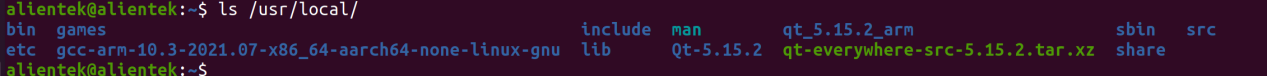
1. 将交叉编译的工具拷贝到/usr/local/下

cp -r /usr/local/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu /usr/local/gcc-arm-10.3-2021.07-x86\_64-aarch64-none-linux-gnu/





1. 解压QT源码，在源码中配置交叉编译

进入源码的qtbase/mkspecs

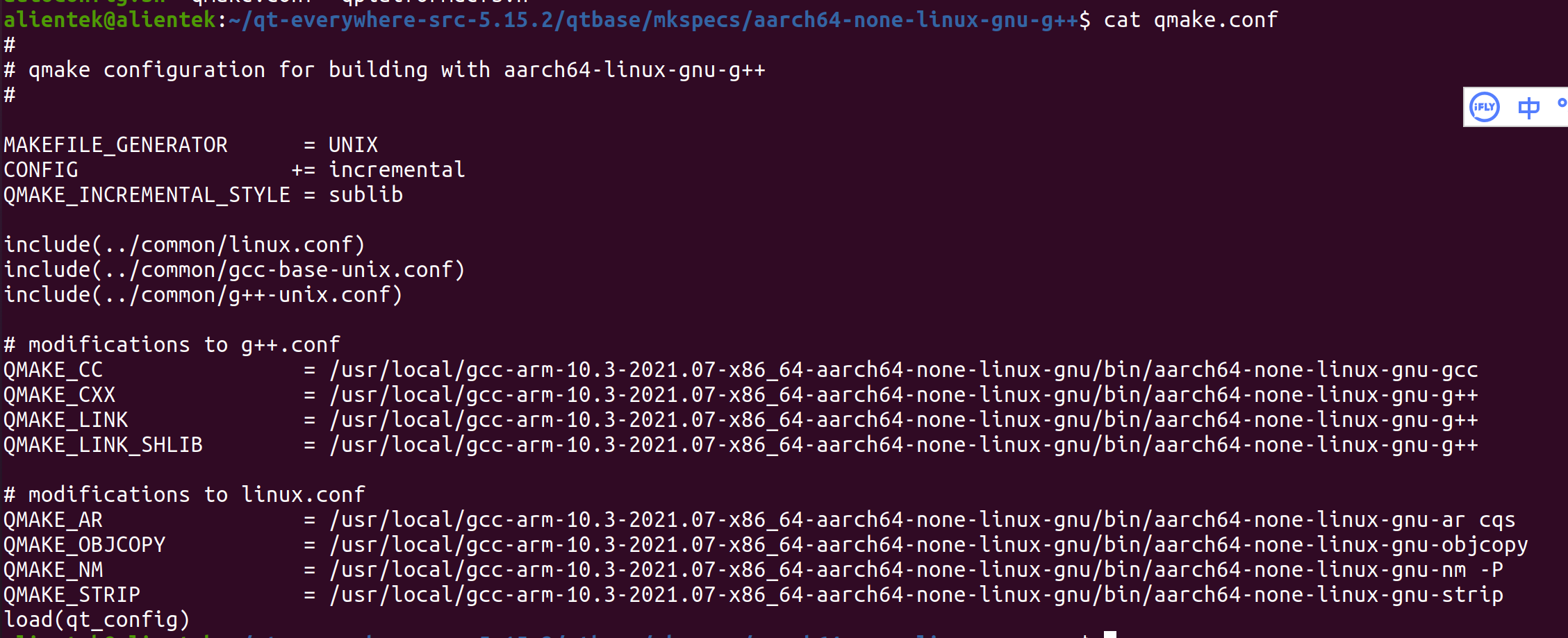
/home/alientek/qt-everywhere-src-5.15.2/qtbase/mkspecs

