
XuanTie QEMU 用户手册

Release

Release v3.8

12 月 09, 2021

Copyright © 2020 平头哥半导体有限公司，保留所有权利。

本文档的产权属于平头哥半导体有限公司 (下称“平头哥”)。本文档仅能分布给:(i) 拥有合法雇佣关系，并需要本文档的信息的平头哥员工，或 (ii) 非平头哥组织但拥有合法合作关系，并且其需要本文档的信息的合作方。对于本文档，禁止任何在专利、版权或商业秘密过程中，授予或暗示的可以使用该文档。在没有得到平头哥半导体有限公司的书面许可前，不得复制本文档的任何部分，传播、转录、储存在检索系统中或翻译成任何语言或计算机语言。

商标申明

平头哥的 LOGO 和其它所有商标归平头哥半导体有限公司及其关联公司所有，未经平头哥半导体有限公司的书面同意，任何法律实体不得使用平头哥的商标或者商业标识。

注意

您购买的产品、服务或特性等应受平头哥商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，平头哥对本文档内容不做任何明示或默示的声明或保证。由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。平头哥半导体有限公司不对任何第三方使用本文档产生的损失承担任何法律责任。

Copyright © 2021 T-HEAD Semiconductor Co.,Ltd. All rights reserved.

This document is the property of T-HEAD Semiconductor Co.,Ltd. This document may only be distributed to: (i) a T-HEAD party having a legitimate business need for the information contained herein, or (ii) a non-T-HEAD party having a legitimate business need for the information contained herein. No license, expressed or implied, under any patent, copyright or trade secret right is granted or implied by the conveyance of this document. No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise without the prior written permission of T-HEAD Semiconductor Co.,Ltd.

Trademarks and Permissions

The T-HEAD Logo and all other trademarks indicated as such herein are trademarks of Hangzhou T-HEAD Semiconductor Co.,Ltd. All other products or service names are the property of their respective owners.

Notice

The purchased products, services and features are stipulated by the contract made between T-HEAD and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied. The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

平头哥半导体有限公司 T-HEAD Semiconductor Co.,LTD

地址: 杭州市余杭区向往街 1122 号欧美金融城 (EFC) 英国中心西楼 T6

邮编: 311121

网址: www.t-head.cn

Contents:

第一章 XuanTie QEMU	1
1.1 简介	1
1.2 支持的 CPU 型号	1
1.3 框架结构	3
第二章 编译	4
2.1 推荐环境	4
2.2 获取可执行程序	4
第三章 使用示例	5
3.1 简易示例	5
3.2 使用 gdb 调试	6
3.3 RISC-V 示例	6
第四章 选项	7
4.1 常用选项	7
第五章 cskysim	8
5.1 Linux 平台下的使用	8
5.2 Windows 平台下的使用	8
第六章 XuanTie QEMU 用户模式	9
6.1 简介	9
6.2 从源码编译	9
6.3 用户模式简易示例	9
6.4 使用 gdb 调试	10
6.5 RISC-V 用户模式	10
Index	11

第一章 XuanTie QEMU

1.1 简介

XuanTie QEMU 是一个以开源项目 QEMU 为基础，支持 XuanTie 处理器的软件模拟器，提供了带基本外设的 XuanTie 开发板模板。

此外 XuanTie QEMU 还添加了启动程序 cskysim，动态加载外设，profile 等功能。

XuanTie QEMU 的特点：

1. 支持 Linux 和 windows 操作系统。
2. 支持 RISC-V 和 C-SKY 两种体系结构。
3. 可以模拟多种外设，比如网卡，声卡，USB，磁盘等。
4. 不经过 bootloader 就能引导 C-SKY Linux 内核。
5. 多种输出方式，不依赖于 host 系统，可以是 SSH，模拟串口等。
6. 执行不需要管理员权限。
7. 动态编译技术提供的高速模拟。
8. 可模拟多核心 CPU，甚至多 CPU。

1.2 支持的 CPU 型号

XuanTie QEMU 同步支持所有现存的 XuanTie CPU，支持 C-SKY 和 RISC-V 两种 ARCH，包括但不限于以下 CPU 型号。

C-SKY ARCH CPU 列表：

Table 1.1: CPU 型号

CPU 型号	可选特性
e801	
e802	t
e803	t
e804	d, f, t
s802	t
s803	t
i805	f
c807	f, v
c810	t, v
c860	v
r807	f

此外还支持一些旧的型号和命名方式：

Table 1.2: CPU 型号

ABI	CPU 型号	可选特性
ABIV1	CK510	e
	CK610	e, f
ABIV2	CK801	t
	CK802	j, t
	CK803	e, f, t
	CK804	e, f
	CK805	f
	CK807	e, f
	CK810	e, f, t
	CK860	f, v

