

IDO-EVB3562-V1 Linux 开发手册

1 Linux SDK下载

1.1 源码下载

1.2 源码解压

2 Linux安装SDK编译依赖环境

3 Linux SDK编译

3.1 配置选择

3.2 开始编译

3.2.1 完整编译

3.2.2 编译uboot/kernel

4 Buildroot QT SDK使用方法

4.1 安装QT-SDK包

4.2 编译QT程序

4.3 运行QT程序

5 Ubuntu QT SDK使用方法

5.1 安装QT-SDK包

5.2 测试

5.3 Qt Creator 配置Kit



IDO-EVB3562-V1

Linux 开发手册

深圳触觉智能科技有限公司

www.industio.cn

文档修订历史

版本	PCBA版本	修订内容	修订	审核	日期
V1.0	V1B	创建文档	TWX	IDO	2023/09/06
V1.2	V1B	文档优化	LZR	IDO	2024/04/17
V1.3	V1B	更新SDK下载链接 增加ubuntu文件系统编译选项 增加ubuntu QT SDK使用描述	TWX	IDO	2024/08/14

V1.4	V1B	修正编译kernel生成文件为 boot.img	TWX	IDO	2024/08/2 2
V1.5	V1B	更新Ubuntu QT SDK使用方法	LJH	IDO	2024/09/19

1 Linux SDK下载

1.1 源码下载

从以下地址下载IDO-EVB3562的Linux_SDK:

通过网盘分享的文件: Linux_SDK

链接: https://pan.baidu.com/s/1ypkRUvGxXqZRsoPCy_Vktg

提取码: 1234

1.2 源码解压

由于SDK打包后体积较大,我们在上传到百度云盘前把SDK包按照4GB大小分割了,因此下载后需要合并,命令如下:

▼ Shell

```
1  $ cat Industio-RK3562_Linux5.10_240814.a* > Industio-RK3562_Linux-SDK.tar.g
z
```

结果如下图所示:

```
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$ ls
Industio-RK3562_Linux5.10_0423.aa Industio-RK3562_Linux5.10_0423.ac Industio-RK3562_Linux5.10_0423.ae Patch
Industio-RK3562_Linux5.10_0423.ab Industio-RK3562_Linux5.10_0423.ad Industio-RK3562_Linux5.10_0423.af
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$ cat Industio-RK3562_Linux5.10_0423.a* > Industio-RK3562_Linux-SDK.tar.gz
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$ ls
Industio-RK3562_Linux5.10_0423.aa Industio-RK3562_Linux5.10_0423.ac Industio-RK3562_Linux5.10_0423.ae Industio-RK3562_Linux-SDK.tar.gz
Industio-RK3562_Linux5.10_0423.ab Industio-RK3562_Linux5.10_0423.ad Industio-RK3562_Linux5.10_0423.af Patch
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$
```

合并后,就可以进行解压了

注意: 不要使用sudo解压,否则会导致后面编译出问题,命令如下:

```
1 $ tar -zxvf Industio-RK3562_Linux-SDK.tar.gz
```

结果如下图所示：

```
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$ tar -zxvf Industio-RK3562_Linux-SDK.tar.gz
rk3562_linux/
rk3562_linux/debian/
rk3562_linux/debian/mk-image.sh
rk3562_linux/debian/ubuntu-build-service/
rk3562_linux/debian/ubuntu-build-service/bullseye-desktop-arm64
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/configure
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/live/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/live/0021-silence-systemd.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/live/0022-disable-systemd-services.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/live/22-lightdm-autologin.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/live/0002-add_linaro_to_groups.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/live/0001-setup_user_linaro.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/live/0098-resolvconf.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/hooks/live/0003-check_sudoers_for_admin.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/preseed/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/preseed/dictionaries-common.cfg.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/package-lists/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/package-lists/linaro.list.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/includes.chroot/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/includes.chroot/etc/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/includes.chroot/etc/rc.local
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/includes.chroot/etc/hosts
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/includes.chroot/etc/hostname
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/includes.chroot/etc/kernel-img.conf
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/includes.chroot/etc/default/
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/customization/includes.chroot/etc/default/locale
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/README
rk3562_linux/debian/ubuntu-build-service/bullseye-lxde-arm64/Makefile
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/configure
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/live/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/live/0021-silence-systemd.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/live/0022-disable-systemd-services.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/live/22-lightdm-autologin.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/live/0002-add_linaro_to_groups.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/live/0001-setup_user_linaro.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/live/0098-resolvconf.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/hooks/live/0003-check_sudoers_for_admin.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/preseed/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/preseed/dictionaries-common.cfg.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/package-lists/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/package-lists/linaro.list.chroot
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/includes.chroot/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/includes.chroot/etc/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/includes.chroot/etc/rc.local
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/includes.chroot/etc/hosts
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/includes.chroot/etc/hostname
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/includes.chroot/etc/kernel-img.conf
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/includes.chroot/etc/default/
rk3562_linux/debian/ubuntu-build-service/bullseye-base-armhf/customization/includes.chroot/etc/default/locale
```

解压得到的rk3562-linux即为sdk目录，如下图所示：

```
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux$ ls
app buildroot build.sh debian device docs external kernel kernel-5.10 Makefile output prebuilts README.md rkbin rkflash.sh rockdev tools u-boot yocto
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux$
```

2 Linux安装SDK编译依赖环境

使用Ubuntu22.04 64Bit系统编译EVB3562 SDK。在编译前需执行以下命令安装依赖环境，命令如下：

```

1 $ sudo apt-get install repo git ssh make gcc libssl-dev liblz4-tool \
2 expect g++ patchelf chrpath gawk texinfo chrpath diffstat binfmt-support \
3 qemu-user-static live-build bison flex fakeroot cmake \
4 unzip device-tree-compiler python-pip libncurses-dev expect

