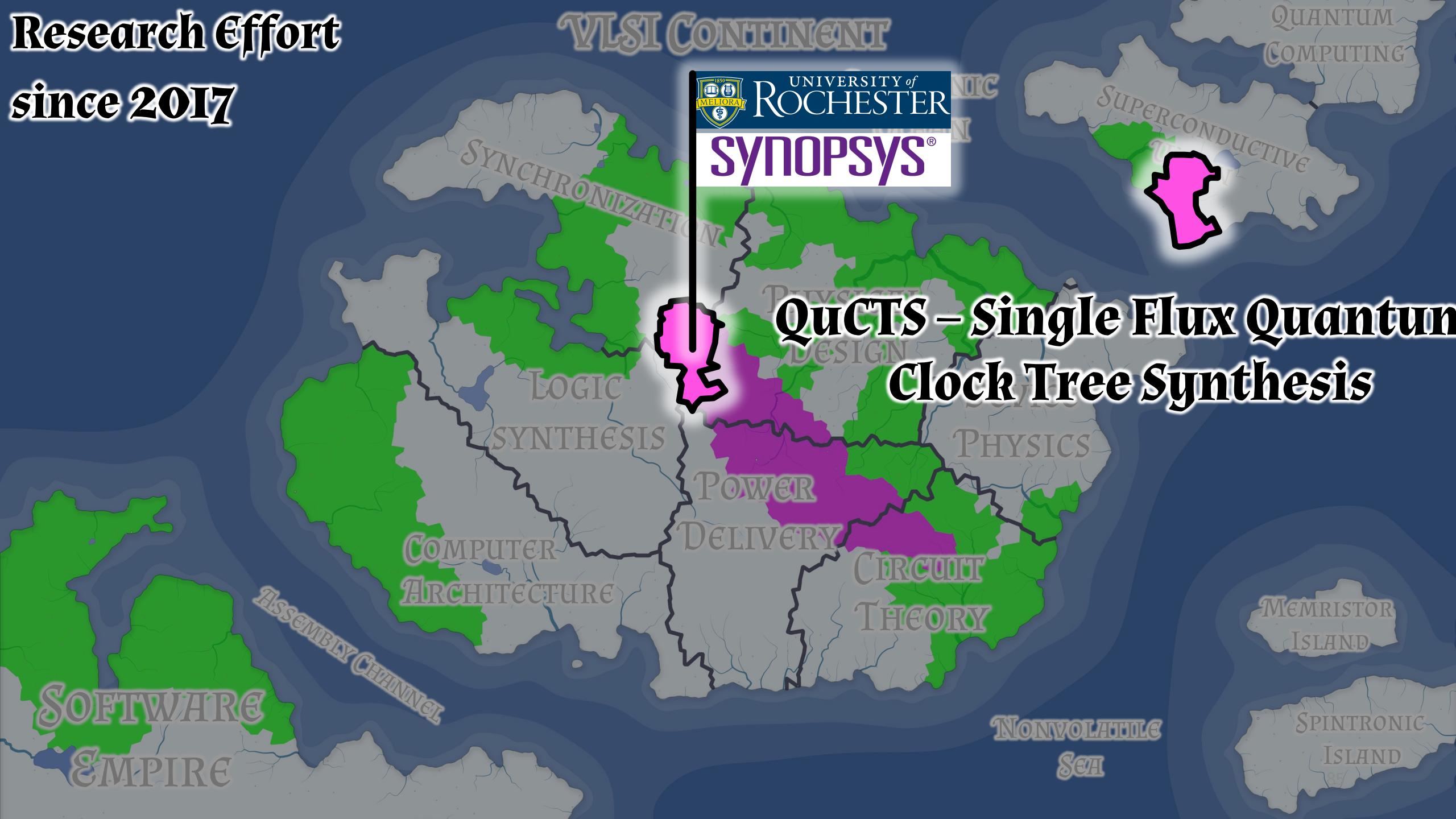
- Infinite grid model extended for finite grids
- Improved accuracy near grid boundaries
- Fast IR drop analysis
  - Runtime independent of grid size
  - Scales quadratically with number of nodes of interest

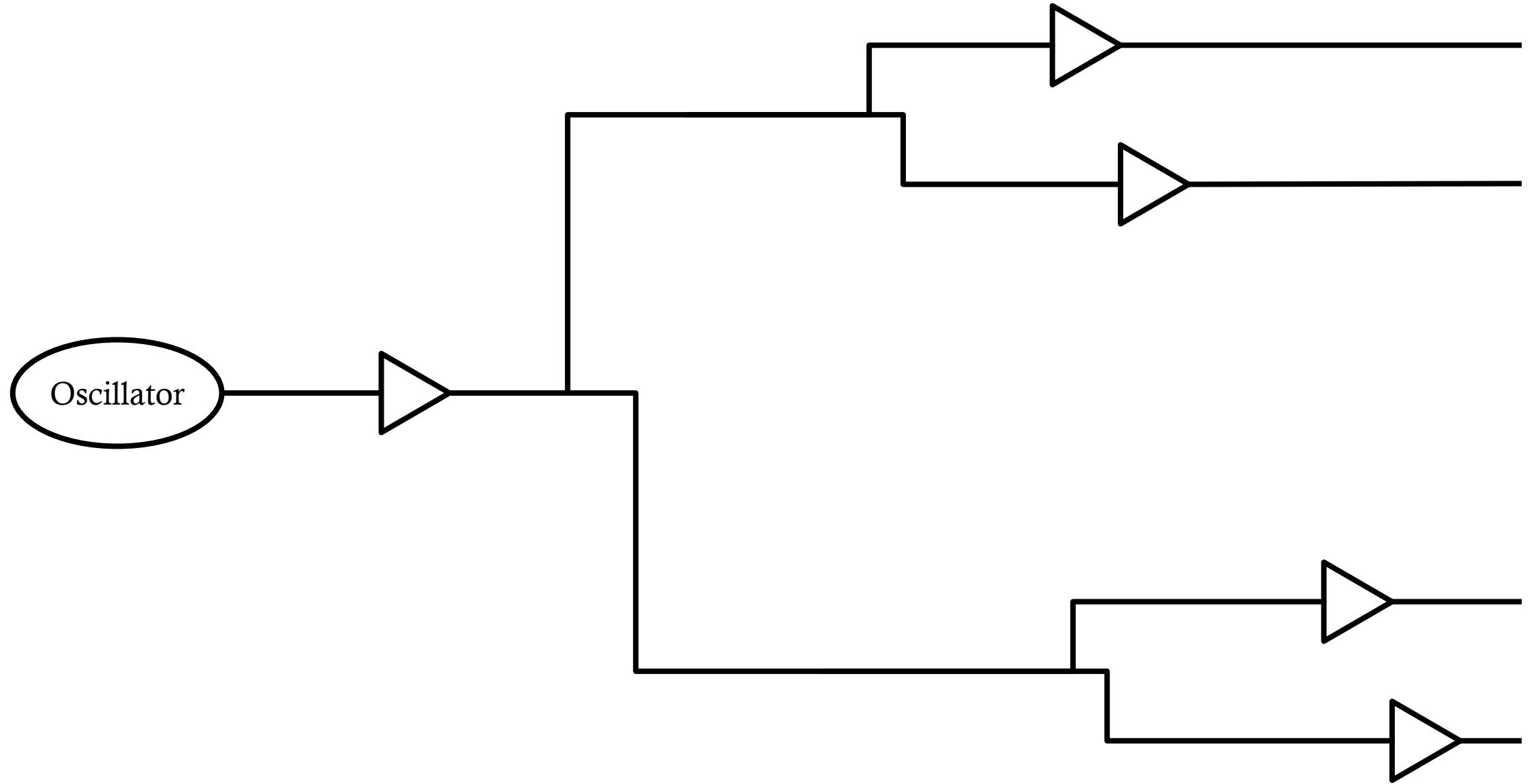
**Research Effort**  
**since 2017**

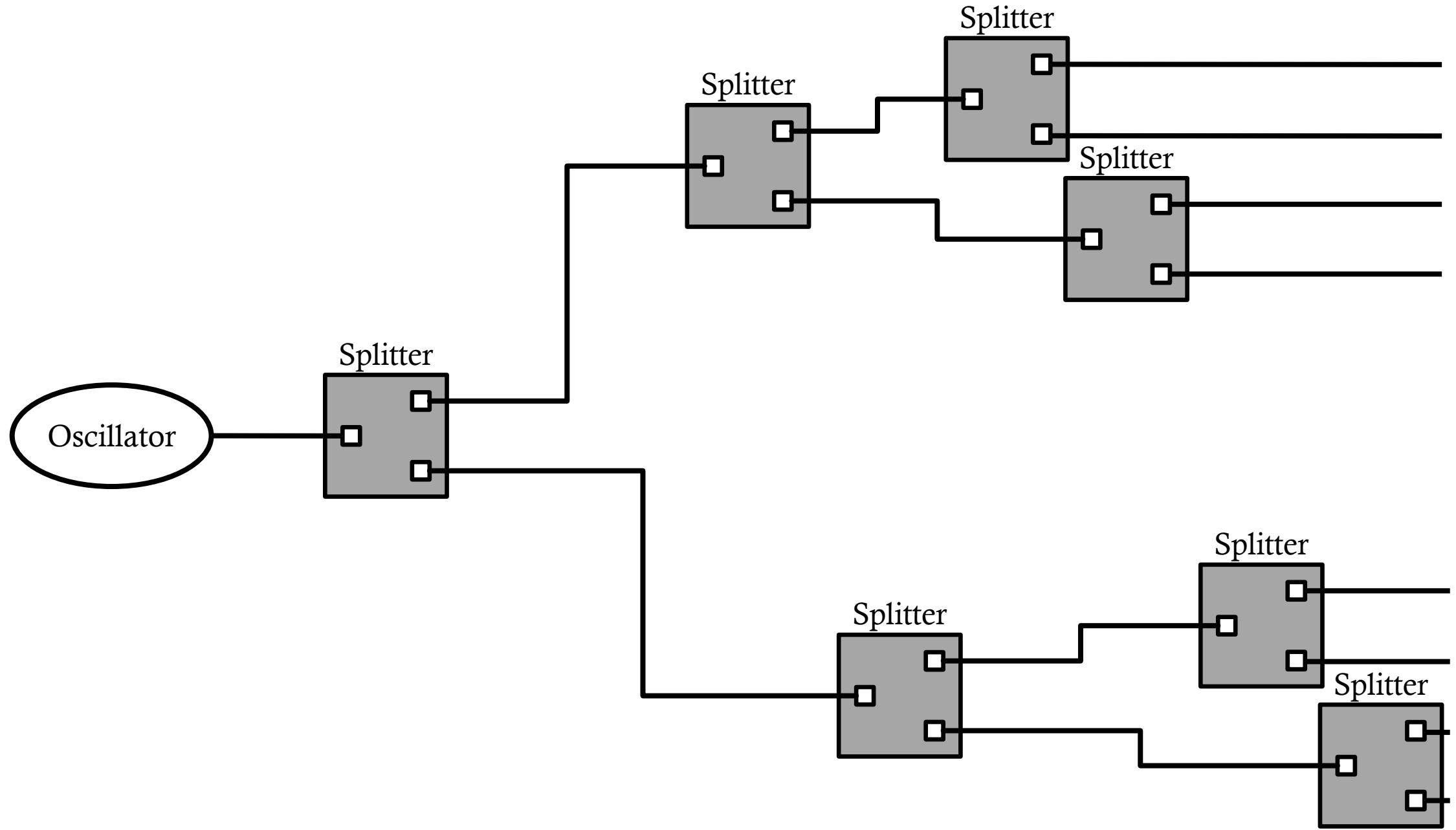
VLSI CONTINENT



# QuCTS – Single Flux Quantum Clock Tree Synthesis

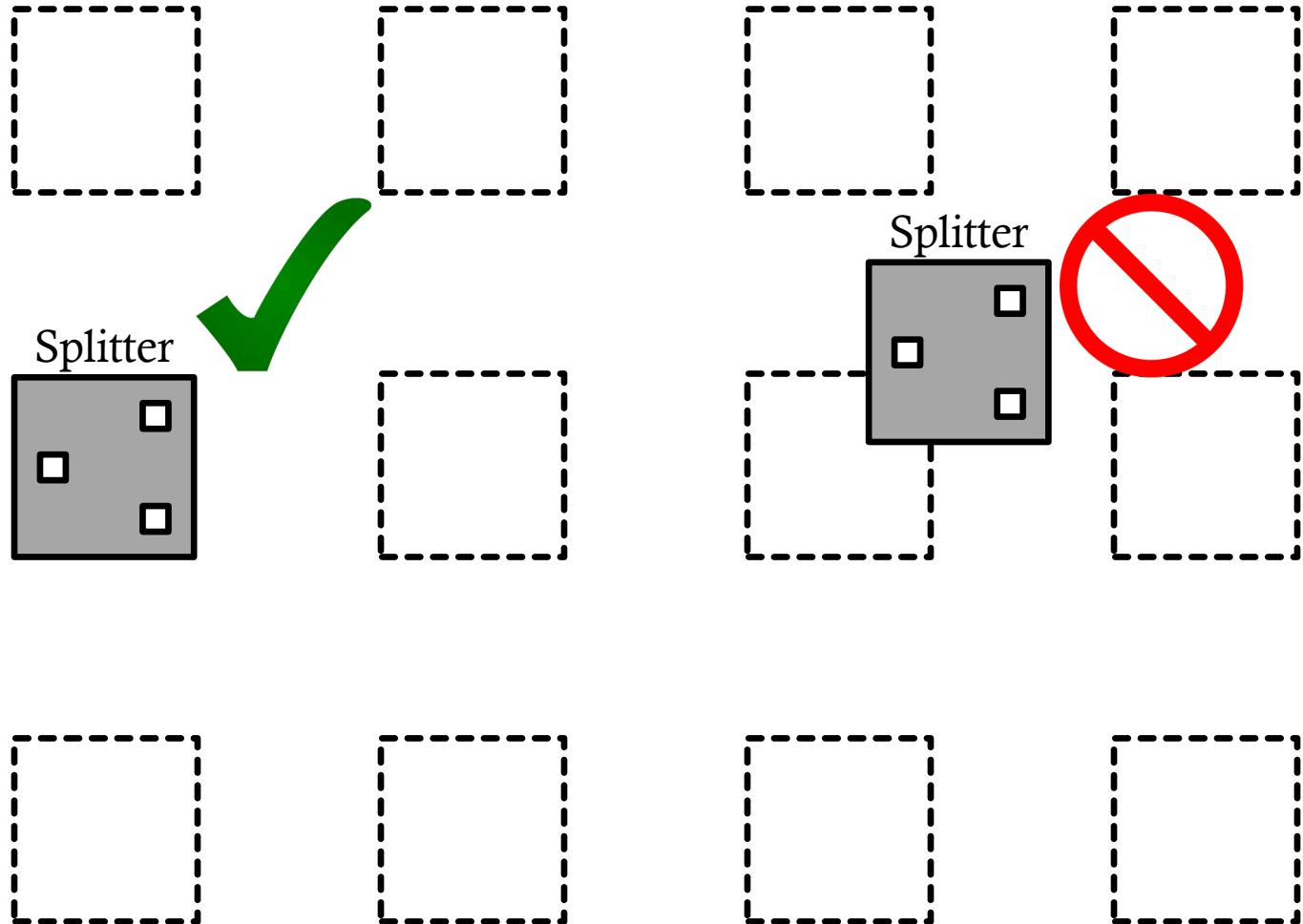






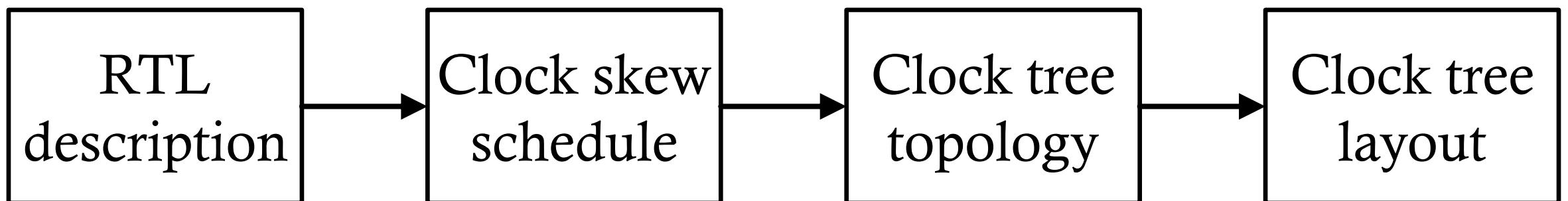
# Restricted Cell Placement

- Gates placed within specific cells
- $N-1$  splitters for  $N$  gates



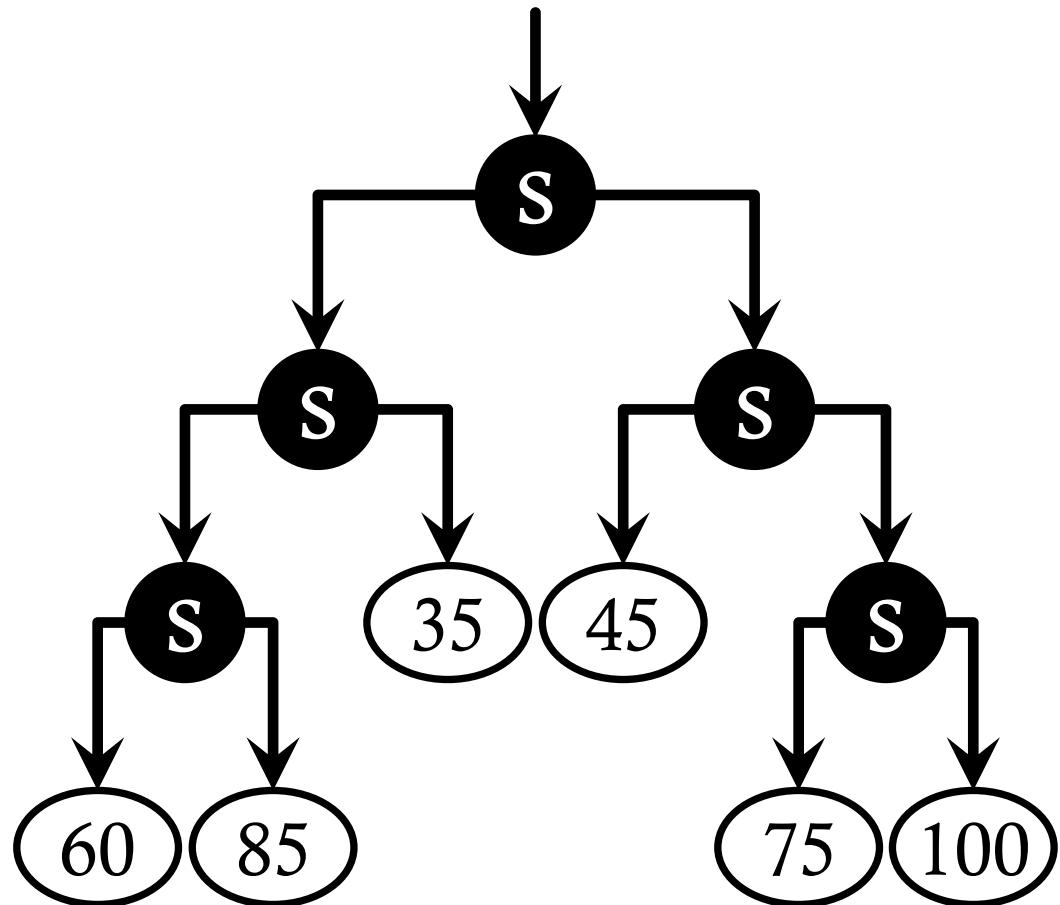
# SFQ Clock Tree Synthesis

- The first SFQ clock tree synthesis algorithm utilizing clock skew
- Input: RTL circuit description
- Output: clock tree layout optimized for robustness and cost

