



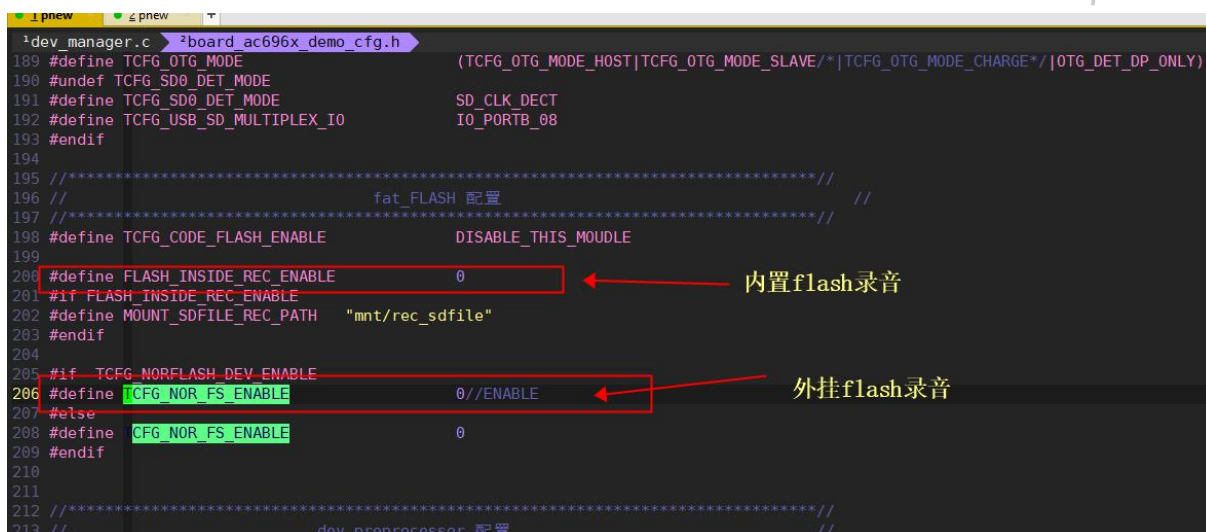
Flash 录音（内置、外置）以及分区操作说明文档

一、概述

支持外挂 flash 录音，内置 flash 录音，外挂 flash 分区（fat+tone+rec）功能。用户可以自己在相关工程自行开发使用。

二、工程使用描述

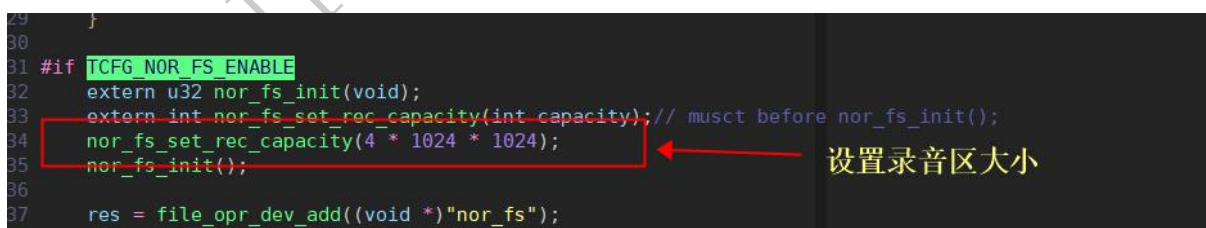
1、打开相应的宏配置



打开对应的宏即可跑相应的录音。

2、录音区的大小

2.1、外挂 flash 录音区

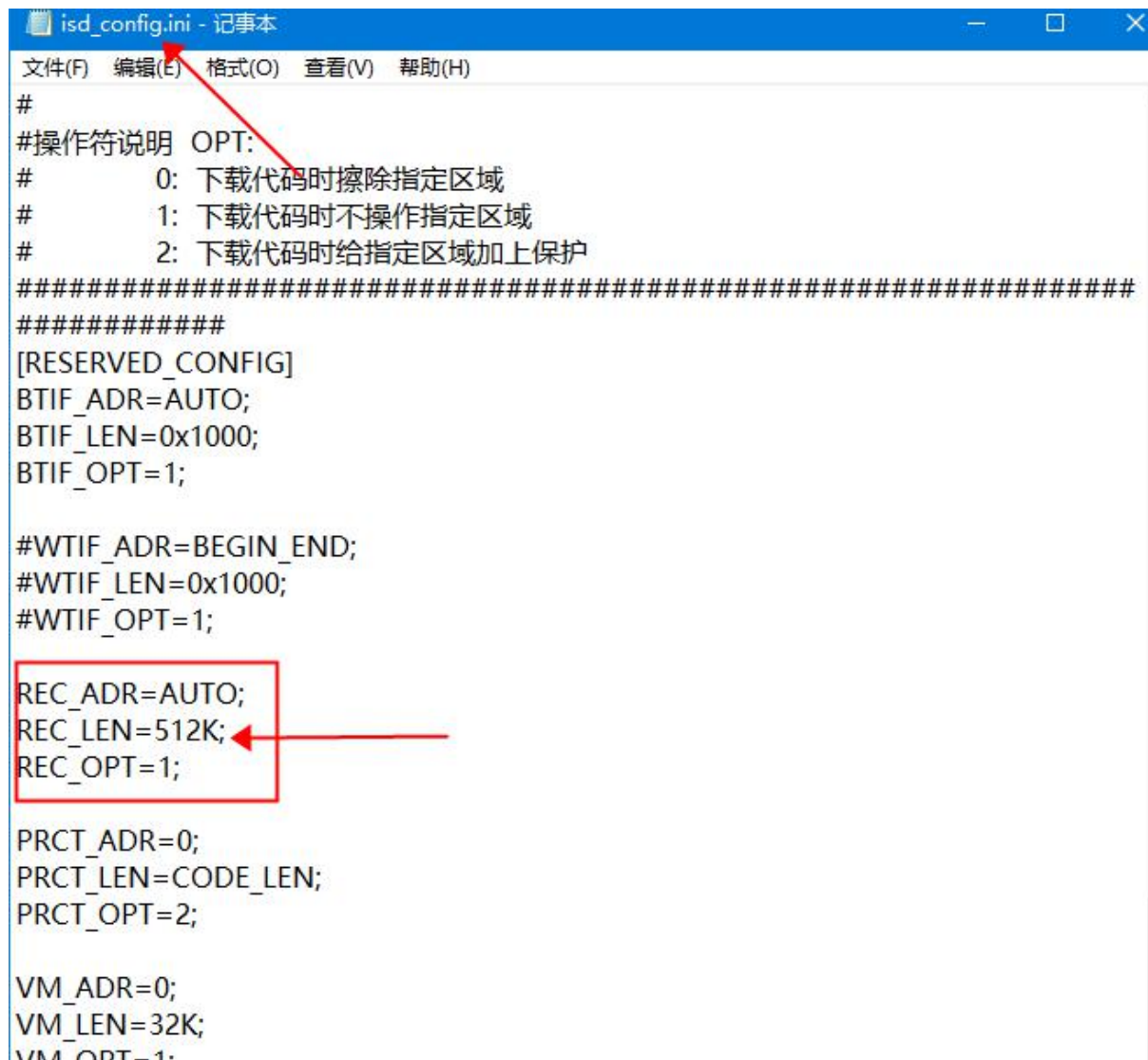


注意大小不能超过 flash 的大小。



2.2、内置 flash 录音区

起始地址由工程自动分配，大小设置如下：