```

结果如下图所示：

```

industio@ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux$
industio@ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux$ sudo apt-get install repo git ssh make gcc libssl-dev liblz4-tool \
> expect g++ patchelf chrpath gawk texinfo chrpath diffstat binfmt-support \
> qemu-user-static live-build bison flex fakeroot cmake \
> unzip device-tree-compiler python-pip libncurses-dev expect
[sudo] password for industio:
正在读取软件包列表... 完成
正在分析软件包的依赖关系树... 完成
正在读取状态信息... 完成
binfmt-support 已经是最新版 (2.2.1-2)。
bison 已经是最新版 (2:3.8.2+dfsg-1build1)。
device-tree-compiler 已经是最新版 (1.6.1-1)。
diffstat 已经是最新版 (1.64-1build2)。
fakeroot 已经是最新版 (1.28-1ubuntu1)。
flex 已经是最新版 (2.6.4-8build2)。
g++ 已经是最新版 (4:11.2.0-1ubuntu1)。
gcc 已经是最新版 (4:11.2.0-1ubuntu1)。
make 已经是最新版 (4.3-4.1build1)。
chrpath 已经是最新版 (0.16-2)。
expect 已经是最新版 (5.45.4-2build1)。
liblz4-tool 已经是最新版 (1.0.3-2build2)。
patchelf 已经是最新版 (0.14.3-1)。
cmake 已经是最新版 (3.22.1-1ubuntu1.22.04.2)。
gawk 已经是最新版 (1:5.1.0-1ubuntu0.1)。
git 已经是最新版 (1:2.34.1-1ubuntu1.10)。
libncurses-dev 已经是最新版 (6.3-2ubuntu0.1)。
libssl-dev 已经是最新版 (3.0.2-0ubuntu1.15)。
live-build 已经是最新版 (3.0~a57-1ubuntu41.22.04.1)。
ssh 已经是最新版 (1:8.9p1-3ubuntu0.7)。
unzip 已经是最新版 (6.0-2ubuntu3.2)。
qemu-user-static 已经是最新版 (1:6.2+dfsg-2ubuntu6.19)。
下列软件包是自动安装的并且现在不需要了：
  apturl-common gir1.2-goa-1.0 gir1.2-snapd-1 hplip-data javascript-common libasound2-dev libblkid-dev libcares-dev libexpat1-dev libffi-dev libgl-dev libgl2.0-dev bin libgl1-mesa-dev libglx-dev
  libhpmud0 libimagequant0 libjs-jquery libjs-sphinxdoc libjs-underscore libmount-dev libpcres16-3 libpcres2-16-0 libpcres2-dev libpcres2-posix3 libpcres3-dev libpcres32-3 libpcrespp0v5 libpng-dev libpng-tools
  libpython3-dev libpython3.10-dev libraqm0 libsane-hpaio libsd11.2deb11 libselinux1-dev libsepol-dev libslang2-dev libtext-dev printer-driver-hpcups printer-driver-postscript-hp python3-dateutil
  python3-debconf python3-dev python3-olefile python3-renderpm python3-reportlab-accell python3-software-properties python3-wheel python3.10-dev sgml-base software-properties-common
  ubuntu-advantage-desktop-daemon unattended-upgrades update-notifier-common uuid-dev xml-core
使用 'sudo apt autoremove' 来卸载它(它们)。
将会同时安装下列软件：
  build-essential dpkg-dev libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libauthen-sasl-perl libdata-dump-perl libdpkg-perl libfile-fcntllock-perl libfile-listing-perl
  libfont-afm-perl libhtml-form-perl libhtml-format-perl libhtml-tree-perl libhttp-cookies-perl libhttp-daemon-perl libhttp-negotiate-perl libio-socket-ssl-perl liblwp-protocol-https-perl
  libmailtools-perl libnet-http-perl libnet-smtp-ssl-perl libnet-ssleay-perl libtext-unidecode-perl libwww-perl libwww-robotrules-perl libxml-libxml-perl libxml-namespacesupport-perl libxml-parser-perl
  libxml-sax-base-perl libxml-sax-expat-perl libxml-sax-perl
建议安装：
  debian-keyring libdigest-hmac-perl libgssapi-perl bzip2 libcrypt-ssleay-perl libauthen-ntlm-perl libxml-sax-expatxs-perl texlive-base texlive-latex-base texlive-plain-generic texlive-fonts-recommended
推荐安装：
  python2-dev
下列软件包将被【卸载】：
  python3-pip
下列【新】软件包将被安装：
  build-essential dpkg-dev libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libauthen-sasl-perl libdata-dump-perl libdpkg-perl libfile-fcntllock-perl libfile-listing-perl
  libfont-afm-perl libhtml-form-perl libhtml-format-perl libhtml-tree-perl libhttp-cookies-perl libhttp-daemon-perl libhttp-negotiate-perl libio-socket-ssl-perl liblwp-protocol-https-perl
  libmailtools-perl libnet-http-perl libnet-smtp-ssl-perl libnet-ssleay-perl libtext-unidecode-perl libwww-perl libwww-robotrules-perl libxml-libxml-perl libxml-namespacesupport-perl libxml-parser-perl
  libxml-sax-base-perl libxml-sax-expat-perl libxml-sax-perl python-pip repo texinfo
升级了 0 个软件包，新安装了 35 个软件包，要卸载 1 个软件包，有 4 个软件包未被升级。
需要下载 2,375 kB/5,766 kB 的归档。
解压后会消耗 21.8 MB 的额外空间。
您希望继续执行吗？ [Y/n] Y

```

3 Linux SDK编译

3.1 配置选择

配置命令如下：

▼ Shell

```
1 //进入sdk目录
2 $ cd rk3562-Linux
3 $ ./build.sh lunch
4 Pick a defconfig:
5
6 1. rockchip_defconfig
7 2. rockchip_rk3562_evb3562_v1b_lvds_defconfig
8 3. rockchip_rk3562_evb3562_v1b_mipi_defconfig
9 Which would you like? [1]:
```

根据屏幕（MIPI/LVDS）配置，输入编号以选择对应的配置，如需要编译MIPI屏的固件，则选择3。如下图所示：

```
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423$ cd rk3562_linux/
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux$ ./build.sh lunch
Log saved at /home/industio/Industio-RK3562_Linux5.10_230423/rk3562_linux/output/log/2024-05-29_16-12-51
Pick a defconfig:
1. rockchip_defconfig
2. rockchip_rk3562_evb3562_v1b_lvds_defconfig
3. rockchip_rk3562_evb3562_v1b_mipi_defconfig
Which would you like? [1]: 3
```

另外，还按需求选择文件系统，命令如下：

▼ Shell

```
1 #Buildroot系统
2 $ export RK_ROOTFS_SYSTEM=buildroot
3 #debian10系统
4 $ export RK_ROOTFS_SYSTEM=debian
5 #ubuntu20.04系统
6 $ export RK_ROOTFS_SYSTEM=ubuntu
```

3.2 开始编译

3.2.1 完整编译

首次编译请在sdk顶层目录执行build.sh

注意：不要使用sudo编译，否则会导致编译失败，命令如下：

▼ Shell

```
1 $ ./build.sh
```

编译完成，如下图所示：

```
Make firmware OK!
----- OK -----
*****rkImageMaker ver 2.23*****
Generating new image, please wait...
Writing head info...
Writing boot file...
Writing firmware...
Generating MD5 data...
MD5 data generated successfully!
New image generated successfully!
Running mk-updateimg.sh - build_updateimg succeeded.
Running 99-all.sh - build_all succeeded.
=====
Start saving images and build info
=====
```

