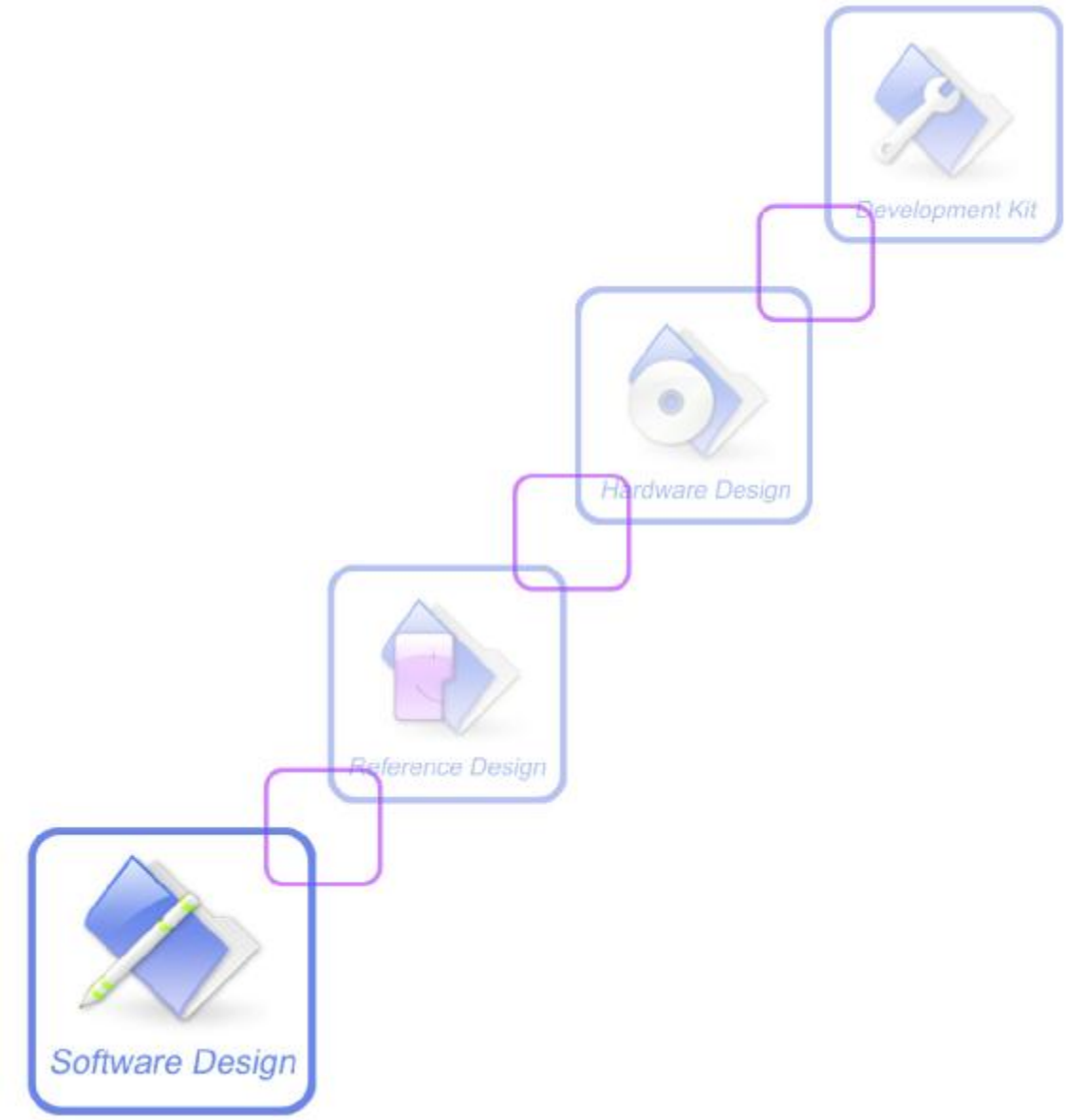
SIM52xx For Android Application Note



Document Title:	SIM52xx For Android Application Note
Version:	0.01
Date:	2010-06-30
Status:	Release
Document ID:	SIM52xx_For_Android_Application_Note_V0.01

General Notes

SIMCom offers this information as a service to its customers, to support application and engineering efforts that use the products designed by SIMCom. The information provided is based upon requirements specifically provided to SIMCom by the customers. SIMCom has not undertaken any independent search for additional relevant information, including any information that may be in the customer's possession. Furthermore, system validation of this product designed by SIMCom within a larger electronic system remains the responsibility of the customer or the customer's system integrator. All specifications supplied herein are subject to change.

