Application Note

32-bit Cortex™-M0 MCU
NuMicro® Family

How to configure the User Configuration

Application Note

Table of Contents-

1 INTRODUCTION

This document explains the sample code, “Smpl_Configuration” which is include in the ***AN_1012_EN.ZIP*** file and demonstrates how to configure the User Configuration in LDROM.

1.1 Features

- Configure the follows User Configurations, Brown Out Level at 2.2/2.6/3.8/4.5V, Brown Out Reset Enable/Disable and CBS(Config Boot Selection) for select boot up from LDROM or APROM after system power-on or H.W reset.
- Erase/Write Data Flash test.

1.2 Limitation

- The file size of the Smpl_Configuration.bin should not large than 4KB for write to LDROM.
- Configure the User Configuration is only available in LDROM.
- Except write the Smpl_Configuration.bin to LDRAM, user must also download the same binary to APROM to let the application can execute in LDROM. The detail download sequence will describe in **EXECUTION Environment Setup and Result** chapter.

2 CODE SECTION

Most the verified functions are declared in Smpl_Configuration.c.

Refer chapter of Calling Sequence for the detail calling sequence

2.1 Main function

In the main function, we set the external 12MHz as the system clock for the later test.

After the hardware initialization is finished, open UART0 and configure baud rate to 115200 for output debug message. Then, call DoFunction() to start your test.

2.2 TestFunction function

Check system boot up is from LDROM or APROM

```
/* Enable ISP function */
DrvFMC_EnableISP(1);

/* Read BS */
printf("Boot up from ");
if (DrvFMC_GetBootSelect() == APROM)
{
        printf("[APROM].\n");
       printf("Must reset to LDROM mode.\n");
      _CPUResetToLDROM();
      DrvFMC_EnableISP(0);
      LOCKREG();
       while(1);
}else
{
       printf("[LDROM].\n\n");
}
```

Read system information

```
……
if (DrvFMC_ReadCID(&u32Data) == 0)            /* Read Company ID */
          printf("Company ID is ..... [0x%08x].\n", u32Data);

if (DrvFMC_ReadDID(&u32Data) == 0)            /* Read Device ID */
          printf("Device ID is ..... [0x%08x].\n", u32Data);
```

```
/* Set DataFlash Base Address */
DrvFMC_Write(CFG1_BASE, DATA_FLASH_BASE);IP_CLOCK)
```

Select an item to test

```
……
printf("| [1] Erase Data Flash  |\n");
printf("| [2] Write Data Flash  |\n");
printf("| [3] Set Configuration |\n");
……
switch (u8Item)
{
        case '1':
        case '2':
                _DataFlashTest(u8Item-'1', TRUE);       /* 0:Erase , 1:Write */
                break;
        case '3':
                _ConfigurationSetting();
                break;
}
……
```

2.3 _DataFlashTest function

This function is used for Erase or Write Data Flash test at start address 0x0001F000.

```
……
if (u8Index == 0)
{
        /* Erase data flash */
        for (u32cnt=DATA_FLASH_BASE; u32cnt<FLASH_END_BASE;
                                                u32cnt+=PAGE_SIZE)
                DrvFMC_Erase(u32cnt);
}else
if (u8Index == 1)
{
        /* Write data flash */
        for (u32cnt=DATA_FLASH_BASE; u32cnt<FLASH_END_BASE; u32cnt+=4)
                DrvFMC_Write(u32cnt, u32cnt);
}
```

```
/* Verify the data flash */
……
```

