Control Synthesis and Delay Sensor Deployment for Efficient ASV designs

CHAOFAN LI <CHAOF@TAMU.EDU>, TEXAS A&M UNIVERSITY

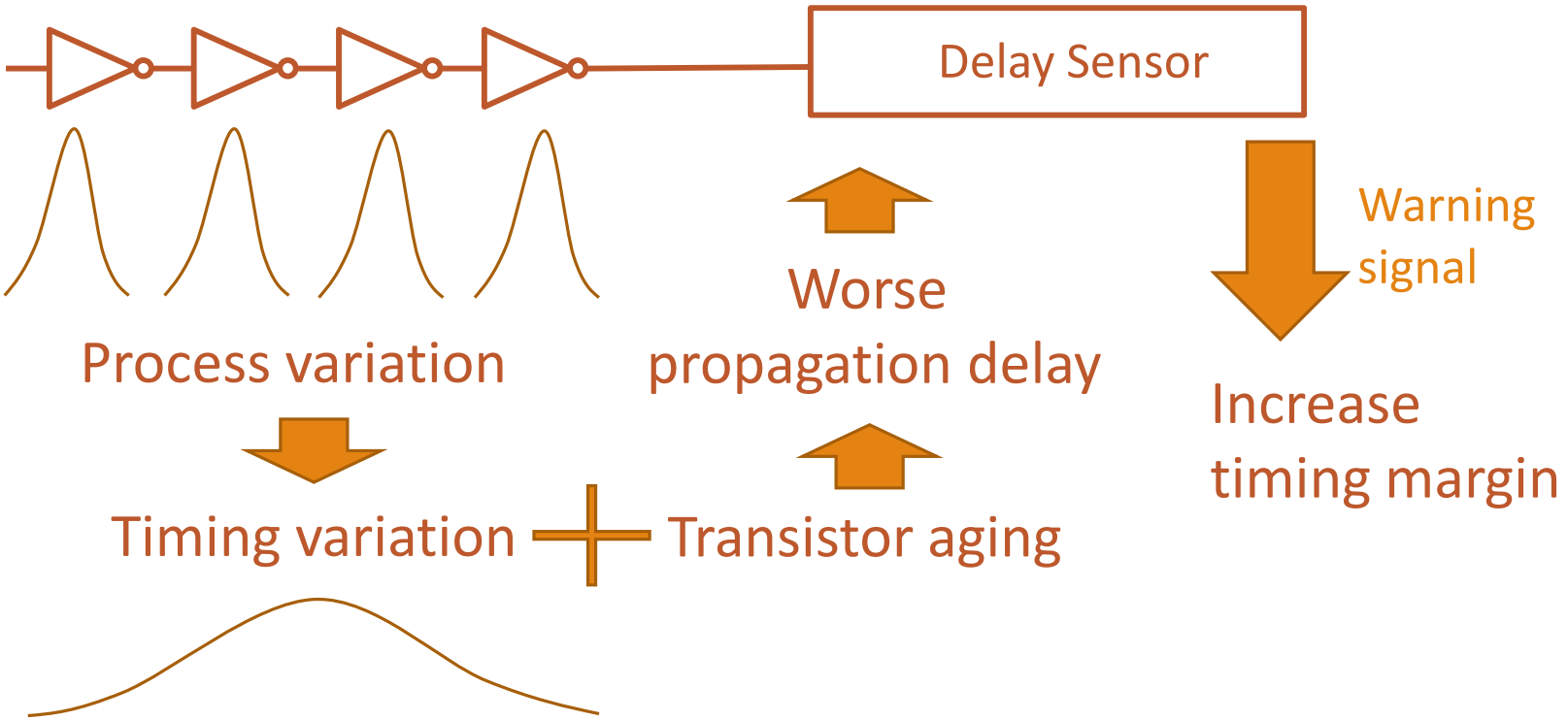
SACHIN S. SAPATNEKAR, UNIVERSITY OF MINNESOTA

JIANG HU, TEXAS A&M UNIVERSITY

This work is partially supported by NSF and Semiconductor Research Corporation (SRC)

Motivation

Resilience against Circuit Variation

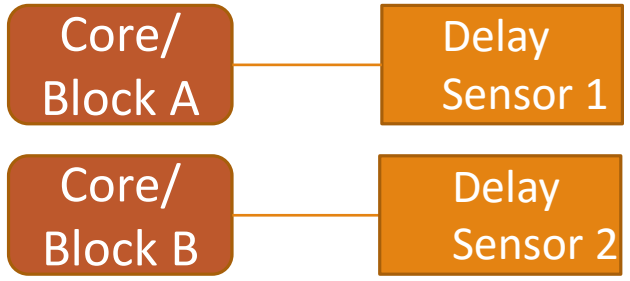


Introduction
Adaptive Design for Variation Tolerance

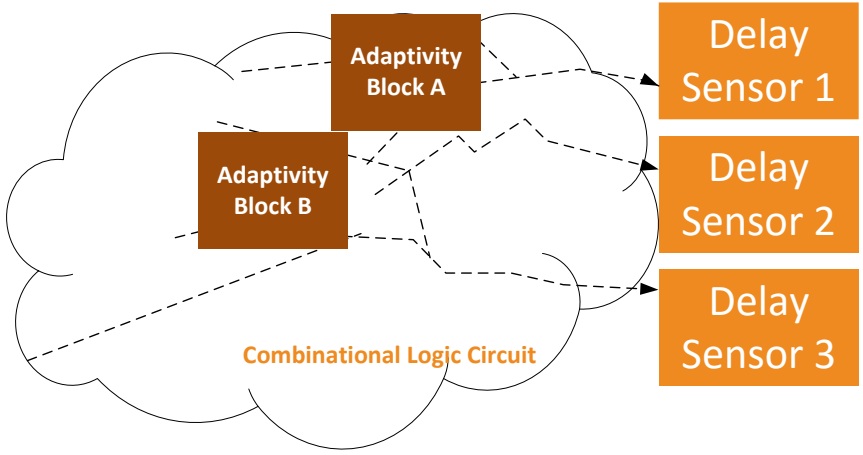
- Delay sensor designs
  - Critical Path Replica
  - Canary Flip-Flop
    - Razor-like shadow flip-flop
- Resilience for variation
  - Adaptive Supply Voltage (ASV)
  - Adaptive Body Bias (ABB)
    - Forward Body Bias (FBB)
    - Reverse Body Bias (RBB)

Introduction
Adaptive Supply Voltage designs

Coarse-grained ASV	Fine-grained ASV
Less adaptive blocks and delay sensors	More adaptive blocks and delay sensors
One-to-one association of sensors and blocks	One sensor may reflect multiple blocks
Simple control	Complex control



