Welcome!

… to the SPP1500 Miniworkshop

“Current Application and Future Evolution of the RAP Model”

January 08, 2016, TU München

10:00 – 10:30	RAP Model Status - Summary on Feedback	Andreas Herkersdorf
10:30 – 10:50	Error Modeling for Memories	Norbert Wehn
11:00 – 11:20	RAP Extensions for addressing Uncertainty Distributions of Components	Faramarz Khosravi
11:30 – 11:50	The Interdependencies in the Scope of RAP	Hussam Amrouch
12:00 – 12:45	Lunch	
12:45 – 13:00	Workgroup planning and assignments	all
13:00 – 14:15	Workgroup meeting(s)	
14:15 – 14:30	Workgroup presentations	all
14:30 – 14:45	Wrap up	Andreas Herkersdorf

Michael Glaß, Ulf Schlichtmann, Norbert Wehn & Andreas Herkersdorf

Questions posted to SPP 1500 partners

- In what form or application do you make use of RAP in your SPP project?
- What are obstacles / inhibitors for RAP applicability in its present form?
- What do you expect from / would you like to see as future RAP development?

SPP1500 Miniworkshop
January 08, 2016, TU München

RAP Model Status
–
Feedback Summary

Andreas Herkersdorf
Technische Universität München

Technische Universität München

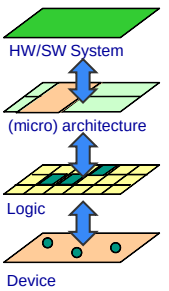
Acknowledgements

The current RAP (Resilience Articulation Point) Model is the result of a collaboration within the DFG SPP 1500 program “Dependable Embedded Systems”

- M. Engel, M. Glaß, J. Henkel, V. Kleeberger, M. Kochte, J. Kuehn, S. Nassif, H. Rauchfuss, W. Rosenstiel, U. Schlichtmann, M. Shafique, T. Schweizer, M. Tahoori, J. Teich, N. Wehn, C. Weis, and H.-J. Wunderlich
- A. Herkersdorf et al., “Resilience Articulation Point (RAP): Cross-layer dependability modeling for nanometer system-on-chip resilience”, Microelectronics Reliability Journal, Elsevier, Volume 54, Issues 6–7, June–July 2014, Pages 1066–1074.
- Veit B. Kleeberger et al.,”A Cross-Layer Technology-Based Study of How Memory Errors Impact System Resilience”, IEEE Micro, 33 (4), July 2013.

DFG SPP1500 „Dependable Embedded Systems“

- Federal priority program of German Research Foundation (DFG)
  - Launched in January 2011, 6 years perspective
  - 11 projects (17 PIs) funded across entire stack of hardware, software, application levels
  - Approx. 9,5 Mio Euro for 6 years
- **Objectives**
  - Investigate new design methods and architectures for dependable, cost-efficient Hardware/Software for Embedded Systems by means of:
    - technology abstraction, self-adaptation, reconfiguration, probabilistic algorithms, effective utilization of redundancies
  - Robust system design with low dependability overhead through cross-layer consideration / optimization

