# 

# RISCV编译及模拟环境riscv-tools

# 搭建及配置文档

版本记录

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 编写人 | 日期 | 备注 |
| 0.1 | 高铮 | 2017年3月28日 | 建立文档 |
| 0.11 | 刘先华、高铮 | 2017年3月28日 | 审核及修订 |
|  |  |  |  |
|  |  |  |  |

一、相关配置说明

**1.1 软硬件配置**

192.168.200.16、192.168.200.17 两台实验室服务器

硬件基本配置为：

CPU: E5-2609 @ 1.9 GHz

Mem: 64G

软件环境为：

OS：CentOS

Linux： 3.10.0-327

**1.2 安装环境说明**

所有源码包放置在/opt/src目录下，其中包括：

|  |  |
| --- | --- |
| riscv-gnu-toolchain | 工具链 |
| riscv-qemu | qemu模拟器 |
| riscv-newlib-spike | spike模拟器和工具链 |
| riscv-tests | riscv测试工具 |

所有工具安装路径为/opt/，其中包括：

|  |  |
| --- | --- |
| riscv-glibc | 包含glibc的工具链 |
| riscv-qemu | qemu模拟器 |
| riscv-newlib-spike | 包含newlib的工具链以及spike模拟器 |

注：由于是搭建公共环境，因此需使用具有root权限的账号搭建环境，下面的相关搭建过程中，可能要根据实际情况增加sudo。

二、具体内容

**2.1 安装riscv-gnu-toolchain with glibc**

1）下载riscv-gnu-toolchain源码包并解压缩

|  |
| --- |
| tar –xf riscv-gnu-toolchain.tar |

2）进入源代码目录，安装基于glibc库的toolchain，使用-prefix指定其安装路径为/opt/riscv-glibc/

|  |
| --- |
| ./configure -prefix=/opt/riscv-glibc  make linux |

**2.2 安装系统运行模式、用户运行模式的riscv-qemu模拟器**

1）下载riscv-qemu的源码包并进行解压

|  |
| --- |
| tar -xf riscv-qemu.tar |

2）进入源代码目录，执行以下命令。执行时应注意权限。

|  |
| --- |
| ./configure --target-list=riscv64-softmmu, riscv32-softmmu, riscv64-linux-user, riscv32-linux-user --prefix=/opt/riscv-qemu  make  make install |

**2.3 安装riscv-toolchain-newlib-spike**

1）下载riscv-qemu的源码包并进行解压

|  |
| --- |
| tar -xf riscv-toolchain-newlib-spike.tar |

2）进入源代码目录，执行以下命令。执行时应注意权限。

|  |
| --- |
| export RISCV=/opt/riscv-newlib-spike  export PATH=$PATH:$RISCV/bin  ./build.sh |

**2.4 使用Whestone进行简单测试**

1）使用工具链编译whestones.c

|  |
| --- |
| /opt/riscv-glibc/bin/riscv64-unknown-linux-gnu-gcc whetstone.c -lm -o whetstone |

2）使用qemu模拟器执行

|  |
| --- |
| /opt/riscv-qemu/bin/qemu-riscv64 -L /opt/riscv-glibc/sysroot/ whetstone |

运行结果未发现错误。

3）使用spike模拟器执行

|  |
| --- |
| /opt/riscv-newlib-spike/bin/spike pk whetstone |

运行结果未发现错误。

**2.5 使用Dhrystone进行简单测试**

1）修改Makefile

* 修改Makefile使Dhrystone使用riscv-gnu-toolchain工具进行编译，将

GCC=gcc修改为GCC=/opt/riscv-glibc/bin/riscv64-unknown-linux-gnu-gcc

* 修改Makefile中对TIMES的宏定义
* 修改Makefile中make clean，加上${UNIX\_PROGS},使其在make clean时能够删除gcc\_dhry2、gcc\_dhry2reg程序。

2）重新编译。执行make clean并再次执行make。

3）执行如下命令使用模拟器riscv-qemu运行dhrystone

|  |
| --- |
| /opt/riscv-qemu/bin/qemu-riscv64 -L /opt/riscv-glibc/sysroot/ gcc\_dry2  /opt/riscv-qemu/bin/qemu-riscv64 -L /opt/riscv-glibc/sysroot/ gcc\_dry2reg |

运行编译出的gcc\_dhry2和gcc\_dhry2reg。其运行结果均未发现错误。

**2.6安装riscv-tests**

1）下载riscv-tests的源码包并进行解压

|  |
| --- |
| tar -xf riscv-tests.tar |

2）进入源代码目录，执行以下命令。执行时应注意权限。

|  |
| --- |
| export RISCV=/opt/riscv-newlib-spike  export PATH=$PATH:$RISCV/bin  autoconf  ./configure --prefix=$RISCV/target  make  make install |