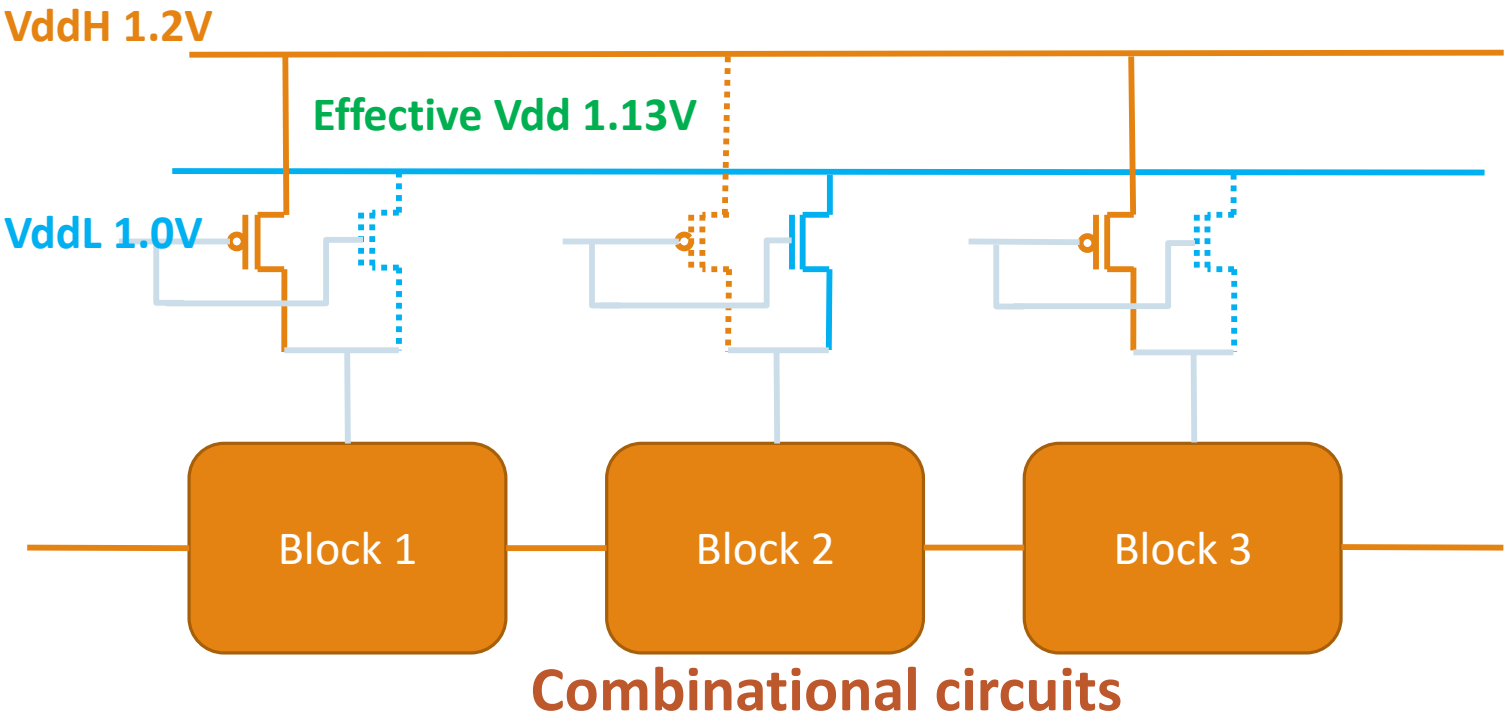
Coarse-grained ASV



Fine-grained ASV

Introduction

Voltage Interpolation[1]