在SDK目录下的rockdev/目录生成完整升级固件update.img，如下图所示：

```
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux$ cd rockdev
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux/rockdev$ ls
boot.img  MiniLoaderAll.bin  misc.img  oem.img  parameter.txt  recovery.img  rootfs.img  uboot.img  update.img  userdata.img
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux/rockdev$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux/rockdev$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux/rockdev$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux/rockdev$
industio@Ubuntu22:~/Industio-RK3562_Linux5.10_230423/rk3562_linux/rockdev$
```

3.2.2 编译uboot/kernel

在调试时，可单独编译uboot或kernel部分，命令如下：

```
1 //编译uboot
2 $ ./build.sh uboot
3
4 //编译kernel
5 $ ./build.sh kernel
```

编译uboot结果，如下图所示：

```
*****boot_merger ver 1.31*****
Info:Pack loader ok.
creating new idblock from loader...
idblock binary saving at idblock.img
pack loader(SPL) okay! Input: /home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/rkbin/RKB00T/RK3562MINIALL.ini
/home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/u-boot
pack loader with new: spl/u-boot-spl.bin
Image(no-signed, version=0): uboot.img (FIT with uboot, trust...) is ready
Image(no-signed): rk3562_spl_loader_v1.04.103.bin (with spl, ddr...) is ready
pack uboot.img okay! Input: /home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/rkbin/RKTRUST/RK3562TRUST.ini
Platform RK3562 is build OK, with new .config(make rk3562_defconfig -j14)
/home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/prebuilts/gcc/linux-x86/aarch64/gcc-arm-10.3-2021.07-x86_64-aarch64-none-linux-gnu/bin/aarch64-none-linux-gn
U-
Wed May 8 09:48:13 CST 2024
+ cd ..
Running 20-loader.sh - build_uboot succeeded.
```

编译kernel结果，如下图所示：

```

make: Entering directory '/home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/kernel-5.10'
# No change to .config
#
make: Leaving directory '/home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/kernel-5.10'
+ make -C kernel/ -j8 CROSS_COMPILE=/home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/prebuilts/gcc/linux-x86/aarch64/gcc-arm-10.3-2021.07-x86_64-aarch64-none
-linux-gnu/bin/aarch64-none-linux-gnu- ARCH=arm64 ido-evb3562-v1b-dsi-mipi.img
make: Entering directory '/home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/kernel-5.10'
CALL scripts/atomic/check-atomics.sh
CALL scripts/checksyscalls.sh
CHK include/generated/compile.h
Image: resource.img (with ido-evb3562-v1b-dsi-mipi.dtb logo.bmp logo_kernel.bmp) is ready
Image: boot.img (with Image resource.img) is ready
Image: zboot.img (with Image lz4 resource.img) is ready
make: Leaving directory '/home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/kernel-5.10'
+ /home/industio/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux/device/rockchip/common/scripts/mk-fitimage.sh kernel/boot.img boot.its kernel/arch/arm64/boot/Image
FIT description: U-Boot FIT source file for arm
Created: Wed May 8 09:53:14 2024
Image 0 (fdt)
Description: unavailable
Created: Wed May 8 09:53:14 2024
Type: Flat Device Tree
Compression: uncompressed
Data Size: 83440 Bytes = 81.48 KiB = 0.08 MiB
Architecture: AArch64
Load Address: 0xffffffff00
Hash algo: sha256
Hash value: 9e65416f4c5f0820927c3a05214de76ca0bc187ede61b9124500b1839414a5f0
Image 1 (kernel)
Description: unavailable
Created: Wed May 8 09:53:14 2024
Type: Kernel Image
Compression: uncompressed
Data Size: 35222016 Bytes = 34396.50 KiB = 33.59 MiB
Architecture: AArch64
OS: Linux
Load Address: 0xffffffff01
Entry Point: 0xffffffff01
Hash algo: sha256
Hash value: 167e32774598805a5a39a41a814e179cc11ec34ace763765d4d2ff3ac8d1f1c
Image 2 (resource)
Description: unavailable
Created: Wed May 8 09:53:14 2024
Type: Multi-File Image
Compression: uncompressed
Data Size: 121344 Bytes = 118.50 KiB = 0.12 MiB
Hash algo: sha256
Hash value: 066f11cc3e4bfa90bc7e090a28f931d099607ccf2dd76b2784e80fabbf9f50d5
Default configuration: 'conf'
Configuration 0 (conf)
Description: unavailable
Kernel: kernel
FDT: fdt
Not Found io-domains in kernel/arch/arm64/boot/dts/rockchip/ido-evb3562-v1b-dsi-mipi.dts
Running io-kernel.sh - build kernel succeeded.
industio@ubuntu22:~/RK3562/Linux_SDK/Industio-RK3562_Linux5.10_230423/rk3562_linux$

```

编译完成后，将在rockdev/目录生成对应的uboot.img和boot.img。

4 Buildroot QT SDK使用方法

以下链接下载QT SDK包：

链接：https://pan.baidu.com/s/1f-ol5_c5BLG_syU149XK1w?pwd=1234

提取码：1234

4.1 安装QT-SDK包

把Industio-RK3562_Buildroot2012_QT5.15.8_230423文件，拷贝到开发主机的自定义目录，命令如下：

```

1 #进入目录下
2 $ cd Industio-RK3562_Buildroot2012_QT5.15.8_230423
3 #解压Industio-RK3562_Buildroot2012_QT5.15.8.tar.gz
4 $ tar -xvf Industio-RK3562_Buildroot2012_QT5.15.8.tar.gz

```

结果如下图所示：

```

industio@Ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423$ ls
Industio-RK3562_Buildroot2012_QT5.15.8.tar.gz Patch
industio@Ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423$ tar -xvf Industio-RK3562_Buildroot2012_QT5.15.8.tar.gz
host/
host/environment-setup
host/relocate-sdk.sh
host/bin/
host/bin/xsltproc
host/bin/tracegen
host/bin/pod2usage
host/bin/pcgrep
host/bin/mk_cmds
host/bin/aarch64-linux-cpp
host/bin/enc2xs
host/bin/autom4te
host/bin/aarch64-linux-gprof
host/bin/lslocks
host/bin/aarch64-linux-gcc.br_real
host/bin/toe
host/bin/flex++
host/bin/dmesg
host/bin/yacc
host/bin/lsipc
host/bin/aarch64-linux-nm
host/bin/autopoint
host/bin/hardlink
host/bin/gawk
host/bin/chatr
host/bin/qmlmin
host/bin/aarch64-buildroot-linux-gnu-lto-dump
host/bin/uuidgen
host/bin/ntfsfix
host/bin/pkg-config
host/bin/captoinfo
host/bin/2to3-3.10
host/bin/libtoolize
host/bin/cal
host/bin/perl5.32.1
host/bin/python3-config
host/bin/xml2-config
host/bin/gperf
host/bin/chacl
host/bin/h2xs
host/bin/setuid
host/bin/m4
host/bin/aarch64-linux-gcc-11.3.0
host/bin/ptargrep
host/bin/instmodsh
host/bin/autoupdate
host/bin/makedevs
host/bin/aarch64-buildroot-linux-gnu-gcc-11.3.0
host/bin/cjpeg
host/bin/2to3
host/bin/aarch64-buildroot-linux-gnu-gcc.br_real
host/bin/aarch64-buildroot-linux-gnu-gcc-nm
host/bin/reset

