Optimal Partitioned Fault-Tolerant Bus Layout for Reducing Power in Nanometer Designs

Shanq-Jang Ruan[1], Edwin Naroska[2]
Chun-Chin Chen[1]

[1]National Taiwan University of Science and Technology
Taipei, Taiwan

[2]Fraunhofer IMS
Duisburg, Germany

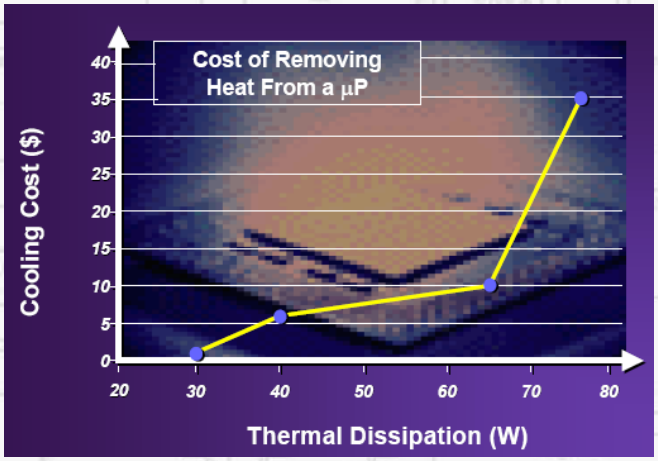
Outline

- Introduction
  - Why do we need low power
  - Source of Power Dissipation
  - Coupling effect in deep-submicro technology
  - Motivation and Observation
- Permutation and Spacing for Low Power Bus Design
  - Coupling Capacitance Model
  - Permutation Technique
  - Spacing Technique
- Bus Partition Flow
- Low Power Fault-Tolerant Bus Architecture
- Graph Theory Algorithm for Fault-Tolerant Bus Layout Optimization
- Experimental Results
- Conclusion

Introduction
-Why do we need low power-

- **Why we need low power fault-tolerant bus design in VDSM**
  - Power has become a critical concern for nowadays design for hand-held and high-performance systems.
  - In VDSM era, on-chip interconnect contribution about 15% to 30% total power dissipation in modern microprocessors.
  - For low power and reliability concerns, a fault-tolerant bus could be used to reducing power while increasing reliability.

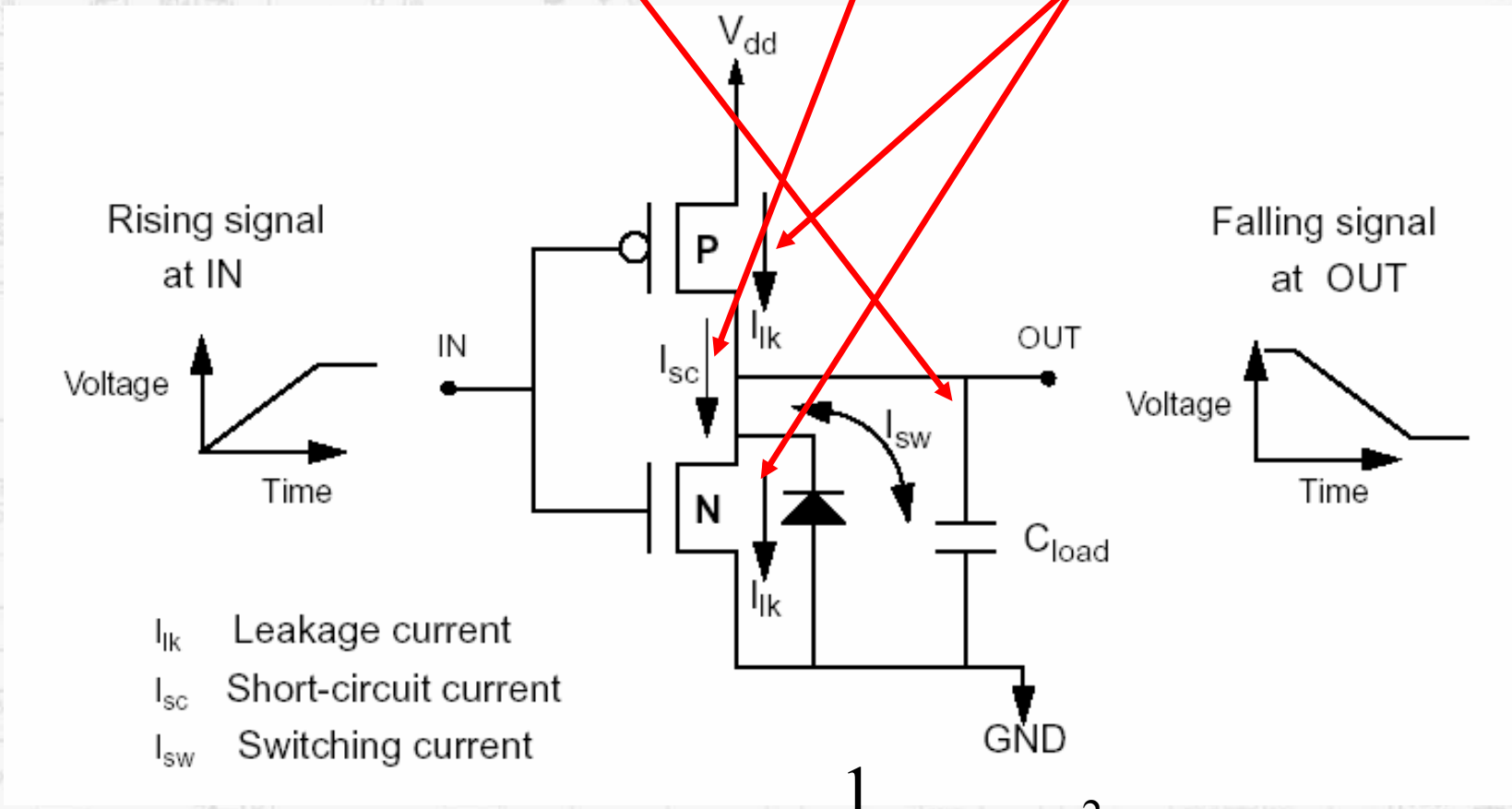
Battery Life



Reliability

Introduction
-Source of Power Dissipation-

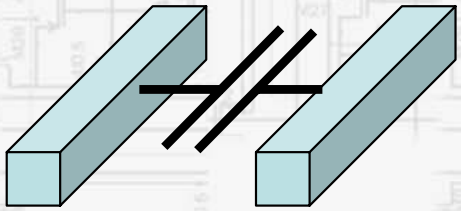
$$P_{avg} = P_{dynamic} + P_{short} + P_{leakage}$$



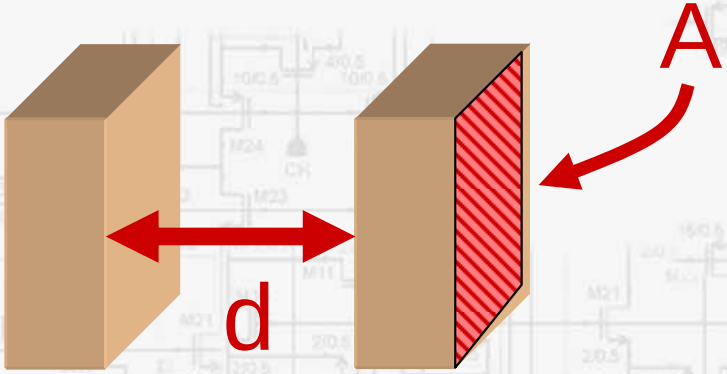
$$P_{dynamic} = \frac{1}{2} C_L \times V_{dd}^2 \times E(sw)$$

Introduction
-Coupling Effect in DSM Technology-

- Geometry downscaling of digital circuits (deep submicron) increases coupling capacitance C between adjacent wires



$$C = \varepsilon \frac{A}{d}$$



This effect also introduce power dissipation

Coupling power becomes significant in deep-submicro. For example, in standard 0.13μm technologies with minimum distance between bus lines, the ratio of coupling and line capacitances is great than eight.

