

IDO-EVB3020 Linux Buildroot应用开发说明

1. 安装交叉编译工具

1.1 交叉编译工具获取

1.2 配置编译环境

配置环境

验证环境是否配置成功

2. 交叉编译Qt程序

命令行编译 Qt 项目

QtCreator配置kit

运行交叉编译后的可执行程序



IDO-EVB3020 Linux Buildroot应用开发说明

深圳触觉智能科技有限公司

www.industio.cn

文档修订历史

版本	修订内容	修订	审核	日期
V1.0	创建文档	FuYingz he		2023/03/ 03

1. 安装交叉编译工具

1.1 交叉编译工具获取

针对使用提供的默认Buildroot固件的用户，可以从以下路径下载交叉编译工具链，此工具链中包含了默认Buildroot固件所包含的必要编译工具和已编译的Qt5.12模块库文件。

下载路径

链接：https://pan.baidu.com/s/1sCt2_zByv3qKL-rAkB5wJQ?pwd=1234

提取码：1234

将下载文件解压至PC端 Ubuntu 的 /opt/industio/ 目录

```
▼ Java |  
1  mkdir -p /opt/industio  
2  tar -xzf px30-buildroot-qt5.12-sdk.tar.gz -C /opt/industio
```

1.2 配置编译环境

配置环境

执行source命令将交叉编译工具的路径添加至当前shell终端环境变量中

```
▼ Java |  
1  $ source /opt/industio/px30-buildroot-qt5.12-sdk/sdk_env_setup
```

如果不希望每次新开一个终端都重新配置，可以将环境变量添加到用户配置文件 ~/.bashrc 中

```
1 echo ". /opt/industio/px30-buildroot-qt5.12-sdk/sdk_env_setup" >> ~/.bashrc
```

验证环境是否配置成功

```
1 $ qmake -v
2 QMake version 3.1
3 Using Qt version 5.12.2 in /opt/industio/px30-buildroot-qt5.12-sdk/aarch64
  -buildroot-linux-gnu/sysroot/usr/lib
4 $ aarch64-linux-gcc -v
5 Using built-in specs.
6 COLLECT_GCC=/opt/industio/px30-buildroot-qt5.12-sdk/bin/aarch64-linux-gcc.
  br_real
7 COLLECT_LTO_WRAPPER=/opt/industio/px30-buildroot-qt5.12-sdk/bin/./libexec
  /gcc/aarch64-buildroot-linux-gnu/8.4.0/lto-wrapper
8 Target: aarch64-buildroot-linux-gnu
9 Configured with: ./configure --prefix=/mnt/work2/fuyz/Rockchip/PX30/EVB302
  0/linux/px30_linux/buildroot/output/rockchip_px30_64/host --sysconfdir=/mnt
  /work2/fuyz/Rockchip/PX30/EVB3020/linux/px30_linux/buildroot/output/rockc
  hip_px30_64/host/etc --enable-static --target=aarch64-buildroot-linux-gnu
  --with-sysroot=/mnt/work2/fuyz/Rockchip/PX30/EVB3020/linux/px30_linux/buil
  droot/output/rockchip_px30_64/host/aarch64-buildroot-linux-gnu/sysroot --e
  nable-__cxa_atexit --with-gnu-ld --disable-libssp --disable-multilib --dis
  able-decimal-float --with-gmp=/mnt/work2/fuyz/Rockchip/PX30/EVB3020/linux/
  px30_linux/buildroot/output/rockchip_px30_64/host --with-mpc=/mnt/work2/fu
  yz/Rockchip/PX30/EVB3020/linux/px30_linux/buildroot/output/rockchip_px30_6
  4/host --with-mpfr=/mnt/work2/fuyz/Rockchip/PX30/EVB3020/linux/px30_linux/
  buildroot/output/rockchip_px30_64/host --with-pkgversion='Buildroot 2018.0
  2-rc3-gf46347050-dirty' --with-bugurl=http://bugs.buildroot.net/ --disable
  -libquadmath --enable-tls --enable-threads --without-isl --without-cloog -
  -with-abi=lp64 --with-cpu=cortex-a35 --enable-languages=c,c++ --with-build
  -time-tools=/mnt/work2/fuyz/Rockchip/PX30/EVB3020/linux/px30_linux/buildro
  ot/output/rockchip_px30_64/host/aarch64-buildroot-linux-gnu/bin --enable-s
  hared --disable-libgomp
10 Thread model: posix
11 gcc version 8.4.0 (Buildroot 2018.02-rc3-gf46347050-dirty)
```

2. 交叉编译Qt程序

命令行编译 Qt 项目

```
1 cd /your/qt/project/directory/  
2 qmake  
3 make
```

将编译得到的可执行文件，拷贝至开发板中。

QtCreator配置kit

如果是使用QtCreator界面编译，则需要添加kit配置

GCC Compilers位置: /opt/industio/px30-buildroot-qt5.12-sdk/bin/aarch64-linux-gcc

G++ Compilers位置: /opt/industio/px30-buildroot-qt5.12-sdk/bin/aarch64-linux-g++

qmake位置: /opt/industio/px30-buildroot-qt5.12-sdk/bin/qmake

运行交叉编译后的可执行程序

Buildroot 文件系统 Qt 自启服务文件为: /etc/init.d/S50launcher

将文件中的“/usr/bin/QLauncher &”替换为需要运行的Qt程序文件，并加上& 配置为后台运行。

```
1 #!/bin/sh
2 #
3 # Start linux launcher...
4 #
5
6 # Load default env variables from profiles(e.g. /etc/profile.d/weston.sh)
7 . /etc/profile
8
9 case "$1" in
10     start)
11         printf "Starting launcher: "
12         export LC_ALL='zh_CN.utf8'
13
14         # Uncomment to disable mirror mode
15         # unset WESTON_DRM_MIRROR
16
17         export XDG_RUNTIME_DIR=${XDG_RUNTIME_DIR:-/var/run}
18         export QT_QPA_PLATFORM=${QT_QPA_PLATFORM:-wayland}
19
20         weston --tty=2 --idle-time=0&
21         {
22             # Wait for weston ready
23             while [ ! -e ${XDG_RUNTIME_DIR}/wayland-0 ]; do
24                 sleep .1
25             done
26             #/usr/bin/QLauncher &
27             /your/qt/app &
28         }&
29     ;;
30     stop)
31         killall QLauncher
32         killall weston
33         printf "stop finished"
34     ;;
35     *)
36         echo "Usage: $0 {start|stop}"
37         exit 1
38     ;;
39 esac
40 exit 0
41
```