密级状态：绝密（　　）　秘密（　　）　内部（ ）　公开（ √ ）

Sofia-3gr 项目 PCBA 测试工具说明

（技术部，系统产品二部）

文件状态： [ ] 正在修改 [√] 正式发布	当前版本：	V1.2
	作　　者：	王剑辉
	完成日期：	2015-06-24
	审　　核：	
	完成日期：	

福州瑞芯微电子有限公司

Fuzhou Rockchips Semiconductor Co . , Ltd

版 本 历 史

版本号	作者	修改日期	修改说明	备注
V1.0	wjh	2015-5-18	初始版本	
V1.1	wjh	2015-6-1	添加 camera 测试和 codec 测试	
V1.2	wjh	2015-6-19	添加 lpsensor 测试和 gps 测试	
V1.3	wjh	2015-6-24	添加 FM 测试和 Udisk 测试	

目 录

1 概述

PCBA 测试工具用于帮助在量产的过程中快速的甄别 PCBA 的好坏，提高生产效率。目前包括屏幕（LCD）、触摸屏测试（TP）、重力感应（gsensor）、无线（wifi）、SD 卡（sdcard）、按键（KEY）测试项目、sim 卡测试（支持双 sim 卡的测试）。

该工具支持通过配置文件 test_config.cfg 对测试项进行配置，具体的配置说明请参第 4 部分“配置文件”

2 PCBA 固件编译打包和运行环境介绍

2.1 编译

PCBA 测试程序位于 Android 源码/extenal/rk-pcba-test 目录下，编译会生成 pcba_core 可执行文件，pcba_core 和 rk-pcab-test/res 下的相关文件在编译的时候会被自动拷贝到 system 的 bin 目录下。

编译说明：

一.进入sofia-3gr项目代码的跟目录下面，执行source build/envsetup.sh；然后执行lunch，会显示以下几个配置项：1. mini_emulator_x86_64-userdebug

2. mini_emulator_x86-userdebug

3. sofia3gr-user

4. sofia3gr-userdebug

5. sofia3gr-eng

这里我们选择5

二．进入/external/rk-pcba-test/目录下面，执行mm -B编译生成pcba_core可执行文件。

2.2 打包

由于 PCBA 运行于 ptest 模式中，因此我们打包固件时必须把 pcba 工具里面使用到的 sh 脚本文件和 .cfg 文件等打包到固件里面，生成一个 system.fls 文件。

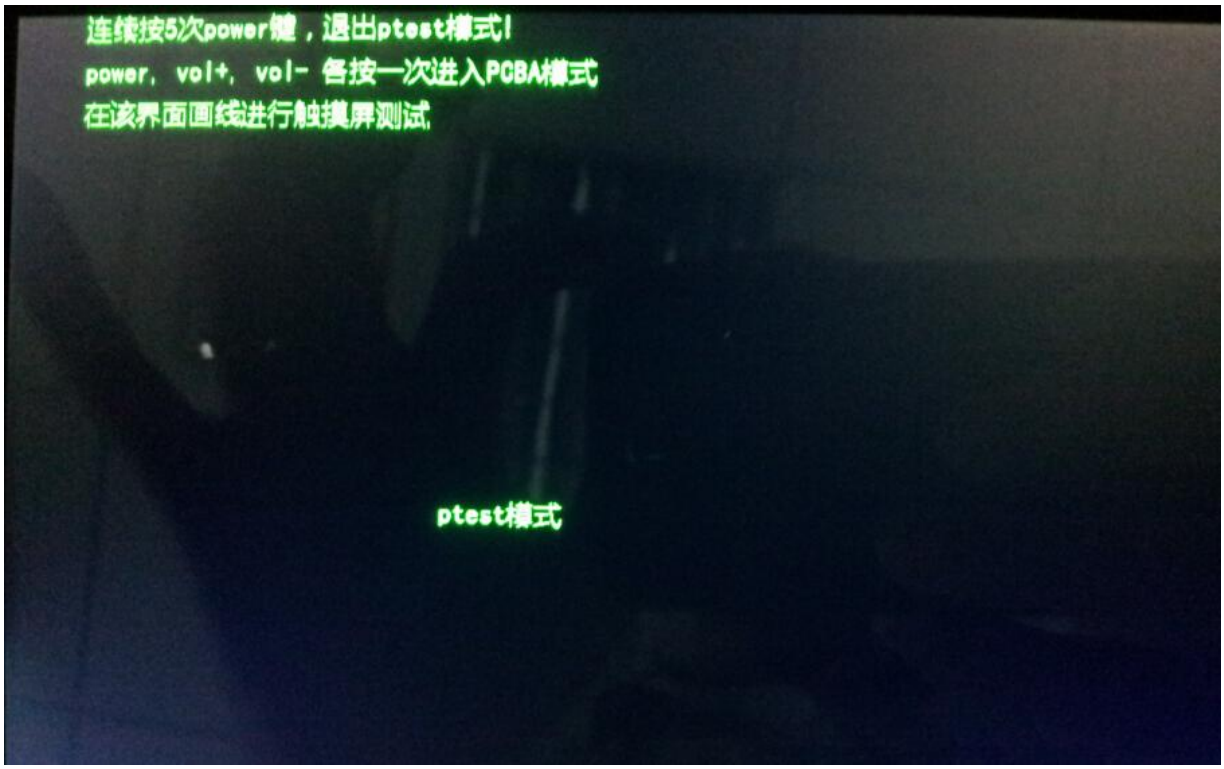
打包的方法：进入根目录下面，执行./mkimage.sh。把 pcba_core 和资源文件一起打包到固件里面。

