Architecture and Design Methodology for Autonomic Systems-on-Chip (ASoC)

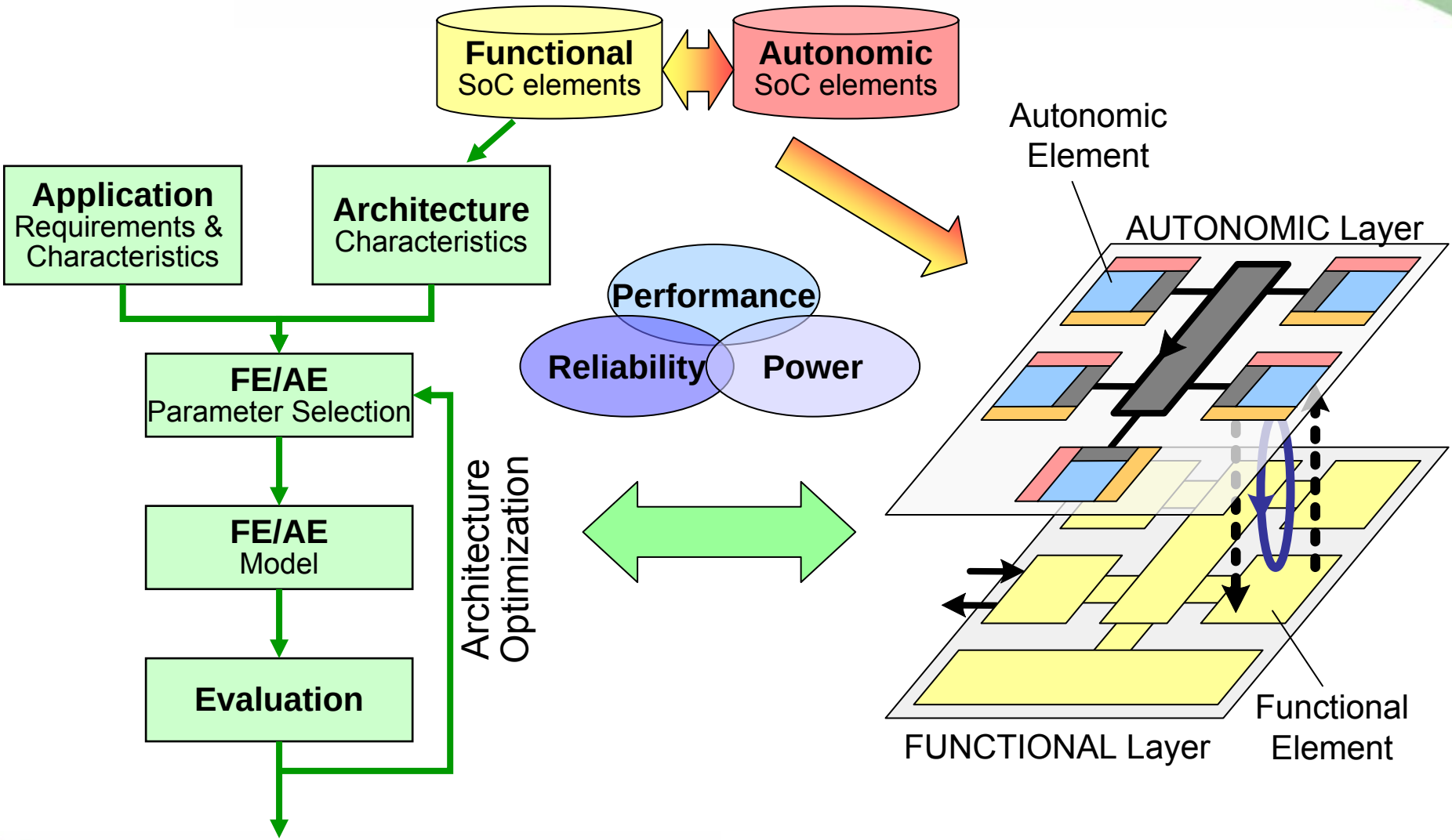
A. Bouajila, A. Bernauer, J. Zeppenfeld, W. Stechele, O. Bringmann,
A. Herkersdorf, W. Rosenstiel