RISC-V ARCH CPU 列表:

Table 1.3: CPU 型号

CPU 型号	可选特性
e902	m, t
e906	f, d, p
e907	f, d, p
c906	f, d, v
c910	v
c920	

1.3 框架结构

XuanTie QEMU 主要基于 QEMU 的 system emulator，模拟基于 XuanTie CPU 的 SOC 和开发板，包括 CPU 和外围设备，如 TIMER、UART 等。运行在真实开发板上的程序或者操作系统可以不经修改，在 XuanTie QEMU 上直接运行。

XuanTie QEMU 结构框架如下：

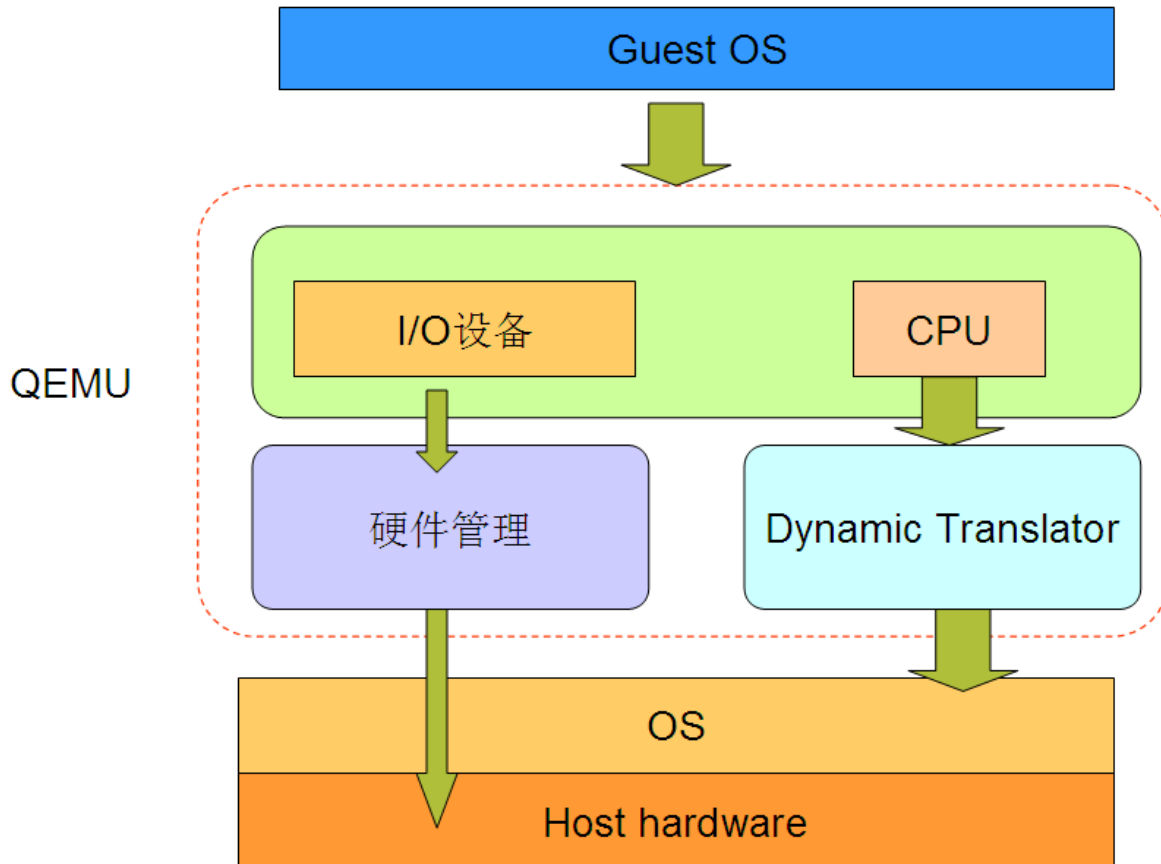


Fig. 1.1: 框架

第二章 编译

2.1 推荐环境

推荐使用以下操作系统版本：

- ubuntu 18.04 64 位
- windows 7 64 位
- windows 10 64 位

2.2 获取可执行程序

XuanTie QEMU 可执行程序有以下几种获取方式：

1. CDS 安装后，以 windows 默认安装为例，可以在 D:\C-Sky\CDS\qemu 下找到。
2. CDK 安装后，以 windows 默认安装为例，可以在 D:\C-Sky\CDK\qemu 下找到。
3. 从 occ.t-head.cn 芯片开放社区获取。
4. 从 T-HEAD 官方获取其他可执行版本。