2.3 使用说明介绍

Sofia-3gr 量产流程介绍：

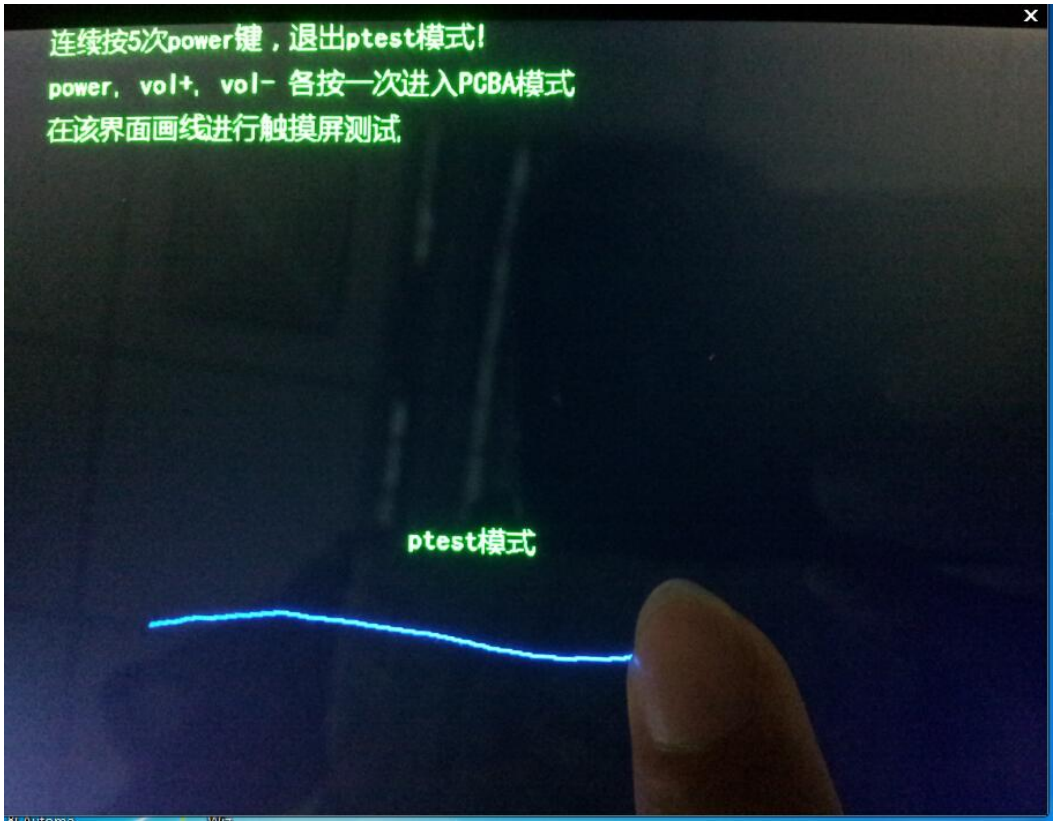
1. 升级烧写固件（成功后，设置下次启动进入 ptest 模式）
2. 写 pcb_sn 号、wifi 校准、2G/3G 校准、写默认的 IMEI 号（成功后，根据下一步功能测试工具的需要来确定进 ptest 模式还是进 Android 模式。假如客户选择用 PCBA 测试工具测试，那么校准成功后应该继续保持进入 ptest 模式；假如客户选择用 devicetest.apk 来测试功能，那么校准成功后必须选择设置下次启动模式为 Android 模式）
3. pcba 功能测试：可选择 pcba 测试工具或者用 devicetest.apk 测试工具。注意：pcba 测试工具是工作在 ptest 模式的，devicetest.apk 是工作在 android 模式的（测试成功后，设置下次启动模式为 Android 模式）
4. 组装整机，并使用 devicetest.apk 测试工具进行整机测试（测试通过后，设置下次启动模式为 ptest 模式）
5. 写 DEV_SN、写 MAC 地址、写 IMEI 号（写号成功后设置启动模式为 Android 模式）

