

*AC630N_bt_data_transfer_
sdk_release_v0.9.1 补丁*

版权所有 2022 杰理科技有限公司未经许可，禁止转载

目录

目录.....	2
AC630N_bt_data_transfer_sdk_release_v0.9.1.....	3
1. 替代工具.....	3
2. 替代 AC635N 的 loader、ota、uboot 文件.....	3
3. 替换 AC635N 的 update.a 和 cpu.a 文件.....	4
4. 修改 AC635N 下载目录下的 isd_config.ini 文件.....	4
5. 修改 AC635N 下载目录下的 testbox_update.c 文件.....	5
6. 修改 AC635N 下载目录下的 update_loader_download.h 文件.....	6
7. 修改 AC635N 下载目录下的 power_interface.h 文件.....	6
8. 替代 AC636N 的 loader、ota、uboot 文件.....	7
9. 替换 AC636N 的 update.a 和 cpu.a 文件.....	7
10. 修改 AC636N 下载目录下的 isd_config.ini 文件.....	8
11. 修改 AC636N 下载目录下的 testbox_update.c 文件.....	8
12. 修改 AC636N 下载目录下的 update_loader_download.h 文件.....	9
13. 修改 AC636N 下载目录下的 power_interface.h 文件.....	9

AC630N_bt_data_transfer_sdk_release_v0.9.1

补丁描述:

日期: 2022-05-24

补丁说明: 该补丁分别更新了 AC635N 和 AC636N 的升级库、CPU 库、工具、OTA 文件, 解决了升级问题。

1. 替代工具

名称	修改日期	类型	大小
AC635N	2022-5-24 10:27	文件夹	
AC636N	2022-5-24 10:27	文件夹	
fw_add.exe	2022-5-21 14:17	应用程序	1,785 KB
isd_download.exe	2022-5-21 14:17	应用程序	2,098 KB
ufw_maker.exe	2022-5-21 14:17	应用程序	1,210 KB

将目录下 fw_add.exe、isd_download.exe 和 ufw_maker.exe 工具放到对于的下载目录下。
(对应的目录路径: sdk/cpu/br23/tools/)

2. 替代 AC635N 的 loader、ota、uboot 文件

名称	修改日期	类型	大小
文件修改	2022-5-24 9:35	文件夹	
cpu.a	2022-5-23 20:51	360压缩	1,035 KB
update.a	2022-5-21 14:18	360压缩	237 KB
br23loader.bin	2022-5-21 14:17	BIN 文件	24 KB
ota_all.bin	2022-5-21 14:17	BIN 文件	248 KB
ota_all_debug.bin	2022-5-21 14:17	BIN 文件	356 KB
ota_nor.bin	2022-5-21 14:17	BIN 文件	17 KB
ota_nor_debug.bin	2022-5-21 14:17	BIN 文件	27 KB
uboot.boot	2022-5-21 14:17	BOOT 文件	8 KB
uboot_no_ota.boot	2022-5-21 14:17	BOOT 文件	5 KB
uboot.boot_debug	2022-5-21 14:17	BOOT_DEBUG ...	13 KB
uboot_no_ota.boot_debug	2022-5-21 14:17	BOOT_DEBUG ...	9 KB
br23loader.uart	2022-5-21 14:17	UART 文件	11 KB

将目录下 loader、ota 和 uboot 文件放到对于的下载目录下。
(对应的目录路径: sdk/cpu/br23/tools/)

3. 替换 AC635N 的 update.a 和 cpu.a 文件

名称	修改日期	类型	大小
文件修改	2022-5-24 9:35	文件夹	
cpu.a	2022-5-23 20:51	360压缩	1,035 KB
update.a	2022-5-21 14:18	360压缩	237 KB
br23loader.bin	2022-5-21 14:17	BIN 文件	24 KB
ota_all.bin	2022-5-21 14:17	BIN 文件	248 KB
ota_all_debug.bin	2022-5-21 14:17	BIN 文件	356 KB
ota_nor.bin	2022-5-21 14:17	BIN 文件	17 KB
ota_nor_debug.bin	2022-5-21 14:17	BIN 文件	27 KB
uboot.boot	2022-5-21 14:17	BOOT 文件	8 KB
uboot_no_ota.boot	2022-5-21 14:17	BOOT 文件	5 KB
uboot.boot_debug	2022-5-21 14:17	BOOT_DEBUG ...	13 KB
uboot_no_ota.boot_debug	2022-5-21 14:17	BOOT_DEBUG ...	9 KB
br23loader.uart	2022-5-21 14:17	UART 文件	11 KB