Introduction
-Motivation and Observation-

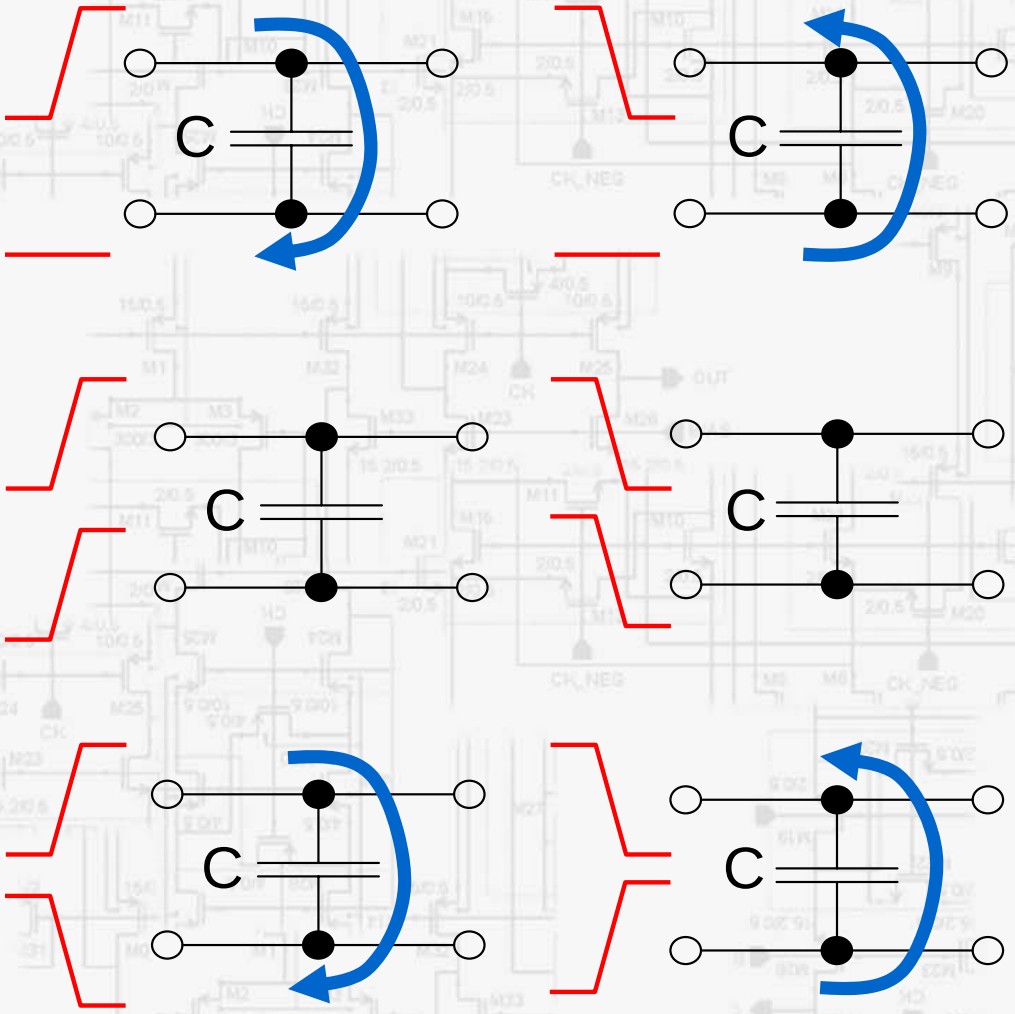
- From 12 SPEC2000 benchmark programs, the number of different transition patterns is far less than theoretical number ($2^{32} \times 2^{32}$).

Prog.	gzip	vpr	gcc	mcf	parser	perlbmk	vorex	bzip2	mesa	art	equake	ammp
Tran. Types	5914	15546	85728	6011	12497	33408	30714	5778	13547	5302	7655	9788

- Signal statistics can be used to decrease coupling!

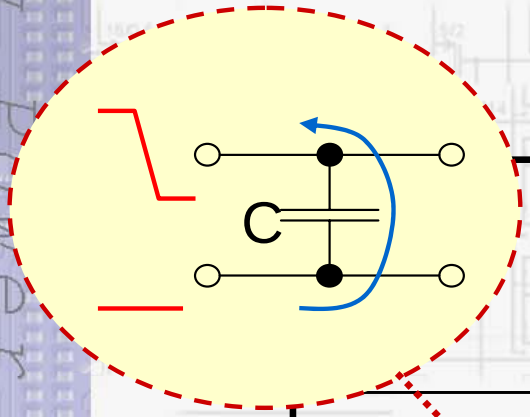
Modeling Coupling Effect for Adjacent Wires

- One wire is static
  - Dissipated power: λ
- Both wires transition to same value
  - Dissipated power: 0
- Wires transition in opposite directions
  - Dissipated power: 4λ



Coupling Effect Table

- Normalized coupling effect table



		new value			
		00	01	10	11
Old value	00	0	1	1	0
	01	1	0	4	1
	10	1	4	0	1
	11	0	1	1	0

Problem Formulation

- Based on probabilities $p_{01,00}(i, j)$, $p_{10,00}(i, j)$, $p_{11,00}(i, j)$, $p_{00,01}(i, j)$, $p_{10,01}(i, j)$, …, $p_{01,11}(i, j)$ the crosstalk caused by a wire pair (i, j) can be formulated as:

$$\begin{aligned} xtalk(i,j) = {} & p_{00,01}(i,j) + p_{10,11}(i,j) + \\ & p_{01,00}(i,j) + p_{11,10}(i,j) + \\ & p_{00,10}(i,j) + p_{01,11}(i,j) + \\ & p_{10,00}(i,j) + p_{11,01}(i,j) + \\ & 4p_{01,10}(i,j) + 4p_{10,01}(i,j) \end{aligned}$$

- Coupling Effect between wires:

