Binding UID to M031 LDROM and APROM Firmware

Example Code Introduction for 32-bit NuMicro® Family

Document Information

Application	This sample code binds UID code to M031 LDROM and APROM firmware.
BSP Version	M031_Series_BSP_CMSIS_V3.04.000
Hardware	NuMaker-M032SE V1.3

1. Overview

Each M031 series microcontroller (MCU) has a 96-bit unique ID (UID) code. This document introduces how to bind the UID to firmware to prevent the dumped user firmware from running in an illegal copy product.

1.1 Principle

The firmware can judge the UID code to determine the execution process of MCU. In this way, even if the MCU firmware is read and downloaded to another MCU with the same part number, the correct operation result cannot be obtained. When firmware needs to be sent to users to upgrade by themselves, but only the device with a specific MCU needs to use the new firmware, the users can bind the UID of a specific MCU to the LDROM and APROM firmware through this sample code.

This sample code contains two projects "LDROM_Bind_UID" and "APROM_Bind_UID". The "LDROM_Bind_UID" is placed in LDROM, and its general principle of secret key judgment is shown in Figure 1-1. The "APROM_Bind_UID" is placed in APROM, and its general principle of secret key judgment is shown in Figure 1-2. The details of the secret key judgment flow are described in Code Description section.

Note: The "LDROM_Bind_UID" needs SRecord toolkit (http://srecord.sourceforge.net/) and uses *srec_cat.exe* to merge the generated program file and secret key file into one file "LDROM_Bind_UID.bin".

