

by: NXP Semiconductors

1 Introduction

This document describes the OTA upgrade process of the wireless gamepad solution based on LPC5528 and NxH3670. The OTA upgrade process summarizes that the PC sends the new firmware to the dongle via USB, and the dongle forwards the new firmware to the gamepad via Bluetooth LE.

Figure 1 shows the system block diagram of the OTA upgrade.

Contents

1	Introduction.....	1
2	Partition table.....	2
3	Image header.....	4
4	OTA upgrade process.....	4
4.1	Preparations.....	4
4.2	GUI tool.....	5
4.3	OTA upgrade.....	7
5	Reference.....	10

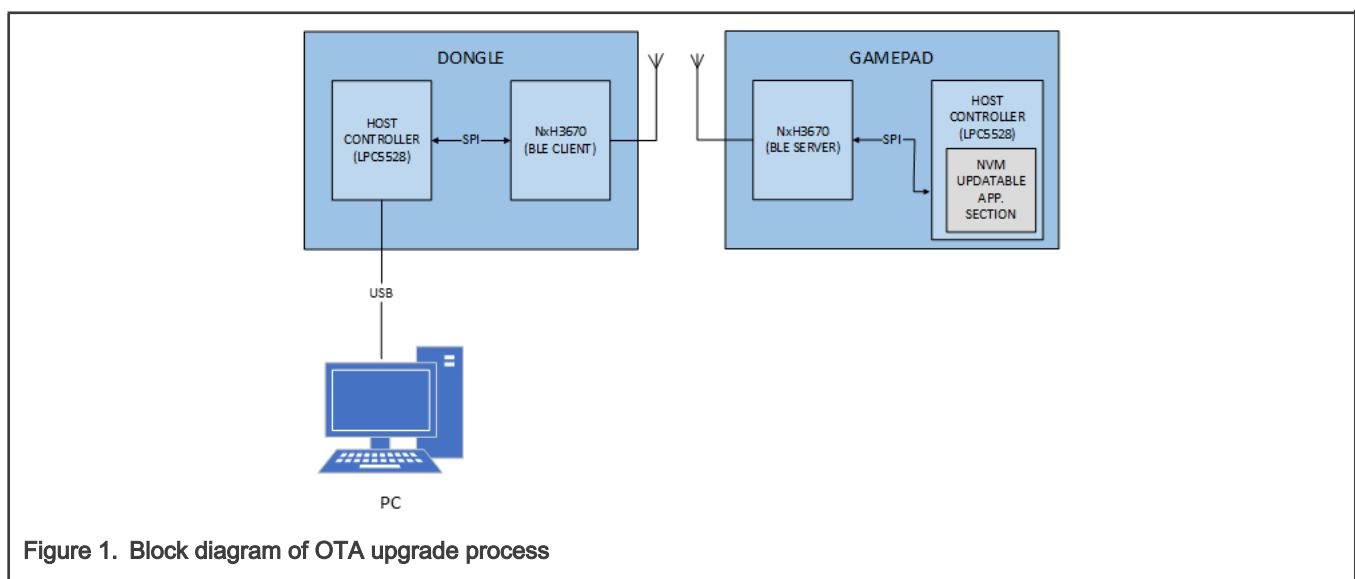


Figure 1. Block diagram of OTA upgrade process

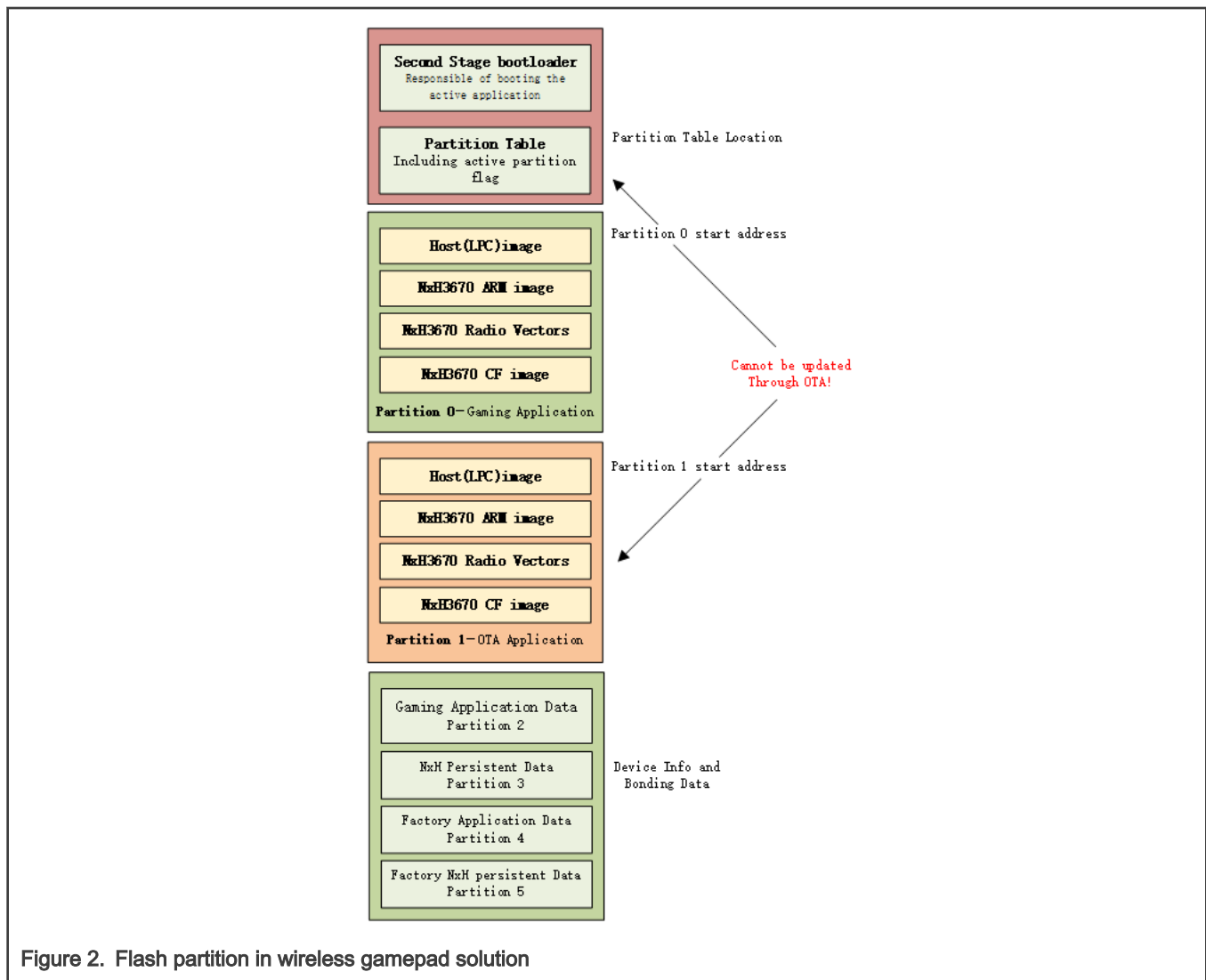
The steps are as follows:

1. PC sends OTA upgrade command to dongle.
2. Dongle forwards this command to the gamepad and establish BLE connection.
3. The host controller on gamepad board check the partition table before enabling the OTA application.
4. Gamepad enable OTA application.
5. If the OTA application is successfully enabled, restart the gamepad and run the OTA application.
6. The gamepad re-establishes a BLE connection with dongle.
7. Dongle notifies the PC after the connection is established.
8. The PC requests the partition table version of gamepad and verify that it is compatible.
9. The PC starts to send the firmware to dongle.
10. Dongle forwards each packet to the gamepad.
11. Gamepad check the updated partition table before enabling new firmware.

12. Gamepad enable new firmware.
13. If an error occurs before enabling the new firmware, continue to run the OTA application and upgrade again.
14. If the new firmware is successfully enabled, the host controller restarts and runs the new firmware, OTA upgrade is completed.

2 Partition table

During the OTA upgrade, both dongle and gamepad need to work in OTA mode. Therefore, LPC5528 stores both the user's application (app) and the OTA application (ota_app) that implements the OTA function. App and ota_app are stored in different locations in the flash. The storage location is determined by the flash partition table. In addition, LPC5528 stores Stage Second bootloader (SSB), Partition Table (PT), and some user data. In this solution, the flash partitions used by dongle and the gamepad are the same, as shown in Figure 2.



