Programming Reconfigurable Devices via FPGA Regions & Device Tree Overlays

A User View Benchmark on a Declarative FPGA Reconfiguration Framework

Stefan Wiehler (University of Ulm, Missing Link Electronics)
Ulrich Langenbach (Beuth University of Applied Sciences Berlin, Missing Link Electronics)
Linux FPGA Framework Architecture

- CPU
  - DTO
  - User Space
    - Kernel Space
  - FPGA Region
    - FPGA Manager
      - Firmware Subsystem
    - FPGA Bridge
      - PR Decoupler
      - Configuration Access Port
    - Region
      - 0
    - PR
  - FPGA Region
    - 0
A Declarative FPGA Reconfiguration Framework

Device Tree Overlay (DTO) → Device Tree → Partial Reconfiguration → FPGA → AES

Load Platform Driver

AES Bitstream
Scheduling Latency - Profiling Results

- Measurement of example system (AES accelerator)
- Measured latencies via `ftrace` function entry and exit timestamps

- Bitstream Size: **5.9 MiB**
- Overall latency: $\approx 135$ ms

![Runtime Breakdown]

- Load Bitstream
- Partial Reconfiguration
- Load Platform Driver
- Rest incl. Framework
Deficiencies / Future Work

Additional Components

• Scheduler / Governor
• User space interface for device tree manipulation
  • Shields & modular embedded systems
  • Reconfigurable systems (FPGA)
• FPGA vendor tools to support DTO generation

Performance Bottlenecks

• Firmware caching
• FPGA reconfiguration interface
  (Scope of FPGA vendors)
Simplifies Reconfigurable Computing on Linux Systems

- Linux DTO & FPGA Framework enables efficient operation of reconfigurable computing systems
  - Supports scheduling time slices in fractions of minutes
  - > 10 s between reconfigurations ≈2 % overhead in our case

- Enables efficient development of heterogeneous systems on MPSoC-FPGAs
  - Reboot free debug and test cycles for fast turnaround times
  - Both static and runtime reconfigurable systems profit
References

