

Test

.

QT	qt4.8	Qtopia ,	Qt	5.7	qt5.6 for wind
JZ2440	ubuntu9.10			ubuntu14.04	
4.3.2	Qt5.6 ,		qt		4.4.3
	4				

- 1
- 2
- 3 tslib
- 4 qt

- 1
arm-**Linux**-gcc-4.4.3.tar.gz . <http://pan.baidu.com/s/1nvJF8ud> oi57
- 2 JZ2440
linux-2.6.22.6.tar.bz2 . <http://pan.baidu.com/s/1gf0oZn1> jrp6
linux-2.6.22.6_jz2440.patch <http://pan.baidu.com/s/1c1CWgTE> uwuu
4.3 LCD_mach-smdk2440.c <http://pan.baidu.com/s/1boZsKwf> 1xml
- 3 busybox
busybox-1.22.1.tar.bz2 . <http://pan.baidu.com/s/1cee6CI> . lv81
- 4 tslib .
tslib-1.4.tar.gz
- 5 qt5.6 <http://pan.baidu.com/s/1jINj3IQ> 6kkc
qt-everywhere-opensource-src-5.6.0.tar.gz <http://pan.baidu.com/s/1kVnV9oN> r7cm

/work

- 1
sudo tar zxvf arm-linux-gcc-4.4.3.tar.gz -C / #
ls /opt/FriendlyARM/toolschain/4.4.3/bin #
 - 2
sudo vi /etc/environment
1. #PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/work/tools/gc
c-3.4.5-glibc-2.3.6/bin"//
 2. PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/opt/FriendlyAR
M/toolschain/4.4.3/bin"//

usr root environment root make install

3

4

arm-linux-gcc -v

4.4.3

jz2440

EABI

jz2440

1

tar jxvf linux-2.6.22.6.tar.bz2

2

cd linux-2.6.22.6

patch -p1 < ../linux-2.6.22.6_jz2440.patch

4.3 JZ2440 LCD 3.5

mv ../4.3 LCD_mach-smdk2440.c arch/arm/mach-s3c2440/mach-smdk2440.c

3

EABI

make menuconfig

kernel features->

<*>use the arm eabi to //

4

makefile 4.4.3

vi Makefile

2.6.32.2

kernel too old

Makefile

VERSION = 2

PATCHLEVEL = 6

SUBLEVEL = 22

EXTRAVERSION = .6

SUBLEVEL = 22 SUBLEVEL = 32

// 4.4.3 gdwarf2

ifdef CONFIG_DEBUG_INFO

CFLAGS += -g

AFLAGS += -gdwarf2

endif

// AFLAGS += -gdwarf2

5

```
make ulmage -j4
ulmage /arch/arm/boot
```

1

```
tar jxvf busybox-1.22.1.tar.bz2
```

2

```
make menuconfig
Busybox Settings ->
general configuration ->
[*] don't use /usr //
build options->
cross compiler prefix = arm-linux- //
installtion options->
busybox installation prefix = /work/my_rootfs //
```

3 &

```
mkdir -p /work/my_rootfs //
make
make install
cd /work/my_rootfs
ls //
/work/my_roofs
```

4 /etc/inittab

```
mkdir -p /work/my_rootfs/etc
vi /work/my_rootfs/etc/inittab
# /etc/inittab
# /etc/init.d/rcS
::sysinit:/etc/init.d/rcS
# shell
::askfirst:/bin/sh
#
::ctrlaltdel:/sbin/reboot
::shutdown:/bin/umount -a -r
```

5 /etc/init.d/rcS

```
mkdir -p /work/my_rootfs/etc/init.d
vi /work/my_rootfs/etc/init.d/rcS
# /bin/sh
#!/bin/sh
#
mount -a
#
mkdir /dev/pts
mount -t devpts devpts /dev/pts
```

```

echo /sbin/mdev > /proc/sys/kernel/hotplug
mdev -s
# IP
#/sbin/ifconfig eth0 192.168.1.17 #nfs
# /etc/fstab
mount -a
exec /etc/rc.local
6 rc.local
vi /work/my_rootfs/etc/rc.local
#!/bin/sh
. /etc/profile // .
7 /etc/fstab
vi /work/my_rootfs/etc/fstab
# device mount-point type options dump fsck order
proc /proc proc defaults 0 0
sysfs /sys sysfs defaults 0 0
tmpfs /tmp tmpfs defaults 0 0
tmpfs /dev tmpfs defaults 0 0
8 /dev
mkdir -p /work/my_rootfs/dev
cd /work/my_rootfs/dev
sudo mknod console c 5 1
sudo mknod null c 1 3
9
mkdir proc mnt tmp sys root usr
10 Lib
cd /opt/FriendlyARM/toolschain/4.4.3/arm-none-linux-gnueabi/lib
cp *.so* /work/my_rootfs/lib -d
cd /opt/FriendlyARM/toolschain/4.4.3/lib
cp *.so* /work/my_rootfs/lib -d

