

准备工作:

确保拨码开关 ompin[5]打到 GND.

SD/inand 启动步骤:

第一层启动从 SD0(inand)启动, 当 iNand 启动做校验和时失败才会转为启动 SD2。而我们做裸机实验时是通过 SD2 来提供裸机程序镜像的, 因此需要先破坏内部 iNand 的 uboot 才可以强迫开发板从 SD2 启动去执行我们的裸机程序。

1.破坏开发板 iNand 中的 uboot:

uboot 环境下擦除开发板 iNand 中的 uboot



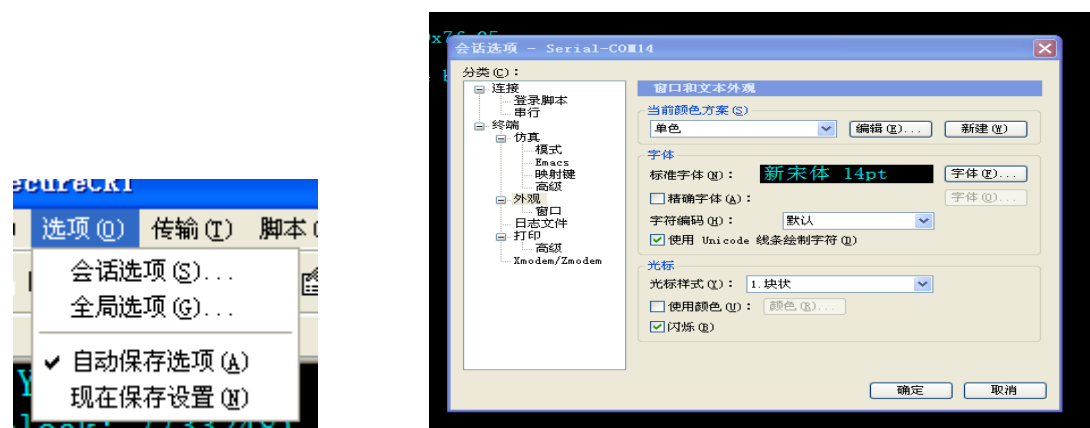
连接串口 uart2, 打开 SecureCRT

开机先回车, 再执行 movi 命令, 确认一下.

```
x210 # movi
Usage:
movi    - sd/mmc r/w sub system for SMDK board

x210 # help movi
movi init - Initialize moviNAND and show card info
movi read {u-boot | kernel} {addr} - Read data from sd/mmc
movi write {fwbl1 | u-boot | kernel} {addr} - Write data to sd/mmc
movi read rootfs {addr} [bytes(hex)] - Read rootfs data from sd/mmc by size
movi write rootfs {addr} [bytes(hex)] - Write rootfs data to sd/mmc by size
movi read {sector#} {bytes(hex)} {addr} - instead of this, you can use "mmc read"
movi write {sector#} {bytes(hex)} {addr} - instead of this, you can use "mmc write"
```

SecureCRT 调整字体背景在”会话选项”中进行



movi - sd/mmc r/w sub system for SMDK board

在 uboot 底下擦除 uboot:

执行(输入):movi write u-boot 0x30000000

//将内存 0x30000000 里面的内容(应该全部是 0)写入到 u-boot 分区(inand)里面

执行命令完毕后,画面显示

```
x210 # movi write u-boot 0x30000000
Writing BL1 to sector 1 (16 sectors).. checksum : 0x7fc05
writing bootloader.. 49, 1024
MMC write: dev # 0, block # 49, count 1024 ... 1024 blocks written: OK
completed
```

重启 X210,发现串口中输出,表明头部校验和失败,已经转入 SD 卡通道启动了.