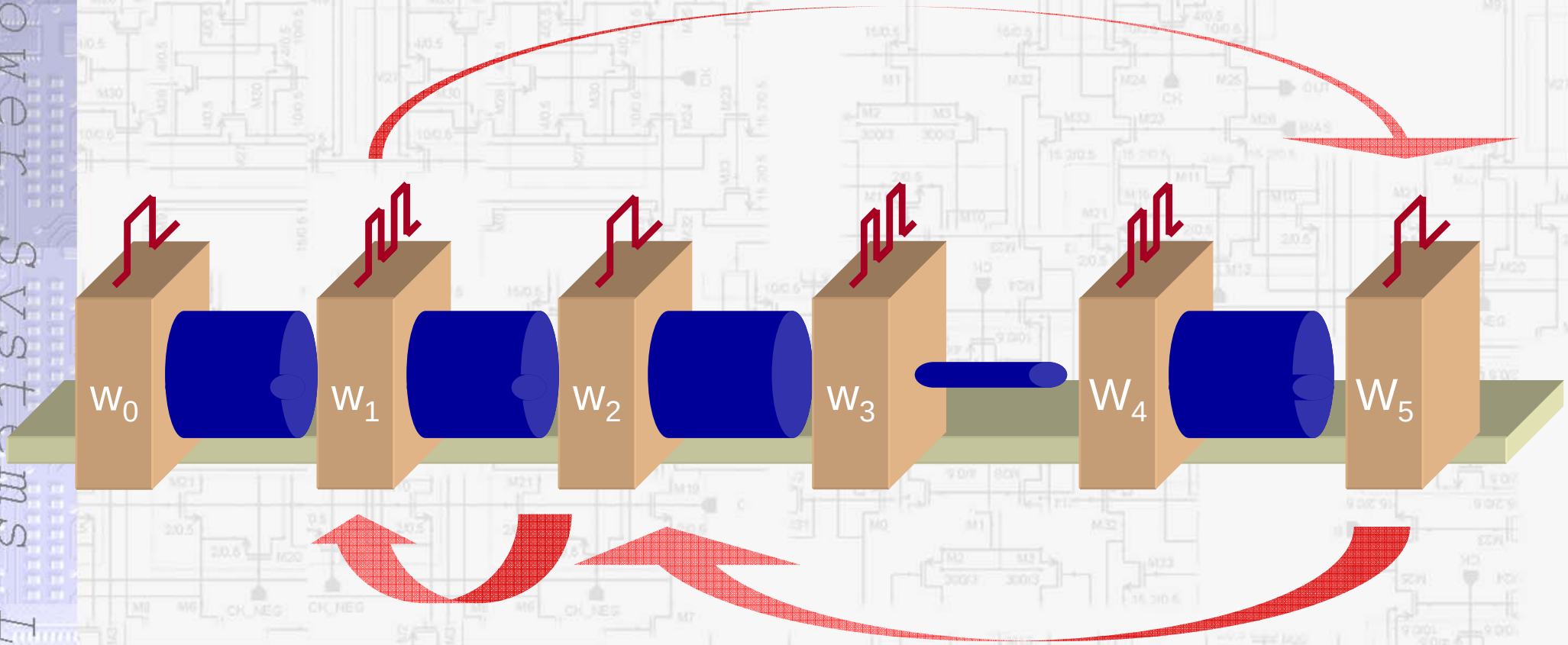
$$xtalk_{eff} = \sum_{i=0}^{n} c_i$$

with

$$c_i = \varepsilon \frac{A}{d_{\min}} \cdot xtalk(w_i, w_{i+1})$$

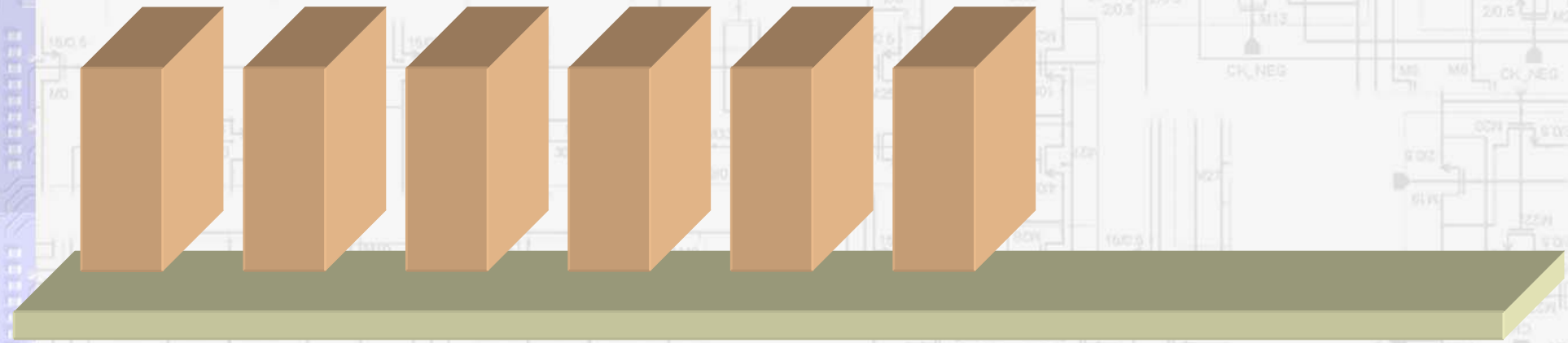
Permutation

- Coupling effect (power dissipation) is dependent on signal statistics of neighboring wires



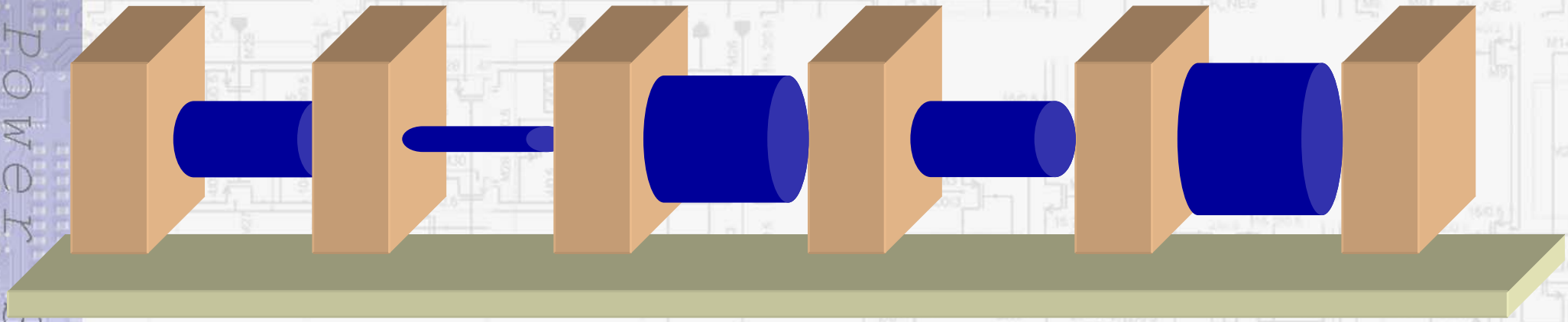
Spacing approach #1/2

- Spacing approaches
  - Equal spacing
  - Coupling aware spacing
- "Equal" spacing

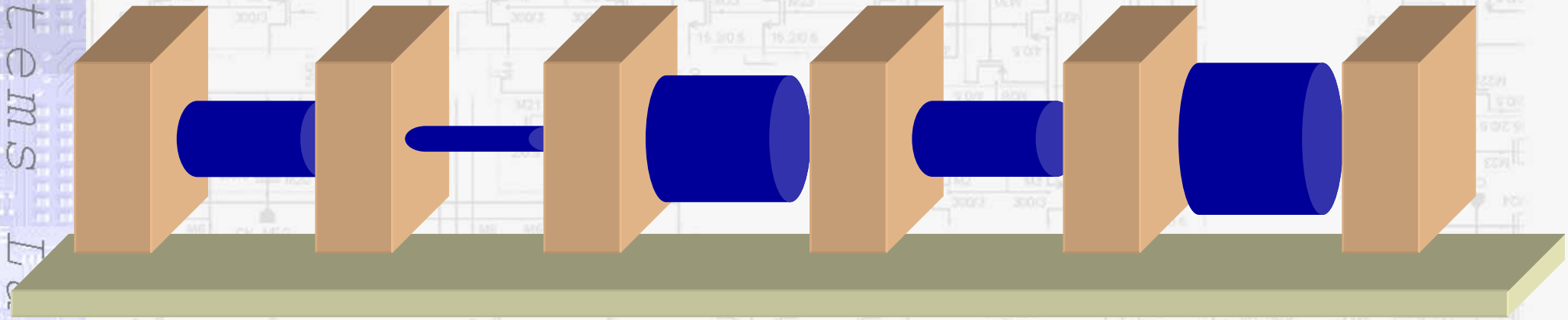


Spacing approach #2/2

- Coupling between adjacent wires is typically different due to traffic



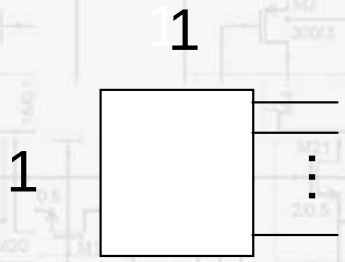
- "Coupling aware" spacing

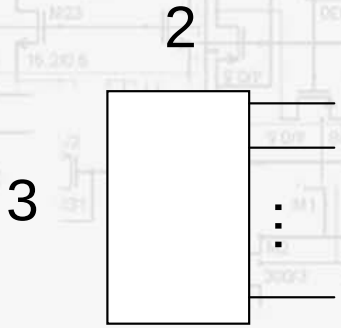


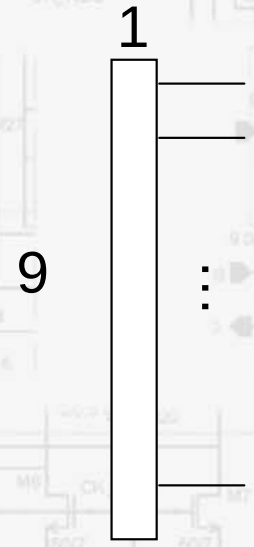
Do We Need Extra Area for Enlarging Wire Spacing?