During the OTA upgrade process, the SSB, partition table, and ota_app part cannot be updated. The specific content of the partition table is determined by the *layout_debug_sdk.yml* file provided in the project. Figure 3 shows the partition table used in this solution.

```

1  # number of partitions: 6
2  # max used size (including ssb and table): 512 KB
3  active_partition: 0
4  layout_version: 0x52
5  app_type: 0          # 0--gamepad, 1-- dongle
6  partition_table_addr: 0x7000
7  entries:
8      # partition_id 0
9      - name: "app"
10         type: firmware
11         base_address: 0x81f0          #0xbf0
12         size: 0x41c00 # (263 KB) 0x41c00 + 0x1400(persdata) = 0x43000
13         offsets:
14             - 0x0          # kl_app | 143 KB (0x8000 - 0x2BE00, 52 KB free)
15             - 0x23c10      # nxh_app | 70 KB (4 KB free)
16             - 0x35410      # rfmac | 18 KB (2 KB free)
17             - 0x39c10      # cf | 32 KB (2 KB free)
18         # partition_id 1
19         - name: "ota"
20             type: firmware
21             base_address: 0x501f0
22             size: 0x22d00 # 140k KB
23             offsets:
24                 - 0x0          # lpc_ota_app | 80 KB
25                 - 0x13e10      # nxh_ota_app | 60 KB
26         # partition_id 2
27         - name: "app_data"
28             type: appdata
29             base_address: 0x4a800
30             size: 0x400 # 1 KB
31             offsets:
32                 - 0x0          # app_data | 1 KB
33         # partition_id 3
34         - name: "nxh_pers_data"
35             type: appdata
36             base_address: 0x4ac00
37             size: 0x400 # 1 KB
38             offsets:
39                 - 0x0          # nxh_pers_data | 1 KB
40         # partition_id 4
41         - name: "app_data_factory"
42             type: factoryreset
43             base_address: 0x4b000
44             size: 0x400 # 1 KB
45             offsets:
46                 - 0x0          # app_data_factory | 1 KB
47         #partition_id 5
48         - name: "nxh_pers_factory"
49             type: factoryreset
50             base_address: 0x4b400
51             size: 0x400 # 1 KB
52             offsets:
53                 - 0x0          # nxh_persistent_data_factory | 1 KB
54

