

荔枝派Zero(全志V3S)运行Qt5程序

文章目录

- 前言
- 一、配置 buildroot
- 二、编译 buildroot
- 三、编译例程
- 四、拷贝到 SD 卡
- 五、上板子运行

前言

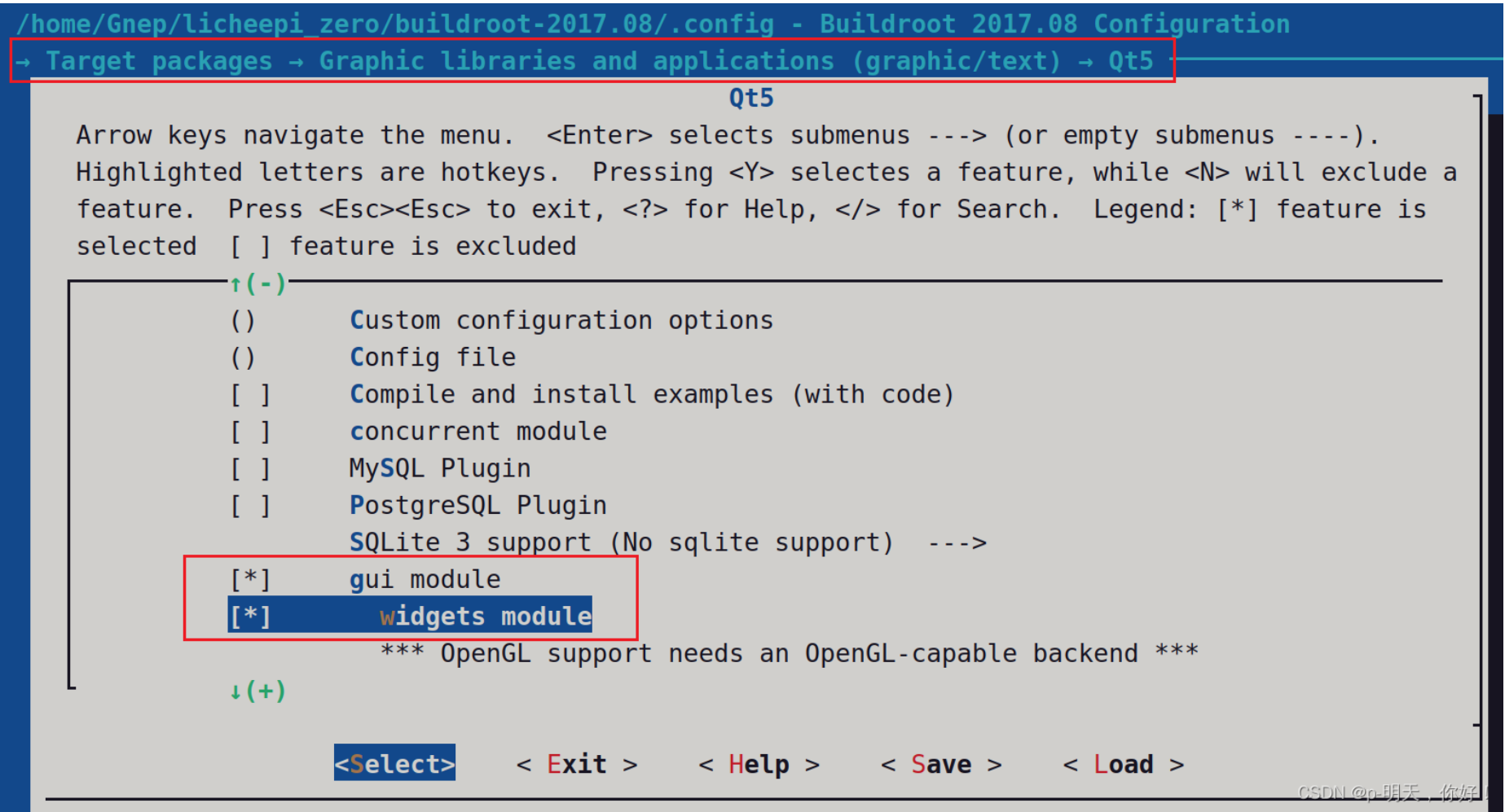
- 本文重新配置 **buildroot** ，利用 buildroot 重新交叉编译 Qt，编译完成后将编译产生的可执行文件拷贝到 SD 卡，板子上电后跑到文件系统下再手动运行。
- 读者需完成 **Uboot** 、**Kernel** 编译，可参考下面，本文只重新配置 rootfs 并重新编译荔枝派Zero(全志V3S)编译Uboot及配置荔枝派Zero(全志V3S)编译Kernel
- 编译 buildroot 时遇到的问题可以参考下面：荔枝派Zero(全志V3S)编译rootfs