```

解压完成后，将解压目录host重命名rk3562_buildroot_qt_sdk，命令如下：



Plain Text |

```
1 $ mv host/ rk3562_buildroot_qt_sdk
```

结果如下图所示：

```

industio@Ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423$ mv host/ rk3562_buildroot_qt_sdk
industio@Ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423$ 
industio@Ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423$ ls
Industio-RK3562_Buildroot2012_QT5.15.8.tar.gz Patch rk3562_buildroot_qt_sdk
industio@Ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423$ 
industio@Ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423$

```

进入buildroot_qt5.14.2_sdk目录，执行relocate-sdk.sh，命令如下：

```

1 $ cd rk3562_buildroot_qt_sdk/
2 $ ./relocate-sdk.sh
3 Relocating the buildroot SDK from /mnt/work2/tanwx/rockchip/rk356x/sdk_from_
  _ido3568/rk356x_linux_sdk/buildroot/output/rockchip_rk3568/host to /home/ro
  nnie/rk3562_buildroot_qt_sdk ...

```

结果如下图所示：

```

industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423$ cd rk3562_buildroot_qt_sdk/
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$ ls
aarch64-buildroot-linux-gnu bin doc environment-setup etc include lib lib64 libexec mkspeccs relocate-sdk.sh sbin share usr
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$ ./relocate-sdk.sh
Relocating the buildroot SDK from /mnt/work2/tanwx/rockchip/rk356x/sdk_from_ido3568/rk356x_linux_sdk/buildroot/output/rockchip_rk3568/host to /home/industio/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk ...
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$

```

把rk3562_buildroot_qt_sdk/bin加入到PATH中，命令如下：

```

1 #查看当前路径
2 $ pwd
3 #把rk3562_buildroot_qt_sdk/bin加入到PATH中
4 $ echo "export PATH=/home/industio/Industio-RK3562_Buildroot2012_QT5.15.8_2
  30423/rk3562_buildroot_qt_sdk/bin:${PATH}" >> ~/.bashrc
5 #立即生效
6 $ source ~/.bashrc

```

结果如下图所示：

```

industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$ pwd
/home/industio/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$ echo "export PATH=/home/industio/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk/bin:${PATH}" >> ~/.bashrc
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$ source ~/.bashrc
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$

```

确认安装成功，命令如下：

```

1 $ which qmake
2 /home/industio/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildro
  ot_qt_sdk/bin/qmake
3 $ qmake -v
4 QMake version 3.1
5 Using Qt version 5.15.8 in /home/industio/Industio-RK3562_Buildroot2012_QT

```

结果如下图所示：

```

industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$ which qmake
/home/industio/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk/bin/qmake
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$
industio@ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$ qmake -v
QMake version 3.1
Using Qt version 5.15.8 in /home/industio/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk/aarch64-buildroot-linux-gnu/sysroot/usr/lib

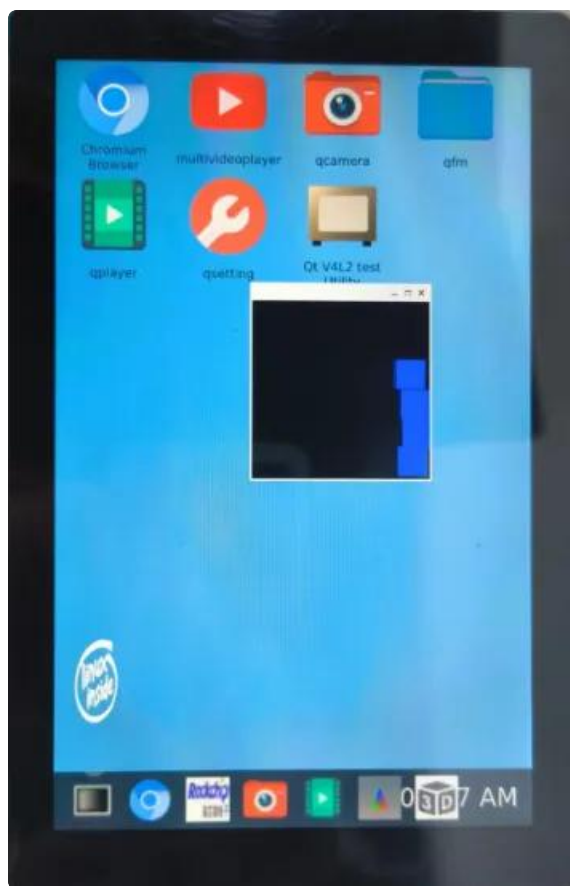
```


4.3 运行QT程序

把刚才编译的moveblocks程序拷贝到开发板中，然后执行它，命令如下：

```
1 [root@RK356X:/]# chmod a+x moveblocks
2 [root@RK356X:/]# ./moveblocks
3 QStandardPaths: wrong permissions on runtime directory /var/run, 7755 instead of 7700
```

运行成功后，屏幕将显示QT程序在运行，结果如下图所示：



5 Ubuntu QT SDK使用方法

到以下链接下载RK3562 ubuntu20 qt5.15 sdk包。

链接：<https://pan.baidu.com/s/1EBHO897HnLzeXph-OrpI3A?pwd=1234>

提取码：1234

5.1 安装QT-SDK包

注：此sdk要求Ubuntu22.04及以上版本系统，否则安装失败。

把前面下载的rk3562_ubuntu20_qt5.15_sdk.tar.gz拷贝到电脑Ubuntu22.04系统中，并解压：

```
1 $ tar -zxvf rk3562_ubuntu20_qt5.15_sdk.tar.gz
```

进入解压目录执行install_sdk.sh（安装到/opt目录，需要sudo权限执行）：

```
1 $ cd rk3562_ubuntu20_qt5.15_sdk
2 $ sudo ./install_sdk.sh
3 $ echo "PATH=/opt/gcc-arm-9.2-2019.12-x86_64-aarch64-none-linux-gnu/bin:$PATH" >> ~/.bashrc
4 $ echo "PATH=/opt/qt-everywhere-src-5.15.8/host/bin:$PATH" >> ~/.bashrc
5 $ source ~/.bashrc
```