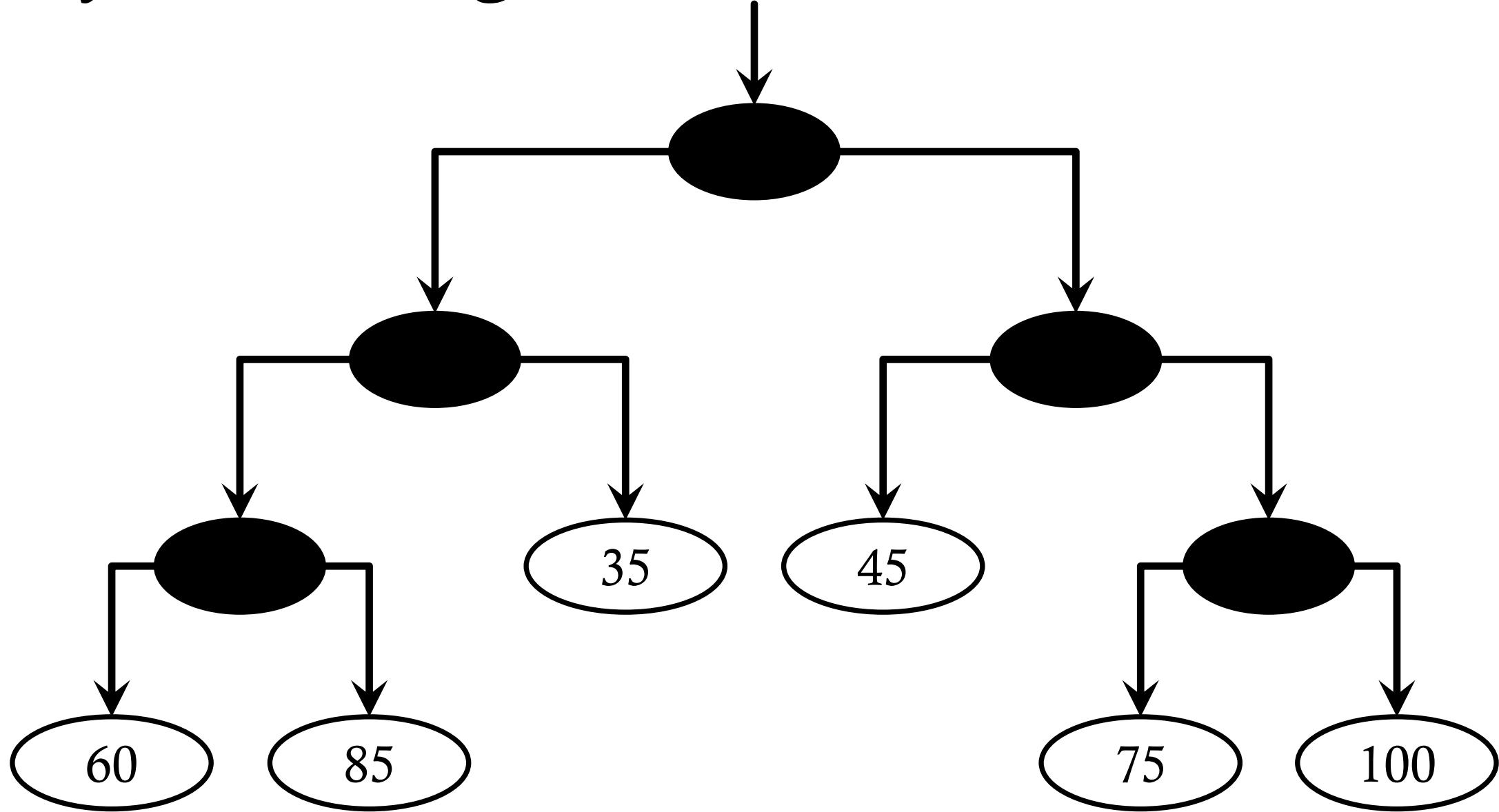


# Delay Balancing: Macro Level

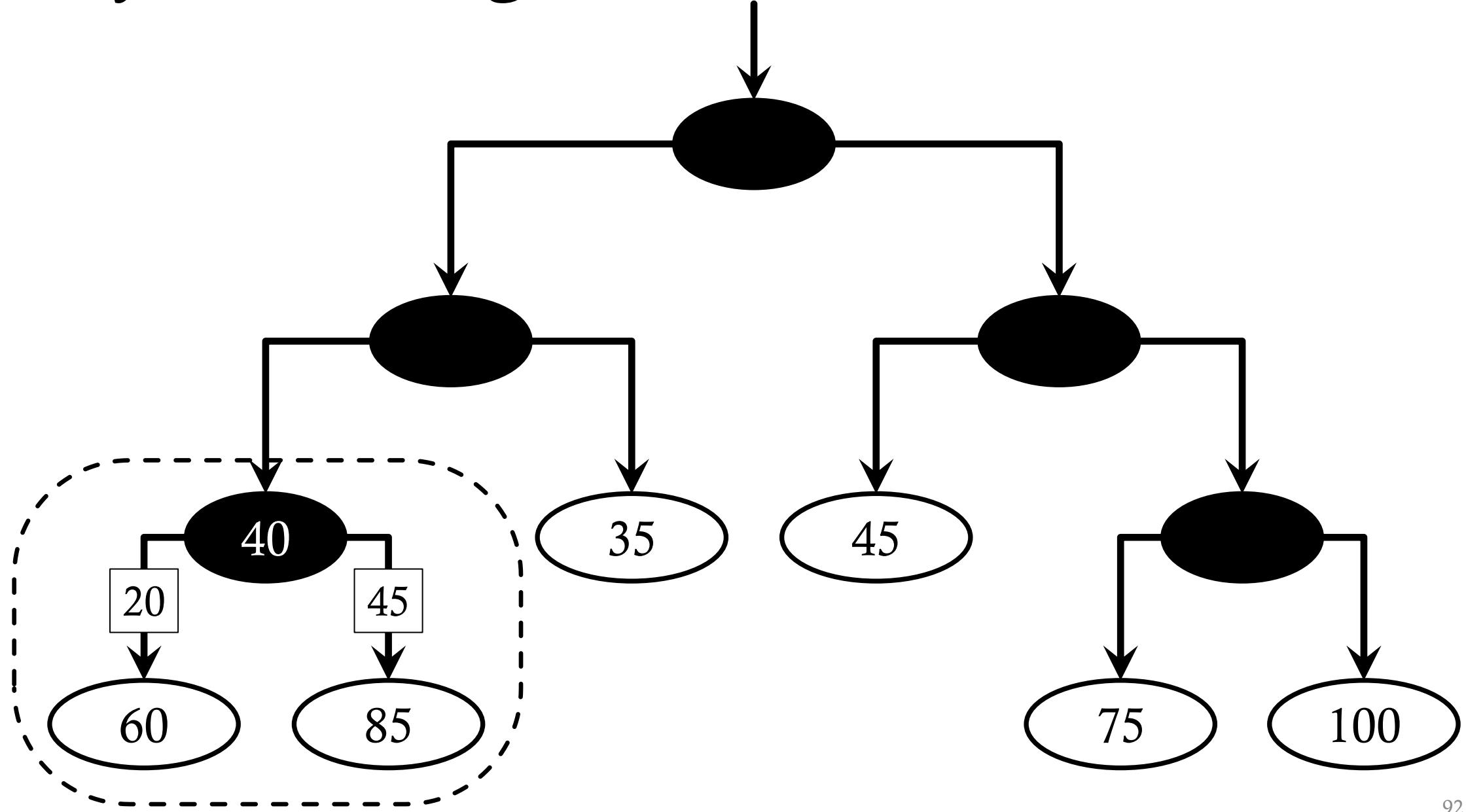
- Clock arrival time determined from clock skew scheduling
- Binary tree satisfies clock arrival time constraints
  - Inherent interconnect delay
    - Snaking to fine tune delay
  - Add JTLs for additional delay



