

Rockchip

# DDR颗粒验证流程说明

发布版本:1.2

作者邮箱:[cym@rock-chips.com](mailto:cym@rock-chips.com)

日期:2018.10.11

文件密级:公开资料

## 前言

对各个芯片平台DDR颗粒兼容性和稳定性的验证流程进行说明。文档分为linux 3.10，linux 4.4，RV1108，RK3308三个章节，请根据实际测试平台情况选择对应章节进行参考。

### 产品版本

芯片名称	内核版本
所有芯片	所有内核版本

### 读者对象

本文档（本指南）主要适用于以下工程师：

品质测试工程师

技术支持工程师

软件开发工程师

### 修订记录

日期	版本	作者	修改说明
2017.11.21	V1.0	陈有敏	初版
2018.03.22	V1.1	陈有敏	公开资料
2018.10.11	V1.2	何智欢	增加RK3308颗粒验证流程说明

[DDR颗粒验证流程说明](#)

[前言](#)

**NOTE**

[Linux 3.10 DDR颗粒验证流程说明](#)

- Linux 3.10 测试固件编译
- Linux 3.10 测试环境搭建
  - 固件烧写
  - 自动搭建测试环境
  - 手动搭建测试环境
- Linux 3.10 确认容量是否正确
- Linux 3.10 定频测试
- Linux 3.10 变频测试
- Linux 3.10 reboot拷机

#### Linux 4.4 DDR颗粒验证流程说明

- Linux 4.4 测试固件编译
- Linux 4.4 测试环境搭建
  - 固件烧写
  - 自动搭建测试环境
  - 手动搭建测试环境
  - 通过U盘和串口搭建测试环境
- Linux 4.4 确认颗粒容量
- Linux 4.4 定频拷机
- Linux 4.4 变频拷机
- Linux 4.4 reboot拷机
- Linux 4.4 sleep拷机（optional）

#### RV1108 DDR颗粒验证流程说明

- RV1108 测试固件编译
- RV1108 测试环境搭建
- RV1108 确认容量是否正确
- RV1108 定频测试
- RV1108 变频测试
- RV1108 reboot拷机

#### RK3308 DDR颗粒验证流程说明

- RK3308 确定容量是否正确
- RK3308 拷机测试
- RK3308 休眠唤醒测试
- RK3308 reboot拷机

---

## NOTE

---

- RV1108平台DDR颗粒验证流程与其它平台不同，RV1108请详见本文档的"RV1108 DDR颗粒验证流程说明"章节；RK3308请详见本文档的"RK3308 DDR颗粒验证流程说明"章节。其他平台，请根据linux kernel版本是linux 3.10还是linux 4.4，选择对应章节进行参考。
- 本文中所述颗粒验证过程需要的DDR测试资源文件随该文档提供。

---

## Linux 3.10 DDR颗粒验证流程说明

---

### Linux 3.10 测试固件编译

---

配置kernel代码的menuconfig，进入System Type，选择打开DDR Test和pm\_tests。

```
1 menuconfig
2 System Type --->
3 [*] /sys/pm_tests/ support
4 [*] DDR Test
```

如果menuconfig中没有 `[*] /sys/pm_tests/ support` 选项，请参考《DDR开发指南》的"DDR如何定频"和"如何enable/disable kernel中的DDR变频功能"章节，分别编译定频固件和变频固件。

## Linux 3.10 测试环境搭建

### 固件烧写

测试开始前，需要先明确测试过程需要的如下信息：

1. 测试固件android版本信息（eg: android4.4, android5.0, android6.0, android7.1 ...）
2. 测试固件操作系统位数（eg: 32bit or 64bit）
3. 测试机器DDR总容量（eg: 512MB or 1GB or 2GB ...）
4. 定频测试，DDR要跑的最高频率（eg: 456MHz or 533MHz ...）
5. 变频测试，DDR要跑的频率范围（eg: 200MHz - 456MHz or 200MHz-533MHz ...）

### 自动搭建测试环境

进入DDR测试资源文件的"linux3.10\_ddr\_test\_files"目录，直接双击push\_files.bat脚本，根据脚本提示和固件类型信息进行选择输入1或者2，自动完成测试环境搭建。自动搭建无异常，可以跳过下面的"手动搭建测试环境"这一章节。

