

RTT Linux Dev

1. 下载RT-Thread源码:

```
git clone https://github.com/RT-Thread/rt-thread.git
```

2. 安装工具链: Ubuntu可以采用官网的文档进行安装即可; Manjaro(Arch)可以直接在软件仓库里安装。 需要安装的软件包如下:

1. arm-none-eabi-gcc
2. arm-none-eabi-gdb
3. scons
4. libncurses5
5. openocd

安装openocd后需要配置规则文件。将 **/usr/share/openocd/contrib/60-openocd.rules** 复制到 **/etc/udev/rules.d** 目录下:

```
sudo cp /usr/share/openocd/contrib/60-openocd.rules /etc/udev/rules.d
```

- 对于Ubuntu用户, 需要将当前用户加入plugdev用户组

```
usermod -aG plugdev $(whoami)
```

- 对于Manjaro用户, 一般没有plugdev用户组, 可以直接将当前用户加入root用户组, 并且将rules文件中的GROUP字段从plugdev修改为root即可。

```
usermod -aG root $(whoami)
```

注意加入用户组之后可能需要注销或重启才能生效。

3. 修改对应开发板的BSP文件夹下的rtconfig.py中的工具链路径为编译器实际所在的路径:

```

# 修改为RTT编译环境，如将 EXEC_PATH = '/usr/bin' 改为：
BSP_LIBRARY_TYPE = None

if os.getenv('RTT_CC'):
    CROSS_TOOL = os.getenv('RTT_CC')
if os.getenv('RTT_ROOT'):
    RTT_ROOT = os.getenv('RTT_ROOT')

# cross_tool provides the cross compiler
# EXEC_PATH is the compiler execute path, for example, CodeSourcery, Keil MDK, IAR
if CROSS_TOOL == 'gcc':
    PLATFORM = 'gcc'
    EXEC_PATH = r'/usr/bin/'
elif CROSS_TOOL == 'keil':
    PLATFORM = 'armcc'
    EXEC_PATH = r'C:/Keil_v5'
elif CROSS_TOOL == 'iar':
    PLATFORM = 'iar'
    EXEC_PATH = r'C:/Program Files (x86)/IAR Systems/Embedded Workbench 8.0'

if os.getenv('RTT_EXEC_PATH'):
    -- INSERT --

```

- #### 4. 生成对应于开发板的工程文件夹: `scons --dist`

- ## 5. 安装配置VSCode

- ## 1. 安装Cortex-Debug插件

1. 根据Usage说明网页配置launch.json文件
2. 配置对应于开发板的调试配置文件，一般位于 **/usr/share/openocd/scripts/board** 目录下，复制到当前目录，并配置launch.json文件
3. 下载对应于开发板芯片的SVD文件，并配置launch.json文件中的svdFile字段

6. 连接开发板，直接采用VSCode左侧的调试面板对开发板进行调试：