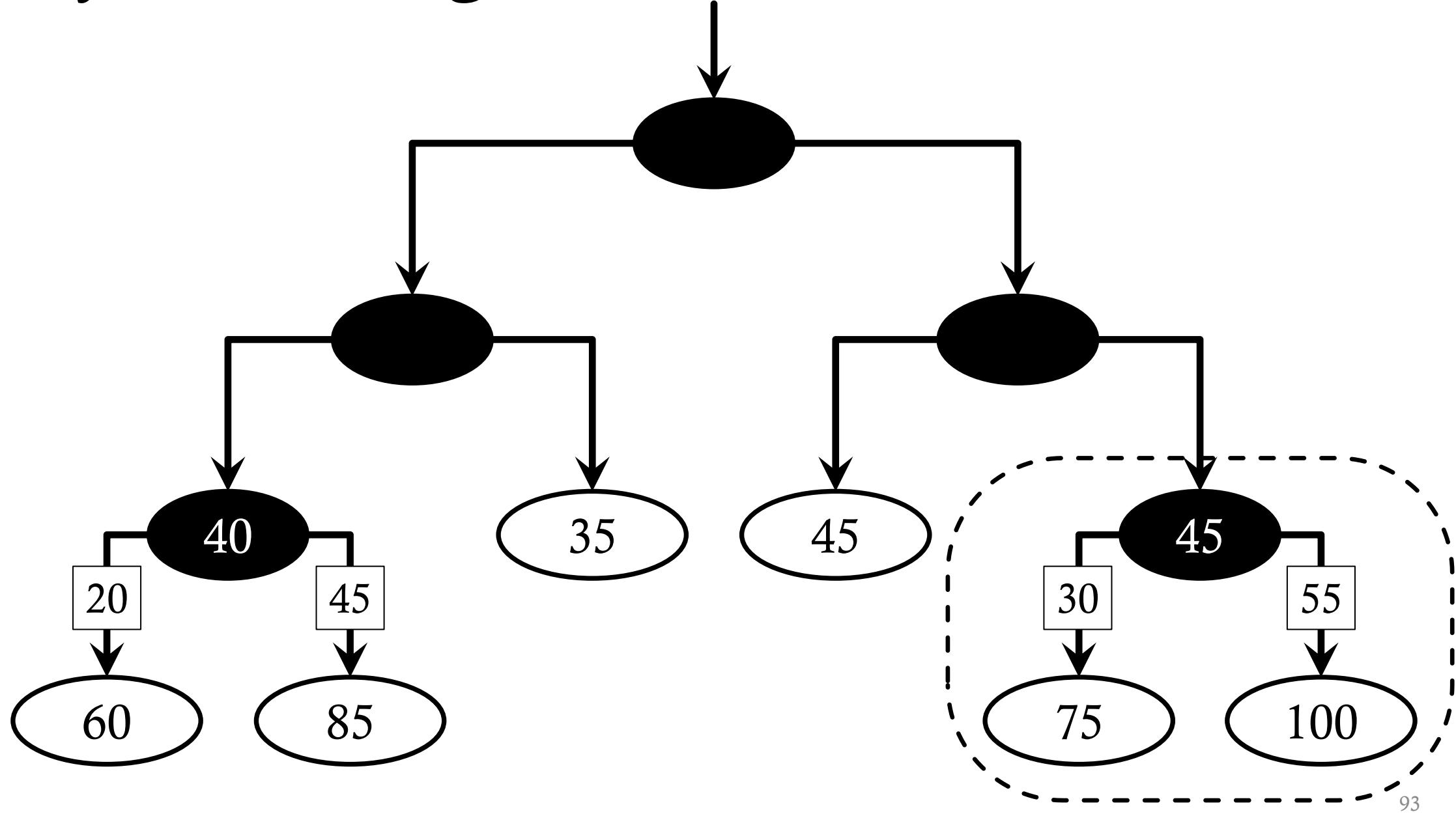
# Delay Balancing: Macro Level



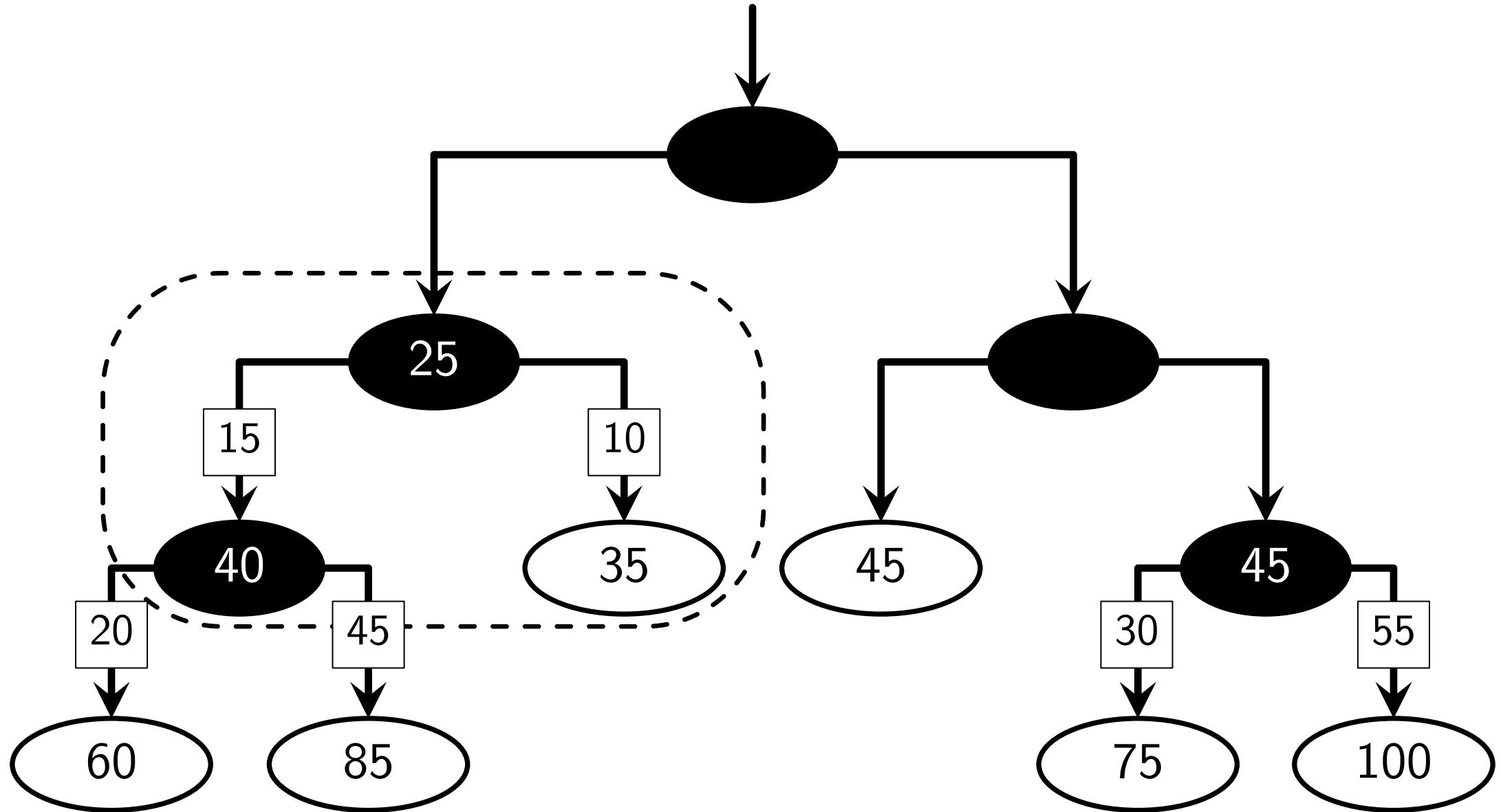
# Delay Balancing: Macro Level



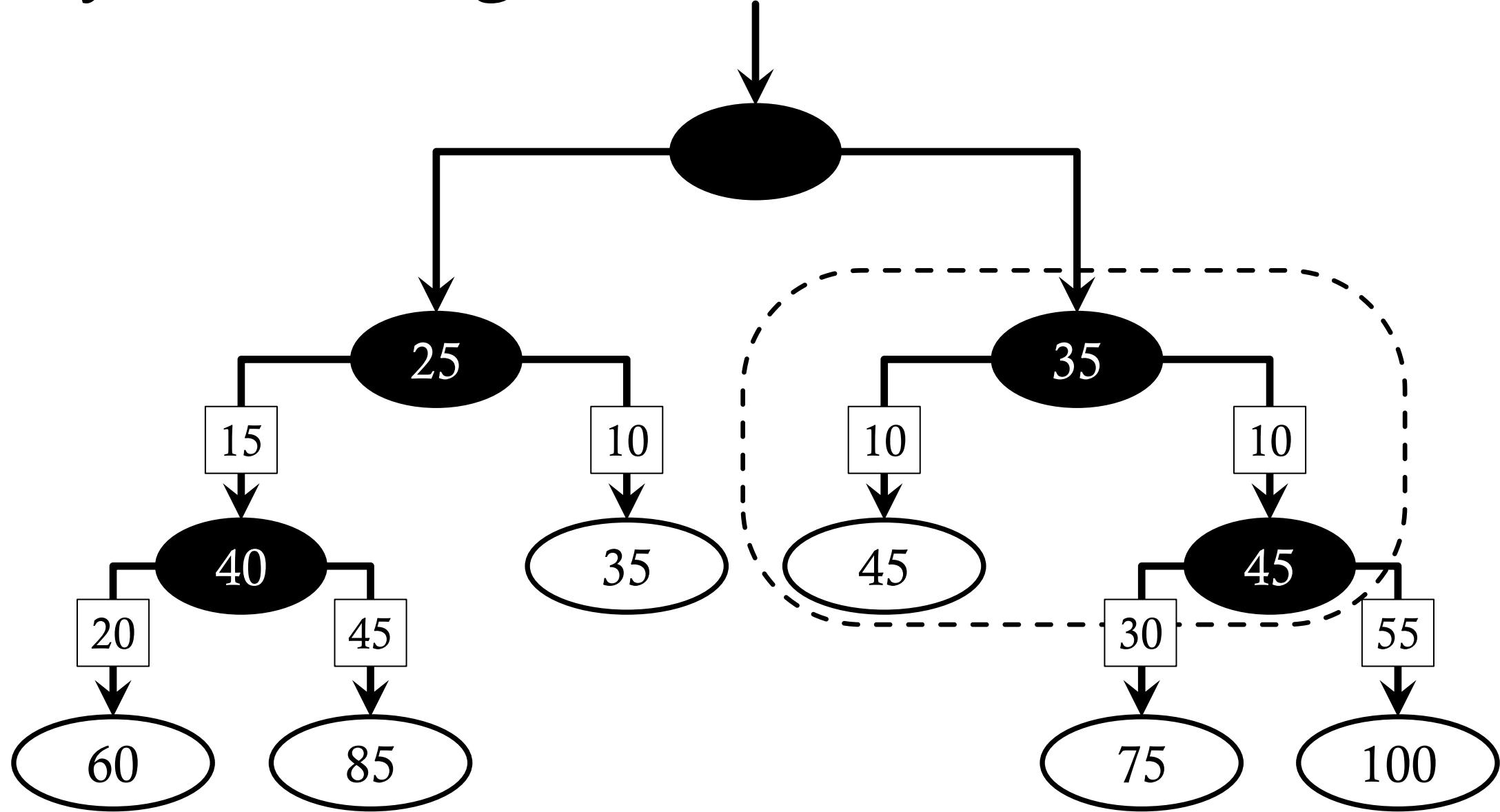
# Delay Balancing: Macro Level



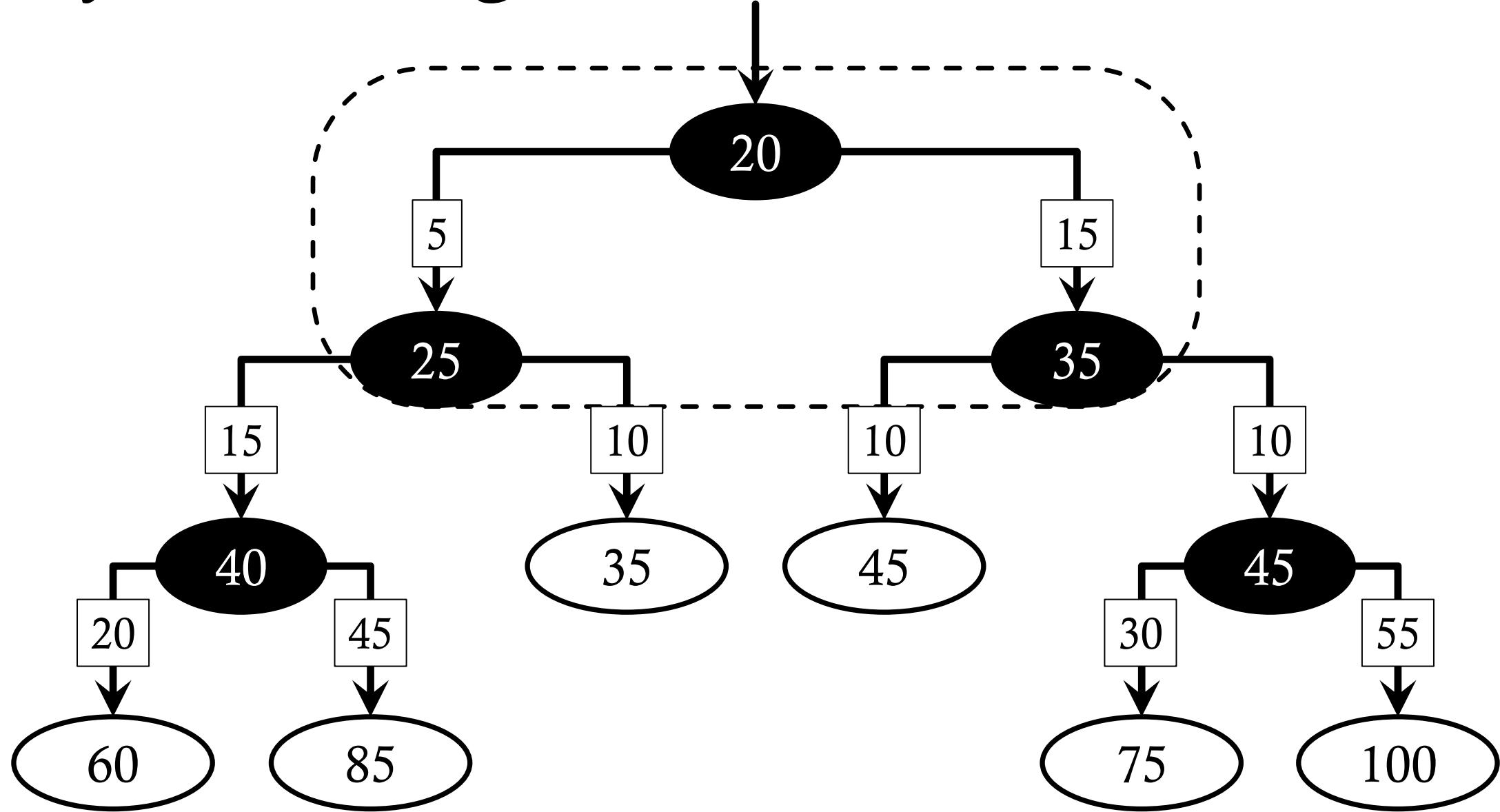
# Delay Balancing: Macro Level



# Delay Balancing: Macro Level

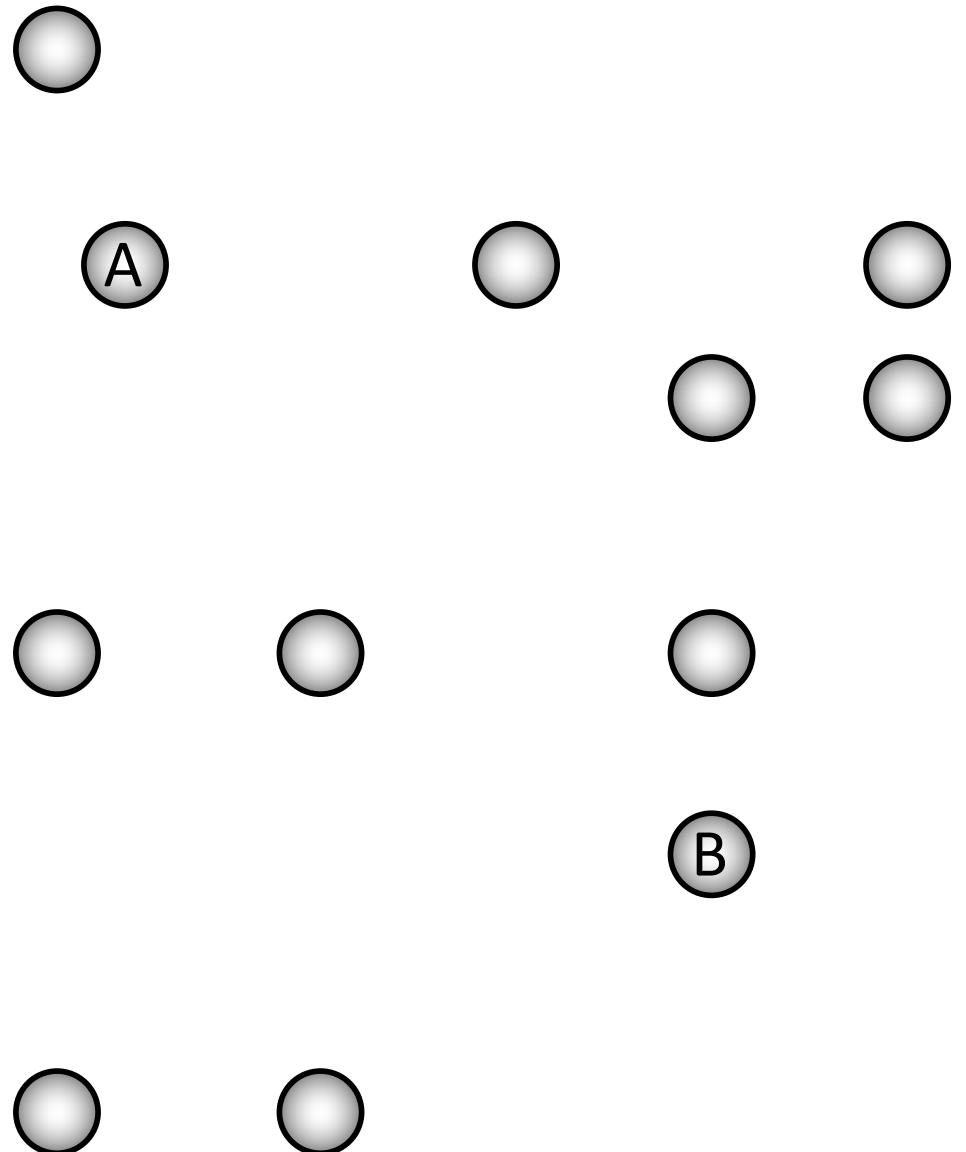


# Delay Balancing: Macro Level



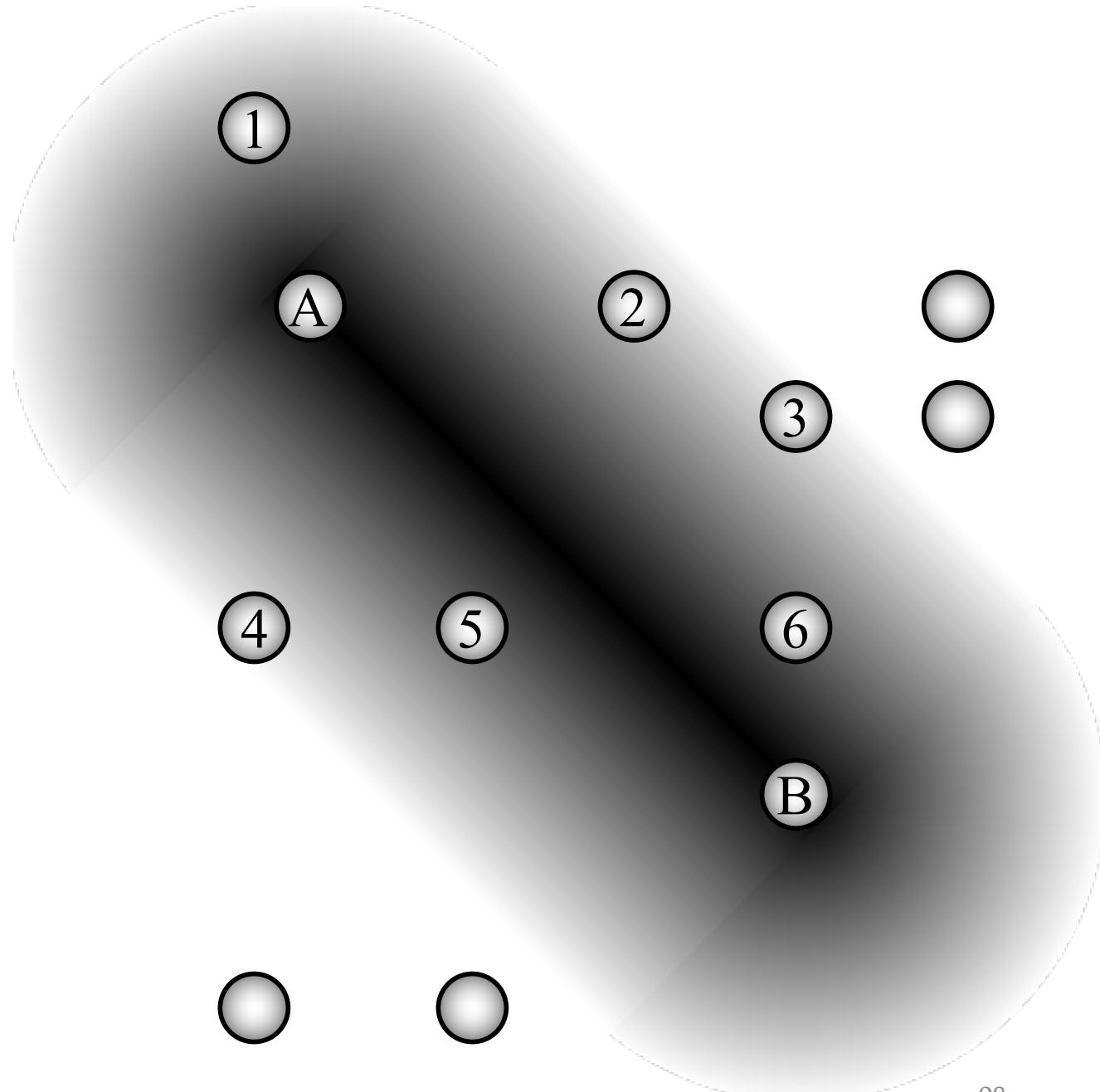
# Proxy Graph

- Find splitter and delay element locations to connect A and B
- Cells closest to line added to proxy graph
- Create proxy graph
  - Nodes are vacant cells
  - Edge weights are Manhattan distance between cells
  - No edge between A and B
- Analyze paths from A to B
  - Choose path minimizing
    - $|\Delta\tau - \Delta t|$
    - $\Delta\tau$  – difference in required arrival time
    - $\Delta t$  – difference in actual arrival time



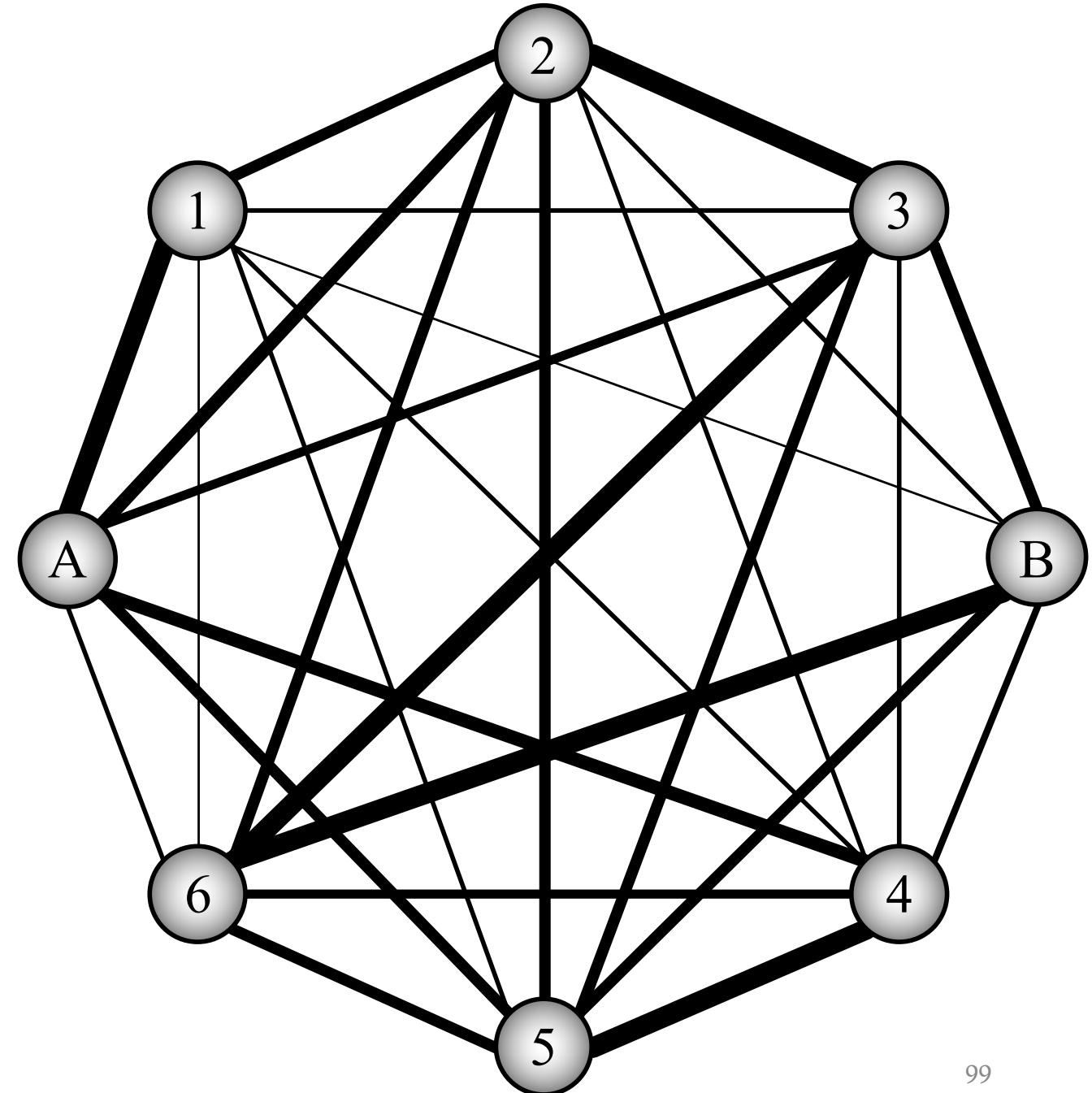
# Proxy Graph

- Find splitter and delay element locations to connect A and B
- Cells closest to line added to proxy graph
- Create proxy graph
  - Nodes are vacant cells
  - Edge weights are Manhattan distance between cells
  - No edge between A and B
- Analyze paths from A to B
  - Choose path minimizing
    - $|\Delta\tau - \Delta t|$
    - $\Delta\tau$  – difference in required arrival time
    - $\Delta t$  – difference in actual arrival time



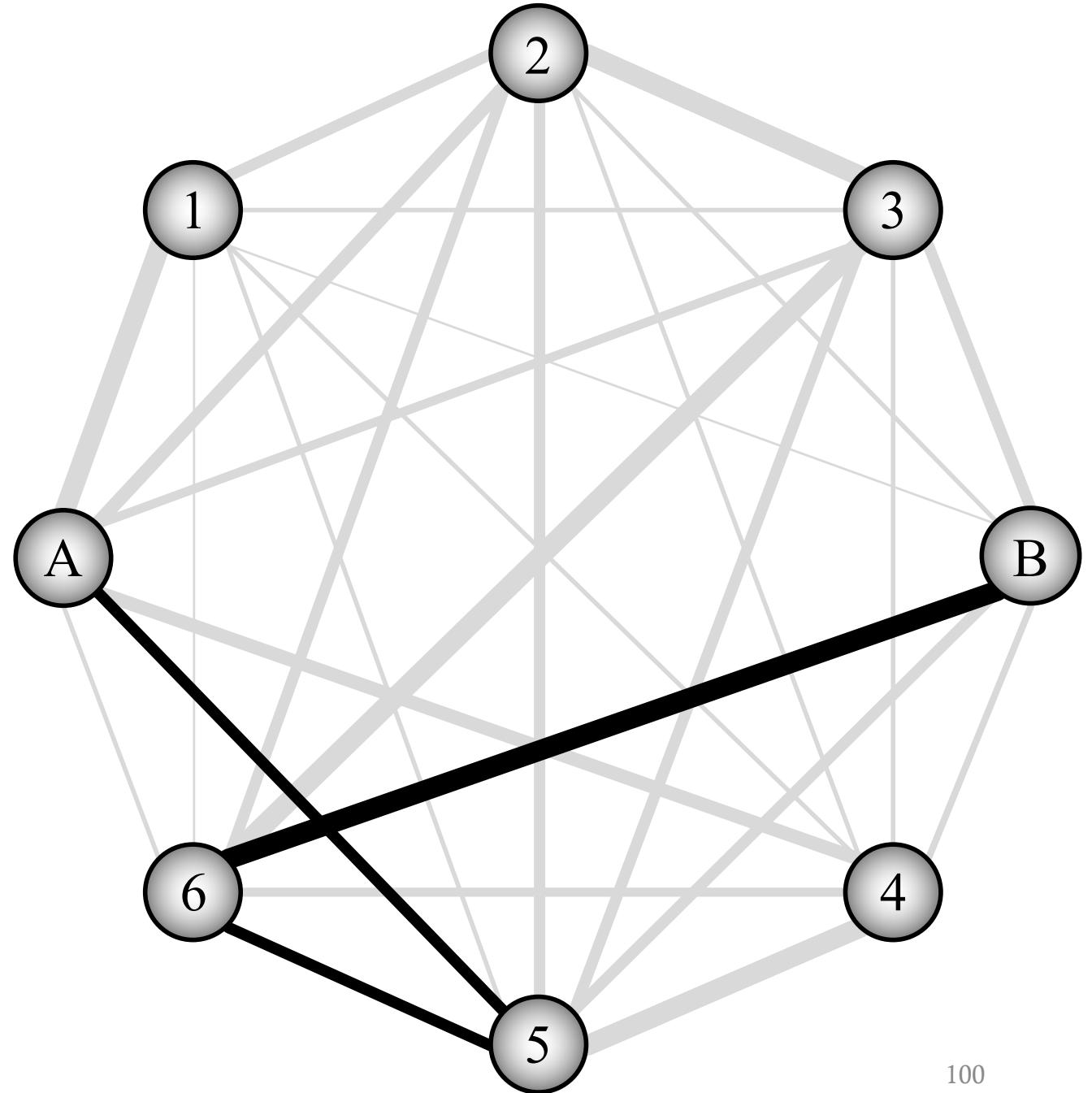
# Proxy Graph

- Find splitter and delay element locations to connect A and B
- Cells closest to line added to proxy graph
- Create proxy graph
  - Nodes are vacant cells
  - Edge weights are Manhattan distance between cells
  - No edge between A and B
- Analyze paths from A to B
  - Choose path minimizing
    - $|\Delta\tau - \Delta t|$
    - $\Delta\tau$  – difference in required arrival time
    - $\Delta t$  – difference in actual arrival time



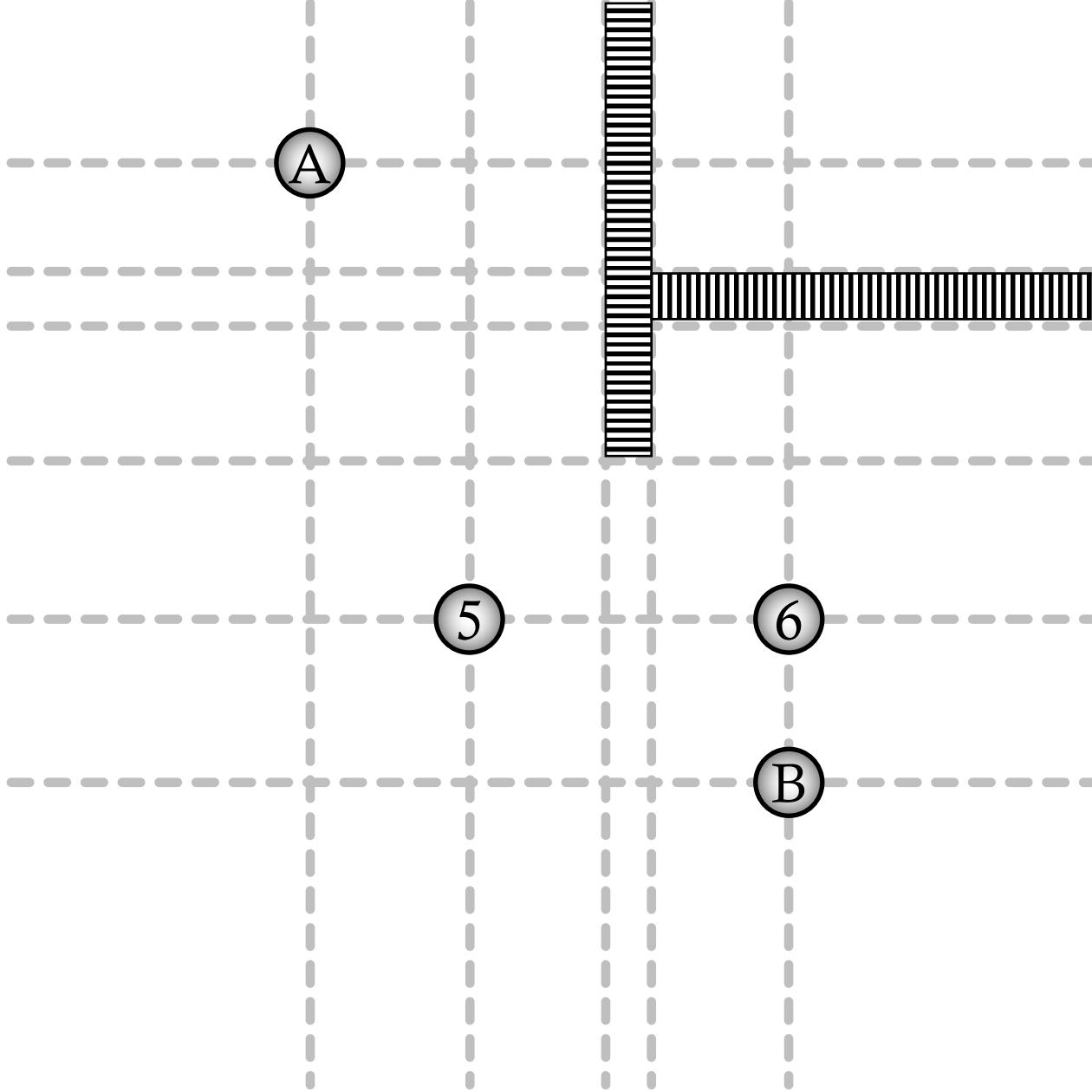
# Proxy Graph

- Find splitter and delay element locations to connect A and B
- Cells closest to line added to proxy graph
- Create proxy graph
  - Nodes are vacant cells
  - Edge weights are Manhattan distance between cells
  - No edge between A and B
- Analyze paths from A to B
  - Choose path minimizing
    - $|\Delta\tau - \Delta t|$
    - $\Delta\tau$  – difference in required arrival time
    - $\Delta t$  – difference in actual arrival time



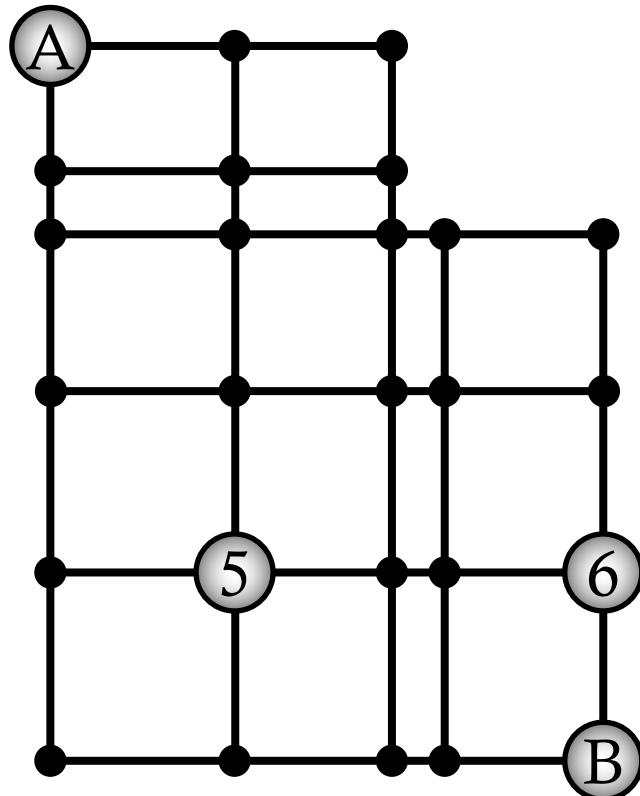
# Hanan Grid

- Embed path A-5-6-B
- Vertical and horizontal lines through
  - Nodes A and B
  - Splitter
  - JTLs
  - Obstacle boundaries
- Routing algorithms determine path



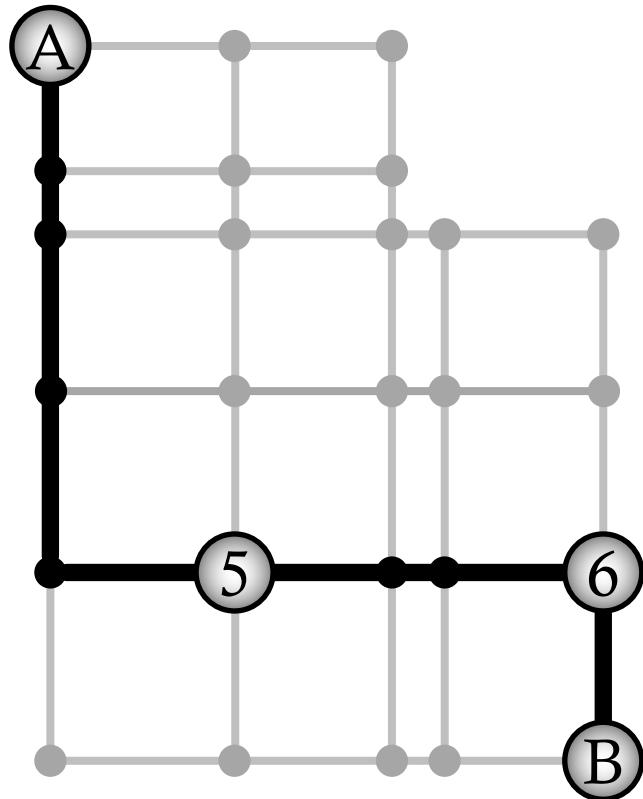
# Hanan Grid Routing

- Embed path A-5-6-B
- Vertical and horizontal lines through
  - Nodes A and B
  - Splitter
  - JTLs
  - Obstacle boundaries
- Routing algorithms determine path



