NXP Semiconductors
Application Note

Document Number: AN12255
Rev. 0, 09/2018

如何使用闪存重新映射功能

URL: https://www.nxp.com/docs/en/application-note/AN12255.pdf

1. 产品介绍

i.MXRT1060芯片支持闪存重新映射功能，允许用户将闪存地址重新映射到FlexSPI接口。

闪存重新映射功能为以下使用案例带来了好处：

- 要闪存多个固件；
- 在满足某条件时，切换其中一个固件以运行；
- 要更新无线应用程序中的固件（通常的过程是下载固件到闪存，执行有效性检查，然后切换到新的固件以运行。闪存重新映射功能可以帮助在固件位于XIPflash的任何地方直接运行固件。)

本文档介绍如何使用闪存重新映射功能。

目录

2. 产品概述

闪存重映射功能仅在XIPflash上工作，它通过FlexSPI与闪存接口。该函数可以将当前的闪存地址重新映射到预期的地址。

3. 如何使用闪存重新映射功能

闪存重新映射功能提供了三个寄存器，以设置开始、结束和偏移地址。设置该地址后，可将其重新映射到指定的闪存地址。

3.1. 闪存地址重新映射设置

闪存重新映射设置有三个寄存器。

表1 闪存地址重新映射设置

寄存器名称	描述
IOMUXC_GPR_GPR30	指定flexspi1 和flexspi2的起始地址
IOMUXC_GPR_GPR31	指定flexspi1 和flexspi2的结束地址
IOMUXC_GPR_GPR32	指定flexspi1 和flexspi2的偏移地址

当ADDR_START[31：12]≤Addr_i[31：12]<ADDR_END[31：12]时，重新映射的地址必须是Addr_o=Addr_i[31：12]+{偏移量[31：12]，12'h0}。否则，Addr_o=Addr_i，其中Addr_i表示原始访问地址，Addr_o表示重新映射的地址。

