Airoha IoT SDK for WiFi Release Notes

Version: 4.8.0

Release date: 18 January 2019

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. ("Airoha") and/or its licensor(s). Airoha cannot grant you permission for any material that is owned by third parties. You may only use or reproduce this document if you have agreed to and been bound by the applicable license agreement with Airoha ("License Agreement") and been granted explicit permission within the License Agreement ("Permitted User"). If you are not a Permitted User, please cease any access or use of this document immediately. Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited. THIS DOCUMENT IS PROVIDED ON AN "AS-IS" BASIS ONLY. AIROHA EXPRESSLY DISCLAIMS ANY AND ALL WARRANTIES OF ANY KIND AND SHALL IN NO EVENT BE LIABLE FOR ANY CLAIMS RELATING TO OR ARISING OUT OF THIS DOCUMENT OR ANY USE OR INABILITY TO USE THEREOF. Specifications contained herein are

Document Revision History

Revision	Date	Description
3.0.0	7 March 2016	Initial release for SDK v3.0.0.
3.1.0	31 March 2016	• Added support for LinkIt 2523 HDK. • Added the change logs for SDK v3.1.0.
3.2.0	2 May 2016	• New Bluetooth stack support in SDK v3.2.0. • New SDK API support for Bluetooth, Wi-Fi and HAL modules. • IAR embedded workbench IDE support for LinkIt 2523 HDK.
3.3.0	30 June 2016	Added the release notes for SDK v3.3.0.
3.3.1	4 July 2016	• Update the HAL module in MT2523 API reference manual. • Update KEIL EULA license path.
3.3.2	28 July 2016	Enhanced the Wi-Fi throughput on MT76x7.
4.0.0	2 September 2016	• Added support for new Bluetooth stack on MT2523. • Updated new features, applied bug fixes and added known issues for SDK 4.0.0.
4.1.0	4 November 2016	• Added the list of documents supporting the chipset. • Updated the features, applied bug fixes and added known issues for SDK 4.1.0.
4.2.0	13 January 2017	• Added support for MT2533D. • Added a migration guide to migrate applications from SDK v4.1.0 to SDK v4.2.0.
4.2.1	10 March 2017	• Fixed an issue in debugger and an issue in Wi-Fi reference design.
4.2.2	12 April 2017	• Fixed five MT2523 issues.
4.3.0	5 May 2017	• Added support for MT7682S. • Fixed software issues. • Added a migration guide to migrate applications from SDK v4.2.x to SDK v4.3.0.
4.3.1	2 June 2017	• Fixed two MT2523 issues.
4.5.0	7 July 2017	• Added support for MT7686. • Fixed software issues. • Added a migration guide to migrate applications from SDK v4.3.x to SDK v4.5.0.
4.5.1	4 August 2017	• Fixed software issues.
4.6.0	20 September 2017	• Added new features for MT25x3/MT7686/MT7682 • Fixed software issues. • Added a migration guide to migrate applications from SDK v4.5.0 to SDK v4.6.0.
4.6.1	15 November 2017	• Fixed software issues.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

Revision	Date	Description
4.6.2	21 December 2017	• Added IAR projects for MT7686 • Fixed software issues.
4.7.0	21 May 2018	• Added new features for MT7682/MT7686/MT7697/MT2523 • Fixed software issues. • Added a migration guide to migrate applications from SDK v4.6.2 to SDK v4.7.0.
4.7.1	19 November 2018	• Fixed software issues.
4.8.0	18 January 2019	• Added new features for MT7682/MT7686/MT7687/MT7697 • Upgraded software modules • Fixed software issues • Added a migration guide for migrating applications from SDK v4.7.1 to SDK v4.8.0

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

Table of Contents

1. Introduction 1

1.1. Architecture layout of the SDK 1
1.2. Knowledgebase 3

2. SDK Version 4.8.0 5

2.1. Main changes 5
2.2. Known issues 6
2.3. Migration 6

3. SDK Version 4.7.1 10

3.1. Main Changes 10

4. SDK Version 4.7.0 11

4.1. Main changes 11
4.2. Known issues 11
4.3. Migration 11

5. SDK Version 4.6.2 13

5.1. Main changes 13

6. SDK Version 4.6.1 14

6.1. Main changes 14

7. SDK Version 4.6.0 15

7.1. Main changes 15
7.2. Known issues 15

8. SDK Version 4.5.1 16

8.1. Main changes 16

9. SDK Version 4.5.0 17

9.1. Main changes 17
9.2. Known issues 17
9.3. Migration 18

10. SDK Version 4.3.1 22

10.1. Main changes 22
10.2. Known issues 22

11. SDK Version 4.3.0 23

11.1. Main changes 23
11.2. Known issues 23
11.3. Migration 24

12. SDK Version 4.2.2 26

12.1. Main changes 26
12.2. Known issues 26

13. SDK Version 4.2.1 27

13.1. Main changes 27
13.2. Known issues 27

14. SDK Version 4.2.0 28

14.1. Main changes 28
14.2. Known issues 28
14.3. Migration 29

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

15. SDK Version 4.1.0 33
15.1. Main changes 33
15.2. Known issues 33
16. SDK Version 4.0.0 34
16.1. Main changes 34
16.2. Known issues 34
17. SDK Version 3.3.2 35
17.1. Main changes 35
18. SDK Version 3.3.0 36
18.1. Main changes 36
18.2. Known issues 36
19. SDK Version 3.2.0 37
19.1. Main changes 37
20. SDK Version 3.1.0 38
20.1. Main changes 38
20.2. Known issues 38
21. SDK Version 3.0.0 39
21.1. Known issues 39

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

List of Tables and Figures

Table 1. Documentation relevance for different chipsets 4
Table 2. FreeRTOS API change list....6
Table 3. Folders and files to migrate....24
Table 4. Structures and enumerations migration for Bluetooth notification services 29
Table 5. Structures and enumerations migration for Bluetooth callback manager services....30
Table 6. Structures and enumerations migration for Bluetooth LE ANCS 31

Figure 1.1-1. Architecture layout of the SDK for LinkIt 7687 HDK 1
Figure 1.1-2. Architecture layout of the SDK for MT7682/MT7686 HDK....2

© 2015 - 2019 Airoha Technology Corp.

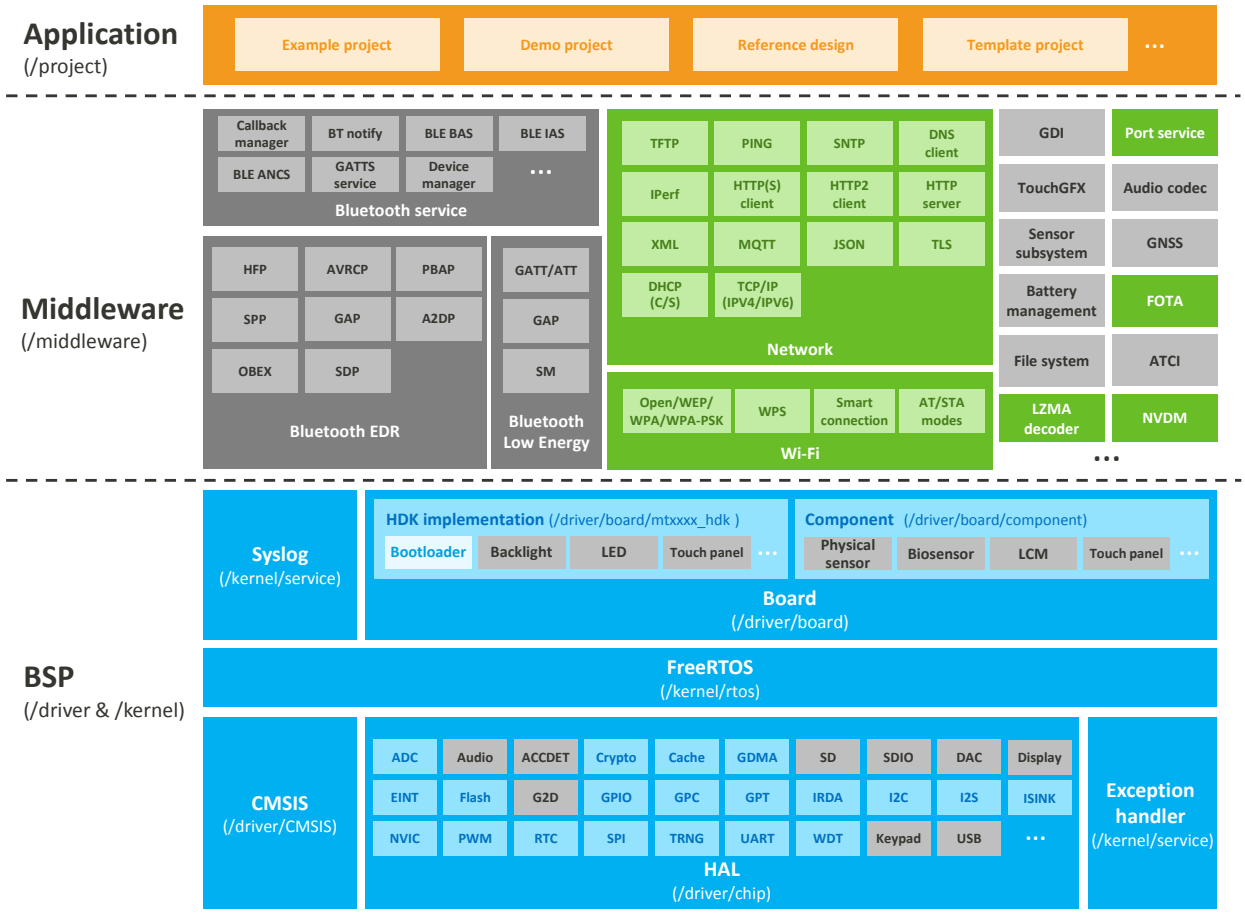
This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

1. Introduction

Airoha IoT SDK provides comprehensive software solution. The SDK supports hardware abstraction layers (HAL), peripheral drivers, Wi-Fi module, FreeRTOS, Lightweight IP (lwIP) and other features.