- Example:
  - All digital phase locking loop (ADPLL)
  - Area utilization is set to 82% for routing with Astro (Synopsys) by different width:length ratios.

Ratio	Chip-Width(μm)	Chip-Length(μm)	Chip-Area (μm)
1:1	549.78	546.32	300355
2:3	445.66	667.28	297380
1:9	181.62	1640.00	297856

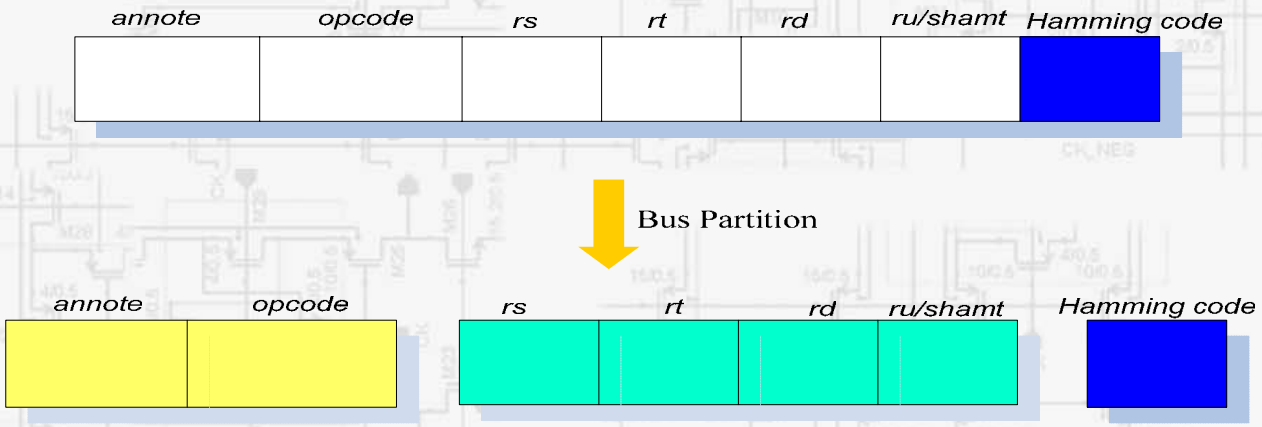




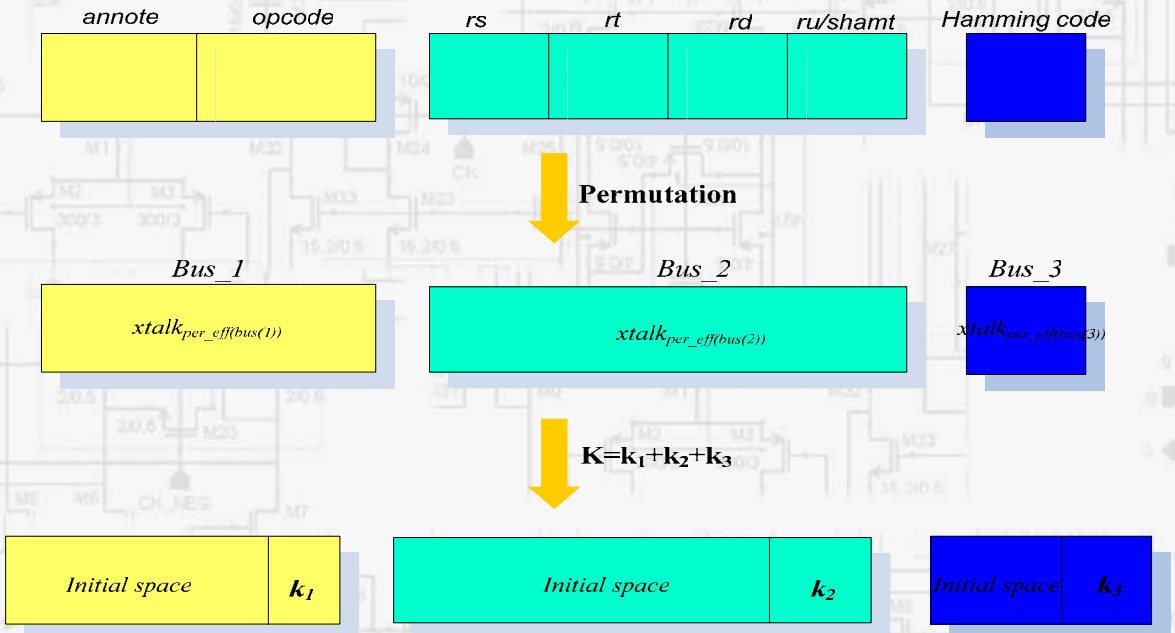


Bus Partition Flow

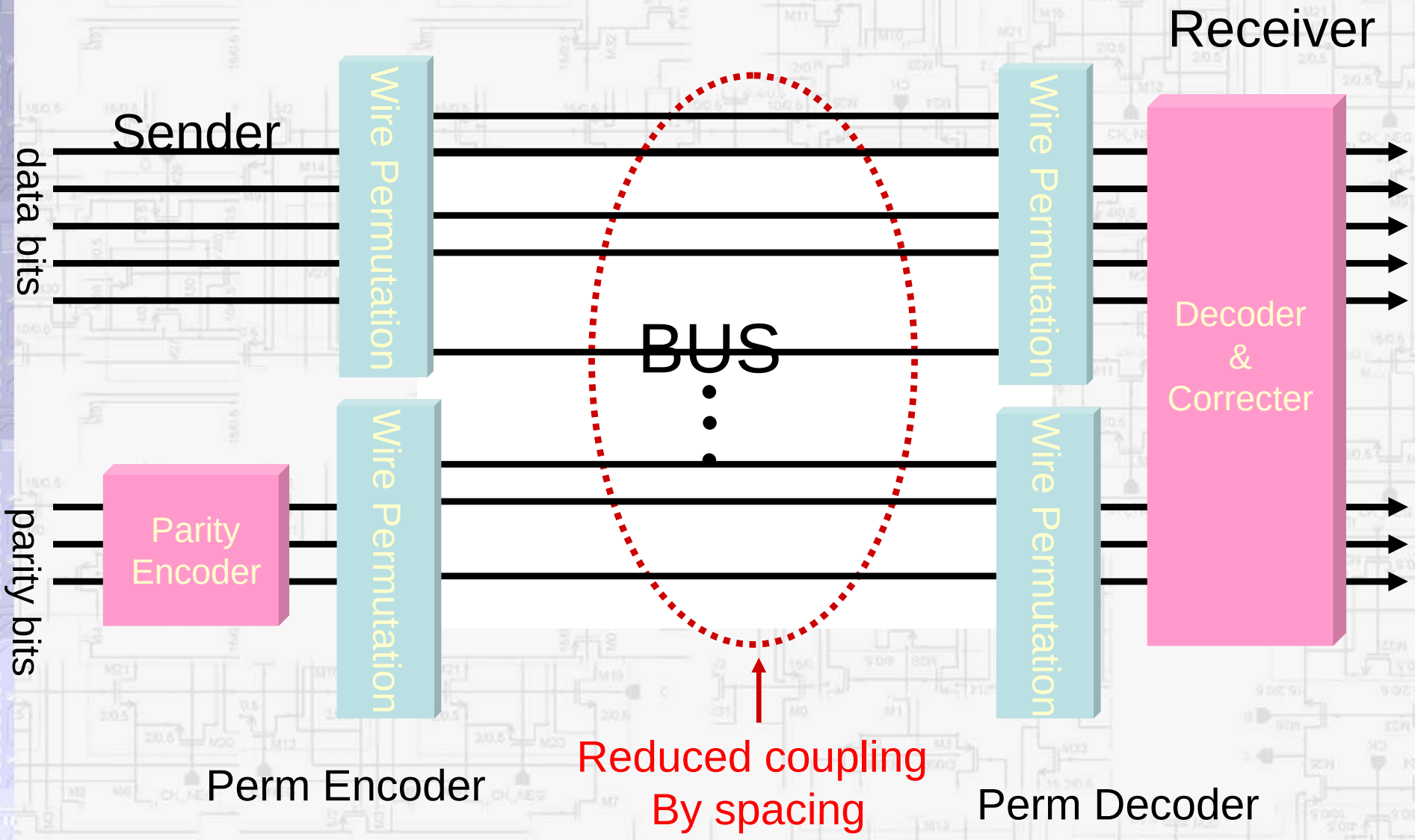
- Bus Partition



- Optimization with Permutation and Spacing



Partitioned fault-tolerant bus Architecture



Graph Theory Algorithm for Fault-tolerant Bus Layout Optimization

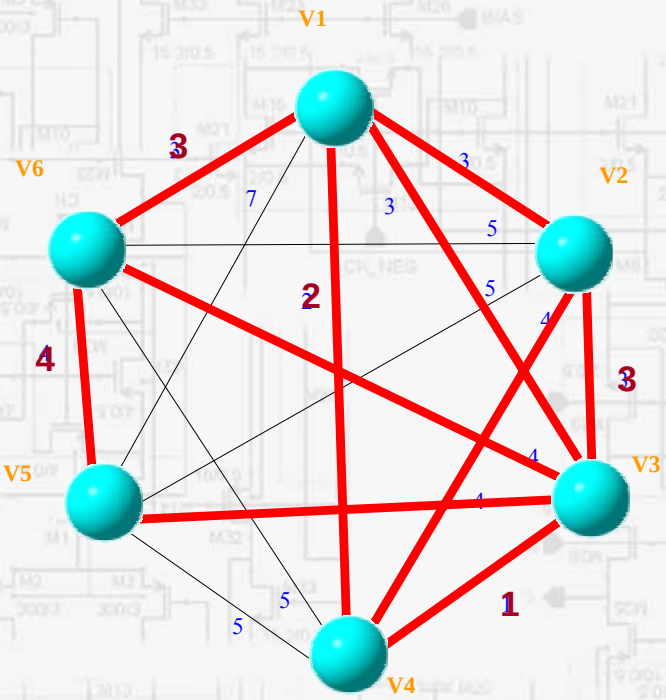
- Graph theory for permutation
  - This problem is similar to the traveling-salesman problem or finding minimum Hamilton path problem .
