

Technology Aware Design (TAD) VAM & SKM

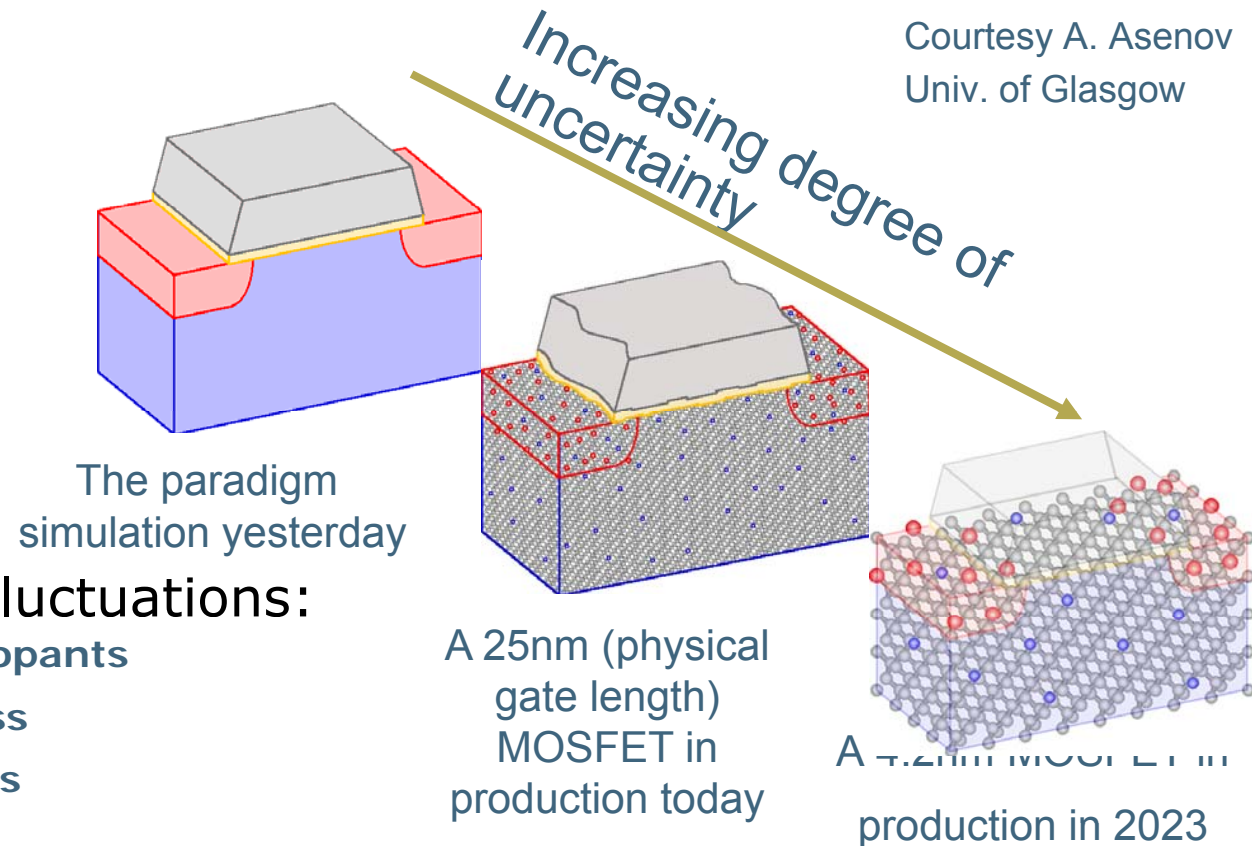
HiPEAC2 – TF on Reliability and Availability,
Paphos, Cyprus, Jan. 2009

S. Mamagkakis
on behalf of the TAD team



The next generation MOSFETs are atomic scale devices

Courtesy A. Asenov
Univ. of Glasgow

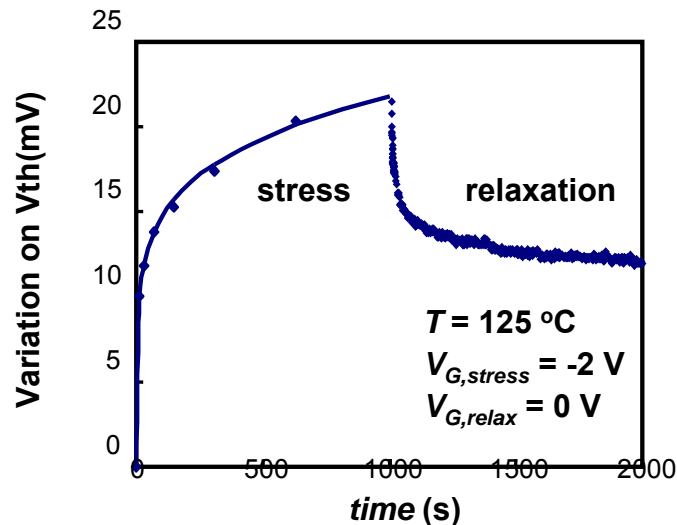


- **Intrinsic Process Fluctuations:**

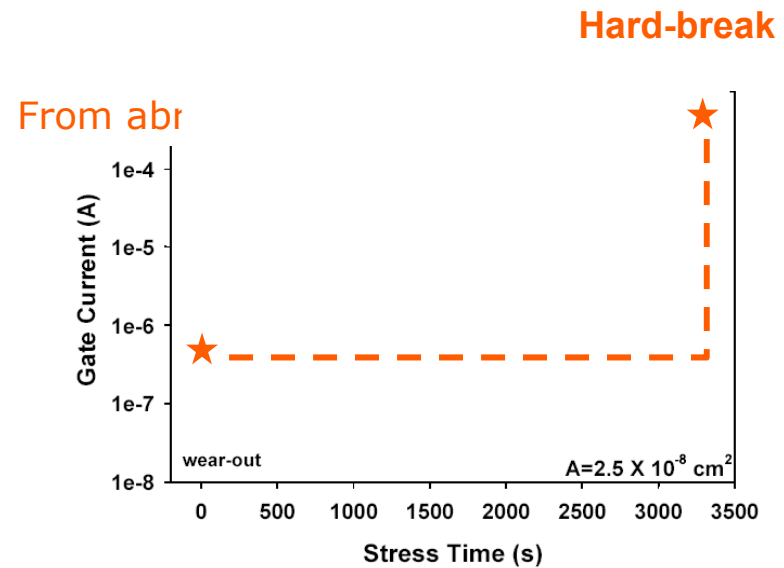
- Random discrete dopants
- Line edge roughness
- Interface roughness
- Poly silicon gate
- Strain
- High-k structure
- Gate tunnelling
-

Electrical degradation: from abrupt to progressive

- Vdd stops scaling due to e.g., variability increase
- Smaller geometries and new (less characterized) materials
- Steady increase in electric fields → from abrupt failure to gradual degradation

