Realtek Ameba Huawei Hilink User Guide

This document introduces Huawei Hilink Protocol on Ameba.

Table of Contents

1 Introduction

Huawei Hilink is a solution in smart home platform. It is composed of LiteOS, Hilink communication protocol, smart device, and smart home cloud.

The classical application is made of a smart device such as air condition, a cellphone with hilink APP and a router which connects with Internet. At first, air condition needs to connect to router with the help of cellphone hilink APP. After that, air condition will register itself to the hilink cloud through Internet. At last, cellphone hilink APP can control the air condition by hilink cloud.

2 Porting Hilink

2.1 Add files to SDK folder

Step 1： Extract Ameba released SDK, eg. sdk-ameba1-v3.4b3.

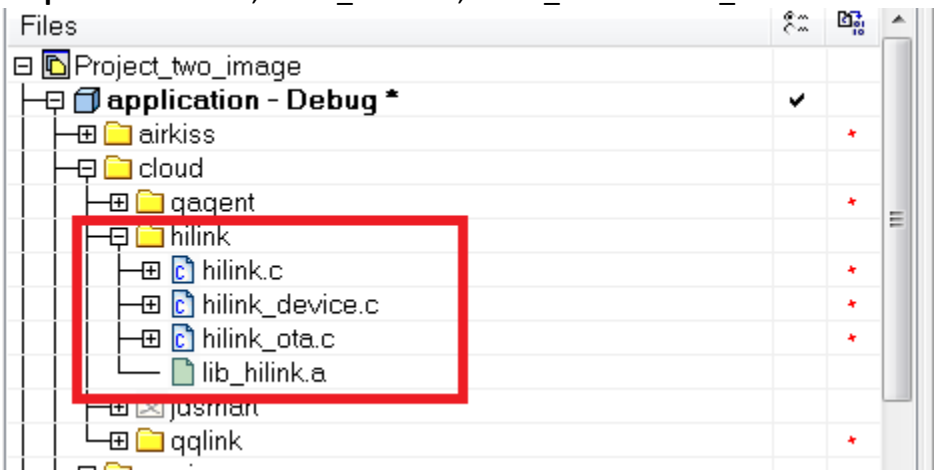
Step 2： Create “hilink” folder in $sdk\component\common\application\.

Step 3： Copy “hilink.c”, “hilink.h”, “hilink_device.c”, “hilink_ota.c”, “hilink_cJSON.h”, “hilink_link.h”, “hilink_osadapter.h”, “hilink_profile.h”, “hilink_socket.h” and “lib_hilink.a” from 3.5_patch_hilink_v1.0.zip to $sdk\component\common\application\hilink

2.2 Add files to IAR project

Step 1: Open IAR project and add a new group “hilink”.

Step 2: Add hilink.c, hilink_device.c, hilink_ota.c and lib_hilink.a into hilink group.



2.3 Other Configuration

2.3.1 platform_opt.h

The file location is $sdk\project\realtek_ameba1_va0_example\inc\platform_opts.h. Please add CONFIG_HILINK define as follows.

```
#define CONFIG_HILINK 1
```

2.3.2 example_entry.c

The file location is $sdk\component\common\example\example_entry.c. You can add hilink entry here if needed.

```
void example_entry(void)
{
    ……
    #if CONFIG_HILINK
        example_hilink();
    #endif
    ……
}
```

2.3.1 image2.icf

The file location is $sdk\ project\realtek_ameba1_va0_example\EWARM-RELEASE\image2.icf. If you want to place hilink in RAM, please add sections ".hilink.text*", ".hilink.data*" and ".xipsec0*" in block .ram_image2.text.

```
33
34 define block .ram_image2.text with fixed order{ section .infra.ram.start*,
35                                                 section .rodata*,
36                                                 block CPP_INIT,
37                                                 section .mon.ram.text*,
38                                                 section .hal.flash.text*,
39                                                 section .hal.gpio.text*,
40                                                 section .text* object main.o,
41                                                 section .text*,
42                                                 section CODE,
43                                                 section .otg.rom.text,
44                                                 section Veneer object startup.o,
45                                                 section __DLIB_PERTHREAD,
46                                                 section .hilink.text*,
47                                                 section .hilink.data*,
48                                                 section .xipsec0*,
49                                                };
50
```