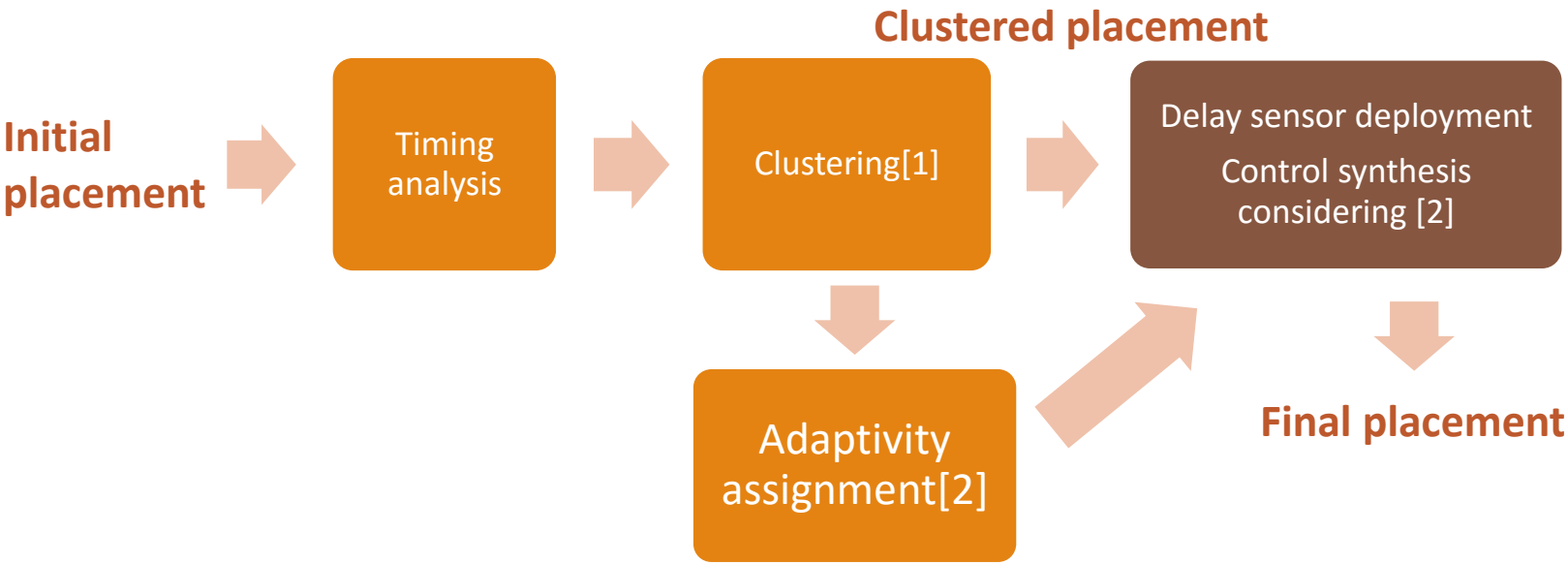


Voltage Interpolation is one of the fine-grained ASV designs

- **Only two voltage regulators**

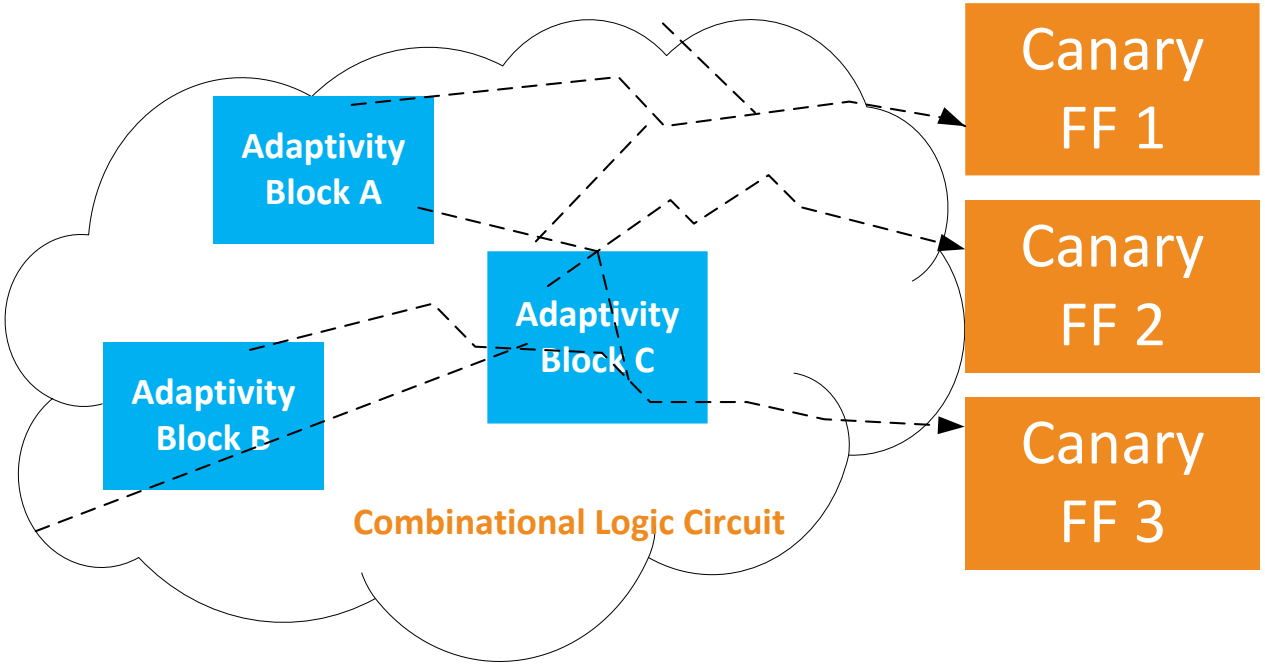
[1] Liang. X. etc. "Revival: A variation-tolerant architecture using voltage interpolation and variable latency". *2008*

Overview of Design Flow



[1] A. Lu, H. He and J. Hu, "Proximity Optimization for Adaptive Circuit Design," ACM International Symposium on Physical Design, 2016.
[2] H. He, J. Wang and J. Hu, "Collaborative Gate Implementation Selection and Adaptivity Assignment for Robust Combinational Circuits," ISLPED, 2015.

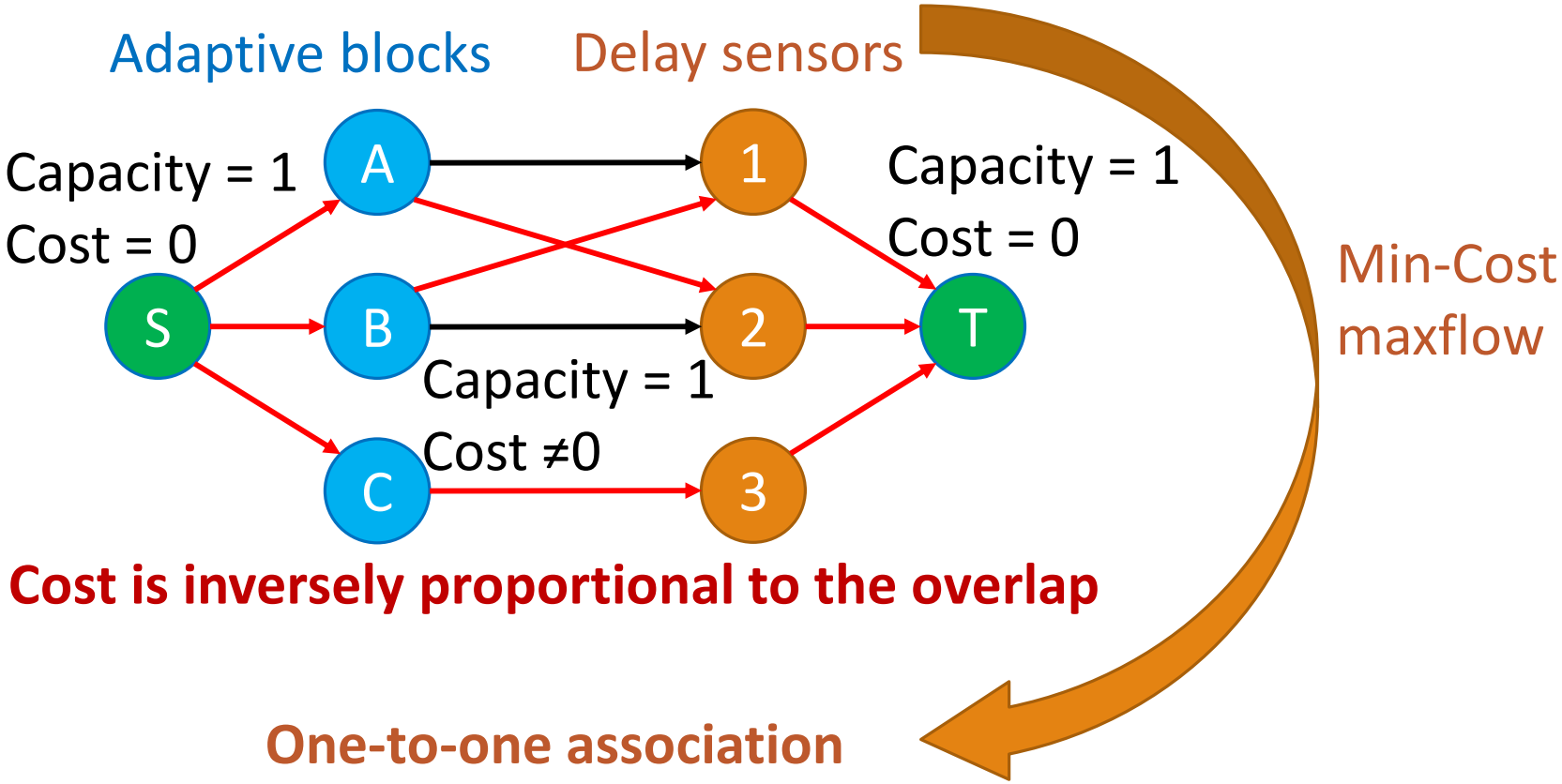
Control Synthesis



Tune some blocks to high VDD with min power overhead such that no warning signal at sensors