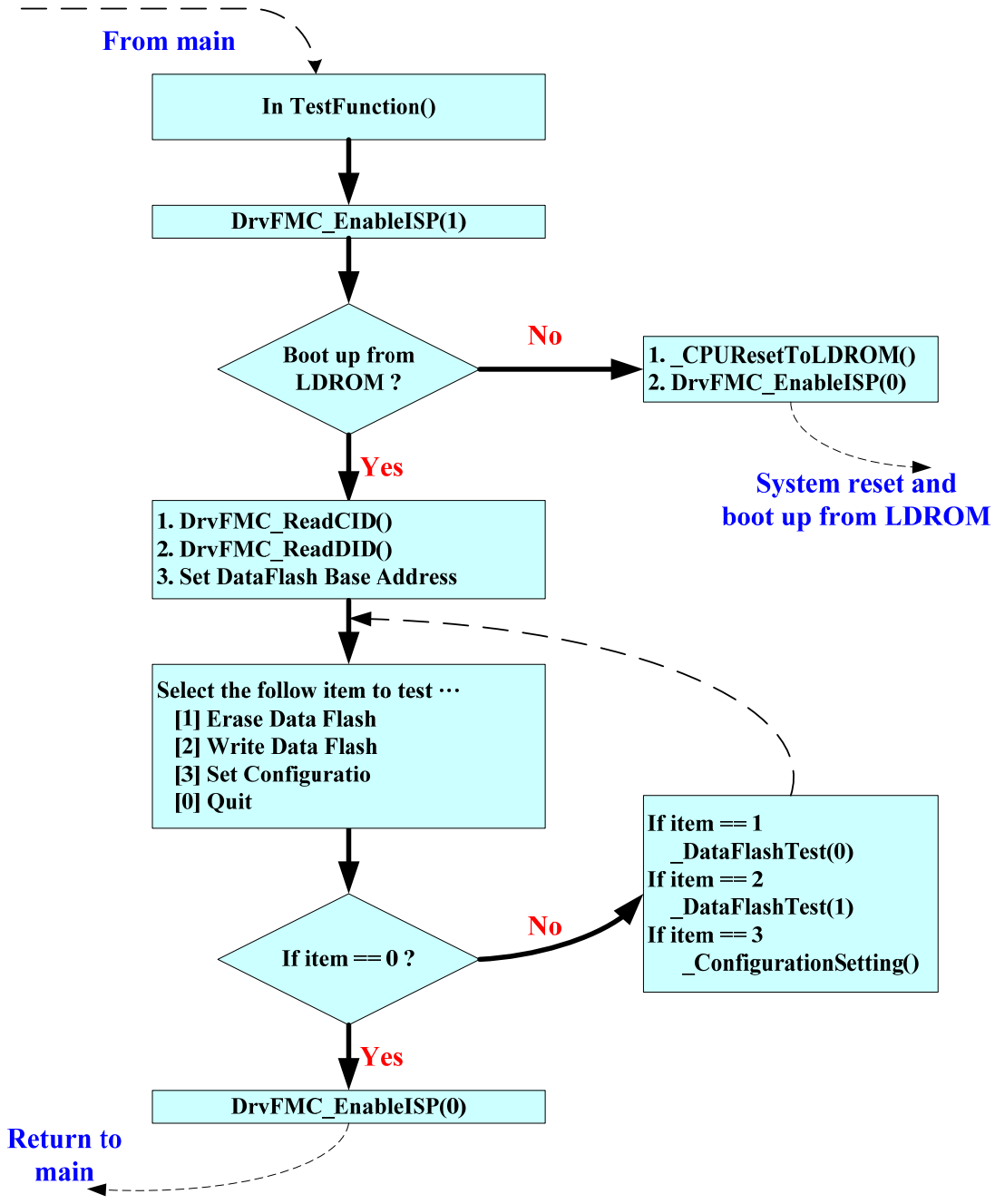
2.4 _ConfigurationSetting function

In _ConfigurationSetting function, user can select an item to configure the select configuration.

