UC15 Android RIL Driver User Guide

UMTS/HSPA Module Series

Rev. UC15_Android_RIL_Driver_User_Guide_V1.0

Date: 2014-03-07

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.
Office 501, Building 13, No.99, Tianzhou Road, Shanghai, China, 200233
Tel: +86 21 5108 6236
Mail: info@quectel.com

Or our local office, for more information, please visit:
http://www.quectel.com/support/salesupport.aspx

For technical support, to report documentation errors, please visit:
http://www.quectel.com/support/techsupport.aspx

GENERAL NOTES

QUECTEL OFFERS THIS INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

About the Document

History

Revision	Date	Author	Description
1.0	2014-03-07	Clare CHEN	Initial

Contents

Table Index

Figure Index

1 Introduction

This document mainly introduces how to integrate RIL driver into Android OS of your target machine and how to modify the configuration files and insert some script files for staring RIL service and PPP dialing.

2 Introduction to Driver Software

2.1. Driver Package

2.1.1. Directory Structure

The file structure of RIL driver package published by Quectel is shown as Figure 1.

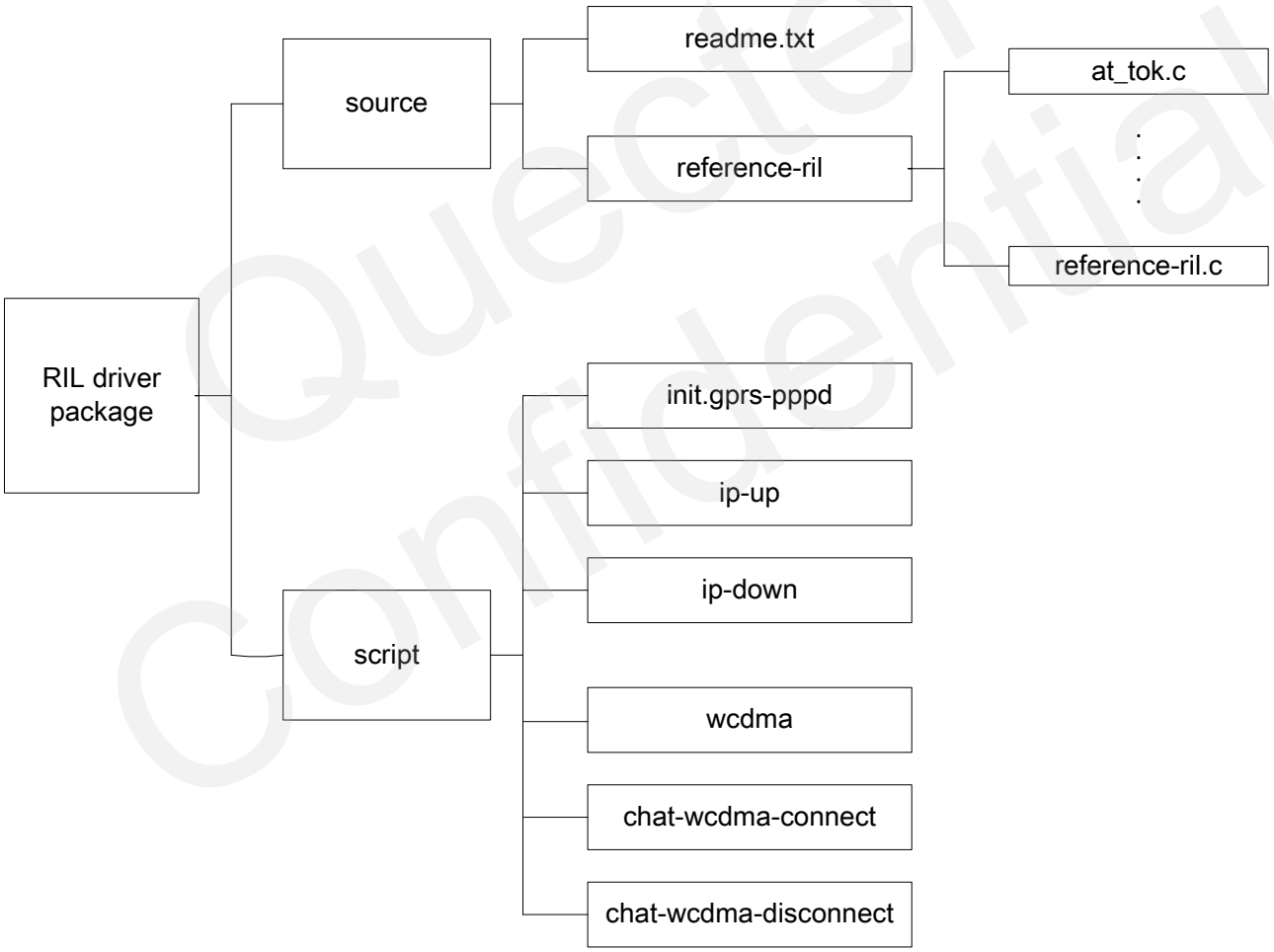

Figure 1: RIL Driver Package Structure

2.1.2. Files Classification

RIL Driver Directory	reference-ril
Script for PPPD	init.gprs-pppd, ip-up, ip-down, wcdma
Script for CHAT	chat-wcdma-connect, chat-wcdma-connect

2.2. Driver Functionalities

When you installed and configured the RIL driver successfully, you can use the following functions in your Android operating system.

Table 1: Supported Functions

Function	Support
SMS	YES
VOICE CALL	YES
DATA SERVICE	YES
SIM TOOL KIT	NO
PHONEBOOK	NO

3 Introduction to System Setup

The first part describes the RIL driver architecture. The rest introduces how to set up Android system with the RIL driver.

3.1. RIL Driver Structure

Android RIL (Radio Interface Layer) provides the abstract layer between Telephony service and Radio hardware.

The following illustration describes the RIL's position in the Android architecture.

