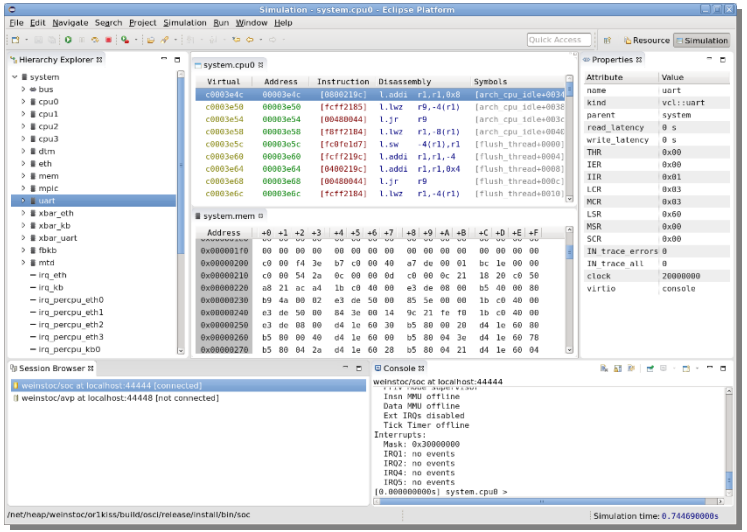
ARM Research Summit 2018

AMVP – An ARM Multicore VP using Parallel SystemC

Jan Henrik Weinstock, 18.09.2018

Virtual Platforms (VPs)

- Simulator that mimics a real hardware platform
  - Accuracy depends on modeling detail
  - Executes unmodified target binary
- Benefits over a hardware prototype
  - Early availability: HW/SW Co-Design
  - Efficient redistribution to SW developers
  - Enables non-intrusive system introspection
- VPs are typically built using SystemC
  - Integrates C/C++ component models
  - High execution speed