一、配置 buildroot

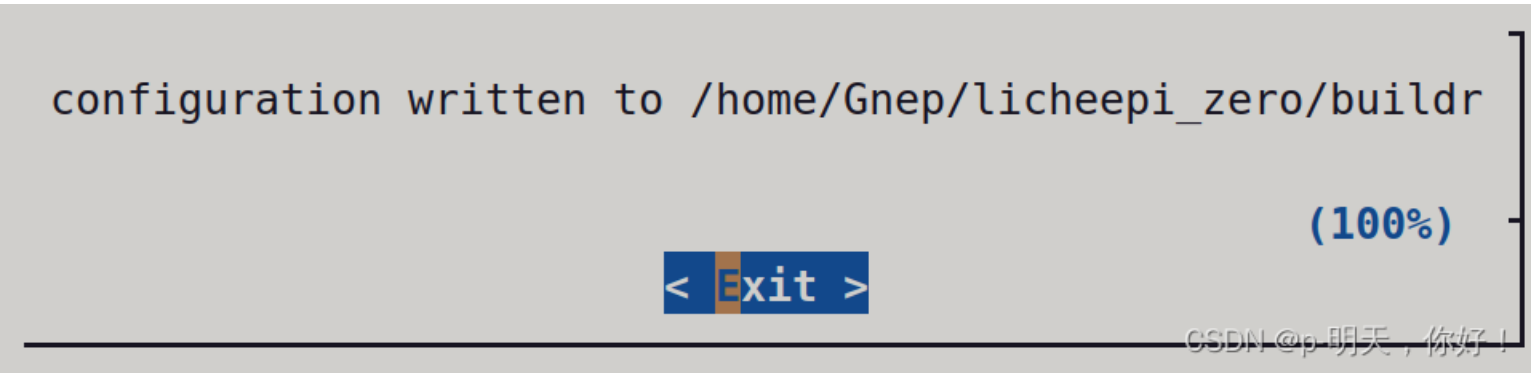
1、在 buildroot-2017.08 根目录下，打开配置界面，命令如下：

```
1 | make menuconfig
```

2、Target packages -> Graphic libraries and applications (graphic/text) -> Qt5 ，勾选上 gui module 和 widgets module



3、保存配置并退出



二、编译 buildroot

回到 buildroot-2017.08 的根目录下，执行 make

三、编译例程

- 确认 qmake 是否正常
可能电脑里面之前就有 qmake，那么必须**指定 qmake 的路径**，再查看 qmake 的版本才可行，否则直接 qmake -v 会显示系统上已存在的 qmake 版本在 buildroot-2017.08 的根目录下执行：

```
1 | output/build/qt5base-5.6.2/bin/qmake -v

Gnep@lpvm:~/licheepi_zero/buildroot-2017.08$ output/build/qt5base-5.6.2/bin/qmake -v
QMake version 3.0 ← 我们要用到的
Using Qt version 5.6.2 in /home/Gnep/licheepi_zero/buildroot-2017.08/output/host/arm-buildroot-linux-gnueabi/fsl/sysroot/usr/lib
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08$ qmake -v
QMake version 2.01a ← 系统上本存在的
Using Qt version 4.8.0 in /opt/gcc-4.4.4-glibc-2.11.1-multilib-1.0/arm-fsl-linux-gnueabi/lib
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08$
```

- 2、编译例程
- ①、选择一个模拟时钟的 demo

```
1 | cd output/build/qt5base-5.6.2/examples/widgets/widgets/analogclock

Gnep@lpvm:~/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/examples/widgets/widgets/analogclock$ ls
analogclock.cpp analogclock.h analogclock.pro main.cpp
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/examples/widgets/widgets/analogclock$
```

再执行以下命令

```
1 | /home/Gnep/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/bin/qmake analogclock.pro
```

如果报下述错误，那就在 buildroot-2017.08 的根目录下先 make clean 以后再执行 make

```
Project ERROR: Unknown module(s) in QT: gui widgets:
```

成功后，会生成一个 Makefile 文件

```
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/examples/widgets/widgets/analogclock$ ls
analogclock.cpp analogclock.h analogclock.pro main.cpp Makefile
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/examples/widgets/widgets/analogclock$
```