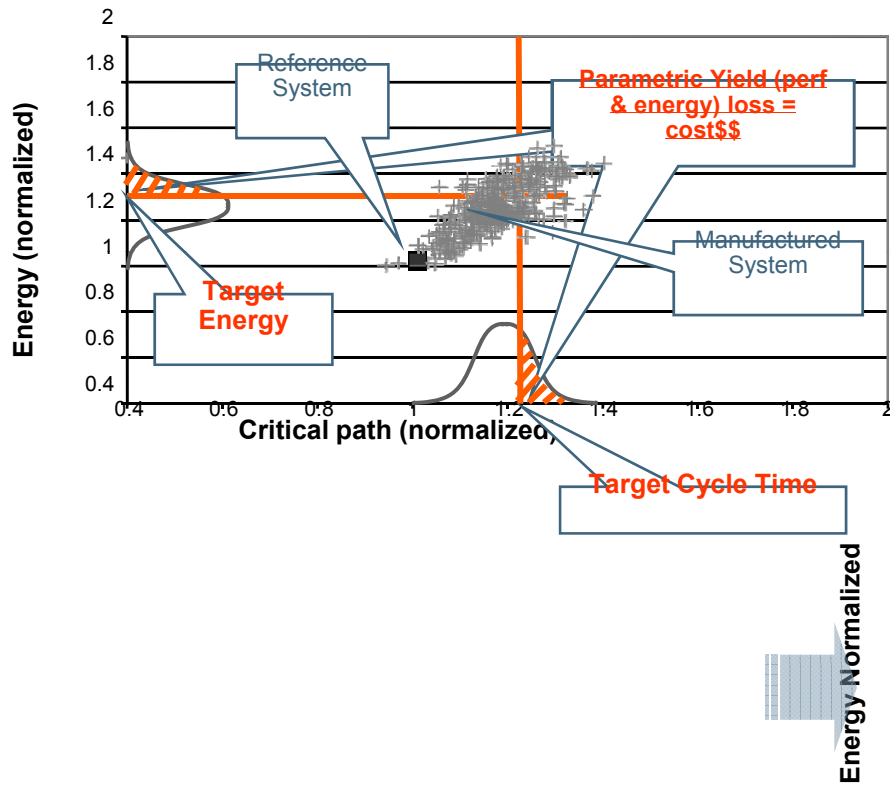


Negative Bias Temperature Instability

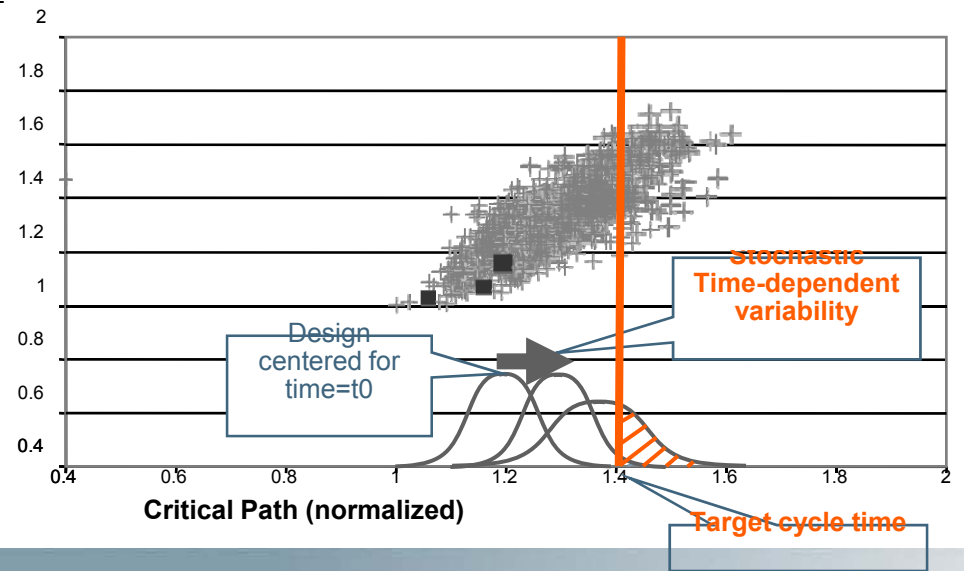


wear-outs in time-dependent Dielectric Breaks

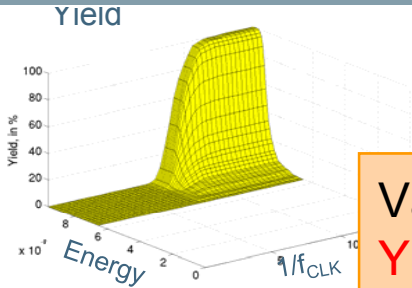
Gradual degradation + Intrinsic Process Fluctuations = Time-Dependent Process Variability



Time-dependent variability severely impacts overall system reliability



Variability Aware Modeling (VAM)



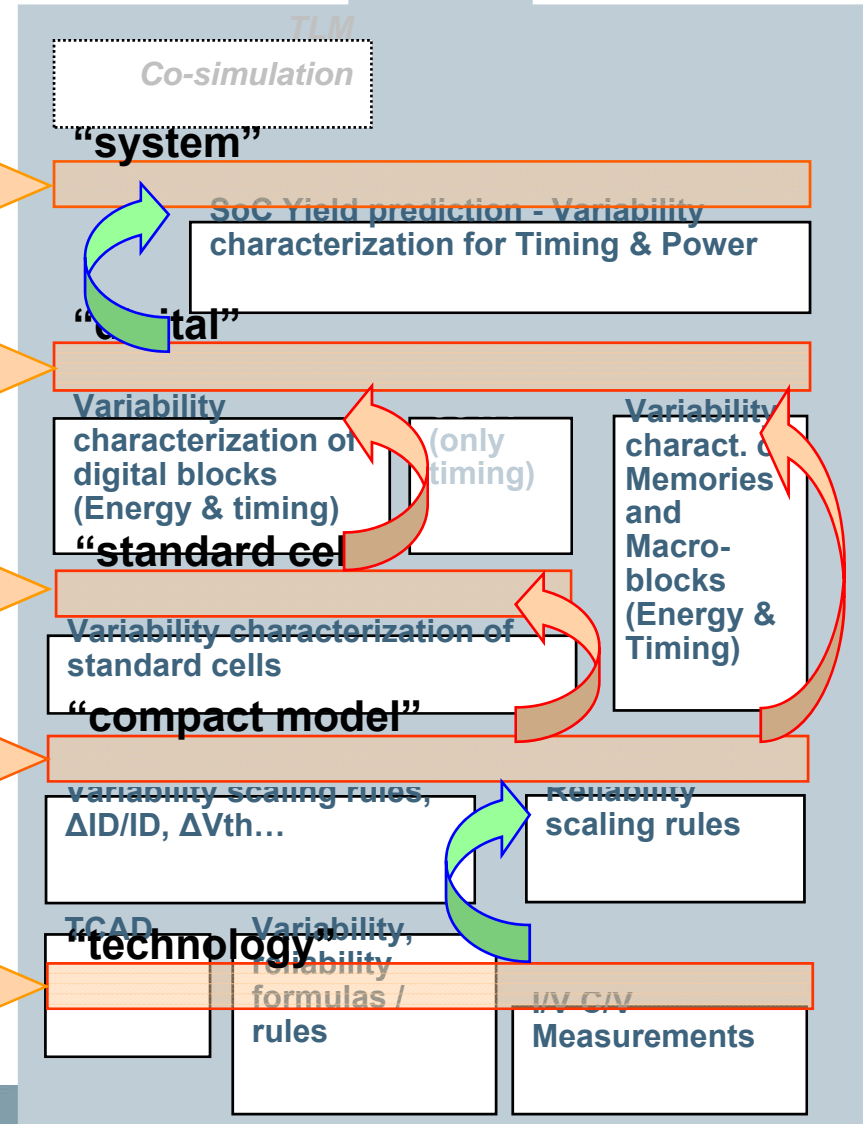
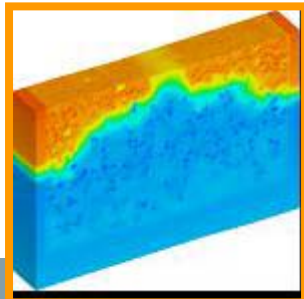
Variability =
YIELD

Variability =
Delay&static&dynamic

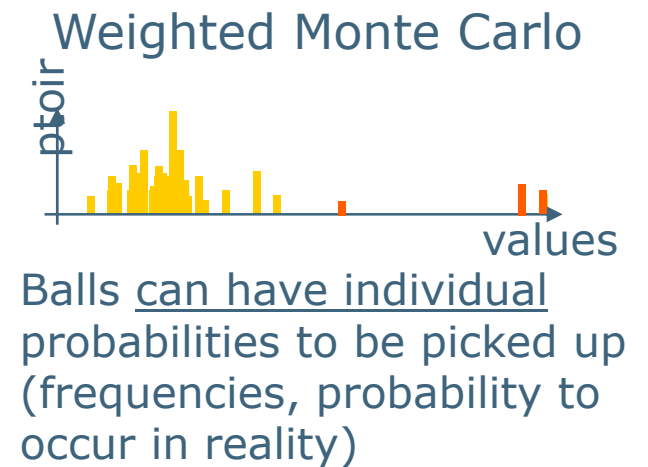
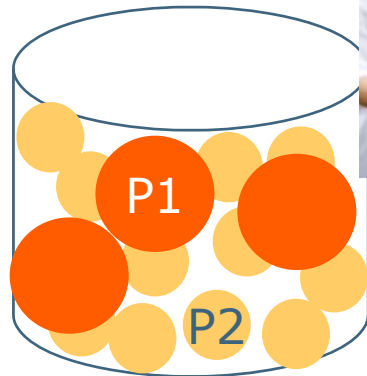
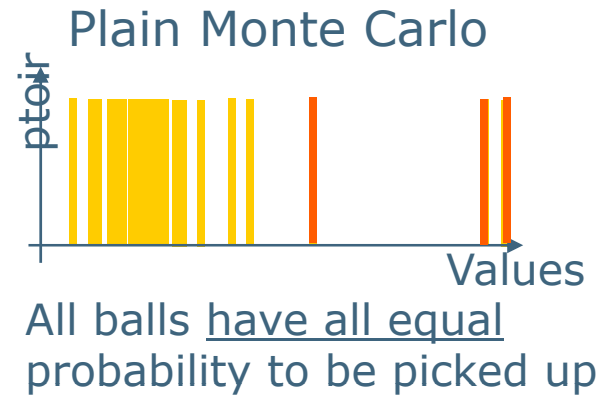
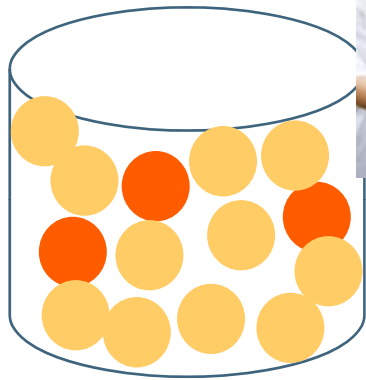
Variability =
Delay&energy (.lib)

Variability =
electrical (V, I, R, C)