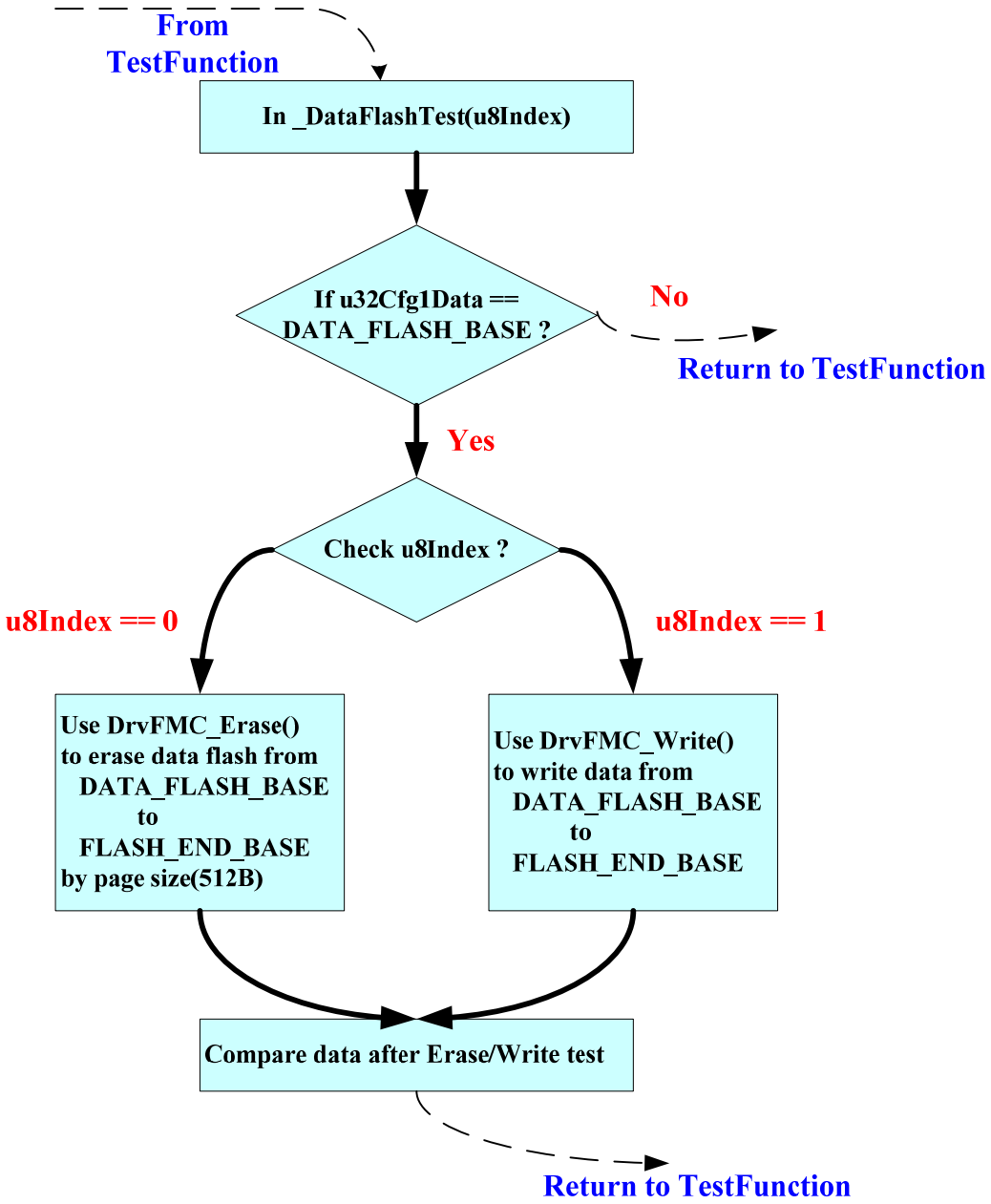
```
……
printf("| [1] Brown Out Level 2.2V    |\n");
printf("| [2] Brown Out Level 2.6V    |\n");
printf("| [3] Brown Out Level 3.8V    |\n");
printf("| [4] Brown Out Level 4.5V    |\n");
printf("| [5] Brown Out Reset %s |\n", (u32Cfg0Data&(1<<20))? "Enable ":"Disable");
printf("| [6] Set Bootup from %s   |\n", (u32Cfg0Data&(1<<7))? "LDROM":"APROM");
……
switch (u8Item)
{
        case '1':
        case '2':
        case '3':
        case '4':
                DrvFMC_Write(CFG0_BASE, (u32Cfg0Data&~(3<<21))|(u8Item-
0x31)<<21);
                break;
        case '5':
                u32Cfg0Data = (u32Cfg0Data&~(1<<20)) | (u32Cfg0Data&(1<<20)?
0:(1<<20));
                DrvFMC_Write(CFG0_BASE, u32Cfg0Data);
                break;
        case '6':
                u32Cfg0Data = (u32Cfg0Data&~(1<<7)) | (u32Cfg0Data&(1<<7)?
0:(1<<7));
                DrvFMC_Write(CFG0_BASE, u32Cfg0Data);
                printf("Power-on reset for the setting becomes effective !!!!\n\n" );
                break;
        default :
                continue;
 }
……
```