例如：

不要设置任何FlexSPI重新映射寄存器，那么它会获取相应访问地址的闪存内容。不要进行任何重新映射。

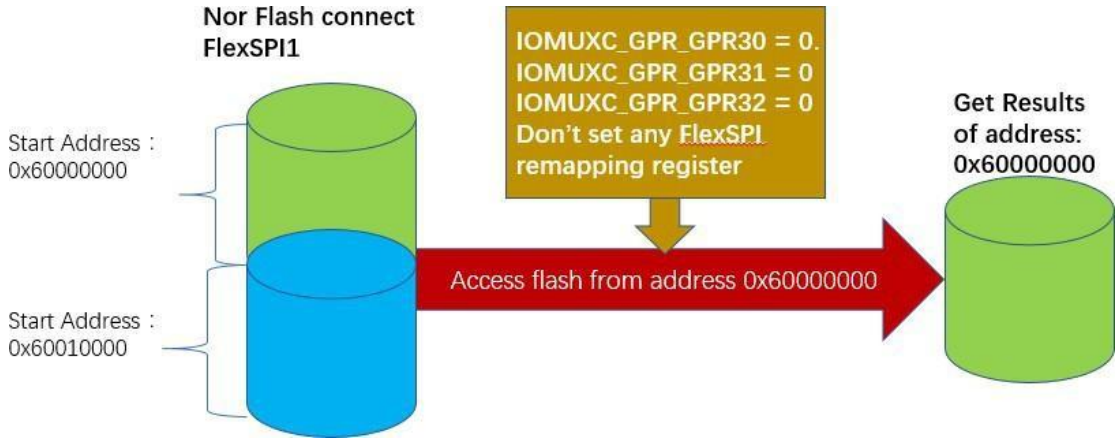

图1.不使用重新映射设置来访问FlexSPI

设置重新映射寄存器后，当尝试访问相同的闪存地址时，它将获取重新映射的地址内容。

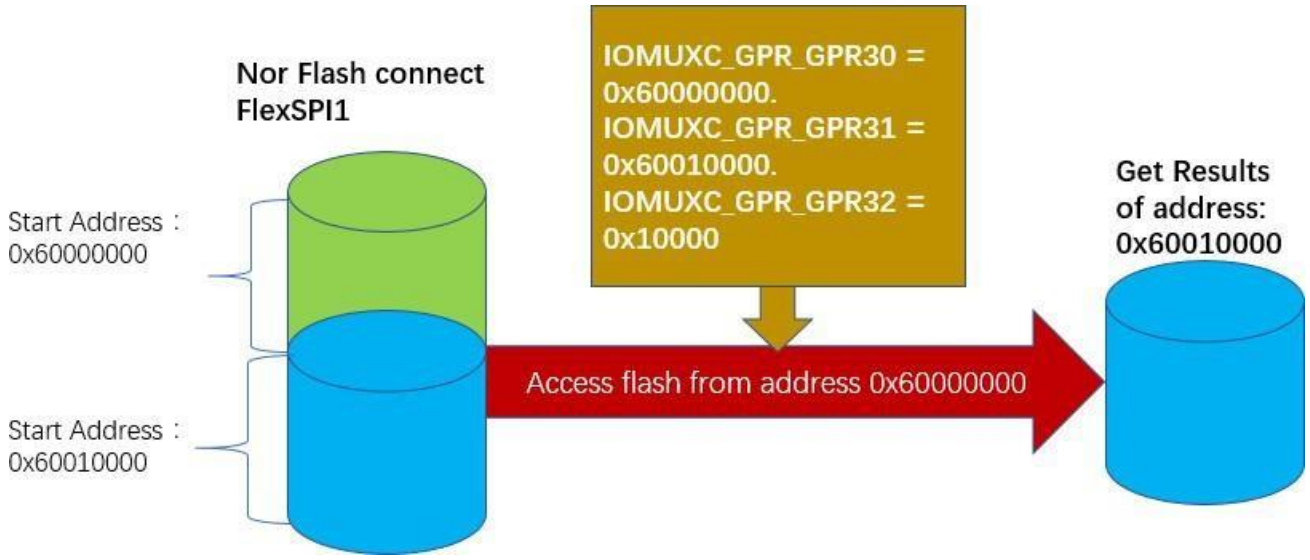

图2.使用重新映射的地址设置来访问FlexSPI

您可以从SW\src\boards\evkmimxrt1060\use_case\flash_remap_test\nor\polling_transfer\获得示例代码。在构建和运行项目后，图3显示了运行结果。