PCBA 程序运行于 ptest 模式中，具体测试流程为：开机进入 Ptest 模式，启动 PCBA 工具，进入 PCBA 首界面（RF 校准就是在此界面下进行）。首界面如下：



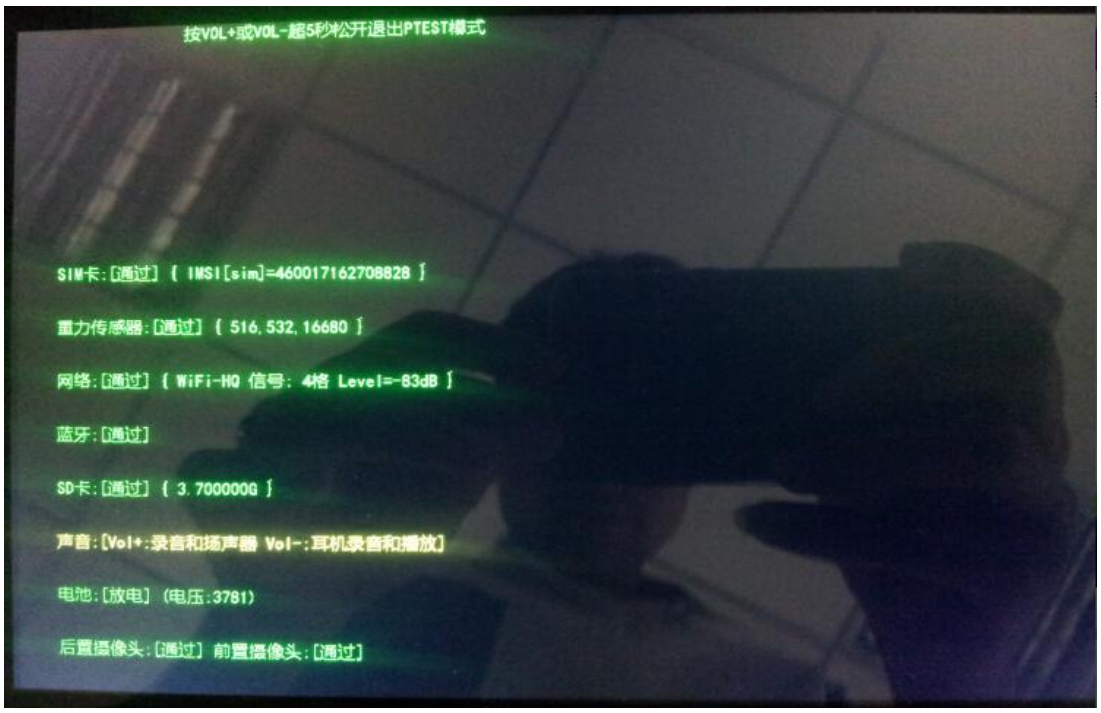
首界面里面有以下几个功能：

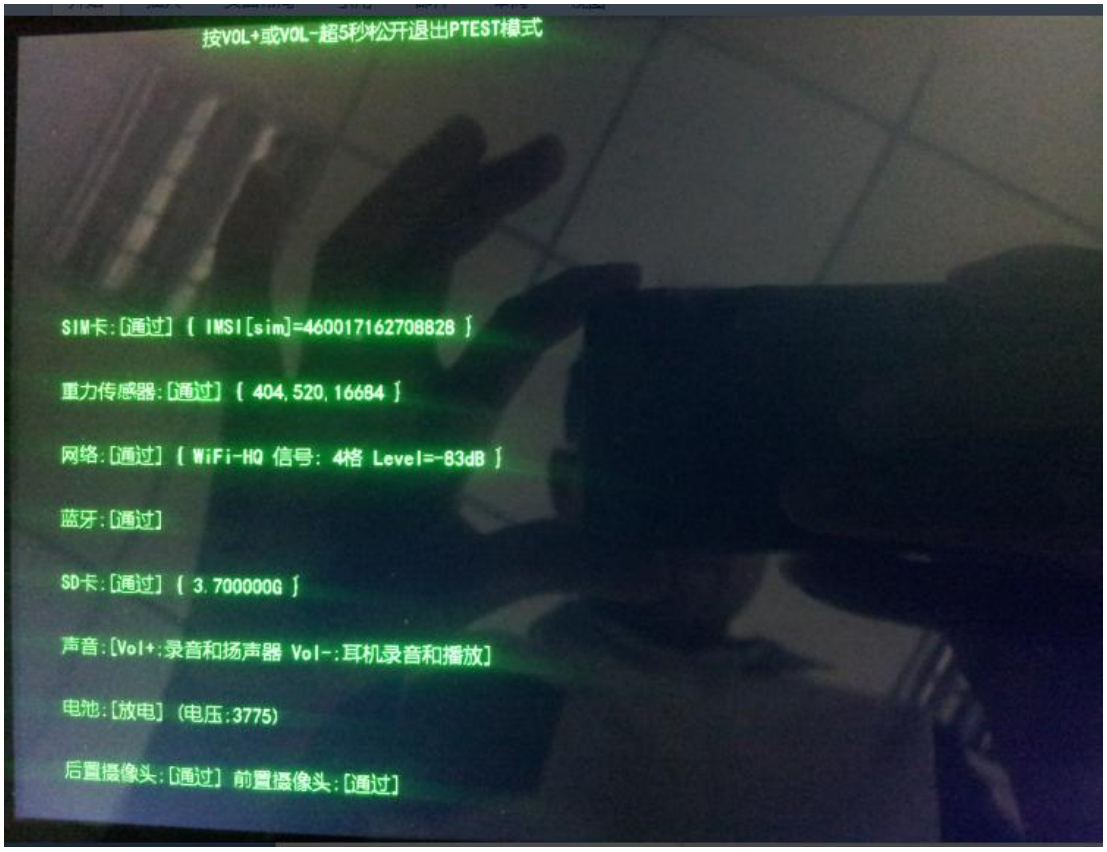
1. 连续按 5 次 power 键，可以退出 ptest 模式，进入 Android 模式。
2. 连续按 power、vol+、vol-键，进入 pcba 功能测试界面，进行 pcba 功能测试（这个功能也进行了 key 按键的测试）
3. 在该界面进行 tp 测试，手指可以随意在屏幕上划动。如下图：



