

WS63V100 BOOT API

## 开发参考

文档版本 01

发布日期 2024-04-10

# 前言

## 概述

本文主要介绍 WS63 Flashboot 中升级相关的 API 接口，具体参考《WS63V100 FOTA 开发指南》中的接口介绍。

## 产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本
WS63	V100

## 读者对象





本文档主要适用于以下工程师：

- 技术支持工程师
- 软件开发工程师

## 符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
----	----

符号	说明
 危险	表示如不避免则将会导致死亡或严重伤害的具有高等级风险的危害。
 警告	表示如不避免则可能导致死亡或严重伤害的具有中等级风险的危害。
 注意	表示如不避免则可能导致轻微或中度伤害的具有低等级风险的危害。
须知	用于传递设备或环境安全警示信息。如不避免则可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “须知”不涉及人身伤害。
 说明	对正文中重点信息的补充说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

修改记录

文档版本	发布日期	修改说明
01	2024-04-10	第一次正式版本发布。
00B01	2024-02-22	第一次临时版本发布。

目 录

前言 .....i

1 接口介绍.....1

# 1 接口介绍

表1-1 升级接口（升级包存储部分）描述

接口名称	描述
uapi_upg_init	升级模块初始化。
uapi_upg_prepare	保存升级包到本地存储器前的准备工作。
uapi_upg_write_package_async/uapi_upg_write_package_sync	将升级包数据写入本地存储器。（异步方式/同步方式）
uapi_upg_read_package	从本地存储器读取升级包数据。
uapi_upg_request_upgrade	申请开始进行本地升级，所有升级包数据全部保存完成后，调用此接口。
uapi_upg_get_storage_size	获取可存放升级包的空间大小。

表1-2 升级接口（本地升级部分）描述

接口名称	描述
uapi_upg_init	升级模块初始化。
uapi_upg_register_progress_callback	注册升级进度通知回调函数，注册后，在本地升级过程中会调用回调函数通知当前进度。
uapi_upg_start	开始本地升级。
uapi_upg_get_result	获取升级结果。

接口名称	描述
uapi_upg_verify_file_head	校验升级包头结构。
uapi_upg_verify_file_image	校验升级包中的升级镜像。
uapi_upg_verify_file	校验整个升级包。
uapi_upg_register_user_defined_verify_func	注册用户自定义字段的校验函数。  升级包结构中预留了 48Byte 用于用户自定义数据的校验。注册自定义校验函数后，被注册的函数会在调用 uapi_upg_verify_file_head 和 uapi_upg_verify_file 函数时被调用到。如果自定义数据校验失败 uapi_upg_verify_file_head 和 uapi_upg_verify_file 会返回失败。

表1-3 升级接口入参及返回值描述

接口原型	参数及返回值说明
errcode_t uapi_upg_init(const upg_func_t *func_list)	<ul style="list-style-type: none"><li>入参说明： func_list: 注册回调列表，upg_func_t 类型。</li><li>返回值：<ul style="list-style-type: none"><li>- ERRCODE_SUCC: 成功。</li><li>- 其他: 失败。</li></ul></li></ul>
errcode_t uapi_upg_start(void)	<ul style="list-style-type: none"><li>入参说明: 无。</li><li>返回值：<ul style="list-style-type: none"><li>- ERRCODE_SUCC: 成功。</li><li>- 其他: 失败。</li></ul></li></ul>
errcode_t uapi_upg_register_progress_callback(uapi_upg_progress_cb func)	<ul style="list-style-type: none"><li>入参说明： func: 回调函数，该函数需业务实现。</li><li>返回值：<ul style="list-style-type: none"><li>- ERRCODE_SUCC: 成功。</li></ul></li></ul>

接口原型	参数及返回值说明
	<ul style="list-style-type: none"><li>- 其他：失败。</li></ul>
<code>errcode_t uapi_upg_get_result(upg_result_t *result, uint32_t *last_image_index)</code>	<ul style="list-style-type: none"><li>• 入参说明：<ul style="list-style-type: none"><li>- result：出参，保存升级结果的内存地址，类型 <code>upg_result_t</code>。</li><li>- last_image_index：出参，保存最后一个处理的镜像的索引。</li></ul></li><li>• 返回值：<ul style="list-style-type: none"><li>- <code>ERRCODE_SUCC</code>：成功。</li><li>- 其他：失败。</li></ul></li></ul>
<code>errcode_t uapi_upg_prepare(upg_prepare_info_t *prepare_info)</code>	<ul style="list-style-type: none"><li>• 入参说明： prepare_info：入参， upg_prepare_info_t*类型，准备信息的指针。</li><li>• 返回值：<ul style="list-style-type: none"><li>- <code>ERRCODE_SUCC</code>：成功。</li><li>- 其他：失败。</li></ul></li></ul>
<code>errcode_t uapi_upg_write_package_async(uint32_t offset, const uint8_t *buff, uint16_t len, uapi_upg_write_done_cb callback)</code>	<ul style="list-style-type: none"><li>• 入参说明：<ul style="list-style-type: none"><li>- offset：入参，uint32_t 类型，相对升级包开头的偏移。</li><li>- buff：入参，const uint8_t *类型，存放升级包数据的 buffer。</li><li>- len：入参，uint16_t 类型，升级包数据 buffer 的长度。</li><li>- callback：入参， uapi_upg_write_done_cb 类型，写入完成的回调函数。</li></ul></li><li>• 返回值：<ul style="list-style-type: none"><li>- <code>ERRCODE_SUCC</code>：成功。</li><li>- 其他：失败。</li></ul></li></ul>
<code>errcode_t</code>	<ul style="list-style-type: none"><li>• 入参说明：</li></ul>