Copyright

This document contains proprietary technical information which is the property of SIMCom Limited., copying of this document and giving it to others and the using or communication of the contents thereof, are forbidden without express authority. Offenders are liable to the payment of damages. All rights reserved in the event of grant of a patent or the registration of a utility model or design. All specification supplied herein are subject to change without notice at any time.

Copyright © Shanghai SIMCom Wireless Solutions Ltd. 2010

Version History

Version	Chapter	Comments
V0.01	New version	

Contents

General Notes 2
Copyright 2
Version History 3
Contents 4
1. Introduction 5
1.2 Overview 5
1.2 References 5
1.3 Terms and Abbreviations 5
2. Modify Linux Usb Serial Driver 6
2.1 Modify Linux Usb Serial Driver 6
2.2 Build the Driver 6
3. Build Android Image ..... 错误！未定义书签。
3.1 Modify The Framework 8
3.2 Build The Android Image 9

1. Introduction

1.2 Overview

This document is a brief descriptions on:

- How to add code modified by SIM52xx for the android source.

1.2 References

The present document is based on the following documents:

[1] SIMCOM_SIM5218_ATC_EN_V1.11.doc.

1.3 Terms and Abbreviations

For the purposes of the present document, the following abbreviations apply:

- API Application Programming Interface
- CPU Central Processing Unit
- LIB Library
- OS Operating System
- PDU Protocol Data Unit
- RAM Random-Access Memory
- ROM Read-Only Memory
- UMTS Universal Mobile Telecommunications System
- USIM Universal Subscriber Identity Module
- WCDMA Wideband Code Division Multiple Access

2. Modify Linux USB SERIAL Driver

2.1 Modify the driver

In fact the kernel with version of 2.6.28 and later has a common driver named usbserial which can be used by our device .We just need to add the device Vendor ID and Product ID to the driver.

There is just one file need to be modified:

drivers\usb\serial\option.c

The flammulated code is our added.

1) drivers\usb\serial\option.c

```
#define SIMCOM_VENDOR_ID                        0x05c6
#define SIMCOM_PRODUCT_5218                     0x9000
#define SIMCOM_PRODUCT_5210                     0x6000
static struct usb_device_id option_ids[] = {
       { USB_DEVICE(OPTION_VENDOR_ID, OPTION_PRODUCT_COLT) },
       { USB_DEVICE(OPTION_VENDOR_ID, OPTION_PRODUCT_RICOLA) },
       { USB_DEVICE(SIMCOM_VENDOR_ID, SIMCOM_PRODUCT_5218) },
       { USB_DEVICE(SIMCOM_VENDOR_ID, SIMCOM_PRODUCT_5210) },


};
```

2.2 Build the driver

The following is the steps on how to build the driver after getting the source packet.

1) Use “make menuconfig” and select the following issues as a built-in driver:

Device Drivers - à

USB support - à

USB Serial Converter support - à

USB driver for GSM and CDMA modems

After configuration, these items will be configed:

CONFIG_USB = m

CONFIG_USB_SERIAL=m

CONFIG_USB_SERIAL_GENERIC=y

CONFIG_USB_SERIAL_OPTION=y

2) Usb “make zImage” to build the kernel image.

3) If everything goes well, arch/arm/boot/zImage would be generate.