### 手动搭建测试环境

如果自动搭建测试环境失败，可以通过手动搭建来完成。请选择"linux3.10\_ddr\_test\_files"目录里的测试文件进行安装。

1. 安装捕鱼达人

```
1 <adb_tool> adb.exe install fishingjoy1.apk
```

2. push google stressapptest 请根据测试机器的固件android版本是不是android4.4，选择对应版本的stressapptest进行push。

```
1 <adb_tool> adb.exe root
2 <adb_tool> adb.exe remount
3 <adb_tool> adb.exe push libstlport.so /system/lib/libstlport.so
4 <adb_tool> adb.exe shell chmod 644 /system/lib/libstlport.so
```

- 如测试机器的固件android版本是android4.4，则选择对应的stressapptest\_android4.4进行push

```
1 <adb_tool> adb.exe push stressapptest_android4.4 /data/stressapptest
2 <adb_tool> adb.exe shell chmod 0777 /data/stressapptest
```

- 如测试机器的固件android版本不是android4.4，则选择对应的stressapptest进行push

```
1 <adb_tool> adb.exe push stressapptest /data/stressapptest
2 <adb_tool> adb.exe shell chmod 0777 /data/stressapptest
```

3. push memtester 请根据测试机器的固件为linux 64bit还是linux 32bit，选择对应的memtester进行push。  
Eg:

- 如测试机器的固件为linux 32bit，则选择对应的memtester\_32bit

```
1 <adb_tool> adb.exe push memtester_32bit /data/memtester
2 <adb_tool> adb.exe shell chmod 644 /data/memtester
```

- 如测试机器的固件为linux 64bit，则选择对应的memtester\_64bit

```
1 <adb_tool> adb.exe push memtester_64bit /data/memtester
2 <adb_tool> adb.exe shell chmod 644 /data/memtester
```

4. sync

```
1 <adb_tool> adb.exe shell sync
```

## Linux 3.10 确认容量是否正确

通过 `<rkxxxx:/ $> cat /proc/meminfo` 查看MemTotal容量是否与测试机器实际容量相符。log eg:

```
1 <rkxxxx:/ $> cat /proc/meminfo
2 MemTotal:      2038904 kB
```

512MB 约等于533504kB

1GB 约等于1048576kB

1.5GB 约等于1582080kB

2GB 约等于2097152kB

3GB 约等于3145728kB

4GB 约等于4194304kB

由于系统内存分配管理差异的原因，得到的容量有些偏差，属于正常。

## Linux 3.10 定频测试

1. 开启捕鱼达人APK
2. 串口控制台上输入su命令

```
1 <rkxxxx:/ $> su
```

3. DDR定频到拷机频率

请根据测试机器所支持的DDR最高频率，进行设置。

Eg:

如果测试机器所支持的DDR最高频率为533MHz。

```
1 | <rkxxxx:/ #> echo set clk_ddr 533000000 > /sys/pm_tests/clk_rate
```

4. google stressapptest拷机，拷机时间12小时以上

- 如果总容量是512MB则申请64MB进行stressapptest

```
1 | <rkxxxx:/ #> /data/stressapptest -s 43200 -i 4 -c 4 -w --stop_on_errors -M 64
```

- 如果总容量是1GB则申请128MB进行stressapptest

```
1 | <rkxxxx:/ #> /data/stressapptest -s 43200 -i 4 -c 4 -w --stop_on_errors -M 128
```

- 如果总容量是2GB则申请256MB进行stressapptest

```
1 | <rkxxxx:/ #> /data/stressapptest -s 43200 -i 4 -c 4 -w --stop_on_errors -M 256
```

- 如果总容量是4GB则申请512MB进行stressapptest

```
1 | <rkxxxx:/ #> /data/stressapptest -s 43200 -i 4 -c 4 -w --stop_on_errors -M 512
```

5. 确认拷机结果 拷机结束，确认机器是否正常，捕鱼达人是否正常运行，stressapptest结果是PASS还是FAIL。stressapptest每隔10秒会打印一条log，log显示测试剩余时间。测试完成后会打印测试结果，如果测试通过打印Status: PASS，测试失败打印Status: FAIL。

6. memtester拷机，拷机时间memtester 12小时以上

- 如果总容量是512MB则申请64MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 64m
```

- 如果总容量是1GB则申请128MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 128m
```

- 如果总容量是2GB则申请256MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 256m
```

- 如果总容量是4GB则申请512MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 512m
```

7. 确认拷机结果 拷机结束，确认机器是否正常，捕鱼达人是否正常运行，memtester是否在正常运行。DDR测试资源文件目录里的memtester程序进行过修改，测试过程如果有发现错误会自动停止测试，如果持续测试12小时以上后，memtester仍在继续运行，说明测试过程没有发现错误。