If you want to place hilink in SDRAM, add sections ".hilink.text*", ".hilink.data*" and ".xipsec0*" in block SDRAM

```
176                                          };
177
178 define block SDRAM with fixed order{ section .sdram.text*,
179                                       section .sdram.data*,
180                                       section .mdns.text*,
181                                       section .mdns.data*,
182                                       section .hilink.text*,
183                                       section .hilink.data*,
184                                       section .xipsec0*,
185                                      };
186 place at start of SDRAM_RAM_region { readwrite,
187                                      block SDRAM,
188                                      //block IMAGE1_DBG
189                                     };
```

3 Example

In the default hilink code, we simulate a smart air condition, after all the steps done, we can use cellphone to control the smart air condition, such as turn it on or off, change the windspeed and so on.

3.1 Definition

Almost all the definitions are in hilink.h. In our example, some macros are related with smart air condition, some are related with OTA.

Macros named “HILINK_DEVICE_XXX”, “HILINK_SERVICE_XXX”, “HILINK_SECURITY_XXX” and “HILINK_SSID_TYPE” are smart air condition related. They are used to init hilink sdk and call the hilink API.

“HILINK_DEVICE_SN” is the serial number of the device, we just use the mac address to instead of, “HILINK_DEVICE_SN_LAST_1” is the last byte of “HILINK_DEVICE_SN”, so does the “HILINK_DEVICE_SN_LAST_3”.

When “HILINK_SECURITY_TYPE” is “B”, this means no security will enable. The front part of “HILINK_SECURITY_KEY” should be all zero and the last 5 bytes should be the last 5 bytes of device mac address.

The other macros such as “HILINK_DEVICE_ID”, “HILINK_DEVICE_MANUFACTORYID” and “HILINK_SERVICE_TYPE” is used to definite the smart air condition by Huawei, so if the device is not smart air condition, please contact Huawei to get the detail information.

OTA related macros will illustrate in section 3.4.

3.2 Smart link

After all the macros are confirmed, build the image and download to Ameba demo board by IAR.

Reset Ameba and it will start hilink smart link process

```
===== Enter Image 2 ====
raul_chen version 1.7.2
interface 2 is initialized
interface 1 is initialized
interface 0 is initialized

Initializing WIFI ...
Start LOG SERVICE MODE

#
WIFI initialized

init_thread(53), Available heap 0xc8b0
No AP profile read from FLASH, start Hilink link...
 01000000000000000000000000000000 00000000000000000000000000000000
 00000000000000000000000000000000 00000000000000000000000000000000
 00000000000000000000000000000000 00000000000000000000000000000000
 00000000000000000000000000000000 00000000000000000000000000000000
 00000000000000000000000000000000 00000000000000000000000000000000
 00000000000000000000000000000000 00000000000000000000000000000000
 00000000000000000000000000000000 00000000000000000000000000000000
 00000000000000000000000000000000 00000000000000000000000000000000

######save flash data: , len 256

######read flash data: , len 256

 hilink_link_init

hilink version:v0.5.4[date:2016-07-01 10:46:37 compiler:arm-elf-eabi-gcc]

device ssid:Hi0190018B0000000000000000070918,ssidlen:32!
promisc_set_mgntframe 1
```

Connect the cellphone to the same router which you want Ameba to connect with, and then open the smart home APP, click the “+” on the right top of screen to enter the searching page.

After a few seconds, it will discover device and display it on screen. If no new device is discovered, click the “重新扫描” on the left bottom of the screen to restart the scanning.