再执行 make，生成可执行文件

```
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/examples/widgets/widgets/analogclock$ ls
analogclock analogclock.cpp analogclock.h analogclock.pro main.cpp Makefile
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/examples/widgets/widgets/analogclock$
```

- ②、再选择一个 demo

```
1 | cd ../../animation/stickman/
2 |
3 | /home/Gnep/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/bin/qmake stickman.pro
4 |
5 | make
```

生成可执行文件 stickman

```
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/examples/widgets/animation/stickman$ ls
animation.cpp graphicsview.cpp lifecycle.h node.cpp rectbutton.h stickman.h
animation.h graphicsview.h main.cpp node.h stickman stickman.pro
animations lifecycle.cpp Makefile rectbutton.cpp stickman.cpp stickman.qrc
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08/output/build/qt5base-5.6.2/examples/widgets/animation/stickman$
```

四、拷贝到 SD 卡

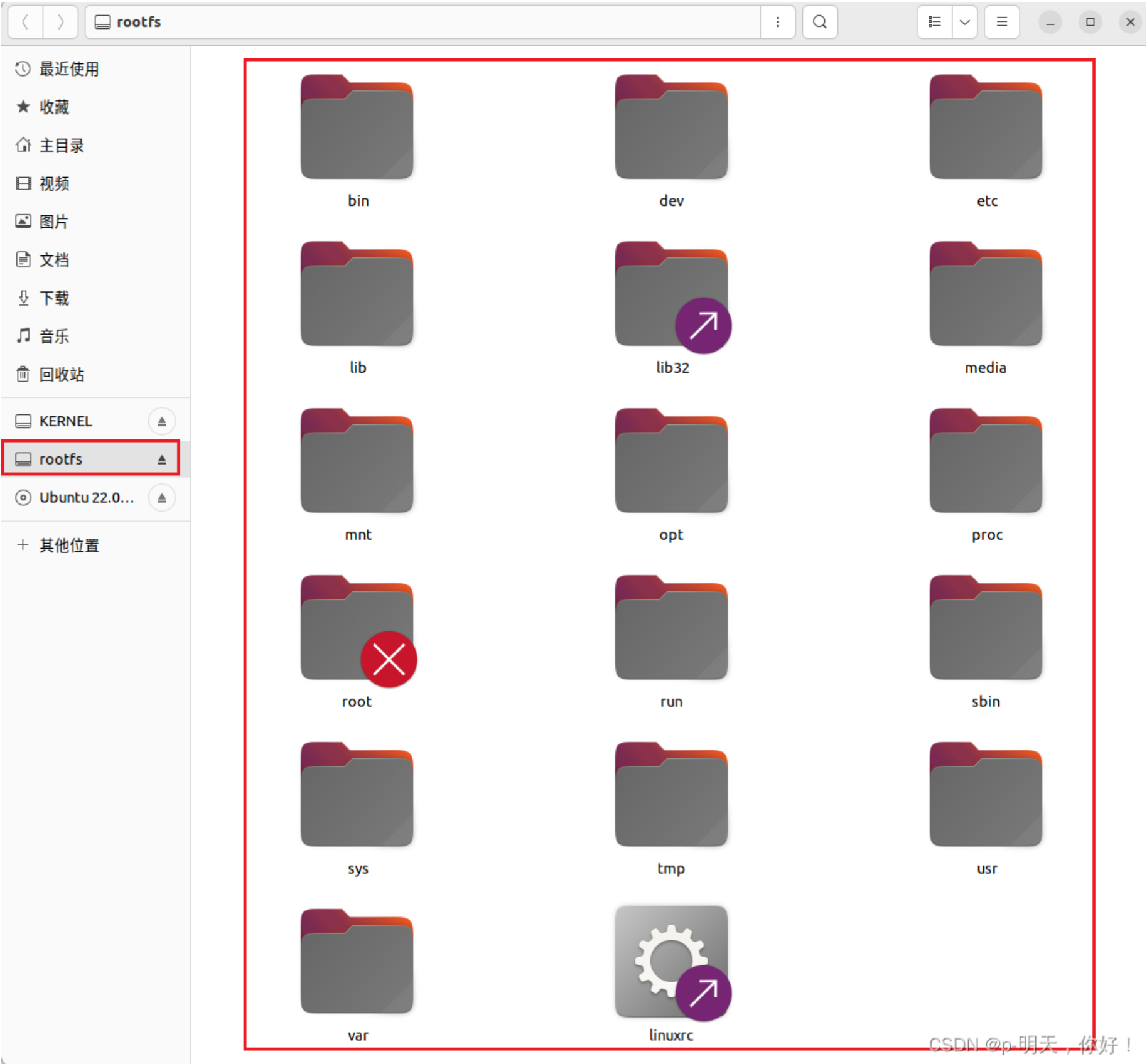
- ①、插上 SD 卡，先把 SD 卡的 rootfs 分区里面的文件删除
- 新打开一个终端

```
1 | sudo rm -rf /media/Gnep/rootfs/*
2 | ls
```