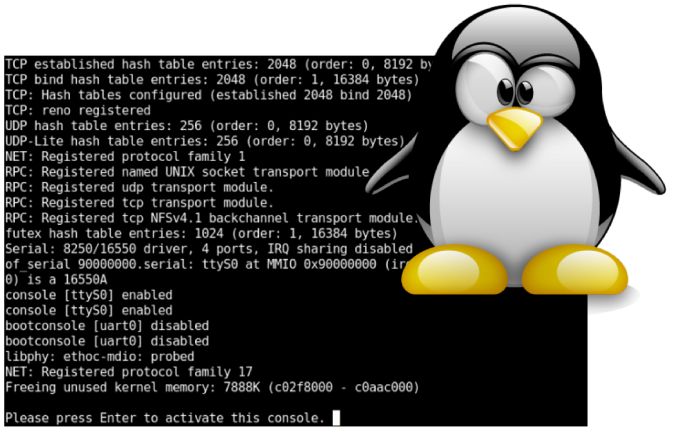


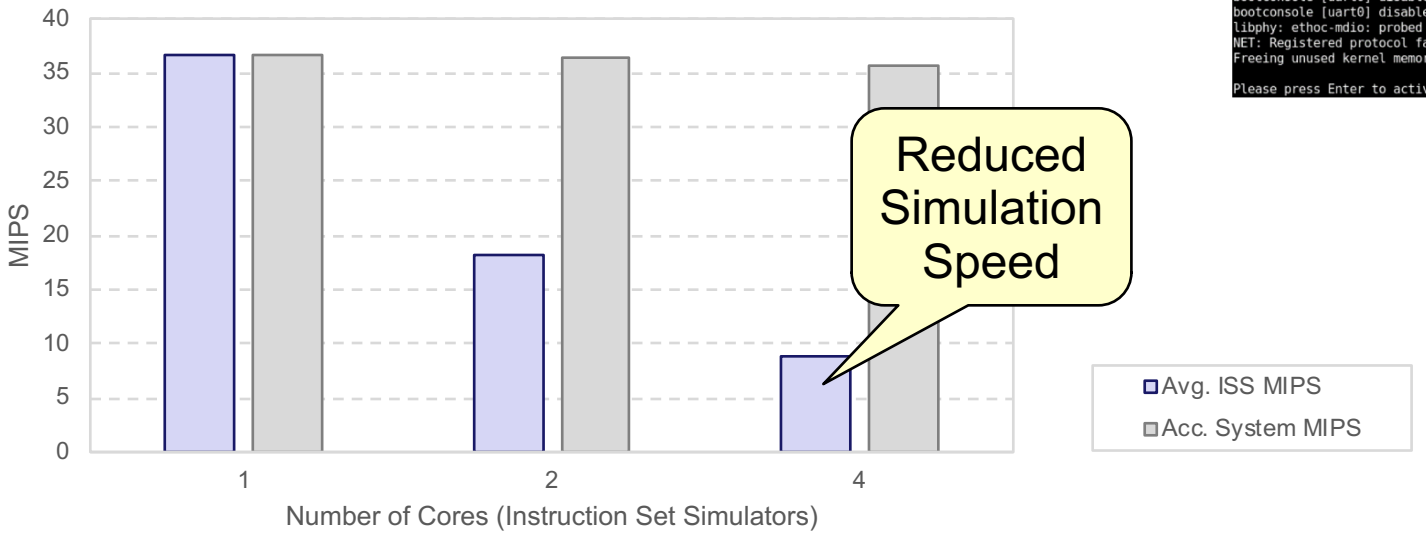
Example VP Graphical User Interface

ICE | RWTH AACHEN UNIVERSITY

VP Simulation Performance

- Example: UC/MC VP performance (MIPS)
  - Target SW: COREMARK/Linux
  - Test single/dual/quad core systems
  - In-house Instruction Set Simulator (ISS)
  - OSCI/Accellera[1] SystemC kernel

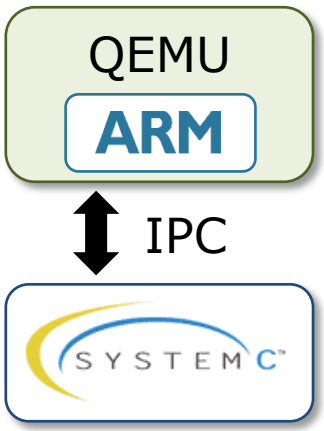




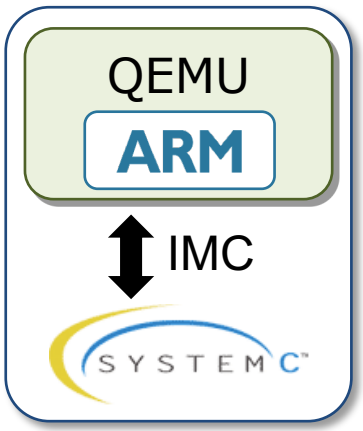
[1] Open SystemC Initiative (OSCI) - http://www.accellera.org/

QEMU and SystemC

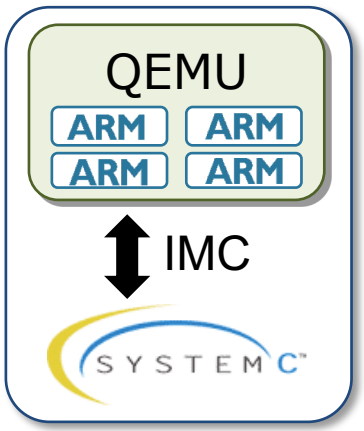
- QEMU only models conventional UP/SMP/HMP designs
  - Supports ARMv7, ARMv8, Thumb, Thumb2, NEON, VFPv2/3/4 …
  - Boots Linux/Android
  - Open Source
- Approaches to couple QEMU and SystemC:



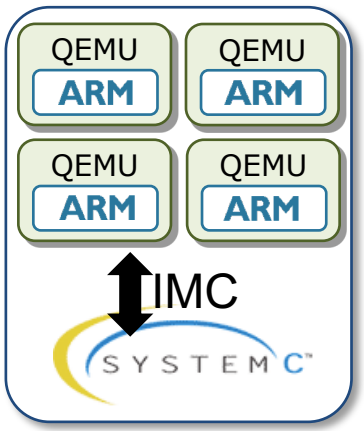
Conventional[1]



GreenSoCs[2]



~FastModels[3]