Control Synthesis
Rule-Based method

Turn one block to high VDD based on one sensor warning



Cost is inversely proportional to the overlap

One-to-one association

Control Synthesis

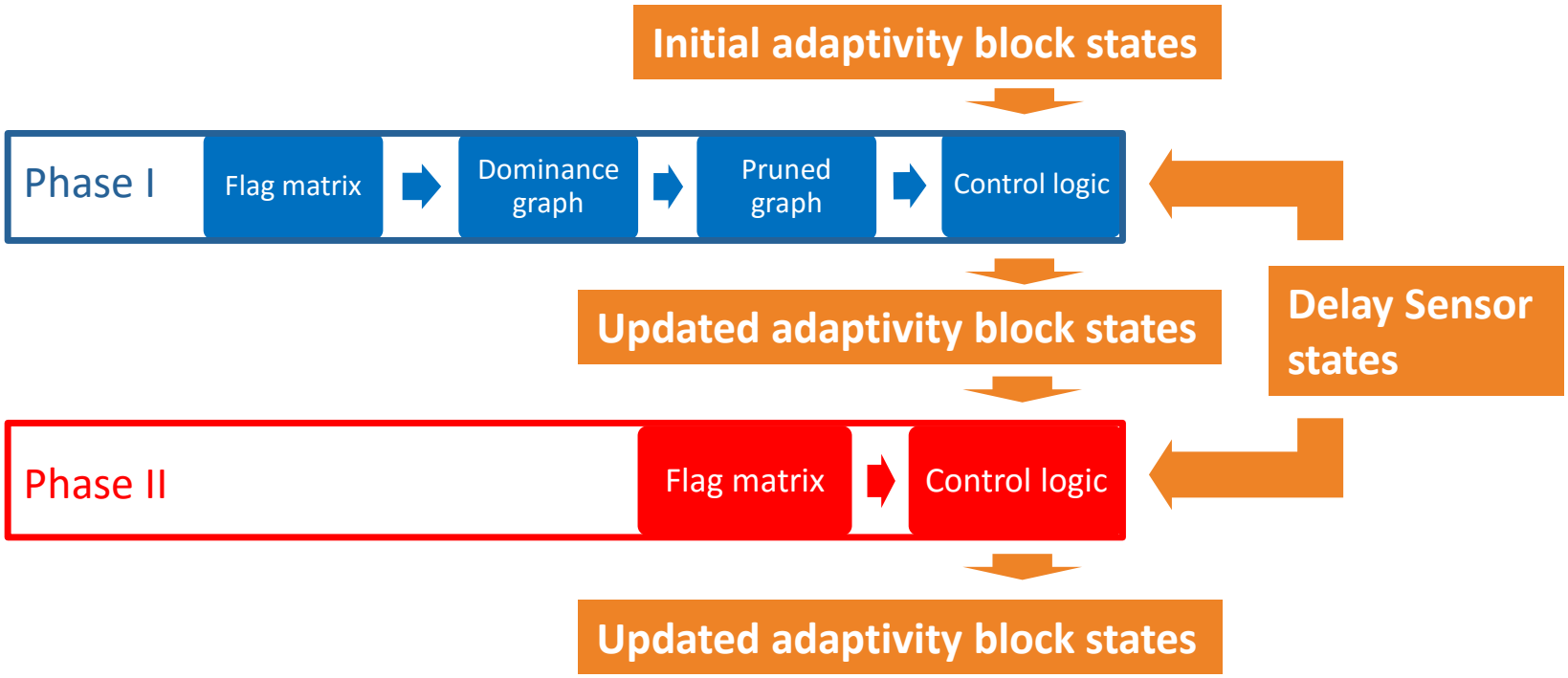
Finite-State Machine (FSM): Overview

Phase I: Initial response

- Ensure all timing warnings responded with minimum power overhead

Phase II: Incremental response

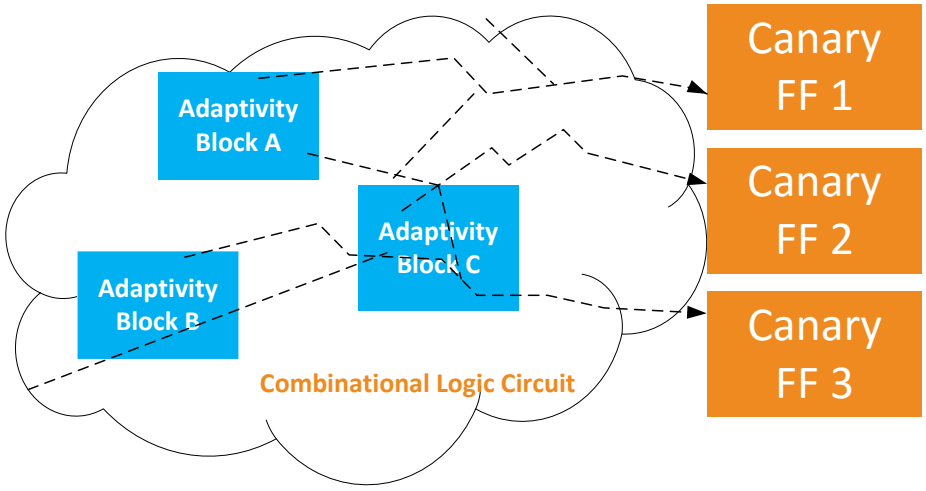
- Ensure all timing warnings removed



Control Synthesis

FSM: Inputs, states and flag matrix

Input vector	State (output) vector	Flag matrix
The states of delay sensors	The voltage levels of adaptivity blocks	Relationship between sensors and adaptive blocks