1.1. Architecture layout of the SDK

The three-layer architecture layout of the SDK for LinkIt 7687 HDK includes Applications, Middleware and BSP, as shown in Figure 1.1-1.



Modules shown in gray color are not supported by MT7687

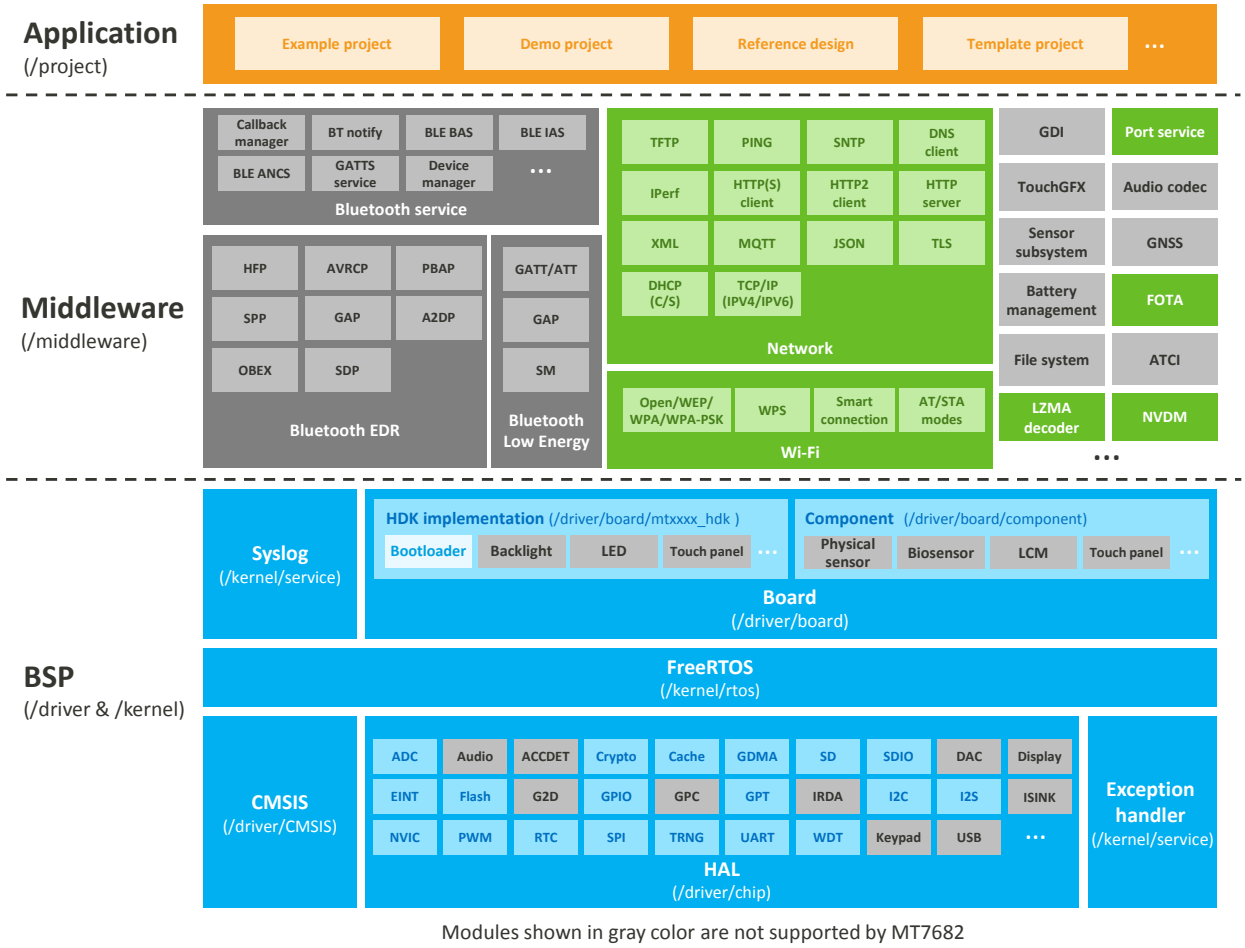
Figure 1.1-1. Architecture layout of the SDK for LinkIt 7687 HDK

A functional block in grey means the HDK does not support the feature. The top layer includes the application projects running on the SDK. They are based on Middleware, OS and HAL layers. These layers provide rich features for application development, such as the Middleware provides the network connectivity, Wi-Fi and Bluetooth Low Energy Stack. The FreeRTOS provides the underlying real-time operating system.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

The three-layer architecture layout of the SDK for MT7682/MT7686 HDK includes Applications, Middleware and BSP, as shown in Figure 1.1-2.



Modules shown in gray color are not supported by MT7682

Figure 1.1-2. Architecture layout of the SDK for MT7682/MT7686 HDK

A functional block in grey means the HDK does not support the feature. The top layer includes the application projects running on the SDK. They are based on Middleware, OS and HAL layers. These layers provide rich features for application development, such as the Middleware provides the network connectivity, Wi-Fi .The FreeRTOS provides the underlying real-time operating system.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

1.2. Knowledgebase

The released SDK includes documentations to guide developers through each module and its features, in a convenient and developer-oriented approach.

The documentations are located under SDK's doc folder.

- Airoha IoT SDK for RTOS Get Started. This guide covers the SDK features, step- by-step setup of the development environment and its usage.
- Airoha IoT SDK for RTOS API reference manual. This reference manual provides detailed description of the APIs in the SDK.
- Airoha IoT SDK for RTOS Open Source Components Guide. This document guides you through the open source modules and the features used in the SDK.
- Airoha IoT SDK for RTOS Memory Layout Developer's Guide. This guide provides details on the memory layout of the SDK, and how to adjust the memory layout for a custom application.
- Airoha IoT SDK for RTOS Wi-Fi Developer's Guide. This document complements the Wi-Fi API reference manual.
- Airoha IoT SDK for RTOS System Log Developer's Guide. This document guides you through the usage of the system logging feature provided in the SDK.
- Airoha IoT SDK for RTOS Firmware Update Developer's Guide. This document guides you to use the FOTA and how to adjust the memory usage of FOTA.
- Airoha IoT SDK for RTOS Internet Middleware API Reference Manual. This reference manual provides details on the usage of internet protocol APIs.
- Airoha IoT SDK GCC Build Environment Guide. This document provides details on how to create and build a project, and how to create a module, with the SDK in the GCC build environment.
- Airoha IoT SDK for RTOS Bluetooth Developer's Guide. This document guides you through the supported Bluetooth library and its usage with reference examples.
- Airoha IoT SDK for RTOS Power Mode Developer's Guide. This document addresses the MCU system's power mode configuration and power consumption measurement focused on power modes provided by Airoha IoT SDK for RTOS.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. ("Airoha") and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

The related chipsets of each document under <sdk_root>/doc folder are listed in Table 1.

Table 1. Documentation relevance for different chipsets

Document file name	MT7687F/ MT7682S/ MT7686	MT7697/ MT7697D
Airoha_IoT_SDK_for_RTOS_Bluetooth_Developers_Guide.pdf		√
Airoha_IoT_SDK_for_RTOS_Firmware_Update_Developers_Guide.pdf	√	√
Airoha_IoT_SDK_for_RTOS_Get_Started_Guide.pdf	√	√
Airoha_IoT_SDK_for_RTOS_Internet_and_Open_Source_Software_Guide.pdf	√	√
Airoha_IoT_SDK_for_RTOS_Memory_Layout_Developers_Guide.pdf	√	√
Airoha_IoT_SDK_for_RTOS_Power_Mode_Developers_Guide.pdf	√	√
Airoha_IoT_SDK_for_RTOS_System_Log_Developers_Guide.pdf	√	√
Airoha_IoT_SDK_for_RTOS_WiFi_Developers_Guide.pdf	√	√
Airoha_IoT_SDK_SDK_GCC_Build_Environment_Guide.pdf	√	√
Airoha_IoT_SDK_for_RTOS_Wi-Fi_Migration_Developers_Guide.pdf	√	√
Airoha_IoT_SDK_SDK_Release_Notes.pdf	√	√

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

2. SDK Version 4.8.0

2.1. Main changes

- Software features and optimization
  - Upgrading FreeRTOS from v8.2.0 to v10.1.1
  - Upgrading lwIP from v1.5.0 to v2.1.2. (include LWIP IPv6)
  - Upgrading mbed TLS from v2.1.0 to v2.7.8
  - Upgrading cJSON to v1.7.8
  - [MT7682/MT7686] Added OPUS codec support
  - [MT7682/MT7686] Support File system (FatFS) for NOR flash
  - [MT7682/MT7686/MT7687/MT7697] [add-on] Added Airkiss AES support with default enabled
  - [MT7682/MT7686] [add-on] Added secure boot support
  - [MT7682] [add-on] Added CMCC Andlink support
  - [MT7686] [add-on] Added story_reader_ref_design project
  - [MT7697] Support Mesh OTA
- Bug fixes
  - [MT7682/MT7686] Fixed a problem where the Max Tx Power setting in Country IE of Beacon packet always zero and cannot be updated by wifi_config_set_country_code API.
  - [MT7682/MT7686] Fixed a problem where the smart connection could fail and show “connsys Abnormal Interrupt: WSAR (0x200)” during a stress test.
  - [MT7682/MT7686] Fixed a problem where the system hanged when repeatedly switching to opmode.
  - [MT7682/MT7686] Fixed a problem that occurred in Soft AP mode where the system crashed when two connected STAs pinged each other.
  - [MT7682/MT7686] Fixed an issue that in SoftAP mode, IphoneX may connect device fail and show cipher error.
  - [MT7687/MT7697] Fixed a problem that occurred in STA mode where the device disconnected from an AP router and failed to connect it again after a device sent a large amount of data, causing a padding error.
  - [MT7687/MT7697] Synced the WPA_Supplicant security patch "unauthenticated eapol-key" from https://w1.fi/security/2018-1/.
  - [MT7687/MT7697] Resolved an issue with fast power saving in STA mode, where the device could not disconnect from the AP router when the AP router powered off.
  - [MT7687/MT7697] Removed the power capability IE in frames as device has no support in Wi-Fi TPC.
  - [MT7687/MT7697] Resolved an issue where the AP mode beacon IE supported channel list was wrong.
  - [MT7687/MT7697] Resolved an issue with system stability during stress testing, stack may overflow.
  - [MT7687/MT7697] Fixed a system hang issue caused by race condition only occurring at boot up if BLE and Wi-Fi were enabled.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

  - [MT7687/MT7697] Removed the AP channel report IE because our device does not support the 802.11k feature and this IE is not useful.
  - [MT7687/MT7697] Fixed a system crash due to the specific AP router sending wrong info in Beacon.
- Notes
  - From this version of SDK started to release by product line, this version is for Wi-Fi product line release.
  - Add-on is not included in standard SDK package, please contact the Airoha AE team to get an add-on package if needed.
  - It includes updated version of some open source software modules in SDK V4.8 (e.g., lwIP). It may increases stack memory usage with updated software module. Please try to increase the stack size of the application task if there is a system crash with “stack overflow” message in the system log.