第三章 使用示例

3.1 简易示例

下面以 CDS/CDK 中的示例 C-SKY UART 程序，来演示系统模式的使用。

该例子中如果 UART 正常工作，将在终端上输出一系列的字母等信息，否则会出现错误信息，甚至没有任何提示。

编译 UART 示例程序的具体过程可以参考 CDS/CDK 的用户手册，下面我们用 XuanTie QEMU 运行该示例编译出的 elf 文件。

```
qemu-system-cskyv2 -machine smartl -kernel /path/of/Uart.elf -nographic
```

QEMU 会在终端，将 UART 示例执行过程的打印，显示如下：

```
Testing uart...
Default configure: Baudrate --- 19200,Parity --- NONE,Wordsize --- 8.
- - -UART0 ready? [y] y

- - - Testing uart mode...
(query mode ): Output is---
  ABCDEFGHIJKLMN- - - [y/n] y    - - -PASS
(interrupt mode ): Output is---
  ABCDEFGHIJKLMN- - - [y/n] y    - - -PASS

- - - Test uart baudrate.
Baudrate is 9600? [y] y :Output is ---
  ABCDEFGHIJKLMN- - -[y/n] y    - - -PASS
Baudrate is 14400? [y] y :Output is ---
  ABCDEFGHIJKLMN- - -[y/n] y    - - -PASS
Baudrate is 38400? [y] y :Output is ---
  ABCDEFGHIJKLMN- - -[y/n] y    - - -PASS
Baudrate is 56000? [y] y :Output is ---
  ABCDEFGHIJKLMN- - -[y/n] y    - - -PASS
Baudrate is 57600? [y] y :Output is ---
  ABCDEFGHIJKLMN- - -[y/n] y    - - -PASS
Baudrate is 115200? [y] y :Output is ---
  ABCDEFGHIJKLMN- - -[y/n] y    - - -PASS
Baudrate is 19200? [y] y :Output is ---
  ABCDEFGHIJKLMN- - -[y/n] y    - - -PASS
```

Fig. 3.1: UART 输出

3.2 使用 gdb 调试

下面仍然以 QEMU 运行 UART 示例程序为例，来说明如何在使用 GDB 来调试 QEMU 上执行的程序。

XuanTie QEMU 使用与上例类似参数，并追加调试的参数，打开端口 23333，等待远程 GDB 调试终端链接，具体如下命令：

```
qemu-system-cskyv2 -machine smartl -kernel /path/of/Uart.elf -nographic -gdb tcp::23333 -S
```

如上，XuanTie QEMU 在等待远程连接到端口 23333。从其他命令行窗口中，用 csky-gdb 接需要调试的 elf 文件：

```
csky-abiv2-elf-gdb /path/of/uart_test/Uart.elf
```

在 GDB 的提示后，输入以下命令连接 QEMU：

```
(cskygdb) target remote localhost:23333
```

本例当中 GDB 连接的目标端口是由参数 `-gdb tcp::23333` 指定为 23333。

接下来便可以与调试硬件开发板一样使用 GDB 进行调试了。例如设断点，单步执行，查看寄存器值等操作。

3.3 RISC-V 示例

对于 RISC-V UART 程序，模拟器需要替换为 `qemu-system-riscv32`，调试器需要替换为 `riscv64-unknown-elf-gdb`。

第四章 选项

4.1 常用选项

以下是一些常用选项，更多的选项可以参考《QEMU Emulator User Documentation》。

-help

显示帮助信息。

-version

显示版本信息。

-machine

选择模拟的开发板，可以输入 `-machine help` 获取一个完整的开发板列表。

-cpu

选择 CPU 类型（例如 `-cpu ck803`），可以输入 `-cpu help` 获取完整的 CPU 列表。

-nographic

禁止所有的图形输出，模拟的串口将会重定向到命令行。

-gdb tcp::port

设置连接 GDB 的端口，（例如 `-gdb tcp::23333`, 将 23333 作为 GDB 的连接端口）

-S

在启动时冻结 CPU ，（例如与 `-gdb` 配合，通过 GDB 控制继续执行）

第五章 cskysim

cskysim 是 T-HEAD 为了简化 XuanTie QEMU 系统模式使用，提供的一个 QEMU 启动程序。cskysim 用 xml 文件整合了常用的 QEMU 参数，为用户提供了 XuanTie 虚拟环境的典型参数。另外，cskysim 跟 QEMU 配合，还提供了动态加载外设等功能。

