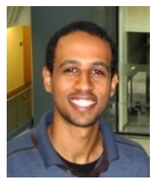




Reliable Decentralized Architectures

Andrea Pellegrini, Biruk Mammo, Valeria Bertacco – University of Michigan

Center for Future Architectures Research

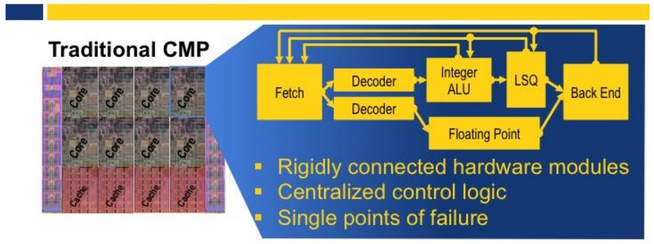
CFAR

STARnet

Theme 5 – Robust Design

TASK 5.2 Fully decentralized architectures

1. Motivation

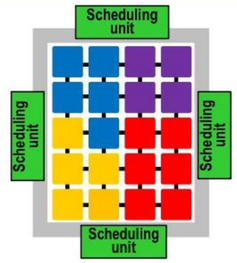


Traditional CMP degrades poorly as fault rates increase

… won't survive high fault rates expected at future technology nodes

2. Decentralized architectures

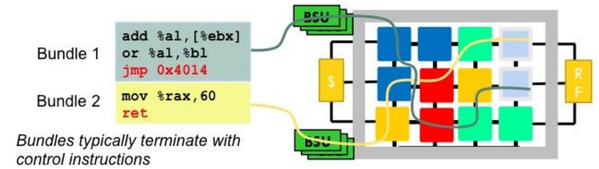
- Sea of redundant HW modules
  - Fine-grained redundancy
- Decentralized control logic
  - Scheduling units schedule execution to HW modules
  - No single points of failure
- Redundant homogeneous interconnect
  - Loosely coupled HW modules
  - Well defined communication interfaces



Examples:
- StageNet [Micro08]
- WaveScalar [Micro03]
- Viper [ISCA12]

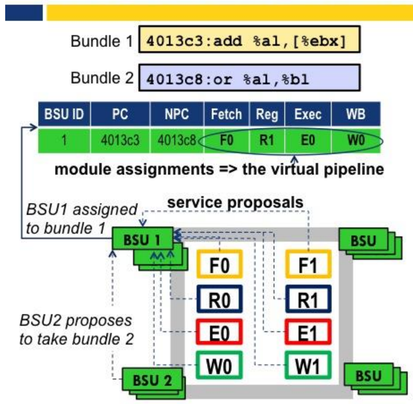
3. Reliable distributed architecture: Cobra

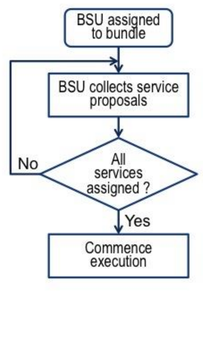
- ISA defines set of ***services*** needed by instructions
- Program is partitioned into instruction ***bundles***
- HW modules ***perform services*** needed by bundles
- ***Virtual pipelines*** formed at runtime
  - HW modules independently ***propose to service*** bundles
  - Bundle Scheduling Units (BSUs) ***assign modules*** to bundles and coordinate execution



Bundles typically terminate with control instructions

4. Cobra execution model

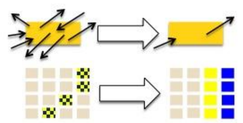




5. Optimizations

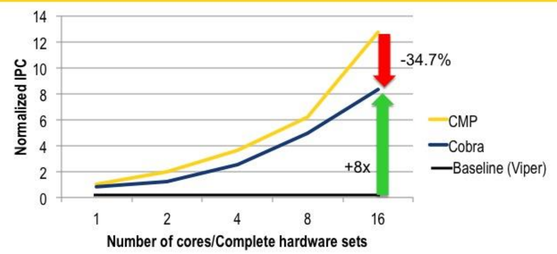
1. Localized HW configuration
   - Reduces instruction execution latency
   - Minimizes interconnect traffic

2. HW configuration transferring
   - Reduces traffic to set up the system
   - Preserves data locality



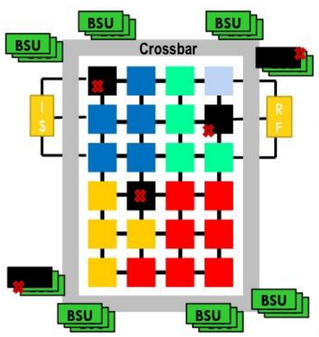
3. Thread-aware cache design
   - Enforces memory access order
   - Enables fast, distributed data caches

6. Performance improvements



- For multi-programmed workloads, optimizations give a performance advantage to Cobra over earlier solutions
- ***8x better performance than Viper at 16 processes, only 34.7% slower than an area-equivalent CMP***

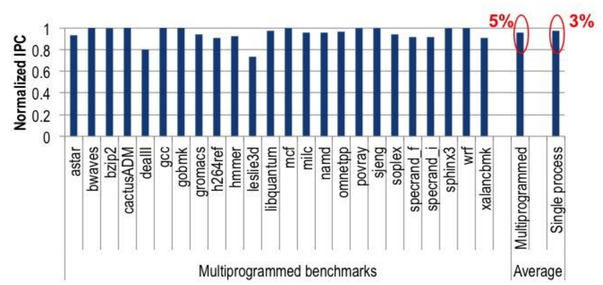
7. Resiliency



1. Fault detection
   - Redundant execution
   - Symptom-based detection
   - Online testing
2. Hardware reconfiguration
   - Disable faulty modules *
3. System state restoration
   - Architectural state and memory restored to previous safe checkpoint

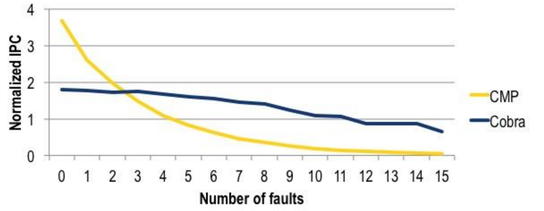
**** System can function as long as there is at least one fault-free HW module per service and at least two BSUs***

8. Fault diagnosis overheads



- Cobra modules tested periodically in 20M cycle intervals
- ***The avg. performance hit of periodic tests is less than 5% – a 3x improvement over a regular core***

9. Resiliency results



- Modeled and simulated using gem5, with SPEC CPU2006
  - CMP: 4 OoO cores, 12 stage pipeline, 128 entry ROB, 32k D$ & I$
  - Cobra: 13 services, 5 HW module types (4-16 copies each)
- ***Beyond 3 faults, Cobra outperforms an area-equivalent CMP of 4 cores***