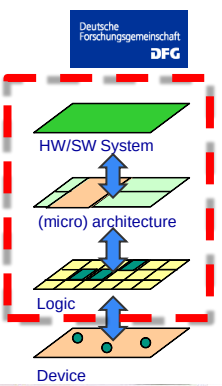


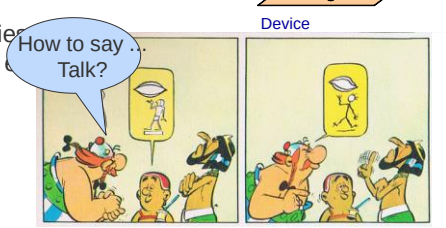
www.SPP1500.itec.kit.edu

Figure source: Joerg Henkel, KIT

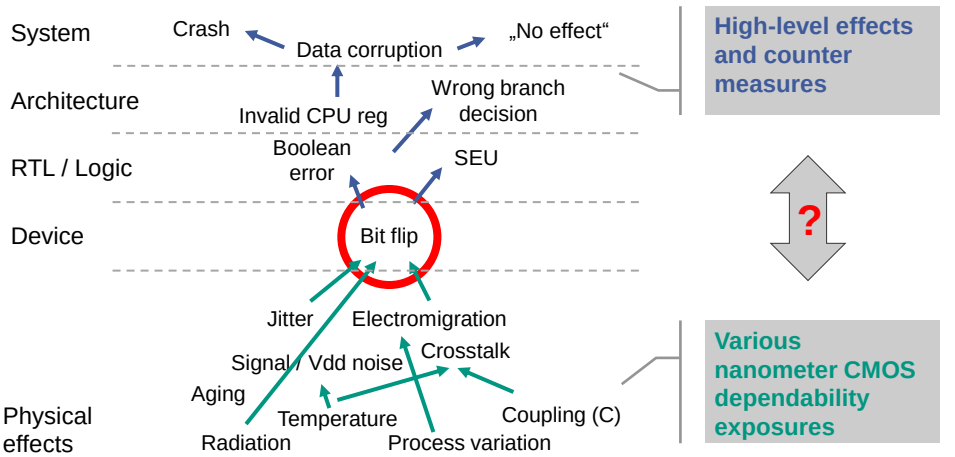
DFG SPP1500 „Dependable Embedded Systems“

- **Cross-layer implies both big potentials and challenges**
  - Potentials
    - Trade-off effectiveness and cost of resilience mechanisms when applied at different levels
    - Exploit diversity in approaches and obtain technology abstraction for higher layers
  - Challenges
    - How to adequately model device layer properties for system level applicability?
    - How to make respective communities with interdisciplinary background talk to each other?