```
#
#操作符说明 OPT:
#      0: 下载代码时擦除指定区域
#      1: 下载代码时不操作指定区域
#      2: 下载代码时给指定区域加上保护
#####
#####
[RESERVED_CONFIG]
BTIF_ADR=0;
BTIF_LEN=0x1000;
BTIF_OPT=1;

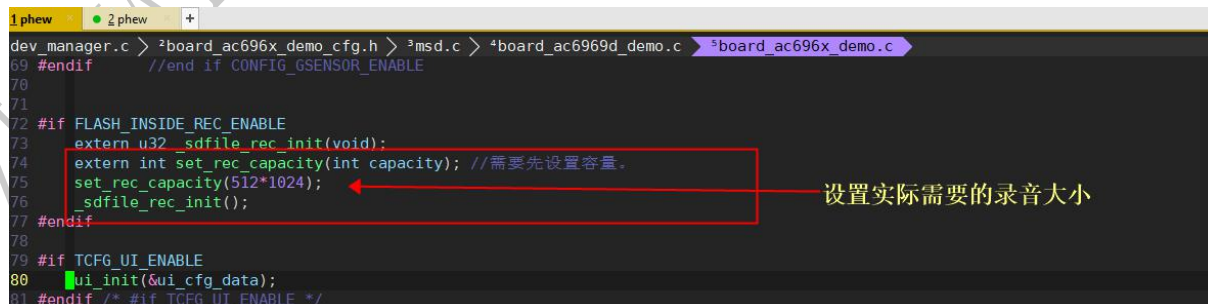
#WTIF_ADR=0;
#WTIF_LEN=0x1000;
#WTIF_OPT=1;

REC_ADR=0;
REC_LEN=512K;
REC_OPT=1;

PRCT_ADR=0;
PRCT_LEN=CODE_LEN;
PRCT_OPT=2;

VM_ADR=0;
VM_LEN=32K;
VM_OPT=1;
```