3 CALLING SEQUENCE

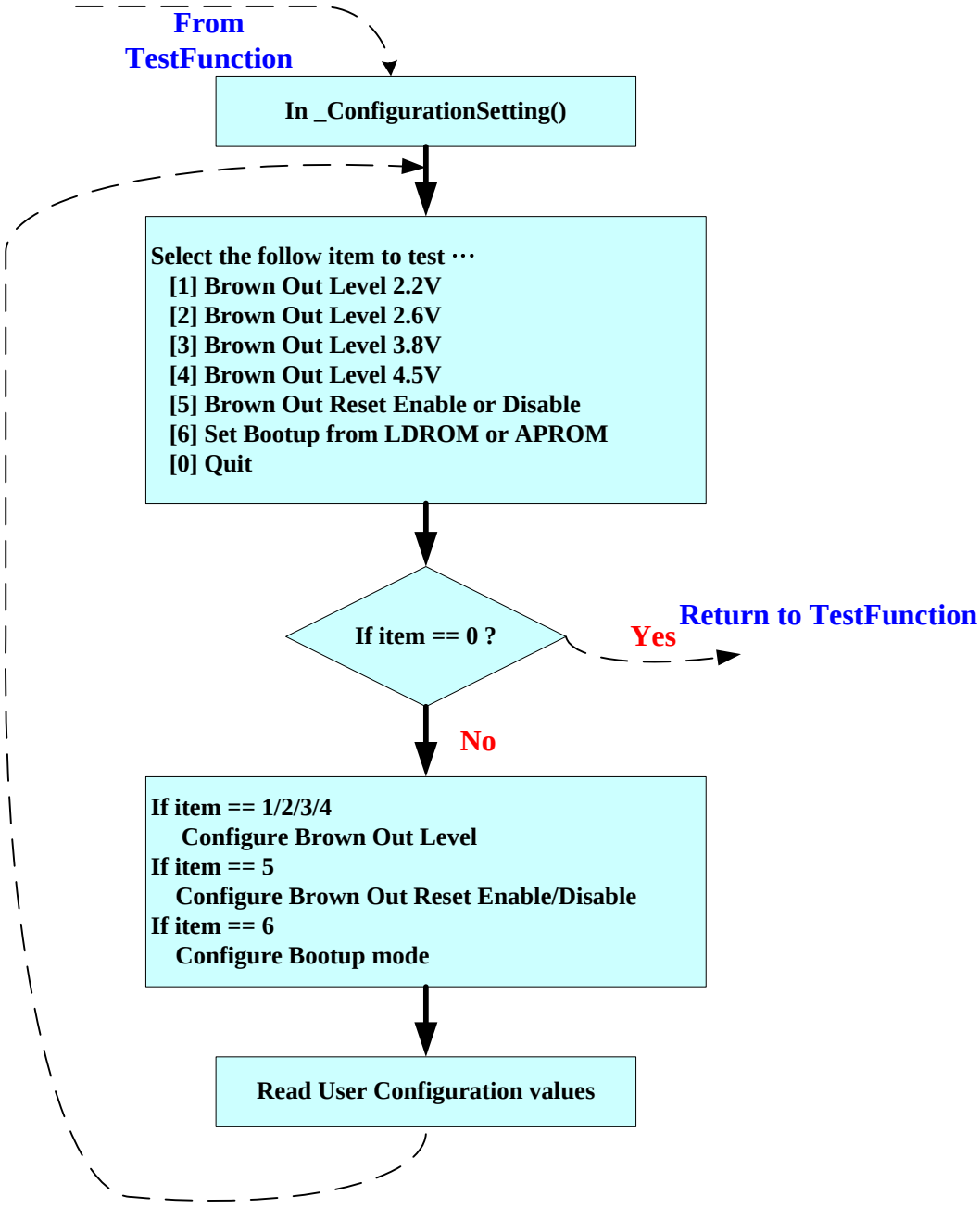
3.1 TestFunction



3.2 _DataFlashTest y Erase/Write Data Flash



3.3 _ConfigurationSetting by Read/Write User Configuration

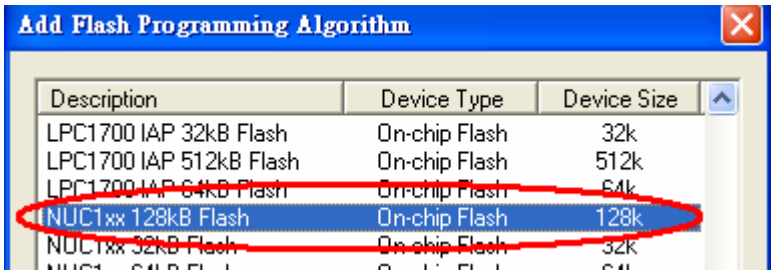


4 EXECUTION ENVIRONMENT SETUP AND RESULT

4.1 Image download sequence

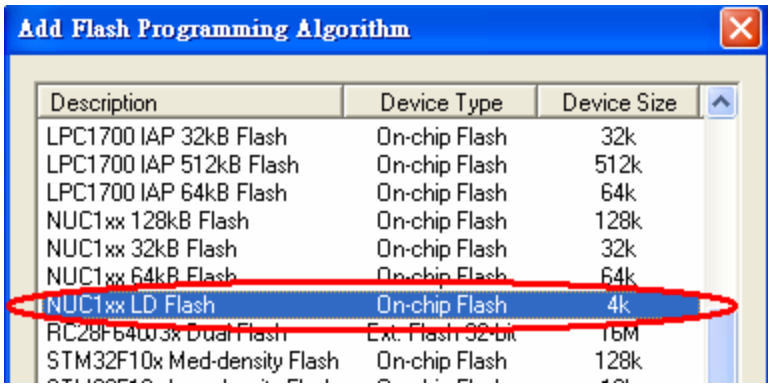
4.1.1 Download image file to APROM

We must download the Smpl_Configuration.bin to APROM as the below Programming Algorithm Setting firstly.



4.1.2 Download image file to LDROM

After **4.1.1**, download the Smpl_Configuration.bin to LDROM as the below Programming Algorithm Setting.



4.2 Execution result

After finished the steps **4.1.1** and **4.1.2**, we can reboot the system to do the detail test.

When power on, there are two conditions may occurred.

4.2.1 Boot up from APROM

Because default **CBS(Config Boot Selection)** is **1(Chip boot from APROM)**, and User Configuration only can be modified in LDROM. So we must reset system and boot up from LDROM. This is why we download the image to APROM as step **4.1.1**. It's guarantee the system can be boot up from LDROM.

4.2.2 Boot up from LDROM

If system is boot up from LDROM, user can test those various functions directly and get the results.

5 REVISION HISTORY

REV.	DATE	DESCRIPTION
1.00	March 09, 2010	1. Initially issued.

Important Notice

Nuvoton products are not designed, intended, authorized or warranted for use as components in systems or equipment intended for surgical implantation, atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, or for other applications intended to support or sustain life. Further more, Nuvoton products are not intended for applications wherein failure of Nuvoton products could result or lead to a situation wherein personal injury, death or severe property or environmental damage could occur.

Nuvoton customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nuvoton for any damages resulting from such improper use or sales.

Please note that all data and specifications are subject to change without notice. All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.