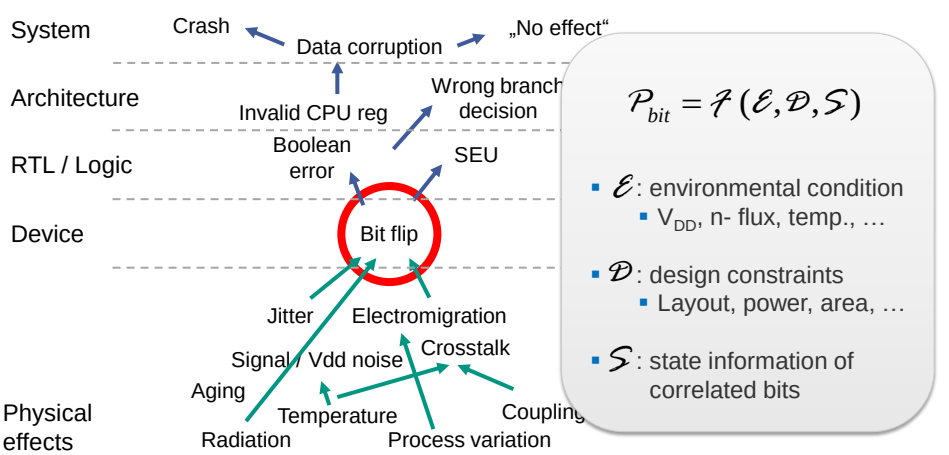




Resilience Articulation Point (RAP) Model

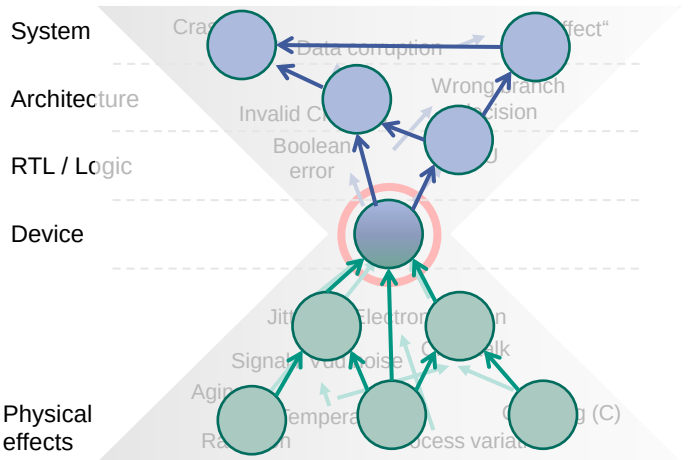


Resilience Articulation Point (RAP) Model

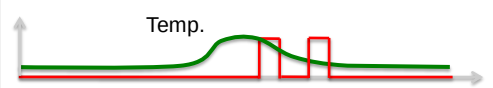
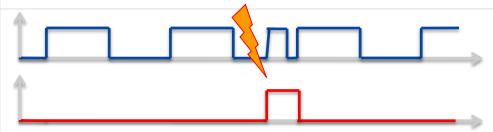


Resilience Articulation Point (RAP) Model

- Bit flip for encapsulating, abstracting and coupling technology with higher levels

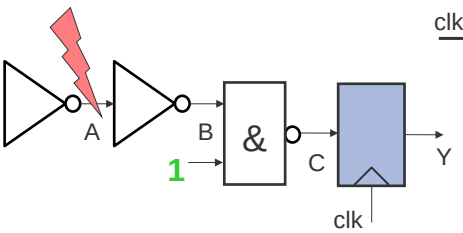


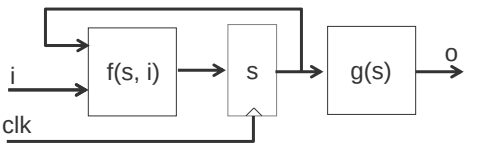
Bit Flip Examples

Typical signal / bit timing	Fault origin / Fault type
	Aging related permanent faults
	Radiation (SET, SEU)
	Temperature hotspots
	SET faults affecting clock network

Soft Errors in Sequential Logic

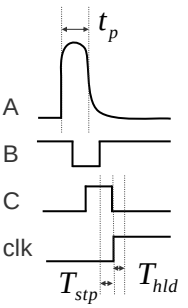
Single Event Transient (SET)





$$P_{bit}(y,t) \approx \frac{T_{stp} + T_{hld} + t_p}{T_{clk}} \cdot P_{sense} \cdot P_{SET}$$

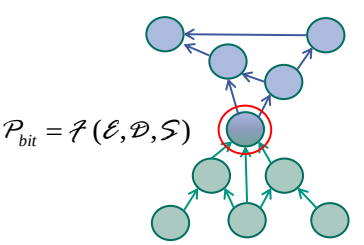
$$t_p = f(E_\alpha, L_{min}, C_L, V_{DD})$$



Error depends on spatial and temporal correlation between fault occurrence and register clocking

Principal RAP Pillars

- Faults from various physical effects, if not masked, will manifest as permanent or transient single or multi-**bit invalidations**
- Probabilistic error functions are used to encapsulate and model different physical effects resulting in bit-flips

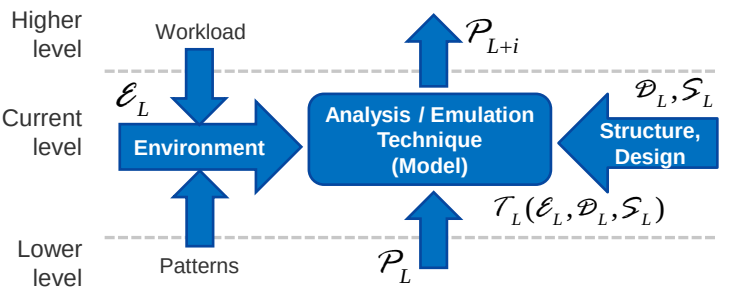


- Transformation functions $\mathcal{T}_L$ convert probabilistic error functions $\mathcal{P}_L$ towards higher abstraction levels

$$\mathcal{P}_{L+i} = \mathcal{T}_L(\mathcal{E}_L, \mathcal{D}_L, \mathcal{S}_L) \circ \mathcal{P}_L$$