```

Figure 3. layout_debug_sdk.yml

It is also necessary to use the LPC5528_NxH3670 Flash Tool to parse this file and write the parsed content into binary, then download it to a fixed location in the Flash. In this solution, the partition table is stored at 0x7000.

As shown in [Figure 3](#), the partition table contains six partitions, where **Partition 0** stores the firmware of the app and **Partition 1** stores the firmware of `ota_app`. After being powered on, the LPC5528 executes the stage second bootloader program, reads the partition table at 0x7000, and determines the partition to boot next according to the `active_partition` flag in the partition table. If `active_partition` is 0, SSB boots the gaming application. If `active_partition` is 1, SSB boots the OTA application to enable the OTA function. By default, the `active_partition` is 0.

3 Image header

Before the OTA upgrade, here is another concept of **image header**, which is the head that is added before the image. The content is as shown in [Figure 4](#).

```

139 typedef struct __attribute__((packed)) {
140     uint32_t length : 24; /**< length in memory size words */
141     uint32_t imgType : 8; /**< imgType, see @ref NVMMGR_EepFileImageType_t */
142     uint32_t address; /**< start byte address */
143     uint32_t checksum; /**< checksum value */
144 } NVMMGR_NvmImageHeader_t;

```

Figure 4. Image header

Image header is 16 bytes in total, including image length, image type, starting address and checksum. Before performing an OTA upgrade, the dongle will read the image header in the remote (Gamepad) flash and compare it with the new imager header.

- If the headers of two images are the same, they are considered the same image. The dongle will skip this part and this image will not be sent to the gamepad.
- If they are different, the new image will be sent to the remote to replace the old image.

Developers can use Keil's after-build command or LPC5528_NxH3670 Flash Tool to generate EPP files with image headers. For details, see *Getting Started with LPC5528 Wireless Gamepad Solution* (document [AN13082](#)).

4 OTA upgrade process

This section introduces the specific steps of the OTA upgrade of wireless gamepad solution.

4.1 Preparations

4.1.1 Hardware

[Figure 5](#) shows a gamepad board and a dongle board.

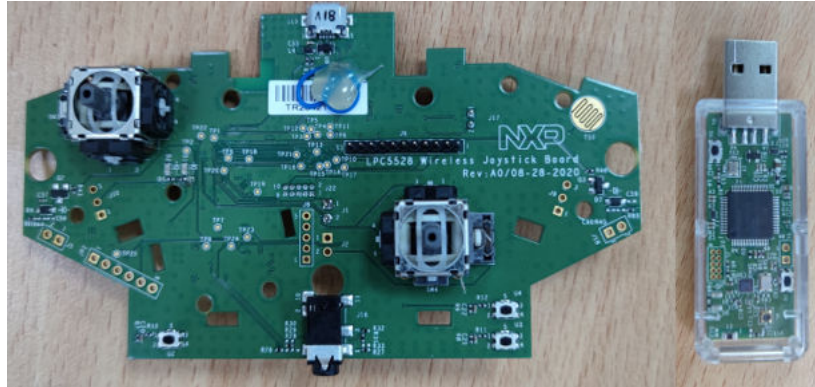


Figure 5. LPC5528 Gamepad board and Dongle board

4.1.2 Software

- Gamepad application
- OTA application
- *layout_debug_sdk.yml* and *flashlist_debug_sdk.yml*
- NxH3670 related firmware
- GUI tool: LPC5528_NxH3670 Flash Tool and NxH3670 Flash Tool

4.2 GUI tool

In order to facilitate users to download programs and perform OTA upgrade, NXP provides NxH3670 Flash tool in the NxH3670 SDK Rev5.2 package, as shown in [Figure 6](#).