确认安装是否成功：

```
1 $ qmake -v
2 QMake version 3.1
3 Using Qt version 5.15.8 in /opt/qt-everywhere-src-5.15.8/ext/lib
```

5.2 测试

以moveblocks为例，可以在以下链接下载：

链接：https://pan.baidu.com/s/10yYA-H6OkXe_U6G_5eq9kA?pwd=nuhb

提取码：[nuhb](#)

解压moveblock.tar，并进入解压目录：

```
1 $ tar -xvf moveblock.tar
2 $ cd moveblock
```

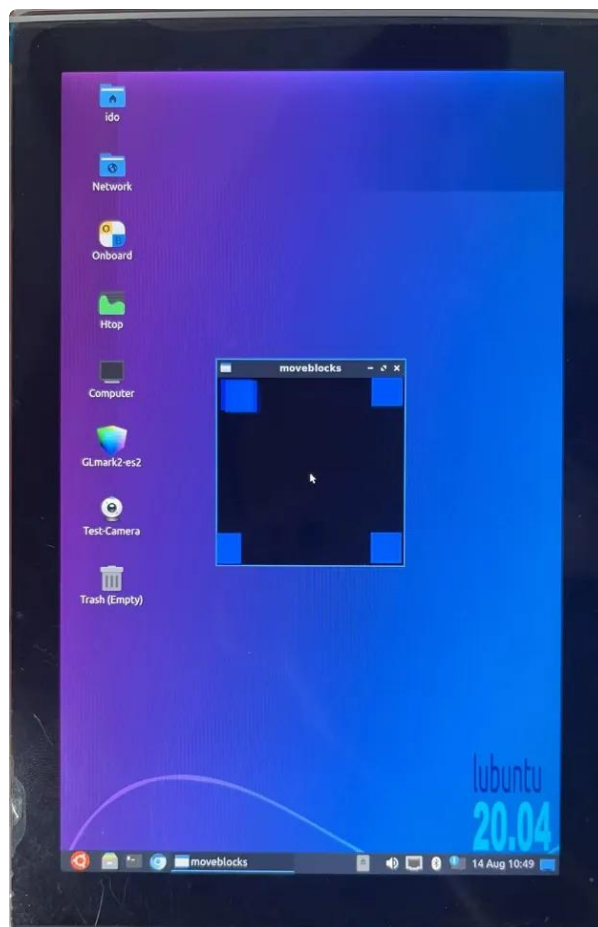
执行qmake和make编译：

```
1 $ qmake .  
2 $ make -j4
```

编译完成，得到moveblock.可执行文件，将其拷贝到主板上运行。

```
1 root@ido:~# export QT_QPA_PLATFORM_PLUGIN_PATH=/usr/local/qt5/plugins  
2 root@ido:~# export QT_QPA_PLATFORM=xcb  
3 root@ido:~# export LD_LIBRARY_PATH=/usr/local/qt5/lib:$LD_LIBRARY_PATH  
4 root@ido:~# chmod a+x ./moveblock.  
5 root@ido:~# ./moveblock
```

运行成功后，屏幕将显示QT程序在运行，结果如下图所示：



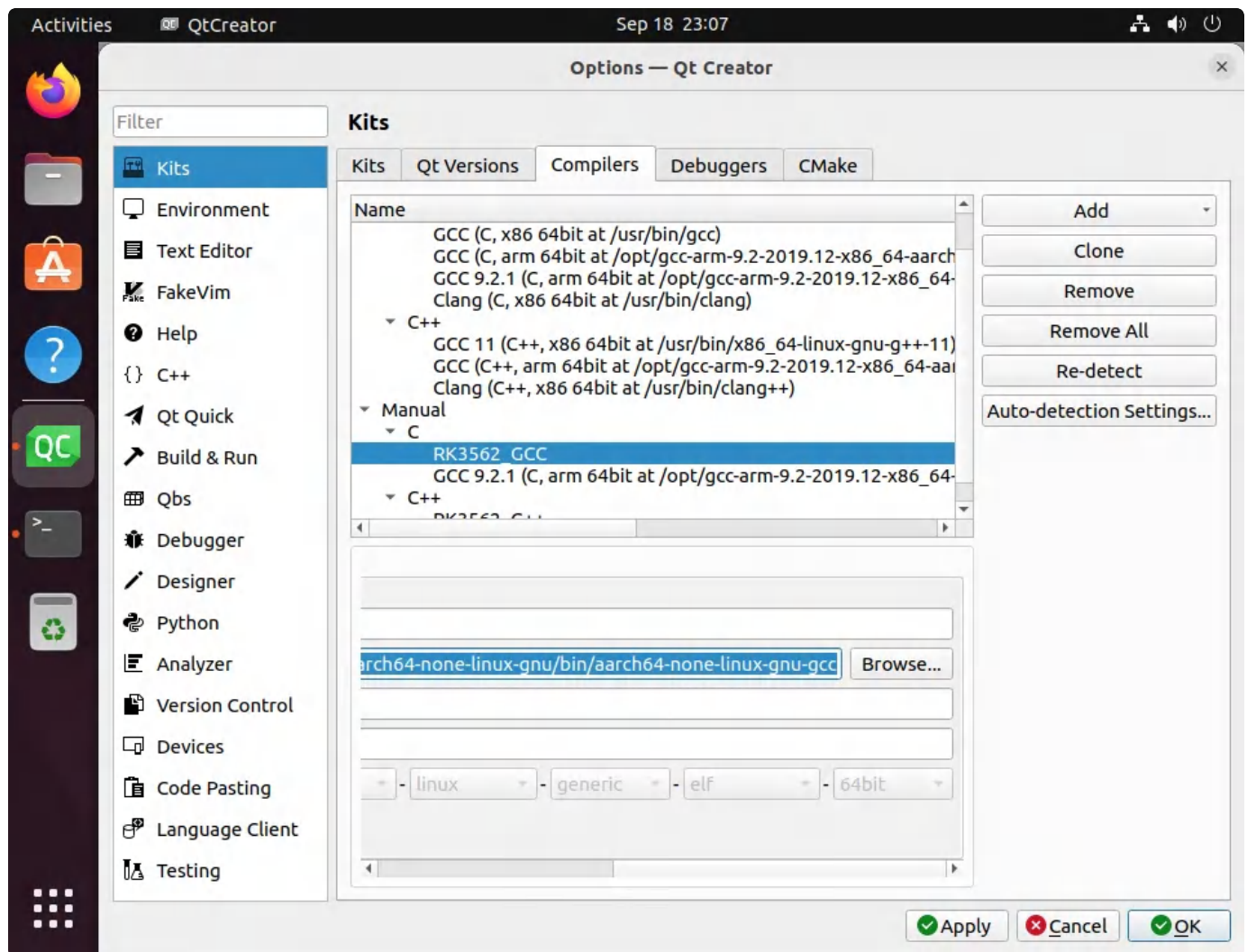
5.3 Qt Creator 配置Kit

软件下载：

https://download.qt.io/archive/online_installers/4.8/qt-online-installer-linux-x64-4.8.0.run