- ②、把 buildroot-2017.08 产生的 rootfs.tar 解压到刚创建的 rootfs 分区根目录
- 在 buildroot-2017.08 根目录下

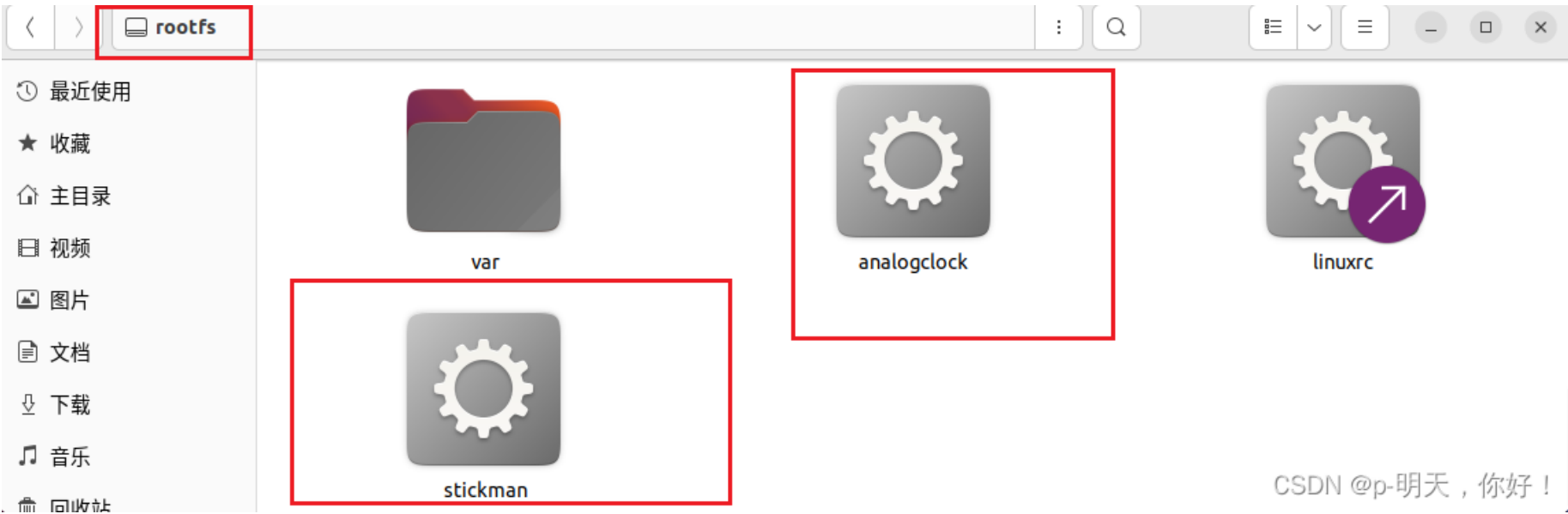
```
1 | find ./ -name rootfs.tar
2 | sudo tar xf ./output/images/rootfs.tar -C /media/Gnep/rootfs/
```

```
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08$ find ./ -name rootfs.tar
./output/images/rootfs.tar
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08$ sudo tar xf ./output/images/rootfs.
tar -C /media/Gnep/rootfs/
Gnep@lpvm:~/licheepi_zero/buildroot-2017.08$
```



③、把刚编译的两个可执行文件拷贝到 SD 卡的 rootfs 分区

```
1 | sudo cp ./output/build/qt5base-5.6.2/examples/widgets/widgets/analogclock/analogclock /media/Gnep/rootfs/
2 |
3 | sudo cp ./output/build/qt5base-5.6.2/examples/widgets/animation/stickman/stickman /media/Gnep/rootfs/
```