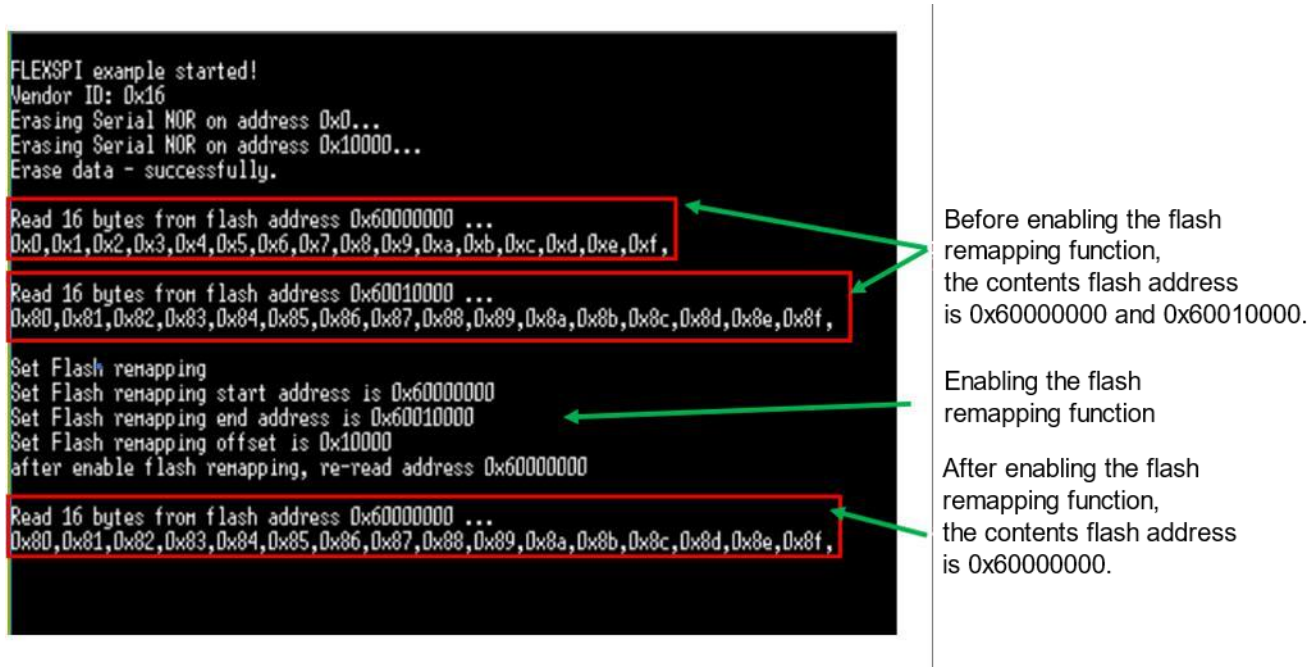

图3.闪存重新映射的运行结果

如图3所示，在读取相同的闪存地址（0x60000000）时，获取重映射的闪存地址内容（0x60001000），其中重映射的寄存器设置如下：

IOMUXC_GPR_GPR30 :0x60000000

IOMUXC_GPR_GPR31 :0x60010000

IOMUXC_GPR_GPR32 : 0x10000

3.2. ROM支持的闪存重新映射

通过fusion位的设置ROM支持闪存重新映射。

3.2.1. 闪存重新映射设置

ROM支持将两个固件下载到闪存上，并通过调用API函数轻松切换固件。要使能闪存重映射功能，可将以下fusion位熔断。

表1是FlexSPI重新映射设置的fusion位映射的一部分。

表2. FlexSPI1的fusion位定义

模块	地址	7	6	5	4	3	2	1	0
FlexSPI 1 - Serial NOR	**0x6E0[7:0]**	**FLEXSPI_RESET_PIN_EN** 0 - Disabled 1 - Enabled	**JEDEC_HW_RESET_EN** 0 - Disabled 1 - Enabled	**xSPI FLASH HOLD TIME** 0 – 500 us/1 - 1ms 2 – 3 ms/3 – 10 ms		**xSPI FLASH BOOT FREQUENCY** 0 – 100 MHz/1 – 120 MHz/2 – 133 MHz/3 - 166 MHz/4 - Reserved 5 – 80 MHz/6 – 60 MHz			SIP_TEST_EN
	0x6E0[15:8]	**xSPI FLASH IMAGE SIZE** 0-FLEXSPI_NOR_SEC_IMAGE_OFFSET*256 KB 1 - 12: 1 – 12 MB 13 – 256 KB, 14 - 512 KB, 15 - 768 KB				**xSPI FLASH DUMMY CYCLE** 0 - Auto probe Others - Dummy cycles (for example, 8 - 8 cycles)			
	0x6E0[23:16]	**FLEXSPI_NOR_SEC_IMAGE_OFFSET[7:0]** Actual offset = 256 KB * fuse value							

0x6E0[23：16]指定图像偏移量，当它不是0时，将使能闪存重新映射。例如，如果image大小约为512KB，则fusion位设置为：0x6E0[23：16]设置为2，0x6E0[15：12]设置为0。