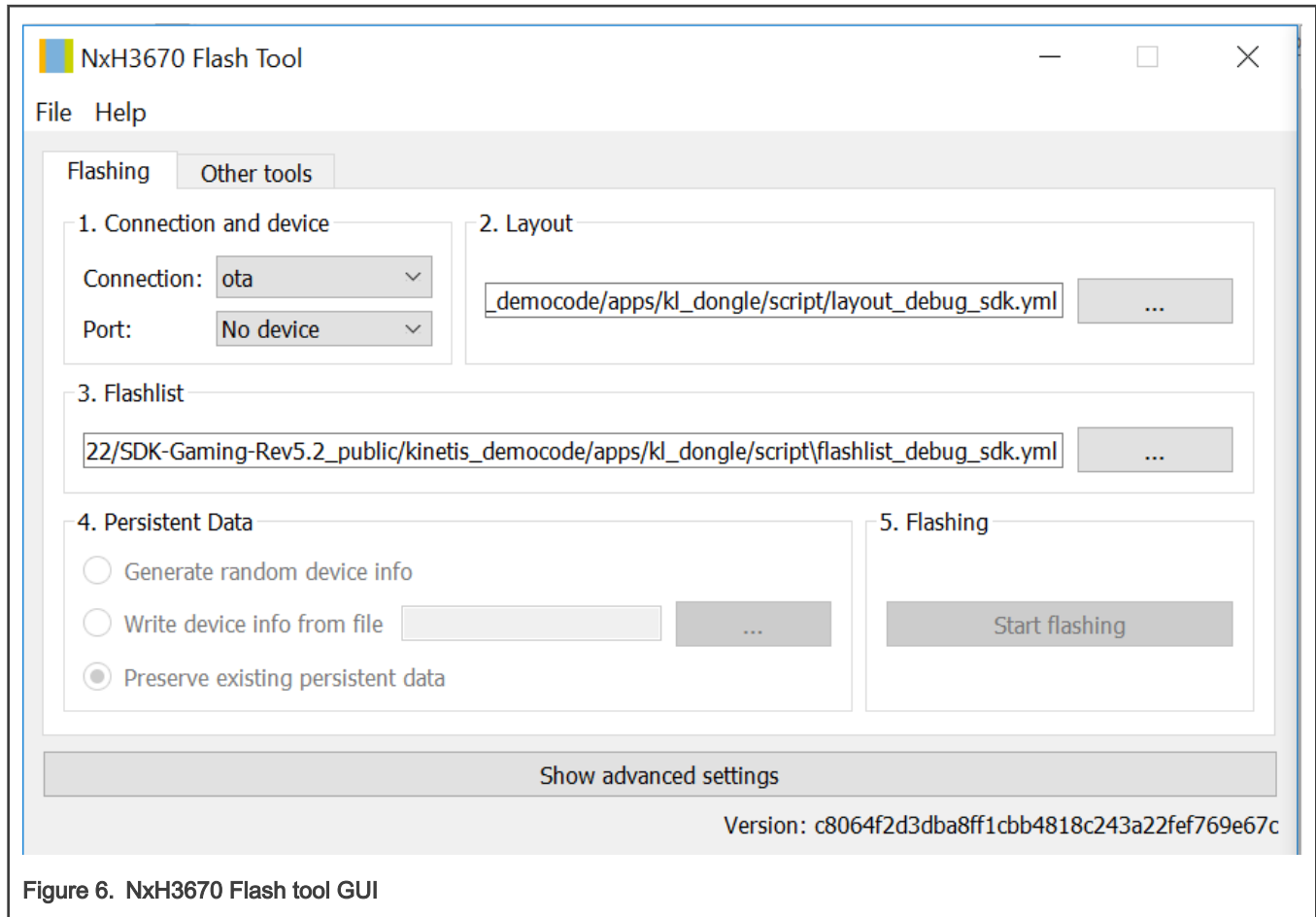


Figure 6. NxH3670 Flash tool GUI

Two main functions are supported by NxH3670 Flash tool:

- Host controller program download

Currently, this tool does not support LPC5528 program download, so we re-developed a LPC5528 NxH3670 Flash tool in this solution, as shown in [Figure 7](#).