- memtester运行过程如果没有发现错误，会持续打印如下log:

```
1 | Loop 10:
2 |   Stuck Address      : ok
3 |   Random Value       : ok
4 |   Compare XOR        : ok
5 |   Compare SUB        : ok
6 |   Compare MUL        : ok
7 |   Compare DIV        : ok
8 |   Compare OR         : ok
9 |   Compare AND        : ok
10 | Sequential Increment: ok
11 | Solid Bits          : ok
12 | Block Sequential    : ok
13 | Checkerboard        : ok
14 | Bit Spread          : ok
15 | Bit Flip            : ok
16 | walking Ones        : ok
17 | walking Zeroes      : ok
```

- memtester运行过程如果有发现错误，会自动停止测试并退出，退出时打印如下log:

```
1 | FAILURE: 0xffffffff != 0xffffbfff at offset 0x03b7d9e4.
2 | EXIT_FAIL_OTHERTEST
```

## Linux 3.10 变频测试

如果机器前面做过定频测试，要重启机器，否则后续的变频命令会无法进行。

1. 开启捕鱼达人APK
2. 串口控制台上输入su命令

```
1 | <rkxxxx:/ $> su
```

3. 后台执行memtester

- 如果总容量是512MB则申请64MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 64m > /data/_log.txt &
```

- 如果总容量是1GB则申请128MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 128m > /data/memtester_log.txt &
```

- 如果总容量是2GB则申请256MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 256m > /data/memtester_log.txt &
```

- 如果总容量是4GB则申请512MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 512m > /data/memtester_log.txt &
```

4. 执行变频命令 根据测试机器支持的DDR频率范围进行设置。 Eg: 如果测试机器支持的DDR频率范围为200MHZ到533MHz。

```
1 | <rkxxxx:/ #> echo 'a:200M-533M-1000000T' > proc/driver/ddr_ts
```

5. 确认拷机结果，拷机时间12小时以上

- 确认捕鱼达人是否正常运行，机器是否正常
- 确认变频程序运行是否正常，变频log是否在正常打印
- 确认memtester是否正常运行 在串口输入 <rkxxxx:/ #> ps | grep memtester，看memtester进程是否存在。 Eg:

```
1 | <rkxxxx:/ #> ps | grep memtester
2 | root      14309 1730  74332  68156      0 5e980bf564 R /data/memtester
```

## Linux 3.10 reboot拷机

开启计算器 Calculator,输入 "839910906=",点击"RebootTest", 拷机时间12小时以上。

## Linux 4.4 DDR颗粒验证流程说明

### Linux 4.4 测试固件编译

使能DDR变频功能，打开测试机器对应的板级DTS文件，找到dfi和dmc节点，配置status = "okay"。

```
1 | &dfi {
2 |     status = "okay";
3 | };
4 |
5 | &dmc {
6 |     status = "okay";
7 |     .....
8 | };
```

关于测试固件编译这里只做简单说明，详细介绍请参考《DDR开发指南》的"如何enable/disable kernel中的DDR变频功能"章节。

### Linux 4.4 测试环境搭建

#### 固件烧写

测试开始前，需要先明确测试过程需要的如下信息：

1. 测试固件操作系统位数（32bit or 64bit）

2. 测试机器DDR总容量（512MB or 1GB or 2GB ...）
3. 定频测试，DDR要跑的最高频率（456MHz or 533MHz ...）

## 自动搭建测试环境

1. 进入DDR测试资源文件的"linux4.4\_ddr\_test\_files"目录，双击push\_files.bat脚本文件，根据脚本提示和固件类型信息进行选择输入1或者2，自动完成测试环境搭建。Note:运行脚本后，需要通过打印的log检查是否每项都有被正常执行，确认是否有报错信息。
2. 自动搭建无异常，可以跳过下面的"手动搭建测试环境"这一章节。

## 手动搭建测试环境

如果自动搭建测试环境失败，可以通过手动搭建来完成。请选择"linux4.4\_ddr\_test\_files"目录里的测试文件进行安装。

1. 安装捕鱼达人

```
1 | <adb_tool> adb.exe install fishingjoy1.apk
```

2. push google stressapptest

```
1 | <adb_tool> adb.exe root
2 | <adb_tool> adb.exe disable-verity
3 | <adb_tool> adb.exe reboot
4 | /* 等机器完成重启，adb出来后，再输入 */
5 | <adb_tool> adb.exe root
6 | <adb_tool> adb.exe push stressapptest /data/stressapptest
7 | <adb_tool> adb.exe shell chmod 0777 /data/stressapptest
8 | <adb_tool> adb.exe remount
9 | <adb_tool> adb.exe push libstlport.so /system/lib/libstlport.so
10 | <adb_tool> adb.exe shell chmod 644 /system/lib/libstlport.so
11 | <adb_tool> adb.exe shell sync
```

