IDO-EVB3562-V1 Linux 开发手册

IDO–EVB3562–V1

Linux 开发手册

文档修订历史

版本	修订内容	修订	审核	日期
V1.0	创建文档	TWX	IDO	2023/09/06
V1.2	文档优化	LZR	IDO	2024/04/17

1 Linux SDK下载

1.1 源码下载

从以下地址下载IDO-EVB3562的Linux_SDK：

链接：https://pan.baidu.com/s/1meK4OH8Gg6ZXGy37icSRzw?pwd=1234

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
$ cat Industio-RK3562_Linux5.10_0423.a* > Industio-RK3562_Linux-SDK.tar.gz
```

结果如下图所示：

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：

```Shell
$ 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：

Shell

```shell
$ 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
```

结果如下图所示：

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    #debian系统
4    $ export RK_ROOTFS_SYSTEM=debian
```

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$
```

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：

```Shell
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/RKBOOT/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-gn
u-
Wed May  8 09:48:13 CST 2024
+ cd ..
Running 20-loader.sh - build_uboot succeeded.
```

编译内核结果，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: 0xffffff00
  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: 0xffffff01
  Entry Point:  0xffffff01
  Hash algo:    sha256
  Hash value:   167e32774598805a5a39a41a814e179cc11cec34ace763765d4d2ff3ac8d1f1c
 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:   066f11cc3e4bfa98bc7e090a28f931d099607ccf2dd76b2784e80fabbf9f50d5
 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 10-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**kernel.img**。

4 QT-SDK下载

以下链接下载QT SDK包：

链接： https://pan.baidu.com/s/1Tkyc6_2TC7CPHEpZIZpSyg?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
```

Plain Text

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/pcregrep
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/chattr
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/setsid
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：

```
$ 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：

```bash
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  mkspecs  relocate-sdk.sh  sbin  share  usr
industio@Ubuntu22:~/Industio-RK3562_Buildroot2012_QT5.15.8_230423/rk3562_buildroot_qt_sdk$ ./relocate-sdk.sh
Relocating the buildroot SDK from /mnt/work1/cainiaocl/RK/RK3562/rk3562_linux/buildroot/output/rockchip_rk3562/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$ 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
```

4.2 编译QT程序

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

把moveblocks.tar拷贝到开发主机下，这里直接考到rk3562_buildroot_qt_sdk目录下，并解压moveblocks.tar，命令如下：

```
1   $ tar -xvf moveblocks.tar
2   ./moveblocks/
3   ./moveblocks/.obj/
4   ./moveblocks/main.cpp
5   ./moveblocks/.moc/
6   ./moveblocks/.pch/
7   ./moveblocks/moveblocks.pro
```

结果如下图所示:

进入解压目录，执行qmake & make ，命令如下：

```
1   $ cd moveblocks/
2   $ qmake .
3   $ make
```

编译完成，会生成moveblocks可执行程序，结果如下图所示：

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：

```
Plain Text

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