ROM为用户提供了轻松切换固件的API函数

这是引导加载程序API入口结构体。

```
typedef struct
{
    const uint32_t version;                          //!< Bootloader version number const
    char *copyright;                                 //!< Bootloader Copyright
    void (*runBootloader)(void *arg);                //!< Function to start the bootloader executing const uint32_t
    *reserved0;                                      //!< Reserved
    const flexspi_nor_driver_interface_t *flexSpiNorDriver; //!< FlexSPI NOR Flash API const uint32_t *reserved1;
                                                     //!< Reserved
    const clock_driver_interface_t *clockDriver; const
    rtwdog_driver_interface_t *rtwdogDriver; const
    wdog_driver_interface_t *wdogDriver; const uint32_t
    *reserved2;
} bootloader_api_entry_t;
```

用户可以通过API入口地址0x0020001c调用这些API函数。

一个例子如下：

```
g_bootloaderTree=(bootloader_api_entry_t*)*(uint32_t*)0x0020001c；
```

引导加载器参数如下：

```
typedef union
{
    struct
    {
        uint32_t imageIndex : 4;
        uint32_t reserved : 12;
        uint32_t serialBootInterface : 4; uint32_t
        bootMode : 4;
        uint32_t tag : 8;
    } B;
    uint32_t U;
} run_bootloader_ctx_t;
```

imageIndex 定义了哪个image要被重新映射去运行。

其中一个例子如下：

```
run_bootloader_ctx_t boot_para;
boot_para.B.imageIndex = 1;                                    // specified firmware index to 1
boot_para.B.serialBootInterface = kEnterBootloader_SerialInterface_USB; boot_para.B.bootMode =
kEnterBootloader_Mode_Default;
boot_para.B.tag = kEnterBootloader_Tag;
g_bootloaderTree->runBootloader( (void *)&boot_para );          // run the index 1 firmware
```

用户可以轻松地更改固件索引以切换固件。

3.2.2. 通过MFGTool烧录多个固件到闪存中

NXP提供了烧录工具，MFGTOOL，可在以下链接获得：
https://www.nxp.com/webapp/Download?colCode=FLASHLOADER-RT106x-1-GA&appType=license&Parent_nodeId=1517584717166704362897&Parent_pageType=product https://cache.nxp.com.cn/secured/assets/downloads/en/programmers/FLASHLOADER-RT106x-1-GA.zip?__gda__=1631531462_ac7044f27fff31d351bffedc8da4995e&fileExt=.zip

使用MFGTOOL，您可以通过USB接口下载固件到闪存中。详情请参见AN12108。

按照AN12108中所示的指南，MFGTOOL工具可以帮助将第一个固件下载到闪存中。

对于第二个固件，您需要按如下所述修改program_flexspinor_image_qspinor.bd文件。

1. 更改程序地址。第二个固件应该重新映射到地址0x60200000，如果固件重新映射到其他地址，则需要相应地修改地址值。

```
constants {
    kAbsAddr_Start= 0x60200000;
    kAbsAddr_Ivt = 0x60201000;
    kAbsAddr_App = 0x60202000;
}
```

2. 擦除第二个固件的闪存地址空间。

```
#2 Erase flash as needed. erase
0x60200000..0x60400000;
```

3. 注释出“程序配置块”，因为闪存配置块默认地址为0x60000000，它不需要为第二个固件重新编程，也可以通过将BOOT_CFG[0]设置为1来启用自动探测。

```
#3. Program config block
# 0xf000000f is the tag to notify Flashloader to program FlexSPI NOR config block to the start of device
#load 0xf000000f > 0x3000;
# Notify Flashloader to response the option at address 0x3000 #enable flexspinor
0x3000;
```

4. 用于重新映射设置的fusion编程。

   将以下命令添加到bd文件中，用于保险编程。

```
#7. set offset address8*256K load
fuse 0x00080000 > 0x2e;
```

您可以按照上述步骤修改bd文件，也可以直接使用附带的bd文件 program_flexspinor_image_qspiflash_offset_2M.bd，生成二级固件的sb文件。 附加的软件包包含用于测试的s-record文件，其中firmware_swap1.srec用于固件1，firmware_swap2.srec用于固件2。