  - To find a optimization solution in a polynomial-time
- Graph theory for spacing
  - Depend on the result of permutation
  - Got more coupling in neighbor wires, got more extra space in neighbor wires of bus.

Graph theory for permutation

- Using a heuristic algorithm similar to Kruskal's minimum spanning tree algorithm.
  - the edge with the smallest weight is selected that does not cause a cycle and does not increase the degree of a node to more than two

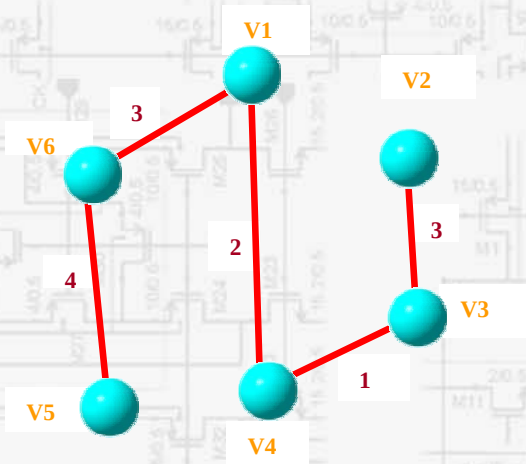
$$M = \begin{bmatrix} 0 & 3 & 3 & 2 & 7 & 3 \\ 3 & 0 & 3 & 4 & 5 & 5 \\ 3 & 3 & 0 & 1 & 4 & 4 \\ 2 & 4 & 1 & 0 & 5 & 5 \\ 7 & 5 & 4 & 5 & 0 & 4 \\ 3 & 5 & 4 & 5 & 4 & 0 \end{bmatrix}$$