2.2. Known issues

There are known issues with this version of the SDK. Avoid the following:

- [MT7686 story_reader_ref_design] Story reader may not play some m4a content completely because content server unexpected terminates data transmission occasionally.

2.3. Migration

Migrate the following module-based applications when upgrading the SDK from version 4.7.1 to version 4.8.0:

1) FreeRTOS is upgraded from v8.2.0 to v10.1.1, please refer to

https://www.freertos.org/FreeRTOS-V10.html to get more information about the FreeRTOS Version 10.

The Table 2 list the phase out APIs after v8.2.0, please be careful to these APIs.

Table 2. FreeRTOS API change list

File	v8.2.0	v10.1.1	Note
task.h	xTaskGenericCreate	N/A	xTaskGenericCreate is phase out since v9.0.0. Another API xTaskCreateStatic can do the similar thing. Please refer to https://www.freertos.org/xTaskCreateStatic.html to get more info
queue.h	xQueueAltGenericSend xQueueAltGenericReceive xQueueAltSendToFront xQueueAltSendToBack xQueueAltReceive xQueueAltPeek	N/A	Phase out since v9.0.0
semphr.h	xSemaphoreAltTake xSemaphoreAltGive	N/A	Phase out since v9.0.0

2) LWIP upgrade from 1.5.0 to 2.1.2

- dhcp pointer has been removed from struct netif

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

- How to migrate - netif_dhcp_data(netif) macro which located in <sdk_root>/middleware/third_party/lwip/src/include/dhcp.h.

- ETHADDR16_COPY macro has been removed from <sdk_root>/middleware/third_party/lwip/src/include/netif/ethernet.h
  - How to migrate – replace ETHADDR16_COPY with memcpy.
    - ~~ETHADDR16_COPY(&ethhdr->src, (struct eth_addr*)(netif->hwaddr));~~
    - memcpy(&ethhdr->src, (struct eth_addr*)(netif->hwaddr), ETH_HWADDR_LEN);

- inet_addr_from_ipaddr has been removed from <sdk_root>/middleware/third_party/lwip/src/include/lwip/inet.h
  - How to migrate – replace inet_addr_from_ipaddr with inet_addr_from_ip4addr

- When IPv4 and IPv6 are both enabled, the ip_addr_t is combined with IPv4 and IPv6 address.
  - How to get IPv4 address – use macro ip_2_ip4
  - How to get IPv6 address – use macro ip_2_ip6
  - The both macros are located in <sdk_root>/middleware/third_party/lwip/src/include/lwip/ip_addr.h

3) API parameters changes for Bluetooth mesh add CTL/HSL server:

   `<sdk_root>/project/mt7697_hdk/apps/ble_mesh_vendor_device/src/mesh/mesh_app_vendor_device.c.`

   - Related examples:
     - ble_mesh_vendor_device;
   - How to migrate
     - Replace the old version of function mesh_create_ctl_server() ,

```
static void mesh_create_ctl_server(uint16_t element_index)
{
    uint8_t element_count;

    ......

    /* CTL server for DemoKit LED, containing 2 elements for Light CTL
Server, Light CTL Temperature Server */

bt_mesh_model_lighting_add_ctl_setup_server(&g_model_handle_ctl_server,
&element_count, element_index, light_ctl_server_msg_handler, NULL);
    LOG_I(mesh_app, "ctl element_index = %d element_count = %d",
element_index, element_count);
    gCTL_server->element_index = element_index;
    gCTL_temperature_server->element_index = gCTL_server->element_index +
1;

    /* Binding temperature with gCTL_server */
    bind_ctl_temperature(gCTL_server, gCTL_temperature_server,
MESH_MODEL_STATE_LIGHTING_CTL_TEMPERATURE, MESH_MODEL_BINDING_BOTH_VALUE);
```

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

```
}
```

- as the new version;

```
static void mesh_create_ctl_server(uint16_t element_index)
{
    uint8_t element_count;
    uint16_t *element_list;

    ......

    /* CTL server for DemoKit LED, containing 2 elements for Light CTL
Server, Light CTL Temperature Server */

bt_mesh_model_lighting_add_ctl_setup_server(&g_model_handle_ctl_server,
&element_count, &element_list, element_index,
light_ctl_server_msg_handler, NULL);
    LOG_I(mesh_app, "ctl element_index = %d element_count = %d",
element_index, element_count);
    gCTL_server->element_index = element_index;
    if(element_list != NULL && element_list[0] == element_index)
    {
        gCTL_temperature_server->element_index = element_list[1];
    }
    else
        LOG_I(mesh_app,"[ERROR]CTL server element_list wrong!\n");

    //bt_mesh_app_util_free element_list allocated when adding model.
    bt_mesh_app_util_free((uint8_t*)element_list);

    /* Binding temperature with gCTL_server */
    bind_ctl_temperature(gCTL_server, gCTL_temperature_server,
MESH_MODEL_STATE_LIGHTING_CTL_TEMPERATURE, MESH_MODEL_BINDING_BOTH_VALUE);
}
```

- Replace the old version of function mesh_create_hsl_server() ,

```
static void mesh_create_hsl_server(uint16_t element_index)
{
    uint8_t element_count;

    ......

    /* HSL server for DemoKit LED, containing 3 elements for Light HSL
Server, Light HSL Hue Server, Light HSL Saturation Server */

bt_mesh_model_lighting_add_hsl_setup_server(&g_model_handle_hsl_server,
&element_count, element_index, light_hsl_server_msg_handler, NULL);
    LOG_I(mesh_app, "hsl element_index = %d element_count = %d\n",
element_index, element_count);
    gHSL_server->element_index = element_index;
    gHSL_hue_server->element_index = gHSL_server->element_index + 1;
    gHSL_saturation_server->element_index = gHSL_hue_server->element_index
+ 1;

    /* Binding hue and saturation with gHSL_server */
    bind_hsl_hue(gHSL_hue_server, gHSL_server,
MESH_MODEL_STATE_LIGHTING_HSL_HUE, MESH_MODEL_BINDING_BOTH_VALUE);
    bind_hsl_saturation(gHSL_saturation_server, gHSL_server,
MESH_MODEL_STATE_LIGHTING_HSL_SATURATION, MESH_MODEL_BINDING_BOTH_VALUE);
}
```

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

- as the new version;

```
static void mesh_create_hsl_server(uint16_t element_index)
{
    uint8_t element_count;
    uint16_t *element_list;

    ......

    /* HSL server for DemoKit LED, containing 3 elements for Light HSL
Server, Light HSL Hue Server, Light HSL Saturation Server */

bt_mesh_model_lighting_add_hsl_setup_server(&g_model_handle_hsl_server,
&element_count, &element_list, element_index,
light_hsl_server_msg_handler, NULL);

    LOG_I(mesh_app, "hsl element_index = %d element_count = %d",
element_index, element_count);
    gHSL_server->element_index = element_index;

    if(element_list != NULL && element_list[0] == element_index)
    {
        gHSL_hue_server->element_index = element_list[1];
        gHSL_saturation_server->element_index = element_list[2];
    }
    else
        LOG_I(mesh_app,"[ERROR]HSL server element_list wrong!\n");

    //bt_mesh_app_util_free element_list allocated when adding model.
    bt_mesh_app_util_free((uint8_t*)element_list);

    /* Binding hue and saturation with gHSL_server */
    bind_hsl_hue(gHSL_hue_server, gHSL_server,
MESH_MODEL_STATE_LIGHTING_HSL_HUE, MESH_MODEL_BINDING_BOTH_VALUE);
    bind_hsl_saturation(gHSL_saturation_server, gHSL_server,
MESH_MODEL_STATE_LIGHTING_HSL_SATURATION, MESH_MODEL_BINDING_BOTH_VALUE);
}
```