3. push memtester 请根据测试机器固件为linux 64bit还是linux 32bit，选择对应的memtester进行push。Eg:

- 如测试固件为linux 32bit，则选择对应的memtester\_32bit

```
1 | <adb_tool> adb.exe push memtester_32bit /data/memtester
2 | <adb_tool> adb.exe shell chmod 644 /data/memtester
```

- 如测试固件为linux 64bit，则选择对应的memtester\_64bit

```
1 | <adb_tool> adb.exe push memtester_64bit /data/memtester
2 | <adb_tool> adb.exe shell chmod 644 /data/memtester
```

4. push ddr\_freq\_scan.sh脚本

```
1 | <adb_tool> adb.exe push ddr_freq_scan.sh /data/ddr_freq_scan.sh
2 | <adb_tool> adb.exe shell chmod 0777 /data/ddr_freq_scan.sh
```



## 通过U盘和串口搭建测试环境

如果测试机器的ADB无法连接，可以通过U盘将测试过程需要用到文件拷贝到测试板，然后通过串口搭建测试环境。

### 1. 准备工作

- 开机后，添加wake\_lock防止机器进入二级待机 `echo 1 > /sys/power/wake_lock`，或者通过设置 Setting->Display->Sleep->Never sleep让机器保持唤醒状态
- U盘接入电脑，将"linux4.4\_ddr\_test\_files"目录拷贝到U盘
- 测试板串口控制台上输入su命令

```
1 | <rk3399:/ $> su
```

- U盘接入测试板，将"linux4.4\_ddr\_test\_files"目录下的测试所需文件拷贝到机器/data目录。U盘正常是加载在/mnt/media\_rw/\*\*\*目录下（\*\*\*：每块U盘的挂载节点名有不同，请用tab键补全）。Eg:

```
1 | <rk3399:/ #> cp /mnt/media_rw/B4FE-5315/linux4.4_ddr_test_files/* /data/
```

### 2. 自动搭建测试环境

```
1 | <rk3399:/ #> chmod 777 /data/test_files_install.sh
2 | <rk3399:/ #> /data/test_files_install.sh
```

需要通过打印的log检查是否每项都有被正常执行。自动搭建无异常，可以跳过下面的"手动搭建测试环境"这一部分。

### 3. 手动搭建测试环境 如果自动搭建测试环境失败，可以通过手动搭建来完成。

- 将libstlport.so拷贝到/system目录

```
1 | <rk3399:/ #> mount -o rw,remount /system
2 | <rk3399:/ #> cp /data/libstlport.so /system/lib/
3 | <rk3399:/ #> chmod 644 /system/lib/libstlport.so
```

- 更改文件权限

```
1 | <rk3399:/ #> chmod 777 /data/memtester /data/stressapptest /data/ddr_freq_scan.sh
```

- 安装捕鱼达人APK

```
1 | <rk3399:/ #> pm install /data/fishingjoy1.apk
```

- sync

```
1 | <rk3399:/ #> sync
```

## Linux 4.4 确认颗粒容量

通过 `<rkxxxx:/ #> cat /proc/meminfo` 查看MemTotal项所示容量是否与测试机器DDR总容量一致。log eg:

```
1 | rkxxxx:/ # cat /proc/meminfo
2 | MemTotal:      2038904 kB
```

512MB 约等于533504kB

1GB 约等于1048576kB

1.5GB 约等于1582080kB

2GB 约等于2097152kB

3GB 约等于3145728kB

4GB 约等于4194304kB

由于系统内存分配管理差异的原因，得到的容量有些偏差，属于正常。

## Linux 4.4 定频拷机

1. 开启捕鱼达人APK
2. 先输入su命令

```
1 | <rkxxxx:/ $> su
```

3. 定频到拷机频率 根据测试机器DDR要跑的最高频率进行设置。Eg:

- 如果是跑928MHz

```
1 | <rkxxxx:/ #> /data/ddr_freq_scan.sh 933000000
```

- 如果是跑800MHz

```
1 | <rkxxxx:/ #> /data/ddr_freq_scan.sh 800000000
```

- 如果是跑600MHz

```
1 | <rkxxxx:/ #> /data/ddr_freq_scan.sh 600000000
```

4. 通过log确认频率是否正确 log eg:

```
1 | 130|rkxxxx:/ # /data/ddr_freq_scan.sh 800000000
2 | already change to 800000000 done.
3 | change frequency to available max frequency done.
```