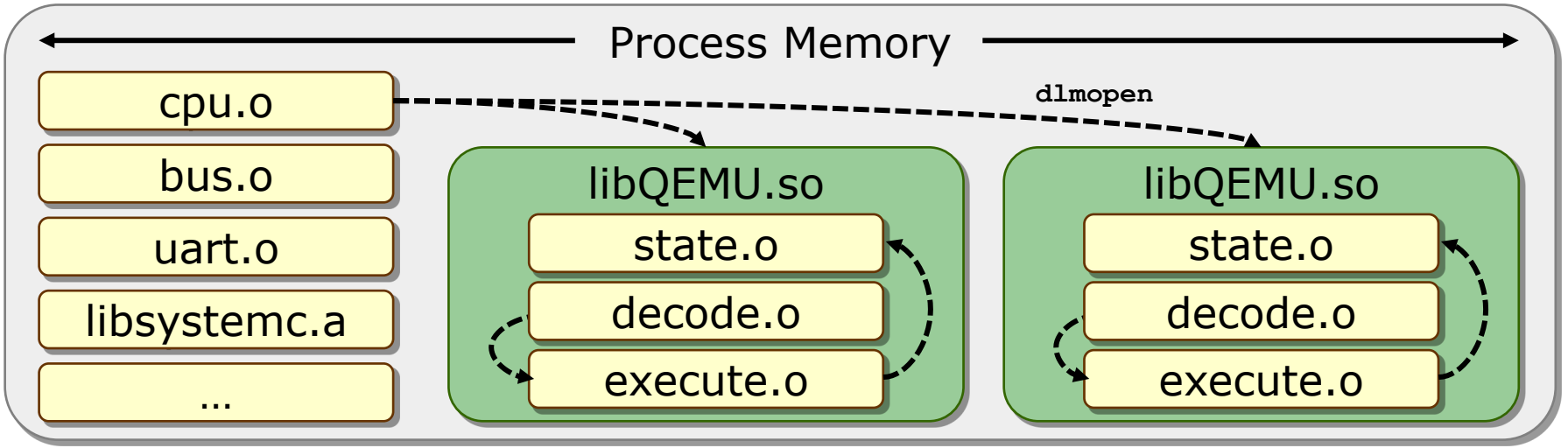


This Work

[1] Fabrice Bellard, http://www.qemu-project.org
[2] GreenSoCs Ltd., http://www.greensocs.com
[3] ARM Holdings plc, http://developer.arm.com/products/system-design/fast-models

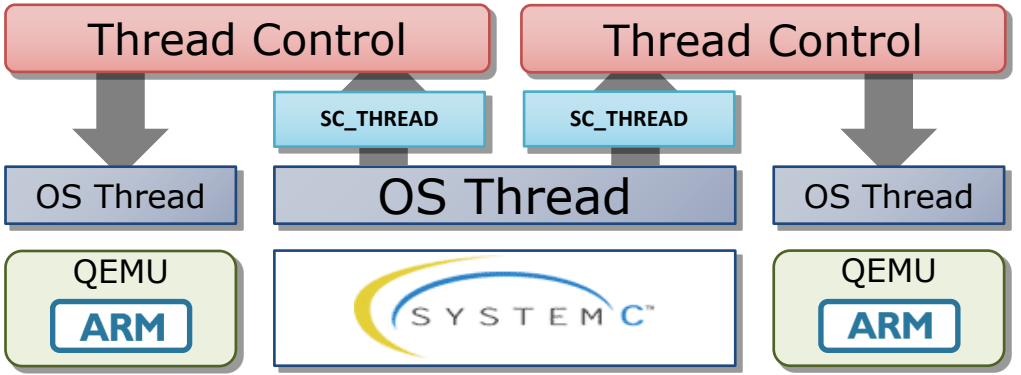
QEMU Multiple Instantiation

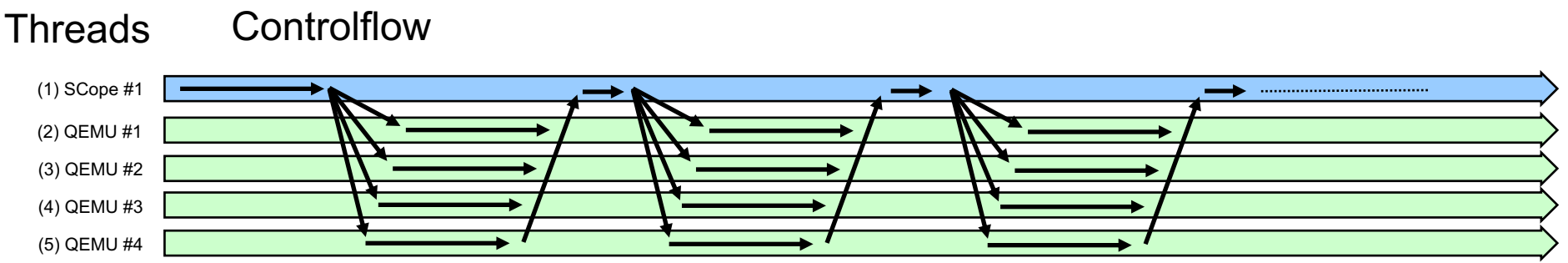
- Convert QEMU into a shared library
- Load multiple copies to replicate global state
  - **dlmopen** loads the same lib into separate namespaces
- Copies must be linked symbolically
  - Resolve symbols at link time (instead of load time)