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

3. SDK Version 4.7.1

3.1. Main Changes

- Bug fixes
  - [MT7682, MT7686] Resolved an issue where the WDT timeout did not always operate as expected.
  - [MT7682, MT7686] Fixed a problem where the system compiled by IAR crashed because there was no 25K Wi-Fi stack data configuration in the linker script.
  - [MT7682, MT7686] Resolved an issue that occurred because the system used an incorrect power limited value when it used 20 MHz bandwidth in STA mode, and the connected AP router used 40 MHz bandwidth.
  - [MT7682, MT7686] Fixed a problem where the system crashed when a device disconnected from an unassociated STA in AP mode.
  - [MT7682, MT7686] Resolved an issue where the Wi-Fi Tx Power efuse setting could not load and take effect.
  - [MT7682, MT7686] Resolved an issue that caused some Wi-Fi WPS connections to fail and not report the Wi-Fi WPS connection event to the upper layer.
  - [MT7682, MT7686] Resolved an issue that prevented a device in STA mode from connecting to an AP that was changing security mode from open to WPA.
  - [MT7682, MT7686] Resolved an issue that prevented a device in STA mode from sending TX packets if it was in fast power saving mode.
  - [MT7682, MT7686] Resolved an issue where a device in STA mode could not connect to the AP because it was continuously switching to opmode.
  - [MT7682, MT7686] Resolved an issue where the device failed to start when using an internal 32 kHz clock while doing Wi-Fi radio off operation.
  - [MT7682, MT7686] Fixed a problem that caused the system to crash when doing a stress test in Wi-Fi repeater mode.
  - [MT7682, MT7686] Resolved an issue where a smartphone could not connect to a device in repeater mode with the WEP security type.
  - [MT7682, MT7686] Made a change so that devices would consume less power while scanning for Wi-Fi.
  - [MT7682, MT7686] Fixed a problem that caused the system to crash while switching to opmode.
- Notes
  - [MT7682] Added support for MT7682SN.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

4. SDK Version 4.7.0

4.1. Main changes

- Software features and optimization
  - [MT7697] Added support for SIG Bluetooth Mesh v1.0.
  - [MT7682, MT7686] Added support for IAR HAL examples.
  - [MT5932] Added Wi-Fi host API on MT2523 for scan function.
  - [MT7682, MT7686] Optimized packet transmission mechanism to reduce packet loss.
  - [MT7682] Added sdio_host_interrupt and sdio_slave_interrupt example projects.
- Bug fixes
  - [MT76x7] A Wi-Fi device may fail to connect to a Wi-Fi router by WPS if there are more than four nearby Wi-Fi routers with stronger signals than the target Wi-Fi router.
  - [MT7682, MT7686] A Wi-Fi device may crash or fail to connect to a Wi-Fi router when in Wi-Fi repeater mode.
- Notes
  - The name of SDK is changed from LinkIt SDK to Airoha IoT SDK since this version.
  - Corrected Bluetooth support descriptions in chapter “SDK Version 4.3.0”.
  - [BLE Mesh] The Friend node only supports storing all incoming messages to one Low Power node.

4.2. Known issues

There are known issues with this version of the SDK. Avoid the following:

- [MT7697] Occasionally a mesh device may stop receiving BLE data while Wi-Fi is in traffic.

4.3. Migration

Migrate the following module-based applications when upgrading the SDK from version 4.6.2 to version 4.7.0:

1) API parameters changes for IPv6 create link local address:

   `<sdk_root>/middleware/third_party/lwip/src/core/netif.c`.

   - Related examples:
     - mdns_publish_service
   - How to migrate — Replace netif_create_ip6_linklocal_address(netif, 0) with netif_create_ip6_linklocal_address(netif, 0, NULL).

2) Enum parameters change for the duplicate L channel function:

   `<sdk_root>/driver/chip/<mt7686/mt7687>/inc/hal_platform.h`

   Remove unused enum parameters and structure's member:

   `<sdk_root>/driver/chip/<mt7686/mt7687>/inc/hal_platform.h`

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

- Related examples:
  - hal_i2s_testcase;
  - i2s_internal_loopback;
  - audio_ref_design; and
  - bt_hfp_codec.
- How to migrate
  - Replace HAL_I2S_TX_MONO_DUPLICATE_DISABLE with HAL_I2S_TX_DUPLICATE_DISABLE.
  - Remove the rx_down_rate setting from hal_i2s_config_t.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

5. SDK Version 4.6.2

5.1. Main changes

- Software features and optimization
  - [MT7686] Added network related IAR project support.
- Bug fixes
  - [MT76x7, MT7682, MT7686, MT5932] Fixed a system crash that occurred when connected to an AP which was sending a probe response frame that was 515 bytes or 516 bytes long.
  - [MT76x7, MT7682, MT7686, MT5932] Fixed an issue that occurred when connecting to an AP which was sending a group key with index 0 in a WPA2PSK or WPAPSK procedure.
  - [MT76x7, MT7682, MT7686, MT5932] Fixed an issue where the device disconnected from an AP which had more than one station connected, and was sending multicast or broadcast data packets while doing group re-key with its connected stations.
  - [MT7682, MT7686, MT5932] Fixed an issue where the system could not go into sleep mode by calling wifi_config_set_radio (0) after the device disconnected from an AP.
  - [MT7682, MT7686, MT5932] Fixed an issue where the Wi-Fi power saving mode setting which was changed by the Wi-Fi power saving mode API, changed back to the default setting when the device reconnected to an AP.
  - [MT76x7] Fixed a system crash that occurred when the device connected to an AP which had added the device to its blacklist.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

6. SDK Version 4.6.1

6.1. Main changes

- Software features and optimization
  - [MT7682, MT7686] Added `iot_sdk_demo` IAR project support.
- Bug fixes
  - [MT7682, MT7686, MT76x7] Fixed WPA2 KRACK vulnerability issue.
  - [MT7682, MT7686, MT76x7] Fixed network example projects crash issue.
  - [MT7682, MT7686, MT76x7] Cannot connect to WEP Wi-Fi access point by Smart Connection.
  - [MT7682, MT7686] Fixed Wi-Fi repeater mode not working issue.
  - [MT2523x, MT2533D] Sometimes A2DP music is not smooth with SBC codec.
  - [MT2523x, MT2533D] Sometimes the system crashes when playing music and the system runs out of memory.
  - [headset_ref_design] Local music sometimes stops playing after making a voice call.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

7. SDK Version 4.6.0

7.1. Main changes

- Software features and optimization
  - [MT25x3] Added support for A2DP source of Bluetooth EDR.
  - [MT25x3] Added support for Bluetooth LE multi-link concurrent operation (1 slave and 4 master Bluetooth LE links).
  - [MT7682, MT7686] Added support for Wi-Fi WPS.
- Bug fixes
  - [`inear_noodles_headset_ref_design`] Cannot power on the device after it's powered off by unplugging the USB charger.
  - [`inear_noodles_headset_ref_design`] Local music sometimes stops playing after switching to the next song.
  - [`inear_noodles_headset_ref_design`] Local music volume is too high.
  - [`cord_free_headset_ref_design`] Local music stops playing on a slave headset after disconnecting from Bluetooth on a smartphone.
  - [`bioband_ref_design`] HRV cannot function properly, after executing the blood pressure unit.
  - [MT7682, MT7686] Wi-Fi throughput is less than expected in A-MPDU. There are two frame aggregation mechanisms for 802.11n (A-MSDU and A-MPDU) that are used to reduce the package header size and improve the overall throughput. The type of frame aggregation is determined between Wi-Fi AP and Wi-Fi station. A memory leak issue is fixed when in A-MPDU resulting low throughput.

7.2. Known issues

There are known issues in this version of the SDK, avoid the following:

- [MT2523x, MT2533D] If Bluetooth LE scanning or advertising takes up to 2 hours or more, a low probability Bluetooth LE link loss may be triggered due to collision between Bluetooth LE links and Bluetooth LE scanning, advertising.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. ("Airoha") and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

8. SDK Version 4.5.1

8.1. Main changes

- Bug fixes
  - [MT2523x, MT2533D] Sometimes the HFP voice is not smooth when transmitting BLE data at the same time.
  - [cord_free_headset_ref_design] Fixed makefile setting of MTK_BT_AT_COMMAND_ENABLE.
  - [headset_ref_design] Fixed inconsistent issue of “FOTA_RESERVED_BASE” address in the memory map.
  - [headset_ref_design] Smartphone APK sometimes cannot indicate “Update success” after completing FOTA process.
  - [inear_noodles_headset_ref_design] A2DP music sometimes doesn't resume after ending a voice call.
  - [inear_noodles_headset_ref_design] Cannot play A2DP music after redialing a voice call several times.
  - [inear_noodles_headset_ref_design] DUT crash occurs after plugging in a USB cable to power on the device.
  - [inear_noodles_headset_ref_design] LED indication of Bluetooth connection and A2DP music playback didn't respond timely.
  - [inear_noodles_headset_ref_design] Local MP3 playback didn't resume after receiving a key tone from smartphone.
  - [inear_noodles_headset_ref_design] No "Connected" voice prompt sound while powering on the device by USB.
  - [MT25x3] Local MP3 playback sometimes doesn't resume after rejecting a voice call.
  - [MT25x3] Power consumption of A2DP playback and MP3 local playback is too high.
  - [MT25x3] System cannot reboot successfully when FOTA package size is over 650KB during FOTA process.
  - [MT7682, MT7686] System cannot reboot successfully by using CLI reboot command during FOTA process.
  - [MT7682, MT7686] Sometimes the system doesn't enter deep sleep mode when connecting to a Wi-Fi AP or enabling sleep manager debug log.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

9. SDK Version 4.5.0

9.1. Main changes

- Software features and optimization
  - [MT7686] Added support for MT7686 chipset with the same feature set of MT7682S.
  - [MT2523 + MT5932] Added a reference design to provide Wi-Fi connectivity on LinkIt 2523 HDK with MT5932 HDK. (`<sdk_root>/project/mt2523_hdk/apps/wifi5932_ref_design`)
  - [MT7682] Added a reference design to support MP3 decoding, AMR decoding and AMR encoding. (`<sdk_root>/project/mt7682_hdk/apps/audio_ref_design`)
  - [MT7682] Added support for W-Fi AP and STA concurrent mode (Repeater mode).
  - [MT25x3] Added support for Audio/Video Remote Control Profile (AVRCP) 1.6 of Bluetooth EDR.
  - [MT25x3] Added support for stereo with two headsets through A2DP and playing from local MP3 file (Advanced Wireless Stereo, AWS).
- Bug fixes
  - [MT25x3] Changing the configuration of connection link in BLE or EDR Piconet might cause receiving data errors.
  - [MT25x3] While inquiry scan and page scan are enabled and EDR link is established, the EDR link might disconnect.
  - [MT25x3] When a headset pauses the local music with AWS and a link is connected between the headset and the host, an incoming mobile terminated (MT) call of the host will result in headset crash.
  - [MT25x3] While debugging with µVision Tools (IAR), suspending the MT25x3 device for over 30 seconds will result in reset by watchdog timer.
  - [MT25x3] If MT call is received from host to the speaker through hands-free profile, the audio source in speaker will not switch to the call, if a Bluetooth speaker is playing MP3 with AWS.
  - [MT7682] Setting Wi-Fi profile with SSID and then setting PSK will result in Wi-Fi connection (in STA mode) failure.
  - [MT7682] In Open-WEP connection, if the connection between MT768x STA and an AP fails, the STA device will not inform the application about the connection failure.
  - [MT7682] If a connection is maintained for over 8 hours between MT768x host in STA mode and an AP with password configuration, the data decryption might be incorrect.
  - [MT7682] In Wi-Fi station mode with special low power state (PS mode is equal to 2), after the connection to an AP is established and DHCP process is complete (AP granted the IP), switching to AP mode will result in system failure.
- Notes
  - MT7682S and MT7686 does not support Keil IDE and IAR embedded workbench IDE in this release.
  - [MT2523x, MT2533D] The power consumption of EDR data transmission as master role will be improved in the future release.