Variability =
geometrical & chemical

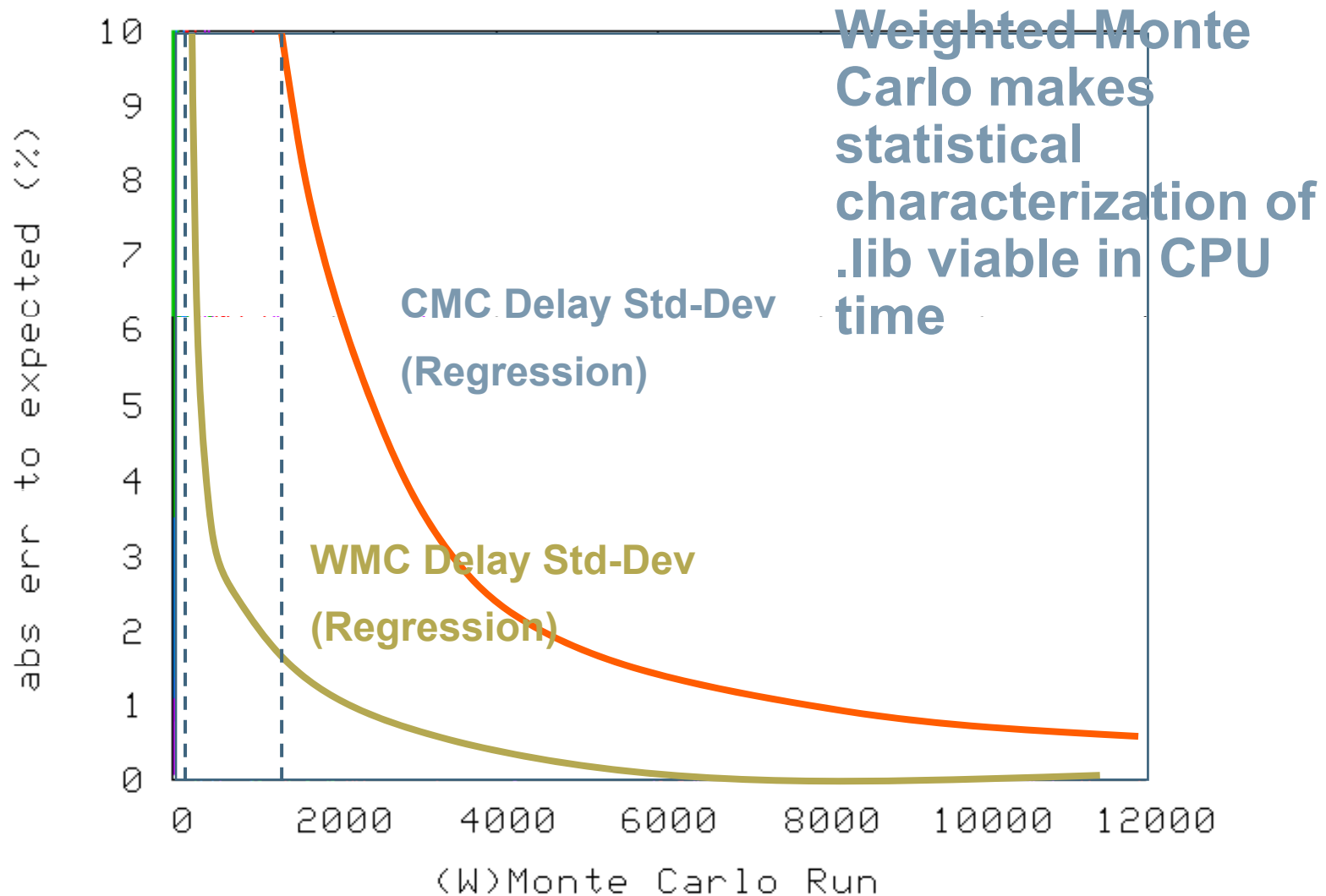


What is Weighted Monte Carlo (WMC)?

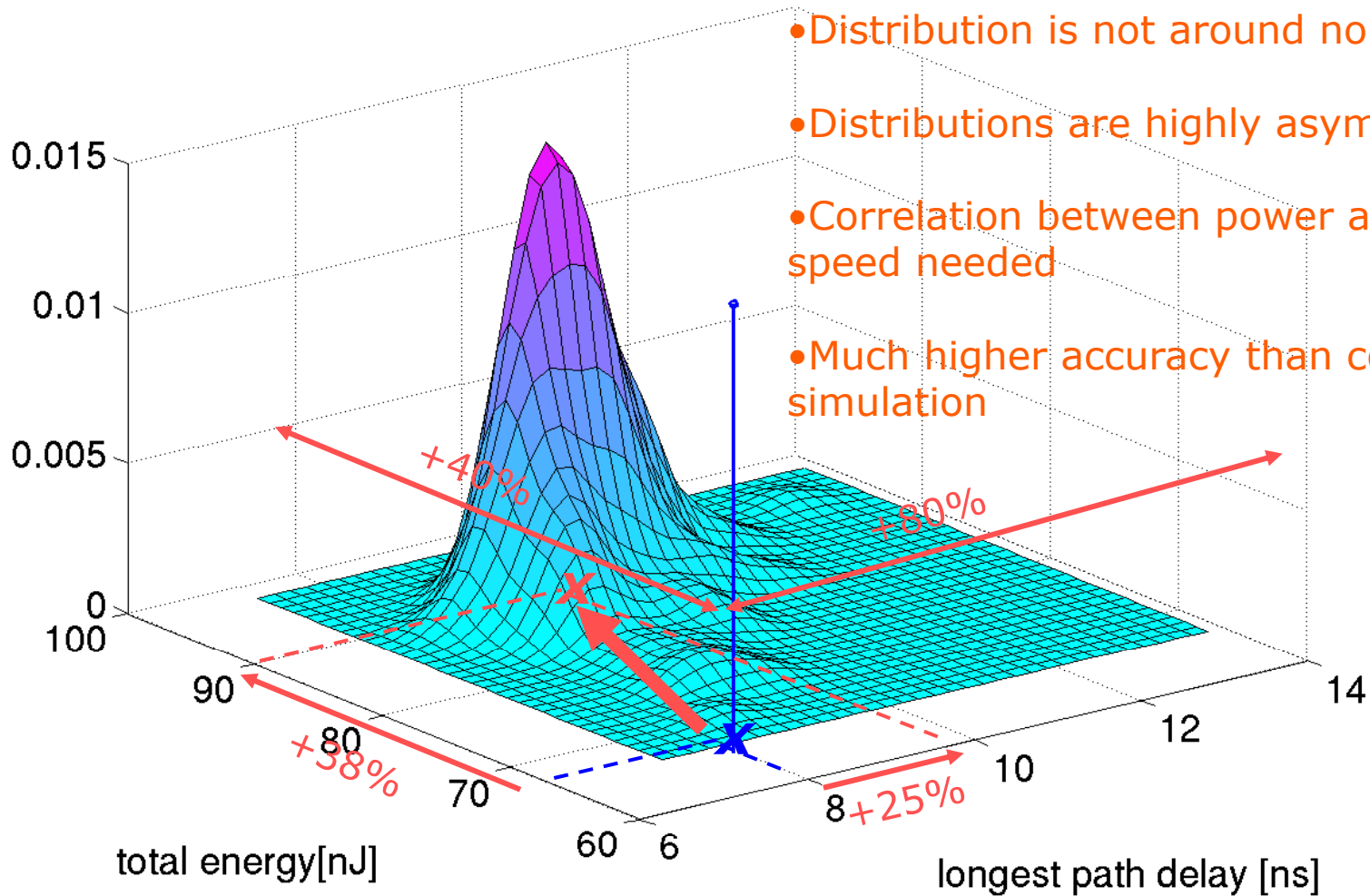


Monte Carlo vs. Weighted Monte Carlo

1.5h 50h



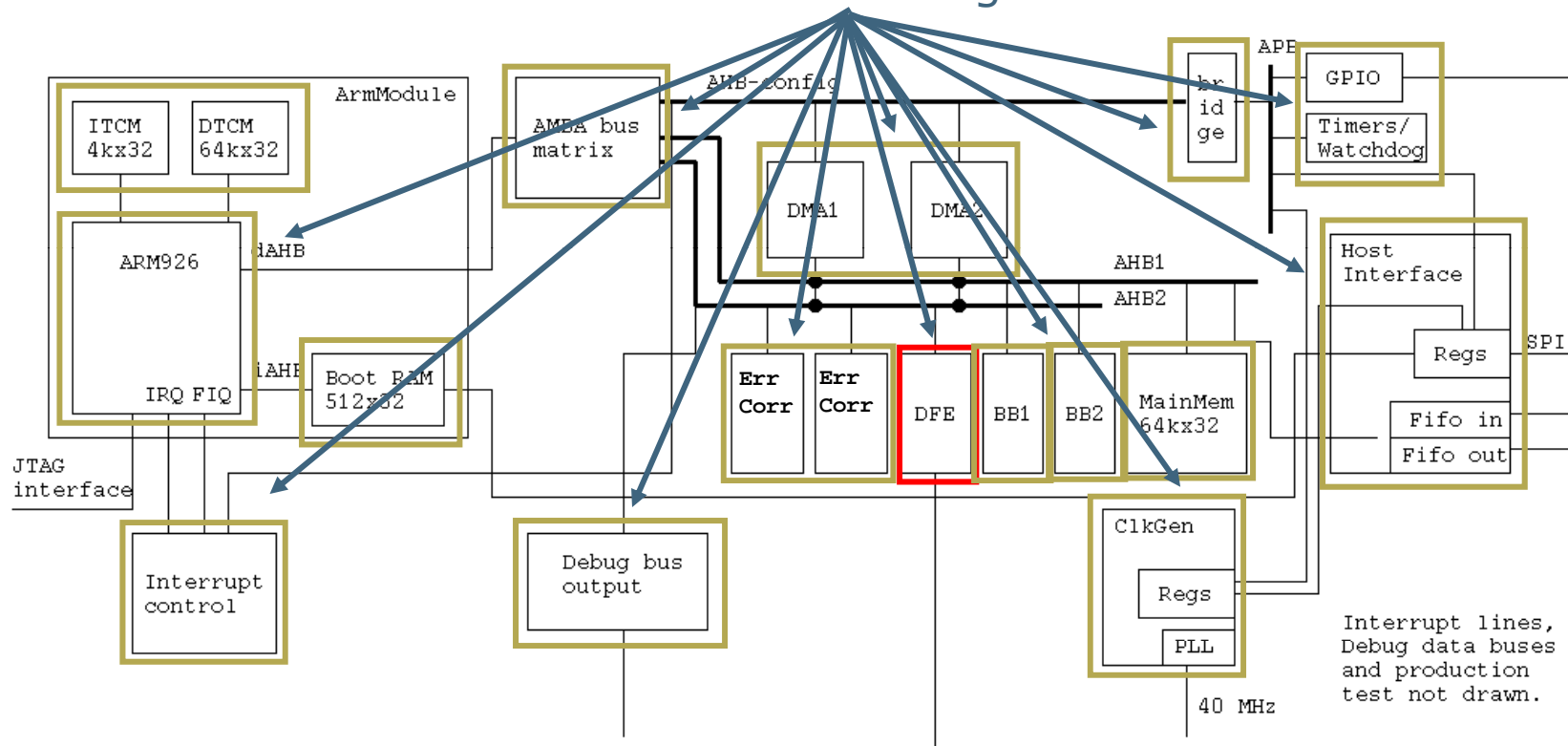
Statistical Output of VAM on Digital Front-End Processor