- OTA upgrade

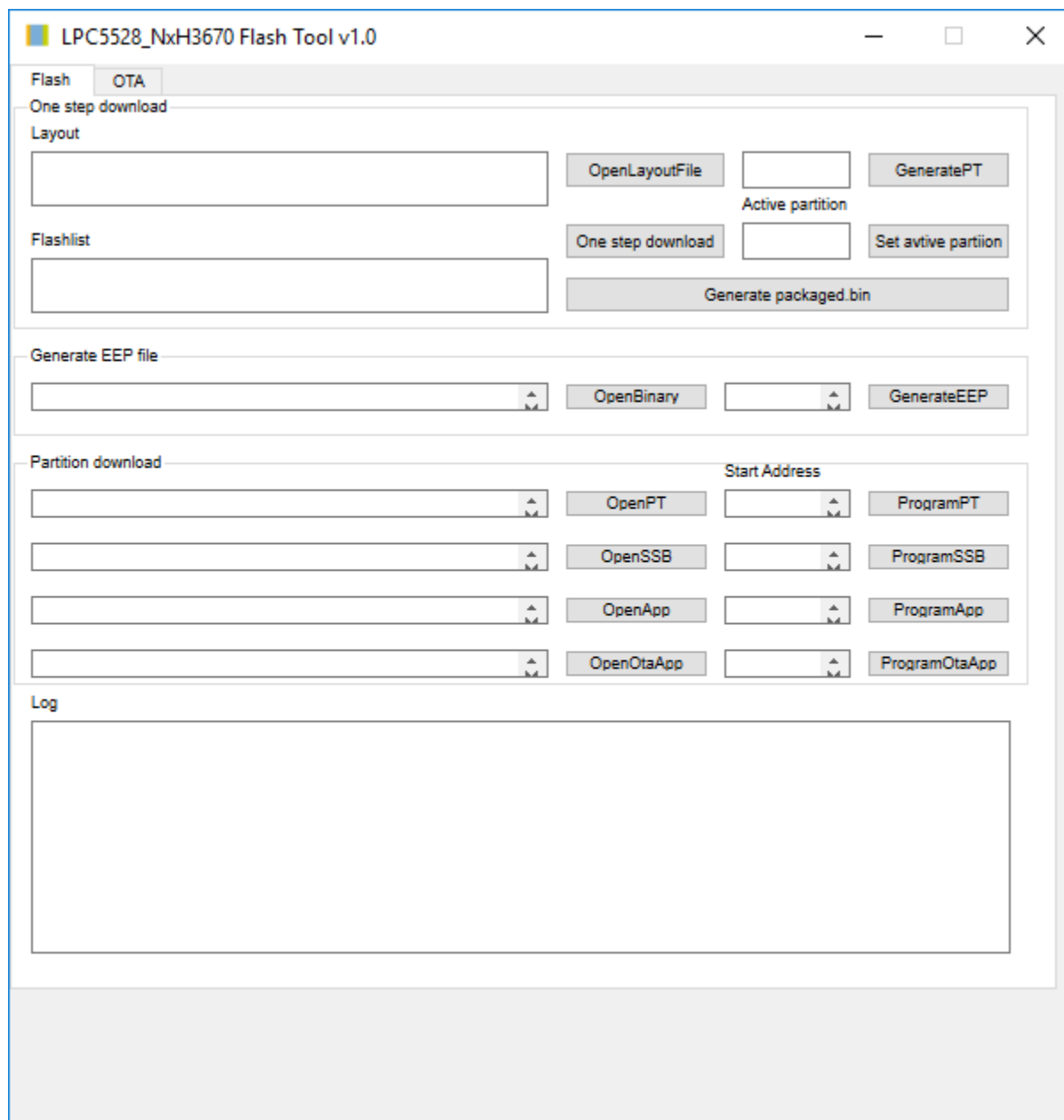


Figure 7. LPC5528 NxH3670 Flash tool v1.0

For the detailed usage LPC5528_NxH3670 Flash tool, see *Getting Started with LPC5528 Wireless Gamepad Solution* (document [AN13082](#)).

NOTE

Before using this tool to download the LPC5528 program, make sure that the MCU enters the ISP mode and the HS USB interface is used.

4.3 OTA upgrade

Users can use the two flash tools mentioned in [GUI tool](#) to implement the OTA upgrade.

The specific steps of OTA upgrade are as follows:

1. Short circuit J2 on the gamepad board and reset LPC5528 to make MCU enter ISP mode.
 - a. Open the LPC5528_NxH3670 Flash tool and select the *layout_debug_sdk.yml* file corresponding to the gamepad application.
 - b. Click the **Generate packaged.bin** button to generate a *packaged.bin*.
 - c. Click the **One step download** button to download the complete firmware of gamepad to the Flash.

