Self-Awareness at the Hardware/Software Interface

Marco Platzner, University of Paderborn
Overview

• The hardware/software interface
  – reconfigurable hardware
  – computing elements design space

• The EPiCS approach
  – self-awareness at the hardware/software interface
  – hardware/software multithreading and heterogeneous multi-cores
Classic View on Hardware and Software

software
- high-level language
- assembly language
- operating system
- instruction set architecture

hardware
- micro architecture
- logic
- transistors
- geometry
## Soft Hardware

<table>
<thead>
<tr>
<th>high-level language</th>
<th>assembly language</th>
<th>operating system</th>
<th>instruction set architecture</th>
<th>micro architecture</th>
<th>logic</th>
<th>transistors</th>
<th>geometry</th>
</tr>
</thead>
</table>

The diagram illustrates the relationship between software and hardware components, highlighting the reconfigurable hardware at the interface. Each level from high-level language to geometry represents a layer in the software/hardware stack.
Reconfigurable Hardware Devices

- Programmable logic blocks and programmable interconnect
  - fine-grained (bit-oriented)
    - Field-programmable Gate Arrays (FPGAs)
    - lookup tables, flip-flops
  - coarse-grained (word-oriented)
    - ALUs, functional units
Computing Element Design Space

flexibility

programmable

general-purpose processor
domain-specific processor
application-specific processor

reconfigurable hardware
application-specific hardware (ASIC)

specialization

fixed function

performance
Overview

• The hardware/software interface
  – reconfigurable hardware
  – computing elements design space

• The EPiCS approach
  – self-awareness at the hardware/software interface
  – hardware/software multithreading and heterogeneous multi-cores
The EPiCS Approach (www.epics-project.eu)

- EPiCS studies **proprioceptive** computing systems
  - proprioceptive = self-aware + self-expressive
  - self-aware: learn and maintain knowledge about internal state & environment
  - self-expressive: determine actions based on goals, values, constraints

- EPiCS looks at several levels of systems
  - compute node level (**hw/sw interface**)
  - network level
  - application level
Proprioception at the Hw/Sw Interface

• Self-awareness: Knowledge about internal state & environment
  – utilization of resources, e.g. cores, interconnect, memories, I/O
  – temperature distribution, failing components
  – changing applications, workloads, quality of service constraints

• Self-expression: Actions based on goals, values, constraints
  – assignment and migration of computations
  – thermal and power management, fault detection and recovery
  – hardware reconfiguration

• Enable compute nodes to autonomously optimize at runtime
  – performance
  – resource usage
  – energy-efficiency
  – reliability
Hardware/Software Multithreading

• Multithreading
  – applications are partitioned into threads
  – threads synchronize and communicate using abstractions provided by the operating system (e.g. semaphores, message boxes)

• Multithreading as a unified programming and execution model for software and hardware
  – circuits are turned into hardware threads
  – eases programming and porting
  – allows for migrating between software and hardware
Heterogeneous Multi-Cores

applications, quality of service requirements, system state

thread assignment & migration, hardware reconfiguration, power & thermal management
ReconOS

• Our operating system for heterogeneous multi-cores
  – leverages eCos and Linux operating systems
  – uses Xilinx FPGA technology (PowerPC and μBlaze CPUs)
  – dynamic reconfiguration of hardware cores

• ReconOS is open source and available for download
  – source code, tool chain, reference designs
  – documentation, tutorials
  – mailing lists

http://github.com/epics/reconos
Example: Video Object Tracking
Summary

• The hw/sw interface has shifted. Reconfigurable hardware allows us to adapt the hardware at runtime.

• Proprioception (self-awareness/expression) at the compute node level enables us to adapt to changing system states and environments.

• EPiCS uses multithreading as unified programming and execution model for heterogeneous multi-cores.
Further Reading

• Reconfigurable hardware and computing

• Self-awareness/expression in computing systems

• Heterogeneous multi-cores
Thanks for Your Interest!