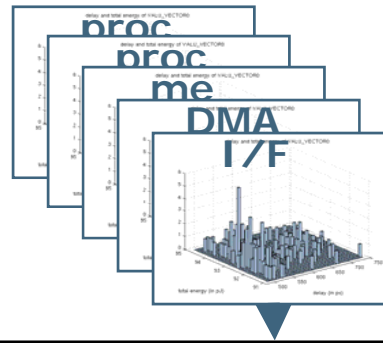
- Dynamic energy has variability!!
- Distribution is not around nominal
- Distributions are highly asymmetric
- Correlation between power and speed needed
- Much higher accuracy than corner simulation

A divide and conquer approach is needed for statistical modeling of SoCs

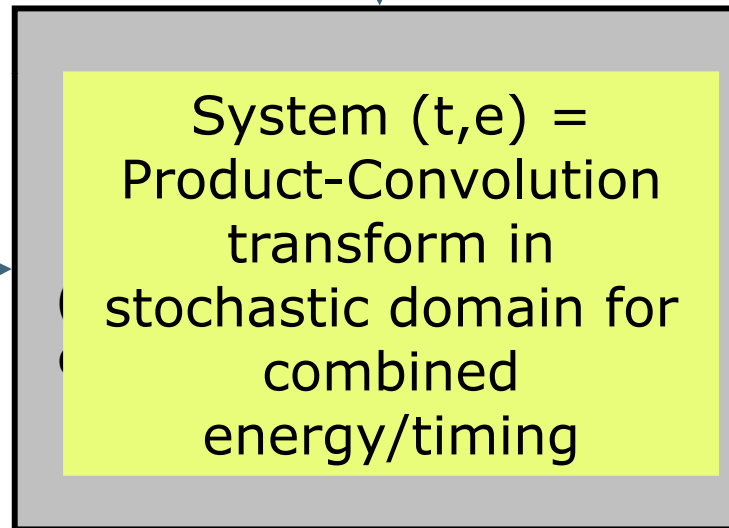
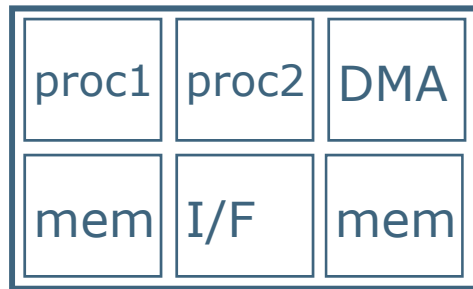
Raise abstraction and bring statistics to SoC level



Integration of component-level statistics to obtain SoC level performance/power variations



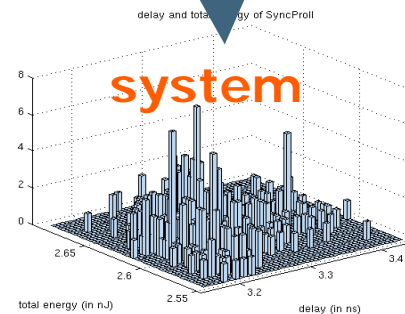
Architecture info



Application info



RTlevel activity



What Can VAM do Intrinsically that Cannot be Done Otherwise

1. **Statistical Analyses of Memories**
 - SSTA and other existing methods handles only Standard Cells
2. **Statistical Analyses of Blocks/Systems**
 - Above levels handled by SSTA or other methodologies
3. **Can Handle Outliers**
 - All other methodologies (e.g. RSM) assume well behaved populations
4. **Practical Statistical Analyses via WMC**
 - Intractable problem for Monte Carlo
5. **Compatible with PathFinding Goals**
 - Maintain all the correlations up to System level - given a Predictive Model
6. **Compatible with Managing Reliability in Design**
 - Framework handles reliability in same way as variability

Nassif, IBM:

“IMEC’s VAM is the only relevant method beyond SSTA”

Reliability mechanisms extensions

- Negative Bias Temperature Instability (NBTI) ✓
- Hot Carrier Degradation (HCD) ✓
- Soft Break Down (SBD in oxide) ✓
- Soft Error ✓
- Breakdown in interconnect (TBD)
- Electro Migration
- ...

How to design under uncertainty



Industry need → Overall project targets

Offering design solutions to 45-sub nm scaling issues

-modeling the imperfection

-how to design with unreliable and unreproducible circuit parts

- 1. Enable propagating variability information up in the standard simulation flow (VAM)**
 1. A standard way to communicate variability information, from foundry to system specification
 2. Enable variability and reliability awareness in simulation flow
 3. Allow assessment of technology and design options in view of reliability and variability
 4. Signoff in view of parametric and functional yield
- 2. Enable a large class of runtime solutions for variability and degradation issues (SKM)**
 1. Abandon the energy hungry "worst case" (guardband, corner simulation) design paradigm
 2. Reduce the burden to industry acceptance of these solutions

Focus 2: Standardized Knobs & Monitors (SKM) Run-time countermeasures

A large family of runtime countermeasures
← fine granularity run time tuning
← does not change hardware design practice
→ exchange speed and power continuously at run time

“Better than worst case” –design (BTWC)

- Circuit parts are nominally designed to be “just fast enough”, with “just enough energy”
- Have an alternate mode which has guaranteed in spec, at the expense of power dissipation
- At run time, most circuit parts run at “just enough energy”; few knobs must be turned high.

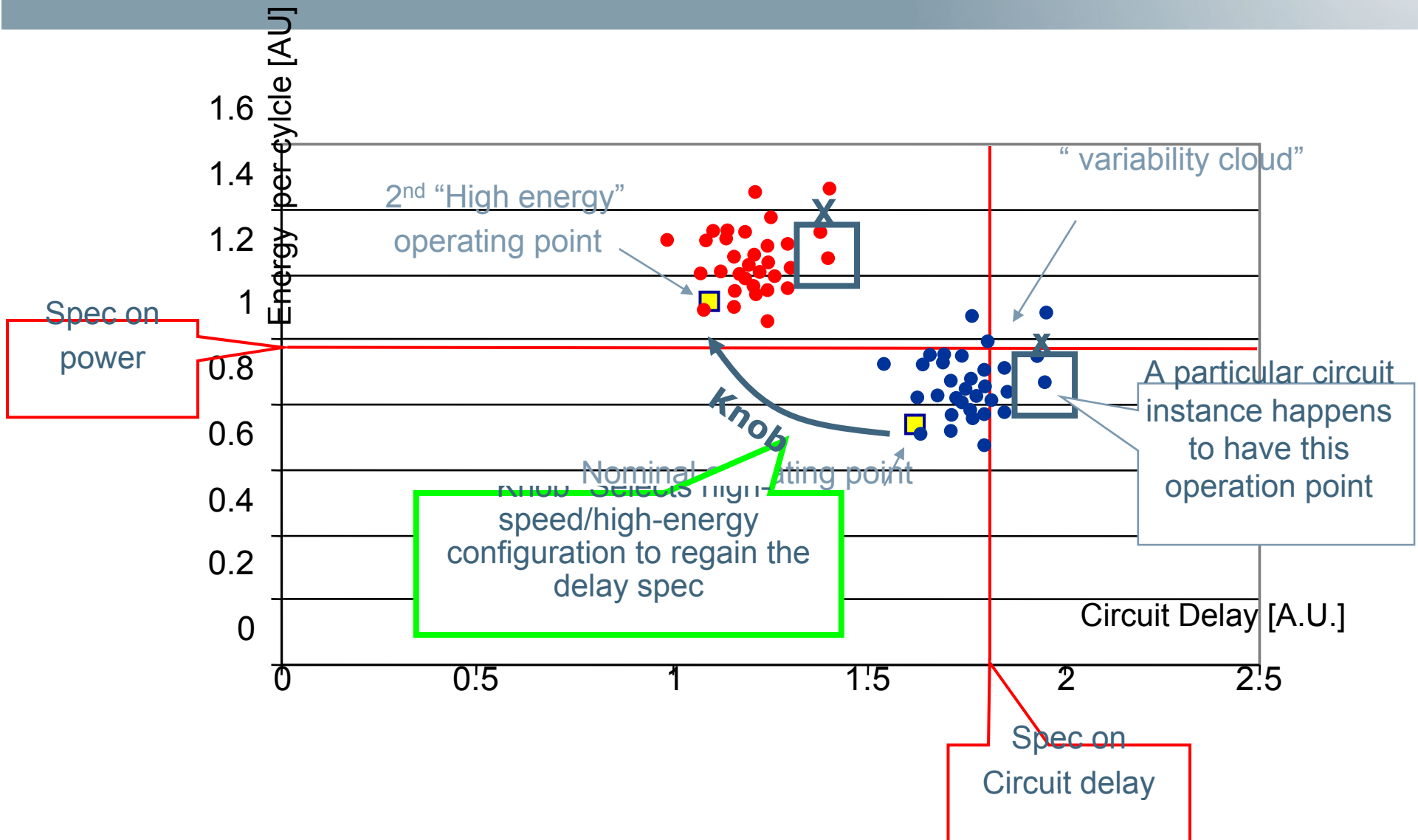
Correction is at run-time. It can thus also compensate for:

- Temperature drift
- Ageing; several degradation effects.

