

密级状态：绝密（ ） 秘密（ ） 内部资料（ ☒ ） 公开（ ）

Mst6A638 编译与烧录（外销）

文件状态： [] 草稿 [] 正在修改 [<input checked="" type="checkbox"/>] 正式发布	文件标识	Mst6A638 编译与烧录（外销）
	文件编号	
	主要内容	
	当前版本	V1.0.0
	作 者	钟 斌
	完成日期	2015-12-21
	审 核	
	审核日期	

历史版本

时间	版本	内容	作者	备注
2015-12-21	V1.0.0	初版	钟斌	

Top-Tech

目录

一、	下载服务器代码.....	4
二、	编译.....	4
1、	Mboot.....	4
2、	PM.....	4
3、	Supernova.....	5
4、	Kernel.....	5
5、	Android	6
6、	打包 USB 升级包.....	7
三、	烧录.....	7
1、	用 ISP 工具烧录 mboot.....	7
2、	用 U 盘烧录 MstarUpgrade.bin.....	10
1)	命令升级.....	10
2)	按键板升级.....	11

一、下载服务器代码

Step1: `mkdir msd6a638_asean && cd msd6a638_asean` //创建一个文件夹进入此文件夹, 这个文件名字可以自己随意命名。

Step2: `repo init -u ssh://mgr023@172.168.1.60/manifests -b msd6a638_asean`

(这个不能直接 copy, 要手动敲进去)

Step4: `repo sync -j8`, 这个需要比较久的时间, 耐心等待, 下载完成后如下图所示。

Step5: `repo forall -c "git config core.filemode false"`

Step6: 下载之后的代码是没有分支, 需要执行下面命令进行切分支。

`repo forall -c "git checkout -b L-638-release-asean"`

在切换分支的过程中, 一直会打印类似下面的信息, 这是正常的。

```
zhongbin@android-server:/media/workspace0/zhongbin/msd6a638_asean$ repo forall -c "git checkout -b L-638-release-asean"
Switched to a new branch 'L-638-release-asean'
Switched to a new branch 'L-638-release-asean'
Switched to a new branch 'L-638-release-asean'
Switched to a new branch 'L-638-release-asean'
Switched to a new branch 'L-638-release-asean'
Switched to a new branch 'L-638-release-asean'
```

step7: 用 `repo branches` 查看是否是所有的工程都切换到了 L-638-release-asean 分支, 如下图:

```
zhongbin@android-server:/media/workspace0/zhongbin/msd6a638_asean$
zhongbin@android-server:/media/workspace0/zhongbin/msd6a638_asean$ repo branches
* L-638-release-asean | in all projects
zhongbin@android-server:/media/workspace0/zhongbin/msd6a638_asean$
```

二、编译

在编译之前, 要将分支切换到 `toptech_development` 这个分支。

1、Mboot

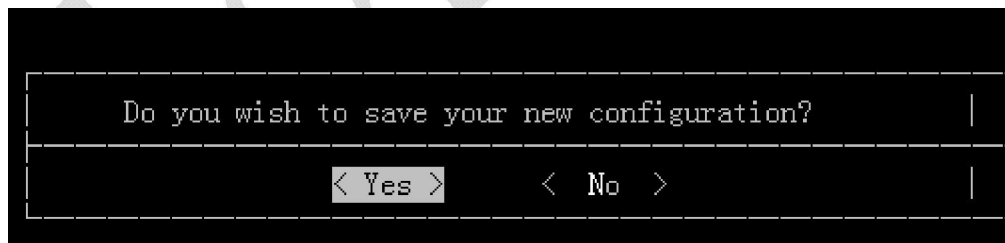
Step1: 进入 `vendor/mstar/mboot/sboot/` 目录

`cd vendor/mstar/mboot/sboot/`

Step2: 拷贝 config 文件, 如果配置没有变化的话, 不用每次编译都要执行此步。

`cp ./configs/monet/.config.monet.android.082b.rom_emmc.tee.ursa6.ddr1866.config`

