Learning Methodology for Autonomic System on Chip (ASoC)

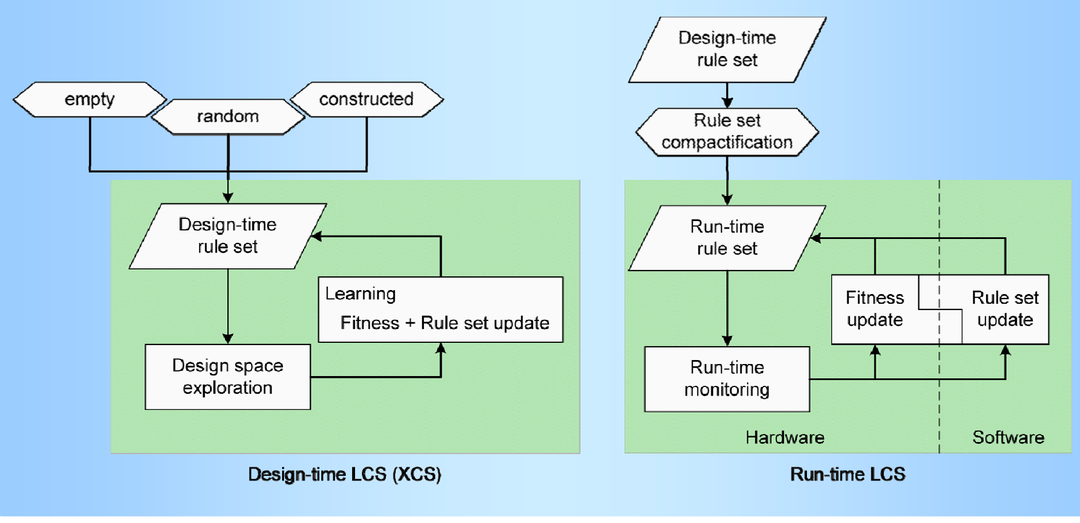
A. Bernauer, W. Rosenstiel
Wilhelm-Schickard Institute for Informatics
Computer Engineering
University of Tübingen

O. Bringmann, B. Sander
Forschungszentrum Informatik
Karlsruhe

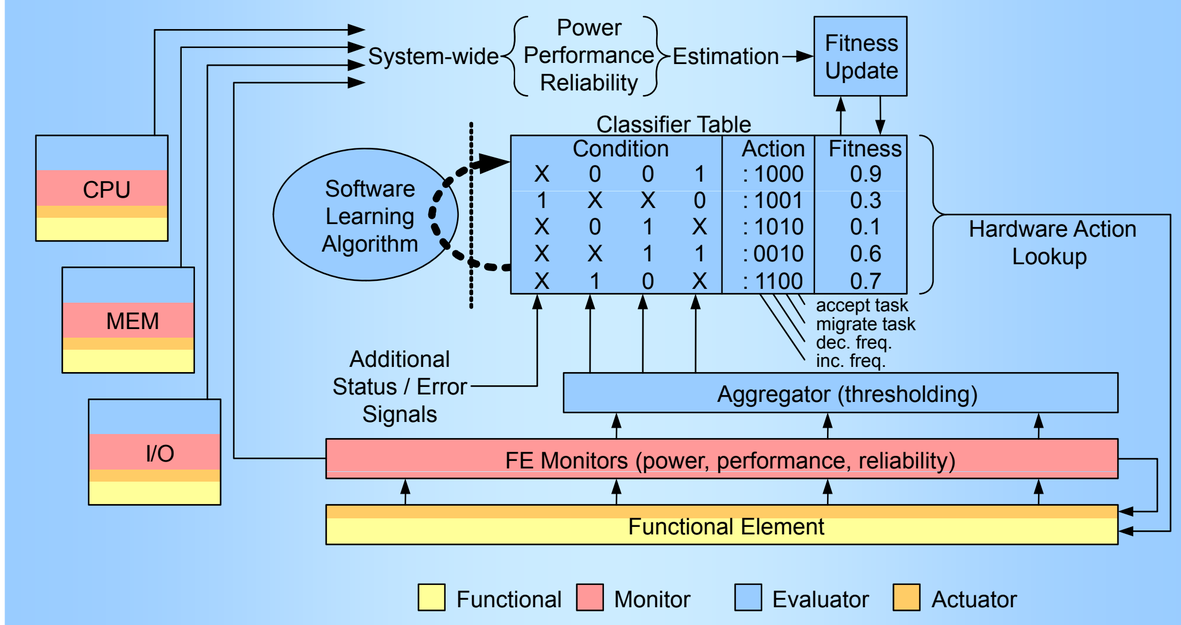
A. Bouajila, J. Zeppenfeld, W. Stechele, A. Herkersdorf
Institute for Integrated Systems
Munich University of Technology