32nm variability relaxes to 65nm “feel”

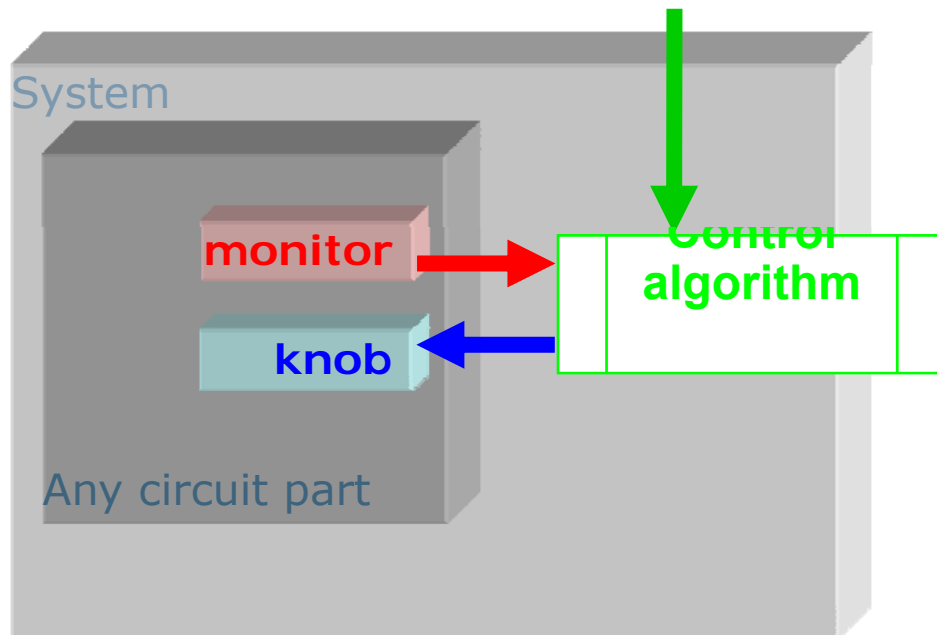
- Spread of variability “cloud” is effectively confined by the SKM system

Fine-grained Knobs & Monitors (SKM)



Equip circuit parts with “Knobs & Monitors”

Application
Environment parameters
Technology knowledge



Example of a Monitor:

(Near) Timing violation circuit

Example of Knobs:

Vdd, Freq., Line drivers with programmable current

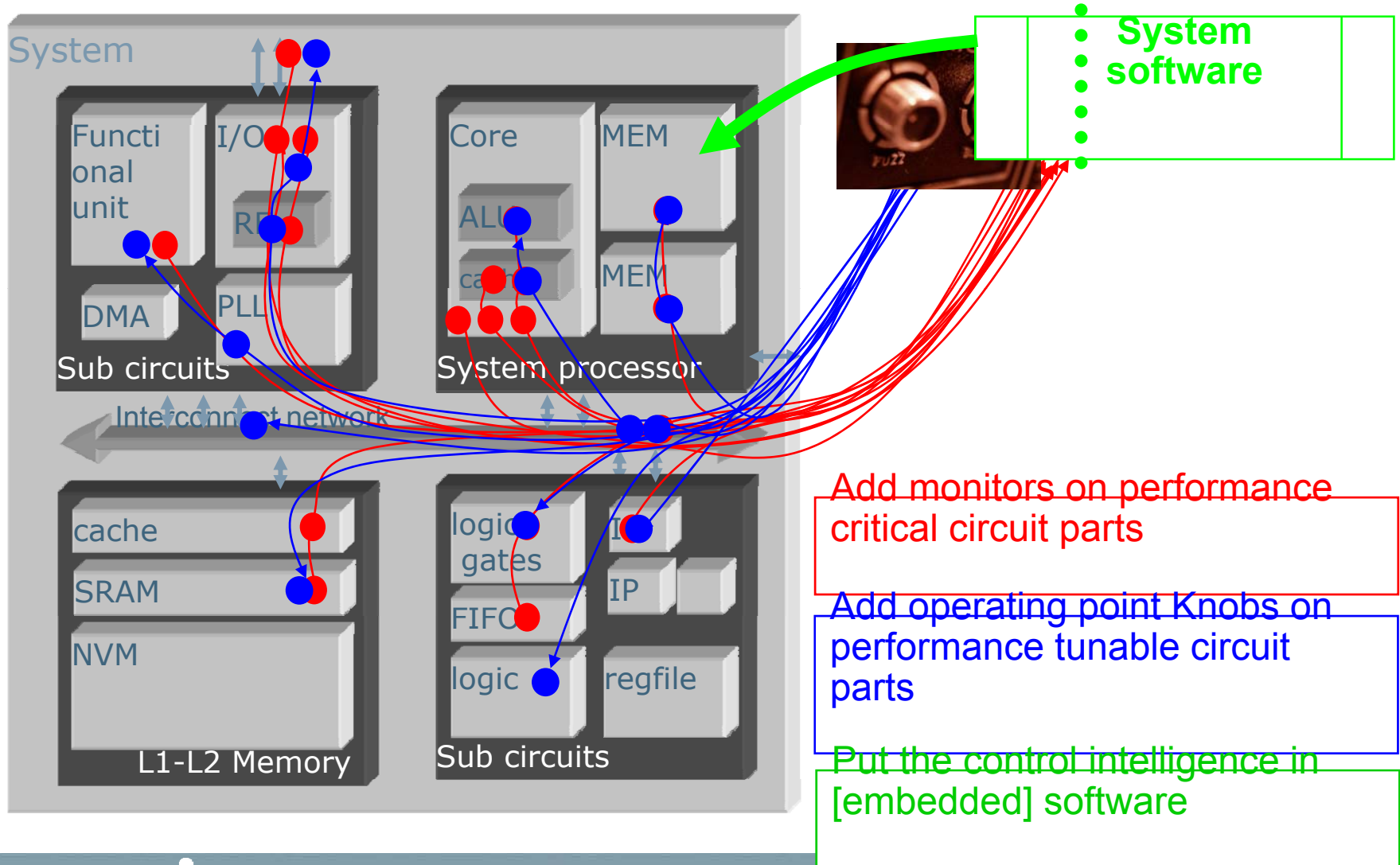
- What consists of?
 1. **Monitor** flags spec (near-) failure
 2. **Knob** changes circuit's operating point so as to regain spec
 3. **Control algorithm**

Knobs and Monitors interact via the circuit part's I/O

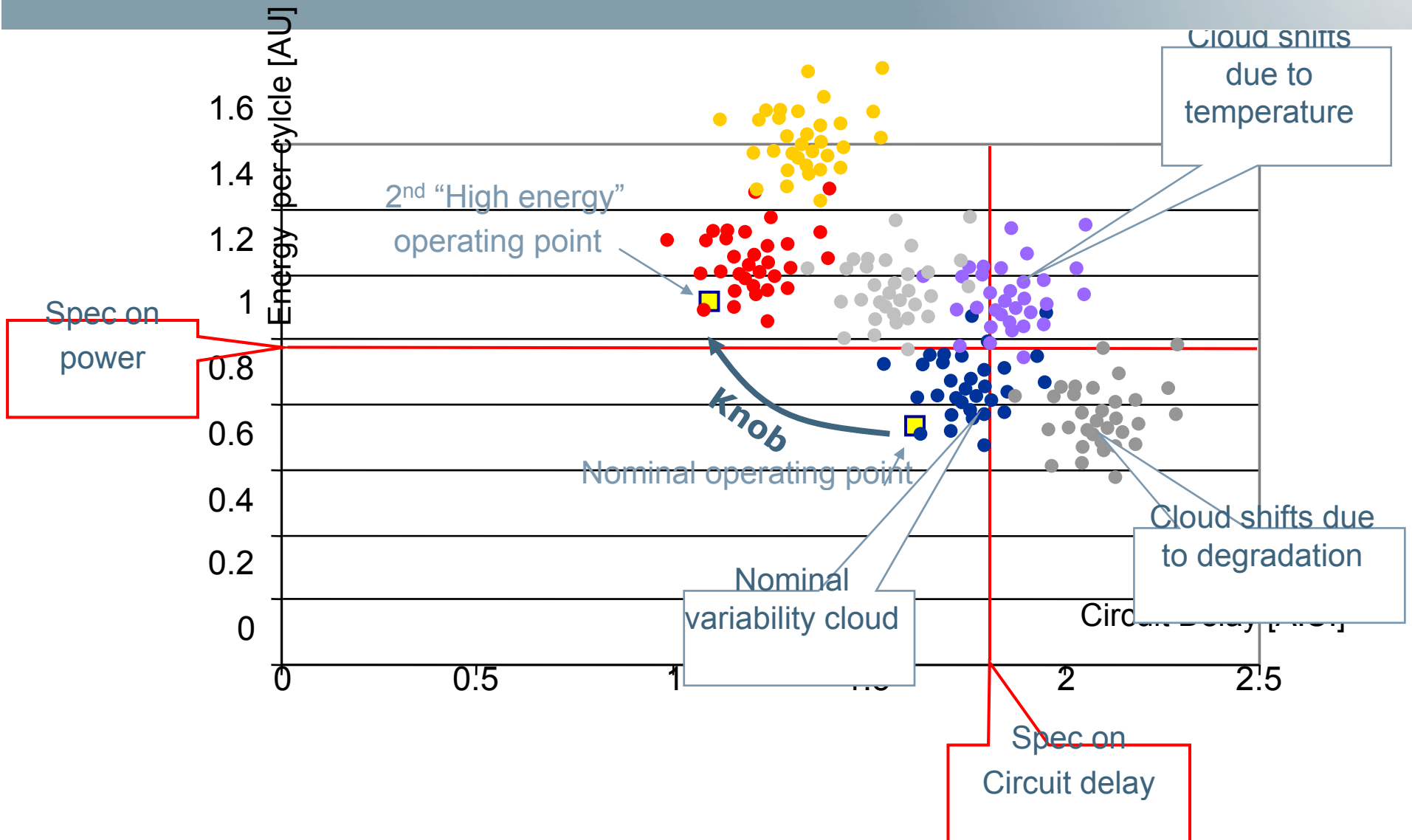
- Why standardizing:
 - Acceptance by HW design community
 - Interchangeability of each part
 - Delegation of design (even across companies)
 - Control algorithms become abstract paradigms