Universität Tübingen

Technische Universität München

FZI - Forschungszentrum Informatik

Project Reminder

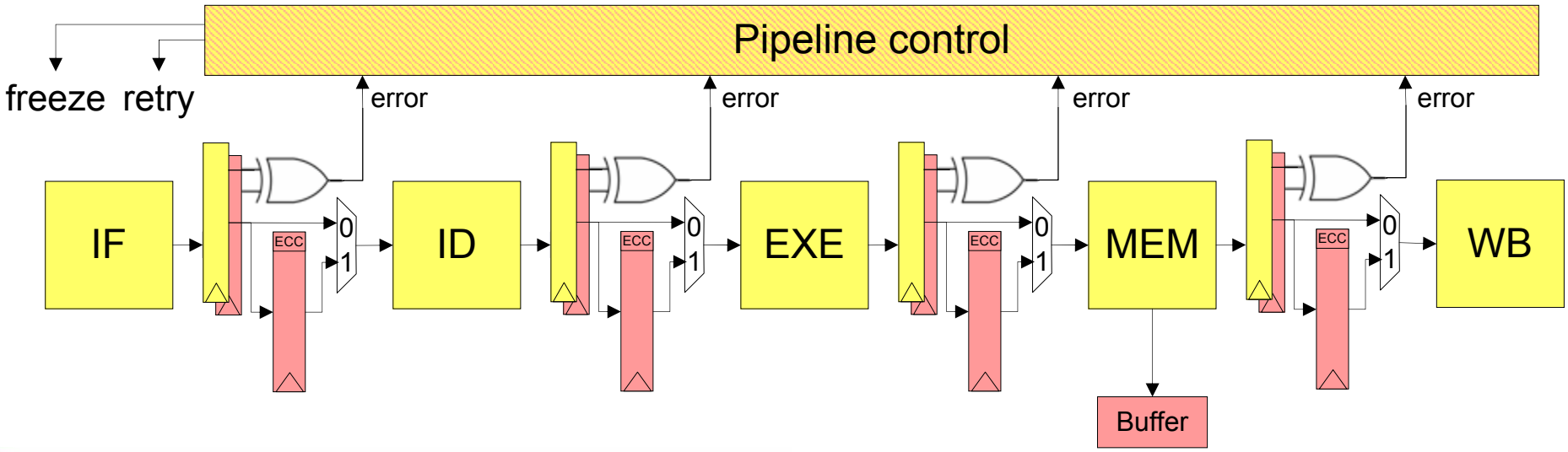


Overview

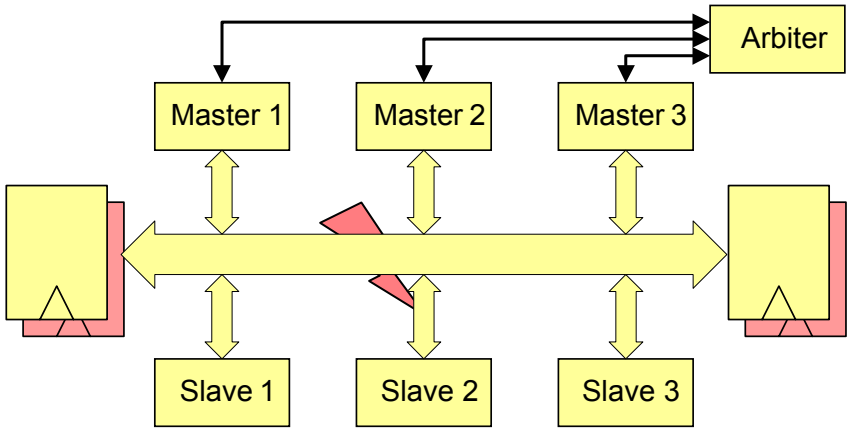
- 2nd year Phase 1 results
  - Self-healing CPU pipeline
  - Self-healing Bus
  - Reliability estimation
- Planned work for Phase 2
  - Monitors for balanced Power Performance Reliability
  - HW/SW Implementation concept for an SoC LCS
  - Autonomic Element Interconnect
  - ASoC simulator and prototype
- Summary & Cooperations

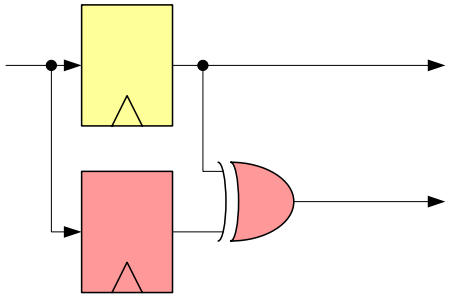
Self-healing CPU Pipeline [VLSI06]

- Error detection using Nicolaidis shadow registers
- History registers keep track of latest pipeline stage registers
- No pipeline flushing necessary → fixed 2-cycle penalty
- Implemented in Leon2 processor (23% area overhead in Virtex II Pro)



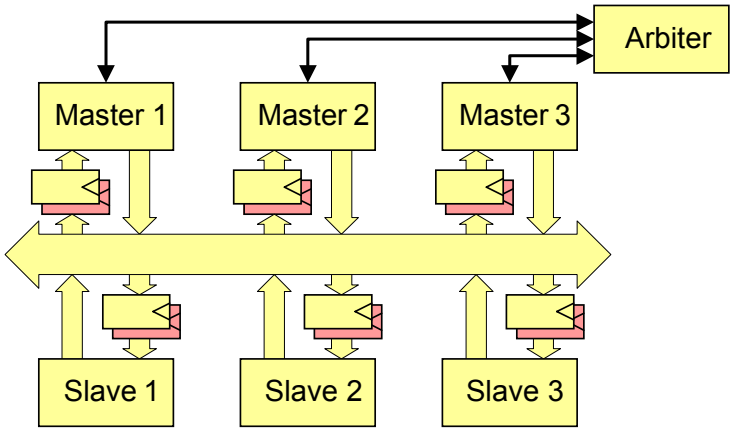
Extension for Bus Protection

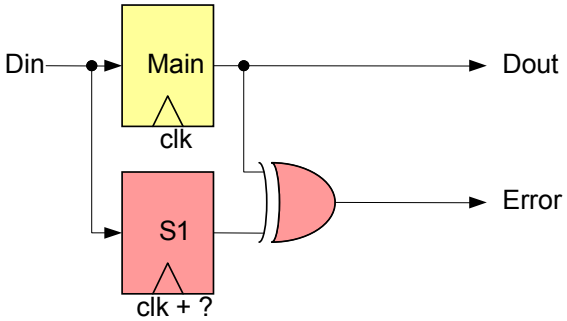




- Detection with shadow register technique
  - Where to insert?
    - Bus ends

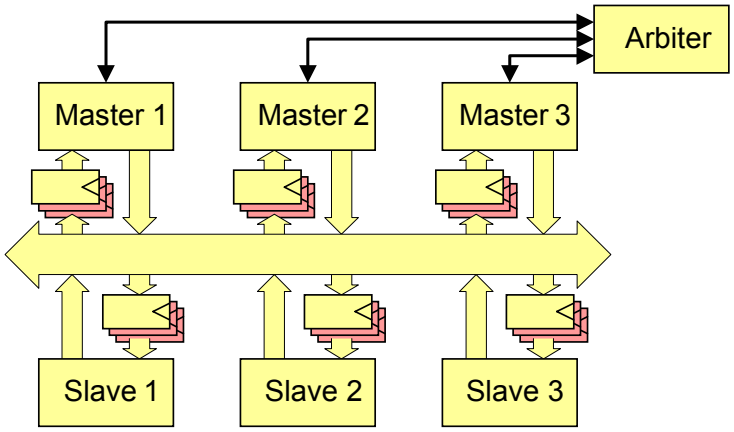
Extension for Bus Protection

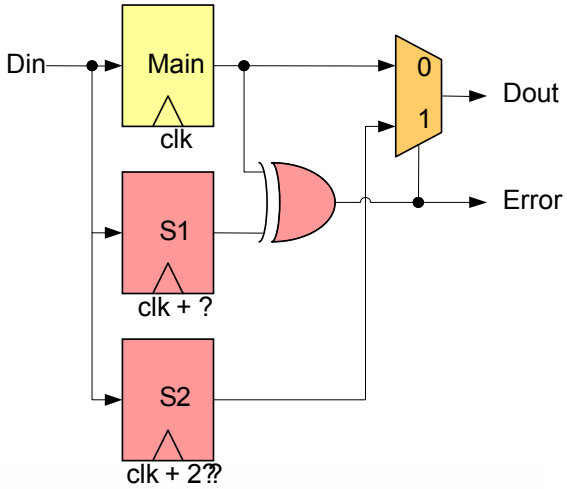




- Detection with shadow register technique
  - Where to insert?
    - Bus ends
    - Primary inputs

