Renode: a flexible, open-source simulation for RISC-V system development

7th RISC-V Workshop, Milpitas, 2017-11-29
Michael Gielda, mgielda@antmicro.com
ANTMICRO

- founded 2009, developing Renode since 2010
- founding member of RISC-V foundation
- embedding intelligence in devices
- turning ideas into **software-driven** products
- autonomous systems: drones, mining, military, robotics, industrial automation
ANTMICRO AND RISC-V

- Proof of Concepts (PoC), demonstrators, prototyping, hardware
- Early software adoption (e.g. OS porting, building BSPs, build systems, device management)
- New software development and testing methodologies using Renode
- Custom FPGA blocks, adaptations, processing systems (especially in FPGA SoC)
WHY DID WE BUILD RENODE?

- embedded = dated workflows, little testing
- needed fast simulator for SW developers
- as close to production as possible
- run unmodified software, no recompiling for another target
- capable of multi-node
- easy reuse of models
- natural extensibility
RENODE IN SHORT

- Instruction Set Simulator (ISS)
- software agnostic
- “SoC” emulator - mimic entire platforms
- run multiple nodes
- cores in C
- peripheral models in Python, C# or any .NET based language
RENODE’S STRENGTHS

• full determinism of execution
• transparent debugging, even in multinode setups
• easy integration with your everyday tools
• rich model abstractions to provide additional functionality ‘for free’
• robust infrastructure for debugging: analyze state of peripherals, create traces of functions, integrate with GDB
• automatic tests and headless mode for integration with CI
RISC-V

• open, natural fit for open source
• modular
• flexible
• encouraging reuse
RENODE

• open source SoC Instruction Set Simulator (ISS)
• modular building blocks (models, plugins)
• flexible structure
• user-friendly language to create new combinations for specific hardware
uart: UART.MiV_CoreUART @ sysbus 0x70001000
clockFrequency: 66000000

cpu: CPU.RiscV @ sysbus
cpuType: "rv32g"

plic: Interrupts.PlatformLevelInterruptController @ sysbus 0x40000000
IRQ -> cpu@1
numberOfSources: 31 //based on release notes
WORK WITH MICROSEMI

• Implementation of RISC-V-based Mi-V platform
• Integration with SoftConsole IDE
• Added first-class Windows support
INTEGRATION WITH MICROSEMI SOFTCONSOLE
WHAT CAN YOU DO WITH RENODE?

- share / replicate work environment
- continuously test complex scenarios
- debug many nodes simultaneously
- fully control the hardware
- record / replay events
- inject faults
- simulate the environment (including wireless networking)
- and more
TYPICAL WORKFLOW
CONTINUOUS INTEGRATION WITH PRODUCTION SW
SUPPORTED PLATFORMS - NOT JUST RISC-V
SELECTED INTEGRATIONS

- GDB: The GNU Project Debugger
- Jenkins
- GitLab
- Robot Framework
- Wireshark
- SoftConsole
RISC-V IS MADE FOR IOT - SO IS RENODE
INTERESTED? TALK TO US IF

• you want to adopt a software-driven point of view / methodology
• you want to be ready for multi-node development
• you'd like to get your RISC-V hardware simulated
THANK YOU FOR YOUR ATTENTION!