Focus 2

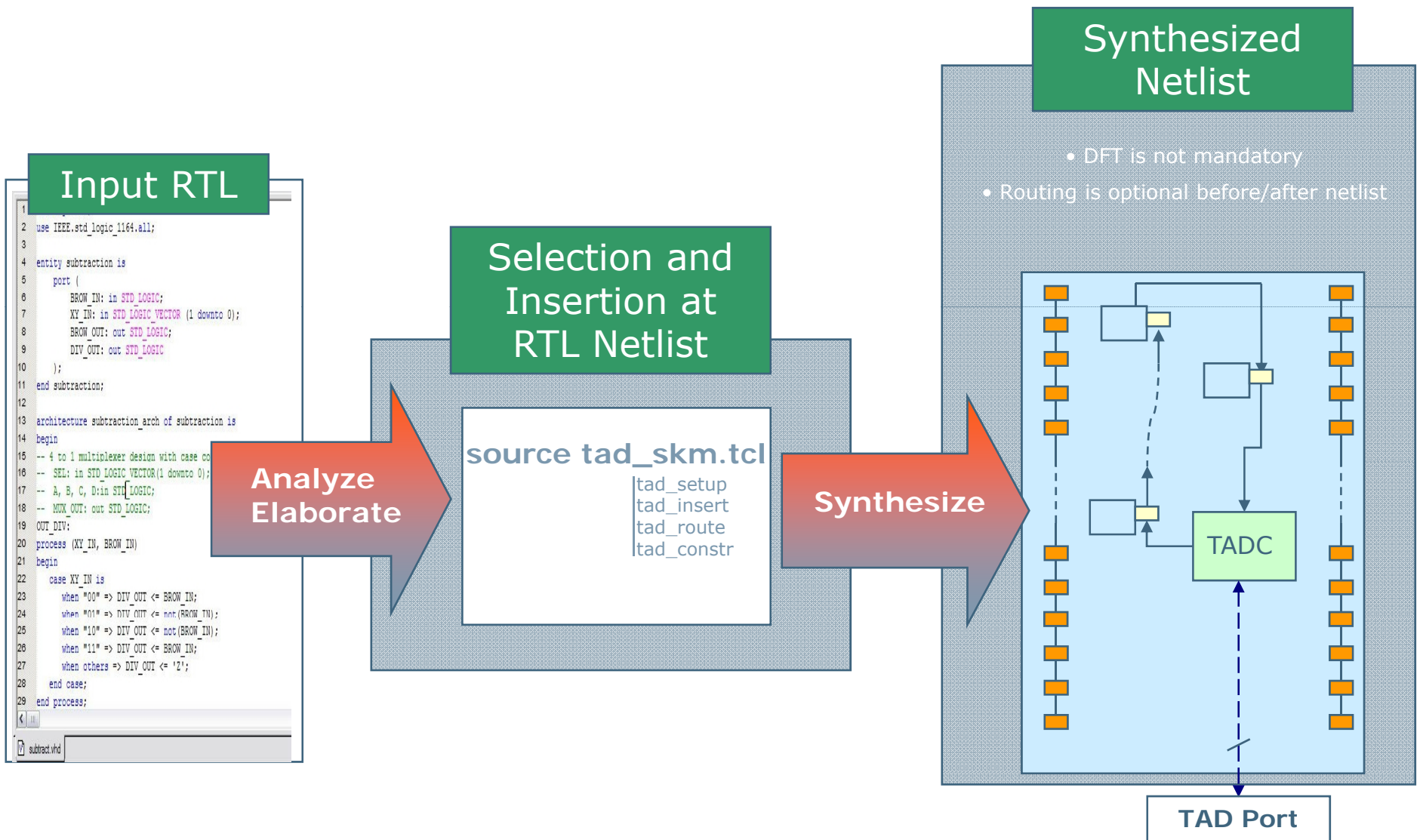
Fine-grained Knobs and Monitors in SoC



Temperature drift and ageing

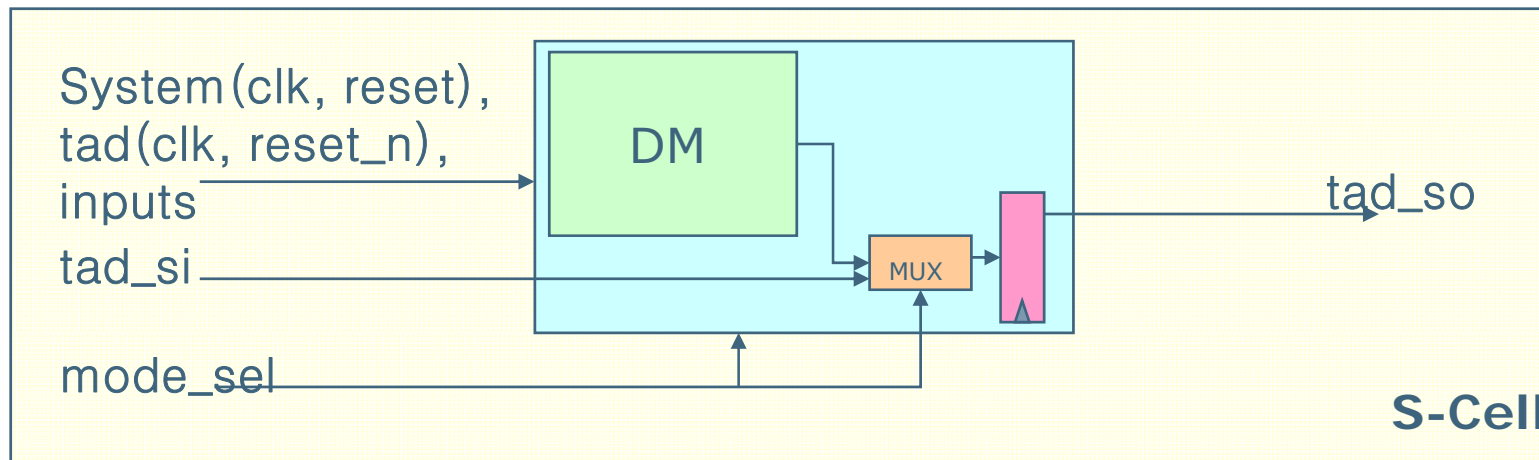


Monitoring the system:

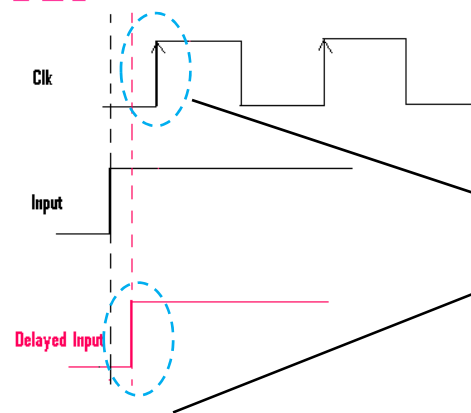
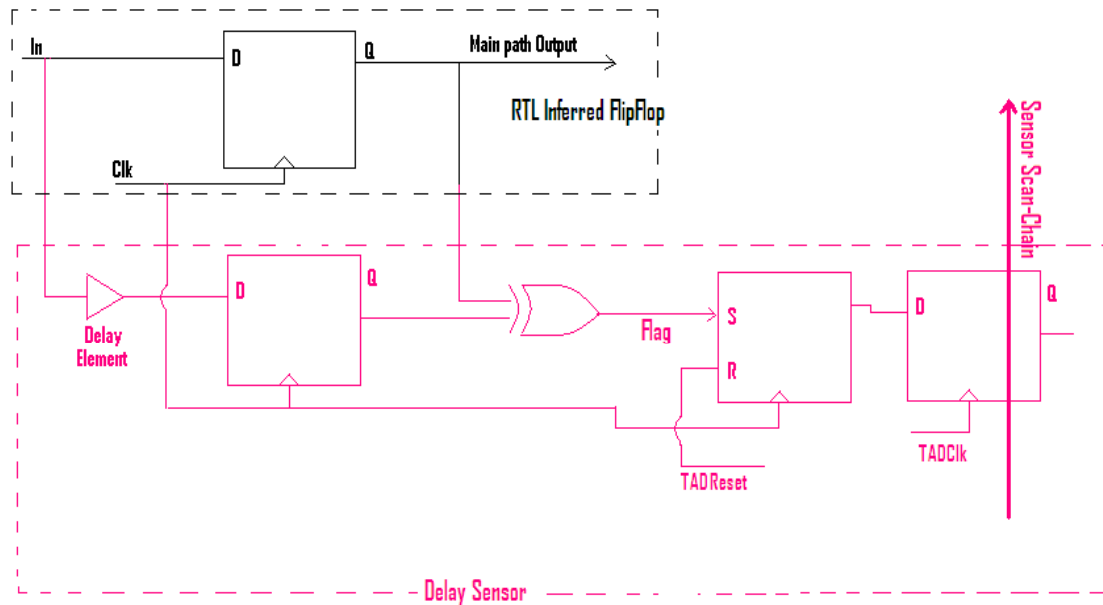


Standard Monitor: S-Cell characteristics

- A white box (=behavioral) representation of monitor included in a Scan-Cell.
- System and Monitor circuitry synthesized by original synthesis flow
- TAD insertion flow interfaces with synthesis tool after elaborate register allocation (design elaboration) and before logic synthesis and technology mapping (design compilation)