Extension for Bus Protection





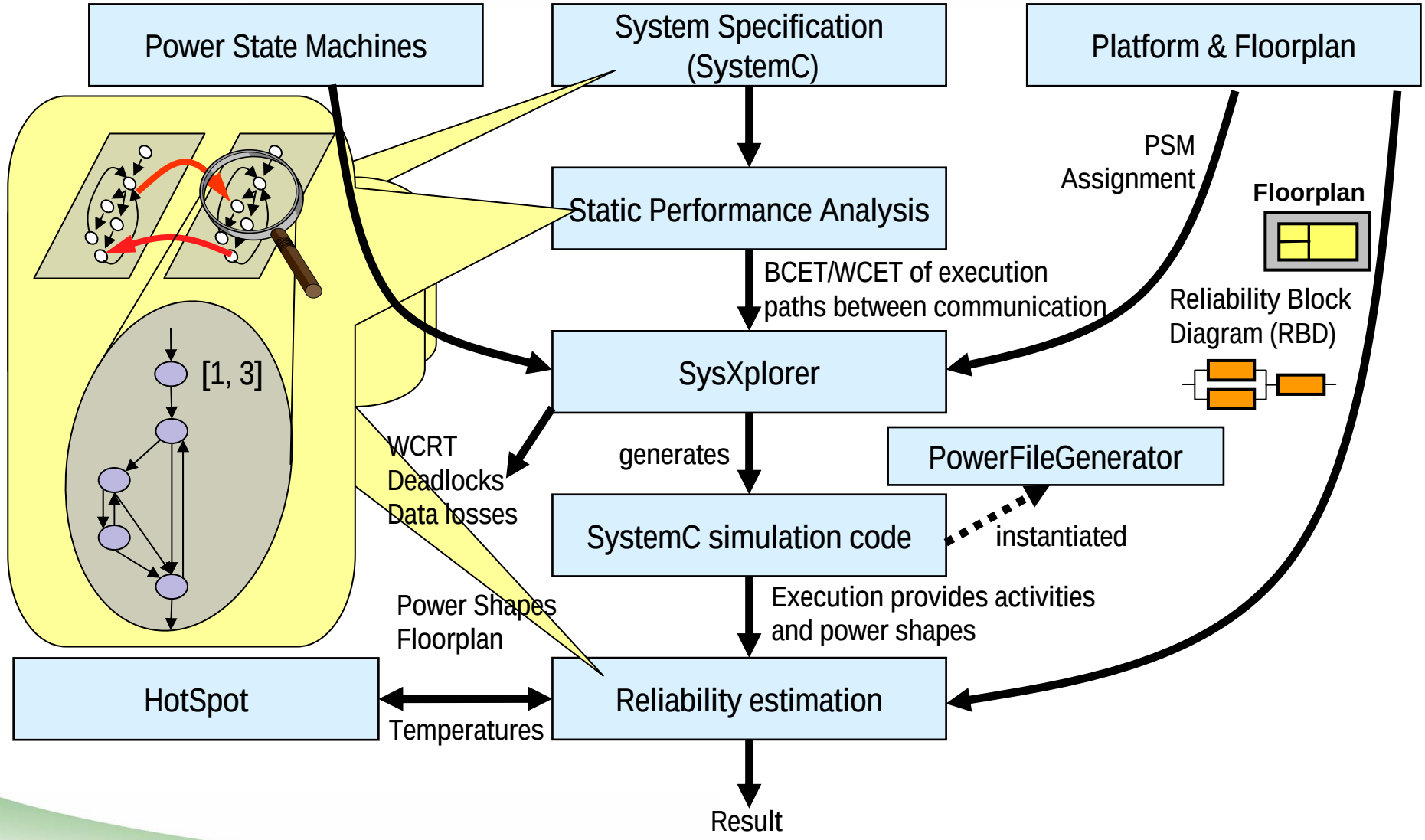
- Detection with shadow register technique
  - Where to insert?
    - Bus ends
    - Primary inputs
- Correction via triplication
  - Avoid retransmission
  - Rollback difficult since bus is not simple pipeline
  - Triplication of registers only
  - Similar area overhead and fault coverage as history registers

Bus Protection Prototype

- Implemented for the Processor Local Bus on a Xilinx Virtex II Pro FPGA
- Pseudorandom fault insertion using an LFSR
- All single errors are properly detected and corrected
- Correction occurs in bus interface, transparently to IP

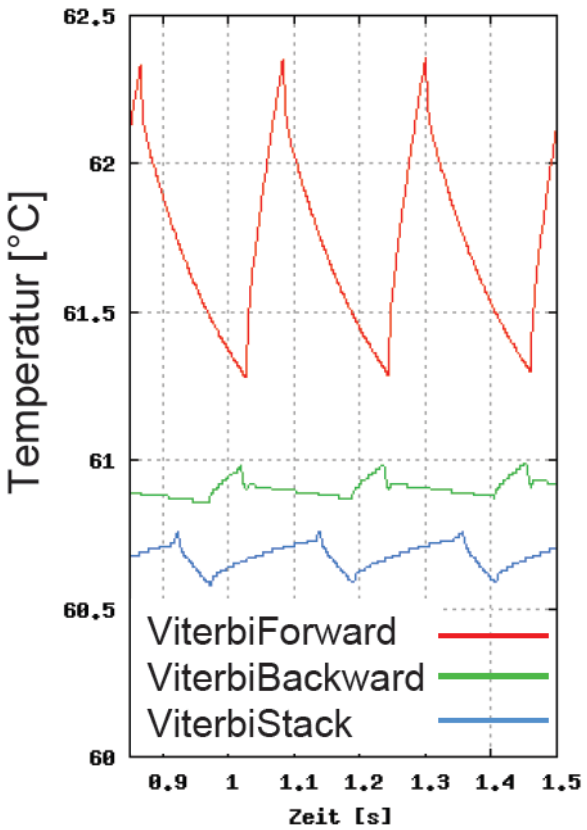
	No Protection	Triplication for control signals	ECC for data signals
DCMs	1	3	1
Slices	695	1017 (46%)	876 (26%)
Slice FFs	647	1159 (79%)	685 (6%)
4-LUTs	994	1251 (26%)	1255 (26%)