Step3: 执行 `make menuconfig`, 不用做任何修改, 然后选 `exit` 项, 再在弹出的菜单中选 `Yes` 即可。



Step4: 编译。 `make clean && make` 这里不能加 `-j*` 进行多线程编译, 否则会报错, 代码中默认是 `-j2` 进行编译的。

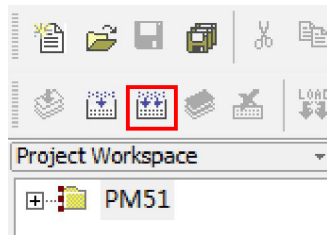
在编译完成后, 会在 `vendor/mstar/mboot/sboot/out/` 目录下生成 `rom_emmc_boot.bin` 和 `mboot.bin`, 在烧录及做生产软件的母片时都是用的到的。

2、PM

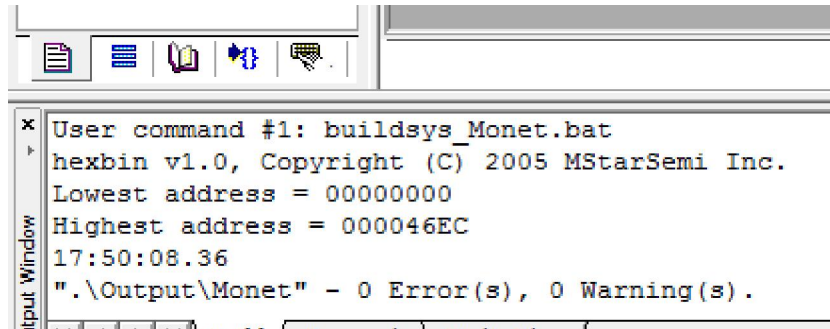
只有在修改遥控器时才要编译。PM 代码是使用 keil 进行编译的, 所以在编译前要安装好 keil 的编译软件。

Step1: 进入到 vendor/mstar/pm/Zenonia/Project/Mstar/Monet/目录下，运行 PM51_Monet.Uv2

Step2: 编译 PM，在 keil 软件点击如下图标进行全编译。



在 keil 中出现如下信息时，表示编译成功。



当然有时候可能会存在一些警告，看看警告是否是无关紧要的，并不是致命警告的话，可以不用管它。

编译完成后，会在 vendor/mstar/pm/Zenonia/Project/Mstar/Monet/ 目录下生成一个名叫 PM51_Monet.bin 的文件。

Step3: 把编译好的 PM51_Monet.bin 拷贝到 Supernova 相对应的目录下，即 \vendor\mstar\supernova\target\PM51_forBootFromROM\monet 目录，并且把 PM.bin 和 STR_PM.bin 替换掉。

3、Supernova

Step1: 修改 make_ext4fs 文件权限

进入到 supernova/target/tools/目录下，执行 chmod 777 make_ext4fs

Step2: 进入到 vendor/mstar/supernova/projects 目录

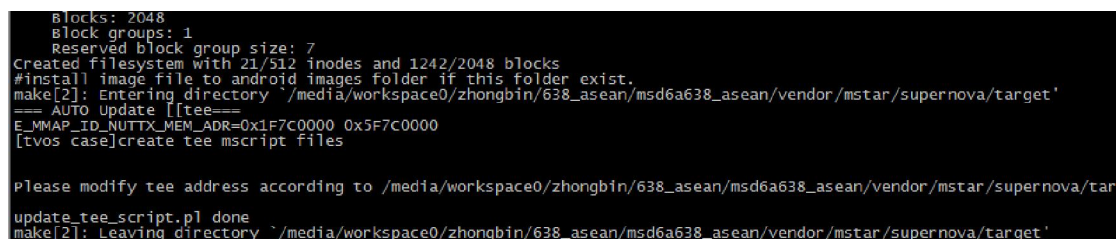
Step3: 执行脚本。

source

buildsettings/build_Monet_082B_ROM_EMMC_TVOS_ES_ASIA_URSA6_CMA_TEE.sh

Step4: make rebuild_all -j8 && make image_all

在编译完成后，会输出如下部分，表示编译成功：



4、Kernel

只有在要修改 IR 时才编译此部分。

Step1: 进入 vendor/mstar/kernel/3.10.40 目录

Step2: cp .config.monet.SMP.arm64.android.emmc.nandflash .config

Step3: ./genlink.sh

Step4: 执行 make menuconfig, 不用做任何修改, 然后选 exit 项即可。

Step5: make clean && make -j8

编译成功后, 会有如下部分信息:

```
LD [M] net/bluetooth/bluetooth.ko
LD [M] net/bluetooth/rfcomm/rfcomm.ko
LD [M] net/wireless/cfg80211.ko
Image arch/arm64/boot/Image.gz is ready
zhongbin@android-server:/media/workspace0/zhongbin/638_asean/msd6a638_asean/vendor/mstar/kernel/3.10.40$
```