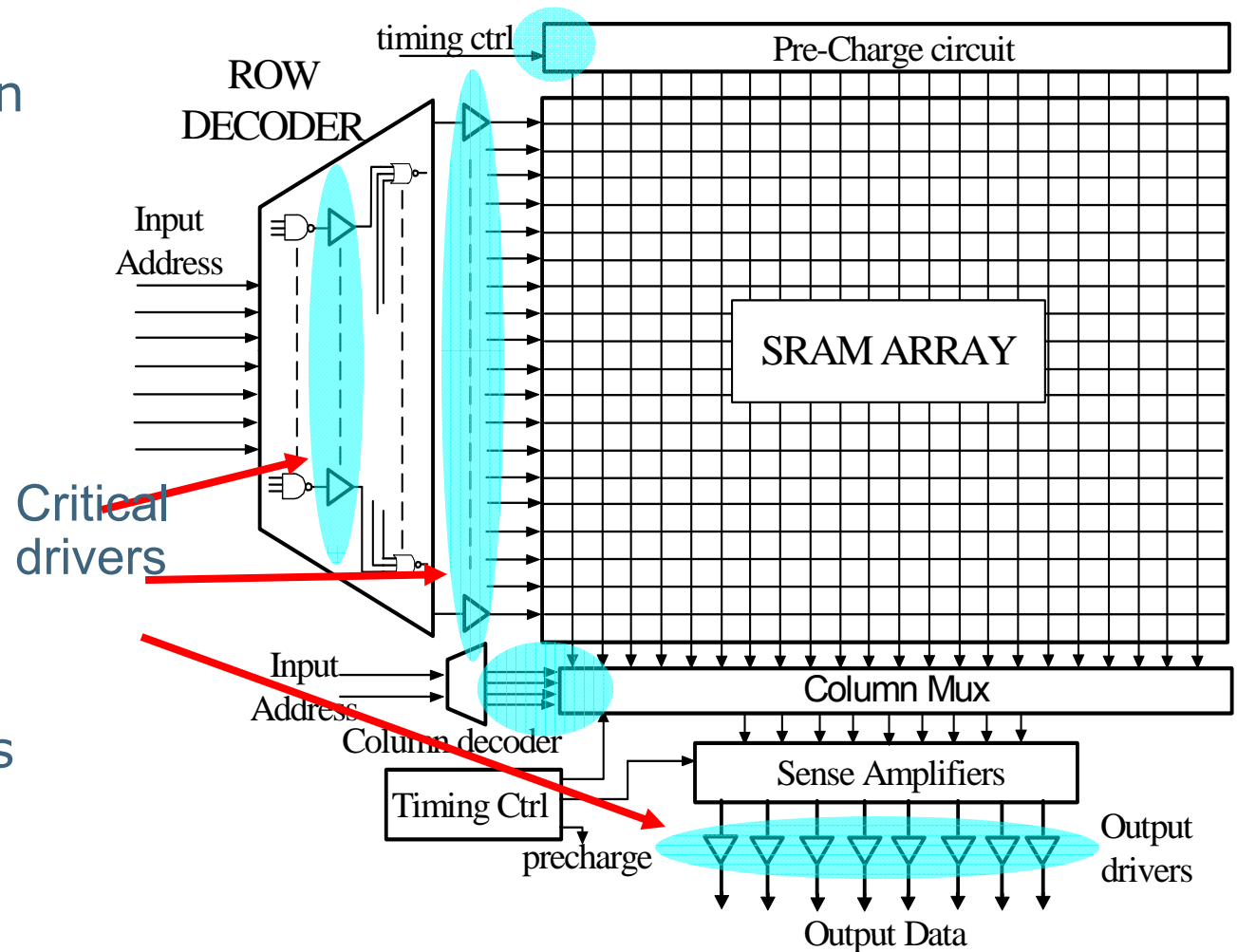
Monitoring the delay of a critical path, actual



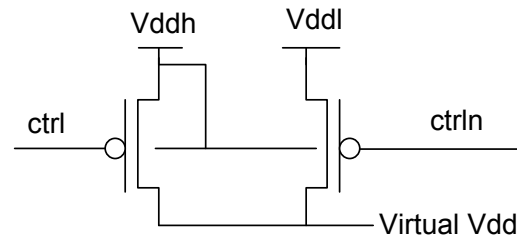
Shadow DFF in the delay monitor fails first, so acts as a *near-failure* warning

Buffer is commonly used and strongly affects memory E/D → good places for knobs

- Buffers/drivers needed/present in different parts of memory architecture
- Limited impact in area but big impact in energy/delay
- Ideal components for cheap Pareto “Config knobs”



Runtime switchable Pareto buffer enables adaptive tuning of the knob with low overhead



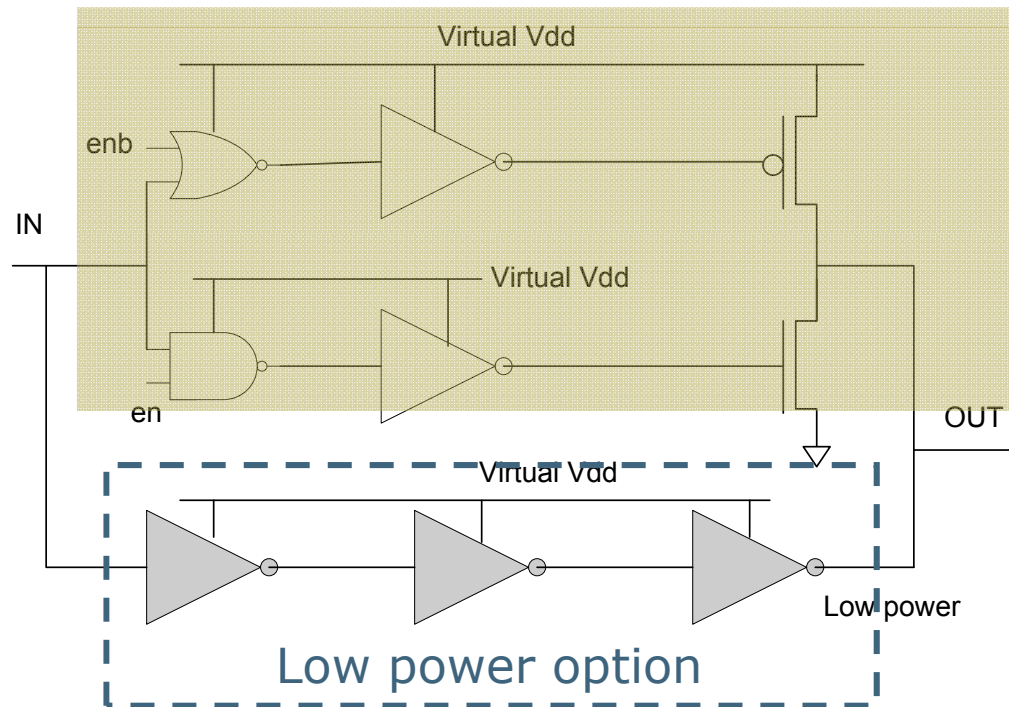
- Up to four options can be included

- Runtime switching not very frequent/depend on tasks

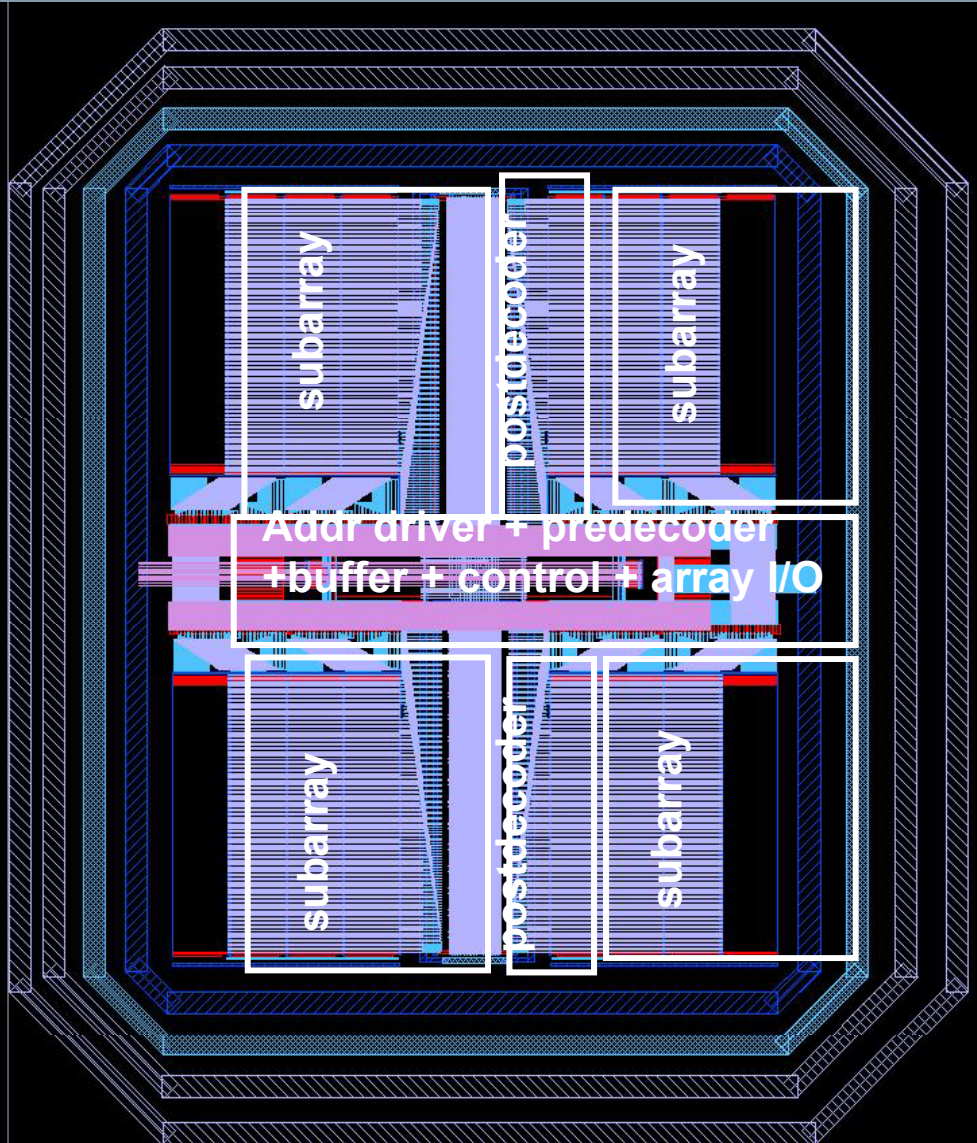
- deployed in IMEC (xDec + WL drivers, etc.)