您可以在*SW\src\boards\evkmimxrt1060\use_case\firmware_swap* 获得源码。

在这两个固件被烧录到闪存后，运行第一个固件并输入字符y来切换另一个固件。

图4显示了正在运行的结果。

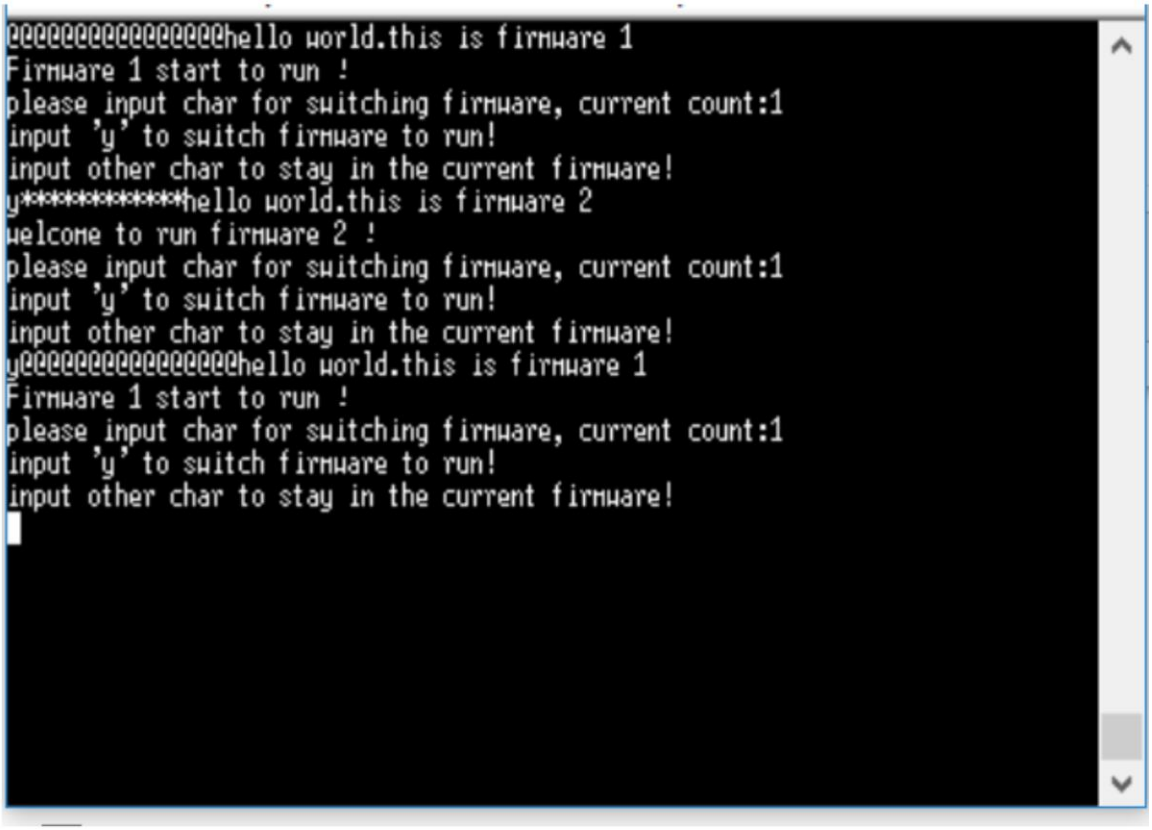

图4.固件交换的运行结果

4. 结论

本文档介绍了如何在i.MXRT1060上使用闪存重新映射功能。闪存重新映射功能给切换固件的应用带来了好处，特别是通过oOn The Air(OTA)的固件升级。该功能使用方便，可提供高可靠性的能力。

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support

Document Number: AN12255
Rev.0
09/2018