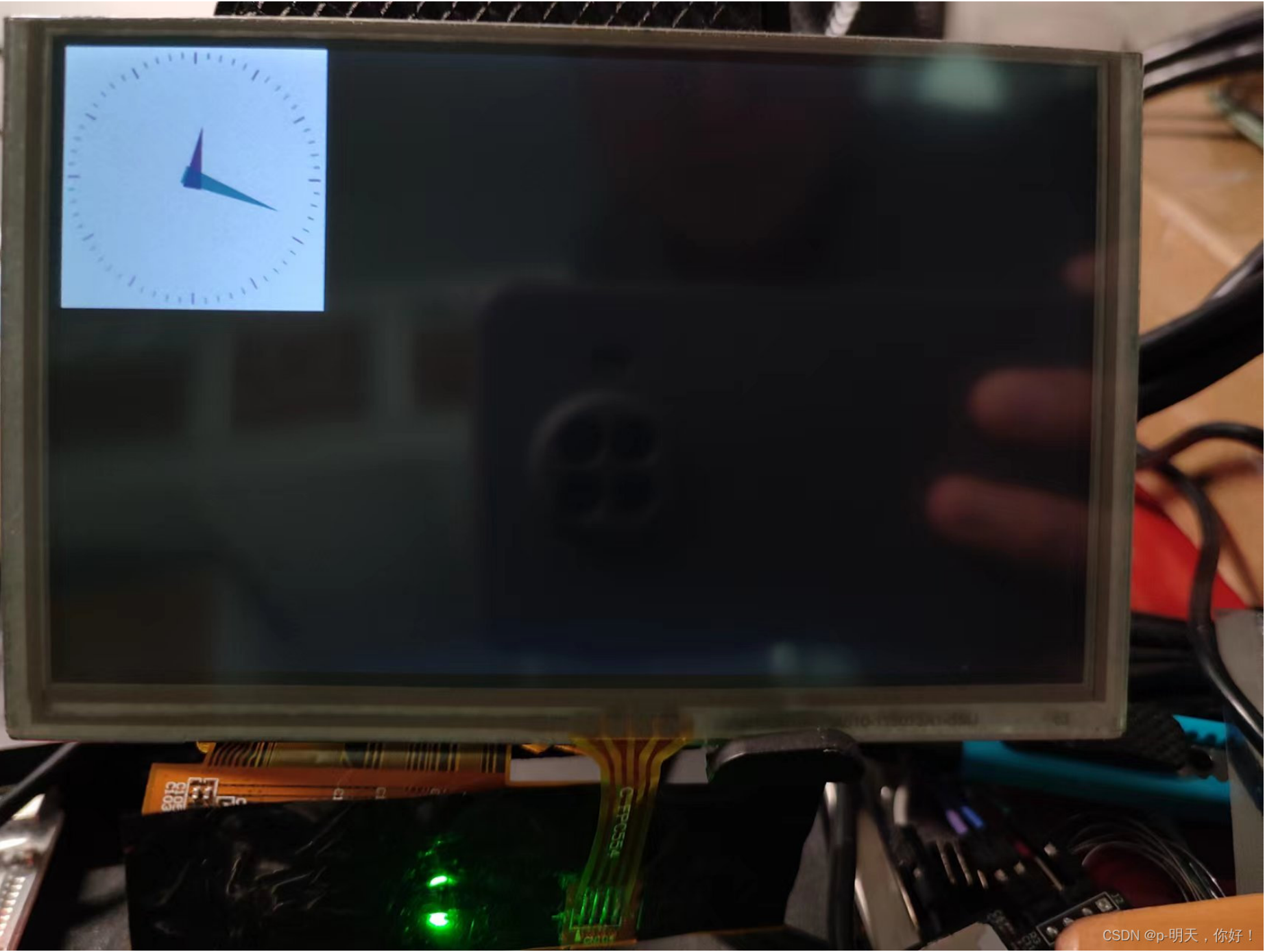


五、上板子运行

1、测试模拟时钟

```
1 | cd /  
2 | ls  
3 | ./analogclock -platform linuxfb
```

```
# cd /  
# ls  
analogclock  lib          mnt          run          tmp  
bin          lib32        opt          sbin         usr  
dev          linuxrc     proc         stickman     var  
etc          media       root        sys  
# ./analogclock -platform linuxfb  
CSDN @p-明天，你好！
```



2、测试火柴人

```
1 | ./stickman -platform linuxfb
```



```
# cd /  
# ls  
analogclock  etc          media        root         sys  
bin          lib          mnt          run          tmp  
charactermap lib32        opt          sbin         usr  
dev         linuxrc     proc         stickman     var  
# ./stickman -platform linuxfb
```

CSDN @p-明天，你好！