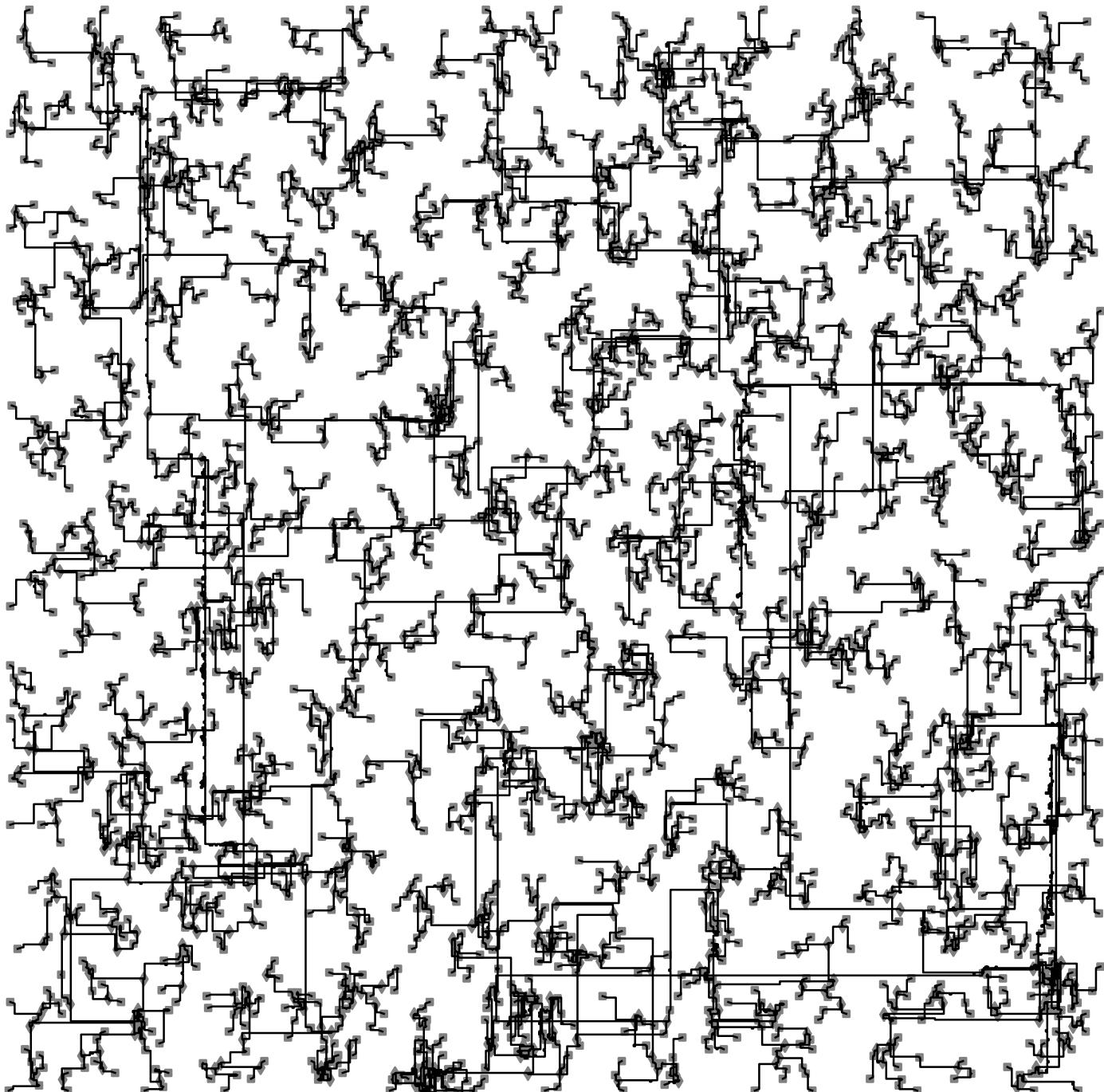
# Hanan Grid Routing

- Embed path A-5-6-B
- Vertical and horizontal lines through
  - Nodes A and B
  - Splitter
  - JTLs
  - Obstacle boundaries
- Routing algorithms determine path



# Synthesis Results

- Benchmark circuits
  - ISCAS'89
  - ITC'99
  - AMD2901
    - 1,050 Clocked gates
    - 52 minutes runtime



# Benchmark Circuits

Circuit	Clocked gates	Delay elements	Total wirelength, mm	Runtime, minutes
AMD2901	1,049	1,241	1,027	53
ISCAS'89 S13207	1,636	2,405	272	73
ITC'99 B14	6,365	5,762	905	212
ISCAS'89 S38417	11,796	11,367	1,002	393
ISCAS'89 S35932	14,914	15,814	6,656	372
ITC'99 B18	45,710	71,090	31,736	2,309

**Research Effort  
since 2017**

**VLSI CONTINENT**

**CRYOGENIC  
OCEAN**



# **Placement of Distributed On-Chip Voltage Regulators**

**SOFTWARE  
EMPIRE**

**ASSEMBLY CHANNEL**

**LOGIC  
SYNTHESIS**

**POWER**

**PHYSICAL  
DESIGN**

**DEVICE  
PHYSICS**

**CIRCUIT  
THEORY**

**NONVOLATILE  
SEA**

**SYNCHRONIZATION**

**SUPERCONDUCTIVE  
UNION**

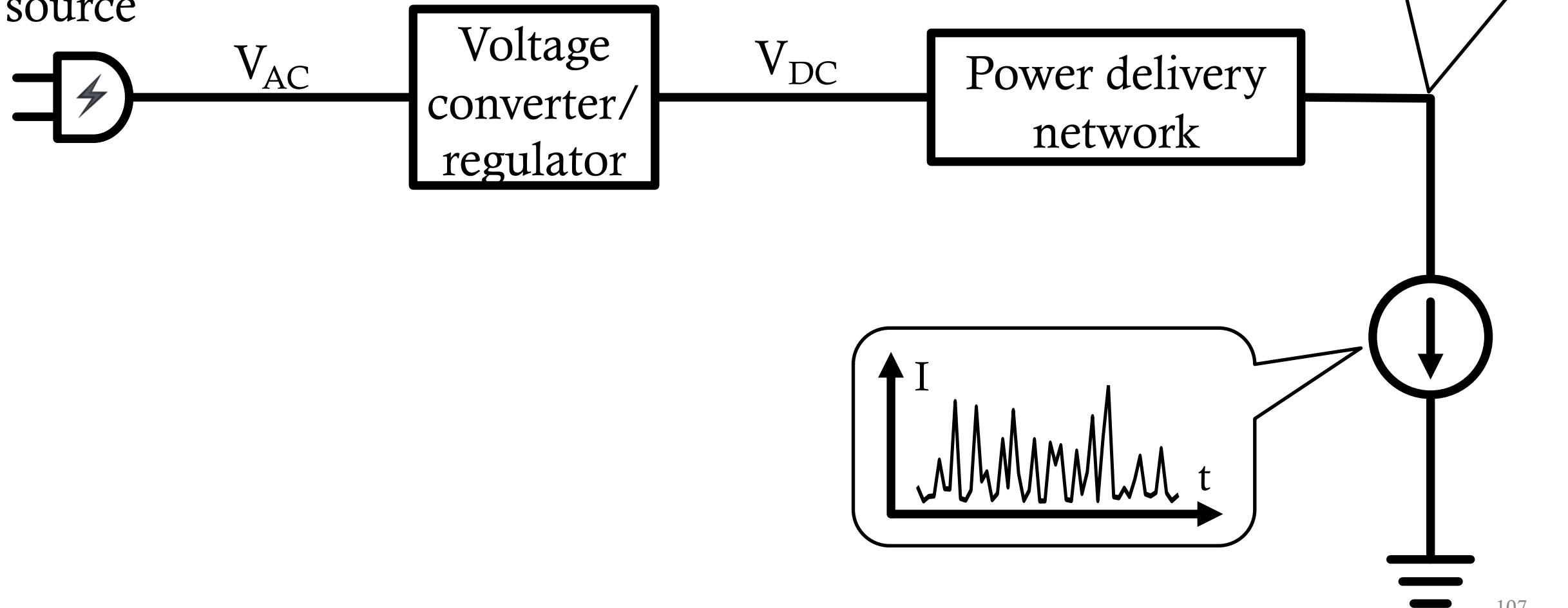
**MEMRISTOR  
ISLAND**

**SPINTRONIC  
ISLAND**

**QUANTUM  
COMPUTING**

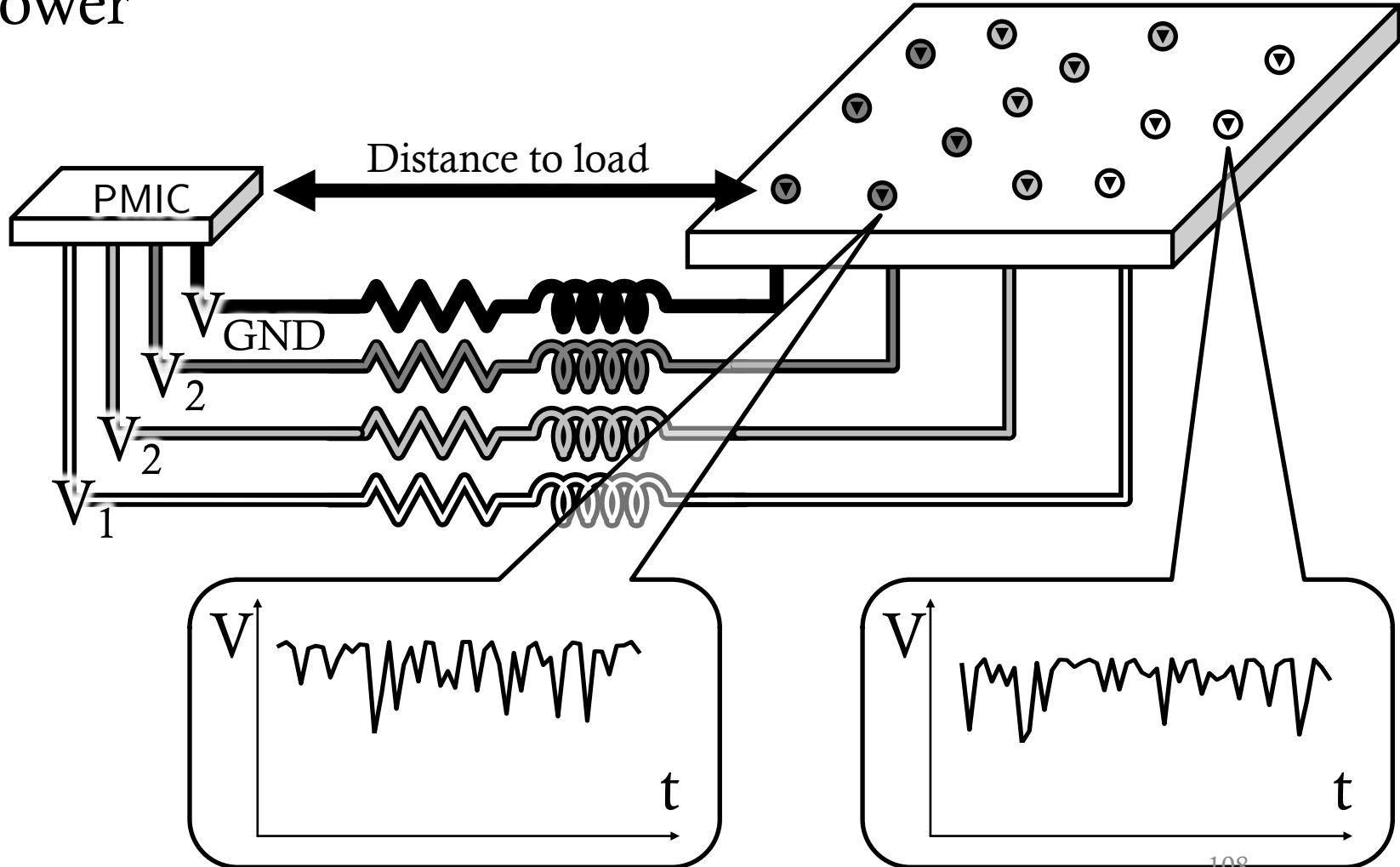
# Noise in VLSI Power Delivery

External  
source



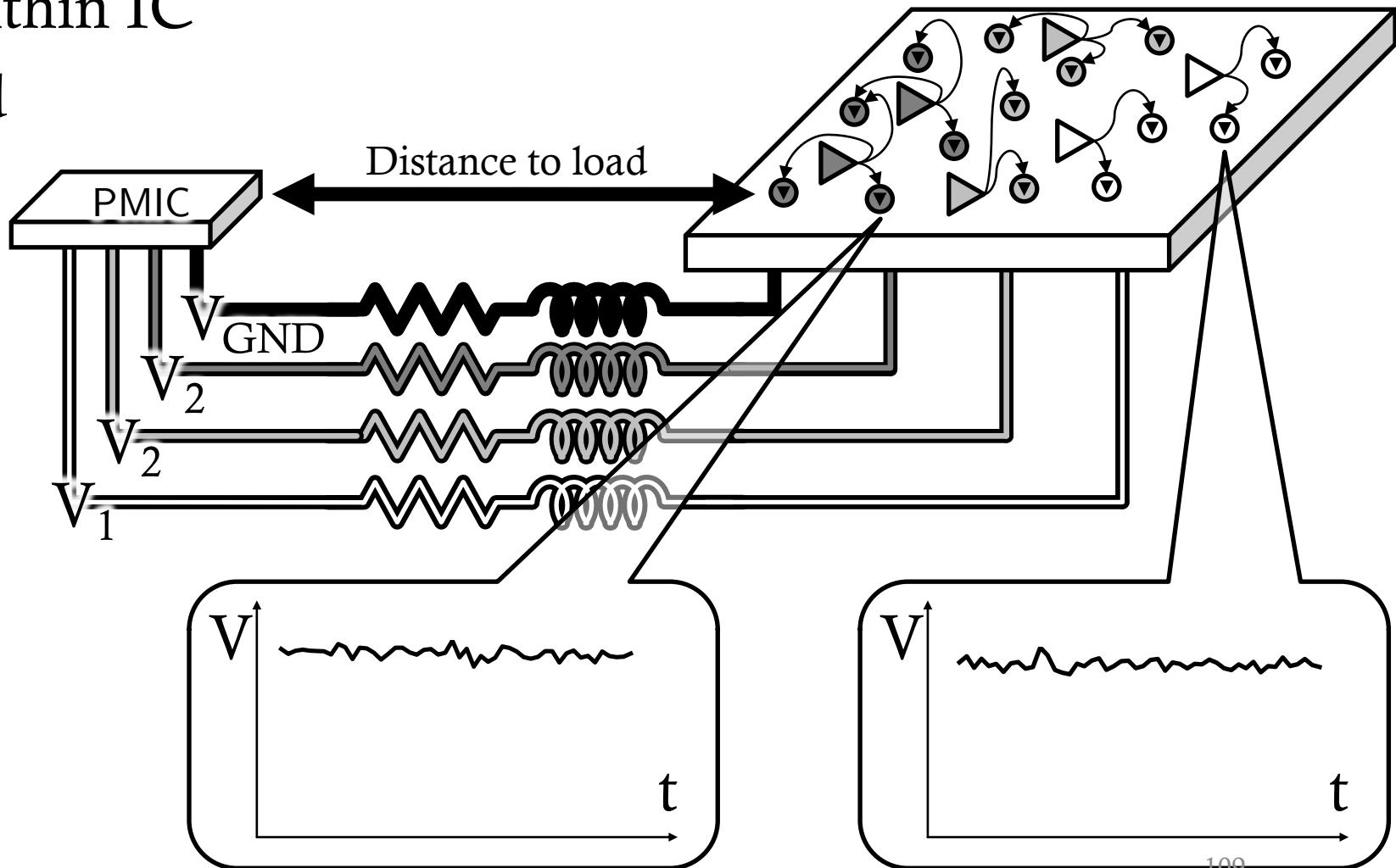
# Conventional Power Delivery

- Single regulator within power management IC (PMIC)
- Large distance to load
- Poor regulation capability



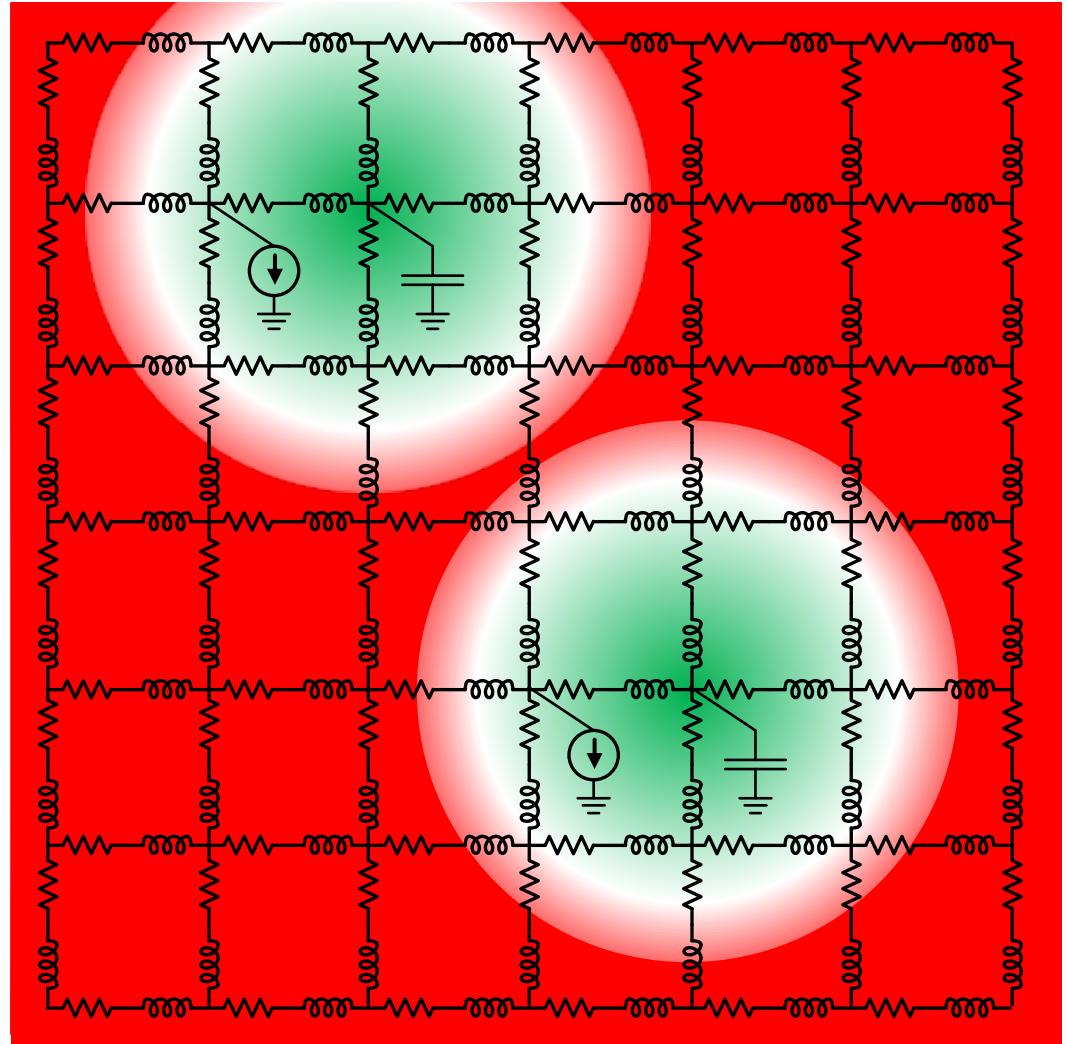
# Distributed Power Delivery

- Distributed regulators within IC
- Minimal distance to load
- Superior regulation capability