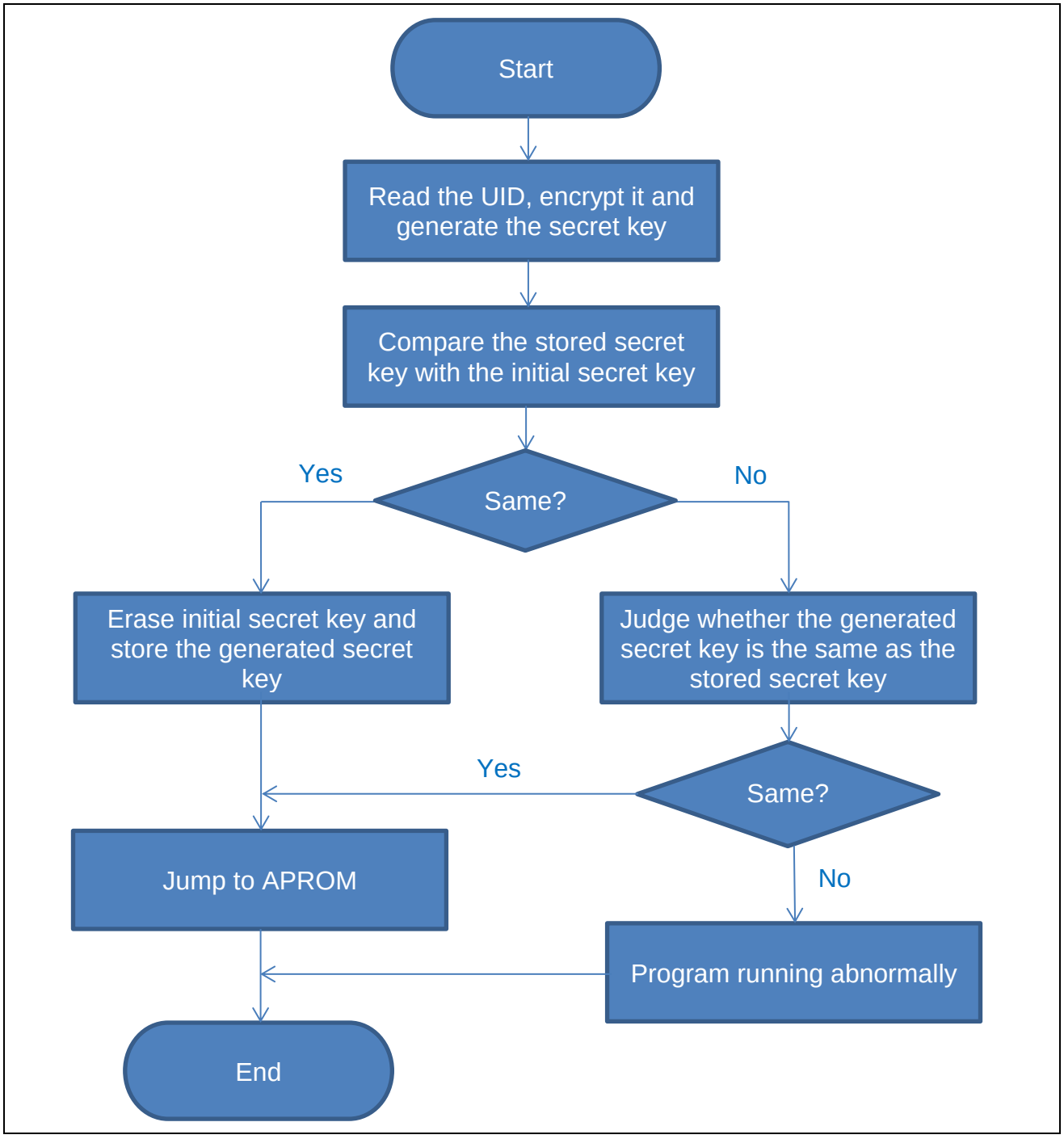

Figure 1-1 LDROM Firmware Secret Key Judgment Flow

Note: To use this method, ensure that the MCU has been powered on and run once before the product shipped from the factory, so that the initial secret key stored in APROM has been erased.