Parallel Quantum

- Model built-in Parallelism
  - Activate via model property
  - Synchronous Transactions
  - Less code to maintain
  - 2.9k vs 40k LoC (SCope[1])





1. J. H. Weinstock, C. Schumacher, R. Leupers, G. Ascheid and L. Tosoratto, Time-decoupled Parallel SystemC Simulation Design, Automation & Test in Europe Conference & Exhibition (DATE), 2014

Exclusive Memory Access

- Thread-safe implementation of **ldrex** and **strex** needed
  - **ldrex** creates load-link (LL) at **addr**
  - LL breaks upon read/write to **addr**
  - **strex** only stores if LL still present

```
atoinc:
    ldrex r0, [addr]
    add   r0, r0, #1
    strex r0, r1, [addr]
    cmp   r1, #0
    bne   atoinc
```

Example: atomic increment using LL/SC

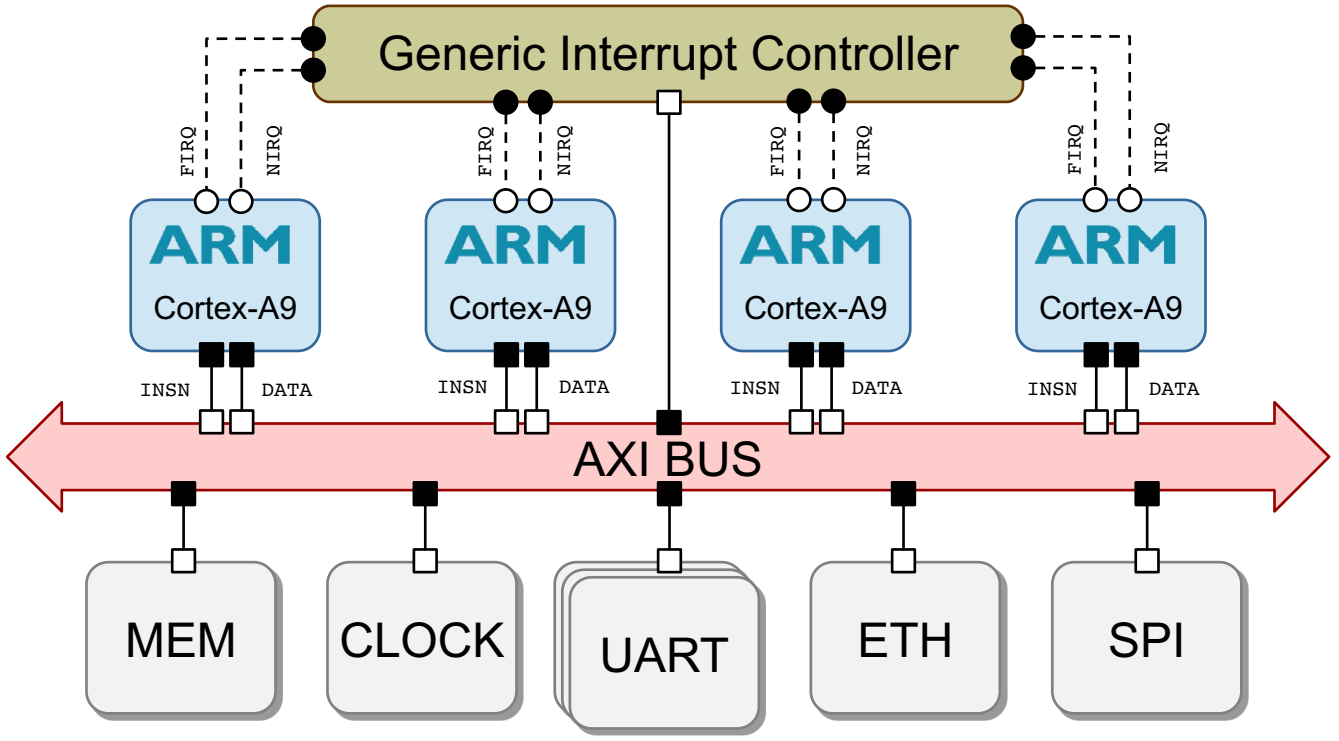
- Problem: Intel (x86) has no matching atomic instructions (only CAS)
  - Option 1: deny DMI access to LL regions ➜ transaction tracing
  - Option 2: retain DMI and emulate as good as possible ➜ ABA Problem

