

# RISC-V ARCHITECTURE TRAINING

## @DEMO: QEMU full system emulator

Jim Wang (<http://phdbreak99.github.io>)

Dec. 2019

# Table of Content

**QEMU**

**Install QEMU**

**Run Freedom-E-SDK on QEMU**

**Boot 64-bit Fedora on QEMU**

# Table of Content

## >>>> QEMU

Install QEMU

Run Freedom-E-SDK on QEMU

Boot 64-bit Fedora on QEMU

# QEMU

- QEMU is a **binary translating emulator**
  - On-the-fly, translate RISC-V instruction to host CPU instruction (e.g. x86)
  - Functional, not accurate; but really fast
    - Good for software debugging, not for hardware debuggin

Type	Example	Performance
Functional	QEMU	100 million to >1 billion instructions per second
Trace-accurate	Spike	10 to 100 million instructions per second
Cycle-accurate	Verilator/rocket-chip	10 to 100 thousand instructions per second

# QEMU (cont'd)

## RISC-V boards

- `virt` - priv v1.10 (16550A UART, virtio-net, virtio-block and device-tree)
- `spike_v1.9` - priv v1.9.1 (HTIF and config-string)
- `spike_v1.10` - priv v1.10 (HTIF and device-tree)
- `sifive_e300` - priv v1.10 (SiFiveUART, HiFive1 compatible)
- `sifive_u500` - priv v1.10 (SiFiveUART and device-tree)

## VirtIO board

- Implements VirtIO MMIO (memory mapped IO)
- Supports VirtIO block devices, network devices, and UART
  - Full system with storage, networking
  - Use PLIC as interrupt controller
- Can boot Busybear Linux
- Can boot Fedora Linux

# Table of Content

## QEMU

>>>> Install QEMU

Run Freedom-E-SDK on QEMU

Boot 64-bit Fedora on QEMU

# Install QEMU

## Install prerequisites

Need newer version GCC, so just installed gcc-8

```
apt-get install software-properties-common  
add-apt-repository ppa:ubuntu-toolchain-r/test  
apt-get update  
apt-get install gcc-8 g++-8
```

## Compile QEMU from source

```
git clone https://git.qemu.org/git/qemu.git  
git submodule update --init --recursive  
  
cd qemu  
./configure --target-list=riscv64-softmmu && make
```

# Table of Content

## QEMU

### Install QEMU

### >>>> Run Freedom-E-SDK on QEMU

### Boot 64-bit Fedora on QEMU



# Run Freedom-E-SDK on QEMU

■ Run examples from Freedom IDE

# Table of Content

**QEMU**

**Install QEMU**

**Run Freedom-E-SDK on QEMU**

**>>>> Boot 64-bit Fedora on QEMU**

# Boot 64-bit Fedora on QEMU

## Download Fedora 64-bit image for RISC-V

```
mkdir disk-images && cd disk-images
wget https://fedorapeople.org/groups/risc-v/disk-images/stage4-disk.img.xz
xzdec -d stage4-disk.img.xz > stage4-disk.img

# Berkeley bootloader
wget https://fedorapeople.org/groups/risc-v/disk-images/bbl
```

# Boot 64-bit Fedora on QEMU (cont'd)

## Boot Linux `~/riscv-git/qemu/run-qemu.sh`

```
./riscv64-softmmu/qemu-system-riscv64 \  
-nographic \  
-machine virt \  
-smp 4 \  
-m 2G \  
-kernel ./disk-images/bbl \  
-object rng-random,filename=/dev/urandom,id=rng0 \  
-device virtio-rng-device,rng=rng0 \  
-append "console=ttyS0 ro root=/dev/vda" \  
-device virtio-blk-device,drive=hd0 \  
-drive file=./disk-images/stage4-disk.img,format=raw,id=hd0 \  
-device virtio-net-device,netdev=usernet \  
-netdev user,id=usernet,hostfwd=tcp::10000-:22
```

Login: root

Password: riscv

# Boot 64-bit Fedora on QEMU (cont'd)

## uname

```
[root@stage4 ~]# uname -a
Linux stage4.fedoraproject.org 4.19.0-rc8 #1 SMP Wed Oct 17 15:11:25 UTC 2018 riscv64 riscv64 riscv64 GNU/Linux
[root@stage4 ~]# cat /proc/cpuinfo
hart      : 0
isa       : rv64imafdcsv
mmu       : sv48

hart      : 1
isa       : rv64imafdcsv
mmu       : sv48

hart      : 2
isa       : rv64imafdcsv
mmu       : sv48

hart      : 3
isa       : rv64imafdcsv
mmu       : sv48
```

# Boot 64-bit Fedora on QEMU (cont'd)

## Hello world

```
[root@stage4 ~]# cat hello.c
#include <stdio.h>

int main(void) {
    printf("Hello world!\n");
}
[root@stage4 ~]# gcc hello.c -o hello
[root@stage4 ~]# ./hello
Hello world!
```

## Size difference

- In QEMU (with Linux)

```
-rwxr-xr-x 1 root root 7936 Jul  5 05:39 hello
```

- With newlib

```
-rwxr-xr-x  1 jimw  1876110778    20880 Jul 22 09:07 hello.elf
```

- 2.63x in size, because Linux version binary doesn't include system call functions

# Boot 64-bit Fedora on QEMU (cont'd)

## Python

```
[root@stage4 ~]# python3
Python 3.6.4 (default, Mar 20 2018, 00:39:12)
[GCC 7.3.1 20180303 (Red Hat 7.3.1-5)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import os
>>> os.uname()
posix.uname_result(sysname='Linux', nodename='stage4.fedoraproject.org', release='4.19.0-rc8', version='#1 SMP W
>>> exit()
```

감사합니다 Natick  
Grazie Danke Ευχαριστίες Dalu  
Thank You Köszönöm  
Спасибо Dank Tack  
谢谢 Merci Seé  
ありがとう

Obrigado