PCBA 功能测试界面：

当在首界面中，连续按 power、vol+、vol- 三个按键时，pcba 测试工具会自动进入功能测试界面。功能测试界面如下图：





全部测试项通过之后，可以长按 vol+、vol-其中的任意一个按钮 5s，系统会退出 ptest 模式，进入 Android 模式。

3 测试项

3.1 测试项分类说明：

测试项分别有 “红”,“黄”,“绿” 三种颜色表示不同的测试状态

黄色：未测试项或者正在测试的项

绿色：测试通过项（音频测试项表示测试完成）

红色：测试未通过项

3.2 测试项详细说明：

3.2.1 重力感应(gsensor)测试

Gsensor 为自动测试项，显示读取到的 Gsensor 坐标，成功读取数据即测试成功。

3.2.2 无线网络（wifi 测试）

Wifi 为自动测试项，会自动扫描周边的 AP，显示第一个扫描到的的那个 AP 名字和信号强度。

3.2.3 SD 卡（sdcard）测试

Sdcard 为自动测试项，插入 sdcard，如果 SD 卡正常识别到，则会提示测试成功。如果插入 sdcard 卡没有识别到，再次拔插 sdcard，2s 后还是没有提示测试成功，则 SD 卡测试失败。

SD card 必须为 FAT32 格式，不支持其他格式！整个卡只能包含一个分区。如果不符合要求，请通过格式化来格式成标准格式。

3.2.4 屏幕（LCD）测试

LCD 为自动测试项，开机屏幕正常显示，就说明 LCD 贴片正常，（同时测试显示的字体都有颜色），需要测试人员判断 LCD 显示是否正常。

3.2.5 按键（KEY）测试

按键为手动测试项目，PCBA 测试工具从首界面连续按 power、vol+、vol-按键进入到功能测试界面，这个过程中就实现了 KEY 的测试。

3.2.6 TP 测试

TP 为手动测试，直接在屏幕上画线，lcd 会根据手指画出划动轨迹。

3.2.7 SIM 卡测试

SIM 卡为自动测试，插入 SIM 卡。PCBA 会自动读取 SIM 卡里面的 IMSI 信息，并把 IMSI 信息显示到界面上(这个测试项前提必须要在校准工位上写一个默认的 IMEI 号)，目前最新版本直接双卡测试，需要自行配置 config 文件。

配置单卡槽或者双卡槽的方法：打开 external\rk-pcba-test\res\下面的 test_config_sofia.cfg 文件找到 sim 项，sim_counts =1，表示单卡槽；sim_counts = 2 表示双卡槽，只需要修改这个属性即可。

3.2.8 Camera 测试

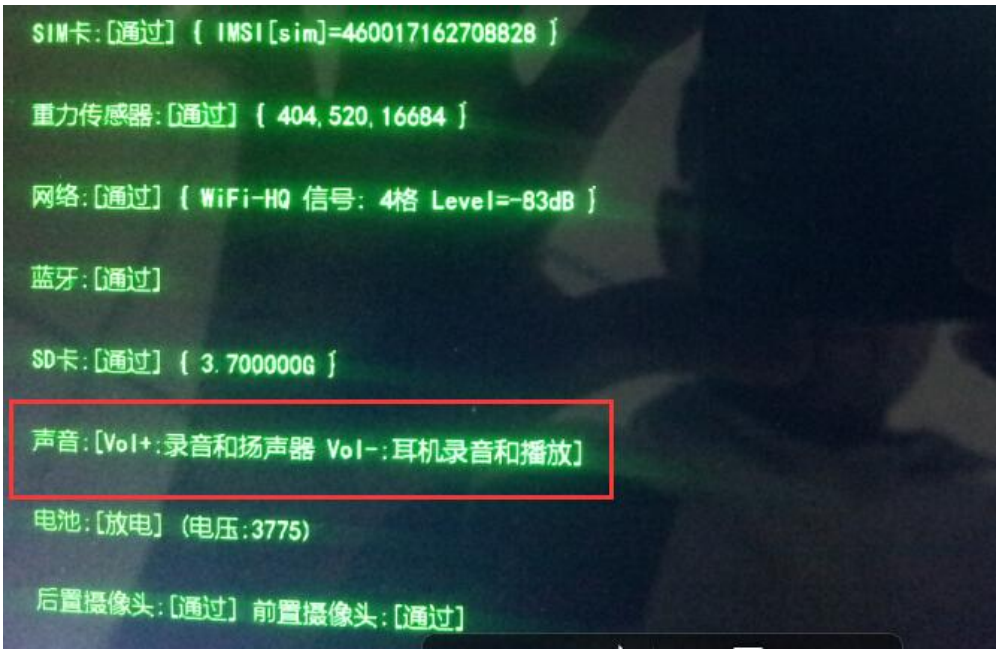
Camera 测试为自动测试。PCBA 会自动从前后置摄像头里面采集一帧数据，采集到数据，即 Camera 测试成功。