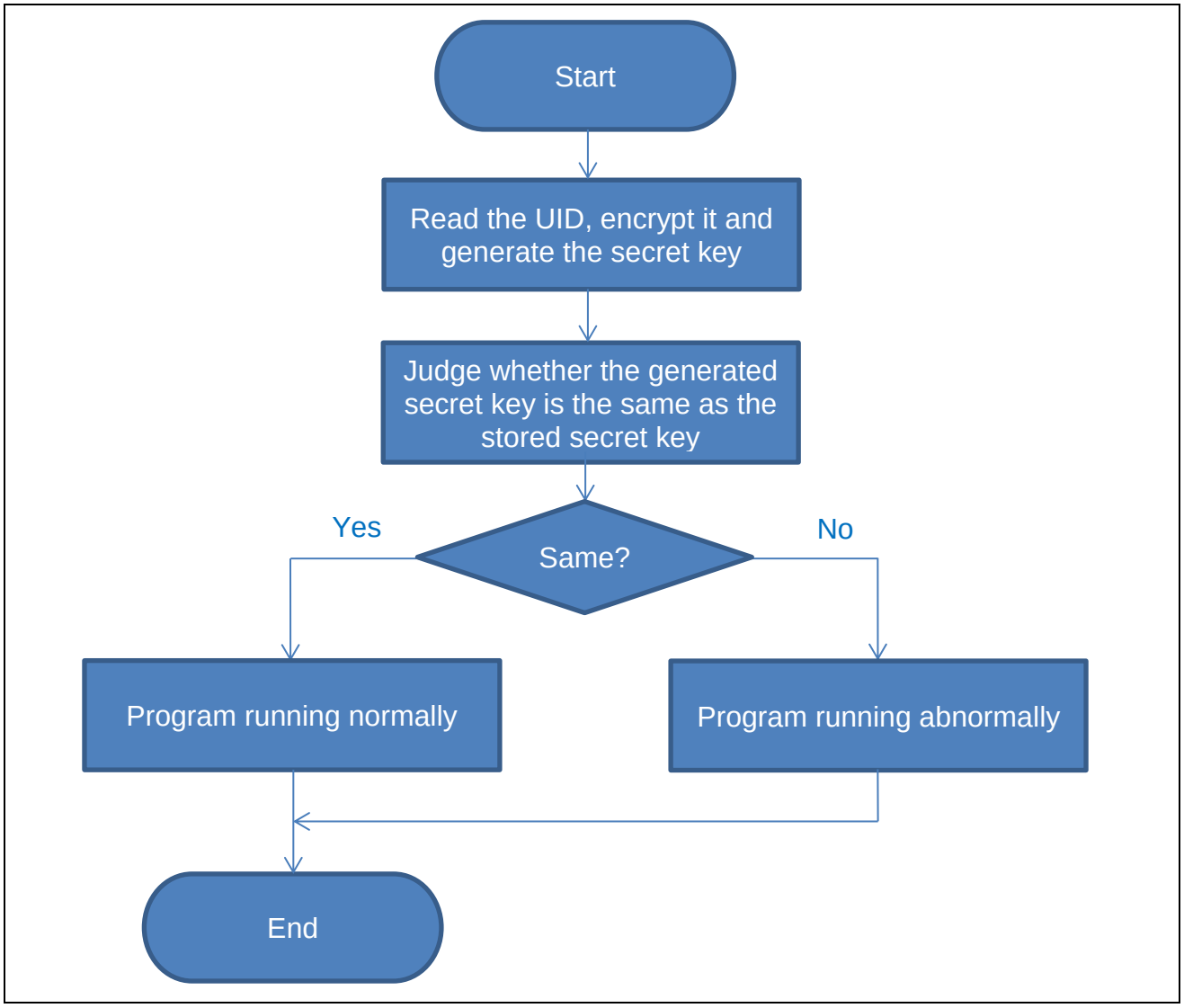

Figure 1-2 APROM Firmware Secret Key Judgment Flow

1.2 Demo Result

Download the *LDROM_Bind_UID.bin* and *APROM_Bind_UID.bin* to NuMaker-M032SE V1.3 board, and set its booting mode to “LDROM with IAP” by NuMicro ICP Programming Tool. The printed information during execution is shown in Figure 1-3.

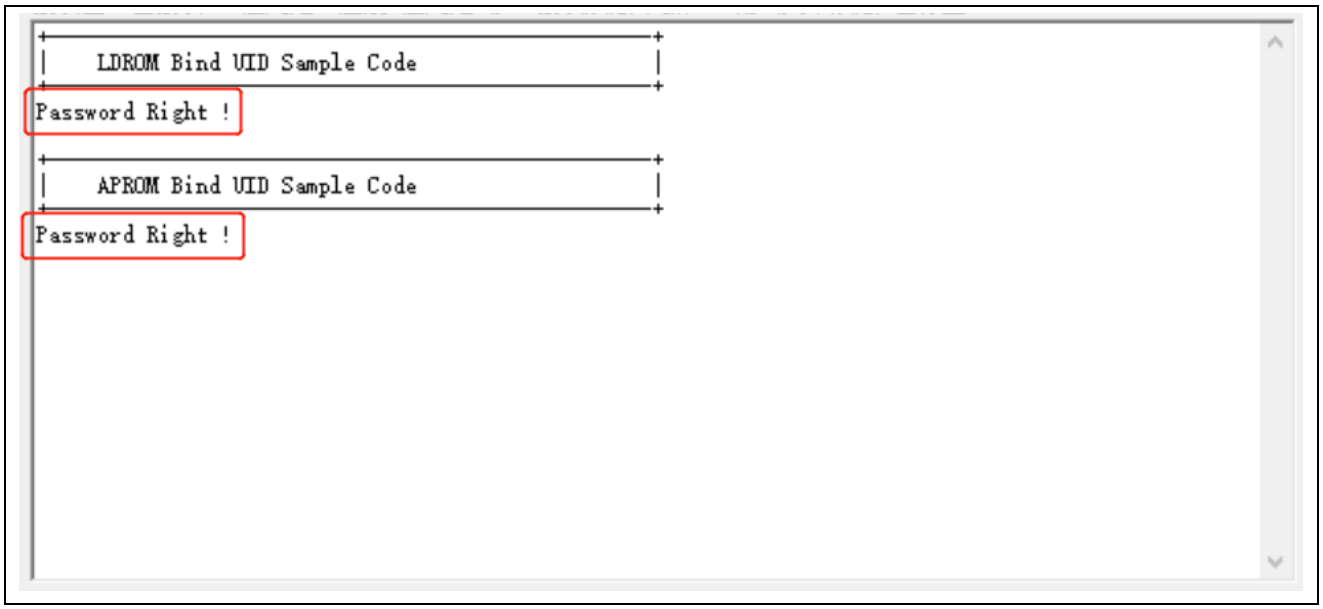

Figure 1-3 Printed Information when Executing Normally

Read the LDROM firmware and APROM firmware form NuMaker-M032SE V1.3 by NuMicro ICP Programming Tool, download them to another NuMaker-M032SE V1.3 board, and set its booting mode to “LDROM with IAP”, too. The printed information during execution is shown in Figure 1-4.