将目录下 cpu.a 和 update.a 文件替换工程对应的库。
(对应的目录路径: sdk/include_lib/liba/br23/)。

4. 修改 AC635N 下载目录下的 isd_config.ini 文件

```

1 isd_config.ini
2
3 [EXTRA_CFG_PARAM]
4 NEW_FLASH_FS=YES; // 单备份
5 FORCE_VM_ALIGN=YES; // 强制VM对齐
6 CHIP_NAME=AC635N; // 芯片型号
7 ENTRY=0x1E00120; // 程序入口地址
8 PID=AC635N_TRANS; // 长度16byte, 示例: 芯片封装_应用方向_方案名称
9 VID=0.01;
10
11 RESERVED_OPT=0;
12 SPECIAL_OPT=0;
13 FORCE_4K_ALIGN=YES;
14
15 #DOWNLOAD_MODEL=SERIAL; //usb
16 DOWNLOAD_MODEL=usb; //
17 SERIAL_DEVICE_NAME=JlVirtualJtagSerial;
18 SERIAL_BAUD_RATE=1000000;
19 SERIAL_CMD_OPT=0;
20 SERIAL_CMD_RATE=100; [n*10000]
21 SERIAL_CMD_RES=0;
22 SERIAL_INIT_BAUD_RATE=9600;
23 LOADER_BAUD_RATE=1000000;
24 LOADER_ASK_BAUD_RATE=1000000;
25 #SERIAL_SEND_KEY=YES;
26 BEFORE_LOADER_WAIT_TIME=100;
27 FLASH_BIN_CNT=6;
28 NEED_RESERVED_4K=YES; // 关闭4K保留
29
30 [CHIP_VERSION]
31 SUPPORTED_LIST=A,B,D;
32
33 #UBOOT配置项, 请勿随意调整顺序 #
34 [SYS_CFG_PARAM]
35 #data_width[0 1 2 3 4] 3的时候uboot自动识别2或者4线
36 #clk [0-255]
37 #mode:
38 # 0 RD_OUTPUT, 1 cmd 1 addr
39 # 1 RD_I/O, 1 cmd x addr
40 # 2 RD_I/O_CONTINUE] no_send_cmd x add
41
42 #port:
43 # 0 优先选A端口 CS:PD3 CLK:PD0 D0:PD1 D1:PD2 D2:PB7 D3:PB
44 # 1 优先选B端口 CS:PA13 CLK:PD0 D0:PD1 D1:PA14 D2:PA15 D3:PB
45
46 #SPI=2_3_0_0; #width_clk_mode_port;
47 #OSC=btosc;
48 #OSC_FREQ=12Mhz; #[24Mhz 12Mhz]
49 #SYS_CLK=24Mhz; #[48Mhz 24Mhz]
50 #UTTX=PA05; //uboot串口tx
51 #UTBD=1000000; //uboot串口波特率
52 #UTRX=PB01; 串口升级[PB00 PB05 PA05]
53 #RESET=PB01_00_0; //port口_长按时间_有效电平 (长按时间有00、01、0
54 位为秒, 当长按时间为00时, 则关闭长按复位功能。)
55 #UPDATE_JUMP=0; // 是否支持升级过程中维持I/O电平, 用于一些使用I/O
56
57 #0:disable
58 #1:PA9 PA10
59 #2:USB
60 #3:PB1 PB2
61 #4:PB6 PB7
62
63 #sdtap=2;
64 #psram=0;
65
66 #VLVD=4; //VDDIO_LVD档位, 0: 1.9V 1: 2.0V 2: 2.1V 3: 2.2V 4:
67 V 7: 2.6V
68
69 #EADDR=0; //ENTRY_ADDR;
70 #OFFSET=1; // n * 4k