Application-dependent Reliability estimation

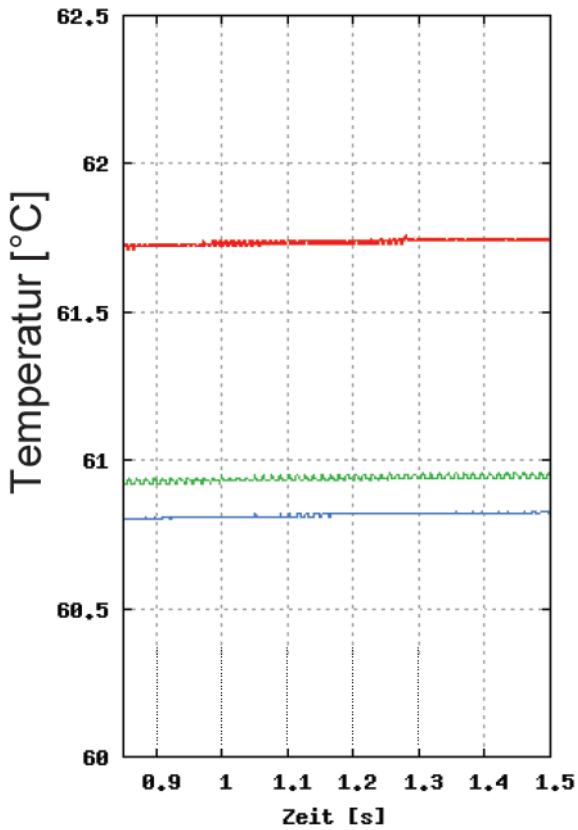


Experimental Results

with Power-Mgt



without Power-Mgt



Arrhenius relationship

$$\mathrm{MTTF} = A \cdot J^{-n} \cdot e^{\frac{E_a}{k \cdot T}}$$

Power management **reduces** "hot temperature errors" by 2%

Coffin-Manson relationship

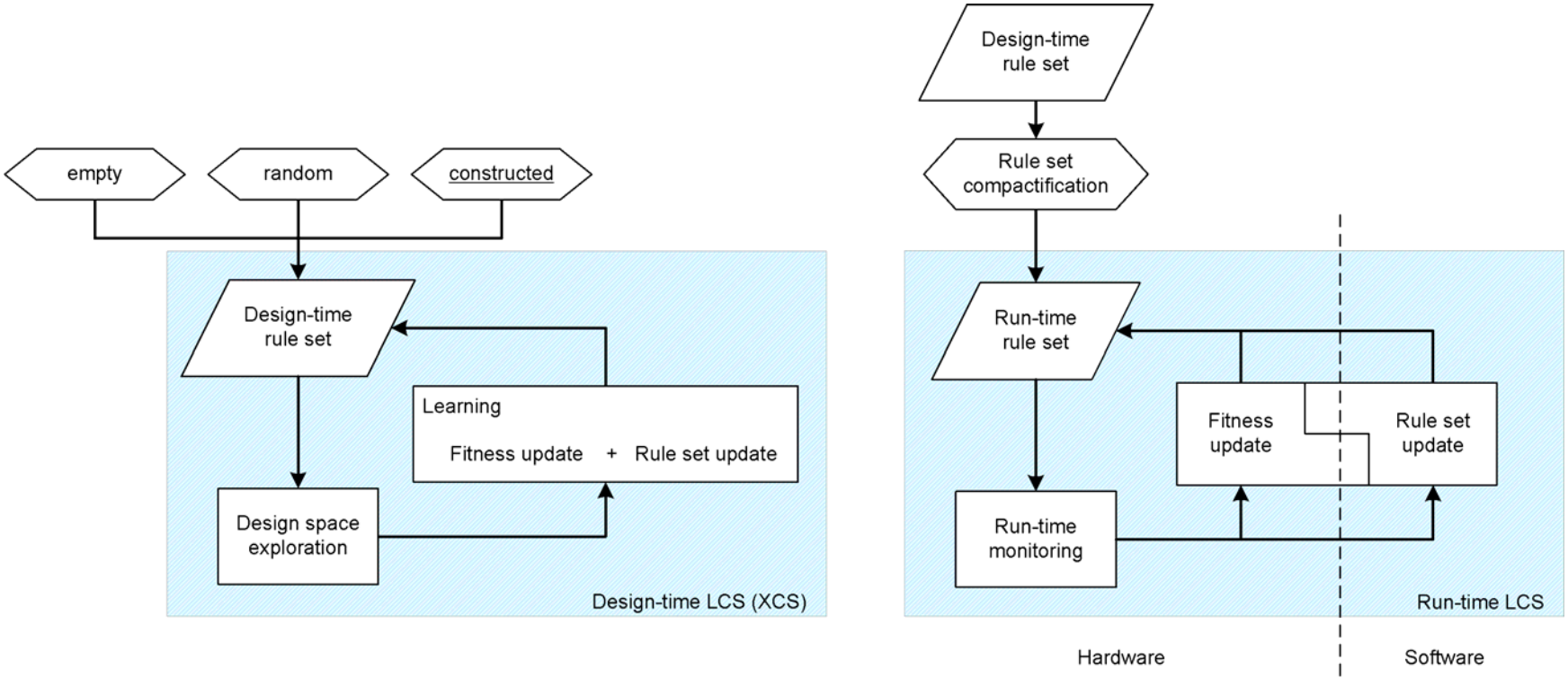
$$N = B \cdot (\Delta T)^{-q}$$

Power management **enlarges** „thermal cycling errors" by 40x

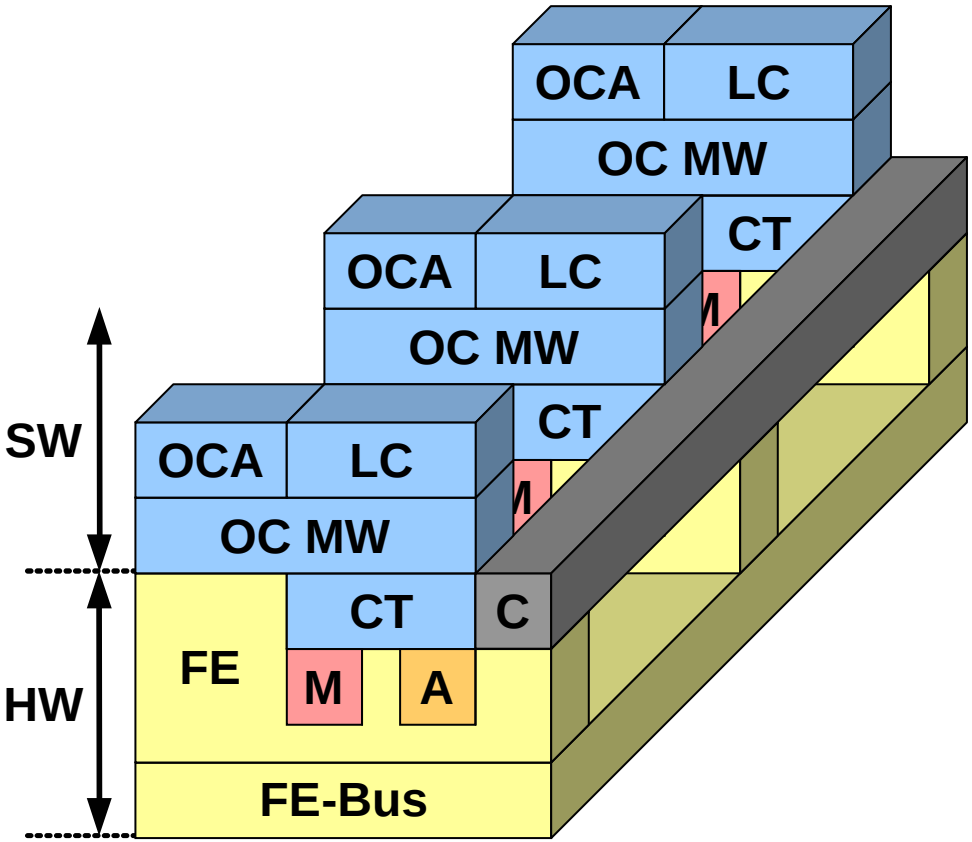
- Allows application-dependent reliability comparison of different design alternatives