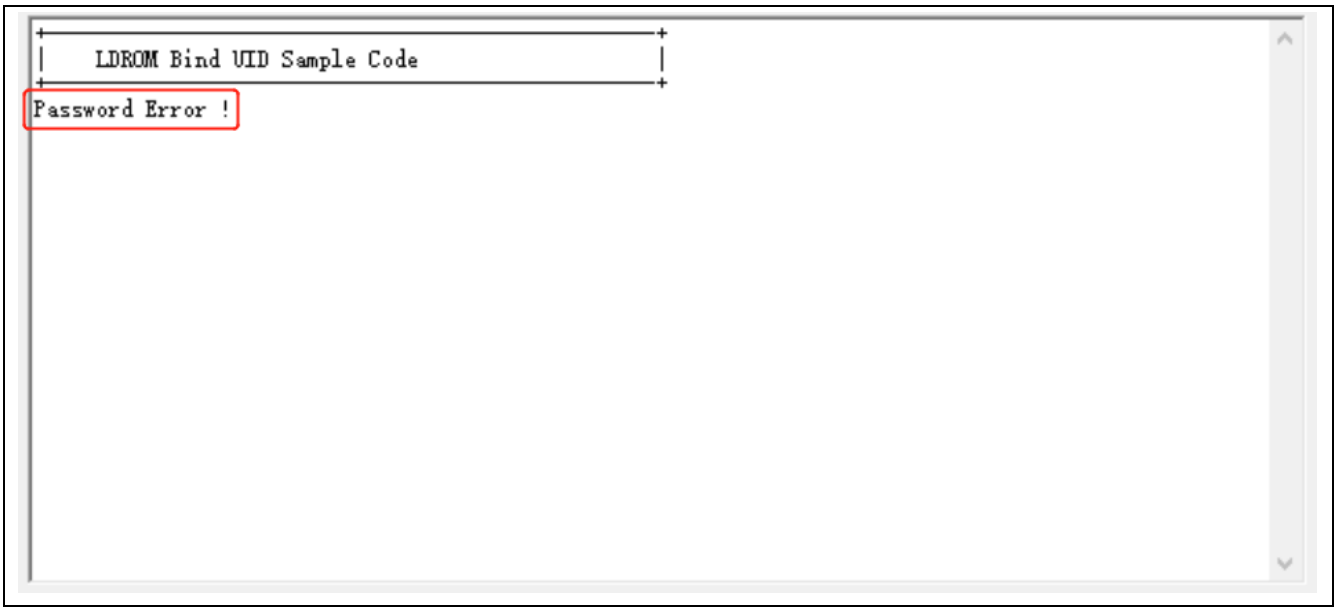

Figure 1-4 Printed Information when LDROM Firmware Executing Normally

Read APROM firmware form NuMaker-M032SE V1.3 by NuMicro ICP Programming Tool, download it to another NuMaker-M032SE V1.3 board, and set its booting mode to “APROM”. The printed information during execution is shown in Figure 1-5.