3.2.9 Codec 测试

Codec 测试为自动测试。PCBA 会自动判断当前测试板是否有听筒（EP），有听筒的测试方式：按 power 键，在听筒有播放音乐；按 vol+键，先录音后播放录制的音频；

按 vol-键，测试耳机录音和播放。没有听筒的测试方式：按 vol+键，先录音后播放录制的音频文件；按 vol-键，测试耳机录音和播放。当 Codec 测试完后，Codec 提示字体变成蓝色。

具体操作有界面提示如下图：



3.2.10 光感(lsensor)测试

lsensor 为自动测试项，显示读取到的 lsensor 值，成功读取数据即测试成功，否则测试失败。

3.2.11 靠近（psensor）测试

psensor 位自动测试项，显示读取到的 psensor 值，成功读取到数据即测试成功，读取失败即测试失败。

3.2.12 GPS（gnss）测试

GPS 测试是自动测试项，这项测试会自动去搜索卫星，当搜索到卫星（1 颗以上），即可判定 GPS 模块贴片没问题。

3.2.13 FM（收音机）测试

FM 测试是自动测试项，进入 pcba 测试之前，必须先接上耳机（因为 FM 播放时，耳机做天线）。这项测试原理是通过发送 AT 指令打开 FM，然后扫描周围的频点，扫描完后，打开扫描到的第一个频道，当第一个频道的 rssi 超过 5dBuV，测试通过。

3.2.14 U 盘（udisk）测试

U 盘测试为自动测试项，插入 U 盘或者插入 OTG 转接线，如果 U 盘正常识别到，则会提示测试成功，否则失败。

所有项测试完成后，请长按 vol+或者 vol-中的一个按键 5s 后松开，则停止测试，把测试结果写入 nvm,退出 ptest 模式,移除 sdcard 和 sim 卡,然后系统会进入 Android 模式。

查看测试结果的方法：

1. 进入 Android 系统，打开计算器，输入（83991906）=调出 Devicetest.apk，点击“检验结果”（Examine Result）的按钮，会显示 pcba 测试结果。
2. 写号工具查看：打开写号工具，假如你不想写号，可以在设置里面把 SN/MAC/IMEI 的选择框勾去掉，然后点击按钮，即可查看 pcba 测试结果。

4 配置文件

PCBA 所有的测试项目通过一个配置脚本 test_config.cfg 来配置，位于 Androidsrc/external/rk-pcba-test/res/test_config.cfg，用户可以根据项目的硬件配置来配置 test_config.cfg 文件，决定要对哪些模块进行测试，以及给自己的测试程序传递相关的参数。

该脚本使用 ini 文件格式，由段、键和值三者组成，通常一个段表示一个模块配置。目前要求该配置文件使用 UTF-8 编码，其他编译格式可能会导致未知错误。

模块配置示例：

测试模块配置模板

```
[example]
display_name= "Example"
activated = 1
program = "example.sh"
category = 0
```

（1）[example]

Example 表示一个配置模块的名称，如果是cfg文件中自带的模块名称，则 不能改动，否则会导致某个测试项不被测试系统启动。

（2）display_name

display_name表示该测试模块在屏幕上显示的名称，可以根据自己的需要修改。该名称最长为64字节，如果为空，则测试程序不会运行。

（3）activated

activated表示是否测试该模块

0：不测试该模块

1：测试该模块

（4）program

该键值目前没用到，可以不用配置

（5）category

category 表示测试方式

0：自动测试

1：手动测试

屏幕测试

```
[Lcd]
display_name= "lcd"
activated     = 1                    //测试该项
program       = "lcdtester.sh"
category      = 0                    //自动测试
run_type      = 1
```