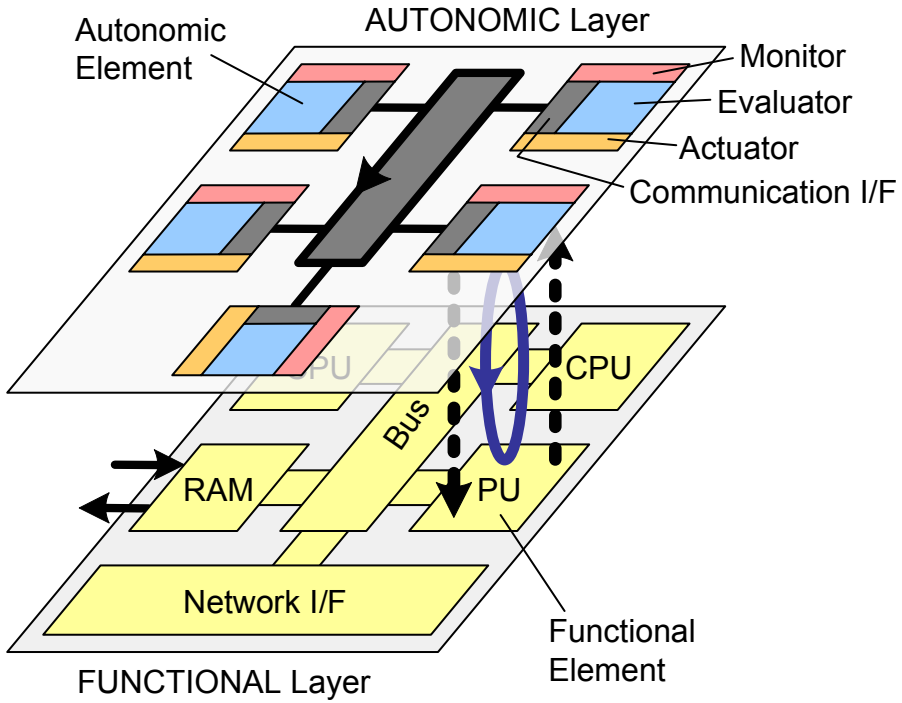
Goals for phase 2

- Two parts in learning: design time and run time
- Design time: learning initial (general) rule set
- Run time: adapting rule set to individual chip