```

tslib

```

1 & &
                                autoconf automake libtool
#sudo apt-get install autoconf automake libtool
tar zxvf tslib-1.4.tar.gz
sudo mkdir -p /usr/local/tslib
cd tslib
./autogen.sh
./configure --host=arm-linux ac_cv_func_malloc_0_nonnull=yes CC=arm-none-linux-gnueabi-gcc
CXX=arm-none-linux-gnueabi-g++ -prefix=/usr/local/tslib
make
sudo make install

```

undefined reference to 'rpl_malloc'
tslib /usr/local/tslib

config.h.in

#undef malloc nr

```
2  tslib
   cd /usr/local/tslib/etc/
   sudo vi ts.conf
      # module_raw input    “#”
3  tslib
   cd /usr/local
   sudo tar zcvf tslib.tar.gz tslib
   mkdir -p /work/my_rootfs/usr/local
   cp tslib.tar.gz /work/my_rootfs/usr/local
   tar zxvf tslib.tar.gz
   rm tslib.tar.gz
4  tslib
   vi /work/my_rootfs/etc/profile

   #!/bin/sh
   export T_ROOT=/usr/local/tslib
   export LD_LIBRARY_PATH=/usr/local/tslib/lib:$LD_LIBRARY_PATH
   export TSLIB_CONSOLEDDEVICE=none
   export TSLIB_FBDEVICE=/dev/fb0
   export TSLIB_TSDEVICE=/dev/input/event0
   export TSLIB_PLUGINDIR=$T_ROOT/lib/ts
   export TSLIB_CONFFILE=$T_ROOT/etc/ts.conf
   export POINTERCAL_FILE=/etc/pointercal
   export TSLIB_CALIBFILE=/etc/pointercal
      tslib          nfs          cd /usr/local/tslib/bin

   ./ts_test

                                     “selected device is not a touchscreen I understand”
0x0100000
```

qt5.6

```
1  tar zxvf qt-everywhere-opensource-src-5.6.0.tar
2
   cd /work/qt-everywhere-opensource-src-5.6.0/qtbase/mkspecs/linux-arm-gnueabi-g++
   vi qmake.conf
      2440
   QT_QPA_DEFAULT_PLATFORM = linuxfb
   QMAKE_CFLAGS += -msoft-float -D__GCC_FLOAT_NOT_NEEDED -march=armv4t -mtune=arm920t
   QMAKE_CXXFLAGS += -msoft-float -D__GCC_FLOAT_NOT_NEEDED -march=armv4t -
   mtune=arm920t
```

```

march    cpu    2440    armv4t
mtune    cpu      2440    arm920t
    A8
QT_QPA_DEFAULT_PLATFORM = linuxfb
QMAKE_CFLAGS += -msoft-float -D__GCC_FLOAT_NOT_NEEDED -march=armv7-a -mtune=cortex-
a8
QMAKE_CXXFLAGS += -msoft-float -D__GCC_FLOAT_NOT_NEEDED -march=armv7-a -
mtune=cortex-a8
    A9
QT_QPA_DEFAULT_PLATFORM = linuxfb
QMAKE_CFLAGS += -msoft-float -D__GCC_FLOAT_NOT_NEEDED -march=armv7-a -mtune=cortex-
a9
QMAKE_CXXFLAGS += -msoft-float -D__GCC_FLOAT_NOT_NEEDED -march=armv7-a -
mtune=cortex-a9

# modifications to g++.conf
QMAKE_CC = arm-linux-gnueabi-gcc
QMAKE_CXX = arm-linux-gnueabi-g++
QMAKE_LINK = arm-linux-gnueabi-g++
QMAKE_LINK_SHLIB = arm-linux-gnueabi-g++

# modifications to linux.conf
QMAKE_AR = arm-linux-gnueabi-ar cqs
QMAKE_OBJCOPY = arm-linux-gnueabi-objcopy
QMAKE_NM = arm-linux-gnueabi-nm -P
QMAKE_STRIP = arm-linux-gnueabi-strip
    -lts          tslib
# modifications to g++.conf
QMAKE_CC = arm-none-linux-gnueabi-gcc -lts
QMAKE_CXX = arm-none-linux-gnueabi-g++ -lts
QMAKE_LINK = arm-none-linux-gnueabi-g++ -lts
QMAKE_LINK_SHLIB = arm-none-linux-gnueabi-g++ -lts

# modifications to linux.conf
QMAKE_AR = arm-none-linux-gnueabi-ar cqs
QMAKE_OBJCOPY = arm-none-linux-gnueabi-objcopy
QMAKE_NM = arm-none-linux-gnueabi-nm -P
QMAKE_STRIP = arm-none-linux-gnueabi-strip
3
sudo mkdir -p /usr/local/Qt5.6
cd ../../../../

./configure -prefix /usr/local/Qt5.6 \
-opensource \
-release \