Transfer to graph and find the minimum hamiltonian path

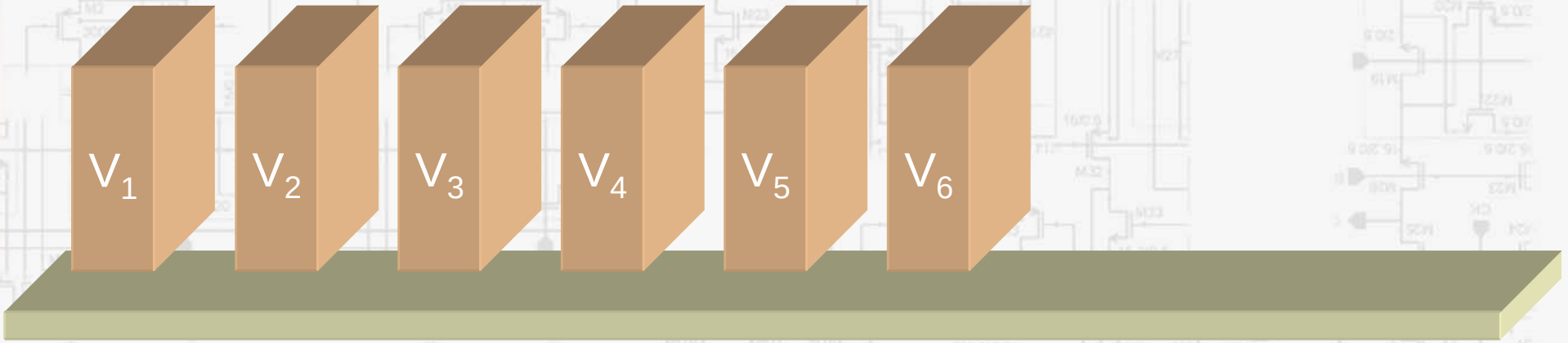


Graph theory for spacing

- Using the result of permutation and extra space K to assign space

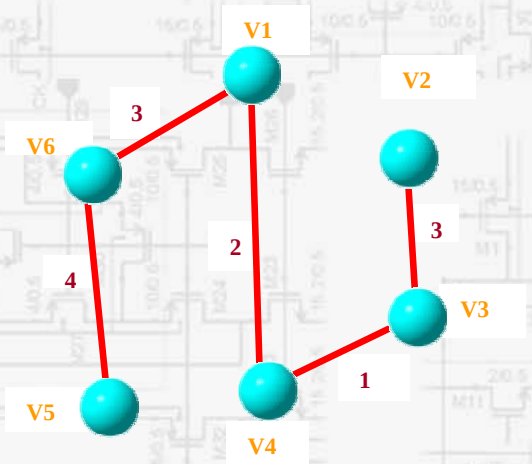


$$d_i = \frac{K \times w_i}{w_1 + w_2 + \cdots + w_{n-1}}$$

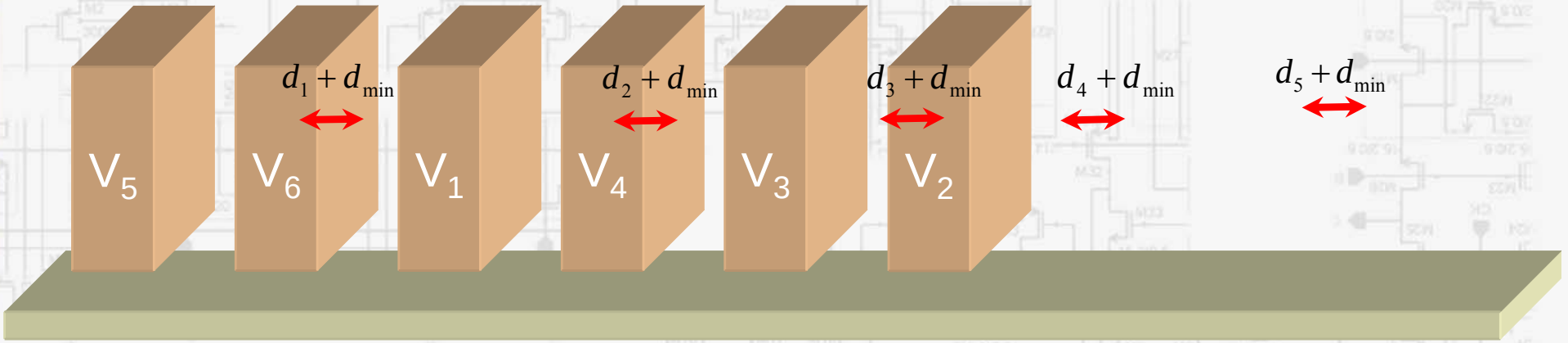


Graph theory for spacing

- Using the result of permutation and extra space K to assign space

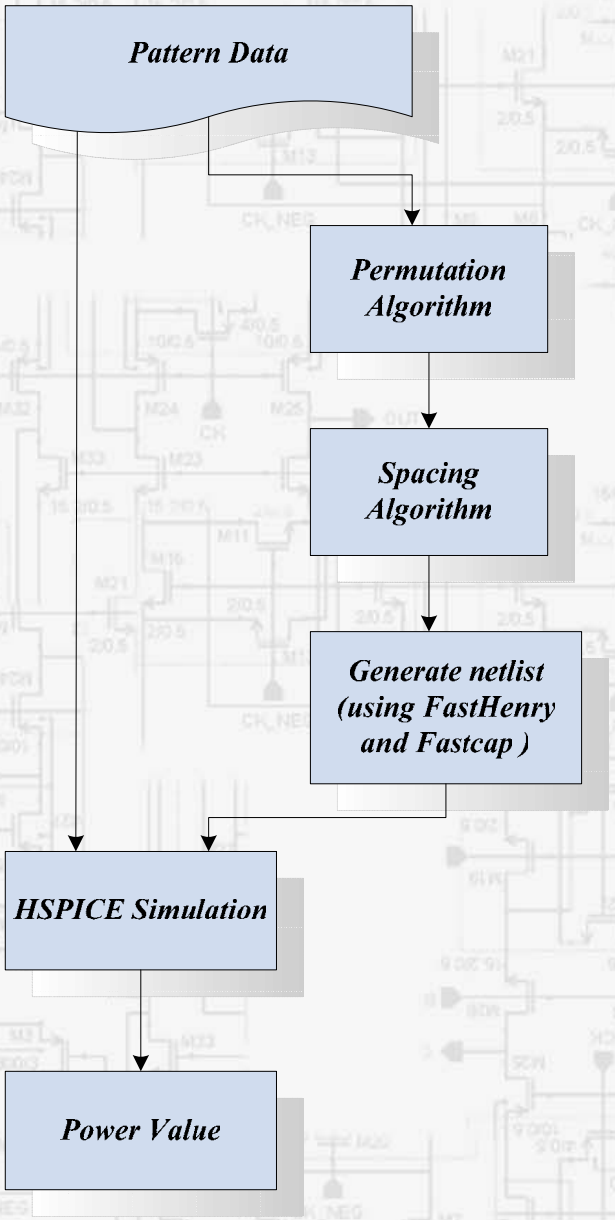


$$d_i = \frac{K \times w_i}{w_1 + w_2 + \cdots + w_{n-1}}$$



Simulation Environment

- Instruction bus of PISA processor architecture (derived from MIPS-IV ISA)
- 12 SPEC2000 benchmark programs
- 20% additional space
- The bus is with 1000μm long, 0.14μm width and 0.25μm height based on 90nm technology process