```

```
1 isd_config.ini
04 VM_LEN=8K;
05 VM_OPT=1;
06
07 [BURNER_CONFIG]
08 SIZE=32;
09
10 [TOOL_CONFIG]
11 1T02_MIN_VER=2.26.5;
112 1T08_MIN_VER=3.1.10;
```

可参考文件修改目录下有对应的 isd_config.ini 文件。

5. 修改 AC635N 下载目录下的 testbox_update.c 文件

```
1 testbox_update.c
2 static void testbox_bt_classic_update_state_cbk(int type, u32 state, void *priv)
3 {
4     update_ret_code_t *ret_code = (update_ret_code_t *)priv;
5
6     switch (state) {
7     case UPDATE_CH_EXIT:
8         log_info("state: %x err: %x\n", ret_code->stu, ret_code->err_code);
9         if (UPDATE_DUAL_BANK_IS_SUPPORT()) {
10             if ((0 == ret_code->stu) && (0 == ret_code->err_code)) {
11                 log_info("bt update succ\n");
12                 update_result_set(UPDATE_SUCC);
13             } else {
14                 log_info("bt update fail\n");
15                 update_result_set(UPDATE_DEV_ERR);
16             }
17         } else {
18             if ((0 == ret_code->stu) && (0 == ret_code->err_code)) {
19                 //update_mode_api(BT_UPDATE);
20                 update_mode_api_v2(BT_UPDATE,
21                                     testbox_bt_classic_update_private_param_fill,
22                                     testbox_bt_classic_update_before_jump_handle);
23             } else if ((0 == ret_code->stu)) { //文件相同的情况
24                 cpu_reset();
25             }
26         }
27         break;
28     }
29 }
30
31 static u8 ble_update_ready_jump_flag = 0;
32 u8 ble_update_get_ready_jump_flag(void)
33 {
34     return ble_update_ready_jump_flag;
35 }
36
37 static void testbox_ble_update_private_param_fill(UPDATE_PARAM *p)
38 {
39     {
40         type = BT_UPDATE;
41         state_cbk = testbox_bt_classic_update_state_cbk;
42         p_op_api = &mp_ch_update_op;
43     }
44 }
45
46 #endif
47 } else if ((0 == ret_code->stu)) { //文件相同的情况, 默认是复位
48     #if TCFG_USER_BLE_ENABLE && (TCFG_BLE_DEMO_SELECT != DEF_BLE_DEMO_NULL) \
49     && (TCFG_BLE_DEMO_SELECT != DEF_BLE_DEMO_ADV) || defined(CONFIG_MESH_CASE_ENABLE) \
50     && (TCFG_BLE_DEMO_SELECT != DEF_BLE_DEMO_CLIENT)
51         extern void ble_module_enable(u8 en);
52         ble_module_enable(0);
53         u8 cnt = 0;
54         while (!check_le_conn_disconnect_flag()) {
55             log_info("wait disconn\n");
56             os_time_delay(2);
57             if (cnt++ > 5) {
58                 break;
59             }
60         }
61         cpu_reset();
62     }
63     break;
64 }
65 }
66
67 void testbox_update_msg_handle(int msg)
68 {
69     log_info("msg: %x\n", msg);
70
71     switch (msg) {
72     case MSG_BT_UPDATE_LOADER_DOWNLOAD_START:
73         if (UPDATE_MODULE_IS_SUPPORT(UPDATE_BT_LMP_EN)) {
74             #if (SMART_BOX_EN) && (CONFIG_WATCH_CASE_ENABLE)
75             app_smartbox_task_prepare(0, SMARTBOX_TASK_ACTION_WATCH_TRANSFER, 0);
76             #endif
77             update_mode_info_t info = {
78                 .type = BT_UPDATE,
79                 .state_cbk = testbox_bt_classic_update_state_cbk,
80                 .p_op_api = &mp_ch_update_op,
81             };
82             testbox_bt_classic_update_state_cbk(UPDATE_BT_LMP_EN, &info);
83         }
84     }
85 }
```