5. google stressapptest拷机，拷机时间12小时以上

- 如果总容量是512MB则申请64MB进行stressapptest

```
1 | <rkxxxx:/ #> /data/stressapptest -s 43200 -i 4 -c 4 -w --stop_on_errors -M 64
```

- 如果总容量是1GB则申请128MB进行stressapptest

```
1 | <rkxxxx:/ #> /data/stressapptest -s 43200 -i 4 -c 4 -w --stop_on_errors -M 128
```

- 如果总容量是2GB则申请256MB进行stressapptest

```
1 | <rkxxxx:/ #> /data/stressapptest -s 43200 -i 4 -c 4 -w --stop_on_errors -M 256
```

- 如果总容量是4GB则申请512MB进行stressapptest

```
1 | <rkxxxx:/ #> /data/stressapptest -s 43200 -i 4 -c 4 -w --stop_on_errors -M 512
```

6. 确认拷机结果 拷机结束，确认机器是否正常，捕鱼达人是否正常运行，stressapptest结果是PASS还是FAIL。  
stressapptest 每隔10秒会打印一条log，log显示测试剩余时间。测试完成后会打印测试结果，如果测试通过打印Status: PASS，测试失败打印Status: FAIL。

7. memtester拷机，拷机时间12小时以上

- 如果总容量是512MB则申请64MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 64m
```

- 如果总容量是1GB则申请128MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 128m
```

- 如果总容量是2GB则申请256MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 256m
```

- 如果总容量是4GB则申请512MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 512m
```

8. 确认拷机结果 拷机结束，确认机器是否正常，捕鱼达人是否正常运行，memtester是否正常运行。"DDR测试资源文件"目录里的memtester程序进行过修改，测试过程如果有发现错误会自动停止测试，如果持续测试12小时后，memtester仍在继续运行，说明测试过程没有发现错误。

- memtester运行过程如果没有发现错误，会持续打印如下log:

```
1 | Loop 10:
2 |   Stuck Address      : ok
3 |   Random value      : ok
4 |   Compare XOR       : ok
5 |   Compare SUB       : ok
6 |   Compare MUL       : ok
7 |   Compare DIV       : ok
```

```
8 | Compare OR : ok
9 | Compare AND : ok
10 | Sequential Increment: ok
11 | Solid Bits : ok
12 | Block Sequential : ok
13 | Checkerboard : ok
14 | Bit Spread : ok
15 | Bit Flip : ok
16 | walking Ones : ok
17 | walking Zeroes : ok
```

- memtester运行过程如果有发现错误，会自动停止测试并退出，退出时打印如下log:

```
1 | FAILURE: 0xffffffff != 0xffffbfff at offset 0x03b7d9e4.
2 | EXIT_FAIL_OTHERTEST
```

## Linux 4.4 变频拷机

1. 开启捕鱼达人APK
2. 先输入su命令

```
1 | <rkxxxx:/ $> su
```

3. 后台执行memtester

- 如果总容量是512MB则申请64MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 64m > /data/memtester_log.txt &
```

- 如果总容量是1GB则申请128MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 128m > /data/memtester_log.txt &
```

- 如果总容量是2GB则申请256MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 256m > /data/memtester_log.txt &
```

- 如果总容量是4GB则申请512MB进行memtester

```
1 | <rkxxxx:/ #> /data/memtester 512m > /data/memtester_log.txt &
```

4. 执行测试脚本

```
1 | <rkxxxx9:/ #> /data/ddr_freq_scan.sh
```

5. 确认拷机结果，拷机时间12小时以上

- 确认捕鱼达人是否正常运行，机器是否正常

- 确认memtester是否正常运行 在串口输入 `<rkxxx:/ #> ps | grep memtester`，看memtester进程是否存在。Eg:

```
1 <rkxxx:/ #> ps | grep memtester
2 root      14309 1730  74332  68156          0 5e980bf564 R /data/memtester
```

- 确认变频脚本运行是否正常运行，变频log是否在正常打印 log eg:

```
1 DDR freq will change to 600000000 8786
2 already change to 600000000 done
3 DDR freq will change to 800000000 8787
4 already change to 800000000 done
5 DDR freq will change to 200000000 8788
6 already change to 200000000 done
```

## Linux 4.4 reboot拷机

为防止做reboot过程机器进入休眠，影响测试，请通过设置 `setting->security->set screen lock->None`，让机器一开机跳过锁屏界面，直接进入主界面。同时通过设置 `Setting->Display->Sleep->Never sleep`让机器保持唤醒状态。

开启计算器 Calculator,输入 "839910906=",点击"RebootTest"，拷机时间12小时以上。

## Linux 4.4 sleep拷机（optional）

拔掉连接ADB的USB线，开启计算器 Calculator,输入 "839910906=",点击"SleepTest"，拷机时间12小时以上。

# RV1108 DDR颗粒验证流程说明

## RV1108 测试固件编译

打开DDR Test和pm\_tests配置。配置kernel代码的menuconfig，进入System Type，选择打开DDR Test和pm\_tests。

