

说明

update_bin 应用程序，提供一种应用层刷机手段，默认只支持iboot 刷机，即可根据需要升级其他分区。

注意：原则上，不建议用户通过应用刷机，除非特殊情况。分区越大，断电导致分区被破坏的风险就越大。

- iboot : mtd0
- kernel:mtd1
- appfs: mtd2
- configs:mtd3

使用

- 修改mkimg_for_kiva_anker_ota.sh 脚本
 - 打开 build_iboot 编译，编译最新iboot
 - 修改 build_appfs.

```
631
632 #####
633 ###      build appfs      ###
634 #####
635 build_appfs ()
636 {
637     echo "start build appfs.."
638
639     # copy app
640     if [ "$COMPILE_TYPE" = "uclibc" ]; then
641         rm $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/uclibc/*
642         cp $FIRMWARE_DIR/ucamera $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/uclibc/
643         cp $FIRMWARE_DIR/hid_update $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/uclibc/
644         cp $FIRMWARE_DIR/update_bin $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/uclibc/
645         cp $FIRMWARE_BACKUP_DIR/iboot.bin $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/uclibc/
646     else
647         rm $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/glibc/*
648         cp $FIRMWARE_DIR/ucamera $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/glibc/
649         cp $FIRMWARE_DIR/hid_update $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/glibc/
650         cp $FIRMWARE_DIR/update_bin $FS_DIR/res/bin/t31/kiva/${SOLUTION_PREFIX}/glibc/
651     fi
652
653 }
```

- 新增 build_daemon

```
#####
###      build daemon      ###
#####
build_daemon ()
{
    echo "start build daemon..."

    cd $EXT_DIR/daemon
    make distclean
    if [ "$1" = "clean" ]; then return; fi
}
```

```

        if [ "$COMPILE_TYPE" = "uclibc" ]; then
            make CROSS_COMPILE=$CROSS_COMPILE_PREFIX
COMPILE_TYPE=uclibc
        else
            make CROSS_COMPILE=$CROSS_COMPILE_PREFIX
COMPILE_TYPE=glibc
        fi
        if [ $? != 0 ]; then
            echo "build daemon faild"
            exit 1
        fi

        cp update_bin $FIRMWARE_DIR/update_bin
        if [ $? != 0 ]; then
            echo "build daemon faild"
            exit 1
        fi

        cd -
        echo "build daemon ok...!"
    }

```

- 修改 kiva_k1_4m_Anker_uclibc_config_sc500ai

```

19 #SIZE=3328K
20 SIZE=8192K
21 IS_ROOT=no
22 STRIPPED=yes
23 COPYLIST:
24
25 # copy to appfs
26 bin/base/uclibc/lrz > bin/
27 bin/base/uclibc/lrz > bin/
28 bin/base/uclibc/adbd > bin/
29 bin/t31/kiva/k1-4m/uclibc/ucamera > bin/
30 bin/t31/kiva/k1-4m/uclibc/anticopy > bin/
31 bin/t31/kiva/k1-4m/uclibc/bid_update > bin/
32 bin/t31/kiva/k1-4m/uclibc/update_bin > bin/
33 bin/t31/kiva/k1-4m/uclibc/iboot.bin > bin/
34 #bin/t31/kiva/k1-4m/uclibc/motor_param_init > bin/
35
36 board/t31/kiva/k1-4m/app_init.sh > init/
37 board/t31/kiva/k1-4m/dameon.sh > init/
38 calibration/t31/kiva/k1-4m/sc500ai-t31.bin > etc/sensor/
39 #calibration/t31/kiva/k1-4m/sc500ai_2m-t31.bin > etc/sensor/
40

```

- 修改启动脚本: app_init_sc500ai.sh

```

1 #!/bin/sh
2 mount -t jffs2 /dev/mtdblock3 /media
3
4 MODULE_DIR=$(uname -r)
5 mkdir -p /tmp/modules/${MODULE_DIR}
6 mkdir -p /lib/modules
7 cd /lib/modules/
8 ln -s /tmp/modules/*
9
10 cd /system/bin
11 ./update_binlibboot
12
13 insmod /system/lib/modules/key_drv.ko
14 insmod /system/lib/modules/videobuf2-vmalloc.ko
15 insmod /system/lib/modules/libcomposite.ko
16 insmod /system/lib/modules/usbcamera.ko af_en=1 dmichannel=2 in_sample_rate=48000
17 insmod /system/lib/modules/gsensor-dw9714.ko
18
19 insmod /system/lib/modules/avpu.ko
20 insmod /system/lib/modules/audio.ko codec_type=1 dmichannel=2
21 insmod /system/lib/modules/tx-isp-t31.ko isp_clk=240000000
22 insmod /system/lib/modules/sensor_sc500ai_t31.ko
23 #insmod /system/lib/modules/sensor_sc500ai_2m_t31.ko
24 #killall -USR1 ucamera
25
26 #cd /system/bin/
27 #anticopy ucamera
28 ucamera &
29
30 sleep 5
31
32 hid_update &
33
34 sleep 3
35
36 adbd &
37
38 #motor_param_init
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99

```

INSERT master\$ app_init_sc500ai.sh[+]

修改完脚本后，正常编译ota固件，通过ankework升级即可，升级完成后，更新完iboot后，需要重启生效。