Input vector

$$s_j = \{0, 1\}$$

$$S_1 \; S_2 \; S_3$$

State vector

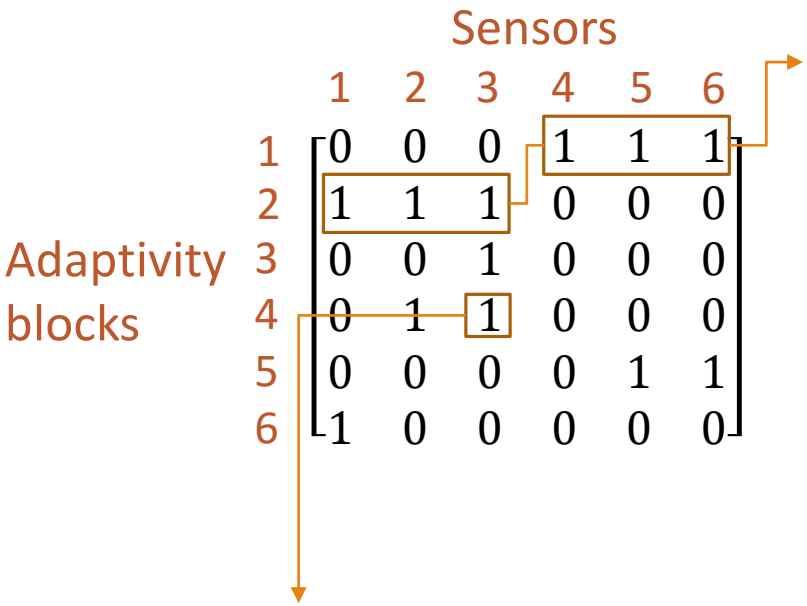
$$b_i = \{0, 1\}$$

$$\begin{matrix} b_A \\ b_B \\ b_C \end{matrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Flag matrix

Control Synthesis
Finite-State Machine: Phase I

Example flag matrix



Definitions:

Maximum Single-Response Scenario

- **$\{s_1,s_2,s_3\}$, $\{s_4,s_5,s_6\}$**

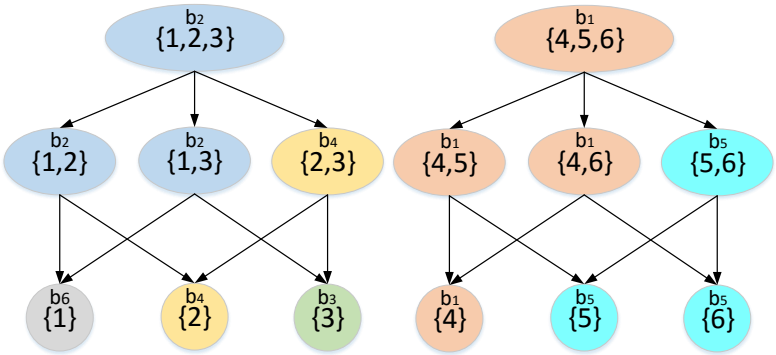
Single-Response Scenario of $\{s_1,s_2,s_3\}$

- **$\{s_1,s_2,s_3\}$**
- **$\{s_1,s_2\}$, $\{s_1,s_3\}$, $\{s_1,s_2\}$**
- **$\{s_1\}$, $\{s_2\}$, $\{s_3\}$**

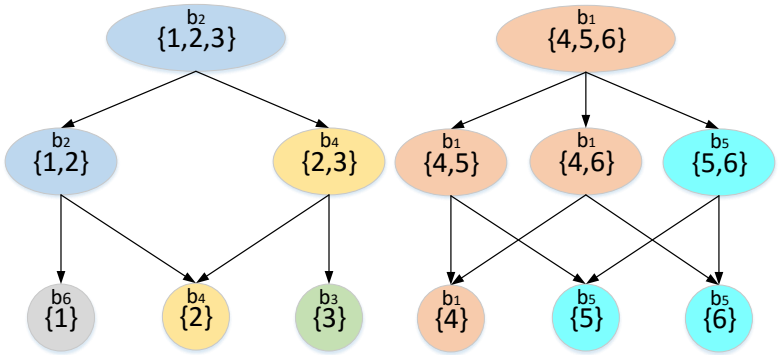
Compensated by adaptivity block #2

Adaptivity block #4 has large overlap with the fan-in cone of sensor 3

Graph Pruning
Finite-State Machine: Phase I



Before Pruning



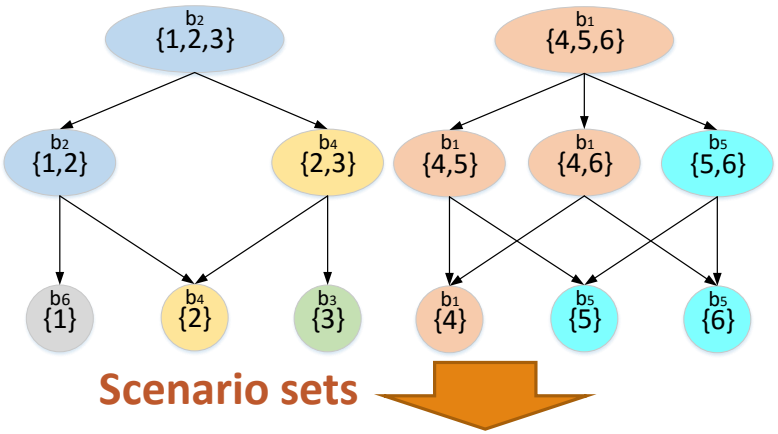
After Pruning

Redundant Sensor Scenario : Overhead larger than that successor scenario

- Overhead of b6 and b3 less than b2
- {1,3} pruned

Initial Response

Finite-State Machine: Phase I



$$\mathfrak{B}_5 = \{\{s_6, s_5\}, \{s_6\}, \{s_5\}\}$$
$$B_5^{max} = \{s_6, s_5\}$$
$$\mathfrak{B}_5^{min} = \{\{s_6\}, \{s_5\}\}$$

$$\mathfrak{B}_4 = \{\{s_2\}, \{s_3, s_2\}\}$$
$$B_4^{max} = \{s_3, s_2\}$$
$$\mathfrak{B}_4^{min} = \{\{s_2\}\}$$

