GR5405 Software Development Kit Release Note

Version: 1.1.8

Release Date: 2025-04-24

Shenzhen Goodix Technology Co., Ltd.

Trademarks and Permissions

GOODiX and other Goodix trademarks are trademarks of Shenzhen Goodix Technology Co., Ltd. All other trademarks and trade names mentioned in this document are the property of their respective holders.

Disclaimer

Information contained in this document is intended for your convenience only and is subject to change without prior notice. It is your responsibility to ensure its application complies with technical specifications.

Shenzhen Goodix Technology Co., Ltd. (hereafter referred to as "Goodix") makes no representation or guarantee for this information, express or implied, oral or written, statutory or otherwise, including but not limited to representation or guarantee for its application, quality, performance, merchantability or fitness for a particular purpose. Goodix shall assume no responsibility for this information and relevant consequences arising out of the use of such information.

Without written consent of Goodix, it is prohibited to use Goodix products as critical components in any life support system. Under the protection of Goodix intellectual property rights, no license may be transferred implicitly or by any other means.

Shenzhen Goodix Technology Co., Ltd.

Headquarters: 26F, Goodix Headquarters, No.1 Meikang Road, Futian District, Shenzhen, China

TEL: +86-755-33338828 Zip Code: 518000

Website: www.goodix.com

Contents

1 SDK V1.1.8

The GR5405 Software Development Kit (SDK) V1.1.8 is the second release for GR5405 System-on-Chips (SoCs).

1.1 Release Overview

1.1.1 Release Package

Table 1-1 Release package

Folder	Description
build	Link-related tools and scripts
components	Bluetooth LE API header, library, and source files
documentation	API reference
drivers	APP driver interface source code and header files
external	Third-party library source code and header files
hal_drv	HAL driver interface source code and header files
platform	Link-related files
projects	Example project files and source code

1.1.2 Notices

- The whole SDK has been comprehensively retested based on the following environments.

Table 1-2 Item version

Item	Name & Version
IDE	Keil MDK-ARM Version 5.20, IAR 9.40.1, GCC gcc-arm-none-eabi-9-2020-q2-update-win32.zip
SoC	GR5405 series
Platform	Windows 7/10
Tools	• GProgrammer V2.0.2 • GRDirect Test Mode Tool V1.5.5 • GRPLT V1.6.0.0.03 • GRUart V2.1 • GRToolbox V2.21 • GRCalibration V1.1.0 • GRMesh V1.0.7

1.1.3 Limitations

- The GR5405 SDK might not work in versions earlier than Keil V5.20.
- There may be some problems with SEGGER J-Link and Keil.

  Visit https://www.segger.com/IDE_Integration_Keil.html#knownproblems for more details.

2 Revision History

2.1 GR5405 SDK V1.1.7

1. Initial release.

2.2 GR5405 SDK V1.1.8

2.2.1 System

1. Added data verification for eFuse UUID, OTP1, and Bluetooth addresses in sys_trim_info_sync(). If verification fails, an error will be returned and the system will enter an infinite loop by default.
2. Implemented logging of POR and full reset events. The specific reset reason can be obtained by calling sys_device_reset_reason().
3. Added the boot_update component, allowing updating the APP bootloader at the application layer.
4. Added the ble_debug component, enabling online retrieval of diagnostic logs via Bluetooth.
5. Added a MAC file for debugging purposes in IAR projects.
6. Resolved an issue where the NMI configuration was not automatically restored after waking up from sleep.
7. Resolved a hardfault issue that could occur due to invoking FPB PATCH during warm boot in edge cases.
8. Resolved a low-probability issue where the Bluetooth LE baseband might hang after waking up due to not checking the Bluetooth LE Comm Timer state before sleep.
9. Resolved a hardfault issue that could occur during byte-wise memory copying with memcpy from the GCC nano library when the FPB module was in use.
10. Fixed the UART_RX pin control logic in the swd_ctrl component, which should be configured to non-floating state.
11. Replaced WFI instructions with WFE to address potential loss of wake-up sources.
12. Optimized the APP bootloader and DFU master to support UART synchronization mode.
13. Optimized the RAM_CODE region configuration in the scatter-loading files for Keil, GCC, and IAR.
14. Optimized the CRC32 algorithm for generating static Bluetooth addresses based on Chip UID, reducing code size by 1 KB.
15. Optimized the Flash QE control logic during the startup process.
16. Updated the SDK to not proactively disable the WDT during the startup process by default. Instead, the WDT is now controlled by the WDT_RUN_ENABLE configuration at the application layer.

2.2.2 Peripheral

1. Added app_uart2lin_wakeup() to simulate sending a standard LIN wake-up signal via I/O.
2. Added the SPI2CAN driver.
3. Resolved an issue where, in sleep mode, starting the App Timer could cause lpcycles configuration failure with high probability.
4. Resolved an issue where the interrupt flag was not cleared during deinitialization of the ARM Timer.
5. Optimized exception handling for the UART2LIN driver.
6. Updated the default initialization of MSIO 7 to digital mode.

2.2.3 Bluetooth

1. Added support for the Mesh protocol stack and example projects.
2. Added support for configuring TX mode and TX power in the RF TX test interface during DTM testing.
3. Added ble_gap_hw_adv_data_get() to get advertising data from the hardware layer.
4. Added ble_gap_ignore_latency_set() to configure the number of times to ignore latency.
5. Added ble_gap_get_adv_addr_by_conidx(), allowing applications to get the corresponding advertising address by connection index.
6. Added ble_gap_reslv_rpa_addr_by_irk(), enabling applications to provide RPA and IRK for the SDK to check whether the IRK could successfully parse the RPA.
7. Added ble_sec_get_pair_info_by_conn_idx(), enabling applications to get pairing information by connection index.
8. Added ble_gap_default_conn_tx_power_set() and ble_gap_default_conn_tx_power_get() to allow configuring and getting the default TX power for connection.
9. Added the LLCP debug functionality.
10. Resolved an issue where the GATT Client could not distinguish between service discovery failure and successful service discovery completion.
11. Resolved an issue where rejecting a connection parameter update caused the application to report an abnormal event.
12. Fixed an issue where a 30-second timeout was not triggered after the slave initiated an SMP request and did not receive a response.
13. Resolved an issue where ble_gap_rpa_list_get() could result in illegal memory access under certain edge cases.
14. Resolved an issue where updating advertising data conflicted with other Bluetooth LE services in specific scenarios.
15. Fixed an advertising data buffer memory leak.
16. Resolved a rare scan assert issue caused by EM update delays when multiple services coexisted.
17. Resolved the probabilistic pairing failure when multiple devices initiated SC pairing simultaneously.
18. Optimized Bluetooth LE SDK white list management strategy.
19. Optimized the master connection method: As a master, always use the identity address for both initial and reconnections.
20. Optimized the formula for calculating the Bluetooth LE activity resource count.
21. Optimized the maximum connection limit: The system now rejects additional connection attempts once the maximum number of connections has been reached, before allowing new connections to be established.
22. Updated the default MTU configuration to 247.
23. Updated the execution time parameters of the frame pushing function to prevent the theoretical risk of baseband lock-up.