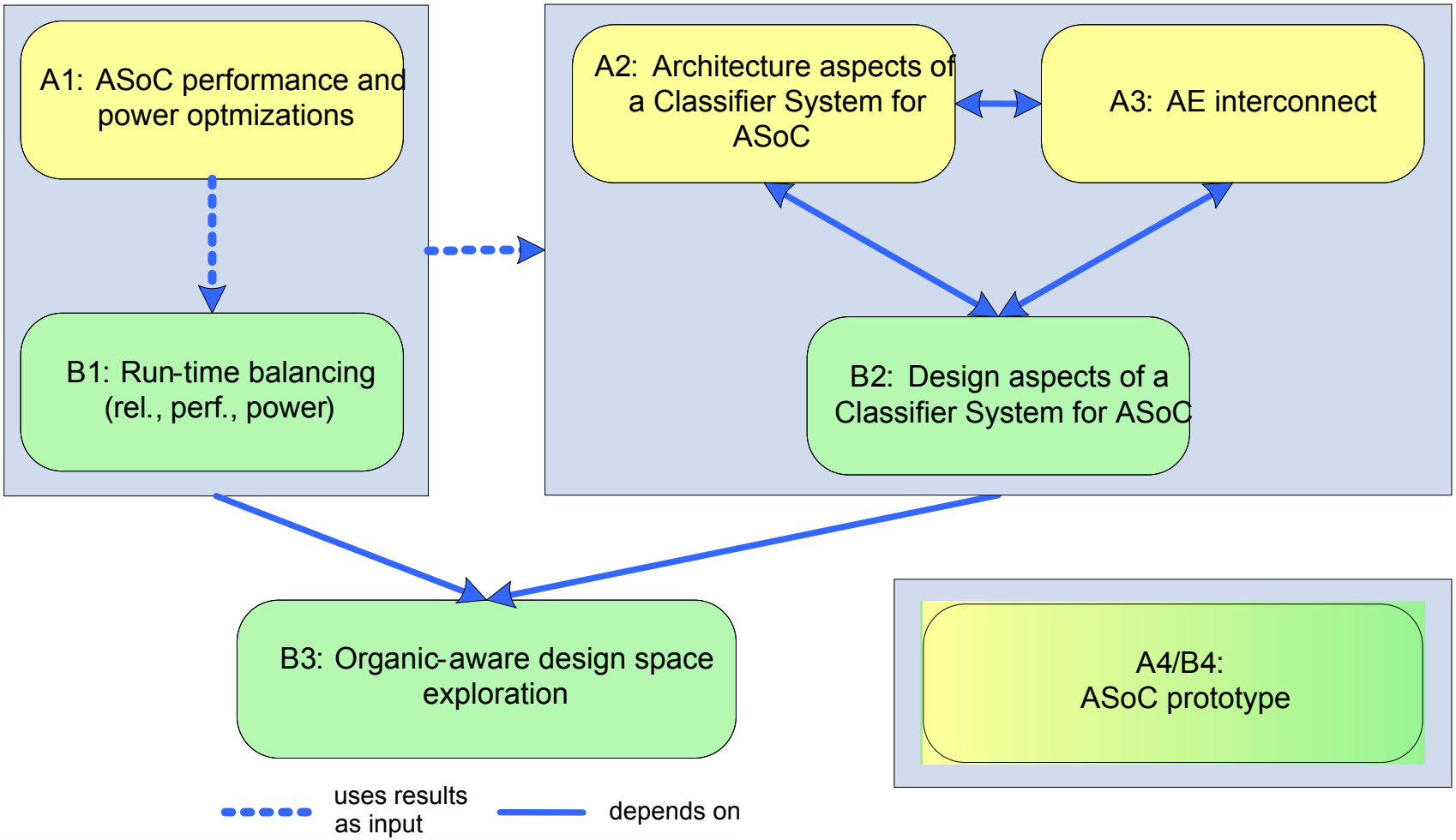


Goals for phase 2



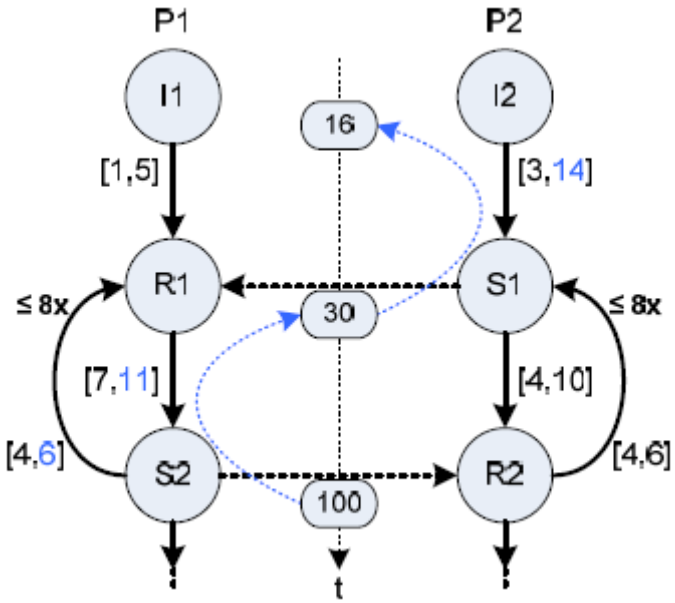


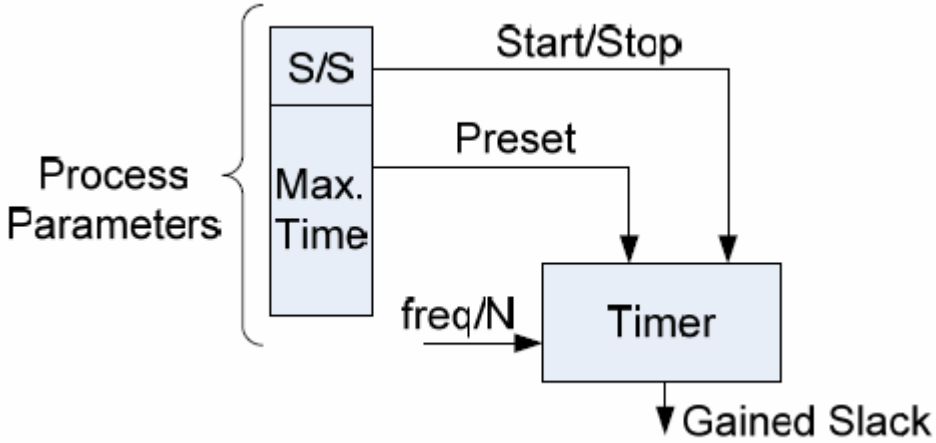
Phase 2 Work Packages



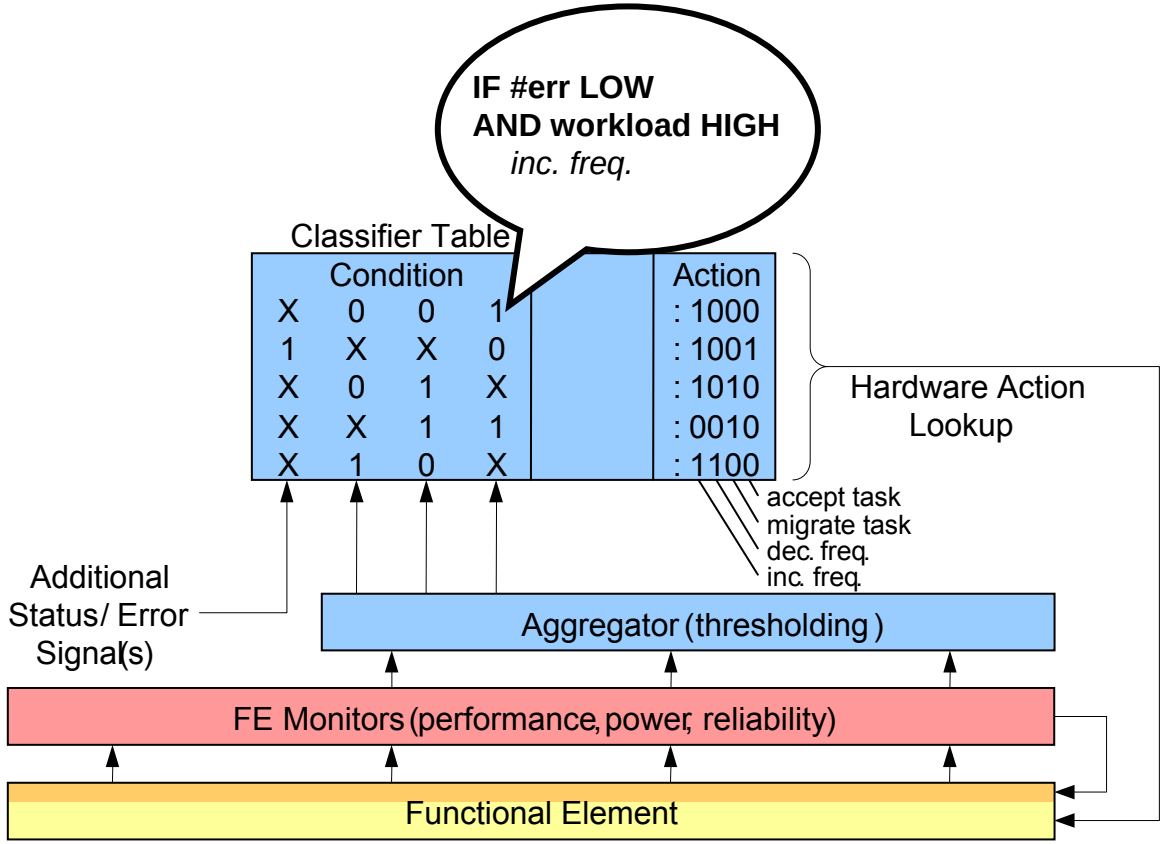
A1/B1: Performance monitors

- Identification of suitable performance monitors
- Evaluation of the benefits and feasibility of such monitors
  - Examples: CPU (idle loops), Bus (busy signal)..
- Run-time SW tasks monitoring (check point insertions)

