gem5 Virtual Machine Acceleration

Ali Saidi

Andreas Sandberg (actually did all the work)

Simulation Speed

- Always a concern with simulators
  - Directly impacts size of workloads that can be run
  - How many experiments can be run?
  - How difficult it is to measure performance
- gem5 nominal speeds
  - FastForwarding/ISS mode: ~3 MIPS
  - Detailed CPU and memory system: ~300 KIPS
  - ~1000x slowdown compared to native execution
- How can we speed up the simulator?

The Architecture for the Digital World® ARM®

Virtualization Is Much Faster

- Virtual machine can be nearly as fast as native execution
- No timing information
  - Can be used for checkpoint generation and sampling
- How to migrate state between VM and gem5
  - Use VM peripherals translate native checkpoints into gem5?
    - Thought about using ARM's FastModels tool here
    - Very hard, requires the same peripheral models and similar state
  - Use gem5 models for virtualization?
    - We have enough peripherals to boot a system
    - Can we used those?
    - Checkpoints would be in gem5 format to begin with

The Architecture for the Digital World® ARM®

Using gem5 peripherals for virtualization

- We can use gem5 peripherals
  - PIO requests
    - Trap on non-memory access
    - Issue atomic requests in gem5
    - Return the result for PIO requests
  - DMA
    - Can happen normally to memory
    - Just have to worry about the interrupts

- We've prototyped a system
  - Booting Linux on a RealView PBX platform works
  - Single-core only; ARM-ISA
  - Not particularly well tested at the minute

The Architecture for the Digital World® ARM®

Checkpoints and Switching Modes

- Ran into many issues with switching CPU models
  - Most basic functionality normally works
    - Atomic ➔ Timing
      - This is what people use normally in gem5
    - Most others or repeated switching tends to be problematic
    - Especially true of Timing ➔ Atomic
      - Draining not working as well as it once did
      - Random Errors
- Spend the last two months fixing this
  - Can repeatedly switch back and forth with Atomic, Timing or O3 CPU
  - Combinations of the above
  - Tested every 1ms

The Architecture for the Digital World® ARM®

Regression Tests

- Need better coverage on regression tests
  - We're going to add some that test this particular issue
  - Would really welcome other tests to be added
    - How to balance run time with tests?
    - How/where to keep binaries required by tests?

Changing Draining

- Addition of DrainManager
  - Allows all objects to register with object if they need draining
  - Previously functionality was in SimObject
    - But many things we drained weren't SimObjects

- Drainable class that holds current draining state
  - Virtual methods to:
    - Drain – get to a checkpointable state
    - memWriteback – Writeback any dirty buffers
    - memInvalidate – Flush all state

The Architecture for the Digital World® ARM®

Next Steps

- Hopefully switching/checkpointing issues solved
  - Get back to getting the VM to work
  - Nearly there for a single-CPU
- Multiple CPUs? Multiple Systems?
  - All sounds great
  - Requires multi-threading the simulator
  - Addressing issues with multiple CPUs can interact with devices
    - Any time is CPU requires help do you stop all other CPUs?

The Architecture for the Digital World® ARM®

Questions?