9.2. Known issues

There are known issues in this version of the SDK, avoid the following:

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

- [MT2523x, MT2533D] Creating an eSCO HFP and a Bluetooth LE GATT profile simultaneously, might add noise during a phone call when GATT is streaming a large amount of data.
- [MT2523x, MT2533D] Sometimes the HFP voice is‘nt smooth when transmitting BLE data at the same time.

9.3. Migration

Migrate the following module based applications when upgrading the SDK from version 4.3.x to version 4.5.0.

1) API naming changes for the HFP codec driver:

   `<sdk_root>/driver/board/mt25x3_hdk/bt_codec/src/bt_hfp_codec.c` and

   `<sdk_root>/driver/board/mt25x3_hdk/bt_codec_BA/src/bt_hfp_codec.c`.

   - Related examples
     - `bt_headset_23alpha, bt_headset_wifi5931, headset_ref_design, iot_sdk_demo, mp3_local_playback, watch_ref_design` under `mt2523_hdk` folder.
     - `watch_ref_design` under `mt2523_watch` folder.
     - `bt_headset_23alpha, bt_headset_wifi5931` under `mt2523s_headset` folder.
     - `headset_gui_ref_design, headset_ref_design` under `mt2533_evb` folder.
   - How to migrate — Replace `bt_codec_task_main()` with `bsp_bt_codec_task_main()`.

2) API naming changed for `io_def_uart_init()`.

   - Related examples — all example projects under the following folders:
     - <sdk_root>/project/linkit7697_hdk/apps
     - <sdk_root>/project/linkit7697_hdk/hal_examples
     - <sdk_root>/project/linkit7697_hdk/templates
     - <sdk_root>/project/mt7687_hdk/apps
     - <sdk_root>/project/mt7687_hdk/hal_examples
     - <sdk_root>/project/mt7687_hdk/templates
   - How to migrate — Replace `io_def_uart_init()` with `bsp_io_def_uart_init()`.

3) API naming changed for BSP LCD and backlight driver.

   `<sdk_root>/driver/board/mt25x3_hdk/lcd/mt25x3_hdk_lcd.c`
   `<sdk_root>/driver/board/mt25x3_hdk/lcd/mt25x3_hdk_lcd.c`
   `<sdk_root>/driver/board/mt25x3_hdk/backlight/mt25x3_hdk_backlight.c and`
   `<sdk_root>/driver/board/mt25x3_hdk/backlight/mt25x3_hdk_backlight.c`

   - Related examples — `audio_mp3_play, gnss_get_location, display_drawing_image, iot_sdk_demo, sensor_subsys_accelerometer, modem_wifi5931_ref_design, mp3_local_playback, watch_ref_design, gdi_display_helloworld`.
   - How to migrate — Replace the APIs as shown below:

BSP_LCD APIs:

Original API name	Modified API name

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

Original API name	Modified API name
BSP_LCD_Init	bsp_lcd_init
BSP_LCD_SetLayerToDefault	bsp_lcd_set_layer_to_default
BSP_LCD_DisplayOn	bsp_lcd_display_on
BSP_LCD_DisplayOff	bsp_lcd_display_off
BSP_LCD_EnterIdle	bsp_lcd_enter_idle
BSP_LCD_ExitIdle	bsp_lcd_exit_idle
BSP_LCD_UpdateScreen	bsp_lcd_update_screen
BSP_LCD_ClearScreen	bsp_lcd_clear_screen
BSP_LCD_ClearScreenBW	bsp_lcd_clear_screen_bw
BSP_LCD_GetParam	bsp_lcd_get_parameter
BSP_LCD_ConfigROI	bsp_lcd_config_roi
BSP_LCD_ConfigLayer	bsp_lcd_config_layer
BSP_LCD_set_index_color_table	bsp_lcd_set_index_color_table
BSP_LCD_register_callback	bsp_lcd_register_callback

BSP_Backlight APIs:

Original API name	Modified API name
BSP_Backlight_enable	bsp_backlight_enable
BSP_Backlight_init	bsp_backlight_init
BSP_Backlight_deinit	bsp_backlight_deinit
BSP_Backlight_init_isink	bsp_backlight_init_isink
BSP_Backlight_set_clock_source_isink	bsp_backlight_set_clock_source_isink
BSP_Backlight_set_step_current_isink	bsp_backlight_set_step_current_isink
BSP_Backlight_set_double_current_isink	bsp_backlight_set_double_current_isink
BSP_Backlight_init_display_pwm	bsp_backlight_init_display_pwm
BSP_Backlight_set_clock_source_display_pwm	bsp_backlight_set_clock_source_display_pwm
BSP_Backlight_set_duty_display_pwm	bsp_backlight_set_duty_display_pwm
BSP_Backlight_set_width_display_pwm	bsp_backlight_set_width_display_pwm
BSP_Backlight_init_lcm_brightness	bsp_backlight_init_lcm_brightness
BSP_Backlight_deinit_lcm_brightness	bsp_backlight_deinit_lcm_brightness
BSP_Backlight_set_step_lcm_brightness	bsp_backlight_set_step_lcm_brightness

4) To migrate USB task definition and configuration in task_def.h under <sdk_root>/gva/project/mt2523_hdk/<project>/inc

```
#ifdef MTK_USB_DEMO_ENABLED
/* USB */
#define USB_TASK_NAME "USB"
#define USB_TASK_STACKSIZE 4096
#define USB_TASK_PRIO TASK_PRIORITY_HIGH
#define USB_QUEUE_LENGTH  500
#endif
```

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

5) Enable HAL_DWT_MODULE_ENABLED for task stack overflow verification.

- Related examples — all example projects under <sdk_root>/project/mtxxxx_xxx/apps and <sdk_root>/project/mtxxxx_xxx/apps/templates folders.
- How to migrate — add HAL_DWT_MODULE_ENABLED definition in <sdk_root>/project/mtxxxx_xxx/apps/<project>/inc/hal_feature_config.h or <sdk_root>/project/mtxxxx_xxx/templates/<project>/inc/hal_feature_config.h

```
#define HAL_WDT_MODULE_ENABLED
```

And add the configPOST_SLEEP_PROCESSING macro in <sdk_root>/project/mtxxxx_xxx/apps/<project>/inc/FreeRTOSConfig.h or <sdk_root>/project/mtxxxx_xxx/templates/<project>/inc/FreeRTOSConfig.h

```
#if (configUSE_TICKLESS_IDLE >= 1)
#if (configCHECK_FOR_STACK_OVERFLOW > 0)
/* restore dwt stack overflow check after deep sleep */
#undef configPOST_SLEEP_PROCESSING
#define configPOST_SLEEP_PROCESSING(x)
\
{
\
    extern void hal_dwt_init(void);
\
    extern void vPortCurrentTaskStackOverflowCheck(void);            \
    hal_dwt_init();
\
    vPortCurrentTaskStackOverflowCheck();
\
}
#endif /* (configCHECK_FOR_STACK_OVERFLOW > 0) */
#endif /* (configUSE_TICKLESS_IDLE >= 1) */
```

6) Migrate HAL example projects and app projects when an error occurs "kernel/service/src/memory_regions.o not found".

a) All example projects under <sdk_root>/project/mtxxxx_xxx/apps and no_rtos_initialize_system project in <sdk_root>/project/mtxxxx_xxx/apps/templates folder

i. remove C_FILES += kernel/service/src/memory_region.c in project's Makefile

b) Other <sdk_root>/project/mtxxxx_xxx/apps/templates folder (except no_rtos_initialize_system project) and all app project

i. Add iot_sdk_demo project's regions_init.c in the app/src folder.

ii. Add regions_init.c path in Makefile:

a. APP_FILES += $(APP_PATH_SRC)/regions_init.c

c) For KEIL and IAR app project

i. Add regions_init.c in src file when project is opened in KEIL/IAR.

ii. Remove memory_region.c file when project is opened in KEIL/IAR.

d) For KEIL and IAR HAL example project

i. Remove memory_region.c file when project is opened in KEIL/IAR.

7) Move MTK_FW_VERSION information from project makefile to <sdk_root>/middleware/MTK/verno/inc/verno.h.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. ("Airoha") and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

a) Related examples — all projects.

b) How to migrate —migrate the project when an error or warning occurs: "MTK_FW_VERSION" redefined.

   i. Remove the codes in project's Makefile.

```
ifdef MTK_FW_VERSION
CFLAGS += -DMTK_FW_VERSION=\"$(MTK_FW_VERSION)\"
endif
```

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. ("Airoha") and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

10. SDK Version 4.3.1

10.1. Main changes

- Bug fixes
  - [MT25x3] Sliding the touch panel for over 20 minutes might stall the system.
  - [MT25x3] Playing music with A2DP over 30 minutes results in system crash.