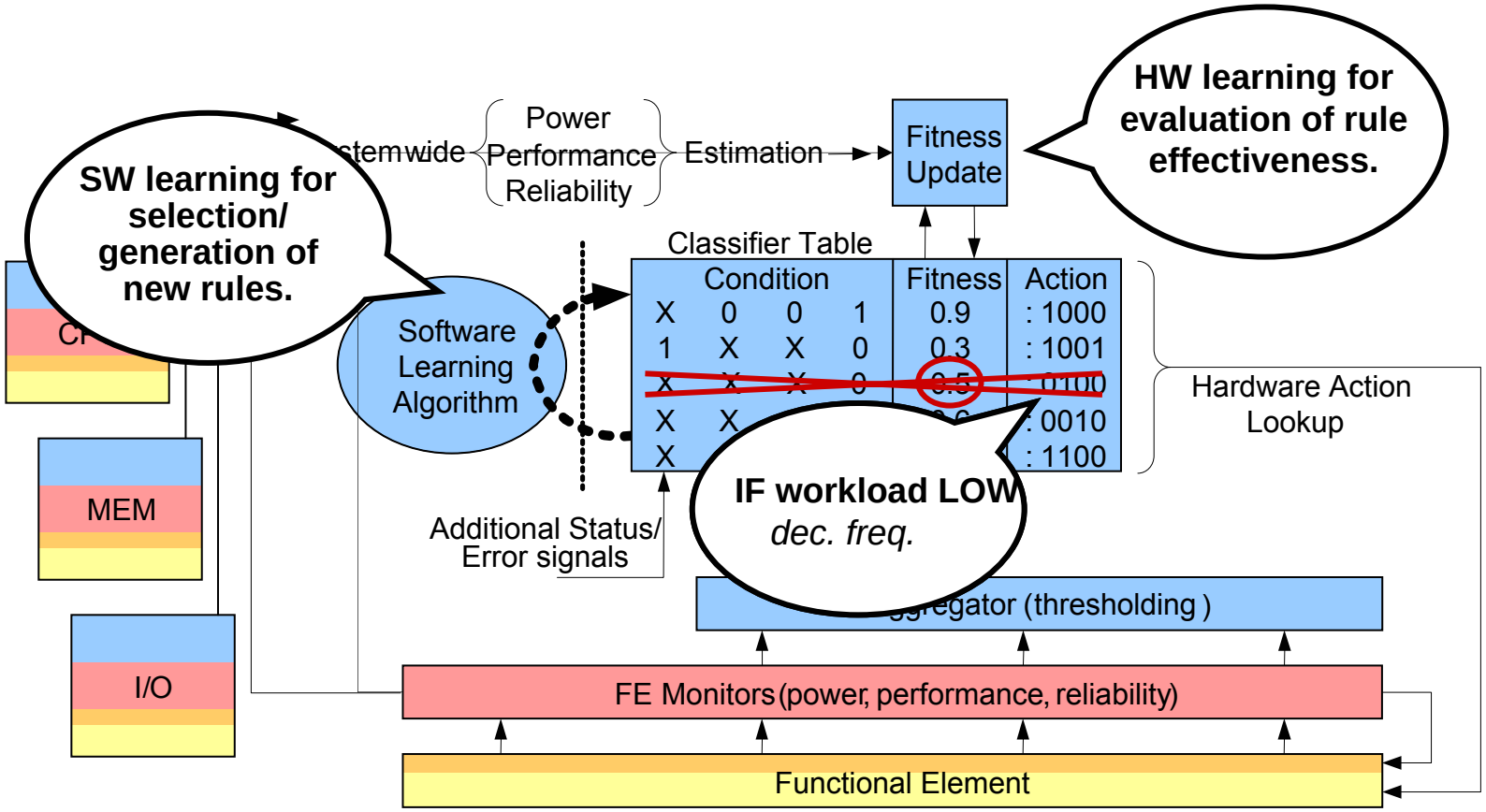




A2/B2 Classifier Table



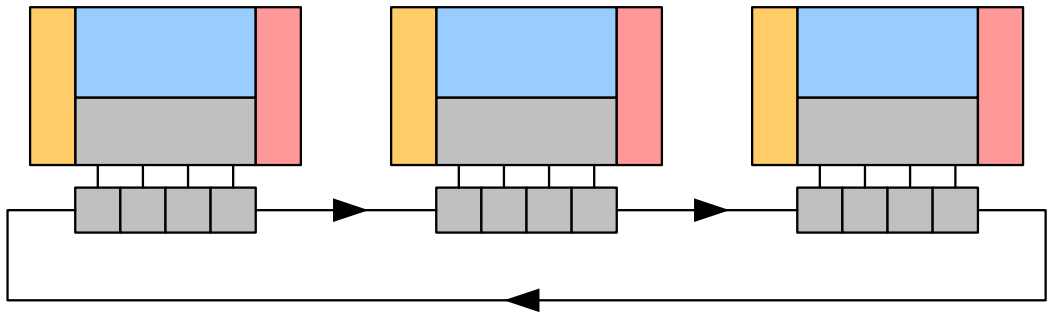
A2/B2: Classifier Table – On-line Learning



- Learning Classifier Systems: [Holland78, Wilson95, Butz06]

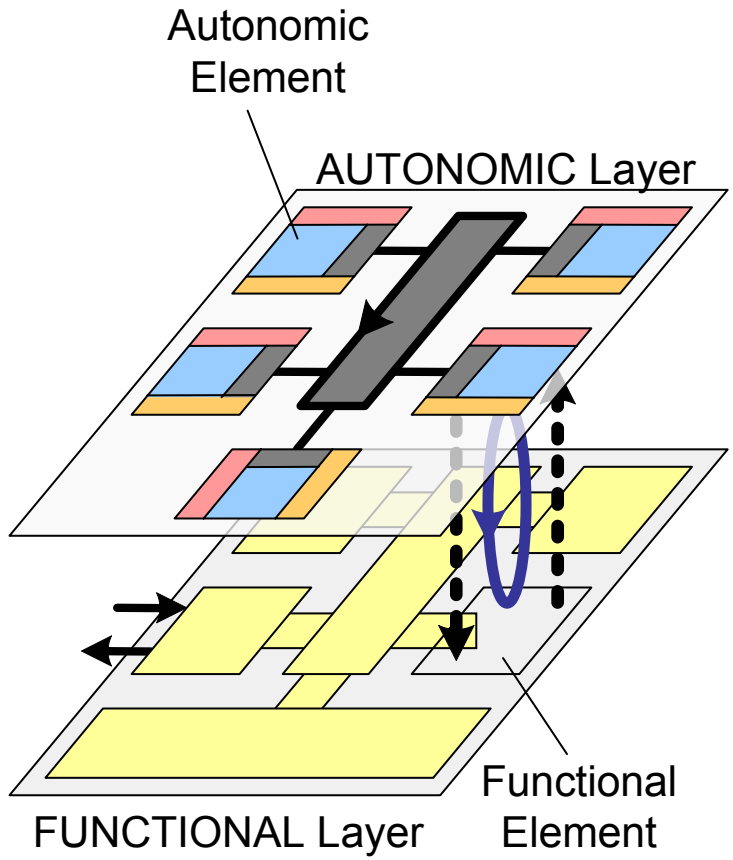
A3: Autonomic element interconnect

- Autonomic and functional interconnects: two different mediums (otherwise complexity problem, interference avoidance)
- Structure of the autonomic interconnect: serial ring looks adequate
- Interconnect should be:
  - Scalable: run-time addition/removal of an AEs
  - Reliable: ECC, retransmission, dual rail..