Power Reduction Compared with the Original bus Architecture

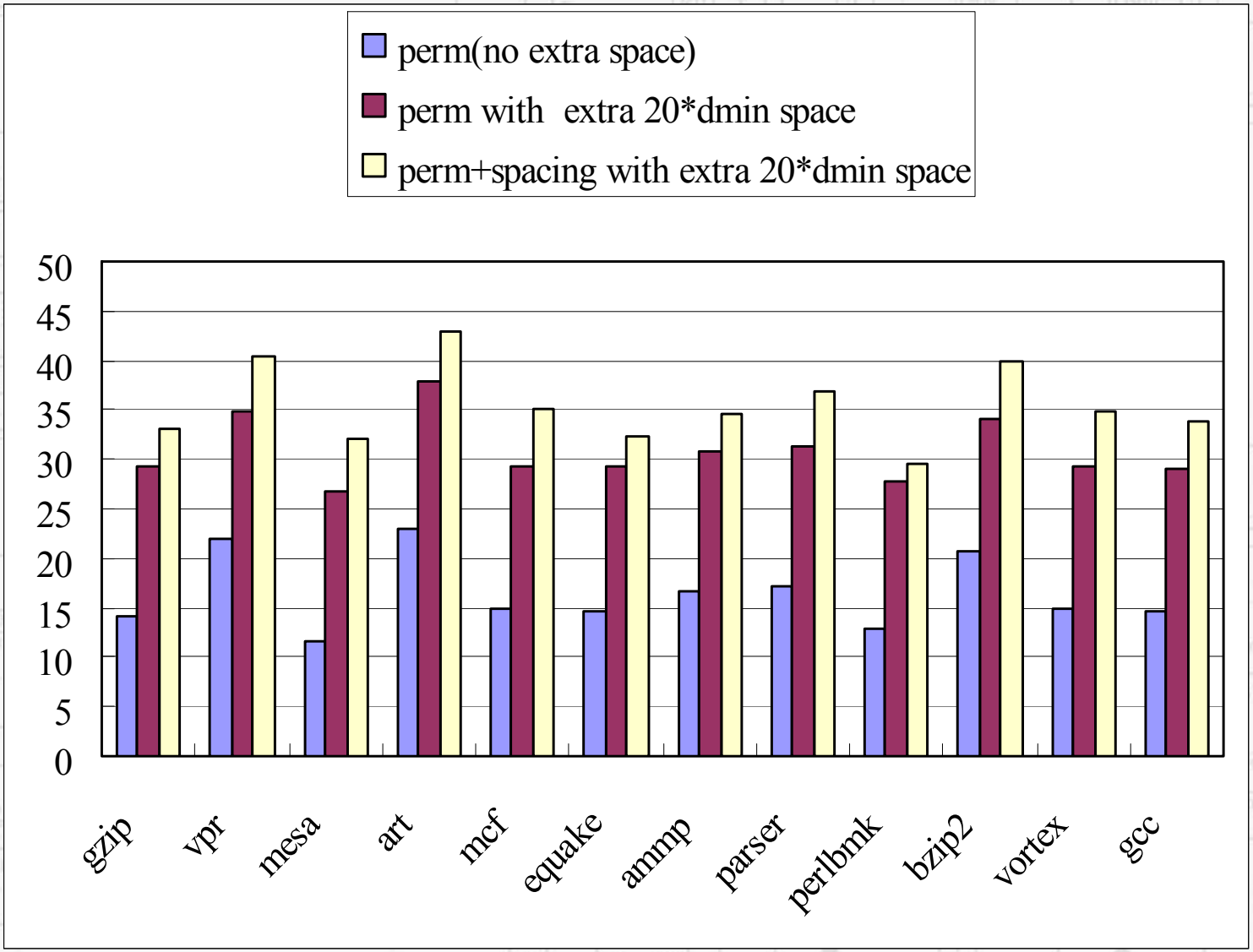


Fig. 9

Power Reduction for Different Parity Bit Positions and Spacing Approaches

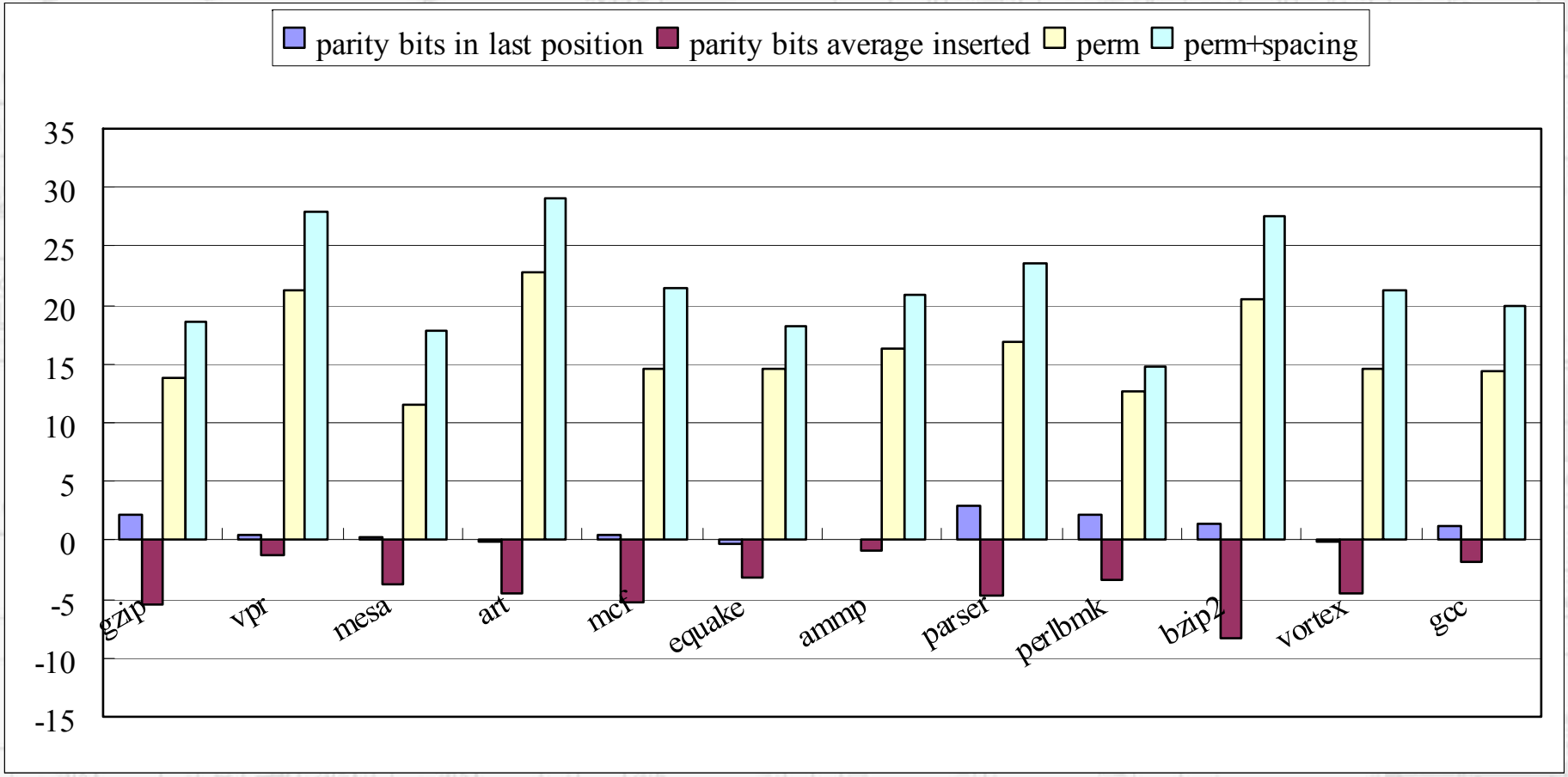
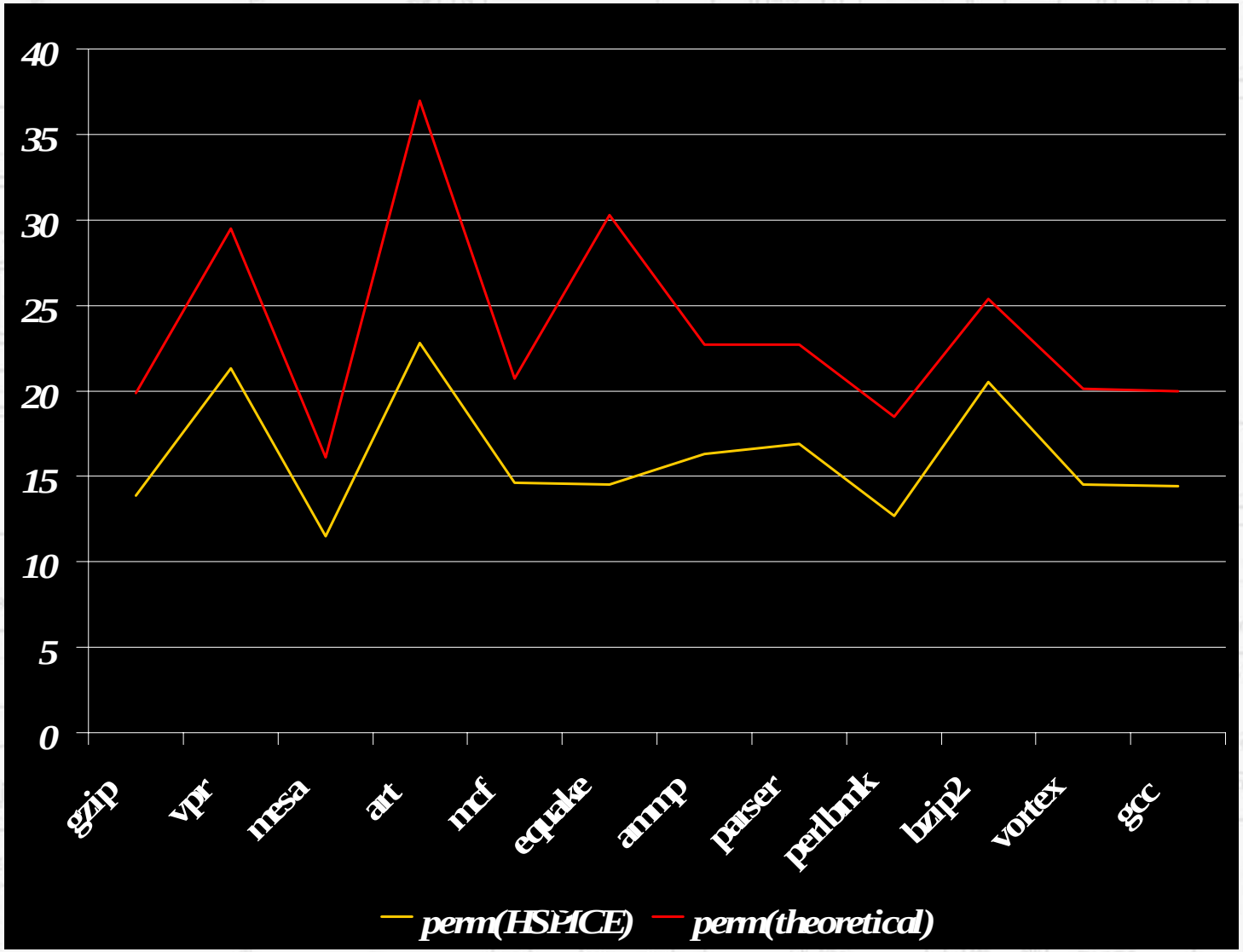