# Placement for Power Delivery: Capacitors

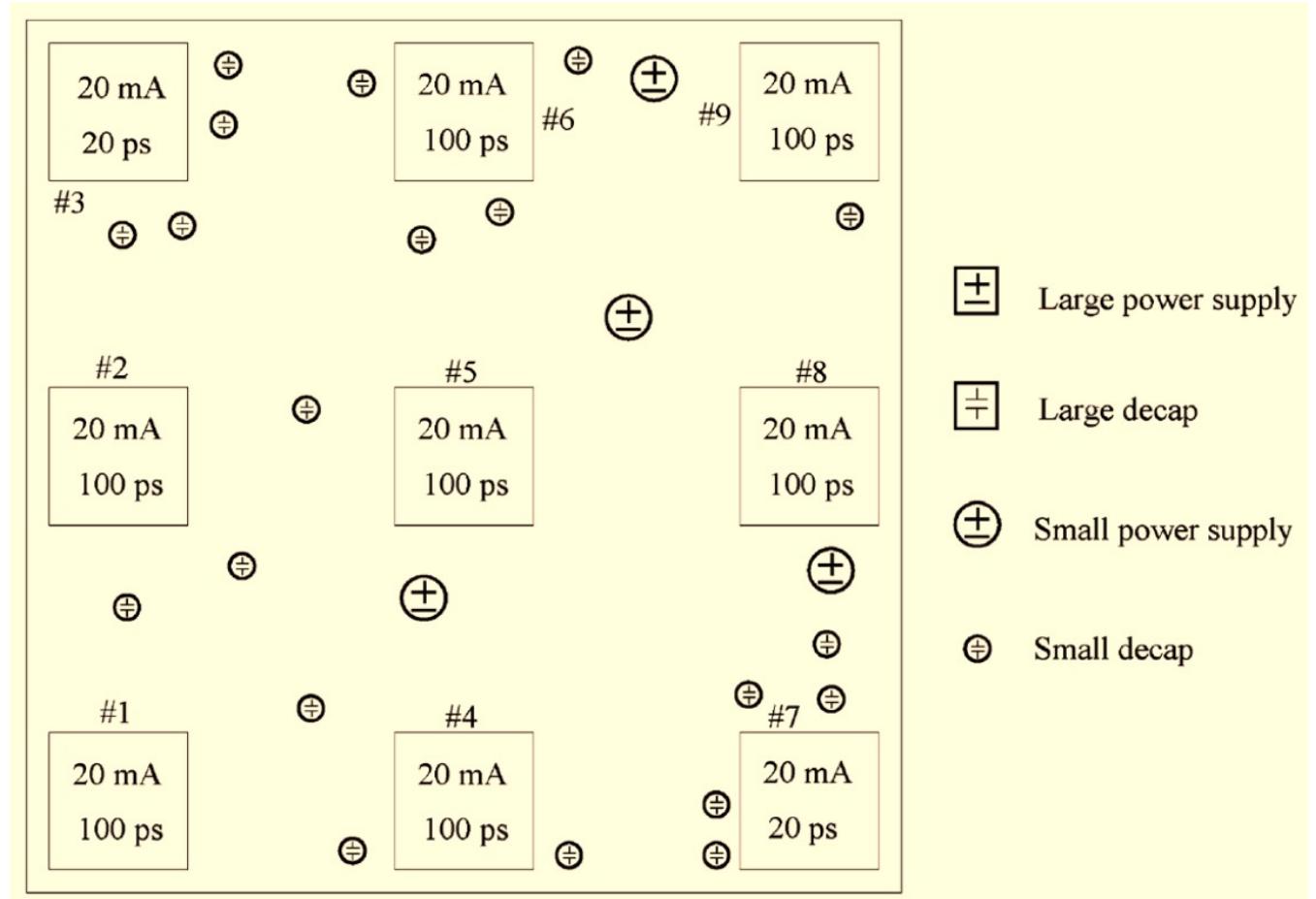
- Effectiveness of a capacitor depends on distance from load
  - Effective radius of decoupling capacitor



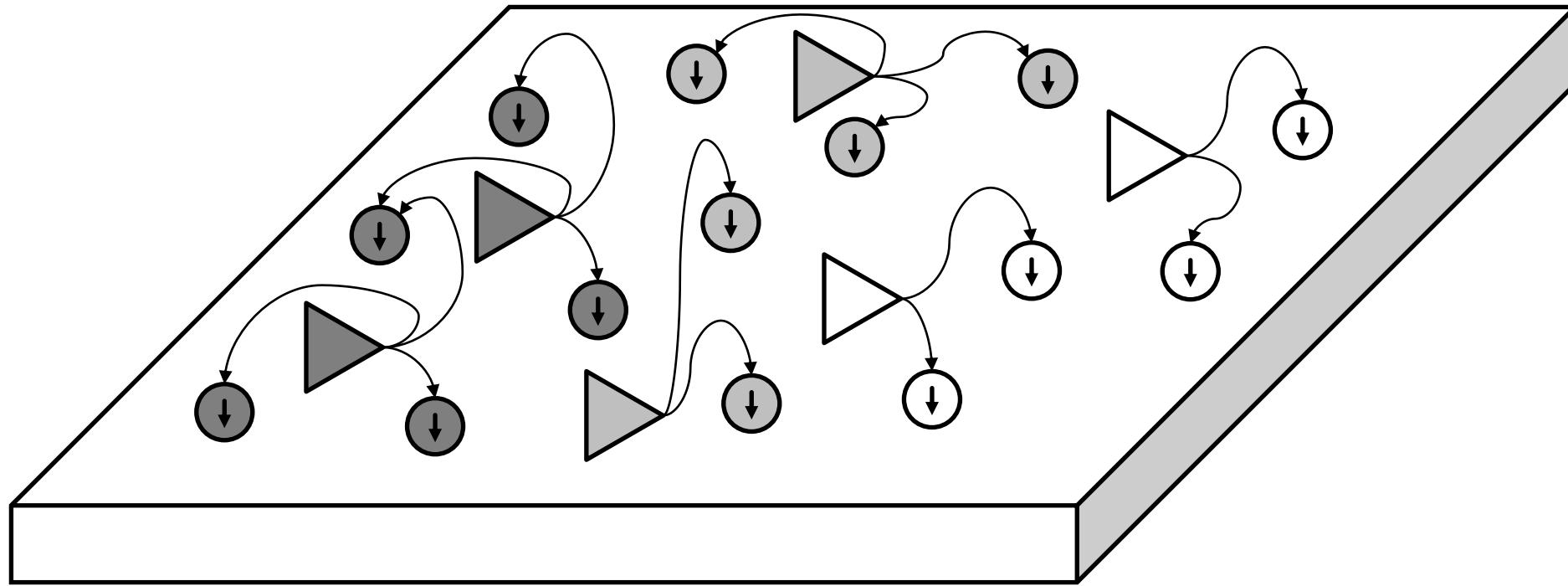
M. Popovich, M. Sotman, A. Kolodny, and E. G. Friedman,  
"Effective Radii of On-Chip Decoupling Capacitors,"  
*IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 16, No. 7, pp. 894-907, July 2008

# Placement for Power Delivery: Capacitors and I/O

- Distributed power delivery
  - decoupling capacitors
  - I/O

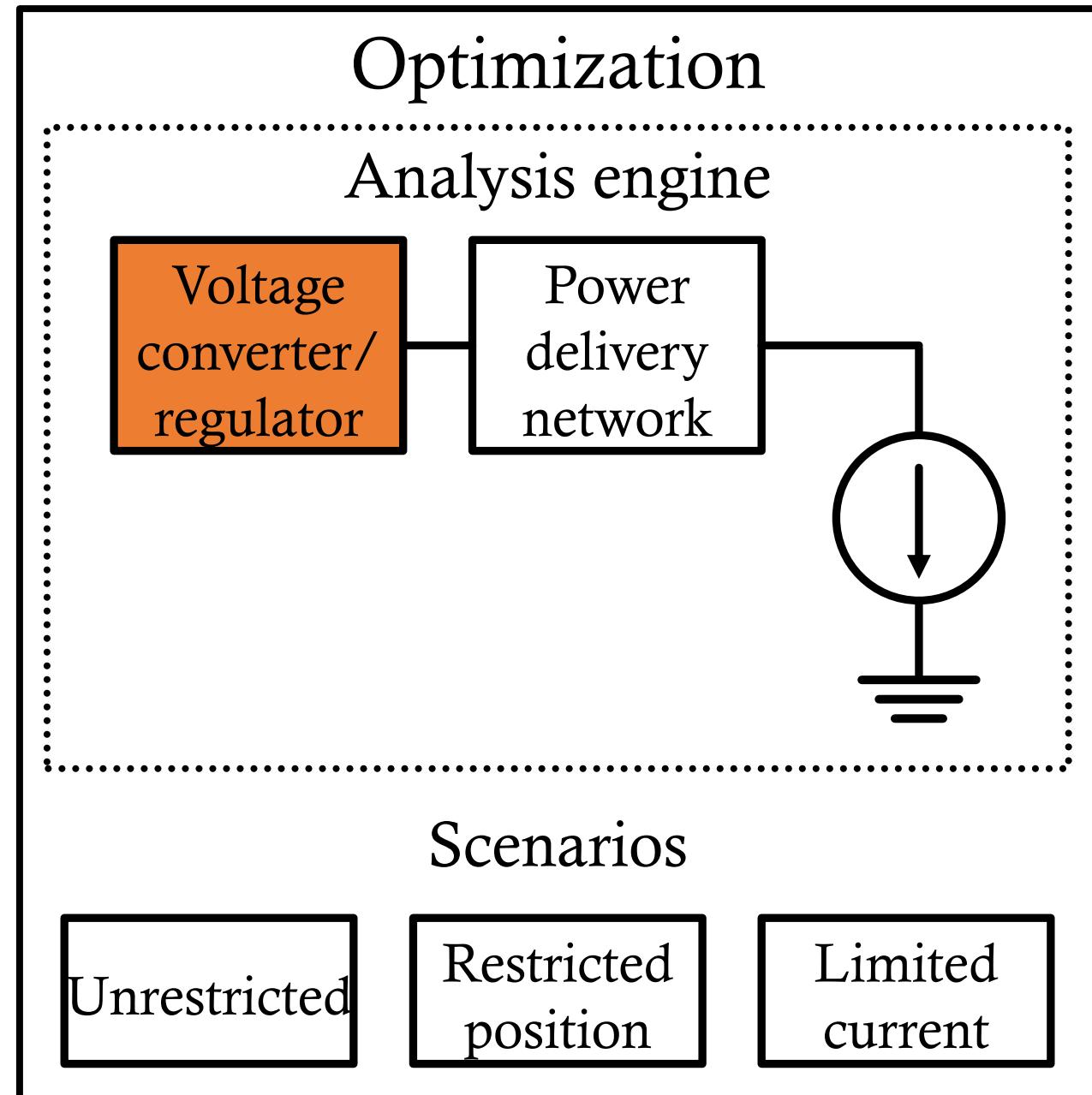


# Placement for Power Delivery: Voltage Regulators



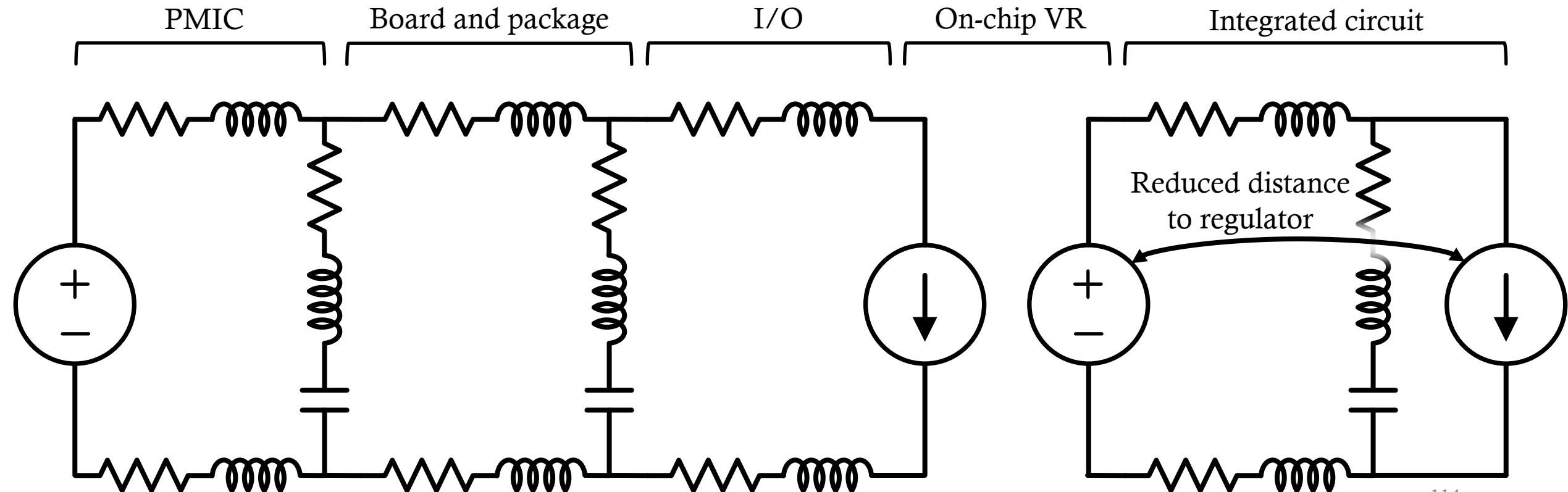
# Placement Procedure

- Objective
  - Minimize worst voltage drop within power network
- Constraints
  - Case one: no restrictions
  - Case two: prohibited areas within the grid
  - Case three: limited regulator current



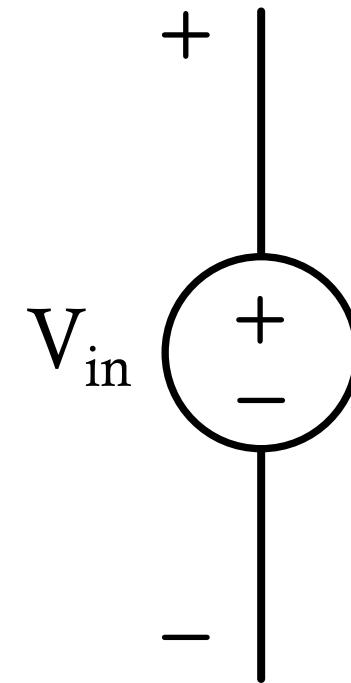
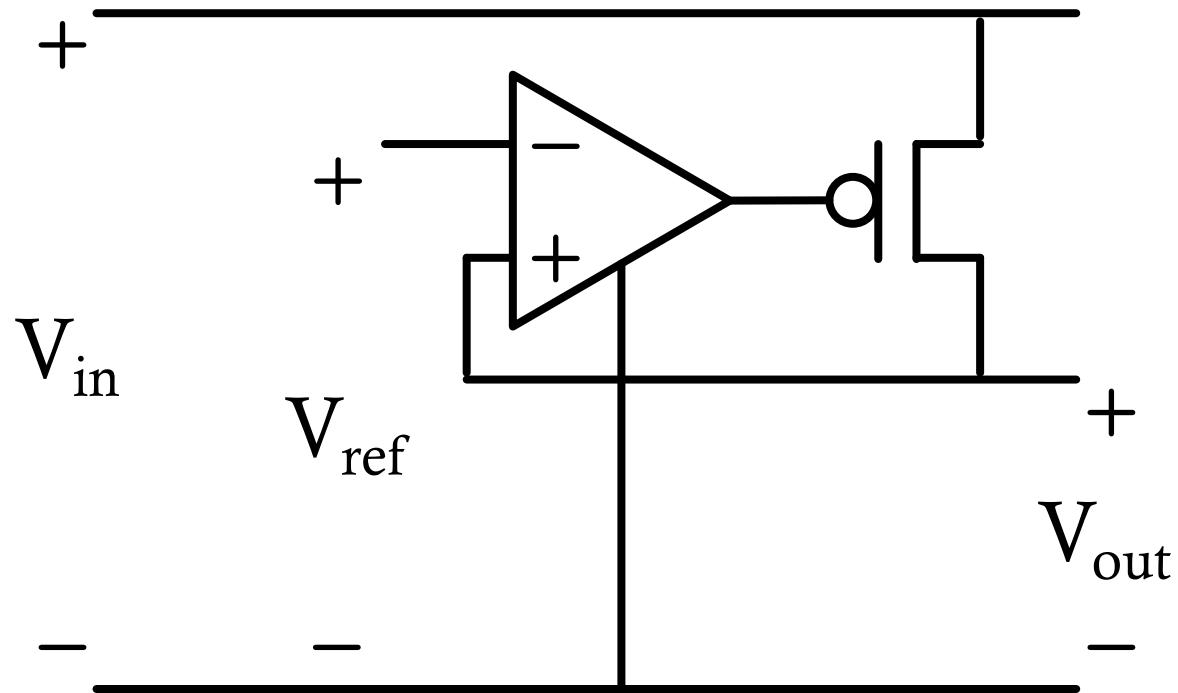
# Regulation Mechanism

- Load circuitry effectively separated from input
- Stable output voltage (input regulation/load regulation)
- Analyze load part of power network



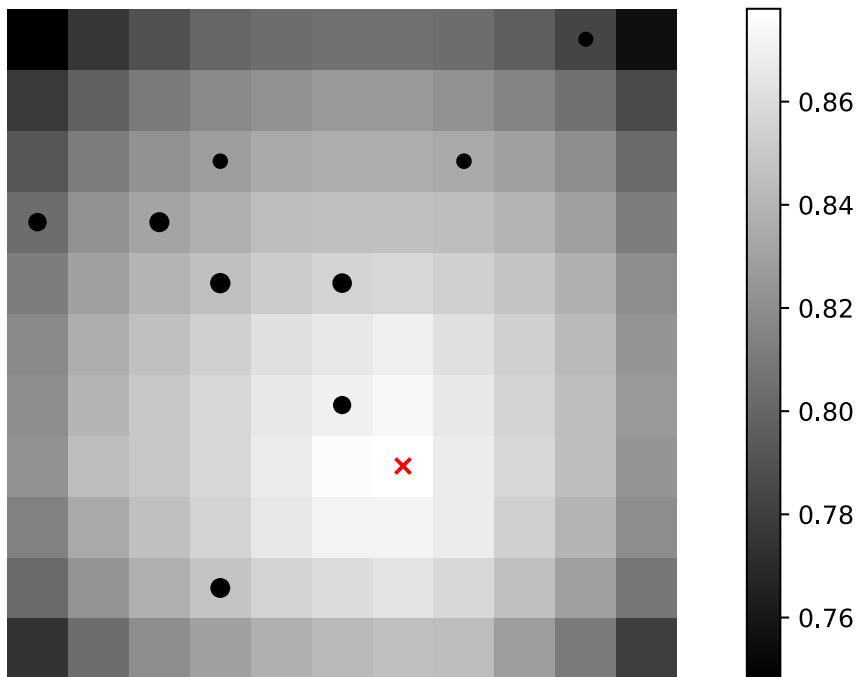
# Voltage Regulator Model

- Detailed model
  - Accurate behavior
  - Computationally expensive
- Simplified model
  - Poor accuracy
  - Computationally efficient
  - **High fidelity**

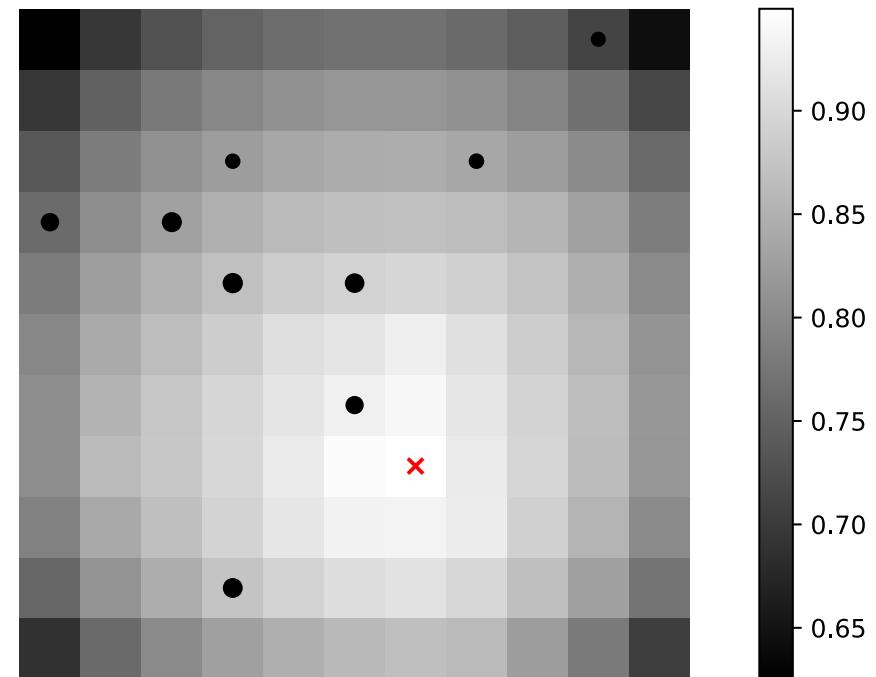


# Fidelity of Regulator Model

- $20 \times 20$  resistive-inductive mesh
- Ten randomly distributed loads
- Place regulator to minimize voltage drop
- Voltage estimated using HSPICE