```

```

-confirm-license \
-xplatform linux-arm-gnueabi-g++ \
-shared \
-qt-zlib \
-no-gif \
-qt-libjpeg \
-no-nis \
-no-opengl \
-no-cups \
-no-glib \
-no-dbus \
-no-rpath \
-no-sse2 -no-sse3 -no-ssse3 -no-sse4.1 -no-sse4.2 \
-no-avx \
-no-openssl \
-nomake tools \
-qreal float \
-qt-libpng \
-tslib \
-nomake examples \
-I /usr/local/tslib/include \
-L /usr/local/tslib/lib

```

```

make -j4
sudo make install

```

libxcb qtbase/src/plugins/platforms/xcb readme

```

4      qt
cd /usr/local
sudo tar zcvf Qt5.6.tar.gz Qt5.6
cp Qt5.6.tar.gz /work/my_rootfs/usr/local/
tar zxvf Qt5.6.tar.gz
rm Qt5.6.tar.gz
rm -r doc include bin mkspecs qml translations

```

```

5      qt      http://doc.qt.io/qt-5/embedded-linux.html

```

```

/etc/profile
export QTDIR=/usr/local/Qt5.6
export LD_LIBRARY_PATH=/usr/local/Qt5.6/lib:$LD_LIBRARY_PATH
export QT_QPA_GENERIC_PLUGINS=tslib
export QT_QPA_FONTDIR=$QTDIR/lib/fonts
export QT_QPA_PLATFORM_PLUGIN_PATH=$QTDIR/plugins
export
QT_QPA_PLATFORM=linuxfb:fb=/dev/fb0:size=480x272:mmSize=480x272:offset=0x0:tty=/dev/tty

```


1

```
export QT_QPA_FB_TSLIB=1
```

3.5 320*240 LCD JZ2440 V3

qt nfs [.NET/jaffre0112/article/details/52589215](http://www.cnblogs.com/afreelife/article/details/52589215) qt

Qt

windows qt windows qt QQ

windows

```
/usr/local/Qt5.6/bin/qmake
```

```
make
```

qmake

QlconvCodec::convertFromUnicode: using Latin-1 for conversion, iconv_open failed
QlconvCodec::convertToUnicode: using Latin-1 for conversion, iconv_open failed

libiconv

<http://pan.baidu.com/s/1c22xb4O> pbld

```
mkdir -p /usr/local/libiconv
```

```
./configure --host=arm-none-linux-gnueabi --prefix=/usr/local/libiconv CC=arm-none-linux-gnueabi-gcc LDFLAGS="-L/opt/FriendlyARM/toolschain/4.4.3/arm-none-linux-gnueabi/sys-root/lib" --enable-static
```

```
make
```

```
sudo make install
```

```
/lib preloadable_libiconv.so /lib , /etc/profile  
export LD_PRELOAD=/lib/preloadable_libiconv.so
```

7 Qt

qt

<http://pan.baidu.com/s/1bp9QFQv> 2u81

DroidSansFallback.ttf /usr/local/Qt5.6/lib/fonts Qt

qt creator qt

QT += gui

```
#include <QFontDatabase>
```

```
int id = QFontDatabase::addApplicationFont("/usr/local/Qt5.6/lib/fonts/DroidSansFallback.ttf");
```

```
QString msyh = QFontDatabase::applicationFontFamilies(id).at(0);
```

```
QFont font(msyh,10);
```

```
font.setPointSize(20);
```

```
this->setFont(font);
```

8 &

8.1 qt Ui 320*240

<http://pan.baidu.com/s/1qXV4C1Y> 908q

8.2 jz2440 3.5 4.3

<http://pan.baidu.com/s/1qYMrxmc> qsk0

8.3

<http://pan.baidu.com/s/1skLtj3J> lbo2

nfs dev console null

/etc/profile 3.5

80 M yaffs 100 M qt app

ssh

<http://blog.csdn.net/lizuobin2/article/details/52664339>

rz

<http://blog.csdn.net/lizuobin2/article/details/52601617>

<http://blog.csdn.net/lizuobin2/article/details/52664657>