可参考文件修改目录下有对应的 testbox_update.c 文件。

6. 修改 AC635N 下载目录下的 update_loader_download.h 文件

```

1 update_loader_download.h
60
61 //update result code;
62 enum {
63     UPDATE_RESULT_ERR_NONE = 0,
64     UPDATE_RESULT_FILE_SIZE_ERR = 0x1, //文件大小错误
65     UPDATE_RESULT_LOADER_SIZE_ERR = 0x2, //loader大小错误
66     UPDATE_RESULT_LOADER_VERIFY_ERR, //update loader校验失败
67     UPDATE_RESULT_REMOTE_FILE_HEAD_ERR, //读升级文件头错误
68
69     UPDATE_RESULT_LOCAL_FILE_HEAD_ERR = 0x5, //读flash文件头错误
70     UPDATE_RESULT_NOT_FIND_TARGET_FILE_ERR, //找不到目标文件
71     UPDATE_RESULT_FILE_OPERATION_ERR, //文件操作失败
72     UPDATE_RESULT_FLASH_DATA_VERIFY_ERR, //flash数据校验失败
73
74     UPDATE_RESULT_UBOOT_NOT_MATCH = 0x9, //UBOOT不匹配
75     UPDATE_RESULT_PRODUCT_INFO_NOT_MATCH = 0xa, //芯片型号不匹配
76     UPDATE_RESULT_EX_DSP_UPDATE_ERR, //外部IC升级出错;
77     UPDATE_RESULT_CFG_UPDATE_ERR, //配置升级出错
78
79     UPDATE_RESULT_FLASH_ERASE_ERR = 0xd, //flash 擦失败(可能是写保护)
80     UPDATE_RESULT_REMOTE_FILE_NOT_MATCH, //升级文件不匹配
81     UPDATE_RESULT_ANC_CFG_UPDATE_ERR, //ANC配置升级出错
82     UPDATE_RESULT_ANC_COEF_UPDATE_ERR, //ANC配置升级出错
83     UPDATE_RESULT_OTA_TWS_NO_RSP, //对耳同步升级传输数据没有回复
84     UPDATE_RESULT_RESOURCE_LIMIT, //资源不足
85     UPDATE_RESULT_OTA_TWS_START_ERR, //对耳启动升级失败
86     UPDATE_RESULT_OTA_TWS_CRC_ERROR, //对耳校验失败
87     UPDATE_RESULT_OTA_APP_EXIT, //升级过程APP强制退出
88     UPDATE_RESULT_TWS_NO_CONNECT, //对耳未连接
89     UPDATE_RESULT_FILE_SAME, //相同文件升级
90 };
91
92 #include "system/task.h"
93 typedef struct _update_type_info_t {
94     int type;
95     u8 task_en;
96 } update_type_info_t;
97
98 NORMAL  release/AC630N_bt_data_transfer_sdk_release_v0.9.0  update_loader_download.h  cpp  utf-8[unix]  33%  60/179

```

可参考文件修改目录下有对应的 update_loader_download.h 文件。

7. 修改 AC635N 下载目录下的 power_interface.h 文件

```

1 power_interface.h
2
3 //<Max hardware wakeup port
4 #define MAX_WAKEUP_PORT 8
5 #define MAX_RTC_PORT 2
6
7 struct wakeup_param {
8     const PORT_FLT filter;
9     const struct port_wakeup *port[MAX_WAKEUP_PORT];
10    const struct port_wakeup *rtc_port[MAX_RTC_PORT];
11    const struct charge_wakeup *charge;
12    const struct alarm_wakeup *alarm;
13    const struct lvd_wakeup *lvd;
14    const struct sub_wakeup *sub;
15 };
16
17 struct reset_param {
18     u8 en;
19     u8 mode;
20     u8 level;
21     u8 iomap; //Port Group, Port Index
22 };
23
24 struct low_power_operation {
25     const char *name;
26     u32(*get_timeout)(void *priv);
27     void (*suspend_probe)(void *priv);
28     void (*suspend_post)(void *priv, u32 usec);
29     void (*resume)(void *priv, u32 usec);
30     void (*resume_post)(void *priv, u32 usec);
31     void (*off_probe)(void *priv);
32 };
33
34 NORMAL  release/AC630N_bt_data_transfer_sdk_release_v0.9.0  ~/repo_data_trans_0520_del/SDK/include_lib/driver/cpu/br23/asm/power_interface.h  cpp  utf-8[unix]  47%  144/306