10.2. Known issues

There are known issues in this version of the SDK, avoid the following:

- [MT2523x, MT2533D] Creating an eSCO HFP and a Bluetooth LE GATT profile simultaneously, might add noise during a phone call when GATT is streaming a large amount of data.
- [MT2523x, MT2533D] During two EDR and two Bluetooth LE multi-link transmission, the link connections might be unstable.
- [MT7682] In Wi-Fi station mode with special low power state (PS mode is equal to 2), after the connection to an access point (AP) is established and DHCP process is complete (AP granted the IP), switching to AP mode will result in system failure.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

11. SDK Version 4.3.0

11.1. Main changes

- Software features and optimization
  - [MT7682S] Added support for MT7682S chipset, including drivers and board support package. MT7682S is based around a highly integrated chipset containing a microcontroller unit (MCU), a low power 1x1 2.4GHz 11b/g/n single-band Wi-Fi subsystem and a power management unit (PMU). The MCU is an ARM Cortex-M4 processor with floating point unit, integrated with 1MB flash memory. It also provides the following features and capabilities.
    - Soft AP
    - STA
    - WPA/WPA2 Security
    - Smart Connection
- Bug fixes
  - [MT7697x, MT25x3] After disconnecting and re-connecting Bluetooth LE connection, the connection may not be established, if master and slave modes are switched.
  - [MT7697x, MT25x3] When a device is connected with Bluetooth LE data connection with another device and triggers a page scan at the same time, the connection might be lost.
  - [MT2523x, MT2533D] Bluetooth HFP has noise sound in the very beginning 400ms to 500ms right after connection is established.
  - [MT2523x, MT2533D] In the Bluetooth EDR low power mode, the time of voice connection establishment might be up to 2 seconds.
  - [MT2523x, MT2533D] In the state of low power mode without CPU clock tick (tickless mode), Bluetooth EDR link might be lost.
- Notes
  - The name of MT2523 Flash Tool is changed to IoT flash tool supporting 2523x and 768x chipsets.
  - MT7682S does not support Keil IDE and IAR embedded workbench IDE in this release.
  - The deprecated Bluetooth middleware under `<sdk_root>/middleware/MTK/bluetooth_BA` will be removed from this version.

11.2. Known issues

There are known issues in this version of the SDK, avoid the following:

- [MT2523x, MT2533D] Creating an eSCO HFP and a Bluetooth LE GATT profile simultaneously, might add noise during a phone call when GATT is streaming a large amount of data.
- [MT2523x, MT2533D] During two EDR and two Bluetooth LE multi-link transmission, the link connections might be unstable.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

- [MT7682] In Wi-Fi station mode with special low power state (PS mode is equal to 2), after the connection to an access point (AP) is established and DHCP process is complete (AP granted the IP), switching to AP mode will result in system failure.

11.3. Migration

Migrate the following module based applications when upgrading the SDK from version 4.2.x to version 4.3.0.

1) Modify the SDK folder structure for more coherent view in SDK v4.3.0.
   a) Apply the modifications for folder structure through a script located at `<V430_codebase_root>/tools/scripts/migration/update_420proj_to_430.pl`. Please run the script under Linux environment to migrate the folders and files listed in Table 3.
   b) Copy `ffconf.h` from `<sdk_roo>/project/mt2523_hdk/apps/atci_register_command/inc/ffconf.h` in v4.3.0 to `inc` folder of the project in v4.2.x and modify the path settings.
   c) Usage:
      i. Copy example projects in v4.2.x to a new path.
      ii. `cd <V430_codebase_root>/tools/scripts/migration`
      iii. `perl update_420proj_to_430.pl <your_example_project_path>`
   d) Limitations
      iv. Must install PERL v5.18.4 or higher.
      v. Must run on Ubuntu 14.01.1 or higher

Table 3. Folders and files to migrate

Folder structure in SDK v4.2.x	Folder structure in SDK v4.3.0
driver/board/mt76x7_hdk/wifi	middleware/MTK/wifi_service
driver/chip/mt2523/lib	prebuilt/driver/chip/mt2523/lib
driver/chip/mt7687/lib	prebuilt/driver/chip/mt7687/lib
middleware/MTK/ble_ancs	prebuilt/middleware/MTK/ble_ancs
middleware/MTK/ble_notify/lib	prebuilt/middleware/MTK/ble_notify/lib
middleware/MTK/homekit	prebuilt/middleware/MTK/homekit
middleware/MTK/minicli/lib	prebuilt/middleware/MTK/minicli/lib
middleware/MTK/minisupp	prebuilt/middleware/MTK/minisupp
middleware/MTK/minorsupc	prebuilt/middleware/MTK/minorsupc
middleware/MTK/nvdm/lib	prebuilt/middleware/MTK/nvdm/lib
middleware/MTK/slp	prebuilt/middleware/MTK/slp
middleware/MTK/sensor_subsys/fusion_algo	prebuilt/middleware/MTK/sensor_subsys/fusion_algo
driver/board/component/audio	prebuilt/driver/board/component/audio
driver/board/component/bt_codec	prebuilt/driver/board/component/bt_codec
middleware/MTK/audio/mp3_codec/lib	prebuilt/middleware/MTK/audio/mp3_codec/lib
middleware/MTK/audio/amr_codec/lib	prebuilt/middleware/MTK/audio/amr_codec

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

Folder structure in SDK v4.2.x	Folder structure in SDK v4.3.0
	/lib
middleware/MTK/battery_management/port/mt2523/lib/fuelgauge	prebuilt/middleware/MTK/fuelgauge
middleware/MTK/bluetooth/lib	prebuilt/middleware/MTK/bluetooth/lib
middleware/third_party/dhcpd	middleware/MTK/dhcpd
kernel/service open source	

2) To migrate Wi-Fi related applications from MT76x7 to MT7682 and from SDK 4.2.x to SDK 4.3.0, please refer to the Wi-Fi migration guide under <sdk_root>/doc folder.

3) XML at <sdk_root>/middleware/third_party/xml.

   - XML middleware folder includes a header file config.h. To eliminate the ambiguity, this file is moved from "middleware/third_party/xml/inc/config.h" to "middleware/third_party/xml/inc/xml/config.h".
   - Applications that include "config.h" in the XML middleware should be replaced with "xml/config.h".

4) Bootloader function names are changed only for MT2523 HDK at <sdk_root>/driver/board/mt25x3_hdk/bootloader/core/src/bl_main.c

   - Replace the function name "bl_main" with "main" in <sdk_root>/project/mt2523_hdk/apps/bootloader/GCC/startup_bootloader.s.
   - Then replace the function name "bl_main" with "__main" in for Keil build at <sdk_root>/project/mt2523_hdk/apps/bootloader/MDK-ARM/startup_bootloader.s.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

12. SDK Version 4.2.2

12.1. Main changes

- Bug fixes
  - [MT2523] Fixed the issue that the charging current might be up to 500mA for around 500ms after waking up from sleep mode.
  - [MT2523] Improved the transmission power and receiving sensitivity for Bluetooth Enhanced Data Rate (EDR) and with Low Energy (LE).
  - [MT2523] Fixed the issue that software initialization fails on µVision® IDE, due to improper initialization of variables with 0s.
  - [MT2523, MT2511] Fixed the issue that heart rate measurement might be inaccurate for a short period after a prolonged use (6 to 8 hours).
  - [MT2523] It's rare to receive incorrect data, if the host, such as PC, transfers over 2KB of data to the device through USB.

12.2. Known issues

There are known issues in this version of the SDK, avoid the following:

- [MT2523x, MT2533D] Creating an eSCO HFP and a Bluetooth LE GATT profile simultaneously might add noise during a phone call when GATT is streaming a large amount of data.
- [MT2523x, MT2533D] During two EDR and two Bluetooth LE multi-link transmission, the link connections might be unstable.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

13. SDK Version 4.2.1

13.1. Main changes

- Bug fixes
  - [MT76x7] Fixed the issue that in-target-reset (reset Cortex-M4 only) fails in debugging mode.
  - [`modem_wifi5931_ref_design`] Fixed the issue of failing in Wi-Fi initialization.
  - [`bioband_ref_design`] Improve the blood pressure range of Systolic blood pressure (SBP) value.

13.2. Known issues

There are known issues in this version of the SDK, avoid the following:

- [MT2523x, MT2533D] Creating an eSCO HFP and a Bluetooth LE GATT profile simultaneously might add noise during a phone call when GATT is streaming a large amount of data.
- [MT2523x, MT2533D] During two EDR and two Bluetooth LE multi-link transmission, the link connections might be unstable.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

14. SDK Version 4.2.0

14.1. Main changes

- Software features and optimization
  - Added support for MT2533D chipset, including drivers and board support package.
  - [Watch reference design] Enabled Bluetooth LE Find Me profile (FMP) on a watch reference design (for licensed customers only).
  - [Watch reference design] Enabled sport application related features, such as heart rate, GNSS and pedometer on watch reference design (for licensed customers only).
  - [MT76x7] Enhanced the Flash Tool to support Ubuntu Linux.
  - [MT2523x, MT2533D] Added the console mode for Flash Tool in Ubuntu Linux and Windows.
  - Added Easy PinMux Tool support in Ubuntu Linux.
  - [MT7697x] Added the support for Wi-Fi, Bluetooth LE and Wi-Fi Bridge in `iot_sdk_demo` example application.
  - [MT76x7] Added the RSSI sorting to match the SSID in scan table.
- Bug fixes
  - [MT2523x, MT2533D] Fixed the system crash issue while establishing A2DP connection with certain Bluetooth dongle and PCs.
  - [MT2523x, MT2533D] Fixed the system timer (SysTick) drifting when the clock (CLK) is at 104 MHz and the clock source is High Frequency Oscillator (HFOSC). SysTick is used by FreeRTOS as a basic timer
  - [Watch reference design] Using any undefined character in a drawing function of watch reference UI will cause a system error.
  - [MT76x7] Fixed the issue that occurs when the power consumption value is higher than the one defined in the specification in DTIM10 mode.
- DTIM 10 mode: DTIM interval = 10, Wi-Fi wakes up every 10 beacon period (1000ms).
- [MT76x7] Fixed the issue of drifting system tick during frequent interrupts in idle mode.
- [MT76x7] Fixed the uneven throughput between transmit and receive operations, the transmitting throughput was much higher than the receiving throughput, when the throughput was set at maximum for both operations at the same time.