```
1 menuconfig
2 System Type --->
3 [*] /sys/pm_tests/ support
4 [*] DDR Test
```

如果menuconfig中没有 `[*] /sys/pm_tests/ support` 选项，请参考《DDR开发指南》的"DDR如何定频"和"如何enable/disable kernel中的DDR变频功能"章节，分别编译定频固件和变频固件。

## RV1108 测试环境搭建

将"rv1108\_dds\_test\_files"目录下的stressapptest和memtester\_32bit 考到SD卡根目录，把卡插到测试板上。

## RV1108 确认容量是否正确

通过 `<rv1108:/ #> cat /proc/meminfo` 查看MemTotal项所示容量是否与测试机器DDR总容量相符。

log eg:

```
1 | <rv1108:/ #> cat /proc/meminfo
2 | MemTotal:      133376 kB
```

64MB 约等于66688kB

128MB 约等于133376kB

256MB 约等于266752kB

512MB 约等于533504kB

由于系统内存分配管理差异的原因，得到的容量有些偏差，属于正常。

## RV1108 定频测试

### 1. DDR定频到拷机频率

请根据测试机器要跑的DDR最高频率，进行设置。

Eg:

如果测试机器要跑的DDR最高频率为800MHz。

```
1 | <rv1108:/ #> echo set clk_dds 800000000 > /sys/pm_tests/clk_rate
```

### 2. google stressapptest拷机，拷机时间12小时以上 如果总容量是128MB则申请16MB进行stressapptest，一般是总容量的八分之一

```
1 | <rv1108:/ #> /mnt/sdcard/stressapptest -s 500 -i 1 -C 1 -W --stop_on_errors -M 16
```

### 3. 确认拷机结果

拷机结束，确认机器是否正常，stressapptest结果是PASS还是FAIL。stressapptest 每隔10秒会打印一条log，log显示测试剩余时间。测试完成后会打印测试结果，如果测试通过打印Status: PASS，如果测试失败打印Status: FAIL。

### 4. memtester拷机，拷机时间12小时以上

如果总容量是128MB则申请16MB进行memtester，一般是总容量的八分之一

```
1 | <rv1108:/ #> /mnt/sdcard/memtester_32bit 16m
```

### 5. 确认拷机结果

拷机结束，确认机器是否正常，memtester是否正常运行。"DDR测试资源文件"目录里的memtester程序进行过修改，测试过程如果有发现错误会自动停止测试，如果持续测试12H+后，memtester仍在继续运行，说明测试过程没有发现错误。

- o memtester运行过程如果没有发现错误，会持续打印如下log:

```
1 | Loop 10:
```

```

2 | Stuck Address      : ok
3 | Random Value       : ok
4 | Compare XOR        : ok
5 | Compare SUB        : ok
6 | Compare MUL        : ok
7 | Compare DIV        : ok
8 | Compare OR         : ok
9 | Compare AND        : ok
10 | Sequential Increment: ok
11 | Solid Bits         : ok
12 | Block Sequential   : ok
13 | Checkerboard       : ok
14 | Bit Spread         : ok
15 | Bit Flip           : ok
16 | walking Ones       : ok
17 | walking Zeroes     : ok

```

- o memtester运行过程如果有发现错误，会自动停止测试并退出，退出时打印如下log:

```

1 | FAILURE: 0xffffffff != 0xffffbfff at offset 0x03b7d9e4.
2 | EXIT_FAIL_OTHERTEST

```

## RV1108 变频测试

如果机器前面做过定频测试，要重启机器，否则后续的变频命令会无法进行。

### 1. 后台执行memtester

如果总容量是128MB则申请16MB进行memtester，一般是总容量的十六分之一

```
1 | <rv1108:/ #> /mnt/sdcard/memtester_32bit 16m > /data/memtester_log.txt &
```

2. 执行变频命令 变频测试频率范围为400MHz到测试机器要跑的DDR最高频率之间进行。Eg: 如果测试机器DDR要跑的最高频率为800MHz。