```

可参考文件修改目录下有对应的 power_interface.h 文件。

8. 替代 AC636N 的 loader、ota、uboot 文件

名称	修改日期	类型	大小
文件修改	2022-5-23 21:03	文件夹	
cpu.a	2022-5-23 14:23	360压缩	1,003 KB
update.a	2022-5-21 15:43	360压缩	237 KB
<input checked="" type="checkbox"/> br25loader.bin	2022-5-21 14:32	BIN 文件	29 KB
<input checked="" type="checkbox"/> ota.bin	2022-5-21 14:32	BIN 文件	195 KB
<input checked="" type="checkbox"/> ota_debug.bin	2022-5-21 14:33	BIN 文件	268 KB
<input type="checkbox"/> uboot.boot	2022-5-21 14:32	BOOT 文件	4 KB
<input type="checkbox"/> uboot_no_ota.boot	2022-5-21 14:32	BOOT 文件	4 KB
<input type="checkbox"/> uboot.boot_debug	2022-5-21 14:32	BOOT_DEBUG ...	6 KB
<input type="checkbox"/> uboot_no_ota.boot_debug	2022-5-21 14:32	BOOT_DEBUG ...	5 KB
<input type="checkbox"/> br25loader.uart	2022-5-21 14:32	UART 文件	12 KB

将目录下 loader、ota 和 uboot 文件放到对于的下载目录下。

(对应的目录路径: sdk/cpu/br25/tools/)。

9. 替换 AC636N 的 update.a 和 cpu.a 文件

名称	修改日期	类型	大小
文件修改	2022-5-23 21:03	文件夹	
cpu.a	2022-5-23 14:23	360压缩	1,003 KB
update.a	2022-5-21 15:43	360压缩	237 KB
<input checked="" type="checkbox"/> br25loader.bin	2022-5-21 14:32	BIN 文件	29 KB
<input checked="" type="checkbox"/> ota.bin	2022-5-21 14:32	BIN 文件	195 KB
<input checked="" type="checkbox"/> ota_debug.bin	2022-5-21 14:33	BIN 文件	268 KB
<input type="checkbox"/> uboot.boot	2022-5-21 14:32	BOOT 文件	4 KB
<input type="checkbox"/> uboot_no_ota.boot	2022-5-21 14:32	BOOT 文件	4 KB
<input type="checkbox"/> uboot.boot_debug	2022-5-21 14:32	BOOT_DEBUG ...	6 KB
<input type="checkbox"/> uboot_no_ota.boot_debug	2022-5-21 14:32	BOOT_DEBUG ...	5 KB
<input type="checkbox"/> br25loader.uart	2022-5-21 14:32	UART 文件	12 KB

将目录下 cpu.a 和 update.a 文件替换工程对应的库。

(对应的目录路径: sdk/include_lib/liba/br25/)。

10. 修改 AC636N 下载目录下的 isd_config.ini 文件

```

isd_config.ini
##### UBOOT配置项, 请勿随意调整顺序
[SYS_CFG_PARAM]
#####
data_width[0 1 2 3 4] 3的时候uboot自动识别2或者4线
#clk [0-255]
#mode:
# 0 RD_OUTPUT, 1 cmd 1 addr
# 1 RD_I/O, 1 cmd x addr
# 2 RD_I/O_CONTINUE no_send_cmd x addr
#port:
# 0 优先选A端口 CS:PD3 CLK:PD0 D0:PD1 D1:PD2 D2:PB7 D3:PD5
# 1 优先选B端口 CS:PA13 CLK:PD0 D0:PD1 D1:PA14 D2:PA15 D3:PD5
SPI=2_3_0_0; #width_clk_mode_port;
#OSC=btosc;
#OSC_FREQ=12MHz; #12MHz 12MHz]
#SYS_CLK=24MHz; #148MHz 24MHz]
#UTXA=PA05; //uboot串口tx
#UTRX=1A09000; //uboot串口波特率
UTRX=PB05;串口升级(PB00 PB05 PA05)
RESET=PB01_00_0; //port口_长按时间_有效电平(长按时间有00、01、02、04、08三个值可选, 单位为秒, 当长按时间为00时, 则关闭长按复位功能。)
UPDATE_JUMP=0;

#0:disable
#1:PA9 PA10
#2:USB
#3:PB1 PB2
#4:PB6 PB7

#sdiap=2;
#psram=1;

VLD=4; //VDDIO_LVD位, 0: 1.9V 1: 2.0V 2: 2.1V 3: 2.2V 4: 2.3V 5: 2.4V 6: 2.5V
V 7: 2.6V

69 EOFSET=1; // n * 4k
#####
isd_config.ini dos:ini utf-8(unix) 62% \ 09/21/2023 11:10:11
NORMAL 7 release/AC630N_bt_data_transfer_sdk_release_v0.9.1 < 100% \ 110/110 : 1
:vs