选择名称与你所用编译工具相近的文件夹拷贝一份，重命名为所用编译器名称

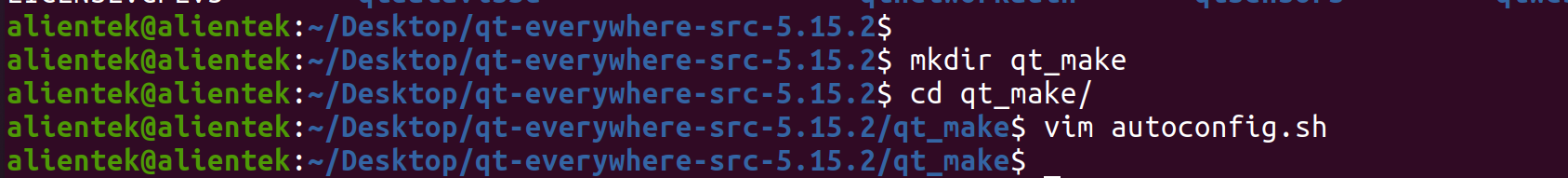
cp -r linux-aarch64-gnu-g++/ aarch64-none-linux-gnu-g++/



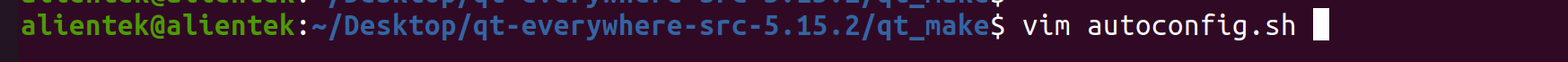
修改qmake.conf文件，修改编译器



返回源码目录下，创建一个文件夹，进行编译。避免污染源码



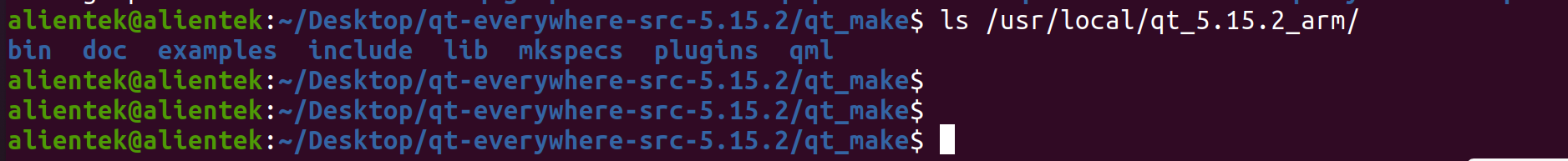
创建编译的脚本，可指定进行修改

给予权限后执行。

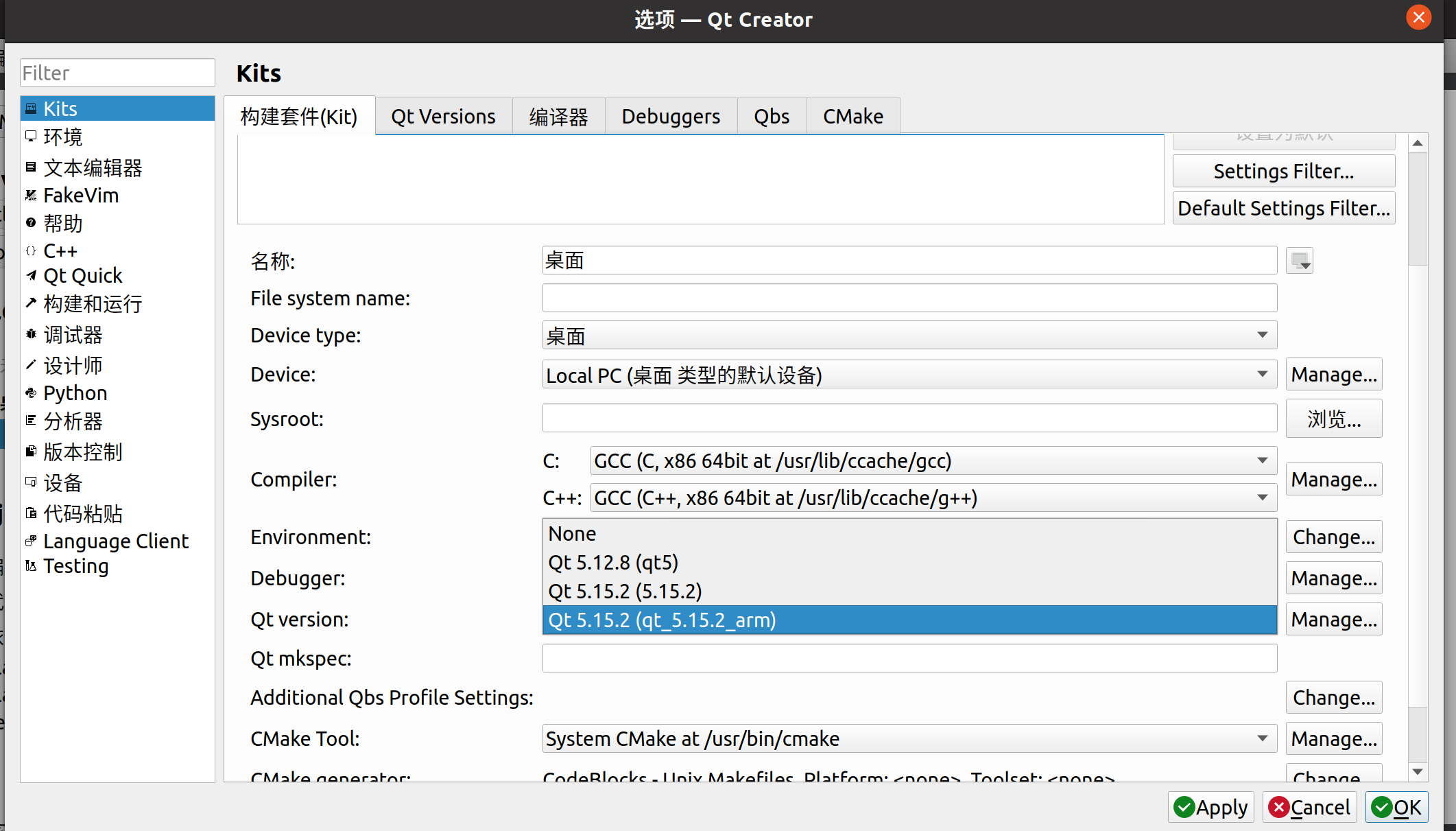
然后执行make

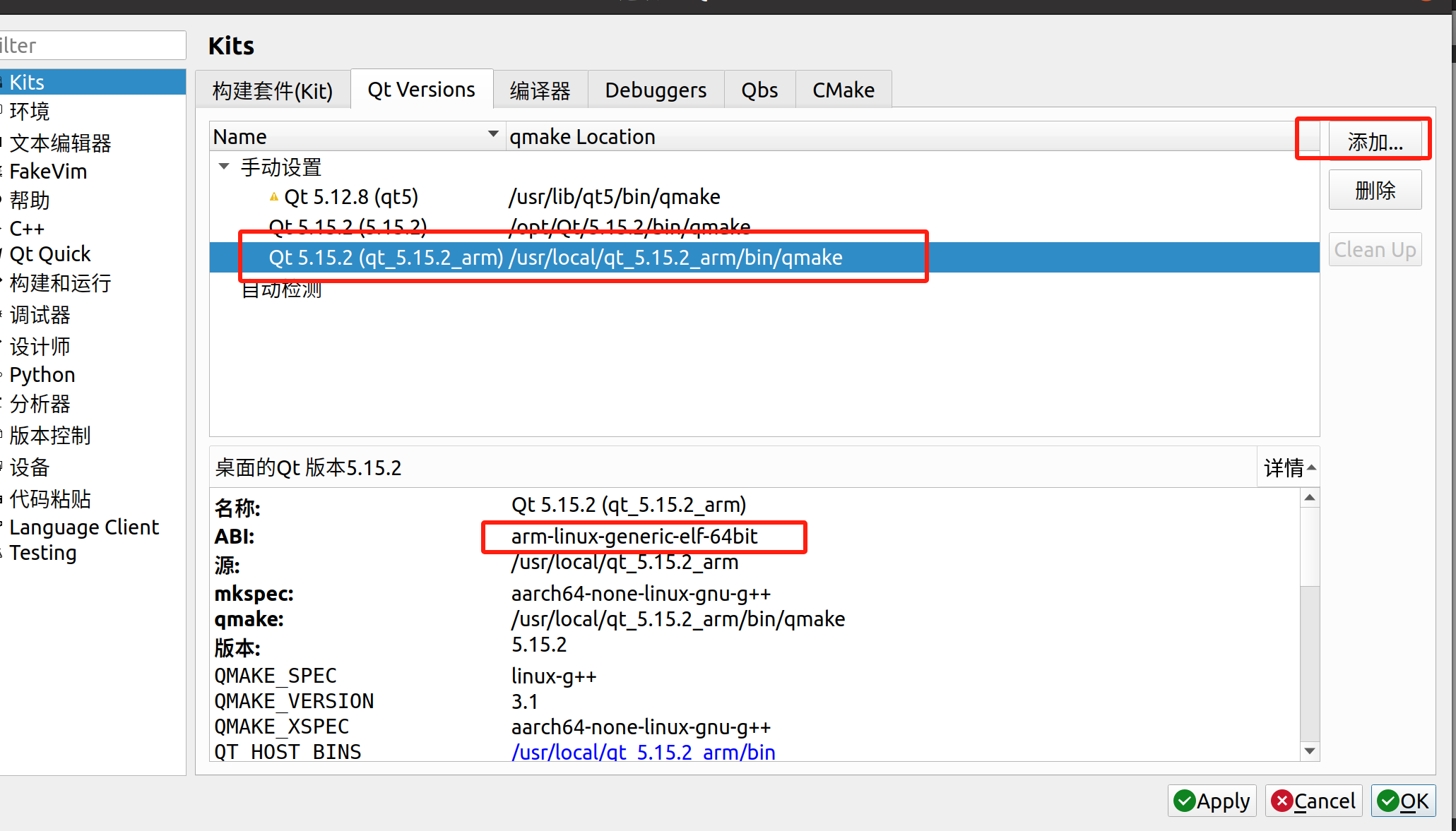
sudo make install

查看是否安装成功



QT中添加、查看





#!/bin/sh

../configure \

-xplatform aarch64-none-linux-gnu-g++ \