asoc@informatik.uni-tuebingen.de

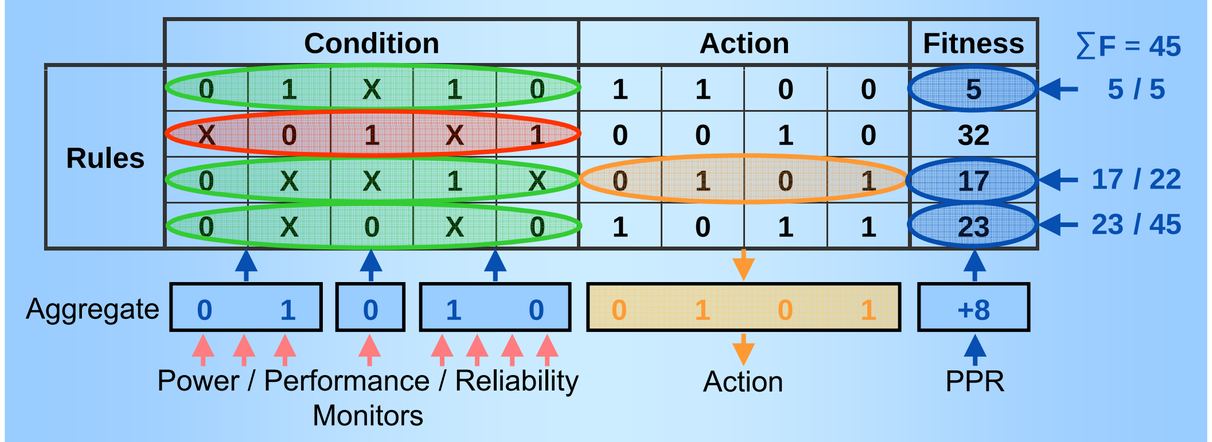
LCS: Design-time and Run-time Learning & Hardware-/Software Partitioning



Run-time learning with a Learning Classifier Table (LCT)



Hardware Implementation of LCT

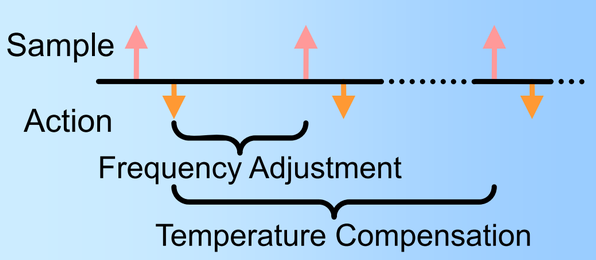



```
1:  ΣF = 0
2:  LOOP over all rules
3:    IF rule matches
4:      ΣF += rule fitness
5:      IF Random % ΣF < rule fitness
6:        act = rule action
7:    END IF
8:  END LOOP
9:  Perform act
```

- Preliminary synthesis results:
  (Xilinx Virtex-II Pro FPGA)
  - 38 Slices
  - 45 Flip-flops / 58 LUTs
  - 1 BRAM, 1 Multiplier
  - Period: 7.620ns
  - Frequency: 131 MHz

Hardware LCT Timing

- Not all actions have the same response delay
- Use timers to set update delay of different rules

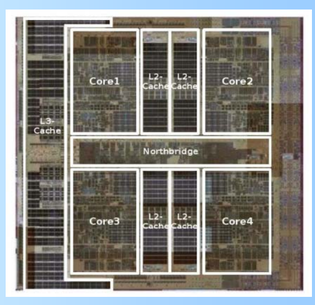


Simulator *ASoCsim*

- Modular simulator for autonomic systems using SystemC
- Low-level run-time simulation
- High-level design-space exploration
- Transaction level modeling
- Contains models5 for
  - power (static and dynamic)
  - soft-errors (Zhu)
  - temperature (HotSpot)
- Easily extendable