```
1 | <rv1108:/ #> echo 'a:400M-800M-1000000T' > proc/driver/ddr_ts
```

### 3. 确认拷机结果，拷机时间12小时以上

- 确认机器是否正常
- 确认变频程序运行是否正常，变频log是否在正常打印
- 确认memtester是否正常运行 在串口输入<rkxxx:/ #> ps | grep memtester，看memtester进程是否存在。Eg:

```

1 | <rkxxx:/ #> ps | grep memtester
2 | root      14309 1730  74332  68156          0 5e980bf564 R /data/memtester

```

## RV1108 reboot拷机

可用1108自带reboot功能，menu -> debug -> reboot test。

# RK3308 DDR颗粒验证流程说明

## RK3308 确定容量是否正确

通过 `<rk3308:/ #> cat /proc/meminfo` 查看MemTotal项所示容量是否与测试机器DDR总容量相符。

log eg:

```
1 <rk3308:/ #> cat /proc/meminfo
2 MemTotal:      246832 kB
3 MemFree:       201800 kB
4
```

64MB 约等于66688kB

128MB 约等于133376kB

256MB 约等于266752kB

512MB 约等于533504kB

由于系统内存分配管理差异的原因，得到的容量有些偏差，属于正常。

## RK3308 拷机测试

由于3308当前不支持变频功能，ddr频率由loader初始化期间设置的频率，到后面是不会修改ddr频率的。DDR颗粒测试请使用最高频率的设置，即800MHz的loader。

### 1. memtester拷机，拷机时间12小时以上

首先要确认测试文件是否存在：

```
1 <rk3308:/ #> ls usr/bin/memtester
2 usr/bin/memtester
```

- 如果有测试文件，memtester测试命令如下：（如果总容量是128MB则申请16MB进行memtester，一般是总容量的八分之一。）

```
1 <rk3308:/ #> memtester 16m > /data/memtester_log.txt &
```

- 如果没有测试文件，则请将随此文件一同发布的memtester\_32bit.32bit（或memtester\_64bit.64bit）文件通过adb push到data分区：（memtester\_32bit.32bit适用于32位系统，memtester\_64bit.64bit适用于64位系统）

32位系统命令：

```
1 adb push \*文件路径*\memtester_32bit.32bit data/memtester
```

64位系统命令：



```
1 | adb push \*文件路径*\memtester_64bit.64bit data/memtester
```

将memtester\_32bit.32bit 文件修改权限为可执行：

```
1 | <rk3308:/ #> chmod 777 /data/memtester
```

memtester测试命令如下：（如果总容量是128MB则申请16MB进行memtester，一般是总容量的八分之一。）

```
1 | <rk3308:/ #> /data/memtester 16m > /data/memtester_log.txt &
```

## 2.确认拷机结果

- 拷机结束，确认机器是否正常。
- 确认memtester 打印是否正常：（注意，memtester出错不会停止测试，需要查看所有打印是否正确）

正确打印

```
1 | Loop 1:
2 |   Stuck Address      : ok
3 |   Random Value       : ok
4 |   Compare XOR        : ok
5 |   Compare SUB        : ok
6 |   Compare MUL        : ok
7 |   Compare DIV        : ok
8 |   Compare OR         : ok
9 |   Compare AND        : ok
10 | Sequential Increment: ok
11 | Solid Bits          : ok
12 | Block Sequential    : ok
13 | Checkerboard        : ok
14 | Bit Spread          : ok
15 | Bit Flip            : ok
16 | walking Ones        : ok
17 | walking Zeroes      : ok
18 | 8-bit writes        : ok
19 | 16-bit writes       : ok
20 |
```

出错打印

```
1 | Loop 92:
2 |   Stuck Address      : ok
3 |   Random Value       : FAILURE: 0x37fe0f4190f6b999 != 0x37fe0f4196f6b999 at offset
   0x0027a958.
4 | FAILURE: 0x2823d0d6f62a4b01 != 0x2823d0d6f02a4b01 at offset 0x0027a958.
5 |   Compare XOR        : FAILURE: 0x4c4f418e764340e8 != 0x4c4f418e704340e8 at offset
   0x0027a958.
6 |   Compare SUB        : FAILURE: 0x2856fb8ee22bd230 != 0xfe5ee503ee2bd230 at offset
   0x0027a958.
7 |   Compare MUL        : FAILURE: 0x00000000 != 0x00000001 at offset 0x0027a958.
```