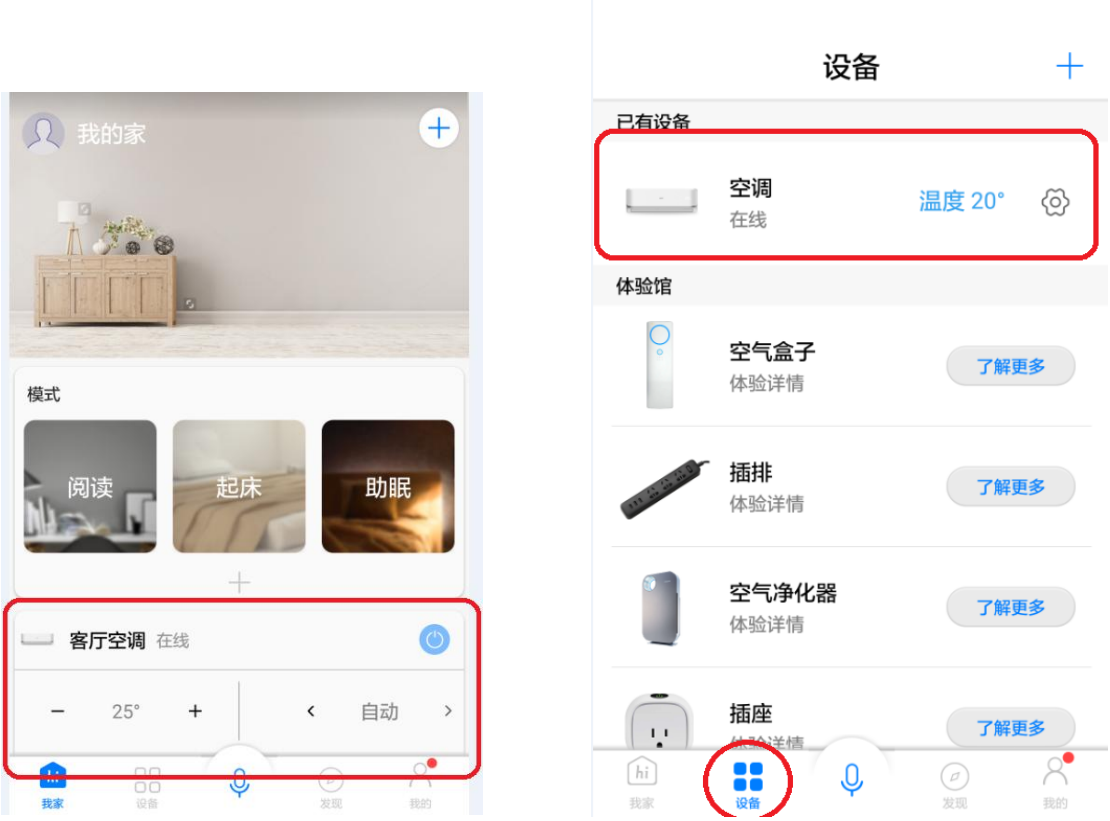
Click the device item and click “下一步” on the next page, smart link will start to configure Ameba, if it works, you will see the log and screen on the cellphone as follows. This means the smart link process is successful and the router wifi profile will be saved to flash. If you reset Ameba again, no need to do smart link again, it will connect to router automatically.


```
LWIP_DHCP: dhcp stop.
Deinitializing WIFI ...
WIFI deinitialized
Initializing WIFI ...
WIFI initialized

RTL8195A[Driver]: set ssid [ADSL_WLAN]

RTL8195A[Driver]: start auth to d8:fe:e3:b3:99:b3

RTL8195A[Driver]: auth success, start assoc

RTL8195A[Driver]: association success(res=9)

RTL8195A[Driver]: set pairwise key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4)

RTL8195A[Driver]: set group key to hw: alg:4(WEP40-1 WEP104-5 TKIP-2 AES-4) keyid:1

Interface 0 IP address : 192.168.0.32
wifi sta connect to ap sucessfull!
[<DEBUG> smartlink/hilink_link.c, 1214] ==> hilink_pack_send_buffer.buffer_out[],outlen[26]

wifi_online_notify Enter!

IP:192.168.0.25, Port: 58279
Hilink Notify Data---------Len=26
00 00 00 00  00 00 00 00  2C 2C 30 30  33 2C 30 31
32 2C 39 31  38 2C 39 30  30 31


notify(1) open socket...


notify(3) socket connect...


notify(4) write...


notify(5) socket close...

wifi_online_notify Exit!

wifi_online_notify OK!
write wifi info to flash,: ssid = ADSL_WLAN, pwd = wlan123987,ssid length = 9, pwd length = 10, channel = 8, security =41943
08
 wlan_wrtie_reconnect_data_to_flash():not the same ssid/passphrase/channel, write new profile to flash
[hilink_link_process]: hilink_link connect successful!!!hilink sdk version v0.5.4[date:2016-07-01 10:46:32 compiler:arm-elf-
eabi-gcc]
```

3.3 Device register

After smart link success, the APP will skip to register page. First enter the device name to mark the device, then choose the location that the device belongs to, at last click the “下一步” item to start register process.