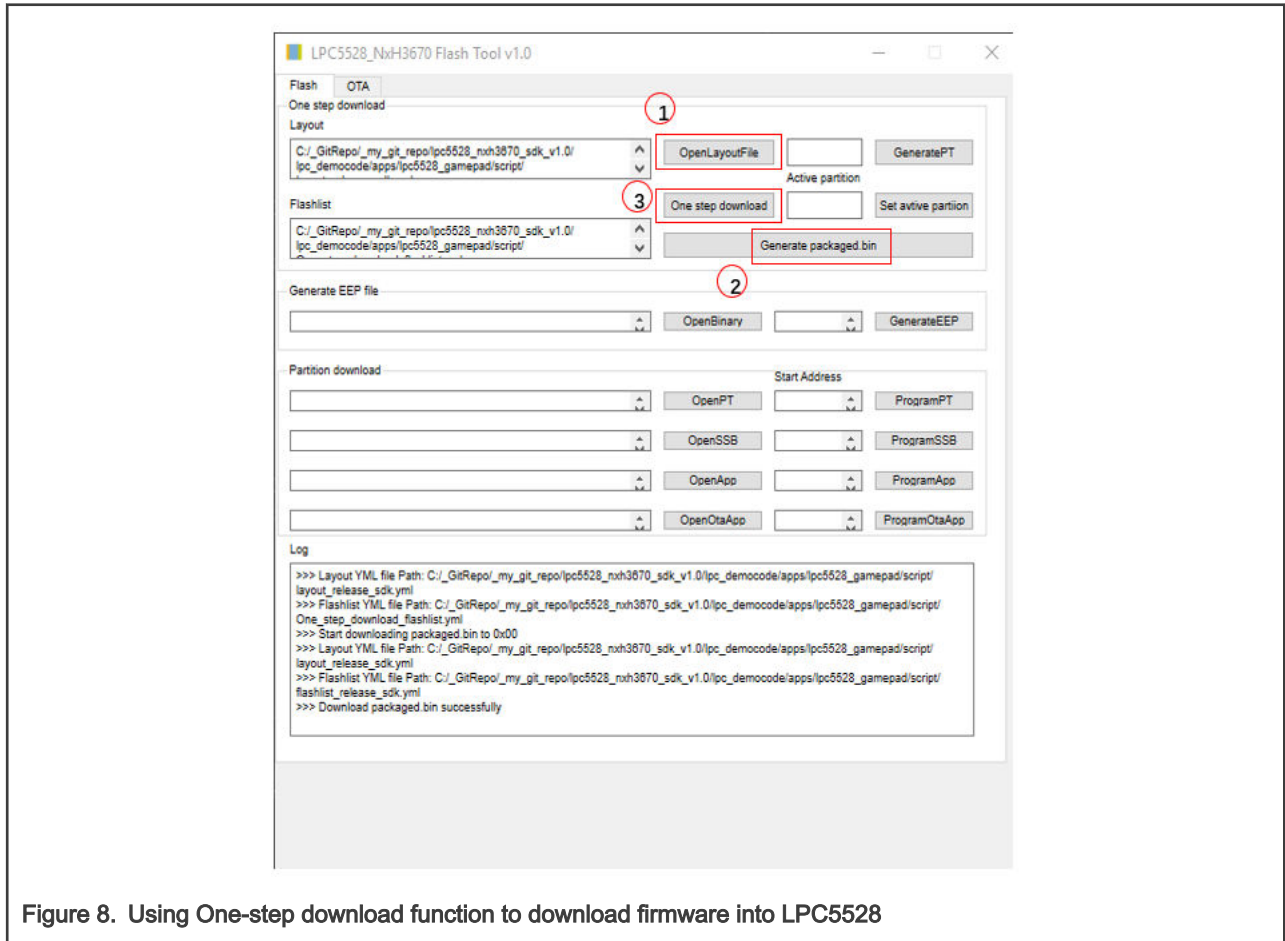


Figure 8. Using One-step download function to download firmware into LPC5528

If you have already generated a *packaged.bin* and do not need to update, there is no need to generate a *packaged.bin* again.

2. Press the ISP button, **U1**, on dongle board and reset the dongle board to make the dongle board enter the ISP mode. Then use the same method as [Step 1](#) to download complete dongle firmware.
3. Reset the dongle board and gamepad board, make them to run respective applications, and wait for the Bluetooth LE connection to be established. After the pairing is successfully, the red LED, **D2**, on the two boards will turn off and the PC recognize two Xbox 360 peripherals, as shown in [Figure 9](#).

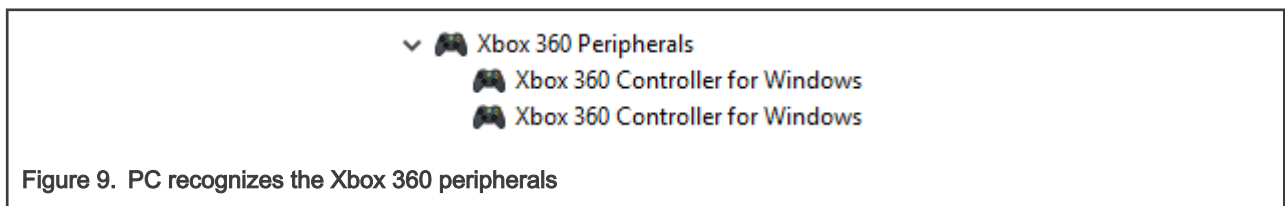


Figure 9. PC recognizes the Xbox 360 peripherals

4. Make the dongle re-enter ISP mode, enter **1** in the **Active partition** window, click the **Set active partition** button, set the active partition of dongle to partition 1 (ota_app), and reset the dongle to make it work in the OTA mode.