14.2. Known issues

There are known issues in this version of the SDK, avoid the following:

- [MT2523x, MT2533D] Creating an eSCO HFP and a Bluetooth LE GATT profile simultaneously might add noise during a phone call when GATT is streaming a large amount of data.
- [MT2523x, MT2533D] During two EDR and two Bluetooth LE multi-link transmissions, the link connections might be unstable.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

14.3. Migration

Migrate the following module based applications when upgrading the SDK from version 4.1.0 to version 4.2.0 and higher.

1) Bluetooth notification services at `<sdk_root>/middleware/MTK/bt_notify`.

- Related structures — the related structures and enumerations are listed in Table 4.
- Related examples — `bt_android_notification`, `fota_download_manager` and `gnss_get_location` under `mt2523_hdk` folder, `watch_demo` under `mt2523_watch` folder.
- How to migrate — replace the structures and enumerations of SDK v4.1.0 with the corresponding ones in SDK v4.2.0, as described in Table 4.

Table 4. Structures and enumerations migration for Bluetooth notification services

Structures and enumeration in SDK v4.1.0 (struct or enum name. field name)	Structures and enumeration in SDK v4.2.0 (struct or enum name. field name)
bt_notify_indication_event_t. BT_NOTIFY_EVENT_NEW_MSG	bt_notify_event_t. BT_NOTIFY_EVENT_NONE
bt_notify_indication_event_t. BT_NOTIFY_EVENT_DATA	bt_notify_event_t. BT_NOTIFY_EVENT_DATA_RECEIVED
bt_noti_data_t. err_code	bt_notify_event_data_t. error_code
bt_noti_data_t. len	bt_notify_event_data_t. length
bt_notify_remote_system_t	bt_notify_remote_system_type_t
bt_notify_notification_action_t. BT_NOTIFY_NOTIFICATION_NEW	bt_notify_action_type_t. BT_NOTIFY_ACTION_TYPE_NEW
bt_notify_notification_action_t. BT_NOTIFY_NOTIFICATION_DELETE	bt_notify_action_type_t. BT_NOTIFY_ACTION_TYPE_DELETE
bt_notify_page_content_t	bt_notify_page_content_list_t
bt_notify_noti_t. noti_action	bt_notify_notification_t. action_content
bt_notify_sms_t. send_number	bt_notify_sms_t. sender_number
bt_notify_call_t. send_number	bt_notify_missed_call_t. sender_number
bt_notify_remote_system_t	bt_notify_remote_system_type_t
bt_notify_callback_t. noti_data	bt_notify_callback_data_t . event_data
bt_notify_result_t. BT_NOTIFY_NO_SUPPORT	bt_notify_result_t. BT_NOTIFY_RESULT_NOT_SUPPORTED
bt_notify_result_t. BT_NOTFIY_REGISTER_RET_INVALID_PARAMETER	bt_notify_result_t. BT_NOTFIY_RESULT_REGISTER_INVALID_PARAMETER
bt_notify_result_t. BT_NOTIFY_REGISTER_RET_SYSTEM_REGISTE	bt_notify_result_t. BT_NOTIFY_RESULT_REGISTER_SYSTEM_ALREAD

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

Structures and enumeration in SDK v4.1.0 (struct or enum name. field name)	Structures and enumeration in SDK v4.2.0 (struct or enum name. field name)
RED	Y_REGISTERED
bt_notify_result_t. BT_NOTIFY_REGISTER_RET_REPEAT_REGISTER	bt_notify_result_t. BT_NOTIFY_RESULT_REGISTER_REPEATED_REGISTRATION
bt_notify_result_t. BT_NOTIFY_REGISTER_RET_NOT_IMPLEMENT	bt_notify_result_t. BT_NOTIFY_RESULT_REGISTER_NOT_IMPLEMENT
bt_notify_result_t. BT_NOTIFY_RET_INVALID_PARAMETER	bt_notify_result_t. BT_NOTFIY_ RESULT_INVALID_PARAMETER
bt_notify_result_t. BT_NOTIFY_ RET_PARSE_PARA_ERROR	bt_notify_result_t. BT_NOTIFY_ RESULT_PARSING_ ERROR
bt_notify_result_t. BT_NOTIFY_RET_NO_CHANNAL	bt_notify_result_t. BT_NOTIFY_RESULT_CHANNAL_UNAVAILABLE
bt_notify_result_t. BT_NOTIFY_RET_FAIL	bt_notify_result_t. BT_NOTIFY_RESULT_FAILED
bt_notify_result_t. BT_NOTIFY_REGISTER_RET_OK	bt_notify_result_t. BT_NOTIFY_RESULT_REGISTER_OK
bt_notify_data_source_t. BT_NOTIFY_DATA_SOURCE_UNKNOW	bt_notify_data_source_t. BT_NOTIFY_DATA_SOURCE_INVALID

5) Bluetooth callback manager service at `<sdk_root>/middleware/MTK/bt_callback_manager`.

- Related structures — the related structures and enumerations are listed in Table 5.
- Related examples — `bt_android_notification`, `fata_download_manager` and `gnss_get_location` under `mt2523_hdk` folder.
- How to migrate — replace the structures and enumerations of SDK v4.1.0 with the corresponding ones in SDK v4.2.0, as described in Table 5.

Table 5. Structures and enumerations migration for Bluetooth callback manager services

Structure and enumeration in SDK 4.1.0 (struct or enum name. field name)	Structure and enumeration in SDK 4.2.0 (struct or enum name. field name)
bt_callback_type_t. bt_callback_type_gatts_get_excute_write_result	bt_callback_type_t. bt_callback_type_gatts_get_execute_write_result
Implement the function `bt_sdps_get_customized_record()` and return the record array.	`Bt_status_t bt_callback_manager_add_sdp_customized_record ( const bt_sdps_record_t * record);` Call this function to add your own record.

6) Bluetooth LE Apple Notification Center Service (ANCS) at `<sdk_root>/middleware/MTK/ble_ancs`.

- Related differences — `ble_ancs_gprot.h` in SDK v4.1.0 and `ble_ancs.h` in SDK v4.2.0. The related structures, enumerations and APIs are listed in Table 6.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

- Related examples — `ble_ancs_ios_notification` under the `mt2523_hdk` folder and `watch_demo` under the `mt2523_watch` folder of SDK v4.1.0.
- How to migrate — replace the header file name, structures, enumerations and APIs as described in Table 6.

Table 6. Structures and enumerations migration for Bluetooth LE ANCS

Structure , enumeration and API in SDK 4.1.0 (struct or enum name. field name)	Structure, enumeration and API in SDK 4.2.0 (struct or enum name. field name)
`ANCS_EVENT_ID_NUM`	`BLE_ANCS_MAX_EVENT_ID_NUMBER`
`ANCS_CATEGORY_ID_NUM`	`BLE_ANCS_MAX_CATEGORY_ID_NUMBER`
`ANCS_NOTIFICATION_ATTR_NUM`	`BLE_ANCS_MAX_NOTIFICATION_ATTR_NUMBER`
`BLE_ANCS_CHAR_NUM`	`BLE_ANCS_MAX_CHARC_NUMBER`

`ble_ancs_uuid_type_t.`
`BLE_ANCS_PRIMARY_SERVICE_UUID`
`BLE_ANCS_NOTIFICATION_SOURCE_UUID`
`BLE_ANCS_CONTROL_POINT_UUID`
`BLE_ANCS_DATA_SOURCE_UUID`
	`ble_ancs_uuid_t .`
	`BLE_ANCS_UUID_PRIMARY_SERVICE`
	`BLE_ANCS_UUID_NOTIFICATION_SOURCE`
	`BLE_ANCS_UUID_CONTROL_POINT`
	`BLE_ANCS_UUID_DATA_SOURCE`

`ancs_msg_type_t.`
`BT_ANCS_GAP_LE_DISCONNECT_IND`
`BT_ANCS_GATTC_WRITE_CHARC`
	`bt_msg_type_t.`
	`BT_GAP_LE_DISCONNECT_IND`
	`BT_GATTC_WRITE_CHARC`

`bt_status_t ble_ancs_event_callback(`
`ancs_msg_type_t msg, bt_status_t status,`
`void *buff);`
	`bt_status_t ble_ancs_event_callback(`
	`bt_msg_type_t msg, bt_status_t status,`
	`void *buffer);`

`bt_status_t`
`ble_ancs_parse_notification_source(`
`ble_ancs_event_notification_t`
`*notif_source,`
`const uint16_t length, uint8_t *data);`
	`bt_status_t`
	`ble_ancs_parse_notification(`
	`ble_ancs_event_notification_t`
	`*notification,`
	`uint16_t length, uint8_t *data);`

7) Bluetooth sink service at `<sdk_root>/middleware/MTK/bt_sink`.
   - Related structures — `bt_sink_srv_am_files_format_t.path` of SDK v4.1.0 and `bt_sink_srv_am_files_format_t.file_type` of SDK v4.2.0.
   - Related examples — `mp3_local_playback` under the `mt2523_hdk` folder.
   - How to migrate — replace the `bt_sink_srv_am_files_format_t.path` in `mp3_local_playback` with `bt_sink_srv_am_files_format_t.file_type`.
8) Memory stick and SD memory card controller configuration file (`msdc_custom_config.h`).
   - Symptom: An error occurs "`Build... hal_msdc.o FAIL`" when building the projects.
   - Related examples — all example projects under `mt2523_hdk` folder.
   - How to migrate — add `<sdk_root>/project/mt2523_hdk/apps/atci_register_command/inc/msdc_custom_config.h` in SDK v4.2.0 to the `inc` folder of the target application of SDK v4.1.0.
9) Common folder at `<sdk_root>/project/common`.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. ("Airoha") and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

- Related examples — all projects using Command Line Interface (CLI) commands.
- How to migrate — use the `<sdk_root>/project/common` folder of SDK v4.2.0, do not replace the folder with SDK v4.1.0.