Load-Linked (ldrex)	Store-Conditional (strex)

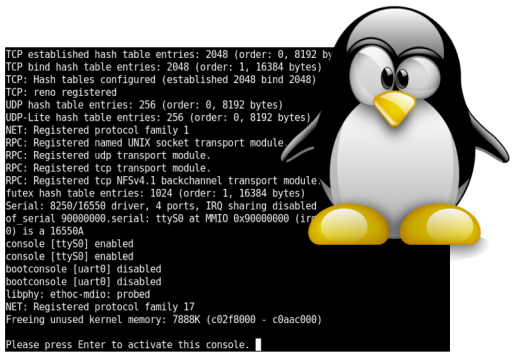
`uint32_t data;`
`memcpy(&data, ptr + addr, 4);`
`core0.link_addr = addr;`
`core0.link_data = data;`
`return data;`
	`if (addr != core0.link_addr)`
	`  return false;`
	`else`
	`  return cas(dmi_ptr + addr,`
	`  core0.link_data,`
	`  data);`

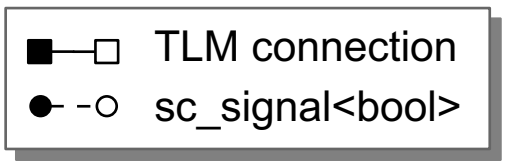
AMVP – Platform Overview

- Quad-core Cortex A9 SMP Design



- Software
  - Linux Kernel
  - Busybox initrd
  - Dhrystone





Single Threaded Performance

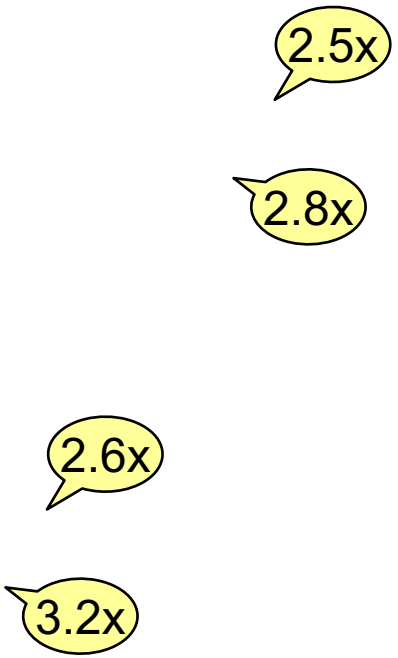
- Dhrystone Benchmark (scattered bare metal, 4.000.000 runs per core)

Single Threaded Performance

- SMP Linux Boot Benchmark (power on → login prompt)

Parallel Performance

- SCope[1] (4 Threads), Parallel Quantum (4 + 1 Threads) vs. OSCI baseline



1. J. H. Weinstock, C. Schumacher, R. Leupers, G. Ascheid and L. Tosoratto, Time-decoupled Parallel SystemC Simulation Design, Automation & Test in Europe Conference & Exhibition (DATE), 2014

Summary & Outlook

- Sequential Operation
  - AMVP is sufficiently detailed to boot and run a realistic OS (Linux)
  - AMVP reaches 100+ MIPS for computational workloads and large quanta (≥ 100us)
  - Future work: test ARMv8 (supported by QEMU, but `ldrex/strex` need more work)

- Parallel Operation
  - Using four worker threads, speedups between 2.5x and 3.2x have been demonstrated

Methodology	SCope Kernel	Parallel Quantum
# Threads	4	5
Speedup Dhrystone	2.8x	2.5x (~11% slower)
Speedup Linux	3.2x	2.6x (~19% slower)
LoC	~40.000	~2900 (~92% less)