```

可参考文件修改目录下有对应的 isd_config.ini 文件。

11. 修改 AC636N 下载目录下的 testbox_update.c 文件

```

testbox_update.c
static void testbox_bt_classic_update_state_cbk(int type, u32 state, void *priv)
{
    update_ret_code_t *ret_code = (update_ret_code_t *)priv;
    switch (state) {
        case UPDATE_CH_EXIT:
            log_info("state:%x err:%x\n", ret_code->stu, ret_code->err_code);
            if (UPDATE_DUAL_BANK_IS_SUPPORT()) {
                if ((0 == ret_code->stu) && (0 == ret_code->err_code)) {
                    log_info("bt update succ\n");
                    update_result_set(UPDATA_SUCC);
                } else {
                    log_info("bt update fail\n");
                    update_result_set(UPDATA_DEV_ERR);
                }
            } else {
                if ((0 == ret_code->stu) && (0 == ret_code->err_code)) {
                    //update_mode_api_v2(BT_UPDATA);
                    update_mode_api_v2(BT_UPDATA);
                    testbox_bt_classic_update_private_param_fill,
                    testbox_bt_classic_update_before_jump_handle);
                } else if ((0 == ret_code->stu)) { //文件相同的情况
                    cpu_reset();
                }
            }
            break;
    }
}

static u8 ble_update_ready_jump_flag = 0;
u8 ble_update_get_ready_jump_flag(void)
{
    return ble_update_ready_jump_flag;
}

static void testbox_ble_update_private_param_fill(UPDATA_PARAM *p)
{
    #endif
} else if ((0 == ret_code->stu)) { //文件相同的情况, 默认是复位
} if (TCFG_USER_BLE_ENABLE && (TCFG_BLE_DEMO_SELECT != DEF_BLE_DEMO_NULL) \
&& (TCFG_BLE_DEMO_SELECT != DEF_BLE_DEMO_ADV) || defined(CONFIG_MESH_CASE_ENABLE)) \
&& (TCFG_BLE_DEMO_SELECT != DEF_BLE_DEMO_CLIENT)
extern void ble_module_enable(u8 en);
ble_module_enable(0);
u8 cnt = 0;
while (1) {
    log_info("wait disconn\n");
    os_time_dly(2);
    if (cnt++ > 5) {
        break;
    }
}
}
cpu_reset();
}
break;
}
}
void testbox_update_msg_handle(int msg)
{
    log_info("msg:%x\n", msg);
    switch (msg) {
        case MSG_BT_UPDATE_LOADER_DOWNLOAD_START:
            if (UPDATE_MODULE_IS_SUPPORT(UPDATE_BT_LMP_EN)) {
                if (SMART_BOX_EN) && (CONFIG_WATCH_CASE_ENABLE)
                    app_smartbox_task_prepare(0, SMARTBOX_TASK_ACTION_WATCH_TRANSFER, 0);
            }
            #endif
            update_mode_info_t info = {
                .type = BT_UPDATA,
                .state_cbk = testbox_bt_classic_update_state_cbk,
                .p_op_api = &tmp_ch_update_op,
            };
            NORMAL 7 release/AC630N_bt_data_transfer_sdk_release_v0.9.1 < 68% \ 148/216 : 1