≡ High speed buffer

High speed option



Memories test vehicle: Configurable drivers in SRAMs (gate size + Vdd knobs)



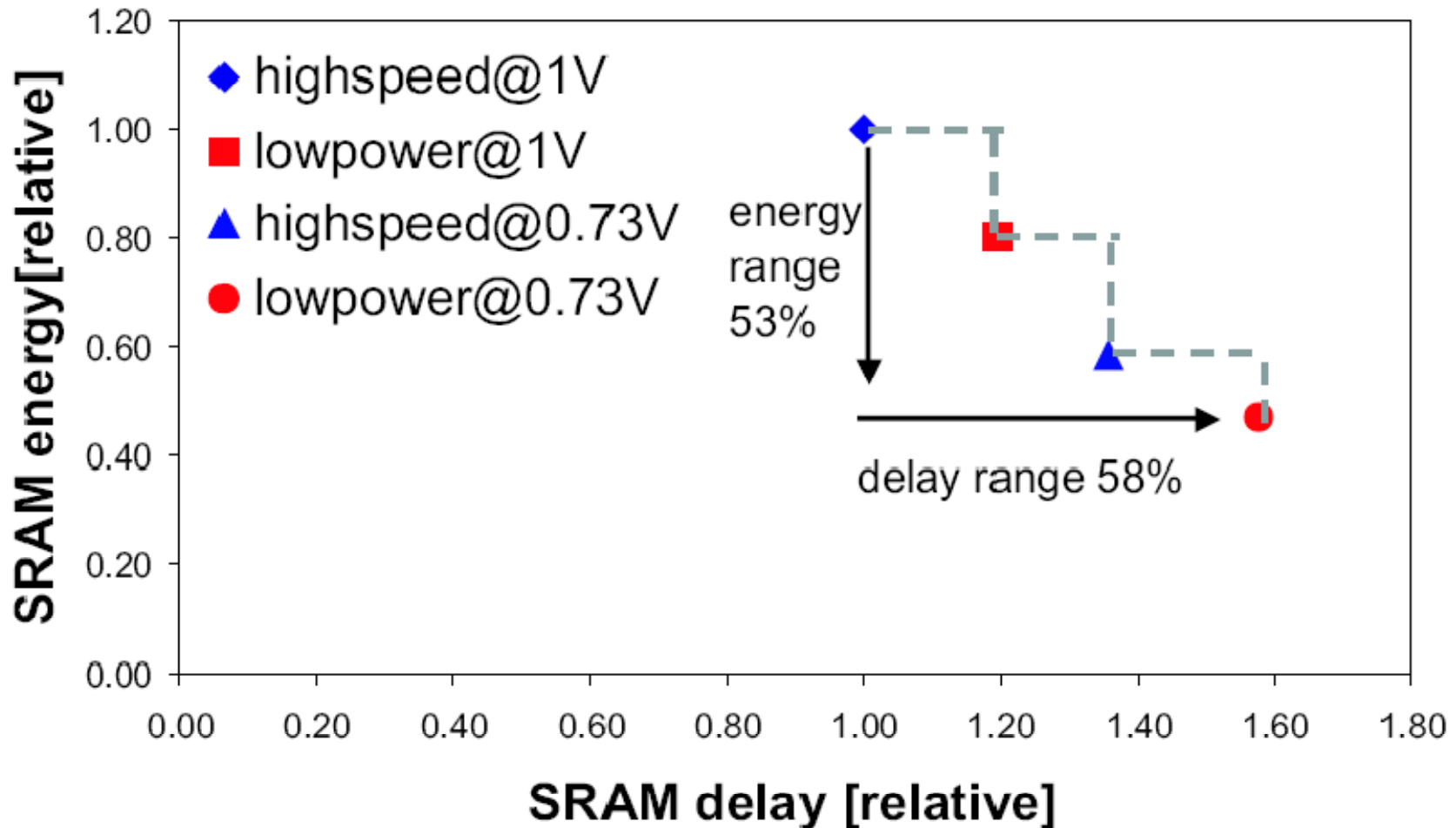
8KB SRAM

- 4 subarrays
- improved control and interface
- 2 voltage islands with runtime switchable buffers

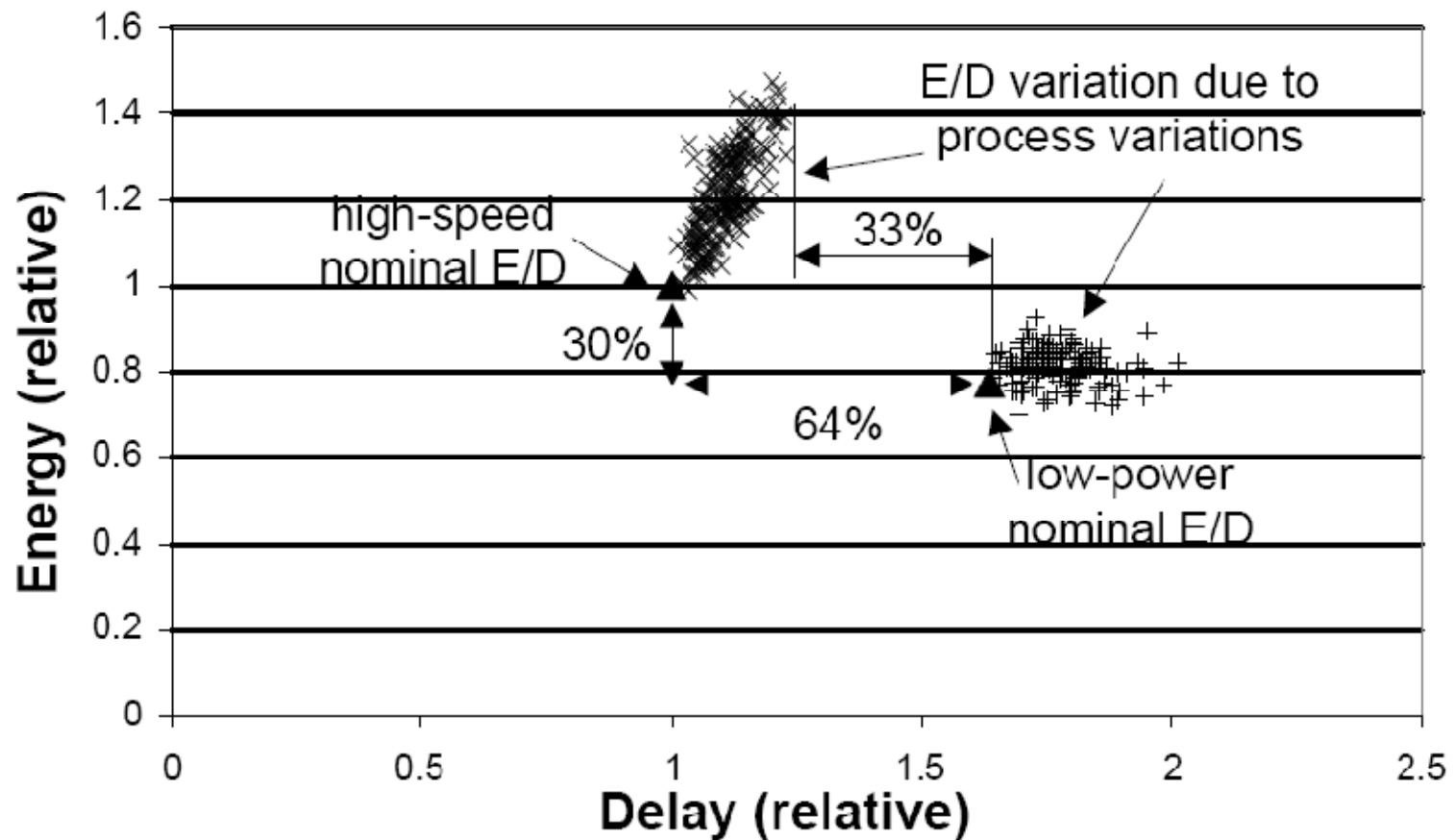
Technology: IMEC
130nm [layout]

Post layout simulation:
65nm BSIM

V_{dd}+ gate size knobs enables large low overhead E/D range in L1 SRAMs for variability compensation



Runtime switchable eSRAM can effectively cope with variability impact



1KB eSRAM with runtime switchable Pareto buffer using gate size knob

Summary

- Analysis to predict yield in power/speed/VDD/°T/time/... domain is feasible. It takes:
 - sophistication of the available variability foundry information
 - Sophistication of variability models
 - Sophistication of flow and tools maintaining correlation and detail
- Solutions to avoid over-pessimism for reliability is possible: It takes:
 - Design for average case and avoid 6-sigma design
 - Add the capability to self-tune the system at run-time to correct from the possible timing violations: allocate slack only when/where necessary

Thank you

aspire invent achieve