在 ini 配置中配置给录音的最大容量。



```
dev_manager.c > ^board_ac696x_demo_cfg.h > ^msd.c > ^board_ac6969d_demo.c > ^board_ac696x_demo.c
69 #endif //end if CONFIG_GSENSOR_ENABLE
70
71
72 #if FLASH_INSIDE_REC_ENABLE
73 extern u32 _sdfile_rec_init(void);
74 extern int set_rec_capacity(int capacity); //需要先设置容量。
75 set_rec_capacity(512*1024);
76 _sdfile_rec_init();
77 #endif
78
79 #if TCFG_UI_ENABLE
80 ui_init(&ui_cfg_data);
81 #endif /* #if TCFG_UI_ENABLE */
```

程序上设置实际需要的大小，如果大于最大容量，会报错。



3、分区处理

外挂 flash 如果需要挂载 FAT 文件系统或者提示音放外挂 flash，要做到不冲突需要有分区处理，可自由设置起始地址来控制，注：fat 文件系统需要放在 0 地址才可以读出。

FAT+TONE+REC 案例地址配置如下：

FAT 的起始地址为 0；

Tone 的起始地址由客户设置的 FAT 系统大小来设置进去,设置接口如下：

```
22 int ret = file_opr_dev_add((void *) "nor_flash");//this is demo
23 ret = file_opr_dev_add((void *) "nor_tone");//this is demo
24 if (!ret){
25     extern int set_tone_startaddr(int offset);
26     set_tone_startaddr(1*1024*1024);
27 }
28 #if TCFG_NOR_FS_ENABLE
```

REC 录音的起始地址，则在 FAT+TONE 的大小之后，设置如下：

```
139
140 #if TCFG_NOR_FS_ENABLE
141     extern int offset_addr;
142     extern int offset_rec;
143     offset_addr = (1024 * 1024) + offset_rec + 4096;
144     /* offset_addr = flash_capacity / 2; */
145     r_printf(">>>[test]:offset_addr = %d\n", offset_addr);
146 #endif
147
148 #if FLASH_CACHE_ENABLE
```

offset_addr 是rec起始地址
这个是设置容纳tone的大小
设置的起始地址

```
msd.c 2dev_manager.c > 3norflash.c >
702 }
703
704 int offset_rec = 0x40000; //set by customer
705 static void read_capacity(const struct usb_device_t *usb_device)
706 {
707     u32 err;
708     u32 capacity_temp;
709     u8 capacity[8] = {0};
710     u8 cur_lun = msd_handle->cbw.bCBWLUN;
711     if (RBC_SDFAIL == check_disk_status(msd_handle->info.dev_handle[cur_lun], cur_lun)) {
712         stall_error(usb_device, 0, 0x0f);
713         return;
714     }
715     dev_ioctl(msd_handle->info.dev_handle[cur_lun], IOCTL_GET_CAPACITY, (u32)&capacity_temp);
716     #if (TCFG_NOR_FS_ENABLE && TCFG_NORFLASH_DEV_ENABLE)
717     if (!strcmp(msd_handle->info.dev_name[cur_lun], "norflash")) {
718         /* r_printf(">>>[test]:capacity_temp1111 = %d\n", capacity_temp); */
719         extern int offset_addr;
720         capacity_temp = (1024*1024) / 512;
721         offset_addr = (1024 * 1024) + offset_rec + 4096;
722         /* capacity_temp = capacity_temp / 2; */
723         /* offset_addr = capacity_temp * 512; */
724     }
725     #endif
726     capacity_temp = cpu_to_be32(capacity_temp - 1);
```

设置不同的起始地址即可获取不同分区，使得互相不会影响。