$$b_5 = (s_6 + s_5) \cdot (\overline{s_4})$$
$$b_4 = s_2 \cdot \overline{s_1}$$

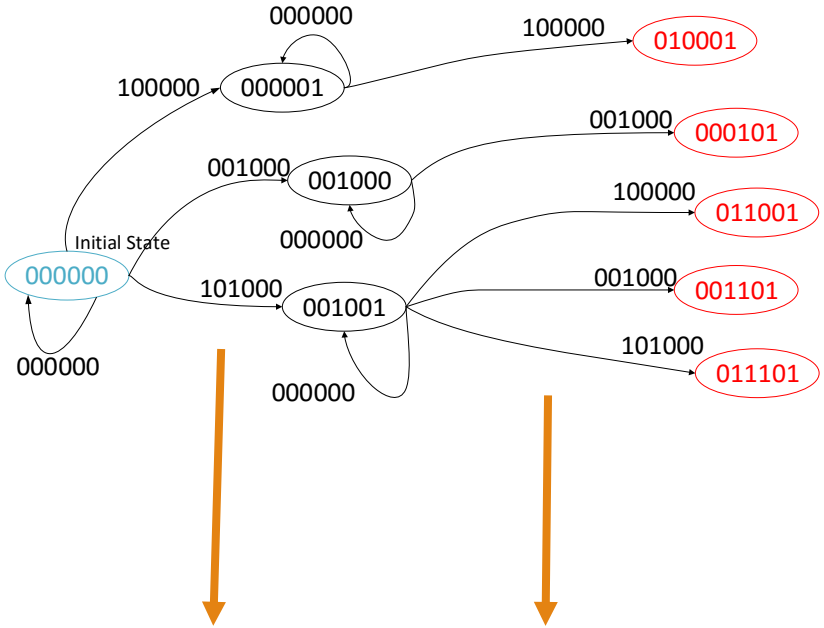
Min scenario is a minterm

$$\{1, 2\} \rightarrow s_1 s_2$$
$$\{\{5\}, \{6\}\} \rightarrow s_6 + s_5$$

Complement of max scenario is a minterm

$$\{4, 5, 6\} - \{5, 6\} \rightarrow \overline{s_4}$$
$$\{1, 2, 3\} - \{2, 3\} \rightarrow \overline{s_1}$$

Incremental Response
Finite-State Machine: Phase II



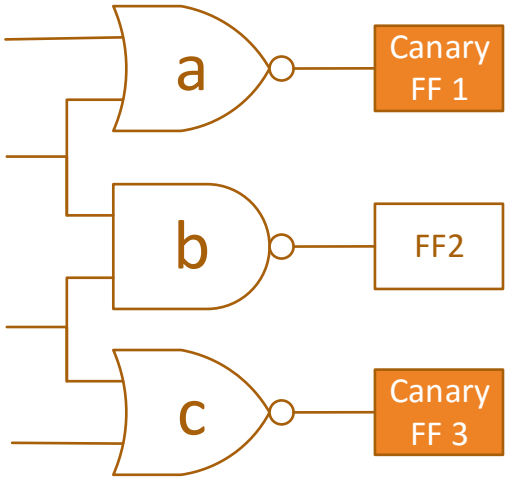
- Partial Finite-State Machine:
  - Numbers in the circle denote tuning knob states
  - Number by the edge denote sensor states

Initial Response | **Incremental Response: Greedy changes until no warnings**

Delay Sensor Deployment

Goals of delay sensors

Monitor all paths whose delay variations pose a risk on timing violation



- **Large overlap among timing paths to different flip-flops**
- **Need tradeoff between coverage and criticality**

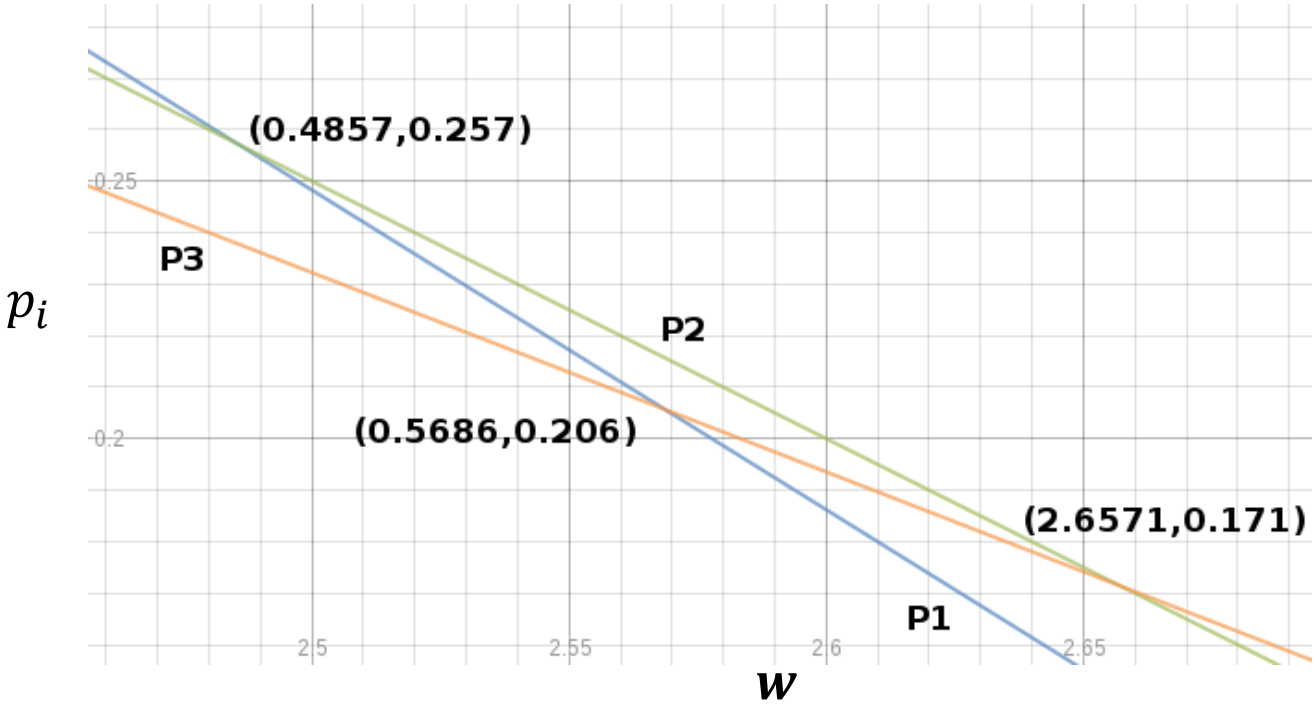
Delay Sensor Deployment

- **Formulation 1: maximize coverage with number of delay sensors no more than a upper bound**
  - **Iteratively select FF with maximum priority value**
  - $\boldsymbol{p_i = (d_i - w) \cdot \bar{d}_i / d_i}$