If the process is successful, APP will skip to the home page and you will find the air condition panel on the screen blew. You can also switch to “设备” and find the air condition on the screen above



Now you can control the air condition by cellphone APP. If you turn it on or set the windspeed, you will see log as follows:


```
{"on":1}
Header:
  len   0x00D9
  ver   0x01
  t     0x01
  tkl   0x04
  code  0x02
Options:
  0x0B   1
  0x802   b34c12cb-0e18-4e9c-8606-5cb426d
292b8
  0x803   /devices/5ce0b134-15d4-401a-b4e
c-5b9863a0dfe2/services/1
  0x804   /apps/6dedf94a-c8aa-4412-869a-e
b7925b8fd49/users/27398037
  0x805   yS

Payload:
{"on":1}


##############################
            Set On to 1
##############################Header:
  len   0x00CE
  ver   0x01
  t     0x01
  tkl   0x04
  code  0x45
```

```
{"windSpeed":3}
Header:
  len   0x00DC
  ver   0x01
  t     0x01
  tkl   0x07
  code  0x02
Options:
  0x0B   1
  0x802   b8b07cdf-46a6-44d6-9d62-9898c6b
393c0
  0x803   /devices/5ce0b134-15d4-401a-b4e
c-5b9863a0dfe2/services/1
  0x804   /apps/6dedf94a-c8aa-4412-869a-e
b7925b8fd49/users/27398037
  0x805   yU

Payload:
{"windSpeed":3}


##############################
            Set WindSpeed to 3
##############################Header:
  len   0x00D1
  ver   0x01
  t     0x01
  tkl   0x07
  code  0x45
```

3.4 OTA

The OTA is achieved by the vendor while not by Huawei, the hilink OTA API “hilink_ota_trig” starts the OTA process.

In our example, we use the free storage cloud provider “qiniu” to upload and download the OTA firmware by HTTP. The home page of “qiniu” is http://www.qiniu.com/

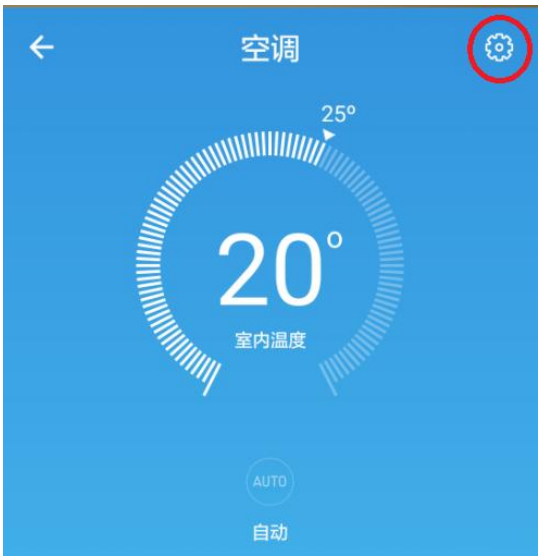
The macros of OTA are also define in hilink.h

HILINK_OTA_SERVER_HOST is the server address which you apply from qiniu

HILINK_OTA_RESOURCE is the OTA firmware name saved on the qiniu cloud

HILINK_VERSION_NAME and HILINK_VERSION_INTRODUCTION are the OTA firmware version information.

Find the device in the “设备” page and click the device to skip into the device page



Click the gear button on the right top to skip into the setting page of this device

Click “升级” to start the OTA process, the device will connect to the storage cloud and download the OTA firmware by HTTP and write it into flash. During the OTA, a progress bar will be displayed

After all the firmware is downloaded and written to flash, the device will restart and at last OTA will be finished on cellphone APP.


```
Payload:
{"errcode":0}

Invalid handle method

http content-length = 352088 bytes, download resource size = 352088 bytes
[hilink_ota_upgrade] Update OTA success!
[hilink_ota_upgrade] Ready to rebootHeader:
  len  0x003B
  ver  0x01
  t    0x01
  tkl  0x02
  code 0x02
Options:
  0x0B  .sys
  0x0B  data
  0x801  90997dcc-6f01-45d6-8234-163ba3c
f2415
  0x805  Dy


;R sysdata90997dcc-6f01-45d6-8234-163ba3cf2415DDy{"devId":"5ce0b134-15d4-401a-b4ec-5b9863a0dfe2","services":
                [{"sid":"ota","data":                                    {"progress":100,"bootTime":10000}}]}]
```