5.1 Linux 平台下的使用

Linux 的安装目录下，lib/qemu 中，可以找到默认的 xml 文件，这些文件提供了通用的 cskysim 配置。

以运行 C-SKY Linux 内核为例，可以用 -soc 参数指定 xml 配置文件，用 -kernel 指定 Linux 内核 elf 文件。如果需要调整 QEMU 参数，可以修改 xml 文件，或者直接在命令行后直接加更多的 QEMU 参数。

```
cskysim -soc lib/qemu/soccfg/cskyv2/dummyh_cfg.xml -kernel path/of/bin/vmlinux
```

在弹出窗口中，通过菜单栏 view->serial0 切换到串口，就可以看到打印输出。

如果要直接在终端输出，命令行之后追加 nographic 即可。

```
cskysim -soc lib/qemu/soccfg/cskyv2/dummyh_cfg.xml -kernel path/of/bin/vmlinux -nographic
```

5.2 Windows 平台下的使用

windows 下 qemu 动态插件使用与 linux 类似，不同的只是动态库文件从 so 变成了 dll。windows 安装后目录组成与 linux 类似，默认位于 C:\Program Files (x86)\C-Sky 下。

以运行 C-SKY Linux 内核为例，-soc 参数指定 xml 配置文件，其他参数用法与 QEMU 相同。弹出窗口后，通过菜单栏 view->serial0 切换到串口，就可以看到打印输出。

```
cskysim.exe -soc soccfg\cskyv2\dummyh_cfg.xml -kernel path\of\bin\vmlinux
```

如果要直接在终端输出，需要先设置 SDL 库的环境变量 SDL_STDIO_REDIRECT=no，命令行之后再追加 nographic 选项。

```
set SDL_STDIO_REDIRECT=no
cskysim.exe -soc soccfg\cskyv2\dummyh_cfg.xml -kernel path\of\bin\vmlinux -nographic
```

对 cskysim 中 xml 文件的详细说明，可以参考《dynsoc 开发手册》

第六章 XuanTie QEMU 用户模式

6.1 简介

XuanTie QEMU 用户模式是提供 XuanTie Linux 应用程序执行环境的一个模式。允许直接执行绝大多数 XuanTie Linux 应用程序。

6.2 从源码编译

这节，以从源码编译 ABIV2 的 C-SKY 用户模式为例，描述了如何从源码编译 C-SKY 用户模式。

编译：

```
mkdir build
../configure --target-list=cskyv2-linux-user
make
```

编译 RISC-V 用户模式。

编译：

```
mkdir build
../configure --target-list=riscv64-linux-user
make
```

如果需要安装可执行程序到本机执行目录，则可以 **安装：**

```
make install
```

6.3 用户模式简易示例

用户模拟是 XuanTie QEMU 模拟 Linux 程序执行环境的模式。

以仅有 `hello world` 打印的 `main.c` 为例。

如果程序正常执行，将在终端上输出 `hello world`，否则会出现错误信息，甚至没有任何提示。

```
qemu-cskyv2 /path/of/a.out
```

6.4 使用 gdb 调试

下面仍然以 XuanTie QEMU 运行 `hello world` 示例程序为例，来说明如何在使用 GDB 来调试 QEMU 上执行的程序。

QEMU 使用与上例类似参数，并追加调试的参数，打开端口 23333，等待远程 GDB 调试终端链接，具体如下命令：

```
qemu-cskyv2 -g 23333 /path/of/a.out
```

如上，QEMU 在等待远程连接到端口 23333。从其他命令行窗口中，用 `csky-gdb` 接需要调试的 elf 文件：

```
csky-abiv2-elf-gdb /path/of/a.out
```

在 GDB 的提示后，输入以下命令连接 QEMU：

```
(cskygdb) target remote localhost:23333
```

本例当中 GDB 连接的端口是由参数 `-g 23333` 指定为 23333。

接下来便可以与调试普通 Linux 应用程序一样使用 GDB 进行调试了。例如设断点，单步执行，查看寄存器值等操作。

6.5 RISC-V 用户模式

对于 RISC-V 用户模式程序，模拟器需要换成 `qemu-riscv64`，调试器需要换成 `riscv64-unknown-linux-gnu-gdb`。

Index

Symbols

- S
 - command line option, [7](#)
- cpu
 - command line option, [7](#)
- gdb tcp::port
 - command line option, [7](#)
- help
 - command line option, [7](#)
- machine
 - command line option, [7](#)
- nographic
 - command line option, [7](#)
- version
 - command line option, [7](#)

C

- command line option
 - S, [7](#)
 - cpu, [7](#)
 - gdb tcp::port, [7](#)
 - help, [7](#)
 - machine, [7](#)
 - nographic, [7](#)
 - version, [7](#)