Step6: 编译完成后, 再回到 msd6a638_asean 目录下, 执行如下命令, 把 kernel 拷贝到 Android 相应的目录: ./development/scripts/copy.sh 后会弹出选项, 输入数字 2 即可。

```
zhongbin@android-server:/media/workspace0/zhongbin/638_asean/msd6a638_asean/vendor$ cd ..
zhongbin@android-server:/media/workspace0/zhongbin/638_asean/msd6a638_asean$ ./development/scripts/copy.sh
/media/workspace0/zhongbin/638_asean/msd6a638_asean
Copy module? 1)Boot: 2)Kernel: 3)Supernova: 4)All :2
```

5、Android

Step1: 进入到 msd6a638_asean 目录, 也就是在下拉代码时创建的文件夹。

Step2: source build/envsetup.sh

Step3: lunch 后, 选 aosp_arbutus-userdebug, 输入对应的数字即可, 如下图中输入 21, 也可以直接输入命令 lunch aosp_arbutus-userdebug

```
zhongbin@android-server:/media/workspace0/zhongbin/638_asean/msd6a638_asean$ lunch
You're building on Linux
Lunch menu... pick a combo:
 1. aosp_arm-eng
 2. aosp_arm64-eng
 3. aosp_mips-eng
 4. aosp_mips64-eng
 5. aosp_x86-eng
 6. aosp_x86_64-eng
 7. mstaraddon-eng
 8. aosp_arbutus32-userdebug
 9. aosp_carambola-userdebug
10. aosp_arbutus-userdebug
11. m_e_arm-userdebug
12. mini_emulator_x86-userdebug
13. mini_emulator_x86_64-userdebug
14. mini_emulator_mips-userdebug
15. mini_emulator_arm64-userdebug
Which would you like? [aosp_arm-eng] 10
```

Step4: make -j8

编译完成之后, 会有如下信息:

```
Size: 1000140800
Block size: 4096
Blocks per group: 32768
Inodes per group: 7568
Inode size: 256
Journal blocks: 5200
Label:
Flexbg size: 16
Flexbg groups: 1
Blocks: 332800
Block groups: 11
Reserved block group size: 87
Created filesystem with 1772/83248 inodes and 167073/332800 blocks
Install system fs image: out/target/product/arbutus/system.img
out/target/product/arbutus/system.img+out/target/product/arbutus/obj/PACKAGING
=1365186153 reserve=14057472

#### make completed successfully (58:02 (mm:ss)) ####

zhongbin@android-server:/media/workspace0/zhongbin/638_asean/msd6a638_asean$
```

Step5: releaseimage.sh

6、打包 USB 升级包

在上面所有部分都编译完成之后，我们都是采用 USB 升级的方式，将编译好的程序更新到主板上。在 msd6a638_asean 目录下，运行 ./development/scripts/toptech_make_usb_upgrade.sh 脚本，然后选择对应的数字即可，如下：