Delay Sensor Deployment
Example of Formulation 1

$$p_i = (d_i - w) \cdot {}^{\bar{d}_i}/_{d_i}$$

d_i is the total path delay, $\bar{d}_i$ is the uncovered path delay

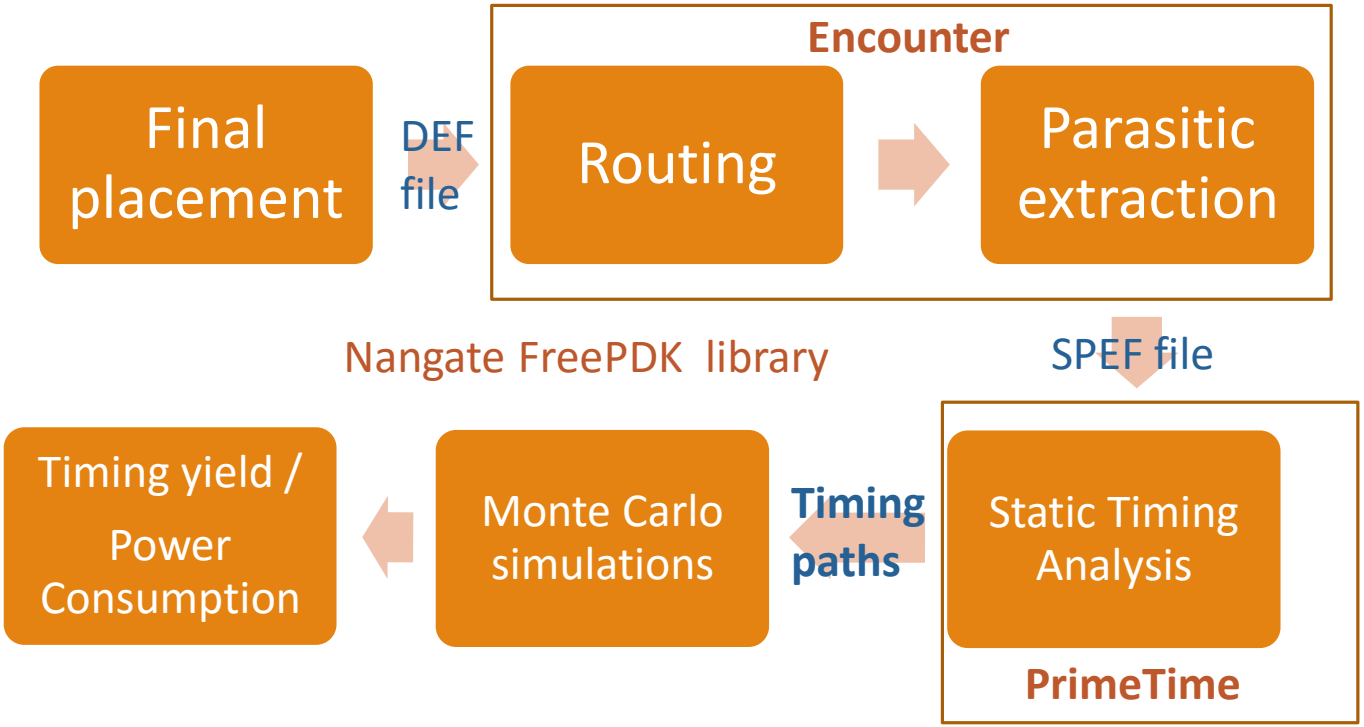


$d = (2.9, 3, 3.1)$
$\bar{d} = (1.8, 1.5, 1.2)$

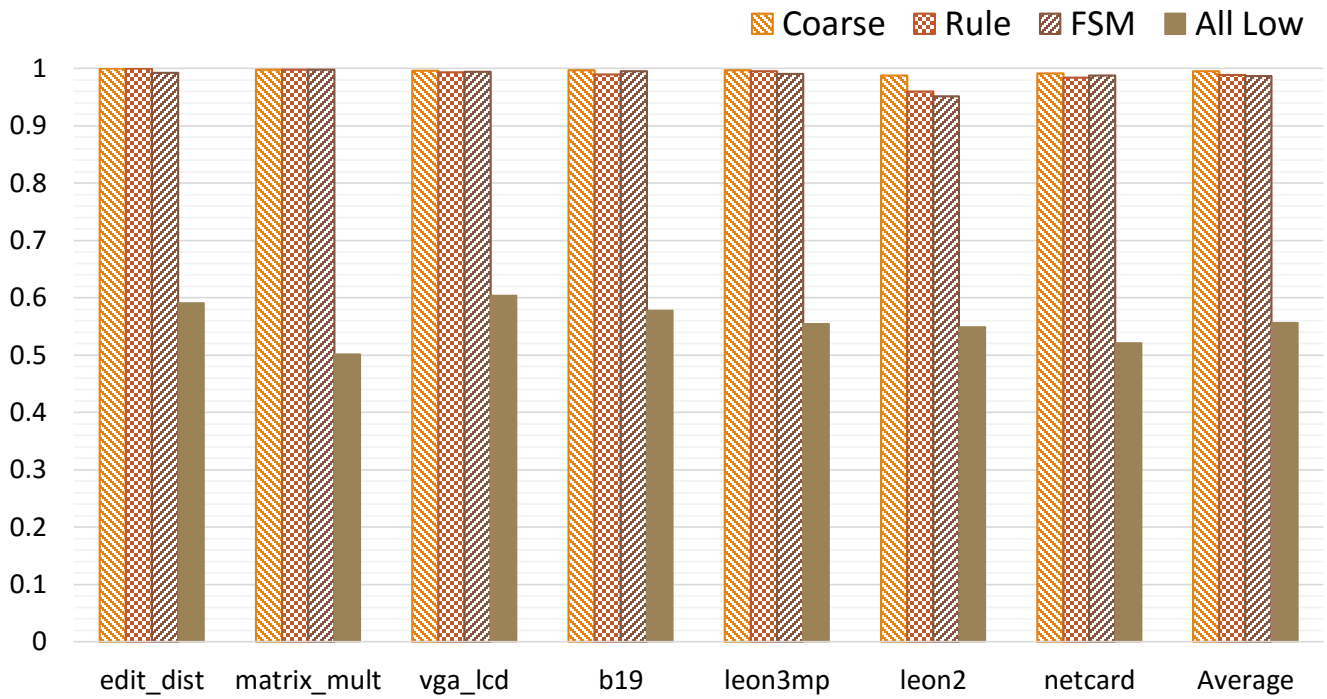
Delay Sensor Deployment

- **Formulation 2: minimize number of delay sensors with coverage no less than a lower bound (Set Cover Problem)**
  - **Covered if $p_i = {\bar{d}_i}/{d_i} < p_{th}$**
    Find minimum number of flips-flops that make almost all critical paths covered
  - **Solved iteratively by choosing FF with maximum p**