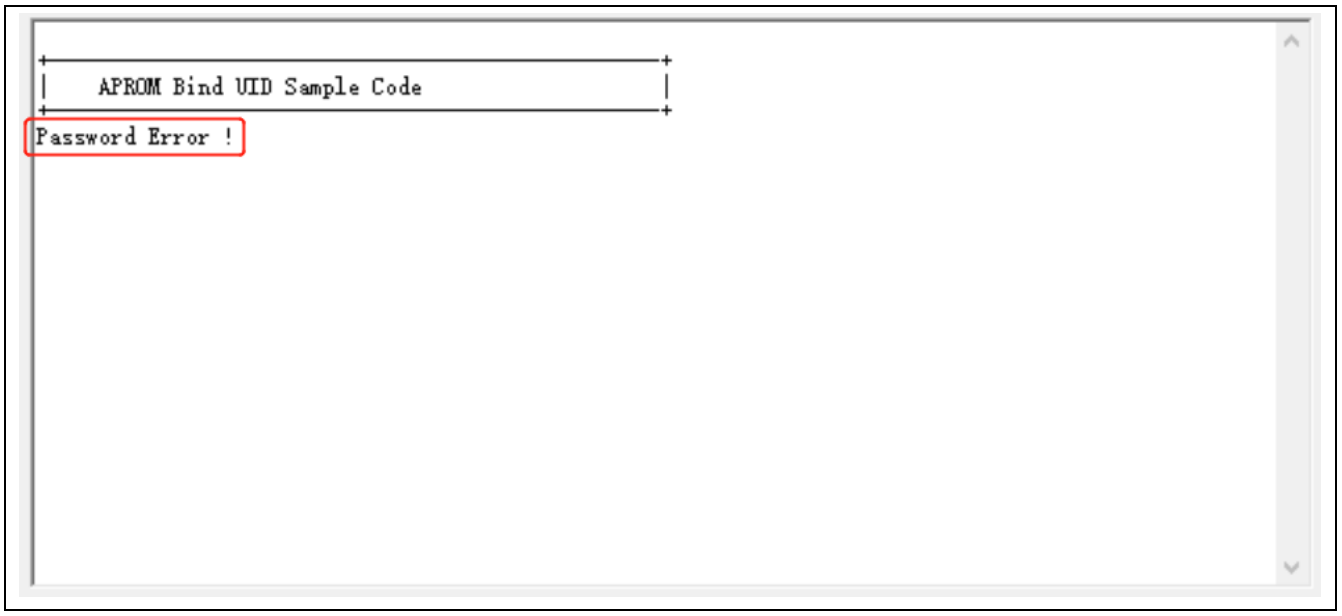

Figure 1-5 Printed Information when APROM Firmware Executing Normally

2. Code Description

Define initial secret key. Users can define a secret key by themselves.

```
#define INIT_PASSWORD_0       0x12121212 /* Defined by user */
#define INIT_PASSWORD_1       0x34343434 /* Defined by user */
#define INIT_PASSWORD_2       0x56565656 /* Defined by user */
```

Encrypt the UID. Users can change it according to their encryption algorithm.

```
void UID_Encrypt(uint32_t au32Uid[3])
{
    uint8_t i;

    /* Encrypt the UID, users can change it according to their encryption algorithm */
    for(i = 0; i < 3; i++)
    {
        au32Uid[i] = (~au32Uid[i]) * 5 / i;
    }
}
```

Judge whether the secret key is the same as the stored initial key. If yes, erase the initial key, and write the new secret key encrypted by UID. If no, judge whether the secret key encrypted by the UID is the same as the stored secret key.

```
uint8_t UID_PasswordCheck(void)
{
    uint8_t i;
    uint32_t au32Uid[3];
    uint32_t *pu32Password = (uint32_t *)PASSWORD_ADDR;

    UID_Read(au32Uid);
    UID_Encrypt(au32Uid);

#if defined(PASSWORD_INIT_ENABLE)
if((pu32Password[0] == INIT_PASSWORD_0) && (pu32Password[1] == INIT_PASSWORD_1)
&& (pu32Password[2] == INIT_PASSWORD_2))
    {
        UID_PasswordWrite(au32Uid);
    }
    else
#endif
    {
        for(i = 0; i < 3; i++)
        {
            if(au32Uid[i] != pu32Password[i])
            {
                return 0;
            }
        }
    }

    return 1;
```

```
}
```

3. Software and Hardware Requirements

3.1 Software Requirements

- BSP version
  - M031_Series_BSP_CMSIS_V3.04.000
- IDE version
  - Keil uVersion 4.74

3.2 Hardware Requirements

- Circuit components
  - NuMaker-M032SE V1.3

4. Directory Information

The directory structure is shown below.

EC_M031_LDROM_APROM_Bind_UID_V1.00	
Library	Sample code header and source files
CMSIS	Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
Device	CMSIS compliant device header file
StdDriver	All peripheral driver header and source files
SampleCode	
ExampleCode	Source file of example code

Figure 4-1 Directory Structure

5. Example Code Execution

1. Browse the sample code folder as described in the Directory Information section and double-click *LDROM_Bind_UID.uvproj* and *APROM_Bind_UID.uvproj*.
2. Enter Keil compile mode.
   - Build
3. Run
   - Download firmware and configuration bits by NuMicro ICP Programming Tool
   - Run

6. Revision History

Date	Revision	Description
2022.04.29	1.00	1. Initially issued.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.