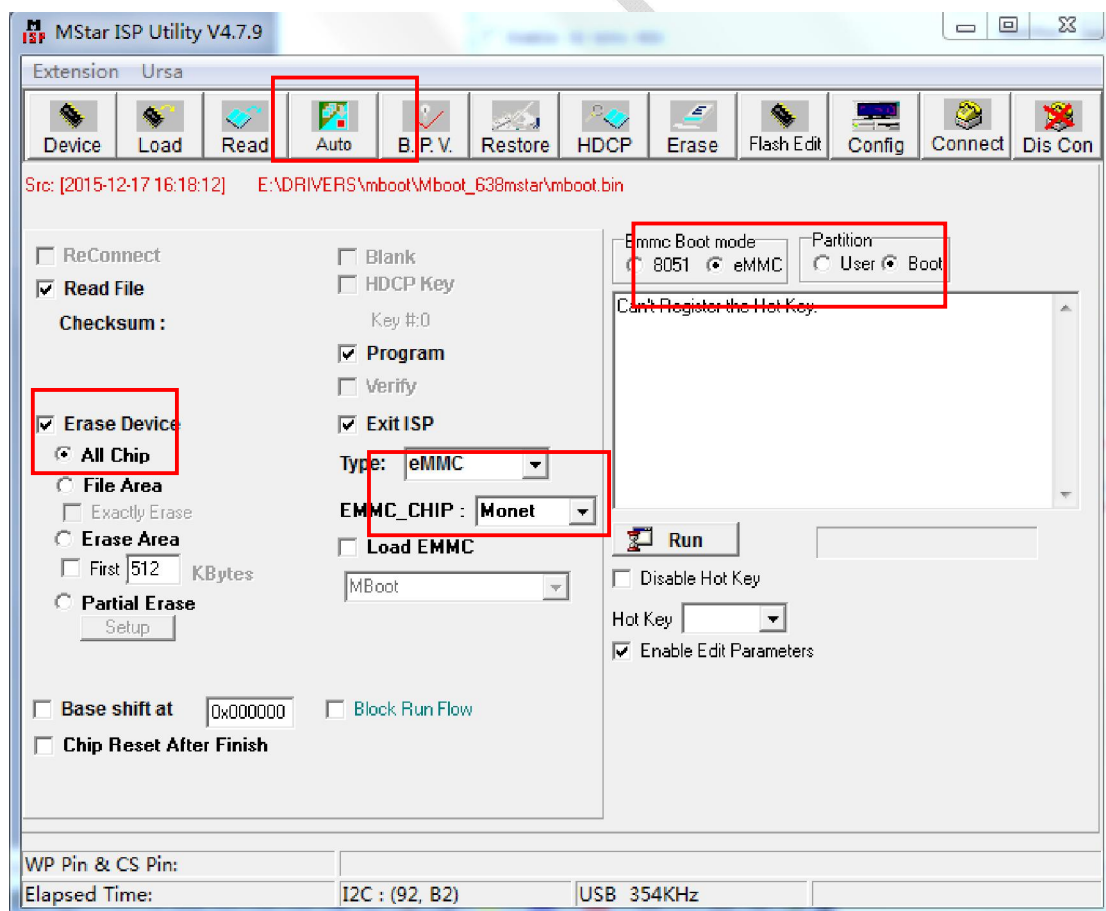
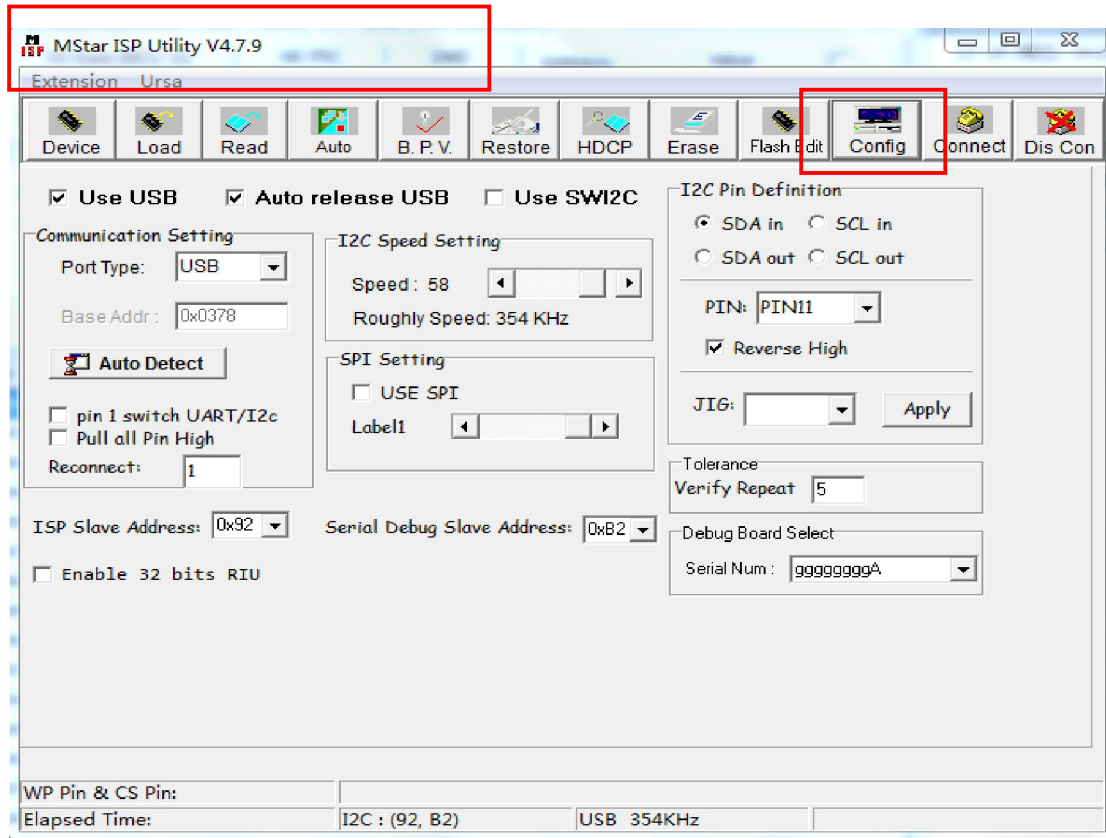
```
zhongbin@toptech-S5520HC:/media/workspace0/zhongbin/Mst638/L-638-release$ ./development/scripts/toptech_make_usb_upgrade.sh
/media/workspace0/zhongbin/Mst638/L-638-release
Update Module? 1)Mboot; 2)Kitkat&Kernel; 3)Supernova; 4)All :4
```