SD checksum Error

再插入烧好的 SD 卡,即可启动.

linux 和 android 系统下,擦除 uboot:

待 X210 完全启动后,在 SecureCRT 中输入

命令:busybox dd if=/dev/zero of=/dev/block/mmcblk0 bs=512 seek=1 count=1

conv=sync

```
/ # busybox dd if=/dev/zero of=/dev/block/mmcblk0 bs=512 seek=1 count=1 conv=syn
c
1+0 records in
1+0 records out
```

命令:sync

```
/ # busybox dd if=/dev/zero of=/dev/block/mmcblk0 bs=512 seek=1 count=1 conv=syn
c
1+0 records in
1+0 records out
/ # sync
/ #
```

共 2 条命令.(破坏 inand 第一个扇区)

解释:这句话的意思就是说把板载的 iNand 的第一个扇区用全 0 来填充,其实就是擦除它,这样我们板载的 iNand 的 bootloader 的开始第 1 个扇区就被破坏了。将来启动时 iROM 还是会先从 iNand 中读取前 16KB,然后计算校验和。这时候因为有 1 个扇区被擦掉了,所以校验和不通过,所以启动失败(会从 SD2 去执行 2nd 启动)。

注意:破坏板载 iNand 的 bootloader 后,不插外部 SD 卡,启动时串口得到:SD checksum Error

2.烧录

SD 卡 PC 环境下烧录镜像:

PC 环境下,用专用工具:browse 找到镜像后,点击 start.将镜像复制到 SD 卡后,将 SD 卡插入至 SD2 口即可.(此方法只能做辅助启动,拔掉 SD 卡后也不能启动)



fastboot 烧录:

fastboot:是 windows 下的一个软件,也是 uboot 下的一个命令.

注意:如果 uboot 已经被擦除,则需要将烧好的 SD 插入,否则 X210 启动不起来,无法进入控制台.

1. 使用 USB 线传输连接 OTG 接口,打开 SecureCRT 串口,开机后,迅速敲回车进入控制台,输入 fastboot 命令.

```
checking mode for fastboot ...
Hit any key to stop autoboot: 0
x210 # fastboot
[Partition table on MoviNAND]
ptn 0 name='bootloader' start=0x0 len=N/A (use hard-coded info. (cmd: movi))
ptn 1 name='kernel' start=N/A len=N/A (use hard-coded info. (cmd: movi))
ptn 2 name='ramdisk' start=N/A len=0x300000(~3072KB) (use hard-coded info. (cmd: movi))
ptn 3 name='config' start=0xB11E00 len=0x1024BC00(~264495KB)
ptn 4 name='system' start=0x10D5DA00 len=0x1024BC00(~264495KB)
ptn 5 name='cache' start=0x20FA9600 len=0x6751800(~105798KB)
ptn 6 name='userdata' start=0x276FAE00 len=0xC3CC2A00(~3207946KB)
Received 17 bytes: download:00060000
Starting download of 393216 bytes
```

2.进入 windows,运行 cmd,切换到 fastboot.exe 所在的文件夹里面(我的电脑文件路径是: E:\fastboot)

相关操作命令:

切换命令:`cd E:\fastboot` //切换至 E:\fastboot 文件夹下

进入命令:`dir` //显示当前文件夹里的文件相关信息

查看命令: `fastboot devices` //命令用来查看当前连接的设备。

```
E:\fastboot>fastboot devices
SMDKC110-01    fastboot
```

表明已经检测到设备,下面就开始烧录镜像文件了.共有 3 个文件,这 3 个文件均需自己拷贝至 e:\fastboot 文件夹里面.

烧 uboot: fastboot flash bootloader android4.0.4/uboot.bin
烧 linux kernel: fastboot flash kernel android4.0.4/zImage-android
烧 android rom: fastboot flash system android4.0.4/x210.img

	烧录的分区	烧录文件的相对完整路径
fastboot flash <u>bootloader</u> android4.0.4/uboot.bin		烧uboot
fastboot flash <u>kernel</u> android4.0.4/zImage-android		烧linux kernel
fastboot flash <u>system</u> android4.0.4/x210.img		烧android rom

一般:我们破坏的是 uboot,所以我们只需要烧录 uboot 即可:

命令:fastboot flash bootloader android4.0.4/uboot.bin

```
E:\fastboot>fastboot flash bootloader android4.0.4/uboot.bin
< waiting for device >
sending 'bootloader' (384 KB)... OKAY
writing 'bootloader'... OKAY
```

重启命令:fastboot reboot //命令用来重启系统

3.UBUNTU 环境下烧录(见 SD 卡部分)