



PET-R16 开发板/全功能板 开发手册



一、编译环境搭建指南

▶ 安装 Ubuntu 14.04 64 位。

安装 openjdk-7-jdk。
 sudo apt-get install openjdk-7-jdk

▶ 安装完成后修改 /etc/profile 中的环境变量,将 openjdk 加入到 JAVA_HOME 中。

输入命令 java -version 检查 java 的主版本号是否为 1.7。 在编译过程中可能会由于系统缺少依赖包而报错,可以通过百度或谷歌查找相关解决方案,一般 通过 sudo apt-get install <需要安装的包名> 这个命令进行在线安装。

二、解压源代码

将源代码压缩文件全部复制到 Ubuntu 系统下,保证所在磁盘剩余空间要大于 50G,使用以下命令解 压源代码:

tar xvJf PET_R16_6.0.1_Source.tar.xz

三、编译安卓 Android

首次编译请严格按照步骤进行内核、uboot、android 的编译,否则编译可能会出现错误。

1、编译内核





2、编译 uboot

首次编译或修改 uboot 代码后需要执行这一步骤。

cd lichee/brandy ./build.sh -p sun8iw5p1

编译完成后正确提示如下

/root/Work/A33_5.1.1_SKD_BASE/lichee/brandy/u-boot-2011.09/../gcc-linaro/bin/arm-linux-gnueabi-objc opy --gap-fill=0xff -0 binary /root/Work/A33_5.1.1_SKD_BASE/lichee/brandy/u-boot-2011.09/sunxi_spl /boot0/boot0_sdcard.axf /root/Work/A33_5.1.1_SKD_BASE/lichee/brandy/u-boot-2011.09/sunxi_spl/boot0/ boot0_sdcard.bin make[1]: Leaving directory `/root/Work/A33_5.1.1_SKD_BASE/lichee/brandy/u-boot-2011.09/sunxi_spl/bo ot0 actool_nand_sun8iw5p1.binâ€^m -> †´/root/Work/A33_5.1.1_SKD_BASE/lichee/brandy/u-boot-2011.09/sunxi_spl/bo ../tools/pack/chips/sun8iw5p1/bin/boot0_nand_sun8iw5p1.binâ€^m ac `boot0_sdcard_sun8iw5p1.binâ€^m -> †´/root/Work/A33_5.1.1_SKD_BASE/lichee/brandy/u-boot-2011.09/../ ac `boot0_sdcard_sun8iw5p1.binâ€^m -> †`/root/Work/A33_5.1.1_SKD_BASE/lichee/brandy/u-boot-2011.09/../ ac `boot0_sdcard_sun8iw5p1/bin/boot0_nand_sun8iw5p1.binâ€^m a./../tools/pack/chips/sun8iw5p1/bin/boot0_sdcard_sun8iw5p1.binâ€^m

3、编译 android



编译完成后会在 lichee/tools/pack 目录下生成 sun8iw5p1_android_d7_uart0.img 系统烧写镜像文件。



四、编译 Linux + QT5.8

请首先新开一个控制台进行编译操作。

首次编译请严格按照步骤进行内核、uboot、Rootfs 的编译,否则编译可能会出现错误。

1、编译内核

cd lichee
./build.sh -p sun8iw5p1_dragonboard
编译完成后正确提示如下
build_ramfs Copy boot.img to output directory
sun8iw5p1 compile Kernel successful
<pre>INFO: build kernel OK. INFO: build rootfs Regenerating dragonboard Rootfs event num 4 /root/Work/A33_5.1.1_SKD_BASE/lichee/buildroot/target/dragonboard/rootfs/etc/profile generating rootfs blocks: 551M -> 768M Creating filesystem with parameters: size: 805306368 Block size: 4096 Block sper group: 32768 Inodes per group: 32768 Inodes per group: 8192 Inode size: 256 Journal blocks: 3072 Label: Blocks: 196608 Block groups: 6 Reserved block group size: 47</pre>
Created filesystem with 10196/49152 inodes and 147561/196608 blocks e2fsck 1.42.12 (29-Aug-2014)
fsck.ext4: Bad magic number in super-block while trying to open rootfs.ext4 success in generating rootfs
Build at: Mon May 8 17:56:23 CST 2017
INFO:INFO: build lichee OK.
INFO:

2、编译 uboot

首次编译或修改 uboot 代码后需要执行这一步骤。



www.gzpeite.net



3、编译 QT Rootfs

首先切换到 pack 目录
cd lichee/tools/pack/
./pack -c sun8iw5p1 -p dragonboard -b d7 -d uart0 -s none
编译完成后正确提示如下
FilePath: rootfś.fex FileLength=1d8c268sys_config.fex Len: 0x1072b config.fex Len: 0xa2e0 split_xxxx.fex Len: 0x200 sys_partition.fex Len: 0x8000 boot0_sdcard.fex Len: 0x8000 u-boot.fex Len: 0x24000 fes1.fex Len: 0x23a0 usbtool.fex Len: 0x23a0 usbtool.fex Len: 0x23a0 aultools.fex Len: 0x26ead aults32.fex Len: 0x238dd cardscript.fex Len: 0x41400 cardscript.fex Len: 0x41400 dinfo.fex Len: 0x4000 arisc.fex Len: 0x4000 arisc.fex Len: 0x4000 vboot-resource.fex Len: 0x4e0c00 vboot-resource.fex Len: 0x4 erv.fex Len: 0x4 boot.fex Len: 0x4 boot.f
/root/Work/PET_A33_6.0.1/lichee/tools/pack/sun8iw5p1_dragonboard_d7_uart0.img
pack finish



4、修改 Rootfs

完成首次编译后,rootfs的所有文件位于 lichee\buildroot\target\dragonboard\rootfs 目录下。 如果需要修改或添加文件,需要将文件复制到 lichee\buildroot\target\dragonboard\extra 相同目录下, 然后再修改,重新编译内核、uboot、rootfs 即可。

例如需要修改 rootfs/etc/init.d/S00peite 这个系统初始化设置脚本文件
cd lichee/buildroot/target/dragonboard
mkdir -p extra/etc/init.d
cp -rf rootfs/etc/init.d/S00peite extra/etc/init.d/S00peite
修改 extra/etc/init.d/S00peite 后重新编译即可生成新的烧写镜像文件

5、更换 Rootfs 为 Linux 或 Linux + QT

首先删除 lichee\buildroot\target\dragonboard\rootfs 目录。 将开发资料《源代码》目录下 rootfs 文件更名为 rootfs.tar.xz



复制 rootfs.tar.xz 到 lichee\buildroot\target\dragonboard 覆盖同名文件 重新编译即可 Rootfs 类型有: Linux_Full --- Linux 全功能版,不包含 QT

Linux_Lite --- Linux 部分功能版,不包含 QT

QT_Full ---- Linux+QT 全功能版

QT_Lite --- Linux+QT 部分功能版

6、交叉编译其他应用

系统使用的交叉编译器为开发资料《源代码/QT_Source》目录下:

gcc-linaro-4.9.4-2017.01-x86_64_arm-linux-gnueabi.tar.xz

编译所需的其他库文件是 sysroot_peite.tar.xz 可根据需要进行解压使用,客户可自行编译其他未包含的 支持库、应用程序等。



五、修改 Linux 内核编译选项

首先切换到 linux 内核目录 cd lichee/linux-3.4/

加载默认配置

make sun8iw5p1smp_linux_defconfig (Linux+QT 系统) make sun8iw5p1smp_android_defconfig (Android 系统)

启动内核配置

make menuconfig



完成内核配置修改后,从新编译 android 或 linux 即可。



六、镜像文件烧写

开发过程中,一般使用 PhoenixSuit 进行镜像文件的烧写,具体操作方式请参考开发工具目录下的 《PhoenixSuit 使用说明文档.pdf》,除了 Android 系统我司的 Linux+QT 系统也支持这种烧写方式。

将开发板的 MicroUSb 接口连接到系统主机后, Linux+QT 系统检测到的设备信息如下:



烧写操作需要首先通过 Micro USB 数据线连接主机的开发板,在进行烧写时如果出现主机识别到新的 设备没有正常安装驱动的情况时,需要手动安装设备驱动程序,驱动程序位于开发工具文件夹内。

注意,在点击烧写镜像后,设备会重启黑屏,此时需要在 PC 端的设备管理区中再手动安装镜像烧写 设备驱动 (AW_Driver)。



七、建立 QT 应用程序编译环境

所需工具位于开发资料的《开发工具/QT》目录下,以下所有操作需要有 ROOT 权限:

- 1、解压交叉编译器 gcc-linaro-4.9.4-2017.01-x86_64_arm-linux-gnueabi.tar.xz tar -xJf gcc-linaro-4.9.4-2017.01-x86_64_arm-linux-gnueabi.tar.xz -C /usr/local
- 2、解压库文件 sysroot_peite_qt.tar.xz tar -xJf sysroot_peite_qt.tar.xz -C /usr/local
- 3、解压安装 qt-creator-opensource-linux-x86_64-4.4.0.tar.xz tar -xJf qt-creator-opensource-linux-x86_64-4.4.0.tar.xz ./qt-creator-opensource-linux-x86_64-4.4.0.run

Qt Creator 4.2.1 Setup

Qt Account - Your unified login to everything Qt

Qt	Login Sign-up	Please log in to Qt Account Email Password Forgot password? Need a Qt Account? Valid email address Password
Create Once. Deploy Everywhere.		Confirm Password
Settings		< <u>B</u> ack Skip Cancel

注意在上面这一步选择 Skip,其他直接选择 Next 即可



4、 启动 qt creator 设置交叉编译器和 QT 库文件路径。

/opt/qt	/opt/qtcreator-4.2.1/bin/qtcreator							
			Qt Creato	r				
<u>F</u> ile <u>E</u> di	t <u>B</u> uild <u>D</u> ebug <u>A</u> nalyze <u>T</u> ools	<u>W</u> indow <u>H</u> elp						
Welcome	Projects	+ New Project		🔚 Open Projec	t			
Edit	Examples	Sessions		Recent Projects				
Design	Tutorials	1 🕨 default						
الله Debug	New to Qt?							
بر Projects	Learn how to develop your own applications and explore							
? Help	Get Started Now							
	L Qt Account							
	Online Community							
	Blogs							
	🕐 User Guide							
-								
\nearrow	P. Type to locate (Ctrl	1 Issues 2 Search Results	3 Application Output	4 Compile Output	5 Debugger Console 💠			

选择菜单 Tools->Options->Build & Run->Compilers, 点击 Add ->Custom->C++ 按钮, 添加 C++编译器, Complier path: /usr/local/gcc-linaro-4.9.4-2017.01-x86_64_arm-linux-gnueabi/bin/arm-linux-gnueabi-g++

		Options				
Filter	Build & Run					
Environment	General Kits G	t Versions Compilers	Debuggers	CMake		
Text Editor	Name		Туре			Add -
🌃 FakeVim	GCC (x86	64bit in /usr/lib64/ccach	e) GCC			Clone
👔 Help	- C		e) GCC			Remove
{} C++	GCC (x86	64bit in /usr/lib64/ccach	e) GCC			Remove
🚄 Qt Quick	▼ Manual	32 Dit in /usr/ubo4/ccach	e) GCC			
🕵 Build & Run	▼ C++		Custom		_	
🔍 Debugger	rene-occ		Custom		•	
💥 Designer						
I Analyzer	Name:	PEITE-GCC				
Version Control	<u>C</u> ompiler path:	4-2017.01-x86_64_a	rm-linux-gnueat	oi/bin/arm-linux-gnueabi-g++	Browse	
Devices	<u>M</u> ake path:	/usr/bin/make			Browse	
🚮 Code Pasting	<u>A</u> BI:	arm 🔻 - linux	- gener	ic • - elf • - 32	2bit 👻	
→ Qbs	Predefined macros:	Empty			Details 🔻	
				Аррі	y <u>C</u> ancel	<u>О</u> К



Add ->Custom->C 添加 C 编译器

Complier path: /usr/local/gcc-linaro-4.9.4-2017.01-x86_64_arm-linux-gnueabi/bin/arm-linux-gnueabi-gcc:

			Options					
Filter	Build & Run							
Environment	General Kits G	t Versions	Compilers	Debuggers	CMake			
Text Editor	Name			Туре				Add -
🔚 FakeVim	▼ C							Clone
Help	GCC (x86 GCC (x86	64bit in /usr/ 32bit in /usr/	/lib64/ccache /lib64/ccache) GCC) GCC				Remove
{} C++	▼ Manual							Kellove
🗸 Qt Quick	▼ C++ PEITE-GCC			Custom				
🚯 Build & Run	τ C							
ଭ Debugger	PEITE-GCC			Custom				•
📡 Designer								
🛄 Analyzer	Name:	PEITE-GCC						
Version Control	<u>C</u> ompiler path:	.4-2017.0	1-x86_64_a	m-linux-gnuea	abi/bin/arm-	linux-gnueabi-gcc	Browse	
Devices	<u>M</u> ake path:	/usr/bin/ma	ke				Browse	
Code Pasting	<u>A</u> BI:	arm	▼ - linux	👻 - gener	ic -	elf - 32	2bit 👻	
🎤 Qbs	Predefined macros:	Empty					Details 🔻	•
						Apply	y <u>C</u> ance	el <u>O</u> K