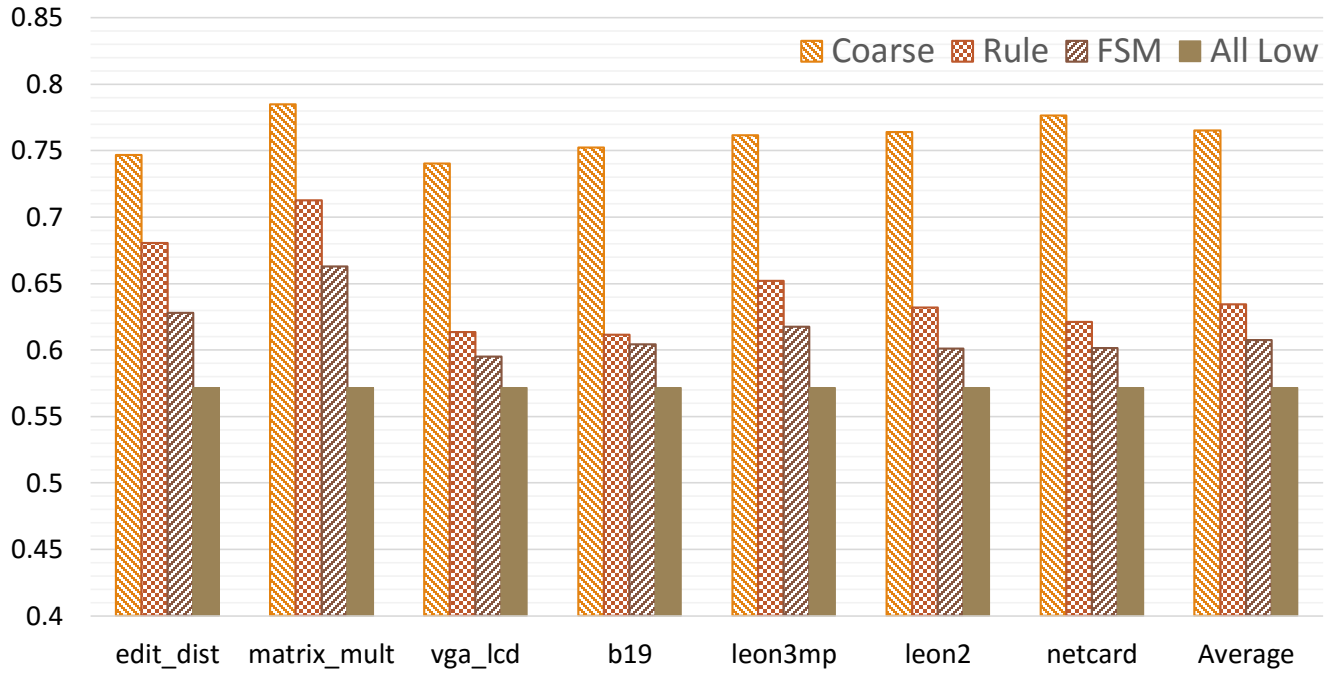
Experiment


Encounter
Final placement
DEF file
Routing
Parasitic extraction
Nangate FreePDK library
SPEF file
Timing yield / Power Consumption
Monte Carlo simulations
Timing paths
Static Timing Analysis
PrimeTime

Results
Timing Yield



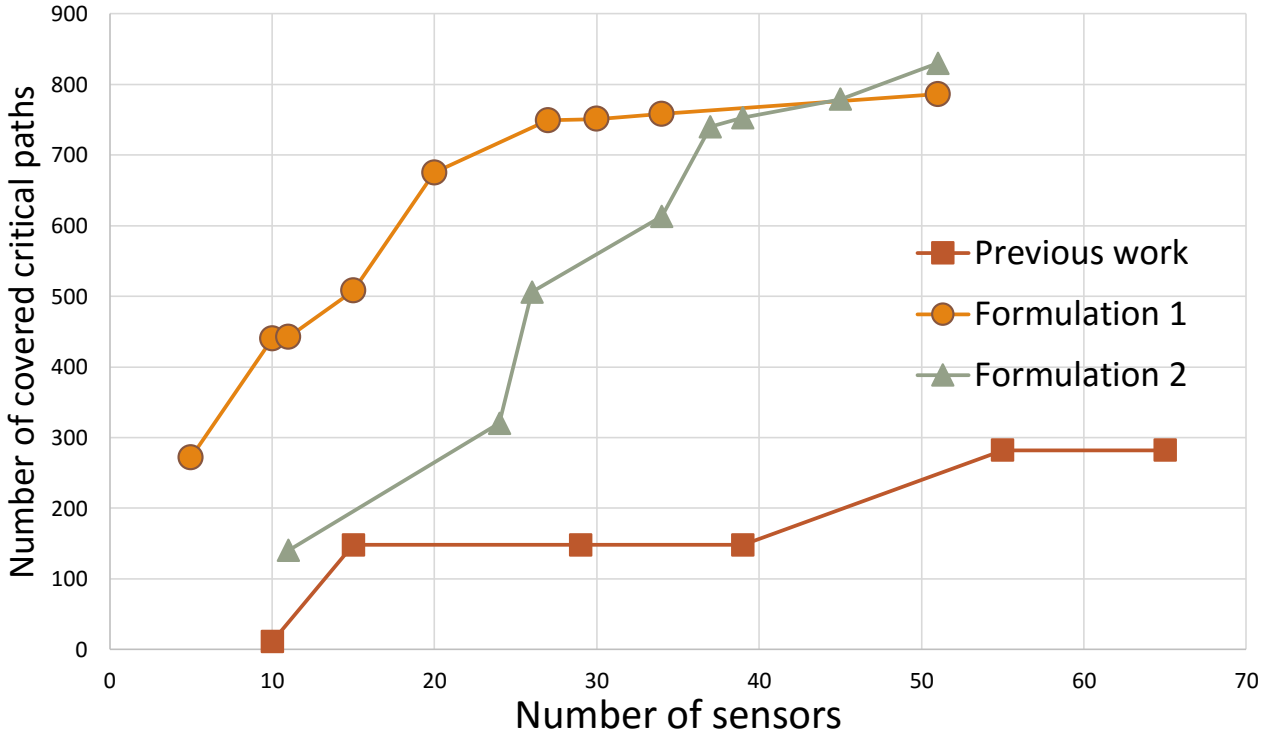
Results
Leakage Power



Results

Delay Sensor Deployment

Result for b19



Conclusion

Proposed two control synthesis methods for efficient ASV designs

- **Rule-based method**
- **Finite-state machine method**

Proposed delay sensor deployment method that achieves better balance between

- **Coverage of timing paths**
- **Criticality of timing paths**

Our methods achieves approximately 20% leakage power reduction compared to coarse-grained designs

Thanks

Q & A