-opensource \

-confirm-license \

-release \

-strip \

-shared \

-optimized-qmake \

-c++std c++11 \

--rpath=no \

-pch \

-skip qt3d \

-skip qtactiveqt \

-skip qtandroidextras \

-skip qtcanvas3d \

-skip qtconnectivity \

-skip qtdatavis3d \

-skip qtdoc \

-skip qtgamepad \

-skip qtlocation \

-skip qtmacextras \

-skip qtnetworkauth \

-skip qtpurchasing \

-skip qtremoteobjects \

-skip qtscript \

-skip qtscxml \

-skip qtsensors \

-skip qtspeech \

-skip qtsvg \

-skip qttools \

-skip qttranslations \

-skip qtwayland \

-skip qtwebengine \

-skip qtwebview \

-skip qtwinextras \

-skip qtx11extras \

-skip qtxmlpatterns \

-make libs \

-make examples \

-nomake tools -nomake tests \

-gui \

-widgets \

-dbus-runtime \

--glib=no \

--iconv=no \

--pcre=qt \

--zlib=qt \

-no-openssl \

--freetype=qt \

--harfbuzz=qt \

-no-opengl \

-linuxfb \

--xcb=no \

--libpng=qt \

--libjpeg=qt \

--sqlite=qt \

-plugin-sql-sqlite \

-recheck-all \

-prefix /usr/local/qt\_5.15.2\_arm