```

8   Compare DIV      : FAILURE: 0xefb2eceaffbf9604 != 0xefb2eceaffbf9605 at offset
   0x0027a958.
9   Compare OR       : FAILURE: 0xcbb2a0e8cfbb8200 != 0xcbb2a0e8cfbb8201 at offset
   0x0027a958.
10  Compare AND      : Sequential Increment: ok
11  Solid Bits       : ok
12  Block Sequential : ok
13  Checkerboard     : ok
14  Bit Spread       : ok
15  Bit Flip         : ok
16  walking Ones     : ok
17  walking Zeroes   : ok
18  8-bit writes     : ok
19  16-bit writes    : ok
20

```

### 3.google stressapptest拷机，拷机时间12小时以上

首先要确认测试文件是否存在：

```

1 | <rk3308:/ #> ls usr/bin/stressapptest
2 | usr/bin/stressapptest

```

- 如果有测试文件，stressapptest测试命令如下：（如果总容量是128MB则申请16MB进行stressapptest，一般是总容量的八分之一。时间设置由-s后面的参数控制，单位是秒。如下为拷机24小时的命令）

```

1 | <rk3308:/ #> stressapptest -s 86400 -i 4 -C 4 -W --stop_on_errors -M 64

```

- 如果没有测试文件，则请将随此文件一同发布的stressapptest\_32bit（或stressapptest\_64bit）文件通过adb push到data分区：（stressapptest\_32bit适用于32位系统，stressapptest\_64bit适用于64位系统）

32位系统命令：

```

1 | adb push \*文件路径*\stressapptest_32bit data/stressapptest

```

64位系统命令：

```

1 | adb push \*文件路径*\stressapptest_64bit data/stressapptest

```

将stressapptest\_32bit 文件修改权限为可执行：

```

1 | <rk3308:/ #> chmod 777 /data/stressapptest

```

stressapptest测试命令如下：（如果总容量是128MB则申请16MB进行stressapptest，一般是总容量的八分之一。时间设置由-s后面的参数控制，单位是秒。如下为拷机24小时的命令）

```

1 | <rk3308:/ #> /data/stressapptest -s 86400 -i 4 -C 4 -W --stop_on_errors -M 64

```

#### 4.确认拷机结果

- 拷机结束，确认机器是否正常。
- stressapptest结果是PASS还是FAIL。stressapptest 每隔10秒会打印一条log，log显示测试剩余时间。测试完成后会打印测试结果，如果测试通过打印Status: PASS，如果测试失败打印Status: FAIL。

## RK3308 休眠唤醒测试

休眠唤醒需要kernel使能自动唤醒功能。打开rk3308.dtsi文件，找到休眠唤醒节点rockchip\_suspend,位或上RKPM\_TIMEOUT\_WAKEUP\_EN，使能自动唤醒：

```
1 rockchip_suspend: rockchip-suspend {
2     ...
3     rockchip,wakeup-config = <
4         (0
5         | RKPM_GPIO0_WAKEUP_EN
6         | RKPM_TIMEOUT_WAKEUP_EN
7         )
8     >;
9 };
10
```

编译好固件后，建议使用3308测试脚本的休眠唤醒测试。首先要确认测试文件是否存在：

```
1 <rk3308:/ #> ls rockchip_test/rockchip_test.sh
2 rockchip_test/rockchip_test.sh
```

- 若有测试文件，则休眠自动唤醒测试命令如下：

```
1 <rk3308:/ #> /rockchip_test/rockchip_test.sh
2 ...
3 please input your test moudle: //串口先输入8<enter>，再输入1<enter>
4 8
5 1
6
```

- 若没有测试文件，可以在串口命令行直接输入命令进行休眠唤醒测试，命令如下：

```
1 <rk3308:/ #> while true; do echo mem > /sys/power/state; sleep 5; done
```

拷机时间12h+，确认机器是否正常。

## RK3308 reboot拷机

建议使用3308测试脚本的reboot测试命令。首先要确认测试文件是否存在：

```
1 <rk3308:/ #> ls rockchip_test/rockchip_test.sh
2 rockchip_test/rockchip_test.sh
```

- 若有测试文件，reboot测试命令如下：

```
1 | <rk3308:/ #> /rockchip_test/rockchip_test.sh
2 | ...
3 | please input your test moudle: //串口输入13<enter>
4 | 13
```

- 若无测试文件，则请将随此文件一同发布的auto\_reboot\_test.sh 文件通过adb push到data分区：

```
1 | adb push \*文件路径*\auto_reboot_test.sh data/.
```

将auto\_reboot\_test.sh 文件修改权限为可执行：

```
1 | <rk3308:/ #> chmod 777 /data/auto_reboot_test.sh
```

reboot 测试命令如下：

```
1 | <rk3308:/ #> /data/auto_reboot_test.sh
```

拷机时间12h+，确认机器是否正常。