接口原型	参数及返回值说明
<code>uapi_upg_write_package_sync(uint32_t offset, const uint8_t *buff, uint16_t len)</code>	<ul style="list-style-type: none"><li>- <code>offset</code>: 入参, <code>uint32_t</code> 类型, 相对升级包开头的偏移。</li><li>- <code>buff</code>: 入参, <code>const uint8_t *</code>类型, 存放升级包数据的 <code>buffer</code>。</li><li>- <code>len</code>: 入参, <code>uint16_t</code> 类型, 升级包数据 <code>buffer</code> 的长度。</li><li>• 返回值:<ul style="list-style-type: none"><li>- <code>ERRCODE_SUCC</code>: 成功。</li><li>- 其他: 失败。</li></ul></li></ul>
<code>errcode_t</code> <code>uapi_upg_read_package(uint32_t offset, uint8_t *buff, uint32_t len)</code>	<ul style="list-style-type: none"><li>• 入参说明:<ul style="list-style-type: none"><li>- <code>offset</code>: 入参, <code>uint32_t</code> 类型, 相对升级包开头的偏移。</li><li>- <code>buff</code>: 出参, <code>uint8_t *</code>类型, 存放升级包数据的 <code>buffer</code>。</li><li>- <code>len</code>: 入参, <code>uint32_t</code> 类型, 读取数据 <code>buffer</code> 的长度。</li></ul></li><li>• 返回值:<ul style="list-style-type: none"><li>- <code>ERRCODE_SUCC</code>: 成功。</li><li>- 其他: 失败。</li></ul></li></ul>
<code>uint32_t</code> <code>uapi_upg_get_storage_size(void)</code>	<ul style="list-style-type: none"><li>• 入参说明: 无。</li><li>• 返回值:<ul style="list-style-type: none"><li>- 0: 失败返回 0。</li><li>- 其他: 成功返回空间大小。</li></ul></li></ul>
<code>errcode_t</code> <code>uapi_upg_request_upgrade(bool reset)</code>	<ul style="list-style-type: none"><li>• 入参说明: <code>reset</code>: 入参, <code>bool</code> 类型, 申请流程结束后是否重启系统。</li><li>• 返回值:<ul style="list-style-type: none"><li>- <code>ERRCODE_SUCC</code>: 成功。</li><li>- 其他: 失败。</li></ul></li></ul>



接口原型	参数及返回值说明
<code>errcode_t uapi_upg_verify_file_head(const upg_package_header_t*pkg_header)</code>	<ul style="list-style-type: none"><li>入参说明： pkg_header：入参， upg_package_header_t*类型，指向 升级包头结构的指针。</li><li>返回值：<ul style="list-style-type: none"><li>ERRCODE_SUCC：成功。</li><li>其他：失败。</li></ul></li></ul>
<code>errcode_t uapi_upg_verify_file_image(const upg_image_header_t*img_header, const uint8_t*hash, uint32_t hash_len, bool verify_old)</code>	<ul style="list-style-type: none"><li>入参说明：<ul style="list-style-type: none"><li>img_header：入参， upg_image_header_t*类型，指向 升级包中升级镜像头结构的指针。</li><li>hash：入参，uint8_t*类型，升级 镜像的 HASH 值。</li><li>hash_len：入参，uint32_t 类型， HASH 的长度（单位：Byte）。</li><li>verify_old：入参，bool 类型，是 否校验旧镜像。</li></ul></li><li>返回值：<ul style="list-style-type: none"><li>ERRCODE_SUCC：成功。</li><li>其他：失败。</li></ul></li></ul>
<code>errcode_t uapi_upg_verify_file(const upg_package_header_t*pkg_header)</code>	<ul style="list-style-type: none"><li>入参说明： pkg_header：入参， upg_package_header_t*类型，指向 升级包头结构的指针。</li><li>返回值：<ul style="list-style-type: none"><li>ERRCODE_SUCC：成功。</li><li>其他：失败。</li></ul></li></ul>
<code>void uapi_upg_register_user_defined_verify_ func(uapi_upg_user_defined_check func, uintptr_t param)</code>	<ul style="list-style-type: none"><li>入参说明：<ul style="list-style-type: none"><li>func：入参， upg_package_header_t*类型，用</li></ul></li></ul>

接口原型	参数及返回值说明
	<p>于校验用户自定义字段的校验函数。</p> <ul style="list-style-type: none"><li>- param: 入参, uintptr_t 类型, 注册参数。</li><li>• 返回值: 无。</li></ul>