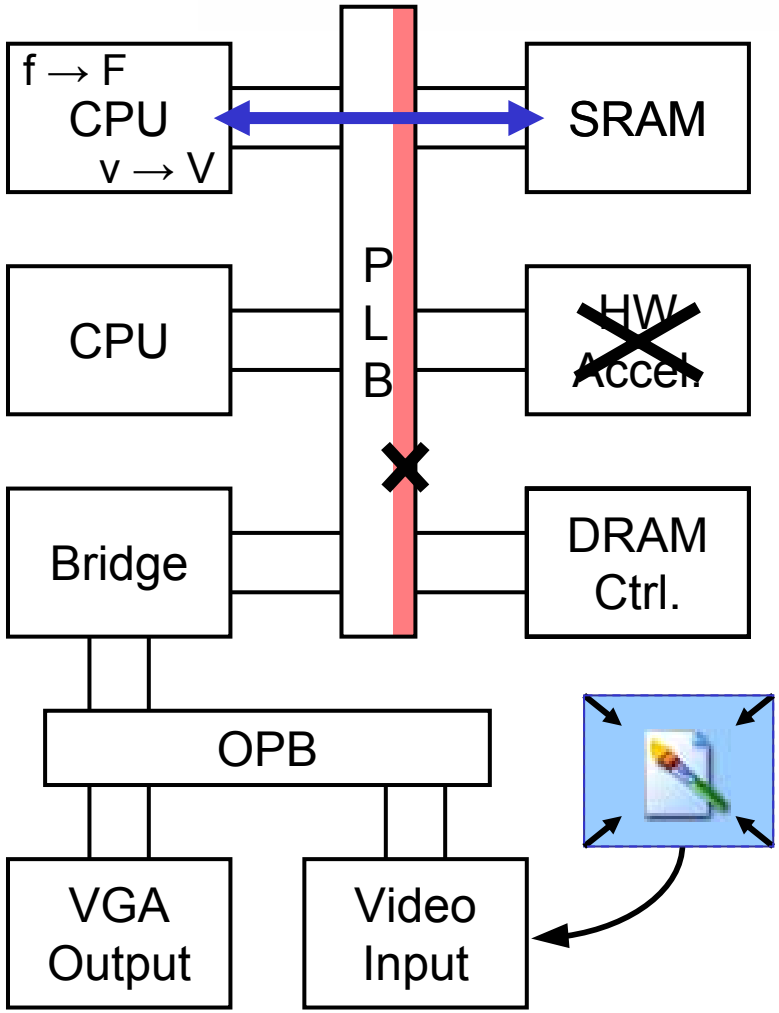


With a 32-bit word register per node:

- Average latency: 11 µs
- Data bandwidth : 1.6 Mb/s
- 24414 Classifier Table updates/s



A4/B4: ASoC prototype



- Dealing with Defects
  - AEs detect and disable faulty:
    - System Components
    - Interconnect Resources
- Performance and power optimization
  - (De-)Activate redundant IPs
  - Scale voltage / frequency
  - Adjust amount of data
    - Image size
    - Frame rate
  - Scale bus capacity
  - Prioritize transfers

A4/B4: ASoC prototype

- SystemC based simulator
  - Simulate initial LCS rule-set
  - Validate the HW/SW LCS implementation
  - Preliminary system evaluation and validation
- FPGA based demonstrator:
  - IPs will be augmented with reliability and performance optimizations
    - at least three Leon3 CPUs
    - Bus interconnect
    - Memory controller
    - Mac
    - Different AEs connected with AE interconnect

Summary

- Balanced Power Performance Reliability optimizations
- Integrating design-time/ run-time learning capabilities to SoC
  - HW/SW implementation of an LCS
  - Distributed AEs
- Validate new ASoC concepts with SystemC simulator and FPGA demonstrator

Cooperations

- Team of Prof. Reif
  - Verification of self-x properties based on logic model
  - Tools for reliability estimations
- Team of Prof. Fey
  - Marching pixels as an application to reliable design
- Teams of Prof. Müller-Schloer/ Prof. Schmeck
  - LCS concept and implementation
- Team of Prof. Maehle/Prof. Brockmann
  - Cooperation on HW/SW Learning
- Team of Prof. Ernst
  - Performance analysis

- DodOrg team in Karlsruhe (Prof. Becker, Prof. Karl, Prof. Brinkschulte, Prof. Henkel)
  - Monitoring, processing elements, middleware

Publications

- [BICC06] A. Bouajila, A. Bernauer, A. Herkersdorf, W. Rosenstiel, O. Bringmann,and W. Stechele. *Error Detection Techniques Applicable in an Architecture Framework and Design Methodology for Autonomic SoC*. In IFIP International Federation for Information Processing, Biologically Inspired Cooperative Computing. August 2006.
- [VLSI06] A. Bouajila, J. Zeppenfeld, W. Stechele, A. Herkersdorf, A. Bernauer,O. Bringmann, and W. Rosenstiel. *Organic Computing at the System-on-Chip Level*. In VLSI-SoC, October 2006. Invited paper.
- [OC06] A. Bernauer, O. Bringmann, W. Rosenstiel, A. Bouajila, W. Stechele, andA. Herkersdorf. *An Architecture for Runtime Evaluation of SoC Reliability*. In Organic Computing Workshop, October 2006,.
- [Lipsa05a] G. Lipsa, A. Herkersdorf, W. Rosenstiel, O. Bringmann, W. Stechele. *Towards a Framework and a Design Methodology for Autonomic SoC*, 2nd IEEE International Conference on Autonomic Computing (ICAC-05), 13-16 June, Seattle, USA
- [Lipsa05b] G. Lipsa, A. Herkersdorf, W. Rosenstiel, O. Bringmann, Walter Stechele. *Towards a Framework and a Design Methodology for Autonomic SoC* , Proceedings Dynamically Reconfigurable Systems, Self-Organization and Emergence, Architecture of Computing Systems (ARCS) 2005, pages 101-108.
- [ZuD] W. Stechele et al., *Autonomic MPSoCs for Reliable Systems.* In: Zuverlässigkeit und Entwurf, GMM-Fachbericht, VDE Verlag, Munich 2007.
- [DSD] A. Lankes et al., *Power estimation of Variant SoCs with TAPES.* In: Euromicro- DSD 2007.
- [edaworkshop] W. Stechele, *Concepts for Autonomic Integrated Systems. I*in: edaWorkshop, Hannover, 2007.
- [Dag06a] A. Herkersdorf, *Relevance of Organic Computing for System on Chip Architectures.* In: International Conference and Research Center for Computer Science, Schloss Dagstuhl, Seminar 06031 “Organic Computing – Controlled Emergence”, January 15-20, 2006.
- [Dag06b] O. Bringmann, *Autonomic A System Level Design Methodology for Organic Systems-on-Chip.* In: International Conference and Research Center for Computer Science, Schloss Dagstuhl, Seminar 06031 “Organic Computing – Controlled Emergence”, January 15-20, 2006.

Thanks for your attention