10) Port service at `<sdk_root>/middleware/MTK/port_service`.

- Symptom — An error occurs "`multiple definition of` `` `log_control_block_atci_serialport'` " when building the projects.
- Related examples — `bt_headset_23alpha`, `iot_sdk_demo`, `iot_sdk_dev`, `low_power_without_psram`, `modem_wifi5931_ref_design`, `mp3_local_playback` under `mt2523_hdk`, `bt_headset_23alpha`, `bt_headset_wifi5931` under `mt2523s_headset` and `headset_gui_ref_design`, `headset_ref_design` under `mt2533_evb` folders.
- How to migrate — remove the `at_command_serial_port.c` file from the `src` folder of a project and modify the related Makefiles and configure files for Keil IDE and IAR embedded workbench IDE.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. ("Airoha") and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

15. SDK Version 4.1.0

15.1. Main changes

Software features and optimization

- [MT2523x] Added a reference design for watch supporting heart rate, GNSS and Bluetooth notification. (`<sdk_root>/project/mt2523_hdk/apps/watch_ref_design`)
- [MT2523x] Added the support to update the firmware of LinkIt 2523 HDK through GATT profile using Smart Device, an Android app on a hand-held device.
- [MT2523x] Added the support for USB mass storage protocol (reference application at `<sdk_root>/project/project/mt2523_watch/apps/watch_demo`).
- [MT2523x] Added the support for MP3 audio playback from an SD card.
- [MT7697x] Added support to Wi-Fi and Bluetooth LE coexistence. Application can control all the links of the two protocols without any interference.

Bug fixes

- [MT2523x] Fixed the issue that the A2DP connection between MT2523x-based device (Client) and certain dedicated device (Server) gets disconnected when the audio transfer is paused, waits for a while, and resumes the playback again.
- [MT2523x] Fixed the issue that LinkIt 2523 HDK cannot establish Bluetooth audio connection (A2DP) with an iPhone with iOS 10 or later versions of OS.
- [MT2523x] Fixed the issue in multiple links where one link (eSCO link) is on call, the other idle link (ACL link) gets disconnected.
- [MT2523x, MT7697x] Fixed the issue in multiple links when master and slave roles coexist at the same time, one of the links might be disconnected.
- [MT76x7] Fixed the issue that the system hangs when an application listens to the Wi-Fi traffic in sniffer mode.

Notes

- This version introduces a new method to add a module. To handle the migration of your application to this SDK, see section 6.2, “Adding a module to the build flow of the project” in `<sdk_root>/doc/Airoha_IoT_SDK_v4_GCC_Build_Environment_Guide.pdf`.

15.2. Known issues

There are known issues in this version of the SDK; developer needs to avoid the following scenarios.

- [MT2523x] Using any undefined character in a drawing function of watch reference UI will cause a system error.
- [MT2523x] Creating an eSCO HFP and a Bluetooth LE GATT profile simultaneously might add noise during a phone call when GATT is streaming a large amount of data.
- [MT2523x] In two BDR and two Bluetooth LE multi-link transmission, the links might be unstable.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

16. SDK Version 4.0.0

16.1. Main changes

Software features and optimization

- The SDK includes a new Bluetooth stack. The Bluetooth stack prior to v4.0.0 will be deprecated in SDK 4.0 and removed after 2016. The new stack has the following characteristics compared to the prior versions of the API.
- The footprint is reduced significantly with the same profile support: GAP, HFP (HF), A2DP (SINK), AVRCP (CT), SPP (Server and Client), PBAP (Client), GATT and SM.
  - Current footprint: 70kB ROM and 20kB RAM.
  - Prior footprint: 330kB ROM and 130kB RAM.
- RAM is configurable and memory usage could be optimized to fit the application requirements according to the section “Memory management” of “Airoha IoT SDK for RTOS Bluetooth Developer's Guide” in the `<sdk_root>/doc` folder.
- The Bluetooth API in SDK v4.0.0 is not backward compatible with the prior versions of the API.
- [MT2523x] LCM driver and iot_sdk_demo demo project based on LinkIt 2523 HDK by SAC support 320 x 320 pixel resolutions.
- Merged RTOS tasks to save resources. Refined the task priority and collected the task configurations into one header file for clarity and better maintainability.
- [MT2523x] Supports logging from USB (USB2 COM port) in the iot_sdk_demo project on the LinkIt 2523 HDK by SAC.
- [MT2523x] Supports MP3 audio file local playback.

Bug fixes

- [MT76x7] Fixed the connection failure issue in security establishment phase when connecting to a legacy AP (802.11a, 802.11g, or 802.11b).

Notes

- Please format the whole flash with the MT76x7 flash tool when flashing the SDK 4.0.0 binary to LinkIt 7687 and 7697 HDK for the first time, as described in the “Formatting the storage” section of MT76x7 Flash Tool Users Guide in the root folder of MT76x7 Flash tool.

16.2. Known issues

There is one known issue when using the SDK; developer needs to avoid the following scenario.

- [MT76x7] The peak throughput may drop from 1 to 5Mbps when connecting to a 40Mhz bandwidth (HT40) AP in repeater mode.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

17. SDK Version 3.3.2

17.1. Main changes

Bug fixes for LinkIt 76x7 HDK

- Fixed the Wi-Fi throughput drop when the external interrupt has not been received for more than 30 seconds.
- Fixed the Wi-Fi throughput drop in the mid-range signal strength (the RSSI is between -60 to -80dbm), improved the connection stability and ping in long-range (the RSSI is less than -80dbm).

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

18. SDK Version 3.3.0

18.1. Main changes

- Software features and optimization
  - [MT2523x] Support LCM with DBI and DSI interfaces in the same firmware. Provide an auto-detection mechanism to select RM69032 (DSI) or ST7789H2 (DBI) LCM.
  - [MT2523x] Support 2D graphics drawing with HAL G2D API.
  - [MT7697x] Support setting Bluetooth radio transmission power.
  - [MT7697D] Support Wi-Fi 5G AP/STA (excluding DFS).
  - [MT76x7] Support Wi-Fi repeater mode.
  - [MT76x7] Provide easy-to-use Wi-Fi initialization API to address the requirement of reading implicit configurations from NVDM.
  - [MT76x7] Support auto-detection of AP's authentication mode and encryption type.
- Tool features and optimization
  - [MT2523G] Enhance the upgrade speed of GNSS firmware on MT2523 flash tool
- Bug fixes
  - [MT76x7] Reduced the time to connect to an access point, where two Wi-Fi access points exist with the same SSID within the Wi-Fi RF visible range.

18.2. Known issues

There is one known issue when using the SDK; developer needs to avoid the scenario listed as below.

- [MT2523x] In multi-link and one of them is on call (eSCO link), the idle link (ACL link) might be disconnected.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

19. SDK Version 3.2.0

19.1. Main changes

- Features and optimization
  - New BLE stack is available for MT7697 with small footprint and support Bluetooth 4.2.
  - Add the capabilities of configuring and retrieving the CPU frequency with DVFS APIs.
  - Support SPI slave on MT76x7 with new HAL SPI APIs.
  - Reduce the interrupt latency in flash read or write operation.
  - IAR tool chain support, pre-integrate HAL, FreeRTOS, Bluetooth, FOTA, GNSS example projects and the IoT demonstration project with IAR IDE tool.
- Bug fixes
  - The `MacAddr`, `IpAddr`, `IpGateWay`, and `IpNetmask` attributes in access point (AP) profile is not workable and those values must be stored in station (STA) profile for both AP and STA modes.
  - The Wi-Fi driver will obtain Wi-Fi MAC address from NVRAM, but not eFuse of the MT76x7.
  - Every event passed to `wifi_connection_register_event_notifier()` can only register one corresponding function handler.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

20. SDK Version 3.1.0

20.1. Main changes

- SDK supports LinkIt 2523 HDK, including peripheral drivers, middleware and demonstration applications.

20.2. Known issues

There are some known issues when using the SDK; developer needs to avoid these scenarios listed as below.

- The `MacAddr`, `IpAddr`, `IpGateWay`, and `IpNetmask` attributes in access point (AP) profile is not workable and those values must be stored in station (STA) profile for both AP and STA modes.
- The Wi-Fi driver will obtain Wi-Fi MAC address from NVRAM, but not eFuse of the MT76x7.
- Every event passed to `wifi_connection_register_event_notifier()` can only register one corresponding function handler.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.

21. SDK Version 3.0.0

21.1. Known issues

There are some known issues when using the SDK; developer needs to avoid these scenarios listed as below.

- The `MacAddr`, `IpAddr`, `IpGateWay` and `IpNetmask` attributes in access point (AP) profile are not workable and those values must be stored in station (STA) profile for both AP and STA modes.
- The Wi-Fi driver will obtain the Wi-Fi MAC address from NVRAM, but not from eFuse for MT76x7.
- Every event passed to `wifi_connection_register_event_notifier` can only register one corresponding function handler.

© 2015 - 2019 Airoha Technology Corp.

This document contains information that is proprietary to Airoha Technology Corp. (“Airoha”) and/or its licensor(s).
Any unauthorized use, reproduction or disclosure of this document in whole or in part is strictly prohibited.