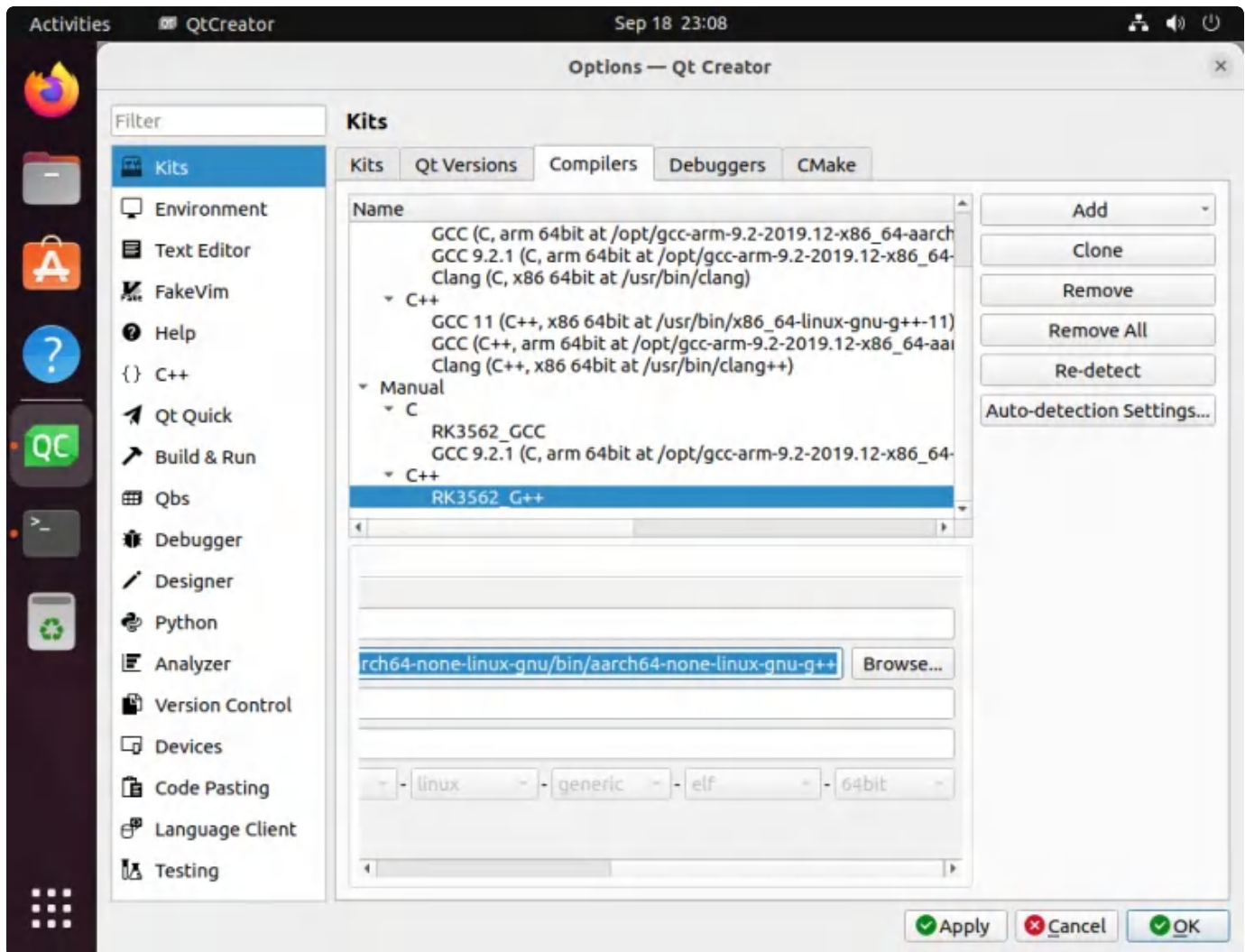
1. GCC 选择

```
▼ Bash |
1 /opt/gcc-arm-9.2-2019.12-x86_64-aarch64-none-linux-gnu/bin/aarch64-none-lin
  ux-gnu-gcc
```