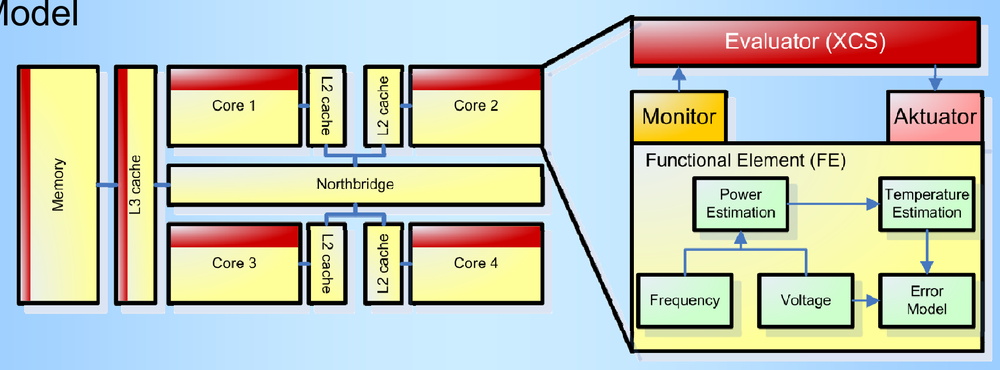
Evaluation of XCS on a SoC

Can XCS find optimal operation points?



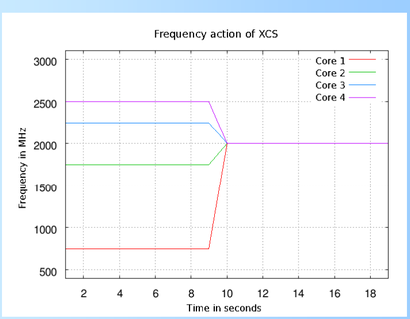
Simulation with ASoCsim

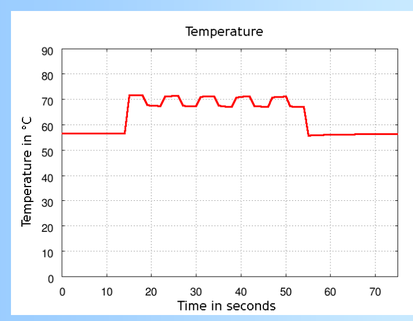
- Applications running on QuadCore (AMD Opteron Barcelona)
- XCS changes frequency, voltage (freely)
- Reward function $R(f,p,t,v) = w_1 \cdot \frac{f}{f_{max}} + w_2 \cdot \left(1 - \frac{p}{p_{max}}\right) + w_3 \cdot rel(t,v,f)$
- Model



Result

Yes, XCS finds *optimal* operation points under various conditions.

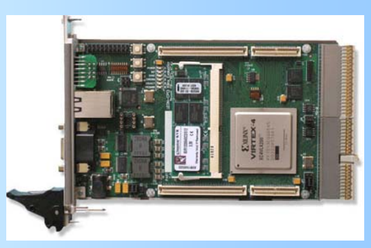




Future work

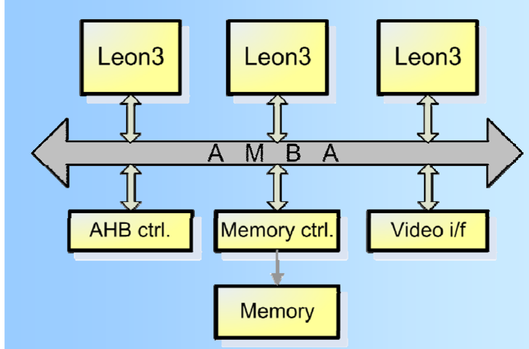
- Distributed XCS
- Software learning interface

Demonstrator



Hardware

- Virtex4 (XC4VLX100 with 49152 slices)
- 256 MB SDRAM
- Ports: JTAG, RS232
- Extra: up to 3 mezzanine I/O cards, SRAM memory
- GRMON Debug tool



Architecture

- 3 self-correcting Leon3
  - Handle timing and transient errors
  - Learn with integrated LCT
- Self-correcting memory controller
- AMBA bus
- I/O
- Error insertion

This work is partially funded by Deutsche Forschungsgemeinschaft (DFG) within the priority program "Organic Computing" (SPP 1183) under the grants HE 4584/3-2 and RO 1030/14-2.