选择菜单 Tools->Options->Build & Run->Qt Versions, 点击 Add 按钮, 配置如下:

	Options	
Filter	Build & Run	
Environment	General Kits Qt Versions Compilers Debuggers CMake	
Text Editor	Name • qmake Location	Add
🚮 FakeVim	Auto-detected	Remove
Help	✓ Manual Qt 5.8.0 (Qt-5.8.0) /usr/local/sysroot_peite_qt/usr/local/Qt-5.8.0/bin/qmake	
⟨} C++		Clean Up
🗸 Qt Quick		
🕕 Build & Run		
ଭ Debugger		
💥 Designer		
🛄 Analyzer		
Version Control		
Devices	Version name: Qt %(Qt:Version) (Qt-5.8.0) β _B	
🚮 Code Pasting	qmake location: /usr/local/sysroot_peite_qt/usr/local/Qt-5.8.0/bin/qmake Browse	
Qbs		
	Qt version 5.8.0 for Embedded Linux Details *	
	Apply Cancel	<u>о</u> к



选择菜单 Tools->Options->Devices,点击 Add 按钮,配置如下:	
Device Configuration Wizard Selection	
Available device types:	
Generic Linux Device	
QNX Device	
Cancel Start Wizard	j
首先需要将开发板与主机在同一局域网内连接好,主机可以正常,	ping 通开发板
New Generic Linux Device Configurat	ion Setup
Connection	

Connection	The name to identify this configuration:	Generic Linux Device	
Summary	The device's host name or IP address:	192.168.1.85	
	The username to log into the device:	root	
	The authentication type:	Password Key	
	The user's password:		
	The file containing the user's private key:	/root/.ssh/id_rsa	Browse



测试通过后的配置如下:

Devices		
Devices Android	QNX	
Device: Generic Lin	ux Device (default for Generic Linux) -	<u>A</u> dd
<u>N</u> ame:	Generic Linux Device	Remove
Туре:	Seneric Linux	Set As Default
Auto-detected:	No	Test
Current state: 0	JIKROWN	Show Running Processes
Type Specific		Deploy Public Key
Machine type:	Physical Device	
Authentication ty	rpe: • Password · Key	
<u>H</u> ost name:	2.168.1.85 SSH port: 22	
Free ports:	10000-10100 Timeout: 10s	
<u>U</u> sername:	root	
Password:	Show password	
Private key file:	Browse Create New	_
	Арр	ly <u>Cancel O</u> K

Build & F	Run								
General	Kits	Qt Versions	Compilers	Debuggers	CMake				
Name:		EPITE					.		
File syst	em nam	e:							
Device t	ype:	Generic	Linux Device				•		
Device:		Generic	Linux Device	default for Ge	neric Linux)		▼ Manage		
Sysroot:		/usr/loc	al/sysroot_peit	e_qt/usr/local	/Qt-5.8.0		Browse		
Compile	C: PEITE-C					▼ Manage			
Compiler		C++: P	C++: PEITE-C++						
Environn	nvironment: No changes to apply.								
Debugge	er:	System	System GDB at /usr/bin/gdb						
Qt versi	on:	Qt 5.8.	Qt 5.8.0 (Qt-5.8.0)						
Qt mksp	ec:								
CMake 1	Fool:	System	CMake at /usi	/bin/cmake			▼ Manage		
CMake o	generato	r: CodeBlo	cks - Unix Mak	efiles, Platforr	m: <none>, Toolset: <</none>	<none></none>	Change 👻		
						Apply C	Cancel OK		

选择菜单 Tools->Options->Build & Run->Kits, 点击 Add 按钮, 配置如下:



八、创建并编译 QT 程序

		New Project	
Choose a template	2:		All Templates -
Projects Application Library Other Project Non-Qt Project Import Project Files and Classes		 Qt Widgets Application Qt Console Application Qt Quick Application Qt Quick Controls 2 Application Qt Quick Controls Application Qt Canvas 3D Application 	Creates a Qt application for the desktop. Includes a Qt Designer- based main window. Preselects a desktop Qt for building the application if available. Supported Platforms: Generic Linux Device
			Cancel Choose
		Qt Widgets Application	
	Introduction and This wizard generates and includes an empty	Project Location a Qt Widgets Application project. The applic widget.	ation derives by default from QApplication
	Name: peite_test Create in: /root Use as default pro	ject location	Browse
			Next > Cancel