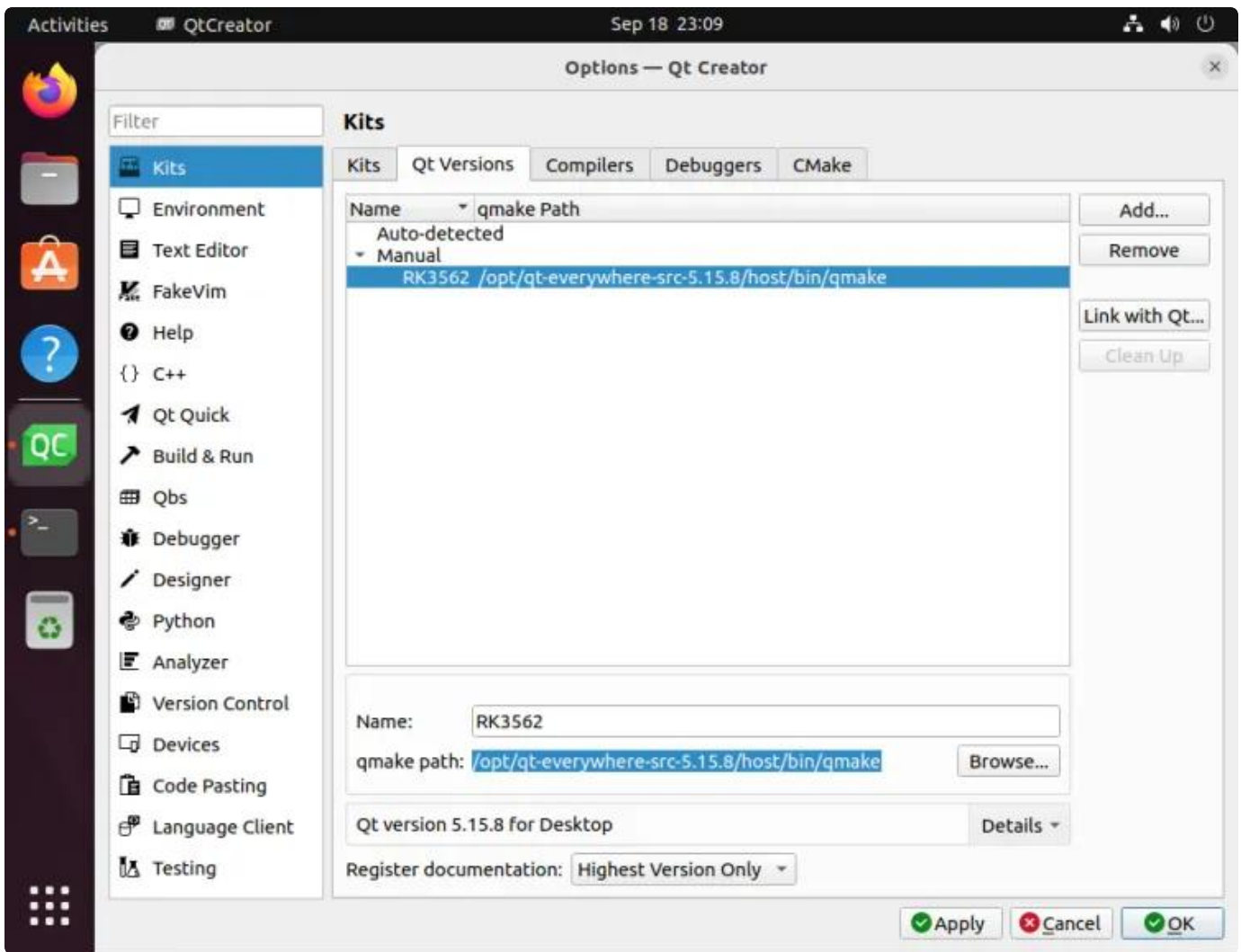
2. G++ 选择

```
1 /opt/gcc-arm-9.2-2019.12-x86_64-aarch64-none-linux-gnu/bin/aarch64-none-linux-gnu-gcc
```



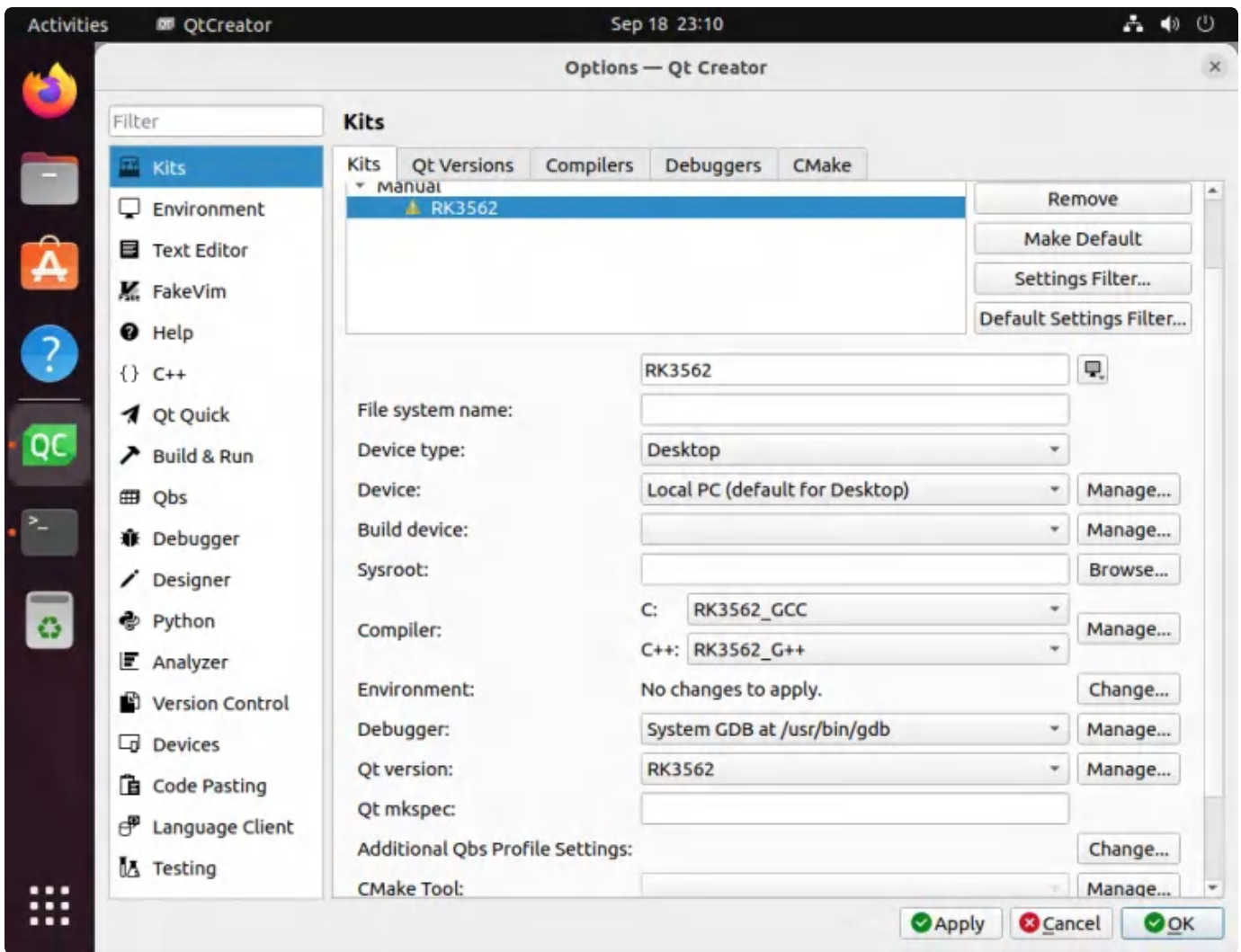
3. qmake 选择

```
1 /opt/qt-everywhere-src-5.15.8/host/bin/qmake
```