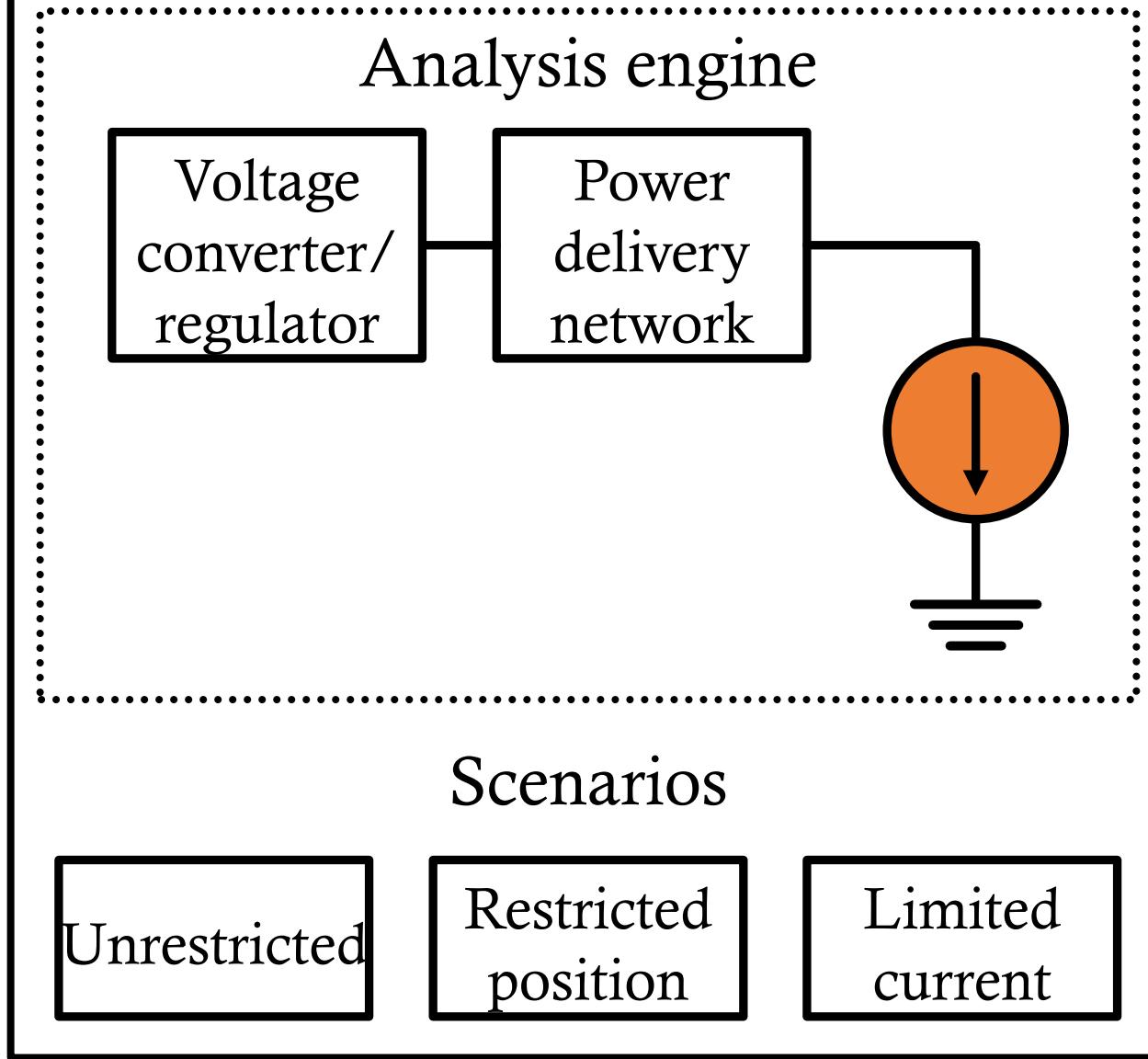


Detailed model



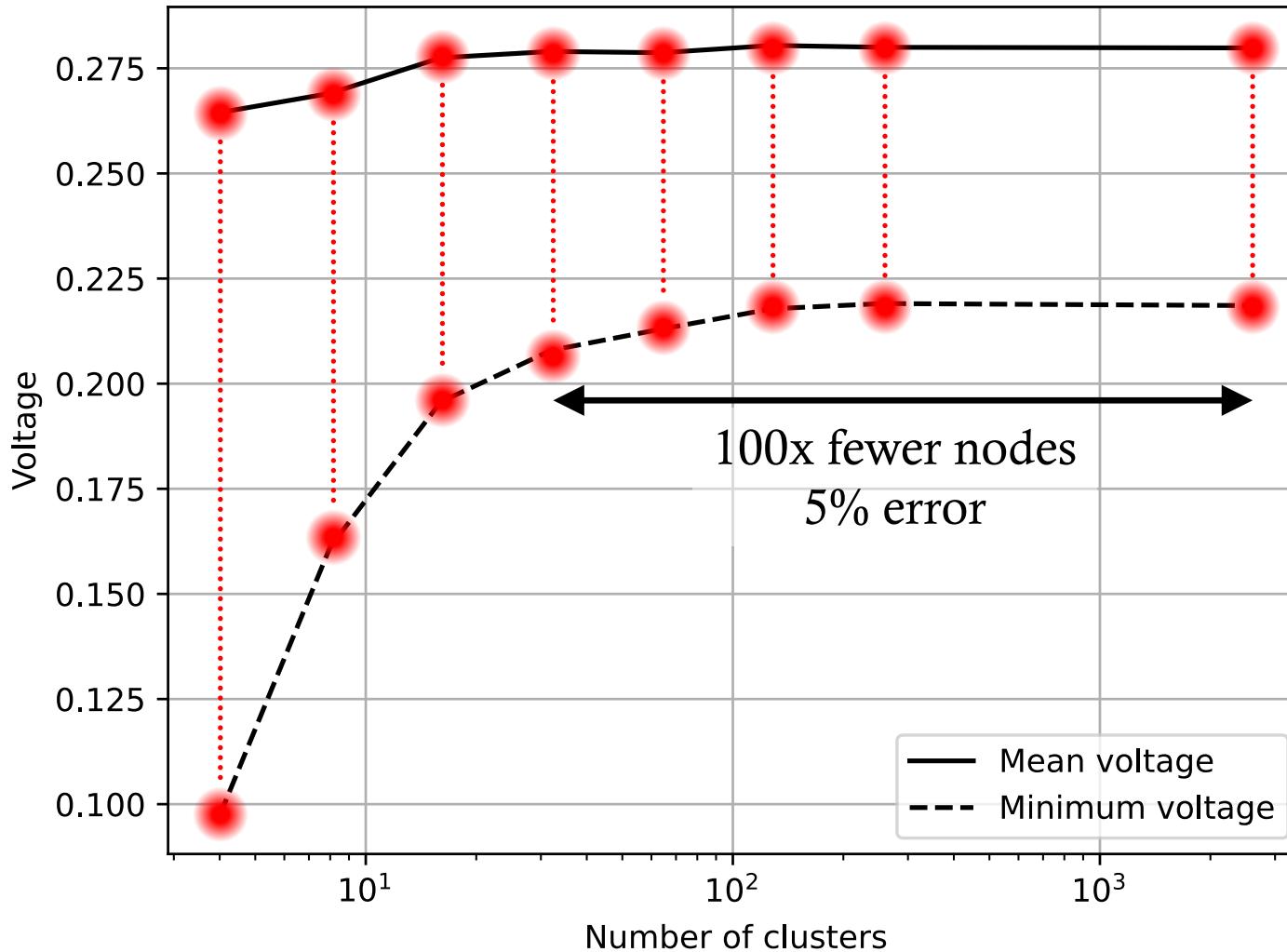
Voltage source model

# Optimization process

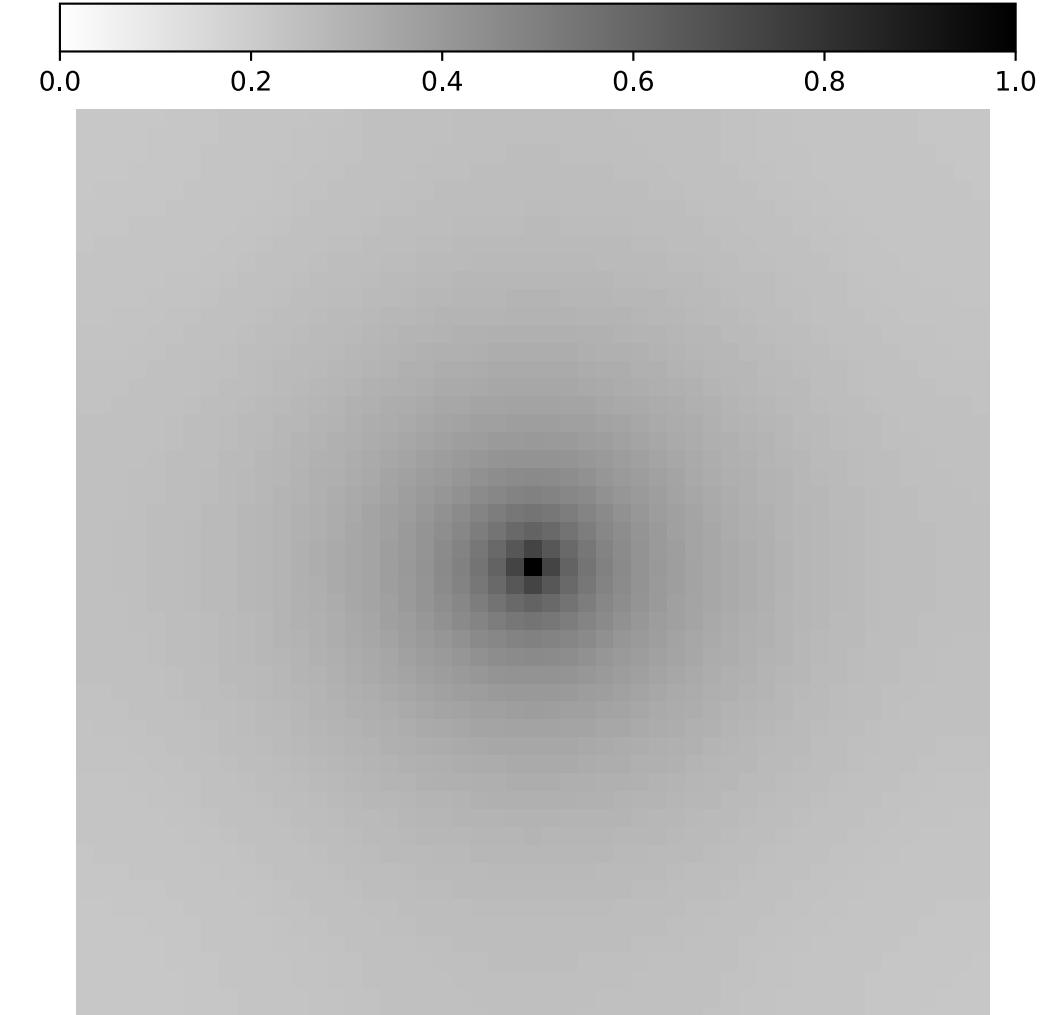


# Load Clustering

- Practical grids contain billions of loads
- Reduce number of loads by clustering

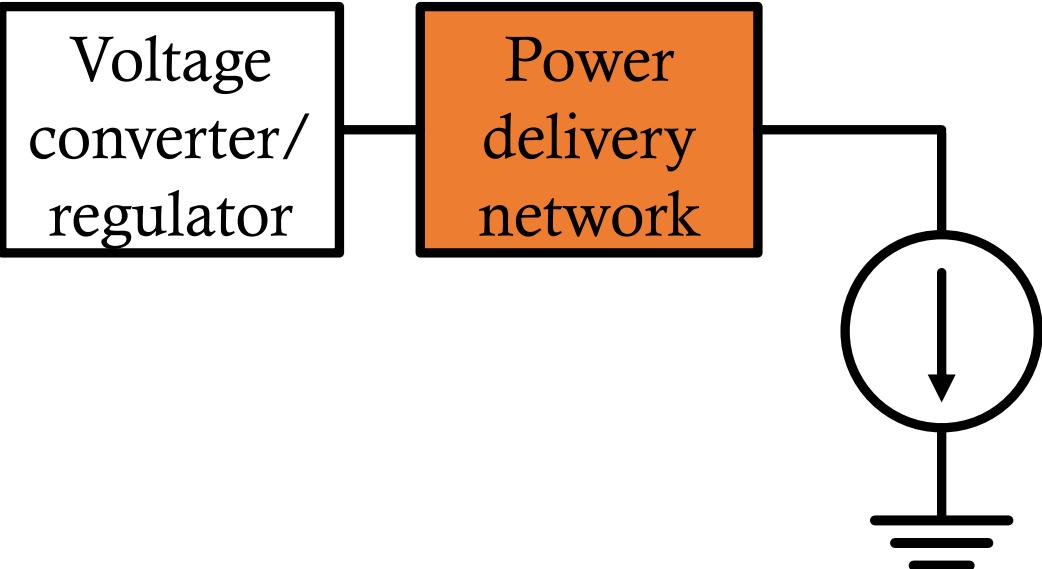


# loads : 4  
min voltage : 97 mV  
mean voltage : 264 mV



# Optimization

## Analysis engine



## Scenarios

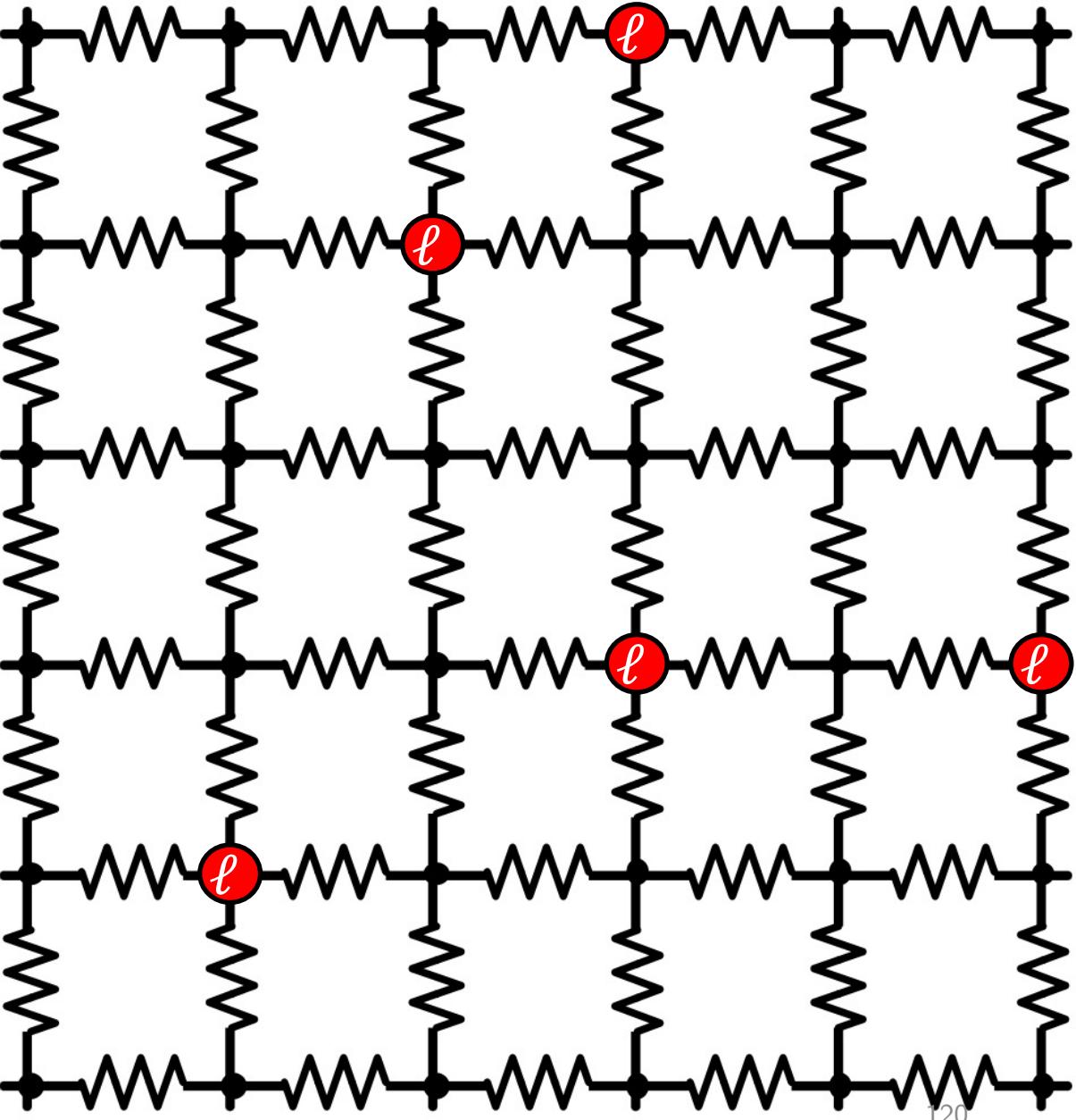
Unrestricted

Restricted position

Limited current

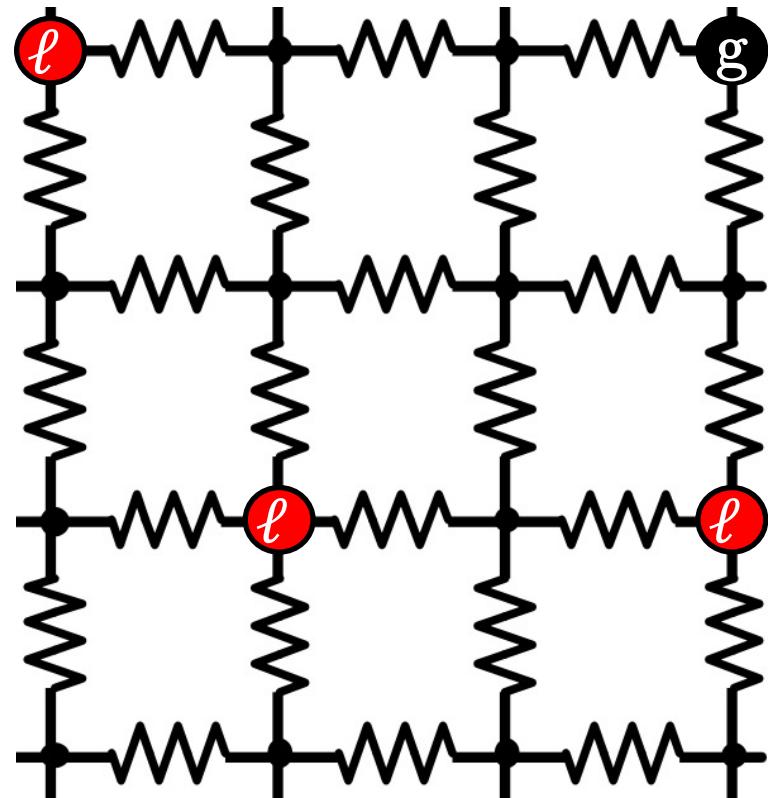
# Efficient Grid Analysis

- MNA-based analysis computationally expensive
- Accelerate using infinity mirror technique



# Load Model

- Loads modeled as current sources
- Loads
$$\mathcal{L} = \{\ell_p \mid p \in [1, \dots, n]\}$$
$$\ell_p = (x_p, y_p, i_p)$$



# Load Model

- Loads modeled as current sources



- Loads

$$\mathcal{L} = \{\ell_p \mid p \in [1, \dots, n]\}$$

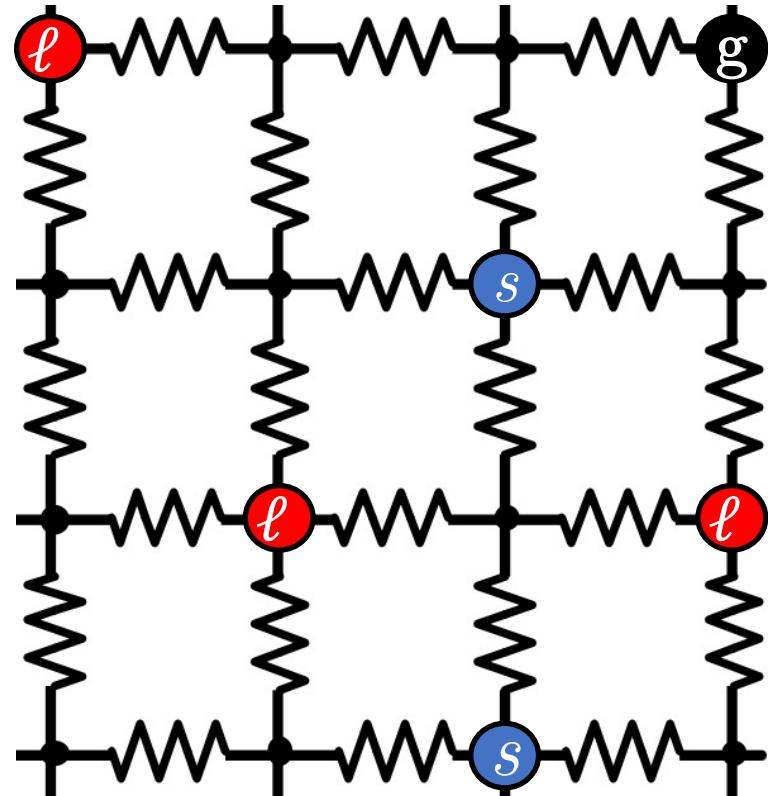
$$\ell_p = (x_p, y_p, i_p)$$

# Source Model

- Source modeled as unknown current sources

$$\mathcal{S} = \{s_q | q \in [1, \dots, m]\}$$

- How to find current from voltage source?