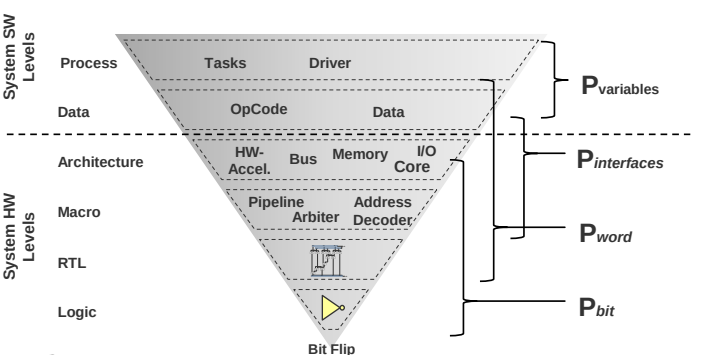
- **Error and transformation functions aren‘t / can‘t be integral parts of the RAP modeling framework!**

Principal RAP Pillars



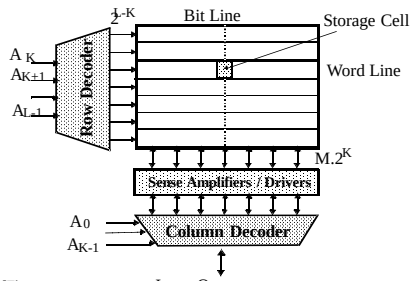
- Each fault type (origin) *j* is expressed as a separate error function
  - $P_{bit}(\vec{x},t) = \sum_j P_{bit}^j(\vec{x},t)$
- Error and transformation functions are abstractions, hence approximate!
- Dynamic program flow considered through workload characteristics in $\mathcal{E}_L$

RAP: The Upper Half of the Hour Glass



- Error model / function $\mathcal{P}_L$ reliefs expert at level *L+i* to know the details below *L*
  - $\mathcal{P}_L$ serves as error injection means, replacing stack of complex lower layer models

From Bit to Word Error(s)



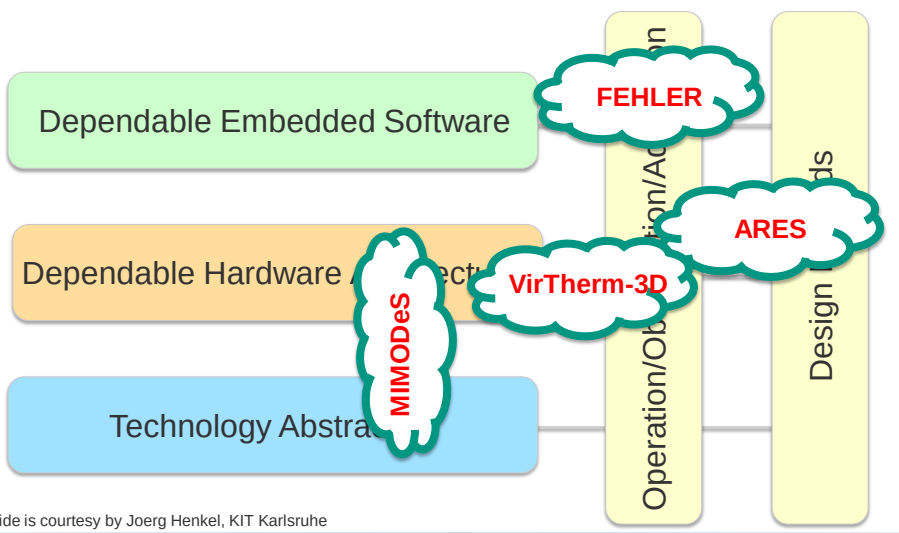
[Figure source: J. Rabaey]

- With assumption of bit error independence

$$P_{word}(\vec{x},t) = \mathrm{T}_{\mathrm{bit}} \circ \mathrm{P}_{\mathrm{bit}}(\vec{\mathrm{x}},\mathrm{t}) = 1 - \prod_{\mathrm{x_i} \in \vec{\mathrm{x}}} (1 - \mathrm{P}_{\mathrm{bit}}(\mathrm{x_i},\mathrm{t}))$$