Fig. 10

Theoretical Optimization Results vs. Realistic Values



Coupling Effect Reduction for Different Partitioned Methods

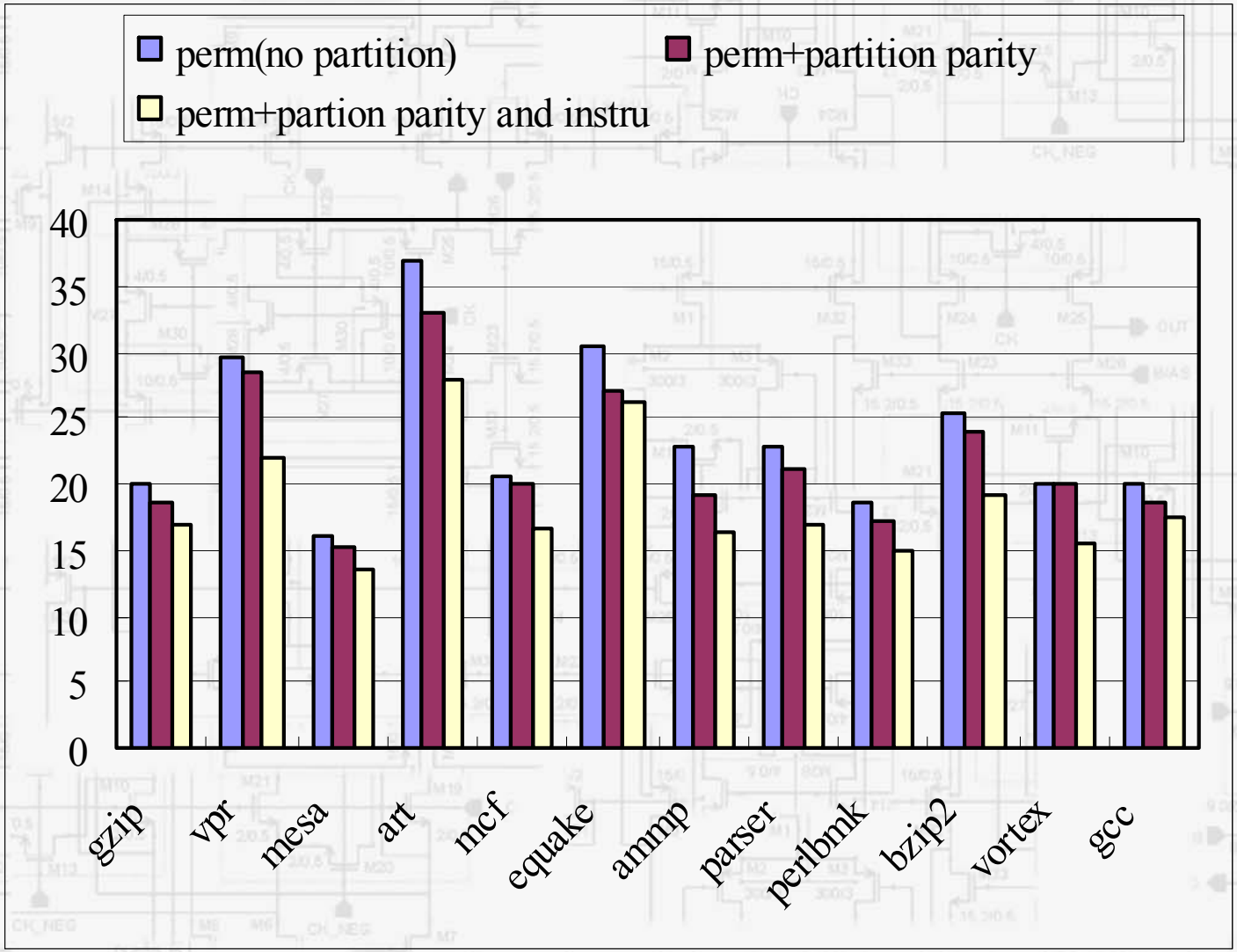


Fig. 12

Power Reduction of Partitioned Fault-tolerant Bus

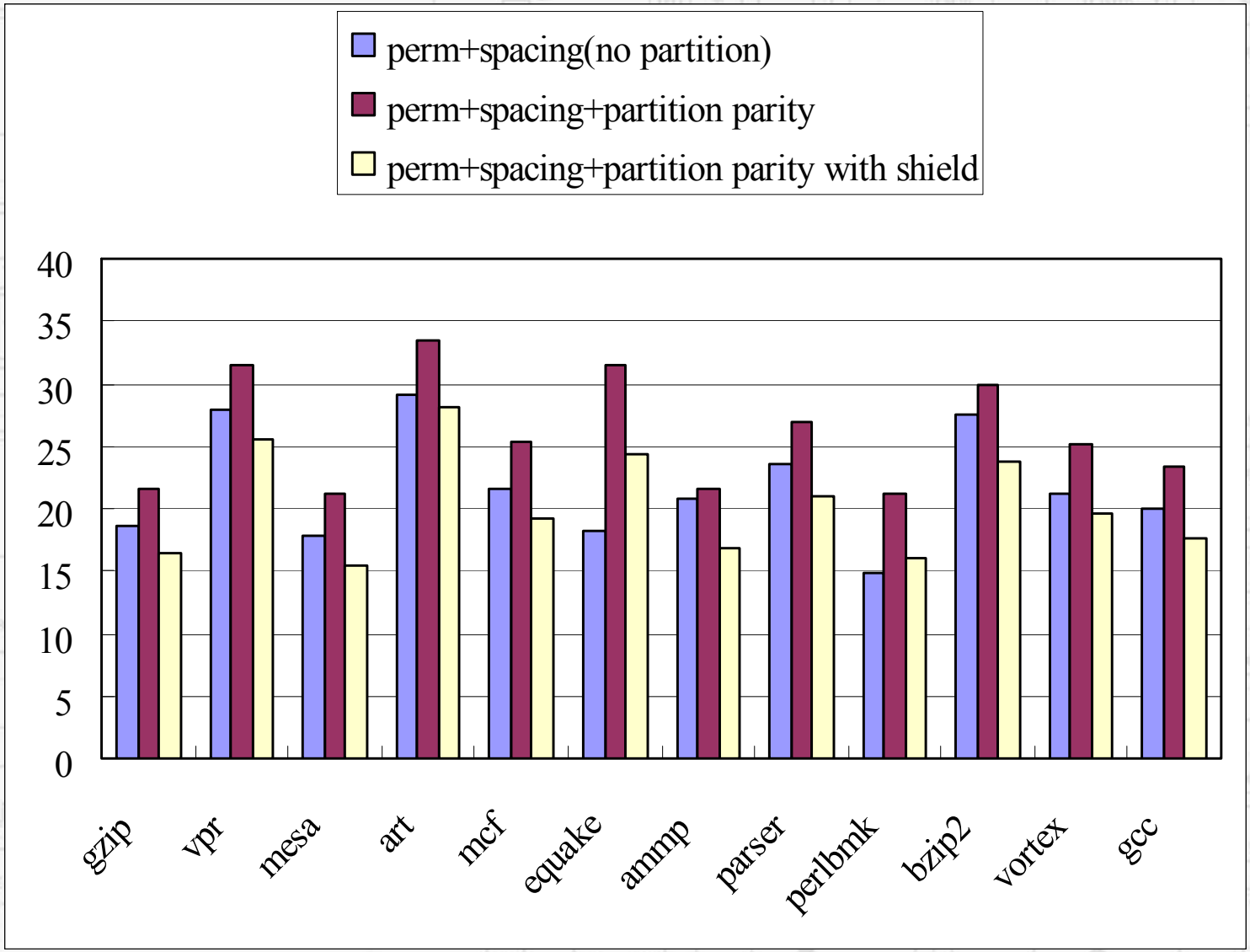


Fig. 13

Conclusion

- Our fault-tolerant bus architecture using graph theory algorithm can efficient reduce coupling compared to realistic experimental result.
- Nearly no additional area and timing delay
- Finding a optimization solution in a polynomial-time
- Partition fault-tolerant bus architecture using graph theory algorithm also can efficient save energy.
- Easy to merge into nowadays' design flow

Thanks for Your Attention!