PET-R16 开发板/全功能板

		Qt Widgets Application				
Location	Kit Selection					
[●] Kits Details Summary	Qt Creator can use the following kits for project peite_test: ▼ Select all kits					
	V P EPITE			Details 🔺		
	✓ Debug /roo	ot/build-peite_test-EPITE-Debug		Browse		
	✓ Release /roo	ot/build-peite_test-EPITE-Release		Browse		
	✓ Profile /roo	/root/build-peite_test-EPITE-Profile		Browse		
			< <u>B</u> ack Next	> Cancel		
		Qt Widgets Application				
Location	Class Informat	ion				

Kits Specify basic information about the classes for which you want to generate skeleton source code files.

<u>C</u> lass name:	MainWindow
Base class:	QMainWindow -
<u>H</u> eader file:	mainwindow.h
Source file:	mainwindow.cpp
<u>G</u> enerate form:	
Form file:	mainwindow.ui
	< <u>B</u> ack <u>N</u> ext > Cancel
	<u>C</u> lass name: <u>B</u> ase class: <u>H</u> eader file: <u>S</u> ource file: <u>G</u> enerate form: <u>F</u> orm file:



PET-R16 开发板/全功能板

Qt Widgets Application						
Location	Project Management					
Kits	Add as a subproject to project:	<none> ~</none>				
Details	Add to version control:	<none> Configure</none>				
⊘ Summary						
	Files to be added in					
	/root/peite_test:					
	main.cpp mainwindow.cpp mainwindow.h mainwindow.ui peite_test.pro					
		< Back Finish Cancel				
完成创建后,需	需要修改工程目录下的 peit	e_test.pro 文件,在文件最后添加下面两行代码				
target.path = /root INSTALLS += target						
peite_test,pro - peite						
<u>File</u> Edit B	uild <u>D</u> ebug <u>A</u> nalyze <u>T</u> e	ools <u>W</u> indow <u>H</u> elp				
Pro	≑ 🖛 🕀 🖻 🖌 🗎	🔪 🖬 👼 peite_test.pro 🛛 🗢 🗢 🗢				
	peite_test 10	ARGET = peite_test				
Welcome	Headers	EMPLATE = app				
	Sources	The following define makes your compiler emit warning				
Edit	Forms	any feature of Qt which as been marked as deprecated depend on your compiler). Please consult the document				
	17 # 18 DB	deprecated API in order to know how to port your code FEINES += OT DEPRECATED WARNINGS				
Design	19					
	20 #	In order to do so, uncomment the following line.				
	22 #	You can also select to disable deprecated APIs only u DEFINES += OT DISABLE DEPRECATED BEFORE=0×0600000 #				
Debug	24					
2	26 SC	DURCES += main.cpp\				
Projects	27	mainwindow.cpp				
•	29 H	EADERS += mainwindow.h				
Help	30 31 FC	DRMS += mainwindow.ui				
	32	arget path = /root				
	34 II	VSTALLS += target				
	35					

然后在 QT Creator 中重新打开工程,编译、运行后,可以在开发板上查看运行效果。



九、定制编译 QT 源码

客户可以自行编译 QT 的源码,可以对 QT 源码进行修改及定制,以下执行步骤需要 ROOT 权限。

- 1、复制 QT_Source 目录及所有文件到编译主机。
- 2、进入 QT_Source 目录,运行 config.sh 进行编译环境准备及选项配置。
- 3、运行 build.sh 编译
- 4、编译完成后的 QT 安装目录为 /usr/local/sysroot_peite_qt/usr/local/Qt-5.8.0



十、联系方式

- 地址 : 广州市天河区大观中路新塘大街鑫盛工业园 A1 栋 201
- 电话 : 020-85625526
- 传真 : 020-85625526-606
- 主页 : <u>http://www.gzpeite.net</u>
- 淘宝店 : <u>https://shop149045251.taobao.com</u>
- 核心板 : 王先生
- 移动电话: 18926288206
- 电子信箱: 18926288206@gzpeite.net
- 业务 QQ: 594190286

定制研发:杨先生

- 移动电话: 18902281981
- 电子信箱: 18902281981@gzpeite.net
- 业务 QQ: 151988801

广州佩特电子科技有限公司

2017年12月