3 Build android

3.1 Modify the Framework

There are three files need to be modified:

hardware/ril/reference-ril/reference-ril.c

hardware/ril/reference-ril/atchannel.c

external/qemu/telephony/android_modem.c

1) hardware/ril/reference-ril/reference-ril.c

```
static void onUnsolicited (const char *s, const char *sms_pdu)
{
    char *line = NULL;
    int err;
    ......
    //line 2068
    //else if (strStartsWith(s, "+CMT:")) {
    else if (strStartsWith(s, "+CMT:")) {
            RIL_onUnsolicitedResponse (
                RIL_UNSOL_RESPONSE_NEW_SMS,
                sms_pdu, strlen(sms_pdu));
        }
    ......
    }
```

2) hardware/ril/reference-ril/atchannel.c

```
/**
 * returns 1 if line is the first line in (what will be) a two-line
 * SMS unsolicited response
 */
static const char * s_smsUnsoliciteds[] = {
//    "+CMTI:",
    "+CMTI:",
    "+CDS:",
    "+CBM:"
};
```

```
void
amodem_receive_sms( AModem  modem, SmsPDU  sms )
{
//#define  SMS_UNSOL_HEADER  "+CMT: 0\r\n"
#define  SMS_UNSOL_HEADER  "+CMTI: 0\r\n"


    if (modem->unsol_func) {
        int    len, max;
        char*  p;
        strcpy( modem->out_buff, SMS_UNSOL_HEADER );
        p   = modem->out_buff + (sizeof(SMS_UNSOL_HEADER)-1);
        max = sizeof(modem->out_buff) - 3 - (sizeof(SMS_UNSOL_HEADER)-1);
        len = smspdu_to_hex( sms, p, max );
        if (len > max) /* too long */
            return;
        p[len]   = '\r';
        p[len+1] = '\n';
        p[len+2] = 0;

        R( "SMS>> %s\n", p );

        modem->unsol_func( modem->unsol_opaque, modem->out_buff );
    }
}
```

3.2 Build the android image

We are building for i.MX51 BBG3 board, "imx51_BBG" is the product names
(see /work/hummer/android /vendor/fsl/product).
After build, check build_imx51_BBG_android.log to make sure no build error.

```
root@ubuntu:/work/hummer/android#make PRODUCT-imx51_BBG-eng 2>&1 | tee build_imx51_BBG_android.log
```

Or

```
root@ubuntu:/work/hummer/android# make PRODUCT-imx51_BBG-eng –j4 2>&1 | tee build_imx51_BBG_android.log
```

Note: ‘N’ is the number of cores of your CPU such as make –j4 to speed up the compiling time.

After completion, the binaries will be generated under
/work/hummer/android /out/target/product/ imx51_BBG /.
Below is the list for reference:

```
root@ubuntu:/work/hummer/android# cd out/target/product/ imx51_BBG /
root@ubuntu: /work/hummer/android/out/target/product/imx51_BBG # ls -al
total 289600
-rw-r--r--    1 root root         7 2010-02-24 22:06 android-info.txt
-rw-r--r--    1 root root        69 2010-02-26 10:05 clean_steps.mk
drwxr-xr-x    2 root root      4096 2010-02-24 20:01 data
-rw-r--r--    1 root root     17618 2010-02-24 22:07 installed-files.txt
drwxr-xr-x 13 root root      4096 2010-02-24 22:06 obj
-rw-r--r--    1 root root        46 2010-02-26 10:05 previous_build_config.mk
-rw-r--r--    1 root root    207488 2010-02-24 22:06 ramdisk.img
drwxr-xr-x    3 root root      4096 2010-02-24 22:06 recovery
-rw-r--r--    1 root root   4702208 2010-02-24 22:06 recovery.img
drwxr-xr-x    8 root root      4096 2010-02-24 20:11 root
drwxr-xr-x    4 root root      4096 2010-02-24 20:11 symbols
drwxr-xr-x 10 root root      4096 2010-02-24 21:53 system
-rw-r--r--    1 root root  81850368 2010-02-24 22:06 system.img
-rw-r--r--    1 root root 209715200 2010-02-24 22:07 userdata.img
root@ubuntu:/work/hummer/android-bak/out/target/product/imx51_BBG#
```

Contact us

Shanghai SIMCom Wireless Solutions Ltd.

Add: Building A, SIM Technology Building, No.633, Jinzhong Road, Changning District 200335

Tel: +86 21 3252 3300

Fax: +86 21 3252 3301

URL: http:/www.sim.com/wm/