三、烧录

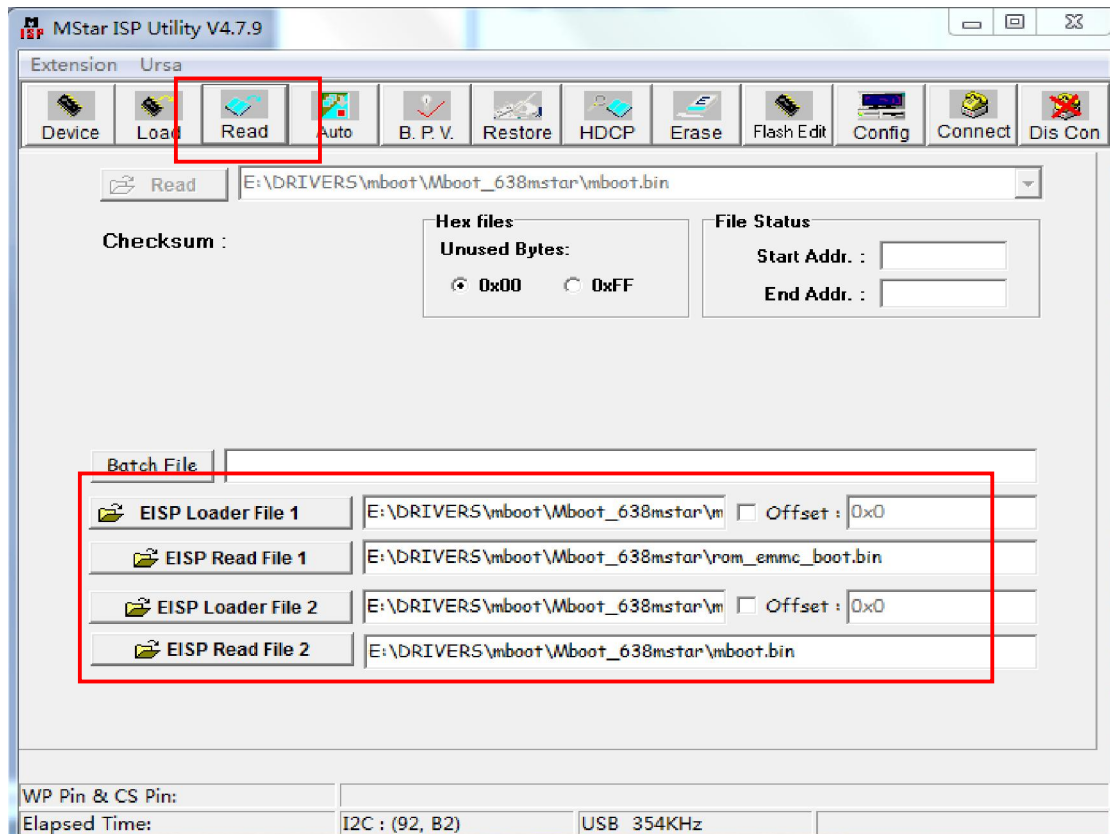
1、用 ISP 工具烧录 mboot

Step1: 运行 ISP 工具，638 方案必须用 V4.7.9 版本的才可以。

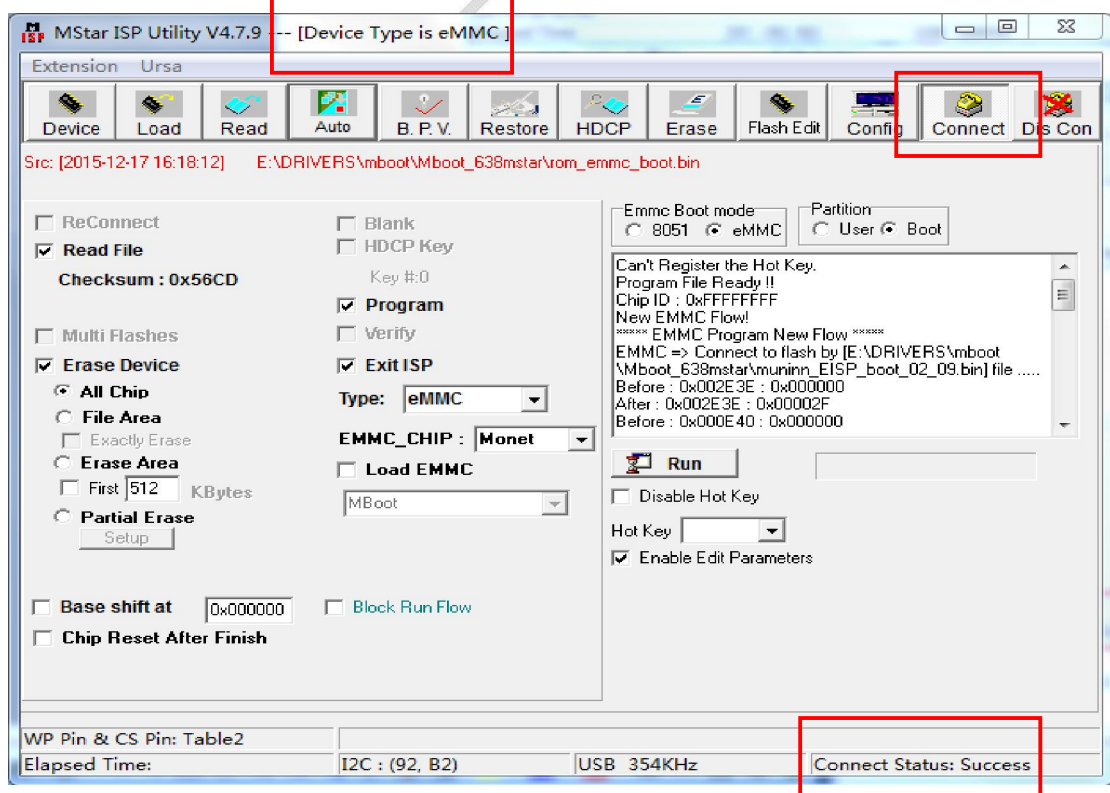
Step2: 设置参数



Step3: 选择文件，这里的文件路径最好不要中文并且不要有空格。

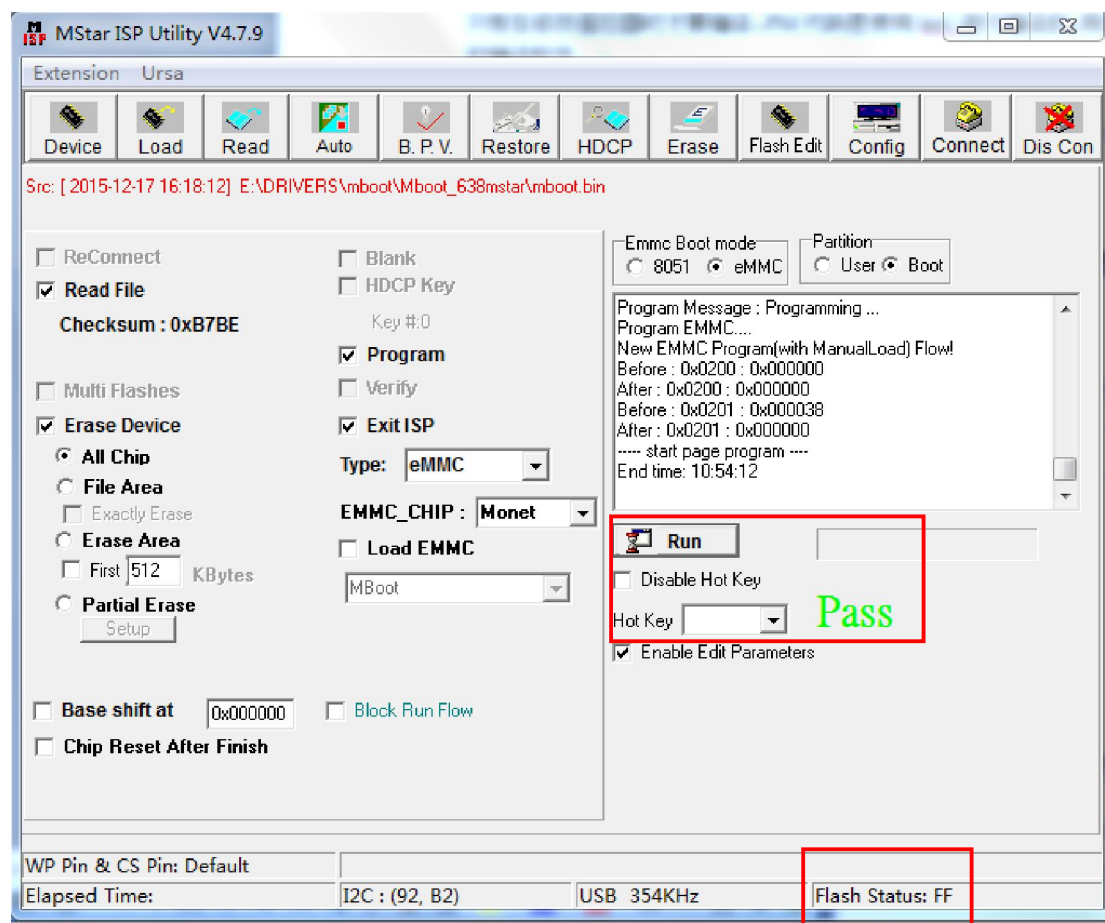


Step4: 点击 ISP 工具的 connect 连接主板



Step5: 点击 Run 进行烧录，因为烧录 Mboot 是要烧录两部分程序的，即 rom_emmc_boot.bin

和 mboot.bin，所以在烧录时，是会显示两次 pass 的，烧录 rom_emmc_boot.bin 时间很短，烧录 mboot.bin 要花 3 分多种，所以在烧录一次 mboot 时，要花 4 分钟左右的时间。烧录成功信息：