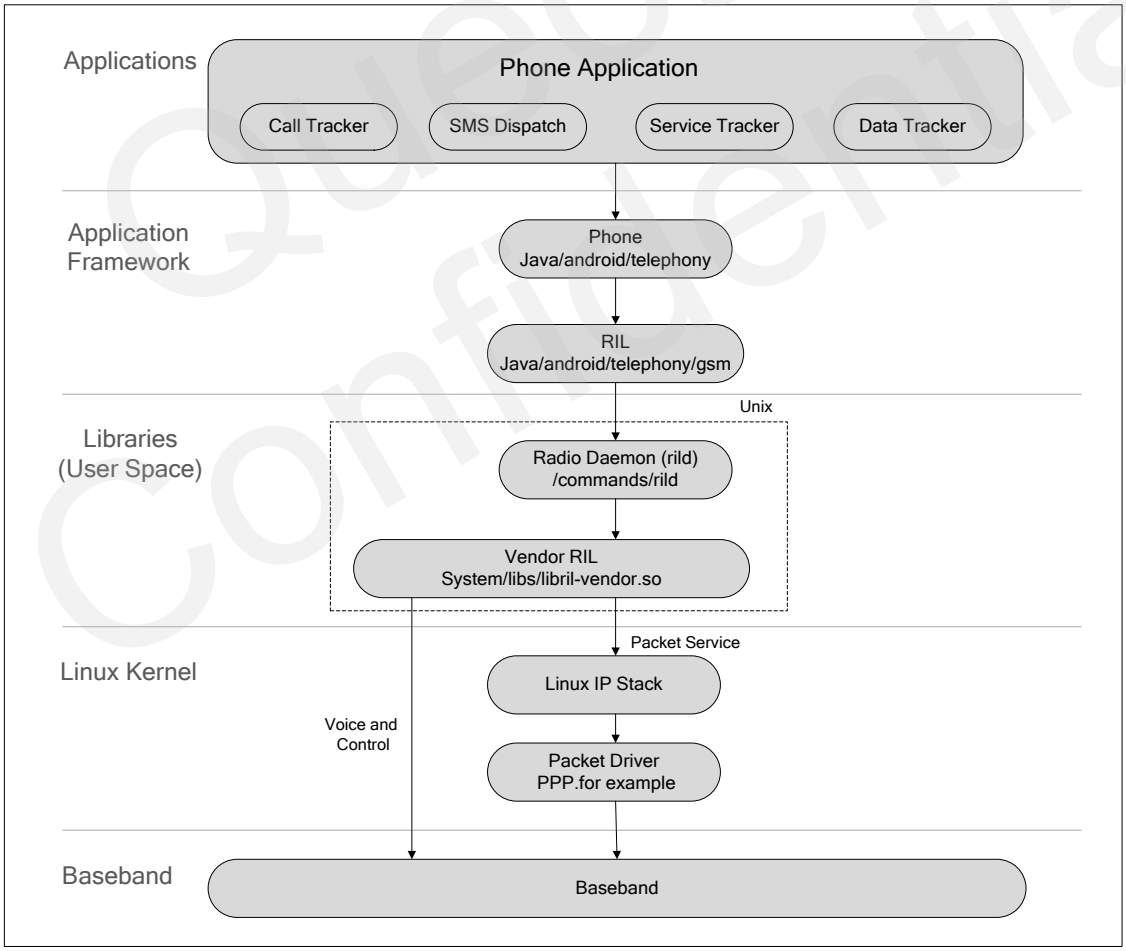

Figure 2: RIL Driver Architecture

The RIL in Android locates between Kernel and Application Framework. It is divided into two parts, one is RILD and the other is Vendor RIL. RILD is responsible for the communication between Socket and Application Framework. Vendor RIL is responsible for communication with Radio via AT command channel and Packet data channel (PDCH). AT command channel is used for communicating with Radio directly and PDCH used for data service.

The java framework of RIL is divided into two parts too. One is RIL module and the other is Phone module. The RIL module is used to communicate with the lower RILD. The Phone module directly provides phone function interfaces to application user who can call them to realize the phone functions.

3.2. Add the Essential Components

For the UC15 module, it adopts USB interface communicating with the target machine, and the function of the upper interface is a serial port, so system has to support the USB to serial port by adding the related kernel configuration items. The method is listed as follows:

1. Input commands in Terminal to open the configuration window:

```
#cd ($kernel_src)          //Locate in the Android kernel source code file directory.
#make menuconfig           //Open the interface of kernel configuration.
```

2. Browse through the menus "Device Driver" → "USB Support"→ "USB Serial Converter Support" and choose:

- "USB Generic Serial Driver"
- "USB driver for GSM and CDMA modems"

3. Browse through the menus "Device Drivers" → "Network Device Support" → "PPP Support" and choose:

- "PPP filtering"
- "PPP support for async serial ports"
- "PPP support for sync tty ports"
- "PPP deflate compression"
- "PPP BSD-compress compression"

4. Save and exit.

5. Re-compile the kernel.

NOTE

For the detailed operation, please refer to the document "*Quectel_UC15_Embedded_Linux_USB_Driver_User Guide_V1.0*".

3.3. Add Device Driver

Modify the source code file "option.c" in Android kernel by adding VID and PID of UC15, so that the OS can recognize it.

The UC15's VID and PID are listed as follows:

- VID – 0x05c6
- PID – 0x9090

Open the file "option.c" in the path of "\drivers\usb\serial" and find the struct array "static structusb_device_idoption_ids[]". Insert "{USB_DEVICE (0x05c6, 0x9090)}," into the array, then save and close it. At last, re-compile the Linux kernel.

NOTE

For the detailed operation, please refer to the document "*Quectel_UC15_Embedded_Linux_USB_Driver_User Guide_V1.0*".

3.4. RIL Driver Integration

At present, Quectel provides RIL driver in the form of source code. You only need to copy the RIL driver source code files to the correct path on your project directory, and recompile the Android system.

1. The source path of the RIL driver files in RIL Driver package is: **RIL Driver package/Source file/reference-ril**

2. The destination path in Android system is: **($Android_src)/hardware/ril/reference-ril**

After the files have been replaced, you have to modify the system configuration so that the necessary services or processes can be loaded when the Android system is started.