# Superposition

- Any source/load distorts voltage within grid
- Voltage at any node as superposition of distortions

$$V_r^g = \sum_{q=1}^m I(s_q) v^g(s_r, s_q) + \sum_{p=1}^n I(\ell_p) v^g(s_r, \ell_p)$$

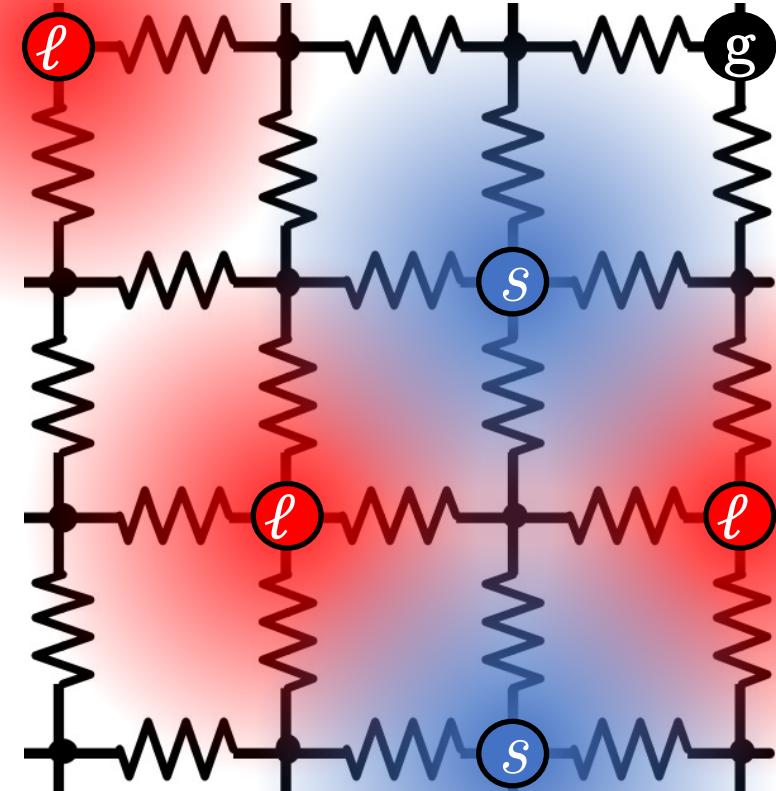
Target voltage

Unknown source current

Influence of source  $q$  on point  $r$

Current at load  $p$

Influence of load  $p$  on point  $r$



# Matrix Equation

- Solved for  $I(s_i)$
- Polynomial complexity with number of sources  $\sim O(m^3)$

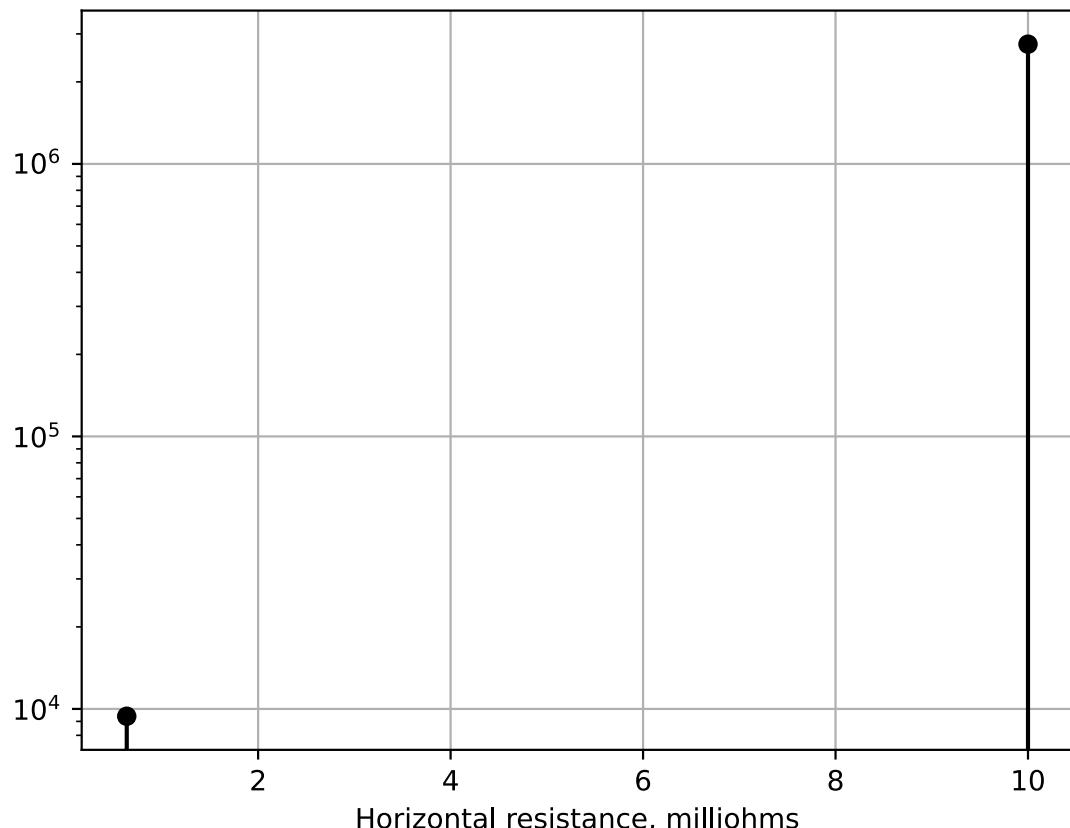
$$\begin{bmatrix} v^g(s_1, s_1) & \dots & v^g(s_m, s_1) \\ \vdots & \ddots & \vdots \\ v^g(s_1, s_{m-1}) & \dots & v^g(s_m, s_{m-1}) \\ 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} I(s_1) \\ \vdots \\ I(s_m) \end{bmatrix} = \begin{bmatrix} V_1^g \\ \vdots \\ V_{m-1}^g \\ 0 \end{bmatrix} - \begin{bmatrix} v^g(\ell_1, s_1) & \dots & v^g(\ell_n, s_1) \\ \vdots & \ddots & \vdots \\ v^g(\ell_1, s_{m-1}) & \dots & v^g(\ell_n, s_{m-1}) \\ 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} I(\ell_1) \\ \vdots \\ I(\ell_n) \end{bmatrix}$$

- Load voltage

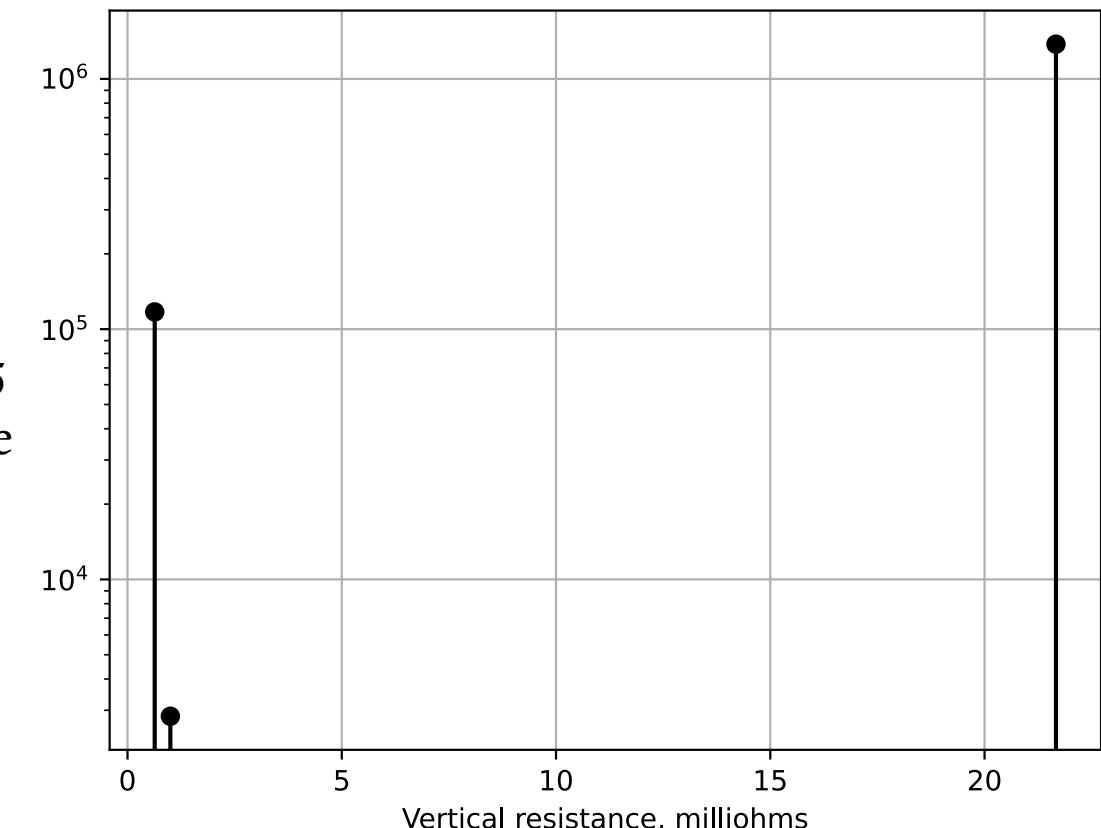
$$\begin{bmatrix} v^g(\ell_1) \\ \vdots \\ v^g(\ell_{n-1}) \end{bmatrix} = \begin{bmatrix} v^g(\ell_1, \ell_1) & \dots & v^g(\ell_n, \ell_1) \\ \vdots & \ddots & \vdots \\ v^g(\ell_1, \ell_{m-1}) & \dots & v^g(\ell_n, \ell_{m-1}) \end{bmatrix} \begin{bmatrix} I(\ell_1) \\ \vdots \\ I(\ell_n) \end{bmatrix} + \begin{bmatrix} v^g(s_1, \ell_1) & \dots & v^g(s_m, \ell_1) \\ \vdots & \ddots & \vdots \\ v^g(s_1, \ell_{m-1}) & \dots & v^g(s_m, \ell_{m-1}) \end{bmatrix} \begin{bmatrix} I(s_1) \\ \vdots \\ I(s_m) \end{bmatrix}$$

# IBMPG Benchmark Circuits - Resistance

- From 30,638 to 1,670,494 nodes
- Irregular mesh topology
- Approximated as regular grid
- Dominant resistance and pitch

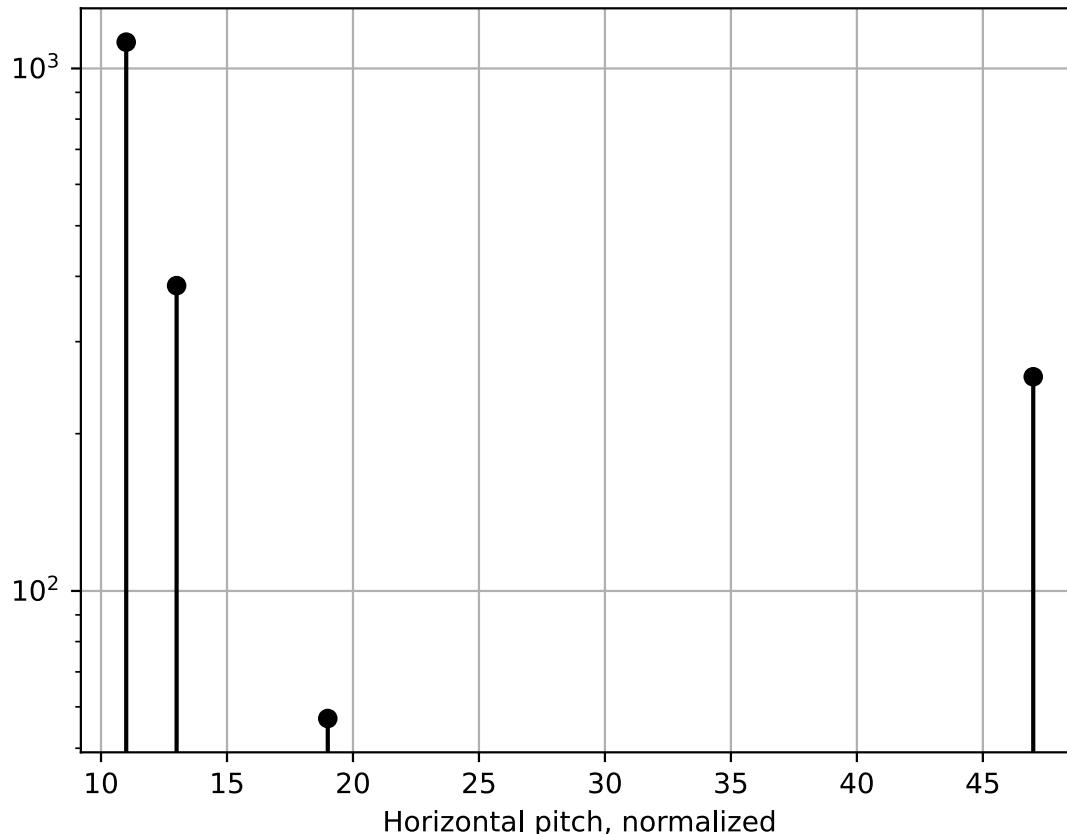


IBMPG 5  
Resistance

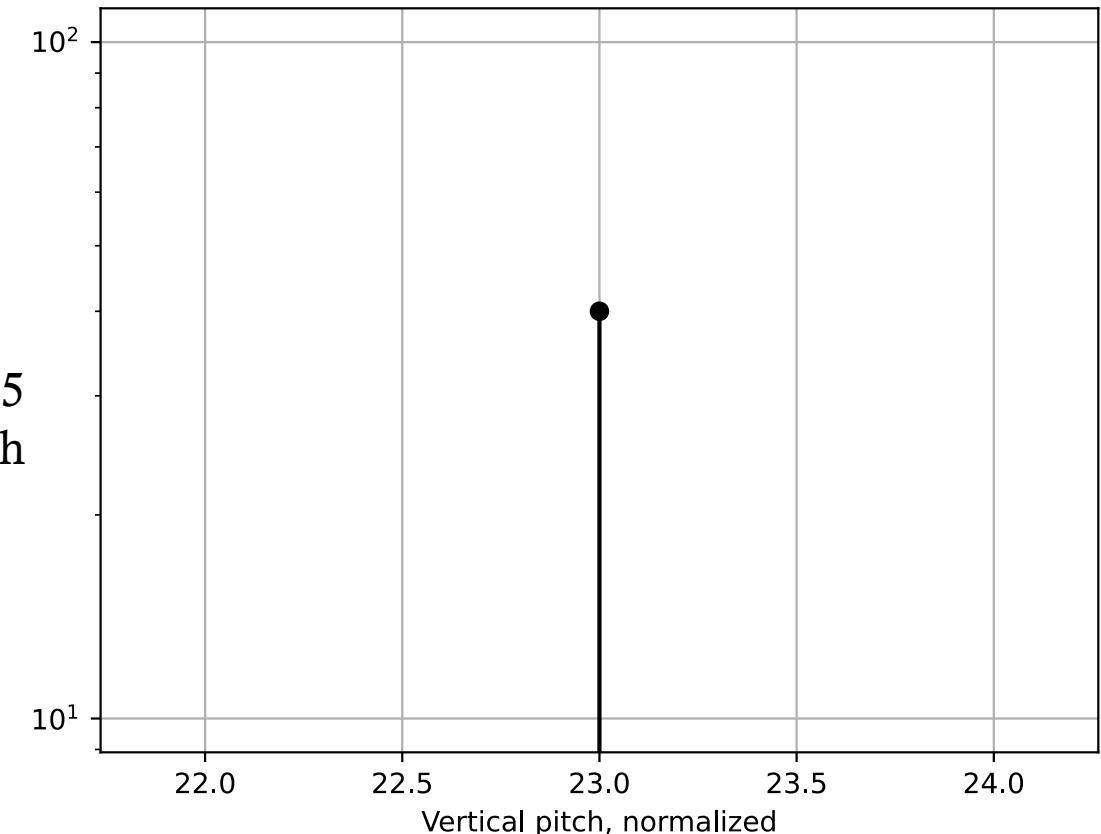


# IBMPG Benchmark Circuits - Pitch

- From 30,638 to 1,670,494 nodes
- Irregular mesh topology
- Approximated as regular grid
- Dominant resistance and pitch



IBMPG 5  
wire pitch

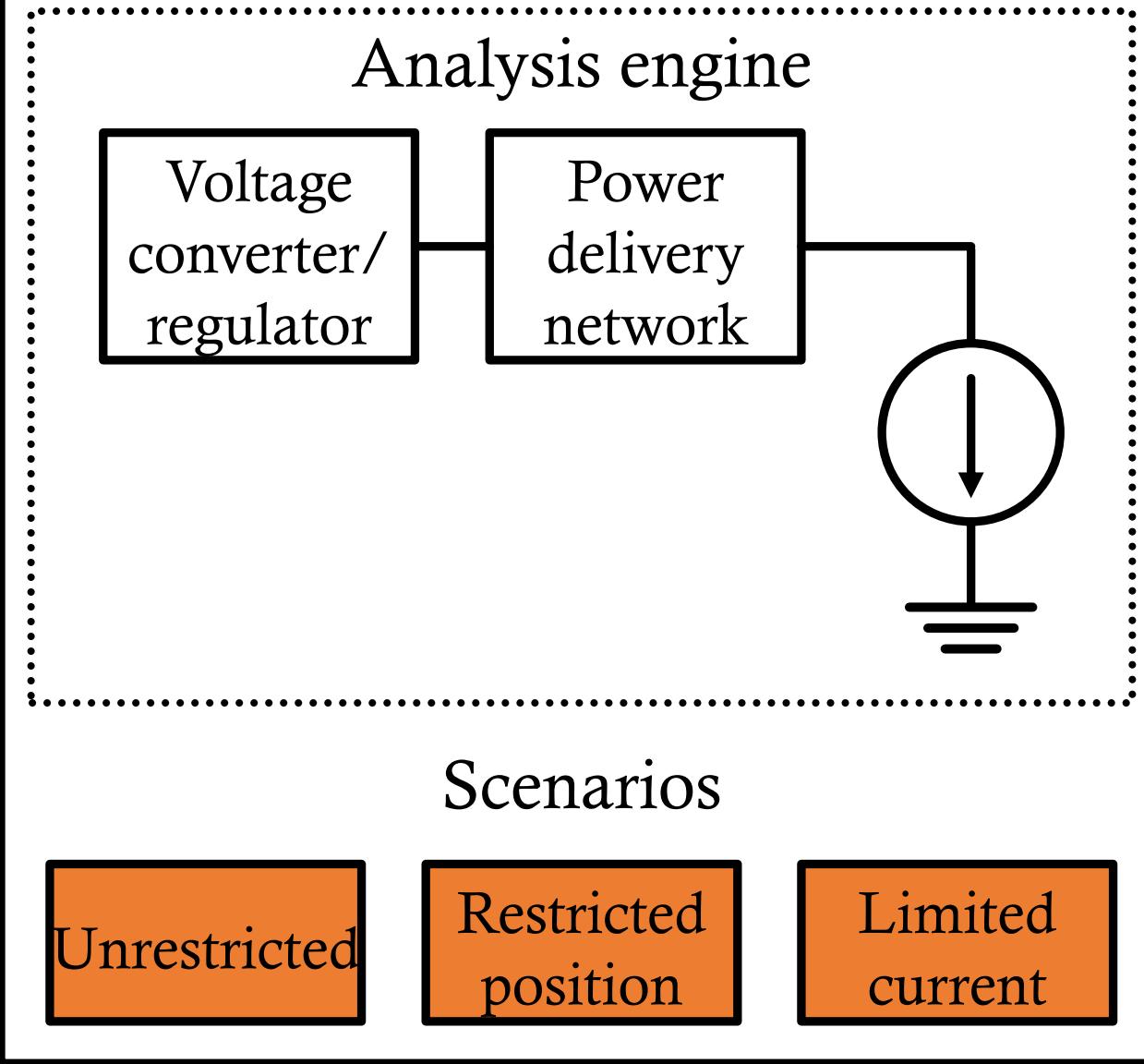


# Simplified Grids

- IBMPG benchmarks modeled as anisotropic grids
- Dominant pitch and resistance

	Pitch		Resistivity, mΩ		Dimensions	
	$x$	$y$	$x$	$y$	$x$	$y$
ibmpg1	2,062	33	5.714	0.635	10	629
ibmpg2	48	72	4.000	16.25	169	113
ibmpg3	864	1,296	0.714	2.407	354	236
ibmpg4	48	24	35.00	32.50	284	571
ibmpg5	82	12	10.00	21.67	129	882
ibmpg6	280	280	0.286	0.464	3,630	3,644

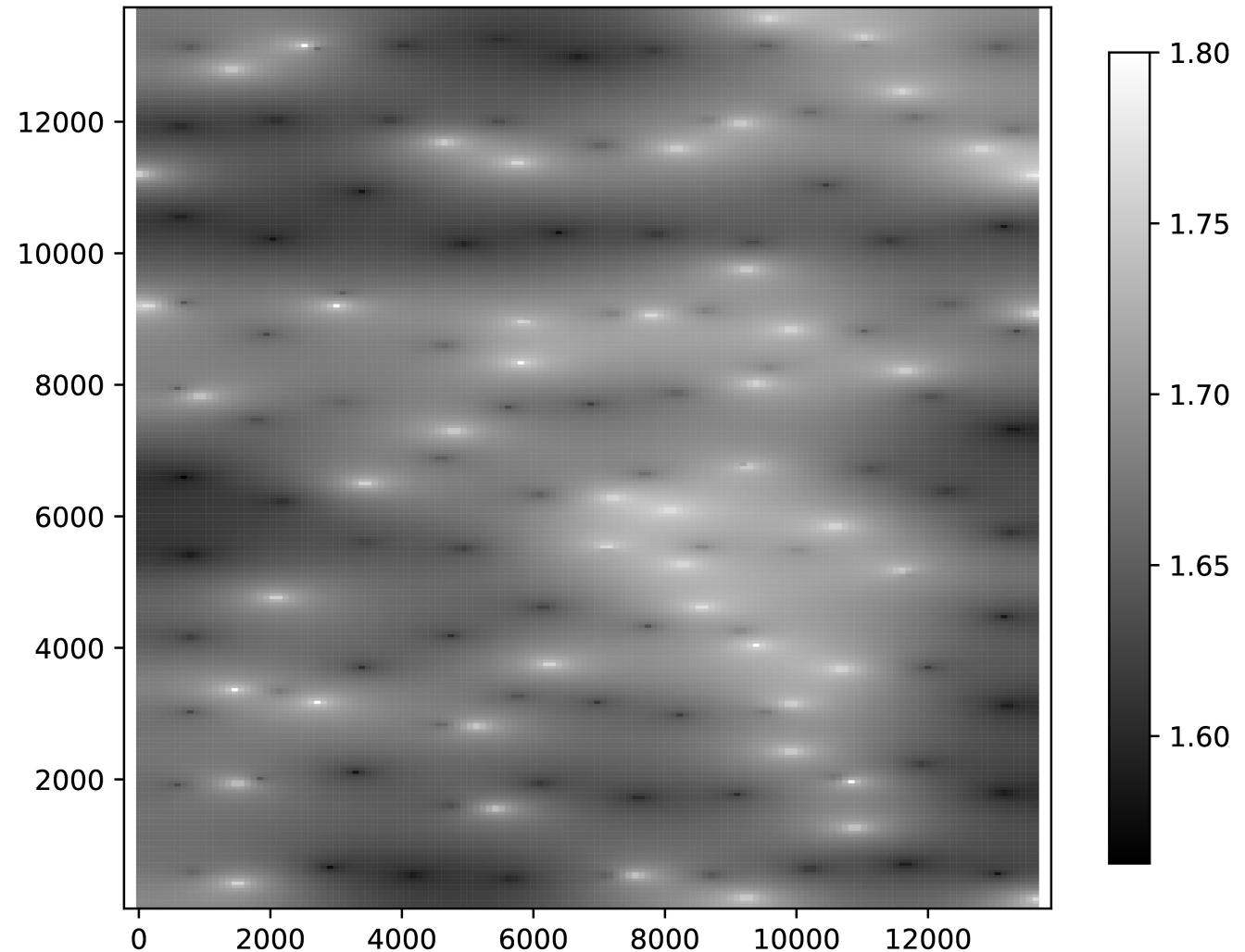
# Optimization process



# Case One

- Unrestricted position and regulator current
- Loads merged into 100 clusters
- Number of regulators varied from 10 to 50