- With dependencies / correlations among bit errors
  - $\mathcal{P}_{word}$ affected by $\mathcal{S}$ and $\mathcal{D}$

Examples for RAP Adoption in SPP 1500 Projects

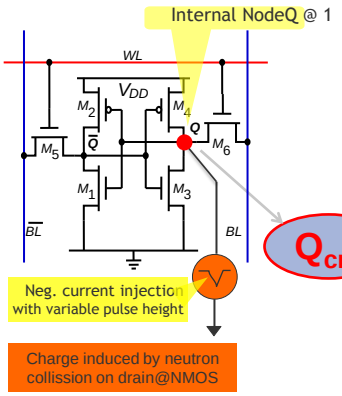


Slide is courtesy by Joerg Henkel, KIT Karlsruhe

SRAM Cell Soft-Error Model

Contributed by N. Wehn (TU Kaiserslautern),
U. Schlichtmann (TU München), S. Nassif (IBM Austin)

SRAM cell@technology



Cosmic ray flux

$$P(N(T)=k)=\exp(-\Phi\cdot A\cdot T)\frac{(\Phi\cdot A\cdot T)^k}{k!}$$

Induced charge distribution

Q_{crit}

$$f_Q(Q_{inj.})=\frac{1}{Q_s}\exp\left(-\frac{Q_{inj.}}{Q_s}\right)$$

Calculation of **Bit flip** probability

$$P_{SEU}(Q>Q_{crit}\mid NodeQ=1)=\int_{Qcrit}^{\infty} f_Q(Q)dQ$$

Physics

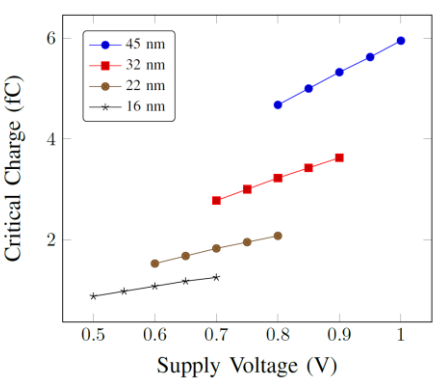
Technology

Memory cell level

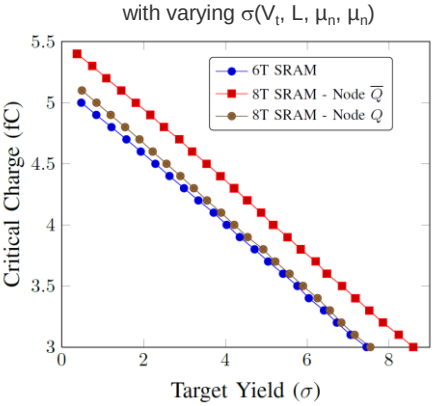
Q_{crit} Dependencies

Contributed by N. Wehn (TU Kaiserslautern),
U. Schlichtmann (TU München), S. Nassif (IBM Austin)

Varying V_{DD}



Architecture dependencies @ 32nm
with varying $\sigma(V_t, L, \mu_n, \mu_n)$

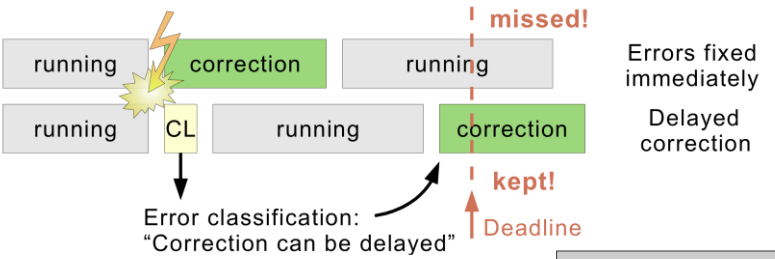


High-performance bulk technologies

Reliable Embedded Software

Contributed by M. Engel, P. Marwedel (TU Dortmund)

- **Error correction under resource/timing constraints**
  - Requires flexible conflict resolution
  - Separate error detection & classification from correction