```

可参考文件修改目录下有对应的 testbox_update.c 文件。（已经进行过 AC635N 的 testbox_update.c 文件修改可忽略这一步）

12. 修改 AC636N 下载目录下的 update_loader_download.h 文件

```
update_loader_download.h
68
//update result code;
enum {
    UPDATE_RESULT_ERR_NONE = 0,
    UPDATE_RESULT_FILE_SIZE_ERR = 0x1, //文件大小错误
    UPDATE_RESULT_LOADER_SIZE_ERR = 0x2, //loader大小错误
    UPDATE_RESULT_LOADER_VERIFY_ERR, //update loader校验失败
    UPDATE_RESULT_REMOTE_FILE_HEAD_ERR, //读升级文件头错误

    UPDATE_RESULT_LOCAL_FILE_HEAD_ERR = 0x5, //读flash文件头错误
    UPDATE_RESULT_NOT_FIND_TARGET_FILE_ERR, //找不到目标文件
    UPDATE_RESULT_FILE_OPERATION_ERR, //文件操作失败
    UPDATE_RESULT_FLASH_DATA_VERIFY_ERR, //flash数据校验失败

    UPDATE_RESULT_UBOOT_NOT_MATCH = 0x09, //UBOOT不匹配
    UPDATE_RESULT_PRODUCT_INFO_NOT_MATCH = 0x0a, //芯片型号不匹配
    UPDATE_RESULT_ANC_CFG_UPDATE_ERR, //ANC配置升级出错;
    UPDATE_RESULT_CFG_UPDATE_ERR, //配置升级出错

    UPDATE_RESULT_FLASH_ERASE_ERR = 0x0d, //flash 擦失败(可能是写保护)
    UPDATE_RESULT_REMOTE_FILE_NOT_MATCH, //升级文件不匹配
    UPDATE_RESULT_ANC_CFG_UPDATE_ERR, //ANC配置升级出错
    UPDATE_RESULT_ANC_COEF_UPDATE_ERR, //ANC配置升级出错
    UPDATE_RESULT_OTA_TWS_NO_RSP, //对耳同步升级传输数据没有回复
    UPDATE_RESULT_RESOURCE_LIMIT, //资源不足
    UPDATE_RESULT_OTA_TWS_START_ERR, //对耳启动升级失败
    UPDATE_RESULT_OTA_TWS_CRC_ERROR, //对耳校验失败
    UPDATE_RESULT_OTA_APP_EXIT, //升级过程APP强制退出
    UPDATE_RESULT_TWS_NO_CONNECT, //对耳未连接
    UPDATE_RESULT_FILE_SAME, //相同文件升级
};

#include "system/task.h"
typedef struct _update_type_info_t {
    int type;
    u8 task_en;
} update_type_info_t;

NORMAL | release/AC630N_bt_data_transfer_sdk_release_v0.9.0 | update_loader_download.h | cpp | utf-8[unix] | 33% | 60/179 | 1
```

可参考文件修改目录下有对应的 update_loader_download.h 文件。（已经进行过 AC635N 的 update_loader_download.h 文件修改可忽略这一步）

13. 修改 AC636N 下载目录下的 power_interface.h 文件

```
power_interface.h
//MAX hardware wakeup port
#define MAX_WAKEUP_PORT 8
144 #define MAX_RTC_PORT 2

struct wakeup_param {
    const PORT_Flt filter;
    const struct port_wakeup *port[MAX_WAKEUP_PORT];
    const struct port_wakeup *rtc_port[MAX_RTC_PORT];
    const struct charge_wakeup *charge;
    const struct alarm_wakeup *alarm;
    const struct lvd_wakeup *lvd;
    const struct sub_wakeup *sub;
};

struct reset_param {
    u8 en;
    u8 mode;
    u8 level;
    u8 iomap; //Port Group, Port Index
};

struct low_power_operation {
    const char *name;
    u32(*get_timeout)(void *priv);
    void (*suspend_probe)(void *priv);
    void (*suspend_post)(void *priv, u32 usec);
    void (*resume)(void *priv, u32 usec);
    void (*resume_post)(void *priv, u32 usec);
    void (*off_probe)(void *priv);
};

NORMAL | release/AC630N_bt_data_transfer_sdk_release_v0.9.0 | ~/repo_data_trans_0520_de/SDK/include/lib/driver/cpu/br23/asm/power_interface.h | cpp | utf-8[unix] | 47% | 144/306 | 1
```

可参考文件修改目录下有对应的 power_interface.h 文件。（已经进行过 AC635N 的 power_interface.h 文件修改可忽略这一步）