Running other architecture operating systems and applications on RISC-V using QEMU

Alistair Francis <Alistair.Francis@wdc.com>
RISC-V Summit – Santa Clara
5th of December 2018
What is QEMU?

• QEMU is a very quick open source (mostly GPLv2) emulator and hypervisor

• It is not cycle accurate, but it is functionally accurate

• It uses the Tiny Code Generator (TCG) to translate different guest architecture instructions to host executable code
  – Supports full system (softMMU) emulation
  – Also supports just Linux/BSD user space translation

• It works similarly to GCC with separate host and target support
  – Currently mainline has RISC-V guest support
Basics of Tiny Code Generator (TCG)

• TCG began as a backend for a C compiler
• TCG can convert TCG ops to target (host) instructions
  – It also performs some optimisations and liveness analysis to improve performance
• TCG will combine guest ops into a TB block
  – The end of a block occurs when a branch/jump instruction is encountered
• TCG currently natively supports these targets (hosts)
  – AArch64, ARMv7, x86, AMD64, MIPS, PPC, PPC64, S390 and Sparc
  – Others can run using TCI (discussed on next slide)
  – RISC-V support is in progress, RFC patches are on the QEMU mailing list
TCG Interpreter (TCI)

- TCI is a backend for TCG that generates byte code instead of host assembly
- This bytecode can then be interpreted by the TCI interpreter
- This allows QEMU to run on hosts that don’t have a native supported backed
  - This unfortunately comes at a great speed cost
Why RISC-V host support is useful?

• Cross architecture Linux user space support is useful for compiler and application testing
  – Allows a quick turn around to test compiled RISC-V user space binaries on Intel PCs
  – Can also be used to simplify cross compilation using QEMU
    • A cross compile suddenly becomes a native compile on your PC

• Full system emulation support gives RISC-V the same capability as other mature architectures
  – This helps demonstrate the RISC-V ISA capabilities and potential
What do we have in mainline today?

• TCG Interpreter (TCI) is currently in mainline
  – This is a generic output of TCG that can then be converted into any host instructions
  – This doesn’t require special RISC-V host knowledge and only requires RISC-V guest support
  – This is very slow though
  – Can currently boot basic cross architecture operating systems on RISC-V
    • Requires a few changes to the configure script to support RISC-V hosts

• Work in progress
  – RFC to implement native host support
  – Can run all supported QEMU guests on top of a RISC-V host
  – Significantly faster then the TCI option
Demo of booting x86 OSes on RISC-V QEMU