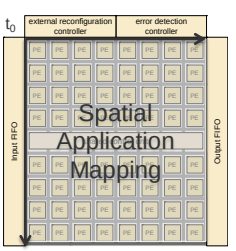
  - Error impact analysis done during compile time based on source code
  - Reliable / unreliable marked variables treated differently during error detection at run time

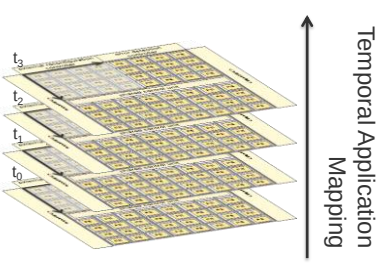
```
//
unreliable int u;
reliable   int v;
```


Dynamic Functional Verification (DFV)

Contributed by W. Rosenstiel (Tübingen University)

- Coarse Grained Reconfigurable Architectures (CGRA)
- Allow mapping of applications into the temporal as well as the spatial domain

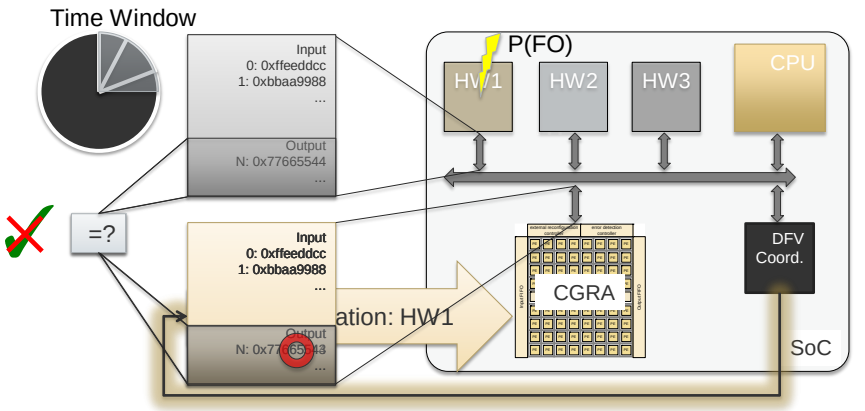




- CGRA "Slow-Down Factor" *s* describes degree of temporal domain usage (see figure above, left s=1, right *s=4*)

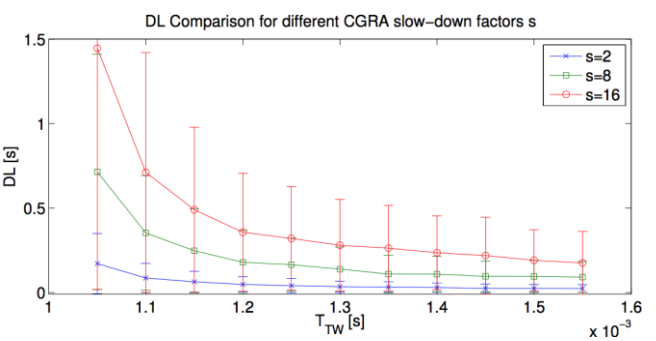
Dynamic Functional Verification (DFV)

Contributed by W. Rosenstiel
(Tübingen University)



Dynamic Functional Verification (DFV)

Contributed by W. Rosenstiel
(Tübingen University)



Exemplary Estimation of Detection Latency *DL* based on P(FO)=10^{-5}
and clock frequency F=100Mhz

Possible workgroup subjects

- Joerg Henkel:
  - We should be able to release RAP models to the public (e.g. opensource) such that RAP cannot only be understood but actually used by other researchers
  - There should be a transaction paper like TCAD where we describe RAP in a reproducible manner such that other researcher can understand, use and probably even extend it

- Is it feasible / realistic to include within the RAP model itself concrete, technology & application specific transformations from lowest to highest abstraction levels?
  - Provision a contiguous RAP model through all layers that all SPP 1500 projects can use
  - Obtain bit error probabilities for different types of gates, SRAM cells, latches, so that we can have a complete bottom-up instance of the RAP hourglass
  - Missing aspect is a showcase across all layers of abstraction

Thanks!