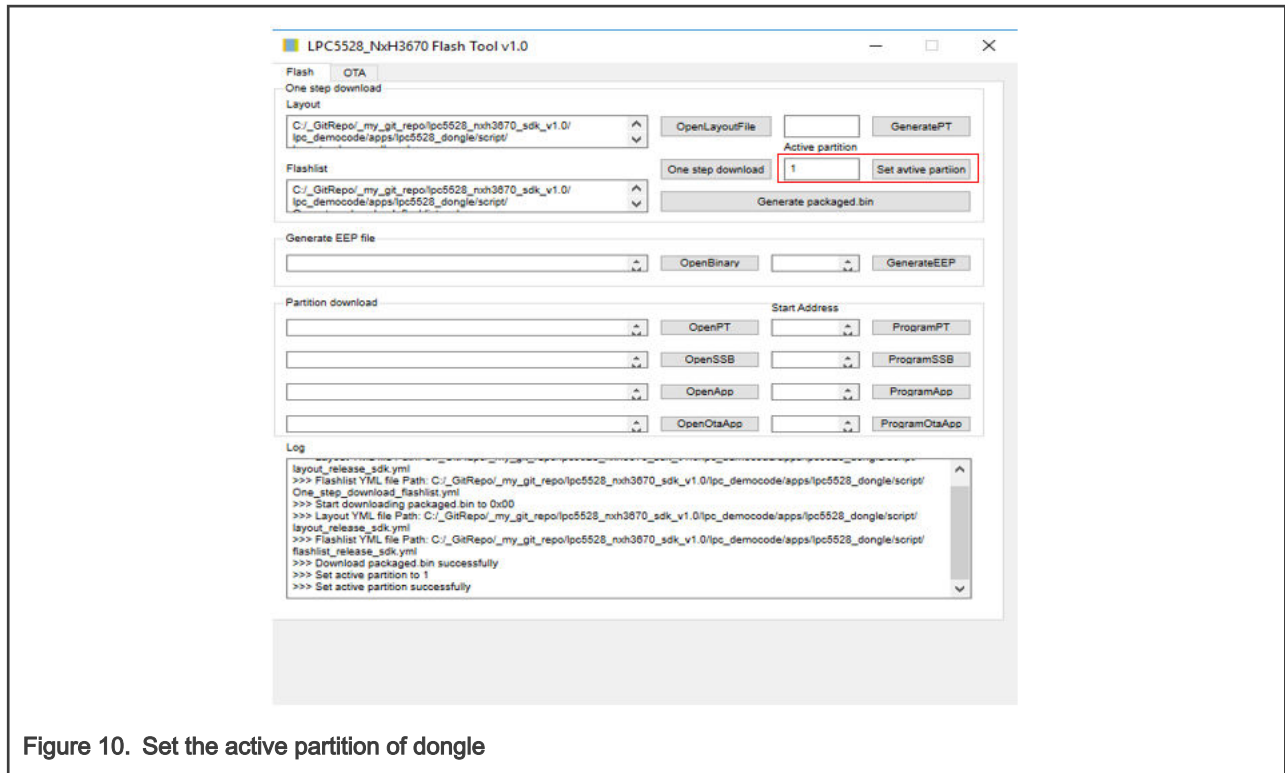


Figure 10. Set the active partition of dongle

5. Use NxH3670 Flash tool to perform OTA upgrade. Open NxH3670 Flash tool and follow the steps below:

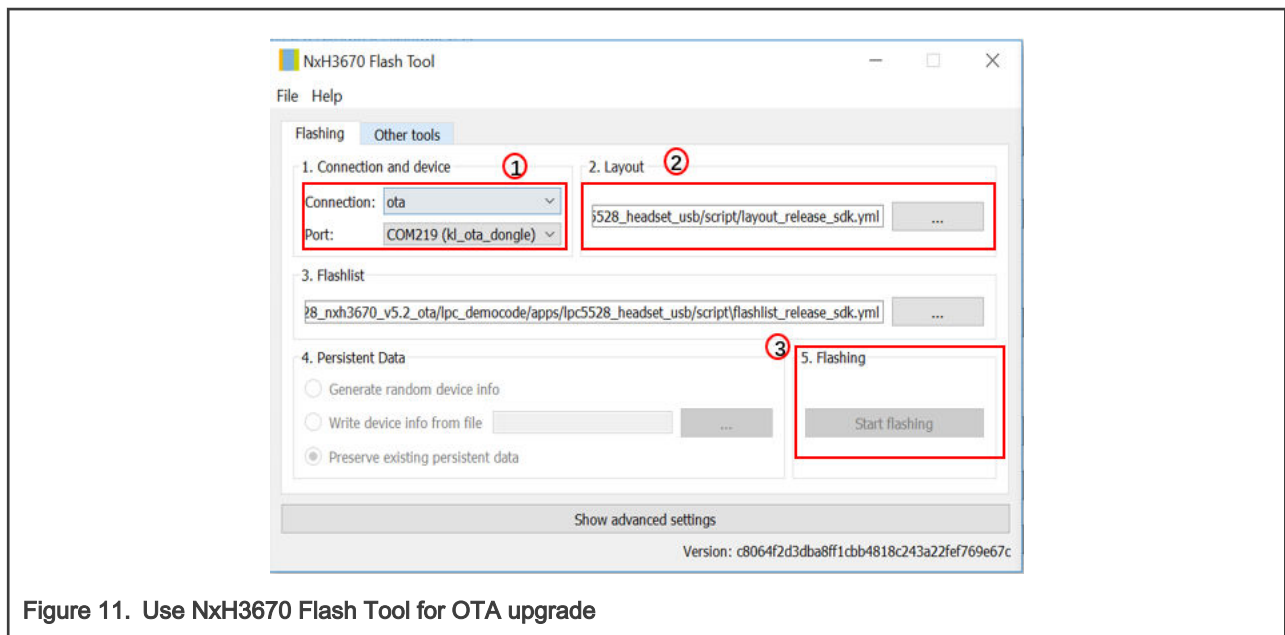


Figure 11. Use NxH3670 Flash Tool for OTA upgrade

- a. Select the connection mode as **ota** and select the VCOM port of the dongle board in the **Port** column.
- b. Select the *layout_debug_sdk.yml* file corresponding to the new firmware.
- c. Click the **Start flashing** button to start the OTA upgrade.

During the OTA upgrade, the log information is printed in the NxH3670 Flash Tool Logger window, as shown in [Figure 12](#).

```

NxH3670 Flash Tool - Logger

#### Checking partition table version ...
+++ reading layout file (C:/_GitRepo/_my_git_repo/
lpc5528_nxh3670_v5.2_ota_demo/appcode/apps/lpc5528_headset_usb/
script/layout_release_sdk.yml) ...
+++ connecting to the remote device ...
+++ switching remote active partition to OTA partition ...
+++ rebooting remote to active partition ...
+++ connecting to the remote device ...
Remote partition table version: 0x52
#### Creating partition table layout ...
+++ no access to partition table for ota connection
#### Flashing binaries ...
+++ reading flash list file ...
+++ checking flash list file ...
--- skipping sbl image
+++ flashing id_app @ index 0 ...
+++ calculating local fingerprint for index 0 ...
local fingerprint: 1358298789
+++ getting remote fingerprint for index 0 ...
remote fingerprint: 1584298847
@ [0] 0x0 -> eep/lpc5528_headset_usb.bin.eep
+++ flashing nxh_app @ index 1 ...
+++ calculating local fingerprint for index 1 ...
local fingerprint: 3319962735
+++ getting remote fingerprint for index 1 ...
remote fingerprint: 3319962735
=== fingerprints match, skipping update ...
+++ flashing rfmac @ index 2 ...
+++ calculating local fingerprint for index 2 ...
local fingerprint: 1098492557