注意：连接 ISP 工具，板子必须处于待机状态。若板子已经烧录了 mboot，又没有烧录主程序时，板子上电是进入的 mboot 命令模式，在这种情况下，只有让板子在上电的瞬间进行 ISP 工具连接操作，当连接不上时，须多试几次。

2、用 U 盘烧录 MstarUpgrade.bin

1) 命令升级

Step1: 接上串口调试工具，连接上 SecureCRT，按上电脑上的 Enter 键，给板子上电后，串口中会出现如下打印，表示进入到 mboot 命令模式

```
Changelist:      24543697
===== set bootargs =====
Hit any key to stop autoboot: 0
<< MStar >>#
<< MStar >>#
<< MStar >>#
<< MStar >>#
<< MStar >>#
```

Step2: 把 MstarUpgrade.bin 放入 U 盘的根目录，插入板子上的 USB2.0 口，注意：U 盘格式必须为 FAT32 的。然后输入 custar 进行升级。

```
<< MStar >># custar
Check USB port[0]:
[USB] usb_lowlevel_init++
[USB] USB EHCI LIB VER: 2014.06.09
[USB] Port 0 is Enabled
[USB] TV_usb_init (UTMI Init) ++
[USB] UTMI Base 1F207500
[USB] UHC Base 1F204800
[USB] USBC Base 1F200E00
```

2) 按键板升级

把 MstarUpgrade.bin 放入 U 盘的根目录，插入板子上的 USB2.0 口，在板子处理待机状态下，长按按键板上的 POWER 键，直致指示红绿灯闪烁或出现升级界面为止，然后就可以松开按键了。