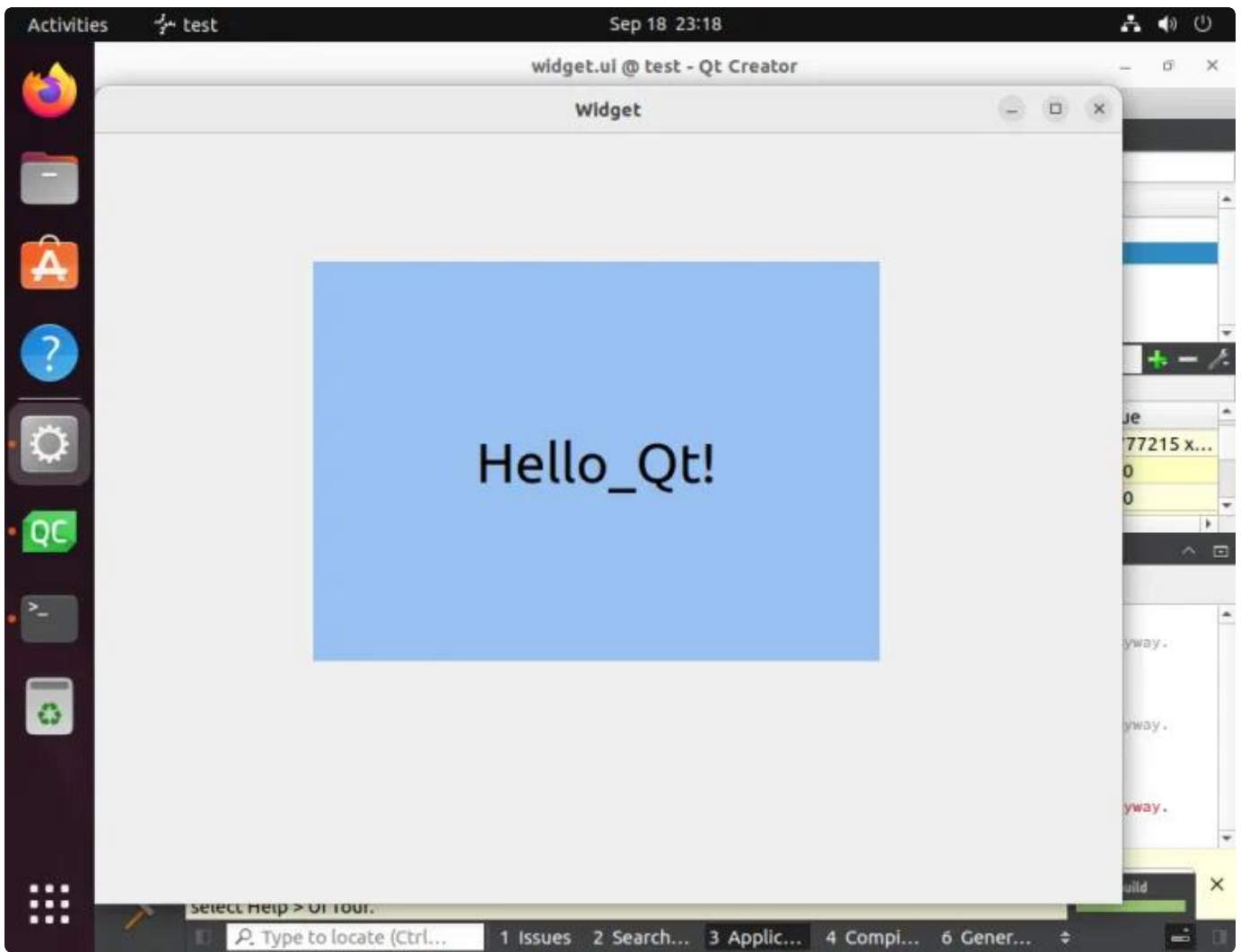
4. Kits 配置

新增一个名为RK3562的kit， 分别选择前面的gcc、g++、QtVersion

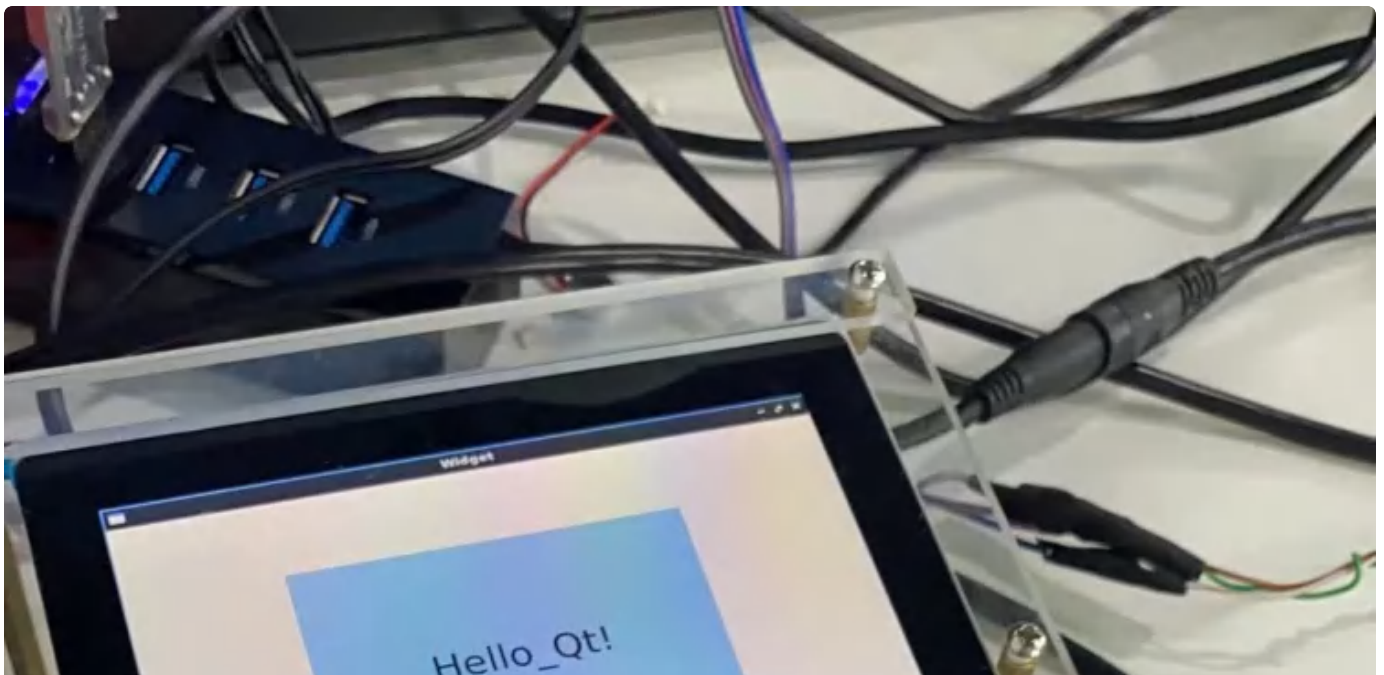


5. 创建测试工程及运行测试

新建一个Qt Widgets Application，命令为 `test`



将编译后的二进制文件拷贝至开发板中，在开发板上执行：





注：如果执行在开发板上的可执行文件不显示字符，则需要执行包含环境变量的脚本，如下：

▼

Bash |

```
1  source /usr/local/qt5/env.sh
```