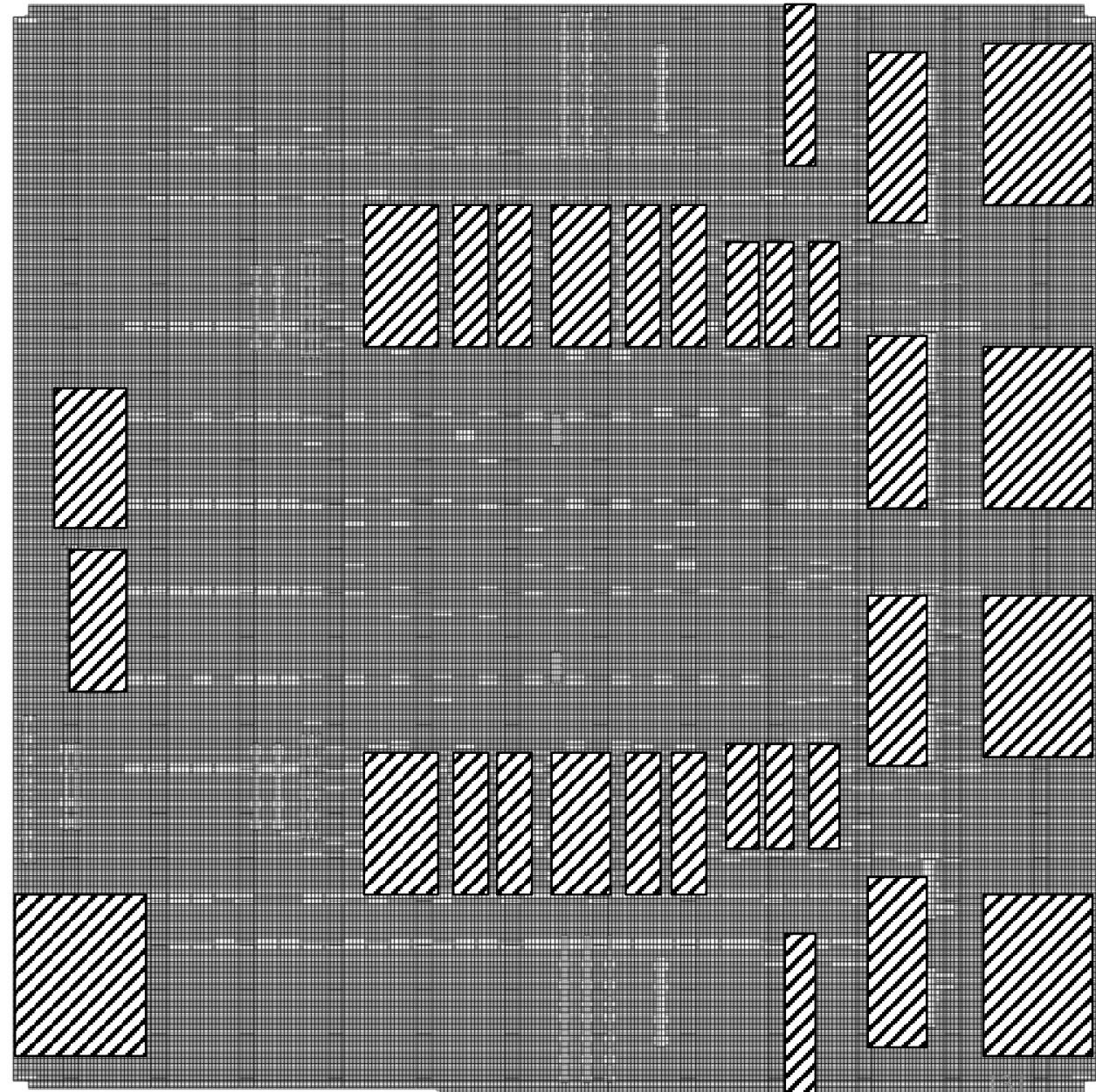
Number of regulators: **10**



# Case Two

- Placement restricted to uncongested regions

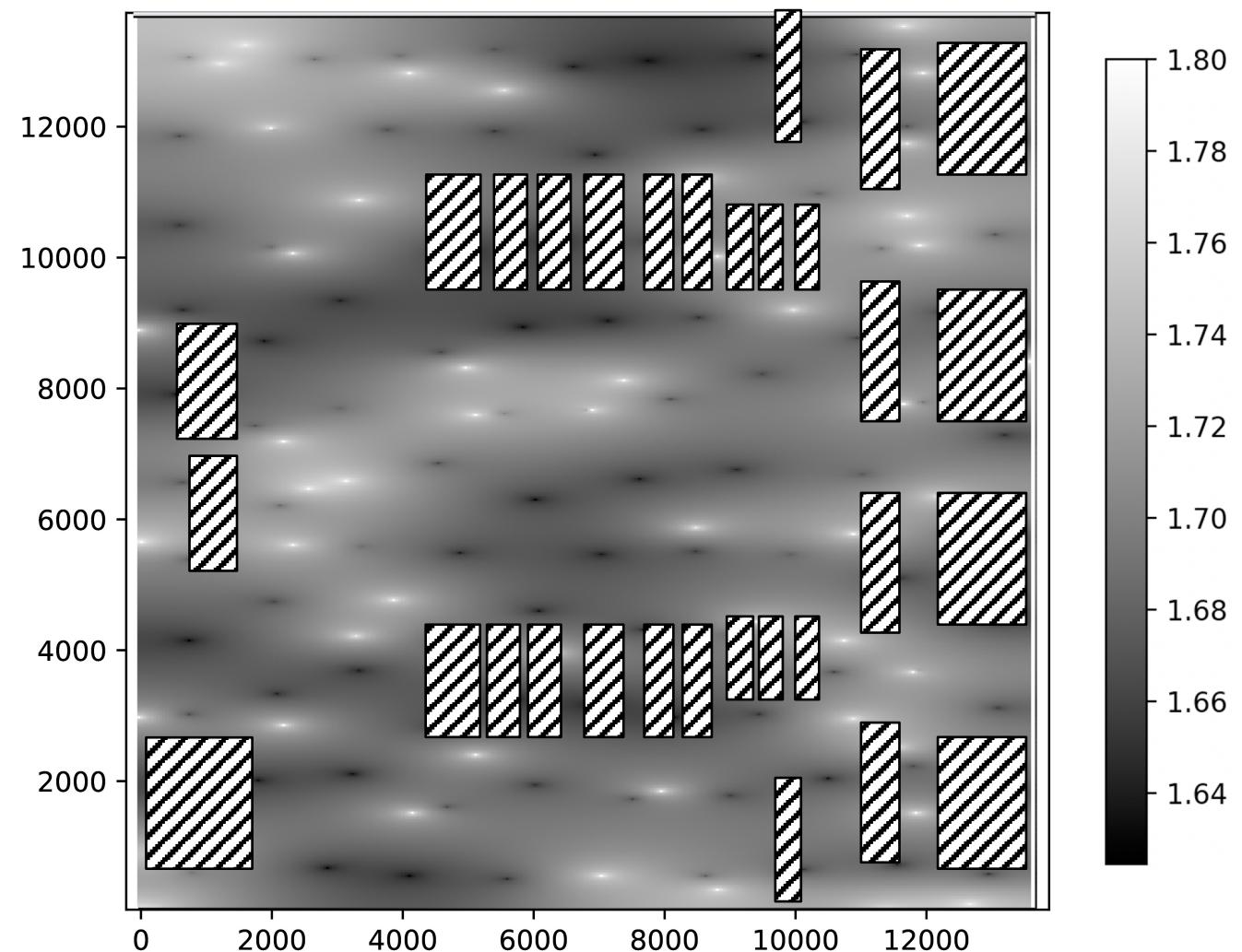
Location of blockages in ibmpg4



# Case Two

- Placement restricted to uncongested regions
- Greater average voltage drop than in case one

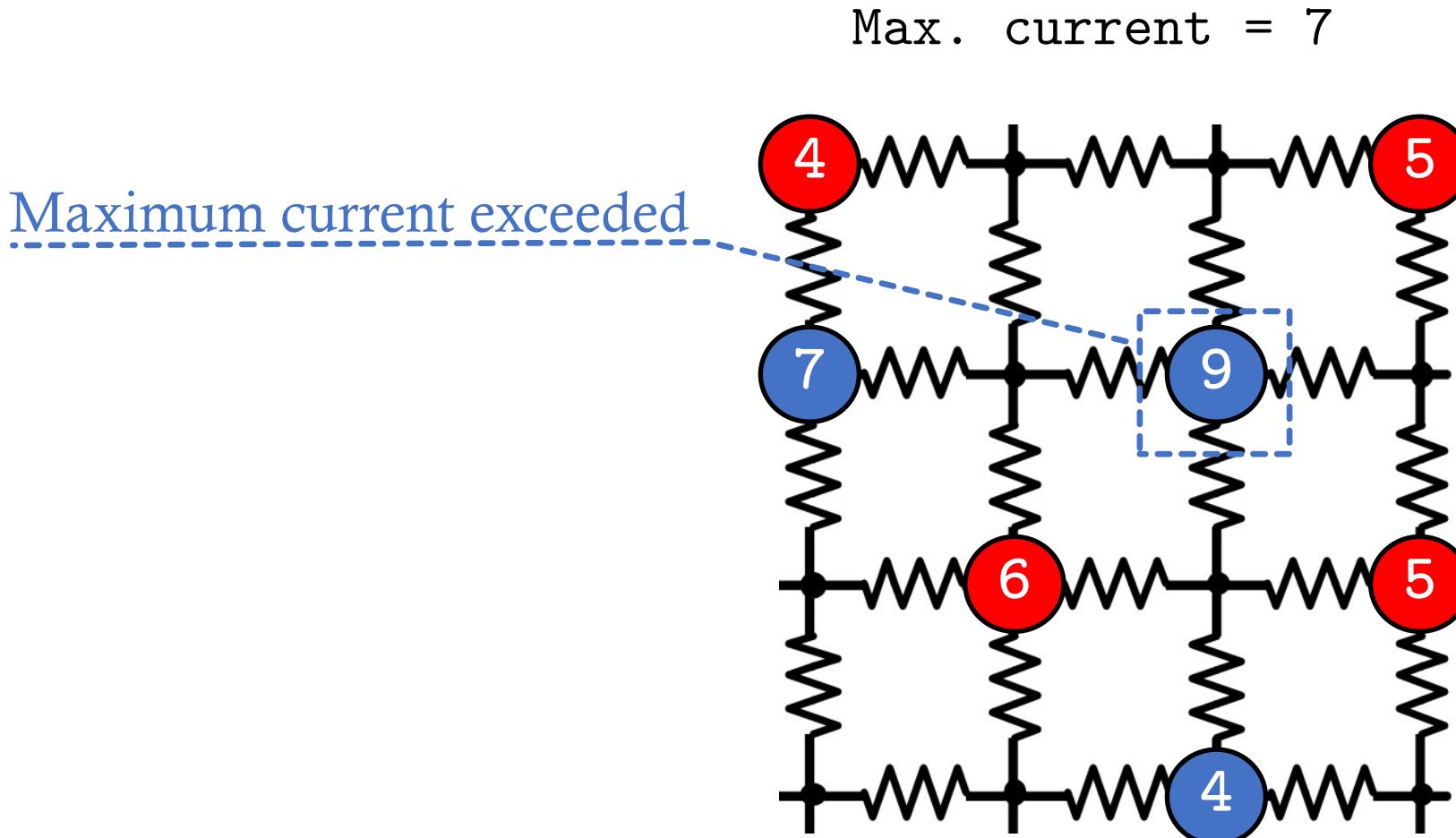
Number of regulators: 10



# Case Three

- Maximum current limited at each regulator
  - Limited area
  - Electromigration
- Prohibit exceeding maximum current
- New model needed

# Limited Current Example

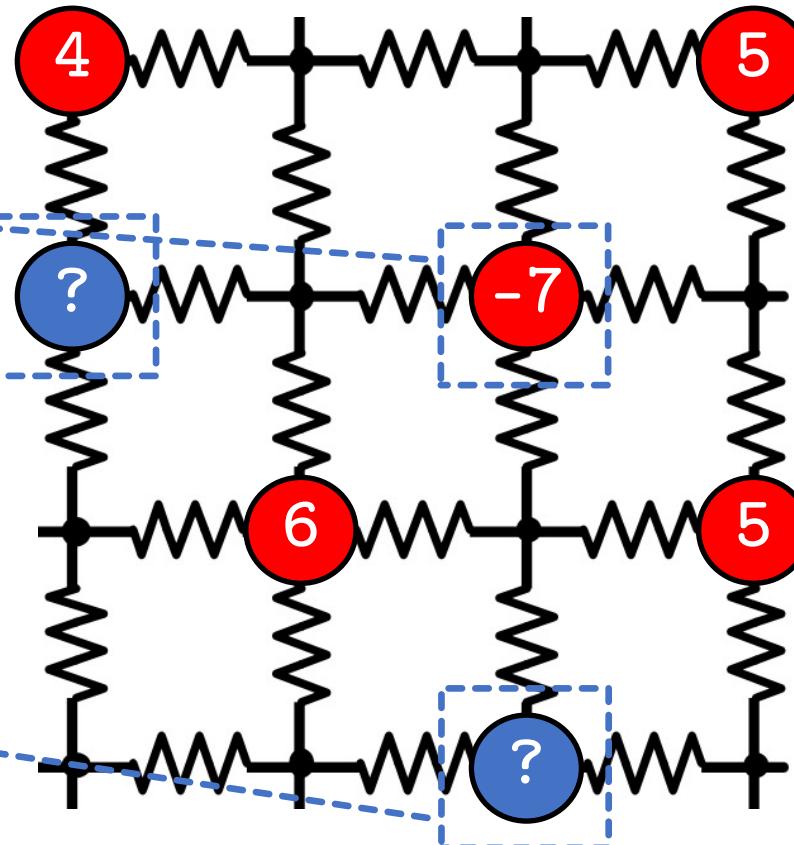


# Limited Current Example

Max. current = 7

Force max. current and  
convert to load

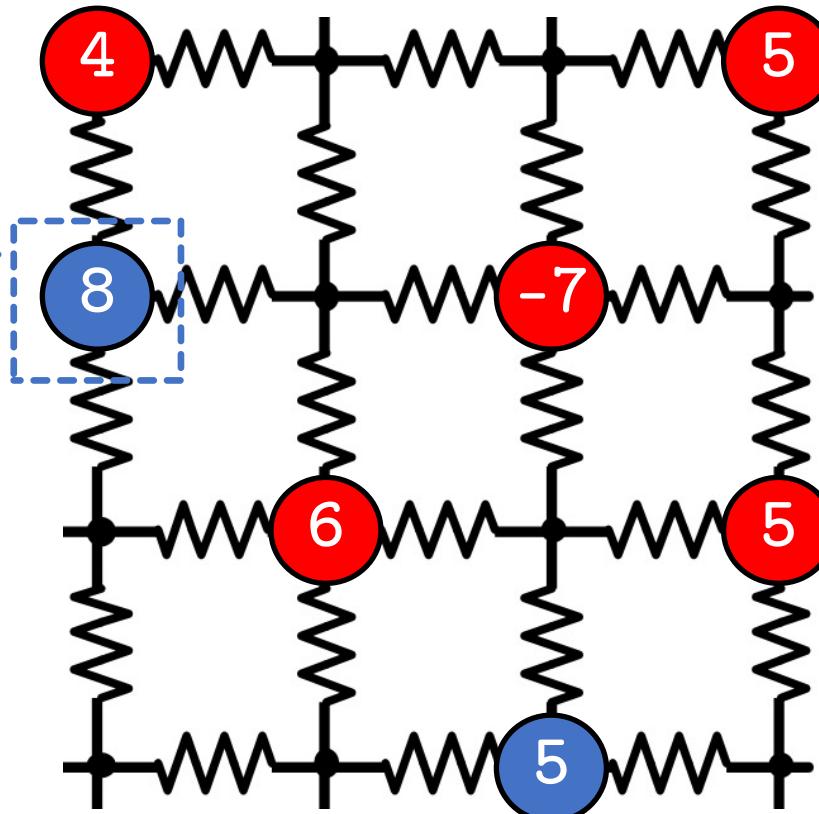
Recalculate current at  
remaining sources



# Limited Current Example

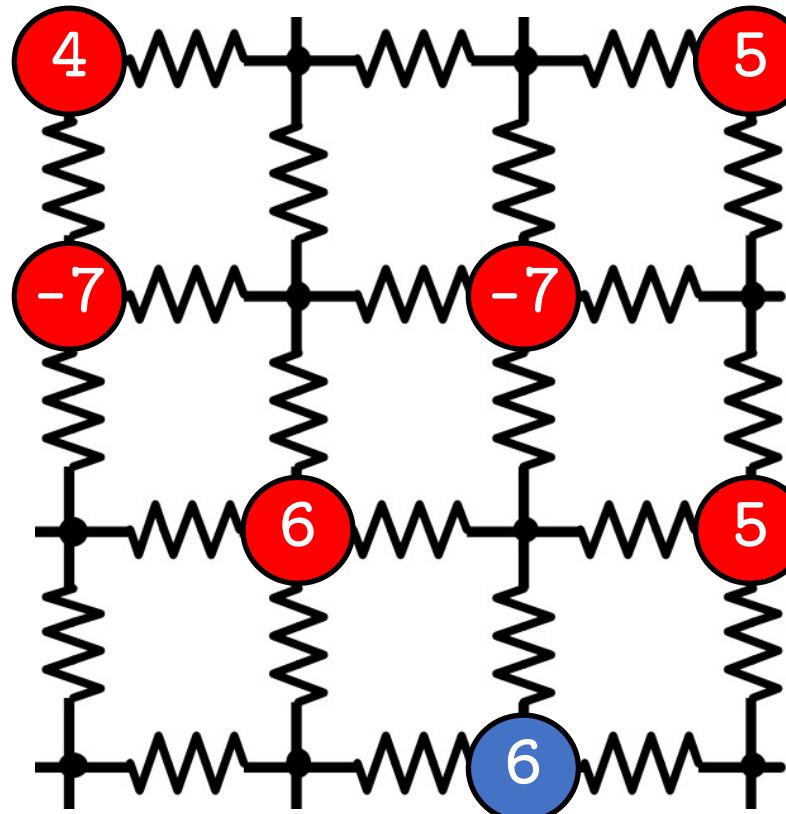
Max current exceeded.  
Repeat until convergence

Max. current = 7



# Limited Current Example

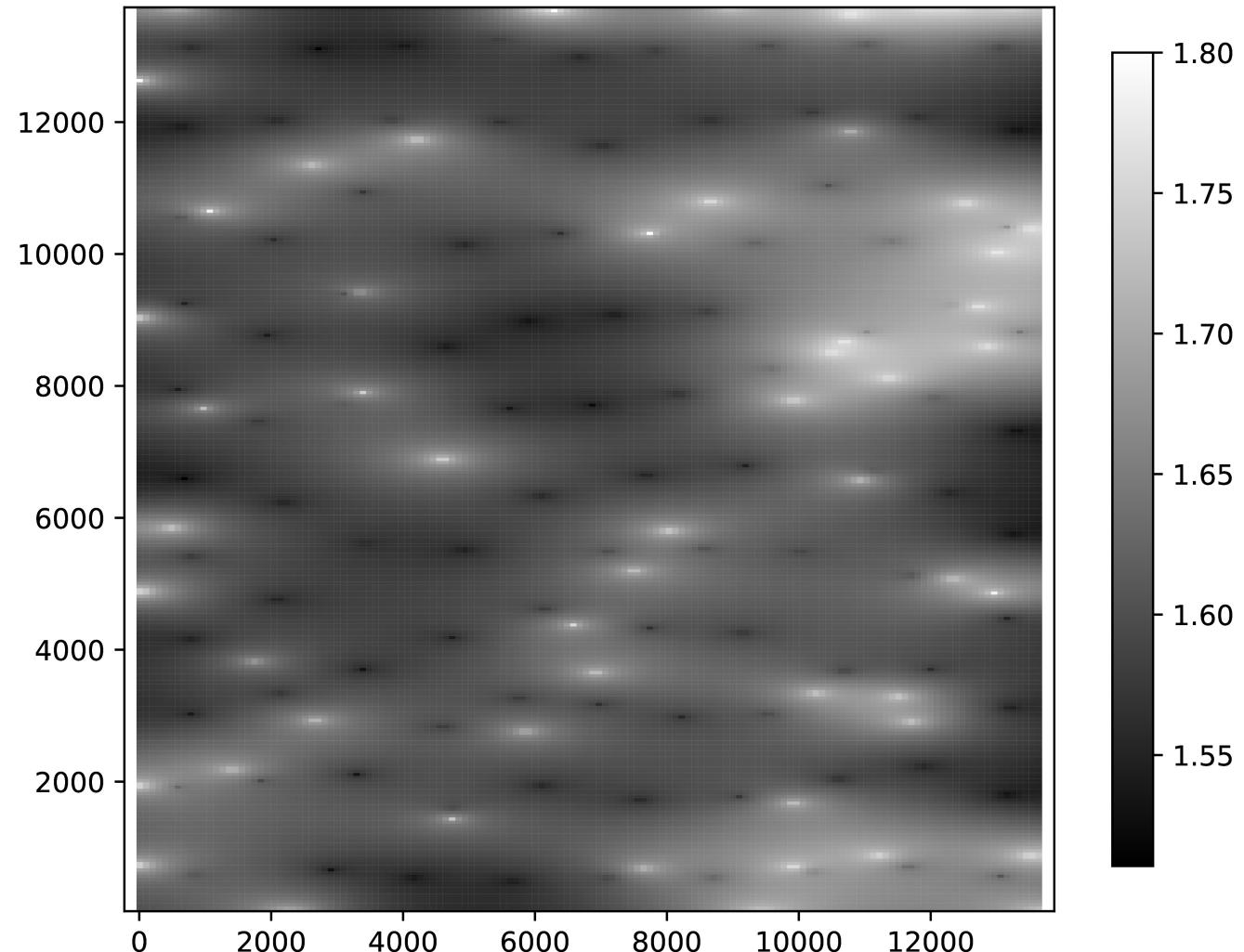
Max. current = 7



# Case Three: Results

- Greater average voltage drop than case one
- Regulators spread more evenly within layout

Number of regulators: 40



# Summary

- Distributed voltage regulation effectively reduces power noise
- Regulator placement methodology is proposed
- Practical cases are supported
  - Restricted position
  - Limited current
- Validated using industrial benchmarks

# Final Remarks

- Graph theory important component of VLSI systems design and analysis
- Graphs are however underutilized in certain areas of VLSI
  - Power delivery
  - Physical design
  - Beyond CMOS technologies
- Five works extending graph theory to underutilized areas
  - Exploratory methodology for power delivery
  - Effective resistance in truncated and finite mesh
  - SPROUT – Smart Power ROUTing tool for power network prototyping
  - QuCTS – single flux Quantum Clock Tree Synthesis
  - Placement of Distributed On-Chip Voltage Regulators

# Research Effort since 2017

## VLSI CONTINENT