3.5. System Configuration

In order to use the RIL driver normally, you have to configure some Android system files. According to the functions you need, you can add or modify the related files selectively.

3.5.1. Implement SMS and VOICE CALL

Add the following strings in "init.rc" that can start the RIL service when Android system is started.

```
Service ril-daemon /system/bin/rild -l /system/lib/ libreference-ril.so -- -d /dev/ttyUSB2
class main
socket rild stream 660 root radio
socket rild-debug stream 660 radio system
user root
group radio cache inet misc audio sdcard_rw log
```

The path of the file "init.rc" is "\rootfs_dir\init.rc".

Get root access.

```
//switchUser();
```

Comment the switchUser() line in main function in rild.c. The path of the file “rild.c” is “\hardware\ril\rild”.

3.5.2. Implement DATA SERVICE

Quectel provides six script files for DATA SERVICE. The files "init.gprs-pppd" are used to start the PPPD process and the rest files are used for PPP dialling.

The RIL driver creates PPP link finally by calling PPPD process, which is started by executing the script file "init.gprs-pppd". So, in addition to start RIL service, you have to add or modify the related script files to prepare for PPP dialling. The detailed steps are described as follows:

1. Add or Replace the Script Files

- Copy "init.gprs-pppd" to the path of "out\target\product\XXX\system\etc".
- Copy "ip-up" and "ip-down" to the path of "out\target\product\XXX\system\etc\ppp".
- Copy "wcdma", "chat-wcdma-connect" and "chat-wcdma-disconnect" to the path of "out\target\product\XXX\system\etc\ppp\peers".

Add the Following Commands into the File "init.rc"

```
#start script "init.gprs-pppd"
service pppd_gprs /etc/init.gprs-pppd /dev/ttyUSB3
user root
group radio cache inet misc
disabled
oneshot
```

2. Set the Port Property

The RIL driver needs to use the USB port, so the port attribute of the devices should be set for read/write. For example, if your android system creates ttyUSB6 for AT port and ttyUSB7 for Modem port of UC15 on your board, you need to confirm that these two ports can be opened by RIL. We changed the attribute of /dev/ttyUSB2 (AT port) and /dev/ttyUSB3 (Modem port) in mainLoop() function in reference-ril.c by default. You must modify the device index according to your developing environment. The source code we provided is as below:

```
chmod("/dev/ttyUSB2", 0777);
chmod("/dev/ttyUSB3", 0777);
```

3. Set the Right of the File "init.gprs-pppd"

RIL driver needs to perform the script "init.gprs-pppd" in the stage of data networking, so the operation permissions of the script file should be set to be executable. Modify the file "android_filesystem_config.h" and insert the following black string into it.

```
static structfs_path_config android_files[] = {
…
{ 00777, AID_ROOT,     AID_SHELL,   "system/etc/init.gprs-pppd" },
…
};
```

The path of the file " android_filesystem_config.h" is "../system/core/include/private/".

4 Debugging Method

4.1. Method of Catching Log

Catch the log of RIL module by typing the following commands HyperTerminal:

```
Adb shell
Logcat –b radio&
```

Catch the log of Android system by typing the following commands in HyperTerminal:

```
Adb shell
Logcat&
```

4.2. Some Common Log Tags

RIL	/hardware/ril/reference-ril/refereince-ril.c
AT	/hardware/ril/reference-ril/atchannel.c
RILD	/hardware/ril/rild/rild.c
RILC	/hardware/ril/libril/ril.cpp
RILB	/frameworks/base/telephony/java/com/android/internal/telephony/BaseCommands.java
RILJ	/frameworks/base/telephony/java/com/android/internal/telephony/gsm/RIL.java
GSM	/frameworks/base/telephony/java/com/android/internal/telephony/gsm/GSMPhone.java

5 Appendix A Reference

Table 2: Terms and Abbreviations

Abbreviation	Description
RIL	Radio Interface Layer
TA	Terminal Adapter
MS	Mobile Station
GSM	Global System for Mobile Communications
WCDMA	Wideband Code Division Multiple Access
VID	Vendor ID
PID	Product ID