+++ getting remote fingerprint for index 2 ...
remote fingerprint: 1098492557
=== fingerprints match, skipping update ...
+++ flashing cf @ index 3 ...
+++ calculating local fingerprint for index 3 ...
local fingerprint: 420803260

Current: 100%
Overall: 100%
Version: c8064f2d3dba8ff1cbb4818c243a22ef769e67c

```

Figure 12. Log information during OTA upgrade

As shown in Figure 12, after receiving the OTA upgrade command from the PC, the dongle will first send a command to the remote (gamepad) to make the gamepad enter the OTA mode, reconnect the gamepad, and then send the specific binaries provided by *flashlist_debug_sdk.yml* file to the gamepad. Before sending, the NxH3670 Flash Tool compare the remote image header and the local (PC) image header firstly. If the contents of the image headers are the same, the upgrade progress of current file will be skipped. If they are different, the current file will be sent to the gamepad and written into the specified position of gamepad. After the OTA process is completed, dongle send the reboot command to restart the gamepad and run the new firmware.

NOTE

If the dongle and gamepad have been paired, and the dongle is already running in OTA mode, you do not need to perform Step 1 to Step 4 and can directly use Step 5 to update the new firmware.

5 Reference

- *Over-the-air firmware update* (document [AN12361](#))
- *Getting started with LPC5528 Wireless Gamepad Solution* (document [AN13082](#))
- *NxH3670 Flash Tool* (document UM11198)

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, ICODE, JCOP, LIFE, VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, Altivec, CodeWarrior, ColdFire, ColdFire+, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, Tower, TurboLink, EdgeScale, EdgeLock, eIQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© NXP B.V. 2020.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: December 16, 2020

Document identifier: AN13083