实时时钟测试

```
[rtc]
display_name= "rtc"
activated     = 1                    //测试该项
program       = "rtctester.sh"
category      = 0                    //自动测试
run_type      = 1
module_args = "20121113.160145" //测试rtc的时候  设置的时间
```

无线测试

```
[wifi]
```

display_name= "wlan"

activated = 1 //测试该项

program = "wifitester.sh"

category = 0 //自动测试

run_type = 1

module_path = "/system/vendor/modules/8192cu.ko"

module_args =

WiFi测试，测试结果测试如下：

“网络：[通过] {“testap” 信号强度 4 格 }”

信号强度为实际扫描到的AP的信号强度，与Android上一样，分为0到4格。

重力感应测试

[gsensor]

display_name= "gsensor"

activated = 1 //测试该项目

program = "gsensortester.sh"

category = 0 //自动测试

run_type = 1

蓝牙测试

[bluetooth]

display_name= "bluetooth"

activated = 1

program =

category =

run_type = 1

chip_type = "sofia-3gr" ; rk903, mt6622, rda587x, rda5990,rtk8723as

SD卡测试

[sdcard]

display_name= "SDcard"

activated = 1 //测试该项目

program = "mmctester_sofia.sh"

category = 0 //自动测试

run_type = 1

该配置脚本可以扩展，如果某个模块需要通过配置脚本传递相关参数，可以扩展相关的键值，比如RTC配置项如下

实时时钟测试

[rtc]

display_name= "rtc"

activated = 1 //测试该项

program = "rtctester.sh"

category = 0 //自动测试

run_type = 1

module_args = "20121113.160145" //测试rtc的时候 设置的时间

在具体的测试程序中，可以通过script_fetch api获得设置的相关键值：

int script_fetch(char *main_name, char *sub_name, int value[], int count)

main_name：测试模块的名称，在test_config.cfg文件中[xxxx]

sub_name:键值，比如activated、display_name、module_args等等。

if(script_fetch("rtc", "module_args", (int *)dt, 8) == 0)

```
{
    trncpy(s, dt, 32);
}
```

这里，可获取在配置文件中设置的rtc测试时module_args设置的值。

测试程序中可以通过ui_print_xy_rgba()接口，打印测试结果到屏幕上，由于屏幕空间有限，原则上，尽量打印简单的结果，一个测试项打印一行，成功用蓝色打印，失败用红色打印。

5 字体

说明：PCBA 2.0以后的版本增加了对中文的支持，并可以支持多种字体大小的配置，包括18*18, 20*20, 24*24, 28*28, 32*32, 36*36,可以通过修改minuitwrp/graphics.c 的头文件来包含修改使用不同大小的字库

(输出到屏幕的中文必须是UTF-8编码格式)

6 测试样例扩展

该测试程序允许用户扩展自己的测试样例。如果因为项目需要，用到了该测试程序中目前还未支持到的模块，可以自己添加测试程序，然后集成到测试框架中。

集成方法如下：

（1）先写好自己的测试程序和头文件。测试程序要封装成 void * xxxx_test(void *argv)格式的接口。

（2）并在 test_config.cfg 里面加入想要的配置。

（3）在pcba测试程序启动的时候，会作为一个线程去启动所有的测试代码，需要在pretest.c的start_test_pthread（）函数中注册自己的测试函数：

```
int start_test_pthread(struct testcase_info *tc_info)
{
    int err;
    printf("%s\n",tc_info->base_info->name);

    if(!strcmp(tc_info->base_info->name, "Lcd"))
    {
        err = pthread_create(&screen_tid, NULL, screen_test,tc_info); //
        if(err != 0)
        {
               printf("create screen test thread error: %s/n",strerror(err));
               return -1;

        }
    }
    else if(!strcmp(tc_info->base_info->name, "rtc"))
    {
        err = pthread_create(&rtc_tid, NULL, rtc_test,tc_info); //
        if(err != 0)
        {
           printf("create rtc test thread error: %s/n",strerror(err));
           return -1;

        }
```